การพัฒนาเครื่องมือวิเคราะห์เพื่อสนับสนุนการอนุมัติสินเชื่อ โดยวิธี Classification Tree

Analytical tool Development for Credit Approval by using Classification Tree

โดย

นางสาวกนกวรรณ จันทรสถาพรจิต

รหัส 43067072

อาจารย์ที่ปรึกษา

ผศ. ดร. วรพจน์ กรีสุระเดช

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
ภาคเรียนที่ 2 ปีการศึกษา 2545
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ชื่อหัวข้อ	การพัฒนาเครื่องมือวิเคราะห์เพื่อสนับสนุนการอนุมัติสินเชื่อ โดยวิธี Classification Tree
นักศึกษา	นางสาวกนกวรรรณ จันทรสถาพรจิต
อาจารย์ที่ปรึกษา	ผศ.ดร. วรพจน์ กรีสุระเดช
ระดับการศึกษา	วิทยาศาสตรมหาบัณฑิต สาขาเทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2545

บทคัดย่อ

ในโลกของธุรกิจที่มีการแข่งขันกันอย่างสูง บริษัทต่างๆ พยายามแสวงหาเทคนิคและวิธีการที่จะทำให้ประสบความสำเร็จเหนือคู่แข่ง Data Mining เป็นอีกเทคนิคหนึ่งที่น่าสนใจและถูกนำมาใช้งานอย่างกว้างขวางในปัจจุบัน เพื่อทำให้สามารถนำข้อมูลที่มีอยู่ไปใช้งานได้อย่างมีประสิทธิภาพและเกิดประโยชน์ต่อธุรกิจมากที่สุด

โครงการนี้ ได้ทำการศึกษาและพัฒนาเครื่องมือ เพื่อใช้ช่วยในการวิเคราะห์และประเมินการอนุมัติสินเชื่อ โดยนำข้อมูลต่างๆ ของลูกค้ามาทำ Data Mining ด้วยวิธี Classification Tree เพื่อหารูปแบบความสัมพันธ์ของข้อมูล และนำผลลัพธ์ที่ได้มาใช้ในการประเมินลูกค้า เพื่อลดความเสี่ยง และทำให้การพิจารณาอนุมัติสินเชื่อเป็นไปอย่างมีประสิทธิภาพมากขึ้น

Title	Analytical tool Development for Credit Approval by using Classification Tree
Student	Ms. Kanokwan Jantarasathpornjit
Advisor	Asst.Prof. Dr. Worapoj Kreesuradej
Level of Study	Information of Science in Information Technology
Major	Information Science
Academic Year	2002

ABSTRACT

In the high competitive world, many companies are trying to look for techniques and approaches to overcome their competitors. Data mining is one of interesting technique that is widely used at present in order to take advantages of data usage effectively in business world.

This project will study and develop analytical tool for credit approval through the use of data mining with classification tree methodology in order to find relationship within the data. The result of such relationship will be used to evaluate customer's credit effectively and reduce the risk of bad credit approval.

กิตติกรรมประกาศ

ในการจัดทำโครงการการพัฒนาระบบงานนี้ ข้าพเจ้าขอขอบพระคุณท่าน ผศ.ดร.วรพจน์ กรีสุระเดช อาจารย์ที่ปรึกษา ที่ได้กรุณาให้คำปรึกษาและแนะนำแนวทางในการแก้ไขปัญหาต่าง ทำให้โครงงานนี้สำเร็จลุล่วงมาได้ ขอขอบคุณครอบครัวของข้าพเจ้าที่คอยสนับสนุนและเป็นกำลังใจในการเรียนตลอดมา และขอขอบคุณเพื่อนๆ ทุกคนที่มีส่วนให้ความช่วยเหลือ ให้คำแนะนำ ตลอดจนคอยเป็นกำลังใจ และสนับสนุนให้ผลงานนี้สำเร็จลุล่วงด้วยดี

คนกวรรณ จันทรสถาพรจิต

กุมภาพันธ์ 2546

สารบัญ

หน้า

บทที่ 1
บทนำ

1.1 หลักการและเหตุผล

ในปัจจุบันเทคโนโลยีทางคอมพิวเตอร์ได้ก้าวหน้าไปอย่างรวดเร็ว รวมทั้งยังเข้ามามีบทบาทในชีวิตประจำวัน และเป็นส่วนสำคัญอย่างหนึ่งในการดำเนินกิจกรรมต่างๆ โดยเฉพาะอย่างยิ่งการดำเนินกิจกรรมทางธุรกิจ ที่ต้องมีการแข่งขันกันในทุกๆ ด้านเพื่อความได้เปรียบเหนือคู่แข่ง การตัดสินใจที่ถูกต้องและรวดเร็วจึงมีความสำคัญ การตัดสินใจที่ล่าช้าหรือการที่ไม่สามารถแก้ไขปัญหาได้ทันท่วงทีอาจทำให้เกิดผลเสียหายตามมา โดยเฉพาะอย่างยิ่งอาจทำให้สูญเสียโอกาสในการแข่งขัน ซึ่งการที่จะทำให้เกิดความได้เปรียบในเชิงธุรกิจได้นั้น ส่วนหนึ่งจะต้องได้มาซึ่งระบบสารสนเทศที่สามารถให้ข้อมูลที่ถูกต้องและทำให้ตัดสินใจหรือดำเนินการต่างๆ ได้ทันเวลา และเพื่อประโยชน์ในการกำหนดกลยุทธ์การวางแผนที่มีประสิทธิภาพ จึงมีการค้นคว้าเทคนิคและวิธีการใหม่ๆ ขึ้นมา ซึ่ง Data Mining เป็นอีกเทคนิคหนึ่งที่มีความสามารถสูง และเหมาะสำหรับนำมาใช้ในการช่วยวิเคราะห์ข้อมูลในเชิงธุรกิจ เพื่อหารูปแบบความสัมพันธ์ของข้อมูลที่มีอยู่ในฐานข้อมูลขนาดใหญ่ และนำผลลัพธ์ที่ได้จากการวิเคราะห์นั้นไปประยุกต์ใช้ในการพัฒนาและปรับปรุงรูปแบบกิจกรรมทางธุรกิจ รวมถึงขั้นตอนการดำเนินงานให้เป็นไปอย่างมีประสิทธิภาพและเกิดประโยชน์สูงสุด

1.2 วัตถุประสงค์

1. เพื่อศึกษาขั้นตอน วิธีการทำ Data Mining โดยการใช้ Classification Tree
2. นำความรู้ Data Mining และ Algorithm ที่ศึกษา มาพัฒนาเครื่องมือวิเคราะห์เพื่อสนับสนุนการอนุมัติสินเชื่อ โดยใช้ Classification Tree
3. เพื่อนำข้อมูล หรือผลลัพธ์ที่ได้จากการทำ Data Mining ไปใช้ในการสนับสนุนการตัดสินใจอนุมัติสินเชื่อ เพื่อลดความเสี่ยงขององค์กรในการให้สินเชื่อ

1.3 ขอบเขตการดำเนินงาน

โครงการพัฒนาระบบงานนี้ จะทำการพัฒนาระบบที่สามารถช่วยวิเคราะห์ข้อมูลลูกค้าที่มาขอสินเชื่อว่ามีความน่าเชื่อถือมากน้อยเพียงใด และมีความเหมาะสมที่จะได้รับการอนุมัติสินเชื่อหรือไม่ โดยจะนำเสนอผลลัพธ์ที่ได้ในรูปแบบของ Decision Tree

1.4 ขั้นตอนการดำเนินงาน

1. กำหนดวัตถุประสงค์และเป้าหมายในการดำเนินงาน
2. ทำการศึกษาขั้นตอนการทำ Data Mining โดยใช้ Classification Tree
3. ศึกษา SLIQ algorithm และการนำมาประยุกต์ใช้กับระบบงาน
4. เตรียมข้อมูลที่จะนำมาใช้ในรูปแบบที่เหมาะสมกับ Algorithm
5. ทำการออกแบบระบบ และกำหนดขอบเขตการทำงานของระบบที่จะทำการพัฒนา
6. พัฒนาระบบโดยนำ SLIQ algorithm ที่ทำการศึกษามาทำการสร้าง Classification Tree
7. ทำการทดลองใช้งานจริง โดยนำข้อมูลที่เตรียมไว้มาทำการทดสอบกับระบบ
8. ปรับปรุงและแก้ไขข้อผิดพลาดที่เกิดขึ้น
9. สรุปผลการศึกษา

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. เข้าใจหลักการและขั้นตอนการทำ Data Mining โดยใช้ Classification Tree
2. ระบบสามารถช่วยวิเคราะห์ข้อมูลลูกค้าที่มาขอสินเชื่อได้ว่า มีความน่าเชื่อถือมากน้อยเพียงใด จึงเป็นประโยชน์ต่อธุรกิจทางด้านนี้ และมีผลต่อการลดความเสี่ยงขององค์กร
3. เพิ่มทักษะในการเขียนและพัฒนาโปรแกรม
4. เป็นแนวทางและตัวอย่างในการประยุกต์ใช้ Data Mining กับงานด้านธุรกิจ

บทที่ 2

ทฤษฎีของ Data Mining

ในปัจจุบัน Data Mining และ Data Warehouse ได้รับความนิยมอย่างกว้างขวางในธุรกิจต่างๆ เช่นเดียวกันเราก็คงเคยได้ทราบถึงแนวความคิดของ Data Mining จากหนังสือ บทความ รวมถึงงานวิจัยด้านฐานข้อมูล Data Mining เกิดจากแนวความคิดของหลักการทางสถิติ การปฏิวัติแนวความคิดใหม่ของฐานข้อมูล และการเรียนรู้ของเครื่องจักรกล(Machine Learning) เนื่องจากข้อมูลในฐานข้อมูลหรือ Data Warehouse มีเป็นจำนวนมาก ยากที่จะวิเคราะห์ถึงความสัมพันธ์และแนวโน้มต่างๆ ได้อย่างครบถ้วน จึงได้มีการศึกษาค้นคว้าเทคโนโลยีใหม่ขึ้น คือ Data Mining มาใช้ช่วยในการวิเคราะห์ข้อมูลเพื่อให้ทราบถึงความสัมพันธ์ในรูปแบบต่างๆ ที่อยู่ในฐานข้อมูลขนาดใหญ่ได้

Data Mining เป็นเทคนิคใหม่ซึ่งมีประสิทธิภาพในการวิเคราะห์ข้อมูลต่างๆ รวมถึงยังเป็นเครื่องมือที่ดีในการพยากรณ์แนวโน้มและพฤติกรรมของข้อมูล และยังสามารถเก็บความรู้นั้นไว้เพื่อใช้ช่วยในการตัดสินใจ เพราะ Data Mining สามารถตอบคำถามทางธุรกิจ ซึ่งส่วนใหญ่เป็นปัญหาเกี่ยวกับแนวโน้มได้ดี Data Mining เป็นกระบวนการที่เป็นการนำเอาข้อมูลที่ซ่อนอยู่ภายใต้ข้อมูลซึ่งข้อมูลเหล่านี้อาจมาจากฐานข้อมูลขนาดใหญ่ ข้อมูลที่ได้เป็นข้อมูลที่ไม่ทราบมาก่อน และข้อมูลที่ได้จากการทำ Data mining เหล่านี้ก็เป็นข้อมูลที่มีประโยชน์ สามารถนำข้อมูลนี้ไปใช้ช่วยเป็นแนวทางในการตัดสินใจใดๆ ที่ก่อให้เกิดผลประโยชน์ในทางธุรกิจ ซึ่งถือว่าเป็นจุดประสงค์หลักของการทำ Data Mining

2.1 Data Mining คืออะไร

Data Mining คือ กระบวนการค้นหาสารสนเทศที่เป็นประโยชน์ โดยได้จากการหาความสัมพันธ์และรูปแบบทั่วไปของข้อมูลที่มีอยู่จากข้อมูลจำนวนมาก ซึ่งในบางครั้งจะเรียก Data Mining ว่าเป็นการค้นหาความรู้ใหม่จากฐานข้อมูล หรือ Knowledge Discovery in Database (KDD) โดยจุดประสงค์หลักของการค้นหาข้อมูลทั้งหมด ก็เพื่อสร้างรูปแบบที่สามารถเข้าใจได้ง่ายและสะดวกในการที่จะตีความพื้นฐานของข้อมูลนั้นๆ

Data Mining เป็นความสามารถของมนุษย์ร่วมกับคอมพิวเตอร์ โดยมนุษย์เป็นผู้ออกแบบฐานข้อมูล อธิบายปัญหา และกำหนดจุดมุ่งหมายต่างๆ ส่วนคอมพิวเตอร์ทำหน้าที่กลั่นกรองข้อมูลที่ผ่านเข้ามาและทำการค้นหาแบบแผนที่ตรงตามจุดมุ่งหมายที่ได้กำหนดไว้ ซึ่งเทคนิคของ Data Mining ก็คือพยายามที่จะค้นหากระบวนการ กฎเกณฑ์ที่แน่นอนและมีแบบแผนอัตโนมัติที่จะนำมาใช้ในการดึงข้อมูลที่ถูกจัดเก็บเอาไว้ในฐานข้อมูลที่มีจำนวนมากๆ นำมาใช้ให้เกิดประโยชน์ ซึ่งกระบวนการค้นหาสารสนเทศจากคลังข้อมูลนี้ต้องผ่านกระบวนการจัดเตรียมข้อมูล (Data Preparation) การค้นหาและจัดรูปแบบ (Search for Pattern) จนกระทั่งได้ข้อมูลตามต้องการก่อน เพราะแม้ว่าข้อมูลจะมีการจัดเก็บมาแล้วอย่างเป็นระบบก็ตาม แต่ถ้าขาดกระบวนการในการจัดการสารสนเทศอย่างมีประสิทธิภาพและถูกวิธีแล้ว ข้อมูลต่างๆ ที่เก็บไว้ก็จะไม่มีประโยชน์เลย

2.2 ขั้นตอนการทำงานของ Data Mining

ขั้นตอนการทำงานของ Data Mining เป็นกระบวนการของการสร้างแบบจำลอง (Model) โดยสร้างแบบจำลองของกลุ่มข้อมูลเพื่อสร้างความเข้าใจในแนวโน้ม รูปแบบ และความสัมพันธ์กันของกลุ่มข้อมูลเพื่อใช้ในการทำนายข้อมูลเหล่านั้น ซึ่งกระบวนการของ Data Mining ประกอบด้วย 5 ขั้นตอน คือ

1. กำหนดวัตถุประสงค์ในการทำ Data Mining (Objective Determination)
2. เตรียมข้อมูล (Data Preparation)
3. ทำดาต้าไมนิง (Data Mining)
4. ทำการวิเคราะห์ผลลัพธ์ที่ได้ (Analysis of Results)
5. นำสารสนเทศที่ได้ไปใช้ประโยชน์ (Assimilation of knowledge)

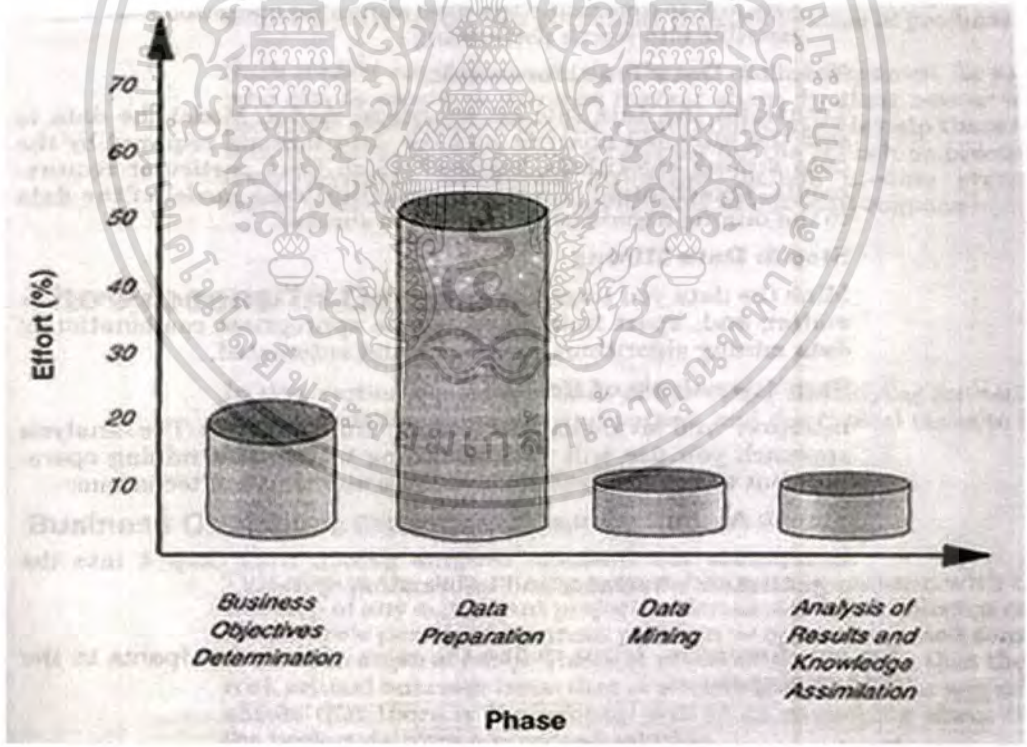
รูปที่ 2.1 กระบวนการทำ Data Mining

1. ขั้นตอนการกำหนดวัตถุประสงค์ในการทำ Data Mining

การกำหนดจุดประสงค์และปัญหาที่ชัดเจนจะเป็นตัวกำหนดทิศทางการทำ Data Mining ดังนั้น ในการกำหนดจุดประสงค์ของงานจะต้องเข้าใจถึงปัญหาและความต้องการของงานนั้นๆ รวมทั้งต้องดูถึงความเป็นไปได้ด้วยว่าวิธีการ Data Mining เหมาะกับการหาคำตอบของปัญหานั้น ๆ หรือไม่ การกำหนดถึงความต้องการนั้นจะมุ่งประเด็นถึงคำตอบที่ได้เพื่อจะนำไปใช้ให้เกิดประโยชน์ แต่จะไม่ใช่เกิดจากการตั้งสมมติฐาน และนอกจากนั้นยังเป็นการกำหนดถึงแหล่งที่มาของข้อมูลที่จะทำการ Mining อีกด้วย

2. ขั้นตอนการเตรียมข้อมูล

หากกล่าวถึง Data Mining คนโดยทั่วไปอาจคิดว่าควรจะให้ความสนใจกับขั้นตอนการ Mining หรือการค้นหาลักษณะทิศทางของข้อมูลมากที่สุด แต่ที่จริงแล้วการ Mining ข้อมูลเป็นเพียงกระบวนการหนึ่งในการทำ Data Mining เท่านั้น จากรูปที่ 2.2 ซึ่งแสดงเปอร์เซ็นต์การทำงานของแต่ละขั้นตอน



รูปที่ 2.2 แสดงเปอร์เซ็นต์การทำงานในแต่ละขั้นตอนของ Data Mining [Cabena et al. 1998]

จะเห็นว่าการเตรียมข้อมูลสำหรับการทำ Data Mining นั้น เป็นขั้นตอนที่สำคัญ และเป็นช่วงที่ใช้เวลามากที่สุด โดยปกติแล้วจะใช้เวลาประมาณ 60 เปอร์เซ็นต์ ของเวลาทั้งหมดในการเตรียมข้อมูล เนื่องมาจากอาจต้องมีการนำข้อมูลมาจากหลายแหล่ง และนำมารวมกัน เพื่อดูความสัมพันธ์ของข้อมูล ซึ่งข้อมูลที่ได้จากขั้นตอนนี้จะต้องมีความชัดเจน และถูกต้อง ดังนั้น จุดที่ต้องให้ความสำคัญคือการ Clean ข้อมูลและประเด็นของข้อมูล ส่วนการทำ Mining นั้น จริงๆ มีเพียง 10 เปอร์เซ็นต์ โดยขั้นตอนการเตรียมข้อมูลนี้จะแบ่งออกเป็น 3 ขั้นตอนย่อย ดังนี้

1.) <u>การคัดเลือกข้อมูล</u> (Data Selection)

จุดประสงค์หลักคือ การระบุลักษณะและคัดเลือกข้อมูลที่ต้องการ และนำข้อมูลที่ไม่ต้องการออกไป ซึ่งการเลือกข้อมูลนี้ก็ขึ้นอยู่กับวัตถุประสงค์ที่ได้กำหนดไว้ตั้งแต่ต้น และการเลือกข้อมูลนี้จำเป็นจะต้องเข้าใจความหมาย ทราบประเภทของข้อมูล และค่าที่สามารถเป็นไปได้ ซึ่งตัวแปรข้อมูลแบ่งได้ 2 ลักษณะ คือ

➢ *แบบ Categorical*

- Nominal : คือตัวแปรที่ลำดับของข้อมูลไม่มีความสำคัญ(ลำดับไม่มีผลกับค่า) เช่น สถานภาพ (Single, Married, Divorced)
- Ordinal : คือตัวแปรที่ลำดับของข้อมูลมีความสำคัญ(ลำดับมีผลกับค่า) เช่น อัตราการใช้บัตรเครดิตของลูกค้า (good, regular, poor) ซึ่งแต่ลักษณะของข้อมูลจะสามารถบอกถึงจำนวนหรือความถี่มากน้อยได้

