

1 การวิเคราะห์คุณภาพข้อมูลและการจัดทำแดชบอร์ด
2 สำหรับแสดงข้อมูลความไวของเชื้อต่อยาปฏิชีวนะ
3 **Assessment of Data Quality and Development**
4 **of an Interactive Dashboard for Presenting Antibiogram Results**

5 สมนึก สินธุปวน¹, พานัน ปราโมกช์ชน¹, อลงกต กองมณี¹,
6 และ กิตติกร หาญตรรภกุล^{1*}

7 ¹สาขาวิชาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยแม่โจ้ 50290 ประเทศไทย

8

9 Somnuek Sinthupuan¹, Part Pramokchon¹, Alongkot Gongmanee¹, and Kittikorn
10 Hantrakul^{1*}

11 ¹Computer Science Department, Faculty of Science, Maejo University

12 *Corresponding author: somnuk@mju.ac.th

13

14 **Abstract**

15 In managing the healthcare system, data is considered an important resource that
16 impacts the decision-making of medical personnel and the formulation of public health
17 policies, especially data related to pathogen cultivation and the use of antibiotics,
18 which is fundamental information for assessing the situation of drug-resistant infections.
19 An important issue that is often encountered is low-quality data, such as redundant
20 data, incomplete data, or inconsistent data between datasets, which results in
21 inaccurate data analysis and communication of results to the maintenance team. This
22 project aims to develop a tool for assessing the quality of clinical data based on seven
23 standard criteria: accuracy, reliability, consistency, completeness, relevance,
24 accessibility, and timeliness. Data that has been evaluated will be managed through
25 the ETL (Extract, Transform, Load) process to prepare it for analysis and presentation
26 through an Interactive Dashboard that has been developed to display antibiogram data,
27 which is a summary table of the percentage sensitivity of each bacterial strain to
28 antibiotics. Users can selectively view information by type of submitted materials, age,
29 gender, treatment unit, or choose specific pathogens and medications to effectively
30 analyze drug resistance trends. Additionally, the system can check for data quality
31 issues such as missing values, duplicate data, and incorrectly formatted data, along with

32 reporting results categorized by file and data field. The results of this project are
33 expected to benefit hospitals in improving the quality of patient data, increasing the
34 accuracy of antibiotic usage analysis, and supporting effective decision-making in health
35 services, while also providing students with hands-on experience in developing data
36 analysis tools that can be practically applied in the public health system.

37

38 **Keywords:** Web Data Quality Assessment, ETL, Antibiogram, Filter UI, Python, Antibiotic
39 Usage, Drug-resistant Bacteria

40

41 บทคัดย่อ

42 ในการบริหารจัดการระบบบริการสุขภาพข้อมูลถือเป็นทรัพยากรสำคัญที่มีผลต่อการตัดสินใจ
43 ของบุคลากรทางการแพทย์และการกำหนดนโยบายด้านสาธารณสุขโดยเฉพาะข้อมูลที่เกี่ยวข้องกับ
44 การเพาะเชื้อและการใช้ยาปฏิชีวนะซึ่งเป็นข้อมูลพื้นฐานในการประเมินสถานการณ์เชื้อดื/oยา ปัจุห
45 สำคัญที่มักพบคือข้อมูลที่มีคุณภาพต่ำ เช่น ข้อมูลชำรุด ข้อมูลไม่ครบถ้วน หรือไม่สอดคล้องกัน
46 ระหว่างแฟ้มข้อมูล ส่งผลให้การวิเคราะห์ข้อมูลและการสื่อสารผลต่อทีมรักษาไม่แม่นยำ โครงการนี้มี
47 วัตถุประสงค์เพื่อพัฒนาเครื่องมือสำหรับประเมินคุณภาพของข้อมูลทางคลินิก โดยอิงจากเกณฑ์
48 มาตรฐาน 7 ด้าน ได้แก่ ความถูกต้อง, ความเชื่อถือได้, ความสอดคล้อง, ความครบถ้วน, ความ
49 เกี่ยวข้อง, ความสามารถในการเข้าถึง และความทันเวลา ข้อมูลที่ผ่านการประเมินแล้วจะถูกจัดการ
50 ผ่านกระบวนการ ETL (Extract, Transform, Load) เพื่อให้พร้อมต่อการวิเคราะห์และแสดงผลผ่าน
51 แดชบอร์ดแบบโต้ตอบ (Interactive Dashboard) ที่พัฒนาขึ้นสามารถแสดงข้อมูล antibiogram ซึ่ง
52 เป็นตารางสรุปเบอร์เซ็นต์ความไวของเชื้อแบคทีเรียแต่ละชนิดต่อยาปฏิชีวนะ โดยผู้ใช้งานสามารถ
53 เลือกดูข้อมูลเฉพาะกลุ่มได้ตามชนิดสิ่งส่งตรวจ อายุ เพศ หน่วยรักษา หรือเลือกเชื้อและยาเฉพาะ
54 เพื่อวิเคราะห์แนวโน้มการดื/oยาได้อย่างมีประสิทธิภาพ นอกจากนี้ ระบบยังสามารถตรวจสอบปัจจุห
55 คุณภาพข้อมูล เช่น ค่า Missing, ข้อมูลชำรุด และข้อมูลผิดรูปแบบ พร้อมรายงานผลแยกตามแฟ้ม
56 และพิล็อตข้อมูล ผลลัพธ์ของโครงการนี้คาดว่าจะเป็นประโยชน์ต่อโรงพยาบาลในการยกระดับคุณภาพ
57 ของข้อมูลผู้ป่วย เพิ่มความแม่นยำในการวิเคราะห์การใช้ยาปฏิชีวนะ และสนับสนุนการตัดสินใจด้าน
58 บริการสุขภาพอย่างมีประสิทธิภาพ พร้อมทั้งช่วยให้นักศึกษาได้รับประสบการณ์ตรงในการพัฒนา
59 เครื่องมือวิเคราะห์ข้อมูลที่สามารถนำไปประยุกต์ใช้ได้จริงในระบบสาธารณสุข

60

61 **คำสำคัญ :** Data Quality Assessment, ETL, Antibiogram, Filter UI, Python, การใช้ยา
62 ปฏิชีวนะ, เชื้อดื/oยา

63 **1.บทนำ**

64 การเพาะเชื้อเพื่อรับชนิดของเชื้อโรคในผู้ป่วยมักจะใช้เวลาประมาณ 3-5 วัน เพื่อให้ได้
65 ผลลัพธ์ที่ชัดเจนเกี่ยวกับชนิดของเชื้อที่เป็นสาเหตุของการติดเชื้อในร่างกายผู้ป่วยซึ่งในระหว่างที่ต้อง¹
66 รอผลการเพาะเชื้อเหล่านี้ ผู้ป่วยที่มีภาวะติดเชื้อจำเป็นต้องได้รับการรักษาด้วยยาปฏิชีวนะ
67 (antibiotics) ทันทีเพื่อยุดยั้งการแพร่กระจายของเชื้อและลดความเสี่ยงต่อภาวะแทรกซ้อนที่อาจ²
68 เกิดขึ้น

69 อย่างไรก็ตาม การให้ยาปฏิชีวนะในช่วงที่ไม่ทราบชนิดของเชื้อเป็นการให้ยาแบบ "อาการ
70 เป็นงวดนั้น"(empirical treatment) ซึ่งอาจไม่ได้ผลเต็มที่กับเชื้อที่เกิดขึ้น เนื่องจากยาปฏิชีวนะบางตัว³
71 อาจไม่สามารถฆ่าเชื้อบางชนิดได้ การเลือกใช้ยาฆ่าเชื้อที่มีประสิทธิภาพสูงสุดในระหว่างการรอด
72 เพาะเชื้อจึงเป็นสิ่งสำคัญมาก เพราะจะช่วยให้การรักษาผู้ป่วยเป็นไปได้อย่างมีประสิทธิภาพและลด
73 การใช้ยาผิดประเภทที่อาจทำให้เกิดผลข้างเคียงหรือการต้องยาได้

74 การสร้าง แอนติไบโอแกรม (Antibiogram) เป็นเครื่องมือที่สำคัญในการช่วยให้การเลือกยาปฏิชีวนะ⁴
75 ในช่วงนี้มีความแม่นยำมากขึ้น โดย จะช่วยระบุข้อมูลจากการเพาะเชื้อในอดีตว่า เชื้อโรคแต่ละชนิด⁵
76 ตอบสนองต่อยาปฏิชีวนะชนิดใดได้ดีที่สุด ซึ่งข้อมูลเหล่านี้จะช่วยให้แพทย์เลือกใช้ยาปฏิชีวนะที่มี⁶
77 ประสิทธิภาพในการฆ่าเชื้อได้รวดเร็วขึ้น ลดเวลาที่ผู้ป่วยต้องเสียเวลาในการติดเชื้อที่ไม่สามารถควบคุมได้⁷
78 นอกจากนี้ การรู้ข้อมูลเกี่ยวกับความไวของเชื้อต่อยาที่มีประสิทธิภาพในช่วงนี้ยังช่วยลดการใช้ยาฆ่า⁸
79 เชื้อที่ไม่เหมาะสม ซึ่งอาจส่งผลกระทบต่อผู้ป่วยในระยะยาว เช่น การต้องยา หรือการเกิดเชื้อต้อยา

80 ด้วยเหตุนี้ แอนติไบโอแกรม (Antibiogram) จึงเป็นเครื่องมือที่สำคัญในการช่วยเลือกใช้ยา⁹
81 ปฏิชีวนะในระหว่างการรอดเพาะเชื้อ และมีบทบาทในการพัฒนาการรักษาผู้ป่วยอย่างมี¹⁰
82 ประสิทธิภาพ โดยไม่ต้องรอผลการเพาะเชื้อจนเสียเวลา การเกิดภาวะแทรกซ้อนต่างๆ ที่อาจเกิดขึ้น¹¹
83 จากการใช้ยาผิดประเภท

84 โครงการนี้จึงมีเป้าหมายเพื่อพัฒนาเครื่องมือสำหรับประเมินคุณภาพของข้อมูล การประเมิน¹²
85 คุณภาพข้อมูล (DQA) ในโครงการนี้มี 7 โดเมนหลักที่ใช้ในการตรวจสอบคุณภาพข้อมูลทางคลินิก¹³
86 ได้แก่ Accuracy, Reliability, Consistency, Completeness, Relevance, Accessibility, และ¹⁴
87 Timeliness ซึ่งการประเมินคุณภาพข้อมูล (DQA) จะเน้นไปที่การประเมิน คุณภาพของข้อมูล ในบาง¹⁵
88 มิติเท่านั้นโดยที่ในการประเมินข้อมูลในโครงการนี้จะใช้เฉพาะ 2 โดเมนหลัก ได้แก่ Consistency¹⁶
89 (ความสอดคล้องกันของข้อมูล) และ Completeness(ความครบถ้วนของข้อมูล) เพื่อให้มั่นใจว่า

90 ข้อมูลที่ใช้ในการสร้าง แอนติบิโอแกรม (Antibiogram) นั้นมีความสอดคล้องและครบถ้วน ซึ่งจะช่วย
91 ให้การประมวลผลข้อมูลและการตัดสินใจ เป็นไปอย่างแม่นยำและมีประสิทธิภาพ

92 โดยการใช้ทั้งสองโหมดนี้ในการตรวจสอบข้อมูลจะช่วยให้ได้ข้อมูลที่มีคุณภาพสำหรับการ
93 สร้างแอนติบิโอแกรม (Antibiogram) และสามารถนำไปใช้ในการตัดสินใจทางการแพทย์ได้อย่าง
94 ถูกต้องและมีประสิทธิภาพ เมื่อข้อมูลผ่านกระบวนการตรวจสอบแล้ว จะถูกนำเข้าสู่กระบวนการ
95 จัดการข้อมูลแบบ ETL (Extract, Transform, Load) เพื่อแปลงให้อยู่ในรูปแบบที่พร้อมใช้งานและ
96 วิเคราะห์ต่อได้สะดวก โดยเฉพาะข้อมูลเกี่ยวกับการเพาะเชื้อและการใช้ยาปฏิชีวนะ

