การเปรียบเทียบการคัดเลือกตัวแปรอิสระที่มีปัญหาสหสัมพันธ์เชิงเส้นพหุ ด้วยวิธีการถดถอยแบบริดจ์และการค้นหาแบบต้องห้าม

นิศาชล งามประเสริฐสิทธิ์

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
วิทยาศาสตรมหาบัณฑิต (สถิติ)
คณะสถิติประยุกต์
สถาบันบัณฑิตพัฒนบริหารศาสตร์
2555

การเปรียบเทียบการคัดเลือกตัวแปรอิสระที่มีปัญหาสหสัมพันธ์เชิงเส้นพหุด้วย
วิธีการถดถอยแบบริดจ์และการค้นหาแบบต้องห้าม
นิศาชล งามประเสริฐสิทธิ์
คณะสถิติประยุกต์

รองศาสตราจารย์..ที่ปรึกษาวิทยานิพนธ์หลัก
(ดร. จิราวัลย์ จิตรถเวช)

คณะกรรมการสอบวิทยานิพนธ์ ได้พิจารณาแล้วเห็นสมควรอนุมัติให้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรมหาบัณฑิต (สถิติ)

รองศาสตราจารย์..ประธานกรรมการ
(ดร. วิชิต หล่อจีระชุณห์กุล)

รองศาสตราจารย์..กรรมการ
(ดร. จิราวัลย์ จิตรถเวช)

ผู้ช่วยศาสตราจารย์..กรรมการ
(ดร. แสงหล้า ชัยมงคล)

รองศาสตราจารย์..รักษาราชการแทนคณบดี
(ดร. ระวีวรรณ เอื้อพันธ์วิริยะกุล)

ตุลาคม 2555

บทคัดย่อ

ชื่อวิทยานิพนธ์ การเปรียบเทียบการคัดเลือกตัวแปรอิสระที่มีปัญหาสหสัมพันธ์เชิงเส้นพหุด้วยวิธีการถดถอยแบบริดจ์และการค้นหาแบบต้องห้าม

ชื่อผู้เขียน นางสาวนิศาชล งามประเสริฐสิทธิ์

ชื่อปริญญา วิทยาศาสตรมหาบัณฑิต (สถิติ)

ปีการศึกษา 2555

การศึกษามีวัตถุประสงค์เพื่อเปรียบเทียบการคัดเลือกตัวแปรอิสระในการวิเคราะห์การถดถอยเชิงเส้นพหุที่ตัวแบบมีตัวแปรอิสระที่เกี่ยวข้องและไม่เกี่ยวข้องกับตัวแปรตาม โดยตัวแปรอิสระที่เกี่ยวข้องมีความสัมพันธ์กันสูง 0.95, 0.99, 0.999 และ 0.9999 การคัดเลือกตัวแปรอิสระใช้วิธีการถดถอยแบบขั้นตอนที่ประมาณค่าสัมประสิทธิ์การถดถอยด้วยวิธีกำลังสองน้อยที่สุดและวิธีการถดถอยแบบริดจ์โดยใช้วิธีการประมาณค่าพารามิเตอร์ริดจ์4 วิธีคือ วิธีโฮเอิร์ล,เคนนาร์ดและ บาลด์วิน (Hoerl, Kennard and Baldwin) วิธีลอว์เลสและแวง (Lawless and Wang) วิธีนอมูระ (Nomura)และวิธีคาลาฟและชูเกอร์(Khalaf and Shukur)กับการค้นหาแบบต้องห้ามที่ใช้ฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ย(MSE) และค่าความคลาดเคลื่อนกำลังสองเฉลี่ยปรับด้วยฟังก์ชันการลงโทษ(Penalty function) เกณฑ์ที่ใช้ในการเปรียบเทียบการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบคือร้อยละของจำนวนครั้งที่แต่ละวิธีสามารถคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบได้ตามตัวแบบจริง, ตัวแบบ Overspecification, ตัวแบบ Underspecification และตัวแบบ Misspecification การศึกษาใช้วิธีการจำลองข้อมูล กำหนดขนาดตัวอย่างเท่ากับ 20, 60 และ 100 และกระทำซ้ำในแต่ละสถานการณ์ 500 ครั้ง

จากการศึกษาพบว่าวิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยและค่าความคลาดเคลื่อนกำลังสองเฉลี่ยปรับด้วยฟังก์ชันการลงโทษมีร้อยละของการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบได้ถูกต้องตามตัวแบบจริงมากกว่าวิธีการถดถอยแบบขั้นตอนที่ประมาณค่าสัมประสิทธิ์การถดถอยด้วยวิธีกำลังสองน้อยที่สุดและวิธีการถดถอยแบบริดจ์ในทุกขนาดตัวอย่างเมื่อสัมประสิทธิ์สหสัมพันธ์ระหว่างตัวแปรอิสระเป็น 0.95, 0.99 และ 0.999 แต่เมื่อค่าสัมประสิทธิ์สหสัมพันธ์ระหว่างตัวแปรอิสระสูงขึ้นเป็น 0.9999 วิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็น

ค่าความคลาดเคลื่อนกำลังสองเฉลี่ยและวิธีการถดถอยแบบขั้นตอนที่มีการประมาณค่าพารามิเตอร์กำลังสองน้อยที่สุดและแบบริดจ์ มีร้อยละของการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบได้ถูกต้องมีค่าต่ำเมื่อขนาดตัวอย่างเท่ากับ 20 แต่จะค่อยๆเพิ่มขึ้นเมื่อขนาดตัวอย่างเพิ่มขึ้นและมีร้อยละของตัวแบบ Underspecificationลดลงอย่างเห็นได้ชัด ในขณะที่วิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยปรับด้วยฟังก์ชันการลงโทษมีร้อยละการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบได้ถูกต้องสูงและค่อนข้างคงที่โดยไม่ขึ้นกับขนาดตัวอย่างและค่าสัมประสิทธิ์สหสัมพันธ์ นอกจากนั้นผลการศึกษาไม่พบตัวแบบ Underspecification และ Misspecification เลย มีเพียงตัวแบบ Overspecificationซึ่งเป็นปัญหาที่รุนแรงน้อยกว่าการที่ตัวแบบมีตัวแปรอิสระที่เกี่ยวข้องขาดหายไป

คำสำคัญ: การคัดเลือกตัวแปร, การค้นหาแบบต้องห้าม, วิธีการถดถอยแบบขั้นตอน, วิธีการถดถอยแบบริดจ์, สหสัมพันธ์เชิงเส้นพหุ

ABSTRACT

Title of Thesis	A Comparison ofVariable Selection byRidge Regression and Tabu Search with Multicolinearity
Author	Miss NisachonNgamprasertsit
Degree	Master of Science(Statistics)
Year	2012

The purpose of this study is to comparevariable selection methods for multiple linear regression models that have both relative and non-relative variables in full model when predictor variables are highly correlated 0.95, 0.99, 0.999 and 0.9999. The variables are iterativelyby the stepwise regression method. The multiple regression coefficients are estimated by the method of Ordinary Least Square and Ridge Regression by Hoerl, Kennard and Baldwin, Lawless and Wang, Nomura andKhalaf and Shukur methods and by the Tabu Search using two objective functions: mean squared error (MSE) and mean squared error augmented by apenalty function.The criterion of comparison is the percentage of selecting the correct models, the overspecified models, the underspecified models and the misspecified models.The comparisons, using simulation data, are performed with sample size 20, 60 and 100 and repeats 500 times.

From the result of this research,the percentages of selecting the correct models of Tabu Search using both objective functions are higher than those of the stepwise regression method when the multiple regression coefficients are estimated by the method of Ordinary Least Square and Ridge Regression for all sample size when correlations are 0.95, 0.99 and 0.999. But when correlation is 0.9999 and the percentages of selecting the correct models byTabu Search using objective function ofmean squared error and by the stepwise method with OLS estimates and ridge estimates are low when the sample size 20 but increase as the sample size increasesandthe percentages of selecting the overspecified models, the underspecified models and the misspecified models clearly decrease. But the percentage of selecting the correct models of Tabu Search using objective function of mean squared error augmented by a penalty function is high and quite stable, regardless of the sample size and correlation. Moreover the simulation result by Tabu Search using objective function of mean squared error augmented with apenalty function does not select any of the underspecified models and the misspecified models,only select a fewoverspecified models when effects are less serious than those of the underspecified models.

Keywords: Variable selection, Tabu search, Stepwise regression, Ridge regression, Multicollinearity

กิตติกรรมประกาศ

วิทยานิพนธ์นี้สำเร็จลุล่วงไปได้ด้วยดีเนื่องมาจากผู้วิจัยได้รับความกรุณาและช่วยเหลือเป็นอย่างดียิ่งจาก รองศาสตราจารย์ ดร. จิราวัลย์ จิตรถเวช ผู้ซึ่งเป็นทั้งอาจารย์ที่ปรึกษาและอาจารย์ผู้ควบคุมวิทยานิพนธ์ ที่ได้กรุณาสละเวลาให้ความช่วยเหลือ ให้คำปรึกษา ข้อชี้แนะ ข้อคิดเห็นตลอดจนติดตามความคืบหน้าในการจัดทำวิทยานิพนธ์ทุกขั้นตอน รวมทั้งตรวจสอบและแก้ไขข้อบกพร่องต่างๆ อันเป็นประโยชน์ในการจัดทำวิทยานิพนธ์ จนกระทั่งวิทยานิพนธ์สำเร็จเป็ น รูปเล่ม ผู้เขียนขอขอบพระคุณและสำนึกในบุญคุณยิ่ง

ขอขอบพระคุณท่านคณะกรรมการสอบวิทยานิพนธ์ ซึ่งได้แก่ รองศาสตราจารย์ ดร.วิชิต หล่อจีระชุณห์กุล และผู้ช่วยศาสตราจารย์ ดร.แสงหล้า ชัยมงคล ที่ช่วยให้คำแนะนำ และแนวคิดที่เป็นประโยชน์ตลอดจนช่วยตรวจสอบ แก้ไขให้วิทยานิพนธ์นี้สมบูรณ์มากขึ้น

ขอขอบพระคุณคณาจารย์ทุกท่านของภาควิชาคณิตศาสตร์และสถิติ คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยธรรมศาสตร์ ที่ได้เป็นผู้เริ่มต้นในการถ่ายทอดความรู้ทางสถิติให้กับผู้วิจัย

ขอขอบพระคุณ คณาจารย์ทุกท่านของคณะสถิติประยุกต์ สถาบันบัณฑิตพัฒนบริหารศาสตร์ ที่ได้ถ่ายทอดความรู้ และให้คำปรึกษาในทุกด้าน รวมทั้งเจ้าหน้าที่คณะสถิติประยุกต์ และเจ้าหน้าที่ห้องสมุด โดยเฉพาะพี่นิลาวัลย์ เติมผล และพี่ดุษฎี กองเพิ่มพูล ที่ให้ความช่วยเหลือ และอำนวยความสะดวกในการ จัดทำวิทยานิพนธ์ ครั้งนี้เป็นอย่างดี ขอขอบคุณเพื่อนๆอันเป็นที่รักของผู้วิจัยทุกคน สำหรับกำลังใจและความช่วยเหลือที่มีให้มาโดยตลอด โดยเฉพาะกานต์ณัฐ ณ บางช้าง ที่ให้คำปรึกษาในเรื่องทฤษฎี และสาธิต เทพประทานกิจ ที่ให้ความช่วยเหลือในเรื่องการเขียนโปรแกรม ซึ่งเป็นประโยชน์ต่อการทำวิทยานิพนธ์เป็นอย่างมาก

สุดท้ายนี้ ขอขอบพระคุณ และมอบความสำเร็จทั้งหมดจากการทำวิทยานิพนธ์เล่มนี้แด่คุ ณ แม่สายพิน วรรณกร และ คุณน้าวรรณนา วรรณกร ตลอดจนญาติพี่น้องของผู้วิจัย ที่ให้ความห่วงใย ช่วยเป็นกำลังใจ ส่งเสริม สนับสนุนในทุก ๆด้าน ซึ่งเป็นแรงใจที่สำคัญยิ่งของผู้วิจัย จนกระทั่งวิทยานิพนธ์นี้ประสบความสำเร็จดังตั้งใจ

นิศาชล งามประเสริฐสิทธิ์
กันยายน 2555

สารบัญ

สารบัญตาราง

ตารางที่ หน้า

สารบัญภาพ

ภาพที่ หน้า

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของปัญหา

ในปัจจุบันนี้วิธีการทางสถิติถูกนำมาใช้ให้เกิดประโยชน์ในงานด้านต่างๆเป็นอย่างมาก การวิเคราะห์การถดถอย (Regression Analysis) เป็นวิธีการทางสถิติวิธีหนึ่งที่ได้รับความนิยมอย่างแพร่หลาย ซึ่งวิธีการวิเคราะห์การถดถอยเชิงเส้นพหุ (Multiple Linear Regression Analysis) เป็นการศึกษาความสัมพันธ์ระหว่างตัวแปรตามกับตัวแปรอิสระที่มีจำนวนมากกว่า 1 ตัวแปร โดยตัวแปรอิสระและตัวแปรตามมีความสัมพันธ์ในรูปเชิงเส้น แต่การใช้ตัวแปรอิสระมากเกินไปอาจทำให้ค่าพยากรณ์ที่ได้มีความคลาดเคลื่อนสูงและเกิดปัญหาตัวแปรอิสระบางตัวมีพหุสัมพันธ์เชิงเส้น (Multicollinearity) ต่อกัน ทำให้การประมาณค่าสัมประสิทธิ์การถดถอยที่ได้มีค่าไม่เสถียรและมีค่าความคลาดเคลื่อนสูง วิธีการแก้ปัญหาที่นิยมใช้คือวิธีวิเคราะห์การถดถอยแบบริดจ์ (Ridge Regression) โดยไม่จำเป็นต้องตัดตัวแปรที่มีความสัมพันธ์กันสูงออกจากตัวแบบ หลักการของวิธีนี้คือ การทำให้ค่าสัมประสิทธิ์การถดถอยพหุมีค่าความคลาดเคลื่อนกำลังสองเฉลี่ยต่ำลง เนื่องจากค่าความคลาดเคลื่อนกำลังสองเฉลี่ยของวิธีกำลังสองน้อยที่สุดเป็นฟังก์ชันของ $(\mathbf{X'X})^{-1}$ ดังนั้นการลดค่าความคลาดเคลื่อนกำลังสองเฉลี่ยจึงต้องพยายามลดค่า $(\mathbf{X'X})^{-1}$ ให้ต่ำลงซึ่งจะทำได้โดยการบวกค่าคงตัวที่มากกว่าศูนย์กับสมาชิกทุกตัวบนเส้นทแยงมุม จะได้ตัวประมาณค่าสัมประสิทธิ์การถดถอยพหุแบบริดจ์ $\hat{\boldsymbol{\beta}}_r = (\mathbf{X'X} + r\mathbf{I_p})^{-1}\mathbf{X'y}$ เมื่อ $r \geq 0$ ตัวประมาณค่ามีสมบัติที่มีความเอนเอียง (bias) และในการวิเคราะห์จะต้องมีการประมาณค่าพารามิเตอร์ r ซึ่งมีผู้ศึกษาวิธีการประมาณค่าพารามิเตอร์ r หลายวิธีเช่นวิธีของ Hoerl, Kennard and Baldwin (HKB) (1975) วิธีของ McDonald and Galarneau (MC & D) (1975) และวิธีของ Khalaf and Shukur (2005)เป็นต้น และมีผู้สนใจศึกษาวิจัยเกี่ยวกับวิธีการถดถอยแบบริดจ์และวิธีการประมาณค่าพารามิเตอร์ r อีกมาก เช่น Dean W. Wichernและ Gilbert A. Churchill (1978: 301-311)ได้ศึกษาเปรียบเทียบตัวประมาณแบบริดจ์ด้วยวิธีของ Hoerl and Kennard (1970a); Hoerl, Kennard and Baldwin (1975); McDonald and Galarneau (1975); Lawless and Wang (1976) และ KhuriandMyers (1979) นุสรา สถิตโพธิ์ศรี (2535) เสนอการเปรียบเทียบตัวประมาณค่าสัมประสิทธิ์การถดถอยพหุในกรณีที่เกิดพหุสัมพันธ์

ระหว่างตัวแปรอิสระโดยวิธีการถดถอยแบบริดจ์ และวิธีการถดถอยลาเท้นรูทธ์ และ ธันยากร ต้นชลขันธ์ (2538) เสนอการเปรียบเทียบการประมาณค่าสัมประสิทธิ์การถดถอยพหุโดยใช้วิธีกำลังสองน้อยที่สุด วิธีการถดถอยแบบริดจ์ และวิธีที่ใช้หลักการของริดจ์และสไตน์ ในกรณีเกิดพหุสัมพันธ์ระหว่างตัวแปรอิสระ เป็นต้น(ในทุกกรณีที่ทำการศึกษา การจำลองตัวแปรอิสระในตัวแบบมีแต่เฉพาะตัวแปรอิสระที่เกี่ยวข้องกับตัวแปรตามเท่านั้น)

ในการศึกษาครั้งนี้ได้มีการแบ่งตัวแปรอิสระออกเป็น 2 กลุ่มคือตัวแปรอิสระที่เกี่ยวข้องกับตัวแปรตามและตัวแปรอิสระที่ไม่เกี่ยวข้องกับตัวแปรตาม ตัวแปรอิสระที่อยู่ในกลุ่มของตัวแปรที่เกี่ยวข้องจะเป็นตัวแปรอิสระที่มีทฤษฎีในสาขาวิชาที่เกี่ยวข้องหรือการศึกษาในอดีตสนับสนุนว่าตัวแปรอิสระเหล่านี้มีความสำคัญต่อตัวแปรตาม สำหรับตัวแปรที่อยู่ในกลุ่มตัวแปรที่ไม่เกี่ยวข้องเป็นตัวแปรที่ผู้วิจัยมีความไม่แน่ใจว่าสมควรอยู่ในตัวแบบเพื่ออธิบายตัวแปรตามหรือไม่ เนื่องจากไม่มีทฤษฎีในสาขาวิชาที่เกี่ยวข้องหรือการศึกษาในอดีตสนับสนุนเหมือนตัวแปรอิสระในกลุ่มแรก จึงต้องอาศัยวิธีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบ ซึ่งวิธีที่นิยมใช้กันอย่างแพร่หลายคือ วิธีพิจารณาทุกตัวแบบที่เป็นไปได้ (All Possible Regression) วิธีเลือกตัวแปรอิสระแบบไปข้างหน้า (Forward Selection) วิธีตัดตัวแปรอิสระออกแบบถอยหลัง (Backward Elimination) วิธีการถดถอยแบบขั้นตอน (Stepwise Regression) (Montgomery, Peck and Vining, 2006: 270) วิธีการปรับปรุงค่า R^2 สูงสุด (MAXR: Maximum R^2 Improvement) วิธีการปรับปรุงค่า R^2 ต่ำสุด (MINR: Minimum R^2 Improvement) ซึ่งวิธีการคัดเลือกเหล่านี้เป็นวิธีการในโปรแกรมสำเร็จรูปทางสถิติ SAS (Games and Lerner, 1981) และในโปรแกรมสำเร็จรูปทางสถิติทั่วไปนอกจากนี้ยังมีวิธีการหาค่าอุตตมะ(Optimum)โดยการจัดกลุ่ม (Combinatorial) ที่สามารถใช้คัดเลือกตัวแปรอิสระที่เหมาะสมได้เช่นกัน เช่น วิธีการค้นหาแบบต้องห้าม (Tabu Search) เป็นกระบวนการสำหรับแก้ปัญหาเพื่อหาค่าเหมาะสมที่สุด โดยใช้ข้อจำกัดและเวลาในการประมวลผลน้อยที่สุด (Glover, 1990: 74) เกณฑ์ที่ใช้หยุดกระบวนการทำงาน คือจำนวนรอบสูงสุดและการลดลงของค่าฟังก์ชันเป้าหมาย (ObjectiveFunction) วิธีนี้เป็นวิธีที่ใช้ระยะเวลาน้อยในการคัดเลือกตัวแปรเข้าสู่ตัวแบบ และได้ตัวแบบที่มีความคลาดเคลื่อนในการพยากรณ์ต่ำ หรือกล่าวได้ว่าเป็นวิธีที่ช่วยในการค้นหาตัวแปรอิสระที่เกี่ยวข้องกับตัวแปรตามที่จำเป็นในตัวแบบเพื่อลดปัญหาการมีจำนวนตัวแปรอิสระในตัวแบบมากเกินไป (Over Specification) และน้อยเกินไป (Under Specification) วิธีการค้นหาแบบต้องห้าม มีผู้สนใจนำมาใช้ในการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบ เช่น Drezner and George (1999: 349-367) ได้เสนอวิธีการคัดเลือกตัวแปรอิสระในการวิเคราะห์การถดถอยเชิงเส้นพหุโดยวิธีการค้นหาแบบต้องห้าม เปรียบเทียบกับวิธีการถดถอยแบบขั้นตอนและวิธีการปรับปรุงค่า R^2 สูงสุด บุษยา ปภาพจน์ (2548) ได้ทำการศึกษาเปรียบเทียบวิธีการคัดเลือกตัวแบบที่เหมาะสมที่สุด

ในการวิเคราะห์การถดถอยเชิงเส้นพหุโดยเปรียบเทียบการคัดเลือกตัวแบบ 4 วิธีคือ วิธีการค้นหาแบบต้องห้าม วิธีการถดถอยแบบขั้นตอน วิธีเลือกตัวแปรอิสระแบบไปข้างหน้า และวิธีการปรับปรุงค่า R^2 สูงสุดโดยทั้ง Drezner and George (1999)และบุษยาปภาพจน์ (2548) ศึกษาเฉพาะกรณีที่ไม่มีสหสัมพันธ์เชิงเส้นพหุระหว่างตัวแปรอิสระ และกานต์ณัฐ ณ บางช้าง (2554) ได้ทำการศึกษาเปรียบเทียบระหว่างการคัดเลือกตัวแปรอิสระโดยใช้วิธีการค้นหาแบบต้องห้ามที่ใช้ฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ย (MSE) และค่าเฉลี่ยความคลาดเคลื่อนสัมบูรณ์ (MAE) กับวิธีการถดถอยแบบขั้นตอนที่ประมาณค่าพารามิเตอร์ด้วยวิธีกำลังสองน้อยที่สุด โดยแบ่งตัวแปรอิสระที่ทำการคัดเลือกแบ่งออกเป็น 2 กลุ่มคือ ตัวแปรที่เกี่ยวข้องและไม่เกี่ยวข้องกับตัวแปรตาม การประมาณค่าพารามิเตอร์ใช้วิธีกำลังสองน้อยที่สุดในกรณีที่มีและไม่มีสหสัมพันธ์เชิงเส้นพหุระหว่างตัวแปรอิสระภายในกลุ่มของตัวแปรอิสระที่มีความเกี่ยวข้องกับตัวแปรตาม ซึ่งค่าสัมประสิทธิ์สหสัมพันธ์ระหว่างตัวแปรอิสระมีค่าระหว่าง 0.95-0.99

ในการศึกษาครั้งนี้ได้เปรียบเทียบการคัดเลือกตัวแปรอิสระของตัวแบบการถดถอยเชิงเส้นพหุ ในกรณีที่ตัวแปรอิสระมี2กลุ่มคือ ตัวแปรอิสระที่เกี่ยวข้องและไม่เกี่ยวข้องกับตัวแปรตาม ในกลุ่มของตัวแปรอิสระที่มีความเกี่ยวข้องกันกับตัวแปรตามมีสหสัมพันธ์เชิงเส้นพหุกันสูงโดยใช้วิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยมีค่าต่ำสุด (Mean Squares Error:MSE) และค่าความคลาดเคลื่อนกำลังสองเฉลี่ยปรับด้วยฟังก์ชันการลงโทษ (Penalty Function: MSE $+\dfrac{c\hat{\boldsymbol{\beta}}'\hat{\boldsymbol{\beta}}}{n-p}$) เพื่อเปรียบเทียบกับการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบด้วยวิธีการถดถอยแบบขั้นตอนที่มีการประมาณค่าสัมประสิทธิ์การถดถอยด้วยวิธีกำลังสองน้อยที่สุด (Ordinary Least Square Method: OLS) กับวิธีการถดถอยแบบริดจ์ (Ridge Regression)ที่ใช้วิธีการประมาณค่าคงตัว r 4 วิธีคือวิธีโฮเอิร์ล เคนนาร์ด และ บาลด์วิน (Hoerl, Kennard and Baldwin: r_{HKB}) วิธีลอว์เลสและแวง (Lawless and Wang: r_{LW}) วิธีนอมูระ (Nomura: r_{HMO})และวิธีคาลาฟและชูเกอร์ (Khalaf and Shukur: r_{KS})

1.2 วัตถุประสงค์ของการวิจัย

1)เพื่อศึกษาวิธีการประมาณค่าพารามิเตอร์ของตัวแบบการถดถอยเชิงเส้นพหุโดยวิธีกำลังสองน้อยที่สุดและวิธีการถดถอยแบบริดจ์

2) เพื่อเปรียบเทียบความถูกต้องในการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบการถดถอยเชิงเส้นพหุที่มีสหสัมพันธ์เชิงเส้นพหุกันสูงโดยใช้วิธีการค้นหาแบบต้องห้าม (Tabu Search) ที่มี

ฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยและค่าความคลาดเคลื่อนกำลังสองเฉลี่ยปรับด้วยฟังก์ชันการลงโทษกับวิธีการถดถอยแบบขั้นตอนที่มีการประมาณค่าพารามิเตอร์แบบกำลังสองน้อยที่สุดและแบบริดจ์

1.3 ขอบเขตการวิจัย

ตัวแบบที่ใช้ในการศึกษาเป็นตัวแบบการถดถอยเชิงเส้นพหุ ที่ความคลาดเคลื่อนมีการแจกแจงปกติมีค่าเฉลี่ยเท่ากับ 0 และความแปรปรวนคงที่เท่ากับ σ^2 ตัวแปรอิสระมีการแจกแจงเอกรูป k ตัว ซึ่งแยกเป็น 2 กลุ่มคือตัวแปรอิสระที่เกี่ยวข้องกับตัวแปรตามจำนวน q ตัว ตัวแปรเหล่านี้มีทฤษฎีบทหรือผลการศึกษาในอดีตสนับสนุนว่ามีความสำคัญกับตัวแปรตามและมีตัวแปรอิสระ 1 คู่ในกลุ่มนี้มีความสัมพันธ์กันสูงในระดับ 0.999-0.9999 ตัวแปรอิสระอีกกลุ่มหนึ่งที่ไม่มีทฤษฎีบทหรือผลการศึกษาในอดีตสนับสนุนและผู้วิจัยไม่แน่ใจว่าตัวแปรเหล่านี้ควรอยู่ในตัวแบบหรือไม่มีจำนวน $k-q$ ตัวมีตัวแบบดังนี้

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon} \tag{1.1}$$

โดยที่ $\mathbf{y}$ เป็น เวกเตอร์ค่าสังเกตของตัวแปรตาม $\mathbf{y}$ ขนาด $n \times 1$

$\mathbf{X}$ เป็น เมตริกซ์ของตัวแปรอิสระขนาด $n \times (k+1)$

$\boldsymbol{\beta}$ เป็น เวกเตอร์ของพารามิเตอร์ของตัวแบบขนาด $(k+1) \times 1$

$\boldsymbol{\varepsilon}$ เป็น เวกเตอร์ของความคลาดเคลื่อนสุ่ม $n \times 1$

ฐานคติของการถดถอยเชิงเส้นพหุความคลาดเคลื่อนสุ่มมีข้อกำหนดดังนี้ $\boldsymbol{\varepsilon} \sim N_n(\mathbf{0}, \sigma^2\mathbf{I})$

เมื่อ $\mathbf{I}$ เป็นเมตริกซ์เอกลักษณ์ขนาด $n \times n$

n เป็นขนาดตัวอย่าง

k เป็นจำนวนตัวแปรอิสระในตัวแบบ

วิธีการคัดเลือกตัวแปรอิสระที่ใช้คือวิธีการค้นหาแบบต้องห้าม และวิธีการคัดเลือกตัวแปรอิสระแบบขั้นตอน วิธีการค้นหาแบบต้องห้ามการประมาณค่าพารามิเตอร์ของตัวแบบกำหนดฟังก์ชันเป้าหมาย 2 ฟังก์ชัน คือ ฟังก์ชันค่าความคลาดเคลื่อนกำลังสองเฉลี่ย (F_1)ตามที่แสดงในสมการ (1.2) และฟังก์ชันค่าความคลาดเคลื่อนกำลังสองเฉลี่ยปรับด้วยฟังก์ชันการลงโทษ (F_2) แสดงไว้ในสมการที่ (1.3)ค่าความคลาดเคลื่อนกำลังสองเฉลี่ยคือค่าผลรวมของค่าจริงลบด้วยค่าพยากรณ์ยกกำลังสองหารด้วยองศาเสรีนั่นเอง

$$F_1 = \frac{(\mathbf{y} - \hat{\mathbf{y}})'(\mathbf{y} - \hat{\mathbf{y}})}{n-k-1} \tag{1.2}$$

$$F_2 = \frac{(\mathbf{y}-\hat{\mathbf{y}})'(\mathbf{y}-\hat{\mathbf{y}})}{n-k-1} + \frac{c\hat{\boldsymbol{\beta}}'\hat{\boldsymbol{\beta}}}{n-p} \quad (1.3)$$

เมื่อ $\mathbf{y}$ เป็นเวกเตอร์ค่าสังเกต

$\hat{\mathbf{y}}$ เป็นเวกเตอร์ค่าพยากรณ์ที่ได้มาจากสมการการถดถอยที่แทนสัมประสิทธิ์การถดถอยที่ประมาณได้จากวิธีการค้นหาแบบต้องห้าม

$\hat{\boldsymbol{\beta}}$ เป็นเวกเตอร์ของค่าสัมประสิทธิ์การถดถอยที่ประมาณจากวิธีการค้นหาแบบต้องห้าม

c เป็นค่าประมาณด้วยวิธีการค้นหาแบบต้องห้ามที่ทำให้ค่าฟังก์ชันความคลาดเคลื่อนกำลังสองเฉลี่ยปรับด้วยฟังก์ชันการลงโทษมีค่าต่ำสุด โดยกำหนดพิสัย $c = (0,1)$

n เป็นจำนวนค่าสังเกต

$p = k+1$ เป็นจำนวนพารามิเตอร์ที่ต้องประมาณค่าในตัวแบบ

การคัดเลือกตัวแปรอิสระโดยใช้วิธีการค้นหาแบบต้องห้าม ใช้วิธีประมาณค่าพารามิเตอร์แบบต้องห้ามและค้นหาตัวแบบเพื่อทำให้ฟังก์ชัน F_1 และ F_2 มีค่าต่ำสุด

สำหรับการคัดเลือกตัวแปรอิสระแบบขั้นตอนใช้วิธีการประมาณค่าพารามิเตอร์ของตัวแบบ 2 วิธี คือวิธีประมาณค่ากำลังสองน้อยที่สุดและวิธีการประมาณค่าพารามิเตอร์แบบริดจ์ การประมาณค่าพารามิเตอร์แบบริดจ์ใช้วิธีการประมาณค่าคงตัว r 4 วิธีคือ วิธี โฮเอิร์ล เคนนาร์ด และบาลด์วิน (Hoerl, Kennard and Baldwin: r_{HKB}) วิธีลอว์เลสและแวง (Lawless and Wang: r_{LW}) วิธีนอมูระ (Nomura: r_{HMO}) และวิธีคาลาฟและชูเกอร์ (Khalaf and Shukur: r_{KS}) โดยกำหนดระดับนัยสำคัญในการเลือกตัวแปรอิสระเข้าและออกจากตัวแบบเป็น 0.05

เกณฑ์การตัดสินใจในการเปรียบเทียบการคัดเลือกตัวแปรอิสระ คือร้อยละของจำนวนครั้งที่แต่ละวิธีสามารถคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบได้ตามตัวแบบจริง (CORRECT, %), ตัวแบบ Overspecification (OVER, %), ตัวแบบUnderspecification (UNDER, %) และตัวแบบ Misspecification (MISS, %) โดยคำนวณจากจำนวนครั้งที่คัดเลือกตัวแปรอิสระได้ถูกต้องตามตัวแบบจริง (ตัวแบบ Overspecification, ตัวแบบ Underspecificationหรือ ตัวแบบ Misspecification) หารด้วยจำนวนครั้งในการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบทั้งหมดคูณด้วย 100

กำหนดให้จำนวนทำซ้ำในแต่ละสถานการณ์เท่ากับ m

1.4 นิยามคำศัพท์ที่เกี่ยวข้อง

ตัวแบบจริง (True Model)หมายถึง ตัวแบบการถดถอยเชิงเส้นพหุที่มีตัวแปรอิสระที่เกี่ยวข้องกับตัวแปรตามครบทุกตัวและความคลาดเคลื่อนเชิงสุ่มที่กำหนดขึ้น

ตัวแบบเต็มรูป (Full Model)หมายถึง ตัวแบบการถดถอยเชิงเส้นพหุที่ประกอบด้วยตัวแปรอิสระจำนวนหนึ่งที่เกี่ยวข้องกับตัวแปรตามโดยมีทฤษฎีที่เกี่ยวข้องสนับสนุนและหรือการศึกษาในอดีตที่ผ่านมาและตัวแปรที่ไม่เกี่ยวข้องกับตัวแปรตามจำนวนหนึ่ง

ตัวแบบ Misspecificationหมายถึง ตัวแบบที่มีตัวแปรอิสระบางตัวแปรที่เกี่ยวข้องกับตัวแปรตามขาดหายไปจากตัวแบบและมีตัวแปรอิสระบางตัวแปรที่ไม่เกี่ยวข้องกับตัวแปรตามรวมอยู่ในตัวแบบ

ตัวแบบOverspecificationหมายถึง ตัวแบบที่ประกอบด้วยตัวแปรอิสระที่เกี่ยวข้องกับตัวแปรตามครบทุกตัว และยังมีตัวแปรอิสระบางตัวที่ไม่เกี่ยวข้องกับตัวแปรตามรวมอยู่ในตัวแบบ

ตัวแบบUnderspecificationหมายถึง ตัวแบบที่มีตัวแปรอิสระบางตัวแปรที่เกี่ยวข้องกับตัวแปรตามขาดหายไปจากตัวแบบ

1.5 ประโยชน์ที่ได้รับจากการวิจัย

1) เป็นประโยชน์ในการนำไปใช้คัดเลือกตัวแปรอิสระในกรณีที่ตัวแปรอิสระมีความสัมพันธ์ต่อกันสูง

2)เป็นแนวทางในการศึกษาและคัดเลือกตัวแปรในตัวแบบการถดถอยเชิงเส้นพหุที่เหมาะสม ด้วยเทคนิคอื่นๆต่อไป

บทที่ 2

ทฤษฎีและผลงานวิจัยที่เกี่ยวข้อง

ในบทนี้แบ่งเป็น 2 หัวข้อ ในหัวข้อ 2.1 กล่าวถึงทฤษฎีที่เกี่ยวข้อง หัวข้อ 2.2 ผลงานวิจัยที่เกี่ยวข้องกับการประมาณค่าพารามิเตอร์การถดถอยพหุ โดยวิธีกำลังสองน้อยที่สุด วิธีการถดถอยแบบริดจ์ และการคัดเลือกตัวแบบการถดถอยพหุ

2.1 ทฤษฎีที่เกี่ยวข้อง

ในหัวข้อนี้จะแบ่งออกเป็น 5 หัวข้อย่อย ได้แก่ ตัวแบบ ฐานคติและทฤษฎีที่เกี่ยวข้อง, การประมาณค่าคงตัว r ในการถดถอยแบบริดจ์ วิธีการคัดเลือกตัวแปรอิสระ การทดสอบสำหรับการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบการถดถอย และการค้นหาแบบต้องห้าม

2.1.1 ตัวแบบ ฐานคติ และทฤษฎีที่เกี่ยวข้อง

ตัวแบบและฐานคติของการถดถอยเชิงเส้นพหุ (Multiple Linear Regression Model)

ตัวแบบการถดถอยเชิงเส้นพหุ $$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon} \tag{1.1}$$

โดยที่ $\mathbf{y}$ เป็น เวกเตอร์ค่าสังเกตของตัวแปรตาม $\mathbf{y}$ ขนาด $n \times 1$

$\mathbf{X}$ เป็น เมตริกซ์ของตัวแปรอิสระขนาด $n \times (k+1)$

$\boldsymbol{\beta}$ เป็น เวกเตอร์ของพารามิเตอร์ของตัวแบบขนาด $(k+1) \times 1$

$\boldsymbol{\varepsilon}$ เป็น เวกเตอร์ของความคลาดเคลื่อนสุ่ม $n \times 1$

ฐานคติของการถดถอยเชิงเส้นพหุความคลาดเคลื่อนสุ่มมีข้อกำหนดดังนี้ $\boldsymbol{\varepsilon} \sim N_n(\mathbf{0}, \sigma^2\mathbf{I})$

เมื่อ $\mathbf{I}$ เป็นเมตริกซ์เอกลักษณ์ขนาด $n \times n$

n เป็นขนาดตัวอย่าง

k เป็นจำนวนตัวแปรอิสระในตัวแบบ

ทฤษฎีที่ 2.1 ตัวแบบการถดถอยเชิงเส้นพหุ $\mathbf{y} = \mathbf{X\beta} + \boldsymbol{\varepsilon}$ ภายใต้ฐานคติที่ $\boldsymbol{\varepsilon}$ มีการแจกแจงแบบปกติมีค่าเฉลี่ยเท่ากับ 0 และค่าความแปรปรวนเท่ากับ $\sigma^2\mathbf{I}$ จะได้ตัวประมาณกำลังสองน้อยที่สุดของ $\boldsymbol{\beta}$ มีค่าเท่ากับ $(\mathbf{X'X})^{-1}\mathbf{X'y}$ และมีสมบัติเป็นตัวประมาณค่าที่ไม่เอนเอียง มีความแปรปรวนเท่ากับ $\sigma^2(\mathbf{X'X})^{-1}$

พิสูจน์ ตัวประมาณกำลังสองน้อยที่สุดคือตัวประมาณค่าที่ทำให้ผลรวมของความคลาดเคลื่อนกำลังสองมีค่าน้อยที่สุด

จาก $\mathbf{y} = \mathbf{X\beta} + \boldsymbol{\varepsilon}$

$$\boldsymbol{\varepsilon} = \mathbf{y} - \mathbf{X\beta}$$

$$\begin{aligned}
\boldsymbol{\varepsilon}'\boldsymbol{\varepsilon} &= (\mathbf{y} - \mathbf{X\beta})'(\mathbf{y} - \mathbf{X\beta}) \\
&= \mathbf{y'y} - \boldsymbol{\beta}'\mathbf{X'y} - \mathbf{y'X\beta} + \boldsymbol{\beta}'\mathbf{X'X\beta} \\
&= \mathbf{y'y} - \mathbf{2y'X\beta} + \boldsymbol{\beta}'\mathbf{X'X\beta}
\end{aligned}$$

$$\left.\frac{\partial(\boldsymbol{\varepsilon}'\boldsymbol{\varepsilon})}{\partial(\boldsymbol{\beta})}\right|_{\beta=\hat{\beta}} = -2\mathbf{X'y} + 2\mathbf{X'X}\hat{\boldsymbol{\beta}} = 0$$

นั่นคือ $\mathbf{X'X}\hat{\boldsymbol{\beta}} = \mathbf{X'y}$

$$\hat{\boldsymbol{\beta}} = (\mathbf{X'X})^{-1}\mathbf{X'y} \tag{2.1}$$

สมบัติของตัวประมาณค่ากำลังสองน้อยที่สุด

$$\begin{aligned}
E(\hat{\boldsymbol{\beta}}) &= E[(\mathbf{X'X})^{-1}\mathbf{X'y}] \\
&= E[(\mathbf{X'X})^{-1}\mathbf{X'}(\mathbf{X\beta} + \boldsymbol{\varepsilon})] \\
&= E[(\mathbf{X'X})^{-1}\mathbf{X'X\beta} + (\mathbf{X'X})^{-1}\mathbf{X'}\boldsymbol{\varepsilon}] \\
&= \boldsymbol{\beta}
\end{aligned}$$

เนื่องจาก $E(\boldsymbol{\varepsilon}) = \mathbf{0}$ และ $(\mathbf{X'X})^{-1}\mathbf{X'X} = \mathbf{I}$ ดังนั้น $\hat{\boldsymbol{\beta}}$ เป็นตัวประมาณไม่เอนเอียงของ $\boldsymbol{\beta}$

$$\begin{aligned}
Var(\hat{\boldsymbol{\beta}}) &= Var[(\mathbf{X'X})^{-1}\mathbf{X'y}] \\
&= (\mathbf{X'X})^{-1}\mathbf{X'}Var(\mathbf{y})[(\mathbf{X'X})^{-1}\mathbf{X'}]' \\
&= \sigma^2(\mathbf{X'X})^{-1}\mathbf{X'X}(\mathbf{X'X})^{-1} \\
&= \sigma^2(\mathbf{X'X})^{-1}
\end{aligned} \tag{2.2}$$

ทฤษฎีที่ 2.2 ถ้า $\hat{\boldsymbol{\beta}}_{\mathbf{r}}$ เป็นตัวประมาณค่าการถดถอยแบบริดจ์ที่มีตัวแบบการถดถอยเชิงเส้นพหุ $\mathbf{y} = \mathbf{X\beta} + \boldsymbol{\varepsilon}$ ภายใต้ฐานคติของ $\boldsymbol{\varepsilon}$ มีการแจกแจงแบบปกติมีค่าเฉลี่ยเท่ากับ 0 และมีความแปรปรวนเท่ากับ $\sigma^2\mathbf{I}$ ตัวประมาณค่า $\hat{\boldsymbol{\beta}}_{\mathbf{r}} = (\mathbf{X'X} + r\mathbf{I})^{-1}\mathbf{X'y}$ เมื่อ $r > 0$ เป็นตัวประมาณค่าที่เอนเอียงของ $\boldsymbol{\beta}$ ความแปรปรวนของ $\hat{\boldsymbol{\beta}}$ มีค่าเท่ากับ $\sigma^2(\mathbf{X'X} + r\mathbf{I})^{-1}\mathbf{X'X}(\mathbf{X'X} + r\mathbf{I})^{-1}$ และค่าความคลาดเคลื่อน

กำลังสองเฉลี่ยมีค่าเท่ากับ $\sigma^2 \sum_{j=1}^{p} \frac{\lambda_j}{(\lambda_j + r)^2} + r^2 \boldsymbol{\beta}'(\mathbf{X'X} + r\mathbf{I})^{-2}\boldsymbol{\beta}$ เมื่อ λ_j เป็นค่าเฉพาะ(Eigenvalue)ที่ j ของเมตริกซ์ $\mathbf{X'X};\ j = 1, 2,..., p$ (Montgomery, Peck and Vining, 2006: 311-312)

พิสูจน์ $(\mathbf{X'X} + r\mathbf{I})\hat{\boldsymbol{\beta}}_\mathbf{r} = \mathbf{X'y}$

$$\hat{\boldsymbol{\beta}}_\mathbf{r} = (\mathbf{X'X} + r\mathbf{I})^{-1}\mathbf{X'y} \text{ เมื่อ } r > 0 \tag{2.3}$$

หรือ $\hat{\boldsymbol{\beta}}_\mathbf{r} = (\mathbf{X'X} + r\mathbf{I})^{-1}(\mathbf{X'X})\hat{\boldsymbol{\beta}}$

ดังนั้น

$$\begin{aligned} E(\hat{\boldsymbol{\beta}}_\mathbf{r}) &= (\mathbf{X'X} + r\mathbf{I})^{-1}\mathbf{X'X}E(\hat{\boldsymbol{\beta}}) \\ &= (\mathbf{X'X} + r\mathbf{I})^{-1}\mathbf{X'X}\hat{\boldsymbol{\beta}} \\ &= \mathbf{Z_r}\hat{\boldsymbol{\beta}}; \end{aligned}$$

เมื่อให้ $\mathbf{Z_r} = (\mathbf{X'X} + r\mathbf{I})^{-1}\mathbf{X'X}$

ดังนั้น $\hat{\boldsymbol{\beta}}_\mathbf{r}$ เป็นตัวประมาณที่เอนเอียงของ $\boldsymbol{\beta}$

$$\begin{aligned} Bias(\hat{\boldsymbol{\beta}}_\mathbf{r}) &= E(\hat{\boldsymbol{\beta}}_\mathbf{r}) - \boldsymbol{\beta} \qquad \text{(Montgomery, Peck and Vining, 2006: 311-312)} \\ &= -r(\mathbf{X'X} + r\mathbf{I})^{-1}\boldsymbol{\beta} \end{aligned}$$

$$\begin{aligned} Var(\hat{\boldsymbol{\beta}}_\mathbf{r}) &= Var[(\mathbf{X'X} + r\mathbf{I})^{-1}\mathbf{X'y}] \\ &= (\mathbf{X'X} + r\mathbf{I})^{-1}\mathbf{X'}Var(\mathbf{y})[(\mathbf{X'X} + r\mathbf{I})^{-1}\mathbf{X'}]' \\ &= \sigma^2(\mathbf{X'X} + r\mathbf{I})^{-1}\mathbf{X'X}(\mathbf{X'X} + r\mathbf{I})^{-1} \end{aligned} \tag{2.4}$$

$$\begin{aligned} MSE(\hat{\boldsymbol{\beta}}_\mathbf{r}) &= E(\hat{\boldsymbol{\beta}}_\mathbf{r} - \boldsymbol{\beta})^2 \\ &= Var(\hat{\boldsymbol{\beta}}_\mathbf{r}) + [E(\hat{\boldsymbol{\beta}}_\mathbf{r}) - \boldsymbol{\beta}]^2 \\ &= Var(\hat{\boldsymbol{\beta}}_\mathbf{r}) + (Bias(\hat{\boldsymbol{\beta}}_\mathbf{r}))^2 \\ &= \sigma^2 tr[(\mathbf{X'X} + r\mathbf{I})^{-1}\mathbf{X'X}(\mathbf{X'X} + r\mathbf{I})^{-1}] \\ &\quad + r^2\boldsymbol{\beta}'(\mathbf{X'X} + r\mathbf{I})^{-2}\boldsymbol{\beta} \\ &= \sigma^2 tr(\mathbf{X'X}(\mathbf{X'X} + r\mathbf{I})^{-2}) + r^2\boldsymbol{\beta}'(\mathbf{X'X} + r\mathbf{I})^{-2}\boldsymbol{\beta} \\ &= \sigma^2 \sum_{j=1}^{p} \frac{\lambda_j}{(\lambda_j + r)^2} + r^2\boldsymbol{\beta}'(\mathbf{X'X} + r\mathbf{I})^{-2}\boldsymbol{\beta} \end{aligned} \tag{2.5}$$

เมื่อ λ_j เป็นค่าเฉพาะที่ j ของเมตริกซ์ $\mathbf{X'X};\ j = 1, 2,..., p$

ทฤษฎีที่ 2.3 ถ้า $\hat{\boldsymbol{\beta}}_\mathbf{r}$ เป็นตัวประมาณค่าการถดถอยแบบริดจ์ที่มีคุณสมบัติตามทฤษฎีที่ 2.2 แล้วค่าความคลาดเคลื่อนยกกำลังสองของการถดถอยแบบริดจ์ (Sum of Squares Error) มีค่าเท่ากับ $SSE_r = SSE + (\hat{\boldsymbol{\beta}}_\mathbf{r} - \hat{\boldsymbol{\beta}})'\mathbf{X'X}(\hat{\boldsymbol{\beta}}_\mathbf{r} - \hat{\boldsymbol{\beta}})$ เมื่อ SSE เป็นค่าความคลาดเคลื่อนยกกำลังสองที่ได้จากการประมาณค่าพารามิเตอร์ของตัวแบบโดยวิธีกำลังสองน้อยที่สุด นั่นคือ $SSE = (\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}})'(\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}})$

พิสูจน์ $SSE_r = \mathbf{e_r}'\mathbf{e_r}$

$$\begin{aligned}
SSE_r &= (\mathbf{y}-\mathbf{X}\hat{\boldsymbol{\beta}}_\mathbf{r})'(\mathbf{y}-\mathbf{X}\hat{\boldsymbol{\beta}}_\mathbf{r}) \\
&= \mathbf{y'y} - \mathbf{y'X}(\mathbf{X'X}+r\mathbf{I})^{-1}\mathbf{X'X}\hat{\boldsymbol{\beta}} - \hat{\boldsymbol{\beta}}'\mathbf{X'X}(\mathbf{X'X}+r\mathbf{I})^{-1}\mathbf{X'y} \\
&\quad + \hat{\boldsymbol{\beta}}'\mathbf{X'X}(\mathbf{X'X}+r\mathbf{I})^{-1}\mathbf{X'X}(\mathbf{X'X}+r\mathbf{I})^{-1}\mathbf{X'X}\hat{\boldsymbol{\beta}} \\
&= \mathbf{y'y} - \mathbf{y'X}\hat{\boldsymbol{\beta}}_\mathbf{r} - \hat{\boldsymbol{\beta}}_\mathbf{r}'\mathbf{X'y} + \hat{\boldsymbol{\beta}}_\mathbf{r}\mathbf{X'X}\hat{\boldsymbol{\beta}}_\mathbf{r} \\
&= \mathbf{y'y} - \mathbf{y'X}\hat{\boldsymbol{\beta}} - \hat{\boldsymbol{\beta}}'\mathbf{X'y} + \hat{\boldsymbol{\beta}}'\mathbf{X'X}\hat{\boldsymbol{\beta}} - \hat{\boldsymbol{\beta}}'\mathbf{X'X}\hat{\boldsymbol{\beta}}_\mathbf{r} - \hat{\boldsymbol{\beta}}_\mathbf{r}'\mathbf{X'X}\hat{\boldsymbol{\beta}} \\
&\quad + \hat{\boldsymbol{\beta}}_\mathbf{r}'\mathbf{X'X}\hat{\boldsymbol{\beta}}_\mathbf{r} + \hat{\boldsymbol{\beta}}'\mathbf{X'X}\hat{\boldsymbol{\beta}} \\
&= (\mathbf{y}-\mathbf{X}\hat{\boldsymbol{\beta}})'(\mathbf{y}-\mathbf{X}\hat{\boldsymbol{\beta}}) + (\hat{\boldsymbol{\beta}}_\mathbf{r}-\hat{\boldsymbol{\beta}})'\mathbf{X'X}(\hat{\boldsymbol{\beta}}_\mathbf{r}-\hat{\boldsymbol{\beta}}) \\
&= SSE + (\hat{\boldsymbol{\beta}}_\mathbf{r}-\hat{\boldsymbol{\beta}})'\mathbf{X'X}(\hat{\boldsymbol{\beta}}_\mathbf{r}-\hat{\boldsymbol{\beta}}) \qquad (2.6)
\end{aligned}$$

