



การประเมินประสิทธิภาพของเทคนิคการเรียนรู้ของเครื่องสำหรับการจำแนกโรค
ความดันโลหิตสูงจากข้อมูลทางคลินิก

Performance Evaluation of Machine Learning Techniques for
Hypertension Classification Using Clinical Data

ปริญญานิพนธ์

ของ

นายธีระชาติ พูลพัฒน์

โครงงานปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต

สาขาวิชาเทคโนโลยีสารสนเทศ

คณะวิทยาการสารสนเทศ มหาวิทยาลัยมหาสารคาม

ปีการศึกษา 1/2569



การประเมินประสิทธิภาพของเทคนิคการเรียนรู้ของเครื่องสำหรับการจำแนกโรค
ความดันโลหิตสูงจากข้อมูลทางคลินิก

Performance Evaluation of Machine Learning Techniques for
Hypertension Classification Using Clinical Data

ปริญญานิพนธ์

ของ

นายธีระชาติ พูลพัฒน์

โครงการปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต

สาขาวิชาเทคโนโลยีสารสนเทศ

คณะวิทยาการสารสนเทศ มหาวิทยาลัยมหาสารคาม

ปีการศึกษา 1/2569

ชื่อโครงการปริญญานิพนธ์	: การประเมินประสิทธิภาพของเทคนิคการเรียนรู้ของเครื่อง สำหรับการจำแนกโรคความดันโลหิตสูงจากข้อมูลทางคลินิก
ผู้จัดทำปริญญานิพนธ์	: นายธีระชาติ พูลพัฒน์
อาจารย์ที่ปรึกษาปริญญานิพนธ์	: อาจารย์ ดร.อาทิตยาพร โรจรัตน์

บทคัดย่อ

การวิจัยนี้มีวัตถุประสงค์เพื่อประเมินและเปรียบเทียบประสิทธิภาพของเทคนิคการเรียนรู้ของเครื่อง (Machine Learning) และการเรียนรู้เชิงลึก (Deep Learning) สำหรับการจำแนกภาวะความดันโลหิตสูงจากข้อมูลทางคลินิก และวิเคราะห์ปัจจัยเสี่ยงสำคัญผ่านเทคนิค Explainable Artificial Intelligence (XAI)

ใช้ชุดข้อมูล African Hypertension and Cardiovascular Disease Dataset จำนวน 10,000 ระเบียบ หลังจากคัดกรองตัวแปรที่มีความเสี่ยงต่อ Data Leakage ออก 6 ตัวแปร ดำเนินการเปรียบเทียบแบบจำลอง 9 แบบ ได้แก่ XGBoost, Random Forest, LightGBM, CatBoost, MLP, TabNet, FT-Transformer, SAINT และ TabTransformer โดยแบ่งข้อมูลเป็นสัดส่วน 80:20 ด้วย Stratified Sampling ปรับแต่งไฮเปอร์พารามิเตอร์ด้วย Optuna ภายใต้ Stratified 5-Fold Cross-Validation และประเมินผลด้วย Accuracy, Precision, Recall, F1-score และ AUC-ROC พร้อมวิเคราะห์ความแตกต่างทางสถิติด้วย Paired t-test, Friedman Test และ Nemenyi Post-hoc Test

XGBoost ให้ผลการจำแนกที่ดีที่สุดบนชุดทดสอบ โดยมีค่า AUC เท่ากับ 0.9209 ตามด้วย LightGBM (0.9201) และ CatBoost (0.9189) แบบจำลองส่วนใหญ่มี AUC สูงกว่า 0.90 แสดงศักยภาพในการทำนายสูง อย่างไรก็ตาม TabTransformer มีประสิทธิภาพต่ำกว่าแบบจำลองอื่นอย่างมีนัยสำคัญ ($p < 0.05$) ผลการวิเคราะห์ XAI ด้วย SHAP และ LIME ชี้ให้เห็นว่า ระดับครีเอตินิน, LDL Cholesterol, คอเลสเตอรอลรวม, อายุ และพฤติกรรมการบริโภคอาหารเค็ม เป็นปัจจัยเสี่ยงสำคัญที่สุด

คำสำคัญ: การเรียนรู้ของเครื่อง; ภาวะความดันโลหิตสูง; ข้อมูลทางคลินิก; XGBoost; ปัญหาประติษฐ์ที่อธิบายได้

Thesis Title : Performance Evaluation of Machine Learning Techniques for
Hypertension Classification Using Clinical Data

Author : Teerachad Phoonphut

Supervisor : Artitayaporn Rojarath, Ph.D.

Abstract

This study aims to evaluate and compare the performance of Machine Learning and Deep Learning techniques for hypertension classification using clinical data. and identifies important risk factors through Explainable Artificial Intelligence (XAI) techniques.

The African Hypertension and Cardiovascular Disease Dataset containing 10,000 records was employed. Six variables associated with potential data leakage were removed during preprocessing. Nine models, including XGBoost, Random Forest, LightGBM, CatBoost, Multi-Layer Perceptron (MLP), TabNet, FT-Transformer, SAINT, and TabTransformer, were evaluated. The dataset was divided into training and testing sets using an 80:20 ratio with stratified sampling. Hyperparameter optimization was performed using Optuna with Stratified 5-Fold Cross-Validation. Model performance was assessed using Accuracy, Precision, Recall, F1-score, and AUC-ROC. Statistical significance was examined using the Paired t-test, Friedman Test, and Nemenyi Post-hoc Test.

The experimental results showed that XGBoost achieved the highest performance on the test dataset with an AUC of 0.9209, followed by LightGBM (0.9201) and CatBoost (0.9189). Most models achieved AUC values above 0.90, demonstrating strong predictive capability. However, TabTransformer showed significantly lower performance compared with other models ($p < 0.05$). The XAI analysis using SHAP and LIME identified creatinine level, LDL cholesterol, total cholesterol, age, and high-salt dietary behavior as the most influential risk factors.

Keywords: Machine Learning; Hypertension; Clinical Data; XGBoost; Explainable Artificial Intelligence (XAI)

กิตติกรรมประกาศ

วิจัยฉบับนี้สำเร็จลุล่วงได้ด้วยดีด้วยความกรุณาอย่างยิ่งจาก อาจารย์ ดร.อาทิตย์พร โรจรัตน์ อาจารย์ที่ปรึกษางานวิจัย ที่ได้ให้คำแนะนำ แนวคิด และตรวจแก้ไขข้อบกพร่องต่าง ๆ ด้วยความเอาใจใส่และเกื้อหนุนมาโดยตลอด ผู้วิจัยขอกราบขอบพระคุณเป็นอย่างสูงไว้ ณ ที่นี้ ขอขอบคุณคณาจารย์ภาควิชาเทคโนโลยีสารสนเทศ คณะวิทยาการสารสนเทศ มหาวิทยาลัยมหาสารคาม ทุกท่าน ที่ได้ประสิทธิ์ประสาทวิชาความรู้อันเป็นรากฐานสำคัญในการทำวิจัยครั้งนี้ รวมทั้งขอขอบคุณเจ้าของชุดข้อมูล African Hypertension and Cardiovascular Disease Dataset ที่ได้เอื้อเฟื้อข้อมูลเพื่อการศึกษาและพัฒนาเนื้อหาทางวิชาการให้สมบูรณ์ยิ่งขึ้นสุดท้ายนี้ ขอกราบขอบพระคุณบิดา มารดา กัลยาณมิตร ตลอดจนเพื่อน ๆ และผู้มีส่วนเกี่ยวข้องทุกท่าน ที่คอยเป็นที่ปรึกษา ให้กำลังใจ และสนับสนุนในทุก ๆ ด้าน เสมอมา จนกระทั่งรายงานการวิจัยฉบับนี้สำเร็จลุล่วงด้วยดี

ธีระชาติ พูลพัฒน์

สารบัญ

หน้า

บทคัดย่อ.....	ก
Abstract.....	ข
กิตติกรรมประกาศ.....	ค
สารบัญ.....	ง
สารบัญตาราง.....	ฉ
สารบัญภาพ.....	ช
บทที่ 1 บทนำ.....	1
1.1 หลักการและเหตุผล	1
1.2 วัตถุประสงค์	2
1.3 ขอบเขตของการวิจัย	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ.....	3
1.5 อุปกรณ์และเครื่องมือที่ใช้	3
1.6 แผนการดำเนินงานตลอดโครงการ.....	8
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง.....	9
2.1 ทฤษฎีที่เกี่ยวข้อง	9
2.2 งานวิจัยที่เกี่ยวข้อง.....	30
บทที่ 3 แผนการดำเนินงานวิจัย.....	34
3.1 กรอบแนวคิดการวิจัยและสภาพแวดล้อมการทดลอง	34
3.2 การรวบรวมชุดข้อมูลที่ใช้ในการวิจัย	36
3.3 การวิเคราะห์ข้อมูลเชิงสำรวจ.....	36
3.3.2 สำนักรวคุณภาพข้อมูล.....	37
3.2.3 การวิเคราะห์ตัวแปรเดียว.....	38
3.2.4 การวิเคราะห์หลายตัวแปร.....	39
3.2.5 ความไม่สมดุลทางข้อมูล.....	40
3.3 การเตรียมข้อมูล	41
3.3.1 การลบตัวแปรที่ไม่ใช่ปัจจัยเสี่ยง.....	41

3.3.2 การแบ่งชุดข้อมูล.....	42
3.3.3 การเข้ารหัสตัวแปร.....	44
3.3.4 การปรับมาตรฐานของตัวแปร.....	45
3.4 การพัฒนาตัวแบบ.....	46
3.4.1 การพัฒนากลุ่มตัวแบบทรีแบบรวม.....	46
3.4.2 การพัฒนาตัวแบบกลุ่มการเรียนรู้เชิงลึก.....	47
3.4.3 การพัฒนาตัวแบบกลุ่มทรานส์ฟอร์เมอร์.....	48
3.4.4 กระบวนการปรับค่าพารามิเตอร์.....	49
3.5 กลยุทธ์การตรวจสอบไขว้	55
3.6 การวัดประสิทธิภาพของตัวแบบ	56
3.7 วิเคราะห์ความสามารถในการอธิบายผล	56
บทที่ 4 ผลการทดลอง.....	57
4.1 ผลการเตรียมข้อมูล	57
4.2 ผลการปรับแต่งไฮเปอร์พารามิเตอร์และการตรวจสอบไขว้	59
4.3 ผลการประเมินแบบจำลองบนชุดข้อมูลทดสอบ	59
4.4 ผลการทดสอบนัยสำคัญทางสถิติของความแตกต่างด้านประสิทธิภาพ	60
4.5 ผลการวิเคราะห์ความสำคัญของตัวแปร	65
4.6 ผลการวิเคราะห์ความสอดคล้องของค่าความน่าจะเป็น	66
4.8 สรุปผลการวิจัยในภาพรวม.....	67
บทที่ 5 สรุปผลการวิจัย อภิปรายผล และข้อเสนอแนะ.....	68
5.1 สรุปผลการวิจัย	68
5.2 อภิปรายผล	70
5.3 ข้อจำกัดของงานวิจัย	71
5.4 ข้อเสนอแนะ	72
เอกสารอ้างอิง.....	74
ภาคผนวก.....	78

สารบัญตาราง

	หน้า
ตารางที่ 1.1 แผนการดำเนินงานตลอดโครงการ	8
ตารางที่ 2.1 เกณฑ์ระดับโรคความดันโลหิตสูง	9
ตารางที่ 2.2 ตารางสรุปงานวิจัยที่เกี่ยวข้อง	32
ตารางที่ 3.1 ขั้นตอนการดำเนินงานทดลอง	34
ตารางที่ 3.2 สภาพแวดล้อมด้านซอฟต์แวร์	35
ตารางที่ 3.3 สภาพแวดล้อมด้านฮาร์ดแวร์	35
ตารางที่ 3.4 การกำหนดค่าการตรวจสอบไขว้.....	36
ตารางที่ 3.5 รายละเอียดของตัวแปรทั้งหมดในชุดข้อมูล	36
ตารางที่ 3.6 ตารางสำรวจค่าที่อยู่นอกช่วงปกติ	37
ตารางที่ 3.7 ผลการเปรียบเทียบค่า Accuracy ของตัวแบบภายหลังการใช้เทคนิคจัดการข้อมูลไม่สมดุล	40
ตารางที่ 3.8 ตารางข้อมูลหลังจากที่ตัดข้อมูลที่มีข้อมูลรั่วไหลและไม่เกี่ยวข้องออก	42
ตารางที่ 3.9 ตารางข้อมูลที่แบ่งในแต่ละส่วน.....	43
ตารางที่ 3.10 ขอบเขตการค้นหาไฮเปอร์พารามิเตอร์ของกลุ่มตัวแบบทรีแบบรวม.....	47
ตารางที่ 3.11 ขอบเขตการค้นหาไฮเปอร์พารามิเตอร์ของกลุ่มตัวแบบการเรียนรู้เชิงลึก	48
ตารางที่ 3.12 ขอบเขตการค้นหาไฮเปอร์พารามิเตอร์ของกลุ่มตัวแบบกลุ่มทรานส์ฟอร์เมอร์.....	48
ตารางที่ 3.13 การปรับแต่งพารามิเตอร์ของ XGBoost ด้วย Optuna.....	49
ตารางที่ 3.14 การปรับแต่งพารามิเตอร์ของ Random Forest ด้วย Optuna	50
ตารางที่ 3.15 การปรับแต่งพารามิเตอร์ของ LightGBM ด้วย Optuna	50
ตารางที่ 3.16 การปรับแต่งพารามิเตอร์ของ CatBoost ด้วย Optuna.....	51
ตารางที่ 3.17 การปรับแต่งพารามิเตอร์ของ MLP ด้วย Optuna	52
ตารางที่ 3.18 การปรับแต่งพารามิเตอร์ของ TabNet ด้วย Optuna	52
ตารางที่ 3.19 การปรับแต่งพารามิเตอร์ของ FT-Transformer ด้วย Optuna.....	53
ตารางที่ 3.20 การปรับแต่งพารามิเตอร์ของ SAINT ด้วย Optuna.....	54
ตารางที่ 3.21 การปรับแต่งพารามิเตอร์ของ TabTransformer ด้วย Optuna.....	55
ตารางที่ 4.1 ลักษณะทั่วไปของชุดข้อมูล.....	57
ตารางที่ 4.2 การแบ่งชุดข้อมูล	58
ตารางที่ 4.3 ผลการตรวจสอบไขว้ของแบบจำลองพื้นฐานทั้งหมด.....	59
ตารางที่ 4.4 ผลการประเมินประสิทธิภาพของแบบจำลองบนชุดข้อมูลทดสอบ	60
ตารางที่ 4.5 อันดับเฉลี่ยของแต่ละแบบจำลองจากการตรวจสอบไขว้	61
ตารางที่ 4.6 ผลการทดสอบ Friedman Test.....	61

ตารางที่ 4.7 ผลการเปรียบเทียบกลุ่มแบบจำลองต้นไม้ของค่า AUC	62
ตารางที่ 4.8 ผลการเปรียบเทียบกลุ่มแบบจำลองต้นไม้ของค่า Accuracy	63
ตารางที่ 4.9 ผลการเปรียบเทียบกลุ่มการเรียนรู้เชิงลึกของค่า AUC	64
ตารางที่ 4.10 ผลการเปรียบเทียบกลุ่มการเรียนรู้เชิงลึกของค่า Accuracy	64
ตารางที่ 4.11 ผลการเปรียบเทียบกลุ่มทรานส์ฟอร์เมอร์ของค่า AUC	64
ตารางที่ 4.12 ผลการเปรียบเทียบกลุ่มทรานส์ฟอร์เมอร์ของค่า Accuracy	65
ตารางที่ 4.13 ค่าความสำคัญของตัวแปร 10 อันดับแรก	65
ตารางที่ 4.14 ผลการวิเคราะห์ Brier Score และ Mean Calibration Gap ของแบบจำลองทั้งหมด	66

สารบัญภาพ

	หน้า
ภาพประกอบที่ 1.1 สัญลักษณ์ของ Python.....	4
ภาพประกอบที่ 1.2 สัญลักษณ์ของ Weka	4
ภาพประกอบที่ 1.3 สัญลักษณ์ของ Visual Studio Code	5
ภาพประกอบที่ 1.4 สัญลักษณ์ของ Google Colab	5
ภาพประกอบที่ 1.5 สัญลักษณ์ของ Microsoft Excel	5
ภาพประกอบที่ 1.6 สัญลักษณ์ของ Pandas.....	6
ภาพประกอบที่ 1.7 สัญลักษณ์ของ NumPy.....	6
ภาพประกอบที่ 1.8 สัญลักษณ์ของ scikit-learn	6
ภาพประกอบที่ 1.9 สัญลักษณ์ของ PyTorch	6
ภาพประกอบที่ 1.10 สัญลักษณ์ของ XGBoost	7
ภาพประกอบที่ 1.11 สัญลักษณ์ของ TensorFlow	7
ภาพประกอบที่ 1.12 สัญลักษณ์ของ Optuna	7
ภาพประกอบที่ 1.13 สัญลักษณ์ของ SHAP	7
ภาพประกอบที่ 1.14 สัญลักษณ์ของ LIME	8
ภาพประกอบที่ 1.15 สัญลักษณ์ของ Matplotlib.....	8
ภาพประกอบที่ 2.1 ภาพการทำงานของการเรียนรู้ของเครื่อง.....	10
ภาพประกอบที่ 2.2 กระบวนการจำแนกประเภทแบบการเรียนรู้แบบมีผู้สอน	11
ภาพประกอบที่ 2.3 ภาพการทำงานของเอ็กซ์จีบูสต์	11
ภาพประกอบที่ 2.4 ภาพการทำงานของการเรียนรู้เชิงลึก	14
ภาพประกอบที่ 2.5 ภาพการทำงานของตัวแบบแท็บเน็ต	15
ภาพประกอบที่ 2.6 ภาพการเลือกการนำเข้าคุณลักษณะของแท็บเน็ต	17
ภาพประกอบที่ 2.7 ภาพการแสดงผลสถาปัตยกรรมภายในของแท็บเน็ต	17
ภาพประกอบที่ 2.8 ภาพการเลือกคุณลักษณะและการจำแนกประเภท	18
ภาพประกอบที่ 2.9 ภาพการ Feature Tokenizer transforms.....	18
ภาพประกอบที่ 2.10 ภาพโครงสร้างของแท็บทรานฟอเมอร์.....	22
ภาพประกอบที่ 2.11 ภาพการวัดประสิทธิภาพแบบ AUC-ROC Curve.....	28
ภาพประกอบที่ 3.1 ภาพรวมของกระบวนการดำเนินงานวิจัย	34
ภาพประกอบที่ 3.2 ฮิสโตแกรมแสดงการกระจายของตัวแปรเชิงตัวเลข	38
ภาพประกอบที่ 3.3 Correlation Matrix แสดงความสัมพันธ์ระหว่างตัวแปรเชิงตัวเลขด้วยค่าสหสัมพันธ์ ...	39
ภาพประกอบที่ 3.4 ภาพเปรียบเทียบจำนวนในคลาสผลลัพธ์	40

ภาพประกอบที่ 3.5 ภาพแสดงการแบ่งชุดข้อมูลต่อจำนวนข้อมูลทั้งหมด	43
ภาพประกอบที่ 3.6 การกระจายของตัวแปรเป้าหมายในชุดข้อมูลฝึกสอน	43
ภาพประกอบที่ 3.7 การกระจายของตัวแปรเป้าหมายในชุดข้อมูลทดสอบ.....	44
ภาพประกอบที่ 3.8 การเปรียบเทียบการกระจายของข้อมูลก่อนและหลังการปรับมาตรฐาน	45
ภาพประกอบที่ 4.1 ภาพ t-SNE แสดงการกระจายของข้อมูลในชุดฝึกสอน	58

บทที่ 1

บทนำ

1.1 หลักการและเหตุผล

โรคความดันโลหิตสูง (Hypertension) ถือเป็นปัญหาสาธารณสุขที่สำคัญทั้งในประเทศไทยและทั่วโลก เนื่องจากเป็นภาวะที่พบได้บ่อย มีลักษณะเรื้อรัง และเกี่ยวข้องโดยตรงกับภาวะโรคหัวใจและหลอดเลือดในระดับประชากร ข้อมูลจากการสำรวจสุขภาพประชาชนไทยโดยการตรวจร่างกาย ครั้งที่ 7 พ.ศ. 2567–2568 ซึ่งเก็บข้อมูลภาคสนามระหว่างเดือนสิงหาคม 2567 ถึงเมษายน 2568 พบว่าประชาชนไทยอายุ 15 ปีขึ้นไปมีความชุกของโรคความดันโลหิตสูงร้อยละ 29.5 เพิ่มขึ้นจากร้อยละ 25.4 ในการสำรวจปี 2563 สะท้อนว่าประมาณ 3 ใน 10 คนกำลังเผชิญภาวะดังกล่าว [1] สถานการณ์นี้สอดคล้องกับแนวโน้มระดับโลก โดยองค์การอนามัยโลกรายงานว่าจำนวนผู้ใหญ่ที่มีภาวะความดันโลหิตสูงทั่วโลกเพิ่มขึ้นจาก 650 ล้านคนในปี 1990 เป็นประมาณ 1.4 พันล้านคนในปี 2024 ผู้ป่วยจำนวนมากไม่ทราบว่าตนเองมีโรค เนื่องจากภาวะนี้มักไม่แสดงอาการเด่นชัดในระยะแรก การตรวจวัดความดันโลหิตจึงเป็นวิธีสำคัญในการค้นหาผู้ป่วยก่อนเกิดภาวะแทรกซ้อนรุนแรง อีกทั้งยังเป็นปัจจัยเสี่ยงสำคัญของโรคหัวใจและหลอดเลือด ซึ่งองค์การอนามัยโลกระบุว่าเป็นสาเหตุการเสียชีวิตอันดับหนึ่งของโลก โดยมีผู้เสียชีวิตประมาณ 19.8 ล้านคนในปี 2022 [2] ด้วยเหตุนี้โรคความดันโลหิตสูงจึงมักถูกกล่าวถึงในเชิงสาธารณสุขว่าเป็น "เพชฌฆาตเงียบ" เนื่องจากผู้ป่วยมักรับรู้ความผิดปกติเมื่อโรคลุกลามไปสู่ภาวะหลอดเลือดสมอง โรคหัวใจ หรือไตเสื่อมแล้ว [3]

ดังนั้น การพัฒนาเครื่องมือคัดกรองและทำนายเชิงรุกเพื่อสนับสนุนการวินิจฉัยและบริหารจัดการโรคความดันโลหิตสูงจึงมีความสำคัญอย่างยิ่ง งานวิจัยที่ผ่านมาแสดงให้เห็นว่าเทคนิคการเรียนรู้ของเครื่องมีศักยภาพในการทำนายหรือจำแนกภาวะความดันโลหิตสูงได้อย่างมีประสิทธิภาพ โมเดลเหล่านี้แบ่งออกเป็นสามกลุ่มหลัก ได้แก่ กลุ่มต้นไม้ตัดสินใจและการเรียนรู้แบบรวมกลุ่ม เช่น XGBoost, Random Forest, LightGBM และ CatBoost ซึ่งได้รับความนิยมเนื่องจากมีความแม่นยำสูงและสามารถจัดการกับข้อมูลไม่สมดุลได้ดี ตัวอย่างเช่น การศึกษาที่เปรียบเทียบโมเดลเหล่านี้บนชุดข้อมูลคลินิกโรคความดันโลหิตสูง พบว่า CatBoost ให้ผลลัพธ์ที่ดีที่สุด [4] อีกกลุ่มคือการเรียนรู้เชิงลึกและ แบบจำลองที่อิงกลไกความใส่ใจเช่น Multi-Layer Perceptron (MLP), TabNet ที่ใช้กลไกความใส่ใจแบบลำดับ รวมถึงสถาปัตยกรรมทรานส์ฟอร์เมอร์สำหรับข้อมูลตาราง ได้แก่ FT-Transformer, SAINT และ TabTransformer ที่ใช้กลไกความใส่ใจในตนเองในการเรียนรู้ปฏิสัมพันธ์ระหว่างตัวแปร งานวิจัยล่าสุดพบว่าโมเดลกลุ่มนี้ให้ผลดีบนชุดข้อมูลขนาดใหญ่และมีโครงสร้างชัดเจน แต่ประสิทธิภาพจะลดลงเมื่อข้อมูลมีความซับซ้อนและไม่สมดุล ในขณะที่โมเดลดั้งเดิมอย่าง XGBoost ยังคงสามารถแข่งขันได้ในหลายกรณี สะท้อนว่าการเลือกโมเดลควรพิจารณาจากลักษณะข้อมูลเป็นสำคัญ และไม่มีโมเดลใดที่เหนือกว่าอย่างสมบูรณ์ [5]

อย่างไรก็ตาม เมื่อพิจารณาเชิงระเบียบวิธีอย่างละเอียด ยังพบข้อจำกัดที่สำคัญในองค์ความรู้ปัจจุบันหลายประการ ประการแรก งานวิจัยจำนวนมากไม่น้อยยังคงใช้ตัวแปรที่มีความเสี่ยงต่อการก่อให้เกิดข้อมูลรั่วไหล

เช่น ตัวแปรที่เป็นผลลัพธ์ทางคลินิก ซึ่งอาจทำให้แบบจำลองมีสมรรถนะที่ดีเกินจริงและไม่สะท้อนความสามารถในการทำนายที่แท้จริงเมื่อนำไปใช้กับข้อมูลใหม่ ประการที่สอง การประเมินสมรรถนะแบบจำลองในหลายการศึกษาอาจอาศัยการแบ่งชุดข้อมูลเพียงครั้งเดียว โดยไม่ได้ใช้กรอบการทดสอบแบบไขว้ ทำให้ผลลัพธ์อาจมีความคลาดเคลื่อนสูง ประการที่สาม แม้แบบจำลองกลุ่มการเรียนรู้เชิงลึกและ การเพิ่มประสิทธิภาพแบบไล่ระดับจะให้ความแม่นยำสูง แต่มักถูกมองว่าตีความยาก

การศึกษานี้จึงมีกระบวนการพัฒนาแบบจำลองให้มีความรัดกุมมากขึ้น โดยตัดตัวแปรที่มีความเสี่ยงต่อการก่อให้เกิดข้อมูลรั่วไหล ประเมินแบบจำลองฐานหลายชนิด ครอบคลุมทั้งกลุ่มต้นไม้ตัดสินใจ (XGBoost, Random Forest, LightGBM, CatBoost) กลุ่มการเรียนรู้เชิงลึก (MLP, TabNet) และกลุ่มของทรานส์ฟอร์เมอร์ (FT-Transformer, SAINT, TabTransformer) ภายใต้กรอบการตรวจสอบแบบไขว้ และใช้วิธีอธิบายแบบจำลอง (SHAP และ LIME) ควบคู่กัน เพื่อให้ผลการศึกษาเชื่อถือได้ โปร่งใส และตีความได้มากขึ้นอันจะเป็นประโยชน์ต่อการพัฒนาระบบสนับสนุนการตัดสินใจทางคลินิกที่มีความแม่นยำและน่าเชื่อถือมากขึ้น

1.2 วัตถุประสงค์

1.2.1 พัฒนาและประเมินแบบจำลองการทำนายภาวะความดันโลหิตสูง โดยตัดตัวแปรที่มีความเสี่ยงต่อการเกิดข้อมูลรั่วไหล (Data Leakage) เปรียบเทียบประสิทธิภาพของแบบจำลองหลายกลุ่ม ได้แก่ XGBoost, Random Forest, LightGBM, CatBoost, MLP, TabNet, FT-Transformer, SAINT และ TabTransformer ภายใต้กรอบ Cross-Validation เพื่อระบุแบบจำลองเดียวที่มีสมรรถนะสูงที่สุด

1.2.2 วิเคราะห์และอธิบายผลการทำนายของแบบจำลอง โดยใช้เทคนิค SHAP และ LIME เพื่อเพิ่มความโปร่งใส ความน่าเชื่อถือ และความสามารถในการตีความ พร้อมอธิบายบทบาทของตัวแปรทำนายที่สำคัญในเชิงคลินิก

1.3 ขอบเขตของการวิจัย

1.3.1 แหล่งข้อมูล

1.3.1.1 ใช้ชุดข้อมูลสาธารณะ African Hypertension & CVD Dataset (2024) Version 1.0 จากแพลตฟอร์ม HuggingFace โดยเลือกชุดข้อมูล extra_large_10000 จำนวน 10,000 ระเบียบ ซึ่งเป็นข้อมูลสังเคราะห์ภายใต้สัญญาอนุญาต CC BY 4.0 และไม่มีข้อมูลที่สามารถระบุตัวตนของบุคคลได้

1.3.1.2 ข้อมูลมีลักษณะเป็นข้อมูลแบบตาราง ประกอบด้วยตัวแปรด้านประชากรศาสตร์ พฤติกรรมสุขภาพ และตัวชี้วัดทางคลินิก โดยมีตัวแปรเป้าหมาย คือ สถานะความดันโลหิตสูง แบ่งเป็น 2 กลุ่ม ได้แก่ Hypertensive และ Normal

1.3.2 โมเดลที่ใช้ในการศึกษา

1. XGBoost
2. Random forest
3. LightGBM

4. CatBoost
5. TabNet
6. MLP
7. FT-Transformer
8. SAINT
9. TabTransformer

1.3.3 เทคนิคและการประเมินประสิทธิภาพ

1.3.3.1 ทำการแบ่งข้อมูลออกเป็น ชุดข้อมูลฝึกสอนร้อยละ 80 และชุดข้อมูลทดสอบร้อยละ 20 โดยใช้วิธีการแบ่งข้อมูลแบบคงสัดส่วนของคลาสและทำการฝึกแบบไขว้

1.3.3.2 ประเมินประสิทธิภาพของตัวแบบโดยใช้ตัวชี้วัด ได้แก่ Accuracy, AUC-ROC, Sensitivity, Specificity, Precision, Recall และ F1-score

1.3.3.3 เทคนิคการอธิบายผลลัพธ์ ใช้เทคนิค Explainable Artificial Intelligence (XAI) เพื่ออธิบายผลลัพธ์ของตัวแบบ ได้แก่ SHAP LIME

1.3.4 ข้อจำกัดของการวิจัย

1.3.4.1 การวิจัยนี้มุ่งเน้นการพยากรณ์โรคความดันโลหิตสูงจากข้อมูลแบบตารางเท่านั้น โดยไม่ได้รวมข้อมูลประเภทอื่น เช่น ภาพทางการแพทย์ สัญญาณชีพแบบต่อเนื่อง หรือข้อความจากเวชระเบียน

1.3.4.2 ระบบที่พัฒนาขึ้นมีวัตถุประสงค์เพื่อสนับสนุนการวิเคราะห์ข้อมูลทางการแพทย์เท่านั้น และไม่สามารถใช้แทนการวินิจฉัยของแพทย์ได้

1.3.4.3 การศึกษานี้เป็นการวิจัยเชิงเปรียบเทียบ โดยไม่ได้มีการพัฒนาสถาปัตยกรรมของแบบจำลองใหม่

1.3.4.4 เนื่องจากชุดข้อมูลที่ใช้เป็นข้อมูลสังเคราะห์ ผลลัพธ์ที่ได้จึงสะท้อนศักยภาพเชิงเทคนิคของตัวแบบ และไม่สามารถอ้างอิงแทนข้อมูลผู้ป่วยจริงได้โดยตรง

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1.4.1 ได้องค์ความรู้เชิงเปรียบเทียบเกี่ยวกับประสิทธิภาพของแบบจำลองในการพยากรณ์โรคความดันโลหิตสูง และเป็นแนวทางในการเลือกใช้แบบจำลองที่เหมาะสมสำหรับข้อมูลแบบตารางทางการแพทย์เกณฑ์การประเมินประสิทธิภาพของตัวแบบอย่างเป็นมาตรฐานและเป็นธรรม

1.4.2 ได้แนวทางการประยุกต์ใช้เทคนิค Explainable Artificial Intelligence (XAI) ได้แก่ SHAP และ LIME เพื่ออธิบายผลการทำนายและปัจจัยเสี่ยง พร้อมทั้งเป็นแนวทางในการประเมินประสิทธิภาพของแบบจำลองอย่างเป็นมาตรฐาน

1.5 อุปกรณ์และเครื่องมือที่ใช้

1.5.1 อุปกรณ์ที่ใช้

1.5.1.1 แล็ปท็อป

1. CPU Intel Core i5
2. RAM 8 GB
3. ระบบปฏิบัติการ Windows 11

1.5.1.2 อุปกรณ์จัดเก็บข้อมูล

1. ใช้อุปกรณ์จัดเก็บข้อมูลแบบ Solid State Drive

1.5.1.3 ระบบเครือข่าย

1. ใช้สำหรับดาวน์โหลดชุดข้อมูล
2. ใช้สำหรับการทบทวนวรรณกรรม
3. ใช้เพื่อเข้าถึงสมุดบันทึกออนไลน์สำหรับเป็นสภาพแวดล้อมในการทดสอบ
4. และอื่น ๆ ที่เกี่ยวข้อง

1.5.2 เครื่องมือที่ใช้

1.5.2.1 Python



ภาพประกอบที่ 1.1 สัญลักษณ์ของ Python

ไพธอน [6] เป็นภาษาโปรแกรมระดับสูงที่ถูกออกแบบมาให้มีโครงสร้างเรียบง่าย อ่านและเขียนได้เข้าใจง่ายเผยแพร่ครั้งแรกในปี ค.ศ. 1991 โดยมีแนวคิดหลักคือการลดความซับซ้อนของโค้ดและเพิ่มประสิทธิภาพในการพัฒนาโปรแกรม ไพธอนเหมาะสำหรับทั้งผู้เริ่มต้นและนักพัฒนาที่มีประสบการณ์ ปัจจุบันไพธอน เป็นซอฟต์แวร์แบบโอเพนซอร์สและได้รับการพัฒนาอย่างต่อเนื่องโดยชุมชนนักพัฒนาทั่วโลกโดยสัญลักษณ์ของไพธอนแสดงดังภาพประกอบที่ 1.1

1.5.2.2 Weka



ภาพประกอบที่ 1.2 สัญลักษณ์ของ Weka

เวกา [7] เป็นซอฟต์แวร์สำหรับการวิเคราะห์ข้อมูลและการเรียนรู้ของเครื่องโดยออกแบบมาเพื่อช่วยในการทำเหมืองข้อมูลและการสร้างตัวแบบการเรียนรู้ของเครื่องโดยไม่จำเป็นต้องเขียนโปรแกรมพัฒนาโดยทีมนักวิจัยจาก มหาวิทยาลัย Waikato ประเทศนิวซีแลนด์โดยสัญลักษณ์ของ Weka แสดงดังภาพประกอบที่ 1.2 โดย Weka ได้รับความนิยมอย่างแพร่หลายในการศึกษาและงานวิจัยมีส่วนติดต่อผู้ใช้แบบ

กราฟิกที่ใช้งานง่าย ผู้ใช้สามารถนำเข้าข้อมูล ทดลองใช้อัลกอริทึมต่าง ๆ และเปรียบเทียบผลลัพธ์ได้สะดวก อีกทั้งยังรองรับอัลกอริทึมมาตรฐานจำนวนมาก

1.5.2.3 Visual Studio Code



ภาพประกอบที่ 1.3 สัญลักษณ์ของ Visual Studio Code

Visual Studio Code [8] เป็นโปรแกรมแก้ไขซอร์สโค้ดแบบโอเพนซอร์สที่มีน้ำหนักเบา แต่มีความสามารถสูง เหมาะสำหรับการพัฒนาโปรแกรมหลากหลายภาษา พัฒนาโดยบริษัท Microsoft ซึ่งมุ่งเน้นการสร้างเครื่องมือพัฒนาซอฟต์แวร์ที่ใช้งานง่าย มีประสิทธิภาพได้รับความนิยมอย่างแพร่หลายในหมู่นักพัฒนา นักวิจัย และนักวิทยาการข้อมูล มีอินเทอร์เฟซที่ใช้งานง่าย รองรับการเขียนโค้ดหลายภาษา และมีระบบส่วนเสริมที่หลากหลายโดยสัญลักษณ์ของ Visual Studio Code แสดงให้เห็นดังภาพประกอบที่ 1.3

1.5.2.4 Google Colab



ภาพประกอบที่ 1.4 สัญลักษณ์ของ Google Colab

Google Colab ดังภาพประกอบที่ 1.4 เป็นสัญลักษณ์ของ Google Colab [9] เป็นแพลตฟอร์มสำหรับการเขียนและรันโค้ดภาษาไพธอน บนระบบคลาวด์ ผ่านเว็บเบราว์เซอร์ โดยไม่จำเป็นต้องติดตั้งโปรแกรมหรือกำหนดค่าสภาพแวดล้อมบนเครื่องผู้ใช้พัฒนาบนพื้นฐานของ Jupyter Notebook พัฒนาโดยบริษัท Google ภายใต้แนวคิดการสนับสนุนการศึกษา การวิจัย และการพัฒนาเทคโนโลยีปัญญาประดิษฐ์ เหมาะสำหรับการทดลองสร้างตัวแบบการเรียนรู้ของเครื่องโดยไม่ต้องมีเครื่องคอมพิวเตอร์ประสิทธิภาพสูง

1.5.2.5 Microsoft Excel



ภาพประกอบที่ 1.5 สัญลักษณ์ของ Microsoft Excel

Microsoft Excel [10] เป็นโปรแกรมสเปรดชีต ที่ใช้สำหรับการจัดการ คำนวณ และวิเคราะห์ข้อมูลในรูปแบบตาราง พัฒนาโดยบริษัท Microsoft และเปิดตัวครั้งแรกในปี ค.ศ. 1985 โดยเป็นส่วนหนึ่งของชุดโปรแกรม Microsoft Office ถือเป็นหนึ่งในโปรแกรมจัดการข้อมูลที่ได้รับความนิยมสูงสุดใน

โลก เนื่องจากใช้งานง่าย เหมาะสำหรับผู้ใช้ทุกระดับ และสามารถนำไปประยุกต์ใช้ได้หลากหลาย โดยสัญลักษณ์ของ Microsoft Excel แสดงดังภาพประกอบที่ 1.5

1.5.2.6 Pandas



ภาพประกอบที่ 1.6 สัญลักษณ์ของ Pandas

Pandas [11] ใช้สำหรับการจัดการข้อมูลเชิงตาราง การทำความสะอาดข้อมูล และการเตรียมข้อมูลให้อยู่ในรูปแบบที่เหมาะสมสำหรับการวิเคราะห์และการสร้างแบบจำลอง โดยสัญลักษณ์ของ Pandas แสดงในภาพประกอบที่ 1.6

1.5.2.7 NumPy



ภาพประกอบที่ 1.7 สัญลักษณ์ของ NumPy

NumPy [12] ใช้สำหรับการคำนวณเชิงตัวเลขซึ่งเป็นพื้นฐานสำคัญในการประมวลผลข้อมูลและการคำนวณทางคณิตศาสตร์ โดยสัญลักษณ์ของ NumPy แสดงในภาพประกอบที่ 1.7

1.5.2.8 scikit-learn



ภาพประกอบที่ 1.8 สัญลักษณ์ของ scikit-learn

scikit-learn [13] ใช้สำหรับการเตรียมข้อมูล การแบ่งชุดข้อมูล การคัดเลือกคุณลักษณะ และการประเมินประสิทธิภาพของแบบจำลองการเรียนรู้ของเครื่อง โดยสัญลักษณ์ของ scikit-learn แสดงในภาพประกอบที่ 1.8

1.5.2.9 PyTorch



ภาพประกอบที่ 1.9 สัญลักษณ์ของ PyTorch

PyTorch [14] ใช้สำหรับการพัฒนาและฝึกสอนแบบจำลองการเรียนรู้เชิงลึก โดยสนับสนุนการสร้างโครงสร้างแบบจำลองที่มีความยืดหยุ่นและมีประสิทธิภาพสูง โดยสัญลักษณ์ของ PyTorch แสดงในภาพประกอบที่ 1.9

1.5.2.10 XGBoost



ภาพประกอบที่ 1.10 สัญลักษณ์ของ XGBoost

XGBoost [15] ใช้สำหรับพัฒนาแบบจำลองประเภท Gradient Boosting โดยมีประสิทธิภาพ โดยสัญลักษณ์ของ XGBoost แสดงในภาพประกอบที่ 1.10

1.5.2.11 TensorFlow



ภาพประกอบที่ 1.11 สัญลักษณ์ของ TensorFlow

TensorFlow [16] เป็นไลบรารีสำหรับการพัฒนาแบบจำลองการเรียนรู้เชิงลึก ที่รองรับการคำนวณแบบโครงข่ายประสาทเทียม โดยสัญลักษณ์ของ TensorFlow แสดงในภาพประกอบที่ 1.11

1.5.2.12 Optuna



ภาพประกอบที่ 1.12 สัญลักษณ์ของ Optuna

Optuna [17] เป็นเครื่องมือสำหรับการปรับแต่งค่าพารามิเตอร์ของแบบจำลอง แบบอัตโนมัติ โดยใช้เทคนิคการค้นหาที่มีประสิทธิภาพเพื่อหาชุดค่าพารามิเตอร์ที่เหมาะสมที่สุดสำหรับการเพิ่มประสิทธิภาพของแบบจำลอง โดยสัญลักษณ์ของ Optuna แสดงในภาพประกอบที่ 1.12

1.5.2.13 SHAP



ภาพประกอบที่ 1.13 สัญลักษณ์ของ SHAP

SHAP [18] เป็นเทคนิคในกลุ่ม Explainable Artificial Intelligence ที่ใช้สำหรับอธิบายผลลัพธ์ของแบบจำลอง โดยคำนวณค่าความสำคัญของตัวแปรแต่ละตัวต่อการตัดสินใจของแบบจำลอง ทำให้สามารถตีความผลลัพธ์ของแบบจำลองได้อย่างโปร่งใส โดยสัญลักษณ์ของ SHAP แสดงในภาพประกอบที่ 1.13

[illegible]

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 ทฤษฎีที่เกี่ยวข้อง

2.1.1 โรคความดันโลหิตสูง

โรคความดันโลหิตสูง หมายถึง ภาวะที่ความดันโลหิตในหลอดเลือดแดงสูงกว่าเกณฑ์ปกติอย่างต่อเนื่อง โดยทั่วไปวินิจฉัยเมื่อค่าความดันโลหิตซิสโตลิกตั้งแต่ 140 มิลลิเมตรปรอทขึ้นไป หรือค่าความดันโลหิตไดแอสโตลิกตั้งแต่ 90 มิลลิเมตรปรอทขึ้นไปหรือจำแนกระดับความดันโลหิต ได้ดังแสดงในตารางที่ 2.1 ภาวะนี้มักไม่แสดงอาการเด่นชัดในระยะแรก ผู้ป่วยจำนวนมากจึงไม่ทราบว่าตนเองมีภาวะดังกล่าวจนนำไปสู่การเกิดภาวะแทรกซ้อนอื่น ๆ [21]

อาการในระยะเริ่มต้นมักมีลักษณะการดำเนินโรคแบบไม่แสดงอาการผู้ป่วยส่วนใหญ่จึงไม่ทราบว่าตนเองมีภาวะดังกล่าวในบางรายอาจปรากฏอาการเบื้องต้นซึ่งไม่จำเพาะเจาะจงต่อโรค ได้แก่ ปวดศีรษะ โดยเฉพาะบริเวณท้ายทอย ตาพร่ามัว ใจสั่น เป็นต้น [22]

ในกรณีที่รุนแรงผู้ป่วยอาจเข้าสู่ภาวะวิกฤตความดันโลหิตสูงซึ่งจำแนกเป็น ความดันสูงรีบด่วนและความดันสูงฉุกเฉิน โดยกลุ่มความดันสูงฉุกเฉินจะมีภาวะอวัยวะเป้าหมายถูกทำลายเฉียบพลันร่วมด้วย เช่น ภาวะเลือดออกในสมอง กล้ามเนื้อหัวใจขาดเลือดเฉียบพลัน หรือไตวายเฉียบพลัน ซึ่งจัดเป็นภาวะฉุกเฉินทางการแพทย์ที่ต้องได้รับการรักษาทันที [23]

ตารางที่ 2.1 เกณฑ์ระดับโรคความดันโลหิตสูง

ระดับความดันโลหิต	ความดันตัวบน	ความดันตัวล่าง
ความดันปกติ	< 120	< 80
สูงกว่าปกติ	120 – 129	< 80
ความดันสูง ระดับที่ 1	130 – 139	80 – 89
ความดันสูง ระดับที่ 2	≥ 140	≥ 90

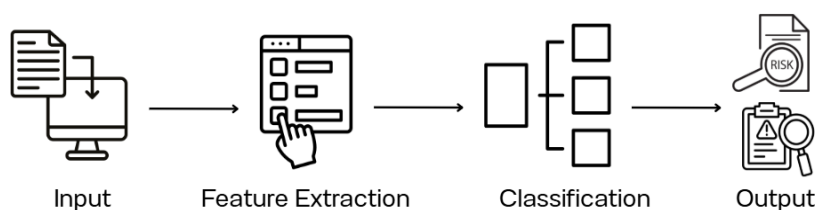
จากตารางที่ 2.1 แสดงเกณฑ์การจำแนกระดับความดันโลหิต ซึ่งใช้ในการอธิบายระดับและลักษณะของภาวะความดันโลหิตสูง โดยพิจารณาจากค่าความดันโลหิตตัวบน (Systolic Blood Pressure: SBP) และค่าความดันโลหิตตัวล่าง (Diastolic Blood Pressure: DBP) ทั้งนี้ เกณฑ์ดังกล่าวใช้เพื่ออธิบายลักษณะและระดับของภาวะความดันโลหิตสูงตามหลักทางคลินิก และไม่ได้ถูกนำมาใช้เป็นตัวแปรนำเข้าในการพัฒนาแบบจำลองการเรียนรู้ของเครื่อง โดยการศึกษาครั้งนี้มีวัตถุประสงค์หลักในการจำแนกประเภทของภาวะความดันโลหิตสูงจากข้อมูลทางคลินิก

2.1.2 การเรียนรู้ของเครื่อง

การเรียนรู้ของเครื่อง คือปัญญาประดิษฐ์ที่มุ่งพัฒนาอัลกอริทึมและแบบจำลองเชิงสถิติให้สามารถเรียนรู้รูปแบบและความสัมพันธ์จากข้อมูลได้ โดยไม่จำเป็นต้องกำหนดกฎหรือเงื่อนไขในการตัดสินใจไว้ล่วงหน้าทุกขั้นตอนโดยแบ่งออกเป็น 2 ประเภทหลัก ได้แก่ การเรียนรู้แบบมีผู้สอน (Supervised Learning) ซึ่งใช้ข้อมูลที่มีฉลากคำตอบในการฝึกแบบจำลอง และการเรียนรู้แบบไม่มีผู้สอน (Unsupervised Learning) ซึ่งมุ่งค้นหาโครงสร้างหรือรูปแบบที่ซ่อนอยู่ภายในข้อมูลโดยไม่อาศัยฉลากคำตอบ [24]

สำหรับงานด้านสุขภาพ การเรียนรู้ของเครื่องมีข้อดีคือสามารถเรียนรู้ความสัมพันธ์แบบไม่เป็นเชิงเส้น ระหว่างปัจจัยเสี่ยงและตัวแปรทางคลินิกที่มีความซับซ้อนได้อย่างมีประสิทธิภาพ โดยเฉพาะอย่างยิ่งในกลุ่มแบบจำลองการเรียนรู้เชิงลึกซึ่งสามารถเรียนรู้และสกัดคุณลักษณะจากข้อมูลได้โดยอัตโนมัติ ลดความจำเป็นในการออกแบบคุณลักษณะด้วยมนุษย์เมื่อเทียบกับแบบจำลองการเรียนรู้ของเครื่องแบบดั้งเดิม [25]

งานวิจัยนี้มุ่งเน้นการเรียนรู้แบบมีผู้สอน (Supervised Learning) ในลักษณะปัญหาการจำแนกประเภทเพื่อทำนายภาวะความดันโลหิตสูงจากปัจจัยเสี่ยงทางคลินิก โดยศึกษาแบบจำลองครอบคลุมทั้งกลุ่มการเรียนรู้ร่วมกันและกลุ่มการเรียนรู้เชิงลึก

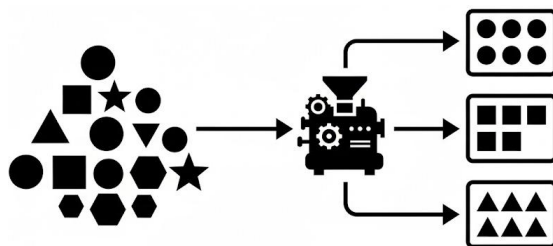


ภาพประกอบที่ 2.1 ภาพการทำงานของเครื่องการเรียนรู้ของเครื่อง

ภาพประกอบที่ 2.1 จะแสดงกระบวนการทำงานของการเรียนรู้ของเครื่องสำหรับการจำแนกประเภท โดยเริ่มจากการรับข้อมูลนำเข้า (Input) จากนั้นดำเนินการสกัดคุณลักษณะ (Feature Extraction) เพื่อเตรียมข้อมูลสำหรับการเรียนรู้ ก่อนนำเข้าสู่แบบจำลองเพื่อจำแนกประเภท และแสดงผลการทำนาย (Output) ซึ่งสามารถนำไปใช้สนับสนุนการตัดสินใจในงานด้านสุขภาพและการคัดกรองโรคได้

