การจำแนกประเภทความผิดปกติของเครื่องจักรจากข้อมูลการผลิต

CLASSIFICATION OF MACHINE MALFUNCTIONS FROM PRODUCTION LINE DATA

อารีรัตน์ ชื่นชม

ARREERAT CHUENCHOM

การค้นคว้าอิสระนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิชาวิทยาการข้อมูลและการวิเคราะห์

ศูนย์วิเคราะห์ข้อมูลดิจิทัลอัจฉริยะพระจอมเกล้าลาดกระบัง คณะวิทยาศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2567

KMITL-2024-SC-M-017-041

CLASSIFICATION OF MACHINE MULFUNCTIONS FROM PRODUCTION LINE DATA

ARREERAT CHUENCHOM

AN INDIVIDUAL STUDY SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENT FOR THE DEGREE OF MASTER OF SCIENCE IN
DATA SCIENCE AND ANALYTICS
KMITL DIGITAL ANALYTICS AND INTELLIGENCE CENTER SCHOOL OF SCIENCE
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
2024

KMITL-2024-SC-M-017-041

หัวข้อการค้นคว้าอิสระ การจำแนกประเภทความผิดปกติของเครื่องจักรจากข้อมูลการผลิต
ชื่อนักศึกษา นางสาวอารีรัตน์ ชื่นชม
รหัสประจำตัว 63605090
ปริญญา วิทยาศาสตรมหาบัณฑิต (วิทยาการข้อมูลและการวิเคราะห์)
ศูนย์วิเคราะห์ข้อมูลดิจิทัลอัจฉริยะพระจอมเกล้าลาดกระบัง
พ.ศ. 2567
อาจารย์ที่ปรึกษาการค้นคว้าอิสระ ผู้ช่วยศาสตราจารย์ ดร.วรางคณา กิ้มปาน

บทคัดย่อ

ปัญหาเครื่องจักรทำงานผิดปกติระหว่างทำการผลิต เป็นปัญหาหลักที่ส่งผลทำให้ประสิทธิภาพกำลังในการผลิตและปริมาณการผลิตที่ลดลง ก่อให้เกิดความเสียหายในการผลิตและในเชิงธุรกิจ ด้วยเหตุนี้ทำให้ทางโรงงานอุตสาหกรรมแต่ละแห่งมีความสนใจนำเทคโนโลยีดิจิทัลเข้ามาแก้ไขปัญหาที่เกิดขึ้น ในการค้นคว้าอิสระนี้มีวัตถุประสงค์เพื่อศึกษาเทคนิคการนำโมเดลอัลกอริทึมของการเรียนรู้ของเครื่อง มาปรับใช้ในการแก้ไขปัญหาการทำงานของเครื่องจักรที่เกิดขึ้น และพัฒนาแบบจำลองการตรวจสอบความผิดปกติการทำงานของเครื่องจักรสำหรับข้อมูลที่ไม่สมดุล โดยใช้เทคนิคการสุ่มเพิ่มข้อมูล มาวิเคราะห์และจำแนกสถานะความผิดปกติของเครื่องจักร และเปรียบเทียบประสิทธิภาพของวิธีการจำแนกกลุ่มจากชุดข้อมูลตัวอย่างของเครื่องจักรที่รวบรวมไว้ โดยเลือกใช้อัลกอริทึมกราเดี้ยนบูทติ้งแมชชีน วิธีการค้นหาเพื่อนบ้านใกล้ที่สุด k ตัว และอัลกอริทึมแรนดอมฟอเรส จากการเปรียบเทียบประสิทธิภาพของวิธีการจำแนกกลุ่มจากชุดข้อมูลตัวอย่าง โดยเปรียบเทียบจากค่าความถูกต้อง วิธีการจำแนกกลุ่มที่มีประสิทธิภาพการทำนายความผิดปกติของเครื่องจักรที่ดีที่สุดคือ อัลกอริทึมกราเดี้ยนบูทติ้งแมชชีน ซึ่งมีค่าความแม่นยำเท่ากับ 93.30% อันดับที่สองคือ อัลกอริทึมแรนดอมฟอเรส มีความแม่นยำเท่ากับ 81.03% และอัลกอริทึมที่มีค่าแม่นยำต่ำที่สุดคือ วิธีการค้นหาเพื่อนบ้านใกล้ที่สุด k ตัว มีค่าแม่นยำเท่ากับ 72.17%

คำสำคัญ: การจำแนกความผิดปกติของเครื่องจักร ข้อมูลไม่สมดุล วิธีการค้นหาเพื่อนบ้านใกล้ที่สุด k ตัว อัลกอริทึมกราเดี้ยนบูทติ้งแมชชีน อัลกอริทึมแรนดอมฟอเรส

Individual Study Title	Classification of Machine Malfunctions from Production Line Data
Student Name	Miss Arreerat Chuenchom
Student ID	63605090
Degree	Master of Science (Data science and Analytics) KMITL Digital Analytics and Intelligence Center
Year	2024
Individual Study Advisor	Asst.Prof.Dr. Warangkhana Kimpan

ABTRACT

Machinery malfunctioning during production problems is the main problem that results in the efficiency of production capacity and production volume decreasing which causes damage in production and in business. For this reason, each industrial factory is interested in using digital technology to solve problems. In this independent study, the objective is to study techniques for applying machine learning algorithm models to solve the operation of machines problems and develop a machine fault detection model for unbalanced data. The machine's abnormal conditions were analyzed and categorized using the random data addition approach. Then, the performance of the classification models was compared from the sample datasets collected by the machines in production. The models are Gradient Boosting Machine algorithm, k-nearest Neighbor algorithm, and Random Forest algorithm, used to compare the performance in accuracy of the classification models from the sample dataset. The best classification model in predicting machine malfunctions is Gradient Boosting Machine algorithm which has an average accuracy of 93.30% compared to Random Forest algorithm which has accuracy of 81.03% and k-Nearest Neighbors algorithm which has an average accuracy of 72.17%.

Keywords: Classification of machine malfunctions, Imbalanced Data, k-Nearest Neighbors Method, Gradient Boosting Machine Algorithm, Random Forest Algorithm.

กิตติกรรมประกาศ

ในการศึกษาค้นคว้าอิสระหัวข้อเรื่อง “การจำแนกประเภทความผิดปกติของเครื่องจักรจากข้อมูลการผลิต” ฉบับนี้ สามารถดำเนินการจนประสบความสำเร็จลุล่วงไปด้วยดี ด้วยความกรุณาอย่างยิ่งจากผู้ช่วยศาสตราจารย์ ดร.วรางคณา กิ้มปาน อาจารย์ที่ปรึกษาการศึกษาค้นคว้าอิสระ ที่ได้เสียสละเวลาอันมีค่าในการให้คำปรึกษา แนะนำ ผลักดัน ตรวจสอบความถูกต้องตลอดจนปรับปรุงในเนื้อหา และคำแนะนำสำหรับการแก้ไขข้อบกพร่องต่าง ๆ พร้อมทั้งเสนอแนวคิดให้ความรู้อันเป็นประโยชน์ ทำให้ผลงานการค้นคว้าอิสระฉบับนี้สำเร็จตรงตามเป้าหมายอย่างมีประสิทธิภาพ รวมถึงข้าพเจ้าได้รับความอนุเคราะห์จากอาจารย์หลากหลายท่าน โดยเฉพาะอย่างยิ่งคณาจารย์ประจำหลักสูตรวิทยาการข้อมูลและการวิเคราะห์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ได้ถ่ายทอดความรู้ให้แก่ข้าพเจ้า คอยให้กำลังใจทำให้งานวิจัยนี้สำเร็จลุล่วงตามวัตถุประสงค์

ขอขอบพระคุณคุณพ่อคุณแม่และครอบครัว รวมถึงเพื่อน ๆ ทุกคนซึ่งเป็นกำลังใจที่สำคัญในการพัฒนาตนเอง รวมถึงให้การสนับสนุนและความช่วยเหลือทุกอย่างด้วยดีเสมอมา จนทำให้ข้าพเจ้าสามารถดำเนินงานจนสำเร็จผ่านไปได้ด้วยดี

นางสาว อารีรัตน์ ชื่นชม

สารบัญ

สารบัญ (ต่อ)

หน้า

สารบัญ (ต่อ)

หน้า

สารบัญตาราง

ตารางที่ **หน้า**

สารบัญรูป

สารบัญรูป (ต่อ)

รูปที่ **หน้า**

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันแต่ละอุตสาหกรรมการผลิตได้ให้ความสำคัญเกี่ยวกับเทคโนโลยีดิจิทัล เพื่อนำมาเป็นตัวช่วยในการเพิ่มประสิทธิภาพในการผลิต ทำให้อุตสาหกรรมการผลิตทำงานได้อย่างรวดเร็ว และยืดหยุ่น รวมถึงส่งเสริมกำลังในการผลิตและกระจายสินค้าได้อย่างมีประสิทธิภาพมากขึ้น โดยถือว่าเป็นการปฏิวัติอุตสาหกรรมในรูปแบบหนึ่ง หลักแนวคิดในการเปลี่ยนแปลงวิธีการผลิตและระบบการผลิตนี้มีวัตถุประสงค์หลักคือ ต้องการเพิ่มประสิทธิภาพและทำให้ผลผลิตมีปริมาณเพิ่มขึ้น เพื่อให้สามารถกระจายสินค้าและบริการไปให้ทั่วถึง ทำให้หน่วยงานของแต่ละอุตสาหกรรมการผลิตได้พยายามคิดค้นหาวิธีการแก้ไข และตอบสนองเพื่อเพิ่มประสิทธิภาพและกำลังการผลิตมากยิ่งขึ้น ตัวอย่างเช่น การปรับเปลี่ยนวิธีการผลิตสินค้าของโรงงานให้มีลักษณะอุตสาหกรรมอัตโนมัติ (Industrial Automation) โดยการใช้เครื่องจักรหรือหุ่นยนต์ทดแทนแรงงานคน อีกทั้งยังได้นำเทคโนโลยีสารสนเทศและคอมพิวเตอร์มาสร้างเครือข่ายในการเชื่อมต่อระบบของกระบวนการผลิตในรูปแบบอินเทอร์เน็ตของสรรพสิ่ง (Internet of Things) ทำให้กระบวนการผลิตสินค้าเชื่อมกับเทคโนโลยีดิจิทัลมากยิ่งขึ้น รวมไปถึงการนำระบบปัญญาประดิษฐ์เข้ามาใช้ร่วมกับระบบแรงงานคนในกระบวนการผลิต ทำให้ระบบการผลิตมีประสิทธิภาพดีขึ้น สามารถรองรับปริมาณความต้องการทางการตลาดและวัตถุดิบ รวมถึงช่วยลดค่าใช้จ่ายในด้านแรงงานและพลังงานได้มากขึ้นด้วย

ในส่วนของปัญหาที่เกิดขึ้นในกระบวนการผลิตคือ กำลังในการผลิตที่ต่ำ ทำให้ยอดผลผลิตต่ำหรือได้ปริมาณน้อยกว่าแผนการที่วางไว้ ด้วยเหตุนี้จึงนำไปสู่วิธีการค้นหาแผนการวิเคราะห์ข้อมูลขนาดใหญ่เพื่อเพิ่มกำลังในการผลิต โดยนำเทคโนโลยีการวิเคราะห์ข้อมูลขนาดใหญ่มาแก้ไขปัญหาที่เกิดขึ้น เพื่อให้ได้ตามวัตถุประสงค์หลักขององค์กรที่ตั้งไว้

1.2 วัตถุประสงค์ของงานวิจัย

1) เพื่อศึกษาเทคนิคการนำโมเดลอัลกอริทึมของการเรียนรู้ของเครื่อง มาปรับใช้ในการแก้ไขปัญหาการทำงานของเครื่องจักร โดยใช้โมเดลอัลกอริทึมทั้งหมด 3 แบบ ได้แก่ วิธีการค้นหาเพื่อนบ้านใกล้ที่สุด k ตัว อัลกอริทึมกราเดี้ยนบูทติ้งแมชชีน อัลกอริทึมแรนดอมฟอเรส

