แบบจำลองทำนายโรคหลอดเลือดสมองด้วยเทคนิคการเรียนรู้ของเครื่อง

พชร ดอกชะเอม[1*], เรืองศักดิ์ ตระกูลพุทธิรักษ์[2]

บทคัดย่อ

โรคหลอดเลือดสมอง (Stroke) เป็นหนึ่งในสาเหตุการเสียชีวิตและทุพพลภาพที่สำคัญของประชากรทั่วโลก การวินิจฉัยโรคหลอดเลือดสมองในระยะเริ่มแรกมีความสำคัญอย่างมากในการลดอัตราการเสียชีวิตและความพิการที่ตามมา อย่างไรก็ตามการวินิจฉัยโรคหลอดเลือดสมองต้องอาศัยความเชี่ยวชาญของแพทย์ ซึ่งมีอยู่อย่างจำกัด ผู้วิจัยจึงเล็งเห็นการนำเทคนิคการเรียนรู้ของเครื่อง (Machine Learning) ในการสร้างแบบจำลองเพื่อช่วยในการจำแนกผู้ป่วยโรคหลอดเลือดสมอง โดยอาศัยข้อมูลคุณลักษณะของผู้ป่วยในการสร้างแบบจำลองเพื่อลดภาระของแพทย์และทำให้สามารถช่วยลดระยะเวลาคัดกรองผู้ป่วยได้

งานวิจัยนี้เป็นการศึกษาการสร้างแบบจำลองด้วยเทคนิคการเรียนรู้ของเครื่อง โดยชุดข้อมูลที่นำมาใช้ในการสร้างแบบจำลองมาจากเว็บไซต์ Kaggle ซึ่งเป็นข้อมูลทางคลินิกของผู้ป่วยมี 2 ประเภทผู้ คือ ผู้ป่วยปกติและผู้ป่วยโรคหลอดเลือดสมอง จำนวนทั้งหมด 5,110 คน ในการศึกษาข้อมูลชุดนี้มีลักษณะชุดข้อมูลไม่สมดุล (Imbalanced Data) ซึ่งอาจส่งผลต่อประสิทธิภาพของแบบจำลอง ทำให้ต้องนำเทคนิคการจัดการข้อมูลไม่สมดุลของข้อมูลด้วยวิธีต่างๆมาช่วยในการจัดการข้อมูลร่วมด้วย ในการหาแบบจำลองที่มีประสิทธิภาพดีที่สุดจะมาจากทำการเปรียบเทียบการสร้างแบบจำลองด้วยอัลกอริทึมที่หลากหลายได้แก่ Logistic Regression, Decision Tree, Random Forest, XGBoost, LightGBM, AdaBoost และCatBoost

การเปรียบเทียบจะใช้ตัววัดประสิทธิภาพที่มาจากผลลัพธ์การทำนายของแบบจำลองด้วย Confusion Matrix ประกอบด้วย ความแม่นยำ (Accuracy), ความอ่อนไหว, F1-score, Specificity, ROC Curve และความแม่นยำสมดุล (Balanced Accuracy) แต่ในงานวิจัยนี้จะให้ความสำคัญกับความแม่นยำสมดุลเป็นตัววัดประสิทธิภาพหลัก เป็นเพราะชุดข้อมูลไม่สมดุลที่มีความต่างของจำนวนประเภทข้อมูลทั้งสอง ทำให้ต้องเลือกใช้ตัววัดประสิทธิภาพที่ให้ความสำคัญกับน้ำหนักของประเภทจำนวนข้อมูล จากผลการสร้างแบบจำลองพบว่าแบบจำลองที่สร้างด้วยอัลกอริทึม AdaBoost ให้ประสิทธิภาพสูงที่สุดด้วยค่าความแม่นยำสมดุลที่ 0.72 และหากผู้ศึกษาต้องการเพิ่มประสิทธิภาพของแบบจำลองสามารถทำได้โดยการเพิ่มตัวอย่างข้อมูล และการปรับจูนพารามิเตอร์ (Parameter-Tuning) ด้วยอัลกอริทึม GridSearchCV

คำสำคัญ : การเรียนรู้ของเครื่อง, ชุดข้อมูลไม่สมดุล, ความแม่นยำสมดุล

---

[1] หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิทยาการข้อมูล คณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ กรุงเทพฯ 10110

[2] คณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ กรุงเทพฯ 10110

* Corresponding author: E-mail address: Potchara.dockchaam@g.swu.ac.th

STROKE PREDICTION MODEL USING MACHINE LEARNING TECHNIQUES

Potchara Dockchaam[1*], Ruangsak Trakunphutthirak[2]

Abstract

Stroke is one of the leading causes of death and disability worldwide. Early diagnosis of stroke is crucial in reducing mortality rates and subsequent disabilities. However, diagnosing stroke requires the expertise of medical professionals, which is limited. Researchers have recognized the potential of using Machine Learning techniques to create models that can help classify stroke patients based on patient characteristic data, thereby reducing the burden on doctors and enabling faster patient screening.

This research involved the study of model creation using Machine Learning techniques, with the dataset used for model creation coming from the Kaggle website. This dataset includes clinical data of 5,110 samples, comprising both normal individuals and stroke patients, and features imbalanced data, which can affect the performance of the model. Various techniques were employed to manage the imbalanced data. The study compared different models created using various algorithms including Logistic Regression, Decision Tree, Random Forest, XGBoost, LightGBM, AdaBoost, and CatBoost.

The comparison used performance metrics derived from the Confusion Matrix, including Accuracy, Sensitivity, F1-score, Specificity, ROC Curve, and Balanced Accuracy. However, this research prioritized Balanced Accuracy as the main performance metric due to the imbalanced data set, which required a performance metric that considered the weight of the data categories. The results showed that the model created with the AdaBoost algorithm had the highest performance with a Balanced Accuracy score of 0.72. If researchers want to improve the model's performance, they can do so by increasing the sample size and performing parameter tuning using the GridSearchCV algorithm.