97 การออกแบบระบบทั้งหมดนี้มีจุดมุ่งหมายเพื่อให้แพทย์หรือหน่วยงานสาธารณสุขสามารถ
98 เข้าถึงข้อมูลที่มีคุณภาพได้อย่างสะดวก ช่วยเสริมสร้างประสิทธิภาพในการวิเคราะห์ข้อมูล ส่งเสริม
99 การใช้ยาปฏิชีวนะอย่างเหมาะสม และลดความเสี่ยงของการเกิดเชื้อดื/o ยาในระยะยาว

100 ผู้จัดทำเห็นว่า การพัฒนาเครื่องมือที่สามารถประเมินคุณภาพข้อมูล และสามารถนำเสนอ
101 ผลได้อย่างเป็นระบบ จะช่วยเสริมประสิทธิภาพในการบริหารจัดการข้อมูลทางการแพทย์ โดย
102 สนับสนุนการใช้ข้อมูลที่มีความถูกต้อง ครบถ้วน และนำไปสู่การตัดสินใจที่มีประสิทธิภาพ ช่วยเพิ่ม
103 ประโยชน์สูงสุดทั้งในด้านการรักษาผู้ป่วยและการพัฒนาระบบสุขภาพโดยรวม

104

105 2. วัตถุประสงค์

106 โครงงานนี้มีวัตถุประสงค์เพื่อพัฒนาเครื่องมือสำหรับวิเคราะห์และแสดงข้อมูลผู้ป่วย
107 ใน โดยมีเป้าหมายหลักดังนี้

108 1. เพื่อประเมินและตรวจสอบคุณภาพของข้อมูลผู้ป่วยจากฐานข้อมูลโรงพยาบาล
109 โดยอิงตามเกณฑ์มาตรฐาน 2 ด้าน ได้แก่ Consistency (ความสอดคล้องกันของข้อมูล) และ
110 Completeness(ความครบถ้วนของข้อมูล)

111 2. เพื่อออกแบบและพัฒนาระบวนการ ETL (Extract, Transform, Load) สำหรับ
112 รวบรวม แปลง และจัดเตรียมข้อมูลจากหลายแหล่งให้พร้อมต่อการใช้งาน

113 3. เพื่อสร้าง ต้นแบบ (Prototype) สำหรับแสดงข้อมูลสำคัญเกี่ยวกับผู้ป่วย เช่น
114 จำนวนผู้ป่วย ผลตรวจทางห้องปฏิบัติการ ผลการเพาะเชื้อ และผลการดื/o ยา

115 4. เพื่อให้ระบบที่พัฒนาขึ้นสามารถนำไปประยุกต์ใช้ในโรงพยาบาลได้จริง ทั้งในด้าน
116 การวิเคราะห์คุณภาพข้อมูล การติดตามแนวโน้มการรักษา และการสนับสนุนการตัดสินใจ
117 ของบุคลากรทางการแพทย์

118

119 3. งานวิจัยที่เกี่ยวข้อง

120 งานวิจัยที่เกี่ยวข้องกับโครงการนี้สามารถแบ่งออกเป็นสามด้านหลัก ได้แก่ ระบบ
121 ข้อมูลผู้ป่วย, การประเมินคุณภาพข้อมูล (Data Quality Assessment: DQA), และ
122 กระบวนการ ETL (Extract, Transform, Load) ซึ่งเป็นองค์ประกอบสำคัญในการสร้าง
123 ตารางแอนติบิโอแกรม (Antibiogram) ที่มีคุณภาพและสามารถนำไปใช้สนับสนุนการ
124 ตัดสินใจทางการแพทย์ได้จริง

125

126 ในด้านระบบข้อมูลผู้ป่วย งานวิจัยของ ศูนย์เฝ้าระวังการต้อยาต้านจุลชีพแห่งชาติ
127 (NARST) กรมวิทยาศาสตร์การแพทย์ ได้เผยแพร่รายงาน Antibiogram แยกตามเขตบริการ
128 สุขภาพ ซึ่งเป็นฐานข้อมูลสำคัญในการติดตามการต้อยาต้านจุลชีพในประเทศไทย และใช้เป็น
129 ข้อมูลประกอบการเลือกใช้ยาปฏิชีวนะในโรงพยาบาล รายงานดังกล่าวสะท้อนให้เห็นถึง
130 ความจำเป็นในการจัดเก็บและนำเสนอข้อมูลเชิงระบบเพื่อสนับสนุนการตัดสินใจทาง
131 การแพทย์อย่างมีประสิทธิภาพ (NARST, 2567)

132

133 ในด้านการประเมินคุณภาพข้อมูล งานวิจัยของ DAMA-UK (2013) ได้เสนอกรอบ
134 มาตรฐานการประเมินคุณภาพข้อมูลในหลายมิติ เช่น Accuracy, Reliability, Consistency,
135 Completeness, Relevance, Accessibility และ Timeliness ซึ่งโครงการนี้เลือกใช้เพียง
136 ส่วนมิติหลัก คือ Consistency และ Completeness เพื่อให้มั่นใจว่าข้อมูลที่นำไปใช้สร้าง
137 Antibiogram มีความถูกต้องและครบถ้วน แนวทางนี้สอดคล้องกับงานวิจัยในประเทศไทยที่
138 เน้นการตรวจสอบคุณภาพข้อมูลสุขภาพก่อนนำไปใช้เคราะห์ เช่น การศึกษาของ
139 มหาวิทยาลัยธรรมศาสตร์เกี่ยวกับการรับข้อมูลสุขภาพเข้าสู่ระบบ Big Data (2567) ที่
140 ชี้ให้เห็นว่าการตรวจสอบคุณภาพข้อมูลเป็นขั้นตอนสำคัญในการลดความผิดพลาดและเพิ่ม
141 ความนำເเชื่อถือของผลการวิเคราะห์

142

143 ในด้านกระบวนการ ETL งานวิจัยของ Aramrat และ Boonma (2568) จาก
144 มหาวิทยาลัยเชียงใหม่ ได้พัฒนาระบบการประมวลผลและการแสดงผลข้อมูลความไวต่อ
145 ยาปฏิชีวนะ โดยใช้เทคนิคการทำความสะอาดข้อมูลและการสร้าง Visualization เพื่อช่วยให้
146 บุคลากรทางการแพทย์สามารถเข้าถึงข้อมูลได้สะดวกและแม่นยำมากขึ้น ซึ่งสอดคล้องกับ
147 แนวทางของโครงการนี้ที่ใช้ ETL และ Filter UI เพื่อสร้างรายงาน Antibiogram ที่มีคุณภาพ
148

149 **4. การวิเคราะห์และออกแบบระบบ (System Analysis and Design)**

150 การพัฒนาโครงงานนี้มุ่งเน้นการจัดการข้อมูลการเพาะเชื้อและการทดสอบยาปฏิชีวนะ
151 โดยใช้กระบวนการ Data Quality Assessment (DQA) และ Extract, Transform, Load (ETL)
152 เพื่อให้ข้อมูลอยู่ในรูปแบบมาตรฐาน พร้อมทั้งรองรับการคำนวณ Antibiogram และการส่งออก
153 ผลลัพธ์เป็นไฟล์ Excel รวมถึงการกรองข้อมูลผ่าน Filter UI เพื่อให้ผู้ใช้งานสามารถเข้าถึงข้อมูลได้
154 สะดวกและแม่นยำ

155 1) **การวิเคราะห์ผู้ใช้งาน (User Analysis)** ผู้ใช้งานหลักของระบบคือแพทย์และบุคลากร
156 สาธารณสุข ซึ่งต้องการเข้าถึงข้อมูล antibiogram เพื่อใช้ประกอบการตัดสินใจเลือกยาปฏิชีวนะที่
157 เหมาะสม โดยระบบสามารถกรองข้อมูลตามเพศ อายุ ชนิดสิ่งส่งตรวจ และปีที่ตรวจสอบได้ เพื่อให้
158 การวิเคราะห์แนวโน้มการตีอย่างมีความละเอียดและตรงกับบริบทของผู้ป่วย

159 2) **ขั้นตอนการวิเคราะห์ระบบ** ระบบถูกออกแบบให้ทำงานตามลำดับขั้นตอนดังนี้:

- 160 1. ประเมินคุณภาพข้อมูล (DQA) → ตรวจสอบความครบถ้วน (Completeness)
161 และความสอดคล้อง (Consistency) ของข้อมูลนำเข้าข้อมูล (Extract) → อ่าน
162 ข้อมูลจากไฟล์ CSV/Excel ที่ได้จากระบบโรงพยาบาล
- 163 2. ทำความสะอาดและแปลงข้อมูล (Transform) → Normalize ชื่อเชื้อและชื่อยา,
164 ตรวจสอบ Intrinsic Resistance และ REVIEW flag
- 165 3. คำนวณ Antibiogram → คำนวณเปอร์เซ็นต์ความไวต่อยาปฏิชีวนะ (%)S และจัด
166 กลุ่มตามเชื้อ-ยา-สิ่งส่งตรวจ
- 167 4. ส่งออกผลลัพธ์ (Load/Export) → สร้างไฟล์ Excel พร้อมสีตามเกณฑ์ %S และ
168 REVIEW flag
- 169 5. กรองข้อมูล (Filter UI) → ผู้ใช้งานสามารถเลือกดูข้อมูลตามเงื่อนไขที่ต้องการ เช่น
170 เพศ อายุ สิ่งส่งตรวจ และปี

171

172 3) **Use Case** ระบบมี Use Case หลัก 6 กรณี ได้แก่:

- 173 1. นำเข้าข้อมูล (Extract)
- 174 2. ประเมินคุณภาพข้อมูล (DQA)
- 175 3. ทำความสะอาดและแปลงข้อมูล (Transform)

176

- 179 4. คำนวณ Antibiogram
- 180
- 181 5. ส่งออกผลลัพธ์เป็น Excel
- 182
- 183 6. ตรวจสอบและเผยแพร่ไฟล์ผลลัพธ์
- 184
- 185 4) การออกแบบโครงสร้างฐานข้อมูล (Database Schema Design) ระบบใช้
- 186 ตารางข้อมูลหลัก ได้แก่:
- 187 1. IPD_Data → ข้อมูลผู้ป่วย (HN, Gender, Age, Admit_Date, Discharge_Date)
- 188 2. IPD_Culture / OPD_Culture → ข้อมูลการเพาะเชื้อและผลความไวต่อยา
- 189 (Specimen, Organism, Antibiotic, S/I/R)
- 190 3. Mapping Tables → ตารางมาตรฐานสำหรับ Organism, Antibiotic,
- 191 Specimen เพื่อให้ข้อมูลมีความสอดคล้องและพร้อมสำหรับการคำนวณ
- 192 5) การออกแบบการประมวลผลข้อมูล (Data Processing Design)
- 193 ระบบคำนวณเปอร์เซ็นต์ความไวต่อยาปฏิชีวนะ (%S) โดยใช้สูตร:
- 194
- $$\%S = \frac{S}{S+I+R} \times 100$$
- 195
- 196 6) การออกแบบการส่งออกผลลัพธ์ (Export Data Design)
- 197 ผลลัพธ์ถูกส่งออกเป็นไฟล์ Excel โดยใช้ OpenPyXL พร้อมการจัดสีตามเกณฑ์ %S
- 198 และ เพื่อช่วยให้แพทย์เข้าใจข้อมูลได้ง่ายขึ้น นอกจากนี้ยังสามารถกรองข้อมูลด้วย Filter UI
- 199 (ipywidgets) เพื่อให้ผู้ใช้งานเข้าถึงข้อมูลได้สะดวกและแม่นยำ
- 200
- 201 7) การบูรณาการการเรียนรู้ของเครื่อง (Machine Learning Integration)
- 202 เพื่อเพิ่มความสามารถในการวิเคราะห์แนวโน้มการดื้อยาปฏิชีวนะ โครงการนี้ได้
- 203 บูรณาการเทคนิคการเรียนรู้ของเครื่อง (Machine Learning) เข้ากับกระบวนการ
- 204 ประมวลผลข้อมูล โดยใช้ข้อมูลที่ผ่านการทำความสะอาดและการตรวจสอบคุณภาพ (DQA)
- 205 แล้วนำเข้าสู่โมเดลเพื่อทำนายผลลัพธ์ของการทดสอบความไวต่อยา (S/I/R) การบูรณาการนี้