2) เพื่ออภิปรายและเปรียบเทียบผลลัพธ์ของอัลกอริทึมในการจำแนกประเภท (Classification) ของประสิทธิภาพการทำงานโดยรวมของเครื่องจักรที่ใช้ในกระบวนการผลิต

1.3 ขอบเขตของงานวิจัย

1) ในการศึกษาครั้งนี้มุ่งเน้นไปที่การศึกษาแนวโน้มการทำงานที่ผิดปกติของเครื่องจักรที่ใช้ในกระบวนการผลิตสินค้าของโรงงานอุตสาหกรรม

2) ใช้ชุดข้อมูลจากแหล่งข้อมูลบนเว็บไซต์ Machine Learning Repository ของมหาวิทยาลัยแคลิฟอร์เนียเออร์ไวน์ (University of California Irvine: UCI) เป็นชุดข้อมูลตัวอย่างในระบบการบำรุงเชิงรักษาของเครื่องจักรในโรงงานแห่งหนึ่ง (AI4I 2020 Predictive Maintenance Dataset)

3) ตัวแปรอิสระที่ใช้ในการศึกษา ได้แก่ หมายเลขอ้างอิง รหัสสินค้า ขนาดของสินค้า เครื่องมือวัดอุณหภูมิ เครื่องมือวัดความดัน แรงบิดมอเตอร์ เวลาชดเชยของเครื่องจักรและความเร็วมอเตอร์

4) ตัวแปรตามที่ใช้ในการศึกษา ได้แก่ สถานะการทำงานของเครื่องจักรที่เกี่ยวข้องในกระบวนการผลิต ประกอบด้วย สถานะปกติ สถานะความผิดปกติต่าง ๆ ที่เกิดขึ้นของเครื่องจักร

5) เลือกใช้โมเดล Gradient Boosting Machine, k-Nearest Neighbors และ Random Forest ในการสร้างแบบจำลองในการคาดการณ์ประสิทธิภาพการทำงานโดยรวมของเครื่องจักรที่ใช้ในกระบวนการผลิต อัลกอริทึมที่เลือกมาทดลองเหล่านี้จัดเป็นอัลกอริทึมชนิดการจำแนกประเภทของการเรียนรู้ของเครื่อง (Machine Learning)

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1) ทราบถึงแนวโน้มการทำงานที่ผิดปกติของเครื่องจักรรวมถึงปัจจัยเสี่ยงร่วมอื่น ๆ ที่เกิดขึ้นในกระบวนการผลิตสินค้า

2) ทราบข้อมูลคาดการณ์ที่แสดงออกมาเพื่อนำไปสู่การวิเคราะห์สถานการณ์แบบล่วงหน้า เพื่อนำมาปรับเปลี่ยนวิธีการผลิตในกระบวนการผลิตให้เหมาะสม โดยใช้เทคโนโลยีการวิเคราะห์ข้อมูลมาเป็นตัวบ่งชี้และตัดสินใจ

3) สามารถนำข้อมูลเหล่านี้ไปใช้เป็นแนวทางในการลดปัญหาความผิดพลาดการทำงานของเครื่องจักร อาจทำให้เกิดผลกระทบต่อกระบวนการผลิตในอุตสาหกรรม และนำมาปรับเปลี่ยนวิธีการผลิตในกระบวนการผลิตให้เหมาะสม

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 แนวคิดเกี่ยวกับระบบการผลิต

ระบบการผลิต (Production System) หมายถึง การผลิตเป็นกระบวนการที่ทำให้เกิดการสร้างสิ่งหนึ่งสิ่งใดขึ้นมาจากการใช้ทรัพยากรหรือปัจจัยการผลิตที่มีอยู่ การดำเนินการผลิตจะเป็นไปตามลำดับขั้นตอนของการกระทำก่อนหลัง เพื่อให้การผลิตบรรลุวัตถุประสงค์ดังกล่าวนั้น จึงจำเป็นต้องมีการจัดการให้อยู่ในรูปของระบบการผลิต ซึ่งประกอบด้วยส่วนที่สำคัญ 3 ส่วน คือ ปัจจัยการผลิต (Input) กระบวนการแปลงสภาพ (Conversion Process) หรือเรียกอีกอย่างหนึ่งว่ากระบวนการผลิต (Process) และผลผลิต (Output)

ในระบบการผลิตที่มีประสิทธิภาพจะต้องคำนึงถึงปัจจัยด้านปริมาณ คุณภาพ เวลา และราคา ซึ่งปัจจัยทั้งหมดนี้จะต้องนำมารวมไว้ในระบบการผลิต โดยกำหนดการวางแผนและควบคุมการผลิตเป็นขั้นตอนหลักในกิจกรรมต่าง ๆ ที่อยู่ในระบบการผลิตนั้น สามารถจำแนกได้เป็น 3 ขั้นตอน คือ

1) การวางแผน (Planning) เป็นขั้นตอนของการวิเคราะห์ข้อมูลที่มีอยู่ และวางแผนการใช้ทรัพยากรให้ตรงตามเป้าหมายที่ต้องการ และดำเนินการเป็นไปอย่างมีประสิทธิภาพ โดยกำหนดเป้าหมายย่อยไว้ในแผนกต่าง ๆ ในเทอมของเวลาที่กำหนดไว้ก่อนล่วงหน้า และจากเป้าหมายย่อย ๆ ที่ถูกกำหนดขึ้นเหล่านี้ เพื่อส่งผลไปยังเป้าหมายที่ต้องการ

2) การดำเนินงาน (Operation) เป็นขั้นตอนของการดำเนินการผลิต โดยดำเนินการตามรายละเอียดต่าง ๆ ในขั้นตอนการวางแผนที่ถูกกำหนดไว้ในแผนการผลิต

3) การควบคุม (Control) เป็นขั้นตอนของการตรวจตรา ให้คำแนะนำและติดตามผลเกี่ยวกับการดำเนินงาน โดยใช้การป้อนกลับของข้อมูล (Feedback Information) ในทุก ๆ ขั้นตอนการดำเนินงาน ผ่านกลไกการควบคุม (Control Mechanism) โดยที่กลไกเหล่านี้ทำหน้าที่ปรับปรุงแผนงานเพื่อให้บรรลุตามเป้าหมายที่กำหนดไว้

ปัจจัยในการผลิต ประกอบด้วย คน วัตถุดิบ เครื่องจักร พลังงาน เงินและข้อมูลข่าวสารที่มีผลต่อกระบวนการผลิต ได้แก่ การเตรียมวัตถุดิบต่าง ๆ การนำส่วนประกอบต่าง ๆ เข้าด้วยกันการสร้างรูปทรง การตกแต่ง รูปทรงตลอดทั้งการบรรจุผลิตภัณฑ์เพื่อการจำหน่าย ในส่วนที่เป็นผลผลิตได้แก่ ผลิตภัณฑ์สำเร็จรูป (Products) ผลผลิตจะออกมาในรูปของสินค้าหรือบริการ ซึ่งรวมเรียกว่าระบบการผลิต นอกจากนี้การควบคุมและการตรวจสอบคุณภาพเป็นสิ่งที่สำคัญต่อกระบวนการผลิตเพื่อให้แน่ใจว่าสินค้าที่ผลิตออกมาตรงตามลูกค้าต้องการ และตรงตามมาตรฐานที่กำหนดไว้

2.2. วิธีการประเมินผลประสิทธิภาพของเครื่องจักร

ประสิทธิผลโดยรวมของเครื่องจักรเป็นตัววัดชนิดหนึ่งในระบบการบำรุงรักษาแบบทวีผล (Total Productive Manufacturing) เพื่อต้องการให้ทุกคนภายในองค์กรได้มีส่วนร่วมในการจัดการควบคุม ดูแลประสิทธิผลโดยรวมของเครื่องจักร ซึ่งถือว่าเป็นเป้าหมายที่สำคัญของกระบวนการผลิตในโรงงานอุตสาหกรรม ในกิจกรรมการปรับปรุงประสิทธิภาพเครื่องจักรเพื่อลดความสูญเสีย ถือเป็นกิจกรรมพื้นฐานสำหรับระบบการบำรุงรักษาแบบทวีผล โดยพิจารณาการสูญเสียของเครื่องจักรและบันทึกความสูญเสียที่เกิดขึ้น เพื่อทำการคำนวนและวิเคราะห์ค่าประสิทธิผลโดยรวมของเครื่องจักร (Overall Equipment Effectiveness: OEE) ทำให้เราทราบได้ว่าความสูญเสียตัวใดที่จะทำให้ประสิทธิผลโดยรวมของเครื่องจักรต่ำ หลักการความสูญเสีย 6 ประการ (Six Big Losses) ในกระบวนการผลิต ได้แก่

1) ความสูญเสียจากการหยุดของเครื่องจักร (Breakdown) เป็นความสูญเสียที่เกิดจากการขัดข้องของตัวเครื่องจักร หรือการหยุดเครื่อง โดยไม่มีการวางแผนการหยุดล่วงหน้า การบันทึกความสูญเสียของประเภทนี้ จะต้องระบุลักษณะของการหยุดของเครื่องจักร เช่น ลูกปืนแตก สายพานขาด หรือมอเตอร์ไหม้ เป็นต้น สามารถนำข้อมูลไปวิเคราะห์เพื่อหาสาเหตุในการปรับปรุง ฟื้นฟูสภาพของเครื่องจักรได้ถูกต้อง

2) ความสูญเสียการติดตั้งเตรียมงาน (Set Up and Adjustment Loss) เป็นความสูญเสียที่เกิดจากการเปลี่ยนรุ่นการผลิตในแต่ละครั้ง ซึ่งเป็นผลมาจากความต้องการผลิตภัณฑ์ของผู้บริโภคที่หลากหลาย ทำให้ผู้ผลิตมีความจำเป็นต้องมีการปรับเปลี่ยนการผลิตบ่อยครั้งมากขึ้น

3) ความสูญเสียจากการหยุดเพียงชั่วขณะ (Idling Time and Minor Stops Losses) เป็นความสูญเสียที่เกิดจากเครื่องขัดข้อง ทำให้เครื่องจักรหยุดผลิตเป็นเวลาช่วงสั้น ๆ ความสูญเสียประเภทนี้มักมีอยู่มากในกระบวนการผลิตจริง

4) ความสูญเสียด้านความเร็วในการผลิต เมื่อเครื่องจักรเสื่อมสภาพ (Reduced Speed Losses) เป็นความสูญเสียที่เกิดจากความเร็วจริงที่ใช้งานต่ำกว่าความเร็วมาตรฐานของเครื่องจักรที่กำหนดไว้ ส่งผลทำให้ผลผลิตสินค้าได้น้อยกว่าแผนตามระยะเวลาที่กำหนด

5) ความสูญเสียในระหว่างกระบวนการผลิตหรืองานเสียหาย ในกรณีเครื่องจักรเสื่อมสภาพจนไม่สามารถผลิตได้ (Quality Defect and Rework) ทำให้ผลิตภัณฑ์ไม่เป็นไปตามความต้องการของผู้บริโภค ในการจัดเก็บข้อมูลความสูญเสียประเภทนี้ มีวิธีการโดยการคัดแยกลักษณะการเสียหายของสินค้า รวมถึงการแยกตามชนิดของเครื่องในกรณีที่เป็นกระบวนการผลิตแบบต่อเนื่อง เพื่อให้ง่ายต่อการนำข้อมูลไปวิเคราะห์และวางแผนการผลิตในขั้นตอนต่อไป