Keywords : Machine Learning, Imbalanced Data, Machine Learning, Balanced Accuracy

---

[1] Data Science, Faculty of Science, Srinakharinwirot University, Bangkok, 10110, Thailand

[2] Faculty of Science, Srinakharinwirot University, Bangkok, 10110, Thailand

* Corresponding author: E-mail address: Potchara.dockchaam@g.swu.ac.th

บทนำ

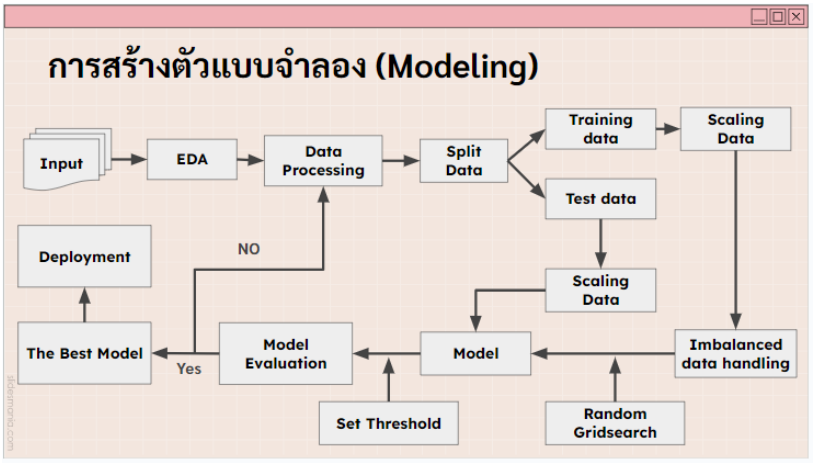
โรคหลอดเลือดสมอง (Stroke) เป็นหนึ่งในโรคที่เกิดขึ้นเมื่อการไหลเวียนของเลือดไปยังสมองถูกขัดขวางทำให้เซลล์สมองถูกทำลายและเกิดความเสียหายได้ ซึ่งความเสียหายนี้ส่งผลให้เกิดผลกระทบต่อร่างกายอย่างมาก เช่น ปากเบี้ยว ไม่มีแรง หรือถึงขั้นเสียชีวิต และยังมีโอกาสทำให้เกิดโรคแทรกซ้อนต่างๆระหว่างการรักษา ไม่ว่าจะเป็น อัมพฤกษ์ อัมพาต แต่สามารถลดความเสี่ยงและความเสียหายได้หากผู้ป่วยรักษาได้รับการรักษาได้อย่างถูกต้อง และได้รับคำแนะนำก่อนที่จะเป็นโรคหลอดเลือดสมอง [1]

ทางผู้วิจัยเห็นว่าการทำนายผู้ป่วยที่มีโอกาสจะเป็นโรคหลอดเลือดสมอง มีความสำคัญในการตัดสินใจที่จะช่วยวางแผนในการรักษาแก่ผู้ป่วยได้อย่างทันท่วงที จึงมีความคิดที่จะนำเอาเทคโนโลยีการเรียนรู้ของเครื่อง (Machine Learning) มาใช้สร้างแบบจำลองจำแนกเพื่อทำนายผู้ป่วยโรคหลอดเลือดสมองและผู้ป่วยปกติ ซึ่งเป็นการเรียนรู้ให้กับแบบจำลองแบบการเรียนรู้แบบมีผู้สอน Supervised Learning โดยใช้ข้อมูลทางการแพทย์และคลินิกเพื่อนำมาใช้คำนวณสร้างแบบจำลอง

ในงานวิจัยนี้ได้ทำการตั้งสมมติฐานในการศึกษาการสร้างแบบจำลอง ได้ทั้งหมด 3 ข้อ ดังนี้

1. ตัวประเมินประสิทธิภาพของแบบจำลอง มีปัญหาความไม่เหมาะสมในการคำนวณ กับชุดข้อมูลไม่สมดุล สามารถแก้ปัญหาด้วยหน่วยวัดประสิทธิภาพ ความแม่นยำสมดุล (Balanced Accuracy)
2. แบบจำลองที่ซับซ้อนมีโครงสร้างการคำนวณที่ซับซ้อนกว่าแบบจำลองพื้นฐาน ทำให้การจำแนกมีประสิทธิภาพที่สูงกว่า
3. การปรับ Threshold สามารถช่วยแก้ปัญหาของแบบจำลองในการตรวจจับการจำแนกกลุ่มข้อมูลจำนวนมาก (Majority Class) และกลุ่มข้อมูลจำนวนน้อย (Minority Class) อย่างไร

วิธีดำเนินการ



รูปที่ 1 แสดงกระบวนการดำเนินงานวิจัย

รูปที่ 1 อธิบายถึงกระบวนการสร้างแบบจำลองการทำนายโดนเริ่มจากขั้นตอนการนำเข้าข้อมูล การสำรวจข้อมูล เพื่อวิเคราะห์ความสัมพันธ์ระหว่างตัวแปรต่างๆ ที่เกี่ยวข้องกับการจำแนกผู้ป่วยโรคหลอดเลือดสมอง การเตรียมข้อมูล และคุณลักษณะ

ที่เกี่ยวข้องกับการจำแนก โดยขั้นตอนการจัดการข้อมูลจะตัดคุณลักษณะ id เนื่องจากไม่เกี่ยวข้องกับการทำนาย และลักษณะข้อมูลจัดเป็นกลุ่ม ประกอบด้วยคุณลักษณะ heart disease, marital status, gender, smoking status, work type, type of residence จะถูกเปลี่ยนให้เป็นตัวเลขด้วยเทคนิค Encoder หลังจากนั้นจะถูกทำ Scaling เพื่อลดการกระจายตัวของข้อมูล พร้อมกับลักษณะข้อมูลเชิงตัวเลข ที่ประกอบด้วยคุณลักษณะ age, average glucose level, body mass index ดังนั้นการสร้างแบบจำลองสามารถแบ่งเป็นขั้นตอนได้ ดังนี้