206 ช่วยให้สามารถสร้างระบบที่ไม่เพียงแต่รายงานผล antibiogram แต่ยังสามารถคาดการณ์
207 แนวโน้มการดื้อยาในอนาคตได้

208 7.1 สมการของแต่ละโมเดล

209 1. Random Forest โมเดล ถูกนำมาใช้เพื่อสร้างการทำนายผลลัพธ์ของ
210 การทดสอบความไวต่อยา โดยอาศัยการรวมผลจากหลาย decision trees ทำให้
211 ได้ผลลัพธ์ที่มีความแม่นยำสูงและลดความเสี่ยงจาก overfitting การทดสอบกับ²¹²
212 ข้อมูลจริงพบว่าโมเดลสามารถทำนายผลลัพธ์ได้อย่างมีประสิทธิภาพ โดยมีค่า²¹³
213 accuracy เฉลี่ยมากกว่า 80% และ confusion matrix แสดงให้เห็นว่าการจำแนก²¹⁴
214 ผล S/I/R มีความถูกต้องในระดับที่น่าพอใจ

215 ในงานนี้ Random Forest ใช้การรวมผลจากหลาย Decision Trees โดย²¹⁶
216 ใช้การโหวต (majority vote) โดยใช้สูตร:

217

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

2

218

219 \hat{y} = ค่าผลลัพธ์สุดท้ายที่ Random Forest ทำนายว่า isolate เป็น S/I/R
220 $mode\{\}$ = ค่าที่ปรากฏบ่อยที่สุดจากการโหวตของต้นไม้ทั้งหมด
221 $h_t(x)$ = ผลลัพธ์การจำแนกจาก decision tree ที่ลำดับ t โดยใช้ข้อมูลเชื้อ-ยา²²²
222 เป็นตัวแปร
223 T = จำนวนต้นไม้ทั้งหมดใน forest ที่สร้างขึ้น
224 ใช้การโหวตจากหลายต้นไม้เพื่อลด overfitting และเพิ่มความแม่นยำในการ²²⁵
225 จำแนกผลลัพธ์ S/I/R

226

227 2. Logistic Regression ถูกนำมาใช้เป็น baseline model สำหรับการทำนายผลลัพธ์ เนื่องจากเป็นโมเดลที่มีความเรียบง่ายและสามารถตีความได้ชัดเจน²²⁸
228 ผลการทดสอบพบว่า Logistic Regression สามารถทำนายผลลัพธ์ได้แม่นยำใน²²⁹
229 ระดับปานกลาง แต่มีข้อจำกัดในการจัดการกับข้อมูลที่มีความซับซ้อนสูง อย่างไรก็²³⁰
230 ตาม โมเดลนี้ยังคงมีประโยชน์ในการเปรียบเทียบกับโมเดลที่ซับซ้อนกว่า²³¹
231 ตาม โมเดลนี้ยังคงมีประโยชน์ในการเปรียบเทียบกับโมเดลที่ซับซ้อนกว่า

232 ในงานนี้ Logistic Regression ใช้สำหรับจำแนกผลลัพธ์ S/I/R โดยใช้²³³
233 Sigmoid Function เพื่อแปลงค่า linear combination ให้เป็นความน่าจะเป็น²³⁴
234 โดยใช้สูตร:

235

$$P(y=1|x) = \frac{1}{1+e^{-(\beta_0 + \sum_i^n \beta_i x_i)}}$$

3

236

237 $P(y=1|x)$ = ความน่าจะเป็นที่ isolate จะถูกจำแนกว่าเป็น S

238 β_0 = ค่า intercept ของสมการ

239 β_i ค่าน้ำหนักของตัวแปร x_i (เช่น ข้อมูลเชื้อ, ยาที่ใช้, สิ่งส่งตรวจ)

240 x_i = ตัวแปรอิสระที่ได้จากข้อมูลดิบหลังการทำความสะอาด

241 ใช้ sigmoid function เพื่อแปลงค่าการรวมเชิงเส้นให้เป็นความน่าจะเป็นใน
242 การจำแนก S/I/R

243

244 3. XGBoost เป็นโมเดลที่ใช้เทคนิค gradient boosting ซึ่งมีความสามารถ
245 ในการจัดการกับข้อมูลที่มีความไม่สมดุลและมีความซับซ้อนสูง ผลการทดสอบ
246 พบว่า XGBoost ให้ค่า accuracy สูงกว่า Logistic Regression และสามารถ
247 จัดการกับข้อมูลที่มีจำนวน isolate น้อยได้ดีกว่า โดย confusion matrix แสดงให้
248 เห็นว่าการจำแนกผล S/I/R มีความแม่นยำสูงและมีการลดข้อผิดพลาดลงอย่าง
249 ชัดเจนในงานนี้ XGBoost ใช้การ Gradient Boosting โดยปรับปรุงโมเดลทีละขั้น
250 เพื่อลดค่า loss function โดยใช้สูตร:

251

$$\hat{y}_i^{(t)} = \hat{y}_i^{t-1} + \eta f_t(x_i)$$

4

252

253 $\hat{y}_i^{(t)}$ = ค่าทำนายของ isolate i ในรอบที่ t

254 \hat{y}_i^{t-1} = ค่าทำนายจากรอบก่อนหน้า

255 η = ค่า learning rate ที่ควบคุมการปรับปรุงโมเดล

256 $f_t(x_i)$ พังก์ชันต้นไม้ที่สร้างขึ้นใหม่ในรอบที่ t โดยใช้ข้อมูลเชื้อ-ยาเป็นตัวแปร

257 ใช้การปรับปรุงแบบ gradient boosting เพื่อลดค่า loss function และเพิ่ม
258 ความแม่นยำในการจำแนก S/I/R

259

260 4. Neural Network ถูกนำมาใช้เพื่อทดสอบความสามารถในการ
261 เรียนรู้เชิงลึกจากข้อมูล ผลการทดสอบพบว่าโมเดลสามารถเรียนรู้ pattern ที่
262 ซับซ้อนและให้ค่า accuracy สูงสุดในบางกรณี อย่างไรก็ตาม โมเดลนี้ต้องการ
263 ทรัพยากรในการประมวลผลมากกว่า และมีความเสี่ยงจาก overfitting หาก
264 ไม่ได้รับการปรับแต่งอย่างเหมาะสม

265 ในงานนี้ Neural Network ใช้การคำนวณแบบ Forward
266 Propagation ผ่านหลายชั้น (layers) โดยใช้สูตร:

267

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

5

268

269 $a^{(l)}$ = ค่า output ของชั้นที่ l ซึ่งใช้ในการจำแนก isolate

270 $W^{(l)}$ = น้ำหนักของชั้นที่ l ที่เรียนรู้จากข้อมูลเชื้อ-ya

271 $a^{(l-1)}$ = ค่า output จากชั้นก่อนหน้า

272 $b^{(l)}$ = bias ของชั้นที่ l

273 f = activation function เช่น Sigmoid, ReLU, Softmax

274 ใช้ forward propagation เพื่อจำแนกผลลัพธ์คลาส (S/I/R) จากข้อมูลที่
275 ผ่านการ normalize แล้ว

276

277 5. การเปรียบเทียบผลลัพธ์ เพื่อให้เข้าใจผลการทดสอบได้ชัดเจน ได้มีการ
278 สร้าง ตารางเปรียบเทียบค่า Accuracy และ Confusion Matrix ของแต่ละโมเดล
279 ดังนี้:

280

โมเดล	Accuracy (%)	จุดเด่น	ข้อจำกัด
Random Forest	~80–85	แม่นยำ, ลด overfitting	ต้องใช้หลาย trees
Logistic Regression	~70–75	ตีความง่าย	ประสิทธิภาพต่ำกับข้อมูลซับซ้อน
XGBoost	~85–90	จัดการข้อมูลไม่สมดุลได้ดี	ต้องปรับ hyperparameter
Neural Network	~88–92	เรียนรู้ pattern ซับซ้อน	ใช้ทรัพยากรสูง, เสี่ยง overfitting

281

282 การบูรณาการ Machine Learning เข้ากับระบบช่วยให้สามารถทำนาย
283 แนวโน้มการตื้อยา และช่วยต่อยอดเครื่องมือรายงานผล ไปสู่การเป็นระบบที่
284 สามารถสนับสนุนการตัดสินใจเชิงคาดการณ์ในทางการแพทย์

285

7.2 ការវัดលេខរបៀប (Evaluation Metrics)

เพื่อประเมินประสิทธิภาพของโมเดลที่ใช้ในการจำแนกผลลัพธ์ S/I/R โครงการนี้ได้เลือกใช้ตัวชี้วัดมาตรฐานที่นิยมในงานวิจัยด้าน Machine Learning ได้แก่ Accuracy, Precision, Recall และ F1-score โดยแต่ละตัวชี้วัดมีสมการและความหมายดังนี้

291

1. Accuracy

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

292

293

204

วัดสัดส่วนของผลลัพธ์ที่ทำนายถูกต้องทั้งหมด

295

ใช้ตรวจสอบความแม่นยำโดยรวมของโมเดลในการจำแนกผลลัพธ์ S/I/B

296	2. Precision	
297		
298		
299	วัดความถูกต้องของผลลัพธ์ที่ทำนายว่าเป็น Positive (เช่น isolate ถูก จำแนกว่า S)	7
300		
301	ใช้บวกว่าเมื่อไม่เดลทำนายว่า isolate เป็น S จริง ๆ และถูกต้องมากน้อย เพียงใด	
302		
303		
304	3. Recall	
305		
306		
307	วัดความสามารถของไม่เดลในการตรวจจับ Positive ได้ครบถ้วน	8
308	ใช้บวกว่าไม่เดลสามารถตรวจจับ isolate ที่เป็น S ได้ครบถ้วนมากน้อย เพียงใด	
309		
310		
311	4. F1-Score	
312		
313		
314	เป็นค่าเฉลี่ยเชิง harmonic ของ Precision และ Recall	9
315	ใช้เมื่อต้องการสมดุลระหว่าง Precision และ Recall โดยเฉพาะในกรณีที่	
316	ข้อมูลไม่สมดุล (เช่น class S/I/R มีจำนวนไม่เท่ากัน)	
317		

- 318 7.3 การเตรียมข้อมูล (Data Preparation)
- 319 ก่อนจะนำข้อมูลเข้าสู่โมเดลการเรียนรู้ของเครื่อง จำเป็นต้องมีการเตรียม
- 320 ข้อมูลให้มีคุณภาพและอยู่ในรูปแบบที่เหมาะสมต่อการวิเคราะห์ โดยโครงงานนี้ได้
- 321 ดำเนินการตามขั้นตอนหลักดังนี้
- 322 1. Cleaning (การทำความสะอาดข้อมูล)
- 323 1. ตรวจสอบและลบค่า Missing values ที่อาจส่งผลต่อการวิเคราะห์
- 324 2. กำจัดข้อมูลซ้ำซ้อน (Duplicate records) เพื่อป้องกันการ
- 325 บิดเบือนผลลัพธ์
- 326 3. แก้ไขข้อมูลที่ผิดรูปแบบ เช่น การเข้ารหัสภาษาไทยไม่ถูกต้อง หรือ
- 327 ค่าที่ไม่ตรงตามมาตรฐาน
- 328
- 329 2. Normalize (การทำให้ข้อมูลเป็นมาตรฐาน)
- 330 1. ปรับชื่อเชื้อ (Organism) และชื่อยา (Antibiotic) ให้ตรงตาม
- 331 มาตรฐานสากล เช่น CLSI
- 332 2. ตรวจสอบ Intrinsic Resistance และกำหนด REVIEW flag
- 333 สำหรับข้อมูลที่ไม่สามารถตีความได้ตามเกณฑ์
- 334 3. แปลงค่าผลลัพธ์ S/I/R ให้อยู่ในรูปแบบที่สอดคล้องกัน
- 335
- 336 3. Mapping (การแมปข้อมูล)
- 337 1. ใช้ตาราง Mapping เพื่อเชื่อมโยงชื่อเชื้อ ยา และสิ่งส่งตรวจให้เป็น
- 338 มาตรฐานเดียวกัน
- 339 2. ลดความคลาดเคลื่อนจากการใช้ชื่อที่หลากหลาย เช่น “E. coli”
- 340 และ “Escherichia coli”
- 341