6) ความสูญเสียเมื่อเริ่มผลิตงานใหม่เพราะเครื่องจักรไม่คงที่ (Startup Reject) เป็นความสูญเสียที่เกิดจากเครื่องจักรในช่วงเริ่มต้นผลิต ความสูญเสียประเภทนี้เป็นความสูญเสียด้านคุณภาพของผลิตภัณฑ์ ปริมาณการสูญเสียมากหรือน้อยขึ้นอยู่ลักษณะการทำงานของเครื่องจักร

สิ่งที่สำคัญของความสูญเสีย 6 ประการ เป็นเหตุการณ์ที่ก่อให้เกิดความสูญเสียหรือความเสียหายระหว่างการผลิต จะต้องสามารถแบ่งแยกชนิดของความสูญเสียที่เกิดขึ้นจริงให้ได้ เพื่อจะได้แสดงค่าประสิทธิผลโดยรวมของเครื่องจักร โดยขั้นตอนการประเมินผลประสิทธิภาพโดยรวมของเครื่องจักร (Overall Equipment Effectiveness) เพื่อเพิ่มประสิทธิภาพการทำงานของเครื่องจักร สามารถคำนวณได้จากสมการที่ (2.1)

$$OEE = A \times P \times Q \tag{2.1}$$

ประกอบด้วย อัตราการเดินเครื่องจักร (Availability: A) โดยพิจารณาจากความสูญเสียจากการหยุดของเครื่องจักร และความสูญเสียการติดตั้งเตรียมงานก่อนการผลิต โดยคำนวณจากสมการที่ (2.2)

$$A = \frac{\text{เวลารับภาระงาน} - \text{เวลาเครื่องจักรหยุด}}{\text{เวลารับภาระการผลิตทั้งหมด}} \tag{2.2}$$

ค่าประสิทธิภาพของเครื่องจักร (Performance Efficiency: P) โดยพิจารณาจากความสูญเสียจากการหยุดเพียงชั่วขณะและความสูญเสียด้านความเร็วในการผลิต โดยคำนวณจากสมการที่ (2.3)

$$P = \frac{\text{เวลาเดินเครื่องสุทธิ}}{\text{เวลาเดินเครื่อง}} \tag{2.3}$$

อัตราคุณภาพ (Quality Rate: Q) โดยพิจารณาจากความสูญเสียในระหว่างกระบวนการผลิตหรืองานเสียหายและความสูญเสียเมื่อเริ่มผลิตงานใหม่เพราะเครื่องจักรไม่คงที่ โดยคำนวณจากสมการที่ (2.4)

$$Q = \frac{\text{จำนวนชิ้นงานที่ผลิตได้} - \text{จำนวนชิ้นงานเสีย}}{\text{เวลาเดินเครื่อง}} \tag{2.4}$$

2.3 แนวคิดเกี่ยวกับการเรียนรู้ของเครื่อง

การเรียนรู้ของเครื่อง (Machine Learning) หมายถึง ความสามารถของระบบสารสนเทศในการหาหนทางแก้ปัญหาด้วยตนเอง โดยปราศจากการป้อนคำสั่งของโปรแกรมเมอร์ การเรียนรู้ของเครื่องเป็นรูปแบบหนึ่งของการวิเคราะห์ข้อมูล ที่ดำเนินการวิเคราะห์ด้วยแบบจำลองอย่างอัตโนมัติ ถือว่าเป็นส่วนหนึ่งของปัญญาประดิษฐ์ (Artificial Intelligence) โดยการเรียนรู้ของเครื่องทำหน้าที่จดจำรูปแบบในฐานข้อมูล และช่วยให้ระบบสารสนเทศสามารถเข้าใจรูปแบบบนพื้นฐานของอัลกอริทึมและชุดข้อมูลที่มีอยู่ วิเคราะห์เพื่อพัฒนากระบวนการแก้ปัญหาที่เหมาะสม ดังนั้นในการเรียนรู้ประดิษฐ์ (Artificial Knowledge) ถูกสร้างขึ้นบนพื้นฐานของประสบการณ์ เพื่อทำให้โปรแกรมซอฟต์แวร์สามารถสร้างวิธีการได้อย่างอิสระ โดยมนุษย์เป็นผู้กำหนดและป้อนคำสั่งให้กับชุดอัลกอริทึมและข้อมูลที่จำเป็นลงในระบบไว้ล่วงหน้า รวมถึงกฎการวิเคราะห์ข้อมูลตามลำดับเพื่อการจดจำรูปแบบต่าง ๆ (Patterns) ในคลังข้อมูล ซึ่งในระบบของโมเดลนั้นจะสามารถแสดงผลการทำงานของการเรียนรู้ของเครื่องได้

ประเภทโมเดลอัลกอริทึมของการเรียนรู้ของเครื่อง (Machine Learning) แบ่งออกเป็นทั้งหมด 3 ประเภท ได้แก่ การเรียนรู้แบบมีผู้สอน (Supervised Learning) การเรียนรู้แบบไม่มีผู้สอน (Unsupervised Learning) และการเรียนรู้แบบเสริมกำลัง (Reinforcement Learning)

2.3.1 การเรียนรู้แบบมีผู้สอน (Supervised Learning)

การเรียนรู้แบบมีผู้สอน (Supervised Learning) คือ อัลกอริทึมที่จำเป็นต้องใช้ข้อมูลในส่วนสำหรับการฝึกสอน (Training) และส่วนที่รับกลับมาเพื่อปรับปรุง (Feedback) จากมนุษย์เพื่อที่จะเรียนรู้ความสัมพันธ์ระหว่างข้อมูลที่ถูกป้อนเข้ามาสู่ข้อมูลที่ออกไป เมื่อทราบผลลัพธ์ของข้อมูล อัลกอริทึมนี้จะสามารถทำนายข้อมูลใหม่ได้ โดยวิธีการเรียนรู้นั้นถูกกำหนดด้วยโมเดลตัวอย่างไว้ล่วงหน้า เพื่อให้แน่ใจว่ามีการจัดสรรข้อมูลอย่างเพียงพอกับกลุ่มโมเดลที่เกี่ยวข้องของอัลกอริทึมเหล่านี้ ระบบจะเรียนรู้บนพื้นฐานของอินพุตและเอาท์พุตตามที่กำหนดไว้ ส่วนโปรแกรมเมอร์ซึ่งทำหน้าที่เสมือนครูผู้สอน เป็นผู้จัดสรรค่าที่เหมาะสมสำหรับข้อมูลอินพุตโดยเฉพาะ (Particular Input) ซึ่งเป้าหมายที่สำคัญของการเรียนรู้คือ ได้เรียนรู้บริบทในระบบของการคำนวณต่อเนื่องกับอินพุตและเอาท์พุตที่แตกต่างกัน และเพื่อสร้างการเชื่อมต่อระหว่างข้อมูล

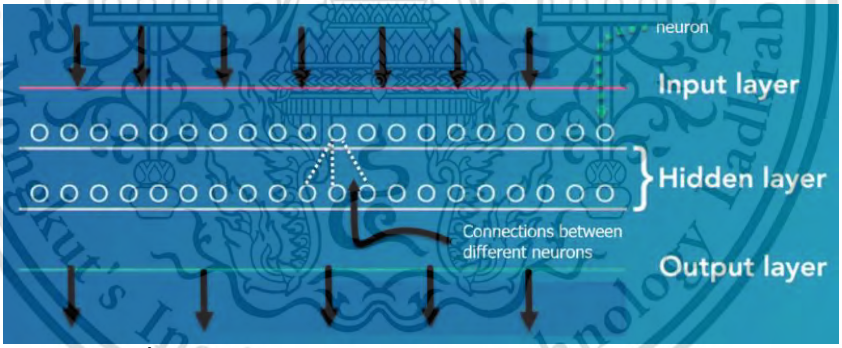
ลักษณะของอัลกอริทึมการเรียนรู้แบบมีผู้สอนแบ่งออกเป็น 2 ประเภทหลัก ได้แก่ การจำแนกประเภท (Classification) และการถดถอย (Regression) แต่ละอัลกอริทึมการเรียนรู้แบบมีผู้สอนแต่ละประเภทมีความแตกต่างกัน ซึ่งในส่วนโมเดลอัลกอริทึมการเรียนรู้แบบมีผู้สอนของการจำแนกประเภท (Classification Supervised Learning) เหมาะกับการตั้งสมมติฐานข้อมูลลักษณะกลุ่ม ประเภทหรือชุดข้อมูลที่ไม่มีความต่อเนื่อง ยกตัวอย่างเช่น ใช่/ไม่ใช่ ชาย/หญิง หรือลำดับตัวเลข เป็นต้น ถ้าหากผู้วิจัยเลือกใช้โมเดลอัลกอริทึมแบบการถดถอย เหมาะกับการ

ตั้งสมมติฐานข้อมูลที่มีความต่อเนื่องหรือไม่ได้รวมกันเป็นกลุ่ม ได้แก่ 0 - 1000 เป็นต้น

ขั้นตอนวิธีการประเมินผลของโมเดลอัลกอริทึมของการจำแนกประเภท นิยมเลือกใช้ Confusion Metrix เป็นตัววัดค่าความแม่นยำ (Accuracy) ในขณะที่วิธีการประเมินผลของโมเดลอัลกอริทึมของการถดถอย มักจะเลือกใช้วิธีการวัดค่าความแม่นยำด้วยวิธีการหาค่า R-squared ค่า Mean Absolute Percentage Error (MAPE) หรือ Root Mean Squared Error (RMSE) เป็นต้น

ตัวอย่างโมเดลประเภทการเรียนรู้แบบมีผู้สอน ได้แก่

1) โครงข่ายการเรียนรู้เชิงลึก (Deep Neural Networks) เป็นอัลกอริทึมที่นำหลักการทำงานของระบบโครงข่ายประสาทในสมองของมนุษย์ เซลล์สมองมนุษย์ที่เรียกว่า นิวรอน (Neuron) ก่อตัวเป็นเครือข่ายที่ซับซ้อนและเชื่อมต่อถึงกันสูง และส่งสัญญาณไฟฟ้าให้ต่อกันและกัน เพื่อช่วยให้มนุษย์ประมวลผลได้ [6] ในโครงข่ายการเรียนรู้เชิงลึกประกอบด้วยนิวรอนเทียม เป็นโมดูลซอฟต์แวร์ชนิดหนึ่ง หน่วยของนิวรอนเทียมเรียกว่า “โหนด” ในแต่ละโหนดทำงานร่วมกันเพื่อวิเคราะห์และแก้ไขปัญหา โดยทั่วไปหลักการทำงานของเทคนิค Deep Neural Networks แบ่งออกเป็น 3 เลเยอร์ ได้แก่ เลเยอร์อินพุต เลเยอร์ซ่อน เลเยอร์เอาต์พุต หลักการทำงานของ Deep Neural Networks แสดงดังรูปที่ 2.1



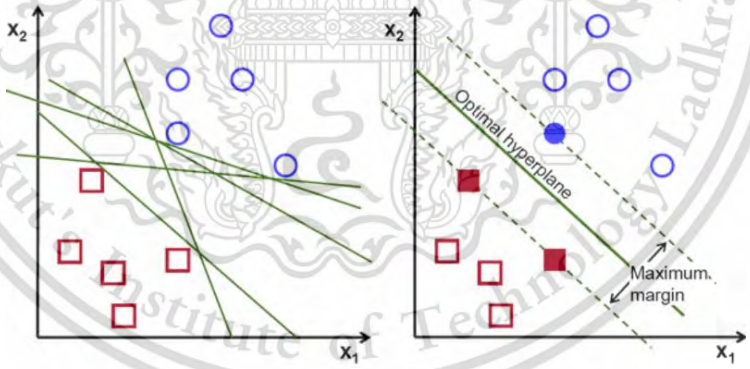
รูปที่ 2.1 หลักการทำงานของ Deep Neural Networks