ขั้นตอนที่ 1 : แนะนำชุดข้อมูลที่ใช้ในการศึกษานี้

ในงานวิจัยนี้ได้ใช้ข้อมูลสาธารณะจากเว็บไซต์ Kaggle ซึ่งเป็นข้อมูลทางคลินิกเกี่ยวกับคุณลักษณะทางด้านร่างกายของผู้ป่วย 2 ประเภท คือ ผู้ป่วยปกติและผู้ป่วยโรคหลอดเลือดสมอง มีจำนวนข้อมูลทั้งหมด 5,110 ตัวอย่าง โดยข้อมูลจะมีคุณลักษณะทั้งหมด 12 คุณลักษณะ ประกอบด้วย id, gender, age, hypertension, heart_disease, ever_married, work_type, Residence_type:, avg_glucose_level, bmi, smoking_status และ stroke โดยคุณลักษณะที่ถูกตัดออกคือ id เนื่องจากไม่เกี่ยวข้องกับการนำไปคำนวณการสร้างแบบจำลอง

ตาราง 1 แสดงคุณลักษณะของชุดข้อมูลและคำอธิบายข้อมูล

ลำดับ	ข้อมูลตัวแปร (Variable)	คำอธิบายข้อมูล (Description)
1	id	ตัวระบุตำแหน่งข้อมูล
2	gender	เพศของผู้ป่วย
3	age	อายุของผู้ป่วย
4	hypertension	โรคความดันโลหิตสูง
5	heart_disease	โรคหัวใจ
6	ever_married	สถานะของผู้ป่วย
7	work_type	ประเภทอาชีพของผู้ป่วย
8	Residence_type	ตำแหน่งของที่อยู่อาศัยของผู้ป่วย
9	avg_glucose_level	ปริมาณน้ำตาลในเลือดของผู้ป่วย
10	bmi	ดัชนีความสมดุลร่างกายของผู้ป่วย
11	smoking_status	สถานะการสูบบุหรี่ของผู้ป่วย
12	stroke	อาการป่วยโรคหลอดเลือดสมองของผู้ป่วย

ขั้นตอนที่ 2 : การนำเข้าข้อมูล ตรวจสอบข้อมูล และพิจารณาข้อมูล

ใช้ภาษาไพธอน (Python) ในการนำเข้าและวิเคราะห์ โดยเริ่มจากการนำเข้าไฟล์ชุดข้อมูลด้วยการใช้ไลบรารี่ Pandas และสำรวจข้อมูลด้วยวิธีการทางสถิติ

	id	age	hypertension	heart_disease	avg_glucose_level	bmi	stroke
count	5110.000000	5110.000000	5110.000000	5110.000000	5110.000000	4909.000000	5110.000000
mean	36517.829354	43.226614	0.097456	0.054012	106.147677	28.893237	0.048728
std	21161.721625	22.612647	0.296607	0.226063	45.283560	7.854067	0.215320
min	67.000000	0.080000	0.000000	0.000000	55.120000	10.300000	0.000000
25%	17741.250000	25.000000	0.000000	0.000000	77.245000	23.500000	0.000000
50%	36932.000000	45.000000	0.000000	0.000000	91.885000	28.100000	0.000000
75%	54682.000000	61.000000	0.000000	0.000000	114.090000	33.100000	0.000000
max	72940.000000	82.000000	1.000000	1.000000	271.740000	97.600000	1.000000

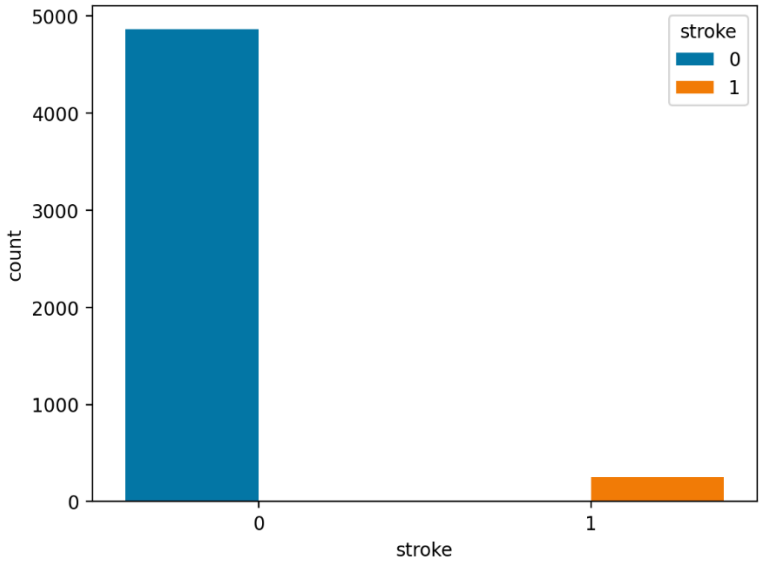
รูปที่ 2 แสดงการตรวจสอบข้อมูลเบื้องต้นด้วยวิธีทางสถิติ

จากรูปที่ 2 แสดงจำนวนและสถิติพื้นฐานของข้อมูลมีทั้งหมด 5,110 ตัวอย่าง แต่ในรายการ bmi มีเพียง 4,909 ตัวอย่าง ทำให้ต้องทำการเติมข้อมูลด้วยการใช้เทคนิค KNN Imputation ในการเติมข้อมูลด้วยการใช้ข้อมูลที่อยู่ใกล้ในการคำนวณ[2]

```
smoking_status
never smoked       1892
Unknown            1544
formerly smoked     885
smokes              789
Name: count, dtype: int64
```

รูปที่ 3 ข้อมูลถานะการสูบบุหรี่ของผู้ป่วย

จากรูปที่ 3 ข้อมูลสถานะการสูบบุหรี่ของผู้ป่วยมีจำนวน 1,544 รายการที่ไม่ทราบสถานะ (Unknown) ทางผู้วิจัยได้ทำการเปลี่ยนเป็นสถานะเป็นไม่สูบบุหรี่ (Never Smoked) เนื่องจากอัตราส่วนของผู้ที่ไม่ทราบสถานะและไม่เป็นโรคหัวใจ กับ ผู้ที่ไม่สูบบุหรี่และไม่เป็นโรคหัวใจ มีอัตราส่วนที่ใกล้เคียงกัน