ทฤษฎีที่ 2.4 ถ้า $\hat{\boldsymbol{\beta}}$ เป็นตัวประมาณค่ากำลังสองน้อยที่สุดของ $\boldsymbol{\beta}$ และ $\hat{\boldsymbol{\beta}}_\mathbf{r}$ เป็นตัวประมาณค่าการถดถอยแบบริดจ์ของ $\boldsymbol{\beta}$ แล้วจะได้ว่า $MSE(\hat{\boldsymbol{\beta}}_\mathbf{r}) < MSE(\hat{\boldsymbol{\beta}})$ เสมอเมื่อค่า $r > 0$

พิสูจน์ ในการประมาณค่าสัมประสิทธิ์การถดถอยพหุคูณ $\boldsymbol{\beta}$ ด้วยวิธีกำลังสองน้อยที่สุด จะได้ตัวประมาณ $\boldsymbol{\beta}$ ด้วยวิธีกำลังสองน้อยสุด มีค่าความคลาดเคลื่อนกำลังสองเฉลี่ยดังนี้

$$\begin{aligned}
MSE(\hat{\boldsymbol{\beta}}) &= Var(\hat{\boldsymbol{\beta}}) \\
&= \sigma^2(\mathbf{X'X})^{-1} \qquad (2.7)
\end{aligned}$$

ค่าความคลาดเคลื่อนกำลังสองเฉลี่ยของ $\hat{\boldsymbol{\beta}}$ มีค่าเท่ากับค่าความแปรปรวนของ $\hat{\boldsymbol{\beta}}$ เนื่องจากตัวประมาณค่าของพารามิเตอร์ $\boldsymbol{\beta}$ โดยวิธีกำลังสองน้อยที่สุดเป็นตัวประมาณค่าไม่เอนเอียง $E(\hat{\boldsymbol{\beta}}) = \boldsymbol{\beta}$ การประมาณค่าพารามิเตอร์ $\boldsymbol{\beta}$ โดยวิธีการถดถอยแบบริดจ์จะได้ตัวประมาณ $\hat{\boldsymbol{\beta}}_\mathbf{r}$ ที่เอนเอียง กล่าวคือ $E(\hat{\boldsymbol{\beta}}_\mathbf{r}) = \mathbf{Z_r}\boldsymbol{\beta}$ และค่าความคลาดเคลื่อนกำลังสองเฉลี่ยของ $\hat{\boldsymbol{\beta}}_\mathbf{r}$ มีค่าดังสมการ (2.6) ดังนี้

$$\begin{aligned}
MSE(\hat{\boldsymbol{\beta}}_\mathbf{r}) &= Var(\hat{\boldsymbol{\beta}}_\mathbf{r}) + (Bias(\hat{\boldsymbol{\beta}}_\mathbf{r}))^2 \\
&= \sigma^2\sum_{j=1}^{p}\frac{\lambda_j}{(\lambda_j+r)^2} + r^2\boldsymbol{\beta}'(\mathbf{X'X}+r\mathbf{I})^{-2}\boldsymbol{\beta} \\
&= \gamma_1(r) + \gamma_2(r)
\end{aligned}$$

พิจารณาค่าความคลาดเคลื่อนกำลังสองเฉลี่ยของ $\hat{\boldsymbol{\beta}}_\mathbf{r}$ เพื่อหาขีดจำกัดของความแปรปรวนและความเอนเอียงกำลังสองของตัวประมาณ $\hat{\boldsymbol{\beta}}_\mathbf{r}$

โดยการหาขีดจำกัดของอนุพันธ์อันดับที่ 1 ของฟังก์ชัน $\gamma_1(r)$ และ $\gamma_2(r)$ เทียบกับค่าพารามิเตอร์ r

$$\lim_{r\to 0^+}\left(\frac{\partial\gamma_1}{\partial r}\right) = -2\sigma^2\sum_{j=1}^{p}\frac{1}{\lambda_j^2}$$

$$\lim_{r\to 0^+}\left(\frac{\partial\gamma_2}{\partial r}\right) = 0$$

ดังนั้นฟังก์ชัน $\gamma_1(r)$ มีอนุพันธ์อันดับ1เป็นลบมีค่าลู่เข้าสู่ $-2\sigma^2\sum_{j=1}^{p}\frac{1}{\lambda_i^2}$ เมื่อ $r \to 0^+$ [และ $\mathbf{X'X}$ เป็นเมตริกซ์เชิงตั้งฉาก และมีค่าลู่เข้าสู่ $-\infty$]เมื่อตัวแปรอิสระมีความสัมพันธ์กันเมตริกซ์ $\mathbf{X'X}$ มีสภาพไม่เหมาะสม เกือบจะเป็นเมตริกซ์เอกฐาน(Singular Matrix)ทำให้ตัวประมาณค่า $\boldsymbol{\beta}$ ที่ได้จากวิธีกำลังสองน้อยที่สุดมีค่าไม่เสถียร เปลี่ยนแปลงอย่างรวดเร็ว เมื่อมีการเปลี่ยนแปลงค่าของข้อมูลเพียงเล็กน้อย สำหรับค่าของ $Var(\hat{\boldsymbol{\beta}})$ ก็จะมีค่ามากสุด ส่วนฟังก์ชัน $\gamma_2(r)$ มีอนุพันธ์เท่ากับศูนย์ เมื่อ $r \to 0^+$ กล่าวคือ เมื่อ $r > 0$ จะมีความเอนเอียงกำลังสองของตัวประมาณ $\hat{\boldsymbol{\beta}}_\mathbf{r}$ เพียงเล็กน้อย แต่ความแปรปรวนของตัวประมาณ $\hat{\boldsymbol{\beta}}_\mathbf{r}$ จะมีค่าลดลงมาก จึงทำให้ค่าความคลาดเคลื่อนกำลังสองเฉลี่ยของวิธีการถดถอยแบบริดจ์มีค่าลดลงและมีค่าน้อยกว่าค่าความคลาดเคลื่อนกำลังสองเฉลี่ยที่ได้จากวิธีกำลังสองน้อยสุด

ทฤษฎีที่ 2.5 ตัวแบบการถดถอยเชิงเส้นพหุ $\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$ เมื่อ $\boldsymbol{\varepsilon}$ มีการแจกแจงแบบปกติมีค่าเฉลี่ย 0 และความแปรปรวน $\sigma^2\mathbf{I}$ จะได้ตัวประมาณค่ากำลังสองน้อยที่สุด $\hat{\boldsymbol{\beta}} = (\mathbf{X'X})^{-1}\mathbf{X'y}$ ถ้ากำหนดให้ $\mathbf{y} = \mathbf{Z}\boldsymbol{\alpha} + \boldsymbol{\varepsilon}$ เป็นตัวแบบการถดถอยคาโนนิคอล (Canonical Form) เมื่อ $\mathbf{Z} = \mathbf{XT}$, $\boldsymbol{\alpha} = \mathbf{T'}\boldsymbol{\beta}$ และ $\mathbf{Z'Z} = \mathbf{T'X'XT} = \boldsymbol{\Lambda}$ เมตริกซ์ $\boldsymbol{\Lambda}$ เป็นเมตริกซ์ทแยงมุมขนาด $p \times p$ ที่มีสมาชิกเป็นค่าเฉพาะของ $\mathbf{X'X}$ และ $\mathbf{T}$ เป็นเมตริกซ์เชิงตั้งฉากขนาด $p \times p$ ซึ่งมีสมาชิกในสดมภ์ของเมตริกซ์เป็นเวกเตอร์เฉพาะของ $\mathbf{X'X}$ ที่สอดคล้องกับค่าเฉพาะ λ จะได้ว่า $Var(\hat{\boldsymbol{\beta}}) = \mathbf{T}\boldsymbol{\Lambda}^{-1}\mathbf{T}'\sigma^2$

พิสูจน์

$$\begin{aligned} \mathbf{y} &= \mathbf{ZT'T}\boldsymbol{\alpha} + \boldsymbol{\varepsilon} \\ &= \mathbf{Z}\boldsymbol{\alpha} + \boldsymbol{\varepsilon} \qquad \text{เนื่องจาก } \mathbf{T'T} = \mathbf{I} \end{aligned}$$

ตัวประมาณค่ากำลังสองน้อยที่สุดในรูปคาโนนิคอล

$$\hat{\boldsymbol{\alpha}} = (\mathbf{Z'Z})^{-1}\mathbf{Z'y} = \boldsymbol{\Lambda}^{-1}\mathbf{Z'y} \tag{2.8}$$

$$\begin{aligned} Var(\hat{\boldsymbol{\alpha}}) &= Var[(\mathbf{Z'Z})^{-1}\mathbf{Z'y}] \\ &= (\mathbf{Z'Z})^{-1}\mathbf{Z}'Var(\mathbf{y})[(\mathbf{Z'Z})^{-1}\mathbf{Z}']' \\ &= \sigma^2(\mathbf{Z'Z})^{-1}\mathbf{Z'Z}(\mathbf{Z'Z})^{-1} \\ &= \sigma^2(\mathbf{Z'Z})^{-1} \\ &= \sigma^2\boldsymbol{\Lambda}^{-1} \end{aligned} \tag{2.9}$$

เมื่อกำหนด $\boldsymbol{\beta} = \mathbf{T}\boldsymbol{\alpha}$

จะได้ว่า $\hat{\boldsymbol{\beta}} = \mathbf{T}\hat{\boldsymbol{\alpha}}$

และ

$$\begin{aligned} Var(\hat{\boldsymbol{\beta}}) &= Var(\mathbf{T}\hat{\boldsymbol{\alpha}}) \\ &= \mathbf{T}Var(\hat{\boldsymbol{\alpha}})\mathbf{T'} \end{aligned}$$

$$Var(\hat{\boldsymbol{\beta}}) = \mathbf{T}\boldsymbol{\Lambda}^{-1}\mathbf{T}'\sigma^2 \tag{2.10}$$

2.1.2 การประมาณค่าคงตัว r ในการถดถอยแบบริดจ์

ในการศึกษาครั้งนี้จะศึกษาเปรียบเทียบการคัดเลือกตัวแบบโดยวิธีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบแบบขั้นตอนที่มีการประมาณค่า r ที่เป็นค่าคงตัว ในการประมาณค่าพารามิเตอร์ของตัวแบบการถดถอย 4 วิธีจากตัวประมาณ $\hat{\boldsymbol{\beta}}_{\mathbf{r}} = (\mathbf{X'X} + r\mathbf{I})^{-1}\mathbf{X'y}$

1) วิธีโฮเอิร์ล, เคนนาร์ด และ บาลด์วิน (Hoerl, Kennard and Baldwin: r_{HKB}) (1975)

$$r_{HKB} = \frac{p\hat{\sigma}^2}{\hat{\boldsymbol{\alpha}}'\hat{\boldsymbol{\alpha}}} \tag{2.11}$$

จากตัวประมาณค่าการถดถอยแบบริดจ์สามารถเขียนในรูปคาโนนิคอลได้ดังนี้ $\hat{\boldsymbol{\alpha}}_{\mathbf{r}} = (\mathbf{Z'Z} + \mathbf{R})^{-1}\mathbf{Z'y}$ โดยที่ $\mathbf{R}$ เป็นเมตริกซ์ของค่าคงตัว $r_j,\ j = 1, 2, ..., p$ และ $r_j \geq 0$ (Hoerl and Kennard, 1970a: 63)

โฮเอิร์ลและเคนนาร์ดได้เสนอตัวประมาณสัมประสิทธิ์การถดถอยแบบริดจ์และการประมาณค่าคงตัว r โดยพิจารณาจากค่าความคลาดเคลื่อนกำลังสองเฉลี่ยของ $\hat{\boldsymbol{\alpha}}_{\mathbf{r}}$ ดังนี้

$$MSE(\hat{\boldsymbol{\alpha}}_{\mathbf{r}}) = \hat{\sigma}^2 \sum_{j=1}^{p} \frac{\lambda_j}{(\lambda_j + r_j)^2} + \sum_{j=1}^{p} \frac{r_j^2 \hat{\alpha}_j^2}{(\lambda_j + r_j)^2}$$

โฮเอิร์ลและเคนนาร์ดต้องการเลือกค่า r_{HK} ที่ทำให้ค่าความคลาดเคลื่อนกำลังสองเฉลี่ยของ $\hat{\boldsymbol{\alpha}}_{\mathbf{r}}$ มีค่าต่ำสุดโดยการหาอนุพันธ์บางส่วน (Partial Derivative)เทียบกับค่า r_j และกำหนดให้มีค่าเท่ากับศูนย์

$$\frac{d}{d(r_j)}\left(\frac{\hat{\sigma}^2\lambda_j + \hat{\alpha}_j^2 r_j^2}{(\lambda_j + r_j)^2}\right) = \hat{0}$$

$$\frac{-2(\lambda_j + r_j)\lambda_j\hat{\sigma}^2 + 2(\lambda_j + r_j)^2\hat{\alpha}_j^2 r_j - 2(\lambda_j + r_j)\hat{\alpha}_j^2 r_j^2}{(\lambda_j + r_j)^4} = 0$$

$$-\lambda_j\hat{\sigma}^2 + \lambda_j\hat{\alpha}_j^2 r_j = 0$$

$$r_j = \frac{\hat{\sigma}^2}{\hat{\alpha}_j^2} \tag{2.12}$$

แต่เนื่องจาก $\boldsymbol{\alpha} = \mathbf{T'}\boldsymbol{\beta}$ ดังนั้นอาจมีกรณีที่ $\boldsymbol{\alpha}_{\mathbf{j}}$ มีค่าเท่ากับศูนย์ซึ่งจะทำให้ไม่สามารถหาค่า r ได้ และเนื่องจากค่า r มีผลต่อความเอนเอียงยกกำลังสองและค่าความแปรปรวนของตัวประมาณ ดังนั้นโฮเอิร์ลและเคนนาร์ด(Hoerl and Kennard: r_{HK})(1970a: 63-67) จึงกำหนดค่า r ที่เหมาะสมเพียงค่าเดียวโดยกำหนดให้ค่า $\hat{\alpha}_j$ จากสมการ (2.12) เป็นค่ามากที่สุด จะได้

$$r_{HK} = \frac{\hat{\sigma}^2}{\hat{\alpha}_{\max}^2} \tag{2.13}$$

เมื่อ $\hat{\alpha}_{\max} = \max(\hat{\alpha}_1, \hat{\alpha}_2, ..., \hat{\alpha}_p)$

และ $\hat{\sigma}^2 = \dfrac{\mathbf{y'y} - \hat{\boldsymbol{\alpha}}'\mathbf{T'X'y}}{n-p}$

จากสมการ(2.12)โฮเอิร์ล เคนนาร์ด และ บาลด์วิน (Hoerl, Kennard and Baldwin: r_{HKB}) เสนอวิธีการกำหนดค่า r ที่เหมาะสมเพื่อช่วยลดผลกระทบที่จะเกิดขึ้นเมื่อ $\hat{\alpha}_j^2$ มีค่าน้อยโดยการหาค่าเฉลี่ยฮามอร์นิคของค่า r_j

$$\frac{1}{r_{HKB}} = \frac{\sum_{j=1}^{p} \frac{1}{r_j}}{p}$$

$$= \frac{\sum_{j=1}^{p} \frac{\hat{\alpha}_j^2}{\hat{\sigma}^2}}{p}$$

$$= \frac{\hat{\boldsymbol{\alpha}}'\hat{\boldsymbol{\alpha}}}{p\hat{\sigma}^2}$$

ดังนั้น $$r_{HKB} = \frac{p\hat{\sigma}^2}{\hat{\boldsymbol{\alpha}}'\hat{\boldsymbol{\alpha}}}$$

เมื่อ p เป็นจำนวนพารามิเตอร์ที่ต้องประมาณค่าในตัวแบบ

$\hat{\sigma}^2 = \dfrac{\mathbf{y'y} - \hat{\boldsymbol{\alpha}}'\mathbf{T'X'y}}{n-p}$

และ $\hat{\boldsymbol{\alpha}} = \mathbf{T}'\hat{\boldsymbol{\beta}}$

2) วิธีลอว์เลสและแวง (Lawless and Wang: r_{LW}) (1976)

$$r_{LW} = \frac{p\hat{\sigma}^2}{\sum_{i=1}^{p} \lambda_i \hat{\alpha}_i^2} \tag{2.14}$$

เมื่อ $\hat{\alpha}_i$ เป็นสมาชิกตัวที่ i ของเวกเตอร์ประมาณค่าพารามิเตอร์ $\hat{\boldsymbol{\alpha}}$

λ_i เป็นสมาชิกตัวที่ i ของเมตริกซ์ $\boldsymbol{\Lambda}$

3) วิธีนอมูระ (Nomura: r_{HMO})(1988)

$$r_{HMO} = p\hat{\sigma}^2 / \sum_{i=1}^{p} [\hat{\alpha}_i^2 / \{1 + (1 + \lambda_i (\hat{\alpha}_i^2 / \hat{\sigma}^2)^{1/2})\}] \tag{2.15}$$

เมื่อ p เป็นจำนวนพารามิเตอร์ที่ต้องประมาณค่าในตัวแบบ

และ $\hat{\sigma}^2 = \dfrac{\mathbf{y'y} - \hat{\boldsymbol{\alpha}}'\mathbf{T'X'y}}{n-p}$

4) วิธีคาลาฟและชูเกอร์(Khalaf and Shukur: r) (2005: 1179)

$$r_{KS} = \frac{t_m\hat{\sigma}^2}{t_m\hat{\alpha}^2_{\max} + (n-p)\hat{\sigma}^2} \qquad (2.16)$$

เมื่อ t_m เป็นค่าเฉพาะที่มีค่ามากที่สุด (the largest eigenvalue) ของเมตริกซ์ $\mathbf{X'X}$

$\hat{\alpha}_{\max} = \max(\hat{\alpha}_1, \hat{\alpha}_2, ..., \hat{\alpha}_p)$

และ $\hat{\sigma}^2 = \dfrac{\mathbf{y'y - \hat{\alpha}'T'X'y}}{n-p}$

จากการประมาณค่าโดยวิธีโฮเอิร์ลและเคนนาร์ด(Hoerl and Kennard: r_{HK}) (1970a: 63) ในสมการ (2.13)คาลาฟและชูเกอร์ต้องการลดขนาดของ r_{HK} และควบคุมความแปรปรวนที่อาจเกิดจากขนาดตัวอย่างที่แตกต่างกัน จึงทำการบวกค่า $(n-p)\hat{\sigma}^2/t_m$ ให้กับตัวส่วนของ r_{HK} ดังนี้

$$r_{KS} = \frac{\hat{\sigma}^2}{\hat{\alpha}^2_{\max} + \dfrac{(n-p)\hat{\sigma}^2}{t_m}}$$

$$= \frac{\hat{\sigma}^2}{\dfrac{t_m\hat{\alpha}^2{}_{\max} + (n-p)\hat{\sigma}^2}{t_m}}$$

ดังนั้นค่าคงตัว r ของวิธีคาลาฟและชูเกอร์มีค่าเท่ากับ

$$r_{KS} = \frac{t_m\hat{\sigma}^2}{t_m\hat{\alpha}^2_{\max} + (n-p)\hat{\sigma}^2}$$

2.1.3 วิธีการคัดเลือกตัวแปรอิสระ

เนื่องจากการวิเคราะห์การถดถอยเป็นการศึกษาความสัมพันธ์ระหว่างตัวแปรอิสระและตัวแปรตามโดยนำข้อมูลของตัวแปรอิสระมาใช้ในการอธิบายตัวแปรตาม ซึ่งมีแนวคิดพื้นฐานมาจากการวิเคราะห์สหสัมพันธ์ (Correlation Analysis) ที่สามารถบอกได้ถึงระดับความสัมพันธ์ระหว่างตัวแปรอิสระแต่ละตัวกับตัวแปรตาม ดังนั้นจึงมีการพิจารณาถึงลำดับของตัวแปรอิสระที่ถูกเลือกเข้ามาอยู่ในตัวแบบการถดถอย นำมาสู่วิธีการในการเลือกตัวแปรอิสระเข้ามาในตัวแบบการถดถอยที่ให้ตัวแบบการถดถอยที่เหมาะสมที่สุด ซึ่งวิธีการเลือกตัวแปรอิสระดังกล่าวมีหลายวิธี (Montgomery, Peck and Vining, 2006: 277)แต่ในการศึกษาครั้งนี้ได้ทำการคัดเลือกตัวแปรอิสระโดยใช้วิธีการประมาณค่าพารามิเตอร์แบบกำลังสองน้อยที่สุดและการประมาณค่าพารามิเตอร์แบบริดจ์และใช้วิธีการถดถอยแบบขั้นตอนโดยที่คำนวณหาความคลาดเคลื่อนกำลังสองโดยวิธีการแบบริดจ์เปรียบเทียบกับการคัดเลือกตัวแปรอิสระที่ใช้วิธีการประมาณค่าพารามิเตอร์โดยวิธีการค้นหาแบบต้องห้าม ซึ่งมีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยมีค่าต่ำสุดและค่าความ

คลาดเคลื่อนกำลังสองเฉลี่ยปรับด้วยฟังก์ชันการลงโทษและใช้การทดสอบ t (t – Test) ในการคัดเลือกตัวแปรอิสระ

วิธีการถดถอยแบบขั้นตอนเป็นวิธีการเลือกตัวแปรอิสระเข้าในตัวแบบการถดถอยครั้งละหนึ่งตัวแปร โดยตัวแปรอิสระใดที่ถูกเลือกเข้าอยู่ในตัวแบบการถดถอยแล้วอาจจะถูกนำออกไปได้ภายหลัง หากพบว่าตัวแปรอิสระนั้นไม่มีนัยสำคัญนั่นคือต้องทดสอบว่าตัวแปรอิสระที่อยู่ในตัวแบบการถดถอยแล้ว มีส่วนในการอธิบายความผันแปรของตัวแปรตามในขณะที่มีตัวแปรอิสระอื่นอยู่ในตัวแบบการถดถอยหรือไม่ วิธีการถดถอยแบบขั้นตอนจึงเป็นวิธีที่รวมขั้นตอนของวิธีเลือกตัวแปรอิสระแบบไปข้างหน้าและวิธีตัดตัวแปรอิสระออกแบบถอยหลัง โดยขั้นตอนของวิธีการถดถอยแบบขั้นตอนแบ่งได้ ดังนี้

1) เริ่มต้นพิจารณาค่าสัมประสิทธิ์สหสัมพันธ์ระหว่างตัวแปรอิสระกับตัวแปรตามโดยเลือกตัวแปรอิสระซึ่งมีค่าสัมประสิทธิ์สหสัมพันธ์กับตัวแปรตามสูงที่สุด เข้าสู่ตัวแบบเป็นตัวแรกเหมือนกับวิธีเลือกตัวแปรอิสระแบบไปข้างหน้า

2) ตัวแปรอิสระที่เข้าสู่ตัวแบบแล้วจะมีการทดสอบนัยสำคัญของตัวแปรอิสระที่นำเข้าไปว่าสมควรอยู่ในตัวแบบหรือไม่โดยใช้ค่าสถิติ t (t – Value) หรือ ค่าสถิติ F (F – Value)หากพบว่าตัวแปรอิสระที่นำเข้าสู่ตัวแบบไม่มีนัยสำคัญ ขั้นตอนการเลือกก็จะหยุด แต่หากปฏิเสธสมมติฐานหลักจะไปดำเนินการในข้อที่ 3

3) พิจารณาค่าสัมประสิทธิ์สหสัมพันธ์บางส่วนระหว่างตัวแปรอิสระที่เหลือทุกตัวกับตัวแปรตาม เมื่อกำหนดให้ตัวแปรอิสระที่นำเข้าสู่ตัวแบบแล้วเป็นค่าคงตัว โดยจะเลือกตัวแปรอิสระที่ให้ค่าสัมประสิทธิ์สหสัมพันธ์บางส่วนกับตัวแปรตามสูงที่สุด เข้ามาอยู่ในตัวแบบเป็นตัวต่อมา

4) หลังจากนั้นจึงทำการทดสอบความมีนัยสำคัญของตัวแปรที่เข้ามาเป็นตัวสุดท้ายก่อน โดยพิจารณาจากค่าสถิติ F บางส่วน ถ้ามีค่าน้อยกว่าเกณฑ์ที่กำหนดไว้จะต้องนำตัวแปรนั้นออกจากตัวแบบ แต่ถ้าค่าสถิติ F บางส่วนมีค่ามากกว่าเกณฑ์ที่กำหนดไว้ ก็จะย้อนกลับไปทดสอบตัวแปรอิสระตัวก่อนหน้าที่เข้ามาอยู่ในตัวแบบ โดยพิจารณาจากค่าสถิติ F บางส่วนเช่นเดียวกัน

ทำซ้ำในขั้นตอนที่สามและสี่ต่อไปเรื่อยๆ จนกระทั่งไม่สามารถนำตัวแปรอิสระใดออกจากตัวแบบได้หรือไม่สามารถนำตัวแปรอิสระใดเข้าสู่ตัวแบบได้อีก

2.1.4 การทดสอบสำหรับการเลือกตัวแปรอิสระเข้าสู่ตัวแบบการถดถอย

1) การทดสอบ F รวม (OverallF Test) ใช้ในการทดสอบว่ามีตัวแปรอิสระอย่างน้อย 1 ตัวอยู่ในตัวแบบหรือไม่ โดยมีการทดสอบสมมติฐาน $H_0 : \beta_1 = \beta_2 = ... = \beta_k = 0$ เทียบ

กับ $H_1: \exists\beta_i = 0;\ i = 1, 2, ..., k$ โดยใช้สถิติทดสอบเป็น $\mathrm{F}_0 = \dfrac{SSR/k}{SSE/(n-k-1)} = \dfrac{\mathrm{MSR}}{\mathrm{MSE}}$ จะปฏิเสธสมมติฐาน $H_0: \beta_1 = \beta_2 = ... = \beta_k = 0$ เมื่อ $\mathrm{F}_0 > \mathrm{F}_{\alpha,k,n-k-1}$ ถ้าไม่ปฏิเสธ H_0 แสดงว่าไม่มีตัวแปรอิสระอยู่ในตัวแบบเลย

2) การทดสอบ β แต่ละตัว

การทดสอบ t (t – Test)ใช้ในการทดสอบว่าตัวแปรอิสระที่ถูกเลือกเข้ามาอยู่ในตัวแบบทำให้ผลบวกกำลังสองของการถดถอย (SSR) เพิ่มขึ้นและผลบวกกำลังสองของความคลาดเคลื่อน (SSE) ลดลงหรือไม่เพื่ออธิบายว่าตัวแปรอิสระนั้นมีความสามารถในการอธิบายตัวแปรตามได้สูงขึ้น โดยมีการทดสอบสมมติฐาน $H_0: \beta_i = 0$ เทียบกับ $H_0: \beta_i \neq 0$ โดยใช้สถิติทดสอบเป็น $t_0 = \dfrac{\hat{\beta}_i}{SE(\hat{\beta}_i)}$ โดยที่ $SE(\hat{\beta}_i) = \sqrt{\dfrac{\sum_{i=1}^{m}(\hat{\beta}_i - \bar{\hat{\beta}})^2}{m}}$ จะปฏิเสธสมมติฐาน H_0 เมื่อ $|t_0| > t_{\frac{\alpha}{2},n-k-1}$ ถ้าไม่ปฏิเสธ H_0 แสดงว่าตัวแปรอิสระนั้นจะต้องนำออกจากตัวแบบ แต่หากปฏิเสธ H_0 แสดงว่าตัวแปรอิสระนั้นสามารถอยู่ในตัวแบบได้

การทดสอบ F(F - Test)ใช้ในการทดสอบ เช่นเดียวกันกับ t –Test คือ ใช้ในการทดสอบว่าตัวแปรอิสระที่ถูกเลือกเข้ามาสามารถอยู่ในตัวแบบได้หรือไม่ โดยมีการทดสอบสมมติฐาน $H_0: \beta_i = 0$ เทียบกับ $H_1: \beta_i \neq 0$ ใช้สถิติทดสอบ เป็น

$$\mathrm{F}_0 = \frac{SSR}{SSE/(n-2)}$$

จะปฏิเสธสมมติฐาน $H_0: \beta_i = 0$ เมื่อ $\mathrm{F}_0 \geq \mathrm{F}_{\alpha,(1,n-2)}$ โดยให้ผลสรุปเช่นเดียวกันกับในt – Test เมื่อกำหนดระดับนัยสำคัญของการทดสอบสมมติฐานเท่ากับ α

3) การทดสอบเอฟบางส่วน (Partial F Test) ใช้ในการทดสอบตัวแปรอิสระตัวปัจจุบันที่ถูกเลือกเข้ามาในตัวแบบ เมื่อกำหนดให้ตัวแปรที่พิจารณาเป็นตัวแปรที่เข้ามาเป็นตัวสุดท้าย เมื่อตัวแปรอื่นๆอยู่ในตัวแบบแล้ว ซึ่งใช้ในการทดสอบสมมติฐาน $H_0: \beta_j = 0$เทียบกับ $H_1: \beta_j \neq 0$; เมื่อ $j = 1, 2, ..., k$ สำหรับบางค่า j โดยมีสถิติทดสอบเป็น

$$\mathrm{F}_0 = \frac{SSR(\beta_j \mid \beta_0, ..., \beta_{j-1}, \beta_{j+1}, ..., \beta_k)}{\mathrm{MSE}}$$

จะปฏิเสธ H_0 เมื่อ $\mathrm{F}_0 > \mathrm{F}_{\alpha,k,n-k-1}$ ถ้าไม่ปฏิเสธ H_0 แสดงว่าตัวแปรอิสระนั้นจะต้องนำออกจากตัวแบบ แต่หากปฏิเสธ H_0 แสดงว่าตัวแปรอิสระนั้นสามารถอยู่ในตัวแบบได้ เมื่อกำหนดระดับนัยสำคัญของการทดสอบสมมติฐานเท่ากับ α

2.1.5 การค้นหาแบบต้องห้าม (Tabu Search)

การค้นหาแบบต้องห้ามเป็นวิธีการหาคำตอบแบบมีเหตุผล (Metaheuristic) วิธีหนึ่ง โดยวิธีดังกล่าวถูกดัดแปลงมาจากวิธีการทางปัญญาประดิษฐ์ (Artificial Intelligence) ให้มีความเหมาะสมในการใช้คำตอบเดิมในการตอบปัญหาเดิมที่มีวัตถุประสงค์เปลี่ยนแปลงไป และมีการนำมาใช้ในการหาคำตอบที่ใกล้เคียงกับค่าอุตตมะ(Optimum)(Glover, 1990: 74)ซึ่งนิยมใช้กับปัญหาในการตัดสินใจ เช่น ปัญหาการหาเส้นทางเดินของพนักงานขาย (Travelling Salesman Problem) ปัญหาการจัดช่องทางในการให้บริการในร้านสะดวกซื้อ ปัญหาการจัดตารางเวลาการทำงานในโรงงาน (Job Shop Scheduling) เป็นต้น ขั้นตอนของวิธีการนี้ไม่ซับซ้อนและผลลัพธ์ที่ได้มีประสิทธิภาพสูง แนวคิดของวิธีการนี้ คือ ใช้การจดจำจากรอบการทำซ้ำที่ผ่านมา โดยใช้หน่วยความจำในเครื่องคอมพิวเตอร์มาช่วยในการประมวลผล นอกจากนี้ยังสามารถใช้ในการหาคำตอบของปัญหาที่มีความซับซ้อนและมีตัวแปรที่ใช้พิจารณาเป็นจำนวนมาก วิธีการนี้ช่วยป้องกันการเกิดปัญหาคำตอบที่มีค่าต่ำสุดเฉพาะที่(Local Minimum) โดยแก้ไขการย้อนกลับไปหาคำตอบเดิมที่อาจจะมีค่ามากกว่าคำตอบที่มีค่าต่ำสุดเฉพาะที่ และดำเนินการค้นหาคำตอบต่อไปจนกระทั่งได้คำตอบที่ใกล้เคียงกับค่าต่ำสุดทั่วไป (Global Minimum)หรือค้นหาคำตอบต่อไปอีกระยะหนึ่ง (อาทิเช่น Fred Glover กำหนดจำนวนรอบการค้นหาเป็น 30 รอบ) และไม่พบคำตอบที่มีค่าต่ำกว่าค่าต่ำสุดทั่วไปที่ได้ค้นพบแล้ว รูปแบบการค้นหาคำตอบประกอบด้วยการค้นหาที่สำคัญ 2 รูปแบบ คือ การค้นหาคำตอบโดยใช้หน่วยความจำระยะสั้น (Short TermMemory) ซึ่งเป็นหน่วยความจำตามเวลา (Recency Base Memory) หรือการค้นหาที่จดจำอดีตในการค้นหาที่ผ่านมาในระยะสั้น และการค้นหาคำตอบโดยการใช้หน่วยความจำระยะยาว (Long Term Memory) ซึ่งถือเป็นหน่วยความจำตามความถี่ (Frequency Base Memory) หมายถึง การค้นหาที่จดจำอดีตเพื่อช่วยให้การค้นหาคำตอบเป็นไปอย่างมีประสิทธิภาพมากขึ้น มีองค์ประกอบหลัก ดังนี้

1) ฟังก์ชันเป้าหมาย (Objective Function) ใช้ในการประกอบการตัดสินใจในการค้นหาคำตอบ โดยจะเป็นตัวประเมินการค้นหาคำตอบที่อยู่ในแต่ละขั้นตอน

2) ข้อจำกัดการเป็นต้องห้าม (Tabu Restriction) คือ การควบคุมการย้อนกลับ (Reverse) หรือการวนรอบ (Cyclic) อยู่ในคำตอบเดิม โดยการกำหนดข้อจำกัดการเป็นต้องห้ามซึ่งขึ้นอยู่กับแต่ละปัญหา เช่น กำหนดระยะเวลาหรือจำนวนรอบที่จะต้องอยู่ในรายการต้องห้ามเป็นอย่างน้อย ฯลฯ

3) รายการต้องห้าม (Tabu List) เป็นรายการค่าของตัวแปรที่ถูกปรับเปลี่ยนในแต่ละรอบของการคัดเลือกค่าของตัวแปร เพื่อป้องกันไม่ให้กลับมาที่คำตอบเดิม

4) ข้อมูลความถี่ (Frequency Move) เป็นการใช้หน่วยความจำระยะยาวบันทึกการค้นหาทุกค่าตลอดช่วงของการค้นหา โดยใช้หลักการสร้างความหลากหลาย (Diversification) เพื่อค้นหาคำตอบในบริเวณที่แตกต่างจากที่ค้นพบมาแล้ว

5) เกณฑ์ความปรารถนา (Aspiration Criteria) เป็นรูปแบบการควบคุมที่ช่วยในการค้นหาคำตอบที่เหมาะสมเป็นไปอย่างรวดเร็ว โดยกำหนดเงื่อนไขสำหรับการค้นหาที่เป็นรายการต้องห้ามซึ่งให้คำตอบที่เหมาะสมที่สุดตั้งแต่การเริ่มค้นหาจนถึงปัจจุบัน จนสามารถยกเลิกการห้ามสำหรับการค้นหานั้นได้ อย่างไรก็ตามเกณฑ์ดังกล่าวนี้ขึ้นอยู่กับลักษณะของแต่ละปัญหา

ขั้นตอนทั่วไปในการทำงานสรุปได้ดังนี้

1) เริ่มจากการหาคำตอบเริ่มต้น โดยมีการจัดทำโครงสร้างหน่วยความจำมาใช้ในการเก็บค่า

2) ต่อมาสร้าง Candidate และเลือกคำตอบจากใน Candidate ที่มีความเหมาะสมที่สุดโดยจะหยุดการค้นหาคำตอบตามเงื่อนไขที่กำหนดไว้

3) ปรับปรุง (Update) คำตอบโดยการแทนที่คำตอบที่ดีที่สุดในปัจจุบันด้วยคำตอบที่ดีกว่า จากนั้นจึงเก็บคำตอบบันทึกไว้ด้วย

ดังนั้นจะพบว่าวิธีการค้นหาแบบต้องห้าม (Tabu Search) เป็นวิธีการแก้ปัญหาที่ไม่ซับซ้อน แต่ความสำคัญอยู่ที่การออกแบบโครงสร้างและการจัดการข้อมูลในหน่วยความจำ รวมถึงการกำหนดลักษณะ (Attribute) ของ Candidate มีขั้นตอนในการทำงานดังนี้

1) กำหนดค่าพิสัยเริ่มต้น โดยใช้การพิจารณาจากตัวประมาณค่า OLS

2) สร้างคำตอบเริ่มต้นโดยกำหนดค่าตัวแปรอิสระต่างๆจากการสุ่มค่าในพิสัยที่กำหนดพิสัยของตัวแปรอิสระแต่ละตัวอาจแตกต่างกันได้ตามการประมาณการเริ่มต้น

3) สร้างเซตคำตอบข้างเคียง (Neighborhood) ของค่าคำตอบปัจจุบันที่กำหนดขึ้น

4) การตรวจสอบรายการต้องห้าม (Tabu List) ซึ่งเป็นรายการที่บันทึกข้อมูลกระบวนการค้นหาในอดีต โดยถ้าคำตอบข้างเคียงนั้นเป็นรายการต้องห้ามเกินระยะเวลาที่กำหนดให้ยกเลิกการเป็นรายการต้องห้ามของคำตอบข้างเคียงนั้น

5) ดำเนินการค้นหาคำตอบจากเซตคำตอบข้างเคียง โดยคำตอบข้างเคียงนั้นต้องไม่เป็นรายการต้องห้ามและเป็นคำตอบที่ดีกว่าคำตอบที่มีค่าฟังก์ชันเป้าหมายดีที่สุดจากเซตคำตอบข้างเคียง

6) กำหนดเกณฑ์ความปรารถนา (Aspiration Criteria) ในการตรวจสอบคำตอบข้างเคียง ถ้าคำตอบข้างเคียงนั้นเป็นรายการต้องห้าม แต่สามารถให้คำตอบที่มีค่าฟังก์ชันเป้าหมายที่ดีกว่าก็สามารถเลือกมาเป็นคำตอบได้

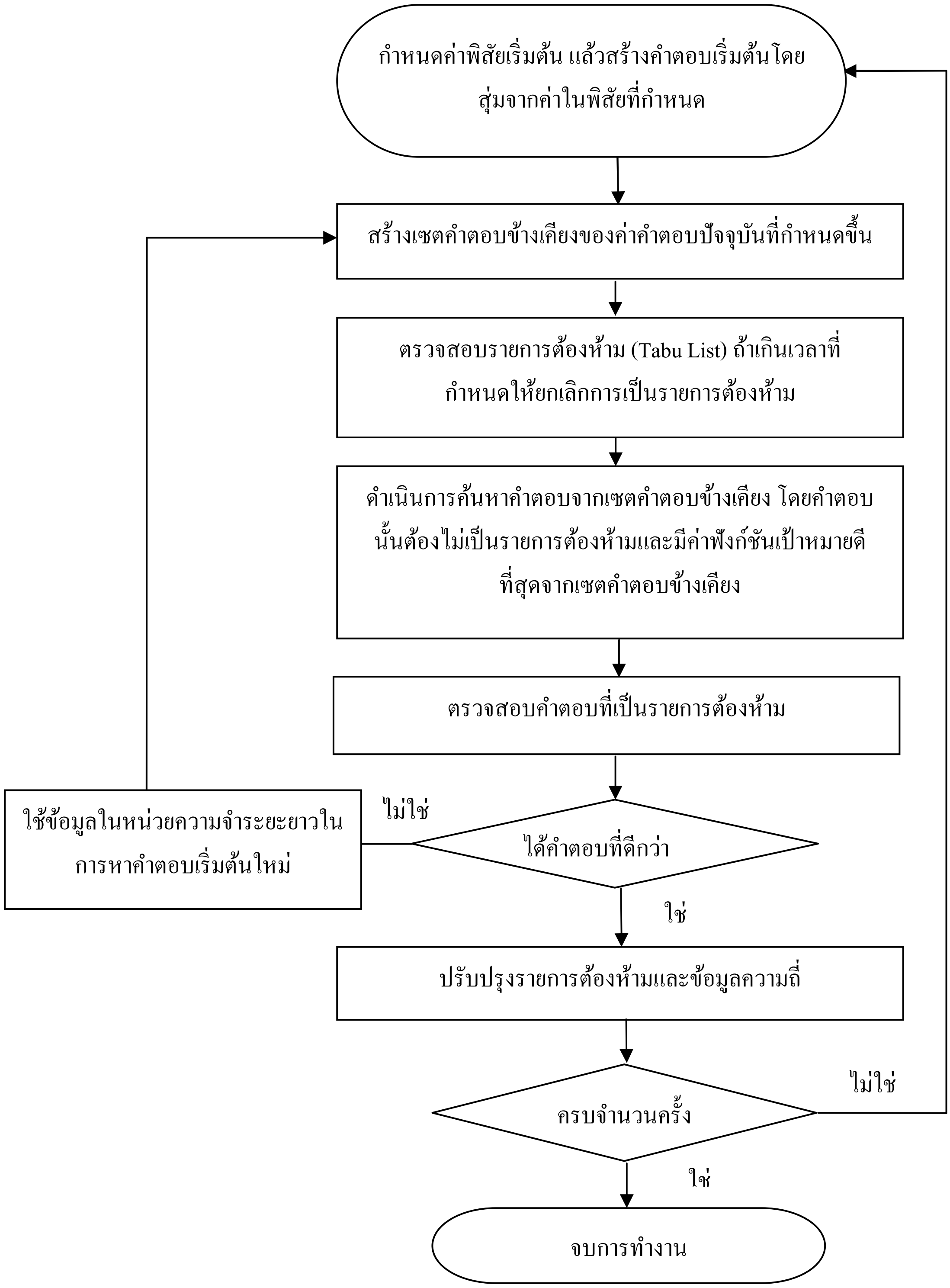
7) ถ้าไม่สามารถหาคำตอบที่มีค่าฟังก์ชันเป้าหมายดีกว่าปัจจุบันได้ให้เรียกใช้หน่วยความจำระยะยาว (Long Term Memory) มาใช้ในการหาคำตอบเริ่มต้นใหม่

8) ย้ายตำแหน่ง (Move) ไปยังคำตอบปัจจุบันที่ถูกเลือก แล้วปรับปรุงรายการต้องห้าม (Tabu List) ให้เป็นปัจจุบันและเพิ่มจำนวนครั้งในการเลือกข้อมูลความถี่ (Frequency Move)

9) ทำซ้ำจนกระทั่งได้ค่าฟังก์ชันเป้าหมายอยู่ในเกณฑ์ที่กำหนดขึ้นหรือครบตามจำนวนครั้งที่กำหนด

เนื่องจากวิธีการค้นหาแบบต้องห้ามเป็นวิธีที่มีประสิทธิภาพในการค้นหาคำตอบที่ใกล้เคียงกับค่าอุตตมะ จึงมีนักวิจัยจำนวนมากนำวิธีการค้นหาแบบต้องห้ามไปประยุกต์ใช้ในสาขาต่างๆ โดยมีการนำวิธีการค้นหาแบบต้องห้ามมาประยุกต์ใช้ในการคัดเลือกตัวแปรอิสระในการวิเคราะห์การถดถอยเชิงเส้นพหุ สรุปขั้นตอนการทำงานได้ตามภาพที่ 2.1

อนึ่งการค้นหาแบบต้องห้ามไม่ได้ใช้การประมาณค่าสัมประสิทธิ์การถดถอย $(\boldsymbol{\beta})$ แบบ OLS หรือ Maximum Likelihood ดังในวิธีที่ได้กล่าวมาทั้งหมด แต่การค้นหาแบบต้องห้ามจะค้นหาค่าของสัมประสิทธิ์การถดถอย $(\boldsymbol{\beta})$ ที่ทำให้ฟังก์ชันเป้าหมายมีค่าต่ำสุด จึงไม่ต้องคำนวณค่า $(\mathbf{X'X})^{-1}$ ดังนั้นการค้นหาแบบต้องห้ามจึงสามารถใช้ได้กับข้อมูลทั้งที่ไม่มีและมีสหสัมพันธ์เชิงเส้นพหุ (Multicollinearity)



ภาพที่ 2.1 ขั้นตอนวิธีการค้นหาแบบต้องห้าม

โดยทั่วไปฟังก์ชันเป้าหมายคือ ค่าความคลาดเคลื่อนกำลังสองเฉลี่ย(F_1)

$$F_1 = \frac{(\mathbf{y}-\hat{\mathbf{y}})'(\mathbf{y}-\hat{\mathbf{y}})}{n-k-1}$$

โดยค้นหาค่า $\hat{\boldsymbol{\beta}}$ ที่ทำให้ F_1 มีค่าต่ำสุด

อีกแนวคิดหนึ่งที่นำมาใช้ในการศึกษานี้คือนำ $\hat{\boldsymbol{\beta}}'\hat{\boldsymbol{\beta}}$ มาพิจารณาด้วยในการหาค่าต่ำสุดของ F_1 กล่าวคือต้องการให้ F_1 มีค่าน้อยในขณะเดียวกันกับค่า $\hat{\boldsymbol{\beta}}'\hat{\boldsymbol{\beta}}$ ต้องไม่มากด้วย จึงนำมาสู่ F_2 ซึ่งเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยปรับด้วยฟังก์ชันการลงโทษ

$$F_2 = \frac{(\mathbf{y}-\hat{\mathbf{y}})'(\mathbf{y}-\hat{\mathbf{y}})}{n-k-1} + \frac{c\hat{\boldsymbol{\beta}}'\hat{\boldsymbol{\beta}}}{n-p}$$

โดย $c, 0 < c \le 1$ เป็นพารามิเตอร์ที่ต้องค้นหาค่าที่เหมาะสมพร้อมๆกับ $\hat{\boldsymbol{\beta}}$ เพื่อทำให้ F_2 มีค่าต่ำสุด

$\hat{\mathbf{y}}$ เป็นเวกเตอร์ค่าพยากรณ์ที่ได้มาจากสมการการถดถอยที่แทนสัมประสิทธิ์การถดถอยที่ประมาณได้จากวิธีการค้นหาแบบต้องห้าม

$\hat{\boldsymbol{\beta}}$ เป็นเวกเตอร์ของค่าสัมประสิทธิ์การถดถอยที่ประมาณจากวิธีการค้นหาแบบต้องห้าม

2.2 ผลงานวิจัยที่เกี่ยวข้อง

งานวิจัยที่เกี่ยวข้องกับการประมาณค่าพารามิเตอร์การถดถอยพหุโดยวิธีกำลังสองน้อยที่สุด วิธีแบบริดจ์และการคัดเลือกตัวแบบการถดถอยพหุ

Holland (1975) ได้เสนอวิธีการค้นหาคำตอบแบบเจนเนติก(Genetic Algorithm) ซึ่งเป็นวิธีการค้นหาคำตอบที่เลียนแบบมาจากการคัดเลือกพันธุกรรมตามธรรมชาติ โดยพันธุกรรมของสิ่งมีชีวิตจะมีการพัฒนาโดยการจัดสรรสิ่งที่ดีที่สุดในสายพันธุ์ เพื่อสืบทอดไปยังรุ่นต่อไป ในการค้นหาคำตอบจะใช้การพิจารณาคำตอบที่ได้จากขั้นตอนการคำนวณก่อนหน้าเพื่อค้นหาคำตอบที่ดีกว่าในขั้นตอนต่อไป ซึ่งการค้นหาคำตอบด้วยวิธีนี้มีการพัฒนาอย่างรวดเร็วทำให้เป็นที่สนใจศึกษาอย่างกว้างขวาง โดยวิธีนี้มีจุดเด่น คือ สามารถใช้ในการคำนวณได้ทั้งในกรณีตัวแปรต่อเนื่องและไม่ต่อเนื่อง รวมทั้งกรณีที่มีตัวแปรสุ่มทั้งสองประเภท โดยหลักการคำนวณคำตอบจะพิจารณากำหนดในรูปของโครโมโซม จากนั้นจะสร้างและปรับปรุงคุณภาพของประชากรโดยผ่านกระบวนการต่างๆ เช่น Mutation เพื่อค้นหาคำตอบที่ใกล้เคียงกับค่าอุตตมะ ขั้นตอนจะเริ่มจากการกำหนดฟังก์ชันเป้าหมาย (Objective Function) จากนั้นกำหนดและระบุถึงสิ่งที่ใช้แทนเจนเนติก ต่อมากำหนดและระบุถึงการดำเนินการของเจนเนติก แล้วทำการสุ่มคำตอบเริ่มต้น แล้วจึงใช้แนวคิดของวิวัฒนาการโดยใช้หลักการคัดเลือกทางธรรมชาติ เช่น Crossover การจำลอง

(Duplicate) การลบ (Delete) ซึ่งจะทำให้ได้คำตอบในรูปทั่วไป (General Solution) ที่เป็นคำตอบของพารามิเตอร์ตามที่ต้องการ

Knox (1989) ได้ศึกษาปัญหาการหาเส้นทางเดินของพนักงานขาย (Traveling Salesman Problem) และเสนอให้ใช้วิธีการค้นหาแบบต้องห้ามในการค้นหาคำตอบโดยปัญหาดังกล่าวเป็นปัญหาพื้นฐานที่เกิดขึ้นในการขนส่งแต่ไม่สามารถแก้ไขได้ Karl Menger (อ้างอิงใน Knox, 1989) ศึกษารูปแบบทั่วไปของปัญหาและใช้ตัวแบบทางคณิตศาสตร์เป็นครั้งแรกในทศวรรษ 1930 พบว่าปัญหานี้เป็นปัญหาในด้านการจัดกลุ่ม (Combinatorial) ซึ่งไม่สามารถหาคำตอบที่ใกล้เคียงกับค่าอุตมะในวิธีการคำนวณปกติ จึงมีการนำปัญหานี้มาใช้การค้นหาคำตอบแบบมีเหตุผล (Metaheuristic) การค้นหาคำตอบจะพิจารณาจากเงื่อนไขที่กำหนดไว้เบื้องต้น และใช้หน่วยความจำในการค้นหาคำตอบถัดไปจากเซตคำตอบปัจจุบัน นอกจากนี้พบว่าวิธีการค้นหาแบบต้องห้ามยังมีความเหมาะสมกับปัญหาด้านการจัดกลุ่ม เนื่องจากปัญหาในลักษณะดังกล่าวมีจำนวนตัวแปรที่นำมาพิจารณามากส่งผลให้มีมิติขนาดใหญ่ ทำให้มีความซับซ้อนในการพิจารณาค้นหาคำตอบโดยวิธีการแบบปกติ ดังนั้นวิธีการค้นหาแบบต้องห้ามจึงสามารถนำมาประยุกต์ใช้ได้เป็นอย่างดีกับปัญหาการหาเส้นทางเดิน โดยใช้ระยะเวลาในการคำนวณน้อยกว่าวิธี Simulated Annealing