จากรูปที่ 2.1 หลักการทำงานของ Deep Neural Networks คือ การใช้ลำดับชั้นของระบบโครงข่ายประสาท (Neuron) มาต่อกัน โดยลำดับชั้นแรก ทำหน้าที่รับข้อมูล (Input Layer) และส่งข้อมูลไปยังลำดับชั้นสุดท้าย เพื่อทำหน้าที่แสดงผลลัพธ์การประมวลผลออกมา (Output Layer) ส่วนลำดับชั้นที่อยู่ระหว่างชั้นแรกและชั้นสุดท้ายคือ เลเยอร์ซ่อน (Hidden Layer) โดยทั่วไปในโครงข่ายการเรียนรู้เชิงลึกมักจะมีเลเยอร์ซ่อนอยู่หลากหลายชั้น โดยส่วนที่เป็นนิวรอนเทียมในแต่ละโหนดถูกให้เชื่อมต่อถึงกันเป็นจำนวนมาก ในส่วนที่เป็นจำนวนเครือข่ายสามารถบ่งบอกถึงน้ำหนักที่แสดงถึงการเชื่อมต่อระหว่างโหนดหนึ่งกับโหนดอื่น โหนดที่มีค่าน้ำหนักสูงกว่าจะมีอิทธิพลต่อโหนด

อื่น ๆ มากกว่าการนำมาใช้งานนั้นเหมาะกับงานที่มีความซับซ้อนมีข้อมูลเข้าเป็นจำนวนมากหรือเป็นข้อมูลที่ไม่มีโครงสร้าง (Unstructured Data) ได้แก่ การวิเคราะห์รูปภาพ การแปลภาษาอัตโนมัติ เป็นต้น

2) ซัพพอร์ตเวคเตอร์แมชชีน (Support Vector Machine) เป็นอัลกอริทึมจำแนกเชิงเส้นแบบไบนารี่ ลักษณะการแบ่งแยกข้อมูลแบ่งออกเป็นสองชนิด ในการจำแนกข้อมูลของอัลกอริทึมซัพพอร์ตเวคเตอร์แมชชีนจะทำงานได้ดีในด้านการจำแนกข้อมูลที่มีลักษณะมิติจำนวนมาก ถือว่าเป็นอัลกอริทึมหนึ่งที่มีประสิทธิภาพอย่างมาก เหมาะสำหรับวิเคราะห์ข้อมูลและจำแนกเพื่อแก้ปัญหาการหาค่าที่เหมาะสม (Optimization) โดยกำหนดค่าสัมประสิทธิ์ Convex ε-insensitive loss function ของสมการ นำมาวางในฟีเจอร์สเปซ เพื่อสร้างเส้นตรงสำหรับแบ่งแยกกลุ่มข้อมูลที่เรียกว่า “Optimal Separating Hyperplane” ที่ดีที่สุดและถูกป้อนเข้าสู่กระบวนการสอนให้ระบบเรียนรู้

หลักการทำงานของซัพพอร์ตเวคเตอร์แมชชีน อาศัยการสร้างเส้นแบ่ง หรือเรียกอีกชื่อว่า “ไฮเปอร์เพลน” (Hyperplane) เพื่อแบ่งจุดข้อมูลออกเป็น 2 ชุดด้วยระยะห่างที่มากที่สุด เรียกว่า Maximum Margin จากนั้นอัลกอริทึมนี้จะทำการหาว่าไฮเปอร์เพลนใดเป็นเส้นที่ใช้แบ่งจุดข้อมูลได้อย่างถูกต้องมากที่สุด (Optimal Hyperplane) ดังรูปที่ 2.2

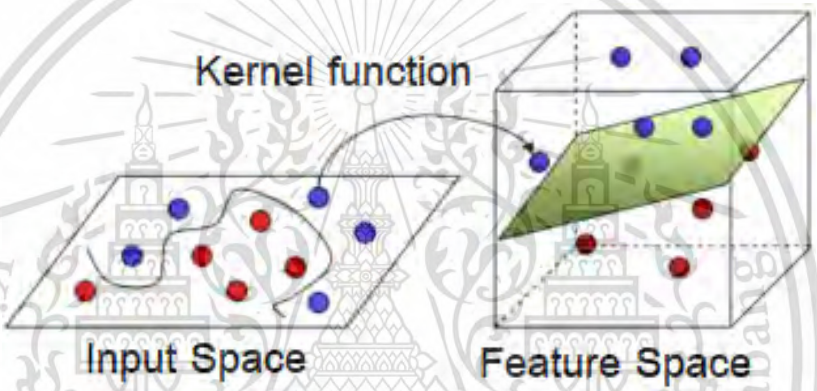


รูปที่ 2.2 อัลกอริทึมซัพพอร์ตเวคเตอร์แมชชีนใน 2 มิติ

จากรูปที่ 2.2 สามารถจำแนกข้อมูลออกเป็น 2 คลาส โดยใช้ไฮเปอร์เพลนที่เป็นเส้นตรงในการแบ่งข้อมูล จะเห็นว่ามีเส้นตรงหลายเส้นที่สามารถแบ่งแยกข้อมูลออกจากกันได้ แต่เส้นตรงใดจะถูกพิจารณาให้เป็นเส้นตรงที่ดีที่สุดนั้น จะพิจารณาจากผลรวมของระยะห่างระหว่างเส้นไฮเปอร์เพลนกับเส้นตรงที่ลากผ่านข้อมูลที่อยู่ใกล้ที่สุดและขนานกับเส้นไฮเปอร์เพลนของข้อมูลในแต่

ละกลุ่มที่มากที่สุด เมื่อกำหนดเส้นขอบของแต่ละฝั่ง จึงเกิดเส้นแบ่งพรมแดนข้อมูลและทำการหาจุดข้อมูลที่อยู่ใกล้กับเส้นแบ่งนี้ จุดข้อมูลดังกล่าวถูกเรียกว่า “Support Vector” ถือว่าเป็นตัวแปรที่สำคัญต่อการหาค่า Maximum Margin เนื่องจากค่าพารามิเตอร์ Support Vector มีผลต่อลักษณะการกระจายตัวของข้อมูล ถ้าหากปรับค่าพารามิเตอร์ Support Vector ไม่ดี อาจจะทำให้ข้อมูลเกิด Outlier ได้

จากตัวอย่างข้างต้นเป็นการสร้างไฮเปอร์เพลนที่เป็นเส้นตรง ซึ่งสามารถใช้งานได้ดีกับปัญหาที่มีลักษณะข้อมูลเชิงเส้น นอกจากนี้สามารถนำอัลกอริทึมซัพพอร์ตเวคเตอร์แมชชีน มาใช้กับชุดข้อมูลที่ไม่เป็นเชิงเส้น เพื่อแก้ปัญหาและหาความสัมพันธ์ระหว่างข้อมูล ดังรูปที่ 2.3



รูปที่ 2.3 รูปแบบการทำงานของฟังก์ชัน Kernel

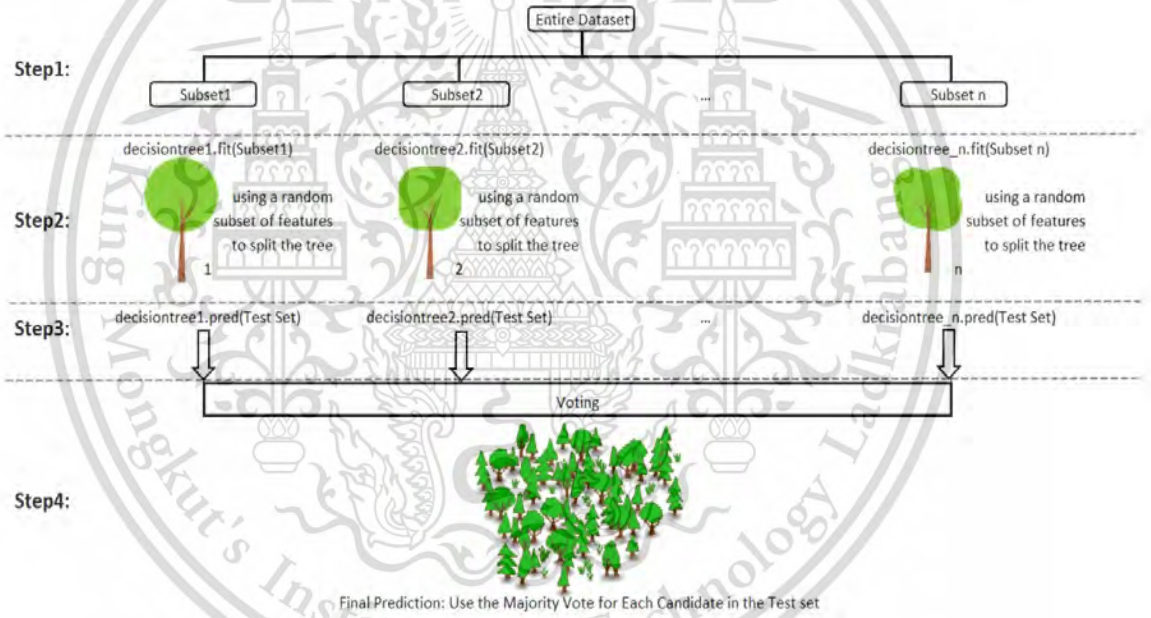
จากรูปที่ 2.3 ในกรณีชุดข้อมูลไม่เป็นเชิงเส้นนั้นไม่สามารถถูกแบ่งได้ด้วย Linear Hyperplane แต่สามารถเลือกใช้วิธีการใช้ฟังก์ชัน Kernel เพื่อหารูปแบบและความสัมพันธ์ของชุดข้อมูลไม่เป็นเชิงเส้น โดยแยกเป็น Non Linear เพื่อให้ได้ผลลัพธ์ที่มีความถูกต้องแม่นยำสูง ฟังก์ชัน Kernel เป็นฟังก์ชัน Non Line ที่สร้างจากข้อมูลที่มีอยู่เดิมแล้วเปลี่ยนให้อยู่ในมิติใหม่ ทำให้ได้ Hyperplane เป็นวงกลม ดังนั้นอัลกอริทึมซัพพอร์ตเวคเตอร์แมชชีน เหมาะกับการใช้แก้ปัญหาชุดข้อมูลที่มีคุณลักษณะจำนวนมาก แต่มีปริมาณข้อมูลน้อยถึงปานกลาง ถ้าหากใช้กับชุดข้อมูลที่มีปริมาณมาก จะส่งผลต่อประสิทธิภาพของอัลกอริทึมลดลงและใช้เวลาในการฝึกฝนข้อมูลเพิ่มขึ้น

3) การเรียนรู้แบบกลุ่ม (Ensemble Learning) เป็นวิธีการรวบรวมโมเดลมากกว่าหนึ่งชนิดมาประกอบ และให้โมเดลเหล่านั้นได้ทำงานร่วมกัน โดยใช้ข้อมูลชุดเดียวกัน เมื่อได้ผลการจำแนกของแต่ละตัวจำแนกแล้ว ก็จะนำผลลัพธ์เหล่านั้นมาผ่านการรวบรวมข้อมูล ใช้ตัวจำแนกมากกว่าหนึ่งตัวนำไปตัดสินใจโดยการสร้างแบบหลายตัว เพื่อจำแนกประเภทของข้อมูล ในการวิจัยได้

นำอัลกอริทึมการเรียนรู้แบบกลุ่มมาใช้ในการทดลอง ได้แก่ Gradient Boosting Machine และ Random Forest

Random Forest เป็นอัลกอริทึมประเภทการเรียนรู้แบบกลุ่มชนิดหนึ่ง ซึ่งถูกสร้างจากกฎการตัดสินใจต้นไม้ (Decision Tree) เมื่อโมเดลผู้ทำนายแต่ละตัวจะต้องเรียนรู้อย่างเป็นอิสระต่อกันให้มากที่สุด โดยที่ไม่ต้องนำข้อมูลจากผู้อื่นมาเป็นส่วนในการตัดสินใจ โดยส่วนใหญ่มักเลือกใช้โมเดล Random Forest สำหรับแก้ปัญหาในเรื่องข้อมูลที่มีมิติสูง (High Dimensional)