➢ *แบบ Quantitative*

- Continuous : จะเก็บค่าตัวเลขที่เป็นจำนวนจริง (Real number) เช่น ค่าใช้จ่ายของบริษัทเฉลี่ยต่อเดือน
- Discrete : จะเก็บค่าตัวเลขที่เป็นจำนวนเต็ม (Integer) เช่น จำนวนพนักงานในบริษัท

2.) <u>การกรองข้อมูล</u> (Data Preprocessing)

ในกระบวนการนี้จะมีปริมาณข้อมูลส่วนหนึ่งที่ถูกเลือกเข้ามาจากกระบวนการ Data Selection ซึ่งข้อมูลเหล่านี้จะต้องเป็นข้อมูลที่ถูกต้องพร้อมสำหรับการทำ Mining แต่บางครั้งข้อมูลที่ได้นี้อาจยังมีข้อมูลที่ไม่ถูกต้อง จึงต้องทำการตรวจสอบก่อนโดยใช้หลักการทางสถิติ เช่น ข้อมูลที่เป็น Categorical การวัดการกระจายของข้อมูลจะทำให้เข้าใจข้อมูลที่มีอยู่ได้ดียิ่งขึ้น วิธีการที่ง่ายที่สุดคือการนำข้อมูลนั้นไปสร้างกราฟ ซึ่งจะช่วยให้เห็นความโน้มเอียงของข้อมูลและข้อมูลที่ผิดปกติได้ ส่วนข้อมูลที่เป็นตัวเลข การวิเคราะห์ข้อมูลทำได้โดย การหาค่าสูงสุด(Max) ค่าต่ำสุด(Min)

ค่าเฉลี่ย(Mean) ค่าฐานนิยม(Mode) ค่ามัธยฐาน(Median) ซึ่งเราจะเห็นข้อมูลที่ผิดปกติในขั้นตอนนี้คือ

- *Noisy Data* เป็นข้อมูลที่มีลักษณะแตกต่างจากข้อมูลที่คาดการณ์ไว้ หรือที่ควรจะเป็น ซึ่งอาจจะเกิดจากการป้อนข้อมูลผิด เช่น บันทึกค่าเงินเดือนพนักงานติดลบ หรือบันทึกส่วนสูงเป็น 560 ซ.ม. เป็นต้น ซึ่งค่าเหล่านี้ควรถูกแก้ไข หรือไม่นำมาวิเคราะห์ ดังนั้นจึงควรมีขั้นตอนของการตรวจสอบข้อมูลก่อนนำไปใช้
- *Missing Value* ข้อมูลที่ไม่ได้ถูกเลือกมาจากขั้นตอน Data Selection คือมีข้อมูลบางส่วนหายไป อาจเกิดจากความผิดพลาดของคนหรือไม่มีข้อมูลส่วนนี้ในขณะที่รับข้อมูล ถ้าข้อมูลที่ขาดมีจำนวนน้อย อาจแก้ไขโดยการตัดข้อมูลนั้นทิ้งทั้งรายการ แต่ถ้าข้อมูลที่ขาดไปมีมากอาจต้องบันทึกส่วนที่หายไปด้วยค่าเฉลี่ย (สำหรับข้อมูลที่เป็น Categorical อาจบันทึกด้วยค่าฐานนิยมแทนหรือบันทึกเป็น "Unknow")

3.) การแปลงข้อมูล (Data Transformation)

เป็นการแปลงข้อมูลให้อยู่ในรูปแบบที่เหมาะสมที่พร้อมจะนำไปวิเคราะห์ตาม Algorithm ของ Data Mining ที่จะใช้ ซึ่งจะมีลักษณะเฉพาะแตกต่างกันไป เช่น การแปลงข้อมูลให้เป็นช่วงเพื่อใช้กับ Decision Tree หรือการปรับอัตราส่วนตัวเลขให้อยู่ในช่วง 0-1 เพื่อใช้กับ Algorithm ใน Neural Network

3. ขั้นตอนการทำ Data Mining

ขั้นตอนนี้ถือว่าเป็นส่วนสำคัญที่สุดของการทำ Data Mining เพราะการเลือกเอาวิธีการและ Algorithm ในการทำ Mining ที่เหมาะสม ก็จะทำให้การ Mining ได้ผลอย่างรวดเร็วและถูกต้องตามจุดประสงค์ที่ต้องการ ในขั้นตอนนี้จะเป็นการประมวลผลข้อมูลตาม Algorithm ที่ได้กำหนดไว้ ซึ่งจะมีความสัมพันธ์กับการวิเคราะห์ข้อมูลและขั้นตอนที่ผ่านมา โดยเมื่อทำในส่วนของ Data Mining แล้วอาจต้องย้อนกลับไปทำในขั้นตอนของการเตรียมข้อมูลใหม่ ในการพัฒนา Data Mining นั้นจะเกี่ยวข้องกับการใช้ Algorithm หลายๆ แบบ ซึ่งแต่ละแบบก็มีข้อดีและข้อเสียแตกต่างกันไป

4. ขั้นตอนการวิเคราะห์ผลลัพธ์ที่ได้

เป็นการวิเคราะห์ผลของการประมวลผล ซึ่งจะทำการแปลความหมายผลลัพธ์ที่ได้จากขั้นตอนการทำ Mining ว่าสามารถนำมาใช้ได้ตามวัตถุประสงค์ที่ต้องการหรือไม่ รวมทั้งเป็นการประเมินถึงความถูกต้องของผลลัพธ์ที่ได้จากการทำ ซึ่งก็เป็นส่วนสำคัญเช่นกัน เนื่องจากบางครั้งผลที่ได้อาจจะยัง

มีข้อผิดพลาดอยู่บ้าง โดยจะต้องทำการนำแบบจำลองที่ได้ไปทำการทดสอบกับข้อมูลชุดอื่นว่าได้ผลลัพธ์ที่ถูกต้องเช่นเดียวกันหรือไม่ ซึ่งการทำงานในส่วนนี้จำเป็นต้องใช้ทักษะในการวิเคราะห์ข้อมูลและการวิเคราะห์ทางธุรกิจเข้ามาช่วยด้วย

5. ขั้นตอนการนำสารสนเทศที่ได้ไปใช้ประโยชน์

เป็นขั้นตอนสุดท้ายของกระบวนการทั้งหมด ซึ่งเป็นการรวบรวมความเข้าใจในแบบจำลองที่เป็นผลมาจากขั้นตอนการวิเคราะห์ผลลัพธ์ที่ได้ มารวมเข้ากับส่วนความรู้ทางธุรกิจเพื่อที่จะนำเสนอถึงวิธีการที่จะนำผลที่ได้นี้ไปใช้ให้เกิดประโยชน์

2.3 เทคนิคในการทำ Data Mining

การทำ Data Mining ประกอบด้วย 4 model หลัก คือ

1. การสร้างแบบจำลองพยากรณ์ (Predictive Modeling)
2. การแบ่งส่วนฐานข้อมูล (Database Segmentation)
3. การวิเคราะห์ความสัมพันธ์ (Link Analysis)
4. การตรวจสอบค่าเบี่ยงเบน (Deviation Detection)

Predictive Modeling		**Database Segmentation**	
	➢ **Classification**		➢ **Demographic clustering**
	➢ **Value prediction**		➢ **Neural clustering**
Link Analysis		**Deviation Detection**	
	➢ **Associations discovery**		➢ **Visualization**
	➢ **Sequential pattern discovery**		➢ **Statistics**
	➢ **Similar time sequence discovery**		

รูปที่ 2.3 เทคนิคแบบต่างๆ ของ Data Mining

- **การสร้างแบบจำลองพยากรณ์ (Predictive Modeling)**

เป็นการทำนายถึงความเป็นไปได้ โดยใช้การสังเกตจากรูปแบบของข้อมูลที่มีอยู่ คือเราจะใช้ Model นี้ในการวิเคราะห์ฐานข้อมูลที่มีอยู่เพื่อตัดสินใจเลือกลักษณะข้อมูลที่ต้องการ โดยมีลักษณะเป็นการเรียนรู้จากกลุ่มข้อมูลที่ได้กำหนดไว้ แล้วจึงนำไปวิเคราะห์กลุ่มข้อมูลที่ต้องการ ซึ่งวิธีนี้เรียกว่า Supervised Learning ดังนั้นข้อมูลที่มีอยู่ต้องสมบูรณ์ จึงจะทำให้ผลลัพธ์ออกมาถูกต้อง เพราะเราต้องนำข้อมูลในอดีตมาสร้างแบบจำลอง การทำงานจะแบ่งออกเป็น 2 ขั้นตอน คือ

1.) Training Phase คือขั้นตอนการสร้างแบบจำลองขึ้นมาใหม่โดยใช้ข้อมูลในอดีต ซึ่งจะใช้ข้อมูลประมาณ 80% ของข้อมูลทั้งหมด

2.) Testing Phase คือขั้นตอนที่ใช้ทำการทดสอบแบบจำลองที่สร้างว่ามีความเหมาะสมหรือไม่ โดยจะนำข้อมูลส่วนที่เหลือ 20% จากช่วง Training Phase มาใช้ทดสอบแบบจำลองที่สร้างขึ้น

Predictive Modeling ยังสามารถแบ่งย่อยได้อีก เป็น 2 เทคนิคคือ

1. Classification : เป็นการทำนายว่าสิ่งนั้นควรอยู่ในกลุ่มไหน ซึ่งเป็นการแบ่งกลุ่มของข้อมูลตามชนิดของกลุ่มข้อมูลที่ควรจะเป็น และสามารถแบ่งกลุ่มข้อมูลได้อย่างชัดเจน เช่น การจัดกลุ่มของลูกค้าเพื่อพิจารณาว่าควรจะให้วงเงินสินเชื่อเพิ่มขึ้นหรือไม่ เป็นต้น ซึ่งวิธีที่นิยมใช้คือ Tree Induction และ Neural Induction

2. Value prediction : เป็นการทำนายถึง ค่าความต่อเนื่องของข้อมูล เป็นการทำนายค่าที่เป็นตัวเลข เช่น การทำนายราคาหุ้น เป็นต้น โดยมีวิธีที่ใช้คือ Linear Regression และ Nonlinear Regression

- **การแบ่งส่วนฐานข้อมูล (Database Segmentation)**

จะเป็นการแบ่งหรือจัดกลุ่มของข้อมูลที่มีลักษณะคล้ายกัน หรือมีคุณสมบัติใกล้เคียงกันในหลายๆ ด้าน ให้เป็นข้อมูลกลุ่มเดียวกัน ซึ่งแต่ละกลุ่มจะถูกเรียกว่า Segments หรือ Clusters การแบ่งกลุ่มข้อมูลนี้เราจะไม่สามารถกำหนดได้ว่าข้อมูลควรจะอยู่กลุ่มใด แต่จะเป็นการกำหนดกลุ่มของข้อมูลจากธรรมชาติของข้อมูลเอง ไม่ได้ใช้ความรู้สึกหรือประสบการณ์ในการตัดสินใจแบ่งกลุ่มข้อมูล และข้อมูลจะถูกจัดการโดย Algorithm ที่เหมาะสม จึงเรียกว่าเป็นรูปแบบของ Unsupervised Learning ซึ่งสามารถแบ่งย่อยตามวิธีที่ใช้ คือ Demographic Clustering และ Neural Clustering

- **การวิเคราะห์ความสัมพันธ์ (Link Analysis)**

เป็นการศึกษาวิเคราะห์ความสัมพันธ์ของข้อมูลหรือกลุ่มของข้อมูล ว่ามีความสัมพันธ์กันหรือไม่ อย่างไร และถ้ามีความสัมพันธ์กันจะสัมพันธ์กันในรูปแบบลักษณะใด โดยเรียกความสัมพันธ์นี้ว่าเป็น "Association" เป็นแบบจำลองที่นิยมกันมากในการวิเคราะห์เพื่อหาความสัมพันธ์ระหว่าง ลูกค้ากับ สินค้าหรือบริการ สามารถแบ่งย่อยได้เป็น 3 ลักษณะ คือ

1. Association Discovery : เป็นการวิเคราะห์ข้อมูลที่เกิดขึ้นพร้อมกันภายในกลุ่มข้อมูลเดียวกัน เป็นเทคนิคหนึ่งที่ได้รับความนิยมมาก ซึ่งมักใช้ในการวิเคราะห์ถึงพฤติกรรมการซื้อของผู้บริโภค จึงมีชื่อเรียกอีกอย่างว่า Market basket analysis

2. Sequential Pattern Discovery : เป็นการศึกษาความสัมพันธ์ระหว่างข้อมูล โดยเทียบข้อมูลกับเวลา ซึ่งเป็นการศึกษาพฤติกรรมในระยะยาว (Long Term Behavior)

3. Similar Time Sequence Discovery : เป็นการศึกษาพฤติกรรมของข้อมูลที่เกิดขึ้นทั้งหมดหรือเกิดขึ้นในช่วงเวลาเดียวกัน เพื่อหาความสัมพันธ์ระหว่างกลุ่มของข้อมูลเหล่านี้

- **การตรวจสอบค่าเบี่ยงเบน (Deviation Detection)**

เป็นเทคนิคที่ใช้ทำการหาค่าที่มีความแตกต่างไปจากค่ามาตรฐาน ว่ามีค่ามากน้อยเพียงใด เป็นแบบจำลองที่ใช้เทคนิคทางสถิติ (Statistics) เพื่อใช้วัดความน่าเชื่อถือของข้อมูล และการแสดงให้เห็นภาพ (Visualization) ซึ่งเป็นการสรุปข้อมูลให้แสดงผลออกมาในรูปแบบ Graphic เช่น Histograms Scatter Plots หรือ กราฟวงกลม เป็นต้นเพื่อให้สามารถเข้าใจได้ง่าย นอกจากนี้ Visualization ยังสามารถนำไปใช้ร่วมกับเทคนิคอื่นๆ โดยใช้ในการแสดงผลที่ได้ในรูปแบบของกราฟฟิก ทำให้เข้าใจได้ง่ายขึ้นอีกด้วย

นอกจากนี้แต่ละเทคนิคของ Data Mining ก็ยังมี Algorithm ต่างๆ ของแต่ละวิธีด้วย ซึ่งการที่จะเลือกใช้ Algorithm ใดนั้น ก็ขึ้นอยู่กับปัจจัยหลายๆ อย่างอีก เช่น ข้อจำกัดในการทำ ลักษณะของข้อมูล ชนิดของข้อมูล และจำนวนข้อมูลที่มีอยู่ ซึ่งบางครั้งก็อาจต้องมีการเปลี่ยนแปลง หากเทคนิคนั้นไม่เหมาะสม สิ่งที่สำคัญของกระบวนการนำมาใช้อยู่ที่การกำหนดกลุ่มของข้อมูลที่จะนำมาทำ และการสร้าง Model ซึ่งหากทำการกำหนดและเลือกใช้อย่างเหมาะสมแล้ว ก็จะทำให้ผลของการทำ Data mining เป็นไปอย่างถูกต้องและรวดเร็ว

จากที่กล่าวมาจะเห็นได้ว่า Data Mining มีเทคนิคและวิธีการที่สามารถนำมาใช้งานอยู่หลายวิธี ซึ่งเราจะต้องเลือกใช้ให้เหมาะสมกับงานประเภทต่างๆ และขึ้นอยู่กับรูปแบบ Application ที่ต้องการนำมาใช้งานด้วย ตัวอย่างการนำ Data Mining ไปประยุกต์ใช้ในงานด้านต่างๆ แสดงในรูปที่ 2.4

Market Management		Risk Management	Fraud Management
Target Marketing		*Forecasting*	
Customer Relationship		*Customer retention*	
Market basket analysis		*Improved underwriting*	*Fraud detection*
Cross selling		*Quality control*	
Market segmentation		*Competitive analysis*	
Predictive Modeling	**Database Segmentation**	**Link Analysis**	**Deviation Detection**
Classification	*Demographic clustering*	*Association discovery*	*Visualization*
Value prediction	*Neural clustering*	*Sequential pattern discovery*	*Statistics*
		Similartime sequence discovery	

รูปที่ 2.4 ตัวอย่างการนำ Data Mining ไปใช้ในงานด้านต่างๆ [Cabena et al. 1998]

สำหรับโครงงานพัฒนาระบบงานที่จะนำเสนอนี้จะใช้วิธีของ Predictive Modeling ด้วยเทคนิค Classification โดยนำ SLIQ algorithm มาประยุกต์ใช้ ซึ่งจะกล่าวถึงรายละเอียดในบทถัดไป

บทที่ 3

การจัดกลุ่ม (Classification)

3.1 ความหมายของ Classification และหลักการทำงาน

Classification เป็นเทคนิคหนึ่งของ Data mining ที่ใช้ใน Predictive modeling ซึ่งมีการทำงานแบบ Supervised Learning กล่าวคือ สามารถสร้างแบบจำลองการจัดหมวดหมู่ได้จากกลุ่มตัวอย่างของข้อมูลที่ได้กำหนดไว้ก่อนล่วงหน้า และสามารถพยากรณ์กลุ่มของรายการที่ยังไม่เคยนำมาจัดหมวดหมู่ได้ด้วย ซึ่งการทำงานประกอบด้วย 2 ขั้นตอนใหญ่ๆ คือ

1. ขั้นตอนการเรียนรู้ (Training Phase)

เป็นการนำเอากลุ่มข้อมูลตัวอย่าง (Training Set) มาทำการวิเคราะห์โดยใช้ Algorithm ของ Classification เพื่อทำการเรียนรู้ และทำการสร้าง Model ที่จะสามารถอธิบายถึงลักษณะความสัมพันธ์ของข้อมูลที่ซ่อนอยู่อยู่ภายในฐานข้อมูล ซึ่ง Model นี้จะมีลักษณะที่กลุ่มของข้อมูลถูกทำการแจกแจงออกเป็น Class ต่างๆ ด้วย Classification rule และ Class แต่ละ Class นี้ก็จะมีลักษณะเฉพาะกลุ่มที่สามารถจะสรุปออกมาเป็นรูปแบบความสัมพันธ์ได้

2. ขั้นตอนการทดสอบข้อมูล (Testing Phase)

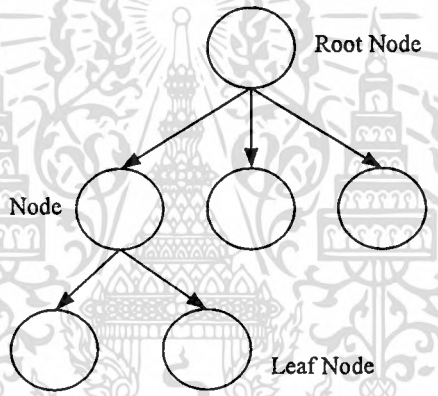
Test Data จะถูกนำมาทดสอบเพื่อดูความถูกต้องของ Classification Rule ที่ถูกสร้างมาจากขั้นตอนการ Training โดยขั้นตอนนี้จะเป็นการพิจารณาว่า Classification rule ที่ถูกสร้างขึ้นมีความเหมาะสมที่จะสามารถนำไปใช้กับกลุ่มข้อมูลใหม่ๆ ได้หรือไม่

สำหรับเทคนิคที่นิยมใช้ใน Classification นั้นมีอยู่หลายเทคนิค ตัวอย่างเช่น

- Decision Trees
- Bayesian Networks
- Genetic Algorithm
- Neural Networks

3.2 Classification Tree

เป็นการนำข้อมูลมาสร้าง Predictive Model ในรูปของ Decision Tree ซึ่งแสดงผลอยู่ในรูปแบบของแผนภูมิต้นไม้ โครงสร้างของ Decision Tree จะประกอบด้วย Node ต่างๆ โดย Node แรกสุดจะเรียกว่า Root Node แล้วแตกต่อไปเป็น Node ลูก และที่ Node ลูกก็อาจจะแตกเป็น Node ต่อไปอีกได้ ลูกแต่ละระดับอาจมีมากกว่า 2 Node ก็ได้ ส่วน Node ที่ระดับสุดท้ายจะเรียกว่า Leaf Node โดยแต่ละกิ่งของ Tree จะแสดงถึงผลที่เกิดจากการทดสอบเพื่อจัดประเภท (Classification) ส่วนที่ปลายสุดของแต่ละโหนด (Leaf node) ของ Tree จะเป็นกลุ่มของข้อมูลที่ถูกจัดกลุ่มตามประเภทของข้อมูลที่มีอยู่ โดยจาก Root Node จนถึง Leaf Node จะมีเพียงเส้นทางเดียวเท่านั้น ซึ่งเส้นทางนี้จะอธิบายถึงกฎที่ใช้สำหรับการจัดหมวดหมู่ของแต่ละกลุ่ม



