รายงานวิจัยฉบับสมบูรณ์

โครงการ อัลกอริทึมเรียนรู้สำหรับสร้างตัวจำแนกประเภทมีความถูกต้องสูง

Learning Algorithms for Constructing Highly Accurate Classifiers

โดย บุญเสริม กิจศิริกุล

ตุลาคม 2552

สัญญาเลขที่ RSA4980003

รายงานวิจัยฉบับสมบูรณ์

โครงการ อัลกอริทึมเรียนรู้สำหรับสร้างตัวจำแนกประเภทมีความถูกต้องสูง
Learning Algorithms for Constructing Highly Accurate Classifiers

บุญเสริม กิจศิริกุล
ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์
จุฬาลงกรณ์มหาวิทยาลัย

สนับสนุนโดยสำนักงานกองทุนสนับสนุนการวิจัย

กิตติกรรมประกาศ

ผู้วิจัยขอขอบคุณสำนักงานกองทุนสนับสนุนการวิจัยที่ได้ให้ทุนตลอดระยะเวลา 3 ปี (30 สค. 2549 – 29 สค. 2552) สำหรับโครงการ “อัลกอริทึมเรียนรู้สำหรับสร้างตัวจำแนกประเภทมีความถูกต้องสูง” และขอขอบคุณภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ที่ได้ให้โอกาส สถานที่ตลอดจนทรัพยากรต่างๆ ที่ใช้ในงานวิจัย ขอขอบคุณผู้ช่วยวิจัยและผู้มีส่วนร่วมในโครงการนี้ทุกท่าน

บุญเสริม กิจศิริกุล

ตุลาคม 2552

บทคัดย่อ

รหัสโครงการ: RSA4980003

ชื่อโครงการ: อัลกอริทึมเรียนรู้สำหรับสร้างตัวจำแนกประเภทมีความถูกต้องสูง

ชื่อนักวิจัย: นายบุญเสริม กิจศิริกุล

E-mail Address: boonserm.k@chula.ac.th

ระยะเวลาโครงการ: 30 สค. 2549 – 29 สค. 2552

วัตถุประสงค์ของงานวิจัยนี้เพื่อพัฒนาฟังก์ชันเคอร์เนลที่ยืดหยุ่นมากพอที่จะปรับตัวเองให้เหมาะกับงานที่ทำและเพื่อพัฒนาเทคนิคกลุ่มก้อนตัวจำแนกที่สามารถลดไบแอสและแวเรียนซ์ เราได้นำเสนอการพัฒนาฟังก์ชันเคอร์เนลใหม่สำหรับซัพพอร์ตเวกเตอร์แมชชีน (เอสวีเอ็ม) เอสวีเอ็มเป็นอัลกอริทึมเรียนรู้ที่มีพื้นฐานมาจากแนวคิดเรื่องการลดความเสี่ยงเชิงโครงสร้างให้ต่ำสุด เอสวีเอ็มถูกใช้ในงานประยุกต์ต่างๆ เช่น การรู้จำรูปแบบ การประมาณฟังก์ชัน เป็นต้น โดยปกติเอสวีเอ็มจะทำการแบ่งข้อมูลแบบเชิงเส้นในปริภูมิลักษณะโดยใช้เคอร์เนลที่นิยามขึ้น เคอร์เนลเหล่านี้จะส่งเวกเตอร์นำเข้าไปยังปริภูมิใหม่ที่มีมิติสูงมาก ซึ่งคาดหมายว่าการแบ่งเชิงเส้นจะทำได้ง่ายขึ้น จากนั้นระนาบหลายมิติเชิงเส้นที่ใช้แบ่งข้อมูลจะถูกค้นพบโดยการทำให้ระยะขอบระหว่างข้อมูลทั้งสองประเภทมีค่ามากสุด ดังนั้นความซับซ้อนของระนาบหลายมิติที่ใช้แบ่งแยกจะขึ้นกับคุณสมบัติของเคอร์เนลที่ใช้

ฟังก์ชันเคอร์เนลที่นิยมใช้มีหลายชนิด เช่น linear kernel, polynomial kernel, sigmoid kernel, RBF kernel เป็นต้น เคอร์เนลแต่ละประเภทจะเหมาะกับงานบางชนิดและเราจะต้องเลือกเคอร์เนลเองหรือใช้ความรู้ก่อนหน้าเลือกให้เหมาะกับงานที่ทำ เคอร์เนลอาร์บีเอฟเป็นเคอร์เนลแบบหนึ่งที่ประสบผลความสำเร็จในปัญหาหลายๆ ด้าน แต่ก็ยังมีข้อจำกัดอยู่บ้างในปัญหาที่ซับซ้อน ดังนั้นเราจึงนำเสนอการปรับปรุงประสิทธิภาพของการจำแนกประเภทข้อมูลโดยการรวมเคอร์เนลอาร์บีเอฟหลายๆ ตัวที่สเกลต่างๆ เคอร์เนลเหล่านี้จะถูกรวมกันแบบมีน้ำหนัก โดยที่น้ำหนักและความกว้างของเคอร์เนลอาร์บีเอฟและพารามิเตอร์สามัญของเอสวีเอ็มจะถูกเรียกรวมกันว่า ไฮเปอร์พารามิเตอร์ เราเสนอการใช้กลยุทธ์ทางวิวัฒนาการเพื่อกำหนดค่าเหมาะสมของไฮเปอร์พารามิเตอร์เหล่านี้ ผลการทดลองแสดงให้เห็นว่าเคอร์เนลอาร์บีเอฟแบบหลายสเกลที่นำเสนอให้ผลดีกว่าเคอร์เนลอาร์บีเอฟเดี่ยว เมื่อเอสวีเอ็มใช้เคอร์เนลอาร์บีเอฟแบบหลายสเกล เอสวีเอ็มสามารถเรียนรู้จากข้อมูลได้อย่างดี

เรายังได้ขยายวิธีการที่นำเสนอให้ใช้กับปัญหาการประมาณฟังก์ชัน (การถดถอย) เมื่อเอสวีเอ็มถูกประยุกต์ใช้กับปัญหาการถดถอย เราจะเรียกว่าซัพพอร์ตเวกเตอร์รีเกรซชัน (เอสวีอาร์) ในทำนองเดียวกับเอสวีเอ็ม เรานำเสนอที่จะปรับปรุงประสิทธิภาพของการถดถอยโดยใช้การรวมเคอร์เนลอาร์บีเอฟแบบหลายสเกลและใช้กลยุทธ์ทางวิวัฒนาการเพื่อกำหนดค่าเหมาะสมสำหรับไฮเปอร์พารามิเตอร์ ผลการทดลองแสดงให้เห็นว่าความสามารถของวิธีการที่นำเสนอวัดโดยค่าความผิดพลาดเฉลี่ยแบบสมมาตร (SMAPE) เอสวีอาร์ที่ใช้เคอร์เนลอาร์บีเอฟแบบหลายสเกลให้ค่า SMAPE เฉลี่ยต่ำสุดในการทดสอบ

กลุ่มก้อนตัวจำแนกเป็นเทคนิคอีกประเภทหนึ่งที่ใช้สำหรับสร้างตัวจำแนกประเภทมีความถูกต้องสูง เรานำเสนอการใช้กลุ่มก้อนตัวจำแนกแบบหลายระดับที่รวม Adaboost กับ Bagging เนื่องจาก Bagging สามารถลดแวเรียนซ์ของตัวเรียนรู้พื้นฐานได้อย่างมีประสิทธิภาพ กุญแจความสำเร็จของ Bagging คือตัวเรียนรู้พื้นฐานที่ต้องมีคุณสมบัติของไบแอสต่ำ ไม่ว่าจะมีแวเรียนซ์เท่าไรก็ตาม ในทางกลับกันเราพบว่าในหลายๆ การศึกษาวิจัย Adaboost แสดงให้เห็นว่าเป็นตัวเรียนรู้ที่มีไบแอสต่ำมาก กล่าวคือ Adaboost สามารถปรับตัวให้ตรงกับตัวอย่างฝึกได้ดีมาก ดังนั้นจึงเป็นที่น่าสนใจที่จะรวม Adaboost กับ Bagging โดยทำเป็นกลุ่มก้อนตัวจำแนกแบบหลายระดับ กลุ่มก้อนตัวจำแนกที่นำเสนอจะใช้ Adaboost เป็นตัวเรียนรู้พื้นฐานของ Bagging เหตุผลก็คือ Adaboost สามารถใช้ไบแอสต่ำแต่อาจมีแวเรียนซ์สูง ซึ่งจะถูกลดโดย Bagging อีกทีหนึ่ง เรายัง

ได้นำเสนอเครื่องมือวิเคราะห์สำหรับจินตทัศน์ผลของกลุ่มก้อนตัวจำแนกเมื่อใช้ตัวเรียนรู้พื้นฐาน เครื่องมือวิเคราะห์ที่นำเสนอจะตรวจสอบพฤติกรรมของตัวเรียนรู้แบบกลุ่มก้อน เครื่องมือนี้จะไม่ได้คำนวณไบแอส-แวเรียนซ์ของตัวเรียนรู้ แต่จะวิเคราะห์ผลกระทบของกลุ่มก้อนที่มีต่อตัวเรียนรู้พื้นฐานในระดับตัวอย่าง ด้วยวิธีการนี้ เราสามารถจะสังเกตการเปลี่ยนแปลงของค่าผิดพลาดเฉลี่ยของแต่ละตัวอย่างระหว่าง “ก่อน” และ “หลัง” ประยุกต์ใช้วิธีการกลุ่มก้อน ดังนั้นจะทำให้เราสามารถประเมินประสิทธิภาพของกลุ่มก้อนตัวจำแนกได้

คำหลัก: อัลกอริทึมเรียนรู้, ซัพพอร์ตเวกเตอร์แมชชีน, ซัพพอร์ตเวกเตอร์รีเกรชชัน, เคอร์เนล, ฟังก์ชันเรเดียลเบสิส, กลุ่มก้อนตัวจำแนก, ไบแอส-แวเรียนซ์

Abstract

Project Code: RSA4980003
Project Title: Learning Algorithms for Constructing Highly Accurate Classifiers
Investigator: Boonserm Kijsirikul
E-mail Address: boonserm.k@chula.ac.th
Project Period: August 30, 2006 – August 29, 2009

The objectives of this research are to develop new kernel functions that are flexible enough to adjust themselves to the task under consideration, and to develop ensemble techniques that are able to reduce both bias and variance. First, we propose to develop new kernel functions for Support Vector Machines (SVMs). SVMs are learning algorithms based on the idea of empirical risk minimization principle. They have been widely used in many applications such as pattern recognition and function approximation. Basically, SVM operates a linear separation in an augmented space by means of some defined kernels. These kernels map the input vectors into a very high dimensional space where linear separation is more likely. Then, a linear separating hyperplane is found by maximizing the margin between two classes in this space. Hence, the complexity of the separating hyperplane depends on the nature and the properties of the used kernel.

There are many types of kernel functions such as linear kernel, polynomial kernel, sigmoid kernel, and RBF kernel. Each kernel function is suitable for some tasks, and it must be chosen for the tasks under consideration by hand or using prior knowledge. The RBF kernel is a most successful kernel in many problems, but still has the restrictions in some complex problems. Therefore, we propose to improve the efficiency of classification by using the combination of RBF kernels at different scales. These kernels are combined by including weights. The weights, the widths of the RBF kernels, and regularization parameter of SVM are called *hyperparameters*. We propose to use the evolutionary strategy for determining the appropriate values of these hyperparameters. The experimental results show that the proposed multi-scale RBF kernels yield the better results than single RBF kernels. When SVM uses the multi-scale RBF kernel, it is able to learn from data very well.

We also extend our method to deal with the problem of function approximation (regression). When SVM is applied to the regression problem, it is called Support Vector Regression (SVR). Similar to SVM, we propose to improve the efficiency of the regression by using the combination of RBF kernels at different scales, and employ an evolutionary strategy for determining the appropriate values of the hyperparameters. The experimental results show the ability of the proposed method on symmetric mean absolute percentage error (SMAPE). SVR with the multi-RBF kernel yields the lowest average SMAPE on testing.

Ensemble is another technique for constructing highly accurate classifiers. We propose to use multi-level ensemble which combines Adaboost and Bagging. As Bagging can efficiently reduce the variance of it base learners, the key success of Bagging is the low-bias property of its base learners, no matter how much variance its base learners will have. On the other hand, in many studies, AdaBoost appears to be a very low bias learner; it fits training examples very well. Therefore, it is promising to combine AdaBoost and Bagging by constructing a multi-level ensemble. The proposed ensemble employs AdaBoost as base learners of Bagging. The reason of this combination is that AdaBoost can provide low bias but may introduce high variance that in turn can be reduced by Bagging. We also propose an analysis tool for visualizing the effects of an ensemble to its base learners. The proposed analysis tool investigates practical behaviors of an ensemble learner. The tool does not calculate the bias-variance of a learner, but instead the tool analyzes the effects of an ensemble method to its base learners *in the instance level*. By this approach we are able to see the change of mean errors of each instance between "*before*" and "*after*" applying the ensemble method, and thus we can properly evaluate the performance of the ensemble.

Keywords: Learning Algorithm, Support Vector Machines, Support Vector Regression, Kernels, Radial Basis Functions, Ensemble, Bias-Variance.

I. เนื้อหางานวิจัย

เทคนิคการเรียนรู้ของเครื่องจำนวนมากได้ถูกนำเสนอ เช่น การเรียนรู้ต้นไม้ตัดสินใจ นิวรอลเน็ตเวิร์ก การเรียนรู้กฎ การโปรแกรมตรรกะเชิงอุปนัย เป็นต้น เทคนิคบางวิธีเหมาะกับการเรียนรู้โมเดลที่ต้องการการทำความเข้าใจได้ด้วยมนุษย์ เช่น ต้นไม้ตัดสินใจ การเรียนรู้กฎ เป็นต้น ส่วนเทคนิคบางวิธีสร้างตัวจำแนกประเภทที่มีความถูกต้องสูงมากแต่อาจมีข้อด้อยที่ความรู้ที่เรียนได้ทำความเข้าใจได้ยาก ในงานวิจัยนี้เราเน้นเรื่องการสร้างตัวจำแนกประเภทที่มีความถูกต้องสูง

เทคนิคการเรียนรู้ที่ค่อนข้างใหม่วิธีหนึ่งในปัจจุบันก็คือซัพพอร์ตเวกเตอร์แมชชีน – เอสวีเอ็ม (Support Vector Machine – SVM) ประสิทธิภาพของเอสวีเอ็มได้รับการทดสอบในหลายๆ งานที่นำไปประยุกต์ใช้ เช่น การรู้จำรูปแบบ การทำนายอนุกรมเวลา เป็นต้น เอสวีเอ็มจะทำการแบ่งแยกเชิงเส้นของข้อมูลสองประเภท ในกรณีที่ข้อมูลสองประเภทแยกจากกันไม่ได้ในปริภูมินำเข้า (input space) เอสวีเอ็มจะใช้ฟังก์ชันเคอร์เนล (kernel function) ที่สอดคล้องกับเงื่อนไขของ Mercer [Vapnik, 1995] เพื่อส่งข้อมูลนำเข้าไปยังปริภูมิลักษณะที่มีมิติสูงขึ้น ซึ่งในปริภูมิใหม่นี้การแบ่งแยกข้อมูลอาจทำได้ง่ายขึ้น จากนั้นเอสวีเอ็มจะทำการแบ่งแยกข้อมูลแบบเชิงเส้นในปริภูมิใหม่นี้โดยการทำให้ระยะขอบระหว่างข้อมูลจากสองประเภทมีค่ากว้างสุด ดังนั้นประสิทธิภาพของการจำแนกประเภทจะขึ้นกับเคอร์เนลที่เลือกใช้ อย่างไรก็ดีเคอร์เนลแต่ละฟังก์ชันจะเหมาะกับงานบางชนิดเท่านั้น และเราต้องเลือกเองด้วยมือ เพื่อแก้ปัญหาการเลือกเคอร์เนลโดยคน งานวิจัยนี้เราจึงนำเสนอฟังก์ชันเคอร์เนลใหม่ที่สามารถปรับพารามิเตอร์ได้เองอย่างอัตโนมัติให้เหมาะกับงานที่ทำ

วิธีการสร้างตัวจำแนกประเภทมีความถูกต้องสูงอีกวิธีหนึ่งคือกลุ่มก้อนตัวจำแนก (ensemble) ซึ่งก็เป็นวิธีการที่ประสบความสำเร็จสูง [Breiman, 1996; Freund & Schapire, 1996] กลุ่มก้อนตัวจำแนกเป็นการรวมตัวจำแนกพื้นฐานหลายๆ ตัวเข้าด้วยกัน โดยที่ตัวจำแนกพื้นฐานแต่ละตัวมีความถูกต้องพอควร ประสิทธิภาพที่ดีของกลุ่มก้อนตัวจำแนกสามารถอธิบายได้ด้วยทฤษฎีของความเอนเอียง-ความแปรปรวน (bias-variance theory) [Domingos, 2000] ทฤษฎีนี้แยกความผิดพลาดทั่วไปออกเป็นสามส่วนคือไบแอส, แวเรียนซ์ และนอยส์ อัลกอริทึมที่เรียนรู้มโนทัศน์ในรูปแบบที่จำกัดเกินไปจะมีไบแอสมาก และอัลกอริทึมที่เรียนรู้มโนทัศน์ที่แตกต่างอย่างมากเมื่อมีการเปลี่ยนแปลงเล็กน้อยในข้อมูลฝึก (training examples) จะมีแวเรียนซ์สูง หากเราเข้าใจถึงทฤษฎีไบแอส-แวเรียนซ์ก็จะช่วยให้เราสามารถเลือกตัวเรียนรู้สำหรับปัญหาหนึ่งๆ ได้เหมาะสม ในงานวิจัยนี้เราจะสร้างกลุ่มก้อนตัวจำแนกที่สามารถลดทั้งไบแอสและแวเรียนซ์เพื่อสร้างตัวจำแนกประเภทมีความถูกต้องสูง

วัตถุประสงค์

วัตถุประสงค์ของการวิจัยในโครงการนี้มีดังต่อไปนี้

- เพื่อพัฒนาฟังก์ชันเคอร์เนลที่ยืดหยุ่นมากพอที่จะปรับตัวเองเข้ากับงานที่ทำ
- เพื่อพัฒนาเทคนิคกลุ่มก้อนตัวจำแนกที่สามารถลดไบแอสและแวเรียนซ์

ระเบียบวิธีวิจัยและผลที่ได้

1. การวิจัยพัฒนาฟังก์ชันเคอร์เนลสำหรับซัพพอร์ตเวกเตอร์แมชชีน

ซัพพอร์ตเวกเตอร์แมชชีน (เอสวีเอ็ม) เป็นอัลกอริทึมเรียนรู้ที่พัฒนาโดย Vapnik [Vapnik, 1998] และได้ถูกนำมาใช้อย่างมีประสิทธิภาพในงานประยุกต์ต่างๆ เช่น การรู้จำรูปแบบ การประมาณฟังก์ชัน เป็นต้น เอสวีเอ็มจะแบ่งแยกข้อมูลแบบเชิงเส้นในปริภูมิลักษณะ (feature space) โดยใช้ฟังก์ชันเคอร์เนลที่สอดคล้องกับเงื่อนไข

ของ Mercer [Shawe-Taylor & Cristianini, 2004] ฟังก์ชันเคอร์เนลเหล่านี้จะส่งข้อมูลฝึก (training data) ที่แสดงในรูปเวกเตอร์ไปยังปริภูมิที่มีมิติสูงมากๆ ที่ซึ่งการแบ่งแยกเชิงเส้นทำได้ง่ายขึ้น และจะทำการหาระนาบหลายมิติที่มีระยะขอบมากสุด (maximum margin) ในปริภูมินี้ ดังนั้นความซับซ้อนของระนาบหลายมิติแบ่งแยกดีสุดนี้จะขึ้นกับคุณสมบัติและประสิทธิภาพของฟังก์ชันเคอร์เนลที่เลือก

ฟังก์ชันเคอร์เนลมีหลายประเภทเช่น linear kernel, polynomial kernel, sigmoid kernel, RBF kernel เป็นต้น เคอร์เนลแต่ละประเภทจะเหมาะกับงานบางชนิดและเราจะต้องเลือกเคอร์เนลเองให้เหมาะกับงานที่ทำ เคอร์เนลอาร์บีเอฟ (RBF kernel) เป็นเคอร์เนลแบบหนึ่งที่มีประสิทธิภาพสูงและได้รับความสำเร็จในการใช้กับงานประยุกต์หลายๆ ด้าน แต่ก็ยังมีข้อจำกัดอยู่บ้าง ในงานวิจัยนี้เราจึงนำเสนอการปรับปรุงประสิทธิภาพของการจำแนกประเภทข้อมูลของเคอร์เนลอาร์บีเอฟ โดยการรวมเคอร์เนลอาร์บีเอฟหลายๆ ตัวที่สเกลต่างๆ เคอร์เนลเหล่านี้จะถูกรวมกับแบบมีน้ำหนัก (weight) โดยที่น้ำหนักและความกว้าง (width) ของเคอร์เนลอาร์บีเอฟและพารามิเตอร์สามัญ (regularization parameter) ของเอสวีเอ็มจะถูกเรียกรวมกันว่า ไฮเปอร์พารามิเตอร์ (hyperparameter)

เพื่อที่จะให้ได้ความถูกต้อง (accuracy) ที่สูง เราจำเป็นต้องตรวจสอบไฮเปอร์พารามิเตอร์เหล่านี้จำนวนมาก โดยปกติไฮเปอร์พารามิเตอร์เหล่านี้จะถูกตรวจสอบโดยใช้การค้นหาแบบกริด (Grid Search) ซึ่งจะลองเปลี่ยนค่าไฮเปอร์พารามิเตอร์เป็นช่วงๆ ที่กำหนดไว้ แต่วิธีการนี้จะเสียเวลาทำงานมาก ดังนั้นเราจึงเสนอการใช้ กลยุทธ์ทางวิวัฒนาการ – อีเอส (evolutionary strategy – ES) มาใช้เลือกค่าของไฮเปอร์พารามิเตอร์

1.1 ซัพพอร์ตเวกเตอร์แมชชีนและกลยุทธ์ทางวิวัฒนาการ

หัวข้อนี้กล่าวถึงความรู้ภูมิหลังที่จะนำไปประยุกต์ใช้สำหรับการพัฒนาฟังก์ชันเคอร์เนลแบบหลายสเกล ซึ่งได้แก่ ซัพพอร์ตเวกเตอร์แมชชีนและกลยุทธ์ทางวิวัฒนาการ

1.1.1 การจำแนกข้อมูลของซัพพอร์ตเวกเตอร์แมชชีน

1) ซัพพอร์ตเวกเตอร์แมชชีนเชิงเส้น (Linear support vector machine)

แนวคิดหลักๆ ในการจำแนกข้อมูลของซัพพอร์ตเวกเตอร์แมชชีนคือ การสร้างระนาบหลายมิติที่ใช้แบ่งข้อมูลออกเป็นสองกลุ่ม สมมติมีเซตของข้อมูล D ที่ประกอบด้วยตัวอย่างจำนวน l ตัวในปริภูมิอันดับ n ที่มีสองกลุ่ม (+1 และ –1)

$$D = \left\{ \left(\mathbf{x}_k, y_k \right) \middle| \, k \in \{1,..,l\}, \mathbf{x}_k \in \Re^n, y_k \in \{+1,-1\} \right\} \tag{1}$$

ระนาบหลายมิติในปริภูมิอันดับ n ถูกกำหนดโดย $(\mathbf{w},b)$ เมื่อ w คือเวกเตอร์ในปริภูมิอันดับ n ที่ตั้งฉากกับระนาบหลายมิติ และ b คือค่าคงที่ ระนาบหลายมิติ $(\mathbf{w}\cdot x)+b$ จะแบ่งข้อมูลได้ก็ต่อเมื่อ

$$\begin{aligned} (\mathbf{w}\cdot\mathbf{x}_i)+b > 0 \quad &\text{if} \quad y_i = +1 \\ (\mathbf{w}\cdot\mathbf{x}_i)+b < 0 \quad &\text{if} \quad y_i = -1 \end{aligned} \tag{2}$$

ถ้าเราต้องการค่า $\mathbf{w}$ และ b ที่ทำให้จุดที่อยู่ใกล้กับระนาบหลายมิติมากที่สุดมีระยะห่าง $1/|\mathbf{w}|$ แล้ว จะได้

$$\begin{aligned} (\mathbf{w}\cdot\mathbf{x}_i)+b \geq 1 \quad &\text{if} \quad y_i = +1 \\ (\mathbf{w}\cdot\mathbf{x}_i)+b \leq -1 \quad &\text{if} \quad y_i = -1 \end{aligned} \tag{3}$$

ซึ่งเท่ากับ

$$y_i[(\mathbf{w}\cdot\mathbf{x}_i)+b] \geq 1 \quad \forall i \tag{4}$$

ในการค้นหาระนาบหลายมิติแบ่งแยกดีสุด (optimal separating hyperplane) จะต้องค้นหาระนาบหลายมิติที่มีระยะห่างระหว่างข้อมูลที่ใช้ฝึกกับระนาบหลายมิติที่น้อยที่สุดมีค่ามากที่สุด ระยะห่างหรือมาร์จินระหว่างข้อมูลตัวอย่างสองตัวจากประเภทที่แตกต่างกันมีค่าเท่ากับ

$$d(\mathbf{w},b) = \min_{\{\mathbf{x}_i | y_i=1\}} \frac{(\mathbf{w}\cdot\mathbf{x}_i)+b}{|\mathbf{w}|} - \max_{\{\mathbf{x}_i | y_i=-1\}} \frac{(\mathbf{w}\cdot\mathbf{x}_i)+b}{|\mathbf{w}|} \quad (5)$$

จากสมการที่ (3) ค่าที่น้อยที่สุดและมากที่สุดที่เหมาะสมคือ ± 1 ดังนั้นจะได้ว่าเราต้องค้นหาระนาบที่ทำให้ค่าของฟังก์ชัน

$$d(\mathbf{w},b) = \frac{1}{|\mathbf{w}|} - \frac{-1}{|\mathbf{w}|} = \frac{2}{|\mathbf{w}|} \quad (6)$$