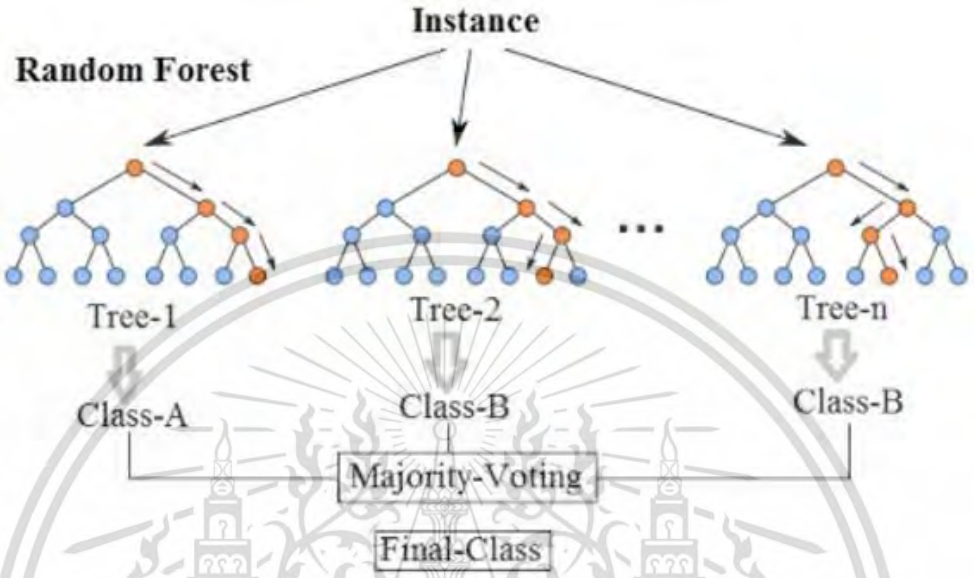
ลักษณะข้อมูลที่มิติสูงเป็นลักษณะที่มีจำนวนตัวแปรอิสระมากกว่าจำนวนตัวอย่างของข้อมูล โดยค่าสหสัมพันธ์ระหว่างต้นไม้การตัดสินใจแต่ละต้นจะถูกสร้างให้มีความเป็นอิสระต่อกัน เพื่อต้องการลดจำนวนตัวแปรในการแปลงข้อมูล (Transform Data) ก่อนนำชุดข้อมูลนั้นมาวิเคราะห์และประมวลผลในขั้นตอนต่อไป เพื่อหาผลลัพธ์ที่ดีที่สุดในการทำนาย โดยใช้เทคนิค “Bagging” หรือเรียกอีกชื่อหนึ่งว่า “Boostrapping” ดังรูปที่ 2.4



รูปที่ 2.4 หลักการทำงานของเทคนิค Bagging

จากรูปที่ 2.4 หลักการทำงานของเทคนิค Bagging เริ่มจากนำข้อมูลทั้งหมดแบ่งเป็นชุดข้อมูลสำหรับเทรนนิ่งในแต่ละเซต จากนั้นนำชุดข้อมูลทั้งหมดเข้าไปที่โมเดล เพื่อทำการเทรนนิ่งชุดข้อมูลและสร้างชุดข้อมูลใหม่ เรียกว่า “Classifier” และทำการเลือกชุดข้อมูลที่ดีที่สุดโดยการทำการโหวด (Voting) ลักษณะเฉพาะของเทคนิค Bagging คือทำให้ลดการเกิด Overfitting และความแปรปวน (Variance) ของข้อมูลได้ดีและสามารถเพิ่มจำนวนต้นไม้พร้อมกันได้ เนื่องจากวิธีการ Bagging ในการตัดสินใจ สามารถเลือกใช้ข้อมูลเดียวกันได้ นอกจากนี้ยังมีวิธี Pasting ที่จะไม่สามรถ

เลือกใช้การสุ่มมรายการซ้ำกันได้ ทำให้ความแปรปรวน (Variance) ของวิธีการ Bagging ลดลง เพราะมีการเลือกข้อมูลที่ซ้ำกันได้ และทำให้ประสิทธิภาพของโมเดลมีความเสถียรและแม่นยำมากกว่าวิธีการ Pasting ตามรูปที่ 2.5



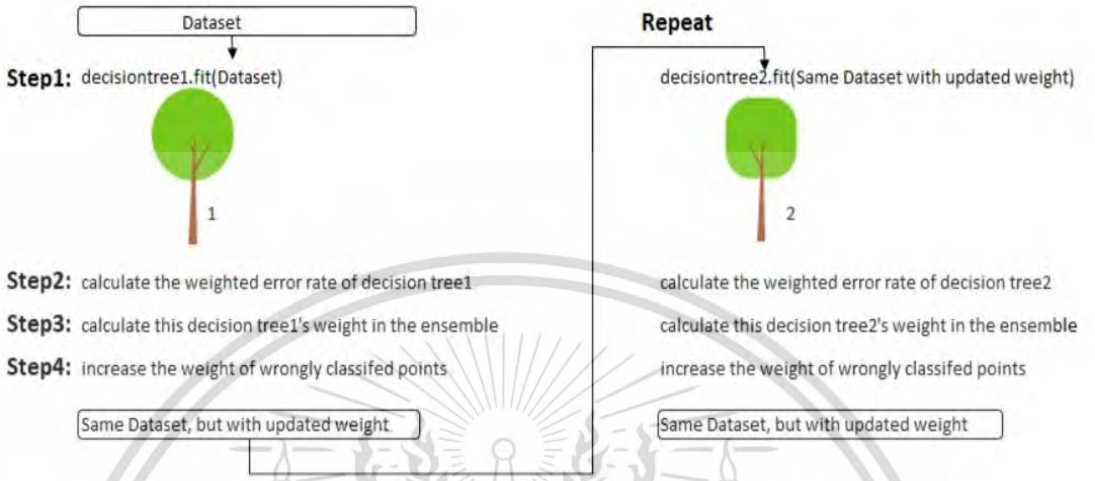
รูปที่ 2.5 แสดงตัวอย่างของต้นไม้ตัดสินใจอย่างง่าย

จากรูปที่ 2.5 หลักการทำงานของอัลกอริทึม Random Forest เริ่มจากการสุ่มตัวอย่างชุดใหม่จากจำนวนข้อมูลทั้งหมด โดยใช้วิธีสุ่มแบบแทนที่ให้ได้ออกมา โดยมีลักษณะไม่เหมือนกันมาสร้างโมเดลต้นไม้ตัดสินใจ (Decision Tree) และในแต่ละต้นไม้ประกอบด้วยโหนดของการตัดสินใจ (Decision Node) และเชื่อมต่อกันด้วยกิ่งก้านต่างๆ (Branches) ขยายออกจากโหนดราก (Root Node) ซึ่งในแต่ละโหนดการตัดสินใจแสดงถึงการทดสอบคุณลักษณะ (Instance) ของข้อมูล โดยในแต่ละกิ่งก้านจะนำไปสู่โหนดการตัดสินใจอีกโหนดหนึ่ง หรือสิ้นสุดอยู่ที่โหนดใบ จากนั้นทำการสุ่มตัวแปรด้วยวิธีการ Majority Voting เป็นวิธีการโหวตสุ่มเลือกตัวแปรที่มีคำตอบเหมือนกันหรือคำตอบที่มากที่สุด เพื่อให้ได้ผลลัพธ์หรือกลุ่ม (Class) ที่ถูกต้องที่สุดในการวิเคราะห์ข้อมูล

อัลกอริทึม Gradient Boosting เป็นโมเดลทำงานประมวลผลแบบต่อเนื่อง ซึ่งจะใช้เทคนิค “Boosting Ensemble” โดยการทำงานของอัลกอริทึมนี้จะใช้หลักการเดียวกับของอัลกอริทึม Decision Tree โดยทำหน้าที่เรียนรู้จากค่าความผิดพลาดที่เกิดขึ้นในโครงข่าย หรือเรียกว่า “Parallel Tree”

โดยทั่วไปอัลกอริทึม Gradient Boosting เป็นเทคนิคการเรียนรู้ของเครื่องเลือกใช้กลุ่มผู้เรียนที่อ่อนแอ เพื่อฝึกซ้ำและรวมเข้าด้วยกันเพื่อสร้างแบบจำลองที่มีประสิทธิภาพสูงสุด โดยทั่วไปโครงสร้างพื้นฐานในการประมวลผลเป็นแบบแผนผังในการตัดสินใจ

อัลกอริทึมตระกูล Gradient Boosting ถือว่าเป็นอัลกอริทึมที่มีความยืดหยุ่น แม่นยำและมีประสิทธิภาพที่ดีที่สุด หลักการทำงานของเทคนิค Boosting Ensemble แสดงดังรูปที่ 2.6



รูปที่ 2.6 หลักการทำงานของเทคนิค Boosting Ensemble

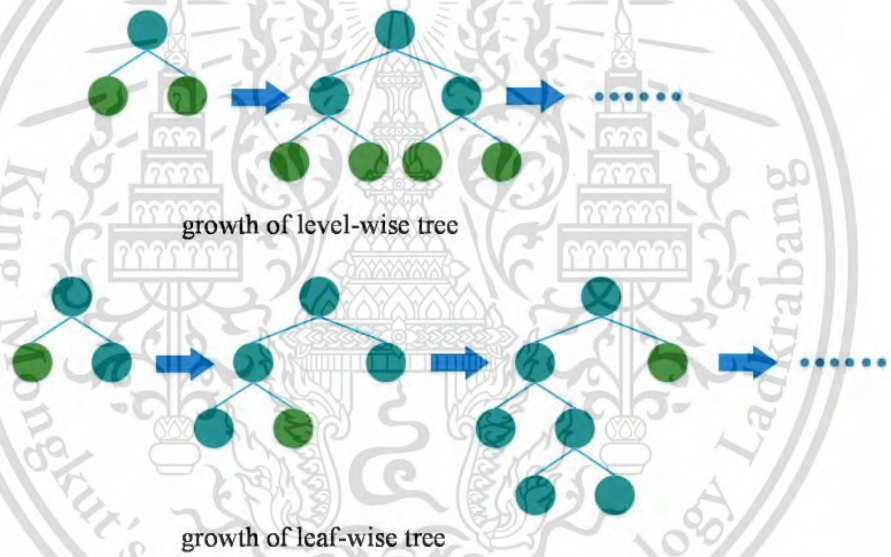
จากรูปที่ 2.6 หลักการทำงานของเทคนิค Boosting Ensemble ทำการสร้างโมเดลและรวบรวมต้นไม้ตัดสินใจที่อ่อนแอหลาย ๆ รายการตามลำดับ โดยกำหนดน้ำหนักให้กับผลลัพธ์ของต้นไม้แต่ละรายการ จากนั้นจึงทำให้การจัดประเภทที่ไม่ถูกต้องจากต้นไม้การตัดสินใจแรกมีน้ำหนักมากขึ้นและป้อนข้อมูลไปยังต้นไม้ถัดไป โดยเทรนนิ่งข้อมูลหลาย ๆ รอบ ในวิธีการ Boosting จึงรวบรวมกฎที่อ่อนแอเหล่านี้เข้าด้วยกัน ลักษณะการเรียนรู้ของเทคนิค Boosting มีความแตกต่างกับเทคนิค Bagging คือวิธีการฝึกฝน ซึ่งการปรับปรุงความแม่นยำข้อมูลของเทคนิค Bagging ใช้วิธีการฝึกฝนหลายโปรแกรมพร้อมกันด้วยชุดข้อมูลหลายชุด ซึ่งตรงข้ามกับวิธีการของเทคนิค Boosting เลือกใช้วิธีการฝึกฝนโปรแกรมเรียนรู้ที่อ่อนแอทีละโปรแกรม