2.1.3 การจำแนกประเภท

การจำแนกประเภท (Classification) ในบริบทของการเรียนรู้ของเครื่อง คือกระบวนการสร้างแบบจำลองเพื่อทำนายฉลากให้กับข้อมูลอินพุต โดยผลลัพธ์เป็นข้อมูลแบบไม่ต่อเนื่องเช่น การจำแนกเป็น 2 กลุ่ม หรือหลายกลุ่ม กระบวนการจำแนกประเภทส่วนใหญ่อยู่ในรูปแบบการเรียนรู้แบบมีผู้สอน ซึ่งใช้ข้อมูลตัวอย่างที่มีการกำหนดฉลากไว้ล่วงหน้าในการฝึกแบบจำลอง เพื่อให้สามารถเรียนรู้ความสัมพันธ์ระหว่างคุณลักษณะของข้อมูลและผลลัพธ์ที่ต้องการทำนาย จากนั้นจึงนำแบบจำลองไปใช้ทำนายข้อมูลใหม่ที่ไม่เคยพบมาก่อน ดังแสดงในภาพที่ 2.2 กระบวนการทำงานของการจำแนกประเภท โดยนำข้อมูลที่มีคุณลักษณะหลากหลายเข้าสู่แบบจำลองการเรียนรู้ของเครื่อง เพื่อเรียนรู้รูปแบบของข้อมูลและจำแนกข้อมูลออกเป็นกลุ่มหรือคลาสที่เหมาะสมตามผลการทำนายของแบบจำลอง [26]

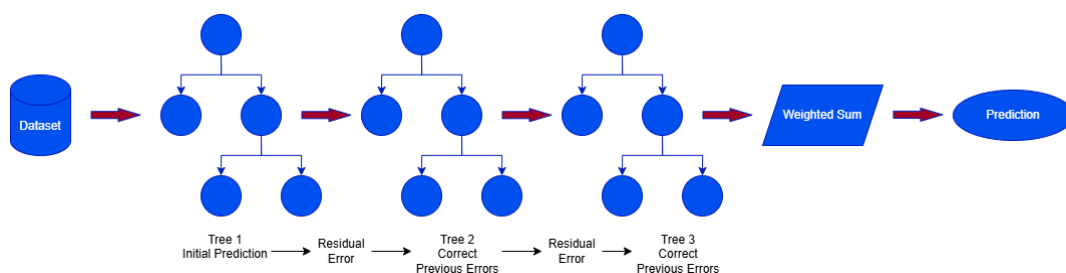


ภาพประกอบที่ 2.2 กระบวนการจำแนกประเภทแบบการเรียนรู้แบบมีผู้สอน

จากภาพที่ 2.2 แสดงให้เห็นกระบวนการทำงานของการจำแนกประเภท โดยเริ่มจากการนำข้อมูลที่ประกอบด้วยคุณลักษณะต่าง ๆ เข้าสู่แบบจำลองการเรียนรู้ของเครื่อง แบบจำลองจะเรียนรู้รูปแบบและความสัมพันธ์ระหว่างคุณลักษณะของข้อมูลกับผลลัพธ์ที่มีการกำหนดค่าไว้ จากนั้นจึงนำความรู้ที่ได้ไปใช้ในการจำแนกข้อมูลใหม่ออกเป็นกลุ่มหรือคลาสตามผลการทำนายของแบบจำลอง ซึ่งในการศึกษาครั้งนี้ประยุกต์ใช้แนวคิดการจำแนกประเภทเพื่อพัฒนาแบบจำลองสำหรับจำแนกภาวะโรคความดันโลหิตสูงจากข้อมูลทางคลินิก โดยกำหนดผลลัพธ์การจำแนกเป็นประเภทของภาวะความดันโลหิตสูงตามข้อมูลที่ใช้ในการศึกษา

2.1.4 เอ็กซ์จีบูสต์

เอ็กซ์จีบูสต์ (XGBoost: Extreme Gradient Boosting) เป็นอัลกอริทึมในกลุ่ม Gradient Boosting ที่สร้างแบบจำลองจากชุดของต้นไม้ตัดสินใจหลายต้น โดยต้นไม้แต่ละต้นจะถูกสร้างขึ้นอย่างต่อเนื่องเพื่อปรับปรุงผลการทำนายของแบบจำลองจากต้นไม้อ่อนหน้า ผ่านกระบวนการลดค่าฟังก์ชันความสูญเสีย (Loss Function) ด้วยแนวคิดการไล่ระดับ (Gradient Descent) ดังภาพประกอบที่ 2.3 กระบวนการดังกล่าวทำให้แบบจำลองสามารถลดข้อผิดพลาดของการทำนายและเพิ่มประสิทธิภาพของแบบจำลองได้อย่างต่อเนื่อง



ภาพประกอบที่ 2.3 ภาพการทำงานของเอ็กซ์จีบูสต์

จากภาพที่ 2.3 แสดงกระบวนการทำงานของอัลกอริทึม XGBoost ซึ่งประกอบด้วยต้นไม้ตัดสินใจหลายต้นที่ถูกสร้างขึ้นอย่างต่อเนื่อง โดยต้นไม้แต่ละต้นจะเรียนรู้เพื่อปรับปรุงผลการทำนายจากต้นไม้อ่อนหน้า และนำผลลัพธ์จากต้นไม้ทั้งหมดมารวมกันเพื่อให้ได้ผลการทำนายสุดท้าย ทั้งนี้ กระบวนการสร้างแบบจำลองของ XGBoost สามารถอธิบายได้ด้วยสมการทางคณิตศาสตร์ โดยแบบจำลองจะรวมผลลัพธ์จากต้นไม้ตัดสินใจแต่ละต้นเพื่อคำนวณเป็นค่าการทำนายของข้อมูลแต่ละตัวอย่าง ดังแสดงในสมการที่ 2.1

$$y^i = \sum_{k=1}^K f_k(x_i), \quad f_k \in \mathcal{F}$$

สมการที่ 2.1 สมการของ XGBoost (1)

โดยที่ $\hat{y}^i$	คือค่าที่ตัวแบบทำนายสำหรับข้อมูลตัวอย่างที่ i
f_k	คือฟังก์ชันของต้นไม้ตัดสินใจลำดับที่ k
K	คือจำนวนต้นไม้ทั้งหมด
$\mathcal{F}$	คือเซตของต้นไม้ตัดสินใจ

จากสมการที่ 2.1 แสดงให้เห็นว่า XGBoost สร้างค่าการทำนายของข้อมูลแต่ละตัวอย่างจากการรวมผลลัพธ์ของต้นไม้ตัดสินใจหลายต้นเข้าด้วยกัน โดย $f_k(x_i)$ แทนผลลัพธ์ที่ได้จากต้นไม้ตัดสินใจต้นที่ k เมื่อรับข้อมูลตัวอย่างที่ i เข้ามาเป็นอินพุต จากนั้นผลลัพธ์ของต้นไม้แต่ละต้นจะถูกนำมารวมกันตั้งแต่ต้นไม้นั้นแรกจนถึงต้นไม้นั้นสุดท้าย เพื่อคำนวณเป็นค่าการทำนายสุดท้าย $\hat{y}^i$ ของแบบจำลอง ดังนั้น สมการดังกล่าวจึงสะท้อนหลักการสำคัญของ XGBoost ที่ใช้การรวมผลจากต้นไม้ตัดสินใจหลายต้นเพื่อเพิ่มความสามารถในการเรียนรู้และปรับปรุงความแม่นยำของการทำนาย

2.1.5 ป่าสุ่ม

ป่าแบบสุ่ม (Random Forest) เป็นอัลกอริทึมที่ประยุกต์ใช้แนวคิดการรวมแบบจำลอง (Ensemble Learning) โดยอาศัยวิธีการสุ่มตัวอย่างข้อมูลแบบ Bagging ร่วมกับการสุ่มเลือกตัวแปรบางส่วน (Feature Subsampling) สำหรับใช้ในการสร้างต้นไม้ตัดสินใจแต่ละต้น วิธีการดังกล่าวช่วยลดความสัมพันธ์ระหว่างต้นไม้แต่ละต้นและลดความแปรปรวนของแบบจำลองโดยรวม ทำให้แบบจำลองมีความสามารถในการเรียนรู้ข้อมูลได้อย่างมีประสิทธิภาพ นอกจากนี้ Random Forest ยังสามารถประเมินความสำคัญของตัวแปร (Feature Importance) ซึ่งเป็นประโยชน์ต่อการวิเคราะห์ว่าตัวแปรใดมีส่วนสำคัญต่อผลการจำแนกประเภท

$$\hat{y} = \text{mode}\{h_1(x), h_2(x), \dots, h_B(x)\}$$

สมการที่ 2.2 สมการการทำนายของป่าแบบสุ่ม

โดยที่ $\hat{y}$	คือผลลัพธ์ที่แบบจำลองพยากรณ์
$h_B(x)$	คือผลการพยากรณ์จากต้นไม้ลำดับที่ h
B	คือจำนวนต้นไม้ทั้งหมด
$\text{mode}(\cdot)$	คือฟังก์ชันเลือกค่าที่ได้รับการลงคะแนนเสียงมากที่สุด

จากสมการที่ 2.2 แสดงหลักการการทำนายของ Random Forest สำหรับงานจำแนกประเภท โดยแบบจำลองจะรวบรวมผลการทำนายจากต้นไม้ตัดสินใจทั้งหมด แล้วใช้การลงคะแนนเสียงข้างมาก (Majority Voting) เพื่อเลือกคลาสที่ได้รับการทำนายจากต้นไม้มากที่สุดเป็นผลลัพธ์สุดท้ายของแบบจำลอง ทั้งนี้ Random Forest มีจุดเด่นในการลดความแปรปรวนของแบบจำลองและลดโอกาสเกิดการเรียนรู้เกินพอดี

(Overfitting) เมื่อเปรียบเทียบกับการใช้ต้นไม้ตัดสินใจเพียงต้นเดียว อีกทั้งยังสามารถนำไปประยุกต์ใช้กับงานจำแนกประเภทที่มีข้อมูลและตัวแปรหลายประเภทได้อย่างมีประสิทธิภาพ [27]

2.1.6 ไลต์จีบีเอ็ม

ไลต์จีบีเอ็ม (LightGBM: Light Gradient Boosting Machine) เป็นอัลกอริทึมในกลุ่ม Gradient Boosting ที่พัฒนาขึ้นเพื่อเพิ่มประสิทธิภาพและความรวดเร็วในการฝึกแบบจำลอง โดยใช้กลยุทธ์การเติบโตของต้นไม้แบบ Leaf-wise ซึ่งจะเลือกใบของต้นไม้ที่สามารถลดค่าความสูญเสียได้มากที่สุดเพื่อทำการแบ่งต่อไป ร่วมกับการแบ่งค่าของตัวแปรออกเป็นช่วงด้วยวิธี Histogram-based Binning ซึ่งช่วยลดการใช้หน่วยความจำและเพิ่มความเร็วในการประมวลผล โดยยังคงประสิทธิภาพในการทำนายที่ดี

$$y^i = \sum_{k=1}^K f_k(x_i), f_k \in$$

สมการที่ 2.3 สมการการทำนายของไลต์จีบีเอ็ม

โดยที่ K คือจำนวนต้นไม้ตัดสินใจ

$f_k(x_i)$ คือผลลัพธ์จากต้นไม้ลำดับที่ k สำหรับข้อมูล x_i

จากสมการที่ 2.3 แสดงหลักการสร้างผลการทำนายของ LightGBM โดยแบบจำลองจะรวมผลลัพธ์จากต้นไม้ตัดสินใจหลายต้นที่ถูกสร้างขึ้นอย่างต่อเนื่อง เพื่อปรับปรุงผลการทำนายของแบบจำลองในแต่ละรอบ และนำผลลัพธ์ของต้นไม้ทั้งหมดมารวมกันเป็นค่าการทำนายสุดท้ายของข้อมูลแต่ละตัวอย่าง

ทั้งนี้ LightGBM มีความแตกต่างจาก Gradient Boosting Decision Tree (GBDT) ทั่วไปในด้านกลยุทธ์และเทคนิคที่ใช้เพื่อเพิ่มประสิทธิภาพในการฝึกแบบจำลอง โดยใช้การเติบโตของต้นไม้แบบ Leaf-wise Tree Growth ร่วมกับเทคนิค Gradient-based One-Side Sampling (GOSS) และ Exclusive Feature Bundling (EFB) ซึ่งช่วยลดจำนวนข้อมูลและจำนวนตัวแปรที่ต้องใช้ในการประมวลผล ลดการใช้หน่วยความจำ และเพิ่มความเร็วในการฝึกแบบจำลอง โดยยังคงประสิทธิภาพในการพยากรณ์ในระดับสูง [28]

2.1.7 แคตบูสต์

แคตบูสต์ (CatBoost: Categorical Boosting) เป็นอัลกอริทึมในกลุ่ม Gradient Boosting Decision Tree (GBDT) ที่ได้รับการออกแบบมาเพื่อรองรับข้อมูลเชิงหมวดหมู่ (Categorical Data) ได้อย่างมีประสิทธิภาพ โดยแบบจำลองจะสร้างต้นไม้ตัดสินใจหลายต้นอย่างต่อเนื่อง และนำผลลัพธ์จากต้นไม้แต่ละต้นมารวมกันเพื่อคำนวณเป็นผลการทำนายสุดท้ายของข้อมูลแต่ละตัวอย่าง ดังแสดงในสมการที่ 2.4

$$\hat{y}_i = \sum_{k=1}^K f_k(x_i)$$

สมการที่ 2.4 สมการการทำนายของแคตบูสต์

โดยที่ $(\hat{y}_i)$

คือค่าผลการทำนายของแบบจำลองสำหรับข้อมูลตัวอย่างที่ i

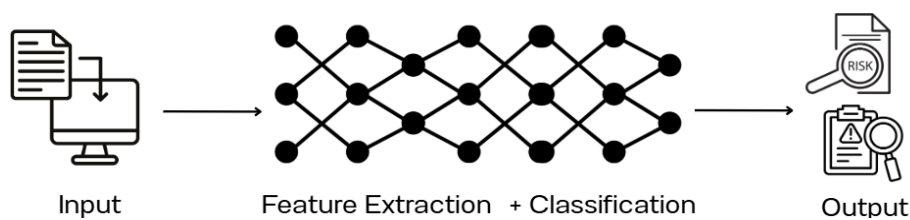
$(f_k(x_i))$ คือผลลัพธ์จากต้นไม้ตัดสินใจลำดับที่ k ของข้อมูลตัวอย่าง (x_i)
 K คือจำนวนต้นไม้ตัดสินใจทั้งหมด

จากสมการที่ 2.4 แสดงหลักการทำงานของ CatBoost ซึ่งค่าการทำนายของข้อมูลตัวอย่างแต่ละรายการเกิดจากการรวมผลลัพธ์ของต้นไม้ตัดสินใจทั้งหมดที่สร้างขึ้นในแบบจำลอง โดยต้นไม้แต่ละต้นจะมีส่วนช่วยปรับปรุงผลการทำนายของแบบจำลองจากรอบก่อนหน้า ส่งผลให้แบบจำลองสามารถเรียนรู้รูปแบบและความสัมพันธ์ที่ซับซ้อนของข้อมูลได้

ทั้งนี้ CatBoost มีจุดเด่นที่สำคัญคือการใช้วิธี Ordered Boosting ซึ่งช่วยลดปัญหา Prediction Shift ที่อาจเกิดขึ้นระหว่างกระบวนการฝึกแบบจำลอง และการจัดการข้อมูลเชิงหมวดหมู่ด้วยวิธี Ordered Target Statistics หรือ Ordered Target Encoding โดยคำนวณสถิติจากข้อมูลที่มีลำดับอย่างเหมาะสมเพื่อลดการใช้ข้อมูลเป้าหมายของตัวอย่างเดียวกันในการเข้ารหัส ซึ่งช่วยลดความเสี่ยงของการเกิดข้อมูลรั่วไหล (Target Leakage) และเพิ่มความน่าเชื่อถือของแบบจำลอง [29]

2.1.8 การเรียนรู้เชิงลึก

การเรียนรู้เชิงลึก (Deep learning) เป็นสาขาหนึ่งของการเรียนรู้ของเครื่อง [30] ที่ใช้โครงข่ายประสาทเทียมซ้อนกันหลายชั้นเพื่อเรียนรู้ตัวแทนเชิงซับซ้อนของข้อมูลโดยอัตโนมัติแสดงดังภาพประกอบที่ 2.4 การเรียนรู้เชิงลึกสามารถเรียนรู้พีเจอร์และจับความสัมพันธ์เชิงไม่เชิงเส้นได้โดยอัตโนมัติ จึงช่วยให้ตัวแบบมีความแม่นยำสูงขึ้น ในงานด้านข้อมูลสุขภาพ ข้อดีของการเรียนรู้เชิงลึก คือสามารถเรียนรู้รูปแบบที่ซับซ้อนกว่าการจำลองเชิงเส้นแบบเดิมได้ดี แต่เมื่อใช้กับข้อมูลแบบตารางแล้ว พบว่ามีข้อจำกัด เช่น ข้อมูลตารางไม่มีโครงสร้างลำดับ [31]



ภาพประกอบที่ 2.4 ภาพการทำงานของ การเรียนรู้เชิงลึก

จากภาพประกอบที่ 2.5 แสดงกระบวนการทำงานของแบบจำลองโครงข่ายประสาทเทียมเชิงลึก โดยเริ่มจากการนำข้อมูลอินพุตเข้าสู่แบบจำลอง จากนั้นข้อมูลจะถูกส่งผ่านชั้นต่าง ๆ ของโครงข่ายเพื่อสกัดคุณลักษณะ (Feature Extraction) และเรียนรู้รูปแบบของข้อมูล ก่อนเข้าสู่กระบวนการจำแนกประเภท (Classification) และได้ผลลัพธ์การทำนายเป็นเอาต์พุต (Output) ซึ่งสามารถนำไปใช้ในการจำแนกประเภทของภาวะความดันโลหิตสูงตามวัตถุประสงค์ของการศึกษาได้

2.1.9 โครงข่ายประสาทเทียมแบบหลายชั้น

โครงข่ายประสาทเทียมแบบหลายชั้น (Multilayer Perceptron: MLP) เป็นโครงข่ายประสาทเทียมแบบส่งไปข้างหน้า (Feedforward Neural Network) ที่ประกอบด้วยชั้นนำเข้า (Input Layer) ชั้นซ่อน (Hidden Layer) อย่างน้อยหนึ่งชั้น และชั้นส่งออก (Output Layer) โดยแต่ละโหนดในชั้นต่าง ๆ เชื่อมต่อกัน

ด้วยค่าน้ำหนัก (Weight) และมีค่าไบแอส (Bias) ซึ่งค่าดังกล่าวจะถูกปรับปรุงระหว่างกระบวนการฝึกแบบจำลองด้วยวิธีการแพร่ย้อนกลับของความผิดพลาด (Backpropagation) ร่วมกับกระบวนการปรับค่าพารามิเตอร์เพื่อลดค่าความสูญเสียของแบบจำลอง

$$a^{(l)} = \sigma(W^{(l)}a^{(l-1)} + b^{(l)})$$

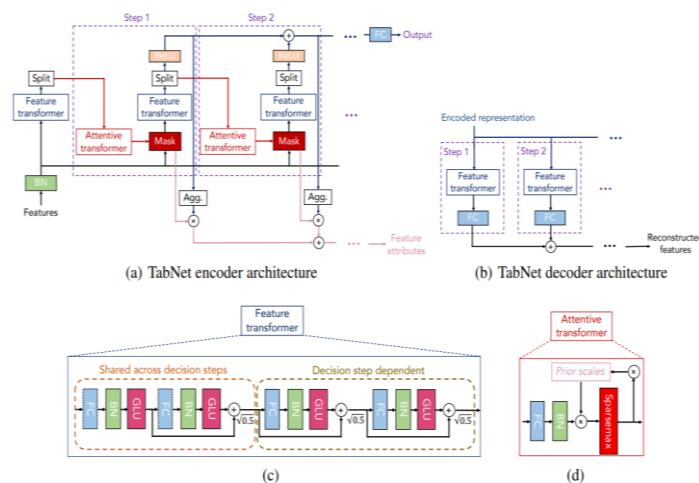
สมการที่ 2.5 สมการการทำนายของโครงข่ายประสาทเทียมแบบหลายชั้น

โดยที่ $a^{(l)}$ คือผลลัพธ์ของชั้นที่ l
 $W^{(l)}$ คือเมทริกซ์ค่าน้ำหนัก
 $b^{(l)}$ คือเวกเตอร์ไบแอส
 $\sigma(\cdot)$ คือฟังก์ชันกระตุ้น

จากสมการที่ 2.5 แสดงให้เห็นว่า MLP คำนวณผลลัพธ์ของแต่ละชั้นโดยนำผลลัพธ์จากชั้นก่อนหน้ามาคูณกับเมทริกซ์ค่าน้ำหนัก จากนั้นบวกด้วยเวกเตอร์ไบแอสและนำผลลัพธ์ที่ได้ผ่านฟังก์ชันกระตุ้น กระบวนการดังกล่าวจะดำเนินต่อเนื่องผ่านชั้นต่าง ๆ จนถึงชั้นส่งออกเพื่อสร้างผลการทำนายสุดท้าย ทั้งนี้ MLP สามารถเรียนรู้ความสัมพันธ์แบบไม่เป็นเชิงเส้น (Nonlinear Relationship) ระหว่างคุณลักษณะของข้อมูลกับผลลัพธ์ได้ดี จึงเหมาะสำหรับงานจำแนกประเภทและการพยากรณ์ที่มีรูปแบบความสัมพันธ์ซับซ้อน ซึ่งสามารถนำไปใช้ในการจำแนกประเภทของภาวะความดันโลหิตสูงตามวัตถุประสงค์ของการศึกษาได้ [32]

2.1.10 แท็บเน็ต

แท็บเน็ต (TabNet) เป็นสถาปัตยกรรมหนึ่งของการเรียนรู้เชิงลึก ที่ออกแบบมาเพื่อข้อมูลแบบตารางโดยเฉพาะ ซึ่งมีความโดดเด่นในเรื่องการตีความผลลัพธ์ ประกอบกับประสิทธิภาพสูง หลักการสำคัญของ TabNet คือการใช้กลไก attention แบบเป็นขั้นตอนในการเลือกฟีเจอร์ที่สำคัญมาใช้ในการตัดสินใจทีละขั้น ในแต่ละขั้น TabNet จะมีการทำงานแบบสองส่วนคือ การเลือกฟีเจอร์ที่ให้ข้อมูลมากที่สุด และการประมวลผลผ่านโครงข่าย ดังในภาพประกอบที่ 2.5 จะแสดงให้เห็นภาพของกระบวนการทำงานของ TabNet ในแต่ละขั้นของการทำงาน [33]



ภาพประกอบที่ 2.5 ภาพการทำงานของตัวแบบแท็บเน็ต

สมการการเลือกคุณลักษณะ

$$M^{(t)} = \text{Sparsemax}(P^{(t)} \cdot h^{(t-1)})$$

สมการที่ 2.6 สมการของแท็บเน็ต (1)

โดยที่	$M^{(t)}$	คือ mask สำหรับเลือกคุณลักษณะว่าฟีเจอร์ไหนสำคัญ
	$h^{(t-1)}$	คือตัวแทนจากขั้นก่อนหน้า
	Sparsemax	คือทำให้การเลือกคุณลักษณะมีความเบาบาง
	$h^{(t-1)}$	คือตัวแทนที่ซ่อนจากขั้นตอนก่อนหน้า

จากสมการที่ 2.6 ซึ่งใช้แสดงกระบวนการสร้าง Mask ของ Attentive Transformer เป็นรูปแบบย่อเพื่ออธิบายแนวคิดพื้นฐานของการเลือกคุณลักษณะ โดยในสถาปัตยกรรม TabNet จริงจะมีการนำตัวแปรมาตราส่วนก่อนหน้ามาควบคุมการเลือกคุณลักษณะร่วมด้วยซึ่งช่วยลดโอกาสในการเลือกใช้คุณลักษณะเดิมซ้ำ และส่งเสริมให้แต่ละ Decision Step สามารถเรียนรู้คุณลักษณะที่แตกต่างกันได้อย่างมีประสิทธิภาพ

$$X^{(t)} = M^{(t)} \odot X$$

สมการที่ 2.7 สมการของแท็บเน็ต (2)

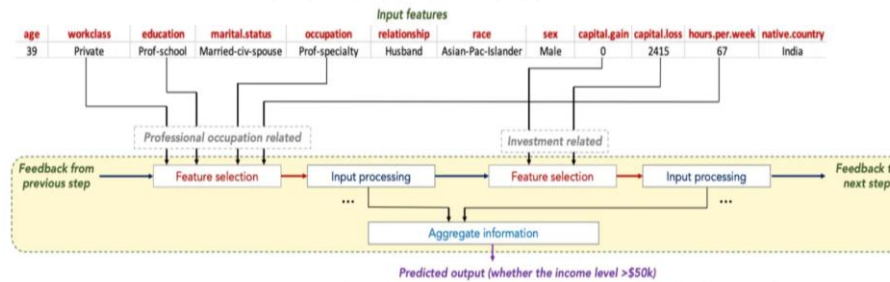
โดยที่	$\odot$	คือการคูณเชิงองค์ประกอบ
	$M^{(t)}$	คือ mask สำหรับเลือกคุณลักษณะในขั้นที่ t
	X	คือข้อมูลต้นฉบับ
	$X^{(t)}$	คือข้อมูลใหม่ในขั้นตอนที่ t

สมการเอาต์พุตรวม

$$\hat{y} = \sum_{t=1}^T f^{(t)}(X^{(t)})$$

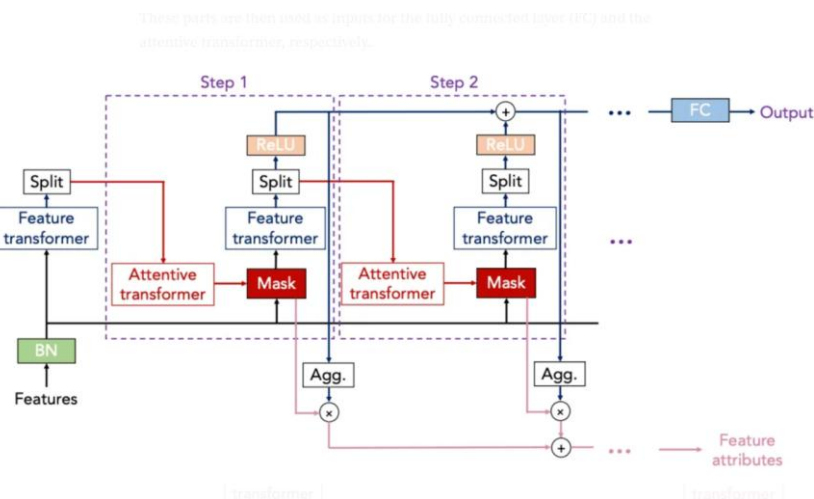
สมการที่ 2.8 สมการของแท็บเน็ต (3)

โดยที่	T	คือจำนวนขั้นตอนการตัดสินใจ
	$f^{(t)}$	คือฟังก์ชันแปลงคุณลักษณะในแต่ละขั้น
	$M^{(t)}$	คือ mask สำหรับเลือกคุณลักษณะในขั้น
	$\hat{y}$	คือค่าที่โมเดลทำนายในผลลัพธ์สุดท้าย
	$X^{(t)}$	คือข้อมูลที่ผ่านการเลือกฟีเจอร์แล้ว



ภาพประกอบที่ 2.6 ภาพการเลือกการนำเข้าคุณลักษณะของแท็บเน็ต

จากภาพประกอบที่ 2.6 แสดงถึงโครงสร้างการทำงานของ TabNet ในกระบวนการเลือกคุณลักษณะ โดยโมเดลจะรับข้อมูลคุณลักษณะนำเข้า แล้วใช้กลไก attention เพื่อคัดเลือกคุณลักษณะที่สำคัญในแต่ละขั้นตอน จากนั้นทำการประมวลผลข้อมูล และรวมข้อมูล ก่อนส่งต่อไปยังขั้นตอนถัดไปแบบลำดับ พร้อมมีการส่งข้อมูลย้อนกลับ เพื่อปรับปรุงการเลือกคุณลักษณะในรอบถัดไป สุดท้ายจึงนำไปใช้ในการทำนายผลลัพธ์ของแบบจำลองและใน ภาพประกอบที่ 2.7 แสดงสถาปัตยกรรมภายในของโมเดล TabNet ซึ่งประกอบด้วยหลายขั้นตอนการตัดสินใจ โดยในแต่ละขั้นตอนจะมีส่วนของ Feature Transformer สำหรับแปลงข้อมูล และ Attentive Transformer สำหรับสร้าง mask เพื่อเลือกคุณลักษณะที่สำคัญ จากนั้นข้อมูลจะถูกแยก (Split) และรวม เพื่อสะสมสารสนเทศจากแต่ละขั้นตอน พร้อมทั้งมีการส่งต่อและปรับปรุงค่าคุณลักษณะ อย่างต่อเนื่อง จนได้ผลลัพธ์สุดท้ายผ่าน Fully Connected Layer (FC) เพื่อใช้ในการทำนายค่าเป้าหมายของแบบจำลอง



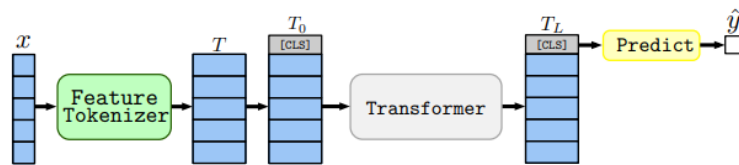
ภาพประกอบที่ 2.7 ภาพการแสดงสถาปัตยกรรมภายในของแท็บเน็ต

จากภาพประกอบที่ 2.7 สามารถอธิบายลำดับการทำงานของแบบจำลอง TabNet ได้ว่า ข้อมูลนำเข้าจะถูกส่งผ่านกระบวนการตัดสินใจทีละขั้นตอน โดย Attentive Transformer จะทำหน้าที่กำหนด Attention Mask เพื่อคัดเลือกคุณลักษณะที่มีความสำคัญในแต่ละขั้นตอน จากนั้น Feature Transformer จะนำคุณลักษณะที่ได้รับการคัดเลือกไปเรียนรู้และสร้างตัวแทนข้อมูลใหม่ ก่อนแบ่งผลลัพธ์ออกเป็น ส่วนสำหรับการทำนายและส่วนที่ส่งต่อไปยังขั้นตอนถัดไป กระบวนการนี้ช่วยให้แบบจำลองสามารถมุ่งเน้นไป

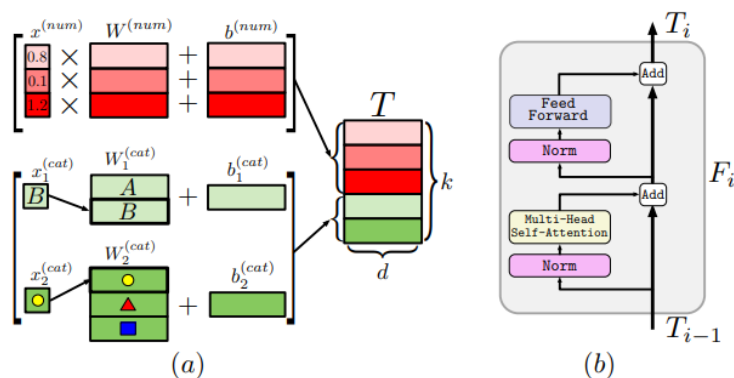
ยังคุณลักษณะที่แตกต่างกันในแต่ละขั้นตอน และสะสมสารสนเทศที่ได้จากหลายขั้นตอนเข้าด้วยกัน ก่อนส่งผ่าน Fully Connected Layer (FC) เพื่อสร้างผลลัพธ์สำหรับการทำนายค่าเป้าหมาย [33]

2.1.11 เอฟที-ทรานฟอร์มเมอร์

เอฟที-ทรานฟอร์มเมอร์ (FT-Transformer) เป็นสถาปัตยกรรมที่ใช้กลไกการแปลงพีเจอร์แต่ละตัวให้เป็นโทเค็นของตัวเอง จากนั้นคำนวณระหว่างโทเค็นเหล่านั้นเพื่อให้ตัวแบบเรียนรู้ปฏิสัมพันธ์ระหว่างพีเจอร์ทุกคู่ เอฟที-ทรานฟอร์มเมอร์แตกต่างจากการใช้ทรานฟอร์มเมอร์ในการประมวลผลภาษาธรรมชาติ (NLP) ที่ข้อมูลมีลำดับของคำชัดเจน ข้อมูลตารางไม่มีลำดับเชิงเวลาโดยธรรมชาติ จึงต้องแทนแต่ละพีเจอร์เป็น embedding และให้การเรียนรู้ของตัวเองทำงานบนพีเจอร์เหล่านั้นทั้งหมดพร้อมกัน โครงสร้างนี้ทำให้เอฟที-ทรานฟอร์มเมอร์ สามารถเข้าถึงความสัมพันธ์ข้ามพีเจอร์ทั้งชนิดตัวเลขและหมวดหมู่ได้โดยทั่วถึง [34] จุดเด่นของเอฟที-ทรานฟอร์มเมอร์ สำหรับข้อมูลตารางคือสามารถจัดการกับข้อมูลแบบผสมและปฏิสัมพันธ์ของพีเจอร์ได้ดี โดยไม่ต้องออกแบบคุณลักษณะเพิ่มเติมมากแต่ต้องพึ่งข้อมูลจำนวนมากในการฝึก โดยภาพประกอบที่ 2.8 คือการเลือกคุณลักษณะโดยเริ่มจากข้อมูลนำเข้า x ที่ถูกแปลงเป็นโทเค็นผ่าน Feature Tokenizer จากนั้นเพิ่มโทเค็นพิเศษ [CLS] เพื่อใช้เป็นตัวแทนของข้อมูลทั้งหมด แล้วส่งผ่านชั้น Transformer เพื่อเรียนรู้ความสัมพันธ์ระหว่างคุณลักษณะ สุดท้ายใช้โทเค็น [CLS] ในการทำนายผลลัพธ์ $\hat{y}$ ผ่านขั้นตอนการพยากรณ์ [35]



ภาพประกอบที่ 2.8 ภาพการเลือกคุณลักษณะและการจำแนกประเภท



ภาพประกอบที่ 2.9 ภาพการ Feature Tokenizer transforms

จากภาพประกอบที่ 2.9 แสดงกระบวนการแปลงข้อมูลเชิงตารางเข้าสู่โมเดล Transformer โดยในส่วน (a) เป็นการแปลงคุณลักษณะเชิงตัวเลขและเชิงหมวดหมู่ผ่านการคูณน้ำหนัก และบวกไบแอสเพื่อสร้างเวกเตอร์ตัวแทน จากนั้นรวมเป็นเมทริกซ์ T ที่มีมิติ $k \times d$ สำหรับใช้เป็นอินพุตของโมเดล ในส่วน (b) แสดงโครงสร้างของ Transformer Block ซึ่งประกอบด้วย Multi-Head Self-Attention และ Feed Forward

Network โดยมีการทำ Normalization และการเชื่อมต่อแบบ Residual เพื่อช่วยให้โมเดลสามารถเรียนรู้ความสัมพันธ์ระหว่างคุณลักษณะได้อย่างมีประสิทธิภาพ ก่อนส่งต่อไปยังชั้นถัดไปในลำดับการประมวลผล

สมการการแปลงคุณลักษณะเป็น Token

$$z_j = x_j \cdot W_j + b_j$$

สมการที่ 2.9 สมการของเอฟที-ทรานฟอร์มเมอร์ (1)

โดยที่	x_j	คือคุณลักษณะลำดับที่ j
	W_j	คือน้ำหนักของอินพุต
	b_j	คือค่าความลำเอียง
	z_j	คือค่าผลลัพธ์ก่อน activation

จาก สมการที่ 2.9 คือขั้นตอนการเปลี่ยนข้อมูลแต่ละคุณลักษณะให้เป็น Token ที่แบบจำลองสามารถนำไปเรียนรู้ความสัมพันธ์ระหว่างคุณลักษณะต่าง ๆ ได้

สมการ Self-Attention

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d}}\right)V$$

สมการที่ 2.10 สมการของเอฟที-ทรานฟอร์มเมอร์ (2)

โดยที่	Q	คือสิ่งที่เราต้องการหา
	K	คือตัวบอกลักษณะของข้อมูล
	V	คือข้อมูลจริงที่ต้องการนำมาใช้
	QK^T	คือการคูณเมทริกซ์
	$\sqrt{d}$	คือขนาดของเวกเตอร์ (dimension)
	softmax	คือแปลงค่าให้เป็น ความน่าจะเป็น
	QK^T	คือการคูณเมทริกซ์

จากสมการที่ 2.10 เป็นกลไกที่ช่วยให้แบบจำลองค้นหาความสัมพันธ์และกำหนดระดับความสำคัญระหว่างคุณลักษณะต่าง ๆ

สมการ Multi-Head Attention

$$\text{MHA}(X) = \text{Concat}(\text{head}_1, \dots, \text{head}_h)W_o$$

สมการที่ 2.11 สมการของเอฟที-ทรานฟอร์มเมอร์ (3)

โดยที่	MHA	คือผลลัพธ์ของ Multi-Head Attention
	$\text{head}_1, \dots, \text{head}_h$	คือ Attention หลายชุด
	Concat	คือการนำผลลัพธ์ของทุก head มาต่อกัน
	W_o	คือเมทริกซ์น้ำหนัก

จากสมการที่ 2.11 แสดงกระบวนการ Multi-Head Attention ซึ่งเป็นการนำกลไก Self-Attention หลายชุด หรือหลาย Head มาประมวลผลข้อมูลพร้อมกัน แต่ละ Head สามารถเรียนรู้รูปแบบความสัมพันธ์ระหว่างคุณลักษณะที่แตกต่างกันได้ หลังจากแต่ละ Head ประมวลผลข้อมูลแล้ว ผลลัพธ์จากทุก Head จะถูกนำมาต่อกันด้วยกระบวนการ Concatenation จากนั้นนำไปคูณกับเมทริกซ์น้ำหนัก W_o เพื่อสร้างผลลัพธ์สุดท้ายของ Multi-Head Attention การใช้หลาย Head ช่วยให้แบบจำลองสามารถเรียนรู้ความสัมพันธ์ของข้อมูลได้ หลากมุมมองพร้อมกัน เช่น Head หนึ่งอาจเรียนรู้ความสัมพันธ์ระหว่างคุณลักษณะด้านประชากรศาสตร์ ในขณะที่อีก Head อาจเรียนรู้ความสัมพันธ์ระหว่างคุณลักษณะทางคลินิก

สมการผลลัพธ์ทำนายสุดท้าย

$$\hat{y} = \text{MLP}(\text{Transformer}(Z))$$

สมการที่ 2.12 สมการของ FT-Transformer (4)

โดยที่	MLP	คือโครงข่ายประสาทเทียมหลายชั้น
	Transformer	คือการนำเข้าโมเดลทรานส์ฟอร์มเมอร์
	Z	คือการนำเข้าของทรานส์ฟอร์มเมอร์
	$\hat{y}$	คือค่าทำนายสุดท้าย

จากสมการที่ 2.12 แสดงกระบวนการสร้าง ผลลัพธ์การทำนายสุดท้าย ของ FT-Transformer โดยข้อมูลที่ผ่านกระบวนการ Transformer และการเรียนรู้ความสัมพันธ์ระหว่างคุณลักษณะแล้ว จะถูกส่งเข้าสู่โครงข่ายประสาทเทียมแบบหลายชั้น หรือ Multi-Layer Perceptron (MLP) เพื่อประมวลผลและแปลงตัวแทนข้อมูลให้เป็นค่าผลลัพธ์ที่ใช้สำหรับการทำนาย สำหรับปัญหาการจำแนกแบบสองกลุ่ม เช่น การจำแนกภาวะความดันโลหิตสูง ผลลัพธ์สุดท้ายสามารถแสดงเป็นค่าความน่าจะเป็นของแต่ละกลุ่ม ซึ่งสามารถนำไปใช้กำหนดผลการจำแนกว่าเป็นกลุ่มปกติหรือกลุ่มที่มีภาวะความดันโลหิตสูงได้

2.1.12 เซนต์

เซนต์ (SAINT: Row Attention and Column Attention for Tabular Data) เป็นสถาปัตยกรรมการเรียนรู้เชิงลึกสำหรับข้อมูลแบบตารางที่พัฒนาต่อยอดจากแนวคิดของ Transformer โดยมุ่งเน้นการเรียนรู้ความสัมพันธ์ของข้อมูลทั้งในระดับคุณลักษณะและระดับตัวอย่าง จุดเด่นของ SAINT คือการใช้กลไกความใส่ใจ (Attention Mechanism) ทั้งในแนวนอน (Column Attention) และแนวแถว (Row Attention) เพื่อให้แบบจำลองสามารถเรียนรู้ความสัมพันธ์ระหว่างคุณลักษณะภายในตัวอย่างเดียวกัน รวมถึง

ความสัมพันธ์ระหว่างตัวอย่างที่แตกต่างกันในชุดข้อมูล นอกจากนี้ SAINT ยังสนับสนุนการเรียนรู้แบบคอนทราสทีฟพรีเทรนนิ่ง (Contrastive Pre-training) เพื่อเรียนรู้ตัวแทนข้อมูลที่มีความเหมาะสมก่อนนำไปใช้งานจำแนกประเภท ส่งผลให้แบบจำลองสามารถเรียนรู้โครงสร้างและความสัมพันธ์ที่ซับซ้อนของข้อมูลแบบตารางได้อย่างมีประสิทธิภาพ [36]

$$e_i = f_i(x_i)$$

สมการที่ 2.13 สมการของเซนต์ (1)

โดยที่ x_i คือค่าของคุณลักษณะที่ i
 e_i คือเวกเตอร์ตัวแทนของคุณลักษณะดังกล่าว

SAINT นำข้อมูลจาก หลายตัวอย่าง (Rows) มาเรียนรู้ความสัมพันธ์ระหว่างตัวอย่างผ่าน Attention ทำให้ตัวอย่างหนึ่งสามารถใช้สารสนเทศจากตัวอย่างอื่น ๆ ในชุดข้อมูลมาช่วยสร้างตัวแทนข้อมูลได้ สามารถอธิบายเชิงแนวคิดได้ว่า

$$H_{\text{row}} = \text{Attention}(Q_{\text{row}}, K_{\text{row}}, V_{\text{row}})$$

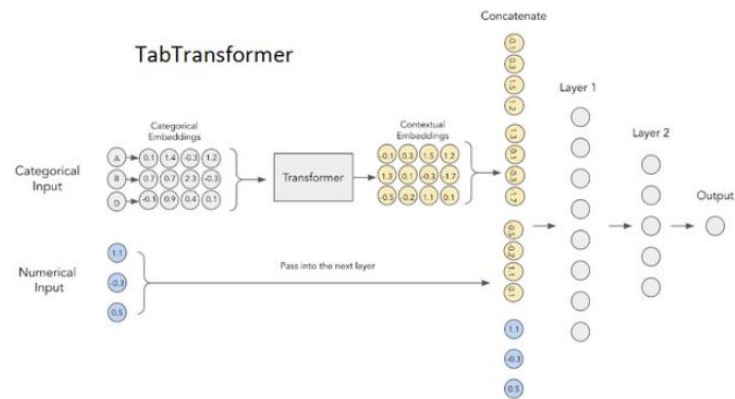
สมการที่ 2.14 สมการของเซนต์ (2)

โดยที่ $Q_{\text{row}}, K_{\text{row}}, V_{\text{row}}$ คือตัวแทนข้อมูลที่ใช้เพื่อเรียนรู้ความสัมพันธ์กับตัวอย่าง
 H_{row} คือผลลัพธ์หลังผ่าน Row Attention

จากสมการที่ 2.14 ซึ่งใช้สำหรับเรียนรู้ความสัมพันธ์ระหว่างข้อมูลในแนวแถว หรือระหว่างตัวอย่าง (Samples) ต่าง ๆ ภายในชุดข้อมูล โดยนำข้อมูลเข้าสู่กระบวนการ Attention ผ่านองค์ประกอบ Query, Key และ Value แล้วคำนวณน้ำหนักความสนใจเพื่อพิจารณาว่าตัวอย่างใดมีความสัมพันธ์กับตัวอย่างอื่นมากน้อยเพียงใด ผลลัพธ์ที่ได้คือ H_{row} ซึ่งเป็นตัวแทนข้อมูลที่ได้รับการปรับปรุงจากสารสนเทศของตัวอย่างอื่น ๆ และสามารถนำไปใช้ในขั้นตอนการเรียนรู้และการจำแนกประเภทในลำดับถัดไป

2.1.13 แท็บทรานฟอเมอร์

แท็บทรานฟอเมอร์ เป็นสถาปัตยกรรมการเรียนรู้เชิงลึกสำหรับข้อมูลแบบตารางที่พัฒนาต่อยอดจากแนวคิดของทรานฟอเมอร์ โดยใช้กลไกการให้ความสนใจตนเอง เพื่อเรียนรู้ความสัมพันธ์ระหว่างคุณลักษณะเชิงหมวดหมู่ผ่านการแปลงคุณลักษณะเหล่านี้ให้เป็นเวกเตอร์ฝังตัว โครงสร้างของแท็บทรานฟอเมอร์ ประกอบด้วยการฝังตัวของคุณลักษณะเชิงหมวดหมู่เข้าสู่ชั้นทรานฟอเมอร์ จากนั้นผลลัพธ์ที่ได้จะถูกรวมกับคุณลักษณะเชิงตัวเลข และส่งต่อไปยังโครงข่ายประสาทเทียมแบบหลายชั้นเพื่อการทำนายผลลัพธ์ขั้นสุดท้าย [37] ดังที่แสดงในภาพประกอบที่ 2.10



ภาพประกอบที่ 2.10 ภาพโครงสร้างของแท็บทรานฟอเมอร์

จากภาพประกอบที่ 2.10 แสดงให้เห็นสถาปัตยกรรมของแบบจำลอง TabTransformer ซึ่งเป็นแบบจำลองการเรียนรู้เชิงลึกที่ออกแบบมาเพื่อประมวลผลข้อมูลแบบตาราง โดยมีแนวคิดสำคัญในการแยกการประมวลผลระหว่างคุณลักษณะเชิงหมวดหมู่ (Categorical Features) และคุณลักษณะเชิงตัวเลข (Numerical Features) ในขั้นต้น ก่อนนำผลลัพธ์จากทั้งสองประเภทมารวมกันเพื่อสร้างผลการทำนาย โดยสามารถอธิบายลำดับการทำงานได้ดังนี้

ในขั้นตอนแรก ข้อมูลนำเข้าจะถูกแบ่งออกเป็นคุณลักษณะเชิงหมวดหมู่และคุณลักษณะเชิงตัวเลข คุณลักษณะเชิงหมวดหมู่ เช่น เพศ หรือประเภทของข้อมูล จะถูกส่งผ่านชั้น Categorical Embedding เพื่อแปลงค่าของแต่ละหมวดหมู่ให้อยู่ในรูปของเวกเตอร์ตัวเลขที่สามารถนำไปประมวลผลด้วยโครงข่ายประสาทเทียมได้ ขณะที่คุณลักษณะเชิงตัวเลขจะถูกส่งเข้าสู่กระบวนการประมวลผลโดยไม่ผ่านชั้น Embedding ในส่วนของโครงสร้าง Transformer

จากนั้นเวกเตอร์ Embedding ของคุณลักษณะเชิงหมวดหมู่จะถูกส่งเข้าสู่ Transformer Block ซึ่งใช้กลไก Self-Attention เพื่อเรียนรู้ความสัมพันธ์ระหว่างคุณลักษณะเชิงหมวดหมู่แต่ละตัว ทำให้แบบจำลองสามารถพิจารณาบริบทและความสัมพันธ์ระหว่างคุณลักษณะร่วมกันได้ ผลลัพธ์จากกระบวนการดังกล่าวเรียกว่า Contextual Embeddings ซึ่งเป็นตัวแทนของคุณลักษณะเชิงหมวดหมู่ที่ได้รับการปรับปรุงโดยคำนึงถึงความสัมพันธ์กับคุณลักษณะอื่นภายในข้อมูลเดียวกัน

ในขั้นตอนถัดมา Contextual Embeddings ที่ได้จาก Transformer จะถูกนำมารวมกับคุณลักษณะเชิงตัวเลขด้วยกระบวนการ Concatenation เพื่อสร้างเวกเตอร์คุณลักษณะรวม (Feature Vector) ที่ประกอบด้วยสารสนเทศจากทั้งคุณลักษณะเชิงหมวดหมู่และคุณลักษณะเชิงตัวเลข จากนั้นเวกเตอร์ดังกล่าวจะถูกส่งเข้าสู่โครงข่ายประสาทเทียมแบบหลายชั้น (Multi-Layer Perceptron: MLP) เพื่อเรียนรู้รูปแบบและความสัมพันธ์เพิ่มเติมก่อนสร้างผลลัพธ์การทำนาย

ในขั้นตอนสุดท้าย ผลลัพธ์จากชั้น MLP จะถูกส่งเข้าสู่ชั้นเอาต์พุต (Output Layer) เพื่อสร้างค่าทำนายตามลักษณะของปัญหา เช่น การจำแนกข้อมูลออกเป็นกลุ่มปกติและกลุ่มที่มีภาวะความดันโลหิตสูง