สูงที่สุด ดังนั้นปัญหานี้จึงเท่ากับ

- ลดค่าของ $|\boldsymbol{w}|^2/2$ ให้ต่ำที่สุด
- โดยขึ้นกับเงื่อนไขต่อไปนี้

  (1) $y_i[(\mathbf{w}\cdot\mathbf{x}_i)+b] \geq 1 \quad \forall i$

สำหรับกรณีที่ไม่สามารถแบ่งข้อมูลด้วยระนาบหลายมิติโดยไม่มีข้อผิดพลาดเกิดขึ้นนั้น เราจำเป็นต้องปรับเงื่อนไขโดยเพิ่มพจน์ค่าปรับ (penalty term) ซึ่งประกอบด้วยผลรวมของความคลาดเคลื่อน ξ_i จากขอบเขตเข้าไปในเงื่อนไข แล้วนำไปคูณกับเรกูลาไรเซชันพารามิเตอร์ C เราเรียกกรณีนี้ว่าเป็น เอสวีเอ็มแบบระยะขอบชนิดอ่อน (soft margin SVM) ดังนั้นปัญหาตอนนี้คือ

- ลดค่าของ $\frac{|\mathbf{w}|^2}{2} + C\sum_{i=1}^{l}\xi_i$ ให้ต่ำที่สุด
- โดยขึ้นกับเงื่อนไขต่อไปนี้

  (1) $y_i[(\mathbf{w}\cdot\mathbf{x}_i)+b] \geq 1-\xi_i,$

  (2) $\xi_i \geq 0 \quad \forall i$

  นำลากรานจ์มาใช้กับปัญหานี้ ดังนั้นสามารถแปลงปัญหาเป็น
- ลดค่าของฟังก์ชันด่านล่างนี้ให้ต่ำที่สุด

$$L(\mathbf{w},b,\alpha) = \sum_{i=1}^{l}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{l}\alpha_i\alpha_j y_i y_j(\mathbf{x}_i\cdot\mathbf{x}_j) \quad (7)$$

- โดยขึ้นกับเงื่อนไขต่อไปนี้

  (1) $0 \leq \alpha_i \leq C, \ \forall i$

  (2) $\sum_{i=1}^{l}\alpha_i y_i = 0$

เมื่อ α_i เรียกว่าตัวคูณลากรานจ์ ตัวอย่างข้อมูลฝึกแต่ละตัวจะมีตัวคูณลากรานจ์หนึ่งตัว ตัวอย่างข้อมูลฝึกที่มีค่า $\alpha_i > 0$ เรียกว่าซัพพอร์ตเวกเตอร์ (support vector) และเป็นซัพพอร์ตเวกเตอร์ที่ทำให้ค่าทางขวามือของสมการที่ (4) เท่ากับ 1 ส่วนตัวอย่างข้อมูลฝึกตัวอื่นๆ ที่มี $\alpha_i = 0$ สามารถตัดออกไปจากเซตของตัวอย่างข้อมูลฝึกได้โดยไม่มีผลกระทบใดๆ ต่อผลลัพธ์ของระนาบหลายมิติที่จะเป็นระนาบหลายมิติแบ่งแยกดีสุด

ให้ α^0 ซึ่งเป็นเวกเตอร์ในปริภูมิอันดับ l แทนค่าต่ำสุดของ $L(\mathbf{w},b,\alpha)$ ถ้า $\alpha_i^0 > 0$ แล้ว $\mathbf{x}_i$ คือซัพพอร์ตเวกเตอร์ ระนาบหลายมิติแบ่งแยกดีสุด $(\mathbf{w}^0, b^0)$ สามารถเขียนในพจน์ของ α^0 และข้อมูลฝึก โดยเฉพาะอย่างยิ่งในพจน์ของซัพพอร์ตเวกเตอร์ได้ดังนี้

$$\mathbf{w}^0 = \sum_{i=1}^{l} \alpha_i^0 y_i \mathbf{x}_i = \sum_{\text{support vectors}} \alpha_i^0 y_i \mathbf{x}_i \tag{8}$$

$$b^0 = 1 - \mathbf{w}^0 \cdot \mathbf{x}_i \text{ for } \mathbf{x}_i \text{ with } y_i = 1 \text{ and } 0 < \alpha_i < C \tag{9}$$

ระนาบหลายมิติแบ่งแยกดีสุดจะจำแนกจุดต่างๆ ตามเครื่องหมายของผลลัพธ์ของฟังก์ชัน $f(\mathbf{x})$

$$\begin{aligned} f(\mathbf{x}) &= \operatorname{sign}\left(\mathbf{w}^0 \cdot \mathbf{x} + b^0\right) \\ &= \operatorname{sign}\left(\sum_{\text{suport vec tors}} \alpha_i^0 \, y_i \, (\mathbf{x}_i \cdot \mathbf{x}) + b^0 \right) \end{aligned} \tag{10}$$

2) ซัพพอร์ตเวกเตอร์แมชชีนไม่เชิงเส้น (Non-linear support vector machine)

อัลกอริทึมที่ได้กล่าวมาแล้วใช้ได้กับข้อมูลที่สามารถแบ่งด้วยระนาบหลายมิติเชิงเส้นได้เท่านั้น ซัพพอร์ตเวกเตอร์แมชชีนแก้ปัญหากรณีที่ข้อมูลแบ่งไม่ได้ด้วยระนาบเชิงเส้นโดยส่งข้อมูลตัวอย่างไปยังปริภูมิลักษณะอันดับสูง (higher-dimensional feature space) โดยเลือกใช้ฟังก์ชันการส่งที่ไม่เป็นเชิงเส้น นั่นคือ เราเลือกฟังก์ชันในการส่ง $\Phi : \Re^n \mapsto \mathrm{H}$ เมื่อปริภูมิของ $\mathbf{H}$ มีอันดับสูงกว่าปริภูมิอันดับ n เราสามารถค้นหาระนาบหลายมิติแบ่งแยกดีสุดในปริภูมิอันดับสูงที่เทียบเท่ากับการแยกที่ไม่เป็นเชิงเส้นใน $\Re^n$

ข้อมูลฝึกที่ใช้ในการเรียนรู้ในสมการที่ (7) อยู่ในรูปของผลคูณเชิงสเกลาร์ระหว่างเวกเตอร์ ดังนั้นในปริภูมิอันดับสูงเราสามารถจัดการกับข้อมูลในรูปของ $\Phi(\mathbf{x}_i)\cdot\Phi(\mathbf{x}_j)$ ถ้าปริภูมิของ $\mathbf{H}$ มีอันดับสูงจะทำให้ยุ่งยากหรือใช้การคำนวณมาก อย่างไรก็ตามถ้าเรามีฟังก์ชันเคอร์เนลที่สามารถคำนวณ $K(\mathbf{x}_i,\mathbf{x}_j)=\Phi(\mathbf{x}_i)\cdot\Phi(\mathbf{x}_j)$ แล้วเราสามารถใช้ฟังก์ชันนี้แทนที่ $\mathbf{x}_i\cdot\mathbf{x}_j$ ทุกๆ ที่ในการคำนวณ และไม่จำเป็นต้องรู้ฟังก์ชันที่ใช้ส่ง Φ จริงๆ ฟังก์ชันเคอร์เนลที่นิยมใช้ มีดังตารางที่ 1 ด้านล่างนี้

ตารางที่ 1 ตัวอย่างฟังก์ชันเคอร์เนล

Kernel	Formula
Linear	$K(\mathbf{x},\mathbf{y}) = \mathbf{x}\cdot\mathbf{y}$
Homogeneous Polynomial	$K(\mathbf{x},\mathbf{y}) = \left(\mathbf{x}\cdot\mathbf{y}\right)^d$
Inhomogeneous Polynomial	$K(\mathbf{x},\mathbf{y}) = \left\lvert\mathbf{x}\cdot\mathbf{y} + c\right\rvert^d$
Exponential RBF	$K(\mathbf{x},\mathbf{y}) = e^{-\gamma\lVert\mathbf{x}-\mathbf{y}\rVert}$
Gaussian RBF	$K(\mathbf{x},\mathbf{y}) = e^{-\gamma\lVert\mathbf{x}-\mathbf{y}\rVert^2}$
Multi-quadratic	$K(\mathbf{x},\mathbf{y}) = -\sqrt{\lVert\mathbf{x}-\mathbf{y}\rVert^2 + c^2}$

1.1.2 กลยุทธ์ทางวิวัฒนาการ

กลยุทธ์ทางวิวัฒนาการ (อีเอส) เป็นขั้นตอนวิธีทางวิวัฒนาการชนิดหนึ่ง ถูกพัฒนาโดย Rechenberg and Schwefel [Beyer & Schwefel, 2002] อีเอสถูกพัฒนาขึ้นมาเพื่อแก้ปัญหาหลากหลายที่เกี่ยวกับการหาค่าเหมาะ

สุด โดยจะเลียนแบบกระบวนการวิวัฒนาการทางธรรมชาติ อีเอสใช้เวกเตอร์จำนวนจริงความยาวจำกัดเพื่อแทนผลเฉลย อัลกอริทึมพื้นฐานของอีเอสแสดงในรูปที่ 1

Basic ES algorithm:

Step 1: Randomly generate a parent population of μ solutions.

Step 2: Evaluate all parents to determine their fitness.

Step 3: Apply reproduction operators to create λ offstring.

Step 4: Evaluate and keep the μ fittest individuals.

Step 5: Go to Step 3 unless an acceptable solution has been found or a fixed number of generations has been produced and evaluated.

รูปที่ 1 อัลกอริทึมอีเอสพื้นฐาน

แต่ละจุดในปริภูมิค้นหาจะเป็นปัจเจก (individual) หรือผลเฉลย (solution) ที่เป็นไปได้ อีเอสใช้ประชากร (population) ที่มีปัจเจกจำนวน μ ตัว เพื่อค้นหาผลเฉลยที่ดีขึ้น ประชากรเริ่มต้นจะถูกสร้างขึ้นอย่างสุ่มและควรจะกระจายทั่วๆ ปริภูมิค้นหา เพื่อที่ว่าทุกบริเวณจะถูกค้นหาได้อย่างทั่วถึง ปัจเจกแต่ละตัวในแต่ละรุ่น (generation) จะถูกประเมินเพื่อหาค่าความเหมาะ (fitness) และเป้าหมายของการค้นหาคือปัจเจกที่มีค่าความเหมาะสูงสุด

ตัวกระทำการหลักในอีเอสคือการกลายพันธ์เกาส์เซียน (Gaussian mutation) ส่วนตัวกระทำการอีกตัวก็คือ การรวมใหม่แบบค่ากลาง (intermediate recombination) ซึ่งจะนำเวกเตอร์แม่ 2 ตัวมาหาค่าเฉลี่ยเป็นตัวลูก จากนั้นทั้งแม่และลูกทั้งหมดจะถูกประเมินค่าความเหมาะและตัวที่เหมาะสุดจำนวนหนึ่งจะอยู่รอดในรุ่นต่อไป ส่วนตัวที่ไม่เหมาะจะถูกคัดออก

ในแต่ละรุ่น ปัจเจกจะถูกรวมใหม่และกลายพันธ์เพื่อผลิตลูก อีเอสสามารถค้นหาปริภูมิหลายๆ บริเวณพร้อมกันได้ ซึ่งจะช่วยลดเวลาที่จะค้นหาผลเฉลยที่ดี อัลกอริทึมนี้จะหยุดเมื่อจำนวนรุ่นเกินกว่าค่าที่กำหนดหรือได้ผลเฉลยที่มีค่าดีกว่าค่าที่กำหนด อัลกอริทึมอีเอสมีหลายเวอร์ชัน $(\mu+\lambda)$-ES และ (μ, λ)-ES เป็นเวอร์ชันที่รู้จักกันดีและมีการใช้งานอย่างแพร่หลาย ในกรณีของ $(\mu+\lambda)$-ES นี้ แม่ μ ตัวจะผลิตลูก λตัว แล้วทั้งแม่และลูกจะแข่งขันโดยเท่าเทียมเพื่อการอยู่รอดในรุ่นถัดไป ส่วนกรณีของ (μ, λ)-ES นั้น แม่ μ ตัวจะผลิตลูก λ $(>\mu)$ ตัว แล้วลูกที่ดีสุด μ ตัวจะถูกคัดเลือกให้อยู่รอดในรุ่นถัดไป ในงานวิจัยนี้เราเลือกใช้ $(\mu+\lambda)$-ES สำหรับการสร้างเคอร์เนลที่นำเสนอ

1.2 เคอร์เนลอาร์บีเอฟแบบหลายสเกล (Multi-scale RBF kernel)

เคอร์เนลหลากหลายประเภทสามารถนำมาใช้กับเอสวีเอ็มได้ เคอร์เนลบางประเภทจะเหมาะกับปัญหาบางชนิดและเคอร์เนลอีกหลายประเภทที่ถูกออกแบบสำหรับปัญหาเฉพาะอย่าง เกาส์เซียนอาร์บีเอฟเป็นเคอร์เนลที่ถูกใช้อย่างกว้างขวางในหลายๆ ปัญหา อาร์บีเอฟใช้ระยะยูคลิดระหว่างจุดสองจุดในปริภูมินำเข้าเพื่อหาสหสัมพันธ์ในปริภูมิลักษณะ จุดที่อยู่ใกล้กันจะมีสหสัมพันธ์สูงในขณะที่จุดที่อยู่ไกลกันจะมีสหสัมพันธ์ต่ำ ค่าสหสัมพันธ์นี้ค่อนข้างเรียบและอาร์บีเอฟนี้มีพารามิเตอร์ตัวเดียวคือความกว้างที่สามารถปรับแต่งได้และไม่เพียงพอสำหรับปัญหาที่ซับซ้อน

วิธีการหนึ่งที่จะสร้างเคอร์เนลที่ดีขึ้นสามารถทำได้โดยการปรับความเร่งของการลดความเร็วในแต่ละช่วงของระยะห่างระหว่างจุดสองจุด นอกจากนี้เคอร์เนลควรมีคุณสมบัติที่ดีของอาร์บีเอฟ เราจึงนำเสนอการรวมอาร์บีเอฟที่สเกลต่างๆ ตามสูตรด้านล่างนี้

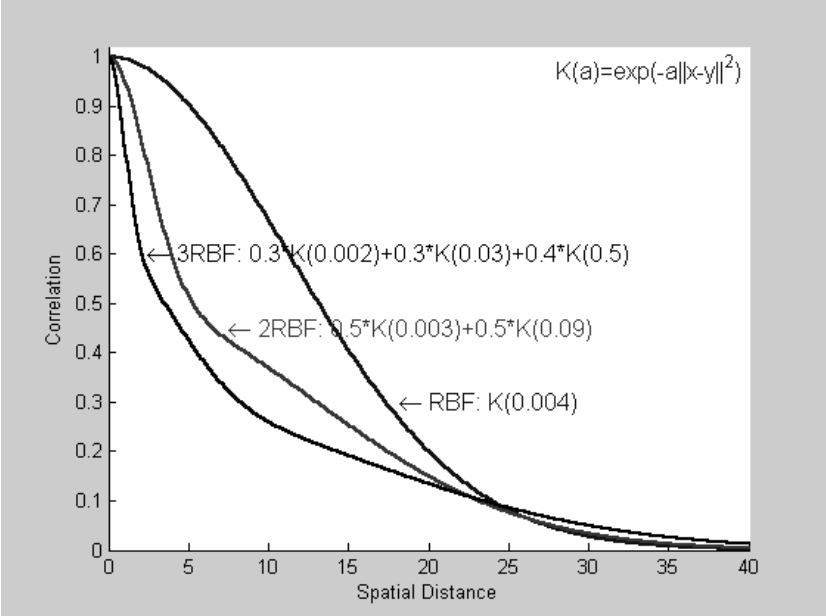
$$K(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{n} a_i K(\mathbf{x},\mathbf{y},\gamma_i) \tag{11}$$

โดยที่ n เป็นจำนวนเต็มบวก, $a_i \geq 0$ for $i = 1,...,n$ เป็นค่าน้ำหนักตัวเลขบวก และ

$$K(\mathbf{x},\mathbf{y},\gamma_i) = e^{-\gamma_i \|\mathbf{x}-\mathbf{y}\|^2} \tag{12}$$

เป็นอาร์บีเอฟที่มีความกว้าง $\gamma_i \geq 0$ for $i = 1,...,n$

สหสัมพันธ์ในปริภูมิลักษณะ (ความสัมพันธ์ระหว่างฟังก์ชันเคอร์เนลและระยะระหว่างจุดสองจุดในปริภูมินำเข้า) ของเคอร์เนลอาร์บีเอฟหลายสเกลนี้ในกรณีของ n = 1, 2, 3 แสดงในรูปที่ 2



รูปที่ 2 กราฟ RBF, 2-RBF และ 3-RBF

รูปที่ 2 แสดงให้เห็นว่าสหสัมพันธ์ของเคอร์เนลอาร์บีเอฟจะค่อนข้างเรียบ ในขณะที่ค่าของ 2-RBF, 3-RBF ค่อนข้างยืดหยุ่นมากกว่า ซึ่งสามารถอธิบายได้ว่าการเพิ่มพารามิเตอร์ที่ปรับได้จะช่วยให้เคอร์เนลมีความยืดหยุ่นในการปรับตัวได้มากขึ้น โดยทั่วไป เราจะไม่รู้ฟังก์ชันที่ส่งข้อมูลในปริภูมินำเข้าไปยังปริภูมิลักษณะ อย่างไรก็ดีการมีอยู่ของฟังก์ชันเหล่านี้สามารถรับประกันได้โดยทฤษฎีของ Mercer

ในสมการที่ (11) กรณีที่จำนวนพจน์ของฟังก์ชันอาร์บีเอฟเท่ากับ n เคอร์เนลจะมีพารามิเตอร์ $2n$ ตัว (พารามิเตอร์ n ตัวสำหรับปรับน้ำหนักและพารามิเตอร์อีก n ตัวที่เป็นค่าความกว้างของอาร์บีเอฟ) อย่างไรก็ดี เราสามารถลดจำนวนพารามิเตอร์ให้เหลือ $2n-1$ ตัวได้ โดยกำหนดให้พารามิเตอร์ตัวแรกมีค่าเท่ากับ 1 ดังนั้นในงานวิจัยนี้เราจะใช้สมการด้านล่างนี้แทนเคอร์เนลอาร์บีเอฟแบบหลายสเกล

$$K(\mathbf{x},\mathbf{y}) = K(\mathbf{x},\mathbf{y},\gamma_0) + \sum_{i=1}^{n-1} a_i K(\mathbf{x},\mathbf{y},\gamma_i) \tag{13}$$

เวลาที่ฟังก์ชันอาร์บีเอฟถูกรวมเข้าด้วยกัน ระนาบแบ่งแยกจะยืดหยุ่นมากขึ้นกว่าการใช้อาร์บีเอฟเดี่ยว อย่างไรก็ดีเคอร์เนลอาร์บีเอฟแบบหลายสเกลนี้อาจทำให้เกิดปัญหาการปรับเหมาะเกินไป (overfitting problem) ซึ่งปัญหานี้สามารถแก้ไขได้โดยการเลือกใช้ฟังก์ชันจุดประสงค์ (objective function) ที่เหมาะสมสำหรับอัลกอริทึมอีเอส

1.3 การหาค่าไฮเปอร์พารามิเตอร์ของเอสวีเอ็ม

วิธีการที่ใช้ในงานวิจัยนี้จะรวมเทคนิคสองอย่างคือ เอสวีเอ็มและอีเอส โดยใช้อีเอสเพื่อปรับแต่งไฮเปอร์พารามิเตอร์ของเอสวีเอ็มซึ่งใช้เคอร์เนลอาร์บีเอฟแบบหลายสเกล และดังเช่นที่แสดงก่อนหน้านี้เรามีพารามิเตอร์ทั้งสิ้น $2n-1$ ตัวในกรณีที่ใช้อาร์บีเอฟ n พจน์ พารามิเตอร์เหล่านี้จะส่งผลกระทบต่อประสิทธิภาพของเคอร์เนลที่นำเสนอ ในกรณีของเอสวีเอ็มแบบระยะขอบชนิดอ่อนนั้น จะมีเรกูลาไรเซชันพารามิเตอร์ C อีกตัวหนึ่งที่ส่งผลกระทบต่อประสิทธิภาพการจำแนกประเภทข้อมูล เราเรียกพารามิเตอร์ของฟังก์ชันเคอร์เนลและเรกูลาไรเซชันพารามิเตอร์รวมกันว่าไฮเปอร์พารามิเตอร์ และได้พิจารณานำอีเอสมาเพื่อหาค่าเหมาะสมของไฮเปอร์พารามิเตอร์ โดยเวอร์ชันของอีเอสที่เราเลือกใช้คือ (μ+λ)-ES ซึ่ง แม่ μ ตัวจะผลิตลูก λ ตัว แล้วทั้งแม่และลูกจะแข่งขันอย่างเท่าเทียมเพื่อการอยู่รอด

กำหนดให้ $\vec{v}$ เป็นเวกเตอร์จำนวนจริงเลขบวกหรือศูนย์ของไฮเปอร์พารามิเตอร์ที่มีมิติเท่ากับ $2n$ เราจะเขียนเวกเตอร์ $\vec{v}$ ในรูปแบบดังนี้

$$\vec{v} = (C, \gamma_0, a_1, \gamma_1, a_2, \gamma_2, ..., a_{n-1}, \gamma_{n-1}) \tag{14}$$

โดยที่ C คือเรกูลาไรเซชันพารามิเตอร์, $\gamma_i \geq 0$ for $i = 1,...,n-1$ เป็นความกว้างของอาร์บีเอฟ, $a_i \geq 0$ for $i = 1,...,n-1$ เป็นน้ำหนักของอาร์บีเอฟและ n คือจำนวนพจน์ของอาร์บีเอฟ เป้าหมายของเราคือพยายามหา $\vec{v}$ ที่ทำให้ค่าฟังก์ชันจุดประสงค์ $g(\vec{v})$ สูงสุด (5+10)-ES ได้ถูกนำมาใช้เพื่อปรับค่าไฮเปอร์พารามิเตอร์เหล่านี้ อัลกอริทึมของ (5+10)-ES แสดงในรูปที่ 3

อัลกอริทึมเริ่มจากรุ่นที่ 0 (t=0) และเลือกเวกเตอร์ผลเฉลี่ย 5 ตัว ($\vec{v}_1,..., \vec{v}_5 \in R_+^{2n}$) พร้อมด้วยส่วนเบี่ยงเบนมาตรฐาน $\vec{\sigma} \in R_+^{2n}$ แบบสุ่ม และเวกเตอร์ผลเฉลี่ยทั้งห้าตัวนี้จะถูกประเมินเพื่อหาค่าความเหมาะ จากนั้นเราใช้การรวมใหม่แบบค่ากลางเพื่อสร้างผลเฉลยใหม่ 10 ตัว ซึ่งแต่ละตัวเกิดจากค่าเฉลี่ยของผลเฉลยเดิม 1 คู่

```
t = 0;
initialization(v1,..., v5 , σ );
evaluate g(v1),..., g(v5);
while (t < 1000) do
    for i =1 to 10 do
        v'i = recombination(v1,..., v5 );
        v'i = mutate(v'i );
        evaluate g(v'i);
    end
    (v1,..., v5 ) = select(v1,..., v5 , v'1,..., v'10 )
    σ = mutate σ ( σ );
    t = t +1;
end
```

รูปที่ 3 อัลกอริทึม (5+10)-ES

$$\vec{v}_1' = \frac{1}{2}(\vec{v}_1 + \vec{v}_2) \quad (15)$$

$$\vec{v}_2' = \frac{1}{2}(\vec{v}_1 + \vec{v}_3) \quad (16)$$

$$\vdots$$

$$\vec{v}_{10}' = \frac{1}{2}(\vec{v}_4 + \vec{v}_5) \quad (17)$$

หลังจากนั้น ผลเฉลยเหล่านี้จะกลายพันธ์โดยใช้สูตรด้านล่างนี้

$$mutate(\vec{v}) = (C + z_1, \gamma_0 + z_2, \ldots, a_{n-1} + z_{2n-1}, \gamma_{n-1} + z_{2n}) \quad (18)$$

$$z_i \sim N_i(0, \sigma_i^2) \quad (19)$$

$\vec{v}_i'$ แต่ละตัวจะกลายพันธ์โดยการบวก $(z_1, z_2, \ldots, z_{2n})$ เพิ่มเข้าไป โดยที่ z_i เป็นค่าสุ่มจากการกระจายปกติที่มีค่าเฉลี่ยเป็นศูนย์และความแปรปรวนเป็น σ_i^2 ในแต่ละรุ่นส่วนเบี่ยงเบนมาตรฐานจะถูกปรับโดยสมการด้านล่างนี้

$$mutate_{\sigma}(\vec{\sigma}) = (\sigma_1 \cdot e^{z_1}, \sigma_2 \cdot e^{z_2}, \ldots, \sigma_{2n} \cdot e^{z_{2n}}) \quad (20)$$

$$z_i \sim N_i(0, \tau^2) \quad (21)$$

โดยที่ τ เป็นค่าคงที่ใดๆ

ผลเฉลย 5 ตัวที่มีค่าความเหมาะสูงสุดจะถูกเลือกจากผลเฉลย 5+10 ตัวเดิม เพื่อใช้เป็นแม่ในรุ่นถัดไป อัลกอริทึมนี้จะวนจนกระทั่งจำนวนรุ่นที่ผลิตได้ถึงค่าที่กำหนดหรือผลเฉลยมีค่าดีเข้าเกณฑ์ที่ตั้งไว้

1.4 ฟังก์ชันจุดประสงค์ที่ใช้ในอีเอส

ส่วนสำคัญส่วนหนึ่งของอัลกอริทึมเชิงวิวัฒน์คือการนิยามฟังก์ชันจุดประสงค์สำหรับงานที่ทำ ในกรณีของเราคือการประเมินไฮเปอร์พารามิเตอร์ ซึ่งก็มีหลายวิธีในการนิยามฟังก์ชัน ในงานวิจัยนี้เรานำเสนอฟังก์ชันจุดประสงค์สามฟังก์ชันด้วยกันคือ (1) ความถูกต้องของการฝึก (training accuracy) (2) ขอบเขตของค่าผิดพลาดโดยนัยทั่วไป (bound of generalization error) (3) การตรวจสอบไขว้แบบเซตย่อยบนค่าความถูกต้องของการฝึก (subset cross-validation on training accuracy)

1.4.1 ความถูกต้องของการฝึก

ความถูกต้องของการฝึกเป็นมาตรวัดประสิทธิภาพในการเรียนรู้ ซึ่งจะแสดงความสามารถของเครื่องโดยคำนวณจากค่าความถูกต้องบนชุดข้อมูลฝึก ฟังก์ชันนี้เป็นฟังก์ชันที่ง่ายสุดที่จะใช้ในอัลกอริทึมอีเอส สูตรของฟังก์ชันนี้แสดงในสูตรด้านล่าง

$$g(\vec{v}) = \left(1 - \frac{\sum_{i=1}^{l} |y_i - f(\mathbf{x}_i)|}{2l}\right) \quad (22)$$