องค์ประกอบหลักของ Gradient Boosting มีทั้งหมด 3 ส่วนได้แก่ Loss Function ทำหน้าที่คำนวณโมเดลในการทำนายเพื่อประมวลผล ส่วนที่สองคือ Weak Learner ทำหน้าที่ตัวช่วยจำแนกข้อมูลในการประมวลผล เพื่อลดข้อผิดพลาดในการทำงานของโมเดล และส่วนที่สามคือ Additive Model ทำหน้าที่ตัวบ่งชี้การเรียนรู้ของโมเดลให้มีการเรียนรู้อย่างต่อเนื่องและจัดการทีละขั้นตอน ซึ่งเป็นแนวทางแบบวนซ้ำและต่อเนื่องในการเพิ่มต้นไม้ในส่วนที่เป็นผู้เรียนอ่อนแอ เพื่อที่จะทำให้โมเดลมีความแม่นยำในการประมวลผลมากยิ่งขึ้น

เนื่องจากประสิทธิภาพการทำงานของอัลกอริทึม Gradient Boosting มีความแม่นยำและประสิทธิภาพสูง จึงได้มีการนำเทคนิคอัลกอริทึมชนิดนี้พัฒนาจนเป็นอัลกอริทึมใหม่ขึ้นมาเรียกว่า อัลกอริทึม Light Gradient Boosting Machine (LGBM)

อัลกอริทึม LGBM คืออัลกอริทึมชนิดหนึ่งใช้เทคนิคต้นไม้ตัดสินใจมาสร้างโมเดลและทำการสุ่มตัวแปรเพื่อวิเคราะห์ข้อมูล เป็นเฟรมเวิร์คที่มีประสิทธิภาพสูง LGBM พัฒนามาจากเทคนิค Gradient Boosting โดยถือว่าเป็นรูปแบบเทคนิคหนึ่งของการเรียนรู้ของเครื่องสำหรับการแก้ปัญหาการจำแนกประเภทและการถดถอย ในการประมวลผลการเรียนรู้และเทรนข้อมูลใช้วิธีการเรียกว่า Gradient-based One-Side Sampling

รูปแบบการทำงานของอัลกอริทึม LGBM ใช้หลักการทำงานของอัลกอริทึมต้นไม้ตัดสินใจ โดยสร้างต้นไม้หลาย ๆ ต้นมาเพื่อสร้างข้อมูลขึ้นมา และนำข้อมูลที่ถูกสร้างนี้มาใช้สอนโมเดล ทำให้โมเดลเกิดการเรียนรู้และสร้างต้นไม้จากการเรียนรู้ในแต่ละครั้ง ในขั้นตอนการทำงาน คืออัลกอริทึม LGBM จะทำการค้นหาตัวแปรต้นที่ส่งผลอย่างมีนัยสำคัญต่อค่าตัวแปรตามที่สนใจ และแยกส่วนระดับกิ่งของต้นไม้ เรียกว่า Level-wise ขณะเดียวกันอัลกอริทึม LGBM สามารถแยกความลึกของต้นไม้เพื่อเพิ่มระดับของต้นไม้ เรียกว่า Leaf-wise แสดงดังรูปที่ 2.7

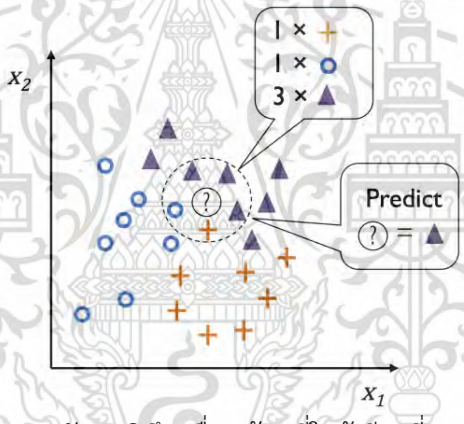


รูปที่ 2.7 ไดอะแกรมการแยกต้นไม้แบบ Level-wise และ Leaf-wise

ตามรูปที่ 2.7 จะเห็นได้ว่าโครงสร้างของอัลกอริทึม LGBM หลังจากทำการแยกกิ่งไม้แบบ Leaf-wise ของ มีความซับซ้อนมากกว่าแบบ Level-wise เนื่องจากความลึกของต้นไม้มากกว่า ทำให้เกิดความไม่พอดี (Overfitting) ของชุดข้อมูลได้ ถ้าหากมีการปรับค่าพารามิเตอร์ความลึกของอัลกอริทึมให้เหมาะสมกับชุดข้อมูล อัลกอริทึมช่วยทำให้การเติบโตของชุดข้อมูลลดการสูญเสียได้ และมีความแม่นยำในการทำนายมากกว่าได้มากกว่าอัลกอริทึมอื่น ส่วนใหญ่การแบ่งข้อมูลของอัลกอริทึม LGBM มักนิยมเลือกใช้เทคนิค K-Fold Cross-Validation พร้อมทั้งปรับค่าพารามิเตอร์ให้เหมาะสมเพื่อลดปัญหาการเกิด Overfitting

4) เพื่อนบ้านที่ใกล้เคียงที่สุด K ตัว (k-Nearest Neighbors) เป็นอัลกอริทึมประเภท Supervised Learning ซึ่งเหมาะสำหรับการแบ่งข้อมูลเป็นหมวดหมู่ข้อมูล โดยนำข้อมูลอื่น ๆ มาเปรียบเทียบกับตัวข้อมูลที่สนใจว่ามีความใกล้เคียงกันมากน้อยเพียงใด ถ้าหากข้อมูลที่สนใจนั้นมีความใกล้เคียงกับข้อมูลในชุดทดสอบใดมากที่สุด แสดงว่าข้อมูลนั้นอยู่ในกลุ่มชุดข้อมูลในชุดทดสอบเดียวกัน โดยส่วนใหญ่จะใช้สำหรับแก้ปัญหาจำนวนกลุ่มที่กำหนดแน่นอน แต่ถ้าหากมีข้อมูลบางตัวที่ไม่สามารถระบุได้ว่าข้อมูลนั้นอยู่ในกลุ่มใด เราจะสามารถนำมาจัดกลุ่มให้กับข้อมูลนั้นได้

หลักการทำงานของอัลกอริทึมเพื่อนบ้านที่ใกล้เคียงที่สุด k ตัว คือ การจัดกลุ่มหรือจำแนกประเภทข้อมูลใหม่ โดยอ้างอิงกับชุดข้อมูลที่อยู่ใกล้ที่สุดกับข้อมูลชุดทดสอบ เพื่อคาดเดาและจำแนกประเภทข้อมูลใหม่ อัลกอริทึมทำงานแบบไม่มีขั้นตอนการเรียนรู้ที่ซับซ้อน แต่จะใช้ข้อมูลที่มีอยู่ในการตัดสินใจ ซึ่งอัลกอริทึมนี้ถือว่าเป็น Instance-Based Learning ชนิดหนึ่ง ซึ่งหลักการทำงานของอัลกอริทึมประเภทเพื่อนบ้านที่ใกล้เคียงที่สุด k ตัว แสดงดังรูปที่ 2.8



รูปที่ 2.8 อัลกอริทึมเพื่อนบ้านที่ใกล้เคียงที่สุด k ตัว

จากรูปที่ 2.8 หลักการทำงานของอัลกอริทึมการจำแนกเพื่อนบ้านที่ใกล้ที่สุดเป็นลักษณะ Majority Voting โดยพิจารณาจากระยะที่ใกล้ที่สุดระหว่างข้อมูลทดสอบและข้อมูลในชุดข้อมูลการฝึกฝน เรียกว่า Euclidean Distance ระยะนี้วัดความคล้ายคลึงของคุณสมบัติระหว่างข้อมูล จากนั้นระบุจำนวนโหนด k รายการที่ใกล้ที่สุด เพื่อนับจำนวนรายการในแต่ละกลุ่มหรือประเภทข้อมูล และกำหนดกลุ่มหรือประเภทข้อมูลของข้อมูลทดสอบตามจำนวนมากที่สุดในโหนด k เพื่อที่จะสามารถทำนายผลลัพธ์ของข้อมูลทดสอบเหล่านี้ได้อย่างมีประสิทธิภาพ การกำหนดเลือกค่า k ถือว่าเป็นปัจจัยสำคัญต่อการจำแนกชุดข้อมูล ถ้าหากชุดข้อมูลในการฝึกฝนนั้นมีความซับซ้อน แนวโน้มของข้อมูลมีการเอียงในการจัดกลุ่มเพิ่มขึ้น ค่า k ควรจะมีค่าน้อยเพื่อลดความซับซ้อนและเพิ่มประสิทธิภาพในการคำนวณ ส่วนในกรณีที่มีจำนวนข้อมูลในชุดการฝึกฝนน้อย ค่า k ควรจะมากขึ้นเพื่อป้องกันการเอียงในการจำแนกประเภทของชุดข้อมูล

2.3.2 การเรียนรู้แบบไม่มีผู้สอน (Unsupervised Learning)

การเรียนรู้แบบไม่มีผู้สอน (Unsupervised Learning) คือ อัลกอริทึมที่ตรวจสอบเฉพาะข้อมูลที่ป้อนเข้ามาเท่านั้น โดยปราศจากผลลัพธ์ที่เกิดขึ้น หลักการใช้งานสำหรับอัลกอริทึมการเรียนรู้แบบไม่มีผู้สอน เพื่อหารูปแบบและการแบ่งประเภทกลุ่มของข้อมูล (Clustering) ลักษณะการใช้งานเหมาะกับการสำรวจ และนำข้อมูลมาจัดเป็นกลุ่ม โดยข้อมูลที่อยู่ในกลุ่มเดียวกันจะมีความสัมพันธ์หรือมีลักษณะที่คล้ายคลึงกัน ตัวอย่างโมเดลการเรียนรู้แบบไม่มีผู้สอนได้แก่

1) การแบ่งข้อมูลแบบเคมีน (K-Means Clustering) หรือเรียกอีกอย่างหนึ่งว่า การวิเคราะห์กลุ่มแบบไม่เป็นขั้นตอน (Nonhierarchical Cluster Analysis) เป็นการกำหนดข้อมูลที่มีความแปรปรวนเท่ากันจัดอยู่ในกลุ่มเดียวกัน เทคนิคนี้สามารถปรับนำมาใช้กับข้อมูลตัวอย่างจำนวนมากได้ดี โดยกำหนดค่า k แทนค่าจำนวนกลุ่ม หลังจากนั้นนำข้อมูลทั้งหมดจัดเข้ากลุ่ม โดยทำการคำนวณระยะห่างจากตำแหน่งของข้อมูลและจุดศูนย์กลางด้วยวิธีการหาจุดศูนย์กลางของแต่ละกลุ่ม สามารถคำนวณได้จากสมการที่ (2.5)

$$\text{Distance} = \sqrt{(X_1 - X_2)^2 + (Y_1 - Y_2)^2} \qquad (2.5)$$

วิธีการหาจุดศูนย์กลางของแต่ละกลุ่ม เริ่มจากกำหนดค่า k คือจำนวนกลุ่ม และกำหนดจุดศูนย์กลางเริ่มต้น k จุด จากนั้นนำวัตถุทั้งหมดจัดเข้ากลุ่ม โดยทำการหาค่าระยะห่างระหว่างข้อมูลกับจุดศูนย์กลาง เพื่อนำข้อมูลมาจัดกลุ่ม (Cluster) กับจุดศูนย์กลางที่ใกล้ที่สุด และหาค่าเฉลี่ยของแต่ละกลุ่ม เพื่อกำหนดเป็นจุดศูนย์กลางใหม่ และทำการคำนวณหาค่าระยะห่างใหม่อีกครั้งจนกว่าค่าเฉลี่ยหรือจุดศูนย์กลางในแต่ละกลุ่มไม่มีการเปลี่ยนแปลง