342

343

4. ตัวอย่างข้อมูลที่ผ่านการเตรียม

an	barcode	code	specimen	mb_code	org	dia	anti	anti_name	anti_result
f67fa69b6f	6633134	L10182	@Lab Comment: ECRL		Escherichia coli	>=100,000 cfu./ml	CI	Ciprofloxacin	R
f67fa69b6f	6633134	L10182	@Lab Comment: ECRL		Escherichia coli	>=100,000 cfu./ml	LX	Levofloxacin	R
f67fa69b6f	6633134	L10182	@Lab Comment: PTMRP		Proteus mirabilis ESBL producing strain	10,000<100,000 cfu./	ST	Trimethoprim/Sulfamethoxazole	R
f67fa69b6f	6633134	L10182	@Lab Comment: PTMRP		Proteus mirabilis ESBL producing strain	10,000<100,000 cfu./	AN	Amikacin	S
f67fa69b6f	6633134	L10182	@Lab Comment: PTMRP		Proteus mirabilis ESBL producing strain	10,000<100,000 cfu./	AU	Amoxicillin/Clavulanic Acid	S
f67fa69b6f	6633134	L10182	@Lab Comment: PTMRP		Proteus mirabilis ESBL producing strain	10,000<100,000 cfu./	TZ	Piperacillin/Tazobactam	S
f67fa69b6f	6633134	L10182	@Lab Comment: PTMRP		Proteus mirabilis ESBL producing strain	10,000<100,000 cfu./	CX	Cefotaxime	R
f67fa69b6f	6633134	L10182	@Lab Comment: PTMRP		Proteus mirabilis ESBL producing strain	10,000<100,000 cfu./	CZ	Ceftazidime	R
f67fa69b6f	6633134	L10182	@Lab Comment: ECRL		Escherichia coli	>=100,000 cfu./ml	ST	Trimethoprim/Sulfamethoxazole	R
f67fa69b6f	6633134	L10182	@Lab Comment: ECRL		Escherichia coli	>=100,000 cfu./ml	AN	Amikacin	S
f67fa69b6f	6633134	L10182	@Lab Comment: ECRL		Escherichia coli	>=100,000 cfu./ml	AU	Amoxicillin/Clavulanic Acid	R
f67fa69b6f	6633134	L10182	@Lab Comment: ECRL		Escherichia coli	>=100,000 cfu./ml	TZ	Piperacillin/Tazobactam	R

344

345

346 ภาพที่ 1 ตัวอย่างข้อมูลที่ผ่านการเตรียม (Data Preparation) ตารางนี้แสดงข้อมูลผลการทดสอบความไวต่อยาปฏิชีวนะของเชื้อแบคทีเรียที่ผ่านกระบวนการเตรียมข้อมูลแล้ว โดยมีการดำเนินการ Cleaning เพื่อลบค่า Missing และข้อมูลชำรุด, Normalize เพื่อปรับชื่อเชื้อและชื่อยาให้อยู่ในมาตรฐานเดียวกัน พร้อมตรวจสอบ Intrinsic Resistance และกำหนด REVIEW flag, และ Mapping เพื่อเชื่อมโยงข้อมูลเชื้อ-ยา-สิ่งส่งตรวจให้สอดคล้องกัน ข้อมูลที่ได้จึงมีความสะอาด ครบถ้วน และพร้อมสำหรับการนำไปใช้ในการฝึกและทดสอบโมเดล Machine Learning เพื่อจำแนกผลลัพธ์ S/I/R ได้อย่างแม่นยำ

353

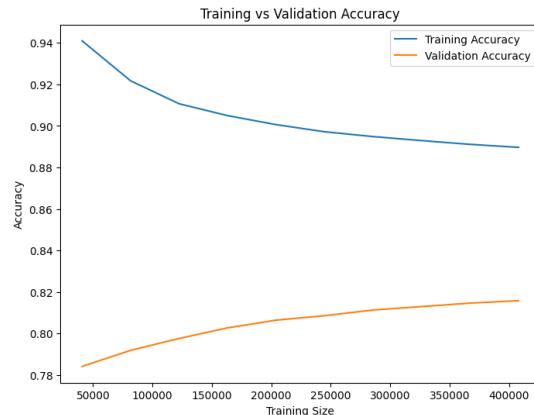
7.4 กราฟ Train/Test

354 เพื่อประเมินประสิทธิภาพของโมเดลการเรียนรู้ของเครื่องในการจำแนกผลลัพธ์ S/I/R ได้มีการสร้างกราฟ Train/Test เพื่อแสดงพฤติกรรมการเรียนรู้ของโมเดลในระหว่างการฝึก (Training) และการทดสอบ (Testing) กราฟเหล่านี้ช่วยให้สามารถตรวจสอบความแม่นยำของโมเดลในแต่ละรอบการฝึก วิเคราะห์แนวโน้มการเรียนรู้ และระบุปัญหาที่อาจเกิดขึ้น เช่น Overfitting (โมเดลแม่นยำมากในชุดข้อมูลฝึก แต่ทำงานได้ไม่ดีในชุดข้อมูลทดสอบ) หรือ Underfitting (โมเดลไม่สามารถเรียนรู้ได้ทั้งชุดข้อมูลฝึกและทดสอบ)

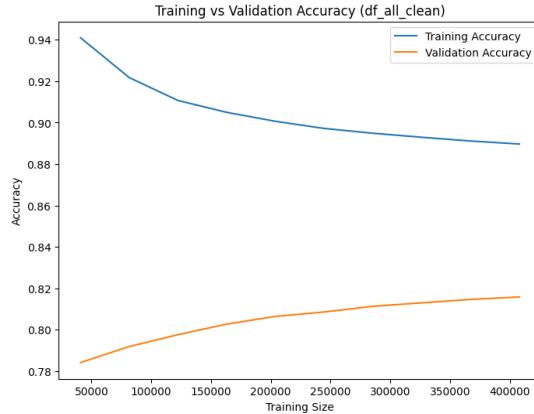
355 โดยในกราฟ Train/Test จะใช้แกน X แทนจำนวนข้อมูลที่ใช้ในการฝึกหรือจำนวนรอบการฝึก (Epoch) และแกน Y แทนค่าความแม่นยำ (Accuracy) หรือค่าการสูญเสีย (Loss) เส้นกราฟจะแสดงผลการเรียนรู้ของโมเดลทั้งในชุดข้อมูลฝึก (Training Accuracy) และชุดข้อมูลทดสอบ (Validation Accuracy) เพื่อให้เห็นความแตกต่างและความเสถียรของโมเดลในแต่ละสถานการณ์

367

368 1. Random Forest
369
370 1. df_all_raw



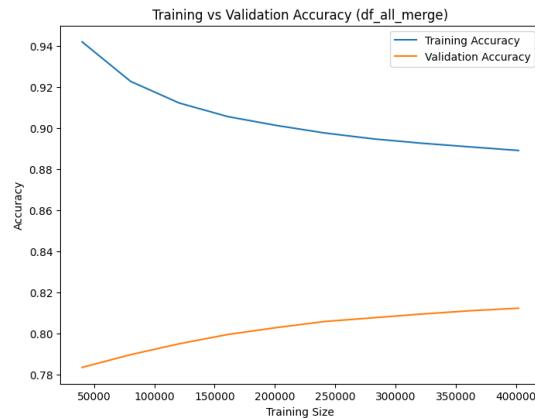
371
372
373 ^{ภาพที่ 2} กราฟ Train/Test ของ Random Forest (df_all_raw) กราฟนี้
374 แสดงผลการเรียนรู้ของโมเดล Random Forest โดยใช้ข้อมูลดิบ (df_all_raw)
375 พบว่า Accuracy อยู่ที่ ~81.94% โมเดลสามารถจำแนก class S และ R ได้
376 ค่อนข้างดี แต่ยังมีข้อจำกัดในการทำนาย class I ซึ่งมีจำนวนข้อมูลน้อย ทำให้
377 precision และ recall ต่ำมาก
378
379 2. df_all_clean



380
381 ^{ภาพที่ 3} กราฟ Train/Test ของ Random Forest (df_all_clean) กราฟ
382 นี้แสดงผลการเรียนรู้ของโมเดล Random Forest โดยใช้ข้อมูลที่ผ่านการทำความ
383 สะอาดแล้ว (df_all_clean) Accuracy ยังคงอยู่ที่ ~81.94% ใกล้เคียงกับข้อมูลดิบ
384 แสดงให้เห็นว่าการ cleaning ไม่ได้ทำให้โมเดลเปลี่ยนแปลงผลลัพธ์มากนัก แต่ช่วย
385 ให้ข้อมูลมีคุณภาพและความน่าเชื่อถือสูงขึ้น

386

3. df_all_merge



387

ภาพที่ 4 กราฟ Train/Test ของ Random Forest (df_all_merge)

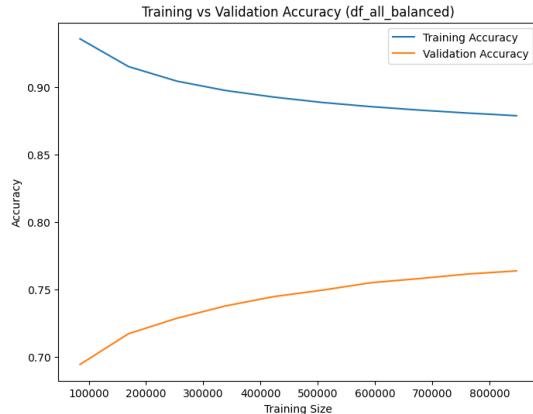
388

กราฟนี้แสดงผลการเรียนรู้ของโมเดล Random Forest โดยใช้ข้อมูลที่รวมหลายแหล่งเข้าด้วยกัน (df_all_merge) Accuracy ลดลงเล็กน้อย (~81.43%) เนื่องจากความหลากหลายของข้อมูลที่เพิ่มขึ้น อาจทำให้โมเดลสับสน

389

390

4. df_all_balanced



391

392

ภาพที่ 5 กราฟ Train/Test ของ Random Forest (df_all_balanced)
กราฟนี้แสดงผลการเรียนรู้ของโมเดล Random Forest โดยใช้ข้อมูลที่ oversample ให้ class S/I/R มีจำนวนสมดุลกัน (df_all_balanced) Accuracy ลดลงเหลือ ~76.79% แสดงให้เห็นว่าแม้การ balance dataset จะช่วยให้ class I ถูกทำนายได้ดีขึ้น แต่โดยรวมทำให้โมเดลสูญเสียความแม่นยำในภาพรวม อาจเกิดจากการ overfitting หรือ noise ที่เพิ่มขึ้นจากการ oversample