โดยที่ $\mathbf{x}_i \in R^N$ เป็นข้อมูลฝึก, $y_i \in \{-1, 1\}$ เป็นฉลาก และ $f(x_i)$ เป็นฟังก์ชันตัดสินใจของ $\mathbf{x}_i$ สำหรับ $i = 1, \ldots, l$ เราคาดหวังว่าชุดของไฮเปอร์พารามิเตอร์ที่เหมาะสมควรจะให้ค่าความถูกต้องของการฝึกมีค่ามาก

ซึ่งการเรียนรู้ไฮเปอร์พารามิเตอร์โดยใช้ความถูกต้องของการฝึกเป็นฟังก์ชันจุดประสงค์จะถูกทดสอบประสิทธิภาพบนชุดข้อมูลทดสอบต่อไป

1.4.2 ขอบเขตของค่าผิดพลาดโดยนัยทั่วไป

ค่าผิดพลาดโดยนัยทั่วไปของอัลกอริทึมการเรียนรู้ของเครื่องคือฟังก์ชันที่วัดประสิทธิภาพของเครื่องในการจำแนกประเภทข้อมูล เราสามารถนิยาม Q ซึ่งเป็นเซตของฟังก์ชันจำนวนจริงบนลูกบอลรัศมี R ใน R^n ให้เป็น

$$Q = \left\{ \mathbf{x} \mapsto \mathbf{w} \cdot \mathbf{x} \ : \ \|\mathbf{w}\| \le 1, \|\mathbf{x}\| \le R \right\} \tag{23}$$

จะมีค่าคงที่ c ซึ่งสำหรับการกระจายความน่าจะเป็นใดๆ ด้วยความน่าจะเป็นอย่างน้อย $1-\delta$ บนตัวอย่าง $\mathbf{z}$ ที่ถูกหยิบขึ้นมาอย่างอิสระต่อกันทั้งสิ้น l ตัว ถ้าตัวจำแนกประเภท $f = \operatorname{sgn}(q) \in \operatorname{sgn}(Q)$ มีระยะขอบอย่างน้อย γ บนทุกอย่างใน $\mathbf{z}$ แล้วจะได้ว่าค่าผิดพลาดของ h จะไม่มากกว่า

$$\frac{c}{l}\left(\frac{R^2}{\gamma^2}\log^2 l + \log(1/\delta)\right) \tag{24}$$

นอกจากนั้น ด้วยความน่าจะเป็นอย่างน้อย $1-\delta$, ตัวจำแนกประเภท $h \in \operatorname{sgn}(B)$ ใดๆ จะมีค่าผิดพลาดไม่เกิน

$$\frac{k}{l} + \sqrt{\frac{c}{l}\left(\frac{R^2}{\gamma^2}\log^2 l + \log(1/\delta)\right)} \tag{25}$$

โดยที่ k คือจำนวนตัวอย่างมีฉลาก z ที่มีระยะขอบน้อยกว่า γ [Bartlett & Shawe-Taylor, 1999] ดังนั้นเราสามารถระบุขอบเขตบนของค่าความผิดพลาดของเอสวีเอ็มแม้ว่าเคอร์เนลจะทำให้เกิดปริภูมิลักษณะที่มีมิติอนันต์ [Bartlett & Shawe-Taylor, 1999]

สูตร (25) มีชื่อเรียกว่า ขอบเขตของค่าผิดพลาดโดยนัยทั่วไป พจน์ $\frac{k}{l}$ คือความเสี่ยงเชิงประจักษ์ (empirical risk) ซึ่งถูกนิยามให้เป็นอัตราค่าผิดพลาดโดยเฉลี่ยที่วัดบนข้อมูลฝึกชุดหนึ่งๆ และ $\frac{R^2}{\gamma^2}$ เทียบได้กับมิติวีซี (VC-dimension) มิติวิซีก็คือมาตรวัดประสิทธิภาพของอัลกอริทึมจำแนกประเภทข้อมูลที่ถูกนำเสนอโดย Vapnik และ Chervonenkis [Vapnik & Chervonenkis, 1971] กล่าวโดยง่าย ประสิทธิภาพของพื้นผิวตัดสินใจในการจำแนกประเภทข้อมูลจะสัมพันธ์โดยตรงกับความซับซ้อนของพื้นผิวตัดสินใจนั้น

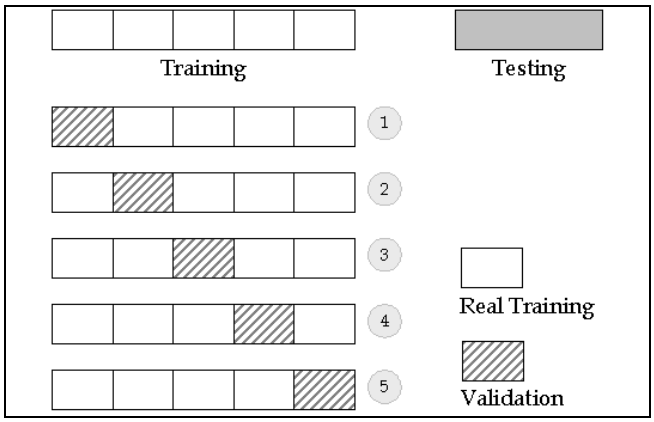
มิติวีซีนี้วัดความซับซ้อนของปริภูมิสมมติฐานซึ่งถูกกำหนดโดยจำนวนตัวอย่างที่ต่างกันที่สามารถแยกได้อย่างสมบูรณ์โดยสมมติฐานนั้น [Mitchell, 1997] เนื่องจากขอบเขตบนในสูตรที่ (25) พิจารณาถึงค่าผิดพลาดของการฝึกและมิติวีซี ดังนั้นเราคาดหวังว่าประสิทธิภาพโดยนัยทั่วไปของเอสวีเอ็มที่ใช้เคอร์เนลที่นำเสนอสามารถประมาณค่าได้ด้วยขอบเขตบนนี้

1.4.3 การตรวจสอบไขว้ของเซตย่อยบนค่าความถูกต้องของการฝึก

แม้ว่าค่าความถูกต้องของการฝึกสามารถคำนวณได้ง่าย แต่ว่าฟังก์ชันจุดประสงค์นี้อาจทำให้เกิดการปรับเหมาะเกินไปกับข้อมูล บางครั้งข้อมูลประกอบด้วยสัญญาณรบกวนและถ้าฟังก์ชันตัดสินใจปรับเหมาะกับข้อมูลมีสัญญาณรบกวนเกินไป ก็จะทำให้มโนทัศน์ที่เรียนมาผิดพลาดได้ ดังนั้นเราจึงเสนอให้ใช้การฝึกฟังก์ชันตัดสินใจ

ด้วยชุดข้อมูลหลายๆ ชุด พารามิเตอร์ที่ดีจะต้องทำงานได้ดีบนเซตข้อมูลฝึกหลายๆ ชุด อย่างไรก็ดีเนื่องจากเรามีข้อมูลฝึกจำกัด เราจึงพิจารณาการใช้การตรวจสอบไขว้ของเซตย่อย

ในตอนเริ่มต้น ชุดข้อมูลฝึกจะถูกแบ่งออกเป็น 5 ส่วนเท่าๆ กัน แต่ละส่วนมีจำนวนข้อมูลใกล้เคียงกัน ในแต่ละรุ่นของอัลกอริทึมอีเอส ตัวจำแนกประเภทที่มีพารามิเตอร์เดียวกันจะถูกฝึกและทดสอบห้าครั้ง ในรอบที่ j (j=1,…,5) ตัวจำแนกประเภทจะถูกฝึกบนเซตย่อยทุกเซตยกเว้นเซตที่ j หลังจากนั้นค่าความถูกต้องของการจำแนกประเภทจะถูกประเมินบนเซตย่อยที่ j การแบ่งชุดข้อมูลลักษณะนี้แสดงในรูปที่ 4



รูปที่ 4 การแบ่งข้อมูลฝึกออกเป็น 5 เซตย่อย

เฉพาะข้อมูลฝึกจริง (real training data) ที่ถูกใช้เพื่อสร้างตัวจำแนกประเภทด้วยพารามิเตอร์ชุดเดียวกัน หลังจากนั้นเซตตรวจสอบ (validation set) จะถูกใช้สำหรับคำนวณค่าความถูกต้องของตัวจำแนกประเภท ค่าเฉลี่ยของค่าความถูกต้องห้าครั้งจะถูกใช้เป็นฟังก์ชันจุดประสงค์ $g(\vec{v})$

$$g(\vec{v}) = \frac{\sum_{j=1}^{5}\left(1-\frac{\sum_{i=1}^{l_j}|y_i - f(\mathbf{x}_i)|}{2l_j}\right)}{5} \qquad (26)$$

ฟังก์ชันนี้สามารถมองได้ว่าเป็นตัวประมาณค่าที่ดีของค่าความถูกต้องโดยนัยทั่วไป เราได้ทดลองประยุกต์ใช้การตรวจสอบไขว้ของเซตย่อยกับขอบเขตของค่าผิดพลาดโดยนัยทั่วไป แต่พบว่าขอบเขตของค่าผิดพลาดโดยนัยทั่วไปได้พยายามแก้ปัญหาการปรับเหมาะเกินไปไว้อยู่แล้ว ดังนั้นจึงไม่มีความจำเป็นต้องใช้การตรวจสอบไขว้ของเซตย่อยกับขอบเขตของค่าผิดพลาดโดยนัยทั่วไปอีก ผลการทดลองนี้จะแสดงในหัวข้อต่อไป

1.5 การทดลองและผล

เพื่อวัดประสิทธิภาพของวิธีการที่นำเสนอ เราได้ทำการทดลองเพื่อฝึกและทดสอบเคอร์เนลแบบหลายสเกลกับเอสวีเอ็มบนชุดข้อมูลทั้งสิ้น 15 ชุดจาก UCI repository [Blake et al., 1998] ชุดข้อมูลแต่ละชุดมีสองประเภท จำนวนคุณลักษณะ จำนวนตัวอย่างของชุดข้อมูลแต่ละชุดแสดงในตารางที่ 2

ตารางที่ 2 ตัวอย่างฟังก์ชันเคอร์เนล

ชุดข้อมูล	จำนวนคุณลักษณะ	จำนวนตัวอย่าง
Checkers	2	192
Spiral	2	582
LiverDisorders	6	345
IndiansDiabetes	8	768
ThreeOfNine	9	512
TicTacToe	9	958
BreastCancer	10	699
ParityBits	10	1024
SolarFlare	10	1066
ClevelandHeart	13	270
Australian	14	690
German-org	24	1000
Ionosphere	34	351
Tokyo	44	959
Sonar	60	208

เราใช้การตรวจสอบไขว้ 5 พับ (5-fold cross-validation) เพื่อประเมินผลในการทดลอง อีเอสถูกใช้หาไฮเปอร์พารามิเตอร์ที่เหมาะสมของทั้งเคอร์เนลอาร์บีเอฟดั้งเดิมและเคอร์เนลที่นำเสนอ ค่า τ ในอัลกอริทึมอีเอสตั้งค่าให้เป็น 1 จำนวนพจน์ของอาร์บีเอฟเป็นจำนวนเต็มบวกที่น้อยกว่าหรือเท่ากับ 10 ความกว้างของอาร์บีเอฟ (γ_i) น้ำหนักของอาร์บีเอฟ (a_i) และเรกูลาไรเซชันพารามิเตอร์ (C) เป็นจำนวนจริงระหว่าง 0.0 ถึง 10.0 จำนวนรุ่นที่ใช้ในอัลกอริทึมอีเอสไม่เกิน 1000 รุ่น ค่าเฉลี่ยความถูกต้องจากการตรวจสอบไขว้ 5 พับสำหรับชุดข้อมูลแต่ละชุดแสดงในตารางที่ 3

ตารางที่ 3 เปอร์เซนต์ของค่าเฉลี่ยความถูกต้องของเคอร์เนลอาร์บีเอฟเดี่ยว

ชุดข้อมูล	*GridSearch*	**ฟังก์ชันจุดประสงค์ในอีเอส**			
		TrnAcc	*BdOfGenErr*	*5SubsetOnTrnAcc*	*5SubsetOnBdOfGenErr*
Checkers	83.32	**89.58**	88.56	83.82	80.19
Spiral	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
LiverDisorders	61.74	**69.57***	68.41	66.67	62.61
IndiansDiabete	64.97	72.52*	68.61	**73.30***	65.10
ThreeOfNine	53.51	53.51	**100.00****	**100.00****	**100.00****
TicTacToe	65.34	65.34	98.96**	**99.69****	98.75**
BreastCancer	86.41	**95.28***	94.85*	94.56*	93.85*
ParityBits	48.05	48.05	**80.46****	75.78**	48.05
SolarFlare	80.87	**83.30**	82.93	80.96	82.93
ClevelandHeart	55.56	55.55	55.55	**78.15***	62.59
Australian	**55.51**	**55.51**	**55.51**	**55.51**	**55.51**
German-org	70.10	**70.40**	70.00	70.30	70.00
Ionosphere	66.10	66.10	**95.45****	95.16**	92.31**
Tokyo	81.02	91.24*	88.32*	**91.66***	88.53*
Sonar	70.67	**91.35****	**91.35****	88.92*	88.43*
Average	69.54	73.82	82.60	**83.63**	79.26

ระดับของนัยสำคัญทางสถิติสำหรับความแตกต่างระหว่างวิธีการที่นำเสนอและการค้นหาแบบกริด: * คือ 0.01, ** คือ 0.001

ฟังก์ชันจุดประสงค์ของอัลกอริทึมอีเอสที่ใช้ในการทดลองคือ ค่าความถูกต้องของการฝึก (*TrnAcc*) ขอบเขตของค่าผิดพลาดโดยนัยทั่วไป (*BdOfGenErr*) การตรวจสอบไขว้ 5 พับบนค่าความถูกต้องของการฝึก (*5SubsetOnTrnAcc*) และ การตรวจสอบไขว้ 5 พับบนขอบเขตของค่าความผิดพลาดโดยนัยทั่วไป (*5SubsetOnBdOfGenErr*) ผลการทดลองถูกเปรียบเทียบกับการค้นหาแบบกริด (*GridSearch*) ซึ่งจะลองเปลี่ยนค่าไฮเปอร์พารามิเตอร์เป็นช่วงๆ ที่กำหนดไว้แล้วเลือกไฮเปอร์พารามิเตอร์ที่ดีที่สุดสำหรับทดสอบกับชุดข้อมูลทดสอบ ผลการทดลองในตารางเป็นค่าเฉลี่ยความถูกต้องที่ประเมินด้วยวิธีการตรวจสอบไขว้แบบ 5 พับ

ผลการทดลองแสดงให้เห็นถึงความสามารถของอัลกอริทึมอีเอสที่เหนือกว่าการค้นหาแบบกริด แม้ว่าการค้นหาแบบกริดจะใช้งานได้ง่าย แต่ประสิทธิภาพจะขึ้นกับขนาดของช่วงที่เลือกปรับค่าไฮเปอร์พารามิเตอร์ ถ้าขนาดของช่วงเล็กไปก็จะเสียเวลาในการคำนวณอย่างมาก ในทางกลับกันหากขนาดของช่วงใหญ่ไป การค้นหาแบบกริดก็อาจไม่พบผลเฉลยที่ดี ในขณะที่อัลกอริทึมอีเอสใช้กระบวนการสุ่มเพื่อค้นหาผลเฉลยที่แตกต่างกันไปพร้อมๆ กัน ผลเฉลยเหล่านี้ถูกปรับแก้ให้หลากหลายเพื่อสำรวจหาผลเฉลยที่ดีที่สุด

เวลาที่เราใช้ฟังก์ชันจุดประสงค์ต่างๆ กัน เราพบว่า *TrnAcc* ให้ค่าความถูกต้องดีสำหรับชุดข้อมูลทดสอบและ *BdOfGenErr* ให้ผลที่ดีกว่า *TrnAcc* ในชุดข้อมูลหลายชุดและให้ค่าเฉลี่ยค่าความถูกต้องที่สูงกว่า ผลการทดลองเหล่านี้แสดงให้เห็นว่า *TrnAcc* ไม่ใช่ฟังก์ชันจุดประสงค์ที่ดีที่สุด และในความเป็นจริงแล้ว *TrnAcc* ยังทำให้อีเอสสร้างตัวจำแนกประเภทที่ปรับเหมาะเกินไปกับชุดข้อมูลฝึกอีกด้วย ในทางกลับกัน *BdOfGenErr* เป็นการประมาณค่าของประสิทธิภาพโดยนัยทั่วไปของเอสวีเอ็มซึ่งพิจารณาทั้งค่าความผิดพลาดของการฝึกและมิติวีซี ทำให้สามารถหลีกเลี่ยงปัญหาการปรับเหมาะเกินไปได้ ส่งผลให้ได้ประสิทธิภาพที่ดีกว่า

แม้ว่า *BdOfGenErr* สามารถปรับปรุงประสิทธิภาพของการจำแนกประเภทข้อมูลที่ดีขึ้น แต่ว่า *5SubsetOnTrnAcc* มีประสิทธิภาพสูงกว่า เวลาที่เราใช้ *5SubsetOnTrnAcc* เป็นฟังก์ชันจุดประสงค์ ค่าเฉลี่ยค่าความถูกต้องจะมีค่าสูงกว่าค่าของฟังก์ชันจุดประสงค์อื่นๆ นอกจากนั้นค่าความถูกต้องสำหรับชุดข้อมูลทดสอบของ *5SubsetOnTrnAcc* ก็ยังสูงกว่าค่าของ *GridSearch* อย่างมีนัยสำคัญในชุดข้อมูลหลายชุด การแบ่งข้อมูลฝึกออกเป็น 5 เซตย่อยทำให้แก้ปัญหาการปรับเหมาะเกินไปได้อย่างดี ไฮเปอร์พารามิเตอร์ที่ทำงานได้ดีสำหรับเซตย่อยทั้งห้าจะมีโอกาสน้อยที่จะปรับเหมาะเกินไปกับข้อมูลฝึก ดังนั้น *5SubsetOnTrnAcc* จึงเป็นฟังก์ชันจุดประสงค์ที่ดีสำหรับเคอร์เนลที่นำเสนอ

นอกจากนี้เรายังได้ทำการทดลองใช้ *5SubsetOnBdOfGenErr* แต่ค่าความถูกต้องเฉลี่ยไม่ดีกว่า *BdOfGenErr* ซึ่งอาจมาจากเหตุผลที่ว่าขอบเขตของค่าผิดพลาดโดยนัยทั่วไปนี้ได้พิจารณาเรื่องการปรับเหมาะเกินไปไว้เรียบร้อยแล้วโดยให้น้ำหนักทั้งค่าผิดพลาดของการฝึกและมิติวีซีของเอสวีเอ็ม ดังนั้นการตรวจสอบไขว้เซตย่อยจึงอาจไม่จำเป็นสำหรับขอบเขตของค่าผิดพลาดโดยนัยทั่วไป และเราจะไม่นำมาพิจารณาใช้กับอัลกอริทึมอีเอสในการทดลองหลังจากนี้

สำหรับเคอร์เนลแบบหลายสเกลที่นำเสนอนั้น ค่าความถูกต้องเฉลี่ยของฟังก์ชันจุดประสงค์แต่ละฟังก์ชันสำหรับ *n-RBF* เมื่อ n=2, 3,4 และ 5 แสดงในตารางที่ 4 – ตารางที่ 7 ผลการทดลองเหล่านี้แสดงความสามารถของฟังก์ชันจุดประสงค์แต่ละฟังก์ชันและแสดงให้เห็นว่าค่าความถูกต้องเฉลี่ยบนชุดข้อมูล 15 ชุดของ *5SubsetOnTrnAcc* ดีกว่าค่าของ *TrnAcc* และ *BdOfGenErr* และยังพบว่า *5SubsetOnTrnAcc* ยังให้ค่าความถูกต้องสูงสุดในชุดข้อมูลหลายๆ ชุด และดีกว่าของ *TrnAcc* อย่างมีนัยสำคัญในชุดข้อมูลบางชุด

เมื่อเราใช้ *BdOfGenErr* เป็นฟังก์ชันจุดประสงค์ ค่าความถูกต้องเฉลี่ยบนชุดข้อมูลทั้ง 15 ชุดมีค่าดีกว่าค่าของ *TrnAcc* นอกจากนี้ค่าความถูกต้องเฉลี่ยของ *BdOfGenErr* ก็ดีกว่าค่าของ *TrnAcc* อย่างมีนัยสำคัญในชุดข้อมูลบางชุด เช่น ThreeOfNine, TicTacToe, ParityBits และ Ionosphere อย่างไรก็ดีค่าความถูกต้องเฉลี่ยของ *BdOfGenErr* ไม่มากกว่าค่าของ *5SubsetOnTrnAcc* ในชุดข้อมูลหลายชุด กราฟค่าความถูกต้องเฉลี่ยบนชุดข้อมูล 15 ชุดของฟังก์ชันจุดประสงค์ทั้งหลายแสดงในรูปที่ 5

ตารางที่ 4 เปอร์เซนต์ความถูกต้องของเคอร์เนล 2-RBF

ชุดข้อมูล	ฟังก์ชันจุดประสงค์ในอีเอส		
	TrnAcc	*BdOfGenErr*	*5SubsetOnTrnAcc*
Checkers	**90.62**	89.07	84.35
Spiral	**100.00**	**100.00**	**100.00**
LiverDisorders	70.73	**71.31**	67.25
IndiansDiabetes	72.39	71.35	**74.86**
ThreeOfNine	53.51	**100.00****	**100.00****
TicTacToe	65.34	98.33**	**99.69****
BreastCancer	95.56	95.42	**95.99**
ParityBits	48.05	79.49**	**80.18****
SolarFlare	**83.21**	82.93	80.96
ClevelandHeart	55.55	62.22	**83.34****
Australian	**55.51**	**55.51**	**55.51**
German-org	70.80	70.00	**73.40**
Ionosphere	66.10	95.45**	**96.29****
Tokyo	89.05	87.79	**91.66**
Sonar	**92.78**	92.30	88.92
Average	73.95	83.41	**84.83**

****** คือระดับของนัยสำคัญทางสถิติที่ 0.001 สำหรับความแตกต่างระหว่าง *TrnAcc* ที่ใช้เป็นฟังก์ชันจุดประสงค์ของอีเอสกับฟังก์ชันอื่น

ตารางที่ 5 เปอร์เซนต์ความถูกต้องของเคอร์เนล 3-RBF

ชุดข้อมูล	ฟังก์ชันจุดประสงค์ในอีเอส		
	TrnAcc	*BdOfGenErr*	*5SubsetOnTrnAcc*
Checkers	**90.62**	89.07	83.82
Spiral	**100.00**	**100.00**	**100.00**
LiverDisorders	**71.02**	**71.02**	68.99
IndiansDiabetes	73.55	71.34	**75.26**
ThreeOfNine	53.51	**100.00****	**100.00****
TicTacToe	65.34	98.54**	**99.69****
BreastCancer	94.99	95.28	**96.42**
ParityBits	48.05	79.49**	**80.18****
SolarFlare	**83.21**	82.93	80.96
ClevelandHeart	55.55	71.85	**82.96****
Australian	**55.51**	**55.51**	**55.51**
German-org	72.00	70.00	**73.70**
Ionosphere	66.10	**96.30****	96.01**
Tokyo	90.19	89.47	**91.66**
Sonar	**92.30**	**92.30**	88.92
Average	74.13	84.21	**84.94**

****** คือระดับของนัยสำคัญทางสถิติที่ 0.001 สำหรับความแตกต่างระหว่าง *TrnAcc* ที่ใช้เป็นฟังก์ชันจุดประสงค์ของอีเอสกับฟังก์ชันอื่น

ตารางที่ 6 เปอร์เซนต์ความถูกต้องของเคอร์เนล 4-RBF

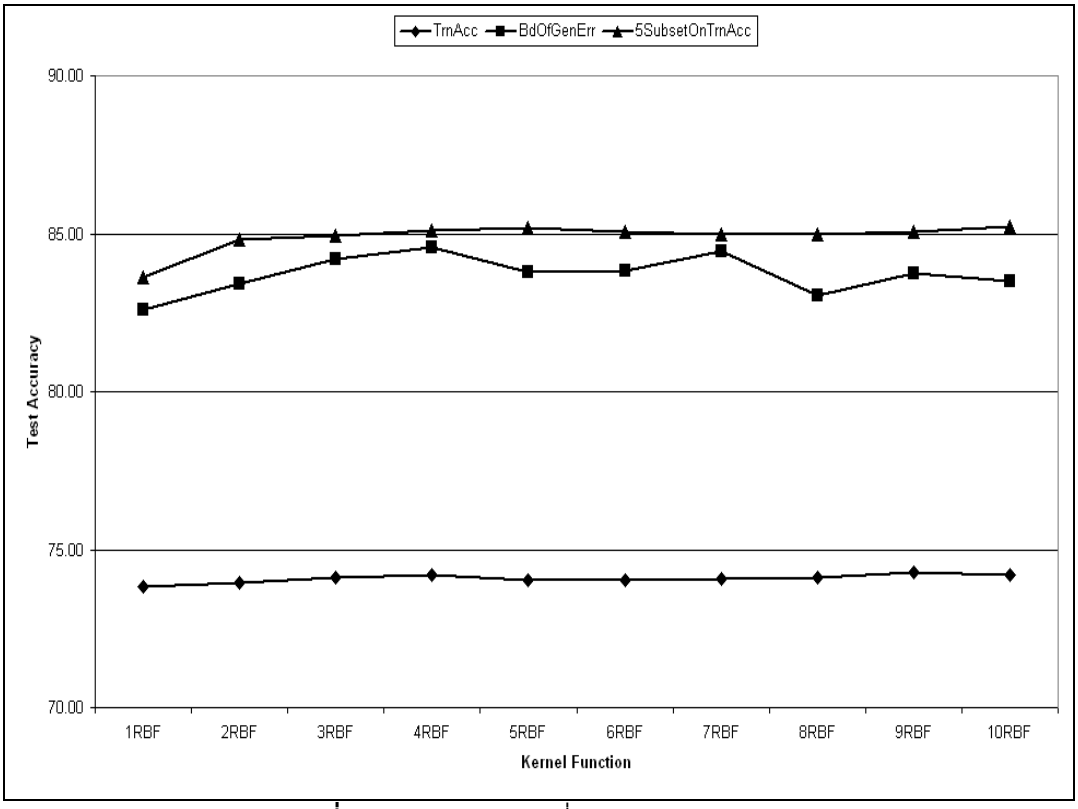
ชุดข้อมูล	ฟังก์ชันจุดประสงค์ในอีเอส		
	TrnAcc	*BdOfGenErr*	*5SubsetOnTrnAcc*
Checkers	**90.09**	88.54	83.82
Spiral	**100.00**	**100.00**	**100.00**
LiverDisorders	**70.15**	68.99	69.57
IndiansDiabetes	74.47	71.98	**75.26**
ThreeOfNine	53.51	99.80**	**100.00****
TicTacToe	65.34	98.44**	**99.69****
BreastCancer	95.85	95.85	**96.42**
ParityBits	48.05	79.49**	**80.18****
SolarFlare	**83.21**	82.93	80.96
ClevelandHeart	55.55	78.15**	**83.33****
Australian	**55.51**	**55.51**	55.22
German-org	73.50	70.00	**75.40**
Ionosphere	66.10	95.72**	**96.01****
Tokyo	89.15	90.30	**91.56**
Sonar	**92.78**	**92.78**	88.92
Average	74.22	84.57	**85.09**

