การให้คะแนนความน่าเชื่อถือด้วยวิธี Support Vector Machine โดยอาศัยข้อมูลจากสถาบันการเงินไทย

CREDIT SCORING WITH A SUPPORT VECTOR MACHINE BASED ON THE DATA OF THAI FINANCIAL INSTITUTE

อุษณี วรชาติเดชชัย

USANEE WORRACHARTDATCHAI

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมสารสนเทศ

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2550

การให้คะแนนความน่าเชื่อถือด้วยวิธี Support Vector Machine โดยอาศัยข้อมูลจากสถาบันการเงินไทย

CREDIT SCORING WITH A SUPPORT VECTOR MACHINE BASED ON THE DATA OF THAI FINANCIAL INSTITUTE

อุษณี วรชาติเดชชัย

USANEE WORRACHARTDATCHAI

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมสารสนเทศ

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2550

KMITL-2007-EN-M-230-056

CREDIT SCORING WITH A SUPPORT VECTOR MACHINE BASED ON THE DATA OF THAI FINANCIAL INSTITUTE

USANEE WORRACHARTDATCHAI

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
MASTER OF ENGINEERING IN INFORMATION ENGINEERING
SCHOOL OF GRADUATE STUDIES
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
2007

KMITL-2007-EN-M-230-056

หัวข้อวิทยานิพนธ์	การให้คะแนนความน่าเชื่อถือด้วยวิธี Support Vector Machine โดยอาศัยข้อมูลจากสถาบันการเงินไทย
นักศึกษา	นางสาวอุษณี วรชาติเดชชัย
รหัสนักศึกษา	47061119
ปริญญา	วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชา	วิศวกรรมสารสนเทศ
พ.ศ.	2550
อาจารย์ที่ปรึกษาวิทยานิพนธ์	รศ.ดร.ปิติเขต สู้รักษา

บทคัดย่อ

วิทยานิพนธ์ฉบับนี้ เสนอการให้คะแนนความน่าเชื่อถือของลูกค้าจากสถาบันการเงินแห่งประเทศไทยด้วยแบบจำลองที่สร้างด้วยหลักการของซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine) และ ลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีน (Least Square Support Vector Machine: LS-SVM) ซึ่งการพยากรณ์คะแนนความน่าเชื่อถือลูกค้าใหม่ของสถาบันการเงินนั้น เป็นวิธีที่จำเป็นสำหรับการจัดลำดับความน่าเชื่อถือ ในการแบ่งแยกประเภทของลูกค้าหรือสิ่งต่าง ๆ ตามลำดับกลุ่มของความเสี่ยง

ในปัจจุบันมีวิธีแตกต่างกันมากมาย ที่ถูกนำมาสร้างแบบจำลองในการจัดประเภทลูกค้าเพื่อประเมินค่าความเสี่ยงและผลกำไรในการให้เงินทุนกับลูกค้า ในข้อเสนอวิทยานิพนธ์นี้ได้เลือกหลักการของซัพพอร์ตเวกเตอร์แมชชีน โดยเปรียบเทียบผลกับหลักการของลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีน มาใช้ในการทดลอง ซึ่งหลักการของซัพพอร์ตเวกเตอร์แมชชีน เป็นวิธีที่ค่อนข้างใหม่และกำลังได้รับความนิยมในการสร้างแบบจำลองเพื่อแยกประเภทของลูกค้า ส่วนหลักการของลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีนเป็นการนำสมการของซัพพอร์ตเวกเตอร์แมชชีน มาเขียนใหม่หรือการนำสมการซัพพอร์ตเวกเตอร์แมชชีน มาดัดแปลงเพื่อให้ได้ค่าที่ละเอียด และแม่นยำมากยิ่งขึ้น หลักการของลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีนนั้นใกล้เคียงกับกฎเกณฑ์ของระบบโครงข่ายและวิธีการเกาส์เซียน (Gaussian) เป็นอย่างมากแต่ได้มีการเน้นเพิ่มเกี่ยวกับการแสดงค่าของข้อมูลให้มีความละเอียดมากขึ้น ทำให้การที่นำหลักการลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีนมาใช้ในการแยกประเภทข้อมูลนั้น สามารถที่จะทำให้แบ่งประเภทข้อมูลออกเป็นหลายประเภทได้โดยมีความละเอียดมากยิ่งขึ้น โดยแบบจำลองที่ดีที่สุดที่หาได้จะสามารถทำการแบ่งประเภทของลูกค้าออกเป็นสี่กลุ่มด้วยกันคือ

1. ดีมาก
2. ค่อนข้างดี
3. อาจจะไม่ดี (เสี่ยง)
4. ไม่ดี

Thesis Title	Credit Scoring with a Support Vector Machine Based on the Data of Thai Financial Institute
Student	Miss Usanee Worrachartdatchai
Student ID.	47061119
Degree	Master of Engineering
Program	Information Engineering
Year	2005
Thesis Advisor	Assoc. Prof. Dr. Pitikhate Sooraksa

ABSTRACT

This thesis proposes credit scoring by means of Support Vector machine and Least Square Support Vector Machine based on the data of Thai financial institute. The quantitative method known as credit scoring has been developed for the credit assessment problem. Credit scoring is essentially an application of classification techniques, which classify credit customers into different risk groups. The Financial institutions are being more and more obliged to build credit scoring models assessing the risk of default of their clients. Support Vector Machine is a promising new technique that has recently emanated and become popular for data classification. Least Squares Support Vector Machines are re-formulations to the standard Support Vector Machine. The cost function is a regularized least squares function with equality constraints. The solution can be found efficiently by iterative method like the Gaussian algorithm. Then in this thesis, both Support Vector Machine and Least Squares Support Vector Machine are considered by applying to the credit scoring with the data of Thai financial institutions. The optimum model will be able to divide the group of customers into four groups as the following:

1. Very good
2. Rather good
3. Suspiciously bad
4. Very bad.

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยดี เนื่องจากได้รับความอนุเคราะห์ ความช่วยเหลือ ให้ความรู้ คำแนะนำ คำปรึกษา แนวทางในการวิจัย และแนวทางการแก้ไขปัญหาต่าง ๆ เป็นอย่างดี จาก รศ.ดร.ปิติเขต สู้รักษา อาจารย์ผู้ควบคุมวิทยานิพนธ์

ขอขอบพระคุณท่านกรรมการสอบทุกท่าน ที่ได้เสียสละเวลาอันมีค่าในการตรวจสอบวิทยานิพนธ์ในครั้งนี้ และช่วยแก้ไขวิทยานิพนธ์ฉบับนี้ให้สมบูรณ์ยิ่งขึ้น

ขอขอบคุณบัณฑิตศึกษาและบัณฑิตวิทยาลัย คณะวิศวกรรมศาสตร์ที่ให้ความช่วยเหลือในเรื่องต่าง ๆ รวมถึงให้ทุนสนับสนุนการทำวิทยานิพนธ์

ขอขอบคุณหยี่อี้ อาเตี๋ยว มาม๊า และครอบครัวที่คอยดูแลเอาใจใส่ ให้ความช่วยเหลือ และเป็นกำลังใจที่ดีมาโดยตลอด

ขอขอบคุณพี่ ๆ ทุกคน รวมถึงหัวหน้าที่บริษัทคิมเบอร์ลี่ คล๊าก ที่ให้ความสะดวกในการไปเรียนในวันที่ต้องทำงาน รวมถึงเอาใจใส่สอบถาม และเป็นกำลังใจให้ตลอดมา

สุดท้ายต้องขอขอบคุณตั้มที่เป็นทั้งผู้ผลักดันให้มีจุดเริ่มต้นที่ดี ขอบคุณที่คอยให้กำลังใจเวลาท้อ และคอยกระตุ้นให้ตั้งใจเรียน ให้ตั้งใจทำวิทยานิพนธ์อย่างสม่ำเสมอจนสามารถสำเร็จลุล่วงไปด้วยดี

คุณค่าและประโยชน์อันพึงมีจากวิทยานิพนธ์ฉบับนี้ ผู้วิจัยขอบมอบแด่ผู้มีพระคุณทุกท่าน

อุษณี วรชาติเดชชัย

สารบัญ

หน้า

สารบัญ (ต่อ)

หน้า

สารบัญตาราง

สารบัญรูป

สารบัญรูป (ต่อ)

รูปที่ หน้า

สารบัญรูป (ต่อ)

รูปที่ หน้า

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ณ สภาวะปัจจุบันในประเทศไทย ประชากรเป็นจำนวนมากได้มีส่วนเกี่ยวข้องในการกู้ยืมเงินจากสถาบันการเงินต่าง ๆ เพื่อใช้ในการลงทุนหรือใช้ในการดำเนินชีวิตประจำวัน และจากการสำรวจได้พบว่าส่วนใหญ่สถาบันการเงินได้ใช้ทรัพยากรมนุษย์ในการคิดและตัดสินใจในการอนุมัติวงเงินให้กับลูกค้า ซึ่งเป็นหนึ่งในปัญหาของการทำงานคือ พนักงานจะต้องใช้เวลาในการคิดและการตัดสินใจค่อนข้างมากต่อลูกค้าหนึ่งราย ดังนั้นเมื่อได้ทำการเทียบกับความต้องการของลูกค้าในสถานการณ์ปัจจุบันแล้วจะเห็นว่า ทรัพยากรมนุษย์ที่นำมาใช้ทำงานไม่เพียงพอต่อความต้องการของลูกค้า นอกจากนั้นยังมีความเสี่ยงต่อการตัดสินใจผิดพลาดอีกด้วย

ดังนั้นนักวิจัยจำนวนมากได้มุ่งความสนใจเพื่อที่จะทำการพัฒนาระบบคอมพิวเตอร์เพื่อมาช่วยตัดสินใจในการจัดแบ่งประเภทของลูกค้าเพื่อให้มีความสะดวก รวดเร็ว และมีความถูกต้องในการอนุมัติวงเงินให้กับลูกค้ามากยิ่งขึ้น

ปัญหาที่นำมาสู่การวิจัยต่อการแบ่งกลุ่มประเภทลูกค้าที่เป็นที่สนใจอยู่ในปัจจุบันนั้น คือ ปัญหาทางการเพิ่มความถูกต้องและความละเอียดในการแบ่งประเภทของลูกค้า เพื่อที่จะได้ทำการอนุมัติวงเงินกู้ของสถาบันการเงินให้กับลูกค้าในปัจจุบันได้อย่างมีประสิทธิภาพมากขึ้น

1.1.1 เพราะเหตุใดปัญหานี้ถึงน่าสนใจ

จากที่ได้กล่าวในข้างต้นมาแล้วว่า สถาบันการเงินส่วนใหญ่ในประเทศไทย ยังคงใช้ทรัพยากรมนุษย์ในการตัดสินใจในการอนุมัติวงเงินให้กับลูกค้า ดังนั้นการนำระบบการตัดสินใจด้วยคอมพิวเตอร์ที่ใช้วิธีการตัดสินใจจากสถิติที่ได้มีการเก็บข้อมูลไว้ มาเป็นตัวช่วยหลักในการตัดสินใจ มาช่วยจัดว่าเป็นงานที่น่าสนใจและมีประโยชน์ในการพัฒนาเทคโนโลยีในประเทศไทยให้มีความก้าวหน้ายิ่งขึ้นดังนี้

1. เป็นงานวิจัยที่ยังไม่มีผู้ใดนำมาปรับเปลี่ยนใช้ ให้เข้ากับระบบการตัดสินใจประเทศไทยมาก่อนจากการค้นหาและทบทวนเอกสารงานวิจัยที่เกี่ยวข้องทั้งหมด
2. ผลงานวิจัยที่ได้ จะช่วยลดเวลาการทำงานแต่เพิ่มความมีประสิทธิภาพและความถูกต้องต่อการตัดสินใจมากขึ้น
3. ผลงานวิจัยที่ได้จะช่วยสร้างความมั่นใจ ความยุติธรรม ในการอนุมัติวงเงินให้ลูกค้ามากยิ่งขึ้น
4. สามารถนำข้อมูลที่มีอยู่แล้วมาใช้ให้เกิดประโยชน์

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

จากการที่สังคมในยุคปัจจุบันมีการแข่งขันกันสูงในทุก ๆ ด้าน ไม่ว่าจะเป็นการแข่งขันทางเศรษฐกิจ การแข่งขันในสถาบันการเงินถือเป็นองค์กรหนึ่งที่มีการเก็บข้อมูลเกี่ยวกับลูกค้าที่เข้ามาใช้ในสถาบันการเงินอยู่เป็นจำนวนมาก โดยที่ข้อมูลเหล่านี้สามารถนำมาใช้ประโยชน์ได้มากแต่ไม่ได้ถูกนำมาใช้งานให้เป็นประโยชน์อย่างจริงจัง งานวิจัยนี้จึงมีวัตถุประสงค์เพื่อสร้างแบบจำลองในการให้คะแนนความน่าเชื่อถือของลูกค้าจากสถาบันการเงินในประเทศไทย โดยอาศัยวิธีการทำงานของซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine : SVM) มาเป็นตัวการจำแนกประเภทของลูกค้านี้ โดยเปรียบเทียบกับหลักการของลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีน (Least Square Support Vector Machine : LS-SVM) ซึ่งจะเห็นได้ว่าหลักการใดสามารถทำให้ได้ผลลัพธ์ที่มีความละเอียดมากกว่ากัน และหลักการของลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีนยังสามารถทำให้แบ่งประเภทของข้อมูลได้มากกว่าสองประเภท ซึ่งในหลักการของซัพพอร์ตเวกเตอร์แมชชีนจะสามารถแบ่งข้อมูลได้เพียงสองประเภทเท่านั้น โดยที่ทั้งสองหลักการจะใช้ค่าจากคุณสมบัติของลูกค้า (Attribute) ซึ่งเป็นปัจจัยหลักที่มีผลต่อความน่าเชื่อถือต่อสถาบันการเงินที่จะอนุมัติเงินกู้ให้ เมื่อสามารถหาแบบจำลองที่มีความน่าเชื่อถือสูงได้แล้วนั้น จึงสามารถที่จะนำข้อมูลของลูกค้าใหม่มาทำการทดสอบโดยผ่านแบบจำลองนี้ ในที่สุดก็จะสามารถทำนายได้ว่าลูกค้าคนนี้ควรจะอยู่ในประเภทลูกค้ากลุ่มไหนและมีความน่าเชื่อถือมากเพียงใด เพื่อที่จะลดอัตราความเสี่ยงต่อสถาบันการเงินได้

จากการทบทวนเอกสารและงานวิจัยที่เกี่ยวข้องพบว่า ยังไม่มีผู้ใดเคยเสนอวิธีซัพพอร์ตเวกเตอร์แมชชีนมาใช้เป็นวิธีการตัดสินใจ แบ่งประเภทของลูกค้า เพื่อที่จะอนุมัติวงเงินกู้ ในสถาบันการเงินของประเทศไทยมาก่อน ถึงแม้ว่าจะมีการนำหลักการซัพพอร์ตเวกเตอร์แมชชีนไปใช้ในประเทศอื่น ๆ เช่น เยอรมัน ได้หวัน จีน แต่จากการค้นคว้าแล้วพบว่า เนื่องจากหลักการในการเก็บข้อมูลของลูกค้า การคำนวณ การตัดสินในแต่ละประเทศนั้นมีต่าง ๆ กันไป ทำให้ส่งผลโดยต้องทำการปรับเปลี่ยนระบบปฏิบัติการของโปรแกรมออกไปด้วย หรือจะกล่าวอีกนัยหนึ่งได้ว่าไม่สามารถนำเอาโปรแกรมที่ใช้งานกับข้อมูลลูกค้าประเทศหนึ่ง มาใช้กับข้อมูลลูกค้าของอีกประเทศหนึ่งได้ อีกทั้งงานวิจัยนี้ยังได้มีการนำหลักการของลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีนมาเปรียบเทียบกันเพื่อความแม่นยำถูกต้องมากขึ้นอีกด้วย ทำให้งานวิจัยนี้อาจนับได้ว่าเป็นชิ้นแรกที่นำหลักการเหล่านี้มาประยุกต์ให้ใช้งานได้กับข้อมูลลูกค้าในประเทศไทยได้จริง

การประยุกต์ใช้การเรียนรู้ ด้วยวิธีซัพพอร์ตเวกเตอร์แมชชีน กับการให้คะแนนความน่าเชื่อถือของลูกค้าจากสถาบันการเงินมีวัตถุประสงค์ดังนี้

1. เพื่อใช้สร้างแบบจำลองในการทำนายความน่าเชื่อถือของลูกค้า จากสถาบันการเงินในประเทศไทย

2. เพื่อวิเคราะห์หาคุณสมบัติหรือตัวแปร (Parameter) ที่มีผลความน่าเชื่อถือของลูกค้าจากสถาบันการเงินของประเทศไทย
3. เพื่อให้สามารถนำแบบจำลองที่ได้ มาจัดแบ่งกลุ่มประเภทของลูกค้า ออกเป็นสี่ประเภท
4. เพื่อเป็นอีกทางเลือกหนึ่งที่น่าสนใจให้กับสถาบันการเงินในประเทศไทย

1.3 สมมติฐานของการศึกษา

หลักการพื้นฐานของซัพพอร์ตเวกเตอร์แมชชีนนั้น สามารถนำมาใช้แก้ปัญหาด้วยแบบที่เป็นเชิงเส้นและไม่เป็นเชิงเส้น ด้วยรูปแบบสมการพีชคณิตที่มีกำลังสอง (Quadratic programming) สำหรับวิธี ลีซท์ สแควร์ ซัพพอร์ตเวกเตอร์แมชชีน นั้นจะสามารถที่จะแก้ปัญหาที่เป็นแบบเชิงเส้นและไม่เป็นเชิงเส้นด้วยข้อจำกัดของสมการที่เท่ากับและไม่เท่ากับได้อีกด้วย ดังนั้นการแบ่งประเภทข้อมูลด้วยวิธี ลีซท์ สแควร์ ซัพพอร์ตเวกเตอร์แมชชีน สามารถที่จะทำให้ได้ค่าที่ละเอียดมากยิ่งขึ้นและมีความน่าเชื่อถือมากขึ้น ดังนั้นทำให้ผู้วิจัยสามารถที่จะนำทั้งหลักการของ ลีซท์ สแควร์ ซัพพอร์ตเวกเตอร์แมชชีน มาใช้ในการแบ่งประเภทข้อมูลได้มากกว่าสองประเภทด้วยกัน ด้วยผลลัพธ์ที่มีค่าความละเอียดและความถูกต้องมากขึ้นกว่าหลักการของซัพพอร์ตเวกเตอร์แมชชีน

1.4 ทฤษฎีหรือแนวคิดที่ใช้ในการวิจัย

จากข้อดีของการสร้างแบบจำลอง เพื่อการทำนายผลการให้คะแนนความน่าเชื่อถือของลูกค้าจากสถาบันทางการเงินนั้น จึงเป็นที่มาของงานวิจัยนี้ว่า ทำอย่างไร ถึงจะสามารถทำนายความน่าเชื่อถือของลูกค้าใหม่ของสถาบันการเงินในประเทศไทยได้ และยังสามารถใช้ประโยชน์จากฐานข้อมูลที่มีอยู่ให้เกิดประโยชน์ได้มากที่สุด ดังนั้นผู้วิจัยจึงได้เสนอแนวความคิดในการสร้างแบบจำลองในการให้ให้คะแนนความน่าเชื่อถือของลูกค้าโดยใช้อัลกอลิธึม (Algorithm) ของซัพพอร์ตเวกเตอร์แมชชีน ซึ่งเป็นทฤษฎีที่น่าสนใจของงานประเภทการจัดหมวดหมู่ข้อมูล โดยที่อัลกอริธึม ซัพพอร์ตเวกเตอร์แมชชีน นี้ได้ถูกพัฒนามาจากวิธีพื้นฐานของระบบโครงข่ายประสาทเทียม (Neural Network)

ซัพพอร์ตเวกเตอร์แมชชีน เป็นวิธีหนึ่งที่มีประสิทธิภาพสำหรับการแก้ปัญหาเกี่ยวกับการจัดเรียงประเภทข้อมูลแบบที่ไม่เป็นเชิงเส้น เกี่ยวกับฟังก์ชันที่ไว้ใช้ในการคาดคะเนค่าต่าง ๆ โดยนำหลักการเคอร์เนล (Kernel) มาใช้ในการพัฒนาประสิทธิภาพของโปรแกรมโดยทั่วไป ซึ่งแต่เดิมมีการนำหลักการซัพพอร์ตเวกเตอร์แมชชีน มาใช้เกี่ยวกับอรรถธิบายของนิยามการเรียนรู้ระบบสถิติและโครงสร้างที่มีความเสี่ยงน้อยที่สุด ด้วยวิธีนี้สามารถนำมาใช้แก้ปัญหาที่ดีที่สุดได้ โดยที่ลีซท์สแควร์ ซัพพอร์ตเวกเตอร์แมชชีน เป็นการนำสมการของ ซัพพอร์ตเวกเตอร์แมชชีน มาเขียนใหม่หรือการนำสมการ ซัพพอร์ตเวกเตอร์แมชชีน มาดัดแปลงเพื่อให้ได้ค่าที่ละเอียดและแม่นยำมาก

ยิ่งขึ้นในเงื่อนไขเคเคที (Karush-Kuhn-Tucker : KKT) ซึ่งจะกล่าวถึงมากขึ้นในบทถัดไป โดยที่หลักการของ ลีซท์ สแควร์ ซัพพอร์ตเวกเตอร์แมชชีนนั้น ใกล้เคียงกับกฎเกณฑ์ของระบบโครงข่ายและวิธีการเกาส์เซียน (Gaussian) เป็นอย่างมากแต่ได้มีการเน้นเพิ่มเกี่ยวกับการแสดงค่าของข้อมูลให้มีความละเอียดมากขึ้น ทำให้การที่นำ ลีซท์ สแควร์ ซัพพอร์ตเวกเตอร์แมชชีน มาใช้ในการแยกประเภทข้อมูลนั้น สามารถที่จะขยายไปทำการแก้ปัญหาการแบ่งประเภทข้อมูลออกเป็นหลาย ๆ ประเภทได้ละเอียดยิ่งขึ้นได้ นอกจากค่าเอาท์พุท (Output) y หนึ่งค่าแล้ว จะยังกล่าวถึงค่าเอาท์พุทหลาย ๆ ค่า $y^{(i)}$ ได้โดยที่ $i = 1,...,n_y$ และ n_y เป็นจำนวนของค่าเอาท์พุทในส่วนของการเขียนโปรแกรมเพื่อเลือกค่าเอาท์พุทนั้นเป็นส่วนที่สำคัญ ซึ่งโดยหลักแล้วมีความเป็นไปได้ที่จะทำการแปลงข้อมูลเป็นรหัส (Encode) 2^{n_y} ประเภทด้วยกันโดยหมายความถึงจำนวนของ n_y เอาท์พุท แต่ปกติแล้ววิธีนี้ไม่ได้เป็นวิธีที่ดีที่สุด ส่วนวิธีที่ดีที่สุดคือการนำเอาท์พุทหลาย ๆ ค่า มาเป็นจำนวนของประเภทของข้อมูลซึ่งจะกล่าวต่อไปอย่างละเอียดในบทที่สอง

1.5 การเปรียบเทียบระหว่างวิธีการที่นำเสนอกับวิธีการแบบพื้นฐาน

จากการค้นคว้าแล้วได้ทราบว่าทางต่างประเทศ [5] ได้นำวิธี ซัพพอร์ตเวกเตอร์แมชชีน มาใช้ในการคำนวณโดยจะแบ่งประเภทของลูกค้าออกเป็นสองประเภทเท่านั้นคือ อนุมัติ กับ ไม่อนุมัติ ซึ่งจะเห็นได้ว่าขอบเขตการทำงานจะค่อนข้างกว้างเกินกว่าจะนำมาใช้งานจริงได้ เพราะในความเป็นจริงแล้วประเภทของลูกค้าควรจะแตกย่อยออกได้มากกว่าสองประเภท แต่วิธีนี้ผู้วิจัยได้เสนอและแสดงให้เห็นว่า สามารถทำการแบ่งประเภทข้อมูลลูกค้าได้มากกว่าสองประเภท และสามารถนำมาประยุกต์ใช้กับระบบข้อมูลของประเทศไทยได้จริง

1.6 ขอบเขตการวิจัย

ในวิทยานิพนธ์ฉบับนี้ ได้นำเสนอวิธีการสร้างแบบจำลองด้วยวิธีซัพพอร์ตเวกเตอร์แมชชีน และ ลีซท์ สแควร์ ซัพพอร์ตเวกเตอร์แมชชีน ที่นำมาใช้ในการแยกประเภทกลุ่มลูกค้าของสถาบันการเงินในประเทศไทย เปรียบเทียบกับวิธีการแบบพื้นฐานซัพพอร์ตเวกเตอร์แมชชีน โดยใช้คอมพิวเตอร์ในการสร้างแบบจำลองด้วยโปรแกรม MATLAB [20] ซึ่งผลที่ได้นั้นแสดงถึงประสิทธิภาพและความน่าเชื่อถือ ของแบบจำลองในการแยกประเภทกลุ่มของลูกค้าในสถาบันการเงินของประเทศไทย

1.7 ขั้นตอนของการศึกษา

วิทยานิพนธ์ฉบับนี้ได้แบ่งเนื้อหาออกเป็น 5 บทด้วยกันคือ

บทที่ 1 กล่าวถึงความเป็นมาของงานวิจัย ความมุ่งหมายและวัตถุประสงค์ สมมติฐาน ทฤษฎีที่ใช้ ขอบเขตของการวิจัย และขั้นตอนการศึกษา

บทที่ 2 กล่าวถึงทฤษฎีพื้นฐานที่ใช้ในการวิจัย หลักการพยากรณ์และพื้นฐานของระบบโครงข่ายประสาทเทียม หลักการพื้นฐานของ ซัพพอร์ตเวกเตอร์แมชชีน และ ลีซท์ สแควว ซัพพอร์ตเวกเตอร์แมชชีน

บทที่ 3 กล่าวถึงวิธีดำเนินการวิจัย ซึ่งรวมถึงการแปลงรูปแบบของข้อมูล การกำหนดสัดส่วนขอบเขต (Scaling) ของข้อมูล การหาค่าตัวแปรให้ได้ค่าที่ดีที่สุดด้วยวิธีของคลอส วาริเดชัน (Cross Validation) และ กริดเสิร์จ (Grid Search) การฝึกฝน (Training) ข้อมูล และสุดท้ายคือวิธีการทดสอบ (Testing) ข้อมูล

บทที่ 4 กล่าวถึงผลการทดลองที่ได้รับจากหลักการของทั้งซัพพอร์ตเวกเตอร์แมชชีนและลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีน จากนั้นจะกล่าวถึงการประเมินหาประสิทธิภาพของแบบจำลองที่ได้มาจากหลักการทั้งสอง เพื่อแสดงให้เห็นว่าวิธีการที่นำเสนอนั้นสามารถที่ช่วยให้แบบจำลองมีคุณภาพและน่าเชื่อถือมากขึ้น

บทที่ 5 เป็นบทสรุปผลการวิจัยและข้อเสนอแนะ

บทที่ 2
ทฤษฎีพื้นฐานที่ใช้ในการวิจัย

ในหัวข้อนี้จะกล่าวถึงทฤษฎีพื้นฐานต่าง ๆ ที่เกี่ยวข้องในการวิจัย ซึ่งรวมถึงพื้นฐานของระบบโครงข่ายประสาทเทียม (Neural Network) ซึ่งระบบนี้เป็นรากฐานของการวิจัยในเรื่องเกี่ยวกับ การเปรียบเทียบรูปแบบต่าง ๆ การพยากรณ์ข้อมูลล่วงหน้า และการแบ่งแยกประเภทกลุ่มของข้อมูล ซึ่งในวิทยาพนธ์นี้จะนำรากฐานของระบบโครงข่ายประสาทเทียมมาใช้ในการแบ่งแยกประเภทกลุ่มของข้อมูลโดยเฉพาะ มีนักวิจัยหลายท่านได้ให้ความสนใจและได้เริ่มต้นงานวิจัยจากรากฐานของระบบโครงข่ายประสาทเทียม และได้มีการพัฒนาออกมาเป็นทฤษฎีต่าง ๆ มากขึ้น โดยที่หลักการซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine : SVM) และ หลักการลีซท์ สแควซัพพอร์ตเวกเตอร์แมชชีน (Least Square Support Vector Machine : LS-SVM) ก็ได้ถูกพัฒนามาจากระบบโครงข่ายประสาทเทียมด้วยเช่นกัน นอกจากนี้ยังจะกล่าวถึงทฤษฎีพื้นฐานของหลักการซัพพอร์ตเวกเตอร์แมชชีน และข้อแตกต่างของหลักการลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีน รวมถึงวิธีการทำ คลอส วาริเดชัน (Cross Validation) และกริดเสิร์จ (Grid Search) ซึ่งเนื้อหาทั้งหมดนี้จำเป็นสำหรับการศึกษา และประเมินประสิทธิภาพของการสร้างแบบจำลองในการแบ่งประเภทของลูกค้าในสถาบันการเงินแห่งประเทศไทย ดังนั้นจะทำการนำเสนอหลักการเหล่านี้อย่างละเอียดต่อไป

2.1 การพยากรณ์ (Forecasting)

นิยามศัพท์เฉพาะที่ใช้ในส่วนของการพยากรณ์นี้ มีดังนี้

การพยากรณ์ คือ ศิลปะ และ ศาสตร์ ของการทำนายอนาคต

การทำนาย คือ การที่คิดว่าอะไรจะเกิดขึ้นในอนาคต

การวางแผน คือ การคิดสร้างให้อนาคตเป็นในสิ่งที่ควรจะเป็น

ดังนั้นจะสรุปได้ว่า การพยากรณ์คือการทำนายสถานการณ์ในอนาคต เพื่อหาข้อสรุปสำหรับช่วยตัดสินใจและเตรียมรับสถานการณ์ที่คาดว่าจะเกิดขึ้นได้ [1]

2.1.1 ประเภทของการพยากรณ์

2.1.1.1 การพยากรณ์เชิงปริมาณ (Quantitative forecasting)

1. ตัวแบบคณิตศาสตร์
2. ตัวแบบอนุกรมเวลา
3. ตัวแบบการถดถอย

2.1.1.2 การพยากรณ์เชิงคุณภาพ (Qualitative forecasting)

1. วินิจฉัย (Judgment)
2. ลางสังหรณ์ (Intuition)

2.1.2 ขั้นตอนการพยากรณ์เชิงปริมาณ

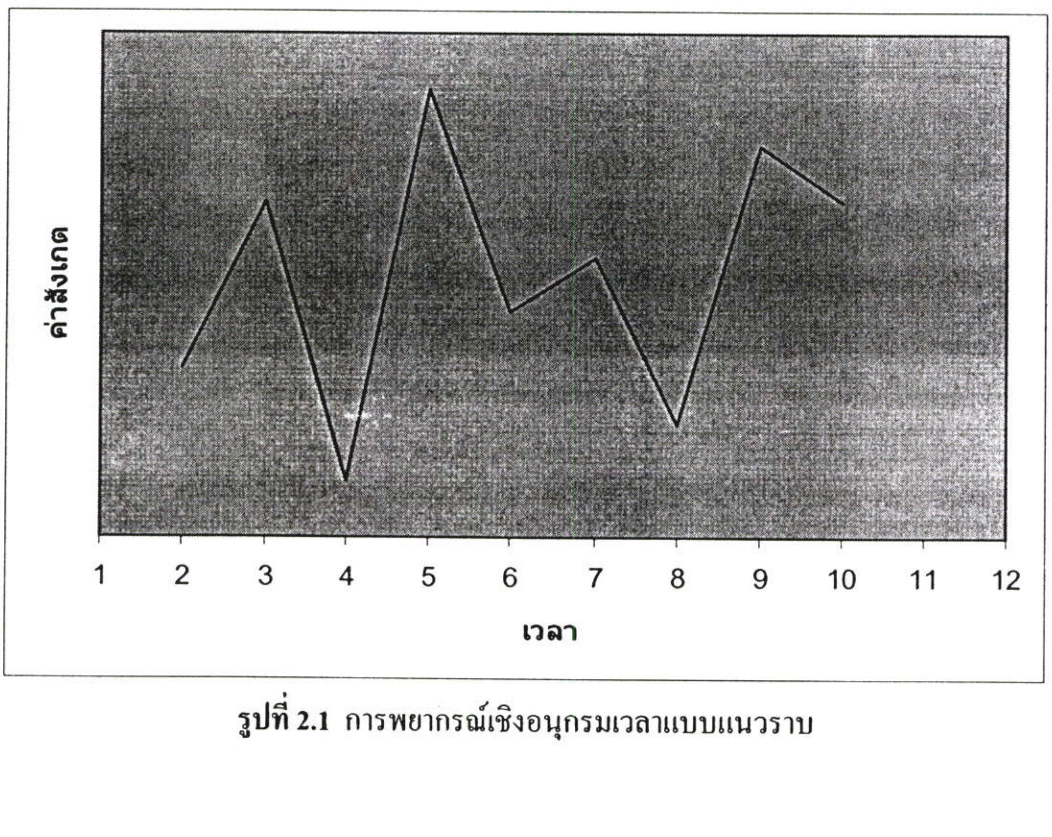
1. กำหนดวัตถุประสงค์
2. กำหนดความยาวนานของการพยากรณ์ (Forecasting horizon) อย่างเหมาะสม
3. เลือกใช้วิธีพยากรณ์ให้เหมาะสมกับรูปแบบของข้อมูล
4. รวบรวมและวิเคราะห์ข้อมูล
5. นำผลไปใช้

2.1.3 การพยากรณ์เชิงอนุกรมเวลา

อนุกรมเวลา (Time series) คือ เซตของข้อมูลเชิงปริมาณที่จัดเก็บในช่วงเวลาหนึ่ง เทคนิคนี้มีหลายวิธี ก่อนเลือกใช้วิธีใดนั้นต้องตรวจสอบข้อมูลว่ามีรูปแบบ (Pattern) ลักษณะใด โดยที่แต่ละลักษณะจะมีความแตกต่างกันดังรูปที่ 2.1 ถึงรูปที่ 2.4 ดังนี้

2.1.3.1 แบบแนวราบ (Horizontal pattern)

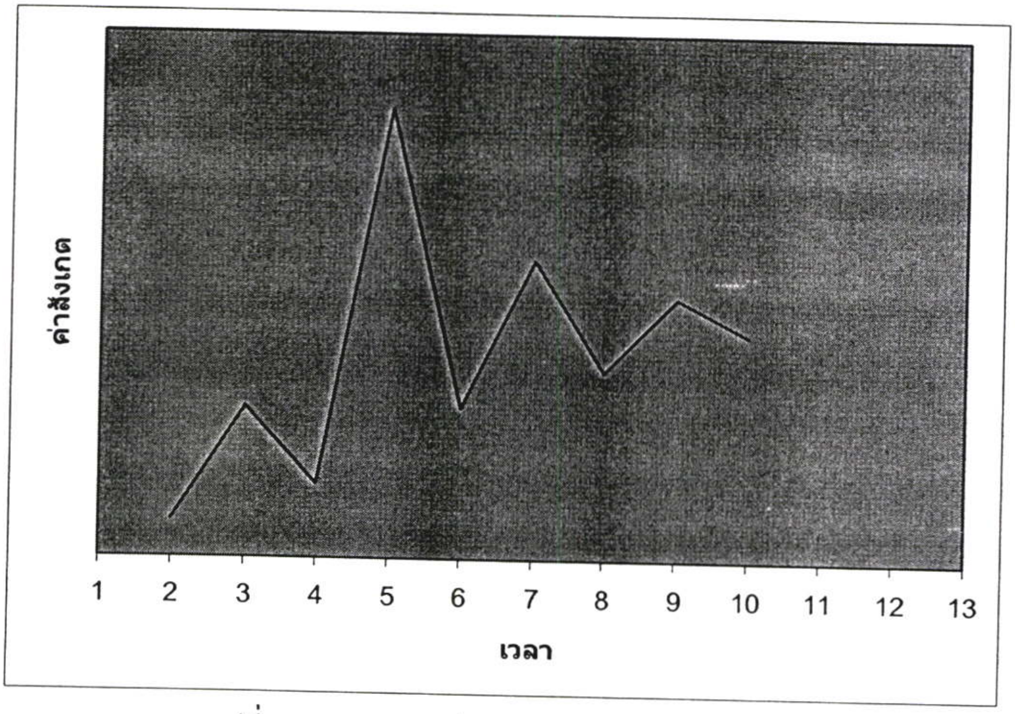
จากรูปที่ 2.1 ค่าในแกน Y (แนวตั้ง) เป็นค่าจากการสังเกต และค่าในแกน X (แนวนอน) เป็นค่าของเวลา ซึ่งกราฟที่ได้จะอยู่ในลักษณะของแนวนอน



รูปที่ 2.1 การพยากรณ์เชิงอนุกรมเวลาแบบแนวราบ

2.1.3.2 แบบแนวโน้ม (Trend pattern)

จากรูปที่ 2.2 ค่าในแกน Y (แนวตั้ง) เป็นค่าจากการสังเกต และค่าในแกน X (แนวนอน) เป็นค่าของเวลา ซึ่งกราฟที่ได้จะอยู่ในลักษณะของแนวที่โน้มเอียงสูงขึ้นจากซ้ายไปขวา



รูปที่ 2.2 การพยากรณ์เชิงอนุกรมเวลาแบบแนวโน้ม

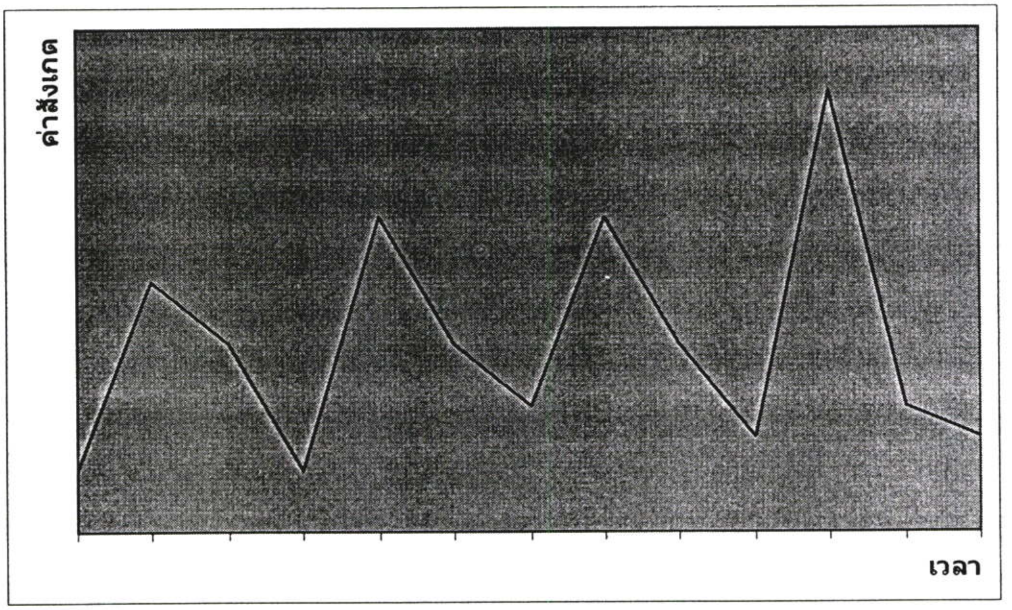
2.1.3.3 ข้อมูลที่เกิดขึ้นอย่างสุ่มหรือไม่ปกติ (Random or irregular pattern)

อนุกรมเวลาที่เกิดขึ้นอย่างไม่มีแบบแผนแน่นอน ตายตัว ทำให้ไม่อาจคาดการณ์ได้ว่าจะเกิดหรือไม่เกิด เช่น ภาวะฝนแล้ง น้ำท่วม ไฟไหม้ สงครามจราจล การนัดหยุดงาน

2.1.3.4 แบบฤดูกาล (Seasonal pattern)

จากรูปที่ 2.3 ค่าในแกน Y (แนวตั้ง) เป็นค่าจากการสังเกต และค่าในแกน X (แนวนอน) เป็นค่าของเวลา ซึ่งกราฟที่ได้จะอยู่ในลักษณะคล้ายๆกันในแต่ละช่วงของเวลา

ซึ่งในรูปแบบของฤดูกาลนั้น จะมีช่วงเวลาที่ค่อนข้างแน่นอนและตายตัว โดยที่จะมักจะสามารถคาดการณ์ได้ล่วงหน้าว่าจะเกิดเหตุการณ์นั้นซ้ำอีกเมื่อใด