รูปที่ 3.1 ตัวอย่างของ Decision Tree

ข้อดีของการใช้ Decision tree ในการทำ Classification

- ใช้เวลาในการทำงานน้อย เมื่อเปรียบเทียบกับเทคนิคของ Classification แบบอื่นๆ และไม่สิ้นเปลืองทรัพยากรในการสร้างมากนัก
- เป็นเทคนิคที่ง่ายต่อการตีความ รวมทั้งผลลัพธ์ที่ได้ก็ง่ายต่อการทำความเข้าใจ
- ผลที่ได้สามารถแปลงเป็นภาษา SQL ซึ่งทำให้สะดวกในการจัดการกับข้อมูลในฐานข้อมูล
- ผลลัพธ์ที่ได้จาก Decision Tree มีความถูกต้องแม่นยำเมื่อเทียบเคียงกับเทคนิคของ Classification แบบอื่น ๆ

ขั้นตอนการสร้าง Classification Tree แบ่งย่อยเป็น 2 ส่วนคือ

1. **Tree Building** เป็นขั้นตอนในการสร้าง Tree จากบนลงล่าง Top-Down โดยเริ่มต้นสร้างจากตำแหน่ง Root node แล้วใช้ Algorithm ในการคำนวณหา Attribute ที่เหมาะสมที่สุดที่จะใช้ในการแตกกิ่งออกมา

2. **Tree Pruning** เป็นขั้นตอนที่จะทำการกำจัดกิ่งหรือส่วนของข้อมูลที่ไม่เกี่ยวข้องออกไป ซึ่งส่วนนี้เกิดจากข้อมูลที่ใช้ในการสร้าง Tree บางส่วนมีข้อผิดพลาด (Noise or Outliers) ข้อมูลที่ผิดปกตินี้จึงปรากฏให้เห็นหากทำการสร้าง Tree ที่มีขนาดใหญ่เกินไป คือทำให้เกิดปัญหาการเข้าถึงข้อมูลที่แตกย่อยมากเกินไป ที่เรียกว่า Overfitting หรือ Overtraining

Classification Tree มี algorithm อยู่หลายแบบให้สามารถเลือกใช้ โดยทั่วไปจะแตกต่างกันที่หลักการในการสร้าง หรือการเลือกพารามิเตอร์ที่จะทำการแตกกิ่งเพื่อที่จะสร้าง Tree หรือหลักการในการ Pruning

ตัวอย่าง Algorithm ของ Classification Tree

- CLS (1966) : เป็นหนึ่งใน algorithm เริ่มแรกของการทำ Decision Tree
- CART (1984) : Classification And Regression Trees
- ID3 (1986) : Induction Decision Tree พัฒนาโดย Quilan
- C4.5 (1993) : Decision Tree Induction Algorithm ซึ่งพัฒนาต่อมาจาก ID3
- SLIQ (1996) : A Fast Scalable Classifier for Data mining
- SPRINT (1996) : A Scalable Parallel Classifier for Data mining
- PUBLIC (1998) : ใช้เทคนิค Tree splitting and Tree pruning Integration
- RainForest (1998) : A Framework of Fast Decision Tree Construction of Large Dataset

สำหรับโครงงานพัฒนาระบบงานนี้ได้เลือกใช้ SLIQ algorithm ซึ่งจะกล่าวถึงรายละเอียดในหัวข้อต่อไป

3.3 SLIQ algorithm

SLIQ (Supervised Learning In Quest) เป็น Decision Tree Classifier รูปแบบใหม่ ซึ่งมีความสามารถในการทำ Classification ดีกว่าแบบอื่นๆ โดยใช้เทคนิคของการ Pre-Sorting , Breadth First Growth เทคนิคใหม่เหล่านี้ช่วยให้ SLIQ สามารถที่จะจัดการประมวลผลข้อมูลขนาดใหญ่บน Disk แทนที่จะทำบนหน่วยความจำของเครื่อง ซึ่งจุดนี้เป็นจุดที่ทำให้ SLIQ เป็นวิธีการที่น่าสนใจและเหมาะต่อการทำ Data Mining

ถึงแม้ว่าจะมีการศึกษาในเรื่อง Classification มาเป็นเวลานานตั้งแต่ในอดีต แต่ Algorithm ของ Classification ส่วนใหญ่ก็ถูกออกแบบมาสำหรับทำงานประมวลผลบนหน่วยความจำเท่านั้น ซึ่งทำให้จำกัดขีดความสามารถในการจัดการฐานข้อมูลที่มีขนาดใหญ่ การออกแบบ Classifier ที่ได้สัดส่วนและสามารถจัดการกับ Training Data ที่มีขนาดใหญ่ จึงถือว่าเป็นปัญหาที่สำคัญ SLIQ Algorithm ได้ถูกสร้างขึ้นในรูปแบบที่ไม่จำเป็นต้องเก็บข้อมูลขนาดใหญ่บนหน่วยความจำของเครื่อง

ข้อดีของ SLIQ

1. ใช้เทคนิค Pre-Sorting ในการจัดการกับข้อมูลแบบตัวเลข ทำให้ช่วยลดเวลาในการประมวลผล เนื่องจากไม่ต้องมีการจัดเรียงข้อมูลใหม่ทุกๆ ครั้ง
2. โครงสร้างการจัดการข้อมูล มีการนำเทคนิคของการแบ่งข้อมูลออกเป็น Class list และ Attribute list แล้วใช้ Index เข้ามาช่วยในการอ้างอิงถึงกัน ทำให้ช่วยลดขนาดของข้อมูลที่ใช้ในการประมวลผล
3. สามารถใช้งานกับข้อมูลจำนวนมากได้ โดยยังคงมีความถูกต้องในการทำงานค่อนข้างสูง ทำให้สามารถรองรับการขยายตัวของระบบได้ (Scalable)

SLIQ ก็เหมือนกับ Decision Tree Classifier อื่นๆ ทั่วไป คือจะมีกระบวนการอยู่ 2 ขั้นตอน คือ Tree Building และ Tree Pruning

- **Tree Building**

ในขั้นตอนแรกนี้ Decision Tree จะทำการแบ่ง Training Data ออกเป็นส่วนๆ โดย Training Set จะถูกแยกออกเป็น 2 ส่วน หรือมากกว่าโดยใช้ Attribute ซึ่งขั้นตอนนี้จะทำซ้ำไปเรื่อยๆ จนกว่าตัวอย่างข้อมูลในแต่ละส่วนจะขึ้นกับ Class ใด Class หนึ่ง

MakeTree(Training Data T)

Partition(T);

Partition (Data S)

If (all points in S are in the same class) **then** return;

Evaluate splits for each attribute A

Use best split found to partition S into S1 and S2 ;

Partition (S1);

Partition (S2);

รูปที่ 3.2 Algorithm ในการสร้าง Tree [Mehta et al. 1996]

- **Tree Pruning**

หลังจากขั้นตอนแรกสิ้นสุดลง เราจะได้ Tree ที่แบ่ง Training Data Set เรียบร้อยแล้ว แต่อาจจะหมายถึงว่า แต่ละกิ่งที่ถูกสร้างใน Tree อาจมีสิ่งแปลกปลอม (Noise) หรือข้อมูลที่บกพร่องได้ ซึ่งจะทำให้เกิดข้อผิดพลาดได้เมื่อมีการทำ Classifying Test Data ฉะนั้นจุดประสงค์หลักของการทำ Tree Pruning เพื่อกำจัดกิ่งที่จะทำให้เกิดข้อผิดพลาด (Error) ต่างๆ โดยการเลือก Subtree ที่มีข้อผิดพลาดน้อยที่สุด

ในช่วง Tree Growth Phase มีปัญหา 2 ข้อที่เกี่ยวข้องกับประสิทธิภาพ

1. เราต้องหาจุดแยกที่จำกัดเขตของ Node Test
2. แยก Data ออกเป็นส่วนๆ หลังจาก Split point ถูกเลือกแล้ว

ใน Classifier แบบดังเดิมนั้น วิธี Depth-First Growth และการเรียงข้อมูลซ้ำๆ ที่ทุกๆ Node ของ Tree จะถูกใช้เพื่อให้ได้การ Split ของ Numeric Attribute ที่ดีที่สุด แต่ในปัจจุบัน SLIQ จะถูกใช้แทนที่วิธีในอดีต โดยการใช้เทคนิค One-Time Pre-Sorting ร่วมกับ Breadth-First Tree Growing

SLIQ Pre-Sorting and Breadth First Growth

ในการทำ Pre-Sorting จะเริ่มต้นด้วยการสร้าง List ของ Training สำหรับแต่ละ Attribute แยกออกมา หลังจากนั้นจะสร้าง List ขึ้นมาอีก เรียกว่า Class list เพื่อใช้ในการอ้างถึง สำหรับ List ของ Attributes ต่างๆ ภายใน Attributes List จะประกอบไปด้วย 2 ฟิลด์ คือ Attribute Value และ Index เพื่ออ้างถึงของ Class List ส่วนใน Class List ก็จะประกอบไปด้วย 2 ฟิลด์ คือ Class Label และ อีกฟิลด์

สำหรับอ้างถึง Leaf Node ใน Decision Tree ในแต่ละ Leaf Node ของ Decision Tree จะหมายถึงการแบ่งของ Training Data โดยการแบ่งจะถูกกำหนดให้สอดคล้องตรงกับเส้นทางการเดินทางจาก Node ไปที่ Root ส่วนที่แบ่งออกมานั้น แยกแยะได้โดยใช้ Class List

เริ่มต้นโดยการกำหนดให้ฟิลด์ที่เป็น Leaf ของ Class List ทั้งหมดชี้ไปที่ Root ของ Decision Ttree หลังจากนั้นจะเริ่มมีการส่งต่อของ Training Data โดยกระจายค่าของ Attributes ให้กับทุกๆ Lists ซึ่งใน รูปที่ 3.3 จะแสดงถึงชุด Training Data ก่อนและหลัง Pre-Sorting

Trainring Data

	age	salary	class
1	30	65	G
2	23	15	B
3	40	75	G
4	55	40	B
5	55	100	G
6	45	60	G

After Pre-Sorting

Age List

age	class list index
23	2
30	1
40	3
45	6
55	5
55	1

Salary List

salary	class list index
15	2
40	4
60	6
65	1
75	3
100	5

Class list

	class	leaf
1	G	N1
2	B	N1
3	G	N1
4	B	N1
5	G	N1
6	G	N1

รูปที่ 3.3 ตัวอย่างการทำ Pre-Sorting

3.3.1 การแตกกิ่งของ Tree (SLIQ Process Node Split)

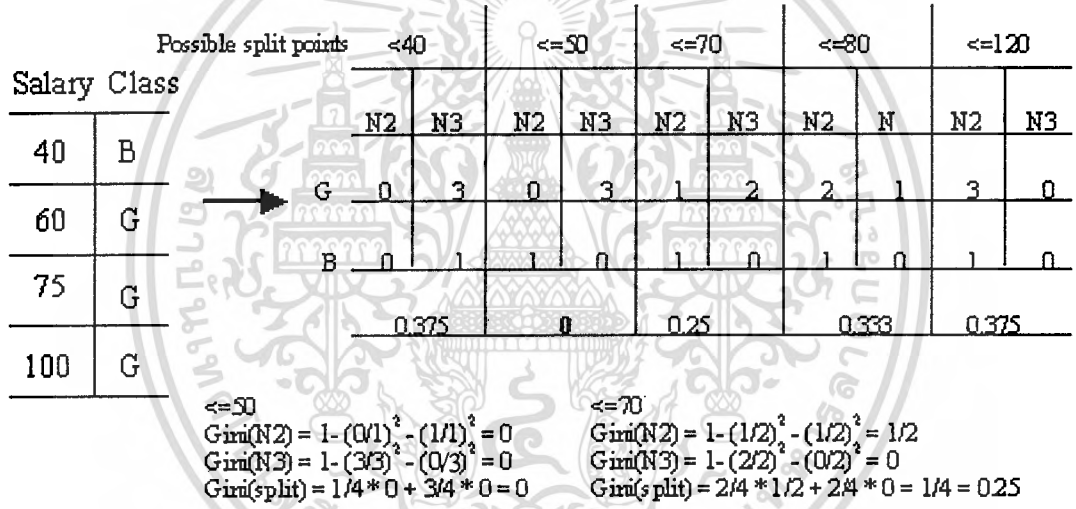
Gini Index ถูกเสนอขึ้นครั้งแรกโดย Breiman et al refbreiman แต่ถูกใช้โดย SLIQ ในการประเมินหาจุดดีของการ Split ในแบบต่างๆสำหรับ Attribute

$$gini\ (T) = 1 - \sum_{j=1}^{n} p_j^2$$

หากข้อมูล set T ประกอบไปด้วยตัวอย่างจาก n Classes โดย gini (T) จะถูกกำหนดให้ p_j เป็น Relative Frequency ของ Class j ใน T หากมีการแยกแล้วแบ่ง T เป็นสอง Subsets คือ T1 และ T2 ซึ่ง Index ของข้อมูล Gini จะถูกแบ่ง Split (S) จะได้ลักษณะนี้

$$gini_{split}(T) = \frac{N_1}{N} gini(T_1) + \frac{N_2}{N} gini(T_2)$$

โดยที่ N1, N2, และ N จะคือจำนวนของ records ใน T1, T2, และ T หลังจากการคำนวณ จุดที่มีค่า Gini Splitting Index โดยเทียบกับ Salary attribute list ซึ่ง Split points ทั้ง 5 ได้ถูกกำหนดเพื่อหาจุดที่เป็น Gini Splitting Index Value และค่าที่ได้ Gini Index Value ที่เล็กที่สุดคือ (<=50) จะถูกเลือกเป็น Split point



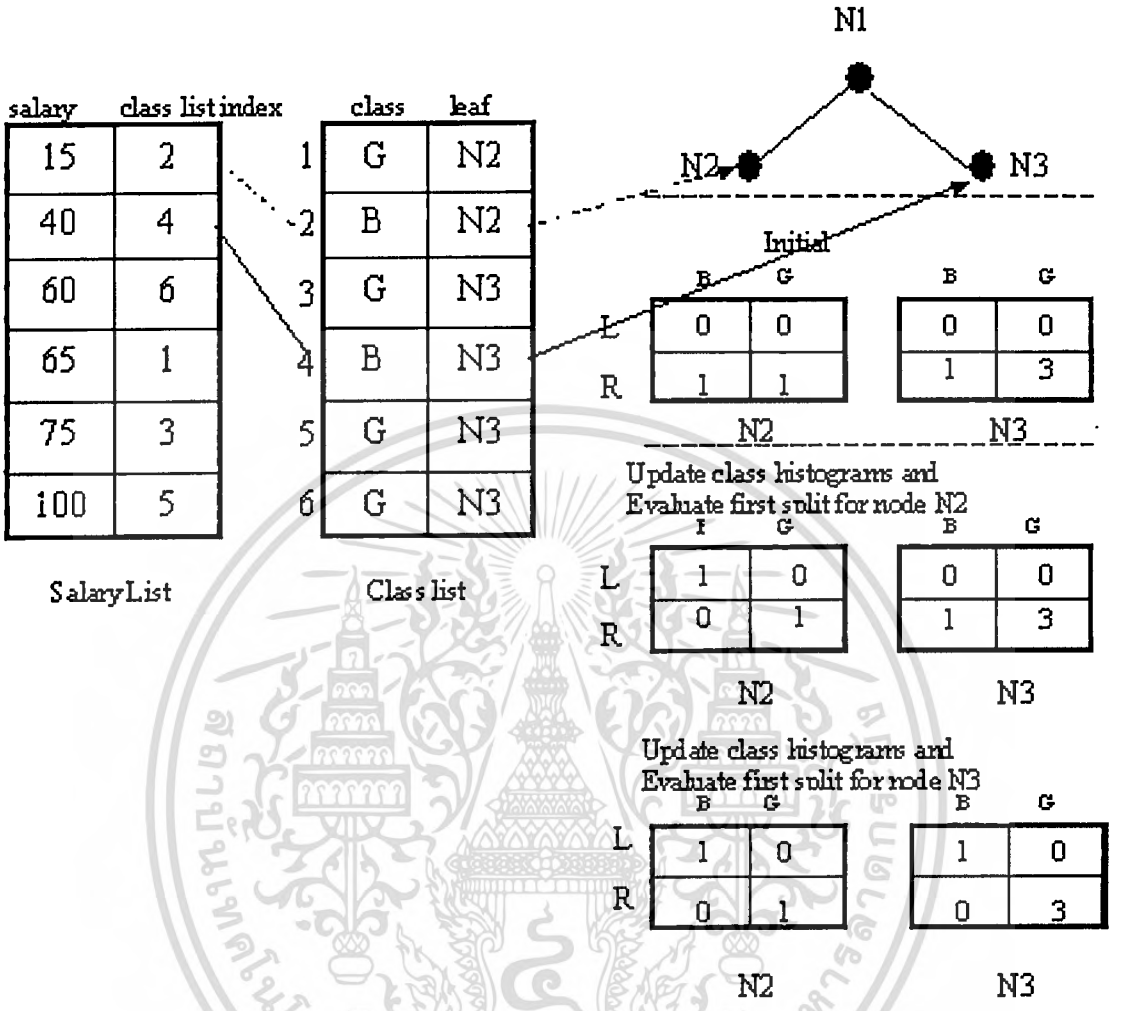
รูปที่ 3.4 ตัวอย่างการคำนวณ Gini Index

ในการคำนวณหา Data Split จาก Leave ทั้งหมดจะถูกกระทำและประเมินไปพร้อมๆ กัน Split point ที่ดีที่สุดจะถูกเก็บไว้ที่แต่ละ Leaf Node โดย Algorithm ดังกล่าวจะแสดงถึงกระบวนการทั้งหมด

```
EvaluateSplits()
    For each attribute A do
        Traverse attribute list of A
        For each value v in the attribute list do
            Find the corresponding entry in the class list , and
                Hence the corresponding class and the leaf node (say l)
            Update the class histogram in the leaf l
            If A is a numeric attribute then
                Compute splitting index for test (A <=v ) for leaf l
        If A is a categorical attribute then
            For each leaf of the tree do
                Find subset of A with best Split
```

รูปที่ 3.5 Algorithm ในการแตกกิ่ง [Mehta et al. 1996]

ในการคำนวณหา Gini Splitting Index สำหรับ Attribute นั้น Class Histogram ที่ซึ่งติดแนบมากับแต่ละ Leaf Node จะถูกใช้กับ SLIQ ในการเก็บสะสม Frequency Distribution ของ Class Values โดยข้อมูลจะต้องตรงกับที่ Node นั้นๆ สำหรับ Numeric Attribute ที่เป็นตัวเลข Histogram List จะเป็นคู่ในลักษณะ <Class, Freq> และใน Categorical Attribute จะแตกต่างกันโดยจะเป็น List แบบ 3 ช่องคือ <Attribute value, Class, Freq> ในรูปที่ 3.6 ได้แสดงถึงการหาค่า Split point จาก Salary attribute สำหรับขั้นที่สองของ Decision Tree ซึ่งในที่นี้จะสมมุติว่าข้อมูลได้ถูก Split จาก Age attribute มาก่อนแล้วโดยใช้ split ที่ Age<= 35 แต่ละ Class Histogram จะสะท้อนให้เห็นถึงการกระจายของ Leaf Node ซึ่งเป็นผลมากจาก Decision Tree ในรูปที่ 3.6 ค่า L จะหมายถึงการกระจายที่น่าพอใจ ในขณะที่ค่า R จะหมายถึง ผลลัพธ์ของการทดสอบที่ไม่น่าพอใจ ค่าแรกใน Salary list จะเป็น N2 ฉะนั้น การ Split ครั้งแรกจะถูกกำหนดค่าไว้ที่ (Salary<=15) สำหรับ N2 และหลังการ Split จากตัวอย่างดังกล่าว (Salary 15, Class Index 2) ค่าที่ได้จะส่งให้ไปทางซ้ายของกิ่งหนึ่งขั้น และที่เหลือไปทางขวาของกิ่ง ทั้งหมดซึ่ง Class Histogram ของ Node N2 ก็จะถูกบันทึก ต่อมาคือ Split (Salary=<40) จะถูกกำหนดไว้สำหรับ Node N3 โดยหลังการ Split จากตัวอย่างดังกล่าว (Salary 40, Class Index4) จะอยู่ที่ฝั่งซ้ายของกิ่ง และ Class Histogram ของ Node N3 จะบันทึกการเปลี่ยนแปลงที่เกิดขึ้น



รูปที่ 3.6 ตัวอย่างของการแตกกิ่ง

ในการทำ Split ของ Categorical Attribute นั้น SLIQ จะใช้วิธีลูกผสมในการกำหนด Subset ของ set S และค่าของ Attribute โดยที่จำนวน Element ใน S ต้องน้อยกว่า Threshold มิเช่นนั้น Element S ที่ได้จาก Best Split จะถูกเพิ่มเข้าไปที่ Set เดิมของ S ซึ่งว่างเปล่า กระบวนการดังกล่าวจะทำซ้ำไปเรื่อยๆ โดยที่ไม่มีการเปลี่ยนแปลงของการ Split เลย

การ Update Class list

ขั้นตอนต่อไปก็คือการสร้าง Child Node สำหรับแต่ละ Leaf Node และการ Update ค่าใน Class List ซึ่งใช้ Algorithm ตามรูปที่ 3.7

UpdateLabels()