****** คือระดับของนัยสำคัญทางสถิติที่ 0.001 สำหรับความแตกต่างระหว่าง *TrnAcc* ที่ใช้เป็นฟังก์ชันจุดประสงค์ของอีเอสกับฟังก์ชันอื่น

ตารางที่ 7 เปอร์เซนต์ความถูกต้องของเคอร์เนล 5-RBF

ชุดข้อมูล	ฟังก์ชันจุดประสงค์ในอีเอส		
	TrnAcc	*BdOfGenErr*	*5SubsetOnTrnAcc*
Checkers	**90.10**	88.02	83.82
Spiral	**100.00**	**100.00**	**100.00**
LiverDisorders	**70.15**	69.57	68.99
IndiansDiabetes	73.17	71.99	**74.61**
ThreeOfNine	53.51	**100.00****	**100.00****
TicTacToe	65.34	98.44**	**99.69****
BreastCancer	95.42	94.99	**96.42**
ParityBits	48.05	79.49**	**80.18****
SolarFlare	**83.12**	82.93	80.96
ClevelandHeart	55.55	68.89	**83.33****
Australian	55.65	55.51	**56.96**
German-org	71.20	70.00	**76.70***
Ionosphere	66.10	**96.01****	95.73**
Tokyo	90.82	88.94	**91.66**
Sonar	**92.30**	**92.30**	88.92
Average	74.03	83.81	**85.20**

ระดับของนัยสำคัญทางสถิติสำหรับความแตกต่างระหว่าง *TrnAcc* ที่ใช้เป็นฟังก์ชันจุดประสงค์ของอีเอสกับฟังก์ชันอื่น: : * คือ 0.01, ** คือ 0.001



รูปที่ 5 ค่าความถูกต้องเฉลี่ยบนชุดข้อมูล 15 ชุด

กราฟในรูปที่ 5 แสดงให้เห็นว่าค่าความถูกต้องเฉลี่ยบนชุดข้อมูล 15 ชุดของ *5SubsetOnTrnAcc* สูงที่สุดสำหรับทุกค่าของ *n* ใน *n-RBF* ในขณะที่ค่าความถูกต้องเฉลี่ยของ *BdOfGenErr* จะมีค่าน้อยกว่าเล็กน้อย เมื่อเราใช้ *5SubsetOnTrnAcc* เป็นฟังก์ชันจุดประสงค์ในอัลกอริทึมอีเอส พบว่าค่าความถูกต้องเฉลี่ยของ *n-RBF* จะมีค่าสูงขึ้นเมื่อจำนวนพจน์ของ *RBF* มากขึ้น ในขณะที่ค่าความถูกต้องเฉลี่ยของ *BdOfGenErr* ลดลงบ้างในชุดข้อมูลบางชุด

แม้ว่า *BdOfGenErr* จะไม่ได้ให้ค่าความถูกต้องมากกว่าค่าของ *5SubsetOnTrnAcc* แต่ *BdOfGenErr* สามารถคำนวณได้รวดเร็วกว่า เนื่องจาก *5SubsetOnTrnAcc* ต้องสร้างตัวจำแนกประเภท 5 ครั้งเพื่อประเมินไฮเปอร์พารามิเตอร์ ดังนั้น *BdOfGenErr* จึงเป็นตัวเลือกที่ดีของฟังก์ชันจุดประสงค์ซึ่งให้ผลที่ดีและง่ายต่อการนำไปใช้

ในกรณีทั่วไป การฝึกเอสวีเอ็มเป็นกระบวนการแบบออฟไลน์ (off-line) ดังนั้นเวลาฝึกอาจไม่เป็นปัญหา และในกรณีเหล่านี้ *5SubsetOnTrnAcc* จะเป็นตัวเลือกที่ดีที่สุดซึ่งให้ค่าความถูกต้องสูงสุดในชุดข้อมูลหลายชุด นอกจากนี้เราพบว่าเมื่อเราใช้ *5SubsetOnTrnAcc* เป็นฟังก์ชันจุดประสงค์ ค่าความถูกต้องเฉลี่ยบนชุดข้อมูลทั้ง 15 ชุดมีแนวโน้มเพิ่มขึ้นเมื่อเราเพิ่มจำนวนพจน์ของเคอร์เนล RBF

1.6 สรุปผลการวิจัยพัฒนาฟังก์ชันเคอร์เนลสำหรับซัพพอร์ตเวกเตอร์แมชชีน

งานวิจัยในส่วนนี้ได้นำเสนอการรวมเชิงเส้นแบบมีน้ำหนักของเคอร์เนลอาร์บีเอฟแบบหลายสเกลสำหรับซัพพอร์ตเวกเตอร์แมชชีน ในงานจำแนกประเภทข้อมูล เคอร์เนลอาร์บีเอฟเป็นเคอร์เนลที่ได้รับการใช้งานอย่างกว้างขวางเนื่องจากประสิทธิภาพที่ดี ในงานวิจัยนี้เราได้แสดงให้เห็นว่าเคอร์เนลอาร์บีเอฟสามารถปรับปรุงประสิทธิภาพยิ่งขึ้นได้อีกโดยใช้การรวมของเคอร์เนลอาร์บีเอฟหลายๆ ฟังก์ชันเข้าด้วยกันและทำให้ยืดหยุ่นได้มากขึ้นในปัญหาที่ซับซ้อน

จากนั้นเราได้นำเสนอการประยุกต์ใช้กลยุทธ์ทางวิวัฒนาการ เพื่อหาค่าเหมาะสมของไฮเปอร์พารามิเตอร์ของเอสวีเอ็ม นอกจากนั้นเราได้นำเสนอฟังก์ชันจุดประสงค์ในอีเอสคือความถูกต้องของการฝึก ขอบเขตค่าผิดพลาดโดยนัยทั่วไป และการตรวจสอบไขว้ของเซตย่อยบนค่าความถูกต้องของการฝึก ผลที่ได้แสดงให้เห็นว่าการตรวจสอบไขว้ของเซตย่อยบนค่าความถูกต้องของการฝึกเป็นฟังก์ชันจุดประสงค์ที่ดีสุด ให้ค่าความถูกต้องสูงสุด

ผลการทดลองแสดงให้เห็นถึงประสิทธิภาพของวิธีการที่นำเสนอ ซึ่งเคอร์เนลแบบหลายสเกลให้ผลที่ดีกว่าเคอร์เนลอาร์บีเอฟแบบดั้งเดิม เอสวีเอ็มที่ใช้เคอร์เนลที่นำเสนอสามารถเรียนรู้จากตัวอย่างฝึกได้ดี นอกจากนั้นกลยุทธ์ทางวิวัฒนาการมีประสิทธิภาพดีในการหาค่าเหมาะสมของไฮเปอร์พารามิเตอร์ เคอร์เนลที่นำเสนอนี้จึงเหมาะกับปัญหาที่เราไม่มีความรู้ก่อนหน้าว่าควรเลือกพารามิเตอร์เช่นความกว้างของอาร์บีเอฟอย่างไร โดยวิธีการที่นำเสนอสามารถปรับพารามิเตอร์ที่เหมาะสมได้ดี

2. การวิจัยพัฒนาฟังก์ชันเคอร์เนลสำหรับซัพพอร์ตเวกเตอร์รีเกรซชัน

หัวข้อนี้กล่าวถึงการประยุกต์ใช้เทคนิคของฟังก์ชันเคอร์เนลอาร์บีเอฟแบบหลายตัว สำหรับซัพพอร์ตเวกเตอร์รีเกรซชัน – เอสวีอาร์ (support vector regression – SVR)

ซัพพอร์เวกเตอร์แมชชีน (Support vector machine: SVM) เป็นขั้นตอนวิธีสำหรับการเรียนรู้แบบหนึ่ง ซึ่งนำเสนอโดย Vapnik et al. [Vapnik, 1998] โดยอาศัยแนวความคิดของการลดความเสี่ยงบนข้อมูลตัวอย่าง ขั้นตอนวิธีนี้ถูกนำไปใช้อย่างแพร่หลายในงานหลายด้าน เช่น การรู้จำรูปแบบ หรือ การประมาณค่าฟังก์ชัน เอสวีเอ็มจะใช้ระนาบเชิงเส้นบนปริภูมิลักษณะ (feature space) ในการแบ่งแยกข้อมูลหรือประมาณค่า ซึ่งจะใช้ฟังก์ชันเคอร์เนลที่กำหนดก่อนไว้แล้วในการคำนวณ โดยที่ฟังก์ชันเคอร์เนลเหล่านี้สอดคล้องตามทฤษฎีบทของ Mercer (Mercer's Theorem) [Schölkopf et al.,1998; Ayat et al., 2001] ฟังก์ชันเคอร์เนลจะแปลงเวกเตอร์นำเข้าไปยังปริภูมิที่มีมิติสูงขึ้น ที่ซึ่งสามารถแบ่งแยกข้อมูลได้โดยใช้ระนาบเชิงเส้น [Ayat et al., 2001] ฟังก์ชันเคอร์เนลที่ใช้กันมีหลายชนิด เช่น เคอร์เนลเชิงเส้น เคอร์เนลพหุนาม และ เคอร์เนลเรเดียลเบสิสฟังก์ชัน (Radial Basis Function: RBF) เคอร์เนลแต่ละตัวจะเหมาะสมกับงานที่แตกต่างกันออกไป โดยทั่วไปแล้วการเลือกใช้ฟังก์ชันเคอร์เนลตัวใดจะถูกกำหนดโดยผู้ใช้งาน หรือโดยอาศัยความรู้ก่อนหน้า [Kecman, 2001]

เคอร์เนลอาร์บีเอฟเป็นฟังก์ชันเคอร์เนลตัวหนึ่งที่ใช้ได้ดีในหลายๆ ปัญหา แต่ในปัญหาที่มีความซับซ้อนมากๆ เคอร์เนลอาร์บีเอฟนี้ยังคงไม่สามารถให้ประสิทธิภาพที่พึงพอใจได้ ดังนั้นในงานวิจัยนี้จึงได้นำเสนอการปรับปรุงประสิทธิภาพของการประมาณค่า โดยใช้การรวมกันของเคอร์เนลอาร์บีเอฟหลายตัว ที่ความกว้างต่างกัน และมีการใช้ค่าถ่วงน้ำหนัก ค่าความกว้างและค่าถ่วงน้ำหนักของอาร์บีเอฟแต่ละตัว รวมถึงค่าเบี่ยงเบนของการประมาณค่า (Deviation of Approximation) และ พารามิเตอร์สามัญ (Regularization Parameter) ของเอสวีเอ็ม เป็นพารามิเตอร์ที่ต้องทำการปรับให้เหมาะสม โดยทั่วไปแล้ว พารามิเตอร์เหล่านี้จะถูกปรับค่าโดยใช้วิธีการค้นหาแบบกริด ซึ่งทำโดยการเปลี่ยนค่าของพารามิเตอร์ทีละขั้นในช่วงของค่าที่เป็นไปได้ แต่วิธีการค้นหาแบบนี้จะใช้เวลานาน โดยเฉพาะเมื่อมีพารามิเตอร์จำนวนมาก

มีงานวิจัยที่นำขั้นตอนวิธีเชิงวิวัฒน์มาใช้กับฟังก์ชันเคอร์เนลของเอสวีเอ็ม เช่น Howley และ Madden [Howley & Madden, 2005] นำเสนอการโปรแกรมเชิงพันธุกรรม สำหรับการสร้างฟังก์ชันเคอร์เนลของเอสวีเอ็ม แต่วิธีการของพวกเขาไม่ได้รับรองว่าฟังก์ชันเคอร์เนลที่ได้ออกมาจะสอดคล้องตามทฤษฎีบทของ Mercer แต่สำหรับวิธีการที่นำเสนอในงายวิจัยนี้ใช้การรวมกันของเคอร์เนลอาร์บีเอฟหลายตัวที่ความกว้างต่างกัน และได้สามารถพิสูนจ์ได้ว่ามันสอดคล้องตามทฤษฎีบทของ Mercer นอกจากนี้ยังมีงานวิจัยที่พยายามแก้ปัญหาการเลือกค่าพารามิเตอร์ อาทิเช่น ในงานวิจัยของ [Chapelle et al., 2002] ในปี ค.ศ. 2002 พวกเขาได้นำเสนอ

ขั้นตอนวิธีที่ใช้เกรเดียน (Gradient) ในการปรับพารามิเตอร์ของเอสวีเอ็ม ซึ่งเป็นวิธีการที่มีประสิทธิภาพ แต่มีรายละเอียดในการคำนวณ$มาก ทำให้ขั้นตอนวิธีนี้ยุ่งยากซับซ้อน

กลยุทธเชิงวิวัฒน์แบบปรับตัวได้โดยใช้เมตริกซ์ความแปรปรวนร่วม (Covariance Matrix Adaptation Evolution Strategy: CMA-ES) [Friedrichs & Igel, 2004] เป็นอีกวิธีหนึ่งที่ใช้หาค่าพารามิเตอร์ของฟังก์ชันเคอร์เนล นอกจากนั้นยังมีการใช้ขั้นตอนวิธีเชิงพันธุกรรม [Xuefeng & Fang, 2002] กลยุทธเชิงวิวัฒน์แบบขนานไม่เข้าจังหวะ (Asynchronous Parallel Evolution Strategy) [Runarsson, 2004], ขั้นตอนวิธีการโปรแกรมแบบไม่เป็นเชิงเส้น [Schittkowski, 2005] และ การหาค่าเหมาะสมโดยรวมโดยใช้แบบจำลองเป็นพื้นฐาน (Model-based Global Optimization) [Fröhlich & Zell, 2005] ซึ่งขั้นตอนวิธีเหล่านี้เป็นตัวอย่างของวิธีการที่มีผู้นำเสนอให้นำมาใช้ในการหาค่าพารามิเตอร์ของเอสวีเอ็ม

จากการศึกษาพบว่ากลยุทธ์เชิงวิวัฒน์ (Evolutionary Strategy: ES) เป็นวิธีการที่ง่ายและเหมาะสมจะนำมาใช้ในการปรับพารามิเตอร์ของเอสวีเอ็ม วิธีการนี้ใช้กระบวนการสุ่มเพื่อเลือกพารามิเตอร์ที่เหมาะสม และใช้การรวมกันใหม่ (Recombination) และการกลายพันธุ์ (Mutation) เพื่อสร้างคำตอบใหม่ๆ ขึ้นมา ดังนั้นงานวิจัยนี้จึงเลือกใช้อีเอสเพื่อปรับค่าพารามิเตอร์ของเอสวีเอ็มและฟังก์ชันเคอร์เนล ในส่วนของฟังก์ชันวัตถุประสงค์ (Objective Function) ซึ่งเป็นส่วนสำคัญในขั้นตอนวิธีเชิงวิวัฒน์ งานวิจัยนี้ได้นำเสนอการใช้คุณสมบัติความเสถียรของเอ็สวีเอ็มบนปัญหาการหาค่าถดถอย เพื่อประเมินค่าความเหมาะสมของพารามิเตอร์

2.1 ซัพพอร์ตเวกเตอร์แมชชีนและฟังก์ชันเคอร์เนล

ซัพพอร์ตเวกเตอร์แมชชีน สามารถแบ่งออกได้เป็น ซัพพอร์ตเวกเตอร์แมชชีนที่ใช้ในการแบ่งแยกข้อมูล และซัพพอร์ตเวกเตอร์แมชชีนที่ในการหาค่าถดถอย ในงานวิจัยส่วนนี้ เราสนใจซัพพอร์ตเวกเตอร์แมชชีนที่ใช้ในปัญหาการหาค่าถดถอย (Support Vector Regression: SVR) ซึ่งเป็นวิธีการหนึ่งที่มีประสิทธิภาพในการประมาณค่าจำนวนจริง สำหรับปัญหาที่ไม่เป็นเชิงเส้น เอสวีอาร์ จะใช้วิธีการของเคอร์เนลเพื่อสร้างตัวประมาณค่าที่ซับซ้อนและไม่เป็นเชิงเส้นในปริภูมินำเดิม ในส่วนนี้ เอสวีอาร์ และ ฟังก์ชันเคอร์เนลจะถูกบรรยายโดยย่อ

2.1.1 ซัพพอร์ตเวกเตอร์แมชชีนบนปัญหาการหาค่าถดถอย

สมมติข้อมูลฝึกอยู่ในรูป $\{(\mathbf{x}_1, y_1),\ldots,(\mathbf{x}_l, y_l)\} \subset X \times R$ โดยที่ X แทนปริภูมิของข้อมูลนำเข้า ในกรณีของเอสวีอาร์ จุดมุ่งหมายของเราคือ ต้องการหาฟังก์ชัน $f(\mathbf{x})$ ที่ให้ค่าต่างจากค่าที่แท้จริง y_i ไม่เกิน ε สำหรับทุกๆข้อมูลฝึก และในขณะเดียวกันมันแบนราบมากที่สุดเท่าที่จะเป็นได้ [Smola & Schölkopf, 1998] หรือกล่าวอีกนัยหนึ่งได้ว่า เราไม่สนใจความผิดพลาด ตราบใดที่ยังไม่เกินค่า ε แต่เราจะไม่ยอมรับความเบี่ยงเบนใดๆที่มากกว่าค่านี้

ในกรณีที่ f เป็นฟังก์ชันเชิงเส้น และเขียนในรูปของสมการได้ดังนี้

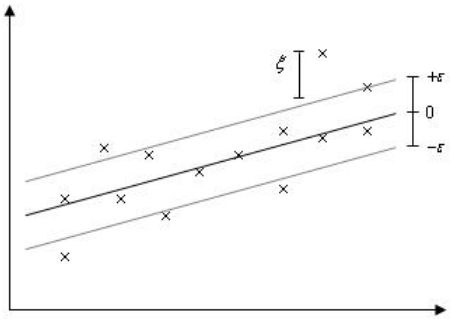
$$f(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle + b \text{ เมื่อ } \mathbf{w} \in X \text{ , } b \in R \text{ .} \tag{27}$$

ความแบนราบในกรณีนี้หมายความว่า เราต้องการ $\mathbf{w}$ ที่มีค่าน้อยๆ วิธีการหนึ่งที่ทำได้คือการหาขนาดที่เล็กที่สุดของ $\mathbf{w}$ นั่นคือ หาค่าต่ำสุดของ $\|\mathbf{w}\|^2 = \langle \mathbf{w}, \mathbf{w} \rangle$

ในบางครั้งเราอาจยอมให้มีความผิดพลาดเกิดขึ้นได้บ้าง กรณีนี้จะใช้ฟังก์ชันของค่าผิดพลาดแบบระยะขอบอ่อน (Soft Margin Loss Function) ซึ่งมีการใช้ตัวแปรช่วย (Slack Variables) ξ_i , ξ_i^* เพื่อจัดการกับเงื่อนไขอื่นๆ ที่ไม่สอดคล้องกับปัญหาการหาค่าที่เหมาะสม [Howley & Madden, 2005] ทำให้ปัญหาการหาค่าที่เหมาะสมของเราเขียนได้ดังต่อไปนี้

$$
\begin{aligned}
\text{minimize} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{l}(\xi_i + \xi_i^*) \\
\text{subject to} \quad & y_i - \langle \mathbf{w}, \mathbf{x}_i \rangle - b \leq \varepsilon + \xi_i \\
& \langle \mathbf{w}, \mathbf{x}_i \rangle + b - y_i \leq \varepsilon + \xi_i^* \\
& \xi_i, \ \xi_i^* \geq 0.
\end{aligned} \tag{28}
$$

โดยที่ ค่าคงที่ $C > 0$ เป็นตัวกำหนดการแลกเปลี่ยนระหว่างความแบบราบของ f และจำนวนตัวอย่างซึ่งเบี่ยงเบนจากค่าที่แท้จริงไปมากกว่า ε โดยส่วนมากปัญหาการหาค่าที่เหมาะสมนี้ สามารถแก้ได้ง่ายขึ้นในรูปแบบที่เป็นคู่กันของมัน (Dual Formulation) [Smola & Schölkopf, 1998] ตัวอย่างของเอสวีอาร์เชิงเส้นแสดงในรูปที่ 6



รูปที่ 6 ตัวอย่างของเอสวีอาร์เชิงเส้นที่เป็นแบบระยะขอบอ่อน

2.1.2 ฟังก์ชันเคอร์เนล

ในกรณีทั่วไป การหาระนาบเชิงเส้นที่แท้จริงในปริภูมินำเข้ามักจะไม่สามารถทำได้ แต่มีวิธีการที่สำคัญอันหนึ่งที่ทำให้เอสวีอาร์สามารถสร้างฟังก์ชันการประมาณค่าแบบไม่เป็นเชิงเส้นที่ซับซ้อนขึ้นในปริภูมิเดิมได้ วิธีการนี้ทำโดยการส่งผ่านปริภูมิขำเข้าไปยังปริภูมิแต่งเติมทีมีมิติสูงขึ้นโดยใช้ ฟังก์ชันส่งผ่าน $\Phi : X \to R^N$ จากนั้นทำการหาฟังก์ชันการประมาณค่าเชิงเส้นในปริภูมิใหม่นี้ [Schölkopf & Smola, 2002] วิธีการนี้สามารถทำได้โดยการแทนตัวอย่างฝึก $\mathbf{x}_i$ ด้วย $\Phi(\mathbf{x}_i)$ แต่คุณสมบัติอีกอย่างหนึ่งของเอสวีอาร์ คือ ไม่จำเป็นต้องรู้ฟังก์ชันการส่งผ่าน Φ แค่ผลคูณภายในบนปริภูมิแต่งเติมที่เรียกว่าฟังก์ชันเคอร์เนลเท่านั้นที่ต้องกำหนด

$$K(\mathbf{x}, \mathbf{y}) = \Phi(\mathbf{x}) \cdot \Phi(\mathbf{y}) \tag{29}$$

ตัวอย่างของฟังก์ชันเคอร์เนลแสดงในตารางที่ 1 แต่ละเคอร์เนลสอดคล้องกับปริภูมิแต่งเติมที่แตกต่างกันออกไป และเนื่องจากไม่รู้ฟังก์ชันการส่งผ่านไปยังปริภูมิแต่งเติมที่แท้จริง ฟังก์ชันเชิงเส้นที่เหมาะสมจึงสามารถหาได้อย่างมีประสิทธิภาพในปริภูมิแต่งเติมที่มีมิติสูงมากๆ

โดยทั่วไป เราไม่รู้ฟังก์ชันซึ่งส่งผ่านปริภูมินำเข้าไปยังปริภูมิแต่งเติม แต่อย่างไรก็ตาม การมีอยู่จริงของฟังก์ชันเหล่านี้สามารถรับรองได้โดยใช้ทฤษฎีบทของ Mercer [Kecman, 2001; Schölkopf et al.,1998] ทฤษฎีบทของ Mercer แสดงในรูปที่ 7

ทฤษฎีบทของเมอร์เซอร์ ฟังก์ชันสมมาตร $K(x,y)$ ใดๆในปริภูมินำเข้า สามารถเขียนให้อยู่ในรูปของผลคูณภายในบนปริภูมิแต่งเติมได้ ถ้า

$$\iint K(x,y)\,g(x)\,g(y)\,dx\,dy \;\geq\; 0$$

เป็นจริงสำหรับทุก $g \neq 0$ ที่ซึ่ง $\int g^2(u)\,du \;<\; \infty$ นั่นคือ ฟังก์ชัน K สามารถขยายในรูปแบบของ Φ_i

$$K(x,y) \;=\; \sum_{i=1}^{\infty} \lambda_i\, \Phi_i(x)\; \Phi_i(y)$$

โดยที่ $\lambda_i \geq 0$ ในกรณีนี้ การส่งผ่านจากปริภูมินำเข้าไปยังปริภูมิแต่งเติม สามารถเขียนได้เป็น

$$\Phi : x \rightarrow (\sqrt{\lambda_1}\;\Phi_1(x)\;,\; \sqrt{\lambda_2}\;\Phi_2(x), \ldots)$$

ดังนั้น K สามารถเขียนในรูปผลคูณภายในได้

$$\Phi(x)\cdot\Phi(y) \;=\; \sum_{i=1}^{\infty} \lambda_i\, \Phi_i(x)\; \Phi_i(y) \;=\; K(x,y)$$

รูปที่ 7 ทฤษฎีบทของ Mercer

ฟังก์ชันที่จะนำมาใช้เป็นเคอร์เนล ควรจะสอดคล้องตามทฤษฎีบทของ Mercer [Taylor & Cristianini, 2004] นอกจากนั้นยังมีตัวดำเนินการบนฟังก์ชันเคอร์เนลซึ่งยังคงรักษาคุณสมบัติของเคอร์เนล ตัวดำเนินการเหล่านี้แสดงในทฤษฎีบทย่อยที่ 1

ทฤษฎีบทย่อยที่ 1. (คุณสมบัติปิด) ให้ K_1 และ K_2 เป็นฟังก์ชันเคอร์เนลบน $X \times X$, $X \subseteq R^n$, $a \in R^+$, $f(\cdot)$ เป็นฟังก์ชันของค่าจำนวนจริงบน X, $\Phi : X \rightarrow R^N$, K_3 เป็นฟังก์ชันเคอร์เนลบน $R^N \times R^N$, และ B เป็นเมตริกซ์สมมาตรขนาด $n \times n$ ฟังก์ชันต่อไปนี้ยังคงสามารถใช้เป็นเคอร์เนล [Taylor & Cristianini, 2004]:

(i) $K(x,y) = K_1(x,y) + K_2(x,y)$,

(ii) $K(x,y) = aK_1(x,y)$,

(iii) $K(x,y) = K_1(x,y)K_2(x,y)$,

(iv) $K(x,y) = f(x)f(y)$,

(v) $K(x,y) = K_3(\Phi(x),\Phi(y))$,

(vi) $K(x,y) = x'By$.