จิรายุส พู่มนตรี (2533) ทำการศึกษาเปรียบเทียบตัวประมาณค่าพารามิเตอร์ของตัวแบบการถดถอยเชิงเส้นพหุที่ใช้วิธีแบบริดจ์ โดยกำหนดค่า k ตามวิธีโฮเอิร์ล เคนนาร์ด และ บาลด์วิน (Hoerl, Kennard and Baldwin, 1975)วิธีโฮเอิร์ล เคนนาร์ด(Hoerland Kennard, 1970a) วิธีซีซานลี (Tze San Lee, 1986)และวิธีการประมาณค่าพารามิเตอร์โดยใช้วิธีค้นหาแบบทวิ (Binary Search) เกณฑ์ที่ใช้ในการเปรียบเทียบคือ อัตราส่วนของค่าเฉลี่ยความคลาดเคลื่อนกำลังสอง ภายใต้ฐานคติว่าการแจกแจงของค่าความคลาดเคลื่อนมีการแจกแจงปกติกำหนดค่าเฉลี่ยเท่ากับ 1 ส่วนเบี่ยงเบนมาตรฐานเท่ากับ 0.05, 0.10 และ 0.15 และค่าสัมประสิทธิ์การถดถอยพหุจากเวกเตอร์เฉพาะ (Eigenvector)ซึ่งสอดคล้องกับเวกเตอร์เฉพาะที่ต่ำที่สุดและที่มากที่สุด การแจกแจงปกติปลอมปนที่กำหนดสเกลแฟกเตอร์เท่ากับ 3 และ 10 เปอร์เซนต์การปลอมปนเท่ากับ 5 และ 10 และค่าสัมประสิทธิ์การถดถอยพหุจากเวกเตอร์เฉพาะซึ่งสอดคล้องกับเวกเตอร์เฉพาะที่ต่ำที่สุด การแจกแจงลอกนอร์มอลได้กำหนดค่าเฉลี่ยเท่ากับ 1 และค่าความเบี่ยงเบนมาตรฐานเท่ากับ 0.22, 0.59และ 1.00 ในการศึกษาได้กำหนดจำนวนตัวแปรอิสระเท่ากับ 3 กำหนดระดับความสัมพันธ์ระหว่างตัวแปรอิสระเป็น (0.70, 0.30), (0.90, 0.90) และ (0.99, 0.99) ในการศึกษาได้ใช้การจำลองตัวแบบโดยใช้เทคนิคมอนติคาร์โลและกระทำซ้ำ 200 ครั้งในแต่ละกรณี ผลการศึกษาสรุปได้ดังนี้ 1) กรณีความคลาดเคลื่อนมีการแจกแจงปกติและปกติปลอมปนวิธีการประมาณค่าพารามิเตอร์ของตัวแบบ

โดยการค้นหาแบบทวิ (Binary Search)ให้ผลดีกว่าในทุกกรณี ซึ่งผลที่ได้ใกล้เคียงกับผลที่ได้จากวิธีวิธีโฮเอิร์ล เคนนาร์ด และ บาลด์วิน,วิธีโฮเอิร์ล เคนนาร์ดโดยที่ค่าเฉลี่ยความคลาดเคลื่อนกำลังสองแปรผันตามระดับความสัมพันธ์ ความแปรปรวน และจำนวนตัวแปรอิสระ แต่แปรผกผันกับขนาดตัวอย่าง 2)กรณีที่ความคลาดเคลื่อนมีการแจกแจงลอกนอร์มอล พบว่าวิธีการประมาณค่าพารามิเตอร์ของตัวแบบโดยการค้นหาแบบทวิ (Binary Search) ให้ผลดีกว่าในทุกกรณี ซึ่งผลที่ได้ใกล้เคียงกับผลที่ได้จากวิธีซีซานลี เมื่อความแปรปรวนสูง ขนาดตัวอย่างน้อย จำนวนตัวแปรอิสระมาก และระดับความสัมพันธ์สูง โดยที่ค่าเฉลี่ยความคลาดเคลื่อนกำลังสองจะแปรผันตามระดับความสัมพันธ์ ความแปรปรวน และจำนวนตัวแปรอิสระแต่แปรผกผันกับขนาดตัวอย่าง

นุสรา สถิตโพธิ์ศรี (2535) ได้ทำการศึกษาเปรียบเทียบตัวประมาณค่าสัมประสิทธิ์การถดถอยพหุ ในกรณีที่เกิดพหุสัมพันธ์ระหว่างตัวแปรอิสระ โดยใช้การประมาณค่าพารามิเตอร์ด้วยวิธีกำลังสองน้อยที่สุด (OLS) และการประมาณค่าการถดถอยแบบริดจ์ เมื่อค่าพารามิเตอร์แบบริดจ์คำนวณจากวิธี วิธีโฮเอิร์ล, เคนนาร์ด และ บาลด์วิน (Hoerl, Kennard and Baldwin) และวิธีลอว์เลสและแวง (Lawless and Wang)และตัวประมาณการถดถอยแบบลาเท้นรูทธ์(LR) เมื่อความคลาดเคลื่อนมีการแจกแจงปกติ ปกติปลอมปน และลอกนอร์มอล โดยที่ขนาดตัวอย่างเท่ากับ 10, 30 และ 50 และศึกษาเมื่อจำนวนตัวแปรอิสระเท่ากับ 3 และ 5 ตามลำดับ กำหนดระดับการเกิดพหุสัมพันธ์ระหว่างตัวแปรอิสระเป็น 9 ช่วง ข้อมูลที่ใช้ในการวิจัยครั้งนี้มาจากการจำลองโดยใช้หลักการของ Mutinormalทำซ้ำ 1,000 ครั้ง ในแต่ละสถานการณ์ เกณฑ์ที่ใช้ในการเปรียบเทียบ คือ ค่าเฉลี่ยความคลาดเคลื่อนกำลังสอง (AMSE)ผลการศึกษาสรุปได้ดังนี้ เมื่อความคลาดเคลื่อนของข้อมูลมีการแจกแจงปกติโดยมีค่าเฉลี่ยเท่ากับ 0 และค่าความแปรปรวนเท่ากับ 1พบว่าตัวประมาณ LR ให้ค่า AMSE ต่ำที่สุด ทั้งในกรณีที่จำนวนตัวแปรอิสระเท่ากับ 3 และ 5 ยกเว้นเมื่อตัวอย่างมีขนาดเล็กและระดับการเกิดพหุสัมพันธ์สูงขึ้นตัวประมาณ HKB เป็นตัวประมาณที่ดีที่สุด เมื่อความคลาดเคลื่อนมีการแจกแจงปกติปลอมปน พบว่า โดยส่วนใหญ่ ตัวประมาณ LR เป็นตัวประมาณที่ดีที่สุด ยกเว้นเมื่อ ตัวอย่างมีขนาดเล็กและปานกลาง และระดับความสัมพันธ์สูงขึ้น ตัวประมาณ HKB จะให้ค่า AMSE ต่ำที่สุด นอกจากนี้ ในกรณีที่ค่าสเกลแฟคเตอร์และเปอร์เซนต์ของการปลอมปนมีค่าสูงขึ้น ตัวประมาณ HKB จะให้ค่า AMSE ต่ำที่สุดในทุกกรณีของระดับความสัมพันธ์ที่ทำการศึกษา สำหรับในกรณีที่ความคลาดเคลื่อนมีการแจกแจงลอกนอร์มอลที่มีค่าเฉลี่ยเท่ากับ0 และความแปรปรวนเท่ากับ 1สรุปได้ว่า ตัวประมาณ LR เป็นตัวประมาณที่ให้ค่า AMSE ต่ำที่สุดในทุกกรณี จากการพิจารณาถึงปัจจัยที่มีผลต่อค่าเฉลี่ยความคลาดเคลื่อนกำลังสอง พบว่าการเพิ่มจำนวนตัวแปรอิสระ การเพิ่มค่าการเกิดพหุสัมพันธ์ ทำให้ค่า AMSEสูงขึ้น และการเพิ่มขนาดตัวอย่างมีผลทำให้ค่า

AMSE ลดลง นอกจากนี้เมื่อค่าความคลาดเคลื่อนมีการแจกแจงปกติ ตัวประมาณค่าพารามิเตอร์ทั้ง 4 วิธี ให้ค่า AMSEต่ำกว่าในกรณีที่ค่าความคลาดเคลื่อนมีการแจกแจงปกติปลอมปนและลอกนอร์มอล

Wasserman and Sudijanto (1994) ได้ทำการศึกษาการคัดเลือกตัวแบบในการวิเคราะห์การถดถอยเชิงเส้น โดยวิธี Genetic Algorithm ซึ่งเป็นวิธีการที่ใช้แนวคิดมาจากกระบวนการทางพันธุกรรม (Genetic Algorithm) ในการจำลองโครโมโซม และนำมาเปรียบเทียบกับวิธีการถดถอยแบบขั้นตอน เกณฑ์ที่ใช้ในการพิจารณาเลือกตัวแปรอิสระเข้าในตัวแบบคือ AIC และพิจารณาเปรียบเทียบทั้ง 2 วิธี โดยใช้ค่า MSE ข้อมูลที่ใช้ในการวิจัยมาจากการจำลองข้อมูล โดยตัวแบบเต็มมีตัวแปรอิสระทั้งหมด 24 ตัวแปร ในวิธี Genetic Algorithm มีการสุ่มตัวแบบขึ้นมา 100 ตัวแบบ เพื่อใช้เป็นประชากรเริ่มต้น ผลการศึกษาจากทั้งสองวิธีพบว่าตัวแปรอิสระที่อยู่ในตัวแบบทั้งหมด 7 ตัวแปร คือ X_4, X_8, X_{14}, X_{15}, X_{17}, X_{20} และ X_{22} แต่ตัวแบบที่ได้จากวิธี Genetic Algorithm จะให้ค่า MSE เป็น 1,417.49 ในขณะที่วิธีเพิ่มตัวแปรอิสระแบบขั้นตอน จะให้ค่า MSE เป็น 1,548.52 ดังนั้นทำให้สามารถสรุปได้ว่าในกรณีที่มีจำนวนตัวแปรอิสระมาก วิธี Genetic Algorithm จะมีความแม่นยำสูงกว่าวิธีการถดถอยแบบขั้นตอน

ธันยากร ต้นชลขันธ์ (2538) ได้ทำการศึกษาเปรียบเทียบการประมาณค่าสัมประสิทธิ์การถดถอยพหุเมื่อเกิดพหุสัมพันธ์ระหว่างตัวแปรอิสระ โดยการเปรียบเทียบวิธีกำลังสองน้อยที่สุด (OLS) วิธีการถดถอยแบบริดจ์ (RR) และวิธีที่ใช้หลักการของริดจ์และสไตน์ (RS)เกณฑ์การเปรียบเทียบ คือ เปอร์เซนต์อัตราส่วนของค่าเฉลี่ยความคลาดเคลื่อนกำลังสอง สำหรับการแจกแจงของค่าความคลาดเคลื่อนที่ศึกษามีดังนี้ การแจกแจงปกติที่มีค่าเฉลี่ยเท่ากับ 1และค่าความเบี่ยงเบนมาตรฐานเทากับ 0.5, 0.10 และ 0.15 การแจกแจงปกติปลอมปนที่มีสเกลแฟคเตอร์เท่ากับ3 และ10 เปอร์เซ็นต์การปลอมปนเท่ากับ5 และ 10 และการแจกแจงลอกนอร์มอลซึ่งมีค่าเฉลี่ยเท่ากับ 1ความแปรปรวนเท่ากับ 0.05, 0.30 และ 0.70 กล่าวคือค่าสัมประสิทธิ์การแปรผัน (C.V.) เท่ากับ 22%, 59% และ 100% ตามลำดับค่าสัมประสิทธิ์การถดถอยพหุที่ใช้เป็นค่าเริ่มต้นได้จากเวกเตอร์เฉพาะซึ่งสอดคล้องกับค่าเฉพาะที่มีค่าต่ำที่สุด ในกรณีที่จำนวนตัวแปรอิสระเท่ากับ 3 กำหนดระดับความสัมพันธ์ระหว่างตัวแปรอิสระเป็น 3 ระดับคือระดับต่ำเท่ากับ0.10 และ 0.30 ระดับปานกลางเท่ากับ0.50 และ 0.70 และระดับสูงเท่ากับ0.90 และ 0.99 ส่วนกรณีที่จำนวนตัวแปรอิสระเท่ากับ 5 ได้กำหนดระดับความสัมพันธ์ระหว่างตัวแปรอิสระเป็น 3 ระดับคือ ระดับต่ำเท่ากับ 0.10 และ 0.01, 0.30 และ 0.30ระดับปานกลางเท่ากับ0.05 และ 0.05,0.70 และ0.70และระดับสูงเท่ากับ0.90 และ 0.90, 0.99 และ 0.99ขนาดตัวอย่างที่ศึกษา = 30,50และ 100 ในการวิจัยได้จำลองข้อมูลที่ใช้ในตัวแบบด้วยเครื่องคอมพิวเตอร์โดยใช้เทคนิคมอนติคาร์โลและกระทำซ้ำ 500 ครั้ง ในแต่ละกรณี สรุปผลการเปรียบเทียบเปอร์เซ็นต์อัตราส่วนของค่าเฉลี่ยความคลาดเคลื่อนกำลังสองได้ดังนี้ 1.

กรณีที่ความคลาดเคลื่อนมีการแจกแจงปกติและปกติปลอมปน พบว่าสำหรับทุกการแจกแจงส่วนใหญ่วิธีการถดถอยแบบริดจ์จะให้ผลดีที่สุด ส่วนวิธีหลักการของริดจ์และสไตน์จะให้ผลดีที่สุดในกรณีที่ความคลาดเคลื่อนมีการแจกแจงปกติเมื่อจำนวนตัวแปรอิสระเท่ากับ 3 โดยที่ระดับพหุสัมพันธ์มีค่าอยู่ในระดับต่ำมีค่าเท่ากับ0.10 และ 0.30และระดับปานกลางมีค่าเท่ากับ0.05และให้ผลดีที่สุดสำหรับจำนวนตัวแปรอิสระเท่ากับ 5 เมื่อระดับพหุสัมพันธ์มีค่าอยู่ในระดับต่ำคือ 0.10 และ 0.30และระดับปานกลางคือ 0.5ซึ่งในกรณีนี้ σ เท่ากับ 0.05 ขนาดตัวอย่างเท่ากับ 100สเกลแฟคเตอร์เท่ากับ 3 และเปอร์เซ็นต์การปลอมปนเท่ากับ 5 และ 10 ค่าเฉลี่ยความคลาดเคลื่อนกำลังสองแปรผันตามปัจจัยต่อไปนี้จากมากไปหาน้อย จำนวนตัวแปรอิสระ สเกลแฟคเตอร์เปอร์เซนต์การปลอมปน ระดับความสัมพันธ์ และความแปรปรวน แต่แปรผกผันกับขนาดตัวอย่าง 2.กรณีที่ความคลาดเคลื่อนมีการแจกแจงลอกนอร์มอลพบว่าวิธีการถดถอยแบบริดจ์ให้ผลดีที่สุดทุกกรณีเมื่อ C.V. = 22%, 59% และ 100% เพราะว่าวิธี RRจะให้ผลดีเมื่อ C.V. มีค่าเพิ่มขึ้น ค่าเฉลี่ยความคลาดเคลื่อนกำลังสองแปรผันตามปัจจัยต่อไปนี้จากมากไปหาน้อย จำนวนตัวแปรอิสระ ระดับความสัมพันธ์และความแปรปรวน แต่แปรผกผันกับขนาดตัวอย่าง

Siary and Berthiau (1997) ได้ทำการศึกษาวิธีการในการหาค่าใกล้เคียงกับค่าอุตตมะของฟังก์ชันที่มีตัวแปรเป็นแบบต่อเนื่อง โดยใช้การค้นหาคำตอบแบบต้องห้ามและศึกษาอิทธิพลของพารามิเตอร์หลักในการลู่เข้าไปยังค่าที่เหมาะสม ซึ่งวัดประสิทธิภาพของการค้นหาโดยใช้ทดสอบกับฟังก์ชันที่นิยมใช้ในการหาค่าน้อยที่สุด ประกอบด้วย ฟังก์ชันโกลสไตน์-ไพรซ์(Goldstein-Price) ที่ประกอบด้วยตัวแปรอิสระจำนวน 2 ตัว ฟังก์ชันฮาร์ทแมน (Hartmann) ที่ประกอบด้วยตัวแปรอิสระจำนวน 3 ตัวแปร และฟังก์ชันโรเซนบร็อก (Rosenbrock)ที่ประกอบด้วยตัวแปรอิสระจำนวน n ตัวแปร ในงานวิจัยนี้กำหนดให้ n= 2, 5 และ 10 โดยแบ่งการศึกษาออกเป็น 2 ส่วน คือ ส่วนที่ 1 เปรียบเทียบผลลัพธ์ในการค้นหาคำตอบแบบต้องห้ามกับผลลัพธ์ที่ได้จากการค้นหาคำตอบโดยใช้วิธี Simulated Annealing (SA) ส่วนที่ 2 ทำการวิเคราะห์ความไวของการค้นหาคำตอบแบบต้องห้ามโดยพิจารณาจากพารามิเตอร์หลักผลการศึกษาพบว่า ในส่วนที่ 1 จากการทำซ้ำจำนวน 100 ครั้ง อัตราของการประสบความสำเร็จในการหาค่าน้อยที่สุดจะสูงและผลลัพธ์ที่ได้ลู่เข้าไปยังฟังก์ชันเป้าหมาย ในกรณีที่ฟังก์ชันมีจำนวนตัวแปรอิสระ 2 หรือ 3 ตัวแปร นอกจากนี้จำนวนครั้งที่ใช้ในการคำนวณฟังก์ชันกรณีการค้นหาคำตอบแบบ SA สูงกว่าการค้นหาคำตอบแบบต้องห้ามประมาณ 30 เท่า แต่เมื่อมีจำนวนตัวแปรเพิ่มมากขึ้น กรณีจำนวน 10 ตัวแปรพบว่าจำนวนครั้งที่ใช้ในการคำนวณฟังก์ชันโดยค้นหาคำตอบแบบต้องห้ามจะสูงกว่าแบบ SA ประมาณ 20 เท่า ผลการศึกษาในส่วนที่ 2 จากการทำการทดสอบโดยเปลี่ยนค่ารัศมี (Radius : ε) ให้มีค่าระหว่าง 0.001 กับ 0.2 พบว่าค่าที่เหมาะสมของพารามิเตอร์ขึ้นอยู่กับแต่ละฟังก์ชัน แต่ความถูกต้องของผลลัพธ์

และความเร็วในการลู่เข้ามีความสัมพันธ์ในทางตรงกันข้ามกับค่ารัศมี ตัวอย่างเช่น ในกรณี $\varepsilon > 0.05$ จะใช้เวลาในการประมวลผล น้อยกว่าในกรณี $\varepsilon < 0.005$ ซึ่งค่าที่เหมาะสมที่สุดของรัศมี คือ มีค่าประมาณ 0.01

Drezner and George (1999) ได้เสนอวิธีการคัดเลือกตัวแปรอิสระในการวิเคราะห์การถดถอยเชิงเส้นพหุโดยวิธีการค้นหาแบบต้องห้ามเปรียบเทียบกับวิธีการถดถอยแบบขั้นตอนและวิธีการปรับปรุงค่า R^2 สูงสุด (Maximum R^2 Improvement)เกณฑ์ที่ใช้ในการตัดสินใจ คือ อัตราส่วนระหว่างนัยสำคัญของแต่ละวิธีการกับนัยสำคัญของวิธีการถดถอยที่เป็นไปได้ทั้งหมด โดยถือว่าค่าอัตราส่วนที่มีผลลัพธ์เป็น 1 เป็นอัตราส่วนที่เหมาะสมในการคัดเลือกตัวแปรอิสระ และใช้ค่าสัมประสิทธิ์ตัวกำหนด (R^2) เป็นเกณฑ์ในการคัดเลือกตัวแปรอิสระ ทำการศึกษาที่ขนาดตัวอย่างเท่ากับ 50 และ 150 จำนวนตัวแปรอิสระแบ่งเป็น 2 กลุ่ม คือ ขนาดกลางมีจำนวนตัวแปรอิสระ17 ถึง 26ตัวแปรและขนาดใหญ่มีจำนวนตัวแปรอิสระ30 กับ 80 ตัวแปรตามลำดับ ผลการศึกษาพบว่ากรณีกลุ่มตัวแปรอิสระขนาดกลางวิธีการค้นหาแบบต้องห้ามจะให้ค่าอัตราส่วนเป็น 1 ทุกกรณี วิธีการถดถอยแบบขั้นตอนและวิธีการปรับปรุงค่า R^2 สูงสุด ให้ผลที่เหมือนกัน คือ ค่าอัตราส่วนเป็น 1 เพียง 3 กรณี ในกรณีจำนวนตัวแปรอิสระมีจำนวนมาก วิธีการค้นหาแบบต้องห้ามจะให้ผลดีที่สุด คือ ค่าอัตราส่วนเป็น 1 ทุกกรณี แต่วิธีการถดถอยแบบขั้นตอนจะให้ค่าอัตราส่วนเป็น 1 เพียง 2 กรณี และวิธีการปรับปรุงค่า R^2 สูงสุด ให้ค่าอัตราส่วนเป็น 1 เพียง 5 กรณีเท่านั้น ผลการศึกษาสรุปได้ว่าวิธีการค้นหาแบบต้องห้ามมีประสิทธิภาพในการคัดเลือกตัวแปรอิสระได้ดีที่สุด

Pasha(2002) ได้ทำการศึกษาเปรียบเทียบวิธีการคัดเลือกตัวแปรอิสระในการวิเคราะห์การถดถอยเชิงเส้นพหุ โดยศึกษาวิธีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบโดยวิธีการเลือกแบบไปข้างหน้า (Forward Selection) วิธีตัดออกแบบถอยหลัง (Backward Elimination) และวิธีการถดถอยแบบขั้นตอน (Stepwise Regression) ศึกษาจากข้อมูลจริงที่มีจำนวนตัวแปรอิสระ 5 ตัวแปร และขนาดตัวอย่างเท่ากับ 28 ผลการศึกษาพบว่าวิธีการคัดเลือกตัวแปรด้วยวิธีเลือกแบบไปข้างหน้าและวิธีตัดออกแบบถอยหลัง ให้ผลลัพธ์ในการคัดเลือกตัวแปรอิสระเหมือนกัน คือได้ตัวแปรอิสระ X_2, X_4 และ X_5 ในขณะที่วิธีการถดถอยแบบขั้นตอนคัดเลือกตัวแปรอิสระได้เพียง X_2 และ X_4 ซึ่งผลการทดสอบโดยใช้สถิติ t พบว่า X_5 ไม่มีนัยสำคัญทางสถิติ แสดงให้เห็นว่าวิธีการถดถอยแบบขั้นตอนคัดเลือกตัวแปรอิสระได้เหมาะสมกว่าอีก 2 วิธี สำหรับข้อมูลชุดดังกล่าว

เบ็ญจวรรณ ชัยกิจ (2545) ได้ทำการศึกษาเปรียบเทียบตัวประมาณริดจ์สำหรับการถดถอยแบบริดจ์ ซึ่งใช้วิธีการหาค่าพารามิเตอร์ริดจ์3 วิธี คือ วิธีโฮเอิร์ล เคนนาร์ด(Hoerl Kennard;HK) วิธีค้นหาข้อมูลแบบลำดับ (SEQ) และวิธีเบส์โดยกระทำภายใต้เงื่อนไขของการแจกแจงของความคลาดเคลื่อนเป็นแบบปกติ ระดับส่วนเบี่ยงเบนมาตรฐานของความคลาดเคลื่อนเท่ากับ 0.05, 0.5, 1

และ 3 ขนาดตัวอย่าง 10, 30, 50 และ 100และระดับความสัมพันธ์ของตัวแปรอิสระเท่ากับ 0.1, 0.3, 0.5, 0.7, 0.9 และ 0.99โดยพิจารณา 2 กรณี กล่าวคือ กรณีแรกตัวแปรอิสระมีความสัมพันธ์กัน 1 กลุ่ม (จำนวนตัวแปรอิสระเท่ากับ 3 และ 5 ตัวแปร) และกรณีที่สองตัวแปรอิสระมีความสัมพันธ์กัน 2 กลุ่ม ตัวแปรอิสระในกลุ่มที่ 1 (จำนวนตัวแปรอิสระเท่ากับ 2 หรือ 3 ตัวแปร) และกลุ่มที่ 2 (จำนวนตัวแปรอิสระเท่ากับ 2 ตัวแปร) ข้อมูลที่ใช้ในการวิจัยได้จากการจำลองโดยเทคนิคมอนติคาร์โล ทำซ้ำ 1,000 ครั้งในแต่ละสถานการณ์ที่กำหนด เพื่อเปรียบเทียบค่าเฉลี่ยของความคลาดเคลื่อนยกกำลังสอง (AMSE) ของตัวประมาณริดจ์แบบต่าง ๆ ข้างต้น ซึ่งผลการวิจัยสรุปได้ดังนี้ 1) กรณีที่ 1 ตัวแปรอิสระมีความสัมพันธ์กัน 1 กลุ่ม เมื่อจำนวนตัวแปรอิสระเท่ากับ 3 โดยทั่วไปวิธีเบส์ให้ค่า AMSE น้อยที่สุดทุกระดับความสัมพันธ์และทุกขนาดตัวอย่าง โดยที่ส่วนเบี่ยงเบนมาตรฐานของความคลาดเคลื่อนมีค่าเท่ากับ 0.05 0.5 และ 1 ส่วนวิธี SEQ ให้ค่า AMSE น้อยที่สุดโดยทั่วไปเมื่อระดับความสัมพันธ์มีค่าสูงมาก ส่วนเบี่ยงเบนมาตรฐานของความคลาดเคลื่อนมีค่าเท่ากับ 3 และจำนวนตัวแปรอิสระเท่ากับ 5โดยทั่วไปวิธีเบส์ให้ค่า AMSE น้อยที่สุดทุกระดับความสัมพันธ์และทุกขนาดตัวอย่าง โดยที่ส่วนเบี่ยงเบนมาตรฐานของความคลาดเคลื่อนมีค่าเท่ากับ 0.05 และ 0.5 และถ้าส่วนเบี่ยงเบนมาตรฐานของความคลาดเคลื่อนมีค่าเท่ากับ 1 วิธีเบส์ให้ค่า AMSE น้อยที่สุดเมื่อขนาดตัวอย่างมีค่าเท่ากับ 30, 50 และ100 ส่วนวิธี SEQ ให้ค่า AMSE น้อยที่สุดโดยทั่วไปเมื่อระดับความสัมพันธ์มีค่าสูงมากและส่วนเบี่ยงเบนมาตรฐานของความคลาดเคลื่อนมีค่าเท่ากับ 3 2) กรณีที่ 2 ตัวแปรอิสระมีความสัมพันธ์กัน 2 กลุ่มโดยทั่วไปวิธีเบส์ให้ค่า AMSE น้อยที่สุด เมื่อระดับความสัมพันธ์ของตัวแปรอิสระในแต่ละกลุ่มมีค่าต่ำ และปานกลาง และทุกขนาดตัวอย่าง โดยที่ส่วนเบี่ยงเบนมาตรฐานของความคลาดเคลื่อนมีค่าเท่ากับ 0.05 แต่ถ้าระดับความสัมพันธ์ของตัวแปรอิสระในกลุ่มที่ 1 หรือกลุ่มที่ 2 มีค่าสูง และส่วนเบี่ยงเบนมาตรฐานของความคลาดเคลื่อนมีค่าเท่ากับ 0.5 และ 1 วิธีเบส์ให้ค่า AMSE น้อยที่สุดเมื่อขนาดตัวอย่างมีค่าเท่ากับ 30, 50 และ 100 ส่วนวิธี SEQ ให้ค่า AMSE น้อยที่สุด โดยทั่วไปเมื่อระดับความสัมพันธ์มีค่าสูงมาก และส่วนเบี่ยงเบนมาตรฐานของความคลาดเคลื่อนมีค่าเท่ากับ 3 ค่า AMSE จะแปรผันตามจำนวนตัวแปรอิสระที่มีความสัมพันธ์กัน ระดับความสัมพันธ์ของตัวแปรอิสระ และส่วนเบี่ยงเบนมาตรฐานของความคลาดเคลื่อน โดยเรียงลำดับจากมากไปน้อย แต่ค่า AMSE จะแปรผกผันกับจำนวนตัวแปรอิสระและขนาดตัวอย่าง โดยเรียงลำดับจากมากไปน้อย

บุญจิรา มากอ้น (2545) ได้ทำการศึกษาโดยเปรียบเทียบเกณฑ์การคัดเลือกตัวแบบในการวิเคราะห์การถดถอยเชิงเส้นแบบไม่ติดกลุ่ม เกณฑ์การคัดเลือกตัวแบบสำหรับการวิจัยในครั้งนี้ คือ เกณฑ์การคัดเลือกตัวแบบโดยเกณฑ์สารสนเทศอาไคเคะ(AIC) และเกณฑ์สารสนเทศของเบส์ (BIC) การคัดเลือกตัวแบบของทั้งสองเกณฑ์จะเลือกตัวแบบที่ให้ค่าของทั้งสองเกณฑ์ต่ำที่สุด ข้อมูล

ที่ใช้ในการวิจัยครั้งนี้มาจากการจำลองด้วยเทคนิคมอนติคาร์โลโดยโปรแกรม S-PLUS 2000 ใช้ตัวแปรอิสระเริ่มต้นเป็น 2 3 และ 4 ตัวแปร ค่าความคลาดเคลื่อนมีการแจกแจงแบบปกติที่มีค่าเฉลี่ยเป็น 0 และส่วนเบี่ยงเบนมาตรฐานเป็น 1, 5, 10 และ 15 ขนาดตัวอย่างเป็น 25, 50, 75 และ 100 ระดับความสัมพันธ์ของตัวแปรอิสระ คือ 0 0.05 และ 0.99 ค่าสัดส่วนของการเลือกผิดกำหนดเป็น 1% 5% และ 10% ระดับนัยสำคัญที่ใช้ คือ 0.01 และ 0.05 เกณฑ์ที่ใช้ในการเปรียบเทียบ คือ ค่าสัดส่วนของการคัดเลือกตัวแบบผิด ผลการศึกษาสรุปได้ดังนี้ 1) เมื่อจำนวนตัวแปรอิสระเริ่มต้นเป็น2ตัวแปรเกณฑ์ AIC และ BIC มีค่าสัดส่วนของการคัดเลือกตัวแบบผิดเป็น 0 ทุกระดับความสัมพันธ์ของตัวแปรอิสระขนาดตัวอย่างและส่วนเบี่ยงเบนมาตรฐาน2)เมื่อจำนวนตัวแปรอิสระเริ่มต้นเป็น 3 และ 4 ตัวแปร เกณฑ์ AIC มีค่าสัดส่วนของการเลือกผิดต่ำกว่าเกณฑ์ BIC สำหรับทุกสถานการณ์ โดยสรุปเกณฑ์ AIC และ BIC จะมีค่าสัดส่วนของการเลือกผิดสูงขึ้น เมื่อจำนวนตัวแปรอิสระมีจำนวนมากขึ้นและมีค่าลดลงเมื่อขนาดตัวอย่างมีจำนวนมากขึ้น

วรรัตน์ ราชกิจจา (2547) ได้ทำการศึกษาเปรียบเทียบวิธีการประมาณค่าสัมประสิทธิ์การถดถอยพหุ เมื่อเกิดพหุสัมพันธ์ระหว่างตัวแปรอิสระ โดยเปรียบเทียบวิธีการถดถอยเอ็มพิริคัลเบส์ริดจ์ (EB)วิธีการถดถอยเอ็มพิริคัลเบส์ริดจ์แบบลำดับขั้น (HB) และวิธีการถดถอยแบบเอ็มพิริคัลเบส์ริดจ์แบบแบ่งส่วน (DB) เกณฑ์ที่ใช้ในการตัดสินใจ คือ ค่าเฉลี่ยความคลาดเคลื่อนกำลังสองเฉลี่ย (Average mean Square Error (AMSE)) และเกณฑ์ที่ใช้ในการเปรียบเทียบประสิทธิภาพของวิธีการทั้ง 3 วิธีจะใช้ค่าอัตราส่วนผลต่างของค่าเฉลี่ยความคลาดเคลื่อนกำลังสองเฉลี่ย (Ratio of Different Average Mean Square Error (RDAMSE)) ซึ่งได้ศึกษาในกรณีที่จำนวนตัวแปรอิสระเท่ากับ 7 และ 9 จำนวนค่าลักษณะเฉพาะที่เหลืออยู่เท่ากับ 3 และ 5 ขนาดตัวอย่างเท่ากับ 30, 50, 70และ 100 ความคลาดเคลื่อนมีการแจกแจงปกติด้วยค่าเฉลี่ยเท่ากับ 0 ส่วนเบี่ยงเบนมาตรฐานเท่ากับ 1,5 และ 10 ตามลำดับ ระดับความสัมพันธ์ระหว่างตัวแปรอิสระมี 3 ระดับ คือ ระดับต่ำ (ρ =0.10, 0.30) ระดับปานกลาง (ρ =0.50, 0.70) และระดับสูง (ρ =0.80, 0.90) ในการวิจัยครั้งนี้ได้ทำการจำลองด้วยเทคนิคมอนติคาร์โลซึ่งกระทำซ้ำ 1,000 ครั้งในแต่ละสถานการณ์ ผลของการวิจัยสามารถสรุปได้ดังนี้ กรณีที่ 1จำนวนตัวแปรอิสระเท่ากับ 7 ในทุกรณี วิธี DB ให้ค่า AMSE น้อยที่สุด รองลงมาคือวิธี HB และ วิธี EB ตามลำดับ โดยที่ค่า AMSE มีแนวโน้มลดลงเมื่อขนาดตัวอย่างเพิ่มขึ้น และมีแนวโน้มเพิ่มขึ้นเมื่อระดับความสัมพันธ์ ส่วนเบี่ยงเบนมาตรฐานและจำนวนค่าลักษณะเฉพาะที่เหลืออยู่เพิ่มขึ้น ส่วนค่า RDAMSE มีแนวโน้มลดลงเมื่อขนาดตัวอย่างเพิ่มขึ้นและมีแนวโน้มเพิ่มขึ้น เมื่อระดับความสัมพันธ์ ส่วนเบี่ยงเบนมาตรฐาน และจำนวนค่าลักษณะเฉพาะที่เหลืออยู่เพิ่มขึ้น กรณีที่ 2จำนวนตัวแปรอิสระเท่ากับ 9 ให้ผลลัพธ์เช่นเดียวกันกับในกรณีที่ 1 ค่า AMSE แปรผันตามปัจจัย

ต่อไปนี้จากมากไปน้อยคือ ระดับความสัมพันธ์ ส่วนเบี่ยงเบนมาตรฐานจำนวนตัวแปรอิสระ และจำนวนค่าลักษณะเฉพาะที่เหลืออยู่ แต่แปรผกผันกับขนาดตัวอย่าง

Eksioglu, Demirer and Capar (2005) ได้ศึกษาการคัดเลือกตัวแบบที่เหมาะสมในการวิเคราะห์การถดถอยเชิงเส้นพหุโดยใช้วิธีการหาค่าใกล้เคียงกับค่าอุตตมะ (Optimal) โดยใช้วิธีการโปรแกรมเชิงคณิตศาสตร์ (Mathematical Programming) วิธีการลากรานจ์(Lagrangian Relaxation) และวิธีการ GRASP(Greedy Randomized Adaptive Search Procedure) ซึ่งสองวิธีหลังเป็นวิธีการแบบฮิวริสติกส์(Heuristic) โดยนำผลลัพธ์ที่ได้จากการคัดเลือกจากทั้งสามวิธีการมาเปรียบเทียบกับวิธีการถดถอยแบบขั้นตอนวิธีตัดออกแบบถอยหลัง และวิธีเลือกแบบไปข้างหน้า ซึ่งเป็นวิธีการที่มีในโปรแกรมสำเร็จรูปทางสถิติโดยเกณฑ์ที่นำมาใช้ในการเปรียบเทียบคือ ค่าสัมประสิทธิ์ตัวกำหนดแบบปรับ (R_a^2) ข้อมูลที่ใช้ในการวิจัยนี้นำมาจากหนังสือ Introduction to Linear Regression Analysis ของ Montgomery and Peck (1992) จำนวน 12 ชุดข้อมูล ชุดข้อมูลนี้ประกอบไปด้วยตัวแปรอิสระทั้งหมด 7 ตัวแปร ผลการศึกษาพบว่าวิธี GRASP วิธีลากรานจ์ (Lagrangian)และวิธีตัดออกแบบถอยหลังคัดเลือกตัวแบบได้ดีใน 5 ชุดข้อมูล ในขณะที่วิธีเลือกแบบไปข้างหน้าและวิธีการถดถอยแบบขั้นตอนคัดเลือกตัวแบบได้ดีเพียงชุดข้อมูลเดียว สำหรับวิธีการหาค่าใกล้เคียงกับค่าอุตตมะคัดเลือกตัวแบบได้ดีใน 6 ชุดข้อมูล นอกจากนี้ยังพบว่าวิธีการหาค่าใกล้เคียงกับค่าอุตตมะใช้เวลาในการประมวลผลมากที่สุดเมื่อเปรียบเทียบกับวิธีอื่นๆ แต่ผลลัพธ์เป็นชุดของตัวแบบที่ให้เป็นทางเลือกในการนำไปพิจารณาใช้ในสถานการณ์จริง ในขณะที่วิธีการที่มีในโปรแกรมสำเร็จรูปทางสถิติจะให้ผลลัพธ์ออกมาเป็นตัวแบบที่ดีที่สุดเพียงตัวแบบเดียว

บุษยา ปภาพจน์ (2548) ได้ทำการศึกษาเปรียบเทียบวิธีการคัดเลือกตัวแบบที่เหมาะสมที่สุดในการวิเคราะห์การถดถอยเชิงเส้นพหุโดยเปรียบเทียบการคัดเลือกตัวแบบ 4 วิธี คือ วิธีการค้นหาแบบต้องห้าม (Tabu Search) วิธีการถดถอยแบบขั้นตอน (SR) วิธีเลือกตัวแปรอิสระแบบไปข้างหน้า (FS) วิธีการปรับปรุงค่า R^2 สูงสุด(MAXR)โดยแบ่งการศึกษาออกเป็น 2 ส่วน คือ ส่วนที่ 1 เปรียบเทียบประสิทธิภาพของวิธีการคัดเลือกตัวแบบเทียบกับวิธีตัวแบบการถดถอยที่เป็นไปได้ทั้งหมดโดยใช้อัตราส่วนค่านัยสำคัญเฉลี่ยของตัวแบบที่คัดเลือกได้ทั้ง 4 วิธีกับวิธีการถดถอยที่เป็นไปได้ทั้งหมด และส่วนที่ 2 เปรียบเทียบประสิทธิภาพในการพยากรณ์จากค่าเฉลี่ยของค่าความคลาดเคลื่อนกำลังสองเฉลี่ย (AMSE) โดยในการศึกษากำหนดให้ค่าความคลาดเคลื่อนเป็นการแจกแจงปกติที่มีค่าเฉลี่ยเท่ากับ 0 ความแปรปรวนเท่ากับ 25 ข้อมูลที่ใช้ทุกตัวแปรเป็นอิสระต่อกัน ดังนั้นข้อมูลที่ใช้จึงไม่เกิดปัญหาสหสัมพันธ์เชิงเส้นพหุแต่อย่างใด ขนาดของตัวอย่างที่ศึกษามี 3 ระดับ คือ กลุ่มที่มีตัวอย่างขนาดเล็ก (n=20,30) กลุ่มที่มีตัวอย่างขนาดกลาง (n=50,100) และกลุ่มที่มีตัวอย่างขนาดใหญ่ (n=150,200) จำนวนตัวแปรอิสระที่ใช้แบ่งเป็น 2 กลุ่ม คือ กลุ่มตัวแปรอิสระ

ขนาดเล็กเท่ากับ 5, 7 และ 9 กลุ่มตัวแปรอิสระขนาดกลางเท่ากับ 11, 15, 17 และ 20 ข้อมูลที่ใช้ในการวิจัยได้มาจากการจำลองด้วยเทคนิคมอนติคาร์โล และทำการทดลองซ้ำ 500 ครั้ง ภายใต้สถานการณ์ที่กำหนด พบว่ามีผลสรุปดังนี้ ผลการศึกษาในส่วนที่ 1 พบว่าเมื่อกลุ่มตัวอย่างและจำนวนตัวแปรอิสระมีขนาดเล็ก วิธี SR และวิธี FS มีประสิทธิภาพในการคัดเลือกตัวแบบได้ดีที่สุด สำหรับกลุ่มตัวอย่างและจำนวนตัวแปรอิสระขนาดกลาง วิธีการค้นหาแบบต้องห้ามมีประสิทธิภาพในการคัดเลือกตัวแบบได้ดีที่สุด และเมื่อตัวอย่างมีขนาดใหญ่วิธีการค้นหาแบบต้องห้ามมีประสิทธิภาพในการคัดเลือกตัวแบบได้ดีที่สุดทุกกลุ่มตัวแปรอิสระ ส่วนผลการศึกษาในส่วนที่ 2 จากค่า AMSE พบว่าเมื่อกลุ่มตัวอย่างและจำนวนตัวแปรอิสระมีขนาดเล็ก วิธี MAXR และวิธีการค้นหาแบบต้องห้ามมีประสิทธิภาพในการพยากรณ์ที่ดีที่สุดในทุกกลุ่มตัวแปรอิสระ สำหรับวิธี MAXR ถึงแม้ว่าจะไม่ได้ให้อัตราส่วนค่านัยสำคัญเฉลี่ยและค่า AMSE ต่ำที่สุด เท่ากับวิธีการค้นหาแบบต้องห้ามแต่สามารถให้ค่าที่ใกล้เคียงกับค่าต่ำสุด อีกทั้งวิธีนี้มีอยู่ในโปรแกรมสำเร็จรูปทางสถิติ จึงเป็นวิธีการที่เหมาะสมที่จะนำมาใช้ในการคัดเลือกตัวแบบในการวิเคราะห์การถดถอยเชิงเส้นพหุ

ภัทรสุดา สุดแสน (2548) ได้ทำการศึกษาโดยเปรียบเทียบเกณฑ์การคัดเลือกตัวแบบการถดถอยเชิงเส้นเมื่อตัวอย่างมีขนาดเล็กโดยจะเปรียบเทียบเกณฑ์การคัดเลือกตัวแบบการถดถอย 4 เกณฑ์ ได้แก่เกณฑ์สารสนเทศของอาไคเคะที่ปรับแก้ (Corrected Akaike's Information Criterion (AIC)) เกณฑ์สารสนเทศของชวาร์ซที่ปรับแก้ (Corrected Schwarz's Information Criterion (SIC)) เกณฑ์สารสนเทศของแฮนแนนและควินน์ที่ปรับแก้ (Corrected Hannan and Quinn's Information Criterion (HQ)) และเกณฑ์สารสนเทศของคูลส์แบล็คที่ปรับแก้ (Corrected Kullback's Information Criterion (KIC)) โดยใช้เกณฑ์การตัดสินใจคือค่าเฉลี่ยของค่าคลาดเคลื่อนกำลังสองเฉลี่ย (Average of Mean Squares Error (AMSE)) และใช้อัตราส่วนผลต่างของค่าเฉลี่ยของค่าคลาดเคลื่อนกำลังสองเฉลี่ย (Ratio of Different Average Mean Squares Error (RDAMSE)) เพื่อเปรียบเทียบประสิทธิภาพของเกณฑ์การคัดเลือกตัวแบบทั้ง 4 เกณฑ์การแจกแจงของความคลาดเคลื่อนสุ่มที่ศึกษาคือ การแจกแจงแบบปกติที่มีค่าเฉลี่ย 0 และความแปรปรวนเป็น 1, 2, 4, 8 และ 16 ขนาดตัวอย่างที่ใช้คือ 12, 15, 18, 21, 24, 27 และ 30 และจำนวนตัวแปรอิสระในตัวแบบการถดถอยคือ 3, 5 และ 7ตัวแปร ข้อมูลที่ใช้ในการวิจัยใช้โปรแกรมคอมพิวเตอร์จำลองด้วยเทคนิคมอนติคาร์โลกระทำซ้ำ 1,000 ครั้ง ในแต่ละสถานการณ์ ซึ่งผลการวิจัยสรุปได้ดังนี้ปัจจัยที่มีผลต่อค่าเฉลี่ยของค่าคลาดเคลื่อนกำลังสองเฉลี่ย (AMSE) ของทุกเกณฑ์การคัดเลือกตัวแบบได้แก่ จำนวนตัวแปรอิสระและความแปรปรวนของความคลาดเคลื่อนสุ่ม และขนาดตัวอย่าง โดยที่ AMSE แปรผันตามจำนวนตัวแปรอิสระและความแปรปรวนของความคลาดเคลื่อนสุ่มแต่ AMSE แปรผกผันกับขนาดตัวอย่างปัจจัย

ดังกล่าวข้างต้นส่งผลกระทบกับเกณฑ์การคัดเลือกตัวแบบดังกรณีต่อไปนี้ 1)กรณีขนาดตัวอย่างเท่ากับ 12 ถึง 15 เกณฑ์การคัดเลือกตัวแบบที่ดีที่สุดคือ เกณฑ์ HQ รองลงมาคือ เกณฑ์ AIC SIC และ KIC ตามลำดับสำหรับทุกระดับความแปรปรวนของความคลาดเคลื่อนสุ่มและจำนวนตัวแปรอิสระ 2)กรณีขนาดตัวอย่างเท่ากับ 18 เกณฑ์การคัดเลือกตัวแบบที่ดีที่สุดมีสองเกณฑ์คือ เกณฑ์ HQ และเกณฑ์ AIC และรองลงมาคือเกณฑ์ KIC และ SIC มีค่า AMSE ต่ำกว่า SIC เพียงเล็กน้อยสำหรับทุกระดับความแปรปรวนของความคลาดเคลื่อนสุ่มและจำนวนตัวแปรอิสระ 3)กรณีขนาดตัวอย่างเท่ากับ 21 ถึง 30 เกณฑ์การคัดเลือกตัวแบบที่ให้ค่า AMSE น้อยที่สุดคือ เกณฑ์ AIC รองลงมาคือเกณฑ์ HQ KIC และ SIC ตามลำดับสำหรับทุกระดับความแปรปรวนของความคลาดเคลื่อนสุ่มและจำนวนตัวแปรอิสระเมื่อพิจารณาค่า RDAMSE พบว่าประสิทธิภาพในการคัดเลือกตัวแบบของเกณฑ์การคัดเลือกตัวแบบจะแบ่งออกเป็นสองกลุ่มอย่างเห็นได้ชัดคือกลุ่มที่ 1 ได้แก่ เกณฑ์ HQ และเกณฑ์ AIC มีประสิทธิภาพใกล้เคียงกันและดีที่สุดสำหรับทุกขนาดตัวอย่างและความแปรปรวนของความคลาดเคลื่อนสุ่มส่วนกลุ่มที่ 2 ได้แก่ เกณฑ์ KIC และ SIC ซึ่งมีประสิทธิภาพใกล้เคียงกันแต่แตกต่างและต่ำกว่ากลุ่มแรกสำหรับทุกขนาดตัวอย่างและความแปรปรวนของความคลาดเคลื่อนสุ่ม

Cetin and Erar (2006) ได้ทำการศึกษาการคัดเลือกตัวแบบในการวิเคราะห์การถดถอยเชิงเส้นพหุในกรณีที่ข้อมูลมีค่าผิดปกติ (Outlier) โดยเสนอวิธีการคัดเลือกตัวแบบโดยใช้วิธี Robust ซึ่งคำนวณมาจากตัวประมาณของ Andrew และตัวประมาณของ Huber กับHampelและเกณฑ์ที่ใช้ในการพิจารณา คือ C_p AICC และ AICF โดยสองเกณฑ์หลังสร้างโดยใช้วิธี Robust ข้อมูลที่ใช้ในการวิจัยมาจากการจำลองด้วยเทคนิคมอนติคาร์โล ประกอบด้วยตัวแปรอิสระทั้งหมด 5 ตัวแปรที่มีการแจกแจงเอกรูป (Uniform Distribution) กำหนดค่าความคลาดเคลื่อนมีค่าเฉลี่ยเป็น 0 ความแปรปรวนเป็น 0.1 1 และ 10 มีการทำซ้ำทั้งหมด 100 ครั้ง ผลการศึกษาพบว่าในกรณีที่ค่าความคลาดเคลื่อนมีความแปรปรวนเป็น 0.1 กับ 1 และมีตัวแปรอิสระไม่เกิน 3 ตัวแปร เกณฑ์ C_p สามารถคัดเลือกตัวแบบได้ถูกต้องทั้งหมด ในกรณีที่ค่าความคลาดเคลื่อนมีความแปรปรวนเป็น 0.1 และมีตัวแปรอิสระ 3 ตัวแปร เกณฑ์ AICC สามารถคัดเลือกตัวแบบได้ถูกต้องทั้งหมด ในขณะที่มีตัวแปรอิสระ 4 ตัวแปร เกณฑ์ AICF สามารถคัดเลือกตัวแบบได้ถูกต้องทั้งหมดเช่นกัน นอกจากนี้ยังสามารถสรุปได้ว่าเมื่อค่าความแปรปรวนเพิ่มขึ้น ประสิทธิภาพในการคัดเลือกตัวแบบจะลดลง

วลัยทิพย์ บุญญาติศัย(2549) ทำการศึกษาเกณฑ์การคัดเลือกตัวแบบ เพื่อให้ได้ตัวแบบที่เหมาะสมที่สุดภายใต้ข้อมูลชุดเดียวกัน โดยเกณฑ์ที่ใช้ในการพิจารณา คือ ร้อยละของความผิดพลาดโดยเฉลี่ยของค่าพยากรณ์กับค่าสังเกต (MAPE) ซึ่งได้ศึกษาในกรณีที่จำนวนตัวแปรอิสระเท่ากับ 3 และ 4 การแจกแจงของความคลาดเคลื่อนที่ศึกษา คือ การแจกแจงปกติที่มีค่าเฉลี่ยเท่ากับ 0