สมการแปลงตัวแปรเชิงหมวดหมู่แต่ละตัวเป็นเวกเตอร์ฝัง

$$E_j = \text{Embedding}(x_j), j = 1, 2, \dots, m$$

สมการที่ 2.15 สมการของแท็บทรานฟอเมอร์ (1)

โดยที่ x_j	คือพีเจอร์ตัวที่ j
E_j	คือเวกเตอร์ embedding ของพีเจอร์ j
Embedding	คือฟังก์ชัน embedding
m	คือจำนวนพีเจอร์หนึ่งถึงทั้งหมด

จากสมการที่ 2.15 แสดงกระบวนการแปลงคุณลักษณะเชิงหมวดหมู่แต่ละตัวให้อยู่ในรูปแบบเวกเตอร์ฝังตัว (Embedding Vector) โดยค่าของคุณลักษณะ x_j จะถูกส่งผ่านฟังก์ชัน Embedding เพื่อแปลงค่าที่อยู่ในรูปแบบหมวดหมู่ให้เป็นเวกเตอร์ตัวเลข E_j ซึ่งสามารถนำไปประมวลผลด้วยโครงสร้าง Transformer ได้ การแปลงดังกล่าวช่วยให้แบบจำลองสามารถเรียนรู้ความสัมพันธ์และลักษณะเฉพาะของแต่ละหมวดหมู่ได้อย่างเหมาะสม

กลไกความใส่ใจแบบผลคูณจุดที่ปรับขนาด

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d}}\right)V$$

สมการที่ 2.16 สมการของแท็บทรานฟอเมอร์ (2)

โดยที่ Q	คือสิ่งที่เราต้องการหา
K	คือตัวบอกลักษณะของข้อมูล
V	คือข้อมูลจริงที่ต้องการนำมาใช้
QK^T	คือการคูณเมทริกซ์
$\sqrt{d}$	คือขนาดของเวกเตอร์
softmax	คือแปลงค่าให้เป็น ความน่าจะเป็น
QK^T	คือการคูณเมทริกซ์

จากสมการที่ 2.16 แสดงการทำงานของกลไก Self-Attention ซึ่งใช้สำหรับเรียนรู้ความสัมพันธ์ระหว่างคุณลักษณะเชิงหมวดหมู่ที่ผ่านการแปลงเป็น Embedding แล้ว โดยนำ Query (Q) และ Key (K) มาคำนวณความสัมพันธ์ผ่านการคูณเมทริกซ์ จากนั้นปรับขนาดด้วย d และนำเข้าสู่ฟังก์ชัน softmax เพื่อสร้างค่าน้ำหนักความใส่ใจ ก่อนนำไปคูณกับ Value (V) เพื่อสร้างผลลัพธ์ที่สะท้อนระดับความสำคัญของคุณลักษณะแต่ละตัวต่อคุณลักษณะอื่น ๆ

สมการ Multi-Head Attention

$$\text{MHA}(X) = \text{Concat}(\text{head}_1, \dots, \text{head}_h)W_o$$

สมการที่ 2.17 สมการของแท็บทรานฟอเมอร์ (3)

โดยที่	MHA	คือผลลัพธ์ของ Multi-Head Attention
	$\text{head}_1, \dots, \text{head}_h$	คือ Attention หลายชุด
	Concat	คือการนำผลลัพธ์ของทุก head มาต่อกัน
	W_o	คือเมทริกซ์น้ำหนัก

จากสมการที่ 2.17 แสดงการทำงานของ Multi-Head Attention ซึ่งเป็นการนำกลไก Self-Attention หลายชุดมาประมวลผลข้อมูลพร้อมกัน โดยแต่ละ Head สามารถเรียนรู้ความสัมพันธ์ของคุณลักษณะในมุมมองที่แตกต่างกัน จากนั้นผลลัพธ์จาก Attention แต่ละชุดจะถูกนำมาต่อกันด้วยกระบวนการ Concatenation และแปลงด้วยเมทริกซ์น้ำหนัก W_o เพื่อสร้างผลลัพธ์ของ Multi-Head Attention สำหรับส่งต่อไปยังขั้นตอนถัดไป

สมการ Transformer Encoder Layer

$$Z = \text{LayerNorm}(X + \text{MHA}(X)),$$

สมการที่ 2.18 สมการของแท็บทรานฟอเมอร์ (4)

โดยที่	Z	คือสิ่งที่เราต้องการหา
	LayerNorm	คือการปรับค่าเลเยอร์ให้เป็นมาตรฐาน
	X	คือการนำเข้าเดิมของชั้น
	MHA	คือผลลัพธ์จาก Multi-Head Attention

จากสมการที่ 2.18 แสดงกระบวนการประมวลผลข้อมูลภายใน Transformer Encoder โดยนำข้อมูลนำเข้า X มารวมกับผลลัพธ์จาก Multi-Head Attention ผ่านการเชื่อมต่อแบบ Residual Connection และปรับค่าด้วย Layer Normalization เพื่อให้การเรียนรู้มีเสถียรภาพ จากนั้นผลลัพธ์ Z จะถูกส่งเข้าสู่ Feed Forward Network (FFN) และนำมารวมกับ Z อีกครั้งผ่าน Residual Connection ก่อนทำ Layer Normalization เพื่อสร้างผลลัพธ์ H ซึ่งเป็นตัวแทนคุณลักษณะเชิงหมวดหมู่ที่ผ่านการเรียนรู้จาก Transformer แล้ว

$$H = \text{LayerNorm}(Z + \text{FFN}(Z))$$

สมการที่ 2.19 สมการของแท็บทรานฟอเมอร์ (5)

โดยที่	Z	คือผลลัพธ์จากขั้นตอนก่อนหน้า
	LayerNorm	คือการปรับค่าเลเยอร์ให้เป็นมาตรฐาน
	FFN(Z)	คือ Feed Forward Network
	H	คือผลลัพธ์สุดท้ายของชั้นนี้

สมการ Feature Concatenation

นำ output จาก Transformer (categorical features) มาต่อกับ numerical features

$$F = \text{Concat}(H_{\text{cat}}, X_{\text{num}})$$

สมการที่ 2.20 สมการของ TabTransformer (6)

โดยที่	F	คือฟีเจอร์เวกเตอร์สุดท้าย
	Concat	คือการนำเวกเตอร์มาต่อกัน
	H_{cat}	คือการฝังคุณสมบัติหมวดหมู่
	X_{num}	คือฟีเจอร์ที่เป็นตัวเลข

จากสมการที่ 2.20 แสดงการรวมคุณลักษณะที่ผ่านการประมวลผลจาก Transformer ซึ่งเป็นคุณลักษณะเชิงหมวดหมู่ H_{cat} เข้ากับคุณลักษณะเชิงตัวเลข X_{num} ด้วยกระบวนการ Concatenation เพื่อสร้างเวกเตอร์คุณลักษณะ F ที่ประกอบด้วยสารสนเทศจากทั้งคุณลักษณะเชิงหมวดหมู่และคุณลักษณะเชิงตัวเลข ซึ่งจะถูกนำไปใช้เป็นข้อมูลนำเข้าสำหรับการทำนายของแบบจำลอง

สมการ Prediction Head (MLP)

$$\hat{y} = \text{MLP}(F)$$

สมการที่ 2.21 สมการของแท็บทรานฟอเมอร์ (7)

โดยที่	$\hat{y}$	คือค่ารวมที่โมเดลทำนาย
	MLP	คือโครงข่ายประสาทเทียมหลายชั้น
	F	คือฟีเจอร์เวกเตอร์ที่รวมแล้ว

จากสมการที่ 2.21 แสดงขั้นตอนการสร้างผลลัพธ์การทำนายของ TabTransformer โดยเวกเตอร์คุณลักษณะ F ที่ได้จากการรวมคุณลักษณะเชิงหมวดหมู่และเชิงตัวเลขจะถูกส่งเข้าสู่โครงข่ายประสาทเทียมแบบหลายชั้น (MLP) เพื่อประมวลผลและสร้างค่าทำนายสุดท้าย $\hat{y}$ สำหรับใช้ในการจำแนกประเภทข้อมูล

2.1.14 วิธีการ OPTUNA

Optuna เป็นเฟรมเวิร์กสำหรับการปรับแต่งไฮเปอร์พารามิเตอร์ (Hyperparameter Optimization) ที่ใช้สำหรับค้นหาชุดค่าของไฮเปอร์พารามิเตอร์ที่เหมาะสมสำหรับแบบจำลองการเรียนรู้ของเครื่อง โดยอาศัยแนวคิดการหาค่าเหมาะสมแบบเบย์ (Bayesian Optimization) ร่วมกับขั้นตอนวิธี Tree-structured Parzen Estimator (TPE) ในการสร้างแบบจำลองของพื้นที่การค้นหาและประเมินแนวโน้มของชุดค่าไฮเปอร์พารามิเตอร์ที่มีโอกาสให้ผลลัพธ์ที่ดี จากนั้นจึงเลือกชุดค่าที่เหมาะสมสำหรับการทดลองในรอบ

ถัดไป กระบวนการดังกล่าวช่วยลดจำนวนการทดลองที่ไม่จำเป็นและเพิ่มประสิทธิภาพในการค้นหาค่าที่เหมาะสมเมื่อเปรียบเทียบกับการค้นหาแบบสุ่มหรือการทดลองทุกค่าที่เป็นไปได้ [38]

$$\theta^* = \arg \max_{\theta \in \Theta} f(\theta)$$

สมการที่ 2.22 สมการของ Optuna (1)

โดยที่	θ^*	คือชุดไฮเปอร์พารามิเตอร์ที่เหมาะสมที่สุด
	θ	คือชุดไฮเปอร์พารามิเตอร์ที่นำมาทดลอง
	Θ	คือพื้นที่ของไฮเปอร์พารามิเตอร์ที่กำหนดไว้
	$f(\theta)$	คือค่าประสิทธิภาพที่ได้จากชุดไฮเปอร์พารามิเตอร์ θ
	$\arg \max$	คือการเลือกชุดไฮเปอร์พารามิเตอร์ที่ให้ค่าประสิทธิภาพสูงสุด

ในการวิจัยนี้ Optuna ถูกนำมาใช้สำหรับปรับแต่งไฮเปอร์พารามิเตอร์ของแบบจำลองการเรียนรู้ของเครื่องแต่ละประเภท โดยกำหนดค่า ROC-AUC บนชุดข้อมูลตรวจสอบ (Validation Set) เป็นเกณฑ์ในการประเมินประสิทธิภาพของแต่ละการทดลอง และดำเนินการค้นหาเป็นจำนวนรอบตามที่กำหนด เพื่อให้ได้ชุดไฮเปอร์พารามิเตอร์ที่เหมาะสมก่อนนำแบบจำลองไปประเมินประสิทธิภาพในขั้นตอนต่อไป

2.1.15 การวัดประสิทธิภาพตัวแบบ

ในการประเมินประสิทธิภาพของตัวแบบจำแนกประเภท มักใช้เมตริกหลายตัวเพื่อให้ได้ภาพรวมครบถ้วน ได้แก่ ความแม่นยำ ซึ่งเป็นสัดส่วนของผลการทำนายที่ถูกต้องทั้งหมด ทั้งกรณี ผลทำนายเป็นบวก และผลทำนายเป็นลบ , ความเที่ยงตรงที่เป็นสัดส่วนของผลทำนายที่เป็นบวกที่ถูกต้องจริง , การเรียกคืน และค่าความไว ที่เป็นสัดส่วนของการทำนายที่เป็นผลบวกจริงทั้งหมดที่ตัวแบบสามารถจับได้ และ F1-score ซึ่งเป็นค่าเฉลี่ยเชิงฮาร์โมนิกของความเที่ยงตรง และ ค่าการเรียกคืนนอกจากนี้ใช้ ROC-AUC พื้นที่ใต้โค้ง ROC การใช้เมตริกหลายตัวเหล่านี้สำคัญโดยเฉพาะเมื่อข้อมูลมีความไม่สมดุล ตัวชี้วัดเหล่านี้ทั้งหมดคำนวณจากเมทริกซ์สับสนเป็นพื้นฐาน [39]

- 1) Accuracy แสดงสัดส่วนของการทำนายที่ถูกต้องทั้งหมดเมื่อเทียบกับจำนวนข้อมูลทั้งหมด โดยคำนวณได้จากสมการที่ 2.23

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

สมการที่ 2.23 สมการของ Accuracy

โดยที่	TP	คือจริงและใช่
	TN	คือจริงและไม่ใช่
	FP	คือไม่จริงและใช่

FN คือไม่จริงและใช่

- 2) Precision แสดงความแม่นยำของการทำนายว่ากลุ่มที่ตัวแบบระบุว่าในกลุ่มเสี่ยงนั้นถูกต้องจริงมากน้อยเพียงใด โดยคำนวณได้จากสมการที่ 2.24

$$Precision = \frac{TP}{TP + FP}$$

สมการที่ 2.24 สมการของ Precision

โดยที่ TP คือจริงและใช่

FP คือไม่จริงและใช่

- 3) Recall หรือ Sensitivity แสดงความสามารถของตัวแบบในการตรวจจับผู้ป่วยที่มีความเสี่ยงได้ครบถ้วน โดยคำนวณได้จากสมการที่ 2.25

$$Recall = \frac{TP}{TP + FN}$$

สมการที่ 2.25 สมการของ Recall

โดยที่ TP คือจริงและใช่

FN คือไม่จริงและใช่

- 4) F1-score เป็นค่าที่ใช้ประเมินสมดุลระหว่าง Precision และ Recall โดยเฉพาะในกรณีที่ข้อมูลมีความไม่สมดุล ซึ่งคำนวณได้จากสมการที่ 2.26

$$F1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

สมการที่ 2.26 สมการของ F1-score

โดยที่ $Precision$ คือค่าความเที่ยงตรง

$Recall$ คือค่าการเรียกคืน

- 5) ROC Curve แสดงความสัมพันธ์ระหว่างอัตราการทำนายถูกของกลุ่มเสี่ยง (True Positive Rate) และอัตราการทำนายผิดของกลุ่มไม่เสี่ยง (False Positive Rate) ในเกณฑ์ที่แตกต่างกัน โดยแสดงใน สมการที่ 2.27 และ 2.28

$$TPR = \frac{TP}{TP + FN}$$

สมการที่ 2.27 สมการของ True Positive Rate

โดยที่ TP คือจริงและใช่

TN คือจริงและไม่ใช่

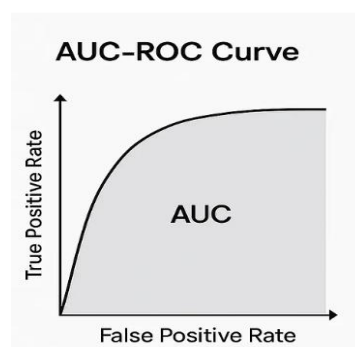
FN คือไม่จริงและใช่

$$FPR = \frac{FP}{FP + TN}$$

สมการที่ 2.28 สมการของ False Positive Rate

โดยที่ TN คือจริงและไม่ใช่
 FP คือไม่จริงและใช่
 FN คือไม่จริงและไม่ใช่

- 6) ค่า AUC (Area Under the Curve) คือพื้นที่ใต้กราฟ ROC ซึ่งใช้เป็นตัวชี้วัดความสามารถโดยรวมของตัวแบบในการแยกแยะระหว่างกลุ่มที่มีและไม่มีความเสี่ยงแสดงในให้เห็นดังภาพประกอบที่ 2.11



ภาพประกอบที่ 2.11 ภาพการวัดประสิทธิภาพแบบ AUC-ROC Curve

2.1.16 การอธิบายผลของตัวแบบ

การอธิบายผลของตัวแบบ เป็นประเด็นสำคัญในการประยุกต์ใช้เทคนิคการเรียนรู้ของเครื่องและการเรียนรู้เชิงลึกในงานด้านการแพทย์ เนื่องจากตัวแบบจำนวนมากมีลักษณะเป็นกล่องดำหรือแบล็ค-บ็อกซ์ ซึ่งแม้จะให้ประสิทธิภาพในการพยากรณ์สูงแต่ยังขาดความสามารถในการอธิบายเหตุผลเบื้องหลังการตัดสินใจของตัวแบบ งานวิจัยนี้จึงประยุกต์ใช้แนวคิด ปัญญาประดิษฐ์ที่อธิบายได้ Explainable AI (XAI) เพื่อเพิ่มความโปร่งใส ความเข้าใจและความน่าเชื่อถือของผลลัพธ์การพยากรณ์ โดยมุ่งเน้นการอธิบายผลเชิงคุณลักษณะภายใต้มาตรฐานเดียวกัน ในการศึกษาได้เลือกใช้เทคนิค SHAP และ LIME ซึ่งเป็นเทคนิคในกลุ่ม Model-Agnostic Explainable AI ที่สามารถประยุกต์ใช้กับตัวแบบที่มีสถาปัตยกรรมแตกต่างกัน ได้แก่ XGBoost, Random forest, LightGBM, CatBoost, MLP, TabNet, FT-Transformer และ TabTransformer เพื่อสนับสนุนการวิเคราะห์เชิงเปรียบเทียบและการตีความผลการทดลอง

1. ชาร์ป

ชาร์ป หรือ SHAP (SHapley Additive exPlanations) เป็นเทคนิคการอธิบายผลของตัวแบบที่มีพื้นฐานจากทฤษฎีเกมแบบร่วมมือ (cooperative game theory) โดยประเมินอิทธิพลของคุณลักษณะแต่ละตัวต่อผลลัพธ์การพยากรณ์ในรูปของค่า Shapley value ซึ่งสะท้อนส่วนร่วมของคุณลักษณะนั้นต่อผลลัพธ์โดยรวมของตัวแบบ [40]

SHAP Value ϕ_i สำหรับ Feature i คำนวณจาก

$$\phi_i(f) = \sum_{S \subseteq F \setminus \{i\}} \frac{|S|! (|F| - |S| - 1)!}{|F|!} (f(S \cup \{i\}) - f(S))$$

สมการที่ 2.29 สมการของ SHAP (1)

โดยที่	F	คือเซตของ Features ทั้งหมด
	S	คือเซตย่อยของ features ที่ไม่รวม Feature i
	$f_S(x_S)$	คือค่าทำนายเมื่อใช้เฉพาะ Features ใน S
	$S \subseteq F \setminus \{i\}$	คือทุกสับเซตของฟีเจอร์ที่ไม่มีฟีเจอร์ i
	$\frac{ S ! (F - S - 1)!}{ F !}$	คือค่าน้ำหนัก
	$f_{S \cup \{i\}}(x_{S \cup \{i\}})$	คือค่าทำนายเมื่อ มี feature i

$$f(x) = \phi_0 + \sum_{i=1}^M \phi_i$$

สมการที่ 2.30 สมการของ SHAP (2)

โดยที่	ϕ_i	คือค่า SHAP ของคุณลักษณะลำดับที่ i
	ϕ_0	คือค่า base value
	$f(x)$	คือค่าที่โมเดลทำนาย
	$\sum_{i=1}^M \phi_i$	คือการรวมผลกระทบของทุกฟีเจอร์

2. โล้ม

โล้ม หรือ LIME (Local Interpretable Model-agnostic Explanations) LIME (Local Interpretable Model-agnostic Explanations) เป็นเทคนิคการอธิบายผลลัพธ์ของตัวแบบที่ไม่ขึ้นกับโครงสร้างภายในของตัวแบบ และมุ่งเน้นการอธิบายผลในระดับเฉพาะกรณี (local explanation) โดยประมาณพฤติกรรมของตัวแบบพยากรณ์หลักด้วยตัวแบบเชิงอธิบายที่มีความเรียบง่ายและเข้าใจง่าย ในบริเวณใกล้เคียงกับจุดข้อมูลที่สนใจ แนวคิดหลักของ LIME คือการสร้างข้อมูลใหม่จากการรบกวน (perturbation) รอบจุดข้อมูลดังกล่าว และใช้ฟังก์ชันถ่วงน้ำหนักเพื่อให้ความสำคัญกับข้อมูลที่อยู่ใกล้จุดเป้าหมายมากกว่า [41]

กระบวนการของโล้ม ได้ดังสมการ

$$g^* = \arg \min_{g \in \mathcal{G}} [L(f, g, \pi_x) + \Omega(g)]$$

สมการที่ 2.31 สมการของ LIME

โดยที่ g^*	คือตัวแบบอธิบายที่เหมาะสมที่สุด
g	คือตัวแบบอธิบาย
G	คือเซตของตัวแบบอธิบายที่เข้าใจง่าย
f	คือตัวแบบพยากรณ์หลัก
$L(f, g, \pi_x)$	คือฟังก์ชันวัดความใกล้เคียงระหว่าง f และ g รอบจุดข้อมูล x
π_x	คือฟังก์ชันถ่วงน้ำหนักตามความใกล้เคียงของข้อมูลกับ
$\Omega(g)$	คือ regularization เพื่อควบคุมความซับซ้อนของตัวแบบอธิบาย

3. ขอบเขตและข้อจำกัดของปัญญาประดิษฐ์ที่อธิบายได้

แม้ว่าเทคนิคปัญญาประดิษฐ์ที่อธิบายได้ เช่น ชาร์ป และ โลม จะช่วยเพิ่มความโปร่งใสและความสามารถในการตีความการทำงานของตัวแบบการเรียนรู้ของเครื่อง แต่การอธิบายผลดังกล่าวยังคงมีข้อจำกัด โดย ชาร์ป และ โลม สะท้อนเพียงความสัมพันธ์เชิงสถิติระหว่างคุณลักษณะกับผลลัพธ์การพยากรณ์ของตัวแบบเท่านั้น มิได้บ่งชี้ถึงความเป็นเหตุและผล ทางคลินิกโดยตรงและงานวิจัยด้าน ปัญญาประดิษฐ์ที่อธิบาย ในบริบททางการแพทย์ชี้ให้เห็นว่า การอธิบายผลของตัวแบบอาจส่งผลแตกต่างกันตามกลุ่มผู้ใช้งาน โดยสำหรับผู้เชี่ยวชาญทางการแพทย์ การอธิบายผลสามารถช่วยสนับสนุนความเข้าใจ ตรวจสอบความสมเหตุสมผลของผลลัพธ์ และเพิ่มความเชื่อมั่นต่อระบบปัญญาประดิษฐ์ ขณะที่ในกลุ่มบุคคลทั่วไปหรือผู้ที่ขาดความรู้เชิงคลินิก การอธิบายผลอาจนำไปสู่ความเข้าใจผิด ความเชื่อมั่นเกินจริง หรือการตีความผลลัพธ์ในเชิงเหตุและผลที่ไม่ถูกต้อง [42]

งานวิจัยนี้จึงประยุกต์ใช้เทคนิคปัญญาประดิษฐ์ที่อธิบายได้เพื่อสนับสนุนการวิเคราะห์เชิงเปรียบเทียบและกาตีความผลการทดลองภายใต้มาตรฐานเดียวกันเท่านั้น มิได้มีวัตถุประสงค์เพื่อใช้เป็นเครื่องมือสนับสนุนการตัดสินใจแทนแพทย์โดยตรง

2.2 งานวิจัยที่เกี่ยวข้อง

งานวิจัยของ Andishgar และคณะ (2024) ได้ศึกษาปัจจัยเสี่ยงของโรคความดันโลหิตสูง ซึ่งพบว่ามี ความแตกต่างกันในแต่ละภูมิภาค โดยมีวัตถุประสงค์เพื่อพัฒนาแบบจำลองสำหรับพยากรณ์ความเสี่ยงของการเกิดโรคความดันโลหิตสูงในระยะเวลา 5 ปี ในกลุ่มประชากรชนบทของประเทศอิหร่าน การศึกษานี้ดำเนินการในลักษณะการติดตามผลระยะยาวโดยใช้ข้อมูลจาก Fasa Adults Cohort Study และประยุกต์ใช้อัลกอริทึมการเรียนรู้ของเครื่องแบบมีผู้สอน จำนวน 10 รูปแบบ อาทิ Logistic Regression (LR), Support Vector Machine (SVM), Random Forest (RF) และ Light Gradient Boosting Machine (LGBM) พร้อมทั้งประยุกต์ใช้เทคนิค SHapley Additive exPlanations (SHAP) เพื่อคัดเลือกคุณลักษณะที่สำคัญ ผลการศึกษาพบว่าแบบจำลอง LGBM ซึ่งใช้คุณลักษณะสำคัญ 30 อันดับแรก ให้ประสิทธิภาพดีที่สุด โดยมีค่า Area Under the Curve (AUC) เท่ากับ 0.67 และปัจจัยพยากรณ์หลักที่ระบุได้ ได้แก่ ค่าความดันโลหิตซิสโตลิก (SBP) ตั้งต้น เพศ และอัตราส่วนรอบเอวต่อสะโพก (WHR) [43] สอดคล้องกับแนวคิดในการพัฒนาแบบจำลองที่มีทั้ง

ความแม่นยำและความสามารถในการอธิบายผลงานวิจัยของ Asal (2025) ได้นำเสนอเฟรมเวิร์กการเรียนรู้เชิงลึกที่ใช้สถาปัตยกรรม Transformer เพื่อตอบสนองความต้องการดังกล่าวในการตรวจจับภาวะหัวใจหยุดเต้น โดยใช้ข้อมูลทางคลินิกในรูปแบบตาราง การศึกษานี้ได้เปรียบเทียบประสิทธิภาพระหว่างแบบจำลอง TabTransformer และ FT-Transformer พร้อมทั้งใช้เทคนิค SHAP และ LIME เพื่ออธิบายกระบวนการตัดสินใจของแบบจำลอง ผลการทดลองแสดงให้เห็นว่าแบบจำลอง FT-Transformer มีความแม่นยำสูงถึงร้อยละ 98.1 ซึ่งเหนือกว่าแบบจำลอง TabTransformer ที่มีความแม่นยำร้อยละ 84.5 อย่างมีนัยสำคัญ โดยคุณลักษณะที่ส่งผลต่อการตัดสินใจของแบบจำลองมากที่สุด ได้แก่ จำนวนหลอดเลือดหลัก (ca) ผลการทดสอบความเครียดด้วยแทลเลียม (thal) และประเภทของอาการเจ็บหน้าอก (cp) [44]

งานวิจัยของ Upadhyayula และ Pothugunta (2025) ได้ทำการศึกษาเปรียบเทียบประสิทธิภาพระหว่างแบบจำลอง Transformer กับแบบจำลองการเรียนรู้ของเครื่องแบบดั้งเดิมในการพยากรณ์โรคหัวใจและหลอดเลือด โดยเฉพาะอย่างยิ่งภายใต้สถานการณ์ที่ข้อมูลมีความไม่สมดุล การศึกษานี้เปรียบเทียบแบบจำลองกลุ่มทรานส์ฟอร์เมอร์ ได้แก่ FT-Transformer, SAINT และ TabNet กับอัลกอริทึมการเรียนรู้ของเครื่องแบบดั้งเดิม ได้แก่ XGBoost และ Logistic Regression บนชุดข้อมูลสาธารณะ 3 ชุดที่มีขนาดและระดับความซับซ้อนแตกต่างกัน (UCI, Framingham และ Kaggle) ผลการวิจัยชี้ให้เห็นว่าประสิทธิภาพของแบบจำลองมีความสัมพันธ์อย่างมีนัยสำคัญกับลักษณะของชุดข้อมูลที่ใช้ โดยแบบจำลอง FT-Transformer และ Random Forest ให้ค่า AUC 0.99 บนชุดข้อมูลที่มีความสมดุล (UCI) แต่เมื่อทดสอบกับชุดข้อมูลที่ไม่สมดุล (Framingham) แบบจำลองส่วนใหญ่กลับมีความไว ต่ำ แม้จะผ่านการปรับสมดุลข้อมูลแล้วก็ตาม โดยมีเพียงแบบจำลอง FT-Transformer ที่ยังคงให้ผลลัพธ์ที่สมดุลที่สุด [5] นอกจากนี้ ยังมีงานวิจัยของ Srinivas และคณะ (2025) ที่นำเสนอระบบการเรียนรู้เชิงลึกโดยใช้สถาปัตยกรรม TabNet ร่วมกับกระบวนการเตรียมข้อมูลขั้นสูงแบบจำลองที่พัฒนาขึ้นมีความแม่นยำร้อยละ 73 และมีค่า ROC-AUC เท่ากับ 0.73 โดยปัจจัยพยากรณ์ที่สำคัญที่สุดที่ระบุได้ ได้แก่ ระดับคอเลสเตอรอล ความดันโลหิตซิสโตลิก และอายุ สอดคล้องกับประเด็นความเชื่อมโยงระหว่างภาวะความดันโลหิตสูงกับความเสี่ยงต่อโรคหัวใจ งานวิจัยของ Si และคณะ (2025) ได้ทำการศึกษาเพื่อระบุความเสี่ยงของการเกิดโรคหัวใจในกลุ่มผู้ป่วยสูงอายุที่มีภาวะความดันโลหิตสูง โดยใช้ข้อมูลจากการศึกษาติดตามผลระยะยาว 7 ปี China Health and Retirement Longitudinal Study: CHARLS และพัฒนาแบบจำลองด้วยวิธี Logistic Regression, XGBoost และ Deep Neural Network ผลการศึกษาพบว่าร้อยละ 26.03 ของกลุ่มตัวอย่างเกิดโรคหัวใจภายในระยะเวลาที่ติดตาม โดยแบบจำลอง DNN ให้ค่า ROC-AUC สูงที่สุดที่ 0.67 ในขณะที่ Logistic Regression มีความสามารถในการเทียบมาตรฐานที่ดีที่สุด และ XGBoost ให้ประโยชน์ทางคลินิกที่ชัดเจนที่สุด [45]

นอกจากนี้งานวิจัยของ Xu และคณะ (2025) ได้กล่าวถึงประเด็นความทึบของอัลกอริทึมปัญญาประดิษฐ์ (AI) ในบริบทการดูแลสุขภาพ ซึ่งมีความเกี่ยวข้องโดยตรงกับการใช้เทคนิคปัญญาประดิษฐ์ที่อธิบายได้ ในงานวิจัยข้างต้นหลายชิ้น โดยชี้ให้เห็นว่าแม้ XAI จะถูกนำมาใช้เพื่อเพิ่มความเข้าใจในกระบวนการตัดสินใจของ AI แต่ก็มีความกังวลว่าอาจนำไปสู่การพึ่งพาหรือความเอนเอียงที่มากเกินไปได้เช่นกัน คณะผู้วิจัยได้ทำการทดลองขนาดใหญ่สองชุดเพื่อเปรียบเทียบผลกระทบของ XAI โดยเฉพาะคำอธิบายที่สร้างจาก Multimodal

Large Language Models ที่มีต่อประสิทธิภาพในการวินิจฉัยโรค ระหว่างกลุ่มบุคคลทั่วไป (จำนวน 623 คน) และแพทย์ปฐมภูมิ (จำนวน 153 คน) โดยใช้แบบจำลอง AI สำหรับการวินิจฉัยโรคผิวหนังเป็นกรณีศึกษา ผลการวิจัยพบว่าคำอธิบายที่สร้างจาก LLM มีลักษณะเป็น "ดาบสองคม" กล่าวคือ กลุ่มผู้ใช้ทั่วไปแสดงให้เห็นว่าเชื่อระบบอัตโนมัติมากเกินไปได้รับคำอธิบายดังกล่าว ในขณะที่แพทย์ผู้มีความรู้ทางการแพทย์ยังคงรักษาความยืดหยุ่นในการตัดสินใจและได้รับประโยชน์จากคำอธิบายโดยไม่คำนึงถึงความถูกต้องของผลลัพธ์จาก AI แต่อย่างใด [42]

ตารางที่ 2.2 ตารางสรุปงานวิจัยที่เกี่ยวข้อง

งานวิจัย	วัตถุประสงค์	วิธีการ/อัลกอริทึมที่ใช้	ผลการศึกษา
Andishgar และคณะ (2024)	เพื่อพัฒนาแบบจำลองสำหรับพยากรณ์ความเสี่ยงของการเกิดโรคความดันโลหิตสูงในระยะเวลา 5 ปี ในกลุ่มประชากรชนบทของประเทศอิหร่าน	อัลกอริทึมแบบมีผู้สอน 10 รูปแบบ เช่น Logistic Regression (LR), SVM, Random Forest (RF) และ LightGBM พร้อมใช้เทคนิค SHAP คัดเลือกคุณลักษณะ	แบบจำลอง LGBM (ใช้คุณลักษณะสำคัญ 30 อันดับแรก) ให้ประสิทธิภาพดีที่สุด (AUC = 0.67) ปัจจัยพยากรณ์หลักคือ ความดันซิสโตลิก เพศ และอัตราส่วนรอบเอวต่อสะโพก
Asal (2025)	เพื่อตรวจจับภาวะหัวใจหยุดเต้นโดยใช้ข้อมูลทางคลินิกในรูปแบบตาราง	เฟรมเวิร์กการเรียนรู้เชิงลึก สถาปัตยกรรม Transformer ได้แก่ TabTransformer และ FT-Transformer พร้อมใช้ SHAP และ LIME อธิบายผล	FT-Transformer มีความแม่นยำสูงสุด 98.1% (TabTransformer 84.5%) ปัจจัยหลักคือ จำนวนหลอดเลือดหลัก (ca) ผลทดสอบความเครียดด้วยแทลเลียม (thal) และประเภทอาการเจ็บหน้าอก (cp)
Upadhyayula และ Pothugunta (2025)	เพื่อศึกษาเปรียบเทียบประสิทธิภาพระหว่างแบบจำลอง Transformer กับ ML แบบดั้งเดิมในการพยากรณ์โรคหัวใจและหลอดเลือดภายใต้ข้อมูลที่ไม่สมดุล	กลุ่ม Transformer (FT-Transformer, SAINT, TabNet) เปรียบเทียบกับกลุ่มดั้งเดิม (XGBoost, LR) ทดสอบบนชุดข้อมูลสาธารณะ	FT-Transformer และ Random Forest ให้ AUC 0.99 บนข้อมูลสมดุล แต่เมื่อข้อมูลไม่สมดุล FT-Transformer เป็นแบบจำลองเดียวที่ยังคงให้ผลลัพธ์ที่สมดุลดีที่สุด
Srinivas และคณะ (2025)	เพื่อนำเสนอระบบการเรียนรู้เชิงลึกเพื่อการทำนายโรค	ใช้สถาปัตยกรรม TabNet ร่วมกับกระบวนการเตรียมข้อมูลขั้นสูง	มีความแม่นยำ 73% (ROC-AUC 0.73) ปัจจัยพยากรณ์สำคัญ ได้แก่ ระดับคอเลสเตอรอล ความดันโลหิตซิสโตลิก และอายุ
Si และคณะ (2025)	เพื่อระบุความเสี่ยงของการเกิดโรคหัวใจในกลุ่มผู้ป่วยสูงอายุที่มีภาวะความดันโลหิตสูง	Logistic Regression, XGBoost และ Deep Neural Network (DNN) วิเคราะห์ข้อมูลจากติดตามผลระยะยาว 7 ปี (CHARLS)	พบผู้ป่วยเกิดโรคหัวใจ 26.03% โดยแบบจำลอง DNN ให้ค่า ROC-AUC สูงสุด (0.67) LR เทียบมาตรฐานดีที่สุด และ XGBoost ให้ประโยชน์ทางคลินิกชัดเจนสุด

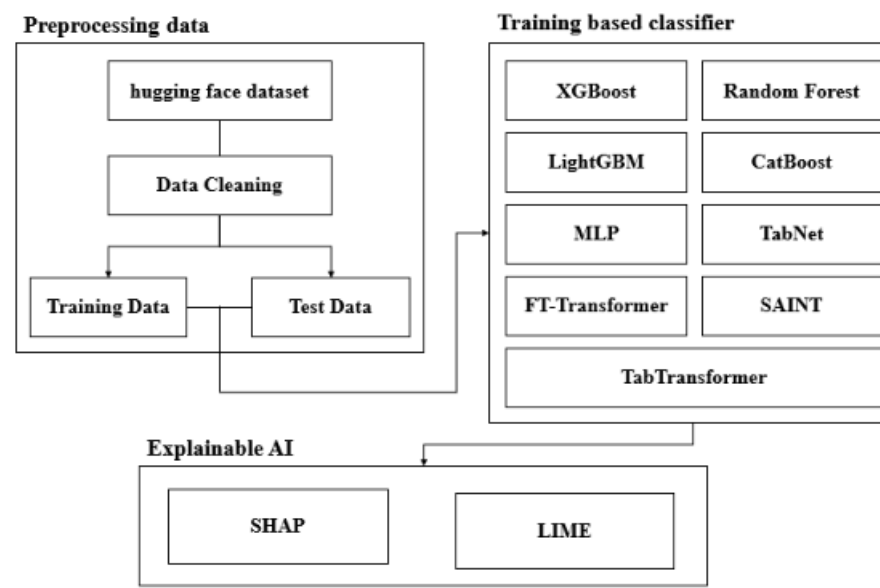
Xu และคณะ (2025)	เพื่อเปรียบเทียบผลกระทบ ของเทคนิคปัญญาประดิษฐ์ที่ อธิบายได้ (XAI) ต่อ ประสิทธิภาพในการ วินิจฉัยโรคระหว่างบุคคล ทั่วไปและแพทย์ปฐมภูมิ	ใช้แบบจำลอง AI วินิจฉัยโรค ผิวหนังร่วมกับคำอธิบายที่สร้าง จาก Multimodal Large Language Models (LLM)	คำอธิบายจาก LLM เป็น "ดาบสอง คม" ผู้ใช้ทั่วไปเชื่อระบบอัตโนมัติมาก เกินไป ขณะที่แพทย์ผู้มีประสบการณ์ คงความยืดหยุ่นในการตัดสินใจและได้ ประโยชน์จาก XAI มากกว่า
------------------------------------	---	---	---

จากตารางที่ 2.2 พบว่า งานวิจัยที่เกี่ยวข้องกับการพยากรณ์ความเสี่ยงโรคความดันโลหิตสูงและโรคหัวใจ และหลอดเลือด นิยมประยุกต์ใช้อัลกอริทึมการเรียนรู้ของเครื่องทั้งแบบดั้งเดิม (เช่น Logistic Regression, Random Forest, XGBoost, LightGBM) และสถาปัตยกรรมการเรียนรู้เชิงลึก (เช่น DNN, TabNet, FT-Transformer) เพื่อเพิ่มประสิทธิภาพในการจำแนกข้อมูล นอกจากนี้การศึกษานี้ยังควบคู่ไปกับการประยุกต์ใช้เทคนิคปัญญาประดิษฐ์ที่อธิบายได้ (Explainable AI หรือ XAI) เช่น SHAP และ LIME เพื่ออธิบายกระบวนการตัดสินใจของแบบจำลองและเพิ่มความน่าเชื่อถือในการนำไปใช้ในบริบททางการแพทย์

บทที่ 3

แผนการดำเนินงานวิจัย

งานวิจัยนี้มุ่งพัฒนาและประเมินแบบจำลองการเรียนรู้ของเครื่องสำหรับทำนายความเสี่ยงโรคความดันโลหิตสูง โดยไม่ได้มีวัตถุประสงค์เพื่อวินิจฉัยแทนแพทย์ ประยุกต์ใช้เทคนิคการเรียนรู้เชิงลึกและแบบจำลองสำหรับข้อมูลเชิงตาราง ครอบคลุมขั้นตอนตั้งแต่การรวบรวมและเตรียมข้อมูล การพัฒนาและปรับแต่งแบบจำลอง การประเมินประสิทธิภาพ ไปจนถึงการอธิบายผลลัพธ์ด้วยเทคนิคปัญญาประดิษฐ์เชิงอธิบาย ดังแสดงในภาพที่ 3.1



ภาพประกอบที่ 3.1 ภาพรวมของกระบวนการดำเนินงานวิจัย

3.1 กรอบแนวคิดการวิจัยและสภาพแวดล้อมการทดลอง

จากภาพประกอบที่ 3.1 กรอบแนวคิดการวิจัยของงานวิจัยนี้แบ่งออกเป็นขั้นตอนหลักตามลำดับ ได้แก่ การรวบรวมข้อมูล การวิเคราะห์ข้อมูลเชิงสำรวจ การทำความสะอาดและเตรียมข้อมูล การพัฒนาและปรับแต่งแบบจำลองทั้งเก้าประเภท การประเมินประสิทธิภาพภายใต้กรอบ Stratified 5-Fold Cross-Validation และการวิเคราะห์ความสามารถในการอธิบายผลด้วยเทคนิค Explainable AI รายละเอียดของแต่ละขั้นตอนในกระบวนการทดลองสรุปได้ดังตารางที่ 3.1

ตารางที่ 3.1 ขั้นตอนการดำเนินงานทดลอง

ลำดับ	ขั้นตอน	รายละเอียดโดยสรุป
1	Data Acquisition	รวบรวมชุดข้อมูล African Hypertension & CVD Dataset จาก HuggingFace
2	Exploratory Data Analysis	ตรวจสอบการกระจายตัวของตัวแปรเป้าหมายและค่าสหสัมพันธ์ระหว่างตัวแปร

3	Data Cleaning	ตัดตัวแปรที่มีความเสี่ยงต่อการรั่วไหลของข้อมูล จัดการค่าว่างและแปลงตัวแปรบูลีน
4	Data Preparation	แบ่งชุดข้อมูล Train/Test แบบ Stratified, เข้ารหัสตัวแปร, ปรับมาตรฐานตัวแปรเชิงตัวเลข
5	Model Development & Tuning	ฝึกและปรับจูนแบบจำลองทั้ง 9 ประเภทด้วย Optuna ภายใต้ Stratified 5-Fold CV
6	Model Evaluation	ประเมินผลบนชุดทดสอบด้วย Accuracy, Precision, Recall, F1, ROC-AUC, Gini, Brier Score
7	Explainable AI	วิเคราะห์ SHAP, LIME

ตารางที่ 3.2 สภาพแวดล้อมด้านซอฟต์แวร์

องค์ประกอบ	รายละเอียด
ภาษาโปรแกรม	Python
แพลตฟอร์มรันโค้ด	Google Colaboratory
การจัดการข้อมูล	pandas, numpy
การเตรียมข้อมูล	scikit-learn, imbalanced-learn
โมเดลกลุ่มต้นไม้แบบรวม	xgboost, scikit-learn (RandomForest), lightgbm, catboost
โมเดลกลุ่มการเรียนรู้เชิงลึก	PyTorch, pytorch-tabnet, rtdl_revisiting_models (FT-Transformer), tab_transformer_pytorch
การปรับแต่งไฮเปอร์พารามิเตอร์	optuna
การอธิบายผลแบบจำลอง	shap, lime
การแสดงผลเชิงภาพ	matplotlib, seaborn

ตารางที่ 3.2 นี้สรุปภาษาโปรแกรม แพลตฟอร์ม และไลบรารีที่ใช้ในการดำเนินการวิจัย ครอบคลุมการจัดการข้อมูล การเตรียมข้อมูล การพัฒนาแบบจำลอง การปรับแต่งไฮเปอร์พารามิเตอร์ การอธิบายผลการทำงานของแบบจำลอง และการแสดงผลเชิงภาพ เพื่อสนับสนุนการดำเนินการทดลองและการประเมินประสิทธิภาพของแบบจำลองอย่างเป็นระบบ

ตารางที่ 3.3 สภาพแวดล้อมด้านฮาร์ดแวร์

องค์ประกอบ	รายละเอียด
สภาพแวดล้อมการรันโค้ดหลัก	Google Colaboratory (Cloud-based Notebook)
หน่วยประมวลผล	CPU (การฝึกแบบจำลองเชิงลึกในการทดลองนี้ดำเนินการบน CPU ไม่ได้ใช้ GPU)
อุปกรณ์สำหรับพัฒนาและทบทวนวรรณกรรม	แล็ปท็อป CPU Intel Core i5, RAM 8 GB, ระบบปฏิบัติการ Windows 11

อุปกรณ์จัดเก็บข้อมูล	Solid State Drive (SSD) และ Google Drive สำหรับการบันทึกจุดตรวจ (Checkpoint)
----------------------	---

จากตารางที่ 3.3 การฝึกแบบจำลองเชิงลึกในงานวิจัยนี้ดำเนินการบนหน่วยประมวลผลกลาง (CPU) ของแพลตฟอร์ม Google Colaboratory โดยไม่ได้ใช้หน่วยประมวลผลกราฟิก (GPU) ซึ่งเป็นปัจจัยสำคัญที่ส่งผลต่อระยะเวลาการฝึกแบบจำลอง โดยเฉพาะสถาปัตยกรรมทรานส์ฟอร์มเมอร์ที่มีความซับซ้อนสูง ดังจะเห็นได้จากผลเปรียบเทียบเวลาการฝึกของแต่ละแบบจำลองในบทที่ 4

ตารางที่ 3.4 การกำหนดค่าการตรวจสอบไขว้

รายการ	ค่าที่กำหนด
วิธีการแบ่งข้อมูล	Stratified
จำนวน Fold	5
Random Seed	42
ตัวชี้วัดเป้าหมายสำหรับ Optuna	ROC-AUC
จำนวน Trials ต่อแบบจำลอง	30

ตารางที่ 3.4 จะสรุปการตั้งค่าการฝึกสอนและประเมินแบบจำลองที่ใช้ในการวิจัย โดยใช้ Stratified 5-Fold Cross-Validation ร่วมกับการปรับแต่งไฮเปอร์พารามิเตอร์ด้วย Optuna

3.2 การรวบรวมชุดข้อมูลที่ใช้ในการวิจัย

งานวิจัยนี้ใช้ชุดข้อมูล African Hypertension and Cardiovascular Disease Dataset ที่เผยแพร่สาธารณะผ่านแพลตฟอร์ม HuggingFace โดยเลือกชุดข้อมูล extra_large_10000 จำนวน 10,000 ระเบียบ ซึ่งเป็นข้อมูลสังเคราะห์ภายใต้สัญญาอนุญาต CC BY 4.0 และไม่มีข้อมูลที่สามารถระบุตัวตนของบุคคลได้

3.3 การวิเคราะห์ข้อมูลเชิงสำรวจ

3.3.1 ทำความเข้าใจข้อมูล

ตรวจสอบจำนวนตัวอย่างและตัวแปร จำแนกประเภทข้อมูล ได้ในตารางที่ 3.5

ตารางที่ 3.5 รายละเอียดของตัวแปรทั้งหมดในชุดข้อมูล

Feature	Description	Type
patient_id	รหัสประจำตัวผู้ป่วย	Identifier
age	อายุของผู้ป่วย (ปี)	Numerical
sex	เพศของผู้ป่วย (Male, Female)	Categorical
residence	พื้นที่อยู่อาศัย (Urban, Rural)	Categorical
bmi	ดัชนีมวลกายของผู้ป่วย	Numerical
family_history_hypertension	ประวัติครอบครัวเป็นโรคความดันโลหิตสูง	Binary

diabetes	สถานะโรคเบาหวาน	Binary
smoking	พฤติกรรมการสูบบุหรี่	Binary
alcohol_heavy	การดื่มแอลกอฮอล์ในปริมาณมาก	Binary
physically_active	การออกกำลังกายสม่ำเสมอ	Binary
high_salt_diet	การบริโภคอาหารเค็ม	Binary
hypertension_status	สถานะความดันโลหิต (Normal / Hypertensive)	Target
systolic_bp_mmhg	ความดันโลหิตตัวบน (mmHg)	Numerical
diastolic_bp_mmhg	ความดันโลหิตตัวล่าง (mmHg)	Numerical
diagnosed	การได้รับการวินิจฉัยโรคความดันโลหิตสูง	Binary
on_medication	การรับประทานยาควบคุมความดันโลหิต	Binary
medication_adherence	ความสม่ำเสมอในการใช้ยา	Categorical
bp_controlled	การควบคุมความดันโลหิตได้หรือไม่	Binary
stroke_history	ประวัติโรคหลอดเลือดสมอง	Binary
myocardial_infarction	ประวัติกล้ามเนื้อหัวใจตาย	Binary
heart_failure	ประวัติภาวะหัวใจล้มเหลว	Binary
total_cholesterol_mg_dl	คอเลสเตอรอลรวมในเลือด	Numerical
ldl_mg_dl	ระดับ LDL Cholesterol	Numerical
hdl_mg_dl	ระดับ HDL Cholesterol	Numerical
creatinine_mg_dl	ระดับ Creatinine ในเลือด	Numerical

จากข้อมูลที่แสดงในตารางที่ 3.5 ผู้วิจัยได้พิจารณาตัดตัวแปรบางส่วนที่ไม่เกี่ยวข้องโดยตรงกับการทำนาย หรือเป็นตัวแปรที่อาจก่อให้เกิดการรั่วไหลของข้อมูล ออกจากกระบวนการวิเคราะห์ เพื่อให้ตัวแบบสามารถเรียนรู้ความสัมพันธ์ของข้อมูลได้อย่างเหมาะสมและลดความเอนเอียงของผลลัพธ์ โดยตัวแปรที่ใช้ในการวิเคราะห์ประกอบด้วยข้อมูลตามตารางที่ 3.6

3.3.2 สํารวจคุณภาพข้อมูล

ตรวจสอบข้อมูลที่สูญหายตรวจสอบค่าผิดปกติ วิเคราะห์ความผิดปกติของข้อมูลเพื่อลดความลำเอียงได้ดังตารางที่ 3.6