ทฤษฎีบทย่อยนี้แสดงให้เห็นว่า ฟังก์ชันเคอร์เนลมีคุณสมบัติปิดบนการบวก และการคูณด้วยค่าคงที่ และยิ่งไปกว่านั้น ฟังก์ชันเคอร์เนลที่ซับซ้อนสามารถสร้างขึ้นจากฟังก์ชันเคอร์เนลพื้นฐาน คุณสมบัติบางข้อจากทฤษฎีบทย่อยนี้จะถูกนำมาใช้สำหรับการรวมกันของเคอร์เนลอาร์บีเอฟหลายตัวในส่วนถัดไป

2.2 การรวมเคอร์เนลอาร์บีเอฟหลายตัวแบบเสถียรและปรับตัวได้

ในหัวข้อนี้ การรวมเคอร์เนลอาร์บีเอฟหลายตัวแบบเสถียรและปรับตัวได้ถูกนำเสนอสำหรับเอสวีอาร์ การการรวมเคอร์เนลอาร์บีเอฟหลายตัวจะได้ฟังก์ชันเคอร์เนลที่มีความยืดหยุ่นต่อปัญหาที่เราสนใจมากขึ้น จากนั้น กลยุทธเชิงวิวัฒน์จะถูกปรับใช้ในการหาค่าพารามิเตอร์ของเอสวีอาร์ และ พารามิเตอร์ของฟังก์ชันเคอร์เนลที่นำเสนอ แต่เคอร์เนลที่นำเสนอนั้นอาจจะสอดคล้องกับข้อมูลฝึกจนเกินไปได้ ดังนั้นคุณสมบัติความเสถียร

ของเอสวีอาร์จึงถูกนำมาใช้เป็นฟังก์ชันวัตถุประสงค์ ในกระบวนการเชิงวิวัฒน์ เพื่อหลีกปัญหาการสอดคล้องกับข้อมูลฝึกจนเกินไปดังกล่าว

การรวมเคอร์เนลอาร์บีเอฟหลายตัว

เคอร์เนลอาร์บีเอฟถูกนำไปใช้อย่างแพร่หลายในปัญหาหลายๆด้าน เคอร์เนลนี้ใช้ระยะทางแบบยูคลิด (Euclidean Distance) ระหว่างสองจุดในปริภูมิเดิม เพื่อหาความสัมพันธ์ในปริภูมิแต่งเติม [Ayat et al., 2001] ค่าความสัมพันธ์นี้ค่อนข้างต่อเนื่อง และมีพารามิเตอร์เพียงตัวเดียวสำหรับปรับความกว้างของเคอร์เนลอาร์บีเอฟ ทำให้มีประสิทธิภาพไม่เพียงพอสำหรับบางปัญหาที่มีความซับซ้อน เพื่อที่จะได้เคอร์เนลที่มีประสิทธิภาพดีขึ้น การรวมเคอร์เนลอาร์บีเอฟหลายตัวที่ความกว้างต่างกันจึงถูกนำเสนอ เคอร์เนลดังกล่าวแสดงในสมการต่อไปนี้

$$K(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{n} a_i K(\mathbf{x},\mathbf{y},\gamma_i) \tag{30}$$

โดยที่ n เป็นจำนวนเต็มบวก, $a_i \geq 0$ สำหรับ $i = 1,...,n$ เป็นค่าถ่วงน้ำหนักซึ่งเป็นค่าคงที่ที่ไม่เป็นจำนวนลบ, และ

$$K(\mathbf{x},\mathbf{y},\gamma_i) = \exp(-\gamma_i \left\|\mathbf{x}-\mathbf{y}\right\|^2) \tag{31}$$

เป็นเคอร์เนลอาร์บีเอฟที่ความกว้าง γ_i เมื่อ $i = 1,...,n$

การรวมเคอร์เนลอาร์บีเอฟหลายตัวนี้ยังคงสอดคล้องตามทฤษฎีบทของ Mercer อาร์บีเอฟเป็นเคอร์เนลที่สอดคล้องกับทฤษฎีของ Mercer ที่รู้จักกันเป็นอย่างดี ดังนั้นการรวมแบบเชิงเส้นที่ไม่เป็นลบของอาร์บีเอฟยังคงสามารถใช้เป็นฟังก์ชันเคอร์เนลได้ตามทฤษฎีบทย่อยที่ 1 ข้อ (i) และ (ii) ยิ่งไปกว่านั้น เคอร์เนลนี้ยังมีความยืดหยุ่นมาขึ้น เนื่องจากมันมีจำนวนพารามิเตอร์ที่ปรับค่าได้เพิ่มมากขึ้น

ดังเช่นที่แสดงใน (30) เมื่อใช้อาร์บีเอฟ n ตัว จะมีพารามิเตอร์ $2n$ ตัว (n ตัว สำหรับการปรับค่าถ่วงน้ำหนัก และ n ตัว สำหรับความกว้างของอาร์บีเอฟ) แต่เราสังเกตพบว่า จำนวนของพารามิเตอร์สามารถลดลงเหลือ $2n-1$ โดยกำหนดค่าถ่วงน้ำหนักของอาร์บีเอฟตัวแรกเป็น 1 การรวมเคอร์เนลอาร์บีเอฟหลายตัวจะอยู่ในรูปสมการดังต่อไปนี้

$$\mathrm{K}(\mathbf{x},\mathbf{y}) = K(\mathbf{x},\mathbf{y},\gamma_0) + \sum_{i=1}^{n-1} a_i K(\mathbf{x},\mathbf{y},\gamma_i) \tag{32}$$

ซึ่งการรวมเคอร์เนลอาร์บีเอฟหลายตัวแบบนี้ จะถูกใช้ในหัวข้อที่จะกล่าวถึงต่อไป

2.3 กลยุทธเชิงวิวัฒน์

กลยุทธเชิงวิวัฒน์ (อีเอส) ได้ถูกกล่าวถึงในหัวข้อที่ 1.1.2 และ 1.3 เพื่อหาค่าพารามิเตอร์ที่เหมาะสมสำหรับเอส-วีเอ็ม ในทำนองเดียวกัน ในหัวข้อนี้อีเอสจะถูกนำมาใช้เพื่อให้ได้ค่าพารามิเตอร์ของเอสวีอาร์และฟังก์ชันเคอร์เนลที่เหมาะสม อันที่จริงแล้วอีเอสมีอยู่หลายแบบ แต่ในงานวิจัยนี้เลือกใช้ ($\mu+\lambda$)-อีเอส ที่ใช้ประชาการรุ่นพ่อแม่ μ ตัว ในการสร้างประชากรรุ่นลูก λ ตัว จากนั้นทั้งรุ่นพ่อแม่และรุ่นลูกแข่งขันกันอย่างเท่าเทียม เพื่อที่จะมีชีวิตรอดไปเป็นประชากรในรุ่นถัดไป [deDoncker et al., 1996] ในการทดลอง (5+10)-อีเอส ถูกนำมาใช้ในการปรับพารามิเตอร์ของเรา และอัลกอริทึมนี้แสดงในรูปที่ 8

```
t = 0;
initialization( v_1,..., v_5 , σ );
evaluate  f(v_1),..., f(v_5) ;
while (t < 1500) do
    for i =1 to 10 do
        v'_i  = recombination( v_1,..., v_5 );
        v'_i  = mutate (v'_i) ;
        evaluate  f(v'_i) ;
    end
    ( v_1,..., v_5 )                                  =
select( v_1,..., v_5 , v'_1,..., v'_10 )
    σ = mutate_σ (σ) ;
    t = t+1;
end
```

รูปที่ 8 ขั้นตอนวิธีของ (5+10)-อีเอส

ขั้นตอนวิธีนี้ใช้คำตอบ 5 ตัว เพื่อสร้างคำตอบขึ้นมาใหม่อีก 10 ตัว โดยวิธีการรวมกันใหม่ (Recombination) คำตอยใหม่ๆจะถูกกลายพันธุ์ (Mutation) และ ประเมินค่า (Evaluate) แต่คำตอบที่ดีที่สุดเพียงแค่ 5 ตัว จะถูกเลือกจากคำตอบทั้งหมด 5+10 ตัว เพื่อไปเป็นประชากรรุ่นพ่อแม่ในรอบถัดไป กระบวนการนี้จะถูกทำซ้ำ จนกระทั่งจำนวนรอบครบตามที่กำหนดเอาไว้ หรือ จนกว่าจะได้คำตอบที่พึงพอใจ

-- *การเริ่มต้น:* ให้ $\vec{v}$ เป็นเวกเตอร์ของค่าจำนวนจริงที่ไม่เป็นลบ ซึ่งมี $2n+1$ มิติ เวกเตอร์ $\vec{v}$ อยู่ในรูปแบบ

$$\vec{v} = (C, \varepsilon, \gamma_0, a_1, \gamma_1, a_2, \gamma_2, \ldots, a_{n-1}, \gamma_{n-1}) \tag{33}$$

โดยที่ C เป็นพารามิเตอร์สามัญ, ε เป็นส่วนเบี่ยงเบนของการประมาณค่า, γ_i เมื่อ $i = 0,...,n-1$ เป็นความกว้างของอาร์บีเอฟ, a_i เมื่อ $i = 1,...,n-1$ เป็นค่าถ่วงน้ำหนัก, และ n เป็นจำนวนอาร์บีเอฟ

ขั้นตอนวิธี (5+10)-อีเอส เริ่มต้นที่ *t*=0 เลือกคำตอบ 5 ตัว $\vec{v}_1,..., \vec{v}_5$ และ ส่วนเบี่ยงเบนมาตรฐาน $\vec{\sigma} \in R_+^{2n+1}$ โดยใช้วิธีการสุ่ม คำตอบเริ่มต้น 5 ตัวนี้จะถูกประเมินโดยการคำนวนค่าความเหมาะสม (Fitness) เป้าหมายเพื่อให้ได้ $\vec{v}$ ที่ให้ค่าฟังก์ชันวัตถุประสงค์ $f(\vec{v})$ ที่เหมาะสม

-- *การรวมกันใหม่:* ในแต่ละรอบ คำตอบที่ดีที่สุด 5 ตัว จะถูกกำหนดค่าความน่าจะเป็นที่จะถูกเลือกไปสร้างคำตอบใหม่ คำตอบเหล่าที่จะถูกเรียงลำดับตามค่าฟังก์ชันวัตถุประสงค์ โดยที่ $\vec{v}_i$ มีค่าความเหมาะสมสูงกว่า $\vec{v}_{i+1}$ จากนั้นค่าความน่าจะเป็นของแต่ละคำตอบจะถูกกำหนดโดย

$$\begin{aligned} \text{Prob}(\vec{v}_i) &= \frac{\mu-(i-1)}{\sum_{j=1}^{\mu} j} \\ &= \frac{\mu-(i-1)}{\mu(\mu+1)/2} \\ &= \frac{2}{\mu}\left(1-\frac{i}{\mu+1}\right) \end{aligned} \tag{34}$$

เมื่อ i = 1, 2, ..., μ , และ μ เป็นจำนวนคำตอบที่เหมาะสมสุด ในกรณีนี้ μ มีค่าเป็น 5

หลังจากนั้น สองคำตอบใดๆ จะถูกเลือกขึ้นมาอย่างสุ่มตามความน่าจะเป็นของ 5 คำตอบเดิมนี้ จากนั้นคำนวณค่าเฉลี่ยของสองคำตอบนี้เพื่อใช้เป็นคำตอบใหม่ วิธีการนี้เรียกว่า การรวมกันใหม่แบบค่ากลางโดยรวม (Global Intermediary Recombination Method) และจะใช้ในการสร้างคำตอบใหม่อีก 10 ตัว

-- *การกลายพันธุ์:* เวกเตอร์ $\vec{v}_i'$ เมื่อ i =1,...,10 จะถูกกลายพันธุ์ โดยการบวกด้วย ($z_1, z_2, ..., z_{2n+1}$), และ z_i เป็นค่าจากการสุ่มบนการกระจายตัวปรกติ (Normal Distribution) ที่มีค่าเฉลี่ย 0 และความแปรปรวน σ_i^2

$$\textit{mutate}(\vec{v}) = (C + z_1, \varepsilon + z_2, \gamma_0 + z_3, \ldots, a_{n-1} + z_{2n}, \gamma_{n-1} + z_{2n+1}) \quad (35)$$
$$z_i \sim N_i(0, \sigma_i^2)$$

นอกจากนั้น ในแต่ละรอบ ค่าเบี่ยงเบนมาตรฐานจะถูกปรับโดย

$$\textit{mutate}_{\sigma}(\vec{\sigma}) = (\sigma_1 \cdot e^{z_1}, \sigma_2 \cdot e^{z_2}, \ldots, \sigma_{2n+1} \cdot e^{z_{2n+1}}) \quad (36)$$
$$z_i \sim N_i(0, \tau^2),$$

เมื่อ τ เป็นค่าคงที่ใดๆ

2.4 การประเมินโดยใช้ความเสถียร

ส่วนสำคัญที่สุดและเป็นส่วนที่ยากที่สุดอันหนึ่งของขั้นตอนวิธีเชิงวิวัฒน์ คือ จะกำหนดฟังก์ชันวัตถุประสงค์อย่างไรให้เหมาะสมกับงานที่เรากำลังสนใจ ในกรณีของการประเมินค่าพารามิเตอร์ มีหลายวิธีในการกำหนดฟังก์ชันวัตถุประสงค์ โดยทั่วไปจะใช้ ค่าเปอร์เซ็นต์ความผิดพลาด ผลรวมความผิดพลาดกำลังสอง หรือค่าเฉลี่ยความผิดพลาดกำลังสอง เพื่อวัดประสิทธิภาพของการหาค่าถดถอย แต่ฟังก์ชันเหล่านี้อาจจะสอดคล้องกับข้อมูลฝึกจนเกินไปได้ บางครั้งข้อมูลฝึกของเรามีข้อมูลรบกวนมาปน ซึ่งจะทำให้แบบจำลองที่เรียนรู้ออกมามีความผิดพลาด

ด้วยเหตุนี้ คุณสมบัติความเสถียรของเอสวีอาร์จึงถูกพิจารณานำมาใช้เป็นฟังก์ชันวัตถุประสงค์ในกระบวนการรเชิงวิวัฒน์ ในงานวิจัยนี้จะใช้หลักการของความเสถียรที่เสนอโดย Bousquet & Elisseeff [2002] พวกเขาได้นิยามเกี่ยวกับคุณสมบัติความเสถียรของขั้นตอนวิธีสำหรับการเรียนรู้ และ แสดงถึงวิธีการใช้คุณสมบัติเหล่านี้เพื่อให้ได้ขอบเขตของความผิดพลาดโดยทั่วไป [Bousquet & Elisseeff, 2002] วิธีการของพวกเขาสามารถนำไปใช้ดีทั้งในกรอบงานการหาค่าถดถอยและการจำแนกข้อมูล ในงานวิจัยนี้เราสนใจปัญหาการหาค่าถดถอย และจะใช้คุณสมบัติความเสถียรของเอสวีอาร์ เพื่อหลีกเลี่ยงการเหมาะสมกับข้อมูลฝึกมากเกินไปในกระบวนการเชิงวิวัฒน์

กำหนดให้ $K(\cdot)$ เป็นเคอร์เนลที่รู้ขอบเขตของค่าที่เป็นไปได้ โดยที่ $K(\mathbf{x}, \mathbf{y}) \le \kappa^2$ และ $y \in [0, B]$ ขอบเขตของความผิดพลาดที่ความน่าจะเป็นน้อยกว่า $1-\delta$ บนข้อมูลที่เก็บรวบรวมมาอย่างสุ่มขนาด m สามารถคำนวณได้โดย

$$R \le R_{emp} + \frac{\kappa^2}{\lambda m} + \left(\frac{2\kappa^2}{\lambda} + \kappa \sqrt{\frac{B}{\lambda}} \right) \sqrt{\frac{\ln(1/\delta)}{2m}} \quad (37)$$

เมื่อ R เป็นค่าความผิดพลาดโดยทั่วไป, R_{emp} เป็นค่าความผิดพลาดที่ได้จากการทดลอง, และ λ เป็นพารามิเตอร์สามัญของเอสวีอาร์ โดย $\lambda = 1/C$

การคำนวณทางขวาของอสมการ (37) จะใช้เป็นฟังก์ชันวัตถุประสงค์ในการประเมินพารามิแตอร์ของเอสวีอาร์ และ การรวมเคอร์เนลอาร์บีเอฟหลายตัวที่เรานำเสนอ เราสันนิษฐานว่าเซตของพารามิเตอร์ที่เหมาะสมควรจะให้ค่าขอบเขตของความผิดพลาดต่ำ ฟังก์ชันวัตถุประสงค์นี้จะถูกทดสอบในส่วนถัดไป

2.5 ผลการทดลอง

เพื่อที่จะประเมินประสิทธิภาพของวิธีการที่นำเสนอ เอสวีอาร์ที่ใช้การรวมเคอร์เนลอาร์บีเอฟหลายตัวแบบเสถียรและปรับตัวได้จะถูกสอนและทดสอบบนชุดข้อมูล 4 ชุดที่เป็นปัญหาการหาค่าถดถอย จากคลังข้อมูลการเรียนรู้ของเครื่องจักร UCI [Asuncion & Newman, 2007] ชุดข้อมูลเหล่านี้มาจากหลายๆ ปัญหาจริง จำนวนคุณลักษณะ และ จำนวนข้อมูลของแต่ละชุดข้อมูลแสดงในตารางที่ 8

ตารางที่ 8 ชุดข้อมูลจากคลังข้อมูล UCI

ชุดข้อมูล	auto_mpg	cpu_performance	housing	servo
จำนวนคุณลักษณะ	7	6	13	4
จำนวนข้อมูล	392	209	506	167

แต่ละชุดข้อมูลจะถูกประเมินประสิทธิภาพโดยใช้การตรวจสอบไขว้ 5 พับ (5-fold cross-validation) เริ่มต้นจากการแบ่งข้อมูลฝึกออกเป็น 5 ส่วนเท่าๆ กัน แต่ละส่วนย่อยนี้จะถูกสอนและประเมิน 5 ครั้ง ในรอบที่ j (j=1, 2, 3, 4, 5) ทุกส่วนย่อยยกเว้นส่วนที่ j จะถูกใช้ในการสอน จากนั้นจึงคำนวณค่าเฉลี่ยของประสิทธิภาพในการทำนายค่า 5 ครั้งนี้ การแบ่งข้อมูลเพื่อทำการตรวจสอบไขว้ 5 พับแสดงไว้ในรูปที่ 4

สำหรับการวัดประสิทธิภาพของการทำนายค่าด้วยวิธีการที่นำเสนอ เราจะใช้ค่าสมบูรณ์ของเปอร์เซ็นต์ความผิดพลาดเฉลี่ยแบบสมมาตร (Symmetric Mean Absolute Percentage Error: SMAPE) [Hyndman & Koehler, 2006] ซึ่งค่าความผิดพลาดนี้คำนวณได้จาก

$$\text{SMAPE} = \frac{1}{l}\sum_{i=1}^{l}\left|\frac{y_i - \hat{y}_i}{(y_i + \hat{y}_i)/2}\right| \times 100 \qquad (38)$$

โดยที่ y_i เมื่อ $i = 1,...,l$ เป็นค่าเป้าหมายที่แท้จริงของข้อมูลฝึก, $\hat{y}_i$ เมื่อ $i = 1,...,l$ เป็นค่าที่ได้จากการทำนาย, และ l เป็นจำนวนข้อมูลฝึก

กลยุทธเชิงวิวัฒน์ถูกนำมาใช้เพื่อหาค่าของพารามิเตอร์ที่เหมาะสม ค่าของ τ ในกระบวนการเชิงวิวัฒน์ถูกกำหนดเป็น 1.0 ความกว้างของอาร์บีเอฟ (γ_i) ค่าถ่วงน้ำหนัก (a_i) ส่วนเบี่ยงเบนของการประมาณค่า (ε) และ พารามิเตอร์สามัญ (C) เป็นจำนวนจริงที่มีค่าอยู่ระหว่าง 0.0 และ 10.0 จำนวนอาร์บีเอฟกำหนดไว้เป็น 10 พารามิเตอร์เหล่านี้จะถูกค้นหาเป็นจำนวน 1500 รอบของอีเอส ผลการทดลองแสดงในตารางที่ 9

วิธีการที่นำเสนอ (การรวมเคอร์เนลอาร์บีเอฟหลายตัวแบบเสถียรและปรับตัวได้: *stable_obj*) ซึ่งใช้คุณสมบัติความเสถียรของเอสวีอาร์เป็นฟังก์ชันวัตถุประสงค์ในกระบวนการเชิงวิวัฒน์ ถูกนำมาเปรียบเทียบกับฟังก์ชันวัตถุประสงค์อื่นๆ ซึ่งได้แก่ ค่าเฉลี่ยของรากที่สองของความผิดพลาด (*mse_obj*) และ ค่าสมบูรณ์ของเปอร์เซ็นต์ความผิดพลาดเฉลี่ยแบบสมมาตร (*smape_obj*) นอกจากนั้นวิธีการที่นำเสนอยังถูกนำไป

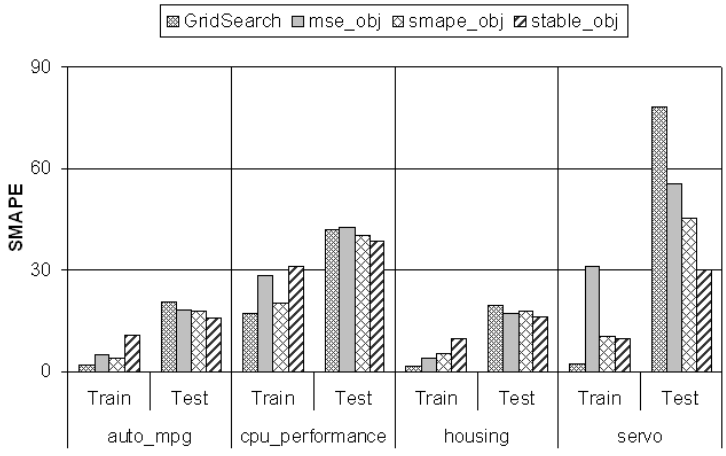
เปรียบเทียบกับการค้นหาแบบกริดโดยใช้เคอร์เนลอาร์บีเอฟเดี่ยวๆ ผลการทดลองแสดงให้เห็นถึงความสามารถของวิธีการที่นำเสนอ ซึ่งให้ค่าเฉลี่ยของความผิดพลาดน้อยกว่าฟังก์ชันวัตถุประสงค์อื่นๆ และน้อยกว่าการค้นหาแบบกริดในทุกชุดข้อมูล และค่าเฉลี่ยของความผิดพลาดบนทั้ง 4 ชุดข้อมูล ของวิธีการที่นำเสนอ ก็ยังน้อยที่สุดเมื่อเทียบกับวิธีการอื่นๆ

ตารางที่ 9 ค่าเฉลี่ยของความผิดพลาดบนการตรวจสอบไขว้ 5 พับ

ชุดข้อมูล	อาร์บีเอฟการค้นหาแบบกริด	การรวมเคอร์เนลอาร์บีเอฟหลายตัวแบบปรับค่าได้		
		mse_obj	*smape_obj*	*stable_obj*
auto_mpg	20.5728	18.1050	17.8098	**15.5734**
cpu_performance	41.7046	42.3626	40.4926	**38.7447**
Housing	19.4757	16.9928	18.0143	**15.9475**
Servo	78.0434	55.3948	45.3529	**29.8615**
ค่าเฉลี่ย	39.9491	33.2138	30.4174	**25.0317**

ในการทดลองนี้ การค้นหาแบบกริดถูกนำมาใช้สำหรับการหาพารามิเตอร์ของเอสวีอาร์ และ เคอร์เนลอาร์บีเอฟเดี่ยวๆ เพื่อหาค่าของ ε, C, และ γ การค้นหาด้วยวิธีนี้จะได้ค่าเฉลี่ยของความผิดพลาดในช่วงสอนต่ำ แต่วิธีการนี้ต้องใช้เวลานาน ซึ่งขึ้นอยู่กับขนาดของขั้นที่ใช้ในการค้นหาคำตอบ และ จำนวนพารามิเตอร์ที่ต้องการปรับค่า แต่ถึงแม้ว่าการรวมเคอร์เนลอาร์บีเอฟหลายตัวจะได้ฟังก์ชันเคอร์เนลที่มีความยืดหยุ่นต่อปัญหาที่เรากำลังสนใจมากขึ้น แต่ก็มีจำนวนพารามิเตอร์เพิ่มมากขึ้นด้วย ทำให้เซตของพารามิเตอร์ทั้งหมดที่เป็นไปได้เพิ่มมากขึ้น ซึ่งทำให้การค้นหาแบบกริดยิ่งใช้เวลานานขึ้นมาก ด้วยเหตุนี้การค้นหาแบบกริดจึงไม่เหมาะสมที่จะนำมาใช้กับการรวมเคอร์เนลอาร์บีเอฟหลายตัว

อย่างไรก็ตาม ด้วยความช่วยเหลือของอีเอส ทำให้วิธีการที่นำเสนอนี้สามารถหาพารามิเตอร์ที่เหมาะสมได้อย่างสะดวกยิ่งขึ้น วิธีการที่นำเสนอถูกนำไปเปรียบเทียบกับฟังก์ชันวัตถุประสงค์อื่นๆที่นำมาใช้กับการรวมเคอร์เนลอาร์บีเอฟหลายตัว เพื่อที่จะยืนยันว่าคุณสมบัติความเสถียรของเอ็สวีอาร์เหมาะสมที่จะนำมาใช้เป็นฟังก์ชันวัตถุประสงค์ แผนภูมิของค่าเฉลี่ยความผิดพลาดบนการตรวจสอบไขว้ 5 พับ เพื่อเปรียบเทียบระหว่างช่วงสอน และ ช่วงทดสอบแสดงในรูปที่ 9