For each attribute A used in a split **do**

Traverse attribute list of A

For each value v in the attribute list **do**

Find the corresponding entry in the class list (say e)

Find the new class c to which v belongs by applying

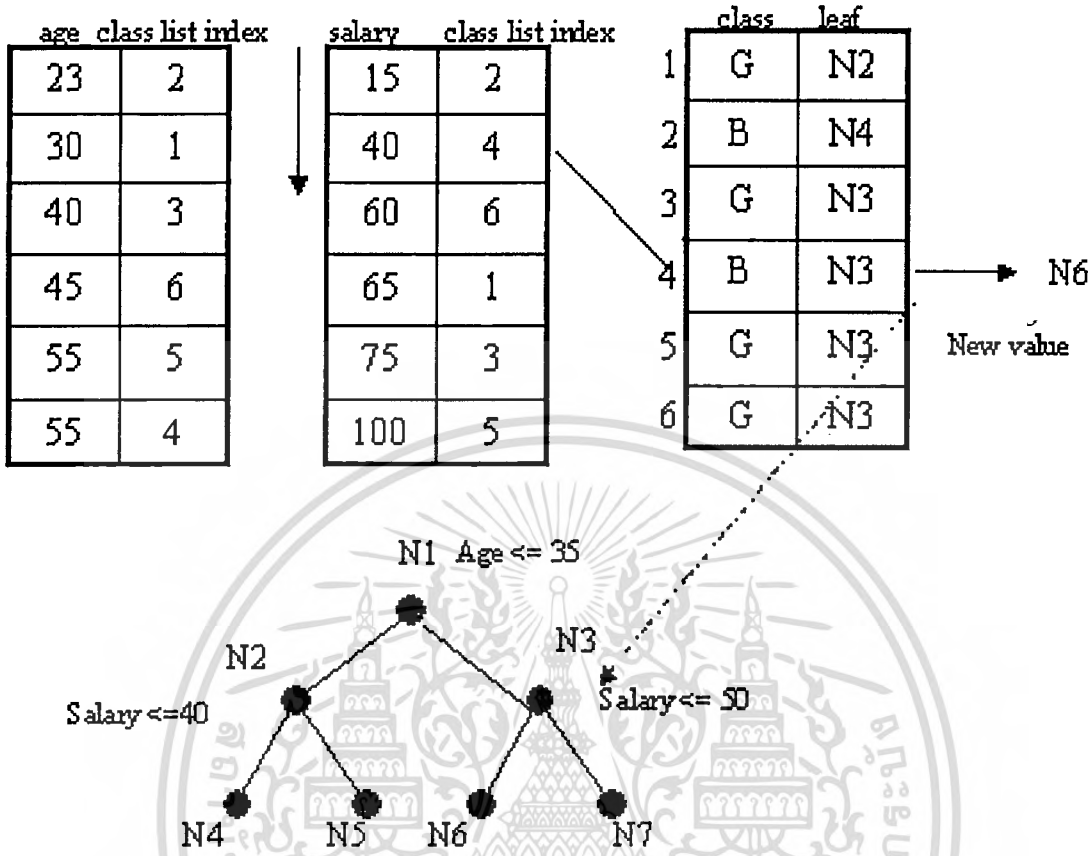
The splitting test at node referenced from e

Update the class label for e to c

Update node referenced in e to the child corresponding to the class c

รูปที่ 3.7 Algorithm ในการ Update Class list [Mehta et al. 1996]

พิจารณารูปที่ 3.8 จะเห็นว่า Class List ถูก Update หลังจาก Node N2 และ Node N3 ถูก Split ออกด้วย Salary Attribute ซึ่งตาม Algorithm นั้น Salary Attribute จะเดินทางและ Class List Entry (Entry 4) ซึ่งมีความเกี่ยวข้องกับค่าของ Salary คือ 40 จะถูก Update ขั้นแรก Leaf ใน Entry 4 ของ Class List จะถูกใช้ในการค้นหา Node ที่ซึ่งในตัวอย่างมีอยู่แล้ว (เช่นใน N3) หลังจากนั้น Split ที่ถูกเลือกโดย N3 จะถูกนำมาใช้ในการค้นหา Child ตัวใหม่ ซึ่งในตัวอย่างมีอยู่แล้ว (เช่นใน N6) ฟิลด์ Leaf ของ Entry 4 ใน Class List จะถูก Update เพื่อให้สอดคล้องกับค่าใหม่ที่เกิดขึ้น



รูปที่ 3.8 ตัวอย่างในการ Update Class list

3.3.2 Tree Pruning

ในการสร้าง Tree ซึ่งจะต้องมีการแตกกิ่งลงไปเรื่อยๆ นั้น เมื่อแตกกิ่งจนเสร็จสิ้นแล้ว เราจะต้องทำการตัดกิ่งของ Tree บางกิ่งออก เพราะเมื่อ Tree มีกิ่งมากๆ แล้วนั้น ก็จะแสดงให้เห็นถึงความผิดปกติของข้อมูลบางส่วนใน Training Data อันเนื่องมากจากข้อมูลบางส่วนที่อาจจะเป็น Noise จึงต้องทำการตัดกิ่งที่มีค่าความน่าเชื่อถือน้อยที่สุดออกไป เพื่อเป็นการปรับปรุงคุณภาพของ Tree และเป็นการปรับแก้การจัดกลุ่ม (Classify) ข้อมูลให้เป็นไปอย่างถูกต้อง ซึ่งหลักการทำงานของการ Pruning ก็คือ การกำจัดการแตกกิ่งในระดับล่างๆของ Tree ที่ให้ค่า Estimate error rate สูงๆ เพื่อให้ Accuracy ของ Tree ที่สร้างดีที่สุด

เทคนิคการ Pruning สามารถแบ่งได้เป็น 2 เทคนิคคือ Pre Pruning และ Post Pruning

- **Pre-Pruning**

จะทำการกำจัดกิ่ง โดยการหยุดการแตกกิ่งตั้งแต่ในช่วงการสร้าง Tree ซึ่งจะทำให้ Node ที่จะแตกต่อไป กลายเป็น Leaf node ซึ่งโดยปกติจะมีการวัดค่าทางสถิติ หรือค่า Information Gain เพื่อบอกถึงความเหมาะสมของการแตกกิ่ง ถ้าพบว่าผลของการเลือกค่าที่ดีที่สุดในการแตกกิ่งน้อยกว่าค่า Threshold ที่ตั้งไว้ การแตกกิ่งก็จะถูกทำให้หยุดลง ซึ่งก็เป็นเรื่องยากที่จะทำการเลือกค่า Threshold ที่เหมาะสม

- **Post Pruning**

จะทำการกำจัดกิ่งออกจาก Tree ที่โตเต็มที่แล้ว โดยการกำจัดกิ่งบางกิ่งออกไป โดยที่แต่ละโหนดจะถูกทำการคำนวณหาค่า Expect Error rate เมื่อ SubTree ใต้โหนดนั้นถูกตัดไป ซึ่งถ้าการ Pruning ทำให้ค่า Expect Error rate สูงขึ้น Subtree นั้นก็จะถูกเก็บเอาไว้ แต่ถ้าทำให้ค่าต่ำลงก็จะทำการตัดกิ่งนั้นออกไป และหลังจากทำการ Pruned Tree แล้ว Test set ก็จะถูกนำมาหาค่า Accuracy ของแต่ละ Tree ซึ่ง Decision Tree ที่ได้ค่า Expect Error rate ต่ำสุดหรือ Accuracy สูงสุดก็จะถูกเลือกนำมาใช้

นอกจากสองเทคนิคนี้ที่เลือกทำการพิจารณาค่า Expect Error rate แล้วยังมีอีกเทคนิคหนึ่งที่พิจารณาถึงจำนวน Bit ที่จะใช้ในการ Encode Model โดย Best Pruned Treeคือค่าของ Tree ที่ใช้จำนวนบิตต่ำสุดในการ Encoding ซึ่งทำให้ไม่ต้องมีการแบ่งข้อมูลออกไปเป็น Test set

Cross Validation

สำหรับโครงงานพัฒนาระบบนี้ใช้วิธีการ Pruning โดยใช้เทคนิคของ Cross Validation ซึ่งเป็นเทคนิคแบบ Post Pruning โดยจะทำการแบ่ง กลุ่มข้อมูลแบบสุ่ม (Random) ออกเป็นสองส่วนคือ Training Set และ Test Set โดย%ของข้อมูลที่จะใช้เป็น Training Set และ Test Set ก็ขึ้นอยู่กับปัจจัยหลายอย่าง เช่น จำนวนข้อมูลที่มีอยู่

ซึ่งหลักการทำงานของการ Pruning แบบ Cross Validation ก็คือ จะทำการเช็คค่า Accuracy ของโหนดที่ประกอบด้วย Sub Tree ว่าเมื่อทำการเปรียบเทียบค่า Accuracy ระหว่างการมี Sub Tree กับการไม่มี Sub Tree แบบไหนให้ค่า Accuracy ดีกว่ากัน

ดังนั้นถ้าทำการพบว่าค่า Accuracy ของการไม่มี Sub Tree ดีกว่าการมี Sub Tree ก็จะทำการ Pruning ตัด Sub Tree นั้นทิ้งไป และก็จะทำการทำซ้ำไปเรื่อยโดยเริ่มจากด้านล่างของสู่ด้านบนของ Tree จนกระทั่งไม่มี Sub Treeที่จะก่อให้เกิดการ Pruning

การวัด Accuracy

ความแม่นยำ (Accuracy) เป็นสิ่งสำคัญที่ใช้เป็นตัววัดค่าความแม่นยำในการแบ่งกลุ่มข้อมูลในอนาคต หรือ กลุ่มข้อมูลที่ไม่ได้ผ่านการเรียนรู้มาก่อน ซึ่งจะเป็นข้อมูลที่ใช้ในการ Pruning และบอกถึงค่าความแม่นยำของ Model ที่ได้

การวัดความแม่นยำ กรณีที่ถ้าบอกว่ามีความแม่นยำถึง 90 % ของค่าที่ตอบที่เป็น Positive sample อาจจะไม่ถูกต้องนัก ถ้าข้อมูลที่ใช่มีเพียง 3-4 % ของ Training data ดังนั้นวิธีที่น่าจะคำนวณความแม่นยำให้การแบ่งกลุ่มข้อมูลควรพิจารณาในทุกๆค่าที่เป็นไปได้ของข้อมูล

Sensitivity เป็นการวัดความสามารถในการจดจำรูปแบบของ Positive sample ในขณะที่ Specificity เป็นความสามารถในการจดจำรูปแบบของ Negative sample ซึ่งมีสูตรดังสมการข้างล่างนี้ โดยมี Precision เป็นการวัดเปอร์เซ็นต์ของจำนวนข้อมูลที่คาดว่าจะเป็น Positive sample แล้วค่าจริงๆ ก็เป็น Positive sample

$$\text{Sensitivity} = t\_pos \ / \ Pos$$

$$\text{Specificity} = t\_neg \ / \ Neg$$

$$\text{Precision} = t\_pos \ / \ (t\_pos + f\_pos)$$

โดยที่ t_pos คือ จำนวนของค่าที่เป็นจริงของข้อมูล Positive

t_neg คือ จำนวนของค่าที่เป็นจริงของข้อมูล Negative

f_pos คือ จำนวนของค่าที่เป็นเท็จของข้อมูล Positive

Pos คือ จำนวนของข้อมูล Positive

Neg คือ จำนวนของข้อมูล Negative

ซึ่งจาการคำนวณหาค่า Sensitivity และ Specificity ก็จะทำให้เราสามารถทำการหาค่า Accuracy ได้โดย สมการการหาค่า Accuracy จะอยู่ในรูปของฟังก์ชั่นกับค่า Sensitivity และค่า Specificity ดังสมการข้างล่างนี้

$$\text{Accuracy} = \text{Sensitivity} * Pos \ / \ (Pos + Neg) + \text{Specificity} * Neg \ (Pos + Neg)$$

โดยที่ Pos คือ จำนวนของข้อมูล Positive

Neg คือ จำนวนของข้อมูล Negative

นอกจากนี้เรายังสามารถทำการเปรียบเทียบ Model ที่สร้างว่า Model ใดมีความถูกต้องในการสร้าง Model จากข้อมูลชุดเดียวกัน โดยการเปรียบเทียบจากค่า Accuracy ที่ได้ โดย Model ที่มีค่า Accuracy สูงก็แสดงว่ามีความถูกต้องในการ Predict ค่าของข้อมูลสูง

บทที่ 4
ระบบสนับสนุนการพิจารณาอนุมัติสินเชื่อเบื้องต้น

ในบทนี้จะกล่าวถึงรายละเอียดทั้งหมดของระบบงานที่ทำการพัฒนาขึ้น ตั้งแต่การเตรียมข้อมูล ขั้นตอนการทำงานของระบบ ตลอดจนการนำผลลัพธ์ที่ได้ไปใช้งาน

4.1 วัตถุประสงค์

วัตถุประสงค์ของการทำ Data Mining สำหรับระบบงานนี้ คือ ช่วยให้องค์กรที่มีการให้สินเชื่อรู้จักลูกค้าได้ดียิ่งขึ้น ทั้งในด้านลักษณะทั่วไป เช่น อายุ เพศ การศึกษา อาชีพ รายได้ ฯลฯ รวมไปถึงพฤติกรรม นั่นหมายถึงการจะได้เห็น Relation และ Pattern ต่างๆ ที่ซ่อนอยู่ในข้อมูล ซึ่ง Pattern ดังกล่าวนี้จะช่วยองค์กรในการควบคุมดูแล และบริหารความเสี่ยงจากการให้สินเชื่อให้เป็นไปอย่างมีประสิทธิภาพ

4.2 เครื่องมือที่ใช้ในการพัฒนาระบบ

สำหรับการพัฒนาระบบงานนี้ได้ใช้ Microsoft Visual Basic 6.0 ในการพัฒนาระบบทั้งหมด เนื่องจาก MS Visual Basic เป็น Development Tool ที่สามารถพัฒนาบนระบบปฏิบัติการ Windows และการทำงานเป็นแบบ Visual Programming ทำให้สามารถเลือกออกแบบการทำงานของระบบได้อย่างสะดวกและรวดเร็ว รวมถึงสามารถทำการติดต่อกับระบบฐานข้อมูลได้ จึงเหมาะกับระบบงานนี้ที่ต้องมีการติดต่อและนำเอาข้อมูลจากฐานข้อมูลมาประมวลผล

4.3 โครงสร้างและรายละเอียดของระบบ

ระบบนี้จะแบ่งการทำงานออกเป็น 2 ส่วน คือ

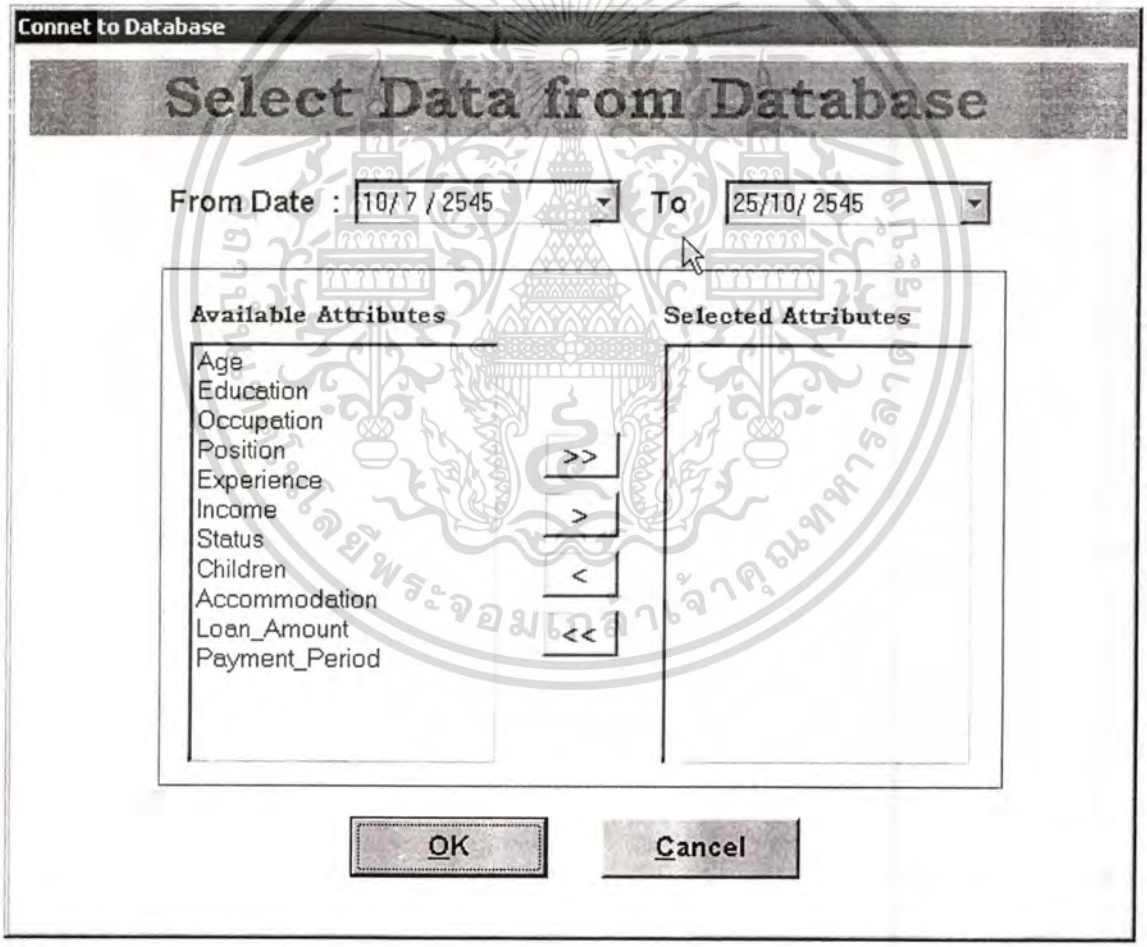
1. การสร้างแบบจำลอง (Model Building) - เป็นขั้นตอนในการนำข้อมูลลูกค้าในอดีตที่มีอยู่มาทำการสร้าง Classification Tree Model โดยวิธีการของ SLIQ Algorithm
2. การทำนายข้อมูล (Data Prediction) - เป็นการนำ Classification Tree Model ที่ได้มาทำนายผลของลูกค้าคนใหม่ที่มาทำการขอกู้ว่ามีความน่าเชื่อถือมากน้อยเพียงใด

4.3.1 การสร้างแบบจำลอง (Model Building)

ข้อมูลที่ต้องนำเข้า (Input) : มี 2 ส่วน ดังนี้

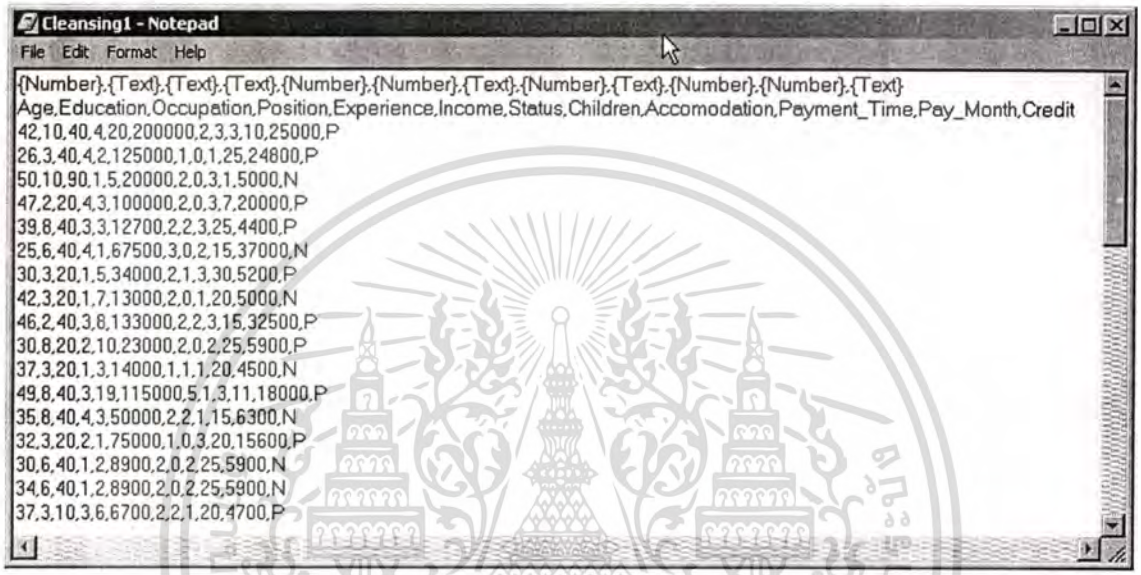
ส่วนแรกคือ ข้อมูลของลูกค้าในอดีตที่จะนำมาใช้ในการสร้าง Classification Tree Model ซึ่งในระบบนี้สามารถเลือกได้ 2 แบบคือ

- รับข้อมูลจาก Database - โดยระบบจะทำการแสดง Attribute ต่างๆ ที่เป็นข้อมูลเบื้องต้นของลูกค้าที่จำเป็นต้องใช้ในการขอสินเชื่อ ซึ่งผู้ใช้สามารถเลือกข้อมูลได้ตามช่วงวันที่ และเลือกได้ว่าต้องการนำ Attribute ใดไปใช้ในการสร้าง Model บ้าง ตัวอย่างดังแสดงในรูป 4.1



รูปที่ 4.1 หน้าจอแสดงส่วนที่รับข้อมูลจาก Database

- รับข้อมูลจาก Text file - โดยที่ Text file ที่จะนำมาใช้นี้จะต้องมีรูปแบบ (Format) ตามที่กำหนด กล่าวคือข้อมูลแต่ละตัวจะต้องคั่นด้วยเครื่องหมาย " , " และบรรทัดแรกจะต้องเป็นชื่อชนิดของข้อมูลในแต่ละ Attribute ที่อยู่ในเครื่องหมาย " {...} " ส่วนบรรทัดที่ 2 จะเป็นชื่อของแต่ละ Attribute ที่ได้ทำการเลือกมา ตัวอย่างดังแสดงในรูป 4.2



รูปที่ 4.2 ตัวอย่างรูปแบบของ Text file ที่จะนำมาใช้เป็น Input

ส่วนที่สองคือ Parameter อีก 3 ตัวที่ผู้ใช้จะต้องกำหนดเป็น Input ให้กับระบบ คือ

- Maximum Tree Level - จำนวน Level สูงสุดของ Tree ที่ต้องการ
- Minimum Quantity at Leaf Node - จำนวนข้อมูลอย่างน้อยที่สุดที่ต้องมีในแต่ละ Node
- Percentage of Training Data - ร้อยละของจำนวนข้อมูลทั้งหมดที่ต้องการนำไปใช้ในการ Training เพื่อสร้าง Tree โดยที่จะต้องมีค่าไม่น้อยกว่า 50%

สำหรับข้อมูลที่จะนำมาใช้ในระบบนี้ เป็นข้อมูลในอดีตของลูกค้าที่ยื่นใบคำขอสินเชื่อจากธนาคารแห่งหนึ่ง โดยมีรายละเอียดดังนี้

ชื่อข้อมูลภาษาอังกฤษ	ชื่อข้อมูลภาษาไทย	ชนิดของข้อมูล
Age	อายุ	Number
Education	การศึกษา	Number
Occupation	อาชีพ	Text
Position	ตำแหน่งงาน	Text
Experience	ประสบการณ์การทำงาน	Number
Income	รายได้/เดือน	Number
Status	สถานภาพสมรส	Text
Children	จำนวนบุตร	Number
Accommodation	ประเภทที่อยู่อาศัย	Text
Loan_Amount	จำนวนเงินที่กู้ยืม	Number
Payment_Period	ระยะเวลาในการผ่อนชำระ (ปี)	Number
Credit	ความน่าเชื่อถือ	Text

ตาราง 4.1 ตารางแสดงข้อมูลลูกค้า

Education Code	Education Description
1	ปริญญาเอก
2	ปริญญาโท
3	ปริญญาตรี
4	อนุปริญญา
5	ปวท.
6	ปวส.
7	ปวช.