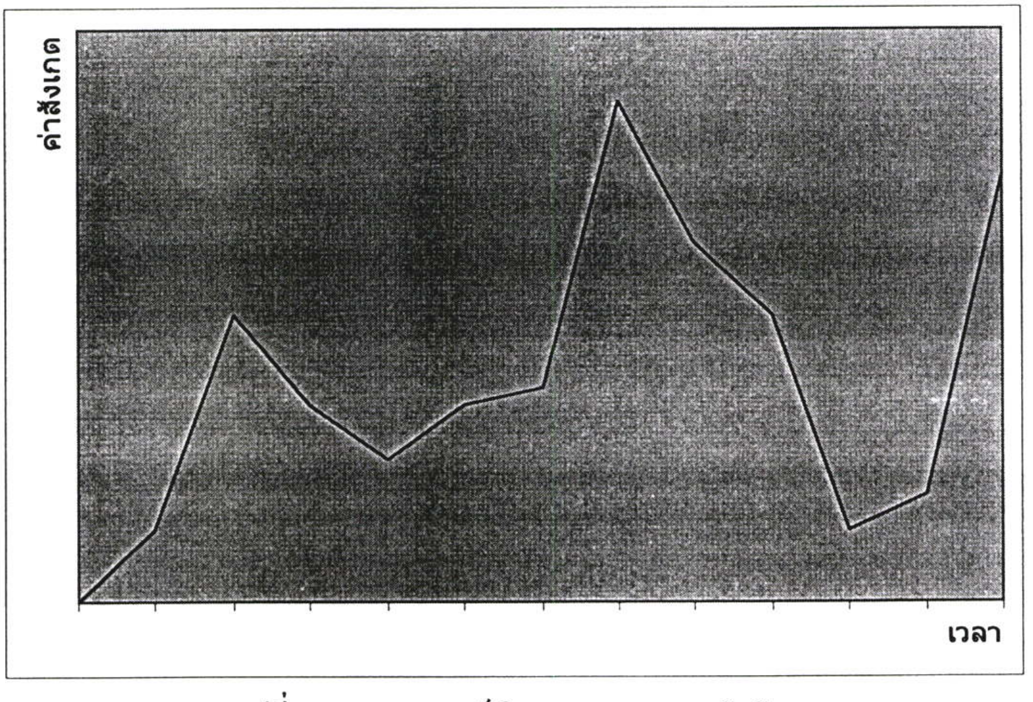


รูปที่ 2.3 การพยากรณ์เชิงอนุกรมเวลาแบบฤดูกาล

2.1.3.5 แบบวัฏจักร (Cyclical pattern)

จากรูปที่ 2.4 ค่าในแกน Y (แนวตั้ง) เป็นค่าจากการสังเกต และค่าในแกน X (แนวนอน) เป็นค่าของเวลา ซึ่งกราฟที่ได้จะอยู่ในลักษณะที่แตกต่างกันโดยสิ้นเชิง

ซึ่งลักษณะของวัฏจักรการเกิดของเหตุการณ์ จะมีช่วงความยาวนานที่ไม่คงที่ ซึ่งรวมถึงค่าสังเกตที่เกิดขึ้นก็มีขนาด (Magnitude) ไม่แน่นอน



รูปที่ 2.4 การพยากรณ์เชิงอนุกรมเวลาแบบวัฏจักร

2.2 ระบบโครงข่ายประสาทเทียม

ในรอบสิบปีที่ผ่านมา ได้มีผลงานวิจัยโดยที่นำหลักการของระบบโครงข่ายประสาทเทียมมาใช้ประโยชน์ในขอบเขตที่แตกต่างกันไป แต่จะมีอยู่หนึ่งหลักการที่เป็นหลักสำคัญและมีชื่อเสียงมากที่ถูกนำมาใช้ซึ่งเรียกว่า มัลติเลเยอร์ เพอร์เซพตรอน (Multilayer Perceptron : MLP) [2], [14] โดยมีหลักการดังนี้ คือ

2.2.1 มัลติเลเยอร์ เพอร์เซพตรอน

มัลติเลเยอร์ เพอร์เซพตรอน เป็นหนึ่งในแบบจำลองของระบบโครงข่ายประสาทเทียมที่ถูกนำมาประยุกต์ใช้กันอย่างกว้างขวาง ซึ่งมัลติเลเยอร์ เพอร์เซพตรอน จะเหมาะสมในการแบ่งประเภทข้อมูลในรูปแบบง่าย ๆ เท่านั้น เมื่อทำการแบ่งประเภทข้อมูลออกเป็นสองประเภทด้วยกันจะหมายความว่าข้อมูลทั้งสองประเภทนั้นจะถูกทำการแบ่งโดยระนาบในมิติ (Dimension) ที่สูงกว่าในเนื้อที่ของข้อมูลตัวอย่าง (Sample space) จากการวิจัยก่อนหน้านี้มีผู้ที่ทำการพิสูจน์ได้ว่าเพอร์เซพตรอน (Perceptron) เป็นการแบ่งข้อมูลที่เป็นแบบเชิงเส้นในขอบเขตของส่วนอินพุท (input space) โดยทำการสร้างระนาบหลาย ๆ อัน (Hyperplane) เท่านั้นเนื่องจาก เพอร์เซพตรอนไม่สามารถที่จะทำการแก้ปัญหาขอบเขตของ XOR ได้เพราะปัญหานี้จะสามารถแก้ได้โดยการสร้างขอบเขตการตัดสินใจของข้อมูลที่ไม่เป็นแบบเชิงเส้น แต่เนื่องจากข้อมูลของความเป็นจริงไม่ได้มีแต่ข้อมูลที่เป็นแบบเชิงเส้นเท่านั้น ดังนั้นนักวิจัยส่วนใหญ่จึงไม่สามารถนำ เพอร์เซพตรอนเดียวมาใช้ในการแก้ปัญหาการแบ่งประเภทของข้อมูลได้ จึงต้องนำ มัลติเลเยอร์ เพอร์เซพตรอนมาใช้ในการแก้ปัญหานี้

มัลติเลเยอร์ เพอร์เซพตรอน ได้ถูกนำมาประยุกต์ใช้อย่างมีประสิทธิภาพในการแก้ปัญหาที่ยาก ๆ และแตกต่างกันออกไปโดยทำการฝึกฝน (Training) ข้อมูลภายใต้การดูแลคุณสมบัติให้ถูกต้องเหมาะสมตามอัลกอริธึม ที่เรียกว่า วิธีการแพร่กระจายถอยหลัง (Back – Propagation : BP) โดยพื้นฐานแล้วการเรียนรู้จากวิธีการแพร่กระจายกลับคืนจะประกอบด้วย ตัวที่ผ่านเข้าไปในลำดับชั้น (layer) ที่แตกต่างกันของระบบโครงข่าย โดยที่จะมีตัวที่เดินหน้า (Forward) ผ่านเข้าไป และตัวที่ถอยหลังกลับ (Backward) ผ่านเข้าไป สำหรับตัวที่เดินหน้าผ่าน อินพุทเวกเตอร์ (Input vector) จะถูกนำมาใช้ที่เซนโซรีโหนด (Sensory node) ของระบบโครงข่ายและมันจะส่งผลกระทบต่อการแพร่กระจายเข้าไปในชั้นของระบบโครงข่ายชั้นต่อชั้น สุดท้ายแล้วกลุ่มของเอาท์พุทที่ถูกสร้างออกมานั้นจะเป็นค่าตอบรับจริงของระบบโครงข่าย ในระหว่างตัวเดินหน้าผ่าน ค่าน้ำหนัก (Weights) ของระบบโครงข่ายจะถูกทำให้เป็นค่าคงที่ ซึ่งในทางกลับกันของตัวที่ถอยหลังกลับผ่านค่าน้ำหนัก จะถูกเปลี่ยนไปโดยขึ้นอยู่กับกฎการตรวจสอบค่าความผิดพลาด (Error correction rule) โดยเฉพาะแล้วค่าตอบรับจริงของระบบโครงข่ายจะถูกนำมาหักลบออกจากค่าตอบรับที่ต้องการ ซึ่งจะทำให้เกิดค่าสัญญาณที่ผิดพลาด (Error signal) ได้ โดยที่สัญญาณความผิดพลาดนั้นเป็นผลที่เกิด

จากการแพร่กระจายแบบถอยหลังผ่านเข้าไปในระบบโครงข่ายโดยเรียกว่า การต่อต้านทิศทางของการเชื่อมต่อ

ในเรื่องของการวิเคราะห์ความน่าเชื่อถือของการจัดประเภทข้อมูลนั้น โครงสร้างของ มัลติเลเยอร์ เพอร์เซพตรอน นั้นจะประกอบด้วย ชั้นอินพุท (Input layer) ชั้นซ่อน (Hidden layer) และชั้นเอาท์พุท (Output layer) สำหรับชั้นอินพุทนั้นจะประกอบด้วยหลาย ๆ โหนด (Node) ซึ่งแสดงถึงดรรชนีทางการเงิน ซึ่งดรรชนีเหล่านี้จะถูกเลือกมาใช้ในการวิเคราะห์ส่วนประกอบที่สำคัญ การวิเคราะส่วนประกอบรอบนอก และอื่น ๆ สำหรับชั้นที่ถูกซ่อนไว้ ปกติจะใช้สำหรับลอจิสทิกฟังก์ชัน (Logistic function) หรือ ซิกมอยด์ฟังก์ชัน (Sigmoid function) และสุดท้ายชั้นเอาท์พุท จะมีเพียงแค่หนึ่ง หรือสองโหนด โดยที่มันจะสร้างผลลัพท์ของการวิเคราะห์ค่าความเสี่ยงของความเชื่อถือนั้น ๆ ออกมาได้

ฉะนั้น จากการสร้างแบบจำลองของ McCulloch-Pitts ประสาทเทียมได้ถูกนำมาจำลองเป็นสถิติขององค์ประกอบที่ไม่เป็นแบบเชิงเส้นอย่างง่าย ๆ โดยทำการนำผลรวมของสัญญาณที่เข้ามา x_i นำมาคูณกับค่าตัวถ่วงน้ำหนัก w_i ที่เชื่อมต่อระหว่างชั้นอินพุทและชั้นซ่อน หลังจากทำการบวกค่าไบแอซ (Bias) ในเทอมของ b (ซึ่งบางทีจะเรียกว่า เธรชโชล (Threshold) โดยผลลัพธ์แอคทิเวชัน (Activation) a จะแสดงตามสมการที่ (2.1) ดังนี้

$$a = \sum_i w_i x_i + b \tag{2.1}$$

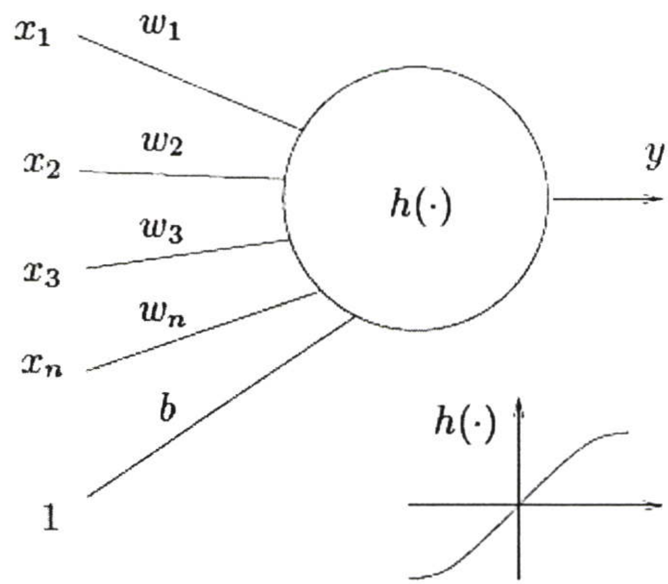
จากสมการ (2.1) จะถูกส่งต่อเข้าไปในสถิติของข้อมูลที่ไม่เป็นแบบเชิงเส้น $h(\cdot)$ ซึ่งเรียกว่า แอคทิเวชันฟังก์ชัน (activation function) ทำให้ได้ผล y ดังเช่นสมการที่ (2.2) ดังนี้

$$y = h\left(\sum_{i=1}^{n} w_i x_i + b\right) \tag{2.2}$$

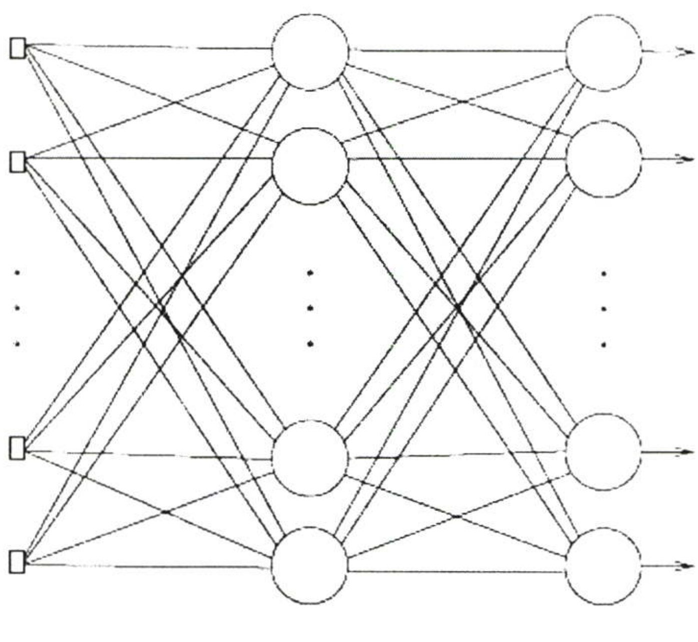
ความที่ไม่เป็นแบบเชิงเส้นนั้น เป็นเรื่องธรรมดาของชนิดของค่าความอิ่มตัว (Saturation) เช่น $\tan(\cdot)$ ตามหลักชีววิทยาแล้ว ความเหมือนกันกับระบบประสาทเทียมนี้จะขึ้นอยู่กับการรวบรวมข้อมูลของสัญญาณที่กำลังเข้ามาซึ่งจะมากกว่าค่าของเธรชโชลหรือไม่ โดยที่แบบจำลองของ McCulloch-Pitts นี้เป็นแบบจำลองในรูปแบบที่ง่ายต่อหลักชีววิทยาของประสาทเทียมซึ่งได้ถูกนำมาใช้บ่อย ๆ ภายในแบบจำลองของโครงข่ายประสาทเทียม หรือในอีกทางหนึ่งจะทราบว่ายังมีแบบจำลองที่ถูกนำมาใช้อย่างผิด ๆ สำหรับหลักทางชีวภาพของระบบประสาทและเป้าหมายของ

ระบบโครงข่ายประสาทเทียมจะไม่ได้ถูกนำไปใช้ให้เกิดประโยชน์แต่จะค่อนข้างเป็นไปในทางการสร้างแบบจำลองทางคณิตศาสตร์ที่มีประสิทธิภาพสูง

จากรูปที่ 2.5 นี้ได้แสดงถึงโครงสร้างระบบโครงข่ายประสาทเทียมของ มัลติเลเยอร์ เพอร์เซพตรอน ซึ่งประกอบด้วย สี่อินพุทโหนด สี่โหนดที่ซ่อนไว้ และ หนึ่งเอาท์พุทโหนด



(ก)



(ข)

รูปที่ 2.5 การแสดงโครงข่ายประสาทเทียมของ (ก) หนึ่งเพอร์เซพตรอน (ข) มัลติเลเยอร์ เพอร์เซพตรอน แบบหลายชั้น [14]

แบบจำลองโครงสร้างของระบบประสาทเทียมหนึ่งชั้นจะแสดงได้ดังรูปที่ 2.5 (ก) ส่วนรูปที่ 2.5 (ข) ได้แสดงถึงแบบจำลองที่มีประสิทธิภาพมากของโครงสร้างของระบบประสาทเทียมแบบหลายชั้น หรือมัลติเลเยอร์ เพอร์เซพตรอน ซึ่งได้ถูกเพิ่มหนึ่งหรือมากกว่าของชั้นที่ถูกปิดบังอยู่ ซึ่งจะเป็นสถิติของแบบจำลองของข้อมูลที่ไม่เป็นแบบเชิงเส้น $y = f(x)$ ซึ่งได้ถูกอธิบายตามสมการที่ (2.3) เมทริกซ์ เวกเตอร์ (Matrix vector) ดังต่อไปนี้

$$y = W\tanh(Vx + \beta) \tag{2.3}$$

เมื่อ $x \in \Re^{n}$ เป็นค่าอินพุท

$y \in \Re^{n_y}$ เป็นค่าเอาท์พุท

$\beta \in \Re^{n_h}$ เป็นค่าไบแอซ ซึ่งประกอบด้วย ค่าเธรชโชลของ n_h เรียกว่านิวรอนซ่อน (Hidden neurons)

$W \in \Re^{n_y \times n_h}$ เป็นค่าถ่วงน้ำหนักที่เชื่อมระหว่างชั้นอินพุทกับชั้นซ่อน

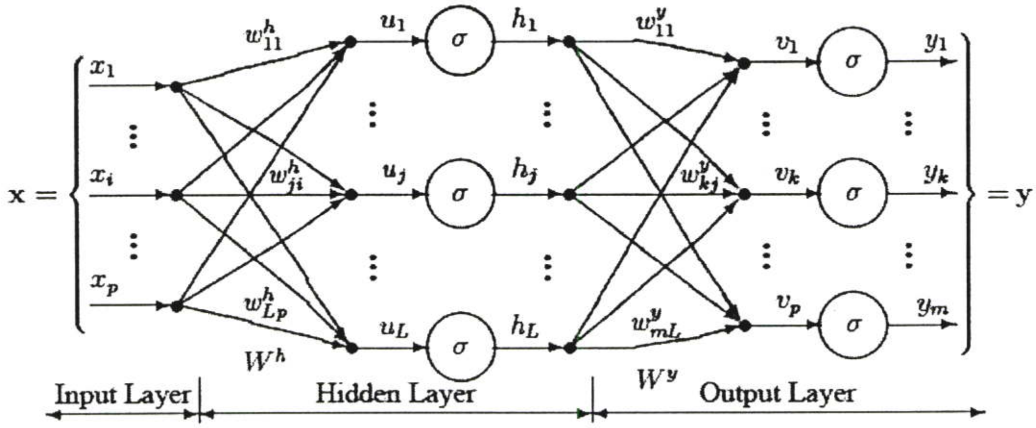
$V \in \Re^{n_h \times n}$ เป็นค่าถ่วงน้ำหนักที่เชื่อมระหว่างชั้นซ่อนและชั้นเอาท์พุท

จากคำอธิบายเหล่านี้ แอคทิเวชันฟังก์ชันแบบเชิงเส้นได้ถูกนำมาใช้สำหรับชั้นเอาท์พุทขึ้นอยู่กับการประยุกต์ (Application) ซึ่งแต่ละคนเลือกนำแต่ละฟังก์ชันมาใช้ สำหรับปัญหาของการประมาณค่าจากฟังก์ชันที่ไม่เป็นแบบเชิงเส้นนั้น บางคนใช้แอคทิเวชันฟังก์ชันแบบเชิงเส้นในชั้นเอาท์พุท หรือบางครั้งก็ได้เลือกโครงข่ายประสาทเทียบที่มีชั้นซ่อนอยู่สองชั้นนำมาใช้ได้เหมือนกัน

เหตุผลหนึ่ง ของการประสบความสำเร็จด้วยระบบโครงข่ายประสาทเทียมในอดีตนั้นคือความเป็นจริงที่ระบบนี้ สามารถประมาณค่าได้จริงอย่างกว้างขวางและเป็นสากล (Universal approximation) โดยได้ถูกทำการพิสูจน์ในทางคณิตศาสตร์แล้วว่า มัลติเลเยอร์ เพอร์เซพตรอน สามารถประมาณค่าต่าง ๆ ที่ไม่เป็นแบบเชิงเส้นอย่างต่อเนื่องได้อย่างดีและมีประสิทธิภาพ

2.2.1.1 แบบจำลองของโครงข่ายประสาทเทียมหลายชั้นในแบบการมุ่งไปข้างหน้า (The Multilayer Feed-Forward Neural Network Model)

ระบบโครงข่ายประสาทเทียมของแบบจำลองที่ได้แสดงตัวอย่างดังรูปที่ 2.6 นั้นจะมีชั้นของอินพุท ที่อยู่ด้านซ้ายมือ ต่อจากนั้นจะมีชั้นซ่อน อยู่ตรงกลางและสุดท้ายด้านขวามือจะเป็นชั้นของเอาท์พุทซึ่งแต่ละชั้นจะประกอบไปด้วยประสาทเทียมสามจุดด้วยกัน โดยที่ค่า $x_1,...,x_p$ คือค่าอินพุท ซึ่งค่า i และ p เป็นจำนวนของค่าอินพุท x ซึ่งรูปที่ 2.6 นี้ได้ถูกอธิบายขั้นตอนการทำงานไว้ในแต่ละชั้นดังนี้



รูปที่ 2.6 แบบจำลองของโครงข่ายประสาทเทียมสามชั้นในแบบการมุ่งไปข้างหน้า [22]

2.2.1.1.1 ชั้นอินพุท

เวกเตอร์ของค่าตัวแปรที่ไว้ใช้ทำนาย $(x_1,\dots,x_p)$ จะถูกนำมาแสดงอยู่ในชั้นของอินพุท โดยที่ชั้นอินพุทจะมีการประมวลผลก่อน โดยที่ชั้นอินพุทจะทำการตั้งมาตรฐานค่าเหล่านี้ด้วยการลบออกจากค่ามัธยฐานและหารด้วยค่าขอบเขตระหว่างค่าตั้งฉาก จากนั้นจะทำการกระจายค่าเหล่านี้ไปยังแต่ละจุดประสาทเทียมในชั้นซ่อน นอกจากนั้นตัวแปรที่ไว้ใช้ทำนาย จะมีค่าอินพุทคงที่ของ 1.0 ซึ่งเรียกว่า ไบแอซโดยที่จะถูกใส่เข้าไปในแต่ละชั้นซ่อนด้วย ค่าไบแอซจะถูกนำมาจับคู่กับค่าน้ำหนักและถูกเพิ่มไว้ในผลรวมของประสาทเทียมที่ถูกส่งเข้าไป

2.2.1.1.2 ชั้นซ่อน

ตัวประสาทเทียมที่เข้ามาถึงในชั้นนี้ ค่าจากแต่ละประสาทเทียมในชั้นซ่อนเป็นค่าที่ถูกคูณด้วยค่าน้ำหนัก (w_{ji}) และค่าน้ำหนักที่ได้จะถูกเพิ่มเข้าไปด้วยกันกับค่าไบแอซ u_j โดยค่า j เป็นค่าของลำดับชั้นซ่อน โดยที่ค่าผลรวมน้ำหนัก u_j จะถูกส่งเข้าไปในฟังก์ชัน σ ซึ่งผลลัพธ์ที่เป็นค่า h_j และจะถูกส่งจากชั้นซ่อนไปยังชั้นเอาท์พุท

2.2.1.1.3 ชั้นเอาท์พุท

ตัวประสาทเทียมที่เข้ามาถึงในชั้นเอาท์พุทนี้ แต่ละค่าจากชั้นซ่อนได้ถูกนำมาคูณด้วยค่าน้ำหนัก (w_{ji}) และค่าน้ำหนักที่ได้จะถูกเพิ่มเข้าไปด้วยกันกับค่า v_j ค่าผลรวมน้ำหนัก v_j จะถูกส่งเข้าไปในฟังก์ชัน σ ซึ่งผลลัพธ์ที่เป็นค่า y_k และค่าเหล่านี้จะถูกส่งออกมานอกโครงข่าย

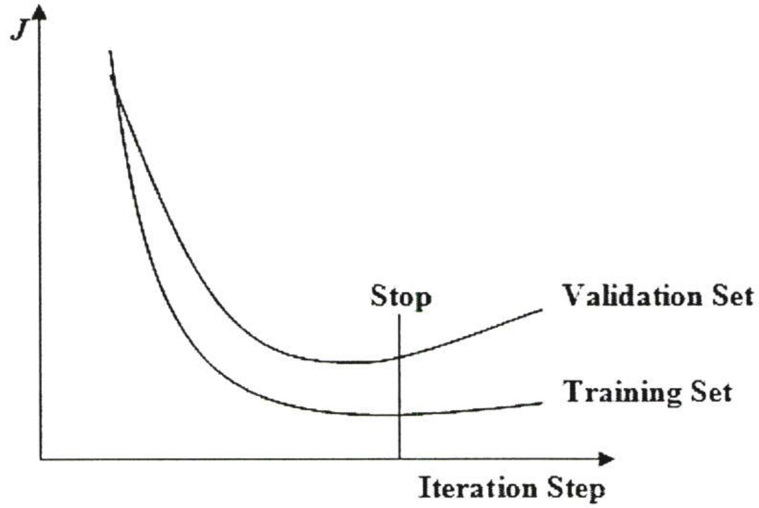
2.2.2 การแบ่งแยกประเภท (Classification)

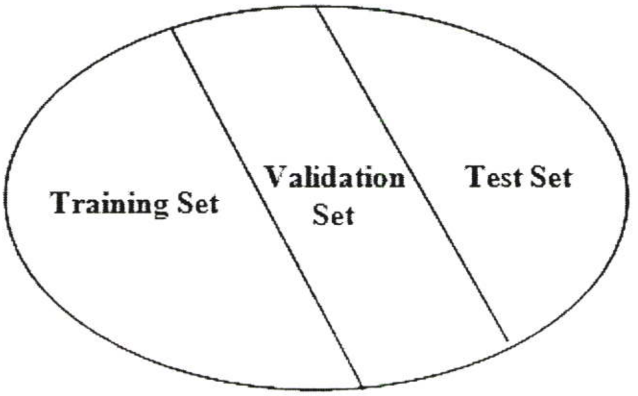
ขั้นตอนแรกสำหรับการฝึกฝนของมัลติเลเยอร์ เพอร์เซพตรอน และโครงข่ายที่มุ่งไปข้างหน้านั้น โดยทั่วไปจะเป็นวิธีของการแพร่กระจายแบบถอยหลัง ในกรณีนี้จะเข้าไปเกี่ยวข้องกับการลดค่าน้ำหนักที่เชื่อมต่อกันระหว่างฟังก์ชันต่าง ๆ

$$\min_{\theta \in \Re^p} J_{train}(\theta) = \frac{1}{N} \sum_{k=1}^{N} \left\| y_k - f(x_k;\theta) \right\|_2^2 \qquad (2.4)$$

ในขณะที่ $\theta = [w(:), V(:), \beta] \in \Re^p$ เป็นเวกเตอร์ของค่า p ที่ประกอบด้วยค่าน้ำหนักและไบแอซ ในชุดของการฝึกฝนข้อมูลอินพุทและเอาท์พุทนั้นคือ $\{x_k, y_k\}_{k=1}^{N}$ โดยที่ค่า N เป็นจำนวนของข้อมูลในชุดฝึกฝน วิธีพื้นฐานของการแพร่กระจายถอยหลัง (BP) เป็นสิ่งที่จำเป็นที่สุดเพื่อที่จะทำให้ได้ฟังก์ชันที่ต้องการ โดยการปรับปรุงด้วยรูปแบบของกำลังการเคลื่อนไหวของวัตถุ (Momentum term) และอัตราการเรียนรู้เปลี่ยนแปลงนั้นได้มีการพัฒนาด้วยเช่นเดียวกัน

ปัญหาที่โดยทั่วไปทราบกันดี คือ การฝึกฝนข้อมูลที่มีความลงตัวมากจนเกินไป (OverFit) ซึ่งเป็นปัญหาที่ควรหลีกเลี่ยงไม่ให้เกิดขึ้น ซึ่งปัญหานี้จะเกิดขึ้นโดยเฉพาะเมื่อกำลังดำเนินการทำฟังก์ชันจากสมการที่ (2.4) ให้สมบูรณ์ในกรณีที่ทำการฝึกฝนข้อมูลจนกระทั่งข้อมูลมีค่าต่ำสุดของค่าที่นำมาฝึกฝนเหล่านี้ เนื่องจากเหตุนี้ ทำให้ปกติแล้วมักจะทำการแบ่งจำนวนของข้อมูลทั้งหมดออกเป็นชุดของการฝึกฝนข้อมูล ชุดของการหาค่าที่ถูกต้อง (Validation set) และชุดที่นำมาใช้ในการทดสอบ (Test set) ในส่วนของชุดของการหาค่าที่ถูกต้องนั้นจะถูกนำมาใช้ในการที่จะตัดสินใจเกี่ยวกับเมื่อไหร่ที่จะต้องหยุดการฝึกฝนข้อมูล ส่วนข้อมูลในชุดของการทดสอบนั้นจะถูกเก็บไว้โดยไม่มีส่วนร่วมอยู่ในชุดของการฝึกฝนข้อมูล ชุดของการหาค่าที่ถูกต้องแต่อย่างใดทั้งสิ้นดังรูปที่ 2.7





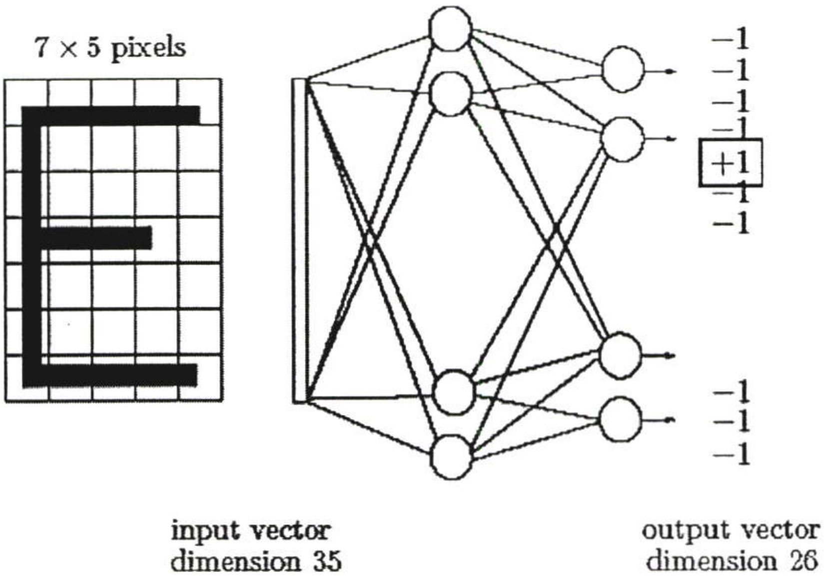
รูปที่ 2.7 การแบ่งแยกประเภทข้อมูล

เมื่อนำวิธีมัลติเลเยอร์ เพอร์เซพตรอนมาใช้ ซึ่งมักจะใช้ร่วมกับสมการที่ (2.3) ซึ่งประกอบไปด้วยชั้นซ่อนหนึ่งชั้นซึ่งอยู่ในฟังก์ชัน tanh และชั้นของเอาท์พุทที่ประกอบด้วยประสาทเทียมที่เป็นแบบเชิงเส้น ในกรณีของการแบ่งประเภทของข้อมูลด้วยวิธีมัลติเลเยอร์ เพอร์เซพตรอน นั้นสามารถที่จะนำฟังก์ชัน tanh มาใช้ได้เหมือนกันในชั้นของเอาท์พุทที่เป็นเหมือนชั้นซ่อนอยู่ แต่กระนั้นข้อมูลที่เป็นแบบเชิงเส้นก็ยังถูกนำมาใช้กับฟังก์ชันในสมการที่ (2.3) ด้วยเช่นกัน โดยปกติแล้วระบบโครงข่ายจะทำการฝึกฝนข้อมูลเหล่านั้นในสมการที่ (2.3) และทำการแบ่งประเภทข้อมูลด้วยการหาค่าจาก $y(x) = sign[f(x)]$ หรือจากสมการที่ (2.5) นี้

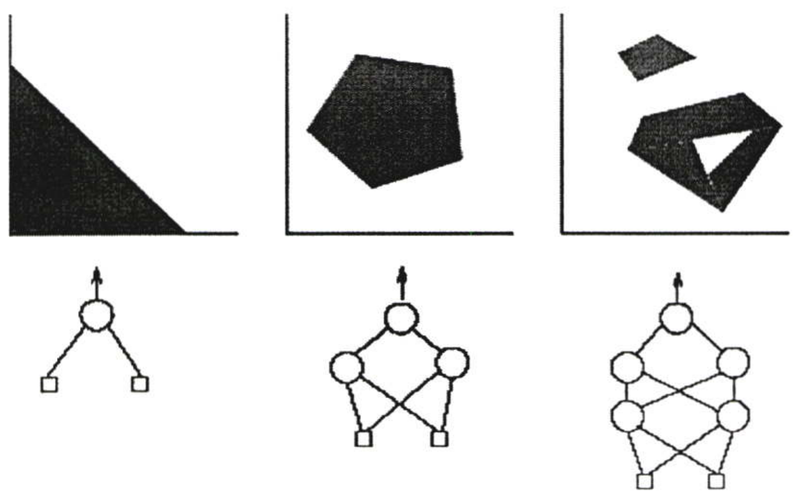
$$y(x) = sign[W\tanh(Vx+\beta)] \tag{2.5}$$

โครงข่ายที่ถูกนำมาฝึกฝนจนได้ค่าเอาท์พุทที่ต้องการคือ $y_k \in \{-1,+1\}$ สำหรับกรณีที่มีข้อมูลสองประเภท สำหรับการที่มีข้อมูลหลาย ๆ ประเภทนั้น สามารถที่จะนำค่าเอาท์พุทที่เพิ่ม

ขึ้นมาแสดงเป็นประเภทของข้อมูลต่าง ๆ ได้ จากรูปที่ 2.8 นั้นได้แสดงตัวอย่างการแยกแยะตัวหนังสือในจดหมายด้วยหลักการของมัลติเลเยอร์ เพอร์เซพตรอน โดยที่รูป 2.8 (ก) ได้แสดงถึงการเข้าถึงมัลติเลเยอร์ เพอร์เซพตรอน ด้วยวิธีความถดถอย โดยแบ่งประเภทข้อมูลออกเป็นหลายประเภท และรูป 2.8 (ข) เป็นการตัดสินใจหาค่าขอบเขตแบบซับซ้อน โดยสามารถที่จะทำได้ด้วยการเพิ่มชั้นซ่อนเข้าไปอีก



(ก)



(ข)

รูปที่ 2.8 (ก) การแยกแยะตัวอักษรด้วยหลักการมัลติเลเยอร์ เพอร์เซพตรอน (ข) การเพิ่มจำนวนชั้นซ่อน [14]

2.3 ซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machines)

ในส่วนนี้จะกล่าวถึงรายละเอียดทฤษฎีมาตรฐานของหลักการซัพพอร์ตเวกเตอร์แมชชีน ซึ่งจะกล่าวถึงในรูปแบบที่เป็นเชิงเส้นและไม่เป็นเชิงเส้น ของการแบ่งประเภทข้อมูลด้วยหลักการซัพพอร์ตเวกเตอร์แมชชีนอีกด้วย

หลักการซัพพอร์ตเวกเตอร์แมชชีน นั้นเป็นหลักการที่ค่อนข้างใหม่ในการจัดประเภทข้อมูล โดยใช้รากฐานมาจากระบบโครงข่ายประสาทเทียม ซึ่งงานในการจัดประเภทข้อมูลนั้นโดยปกติแล้วจะประกอบไปด้วยส่วนของการฝึกฝนข้อมูล (Training Data) และส่วนของการทดสอบข้อมูล (Testing Data) ซึ่งจะประกอบไปด้วยข้อมูลตัวอย่างจำนวนหนึ่งมาใช้ในการทำการทดลอง แต่ละข้อมูลตัวอย่างในชุดของการฝึกฝน จะเก็บหนึ่งค่าที่เป็นค่าเป้าหมาย (Target) หรือในที่นี้จะเรียกว่า คลาส ลาเบล (Class labels) และเก็บหลาย ๆ ค่าที่เป็นค่าคุณสมบัติ (Attributes) หรือในที่นี้จะใช้คำว่า ฟีทเจอร์ (Features) โดยที่เป้าหมายของหลักการซัพพอร์ตเวกเตอร์แมชชีนนั้น คือการสร้างแบบจำลองอันหนึ่ง ที่จะสามารถทำนายค่าเป้าหมายของข้อมูลตัวอย่างในชุดของการทดสอบซึ่งจะให้เฉพาะค่าคุณสมบัติเท่านั้น [3-7]

กำหนดให้ข้อมูลตัวอย่างในชุดของการฝึกฝนเป็น $(x_i, y_i), i = 1,\ldots,l$ โดยที่ $x_i \in R^n$ และ $y_i \in \{1,-1\}^l$ ซึ่งหลักการซัพพอร์ตเวกเตอร์แมชชีนจะต้องการผลจากการแก้สมการที่ (2.6) ดังต่อไปนี้

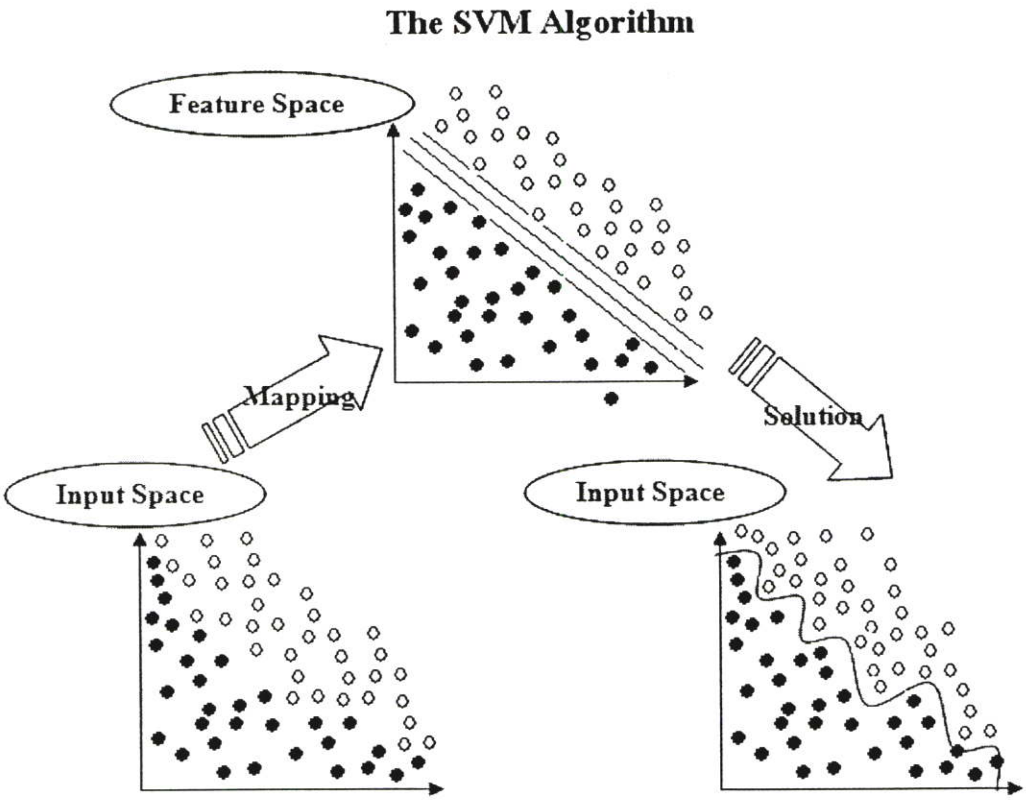
$$\min_{w,b,\xi} \quad \frac{1}{2} w^T w + C \sum_{i=1}^{l} \xi_i \tag{2.6}$$

โดยที่ $y_i(w^T \phi(x_i) + b) \geq 1 - \xi_i, \xi_i \geq 0$

โดยที่ $\phi(x_i)$ คือ การจับคู่ระหว่างฟังก์ชันเคอร์เนล กับ ค่าอินพุท x_i

และ ξ_i คือ ค่า สแลค (Slack Value)

จากสมการที่ (2.6) นี้ เทรนนิ่ง เวกเตอร์ x_i จะถูกจับคู่กับตัวแปรในชั้นที่สูงกว่าด้วย ϕ ดังนั้น ซัพพอร์ตเวกเตอร์แมชชีน จะทำการค้นหาแนวเขตหรือระนาบ (Hyperplane) ที่ทำการแบ่งแยกข้อมูลออกจากกัน โดยจะมีการกำหนดเส้นขอบสูงสุดและต่ำสุดของข้อมูลอีกด้วย โดยจะมีค่า C > 0 เป็นตัวแปรที่มาทำการถ่วงในส่วนของค่าความผิดพลาด (Error) และรูปที่ 2.9 นี้ได้แสดงถึงหลักการพื้นฐานของ ซัพพอร์ตเวกเตอร์แมชชีน