รูปที่ 9 ค่าเฉลี่ยของความผิดพลาดบนข้อมูลฝึกและข้อมูลทดสอบ

จากแผนภูมินี้ แสดงค่าเฉลี่ยของความผิดพลาดของ *stable_obj* เปรียบเทียบกับค่าเฉลี่ยของความผิดพลาดของ *mse_obj*, *smape_obj*, และ การค้นหาแบบกริด พบว่า *stable_obj* ไม่ได้ให้ค่าเฉลี่ยของความผิดพลาดต่ำสุดในช่วงสอน แต่ค่าเฉลี่ยของความผิดพลาดในช่วงทดสอบของ *stable_obj* ต่ำกว่าวิธีการอื่นๆในทุกชุดข้อมูล นั่นหมายความว่า *stable_obj* ไม่ได้สอดคล้องกับข้อมูลฝึกจนเกินไป และ คุณสมบัติความเสถียรก็เป็นฟังก์ชันวัตถุประสงค์ที่ดีสำหรับกรณีของการใช้การรวมกันของหลายเคอร์เนลอาร์บีเอฟ ขณะที่การใช้ *mse_obj* และ *smape_obj* อาจทำให้เกิดปัญหาการสอดคล้องกับข้อมูลฝึกจนเกินไปได้ เนื่องจากเราพบว่าค่าเฉลี่ยของความผิดพลาดบนข้อมูลฝึกของ *mse_obj* และ *smape_obj* ต่ำกว่า *stable_obj* บนชุดข้อมูลฝึก auto_mpg, cpu_performance, และ housing แต่ค่าเฉลี่ยของความผิดพลาดบนข้อมูลทดสอบสูงกว่า *stable_obj* บนชุดข้อมูลฝึกเหล่านี้

2.6 สรุปการวิจัยพัฒนาฟังก์ชันเคอร์เนลสำหรับซัพพอร์ตเวกเตอร์รีเกรชชัน

การรวมกันแบบเชิงเส้นที่ไม่เป็นจำนวนลบของเคอร์เนลอาร์บีเอฟหลายตัวถูกนำเสนอสำหรับซัพพอร์ตเวกเตอร์แมชชีนบนปัญหาการหาค่าถดถอย ฟังก์ชันเคอร์เนลที่นำเสนอนี้สอดคล้องตามทฤษฎีบทของ Mercer จากนั้นกลยุทธเชิงวิวัฒน์ถูกนำมาใช้เพื่อหาค่าพารามิเตอร์ที่เหมาะสมของเอสวีอาร์ แม้ว่าการค้นหาแบบกริดจะเป็นวิธีการที่ดีในการหาค่าของพารามิเตอร์ แต่การค้นหาแบบนี้จะใช้เวลานานกว่า เมื่อเราใช้ฟังก์ชันเคอร์เนลที่เกิดจากการรวมอาร์บีเอฟหลายตัว จะทำให้มีพารามิเตอร์เพิ่มมากขึ้น และเซตของพารามิเตอร์ที่เป็นไปได้ก็จะใหญ่ขึ้นด้วย ดังนั้นการค้นหาแบบกริดจึงไม่ใช่วิธีการที่เหมาะสม สำหรับกรณีที่เลือกใช้การรวมเคอร์เนลอาร์บีเอฟหลายตัว ด้วยเหตุนี้ อีเอสจึงเป็นวิธีการที่เหมาะสมมากกว่าสำหรับการค้นหาค่าของพารามิเตอร์หลายตัวพร้อมกัน

แต่การใช้ฟังก์ชันเคอร์เนลที่มีความยืดหยุ่นมากขึ้นนี้ อาจเป็นสาเหตุทำให้เกิดการสอดคล้องกับข้อมูลฝึกมากจนเกินไปได้ เพื่อที่จะหลีกเลี่ยงปัญหานี้ คุณสมบัติความเสถียรของเอสวีอาร์ จึงถูกนำมาใช้เป็นฟังก์ชันวัตถุประสงค์ในกระบวนการเชิงวิวัฒน์ ผลการทดลองแสดงให้เห็นถึงความสามารถของวิธีการที่นำเสนอ โดยดูจากค่าเฉลี่ยของความผิดพลาดทั้งในช่วงสอนและช่วงทดสอบ แม้ว่าค่าเฉลี่ยของความผิดพลาดในช่วงสอนของเอสวีอาร์ที่ใช้การรวมเคอร์เนลอาร์บีเอฟหลายตัวแบบเสถียรและปรับตัวได้ จะไม่ได้ต่ำกว่าการใช้ฟังก์ชันวัตถุประสงค์ตัวอื่นและการค้นหาแบบกริด แต่การรวมเคอร์เนลอาร์บีเอฟหลายตัวแบบเสถียรและปรับตัวได้นี้ให้ค่าเฉลี่ยของความผิดพลาดในช่วงทดสอบที่ต่ำที่สุดเมื่อเทียบกับวิธีการอื่นๆ ในการทดลอง ซึ่งนั่นหมายความว่า วิธีการที่นำเสนอนี้ไม่ได้สอดคล้องกับข้อมูลฝึกจนเกินไปและให้ผลการทำนายค่าที่ดี ด้วยเหตุนี้วิธีการที่นำเสนอจึงมีความเหมาะสมอย่างยิ่งบนปัญหาที่มีความยุ่งยากซับซ้อน และเราไม่มีความรู้ก่อนหน้าเกี่ยวกับพารามิเตอร์ต่างๆ เลย ยิ่งไปกว่านั้นวิธีการดังกล่าวยังสามารถปรับใช้กับฟังก์ชันเคอร์เนลอื่นๆ เช่น เคอร์เนลพหุนามหรือเคอร์เนลอนุกรมฟูริเย่ต์ได้อีกด้วย

3. การวิจัยกลุ่มก้อนตัวจำแนกที่สามารถลดไบแอสและแวเรียนซ์

การประมาณความผิดพลาดของอัลกอริทึมการเรียนรู้นั้นเป็นประเด็นสำคัญในวงการการเรียนรู้ของเครื่องมาช้านานในงานวิจัยของ Geman et al. [1992] ได้นำเสนอวิธีการที่เรียกว่าการแยกองค์ประกอบไบแอส-แวเรียนซ์ (Bias-Variance Analysis) ของความผิดพลาดในปัญหาการถดถอย ซึ่งวิธีการนี้ไม่เพียงแต่ให้การประมาณค่าความผิดพลาดเท่านั้น แต่ยังให้สิ่งที่เรียกว่า*ไบแอส* และ *แวเรียนซ์* อีกด้วย โดยสามารถเขียนคร่าวๆ ได้ว่า

ความผิดพลาดทั่วไป = นอยส์ + ไบแอส + แวเรียนซ์

โดยไบแอสคือสิ่งชี้วัดสมรรถภาพของอัลกอริทึม ขณะที่แวเรียนซ์คือสิ่งชี้วัดเสถียรภาพของอัลกอริทึม องค์ประกอบทั้งสองนี้ไม่เพียงแต่จะบอกความผิดพลาดของอัลกอริทึมการเรียนรู้เท่านั้น หากแต่ยังแฝงนัยยะในการปรับปรุงอัลกอริทึมการเรียนรู้เอาไว้ เช่น Breiman [1996] ได้คิดค้นอัลกอริทึมการเรียนรู้แบบกลุ่มก้อนตัวจำแนกชื่อว่า Bagging ซึ่งถูกออกแบบมาให้กำจัดแวเรียนซ์อย่างสิ้นเชิง และนำไปสู่การเรียนรู้ที่ให้ความผิดพลาดลดลง ด้วยเหตุนี้เอง การวิเคราะห์ไบแอส-แวเรียนซ์ จึงเป็นหัวข้อในการวิจัยที่มีผู้สนใจกันเป็นอันมาก [Valentini, 2005; Suen et al., 2005]

จากความสำเร็จเช่นนี้ ทำให้มีความพยายามอย่างยิ่งในการขยายขอบเขตการวิเคราะห์ไบแอส-แวเรียนซ์ไปสู่ปัญหาการจำแนก [Kong & Dietterich, 1995; Kohavi & Wolpert, 1996; Tibshirani, 1996; Breiman, 1998;Domingos, 2000; James, 2003] อย่างไรก็ตาม James [2003] ได้แสดงให้เห็นว่าการขยายขอบเขตการวิเคราะห์อย่างตรงไปตรงมานั้นเป็นไปไม่ได้กับฟังก์ชันความสูญเสียทั่วไป (general loss function) กล่าวคือความหมายของไบแอสและแวเรียนซ์แบบดั้งเดิมนั้นไปด้วยกันไม่ได้กับคุณสมบัติการบวก โดย James เลือกที่จะคงค่าไบแอสและแวเรียนซ์ที่มีความหมายดั้งเดิมไว้ และนิยามอีกสองค่าที่สอดคล้องกับไบแอสและแวเรียนซ์ขึ้นมาเพื่อให้ได้คุณสมบัติการบวก

หัวข้อนี้จะมีการนำเสนอกรอบการวิเคราะห์ใหม่โดยใช้ชื่อว่าการวิเคราะห์โซน ซึ่งได้ถูกออกแบบขึ้นมาเพื่อให้ภาพอันกระจ่างของอัลกอริทึมการเรียนรู้สำหรับปัญหาการจำแนก ในกรอบการวิเคราะห์นี้ ความผิดพลาดในการเรียนรู้ไม่เพียงแต่จะถูกแบ่งเป็นสองส่วนคือไบแอสและแวเรียนซ์เท่านั้น แต่จะถูกแบ่งเป็น K ส่วน โดย K แสดงถึงจำนวนของโซน ในสถานการณ์ที่ไม่มีนอยส์ กรอบการวิเคราะห์โซนนี้สามารถถือได้ว่าเป็นรูปทั่วไปในกรอบการวิเคราะห์ของ Domingos [2000] ซึ่งเมื่อ K เท่ากับ 2 โซนที่ได้ของเราก็จะเทียบเท่ากับโซนไบแอสและโซนไม่ไบแอส ยิ่งไปกว่านั้น การวิเคราะห์แบบโซนยังสามารถแสดงให้เห็นรูปแบบต่างๆ ที่ไม่เคยปรากฏมาก่อนเลยในการวิเคราะห์อัลกอริทึมการเรียนรู้แบบกลุ่มก้อนตัวจำแนกที่แล้วๆ มา

3.1 การวิเคราะห์โซนสำหรับปัญหาการจำแนก

ในหัวข้อนี้จะกล่าวถึงการวิเคราะห์ไบแอส-แวเรียนซ์ อย่างคร่าวๆ จากนั้นจึงเป็นที่มาและนิยามของการวิเคราะห์โซน

3.1.1 การแยกองค์ประกอบไบแอส-แวเรียนซ์

กรอบการวิเคราะห์ที่แสดงในที่นี้คือกรอบการวิเคราะห์ของ Domingos [2000] โดยเราจะสนใจปัญหาการจำแนกเท่านั้น นั่นคือปัญหาที่มีฟังก์ชันความสูญเสียแบบ 0/1 (0/1 loss function)

ให้ D เป็นชุดข้อมูลฝึกที่มี N ตัวอย่าง ซึ่งมีลักษณะเหมือนกันและเป็นอิสระต่อกัน (i.i.d. data) จากการแจกแจงความน่าจะเป็นที่ไม่ทราบค่าแต่ว่าคงที่ โดยแต่ละตัวอย่าง $(\mathbf{x}_j, t_j)$ เป็นคู่ของตัวอย่าง $\mathbf{x} \in X$ และ เป้าหมาย $t \in C$ โดยที่ X คือสเปซของตัวอย่าง และ C คือสเปซของประเภทที่เป็นไปได้ กำหนดให้ A เป็นอัลกอริทึมการเรียนรู้ และ $A(D) = f_D$ คือตัวจำแนกประเภท ซึ่งเป็นผลลัพธ์ที่ได้จากการเรียนรู้โดยใช้อัลกอริทึมการเรียนรู้ A รับชุดข้อมูลฝึก D เพื่อความง่าย เราจะกำหนดให้ f_D เป็น*ฟังก์ชัน*ของ $\mathbf{x}$ และจะทำนายประเภท y โดยที่ $y = f_D(\mathbf{x})$

ให้ $L(t, y)$ เป็นฟังก์ชันความสูญเสียแบบ 0/1 นั่นคือ $L(t, y) = 0$ เมื่อ $t = y$ ส่วนในกรณีอื่น $L(t, y) = 1$ เรานิยามค่าคาดหวังความสูญเสีย (expected loss) หรือค่าความผิดพลาดทั่วไป EL ของอัลกอริทึมการเรียนรู้ A ต่อตัวอย่าง $\mathbf{x}$ ใดๆ ดังนี้

$$EL(A, \mathbf{x}) = E_{t,D}[L(t, f_D(\mathbf{x}))] \tag{39}$$

เพื่อความกระชับ เราจะเขียนแทน $EL(A,\mathbf{x})$ ด้วย $EL(\mathbf{x})$ ซึ่งเป้าหมายของการวิเคราะห์ไบแอส-แวเรียนซ์ ก็คือการแยกองค์ประกอบของค่าคาดหวังความสูญเสียนี้ออกเป็นไบแอสและแวเรียนซ์

นิยาม*การทำนายที่เหมาะสมสุด* $y_*(\mathbf{x})$ ให้เป็นประเภทที่ปรากฏมากสุดจากการแจกแจงของ $\mathbf{x}$ หรือเราสามารถนิยามได้ว่า คือค่า y ที่ทำให้ $E_t[L(t,y)\,|\,\mathbf{x}]$ มีค่าต่ำที่สุด นั่นคือ

$$y_*(x) = \arg\min_y E_t[L(t,y)\,|\,\mathbf{x}] \tag{40}$$

*ตัวจำแนกเบส์*ที่รู้จักกันดีถูกนิยามให้เป็นตัวจำแนกที่ดีที่สุด โดยสามารถทำนาย y_* สำหรับ $\mathbf{x}$ ได้เสมอ และให้ความผิดพลาดน้อยสุดที่เป็นไปได้ คือ $N(\mathbf{x})$ หรือที่เรียกว่า*นอยส์*ของ $\mathbf{x}$ นั่นเอง

$$N(\mathbf{x}) = E_t[L(t,y_*)\,|\,\mathbf{x}] \tag{41}$$

นอกจากนี้เรายังนิยาม *ผลการทำนายหลัก* $y_m(\mathbf{x})$ หรือแนวโน้มในการทำนายของอัลกอริทึมต่อตัวอย่าง $\mathbf{x}$ ดังนี้

$$y_m(\mathbf{x}) = \arg\min_{y'} E_D[L(f_D(\mathbf{x}),y')] \tag{42}$$

ไบแอส $B(\mathbf{x})$ ของอัลกอริทึมการเรียนรู้ต่อตัวอย่าง $\mathbf{x}$ สามารถนิยามได้ว่า $B(\mathbf{x})=1$ ถ้า $y_m \neq y_*$ และ $B(\mathbf{x})=0$ ถ้า $y_m = y_*$ หรือ

$$B(\mathbf{x}) = L(y_m, y_*) \tag{43}$$

ค่านี้จะเป็นตัวชี้ว่าการทำนายหลักถูกต้องหรือไม่ เราจะเรียก $\mathbf{x}$ ที่มีค่าไบแอสเท่ากับ 0 และ 1 ว่าตัวอย่างไม่ไบแอส และตัวอย่างไบแอส (ของอัลกอริทึมการเรียนรู้) ตามลำดับ

แวเรียนซ์ $V(\mathbf{x})$ คือค่าคาดหวังความสูญเสียของแต่ละการทำนายเมื่อเทียบกับผลการทำนายหลัก นั่นคือ

$$V(\mathbf{x}) = E_D[L(f_D(\mathbf{x}),y_m)] \tag{44}$$

เมื่อเปรียบเทียบกับนิยามนอยส์ ไบแอส และแวเรียนซ์ ของปัญหาถดถอย [Geman et al., 1992] กลุ่มนิยามข้างต้นไม่สามารถนำมาบวกกันโดยตรงแล้วให้ค่าความผิดพลาดได้ ความจริงแล้วในปัญหาการจำแนก แวเรียนซ์อาจจะช่วยให้เกิดความถูกต้องยิ่งขึ้นในตัวอย่างไบแอส นั่นเป็นเพราะว่าในตัวอย่างไบแอส ผลการทำนายหลักนั้นผิด การมีแวเรียนซ์จะช่วยกลับผลการทำนายให้กลายเป็นถูกได้ ส่วนตัวอย่างที่ไม่ไบแอสนั้น แวเรียนซ์จะส่งผลตรงกันข้าม

ในปัญหาการจำแนกโดยทั่วไป Domingos แสดงให้เห็นว่าค่าคาดหวังความสูญเสียของอัลกอริทึมต่อตัวอย่าง $\mathbf{x}$ สามารถแยกองค์ประกอบได้ดังนี้

$$EL(\mathbf{x}) = c_1(\mathbf{x})N(\mathbf{x}) + B(\mathbf{x}) + c_2(\mathbf{x})V(\mathbf{x}) \tag{45}$$

ในกรณีที่มีสองประเภท $c_1(\mathbf{x}) = (1-2B(\mathbf{x}))(1-2V(\mathbf{x}))$ และ $c_2(\mathbf{x}) = 1-2B(\mathbf{x})$ สังเกตว่าผลของ $V(\mathbf{x})$ ขึ้นกับ $c_2(\mathbf{x})$ ซึ่งในที่นี้ $c_2(\mathbf{x})$ เท่ากับ -1 ในตัวอย่างไบแอส และ +1 ในตัวอย่างไม่ไบแอส เพื่อที่จะแบ่งแยกความแตกต่างของแวเรียนซ์ที่ให้ผลต่างกันนี้ Domingos จึงให้แวเรียนซ์ไม่ไบแอส $V_u(\mathbf{x})$ เป็นแวเรียนซ์ที่เกิดขึ้นเมื่อ $B(\mathbf{x})=0$ และ แวเรียนซ์ไบแอส $V_b(\mathbf{x})$ เป็นแวเรียนซ์ที่เกิดขึ้นเมื่อ $B(\mathbf{x})=1$ จากมุมมองนี้เอง ทำให้เราสามารถเขียนผลของแวเรียนซ์ได้ว่า

$$c_2(\mathbf{x})V(\mathbf{x}) = V_u(\mathbf{x}) - V_b(\mathbf{x}) \tag{46}$$

จากทั้งหมดที่กล่าวมา ในกรณีที่ไร้นอยส์และมีสองประเภท เราสามารถสรุปได้ว่า

$$E_{\mathbf{x}}[EL(\mathbf{x})] = E_{\mathbf{x}}[B(\mathbf{x})] + E_{\mathbf{x}}[V_u(\mathbf{x})] - E_{\mathbf{x}}[V_b(\mathbf{x})] \quad (47)$$

สมการนี้เองที่เป็นหัวใจในงานของ [Valentini, 2005; Domingos, 2000; Valentini & Dietterich, 2003; Valentini & Dietterich, 2004]

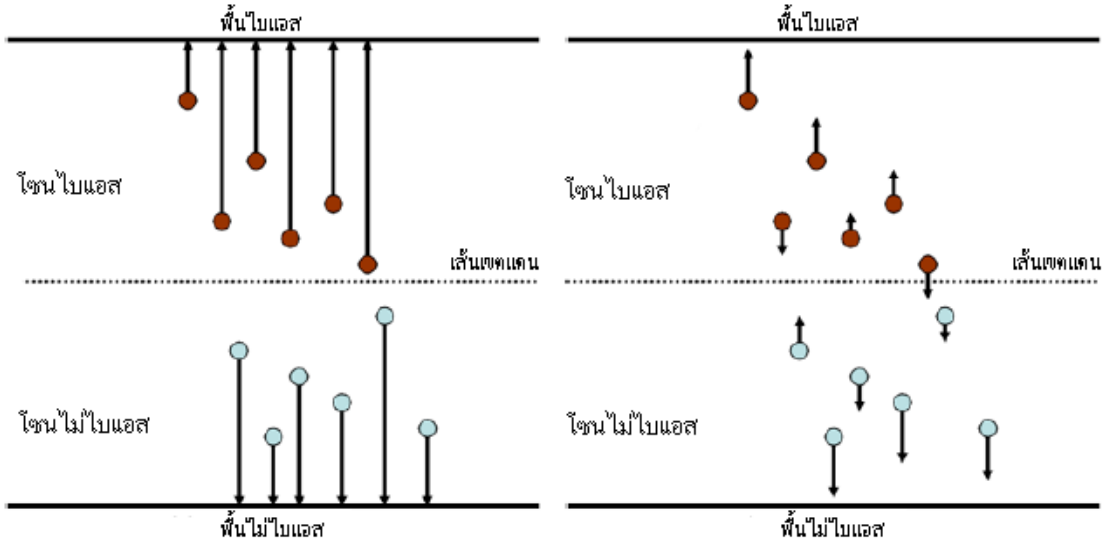
3.1.2 ที่มาของการวิเคราะห์โซน

การวิเคราะห์โซนได้รับแรงบันดาลใจมาจากการวิเคราะห์ไบแอส-แวเรียนซ์ ของ Bagging [Breiman, 1996] โดย Bagging นับว่าเป็นหนึ่งในเทคนิคกลุ่มก้อนตัวจำแนกที่โด่งดัง มีประสิทธิภาพ และมีความคงทน อย่างไรก็ตามก็ยังไม่สามารถหาเหตุผลได้ว่าทำไม Bagging จึงทำงานได้ดีมาก โดยเริ่มแรกนั้นเชื่อกันว่าเป็นเพราะ Bagging สามารถกำจัดแวเรียนซ์ให้หมดไป [Valentini & Dietterich, 2003] นั่นคือ หลังจากใช้ Bagging แล้ว ทั้ง $V_u(\mathbf{x})$ และ $V_b(\mathbf{x})$ จะกลายเป็น 0 ไปทั้งหมด สำหรับทุกๆ $\mathbf{x}$ เพราะฉะนั้นจะได้ว่า $E_{\mathbf{x}}[EL(\mathbf{x})] = E_{\mathbf{x}}[B(\mathbf{x})]$ หรือค่าคาดหวังความสูญเสียจะลดลงมาเท่ากับค่าคาดหวังไบแอส

การเคลื่อนไหวใน*อุดมคติ*ของค่าความผิดพลาดในแต่ละตัวอย่าง $\mathbf{x}$ ก่อนและหลัง Bagging สามารถแสดงได้ดังรูปที่ 10 (ซ้าย) ซึ่งเราสามารถเขียนสมการ เพื่ออธิบายการเคลื่อนไหวได้ดังนี้

$$E_{\mathbf{x}}[EL(\mathbf{x})] = E_{\mathbf{x}}[B(\mathbf{x}) - V_b(\mathbf{x})] + E_{\mathbf{x}}[V_u(\mathbf{x})] \quad (48)$$

นั่นคือค่าคาดหวังความสูญเสียถูกแยกองค์ประกอบออกเป็นความสูญเสียจากโซนไบแอส และความสูญเสียจากโซนไม่ไบแอส ซึ่งหลังจากทำ Bagging แล้ว ความสูญเสียที่เป็นผลมาจากโซนไม่ไบแอสจะลดลงจนเหลือศูนย์ อย่างไรก็ตาม การเคลื่อนไหวในอุดมคตินี้ไม่สามารถอธิบายการทำงานของ Bagging ได้อย่างสมบูรณ์แบบ โดย Buja and Stuetzle [2006] ยกตัวอย่างที่แสดงให้เห็นว่า Bagging เพิ่มค่าไบแอสและแวเรียนซ์กำลังสอง Grandvalet [2004] ให้ข้อสรุปอย่างดีสำหรับคำอธิบายเชิงทฤษฎีของ Bagging ในแนวที่ต่างออกไป อย่างไรก็ตามการทดลองเชิงปฏิบัติก็สนับสนุนว่าความคิดของ Breiman มีความถูกต้องอยู่บ้าง นั่นคือ Bagging มักจะลดแวเรียนซ์ของตัวอย่างได้จริง (แต่ไม่เท่ากับศูนย์) ดังรูปที่ 10 (ขวา)



รูปที่ 10 (ซ้าย) การเคลื่อนที่ในอุดมคติเมื่อทำ Bagging (ขวา) การเคลื่อนที่ในความเป็นจริงเมื่อทำ Bagging

ในการพัฒนาการวิเคราะห์โซน เรามุ่งเน้นต่อคำถามที่ว่า ถ้าการเคลื่อนที่แบบอุดมคติตามรูปที่ 10 (ซ้าย) ไม่เกิดขึ้น เราจะอธิบายลักษณะการเคลื่อนที่จริงๆ อย่างไร ในหัวข้อ 3.3.2 เราจะแสดงว่า การเคลื่อนที่จริงๆ ของตัวอย่างต่างๆ หลังจากใช้ Bagging จะเหมือนกับรูปที่ 10 (ขวา) นั่นคือ ตัวอย่างไม่ไบแอสที่อยู่ใกล้พื้นจะ

เข้าไปใกล้พื้นมากกว่าตัวอย่างไม่ไบแอสที่อยู่ไกลออกไป ในการวิเคราะห์โซน นอกจากการใช้ Bagging แล้ว ยังสามารถนำไปใช้จับการเคลื่อนไหวของการเปลี่ยนบริบทอื่นๆ ได้ ตัวอย่างการเปลี่ยนบริบทก็เช่น การใช้การเรียนรู้แบบกลุ่มก้อนตัวจำแนกต่างๆ

3.1.3 นิยามของโซนและการวิเคราะห์

เมื่อกำหนดพารามิเตอร์ K มา เราสามารถนิยามโซนทั้งหมด K โซน คือ $Z_1, Z_2, ..., Z_K$ ได้ดังนี้

$$Z_{\mathrm{k}} = \left\{ x \mid EL(x) \in \left[\frac{k-1}{K}, \frac{k}{K} \right) \right\} \tag{49}$$

ยกตัวอย่างเช่น เมื่อ $K = 10$ ตัวอย่าง $\mathbf{x}$ ที่ $EL(\mathbf{x}) \in [0, 0.1)$ จะถูกกำหนดให้อยู่ในโซนแรก ส่วนตัวอย่าง $\mathbf{x}$ ที่ $EL(\mathbf{x}) \in [0.1, 0.2)$ จะถูกกำหนดให้อยู่ในโซนที่สอง เป็นอย่างนี้ต่อไปเรื่อยๆ โดยที่ตัวอย่างเดียวจะอยู่ได้เพียงโซนเดียว

ลองพิจารณากรณีอย่างง่ายที่ $K = 2$ คือมีแค่โซนไบแอสและโซนไม่ไบแอส ค่าคาดหวังความสูญเสียในสมการ ... สามารถเขียนได้ว่า

$$E_{\mathbf{x}}[EL(\mathbf{x})] = \sum_{\mathbf{x} \in Z_1} P(\mathbf{x}) EL(\mathbf{x}) + \sum_{\mathbf{x} \in Z_2} P(\mathbf{x}) EL(\mathbf{x}) \tag{50}$$