ตาราง 4.2 ข้อมูลรหัสการศึกษาและความหมาย

Education Code	Education Description
8	มัธยมปลาย
9	มัธยมต้น
10	ประถมศึกษา
11	ประกาศนียบัตรประโยคมัธยมครู
12	อื่นๆ

ตาราง 4.2 (ต่อ) ข้อมูลรหัสการศึกษาและความหมาย

Occupation Code	Occupation Description
00	ผู้ประกอบอาชีพแพทย์
10	ข้าราชการ, รัฐวิสาหกิจ, ลูกจ้างประจำ
20	พนักงานและลูกจ้าง บริษัท, ห้างร้าน, สำนักงาน
30	ผู้ประกอบอาชีพอิสระอื่นๆ
40	ผู้ประกอบการ, เจ้าของ, กรรมการ, หุ้นส่วน
50	พนักงานและลูกจ้าง, กลุ่มบุคคลในสถาบันที่ไม่แสวงหากำไร
60	ผู้ประกอบอาชีพการศึกษา
70	ผู้ประกอบอาชีพในวงการบันเทิง และสื่อสารมวลชน และผู้ประกอบอาชีพเกี่ยวกับการเขียน
90	อื่นๆ

ตาราง 4.3 ข้อมูลรหัสอาชีพและความหมาย

Position Code	Position Description
1	ผู้บริหารระดับสูง-กลาง, ข้าราชการระดับสูง-กลาง
2	รองผู้จัดการ, หัวหน้า, ผู้ช่วย
3	พนักงานทั่วไป, ผู้ประกอบวิชาชีพอิสระ
4	ข้าราชการทั่วไป, ผู้ชำนาญการ, รับจ้าง
5	อื่นๆ

ตาราง 4.4 ข้อมูลรหัสตำแหน่งงานและความหมาย

Status Code	Status Description
1	โสด
2	สมรสจดทะเบียน
3	สมรสไม่จดทะเบียน
4	หม้าย
5	หย่า
6	อื่นๆ

ตาราง 4.5 ข้อมูลรหัสสถานภาพสมรสและความหมาย

Accommodation Code	Accommodation Description
1	อาศัยอยู่กับบิดา/มารดา
2	บ้านเช่าอาศัย
3	ที่อยู่ตนเอง/คู่สมรส

ตาราง 4.6 ข้อมูลรหัสประเภทที่อยู่อาศัยและความหมาย

Credit Code	Credit Description
P	น่าเชื่อถือ (เครดิตดี)
N	ไม่น่าเชื่อถือ (เครดิตไม่ดี)

ตาราง 4.7 ข้อมูลรหัสความน่าเชื่อถือและความหมาย

การทำงาน (Process) :

การทำ Classification จะใช้ข้อมูลในอดีตของลูกค้าที่มีอยู่ ที่เราทราบค่าของ Class ที่เราสนใจแล้ว ซึ่งในที่นี้ Class ที่เราสนใจก็คือลักษณะการเป็นลูกหนี้ที่มีเครดิตดี และเครดิตไม่ดีของธนาคารแห่งหนึ่ง และนำข้อมูลที่เราทราบค่าเหล่านั้นแล้วมาทำการสร้าง Classification Tree Model โดยใช้วิธีการของ SLIQ Algorithm โดยจะทำการสร้าง Tree ตามเงื่อนไขของ Parameter 3 ตัวที่ได้กำหนดไว้ในส่วนของ Input ที่ได้กล่าวมาแล้ว

ผลลัพธ์ที่ได้ (Output) :

เมื่อผู้ใช้สั่งให้ระบบทำการสร้าง Model แล้ว ผลที่ได้จากการทำ Data Mining ด้วยวิธี Classification จะแสดงผลลัพธ์ให้ 3 ส่วน คือ

- Classification Tree Model : แสดงรูปของ Tree ที่ได้จากการประมวลผล
- Classification Rule : แสดงเงื่อนไขของผลลัพธ์ทั้งหมดที่ปรากฏใน Tree ออกมาเป็นภาษาเขียน
- Model Evaluation : แสดงจำนวนข้อมูลทั้งหมดที่ใช้ในการสร้าง Tree (Number of Data) และค่าความน่าเชื่อถือ (Accuracy of Model) ของ Tree ที่ได้

ในส่วนของ Classification Tree Model นั้น ผู้ใช้สามารถทำการบันทึก Model ที่สร้างเสร็จแล้วไว้ได้ โดย File ที่ทำการบันทึกจะอยู่ในรูปแบบ " *.tree " ซึ่งจะนำไปใช้ในส่วนของ Data Prediction ต่อไป

4.3.2 การทำนายข้อมูล (Data Prediction)

ข้อมูลที่ต้องนำเข้า (Input) : มี 2 ส่วน ดังนี้

ส่วนแรกคือ Classification Tree Model ที่จะต้องนำมาใช้ในการทำนายข้อมูล ซึ่งผู้ใช้สามารถเลือกได้จาก File ที่อยู่ในรูปของ *.tree ที่ได้ทำการบันทึกไว้

ส่วนที่สองคือ Customer's Profile หรือประวัติของลูกค้ารายใหม่ ที่ต้องการนำมาทำนายผลในเบื้องต้นว่ามีความน่าจะเป็นที่ควรจะอนุมัติสินเชื่อให้หรือไม่ ในส่วนนี้จึงเป็นข้อมูลประวัติโดยทั่วไปของลูกค้าตามใบคำขอสินเชื่อเท่านั้น

การทำงาน (Process) :

เนื่องจาก Classification Tree เป็นหนึ่งในวิธีการของการสร้างแบบจำลองพยากรณ์ (Predictive Modeling) ดังนั้นเราจึงนำ Tree Model ที่ได้จากการทำ Classification มาใช้ในการทำนายข้อมูลที่เราสนใจได้ ในระบบนี้เราจึงนำข้อมูลลูกค้ารายใหม่ที่ต้องการทำนายความน่าจะเป็นในการเห็นสมควรอนุมัติสินเชื่อว่าควรจะให้ผ่าน หรือไม่ผ่าน ขั้นตอนการทำงานก็คือระบบจะนำข้อมูลเหล่านั้นมาผ่าน Tree Model ที่ผู้ใช้ได้เลือกไว้ แล้วจึงสรุปผลที่ได้ออกมา

ผลลัพธ์ที่ได้ (Output) :

ผลลัพธ์ที่ได้จะแสดงให้เห็นใน Tree ว่าข้อมูลตกอยู่ในกลุ่มใด

- ถ้าผลลัพธ์ของข้อมูลที่ได้คือ "ผ่าน" จะปรากฏรูป  ในตำแหน่งกลุ่มที่ข้อมูลนั้นๆ อยู่
- ถ้าผลลัพธ์ของข้อมูลที่ได้คือ "ไม่ผ่าน" จะปรากฏรูป  ในตำแหน่งกลุ่มที่ข้อมูลนั้นๆ อยู่

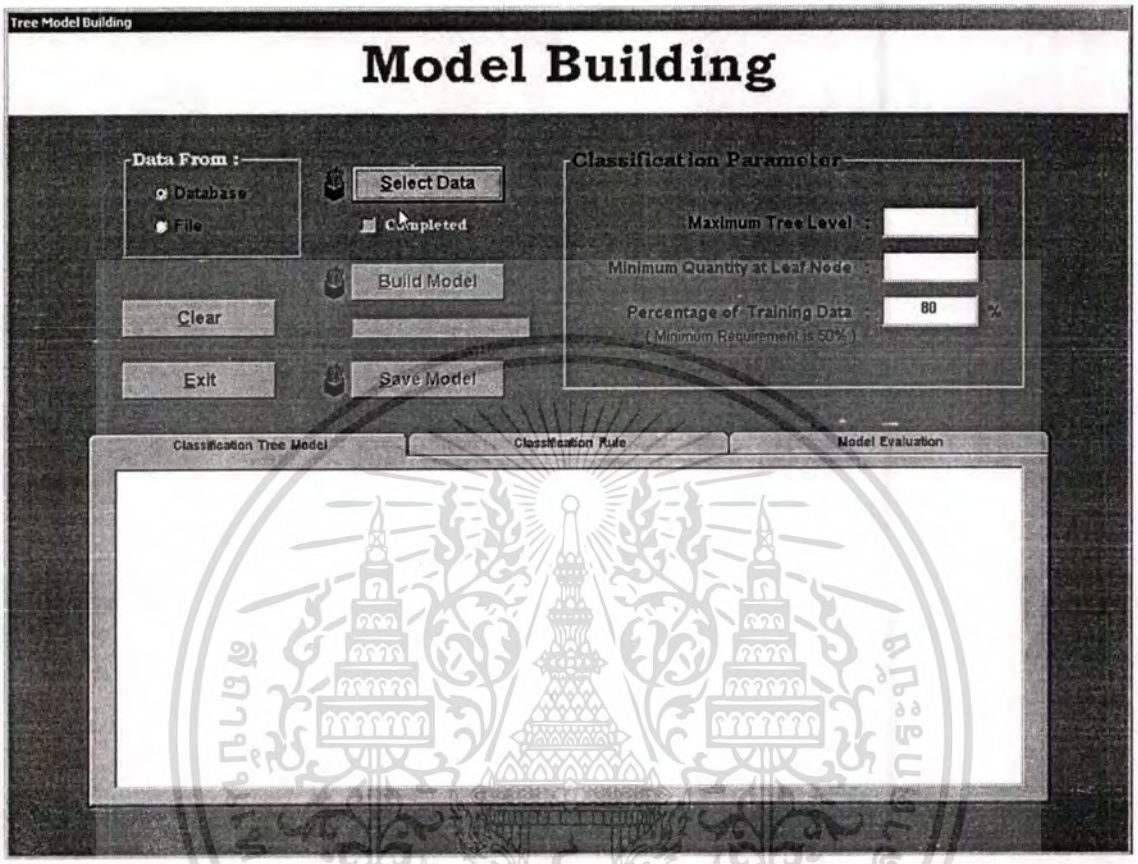
และจะแสดงผลของการทำนาย(Prediction Result) ว่าผ่าน(Pass) หรือไม่ผ่าน(Not Pass) รวมถึงแสดงค่าความเชื่อมั่น(Percentage of Confidential) ของผลลัพธ์ที่ได้

4.4 ขั้นตอนและรายละเอียดการใช้งาน

รูปที่ 4.3 หน้าจอหลักของระบบสนับสนุนการตัดสินใจอนุมัติสินเชื่อเบื้องต้น

ส่วนแรกนี้เป็นหน้าจอหลักของระบบสนับสนุนการตัดสินใจอนุมัติสินเชื่อเบื้องต้น ซึ่งประกอบด้วย

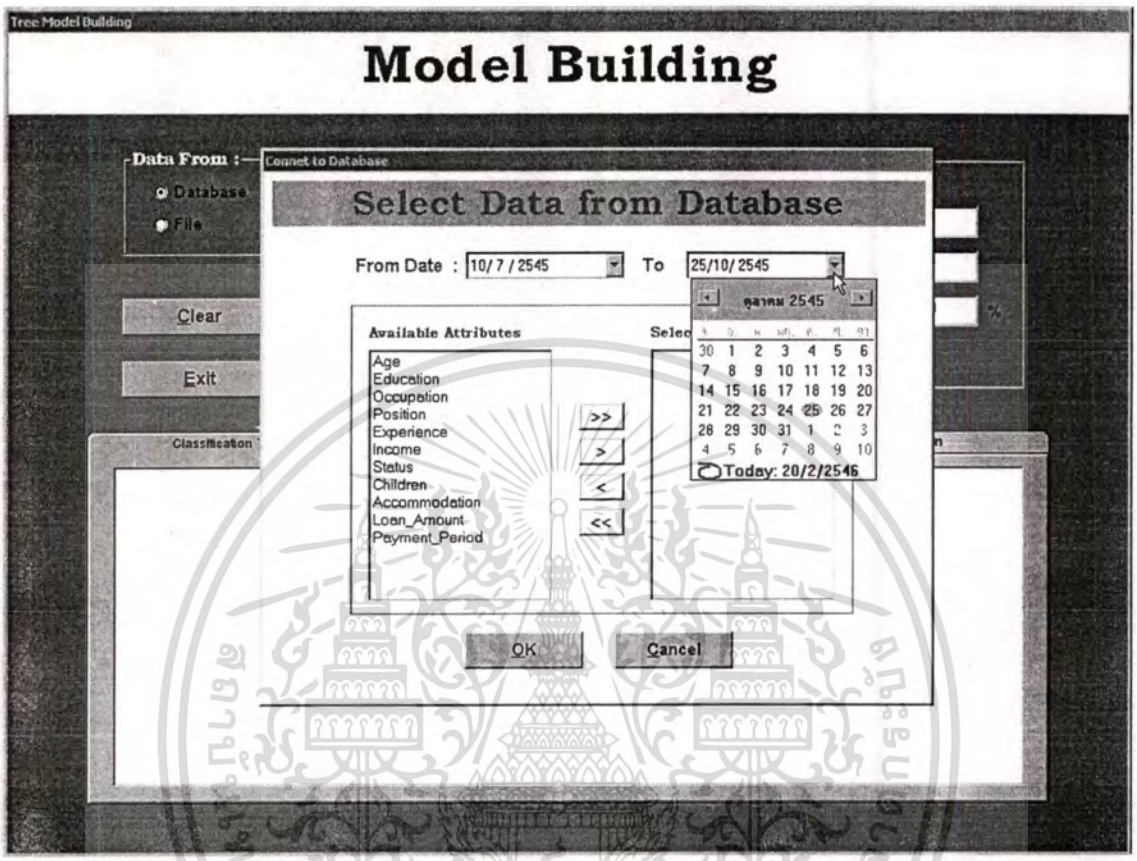
- Model Building : เป็นส่วนของการสร้าง Classification Tree Model
- Data Prediction : เป็นส่วนของการทำนายข้อมูลลูกค้าที่มาทำการขอสินเชื่อ
- About Program : บอกที่มาและวัตถุประสงค์ของโปรแกรม
- Exit : ออกจากระบบ



รูปที่ 4.4 หน้าจอส่วนที่ใช้สร้าง Classification Tree Model

หน้าจอนี้เป็นส่วนที่ใช้ในการสร้าง Classification Tree Model

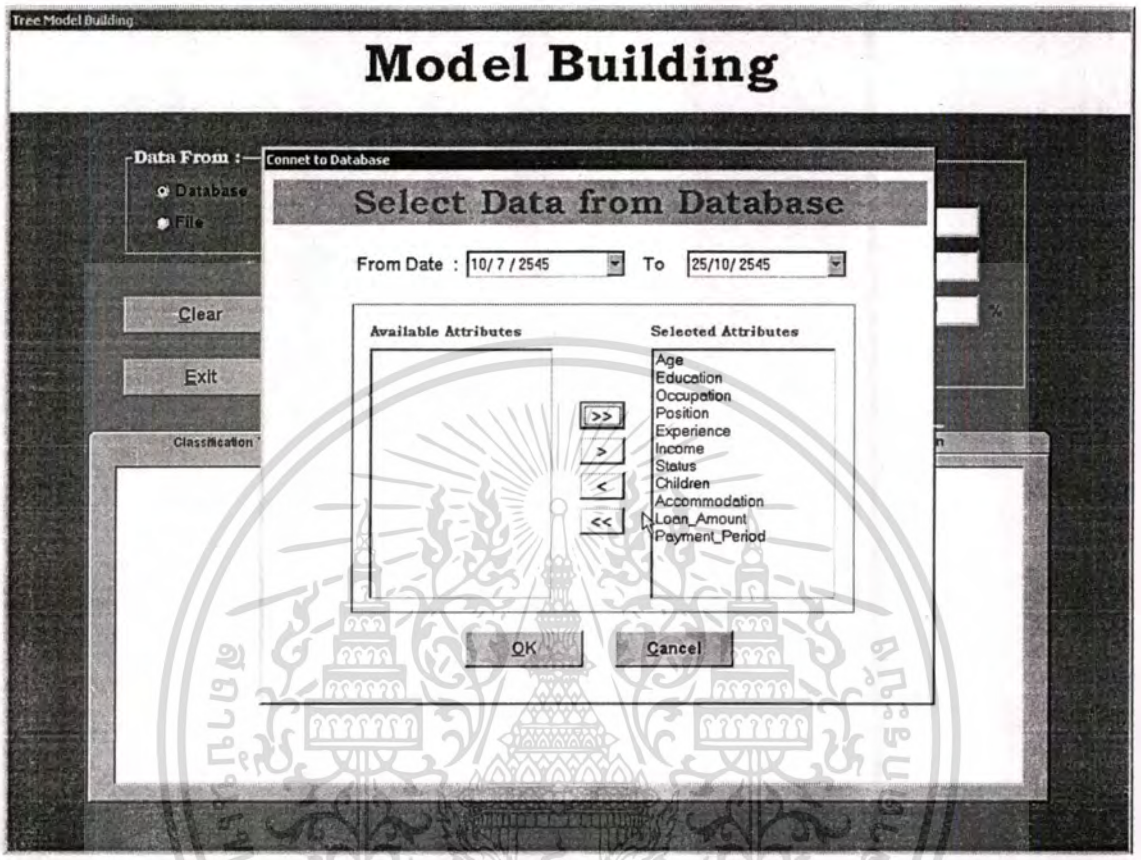
- ผู้ใช้สามารถเลือกได้ว่าจะนำข้อมูลจาก Database หรือ File มาใช้ในการสร้าง Tree Model
- ปุ่ม Select Data มีไว้สำหรับนำข้อมูลเข้า
- ปุ่ม Build Model มีไว้สำหรับสร้าง Classification Tree Model
- ปุ่ม Save Model มีไว้สำหรับบันทึก Tree Model ที่ได้จากการประมวลผล
- ปุ่ม Clear มีไว้สำหรับ Clear หรือ Reset หน้าจอ
- ปุ่ม Exit มีไว้สำหรับออกจากหน้าจอนี้



รูปที่ 4.5 หน้าจอส่วนที่ใช้ในการเลือกข้อมูลจาก Database แสดงส่วนที่ต้องทำการเลือกช่วงวันที่ของข้อมูล