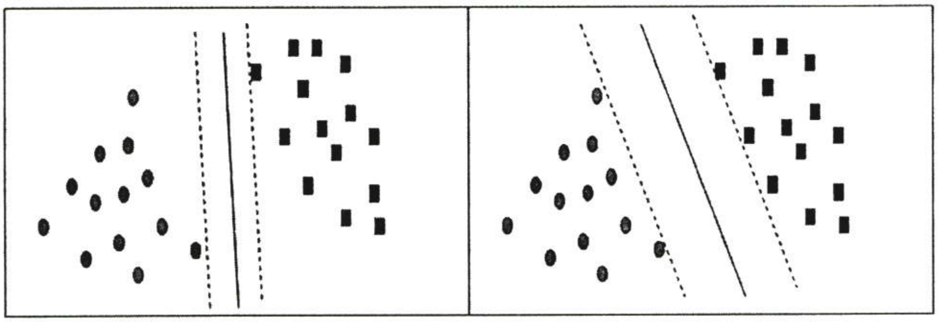


รูปที่ 2.9 ซัพพอร์ตเวกเตอร์แมชชีน อัลกอริธึม

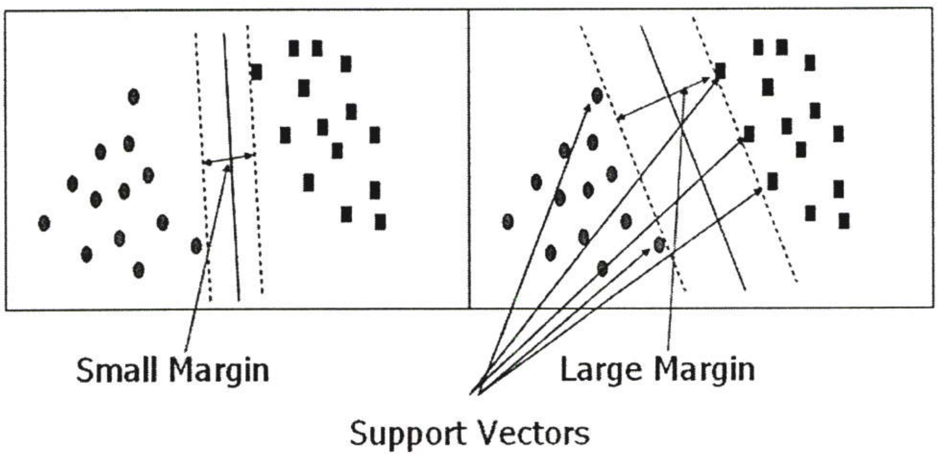
2.3.1 ตัวอย่างการใช้ ซัพพอร์ตเวกเตอร์แมชชีน แบบสองมิติ

ก่อนที่จะพิจารณาถึงระนาบหลาย ๆ อันนั้น ควรจะทำความเข้าใจกับตัวอย่างของสองมิติก่อน สมมติว่าข้อมูลที่มีตัวแปรเป้าหมายอยู่สองตัว และต้องการที่จะแบ่งข้อมูลนี้ออกเป็นสองประเภทด้วยกัน โดยจะสมมติด้วยว่าตัวแปรที่ไว้ทำนายสองตัวนั้นเป็นค่าแบบต่อเนื่อง (Continuous values) ถ้าทำการวาดกราฟ (Graph) จุดข้อมูลโดยใช้ของของตัวทำนายตัวหนึ่งบนแกน X และตัวทำนายอีกตัวหนึ่งบนแกน Y ก็อาจจะสรุปได้ด้วยรูปภาพ ยกตัวอย่างดังรูปที่ 2.10 โดยที่ประเภทของตัวแปรเป้าหมายตัวหนึ่งได้ถูกแสดงด้วยรูปสี่เหลี่ยม ในขณะที่อีกประเภทหนึ่งแสดงดังรูปไข่

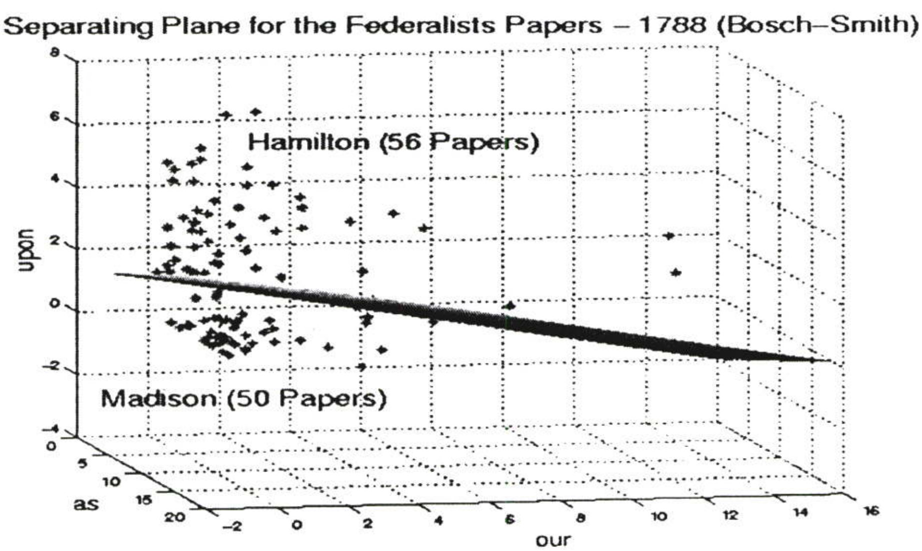
กรณีที่ประเภทหนึ่งตกไปในส่วนมุมด้านซ้ายล่าง และอีกกรณีที่อีกประเภทหนึ่งไปตกอยู่ในส่วนของมุมบนขวา สองกรณีนี้ได้แยกออกจากกันอย่างสิ้นเชิง การวิเคราะห์ด้วย หลักการซัพพอร์ตเวกเตอร์แมชชีน จะพยายามที่จะค้นหา หนึ่งมิติของระนาบ ยกตัวอย่างเช่นเส้นแบ่งกลางระหว่างเส้นประดังรูปที่ 2.10 (ก) ที่ได้แบ่งข้อมูลออกเป็นสองประเภทโดยขึ้นอยู่กับค่าเป้าหมายของแต่ละประเภท ส่วนเส้นประที่ขนานกันสองเส้นไปกับเส้นแบ่งข้อมูล ดังรูปที่ 2.10 (ข) นั้น จะมีระยะห่างซึ่งถูกเรียกว่า ค่าขอบเขต (Margin) เวกเตอร์ (จุดของข้อมูล) ที่อยู่ใกล้กับค่าขอบเขตจะเรียกว่าซัพพอทเวกเตอร์ (Support Vector) โดยที่รูป 2.10 (ค) แสดงถึงการแบ่งข้อมูลสองประเภทในมุมมองสามมิติ



(ก)



(ข)

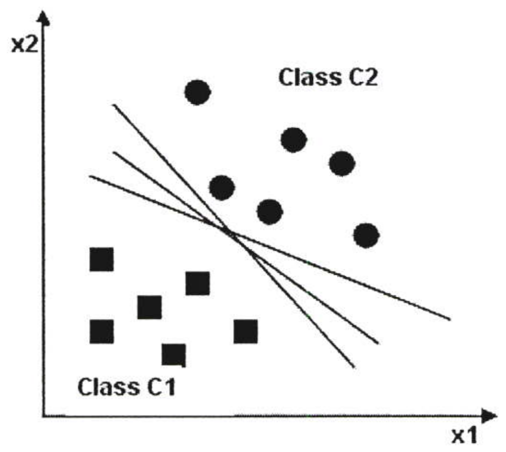


(ค)

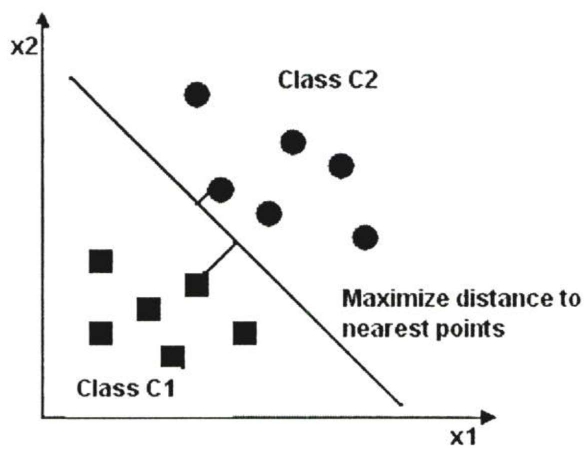
รูปที่ 2.10 ตัวอย่างการแยกประเภทข้อมูลออกเป็นสองประเภท ด้วยหลักการซัพพอร์ตเวกเตอร์แมชชีน [22]

2.3.2 ค่าของขอบเขต (Margin)

นอกจากที่ค่าถ่วงน้ำหนัก จะเป็นส่วนที่สำคัญในการสร้างแบบจำลอง ของระบบโครงข่ายประสาทเทียมสำหรับการแบ่งประเภทของข้อมูลแล้ว ค่าของขอบเขต (Margin) ของข้อมูลก็ยังเป็นส่วนที่สำคัญในลำดับต้น ๆ ของการสร้างความเข้าใจและความมีประสิทธิภาพในทฤษฎีของระบบ ซัพพอร์ตเวกเตอร์แมชชีน อีกด้วย จากรูป 2.11 ได้แสดงถึงตัวอย่างการแยกประเภทของข้อมูลในพื้นที่ของอินพุทแบบสองมิติ (dimension) จะสามารถเห็นได้ว่ามีระนาบหลาย ๆ ระนาบ เป็นตัวแบ่งแยกข้อมูลของเป็นสองประเภทด้วยกัน (โดยที่ข้อมูลแสดงด้วยค่า ● และ ■) อีกด้านหนึ่ง ก็จะสามารถกำหนดให้มีเพียงหนึ่งระนาบเท่านั้นที่จะมาเป็นตัวแบ่งแยกประเภทของข้อมูล



(ก)



(ข)

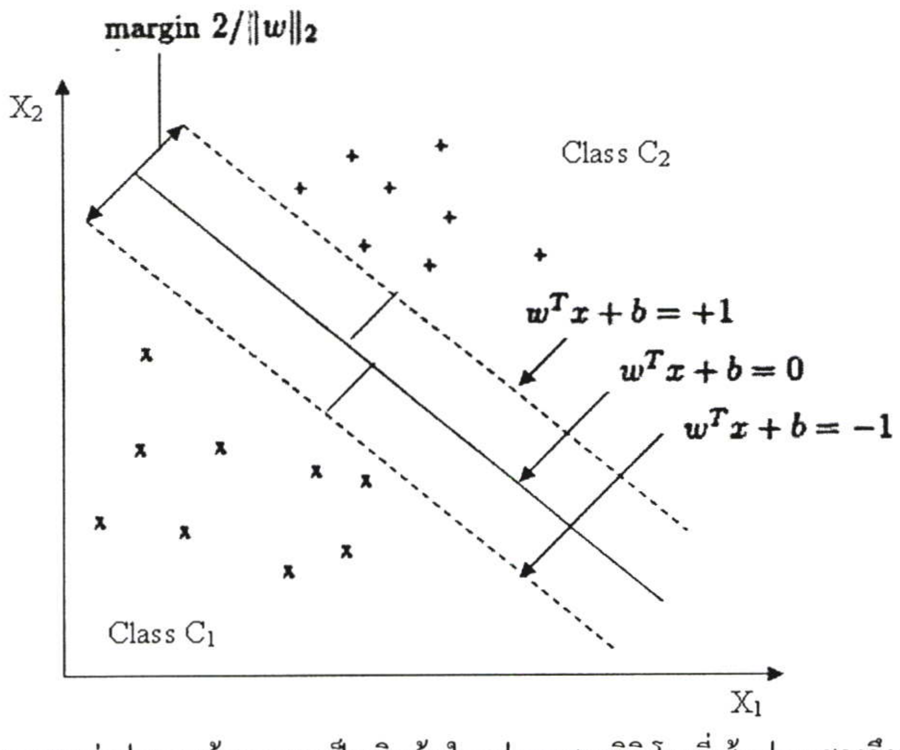
รูปที่ 2.11 การแบ่งประเภทข้อมูลแบบเป็นเชิงเส้น (ก) การแบ่งด้วยหลาย ๆ ระนาบ (ข) การแบ่งด้วยหนึ่งระนาบ

การที่จะประสบความสำเร็จในการแบ่งแยกประเภทข้อมูลนี้ได้ นักวิจัยชื่อ Vapnik [14] ได้มีการพิจารณาถึงการกำหนดสัดส่วนขอบเขตของข้อมูล (Scaling) ขึ้นมาใหม่ ยกตัวอย่างเช่น กำหนด จุดที่ใกล้ที่สุดที่ระนาบนั้น ๆ สามารถยอมรับได้ $\left|w^T x_k + b\right| = 1$ โดยที่จะได้รูปแบบตรงตามคุณสมบัติที่ต้องการสำหรับ (w,b) ของ ระนาบนั้น $y_k\left(w^T x_k + b\right) \geq 1$

ในกรณีนี้ค่าของขอบเขตจะเท่ากับ $2/\|w\|_2$ ดังนั้นจุดที่ใกล้ระนาบที่สุด จะมีระยะทางความห่างอยู่ที่ $1/\|w\|_2$ และในทางนี้เองจะสามารถ เพิ่มขนาดของระยะทางมาให้ใกล้จุดของข้อมูลทั้งสองประเภทได้ ซึ่งนี่ก็คือคุณสมบัติที่น่าพึงพอใจในการเพิ่มขนาดของค่าขอบเขตซึ่งเกิดจากการลดขนาดของ $\|w\|_2$ จากการลดขนาดของ $w^T w$ นี้จะทำให้มีความสัมพันธ์กันมากขึ้นที่จะใช้ค่าในเทอมของความถ่วงน้ำหนักลดลงในการฝึกฝนข้อมูลในระบบโครงข่ายประสาทเทียม

2.3.3 การแบ่งประเภทด้วยหลักการซัพพอร์ตเวกเตอร์แมชชีนแบบเป็นเชิงเส้น

ถึงแม้ว่าหลักการซัพพอร์ตเวกเตอร์แมชชีนแบบเป็นเชิงเส้นทั่วไป จะเพิ่งเกิดขึ้นไม่นานมานี้ แต่รากฐานของหลักการซัพพอร์ตเวกเตอร์แมชชีนได้เข้าไปถึงวิธี ในการสร้างระนาบที่ดีที่สุดที่มาใช้ในการแบ่งแยกประเภทของข้อมูลและประยุกต์ใช้ให้เข้ากับการใช้งาน (Application) ต่าง ๆ ที่มีอยู่จริงในปัจจุบัน และนี่คือสูตรต้นฉบับของหลักการซัพพอร์ตเวกเตอร์แมชชีนแบบเชิงเส้นที่เคยถูกนำมาใช้ในการแบ่งแยกประเภทของข้อมูลในอดีต [14] ซึ่งได้ถูกแสดงดังรูปที่ 2.12



รูปที่ 2.12 การแบ่งประเภทข้อมูลแบบเป็นเชิงเส้นในรูปแบบสองมิติ โดยที่เส้นประแสดงถึงค่าขอบเขต

พิจารณาจากกลุ่มข้อมูลในส่วนของการฝึกฝน $\{x_k, y_k\}_{k=1}^{N}$ ด้วยข้อมูลอินพุท $x_k \in \Re^n$ และข้อมูลเอาท์พุท $y_k \in \Re$ กับค่าของคลาส ลาเบล $y_k \in \{-1,+1\}$ และตัวแบ่งประเภทข้อมูลแบบเชิงเส้น (Linear classifier)

$$y(x) = sign\left[w^T x + b\right] \tag{2.7}$$

เมื่อข้อมูลทั้งสองประเภทได้ถูกทำการแบ่งแล้ว จะสามารถกล่าวได้ว่า

$$\begin{aligned} w^T x_k + b &\geq +1, \text{ ถ้า } y_k = +1 \\ w^T x_k + b &\leq -1, \text{ ถ้า } y_k = -1 \end{aligned} \tag{2.8}$$

สองเซตของสมการที่ (2.8) นี้สามารถถูกนำมารวมเป็นหนึ่งเซตได้ดังสมการที่ (2.9)

$$y_k \left[w^T x_k + b\right] \geq 1 \qquad k = 1,\ldots,N \tag{2.9}$$

สมการของซัพพอร์ตเวกเตอร์แมชชีนได้ถูกกำหนดขึ้น โดยนำมาจากส่วนที่ดีที่สุดของทฤษฎีนี้ ซึ่งทั่ว ๆ ไปจะถูกเริ่มต้นด้วยการกำหนดค่าปัญหาในขอบเขตของค่าความถ่วงน้ำหนักก่อนเป็นอย่างแรก โดยนำค่าความถ่วงน้ำหนักมาเป็นข้อจำกัด เพื่อที่จะให้ได้ค่าที่ดีที่สุด จากนั้นได้มีการกำหนดสมการของลากรองจ์ (Lagrangian) ขึ้น และสุดท้ายในการแก้ปัญหาของตัวคูณของสมการของลากรองจ์ (Lagrange Multiplier) ในส่วนหลังนี้จะถูกเรียกว่าค่าซัพพอร์ต (Support values)

สมการที่ดีที่สุดในการแก้ปัญหาได้ถูกแสดงไว้ โดยจะเพิ่มค่าของขอบเขตให้เข้าไปสู่ค่าจริงได้ ซึ่งจากการฝึกฝนข้อมูลนั้น ทำเพื่อต้องการที่จะให้ข้อมูลได้ถูกแบ่งประเภทได้อย่างถูกต้องที่สุด และปัญหาแรกสำหรับค่าถ่วงน้ำหนัก w คือ

$$\begin{bmatrix} \min\limits_{w,b} J_p(w) = \frac{1}{2} w^T w & \\ \text{such that} \quad y_k \left[w^T x_k + b\right] \geq 1, \; k = 1,\ldots,N \end{bmatrix} \tag{2.10}$$

สมการของลากรองจ์ สำหรับปัญหานี้คือ

$$L\left(w,b;\alpha \right) = \frac{1}{2} w^{T} w - \sum_{k=1}^{N} \alpha_{k} \left(y_{k} \left[w^{T} x_{k} + b \right] - 1 \right) \tag{2.11}$$

ด้วยตัวคูณของสมการของลากรองจ์ $\alpha_k \geq 0$ สำหรับ $k = 1,\ldots,N.$ และวิธีการแก้ปัญหาจะมีลักษณะเฉพาะซึ่งถูกรับผิดชอบโดย สมการของลากรองจ์ดังนี้

$$\max_{\alpha} \min_{w,b} L\left(w,b;\alpha \right). \tag{2.12}$$

ได้มาจาก

$$\begin{cases} \dfrac{\partial L}{\partial w} = 0 \rightarrow w = \displaystyle\sum_{k=1}^{N} \alpha_{k} y_{k} x_{k} \\ \dfrac{\partial L}{\partial b} = 0 \rightarrow \displaystyle\sum_{k=1}^{N} \alpha_{k} y_{k} = 0 \end{cases} \tag{2.13}$$

สำหรับสมการที่ใช้แบ่งประเภทข้อมูล คือ

$$y(x) = sign\left[\sum_{k=1}^{N} \alpha_{k} y_{k} x_{k}^{T} x + b \right]. \tag{2.14}$$

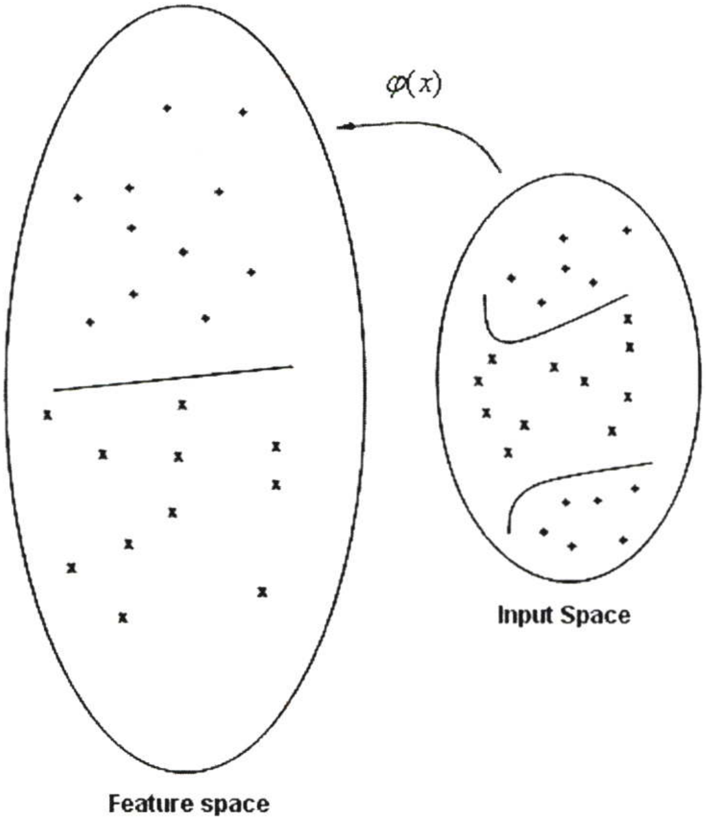
โดยทำการแทนที่ของสมการที่ (2.13) สำหรับค่า w ในสมการของลากรองจ์ (2.11) ซึ่งได้มาจากสมการที่ (2.15) ดังต่อไปนี้

$$\left[\begin{array}{ll} \max\limits_{\alpha} J_D(\alpha) = -\frac{1}{2}\sum\limits_{k,l=1}^{N} y_k y_l x_k^T x_l \alpha_k \alpha_l + \sum\limits_{k=1}^{N}\alpha_k \\ \text{such that} \quad \sum\limits_{k=1}^{N}\alpha_k y_k = 0,\ \alpha_k \geq 0,\ \forall k \end{array}\right] \tag{2.15}$$

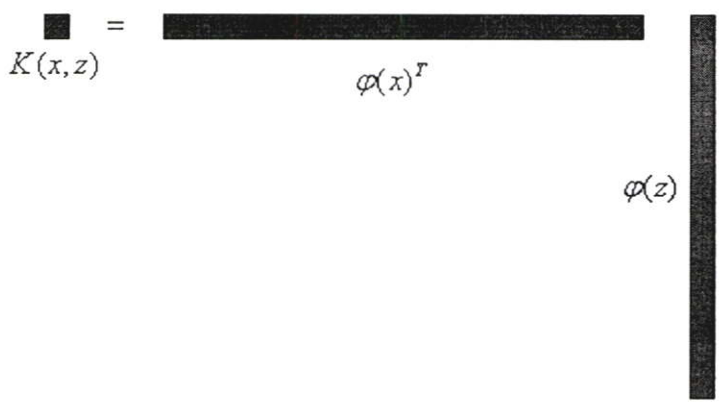
โดยที่ปัญหานี้ถูกแก้ปัญหาใน $\alpha = [\alpha_i;...;\alpha_N]$ โดยไม่ได้ใช้ค่าถ่วงน้ำหนัก w

2.3.4 การแบ่งประเภทด้วยหลักการซัพพอร์ตเวกเตอร์แมชชีนแบบไม่เป็นเชิงเส้น

ส่วนที่เพิ่มเติมจากการแบ่งประเภทข้อมูล ด้วยหลักการซัพพอร์ตเวกเตอร์แมชชีนแบบเป็นเชิงเส้นไปเป็นการแบ่งประเภทข้อมูลด้วยหลักการซัพพอร์ตเวกเตอร์แมชชีนแบบไม่เป็นเชิงเส้นนั้นค่อนข้างที่จะตรงไปตรงมา โดยสามารถที่จะแทนที่ค่า x ด้วย $\varphi(x)$ และนำหลักการของ เคอร์เนล (Kernel) มาใช้ในส่วนที่เป็นไปได้ อย่างไรก็ตามควรที่จะทราบว่า $\varphi(x)$ สามารถที่จะเป็นมิติที่ไม่สิ้นสุดได้และค่าเวกเตอร์ w ก็เช่นกัน ในขณะที่หลักการซัพพอร์ตเวกเตอร์แมชชีนแบบเป็นเชิงเส้นสามารถที่จะแก้ปัญหาได้ดีใน w ที่เป็นปัญหาในค่าซัพพอร์ต α และจะไม่เกิดปัญหาเดียวกันกับหลักการซัพพอร์ตเวกเตอร์แมชชีนแบบไม่เป็นเชิงเส้น เนื่องจากในปัญหาหลักนั้นค่า w สามารถที่จะเป็นมิติที่ไม่สิ้นสุดได้ [14]

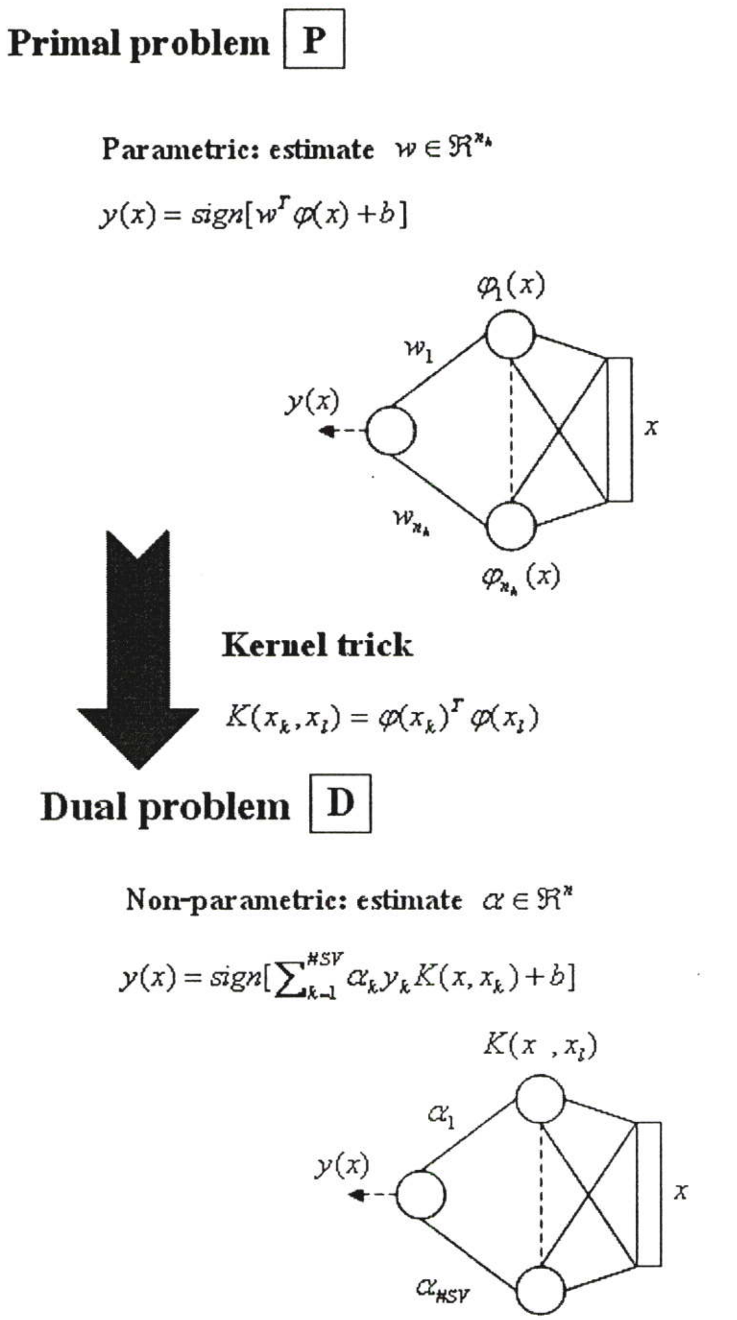


(ก)

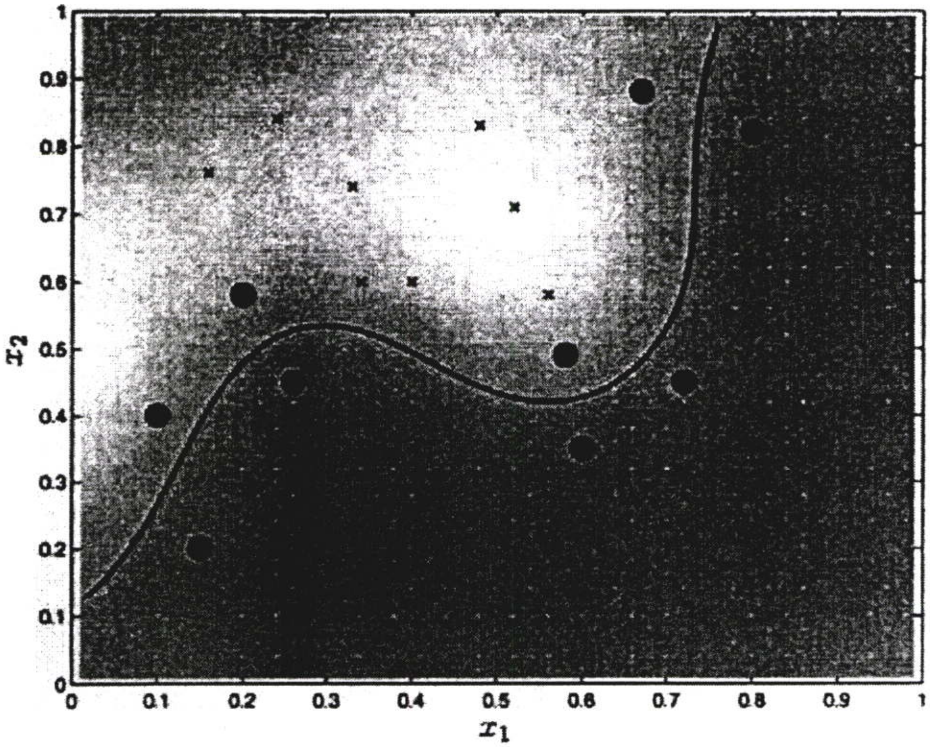


(ข)

รูปที่ 2.13 (ก) การจับคู่ค่าอินพุทไปยังค่าฟีทเจอร์ในมิติที่สูงกว่า แบบเป็นเชิงเส้น (ข) แสดงถึงการใช้หลักการเคอร์เนล ซึ่งก็คือ K(x,z)



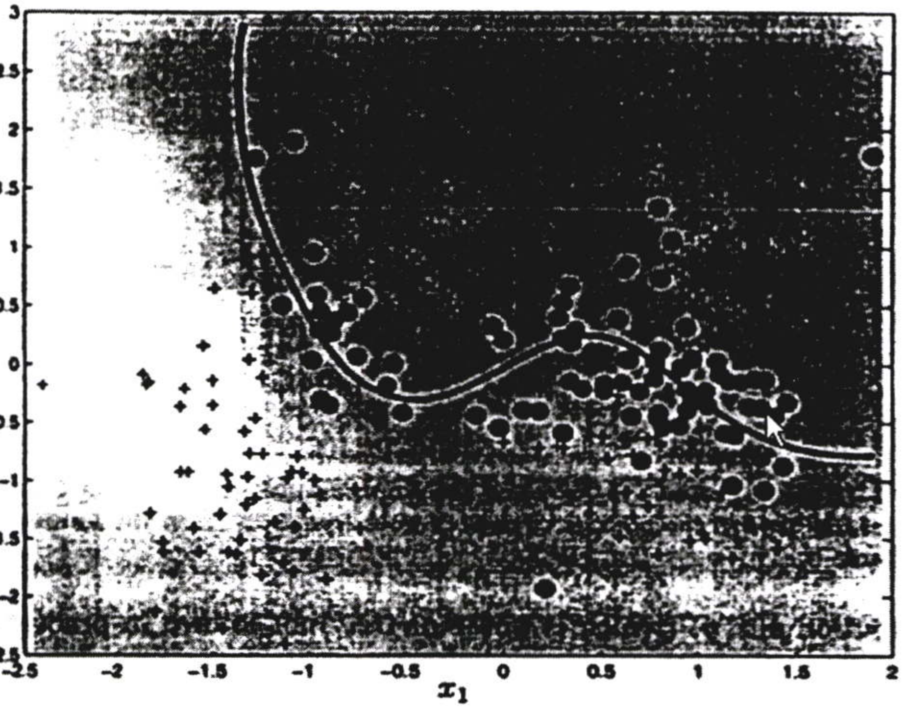
รูปที่ 2.14 การประยุกต์จากโครงข่ายประสาทเทียมมาใช้เป็นหลักการซัพพอร์ตเวกเตอร์แมชชีน [14]



รูปที่ 2.15 การแบ่งประเภทข้อมูลออกเป็นสองประเภทด้วย หลักการซัพพอร์ตเวกเตอร์แมชชีน แบบที่ไม่เป็นเชิงเส้น [14]

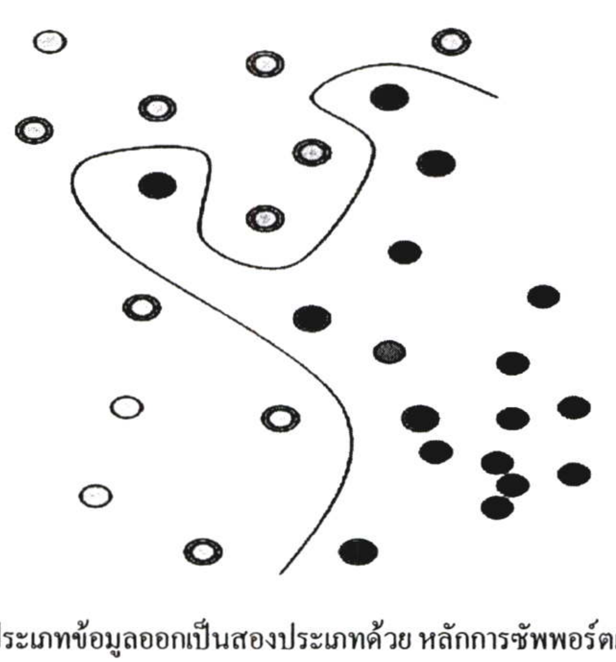
โดยรูปที่ 2.13 ได้แสดงถึงลักษณะทางกายภาพในการแก้ปัญหาของการจัดแบ่งประเภทข้อมูลแบบที่ไม่เป็นเชิงเส้น เพื่อที่จะให้เห็นมุมมองได้ชัดเจนขึ้น โดยรูปที่ 2.13 (ก) เป็นการจับคู่ค่าอินพุทไปยังค่าฟีทเจอร์ในมิติที่สูงกว่าแบบเป็นเชิงเส้น และรูปที่ 2.13 (ข) แสดงถึงการนำหลักการเคอร์เนล ซึ่งก็คือ K(x,z) เข้ามาใช้ช่วยในการจับคู่ค่าอินพุทกับค่าฟีทเจอร์ในมิติที่สูงกว่า นอกจากนั้นรูปที่ 2.14 ยังแสดงถึงการประยุกต์จากโครงข่ายประสาทเทียมมาใช้เป็นหลักการซัพพอร์ตเวกเตอร์แมชชีน

สำหรับรูปที่ 2.15 ได้แสดงถึงการแบ่งประเภทข้อมูลออกเป็นสองประเภทโดยนำหลักการของซัพพอร์ตเวกเตอร์แมชชีนแบบที่ไม่เป็นเชิงเส้นมาใช้ โดยใช้โปรแกรมทางคอมพิวเตอร์เป็นผู้ดำเนินการ และรูปที่ 2.16 แสดงถึงการแบ่งประเภทของข้อมูลออกเป็นสองประเภท โดยนำหลักการของซัพพอร์ตเวกเตอร์แมชชีนแบบที่ไม่เป็นเชิงเส้นมาใช้ร่วมกับหลักการเคอร์เนล แบบอาร์บีเอฟ (RBF)



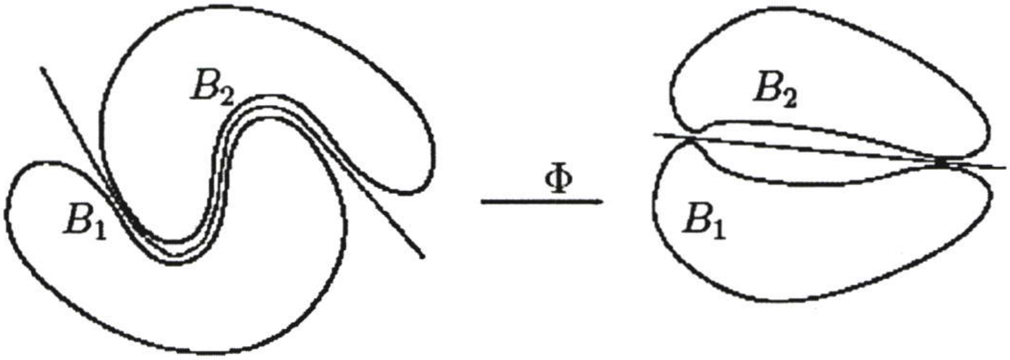
รูปที่ 2.16 การแบ่งประเภทข้อมูลออกเป็นสองประเภทด้วย หลักการซัพพอร์ตเวกเตอร์แมชชีน แบบที่ไม่เป็นเชิงเส้น และ เคอร์เนลแบบอาร์บีเอฟ (RBF) [12]

วิธีที่ง่ายที่สุดในการแยกประเภทออกเป็นสองกลุ่มคือการใช้เส้นตรง พื้นแบนราบ หรือจำนวนมิติของระนาบ แต่ถ้าข้อมูลที่ถูกแบ่งนั้นถูกทำการแบ่งด้วยขอบเขตที่ไม่เป็นเชิงเส้นจะดังรูปที่ 2.17



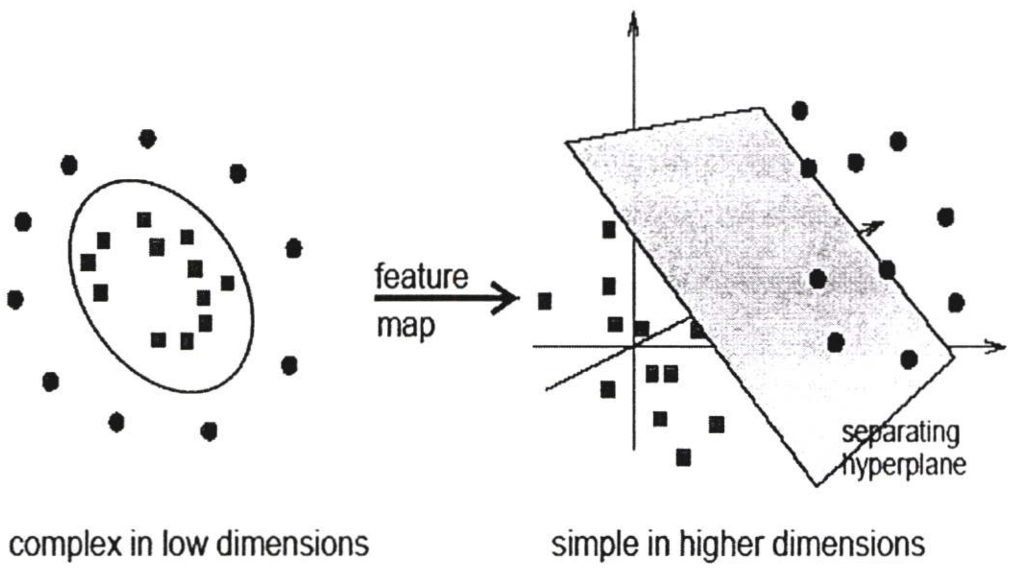
รูปที่ 2.17 การแบ่งประเภทข้อมูลออกเป็นสองประเภทด้วย หลักการซัพพอร์ตเวกเตอร์แมชชีน แบบที่ไม่เป็นเชิงเส้น [22]

ซึ่งในกรณีนี้นั้น จะต้องการเส้นแบ่ง ที่เป็นแบบที่ไม่เป็นเชิงเส้นมาใช้ในการแบ่งแยกประเภทข้อมูล นอกจากนั้นจะต้องสร้างรูปแบบของเส้นแบ่ง แบบที่ไม่เป็นเชิงเส้นให้ได้เหมาะสม โดยที่นำหลักการซัพพอร์ตเวกเตอร์แมชชีน มาใช้โดยนำฟังก์ชัน เคอร์เนล มาเป็นตัวจับคู่ข้อมูลไปยังพื้นที่ต่าง ๆ กันไป โดยที่ระนาบจะสามารถถูกนำมาใช้เป็นตัวแบ่งประเภทดังรูปที่ 2.18



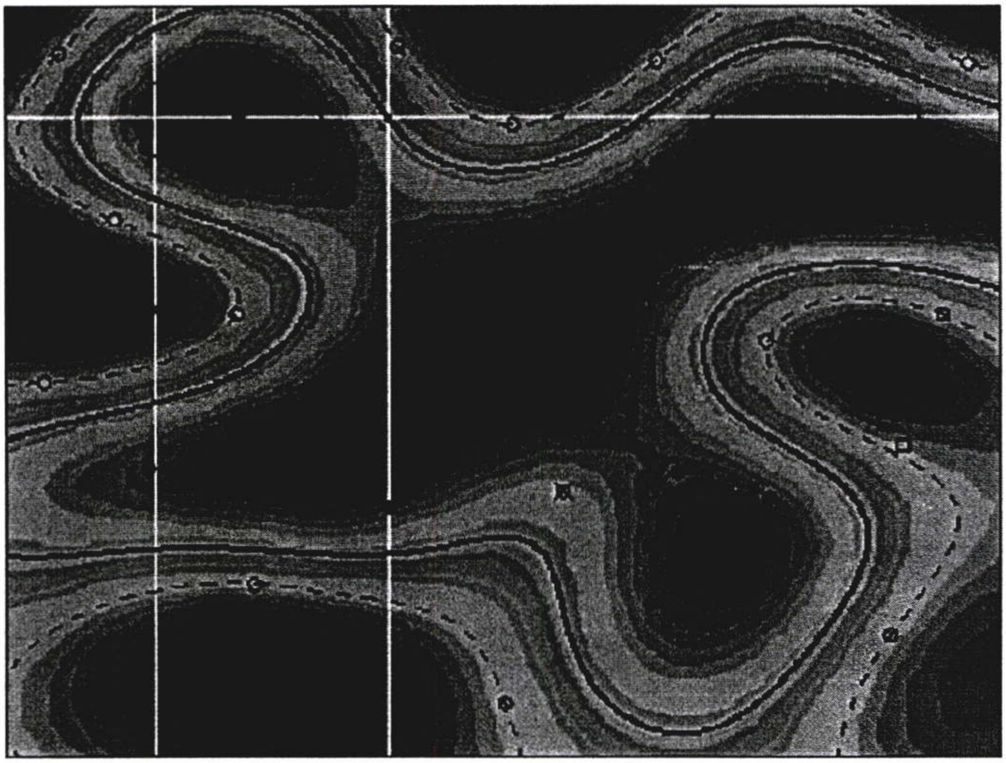
รูปที่ 2.18 การแบ่งประเภทข้อมูลออกเป็นสองประเภทด้วย หลักการซัพพอร์ตเวกเตอร์แมชชีน แบบที่ไม่เป็นเชิงเส้น ด้วย เคอร์เนล [22]