ส่วนเบี่ยงเบนมาตรฐานเท่ากับ 1, 2, 3 และ 5 ตามลำดับ ขนาดตัวอย่างเท่ากับ 20, 35 และ 50 ตามลำดับ โดยกำหนดระดับความสัมพันธ์ของตัวแปรอิสระเป็น 3 ระดับ คือ ระดับต่ำ ($\rho = 0.05 - 0.35$) ระดับกลาง ($\rho = 0.40 - 0.65$) และระดับสูง ($\rho = 0.70 - 0.95$) ในการวิจัยครั้งนี้ได้จำลองด้วยเทคนิคมอนติคาร์โล ทำซ้ำ 500 ครั้ง ผลสรุปที่ได้มีดังนี้ กรณีที่ 1 จำนวนตัวแปรอิสระเท่ากับ 3 เมื่อขนาดตัวอย่างเท่ากับ 20 แบ่งออกได้อีกเป็น เมื่อส่วนเบี่ยงเบนมาตรฐานเท่ากับ 1 และ 2 และตัวแปรอิสระมีความสัมพันธ์ในระดับกลาง ตัวแบบที่ได้รับการคัดเลือก คือ ตัวแบบลดรูปที่มีการตัดตัวแปรออกหนึ่งตัว แต่ถ้าตัวแปรอิสระมีความสัมพันธ์ในระดับสูง ตัวแบบที่ได้รับการคัดเลือก คือ ตัวแบบลดรูปที่มีการตัดตัวแปรออกสองตัว เมื่อส่วนเบี่ยงเบนมาตรฐานเท่ากับ 3 และ 5 และตัวแปรอิสระมีความสัมพันธ์ในระดับต่ำขึ้นไป ตัวแบบที่ได้รับการคัดเลือก คือ ตัวแบบลดรูปที่มีการตัดตัวแปรออกหนึ่งตัว แต่ถ้าตัวแปรอิสระมีความสัมพันธ์ในระดับสูง ตัวแบบที่ได้รับการคัดเลือก คือ ตัวแบบลดรูปที่มีการตัดตัวแปรออกสองตัว เมื่อขนาดตัวอย่างเท่ากับ 35 แบ่งออกได้อีกเป็น เมื่อส่วนเบี่ยงเบนมาตรฐานเท่ากับ 1 และ 2 และตัวแปรอิสระมีความสัมพันธ์ในระดับกลางขึ้นไป ตัวแบบที่ได้รับการคัดเลือก คือ ตัวแบบลดรูปที่มีการตัดตัวแปรออกหนึ่งตัว เมื่อส่วนเบี่ยงเบนมาตรฐานเท่ากับ 3 และ 5 และตัวแปรอิสระมีความสัมพันธ์ในระดับกลางขึ้นไป ตัวแบบที่ได้รับการคัดเลือก คือ ตัวแบบลดรูปที่มีการตัดตัวแปรออกหนึ่งตัว แต่ถ้าตัวแปรอิสระมีความสัมพันธ์ในระดับสูง ตัวแบบที่ได้รับการคัดเลือก คือ ตัวแบบลดรูปที่มีการตัดตัวแปรออกสองตัว เมื่อขนาดตัวอย่างเท่ากับ 50 ทุกค่าของส่วนเบี่ยงเบนมาตรฐานและตัวแปรอิสระมีความสัมพันธ์ในระดับสูง ตัวแบบที่ได้รับการคัดเลือก คือ ตัวแบบลดรูปที่มีการตัดตัวแปรออกหนึ่งตัว กรณีที่ 2 จำนวนตัวแปรอิสระเท่ากับ 4 เมื่อขนาดตัวอย่างเท่ากับ 20 แบ่งได้เป็น เมื่อส่วนเบี่ยงเบนมาตรฐานเท่ากับ 1 และ 2 ให้ผลลัพธ์เช่นเดียวกับกรณีส่วนเบี่ยงเบนมาตรฐานเท่ากับ 3 และ 5 ในกรณีที่ 1 เมื่อส่วนเบี่ยงเบนมาตรฐานเท่ากับ 3 และ 5 ตัวแปรอิสระมีความสัมพันธ์ในระดับต่ำ ตัวแบบที่ได้รับการคัดเลือก คือ ตัวแบบลดรูปที่มีการตัดตัวแปรออกหนึ่งตัว แต่ถ้าตัวแปรอิสระมีความสัมพันธ์ในระดับสูง ตัวแบบที่ได้รับการคัดเลือก คือ ตัวแบบลดรูปที่มีการตัดตัวแปรอิสระออกสามตัว เมื่อขนาดตัวอย่างเท่ากับ 35 ให้ผลลัพธ์เช่นเดียวกันกับกรณีที่ 1 เมื่อขนาดตัวอย่างเท่ากับ 50 และส่วนเบี่ยงเบนมาตรฐานเท่ากับ 1 และ 2 และตัวแปรอิสระมีความสัมพันธ์ในระดับสูง ตัวแบบที่ได้รับการคัดเลือก คือ ตัวแบบลดรูปที่มีการตัดตัวแปรออกหนึ่งตัว เมื่อส่วนเบี่ยงเบนมาตรฐานเท่ากับ 3 และ 5 และตัวแปรอิสระมีความสัมพันธ์ในระดับกลางขึ้นไป ตัวแบบที่ได้รับการคัดเลือก คือ ตัวแบบลดรูปที่มีการตัดตัวแปรออกหนึ่งตัว ส่วนกรณีอื่นๆ ภายใต้ขอบเขตการวิจัยนี้ ตัวแบบที่ได้รับการคัดเลือก คือ ตัวแบบเต็มรูป โดยสรุปได้ว่าค่า MAPE แปรผันตามปัจจัย อันประกอบด้วย ระดับความสัมพันธ์ ส่วนเบี่ยงเบนมาตรฐาน และจำนวนตัวแปรอิสระ แต่แปรผกผันกับขนาดตัวอย่าง

Kapetanious (2007) ได้ทำการศึกษาการคัดเลือกตัวแบบการวิเคราะห์การถดถอยในกรณีที่เกิดปัญหาทางเศรษฐมิติ (Econometrics) โดยใช้วิธีการ Simulated Annealing (SA) และวิธีการ Genetic Algorithm เปรียบเทียบกับวิธีการแบบเบย์ที่ใช้เกณฑ์ AIC และ BIC การพิจารณาเปรียบเทียบระหว่างแต่ละวิธีใช้ค่า Relative Root Mean Square Forecast Error (RMSFE)ข้อมูลที่ใช้ในการวิจัยเป็นข้อมูลจากธนาคารกลางในประเทศเนเธอร์แลนด์ระหว่างเดือนมกราคม ปี ค.ศ. 1959 จนถึงเดือนธันวาคม ปี ค.ศ. 1998 ข้อมูลดังกล่าวประกอบด้วยตัวแปรอิสระทั้งหมด 147 ตัวแปร เป็นข้อมูลที่ผ่านการแปลงให้มีการแจกแจงปกติที่มีค่าเฉลี่ยเป็น 0 และค่าความแปรปรวนเป็น 1 ผลการศึกษาพบว่าวิธีการ Simulated Annealing และวิธีการ Genetic Algorithm ให้ความแม่นยำในการพยากรณ์สูงกว่าวิธีการแบบเบย์ที่ใช้เกณฑ์ของ AIC และ BIC โดยเฉพาะอย่างยิ่งในกรณีที่มีจำนวนตัวแปรอิสระมาก

Pacheco, Casdo and Nunez (2008) ทำการศึกษาการคัดเลือกตัวแบบที่เหมาะสมในการวิเคราะห์การถดถอยแบบโลจิสติกส์(Logistic Regression)โดยใช้วิธีการค้นหาคำตอบแบบต้องห้าม ซึ่งมีวัตถุประสงค์ในการค้นหาตัวแบบที่มีจำนวนตัวแปรน้อยที่สุดแต่มีประสิทธิภาพในการพยากรณ์สูงที่สุด ทำการเปรียบเทียบผลลัพธ์ที่ได้จากการค้นหาคำตอบแบบต้องห้ามกับวิธีการถดถอยแบบขั้นตอนวิธีตัดออกแบบถอยหลังและวิธีเลือกแบบไปข้างหน้า ซึ่งเป็นวิธีที่สามารถเรียกใช้ได้ในโปรแกรมสำเร็จรูปทางสถิติเกณฑ์ที่นำมาใช้ในการพิจารณา คือ ค่าเฉลี่ยของอัตราส่วนในการประสบความสำเร็จ (Mean Success Ratio) ข้อมูลที่นำมาใช้ในการทดสอบตัวแบบในการวิจัยมีจำนวน10 ชุดซึ่งนำมาจากฐานข้อมูลของมหาวิทยาลัยแคลิฟอร์เนีย โดยแบ่งข้อมูลออกเป็น 2 ส่วน คือ ส่วนแรกใช้ในการเปรียบเทียบการค้นหาคำตอบแบบต้องห้ามกับวิธีข้างต้นที่มีในโปรแกรมสำเร็จรูป ส่วนที่สองใช้ในการสร้างตัวแบบ ผลการศึกษาพบว่าวิธีเลือกแบบไปข้างหน้าและวิธีการถดถอยแบบขั้นตอนดีกว่าวิธีการตัดออกแบบถอยหลัง ส่วนการค้นหาคำตอบแบบต้องห้ามผลลัพธ์ที่ได้มีความเหมาะสมมากกว่าวิธีที่ใช้ในโปรแกรมสำเร็จรูป ยกเว้นในกรณีฐานข้อมูลการเลี้ยงเด็ก นอกจากนี้ยังพบว่าในกรณีผลลัพธ์ที่มีค่าใกล้เคียงกันวิธีการค้นหาคำตอบแบบต้องห้ามใช้จำนวนตัวแปรอิสระน้อยกว่าวิธีที่ใช้ในโปรแกรมสำเร็จรูป

สุรสิทธิ์ ฤทธิ์สมิตชัย (2551) ได้ทำการศึกษาโดยเปรียบเทียบเกณฑ์ที่ใช้ในการคัดเลือกตัวแบบการถดถอยเชิงเส้นโดยทำการเปรียบเทียบภายในระแนงตัวแบบ (Lattice) และเกณฑ์ที่ใช้ในการคัดเลือกตัวแบบมี 2 เกณฑ์ คือ ค่าผลรวมความคลาดเคลื่อนกำลังสอง (Residual Sum of Squares : RRS) และค่าประมาณค่าเฉลี่ยของความผิดพลาดจากการพยากรณ์กำลังสอง (Mean Square Prediction : MSPE)ซึ่งในการศึกษาครั้งนี้มีจำนวนตัวแปรอิสระเป็น 3 และ 4 ตัวแปร ความคลาดเคลื่อนมีการแจกแจงแบบปกติที่มีค่าเฉลี่ยเท่ากับ 0 และส่วนเบี่ยงเบนมาตรฐานเท่ากับ 1, 2, 3 และ5 มีขนาดตัวอย่างเป็น

20,35 และ 50 ข้อมูลที่ใช้ในงานวิจัยนี้มาจากการจำลองโดยคอมพิวเตอร์ โดยใช้เทคนิคมอนติคาร์โล มีการทำซ้ำ 500 ครั้ง ผลการศึกษาพบว่า ในกรณีที่ความสัมพันธ์ระหว่างตัวแปรอิสระทุกคู่มีค่า $\rho < 0.55$ เมื่อใช้เกณฑ์การคัดเลือกตัวแบบเป็นเกณฑ์ RSS หรือ MSPE การคัดเลือกตัวแบบด้วยวิธีการเปรียบเทียบจากทุกตัวแบบที่เป็นไปได้ทั้งหมด (All Possible Models) และการคัดเลือกตัวแบบด้วยวิธีการเปรียบเทียบภายในระแนงตัวแบบให้ผลการคัดเลือกตัวแบบเหมือนกัน คือ ตัวแบบเต็มรูป (Full Model) ส่วนกรณีที่ความสัมพันธ์ระหว่างตัวแปรอิสระบางคู่มีค่า $\rho \geq 0.55$ การคัดเลือกตัวแบบด้วยวิธีการเปรียบเทียบภายในระแนงตัวแบบมีโอกาสเลือกตัวแบบลดรูป (Reduced Model) มากกว่าการคัดเลือกตัวแบบด้วยวิธีการเปรียบเทียบจากทุกตัวแบบที่เป็นไปได้ทั้งหมด นอกจากนั้นการใช้เกณฑ์ MSPE มีโอกาสเลือกได้ตัวแบบลดรูปมากกว่าการใช้เกณฑ์ RSS โดยที่เกณฑ์ RSS แปรผันตามขนาดตัวอย่างและส่วนเบี่ยงเบนมาตรฐานของความคลาดเคลื่อน ส่วนเกณฑ์ MSPE แปรผันตามเฉพาะส่วนเบี่ยงเบนมาตรฐานของความคลาดเคลื่อนเท่านั้น โดยที่ระดับความสัมพันธ์ของตัวแปรอิสระไม่มีผลต่อค่าของเกณฑ์ทั้งสอง

วริดา พลาศรี (2552)ได้ทำการศึกษาเปรียบเทียบวิธีการประมาณค่าสัมประสิทธิ์ความถดถอยเชิงพหุเมื่อเกิดพหุสัมพันธ์ระหว่างตัวแปรอิสระ โดยเปรียบเทียบวิธีความถดถอยแบบริดจ์และวิธีความถดถอยบูตสแตรปแบบริดจ์ เกณฑ์การเปรียบเทียบที่ใช้ สำหรับการประมาณค่าแบบจุดคือค่าเฉลี่ยความคลาดเคลื่อนกำลังสอง และสำหรับการประมาณค่าแบบช่วงคือค่าความยาวเฉลี่ยของช่วงความเชื่อมั่นซึ่งแบ่งเป็น 2 ขั้นตอนย่อย ขั้นแรกพิจารณาค่าสัมประสิทธิ์ความเชื่อมั่นที่ได้จากแต่ละวิธีมีค่าไม่ต่ำกว่าค่าสัมประสิทธิ์ความเชื่อมั่นที่กำหนด ขั้นต่อไปทำการเปรียบเทียบค่าความยาวเฉลี่ยของช่วงความเชื่อมั่นโดยศึกษาภายใต้เงื่อนไขของความคลาดเคลื่อนมีการแจกแจงปกติที่มีค่าเฉลี่ยเท่ากับ 0 และส่วนเบี่ยงเบนมาตรฐานเท่ากับ 1, 5 และ 10 จำนวนตัวแปรอิสระที่ใช้เท่ากับ 2, 3, 4 และ 5 ขนาดตัวอย่างเท่ากับ 15, 30, 50 และ 100 โดยแบ่งระดับความสัมพันธ์ระหว่างตัวแปรอิสระเป็นต่ำ (0.3) ปานกลาง (0.6) และสูง (0.9) วิธีการประมาณค่าพารามิเตอร์แบบริดจ์มี 3 วิธีคือวิธีคาร์ลาร์ฟและชูเกอร์(Khalaf and Shukur) วิธีโฮเอิร์ล, เคนนาร์ดและบาลด์วินที่ปรับแก้ (New HKB) และวิธีลอว์เลสและแวงที่ปรับแก้ (New LW)และในการประมาณค่าแบบช่วงกำหนดค่าแบบช่วงกำหนดค่าสัมประสิทธ์ความเชื่อมั่นเท่ากับ 0.90, 0.95 และ 0.99 ตามลำดับ ในการวิจัยครั้งนี้มาจากการจำลองด้วยเทคนิคมอนติคาร์โลซึ่งกระทำซ้ำ 500 ครั้งในแต่ละสถานการณ์ด้วยโปรแกรม R 2.8.1 ผลการวิจัยสามารถสรุปได้ดังนี้ 1)กรณีการประมาณค่าแบบจุด พบว่า 97% ของจำนวนสถานการณ์จำลองทั้งหมด วิธีความถดถอยบูตสแตรปแบบริดจ์ให้ค่าเฉลี่ยความคลาดเคลื่อนกำลังสองน้อยที่สุด โดยที่ค่าเฉลี่ยความคลาดเคลื่อนกำลังสองมีค่าเพิ่มขึ้นเมื่อระดับความสัมพันธ์ ส่วนเบี่ยงเบนมาตรฐาน และจำนวนตัวแปรอิสระเพิ่มขึ้นและค่าความคลาดเคลื่อน

กำลังสองมีค่าลดลงเมื่อขนาดตัวอย่างเพิ่มขึ้น 2)กรณีการประมาณค่าแบบช่วง พบว่าจากจำนวนสถานการณ์จำลองทั้งหมด วิธีความถดถอยแบบริดจ์ให้ค่าสัมประสิทธิ์ความเชื่อมั่นไม่ต่ำกว่าที่กำหนดมากกว่าวิธีความถดถอยบูตสแตรปแบบริดจ์และ 66% ของสถานการณ์จำลองทั้งหมด วิธีความถดถอยบูตสแตรปแบบริดจ์ให้ค่าความยาวเฉลี่ยของช่วงความเชื่อมั่นต่ำกว่าวิธีความถดถอยแบบริดจ์

Ghani and Ahmad (2010)ได้ทำการศึกษาการคัดเลือกตัวแบบการวิเคราะห์การถดถอยเชิงเส้นพหุ ด้วยวิธีการถดถอยแบบขั้นตอน ที่ใช้วิธีเลือกแบบไปข้างหน้า และวิธีตัดออกแบบถอยหลังมาใช้ร่วมกัน เกณฑ์ที่ใช้ในการคัดเลือกตัวแบบ คือ ค่าสัมประสิทธิ์ตัวกำหนดแบบปรับ (R_a^2) ตัวแบบที่เหมาะสม คือ ตัวแบบที่ให้ค่าสัมประสิทธิ์ตัวกำหนดแบบปรับสูงที่สุด ระดับนัยสำคัญที่ใช้ในการทดสอบ คือ 0.05 ข้อมูลที่ใช้ในการวิจัยนี้มาจากข้อมูลพื้นที่การทำประมงในรัฐตรังกานู ประเทศมาเลเซีย ระหว่างปี ค.ศ. 1968 – 2007 โดยมีตัวแปรอิสระจำนวน 3 ตัวแปร คือ จำนวนชาวประมง จำนวนเรือประมง และจำนวนใบอนุญาตในการทำประมง ผลการศึกษาพบว่ามีการนำตัวแปรจำนวนชาวประมงและจำนวนใบอนุญาตในการทำประมงเข้ามาอยู่ในตัวแบบ ผลการศึกษาสรุปได้ว่าจำนวนชาวประมงและจำนวนใบอนุญาตในการทำประมงเป็นปัจจัยที่ส่งผลต่อพื้นที่ในการทำประมง

กานต์ณัฐ ณ บางช้าง(2554) ได้เสนอวิธีการคัดเลือกตัวแปรในการวิเคราะห์การถดถอยตัวแบบเชิงเส้นพหุ โดยใช้วิธีการค้นหาแบบต้องห้าม ที่ใช้ฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยและค่าเฉลี่ยความคลาดเคลื่อนสัมบูรณ์ เปรียบเทียบกับเทคนิคการคัดเลือกตัวแบบโดยวิธีการถดถอยแบบขั้นตอน เกณฑ์ที่ใช้ในการเปรียบเทียบคือ ร้อยละของจำนวนครั้งที่คัดเลือกตัวแบบได้ถูกต้อง โดยศึกษาทั้งในกรณีที่ไม่มีและมีปัญหาสหสัมพันธ์เชิงเส้นพหุ โดยใช้วิธีการจำลองข้อมูล มีผลการศึกษาคือ กรณีที่ไม่มีปัญหาสหสัมพันธ์เชิงเส้นพหุ ความสามารถในการคัดเลือกตัวแบบของวิธีการถดถอยแบบขั้นตอนและการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยมีร้อยละของความถูกต้องในการคัดเลือกตัวแปรเข้าสู่ตัวแบบใกล้เคียงกัน แต่หากใช้วิธีการค้นหาแบบต้องห้ามที่ใช้ฟังก์ชันเป้าหมายเป็นค่าเฉลี่ยความคลาดเคลื่อนสัมบูรณ์ มีร้อยละความถูกต้องของการคัดเลือกตัวแปรเข้าสู่ตัวแบบต่ำกว่า กรณีที่ใช้ค่าความคลาดเคลื่อนกำลังสองเฉลี่ยเล็กน้อยในกรณีที่มีปัญหาสหสัมพันธ์เชิงเส้นพหุ วิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยและค่าเฉลี่ยความคลาดเคลื่อนสัมบูรณ์มีร้อยละของตัวแบบที่คัดเลือกได้ถูกต้องสูงกว่าวิธีการถดถอยแบบขั้นตอน

บทที่ 3

การดำเนินงานวิจัย

การวิจัยนี้มีวัตถุประสงค์เพื่อคัดเลือกตัวแบบในการวิเคราะห์การถดถอยเชิงเส้นพหุ เมื่อตัวแปรอิสระมีความสัมพันธ์กันสูง 0.95, 0.99, 0.999 และ 0.9999 โดยใช้วิธีการค้นหาแบบต้องห้าม (Tabu Search) ที่ใช้ฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ย (MSE) และค่าความคลาดเคลื่อนกำลังสองเฉลี่ยปรับด้วยฟังก์ชันการลงโทษ (Penalty Function) เปรียบเทียบกับวิธีการถดถอยแบบขั้นตอนที่ประมาณค่าสัมประสิทธิ์การถดถอยด้วยวิธีกำลังสองน้อยที่สุดและวิธีการถดถอยแบบริดจ์โดยใช้วิธีการประมาณค่าพารามิเตอร์ริดจ์(r) 4 วิธีคือ วิธีโฮเอิร์ล, เคนนาร์ด และบาลด์วิน (Hoerl, Kennard and Baldwin: r_{HKB}) วิธีลอว์เลสและแวง (Lawless and Wang: r_{LW}) วิธีนอมูระ (Nomura: r_{HMO})และวิธีคาลาฟและชูเกอร์(Khalaf and Shukur: r_{KS})ขนาดตัวอย่างที่ใช้ในการศึกษา คือ 20, 60 และ 100 ข้อมูลที่ใช้ในการวิจัยได้มาจากการสร้างแบบจำลอง โดยทำซ้ำ 500 ครั้ง ในแต่ละขนาดตัวอย่าง โปรแกรมที่ใช้ในการวิเคราะห์ข้อมูล คือ โปรแกรม Matlab Version 2011aและการคัดเลือกตัวแบบโดยวิธีการค้นหาแบบต้องห้ามดัดแปลงมาจาก “Ts Directed by Direct Search Method for Nonlinear Global Optimization” (Hedar and Fukushima, 2003: 7-18)

3.1 ขั้นตอนการดำเนินงานวิจัย

งานวิจัยนี้ศึกษากรณีตัวแปรอิสระมีการแจกแจงเอกรูป (Uniform Distribution) และความคลาดเคลื่อนมีการแจกแจงปกติ (Normal Distribution) กำหนดจำนวนตัวแปรอิสระในตัวแบบเต็มรูป (Full Model) $k=7$ ตัวแปรและจำนวนตัวแปรอิสระในตัวแบบจริง (True Model) ที่ใช้ในการหาค่าตัวแปรตาม $q=5$ ตัวแปร ขนาดตัวอย่างที่ใช้คือ 20, 60 และ 100 จำนวนทำซ้ำในแต่ละสถานการณ์ $m=500$ โดยมีขั้นตอนการดำเนินงาน ดังนี้

1) สร้างประชากร N = 200,000จำนวน 2 กลุ่ม แต่ละกลุ่มประกอบด้วย $X_1, X_2, X_3,$ X_4, X_5, X_6, X_7 (กำหนดค่าสัมประสิทธิ์สหสัมพันธ์ระหว่างตัวแปรอิสระ X_1 และ X_3 ในประชากรกลุ่มที่ 1 เท่ากับ 0.999 และในประชากรกลุ่มที่ 2 เท่ากับ 0.9999)

2) สร้างตัวแปรอิสระสองตัวที่มีความสัมพันธ์กันโดยใช้วิธีรากที่สอง (Square Root Method) โดยจำลองตัวแปรช่วย U_1 และ U_2 ที่มีการแจกแจงเอกรูป ที่เป็นอิสระต่อกันโดย $U_i \sim (a,b)$ และ X_i เป็นตัวแปรอิสระที่ $\sigma_i^2 = Var(X_i) = \dfrac{(b-a)^2}{12}$, $E(X_i) = \dfrac{a+b}{2}$, เมื่อคำนวณค่าของตัวแปร $X_1 = \mu_1 + \sigma_1 U_1$ โดย μ_1 และ σ_1 คือ ค่าเฉลี่ยและค่าเบี่ยงเบนมาตรฐานของตัวแปร X_1 ตามลำดับ คำนวณ $X_3 = \mu_3 + \sigma_3(\rho U_1 + \sqrt{1-\rho_{13}{}^2 U_2})$ โดย μ_3 และ σ_3 คือ ค่าเฉลี่ยและค่าเบี่ยงเบนมาตรฐานของตัวแปร X_3 ตามลำดับ เมื่อ ρ_{13} เป็นค่าสัมประสิทธิ์สหสัมพันธ์ในประชากรระหว่างตัวแปรอิสระ X_1 และ X_3 โดยประชากรกลุ่มที่ 1 กำหนดค่าสัมประสิทธิ์สหสัมพันธ์ $\rho_{13} =$ 0.999กลุ่มที่ 2 กำหนดค่าสัมประสิทธิ์สหสัมพันธ์ $\rho_{13} = 0.9999$

3) กำหนดให้ตัวแปรอิสระมีการแจกแจงเอกรูป โดย
$X_1 \sim U(20,100),\ X_2 \sim U(10,120),\ X_3 \sim U(5,90),\ X_4 \sim U(-30,30), X_5 \sim U(-60,60),$
$X_6 \sim U(-20,40),\ X_7 \sim U(-50,50)$

4) สร้างความคลาดเคลื่อนที่เป็นอิสระต่อกันที่มีการแจกแจงปกติ $\boldsymbol{\varepsilon} \sim (0,30)$

5) กำหนดค่าสัมประสิทธิ์การถดถอยคือ $\boldsymbol{\beta}' = [30\ 20\ 10\ -5\ 15\ 4\ 0\ 0]$ ตัวแปรอิสระที่เกี่ยวข้องกับตัวแปรตาม $\mathbf{y}$ คือ X_1, X_2, X_3, X_4 และ X_5 และตัวแปรอิสระที่ไม่เกี่ยวข้องกับตัวแปรตาม $\mathbf{y}$ คือ X_6, X_7

6) สร้างตัวแปรตาม $\mathbf{y}$ ภายใต้ตัวแบบ

$$\mathbf{y} = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \beta_4 X_4 + \beta_5 X_5 + \beta_6 X_6 + \beta_7 X_7 + \boldsymbol{\varepsilon}$$

7) เมื่อได้ประชากรตามที่ต้องการ ดำเนินการสุ่มตัวอย่างขนาด 20, 60 และ 100 แล้วคำนวณค่าสัมประสิทธิ์สหสัมพันธ์ระหว่าง X_1 และ X_3 ในแต่ละตัวอย่างที่สุ่มได้

8) เนื่องจากปัญหาสหสัมพันธ์เชิงเส้นพหุ เป็นปัญหาที่พบบ่อยในตัวอย่าง ดังนั้นหากพบว่าตัวอย่างสุ่มชุดใดมีค่าสัมประสิทธิ์ระหว่าง X_1 และ X_3 ต่ำกว่ากำหนดจะไม่นำมาพิจารณาและจะพิจารณาแบ่งกลุ่มของตัวอย่างที่ได้จากการสุ่มตามค่าสัมประสิทธิ์สหสัมพันธ์ระหว่าง X_1 และ X_3 (r_{13}) ดังนี้

$0.945 \le r_{13} \le 0.954$ จัดให้อยู่ในกลุ่มค่าสัมประสิทธิ์สหสัมพันธ์เท่ากับ 0.95

$0.985 \le r_{13} \le 0.994$ จัดให้อยู่ในกลุ่มค่าสัมประสิทธิ์สหสัมพันธ์เท่ากับ0.99

$0.9985 \le r_{13} \le 0.9994$ จัดให้อยู่ในกลุ่มค่าสัมประสิทธิ์สหสัมพันธ์เท่ากับ0.999

$0.99985 \le r_{13} \le 0.99994$ จัดให้อยู่ในกลุ่มค่าสัมประสิทธิ์สหสัมพันธ์เท่ากับ0.9999

ทำเช่นนี้จนครบ m ครั้ง ในแต่ละกลุ่มค่าสัมประสิทธิ์สหสัมพันธ์และขนาดตัวอย่าง

9) ทำการคัดเลือกตัวแบบที่เหมาะสมจากวิธีการคัดเลือกตัวแบบ2วิธี คือ วิธีการถดถอยแบบขั้นตอน(Stepwise Regression)ที่ประมาณค่าพารามิเตอร์ด้วยวิธีกำลังสองน้อยที่สุดและวิธีการถดถอยแบบริดจ์โดยกำหนดให้ระดับนัยสำคัญของการคัดเลือกตัวแปรอิสระเข้าและนำตัวแปรอิสระออกจากตัวแบบมีค่าเท่ากันคือ 0.05และวิธีการค้นหาแบบต้องห้าม(Tabu Search)

10) คำนวณร้อยละของจำนวนครั้งที่คัดเลือกได้ตัวแบบจริง, ตัวแบบOverspecification, Underspecification และ Misspecificationจากการทำซ้ำจำนวน m ครั้งในแต่ละวิธี

3.2 ขั้นตอนการคัดเลือกตัวแปรเข้าสู่ตัวแบบ

3.2.1 ขั้นตอนการคัดเลือกตัวแปรเข้าสู่ตัวแบบด้วยวิธีการถดถอยแบบขั้นตอน

กำหนดจำนวนตัวแปรอิสระที่ใช้ในการคัดเลือก k ตัวแปร

3.2.1.1 เลือกตัวแปรอิสระตัวที่ iเข้าสู่ตัวแบบ โดยพิจารณาเลือกตัวแปรอิสระที่มีสหสัมพันธ์กับตัวแปรตาม $\mathbf{y}$ มากที่สุดคือ $\max\limits_{1\le i\le k}\left(r_{yi}\right)$ โดยที่ r_{yi} เป็นค่าสัมประสิทธิ์สหสัมพันธ์ระหว่างตัวแปรตาม $\mathbf{y}$ กับตัวแปรอิสระ x_i จากนั้นทดสอบสมมติฐาน $H_0 : \beta_i = 0$ เทียบกับ $H_1 : \beta_i \ne 0$ โดยใช้สถิติทดสอบเอฟ(F-test) มีตัวสถิติทดสอบ $F = \dfrac{SSR(X_i)}{SSE(X_i)/(n-2)}$ จะปฏิเสธ H_0 เมื่อ $F \ge F_{\alpha,(1,n-2)}$

โดย $SSR(X_i)$ คือผลรวมกำลังสองของการถดถอยที่ตัวแบบมีตัวแปรอิสระ X_i ; $SSR = \hat{\boldsymbol{\beta}}'\mathbf{X}'\mathbf{y} - n\bar{y}^2$

และ $SSE(X_i)$ คือผลรวมกำลังสองของความคลาดเคลื่อนที่ตัวแบบมีตัวแปรอิสระ X_i ;

$SSE = \mathbf{y}'\mathbf{y} - \hat{\boldsymbol{\beta}}'\mathbf{X}'\mathbf{y}$

ถ้าไม่ปฏิเสธสมมติฐานหลักจะหยุดการพิจารณาการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบ แต่ถ้าปฏิเสธสมมติฐานหลัก แสดงว่าตัวแปรอิสระ X_i มีส่วนในการอธิบายความผันแปรของตัวแปรตาม สรุปว่าตัวแปรอิสระ X_i สมควรอยู่ในรูปแบบการถดถอย

3.2.1.2 เลือกตัวแปรอิสระตัวที่ j เข้าสู่ตัวแบบเป็นตัวถัดไป พิจารณาจาก $\max\limits_{\substack{1\le j\le k\\ j\ne i}}(r_{y\,j.i})$ โดยที่ $r_{y\,j.i}$ เป็นค่าสัมประสิทธิ์สหสัมพันธ์บางส่วน(Partial Correlation) ระหว่างตัว

แปรตาม **y** กับตัวแปรอิสระ x_j เมื่อค่าของตัวแปรอิสระ x_i เป็นค่าคงตัว จากนั้นทดสอบสมมติฐาน $H_0 : \beta_j = 0$ เทียบกับ $H_1 : \beta_j \neq 0$ โดยใช้สถิติทดสอบเอฟบางส่วน

$F = \dfrac{SSR(X_j | X_i)}{SSE(X_i, X_j)/(n-3)}$ จะปฏิเสธ H_0 เมื่อ $F \geq F_{\alpha,(1,n-3)}$

โดย $SSR(X_j | X_i)$ คือผลรวมกำลังสองของการถดถอยบางส่วน

$SSR(X_j | X_i) = SSR(X_i, X_j) - SSR(X_i)$

ถ้าไม่ปฏิเสธสมมติหลักแสดงว่าตัวแปรอิสระตัวที่ j ไม่มีส่วนในการอธิบายความผันแปรของตัวแปรตามเมื่อมีตัวแปรอิสระ iอยู่ในตัวแบบ จึงนำตัวแปรอิสระตัวที่ j ออกจากตัวแบบ แต่หากปฏิเสธสมมติฐานหลักแสดงว่าตัวแปรอิสระตัวที่ j มีส่วนในการอธิบายความผันแปรของตัวแปรตามเมื่อมีตัวแปรอิสระตัวที่ iอยู่ในตัวแบบ จึงสามารถคงตัวแปรอิสระตัวที่ j ไว้ในตัวแบบ

3.2.1.3 หลังจากนั้นพิจารณาตัวแปรอิสระทุกตัวที่เข้ามาก่อนหน้าตัวแปรอิสระ X_j โดยถือว่าตัวแปรที่เข้ามาก่อนหน้านี้เข้ามาในตัวแบบเป็นตัวสุดท้าย ทดสอบสมมติฐาน $H_0 : \beta_i = 0$ เทียบกับ $H_1 : \beta_i \neq 0$ โดยใช้สถิติทดสอบเอฟบางส่วน $F = \dfrac{SSR(X_i | X_j)}{SSE(X_i, X_j)/(n-3)}$ จะปฏิเสธ H_0 เมื่อ $F \geq F_{\alpha,(1,n-3)}$ โดยถือว่าตัวแปรอิสระตัวที่เข้ามาใหม่อยู่ในตัวแบบและพิจารณาตัวแปรอิสระทุกตัวที่เข้ามาก่อนหน้านี้เป็นตัวแปรตัวสุดท้ายที่เข้ามาในตัวแบบ ถ้าไม่ปฏิเสธสมมติฐานหลักแสดงว่าตัวแปรอิสระตัวที่ i ไม่มีส่วนในการอธิบายความผันแปรของตัวแปรตามเมื่อมีตัวแปรอิสระตัวที่ j เข้ามาอยู่ในตัวแบบ จึงนำตัวแปรอิสระตัวที่ i ออกจากตัวแบบ แต่หากปฏิเสธสมมติฐานหลักแสดงว่าตัวแปรอิสระตัวที่ iมีส่วนในการอธิบายความผันแปรของตัวแปรตาม เมื่อมีตัวแปรอิสระตัวที่ j เข้ามาอยู่ในตัวแบบ จึงสามารถคงตัวแปรอิสระตัวที่ iไว้ในตัวแบบ และทำขั้นตอนของการคำนวณค่าพารามิเตอร์โดยใช้วิธีที่กล่าวไว้ในหัวข้อ 3.1.9

3.2.1.4 ทำซ้ำ 3.2.1.2และ3.2.1.3 จนกว่าจะไม่สามารถนำตัวแปรอิสระเข้ามาในตัวแบบหรือไม่สามารถนำตัวแปรอิสระออกจากตัวแบบได้ ขั้นตอนการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบจะจบลง

3.2.2 ขั้นตอนการประมาณค่าพารามิเตอร์และคัดเลือกตัวแปรเข้าสู่ตัวแบบโดยวิธีการค้นหาแบบต้องห้าม

3.2.2.1 กำหนดพิสัยของค่าพารามิเตอร์ ดังนี้

$b_0 = (10,50)$, $b_1 = (0,40)$, $b_2 = (0,20)$, $b_3 = (-10,0)$, $b_4 = (0,50)$, $b_5 = (0,10)$, $b_6 = (-5,5)$, $b_7 = (-10,10)$ กำหนดระยะที่อยู่ในรายการต้องห้ามเท่ากับ 10 รอบ

3.2.2.2 สุ่มค่าเริ่มต้นของพารามิเตอร์จากค่าในพิสัยที่กำหนด

3.2.2.3 นำค่าเริ่มต้นของพารามิเตอร์มาคำนวณในฟังก์ชันเป้าหมายตามที่ระบุในสมการ (1.2) และ (1.3)

3.2.2.4 สร้างเซตคำตอบข้างเคียงของค่าพารามิเตอร์ปัจจุบันในตัวแบบ

3.2.2.5 ตรวจสอบรายการต้องห้ามหากพบว่าไม่เกินระยะที่กำหนด ไม่ยกเลิกการเป็นรายการต้องห้าม

3.2.2.6 ดำเนินการค้นหาค่าพารามิเตอร์จากเซตคำตอบข้างเคียงแล้วนำมาคำนวณในฟังก์ชันเป้าหมาย

3.2.2.7 พิจารณาหาค่าพารามิเตอร์ชุดต่อมา หลังจากนั้นนำมาคำนวณค่าฟังก์ชันเป้าหมาย

3.2.2.8 พิจารณาหาค่าพารามิเตอร์จนกระทั่งครบ 10 รอบจะพบว่าพารามิเตอร์ในชุดสุดท้ายจะทำให้ฟังก์ชันเป้าหมายมีค่าใกล้เคียงกับค่าต่ำสุด

3.2.2.9 คัดเลือกตัวแปรโดยใช้สถิติทดสอบที (t-Test) โดยนำตัวแปรอิสระออกจากตัวแบบทีละตัวและทำการประมาณค่าพารามิเตอร์ของตัวแปรที่ยังคงอยู่ในตัวแบบใหม่ทุกครั้งที่นำตัวแปรออกจากตัวแบบ

วิธีการคำนวณค่าพารามิเตอร์โดยวิธีการค้นหาแบบต้องห้ามไม่ได้ใช้วิธีการหาเมตริกซ์ผกผันของ $\mathbf{X'X}$ จึงไม่มีปัญหาเหมือนกับการประมาณค่าพารามิเตอร์โดยใช้เมตริกซ์ผกผันของ $\mathbf{X'X}$ อย่างเช่นในวิธีกำลังสองน้อยที่สุด (OLS)

จากขั้นตอนการคัดเลือกตัวแปรโดยวิธีการค้นหาแบบต้องห้าม สามารถแสดงเป็นตัวอย่างดังนี้

สมมติว่าข้อมูลมีตัวแปรอิสระในตัวแบบเต็มรูปเท่ากับ 4 ตัวแปร แต่ตัวแปรอิสระที่ใช้ในการหาค่าตัวแปรตามเท่ากับ 2 ตัวแปร โดยที่ $\mathrm{X}_1 \sim \mathrm{U}(15,80)$ $\mathrm{X}_2 \sim \mathrm{U}(20,150)$ $\mathrm{X}_3 \sim \mathrm{U}(10,100)$ และ $\mathrm{X}_4 \sim \mathrm{U}(\text{-}50,50)$ กำหนดค่าสัมประสิทธิ์การถดถอยเป็น $\boldsymbol{\beta}' = [100 \;\; 24 \;\; 15 \;\; 0 \;\; 0]$ และ $\varepsilon \sim \mathrm{N}(0,1)$ ข้อมูลชุดนี้มีขนาดตัวอย่างเท่ากับ 25 ใช้วิธีการคัดเลือกตัวแปรโดยวิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็น MSE กำหนดระยะที่อยู่ในรายการต้องห้ามเท่ากับ 10 รอบ มีขั้นตอน ดังนี้

1) กำหนดพิสัยของค่าพารามิเตอร์เป็น $b_0 = (50,150)$, $b_1 = (10,50)$, $b_2 = (0,30), b_3 = (-40,0)$, $b_4 = (0,50)$ โดยพิจารณาจากตัวประมาณค่า OLS

2) สร้างค่าเริ่มต้นของพารามิเตอร์ โดยสุ่มจากค่าในพิสัยที่กำหนด ได้เป็น

$b_0 = 133.11, b_1 = 30.11, b_2 = 17.11, b_3 = -7.11, b_4 = 8.11$ นำมาคำนวณค่าฟังก์ชันเป้าหมาย MSE ได้เท่ากับ 27.56

3) สร้างเซตคำตอบข้างเคียงของค่าพารามิเตอร์ปัจจุบัน ได้เป็น $b_0 = 122.45$, 119.31, 124.97, $b_1 = 29.58$, 28.56, 30.12, $b_2 = 18.34$, 17.08, 18.76, $b_3 = -6.95$, -6.91, -7.01, $b_4 = 7.99$, 7.81, 7.75

4) ตรวจสอบรายการต้องห้ามพบว่า ไม่เกินระยะที่กำหนด จึงไม่ยกเลิกการเป็นรายการต้องห้าม

5) ดำเนินการค้นหาค่าพารามิเตอร์จากเซตคำตอบข้างเคียง ได้เป็น $b_0 = 119.31$, $b_1 = 29.58$, $b_2 = 17.08$, $b_3 = -6.95$, $b_4 = 7.75$ นำมาคำนวณค่าฟังก์ชันเป้าหมาย MSE ได้เท่ากับ 25.33 พบว่าให้ค่าฟังก์ชันเป้าหมายที่ต่ำกว่า จึงนำค่าพารามิเตอร์ที่ได้ไปไว้ในรายการต้องห้าม

6) ดำเนินการค้นหาค่าพารามิเตอร์ใหม่ เริ่มจากสุ่มค่าพารามิเตอร์ในพิสัยที่กำหนด ได้เป็น $b_0 = 117.42$, $b_1 = 29.11$, $b_2 = 17.01$, $b_3 = -5.11$, $b_4 = 6.87$ นำมาคำนวณค่าฟังก์ชันเป้าหมาย MSE ได้เท่ากับ 19.88

7) สร้างเซตคำตอบข้างเคียงของค่าพารามิเตอร์ปัจจุบัน ได้เป็น $b_0 = 115.45$, 113.56, 114.97, $b_1 = 29.08$, 28.97, 29.01, $b_2 = 16.99$, 16.94, 17.15, $b_3 = -4.05$, -4.01, -3.98, $b_4 = 5.87$, 5.31, 5.75

8) ตรวจสอบรายการต้องห้ามพบว่า ไม่เกินระยะที่กำหนด จึงไม่ยกเลิกการเป็นรายการต้องห้าม

9) ดำเนินการค้นหาค่าพารามิเตอร์จากเซตคำตอบข้างเคียง ได้เป็น $b_0 = 113.56$, $b_1 = 29.01$, $b_2 = 16.94$, $b_3 = -3.98$, $b_4 = 5.31$ นำมาคำนวณค่าฟังก์ชันเป้าหมาย MSE ได้เท่ากับ 12.39 พบว่าให้ค่าฟังก์ชันเป้าหมายที่ต่ำกว่า จึงนำค่าพารามิเตอร์ที่ได้ไปไว้ในรายการต้องห้าม

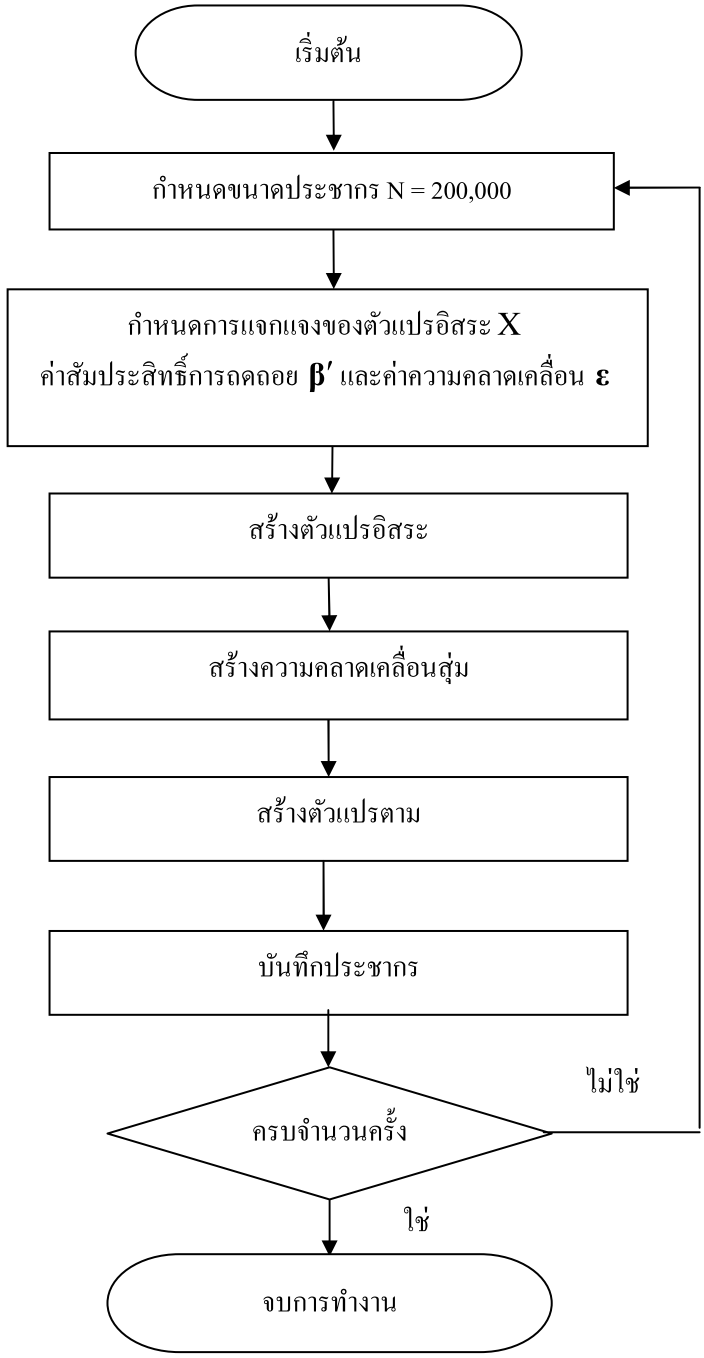
10) ดำเนินการค้นหาค่าพารามิเตอร์ใหม่ จนกระทั่งครบตามจำนวนครั้งที่กำหนด ได้ค่าพารามิเตอร์ ในรอบสุดท้ายเป็น $b_0 = 104.41$, $b_1 = 25.51$, $b_2 = 15.34$, $b_3 = -0.31$, $b_4 = 0.34$ คำนวณค่าฟังก์ชันเป้าหมาย MSE ได้เท่ากับ 1.01

ดังนั้นค่าพารามิเตอร์ที่ได้ คือ $b' = [104.41 \quad 25.51 \quad 15.34 \quad -0.31 \quad 0.34]$

3.3 การจำลองข้อมูล

3.3.1 การจำลองประชากร

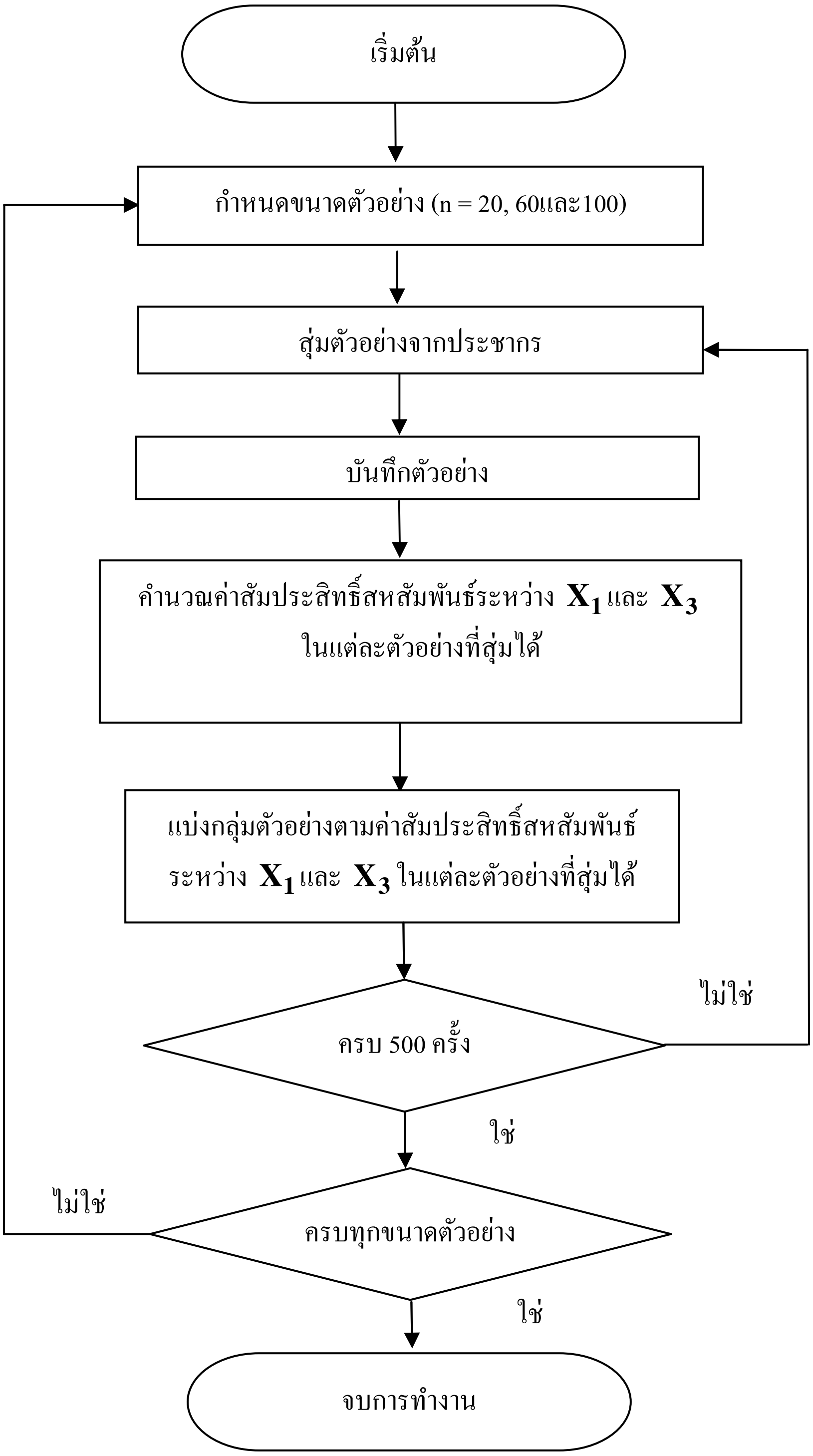
สร้างประชากร2กลุ่มโดยสร้างทีละกลุ่ม กำหนดขนาดประชากรกลุ่มละ N=200,000 กำหนดการแจกแจงของตัวแปรอิสระ X ให้มีการแจกแจงแบบเอกรูปโดย $X_1 \sim U(20,100)$ $X_2 \sim U(10,120),\ X_3 \sim U(5,90),\ X_4 \sim U(-30,30),\ X_5 \sim U(-60,60),\ X_6 \sim (-20,40)$ $X_7 \sim U(-50,50)$ กำหนดค่าสัมประสิทธิ์การถดถอย $\boldsymbol{\beta}' = [30\ 20\ 10\ -5\ 15\ 4\ 0\ 0]$ และกำหนดค่าความคลาดเคลื่อนให้มีการแจกแจงปกติ $\boldsymbol{\varepsilon} \sim (0,30)$ จากนั้นสร้างตัวแปรอิสระโดยให้มีการแจกแจงตามที่กำหนดไว้และสร้างตัวแปรอิสระ X_1 และ X_3 ให้มีความสัมพันธ์กันโดยใช้วิธีรากที่สอง กำหนดค่าสัมประสิทธิ์สหสัมพันธ์ $\rho_{13} = 0.999$ (กลุ่มที่ 2 กำหนดค่าสัมประสิทธิ์สหสัมพันธ์ $\rho_{13} =\ 0.9999$) สร้างความคลาดเคลื่อนที่เป็นอิสระต่อโดยมีการแจกแจงตามที่กำหนดไว้ สร้างตัวแปรตาม $\mathbf{y}$ จากตัวแบบ $\mathbf{y} = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \beta_4 X_4 + \beta_5 X_5 + \beta_6 X_6 + \beta_7 X_7 + \boldsymbol{\varepsilon}$ จากนั้นทำการบันทึกประชากรและทำซ้ำจนครบ 200,000 ครั้ง ดังแสดงในภาพที่ 3.1