รูปที่ 4 ข้อมูลถานะการสูบบุหรี่ของผู้ป่วย

จากรูปที่ 4 จำนวนของผู้ป่วยปกติ (กลุ่มที่ 0) และผู้ป่วยโรคหลอดเลือดสมอง (กลุ่มที่ 1) มีจำนวนของตัวอย่างข้อมูลที่แตกต่างกันมาก ทำให้สามารถสรุปชุดข้อมูลนี้ได้ว่าเป็นชุดข้อมูลไม่สมดุลในการจัดการข้อมูล

ขั้นตอนที่ 3 : การแบ่งข้อมูล

ผู้วิจัยได้นำข้อมูลมาทำการแบ่งข้อมูลออกเป็น ชุดข้อมูลฝึกสอน และชุดข้อมูลทดสอบที่อัตราส่วน 80:20 โดยแบ่งเป็นข้อมูลฝึกฝน (Training data) 4,088 ตัวอย่าง และข้อมูลทดสอบ (Test data) 1,022 ตัวอย่าง

```
X_train.shape, X_test.shape, y_train.shape, y_test.shape
✓ 0.0s
((4088, 10), (1022, 10), (4088,), (1022,))
```

รูปที่ 5 จำนวน**ข้อมูลหลังจากการทำแบ่งข้อมูล**

ขั้นตอนที่ 4 : การจัดการข้อมูล

หลังจากการแบ่งข้อมูลผู้วิจัยได้ลดการกระจายตัวของข้อมูลด้วยการใช้เทคนิค Scaling ด้วย StandardScaling และจัดการข้อมูล Categorical data ด้วย LabelEncoder เพื่อให้คุณลักษณะสามารถนำไปใช้ในการคำนวณได้อย่างเหมาะสม และการที่ข้อมูลมีจำนวนตัวอย่างที่ต่างกันมาก ทำให้ต้องใช้เทคนิคการจัดการชุดข้อมูลไม่สมดุล ในงานวิจัยนี้เทคนิคที่ใช้ประกอบด้วย

การลดข้อมูลแบบสุ่ม, การเพิ่มข้อมูลแบบสุ่ม, SMOTE และ Class Re-weight เพื่อให้ข้อมูลมีความเหมาะสมแก่การนำไปใช้ในการสร้างแบบจำลอง

```
Original      : Counter({0: 3889, 1: 199})
Undersampling : Counter({0: 199, 1: 199})
Oversampling  : Counter({0: 3889, 1: 3889})
SMOTE         : Counter({0: 3889, 1: 3889})
```

รูปที่ 6 จำนวนตัวอย่างหลังจากใช้เทคนิคการจัดการชุดข้อมูลไม่สมดุล

จากรูปที่ 6 จำนวนข้อมูลทดสอบมีจำนวนทั้งหมด 4,088 ตัวอย่าง แบ่งเป็น กลุ่มข้อมูลผู้ป่วยปกติ (0) 3,889 ตัวอย่าง และกลุ่มข้อมูลผู้ป่วยโรคหลอดเลือดสมอง (1) 199 ตัวอย่าง ซึ่งเป็นสัดส่วนที่ต่างกันมาก แต่เมื่อมีการจัดการข้อมูลด้วยเทคนิคการจัดการข้อมูลไม่สมดุล ทำให้สัดส่วนของกลุ่มข้อมูลมีจำนวนที่เท่ากันและเหมาะสมกับการนำไปใช้ในการสร้างแบบจำลอง

ขั้นตอนที่ 5 : การสร้างแบบจำลองสำหรับจัดกลุ่มข้อมูล

ในงานวิจัยนี้ได้ใช้แบบจำลองที่เหมาะสมกับงานจำแนกกลุ่มข้อมูล (Classification) ทางผู้วิจัยได้ดำเนินการทดลองเปรียบเทียบประสิทธิภาพของแบบจำลอง ที่ใช้ชุดข้อมูลปกติ และใช้เทคนิคการจัดการข้อมูลชุดข้อมูลไม่สมดุลที่แตกต่างกัน ประกอบด้วย การลดข้อมูลแบบสุ่ม, การเพิ่มข้อมูลแบบสุ่ม, SMOTE และ Class Re-weight แล้วนำมาทำการสร้างแบบจำลองที่สร้างด้วยอัลกอริทึมพื้นฐาน เช่น Logistic Regression, Decision Tree และอัลกอริทึมซับซ้อน เช่น Random Forest, XGBoost, AdaBoost, LightGBM และ CatBoost ร่วมกับการใช้เทคนิคการจัดการชุดข้อมูลไม่สมดุล ซึ่งรูปแบบของการสร้างแบบจำลองแบ่งเป็น 2 รูปแบบ คือ

- การสร้างแบบจำลองด้วยการใช้ชุดข้อมูลดั้งเดิม และเทคนิคการจัดการชุดข้อมูลไม่สมดุล
- การสร้างแบบจำลองด้วยการใช้ชุดข้อมูลดั้งเดิม และเทคนิคการจัดการชุดข้อมูลไม่สมดุล ร่วมกับการใช้เทคนิคการปรับน้ำหนักของกลุ่มข้อมูลด้วยเทคนิค Class Re-weight

ขั้นตอนที่ 6 : การประเมินประสิทธิภาพของแบบจำลอง

งานวิจัยนี้ได้ประเมินแบบจำลองเพื่อทำการเปรียบเทียบแบบจำลองที่สร้างด้วยอัลกอริทึม และเทคนิคการจัดการชุดข้อมูลไม่สมดุลที่แตกต่างกัน ตัววัดประสิทธิภาพของแบบจำลองคำนวณจากผลลัพธ์ของ Confusion Matrix ที่แบบจำลองทำนายได้ ประกอบด้วย ความแม่นยำ (Accuracy), ความอ่อนไหว (Recall), F1-score, Specificity, Receiver Operating Characteristic Curve (ROC Curve) และความแม่นยำสมดุล (Balanced Accuracy) แต่การที่ชุดข้อมูลที่นำมาใช้ในการสร้างแบบจำลองมีความ