ฟังก์ชันเคอร์เนล อาจจะนำพาข้อมูลไปยังพื้นที่มิติที่สูงกว่าโดยสร้างความน่าจะเป็นของการกระทำการแบ่งประเภทข้อมูลดังรูปที่ 2.19 ของ Florian Makowetz



รูปที่ 2.19 การแบ่งประเภทข้อมูลออกเป็นสองประเภทด้วย หลักการซัพพอร์ตเวกเตอร์แมชชีน แบบที่ไม่เป็นเชิงเส้น ด้วย เคอร์เนล [22]

หลักการของการจับคู่ด้วยฟังก์ชันเคอร์เนลนั้น เป็นวิธีที่มีประสิทธิภาพมาก ซึ่งวิธีนี้ได้ให้แบบจำลองของ หลักการซัพพอร์ตเวกเตอร์แมชชีน มากระทำการแบ่งประเภท ถึงแม้ว่าขอบเขตค่อนข้างที่จะซับซ้อนดังที่แสดงดังรูปที่ 2.20



รูปที่ 2.20 การแบ่งประเภทข้อมูลออกเป็นสองประเภทด้วย หลักการซัพพอร์ตเวกเตอร์แมชชีน แบบที่ไม่เป็นเชิงเส้น ด้วย เคอร์เนล โดยที่ค่าขอบเขตค่อนข้างที่จะซับซ้อน [22]

ในทางที่คล้ายกันสำหรับกรณี หลักการซัพพอร์ตเวกเตอร์แมชชีน [8] แบบเชิงเส้นนั้น สามารถที่จะเขียนสำหรับกรณีที่ไม่เป็นเชิงเส้นได้ดังสมการที่ (2.16) นี้

$$
\begin{aligned}
w^T\varphi(x_k)+b &\geq +1, \quad \text{ถ้า } y_k = +1 \\
w^T\varphi(x_k)+b &\leq -1, \quad \text{ถ้า } y_k = -1
\end{aligned}
\tag{2.16}
$$

ซึ่งเท่ากับสมการที่ (2.17) นี้

$$
y_k\left[w^T\varphi(x_k)+b\right] \geq 1 \qquad k=1,\ldots,N. \tag{2.17}
$$

ในกรณีที่สามารถแบ่งประเภทข้อมูลได้นั้น ณ จุดนี้ไม่มีสมการรูปแบบไหนที่ถูกสร้างของ $\varphi(\cdot):\Re^{n}\rightarrow\Re^{n_h}$ ที่จะแสดงออกมาได้อย่างชัดเจน โดยที่ n_h เป็นจำนวนของหน่วยซ่อนในพื้นที่ของนำหนักที่ได้แสดงถึงการแบ่งประเภทข้อมูลแบบไม่เป็นเชิงเส้นดังรูปที่ 2.5

$$y(x)=sign[w^{T}\varphi(x)+b] \tag{2.18}$$

โดยที่ n_h มีลักษณะเดียวกันกับมิติ n_H
ดังนั้นจะเป็นดังสมการที่ (2.19) นี้

$$\left[\begin{array}{l} \min\limits_{w,b,\xi} J_p(w,\xi)=\frac{1}{2}w^{T}w+c\sum\limits_{k=1}^{N}\xi_k \\ such\ \ that \qquad y_k\left[w^{T}\varphi(x_k)+b\right]\geq 1-\xi_k,\quad k=1,\ldots,N \\ \qquad\qquad\qquad \xi_k\geq 0,\qquad k=1,\ldots,N \end{array}\right] \tag{2.19}$$

สมการของลากรองจ์ สำหรับปัญหานี้คือ

$$L(w,b,\xi;\alpha,v)=J_p(w,\xi)-\sum_{k=1}^{N}\alpha_k\left(y_k\left[w^{T}\varphi(x_k)+b\right]-1+\xi_k\right)-\sum_{k=1}^{N}v_k\xi_k \tag{2.20}$$

ด้วยตัวคูณของสมการของลากรองจ์ $\alpha_k\geq 0, v_k\geq 0$ โดยที่ค่า $k=1,\ldots,N.$ และวิธีการแก้ปัญหานี้ จะมีลักษณะเฉพาะซึ่งถูกรับผิดชอบโดยสมการของลากรองจ์ ดังนี้

$$\max_{\alpha,v}\ \min_{w,b,\xi} L(w,b,\xi;\alpha,v). \tag{2.21}$$

ซึ่งได้มาจาก

$$\begin{cases} \dfrac{\partial L}{\partial w} = 0 \rightarrow \ w = \sum_{k=1}^{N} \alpha_k y_k \varphi(x_k) \\ \dfrac{\partial L}{\partial b} = 0 \rightarrow \ \sum_{k=1}^{N} \alpha_k y_k = 0 \\ \dfrac{\partial L}{\partial \xi_k} = 0 \rightarrow \ 0 \le \alpha_k \le c, k = 1,...,N. \end{cases} \tag{2.22}$$

โดยที่ปัญหาของสมการกำลังสองจะเปลี่ยนเป็นสมการที่ (2.23) ดังนี้

$$\begin{bmatrix} \max\limits_{\alpha} \ J_D(\alpha) = -\frac{1}{2} \sum_{k,l=1}^{N} y_k y_l K(x_k x_l) \alpha_k \alpha_l + \sum_{k=1}^{N} \alpha_k \\ such \ \ that \qquad \sum_{k=1}^{N} \alpha_k y_k = 0, \ \ 0 \le \alpha_k \le c, k = 1,...,N \end{bmatrix} \tag{2.23}$$

ในรูปแบบของสมการกำลังสองนี้ได้ถูกสร้างโดยใช้หลักการ เคอร์เนล ดังสมการที่ (2.24)

$$K(x_k x_l) = \varphi(x_k)^T \varphi(x_l) \tag{2.24}$$

สำหรับค่า $k, l = 1,...,N$ ซึ่งในที่สุดแล้ว สมการของการแบ่งประเภทข้อมูลด้วย หลักการซัพพอร์ตเวกเตอร์แมชชีน แบบไม่เป็นเชิงเส้นคือ

$$y(x) = sign[\sum_{k=1}^{N} \alpha_k y_k K(x, x_k) + b] \tag{2.25}$$

ด้วยค่า α_k ซึ่งเป็นค่าคงที่จำนวนจริงบวก และเป็นการแก้ปัญหาจากสมการยกกำลังสอง (Quadratic Problem : QP) [14] อย่างไรก็ตาม ยังคงต้องการที่จะหาค่า b ด้วยเงื่อนไขที่สมบูรณ์จากระบบเคเคที (Karush-Kuhn-Tucker : KKT) เพื่อที่จะนำมาแก้ปัญหานั้น เป็นดังนี้

$$\begin{cases} \dfrac{\partial L}{\partial w} = 0 \rightarrow w = \sum_{k=1}^{N} \alpha_k y_k \varphi(x_k) \\ \dfrac{\partial L}{\partial b} = 0 \rightarrow \sum_{k=1}^{N} \alpha_k y_k = 0 \\ \dfrac{\partial L}{\partial \xi_k} = 0 \rightarrow c - \alpha_k - v_k = 0 \\ \qquad \alpha_k \{ y_k [w^T \varphi(x_k) + b] - 1 + \xi_k \} = 0 \ , k = 1,\dots,N. \\ \qquad v_k \xi_k = 0, \qquad\qquad k = 1,\dots,N. \\ \qquad \alpha_k \geq 0, \qquad\qquad k = 1,\dots,N. \\ \qquad v_k \geq 0, \qquad\qquad k = 1,\dots,N. \end{cases} \tag{2.26}$$

จาก $v_k \xi_k = 0$ จะมีค่า $w^*, b^*, \xi^*, e\alpha^*, v^*$ มาใช้สำหรับวิธีการแก้ปัญหานี้ โดยที่ค่า $\xi_k^* = 0$ และ $\alpha_k^* \in (0,c)$ เนื่องจากสมการที่ (2.27) ดังนี้

$$y_k [w^T \varphi(x_k) + b] - 1 = 0 \text{ สำหรับ } \alpha_k \in (0,c) \tag{2.27}$$

ซึ่งหมายความได้ว่า สมการนี้สามารถที่จะนำข้อมูลในชุดของการเทรนนิ่งส่วนไหนก็ได้ สำหรับ $0 \leq \alpha_k \leq c$ และใช้สมการนั้นมาคำนวณค่าไบแอซ b

คุณสมบัติของการแบ่งประเภทข้อมูล ด้วยหลักการซัพพอร์ตเวกเตอร์แมชชีนแบบไม่เป็นเชิงเส้นมีดังนี้

2.3.4.1 ทางเลือกของฟังก์ชันเคอร์เนล

ฟังก์ชันเคอร์เนล $K(\cdot,\cdot)$ ที่สามารถนำมาใช้ได้นั้นมีอยู่หลายประเภทด้วยกัน เช่น

1. แบบเชิงเส้น $K(x, x_k) = x_k^T x$
2. แบบโพลิโนเมียล (Polynomial) $K(x, x_k) = (\tau + x_k^T x)^d$
3. แบบอาร์บีเอฟ (RBF) $K(x, x_k) = \exp(-\|x - x_k\|_2^2)/\sigma^2$
4. แบบเอ็มแอลพี (MLP) $K(x, x_k) = \tanh(k_1 x_k^T x + k_2)$

ซึ่งจะกล่าวถึงหลักการทำงานของเคอร์เนลโดยละเอียดต่อไป สำหรับเงื่อนไขของเมอร์เซอร์ (Mercer) จะยึดถือสำหรับค่า σ ในกรณีที่เคอร์เนลแบบอาร์บีเอฟ และค่าที่ τ เป็นบวก ในแบบโพลิโนเมียล แต่ไม่ใช่สำหรับค่า k_1, k_2 ที่เป็นไปได้ทั้งหมดในแบบเอ็มแอลพี

2.3.4.2 วิธีการแก้ปัญหาที่ใช้กันทั่วไป กับใช้ได้ในบางกรณี

จากที่วิธีแก้ปัญหาในกรณีของ หลักการซัพพอร์ตเวกเตอร์แมชชีน แบบเป็นเชิงเส้นกับสมการยกกำลังสอง (QP) ที่เป็นแบบใช้ได้ทั่วไปและใช้ได้ในบางกรณีนั้นได้ จะใช้วิธีเลือกค่าเคอร์เนล ที่เป็นบวกสำหรับ $K(\cdot,\cdot)$ วิธีนี้ได้รับรองว่าเมทริกซ์ในสมการยกกำลังสองนั้นเป็นค่าบวกอย่างแน่นอนและสามารถที่จะนำหลักการ เคอร์เนล มาใช้ได้เช่นกัน

2.3.4.3 สปลาสเนส (Sparseness)

จากที่วิธีแก้ปัญหาในกรณีของ หลักการซัพพอร์ตเวกเตอร์แมชชีน แบบเป็นเชิงเส้นนั้น ค่า α_k หลายตัวมีค่าเท่ากับศูนย์ในเวกเตอร์ ทำให้วิธีแก้ปัญหาในกรณีของ หลักการซัพพอร์ตเวกเตอร์แมชชีน แบบไม่เป็นเชิงเส้น สามารถใช้สมการที่ (2.28) ได้ดังนี้

$$y(x) = sign[\sum_{k=1}^{\#SV} \alpha_k y_k K(x, x_k) + b] \tag{2.28}$$

ในขณะที่ผลรวมได้ถูกนำมาใช้มากกว่าค่า α_k ที่ไม่เท่ากับศูนย์ซึ่งเป็นไปในทางเดียวกัน x_k ของข้อมูลในส่วนของการฝึกฝน จากรูปที่ 2.14 โครงข่ายประสาทเทียมได้ทำให้เกิดภาพนี้ จำนวนของหน่วยซ่อนท่ากับจำนวนของซัพพอร์ตเวกเตอร์ (SV) ที่ได้ค้นหาจากสมการยกกำลังสองนั่นเอง ในกรณีของเคอร์เนลแบบอาร์บีเอฟ จะได้

$$\begin{aligned} y(x) &= sign[\sum_{k=1}^{N} \alpha_k y_k \exp\left(-\|x - x_k\|_2^2 / \sigma^2\right) + b] \\ &= sign[\sum_{k=1}^{\#SV} \alpha_k y_k \exp\left(-\|x - x_k\|_2^2 / \sigma^2\right) + b] \end{aligned} \tag{2.29}$$

โดยที่ $(\#SV)$ เป็นจำนวนของซัพพอร์ตเวกเตอร์ (SV)

2.3.4.4 รูปทรงเรขาคณิต

ซึ่งหมายถึงซัพพอร์ตเวกเตอร์ นั่นเอง

ซัพพอร์ตเวกเตอร์ ได้มาจากปัญหาของสมการยกกำลังสองที่อยู่ใกล้กับเส้นแบ่งเขตที่ไม่เป็นเชิงเส้นที่แสดงดังรูปที่ 2.15 และรูปที่ 2.16

2.3.4.5 ความเป็นรูปแบบของตัวแปร และไม่เป็นตัวแปร (Parametric and Non-parametric)

ทั้งสองปัญหาหลักและรองนั้น มีโครงข่ายประสาทเทียมแสดงอยู่ดังรูปที่ 2.14 โดยที่ปัญหาหลักคือค่าน้ำหนักเป็นรูปแบบของตัวแปร ในขณะที่ในปัญหารองคือรูปแบบที่ไม่เป็นตัวแปร

2.3.4 เคอร์เนล (Kernel)

การนำฟังก์ชันเคอร์เนล [9], [14] มาใช้นั้น เพื่อต้องการที่จะให้ หลักการซัพพอร์ตเวกเตอร์แมชชีน สามารถแบ่งประเภทข้อมูลแบบไม่เป็นเชิงเส้นได้อย่างมีประสิทธิภาพ โดยปราศจากผลกระทบ ถ้าข้อมูลเป็นแบบมิติที่ไม่มีที่สิ้นสุดได้ สำหรับการแยกระหว่างตัวแปรทั้งสองชุด จะเรียกค่าอินพุทที่มีอยู่แล้วว่าค่าคุณสมบัติของปัญหา (ในกรณีนี้คือ x) เมื่อค่าเหล่านั้นได้ถูกจับคู่กับค่าอินพุทใหม่จำนวนหนึ่ง ที่ได้ถูกส่งผ่านมายังส่วนของวิธีการหัดเรียนรู้ (Learning Algorithm) ซึ่งจะเรียกค่าอินพุทใหม่เหล่านี้ว่า ฟีทเจอร์ โดยที่จะกำหนดให้ ϕ เป็นตัวจับคู่ ฟีทเจอร์ ซึ่งจับคู่จากค่าคุณสมบัติไปยังค่าฟีทเจอร์ ยกตัวอย่างเช่น

$$\phi(x) = \begin{bmatrix} x \\ x^2 \\ x^3 \end{bmatrix} \tag{2.30}$$

นอกจากที่จะใช้ หลักการซัพพอร์ตเวกเตอร์แมชชีน กับค่าอินพุทแล้ว ค่าคุณสมบัติดั้งเดิม x อาจจะเรียนรู้เกี่ยวกับค่าฟีทเจอร์ $\phi(x)$ ด้วย โดยทำการแทนที่ค่า x ด้วย $\phi(x)$ ในทุก ๆ ที่ของสมการ

จากอัลกอลิธึมที่ผ่าน ๆ มาได้ถูกเขียนในลักษณะของอินเนอร์ พรอดักท์ (Inner product) $\langle x,z \rangle$ ซึ่งหมายความว่าจะแทนที่อินเนอร์ พรอดักท์ทั้งหมดด้วย $\langle \phi(x),\phi(z) \rangle$ โดยเฉพาะให้ค่าการจับคู่ฟีทเจอร์ ϕ เป็น เคอร์เนล ฟังก์ชันดังนี้

$$K(x,z) = \phi(x)^T \phi(z) \tag{2.31}$$

จากนั้นทุกที่ที่มีค่า $\langle x,z \rangle$ อยู่ในอัลกอริธึม ก็สามารถที่จะแทนที่ด้วยค่า $K(x,z)$ และอัลกอลิธึม จะกลายเป็นการเรียนรู้โดยใช้ ฟีทเจอร์ ϕ

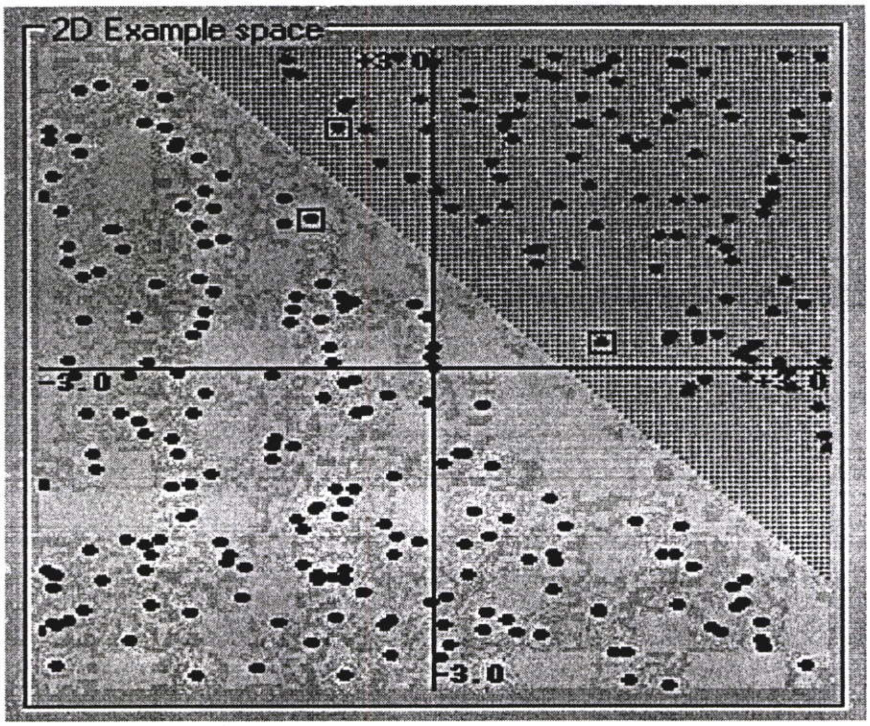
มีหลายฟังก์ชันเคอร์เนล ที่นำมาใช้ในการจับคู่ของข้อมูลถูกนำมาใช้ ซึ่งบางทีอาจจะเป็นจำนวนที่นับไม่ได้เลยก็เป็นได้ แต่ว่ามีอยู่เพียงไม่กี่ฟังก์ชันที่ค้นพบได้ว่าสามารถนำมาใช้งานได้ดี

ในหลาย ๆ การประยุกต์ใช้งาน ซึ่งฟังก์ชันเคอร์เนล ที่ถูกนำมาใช้ตลอดและถูกแนะนำให้ใช้คือ อาร์บีเอฟ (RBF) ซึ่งย่อมาจาก เรเดียล เบสิส ฟังก์ชัน (Radial Basis Function)

โดยรูปที่ 2.21 ถึงรูปที่ 2.27 จะแสดงถึงประเภทต่าง ๆ ของฟังก์ชันเคอร์เนล ที่มีอยู่แล้วและสามารถนำมาใช้งานได้จริง ซึ่งแต่ละรูปจะแสดงถึงฟังก์ชันเคอร์เนลในรูปแบบของกราฟ ที่ได้ถูกนำมาใช้ร่วมกับหลักการของซัพพอร์ตเวกเตอร์แมชชีน เพื่อที่จะได้เห็นถึงลักษณะความแตกต่างในการแบ่งประเภทข้อมูลด้วยการนำฟังก์ชันเคอร์เนลชนิดต่าง ๆ มาใช้ ดังนี้

2.3.4.1 แบบเชิงเส้น (Linear): $u'*v$

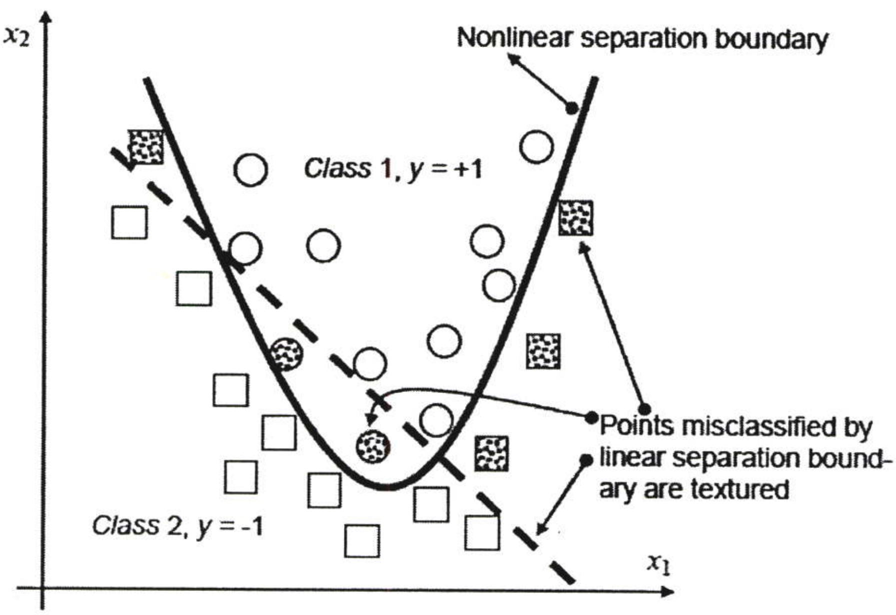
ซึ่งจากรูปที่ 2.21 เห็นได้ว่าข้อมูลทั้งสองประเภทได้ถูกแบ่งในรูปแบบของแนวเชิงเส้นตรง ซึ่งเกิดจากฟังก์ชันเคอร์เนลในแบบเชิงเส้นนั่นเอง



รูปที่ 2.21 ฟังก์ชัน เคอร์เนล แบบเชิงเส้น [22]

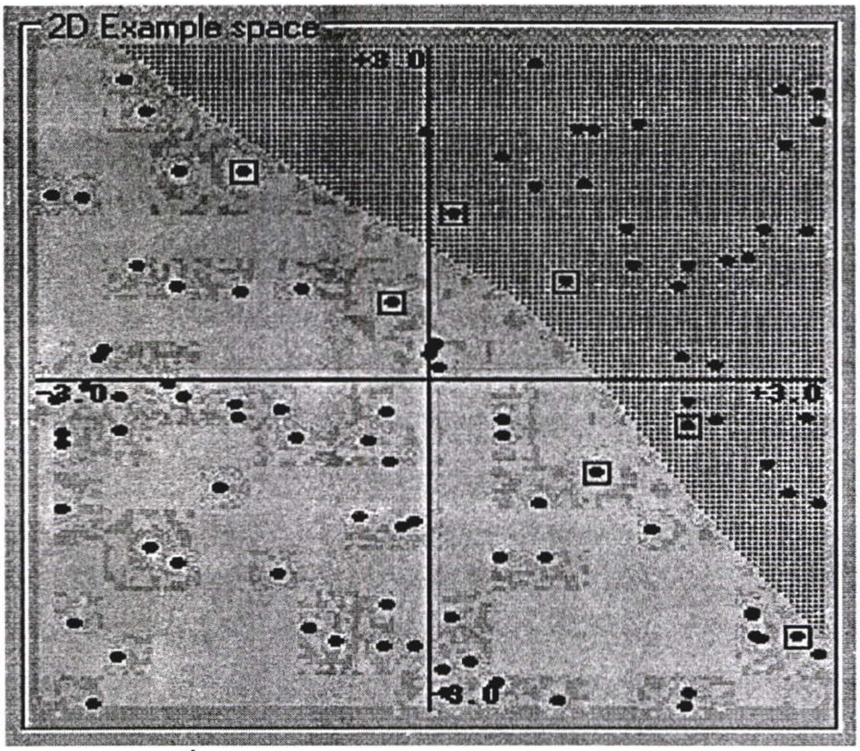
2.3.4.2 โพลิโนเมียล (Polynomial): $(gamma * u'*v + coef0)^\wedge degree$

ฟังก์ชันเคอร์เนลในรูปแบบของโพลิโนเมียลนั้นจะเห็นได้จากรูปที่ 2.22 โดยที่ข้อมูลเหล่านี้ได้ถูกแบ่งประเภทด้วยรูปแบบที่ไม่เป็นเชิงเส้น และแบบเชิงเส้น ซึ่งจะเห็นว่าการที่แบ่งข้อมูลด้วยโพลิโนเมียลนั้น สามารถที่จะตรวจสอบพบว่าการแบ่งแบบเชิงเส้นนั้น มีค่าความผิดพลาดอยู่ โดยที่ค่าจากรูป  และ  แสดงถึงการแบ่งผิดประเภทซึ่งเกิดจากการแบ่งแบบเชิงเส้นนั่นเอง



รูปที่ 2.22 ฟังก์ชัน เคอร์เนล แบบ โพลิโนเมียล อ้างอิงจาก Kecman ปีค.ศ. 2004 [24]

และรุปที่ 2.23 แสดงถึงการใช้ฟังก์ชันเคอร์เนล แบบโพลิโนเมียลร่วมกับหลักการของซัพพอร์ตเวกเตอร์แมชชีน

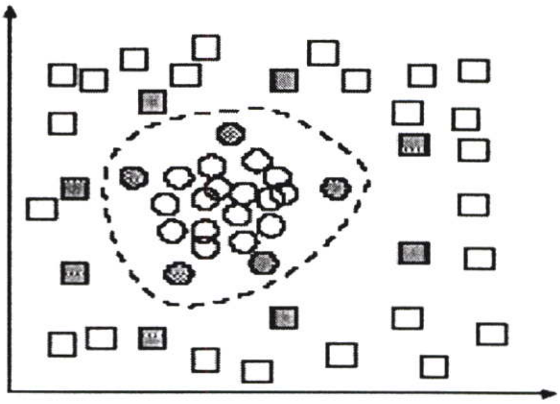


รูปที่ 2.23 ฟังก์ชันเคอร์เนล แบบโพลิโนเมียล [22]

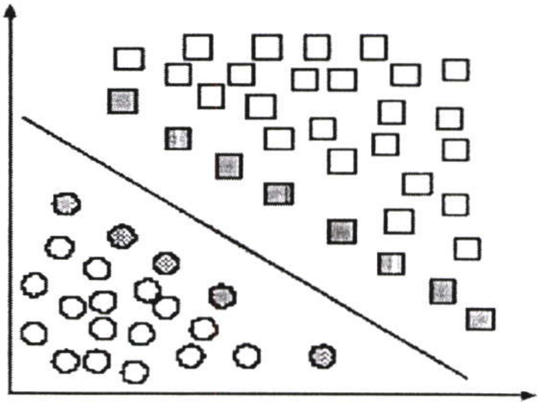
2.3.4.3 เรเดียล เบสิส ฟังก์ชัน (Radial Basis Function : RBF)

$$\exp(-gamma * |u * v|)^{\wedge}2$$

อาร์บีเอฟเป็นฟังก์ชันที่มีผู้นิยมใช้เป็นจำนวนมาก โดยที่ฟังก์ชันเคอร์เนลแบบอาร์บีเอฟนี้จะจับคู่กับตัวอย่างข้อมูลไปยังพื้นที่ของมิติที่สูงกว่า ดังนั้น ฟังก์ชันนี้สามารถที่จะทำงานร่วมกันกับความสัมพันธ์แบบที่ไม่เป็นเชิงเส้น ระหว่างค่าประเภทเป้าหมายและค่าคุณสมบัติที่เป็นตัวทำนายได้ โดยที่พื้นฐานของฟังก์ชันแบบเชิงเส้นนั้นไม่สามารถทำได้ นอกจากนั้น ฟังก์ชันเคอร์เนล แบบเชิงเส้นยังเป็นกรณีพิเศษของอาร์บีเอฟ และฟังก์ชันเคอร์เนลแบบซิกมอยด์ (Sigmoid function) โดยจะทำตัวเหมือนกันกับฟังก์ชันอาร์บีเอฟในด้านของการเก็บตัวแปร โดยที่ฟังก์ชันอาร์บีเอฟจะมีตัวแปรอยู่เพียงไม่กี่ตัวที่จะทำตัวเหมือนฟังก์ชันโพลิโนเมียล และนอกจากนั้นฟังก์ชันอาร์บีเอฟจะมีจำนวนตัวเลขที่คำนวณได้ยาก ๆ อยู่น้อย โดยที่รูปที่ 2.24 ของ Yang, 2003 [23] ได้แสดงถึงการจับคู่ข้อมูลด้วยฟังก์ชันอาร์บีเอฟ



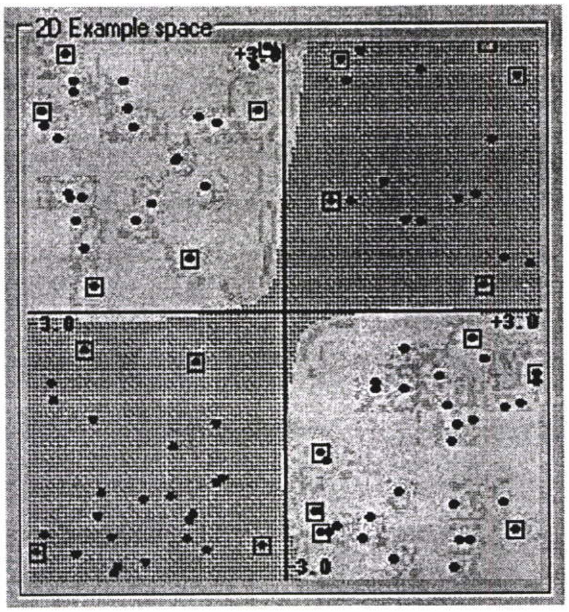
(ก)



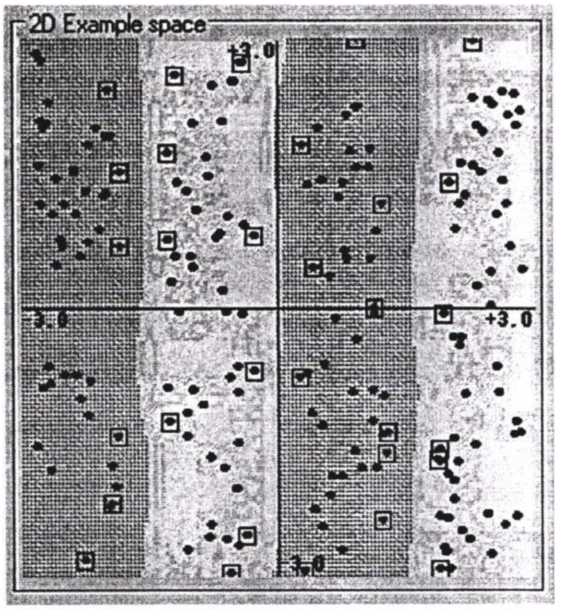
(ข)

รูปที่ 2.24 ฟังก์ชัน เคอร์เนลแบบอาร์บีเอฟ จาก Yang, 2003 [23] (ก) อาร์บีเอฟ (ข) การจับคู่ด้วยฟังก์ชัน อาร์บีเอฟ [22]

การจำแนกประเภทข้อมูลด้วยฟังก์ชันเคอร์เนล แบบอาร์บีเอฟ ในพื้นที่ที่แตกต่างกันนั้น ได้ถูกแสดงดังรูปที่ 2.25



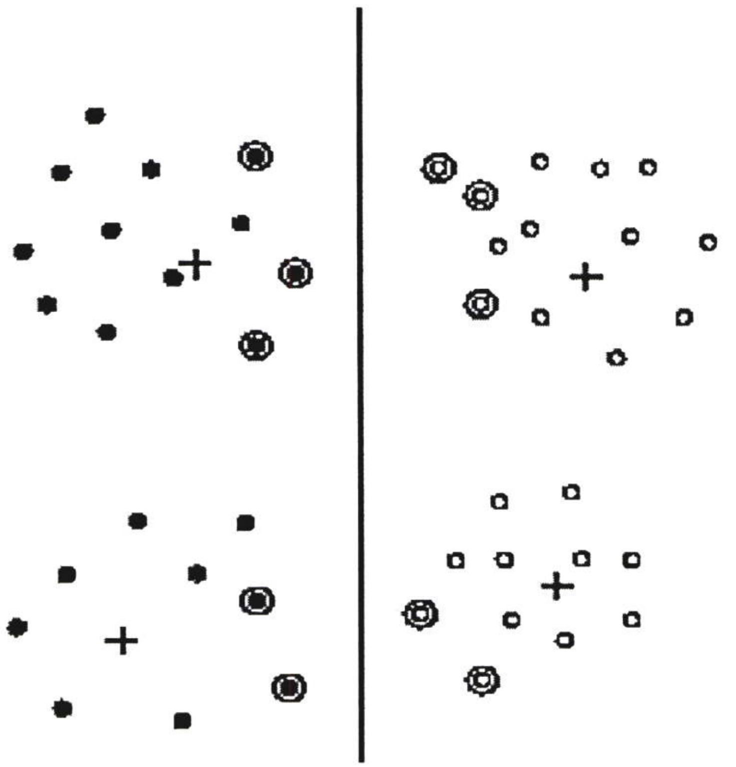
(ก)



(ข)

รูปที่ 2.25 เคอร์เนลอาร์บีเอฟ (ก) พื้นที่เริ่มต้น (ข) พื้นที่แบบที่มีฟีทเจอร์ [22]

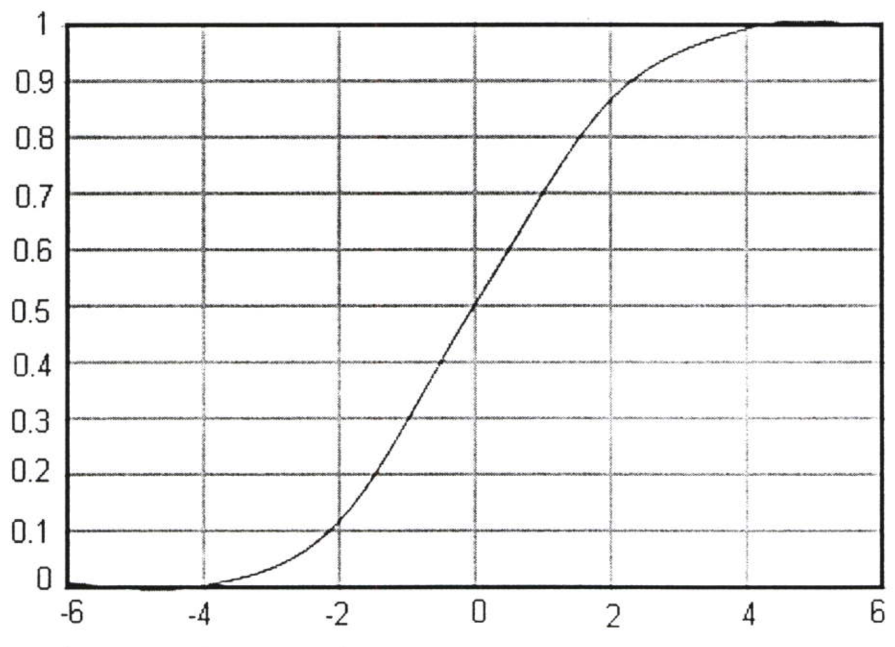
แบบจำลองของ หลักการซัพพอร์ตเวกเตอร์แมชชีนที่ใช้เคอร์เนลแบบอาร์บีเอฟจะมีโครงสร้างแบบโครงข่ายของอาร์บีเอฟ [10] ถึงแม้ว่าวิธีที่จะหาจำนวนโหนดและจุดศูนย์กลางนั้นจะแตกต่างจากมาตรฐานของโครงข่ายของอาร์บีเอฟทั่วไป ซึ่งจุดศูนย์กลางของอาร์บีเอฟ จะขึ้นอยู่กับตัวของซัพพอร์ตเวกเตอร์ ซึ่งดูได้จากรูปที่ 2.26 ของ C. Campbell [25] ด้วยมาตรฐานของโครงข่ายอาร์บีเอฟ จะค้นหาจุดศูนย์กลางของอาร์บีเอฟโหนด ด้วยค่า k ซึ่งก็คือ จุดที่เป็นเครื่องหมายบวก และในทางกลับกัน หลักการซัพพอร์ตเวกเตอร์แมชชีนที่ใช้อาร์บีเอฟเคอร์เนลนั้นจะใช้จุดศูนย์กลางของอาร์บีเอฟโหนด บนตัวซัพพอร์ตเวกเตอร์ ซึ่งก็คือจุดที่เป็นรูปวงกลมนั่นเอง



รูปที่ 2.26 แบบจำลอง หลักการซัพพอร์ตเวกเตอร์แมชชีน ด้วย อาร์บีเอฟเคอร์เนล

2.3.4.4 ซิกมอยด์ (Sigmoid): $\tanh(gamma * u'*v + coef0)$

จากรูปที่ 2.27 ได้แสดงถึงกราฟสำหรับฟังก์ชันเคอร์เนลแบบซิกมอยด์



รูปที่ 2.27 ฟังก์ชันเคอร์เนลแบบซิกมอยด์

2.4 หลักการพื้นฐานของลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีน

ในส่วนนี้จะกล่าวถึงวิธีพื้นฐานของลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีน [11-14] ที่นำมาใช้สำหรับการแบ่งแยกประเภทของข้อมูลที่ไม่เป็นแบบเชิงเส้น ดังนี้

2.4.1 หลักการลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีนสำหรับการแบ่งประเภทข้อมูล

ลักษณะที่ดีที่ได้จากหลักการซัพพอร์ตเวกเตอร์แมชชีน แบบไม่เป็นเชิงเส้นนั้น เป็นสิ่งที่ช่วยทำให้แก้ปัญหาของการแยกประเภทข้อมูลแบบไม่เป็นเชิงเส้นได้นั่นเอง โดยที่จะเสนอการเปลี่ยนแปลงของตัวแบ่งประเภทข้อมูลด้วยหลักการซัพพอร์ตเวกเตอร์แมชชีน ที่นำมาใช้ในการแก้ปัญหาสมการเชิงเส้นซึ่งมีผู้นำไปใช้มากมายในขอบเขตที่ต่าง ๆ กันไป ซึ่งการเปลี่ยนแปลงของหลักการซัพพอร์ตเวกเตอร์แมชชีน นำเสนอโดย Suyken [14] ดังนี้

$$\begin{bmatrix} \min\limits_{w,b} J_p\left(w,e\right) = \frac{1}{2}w^T w + \gamma\frac{1}{2}\sum_{k=1}^{N} e_k^2 \\ such\ \ that \qquad y_k\left[w^T \varphi(x_k) + b\right] = 1 - e_k,\ k = 1,\ldots,N \end{bmatrix} \tag{2.32}$$

สำหรับตัวการแบ่งประเภทข้อมูลในช่องว่างแรกนั้นมาจากสมการที่ (2.33) ดังนี้

$$y(x) = sign\left[w^T \varphi(x) + b\right] \tag{2.33}$$

ในขณะที่ $\varphi(\cdot):\Re^n \longrightarrow \Re^{n_k}$ เป็นการจับคู่ไปยังมิติที่สูงกว่าซึ่งเป็นลักษณะมาตราฐานของหลักการซัพพอร์ตเวกเตอร์แมชชีน สำหรับสมการของ Vapnik [14] นั้นจะถูกนำมาเปลี่ยนแปลงเพียงสองจุด ซึ่งก็คือการแทนที่ของสมการไม่เท่ากับในขณะที่ค่า 1 อยู่ด้านขวามือ จะถูกพิจารณามากกว่าค่าเธรชโชล เนื่องจากเป็นค่าเป้าหมาย ในขณะที่ค่าความผิดพลาด e^k บนค่าเป้าหมายนี้เกิดขึ้น โดยจะทำให้การแยกประเภทข้อมูลผิดพลาดสามารถที่จะนำมาใช้ในกรณีที่มีการกระจายข้อมูลทับซ้อนกันได้ โดยที่ค่าความผิดพลาดเหล่านี้สามารถที่จะนำมาใช้ได้คล้ายกับตัวแปร ξ_k ในสมการของหลักการซัพพอร์ตเวกเตอร์แมชชีนได้ และจุดที่สองคือฟังก์ชันสแคว ลอส (Squared loss) จะถูกนำมาใช้สำหรับตัวแปรของความผิดพลาดเหล่านี้ โดยที่จะเห็นได้ว่าการเปลี่ยนแปลงของทั้งสองจุดนี้จะทำให้สามารถแก้ปัญหาเหล่านี้ได้ง่ายขึ้นอย่างเห็นได้ชัด