ภาพที่ 3.1ขั้นตอนการสร้างประชากร

3.3.2 การสุ่มตัวอย่าง

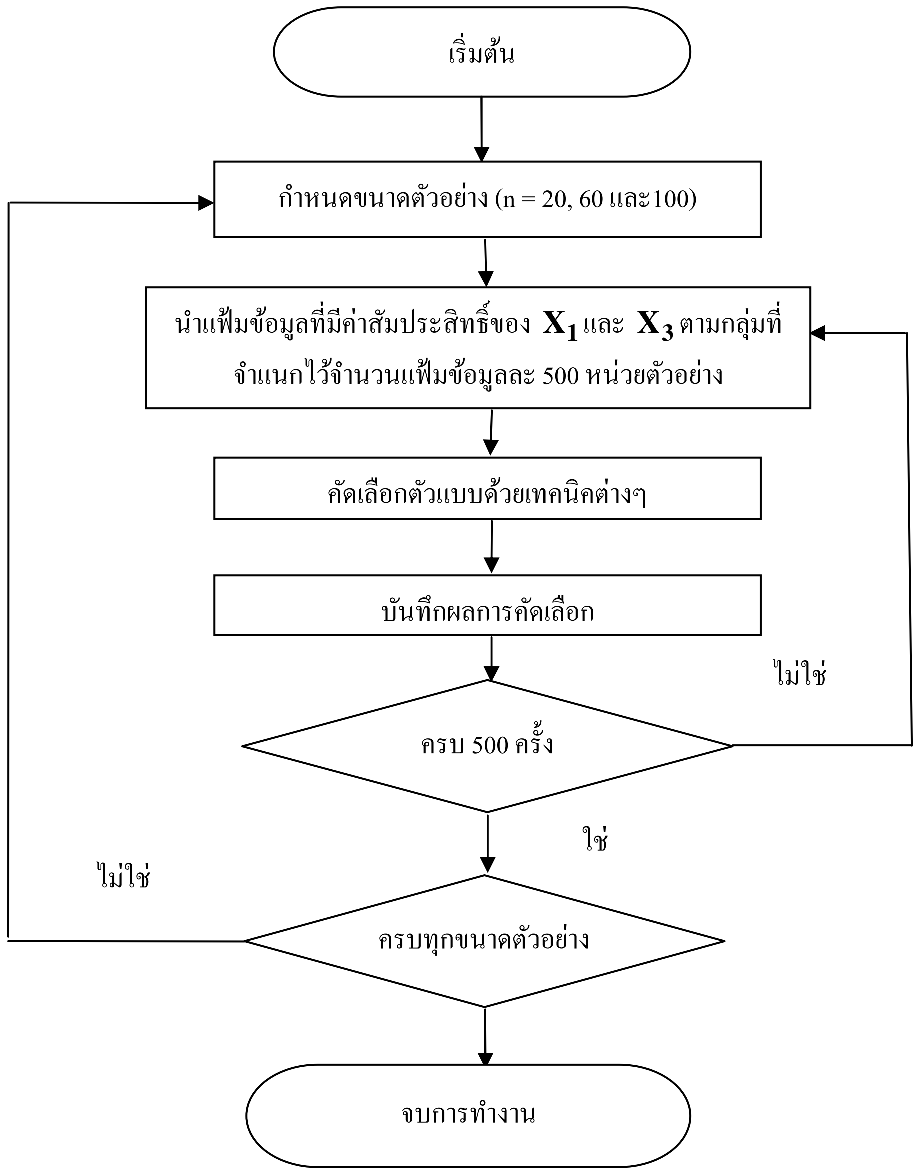
กำหนดขนาดตัวอย่าง 20, 60 และ 100 และดำเนินการสุ่มและบันทึกตัวอย่างที่สุ่มได้ จากนั้นคำนวณค่าสัมประสิทธิ์สหสัมพันธ์ระหว่าง X_1 และ X_3 ในแต่ละตัวอย่างที่สุ่มได้เพื่อทำการแบ่งกลุ่มของตัวอย่างตามช่วงของค่าสัมประสิทธิ์สหสัมพันธ์ระหว่าง X_1 และ X_3 ที่กำหนดไว้ในหัวข้อ 3.1 ข้อ 8)ทำซ้ำจนครบ 500 ครั้งในทุกขนาดตัวอย่าง ดังแสดงในภาพที่ 3.2



ภาพที่ 3.2ขั้นตอนการสุ่มตัวอย่าง

3.3.3 การคัดเลือกตัวแบบ

นำแฟ้มข้อมูลที่มีค่าสัมประสิทธิ์สหสัมพันธ์ของ X_1 และ X_3 ตามกลุ่มที่จำแนกไว้ซึ่งมีจำนวน 500 หน่วยตัวอย่างในแต่ละแฟ้มข้อมูลของแต่ละขนาดตัวอย่าง (n=20, 60 และ100) จากหัวข้อ 3.3.2 มาทำการคัดเลือกตัวแบบที่เหมาะสมจากวิธีการคัดเลือกตัวแบบ 2 วิธีคือวิธีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบแบบขั้นตอนที่ประมาณค่าพารามิเตอร์ด้วยวิธีกำลังสองน้อยที่สุดและวิธีการถดถอยแบบริดจ์ และวิธีการค้นหาแบบต้องห้าม จากนั้นทำการบันทึกผลการคัดเลือก และทำซ้ำเช่นจนครบ 500 ครั้งในแต่ละกลุ่มขนาดตัวอย่าง ดังแสดงในภาพที่ 3.3



ภาพที่ 3.3ขั้นตอนการคัดเลือกตัวแบบ

บทที่ 4

ผลการดำเนินงานวิจัย

ในบทนี้จะนำเสนอผลการดำเนินงานวิจัย โดยแบ่งออกเป็น 3กรณีตามขนาดตัวอย่างและกรณีที่มีการคัดเลือกตัวแปรอิสระโดยการกำหนดค่า r โดยวิธีของโฮเอิร์ล, เคนนาร์ด และ บาลด์วิน (Hoerl, Kennard and Baldwin: r_{HKB}) วิธีลอว์เลสและแวง (Lawless and Wang: r_{LW}) วิธีนอมูระ (Nomura: r_{HMO}) และวิธีคาลาฟและชูเกอร์(Khalaf and Shukur: r_{KS}) และการคัดเลือกตัวแปรอิสระโดยวิธีการค้นหาแบบต้องห้าม ที่ใช้ฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยและค่าความคลาดเคลื่อนกำลังสองเฉลี่ยปรับด้วยฟังก์ชันการลงโทษ $\frac{c\hat{\boldsymbol{\beta}}'\hat{\boldsymbol{\beta}}}{n-p}$ กับวิธีการถดถอยแบบขั้นตอน ในตัวแบบการถดถอยเชิงเส้นพหุ ในกรณีที่ตัวแปรอิสระ X_1 และ X_3 มีสหสัมพันธ์เชิงเส้นพหุกันสูงคือ 0.95, 0.99, 0.999 และ 0.9999 เมื่อจำนวนตัวแปรอิสระในตัวแบบเต็มรูปเท่ากับ 7ตัวแปรและจำนวนตัวแปรอิสระในตัวแบบที่ใช้ในการหาค่าตัวแปรตามเท่ากับ 5ตัวแปร ขนาดตัวอย่างที่ใช้ในการศึกษา คือ 20, 60 และ 100 แบ่งตามกลุ่มของค่าสัมประสิทธิ์สหสัมพันธ์ระหว่างตัวแปรอิสระ X_1 และ X_3 ในกรณีวิธีการค้นหาแบบต้องห้าม กำหนดให้ระยะที่อยู่ในรายการต้องห้ามเท่ากับ 10 รอบ ในวิธีการถดถอยแบบขั้นตอนระดับนัยสำคัญที่ใช้ในการเลือกตัวแปรอิสระเข้ามาในตัวแบบเป็น 0.05 และระดับนัยสำคัญที่ใช้ในการคงตัวแปรอิสระไว้ในตัวแบบเป็น 0.05 เกณฑ์ที่ใช้ในการเปรียบเทียบประสิทธิภาพ คือ ร้อยละของจำนวนครั้งที่คัดเลือกได้ตัวแปรอิสระที่เกี่ยวข้องเข้าสู่ตัวแบบผลจากการศึกษาด้วยวิธีการจำลองข้อมูลได้ผลดังนี้

1) กรณีตัวอย่างมีขนาดเท่ากับ 20 (n=20)

กรณีตัวแปรอิสระ X_1 และ X_3 มีค่าสัมประสิทธิ์สหสัมพันธ์ 0.95 วิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยมีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องมากที่สุดซึ่งใกล้เคียงกับวิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยปรับด้วยฟังก์ชันการลงโทษ คือมีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องร้อยละ 98.2 และ 97.0ตามลำดับ และไม่มีตัวแบบ Underspecification และ Missspecification มีเพียงตัวแบบ Overspecification เพียงร้อยละ 1.8 และ 3.0 ตามลำดับ ซึ่งน้อยกว่าวิธีอื่นๆ โดยวิธีการถดถอยแบบขั้นตอนที่ประมาณค่าสัมประสิทธิ์การถดถอยด้วยวิธีกำลังสองน้อยที่สุดและวิธีการถดถอยแบบริดจ์ที่ประมาณค่าคงตัว r 4 วิธีคือ วิธีโฮเอิร์ล เคนนาร์ด และ บาลด์วิน วิธีลอว์เลสและแวง วิธีนอมูระและวิธีคาลาฟและชูเกอร์มีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องร้อยละ 89.8, 89.6, 89.8, 87.0 และ 39.4 ตามลำดับและมีตัวแบบ Underspecification, Missspecification และ Overspecificationในวิธีการถดถอยแบบริดจ์ที่ประมาณค่าคงตัว r ด้วยวิธีโฮเอิร์ลเคนนาร์ด และ บาลด์วิน และวิธีนอมูระ ส่วนวิธีกำลังสองน้อยที่สุด, วิธีลอว์เลสและแวง และวิธีคาลาฟและชูเกอร์ มีเพียงตัวแบบ Underspecification และ Overspecification แต่วิธีคาลาฟและชูเกอร์มีร้อยละของตัวแบบ Underspecification สูงที่สุดคือ ร้อยละ 56.2

กรณีตัวแปรอิสระ X_1 และ X_3 มีค่าสัมประสิทธิ์สหสัมพันธ์ 0.99วิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยมีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องมากที่สุดซึ่งใกล้เคียงกับวิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยปรับด้วยฟังก์ชันการลงโทษ คือมีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องร้อยละ 98.0และ 97.6 ตามลำดับ และไม่มีตัวแบบ Underspecification และ Missspecification มีเพียงตัวแบบ Overspecification เพียงร้อยละ 2.0 และ 2.4 ตามลำดับ ซึ่งน้อยกว่าวิธีอื่นๆ โดยวิธีการถดถอยแบบขั้นตอนที่ประมาณค่าสัมประสิทธิ์การถดถอยด้วยวิธีกำลังสองน้อยที่สุดและวิธีการถดถอยแบบริดจ์ที่ประมาณค่าคงตัว r 4 วิธีคือ วิธีโฮเอิร์ลเคนนาร์ด และ บาลด์วิน วิธีลอว์เลสและแวง วิธีนอมูระและวิธีคาลาฟและชูเกอร์มีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องร้อยละ 87.2, 88.8, 87.2, 84.8 และ 36.8 ตามลำดับและมีตัวแบบUnderspecification, Missspecification และ Overspecificationในวิธีกำลังสองน้อยที่สุดและวิธีการถดถอยแบบริดจ์ที่ประมาณค่าคงตัว r ด้วยวิธีลอว์เลสและแวงส่วนวิธีโฮเอิร์ล เคนนาร์ด และ บาลด์วิน วิธีนอมูระ และวิธีคาลาฟและชูเกอร์ มีเพียงตัวแบบ Underspecification และ Overspecificationแต่วิธีคาลาฟและชูเกอร์มีร้อยละของตัวแบบ Underspecification สูงที่สุดคือ ร้อยละ 59.6

กรณีตัวแปรอิสระ X_1 และ X_3 มีค่าสัมประสิทธิ์สหสัมพันธ์ 0.999วิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยปรับด้วยฟังก์ชันการลงโทษมีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องมากที่สุดซึ่งใกล้เคียงกับวิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ย คือมีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องร้อยละ 97.8 และ 94.0ตามลำดับ ซึ่งวิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยปรับด้วยฟังก์ชันการลงโทษไม่มีตัวแบบ Underspecification และ Missspecificationมีเพียงตัวแบบ Overspecification เพียงร้อยละ 2.2 แต่วิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยมีตัวแบบ Underspecification, Overspecificationและ Missspecification เล็กน้อยเพียงร้อยละ 4.0, 1.8 และ 0.2 ตามลำดับ โดยวิธีการถดถอยแบบขั้นตอนที่ประมาณค่าสัมประสิทธิ์การถดถอยด้วยวิธีกำลังสองน้อยที่สุดและวิธีการถดถอยแบบริดจ์ที่ประมาณค่าคงตัว r 4 วิธีคือ วิธีโฮเอิร์ล เคนนาร์ด และ บาลด์วินวิธีลอว์เลสและแวง วิธีนอมูระและวิธีคาลาฟและชูเกอร์มีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องร้อยละ 81.0, 86.4, 81.0, 79.8และ 31.2 ตามลำดับ และมีตัวแบบ Underspecification, Missspecification และ Overspecificationในทุกวิธี

กรณีตัวแปรอิสระ X_1 และ X_3 มีค่าสัมประสิทธิ์สหสัมพันธ์ 0.9999วิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยปรับด้วยฟังก์ชันการลงโทษมีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องมากที่สุดร้อยละ 98.4 และไม่มีตัวแบบ Underspecification และ Missspecification มีเพียงตัวแบบที่ Overspecification เพียงร้อยละ 1.6 ในขณะที่วิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ย ที่มีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องเพียงร้อยละ 14.8 และวิธีการถดถอยแบบขั้นตอนที่ประมาณค่าสัมประสิทธิ์การถดถอยด้วยวิธีกำลังสองน้อยที่สุดและวิธีการถดถอยแบบริดจ์ที่ประมาณค่าคงตัว r 4 วิธีคือ วิธีโฮเอิร์ล เคนนาร์ด และ บาลด์วินวิธีลอว์เลสและแวง วิธีนอมูระและวิธีคาลาฟและชูเกอร์มีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องเพียงร้อยละ 18.0, 19.6, 18.0, 18.0 และ 2.4 ตามลำดับและมีตัวแบบ Underspecification, Missspecification และ Overspecificationโดยมีร้อยละของตัวแบบ Underspecificationมากที่สุดในทุกวิธี ดูรายละเอียดได้จากตารางที่ 4.1 และภาพที่ 4.1-4.4

ตารางที่ 4.1ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบ จำแนกตามค่าสัมประสิทธิ์สหสัมพันธ์และวิธีการคัดเลือกตัวแบบ เมื่อ n = 20

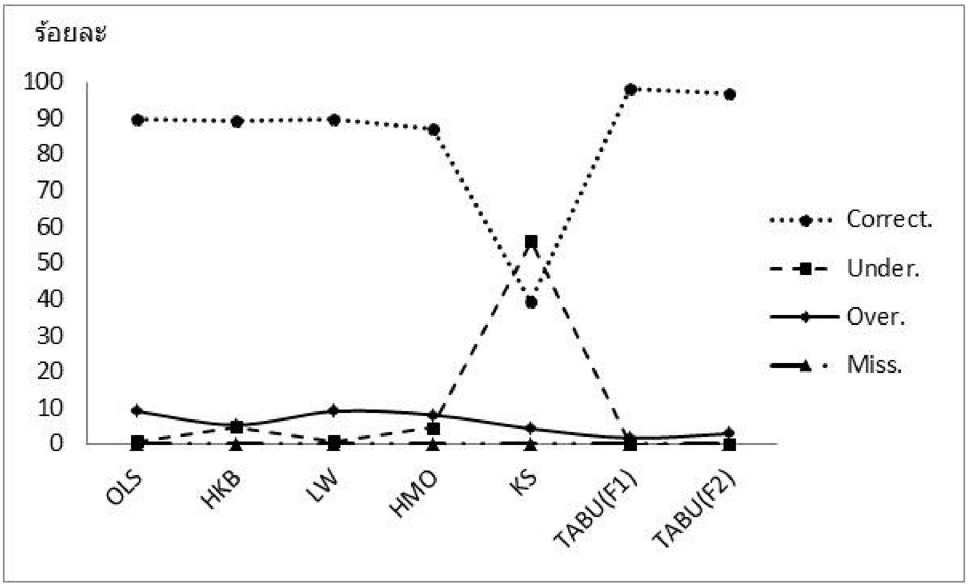
n=20, ρ_{13}=0.95				
วิธีใช้	Correct.	Under.	Over.	Miss.
Stepwise(OLS)	89.8	1.0	9.2	0.0
Stepwise(HKB)	89.6	4.8	5.4	0.2
Stepwise(LW)	89.8	1.0	9.2	0.0
Stepwise(HMO)	87.0	4.6	8.2	0.2
Stepwise(KS)	39.4	56.2	4.4	0.0
TABU(MSE)	98.2	0.0	1.8	0.0
TABU(penalty)	97.0	0.0	3.0	0.0

n=20, ρ_{13}=0.99				
วิธีใช้	Correct.	Under.	Over.	Miss.
Stepwise(OLS)	87.2	1.6	10.8	0.4
Stepwise(HKB)	88.8	4.6	6.6	0.0
Stepwise(LW)	87.2	1.6	10.8	0.4
Stepwise(HMO)	84.8	4.6	10.6	0.0
Stepwise(KS)	36.8	59.6	3.6	0.0
TABU(MSE)	98.0	0.0	2.0	0.0
TABU(penalty)	97.6	0.0	2.4	0.0

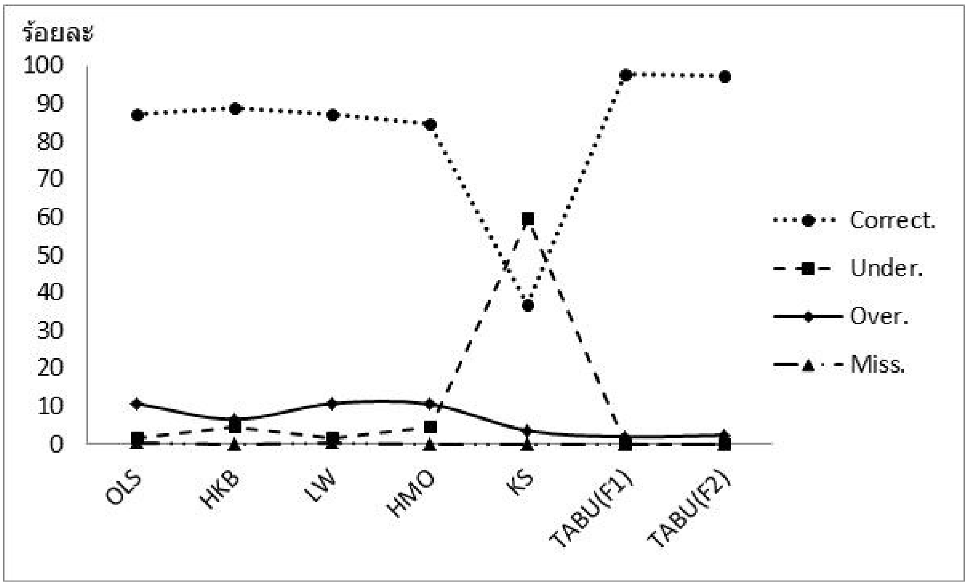
n=20, ρ_{13}=0.999				
วิธีใช้	Correct.	Under.	Over.	Miss.
Stepwise(OLS)	81.0	7.2	10.4	1.4
Stepwise(HKB)	86.4	7.6	5.4	0.6
Stepwise(LW)	81.0	7.2	10.4	1.4
Stepwise(HMO)	79.8	9.2	9.8	1.2
Stepwise(KS)	31.2	64.2	3.8	0.8
TABU(MSE)	94.0	4.0	1.8	0.2
TABU(penalty)	97.8	0.0	2.2	0.0

n=20, ρ_{13}=0.9999				
วิธีใช้	Correct.	Under.	Over.	Miss.
Stepwise(OLS)	18.0	71.6	2.6	7.8
Stepwise(HKB)	19.6	74.0	2.8	3.6
Stepwise(LW)	18.0	71.6	2.6	7.8
Stepwise(HMO)	18.0	72.0	2.6	7.4
Stepwise(KS)	2.4	94.4	0.6	2.6
TABU(MSE)	14.8	83.8	1.2	0.2
TABU(penalty)	98.4	0.0	1.6	0.0

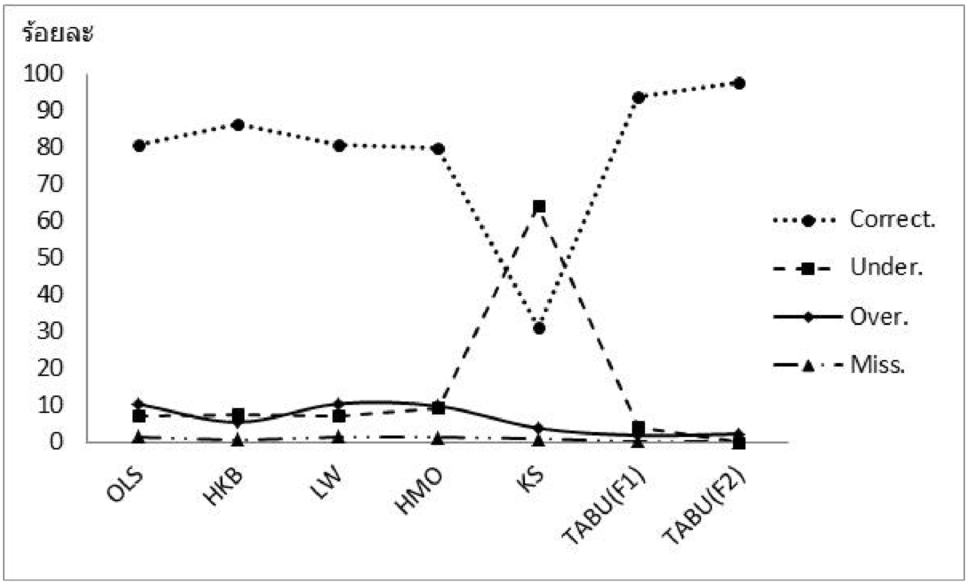
หมายเหตุ: *ρ_{13} คือค่าสัมประสิทธิ์สหสัมพันธ์ระหว่าง X_1 และ X_3



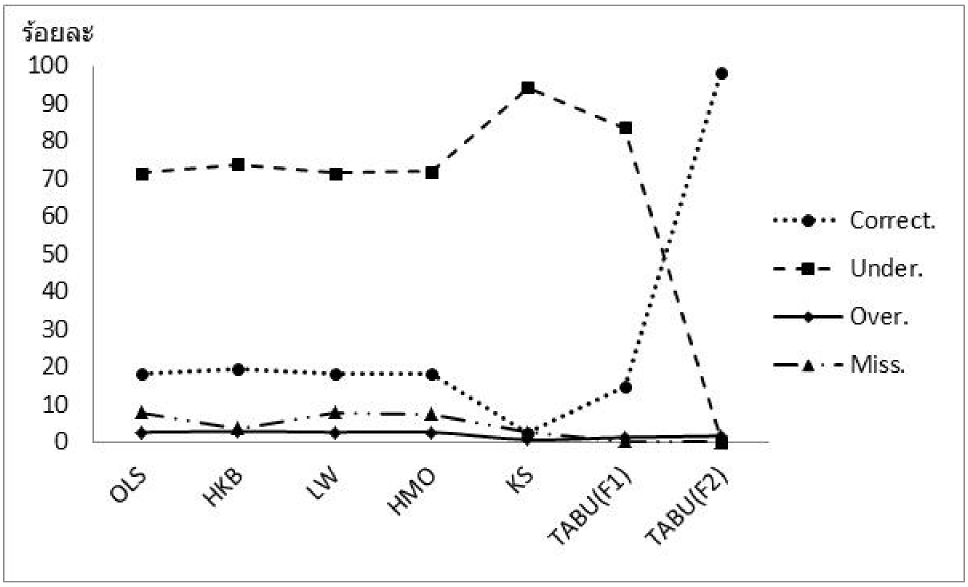
ภาพที่ 4.1 ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบ จำแนกตามวิธีการคัดเลือกตัวแบบเมื่อ n=20, ρ_{13}=0.95



ภาพที่ 4.2 ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบ จำแนกตามวิธีการคัดเลือกตัวแบบเมื่อ n=20, ρ_{13}=0.99



ภาพที่ 4.3 ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบ จำแนกตามวิธีการคัดเลือกตัวแบบเมื่อ n=20, ρ_{13}=0.999



ภาพที่ 4.4 ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบ จำแนกตามวิธีการคัดเลือกตัวแบบเมื่อ n=20, ρ_{13}=0.9999

2) กรณีตัวอย่างมีขนาดเท่ากับ 60 (n=60)

กรณีตัวแปรอิสระ X_1 และ X_3 มีค่าสัมประสิทธิ์สหสัมพันธ์ 0.95วิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายทั้งสองฟังก์ชันมีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องมากที่สุดและเท่ากัน คือถูกต้องร้อยละ 97.2 และไม่มีตัวแบบUnderspecification และ Missspecification มีเพียงตัวแบบOverspecification เพียงร้อยละ 2.8 ซึ่งน้อยกว่าวิธีอื่นๆ โดยวิธีการถดถอยแบบขั้นตอนที่ประมาณค่าสัมประสิทธิ์การถดถอยด้วยวิธีกำลังสองน้อยที่สุดและวิธีการถดถอยแบบริดจ์ที่ประมาณค่าคงตัว r 4 วิธีคือ วิธีโฮเอิร์ล เคนนาร์ด และ บาลด์วินวิธีลอว์เลสและแวง วิธีนอมูระและวิธีคาลาฟและชูเกอร์มีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องร้อยละ 89.6, 93.6, 89.6, 89.6 และ 56.2 ตามลำดับและมีตัวแบบUnderspecification และ Overspecificationเฉพาะในวิธีการถดถอยแบบริดจ์ที่ประมาณค่าคงตัว r ด้วยวิธีคาลาฟและชูเกอร์ส่วนวิธีอื่นๆมีเพียงตัวแบบ Overspecification แต่มีร้อยละของตัวแบบOverspecification มากกว่าวิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายทั้งสองฟังก์ชัน

กรณีตัวแปรอิสระ X_1 และ X_3 มีค่าสัมประสิทธิ์สหสัมพันธ์ 0.99วิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยมีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องมากที่สุดซึ่งใกล้เคียงกับวิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยปรับด้วยฟังก์ชันการลงโทษ คือมีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องร้อยละ 96.6 และ 96.0ตามลำดับ และไม่มีตัวแบบUnderspecification และ Missspecification มีเพียงตัวแบบOverspecificationเพียงร้อยละ 3.4 และ 4.0 ตามลำดับซึ่งน้อยกว่าวิธีอื่นๆ โดยวิธีการถดถอยแบบขั้นตอนที่ประมาณค่าสัมประสิทธิ์การถดถอยด้วยวิธีกำลังสองน้อยที่สุดและวิธีการถดถอยแบบริดจ์ที่ประมาณค่าคงตัว r 4 วิธีคือ วิธีโฮเอิร์ล เคนนาร์ด และ บาลด์วินวิธีลอว์เลสและแวง วิธีนอมูระและวิธีคาลาฟและชูเกอร์มีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องร้อยละ 89.6, 93.0, 89.6, 89.6 และ 59.4 ตามลำดับและมีตัวแบบUnderspecification และ Overspecification เฉพาะในวิธีการถดถอยแบบริดจ์ที่ประมาณค่าคงตัว r ด้วยวิธีคาลาฟและชูเกอร์ส่วนวิธีอื่นๆมีเพียงตัวแบบOverspecification แต่มีร้อยละของตัวแบบOverspecification มากกว่าวิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายทั้งสองฟังก์ชัน

กรณีตัวแปรอิสระ X_1 และ X_3 มีค่าสัมประสิทธิ์สหสัมพันธ์ 0.999วิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยมีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องมากที่สุดซึ่งใกล้เคียงกับวิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยปรับด้วยฟังก์ชันการลงโทษ คือมีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องร้อยละ 98.0และ 97.6 ตามลำดับ และไม่มีตัวแบบUnderspecificationและ

Missspecification มีเพียงตัวแบบOverspecification เพียงร้อยละ 2.0และ 2.4 ตามลำดับซึ่งน้อยกว่าวิธีอื่นๆ โดยวิธีการถดถอยแบบขั้นตอนที่ประมาณค่าสัมประสิทธิ์การถดถอยด้วยวิธีกำลังสองน้อยที่สุดและวิธีการถดถอยแบบริดจ์ที่ประมาณค่าคงตัว r 4 วิธีคือ วิธีโฮเอิร์ลเคนนาร์ด และ บาลด์วินวิธีลอว์เลสและแวง วิธีนอมูระและวิธีคาลาฟและชูเกอร์มีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องร้อยละ 90.4, 94.2, 90.4, 90.4 และ 53.6 ตามลำดับและมีตัวแบบ Underspecification และ Overspecificationเฉพาะในวิธีการถดถอยแบบริดจ์ที่ประมาณค่าคงตัว r ด้วยวิธีคาลาฟและชูเกอร์ ส่วนวิธีอื่นๆมีเพียงตัวแบบOverspecification แต่มีร้อยละของตัวแบบOverspecification มากกว่าวิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายทั้งสองฟังก์ชัน

กรณีตัวแปรอิสระ X_1 และ X_3 มีค่าสัมประสิทธิ์สหสัมพันธ์ 0.9999วิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยปรับด้วยฟังก์ชันการลงโทษมีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องมากที่สุดร้อยละ 97.6 และไม่มีตัวแบบ Underspecification และ Missspecification มีเพียงตัวแบบOverspecification เพียงร้อยละ 2.4 ในขณะที่วิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ย ที่มีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องเพียงร้อยละ 55.2 และมีตัวแบบUnderspecification ร้อยละ 44.2 และ Overspecificationร้อยละ 0.6 ส่วนวิธีการถดถอยแบบขั้นตอนที่ประมาณค่าสัมประสิทธิ์การถดถอยด้วยวิธีกำลังสองน้อยที่สุดและวิธีการถดถอยแบบริดจ์ที่ประมาณค่าคงตัว r 4 วิธีคือ วิธีโฮเอิร์ล เคนนาร์ด และ บาลด์วินวิธีลอว์เลสและแวง วิธีนอมูระและวิธีคาลาฟและชูเกอร์มีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องเพียงร้อยละ 52.4, 72.0, 52.4, 52.4 และ 24.6 ตามลำดับ และมีตัวแบบUnderspecification, Missspecification และ Overspecification โดยมีร้อยละของตัวแบบUnderspecificationมากที่สุดในทุกวิธี ดูรายละเอียดได้จากตารางที่ 4.2 และภาพที่ 4.5-4.8

ตารางที่ 4.2ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบ จำแนกตามค่าสัมประสิทธิ์สหสัมพันธ์และวิธีการคัดเลือกตัวแบบ เมื่อ n = 60

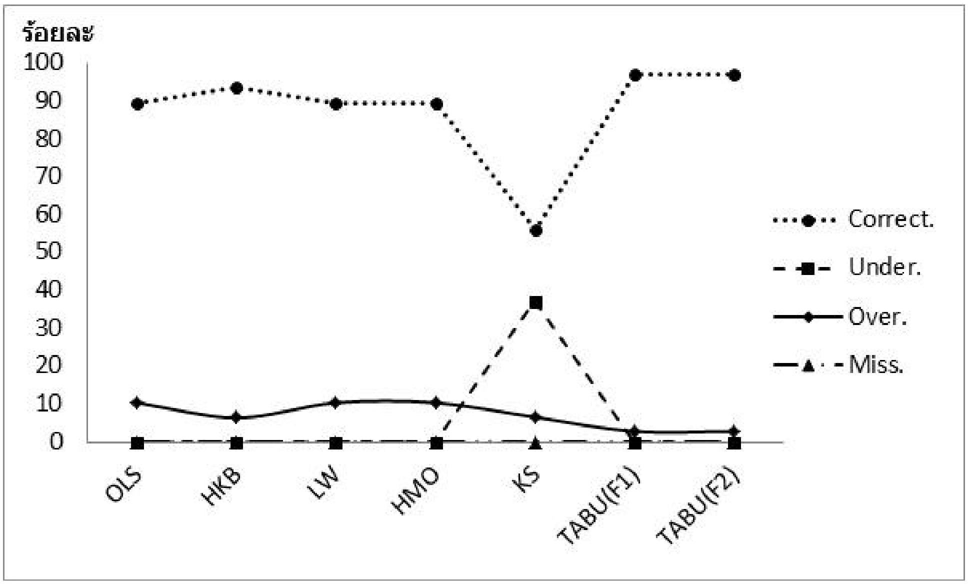
n=60, ρ_{13}=0.95				
วิธีใช้	Correct.	Under.	Over.	Miss.
Stepwise(OLS)	89.6	0.0	10.4	0.0
Stepwise(HKB)	93.6	0.0	6.4	0.0
Stepwise(LW)	89.6	0.0	10.4	0.0
Stepwise(HMO)	89.6	0.0	10.4	0.0
Stepwise(KS)	56.2	37.2	6.6	0.0
TABU(MSE)	97.2	0.0	2.8	0.0
TABU(penalty)	97.2	0.0	2.8	0.0

n=60, ρ_{13}=0.99				
วิธีใช้	Correct.	Under.	Over.	Miss.
Stepwise(OLS)	89.6	0.0	10.4	0.0
Stepwise(HKB)	93.0	0.0	7.0	0.0
Stepwise(LW)	89.6	0.0	10.4	0.0
Stepwise(HMO)	89.6	0.0	10.4	0.0
Stepwise(KS)	59.4	33.2	7.4	0.0
TABU(MSE)	96.6	0.0	3.4	0.0
TABU(penalty)	96.0	0.0	4.0	0.0

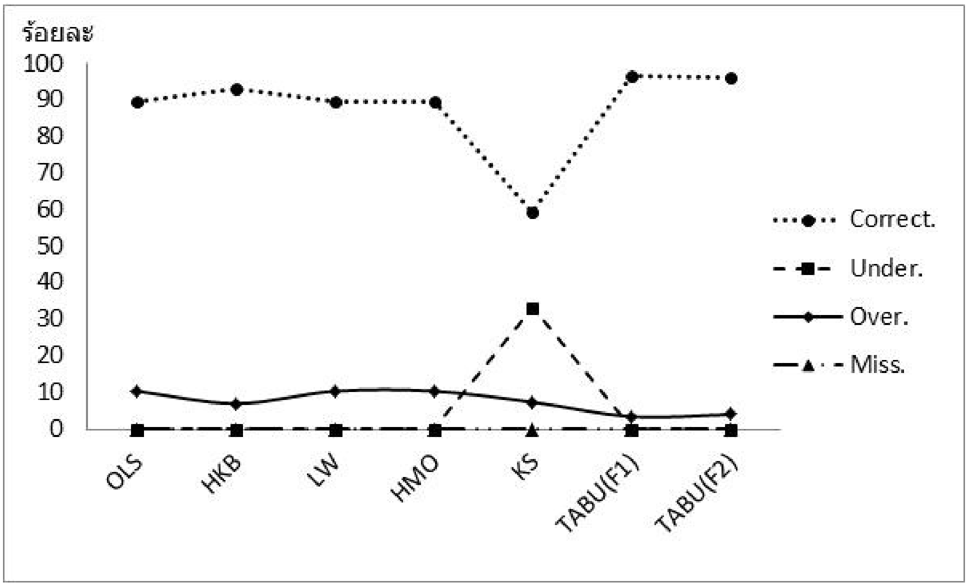
n=60, ρ_{13}=0.999				
วิธีใช้	Correct.	Under.	Over.	Miss.
Stepwise(OLS)	90.4	0.0	9.6	0.0
Stepwise(HKB)	94.2	0.0	5.8	0.0
Stepwise(LW)	90.4	0.0	9.6	0.0
Stepwise(HMO)	90.4	0.0	9.6	0.0
Stepwise(KS)	53.6	37.4	9.0	0.0
TABU(MSE)	98.0	0.0	2.0	0.0
TABU(penalty)	97.6	0.0	2.4	0.0

n=60, ρ_{13}=0.9999				
วิธีใช้	Correct.	Under.	Over.	Miss.
Stepwise(OLS)	52.4	36.2	6.4	5.0
Stepwise(HKB)	72.0	19.2	6.8	2.0
Stepwise(LW)	52.4	36.2	6.4	5.0
Stepwise(HMO)	52.4	36.2	6.4	5.0
Stepwise(KS)	24.6	67.6	4.4	3.4
TABU(MSE)	55.2	44.2	0.6	0.0
TABU(penalty)	97.6	0.0	2.4	0.0

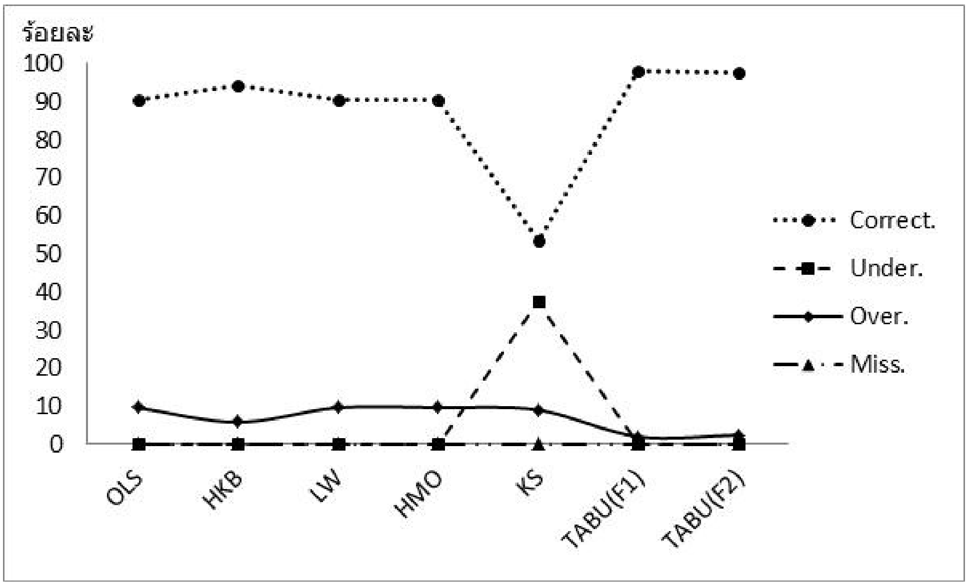
หมายเหตุ: *ρ_{13} คือค่าสัมประสิทธิ์สหสัมพันธ์ระหว่าง X_1 และ X_3



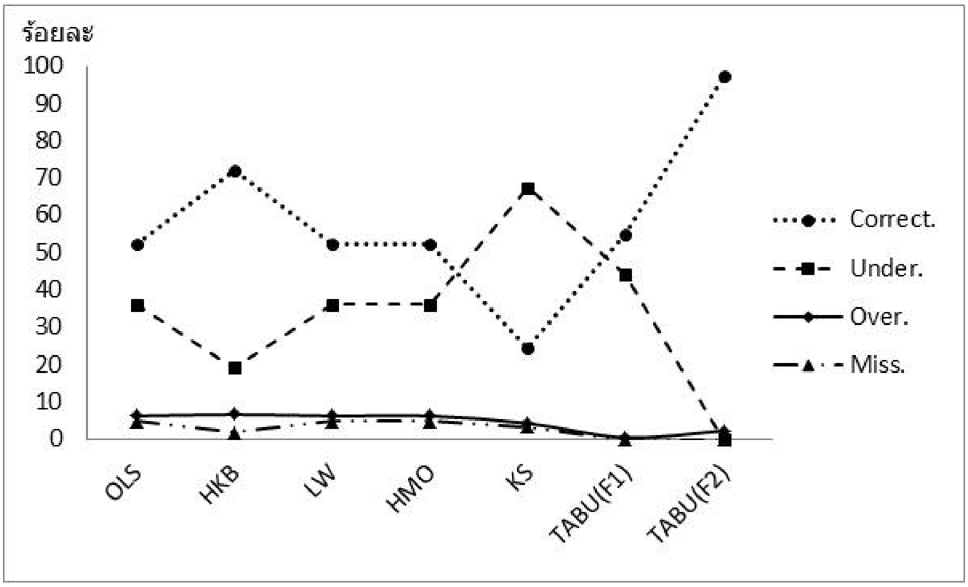
ภาพที่ 4.5 ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบ จำแนกตามวิธีการคัดเลือกตัวแบบเมื่อ n=60, ρ_{13}=0.95



ภาพที่ 4.6 ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบ จำแนกตามวิธีการคัดเลือกตัวแบบเมื่อ n=60, ρ_{13}=0.99



ภาพที่ 4.7 ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบ จำแนกตามวิธีการคัดเลือกตัวแบบเมื่อ n=60, ρ_{13}=0.999



ภาพที่ 4.8 ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบ จำแนกตามวิธีการคัดเลือกตัวแบบเมื่อ n=60, ρ_{13}=0.9999

3) กรณีตัวอย่างมีขนาดเท่ากับ 100 (n=100)

กรณีตัวแปรอิสระ X_1 และ X_3 มีค่าสัมประสิทธิ์สหสัมพันธ์ 0.95วิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยมีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องมากที่สุดซึ่งใกล้เคียงกับวิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยปรับด้วยฟังก์ชันการลงโทษ คือมีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องร้อยละ 97.4 และ 96.6 ตามลำดับ และไม่มีตัวแบบUnderspecification และ Missspecification มีเพียงตัวแบบOverspecification เพียงร้อยละ 2.6 และ 3.4 ตามลำดับซึ่งน้อยกว่าวิธีอื่นๆ โดยวิธีการถดถอยแบบขั้นตอนที่ประมาณค่าสัมประสิทธิ์การถดถอยด้วยวิธีกำลังสองน้อยที่สุดและวิธีการถดถอยแบบริดจ์ที่ประมาณค่าคงตัว r 4 วิธีคือ วิธีโฮเอิร์ล เคนนาร์ด และ บาลด์วิน วิธีลอว์เลสและแวง วิธีนอมูระและวิธีคาลาฟและชูเกอร์มีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องสูงขึ้นเป็นร้อยละ 91.2, 94.4, 91.2, 91.2 และ 67.0 ตามลำดับและมีตัวแบบที่ Underspecification และ Overspecificationเฉพาะในวิธีการถดถอยแบบริดจ์ที่ประมาณค่าคงตัว r ด้วยวิธีคาลาฟและชูเกอร์ส่วนวิธีอื่นๆมีเพียงตัวแบบ Overspecificationแต่มีร้อยละของตัวแบบที่ Overspecificationมากกว่าวิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายทั้งสองฟังก์ชัน

กรณีตัวแปรอิสระ X_1 และ X_3 มีค่าสัมประสิทธิ์สหสัมพันธ์ 0.99 วิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยมีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องมากที่สุดซึ่งใกล้เคียงกับวิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยปรับด้วยฟังก์ชันการลงโทษ คือมีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องร้อยละ 97.0และ 96.8 ตามลำดับ และไม่มีตัวแบบ Underspecificationและ Missspecification มีเพียงตัวแบบOverspecification เพียงร้อยละ 3.0และ 3.2 ตามลำดับซึ่งน้อยกว่าวิธีอื่นๆ โดยวิธีการถดถอยแบบขั้นตอนที่ประมาณค่าสัมประสิทธิ์การถดถอยด้วยวิธีกำลังสองน้อยที่สุดและวิธีการถดถอยแบบริดจ์ที่ประมาณค่าคงตัว r 4 วิธีคือ วิธีโฮเอิร์ลเคนนาร์ด และ บาลด์วิน วิธีลอว์เลสและแวง วิธีนอมูระและวิธีคาลาฟและชูเกอร์มีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องสูงขึ้นเป็นร้อยละ 90.8, 94.8, 90.8, 90.8 และ 62.4 ตามลำดับ และมีตัวแบบ Underspecification และ Overspecificationเฉพาะในวิธีการถดถอยแบบริดจ์ที่ประมาณค่าคงตัว r ด้วยวิธีคาลาฟและชูเกอร์ส่วนวิธีอื่นๆมีเพียงตัวแบบที่Overspecification แต่มีร้อยละของตัวแบบ Overspecification มากกว่าวิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายทั้งสองฟังก์ชัน

กรณีตัวแปรอิสระ X_1 และ X_3 มีค่าสัมประสิทธิ์สหสัมพันธ์ 0.999วิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยมีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องมากที่สุดซึ่งใกล้เคียงมากกับวิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่า

ความคลาดเคลื่อนกำลังสองเฉลี่ยปรับด้วยฟังก์ชันการลงโทษ คือมีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องร้อยละ 96.6และ 96.4ตามลำดับ และไม่มีตัวแบบที่ Underspecification และ Missspecification มีเพียงตัวแบบที่ Overspecification เพียงร้อยละ 3.4 และ 3.6 ตามลำดับซึ่งน้อยกว่าวิธีอื่นๆ โดยวิธีการถดถอยแบบขั้นตอนที่ประมาณค่าสัมประสิทธิ์การถดถอยด้วยวิธีกำลังสองน้อยที่สุดและวิธีการถดถอยแบบริดจ์ที่ประมาณค่าคงตัว r 4 วิธีคือ วิธีโฮเอิร์ลเคนนาร์ด และ บาลด์วินวิธีลอว์เลสและแวง วิธีนอมูระและวิธีคาลาฟและชูเกอร์มีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องร้อยละ 89.8, 93.4, 89.8, 89.8 และ67.4 ตามลำดับและมีตัวแบบที่ Underspecification และ Overspecificationเฉพาะในวิธีการถดถอยแบบริดจ์ที่ประมาณค่าคงตัว r ด้วยวิธีคาลาฟและชูเกอร์ส่วนวิธีอื่นๆมีเพียงตัวแบบที่Overspecification แต่มีร้อยละของตัวแบบที่ Overspecification มากกว่าวิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายทั้งสองฟังก์ชัน

กรณีตัวแปรอิสระ X_1 และ X_3 มีค่าสัมประสิทธิ์สหสัมพันธ์ 0.9999วิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยปรับด้วยฟังก์ชันการลงโทษมีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องมากที่สุดร้อยละ 97.4 และไม่มีตัวแบบ Underspecification และ Missspecification มีเพียงตัวแบบOverspecification เพียงร้อยละ 2.6 ในขณะที่วิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ย ที่มีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องร้อยละ 80.0 และมีตัวแบบ Underspecification ร้อยละ 17.8 และ Overspecificationร้อยละ 2.0 ส่วนวิธีการถดถอยแบบขั้นตอนที่ประมาณค่าสัมประสิทธิ์การถดถอยด้วยวิธีกำลังสองน้อยที่สุดและวิธีการถดถอยแบบริดจ์ที่ประมาณค่าคงตัว r 4 วิธีคือ วิธีโฮเอิร์ล เคนนาร์ด และ บาลด์วินวิธีลอว์เลสและแวง วิธีนอมูระและวิธีคาลาฟและชูเกอร์ มีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบถูกต้องร้อยละ 76.0, 84.0, 76.0, 76.0และ49.6 ตามลำดับและมีตัวแบบ Underspecification, Missspecification และ Overspecificationโดยมีร้อยละของตัวแบบ Underspecificationมากที่สุดในทุกวิธี ดูรายละเอียดได้จากตารางที่ 4.3 และภาพที่ 4.9-4.12

ตารางที่ 4.3ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบ จำแนกตามค่าสัมประสิทธิ์สหสัมพันธ์และวิธีการคัดเลือกตัวแบบ เมื่อ n = 100

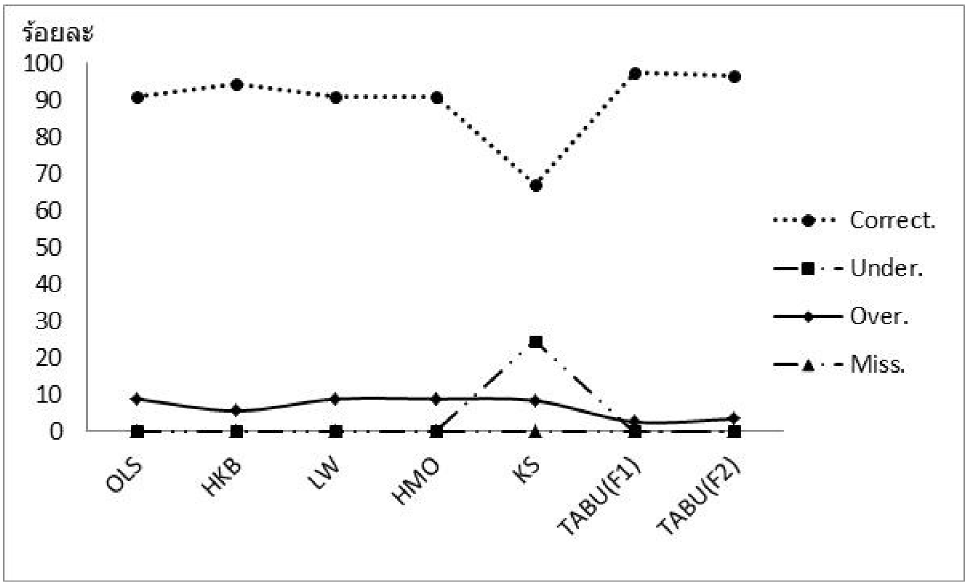
n=100, ρ_{13}=0.95				
วิธีใช้	Correct.	Under.	Over.	Miss.
Stepwise(OLS)	91.2	0.0	8.8	0.0
Stepwise(HKB)	94.4	0.0	5.6	0.0
Stepwise(LW)	91.2	0.0	8.8	0.0
Stepwise(HMO)	91.2	0.0	8.8	0.0
Stepwise(KS)	67.0	24.6	8.4	0.0
TABU(MSE)	97.4	0.0	2.6	0.0
TABU(penalty)	96.6	0.0	3.4	0.0

n=100, ρ_{13}=0.99				
วิธีใช้	Correct.	Under.	Over.	Miss.
Stepwise(OLS)	90.8	0.0	9.2	0.0
Stepwise(HKB)	94.8	0.0	5.2	0.0
Stepwise(LW)	90.8	0.0	9.2	0.0
Stepwise(HMO)	90.8	0.0	9.2	0.0
Stepwise(KS)	62.4	30.4	7.2	0.0
TABU(MSE)	97.0	0.0	3.0	0.0
TABU(penalty)	96.8	0.0	3.2	0.0