ไม่สมดุล ทำให้แบบจำลองมีความอคติต่อกลุ่มข้อมูลที่มีจำนวนน้อย ทำให้ตัววัดประสิทธิภาพบางตัวมีค่าที่มาก และน้อยเกินความเป็นจริงทำให้ต้องเลือกใช้ตัววัดประสิทธิภาพที่สามารถสะท้อนความสามารถในการทำนายกลุ่มข้อมูลได้ทั้งสองกลุ่ม โดยผู้วิจัยเลือกใช้ค่าความแม่นยำสมดุลในการเป็นตัววัดประสิทธิภาพหลัก

ความแตกต่างของความแม่นยำสมดุล และ ROC Curve คือ ROC Curve นั้นถูกคำนวณมาจากมาจากความสามารถในการทำนาย (Predicted scores) แต่ความแม่นยำสมดุล คำนวณมาจากความถี่ที่แบบจำลองสามารถทำนายได้ (Predicted class) ที่มาจาก Confusion Matrix การที่ข้อมูลจำนวนน้อยได้รับการทำนายไม่ว่าจะถูกหรือผิดจะมีโอกาสที่จะส่งผลให้ตัววัดประสิทธิภาพมีการเปลี่ยนแปลงมากกว่า[3]ดังนั้นการเลือกใช้ตัววัดประสิทธิภาพในงานวิจัยนี้เลือกใช้ตัววัดประสิทธิภาพความแม่นยำสมดุล เนื่องจากชุดข้อมูลไม่สมดุลในประเภทข้อมูลที่เราสนใจมีจำนวนที่น้อยกว่า และสนใจผลลัพธ์ของความถี่ของการทำนายผลของแบบจำลอง

ผลการวิจัยและอภิปรายผลการวิจัย

ในการวิจัยศึกษาการสร้างแบบจำลองเพื่อจำแนกผู้ป่วยโรคหลอดเลือดสมอง ซึ่งใช้ข้อมูลคุณลักษณะผู้ป่วยในการเรียนรู้ให้แบบจำลอง โดยใช้เทคนิคการเรียนรู้ของเครื่อง ผู้วิจัยได้ดำเนินการวิจัยโดยการศึกษาตามขบวนการและขั้นตอนต่างๆตลอดจนการประเมินผลของแบบจำลองการจำแนก เพื่อให้สอดคล้องกับสมมติฐานที่ได้ตั้งไว้ดังนี้

1. ตัวประเมินประสิทธิภาพของแบบจำลอง มีปัญหาความไม่เหมาะสมในการคำนวณ กับชุดข้อมูลไม่สมดุล สามารถแก้ปัญหาด้วยหน่วยวัดประสิทธิภาพ ความแม่นยำสมดุล (Balanced Accuracy)
2. แบบจำลองที่ซับซ้อนมีโครงสร้างการคำนวณที่ซับซ้อนกว่าแบบจำลองพื้นฐาน ทำให้การจำแนกมีประสิทธิภาพที่สูงกว่า
3. การปรับ Threshold สามารถช่วยแก้ปัญหาของแบบจำลองในการตรวจจับการจำแนกกลุ่มข้อมูลจำนวนมาก (Majority Class) และกลุ่มข้อมูลจำนวนน้อย (Minority Class) อย่างไร

สมมติฐานที่ 1 ตัวประเมินประสิทธิภาพของแบบจำลอง มีปัญหาความไม่เหมาะสมในการคำนวณ กับชุดข้อมูลไม่สมดุล สามารถแก้ปัญหาด้วยหน่วยวัดประสิทธิภาพ ความแม่นยำสมดุล (Balanced Accuracy)

จากผลการทดลองของงานวิจัย ได้ทำการใช้อัลกอริทึมทั้งหมด 7 อัลกอริทึมในการเปรียบเทียบ แบบจำลองที่สร้างด้วยอัลกอริทึม Adaboost เป็นแบบจำลองที่ให้ประสิทธิภาพที่สุด โดยใช้ความแม่นยำสมดุลเป็นตัววัดประสิทธิภาพหลัก โดยมีหลักการคำนวณดังนี้

$$\text{Balanced Accuracy} = \frac{\text{sensitivity} + \text{specficity}}{2} \qquad (1)$$

ตารางที่ 2 แสดงผลลัพธ์และประสิทธิภาพที่ดีที่สุดของแบบจำลองที่สร้างด้วยอัลกอริทึม Adaboost

Adaboost Algorithm									
Imbalance Type	Threshold	TN	FN	TP	FP	Accuracy	Recall	ROC	Balance Accuracy
Original Data	0.1	581	8	42	391	0.61	0.84	0.70	0.72
	0.2	972	50	0	0	0.95	0.00	0.70	0.50
	0.3	972	50	0	0	0.95	0.00	0.70	0.50
	0.4	972	50	0	0	0.95	0.00	0.70	0.50
	0.5	972	50	0	0	0.95	0.00	0.70	0.50
SMOTE	0.1	0	0	50	50	0.05	1.00	0.60	0.50
	0.2	0	0	50	972	0.05	1.00	0.60	0.50
	0.3	0	0	50	972	0.05	1.00	0.60	0.50
	0.4	0	0	50	972	0.05	1.00	0.60	0.50
	0.5	0	0	50	972	0.05	1.00	0.60	0.50
Undersampling	0.1	581	8	42	391	0.61	0.84	0.71	0.72
	0.2	972	50	0	0	0.95	0.00	0.71	0.50
	0.3	972	50	0	0	0.95	0.00	0.71	0.50
	0.4	972	50	0	0	0.95	0.00	0.71	0.50
	0.5	972	50	0	0	0.95	0.00	0.71	0.50
Oversampling	0.1	0	0	50	972	0.05	1.00	0.60	0.50
	0.2	0	0	50	972	0.05	1.00	0.60	0.50
	0.3	0	0	50	972	0.05	1.00	0.60	0.50
	0.4	0	0	50	972	0.05	1.00	0.60	0.50
	0.5	959	50	0	13	0.94	0.00	0.60	0.49