ในกรณีของการแบ่งประเภทข้อมูลแบบเชิงเส้น สามารถที่จะแก้ปัญหาหลักได้ง่าย ๆ แต่โดยทั่วไปแล้วค่า w อาจจะกลายเป็นมิติที่ไม่มีที่สิ้นสุดได้ ดังนั้นจะต้องนำสมการของการแบ่งประเภทข้อมูลด้วยลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีนแบบไม่เป็นเชิงเส้นมาใช้ในการแก้ปัญหา โดยสมการของลากรองจ์ สำหรับปัญหานี้คือ

$$L(w,b,e,\alpha) = J_p(w,e) - \sum_{k=1}^{N} \alpha_k \left\{ y_k \left[w^T \varphi(x_k) + b \right] - 1 + e_k \right\} \tag{2.34}$$

ในขณะที่ค่า α_k เป็นตัวคูณของ สมการของลากรองจ์ ซึ่งสามารถที่จะเป็นค่าบวกหรือค่าลบ ขึ้นอยู่กับค่าจำกัดความเท่ากัน (Equality constraint)

โดยที่เงื่อนไขที่ทำให้ได้ผลลัพธ์ที่ดีคือ

$$\begin{cases} \dfrac{\partial L}{\partial w} = 0 \rightarrow w = \sum_{k=1}^{N} \alpha_k y_k \varphi(x_k) \\ \dfrac{\partial L}{\partial b} = 0 \rightarrow \sum_{k=1}^{N} \alpha_k y_k = 0 \\ \dfrac{\partial L}{\partial e_k} = 0 \rightarrow \alpha_k = \gamma e_k, & k = 1,\ldots,N \\ \dfrac{\partial L}{\partial \alpha_k} = 0 \rightarrow y_k \left[w^T \alpha(x_k) + b \right] - 1 + e_k = 0, & k = 1,\ldots,N \end{cases} \tag{2.35}$$

โดยจะกำหนดให้ $Z^T = [\varphi(x_1)^T y_1;\ldots;\varphi(x_N)^T y_N]$, $y = [y_1;\ldots y_N], 1_v = [1;\ldots;1]$, $e = [e_1;\ldots;e_N]$, $\alpha = [\alpha_1;\ldots'\alpha_N]$, และกำจัดค่า w และ e จากสมการเชิงเส้นของระบบเคเคที (KKT) โดยการหาค่า α, b ดังสมการที่ (2.36) นี้

$$\left[\begin{array}{c|c} 0 & y^T \\ \hline y & \Omega + 1/\gamma \end{array} \right] \left[\begin{array}{c} b \\ \hline \alpha \end{array} \right] = \left[\begin{array}{c} 0 \\ \hline 1_v \end{array} \right] \tag{2.36}$$

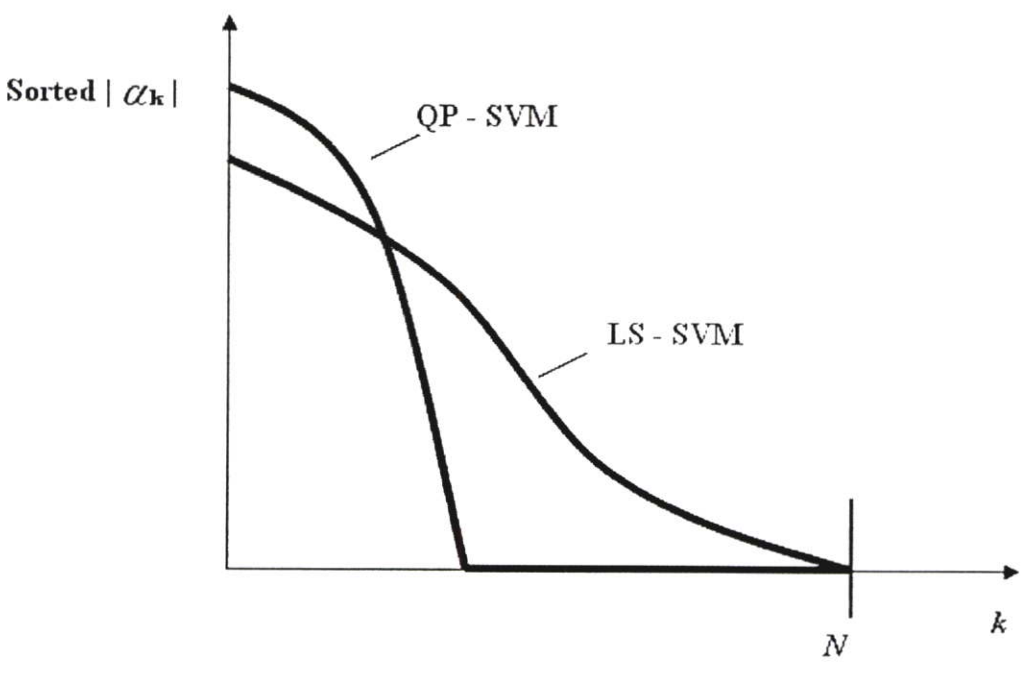
โดยที่ฟังก์ชันเคอร์เนล สามารถที่จะถูกนำมาใช้ภายในเมทริกซ์ดังนี้

$$\begin{aligned} \Omega_{kl} &= y_k y_l \varphi(x_k)^T \varphi(x_l) \\ &= y_k y_l K(x_k x_l), \;\; k,l = 1,\ldots,N \end{aligned} \tag{2.37}$$

ตัวจำแนกประเภทในสองมิติได้มาจาก

$$y(x) = sign\left[\sum_{k=1}^{N} \alpha_k y_k K(x,x_k) + b\right] \tag{2.38}$$

คล้ายคลึงกับกรณีมาตรฐานของ หลักการซัพพอร์ตเวกเตอร์แมชชีน ซึ่งสามารถที่จะแสดงการแก้ปัญหาของสมการในรูปแบบของตัวแปรค่าความผิดพลาดที่ e โดยที่ทำการกำจัด α แทน e



รูปที่ 2.28 การเปรียบเทียบคุณภาพระหว่างสมการยกกำลังสอง (QP) กับหลักการลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีน

คุณสมบัติของการแบ่งประเภทข้อมูลด้วย หลักการลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีน มีดังต่อไปนี้

2.4.1.1 ทางเลือกของฟังก์ชันเคอร์เนล

การเลือกที่จะนำฟังก์ชันเคอร์เนลมาใช้นั้น ควรพิจารณาถึง เคอร์เนล ที่มีค่าเป็นแน่นอน และมีค่าเป็นบวก รวมถึงยังสามารถที่จะนำมาใช้ได้กับเงื่อนไขของ Mercer ได้อีกด้วย ข้อแนะนำทั้งหมดเกี่ยวกับฟังก์ชันเคอร์เนล นั้นได้สามารถนำมาใช้กับ เคอร์เนล ในหลักการลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีน ได้ทั้งนั้นแต่ในที่นี้จะมุ่งเน้นถึง เคอร์เนลแบบเชิงเส้น แบบโพลิโนเมียล และ อาร์บีเอฟ เท่านั้น

2.4.1.2 ระบบเคเคที (KKT) ที่เป็นแก่นของปัญหา

การแก้ปัญหาเชิงเส้นของระบบเคเคที (KKT) เป็นประเด็นพื้นฐานที่มีการวางข้อจำกัดในการแก้ปัญหาแบบที่ไม่เป็นเชิงเส้นทั่ว ๆ ไป ในทุก ๆ จุดที่ซ้ำกันของระบบเคเคที (KKT) ที่คล้ายคลึงกับสมการที่ (2.33) ได้ถูกทำการแก้ปัญหา

2.4.1.3 การขาดแคลนสปลาสเนส (Sparseness) และการแปลความของซัพพอร์ตเวกเตอร์

อุปสรรคของรูปแบบสมการง่าย ๆ คือการขาดแคลนสปลาสเนส และนี่คือเงื่อนไขที่จะทำให้ค่าที่ดีที่สุดที่ $\alpha_k = \gamma e_k$ ในกรณีที่ตัวแบ่งประเภทข้อมูลด้วย หลักการลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีน ทุก ๆ จุดของข้อมูลจะเป็นซัพพอร์ตเวกเตอร์ ซึ่งปกติแล้วค่าของ α_k จะเท่ากับศูนย์ ในรูปที่ 2.28 ได้แสดงถึงการเปรียบเทียบระหว่างการเรียงชุดค่าของ $|\alpha_k|$ ในกรณีของ Vapnik หลักการซัพพอร์ตเวกเตอร์แมชชีน กับกรณีของการแบ่งข้อมูลด้วย หลักการลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีน หลังจากที่ทำให้ค่าเป็นบวกและทำการเรียงชุดค่าจากมากไปหาน้อย ในกรณีของ ลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีนจะทำให้ค่าสเปกตรัมลดลงโดยไม่มีการตัดสินใจระหว่างสีดำกับขาวระหว่างค่า α_k ที่เท่ากับศูนย์หรือไม่เท่ากับศูนย์ ดังนั้นอาจจะพูดได้ว่าในกรณีของ ลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีน ข้อมูลในส่วนของการฝึกฝนนั้นสามารถที่จะสร้างแบบจำลองที่ประกอบไปด้วยข้อมูลที่มีความสำคัญมากกว่าข้อมูลอื่น ๆ จุดข้อมูลที่มีค่ามาก $|\alpha_k|$ จะอยู่ใกล้และไกลกับขอบเขตของการตัดสินใจ

2.4.1.4 ความเป็นรูปแบบของตัวแปร และไม่เป็นตัวแปร (Parametric / Non - parametric)

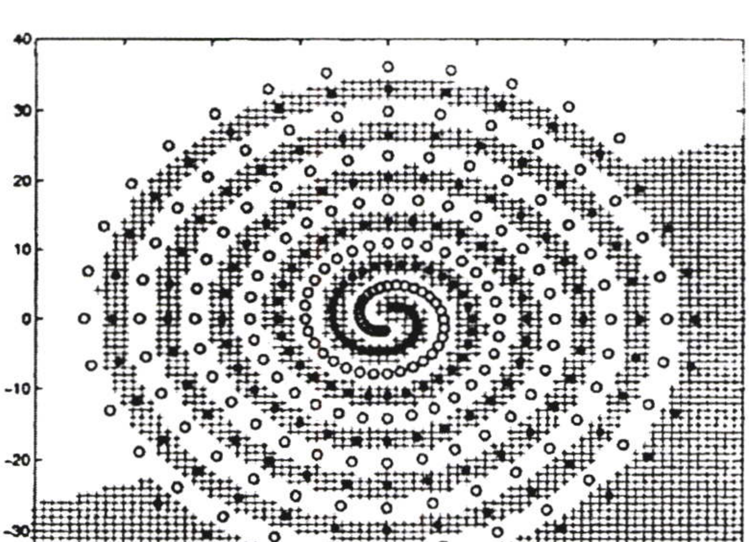
ตัวแบ่งประเภทข้อมูลด้วยหลักการลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีนจะมีโครงประสาทเครือข่ายเริ่มต้นเหมือนกันกับ หลักการซัพพอร์ตเวกเตอร์แมชชีน สำหรับค่าน้ำหนักในปัญหาหลักคือตัวแปรที่มีเวกเตอร์ขนาดคงที่ $w = \Re^{n_h}$ โดยที่ n_h เป็นจำนวนของหน่วยซ่อนใน

โครงข่ายนี้ ในปัญหารองนั้นจะมีปัญหาของการที่ค่าของตัวแปรจะขึ้นอยู่กับขนาดของเวกเตอร์ที่ถูกนำมาแก้ปัญหา $\alpha \in \Re^{n_k}$ แล้วจะเพิ่มมากขึ้นด้วยจำนวนของข้อมูลที่นำมาฝึกฝน N ซึ่งขนาดของระบบเคเคที (KKT) นั้นจะไม่มีผลกระทบต่อมิติของค่าอินพุท n แต่จะขึ้นอยู่กับค่า N เท่านั้น

2.4.1.5 การปรับค่าของตัวแปร (Tuning parameter)

ถ้านำอาร์บีเอฟเคอร์เนล $K(x,x_k) = \exp(-\|x - x_k\|_2^2 / \sigma^2)$ และค่าเฉพาะของผลลัพธ์จาก α และ b ที่ได้มาจากการแก้ปัญหาเชิงเส้นของระบบเคเคที (KKT) มาใช้ ซึ่งจะแตกต่างจากการนำค่าของตัวแปร (γ,σ) มาใช้ โดยที่มีหลายวิธีที่จะมาใช้ในการหาค่าตัวแปรเหล่านี้ ส่วนวิธีที่ง่าย คือการใช้ชุดของการฝึกฝนข้อมูล ชุดของการหาค่าที่ถูกต้องและชุดที่นำมาใช้ในการทดสอบ วิธีการค้นหาค่า (γ,σ) ที่มีความสำคัญมากในการได้แบบจำลองที่ดีที่สุด

ในรูปที่ 2.29 เป็นตัวอย่างของการแบ่งประเภทข้อมูลด้วย หลักการลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีน พร้อมกับ อาร์บีเอฟเคอร์เนล ในกรณีที่มีข้อมูลอยู่เพียงสองชนิด ซึ่งปัญหานี้มีความยาก ซึ่งต้องแก้ปัญหาด้วยวิธีมัลติเลเยอร์ เพอร์เซพตรอน เพราะว่าความเป็นจริงของการตัดสินใจค่าขอบเขตของข้อมูลแบบที่ไม่เป็นเชิงเส้นนั้นต้องถูกนำมาใช้ ซึ่ง หลักการลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีนนั้น จะมีค่าความผิดพลาดค่าศูนย์อยู่ในชุดของการฝึกฝน และทำให้สามารถตัดสินใจค่าขอบเขตได้เป็นอย่างดี ซึ่งปัญหานี้ยากในความรู้สึกที่ข้อมูลที่ไม่เป็นแบบเชิงเส้นมีความซับซ้อน แต่ในทางกลับกันจะรู้สึกว่าปัญหานี้ง่ายต่อการแบ่งประเภทของข้อมูล ในรูปที่ 2.30 ได้แสดงถึงค่าขอบเขตที่ตัดสินใจใช้และค่าที่สนับสนุนกัน



รูปที่ 2.29 การแบ่งประเภทข้อมูลรูปแบบวงกลมสองประเภทที่มีประสิทธิภาพมาก โดยหลัก การของหลักการลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีน กับ เคอร์เนล แบบ อาร์บีเอฟ [12]

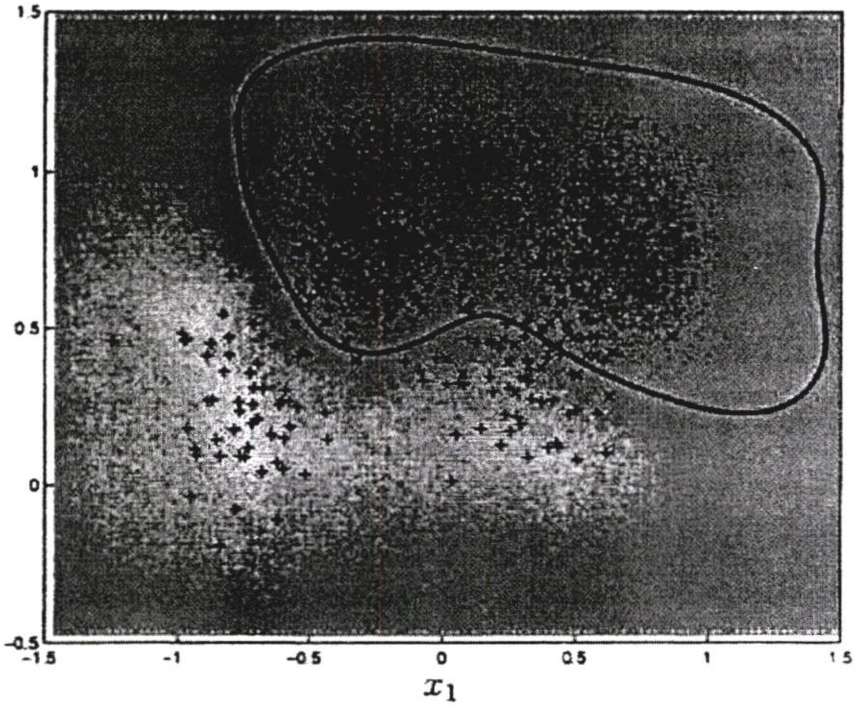
2.4.2 การแบ่งประเภทข้อมูลออกเป็นหลาย ๆ ประเภท (Multi-class Classification)

ในการแยกประเภทข้อมูลสองประเภทด้วยหลักการลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีน นั้น สามารถที่จะขยายไปทำการแก้ปัญหาการแบ่งประเภทข้อมูล ออกเป็นหลาย ๆ ประเภทได้ [15-18] โดยที่ค่าเอาท์พุท y หนึ่งค่า จะสามารถแบ่งออกเป็นค่าเอาท์พุท $y^{(i)}$ หลาย ๆ ค่าได้ โดยที่ $i = 1,...,n_y$ และ n_y เป็นจำนวนของค่าเอาท์พุทในส่วนของการเขียนโปรแกรมเพื่อเลือกค่าเอาท์พุทนั้นเป็นส่วนที่สำคัญ ซึ่งโดยหลักแล้วมีความเป็นไปได้ที่จะทำการแปลงข้อมูลเป็นรหัส (Encode) 2^{n_y} ประเภทด้วยกันโดยหมายความถึงจำนวนของ n_y เอาท์พุท แต่ปกติแล้ววิธีนี้ไม่ได้เป็นวิธีที่ดีที่สุด ส่วนวิธีที่ดีที่สุดคือการนำเอาท์พุทหลาย ๆ ค่ามาเป็นจำนวนของประเภทของข้อมูล ซึ่งจะกล่าวต่อไปภายหลัง โดยในที่นี้ จะมาเน้นในเรื่องของการนำวิธีการแบ่งประเภทข้อมูลหลักการลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีน มาขยายเพื่อทำการแบ่งประเภทข้อมูลออกเป็นหลาย ๆ ประเภทดังนี้

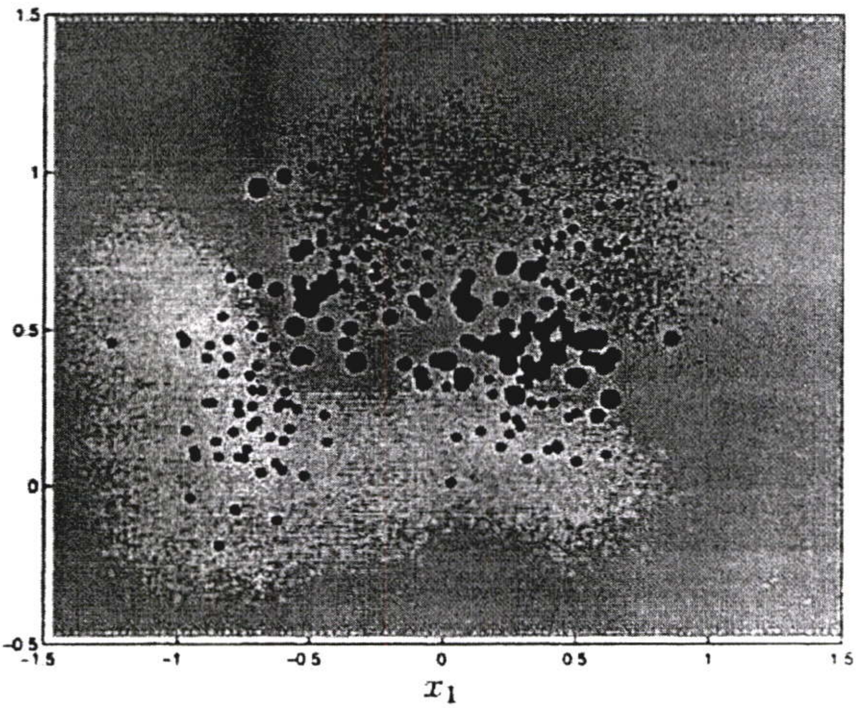
ค่าน้ำหนักแรกของระบบการแบ่งประเภทข้อมูลออกเป็นหลาย ๆ ประเภทนั้นขึ้นอยู่กับจำนวนของเอาท์พุทของตัวแบ่งประเภทข้อมูลดังนี้

$$\begin{cases} y^{(1)}(x) = sign[w^{(1)^T}\varphi^{(1)}(x) + b^{(1)}] \\ y^{(2)}(x) = sign[w^{(1)^T}\varphi^{(2)}(x) + b^{(2)}] \\ \quad\quad . \\ \quad\quad . \\ \quad\quad . \\ y^{(n_y)}(x) = sign[w^{(n_y)^T}\varphi^{(n_y)}(x) + b^{(n_y)}] \end{cases} \tag{2.39}$$

ด้วยการจับคู่กับฟีทเจอร์ในมิติที่สูงกว่า $\varphi^{(i)}(\bullet): \Re^n \longrightarrow \Re^{n_{h_i}}$ (สำหรับ $i = 1,2,...,n_y$) ด้วยมิติ $n_{h_1}, n_{h_2},..., n_{h_{n_y}}$ ซึ่งจะรับผิดชอบกับเวกเตอร์ $w^{(i)} \in \Re^{n_{h_i}}$ กับค่าไบแอซ $b^{(i)} \in \Re$



(ก)



(ข)

รูปที่ 2.30 การแบ่งประเภทข้อมูลไบนารี (Binary) โดยหลักการของลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีนกับ เคอร์เนล แบบ อาร์บีเอฟ [14]

สำหรับการแบ่งประเภทข้อมูลออกเป็นหลาย ๆประเภทจะใช้สมการที่ (2.40) ดังนี้

$$\left[\begin{array}{l} \min\limits_{w^{(i)},b^{(i)},e_k^{(i)}} J_p\left(w^{(i)},e_k^{(i)}\right) = \frac{1}{2}\sum_{i=1}^{n_y}\gamma_i \sum_{k=1}^{N}(e_k^{(i)})^2 \\ \text{such that} \\ y_k^{(1)}\left[w^{(1)^T}\varphi^{(1)}(x_k)+b^{(1)}\right] = 1-e_k^{(1)},\ \ k = 1,\ldots,N \\ y_k^{(2)}\left[w^{(2)^T}\varphi^{(2)}(x_k)+b^{(2)}\right] = 1-e_k^{(2)},\ \ k = 1,\ldots,N \\ . \\ . \\ . \\ y_k^{(n_y)}\left[w^{(n_y)^T}\varphi^{(n_y)}(x_k)+b^{(n_y)}\right] = 1-e_k^{(n_y)},\ \ k = 1,\ldots,N \end{array}\right] \tag{2.40}$$

เมื่อ สมการของลากรองจ์ สำหรับปัญหานี้คือ สมการที่ (2.41)

$$L\left(w^{(i)},b^{(i)},e_k^{(i)};\alpha_k^{(i)}\right) = J_P(w^{(i)},e_k^{(i)}) - \sum_{i=1}^{n_y}\sum_{k=1}^{N}\alpha_k^{(i)}(y_k^{(i)}[w^{(i)^T}\varphi^{(i)}(x_k)+b^{(i)}]-1+e_k^{(i)}) \tag{2.41}$$

ด้วยเงื่อนไขดังนี้

$$\begin{cases} \dfrac{\partial L}{\partial w^{(i)}} = 0 \rightarrow w^{(i)} = \sum\limits_{k=1}^{N}\alpha_k^{(i)} y_k^{(i)}\varphi^{(i)}(x_k) \\ \dfrac{\partial L}{\partial b^{(i)}} = 0 \rightarrow \sum\limits_{k=1}^{N}\alpha_k^{(i)} y_k^{(i)} = 0 \\ \dfrac{\partial L}{\partial e_k^{(i)}} = 0 \rightarrow \alpha_k^{(i)} = \gamma e_k^{(i)} \\ \dfrac{\partial L}{\partial \alpha_k^{(i)}} = 0 \rightarrow y_k^{(i)}[w^{(i)^T}\varphi^{(i)}(x_k)+b^{(i)}] = 1-e_k^{(i)} \end{cases}$$

สำหรับ $k=1,\ldots,N$ และ $i=1,\ldots,n_y$ หลังจากที่ได้กำจัดตัวแปร $w^{(i)}$ และ $e_k^{(i)}$ จะได้สมการของระบบเคเคที ที่ใช้แก้ปัญหาใน $\alpha_k^{(i)},b^{(i)}$ ดังสมการที่ (2.42)

$$\left[\begin{array}{c|c} 0 & Y_M^T \\ \hline Y_M & \Omega_M + D_M \end{array}\right]\left[\begin{array}{c} b_M \\ \hline \alpha_M \end{array}\right] = \left[\begin{array}{c} 0 \\ \hline 1_v \end{array}\right] \tag{2.42}$$

โดยที่

$$Y_M = blockdiag\{\begin{bmatrix} y_1^{(1)} \\ . \\ . \\ . \\ y_N^{(1)} \end{bmatrix},\ldots,\begin{bmatrix} y_1^{(n_y)} \\ . \\ . \\ . \\ y_N^{(n_y)} \end{bmatrix}\}$$

$$\Omega_M = blockdiag\{\Omega^{(1)},\ldots,\Omega^{(n_y)}\}, \quad \Omega_{kl}^{(i)} = y_k^{(i)} y_l^{(i)} K^{(i)}(x_k, x_{il}) \tag{2.43}$$

$$D_M = blockdiag\{D^{(1)},\ldots,D^{(n_y)}\}, \quad D_{kl}^{(i)} = \delta_{kl} / \gamma_i$$

สำหรับค่าของตัวแปรเหล่านี้ ค่า k , $l=1,\ldots,N$, $i=1,\ldots,n_y$, $b_M=[b^{(1)};\ldots;b^{(n_y)}]$, $\alpha_M=[\alpha_1^{(1)};\ldots;\alpha_N^{(1)};\ldots;\alpha_1^{(n_y)};\ldots;\alpha_N^{(n_y)}]$ และ δ_{kl} เรียกว่าเป็นโครเน็คเกอร์ เดลต้า (Kronecker delta) ($\delta_{kl}=1$ ถ้า $k=1$ และ 0 อย่างใดอย่างหนึ่ง) สมการเคอร์เนล ที่นำมาใช้เป็นดังสมการที่ (2.44)

$$\begin{aligned} K^{(i)}(x_k,x_l) &= \varphi^{(i)}(x_k)^T \varphi^{(i)}(x_l) \\ &= \exp(-\|x-x_k\|_2^2 / \sigma^2) \end{aligned} \tag{2.44}$$

สำหรับแต่ละตัวแบ่งประเภทย่อยจะเป็นสิ่งที่สำคัญ ในการทำให้ได้ค่าแตกต่างกันไป สำหรับ γ_i,σ_i ในกรณีของอาร์บีเอฟ เคอร์เนล โดยที่ผลของการแบ่งประเภทข้อมูลในสองมิติเป็นดังสมการที่ (2.45)

$$y^{(i)}(x) = sign[\sum_{k=1}^{N} \alpha_k^{(i)} y_k^i K^{(i)}(x,x_k) + b^{(i)}] \tag{2.45}$$

สำหรับค่า $i=1,\ldots,n_y$

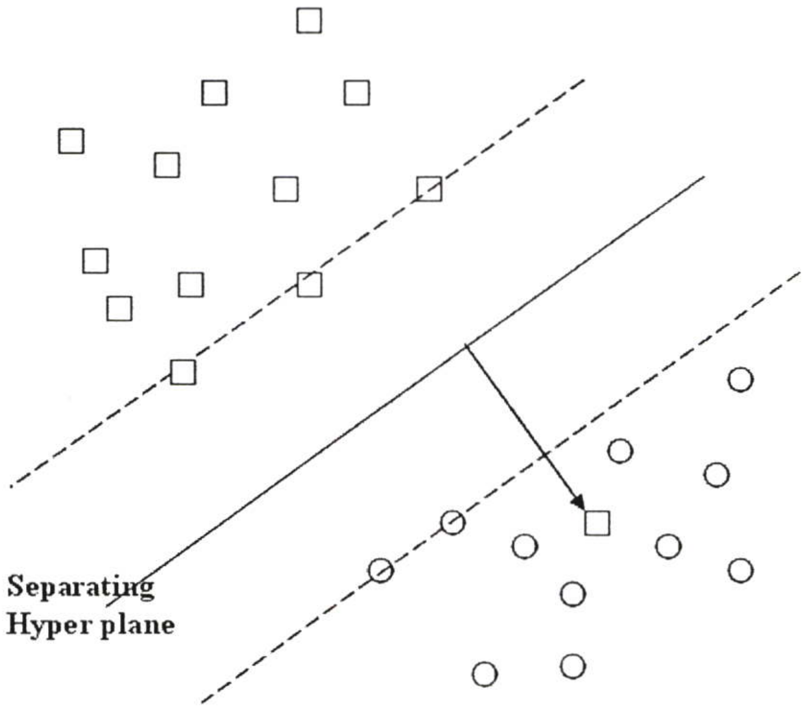
จากสมการที่ (2.42) จะเห็นได้ว่า ไม่ต้องการที่จะทำการแก้สมการที่ (2.42) ทั้งหมด ซึ่งสามารถที่จะแบ่งออกเป็นหลาย ๆ สมการย่อย ๆ n_y ซึ่งในความเป็นจริงไม่เป็นที่น่าแปลกใจเพราะว่านี่เป็นที่รู้จักในปัญหาของข้อมูลหลาย ๆ ประเภทสามารถที่จะถูกแก้ปัญหาเป็นชุดของสองประเภทข้อมูล โดยปกติแล้วสิ่งที่สำคัญคือการเขียนโปรแกรมแก้ปัญหาหาค่าที่ดีที่สุด

โดยทั่วไปปัญหาของการจัดข้อมูลในหลาย ๆ ประเภทนั้นจะถูกแก้ไขโดยการสร้างสมการใหม่ด้วย n_c ประเภทไปยังกลุ่มของปัญหา n_y โดยที่แต่ละประเภทของ C_i นั้น จะเป็น $[y_i^{(1)}; y_i^{(2)}; ...; y_i^{(n_y)}] \in \{-1,+1\}^{n_y}$ โดยที่ค่า $i = 1,...,n_c$ ได้ถูกกำหนดขึ้น ในปัจจุบันมีอยู่หลายวิธีที่จะสร้างชุดของตัวแบ่งข้อมูลสองประเภท ถ้าใช้ค่า n_y เป็นเอาท์พุทที่มีค่าเท่ากับจำนวนของประเภท ($n_c = n_y$) และกำหนดให้การตัดสินใจในการแบ่งข้อมูลระหว่างค่าในแต่ละประเภทเท่ากันกับค่าในประเภทอื่น ๆ ทั้งหมด จะทำให้การที่จะหาค่าความผิดพลาดจากเอาท์พุท (ECOC) นั้น ได้ถูกกระตุ้นด้วยทฤษฎีของข้อมูล เพื่อที่จะสามารถลดความซ้ำซ้อน (Redundancy) ได้ โดยใช้ตัวแบ่งประเภทข้อมูลฐานสองมากกว่าจำนวนของประเภทข้อมูล ($n_y > n_c$) ในเอาท์พุท สำหรับใน 1 VS 1 โปรแกรมเอาท์พุท สามารถที่จะใช้ $(n_c - 1)/2$ ที่มากับตัวแบ่งประเภทข้อมูลฐานสองโดยที่แต่ละตัว จะสามารถทำให้เห็นความแตกต่างกันได้ระหว่างข้อมูลสองประเภทที่แตกต่างกันโดยที่โปรแกรมเอาท์พุทนั้นเป็นค่าความยาว n_y ที่ติดมากับ $\{-1,0,+1\}^{n_y}$ ซึ่งจะใช้ค่า 0 สำหรับข้อมูลประเภทที่ไม่นำมาพิจารณา ยกตัวอย่างเช่น ข้อมูลสามประเภทสามารถที่จะถูกแสดงโดยการ $[-1,+1,0],[-1,0,+1],[0,-1,+1]$ ในขณะที่อีกทางหนึ่งใช้เพียงแค่สองเอาท์พุทแล้วไปทำการเขียนแปลงข้อมูลเป็นสามประเภทด้วยกัน และจะใช้เอาท์พุททั้งสามนี้ในโปรแกรมของ 1 ต่อ 1 ซึ่งจะทำให้ได้ผลลัพธ์คือขอบเขตของข้อมูลนั่นเอง ในส่วนวิธีการถอดรหัส (Decode) นั้น จะทำการนำค่าคลาส ลาเบลหรือค่าเป้าหมาย ซึ่งปกติจะนำมาใช้กำหนดให้ตามค่าระยะทางแฮมมิ่ง (Hamming) ที่น้อยที่สุดจากเอาท์พุท [19] แต่ทางเลือกอื่นก็มีความเป็นไปได้เช่นกัน ซึ่งจะพิจารณาจากโปรแกรมของ 1 VS 1 มาเป็นตัวให้คะแนนระหว่างแต่ละคู่ของประเภทข้อมูล โดยที่ส่วนไหนมีระยะทางแฮมมิ่งไปยังแต่ละประเภทข้อมูลน้อยที่สุด จะได้รับการให้คะแนนมากที่สุด

2.5 คลอส วาริเดชัน และ กริดเสิร์จ

จากหลักการของ หลักการซัพพอร์ตเวกเตอร์แมชชีน นั้นจะเห็นได้ว่าหลักการนี้สามารถสร้าง ระนาบได้หลาย ๆ อัน มาเป็นสิ่งที่นำมาใช้ในการแบ่งประเภทของข้อมูลออกเป็นสองกลุ่มได้อย่างสมบูรณ์ อย่างไรก็ตาม การแบ่งประเภทให้ได้ดีแบบไม่มีข้อผิดพลาดเลยนั้นอาจจะไม่สามารถเป็นไปได้ และด้วยความเป็นจริงที่ข้อมูลต่าง ๆ ไม่ได้มีอยู่เพียงสองประเภท แต่มีอยู่หลายประเภทหลายมิติด้วยกัน ทำให้หลักการของ หลักการซัพพอร์ตเวกเตอร์แมชชีน ไม่สามารถสร้างแบบจำลองออกมาได้ดีพอ ซึ่งจะรู้จักกันว่าปัญหาของข้อมูลเหล่านั้นเรียกกันว่า ข้อมูลที่มีความลงตัวมากจนเกินไป (OverFit) [21] โดยที่รูปที่ 2.31 นี้ได้แสดงถึงการที่ไม่สามารถแบ่งข้อมูลส่วนของ

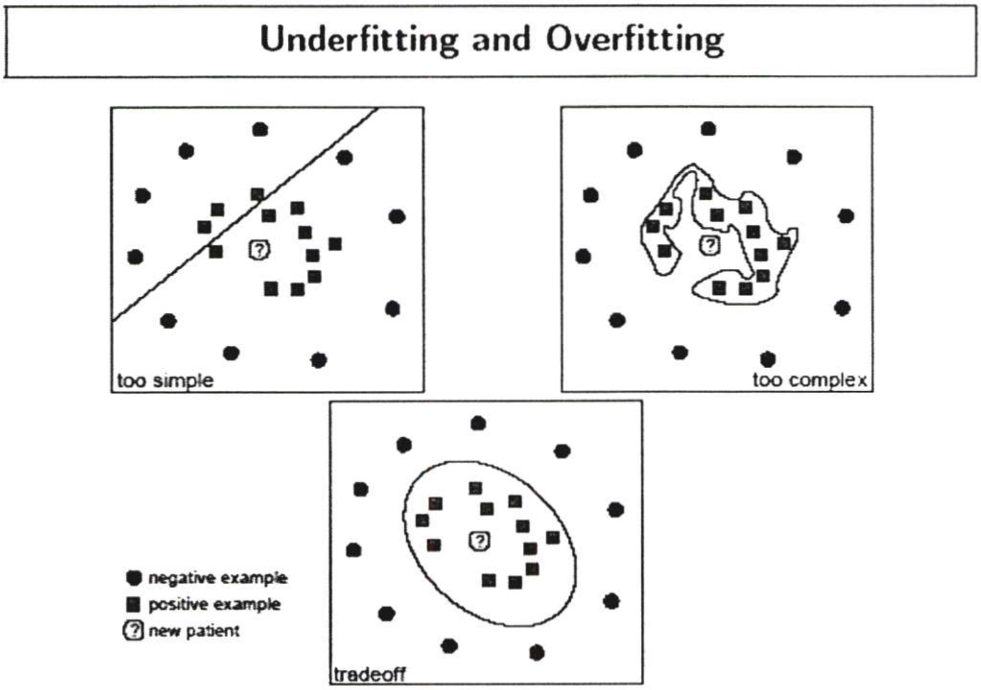
การฝึกฝนได้ ซึ่งได้มาจาก Florian Markowetz [14] และรูปที่ 2.32 ที่ได้แสดงถึงปัญหาความผิดพลาดของข้อมูลที่มีความลงตัวมากจนเกินไปและน้อยจนเกินไป



รูปที่ 2.31 การแบ่งข้อมูลในชุดของการฝึกฝนที่ไม่มีประสิทธิภาพเนื่องจากปัญหา ข้อมูลที่มีความลงตัวมากจนเกินไป (OverFit)

ค่าความถูกต้องของแบบจำลอง หลักการซัพพอร์ตเวกเตอร์แมชชีน นั้นจะขึ้นอยู่กับการเลือกของตัวแปรที่นำมาใช้กับ เคอร์เนล ยกตัวอย่างเช่นค่า C ค่า แกมมา (γ) เป็นต้นโดยวิธีที่นิยมนำมาใช้ในการหาค่าตัวแปรเหล่านี้คือ กริดเสิร์จ และ แพทเทิร์นเสิร์จ (Pattern search) โดยที่ กริดเสิร์จ จะหาค่าของแต่ละตัวแปรตัดกับค่าขอบเขตในการค้นหาโดยใช้ลำดับของรูปทรงเรขาคณิต ส่วน แพทเทิร์นเสิร์จ นั้นจะเริ่มต้นจากจุดศูนย์กลางของค่าขอบเขตในการค้นหา และสร้างลำดับการทดลองในแต่ละทิศทางสำหรับแต่ละตัวแปร ถ้าความพอดีของแบบจำลองมีการพัฒนาขึ้น การค้นหาจากจุดศูนย์กลางจะย้ายไปยังจุดใหม่และกระบวนการก็จะซ้ำไปเรื่อย ๆ ถ้าความพอดีของแบบจำลองไม่มีการพัฒนาขึ้นเลย ขนาดของลำดับการทดลองจะลดลงเรื่อย ๆ จนไม่สามารถทำการค้นหาได้

ส่วนการที่จะหลีกเลี่ยงปัญหาของข้อมูลที่มีความลงตัวมากจนเกินไป หรือเรียกว่าเกินพอดี (OverFit) นั้น วิธีการของคลอส วาริเดชัน ได้ถูกนำมาใช้ในการประเมินค่าความพอดี (Fitting) ของค่าในแต่ละตัวแปรในการทดลองระหว่างกระบวนการการค้นหาแบบ กริดเสิร์จ และ แพทเทิร์นเสิร์จนั่นเอง



รูปที่ 2.32 ข้อมูลที่มีความลงตัวน้อยจนเกินไป (Underfitting) และ ข้อมูลที่มีความลงตัวมากจนเกินไป (Overfitting) [22]

คลอส วาริเดชัน เป็นแบบจำลองที่นำมาใช้ในการหาค่าได้ดีกว่าวิธีอื่น ๆ ปัญหาของวิธีการหาค่าอื่น ๆเป็นเพราะวิธีเหล่านั้น ไม่ได้มีตัวให้บ่งชี้ว่าวิธีนั้นสามารถทำนายค่าข้อมูลใหม่ ๆ ที่ไม่เคยมีการเรียนรู้มาก่อนได้ วิธีหนึ่งที่สามารถแก้ปัญหานี้ได้คือการที่ไม่ใช้ข้อมูลทั้งหมดมาเป็นส่วนของชุดการเทรนนิ่ง โดยจะนำข้อมูลจำนวนหนึ่งออกจากส่วนการเทรนนิ่งข้อมูลชุดแรก จากนั้นเมื่อการเทรนนิ่งเสร็จสมบูรณ์ ข้อมูลจำนวนนั้นที่นำออกมาจะสามารถถูกนำมาใช้ในการทดลองหาประสิทธิภาพของแบบจำลองที่ได้มาจากการเทรนนิ่งด้วยข้อมูลชุดใหม่นั่นเอง และสิ่งนี้คือความคิดพื้นฐานสำหรับประเภทของแบบจำลองวิธีการหาค่าที่เรียกว่า คลอส วาริเดชัน นั่นเอง