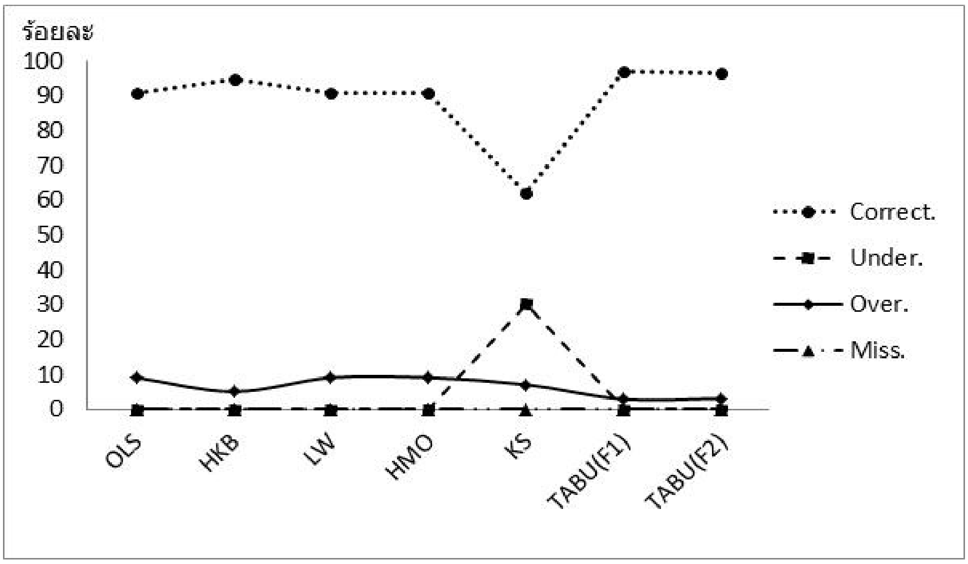
n=100, ρ_{13}=0.999				
วิธีใช้	Correct.	Under.	Over.	Miss.
Stepwise(OLS)	89.8	0.0	10.2	0.0
Stepwise(HKB)	93.4	0.0	6.6	0.0
Stepwise(LW)	89.8	0.0	10.2	0.0
Stepwise(HMO)	89.8	0.0	10.2	0.0
Stepwise(KS)	67.4	23.4	9.2	0.0
TABU(MSE)	96.6	0.0	3.4	0.0
TABU(penalty)	96.4	0.0	3.6	0.0

n=100, ρ_{13}=0.9999				
วิธีใช้	Correct.	Under.	Over.	Miss.
Stepwise(OLS)	76.0	13.6	8.4	2.0
Stepwise(HKB)	84.0	8.0	7.4	0.6
Stepwise(LW)	76.0	13.6	8.4	2.0
Stepwise(HMO)	76.0	13.6	8.4	2.0
Stepwise(KS)	49.6	41.4	7.2	1.8
TABU(MSE)	80.0	17.8	2.0	0.2
TABU(penalty)	97.4	0.0	2.6	0.0

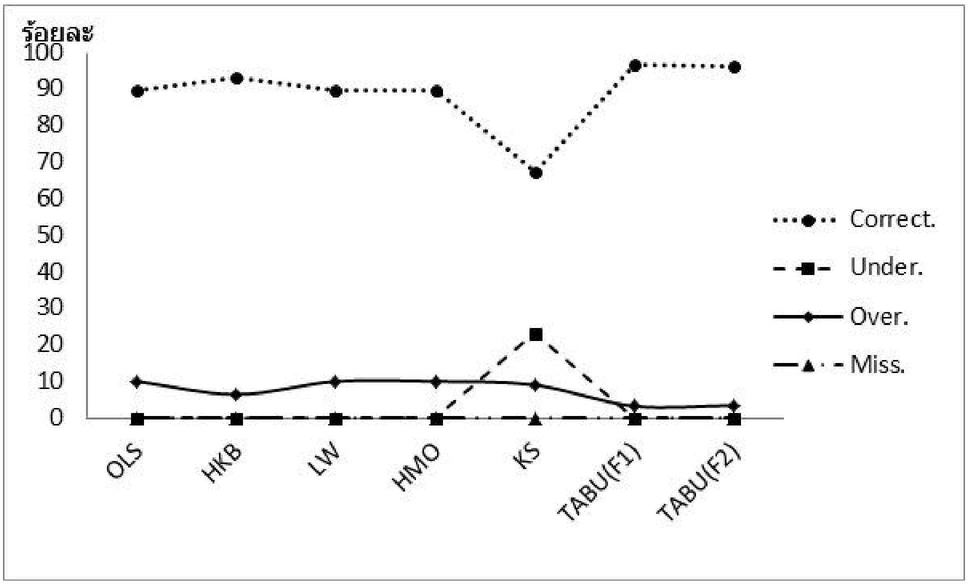
หมายเหตุ: *ρ_{13} คือค่าสัมประสิทธิ์สหสัมพันธ์ระหว่าง X_1 และ X_3



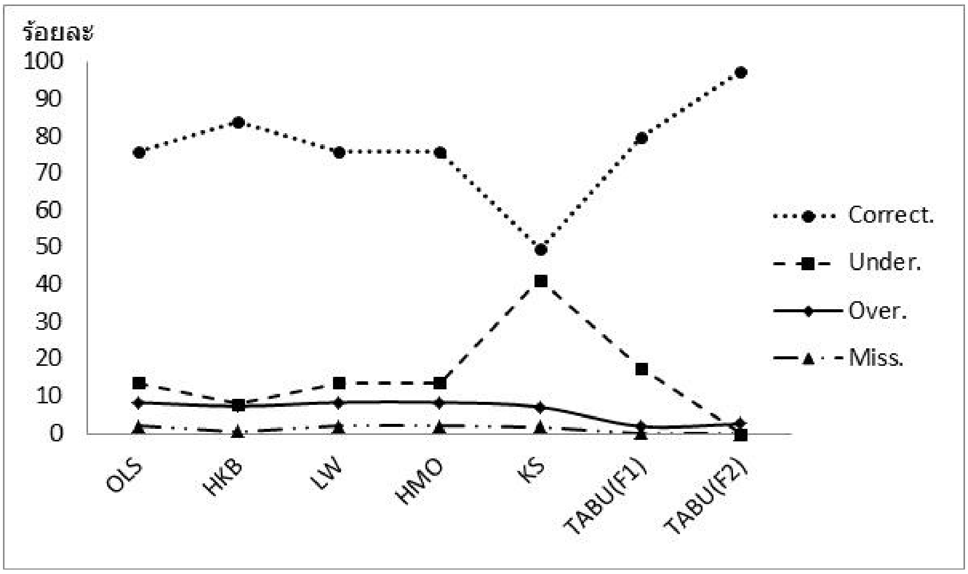
ภาพที่ 4.9 ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบ จำแนกตามวิธีการคัดเลือกตัวแบบเมื่อ n=100, ρ_{13}=0.95



ภาพที่ 4.10 ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบ จำแนกตามวิธีการคัดเลือกตัวแบบเมื่อ n=100, ρ_{13}=0.99

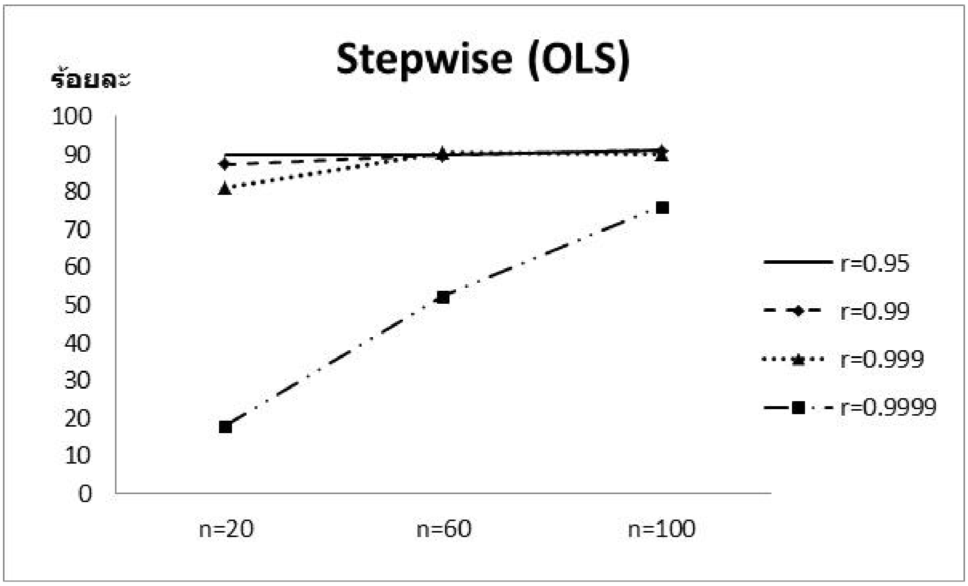


ภาพที่ 4.11ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบ จำแนกตามวิธีการคัดเลือกตัวแบบเมื่อ n=100, ρ_{13}=0.999

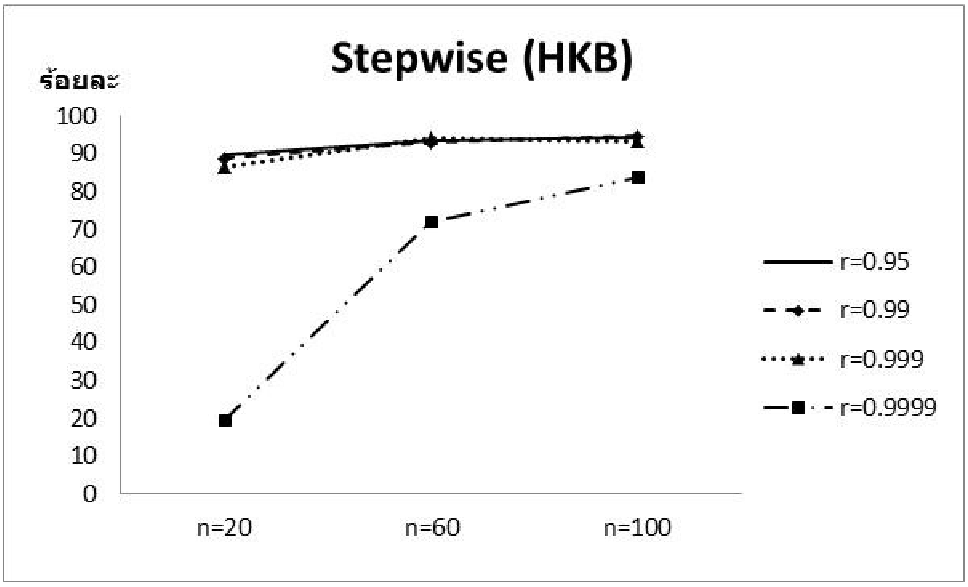


ภาพที่ 4.12 ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบ จำแนกตามวิธีการคัดเลือกตัวแบบเมื่อ n=100, ρ_{13}=0.9999

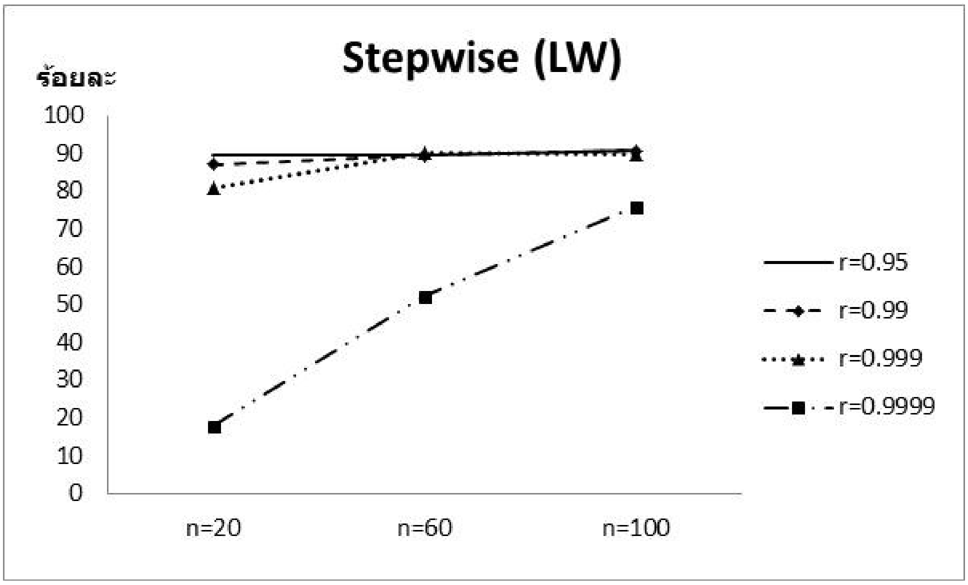
จากผลการศึกษาสามารถสรุปได้ว่า วิธีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบด้วยวิธีการถดถอยแบบขั้นตอน มีร้อยละของการคัดเลือกตัวแปรเข้าสู่ตัวแบบได้ถูกต้องสูงขึ้นเมื่อขนาดตัวอย่างใหญ่ขึ้น แต่จะมีค่าลดลงเมื่อค่าสัมประสิทธิ์สหสัมพันธ์ระหว่างตัวแปรอิสระ X_1 และ X_3 เพิ่มขึ้น แต่วิธีการค้นหาแบบต้องห้าม ที่มีฟังก์ชันเป้าหมายทั้งสองเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ย (MSE) มีร้อยละของการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบได้ถูกต้องค่อนข้างสูงและคงที่ ยกเว้นในกรณีที่ค่าสัมประสิทธิ์สหสัมพันธ์มีค่าสูงมากเป็น 0.9999 มีร้อยละของการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบได้ถูกต้องลดลงมาก แต่จะสูงขึ้นเมื่อขนาดตัวอย่างใหญ่ขึ้นเช่นเดียวกับวิธีการถดถอยแบบขั้นตอน แต่สำหรับวิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยปรับด้วยฟังก์ชันการลงโทษมีร้อยละของการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบได้ถูกต้องค่อนข้างสูงและคงที่ในทุกขนาดตัวอย่างและค่าสัมประสิทธิ์สหสัมพันธ์ดูรายละเอียดได้จากภาพที่ 4.13-4.19



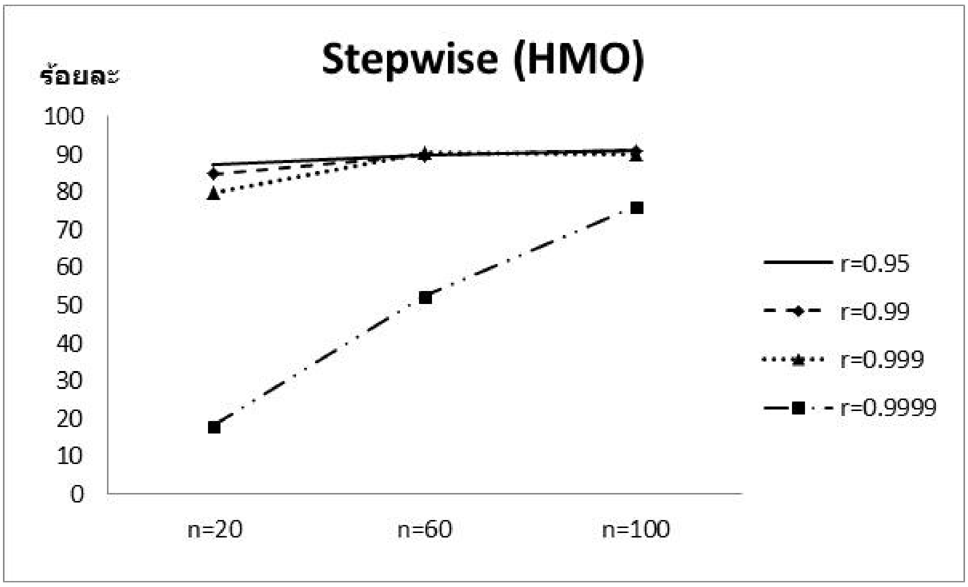
ภาพที่ 4.13 ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบโดยการประมาณค่าพารามิเตอร์แบบ OLSจำแนกตามขนาดตัวอย่างและค่าสัมประสิทธิ์สหสัมพันธ์



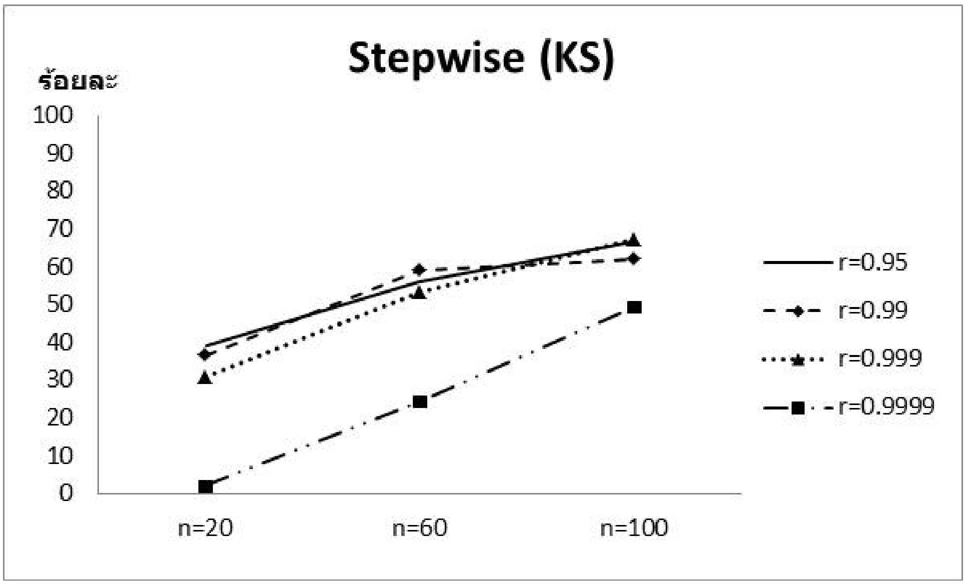
ภาพที่ 4.14 ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบโดยการประมาณค่าพารามิเตอร์วิธีโฮเอิร์ลเคนนาร์ดและบาลด์วินจำแนกตามขนาดตัวอย่างและค่าสัมประสิทธิ์สหสัมพันธ์



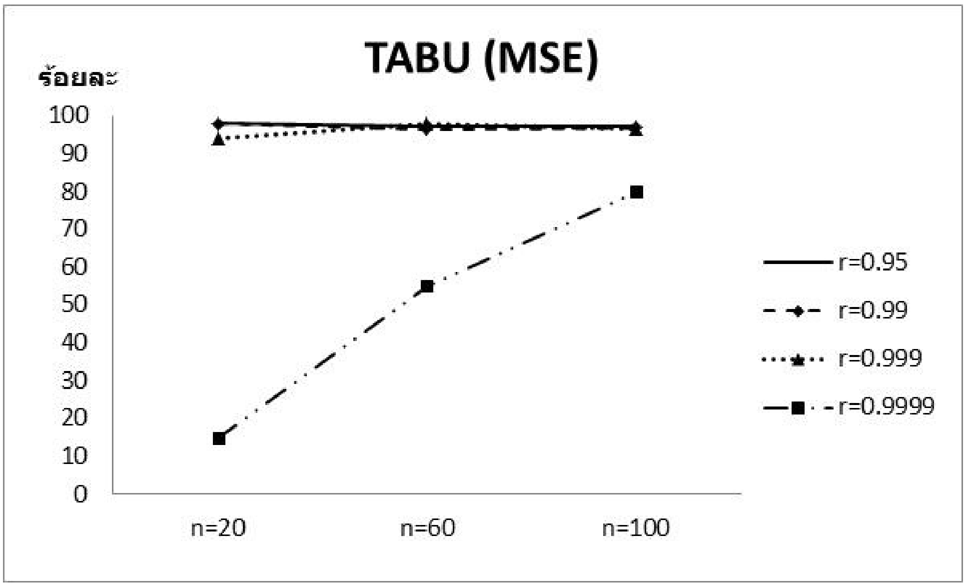
ภาพที่ 4.15ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบ โดยการประมาณค่าพารามิเตอร์วิธีลอว์เลสและแวงจำแนกตามขนาดตัวอย่างและค่าสัมประสิทธิ์สหสัมพันธ์



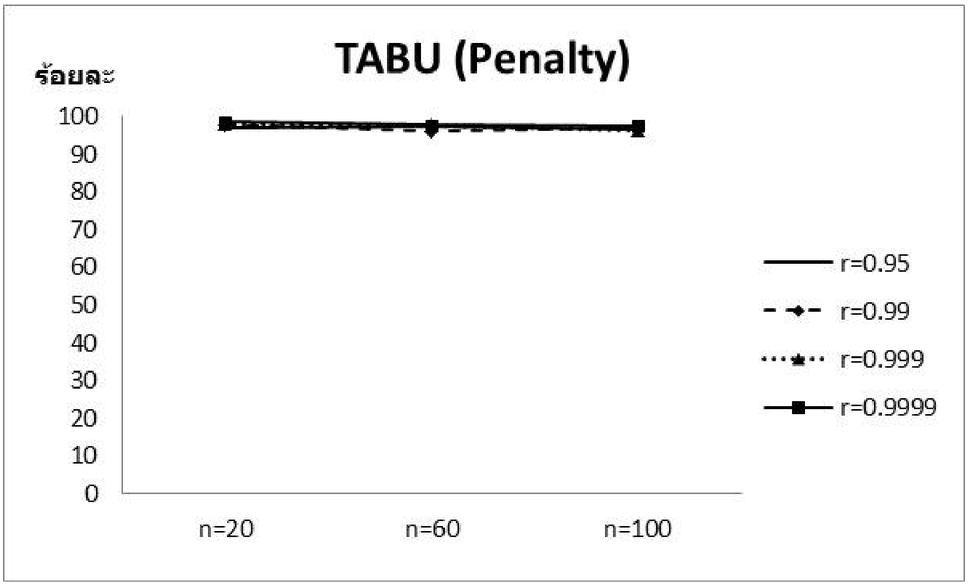
ภาพที่ 4.16ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบ โดยการประมาณค่าพารามิเตอร์วิธีนอมูระจำแนกตามขนาดตัวอย่างและค่าสัมประสิทธิ์สหสัมพันธ์



ภาพที่ 4.17ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบ โดยการประมาณค่าพารามิเตอร์วิธีคาลาฟและชูเกอร์จำแนกตามขนาดตัวอย่างและค่าสัมประสิทธิ์สหสัมพันธ์

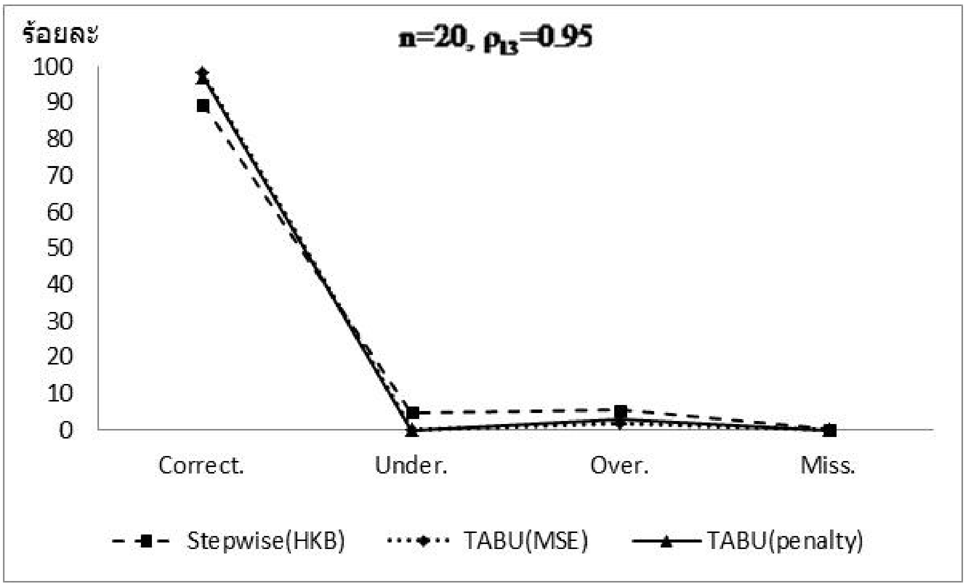


ภาพที่ 4.18ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบ โดยวิธีการค้นหาแบบต้องห้าม(MSE) จำแนกตามขนาดตัวอย่างและค่าสัมประสิทธิ์สหสัมพันธ์

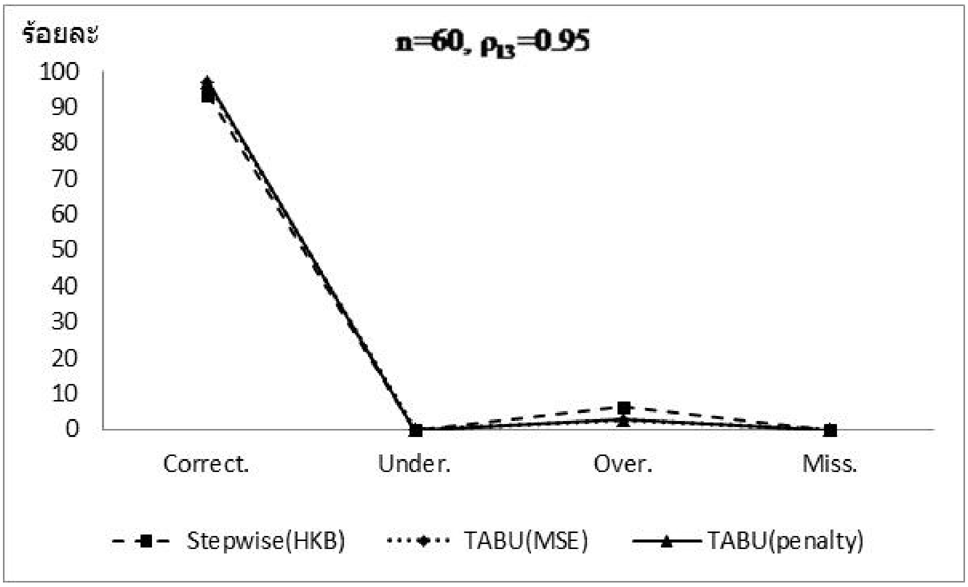


ภาพที่ 4.19 ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบโดยวิธีการค้นหาแบบต้องห้าม(MSE ปรับด้วยฟังก์ชันการลงโทษ)จำแนกตามขนาดตัวอย่างและค่าสัมประสิทธิ์สหสัมพันธ์

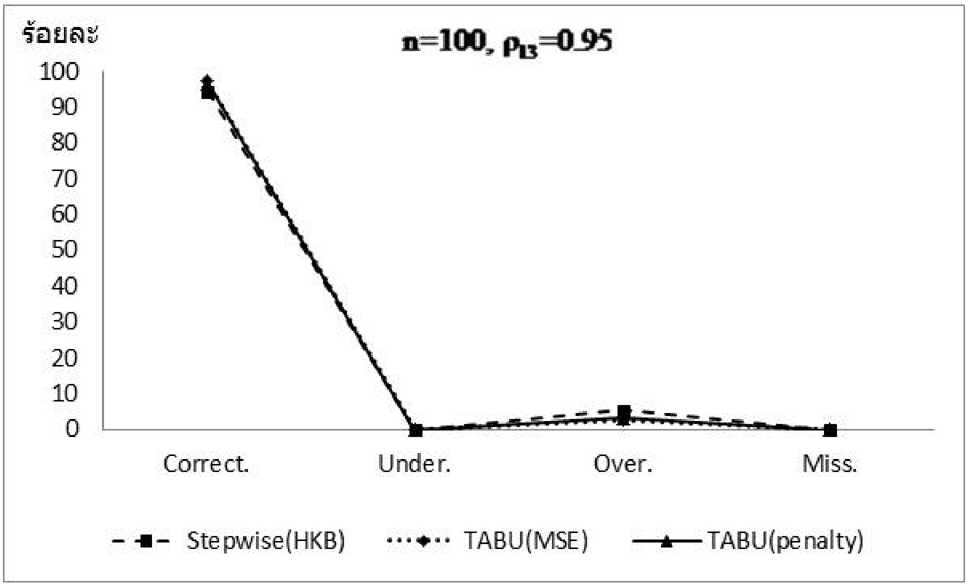
และ จากผลการศึกษาพบว่าเมื่อคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบโดยวิธีการถดถอยแบบขั้นตอนการประมาณค่าพารามิเตอร์ด้วยวิธีการถดถอยแบบริดจ์ที่ประมาณค่าคงตัว r ด้วยวิธี โฮเอิร์ล เคนนาร์ด และ บาลด์วิน ให้ร้อยละการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบได้ถูกต้องมากที่สุดในวิธีการถดถอยแบบขั้นตอนในทุกขนาดตัวอย่างและทุกค่าสัมประสิทธิ์สหสัมพันธ์ ซึ่งได้ร้อยละในการคัดเลือกใกล้เคียงกับวิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยและค่าความคลาดเคลื่อนกำลังสองเฉลี่ยปรับด้วยฟังก์ชันการลงโทษยกเว้นในกรณีที่ตัวแปรอิสระ X_1 และ X_3 มีค่าสัมประสิทธิ์สหสัมพันธ์ 0.9999 และขนาดตัวอย่างเท่ากับ 20 วิธีการถดถอยแบบขั้นตอนการประมาณค่าพารามิเตอร์ด้วยวิธีการถดถอยแบบริดจ์ที่ประมาณค่าคงตัว r ด้วยวิธี โฮเอิร์ล เคนนาร์ด และ บาลด์วิน และ วิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยมีค่าต่ำสุดมีร้อยละของการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบได้ถูกต้องลดลงมากและเพิ่มขึ้นเมื่อขนาดตัวอย่างเพิ่มขึ้น (คือมีร้อยละของตัวแบบที่ Underspecified ค่อนข้างสูงและน้อยลงเมื่อขนาดตัวอย่างเพิ่มขึ้น) ในขณะที่วิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยปรับด้วยฟังก์ชันการลงโทษมีร้อยละของการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบได้ถูกต้องมากที่สุดและค่อนข้างคงที่ในทุกขนาดตัวอย่าง ดังแสดงในภาพที่ 4.20-4.31



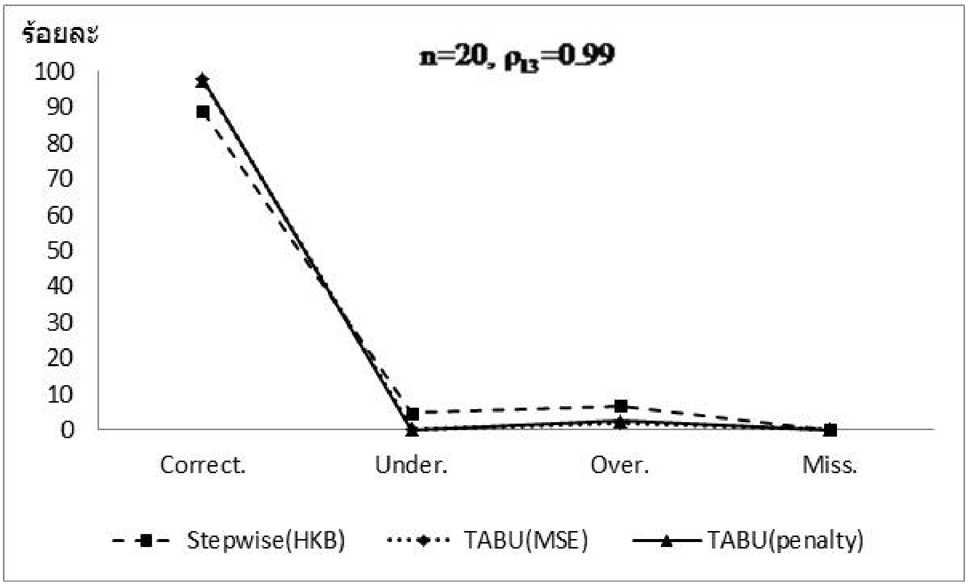
ภาพที่ 4.20ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบจำแนกตามประเภทตัวแบบที่คัดเลือกได้ เมื่อ n=20, ρ_{13}=0.95



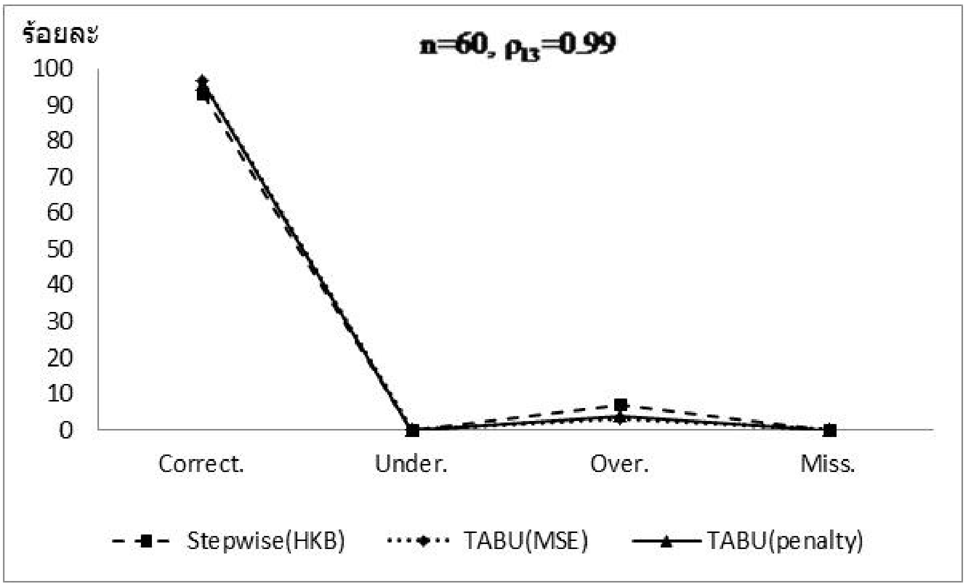
ภาพที่ 4.21ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบจำแนกตามประเภทตัวแบบที่คัดเลือกได้ เมื่อ n=60, ρ_{13}=0.95



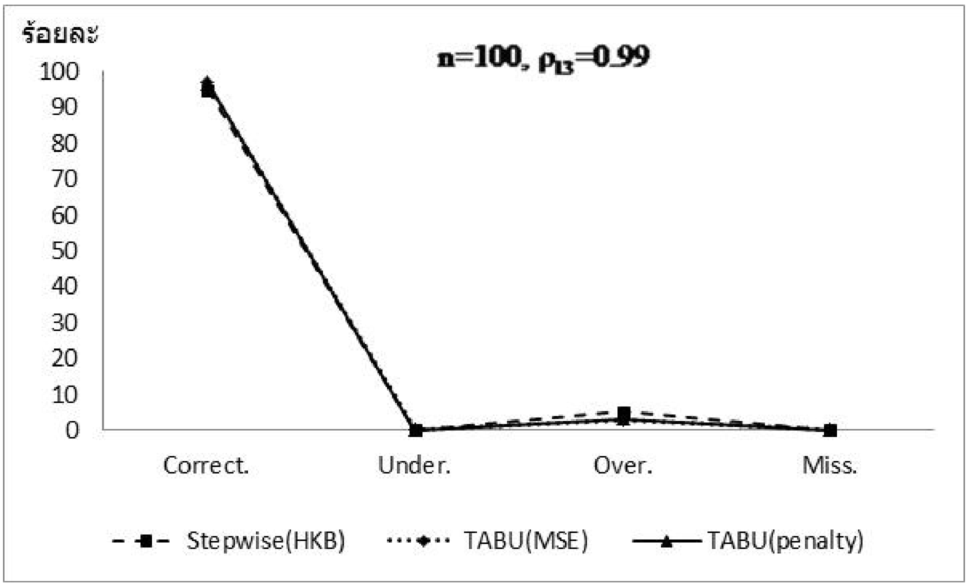
ภาพที่ 4.22ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบจำแนกตามประเภทตัวแบบที่คัดเลือกได้ เมื่อ n=100, ρ_{13}=0.95



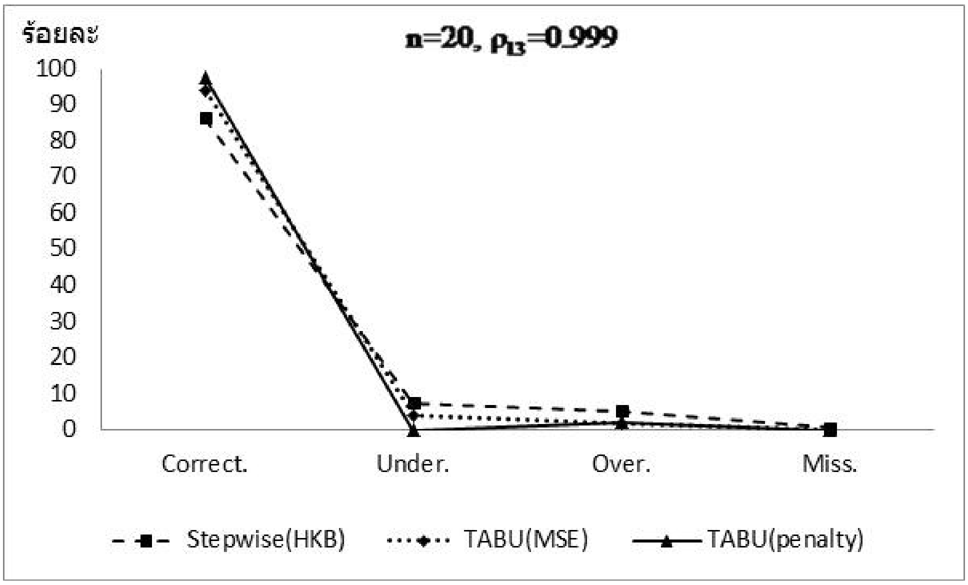
ภาพที่ 4.23ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบจำแนกตามประเภทตัวแบบที่คัดเลือกได้ เมื่อ n=20, ρ_{13}=0.99



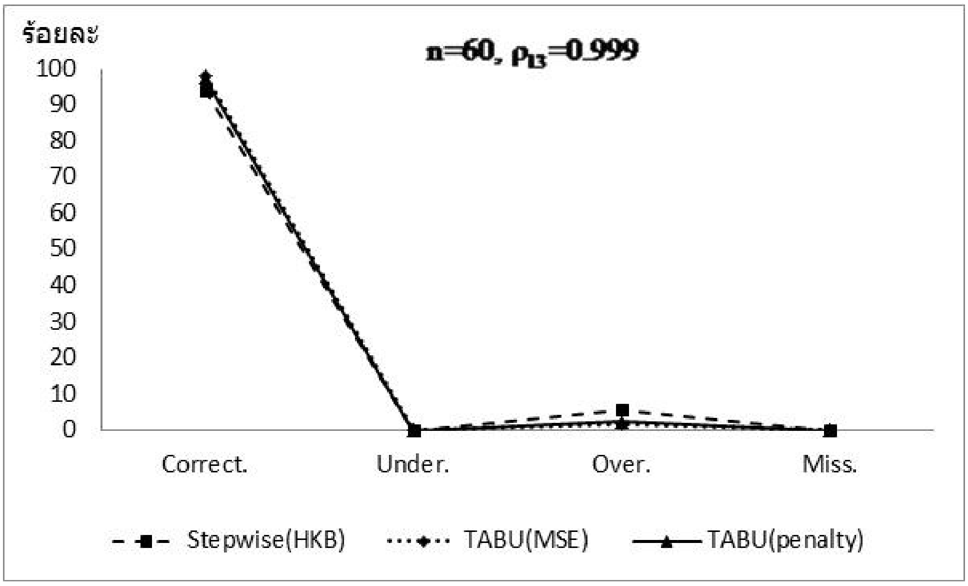
ภาพที่ 4.24ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบจำแนกตามประเภทตัวแบบที่คัดเลือกได้ เมื่อ n=60, ρ_{13}=0.99



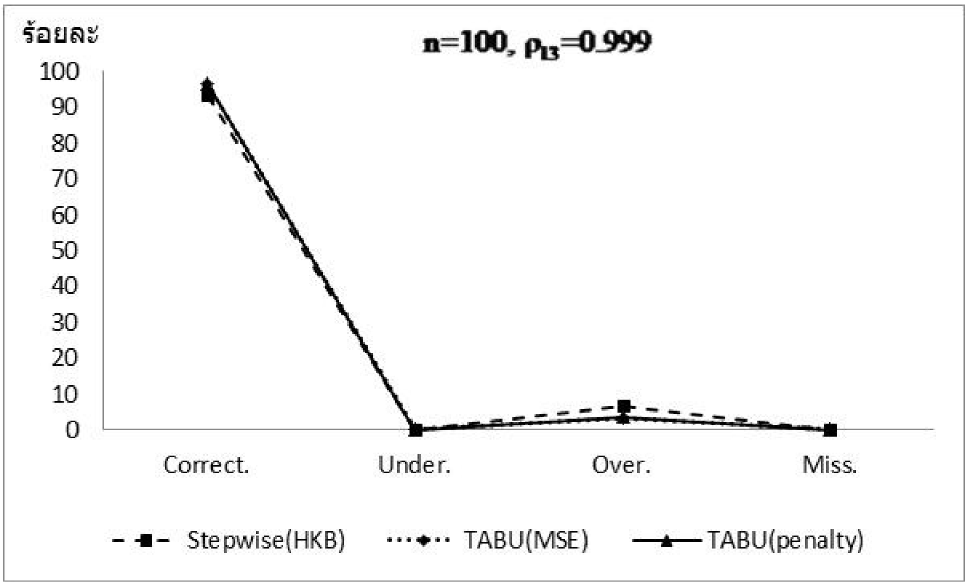
ภาพที่ 4.25ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบจำแนกตามประเภทตัวแบบที่คัดเลือกได้ เมื่อ n=100, ρ_{13}=0.99



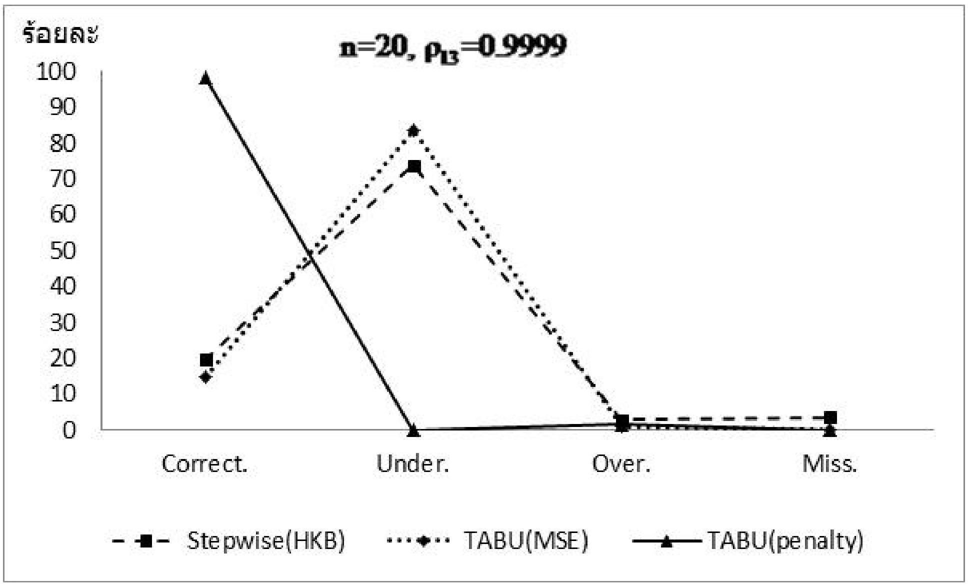
ภาพที่ 4.26ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบจำแนกตามประเภทตัวแบบที่คัดเลือกได้ เมื่อ n=20, ρ_{13}=0.999



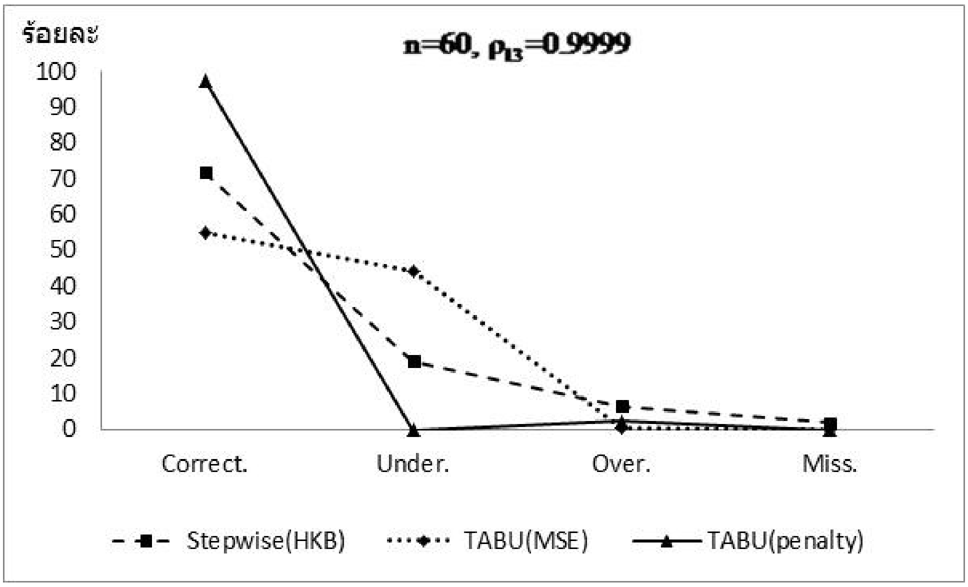
ภาพที่ 4.27ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบจำแนกตามประเภทตัวแบบที่คัดเลือกได้ เมื่อ n=60, ρ_{13}=0.999



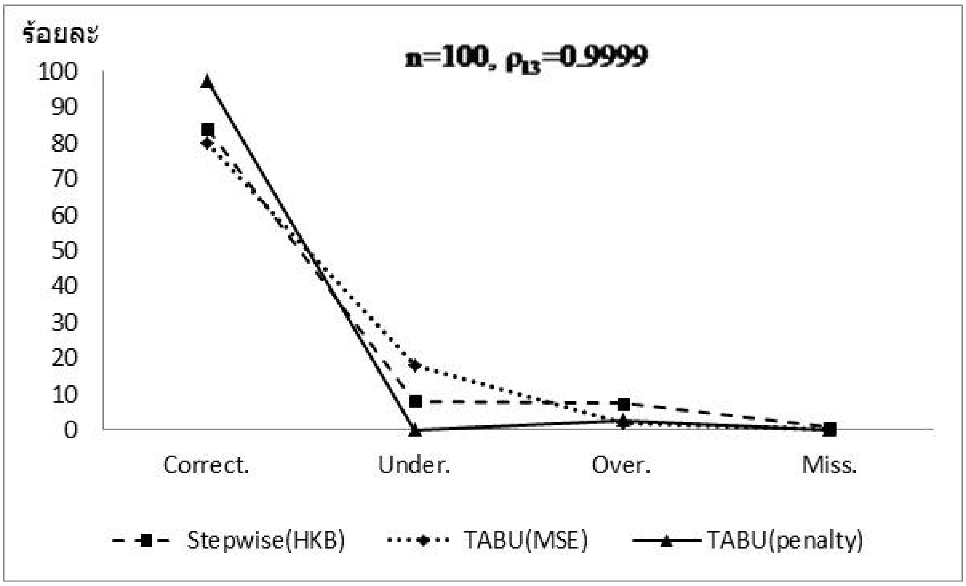
ภาพที่ 4.28ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบจำแนกตามประเภทตัวแบบที่คัดเลือกได้ เมื่อ n=100, ρ_{13}=0.999



ภาพที่ 4.29ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบจำแนกตามประเภทตัวแบบที่คัดเลือกได้ เมื่อ n=20, ρ_{13}=0.9999



ภาพที่ 4.30ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบจำแนกตามประเภทตัวแบบที่คัดเลือกได้ เมื่อ n=60, ρ_{13}=0.9999



ภาพที่ 4.31ร้อยละของจำนวนครั้งที่คัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบจำแนกตามประเภทตัวแบบที่คัดเลือกได้ เมื่อ n=100, ρ_{13}=0.9999

บทที่ 5

สรุปและอภิปรายผลการดำเนินงานวิจัย

งานวิจัยนี้ศึกษากรณีตัวแปรอิสระมีการแจกแจงเอกรูปและความคลาดเคลื่อนมีการแจกแจงปกติ จำนวนตัวแปรอิสระในตัวแบบเต็มรูปเท่ากับ 7 ตัวแปร โดยมีตัวแปรอิสระที่เกี่ยวข้องกับตัวแปรตาม5ตัวแปร และตัวแปรอิสระที่ไม่เกี่ยวข้องกับตัวแปรตาม 2 ตัวแปร ขนาดตัวอย่างในการศึกษา คือ 20, 60และ 100 ข้อมูลที่ใช้ในการวิจัยได้จากการสร้างแบบจำลอง โดยทำซ้ำ 500 ครั้งในแต่ละสถานการณ์

5.1 สรุปผล

กรณีตัวแปรอิสระ X_1 และ X_3 มีค่าสัมประสิทธิ์สหสัมพันธ์ 0.95, 0.99, 0.999 วิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยและค่าความคลาดเคลื่อนกำลังสองเฉลี่ยปรับด้วยฟังก์ชันการลงโทษ มีร้อยละของการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบได้ถูกต้องมากกว่าวิธีการถดถอยแบบขั้นตอนที่ประมาณค่าสัมประสิทธิ์การถดถอยด้วยวิธีกำลังสองน้อยที่สุดและวิธีการถดถอยแบบบริดจ์ในทุกขนาดตัวอย่าง นอกจากนั้นยังไม่มีตัวแบบ Underspecification และ Missspecification มีเพียงตัวแบบ Overspecification ซึ่งเป็นปัญหาที่รุนแรงน้อยกว่าการมีตัวแปรอิสระที่สำคัญขาดหายไปจากตัวแบบ (Underspecification) ยกเว้นกรณีที่ขนาดตัวอย่างเป็น 20 และค่าสัมประสิทธ์สหสัมพันธ์ 0.999 วิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ย มีตัวแบบ Underspecification ร้อยละ 4.0 และตัวแบบ Missspecification ร้อยละ 0.2 ในขณะที่การคัดเลือกตัวแบบด้วยวิธีการถดถอยแบบขั้นตอนมีร้อยละของตัวแบบ Underspecification ในทุกขนาดตัวอย่าง ซึ่งผลการศึกษาในส่วนนี้สอดคล้องกับผลการศึกษาของกานต์ณัฐ ณ บางช้าง ในกรณีที่ตัวแปรอิสระมีความสัมพันธ์กันในระดับ 0.95-0.99 วิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยมีร้อยละของการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบได้ถูกต้องมากกว่าวิธีการถดถอยแบบขั้นตอน