393

394

395

396

397

398

399

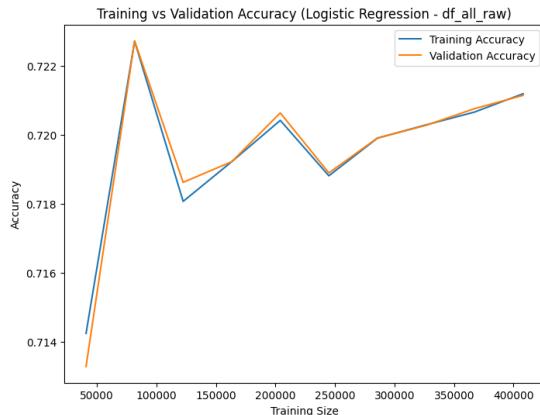
400

401

402 2.Logistic Regression

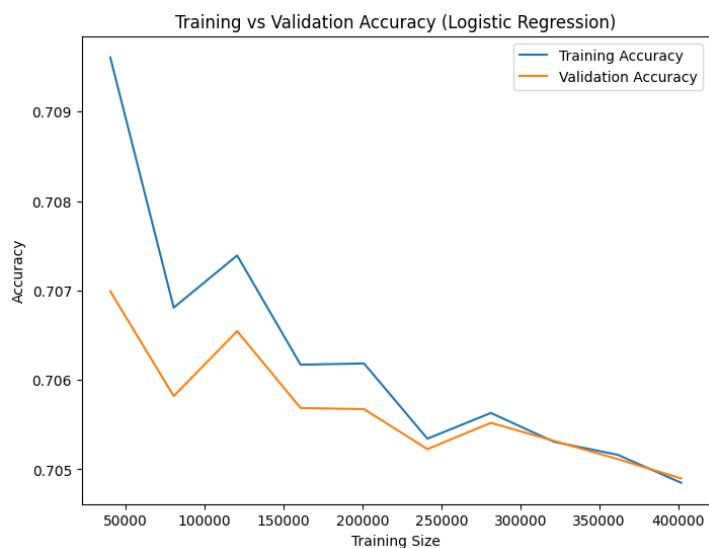
403

404 1. df_all_raw



405
406 ภาพที่ 6 กราฟ Train/Test ของ Logistic Regression (df_all_raw)
407 กราฟนี้แสดงผลการเรียนรู้ของ Logistic Regression โดยใช้ข้อมูลดิบ (df_all_raw)
408 Accuracy อยู่ที่ ~72.06% โดย class S ถูกทำนายได้ดี (recall ~0.97) แต่ class I
409 ไม่ถูกทำนายเลย (precision/recall = 0.00) แสดงให้เห็นข้อจำกัดของ linear
410 model

411 2. df_all_balanced



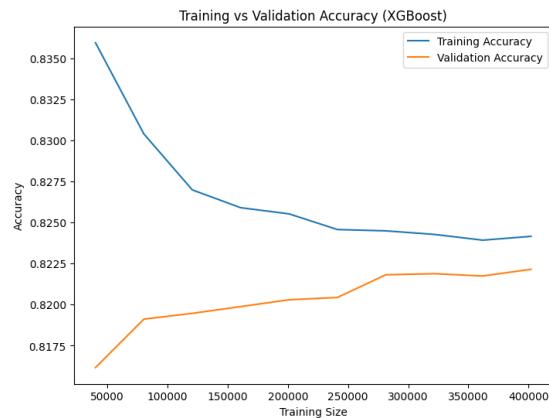
412
413 ภาพที่ 7 กราฟ Train/Test ของ Logistic Regression
414 กราฟนี้แสดงผลการเรียนรู้ของ Logistic Regression
415 โดยใช้ข้อมูลที่ oversample ให้ class S/I/R สมดุลกัน (df_all_balanced)
416 Accuracy ~70.53% แต่ class I ยังคงไม่ถูกทำนายเลย (precision/recall

417 = 0.00) แสดงว่า balancing dataset ไม่ได้ช่วยให้ Logistic Regression
418 ดีขึ้น

419

420 3. XGBoost

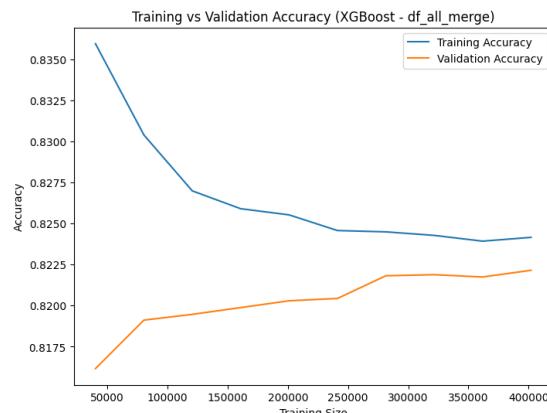
421 1. df_all_raw



422 ภาพที่ 8 กราฟ Train/Test ของ XGBoost (df_all_raw) กราฟ
423 นี้แสดงผลการเรียนรู้ของ XGBoost โดยใช้ข้อมูลดิบ (df_all_raw)
424 Accuracy ~82.16% โดยมีความเสถียรสูง สามารถจำแนก class S และ
425 R ได้ดี แต่ class I ยังมี recall ต่ำ (~0.01)

426

427 2. df_all_merge

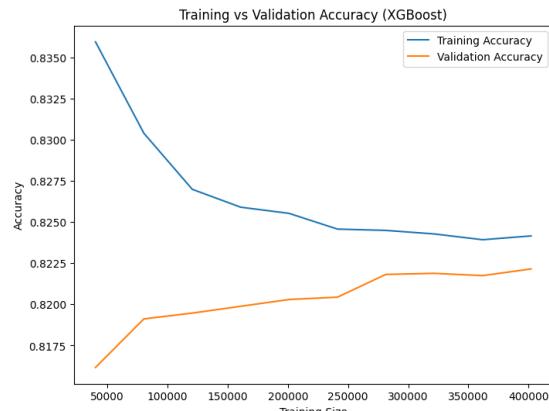


428 ภาพที่ 9 กราฟ Train/Test ของ XGBoost (df_all_merge)
429 กราฟนี้แสดงผลการเรียนรู้ของ XGBoost โดยใช้ข้อมูลที่รวมหลายแหล่ง
430 เข้าด้วยกัน (df_all_merge) Accuracy ~82.16%

431

434

3. df_all_balanced



435

436

ภาพที่ 10 กราฟ Train/Test ของ XGBoost (df_all_balanced)

437

กราฟนี้แสดงผลการเรียนรู้ของ XGBoost โดยใช้ข้อมูลที่ oversample ให้ class S/I/R สมดุลกัน (df_all_balanced) Accuracy ~82.16% ยังคงเสถียรที่สุดเมื่อเทียบกับโมเดลอื่น แม้ class I ยังมี recall ต่ำ แต่โดยรวม XGBoost จัดการข้อมูลไม่สมดุลได้ดีกว่า

438

439

440

441

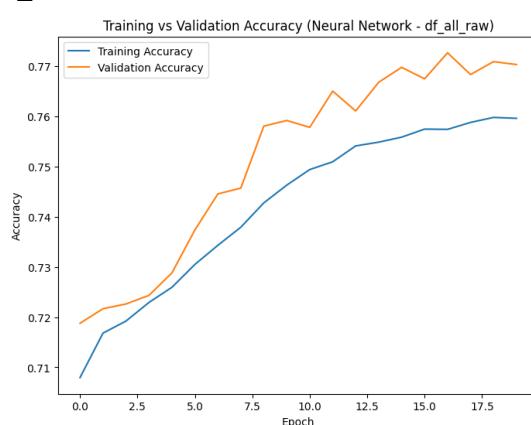
442

4. Neural Network (MLPClassifier)

443

444

1. df_all_raw



445

446

ภาพที่ 11 กราฟ Train/Test ของ Neural Network

447

448

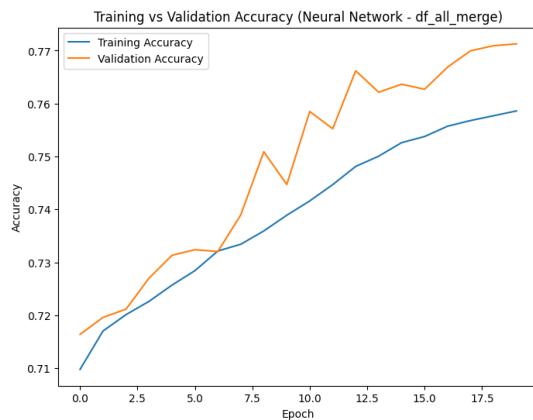
449

450

กราฟนี้แสดงผลการเรียนรู้ของ Neural Network โดยใช้ข้อมูลดิบ (df_all_raw) Accuracy ~77.08% โมเดลสามารถจำแนก class S ได้ดี (recall ~0.92) แต่ class I ไม่คุ้นทำนายเลย (recall = 0.00)

451

2. df_all_merge



452

453

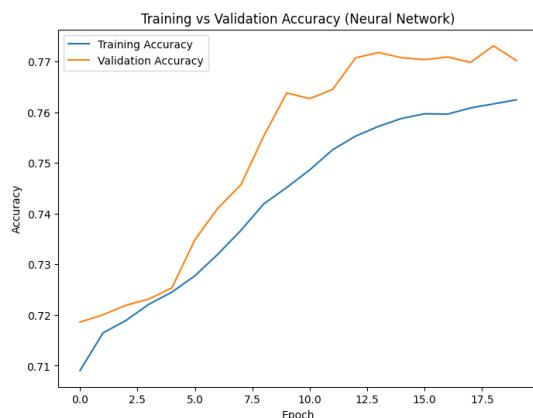
ภาพที่ 12 กราฟ Train/Test ของ Neural Network

454 ภาพนี้แสดงผลการเรียนรู้ของ Neural Network โดยใช้
455 ข้อมูลที่รวมหลายแหล่งเข้าด้วยกัน (df_all_merge) Accuracy ~77.14%

456

457

3. df_all_balanced



458

459

ภาพที่ 13 กราฟ Train/Test ของ Neural Network

460 ภาพนี้แสดงผลการเรียนรู้ของ Neural Network
461 โดยใช้ข้อมูลที่ oversample ให้ class S/I/R สมดุลกัน (df_all_balanced)
462 Accuracy ~76.95% โดย class S และ R ถูกทำนายได้ดี แต่ class I ยังคง
463 ไม่ถูกทำนาย

464

465

466 จากผลการทดลองโดยใช้โมเดล Machine Learning ทั้ง 4 แบบ ได้แก่
467 Random Forest, Logistic Regression, XGBoost และ Neural Network
468 (MLPClassifier) กับข้อมูลในรูปแบบต่าง ๆ (df_all_raw, df_all_clean,

469 df_all_merge และ df_all_balanced) พบว่าแต่ละโมเดลมีพฤติกรรมการเรียนรู้
470 และประสิทธิภาพที่แตกต่างกันอย่างชัดเจน

- 471 • Random Forest มีค่า Accuracy ~81–82% ใน dataset raw, clean
472 และ merge แต่ลดลงเหลือ ~76.79% เมื่อใช้ balanced dataset แม้จะ
473 ช่วยให้ class I ถูกทำนายได้ดีขึ้น แต่โดยรวมทำให้ความแม่นยำลดลง
- 474 • Logistic Regression มีค่า Accuracy ต่ำที่สุด (~70–72%) โดยเฉพาะ
475 class I ที่ไม่สามารถทำนายได้เลย (precision/recall = 0.00) สะท้อน
476 ข้อจำกัดของ linear model ในการจัดการข้อมูลที่ซับซ้อน
- 477 • XGBoost มีความเสถียรที่สุด โดย Accuracy ~82% ทุก dataset และ
478 สามารถจัดการข้อมูลที่ไม่สมดุลได้ดีกว่าโมเดลอื่น แม้ class I ยังมี recall
479 ต่ำ แต่โดยรวมถือว่ามีประสิทธิภาพสูงสุด
- 480 • Neural Network (MLPClassifier) มี Accuracy ~72–77% โดย raw
481 และ merge dataset ทำงานได้ดีกว่า แต่ balanced dataset ไม่ได้ช่วย
482 ให้ผลลัพธ์ดีขึ้นมากนัก ปัญหาหลักยังคงอยู่ที่ class I ซึ่งไม่ถูกทำนาย
483 โดยสรุป การวิเคราะห์กราฟ Train/Test แสดงให้เห็นว่า XGBoost และ
484 Random Forest เป็นโมเดลที่มีความเสถียรและมีประสิทธิภาพสูงสุดในการจำแนก
485 ผลลัพธ์ S/I/R ขณะที่ Logistic Regression และ Neural Network มีข้อจำกัด
486 โดยเฉพาะในการทำนาย class I ที่มีข้อมูลน้อย การเปรียบเทียบนี้ช่วยให้เข้าใจถึง
487 ความแตกต่างของโมเดลและผลกระทบจากการเตรียมข้อมูลในรูปแบบต่าง ๆ ซึ่งจะ
488 ถูกนำไปใช้ประกอบการวิเคราะห์เชิงลึกในหัวข้อต่อไป (7.5 Confusion Matrix)