ซึ่งหมายความว่าข้อมูลชุดแรก แบบจำลอง M_α สามารถที่จะทำนายค่า y_i^* ถ้าเวกเตอร์ y^* ที่ได้รับแตกต่างจากที่ใช้สำหรับแบบจำลองในการทำนายค่าได้ และค่าของเวกเตอร์ของ x^* ที่นำมาใช้ได้แล้ว ดังนั้นค่าเฉลี่ยยกกำลังสองของการทำนายค่าความผิดพลาดนั้นเป็นดังสมการที่ (2.46) นี้

$$n^{-1}\sum_{i=1}^{n}(y_i^* - \hat{y}_i^*)^2 = n^{-1}\sum_{i=1}^{n}(x_i'\beta + \varepsilon_i - x_i'\mathrm{M}(\mathrm{M}'\mathrm{X}'\mathrm{X}\mathrm{M})^{-1}\mathrm{M}'\mathrm{X}'y)^2 \tag{2.46}$$

ให้ y เป็นเงื่อนไขในการทำนายค่าความผิดพลาดดังสมการที่ (2.47)

$$E\left(n^{-1}\sum_{i=1}^{n}(y_i^{\bullet}-\hat{y}_i^{\bullet})^2\big|y\right)=n^{-1}\sum_{i=1}^{n}\left(x_i'(\beta-\mathrm{M}b_\alpha)\right)^2+\sigma^2 \tag{2.47}$$

โดยที่ค่าความผิดพลาดนี้ จะขึ้นอยู่กับ σ^2 ความหลากหลายของค่าในอนาคต และ $(\beta-\mathrm{M}b_\alpha)$ เป็นต้น คลอส วาริเดชัน แต่ละแบบนั้นสามารถที่จะถูกกำหนดให้ไม่มีเงื่อนไขดังสมการที่ (2.48)

$$\begin{aligned}\Gamma_{\alpha n} &= \sigma^2+n^{-1}tr(\mathrm{P}_\alpha)\sigma^2+n^{-1}\beta'\mathrm{X}'(\mathrm{I}_n-\mathrm{P}_\alpha)\mathrm{X}\beta \\ &= \sigma^2+n^{-1}\alpha\sigma^2+n^{-1}\beta'\mathrm{X}'(\mathrm{I}_n-\mathrm{P}_\alpha)\mathrm{X}\beta\end{aligned} \tag{2.48}$$

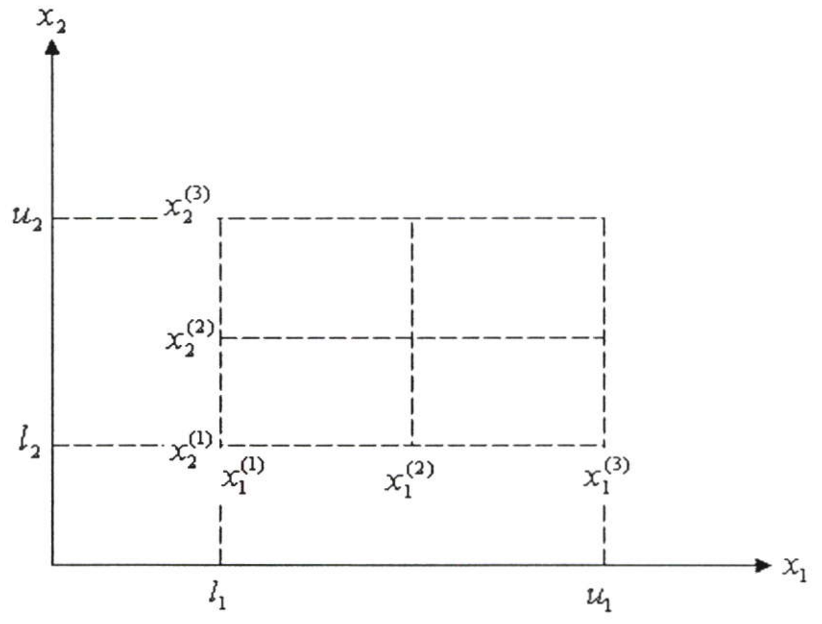
โดยที่ $\mathrm{P}_\alpha=\mathrm{XM}(\mathrm{M}'\mathrm{X}'\mathrm{XM})^{-1}\mathrm{M}'\mathrm{X}'$ เป็น Projection เมทริกซ์ของเมทริกซ์ย่อย XM สำหรับวิธี คลอส วาริเดชัน นั้นได้เลือกแบบจำลองโดยทำให้ค่า $\Gamma_{\alpha n}$ น้อยลงกว่าค่าตัวแปรอื่น นั่นก็คือทางเลือกทั้งหมดที่น่าจะเป็นไปได้ของ M_α นั่นเอง เมื่อชุดข้อมูลชุดหนึ่งสามารถถูกนำมาแบ่งออกเป็นสองส่วนแล้ว ข้อมูลชุดแรกขนาด n_c ได้ถูกนำมาใช้ฝึกฝนให้ได้แบบจำลอง ในขณะที่ข้อมูลอีกชุดหนึ่งขนาด n_v ได้ถูกนำมาใช้ทดสอบเพื่อคาดคะเนหาค่าที่ดีที่สุดได้ วิธีคลอสวาริเดชันนั้น เลือกแบบจำลองด้วยค่าเฉลี่ยที่ดีที่สุด ที่สามารถทำนายบนพื้นฐานของความแตกต่างของวิธีในการแบ่งประเภทข้อมูล การสร้างข้อมูลที่ทำให้ได้ค่าที่ดีที่สุดได้ถูกสร้างจากแบบแผนเมทริกซ์ S ซึ่งก็คือเมทริกซ์ M นั่นเองดังสมการที่ (2.49)

$$S=\left(e'_{S_1},\ldots,e'_{S_i},\ldots,e'_{S_{n_v}}\right) \tag{2.49}$$

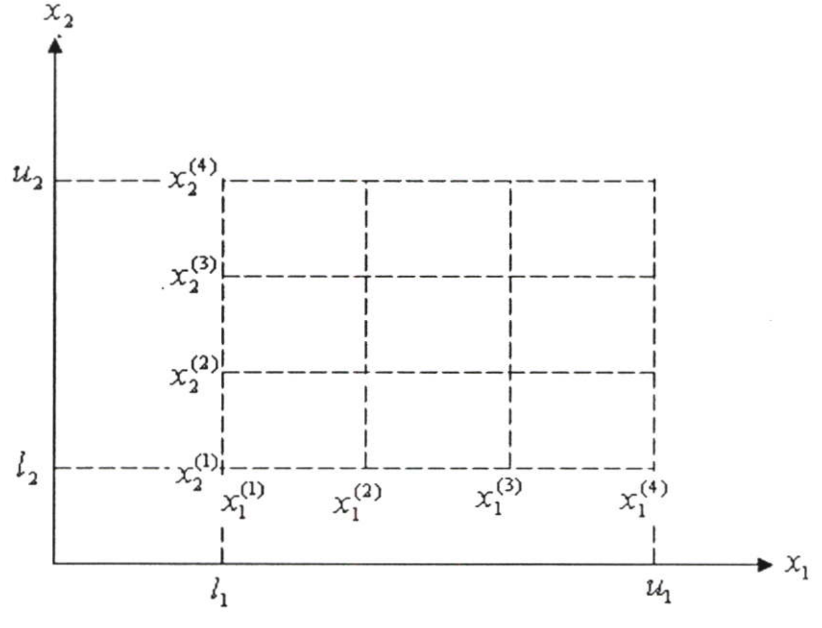
โดยที่ e_{S_1} เป็นลำดับคอลัมน์ S_i-th ของเมทริกซ์เอกลักษณ์ I_n

สำหรับวิธี กริดเสิร์จ นั้น ได้มีส่วนร่วมกับการตั้งค่ากริดที่เหมาะสม ในพื้นที่ที่ได้ถูกออกแบบไว้ ซึ่งวิธีนี้คือการหาค่าฟังก์ชันที่ต้องการในทุก ๆ จุดของกริดและการค้นหาจุดของกริด ที่มีส่วนเกี่ยวข้องกับการได้ค่าฟังก์ชันที่น้อยที่สุด ยกตัวอย่างเช่น ถ้าขอบเขตสูงสุดกับต่ำสุดบนตัวแปรในขอบเขตที่ถูกเลือกไว้เป็น l_i และ u_i ตามลำดับ โดยสามารถที่จะแบ่งขอบเขตของ (l_i,u_i) ไปยัง p_i-1 เท่า ๆ กัน ดังนั้นค่า $x_i^1,x_i^2,\ldots,x_i^{p_i}$ จะแสดงว่าจุดของกริดนั้นเป็นไปตามค่าของแกน x_i $(i=1,2,\ldots,n)$ ซึ่งจากนี้ได้นำไปสู่ผลรวมของจุดของกริด $p_1p_2\ldots p_n$ ในส่วนที่ได้ถูกออกแบบไว้จาก กริด ด้วยค่า $p_i=3$ และ 4 นั้นได้ถูกแสดงไว้ในพื้นที่สองมิติที่ได้ออกแบบไว้ดังรูปที่ 2.33 โดยที่จุดของกริดนั้นสามารถที่จะถูกเลือกเป็นจุดพื้นฐานของการดำเนินการทดลอง ซึ่งค่าของจุดนั้นสามารถที่จะถูกมองได้ว่า วิธีการกริดเสิร์จนั้นต้องการที่จะป้องการค่าตัวเลขของฟังก์ชันที่นำมาหาค่าที่มากจนเกินไปของปัญหาที่มักจะเกิดขึ้นอยู่เป็นประจำ ยกตัวอย่างเช่น ปัญหาที่มีการ

ออกแบบมีตัวแปร 10 ตัว $(n=10)$ จำนวนของจุดในกริด จะเป็น $3^{10}=59,049$ ด้วยค่า $p_i=3$ และ $4^{10}=1,408,576$ ด้วยค่า $p_i=4$ โดยที่ $(i=1,2,...,10)$ อย่างไรก็ตาม สำหรับปัญหาที่เกิดจากการที่มีค่าตัวแปรน้อย ๆ วิธี กริดเสิร์จ นั้นสามารถที่จะถูกนำมาใช้ในการหาให้ได้ค่าเริ่มต้นที่ดี



(ก)



(ข)

รูปที่ 2.33 กริดเสิร์จ ด้วยค่า (ก) $p_i=3$ และ (ข) $p_i=4$

2.6 วิธีการประเมินหาประสิทธิภาพ

ค่าเฉลี่ยของความคลาดเคลื่อนสัมบูรณ์ (Mean Absolute Error : MAE) เป็นวิธีการคำนวณหาค่าความคลาดเคลื่อนเฉลี่ย ระหว่างค่าจริงและค่าพยากรณ์เทียบกับค่าจริง และค่าเฉลี่ยของเปอร์เซ็นต์ความคลาดเคลื่อนสัมบูรณ์ (Mean Absolute Percent Error : MAPE) เป็นวิธีการคำนวณหาค่าร้อยละของความคลาดเคลื่อนเฉลี่ย ระหว่างค่าจริงและค่าพยากรณ์เทียบกับค่าจริงเช่นกัน ซึ่งทั้งสองวิธีมีสูตรในการคำนวณดังนี้

ค่าเฉลี่ยของความคลาดเคลื่อน

$$MAE = \frac{1}{T}\sum_{t=1}^{T}\left| y_t - \hat{y}_t \right| \tag{2.47}$$

ค่าเฉลี่ยของเปอร์เซ็นต์ความคลาดเคลื่อนสัมบูรณ์

$$MAPE = \frac{1}{T}\sum_{t=1}^{T}\left(\left| \frac{y_t - \hat{y}_t}{y_t} \right| \times 100 \right) \tag{2.48}$$

เมื่อ y_t คือ ค่าที่พยากรณ์ขึ้น ณ เวลา t และ $\hat{y}_t$ คือ ค่าข้อมูลจริง ณ เวลา t

จะเห็นได้ว่า ค่าของความแตกต่าง $y_t - \hat{y}_t$ ถูกทำให้เป็นค่าสัมบูรณ์โดยไม่คิดเครื่องหมายเพื่อให้สามารถวัดความคลาดเคลื่อนของการคาดการณ์แต่ละค่าได้ ความคลาดเคลื่อนดังกล่าวถูกทำให้เป็นร้อยละ เพื่อให้สามารถเปรียบเทียบได้ถูกต้องยิ่งขึ้น เนื่องจากไม่มีหน่วยทำให้ไม่ได้รับผลกระทบจากขนาด ของตัวเลข MAPE จึงเหมาะสมที่จะใช้ในการพิจารณาความคลาดเคลื่อนของการคาดการณ์แต่ละค่า และ การเปรียบเทียบความแม่นยำของการคาดการณ์หลาย ๆ ชุด ซึ่งอาจจะใช้ช่วงเวลาของการคาดการณ์ที่ต่างกัน

บทต่อไปเป็นการนำแนวคิดและหลักการทั้งหมดที่ได้ในบทนี้ ไปประยุกต์ใช้ในการสร้างแบบจำลองที่มีประสิทธิภาพ เพื่อนำมาใช้ในการแบ่งประเภทของข้อมูลที่ได้มาจากสถาบันการเงินในประเทศไทย

บทที่ 3

วิธีทดลองการจัดให้คะแนนความน่าเชื่อถือของลูกค้าด้วย ซัพพอร์ตเวกเตอร์แมชชีน และลีซท์ สแควซัพพอร์ตเวกเตอร์แมชชีน

ในบทนี้จะกล่าวถึงขั้นตอนต่าง ๆ ในการพยากรณ์ความน่าเชื่อถือของลูกค้าใหม่ จากสถาบันการเงินไทยด้วยหลักการของ ซัพพอร์ตเวกเตอร์แมชชีน และ ลีซท์ สแควซัพพอร์ตเวกเตอร์แมชชีน โดยได้นำข้อมูลจากสถาบันการเงินในประเทศไทยจริงโดยได้ทำการขอเงื่อนไขการอนุมัติเงินให้กับลูกค้าจากสถาบันการเงินแห่งหนึ่งพร้อมทั้งตัวอย่างข้อมูลจริงของลูกค้าจำนวนหนึ่งโดยมีการปกปิดชื่อของลูกค้า จากนั้นนักวิจัยได้ทำการสร้างข้อมูลของลูกค้าให้มีปริมาณมากขึ้นเพื่อที่จะได้ทำให้การทดลองการสร้างแบบจำลองนี้มีประสิทธิภาพมากขึ้น โดยที่ข้อมูลลูกค้าที่จำลองขึ้นเองนั้นได้ทำการเปรียบเทียบกับตัวอย่างข้อมูลจำนวนหนึ่งที่ได้มาจริงกับสถาบันการเงินแห่งนั้น โดยได้รับการตรวจสอบจากสถาบันว่าข้อมูลที่จำลองขึ้นมา มีความสมเหตุสมผล และสามารถนำมาใช้งานได้จริง

โดยขั้นตอนต่าง ๆ ในการทดลองเป็นดังต่อไปนี้

3.1 การเตรียมข้อมูล

ทำการนำตัวอย่างข้อมูลลูกค้าจำนวน 1000 คน มาแบ่งเป็นชุดของการฝึกฝน (Training) และชุดของการทดสอบ (Testing) ออกเป็นสามอัตราส่วนด้วยกัน โดยกำหนดจำนวนข้อมูลในการทดสอบเป็นค่าคงที่ และกำหนดให้ข้อมูลในส่วนของการฝึกฝนมีอัตราส่วน ดังนี้

1. ชุดของการฝึกฝน 600 คน และชุดของการทดสอบ 200 คน
2. ชุดของการฝึกฝน 700 คน และชุดของการทดสอบ 200 คน
3. ชุดของการฝึกฝน 800 คน และชุดของการทดสอบ 200 คน

โดยเป้าหมายนั้น จะนำอัตราส่วนข้างต้นมาทำการเปรียบเทียบหาค่าที่ดีที่สุด ในการแบ่งประเภทลูกค้าออกเป็นสองกลุ่มด้วยกันคือ

กลุ่มที่ 1 (C1) – ลูกค้าที่มีความน่าเชื่อถือ (Credit) ดีมาก

กลุ่มที่ 2 (C2) – ลูกค้าที่ความน่าเชื่อถือค่อนข้างดี

กลุ่มที่ 3 (C3) – ลูกค้าที่ความน่าเชื่อถือมีความเสี่ยง

กลุ่มที่ 4 (C4) – ลูกค้าที่ความน่าเชื่อถือไม่ดี

3.1.1 ตัวอย่างข้อมูลดิบ

A11 60 A33 A43 20000 A65 A75 4 A93 A101 4 A124 27 A143 A153 2 A173 1 A192 A201 C3

A14 48 A32 A41 90000 A62 A73 2 A92 A101 2 A121 29 A143 A152 1 A173 1 A192 A202 C2

A12 72 A34 A45 80000 A65 A72 9 A94 A101 9 A124 47 A142 A151 0 A173 2 A191 A201 C4

A15 60 A30 A49 50000 A61 A72 0 A95 A103 2 A122 26 A143 A153 3 A174 0 A192 A202 C1

คำอธิบายและความหมายของแต่ละค่าคุณสมบัติ (Attribute) ของข้อมูลลูกค้าจากสถาบันการเงินไทยที่นำมาใช้ในการทดลองโดยที่ค่าคุณสมบัติจะมีสองประเภทด้วยกันคือ แบบที่เป็นประเภทของข้อมูลเชิงคุณภาพ (Qualitative) ยกตัวอย่างเช่นคุณสมบัติที่ 1 และแบบที่เป็นประเภทของข้อมูลเชิงปริมาณ (Quantitative) ยกตัวอย่างเช่นคุณสมบัติที่ 2

คุณสมบัติที่ 1 (เชิงคุณภาพ) ระดับการศึกษา

A11 : ประถมศึกษา

A12 : มัธยมศึกษา

A13 : อนุปริญญา

A14 : ปริญญาตรี

A15 : ปริญญาโท/ปริญญาเอก

คุณสมบัติที่ 2 (เชิงปริมาณ) เดือนที่ต้องการกู้เงิน

คุณสมบัติที่ 3 (เชิงคุณภาพ) ประวัติการชำระเงิน

A30 : ไม่เคยใช้เครดิต/จ่ายเงินครบตามกำหนดทุกงวดเงินให้กับทุกธนาคาร

A31 : จ่ายเงินครบตามกำหนดทุกงวดเงินให้กับธนาคารนี้

A32 : ยังมีรายการที่ต้องจ่ายเงินให้กับธนาคารในปัจจุบันและจ่ายตรงตามกำหนด

A33 : มีประวัติการจ่ายเงินล่าช้าบ้างในอดีต

A34 : มีประวัติการจ่ายเงินไม่ดีทั้งอดีตและปัจจุบัน

คุณสมบัติที่ 4: (เชิงคุณภาพ) จุดประสงค์ในการกู้ยืม

A40 : ซื้อรถ (ใหม่)

A41 : ซื้อรถ (ใช้แล้ว)

A42 : เฟอร์นิเจอร์/อุปกรณ์ต่าง ๆ

A43 : วิทยุ/โทรทัศน์/เครื่องเกี่ยวกับความบันเทิง

A44 : เครื่องใช้ในบ้าน

A45 : การซ่อมแซมบ้าน

A46 : การศึกษา

A47 : ซื้อบ้าน(ใหม่)

A48 : ซื้อบ้าน(มือสอง)

A49 : ธุรกิจ

A410 : อื่น ๆ

คุณสมบัติที่ 5 (เชิงปริมาณ) ปริมาณเงินที่ต้องการ

คุณสมบัติที่ 6 (เชิงคุณภาพ) บัญชีออมทรัพย์/ พันธบัตร

A61 : ... < 10,00 บาท

A62 : 10,000 <= ... < 50,000 บาท

A63 : 50,000 <= ... < 100,000 บาท

A64 : ... >= 100,000 บาท

A65 : ไม่มีเงินเก็บ

คุณสมบัติที่ 7 (เชิงคุณภาพ) ระยะเวลาการทำงาน

A71 : ว่างงาน

A72 : ... < 1 ปี

A73 : 1 <= ... < 4 ปี

A74 : 4 <= ... < 7 ปี

A75 : .. >= 7 ปี

คุณสมบัติที่ 8 (เชิงปริมาณ) อัตราค่างวดของทรัพย์สินที่เคลื่อนย้ายได้ ที่ยังชำระไม่หมด เป็นเปอร์เซ็นต์

คุณสมบัติที่ 9 (เชิงคุณภาพ) สถานะภาพสมรสและเพศ

A91 : ชาย : หย่า

A92 : หญิง : หย่า/สมรส

A93 : ชาย : โสด

A94 : ชาย : สมรส

A95 : หญิง : โสด

คุณสมบัติที่ 10 (เชิงคุณภาพ) กู้ร่วม / ค้ำประกัน

A101 : ไม่มี

A102 : มีผู้กู้ร่วม

A103 : มีผู้ค้ำประกัน

คุณสมบัติที่ 11 (เชิงปริมาณ) ปีที่พักอาศัยในปัจจุบัน

คุณสมบัติที่ 12 (เชิงคุณภาพ) ทรัพย์สมบัติ

A121 : อสังหาริมทรัพย์

A122 : ประกันชีวิต

A123 : รถยนต์/อื่น ๆที่นอกเหนือจาก คุณสมบัติที่ 6

A124 : ไม่มี

คุณสมบัติที่ 13 (เชิงปริมาณ) อายุ

คุณสมบัติที่ 14 (เชิงคุณภาพ) มีผ่อนค่างวดกับที่อื่น

A141 : ธนาคาร

A142 : ร้านค้า / อื่น ๆ

A143 : ไม่มี

คุณสมบัติที่ 15 (เชิงคุณภาพ) บ้านที่อยู่อาศัยในปัจจุบัน

A151 : เช่า

A152 : เป็นเจ้าของ

A153 : เป็นผู้อาศัยในบ้านบิดา/มารดา/ญาติ

คุณสมบัติที่ 16 (เชิงปริมาณ) จำนวนบัตรเครดิตกับสถาบันการเงิน

คุณสมบัติที่ 17 (เชิงคุณภาพ) อาชีพปัจจุบัน

A171 : อื่น ๆ

A172 : ว่างงาน

A173 : ลูกจ้าง

A174 : ผู้จัดการ/ทำกิจการส่วนตัว

คุณสมบัติที่ 18 (เชิงปริมาณ) จำนวนผู้ที่อยู่ในอุปการะ

คุณสมบัติที่ 19 (เชิงคุณภาพ) โทรศัพท์

A191 : ไม่มี

A192 : มี ลงทะเบียนชื่อลูกค้าเอง

คุณสมบัติที่ 20: (เชิงคุณภาพ) รายได้ส่วนบุคคลต่อปี

A201 : ... < 200,000 บาท

A202 : 200,001 <= ... < 500,000 บาท

A203 : 500,001 <= ... < 1,000,000 บาท

A204 : ... >= 1,000,000 บาท

3.2 การแปลงรูปข้อมูล

เนื่องจากข้อมูลที่ได้ มีตัวแปรที่ไม่สามารถนำมาทำการคำนวณได้ จึงต้องทำการแปลงรูปแบบของข้อมูลก่อน โดยได้ทำการแปลงข้อมูลให้เป็นในรูปแบบของสปลาส (Sparse) ซึ่ง

รูปแบบนี้จะแสดงค่าเป็นเมทริกซ์ โดยที่จะเก็บค่าศูนย์ไว้น้อยมาก ซึ่งจะทำให้ค่าในการคำนวณมีความถูกต้องมากยิ่งขึ้น

3.2.1 ตัวอย่างข้อมูลหลังจากที่ได้ทำการแปลงรูปข้อมูลเป็นรูปแบบสปลาสแล้ว

4 1:-0.0434783 2:0.153846 3:0.194444 4:-0.353712 5:-0.450549 6:-0.698113 7:-0.346405 8:-0.0857143 9:-0.5 10:0.171429 11:-0.515789 12:-0.532374 13:-0.0566038 14:-0.710526 15:-0.454545 16:-0.219512 17:-0.266667 18:0.0666667 19:-0.0861607 20:-0.0463614

4 1:-0.217391 2:-0.384615 3:0.222222 4:-0.676856 5:-0.78022 6:-0.735849 7:-0.51634 8:0.0857143 9:-0.666667 10:-0.285714 11:-0.578947 12:-0.64988 13:-0.383648 14:-0.657895 15:-0.181818 16:-0.317073 17:-0.133333 18:0.2 19:0.0241656 20:0.0742636

2 1:0.347826 2:0.307692 3:0.833333 4:-0.0829694 5:-0.582418 6:-0.698113 7:0.24183 8:-0.657143 10:0.142857 11:-0.0210526 12:0.0815348 13:0.396226 14:-0.631579 15:0.272727 16:-0.560976 17:-0.2 19:-0.0861607 20:-0.0463614

4 1:-0.130435 2:-0.384615 3:0.166667 4:-0.519651 5:-0.648352 6:-0.735849 7:-0.581699 8:0.142857 9:-0.666667 10:-0.285714 11:-0.684211 12:-0.70024 13:-0.773585 14:-0.894737 15:-0.454545 16:-0.512195 17:0.533333 18:0.733333 19:0.0241656 20:0.0742636

3 1:-0.478261 2:-0.153846 3:-0.166667 4:-0.117904 5:0.230769 6:0.886792 7:-0.51634 8:0.0857143 9:-0.666667 10:-0.257143 11:0.168421 12:-0.661871 13:-0.00628931 14:0.789474 15:-0.181818 16:-0.463415 17:-0.733333 18:-0.866667 19:0.0241656 20:-0.0192654

3 1:0.478261 2:0.846154 3:0.833333 4:-0.406114 5:-0.934066 6:-0.849057 7:0.869281 8:-1 9:0.833333 10:0.457143 11:0.578947 12:0.853717 13:0.949686 14:-0.315789 15:-0.545455 16:-0.560976 17:-0.666667 18:-0.866667 19:-0.0861607 20:-0.0463614

3 1:0.0434783 2:-0.230769 3:-0.0833333 4:-0.39738 5:-0.604396 6:-0.849057 7:-0.464052 8:-0.0857143 9:-0.666667 10:-0.285714 11:-0.515789 12:-0.57554 13:-0.207547 14:-0.815789 15:0.181818 16:-0.95122 17:0.6 18:0.533333 19:0.0241656 20:0.0890876

1 1:-0.26087 2:-0.230769 3:-0.277778 4:-0.537118 5:-0.604396 6:-0.735849 7:-0.673203 8:0.257143 9:-0.833333 10:-0.2 11:-0.663158 12:-0.767386 13:-0.308176 14:-0.789474 15:-0.727273 16:-0.853659 17:0.133333 18:0.4 19:-0.0861607 20:0.0890876

4 1:-0.434783 2:-0.923077 3:-0.388889 4:-0.68559 5:-0.692308 6:-0.811321 7:-0.869281 8:0.6 9:-1 10:-0.742857 11:-0.884211 12:-0.906475 13:-0.962264 14:-0.868421 15:-0.818182 16:-0.317073 17:0.6 18:0.8 19:-0.0861607 20:0.0742636

2 1:-0.130435 2:-0.153846 3:0.611111 4:-0.187773 5:-0.67033 6:-0.660377 7:-0.0718954 8:-0.428571 9:-0.166667 10:-0.2 11:-0.242105 12:-0.230216 13:-0.459119 14:-0.868421 15:-0.636364 16:-0.317073 17:0.266667 18:0.533333 19:-0.0861607 20:-0.0192654

3.3 การกำหนดสัดส่วนขอบเขตของข้อมูล (Scaling)

จากการศึกษา [7] ได้ทราบว่า การกำหนดสัดส่วนขอบเขตของข้อมูลก่อนที่จะนำมาใช้ในหลักการซัพพอร์ตเวกเตอร์แมชชีนนั้น เป็นส่วนที่มีความสำคัญมากโดยจะมีข้อได้เปรียบหลัก ๆ คือ สามารถหลีกเลี่ยงค่าของ ค่าคุณสมบัติ ที่มีขอบเขตเป็นขนาดใหญ่ และยังสามารถหลีกเลี่ยงตัวเลขที่มีจำนวนยากต่อการคำนวณ โดยที่สัดส่วนขอบเขตของข้อมูลนั้น จะทำให้ค่าตัวเลขของแต่ละค่าคุณสมบัติ มีขนาดง่ายต่อการคำนวณ เพราะว่าค่าของ เคอร์เนล มักจะขึ้นอยู่กับการทำอินเนอร์ โพรดักท์ (Inner products) ของ ค่าคุณสมบัติหรือที่เรียกว่าค่าเวกเตอร์ของฟีทเจอร์ (Feature) ซึ่งถ้าไม่ทำการกำหนดสัดส่วนขอบเขตของข้อมูลอาจจะทำให้เกิดปัญหาของตัวเลขยาก ๆ ขึ้นมาได้ ซึ่งจากการค้นคว้าหาข้อมูล [7] นั้น ได้มีการแนะนำให้ทำการกำหนดสัดส่วนขอบเขตของข้อมูลในแต่ละค่าคุณสมบัติ ในขอบเขตของ [-1, +1] หรือ [0, 1] โดยในการทดลองนี้ทำการกำหนดสัดส่วนขอบเขตของข้อมูลในแต่ละ ค่าคุณสมบัติ ให้อยู่ในของเขต [-1, +1] เนื่องจากขอบเขตกว้างกว่า น่าที่จะให้ค่าของข้อมูลเริ่มต้นมีความละเอียดในการนำมาใช้แบ่งประเภทข้อมูลมากกว่า

3.4 เคอร์เนล (Kernel)

ฟังก์ชันเคอร์เนล ได้ถูกนำมาใช้เพื่อต้องการให้หลักการซัพพอร์ตเวกเตอร์แมชชีนสามารถแบ่งประเภทข้อมูลแบบไม่เป็นเชิงเส้นได้อย่างมีประสิทธิภาพ โดยปราศจากผลกระทบถ้าข้อมูลอยู่ในรูปแบบของมิติที่ไม่มีที่สิ้นสุดได้ เนื่องจากฟังก์ชันเคอร์เนลจะเป็นตัวช่วยในการจับคู่ค่าของข้อมูล โดยสมการดังนี้ $K(x_i, y_i) \equiv \Phi(x_i)^T \Phi(x_j)$ มาใช้ในการคำนวณอีกด้วย โดยที่ฟังก์ชันเคอร์เนล ที่ได้ถูกกล่าวถึงอยู่บ่อย ๆ มีอยู่ด้วยกันสี่ชนิด คือ

1. แบบเป็นเชิงเส้น (Linear) $K(x_i, y_i) \equiv x_i^T x_j$
2. แบบโพลิโนเมียล (Polynomial) $K(x_i, y_i) \equiv (\gamma x_i^T x_j + r)^d, \gamma > 0$
3. แบบอาร์บีเอฟ(RadialBasisFunction:RBF) $K(x_i, y_i) \equiv \exp(-\gamma \| x_i - x_j \|^2), \gamma > 0$
4. แบบซิกมอยด์ (Sigmoid) $K(x_i, y_i) \equiv \tanh(\gamma x_i^T x_j + r)$

สำหรับการทดลองครั้งนี้ได้เลือกเคอร์เนลแบบอาร์บีเอฟ (RBF) มาใช้ เนื่องจากได้ทำการทบทวนเอกสารและงานวิจัยที่เกี่ยวข้องแล้ว [6, 7, 9] ได้พบว่าเคอร์เนลแบบอาร์บีเอฟ (RBF) เป็นฟังก์ชันที่ดีที่สุดเนื่องจากเหตุผลต่าง ๆ ดังนี้

เคอร์เนลแบบอาร์บีเอฟ (RBF) เป็นการจับคู่ตัวอย่างข้อมูลที่ไม่เป็นเชิงเส้นกับตัวแปรในชั้นที่สูงกว่าไปเรื่อย ๆ ซึ่งจะไม่เหมือนกับเคอร์เนลแบบเป็นเชิงเส้นที่จะทำได้เฉพาะข้อมูลที่เป็นเชิงเส้นเท่านั้น ดังนั้นเคอร์เนลแบบอาร์บีเอฟ (RBF) จึงสามารถครอบคลุมข้อมูลที่มีความสัมพันธ์ระหว่างคลาส ลาเบล (class labels) และค่าคุณสมบัติที่ไม่เป็นเชิงเส้นได้อีกด้วย

ข้อมูลที่มีค่าตัวเลขของตัวแปรมากเกินไป จะทำให้มีผลกระทบต่อการเลือกแบบจำลองคือทำให้มีความซับซ้อนยุ่งยากมากขึ้น ดังนั้นถ้าใช้เคอร์เนลแบบโพลิโนเมียล ซึ่งแบบโพลิโนเมียลนี้จะมีค่าตัวแปรมากมายและจะส่งผลให้การทดลองนั้นเป็นไปได้ยากขึ้น เคอร์เนลแบบอาร์บีเอฟ (RBF) จะมีตัวเลขยาก ๆ ที่ใช้ในการคำนวณน้อยกว่าโดยที่ $0 < K_{ij} \leq 1$ ซึ่งต่างจากเคอร์เนลแบบโพลิโนเมียล ที่ค่าอาจจะเข้าไปสู่จำนวนที่ไม่มีที่สิ้นสุดได้ (Infinity) ($\gamma x_i^T x_j + r > 1$) หรือศูนย์ ($\gamma x_i^T x_j + r < 1$) ได้

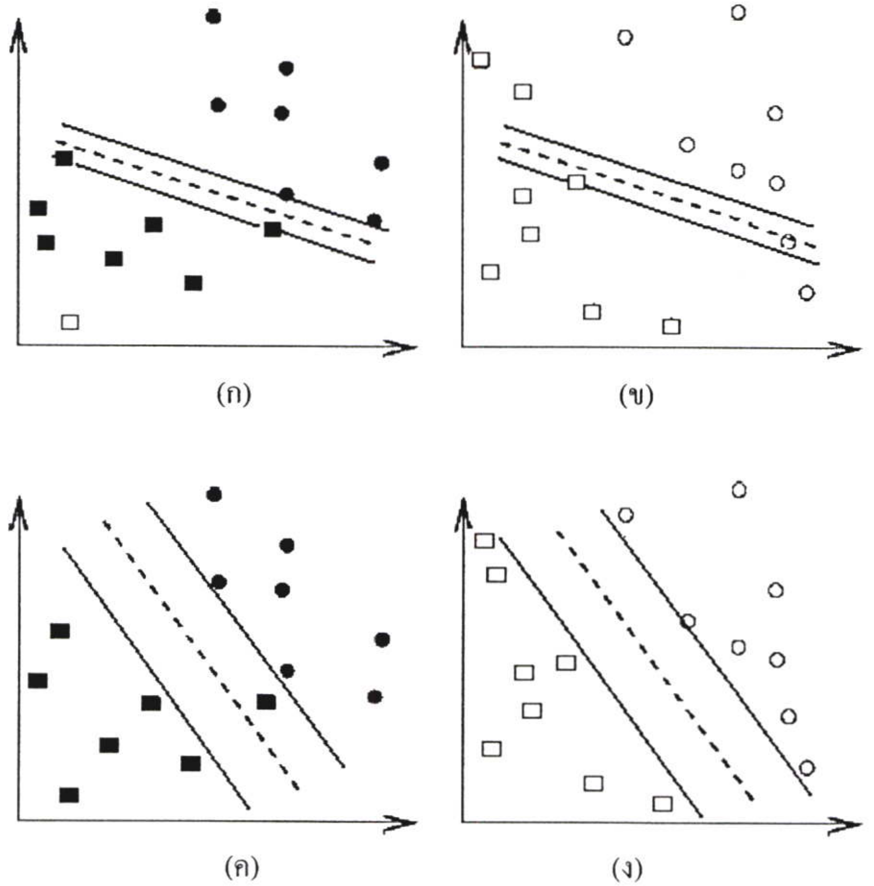
3.5 คลอส วาริเดชัน (Cross Validation) และ กริดเสิร์จ (Grid Search)

ส่วนสำคัญที่สุดในการสร้างแบบจำลองในการจัดประเภทของข้อมูลครั้งนี้คือ ค่าของตัวแปรสองตัวในขณะที่ใช้ร่วมกับเคอร์เนลแบบอาร์บีเอฟ (RBF) คือค่าของตัวแปร C และ γ โดยเป้าหมายในการเลือกค่าของตัวแปรสองตัวให้เหมาะสมที่สุดนี้คือ เพื่อที่จะทำให้การจัดประเภทข้อมูลที่เข้ามาใหม่ ๆ ได้ถูกจัดให้อยู่ในประเภทที่เหมาะสมและถูกต้องมากที่สุด แต่ค่าของสองตัวนี้จะถูกเลือกจากวิธีการ คลอส วาริเดชัน ซึ่งเป็นวิธีที่นิยมนำมาใช้ในปัจจุบัน

โดยเริ่มต้นจากทำการแบ่งข้อมูลตัวอย่างในชุดของการฝึกฝนออกเป็น สองส่วนเท่า ๆ กัน ซึ่งในการทดลองนี้ ได้ทำการแบ่งข้อมูลตัวอย่างในส่วนของการฝึกฝนจาก 600 ข้อมูลออกเป็นสองชุดชุดละ 300 ข้อมูล โดยส่วนแรก (subset *V*) จะถูกนำไปใช้ในการฝึกฝน ก่อนที่จะถูกนำไปใช้ในการทดสอบกับข้อมูลอีกส่วนหนึ่งที่เหลือ (subset *V*-1) วิธีคลอส วาริเดชัน นั้นจะสามารถป้องกันปัญหาของการที่มีข้อมูลมีความลงตัวมากจนเกินไป (Over Fit) ได้

โดยจะสามารถอธิบายได้จาก รูปที่ 3.1 ซึ่งแสดงถึงปัญหาของการจัดประเภทข้อมูล สองประเภท (สี่เหลี่ยม และ วงกลม) โดยที่ข้อมูลวงกลมดำ และสี่เหลี่ยมดำนั้น เป็นข้อมูลของชุดในส่วนของการฝึกฝน ในขณะที่ข้อมูลที่เป็นวงกลมขาวและสี่เหลี่ยมขาว เป็นข้อมูลของชุดในส่วนของการทดสอบ ซึ่งในการทดสอบความถูกต้องของการจัดประเภทข้อมูลในรูปที่ 3.1 (ก) และ 3.1 (ข) นั้น จะเห็นได้ว่า ไม่ได้ผลดีนัก เนื่องจากข้อมูลในส่วนการฝึกฝนนั้นไม่เกิดการข้ามกลุ่มกันขึ้นเลย หรือเรียกอีกทางหนึ่งได้ว่า จำนวนค่าของคุณสมบัติมีความลงตัวพอดีมากเกินไป ทำให้เมื่อเกิดมีค่าคุณสมบัติอื่น ๆ มากขึ้นทำให้เกิดปัญหานี้ได้ จึงส่งผลในการทดสอบข้อมูล ทำให้ไม่ได้ค่าที่ถูกต้องมากนักในส่วนของการทดสอบ ซึ่งในอีกทางหนึ่ง ถ้าสังเกตข้อมูลในชุดของการฝึกฝนและชุดของการทดสอบ จากรูปที่ 3.1 (ค) และ 3.1 (ง) แล้วจะเห็นว่าข้อมูลในส่วนของการฝึกฝนนั้นมี

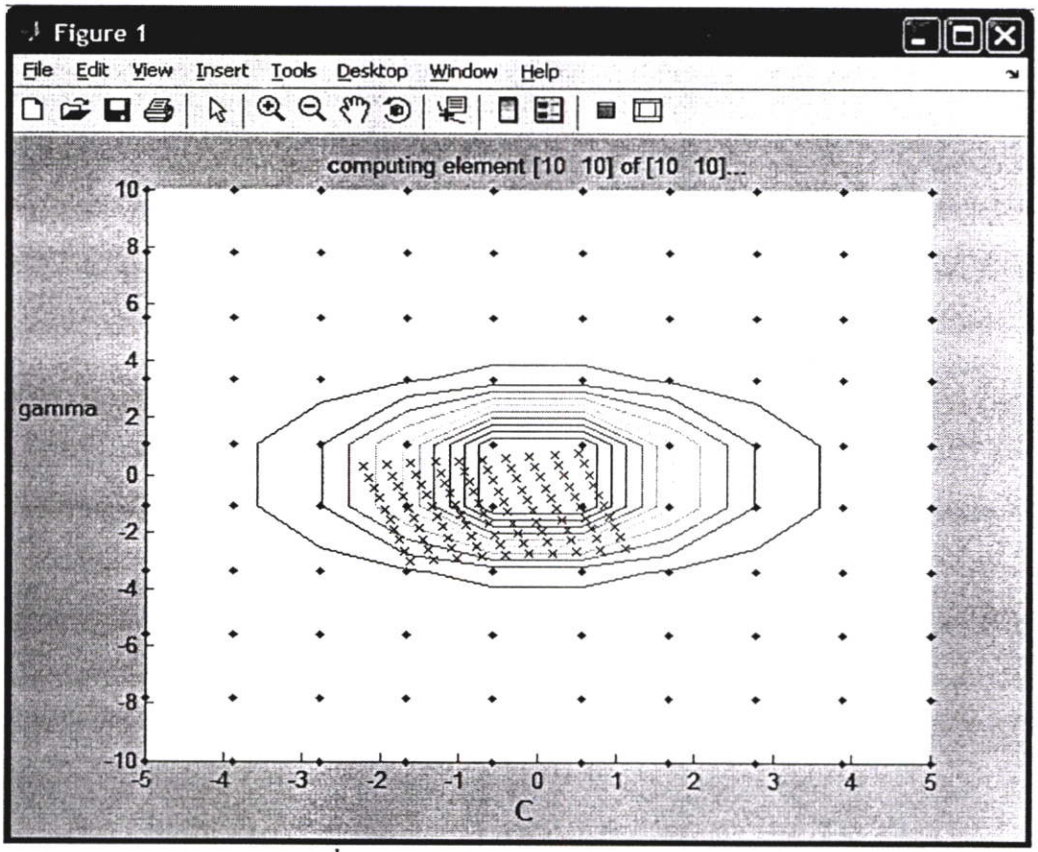
ค่าที่มีการข้ามกลุ่มกันบ้าง จึงทำให้การที่ทำการ คลอส วาริเดชัน นั้นดียิ่งขึ้น แล้วยังส่งผลให้ค่าข้อมูลในการทดสอบ มีความถูกต้องมากขึ้นอีกด้วย