เมื่อ $P(\mathbf{x})$ แทนความน่าจะเป็นในการได้ตัวอย่าง x ในกรณีทั่วไปของ K สามารถเขียนได้ว่า

$$E_{\mathbf{x}}[EL(\mathbf{x})] = \sum_{k=1}^{K} \left(\sum_{\mathbf{x} \in Z_k} P(\mathbf{x}) EL(\mathbf{x}) \right) \tag{51}$$

หรือ

$$E_{\mathbf{x}}[EL(\mathbf{x})] = \sum_{k=1}^{K} P(Z_k) EL(Z_k) \tag{52}$$

เมื่อ $P(Z_k) = \sum_{\mathbf{x} \in Z_k} P(\mathbf{x})$ แทนความน่าจะเป็นที่ตัวอย่างที่ถูกสุ่มขึ้นมาจะอยู่ในโซนที่ K และกำหนดให้ $EL(Z_k) = \sum_{\mathbf{x} \in Z_k} P(\mathbf{x}) EL(\mathbf{x}) / P(Z_k)$ แทนค่าเฉลี่ยค่าคาดหวังความสูญเสียของโซนที่ K ซึ่งในบทความนี้ค่า K จะเท่ากับ 10 เสมอไป ซึ่งเราคิดว่านี้เป็นค่าที่ให้ข้อมูลเพียงพอแก่ความต้องการ

สังเกตว่า $P(Z_k)$ ในที่นี้ไม่เหมือนกับ $P(\mathbf{x})$ ตรงที่ $P(Z_k)$ ขึ้นกับอัลกอริทึมการเรียนรู้ด้วย ในทางปฏิบัติแล้ว เราจะกำหนดให้ $P(\mathbf{x})$ มีการแจกแจงแบบยูนิฟอร์มตามงานของ Valentini & Dietterich [Valentini & Dietterich, 2003; Valentini & Dietterich, 2004] นั่นคือ $P(\mathbf{x}) = 1/N$ เมื่อ N เป็นจำนวนตัวอย่างในชุดข้อมูล ด้วยเหตุนี้จึงได้ว่า $P(Z_k) = N_k / N$ และ

$$EL(Z_k) = \sum_{\mathbf{x} \in Z_k} EL(\mathbf{x}) / N_k \tag{53}$$

เมื่อ N_k เป็นจำนวนตัวอย่างในโซนที่ K

เมื่อบริบทในการเรียนรู้เปลี่ยนแปลง ก็จะมีการเปลี่ยนแปลงในค่าคาดหวังความสูญเสียด้วย นั่นคือ

$$E_{\mathbf{x}}[\Delta EL(\mathbf{x})] = \sum_{k=1}^{K} P(Z_k) \Delta EL(Z_k) \tag{54}$$

เมื่อโซน $\{Z_k\}$ นิยามไว้ก่อนที่จะเปลี่ยนบริบท ในการวิเคราะห์โซน เราจะสนใจในแต่ละ $\Delta EL(Z_k)$ นั่นคือค่าเฉลี่ยในการเคลื่อนที่ของทุกตัวอย่างในโซน ซึ่งกรรมวิธีที่จะประมาณค่านี้แสดงให้เห็นได้ดังรูปที่ 11

กระบวนการตรวจจับการเคลื่อนที่

Inputs :

(1) $\{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N\}$ (2) $EL_1[\mathbf{x}_1 \ldots \mathbf{x}_N]$ (3) $EL_2[\mathbf{x}_1 \ldots \mathbf{x}_N]$

For $i = 1$ to N

$$Insert\left(Z\left[\lceil EL_1(\mathbf{x}_i) * 10 \rceil\right], \mathbf{x}_i\right)$$

For $k = 1$ to 10

{

$$EZ_1(k) = \frac{\sum_{i|\mathbf{x}_i \in Z_k} EL_1(\mathbf{x}_i)}{\sum_{i|\mathbf{x}_i \in Z_k} 1}$$

$$EZ_2(k) = \frac{\sum_{i|\mathbf{x}_i \in Z_k} EL_2(\mathbf{x}_i)}{\sum_{i|\mathbf{x}_i \in Z_k} 1}$$

$$\Delta EZ(k) = EZ_2(k) - EZ_1(k)$$

}

Outputs :

(1) $EZ_1[1 \ldots 10]$ (2) $\Delta EZ[1 \ldots 10]$

รูปที่ 11 กระบวนการตรวจจับการเคลื่อนที่ ซึ่งจะคำนวณค่า $\Delta EL(Z_k)$ เพื่อความง่ายในที่นี้ได้ใช้สัญลักษณ์ $EZ(k)$ แทน $EL(Z_k)$

งานดั้งเดิมของ Valentini & Dietterich [Valentini & Dietterich, 2003; Valentini & Dietterich, 2004] และ Domingos [2000] อิงสมการในหัวข้อ 3.1.1 เป็นสำคัญ แต่ว่าสมการนั้นไม่รองรับกรณีที่มีนอยส์หรือมีประเภทมากเกินกว่าสอง ตรงกันข้ามกับการวิเคราะห์โซนที่อิงสมการข้างต้น ซึ่งง่ายกว่าและสามารถใช้งานได้ทั่วไป ต้องย้ำอีกครั้งหนึ่งว่าการไม่ใส่ใจผลกระทบจากนอยส์จะทำให้ข้อมูลไบแอสและแวเรียนซ์ที่ได้มีความไขว้เขว ตามที่ James [2003] กล่าวไว้ ข้อดีอีกอย่างของการวิเคราะห์โซนก็คือว่า ในกรณีที่ไร้นอยส์ ข้อมูลไบแอสและแวเรียนซ์ก็ยังคงอยู่ในกรอบของการวิเคราะห์โซน ยกตัวอย่างเช่นในกรณีที่ $K = 10$ ห้าโซนแรกจะหมายถึงเขตไม่ไบแอส และห้าโซนหลังจะหมายถึงเขตไบแอส

3.1.4 การประมาณค่าคาดหวังความสูญเสียของตัวอย่าง

เราจะทำตามอย่างงานที่แล้วๆ มา ค่าคาดหวังความสูญเสียของแต่ละตัวอย่าง $\mathbf{x}$ จะประมาณโดยกรรมวิธีเอาต์ออฟแบก (out-of-bag) [Valentini & Dietterich, 2004] โดยในชุดข้อมูล D กรรมวิธีเอาต์ออฟแบกจะสร้างชุดข้อมูลย่อยจำนวน B แล้วกรรมวิธีนี้ก็จะวนซ้ำ B รอบ ในรอบที่ i ชุดข้อมูลย่อย B จะถูกนำมาใช้สร้าง f_{D_i} และชุดข้อมูลที่เหลือ $D - D_i$ จะถูกนำมาใช้สำหรับทดสอบ ค่าประมาณความผิดพลาดของตัวอย่าง $\mathbf{x}$ จะเป็นค่าเฉลี่ยใน B รอบ ทั้งหมดนี้สามารถสรุปเป็นสมการได้ว่า

$$EL(\mathbf{x}) = \frac{\sum_{i|\mathbf{x} \in D - D_i} L(t, f_{D_i}(\mathbf{x}))}{\sum_{i|\mathbf{x} \in D - D_i} 1} \quad (55)$$

การทดลองทั้งหมดในที่นี้ใช้จำนวนรอบเท่ากับ 200 ($B = 200$)

3.2 การกำหนดการทดลอง

ในการทดลอง เราได้เลือก 8 ชุดข้อมูลที่แตกต่างกัน โดยเป็น 6 ชุดข้อมูล UCI (MUSK, SPAM, LETTERRB, LETTERMN, GERMAN และ SPICE2) และ 2 ชุดข้อมูลเทียม (P2 และ DNF) นอกจากนี้เราเติมนอยส์เทียม 10% เข้าไปอีกในแต่ละชุดข้อมูล รวมได้ 16 ชุดข้อมูลทั้งหมด โดยนอยส์จะใส่เข้าไปในชุดข้อมูลฝึกในแต่ละรอบของกรรมวิธีเอาต์ออฟแบกเท่านั้น กล่าวคือ สำหรับแต่ละชุดข้อมูลฝึก D_i จากหัวข้อ 3.1.4 เราสุ่มเลือก 10% ของตัวอย่างทั้งหมดมาเปลี่ยนประเภท ส่วนตัวอย่างชุดทดสอบ $D - D_i$ ที่เหลือไม่มีการเปลี่ยนแปลง

ด้วยเหตุผลใดไม่ทราบแน่ เทคนิคกลุ่มก้อนตัวจำแนกโดยทั่วไปจะคงทนกว่าการเรียนรู้แบบเดี่ยว ๆ เพราะฉะนั้นจึงเป็นที่น่าสนใจในการวิเคราะห์คุณสมบัติความคงทนของเทคนิคกลุ่มก้อนตัวจำแนก สังเกตว่าในการทดลองเกี่ยวกับเทคนิคกลุ่มก้อนตัวจำแนกที่ผ่านๆ มา [Breiman, 1998; Domingos, 2000; Grandvalet, 2004; Bauer & Kohavi, 1999; Webb, 2000; Kuncheva & Whitaker, 2003] ก็ไม่ได้วิเคราะห์เรื่องของความคงทนมากนัก

ชุดข้อมูล LETTERRB และ LETTERMN นั้น เป็นชุดข้อมูลเวอร์ชันสองประเภท R กับ B และ M กับ N ของชุดข้อมูล LETTER ส่วนชุดข้อมูล SPLICE-2 เราใช้แค่สองประเภทของชุดข้อมูล SPLICE เท่านั้น คือ Intron to Exon กับ Exon to Intron ส่วนชุดข้อมูล DNF มีคุณสมบัติแบบบูลีน 20 ตัว (a1-a20) แล้วก็คงเดาได้จากชื่อว่าสมมติฐานที่แท้จริงของชุดข้อมูลนี้อยู่ในรูป Disjunctive Normal Form คือ

$$\text{if } (a1 \,\&\, a2 \,\&\, a3 \,\&\, a4 \,\&\, a5) \rightarrow \text{class } +$$
$$\text{if } (a6 \,\&\, a7 \,\&\, a8 \,\&\, a9 \,\&\, a10) \rightarrow \text{class } +$$

นอกเหนือจากนี้ก็จะเป็นประเภทลบ ส่วนชุดข้อมูล P2 เป็นชุดข้อมูลมาตรฐานโดย Valentini and Dietterich [Valentini & Dietterich, 2003; Valentini & Dietterich, 2004] ในชุดข้อมูลเทียมทั้งสอง เราสร้างทั้งหมด 10000 ตัวอย่าง แล้วเราก็ให้จำนวนตัวอย่างของชุดข้อมูลฝึกสอนในแต่ละรอบของกรรมวิธีเอาต์ออฟแบกเท่ากับ 200 ในชุดข้อมูล P2, DNF, MUSK และ SPAM ด้วยแนวคิดเดียวกับ Bauer & Kohavi [1999] และ Valentini & Dietterich [2003] การกำหนดเช่นนี้ทำให้ตัวอย่างชุดทดสอบมีขนาดใหญ่กว่าตัวอย่างชุดข้อมูลฝึกอย่างน้อย 20 เท่า และทำให้ได้ค่าเฉลี่ยความผิดพลาดที่น่าเชื่อถือ สำหรับชุดข้อมูล LETTERRB, LETTERMN, SPLICE-2 และ GERMAN นั้น เราใช้เพียง 100 ตัวอย่างสำหรับชุดข้อมูลฝึก เนื่องจากขนาดของชุดข้อมูลมีน้อย ในทุกการทดลอง เราใช้ C4.5 เป็นตัวเรียนรู้แบบเดี่ยวๆ และด้วยวิธีการเดียวกับ Bauer and Kohavi [1999] เราจะใช้ตัวเรียนรู้แบบเดี่ยวๆ 25 ตัวมาประกอบกันเป็นกลุ่มก้อนตัวจำแนก

3.3 การทดลอง

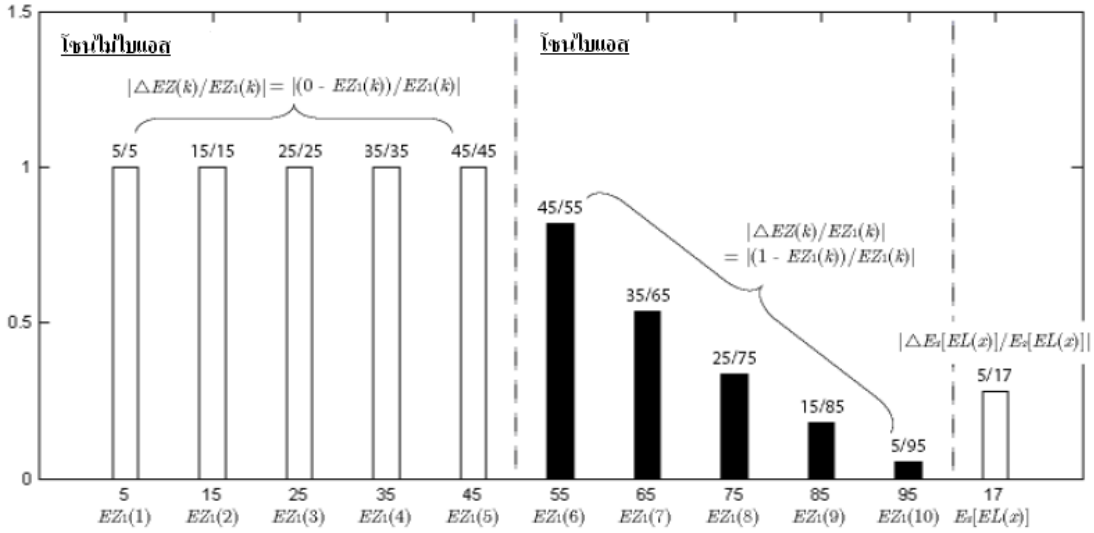
หัวข้อนี้จะเป็นการนำการวิเคราะห์โซนไปใช้กับการเปลี่ยนบริบทในการเรียนรู้ โดยทดสอบกับ Bagging

3.3.1 ตัวอย่างการวิเคราะห์โซน

เพื่อความแม่นยำในการรายงานผลการทดลอง การวิเคราะห์โซนจะให้ภาพดังรูปที่ 12 แทนที่จะเป็นดังรูปที่ 10 โดยรูปที่ 12 นี้จะเป็นกราฟแสดงกรณีที่ Bagging ประพฤติตัวตามอุดมคติ โดยสิบแท่งแรกในภาพจะหมายถึงความเปลี่ยนแปลงในสิบโซน โดยแต่ละแท่งจะเป็นขนาดของความเปลี่ยนแปลงสัมพัทธ์ นั่นคือ $\left|\Delta EZ(k) / EZ_1(k)\right|$ แท่งสีดำหมายถึงความผิดพลาดที่เพิ่มขึ้น (การเคลื่อนที่ไปยังพื้นไบแอส) ส่วนแท่งสีขาวหมายถึงความผิดพลาดที่ลดลง (การเคลื่อนที่ไปยังพื้นไม่ไบแอส) ตัวเลขบนแกนนอนของแต่ละแท่งคือเปอร์เซ็นต์ของ $EZ_1(k)$ สำหรับแท่งท้ายสุดนั้นจะเป็นขนาดของความเปลี่ยนแปลงสัมพัทธ์ในความผิดพลาดรวมทั้งชุดข้อมูล $\left|\Delta E_{\mathbf{x}}[EL(\mathbf{x})] / E_{\mathbf{x}}[EL_1(\mathbf{x})]\right|$

ผลลัพธ์ตามอุดมคติของ Bagging ห้าโซนแรกจะมี $\Delta EZ(k) = -EZ_1(k)$ เพราะว่า Bagging ลดความผิดพลาดเหล่านั้นไปจนหมด ห้าแท่งแรกจึงเป็นสีขาวด้วยขนาดเท่ากับ 1 ตรงกันข้ามกับในห้าโซนหลังที่ $\Delta EZ(k) = 1 - EZ_1(k)$ เพราะว่า Bagging เพิ่มความผิดพลาดไปเป็น 1 ทุกแท่งในห้าโซนหลังนี้จึงมีสีดำ ความผิดพลาดทั้งหมดซึ่งแสดงโดยแท่งสุดท้ายไม่สามารถคำนวณได้จากข้อมูลการวิเคราะห์โซนในรูปที่ 11 ทั้งนี้ในการคำนวณจำเป็นต้องรู้จำนวนตัวอย่างที่ในแต่ละโซนด้วย ซึ่งเป็นข้อมูลที่ไม่ได้แสดงไว้ในที่นี้เนื่องจากจะทำให้แผนภาพซับซ้อนจนเกินไป

ในโซนที่ไม่มีสมาชิก เราจะใส่เลข 0 ไว้ใต้แท่ง ส่วนในบางแท่งที่ยากจะเห็นสี เราจะใส่สัญลักษณ์ 'b' อันหมายถึงสีดำ และ 'w' อันหมายถึงสีขาว ไว้ที่แท่งนั้น



รูปที่ 12 ผลลัพธ์ในอุดมคติหลังทำ Bagging เป็นการทำจินตทัศน์อีกหนึ่งทางเลือกแทนที่จะใช้แบบรูปที่ 10

3.3.2 การวิเคราะห์โซนของ Bagging

รูปที่ 13 แสดงการวิเคราะห์โซนของ Bagging ในชุดข้อมูลทั้งสิบหกชุด ในที่นี้ ตัวแปร EL_1 และ EL_2 ในรูปที่ 11 ก็คือค่าคาดหวังความสูญเสียของ C4.5 "ก่อน" และ "หลัง" การทำ Bagging ตามลำดับ จากผลลัพธ์ที่ได้ เราพบข้อสังเกตที่น่าสนใจที่ไม่เคยปรากฏมาก่อนเลยในการวิเคราะห์ไบแอส-แวเรียนซ์ ดังนี้

(1) *"ขนาด" การเคลื่อนที่ที่ไม่เป็นอุดมคติ*

อย่างที่กล่าวไว้แล้วว่า ขนาดการเคลื่อนที่ $|\Delta EZ(k)|$ ในทางปฏิบัตินั้นอาจไม่ตรงตามอุดมคติ ผลลัพธ์จากรูปที่ 13 ทำให้พบว่าในเขตไม่ไบแอส เฉพาะโซนแรกเท่านั้นที่เกือบเป็นไปตามอุดมคติ ในโซนแรกของ 13 ชุดข้อมูล ความผิดพลาดเฉลี่ยถูกลดลงไปได้มากกว่า 70% และใน 6 ชุดข้อมูล ความผิดพลาดเฉลี่ยถูกลดลงไปได้อย่างน้อย 80% และก็ถูกลดไปได้มากกว่า 50% ใน 14 ชุดข้อมูล แม้ว่าความผิดพลาดเฉลี่ยในโซนที่สามและโซนที่สี่จะลดลง แต่ก็ไม่เคยลงไปแตะระดับ 50% เลย ซึ่งขนาดการเคลื่อนที่ที่ไม่เป็นอุดมคตินี้ก็สามารถพบเห็นได้เช่นกันในโซนไบแอส

การทดลองนี้ใช้ตัวเรียนรู้แบบเดี่ยวๆ เพียง 25 ตัว และตัวอย่างฝึกมีจำนวนน้อย จึงอาจมีการโต้แย้งว่าสิ่งที่สังเกตได้นี้เกิดจากตัวเรียนรู้แบบเดี่ยวๆ ที่ไม่มากพอ หรือตัวอย่างฝึกสอนที่น้อยเกิน อย่างไรก็ตาม เราได้ทดลองสร้างตัวเรียนรู้แบบเดี่ยวๆ มากกว่านี้ โดยใช้ตัวอย่างฝึกสอนมากยิ่งขึ้น ผลก็ยังคงปรากฏเหมือนเดิม

(2) *"ทิศทาง" การเคลื่อนที่ที่แปลกประหลาด*

เราพบว่าการเคลื่อนที่ไม่เป็นไปตามอุดมคติในอีกมุมมองหนึ่ง คือ เราพบว่า*ทิศทาง*การเคลื่อนที่ไม่ได้มีรูปแบบแน่ชัด มีเพียงสองโซนแรกที่รับประกันว่าจะลง ขณะที่โซนสุดท้ายรับประกันว่าจะขึ้น ในชุดข้อมูล

P2+10%NOISE ค่าเฉลี่ยความผิดพลาดของทุกโซนยกเว้นโซนสุดท้ายพร้อมใจกันลง! แม้ตัวอย่างที่ไบแอสมากๆ ที่อยู่ในโซนที่เก้า ซึ่งค่าเฉลี่ยความผิดพลาดสูงถึง 80% ก็ยังถูก Bagging ทำให้ค่าเฉลี่ยความผิดพลาดนี้ลดลงได้ เช่นเดียวกับตัวอย่างในโซนที่เจ็ดของชุดข้อมูล SPLICE-2 ก็มีค่าเฉลี่ยความผิดพลาดลดลง ขณะที่สุดขั้วตรงข้ามคือชุดข้อมูล DNF ที่ตัวอย่างในสองโซนแรกเท่านั้นที่มีค่าเฉลี่ยความผิดพลาดลดลง สิ่งที่สังเกตได้นี้ขัดแย้งกับคำอธิบายผลของ Bagging ตามอุดมคติที่กระทำต่อตัวอย่างไบแอสและตัวอย่างไม่ไบแอส จึงเป็นเรื่องน่าสนใจที่จะหาคำอธิบายปรากฏการณ์นี้

จากข้อสังเกตที่ (1) และ (2) แม้ว่าเราจะพบการเคลื่อนที่ที่ไม่เป็นไปตามอุดมคติทั้งในแง่ของขนาดและทิศทาง เราก็พบพฤติกรรมที่ดีอย่างหนึ่งว่าในแต่ละชุดข้อมูล ถ้า $i < j$ แล้ว ตัวอย่างใน Z_i จะได้ประโยชน์จาก Bagging มากกว่าตัวอย่างใน Z_j ซึ่งคุณสมบัตินี้อาจนำไปประยุกต์ใช้เพื่อการปรับปรุง LOBAG [Valentini & Dietterich, 2003] ได้ ซึ่งจะอธิบายในหัวข้อ 3.4

(3) *Bagging ไม่ใช่อัลกอริทึมแบบกล่องดำ*

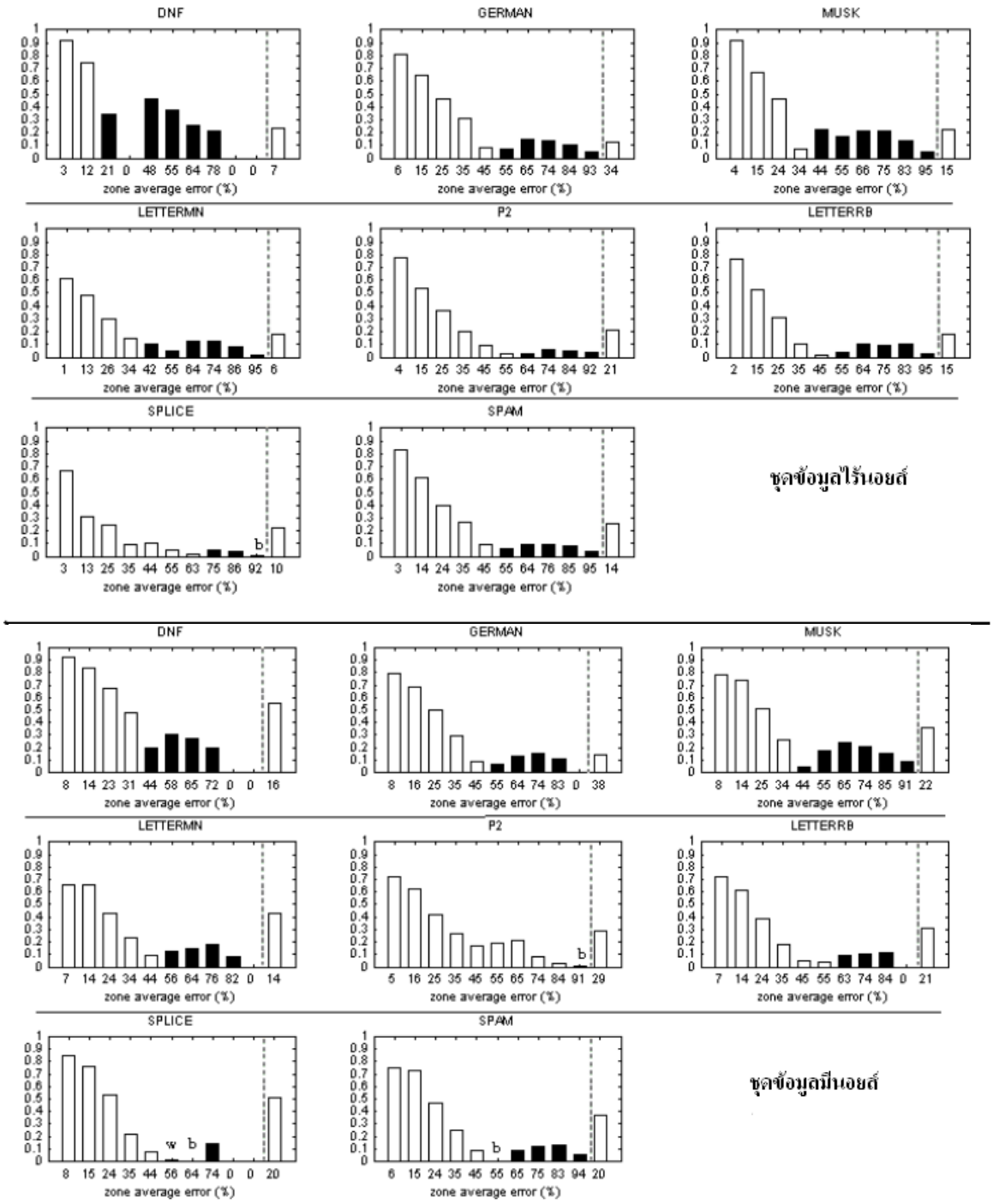
จากพฤติกรรมตามอุดมคติของ Bagging จึงเป็นเรื่องที่ไม่น่าแปลกใจหากบางคนจะคิดว่า Bagging เป็นอัลกอริทึมแบบกล่องดำ ที่ทำให้ตัวอย่างใดๆ ที่มีค่าความผิดพลาดเหมือนๆ กัน (เช่นตัวอย่างที่อยู่ในโซนเดียวกัน) เคลื่อนที่เหมือนๆ กัน นั่นคือประสิทธิภาพของ Bagging ไม่ควรขึ้นอยู่กับบริบทอื่นๆ เช่นตัวอย่างที่มีความผิดพลาด 21% และ 23% เมื่อทำ Bagging แล้วก็ควรจะเคลื่อนที่เช่นเดียวกัน โดยไม่สนใจว่าตัวอย่างนี้มาจากชุดข้อมูลใด