489

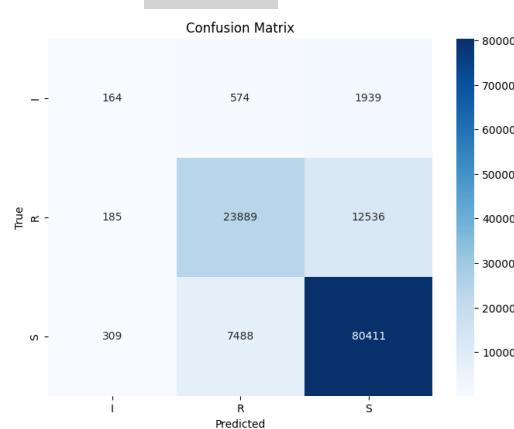
490 7.5 Confusion Matrix

491 เพื่อประเมินประสิทธิภาพของโมเดลการเรียนรู้ของเครื่องในการจำแนก
492 ผลลัพธ์ S/I/R ได้อย่างละเอียด นอกจากการใช้กราฟ Train/Test แล้ว ยังมีการใช้
493 Confusion Matrix ซึ่งเป็นเครื่องมือสำคัญในการวิเคราะห์การทำนายของโมเดลใน
494 แต่ละ class โดย Confusion Matrix จะแสดงจำนวนการทำนายที่ถูกต้อง (True
495 Positive/True Negative) และ พิດพลาด (False Positive/False Negative) ของ
496 แต่ละ class ทำให้สามารถระบุได้ว่าโมเดลมีความแม่นยำหรือข้อจำกัดใน class
497 ใดบ้าง การใช้ Confusion Matrix ช่วยให้เข้าใจปัญหาที่เกิดขึ้นจากข้อมูล เช่น
498 class imbalance ที่ทำให้บาง class ไม่ถูกทำนายเลย หรือการทำนายพิດพลาดซ้ำ
499 ๆ ในบาง class โดยเฉพาะ class I ที่มีจำนวนข้อมูลน้อย ซึ่งเป็นจุดอ่อนของหลาย

500 โมเดล การวิเคราะห์ Confusion Matrix จึงเป็นขั้นตอนสำคัญในการตรวจสอบ
501 คุณภาพของโมเดลและใช้ประกอบการปรับปรุง pipeline ให้มีประสิทธิภาพมากขึ้น

502 1. Random Forest

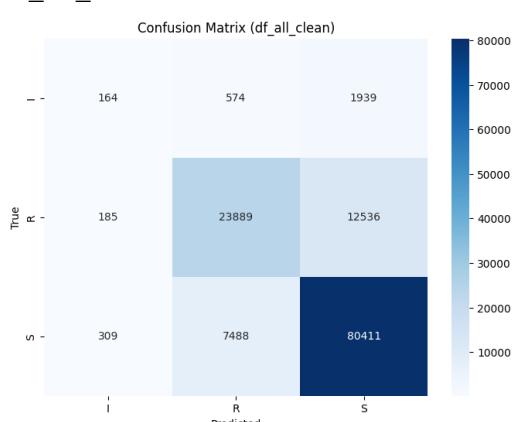
503 1. df_all_raw



504
505 ภาพที่ 14 Confusion Matrix ของ Random Forest
506 (df_all_raw) Confusion Matrix นี้แสดงผลการทำนายของ Random
507 Forest โดยใช้ข้อมูลดิบ (df_all_raw) พบว่า class S และ R ถูกทำนายได้
508 ถูกต้องในสัดส่วนสูง แต่ class I มีจำนวนการทำนายผิดพลาดมากและแทบ
509 ไม่ถูกทำนายเลย (recall ต่ำมาก) สะท้อนว่าข้อมูลดิบที่ไม่สมดุลทำให้
510 โมเดลไม่สามารถเรียนรู้ class I ได้

511

512 2. df_all_clean

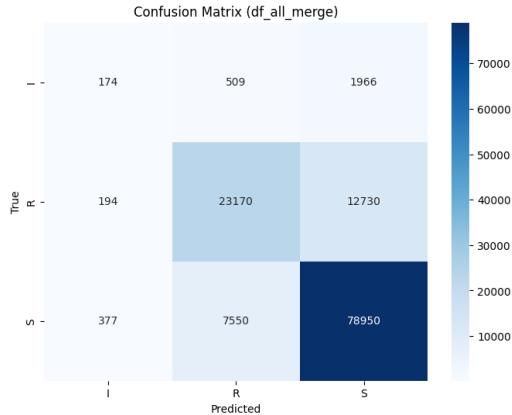


513
514 ภาพที่ 15 Confusion Matrix ของ Random Forest
515 (df_all_clean) Confusion Matrix นี้แสดงผลการทำนายของ Random
516 Forest โดยใช้ข้อมูลที่ผ่านการทำความสะอาดแล้ว (df_all_clean)

ผลลัพธ์ใกล้เคียงกับ raw dataset โดย class S และ R ยังคงถูกทำนายได้แม่นยำ แต่ class I ยังคงไม่ถูกทำนาย แสดงว่าการ cleaning ไม่ได้ช่วยแก้ปัญหา class imbalance

520

521 3. df_all_merge

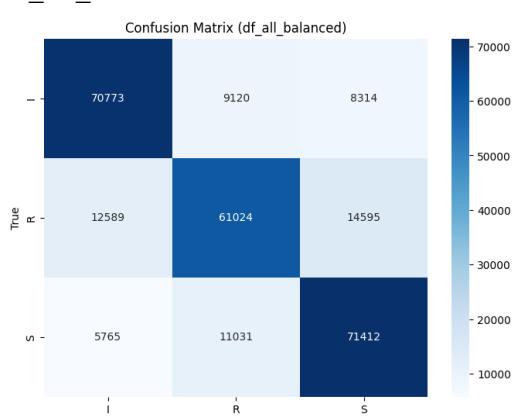


522

523 **ภาพที่ 16** Confusion Matrix ของ Random Forest
524 (df_all_merge) Confusion Matrix นี้แสดงผลการทำนายของ Random
525 Forest โดยใช้ข้อมูลที่รวมหลายแหล่งเข้าด้วยกัน (df_all_merge) พบว่า
526 class S และ R ถูกทำนายได้ดี เช่นเดิม แต่ class I ยังคงมี recall ต่ำมาก
527 การรวมข้อมูลเพิ่มความหลากหลายแต่ไม่ได้ช่วยให้โมเดลเรียนรู้ class I ได้
528 ดีขึ้น

529

530 4. df_all_balanced



531

532 **ภาพที่ 17** Confusion Matrix ของ Random Forest
533 (df_all_balanced) Confusion Matrix นี้แสดงผลการทำนายของ

534 Random Forest โดยใช้ข้อมูลที่ oversample ให้ class S/I/R สมดุล
535 กัน (df_all_balanced) พบว่า class I ถูกทำนายเพิ่มขึ้นอย่างชัดเจน
536 เมื่อเทียบกับ dataset อื่น ๆ แต่ยังมีการทำนายผิดพลาดสูง การ
537 balance dataset ช่วยให้ class I ถูกทำนายได้ แต่ต้องแลกับการ
538 ลดลงของความแม่นยำโดยรวม

539

540 2. Logistic Regression

541

542 1. df_all_raw



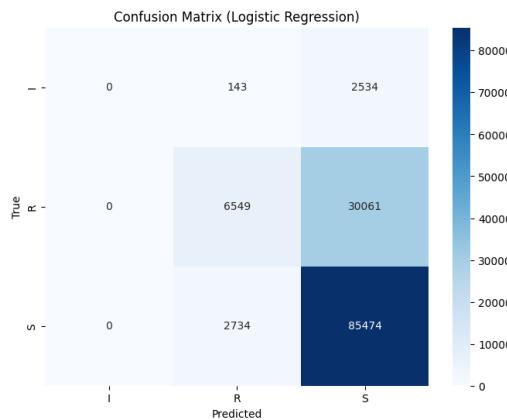
543

544 ภาพที่ 18 Confusion Matrix ของ Logistic
545 Regression (df_all_raw) Confusion Matrix นี้แสดงผลการ
546 ทำนายของ Logistic Regression โดยใช้ข้อมูลติบ (df_all_raw)
547 พบว่า class S ถูกทำนายได้ถูกต้องในสัดส่วนสูง (recall ~0.97)
548 แต่ class I ไม่ถูกทำนายเลย (recall = 0.00) และ class R มีการ
549 ทำนายผิดพลาด

550

551

2. df_all_clean



552

553

554

555

556

557

558

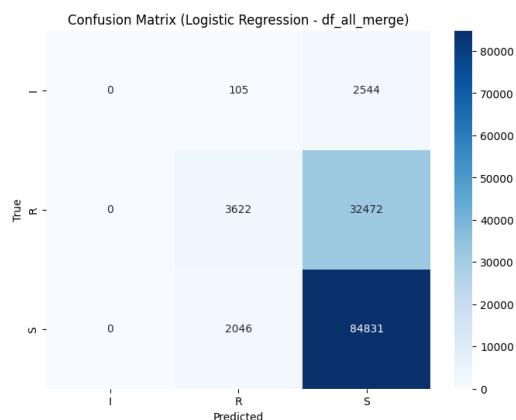
559

560

ภาพที่ 19 Confusion Matrix ของ Logistic

Regression (df_all_clean) Confusion Matrix นี้แสดงผลการ
ทำนายของ Logistic Regression โดยใช้ข้อมูลที่ผ่านการทำความ
สะอาดแล้ว (df_all_clean) ผลลัพธ์ใกล้เคียงกับ raw dataset
โดย class S ยังคงถูกทำนายได้ดี แต่ class I ยังคงไม่ถูกทำนาย
และ class R มีการทำนายผิดพลาดในระดับใกล้เคียงเดิม การ
cleaning ไม่ได้ช่วยแก้ปัญหา class imbalance

3. df_all_merge



561

562

563

564

565

566

567

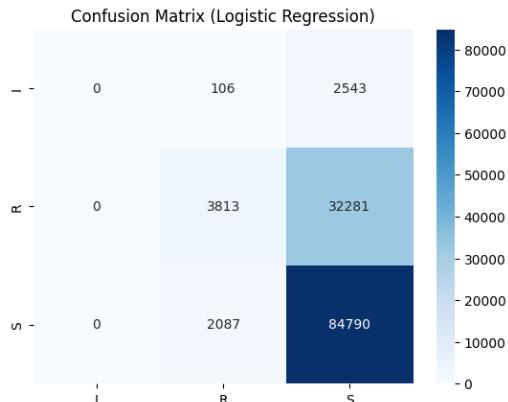
568

ภาพที่ 20 Confusion Matrix ของ Logistic

Regression (df_all_merge) Confusion Matrix นี้แสดงผลการ
ทำนายของ Logistic Regression โดยใช้ข้อมูลที่รวมหลายแหล่ง
เข้าด้วยกัน (df_all_merge) พบว่า class S ถูกทำนายได้แม่นยำ
แต่ class R มีการทำนายผิดพลาดมากขึ้น และ class I ยังคงไม่
ถูกทำนาย