กรณีตัวแปรอิสระ X_1 และ X_3 มีค่าสัมประสิทธิ์สหสัมพันธ์ 0.9999 วิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยและวิธีการถดถอยแบบขั้นตอนที่ประมาณค่าสัมประสิทธิ์การถดถอยด้วยวิธีกำลังสองน้อยที่สุดและวิธีการถดถอยแบบริดจ์ มีร้อยละของการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบได้ถูกต้องลดลงมากและเพิ่มขึ้นเมื่อขนาดตัวอย่างเพิ่มขึ้น (คือมีร้อยละของตัวแบบ Underspecification ค่อนข้างสูงและน้อยลงเมื่อขนาดตัวอย่างเพิ่มขึ้น) แต่วิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยปรับด้วยฟังก์ชันการลงโทษ มีร้อยละของการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบได้ถูกต้องมากที่สุดและค่อนข้างคงที่ในทุกขนาดตัวอย่าง

5.2 อภิปรายผลการดำเนินงานวิจัย

เมื่อพิจารณาเปรียบเทียบประสิทธิภาพในการคัดเลือกตัวแบบจากขนาดตัวอย่าง จะพบว่าเมื่อขนาดตัวอย่างเพิ่มขึ้น วิธีการคัดเลือกตัวแปรอิสระโดยใช้วิธีการถดถอยแบบขั้นตอน มีร้อยละของการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบได้ถูกต้องสูงขึ้นและมีตัวแบบ Underspecification และ Missspecification ลดลงมาก ในขณะที่วิธีการค้นหาแบบต้องห้ามที่ใช้ฟังก์ชันเป้าหมายทั้ง 2 ฟังก์ชันมีร้อยละของการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบได้ถูกต้องค่อนข้างคงที่ในทุกขนาดตัวอย่างและไม่มีตัวแบบ Underspecification และ Missspecification เลย ยกเว้นในกรณีที่ตัวแปรอิสระ X_1 และ X_3 มีค่าสัมประสิทธิ์สหสัมพันธ์ 0.9999

เมื่อพิจารณาเปรียบเทียบประสิทธิภาพในการคัดเลือกตัวแบบจากกรณีข้อมูลที่มีสหสัมพันธ์เชิงเส้นพหุ พบว่าเมื่อค่าสัมประสิทธิ์สหสัมพันธ์ระหว่างตัวแปรอิสระ X_1 และ X_3 เพิ่มสูงขึ้น วิธีการคัดเลือกตัวแปรอิสระโดยใช้วิธีการถดถอยแบบขั้นตอนมีร้อยละของการคัดเลือกตัวแบบได้ถูกต้องลดลง ในขณะที่วิธีการค้นหาแบบต้องห้ามจากทั้ง 2 ฟังก์ชันเป้าหมาย มีร้อยละของการคัดเลือกตัวแบบได้ถูกต้องค่อนข้างคงที่และใกล้เคียงกัน ยกเว้นในกรณีที่ตัวแปรอิสระ X_1 และ X_3 มีค่าสัมประสิทธิ์สหสัมพันธ์สูงขึ้นมากคือ 0.9999 วิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยมีร้อยละของการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบใกล้เคียงกับวิธีการถดถอยแบบขั้นตอนคือมีร้อยละของการคัดเลือกได้ถูกต้องลดลงมากและมีร้อยละการคัดเลือกตัวแบบที่ Underspecification เพิ่มขึ้นมาก เมื่อขนาดตัวอย่างเป็น 20 แต่เมื่อขนาดตัวอย่างเพิ่มขึ้นจะมีร้อยละการคัดเลือกตัวแปรเข้าสู่ตัวแบบได้ถูกต้องมากขึ้นและมีร้อยละของตัวแบบ Underspecification ลดลง

ดังนั้นสามารถสรุปได้ว่าวิธีการคัดเลือกตัวแปรอิสระเข้าสู่ตัวแบบโดยใช้วิธีการค้นหาแบบต้องห้ามที่ใช้ฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ยปรับด้วยฟังก์ชันการลงโทษมีประสิทธิภาพในการคัดเลือกตัวแปรอิสระได้ดีที่สุด กล่าวคือสามารถคัดเลือกตัวแปรเข้าสู่ตัวแบบได้ถูกต้องมากที่สุดและค่อนข้างคงที่ในทุกกรณี โดยความถูกต้องในการคัดเลือกไม่ได้ขึ้นอยู่กับขนาดตัวอย่างและระดับสหสัมพันธ์ระหว่างตัวแปรอิสระอย่างเช่นในวิธีการถดถอยแบบขั้นตอนและวิธีการค้นหาแบบต้องห้ามที่มีฟังก์ชันเป้าหมายเป็นค่าความคลาดเคลื่อนกำลังสองเฉลี่ย แต่ข้อจำกัดของวิธีการค้นหาแบบต้องห้ามคือเมื่อมีข้อมูลเพียงชุดเดียว การประมาณค่าความเบี่ยงเบนมาตรฐานของตัวประมาณค่าสัมประสิทธิ์การถดถอยต้องใช้วิธีการสุ่มซ้ำ (Resampling) เช่น วิธี Bootstrap Resampling วิธี Jackknife Resampling เป็นต้น

เนื่องจากวิธีการคำนวณค่าพารามิเตอร์โดยวิธีการค้นหาแบบต้องห้ามไม่ได้ใช้วิธีการหาเมตริกซ์ผกผันของ $\mathbf{X'X}$ จึงไม่มีปัญหาเหมือนกับการประมาณค่าพารามิเตอร์โดยใช้เมตริกซ์ผกผันของ $\mathbf{X'X}$ อย่างเช่นในวิธีกำลังสองน้อยที่สุดและวิธีการถดถอยแบบริดจ์ เมื่อตัวแปรอิสระมีปัญหาสหสัมพันธ์เชิงเส้นพหุเพราะเมื่อข้อมูลเปลี่ยนแปลงไปเล็กน้อยจะทำให้ค่าเมตริกซ์ผกผันของ $\mathbf{X'X}$ เปลี่ยนไปมากส่งผลให้ค่าประมาณพารามิเตอร์มีค่าไม่เสถียร จึงทำให้ค่าประมาณพารามิเตอร์เชื่อถือไม่ได้และส่งผลถึงการทดสอบสมมุติฐานด้วย จึงอาจสรุปได้ว่าในกรณีที่ข้อมูลมีสหสัมพันธ์เชิงเส้นพหุกันสูง วิธีการค้นหาแบบต้องห้ามช่วยในการประมาณค่าสัมประสิทธิ์การถดถอยได้ใกล้เคียงกับค่าพารามิเตอร์ของตัวแบบมากกว่าวิธีกำลังสองน้อยที่สุดและวิธีการถดถอยแบบริดจ์

5.3 ข้อเสนอแนะ

ในการวิจัยครั้งต่อไป ควรศึกษากรณีตัวแบบการถดถอยไม่เป็นไปตามข้อตกลงเบื้องต้นอื่นๆ เช่น ความคลาดเคลื่อนไม่ได้มีการแจกแจงแบบปกติ ความแปรปรวนของความคลาดเคลื่อนมีค่าไม่คงที่ เป็นต้น นอกจากนี้อาจใช้วิธีการประมาณค่าพารามิเตอร์ริดจ์ด้วยวิธีการอื่นๆ เช่น วิธี McDonald and Galarneau, Khuri and Myers เป็นต้น หรือใช้วิธีการคัดเลือกตัวแบบด้วยวิธีการอื่นๆ เช่น วิธีการค้นหาคำตอบแบบเจนเนติก (Genetic Algorithm) วิธีการค้นหาคำตอบแบบ Simulated Annealing และกำหนดระดับนัยสำคัญในการทดสอบเป็นค่าอื่นๆ

บรรณานุกรม

กานต์ณัฐ ณ บางช้าง. 2554. **การคัดเลือกตัวแปรในตัวแบบการถดถอยเชิงเส้นพหุด้วยวิธีการค้นหาแบบต้องห้าม**. วิทยานิพนธ์ปริญญามหาบัณฑิต สถาบันบัณฑิตพัฒนบริหารศาสตร์.

จิรายุส พู่มนตรี. 2533. **การเปรียบเทียบตัวประมาณริดจ์สำหรับการวิเคราะห์การถดถอยแบบริดจ์.** วิทยานิพนธ์ปริญญามหาบัณฑิต จุฬาลงกรณ์มหาวิทยาลัย.

ธันยากร ต้นชลขันธ์. 2538. **การเปรียบเทียบการประมาณค่าสัมประสิทธิ์การถดถอยพหุโดยใช้วิธีกำลังสองน้อยที่สุด วิธีการถดถอยแบบริดจ์ และวิธีที่ใช้หลักการของริดจ์และสไตน์ ในกรณีเกิดพหุสัมพันธ์ระหว่างตัวแปรอิสระ**. วิทยานิพนธ์ปริญญามหาบัณฑิต จุฬาลงกรณ์มหาวิทยาลัย.

นุสรา สถิตโพธิ์ศรี. 2535. **การเปรียบเทียบตัวประมาณค่าสัมประสิทธิ์ความถดถอยพหุโดยวิธีการถดถอยแบบริดจ์และวิธีการถดถอยแบบลาเท้นรูทธ์ ในกรณีที่เกิดพหุสัมพันธ์ระหว่างตัวแปรอิสระ**. วิทยานิพนธ์ปริญญามหาบัณฑิต จุฬาลงกรณ์มหาวิทยาลัย.

บุญจิรา มากอ้น. 2545. **การเปรียบเทียบเกณฑ์การคัดเลือกตัวแบบการถดถอยแบบไม่ติดกลุ่ม.** วิทยานิพนธ์ปริญญามหาบัณฑิต จุฬาลงกรณ์มหาวิทยาลัย.

บุษยา ปภาพจน์. 2548. **การเปรียบเทียบวิธีการคัดเลือกตัวแบบที่ดีที่สุดในการวิเคราะห์การถดถอยเชิงเส้นพหุ**. วิทยานิพนธ์ปริญญามหาบัณฑิต จุฬาลงกรณ์มหาวิทยาลัย.

เบ็ญจวรรณ ชัยกิจ. 2545. **การเปรียบเทียบตัวประมาณริดจ์ สำหรับการถดถอยแบบริดจ์โดยเน้นแนวคิดแบบเบส์**. วิทยานิพนธ์ปริญญามหาบัณฑิต จุฬาลงกรณ์มหาวิทยาลัย.

ภัทรสุดา สุดแสน. 2548. **การเปรียบเทียบเกณฑ์การคัดเลือกตัวแบบการถดถอยเชิงเส้นเมื่อตัวอย่างมีขนาดเล็ก**. วิทยานิพนธ์ปริญญามหาบัณฑิต จุฬาลงกรณ์มหาวิทยาลัย.

วรรัตน์ ราชกิจจา. 2547. **การเปรียบเทียบการประมาณค่าพารามิเตอร์ในการวิเคราะห์ความถดถอยพหุคูณโดยวิธีเอ็มพิริคัลเบส์ริดจ์รีเกรสชัน เมื่อเกิดพหุสัมพันธ์**. วิทยานิพนธ์ปริญญามหาบัณฑิต จุฬาลงกรณ์มหาวิทยาลัย.

วริดา พลาศรี. 2552. **การประมาณค่าสัมประสิทธิ์ความถดถอยเชิงพหุเมื่อเกิดพหุสัมพันธ์ด้วยวิธีความถดถอยบูตสแตรปแบบริดจ์**. วิทยานิพนธ์ปริญญามหาบัณฑิต จุฬาลงกรณ์มหาวิทยาลัย.

วลัยทิพย์ บุญญาติศัย. 2549. **การคัดเลือกสมการเชิงเส้นที่ดีที่สุดภายใต้แลตทิซ**. วิทยานิพนธ์ปริญญามหาบัณฑิต จุฬาลงกรณ์มหาวิทยาลัย.

สุรสิทธิ์ ฤทธิ์สมิตชัย. 2551. **การเปรียบเทียบเกณฑ์การคัดเลือกตัวแบบความถดถอยเชิงเส้นด้วยแนวคิดเชิงแลตทิซ**. วิทยานิพนธ์ปริญญามหาบัณฑิต จุฬาลงกรณ์มหาวิทยาลัย.

Cetin, M. and Erar, A. 2006. A Simulation Study on Classic and Robust Variable Selection in Linear Regression. **Applied Mathematics and Computation**. 175 (15 April): 1629-1643.

Drezner, Z. and George, A. 1999. Tabu Search Model Selection in Multiple Regression Analysis. **Communication in Statistics - Simulation and Computation**. 28 (April): 349-367.

Eksioglu, B.; Demirer, R. and Capar, I. 2005. Subset Selection in Multiple Linear Regression: A New Mathematical Programming Approach. **Computer & Industrial Engineering**. 49 (August): 155-167.

Games, P. A. and Lerner, J. V. 1981. **Maximum R^2 Improvement and Stepwise Multiple Regression as Related to Over-Fitting**. Boston: Department of Counseling Development and Education Psychology, Boston College

Ghani, I. M. M. and Ahmad, Sabri. 2010. Stepwise Multiple Regression Method to Forecast Fish Landing. **Social and Behavioral Sciences**. 8: 549-554.

Glover, F. 1990. Tabu Search: A Tutorial. **Interfaces**. 20 (November): 74-94.

Han, S. P. 1986. On the Augmented Lagrangian. **Mathematics of Operations Research**. 11 (February): 161-168.

Hedar, A. and Fukushima, M. 2003. TS Directed by Direct Search Methods for Nonlinear Global Optimization. **Technical Report**. 7 (June): 7-18.

Hoerl, Arthur E. and Kennard, Robert W. 1970a. Ridge Regression: Biased Estimation to Nonorthogonal Problem. **Technometrics**. 12 (February): 55-67.

Hoerl, Arthur E. and Kennard, Robert W. 1970b. Ridge Regression: Applications to Nonorthogonal Problem. **Technometrics**. 12 (February): 69-83.

Hoerl, Arthur E.; Kennard, Robert W. and Baldwin, Kent F. 1975. Ridge Regression: Some Simulations. **Communications in Statistics.** 4 (June): 105-123.

Holland, J. 1975. **Adaptation in Natural and Artificial Systems**. MA, USA: The University of Michigan Press

Kapetanious, G. 2007. Variable Selection in Regression Models Using Nonstandard Optimization of Information Criteria. **Computational Statistics & Data Analysis**. 52 (September): 4-15.

Khalaf, G. and Shukur, G. 2005. Choosing Ridge Parameter for Regression Problem. **Communications in Statistics–Theory and Methods.** 34 (Septmber): 1177-1182.

Khuri, A. I. and Myers, Raymond H. 1979. Modified Ridge Analysis. **Technometrics**. 21 (November): 467-473.

Knox, J. 1989. **The Application of Tabu Search to the Symmetric Travelling Salesman Problem**. Boulder: Graduate School of Business, University of Colorado.

Lawless, J. and Wang, P. 1976. A Simulation Study of Ridge and Other Regression Estimators**.** **Communication Statistics–Theory and Methods**. 5 (January): 307-323.

Lee, Tze-San. 1986. Optimum Ridge Parameter Selection. **Journal of the Royal Statistical Society. Series C (Applied Statistics)**. 36 (September): 112-118

McDonald, G. and Galarneau, D. 1975. A Monte Carlo Evaluation of Some Ridge Type Estimators. **Journal of the American Statistical Association.** 70 (June): 407-416.

Montgomery, D. C. and Peck, E. A. 1992. **Introduction to Linear Regression Analysis**. 2nd ed. New York: John Willey & Sons.

Montgomery, D. C.; Peck, E. A. and Vining, G. G. 2006. **Introduction to Linear Regression Analysis**. 4th ed. New Jersey: John Willey & Sons.

Nomura, M. 1988. On the Almost Unbiased Ridge Regression Estimation. **Communication Statistics–Simulations.** 17 (January): 729-743.

Pacheco, J.; Casado, S. and Nunez, L. 2008. A Variable Selection Method Based on Tabu Search for Logistic Regression Models. **European Journal of Operational Research**. 199 (December): 506-511.

Pasha, G. R. 2002. Selection of Variables in Multiple Regression Using Stepwise Regression. **Journal of Research (Science)**. 13 (December): 119-127.

Siary, P. and Berthiau, G. 1997. Fitting of Tabu Search to Optimize Functions of Continuous Variables. **International Journal for Numerical Methods in Engineering.** 40 (July): 2449-2457.

Wasserman, G. S. and Sudijanto, A. 1994. All Subsets Regression Using a Genetic Search Algorithm. **Computers and Industrial Engineering**. 27 (September): 489-492.

Wichern, D. and Churchill, G. 1978. A Comparison of Ridge Estimators. **Technometrics**. 20 (August): 301-311.

ภาคผนวก

ภาคผนวก

โปรแกรมที่ใช้ในการสร้างและวิเคราะห์ข้อมูล

โปรแกรมหลักที่ใช้ในการจำลองข้อมูล ประมาณค่าพารามิเตอร์และคัดเลือกตัวแปร

```
%% Add path
addpath(genpath(fullfile(pwd,'library')));
loadpath = 'data/2012.07.26/';
savepath = 'data/2012.07.26/';
%% Initial timer variable
tpop=0;tdataset=0;
tStepwise=0;tOLSnRidge=0;
tsave=0;tload=0;
tTabu=0;tTabuVarSel=0;texport=0;tnTabuVarSel=0;
%% Generate Population (Rho specified)
tic;
beta = [30;20;10;-5;15;4;0;0];
beta = [30;20;10;-55;15;4;0;0];
xDist = [20,100;10,120;5,90;-30,30;-60,60;-20,40;-50,50];
eDist = [0,30];rho = 0.9999;
pop = genPopwRho(200000,beta,xDist,eDist,rho);
filename = 'popwRho.mat';
save([savepath,filename],'pop');tpop = toc;
%% Generate Dataset (Rho specified)
tic;
temp = load([loadpath,'popwRho.mat']);pop = temp.pop;
dataset = genDatasetwRho(pop,[20 40 60 80 100],500);
```

```
filename = 'datasetwRho.mat';
save([savepath,filename],'dataset');tdataset = toc;
%% Load DATA
tic;
filename = 'datasetwRho.mat';
temp = load([loadpath,filename]);
dataset = temp.dataset;clear temp;
tload = toc;
%% Main
%% Compute OLS & Ridge (correlation case)
tic;
dataset = computeOLSnRidge(dataset);
tOLSnRidge = toc;
% Save DATA
tic;
save([savepath,filename],'dataset');
tsave=tsave+toc;
%% Variable Selection (OLS & Ridge)
tic;
dataset = OLSnRidgeVarSel(dataset);
tStepwise=toc;
% Save DATA
tic;
save([savepath,filename],'dataset');
tsave=tsave+toc;
%% Compute tabu search
tic;
% Upper & Lower bound of beta
u = [50 40 20  0  50 10 5  10];
l = [10 0  0  -10 0  0 -5 -10];
```

```
% u = [50 40 20  -20  50 10 5  10];
% l = [10 0  0  -70 0  0 -5 -10];
dataset = computeTabu(dataset,u,l);
dataset = ncomputeTabu(dataset,u,l);
tTabu=toc;
% Save DATA
tic;
save([savepath,filename],'dataset');
tsave=tsave+toc;
%% Variable Selection (T-test) for tabu search
tic;
% Upper & Lower bound of beta
 u = [50 40 20  0  50 10 5  10];
 l = [10 0  0  -10 0  0 -5 -10];
dataset = TabuVarSel(dataset,u,l);
tic;
save([savepath,filename],'dataset');
tsave=tsave+toc;
dataset = nTabuVarSel(dataset,u,l);
tTabuVarSel=toc;
% Save DATA
tic;
save([savepath,filename],'dataset');
tsave=tsave+toc;
%% Export to xls
tic;
dataset2xls_summary(savepath,filename,dataset);
dataset2xlswRho(savepath,filename,dataset);
dataset2xls_data(savepath,filename,dataset);
texport=toc;
```

```
%% Time Summary
clc;
t = [tpop tdataset tOLSnRidge tStepwise tTabu tTabuVarSel tload tsave texport]';
tall = sum(t);
t_P = t/tall*100;
txt1 = {'','','';'Time','Minutes','%'};
txt2 = {'Gen Pop';'Gen Dataset';'Compute OLS & Ridge';'Stepwise';...
   'Tabu Search';'Tabu Variable Sel.';'Load';'Save';'Export';'';'Total'};
txt3 = cell(numel(t)+2,1);
for i=1:numel(t)
   txt3{i} = sprintf('%5.0f min. %2.0f sec.',t(i)/60,mod(t(i),60));
end
txt3{i+1} = NaN;
txt3{i+2} = sprintf('%5.0f min. %2.0f sec.',tall/60,mod(tall,60));
txt4 = cell(numel(t)+2,1);
for i=1:numel(t)
   txt4{i} = sprintf('%3.2f%%',t_P(i));
end
txt4{i+1} = NaN;
txt4{i+2} = sprintf('%3.2f%%',sum(t_P));
txt = [txt1;txt2 txt3 txt4];disp(txt);
%% Hibernate computer when finish
% system('shutdown -h -f');
```

ฟังก์ชันที่ถูกเรียกไปใช้ในโปรแกรมหลัก

โปรแกรมที่ใช้ในการจำลองข้อมูล

```
function pop = genPopwRho(Npop,beta,xDist,eDist,rho,randomControl)
% Generate population with correlation specified
% using t-copula
%
```

```
% Syntax:
% pop = genData(Npop,beta,xDist,eDist,rho)
% pop = genData(Npop,beta,xDist,eDist,rho,s)
%
% Description:
% Npop is number of population
% rho is desire correlation between x1 & x3
% pop is population structure
% s is random control generation
%
% Example:
% pop = genPopwCorr(100000,0.999)
%
% See also genPop,rng
%%
% If not assign random control generation
if nargin<6
   randomControl = rng;
end
% Config random control generation
rng(randomControl);
% Save random control generation
pop.rng = randomControl;
% Start clock
tAll=tic;
% Create population
% description
pop.gendate = datestr(clock);
pop.detail = sprintf('x1 and x3 has %f correlation',rho);
pop.rhoDesire = rho;
```

```
% number of population
pop.N = Npop;
% Epsilon (normal distribution)
pop.ep.info.dist = sprintf('N(%d,%d)',eDist(1),eDist(2));
pop.ep.data = normrnd(eDist(1),sqrt(eDist(2)),pop.N,1);
pop.ep.info.min = min(pop.ep.data);
pop.ep.info.max =max(pop.ep.data);
pop.ep.info.sd = std(pop.ep.data);
pop.ep.info.mu = mean(pop.ep.data);
% X (uniform distribution)
[r c] = size(xDist);
for i=1:r
    if i~=1 && i~=3
        pop.x{i}.info.dist = sprintf('U(%d,%d)',xDist(i,1),xDist(i,2));
        pop.x{i}.data = unifrnd(xDist(i,1),xDist(i,2),pop.N,1);
        pop.x{i}.info.min = min(pop.x{i}.data);
        pop.x{i}.info.max = max(pop.x{i}.data);
        pop.x{i}.info.sd = std(pop.x{i}.data);
        pop.x{i}.info.mu = mean(pop.x{i}.data);
    end
end
% Gennerate U(0,1) using t copula with degree of freedom = 5
x1nx3 = copularnd('t',pop.rhoDesire,1,pop.N);
% x1nx3 = copularnd('Gaussian',pop.rhoDesire,pop.N);
% x1nx3 = copularnd('gaussian',rho,pop.N);
x1 = x1nx3(:,1);
x3 = x1nx3(:,2);
% adjust to desire range
x1 = (x1*(xDist(1,2)-xDist(1,1)))+xDist(1,1);
x3 = (x3*(xDist(3,2)-xDist(3,1)))+xDist(3,1);
```

```
fprintf('Correaltion of x1 & x3 population = %f\n',corr(x1,x3));
pop.x{1}.info.dist = sprintf('U(%d,%d)',xDist(1,1),xDist(1,2));
pop.x{1}.data = x1;
pop.x{1}.info.min = min(x1);
pop.x{1}.info.max = max(x1);
pop.x{1}.info.sd = std(x1);
pop.x{1}.info.mu = mean(x1);
pop.x{3}.info.dist = sprintf('U(%d,%d)',xDist(3,1),xDist(3,2));
pop.x{3}.data = x3;
pop.x{3}.info.min = min(x3);
pop.x{3}.info.max = max(x3);
pop.x{3}.info.sd = std(x3);
pop.x{3}.info.mu = mean(x3);
% compute Y
pop.beta = beta;
X = zeros(pop.N,r);
for m=1:r
  X(:,m)=pop.x{m}.data;
end
% X = [pop.x{1}.data pop.x{2}.data pop.x{3}.data...
%    pop.x{4}.data pop.x{5}.data pop.x{6}.data pop.x{7}.data];
pop.y.info.detail = sprintf('y = [x0 x1-x%d]*[b0 b1-b%d] + ep',r,r);
pop.y.data = ([ones(pop.N,1) X]*pop.beta)+pop.ep.data;
pop.y.info.min = min(pop.y.data);
pop.y.info.max = max(pop.y.data);
pop.y.info.sd = std(pop.y.data);
pop.y.info.mu = mean(pop.y.data);
pop.rhoMat.info = sprintf('correlation of [y x1-%d]',r);
pop.rhoMat.data = corr([pop.y.data X]);
```

```
t=toc(tAll);
pop.gentime = t;
end
function dataset = genDatasetwRho(pop,sampleSet,n)
% sampleSet = [20,40,100];
% n = 500;
dataset = cell(1,numel(sampleSet));
for i = 1:numel(sampleSet)
    dataset{i}.gendate = datestr(clock);
    dataset{i}.n = sampleSet(i);% 20,40 or 100
    % population info
    dataset{i}.pop.rng = pop.rng;
    dataset{i}.pop.gendate = pop.gendate;
    dataset{i}.pop.detail = pop.detail;
    dataset{i}.pop.gentime = pop.gentime;
    dataset{i}.pop.N = pop.N;
    dataset{i}.pop.rhoDesire = pop.rhoDesire;
    dataset{i}.pop.rhoMat = pop.rhoMat;
    dataset{i}.pop.beta = pop.beta;
    for k=1:numel(pop.x)
        dataset{i}.pop.x{k}.info = pop.x{k}.info;
    end
    dataset{i}.pop.ep.info = pop.ep.info;
    dataset{i}.pop.y.info = pop.y.info;
    % number of dataset from population
    dataset{i}.ndataset = n;%500
    % random samples from population
    replacement = true;
    % Initial rho count
    rho095=0;rho096=0;
```

```
    rho099=0;rho0999=0;rho09999=0;
    lCount=0;mCount=0;gCount=0;count=0;
%     rhoInfo = [0.95,0.96,0.97,0.98,0.99,0.999];
    rhoInfo = [0.95,0.99,0.999];
    for k=1:numel(rhoInfo)
        dataset{i}.rho{k}.rho = rhoInfo(k);
    end
    % Inital timer
    tSamp = tic;
    while(1)
        count = count+1;
        curSel = randsample(1:pop.N,dataset{i}.n,replacement)';
        curRho = corr(pop.x{1}.data(curSel),pop.x{3}.data(curSel));
        % Save position of random sample that satisfy condition
        if curRho>0.9994
            mCount = mCount+1;
        elseif curRho<0.945
            lCount = lCount+1;
        else
%             if curRho>=0.99985 && curRho<=0.99994
%                 if rho09999<dataset{i}.ndataset
%                     rho09999 = rho09999+1;
%                     dataset{i}.rho{5}.curSel(:,rho09999) = curSel;
%                     dataset{i}.rho{5}.curRho = curRho;
%                 end
            if curRho>=0.9985 && curRho<=0.9994
                if rho0999<dataset{i}.ndataset
                    rho0999 = rho0999+1;
                    dataset{i}.rho{3}.curSel(:,rho0999) = curSel;
                    dataset{i}.rho{3}.curRho = curRho;
```

```
        end
      elseif curRho>=0.985 && curRho<=0.994
        if rho099<dataset{i}.ndataset
          rho099 = rho099+1;
          dataset{i}.rho{2}.curSel(:,rho099) = curSel;
          dataset{i}.rho{2}.curRho = curRho;
        end
%       elseif curRho>=0.975 && curRho<=0.984
%         if rho098<dataset{i}.ndataset
%           rho098 = rho098+1;
%           dataset{i}.rho{4}.curSel(:,rho098) = curSel;
%           dataset{i}.rho{4}.curRho = curRho;
%         end
%       elseif curRho>=0.965 && curRho<=0.974
%         if rho097<dataset{i}.ndataset
%           rho097 = rho097+1;
%           dataset{i}.rho{3}.curSel(:,rho097) = curSel;
%           dataset{i}.rho{3}.curRho = curRho;
%         end
%       elseif curRho>=0.955 && curRho<=0.964
%         if rho096<dataset{i}.ndataset
%           rho096 = rho096+1;
%           dataset{i}.rho{2}.curSel(:,rho096) = curSel;
%           dataset{i}.rho{2}.curRho = curRho;
%         end
      elseif curRho>=0.945 && curRho<=0.954
        if rho095<dataset{i}.ndataset
          rho095 = rho095+1;
          dataset{i}.rho{1}.curSel(:,rho095) = curSel;
          dataset{i}.rho{1}.curRho = curRho;
```

```
        end
      else
        gCount = gCount+1;
      end
    end
    % Show rho count
    clc
    Etime = toc(tSamp);
    fprintf('[%d samples dataset][Pop Rho=%f]\n[i=%d][%.2f sec.][%.2f Hz]\n',...
      dataset{i}.n,pop.rhoMat.data(2,4),count,Etime,count/Etime);
    fprintf('Current rho = %f\n',curRho);
    fprintf('Rho095 = %3d\n',rho095);
%     fprintf('Rho096 = %3d\n',rho096);
%     fprintf('Rho097 = %3d\n',rho097);
%     fprintf('Rho098 = %3d\n',rho098);
    fprintf('Rho099 = %3d\n',rho099);
    fprintf('Rho0999 = %3d\n',rho0999);
%     fprintf('Rho09999 = %3d\n',rho09999);
    fprintf('Rho >0.9994 = %d\n',mCount);
    fprintf('Rho in gap = %d\n',gCount);
    fprintf('Rho <0.945 = %d\n',lCount);
    % Compute est. iteration
%     minP = min([rho095,rho096,rho097,rho098,rho099,rho0999,rho0999]/count);
    minP = min([rho095,rho099,rho0999]/count);
    minP = round(minP*10000)/10000;
    estCount = dataset{i}.ndataset/minP;
    elapsed = floor((estCount-count)/(count/Etime));
    fprintf('Estimated iteration is %.0f (%d min %d sec)\n',...
      estCount,floor(elapsed/60),mod(elapsed,60));
    %
```

```
%         if rho099==dataset{i}.ndataset && rho098==dataset{i}.ndataset && ...
%             rho097==dataset{i}.ndataset && rho096==dataset{i}.ndataset && ...
%             rho095==dataset{i}.ndataset && rho0999==dataset{i}.ndataset && ...
%             rho09999==dataset{i}.ndataset
%             break;
%         end
        if rho099==dataset{i}.ndataset &&  ...
           rho095==dataset{i}.ndataset && rho0999==dataset{i}.ndataset
            break;
        end
    end
    % Construct rho condition data
    for k=1:numel(rhoInfo)
        for m=1:numel(pop.x)
            dataset{i}.rho{k}.x{m} = ...
                pop.x{m}.data(dataset{i}.rho{k}.curSel);
        end
        dataset{i}.rho{k}.ep = pop.ep.data(dataset{i}.rho{k}.curSel);
        dataset{i}.rho{k}.y = pop.y.data(dataset{i}.rho{k}.curSel);
    end
    dataset{i}.gentime = Etime;
    dataset{i}.gencount = count;
end
end
function dataset = genDatasetwRho_9999(pop,sampleSet,n)
% sampleSet = [20,40,100];
% n = 500;
dataset = cell(1,numel(sampleSet));
for i = 1:numel(sampleSet)
    dataset{i}.gendate = datestr(clock);
```

```
    dataset{i}.n = sampleSet(i);% 20,40 or 100
    % population info
    dataset{i}.pop.rng = pop.rng;
    dataset{i}.pop.gendate = pop.gendate;
    dataset{i}.pop.detail = pop.detail;
    dataset{i}.pop.gentime = pop.gentime;
    dataset{i}.pop.N = pop.N;
    dataset{i}.pop.rhoDesire = pop.rhoDesire;
    dataset{i}.pop.rhoMat = pop.rhoMat;
    dataset{i}.pop.beta = pop.beta;
    for k=1:numel(pop.x)
        dataset{i}.pop.x{k}.info = pop.x{k}.info;
    end
    dataset{i}.pop.ep.info = pop.ep.info;
    dataset{i}.pop.y.info = pop.y.info;
    % number of dataset from population
    dataset{i}.ndataset = n;%500
    % random samples from population
    replacement = true;
    % Initial rho count
    %rho09_3=0;
    rho09_4=0;
%     rho09_5=0;%rho09_6=0;
    lCount=0;mCount=0;gCount=0;count=0;
    rhoInfo = 0.9999;
    for k=1:numel(rhoInfo)
        dataset{i}.rho{k}.rho = rhoInfo(k);
    end
    % Inital timer
    tSamp = tic;
```

```
    while(1)
        count = count+1;
        curSel = randsample(1:pop.N,dataset{i}.n,replacement)';
        curRho = corr(pop.x{1}.data(curSel),pop.x{3}.data(curSel));
        % Save position of random sample that satisfy condition
        if curRho>0.99994
            mCount = mCount+1;
        elseif curRho<0.99985
            lCount = lCount+1;
        else
%           if curRho>=0.9999985 && curRho<=0.9999994
%               if rho09_6<dataset{i}.ndataset
%                   rho09_6 = rho09_6+1;
%                   dataset{i}.rho{4}.curSel(:,rho09_6) = curSel;
%                   dataset{i}.rho{4}.curRho = curRho;
%               end
%           if curRho>=0.999985 && curRho<=0.999994
%               if rho09_5<dataset{i}.ndataset
%                   rho09_5 = rho09_5+1;
%                   dataset{i}.rho{1}.curSel(:,rho09_5) = curSel;
%                   dataset{i}.rho{1}.curRho = curRho;
%               end
            if curRho>=0.99985 && curRho<=0.99994
                if rho09_4<dataset{i}.ndataset
                    rho09_4 = rho09_4+1;
                    dataset{i}.rho{1}.curSel(:,rho09_4) = curSel;
                    dataset{i}.rho{1}.curRho = curRho;
                end
%           elseif curRho>=0.9985 && curRho<=0.9994
%               if rho09_3<dataset{i}.ndataset
```

```
%                 rho09_3 = rho09_3+1;
%                 dataset{i}.rho{1}.curSel(:,rho09_3) = curSel;
%                 dataset{i}.rho{1}.curRho = curRho;
%              end
           else
              gCount = gCount+1;
           end
        end
        % Show rho count
        clc
        Etime = toc(tSamp);
        fprintf('[%d samples dataset][Pop Rho=%.6f]\n[i=%d][%.2f sec.][%.2f Hz]\n',...
           dataset{i}.n,pop.rhoMat.data(2,4),count,Etime,count/Etime);
        fprintf('Current rho = %f\n',curRho);
%        fprintf('Rho0999 = %3d\n',rho09_3);
        fprintf('Rho09999 = %3d\n',rho09_4);
%        fprintf('Rho099999 = %3d\n',rho09_5);
%        fprintf('Rho0999999 = %3d\n',rho09_6);
        fprintf('Rho >0.99994 = %d\n',mCount);
        fprintf('Rho in gap = %d\n',gCount);
        fprintf('Rho <0.99985 = %d\n',lCount);
        % Compute est. iteration
%        minP = min([rho09_3,rho09_4,rho09_5]/count);
        minP = min([rho09_4]/count);
        minP = round(minP*10000)/10000;
        estCount = dataset{i}.ndataset/minP;
        elapsed = floor((estCount-count)/(count/Etime));
        fprintf('Estimated iteration is %.0f (%d min %d sec)\n',...
           estCount,floor(elapsed/60),mod(elapsed,60));
        if rho09_4==dataset{i}.ndataset
```

```
            break;
        end
    end
    % Construct rho condition data
    for k=1:numel(rhoInfo)
        for m=1:numel(pop.x)
            dataset{i}.rho{k}.x{m} = ...
                pop.x{m}.data(dataset{i}.rho{k}.curSel);
        end
        dataset{i}.rho{k}.ep = pop.ep.data(dataset{i}.rho{k}.curSel);
        dataset{i}.rho{k}.y = pop.y.data(dataset{i}.rho{k}.curSel);
    end
    dataset{i}.gentime = Etime;
    dataset{i}.gencount = count;
end
end
```

โปรแกรมที่ใช้ในการประมาณค่าพารามิเตอร์โดยวิธีกำลังสองน้อยที่สุดและวิธีการถดถอยแบบริดจ์

```
% Find OLS regression parameters
% Usage:
% [bOLS yOLS MSEOLS] = computeOLS(X,Y)
% where X = n x p matrix
%       Y = n x 1 matrix
%       bOLS.b = beta OLS (OLS coefficient)
%       bOLS.se = SE of beta OLS
%       yOLS = X*bOLS
%       MSEOLS = MSE of yOLS
%
% See also: computeRidge
function [bOLS yOLS MSEOLS] = computeOLS(X,Y)
```

```
% X = [x0 .. xp]; % n*p where p = k+1;
% Y = y; % n*1
[n p] = size(X);
% k is number of independent variables
k=p-1;
%% Compute Beta OLS
bOLS.b = (X'*X)\(X'*Y);
%% Compute y & MSE of OLS
yOLS = X*bOLS.b;
MSEOLS = ((Y-yOLS)'*(Y-yOLS))/(n-k-1);
%% Compute SE of bOLS
bOLS.se = sqrt(MSEOLS*diag(inv(X'*X)));
end
% Find ridge regression parameters
% Usage:
% [K bRidge yRidge] = computeRidge(X,Y,flag,b)
% where X = n x p matrix
%       Y = n x 1 matrix
%       K = K value
%       bRidge consists of
%           bRidge.b
%           bRidge.var
%           bRidge.se
%           bRidge.MSE
%       yRidge consists of
%           yRidge.y
%           yRidge.SSE
%           yRidge.MSE
%       b = Original beta for compute beta ridge MSE (*can be ignore)
%
```

```
% See also: computeOLS
function [K bRidge yRidge] = computeRidge(X,Y,flag,b)
%% Compute OLS
[bOLS , ~, MSEOLS] = computeOLS(X,Y);
%% Variance of yOLS = MSEOLS
Yvar = MSEOLS;
%% Compute Ridge
[n p] = size(X);
% k is number of independent variables
k=p-1;
%% Original Beta for compute Beta ridge MSE
if nargin<4
    b = zeros(p,1);
else
    b = b(1:p);
end
%%
I = eye(p);
[V,D] = eig(X'*X,'nobalance');
% [V,D] = eig(X'*X);
[lamda ix] = sort(sum(D),'descend');
if strcmp (flag,'K**')
    tm = max(diag(D));
    S_2 = (Y-(X*bOLS.b))'*(Y-(X*bOLS.b))/(n-p);
    alpha_max = max(D\V'*X'*Y);
    K = (tm*S_2)/((tm*alpha_max^2) + (n-p)*S_2);
    bRidge.b = (X'*X + K*I)\(X'*Y);
else % Case rHKB rHMO rLW
    alpha = D\V'*X'*Y;
    Z = (X*V);
```

```
    sigma_2 = (Y'*Y - alpha'*Z'*Y)/(n-p);
    switch flag
        case 'rHKB'
            K = (p*sigma_2)/(alpha'*alpha);
        case 'rLW'
            K = (p*sigma_2)/sum(D*alpha.^2);
        case 'rHMO'
            d = diag(D);% eigenvalue
            temp = 0;
            for i=1:p
                temp = temp+(alpha(i)^2/1+(1+d(i)*(alpha(i)^2/sigma_2)^0.5));
            end
            K = (p*sigma_2)/temp;
    end
    bRidge.b = (X'*X + K*I)\(X'*Y);
%     bRidge.alpha = alpha;
end
% MSE & SSE of Y Ridge
yRidge.SSE = (Y-yRidge.y)'*(Y-yRidge.y);
yRidge.MSE = yRidge.SSE/(n-k-1);
% Variance, SE & MSE of Beta Ridge
bRidge.var = diag((X'*X + K*I)\X'*Yvar*X/(X'*X + K*I));
bRidge.se = sqrt(bRidge.var);
bRidge.MSE = Yvar*sum(lamda./(lamda+K).^2)+...
    ((K^2)*b'*((inv(X'*X+K*I))^2)*b);
end
function dataset = computeOLSnRidge(dataset)
%Compute OLS & Ridge
Kinfo = {'rHKB','rLW','rHMO','K**','Kbay','Kk'};
nSampleSet = numel(dataset);
```

```
for i=1:nSampleSet
    ndataset = dataset{i}.ndataset;%500
    bPop = dataset{i}.pop.beta;
    for k=1:numel(dataset{i}.rho)
        data = dataset{i}.rho{k};
        data.Kinfo = Kinfo;
        for j=1:ndataset
            y = data.y(:,j);
            x = ones(dataset{i}.n,numel(data.x)+1);
            for m = 1:numel(data.x)
                x(:,m+1) = data.x{m}(:,j);
            end
%             [n p] = size(x);
            % OLS
            [bOLS yOLS MSEOLS] = computeOLS(x,y);
            data.ols.bOLS(:,j) = bOLS.b;
            data.ols.bOLSSE(:,j) = bOLS.se;
            data.ols.MSEOLS(:,j) = MSEOLS;
            % Ridge
            for m=1:numel(Kinfo)
                [K bRidge yRidge] = computeRidge(x,y,Kinfo{m},bPop);
                data.ridge{m}.info = Kinfo{m};
                data.ridge{m}.K(:,j) = K;
                data.ridge{m}.bRidge(:,j) = bRidge.b;
                data.ridge{m}.bRidgeVar(:,j) = bRidge.var;
                data.ridge{m}.bRidgeSE(:,j) = bRidge.se;
                data.ridge{m}.bRidgeMSE(:,j) = bRidge.MSE;
                data.ridge{m}.yRidge(:,j) = yRidge.y;
                data.ridge{m}.yRidgeSSE(:,j) = yRidge.SSE;
                data.ridge{m}.yRidgeMSE(:,j) = yRidge.MSE;
```

```
%             %%
%             data.ridge{m}.V{j} = bRidge.V;
%             data.ridge{m}.D{j} = bRidge.D;
%             data.ridge{m}.alpha(:,j) = bRidge.alpha;
%             data.ridge{m}.W(:,j) = bRidge.W;
%             %%
          end
      end
      for m=1:numel(Kinfo)
        data.ridge{m}.bRidgeSEMean =  mean(data.ridge{m}.bRidgeSE,2);
      end
      data.ols.meanbOLSSE = mean(data.ols.bOLSSE,2);
      dataset{i}.rho{k}=data;
  end
end
end
```

โปรแกรมที่ใช้ในการการคัดเลือกตัวแปรอิสระโดยวิธีการถดถอยแบบขั้นตอน(Stepwise Regression)

```
function dataset = OLSnRidgeVarSel(dataset)
% Variable Selection using stepwise (OLS & Ridge)
nSampleSet = numel(dataset);
Kinfo = {'rHKB','rLW','rHMO','K**'};
% Stepwise for OLS & Ridge estimation
alpha = 0.05;
alphaIn = alpha; alphaOut = alpha;
for i=1:nSampleSet
  ndataset = dataset{i}.ndataset;%500
  for k=1:numel(dataset{i}.rho)
      data = dataset{i}.rho{k};
```

```
% Pre-defined variable selection structure
data.ols.varSel = cell(ndataset,3);
for m=1:numel(Kinfo)
   data.ridge{m}.varSel = cell(ndataset,3);
end
for j=1:ndataset
   clc;
   fprintf('Apply stepwise algorithm....\n');
   fprintf('Sample size of dataset = %d (rho=%.5f)\nProcessing %d of %d\n',...
      dataset{i}.n,data.rho,j,ndataset);
   y = data.y(:,j);
   x = ones(dataset{i}.n,numel(data.x)+1);
   for m = 1:numel(data.x)
     x(:,m+1) = data.x{m}(:,j);
   end
   % Define parameters for stepwise regression function
   [n p] = size(x);
   q = p-1;
   % OLS Estimator
   % Generate partial sum table
   parsum = partialSUM(x,y,'OLS');
   % Stepwise regression function
   [xremain xstore] = stepwiseReg(parsum,n,q,alphaIn,alphaOut);
   data.ols.varSel{j,1} = xstore;
   data.ols.varSel{j,2} = xremain;
   % Classify the result
   if sum(ismember(xstore,{'1','2','3','4','5'}))==5
      if numel(xstore)==5
         data.ols.varSel{j,3} = 1;% Correct
      else
```

```
        data.ols.varSel{j,3} = 3;% Over spec.
    end
else
    if sum(ismember(xstore,{'6','7'}))>0
        data.ols.varSel{j,3} = 4;% Miss spec.
    else
        data.ols.varSel{j,3} = 2;% Under spec.
    end
end
% Ridge Estimator
for m=1:numel(Kinfo)
    % Generate partial sum table
    parsum = partialSUM(x,y,Kinfo{m});
    % Stepwise regression function
    [xremain xstore] = stepwiseReg(parsum,n,q,alphaIn,alphaOut);
    data.ridge{m}.varSel{j,1} = xstore;
    data.ridge{m}.varSel{j,2} = xremain;
    % Classify the result
    if sum(ismember(xstore,{'1','2','3','4','5'}))==5
        if numel(xstore)==5
            data.ridge{m}.varSel{j,3} = 1;% Correct
        else
            data.ridge{m}.varSel{j,3} = 3;% Over spec.
        end
    else
        if sum(ismember(xstore,{'6','7'}))>0
            data.ridge{m}.varSel{j,3} = 4;% Miss spec.
        else
            data.ridge{m}.varSel{j,3} = 2;% Under spec.
        end
```

```
            end
        end
    end
    dataset{i}.rho{k}=data;
  end
end
% Compute yHat yMSE ySSE bSSE of selected X (xstore)
clear x y
for i=1:nSampleSet
  ndataset = dataset{i}.ndataset;%500
  bPop = dataset{i}.pop.beta;
  for k=1:numel(dataset{i}.rho)
    data = dataset{i}.rho{k};
    for j=1:ndataset
      y = data.y(:,j);
      % Choose X that pass variable selection test only
      xIn = sort(cellfun(@str2double,data.ols.varSel{j,1}));
      xdata = data.x(xIn);
      x = ones(dataset{i}.n,numel(xdata)+1);
      for m = 1:numel(xdata)
        x(:,m+1) = xdata{m}(:,j);
      end
      [n p] = size(x);
      % Number of selected variables
      h = p-1;
      % OLS
      [bOLS yOLS MSEOLS] = computeOLS(x,y);
      data.ols.nbOLS{:,j} = bOLS.b;
      data.ols.nbOLSSE{:,j} = bOLS.se;
      data.ols.nMSEOLS(:,j) = MSEOLS;
```

```
        % Ridge
        for m=1:numel(Kinfo)
          [K bRidge yRidge] = computeRidge(x,y,Kinfo{m},bPop([1,xIn+1]));
          data.ridge{m}.nK{:,j} = K;
          data.ridge{m}.nyRidge(:,j) = yRidge.y;
          data.ridge{m}.nyRidgeSSE(:,j) = yRidge.SSE;
          data.ridge{m}.nyRidgeMSE(:,j) = yRidge.MSE;
          data.ridge{m}.nbRidge{:,j} = bRidge.b;
          data.ridge{m}.nbRidgeVar{:,j} = bRidge.var;
          data.ridge{m}.nbRidgeSE{:,j} = bRidge.se;
          data.ridge{m}.nbRidgeMSE(:,j) = bRidge.MSE;
        end
      end
      dataset{i}.rho{k}=data;
  end
end
end
```