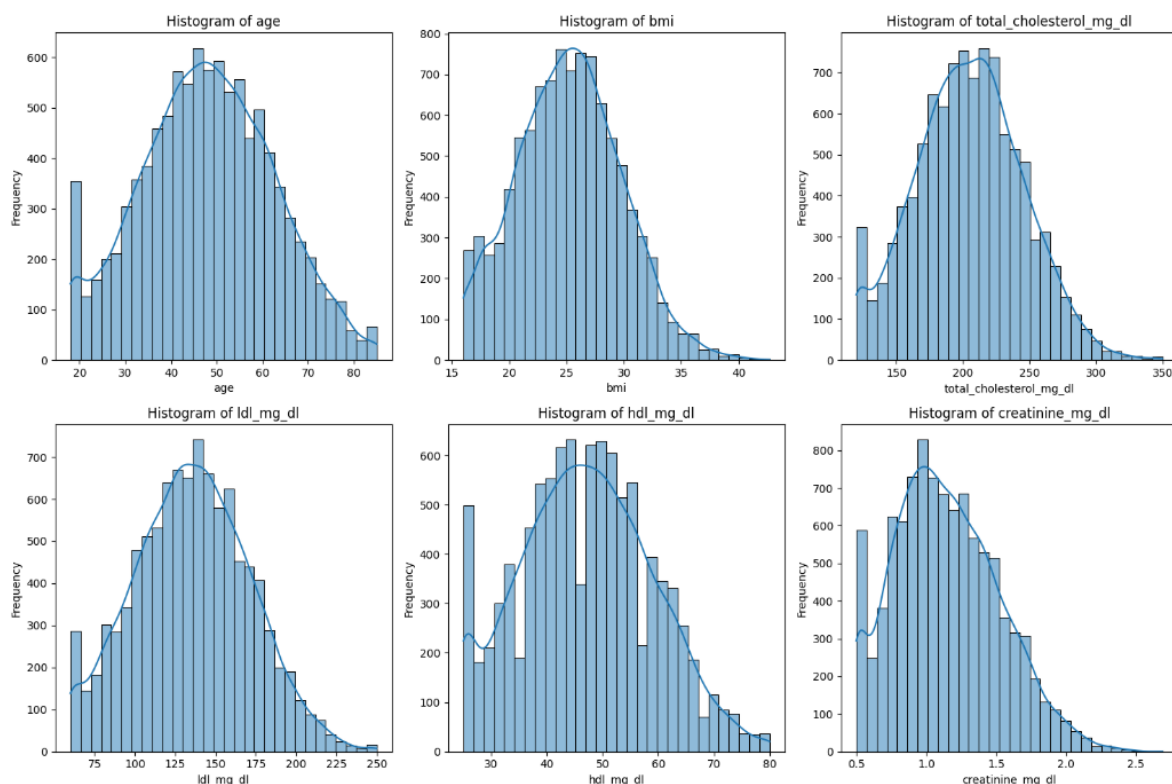
ตารางที่ 3.6 ตารางสำรวจค่าที่อยู่นอกช่วงปกติ

Feature	Q1	Q3	IQR	Lower Bound	Upper Bound	Outliers	%
age	37.90	58.30	20.40	7.30	88.90	0	0.00
bmi	22.00	28.30	6.30	12.55	37.75	75	0.44
total_cholesterol	177.00	232.00	55.00	94.50	314.50	70	0.41
ldl_mg_dl	110.00	160.00	50.00	35.00	235.00	41	0.24
hdl_mg_dl	39.00	55.00	16.00	15.00	79.00	57	0.34
creatinine_mg_dl	0.87	1.40	0.53	0.08	2.20	78	0.46

จากตารางที่ 3.6 พบว่าตัวแปรบางตัวมีค่าที่อยู่นอกช่วงปกติ เช่น BMI, Cholesterol, LDL, HDL และ Creatinine อย่างไรก็ตาม สัดส่วนของ Outliers ในแต่ละตัวแปรมีค่า น้อยกว่า 1% ของข้อมูลทั้งหมด ซึ่งถือว่าอยู่ในระดับต่ำ ดังนั้นในการศึกษาครั้งนี้จึง ไม่ได้ทำการลบ Outliers ออกจากชุดข้อมูล เพื่อคงลักษณะข้อมูลสำหรับการฝึกโมเดล

3.2.3 การวิเคราะห์ตัวแปรเดียว

เพื่อศึกษาลักษณะการกระจายของตัวแปรเชิงตัวเลขในชุดข้อมูลและตรวจสอบความผิดปกติของข้อมูลเบื้องต้น ได้ทำการวิเคราะห์การแจกแจงความถี่ของแต่ละตัวแปรด้วยแผนภูมิฮิสโตแกรม (Histogram) ดังแสดงในรูปที่ 3.2



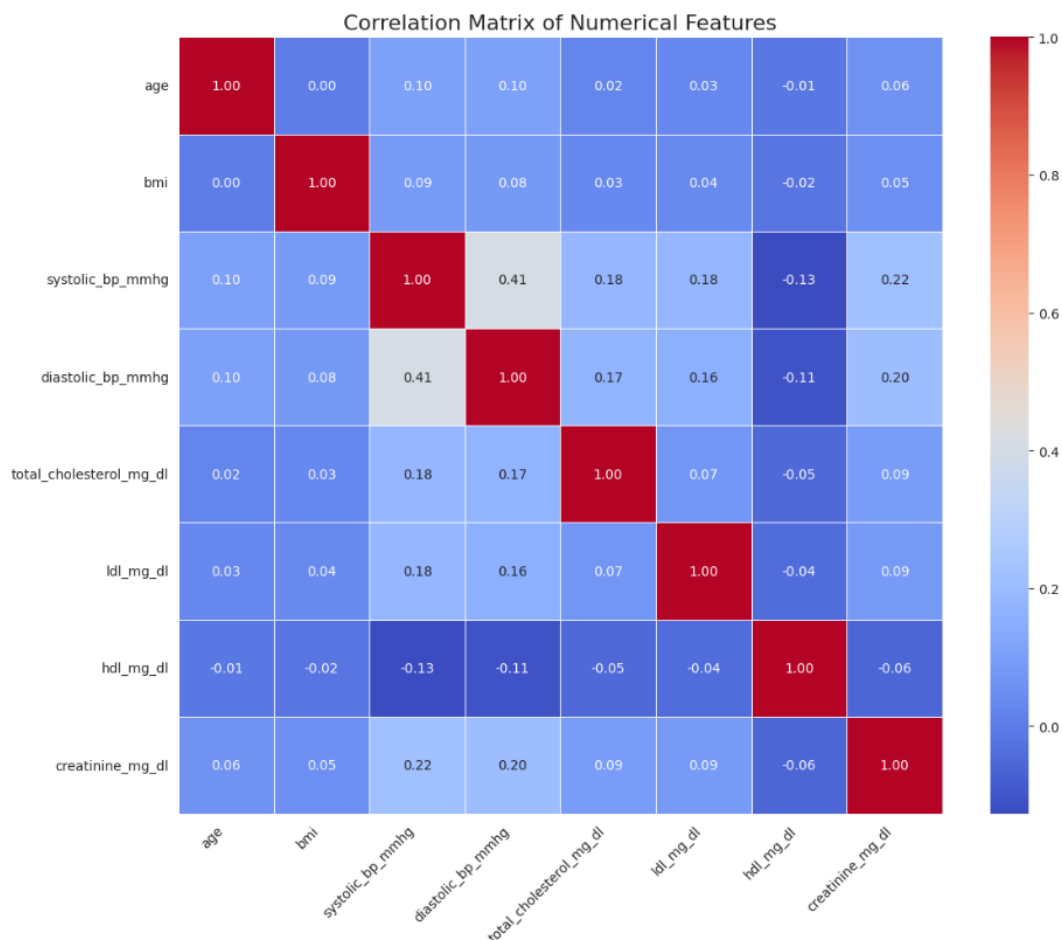
ภาพประกอบที่ 3.2 ฮิสโตแกรมแสดงการกระจายของตัวแปรเชิงตัวเลข

จากรูปที่ 3.2 พบว่า ตัวแปรเชิงตัวเลขส่วนใหญ่มีลักษณะการกระจายใกล้เคียงการแจกแจงแบบปกติ (Normal Distribution) โดยตัวแปรอายุ (age) มีการกระจายอยู่ในช่วงประมาณ 20–85 ปี และมีความหนาแน่นของข้อมูลอยู่ในช่วงอายุประมาณ 40–60 ปี ส่วนดัชนีมวลกาย (BMI) มีการกระจายอยู่ในช่วงประมาณ 17–42 kg/m² โดยข้อมูลส่วนใหญ่กระจุกตัวอยู่บริเวณ 22–30 kg/m² สำหรับตัวแปรทางชีวเคมี ได้แก่ คอเลสเตอรอลรวม (total_cholesterol_mg_dl) คอเลสเตอรอลชนิด LDL (ldl_mg_dl) คอเลสเตอรอลชนิด HDL (hdl_mg_dl) และครีเอตินิน (creatinine_mg_dl) พบว่ามีลักษณะการกระจายแบบเบ้ขวา (Right-skewed Distribution) ในระดับเล็กน้อยถึงปานกลาง โดยเฉพาะตัวแปรครีเอตินินซึ่งมีค่าบางส่วนกระจายอยู่ทางด้านขวาของการแจกแจง แสดงให้เห็นว่ามีผู้ป่วยบางรายที่มีค่าครีเอตินินสูงกว่ากลุ่ม

ข้อมูลส่วนใหญ่ อย่างไรก็ตาม ไม่พบความผิดปกติของการกระจายที่รุนแรงหรือค่าที่เบี่ยงเบนอย่างผิดปกติซึ่งอาจส่งผลกระทบต่อวิเคราะห์ในขั้นตอนถัดไป ผลการวิเคราะห์ดังกล่าวแสดงให้เห็นว่าตัวแปรเชิงตัวเลขมีการกระจายของข้อมูลอยู่ในช่วงที่เหมาะสมและสามารถนำไปใช้ในการพัฒนาแบบจำลองการเรียนรู้ของเครื่องในขั้นตอนต่อไปได้

3.2.4 การวิเคราะห์หลายตัวแปร

เพื่อประเมินความสัมพันธ์ระหว่างตัวแปรเชิงตัวเลข ผู้วิจัยได้ทำการวิเคราะห์ค่าสัมประสิทธิ์สหสัมพันธ์ของเพียร์สัน (Pearson Correlation Coefficient) และแสดงผลในรูปแบบ Correlation Matrix (Heatmap) ดังแสดงในรูปที่ 3.3



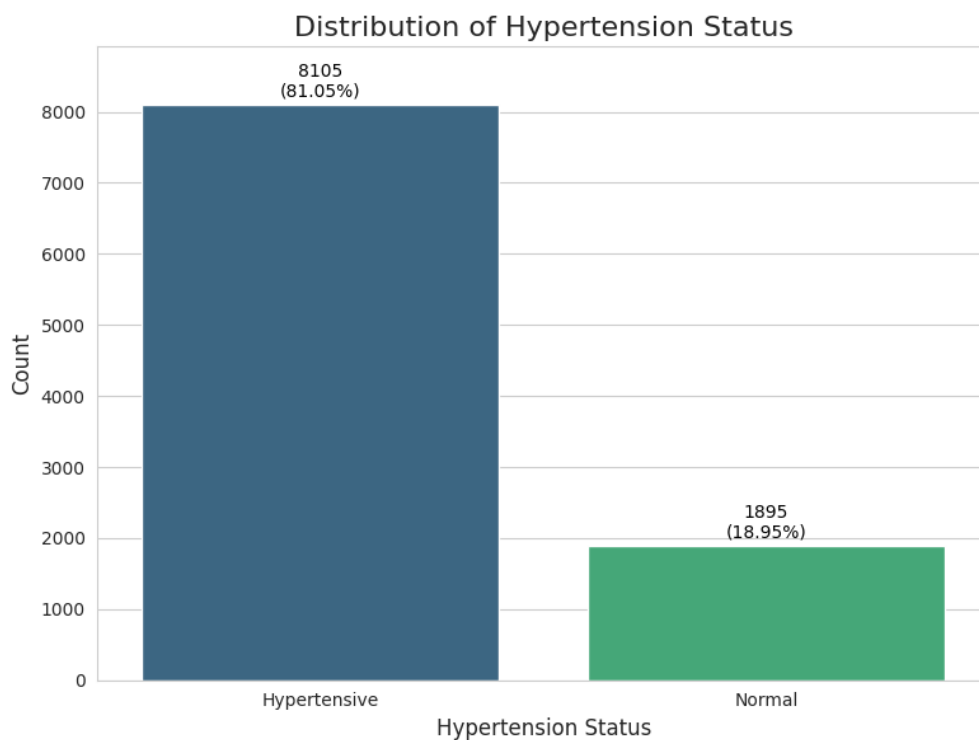
ภาพประกอบที่ 3.3 Correlation Matrix แสดงความสัมพันธ์ระหว่างตัวแปรเชิงตัวเลขด้วยค่าสหสัมพันธ์

จากภาพประกอบที่ 3.3 ผลการวิเคราะห์พบว่า ตัวแปรเชิงตัวเลขส่วนใหญ่มีความสัมพันธ์กันอยู่ในระดับต่ำ โดยมีค่าสัมประสิทธิ์สหสัมพันธ์อยู่ระหว่าง -0.13 ถึง 0.22 ยกเว้นความสัมพันธ์ระหว่าง ค่าความดันโลหิตตัวบน (systolic_bp_mmhg) และ ค่าความดันโลหิตตัวล่าง (diastolic_bp_mmhg) ซึ่งมีค่าสัมประสิทธิ์สหสัมพันธ์เท่ากับ 0.41 จัดเป็นความสัมพันธ์เชิงบวกระดับปานกลาง สะท้อนให้เห็นว่าตัวแปรทั้งสองมีแนวโน้มเปลี่ยนแปลงไปในทิศทางเดียวกัน ซึ่งสอดคล้องกับลักษณะทางสรีรวิทยาของระบบไหลเวียนโลหิต นอกจากนี้ พบว่าตัวแปร creatinine_mg_dl มีความสัมพันธ์เชิงบวกระดับต่ำกับ systolic_bp_mmhg

($r = 0.22$) และ diastolic_bp_mmhg ($r = 0.20$) ขณะที่ตัวแปร HDL cholesterol (hdl_mg_dl) มีความสัมพันธ์เชิงลบในระดับต่ำกับค่าความดันโลหิตทั้งสอง ($r = -0.13$ และ -0.11 ตามลำดับ) ส่วนตัวแปรอื่นๆ เช่น อายุ (age), ดัชนีมวลกาย (BMI), คอเลสเตอรอลรวม (total_cholesterol_mg_dl) และ LDL cholesterol (ldl_mg_dl) มีความสัมพันธ์กับตัวแปรอื่นในระดับต่ำทั้งหมด

3.2.5 ความไม่สมดุลทางข้อมูล

ตรวจสอบสัดส่วนของคลาสเป้าหมาย ใช้เพื่อกำหนดกลยุทธ์การจัดการในขั้นตอนถัดไป



ภาพประกอบที่ 3.4 ภาพเปรียบเทียบจำนวนในคลาสผลลัพธ์

จากภาพประกอบที่ 3.4 แสดงการกระจายของคลาสเป้าหมาย ของชุดข้อมูลที่ใช้ในการศึกษา ซึ่งประกอบด้วยข้อมูลทั้งหมดจำนวน 10,000 ตัวอย่าง โดยแบ่งออกเป็น 2 คลาส ได้แก่ Hypertensive ซึ่งหมายถึงผู้ที่มีภาวะความดันโลหิตสูง จำนวน 8,105 ตัวอย่าง (81.05%) และ Normal ซึ่งหมายถึงผู้ที่มีความดันโลหิตอยู่ในระดับปกติ จำนวน 1,895 ตัวอย่าง (18.95%) จากการวิเคราะห์พบว่าชุดข้อมูลมีลักษณะข้อมูลไม่สมดุล เนื่องจากจำนวนข้อมูลในคลาส Hypertensive มีสัดส่วนมากกว่าคลาส Normal อย่างชัดเจน

ตารางที่ 3.7 ผลการเปรียบเทียบค่า Accuracy ของตัวแบบภายหลังการใช้เทคนิคจัดการข้อมูลไม่สมดุล

MODEL	Original	ADASYN	SMOTE	SMOTENC	SMOTE-Tomek	SMOTE-ENN	UnderSampling
XGBoost	0.8845	0.8515	0.8525	0.8500	0.8550	0.8445	0.8250
Random Forest	0.8760	0.8440	0.8580	0.8620	0.8545	0.8325	0.8075
LightGBM	0.8785	0.8505	0.8600	0.8490	0.8510	0.8325	0.8215

Cat boost	0.8850	0.8510	0.8605	0.8440	0.8530	0.8275	0.8325
MLP	0.8830	0.8485	0.8535	0.8520	0.8525	0.8270	0.8100
TabNet	0.8740	0.8310	0.8325	0.7950	0.8340	0.7960	0.7570
FT-Transformer	0.8825	0.8505	0.8470	0.8360	0.8530	0.8315	0.8255
SAINT	0.8825	0.8065	0.8440	0.8580	0.8505	0.8185	0.7845
Tab transformer	0.8555	0.8045	0.7745	0.7890	0.8005	0.7540	0.7870

ในการศึกษานี้ ได้ทำการเปรียบเทียบวิธีการจัดการปัญหาข้อมูลไม่สมดุลหลายรูปแบบ ประกอบด้วย การใช้ข้อมูลต้นฉบับโดยไม่ปรับสมดุลข้อมูลและเทคนิคการสุ่มเพิ่มตัวอย่างข้อมูลเพื่อศึกษาผลกระทบของแต่ละวิธีต่อประสิทธิภาพของตัวแบบพยากรณ์ โดยผลการเปรียบเทียบจากการประเมินบนชุดข้อมูลทดสอบแสดงดังตารางที่ 3.7 ผลการทดลองพบว่า การใช้ข้อมูลต้นฉบับโดยไม่ปรับสมดุลข้อมูลให้ผลการพยากรณ์ที่มีประสิทธิภาพสูงสุด เมื่อพิจารณาจากค่า Accuracy ดังนั้น การศึกษานี้จึงเลือกใช้ข้อมูลต้นฉบับในการพัฒนาและฝึกสอนตัวแบบสำหรับการทดลองทั้งหมด เนื่องจากการประยุกต์ใช้เทคนิคการสุ่มเพิ่มตัวอย่างข้อมูลไม่สามารถปรับปรุงประสิทธิภาพของแบบจำลองให้ดียิ่งขึ้นเมื่อเทียบกับการใช้ข้อมูลต้นฉบับ

3.3 การเตรียมข้อมูล

การเตรียมข้อมูลมีความสำคัญอย่างยิ่งต่อคุณภาพและประสิทธิภาพของแบบจำลอง งานวิจัยนี้ดำเนินการเตรียมข้อมูลอย่างเป็นระบบตามขั้นตอนย่อยดังต่อไปนี้

3.3.1 การลบตัวแปรที่ไม่ใช่ปัจจัยเสี่ยง

ในขั้นตอนการเตรียมข้อมูลสำหรับการพัฒนาแบบจำลองทำนายภาวะความดันโลหิตสูง ได้ดำเนินการตรวจสอบตัวแปรทั้งหมดในชุดข้อมูลเพื่อคัดเลือกตัวแปรที่เหมาะสม ลดความซับซ้อนของแบบจำลอง และป้องกันปัญหาการรั่วไหลของข้อมูล (Data Leakage) ซึ่งอาจเกิดจากการใช้ตัวแปรที่มีข้อมูลของผลลัพธ์หรือข้อมูลที่เกิดขึ้นภายหลังการวินิจฉัย โดยมีรายละเอียดดังนี้ (1) ตัดตัวแปร patient_id ซึ่งเป็นตัวระบุตัวตนของผู้เข้าร่วมการศึกษาออก เนื่องจากไม่มีความสัมพันธ์กับการทำนาย (2) เติมค่าที่ขาดหาย (Missing Values) ในตัวแปร medication_adherence ด้วยหมวดหมู่ "No_Medication" แทนการลบแถวข้อมูล เพื่อรักษาจำนวนตัวอย่างและสะท้อนความหมายของข้อมูลได้อย่างเหมาะสม (3) ตัดตัวแปรที่มีความเสี่ยงต่อการรั่วไหลของข้อมูลจำนวน 6 ตัวแปร ได้แก่ diagnosed, on_medication, bp_controlled, medication_adherence, systolic_bp_mmhg และ diastolic_bp_mmhg เนื่องจากตัวแปรเหล่านี้ถูก

บันทึกภายหลังการวินิจฉัย หรือมีความสัมพันธ์โดยตรงกับตัวแปรผลลัพธ์ ซึ่งอาจทำให้แบบจำลองเรียนรู้ข้อมูลที่ไม่สามารถใช้ได้ในสถานการณ์จริง ส่งผลให้ค่าประสิทธิภาพของแบบจำลองสูงเกินความเป็นจริง

ทั้งนี้ ตารางที่ 3.8 แสดงรายละเอียดของตัวแปรที่ได้รับการคัดเลือกสำหรับการใช้ในการพัฒนาแบบจำลอง ซึ่งครอบคลุมข้อมูลด้านประชากรศาสตร์ พฤติกรรมสุขภาพ และตัวแปรทางคลินิกที่เกี่ยวข้องกับความเสี่ยงของภาวะความดันโลหิตสูง

ตารางที่ 3.8 ตารางข้อมูลหลังจากที่ตัดข้อมูลที่มีข้อมูลรั่วไหลและไม่เกี่ยวข้องออก

Feature	Description	Type
age	อายุของผู้ป่วย (ปี)	Numerical
sex	เพศของผู้ป่วย (Male, Female)	Categorical
bmi	ดัชนีมวลกายของผู้ป่วย	Numerical
family_history_hypertension	ประวัติครอบครัวเป็นโรคความดันโลหิตสูง	Binary
diabetes	สถานะโรคเบาหวาน	Binary
smoking	พฤติกรรมการสูบบุหรี่	Binary
alcohol_heavy	การดื่มแอลกอฮอล์ในปริมาณมาก	Binary
physically_active	การออกกำลังกายสม่ำเสมอ	Binary
high_salt_diet	การบริโภคอาหารเค็ม	Binary
total_cholesterol_mg_dl	คอเลสเตอรอลรวมในเลือด	Numerical
ldl_mg_dl	ระดับ LDL Cholesterol	Numerical
hdl_mg_dl	ระดับ HDL Cholesterol	Numerical
creatinine_mg_dl	ระดับ Creatinine ในเลือด	Numerical
residence	พื้นที่อยู่อาศัย (Urban, Rural)	Categorical
stroke_history	ประวัติโรคหลอดเลือดสมอง	Binary
myocardial_infarction	ประวัติกล้ามเนื้อหัวใจตาย	Binary
heart_failure	ประวัติภาวะหัวใจล้มเหลว	Binary

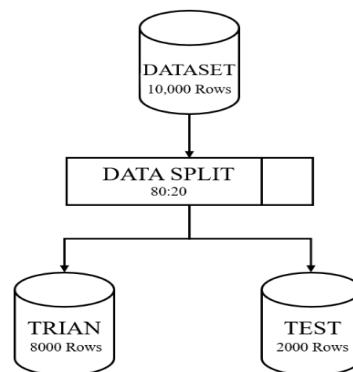
จากตารางที่ 3.8 แสดงตัวแปรที่ใช้ในการวิเคราะห์เพื่อพัฒนาแบบจำลองสำหรับการทำนายภาวะความดันโลหิตสูง ซึ่งประกอบด้วยข้อมูลด้านประชากรศาสตร์ พฤติกรรมสุขภาพ และค่าทางชีวเคมีที่เกี่ยวข้องกับความเสี่ยงของโรคความดันโลหิตสูงเพื่อเข้าสู่ขั้นตอนการเตรียมข้อมูลในขั้นตอนถัดไป

3.3.2 การแบ่งชุดข้อมูล

ชุดข้อมูลถูกแบ่งออกเป็น 2 ส่วน 80:20 เพื่อใช้ในวัตถุประสงค์ที่แตกต่างกัน ได้แก่ ชุดข้อมูลฝึกสอน (Training Set) และชุดข้อมูลทดสอบ (Test Set) โดยมีสัดส่วนดังแสดงในตารางที่ 3.9

ตารางที่ 3.9 ตารางข้อมูลที่แบ่งในแต่ละส่วน

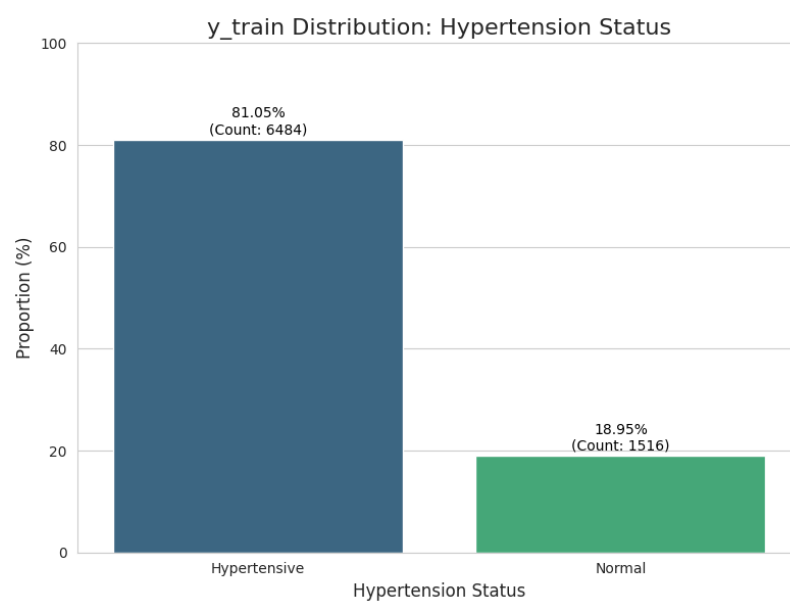
Dataset	Number of Samples	Percentage
Training Set	8,000	80%
Test Set	2,000	20%
Total	10,000	100%



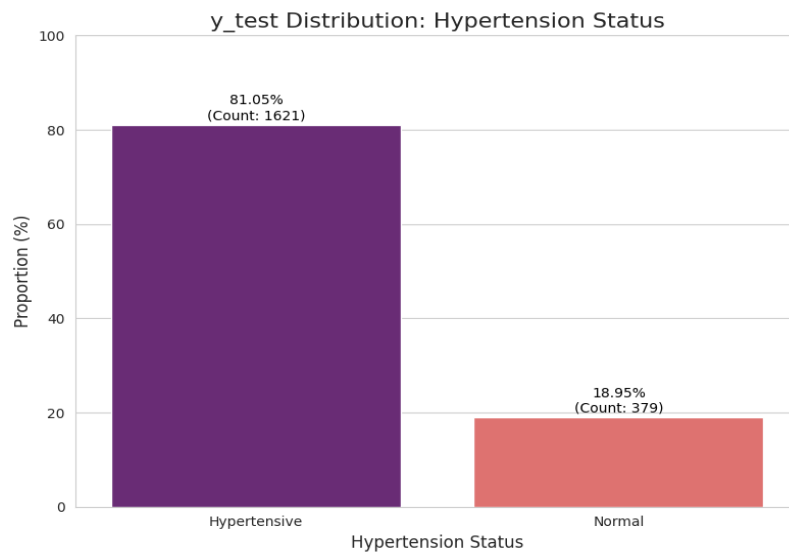
ภาพประกอบที่ 3.5 ภาพแสดงการแบ่งชุดข้อมูลต่อจำนวนข้อมูลทั้งหมด

จากภาพประกอบที่ 3.3 แสดงการแบ่งชุดข้อมูลออกเป็น 2 ส่วน ได้แก่ Training Set จำนวน 8,000 ตัวอย่าง และ Test Set จำนวน 2,000 ตัวอย่าง ซึ่งใช้สำหรับการฝึกโมเดล การปรับค่าพารามิเตอร์ และการประเมินผลลัพธ์ของโมเดลตามลำดับ

หลังจากแบ่งชุดข้อมูลออกเป็นชุดข้อมูลฝึกสอนและชุดข้อมูลทดสอบด้วยสัดส่วน 80:20 ได้ตรวจสอบการกระจายของตัวแปรเป้าหมาย (Hypertension Status) ในแต่ละชุดข้อมูล เพื่อยืนยันว่าการแบ่งข้อมูลยังคงรักษาสัดส่วนของแต่ละคลาสไว้ (Stratified Sampling) ดังแสดงในรูปที่ 3.6 และรูปที่ 3.7



ภาพประกอบที่ 3.6 การกระจายของตัวแปรเป้าหมายในชุดข้อมูลฝึกสอน



ภาพประกอบที่ 3.7 การกระจายของตัวแปรเป้าหมายในชุดข้อมูลทดสอบ

จากรูปที่ 3.6 และรูปที่ 3.7 พบว่า การแบ่งข้อมูลด้วยวิธี Stratified Train-Test Split สามารถรักษาสัดส่วนของตัวแปรเป้าหมายไว้ได้อย่างเหมาะสม โดยทั้งชุดข้อมูลฝึกสอนและชุดข้อมูลทดสอบมีการกระจายของคลาสเหมือนกัน คือ กลุ่มผู้ป่วยที่มีภาวะความดันโลหิตสูง (Hypertensive) คิดเป็นร้อยละ 81.05 และกลุ่มปกติ (Normal) คิดเป็นร้อยละ 18.95 ของข้อมูลทั้งหมด สำหรับชุดข้อมูลฝึกสอน (Training Set) มีผู้ป่วยภาวะความดันโลหิตสูงจำนวน 6,484 ราย และกลุ่มปกติจำนวน 1,516 ราย ขณะที่ชุดข้อมูลทดสอบ (Test Set) มีผู้ป่วยภาวะความดันโลหิตสูงจำนวน 1,621 ราย และกลุ่มปกติจำนวน 379 ราย ซึ่งมีสัดส่วนสอดคล้องกับชุดข้อมูลเดิม ผลการตรวจสอบดังกล่าวแสดงให้เห็นว่าการแบ่งข้อมูลสามารถรักษาการกระจายของตัวแปรเป้าหมายได้อย่างถูกต้อง ลดความเอนเอียงที่อาจเกิดจากการแบ่งข้อมูล และช่วยให้การฝึกสอนและการประเมินประสิทธิภาพของแบบจำลองมีความน่าเชื่อถือมากยิ่งขึ้น ทั้งนี้ ยังพบว่าชุดข้อมูลมีลักษณะ ไม่สมดุลของคลาส (Class Imbalance) โดยกลุ่มผู้ป่วยภาวะความดันโลหิตสูงมีจำนวนมากกว่ากลุ่มปกติอย่างชัดเจน (81.05% ต่อ 18.95%) ซึ่งเป็นประเด็นที่ต้องพิจารณาในการพัฒนาและประเมินแบบจำลองในขั้นตอนต่อไป

3.3.3 การเข้ารหัสตัวแปร

เนื่องจากแบบจำลองการเรียนรู้ของเครื่องส่วนใหญ่ไม่สามารถประมวลผลตัวแปรเชิงหมวดหมู่ในรูปแบบข้อความได้โดยตรง จึงจำเป็นต้องแปลงตัวแปรเหล่านี้ให้อยู่ในรูปแบบตัวเลขก่อนนำไปใช้งาน โดยการเข้ารหัสตัวแปรผลลัพธ์ hypertension_status ให้อยู่ในรูปแบบทวิภาค โดยกำหนด Normal = 0 และ Hypertensive = 1 และ แปลงตัวแปรเชิงกลุ่ม sex และ residence ด้วยวิธี One-Hot Encoding โดยกำหนด drop_first=True เพื่อหลีกเลี่ยงปัญหาภาวะร่วมเส้นตรงเชิงพหุทำให้ได้ตัวแปรใหม่ ได้แก่ sex_Male และ residence_Urban

3.3.4 การปรับมาตรฐานของตัวแปร

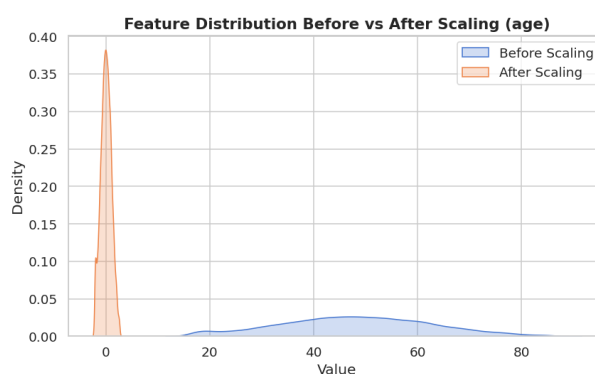
ในขั้นตอนการมาตรฐานข้อมูล ตัวแปรเชิงตัวเลขทุกตัวถูกปรับมาตรฐาน เพื่อให้มีค่าเฉลี่ยเท่ากับ 0 และส่วนเบี่ยงเบนมาตรฐานเท่ากับ 1 ซึ่งช่วยลดผลกระทบจากความแตกต่างของช่วงค่าระหว่างตัวแปร และช่วยให้โมเดลการเรียนรู้ของเครื่องสามารถเรียนรู้รูปแบบของข้อมูลได้อย่างมีประสิทธิภาพมากขึ้น ในการศึกษาครั้งนี้ใช้เทคนิค Standardization หรือ Z-score Normalization โดยใช้เครื่องมือ StandardScaler จากไลบรารี scikit-learn

ภายหลังการเข้ารหัสตัวแปรเชิงกลุ่มทั้งหมด 11 ตัว ได้แก่ family_history_hypertension, diabetes, smoking, alcohol_heavy, physically_active, high_salt_diet, stroke_history, myocardial_infarction, heart_failure, sex_Male และ residence_Urban ถูกนำไปใช้กับแบบจำลองกลุ่มต้นไม้แบบรวม (Ensemble Tree-based Models) โดยไม่ทำการปรับสเกลข้อมูล เนื่องจากแบบจำลองประเภทดังกล่าวไม่ไวต่อขนาดของข้อมูล ในขณะที่แบบจำลองการเรียนรู้เชิงลึก (Deep Learning Models) ซึ่งต้องการข้อมูลอินพุตที่มีมาตราส่วนใกล้เคียงกัน ได้ทำการปรับมาตรฐานเฉพาะตัวแปรเชิงต่อเนื่องด้วยวิธี StandardScaler ก่อนนำไปฝึกสอนแบบจำลอง

$$z = \frac{x - \mu}{\sigma}$$

สมการที่ 3.1 Standardization

โดยที่	x	คือค่าข้อมูลเดิม
	μ	คือค่าเฉลี่ยข้อมูล
	σ	คือส่วนเบี่ยงเบนมาตรฐาน
	z	คือค่าที่ถูกปรับมาตรฐานแล้ว



ภาพประกอบที่ 3.8 การเปรียบเทียบการกระจายของข้อมูลก่อนและหลังการปรับมาตรฐาน

ภาพที่ 3.8 แสดงการเปรียบเทียบการกระจายตัวของตัวแปร age ก่อนและหลังการปรับมาตรฐาน ข้อมูลด้วยเทคนิค Standardization โดยพบว่าภายหลังการปรับมาตรฐานแล้ว ข้อมูลถูกแปลงให้มีค่าเฉลี่ยใกล้เคียงศูนย์และส่วนเบี่ยงเบนมาตรฐานเท่ากับหนึ่ง ซึ่งเป็นลักษณะการกระจายตัวที่เหมาะสมสำหรับการนำไปใช้ในการฝึกแบบจำลอง

ในการศึกษานี้ดำเนินการ Standardization กับตัวแปรเชิงตัวเลข ได้แก่ age, bmi, total_cholesterol_mg_dl, ldl_mg_dl, hdl_mg_dl และ creatinine_mg_dl

ค่าพารามิเตอร์ของ Scaler ได้แก่ ค่าเฉลี่ย (μ) และส่วนเบี่ยงเบนมาตรฐาน (σ) จะถูกคำนวณจากชุดข้อมูลฝึกสอนเท่านั้น และนำค่าพารามิเตอร์ที่ได้ไปใช้แปลงข้อมูลในชุดข้อมูลตรวจสอบความถูกต้อง และในชุดข้อมูลทดสอบ ในภายหลัง

3.4 การพัฒนาตัวแบบ

ในการศึกษานี้ ผู้วิจัยได้พัฒนาตัวแบบการเรียนรู้ของเครื่องและการเรียนรู้เชิงลึกเพื่อใช้ในการจำแนกและทำนายผลลัพธ์ทางคลินิกจากข้อมูลเชิงตาราง โดยเลือกใช้แบบจำลองที่มีโครงสร้างการเรียนรู้ที่แตกต่างกันจำนวน 9 ตัวแบบ แบ่งเป็นกลุ่มโมเดลต้นไม้แบบรวม (Tree-based Ensemble) ได้แก่ XGBoost, Random Forest, LightGBM และ CatBoost กลุ่มการเรียนรู้เชิงลึก (Deep Learning) ได้แก่ Multi-Layer Perceptron (MLP), TabNet และกลุ่มทรานส์ฟอร์มเมอร์ ได้แก่ FT-Transformer, SAINT และ TabTransformer การเลือกใช้ตัวแบบที่หลากหลายดังกล่าวมีวัตถุประสงค์เพื่อเปรียบเทียบประสิทธิภาพของแนวทางการเรียนรู้ที่แตกต่างกันในการจัดการกับข้อมูลทางการแพทย์ ซึ่งมักมีลักษณะเป็นข้อมูลเชิงตาราง และมีความสัมพันธ์ระหว่างตัวแปรที่ซับซ้อน

ในกระบวนการพัฒนาตัวแบบ ข้อมูลที่ผ่านการเตรียมข้อมูลและการปรับสมดุลของข้อมูลแล้วจะถูกแบ่งออกเป็นชุดข้อมูลสำหรับการฝึกสอนและชุดข้อมูลสำหรับการตรวจสอบความถูกต้องของแบบจำลองเพื่อใช้ในการประเมินสมรรถนะของตัวแบบในระหว่างการฝึกสอน สำหรับตัวแบบที่พัฒนาด้วยเฟรมเวิร์ก PyTorch จะใช้การฝึกแบบ mini-batch gradient descent โดยใช้ตัวปรับค่าพารามิเตอร์แบบ Adam optimizer และใช้ Binary Cross-Entropy เป็นฟังก์ชันความสูญเสียสำหรับปัญหาการจำแนกแบบสองกลุ่ม นอกจากนี้ยังมีการใช้เทคนิค Early Stopping และ Learning Rate Scheduler เพื่อลดปัญหาการเกิด overfitting และเพิ่มประสิทธิภาพในการเรียนรู้ของแบบจำลอง ในการประเมินสมรรถนะของตัวแบบ ผู้วิจัยใช้ค่า Area Under the Receiver Operating Characteristic Curve (ROC-AUC) เป็นตัวชี้วัดหลัก เนื่องจากเป็นตัวชี้วัดที่เหมาะสมสำหรับปัญหาการจำแนกแบบสองกลุ่ม โดยเฉพาะในกรณีที่ข้อมูลมีความไม่สมดุลของคลาส ทั้งนี้รายละเอียดของหลักการทำงานและขั้นตอนการพัฒนาของแต่ละแบบจำลองมีดังต่อไปนี้

3.4.1 การพัฒนากลุ่มตัวแบบทรีแบบรวม

กลุ่มนี้ประกอบด้วยตัวแบบสี่ประเภท ได้แก่ XGBoost, Random Forest, LightGBM และ CatBoost ขอบเขตการค้นหาพารามิเตอร์แสดงในตารางที่ 3.10 มาจากฟังก์ชัน objective ของ Optuna โดยพารามิเตอร์แต่ละตัวถูกส่งผ่าน trial.suggest_int, trial.suggest_float หรือ trial.suggest_categorical ตามประเภทของค่าที่เหมาะสม และมีการค้นหาแบบ log-uniform สำหรับพารามิเตอร์ที่มีช่วงค่ากว้างหลายเท่าตัว เช่น learning_rate และค่าปรับความซับซ้อน (reg_alpha, reg_lambda)

ตารางที่ 3.10 ขอบเขตการค้นหาไฮเปอร์พารามิเตอร์ของกลุ่มตัวแบบทรีแบบรวม

ตัวแบบ	พารามิเตอร์	ขอบเขตการค้นหาจริง
XGBoost	n_estimators, max_depth, learning_rate, subsample, colsample_bytree, min_child_weight, gamma, reg_alpha, reg_lambda, max_delta_step	n_estimators int(100,600); max_depth int(3,12); learning_rate float(0.01,0.3,log); subsample float(0.6,1.0); colsample_bytree float(0.6,1.0); min_child_weight float(1.0,10.0); gamma float(0.0,5.0); reg_alpha/reg_lambda float(1e-8,10.0,log); max_delta_step int(0,10)
Random Forest	n_estimators, max_depth, min_samples_split, min_samples_leaf, max_features, bootstrap, criterion, max_leaf_nodes, min_impurity_decrease, ccp_alpha, class_weight	n_estimators int(100,800); max_depth int(3,24); min_samples_split int(2,20); min_samples_leaf int(1,10); max_features {sqrt, log2, None}; bootstrap {True, False}; criterion {gini, entropy}; max_leaf_nodes {None,64,128,256}; min_impurity_decrease float(0.0,0.02); ccp_alpha float(1e-6,1e-2,log); class_weight {None, balanced}
LightGBM	n_estimators, num_leaves, max_depth, learning_rate, min_child_samples, min_child_weight, subsample, subsample_freq, colsample_bytree, reg_alpha, reg_lambda, min_split_gain, max_bin	n_estimators int(100,600); num_leaves int(16,160); max_depth int(3,12); learning_rate float(0.01,0.2,log); min_child_samples int(10,120); min_child_weight float(1e-3,10.0,log); subsample float(0.6,1.0); subsample_freq int(1,7); colsample_bytree float(0.6,1.0); reg_alpha/reg_lambda float(1e-8,10.0,log); min_split_gain float(0.0,1.0); max_bin int(127,511)
CatBoost	iterations, depth, learning_rate, l2_leaf_reg, border_count, random_strength, bagging_temperature, rsm, leaf_estimation_iterations	iterations int(200,1000); depth int(4,10); learning_rate float(0.01,0.3,log); l2_leaf_reg float(1.0,20.0,log); border_count int(32,255); random_strength float(1e-8,10.0,log); bagging_temperature float(0.0,5.0); rsm float(0.6,1.0); leaf_estimation_iterations int(1,10)

3.4.2 การพัฒนาตัวแบบกลุ่มการเรียนรู้เชิงลึก

กลุ่มนี้ประกอบด้วยตัวแบบห้าประเภท ได้แก่ Multi-Layer Perceptron, TabNet, FT-Transformer, SAINT และ TabTransformer ทุกตัวแบบใช้ตัวหยุดเรียนก่อนกำหนดเพื่อลดความเสี่ยงของ

การเรียนรู้เกินพอดีและควบคุมเวลาการฝึกให้อยู่ในงบประมาณที่ใกล้เคียงกับกลุ่มตัวแบบที่รีชอบเขตการค้นหามิติจริงสรุปไว้ในตารางที่ 3.11

ตารางที่ 3.11 ขอบเขตการค้นหาไฮเปอร์พารามิเตอร์ของกลุ่มตัวแบบการเรียนรู้เชิงลึก

ตัวแบบ	พารามิเตอร์	ขอบเขตการค้นหามิติจริง
MLP	n_layers, n_units_{l(i)}, activation, alpha, learning_rate_init, batch_size, beta_1, beta_2, epsilon, tol	n_layers int(1,3); n_units ต่อชั้น int(32,256); activation {relu}; alpha float(1e-5,1e-2,log); learning_rate_init float(1e-4,1e-2,log); batch_size {32,64,128}; beta_1 float(0.85,0.99); beta_2 float(0.9,0.9999); epsilon float(1e-9,1e-7,log); tol float(1e-5,1e-3,log)
TabNet	n_d, n_a, n_steps, gamma, lambda_sparse, lr, weight_decay, batch_size, virtual_batch_size, momentum, mask_type, scheduler_gamma, scheduler_step_size	n_d/n_a int(8,32); n_steps int(3,6); gamma float(1.0,2.0); lambda_sparse float(1e-6,1e-3,log); lr float(1e-3,1e-2,log); weight_decay float(1e-6,1e-3,log); batch_size {512,1024}; virtual_batch_size {128,256}; momentum float(0.01,0.4); mask_type {sparsemax, entmax}; scheduler_gamma float(0.5,0.95); scheduler_step_size int(5,15)

3.4.3 การพัฒนาตัวแบบกลุ่มทรานส์ฟอร์มเมอร์

กลุ่มนี้ประกอบด้วยตัวแบบห้าประเภท ได้แก่ FT-Transformer, SAINT และ TabTransformer ทุกตัวแบบใช้ตัวหยุดเรียนก่อนกำหนดเพื่อลดความเสี่ยงของการเรียนรู้เกินพอดีและควบคุมเวลาการฝึกให้อยู่ในงบประมาณที่ใกล้เคียงกับกลุ่มตัวแบบที่รีชอบเขตการค้นหามิติจริงสรุปไว้ในตารางที่ 3.12

ตารางที่ 3.12 ขอบเขตการค้นหาไฮเปอร์พารามิเตอร์ของกลุ่มตัวแบบกลุ่มทรานส์ฟอร์มเมอร์

FT-Transformer	d_block, n_blocks, n_heads, attention_dropout, ffn_dropout, residual_dropout, lr, weight_decay, batch_size, patience, ffn_d_hidden_multiplier	d_block {128,192}; n_blocks int(1,3); n_heads {2,4}; attention/ffn_dropout float(0.0,0.3); residual_dropout float(0.0,0.2); lr float(1e-4,5e-3,log); weight_decay float(1e-6,1e-2,log); batch_size {512,1024}; patience int(5,12); ffn_d_hidden_multiplier {2,4}
SAINT	dim, depth, heads, dropout, mlp_hidden_multiplier, ff_mult, lr, weight_decay, batch_size, patience, gradient_clip_norm	dim {16,32}; depth int(1,2); heads {2,4}; dropout float(0.0,0.3); mlp_hidden_multiplier {2,4}; ff_mult {2,4,6}; lr float(1e-4,2e-3,log); weight_decay float(1e-6,1e-2,log); batch_size {512,1024};

		patience int(5,12); gradient_clip_norm float(0.5,2.0)
TabTransformer	dim, depth, heads, attn_dropout, ff_dropout, mlp_hidden_multiplier, lr, weight_decay, batch_size, patience	dim {48,64}; depth int(1,3); heads {2,4}; attn_dropout/ff_dropout float(0.0,0.3); mlp_hidden_multiplier {2,4}; lr float(1e-4,2e-3,log); weight_decay float(1e-6,1e-2,log); batch_size {512,1024}; patience int(5,12)

3.4.4 กระบวนการปรับค่าพารามิเตอร์

ตัวแบบทุกประเภทได้รับการปรับค่าไฮเปอร์พารามิเตอร์ด้วยกรอบการทำงาน Optuna โดยใช้กลยุทธ์การสุ่มตัวอย่างแบบ Tree-structured Parzen Estimator (TPE) ซึ่งเป็นค่าตั้งต้นของ Optuna ภายใต้จำนวนรอบการทดลอง (n_trials) เท่ากับ 30 รอบสำหรับทุกตัวแบบ เพื่อควบคุมการค้นหาค่าที่เหมาะสมตามหลักการเปรียบเทียบที่เป็นธรรมชาติแต่ละรอบการทดลองประเมินผลด้วยค่าเฉลี่ย AUC-ROC จากการตรวจสอบไขว้ Stratified 5-Fold (SEED = 42) หลังจากเสร็จสิ้นกระบวนการปรับแต่งแล้ว ชุดค่าพารามิเตอร์ที่ดีที่สุดจะถูกนำไปใช้ในการฝึกแบบจำลองเวอร์ชันสุดท้าย

1. การปรับแต่งพารามิเตอร์ของ XGBoost

สำหรับแบบจำลอง XGBoost ได้มีการปรับแต่งพารามิเตอร์ที่สำคัญหลายรายการเพื่อควบคุมความซับซ้อนของแบบจำลองและลดความเสี่ยงของการเกิด Overfitting พารามิเตอร์เหล่านี้มีอิทธิพลต่อโครงสร้างและพฤติกรรมการเรียนรู้ของต้นไม้ตัดสินใจในกระบวนการ Gradient Boosting

ตารางที่ 3.13 การปรับแต่งพารามิเตอร์ของ XGBoost ด้วย Optuna

Parameter	Value	Description
n_estimators	463	จำนวนต้นไม้ใน ensemble
max_depth	5	ความลึกสูงสุดของแต่ละ decision tree
learning_rate	0.04305	อัตราการเรียนรู้ที่ใช้ในกระบวนการ boosting
gamma	4.0164	ค่าลดความสูญเสียขั้นต่ำที่ยอมให้แบ่งโหนดเพิ่มเติม
min_child_weight	4.1557	ผลรวมน้ำหนักขั้นต่ำของตัวอย่างในแต่ละใบ
max_delta_step	6	ค่าจำกัดการเปลี่ยนแปลงน้ำหนักในแต่ละรอบ
subsample	0.7408	สัดส่วนของตัวอย่างข้อมูลที่ใช้ในการฝึกแต่ละต้นไม้
colsample_bytree	0.8346	สัดส่วนของคุณลักษณะที่สุ่มเลือกสำหรับแต่ละต้นไม้
reg_alpha	2.049e-06	ค่าพารามิเตอร์การปรับโทษแบบ L1
reg_lambda	9.421e-07	ค่าพารามิเตอร์การปรับโทษแบบ L2