ข้อดีของการแบ่งข้อมูลแบบเคมีน เหมาะสำหรับการแบ่งข้อมูลจำนวนกลุ่มน้อย แต่ไม่จำกัดปริมาณข้อมูลที่นำมาใช้ ส่วนข้อจำกัดของการแบ่งข้อมูลแบบเคมีนคือ การกำหนดค่าจำนวนกลุ่ม เป็นได้ยากและประมวลผลกับกลุ่มข้อมูลที่ไม่เป็นลักษณะวงกลม รวมถึงเรื่องของขนาด ความหนาแน่น และรูปร่างของกลุ่มข้อมูล

2) การแบ่งกลุ่มข้อมูลตามลำดับชั้น (Hierarchical Clustering) เป็นการวิเคราะห์กลุ่มแบบลำดับชั้น เป็นการจัดกลุ่มโดยไม่ต้องมีการกำหนดจำนวนกลุ่ม หลักการงานของการแบ่งกลุ่มข้อมูลตามลำดับชั้นเริ่มจากคำนวนระยะห่างจากจุด Centroid ของแต่ละจุดข้อมูล โดยใช้สูตรเดียวกับที่ใช้คำนวนในการแบ่งข้อมูลแบบเคมีน แล้วนำข้อมูลเหล่านั้นมาทำเป็นตารางแล้วรวมเอาแถวและคอลัมพ์ที่มีค่าน้อยที่สุดรวมกันเป็นกลุ่มข้อมูลเดียวกัน หลังจากนั้นนำข้อมูลที่เหลือมาทำซ้ำหลาย ๆ รอบจนสามารถจัดข้อมูลทุกตัวเข้าไปในกลุ่มเดียวกัน

3) อัลกอริทึมสกัดคุณลักษณะ (Principal Component Analysis) เป็นอัลกอริทึมที่ใช้เทคนิคประเภท Unsupervised Learning ในการแบ่งแยกคุณลักษณะจากชุดข้อมูลเดิมเหมาะสำหรับที่ใช้ในการลดมิติของข้อมูล (Dimension Reduction) โดยค้นหาความแปรปรวนของ

คุณลักษณะมากที่สุด เพื่อกำหนดตัวแปรใหม่ (Principal Components) ของข้อมูล หลักการวิธีการสร้างตัวแปรใหม่นั้นเริ่มต้นจากการสร้างตัวแปรขึ้นเป็น Linear Combinations เป็นตัวแปรต้น ซึ่งในตัวแปรแต่ละตัวนั้นจะไม่มีความสัมพันธ์กัน และพิจารณาจากตัวแปรที่สามารถอธิบายความแปรปรวนคุณลักษณะที่ได้มากที่สุดตัวแปรนั้นจะถูกจัดเรียงในคุณลักษณะที่เรียกว่า First Principal Component ส่วนตัวแปรที่จัดเป็นลำดับถัดมา เรียกว่าเป็น Second Principal Component ซึ่งคำนวณในลักษณะเดียวกัน โดยมีเงื่อนไขว่า จะต้องไม่มีความสัมพันธ์กันกับ First Principal Component

2.3.3 การเรียนรู้แบบเสริมกำลัง (Reinforcement Learning)

การเรียนรู้แบบเสริมกำลัง (Reinforcement Learning) คือ อัลกอริทึมทำงานลักษณะคล้ายกับการที่มนุษย์เรียนรู้บางสิ่งบางอย่างด้วยวิธีการลองผิดลองถูก และมีระบบการเรียนรู้เกิดขึ้นระหว่างทางว่าการกระทำแบบใดดีหรือไม่ดี การเรียนรู้แบบเสริมกำลังเป็นวิธีการเรียนรู้รูปแบบหนึ่งที่เกิดมาจากการปฏิสัมพันธ์ระหว่างผู้เรียนรู้กับสิ่งแวดล้อม องค์ประกอบหลักของอัลกอริทึมการเรียนรู้แบบเสริมกำลัง องค์ประกอบหลักของอัลกอริทึมการเรียนรู้แบบเสริมกำลัง ได้แก่

1) ผู้กระทำ (Agent)

2) การกระทำ (Action) ของผู้กระทำที่มีผลกระทบต่อระบบสภาพแวดล้อม

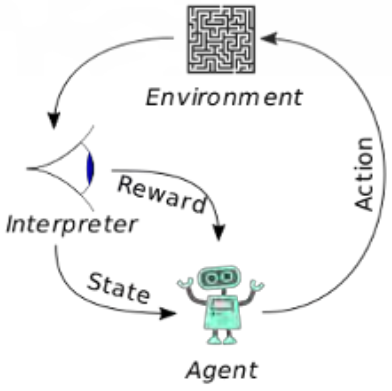
3) ระบบสภาพแวดล้อม (Environment) คือ ระบบปฏิสัมพันธ์ที่ผู้กระทำมีส่วนเกี่ยวข้อง

4) สถานการณ์ของระบบสภาพแวดล้อมที่ทางผู้กระทำสามารถรับรู้ได้ (State)

5) หลักการที่ผู้กระทำเลือกใช้ในการตัดสินใจเลือกเพื่อกระทำ (Policy) หลังจากประเมินสถานการณ์แล้ว

6) ตัวประเมินผลลัพธ์ที่เกิดจากการกระทำของผู้กระทำ (Reward)

โดยทั่วไปหลักการทำงานของการเรียนรู้แบบเสริมกำลัง มีลักษณะรูปแบบดังรูปที่ 2.9

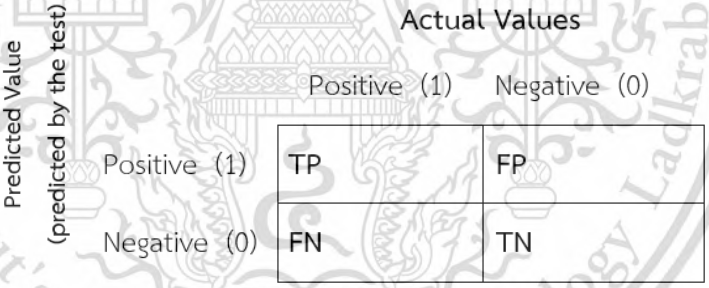


รูปที่ 2.9 หลักการทำงานของการเรียนรู้แบบเสริมกำลัง

จากรูปที่ 2.9 แสดงหลักการทำงานของการเรียนรู้แบบเสริมกำลัง คือ การเรียนรู้ของผู้กระทำที่เกิดจากปฏิสัมพันธ์แบบลองผิดลองถูกระหว่างผู้กระทำ (Agent) กับระบบสภาพแวดล้อม (Environment) โดยผู้กระทำสามารถรับรู้สถานการณ์ของระบบสภาพแวดล้อมโดยผ่านสถานะ (State) จากนั้นให้ทำการเลือกวิธีการกระทำที่ส่งผลต่อระบบ โดยหวังว่าจะได้ผลลัพธ์ที่ดีที่สุด รวมทั้งผลของการเรียนรู้จากข้อผิดพลาดที่เกิดขึ้นในอดีต

2.4 การประเมินประสิทธิภาพการจำแนกประเภท (Classification Performance Evaluation)

วิธีการประเมินประสิทธิภาพการจำแนกประเภทของอัลกอริทึมประเมินผล ได้ทั้งหมด 3วิธี ได้แก่ 1) การวัดค่าความถูกต้องของการเรียนรู้ของเครื่องแบบโดยรวม (Accuracy) 2) การวัดค่าความแม่นยำ (Precision) และ 3) การวัดค่าความระลึกได้ (Recall) นำผลลัพธ์ที่ได้จากกระบวนการผลิตมาสร้างเป็นตารางวัดประสิทธิภาพโมเดลของอัลกอริทึม (Confusion Matrix) ว่ามีประสิทธิภาพเพียงพอที่จะนำมาพัฒนาหรือนำไปใช้งานได้ผลที่ดีที่สุด หลักการวัดประสิทธิภาพจะวัดค่าจากในตารางข้อมูล ดังรูปที่ 2.10



รูปที่ 2.10 การวัดประสิทธิภาพโมเดลของอัลกอริทึม (Confusion Matrix)

จากรูปที่ 2.10 กำหนดให้ สิ่งที่ทำนายตรงกับสิ่งที่เกิดขึ้นจริง ในกรณีทำนายว่าจริง และสิ่งที่เกิดขึ้น คือ เป็นจริง ซึ่งมีค่าเป็น “True Positive (TP)”

สิ่งที่ทำนายตรงกับสิ่งที่เกิดขึ้น ในกรณี ทำนายว่า ไม่จริง และสิ่งที่เกิดขึ้นก็คือ ไม่จริง ซึ่งมีค่าเป็น “True Negative (TN)”

สิ่งที่ทำนายไม่ตรงกับสิ่งที่เกิดขึ้น คือทำนายว่า จริง แต่สิ่งที่เกิดขึ้น คือ ไม่จริง ซึ่งมีค่าเป็น “False Positive (FP)

สิ่งที่ทำนายไม่ตรงกับที่เกิดขึ้นจริง คือทำนายว่าไม่จริง แต่สิ่งที่เกิดขึ้น คือ จริง ซึ่งมีค่าเป็น “False Negative (FN)”

2.4.1 ค่าความถูกต้อง (Accuracy)

การวัดค่าความถูกต้อง คือ การตรวจสอบว่าผลลัพธ์ที่ได้จากระบบ ตรงกับผลผลิตที่ได้ตามเป้าหมายหรือไม่ โดยคิดจากสมการที่ (2.6)

$$\text{Accuracy} = \frac{\mathbf{TP+TN}}{\mathbf{TP+NP+FP+FN}} \quad (2.6)$$

2.4.2 ค่าความแม่นยำ (Precision)

การวัดค่าความแม่นยำ คือ การตรวจสอบว่าผลลัพธ์ที่ได้จากระบบทั้งหมดตรงกับผลผลิตมากน้อยเพียงใด หาจากสัดส่วนของผลผลิตที่ผลิตได้ตรงกับเป้าหมายผลผลิตที่ถูกกำหนดไว้ต่อผลผลิตทั้งหมด โดยคิดจากสมการที่ (2.7)

$$\text{Precision} = \frac{\mathbf{TP}}{\mathbf{TP+FP}} \quad (2.7)$$

2.4.3 ค่าความระลึกได้ (Recall)

ค่าความระลึกได้ คือ การตรวจสอบว่าข้อมูลที่วางแผนไว้ในกระบวนการผลิตตรงกับผลผลิตที่ผลิตได้ในปัจจุบัน ต่อจำนวนปริมาณผลผลิตทั้งหมด โดยคิดจากสมการที่ (2.8)

$$\text{Recall} = \frac{\mathbf{TP}}{\mathbf{TP+FN}} \quad (2.8)$$

2.5 งานวิจัยที่เกี่ยวข้อง

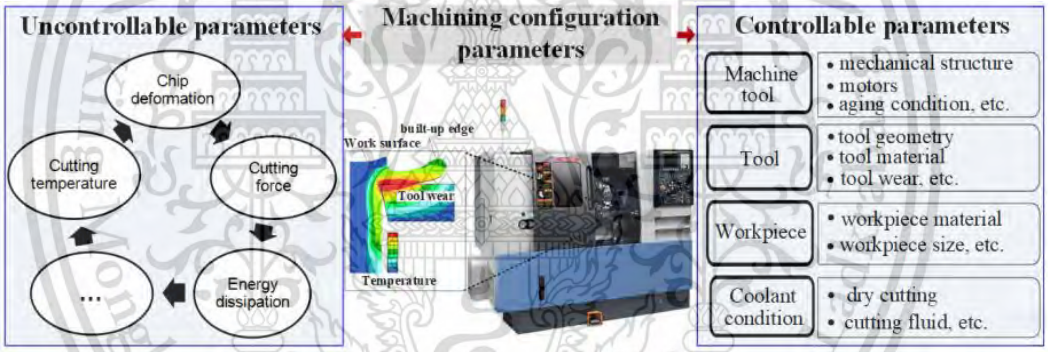
การศึกษาเรื่อง การจำแนกประเภทความผิดปกติของเครื่องจักรจากข้อมูลการผลิต เป็นประเด็นที่น่าสนใจในอุตสาหกรรมการผลิตเป็นอย่างมาก จากการศึกษางานวิจัยที่เกี่ยวข้องในครั้งนี้สามารถสรุปได้ดังนี้