โดยที่เรียกใช้ฟังก์ชันย่อย ดังนี้

```
function [xall xstore] = stepwiseReg(parsum,n,k,alphaIn,alphaOut)
debug = 0;
%%
xall = cell(1,nchoosek(k,1));
for i=1:nchoosek(k,1)
  xall{i} = num2str(i,'%d');
end
result = 1;
xstore = {};
[xall xstore result] = forwardSW(parsum,xall,xstore,alphaIn,alphaOut,n,result);
if debug
  fprintf('Significant Variables : ');
```

```
    fprintf('X%c ',cell2mat(xstore));fprintf('\n');
    fprintf('Unsignificant Variables : ');
    fprintf('X%c ',cell2mat(xall));fprintf('\n');
end
end
function [xall xstore result] = forwardSW(parsum,xall,xstore,alphaIn,alphaOut,n,result)
    debug = 0;
    if result==0,return;end
    if isempty(xall),return;end
    % Using max SSR as criteria
    [val idx] = maxSSR(parsum,xall,xstore);
    %
    ssr = partialSSR(parsum,idx,xstore);
    mse = partialMSE(parsum,[idx,xstore]);
    % Test hypothesis
    ktest = numel([idx,xstore]);
    F = ssr/mse;
    Ftest = finv(1-alphaIn,numel(idx),n-ktest-1);
    if F < Ftest % Accept H0
        result = 0;
        if debug
            fprintf('Terminate at : ');
            fprintf('X%c ',cell2mat(idx));fprintf('\n');
        end
    else % Reject H0
        result = 1;
        if debug
            fprintf('In: X%c | SSR = %.4f\n',cell2mat(idx),val);
        end
        xpick = idx;
```

```
        % Check backward
        [xall xstore] = backwardSW(parsum,xpick,xall,xstore,alphaOut,n);
        % Add significant variable
        xstore = [xstore,idx];
        % Remove added variable from variable pool
        xall(ismember(xall,idx))='';
        % Recursive
        [xall xstore result] = forwardSW(parsum,xall,xstore,alphaIn,alphaOut,n,result);
    end
end
function [xall xstore] = backwardSW(parsum,xpick,xall,xstore,alphaOut,n)
    debug = 0;
    xstoreTemp = xstore;
    for i = 1:numel(xstore)
        xcur = xstore(i);
        xTemp = xstore;
        xTemp(ismember(xTemp,xcur))='';
        xgiven = sort([xTemp,xpick]);
        ssr = partialSSR(parsum,xcur,xgiven);
        mse = partialMSE(parsum,[xcur,xgiven]);
            % Test hypothesis
        ktest = numel([xcur,xgiven]);
        F = ssr/mse;
        Ftest = finv(1-alphaOut,numel(xcur),n-ktest-1);
        if F < Ftest % Accept H0
            if debug
                fprintf('Out: X%c\n',cell2mat(xcur));
            end
            xstoreTemp(ismember(xstoreTemp,xcur))='';
            xall = sort([xall,xcur]);
```

```
        else % Reject H0
        end
    end
    xstore = xstoreTemp;
end
function mse = partialMSE(parsum,xall)
    if numel(xall)==0
        mse = NaN; return;
    else
        xall = sort(xall);
        i = numel(xall);
        xall = cell2mat(xall);
        mse = parsum{i}{ismember(parsum{i}(:,1),xall),4};
    end
end
function [xMax xMaxIdx] = maxSSR(parsum,xtest,xgiven)
% Find Xn which has maximum SSR with given Xm
% Usage:
% [xMax xMaxIdx] = maxSSR(parsum,xtest,xgiven)
% where parsum is partial sum object obtained from partailSUM function
%       xtest is cell string that contains number of X that want to test
%       xgiven is cell string that contains number of X that was given
%
% Ex.1 Find maximum SSR of SSR(X1|X3), SSR(X2|X3), SSR(X4|X3)
% [xMax xMaxIdx] = maxSSR(parsum,{'1','2','4'},{'3'});
%
% Ex.2 Find maximum SSR of SSR(X2|X1X3), SSR(X4|X1X3)
% [xMax xMaxIdx] = maxSSR(parsum,{'2','4'},{'1','3'});
%
% Ex.3 Find maximum SSR of SSR(X1), SSR(X2), SSR(X3), SSR(X4)
```

```
% [xMax xMaxIdx] = maxSSR(parsum,{'1','2','3','4'});
%
% See also: partialSUM | partialSSR
if nargin<3
  xgiven = {};
end
ssrCheck = zeros(numel(xtest),1);
for i=1:numel(xtest)
   ssrCheck(i) = partialSSR(parsum,xtest(i),xgiven);
end
[val idx] = max(ssrCheck);
xMaxIdx = xtest(idx);
xMax = val;
end
% Find partial SSR
% Usage:
% ssr = partialSSR(parsum,xcurrent,xgiven)
%
% Ex. Find SSR(X1X2|X3X4)
% ssr_x1x2_x3x4 = partialSSR(parsum,{'1','2'},{'3','4'})
%
% See also: partialSUM | maxSSR
function ssr = partialSSR(parsum,xcurrent,xgiven)
   if numel(xcurrent)==0
      ssr = NaN; return;
   end
   if nargin<3 || numel(xgiven)==0
      xcurrent = cell2mat(xcurrent);
      j = numel(xcurrent);
      ssr = parsum{j}{ismember(parsum{j}(:,1),xcurrent),2};
```

```
    else
        xgiven = sort(xgiven);
        xall = sort([xcurrent,xgiven]);
        i = numel(xall);
        j = numel(xgiven);
        xall = cell2mat(xall);
        xgiven = cell2mat(xgiven);
        sseAll = parsum{i}{ismember(parsum{i}(:,1),xall),3};
        sseGiven = parsum{j}{ismember(parsum{j}(:,1),xgiven),3};
        ssr = sseGiven - sseAll;
%           ssrAll = parsum{i}{ismember(parsum{i}(:,1),xall),2};
%           ssrGiven = parsum{j}{ismember(parsum{j}(:,1),xgiven),2};
%           ssr = ssrAll-ssrGiven;
    end
end
% Find partial sum
% Usage:
% parsum = partialSUM(x,y,bModel);
% where x is matrix with n x p dimensions
%       y is matrix with n x 1 dimensions
%       bModel is coefficient model (default is OLS)
%
% parsum is an cell array with 1 x n dimensions
% parsum{i} is an cell array with nchoosek(n,i) x 6 dimensions
% parsum{i}{:,1} is String index
% parsum{i}{:,2} is SSR corresponding to string index
% parsum{i}{:,3} is SSE corresponding to string index
% parsum{i}{:,4} is MSE corresponding to string index
% parsum{i}{:,5} is R^2 corresponding to string index
% parsum{i}{:,6} is R^2_a corresponding to string index
```

```
%
% Example find ssr & sse in parsum
% Ex.1 Find partial sum of x1 x2 x4
% ssr_x1x2x4 = parsum{3}{ismember(parsum{3}(:,1),'124'),2}
% sse_x1x2x4 = parsum{3}{ismember(parsum{3}(:,1),'124'),3}
% mse_x1x2x4 = parsum{3}{ismember(parsum{3}(:,1),'124'),4}
%
% See also: partialSSR | maxSSR
function parsum = partialSUM(x,y,bModel)
if nargin<3
   bModel = 'OLS';
end
debug = 0;
[n p] = size(x);
k = p-1;
parsum = cell(1,k);
for i = 1:k
   count = 1;
   curSel = nchoosek(1:k,i);
   [r c] = size(curSel);
   parsum{i} = cell(r,6);
   for j = 1:r
      parsum{i}{count,1} = num2str(curSel(j,:),'%d');
      xpick = x(:,[1 curSel(j,:)+1]);
      ktest = numel(curSel(j,:));
      if det(xpick'*xpick)~=0
         switch bModel
            case 'OLS'
               [~ , yOLS, ~] = computeOLS(xpick,y);
               yHat = yOLS;
```

```
        case 'K**'
            [~ , ~, yRidge] = computeRidge(xpick,y,'K**');
            yHat = yRidge.y;
        case 'rHKB'
            [~ , ~, yRidge] = computeRidge(xpick,y,'rHKB');
            yHat = yRidge.y;
        case 'rLW'
            [~ , ~, yRidge] = computeRidge(xpick,y,'rLW');
            yHat = yRidge.y;
        case 'rHMO'
            [~ , ~, yRidge] = computeRidge(xpick,y,'rHMO');
            yHat = yRidge.y;
        otherwise
            disp('Invalid bModel');
            parsum = NaN;
            return;
    end
    ssr = (yHat'*y - (sum(y)^2)/n);
    sse = y'*y - yHat'*y;
    mse = sse/(n-ktest-1);
    Rsquare = ssr/(ssr+sse);
    Rsquare_a = 1-(mse/var(y));
else
    ssr = NaN;
    sse = NaN;
    mse = NaN;
    Rsquare = NaN;
    Rsquare_a = NaN;
end
parsum{i}{count,2} = ssr;
```

```
      parsum{i}{count,3} = sse;
      parsum{i}{count,4} = mse;
      parsum{i}{count,5} = Rsquare;
      parsum{i}{count,6} = Rsquare_a;
      if debug
        fprintf('Case');
        fprintf(' X%d',curSel(j,:));
        fprintf('\nSSR = %.4f\n',parsum{i}{count,2});
        fprintf('SSE = %.4f\n',parsum{i}{count,3});
        fprintf('MSE = %.4f\n',parsum{i}{count,4});
        fprintf('R^2 = %.4f\n',parsum{i}{count,5});
        fprintf('R^2_a = %.4f\n',parsum{i}{count,6});
      end
      count = count+1;
    end
end
end
```

โปรแกรมที่ใช้ในการประมาณค่าพารามิเตอร์โดยวิธีการค้นหาแบบต้องห้าม (Tabu Search)

```
function dataset = computeTabu(dataset,u,l)
%COMPUTETABU Summary of this function goes here
%   Detailed explanation goes here
nSampleSet = numel(dataset);
for i=1:nSampleSet
    ndataset = dataset{i}.ndataset;%500
    nSample = dataset{i}.n;
    for k=1:numel(dataset{i}.rho)
        data = dataset{i}.rho{k};
        for j=1:ndataset
            clc;
```

```
            fprintf('Apply tabu search algorithm....\n');
            fprintf('Sample size of dataset = %d (rho=%.5f)\nProcessing %d of %d\n',...
              nSample,data.rho,j,ndataset);
            y = data.y(:,j);
            x = ones(nSample,numel(data.x)+1);
            for m = 1:numel(data.x)
              x(:,m+1) = data.x{m}(:,j);
            end
            % Tabu MSE
            % Define objective function
            f = @(b)objFcn(b,x,y);
            % Apply Direct Tabu Search
            [bTabu MSE Count] = DTSps(f,u,l);
            bTabu = bTabu';
            data.tabu.bTabu(:,j) = bTabu;
            data.tabu.MSE(:,j) = MSE;
            data.tabu.Count(:,j) = Count;
            data.tabu.UpperB = u;
            data.tabu.LowerB = l;
        end
        data.tabu.bTabuSD = std(data.tabu.bTabu,1,2);
        dataset{i}.rho{k}=data;
    end
end
end
function dataset = ncomputeTabu(dataset,u,l)
%COMPUTETABU Summary of this function goes here
%   Detailed explanation goes here
nSampleSet = numel(dataset);
for i=1:nSampleSet
```

```
ndataset = dataset{i}.ndataset;%500
nSample = dataset{i}.n;
for k=1:numel(dataset{i}.rho)
    data = dataset{i}.rho{k};
    for j=1:ndataset
        clc;
        fprintf('Apply new tabu search algorithm....\n');
        fprintf('Sample size of dataset = %d (rho=%.5f)\nProcessing %d of %d\n',...
            nSample,data.rho,j,ndataset);
        y = data.y(:,j);
        x = ones(nSample,numel(data.x)+1);
        for m = 1:numel(data.x)
          x(:,m+1) = data.x{m}(:,j);
        end
        % Tabu MSE mod
        nf = @(b)objFcn_mod(b,x,y);
        % Apply Direct Tabu Search
        nu = [u,1];nl = [l,0];
        [bTabu MSE Count] = DTSps(nf,nu,nl);
        bTabu = bTabu';
        data.ntabu.bTabu(:,j) = bTabu(1:end-1);
        data.ntabu.c(:,j) = bTabu(end);
        data.ntabu.MSE(:,j) = MSE;
        data.ntabu.Count(:,j) = Count;
        data.ntabu.UpperB = u;
        data.ntabu.LowerB = l;
    end
    data.ntabu.bTabuSD = std(data.ntabu.bTabu,1,2);
    dataset{i}.rho{k}=data;
end
```

```
end
end
```

โปรแกรมที่ใช้ในการคัดเลือกตัวแปรอิสระโดยวิธีการค้นหาแบบต้องห้าม (Tabu Search)

```
function dataset = TabuVarSel(dataset,u,l)
nSampleSet = numel(dataset);
alpha = 0.05;
for i=1:nSampleSet
  ndataset = dataset{i}.ndataset;%500
  nSample = dataset{i}.n;
  for k=1:numel(dataset{i}.rho)
    data = dataset{i}.rho{k};
    data.nSample = nSample;
    % Compute Tabu for bTabuSD
    if isfield(data,'tabu')==0
      for j=1:ndataset
        clc;fprintf('Apply tabu search algorithm....\n');
        fprintf('Sample size of dataset = %d (rho=%.5f)\nProcessing %d of %d\n',...
          nSample,data.rho,j,ndataset);
        y = data.y(:,j);
        x = ones(nSample,numel(data.x)+1);
        for m = 1:numel(data.x)
          x(:,m+1) = data.x{m}(:,j);
        end
        f = @(b)objFcn(b,x,y);
        [bTabu MSE Count] = DTSps(f,u,l);
        bTabu = bTabu';
        data.tabu.bTabu(:,j) = bTabu;
        data.tabu.MSE(:,j) = MSE;
        data.tabu.Count(:,j) = Count;
```

```
        data.tabu.UpperB = u;
        data.tabu.LowerB = l;
    end
    data.tabu.bTabuSD = std(data.tabu.bTabu,1,2);
end
% t-test Check
data.tabu.alpha = alpha;
data.tabu.varSel = cell(ndataset,3);
for j=1:ndataset
    clc;fprintf('Apply tabu search algorithm (Variable Selection)....\n');
    fprintf('Sample size of dataset = %d (rho=%.5f)\nProcessing %d of %d\n',...
        nSample,data.rho,j,ndataset);
    bTabuSD = data.tabu.bTabuSD;
    bTabu = data.tabu.bTabu(:,j);
    x = ones(dataset{i}.n,numel(data.x)+1);
    for m = 1:numel(data.x)
      x(:,m+1) = data.x{m}(:,j);
    end
    xin = 1:numel(data.x);
    [xin xout] = ntsVarSel(x,xin,bTabu,bTabuSD,data.tabu.alpha);
    tabu.varSel{j,1} = xin;
    tabu.varSel{j,2} = xout;
    if isempty(xout)
        tabu.varSel{j,3} = 0; % do not test futher
    else
        tabu.varSel{j,3} = 1; % continue to test
    end
    tabu.varSel{j,4} = j; % assign index at 1st test
end
% Backward t-test variable selection
```

```
        Gp = {};
        [data,tabu,Gp]=backwardVS(data,tabu,Gp);
        % Classify the result
        for j=1:ndataset
            xin = data.tabu.varSel{j,1};
            if sum(ismember(xin,{'1','2','3','4','5'}))==5
                if numel(xin)==5
                    data.tabu.varSel{j,3} = 1;% Correct
                else
                    data.tabu.varSel{j,3} = 3;% Over spec.
                end
            else
                if sum(ismember(xin,{'6','7'}))>0
                    data.tabu.varSel{j,3} = 4;% Miss spec.
                else
                    data.tabu.varSel{j,3} = 2;% Under spec.
                end
            end
        end
        % save data back to dataset
        dataset{i}.rho{k} = data;
    end
end
%% Comput New MSE
clear x y
for i=1:nSampleSet
    ndataset = dataset{i}.ndataset;%500
    nSample = dataset{i}.n;
    for k=1:numel(dataset{i}.rho)
        data = dataset{i}.rho{k};
```

```
        for j=1:ndataset
            clc;fprintf('Apply tabu search algorithm (Compute new MSE)....\n');
            fprintf('Sample size of dataset = %d (rho=%.5f)\nProcessing %d of %d\n',...
            nSample,data.rho,j,ndataset);
            y = data.y(:,j);
            % Choose X that pass variable selection test only
            xIn = sort(cellfun(@str2double,data.tabu.varSel{j,1}));
            xdata = data.x(xIn);
            ucur = u([1 xIn+1]);lcur = l([1 xIn+1]);
            x = ones(dataset{i}.n,numel(xdata)+1);
            for m = 1:numel(xdata)
               x(:,m+1) = xdata{m}(:,j);
            end
            [n p] = size(x);
            % Number of selected variables
            h = p-1;
            % Tabu
            f = @(b)objFcn(b,x,y);
            [bTabu MSE Count] = DTSps(f,ucur,lcur);
            bTabu = bTabu';
            data.tabu.nbTabu{:,j} = bTabu;
            data.tabu.nMSE(:,j) = MSE;
            data.tabu.nCount(:,j) = Count;
        end
        dataset{i}.rho{k}=data;
    end
end
end
% Recursive backward t-test
function [data,tabu,Gp]=backwardVS(data,tabu,Gp)
```

```
% Update the result to data.tabu.varSel
for j=1:numel(tabu.varSel(:,1))
    if ~tabu.varSel{j,3}
        idx = cell2mat(tabu.varSel(j,4));
        data.tabu.varSel(idx,1:2) = tabu.varSel(j,1:2);
    end
end
% Function terminate check
if sum(cell2mat(tabu.varSel(:,3)))==0,return;end
% Remove done testing case
varSel = tabu.varSel(cell2mat(tabu.varSel(:,3))==1,:);
% Find the unique group
xin = cell(numel(varSel(:,1)),1);
idx = xin;
for j=1:numel(varSel(:,1))
    xin{j} = str2double(varSel{j,1});
    idx{j} = varSel{j,4};
end
xinAll = cell2mat(xin);
varGp = unique(xinAll,'rows');
[nGp nVar] = size(varGp);
Gp = cell(nGp,1);
for m=1:nGp
    Gp{m}.xin = sort(varGp(m,:));
    Gp{m}.idx = [];
    for mn =1:numel(xin)
        if isequal(xin{mn},varGp(m,:))
            Gp{m}.idx = [Gp{m}.idx;idx{mn}];
        end
    end
```

```
  Gp{m}.n = numel(Gp{m}.idx);
  Gp{m}.xout = varSel{cell2mat(varSel(:,4))==Gp{m}.idx(1),2};
end
nSample = data.nSample;
for m=1:nGp
  tabu.bTabu = [];
  % find bTabuSD within group
  for j=1:Gp{m}.n
    idx = Gp{m}.idx(j);
    xin = Gp{m}.xin;
    clc;fprintf('Apply tabu search algorithm....\n');
    fprintf('Sample size of dataset = %d (rho=%.5f)\n',nSample,data.rho);
    fprintf('xin = %s\n',mat2str(xin));
    fprintf('Processing %d of %d\n',j,Gp{m}.n);
    y = data.y(:,idx);
    x = ones(nSample,numel(xin)+1);
    for n = 1:numel(xin)
     x(:,n+1) = data.x{xin(n)}(:,idx);
    end
    ucur = data.tabu.UpperB([1 xin+1]);
    lcur = data.tabu.LowerB([1 xin+1]);
    f = @(b)objFcn(b,x,y);
    [bTabu MSE Count] = DTSps(f,ucur,lcur);
    bTabu = bTabu';
    tabu.bTabu(:,j) = bTabu;
  end
  tabu.bTabuSD = std(tabu.bTabu,1,2);
  % t-test
  for j=1:Gp{m}.n
    idx = Gp{m}.idx(j);
```

```
            xin = Gp{m}.xin;
            bTabuSD = tabu.bTabuSD;
            bTabu = tabu.bTabu(:,j);
            x = ones(nSample,numel(xin)+1);
            for n = 1:numel(xin)
              x(:,n+1) = data.x{xin(n)}(:,idx);
            end
            [xin xout] = ntsVarSel(x,xin,bTabu,bTabuSD,data.tabu.alpha);
            tabu.varSel{idx,1} = xin;
            tabu.varSel{idx,2} = [Gp{m}.xout xout];
            if isempty(xout)
                tabu.varSel{idx,3} = 0; % do not test futher
            else
                tabu.varSel{idx,3} = 1; % continue to test
            end
            tabu.varSel{idx,4} = idx;
        end
    end
    % Call the function again (recursive)
    [data,tabu,Gp]=backwardVS(data,tabu,Gp);
end
% t-test
function [xchk xout] = ntsVarSel(x,xin,b,bSE,alpha)
    [n k] = size(x);
    xout={};xtemp=[];
    betaChk = b(2:end);
    betaSEChk = bSE(2:end);
    for i = 1:k-1
        t0 = betaChk(i)/betaSEChk(i);
        ttest = tinv(1-(alpha/2),n-(k-1)-1);
```

```
        if abs(t0) > ttest % Reject H0: B_h=0
        else % Accept H0: B_h=0
            xtemp = [xtemp i];
        end
    end
    if numel(xtemp)>0% if there exists x that accept H0
        % find b of xout that closest to zero value
        beta = betaChk(xtemp);
        idx = abs(betaChk)==min(abs(beta));
        xout = [xout num2str(xin(idx),'%d')];
        xin(idx)=[];
    end
    %return xin
    xchk=cell(1,numel(xin));
    for i=1:numel(xin)
        xchk{i} = num2str(xin(i),'%d');
    end
end
function dataset = nTabuVarSel(dataset,u,l)
nSampleSet = numel(dataset);
alpha = 0.05;
for i=1:nSampleSet
    ndataset = dataset{i}.ndataset;%500
    nSample = dataset{i}.n;
    for k=1:numel(dataset{i}.rho)
        data = dataset{i}.rho{k};
        data.nSample = nSample;
        % Compute Tabu for bTabuSD
        if isfield(data,'ntabu')==0
            for j=1:ndataset
```

```
        clc;fprintf('Apply new tabu search algorithm....\n');
        fprintf('Sample size of dataset = %d (rho=%.5f)\nProcessing %d of %d\n',...
          nSample,data.rho,j,ndataset);
        y = data.y(:,j);
        x = ones(nSample,numel(data.x)+1);
        for m = 1:numel(data.x)
         x(:,m+1) = data.x{m}(:,j);
        end
        nf = @(b)objFcn_mod(b,x,y);
        nu = [u,1];nl = [l,0];
        [bTabu MSE Count] = DTSps(nf,nu,nl);
        bTabu = bTabu';
        data.ntabu.bTabu(:,j) = bTabu(1:end-1);
        data.ntabu.c(:,j) = bTabu(end);
        data.ntabu.MSE(:,j) = MSE;
        data.ntabu.Count(:,j) = Count;
        data.ntabu.UpperB = u;
        data.ntabu.LowerB = l;
    end
    data.ntabu.bTabuSD = std(data.ntabu.bTabu,1,2);
end
% t-test Check
data.ntabu.alpha = alpha;
data.ntabu.varSel = cell(ndataset,3);
for j=1:ndataset
    clc;fprintf('Apply new tabu search algorithm (Variable Selection)....\n');
    fprintf('Sample size of dataset = %d (rho=%.5f)\nProcessing %d of %d\n',...
        nSample,data.rho,j,ndataset);
    bTabuSD = data.ntabu.bTabuSD;
    bTabu = data.ntabu.bTabu(:,j);
```

```
    x = ones(dataset{i}.n,numel(data.x)+1);
    for m = 1:numel(data.x)
      x(:,m+1) = data.x{m}(:,j);
    end
    xin = 1:numel(data.x);
    [xin xout] = ntsVarSel(x,xin,bTabu,bTabuSD,data.ntabu.alpha);
    ntabu.varSel{j,1} = xin;
    ntabu.varSel{j,2} = xout;
    if isempty(xout)
        ntabu.varSel{j,3} = 0; % do not test futher
    else
        ntabu.varSel{j,3} = 1; % continue to test
    end
    ntabu.varSel{j,4} = j; % assign index at 1st test
end
% Backward t-test variable selection
Gp = {};
[data,ntabu,Gp]=backwardVS(data,ntabu,Gp);
% Classify the result
for j=1:ndataset
    xin = data.ntabu.varSel{j,1};
    if sum(ismember(xin,{'1','2','3','4','5'}))==5
        if numel(xin)==5
            data.ntabu.varSel{j,3} = 1;% Correct
        else
            data.ntabu.varSel{j,3} = 3;% Over spec.
        end
    else
        if sum(ismember(xin,{'6','7'}))>0
            data.ntabu.varSel{j,3} = 4;% Miss spec.
```

```
            else
                data.ntabu.varSel{j,3} = 2;% Under spec.
            end
        end
    end
    % save data back to dataset
    dataset{i}.rho{k} = data;
  end
end
%% Comput New MSE
clear x y
for i=1:nSampleSet
  ndataset = dataset{i}.ndataset;%500
  nSample = dataset{i}.n;
  for k=1:numel(dataset{i}.rho)
    data = dataset{i}.rho{k};
    for j=1:ndataset
      clc;fprintf('Apply new tabu search algorithm (Compute new MSE)....\n');
      fprintf('Sample size of dataset = %d (rho=%.5f)\nProcessing %d of %d\n',...
      nSample,data.rho,j,ndataset);
      y = data.y(:,j);
      % Choose X that pass variable selection test only
      xIn = sort(cellfun(@str2double,data.ntabu.varSel{j,1}));
      xdata = data.x(xIn);
      ucur = [u([1 xIn+1]) 1];lcur = [l([1 xIn+1]) 0];
      x = ones(dataset{i}.n,numel(xdata)+1);
      for m = 1:numel(xdata)
        x(:,m+1) = xdata{m}(:,j);
      end
      [n p] = size(x);
```

```
        % Number of selected variables
        h = p-1;
        % Tabu
        f = @(b)objFcn_mod(b,x,y);
        [bTabu MSE Count] = DTSps(f,ucur,lcur);
        bTabu = bTabu';
        data.ntabu.nbTabu{:,j} = bTabu(1:end-1);
        data.ntabu.nc(:,j) = bTabu(end);
        data.ntabu.nMSE(:,j) = MSE;
        data.ntabu.nCount(:,j) = Count;
      end
      dataset{i}.rho{k}=data;
    end
end
end
%% Recursive backward t-test
function [data,tabu,Gp]=backwardVS(data,tabu,Gp)
    % Update the result to ndata.tabu.varSel
    for j=1:numel(tabu.varSel(:,1))
      if ~tabu.varSel{j,3}
        idx = cell2mat(tabu.varSel(j,4));
        data.ntabu.varSel(idx,1:2) = tabu.varSel(j,1:2);
      end
    end
    % Function terminate check
    if sum(cell2mat(tabu.varSel(:,3)))==0,return;end
    % Remove done testing case
    varSel = tabu.varSel(cell2mat(tabu.varSel(:,3))==1,:);
    % Find the unique group
    xin = cell(numel(varSel(:,1)),1);
```

```
idx = xin;
for j=1:numel(varSel(:,1))
   xin{j} = str2double(varSel{j,1});
   idx{j} = varSel{j,4};
end
xinAll = cell2mat(xin);
varGp = unique(xinAll,'rows');
[nGp nVar] = size(varGp);
Gp = cell(nGp,1);
for m=1:nGp
   Gp{m}.xin = sort(varGp(m,:));
   Gp{m}.idx = [];
   for mn =1:numel(xin)
      if isequal(xin{mn},varGp(m,:))
         Gp{m}.idx = [Gp{m}.idx;idx{mn}];
      end
   end
   Gp{m}.n = numel(Gp{m}.idx);
   Gp{m}.xout = varSel{cell2mat(varSel(:,4))==Gp{m}.idx(1),2};
end
nSample = data.nSample;
for m=1:nGp
   tabu.bTabu = [];
   % find bTabuSD within group
   for j=1:Gp{m}.n
      idx = Gp{m}.idx(j);
      xin = Gp{m}.xin;
      clc;fprintf('Apply new tabu search algorithm....\n');
      fprintf('Sample size of dataset = %d (rho=%.5f)\n',nSample,data.rho);
      fprintf('xin = %s\n',mat2str(xin));
```

```
    fprintf('Processing %d of %d\n',j,Gp{m}.n);
    y = data.y(:,idx);
    x = ones(nSample,numel(xin)+1);
    for n = 1:numel(xin)
      x(:,n+1) = data.x{xin(n)}(:,idx);
    end
    ucur = [data.ntabu.UpperB([1 xin+1]) 1];
    lcur = [data.ntabu.LowerB([1 xin+1]) 0];
    f = @(b)objFcn_mod(b,x,y);
    [bTabu MSE Count] = DTSps(f,ucur,lcur);
    bTabu = bTabu';
    tabu.bTabu(:,j) = bTabu(1:end-1);
end
tabu.bTabuSD = std(tabu.bTabu,1,2);
% t-test
for j=1:Gp{m}.n
    idx = Gp{m}.idx(j);
    xin = Gp{m}.xin;
    bTabuSD = tabu.bTabuSD;
    bTabu = tabu.bTabu(:,j);
    x = ones(nSample,numel(xin)+1);
    for n = 1:numel(xin)
      x(:,n+1) = data.x{xin(n)}(:,idx);
    end
    [xin xout] = ntsVarSel(x,xin,bTabu,bTabuSD,data.ntabu.alpha);
    tabu.varSel{idx,1} = xin;
    tabu.varSel{idx,2} = [Gp{m}.xout xout];
    if isempty(xout)
        tabu.varSel{idx,3} = 0; % do not test futher
    else
```

```
            tabu.varSel{idx,3} = 1; % continue to test
         end
         tabu.varSel{idx,4} = idx;
      end
   end
   % Call the function again (recursive)
   [data,tabu,Gp]=backwardVS(data,tabu,Gp);
end
% t-test
function [xchk xout] = ntsVarSel(x,xin,b,bSE,alpha)
   [n k] = size(x);
   xout={};xtemp=[];
   betaChk = b(2:end);
   betaSEChk = bSE(2:end);
   for i = 1:k-1
      t0 = betaChk(i)/betaSEChk(i);
      ttest = tinv(1-(alpha/2),n-(k-1)-1);
      if abs(t0) > ttest % Reject H0: B_h=0
      else % Accept H0: B_h=0
         xtemp = [xtemp i];
      end
   end
   if numel(xtemp)>0% if there exists x that accept H0
      % find b of xout that closest to zero value
      beta = betaChk(xtemp);
      idx = abs(betaChk)==min(abs(beta));
      xout = [xout num2str(xin(idx),'%d')];
      xin(idx)=[];
   end
   %return xin
```

```
   xchk=cell(1,numel(xin));
   for i=1:numel(xin)
      xchk{i} = num2str(xin(i),'%d');
   end
end
```

โดยที่เรียกใช้ฟังก์ชัน ดังนี้

```
function
[FMin1,XMin1,FCount1,TLm,fTLm,recm,VRLm,frVRLm]=APS(f,n,x,fx,Edg,TL,fTL,rec,VRL,f
rVRL)
%
%%%%%%%%%%%%%%%%%% TS Memory Parameters %%%%%%%%%%%%%%%%%%
%
dTR=0.01*Edg;                 % TR redius
dSTR=2*dTR;                   % Semi-TR outer redius
nTL=5*n;                      % No. of tabus in TL
inTL=length(fTL);             % Index for No. of tabus in TL
dVR=2*Edg;                    % Region redius in VRL
inVRL=length(frVRL);          % Index for No. of regions in VRL
etamax=1; etamin=1/nTL;       % Max & Min Recency Ranked Values
nBP=nTL/2.5;                  % No. of Best Points saved in TL
miumax=1; miumin=1/nTL;       % Max & Min f-value Ranked Values
MaxIts=3*n;
NoImprov=n;
inNoImprov=0;
% Parameters
FCount1=0;
fbetter=fx;
fv=zeros(n+2,1);
id=eye(n);
vflag=0;
```

```
% **************** Main iteration ******************
% Constructing the Neighborhood Search
itc=0;
while(itc < MaxIts & inNoImprov < NoImprov);
   % Check Hitting the STR
   itc=itc+1;
   S=zeros(n+2,n);
   fbetter=fx;
   STRHitTest=0;
   indd=0;
   for j=1:inTL;
      if norm(x-TL(j,:))<dSTR
         indd=indd+1;
      end
   end
   if indd>0; STRHit=zeros(indd,n); end
   indd=0;
   for j=1:inTL;
      if norm(x-TL(j,:))<dSTR
         indd=indd+1;
         dxSTR(indd)=norm(x-TL(j,:));
         STRHit(indd,:)=TL(j,:);
         STRHitTest=1;
      end
   end
   if STRHitTest == 1
      % Constructing the Neighborhood Search under Hitting STR
      ddmax=max(dxSTR);
      if indd == 1
         Tbar=STRHit;
```

```
    else
       Tbar=sum(STRHit)/indd;
    end
    in=1;
    sgn=sign(x-Tbar);
                   while(fbetter >= fx & in <= n)
       if sgn(in)==0; sgn(in)=(-1)^round(rand); end
       S(in,:)=x+ddmax*sgn(in)*id(in,:);
       for k=1:inTL
          if norm(S(in,:)-TL(k,:))<dTR
             vd=TL(k,:)-x;
             newedg=abs(vd(in))+sqrt(dTR^2+(norm(vd))^2-(vd(in))^2);
             S(in,:)=x+newedg*(1+0.1*rand)*sgn(in)*id(in,:);
          end
       end
       fv(in)=feval(f,S(in,:));
       FCount1=FCount1+1;
       if fv(in)<fbetter
          x1=S(in,:);
          f1=fv(in);
          Bflag=1;
          fbetter=fv(in);
       else
          Bflag=0;
       end
       in=in+1;
                   end
 else
    % Constructing the normal Neighborhood Search
    in=1;
```

```
if vflag==1
   sn=sign(v);
else
   rn=rand(n,1);sn=(-1).^round(rn);
end
               while(fbetter >= fx & in <= n)
   EdgL=Edg+rand*Edg;
   if sn(in)==0; sn(in)=(-1)^round(rand); end
   S(in,:)=x+sn(in)*EdgL*id(in,:);
   Sflag=3;
   for k=1:inTL
      if norm(S(in,:)-TL(k,:))<dTR
         vd=TL(k,:)-x;
         newedg=abs(vd(in))+sqrt(dTR^2+(norm(vd))^2-(vd(in))^2);
         S(in,:)=x+newedg*(1+0.1*rand)*sn(in)*id(in,:);
      end
   end
   for j=1:inTL;
      dx1=norm(S(in,:)-TL(j,:));
      if(dx1 < dSTR)
         EdgL= dx1+(1+0.5*rand)*dSTR;
         S(in,:)=x+EdgL*id(in,:);
      end
   end
   fv(in)=feval(f,S(in,:));
   FCount1=FCount1+1;
   if fv(in)<fbetter
      x1=S(in,:);
      f1=fv(in);
      Bflag=1;
```

```
            fbetter=fv(in);
        else
            Bflag=0;
        end
        in=in+1;
                    end
end
if (Bflag == 0)
    df=zeros(n,1);
    u=zeros(n,n);
    v=zeros(1,n);
    for j=1:n;
        df(j)=fv(j)-fx;
                    end
                    for i=1:n;
        w(i)=df(i)/sum(abs(df));
        u(i,:)=-(S(i,:)-x);
        u(i,:)=u(i,:)/norm(u(i,:));
                    end
    for j=1:n; v=v+w(j)*u(j,:); end
    v=v/norm(v);
    S(n+1,:)=x+(0.5*rand)*Edg*v;
    S(n+2,:)=x+(0.5+rand)*Edg*v;
    for j=1:2
        jn=n+j;
        fv(jn)=feval(f,S(jn,:));
    end
    [fvs,is]=sort(fv);
    Stmp=S(is,:); S=Stmp; fv=fvs;
    xnew=S(1,:);
```

```
    fnew=fv(1);
    vflag=1;
        else
    xnew=x1; fnew=f1;
    vflag=0;
        end
        % TL and VRL Update
        if(inTL < nTL)
    inTL=inTL+1; TL(inTL,:)=x;
    fTL(inTL)=fx; rec(inTL)=itc+1;
else
    [sort1,is1]=sort(rec);
    [sort2,is2]=sort(fTL);
    for j=1:nTL;
       MR(is1(j))=etamin+(etamax-etamin)*(j-1)/(nTL-1);
    end
    MFv=miumin*ones(1,nTL);
    for j=1:nBP;
       MFv(is2(j))=miumin+(miumax-miumin)*(nBP-j)/(nBP-1);
    end
    for j=1:nTL;
       Memb(j)=max(MR(j),MFv(j));
    end
    [sortMemb,iMembs]=sort(Memb);
    TLtmp=TL(iMembs,:); TL=TLtmp;
    rectmp=rec(iMembs); rec=rectmp;
    fTLtmp=fTL(iMembs); fTL=fTLtmp;
    TL(1,:)=x; fTL(1)=fx; rec(1)=itc+1;
end
dxVR=zeros(1,inVRL);
```

```
    for j=1:inVRL
        dxVR(j)=norm(x-VRL(j,:));
        if dxVR(j) < dVR
            frVRL(j)=frVRL(j)+1;
        end
    end
            if(min(dxVR) > dVR)
        inVRL=inVRL+1; VRL(inVRL,:)=x; frVRL(inVRL)=1;
    end
    if fnew<fx
        inNoImprov=0;
    else
        inNoImprov=inNoImprov+1;
    end
    x=xnew; fx=fnew;
end
[fTLs,ifTL]=sort(fTL);
TLtmp=TL(ifTL,:);
TL=TLtmp; fTL=fTLs;
rectmp=rec(ifTL); rec=rectmp;
FMin1=fTL(1); XMin1=TL(1,:); XM1=TL;
TLm=TL; fTLm= fTL; recm=rec; VRLm=VRL; frVRLm=frVRL;
function [XMin,FMin,FCount] = DTSps(f,U,L)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%
%           Directed Tabu Search (APS)              %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%
%
% A. Hedar, April, 2003
```

```
%
% Reference:
%       A. Hedar and M. Fukushima, "TS Directed by direct search methods
%       for nonlinear global optimization", Technical report 2003-007,
%       Department of Applied Mathematics and Physics,
%       Kyoto University (June 2003).
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%          function [XMin,FMin,FCount] = DTSps(f,U,L)                %
% Inputs:
%   f = Objective function
%   U = n-Dimension Vector of upper limits of the objective variables
%   L = n-Dimension Vector of lower limits of the objective variables
%
% Outputs:
%   XMin = Best point obtained by DTSps
%   FMin = Best function value obtained by DTSps
%          FCount = Number of function evaluations
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Intial Parameters %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
n=length(U);
Edg=0.1*min(U-L);              % Simplex edge length
FCount=0;                      % Number of function evaluations
tol =1.0D-6;                   % Termanation acurcy
MaxItrs=10*n;
itrc=0;
NoImprv=2*n;
inNoImprv=0;
```

```
H=1;                    % Ratio of accepting Diversification point
dVR=2*Edg;              % Region redius in VRL
%
%%%%%%%%%%% Generation of the initial Solution %%%%%%%%%%
%
r=rand(1,n);
x=L+(U-L).*r;
fx=feval(f,x);
FCount=FCount+1;
fbetter=fx;
S=zeros(n+2,n);
fv=zeros(n+2,1);
id=eye(n);
%
% **************** 1st iteration *****************
% Constructing the Neighborhood Search
in=1;
while(fbetter >= fx & in <= n)
   EdgL=Edg+rand*Edg;
   sn=(-1)^(round(rand));
   S(in,:)=x+sn*EdgL*id(in,:);
   fv(in)=feval(f,S(in,:));
   FCount=FCount+1;
   if fv(in)<fbetter
      x1=S(in,:);
      f1=fv(in);
      Bflag=1;
      fbetter=fv(in);
   else
      Bflag=0;
```

```
    end
    in=in+1;
end
if (Bflag == 0)
    df=zeros(n,1);
    u=zeros(n,n);
    v=zeros(1,n);
    for j=1:n;
        df(j)=fv(j)-fx;
            end
            for i=1:n;
        w(i)=df(i)/sum(abs(df));
        u(i,:)=-(S(i,:)-x);
        u(i,:)=u(i,:)/norm(u(i,:));
            end
            for j=1:n; v=v+w(j)*u(j,:); end
    v=v/norm(v);
    S(n+1,:)=x+(0.5*rand)*Edg*v;
    S(n+2,:)=x+(0.5+0.5*rand)*Edg*v;
    for j=1:2
        jn=n+j;
        fv(jn)=feval(f,S(jn,:));
    end
    FCount=FCount+2;
    [fvs,is]=sort(fv);
    Stmp=S(is,:); S=Stmp; fv=fvs;
    xnew=S(1,:);
    fnew=fv(1);
else
    xnew=x1; fnew=f1;
```

```
end
% TL and VRL Setting
inTL=1; TL(inTL,:)=x; fTL(inTL)=fx; rec(inTL)=1;
inVRL=1; VRL(inVRL,:)=x; frVRL(inVRL)=1;
x=xnew; fx=fnew;
Xold=x; Fold=fx;
%%%%%%%%%%%%%%%%%%%%%%%% Main Loop
%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
while(inNoImprv < NoImprv & itrc < MaxItrs)
[FM1,XM1,FCount1,TL1,fTL1,rec1,VRL1,frVRL1]=APS(f,n,x,fx,Edg,TL,fTL,rec,VRL,frVRL);
  TL=TL1;
  fTL=fTL1;
  rec=rec1;
  VRL=VRL1;
  frVRL=frVRL1;
        FCount=FCount+FCount1;
        % Diversification Solution
        accpt=0;
        while accpt==0
                r=rand(1,n);
                xnw=L+(U-L).*r;
                for j=1:inVRL
       dxV(j)=norm(xnw-VRL(j,:))*(1+0.25*(1-exp(-0.25*(frVRL(j)-1))));
                end
    if min(dxV)>= H*dVR
      accpt=1;
    end
        end
        x=xnw;
```

```
        fx=feval(f,x);
        FCount=FCount+1;
        % Update TL
%
[sortrec,iTLr]=sort(rec);
        TLrtmp=TL(iTLr,:); TL=TLrtmp;
        fTLtmp=fTL(iTLr); fTL=fTLtmp;
        rec=sortrec;
inTL=length(fTL);
for j=1:inTL
   rec(j)=j;
end
%
        [sortfTL,iTL]=sort(fTL);
        TLtmp=TL(iTL,:); TL=TLtmp;
        rectmp=rec(iTL); rec=rectmp;
        fTL=sortfTL;
        fTL(inTL)=fx;
        TL(inTL,:)=x;
        rec(inTL)=inTL+1;
%
inVRL=length(frVRL);
VRL(inVRL+1,:)=x; frVRL(inVRL+1)=1;
%
if FM1 <= Fold
   inNoImprv = 0;
else
   inNoImprv = inNoImprv+1;
end
Xold=XM1; Fold=FM1;
```

```
    itrc=itrc+1;
end
%
%%%%%%%%%%%%%% Intensification Search %%%%%%%%%%%%%%%%%
%
[fTLs,ifTL]=sort(fTL);
TLtmp=TL(ifTL,:);
TL=TLtmp; fTL=fTLs;
XMin=TL(1,:); XM=TL;
FMin=fTL(1);
% Applying N-M method starting from Best Point
if n<5
    maxitt=100*n;
    budget=200*n;
else
    maxitt=5000*n;
    budget=10000*n;
end
for j=1:1
    [x1,FCoun1,f1]=NeMe(n,f,XM(j,:),maxitt,budget);
    if f1 < FMin
%           % Fixed by Por
%           XMin=x1(:,1);
        % Original
         XMin=x1(:,1);
        FMin=f1;
    end
    FCount=FCount+FCoun1;
    end
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%% END
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
   function [z,fcount,f1,lhist,histout,simpdata]=NeMe(n,f,x0,maxit,budget)
%
% Nelder-Mead optimizer, No tie-breaking rule other than MATLAB's sort
%
% C. T. Kelley, December 12, 1996
%
%
% This code comes with no guarantee or warranty of any kind.
%
% function [x,lhist,histout,simpdata] = nelder(x0,f,tol,maxit,budget)
%
% inputs:
%         vertices of initial simplex = x0 (n x n+1 matrix)
%         The code will order the vertices for you and no benefit is
%         accrued if you do it yourself.
%
%      objective function = f
%
%      termination tolerance = tol
%      maximum number of iterations = maxit (default = 100)
%          As of today, dist = | best value - worst value | < tol
%          or when maxit iterations have been taken
%      budget = max f evals (default=50*number of variables)
%               The iteration will terminate after the iteration that
%               exhausts the budget
%
%
% outputs:
```

```
%          final simplex = x (n x n+1) matrix
%
%       number of iterations before termination = itout (optional)
%       iteration histor = histout itout x 5
%         histout = iteration history, updated after each nonlinear iteration
%                = lhist x 5 array, the rows are
%                  [fcount, fval, norm(grad), dist, diam]
%                  fcount = cumulative function evals
%                  fval = current best function value
%                  norm(grad) = current simplex grad norm
%                  dist = difference between worst and best values
%                  diam = max oriented length
%       simpdata = data for simplex gradient restart
%              = [norm(grad), cond(v), bar f]
%
% initialize counters
%
lhist=0; fcount=0;
%
% set debug=1 to print out iteration stats
%
debug=0;
%
% Set the N-M parameters
%
rho=1; chi=2; gamma=.5; sigma=.5;
tol=1.0D-8;
% Generation of the initial point
for i=1:n; z(i,1)=x0(i); end
fv(1)=feval(f,z(:,1));
```

```
fcount=1;
edgel=1;
edge=edgel*ones(n,1);
id=eye(n);
for j=2:n+1; z(:,j)=z(:,1)+edge(j-1)*id(:,j-1); end
for j=2:n+1; fv(j)=feval(f,z(:,j)); end;
fcount=fcount+n;
if nargin < 4 maxit=100*n; end
if nargin < 5 budget=200*n; end
if n >= 10; maxit = 10*maxit; budget=10*budget; end
%
% set the paramters for stagnation detection/fixup
% setting oshrink=0 gives vanilla Nelder-Mead
%
oshrink=1; restartmax=3; restarts=0;
%
%
% Order the vertices for the first time
%
%x=x0;
histout=zeros(maxit*3,5); simpdata=zeros(maxit,3);
itout=0; orth=0;
ztmp=zeros(n,n+1); delf=zeros(n,1);
[fs,is]=sort(fv); ztmp=z(:,is); z=ztmp; fv=fs;
f1=fv(1);
itc=0; dist=fv(n+1)-fv(1);
diam=zeros(n,1);
for j=2:n+1
  v(:,j-1)=-z(:,1)+z(:,j);
  delf(j-1)=fv(j)-fv(1);
```

```
  diam(j-1)=norm(v(:,j-1));
end
sgrad=v'\delf; alpha=1.d-4*max(diam)/norm(sgrad);
lhist=lhist+1;
histout(lhist,:)=[fcount, fv(1), norm(sgrad,inf), 0, max(diam)];
%
% main N-M loop
%
while(itc < maxit & dist > tol & restarts < restartmax & fcount <= budget)
   fbc=sum(fv)/(n+1);
   xbc=sum(z')'/(n+1);
   sgrad=v'\delf;
   simpdata(itc+1,1)=norm(sgrad);
   simpdata(itc+1,2)=cond(v);
   simpdata(itc+1,3)=fbc;
   if(det(v) == 0)
      disp('simplex collapse')
      break
   end
   happy=0; itc=itc+1; itout=itc;
%
% reflect
%
   y=z(:,1:n);
   zbart = sum(y')/n;  % centriod of better vertices
   zbar=zbart';
   zr=(1 + rho)*zbar - rho*z(:,n+1);
   fr=feval(f,zr); fcount=fcount+1;
   if(fr >= fv(1) & fr < fv(n)) happy = 1; zn=zr; fn=fr; end;
%    if(happy==1) disp(' reflect '); end
```

```
%
% expand
%
    if(happy == 0 & fr < fv(1))
        ze = (1 + rho*chi)*zbar - rho*chi*z(:,n+1);
        fe=feval(f,ze); fcount=fcount+1;
        if(fe < fr) zn=ze;  fn=fe; happy=1; end
        if(fe >=fr) zn=zr;  fn=fr; happy=1; end
%       if(happy==1) disp(' expand '); end
    end
%
% contract
%
   if(happy == 0 & fr >= fv(n) & fr < fv(n+1))
%
% outside contraction
%
       zc=(1 + rho*gamma)*zbar - rho*gamma*z(:,n+1);
       fc=feval(f,zc); fcount=fcount+1;
       if(fc <= fr) zn=zc; fn=fc; happy=1; end;
%      if(happy==1) disp(' outside '); end;
   end
%
% inside contraction
%
   if(happy == 0 & fr >= fv(n+1))
       zc=(1 - gamma)*zbar+gamma*z(:,n+1);
       fc=feval(f,zc); fcount=fcount+1;
       if(fc < fv(n+1)) happy=1; zn=zc; fn=fc; end;
%   if(happy==1) disp(' inside '); end;
```

```
  end
%
% test for sufficient decrease,
% do an oriented shrink if necessary
%
  if(happy==1 & oshrink==1)
    zt=z; zt(:,n+1)=zn; ft=fv; ft(n+1)=fn;
%     xt=x; xt(:,n+1)=xn; ft=fv; ft(n+1)=feval(f,xn); fcount=fcount+1;
    fbt=sum(ft)/(n+1); delfb=fbt-fbc; armtst=alpha*norm(sgrad)^2;
    if(delfb > -armtst/n)
      restarts=restarts+1;
      orth=1; diams=min(diam);
      sx=.5+sign(sgrad); sx=sign(sx);
      if debug==1
        [itc, delfb, armtst]
      end
      happy=0;
      for j=2:n+1; z(:,j)=z(:,1);
      z(j-1,j)=z(j-1,j)-diams*sx(j-1); end;
    end
  end
%
% if you have accepted a new point, nuke the old point and
% resort
%
  if(happy==1)
    z(:,n+1)=zn; fv(n+1)=fn;
%     x(:,n+1)=xn; fv(n+1)=feval(f,xn); fcount=fcount+1;
    [fs,is]=sort(fv); ztmp=z(:,is); z=ztmp; fv=fs;
  end
```

```
%
% You're in trouble now! Shrink or restart.
%
%    if(restarts >= restartmax) disp(' stagnation in Nelder-Mead'); end;
  if(happy == 0 & restarts < restartmax)
%       if(orth ~=1) disp(' shrink '); end;
      if(orth ==1)
      if debug == 1 disp(' restart '); end
      orth=0; end;
      for j=2:n+1;
          z(:,j)=z(:,1)+sigma*(z(:,j)-z(:,1));
          fv(j)=feval(f,z(:,j));
      end
      fcount=fcount+n;
      [fs,is]=sort(fv); ztmp=z(:,is); z=ztmp; fv=fs;
  end
%
%  compute the diameter of the new simplex and the iteration data
%
  for j=2:n+1
      v(:,j-1)=-z(:,1)+z(:,j);
      delf(j-1)=fv(j)-fv(1);
      diam(j-1)=norm(v(:,j-1));
  end
  dist=fv(n+1)-fv(1);
  lhist=lhist+1;
  sgrad=v'\delf;
  histout(lhist,:)=[fcount, fv(1), norm(sgrad,inf), dist, max(diam)];
  f1=fv(1);
end
```

ฟังก์ชันเป้าหมายเป็น MSE

```
function MSE = objFcn(b,x,y)
% x = [x0 x1 x2 x3 x4 x5];
% y = y
% n = sample size
% k = number of random variable
[n p]=size(x);
k = p-1;
[r c] = size(b);
if r<c
   b=b';
end
yy=(x*b);
MSE =((y-yy)'*(y-yy))/(n-k-1);
end
```

ฟังก์ชันเป้าหมายเป็น MSE+Penalty Function

```
function MSE_mod = objFcn_mod(b,x,y)
% x = [x0 x1 x2 x3 x4 x5 x6 x7 x8 c];
% y = y
% n = sample size
% k = number of random variable
[n p]=size(x);
k = p-1;
[r c] = size(b);
if r<c
   b=b';
end
c = b(end);
beta = b(1:end-1);
yy=(x*beta);
```

```
MSE =((y-yy)'*(y-yy))/(n-k-1);
MSE_mod = MSE+(abs(c)*(beta'*beta)/(n-p));
% fprintf('MSE=%.2f|c=%.2f|MSE_mod=%.2f\n',MSE,c,MSE_mod);
end
```

ประวัติผู้เขียน

ชื่อ นามสกุล	นางสาวนิศาชล งามประเสริฐสิทธิ์
ประวัติการศึกษา	วิทยาศาสตรบัณฑิต (คณิตศาสตร์) มหาวิทยาลัยธรรมศาสตร์ ปีที่สำเร็จการศึกษา พ.ศ. 2552
ผลงานทางวิชาการ	นิศาชล งามประเสริฐสิทธิ์. 2555.การคัดเลือกตัวแปรในตัวแบบการถดถอยเชิงเส้นพหุในกรณีที่ตัวแปรอิสระมีความสัมพันธ์กันสูงโดยวิธีการถดถอยแบบขั้นตอนกับการค้นหาแบบต้องห้าม.ใน**การประชุมวิชาการสถิติและสถิติประยุกต์ระดับชาติ ครั้งที่13 ประจำปี 2555.“การวิจัยเชิงสถิติเพื่อการพัฒนาสังคมยุคใหม่” จัดโดย ภาควิชาสถิติ คณะวิทยาศาสตร์ มหาวิทยาลัยรามคำแหงร่วมกับสมาคมสถิติแห่งประเทศไทย เครือข่ายวิจัยสถิติศาสตร์ ณ โรงแรมสีดารีสอร์ท จังหวัดนครนายก วันที่ 17-18 พฤษภาคม 2555**
รางวัลหรือทุนการศึกษา	ทุนส่งเสริมการศึกษาประเภทที่ 1 (Full Scholarship) คณะสถิติประยุกต์ สถาบันบัณฑิตพัฒนบริหารศาสตร์