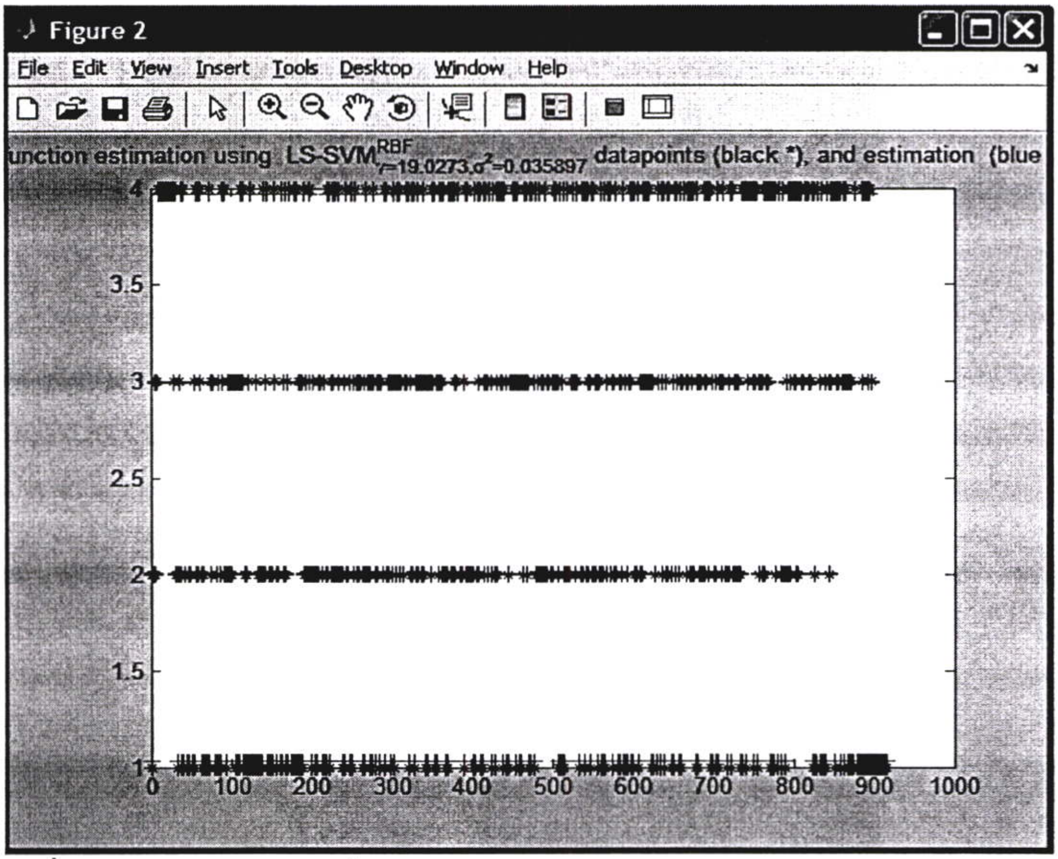
รูปที่ 3.1 แสดงถึงการจัดประเภทข้อมูลด้วยวิธี คลอส วาริเดชัน (ก) และ (ค) การฝึกฝนข้อมูล (ข) และ (ง) การทดสอบข้อมูล

ในการทดลองนี้ยังได้นำ วิธีกริดเสิร์จ มาใช้ในการหาค่า C และ γ ที่ดีที่สุดมาใช้ในการทำ คลอส วาริเดชัน อีกด้วย ในความเป็นจริงแล้วมีวิธีดี ๆ มากมายในการหาค่าทำนองนี้ กรณีนี้ที่ได้เลือกวิธีกริดเสิร์จนั้น เนื่องจากวิธีกริดเสิร์จเป็นวิธีพื้นฐาน และจากการทดลองนี้มีตัวแปรเพียงสองตัวที่ต้องการหาค่า และตัวแปรมีค่าไม่ได้ขึ้นอยู่ซึ่งกันและกันเลย ทำให้สามารถใช้เวลาในการคำนวณน้อยกว่าการที่จะใช้วิธีการขั้นสูงต่าง ๆ โดยเริ่มต้นจากเอาค่าของ (C,γ) ที่ถูกนำมาใช้ในการฝึกฝน แล้วเลือกค่าที่ทำให้การคลอส วาริเดชัน มีค่าถูกต้องดีที่สุด ซึ่งได้พบว่าลำดับเพิ่มขึ้นของค่าเลขยกกำลังพีชคณิต (Exponential) ของ C และ γ เป็นค่าตัวแปรที่ดี เช่น $C = 2^{-5}, 2^{-3}, \ldots, 2^{15}$ $\gamma = 2^{-15}, 2^{-13}, \ldots, 2^{3}$ โดยรูปที่ 3.2 แสดงถึงการค้นหาค่า C และ γ ด้วยวิธีของกริดเสิร์จอย่างคร่าวๆในรอบแรก เพื่อที่จะได้ค่าขอบเขตเบื้องต้นและจะทำการค้นหาค่าที่ดีที่สุดต่อไปโดยรูปที่

3.3 ได้แสดงถึงผลลัพธ์เมื่อทำการนำค่า C และ γ ที่ได้มาจากการค้นหาด้วยวิธีของกริดเสิร์จอย่างคร่าวๆ

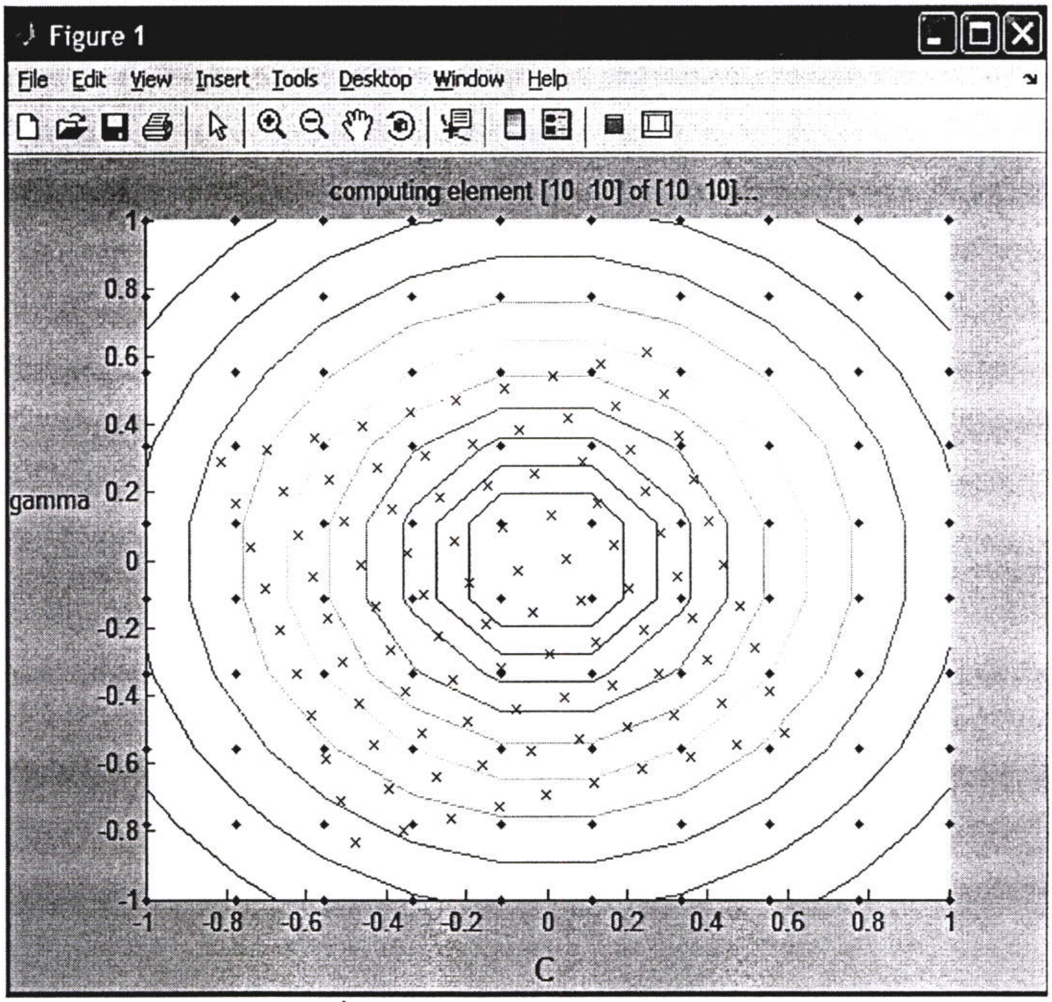


รูปที่ 3.2 การใช้วิธีกริดเสิร์จอย่างคร่าว ๆ



รูปที่ 3.3 กราฟของข้อมูลจริงที่ได้มาจากการประมาณค่าจากแบบจำลองกริดเสริจ์อย่างคร่าว ๆ

เนื่องจากวิธีกริดเสริจ์ยังคงเป็นวิธีที่ใช้เวลาในการคำนวณมาก จึงได้ใช้วิธีในการเลือกตัวเลขอย่างคร่าว ๆก่อน หลังจากที่ได้ค่าขอบเขตที่ดีแล้ว จะทำการค้นหาเข้าไป ด้วยขอบเขตที่ละเอียดขึ้นไปอีกจึงทำให้ได้ค่าที่จำเพาะเจาะจงมากขึ้น โดยค่าจากการค้นหาอย่างคร่าว ๆ ของ (C,γ) คือ $(2^{3}, 2^{-5})$ และหลังจากนั้นได้ทำการค้นหาในค่าละเอียดมากขึ้นทำให้ได้ค่า (C,γ) คือ $(2^{3.25}, 2^{-5.25-})$ ซึ่งแสดงดังรูปที่ 3.4 โดยที่แกนY แสดงถึงค่าของ γ และแกน X แสดงถึงค่าของ C



รูปที่ 3.4 การใช้วิธีกริดเสิร์จอย่างละเอียด

หลังจากที่ได้ค่า (C,γ) ที่ดีที่สุดแล้ว จึงนำค่านี้ไปใช้ในข้อมูลในชุดของการฝึกฝนทั้งหมด เพื่อจะสร้างแบบจำลองในการจัดประเภทข้อมูล ไปใช้ในส่วนของการทดสอบข้อมูล

3.6 การทดสอบข้อมูล

หลังจากที่ทำการฝึกฝนข้อมูลแล้ว จะทำให้ได้แบบจำลองที่ได้มาจากการหาค่า (C,γ) ที่ดีที่สุด มาใช้ในส่วนของการทดสอบข้อมูลโดยทำด้วยวิธีเดียวกันกับการฝึกฝนข้อมูล แต่จะทำการเปลี่ยนข้อมูลชุดใหม่ที่ไว้สำหรับทดสอบข้อมูล โดยเฉพาะที่ได้แบ่งไว้ในการจัดเตรียมข้อมูลก่อนการทดลอง ซึ่งผลของการจัดประเภทข้อมูลจะแสดงในบทถัดไป

บทที่ 4
ผลการทดลองและการประเมินประสิทธิภาพ

ในบทนี้จะกล่าวถึง ผลการทดลองโดยจะแสดงผลของแบบจำลองในการจัดประเภทของลูกค้า และการตรวจสอบหาค่าความผิดพลาด ซึ่งเปรียบเทียบระหว่างหลักการของซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine : SVM) และ ลีซท์ สแควร์ ซัพพอร์ตเวกเตอร์แมชชีน (Least Square Support Vector Machine : LS-SVM) ด้วยวิธีในการหาค่าเฉลี่ยของความคลาดเคลื่อนสัมบูรณ์ (Mean Absolute Error : MAE) ซึ่งเป็นวิธีการคำนวณหาค่าความคลาดเคลื่อนเฉลี่ย ระหว่างค่าจริง และค่าพยากรณ์เทียบกับค่าจริง และการหาค่าเฉลี่ยของเปอร์เซ็นต์ความคลาดเคลื่อนสัมบูรณ์ (Mean Absolute Percent Error : MAPE)

4.1 ผลการทดลองที่ได้จากการทดสอบ (Testing) ข้อมูล

จากการทดลองจะได้ คอนฟิวส์ชันเมทริกซ์ (Confusion Matrix) ที่แสดงค่าข้อมูลจริงและส่วนที่ได้ทำนายในการจัดประเภทข้อมูลไว้ ซึ่งคอนฟิวส์ชันเมทริกซ์เป็นค่าที่สามารถคำนวณเพื่อใช้ในการตรวจสอบความถูกต้องในการจัดประเภทข้อมูลที่ได้จากการทดลองนี้ เมื่อนำวิธีการของซัพพอร์ตเวกเตอร์แมชชีนมาใช้ จะทำให้สามารถแบ่งประเภทของข้อมูลได้เพียงสองประเภทเท่านั้น ซึ่งดูผลได้จากตารางที่ 4.1

ตารางที่ 4.1 คอนฟิวส์ชันเมทริกซ์ที่ได้มาจากหลักการของซัพพอร์ตเวกเตอร์แมชชีน โดยที่ 1 = ดี 2 = ไม่ดี สำหรับอัตราส่วน 600:200

คอนฟิวส์ชัน เมทริกซ์		ค่าที่พยากรณ์ได้ (Predicted)	
(SVM)		1	2
ค่าจริง	1	0.8700	0.2500
(Actual)	2	0.1300	0.7500

ซึ่งสามารถอ่านค่าจากตารางที่ 4.1 ได้ดังนี้

1. ความถูกต้องในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 1 = 87 %
2. ความผิดพลาดในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 1 = 13 %
3. ความถูกต้องในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 2 = 75 %
4. ความผิดพลาดในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 2 = 25 %

ตารางที่ 4.2 คอนฟิวส์ชันเมทริกซ์ที่ได้มาจากหลักการของลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีน โดยที่ 1 = ดี 2 = ค่อนข้างดี 3 = มีความเสี่ยง 4 = ไม่ดี สำหรับอัตราส่วน 600:200

คอนฟิวส์ชัน เมทริกซ์ (LS-SVM)		ค่าที่พยากรณ์ได้ (Predicted) 1	2	3	4
	1	1.000	0	0	0
ค่าจริง	2	0	0.9147	0.0401	0.1363
(Actual)	3	0	0.0500	0.9212	0.0779
	4	0	0.0353	0.0387	0.7858

ซึ่งสามารถอ่านค่าจากตารางที่ 4.2 ได้ดังนี้

1. ความถูกต้องในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 1 = 100 %
2. ความผิดพลาดในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 1 = 0 %
3. ความถูกต้องในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 2 = 91.47 %
4. ความผิดพลาดในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 2 = 8.53 %
5. ความถูกต้องในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 3 = 92.12 %
6. ความผิดพลาดในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 3 = 7.88 %
7. ความถูกต้องในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 4 = 78.58 %
8. ความผิดพลาดในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 4 = 21.42 %

ตารางที่ 4.3 คอนฟิวส์ชันเมทริกซ์ที่ได้มาจากหลักการของซัพพอร์ตเวกเตอร์แมชชีน โดยที่ 1 = ดี 2 = ไม่ดี สำหรับอัตราส่วน 700:200

คอนฟิวส์ชัน เมทริกซ์ (SVM)		ค่าที่พยากรณ์ได้ (Predicted) 1	2
ค่าจริง	1	0.9100	0.2300
(Actual)	2	0.0900	0.7700

ซึ่งสามารถอ่านค่าจากตารางที่ 4.3 ได้ดังนี้

1. ความถูกต้องในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 1 = 91 %
2. ความผิดพลาดในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 1 = 9 %
3. ความถูกต้องในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 2 = 77 %
4. ความผิดพลาดในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 2 = 23 %

ตารางที่ 4.4 คอนฟิวส์ชันเมทริกซ์ที่ได้มาจากหลักการของลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีน โดยที่ 1 = ดี 2 = ค่อนข้างดี 3 = มีความเสี่ยง 4 = ไม่ดี สำหรับอัตราส่วน 700:200

คอนฟิวส์ชัน เมทริกซ์ (LS-SVM)		ค่าที่พยากรณ์ได้ (Predicted) 1	2	3	4
	1	1.000	0	0	0
ค่าจริง	2	0	0.9357	0.0334	0.0734
(Actual)	3	0	0.03417	0.9312	0.0379
	4	0	0.03013	0.0354	0.8887

ซึ่งสามารถอ่านค่าจากตารางที่ 4.4 ได้ดังนี้

1. ความถูกต้องในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 1 = 100%
2. ความผิดพลาดในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 1 = 0 %
3. ความถูกต้องในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 2 = 93.57 %
4. ความผิดพลาดในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 2 = 6.43 %
5. ความถูกต้องในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 3 = 93.12 %
6. ความผิดพลาดในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 3 = 6.88 %
7. ความถูกต้องในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 4 = 88.87 %
8. ความผิดพลาดในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 4 = 11.13 %

ตารางที่ 4.5 คอนฟิวส์ชันเมทริกซ์ที่ได้มาจากหลักการของซัพพอร์ตเวกเตอร์แมชชีน โดยที่ 1 = ดี 2 = ไม่ดี สำหรับอัตราส่วน 800:200

คอนฟิวส์ชัน เมทริกซ์ (SVM)		ค่าที่พยากรณ์ได้ (Predicted) 1	2
ค่าจริง	1	0.9300	0.2100
(Actual)	2	0.0700	0.7900

ซึ่งสามารถอ่านค่าจากตารางที่ 4.5 ได้ดังนี้

1. ความถูกต้องในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 1 = 93 %
2. ความผิดพลาดในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 1 = 7 %
3. ความถูกต้องในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 2 = 79 %
4. ความผิดพลาดในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 2 = 21 %

ตารางที่ 4.6 คอนฟิวส์ชันเมทริกซ์ที่ได้มาจากหลักการของลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีน โดยที่ 1 = ดี 2 = ค่อนข้างดี 3 = มีความเสี่ยง 4 = ไม่ดี สำหรับอัตราส่วน 800:200

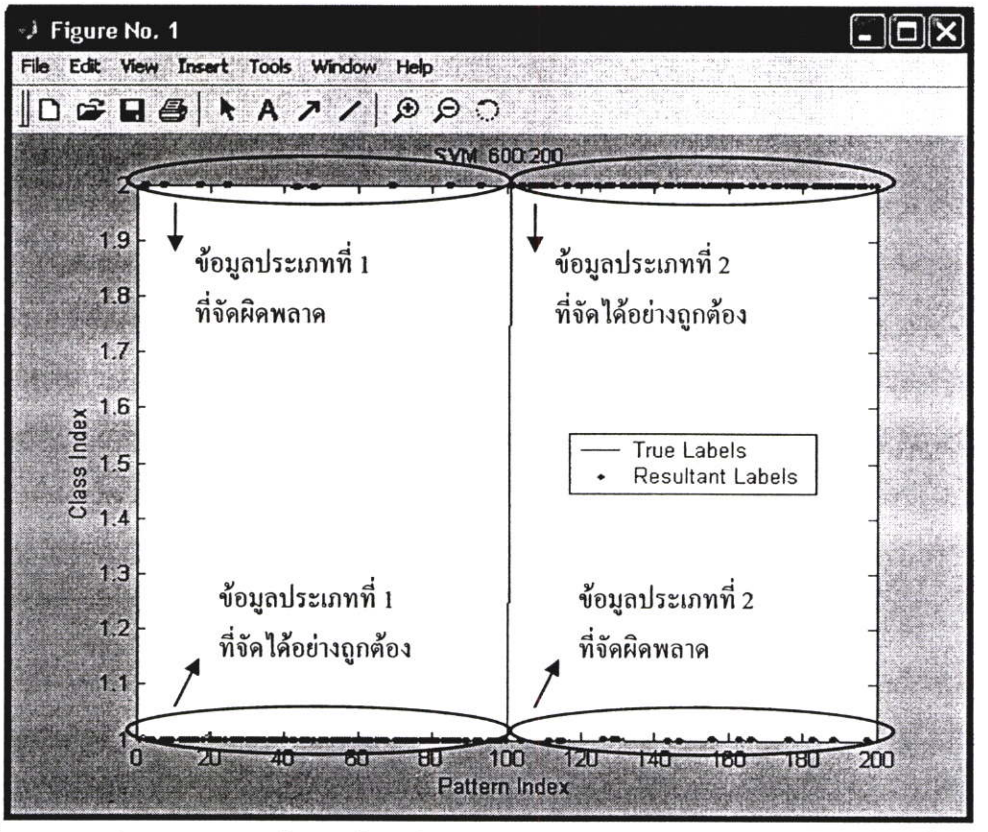
คอนฟิวส์ชัน เมทริกซ์		ค่าที่พยากรณ์ได้ (Predicted)			
(LS-SVM)		1	2	3	4
	1	1.000	0	0	0
ค่าจริง	2	0	0.9521	0.0151	0.0462
(Actual)	3	0	0.015	0.9614	0.0502
	4	0	0.0329	0.0235	0.9036

ซึ่งสามารถอ่านค่าจากตารางที่ 4.6 ได้ดังนี้

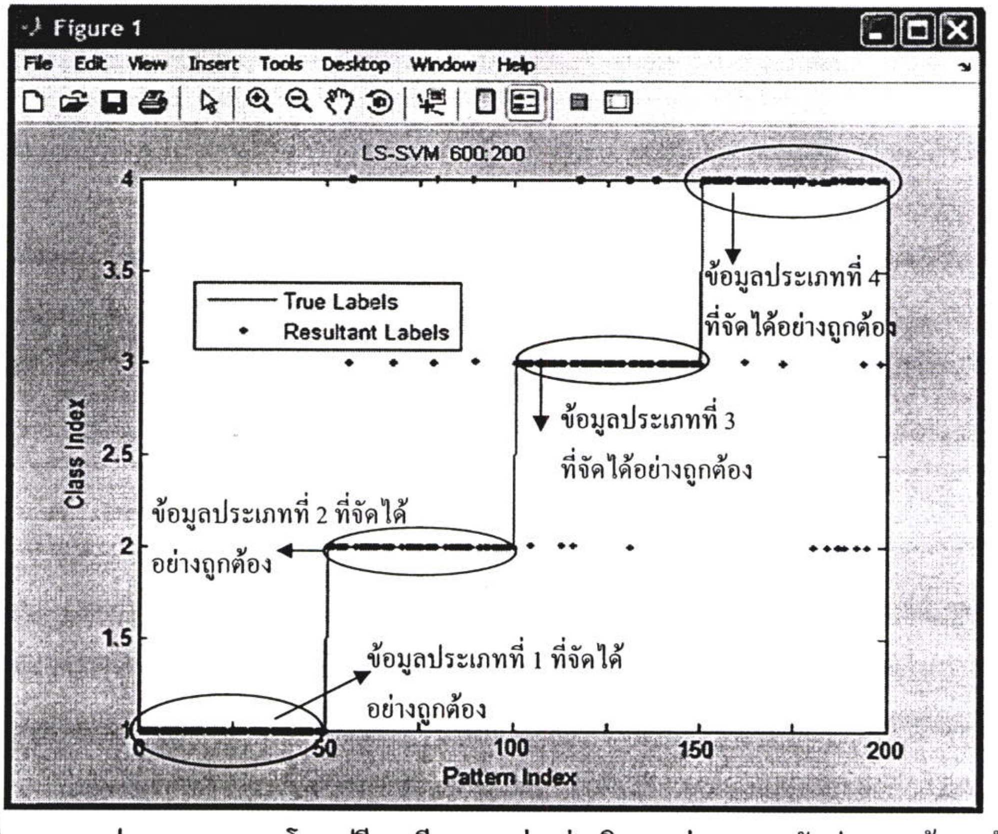
1. ความถูกต้องในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 1 = 100 %
2. ความผิดพลาดในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 1 = 0 %
3. ความถูกต้องในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 2 = 95.21 %
4. ความผิดพลาดในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 2 = 4.79 %
5. ความถูกต้องในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 3 = 96.14 %
6. ความผิดพลาดในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 3 = 3.86 %
7. ความถูกต้องในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 4 = 90.36 %
8. ความผิดพลาดในการทำนายลูกค้าให้อยู่ในกลุ่มที่ 4 = 9.64 %

เมื่อทำการสร้างกราฟเปรียบเทียบ ระหว่างค่าจริงและค่าที่ได้จากการที่ใช้แบบจำลองในการจัดประเภทข้อมูลของทั้งสองหลักการ ซึ่งได้แก่ ซัพพอร์ตเวกเตอร์แมชชีน และ ลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีนนั้น ได้แสดงดังรูปที่ 4.1 ถึงรูปที่ 4.6

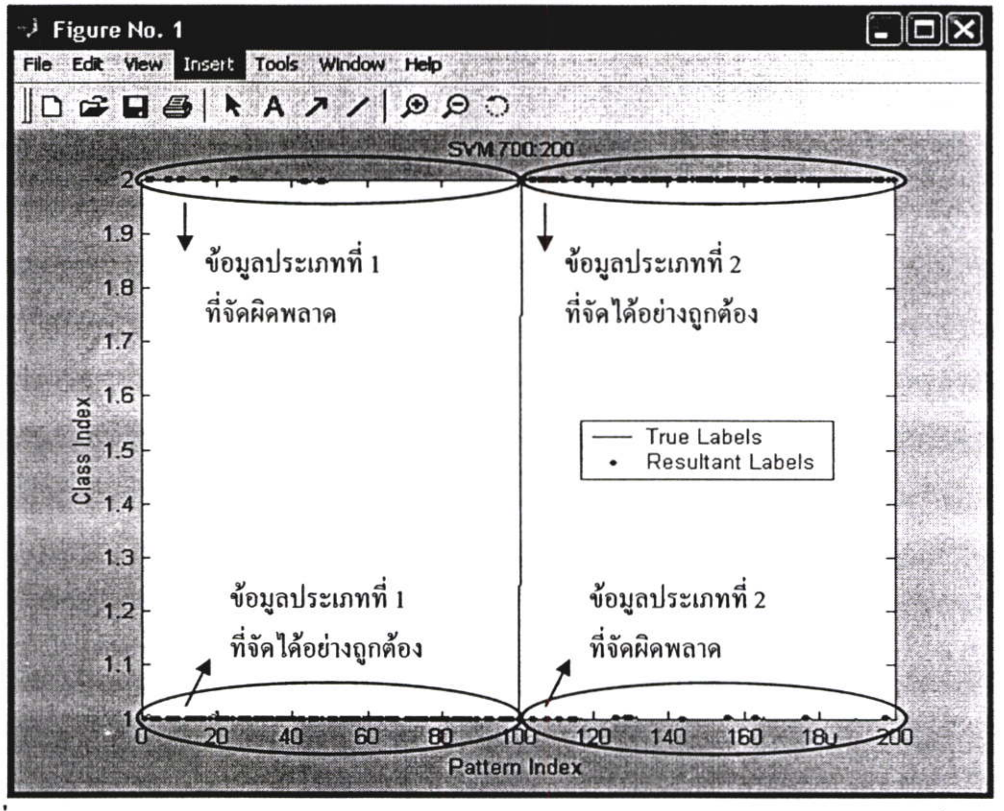
จากรูปดังกล่าวได้แสดงให้เห็นว่า เส้นขีด (—) แสดงถึงค่าของประเภทข้อมูลจริง ที่อยู่ในส่วนของการทดสอบข้อมูล และจุดวงกลม (•) แสดงถึงค่าที่ถูกพยากรณ์เพื่อทำการจัดแบ่งแยกประเภทของข้อมูล ที่ผ่านแบบจำลองมาแล้ว ซึ่งสามารถแบ่งข้อมูลออกเป็นสองประเภท เมื่อใช้หลักการของซัพพอร์ตเวกเตอร์แมชชีน (รูปที่ 4.1 4.3 และ 4.5) และทำการจัดแบ่งแยกประเภทข้อมูลออกเป็นสี่ประเภท เมื่อใช้หลักการของ ลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีน (รูปที่ 4.2 4.4 และ 4.6) โดยที่แกน X (แนวนอน) เป็นจำนวนของข้อมูล และแกน Y (แนวตั้ง) เป็นประเภทของข้อมูล



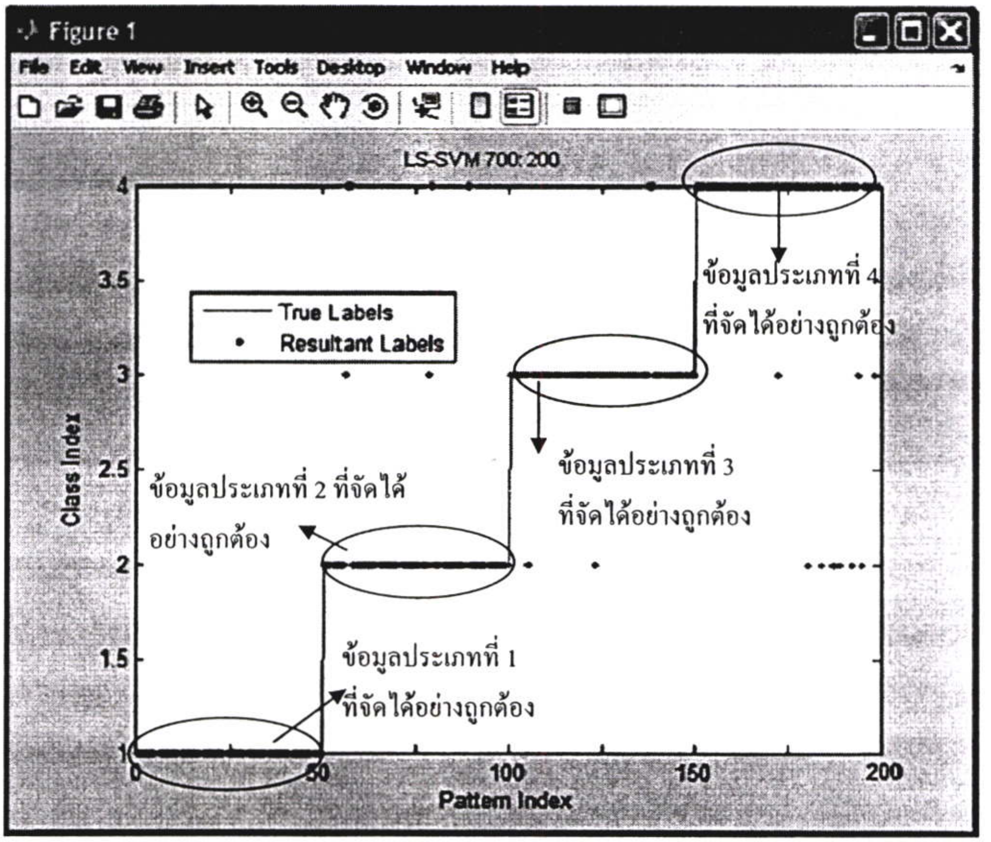
รูปที่ 4.1 กราฟผลการทดลองโดยเปรียบเทียบระหว่างค่าจริงและค่าจากการจัดประเภทข้อมูลด้วยหลักการของซัพพอร์ตเวกเตอร์แมชชีน ด้วยอัตราส่วน 600:200



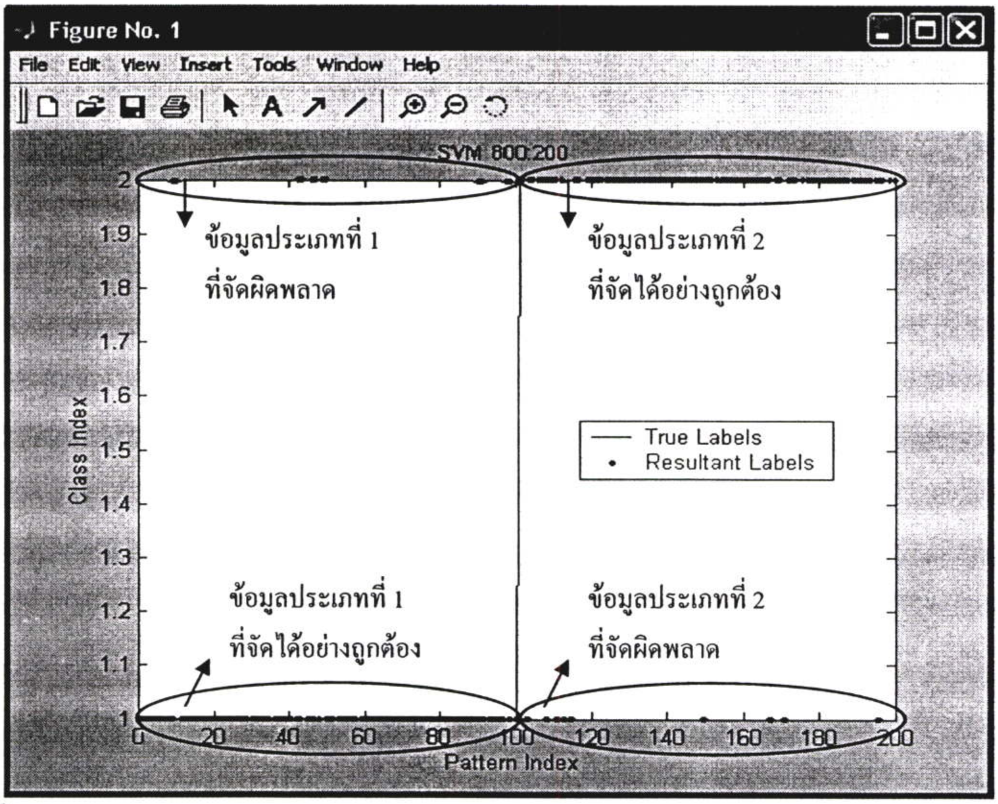
รูปที่ 4.2 กราฟผลการทดลองโดยเปรียบเทียบระหว่างค่าจริงและค่าจากการจัดประเภทข้อมูลด้วยหลักการของลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีน ด้วยอัตราส่วน 600:200



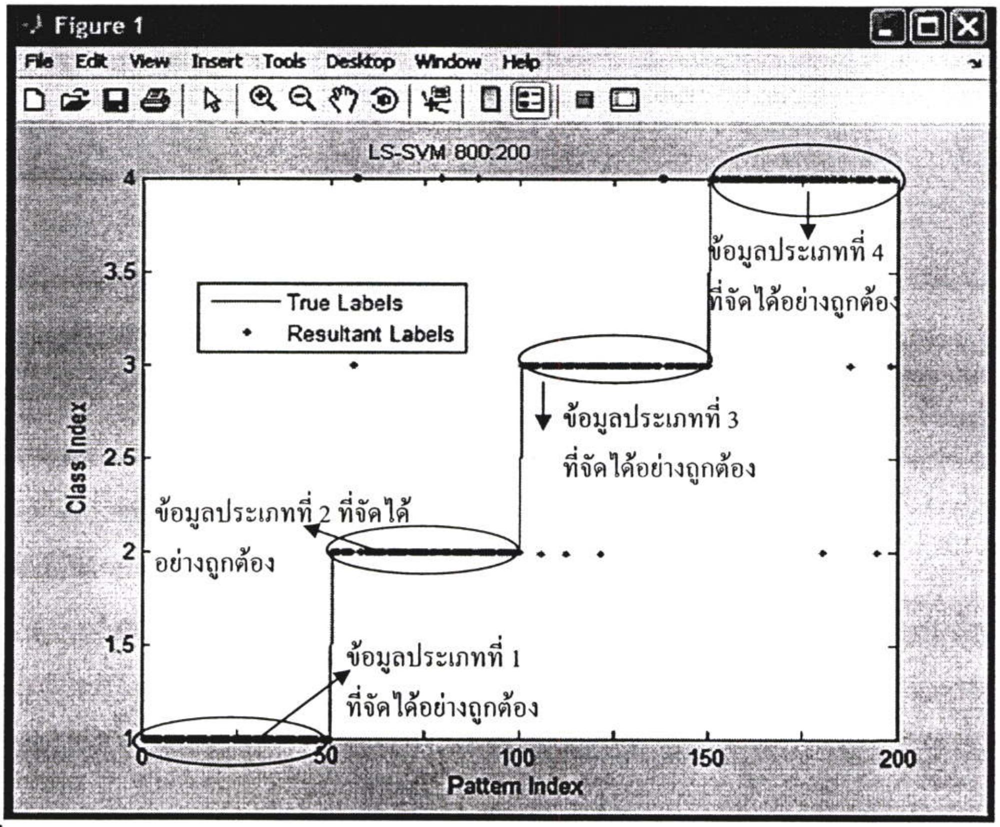
รูปที่ 4.3 กราฟผลการทดลองโดยเปรียบเทียบระหว่างค่าจริงและค่าจากการจัดประเภทข้อมูลด้วยหลักการของซัพพอร์ตเวกเตอร์แมชชีน ด้วยอัตราส่วน 700:200



รูปที่ 4.4 กราฟผลการทดลองโดยเปรียบเทียบระหว่างค่าจริงและค่าจากการจัดประเภทข้อมูลด้วยหลักการของลีซท์ สแควร์ ซัพพอร์ตเวกเตอร์แมชชีน ด้วยอัตราส่วน 700:200



รูปที่ 4.5 กราฟผลการทดลองโดยเปรียบเทียบระหว่างค่าจริงและค่าจากการจัดประเภทข้อมูลด้วยหลักการของซัพพอร์ตเวกเตอร์แมชชีน ด้วยอัตราส่วน 800:200



รูปที่ 4.6 กราฟผลการทดลองโดยเปรียบเทียบระหว่างค่าจริงและค่าจากการจัดประเภทข้อมูลด้วยหลักการของลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีน ด้วยอัตราส่วน 800:200

จากการวิเคราะห์ข้อมูลจากตารางที่ 4.1 ถึง ตารางที่ 4.6 ที่แสดงถึงคอนฟิวส์ชันเมทริกซ์ของทั้งหลักการซัพพอร์ตเวกเตอร์แมชชีนและลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีนนั้น รวมถึงค่าจากรูปที่ 4.1 ถึงรูปที่ 4.6 ที่ได้แสดงผลการทดลองออกมาเป็นกราฟนั้น จะเห็นได้ว่า การจัดแบ่งประเภทข้อมูลให้อยู่ในประเภทที่ 1 นั้นมีความสมบูรณ์ 100 เปอร์เซ็นต์ เนื่องจากกลุ่มบุคคลที่สามารถเข้ามาอยู่ในประเภทที่ 1 หรือเรียกว่าประเภทดีมาก ได้นั้นค่อนข้างที่จะมีค่าคุณสมบัติชัดเจน แตกต่างกับค่าคุณสมบัติของบุคคลในประเภทอื่น ๆ ซึ่งค่อนข้างที่จะมีความคลุมเครือและซ้ำซ้อนมากกว่า ทำให้สามารถแบ่งแยกประเภทได้แม่นยำมากยิ่งขึ้น ดังนั้นจะเห็นได้ว่าการที่มีข้อมูลที่ดีไม่มีการขาดหาย และมีหลักการและค่าคุณสมบัติในการแบ่งประเภทข้อมูลอย่างชัดเจนไม่คลุมเครือนั้น จะสามารถทำให้ง่ายต่อการสร้างแบบจำลอง พร้อมทั้งยังจะทำให้แบบจำลองมีประสิทธิภาพและแม่นยำมากขึ้นอย่างเห็นได้ชัดอีกด้วย

4.2 การประเมินประสิทธิภาพของการจัดประเภทข้อมูล

การเปรียบเทียบค่าความถูกต้อง ของการจัดประเภทข้อมูลลูกค้าของสถาบันการเงิน ในประเทศไทย ได้ถูกแสดงในตารางที่ 4.7 โดยการวัดค่าเพื่อจะมาทำการประเมินประสิทธิภาพของแบบจำลอง จะทำการหาค่าจากวิธีดังนี้