จากตารางที่ 2 หากเราเลือกใช้ตัววัดประสิทธิภาพไม่เหมาะสม ตัววัดประสิทธิภาพจะไม่ได้สะท้อนถึงความสามารถในการทำนายของแบบจำลอง ดังตารางตัวอย่าง อัลกอริทึม Adaboost กับ ชุดข้อมูลดั้งเดิม (Original Data) ค่าความแม่นยำจะมีค่าที่สูงกว่าความเป็นจริง แบบจำลองมีความอคติต่อกลุ่มข้อมูล ทำให้ไม่สามารถทำนายข้อมูลที่เราสนใจได้ถูกต้องทำให้ในงานวิจัยนี้ทางผู้วิจัยเลือกใช้ค่าความแม่นยำสมดุล เนื่องจากแสดงความสามารถในการทำนายข้อมูลทุกประเภท

เมื่อทำการใช้ความแม่นยำสมดุลเป็นตัววัดประสิทธิภาพของแบบจำลอง อัลกอริทึมที่ให้ประสิทธิภาพสูงที่สุด โดยการใช้ความแม่นยำสมดุลมีทั้งหมด 4 แบบจำลอง ซึ่งเป็นแบบจำลองที่สร้างด้วยอัลกอริทึม Adaboost ทั้งหมดแบบจำลองแรกคือ การใช้อัลกอริทึม Adaboost กับการใช้ชุดข้อมูลปกติ และแบบจำลองที่สองคือ การใช้อัลกอริทึม Adaboost ปรับน้ำหนักของข้อมูลด้วย

Class Re-weight และแบบจำลองที่สาม อัลกอริทึม Adaboost ใช้ร่วมกับเทคนิค Undersampling ในการจัดการปัญหาชุดข้อมูลไม่สมดุล และแบบจำลองที่สี่ อัลกอริทึม Adaboost ใช้เทคนิค Undersampling ร่วมกับการใช้ Class Re-weight ที่ Threshold ที่ 0.1

สาเหตุที่การใช้เทคนิค Class Re-weight แล้วประสิทธิภาพได้ไม่ดีขึ้น เป็นเพราะ Adaboost จัดอยู่ในอัลกอริทึมประเภท Boosting model[4] และถูกสร้างมาเพื่อให้มีความสามารถในการจัดการชุดข้อมูลไม่สมดุล[5] โดยมีหลักการทำงาน คือ การสร้างแบบจำลองที่มีประสิทธิภาพสูงจากแบบจำลองที่มีประสิทธิภาพ ด้วยการนำแบบจำลองประสิทธิภาพต่ำมาทำการเรียนรู้ Adaboost จะทำการปรับน้ำหนักของ Feature และข้อมูลไปจนกว่าจะได้แบบจำลองที่มีประสิทธิภาพสูงสุด ดังนั้นการใช้ Class Re-weight ร่วมกับ Adaboost ในชุดข้อมูลนี้จะไม่ได้ผลลัพธ์ที่ดีขึ้นเพราะการปรับน้ำหนักของกลุ่มข้อมูลเป็นค่ามาตรฐานของ Adaboost อยู่แล้ว[6]

สมมติฐานที่ 2 แบบจำลองที่ซับซ้อนมีโครงสร้างการคำนวณที่ซับซ้อนกว่าแบบจำลองพื้นฐาน ทำให้การจำแนกมีประสิทธิภาพที่สูงกว่า

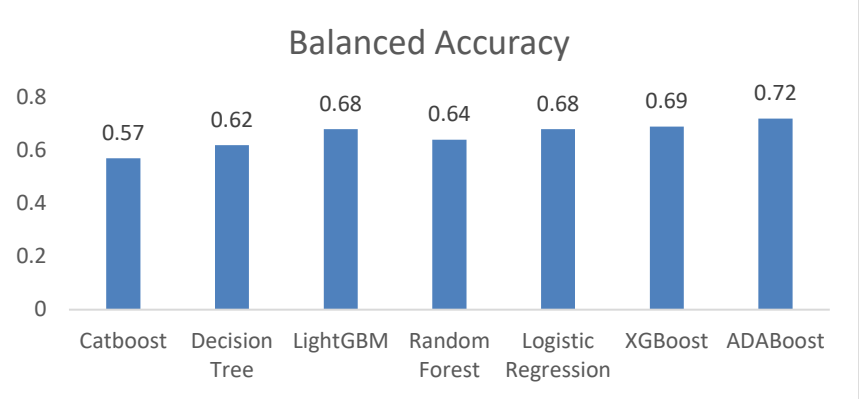
จากผลการศึกษาพบว่าในบางกรณีการใช้แบบจำลองพื้นฐานสามารถให้ความสามารถในการจำแนกได้สูงกว่าแบบจำลองที่ซับซ้อนได้เช่นกัน หากเทคนิคและอัลกอริทึมที่นำมาใช้ไม่เหมาะสมกับข้อมูล[7]

ตารางที่ 3 แสดงผลการเปรียบเทียบประสิทธิภาพของแบบจำลอง Decision Tree และ Catboost

Type	Threshold	TN	FN	TP	FP	Accuracy	Recall	ROC	Balance Accuracy
Decision Tree + Undersampling + Class Re-weight	0.1	844	33	17	128	0.84	0.34	0.62	0.60
	0.2	868	42	8	104	0.86	0.16	0.62	0.53
	0.3	849	33	17	123	0.85	0.30	0.62	0.59
	0.4	868	37	13	104	0.87	0.34	0.62	0.62
	0.5	865	37	13	107	0.86	0.26	0.62	0.57
Catboosting + SMOTE + Class Re-weight	0.1	788	34	6	184	0.79	0.32	0.64	0.57
	0.2	871	42	8	101	0.86	0.16	0.64	0.53
	0.3	915	42	8	57	0.90	0.16	0.64	0.55
	0.4	940	46	4	32	0.92	0.08	0.64	0.52
	0.5	941	48	2	31	0.92	0.04	0.64	0.50