หน้าจอนี้จะปรากฏเมื่อเลือกที่จะนำข้อมูลเข้าจาก Database และคลิก Select Data มีไว้สำหรับทำการเลือกข้อมูลที่มีอยู่ใน Database

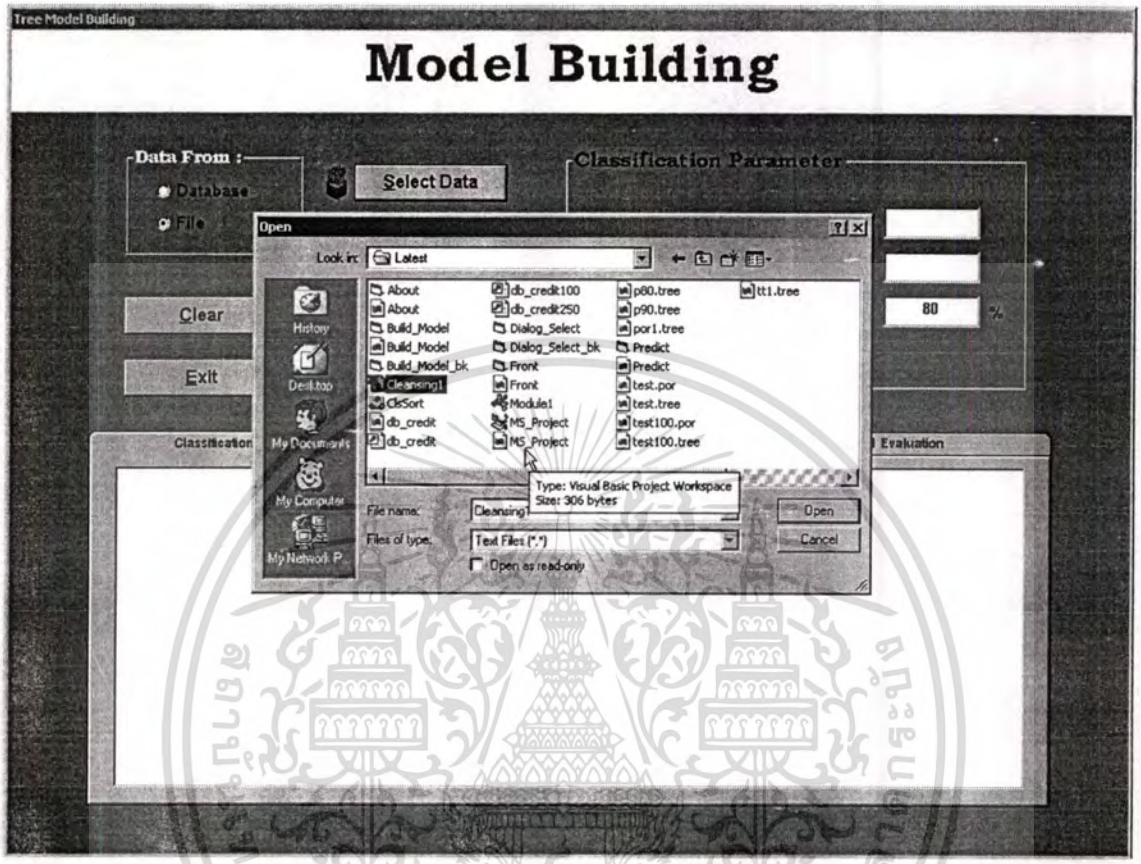
- จะต้องทำการเลือกช่วงวันที่ของข้อมูลที่มีอยู่ว่าต้องการข้อมูลในช่วงวันที่ใด



รูปที่ 4.6 หน้าจอส่วนที่ใช้ในการเลือกข้อมูลจาก Database
แสดงส่วนที่ต้องทำการเลือก Attribute ของข้อมูล

หน้าจอนี้จะปรากฏเมื่อเลือกที่จะนำข้อมูลเข้าจาก Database และคลิก Select Data มีไว้สำหรับทำการเลือกข้อมูลที่มีอยู่ใน Database

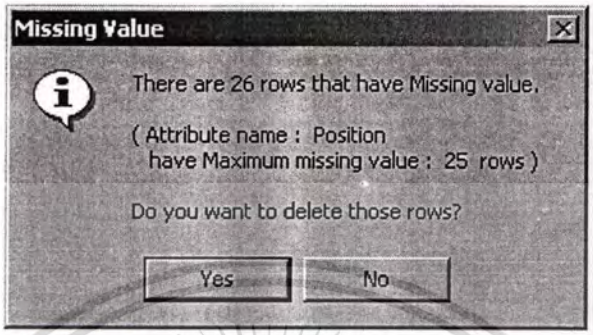
- จะต้องทำการเลือก Attribute ของข้อมูลที่จะนำไปสร้าง Classification Tree Model โดยสามารถเลือกได้ว่าต้องการทั้งหมดหรือต้องการเพียงบางส่วน



รูปที่ 4.7 หน้าจอส่วนที่ใช้ในการเลือกข้อมูลที่ต้องการจาก File

หน้าจอนี้จะปรากฏเมื่อเลือกที่จะนำข้อมูลเข้าจาก File และคลิก Select Data มีไว้สำหรับทำการเลือกข้อมูลจาก Text file ที่เตรียมไว้

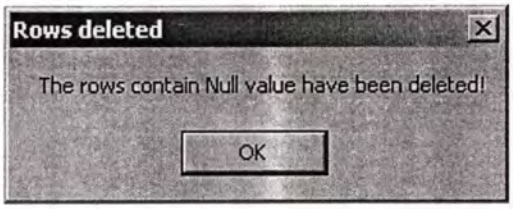
- จะต้องทำการเลือก Text file ที่มีการคั่นระหว่างข้อมูลด้วย " , "
  (ตาม Fomat ที่ได้แสดงในรูปที่ 4.2)



รูปที่ 4.8 Message Box ที่แสดงขึ้นเมื่อปรากฏว่ามี Missing Value

จะปรากฏ Message Box ดังรูปด้านบน เมื่อข้อมูลที่ทำการเลือกมามี Missing Value ปะปนอยู่ และถามยืนยันว่าต้องการลบ Row เหล่านั้นหรือไม่

- ระบบจะบอกจำนวน row ทั้งหมดที่มี Missing Value อยู่
- ระบบจะแสดงชื่อของ Attribute ที่มี Missing Value มากที่สุดพร้อมทั้งบอกจำนวน (เพื่อผู้ใช้อาจจะกลับไปทำการเลือกข้อมูลใหม่โดยตัด Attribute นั้นออก ไม่นำมาใช้ในการสร้าง Tree Model หาก Attribute นั้นๆ มีจำนวนของ Missing Value มากเกินไป)
- ผู้ใช้สามารถเลือกได้ว่าต้องการลบ record ทั้งหมดที่มี Missing Value ปะปนอยู่ออกไปจากระบบหรือไม่

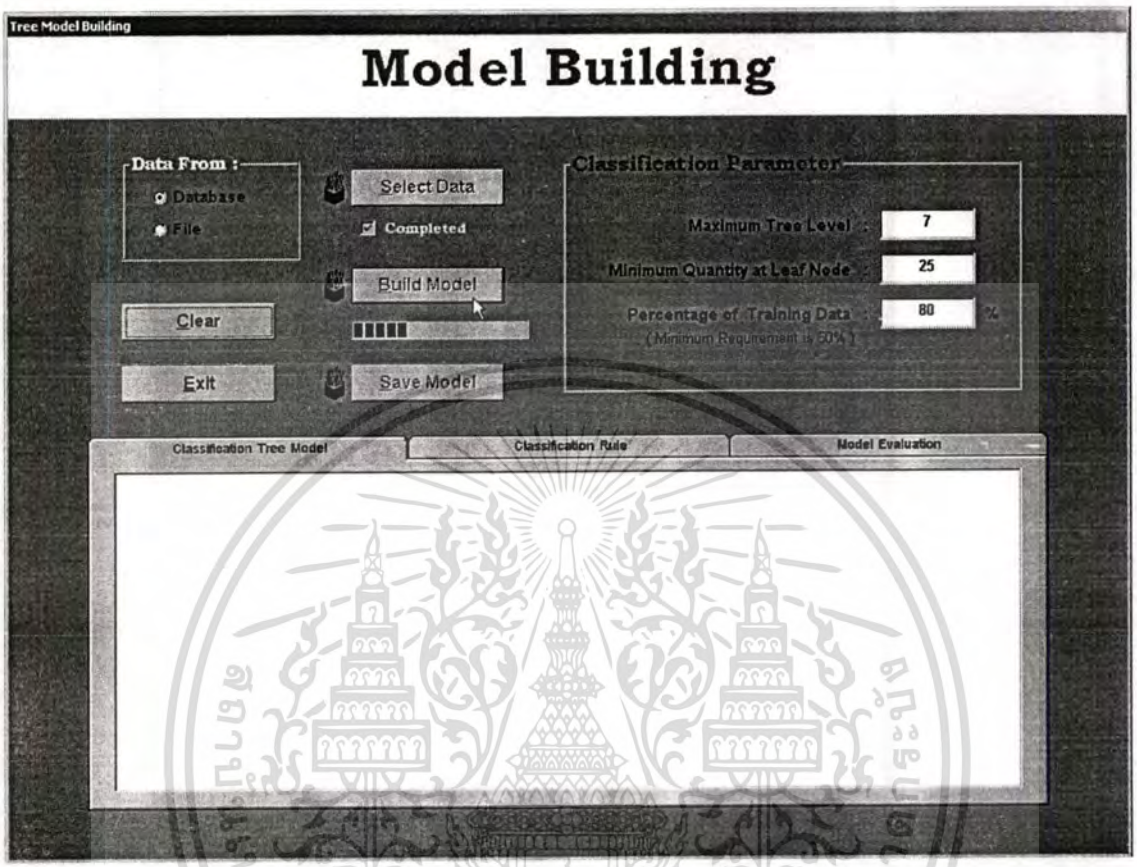


รูปที่ 4.9 Message Box ที่แสดงขึ้นเมื่อผู้ใช้ตอบในรูปที่ 4.8 ว่า "Yes"

จะปรากฏ Message Box ดังรูปด้านบน เมื่อตอบตกลงที่จะทำการลบ Row ทั้งหมดที่มี Missing Value ปะปนอยู่ เพื่อยืนยันว่าข้อมูลดังกล่าวได้ถูกลบออกไปเรียบร้อยแล้ว

รูปที่ 4.10 Message Box ที่แสดงขึ้นเมื่อผู้ใช้ตอบในรูปที่ 4.8 ว่า "No"

จะปรากฏ Message Box ดังรูปด้านบน เมื่อตอบว่าจะไม่ให้ทำการลบ Row ที่มี Missing Value โดยระบบจะกล่าวเตือนว่าให้ไปจัดการข้อมูลเหล่านั้นใน Database ก่อน หรือไม่ก็กลับไปทำการเลือกข้อมูลใหม่อีกครั้ง

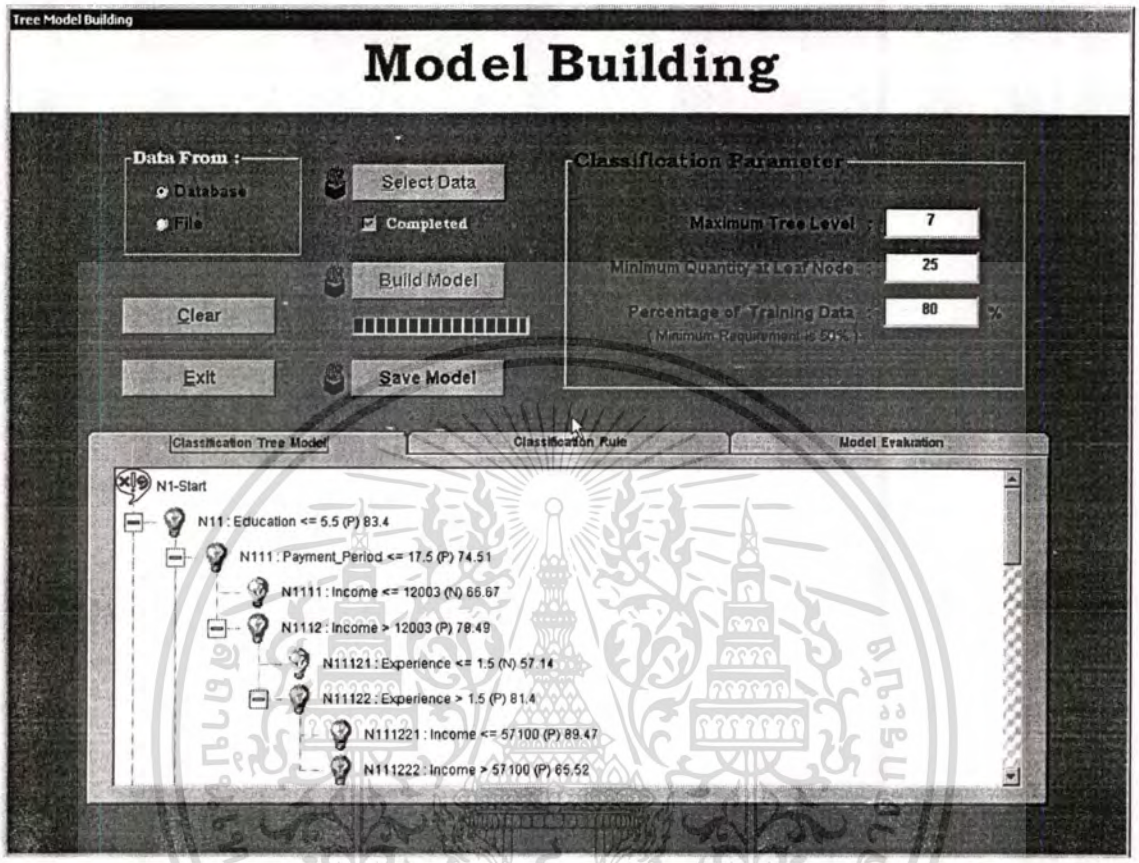


รูปที่ 4.11 หน้าจอขณะที่ระบบกำลังสร้าง Classification Tree Model

หากได้ทำการเลือกข้อมูลนำเข้าเรียบร้อยแล้ว ขั้นตอนต่อไปคือต้องทำการใส่ค่า Parameter 3 ค่า ดังต่อไปนี้

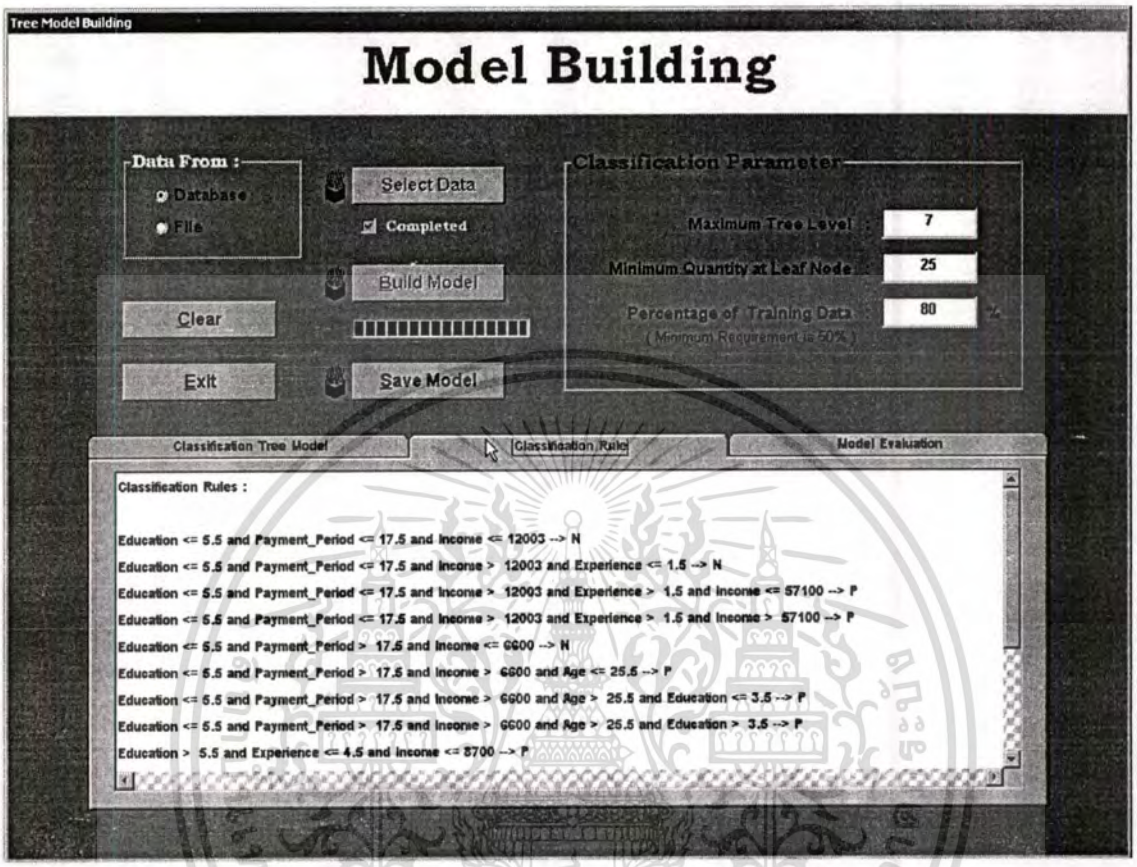
- Maximum Tree Level : จำนวนระดับชั้นสูงสุดของ Tree ที่ต้องการ
- Minimum Quantity at Leaf Node : จำนวนข้อมูลอย่างน้อยที่สุดที่ต้องมีในแต่ละ Node
- Percentage of Training Data : จำนวนเปอร์เซ็นต์ของข้อมูลที่ต้องการนำมาใช้ในการสร้าง Tree Model ซึ่งต้องมีค่าไม่น้อยกว่า 50% (ส่วนที่เหลือจะนำไปใช้ในการวัดความถูกต้องของ Model)

เมื่อทำการใส่ค่า Parameter ครบถ้วนแล้ว ให้คลิกที่ปุ่ม Build Model ระบบก็จะนำข้อมูลทั้งหมดไปประมวลผลเพื่อทำการสร้าง Tree Model ดังแสดงในรูปที่ 4.11



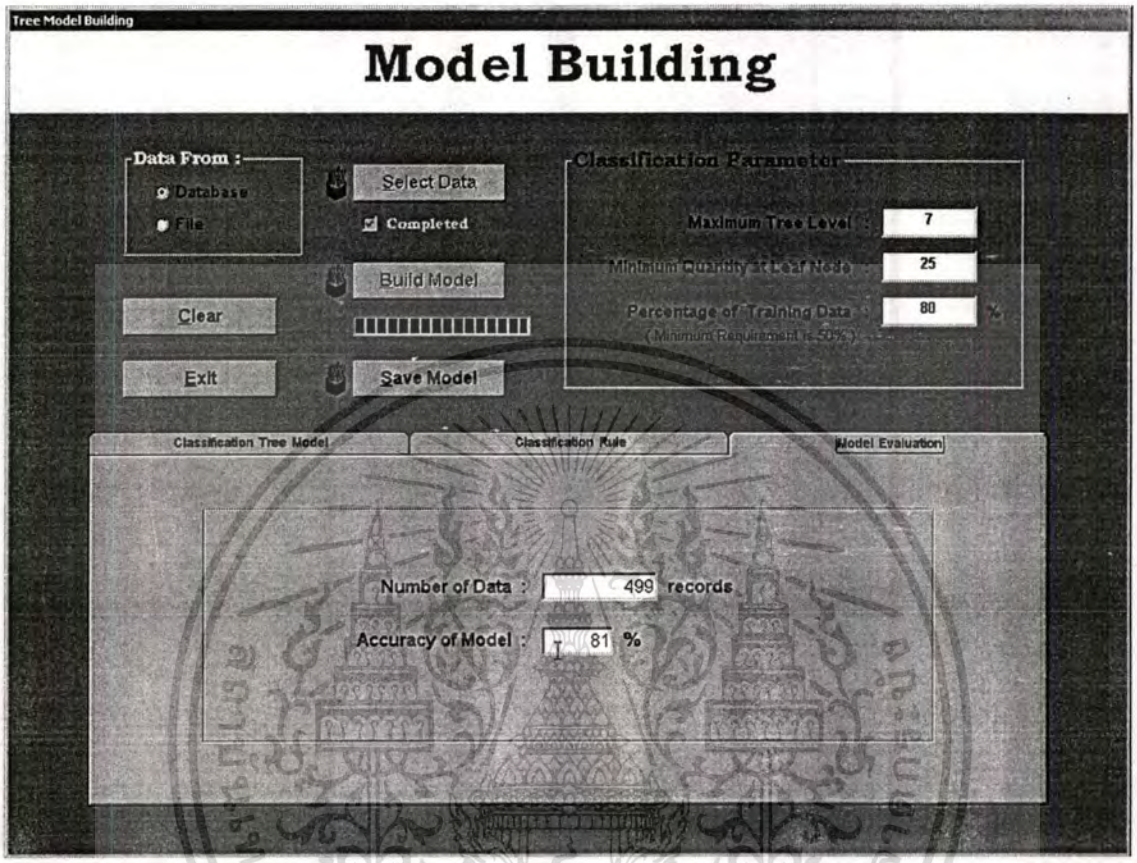
รูปที่ 4.12 หน้าจอเมื่อระบบประมวลผลเสร็จเรียบร้อยแล้ว
แสดงในส่วนของ Classification Tree Model ที่ได้