การหาค่าเฉลี่ยของความคลาดเคลื่อนสัมบูรณ์

$$MAE = \frac{1}{T}\sum_{t=1}^{T}\left|y_t - \hat{y}_t\right| \tag{4.1}$$

การหาค่าเฉลี่ยของเปอร์เซ็นต์ความคลาดเคลื่อนสัมบูรณ์

$$MAPE = \frac{1}{T}\sum_{t=1}^{T}\left(\left|\frac{y_t - \hat{y}_t}{y_t}\right| \times 100\right) \tag{4.2}$$

ตารางที่ 4.7 การเปรียบเทียบความถูกต้องของการจัดประเภทข้อมูลจากจำนวนข้อมูลในส่วนของการฝึกฝนที่แตกต่างกันและข้อมูลในส่วนของการทดสอบเป็นจำนวนคงที่

ข้อมูลจาก	จำนวนข้อมูล		MAE		MAPE	
สถาบันการเงิน	การฝึกฝน	การทดสอบ	SVM	LS-SVM	SVM	LS-SVM
จำนวนข้อมูล ทั้งหมด:1000	600	200	2.4	2.12	2.1%	1.9%
จำนวน คุณสมบัติ (Attribute) : 20	700	200	1.9	1.68	1.7%	1.45%
	800	200	1.2	0.98	1.1%	0.87%

จากตารางที่ 4.7 สามารถสังเกตได้ว่า ถ้าใช้จำนวนข้อมูลในส่วนของการทดสอบ ที่คงที่แล้วเปรียบเทียบกันโดยให้ค่าจำนวนข้อมูลในการฝึกฝนเปลี่ยนแปลงไป โดยมีค่าเพิ่มขึ้นแล้ว จะเห็นได้ว่าค่าของความผิดพลาดจะมีค่าลดลงอย่างต่อเนื่อง ซึ่งทำให้รู้ได้ว่า ถ้ามีการให้ค่าข้อมูลในส่วนของการฝึกฝนยิ่งมาก ยิ่งจะทำให้ความผิดพลาดในการจัดประเภทข้อมูลน้อยลง แต่ต้องตระหนักไว้ด้วยว่าถ้าข้อมูล ณ ส่วนใดส่วนหนึ่งมากเกินไป จะทำให้เกิดปัญหาที่ข้อมูลมีความลงตัวมากจนเกินไป (Over fit) ได้ หรือถ้าข้อมูลน้อยเกินไปจะทำให้เกิดความผิดพลาดมากขึ้นด้วย ดังนั้นจึงต้องเลือกจำนวนที่พอดีที่สุดมาใช้ในการทดลอง และยิ่งกว่านั้น จะเห็นได้ว่าการจัดแบ่งประเภทของข้อมูลด้วยหลักการของทั้ง ซัพพอร์ตเวกเตอร์แมชชีน และ ลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีน จะได้ผลที่แตกต่างกันแต่ยังคงเรียงลำดับในเรื่องของปริมาณข้อมูลในการ ฝึกฝนเหมือนเดิม โดยที่จะเห็นได้ว่าหลักการ ลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีน ยังคงขึ้นอยู่กับปริมาณข้อมูล ถ้ายิ่งมีข้อมูลในการฝึกฝนมากยิ่งขึ้น จะทำให้ได้ค่าความผิดพลาดลดน้อยลง หรืออีกนัยหนึ่งคือมีความถูกต้องมากยิ่งขึ้นนั่นเอง แต่เมื่อทำการเปรียบเทียบระหว่าง ซัพพอร์ตเวกเตอร์แมชชีน กับ ลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีน แล้ว จะเห็นได้ว่าหลักการ ลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีนนั้น ได้แสดงถึงความละเอียดของการจัดแบ่งประเภทข้อมูลดังตารางที่ 4.7 นั่นเอง และนอกจากนั้นยังสามารถวิเคราะห์ได้อีกว่า หลักการ ลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีน ยังให้ผลความถูกต้องแม่นยำในการจัดประเภทข้อมูลแบบที่ไม่เป็นเชิงเส้นได้มากกว่าหลักการของ ซัพพอร์ตเวกเตอร์แมชชีน

บทที่ 5
สรุปผลการวิจัย และข้อเสนอแนะ

5.1 สรุปผลการวิจัย

จากการที่สังคมปัจจุบันมีการแข่งขันกันสูงในทุก ๆ ด้าน ไม่ว่าจะเป็นการแข่งขันทางเศรษฐกิจ การเมือง หรือแม้รวมถึงการแข่งขันในสถาบันการเงิน ซึ่งสถาบันการเงินถือว่าเป็นองค์กรหนึ่ง ที่มีการเก็บข้อมูลเกี่ยวกับลูกค้าที่เข้ามาใช้ในสถาบันการเงินทั้งในปัจจุบันและในอดีตอยู่เป็นจำนวนมาก ผู้วิจัยได้เห็นว่าข้อมูลเหล่านี้ควรที่จะสามารถนำมาใช้ประโยชน์ได้มากแต่สภาวะปัจจุบัน ข้อมูลเหล่านี้ยังไม่ได้ถูกนำมาใช้งานให้เป็นประโยชน์อย่างจริงจัง งานวิจัยนี้จึงมีวัตถุประสงค์ เพื่อสร้างแบบจำลองในการให้คะแนนความน่าเชื่อถือของลูกค้าจากสถาบันการเงินในประเทศไทย โดยเปรียบเทียบวิธีการทำงานของซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine : SVM) และลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีน (Least Square Support Vector Machine : LS-SVM) เพื่อที่จะนำมาใช้ ในการจำแนกประเภทของลูกค้าจากสถาบันการเงินในประเทศไทย โดยดูจากคุณสมบัติของลูกค้าที่มีผลต่อความน่าเชื่อถือต่อสถาบันการเงินที่จะอนุมัติเงินกู้ให้ เมื่อสามารถหาแบบจำลองที่มีความน่าเชื่อถือได้สูงแล้วนั้น ก็สามารถที่จะนำข้อมูลของลูกค้าใหม่มาทำการทดสอบโดยผ่านแบบจำลองนี้ ในที่สุดก็จะสามารถทำนายได้ว่าลูกค้าคนนี้ควรจะอยู่ในประเภทลูกค้ากลุ่มไหนและมีความน่าเชื่อถือมากเพียงใด เพื่อที่จะลดอัตราความเสี่ยงต่อสถาบันการเงินเอง

วิทยานิพนธ์ฉบับนี้นำเสนอหลักการซัพพอร์ตเวกเตอร์แมชชีน และลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีนมาเปรียบเทียบการสร้างแบบจำลองที่มีประสิทธิภาพ และยังสามารถแบ่งประเภทข้อมูลออกเป็นหลายประเภทด้วยกัน โดยผู้วิจัยได้ใช้ความแม่นยำและค่าความผิดพลาดมาเป็นตัวประเมินผล

หลักการซัพพอร์ตเวกเตอร์แมชชีนและ ลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีน นั้นเป็นวิธีที่มีประสิทธิภาพในการจัดแบ่งประเภทของข้อมูลมาก พร้อมทั้งยังเป็นที่นิยมต่อนักวิจัยมากเช่นกัน ซึ่งเห็นได้จาก ค่าเฉลี่ยของความคลาดเคลื่อนสัมบูรณ์ และ ค่าเฉลี่ยของเปอร์เซ็นต์ความคลาดเคลื่อนสัมบูรณ์ ที่ได้มาจากผลลัพธ์นั้นค่อนข้างน้อย ซึ่งแสดงได้ว่าการแบ่งข้อมูลนั้น ๆ มีความถูกต้องสูง และจากการศึกษาเกี่ยวกับประสิทธิภาพของหลักการทั้งสองแล้วนั้น ยังได้ทราบว่าหลักการของลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีนนี้ มีความสามารถในการจัดประเภทข้อมูลได้หลายประเภทหรือที่เรียกว่า Multi-Class Classification อีกด้วย ซึ่งเหมาะที่จะนำไปใช้ในการแยกประเภทกลุ่มข้อมูลที่มีอยู่จริงในชีวิตประจำวันเป็นอย่างมาก

โดยที่ผู้วิจัย ได้เริ่มต้นจากการเตรียมข้อมูลที่ได้มาจากสถาบันการเงินของประเทศไทย จากนั้นจึงนำข้อมูลมาทำการแปลงรูปแบบ เพื่อให้สามารถนำข้อมูลมาใช้งานร่วมกับโปรแกรม

Matlab ได้อย่างครบถ้วน นอกจากนั้นยังมีการกำหนดสัดส่วนขอบเขตของค่าข้อมูลเพื่อให้มีความง่ายต่อการคำนวณมากขึ้น เพราะเนื่องจากเมื่อได้ทำการแปลงข้อมูลแล้วทำให้ได้รับค่าข้อมูลที่มีขอบเขตกว้างและมีตัวเลขที่มีจำนวนยากต่อการคำนวณ จากนั้นจะนำข้อมูลที่แปลงรูปแล้วมาเข้าสู่กระบวนการของคลอส วาริเดชัน (Cross Validation) และกริดเสิร์จ (Grid Search) เพื่อทำให้ได้ค่าตัวแปรหลักที่สำคัญ ๆ อย่างถูกต้องที่สุด และทำให้สามารถผลิตแบบจำลองโดยนำหลักการของเคอร์เน็ล (Kernel) แบบอาร์บีเอฟ (RBF) ซึ่งย่อมาจากเรเดียล เบสิส ฟังก์ชัน (Radial Basis Function) มาใช้ร่วมด้วย ทำให้สามารถสร้างแบบจำลองที่มีคุณภาพได้

หลังจากที่ได้แบบจำลองที่มีคุณภาพแล้ว จึงนำข้อมูลในส่วนของการทดสอบเข้ามาใช้ในการแบ่งแยกประเภทจากแบบจำลองที่หาได้ ซึ่งแบบจำลองที่ได้มาจากหลักการของซัพพอร์ตเวกเตอร์แมชชีนนั้น สามารถที่จะแบ่งประเภทของข้อมูลได้ออกเป็นสองประเภทคือ ดี และ ไม่ดี โดยที่ในแต่ละจำนวนของข้อมูลในส่วนของการฝึกฝนที่แตกต่างกัน ซึ่งในวิทยาพนธ์นี้ได้ทำการเปรียบเทียบจำนวนของข้อมูลในส่วนของการฝึกฝนระหว่าง 600 700 และ 800 ข้อมูลตัวอย่าง โดยให้จำนวนข้อมูลในส่วนของการทดสอบเท่ากันคือ 200 ข้อมูลตัวอย่าง ซึ่งอัตราส่วนที่แตกต่างกันนี้ สามารถให้ผลลัพท์ที่แตกต่างกันอีกด้วย โดยที่ยิ่งมีจำนวนข้อมูลในการฝึกฝนข้อมูลมากขึ้นจะส่งผลให้การจัดแบ่งประเภทข้อมูลในส่วนของการทดสอบมีค่าความผิดพลาดน้อยลงตามลำดับ

สำหรับแบบจำลองที่ได้มาจากหลักการของ ลีซท์ สแคว ซัพพอร์ตเวกเตอร์แมชชีน นั้น สามารถแบ่งประเภทของข้อมูลได้ออกเป็นสี่ประเภทคือ ดี ค่อนข้างดี มีความเสี่ยง และ ไม่ดี โดยที่จำนวนของข้อมูลในส่วนของการฝึกฝน มีผลต่อความถูกต้องของการจัดแบ่งประเภทข้อมูลในส่วนของการทดสอบเช่นเดียวกันกับหลักการของซัพพอร์ตเวกเตอร์แมชชีน

วิธีการที่นำเสนอในวิทยานิพนธ์นี้ เป็นเพียงเทคนิคหนึ่งเท่านั้นที่ช่วยในการแบ่งแยกประเภทของข้อมูลให้นำมาใช้ในความเป็นจริงได้ แต่ก็ยังมีเทคนิควิธีการอื่นที่น่าสนใจ และสามารถที่ทำให้แบบจำลองมีสมรรถนะที่ดี ไม่ว่าจะเป็นเทคนิคของความถดถอย (Regression) ก็ดี อัลกอริทึมข่ายงานความเชื่อเบย์ (Bayesian Belief Network) ก็ดี หรือทฤษฎีการเรียนรู้สถิติ เป็นต้น ซึ่งจะขึ้นอยู่กับความถนัดของผู้ใช้งานนั่นเอง โดยที่ปัจจุบันความต้องการในการขอสินเชื่อกับทางสถาบันการเงินมีมากขึ้นเป็นลำดับ ทำให้สถาบันการเงินต่าง ๆ นั้นค่อนข้างที่จะมีความเสี่ยงมากขึ้น แต่ด้วยที่ความเป็นธุรกิจที่มีอัตราการแข่งขันค่อนข้างสูง สถาบันการเงินเหล่านั้น จึงต้องเสาะหาวิธีที่ทำให้ง่ายต่อการตัดสินใจในการให้สินเชื่อกับบุคคลประเภทต่าง และที่สำคัญที่สุดวิธีเหล่านั้นต้องสามารถทำงานให้ได้ผลลัพธ์รวดเร็วและมีประสิทธิภาพสูงสุด

5.2 ข้อเสนอแนะ

จากการที่ผู้วิจัยได้นำเสนอเทคนิคของซัพพอร์ตเวกเตอร์แมชชีนและ ลีซท์ สแควว ซัพพอร์ตเวกเตอร์แมชชีน สำหรับการแบ่งประเภทลูกค้าของสถาบันการเงินในประเทศไทยไปแล้วนั้น หากผู้ใดสนใจยังสามารถที่จะนำหลักการดังกล่าวไปพัฒนาต่อได้ โดยทำการศึกษาเพิ่มเติมได้อีกว่าค่าของคุณสมบัติ (Attribute) ประเภทใด มีผลต่อการแบ่งประเภทลูกค้าจากสถาบันการเงินมากที่สุด เพื่อที่จะได้เป็นแนวทางให้กับสถาบันการเงินเหล่านั้นนำหลักการนี้ ไปปรับปรุงใช้งานจริงให้ผลได้ดีมากยิ่งขึ้น นอกเหนือจากนั้นยังอาจจะติดตามบุคคลที่ทางสถาบันการเงินได้ตัดสินใจไปแล้ว ว่ามีอัตราแนวโน้มเป็นจริงตามที่แบบจำลองได้ทำการแยกประเภทไว้หรือไม่ เพื่อที่จะได้ยืนยันได้อีกว่าหลักการเหล่านี้มีประสิทธิภาพจริง

บรรณานุกรม

[1] ณัฏฐพันธ์ เขจรนันท์ และคณะ, 2545, **"การวิเคราะห์เชิงปริมาณทางธุรกิจ,"** E-learning, มหาวิทยาลัยแม่ฟ้าหลวง:ธรรกมลการพิมพ์

[2] Su-Lin Pang, Yan-Ming Wang and Yuan-Huai Bai, "**Credit Scoring model based on Neural Network**," Proceeding of the First International conference on Machine Learning and cybernetics, Novermber 2002, pp. 1742-1746.

[3] Chih-Wei Hsu, Chih-Chung Chang and Chih-Jen Lin "**A comparison of methods for multi-class support vector machines,"** IEEE Transactions on Neural Networks, Vol. 13, March 2002, pp.415-425.

[4] Xiang Tian and Feiqi Deng, "**A Credit Scoring Model Using Support Vector Machine**," Intelligent Control and Automation, WCICA. Fifth World Congress, Vol. 3, June 2004, pp.1945 – 1949.

[5] Gavin Cawley, **"MATLAB Support Vector Machine Toolbox,"** University of East Anglia, School of Information Systems, September 2000.

[6] Cristianini Nello and Shawe-Taylor John, , "**An Introduction to Support Vector Machines**," Cambridge University Press, March 2000.

[7] Chih-Wei Hsu, Chih-Chung Chang and Chih-Jen Lin, "**A practical Guide to Support Vector Classification**," Technical report, Department of Computer Science, National Taiwan University, 2003.

[8] Yuh-Jye Lee and Olvi L. Mangasarian, "**RSVM: Reduced Support Vector Machines**," Proceedings of the First SIAM International Conference on Data Mining, Chicago, April 2001, pp. 1-17.

[9] Bernhard Scholkopf, Christopher J C Burges and Alexander J Smola, **"Advances in Kernel Methods – Support Vector Learnng**," MIT Press, 1998.

[10] Alexander J Smola**, "Learning with Kernels**," PhD Thesis, published by GMD, Birlinghoven, 1999.

[11] Van Gestel Tony, Baesens Bart, Suykens Johan, Espinoza Marcelo, Baestaens Dirk-Emma, Vanthienen Jan, De Moor Bart, "**Bankruptcy Prediction with Least Squares Support Vector Machine Classifiers**," Computational Intelligence for Financial Engineering, IEEE International Conference, March 2003, pp.1-8.

[12] Suykens Johan AK. and Vandewalle Joos, "**Least squares support vector machine classifier**," Neural Processing letters, 1999, pp. 293-300

[13] Van Gestel Tony, Suykens Johan, Baesens Bart., Viaene Stijn, Vanthienen Jan, Dedene Guido., De Moor Bart, Vandewalle Joos, "**Benchmarking least Squares Support Vector Machine Classifiers**," Machine Learning, Vol. 54, January 2004, pp.4-32.

[14] Suykens Johan AK., Van Gestel Tony, De Brabanter Jos, De Moor Bart, and Vandewalle Joos, "**Least Squares Support Vector Machines**," World Scientific, 2002.

[15] Suykens Johan AK. and Joos vandewalle, "**Multiclass Least Square Support Vector Machines**," International Joint Conference on Neural Networks (IJCNN '99), Washington DC, July 1999, pp. 900-903

[16] Suykens Johan AK. and Vandewalle Joose, De Moor Bart, "**Optimal Control by Least Squares Support Vector Machines,**" Neural Networks, Vol.14, January 2002, pp.23-35.

[17] Chih-Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin, "**A comparison of methods for multi-class support vector machines**," IEEE Transactions on Neural Networks, Vol. 13, March 2002, pp.415-425.

[18] Suykens Johan AK and Vandewalle Joos, "**Recurrent least squares support vector machines,**" IEEE Transactions on Circuits and Systems, Vol. 47, July 2000, pp.1109-1114.

[19] Van Gestel Tony, Suykens Johan AK, Baestaens Dirk-Emma, Lambrechts Annemie, Lanckriet Gert, Vandaele Bruno, De Illoor B., and Vandewalle Joos. "**Predicting financial time series using least squares support vector machines within the evidence framework,**" *IEEE Transactions on Neural Networks (Special Issue on Financial Engineering),* Vol. 12, July 2001, pp.809-821.

[20] รศ.ดร. มนัส สังวรศิลป์ และ วรรัตน์ ภัทรอมรกุล, 2543, "**คู่มือโปรแกรม Matlab ฉบับสมบูรณ์**," กรุงเทพฯ: อินโฟเพลส

[21] Andrew W. Moore, "**Cross-validation for detecting and preventing overfitting,**" School of Computer Science, Carnegie Mellon University, October 2001.

[22] Phillip H. Sherrod, "**DTREG Predictive Modeling Software**," www.dtreg.com, Copyright 2003-2007.

[23] Yang Haiqin, **"Margin Variations in Support Vector Regression for the Stock market Prediction,"** Masters thesis, The Chinese University of Hong Kong, June 2003.

[24] Kecman Vojislav, **"Support Vector Machines Basics,"** School of Engineering Report 616, The University of Auckland, School of Engineering. April 2004.

[25] Campbell Colin, **"An Introduction to Kernel Methods,"** Radial Basis Function Networks: Design and Applications, 2000, pp.31.

ภาคผนวก

ภาคผนวก ก.

ผลงานวิจัยที่ได้รับการตีพิมพ์เผยแพร่

[1] **Usanee Worrachartdatchai** and Pitikhate Sooraksa, **"Credit Scoring using Least Squares Support Vector Machine based on data of Thai Financial Institutions**," The 9th International Conference on Advanced Communication Technology 2007 (ICACT), pp. 2067-2070, Phoenix Park, Korea, April 12-14, 2007.

THE 9th INTERNATIONAL CONFERENCE on ADVANCED COMMUNICATION TECHNOLOGY

Toward Network Innovation beyond Evolution

icact korea 2007

Phoenix Park, Korea
Feb. 12–14, 2007

PROCEEDINGS
Volume III

IEEE Catalog Number 07EX1671
ISSN 1738-9445
ISBN 978-89-5519-131-8 93560

Host

Ministry of Information and Communication

Organizers

ETRI NIA

Sponsors

IEEE COMMUNICATIONS SOCIETY IEEK ComSoc ITA GANGWON

Credit Scoring using Least Squares Support Vector Machine based on data of Thai Financial Institutions

Usanee Worrachartdatchai, Assoc. Prof. Dr.Pitikhate Sooraksa
Department of Information Engineering, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand
usanee1263@hotmail.com, kspitikh@kmitl.co.th

***Abstract* — The quantitative method known as credit scoring has been developed for the credit assessment problem. Credit scoring is essentially an application of classification techniques, which classify credit customers into different risk groups. The Financial institutions are being more and more obliged to build credit scoring models assessing the risk of default of their clients. Support Vector Machine is a promising new technique that has recently emanated and become popular for data classification. Least Squares Support Vector Machines (LS-SVM) are re-formulations to the standard SVMs. The cost function is a regularized least squares function with equality constraints. The solution can be found efficiently by iterative method like the conjugate Gradient algorithm. Then in this paper, Least Squares Support Vector Machine is considered by approaching to the credit scoring with the data of Thai financial institutions. The optimum model will be able to divide the group of customers into four groups: very good, rather good, suspiciously bad and very bad with high accuracy.**

***Keywords* — credit scoring, support vector machine, least square support vector machine, multiclass, neural network**

1. Introduction

There are many techniques that used for build the credit scoring model assessing the risk of default of their customers. The Support vector machine is the one that the most researchers interested. The support vector machine [1, 2, 3, 4] is a primarily a classier method that perform classification tasks by constructing hyperplanes in a multidimensional space that separates cases of different class labels. While standard SVM solutions involve solving quadratic or linear programming problems, the least squares version of SVM's corresponds to solving a set of linear equations, due to equality instead of inequality constraints in the problem formulation. The Least Square Support Vector Machine (LS-SVM) [5, 6, 7, 8, 9, 10, 11] are re-formulations to the standard SVMs. In this LS-SVM version one finds the solution by solving a linear system instead of quadratic programming. This is due to the use of equality instead of inequality constraints in the problem formulation. The LS-SVM solutions are obtained from quadratic programming problems possessing a global solution. The kernel function [12, 13, 14] is a weighting function used in nonparametric function estimation. It gives the weights of the nearby data points in making an estimate. In practice kernel functions are piecewise continuous, bounded, symmetric around zero, concave at zero, real valued, and for convenience often integrate to one. They can be probability density functions. They often have a bounded domain like [-1, 1]. The kernel functions and parameters can be chosen such that a bound on the VC dimension is minimized. There are number of kernels that can be used in LS-SVM models. These include polynomials, splines, radial basis function (RBF), Sigmod, Linear and multilayer perceptrons.

In this research, we will use the data instances from Thai financial institutions to be trained with the LS-SVM solution and provide the optimum model that is able to divide the group of customer into four groups. The LS-SVM to the multiclass case [15, 16] that we discuss here is related to classical neural network approaches [17, 18] for classification where multi classes are encoded by considering multiple outputs for the network.

2. Methodology

In order to do classification, we normally involve with the training and testing data which consist of some data instances. In the training set, each data instance will contain one "target value" that we called "class labels" and also contain many "attributes" that we called "features". But in the testing set, we will only have the attributes and we will produce an optimum model which is able to predict the accurate target value.

In this experiment, the class labels will be the groups of Thai financial institute's customers (very good, rather good, suspiciously bad and very bad). The features will be the criteria that they normally use to classify their customers. Here, there are given a training set of instance-label pairs (x_i, y_i), where $x_i \in R^n$, $y_i \in \{1,-1\}^l$ and $i = 1,\dots l$

LS-SVM requires the solution of the following optimization problem (1).

$$\min_{w_i, b_i, e_{k,i}} J_{LS}^{(m)}(w_i, b_i, e_{k,i}) = \frac{1}{2}\sum_{i=1}^{m} w_i^T w_i + C\frac{1}{2}\sum_{k=1}^{N}\sum_{i=1}^{m} e_{k,i}^2 \quad (1)$$

Subject to the equality constraints,

$$y_k^{(1)}[w_1^T \alpha_1(x_k) + b_1] = 1 - e_{k,1}, k = 1,\dots,N$$

$$y_k^{(2)}[w_2^T \alpha_2(x_k) + b_2] = 1 - e_{k,2}, k = 1,\dots,N$$

.....

$$y_k^{(m)}[w_m^T \alpha_m(x_k) + b_m] = 1 - e_{k,m}, k = 1,\dots,N$$

In order to achieve this, here training vectors x_i are mapped into a higher dimensional feature space which can be infinite dimensional by the function ϕ. Then LS-SVM finds a linear separating hyperplane with the maximal margin in this higher dimensional space. C > 0 is the penalty parameter of the error terms. Furthermore, (2) is called the kernel function.

$$K(x_i, y_i) \equiv \Phi(x_i)^T \Phi(x_j) \tag{2}$$

3. Experiment

A. Data Preparation

LS-SVM requires that each data instance is represented as a vector of real numbers. Hence, if there are categorical attributes, we first have to convert them into numeric data. In this experiment, we used 1000 data instances and split up into two sets: one used to train and design the system, generating the model parameters, another one used to test the goodness of fit of the model. Both data sets have been spited with the three ratios as show in Table 1.

Table 1. Data set

Data set	Training	Testing
1	600	200
2	700	200
3	800	200

We will compare model that make from the ratio above and find out the optimum model that will be able to divide the group of customers into four groups: very good, rather good, suspiciously bad and very bad with high accuracy.

B. Training Data

1. Kernel

In this experiment, we have chosen the RBF because it's nonlinearly maps samples into a higher dimensional space. As the number of hyper parameters which influences the complexity of model selection. The polynomial kernel has more hyper parameters than the RBF kernel.

Finally, the RBF kernel has less numerical difficulties. One key point is $0 < K_{ij} \leq 1$ in contrast to polynomial kernels of which kernel values may go to infinity (3) or zero (4) while the degree is large. Equation (5) is the RBF kernel.

$$\gamma x_i^T x_j + r > 1 \tag{3}$$

$$\gamma x_i^T x_j + r < 1 \tag{4}$$

$$K(x_i, y_i) \equiv \exp(-\gamma \|x_i - x_j\|^2), \gamma > 0 \tag{5}$$

2. Cross-validation and Grid-search

While we using RBF kernels, there are two parameters: C and γ which we have to seek the best values for that case and we will use it to be model of the classifier, so that classifier can accurately predict unknown data (i.e., testing data). Therefore, some kind of model selection (parameter search) must be done.

We have used the cross validation via parallel grid search that currently support only two parameters are considered: C and γ. Therefore, a common way of cross validation is to separate training data to two parts of which one is considered unknown in training the classifier. Then the prediction accuracy on this set can be more precisely by using grid-search, reflect to the performance on classifying unknown data.

An improved version of this procedure is cross-validation. In v-fold cross-validation, we first divide the training set into v subsets of equal size. Sequentially one subset is tested using the classifier trained on the remaining v − 1 subsets. Thus, each instance of the whole training set is predicted once so the cross-validation accuracy is the percentage of data which are correctly classified. The cross-validation procedure can prevent the over fitting problem.

We have used a "grid-search" on C and using cross-validation. Basically pairs of (C, γ) are tried and the one with the best cross-validation accuracy is picked. We found that trying exponentially growing sequences of C and γ is a practical method to identify the good parameters, for example (6), (7).

$$C = 2^{-5}, 2^{-3}, \ldots, 2^{5} \tag{6}$$

$$\gamma = 2^{-10}, 2^{-13}, \ldots, 2^{10} \tag{7}$$

There are two motivations to choose this simple grid-search approaching to the classifier model. One is that psychologically we may not feel safe to use methods which avoid doing an exhaustive parameter search by approximations or heuristics. The other reason is that the computational time to find good parameters by grid-search is not much more than that by advanced methods since there is only two parameters. Furthermore, the grid-search can be easily parallelized because each (C, γ) is independent. In fact, there are several advanced methods which can save computational cost by, for example, approximating the cross-validation rate. Many of the advanced methods are iterative processes, e.g. walking along a path, which might be difficult for parallelization.

Since doing a complete grid-search may still be time-consuming, we recommend using a coarse grid first. After identifying a "better" region on the grid, a finer grid search on that region can be conducted. After scaling this set, we first use a coarse grid as show in Figure 1. and find that the best (C, γ) is (2^4, 2^{-4}) with the cross-validation rate 78.5%. After that, we used the value of these two parameters from loose grid search to be a model and process for the classification and we got the graph of Actual data and estimation as showed in the Figure 2. And we found that the result wasn't good enough, then we conduct a finer grid search on the neighborhood of (2^4, 2^{-4}) as show in Figure 3. and obtain a better cross-validation rate 78.9% at ($2^{4.25}$, $2^{-4.83}$).

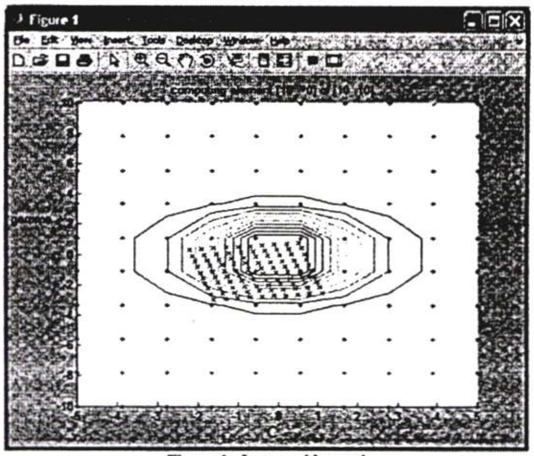


Figure 1. Loose grid search

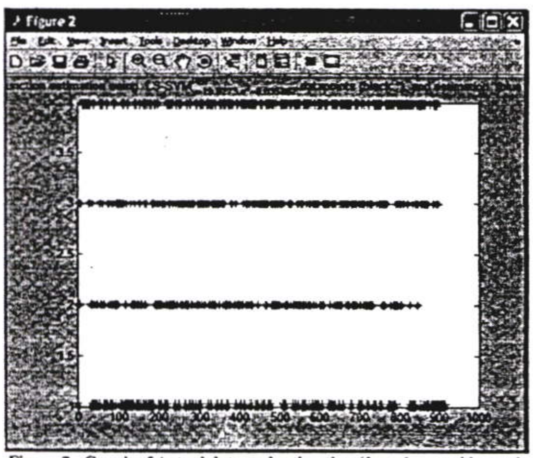


Figure 2. Graph of Actual data and estimation (from loose grid search model)

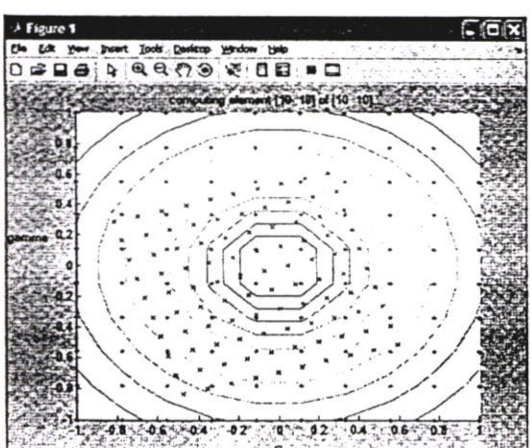


Figure 3. Fine grid-search

After the best (C, γ) is found, the whole training set is trained again to generate the final classifier. The above approach works well for problems with thousands or more data points. For very large data sets, a feasible approach is to randomly choose a subset of the data set, conduct grid-search on them, and then do a better-region-only grid-search on the complete data set.

3. Testing Data & Result

After we have done the training data, we would get the optimum model that came from the best value of (C , γ), so we will do the same thing as we did training but we will change the data instance to be the one that we prepared for the testing only. Then we will get the confusion matrix as show in Table 2 that represents the number of correct rating classification by the optimum model, values to the right of the diagonal imply the number of lower rating by the model compared to the actual rating, left from the diagonal are higher classifications by the model. The numbers 1-4 are represented as 1 = Very good, 2 = Rather good, 3 = suspiciously bad, 4 = Very bad.

Table 2. Confusion Matrix.

Confusion Matrix		Predicted			
		1	2	3	4
Actual	1	1.000	0	0	0
	2	0	0.9300	0.0200	0.0500
	3	0.0400	0	0.9300	0.0300
	4	0.0200	0.0700	0.0300	0.8800

Since we got the confusion matrix, we will be able to generate the graph comparison between Actual data and Predicted data from the best model that we got from the training data as show in Figure 4. The red labels are the predicted values that were generated from the model and the blue labels are the real data.

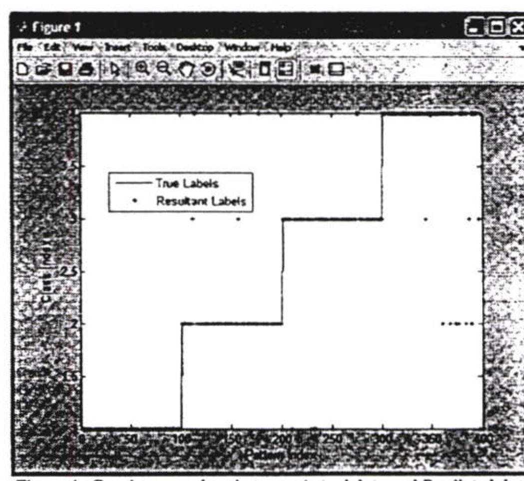


Figure 4. Graph comparison between Actual data and Predicted data from the best model

4. Evaluation

The main purpose of the underlying rating based system was to make accurate decisions for each type of rating category. For the evaluation of the rating models, we used *RMSE* and *MAPE*. As we can see the result that showed in Table 3, number of data training, 600,700, and 800. MAE and MAPE as 1.2/1.9%, 1.11/1.7%, 1.05/1.1% respectively. If we have the number of data in Training set more, the accurate of the result will be increased.

The Mean Absolute Error:

$$MAE = \frac{1}{T}\sum_{t=1}^{T}\left| y_t - \hat{y}_t \right| \quad (7)$$

The Mean Absolute Percentage Error:

$$MAPE = \frac{1}{T}\sum_{t=1}^{T}\left(\left|\frac{y_t - \hat{y}_t}{y_t}\right| \times 100\right) \quad (8)$$

Table 3. The Accuracy Comparison From The Number Of Data In Training And Testing.

Data Set	Number of data: Training	Number of data: Testing	*MAE*	*MAPE*
All data: 1000 and Number of Attribute : 20	600	200	1.2	1.9%
	700	200	1.11	1.7%
	800	200	1.05	1.1%

5. Conclusion

The researcher establishes a credit scoring by using least square support vector machine with the RBF kernel. This classifier is a computationally simple but powerful nonlinear classifier to predict financial data in Thailand. It is used to divide the group of financial institution's customer in Thailand into four groups: very good, rather good, suspiciously bad and very bad. The experiment results show that least square support vector machine can build the optimum model which has high classification accuracy rate and strong suitable ability.

REFERENCES

[1] Cristianini N., Shawe-Taylor J., An Introduction to Support Vector Machines, Cambrdge University Press, 2000.

[2] *Xiang Tian, Feiqi Deng*, "A Credit Scoring Model Using Support Vector Machine," *Intelligent Control and Automation. 2004*. WCICA 2004. Fifth World Congress on ,Volume: 3 , June 2004, pp.1940 – 1944

[3] Chih-Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin, " A practical Guide to Support Vector Classification," Technical report, Department of Computer Science, National Taiwan University, 2003.

[4] Yuh-Jye Lee and Olvi L. Mangasarian, "RSVM: Reduced Support Vector Machines," Proceedings of the First SIAM International Conference on Data Mining, Chicago, April 5-7. 2001.

[5] Suykens J.A.K. Vandewalle J., "Least squares support vector machine classifier," Neural Processing letters, 9(3), 293-300, 1999.

[6] Van Gestel T., Suykens J., Baesens B., Viaene S., Vanthienen J., Dedene G., De Moor B., Vandewalle J., "Benchmarking least Squares Support Vector Machine Classifiers," machine Learning, in press.

[7] Van Gestel. T, Baesens, B, Suykens, J, Espinoza, M, Baestaens, D.-E, Vanthienen, J, De Moor B, "Bankruptcy Prediction with Least Squares Support Vector Machine Classifiers," Computational Intelligence for Financial Engineering, 2003. Proceedings. 2003 IEEE International Conference on 20-23 March 2003, pp. 1-8.

[8] Suykens, J.A.K., Van Gestel, T., De Brabanter, J., De Moor, B., and Vandewalle J.," *Least Squares Support Vector Machines*," World Scientific, in press, 2002.

[9] J.A.K. Suykens and J. vandewalle, "Multiclass Least Square Support Vector Machines," International Joint Conference on Neural Networks (IJCNN '99), Washington DC, July 1999, pp. 900-903

[10] Suykens J.A.K, Vandewalle J, De Moor B., "Optimal Control by Least Squares Support Vector Machines, Neural Networks," 2001, Vol.14(1), pp.23-35.

[11] Suykens J.A.K Gestel V.T., De Brabanter J., De Moor B., Vandewalle J., "Least Squares Support Vector Machines," World Scientific, 2002.

[12] Scholkopf B., Burges C., Smola A. (Eds.), Advances in Kernel Methods – Support Vector Learnng, MIT Press, 1998.

[13] Scholkopf B., Burges C., Smola A. (Eds.), Advances in Kernel Methods – Support Vector Learnng, MIT Press, 1998.

[14] Smola A., Learning with Kernels, PhD Thesis, published by GMD, Birlinghoven, 1999.

[15] Chih-Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin, "A comparison of methods for multi-class support vector machines," IEEE Transactions on Neural Networks, 2002, pp.415-425.

[16] J.A.K. Suykens and J. vandewalle, "Multiclass Least Square Support Vector Machines," International Joint Conference on Neural Networks (IJCNN '99), Washington DC, July 1999, pp. 900-903

[17] Van Gestel, T., Suykens, J.A.K., Baestaens, ID.-E. Lambrechts, A., Lanckriet, G., Vandaele, B., De Illoor, B., and Vandewalle, J. "Predicting financial time series using least squares support vector machines within the evidence framework," *IEEE Transactions on Neural Networks (Special Issue on Financial Engineering)*, 12–809-821,2001.

[18] Su-Lin Pang, Yan-Ming Wang, Yuan-Huai Bai, "Credit Scoring model based on Neural Network," Proceeding of the First International conference on Machine Learning and cybernetics on 4-5 Novermber 2002, pp. 1742-1746

[19] "Matlab 6," Users Guide, MathWork Inc.

ประวัติผู้เขียน

นางสาวอุษณี วรชาติเดชชัย เกิดเมื่อวันที่ 12 มิถุนายน พ.ศ. 2523 ที่จังหวัดกรุงเทพมหานคร สำเร็จการศึกษาปริญญาตรีวิทยาศาสตรบัณฑิต สาขาเทคโนโลยีสารสนเทศ จากภาควิชาเทคโนโลยีสารสนเทศสถาบันเทคโนโลยีนานาชาติสิรินทร มหาวิทยาลัยธรรมศาสตร์ ในปีการศึกษา 2544 และเข้าศึกษาต่อในระดับปริญญาโท หลักสูตรวิศวกรรมศาสตรมหาบัณฑิต สาขาวิศวกรรมสารสนเทศ ภาควิชาวิศวกรรมสารสนเทศ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ในปีการศึกษา 2547 โดยในปี พ.ศ. 2545 ได้เข้าทำงานในตำแหน่งเจ้าหน้าที่สารสนเทศ บริษัทแอ๊ดวานซ์ อินโฟร์ เซอร์วิส จำกัด มหาชน เป็นเวลา 1 ปี จากนั้นได้เดินทางไปศึกษาต่อในด้านภาษาอังกฤษที่ประเทศนิวซีแลนด์ เป็นเวลา 6 เดือน หลังจากเดินทางกลับมายังประเทศไทย ได้เริ่มศึกษาต่อในระดับปริญญาโท พร้อมทั้งได้เข้าทำงานในตำแหน่งนักดูแลระบบความปลอดภัยเครือข่าย บริษัทคิมเบอร์ลี่ คล๊าก จำกัด อีกด้วย รวมเป็นเวลา 2 ปีครึ่ง จนปัจจุบันได้ย้ายที่ทำงานมายังบริษัทเอ็กซอนโมบิล จำกัด ในตำแหน่งนักวิเคราะห์ระบบ SAP และเป็นอาจารย์พิเศษสอนวิชาคอมพิวเตอร์และเทคโนโลยีสารสนเทศให้กับมหาวิทยาลัยกรุงเทพ โดยตลอดมาได้ทำงานเต็มเวลาพร้อมทั้งกับศึกษาปริญญาโทควบคู่มาจนถึงปัจจุบัน