รูปที่ 13 แสดงให้เห็นชัดเจนว่าสมมติฐานกล่องดำไม่เป็นความจริง ความจริงก็คือ Bagging ขึ้นอยู่กับบริบทในการเรียนรู้อื่นๆ ด้วย

(4) *ทิศทางการเคลื่อนที่ของตัวอย่างก่อนและหลังจากใส่นอยส์*

นอกจากข้อสังเกตทั้งหมดที่กล่าวมาแล้ว บางคนยังอาจคิดว่า อย่างน้อย สมมติฐานกล่องดำก็ยังคงเป็นความจริง ถ้าเราพิจารณาสองตัวอย่างในโซนเดียวกันในบริบทการเรียนรู้ที่เหมือนๆ กัน เช่น ตัวอย่างใน Z_3 ของชุดข้อมูล DNF และตัวอย่างใน Z_3 ของชุดข้อมูล DNF+10%NOISE ก็ควรจะเคลื่อนที่ไปในทางเดียวกัน อย่างไรก็ตามความคิดนี้ก็ไม่ถูกต้องเสมอไป ตัวอย่างโต้แย้งสามารถพบได้ในหลายๆ ชุดข้อมูล เช่น ใน Z_7 ของชุดข้อมูล P2 และ P2+10%NOISE ใน Z_5 ของชุดข้อมูล LETTERMN และ LETTERMN+10%NOISE

สิ่งนี้ยังคงเป็นปัญหาที่เปิดกว้างอยู่ต่อไปสำหรับการพัฒนาทฤษฎีใหม่สำหรับ Bagging ที่สามารถอธิบายปรากฏการณ์ทั้งสี่ที่สามารถพบเห็นจากการวิเคราะห์โซนนี้ได้ คำอธิบาย Bagging เชิงทฤษฎีที่มีอยู่ในปัจจุบัน [Buja & Stuetzle, 2006; Grandvalet, 2004; Buhlmann &Yu, 2002] นั้นยังไม่เพียงพอ



รูปที่ 13 การวิเคราะห์โซนของ Bagging

3.4 กลุ่มก้อนตัวจำแนกแบบหลายระดับ

หัวข้อนี้นำเสนอการใช้กลุ่มก้อนตัวจำแนกแบบหลายระดับที่รวม Adaboost กับ Bagging เข้าด้วยกัน เราจะใช้ Bagging เป็นกลุ่มก้อนตัวจำแนกชั้นนอกและใช้ Adaboost เป็นกลุ่มก้อนตัวจำแนกชั้นใน Bagging เนื่องจาก Bagging สามารถลดแวเรียนซ์ของตัวเรียนรู้พื้นฐานได้อย่างมีประสิทธิภาพ กุญแจความสำเร็จของ Bagging คือ ตัวเรียนรู้พื้นฐานที่ต้องมีคุณสมบัติของไบแอสต่ำ ไม่ว่าจะมีแวเรียนซ์เท่าไรก็ตาม ในทางกลับกันเราพบว่าในหลายๆ การศึกษาวิจัย Adaboost แสดงให้เห็นว่าเป็นตัวเรียนรู้ที่มีไบแอสต่ำมาก กล่าวคือ Adaboost สามารถปรับตัวให้ตรงกับตัวอย่างฝึกได้ดีมาก ดังนั้นการรวม Adaboost กับ Bagging โดยทำเป็นกลุ่มก้อนตัวจำแนก

แบบหลายระดับ กลุ่มก้อนตัวจำแนกที่นำเสนอจะใช้ Adaboost เป็นตัวเรียนรู้พื้นฐานของ Bagging เหตุผลก็คือ Adaboost สามารถใช้ไบแอสต่ำแต่อาจมีแวเรียนซ์สูง ซึ่งจะถูกลดโดย Bagging อีกทีหนึ่ง

ในการทดลองเรานำ Adaboost มาทั้งหมด 25 ตัวเพื่อใช้เป็นตัวเรียนรู้พื้นฐานของ Bagging และ Adaboost แต่ละตัวจะทำ 25 รอบ ในการวิเคราะห์โซนของกลุ่มก้อนตัวจำแนกแบบหลายระดับนี้ เราใช้ EL_1 และ EL_2 ในรูปที่ 11 เป็นค่าคาดหวังความสูญเสียของ Adaboost "ก่อน" และ "หลัง" การใช้ Bagging ผลของการวิเคราะห์โซนที่ได้แสดงในรูปที่ 14

ผลที่ได้ยืนยันข้อสังเกตที่เราพบในหัวข้อที่แล้ว กล่าวคือเราคงพบขนาดและทิศทางการเคลื่อนที่ที่ไม่เป็นอุดมคติ นอกจากนี้การเพิ่มน้อยซ์ 10% ส่งผลน้อยมากต่อทิศทางของตัวอย่างหลังใช้ Bagging จุดเด่นของกลุ่มก้อนตัวจำแนกแบบหลายระดับนี้ก็คือ Bagging จะทำหน้าที่ลดความผิดพลาดในโซนแรกๆ หลังจากที่ Adaboost ได้ทำการลดความผิดพลาดในโซนอื่นๆ แล้ว

3.5 แนวทางในอนาคต

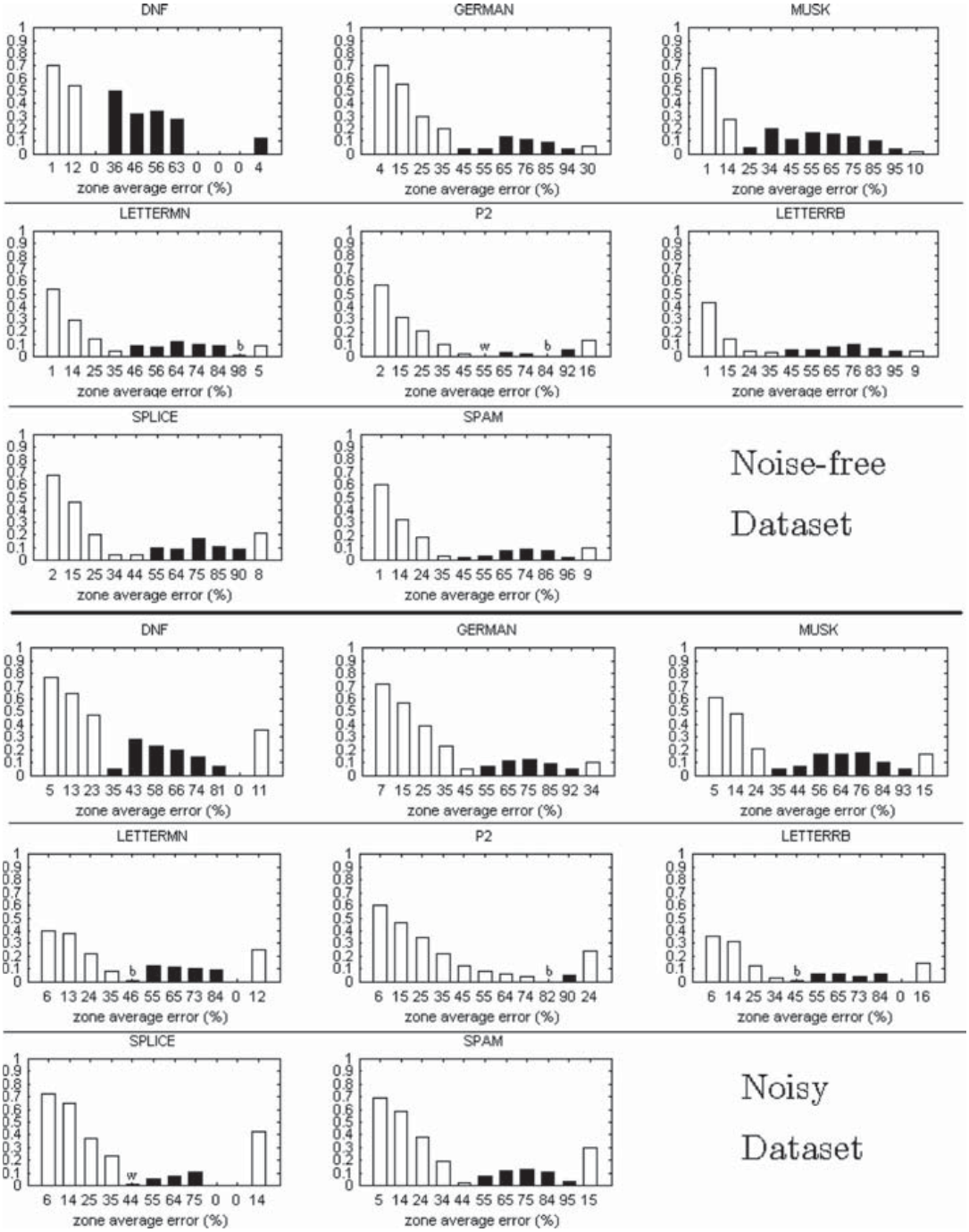
ข้อสังเกตจากหัวข้อที่ผ่านมาจุดประกายที่เป็นประโยชน์ให้สามารถต่อยอดได้ทั้งในทางทฤษฎีและปฏิบัติ โดยมีสองแนวทางที่เป็นไปได้ดังนี้คือ

แนวทางแรก จากการสังเกตที่ได้ในหัวข้อ 3.3.2 แสดงให้เห็นพฤติกรรมเชิงปฏิบัติของ Bagging อย่างหนึ่งคือ Bagging ชื่นชอบตัวอย่างในโซนแรกๆ ข้อสังเกตนี้สามารถนำไปใช้ระบบต่างๆ [Valentini & Dietterich, 2003] โดยแนวคิดของ LOBAG นั้นมีอยู่ว่าให้ใช้ SVM ที่ไบแอสต่ำๆ เป็นตัวเรียนรู้แบบเดี่ยวๆ ของ Bagging ในการเลือก SVM ไบแอสต่ำทำได้โดยการปรับค่าพารามิเตอร์เทียบกับเคอร์เนล และการปรับชนิดของเคอร์เนลไปด้วย ต่อจากนั้นก็คำนวณค่าไบแอสของ SVM โดยใช้นิยามของ Domingos เนื่องจากว่าเค้าไม่รู้ค่าของ $P(\mathbf{x})$ จึงกำหนดให้ $P(\mathbf{x})$ เป็นยูนิฟอร์ม ด้วยสมมติฐานนี้ ไบแอสเฉลี่ยของตัวเรียนรู้จึงเป็นสัดส่วนโดยตรงกับจำนวนตัวอย่างไบแอส

$$E_{\mathbf{x}}[B(\mathbf{x})] = \sum_{\mathbf{x}} P(\mathbf{x})B(\mathbf{x}) \propto \sum_{\mathbf{x}} B(\mathbf{x}) \tag{56}$$

จากสมการข้างต้น $E_{\mathbf{x}}[B(\mathbf{x})]$ ของสองอัลกอริทึมการเรียนรู้จะเท่ากับการเปรียบเทียบจำนวนสมาชิกของโซนไม่ไบแอสในสองอัลกอริทึม ด้วยการนับง่ายๆ สมาชิกทุกตัวในโซนไม่ไบแอสจะสำคัญเท่ากันหมด อย่างไรก็ตามการวิเคราะห์โซนชี้ทางให้เห็นว่า Bagging ไม่ได้ชื่นชอบตัวอย่างไม่ไบแอสทุกตัวเท่าๆ กัน! ถ้าให้ $i < j$ แล้วตัวอย่างใน Z_i จะได้ประโยชน์จาก Bagging มากกว่าตัวอย่างใน Z_j เพราะฉะนั้น ทางหนึ่งที่จะปรับปรุงระบบ LOBAG ก็คือการหาวิธีวัดที่สามารถบอกถึงความไม่เท่าเทียมกันของตัวอย่างไม่ไบแอสได้ ที่น่าสนใจคือนิยามไบแอส-แวเรียนซ์ ของนักวิจัยท่านอื่น เช่น Breiman [1998] หรือ Kohavi & Wolpert [1996]

ประการที่สอง การสังเกตพฤติกรรมของ Bagging ได้ให้แง่คิดบางประการที่สามารถนำมาสานต่องานทางด้านทฤษฎี ผลการวิจัยหลังๆ ค้นพบว่า Bagging มีความเสถียรภาพในแง่ที่เป็นทางการและสามารถป้องกันการโอเวอร์ฟิตได้ [Elisseeff et al., 2005] จากคุณสมบัตินี้เอง ทำให้เราสามารถหาขอบเขตของความผิดพลาดทั่วไปแบบ non-asymptotic ของ Bagging จากความผิดพลาดในข้อมูลฝึกสอนได้เป็นครั้งแรก ซึ่งผลนี้ก็ตรงกับข้อสังเกตของเรา อย่างไรก็ตาม การวิเคราะห์ของ Elisseeff et al. ก็ตั้งอยู่บนสมมติฐานที่จำกัด เช่น ตัวเรียนรู้แบบเดี่ยวๆ ต้องมีเสถียรภาพ และไม่ได้กล่าวอะไรเกี่ยวกับนอยส์ไว้เลย เพราะฉะนั้น ผลลัพธ์จึงสามารถพัฒนาไปได้อีก เราเห็นว่าทฤษฎีใดก็ตามที่จะอธิบาย Bagging ต้องสามารถอธิบายผลการทดลองจากการวิเคราะห์โซนนี้ได้ คำอธิบาย Bagging อื่นๆ [Buja & Stuetzle, 2006; Grandvalet, 2004; Buhlmann &Yu, 2002] ไม่ได้ให้แนวทางอะไรกับผลลัพธ์ที่เราได้มาเลย



รูปที่ 14 การวิเคราะห์โซนของ Bagging

3.6 สรุปการวิจัยกลุ่มก้อนตัวจำแนกที่สามารถลดไบแอสและแวเรียนซ์

หัวข้อนี้นำเสนอการวิเคราะห์โซน ซึ่งเป็นทางเลือกของกรอบการวิเคราะห์ไบแอส-แวเรียนซ์ สำหรับปัญหาการจำแนก เป้าหมายหลักของกรอบการวิเคราะห์ใหม่นี้ก็เพื่อให้ภาพที่เข้าใจง่ายของการเปลี่ยนแปลงพฤติกรรมของอัลกอริทึมการเรียนรู้เมื่อบริบทการเรียนรู้เปลี่ยนไป บริบทสำคัญที่ได้เน้นหนักในที่นี้ก็คือการที่เทคนิคกลุ่มก้อนตัวจำแนกได้ถูกเพิ่มเข้ามาให้กับการเรียนรู้แบบต้นไม้ทั้งในกรณีที่ไร้นอยส์และในกรณีที่มีนอยส์ การวิเคราะห์ของเรานี้ไม่จำเป็นต้องกำหนดว่าไม่ให้มีนอยส์เหมือนกับการวิเคราะห์ไบแอส-แวเรียนซ์ ทั่วไป ยิ่งไปกว่านั้น

การทดลองด้วยกรอบการวิเคราะห์ของเราก็ให้อะไรที่น่าตื่นตาตื่นใจหลายอย่าง ซึ่งไม่สามารถพบเห็นได้ในกรอบการวิเคราะห์ไบแอส-แวเรียนซ์ ซึ่งผลที่ได้นี้เป็นประโยชน์ต่อทั้งการทำความเข้าใจในเชิงทฤษฎีของระบบอย่าง Bagging และการปรับปรุงประสิทธิภาพของระบบอย่าง LOBAG ต่อไป

4. สรุปงานวิจัย

ในงานวิจัยนี้เราได้นำเสนอวิธีการสำหรับการพัฒนาฟังก์ชันเคอร์เนลที่ยืดหยุ่นมากพอที่จะปรับตัวเข้ากับงานที่ทำ และวิธีการสำหรับการพัฒนาเทคนิคกลุ่มก้อนตัวจำแนกที่สามารถลดไบแอสและแวเรียนซ์ ในส่วนแรกเราได้นำเสนอการรวมเชิงเส้นแบบมีน้ำหนักของเคอร์เนลอาร์บีเอฟแบบหลายสเกลสำหรับซัพพอร์ตเวกเตอร์แมชชีนทั้งสำหรับงานจำแนกประเภทข้อมูลและงานรีเกรซชัน ผลการทดลองได้แสดงให้เห็นว่าการรวมเคอร์เนลอาร์บีเอฟแบบหลายสเกลที่นำเสนอนั้นสามารถปรับปรุงประสิทธิภาพของเคอร์เนลอาร์บีเอฟเดี่ยว ๆ ได้อย่างดี และทำให้เคอร์เนลที่ได้ยืดหยุ่นได้มากขึ้น ในส่วนของการพัฒนาเทคนิคกลุ่มก้อนตัวจำแนกนั้น เราได้นำเสนอวิธีการใหม่สำหรับวิเคราะห์พฤติกรรมของเทคนิคกลุ่มก้อนตัวจำแนก ทำให้เราสามารถเข้าใจถึงพฤติกรรมของกลุ่มก้อนตัวจำแนกได้ดียิ่งขึ้น

รายการอ้างอิง

Asuncion, A. and Newman, D.J. (2007) UCI machine learning repository, [http://www.ics.uci.edu/~mlearn/MLRepository.html], Irvine, CA: University of California, Department of Information and Computer Science.

Ayat, N. E., Cheriet, M., Remaki, L., and Suen, C.Y. (2001) KMOD-A new support vector machine kernel with moderate decreasing for pattern recognition. In *Proceedings on Document Analysis and Recognition*, pp. 1215-1219, Seattle, USA.

Bartlett, P. L. and, J. (1999) Generalization performance of support vector machines and other pattern classifiers. *Advances in Kernel Methods - Support Vector Learning*, Schoelkopf, B., Burges, C. J. and Smola, A. J. (eds), pp. 43-54. MIT Press.

Bauer, E. and Kohavi, R. (1999) An empirical comparison of voting classification algorithms: Bagging, boosting and variants, *Machine Learning*, 36, 105-139.

Beyer, H.G. and Schwefel, H.P. (2002) Evolution strategies: A comprehensive introduction., Natural Computing, 1 (1): 3-52.

Blake, C., Keogh, E. and Merz, C. (1998) UCI repository of machine learning databases. Department of Information and Computer Science, University of California, Irvine. [Online]. Available: http://www.ics.uci.edu/~mlearn/MLRepository.html

Bousquet, O. and Elisseeff, A. (2002) Stability and generalization, *Journal of Machine Learning Research*, Vol. 2, pp. 499-526.

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

Breiman, L. (1998) Arcing classifier, *The Annals of Statistics*, 26, 801-849.

Buhlmann, P. and Yu, B. (2002) Analyzing bagging, *The Annals of Statistics*, 30, 927-961, 2002.

Buja, A. and Stuetzle, W. (2006) Observation on bagging, *Statistica Sinica*, 16, 323-351.

Chapelle, O., Vapnik, V., Bousquet, O., and Mukherjee, S. (2002) Choosing multiple parameters for support vector machines. *Machine Learning*, Vol. 46, No. 1, pp. 131 – 159.

Domingos, P. (2000). A unified bias-variance decomposition and its applications. In *Proceedings of the* 17^{th} *International Conference of Machine Learning*, pp. 231-238.

deDoncker, E., Gupta, A. and Greenwood, G. (1996) Adaptive integration using evolutionary strategies, *Proceedings of 3rd International Conference on High Performance Computing*, pp. 94-99, 19-22 December.

Elisseeff, A., Evgeniou, T. and Pontil, M. (2005) Stability of randomized learning algorithms, *Journal of Machine Learning Research*, 6, 55-79.

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

Friedrichs, F., and Igel, C. (2004) Evolutionary tuning of multiple SVM parameters. In *Proceedings of the 12th European Symposium on Artificial Neural Networks (ESANN 2004)*, pp. 519-524.

Fröhlich, H. and Zell, A. (2005) Efficient parameter selection for support vector machines in classification and regression via model-based global optimization, *IEEE International Joint Conference on Neural Networks (IJCNN '05)*, No. 3, pp. 1431-1436.

Geman, S., BienenStock E. and Doursat, R. (1992) Neural networks for the bias-variance dilemma, *Neural Computation*, 4 (1), 1-58.

Grandvalet, Y. (2004) Bagging equalizes influence, *Machine Learning*, 55 (3), 251-270.

Hyndman, R.J. and Koehler, A.B. (2006) Another look at measures of forecast accuracy, *International Journal of Forecasting*, Vol. 22, Issue 4, Netherlands.

Howley, T., and Madden, M.G. (2005) The genetic kernel support vector machine: Description and evaluation. *Artificial Intelligence Review*, Vol. 24, pp. 379 – 395.

James, G.M. (2003) Variance and bias for general loss functions, *Machine Learning*, 51, 115-135.

Kecman, V. (2001) *Learning and Soft Computing: Support Vector Machines, Neural Networks, and Fuzzy Logic Models*, MIT Press, London.

Kohavi, R. and Wolpert, D. (1996) Bias plus variance decomposition for zero-one loss functions, *The 13th International Conference on Machine Learning*, pp. 275-283.

Kong, E. B. and Dietterich, T. (1995) Machine learning bias, statistical bias and statistical variance of decision tree algorithm, *Technical Report, Oregon State University*.

Kuncheva, L. I. and Whitaker, C. J. (2003) Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy, *Machine Learning*, 51, 181-207.

Mitchell , T.M. (1997) Machine Learning. McGraw-Hill, New York.

Runarsson, T.P. (2004) Asynchronous parallel evolutionary model selection for support vector machines. *Neural Information Processing – Letter and Reviews*, Vol. 3, No. 3.

Schittkowski, K. (2005) Optimal parameter selection in support vector machines. *Journal of Industrial and Management Optimization*, Vol. 1, No. 4, pp. 465-476.

Schölkopf, B., Burges, C., and Smola, A. J. (1998) *Advances in Kernel Methods: Support Vector Machines.* MIT Press, Cambridge, MA.

Schölkopf, B. and Smola, A.J. (2002) *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*, MIT Press, London.

Shawe-Taylor J. and Cristianini N. (2004) Kernel methods for pattern analysis. Cambridge University Press, UK.

Smola, A.J. and Schölkopf, B. (1998) A tutorial on support vector regression, *NEUROCOLT Technical Report NC-TR-98-030*, Royal Holloway College, London.

Suen, Y. L., Melville, P. and Mooney, R. J. (2005) Combining bias and variance reduction techniques for regression trees, *The 16th European Conference on Machine Learning*, pp. 741-749.

Tibshirani, R. (1996) Bias, variance and prediction error for classification rules, *Technical Report, University of Toronto*.

Valentini, G. (2005) An experimental bias-variance analysis of SVM ensembles based on resampling techniques, *IEEE Transactions on Systems, Man and Cybernetics*, 35 (6), 1252-1271.

Valentini, G. and Dietterich, T. (2003) Low bias bagged SVMs, *The 20th International Conference on Machine Learning*, pp. 752-759.

Valentini, G. and Dietterich, T. (2004) Bias-variance analysis of SVMs for the development of SVM-based ensemble methods, *Journal of Machine Learning Research*, 5, 725-755.

Vapnik, V. (1995). *The Nature of Statistical Learning Theory*. Springer-Verlag, New York, USA.

Vapnik, V. (1998) *Statistical Learning Theory*. Wiley.

Vapnik V, Chervonenkis A (1971) On the uniform convergence of relative frequencies of events to their probabilities. Theory of Probability and its Applications, 16(2): 264-280.

Webb, G. I. (2000) Multiboosting: A technique for combining boosting and wagging, *Machine Learning*, 40 (2), 159-196.

Xuefeng, L., and Fang, L. (2002) Choosing multiple parameters for SVM based on genetic algorithm. *International Conference on Signal Processing (ICSP'02)*, No. 1, pp. 117-119.

II. ผลที่ได้จากโครงการ

งานวิจัยที่ตีพิมพ์ในวารสารวิชาการระดับนานาชาติ

1. Tanasanee Phienthrakul, Boonserm Kijsirikul, Evolutionary strategies for hyperparameters of support vector machines based on multi-scale radial basis function kernels, *Soft Computing*, 2009.
2. Ratthachat Chatpatanasiri, Prasertsak Pungprasertying, Boonserm Kijsirikul, Zone analysis: A visualization framework for classification problems, *Artificial Intelligence Review*, 2009.

การนำเสนอผลงานในการประชุมวิชาการ

3. Tanasanee Phienthrakul, Boonserm Kijsirikul, Hiroya Takamura and Manabu Okumura, Sentiment classification with SVMs and evolutionary combined kernels, *The 6th International Joint Conference on Computer Science and Software Engineering (JCSSE2009)*, May 13-15, 2009, Phuket, Thailand.
4. Tanasanee Phienthrakul and Boonserm Kijsirikul, Adaptive stabilized multi-rbf kernel for support vector regression, *The 2008 International Joint Conference on Neural Networks (IJCNN2008)*, Hong Kong, June 1-6, 2008.
5. Tanasanee Phienthrakul and Boonserm Kijsirikul, GPES: An algorithm for evolving hybrid kernel functions of support vector machines, *The 2007 IEEE Congress on Evolutionary Computation*, Singapore, September 25-28, 2007.
6. Tanasanee Phienthrakul and Boonserm Kijsirikul, Evolving parameters of multi-scale radial basis function kernels for support vector machines, *The third IASTED International Conference on Advances in Computer Science and Technology (ACST 2007)*, Phuket, April 2-7, 2007.
7. รัฐฉัตร ฉัตรพัฒนศิริ, ประเสริฐศักดิ์ ผุงประเสริฐยิ่ง, การวิเคราะห์โซน: กรอบการวิเคราะห์เชิงจินตภาพสำหรับปัญหาการจำแนก, การประชุมวิชาการวิทยาการคอมพิวเตอร์และวิศวกรรมคอมพิวเตอร์แห่งชาติฯ ครั้งที่ 12 (NCSEC 2008), 171-178, 20-21 พฤศจิกายน 2551

การผลิตบัณฑิต

โครงการนี้ได้ผลิตบัณฑิตในระดับดุษฎีบัณฑิตและมหาบัณฑิตที่ทำวิทยานิพนธ์เกี่ยวข้องกับโครงการดังต่อไปนี้

1. นางสาวธนัสนี เพียรตระกูล ระดับดุษฎีบัณฑิต
2. นายประเสริฐศักดิ์ ผุงประเสริฐยิ่ง ระดับมหาบัณฑิต