จากตารางที่ 3 แบบจำลองที่สร้างด้วยอัลกอริทึม Decision Tree หากสามารถนำมาใช้ร่วมกับเทคนิคการจัดการข้อมูลไม่สมดุลที่มีความเหมาะสมมากกว่าเมื่อวัดด้วยความแม่นยำสมดุล สามารถให้ประสิทธิภาพที่สูงกว่าอัลกอริทึม Catboost ที่มีอัลกอริทึมที่ซับซ้อนกว่าได้ เนื่องมาจากการเลือกใช้เทคนิคการจัดการข้อมูลที่เหมาะสม ทำให้การทำนายของ Decision Tree มีสัดส่วนการทำนายที่ถูกต้องของกลุ่มข้อมูลทั้ง 2 ประเภทของแบบจำลอง Decision Tree ทำได้มีประสิทธิภาพที่สูงกว่า

รูปที่ 7 ประสิทธิภาพความแม่นยำสมดุลที่ดีที่สุดของแบบจำลองที่สร้างด้วยแต่ละอัลกอริทึม



จากรูปที่ 7 การทำงานของ Catboost มีความซับซ้อน ส่งผลให้ผลลัพธ์ที่ได้อาจไม่ดีกับชุดข้อมูลนี้มีค่าความแม่นยำสมดุลอยู่ที่ 0.57 ซึ่งการเลือกใช้ Decision Tree กับข้อมูลชุดนี้ซึ่งให้ผลลัพธ์ที่ดีกว่ามีค่าอยู่ที่ 0.62 แต่ในบางแบบจำลองที่มีความซับซ้อนก็ให้ประสิทธิภาพที่ดีกว่าแบบจำลองพื้นฐาน เช่น XGBoost และ AdaBoost ที่ให้ประสิทธิภาพของแบบจำลองได้ดีกว่า Decision Tree โดยมีค่าความแม่นยำสมดุลอยู่ที่ 0.69 และ 0.72

การเลือกใช้อัลกอริทึมขึ้นอยู่กับความเหมาะสมของชุดข้อมูล เช่น CatBoost เป็นอัลกอริทึมที่ไม่เหมาะกับการใช้กับชุดข้อมูลที่กลุ่มของข้อมูลมีความคล้ายคลึงกัน[8] ดังนั้นเมื่อใช้กับชุดข้อมูลผู้ป่วยโรคหลอดเลือดสมองกับผู้ป่วยปกติที่มีคุณลักษณะของข้อมูลคล้ายคลึงกัน ทำให้ได้ประสิทธิภาพที่ต่ำกว่าแบบจำลองพื้นฐานอย่าง Decision Tree

สมมติฐานที่ 3 การปรับ Threshold สามารถช่วยแก้ปัญหาของแบบจำลองในการตรวจจับการจำแนกกลุ่มข้อมูลจำนวนมาก (Majority Class) และกลุ่มข้อมูลจำนวนน้อย (Minority Class) อย่างไร

การปรับ Threshold เป็นการเปลี่ยนผลลัพธ์การทำนายให้อยู่ในรูปแบบของความน่าจะเป็น โดยค่ามาตรฐานในการให้ความน่าจะเป็นในการทำนายจะอยู่ที่ 0.5 หากเป็นการทำนายแบบ 2 กลุ่มของข้อมูลจะอยู่ที่ 50% : 50% ทำให้โอกาสที่จะทำนายข้อมูลทั้งสองกลุ่มมีค่าเท่ากัน และหากปรับค่าให้อยู่ที่ 0.1 โอกาสในการทำนายกลุ่มข้อมูลที่เราสนใจซึ่งเป็นกลุ่มข้อมูลจำนวนน้อยมีความน่าจะเป็นที่จะเป็นในการทำนายกลุ่มข้อมูลนี้หากมีค่าอยู่ระหว่าง 0-10% จะทำให้แบบจำลองทำนายเป็นกลุ่มข้อมูลที่เราสนใจทำให้โอกาสที่แบบจำลองจะทำนายกลุ่มข้อมูลจำนวนน้อยให้ถูกต้องมีโอกาสมากขึ้น[9]

ตารางที่ 4 ผลลัพธ์การทำนายแบบจำลองที่สร้างด้วยอัลกอริทึม Logistic Regression ร่วมกับเทคนิค SMOTE

Logistic Regression									
Imbalanced Type	Threshold	TN	FN	TP	FP	Accuracy	Recall	ROC	Balance Accuracy
SMOTE	0.1	857	26	24	115	0.86	0.48	0.75	0.68
	0.2	944	45	5	28	0.93	0.10	0.75	0.54
	0.3	965	46	4	7	0.95	0.08	0.75	0.54
	0.4	970	50	0	2	0.95	0.00	0.75	0.50
	0.5	972	50	0	0	0.95	0.00	0.75	0.50

จากตารางที่ 4 แบบจำลองที่สร้างด้วยอัลกอริทึม Logistic Regression ร่วมกับเทคนิค SMOTE ทดลองโดยการปรับค่า Threshold โดยมีค่ามาตรฐานอยู่ที่ 0.5 จะมีการทำนายกลุ่มผู้ป่วยปกติอยู่ที่ 972 ตัวอย่าง และไม่สามารถทำนายผู้ป่วยโรคหลอดเลือดสมองได้เลย และมีค่าความแม่นยำสมดุลที่ 0.50 เมื่อทำการปรับ Threshold ไปจนถึง 0.1 จะเห็นว่าผลของการจำแนกของแบบจำลองให้ผลลัพธ์ในการทำนายที่ไม่เท่ากัน จุดที่ให้ค่าตัวประสิทธิภาพของแบบจำลองสูงที่สุดอยู่ที่ Threshold 0.1 มีค่าความแม่นยำสมดุลที่อยู่ 0.68 จากผลการทำนายถึงแม้ว่าจะทำนายผลของกลุ่มข้อมูลที่เราไม่สนใจผิดเพิ่มมากขึ้น แต่จะได้ผลของการทำนายกลุ่มข้อมูลที่เราสนใจที่ถูกต้องสูงขึ้น ส่งผลให้การทำนายกลุ่มข้อมูลที่มีจำนวนน้อยได้ดีขึ้น ดังตารางตัวอย่างตารางที่ 9 Threshold ที่ 0.1 ให้ผลลัพธ์ของการทำนายกลุ่มผู้ป่วยปกติ 857 ตัวอย่าง และผู้ป่วยโรคหลอดเลือดสมอง 24 ตัวอย่าง ทำให้มีค่าความแม่นยำสมดุลเพิ่มขึ้นอยู่ที่ 0.68 ซึ่งมากกว่าจุด Threshold 0.50 ที่ให้ความแม่นยำสมดุลอยู่ที่ 0.5 เท่านั้น ซึ่งเป็นผลมาจากการที่จุดที่ Threshold สามารถทำนายกลุ่มข้อมูลจำนวนน้อยถูกต้องได้มากขึ้น