จากตารางที่ 3.13 จะแสดงผลการปรับแต่งไฮเปอร์พารามิเตอร์ด้วย Optuna พบว่าชุดค่าพารามิเตอร์ที่เหมาะสมที่สุดสำหรับแบบจำลอง XGBoost ประกอบด้วยจำนวนต้นไม้ 463 ต้น ความลึก

สูงสุดของต้นไม้เท่ากับ 5 และอัตราการเรียนรู้เท่ากับ 0.04305 นอกจากนี้ได้กำหนดค่า subsample และ colsample_bytree เท่ากับ 0.7408 และ 0.8346 ตามลำดับ พร้อมทั้งใช้ค่าการปรับโทษแบบ L1 และ L2 ที่มีค่าน้อยมาก (2.049×10^{-6} และ 9.421×10^{-7}) ซึ่งบ่งชี้ว่าความซับซ้อนของแบบจำลองถูกควบคุมหลักผ่านค่า gamma และ min_child_weight มากกว่าค่าปรับโทษ L1/L2 โดยตรง

2. การปรับแต่งพารามิเตอร์ของ Random Forest

Random Forest เป็นแบบจำลองกลุ่ม Bagging Ensemble ที่สร้างต้นไม้ตัดสินใจจำนวนมากจากการสุ่มตัวอย่างข้อมูลและคุณลักษณะ ในการศึกษาครั้งนี้ ผู้วิจัยพัฒนาแบบจำลอง Random Forest ปรับแต่งค่าพารามิเตอร์ด้วยเฟรมเวิร์ก Optuna ผลการปรับแต่งไฮเปอร์พารามิเตอร์ที่ดีที่สุดแสดงดังตารางที่ 3.14

ตารางที่ 3.14 การปรับแต่งพารามิเตอร์ของ Random Forest ด้วย Optuna

Parameter	Value	Description
n_estimators	230	จำนวนต้นไม้ตัดสินใจในป่า
max_depth	13	ความลึกสูงสุดของแต่ละต้นไม้
max_features	sqrt	จำนวนคุณลักษณะที่สุ่มพิจารณาในแต่ละจุดแบ่ง
max_leaf_nodes	128	จำนวนใบสูงสุดของแต่ละต้นไม้
min_samples_split	12	จำนวนตัวอย่างขั้นต่ำที่ต้องมีก่อนแบ่งโหนด
min_samples_leaf	7	จำนวนตัวอย่างขั้นต่ำในแต่ละใบ
min_impurity_decrease	0.00028	ค่าลดความไม่บริสุทธิ์ขั้นต่ำที่ยอมให้แบ่งโหนด
ccp_alpha	4.54e-06	พารามิเตอร์การตัดแต่งกิ่งแบบ Cost-Complexity
criterion	gini	ฟังก์ชันวัดคุณภาพของการแบ่งโหนด
class_weight	balanced	การถ่วงน้ำหนักคลาสเพื่อชดเชยข้อมูลไม่สมดุล

จากตารางที่ 3.14 แสดงผลการปรับแต่งไฮเปอร์พารามิเตอร์ของแบบจำลอง Random Forest ด้วย Optuna ซึ่งได้ค่าพารามิเตอร์ที่เหมาะสม ได้แก่ การกำหนดจำนวนต้นไม้ ความลึกของต้นไม้ จำนวนโหนดปลายทาง และเงื่อนไขการแบ่งโหนด เพื่อควบคุมความซับซ้อนของแบบจำลองและลดโอกาสการเกิด Overfitting นอกจากนี้ ยังใช้เกณฑ์ Gini Impurity ในการแบ่งโหนด และกำหนด class_weight = balanced เพื่อรองรับข้อมูลที่มีความไม่สมดุล ส่งผลให้แบบจำลองมีความสามารถในการเรียนรู้ข้อมูลได้อย่างมีประสิทธิภาพและมีศักยภาพในการทำนายข้อมูลใหม่ได้ดียิ่งขึ้น

3. การปรับแต่งพารามิเตอร์ของ LightGBM

LightGBM เป็นแบบจำลองกลุ่ม Gradient Boosting ที่ใช้กลยุทธ์การเติบโตของต้นไม้แบบ Leaf-wise ในการศึกษาครั้งนี้ ผู้วิจัยพัฒนาแบบจำลอง LightGBM โดยใช้ไลบรารี LightGBM ในภาษา Python และปรับแต่งค่าพารามิเตอร์ด้วยเฟรมเวิร์ก Optuna ผลการปรับแต่งไฮเปอร์พารามิเตอร์ที่ดีที่สุดแสดงดังตารางที่ 3.15

ตารางที่ 3.15 การปรับแต่งพารามิเตอร์ของ LightGBM ด้วย Optuna

Parameter	Value	Description
-----------	-------	-------------

n_estimators	500	จำนวนรอบการ boosting (จำนวนต้นไม้)
num_leaves	16	จำนวนใบสูงสุดในแต่ละต้นไม้
max_depth	7	ความลึกสูงสุดของแต่ละต้นไม้
learning_rate	0.01720	อัตราการเรียนรู้ในกระบวนการ boosting
min_child_samples	43	จำนวนตัวอย่างขั้นต่ำในแต่ละใบ
min_child_weight	0.005388	ผลรวมน้ำหนักขั้นต่ำของตัวอย่างในแต่ละใบ
min_split_gain	0.6827	ค่าลดความสูญเสียขั้นต่ำที่ยอมให้แบ่งโหนด
subsample	0.9071	สัดส่วนข้อมูลที่สุ่มใช้ในแต่ละรอบ
subsample_freq	6	ความถี่ของการสุ่มตัวอย่างซ้ำ
colsample_bytree	0.7417	สัดส่วนคุณลักษณะที่สุ่มใช้ในต้นไม้
reg_alpha	0.05026	ค่าปรับโทษแบบ L1
reg_lambda	1.39e-08	ค่าปรับโทษแบบ L2
max_bin	494	จำนวนช่วงสูงสุดสำหรับการแบ่งฮิสโตแกรมของตัวแปรเชิงตัวเลข

จากตารางที่ 3.15 ผลการปรับแต่งไฮเปอร์พารามิเตอร์พบว่าค่าที่เหมาะสมที่สุดใช้จำนวนใบสูงสุด 16 ใบ และอัตราการเรียนรู้ค่อนข้างต่ำ ที่ 0.0172 ร่วมกับจำนวนรอบการ boosting ที่มากถึง 500 รอบ ซึ่งสอดคล้องกับแนวทางการฝึกแบบค่อยเป็นค่อยไปเพื่อลดความเสี่ยงของการเกิด Overfitting

4. การปรับแต่งพารามิเตอร์ของ CatBoost

CatBoost เป็นแบบจำลองกลุ่ม Gradient Boosting ที่ออกแบบมาเพื่จัดการตัวแปรเชิงกลุ่มโดยตรง ในการศึกษาครั้งนี้ ผู้วิจัยพัฒนาแบบจำลอง CatBoost โดยใช้ไลบรารี CatBoost ในภาษา Python และปรับแต่งค่าพารามิเตอร์ด้วยเฟรมเวิร์ก Optuna ผลการปรับแต่งไฮเปอร์พารามิเตอร์ที่ดีที่สุดแสดงดังตารางที่ 3.16

ตารางที่ 3.16 การปรับแต่งพารามิเตอร์ของ CatBoost ด้วย Optuna

Parameter	Value	Description
iterations	902	จำนวนรอบการ boosting (จำนวนต้นไม้)
depth	5	ความลึกของต้นไม้แบบสมมาตร
learning_rate	0.018669	อัตราการเรียนรู้ในกระบวนการ boosting
l2_leaf_reg	6.036	ค่าปรับโทษแบบ L2 สำหรับค่าน้ำหนักที่ใบ
border_count	136	จำนวนขอบเขตสำหรับแบ่งค่าตัวแปรเชิงตัวเลข
bagging_temperature	0.4489	ควบคุมความสุ่มของการถ่วงน้ำหนักตัวอย่างแบบ Bayesian Bootstrap
random_strength	1.0531	ระดับสุ่มที่ใช้ในการให้คะแนนการแบ่งโหนด
rsm	0.8200	สัดส่วนคุณลักษณะที่สุ่มใช้ในแต่ละรอบ (Random Subspace Method)
leaf_estimation_iterations	2	จำนวนรอบการประมาณค่าที่ใบในแต่ละขั้นตอน

จากตารางที่ 3.16 ผลการปรับแต่งไฮเปอร์พารามิเตอร์พบว่าค่าที่เหมาะสมที่สุดใช้ความลึกต้นไม้เท่ากับ 5 ซึ่งค่อนข้างดีเกินกว่าโมเดล Boosting อื่น ร่วมกับจำนวนรอบการ boosting 902 รอบ

5. การปรับแต่งพารามิเตอร์ของ MLP

สำหรับแบบจำลอง MLP (Multilayer perceptron) ได้ทำการปรับแต่งทั้งโครงสร้างของสถาปัตยกรรมและพารามิเตอร์ที่เกี่ยวข้องกับการฝึก เพื่อเพิ่มความสามารถของแบบจำลองในการเรียนรู้รูปแบบเชิงไม่เชิงเส้นในข้อมูลเชิงตารางดังแสดงในตารางที่ 3.17

ตารางที่ 3.17 การปรับแต่งพารามิเตอร์ของ MLP ด้วย Optuna

Parameter	Value	Description
n_layers	2	จำนวนชั้นซ่อนของโครงข่ายประสาทเทียม
n_units_l0	122	จำนวนเซลล์ประสาทในชั้นซ่อนที่ 1
n_units_l1	165	จำนวนเซลล์ประสาทในชั้นซ่อนที่ 2
activation	relu	ฟังก์ชันกระตุ้นของชั้นซ่อน
alpha	0.008268	ค่าปรับโทษ L2 (Regularization)
learning_rate_init	0.0011898	อัตราการเรียนรู้เริ่มต้นของตัวเพิ่มประสิทธิภาพ Adam
batch_size	32	ขนาดของชุดข้อมูลย่อยในแต่ละรอบการฝึก
beta_1	0.95205	ค่าพารามิเตอร์โมเมนตัมอันดับ 1 ของ Adam
beta_2	0.98098	ค่าพารามิเตอร์โมเมนตัมอันดับ 2 ของ Adam
tol	8.145e-05	ค่าความคลาดเคลื่อนขั้นต่ำสำหรับเกณฑ์การหยุดฝึก

จากตารางที่ 3.17 ผลการปรับแต่งไฮเปอร์พารามิเตอร์ของแบบจำลอง MLP โดยใช้ Optuna พบว่าค่าพารามิเตอร์ที่เหมาะสมที่สุดประกอบด้วยโครงสร้างโครงข่ายจำนวน 2 ชั้นซ่อน โดยชั้นที่ 1 มีจำนวนเซลล์ประสาท 122 หน่วย และชั้นที่ 2 มีจำนวน 165 หน่วย นอกจากนี้มีการใช้ค่าปรับโทษ L2 (α) เท่ากับ 0.008268 เพื่อช่วยลดปัญหาการเกิด overfitting และใช้อัตราการเรียนรู้เริ่มต้นของตัวเพิ่มประสิทธิภาพ Adam เท่ากับ 0.0011898 เพื่อเพิ่มประสิทธิภาพในการฝึกแบบจำลอง

6. การปรับแต่งพารามิเตอร์ของ TabNet

สำหรับสถาปัตยกรรม TabNet ได้มีการปรับแต่งพารามิเตอร์ที่เกี่ยวข้องกับการแทนคุณลักษณะและกลไก Attention ซึ่งมีผลต่อความสามารถของแบบจำลองในการเลือกคุณลักษณะที่สำคัญในกระบวนการตัดสินใจ โดยผลการปรับพารามิเตอร์แสดงในตารางที่ 3.18

ตารางที่ 3.18 การปรับแต่งพารามิเตอร์ของ TabNet ด้วย Optuna

Parameter	Value	Description
n_d	22	มิติของ decision prediction layer

n_a	29	มิติของ attention embedding layer
n_steps	5	จำนวนขั้นตอนการตัดสินใจแบบลำดับ
gamma	1.1705	พารามิเตอร์ควบคุมการนำคุณลักษณะกลับมาใช้ซ้ำ
lambda_sparse	0.0003587	ค่าปรับโทษความเบาบางของ feature mask
mask_type	entmax	ฟังก์ชัน activation สำหรับการเลือกคุณลักษณะ
lr	0.006494	อัตราการเรียนรู้ของตัวเพิ่มประสิทธิภาพ
momentum	0.16929	ค่าโมเมนตัมของ Batch Normalization
batch_size	512	ขนาดของชุดข้อมูลย่อยในแต่ละรอบการฝึก
virtual_batch_size	128	ขนาดของ virtual batch สำหรับ Ghost Batch Normalization
scheduler_step_size	13	จำนวนรอบก่อนปรับลดอัตราการเรียนรู้
scheduler_gamma	0.94241	อัตราการลดของ learning rate scheduler
weight_decay	1.9594e-05	ค่าปรับโทษ L2 ของน้ำหนักโครงข่าย

จากตารางที่ 3.18 ผลการปรับแต่งไฮเปอร์พารามิเตอร์ของแบบจำลอง TabNet โดยใช้ Optuna พบว่าค่าพารามิเตอร์ที่เหมาะสมที่สุดประกอบด้วย ค่า n_d = 22 และ n_a = 29 ซึ่งกำหนดมิติของการแทนคุณลักษณะและกลไก attention ตามลำดับ โดยกำหนดจำนวนขั้นตอนการตัดสินใจเท่ากับ 5 และค่าพารามิเตอร์การผ่อนคลาย gamma เท่ากับ 1.1705 นอกจากนี้แบบจำลองถูกฝึกด้วยอัตราการเรียนรู้เท่ากับ 0.006494

7. การปรับแต่งพารามิเตอร์ของ FT-Transformer

สำหรับสถาปัตยกรรม FT-Transformer การปรับแต่งพารามิเตอร์มุ่งเน้นไปที่โครงสร้างของ Transformer ซึ่งมีผลต่อความสามารถของแบบจำลองในการเรียนรู้ความสัมพันธ์ระหว่างคุณลักษณะในข้อมูลเชิงตารางโดยผลการปรับแต่งพารามิเตอร์ แสดงในตารางที่ 3.19

ตารางที่ 3.19 การปรับแต่งพารามิเตอร์ของ FT-Transformer ด้วย Optuna

Parameter	Value	Description
d_block	128	มิติของการฝังตัวของคุณลักษณะ (feature embedding)
n_heads	2	จำนวนหัวในกลไก multi-head attention
n_blocks	3	จำนวนชั้นของ transformer encoder
ffn_d_hidden_multiplier	4	ตัวคูณขนาด feed-forward network เทียบกับ d_block
attention_dropout	0.14124	อัตรา dropout ภายในกลไก attention
ffn_dropout	0.24385	อัตรา dropout ภายใน feed-forward network
residual_dropout	0.015271	อัตรา dropout บนเส้นทาง residual connection
lr	0.0011315	อัตราการเรียนรู้ของตัวเพิ่มประสิทธิภาพ Adam
weight_decay	0.009106	ค่าปรับโทษ L2 ของน้ำหนักโครงข่าย
batch_size	512	ขนาดของชุดข้อมูลย่อยในแต่ละรอบการฝึก

epochs	20	จำนวนรอบการฝึกสูงสุด
patience	12	จำนวนรอบที่ยอมให้ประสิทธิภาพไม่ดีขึ้นก่อนหยุดฝึกก่อนกำหนด

จากตารางที่ 3.19 ผลการปรับแต่งไฮเปอร์พารามิเตอร์ของแบบจำลอง FT-Transformer โดยใช้เฟรมเวิร์ก Optuna พบว่าค่าพารามิเตอร์ที่เหมาะสมที่สุดประกอบด้วยมิติของ feature embedding (d_block) เท่ากับ 128 และจำนวน attention heads เท่ากับ 2 โดยสถาปัตยกรรม transformer encoder มีจำนวน 3 ชั้น (n_blocks) นอกจากนี้ใช้ dropout ในกลไก attention, feed-forward และ residual connection ที่ 0.14124, 0.24385 และ 0.015271 ตามลำดับ เพื่อช่วยลดปัญหาการเกิด overfitting และใช้อัตราการเรียนรู้ของตัวเพิ่มประสิทธิภาพ Adam เท่ากับ 0.0011315 ในกระบวนการฝึกแบบจำลอง

8. การปรับแต่งพารามิเตอร์ของ SAINT

SAINT เป็นสถาปัตยกรรมทรานส์ฟอร์เมอร์สำหรับข้อมูลตารางที่เพิ่มกลไก Intersample Attention เข้ากับ Self-Attention ในการศึกษาครั้งนี้ ผู้วิจัยพัฒนาแบบจำลอง SAINT โดยใช้เฟรมเวิร์ก PyTorch และปรับแต่งค่าพารามิเตอร์ด้วยเฟรมเวิร์ก Optuna ผลการปรับแต่งไฮเปอร์พารามิเตอร์ที่ดีที่สุดแสดงดังตารางที่ 3.20

ตารางที่ 3.20 การปรับแต่งพารามิเตอร์ของ SAINT ด้วย Optuna

Parameter	Value	Description
dim	32	มิติของเวกเตอร์ embedding ของคุณลักษณะ
depth	2	จำนวนชั้นของ Transformer Encoder (Self- และ Intersample Attention)
heads	2	จำนวนหัวของกลไก Multi-Head Attention
ff_mult	6	ตัวคูณขนาด Feed-Forward Network เทียบกับมิติ embedding
mlp_hidden_multiplier	2	ตัวคูณขนาดชั้นซ่อนของ MLP prediction head
dropout	0.10658	อัตรา Dropout สำหรับการทำให้ Regularization
lr	0.0003672	อัตราการเรียนรู้ของตัวเพิ่มประสิทธิภาพ Adam
weight_decay	0.0001468	ค่าปรับโทษ L2 ของน้ำหนักโครงข่าย
batch_size	512	ขนาดของชุดข้อมูลย่อยในแต่ละรอบการฝึก
gradient_clip_norm	1.1265	ค่าตัดขอบเขตของเกรเดียนต์เพื่อความปลอดภัยในการฝึก
epochs	15	จำนวนรอบการฝึกสูงสุด
patience	12	จำนวนรอบที่ยอมให้ประสิทธิภาพไม่ดีขึ้นก่อนหยุดฝึกก่อนกำหนด

จากตารางที่ 3.20 ผลการปรับแต่งไฮเปอร์พารามิเตอร์พบว่าค่าที่เหมาะสมที่สุดใช้สถาปัตยกรรมที่ค่อนข้างกระชับ (dim = 32, depth = 2, heads = 2) ร่วมกับอัตราการเรียนรู้ที่ต่ำมาก

(0.00037) ซึ่งสอดคล้องกับความจำเป็นในการฝึกโครงข่ายที่มีความซับซ้อนสูงให้มีเสถียรภาพบนชุดข้อมูลขนาดปานกลาง

9. การปรับแต่งพารามิเตอร์ของ TabTransformer

สำหรับแบบจำลอง TabTransformer การปรับแต่งพารามิเตอร์มุ่งเน้นไปที่โครงสร้างของ Transformer Encoder และพารามิเตอร์ที่เกี่ยวข้องกับการฝึก ซึ่งมีผลต่อความสามารถของแบบจำลองในการเรียนรู้ความสัมพันธ์ระหว่างคุณลักษณะในข้อมูลเชิงตาราง โดยผลของการปรับค่าพารามิเตอร์แสดงในตารางที่ 3.21

ตารางที่ 3.21 การปรับแต่งพารามิเตอร์ของ TabTransformer ด้วย Optuna

Parameter	Value	Description
dim	48	มิติของการฝังตัวของคุณลักษณะในแต่ละคอลัมน์
depth	1	จำนวนชั้นของ transformer encoder
heads	2	จำนวนหัวในกลไก multi-head attention
mlp_hidden_multiplier	2	ตัวคูณขนาดชั้นซ่อนของ MLP prediction head
attn_dropout	0.047646	อัตรา dropout ภายในกลไก attention
ff_dropout	0.087776	อัตรา dropout ภายใน feed-forward network
lr	0.0012539	อัตราการเรียนรู้ของตัวเพิ่มประสิทธิภาพ Adam
weight_decay	9.423e-05	ค่าปรับโทษ L2 ของน้ำหนักโครงข่าย
batch_size	512	ขนาดของชุดข้อมูลย่อยในแต่ละรอบการฝึก
epochs	10	จำนวนรอบการฝึกสูงสุด
patience	11	จำนวนรอบที่ยอมให้ประสิทธิภาพไม่ดีขึ้นก่อนหยุดฝึกก่อนกำหนด

จากตารางที่ 3.21 ผลการปรับแต่งไฮเปอร์พารามิเตอร์ของแบบจำลอง TabTransformer โดยใช้เฟรมเวิร์ก Optuna พบว่าค่าพารามิเตอร์ที่เหมาะสมที่สุดประกอบด้วยมิติของ column embedding เท่ากับ 48 และจำนวน attention heads เท่ากับ 2 โดยสถาปัตยกรรม transformer encoder มีเพียง 1 ชั้น (depth) ซึ่งดีกว่าโมเดลทรานส์ฟอร์เมอร์อื่นในการศึกษานี้อย่างชัดเจน นอกจากนี้มีการใช้ dropout ในกลไก attention และ feed-forward ที่ 0.047646 และ 0.087776 ตามลำดับ และใช้อัตราการเรียนรู้ของตัวเพิ่มประสิทธิภาพ Adam เท่ากับ 0.0012539

3.5 กลยุทธ์การตรวจสอบไขว้

การประเมินระหว่างการพัฒนาตัวแบบทุกขั้นตอน ทั้งการปรับค่าพารามิเตอร์และการประเมินผลเปรียบเทียบ ใช้วิธีการตรวจสอบไขว้แบบแบ่งชั้นจำนวน 5 ส่วน (Stratified 5-Fold Cross-Validation) อย่างสม่ำเสมอ เพื่อรักษาสัดส่วนของตัวแปรผลลัพธ์ในแต่ละส่วนย่อยให้ใกล้เคียงกับชุดข้อมูลฝึกทั้งหมด และเพื่อให้ได้ค่าประมาณประสิทธิภาพที่มีความแปรปรวนต่ำกว่าการแบ่งข้อมูลเพียงครั้งเดียว ในแต่ละรอบ ข้อมูล 4

ส่วนถูกใช้เป็นชุดฝึกภายในและ 1 ส่วนที่เหลือถูกใช้เป็นชุดตรวจสอบซึ่งไม่ผ่านการปรับสมดุลใด ๆ เพื่อให้สะท้อนการกระจายตัวของข้อมูลจริง

3.6 การวัดประสิทธิภาพของตัวแบบ

เมื่อผ่านการฝึกสอนตัวแบบเสร็จจะมีการวัดประสิทธิภาพของตัวแบบ จะถูกประเมินในแง่ปัญหาลำเนกประเภทแบบสองคลาส โดยใช้ตัวชี้วัด ได้แก่ ความแม่นยำ (Accuracy), ความเที่ยงตรง (Precision), ค่าความไว (Recall), ค่าคะแนน F1 (F1-score) และพื้นที่ใต้กราฟ ROC-AUC

3.7 วิเคราะห์ความสามารถในการอธิบายผล

ในขั้นตอนนี้จะนำแนวคิดด้าน Explainable Artificial Intelligence (XAI) มาใช้เพื่อเพิ่มความโปร่งใส (Transparency) และความสามารถในการตีความผลลัพธ์ของโมเดล (Interpretability) ในการวิจัยนี้ได้ดำเนินการวิเคราะห์เพื่อทำความเข้าใจทั้ง โครงสร้างความสัมพันธ์ของข้อมูล และ เหตุผลของการทำนายของโมเดล โดยใช้วิธีการวิเคราะห์ข้อมูลเชิงสถิติร่วมกับเทคนิคจากแนวคิด Explainable Artificial Intelligence เพื่อช่วยอธิบายว่าปัจจัยใดในข้อมูลที่มีผลต่อผลลัพธ์ของโมเดล และเพื่อเพิ่มความน่าเชื่อถือของผลการพยากรณ์ การวิเคราะห์แบ่งออกเป็น 2 วิธี ได้แก่

1. SHAP Model-Agnostic Explainable AI จะใช้ในการวิเคราะห์ความสำคัญของตัวแปรคุณลักษณะต่อผลการทำนายของโมเดลในระดับภาพรวมของข้อมูล (Global Explanation)
2. LIME Model-Agnostic Explainable AI ใช้ในการวิเคราะห์เหตุผลของการทำนายของโมเดลในระดับตัวอย่างข้อมูลเฉพาะหรือที่เรียกว่า (Local Explanation)

บทที่ 4

ผลการทดลอง

การวิจัยนี้มีวัตถุประสงค์เพื่อพัฒนาและเปรียบเทียบประสิทธิภาพของแบบจำลองการเรียนรู้ของเครื่อง และแบบจำลองโครงข่ายประสาทเทียมเชิงลึกสำหรับข้อมูลตารางจำนวน 9 แบบจำลอง ในการทำนายภาวะความดันโลหิตสูงพร้อมทั้งวิเคราะห์ตัวแปรที่มีอิทธิพลต่อการทำนายด้วยเทคนิค SHAP และ LIME ผลการวิจัยนำเสนอตามลำดับขั้นตอนของการทดลอง ดังรายละเอียดต่อไปนี้

4.1 ผลการเตรียมข้อมูล

ชุดข้อมูลที่ใช้ในการวิจัยมีจำนวนทั้งสิ้น 10,000 แถว และ 25 ตัวแปรและมีลักษณะทั่วไปของข้อมูลดังที่แสดงในตารางที่ 4.1 ประกอบด้วยข้อมูลด้านประชากรศาสตร์ ดัชนีมวลกาย พฤติกรรมสุขภาพ ประวัติทางการแพทย์ ตลอดจนผลตรวจทางห้องปฏิบัติการ โดยตัวแปรเป้าหมาย (hypertension_status) มีการกระจายแบบไม่สมดุล คือกลุ่ม Hypertensive ร้อยละ 81.05 (8,105 ราย) และกลุ่ม Normal ร้อยละ 18.95 (1,895 ราย)

ตารางที่ 4.1 ลักษณะทั่วไปของชุดข้อมูล

รายการ	ค่า
จำนวนข้อมูลทั้งหมด	10,000
จำนวนตัวแปร	25
Missing Value	medication_adherence 3,131 (31.31%)
Duplicate	0
จำนวนผู้ป่วย Hypertensive	8,105 (81.05%)
จำนวนกลุ่ม Normal	1,895 (18.95%)

จากการตรวจสอบคุณภาพข้อมูล พบว่าตัวแปร medication_adherence มีค่าว่าง (missing value) จำนวน 3,131 ระเบียบวินัย คิดเป็นร้อยละ 31.31 และไม่พบข้อมูลซ้ำ ในชุดข้อมูล ขั้นตอนการทำความสะอาดและเตรียมข้อมูลประกอบด้วย (1) ตัดตัวแปร patient_id ออกเนื่องจากเป็นรหัสระบุตัวตนที่ไม่มีความหมายเชิงทำนาย (2) เติมค่าว่างของ medication_adherence ด้วยค่า "No_Medication" (3) ตัดตัวแปรที่มีความเสี่ยงต่อการรั่วไหลของข้อมูล ออกจำนวน 6 ตัวแปร ได้แก่ diagnosed, on_medication, bp_controlled, medication_adherence, systolic_bp_mmhg และ diastolic_bp_mmhg เนื่องจากตัวแปรเหล่านี้เป็นผลลัพธ์ที่เกี่ยวข้องโดยตรงกับการวินิจฉัยหรือค่าความดันโลหิตที่ใช้กำหนดตัวแปรเป้าหมาย (4) เข้ารหัสตัว

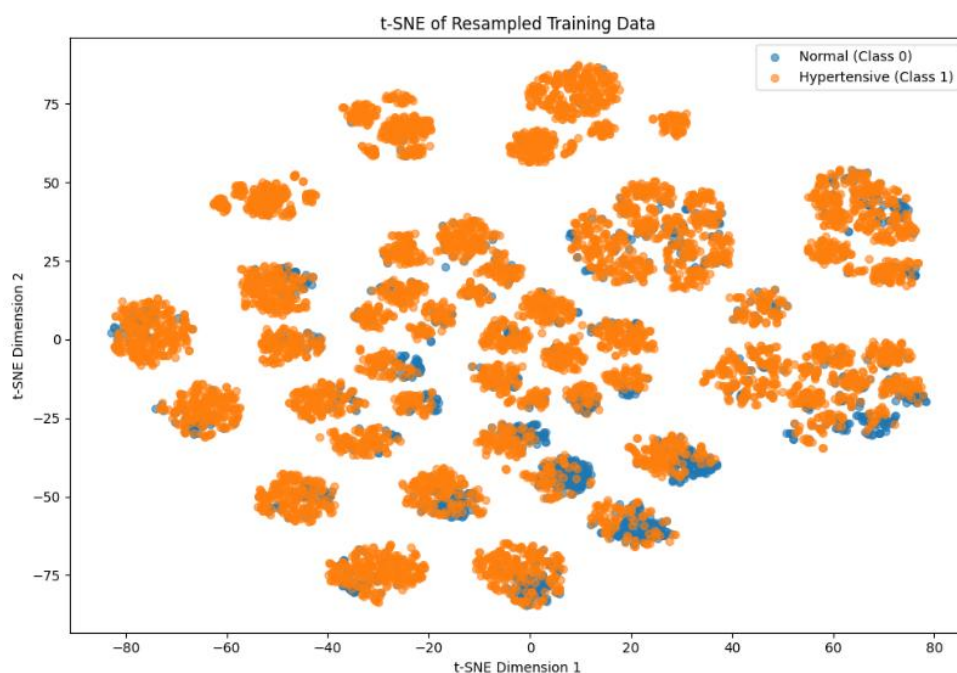
แปรเป้าหมายเป็นค่าไบนารี (Normal = 0, Hypertensive = 1) และ (5) แปลงตัวแปรเชิงกลุ่ม sex และ residence ด้วยวิธี One-Hot Encoding ทำให้ได้ตัวแปรทำนายที่ใช้ในการสร้างแบบจำลองทั้งสิ้น 17 ตัวแปร

จากนั้นแบ่งชุดข้อมูลออกเป็นชุดฝึกสอน (training set) และชุดทดสอบ (test set) ในสัดส่วนร้อยละ 80 ต่อ 20 ดังที่แสดงในตารางที่ 4.2 ด้วยวิธี Stratified Sampling เพื่อรักษาสัดส่วนของกลุ่มเป้าหมายให้ใกล้เคียงกันทั้งสองชุด ได้ชุดฝึกสอนจำนวน 8,000 ราย (Hypertensive 6,484 ราย, Normal 1,516 ราย) และชุดทดสอบจำนวน 2,000 ราย (Hypertensive 1,621 ราย, Normal 379 ราย)

ตารางที่ 4.2 การแบ่งชุดข้อมูล

ชุดข้อมูล	Hypertensive	Normal	รวม
Training	6,484	1,516	8,000
Test	1,621	379	2,000

สำหรับแบบจำลองกลุ่มต้นไม้ตัดสินใจ (Tree-based: XGBoost, RandomForest, LightGBM, CatBoost) ไม่มีการปรับมาตรฐานข้อมูล (scaling) ในขณะที่แบบจำลองกลุ่มโครงข่ายประสาทเทียมเชิงลึก (MLP, TabNet, FT-Transformer, SAINT, TabTransformer) มีการปรับมาตรฐานข้อมูลด้วย StandardScaler และได้ทำการสุ่มเพิ่มข้อมูลกล่าวคือแบบจำลองทุกตัวถูกฝึกสอนด้วยชุดข้อมูลที่มีการกระจายของกลุ่มเป้าหมายแบบไม่สมดุลตามธรรมชาติแสดงได้ตามภาพประกอบที่ 4.1 (81.05% ต่อ 18.95%) เนื่องจากกลุ่มข้อมูลที่ไม่สมดุลให้ค่าความถูกต้องให้ค่าความถูกต้อง (Accuracy) สูงที่สุดตามตารางที่ 3.7 และจำนวนสัดส่วนของข้อมูลทั้งหมดในการฝึกมีการกระจายดังภาพประกอบที่ 4.1



ภาพประกอบที่ 4.1 ภาพ t-SNE แสดงการกระจายของข้อมูลในชุดฝึกสอน

4.2 ผลการปรับแต่งไฮเปอร์พารามิเตอร์และการตรวจสอบไขว้

แบบจำลองทั้ง 9 แบบจำลอง ได้แก่ XGBoost, RandomForest, LightGBM, CatBoost, MLP, TabNet, FT-Transformer, SAINT และ TabTransformer ได้รับการปรับแต่งไฮเปอร์พารามิเตอร์ด้วยเฟรมเวิร์ก Optuna จำนวน 30 รอบการทดลอง (trials) ต่อแบบจำลอง ร่วมกับการตรวจสอบไขว้แบบ Stratified 5-Fold Cross-Validation บนชุดฝึกสอน ผลสรุปค่าเฉลี่ยและส่วนเบี่ยงเบนมาตรฐานของตัวชี้วัดประสิทธิภาพจากการตรวจสอบไขว้ 5 รอบ แสดงดังตารางที่ 4.3

ตารางที่ 4.3 ผลการตรวจสอบไขว้ของแบบจำลองพื้นฐานทั้งหมด

แบบจำลอง	AUC เฉลี่ย	SD ของ AUC	Accuracy	Precision	Recall	F1-Score	เวลาฝึกสอน (วินาที)
XGBoost	0.9236	0.0061	0.8880	0.9102	0.9562	0.9326	53.78
RandomForest	0.9141	0.0085	0.8534	0.9443	0.8704	0.9059	258.28
LightGBM	0.9236	0.0059	0.8891	0.9108	0.9570	0.9333	94.20
CatBoost	0.9234	0.0066	0.8887	0.9100	0.9574	0.9331	470.85
MLP	0.9213	0.0054	0.8889	0.9080	0.9602	0.9333	586.04
TabNet	0.9138	0.0051	0.8855	0.9104	0.9525	0.9309	2385.41
FT-Transformer	0.9234	0.0043	0.8871	0.9103	0.9553	0.9320	531.14
SAINT	0.9243	0.0050	0.8871	0.9069	0.9596	0.9323	400.04
TabTransformer	0.8522	0.0087	0.8470	0.8753	0.9465	0.9093	18831.77

จากตารางที่ 4.3 พบว่าแบบจำลอง SAINT มีค่า AUC เฉลี่ยจากการตรวจสอบไขว้สูงที่สุด (0.9243) รองลงมาคือ XGBoost และ LightGBM ซึ่งมีค่าใกล้เคียงกัน (0.9236) ตามด้วย CatBoost และ FT-Transformer (0.9234) ในขณะที่ TabTransformer มีค่า AUC เฉลี่ยต่ำที่สุดในกลุ่ม (0.8522) และมีความแปรปรวนระหว่างรอบการตรวจสอบไขว้สูงกว่าแบบจำลองอื่น (SD = 0.0087) เมื่อพิจารณาเวลาที่ใช้ในการฝึกสอน พบว่าแบบจำลองกลุ่ม Gradient Boosting (XGBoost, LightGBM) ใช้เวลาฝึกสอนน้อยที่สุด ในขณะที่ TabTransformer ใช้เวลาฝึกสอนมากที่สุดถึง 18,831.77 วินาที (ประมาณ 5.2 ชั่วโมง) ซึ่งสูงกว่าแบบจำลองอื่นอย่างมีนัยสำคัญในทางปฏิบัติ แม้จะให้ผลลัพธ์ด้านประสิทธิภาพต่ำที่สุดก็ตาม

4.3 ผลการประเมินแบบจำลองบนชุดข้อมูลทดสอบ

เมื่อนำแบบจำลองที่ผ่านการฝึกสอนด้วยพารามิเตอร์ที่ดีที่สุดไปประเมินผลบนชุดข้อมูลทดสอบ (test set, $n = 2,000$) ที่แบบจำลองไม่เคยพบเห็นมาก่อน ณ เกณฑ์การจำแนก (classification threshold) เท่ากับ 0.5 ได้ผลลัพธ์ดังตารางที่ 4.4 โดยเรียงลำดับตามค่า AUC สูงที่สุดไปหาต่ำที่สุด

ตารางที่ 4.4 ผลการประเมินประสิทธิภาพของแบบจำลองบนชุดข้อมูลทดสอบ

แบบจำลอง	AUC	Gini	Brier Score	Accuracy	Precision	Recall	F1-Score	เวลาฝึกสอน (วินาที)
XGBoost	0.9209	0.8418	0.0826	0.8830	0.9126	0.9463	0.9291	53.78
LightGBM	0.9201	0.8402	0.0825	0.8875	0.9135	0.9513	0.9320	94.20
CatBoost	0.9189	0.8377	0.0833	0.8855	0.9113	0.9513	0.9309	470.85
SAINT	0.9170	0.8340	0.0855	0.8825	0.9195	0.9371	0.9282	400.04
FT-Transformer	0.9151	0.8302	0.0849	0.8825	0.9081	0.9513	0.9292	531.14
MLP	0.9120	0.8241	0.0883	0.8790	0.9131	0.9402	0.9264	586.04
RandomForest	0.9075	0.8151	0.1135	0.8465	0.9451	0.8606	0.9009	258.28
TabNet	0.8929	0.7858	0.0980	0.8640	0.9056	0.9291	0.9172	2385.41
TabTransformer	0.8375	0.6750	0.1085	0.8545	0.8804	0.9494	0.9136	18831.77

จากตารางที่ 4.4 ผลการทดสอบบนชุดข้อมูลจริงมีแนวโน้มสอดคล้องกับผลจากการตรวจสอบไขว้ กล่าวคือ XGBoost ให้ค่า AUC สูงที่สุด (0.9209) รองลงมาคือ LightGBM (0.9201) และ CatBoost (0.9189) ตามลำดับ ซึ่งเป็นกลุ่มแบบจำลองที่ใช้เทคนิค Gradient Boosting บนต้นไม้ตัดสินใจ ในขณะที่แบบจำลองโครงข่ายประสาทเทียมสำหรับข้อมูลตาราง (SAINT และ FT-Transformer) ให้ผลลัพธ์ใกล้เคียงกับกลุ่ม Gradient Boosting (AUC 0.9170 และ 0.9151 ตามลำดับ) แต่ใช้เวลาฝึกสอนมากกว่าประมาณ 5–10 เท่า แบบจำลอง RandomForest แม้มีค่า Precision สูงสุด (0.9451) แต่มีค่า Recall ต่ำกว่าแบบจำลองอื่นอย่างชัดเจน (0.8606) และมีค่า Brier Score สูงที่สุดในกลุ่มแบบจำลองพื้นฐาน (0.1135) สะท้อนถึงความสามารถในการประมาณค่าความน่าจะเป็นที่ต่ำกว่า ส่วน TabTransformer ให้ผลลัพธ์ต่ำที่สุดในทุกตัวชี้วัดหลัก (AUC = 0.8375) แม้จะใช้เวลาฝึกสอนมากที่สุดในบรรดาแบบจำลองทั้งหมด

4.4 ผลการทดสอบนัยสำคัญทางสถิติของความแตกต่างด้านประสิทธิภาพ

เพื่อยืนยันว่าความแตกต่างของประสิทธิภาพระหว่างแบบจำลองทั้ง 9 แบบที่นำมาเปรียบเทียบในงานวิจัยนี้ ได้แก่ XGBoost, Random Forest, LightGBM, CatBoost, MLP, TabNet, FT-Transformer, SAINT และ TabTransformer มีนัยสำคัญทางสถิติจริง มิใช่เกิดจากความผันแปรโดยบังเอิญของข้อมูล ผู้วิจัยจึงดำเนินการทดสอบทางสถิติ 2 ขั้นตอน ได้แก่ (1) การทดสอบ Friedman Test เพื่อตรวจสอบว่าประสิทธิภาพของแบบจำลองทั้งหมดในภาพรวมมีความแตกต่างกันอย่างมีนัยสำคัญทางสถิติหรือไม่ และ (2) หากพบความแตกต่างอย่างมีนัยสำคัญ จึงดำเนินการทดสอบเปรียบเทียบรายคู่ (post-hoc pairwise comparison) ด้วย Paired t-test ร่วมกับการปรับค่า p-value ด้วยวิธี Holm correction ในหัวข้อถัดไป

4.4.1 ผลการทดสอบ Friedman Test

Friedman Test เป็นสถิติแบบไม่อิงพารามิเตอร์ ที่เหมาะสำหรับการเปรียบเทียบแบบจำลองมากกว่า 2 แบบขึ้นไปที่ทดสอบบนชุดข้อมูลเดียวกันภายใต้การตรวจสอบไขว้ โดยอาศัยหลักการจัดอันดับประสิทธิภาพของแต่ละแบบจำลองในแต่ละ fold แล้วเปรียบเทียบผลรวมอันดับระหว่างแบบจำลองทั้งหมด ซึ่งเหมาะสมกับข้อมูลในงานวิจัยนี้เนื่องจากไม่จำเป็นต้องมีข้อสมมติเรื่องการแจกแจงแบบปกติของค่าประสิทธิภาพ

ตารางที่ 4.5 อันดับเฉลี่ยของแต่ละแบบจำลองจากการตรวจสอบไขว้

แบบจำลอง	Average Rank	
	AUC	Accuracy
XGBoost	3.60	4.10
Random Forest	7.20	8.60
LightGBM	3.20	3.20
CatBoost	3.80	3.60
MLP	5.00	3.40
TabNet	7.60	5.40
FT-Transformer	3.20	4.00
SAINT	2.40	4.30
TabTransformer	9.00	8.40

จากตารางที่ 4.5 แสดงค่าอันดับเฉลี่ยของแต่ละแบบจำลองทั้ง 9 แบบจากผลการตรวจสอบไขว้ โดยค่าอันดับเฉลี่ยที่ต่ำกว่าหมายถึงแบบจำลองมีประสิทธิภาพดีกว่า เมื่อพิจารณาค่า AUC พบว่า SAINT มีค่าอันดับเฉลี่ยที่ดีที่สุด (2.40) รองลงมาคือ LightGBM และ FT-Transformer (3.20) ขณะที่ TabTransformer มีค่าอันดับเฉลี่ยสูงที่สุด (9.00) รองลงมาคือ TabNet (7.60) และ Random Forest (7.20)

สำหรับค่า Accuracy พบว่า LightGBM มีค่าอันดับเฉลี่ยที่ดีที่สุด (3.20) รองลงมาคือ MLP (3.40) และ CatBoost (3.60) ส่วน Random Forest (8.60) และ TabTransformer (8.40) มีค่าอันดับเฉลี่ยสูงที่สุด โดยภาพรวม แบบจำลองในกลุ่ม Gradient Boosting และแบบจำลองทรานส์ฟอร์เมอร์บางส่วน ได้แก่ FT-Transformer และ SAINT มีอันดับเฉลี่ยอยู่ในกลุ่มที่ดีที่สุด ขณะที่ Random Forest, TabNet และ TabTransformer มีอันดับเฉลี่ยอยู่ในลำดับท้าย

ตารางที่ 4.6 ผลการทดสอบ Friedman Test

ตัวชี้วัด	Friedman Statistic	p-value	ผลการทดสอบ
AUC	29.4930	0.0003	แตกต่างอย่างมีนัยสำคัญทางสถิติ
Accuracy	23.3550	0.0029	แตกต่างอย่างมีนัยสำคัญทางสถิติ

จากตารางที่ 4.6 แสดงผลการทดสอบพบว่า AUC มีค่าสถิติ Friedman เท่ากับ 29.4930 และค่า p-value เท่ากับ 0.0003 ขณะที่ด้าน Accuracy มีค่าสถิติ Friedman เท่ากับ 23.3550 และค่า p-value เท่ากับ 0.0029 ซึ่งทั้งสองกรณีมีค่า p-value น้อยกว่า 0.05 จึงปฏิเสธสมมติฐานหลัก และสรุปได้ว่าประสิทธิภาพของแบบจำลองทั้ง 9 แบบมีความแตกต่างกันอย่างมีนัยสำคัญทางสถิติทั้งในด้าน AUC และ Accuracy อย่างไรก็ตาม Friedman Test สามารถบ่งชี้ได้เพียงว่ามีความแตกต่างเกิดขึ้นในภาพรวม แต่ไม่สามารถระบุได้ว่าแบบจำลองคู่ใดแตกต่างกัน ดังนั้น ผู้วิจัยจึงดำเนินการเปรียบเทียบรายคู่ด้วย Paired t-test ร่วมกับการปรับค่า p-value ด้วยวิธี Holm correction ดังแสดงในหัวข้อ 4.4.2

4.4.2 ผลการทดสอบ Paired t-test และ Holm correction

จากผลการทดสอบ Friedman Test ในหัวข้อ 4.4.1 พบว่าแบบจำลองทั้ง 9 แบบมีประสิทธิภาพแตกต่างกันอย่างมีนัยสำคัญทางสถิติทั้งในด้านค่า AUC และ Accuracy ผู้วิจัยจึงดำเนินการทดสอบเปรียบเทียบรายคู่ด้วยสถิติ Paired t-test ภายในแต่ละกลุ่มแบบจำลองที่มีลักษณะสถาปัตยกรรมใกล้เคียงกันได้แก่ กลุ่มแบบจำลองต้นไม้ กลุ่มแบบจำลองการเรียนรู้เชิงลึก และกลุ่มแบบจำลองทรานส์ฟอร์มเมอร์ เพื่อระบุว่าแบบจำลองคู่ใดมีประสิทธิภาพแตกต่างกันอย่างมีนัยสำคัญ ทั้งนี้เนื่องจากในแต่ละกลุ่มมีการเปรียบเทียบมากกว่าหนึ่งคู่พร้อมกัน จึงจำเป็นต้องปรับค่า p-value ด้วยวิธี Holm correction เพื่อควบคุมอัตราความผิดพลาดแบบ Type I ที่อาจสะสมจากการทดสอบซ้ำหลายครั้ง โดยกำหนดระดับนัยสำคัญที่ $\alpha = 0.05$ หากค่า p-value ที่ปรับด้วย Holm correction แล้วมีค่าน้อยกว่า 0.05 จึงจะสรุปได้ว่าแบบจำลองทั้งสองมีประสิทธิภาพแตกต่างกันอย่างมีนัยสำคัญทางสถิติ

4.4.2.1 กลุ่มแบบจำลองต้นไม้

ผลการเปรียบเทียบรายคู่ในกลุ่มแบบจำลองต้นไม้ ซึ่งประกอบด้วย XGBoost, Random Forest, LightGBM และ CatBoost แสดงในตารางที่ 4.7 (ค่า AUC) และตารางที่ 4.8 (ค่า Accuracy) โดยการเปรียบเทียบใช้การทดสอบ Paired t-test จากผลการประเมินในแต่ละ Fold ของการทำ Stratified 5-Fold Cross Validation เนื่องจากแบบจำลองทั้งหมดได้รับการประเมินบนชุดข้อมูลเดียวกันในแต่ละรอบการทดสอบ

ตารางที่ 4.7 ผลการเปรียบเทียบกลุ่มแบบจำลองต้นไม้ของค่า AUC

คู่เปรียบเทียบ	t	p-value	Holm	สรุป
XGBoost vs Random Forest	3.913	0.017	0.086	ไม่มีนัยสำคัญ
XGBoost vs LightGBM	-0.168	0.875	1.00	ไม่มีนัยสำคัญ
XGBoost vs CatBoost	0.041	0.969	1.00	ไม่มีนัยสำคัญ
Random Forest vs LightGBM	-4.262	0.013	0.078	ไม่มีนัยสำคัญ
Random Forest vs CatBoost	-2.739	0.052	0.208	ไม่มีนัยสำคัญ
LightGBM vs CatBoost	0.074	0.944	1.00	ไม่มีนัยสำคัญ

จากตารางที่ 4.7 พบว่าเมื่อพิจารณาค่า p-value ก่อนการปรับ (unadjusted) คู่ XGBoost vs Random Forest ($p = 0.017$) และคู่ Random Forest vs LightGBM ($p = 0.013$) มีความแตกต่างกันอย่างมีนัยสำคัญทางสถิติ อย่างไรก็ตาม เมื่อปรับค่าด้วยวิธี Holm correction แล้ว พบว่าค่า p-value ของทั้งสองคู่เพิ่มขึ้นเป็น 0.086 และ 0.078 ตามลำดับ ซึ่งมากกว่าระดับนัยสำคัญที่กำหนด ($\alpha = 0.05$) จึงสรุปได้ว่า ไม่มีคู่แบบจำลองใดในกลุ่มแบบจำลองต้นไม้ที่มีค่า AUC แตกต่างกันอย่างมีนัยสำคัญทางสถิติ หลังการปรับค่าด้วย Holm correction แม้ว่าคู่ XGBoost vs Random Forest และ Random Forest vs LightGBM จะมีแนวโน้มความแตกต่างที่ใกล้เคียงกับระดับนัยสำคัญมากที่สุดก็ตาม แสดงให้เห็นว่าประสิทธิภาพด้าน AUC ของแบบจำลองกลุ่มต้นไม้ทั้งสี่แบบไม่แตกต่างกันอย่างมีนัยสำคัญเมื่อพิจารณาความเข้มงวดของการปรับค่าสถิติแบบ multiple comparisons