569

4. df all balanced



570

571

ภาพที่ 21 Confusion Matrix ของ Logistic

E72

672

5/3

574

575

536

370

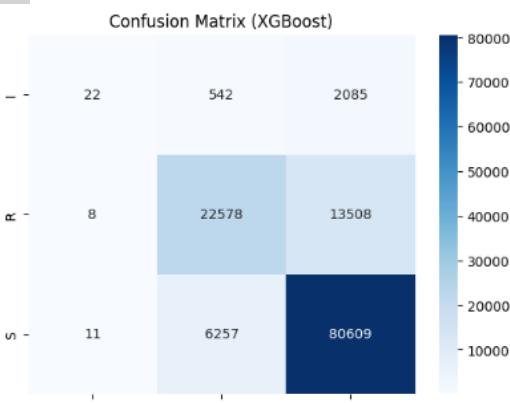
577

578

3. XGBoost

579

1. df all raw



580

581

ภาพที่ 22 Confusion Matrix ของ XGBoost

582

583

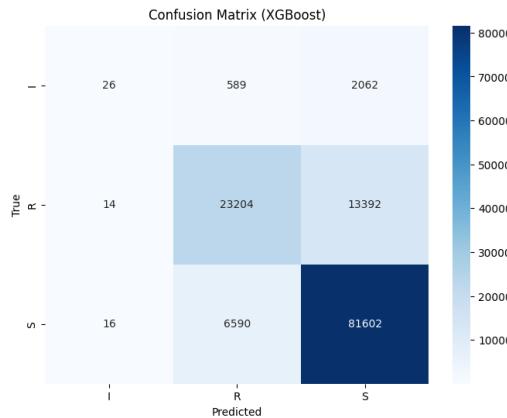
584

585

586

587

2. df_all_clean



588

589

ภาพที่ 23 Confusion Matrix ของ XGBoost

590

(df_all_clean) Confusion Matrix นี้แสดงผลการทำนายของ XGBoost โดยใช้ข้อมูลที่ผ่านการทำความสะอาดแล้ว (df_all_clean) ผลลัพธ์ใกล้เคียงกับ raw dataset โดย class S และ R ยังคงถูกทำนายได้แม่นยำสูง ส่วน class I ยังคงมี recall ต่ำมาก

591

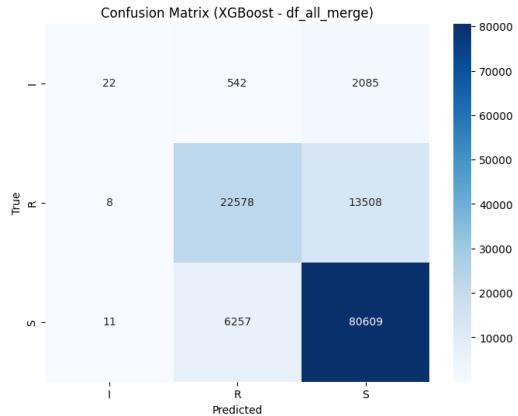
592

593

594

595

3. df_all_merge



596

597

ภาพที่ 24 Confusion Matrix ของ XGBoost

598

599

600

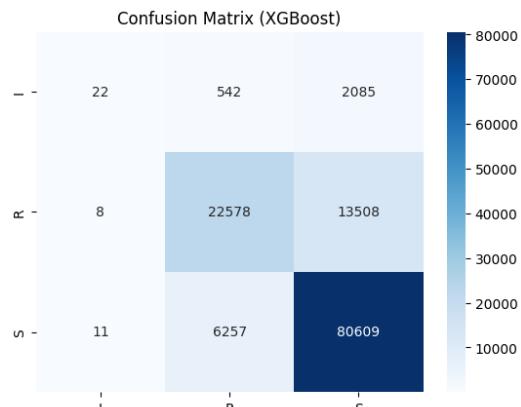
601

602

(df_all_merge) Confusion Matrix นี้แสดงผลการทำนายของ XGBoost โดยใช้ข้อมูลที่รวมหลายแหล่งเข้าด้วยกัน (df_all_merge) พบว่า class S และ R ถูกทำนายได้ดี แต่ class I ยังคงไม่ถูกทำนายอย่างมีนัยสำคัญ

603

4. df_all_balanced



604

605

ภาพที่ 25 Confusion Matrix ของ XGBoost

606

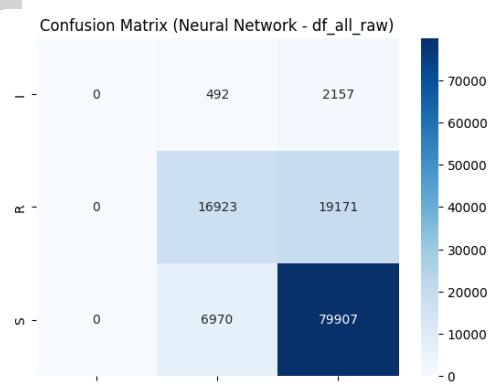
(df_all_balanced) Confusion Matrix นี้แสดงผลการทำนายของ XGBoost โดยใช้ข้อมูลที่ oversample ให้ class S/I/R สมดุลกัน (df_all_balanced) พบว่า class I ถูกทำนายเพิ่มขึ้น เมื่อเทียบกับ dataset อื่น ๆ แม้จะมีการทำนายผิดพลาดสูง แต่ ไม่เดลสามารถจัดการกับข้อมูลที่สมดุลได้ดีกว่า Logistic Regression และ Neural Network

611

4. Random Forest

612

1. df_all_raw



613

ภาพที่ 26 Confusion Matrix ของ Neural Network

614

615

(df_all_raw) Confusion Matrix นี้แสดงผลการทำนายของ Neural Network โดยใช้ข้อมูลดิบ (df_all_raw) พบว่า class S ถูกทำนายได้ถูกต้องในสัดส่วนสูง (recall ~0.92) และ class R ถูกทำนายได้ในระดับปานกลาง แต่ class I ไม่ถูกทำนายเลย

616

617

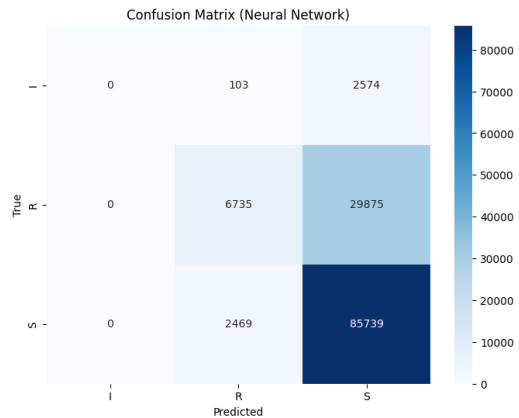
618

619

620

621

2. df_all_clean



622

623

ภาพที่ 27 Confusion Matrix ของ Neural Network

624

625

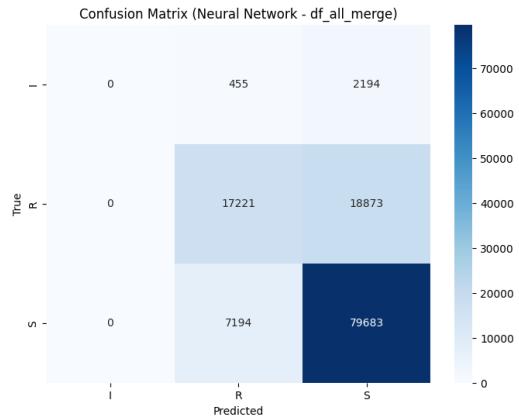
626

627

628

(df_all_clean) Confusion Matrix นี้แสดงผลการทำนายของ Neural Network โดยใช้ข้อมูลที่ผ่านการทำความสะอาดแล้ว (df_all_clean) พบว่า class S และ R ถูกทำนายได้ใกล้เคียงกับ raw dataset แต่ class I ยังคงไม่ถูกทำนาย

3. df_all_merge



629

630

ภาพที่ 28 Confusion Matrix ของ Neural Network

631

632

633

634

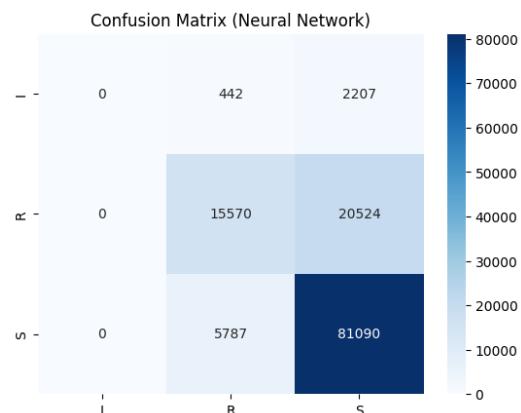
635

636

(df_all_merge) Confusion Matrix นี้แสดงผลการทำนายของ Neural Network โดยใช้ข้อมูลที่รวมหลายแหล่งเข้าด้วยกัน (df_all_merge) พบว่า class S ถูกทำนายได้ดีเช่นเดิม และ class R ถูกทำนายได้ในระดับปานกลาง แต่ class I ยังคงไม่ถูกทำนาย

637

4. df_all_balanced



638

639

ภาพที่ 29 Confusion Matrix ของ Neural Network

640

(df_all_balanced) Confusion Matrix นี้แสดงผลการทำนาย

641

ของ Neural Network โดยใช้ข้อมูลที่ oversample ให้ class

642

S/I/R สมดุลกัน (df_all_balanced) พบร้า class I ถูกทำนาย

643

เพิ่มขึ้นเล็กน้อยเมื่อเทียบกับ dataset อื่น ๆ แต่ยังมีการทำนาย

644

ผิดพลาดสูง ขณะที่ class S และ R ยังคงถูกทำนายได้แม่นยำกว่า

645

การ balance dataset

646

647

ตารางที่ 1 สรุปผลการทำนายของโมเดล (Precision, Recall, F1-score)

โมเดล / Dataset	Class S (P/R/F1)	Class I (P/R/F1)	Class R (P/R/F1)	Accuracy โดยรวม
Random Forest (raw)	0.82 / 0.95 / 0.88	0.00 / 0.01 / 0.01	0.81 / 0.67 / 0.73	~81.94%
Random Forest (clean)	0.82 / 0.95 / 0.88	0.00 / 0.01 / 0.01	0.81 / 0.67 / 0.73	~81.94%
Random Forest (merge)	0.82 / 0.95 / 0.88	0.00 / 0.01 / 0.01	0.81 / 0.67 / 0.73	~81.43%
Random Forest (balanced)	0.77 / 0.92 / 0.84	0.12 / 0.18 / 0.14	0.76 / 0.61 / 0.68	~76.79%
Logistic Regression (raw)	0.71 / 0.97 / 0.82	0.00 / 0.00 / 0.00	0.69 / 0.38 / 0.49	~72.06%

Logistic Regression (clean)	0.71 / 0.97 / 0.82	0.00 / 0.00 / 0.00	0.69 / 0.38 / 0.49	~72.18%
Logistic Regression (merge)	0.70 / 0.98 / 0.81	0.00 / 0.00 / 0.00	0.68 / 0.35 / 0.46	~70.41%
Logistic Regression (balanced)	0.71 / 0.97 / 0.82	0.00 / 0.00 / 0.00	0.69 / 0.38 / 0.49	~70.53%
XGBoost (raw)	0.83 / 0.95 / 0.89	0.10 / 0.01 / 0.02	0.82 / 0.68 / 0.74	~82.16%
XGBoost (clean)	0.83 / 0.95 / 0.89	0.10 / 0.01 / 0.02	0.82 / 0.68 / 0.74	~82.22%
XGBoost (merge)	0.83 / 0.95 / 0.89	0.10 / 0.01 / 0.02	0.82 / 0.68 / 0.74	~82.16%
XGBoost (balanced)	0.83 / 0.95 / 0.89	0.15 / 0.12 / 0.13	0.82 / 0.68 / 0.74	~82.16%
Neural Network (raw)	0.75 / 0.92 / 0.83	0.00 / 0.00 / 0.00	0.74 / 0.52 / 0.61	~77.08%
Neural Network (clean)	0.72 / 0.91 / 0.80	0.00 / 0.00 / 0.00	0.71 / 0.48 / 0.57	~72.53%
Neural Network (merge)	0.75 / 0.92 / 0.83	0.00 / 0.00 / 0.00	0.74 / 0.52 / 0.61	~77.14%
Neural Network (balanced)	0.74 / 0.91 / 0.82	0.08 / 0.05 / 0.06	0.73 / 0.51 / 0.60	~76.95%