เมื่อ Progress bar ที่อยู่ใต้ปุ่ม Build Model ขึ้นมาจนเต็ม หมายถึง Tree Model ได้ถูกสร้างเสร็จเรียบร้อยแล้ว และแสดง Model ที่ได้ในส่วน Tab ของ Classification Tree Model



รูปที่ 4.13 หน้าจอเมื่อระบบประมวลผลเสร็จเรียบร้อยแล้ว แสดงในส่วนของ Classification Rule ที่ได้

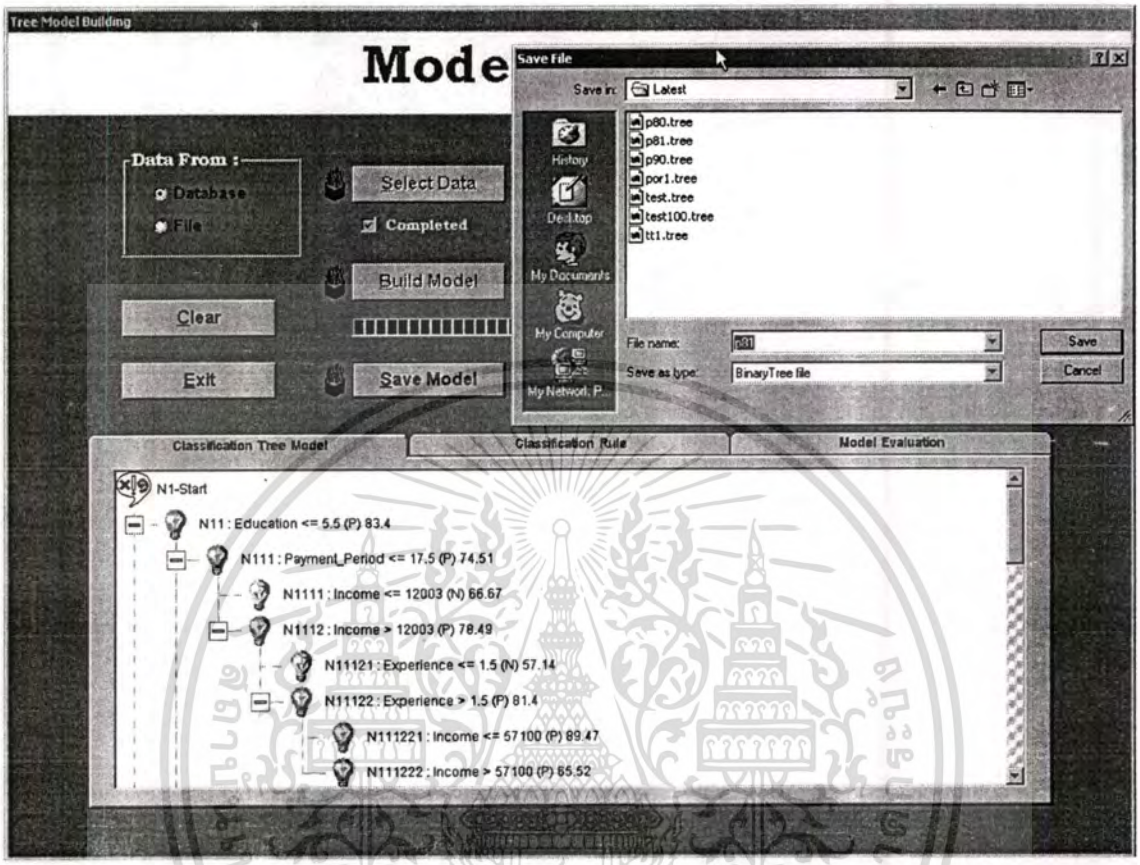
เมื่อ Progress bar ที่อยู่ใต้ปุ่ม Build Model ขึ้นมาจนเต็ม หมายถึง Tree Model ได้ถูกสร้างเสร็จเรียบร้อยแล้ว และแสดงผลลัพธ์ที่ได้จากการแปลความหมายของ Tree Model ในส่วน Tab ของ Classification Rule



รูปที่ 4.14 หน้าจอเมื่อระบบประมวลผลเสร็จเรียบร้อยแล้ว แสดงในส่วนของ Model Evaluation

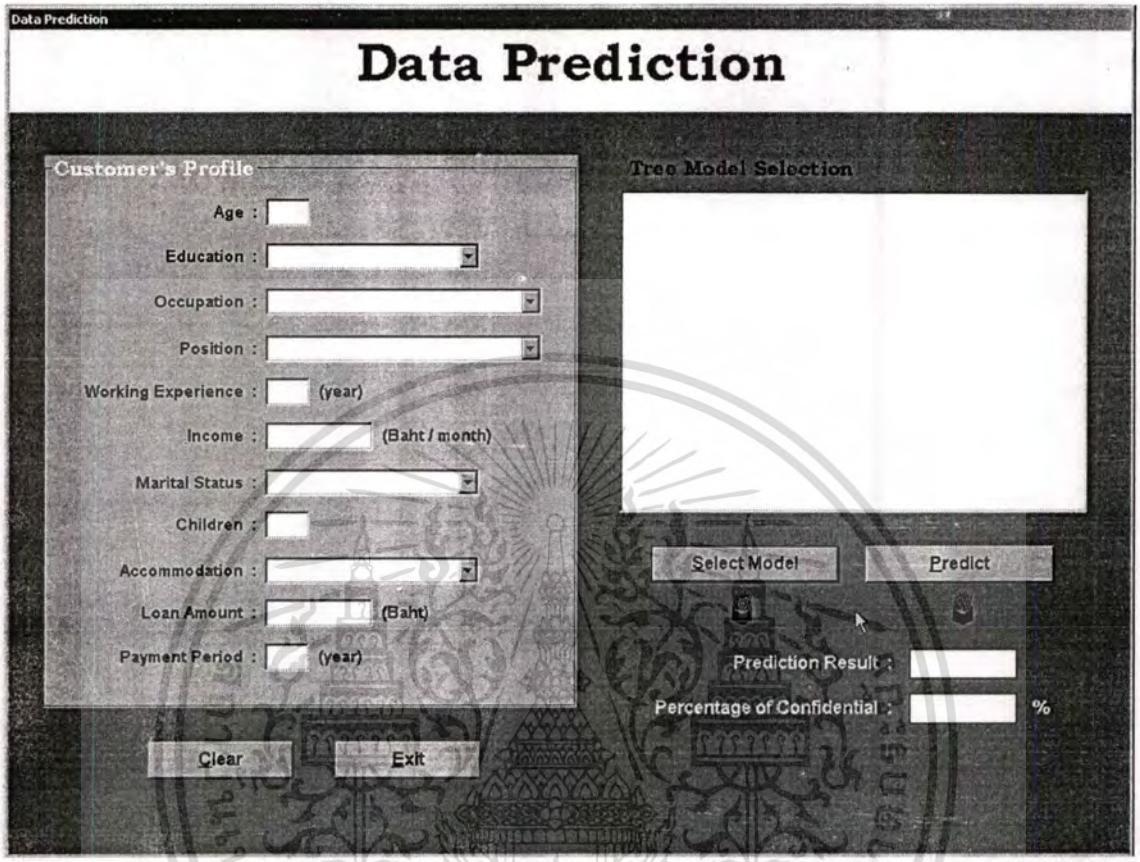
เมื่อ Progress bar ที่อยู่ใต้ปุ่ม Build Model ขึ้นมาจนเต็ม หมายถึง Tree Model ได้ถูกสร้างเสร็จเรียบร้อยแล้ว และแสดงผลความถูกต้องของ Model ในส่วน Tab ของ Model Evaluation ซึ่งจะแสดงข้อมูล 2 ส่วน คือ

- จำนวนข้อมูลที่ใช้ในการสร้าง Tree Model
- Accuracy ของ Model ที่ได้ ว่ามีความน่าเชื่อถือกี่เปอร์เซ็นต์



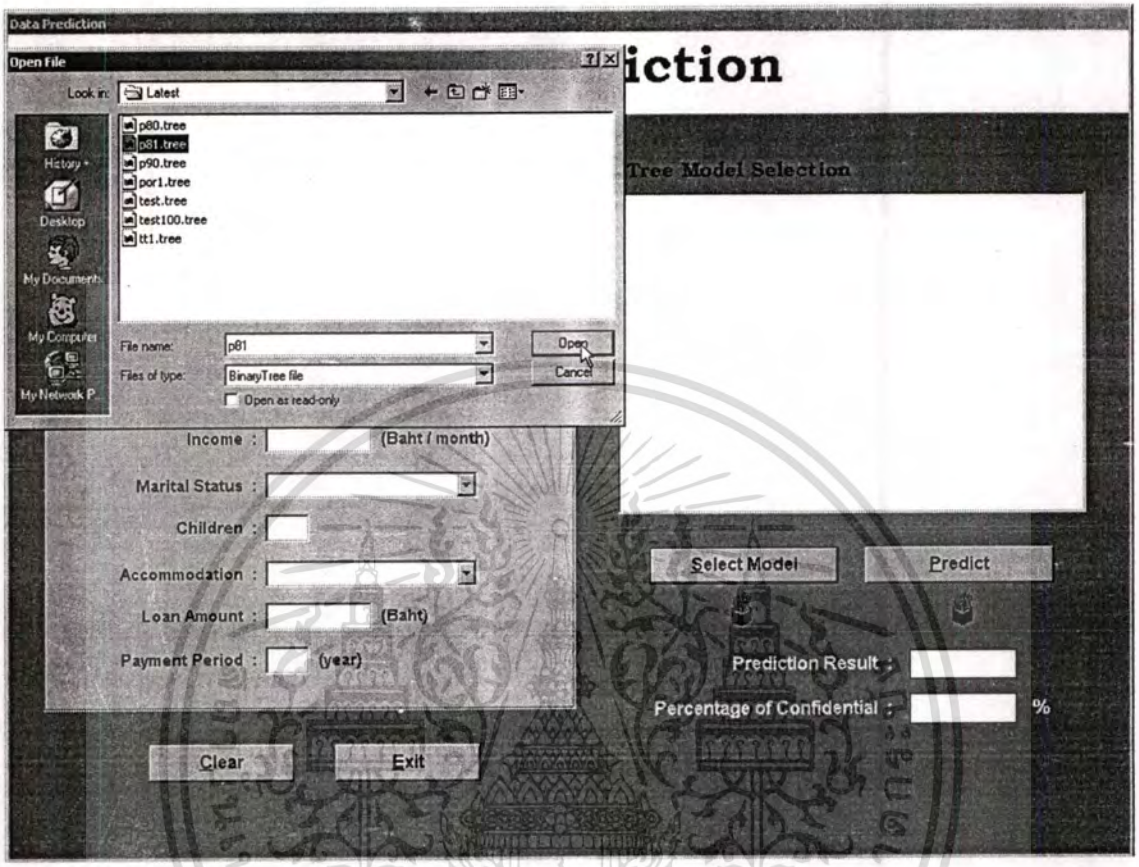
รูปที่ 4.15 หน้าจอแสดงการบันทึก Classification Tree Model

เมื่อ Tree Model ถูกสร้างเสร็จเรียบร้อยแล้ว ผู้ใช้สามารถบันทึก Tree Model ที่ได้เก็บไว้ในรูปแบบของไฟล์ *.tree ดังรูป 4.15 เพื่อนำไปใช้ในการทำนายข้อมูลต่อไป



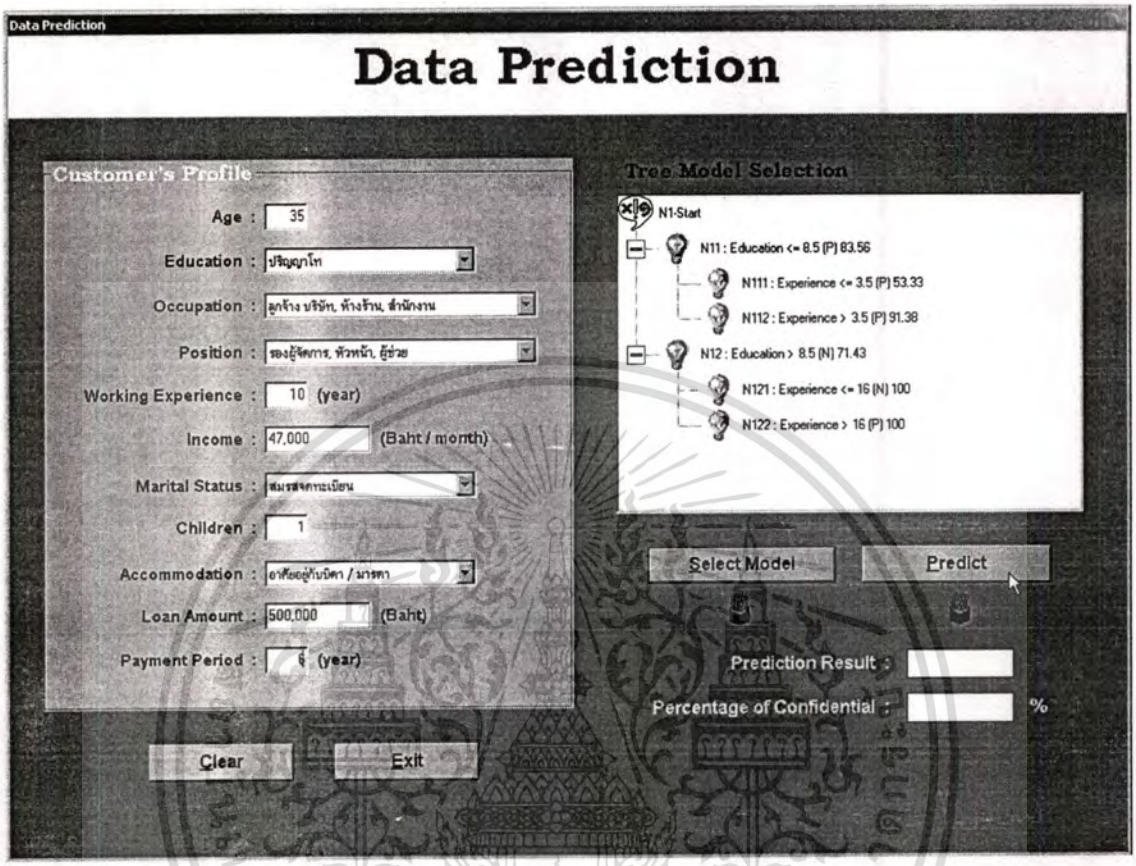
รูปที่ 4.16 หน้าจอส่วนที่ใช้ในการทำนายข้อมูล

หน้าจอนี้มีไว้สำหรับใช้ทำนายข้อมูลลูกค้าที่มาขอสินเชื่อ ว่าสมควรให้อนุมัติหรือไม่ และด้วยความเชื่อมั่นกี่เปอร์เซ็นต์ โดยทำนายจาก Classification Tree Model ที่สร้างไว้



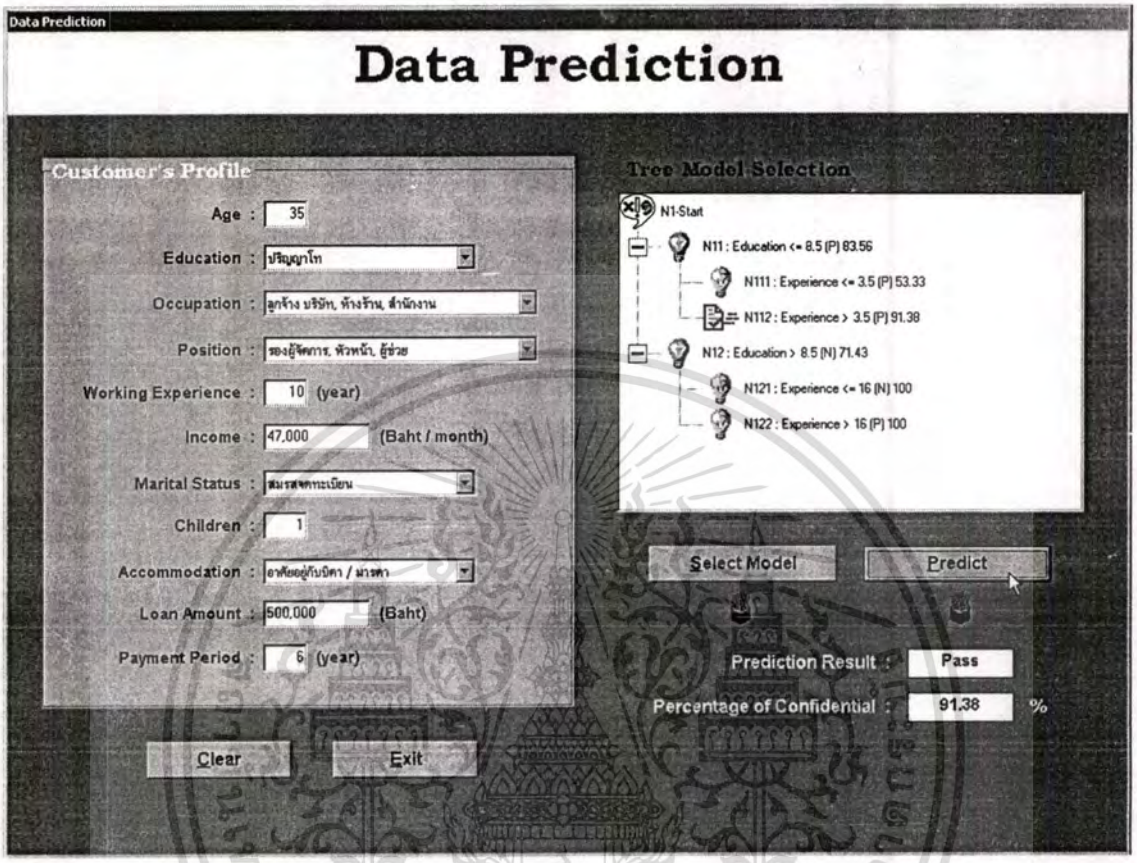
รูปที่ 4.17 หน้าจอแสดงการเลือก Classification Tree Model ที่จะนำมาใช้ในการทำนาย

ผู้ใช้จะต้องทำการเลือก Tree Model ที่จะใช้ในการทำนายข้อมูลก่อน จากไฟล์ *.tree ที่ได้ทำการบันทึกไว้จากหน้าจอ Model Building



รูปที่ 4.18 หน้าจอเมื่อมีข้อมูลครบถ้วนพร้อมที่จะทำการทำนาย

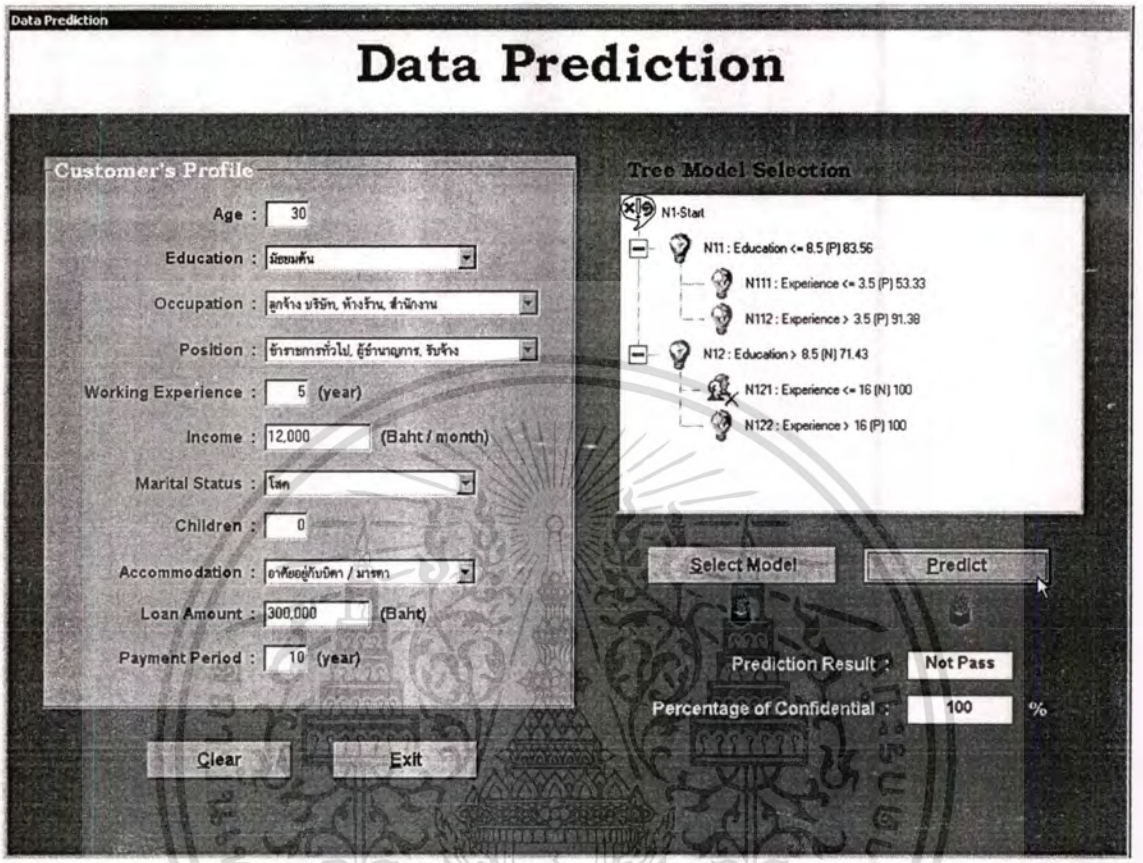
เมื่อทำการเลือก Tree Model เรียบร้อยแล้ว ให้นำข้อมูลลูกค้าที่มาขอสินเชื่อมาใส่ลงในส่วนของ Customer's Profile แล้วคลิกที่ปุ่ม Predict เพื่อทำการทำนายผล