ตารางที่ 4.8 ผลการเปรียบเทียบกลุ่มแบบจำลองต้นไม้ของค่า Accuracy

คู่เปรียบเทียบ	t	p-value	Holm	สรุป
XGBoost vs Random Forest	5.92	0.004	0.024	แตกต่างอย่างมีนัยสำคัญ
XGBoost vs LightGBM	-0.14	0.894	1.000	ไม่มีนัยสำคัญ
XGBoost vs CatBoost	0.23	0.829	1.000	ไม่มีนัยสำคัญ
Random Forest vs LightGBM	-5.11	0.007	0.035	แตกต่างอย่างมีนัยสำคัญ
Random Forest vs CatBoost	-4.01	0.016	0.064	ไม่มีนัยสำคัญ
LightGBM vs CatBoost	0.63	0.561	1.000	ไม่มีนัยสำคัญ

ในทางตรงกันข้าม ผลการเปรียบเทียบค่า Accuracy ในตารางที่ 4.8 พบว่ามีคู่แบบจำลองที่แตกต่างกันอย่างมีนัยสำคัญทางสถิติแม้หลังปรับค่าด้วย Holm correction แล้ว 2 คู่ ได้แก่ คู่ XGBoost vs Random Forest ($t = 5.92$, $p = 0.004$, Holm-adjusted $p = 0.024$) ซึ่งมีนัยสำคัญทางสถิติ คู่ Random Forest vs LightGBM ($t = -5.11$, $p = 0.007$, Holm-adjusted $p = 0.035$) ซึ่งมีนัยสำคัญทางสถิติเช่นกัน ส่วนคู่ XGBoost vs LightGBM, XGBoost vs CatBoost, Random Forest vs CatBoost และ LightGBM vs CatBoost มีค่า Holm-adjusted p-value มากกว่า 0.05 ทั้งหมด จึงไม่มีความแตกต่างกันอย่างมีนัยสำคัญทางสถิติ ผลดังกล่าวชี้ให้เห็นว่า Random Forest มีค่า Accuracy แตกต่างจาก XGBoost และ LightGBM อย่างมีนัยสำคัญทางสถิติ ซึ่งสอดคล้องกับผลอันดับเฉลี่ย (Average Rank) ในตารางที่ 4.5 ที่ Random Forest มีอันดับ Accuracy อยู่ที่ 8.60 ซึ่งต่ำกว่าแบบจำลองอื่นในกลุ่มอย่างชัดเจน ในขณะที่ค่า AUC ของแบบจำลองทั้งสี่แบบไม่แตกต่างกันอย่างมีนัยสำคัญ แสดงให้เห็นว่าความแตกต่างด้านประสิทธิภาพของกลุ่มแบบจำลองต้นไม้ปรากฏชัดเจนในมิติของ Accuracy มากกว่ามิติของ AUC

4.4.2.2 กลุ่มแบบจำลองการเรียนรู้เชิงลึก

การเปรียบเทียบในกลุ่มแบบจำลองการเรียนรู้เชิงลึกประกอบด้วยแบบจำลอง MLP และ TabNet เนื่องจากการเปรียบเทียบเพียงคู่เดียว จึงไม่จำเป็นต้องปรับค่าด้วย Holm correction

ตารางที่ 4.9 ผลการเปรียบเทียบกลุ่มการเรียนรู้เชิงลึกของค่า AUC

คู่เปรียบเทียบ	t	p-value	สรุป
MLP vs TabNet	3.46	0.0260	แตกต่างกันอย่างมีนัยสำคัญ

จากตารางที่ 4.9 ผลการทดสอบพบว่าคู่ MLP vs TabNet มีค่า AUC แตกต่างกันอย่างมีนัยสำคัญทางสถิติ ($t = 3.46$, $p = 0.026$) โดยเมื่อพิจารณาจากอันดับเฉลี่ยในตารางที่ 4.5 พบว่า MLP มีอันดับ AUC เฉลี่ยดีกว่า TabNet (5.00 เทียบกับ 7.60) สะท้อนว่า MLP มีประสิทธิภาพด้าน AUC ดีกว่า TabNet อย่างมีนัยสำคัญทางสถิติ

ตารางที่ 4.10 ผลการเปรียบเทียบกลุ่มการเรียนรู้เชิงลึกของค่า Accuracy

คู่เปรียบเทียบ	t	p-value	สรุป
MLP vs TabNet	2.12	0.1010	ไม่มีนัยสำคัญ

อย่างไรก็ตาม จากตารางที่ 4.10 เมื่อพิจารณาค่า Accuracy พบว่าคู่ MLP vs TabNet ไม่มีความแตกต่างกันอย่างมีนัยสำคัญทางสถิติ ($t = 2.12$, $p = 0.101$) แสดงให้เห็นว่าแม้ MLP จะมีค่า AUC ที่ดีกว่า TabNet อย่างมีนัยสำคัญ แต่ในด้าน Accuracy ทั้งสองแบบจำลองมีประสิทธิภาพไม่แตกต่างกันอย่างมีนัยสำคัญทางสถิติ

4.4.2.3 กลุ่มแบบจำลองทรานส์ฟอร์มเมอร์

การเปรียบเทียบในกลุ่มแบบจำลองทรานส์ฟอร์มเมอร์ประกอบด้วย FT-Transformer (FT), SAINT และ TabTransformer ผลการทดสอบแสดงในตารางที่ 4.11 (AUC) และตารางที่ 4.12 (Accuracy)

ตารางที่ 4.11 ผลการเปรียบเทียบกลุ่มทรานส์ฟอร์มเมอร์ของค่า AUC

คู่เปรียบเทียบ	t	p-value	Holm	สรุป
FT vs SAINT	-0.57	0.6	1.000	ไม่มีนัยสำคัญ
FT vs TabTransformer	13.7	<0.001	<0.001	แตกต่างกันอย่างมีนัยสำคัญ
SAINT vs TabTransformer	11.8	<0.001	<0.001	แตกต่างกันอย่างมีนัยสำคัญ

จากตารางที่ 4.11 ผลการทดสอบพบว่า คู่ FT vs SAINT ไม่มีความแตกต่างกันอย่างมีนัยสำคัญทางสถิติ ($t = -0.57$, $p = 0.600$, Holm = 1.000) คู่ FT vs TabTransformer มีความแตกต่างกัน

อย่างมีนัยสำคัญทางสถิติ ($t = 13.70$, $p < 0.001$, $\text{Holm} < 0.001$) คู่ SAINT vs TabTransformer มีความแตกต่างกันอย่างมีนัยสำคัญทางสถิติ ($t = 11.80$, $p < 0.001$, $\text{Holm} < 0.001$)

ตารางที่ 4.12 ผลการเปรียบเทียบกลุ่มทราแนสฟอร์เมอร์ของค่า Accuracy

คู่เปรียบเทียบ	t	p-value	Holm	สรุป
FT vs SAINT	0.05	0.962	1	ไม่มีนัยสำคัญ
FT vs TabTransformer	4.77	0.009	0.018	แตกต่างกันอย่างมีนัยสำคัญ
SAINT vs TabTransformer	6.08	0.004	0.012	แตกต่างกันอย่างมีนัยสำคัญ

จากตารางที่ 4.12 ผลการทดสอบด้าน Accuracy มีแนวโน้มไปในทิศทางเดียวกัน กล่าวคือ: คู่ FT vs SAINT ไม่มีความแตกต่างกันอย่างมีนัยสำคัญทางสถิติ ($t = 0.05$, $p = 0.962$, $\text{Holm} = 1.000$) คู่ FT vs TabTransformer มีความแตกต่างกันอย่างมีนัยสำคัญทางสถิติ ($t = 4.77$, $p = 0.009$, $\text{Holm} = 0.018$) คู่ SAINT vs TabTransformer มีความแตกต่างกันอย่างมีนัยสำคัญทางสถิติ ($t = 6.08$, $p = 0.004$, $\text{Holm} = 0.012$)

ผลการทดสอบทั้งสองมิติสอดคล้องกันอย่างชัดเจนว่า TabTransformer มีประสิทธิภาพดีกว่า FT-Transformer และ SAINT อย่างมีนัยสำคัญทางสถิติทั้งในด้าน AUC และ Accuracy ในขณะที่ FT-Transformer และ SAINT มีประสิทธิภาพไม่แตกต่างกันอย่างมีนัยสำคัญทางสถิติทั้งสองมิติ ซึ่งสอดคล้องกับผลอันดับเฉลี่ยในตารางที่ 4.5 ที่ TabTransformer มีอันดับ AUC และ Accuracy อยู่ที่ 9.00 และ 8.40 ตามลำดับ ซึ่งต่ำที่สุดหรือใกล้เคียงต่ำที่สุดในบรรดาแบบจำลองทั้งหมด

4.5 ผลการวิเคราะห์ความสำคัญของตัวแปร

เพื่อสร้างความสามารถในการอธิบายผล (Explainability) ของแบบจำลอง การวิจัยนี้ใช้เทคนิค SHAP (SHapley Additive exPlanations) กับแบบจำลองทั้ง 9 แบบจำลอง โดยใช้ TreeExplainer สำหรับแบบจำลองกลุ่มต้นไม้ตัดสินใจ และใช้ Kernel/Deep Explainer ร่วมกับชุดข้อมูลอ้างอิง (background sample) สำหรับแบบจำลองกลุ่มโครงข่ายประสาทเทียม ควบคู่กับเทคนิค LIME (Local Interpretable Model-agnostic Explanations) ซึ่งวิเคราะห์บนแบบจำลอง XGBoost จากตัวอย่างชุดข้อมูลทดสอบจำนวน 50 ตัวอย่าง ผลการวิเคราะห์ความสอดคล้อง (agreement) ระหว่างค่าความสำคัญของตัวแปรที่ได้จาก LIME, SHAP และค่าสัมประสิทธิ์สหสัมพันธ์ (Correlation) กับตัวแปรเป้าหมาย โดยเรียงลำดับตัวแปรที่มีอิทธิพลสูงสุด 10 อันดับแรก แสดงดังตารางที่ 4.13 โดยเปรียบเทียบระหว่าง LIME, SHAP และค่าสัมประสิทธิ์สหสัมพันธ์ ซึ่งใช้แบบจำลอง XGBoost

ตารางที่ 4.13 ค่าความสำคัญของตัวแปร 10 อันดับแรก

ตัวแปร	LIME	SHAP	Correlation
creatinine_mg_dl	0.2508	0.2561	0.1371

ตัวแปร	LIME	SHAP	Correlation
ldl_mg_dl	0.1585	0.1122	0.1145
total_cholesterol_mg_dl	0.1401	0.1098	0.1150
age	0.1150	0.0756	0.0677
bmi	0.0948	0.0629	0.0591
hdl_mg_dl	0.0818	0.0551	0.0782
high_salt_diet	0.0517	0.0660	0.0687
physically_active	0.0339	0.0515	0.0588
residence_Urban	0.0274	0.0438	0.0620
family_history_hypertension	0.0183	0.0539	0.0546

ผลการวิเคราะห์พบว่าตัวแปรที่มีอิทธิพลสูงสุดต่อการทำนายภาวะความดันโลหิตสูงอย่างสม่ำเสมอในทั้งสามวิธี ได้แก่ ระดับครีเอตินิน (creatinine_mg_dl) ระดับ LDL (ldl_mg_dl) คอเลสเตอรอลรวม (total_cholesterol_mg_dl) อายุ (age) และการบริโภคอาหารเค็ม (high_salt_diet) ซึ่งเป็นปัจจัยเสี่ยงที่สอดคล้องกับองค์ความรู้ทางการแพทย์เกี่ยวกับโรคความดันโลหิตสูง เมื่อวัดความสอดคล้องเชิงอันดับ (rank agreement) ด้วยค่าสัมประสิทธิ์สหสัมพันธ์แบบสเปียร์แมน (Spearman's rank correlation) พบว่า LIME กับ SHAP มีความสอดคล้องกันในระดับสูง ($\rho = 0.863$) LIME กับ Correlation มีความสอดคล้องกันสูง ($\rho = 0.892$) และ SHAP กับ Correlation มีความสอดคล้องกันสูงที่สุด ($\rho = 0.939$) ซึ่งยืนยันว่าตัวแปรที่แบบจำลองให้ความสำคัญนั้นมีความน่าเชื่อถือและไม่ขึ้นอยู่กับวิธีการอธิบายผลวิธีใดวิธีหนึ่งเพียงอย่างเดียว

4.6 ผลการวิเคราะห์ความสอดคล้องของค่าความน่าจะเป็น

การวิเคราะห์ความสอดคล้องของค่าความน่าจะเป็น (Probability Calibration) มีความสำคัญต่อการนำแบบจำลองไปใช้ในบริบททางคลินิก เนื่องจากค่าความน่าจะเป็นที่แบบจำลองทำนายควรสะท้อนความเสี่ยงที่เกิดขึ้นจริง การวิจัยนี้ประเมินความสอดคล้องด้วยค่า Brier Score (ยิ่งน้อยยิ่งดี) และค่าช่องว่างการปรับเทียบเฉลี่ย (Mean Calibration Gap) จากเส้นโค้งการปรับเทียบ (Calibration Curve) แบ่งเป็น 10 ช่วง (bins)

ตารางที่ 4.14 ผลการวิเคราะห์ Brier Score และ Mean Calibration Gap ของแบบจำลองทั้งหมด

แบบจำลอง	Brier Score	Mean Calibration Gap
LightGBM	0.0825	0.0094
XGBoost	0.0826	0.0089
CatBoost	0.0833	0.0096

แบบจำลอง	Brier Score	Mean Calibration Gap
FT-Transformer	0.0849	0.0215
SAINT	0.0855	0.0256
MLP	0.0883	0.0289
TabNet	0.0980	0.0351
TabTransformer	0.1085	0.0247
RandomForest	0.1135	0.1371

ผลการวิเคราะห์พบว่า LightGBM มีค่า Brier Score ต่ำที่สุด (0.0825) รองลงมาคือ XGBoost (0.0826) ซึ่งเป็นสองแบบจำลองที่มีทั้งค่า AUC สูงและค่าความน่าจะเป็นที่ปรับเทียบได้ดีที่สุด โดย XGBoost มีค่า Mean Calibration Gap ต่ำที่สุดในบรรดาแบบจำลองทั้งหมด (0.0089) ในทางตรงกันข้าม RandomForest มีค่า Brier Score สูงที่สุด (0.1135) และค่า Mean Calibration Gap สูงที่สุดอย่างชัดเจน (0.1371) ซึ่งบ่งชี้ว่าค่าความน่าจะเป็นที่ RandomForest ทำนายมีความคลาดเคลื่อนจากความจริงที่เกิดขึ้นจริงมากกว่าแบบจำลองอื่นอย่างมีนัยสำคัญในทางปฏิบัติ

4.8 สรุปผลการวิจัยในภาพรวม

โดยสรุป ผลการวิจัยแสดงให้เห็นว่าแบบจำลองส่วนใหญ่จากทั้ง 9 แบบจำลองสามารถทำนายภาวะความดันโลหิตสูงได้อย่างมีประสิทธิภาพ โดยมีค่า AUC สูงกว่า 0.90 ยกเว้น TabTransformer แบบจำลองในกลุ่ม Gradient Boosting (XGBoost, LightGBM, CatBoost) ให้ผลลัพธ์ด้านความแม่นยำสูงสุดโดยใช้เวลาฝึกสอนน้อยที่สุด ในขณะที่แบบจำลองโครงข่ายประสาทเทียมสำหรับข้อมูลตาราง (FT-Transformer, SAINT) ให้ผลลัพธ์ใกล้เคียงกันแต่ใช้ทรัพยากรการคำนวณสูงกว่ามาก การทดสอบทางสถิติ (Paired t-test, Friedman test) ยืนยันว่าความแตกต่างของประสิทธิภาพระหว่างแบบจำลองกลุ่มบนของตารางไม่มีนัยสำคัญทางสถิติ ในขณะที่ TabTransformer มีประสิทธิภาพต่ำกว่าอย่างมีนัยสำคัญ การวิเคราะห์ความสำคัญของตัวแปรด้วย SHAP และ LIME มีความสอดคล้องกันสูงและระบุปัจจัยเสี่ยงที่สอดคล้องกับความรู้ทางการแพทย์ ได้แก่ ตรีอะตินิน ไชมัน LDL คอเลสเตอรอลรวม อายุ และการบริโภคอาหารเค็ม และผลการวิเคราะห์ Calibration ชี้ให้เห็นว่า LightGBM และ XGBoost เป็นแบบจำลองที่มีค่าความน่าจะเป็นสอดคล้องกับความเป็นจริงมากที่สุด ผลการวิจัยเหล่านี้จะถูกนำมาอภิปรายและสรุปเพื่อเสนอแนะแนวทางการนำไปประยุกต์ใช้ในบทที่ 5 ต่อไป

บทที่ 5

สรุปผลการวิจัย อภิปรายผล และข้อเสนอแนะ

การวิจัยนี้มีวัตถุประสงค์เพื่อประเมินและเปรียบเทียบประสิทธิภาพของเทคนิคการเรียนรู้ของเครื่อง (Machine Learning) และการเรียนรู้เชิงลึก (Deep Learning) สำหรับการจำแนกภาวะความดันโลหิตสูงจากข้อมูลทางคลินิก และวิเคราะห์ปัจจัยเสี่ยงสำคัญผ่านเทคนิค Explainable Artificial Intelligence (XAI) ผลการวิจัยสรุปได้ดังนี้

5.1 สรุปผลการวิจัย

การวิจัยนี้พัฒนาและเปรียบเทียบแบบจำลองการเรียนรู้ของเครื่อง 9 แบบจำลอง (XGBoost, RandomForest, LightGBM, CatBoost, MLP, TabNet, FT-Transformer, SAINT และ TabTransformer) สำหรับทำนายภาวะความดันโลหิตสูงจากข้อมูลผู้ป่วยจำนวน 10,000 ราย ผลการวิจัยสรุปได้ดังนี้

5.1.1 ประสิทธิภาพการทำนาย

ผลการทดลองแสดงให้เห็นว่าแบบจำลองส่วนใหญ่จากทั้ง 9 แบบจำลองที่ศึกษา ได้แก่ XGBoost, Random Forest, LightGBM, CatBoost, Multi-Layer Perceptron (MLP), TabNet, FT-Transformer, SAINT และ TabTransformer สามารถทำนายภาวะความดันโลหิตสูงได้อย่างมีประสิทธิภาพ โดยมีค่า AUC (Area Under the Curve) สูงกว่า 0.90 ซึ่งแสดงถึงศักยภาพในการจำแนกสูง แบบจำลองที่มีประสิทธิภาพสูงสุดคือ XGBoost ซึ่งให้ค่า AUC เท่ากับ 0.9209 บนชุดข้อมูลทดสอบ รองลงมาคือ LightGBM (AUC = 0.9201) และ CatBoost (AUC = 0.9189) ตามลำดับ แบบจำลองทั้งสามนี้เป็นสมาชิกของกลุ่มแบบจำลอง Gradient Boosting ซึ่งใช้เทคนิคการรวมต้นไม้ตัดสินใจแบบต่อเนื่อง (Sequential Ensemble) เพื่อปรับปรุงประสิทธิภาพ แบบจำลองโครงข่ายประสาทเทียมสำหรับข้อมูลตาราง (FT-Transformer และ SAINT) ให้ผลลัพธ์ใกล้เคียงกับกลุ่ม Gradient Boosting โดย SAINT มีค่า AUC เท่ากับ 0.9170 และ FT-Transformer มีค่า AUC เท่ากับ 0.9151 อย่างไรก็ตาม แบบจำลองเหล่านี้ใช้เวลาฝึกสอนมากกว่าประมาณ 5-10 เท่า เมื่อเทียบกับแบบจำลองกลุ่ม Gradient Boosting ในทางตรงกันข้าม TabTransformer ให้ผลลัพธ์ต่ำที่สุดในทุกตัวชี้วัดหลัก โดยมีค่า AUC เท่ากับ 0.8375 บนชุดข้อมูลทดสอบ แม้จะใช้เวลาฝึกสอนมากที่สุดในบรรดาแบบจำลองทั้งหมด (18,831.77 วินาที หรือประมาณ 5.2 ชั่วโมง) ผลนี้บ่งชี้ว่าสถาปัตยกรรม TabTransformer ไม่เหมาะสมสำหรับชุดข้อมูลนี้ หรืออาจต้องการการปรับแต่งไฮเปอร์พารามิเตอร์ที่ซับซ้อนมากขึ้น

5.1.2 นัยสำคัญทางสถิติ

เพื่อยืนยันว่าความแตกต่างของประสิทธิภาพระหว่างแบบจำลองมีนัยสำคัญทางสถิติจริง มิใช่เกิดจากความผันแปรโดยบังเอิญ ผู้วิจัยได้ดำเนินการทดสอบทางสถิติหลายขั้นตอน ผลการทดสอบ Friedman Test แสดงว่าประสิทธิภาพของแบบจำลองทั้ง 9 แบบมีความแตกต่างกันอย่างมีนัยสำคัญทางสถิติ โดยค่า

AUC มีค่าสถิติ Friedman เท่ากับ 29.4930 ($p\text{-value} = 0.0003$) และค่า Accuracy มีค่าสถิติ Friedman เท่ากับ 23.3550 ($p\text{-value} = 0.0029$) ทั้งสองกรณีมีค่า $p\text{-value}$ น้อยกว่า 0.05 จึงปฏิเสธสมมติฐานหลัก ผลการทดสอบ Paired t-test ร่วมกับ Holm correction ภายในแต่ละกลุ่มแบบจำลอง เปิดเผยรายละเอียดเพิ่มเติม

5.1.3 แบบจำลอง

5.1.3.1 กลุ่มแบบจำลองต้นไม้

กลุ่มแบบจำลองต้นไม้ (XGBoost, Random Forest, LightGBM, CatBoost) ไม่มีคู่แบบจำลองใดที่มีค่า AUC แตกต่างกันอย่างมีนัยสำคัญทางสถิติหลังการปรับค่าด้วย Holm correction แม้ว่าคู่ XGBoost vs Random Forest และ Random Forest vs LightGBM จะมีแนวโน้มความแตกต่างที่ใกล้เคียงกับระดับนัยสำคัญ อย่างไรก็ตาม ด้านค่า Accuracy พบว่ามีคู่แบบจำลองที่แตกต่างอย่างมีนัยสำคัญ โดยเฉพาะ Random Forest มีค่า Accuracy ต่ำกว่า XGBoost, LightGBM และ CatBoost อย่างมีนัยสำคัญทางสถิติ

5.1.3.2 กลุ่มแบบจำลองการเรียนรู้เชิงลึก

กลุ่มแบบจำลองการเรียนรู้เชิงลึก (MLP, TabNet) ไม่มีความแตกต่างอย่างมีนัยสำคัญทางสถิติระหว่าง MLP และ TabNet ทั้งด้าน AUC และ Accuracy

5.1.3.3 กลุ่มแบบจำลองทรานส์ฟอร์มเมอร์

กลุ่มแบบจำลองทรานส์ฟอร์มเมอร์ (FT-Transformer, SAINT, TabTransformer): FT-Transformer และ SAINT ไม่มีความแตกต่างอย่างมีนัยสำคัญทางสถิติ แต่ TabTransformer มีประสิทธิภาพต่ำกว่า FT-Transformer และ SAINT อย่างมีนัยสำคัญทางสถิติทั้งด้าน AUC และ Accuracy ($p < 0.05$)

ผลการทดสอบทางสถิติเหล่านี้ยืนยันว่าแบบจำลองกลุ่ม Gradient Boosting (XGBoost, LightGBM, CatBoost) มีประสิทธิภาพที่เทียบเท่ากันและเหนือกว่าแบบจำลองอื่นอย่างมีนัยสำคัญ

5.1.4 ตัวแปรที่มีอิทธิพลต่อการทำนาย

การวิเคราะห์ความสำคัญของตัวแปรด้วยเทคนิค SHAP (SHapley Additive exPlanations) และ LIME (Local Interpretable Model-agnostic Explanations) ร่วมกับการวิเคราะห์สัมประสิทธิ์สหสัมพันธ์ (Correlation) ได้ระบุตัวแปรที่มีอิทธิพลต่อการทำนายภาวะความดันโลหิตสูงตัวแปรที่มีอิทธิพลสูงสุด 5 อันดับแรก ที่ปรากฏอย่างสม่ำเสมอในทั้งสามวิธี ได้แก่

1. ระดับครีเอตินิน (Creatinine Level) - ตัวแปรที่มีอิทธิพลสูงสุด ระดับครีเอตินินสูงบ่งชี้ถึงการทำงานของไตที่ลดลง ซึ่งเป็นปัจจัยเสี่ยงที่สำคัญสำหรับความดันโลหิตสูง
2. ระดับ LDL Cholesterol - ไขมันชนิด LDL ที่สูงมีความสัมพันธ์กับความดันโลหิตสูงผ่านกลไกของการอักเสบและการแข็งตัวของหลอดเลือด

3.คอเลสเตอรอลรวม (Total Cholesterol) - ระดับคอเลสเตอรอลรวมที่สูงเป็นปัจจัยเสี่ยงอิสระสำหรับความดันโลหิตสูง

4. อายุ (Age) - ความดันโลหิตมีแนวโน้มเพิ่มขึ้นตามอายุ เป็นปัจจัยเสี่ยงที่ไม่สามารถแก้ไขได้

5. พฤติกรรมการบริโภคอาหารเค็ม (High-Salt Dietary Behavior) – การบริโภคเกลือสูงมีความสัมพันธ์โดยตรงกับการเพิ่มขึ้นของความดันโลหิต

5.1.5 ความสอดคล้องของค่าความน่าจะเป็น

การวิเคราะห์ความสอดคล้องของค่าความน่าจะเป็น (Probability Calibration) มีความสำคัญต่อการนำแบบจำลองไปใช้ในบริบททางคลินิก เนื่องจากค่าความน่าจะเป็นที่แบบจำลองทำนายควรสะท้อนความเสี่ยงที่เกิดขึ้นจริง ผลการวิเคราะห์ Brier Score และ Mean Calibration Gap พบว่า LightGBM มีค่า Brier Score ต่ำที่สุด (0.0825) ซึ่งบ่งชี้ว่ามีความแม่นยำในการประมาณค่าความน่าจะเป็นสูงที่สุด XGBoost มีค่า Brier Score เท่ากับ 0.0826 (เกือบเท่ากับ LightGBM) และมีค่า Mean Calibration Gap ต่ำที่สุดในบรรดาแบบจำลองทั้งหมด (0.0089) ซึ่งบ่งชี้ว่าค่าความน่าจะเป็นที่ XGBoost ทำนายสอดคล้องกับความจริงที่เกิดขึ้นจริงมากที่สุด Random Forest มีค่า Brier Score สูงที่สุด (0.1135) และค่า Mean Calibration Gap สูงที่สุด (0.1371) ซึ่งบ่งชี้ว่าค่าความน่าจะเป็นที่ Random Forest ทำนายมีความคลาดเคลื่อนจากความจริงที่เกิดขึ้นจริงมากกว่าแบบจำลองอื่นอย่างมีนัยสำคัญในทางปฏิบัติ

5.2 อภิปรายผล

ผลการวิจัยแสดงให้เห็นว่าแบบจำลองการเรียนรู้ของเครื่องสามารถนำมาประยุกต์ใช้ในการจำแนกภาวะความดันโลหิตสูงจากข้อมูลทางคลินิกได้อย่างมีประสิทธิภาพ โดยเฉพาะแบบจำลองในกลุ่ม Gradient Boosting ได้แก่ XGBoost, LightGBM และ CatBoost ซึ่งให้ผลการทำนายดีที่สุดเมื่อเปรียบเทียบกับแบบจำลองอื่น ทั้งในด้านค่า AUC ที่สูงกว่า 0.92 ความรวดเร็วในการฝึกสอน ความเสถียรของผลการทำนาย และความสามารถในการจัดการกับข้อมูลที่มีความไม่สมดุลของคลาสได้ดี ความได้เปรียบดังกล่าวน่าจะเกิดจากลักษณะของข้อมูลที่เป็นข้อมูลตาราง ซึ่งประกอบด้วยทั้งตัวแปรเชิงต่อเนื่องและตัวแปรเชิงกลุ่ม ทำให้แบบจำลองประเภทนี้สามารถเรียนรู้ความสัมพันธ์ของข้อมูลได้อย่างมีประสิทธิภาพ

สำหรับแบบจำลองโครงข่ายประสาทเทียมที่ออกแบบมาสำหรับข้อมูลตาราง เช่น FT-Transformer และ SAINT พบว่าสามารถให้ผลการทำนายใกล้เคียงกับกลุ่ม Gradient Boosting แต่ต้องใช้เวลาในการฝึกสอนมากกว่า จึงอาจยังไม่เหมาะสมสำหรับการนำไปใช้งานจริงในสถานการณ์ที่มีข้อจำกัดด้านทรัพยากรการคำนวณ แม้ว่าจะมีศักยภาพในการพัฒนาต่อไปก็ตาม

ในทางกลับกัน TabTransformer ให้ผลการทำนายต่ำกว่าแบบจำลองอื่นอย่างชัดเจน ซึ่งอาจเกิดจากข้อจำกัดด้านความเหมาะสมของสถาปัตยกรรมกับชุดข้อมูลที่ใช้ หรือการปรับแต่งไฮเปอร์พารามิเตอร์ที่ยังไม่เหมาะสม จึงควรมีการศึกษาเพิ่มเติมเพื่อปรับปรุงประสิทธิภาพของแบบจำลองดังกล่าว

การวิเคราะห์ความสำคัญของตัวแปรด้วยเทคนิค SHAP และ LIME ช่วยอธิบายการทำงานของแบบจำลองได้อย่างชัดเจน โดยตัวแปรสำคัญที่พบ ได้แก่ ระดับครีเอตินิน ระดับ LDL Cholesterol ระดับ

คอเลสเตอรอลรวม อายุ และการบริโภคอาหารเค็ม ซึ่งสอดคล้องกับปัจจัยเสี่ยงของภาวะความดันโลหิตสูงที่ได้รับการยอมรับในทางการแพทย์ นอกจากนี้ ผลการวิเคราะห์จาก SHAP, LIME และการวิเคราะห์ความสัมพันธ์ยังให้ผลไปในทิศทางเดียวกัน แสดงให้เห็นว่าการตีความผลของแบบจำลองมีความน่าเชื่อถือและไม่ขึ้นอยู่กับวิธีการวิเคราะห์เพียงวิธีเดียว

นอกจากนี้ การประเมินการปรับเทียบค่าความน่าจะเป็น (Probability Calibration) ยังแสดงให้เห็นว่า XGBoost และ LightGBM ไม่เพียงให้ความแม่นยำในการจำแนกสูง แต่ยังสามารถประมาณค่าความน่าจะเป็นได้ใกล้เคียงกับความเป็นจริง ซึ่งเป็นคุณสมบัติที่สำคัญสำหรับการนำไปใช้สนับสนุนการตัดสินใจทางคลินิก ขณะที่ Random Forest มีค่า Brier Score และ Mean Calibration Gap สูงกว่าแบบจำลองอื่น แสดงว่าค่าความน่าจะเป็นที่ทำนายยังมีความคลาดเคลื่อน หากต้องการนำไปใช้งานจริงควรมีการปรับเทียบค่าความน่าจะเป็นเพิ่มเติมก่อนใช้งานในบริบททางการแพทย์

5.3 ข้อจำกัดของงานวิจัย

ประการแรก ชุดข้อมูลที่ใช้ในการวิจัยเป็นข้อมูล African Hypertension and Cardiovascular Disease Dataset ซึ่งพัฒนาจากข้อมูลของประชากรในทวีปแอฟริกา ทำให้ลักษณะทางพันธุกรรม พฤติกรรมการดำเนินชีวิต และปัจจัยด้านสิ่งแวดล้อมอาจแตกต่างจากประชากรในประเทศหรือภูมิภาคอื่น ดังนั้น ผลการวิจัยอาจไม่สามารถอ้างอิงหรือประยุกต์ใช้กับประชากรทุกกลุ่มได้โดยตรง และควรมีการทดสอบกับชุดข้อมูลจากประชากรอื่นเพิ่มเติม

ประการที่สอง ชุดข้อมูลประกอบด้วยตัวแปรทำนายจำนวน 17 ตัวแปร ซึ่งอาจยังไม่ครอบคลุมปัจจัยเสี่ยงทั้งหมดที่เกี่ยวข้องกับภาวะความดันโลหิตสูง เช่น ความเครียด คุณภาพการนอน พฤติกรรมการรับประทานอาหาร หรือข้อมูลการรักษาทางคลินิกบางส่วน ส่งผลให้แบบจำลองอาจยังไม่สามารถสะท้อนปัจจัยที่มีอิทธิพลต่อการเกิดโรคได้อย่างครบถ้วน

ประการที่สาม ผลการประเมินทั้งหมดดำเนินการภายใต้สภาพแวดล้อมการวิจัย ซึ่งอาจแตกต่างจากการใช้งานจริงในสถานพยาบาล เนื่องจากข้อมูลทางคลินิกจริงอาจมีความคลาดเคลื่อน ความไม่สมบูรณ์ของข้อมูล และความแตกต่างของกระบวนการเก็บข้อมูลในแต่ละหน่วยงาน จึงควรมีการทดสอบแบบจำลองกับข้อมูลจริงก่อนนำไปประยุกต์ใช้

ประการที่สี่ การปรับแต่งไฮเปอร์พารามิเตอร์ดำเนินการด้วยเทคนิค Optuna จำนวน 30 รอบการทดลองต่อแบบจำลอง ซึ่งอาจยังไม่เพียงพอสำหรับแบบจำลองที่มีโครงสร้างซับซ้อน โดยเฉพาะแบบจำลองในกลุ่ม Transformer ที่มีจำนวนพารามิเตอร์มาก ส่งผลให้ประสิทธิภาพของแบบจำลองบางประเภทอาจยังไม่อยู่ในระดับที่เหมาะสมที่สุด

โดยรวม แม้ว่าจะมีข้อจำกัดดังกล่าว แต่งานวิจัยได้ดำเนินการตามแนวทางการพัฒนาแบบจำลองการเรียนรู้ของเครื่องอย่างเป็นระบบ มีการป้องกันการรั่วไหลของข้อมูล (Data Leakage) ใช้การตรวจสอบไขว้แบบ Stratified Cross Validation และประเมินผลด้วยตัวชี้วัดหลายด้าน จึงทำให้ผลการวิจัยมีความน่าเชื่อถือและสามารถใช้เป็นพื้นฐานสำหรับการศึกษาลึกซึ้งในอนาคตได้

5.4 ข้อเสนอแนะ

5.4.1 ข้อเสนอแนะในการนำผลการวิจัยไปใช้

จากผลการวิจัย พบว่าแบบจำลอง XGBoost และ LightGBM มีประสิทธิภาพสูงทั้งในด้านความสามารถในการจำแนกภาวะความดันโลหิตสูง ความเสถียรของผลการทำนาย และการปรับเทียบค่าความน่าจะเป็น จึงเหมาะสำหรับการนำไปประยุกต์ใช้เป็นตัวแบบของระบบคัดกรองหรือระบบสนับสนุนการตัดสินใจทางคลินิก อย่างไรก็ตาม ก่อนนำไปใช้งานจริงควรมีการทดสอบกับข้อมูลจากสถานพยาบาลหรือประชากรเป้าหมาย เพื่อประเมินความสามารถในการใช้งานในสภาพแวดล้อมจริง นอกจากนี้ ผลการวิเคราะห์ด้วยเทคนิค SHAP และ LIME สามารถนำมาใช้ประกอบการอธิบายผลการทำนายของแบบจำลอง ทำให้ผู้ใช้งานสามารถเข้าใจปัจจัยที่มีผลต่อการตัดสินใจของแบบจำลองได้มากขึ้น ซึ่งช่วยเพิ่มความโปร่งใสและความน่าเชื่อถือของระบบ โดยเฉพาะในงานด้านการแพทย์ที่การอธิบายผลมีความสำคัญต่อการตัดสินใจของบุคลากรทางการแพทย์ สำหรับแบบจำลองที่มีค่าความน่าจะเป็นคลาดเคลื่อนจากความเป็นจริง เช่น Random Forest ควรมีการปรับเทียบค่าความน่าจะเป็น (Probability Calibration) เพิ่มเติมก่อนนำไปใช้งาน เพื่อให้ค่าความน่าจะเป็นที่ได้สามารถสะท้อนโอกาสการเกิดโรคได้อย่างเหมาะสม

5.4.2 ข้อเสนอแนะสำหรับการวิจัยครั้งต่อไป

การวิจัยในอนาคตควรขยายการศึกษาด้วยการใช้ชุดข้อมูลจากประชากรที่หลากหลายหรือจากสถานพยาบาลหลายแห่ง เพื่อประเมินความสามารถในการประยุกต์ใช้ของแบบจำลองกับข้อมูลจริง รวมถึงดำเนินการตรวจสอบความถูกต้องภายนอก (External Validation) เพื่อยืนยันความสามารถในการใช้งานในบริบทที่แตกต่างกัน

นอกจากนี้ ควรศึกษาการเพิ่มตัวแปรที่อาจมีความสัมพันธ์กับภาวะความดันโลหิตสูง เช่น พฤติกรรมการนอน ระดับความเครียด การออกกำลังกาย พฤติกรรมการรับประทานอาหาร และข้อมูลทางคลินิกอื่น ๆ เพื่อเพิ่มประสิทธิภาพของแบบจำลองและสะท้อนปัจจัยเสี่ยงได้ครอบคลุมมากยิ่งขึ้น สำหรับแบบจำลองในกลุ่มการเรียนรู้เชิงลึกโดยเฉพาะทรานส์ฟอร์เมอร์ควรมีการศึกษาการปรับแต่งสถาปัตยกรรมและไฮเปอร์พารามิเตอร์เพิ่มเติม รวมถึงการทดลองกับชุดข้อมูลที่มีขนาดใหญ่ขึ้น เพื่อประเมินศักยภาพของแบบจำลองในงานจำแนกข้อมูลทางคลินิก

สุดท้าย ควรมีการศึกษาการประยุกต์ใช้เทคนิคปัญญาประดิษฐ์ที่อธิบายได้ในสภาพแวดล้อมจริง โดยประเมินความเข้าใจและการยอมรับของบุคลากรทางการแพทย์ต่อผลการอธิบายของแบบจำลอง เพื่อสนับสนุนการนำปัญญาประดิษฐ์ไปใช้ในการตัดสินใจทางคลินิกอย่างมีประสิทธิภาพ

บทสรุป

งานวิจัยนี้มีวัตถุประสงค์เพื่อประเมินและเปรียบเทียบประสิทธิภาพของแบบจำลองการเรียนรู้ของเครื่องและการเรียนรู้เชิงลึกจำนวน 9 แบบจำลอง สำหรับการจำแนกภาวะความดันโลหิตสูงจากข้อมูลทางคลินิก โดยดำเนินการพัฒนาแบบจำลองภายใต้กระบวนการที่ลดความเสี่ยงของการรั่วไหลของข้อมูล ใช้การตรวจสอบไขว้แบบ Stratified Cross Validation และประเมินผลด้วยตัวชี้วัดหลายด้าน ได้แก่ Accuracy,

Precision, Recall, F1-score, ROC-AUC รวมถึงการวิเคราะห์ Explainable Artificial Intelligence (XAI) และการประเมิน Probability Calibration ผลการวิจัยพบว่า XGBoost และ LightGBM เป็นแบบจำลองที่ให้ประสิทธิภาพดีที่สุด โดยสามารถจำแนกภาวะความดันโลหิตสูงได้อย่างแม่นยำ มีความเสถียรในการทำนาย และให้ค่าความน่าจะเป็นที่สอดคล้องกับความเป็นจริง ขณะที่การวิเคราะห์ด้วย SHAP และ LIME แสดงให้เห็นว่าปัจจัยสำคัญที่มีผลต่อการทำนาย ได้แก่ ระดับครีเอตินิน ระดับ LDL Cholesterol ระดับคอเลสเตอรอลรวม อายุ และการบริโภคอาหารเค็ม ซึ่งสอดคล้องกับองค์ความรู้ทางการแพทย์และช่วยเพิ่มความน่าเชื่อถือของแบบจำลอง แม้ว่าการวิจัยนี้จะมีข้อจำกัดด้านชุดข้อมูลและการประยุกต์ใช้กับประชากรกลุ่มอื่น แต่ผลการศึกษาส่งเสริมให้เห็นว่าเทคนิคการเรียนรู้ของเครื่อง โดยเฉพาะแบบจำลองในกลุ่ม Gradient Boosting มีศักยภาพในการสนับสนุนการคัดกรองภาวะความดันโลหิตสูงจากข้อมูลทางคลินิกได้อย่างมีประสิทธิภาพ ทั้งยังสามารถใช้เป็นแนวทางสำหรับการพัฒนาระบบสนับสนุนการตัดสินใจทางการแพทย์และการศึกษาวิจัยที่เกี่ยวข้องในอนาคตได้ โดยควรมีการตรวจสอบความถูกต้องกับข้อมูลจากสถานพยาบาลจริง ก่อนนำไปใช้งานในทางปฏิบัติ

เอกสารอ้างอิง

- [1] เริงฤดี ปธานวนิช, วิชัย เอกพลากร, and ก. สำนักงานสถิติแห่งชาติ, *รายงานการสำรวจสุขภาพประชาชนไทยโดยการตรวจร่างกาย ครั้งที่ 7 พ.ศ. 2567–2568*. กรุงเทพมหานคร: คณะแพทยศาสตร์โรงพยาบาลรามาธิบดี มหาวิทยาลัยมหิดล, 2568.
- [2] World Health Organization, *Global report on hypertension 2025: high stakes – turning evidence into action*. Geneva. World Health Organization, 2025.
- [3] R. Fujii, “En route to conquer the silent killer ‘hypertension’: Integration of polygenic risk score with non-genetic determinants,” *Hypertension Research*, vol. 47, no. 11, pp. 3079–3081, 2024, doi: 10.1038/s41440-024-01826-0.
- [4] D. A. Murtiningsih, B. W. Sari, and I. N. Fajri, “Comparison of Light Gradient Boosting Machine, eXtreme Gradient Boosting, and CatBoost with Balancing and Hyperparameter Tuning for Hypertension Risk Prediction on Clinical Dataset,” *Journal of Applied Informatics and Computing*, vol. 9, no. 5, pp. 2753–2763, Oct. 2025, doi: 10.30871/jaic.v9i5.10400.
- [5] S. K. Upadhyayula and R. Pothugunta, “Benchmarking Transformer-Based and Conventional Machine Learning Models for Cardiovascular Disease Prediction on Datasets of Varying Scale and Complexity,” *medRxiv*, p. 2025.08.03.25332878, Jan. 2025, doi: 10.1101/2025.08.03.25332878.
- [6] GeeksforGeeks, “History of Python.” Accessed: Dec. 26, 2025. [Online]. Available: <https://www.geeksforgeeks.org/python/history-of-python/>
- [7] GeeksforGeeks, “Introduction to Weka: Key Features and Applications.” Accessed: Dec. 26, 2025. [Online]. Available: <https://www.geeksforgeeks.org/machine-learning/introduction-to-weka-key-features-and-applications/>
- [8] Microsoft Corporation, “Visual Studio Code documentation.” Accessed: Dec. 26, 2025. [Online]. Available: <https://code.visualstudio.com/docs>
- [9] Google LLC, “Google Colaboratory.” Accessed: Dec. 26, 2025. [Online]. Available: <https://colab.google/>
- [10] Microsoft, “Microsoft Excel.” Accessed: Dec. 26, 2025. [Online]. Available: <https://www.microsoft.com/th-th/microsoft-365/excel#footnote-4>
- [11] I. pandas via NumFOCUS, “About pandas.” Accessed: Mar. 20, 2026. [Online]. Available: <https://pandas.pydata.org/about/>

- [12] NumPy team, “About Us.” Accessed: Mar. 20, 2026. [Online]. Available: <https://numpy.org/about/>
- [13] scikit-learn developers, “scikit-learn Machine Learning in Python.” Accessed: Mar. 20, 2026. [Online]. Available: <https://scikit-learn.org/stable/>
- [14] PyTorch, “PyTorch Foundation.” Accessed: Mar. 20, 2026. [Online]. Available: https://pytorch.org/foundation/?_gl=1*11mzaqr*_up*MQ..*_ga*MjAzMzcyMTEwMi4xNzc0MDI1MTc0*_ga_469Y0W5V62*czE3NzQwMjUxNzMkbzEkZzEkdDE3NzQwMjUyMjkkajQkbDAkaDA
- [15] XGBoost, “XGBoost Documentation.” Accessed: Mar. 20, 2026. [Online]. Available: https://xgboost.readthedocs.io/en/release_3.2.0/
- [16] TensorFlow, “An end-to-end platform for machine learning.” Accessed: Mar. 20, 2026. [Online]. Available: <https://www.tensorflow.org/>
- [17] Optuna, “Optimize Your Optimization.” Accessed: Mar. 20, 2026. [Online]. Available: <https://optuna.org/>
- [18] SHAP, “Welcome to the SHAP documentation.” Accessed: Mar. 20, 2026. [Online]. Available: <https://shap.readthedocs.io/en/latest/>
- [19] geeksforgeeks, “Explainable AI(XAI) Using LIME.” Accessed: Mar. 20, 2026. [Online]. Available: <https://www.geeksforgeeks.org/artificial-intelligence/introduction-to-explainable-ai-xai-using-lime/>
- [20] Matplotlib, “Matplotlib: Visualization with Python.” Accessed: Mar. 20, 2026. [Online]. Available: <https://matplotlib.org/>
- [21] P. K. Whelton *et al.*, “2017 ACC/AHA/AAPA/ABC/ACPM/AGS/APhA/ASH/ASPC/NMA/PCNA Guideline for the Prevention, Detection, Evaluation, and Management of High Blood Pressure in Adults: Executive Summary: A Report of the American College of Cardiology/American Heart Association Task Force on Clinical Practice Guidelines,” *Hypertension*, vol. 71, no. 6, pp. 1269–1324, Jun. 2018, doi: 10.1161/HYP.0000000000000066.
- [22] S. Kowalski, K. Goniewicz, A. Moskal, A. M. Al-Wathinani, and M. Goniewicz, “Symptoms in Hypertensive Patients Presented to the Emergency Medical Service: A Comprehensive Retrospective Analysis in Clinical Settings,” *J. Clin. Med.*, vol. 12, no. 17, p. 5495, Aug. 2023, doi: 10.3390/jcm12175495.

- [23] H. Ma *et al.*, “Clinical value of multiorgan damage in hypertensive crises: A prospective follow-up study,” *The Journal of Clinical Hypertension*, vol. 22, no. 5, pp. 914–923, May 2020, doi: 10.1111/jch.13848.
- [24] M. I. Jordan and T. M. Mitchell, “Machine learning: Trends, perspectives, and prospects,” *Science (1979)*, vol. 349, no. 6245, pp. 255–260, Jul. 2015, doi: 10.1126/science.aaa8415.
- [25] R. Miotto, F. Wang, S. Wang, X. Jiang, and J. T. Dudley, “Deep learning for healthcare: review, opportunities and challenges,” *Brief. Bioinform.*, vol. 19, no. 6, pp. 1236–1246, Nov. 2018, doi: 10.1093/bib/bbx044.
- [26] I. Ivan Belcic, “What is classification in machine learning?” Accessed: Dec. 29, 2025. [Online]. Available: <https://www.ibm.com/think/topics/classification-machine-learning>
- [27] L. Breiman, “Random Forests,” *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, Oct. 2001, doi: 10.1023/A:1010933404324.
- [28] G. Ke *et al.*, “LightGBM: A Highly Efficient Gradient Boosting Decision Tree,” Jul. 2017.
- [29] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, “CatBoost: unbiased boosting with categorical features,” in *Advances in Neural Information Processing Systems*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., Curran Associates, Inc., 2018. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2018/file/14491b756b3a51daac41c24863285549-Paper.pdf
- [30] I. Goodfellow, Y. Bengio, and A. Courville, “Deep Learning,” MIT Press. Accessed: Dec. 16, 2025. [Online]. Available: <http://www.deeplearningbook.org>
- [31] H. Habebh and S. Gohel, “Machine Learning in Healthcare,” *Curr. Genomics*, vol. 22, no. 4, pp. 291–300, Dec. 2021, doi: 10.2174/1389202922666210705124359.
- [32] L. Grinsztajn, E. Oyallon, and G. Varoquaux, “Why do tree-based models still outperform deep learning on typical tabular data?,” in *Proceedings of the 36th International Conference on Neural Information Processing Systems*, in NIPS ’22. Red Hook, NY, USA: Curran Associates Inc., 2022.
- [33] S. Ö. Arik and T. Pfister, “TabNet: Attentive Interpretable Tabular Learning,” *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 8, pp. 6679–6687, May 2021, doi: 10.1609/aaai.v35i8.16826.
- [34] R. Pérez-Jove, C. R. Munteanu, A. Pazos, and J. Vázquez-Naya, “Application of Tabular Transformer Architectures for Operating System Fingerprinting,” Feb. 2025.