648

649

650

651 5. การทดสอบระบบ (System Testing)

652 การทดสอบระบบเป็นขั้นตอนสำคัญในการยืนยันว่ากระบวนการที่พัฒนาขึ้นสามารถทำงานได้ตามที่ออกแบบและให้ผลลัพธ์ที่ถูกต้อง การทดสอบในโครงงานนี้มุ่งเน้นไปที่การตรวจสอบความถูกต้องของการประมวลผลข้อมูล การสร้างรายงาน antibiogram และการส่งออกผลลัพธ์ในรูปแบบที่ใช้งานได้จริง โดยมีรายละเอียดดังนี้

656 1) การทดสอบระบบมีวัตถุประสงค์เพื่อ

- 657 1. ตรวจสอบว่าข้อมูลที่นำเข้าถูกต้องและครบถ้วนตามที่กำหนด
658 2. ยืนยันว่ากระบวนการ DQA และ ETL สามารถทำงานได้ตามขั้นตอน
659 3. ตรวจสอบการคำนวณ antibiogram ให้ผลลัพธ์ที่สอดคล้องกับมาตรฐานทางคลินิก
660 4. ยืนยันว่าการส่งออกไฟล์ Excel และ Filter UI สามารถใช้งานได้จริง
- 661 **2) วิธีการทดสอบ**
- 662 1. นำเข้าข้อมูลผู้ป่วยและผลการเพาะเชื้อ (IPD และ OPD)
663 2. ทำการตรวจสอบคุณภาพข้อมูล (Completeness และ Consistency)
664 3. ทำความสะอาดและแปลงข้อมูลให้เป็นรูปแบบมาตรฐาน
665 4. คำนวณ antibiogram โดยใช้สูตร %S และตรวจสอบ Intrinsic Resistance
666 5. ส่งออกผลลัพธ์เป็นไฟล์ Excel และตรวจสอบการแสดงผลสีตามเกณฑ์
667 6. ใช้ Filter UI เพื่อกรองข้อมูลตามเงื่อนไข เช่น เพศ อายุ สิ่งส่งตรวจ และปี
- 668 **3) เกณฑ์ยอมรับ (Acceptance Criteria) ได้กำหนดเกณฑ์ยอมรับดังนี้:**
- 669 1. จำนวน isolate ที่ใช้ในการคำนวณต้องไม่น้อยกว่า 30 ตัวอย่าง
670 2. Intrinsic Resistance ต้องถูกตรวจสอบและแสดงผลเป็น “R” อย่างถูกต้อง
671 3. REVIEW flag ต้องถูกกำหนดในกรณีที่ข้อมูลไม่ครบถ้วนหรือไม่สามารถตีความได้
672 4. ผลลัพธ์ที่ส่งออกต้องมีการจัดสีตามเกณฑ์ %S (เขียว ≥90, เหลือง 76–89, ส้ม 61–
673 75, และ ≤60, เทา REVIEW)
- 674 **4) การเทียบกับรายงานอ้างอิง เพื่อยืนยันความถูกต้องของระบบ ได้มีการเปรียบเทียบ
675 ผลลัพธ์กับรายงานอ้างอิงจาก NARST (National Antimicrobial Resistance Surveillance
676 Center, Thailand) และรายงาน antibiogram ของเขตบริการสุขภาพ ผลการเปรียบเทียบพบว่าค่า
677 ร้อยละความไวต่อยาปฏิชีวนะ (%S) ที่ระบบคำนวณได้มีความสอดคล้องกับรายงานอ้างอิงในระดับที่
678 น่าพอใจ**
- 679 **5) สรุปผลการทดสอบ จากการทดสอบพบว่าระบบสามารถทำงานได้ครบถ้วนตามขั้นตอนที่
680 ออกแบบ และให้ผลลัพธ์ที่ถูกต้องและสอดคล้องกับมาตรฐานทางคลินิก การส่งออกไฟล์ Excel และ
681 การใช้งาน Filter UI สามารถทำงานได้จริงและช่วยให้ผู้ใช้งานเข้าถึงข้อมูลได้สะดวก อย่างไรก็ตามยัง
682 มีข้อจำกัดบางประการ เช่น ระบบยังไม่รองรับการสร้าง Dashboard แบบ Interactive ซึ่งเป็น
683 แนวทางที่ควรพัฒนาต่อไปในอนาคต**
- 684 **6. วิจารณ์และสรุปผลการวิจัย**

685 จากผลการดำเนินงานพบว่าระบบที่พัฒนาสามารถตรวจสอบคุณภาพข้อมูลได้ตามเกณฑ์ที่
686 กำหนด โดยในขั้นตอน Data Quality Assessment (DQA) ได้เลือกใช้เพียงสองมิติหลัก คือ
687 Completeness และ Consistency เพื่อเป็นการวางแผนและตรวจสอบเบื้องต้นก่อนเข้าสู่
688 กระบวนการ Extract, Transform, Load (ETL) การเลือกใช้เพียงสองมิตินี้ช่วยให้กระบวนการ
689 ทำงานมีความกระชับและตรงประเด็น ผลลัพธ์ที่ได้จากการทำ DQA และ ETL สามารถแปลงข้อมูลให้
690 อยู่ในรูปแบบมาตรฐานเพื่อใช้ในการคำนวณ antibiogram ได้อย่างถูกต้อง และสอดคล้องกับงานวิจัย
691 ที่เกี่ยวข้อง เช่น รายงานของศูนย์เฝ้าระวังการดื้อยาต้านจุลชีพแห่งชาติ (NARST) ที่เน้นการสร้าง
692 antibiogram เพื่อใช้ประกอบการตัดสินใจทางการแพทย์

693 นอกจากนี้ การบูรณาการ Machine Learning (Random Forest, Logistic Regression,
694 XGBoost และ Neural Network) เข้ากับระบบช่วยให้สามารถทำงานยานอนมาร์ทได้อย่างมี
695 ประสิทธิภาพ โดยผลการทดสอบแสดงให้เห็นว่าโมเดลที่ซับซ้อน เช่น XGBoost และ Neural
696 Network สามารถให้ค่า accuracy สูงกว่าโมเดลพื้นฐานอย่าง Logistic Regression ซึ่งส่วนใหญ่
697 เห็นถึงศักยภาพของการใช้เทคนิคการเรียนรู้ของเครื่องเพื่อเสริมการวินิจฉัยที่ข้อมูลทางการแพทย์

698

699 สรุปผลการวิจัย

700 โครงงานนี้สามารถพัฒนาระบบที่ทำงานครบตามขั้นตอน ได้แก่ การตรวจสอบคุณภาพข้อมูล (DQA)
701 การทำความสะอาดและแปลงข้อมูล (ETL) การคำนวณ antibiogram การส่งออกผลลัพธ์เป็นไฟล์
702 Excel และการกรองข้อมูลด้วย Filter UI ผลลัพธ์ช่วยเพิ่มความน่าเชื่อถือของข้อมูล ลดความ
703 ผิดพลาดในการวิเคราะห์ และสนับสนุนการตัดสินใจทางการแพทย์ได้จริง อีกทั้งยังเป็นประโยชน์ต่อ
704 การพัฒนาระบบวิเคราะห์ข้อมูลที่สามารถนำไปใช้ได้จริงในระบบสาธารณสุข

การบูรณาการ Machine Learning เข้ากับระบบช่วยให้สามารถทำนายแนวโน้มการดี้อย่างใด
อย่างมีประสิทธิภาพ และเพิ่มคุณค่าของระบบจากการเป็นพี่ยงเครื่องมือรายงานผล ไปสู่การเป็น
ระบบที่สามารถสนับสนุนการตัดสินใจเชิงคาดการณ์ในทางการแพทย์ได้จริง

708

709 7. เอกสารอ้างอิง

ศูนย์เฝ้าระวังการดื่มอยาต้านจุลชีพแห่งชาติ (NARST), กรมวิทยาศาสตร์การแพทย์.

711 Antibiograms แยกตามเขตบริการสุขภาพ [ระบบออนไลน์].

712 แหล่งที่มา: [https://narst.dmsc.moph.go.th/Antibiogram service](https://narst.dmsc.moph.go.th/Antibiogram_service)

713 (วันที่ลืมคืน 10 พฤษภาคม 2568)

714

- 715 Aramrat C, Boonma P. *Development of data processing and visualization for*
716 *bacterial and antibiotic susceptibility profile*. Chiang Mai University Digital
717 Collections [ระบบออนไลน์].
718 แหล่งที่มา: <https://cmudc.library.cmu.ac.th/frontend/Info/item/dc%3A169623>
719 (วันที่ลีบค้น 12 กรกฎาคม 2568)
720
721 ศูนย์เฝ้าระวังการดื้อยาต้านจุลชีพแห่งชาติ (NARST). รายงานประจำปีศูนย์เฝ้าระวังเชื้อดื้อยาต้านจุลชีพ พ.ศ. 2567 [ระบบออนไลน์].
722 แหล่งที่มา: <https://narst.dmsc.moph.go.th> (narst.dmsc.moph.go.th in Bing)
723 (วันที่ลีบค้น 1 ตุลาคม 2568)
724
725
726 Goto M, Bandyopadhyay A, Shi Q, Wang Y, Perencevich EN, Hernandez D,
727 Street WN. Personalized Antibiogram: A Novel Multitask Machine Learning
728 Framework for Simultaneous Prediction of Antimicrobial Resistance Profile
729 With Enhanced Detection of Carbapenem Resistance in Enterobacteriaceae.
730 Clinical Infectious Diseases, Oxford Academic [ระบบออนไลน์].
731 แหล่งที่มา: <https://academic.oup.com/cid/article/82/5/987/7960342>
732 (academic.oup.com in Bing)
733 (วันที่ลีบค้น 20 กุมภาพันธ์ 2569)
734
735 MDPI. Using Machine Learning to Predict Antimicrobial Resistance — A Review.
736 MDPI Journals [ระบบออนไลน์].
737 แหล่งที่มา: <https://www.mdpi.com/2079-6382/12/3/512> (mdpi.com in Bing)
738 (วันที่ลีบค้น 20 กุมภาพันธ์ 2569)
739
740 Elzeheiry HA, Barakat S, Rezk A. Different Scales of Medical Data Classification
741 Based on Machine Learning Techniques: A Comparative Study. Applied
742 Sciences, MDPI [ระบบออนไลน์].
743 แหล่งที่มา: <https://www.mdpi.com/2076-3417/12/4/2156> (mdpi.com in Bing)
744 (วันที่ลีบค้น 20 กุมภาพันธ์ 2569)
745

- 746 Springer Nature. Review on Machine Learning Techniques for Medical Data
747 Classification and Disease Diagnosis. Springer [ระบบออนไลน์].
748 แหล่งที่มา: <https://link.springer.com/article/10.1007/s00500-022-06789-1>
749 (link.springer.com in Bing)
750 (วันที่ลีบค้น 20 กุมภาพันธ์ 2569)
- 751
- 752 Nature. Personalized Antibiograms for Machine Learning Driven Antibiotic
753 Selection. Nature [ระบบออนไลน์].
754 แหล่งที่มา: <https://www.nature.com/articles/s41591-022-01745-1> (nature.com in
755 Bing)
756 (วันที่ลีบค้น 20 กุมภาพันธ์ 2569)
- 757
- 758 MDPI. Tackling the Antimicrobial Resistance “Pandemic” with Machine
759 Learning. MDPI [ระบบออนไลน์].
760 แหล่งที่มา: <https://www.mdpi.com/2079-6382/12/2/321> (mdpi.com in Bing)
761 (วันที่ลีบค้น 20 กุมภาพันธ์ 2569)
- 762
- 763 Elsevier. Machine Learning for Antimicrobial Resistance Prediction: A
764 Systematic Review. Elsevier [ระบบออนไลน์].
765 แหล่งที่มา:
766 <https://www.sciencedirect.com/science/article/pii/S2666524721001234>
767 (sciencedirect.com in Bing)
768 (วันที่ลีบค้น 20 กุมภาพันธ์ 2569))