รูปที่ 4.19 ตัวอย่างของผลลัพธ์ที่ถูกทำนายว่าผ่าน

ตัวอย่างของผลลัพธ์ที่ได้หากข้อมูลของผู้ขอสินเชื่อได้รับการทำนายจาก Classification Tree Model ว่าข้อมูลลูกค้าดังกล่าวมีความน่าจะเป็นว่าควรจะอนุมัติสินเชื่อให้

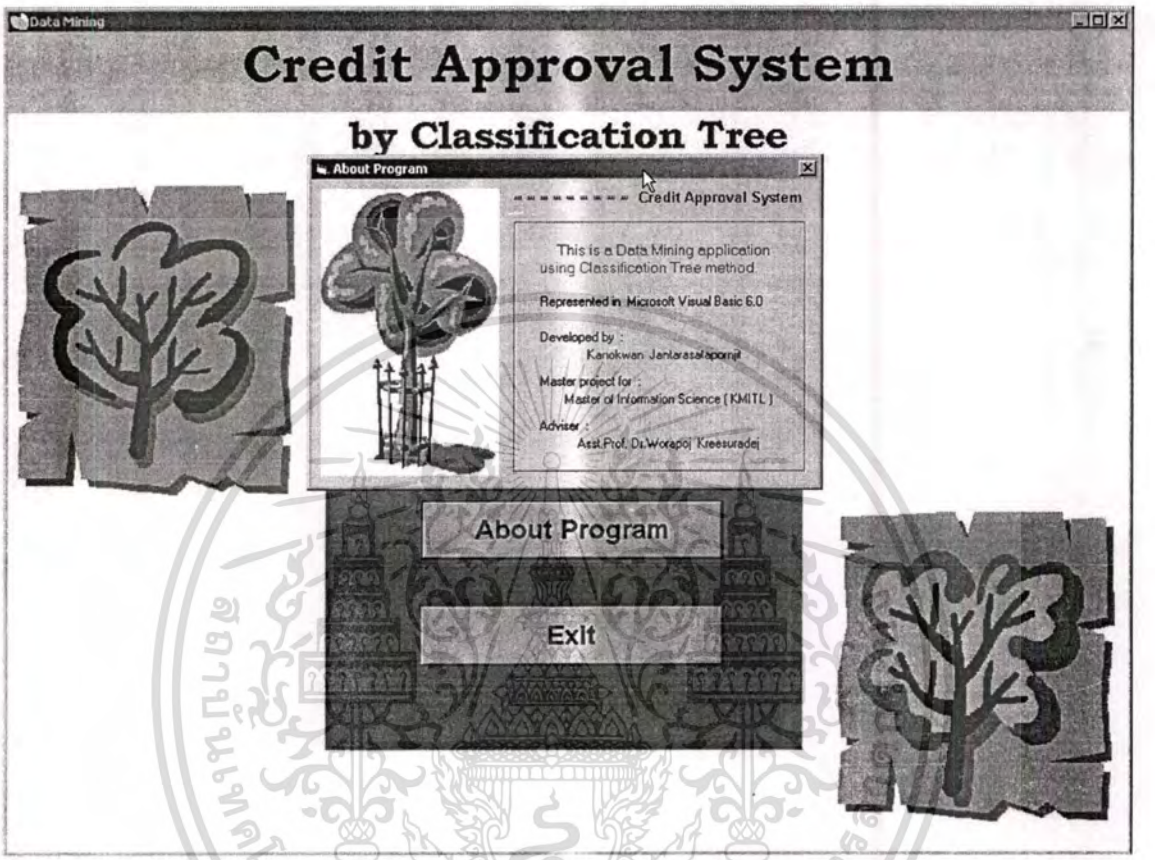
- ผลของการทำนายจะแสดงคำว่า “Pass” และแสดงค่าว่า **ผ่าน** ด้วยความเชื่อมั่นกี่เปอร์เซ็นต์



รูปที่ 4.20 ตัวอย่างของผลลัพธ์ที่ถูกทำนายว่าไม่ผ่าน

ตัวอย่างของผลลัพธ์ที่ได้หากข้อมูลของผู้ขอสินเชื่อได้รับการทำนายจาก Classification Tree Model ว่าข้อมูลลูกค้าดังกล่าวมีความน่าจะเป็นว่าไม่ควรจะอนุมัติสินเชื่อให้

- ผลของการทำนายจะแสดงคำว่า "Not Pass" และแสดงค่าว่า **ไม่ผ่าน** ด้วยความเชื่อมั่นกี่เปอร์เซ็นต์



รูปที่ 4.21 หน้าจอแสดงส่วนที่เรียกดูรายละเอียดที่มาของโปรแกรม

เมื่อคลิกเลือกปุ่ม About Program จะมีหน้าจอขึ้นมาแสดงถึงรายละเอียดของระบบที่ทำการพัฒนา

4.5 การประเมินผลการทำงานของระบบ

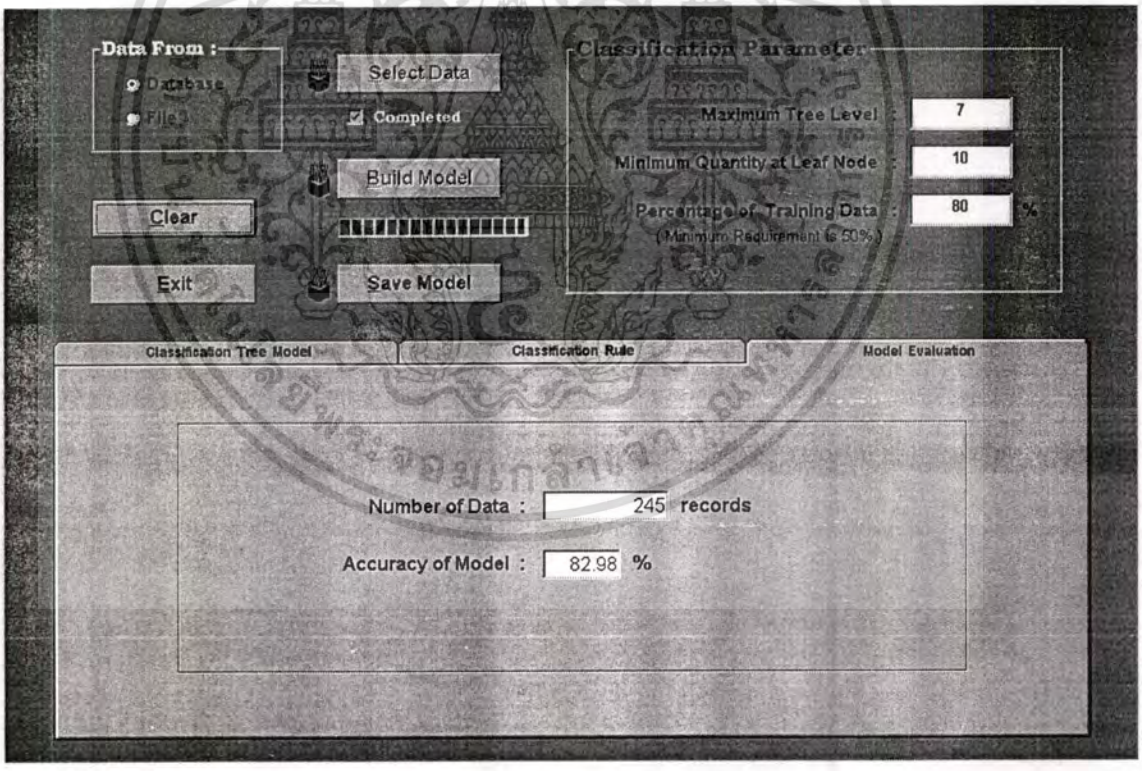
จากชุดข้อมูลที่นำมาทดสอบ ได้ทำการ Random และแบ่งข้อมูลออกเป็น 2 ส่วน คือ

- Training Data จำนวน 80%
- Testing Data จำนวน 20%

- ข้อมูลทดสอบชุดที่ 1

ประกอบด้วยข้อมูลจำนวน	245	records
กำหนด Tree Level ที่มากที่สุด	7	level
จำนวนข้อมูลอย่างน้อยในแต่ละ Node	10	records

ได้ผลลัพธ์ Accuracy of Model เป็น 82.98%

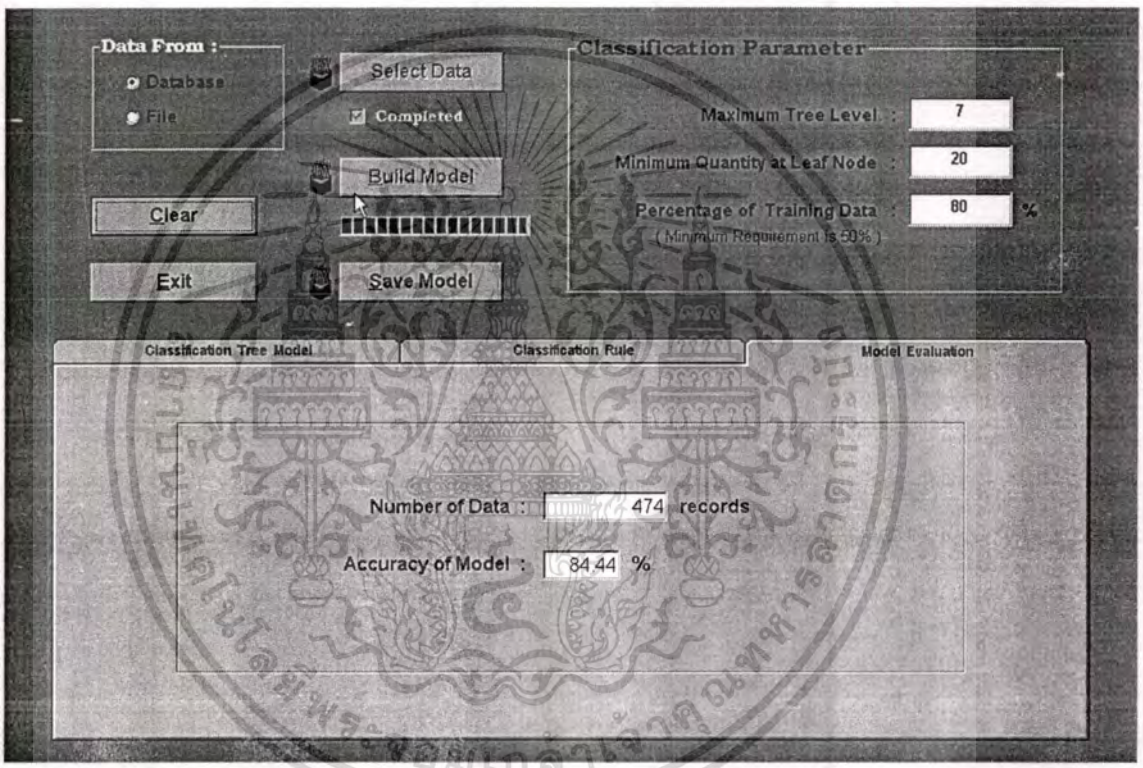


รูปที่ 4.22 ผลการทดสอบข้อมูลชุดที่ 1

- ข้อมูลทดสอบชุดที่ 2

ประกอบด้วยข้อมูลจำนวน	474	records
กำหนด Tree Level ที่มากที่สุด	7	level
จำนวนข้อมูลอย่างน้อยในแต่ละ Node	20	records

ได้ผลลัพธ์ Accuracy of Model เป็น 84.44%



รูปที่ 4.23 ผลการทดสอบข้อมูลชุดที่ 2

จากการทดสอบการทำงานของระบบ ในส่วนของการสร้าง Classification Tree Model ซึ่งได้ทำการตรวจสอบผลการทำงานด้วยข้อมูลทดสอบ 2 ชุด พบว่าระบบนี้สามารถทำงานได้เป็นที่น่าพอใจ และสามารถสร้าง Tree Model ที่มีค่า Accuracy ค่อนข้างสูง ส่วนความผิดพลาดที่เกิดขึ้นประมาณ 10 กว่าเปอร์เซ็นต์นั้นอาจเกิดจากจำนวนข้อมูลที่มีไม่มากพอ รวมถึงอาจเกิดจาก Noisy Data บางส่วนที่มีอยู่ในข้อมูล การทำงานส่วนอื่น เช่น เวลาที่ใช้ในการประมวลผลข้อมูลถือว่าค่อนข้างเร็ว แต่ก็อาจมีปัญหาบางอย่างเกิดขึ้นในระบบ เช่น ในกรณีเมื่อข้อมูลที่ได้ทำการเลือกมามีข้อมูลอยู่ใน Class ใด Class

หนึ่งมากจนเกินไป และทำการกำหนดให้ Tree Level น้อยเกินไป ก็จะทำให้ไม่สามารถเห็นลักษณะของข้อมูลของ Class ที่มีจำนวนน้อยนั้นได้ เป็นต้น

ส่วนการทำงานในส่วนของ Prediction นั้น ก็ขึ้นอยู่กับ Model ว่ามีความน่าเชื่อถือมากน้อยเพียงใด และข้อมูลที่นำมาทำนายก็จะต้องมีโครงสร้างแบบเดียวกันกับ Model ที่เลือกใช้ด้วยเช่นกัน

บทที่ 5
สรุปผลงาน

ในบทนี้จะกล่าวสรุปถึงผลการศึกษา และประเมินถึงระบบที่ได้ทำการพัฒนาทั้งในข้อดีและข้อเสียต่างๆ รวมถึงข้อเสนอแนะเพื่อเป็นแนวทางในการพัฒนาระบบให้ทำงานได้อย่างมีประสิทธิภาพมากขึ้น

5.1 สรุปผลการศึกษา

โครงการพัฒนาระบบงานฉบับนี้มีวัตถุประสงค์หลักคือ เพื่อที่จะศึกษาหลักการทำงานของ Data Mining และนำเสนอการประยุกต์ใช้ Data Mining กับงานทางด้านธุรกิจ ซึ่งเทคนิคการทำงานของ Data Mining นั้นมีอยู่หลายรูปแบบที่สามารถเลือกใช้ได้ ขึ้นอยู่กับความเหมาะสมในแต่ละงาน โดยในระบบงานที่นำเสนอนี้ได้ใช้เทคนิคของการสร้างแบบจำลองพยากรณ์ (Predictive Modeling) และใช้วิธีของ Classification Tree ในการสร้าง Model เพื่อใช้สนับสนุนการตัดสินใจอนุมัติสินเชื่อเบื้องต้น โดยการนำ Tree Model ที่ได้ไปทำนายเครดิตของลูกค้าที่มาทำการขอสินเชื่อจากธนาคาร ว่ามีความน่าเชื่อถือมากน้อยเพียงใด เพื่อช่วยในการควบคุมดูแล และบริหารความเสี่ยงจากการให้สินเชื่อให้เป็นไปอย่างมีประสิทธิภาพ รวมทั้งเป็นแนวทางในการนำไปประยุกต์ใช้เพื่อพิจารณาปัจจัยอื่นๆ ที่มีผลต่อการดำเนินธุรกิจ

ระบบงานที่พัฒนาขึ้นนี้ได้เลือกใช้ SLIQ Algorithm ซึ่งเป็นหนึ่งใน Algorithm ของวิธี Classification Tree ที่มีประสิทธิภาพในการทำงานค่อนข้างสูง คือใช้เวลาในการประมวลผลน้อยกว่า Algorithm แบบเก่าๆ และสามารถรองรับการทำงานของข้อมูลจำนวนมากได้ โดยยังคงมีความถูกต้องในระดับที่ดี

การทำงานของระบบแบ่งเป็น 2 ส่วน คือ

- ส่วนแรกเป็นการสร้าง Classification Tree Model โดยใช้ข้อมูลในอดีตของลูกค้ามาทำการประมวลผลโดยผ่าน SLIQ Algorithm และได้ผลลัพธ์ออกมาในรูปของ Tree Model
- ส่วนที่สองเป็นการทำนายข้อมูล (Data Prediction) โดยใช้ Tree Model ที่ได้จากส่วนแรกมาทำนายเครดิตของลูกค้ารายใหม่ที่มาขอสินเชื่อ ว่ามีความน่าจะเป็นในการเห็นสมควรให้สินเชื่อหรือไม่เพียงใด

ซึ่งผลการทดสอบระบบออกมาอยู่ในระดับที่น่าพอใจ มีความผิดพลาดเกิดขึ้นไม่มากนัก แต่อย่างไรก็ดีระบบนี้เป็นระบบที่ใช้สนับสนุนการตัดสินใจอนุมัติสินเชื่อเบื้องต้นเท่านั้น ในความเป็นจริงแล้ว อาจยังต้องมีปัจจัยอีกหลายอย่างที่ต้องใช้ในการวิเคราะห์อย่างเฉพาะเจาะจงในสินเชื่อแต่ละประเภท หรืออาจต้องใช้ประสบการณ์ตรงจากการทำงานด้านนี้ในการพิจารณาร่วมด้วย ระบบนี้จึงเป็นเพียงแนวทางประกอบการตัดสินใจเพื่อให้ผู้ใช้พิจารณาได้อย่างรวดเร็วและสะดวกมากขึ้น

5.2 ข้อเสนอแนะ

ระบบที่ได้ทำการออกแบบไว้ยังไม่ค่อยยืดหยุ่นและสมบูรณ์มากนัก เนื่องจากมีข้อจำกัดอยู่หลายอย่าง และยังมีอยู่หลายส่วนที่ควรปรับปรุงแก้ไข เพื่อระบบมีความยืดหยุ่นและรองรับการทำงานจริงในองค์กรได้อย่างเหมาะสม ส่วนที่ควรแก้ไขเพิ่มเติมหากจะทำการพัฒนาต่อไปให้ใช้งานได้จริง มีดังนี้

1. ควรมีการเพิ่ม Function การใช้งานในส่วนของการ Cleansing ข้อมูล เพื่อกำจัด Noisy Data ออกไป ซึ่งจะช่วยทำให้ได้ผลลัพธ์ที่มีความถูกต้องมากขึ้น
2. ในส่วนของการทำนายข้อมูล ควรออกแบบให้ยืดหยุ่นและรองรับการใช้งานให้ได้มากกว่านี้ เนื่องจากอาจมีปัจจัยในการพิจารณาเพิ่มมากขึ้น
3. ควรเพิ่มจำนวนข้อมูลที่ใช้ในการทดสอบระบบให้มากขึ้น
4. อาจนำระบบนี้ไปประยุกต์ใช้ร่วมกับงานด้านอื่นๆ ที่เกี่ยวข้อง เพื่อเป็นประโยชน์ในการพัฒนาระบบการทำงาน

บรรณานุกรม

Cabena et al. 1998. **Discovery Data Mining From Concept to Implementation.** New Jersey : Prentice Hall.

Han Jia and Kamber Micheline. 2001. **Data Mining : Concepts and Techniques.** California : Morgan Kaufmanns.

J.C. Shafer et al. 1996. **"SPRINT: A Scalable Parallel Classifier for Data Mining."** In Proc. of the 22th VLDB Conference.

Michael J.A. Berry and Gordon Linoff. 1997. **Data Mining Techniques.** Canada : John Wiley & Sons, Inc.

M. Mehta et al. 1996. **"SLIQ : A fast Scalable Classifier for Data Mining."** 18-32. In Proc. of the fifth Int'l Conference on Extending Database Technology.

Sholom M. Weiss and Nitin Indurkhya. 1998. **Predictive Data Mining.** San Fransisco : Morgan Kaufmanns.

Two Crows Corporation. 1999. **Introduction to Data Mining and Knowledge Discovery** [Online]. Available : http://www.twocrows.com/

ประวัติผู้เขียน

ชื่อ นามสกุล	นางสาว กนกวรรณ จันทรสถาพรจิต
วัน เดือน ปีเกิด	11 เมษายน พ.ศ. 2520
สถานที่เกิด	กรุงเทพมหานคร
วุฒิการศึกษา	ปริญญาตรีสถิติศาสตรบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศเพื่อธุรกิจ ภาควิชาสถิติ คณะพาณิชยศาสตร์และการบัญชี จุฬาลงกรณ์มหาวิทยาลัย
การทำงาน	โปรแกรมเมอร์ ธนาคารไทยพาณิชย์ จำกัด (มหาชน)