สรุปผลการวิจัย

งานวิจัยนี้เป็นการศึกษาการสร้างแบบจำลองเพื่อจำแนกโรคหลอดเลือด โดยใช้อัลกอริทึมพื้นฐานและซับซ้อนร่วมกับการใช้เทคนิคจัดการชุดข้อมูลไม่สมดุลเพื่อหาแบบจำลองที่ให้ค่าความแม่นยำสมดุลสูงที่สุด ซึ่งผลลัพธ์ที่ได้ต้องสอดคล้องกับสมมติฐานที่ได้ตั้งไว้ โดยสามารถสรุปผลการวิจัยได้ดังนี้

ความเหมาะสมในการเลือกใช้ตัววัดประสิทธิภาพของแบบจำลอง ควรที่จะสะท้อนถึงการทำนายในทุกกลุ่มประเภทข้อมูล หากเลือกใช้ตัวประเมินประสิทธิภาพที่ไม่เหมาะสมจะทำให้ไม่สามารถชี้วัดประสิทธิภาพที่แท้จริงของแบบจำลองได้ ซึ่งในงานวิจัยนี้ได้มีการเลือกใช้ความแม่นยำสมดุลเป็นหลัก

แบบจำลองที่ซับซ้อนมีความสามารถในการจำแนกได้ดีกว่าแบบจำลองพื้นฐาน เป็นสิ่งที่ไม่เสมอไป ในงานวิจัยนี้ได้มีการศึกษาโดยสร้างแบบจำลองที่มาจากอัลกอริทึมที่มีความซับซ้อน และอัลกอริทึมพื้นฐาน

การปรับ Threshold สามารถช่วยให้แบบจำลองสามารถที่จะทำนายกลุ่มข้อมูลจำนวนน้อยได้เพิ่มมากขึ้น เพราะการที่ค่ามาตรฐานอยู่ที่ 0.5 จะทำให้การทำนายของแบบจำลองจะให้ความสำคัญกับกลุ่มข้อมูลจำนวนมาก ซึ่งในบางจุดของ Threshold จะไม่มีการทำนายกลุ่มข้อมูลจำนวนน้อยเลย เพื่อให้แบบจำลองสามารถทำนายกลุ่มข้อมูลจำนวนน้อยได้เพิ่มมากขึ้น จึงควรปรับค่า Threshold ให้น้อยลงเพื่อเพิ่มโอกาสที่จะทำนายกลุ่มข้อมูลจำนวนน้อยเพิ่มมากขึ้น

ดังนั้นการเลือกใช้เทคนิคการสร้างแบบจำลอง หรือการใช้ตัววัดประสิทธิภาพของแบบจำลอง ควรคำนึงถึงลักษณะของชุดข้อมูลที่นำมาใช้ในการสร้างแบบจำลอง หากเลือกใช้ไม่เหมาะสมจะทำให้แบบจำลองที่ได้มีประสิทธิภาพที่ไม่เพียงพอต่อการนำไปใช้งาน

กิตติกรรมประกาศ

การจัดทำวิจัยได้รับการสนับสนุนจากบัณฑิตวิทยาลัย มหาวิทยาลัยศรีนครินทรวิโรฒ ในการนำเสนอผลงานวิจัย ผู้วิจัยจึงขอขอบคุณมา ณ ที่นี้

เอกสารอ้างอิง

[1] ป. จิตตนูนท์, "ความรู้โรคหลอดเลือดสมองและพฤติกรรมป้องกันของกลุ่มเสี่ยงโรคหลอดเลือดสมอง : กรณีศึกษาตำบลห้วยนาง จังหวัดตรัง," Songklanagarind Journal of Nursing, vol. 41, pp. 13-25, 2021.

[2] A. Singh, "KNN algorithm: Introduction to K-Nearest Neighbors Algorithm for Regression," 2023. [Online]. Available: https://www.analyticsvidhya.com/blog/2018/08/k-nearest-neighbor-introduction-regression-python/.

[3] J. Czakon. "F1 Score vs ROC AUC vs Accuracy vs PR AUC: Which Evaluation Metric Should You Choose?" https://neptune.ai/blog/f1-score-accuracy-roc-auc-pr-auc (accessed.

[4] S. Dong, A. Khattak, I. Ullah, J. Zhou, and A. Hussain, "Predicting and Analyzing Road Traffic Injury Severity Using Boosting-Based Ensemble Learning Models with SHAPley Additive exPlanations," Int J Environ Res Public Health, vol. 19, no. 5, Mar 2 2022, doi: 10.3390/ijerph19052925.

[5] c. investment. "Adaptive Boosting Algorithm." https://medium.com/cw-quantlab/adaptive-boosting-algorithm-a761f0a0b264 (accessed.

[6] C. Seiffert, T. M. Khoshgoftaar, J. V. Hulse, and A. Napolitano, "Resampling or Reweighting: A Comparison of Boosting Implementations," presented at the 2008 20th IEEE International Conference on Tools with Artificial Intelligence, 2008.

[7] A. R. M and D. T. R. D. Prakash, "A Simple Approach for Selecting the Best Machine Learning Algorithm," International Journal of Scientific & Engineering Research, vol. 12, no. 9, 2021.

[8] J. T. Hancock and T. M. Khoshgoftaar, "CatBoost for big data: an interdisciplinary review," J Big Data, vol. 7, no. 1, p. 94, 2020, doi: 10.1186/s40537-020-00369-8.

[9] Q. Zou, S. Xie, Z. Lin, M. Wu, and Y. Ju, "Finding the Best Classification Threshold in Imbalanced Classification," Big Data Research, vol. 5, pp. 2-8, 2016, doi: 10.1016/j.bdr.2015.12.001.