- [35] Y. Gorishniy, I. Rubachev, V. Khruikov, and A. Babenko, “Revisiting Deep Learning Models for Tabular Data,” *CoRR*, vol. abs/2106.11959, 2021, [Online]. Available: <https://arxiv.org/abs/2106.11959>
- [36] G. Somepalli, M. Goldblum, A. Schwarzschild, C. Bruss, and T. Goldstein, “SAINT: Improved Neural Networks for Tabular Data via Row Attention and Contrastive Pre-Training,” Jul. 2021. doi: 10.48550/arXiv.2106.01342.
- [37] X. Huang, A. Khetan, M. Cvitkovic, and Z. Karnin, “TabTransformer: Tabular Data Modeling Using Contextual Embeddings,” Dec. 2020.
- [38] L. Grinsztajn, E. Oyallon, and G. Varoquaux, “Why do tree-based models still outperform deep learning on typical tabular data?,” in *Proceedings of the 36th International Conference on Neural Information Processing Systems*, in NIPS ’22. Red Hook, NY, USA: Curran Associates Inc., 2022.
- [39] C. M. . Bishop, *Pattern recognition and machine learning*. Springer Science + Business Media, 2009.
- [40] S. Lundberg and S.-I. Lee, “A Unified Approach to Interpreting Model Predictions,” Nov. 2017.
- [41] M. T. Ribeiro, S. Singh, and C. Guestrin, “‘Why Should I Trust You?’: Explaining the Predictions of Any Classifier,” in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, in KDD ’16. New York, NY, USA: Association for Computing Machinery, 2016, pp. 1135–1144. doi: 10.1145/2939672.2939778.
- [42] X. Xu *et al.*, “Explainable AI as a Double-Edged Sword in Dermatology: The Impact on Clinicians versus The Public,” 2025. [Online]. Available: <https://arxiv.org/abs/2512.12500>
- [43] A. Andishgar *et al.*, “Machine learning-based models to predict the conversion of normal blood pressure to hypertension within 5-year follow-up,” *PLoS One*, vol. 19, no. 3, p. e0300201, Mar. 2024, doi: 10.1371/journal.pone.0300201.
- [44] B. Asal, “Transformer-Based Deep Learning Framework for Interpretable Cardiac Arrest Detection Using Tabular Clinical Data,” in *2025 10th International Conference on Computer Science and Engineering (UBMK)*, 2025, pp. 847–852. doi: 10.1109/UBMK67458.2025.11206850.
- [45] F. Si, Q. Liu, and J. Yu, “A prediction study on the occurrence risk of heart disease in older hypertensive patients based on machine learning,” *BMC Geriatr.*, vol. 25, no. 1, p. 27, 2025, doi: 10.1186/s12877-025-05679-1.

ภาคผนวก

การประเมินประสิทธิภาพของเทคนิคการเรียนรู้ของเครื่องสำหรับการจำแนกโรคความดันโลหิตสูงจากข้อมูลทางคลินิก

Performance Evaluation of Machine Learning Techniques for Hypertension Classification Using Clinical Data

ธีระชาติ พูลพัฒน์¹, อาทิตยาพร โรจรัตน์^{2*}

Teerachad Phoonphut¹, Artitayaporn Rojarath^{2*}

บทคัดย่อ

การวิจัยนี้มีวัตถุประสงค์เพื่อประเมินและเปรียบเทียบประสิทธิภาพของเทคนิคการเรียนรู้ของเครื่อง (Machine Learning) และการเรียนรู้เชิงลึก (Deep Learning) สำหรับการจำแนกภาวะความดันโลหิตสูงจากข้อมูลทางคลินิก โดยศึกษาความสามารถของแบบจำลอง และวิเคราะห์ปัจจัยเสี่ยงสำคัญผ่านเทคนิค Explainable Artificial Intelligence (XAI)

ใช้ชุดข้อมูล African Hypertension and Cardiovascular Disease Dataset จำนวน 10,000 ระเบียบ หลังจากคัดกรองตัวแปรที่มีความเสี่ยงต่อ Data Leakage ออก 6 ตัวแปร ดำเนินการเปรียบเทียบแบบจำลอง 9 แบบ ได้แก่ XGBoost, Random Forest, LightGBM, CatBoost, MLP, TabNet, FT-Transformer, SAINT และ TabTransformer โดยแบ่งข้อมูลเป็นสัดส่วน 80:20 ด้วย Stratified Sampling ปรับแต่งไฮเปอร์พารามิเตอร์ด้วย Optuna ภายใต้ Stratified 5-Fold Cross-Validation และประเมินผลด้วย Accuracy, Precision, Recall, F1-score และ AUC-ROC พร้อมวิเคราะห์ความแตกต่างทางสถิติด้วย Paired t-test, Friedman Test และ Nemenyi Post-hoc Test

XGBoost ให้ผลการจำแนกที่ดีที่สุดบนชุดทดสอบ โดยมีค่า AUC เท่ากับ 0.9209 ตามด้วย LightGBM (0.9201) และ CatBoost (0.9189) แบบจำลองส่วนใหญ่มี AUC สูงกว่า 0.90 แสดงศักยภาพในการทำนายสูง อย่างไรก็ตาม TabTransformer มีประสิทธิภาพต่ำกว่าแบบจำลองอื่นอย่างมีนัยสำคัญ ($p < 0.05$) ผลการวิเคราะห์ XAI ด้วย SHAP และ LIME ชี้ให้เห็นว่า ระดับครีเอตินีน, LDL Cholesterol, คอเลสเตอรอลรวม, อายุ และพฤติกรรมการบริโภคอาหารเค็ม เป็นปัจจัยเสี่ยงสำคัญที่สุด ขณะที่การผสมแบบจำลองด้วย Stacking Ensemble ไม่สามารถปรับปรุงผลลัพธ์ได้เหนือกว่า XGBoost อย่างมีนัยสำคัญ

คำสำคัญ: การเรียนรู้ของเครื่อง; ภาวะความดันโลหิตสูง; ข้อมูลทางคลินิก; XGBoost; ปัญญาประดิษฐ์ที่อธิบายได้

Abstract

This study aims to evaluate and compare the performance of Machine Learning and Deep Learning techniques for hypertension classification using clinical data and identifies important risk factors through Explainable Artificial Intelligence (XAI) techniques.

The African Hypertension and Cardiovascular Disease Dataset containing 10,000 records was employed. Six variables associated with potential data leakage were removed during preprocessing. Nine models, including XGBoost, Random Forest, LightGBM, CatBoost, Multi-Layer Perceptron (MLP), TabNet, FT-Transformer, SAINT, and TabTransformer, were evaluated. The dataset was divided into training and testing

sets using an 80:20 ratio with stratified sampling. Hyperparameter optimization was performed using Optuna with Stratified 5-Fold Cross-Validation. Model performance was assessed using Accuracy, Precision, Recall, F1-score, and AUC-ROC. Statistical significance was examined using the Paired t-test, Friedman Test, and Nemenyi Post-hoc Test.

The experimental results showed that XGBoost achieved the highest performance on the test dataset with an AUC of 0.9209, followed by LightGBM (0.9201) and CatBoost (0.9189). Most models achieved AUC values above 0.90, demonstrating strong predictive capability. However, TabTransformer showed significantly lower performance compared with other models ($p < 0.05$). The XAI analysis using SHAP and LIME identified creatinine level, LDL cholesterol, total cholesterol, age, and high-salt dietary behavior as the most influential risk factors.

Keywords: Machine Learning; Hypertension; Clinical Data; XGBoost; Explainable Artificial Intelligence (XAI)

¹ Faculty of Informatics, Mahasarakham University, Mahasarakham 44150, Thailand

² Multi-agent Intelligent Simulation Laboratory (MISL) Research Unit, Department of Information Technology, Faculty of Informatics, Mahasarakham University, Mahasarakham 44150, Thailand

***Corresponding author's email:** artitayaporn.r@msu.ac.th

บทนำ

โรคความดันโลหิตสูง (Hypertension) ถือเป็นปัญหาสาธารณสุขที่สำคัญทั้งในประเทศไทยและทั่วโลก เนื่องจากเป็นภาวะที่พบได้บ่อย มีลักษณะเรื้อรัง และเกี่ยวข้องโดยตรงกับภาวะโรคหัวใจและหลอดเลือดในระดับประชากร ข้อมูลจากการสำรวจสุขภาพประชาชนไทยโดยการตรวจร่างกาย ครั้งที่ 7 พ.ศ. 2567–2568 ซึ่งเก็บข้อมูลภาคสนามระหว่างเดือนสิงหาคม 2567 ถึงเมษายน 2568 พบว่าประชาชนไทยอายุ 15 ปีขึ้นไปมีความชุกของโรคความดันโลหิตสูงร้อยละ 29.5 เพิ่มขึ้นจากร้อยละ 25.4 ในการสำรวจปี 2563 สะท้อนว่าประมาณ 3 ใน 10 คนกำลังเผชิญภาวะดังกล่าว (เริงฤดี ปธานวนิช และคณะ, 2568) สถานการณ์นี้สอดคล้องกับแนวโน้มระดับโลก โดยองค์การอนามัยโลกรายงานว่าจำนวนผู้ใหญ่ที่มีภาวะความดันโลหิตสูงทั่วโลกเพิ่มขึ้นจาก 650 ล้านคนในปี 1990 เป็นประมาณ 1.4 พันล้านคนในปี 2024 ผู้ป่วยจำนวนมากไม่ทราบว่าตนเองมีโรค เนื่องจากภาวะนี้มักไม่แสดงอาการเด่นชัดในระยะแรก การตรวจวัดความดันโลหิตจึงเป็นวิธีสำคัญในการค้นหาผู้ป่วยก่อนเกิดภาวะแทรกซ้อนรุนแรง อีกทั้งยังเป็นปัจจัยเสี่ยงสำคัญของโรคหัวใจและหลอดเลือด ซึ่งองค์การอนามัยโลกระบุว่าเป็นสาเหตุการเสียชีวิตอันดับหนึ่งของโลก โดยมีผู้เสียชีวิตประมาณ 19.8 ล้านคนในปี 2022 (World Health Organization, 2025) ด้วยเหตุนี้ โรคความดันโลหิตสูงจึงมักถูกกล่าวถึงในเชิงสาธารณสุขว่าเป็น "เพชฌฆาตเงียบ" เนื่องจากผู้ป่วยมักไม่รู้ความผิดปกติเมื่อโรคลุกลามไปสู่ภาวะหลอดเลือดสมอง โรคหัวใจ หรือไตเสื่อมแล้ว (Fujii, 2024)

ดังนั้น การพัฒนาเครื่องมือคัดกรองและทำนายเชิงรุกเพื่อสนับสนุนการวินิจฉัยและบริหารจัดการโรคความดันโลหิตสูงจึงมีความสำคัญอย่างยิ่ง งานวิจัยที่ผ่านมาแสดงให้เห็นว่าเทคนิคการเรียนรู้ของเครื่องมีศักยภาพในการทำนายหรือจำแนกภาวะความดันโลหิตสูงได้อย่างมีประสิทธิภาพ โมเดลเหล่านี้แบ่งออกเป็นสองกลุ่มหลัก ได้แก่ กลุ่มต้นไม้ตัดสินใจและการเรียนรู้แบบรวมกลุ่ม เช่น XGBoost, Random Forest, LightGBM และ CatBoost ซึ่งได้รับความนิยมเนื่องจากมีความแม่นยำสูงและสามารถจัดการกับข้อมูลไม่สมดุลได้ดี ตัวอย่างเช่น การศึกษาที่เปรียบเทียบโมเดลเหล่านี้บนชุดข้อมูลคลินิกโรคความดันโลหิตสูง พบว่า CatBoost ให้ผลลัพธ์ที่ดีที่สุด (Murtiningsih et al., 2025) อีกกลุ่มคือการเรียนรู้เชิงลึกและแบบจำลองที่อิงกลไกความเข้าใจเช่น Multi-Layer Perceptron (MLP), TabNet ที่ใช้กลไกความเข้าใจแบบลำดับ รวมถึงสถาปัตยกรรมทรานส์ฟอร์เมอร์สำหรับข้อมูลตาราง ได้แก่ FT-Transformer, SAINT และ TabTransformer ที่ใช้กลไกความเข้าใจในตนเองในการเรียนรู้ปฏิสัมพันธ์ระหว่างตัวแปร งานวิจัยล่าสุดพบว่าโมเดลกลุ่มนี้ให้ผลดีบนชุดข้อมูลขนาดใหญ่และมีโครงสร้างชัดเจน แต่ประสิทธิภาพจะลดลงเมื่อข้อมูลมีความซับซ้อนและไม่สมดุล ในขณะที่โมเดลดั้งเดิมอย่าง XGBoost ยังคงสามารถแข่งขันได้ในหลายกรณี สะท้อนว่าการเลือกโมเดลควรพิจารณาจากลักษณะข้อมูลเป็นสำคัญ และไม่มีโมเดลใดที่เหนือกว่าอย่างสมบูรณ์ (Upadhyayula & Pothugunta, 2025b)

อย่างไรก็ตาม เมื่อพิจารณาเชิงระเบียบวิธีอย่างละเอียด ยังพบข้อจำกัดที่สำคัญในองค์ความรู้ปัจจุบันหลายประการ ประการแรก งานวิจัยจำนวนมากไม่น้อยยังคงใช้ตัวแปรที่มีความเสี่ยงต่อการก่อให้เกิดข้อมูลรั่วไหล เช่น ตัวแปรที่เป็นผลลัพธ์ทางคลินิก ซึ่งอาจทำให้แบบจำลองมีสมรรถนะที่ดีเกินจริงและไม่สะท้อนความสามารถในการทำนายที่แท้จริงเมื่อนำไปใช้กับข้อมูลใหม่ ประการที่สอง การประเมินสมรรถนะแบบจำลองในหลายการศึกษายังอาศัยการแบ่งชุดข้อมูลเพียงครั้งเดียว โดยไม่ได้ใช้กรอบการทดสอบแบบไขว้ ทำให้ผลลัพธ์อาจมีความคลาดเคลื่อนสูง ประการที่สาม แม้แบบจำลองกลุ่มการเรียนรู้เชิงลึกและ การเพิ่มประสิทธิภาพแบบไล่ระดับจะให้ความแม่นยำสูง แต่มักถูกมองว่าตีความยาก

การศึกษานี้จึงมีกระบวนการพัฒนาแบบจำลองให้มีความรัดกุมมากขึ้น โดยตัดตัวแปรที่มีความเสี่ยงต่อการก่อให้เกิดข้อมูลรั่วไหล ประเมินแบบจำลองฐานหลายชนิด ครอบคลุมทั้งกลุ่มต้นไม้ตัดสินใจ (XGBoost, Random Forest, LightGBM, CatBoost) โครงข่ายประสาทเทียมแบบดั้งเดิม (MLP) การเรียนรู้เชิงลึกสำหรับข้อมูลตาราง (TabNet) และกลุ่มของทรานส์ฟอร์เมอร์ (FT-Transformer, SAINT, TabTransformer) ภายใต้กรอบ cross-validation และใช้วิธีอธิบายแบบจำลอง (SHAP และ LIME) ควบคู่กัน เพื่อให้ผลการศึกษาเชื่อถือได้ โปร่งใส และตีความได้มากขึ้น

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

1. โรคความดันโลหิตสูง (Hypertension)

โรคความดันโลหิตสูง หมายถึง ภาวะที่ความดันโลหิตในหลอดเลือดแดงสูงกว่าเกณฑ์ปกติอย่างต่อเนื่อง โดยทั่วไปวินิจฉัยเมื่อค่าความดันโลหิตซิสโตลิกตั้งแต่ 140 มิลลิเมตรปรอทขึ้นไป และ/หรือค่าความดันโลหิตไดแอสโตลิกตั้งแต่ 90 มิลลิเมตรปรอทขึ้นไป ภาวะนี้มักไม่แสดงอาการเด่นชัดในระยะแรก ผู้ป่วยจำนวนมากจึงไม่ทราบว่าตนเองมีภาวะดังกล่าวจนนำไปสู่การเกิดภาวะแทรกซ้อนอื่น ๆ (Whelton et al., 2018)

อาการในระยะเริ่มต้นมักมีลักษณะการดำเนินโรคแบบไม่แสดงอาการผู้ป่วยส่วนใหญ่จึงไม่ทราบว่าตนเองมีภาวะดังกล่าวในบางรายอาจปรากฏอาการเบื้องต้นซึ่งไม่จำเพาะเจาะจงต่อโรค ได้แก่ ปวดศีรษะ โดยเฉพาะบริเวณท้ายทอย ตาพร่ามัว ใจสั่น เป็นต้น (Kowalski et al., 2023)

ในกรณีที่รุนแรงผู้ป่วยอาจเข้าสู่ภาวะวิกฤตความดันโลหิตสูงซึ่งจำแนกเป็น ความดันสูงรีบด่วนและ ความดันสูงฉุกเฉิน โดยกลุ่มความดันสูงฉุกเฉินจะมีภาวะอวัยวะเป้าหมายถูกทำลายเฉียบพลันร่วมด้วย เช่น ภาวะเลือดออกในสมอง กล้ามเนื้อหัวใจขาดเลือดเฉียบพลัน หรือไตวายเฉียบพลัน ซึ่งจัดเป็นภาวะฉุกเฉินทางการแพทย์ที่ต้องได้รับการรักษาทันที (Ma et al., 2020)

2. การเรียนรู้ของเครื่อง (Machine Learning)

การเรียนรู้ของเครื่อง คือปัญญาประดิษฐ์ที่มุ่งพัฒนาอัลกอริทึมและแบบจำลองเชิงสถิติให้สามารถเรียนรู้รูปแบบและความสัมพันธ์จากข้อมูลได้ โดยไม่จำเป็นต้องกำหนดกฎหรือเงื่อนไขในการตัดสินใจไว้ล่วงหน้าทุกขั้นตอนโดยแบ่งออกเป็น 2 ประเภทหลัก ได้แก่ การเรียนรู้แบบมีผู้สอน (Supervised Learning) ซึ่งใช้ข้อมูลที่มีฉลากคำตอบในการฝึกแบบจำลอง และการเรียนรู้แบบไม่มีผู้สอน (Unsupervised Learning) ซึ่งมุ่งค้นหาโครงสร้างหรือรูปแบบที่ซ่อนอยู่ในข้อมูลโดยไม่อาศัยฉลากคำตอบ (Jordan & Mitchell, 2015)

สำหรับงานด้านสุขภาพ การเรียนรู้ของเครื่องมีข้อดีคือสามารถเรียนรู้ความสัมพันธ์แบบไม่เป็นเชิงเส้น ระหว่างปัจจัยเสี่ยงและตัวแปรทางคลินิกที่มีความซับซ้อนได้อย่างมีประสิทธิภาพ โดยเฉพาะอย่างยิ่งในกลุ่มแบบจำลองการเรียนรู้เชิงลึกซึ่งสามารถเรียนรู้และสกัดคุณลักษณะจากข้อมูลได้โดยอัตโนมัติ ลดความจำเป็นในการออกแบบคุณลักษณะด้วยมนุษย์เมื่อเทียบกับแบบจำลองการเรียนรู้ของเครื่องแบบดั้งเดิม (Miotto et al., 2018)

งานวิจัยนี้มุ่งเน้นการเรียนรู้แบบมีผู้สอน (Supervised Learning) ในลักษณะปัญหาการจำแนกประเภทเพื่อทำนายภาวะความดันโลหิตสูงจากปัจจัยเสี่ยงทางคลินิก โดยศึกษาแบบจำลองครอบคลุมทั้งกลุ่มการเรียนรู้ร่วมกันและกลุ่มการเรียนรู้เชิงลึก

3. การเรียนรู้แบบรวมกลุ่มต้นไม้ (Tree-based Ensemble)

3.1 วิธีการ XGBoost

XGBoost นำเสนอกรอบการคำนวณที่ปรับค่าพารามิเตอร์อย่างมีประสิทธิภาพด้วยการประมาณอนุพันธ์อันดับสองโดยทำการพยากรณ์ผลลัพธ์โดยการรวมผลลัพธ์จากต้นไม้ตัดสินใจจำนวน K ต้น ซึ่งแต่ละต้นทำหน้าที่เรียนรู้ข้อผิดพลาดของต้นไม้ก่อนหน้า ผลลัพธ์สุดท้ายของตัวอย่างข้อมูลลำดับที่ i คำนวณจากผลรวมของค่าที่ได้จากต้นไม้ทุกต้น ดังแสดงในสมการที่ 1 โดยที่ F เป็นเซตของต้นไม้ตัดสินใจทั้งหมดที่สามารถสร้างได้ และแต่ละต้นไม้จะกำหนดค่าทำนายจากค่าน้ำหนักของใบที่ตัวอย่างข้อมูลนั้นตกอยู่ ซึ่งช่วยให้แบบจำลองสามารถเรียนรู้ความสัมพันธ์ที่ซับซ้อนของข้อมูลและเพิ่มประสิทธิภาพการพยากรณ์ได้ (Chen & Guestrin, 2016)

$$y^i = \sum_{k=1}^K f_k(x_i), f_k \in \mathcal{F} \quad (1)$$

3.2 วิธีการ Random Forest

ตัวแบบป่าสุ่ม (Random Forest) สร้างชุดของต้นไม้ตัดสินใจจำนวนมากจากกลุ่มตัวอย่างและคุณลักษณะที่สุ่มเลือก แล้วรวมผลลัพธ์ด้วยการลงคะแนนเสียงส่วนใหญ่ ซึ่งช่วยลดความแปรปรวนของตัวแบบเดี่ยวและลดโอกาสเกิดการเรียนรู้เกินพอดี

$$\hat{y} = \text{mode}\{h_1(x), h_2(x), \dots, h_B(x)\} \quad (2)$$

จากสมการที่ 2 โดยที่ $\hat{y}$ คือผลลัพธ์ที่แบบจำลองพยากรณ์ $h_B(x)$ คือผลการพยากรณ์จากต้นไม้ลำดับที่ B คือจำนวนต้นไม้ทั้งหมด และ $\text{mode}(\cdot)$ คือฟังก์ชันเลือกค่าที่ได้รับการลงคะแนนเสียงมากที่สุด ทั้งนี้ Random Forest มีจุดเด่นในการลดความแปรปรวนของแบบจำลอง ลดโอกาสเกิดการเรียนรู้เกินพอดี (Overfitting) และเพิ่มความแม่นยำในการจำแนกประเภท (Breiman, 2001)

3.3 วิธีการ LightGBM

LightGBM เป็นอัลกอริทึมการเรียนรู้แบบ Gradient Boosting Decision Tree (GBDT) ที่สร้างแบบจำลองโดยการรวมผลลัพธ์จากต้นไม้ตัดสินใจหลายต้นเข้าด้วยกัน ผลการพยากรณ์ของตัวอย่างข้อมูลลำดับที่ i คำนวณจากผลรวมของค่าที่ได้จากต้นไม้ทุกต้น ดังสมการโดยที่ K คือจำนวนต้นไม้ตัดสินใจ และ $f_k(x_i)$ คือผลลัพธ์จากต้นไม้ลำดับที่ k สำหรับข้อมูล x_i ทั้งนี้ LightGBM มีความแตกต่างจาก GBDT ที่ใช้การเติบโตของต้นไม้แบบ Leaf-wise Tree Growth ร่วมกับเทคนิค Gradient-based One-Side Sampling (GOSS) และ Exclusive Feature Bundling (EFB) เพื่อเพิ่มความเร็วในการฝึกแบบจำลอง ลดการใช้หน่วยความจำ และยังคงประสิทธิภาพในการพยากรณ์ที่สูง (Ke et al., 2017)

3.3 วิธีการ CatBoost

CatBoost เป็นอัลกอริทึมการเรียนรู้แบบ Gradient Boosting Decision Tree (GBDT) ที่ออกแบบมาเพื่อรองรับข้อมูลเชิงหมวดหมู่ได้อย่างมีประสิทธิภาพ โดยผลการพยากรณ์ของตัวอย่างข้อมูลลำดับที่ i คำนวณจากผลรวมของค่าที่ได้จากต้นไม้ตัดสินใจจำนวน K ต้น โดยที่ $f_k(x_i)$ คือผลลัพธ์จากต้นไม้ลำดับที่ k และ F คือเซตของต้นไม้ตัดสินใจทั้งหมด ทั้งนี้ CatBoost มีจุดเด่นคือการใช้ Ordered Boosting เพื่อลด Prediction Shift และการแปลงข้อมูลเชิงหมวดหมู่ด้วย Target Statistics (Ordered Target Encoding) ซึ่งช่วยลดการเกิดข้อมูลรั่วไหล (Target Leakage) และเพิ่มความแม่นยำของแบบจำลอง (Prokhorenkova et al., 2018)

4. กลุ่มการเรียนรู้เชิงลึก (Deep Learning)

4.1 วิธีการ โครงข่ายประสาทเทียมแบบหลายชั้น

โครงข่ายประสาทเทียมแบบหลายชั้น (Multi-Layer Perceptron) เป็นโครงข่ายประสาทเทียมแบบส่งไปข้างหน้าที่ประกอบด้วยชั้นนำเข้า ชั้นซ่อนหนึ่งชั้นขึ้นไป และชั้นส่งออก โดยแต่ละโหนดเชื่อมต่อกันด้วยน้ำหนักที่ปรับค่าผ่านกระบวนการแพร่ย้อนกลับของความผิดพลาดร่วมกับฟังก์ชันกระตุ้นแบบไม่เชิงเส้น

$$a^{(l)} = \sigma(W^{(l)}a^{(l-1)} + b^{(l)}) \quad (3)$$

โดยที่ $a^{(l)}$ คือผลลัพธ์ของชั้นที่ l , $W^{(l)}$ คือเมทริกซ์ค่าน้ำหนัก $b^{(l)}$ คือเวกเตอร์ไบแอส และ $\sigma(\cdot)$ คือฟังก์ชันกระตุ้น ทั้งนี้ MLP สามารถเรียนรู้ความสัมพันธ์แบบไม่เป็นเชิงเส้น (Nonlinear Relationship) ได้ดี จึงได้รับความนิยมในการแก้ปัญหการจำแนกประเภทและการพยากรณ์ (Grinsztajn et al., 2022a)

4.2 วิธีการ TabNet

TabNet ใช้กลไก ซีเควนเขียน แอตเทนชัน เพื่อเลือกตัวแปรที่เกี่ยวข้องที่สุดในแต่ละขั้นตอนการตัดสินใจ ทำให้มีคุณสมบัติในการตีความผลได้บางส่วนโดยธรรมชาติ โดยประยุกต์ใช้กลไกความใส่ใจร่วมกับโครงสร้างการตัดสินใจแบบต่อเนื่องจำนวน T ขั้นตอน (Sequential Decision Steps) ซึ่งเลียนแบบกระบวนการทำงานของต้นไม้ตัดสินใจ โครงสร้างการไหลของข้อมูลเพื่อคำนวณผลลัพธ์สุดท้ายอธิบายได้ด้วยสมการดังนี้ (Arik & Pfister, 2021)

$$\hat{y} = \sum_{t=1}^T f^{(t)}(X^{(t)}) \quad (4)$$

โดยกำหนดให้ $\hat{y}$ คือค่าผลลัพธ์สุดท้ายที่โมเดลทำนาย T คือจำนวนขั้นตอนการตัดสินใจทั้งหมด $f^{(t)}$ คือฟังก์ชันแปลงคุณลักษณะ ประจำขั้นตอนที่ t และ $X^{(t)}$ คือข้อมูลอินพุตที่ผ่านกลไกการเลือกคุณลักษณะมาแล้วในขั้นตอนดังกล่าวกระบวนการประมวลผลเริ่มต้นจากการนำข้อมูลดิบมาคูณแบบตำแหน่งต่อตำแหน่งกับตัวกรองคุณลักษณะหรือแมสค์ $M^{(t)}$ ที่สร้างจากโมดูล Attentive Transformer เกิดเป็นข้อมูลจำเพาะ $(X^{(t)} = M^{(t)} \odot X)$ จากนั้นข้อมูลนี้จะถูกส่งเข้าสู่ฟังก์ชัน $(f^{(t)})$ เพื่อสกัดคุณลักษณะขั้นสูง ก่อนที่ผลลัพธ์จะย่อจากทุกขั้นตอนการตัดสินใจ ตั้งแต่ขั้นตอนที่ 1 ถึง T จะถูกนำมารวมกัน (Summation) เพื่อส่งออกเป็นคำตอบสุดท้าย การส่งต่อข้อมูลในลักษณะนี้ช่วยให้โมเดลมีคุณสมบัติเด่นด้านการอธิบายได้ (Explainability) ผ่านการวิเคราะห์ค่าน้ำหนักในตัวกรอง $(M^{(t)})$ ของแต่ละขั้นตอน

5.กลุ่มทรานส์ฟอร์เมอร์ (Transformer-based model)

5.1 วิธีการ FT-Transformer

FT-Transformer แปลงตัวแปรทั้งเชิงตัวเลขและเชิงกลุ่มให้อยู่ในรูปโทเคน ก่อนประมวลผลผ่านชั้นของกลไกความใส่ใจในตัวเองของสถาปัตยกรรมทรานส์ฟอร์เมอร์ ทำให้สามารถจับปฏิสัมพันธ์ระหว่างตัวแปรได้อย่างยืดหยุ่น (Gorishniy et al., 2021)

$$\hat{y} = \text{MLP}(\text{Transformer}(Z)) \quad (5)$$

จากสมการที่ 4 FT-Transformer มีการสมการการคำนวณผลลัพธ์สุดท้ายโดยที่ MLP คือโครงข่ายประสาทเทียมหลายชั้น Transformer คือการนำเข้าโมเดลทรานส์ฟอร์เมอร์ Z คือการนำเข้าของทรานส์ฟอร์เมอร์ และ $\hat{y}$ คือค่าทำนายสุดท้าย

5.2 วิธีการ SAINT

SAINT ขยายแนวคิดของ FT-Transformer โดยเพิ่มกลไกความใส่ใจทั้งในแนวนอนและแนวคอลัมน์ พร้อมการเรียนรู้แบบ คอนทราสต์ฟิสิกส์ เพื่อเพิ่มความสามารถในการจำแนกประเภทข้อมูลตาราง (Somepalli et al., 2021)

$$H_{\text{row}} = \text{Attention}(Q_{\text{row}}, K_{\text{row}}, V_{\text{row}}) \quad (6)$$

จากสมการที่ 6 ซึ่งใช้สำหรับเรียนรู้ความสัมพันธ์ระหว่างข้อมูลในแนวนอน หรือระหว่างตัวอย่าง (Samples) ต่าง ๆ ภายในชุดข้อมูล โดยนำข้อมูลเข้าสู่กระบวนการ Attention ผ่านองค์ประกอบ Query, Key และ Value แล้วคำนวณน้ำหนักความสนใจเพื่อพิจารณาว่าตัวอย่างใดมีความสัมพันธ์กับตัวอย่างอื่นมากน้อยเพียงใด ผลลัพธ์ที่ได้คือ H_{row} ซึ่งเป็นตัวแทนข้อมูลที่ได้รับการปรับปรุงจากสารสนเทศของตัวอย่างอื่น ๆ และสามารถนำไปใช้ในขั้นตอนการเรียนรู้และการจำแนกประเภทในลำดับถัดไป

5.3 วิธีการ TabTransformer

TabTransformer เน้นการฝังตัวแปรเชิงกลุ่มด้วยชั้นกลไกความใส่ใจในตัวเองก่อนส่งต่อให้ชั้นโครงข่ายประสาทเทียมแบบหลายชั้นทำนายผล (Huang et al., 2020)

$$\hat{y} = \text{MLP}(F) \quad (7)$$

จากสมการที่ 7 แสดงขั้นตอนการสร้างผลลัพธ์การทำนายของ TabTransformer โดยเวกเตอร์คุณลักษณะ F ที่ได้จากการรวมคุณลักษณะเชิงหมวดหมู่และเชิงตัวเลขจะถูกส่งเข้าสู่โครงข่ายประสาทเทียมแบบหลายชั้น (MLP) เพื่อประมวลผลและสร้างค่าทำนายสุดท้าย $\hat{y}$ สำหรับใช้ในการจำแนกประเภทข้อมูล

6.ขั้นตอนวิธี Optuna

Optuna เป็นโครงร่างการทำงานสำหรับการค้นหาพารามิเตอร์เสริม ที่ใช้แนวทาง การปรับปรุงให้เหมาะสมแบบเบย์ ร่วมกับขั้นตอนวิธีตัวประมาณค่าพาริเซนแบบโครงสร้างต้นไม้ เพื่อเลือกชุดค่าที่มีแนวโน้มให้ผลลัพธ์ที่ดีที่สุดอย่างมีประสิทธิภาพ (Grinsztajn et al., 2022b)

$$\theta^* = \arg \max_{\theta \in \Theta} f(\theta) \quad (8)$$

จากสมการที่ 8 Optuna จะทำการสุ่มหรือใช้คำแนะนำจากอัลกอริทึม (เช่น TPE) เพื่อเลือกชุดไฮเปอร์พารามิเตอร์ (θ) ภายใต้ขอบเขต (Θ) ที่กำหนดไว้ จากนั้นนำไปเทรนโมเดลเพื่อดูผลลัพธ์ประสิทธิภาพ ($f(\theta)$) หลังจากทดลองครบตามจำนวนรอบที่กำหนด สมการนี้จะทำหน้าที่คัดเลือกชุด (θ) ที่ทำคะแนนให้โมเดลได้สูงที่สุดออกมาเป็น (θ^*) เพื่อนำไปใช้งาน

7.ขั้นตอนวิธี SHAP

SHAP (SHapley Additive exPlanations) เป็นวิธีการตีความผลภายหลังที่มีรากฐานจากทฤษฎีเกม โดยคำนวณค่าคอนทริบิวชันของแต่ละตัวแปรต่อค่าพยากรณ์อย่างเป็นธรรม ทำให้สามารถอธิบายทั้งภาพรวมของแบบจำลอง และคำอธิบายเฉพาะรายได้พร้อมกัน (Lundberg & Lee, 2017)

8.ขั้นตอนวิธี LIME

LIME (Local Interpretable Model-agnostic Explanations) ใช้แนวทางที่แตกต่างออกไปโดยการประมาณพฤติกรรมของแบบจำลองที่ซับซ้อนด้วยแบบจำลองเชิงเส้นอย่างง่ายในบริเวณใกล้เคียงจุดข้อมูลที่สนใจ ทำให้ได้คำอธิบายเฉพาะจุดที่เข้าใจง่าย (Ribeiro et al., 2016)

10.งานวิจัยที่เกี่ยวข้อง

งานวิจัยของ Andishgar และคณะ (2024) (Andishgar et al., 2024) ได้ศึกษาปัจจัยเสี่ยงของโรคความดันโลหิตสูง ซึ่งพบที่มีความแตกต่างกันในแต่ละภูมิภาค โดยมีวัตถุประสงค์เพื่อพัฒนาแบบจำลองสำหรับพยากรณ์ความเสี่ยงของการเกิดโรคความดันโลหิตสูงในระยะเวลา 5 ปี ในกลุ่มประชากรชนบทของเทศฮิธราน การศึกษานี้ดำเนินการในลักษณะการติดตามผลระยะยาวโดยใช้ข้อมูลจาก Fasa Adults Cohort Study และประยุกต์ใช้อัลกอริทึมการเรียนรู้ของเครื่องแบบมีผู้สอน จำนวน 10 รูปแบบ อาทิ Logistic Regression (LR), Support Vector Machine (SVM), Random Forest (RF) และ Light Gradient Boosting Machine (LGBM) พร้อมทั้งประยุกต์ใช้เทคนิค SHapley Additive exPlanations (SHAP) เพื่อคัดเลือกคุณลักษณะที่สำคัญ ผลการศึกษาพบว่าแบบจำลอง LGBM ซึ่งใช้คุณลักษณะสำคัญ 30 อันดับแรก ให้ประสิทธิภาพดีที่สุด โดยมีค่า Area Under the Curve (AUC) เท่ากับ 0.67 และปัจจัยพยากรณ์หลักที่ระบุได้ ได้แก่ ค่าความดันโลหิตซิสโตลิก (SBP) ตั้งต้น เพศ และอัตราส่วนรอบเอวต่อสะโพก

(WHR) [43] สอดคล้องกับแนวคิดในการพัฒนาแบบจำลองที่มีทั้งความแม่นยำและความสามารถในการอธิบาย ผลงานวิจัยของ Asal (2025) (Asal, 2025) ได้นำเสนอเฟรมเวิร์กการเรียนรู้เชิงลึกที่ใช้สถาปัตยกรรม Transformer เพื่อตอบสนองความต้องการดังกล่าวในการตรวจจับภาวะหัวใจหยุดเต้นโดยใช้ข้อมูลทางคลินิกในรูปแบบตาราง การศึกษานี้ได้เปรียบเทียบประสิทธิภาพระหว่างแบบจำลอง TabTransformer และ FT-Transformer พร้อมทั้งใช้เทคนิค SHAP และ LIME เพื่ออธิบายกระบวนการตัดสินใจของแบบจำลอง ผลการทดลองแสดงให้เห็นว่าแบบจำลอง FT-Transformer มีความแม่นยำสูงถึงร้อยละ 98.1 ซึ่งเหนือกว่าแบบจำลอง TabTransformer ที่มีความแม่นยำร้อยละ 84.5 อย่างมีนัยสำคัญ โดยคุณลักษณะที่ส่งผลต่อการตัดสินใจของแบบจำลองมากที่สุด ได้แก่ จำนวนหลอดเลือดหลัก (ca) ผลการทดสอบความเครียดด้วยแทลเลียม (thal) และประเภทของอาการเจ็บหน้าอก (cp)

งานวิจัยของ Upadhyayula และ Pothugunta (Upadhyayula & Pothugunta, 2025b) ได้ทำการศึกษาเปรียบเทียบประสิทธิภาพระหว่างแบบจำลอง Transformer กับแบบจำลองการเรียนรู้ของเครื่องแบบดั้งเดิมในการพยากรณ์โรคหัวใจและหลอดเลือด โดยเฉพาะอย่างยิ่งภายใต้สถานการณ์ที่ข้อมูลมีความไม่สมดุล การศึกษานี้เปรียบเทียบแบบจำลองกลุ่มทราנסฟอร์เมอร์ ได้แก่ FT-Transformer, SAINT และ TabNet กับอัลกอริทึมการเรียนรู้ของเครื่องแบบดั้งเดิม ได้แก่ XGBoost และ Logistic Regression บนชุดข้อมูลสาธารณะ 3 ชุดที่มีขนาดและระดับความซับซ้อนแตกต่างกัน (UCI, Framingham และ Kaggle) ผลการวิจัยชี้ให้เห็นว่าประสิทธิภาพของแบบจำลองมีความสัมพันธ์อย่างมีนัยสำคัญกับลักษณะของชุดข้อมูลที่ใช้ โดยแบบจำลอง FT-Transformer และ Random Forest ให้ค่า AUC 0.99 บนชุดข้อมูลที่มีความสมดุล (UCI) แต่เมื่อทดสอบกับชุดข้อมูลที่ไม่สมดุล (Framingham) แบบจำลองส่วนใหญ่กลับมีความไวต่ำ แม้จะผ่านการปรับสมดุลข้อมูลแล้วก็ตาม โดยมีเพียงแบบจำลอง FT-Transformer ที่ยังคงให้ผลลัพธ์ที่สมดุลที่สุด นอกจากนี้งานวิจัยของ Si และคณะ (Si et al., 2025) ได้ทำการศึกษาเพื่อระบุความเสี่ยงของการเกิดโรคหัวใจในกลุ่มผู้ป่วยสูงอายุที่มีภาวะความดันโลหิตสูง โดยใช้ข้อมูลจากการศึกษาติดตามผลระยะยาว 7 ปี China Health and Retirement Longitudinal Study: CHARLS และพัฒนาแบบจำลองด้วยวิธี Logistic Regression, XGBoost และ Deep Neural Network ผลการศึกษาพบว่าร้อยละ 26.03 ของกลุ่มตัวอย่างเกิดโรคหัวใจภายในระยะเวลาที่ติดตาม โดยแบบจำลอง DNN ให้ค่า ROC-AUC สูงที่สุดที่ 0.67 ในขณะที่ Logistic Regression มีความสามารถในการเทียบมาตรฐานที่ดีที่สุด และ XGBoost ให้ประโยชน์ทางคลินิกที่ชัดเจนที่สุด

นอกจากนี้งานวิจัยของ Xu และคณะ (Xu et al., 2025) ได้กล่าวถึงประเด็นความทึบของอัลกอริทึมปัญญาประดิษฐ์ (AI) ในบริบทการดูแลสุขภาพ ซึ่งมีความเกี่ยวข้องโดยตรงกับการใช้เทคนิคปัญญาประดิษฐ์ที่อธิบายได้ ในงานวิจัยข้างต้นหลายชิ้น โดยชี้ให้เห็นว่าแม้ XAI จะถูกนำมาใช้เพื่อเพิ่มความเข้าใจในกระบวนการตัดสินใจของ AI แต่ก็มีหลักฐานบ่งชี้ว่าอาจนำไปสู่การพึ่งพาหรือความเอนเอียงที่มากเกินไปได้เช่นกัน คณะผู้วิจัยได้ทำการทดลองขนาดใหญ่สองชุดเพื่อเปรียบเทียบผลกระทบของ XAI โดยเฉพาะคำอธิบายที่สร้างจาก Multimodal Large Language Models ที่มีต่อประสิทธิภาพในการวินิจฉัยโรค ระหว่างกลุ่มบุคคลทั่วไป (จำนวน 623 คน) และแพทย์ปฐมภูมิ (จำนวน 153 คน) โดยใช้แบบจำลอง AI สำหรับการวินิจฉัยโรคผิวหนังเป็นกรณีศึกษา ผลการวิจัยพบว่าคำอธิบายที่สร้างจาก LLM มีลักษณะเป็น "ดาบสองคม" กล่าวคือ กลุ่มผู้ใช้ทั่วไปแสดงให้เห็นว่าเชื่อบบบอตโนมัติกเกินไปได้รับคำอธิบายดังกล่าว ในขณะที่แพทย์ผู้มีความรู้ทางการแพทย์ยังคงรักษาความยืดหยุ่นในการตัดสินใจและได้รับประโยชน์จากคำอธิบายโดยไม่คำนึงถึงความถูกต้องของผลลัพธ์จาก AI แต่อย่างใด

ขั้นตอนการดำเนินการวิจัย (Research Methodology)

งานวิจัยนี้ออกแบบเป็นการศึกษาเชิงเปรียบเทียบสมรรถนะของแบบจำลองการเรียนรู้ของเครื่อง (Machine Learning Benchmark Study) สำหรับการทำนายภาวะความดันโลหิตสูง (Hypertension) จากข้อมูลเชิงคลินิกและปัจจัยเสี่ยงของผู้ป่วย โดยยึดหลักการดำเนินงานที่สามารถทำซ้ำได้ (Reproducibility) ตรวจสอบย้อนกลับได้ (Traceability) และมีความเข้มงวดทางสถิติ (Statistical Rigor) ตลอดทั้งกระบวนการ ตั้งแต่การเก็บและทำความสะอาดข้อมูล การเตรียม

ข้อมูล การพัฒนาแบบจำลองพื้นฐาน จำนวน 9 แบบจำลอง การปรับจูนไฮเปอร์พารามิเตอร์อย่างเป็นระบบด้วย Optuna การประเมินผลด้วยการตรวจสอบไขว้ (Cross-Validation) การทดสอบนัยสำคัญทางสถิติ hทดลองจนการตรวจสอบความสามารถในการอธิบายผล (Explainability)

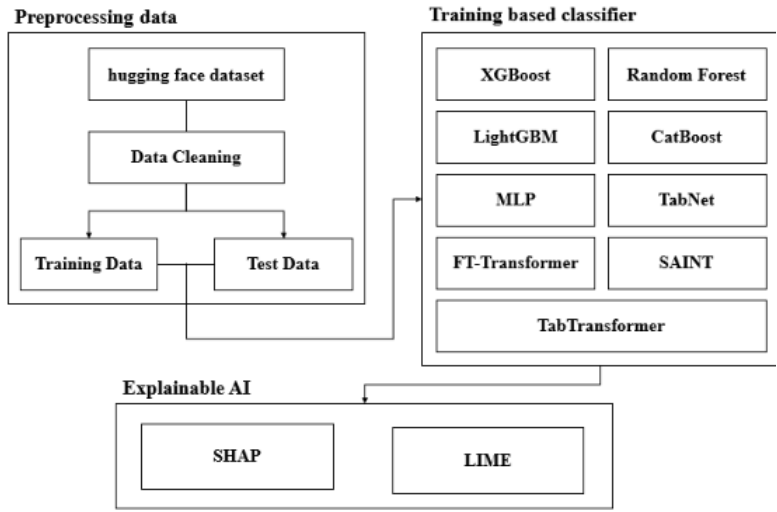


Figure 1 ผังแสดงกระบวนการทำงานของงานวิจัย

1.การเก็บรวบรวมข้อมูล (Data Acquisition)

ชุดข้อมูลที่ใช้ในการศึกษาประกอบด้วยระเบียบผู้ป่วยจำนวน 10,000 ราย และตัวแปร 25 ตัวแปร ครอบคลุมข้อมูลด้านประชากรศาสตร์ (เพศ อายุ ถิ่นที่อยู่) ดัชนีมวลกาย ประวัติทางพันธุกรรมและพฤติกรรมเสี่ยง (ประวัติครอบครัว เบาหวาน การสูบบุหรี่ การดื่มแอลกอฮอล์ กิจกรรมทางกาย การบริโภคเกลือสูง) ค่าความดันโลหิตซิสโตลิกและไดแอสโตลิก ประวัติการวินิจฉัยและการรักษา (การใช้ยา ความสม่ำเสมอในการรับประทานยา การควบคุมความดัน) ประวัติโรคร่วมทางหลอดเลือดและหัวใจ (โรคหลอดเลือดสมอง กล้ามเนื้อหัวใจตาย หัวใจล้มเหลว) และค่าชีวเคมีในเลือด (คอเลสเตอรอลรวม LDL HDL และครีเอตินิน) ตัวแปรเป้าหมาย (target) คือสถานภาพความดันโลหิตสูง (hypertension_status) ซึ่งเป็นตัวแปรทวิภาค 3.1.1 การตรวจสอบคุณภาพข้อมูล (Data Quality Audit) ก่อนเข้าสู่กระบวนการวิเคราะห์ ได้ดำเนินการตรวจสอบคุณภาพข้อมูลอย่างเป็นระบบ 4 ด้าน ได้แก่ โครงสร้างและชนิดข้อมูล (dtypes) ค่าสถิติเชิงพรรณนา ค่าที่หายไป (missing values) และค่าซ้ำซ้อน (duplicates) ผลการตรวจสอบสรุปได้ดังตารางที่ 1

ตารางที่ 1 สรุปผลการตรวจสอบคุณภาพข้อมูลเบื้องต้น

รายการตรวจสอบ	ผลการตรวจสอบ
ขนาดชุดข้อมูล (Shape)	10,000 แถว × 25 คอลัมน์
จำนวนแถวซ้ำซ้อน (Duplicate rows)	0 แถว
ตัวแปรที่มีค่าหายไป	medication_adherence: หายไป 3,131 ค่า (31.31%)
ประเภทตัวแปร	ตัวเลขทศนิยม 8 ตัว, บูลีน 12 ตัว, ข้อความ/หมวดหมู่ 5 ตัว

จากตารางที่ 1 แสดงผลการตรวจสอบคุณภาพข้อมูลเพื่อเตรียมความพร้อมก่อนการนำไปวิเคราะห์ โดยชุดข้อมูลมีขนาด 10,000 แถว และ 25 คอลัมน์ ไม่พบระเบียบข้อมูลที่ซ้ำซ้อน ในส่วนของประเภทตัวแปรมีการกระจายตัวทั้งในรูปแบบตัวเลข บูลีน และข้อหมวดหมู่ ทั้งนี้ พบข้อสังเกตสำคัญในส่วนของค่าสูญหาย ซึ่งปรากฏเฉพาะในตัวแปรพฤติกรรมการรับประทานยา (medication_adherence) จำนวน 3,131 ระเบียบ (คิดเป็นร้อยละ 31.31) ซึ่งข้อมูลการ

ตรวจสอบคุณภาพในตารางนี้ จะถูกนำไปใช้จัดทำระเบียบวิธีทำความสะอาดข้อมูลลำดับถัดไป เพื่อรักษาความถูกต้องเชิงสถิติของแบบจำลองพยากรณ์

2. การทำความสะอาดข้อมูล (Data Cleaning)

กระบวนการทำความสะอาดข้อมูลดำเนินการเป็นขั้นตอนที่ตรวจสอบย้อนกลับได้ โดยเปรียบเทียบภาพรวมข้อมูลก่อนและหลังการประมวลผลในทุกขั้น ประกอบด้วยขั้นตอนหลักดังนี้ (1) การเข้ารหัสตัวแปรหมวดหมู่ (Categorical Encoding): ตัวแปร sex และ residence ถูกแปลงเป็นตัวแปรดัมมี่ (One-Hot Encoding) เป็น sex_Male และ residence_Urban (2) การแปลงตัวแปรเป้าหมาย: hypertension_status ถูกเข้ารหัสเป็นค่าไบนารี (Hypertensive = 1, Normal = 0) (3) การแปลงตัวแปรบูลีนเป็นจำนวนเต็ม (Boolean-to-Integer Conversion) เพื่อความเข้ากันได้กับอัลกอริทึมทุกตระกูลที่ใช้ในงานวิจัย (4) การคัดตัวแปรที่ไม่เกี่ยวข้องกับการทำนายออก เช่น patient_id ซึ่งเป็นตัวระบุเฉพาะราย ภายหลังกระบวนการทำความสะอาดและแปลงข้อมูล ชุดข้อมูลสุดท้ายที่ใช้เป็นตัวแปรต้น (features) สำหรับการฝึกแบบจำลองมีจำนวน 17 ตัวแปร จากทั้งหมด 25 ตัวแปรเดิม

3. การเตรียมข้อมูล (Data Preparation)

3.1 การแบ่งชุดข้อมูลฝึกและทดสอบ (Train-Test Split)

ชุดข้อมูลถูกแบ่งออกเป็นชุดฝึก (Training Set) และชุดทดสอบ (Test Set) ในอัตราส่วน 80:20 ด้วยวิธี stratified split เพื่อรักษาสัดส่วนของตัวแปรเป้าหมายให้เท่ากันในทั้งสองชุด ดังแสดงในตารางที่ 2

ตารางที่ 2 การกระจายตัวของข้อมูลหลังการแบ่งชุดฝึกและชุดทดสอบ

ชุดข้อมูล	จำนวนตัวอย่าง	สัดส่วน Hypertensive	สัดส่วน Normal
ชุดฝึก	8,000	81.05% (n=6,484)	18.95% (n=1,516)
ชุดทดสอบ	2,000	81.05% (n=1,621)	18.95% (n=379)

เพื่อให้เห็นภาพการกระจายตัวของข้อมูลหลังการแบ่งชุดข้อมูลอย่างชัดเจน จึงนำเสนอข้อมูลดังแสดงใน

รูปที่ 2

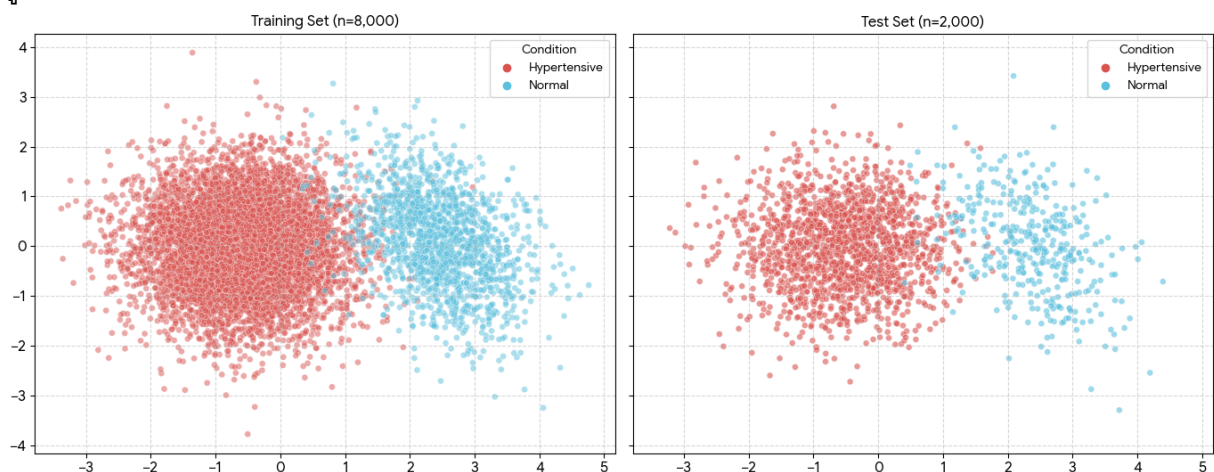


Figure 2 การกระจายตัวของข้อมูลในชุดฝึกสอนและทดสอบ

จากภาพประกอบที่ 2 เป็นการแสดงรายละเอียดการจัดเตรียมชุดข้อมูลเพื่อใช้ในการพัฒนาและประเมินประสิทธิภาพของแบบจำลอง โดยแบ่งข้อมูลทั้งหมดจำนวน 10,000 ตัวอย่าง ออกเป็น 2 ส่วน ด้วยวิธีการสุ่มแบบรักษาสัดส่วนข้อมูล (Stratified Sampling) ในอัตราส่วน 80:20 ได้แก่ ชุดฝึก (Training Set) จำนวน 8,000 ตัวอย่าง และ ชุด

ทดสอบ (Test Set) จำนวน 2,000 ตัวอย่าง โดยทั้งสองชุดข้อมูลประกอบด้วยตัวแปรนำเข้าจำนวน 17 ตัวแปรเท่ากัน ทั้งนี้ ได้มีการควบคุมสัดส่วนของกลุ่มตัวอย่างผู้ที่เป็นโรคความดันโลหิตสูง (Hypertensive) ให้มีสัดส่วนร้อยละ 81.05 และกลุ่มตัวอย่างที่มีความดันโลหิตปกติ (Normal) ให้มีสัดส่วนร้อยละ 18.95 เท่ากันทั้งในชุดฝึกและชุดทดสอบ เพื่อป้องกันความเอนเอียง (Bias) ของแบบจำลองในการเรียนรู้และการประเมินผล

จากตารางที่ 2 พบว่าตัวแปรเป้าหมายมีสัดส่วนไม่สมดุลอย่างชัดเจน (Hypertensive : Normal $\approx$ 4.3 : 1) เป็นข้อพิจารณาสำคัญที่นำไปสู่การออกแบบขั้นตอนการสุ่มเพิ่มข้อมูล (Resampling) ในหัวข้อ 3.3.3 เพื่อป้องกันมิให้แบบจำลองมีอคติไปทางกลุ่มที่มีจำนวนมากกว่า

3.2 การปรับมาตรฐานตัวแปร (Feature Scaling Policy)

เนื่องจากแบบจำลองที่นำมาเปรียบเทียบมีพื้นฐานทางคณิตศาสตร์ต่างกัน งานวิจัยนี้จึงกำหนดนโยบายการปรับมาตรฐานตัวแปรแยกตามกลุ่มของแบบจำลองดังตารางที่ 3 เพื่อให้แต่ละแบบจำลองได้รับข้อมูลนำเข้าในรูปแบบที่เหมาะสมที่สุดกับสถาปัตยกรรมของตน อันเป็นแนวปฏิบัติมาตรฐานของงาน benchmark ที่เปรียบเทียบแบบจำลองข้ามกลุ่ม

ตารางที่ 3 การปรับมาตรฐานตัวแปรจำแนกตามตระกูลแบบจำลอง

ตระกูลแบบจำลอง	แบบจำลอง	การสเกล
Tree-based	XGBoost, RandomForest, LightGBM, CatBoost	ไม่ปรับสเกล
Deep Learning	MLP, TabNet, FT-Transformer, SAINT, TabTransformer	ปรับสเกล

จากตารางที่ 3 แสดงแนวทางการจัดเตรียมข้อมูลและการประมวลผลล่วงหน้าด้านการปรับมาตรฐานส่วนข้อมูล (Data Scaling) สำหรับสถาปัตยกรรมและตระกูลแบบจำลองต่าง ๆ ที่ใช้ในการศึกษา โดยสามารถจำแนกแนวทางการจัดการได้เป็น 3 รูปแบบหลักดังที่แสดงในตาราง และข้อมูลที่ทำกรปรับมาตรฐานแล้วกับยังไม่ปรับมาตรฐานสามารถแสดงให้เห็นถึงขอบของข้อมูลได้ได้ภาพประกอบที่ 2

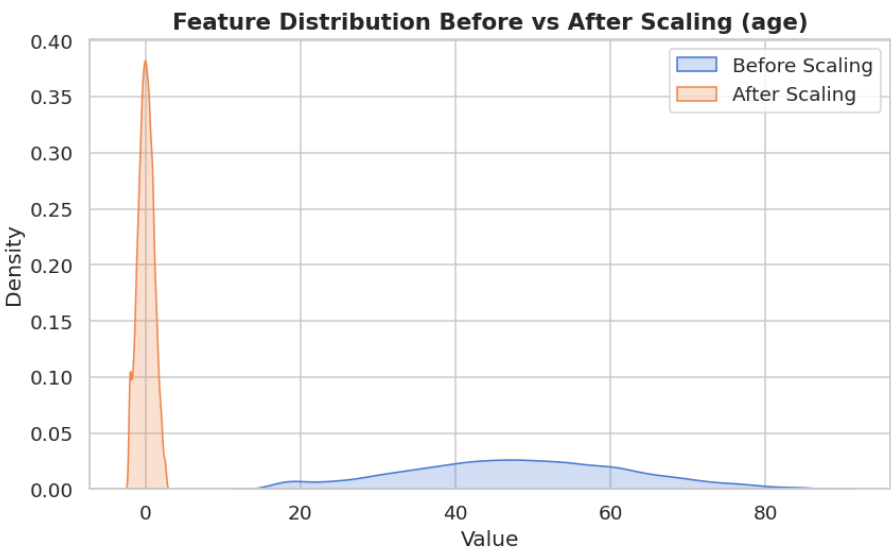


Figure 2 แสดงการเปรียบเทียบการกระจายตัวของตัวแปร age ก่อนและหลังการปรับมาตรฐาน

จากภาพที่ 2 จะแสดงการเปรียบเทียบการกระจายตัวของตัวแปร age ก่อนและหลังการปรับมาตรฐาน ข้อมูลด้วยเทคนิค Standardization โดยพบว่าภายหลังการปรับมาตรฐานแล้ว ข้อมูลถูกแปลงให้มีค่าเฉลี่ยใกล้เคียง ศูนย์และส่วนเบี่ยงเบนมาตรฐานเท่ากับหนึ่ง ซึ่งเป็นลักษณะการกระจายตัวที่เหมาะสมสำหรับการนำไปใช้ในการฝึกแบบจำลอง

3.3 การจัดการความไม่สมดุลของข้อมูล (Class Imbalance Handling)

ในการศึกษานี้ ได้ทำการเปรียบเทียบวิธีการจัดการปัญหาข้อมูลไม่สมดุลหลายรูปแบบ ประกอบด้วย การใช้ข้อมูลต้นฉบับโดยไม่ปรับสมดุลข้อมูล (Original Data) และเทคนิคการสุ่มเพิ่มตัวอย่างข้อมูล (Oversampling) เพื่อศึกษาผลกระทบของแต่ละวิธีต่อประสิทธิภาพของตัวแบบพยากรณ์ โดยผลการเปรียบเทียบจากการประเมินบนชุดข้อมูลทดสอบแสดงดังตารางที่ 4 และการกระจายตัวของข้อมูลในชุดข้อมูลที่ไม่สมดุลแสดงในภาพประกอบที่ 3

ตารางที่ 4 ผลการเปรียบเทียบค่า Accuracy ของตัวแบบภายหลังการใช้เทคนิคจัดการข้อมูลไม่สมดุล

MODEL	Original	ADASYN	SMOTE	SMOTENC	SMOTE-Tomek	SMOTE-ENN	UnderSampling
XGB	0.8845	0.8515	0.8525	0.8500	0.8550	0.8445	0.8250
RF	0.8760	0.8440	0.8580	0.8620	0.8545	0.8325	0.8075
LGBM	0.8785	0.8505	0.8600	0.8490	0.8510	0.8325	0.8215
Cat	0.8850	0.8510	0.8605	0.8440	0.8530	0.8275	0.8325
MLP	0.8830	0.8485	0.8535	0.8520	0.8525	0.8270	0.8100
TabNet	0.8740	0.8310	0.8325	0.7950	0.8340	0.7960	0.7570
FT	0.8825	0.8505	0.8470	0.8360	0.8530	0.8315	0.8255
SAINT	0.8825	0.8065	0.8440	0.8580	0.8505	0.8185	0.7845
Tab	0.8555	0.8045	0.7745	0.7890	0.8005	0.7540	0.7870

จากตารางที่ 4 ผลการทดลองพบว่า การใช้ข้อมูลต้นฉบับโดยไม่ดำเนินการปรับสมดุลข้อมูลให้ผลการพยากรณ์ที่มีประสิทธิภาพสูงสุด เมื่อพิจารณาจากค่า Accuracy ซึ่งเป็นตัวชี้วัดหลักที่ใช้ในการประเมินประสิทธิภาพของแบบจำลอง ดังนั้น การศึกษานี้จึงเลือกใช้ข้อมูลต้นฉบับในการพัฒนาและฝึกสอนตัวแบบสำหรับการทดลองทั้งหมด เนื่องจากการประยุกต์ใช้เทคนิคการสุ่มเพิ่มตัวอย่างข้อมูลไม่สามารถปรับปรุงประสิทธิภาพของแบบจำลองให้ดียิ่งขึ้น เมื่อเทียบกับการใช้ข้อมูลต้นฉบับ และอาจก่อให้เกิดข้อมูลสังเคราะห์ที่ไม่จำเป็นต่อกระบวนการเรียนรู้ของแบบจำลอง

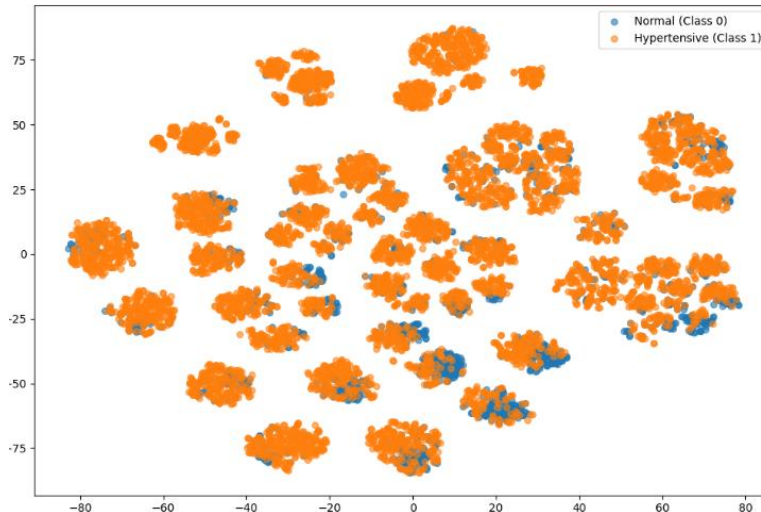


Figure 3 แสดงการเปรียบเทียบการกระจายตัวคลาสมาสมดุ

จากภาพที่ 3 พบว่าข้อมูลมีการกระจายตัวของกลุ่มตัวอย่างระหว่าง Normal (Class 0) และ Hypertensive (Class 1) ไม่เท่ากัน โดยกลุ่ม Hypertensive ซึ่งแสดงด้วยจุดสีส้มมีจำนวนมากกว่ากลุ่ม Normal ซึ่งแสดงด้วยจุดสีฟ้าอย่างเห็นได้ชัด สอดคล้องกับสัดส่วนของข้อมูลที่พบว่ากลุ่ม Hypertensive มีจำนวน 8,105 ตัวอย่าง (81.05%) ขณะที่กลุ่ม Normal มีจำนวน 1,895 ตัวอย่าง (18.95%) นอกจากนี้ ยังพบการกระจายตัวของข้อมูลที่มีการซ้อนทับกันระหว่างสองกลุ่มในหลายบริเวณ ซึ่งสะท้อนว่าข้อมูลบางส่วนมีลักษณะใกล้เคียงกันระหว่างกลุ่มและอาจทำให้ตัวแบบเรียนรู้การจำแนกกลุ่ม Normal ได้ยากขึ้น ดังนั้น จึงได้ทำการทดลองเปรียบเทียบแนวทางการจัดการความไม่สมดุลของข้อมูลหลายวิธี ได้แก่ Original Data, ADASYN, SMOTE, SMOTENC, SMOTE-Tomek, SMOTE-ENN และ UnderSampling และผลการทดลองได้แสดงแล้วดังตารางที่ 4

3.4 การพัฒนาแบบจำลอง (Model Development)

งานวิจัยนี้พัฒนาและเปรียบเทียบแบบจำลองพื้นฐาน (Base Learners) จำนวน 9 แบบจำลอง ครอบคลุม 3 กลุ่มสถาปัตยกรรมหลัก ได้แก่ (1) กลุ่ม Gradient-Boosted / Ensemble Trees (XGBoost, RandomForest, LightGBM, CatBoost) (2) กลุ่ม Deep Learning มาตรฐาน (MLP, TabNet) และ (3) กลุ่ม Transformer สำหรับข้อมูลตาราง (FT-Transformer, SAINT, TabTransformer) การคัดเลือกแบบจำลองในลักษณะนี้ทำให้กรอบการวิจัยครอบคลุมสเปกตรัมของแนวทางการเรียนรู้ของเครื่องที่ใช้กับข้อมูลตาราง (tabular data) ในปัจจุบันอย่างครบถ้วน

3.4.1 การค้นหาไฮเปอร์พารามิเตอร์ (Hyperparameter Optimization)

การปรับจูนไฮเปอร์พารามิเตอร์ของทุกแบบจำลองดำเนินการด้วยกรอบงาน Optuna โดยใช้จำนวนรอบการค้นหา (trials) เท่ากันที่ 30 trials ต่อแบบจำลอง เพื่อควบคุมความเป็นธรรมในการเปรียบเทียบทรัพยากรการค้นหา (Fair Search Budget) พื้นที่การค้นหา (search space) ของแต่ละแบบจำลองแสดงในตารางที่ 5

ตารางที่ 5 พื้นที่การค้นหาไฮเปอร์พารามิเตอร์จำแนกตามแบบจำลอง (Optuna, 30 trials/model)

ตัวแบบ	พารามิเตอร์
XGBoost	n_estimators, max_depth, learning_rate, subsample, colsample_bytree, min_child_weight, gamma, reg_alpha, reg_lambda, max_delta_step

ตัวแบบ	พารามิเตอร์
Random Forest	n_estimators, max_depth, min_samples_split, min_samples_leaf, max_features, bootstrap, criterion, max_leaf_nodes, min_impurity_decrease, ccp_alpha, class_weight
LightGBM	n_estimators, num_leaves, max_depth, learning_rate, min_child_samples, min_child_weight, subsample, subsample_freq, colsample_bytree, reg_alpha, reg_lambda, min_split_gain, max_bin
CatBoost	iterations, depth, learning_rate, l2_leaf_reg, border_count, random_strength, bagging_temperature, rsm, leaf_estimation_iterations
MLP	n_layers, n_units_l{i}, activation, alpha, learning_rate_init, batch_size, beta_1, beta_2, epsilon, tol
TabNet	n_d, n_a, n_steps, gamma, lambda_sparse, lr, weight_decay, batch_size, virtual_batch_size, momentum, mask_type, scheduler_gamma, scheduler_step_size
FT-Transformer	d_block, n_blocks, n_heads, attention_dropout, ffn_dropout, residual_dropout, lr, weight_decay, batch_size, patience, ffn_d_hidden_multiplier
SAINT	dim, depth, heads, dropout, mlp_hidden_multiplier, ff_mult, lr, weight_decay, batch_size, patience, gradient_clip_norm
TabTransformer	dim, depth, heads, attn_dropout, ff_dropout, mlp_hidden_multiplier, lr, weight_decay, batch_size, patience

จากตารางที่ 5 เป็นการสรุปรายการไฮเปอร์พารามิเตอร์ของแต่ละแบบจำลองที่ใช้ในกระบวนการปรับแต่งค่าเพื่อเพิ่มประสิทธิภาพในการประมวลผล โดยครอบคลุมทั้งตัวแปรควบคุมโครงสร้างการตัดสินใจสำหรับกลุ่มแบบจำลอง Tree-based และตัวแปรควบคุมการประมวลผลเชิงลึก เช่น อัตราการเรียนรู้ สถาปัตยกรรมชั้นโครงข่าย และกลไกความสนใจ (Attention Mechanism) สำหรับกลุ่มแบบจำลอง Deep Learning และ Transformer ทั้งนี้ ชุดพารามิเตอร์ดังกล่าวจะถูกนำไปใช้ในกระบวนการค้นหาค่าที่เหมาะสมที่สุดเพื่อเปรียบเทียบประสิทธิภาพของแต่ละแบบจำลองในขั้นตอนต่อไป

3.4.2 การกำหนดค่าการทดลอง (Run Configuration)

การทดลองทั้งหมดดำเนินการภายใต้การทดลองซึ่งมีรายละเอียดดังตารางที่ 6

ตารางที่ 6 การตั้งค่าพารามิเตอร์ระดับการทดลอง (Run Configuration)

พารามิเตอร์การทดลอง	ค่าที่กำหนด
จำนวนรอบการค้นหา Optuna (ต่อแบบจำลอง)	30 trials
จำนวนรอบฝึกสูงสุดของ MLP	100 iterations
จำนวน epoch สูงสุดของ TabNet	30 epochs

พารามิเตอร์การทดลอง	ค่าที่กำหนด
จำนวน epoch ของ FT-Transformer (CV / Final)	10 / 20 epochs
จำนวน epoch ของ SAINT (CV / Final)	10 / 15 epochs
จำนวน epoch ของ TabTransformer (CV / Final)	5 / 10 epochs

จากตารางที่ 6 แสดงรายละเอียดการกำหนดค่าพารามิเตอร์ในระดับการทดลอง (Run Configuration) เพื่อควบคุมสภาพแวดล้อมและสัดส่วนเวลาในการฝึกอบรมแบบจำลองทั้งหมด โดยกระบวนการปรับแต่งไฮเปอร์พารามิเตอร์ได้รับการควบคุมให้ดำเนินการค้นหาผ่านโครงกรอบงาน Optuna ในจำนวนที่เท่ากันคือ 30 รอบการทดลอง (Trials) ต่อหนึ่งแบบจำลอง เพื่อความเป็นธรรมในการทดสอบ (Fair Comparison) ในส่วนของแบบจำลองโครงข่ายประสาทเทียมและสถาปัตยกรรมกลุ่มความสนใจ (Attention-based Models) ได้มีการกำหนดจำนวนรอบในการเรียนรู้ (Iterations/Epochs) ที่เหมาะสมกับขนาดและความซับซ้อนของแต่ละสถาปัตยกรรมเพื่อป้องกันปัญหาการเรียนรู้เกิน (Overfitting) โดยเฉพาะในกลุ่มแบบจำลองไฮบริดและทรานส์ฟอร์เมอร์ (FT-Transformer, SAINT และ TabTransformer) ผู้วิจัยได้แยกการจำกัดจำนวน Epoch ออกเป็นสองระยะอย่างเป็นระบบ ได้แก่ ระยะการตรวจสอบไขว้ (Cross-Validation) เพื่อค้นหาพารามิเตอร์ และระยะการฝึกอบรมรอบสุดท้าย (Final Training) เพื่อเพิ่มประสิทธิภาพและความเสถียรในการพยากรณ์ข้อมูล

4. การประเมินผลแบบจำลอง (Model Evaluation)

4.1 การตรวจสอบไขว้ (Cross-Validation Protocol)

แบบจำลองพื้นฐานทั้ง 9 แบบถูกประเมินด้วยวิธีการตรวจสอบไขว้แบบ 5-fold (5-Fold Cross-Validation) บนชุดข้อมูลฝึกที่ผ่านการสุ่มเพิ่มข้อมูลแล้ว โดยในแต่ละ fold จะมีการบันทึกค่าตัวชี้วัด Accuracy, AUC-ROC, Precision, Recall และ F1-Score พร้อมทั้งบันทึกค่าเฉลี่ย (Mean) และส่วนเบี่ยงเบนมาตรฐาน (Std) ของแต่ละตัวชี้วัดตลอด 5 fold เพื่อสะท้อนทั้งระดับสมรรถนะและความเสถียรของแบบจำลอง (Model Stability)

4.2 ตัวชี้วัดสมรรถนะ (Evaluation Metrics)

การประเมินสมรรถนะของแบบจำลองบนชุดทดสอบ (Test Set) ใช้ตัวชี้วัดหลายมิติร่วมกัน เพื่อให้ครอบคลุมทั้งความสามารถในการจำแนก ความสามารถในการจัดอันดับความเสี่ยง และความน่าเชื่อถือของค่าความน่าจะเป็น ดังสรุปในตารางที่ 7

ตารางที่ 7 ตัวชี้วัดที่ใช้ในการประเมินสมรรถนะแบบจำลอง

ตัวชี้วัด	ความหมาย / วัตถุประสงค์การใช้งาน
AUC-ROC	ความสามารถในการจำแนกระหว่างกลุ่มผู้ป่วยที่เป็นและไม่เป็นโรคความดันโลหิตสูง ไม่ขึ้นกับจุดตัด (threshold)
Gini Coefficient	ดัชนีความสามารถในการจำแนก คำนวณจาก AUC ($Gini = 2 \times AUC - 1$)

ตัวชี้วัด	ความหมาย / วัตถุประสงค์การใช้งาน
Brier Score	ความคลาดเคลื่อนกำลังสองเฉลี่ยของค่าความน่าจะเป็นที่ทำนาย สะท้อนคุณภาพการปรับเทียบ (Calibration)
Accuracy	สัดส่วนการทำนายถูกต้องโดยรวม
Precision / Recall / F1-Score	ความแม่นยำและความครบถ้วนในการระบุกลุ่มผู้ป่วยที่เป็นโรครวมถึงค่าเฉลี่ยฮาร์โมนิก
Confusion Matrix	การกระจายของผลทำนายถูก/ผิดจำแนกตามกลุ่มจริง
Training Time (s)	ต้นทุนเชิงคำนวณของแต่ละแบบจำลอง ใช้ประกอบการพิจารณาความคุ้มค่าเชิงปฏิบัติ (Practical Trade-off)

จากตารางที่ 7 เป็นการอธิบายวัตถุประสงค์และการประยุกต์ใช้ตัวชี้วัดประสิทธิภาพรูปแบบต่าง ๆ ในการทดสอบแบบจำลองพยากรณ์ โดยมีการประเมินทั้งในส่วนของความสามารถในการจำแนกกลุ่ม (ผ่านค่า AUC-ROC, Gini, Accuracy, Precision, Recall และ F1-Score) คุณภาพของค่าความน่าจะเป็นที่พยากรณ์ได้ (ผ่านค่า Brier Score) รวมถึงการแจกแจงผลลัพธ์ผ่าน Confusion Matrix นอกจากนี้ ยังมีการคำนึงถึงมิติด้านทรัพยากรระบบด้วยการบันทึกระยะเวลาในการฝึกอบรมแบบจำลอง (Training Time) ทั้งนี้ ชุดตัวชี้วัดดังกล่าวจะช่วยให้ผู้วิจัยสามารถเปรียบเทียบข้อดีและข้อจำกัดของแต่ละสถาปัตยกรรมแบบจำลองได้อย่างเป็นรูปธรรมและรอบด้าน

5.การวิเคราะห์ความสำคัญของตัวแปรด้วย SHAP (SHAP Comparison)

เพื่อให้แบบจำลองมีความสามารถในการอธิบายผล (Explainability) อันจำเป็นต่อการนำไปประยุกต์ใช้ในบริบททางคลินิก งานวิจัยนี้ใช้วิธี SHAP (SHapley Additive exPlanations) วิเคราะห์ความสำคัญของตัวแปรในระดับภาพรวม (Global Feature Importance) สำหรับแบบจำลองทั้ง 9 แบบ โดยเลือกใช้ตัวอธิบาย (Explainer) ที่เหมาะสมกับสถาปัตยกรรมของแต่ละแบบจำลอง (TreeExplainer สำหรับแบบจำลองตระกูลต้นไม้ และ Kernel/Deep Explainer สำหรับแบบจำลองโครงข่ายประสาทเทียม) จากนั้นจึงสังเคราะห์ผลลัพธ์ข้ามแบบจำลอง (Cross-Model SHAP Feature Importance) เพื่อระบุตัวแปรที่มีอิทธิพลสอดคล้องกันในหลายแบบจำลอง

6.การวิเคราะห์เปรียบเทียบกับ LIME (LIME Comparison)

เนื่องจาก LIME (Local Interpretable Model-agnostic Explanations) ให้คำอธิบายในระดับรายตัวอย่าง (Local Explanation) งานวิจัยนี้จึงกำหนดตัวอย่างทดสอบตัวแทน (Representative Test Sample) เพียงตัวอย่างเดียวและใช้ในการอธิบายผลของทุกแบบจำลองอย่างสอดคล้องกัน เพื่อให้สามารถเปรียบเทียบน้ำหนักความสำคัญของตัวแปร (Feature Weight) ในบริบทเดียวกันข้ามแบบจำลอง ทั้งนี้แบบจำลองโครงข่ายประสาทเทียมบางส่วนที่พึ่งพาการประมวลผลบน GPU (TabNet, FT-Transformer, SAINT, TabTransformer) มีข้อจำกัดทางเทคนิคด้านการย้ายอุปกรณ์คำนวณ (device compatibility) กับกรอบงาน LIME ซึ่งได้บันทึกไว้อย่างโปร่งใสเป็นข้อจำกัดของระเบียบวิธี

7.การวิเคราะห์ความสอดคล้องระหว่าง LIME, SHAP และสหสัมพันธ์ (Agreement Analysis)

เพื่อตรวจสอบความน่าเชื่อถือร่วมกัน (Convergent Validity) ของวิธีการอธิบายผลทั้งสองวิธี งานวิจัยนี้คำนวณค่าสัมประสิทธิ์สหสัมพันธ์อันดับสเปียร์แมน (Spearman Rank Correlation) เปรียบเทียบอันดับความสำคัญของตัวแปรที่ได้จาก LIME, SHAP และค่าสหสัมพันธ์เชิงเดี่ยวระหว่างตัวแปรต้นกับตัวแปรเป้าหมาย (Feature-Target Correlation)

ทั้งสามคู่ (LIME–SHAP, LIME–Correlation, SHAP–Correlation) หากพบค่าสหสัมพันธ์ในระดับสูง จะเป็นหลักฐานสนับสนุนความสอดคล้องกันของกลไกการอธิบายผลทั้งสามแนวทาง และเพิ่มความเชื่อมั่นในตัวแปรที่ถูกระบุว่าเป็นปัจจัยเสี่ยงสำคัญ

ผลการทดลอง (Experimental Result)

ผลสมรรถนะบนชุดทดสอบ ผลการทดสอบนัยสำคัญทางสถิติแบบคู่และแบบพหุแบบจำลอง ผลการวิเคราะห์ความสำคัญของตัวแปรด้วย SHAP และ LIME ผลการวิเคราะห์ความสอดคล้องของวิธีอธิบายผล ผลการเปรียบเทียบความน่าจะเป็น และบทสรุปแบบจำลองที่มีสมรรถนะดีที่สุด ทั้งนี้ค่าตัวเลขทั้งหมดที่รายงานคำนวณจากผลลัพธ์จริงที่บันทึกไว้ในกระบวนการทดลอง (recorded run outputs)

1 .ผลการตรวจสอบไขว้ของแบบจำลองพื้นฐาน (5-Fold Cross-Validation)

ตาราง 8 แสดงค่าเฉลี่ยและส่วนเบี่ยงเบนมาตรฐานของตัวชี้วัดหลักจากการตรวจสอบไขว้ 5 fold บนชุดข้อมูลฝึกที่ผ่านการสุ่มเพิ่มข้อมูลด้วย SMOTENC เรียงลำดับตามค่า AUC-ROC เฉลี่ยจากมากไปน้อย

ตารางที่ 8 ผลการตรวจสอบไขว้ (5-Fold CV) ของแบบจำลองพื้นฐาน 9 แบบ เรียงตาม AUC เฉลี่ย

แบบจำลอง	CV Mean AUC ± SD	CV Mean Accuracy	CV Mean F1	CV Mean Precision	CV Mean Recall
SAINT	0.9243 ± 0.0050	0.8871	0.9323	0.9069	0.9596
LightGBM	0.9236 ± 0.0059	0.8891	0.9333	0.9108	0.9570
XGBoost	0.9236 ± 0.0061	0.8880	0.9326	0.9102	0.9562
CatBoost	0.9234 ± 0.0066	0.8887	0.9331	0.9100	0.9574
FT-Transformer	0.9234 ± 0.0043	0.8871	0.9320	0.9103	0.9553
MLP	0.9213 ± 0.0054	0.8889	0.9333	0.9080	0.9602
RandomForest	0.9141 ± 0.0085	0.8534	0.9059	0.9443	0.8704
TabNet	0.9138 ± 0.0051	0.8855	0.9309	0.9104	0.9525
TabTransformer	0.8522 ± 0.0087	0.8470	0.9093	0.8753	0.9465

จากตารางที่ 8 พบว่าแบบจำลองในกลุ่มบนสุด (SAINT, LightGBM, XGBoost, CatBoost, FT-Transformer) มีค่า AUC เฉลี่ยใกล้เคียงกันมากในช่วง 0.9234–0.9243 และมีส่วนเบี่ยงเบนมาตรฐานต่ำ ($SD \leq 0.0066$) สะท้อนความเสถียรของสมรรถนะระหว่าง fold ในระดับสูง ขณะที่ TabTransformer มีค่า AUC เฉลี่ยต่ำที่สุดอย่างชัดเจน (0.8522) ซึ่งต่ำกว่าอันดับถัดไปถึงประมาณ 0.06 หน่วย บ่งชี้ว่าแบบจำลองนี้ปรับตัวเข้ากับข้อมูลชุดนี้ได้ไม่ดีเท่าแบบจำลองอื่น

2.ผลสมรรถนะบนชุดทดสอบ (Test Set Performance)

เมื่อประเมินบนชุดทดสอบ (n = 2,000 ราย) ซึ่งไม่ได้ผ่านการสุ่มเพิ่มข้อมูล ผลของแบบจำลองพื้นฐานทั้ง 9 แบบสรุปได้ดังตาราง 9 โดยเรียงลำดับตามค่า AUC

ตารางที่ 9 สมรรถนะของแบบจำลองพื้นฐานบนชุดทดสอบ (Test Set, n=2,000)

แบบจำลอง	AUC	Gini	Brier Score	Accuracy	F1	เวลาฝึก (วินาที)
XGBoost	0.9209	0.8418	0.0826	0.8830	0.9291	53.78
LightGBM	0.9201	0.8402	0.0825	0.8875	0.9320	94.20
CatBoost	0.9189	0.8377	0.0833	0.8855	0.9309	470.85
SAINT	0.9170	0.8340	0.0855	0.8825	0.9282	400.04
FT-Transformer	0.9151	0.8302	0.0849	0.8825	0.9292	531.14
MLP	0.9120	0.8241	0.0883	0.8790	0.9264	586.04
RandomForest	0.9075	0.8151	0.1135	0.8465	0.9009	258.28
TabNet	0.8929	0.7858	0.0980	0.8640	0.9172	2385.41
TabTransformer	0.8375	0.6750	0.1085	0.8545	0.9136	18831.77

จากตารางที่ 9 แสดงผลบนชุดทดสอบ กล่าวคือ XGBoost, LightGBM และ CatBoost ครองอันดับต้นด้วยค่า AUC ที่ใกล้เคียงกันมาก ในขณะที่ TabTransformer มีสมรรถนะต่ำที่สุดในทุกตัวชี้วัด และใช้เวลาฝึกนานที่สุด (18,831.77 วินาที หรือประมาณ 5.2 ชั่วโมง) ซึ่งมากกว่า XGBoost ถึงประมาณ 350 เท่า สะท้อนถึงต้นทุนเชิงคำนวณที่ไม่คุ้มค่ากับสมรรถนะที่ได้ ข้อสังเกตสำคัญคือ RandomForest แม้จะมี Precision สูงสุดในบรรดาแบบจำลองพื้นฐาน (0.9451) แต่มีค่า Recall ต่ำที่สุด (0.8606) และมี Brier Score สูงที่สุด (0.1135) ซึ่งหมายความว่าโมเดลนี้มักพลาดในการระบุผู้ป่วยที่เป็นโรคจริง (False Negative สูง) และมีค่าความน่าจะเป็นที่ปรับเทียบได้ไม่ดี ซึ่งเป็นข้อจำกัดสำคัญในบริบทการคัดกรองโรค

3.ผลการวิเคราะห์ความสำคัญของตัวแปรด้วย SHAP

ผลการวิเคราะห์ความสำคัญของตัวแปรด้วยเทคนิค SHAP (SHapley Additive exPlanations) ถูกนำมาใช้เพื่ออธิบายอิทธิพลของตัวแปรที่มีต่อผลการทำนายของแบบจำลอง โดยภาพรวมจากการวิเคราะห์ทั้ง 9 แบบจำลองและการสังเคราะห์ผลข้ามแบบจำลอง พบว่าตัวแปรที่มีความสำคัญสูง ได้แก่ creatinine_mg_dl, ldl_mg_dl, total_cholesterol_mg_dl, age และ high_salt_diet ซึ่งปรากฏอย่างสม่ำเสมอในแบบจำลองหลายประเภท สะท้อนให้

เห็นว่าตัวแปรด้านชีวเคมี อายุ และพฤติกรรมการบริโภคมีบทบาทสำคัญต่อการจำแนกภาวะความดันโลหิตสูงของแบบจำลอง

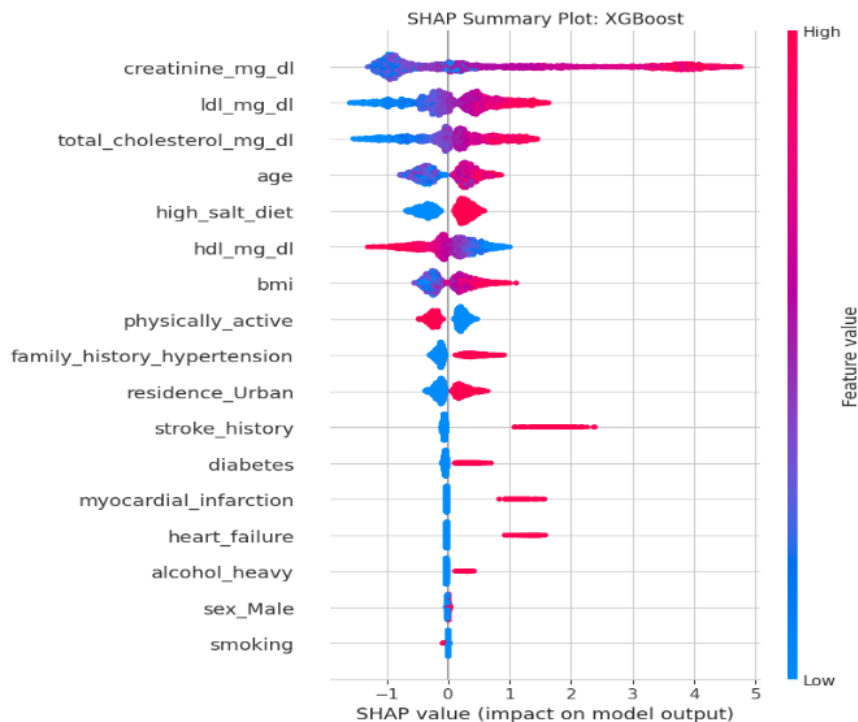


Figure 4 ผลการวิเคราะห์ SHAP Summary Plot ของแบบจำลอง XGBoost

จากรูปที่ 4 พบว่า creatinine_mg_dl เป็นตัวแปรที่มีความสำคัญสูงสุดของแบบจำลอง XGBoost โดยมีการกระจายของค่า SHAP กว้างที่สุดเมื่อเปรียบเทียบกับตัวแปรอื่น และค่าของตัวแปรในระดับสูงมีแนวโน้มสัมพันธ์กับค่า SHAP ทางบวก ซึ่งแสดงถึงการเพิ่มผลการทำนายของแบบจำลองไปในทิศทางของภาวะความดันโลหิตสูง รองลงมา ได้แก่ ldl_mg_dl, total_cholesterol_mg_dl, age และ high_salt_diet ตามลำดับ ทั้งนี้ ค่า SHAP ที่เป็นบวกหมายถึงตัวแปรมีส่วนเพิ่มค่าการทำนายของแบบจำลอง ขณะที่ค่า SHAP ที่เป็นลบหมายถึงมีส่วนลดค่าการทำนายดังกล่าว

นอกจากนี้ ยังพบรูปแบบที่น่าสนใจในตัวแปร hdl_mg_dl และ physically_active โดยค่าของตัวแปรในระดับสูงมีแนวโน้มให้ค่า SHAP ทางลบ สะท้อนว่าตัวแปรทั้งสองมีแนวโน้มลดค่าการทำนายภาวะความดันโลหิตสูงของแบบจำลอง ในขณะที่ตัวแปร family_history_hypertension, stroke_history, diabetes, myocardial_infarction และ heart_failure มีแนวโน้มให้ค่า SHAP ทางบวกเมื่อมีค่าหรือสถานะอยู่ในระดับสูง ซึ่งแสดงถึงบทบาทของประวัติทางสุขภาพและโรคร่วมในการจำแนกผลลัพธ์ของแบบจำลอง

เมื่อพิจารณาผลการวิเคราะห์ร่วมกันทั้ง 9 แบบจำลอง พบว่าตัวแปร creatinine_mg_dl, ldl_mg_dl, total_cholesterol_mg_dl, age และ high_salt_diet มีความสำคัญอย่างสม่ำเสมอในหลายแบบจำลอง แม้แบบจำลองเหล่านั้นจะมีโครงสร้างและกลไกการเรียนรู้ที่แตกต่างกัน ผลดังกล่าวแสดงให้เห็นถึงความสอดคล้องของรูปแบบข้อมูลที่แบบจำลองสามารถเรียนรู้ได้ และสนับสนุนว่าตัวแปรดังกล่าวเป็นตัวแปรสำคัญต่อการทำนายภาวะความดันโลหิตสูงในชุดข้อมูลที่ศึกษา อย่างไรก็ตาม ผล SHAP เป็นการอธิบายอิทธิพลของตัวแปรต่อ ผลการทำนายของแบบจำลอง และไม่ควรตีความว่าเป็นหลักฐานยืนยันความสัมพันธ์เชิงสาเหตุโดยตรง

เอกสารอ้างอิง

- Alfath, A. S., Wardhana, A. K., & Rumini, R. (2025). Hypertension Risk Prediction Using Stacking Ensemble of CatBoost, XGBoost, and LightGBM: A Machine Learning Approach. *Journal of Applied Informatics and Computing*, 9(6), 3146–3156. <https://doi.org/10.30871/jaic.v9i6.10370>
- Andishgar, A., Bazmi, S., Tabrizi, R., Rismani, M., Keshavarzian, O., Pezeshki, B., & Ahmadizar, F. (2024). Machine learning-based models to predict the conversion of normal blood pressure to hypertension within 5-year follow-up. *PLOS ONE*, 19(3), e0300201. <https://doi.org/10.1371/journal.pone.0300201>
- Arik, S. Ö., & Pfister, T. (2021). TabNet: Attentive Interpretable Tabular Learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(8), 6679–6687. <https://doi.org/10.1609/aaai.v35i8.16826>
- Asal, B. (2025). Transformer-Based Deep Learning Framework for Interpretable Cardiac Arrest Detection Using Tabular Clinical Data. *2025 10th International Conference on Computer Science and Engineering (UBMK)*, 847–852. <https://doi.org/10.1109/UBMK67458.2025.11206850>
- Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1), 5–32. <https://doi.org/10.1023/A:1010933404324>
- Chen, T., & Guestrin, C. (2016). XGBoost. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785–794. <https://doi.org/10.1145/2939672.2939785>
- Fujii, R. (2024). En route to conquer the silent killer “hypertension”: Integration of polygenic risk score with non-genetic determinants. *Hypertension Research*, 47(11), 3079–3081. <https://doi.org/10.1038/s41440-024-01826-0>
- Ghorbani, A. A., & Owrangh, K. (n.d.). Stacked generalization in neural networks: generalization on statistically neutral problems. *IJCNN'01. International Joint Conference on Neural Networks. Proceedings (Cat. No.01CH37222)*, 1715–1720. <https://doi.org/10.1109/IJCNN.2001.938420>
- Gorishniy, Y., Rubachev, I., Khrulkov, V., & Babenko, A. (2021). Revisiting Deep Learning Models for Tabular Data. *CoRR*, abs/2106.11959. <https://arxiv.org/abs/2106.11959>
- Grinsztajn, L., Oyallon, E., & Varoquaux, G. (2022a). Why do tree-based models still outperform deep learning on typical tabular data? *Proceedings of the 36th International Conference on Neural Information Processing Systems, NIPS '22*.
- Grinsztajn, L., Oyallon, E., & Varoquaux, G. (2022b). Why do tree-based models still outperform deep learning on typical tabular data? *Proceedings of the 36th International Conference on Neural Information Processing Systems, NIPS '22*.
- Huang, X., Khetan, A., Cvitkovic, M., & Kamin, Z. (2020). *TabTransformer: Tabular Data Modeling Using Contextual Embeddings*. <https://doi.org/10.48550/arXiv.2012.06678>
- Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. *Science*, 349(6245), 255–260. <https://doi.org/10.1126/science.aaa8415>
- Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T.-Y. (2017, July). *LightGBM: A Highly Efficient Gradient Boosting Decision Tree*.
- Kowalski, S., Goniewicz, K., Moskal, A., Al-Wathinani, A. M., & Goniewicz, M. (2023). Symptoms in Hypertensive Patients Presented to the Emergency Medical Service: A Comprehensive Retrospective

- Analysis in Clinical Settings. *Journal of Clinical Medicine*, 12(17), 5495.
<https://doi.org/10.3390/jcm12175495>
- Lundberg, S., & Lee, S.-I. (2017). *A Unified Approach to Interpreting Model Predictions*.
- Ma, H., Jiang, M., Fu, Z., Wang, Z., Shen, P., Shi, H., Feng, X., Chen, Y., Ding, X., Wu, Z., & Zhang, W. (2020). Clinical value of multiorgan damage in hypertensive crises: A prospective follow-up study. *The Journal of Clinical Hypertension*, 22(5), 914–923. <https://doi.org/10.1111/jch.13848>
- Miotto, R., Wang, F., Wang, S., Jiang, X., & Dudley, J. T. (2018). Deep learning for healthcare: review, opportunities and challenges. *Briefings in Bioinformatics*, 19(6), 1236–1246.
<https://doi.org/10.1093/bib/bbx044>
- Murtiningsih, D. A., Sari, B. W., & Fajri, I. N. (2025). Comparison of Light Gradient Boosting Machine, eXtreme Gradient Boosting, and CatBoost with Balancing and Hyperparameter Tuning for Hypertension Risk Prediction on Clinical Dataset. *Journal of Applied Informatics and Computing*, 9(5), 2753–2763. <https://doi.org/10.30871/jaic.v9i5.10400>
- Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., & Gulin, A. (2018). CatBoost: unbiased boosting with categorical features. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems* (Vol. 31). Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2018/file/14491b756b3a51daac41c24863285549-Paper.pdf
- Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). “Why Should I Trust You?”: Explaining the Predictions of Any Classifier. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16*, 1135–1144. <https://doi.org/10.1145/2939672.2939778>
- Si, F., Liu, Q., & Yu, J. (2025). A prediction study on the occurrence risk of heart disease in older hypertensive patients based on machine learning. *BMC Geriatrics*, 25(1), 27.
<https://doi.org/10.1186/s12877-025-05679-1>
- Sifat, I. K., & Kibria, Md. K. (2024). Optimizing hypertension prediction using ensemble learning approaches. *PLOS ONE*, 19(12), e0315865-. <https://doi.org/10.1371/journal.pone.0315865>
- Somepalli, G., Goldblum, M., Schwarzschild, A., Bruss, C., & Goldstein, T. (2021). *SAINT: Improved Neural Networks for Tabular Data via Row Attention and Contrastive Pre-Training*.
<https://doi.org/10.48550/arXiv.2106.01342>
- Upadhyayula, S. K., & Pothugunta, R. (2025a). Benchmarking Transformer-Based and Conventional Machine Learning Models for Cardiovascular Disease Prediction on Datasets of Varying Scale and Complexity. *MedRxiv*, 2025.08.03.25332878. <https://doi.org/10.1101/2025.08.03.25332878>
- Upadhyayula, S. K., & Pothugunta, R. (2025b). Harnessing Transformer Models for Cardiovascular Disease Prediction: A Comparison with Conventional Methods. *MedRxiv*.
<https://doi.org/10.1101/2025.08.03.25332878>
- Whelton, P. K., Carey, R. M., Aronow, W. S., Casey, D. E., Collins, K. J., Dennison Himmelfarb, C., DePalma, S. M., Gidding, S., Jamerson, K. A., Jones, D. W., MacLaughlin, E. J., Muntner, P., Ovbiagele, B., Smith, S. C., Spencer, C. C., Stafford, R. S., Taler, S. J., Thomas, R. J., Williams, K. A., ... Wright, J. T. (2018). 2017 ACC/AHA/AAPA/ABC/ACPM/AGS/APhA/ASH/ASPC/NMA/PCNA

Guideline for the Prevention, Detection, Evaluation, and Management of High Blood Pressure in Adults: Executive Summary: A Report of the American College of Cardiology/American Heart Association Task Force on Clinical Practice Guidelines. *Hypertension*, 71(6), 1269–1324.
<https://doi.org/10.1161/HYP.0000000000000066>

World Health Organization. (2025). *Global report on hypertension 2025: high stakes – turning evidence into action*. Geneva. World Health Organization.

Xu, X., Hu, H., Zhang, H., Wang, W. K., Wang, R., Soenksen, L. R., Badri, O., Jafry, S., Burger, E., Nwandu, L., Mehta, A., Duhaime, E. P., Qasim, A., Lin, H., Pereira, J., Hershon, J., Mui, P., Gru, A. A., Elhadad, N., ... Ghassemi, M. (2025). *Explainable AI as a Double-Edged Sword in Dermatology: The Impact on Clinicians versus The Public*. <https://arxiv.org/abs/2512.12500>

เริงฤดี ปธานวนิช, วิชัย เอกพลากร, & สำนักงานสถิติแห่งชาติ, ก. (2568). รายงานการสำรวจสุขภาพประชาชนไทย โดยการตรวจร่างกาย ครั้งที่ 7 พ.ศ. 2567–2568. คณะแพทยศาสตร์โรงพยาบาลรามาธิบดี มหาวิทยาลัยมหิดล.