2.5.1 การบำรุงเชิงคาดการณ์ของเครื่องจักรในกระบวนการผลิต

Qinge Xiao และคณะ (2021) นำเสนอวิธีการเรียนรู้ของเครื่อง เพื่อจำลองปัญหาการบำรุงรักษาเครื่องจักรในกระบวนการผลิต และเพิ่มประสิทธิภาพของเครื่องจักรในกระบวนการผลิต ได้แก่ พลังงานที่สูญเสียทั้งหมด อายุการใช้งานของเครื่องจักรและปัญหาเครื่องจักรขัดข้อง เป็นต้น โดยใช้วิธีการจำแนกกลุ่มหลายประเภท (Multiple Classification Method)

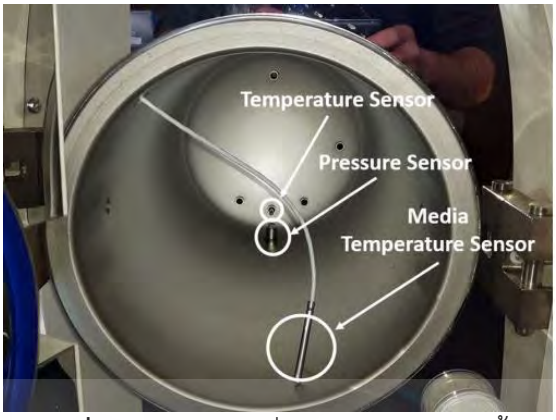
การแก้ปัญหาการบำรุงรักษาเชิงคาดการณ์ ผู้วิจัยนำเสนอเทคโนโลยีการเรียนรู้ของเครื่องแบบการเรียนรู้แบบมีผู้สอน โดยใช้อัลกอริทึมการจำแนกกลุ่มหลายประเภท ที่รู้จักและใช้กันอย่างแพร่หลายเพื่อนำมาเปรียบเทียบประสิทธิภาพของเครื่องจักร และแก้ไขปัญหาให้เครื่องจักรสามารถทำงานได้อย่างต่อเนื่อง

จากการสำรวจเครื่องจักร ผู้วิจัยพบว่าปัจจัยหลายประการที่อาจส่งผลเครื่องจักรสูญเสียพลังงานในการผลิต เช่น อุณหภูมิ ความร้อนสะสมที่เกิดขึ้นของเครื่องจักร แรงดันไฟฟ้า ความเร็วมอเตอร์ เป็นต้น นำไปสู่การบำรุงรักษาเชิงคาดการณ์ (Predictive Maintenance) ซึ่งเป็นกลยุทธ์ที่ใช้ในการจัดการกับปัญหาการบำรุงรักษาและการตัดสินใจเพื่อประเมินประสิทธิภาพพลังงานของเครื่องจักร อีกทั้งสามารถใช้กับปัญหาข้อมูลที่มีความซับซ้อน ลักษณะมิติข้อมูลสูง รวมถึงปัญหาของอุปกรณ์ประกอบของเครื่องจักรได้ เพื่อป้องกันการเกิดปัญหาเครื่องจักรหยุดทำงานโดยไม่คาดคิด และเพิ่มประสิทธิภาพพลังงานให้เครื่องจักร โดยพิจารณาจากพารามิเตอร์ที่ถูกตั้งค่าในเครื่องจักร (Machining Configuration Parameters) แสดงรูปที่ 2.11



รูปที่ 2.11 พารามิเตอร์ของเครื่องจักร

Musagil Musabayli และคณะ (2020) ได้ทำการศึกษาค้นคว้าบทความงานวิจัยเกี่ยวกับการพัฒนาและปรับปรุงประสิทธิภาพเครื่องจักรในกระบวนการผลิต โดยยกตัวอย่างข้อมูลของเครื่องจักรในกระบวนการผลิตหม้อแรงดันไอน้ำ (Steam Sterilizer) เป็นหม้อแรงดันที่ใช้ในกระบวนการผลิตในอุตสาหกรรม เริ่มต้นจากทำการเก็บข้อมูลจากอุปกรณ์เซนเซอร์ต่าง ๆ จากหม้อแรงดันไอน้ำสุญญากาศและชุดหม้อแปลงแรงดันไอน้ำ ถูกบันทึกข้อมูลไว้เป็นจำนวนมากกว่า 1,000 ข้อมูล และนำข้อมูลมาทำนายและวัดประสิทธิภาพการทำงานของเครื่องจักร โดยนำข้อมูลเหล่านี้มาเรียนรู้กับอัลกอริทึมต่าง ๆ ในชนิดจำแนกประเภท มีทั้งหมด 5 วิธี ได้แก่ ซัพพอร์ตเวคเตอร์แมชชีน อัลกอริทึม k-Nearest Neighbours อัลกอริทึม Decision Tree อัลกอริทึม Random Forest และอัลกอริทึม Logistic Regression เพื่อวางแผนเป็นแนวทางในการปรับปรุงระบบบำรุงรักษาเครื่องจักร ในการทดลองผู้วิจัยเลือกลักษณะเซนเซอร์ที่ใช้ในหม้อแรงดันไอน้ำเป็นดังรูปที่ 2.12



รูปที่ 2.12 เซนเซอร์ที่ใช้ในหม้อแรงดันไอน้ำ

จากรูปที่ 2.12 ลักษณะข้อมูลที่นำมาใช้กับอัลกอริทึมได้จากเซนเซอร์ชนิดต่าง ๆ ได้แก่ เซนเซอร์วัดอุณหภูมิ เซนเซอร์วัดแรงดันไอน้ำ ข้อมูลถูกแบ่งออกเป็น 2 ส่วนคือ ข้อมูลที่บันทึกได้จากชุดหม้อแรงดันไอน้ำและชุดข้อมูลที่ได้จากชุดหม้อแปลงแรงดันไอน้ำ โดยวิธีการเก็บข้อมูลเริ่มตั้งแต่ขั้นตอนเครื่องเริ่มทำงาน ก่อนเกิดสุญญากาศขณะที่อุณหภูมิเพิ่มขึ้น การฆ่าเชื้อ และการทำให้แห้งด้วยสุญญากาศ ซึ่งแต่ละขั้นตอนจะมีการเปลี่ยนแปลงอุณหภูมิที่แตกต่างกัน จากนั้นทำการเตรียมข้อมูล (Data Processing) โดยทำการ Rescale ข้อมูลและการทำ Aggregation เพื่อลดมิติข้อมูล รวมถึงการแก้ไขข้อมูล (Data Interpolation) เพื่อเปลี่ยนลักษณะข้อมูลให้เหมาะสมก่อนนำมาใช้กับอัลกอริทึม จากนั้นนำข้อมูลทั้งหมดมาใช้กับอัลกอริทึมสำหรับเรียนรู้และฝึกฝนกับโมเดลที่เลือกไว้ เพื่อทำนายและวัดประสิทธิภาพของหม้อแรงดันไอน้ำ โดยทำการวัดจากค่าความแม่นยำ

หลังจากทำการทดลองผู้วิจัยได้พบว่า การใช้อัลกอริทึม Random Forest ทำให้ค่าวัดจากผลรวมของประสิทธิภาพการทำงานของเครื่องจักรดีที่สุดทั้งสองชุดข้อมูล สามารถสรุปได้ว่าในชุดข้อมูลหม้อแรงดันไอน้ำอัลกอริทึม Random Forest มีค่า accuracy เท่ากับ 83.5% อัลกอริทึม Logistic Regression มีค่า accuracy เท่ากับ 82.0% อัลกอริทึมซัพพอร์ตเวคเตอร์แมชชีนมีค่า accuracy เท่ากับ 80.0% อัลกอริทึม Decision Tree มีค่า accuracy เท่ากับ 79.5% และอัลกอริทึม k-Nearest Neighbours มีค่า 65.0% ส่วนในชุดข้อมูลหม้อแปลงแรงดันไอน้ำ อัลกอริทึม Random Forest มีค่า accuracy เท่ากับ 82.0% อัลกอริทึม Logistic regression มีค่า accuracy เท่ากับ 79.5% อัลกอริทึมซัพพอร์ตเวคเตอร์แมชชีนมีค่า accuracy เท่ากับ 82.0% อัลกอริทึม Decision Tree มีค่า accuracy เท่ากับ 81.0% และอัลกอริทึม k - Nearest Neighbours มีค่า 69.0%

I. EL HASSANI และคณะ (2019) ได้ทำการศึกษาค้นคว้าบทความงานวิจัยเกี่ยวกับการทำนายและประเมินประสิทธิภาพของเครื่องจักรในโรงงานอุตสาหกรรม เพื่อลดปัญหาเครื่องจักรทำงานขัดข้องในระหว่างการผลิต และเพิ่มผลผลิตให้มีปริมาณเพียงพอต่อความต้องการของกลุ่มลูกค้า

ในงานวิจัยได้เริ่มจากการศึกษาและพิจารณาตัวชี้วัด (Key Performance Indicators: KPI) ที่เกี่ยวข้อง ได้แก่ ประสิทธิผลโดยรวมของเครื่องจักร (Overall Equipment Effectiveness: OEE) ซึ่งตัวชี้วัดประสิทธิภาพที่ใช้ในการผลิต สามารถระบุปัญหาของเครื่องจักรที่เกิดขึ้นได้ และนำข้อมูลตรงนี้ไปประมวลผล เพื่อเป็นแนวทางในการตัดสินใจที่จำเป็น และปรับปรุงการทำงานของเครื่องจักรให้ทำงานได้อย่างมีประสิทธิภาพได้ดียิ่งขึ้น จากนั้นนำข้อมูลเหล่านี้มาใช้กับอัลกอริทึมการเรียนรู้ของเครื่องเพื่อปรับปรุงเครื่องจักรในกระบวนการผลิต มีทั้งหมด 4 วิธี ได้แก่ ซัพพอร์ตเวคเตอร์แมชชีน อัลกอริทึม Random Forest อัลกอริทึม Extreme Gradient Boosting และอัลกอริทึมประเภท Deep Neural Networks ทดสอบกับข้อมูลการผลิตในกระบวนการ ได้แก่ ข้อมูลตั้งค่าสำหรับเครื่องจักร (Setups) ข้อมูลความผิดพลาดการทำงานของเครื่องจักร (Breakdown) จำนวนยอดผลิต (Order) ความยาวสายไฟ (Wire Length) จำนวนเครื่องที่ผลิต (Number of Terminals) รวมถึงจำนวนการบรรจุ โดยใช้วิธีการลดคุณลักษณะร่วมกับอัลกอริทึมเครื่องจักรการเรียนรู้ พบว่า การใช้อัลกอริทึม Deep Neural Networks และอัลกอริทึม Random Forest ทำให้ค่าวัดที่ได้จากผลรวมของค่า OEE เฉลี่ยสูงสุด สามารถสรุปได้แก่ อัลกอริทึม Deep Neural Networks มีค่า MAE เท่ากับ 6.27% MAPE 11.76% อัลกอริทึม Random Forest มีค่า MAE เท่ากับ 6.83% MAPE 13.59% อัลกอริทึม Extreme Gradient Boosting มีค่า MAE เท่ากับ 6.43% MAPE 12.59% และอัลกอริทึมซัพพอร์ตเวคเตอร์แมชชีนมีค่า MAE เท่ากับ 6.16% MAPE 12.12%

บทที่ 3

วิธีการดำเนินงานวิจัย

เนื้อหาบทนี้อธิบายถึงขั้นตอนการวิจัย โดยเริ่มจากขั้นตอนการเลือกชุดข้อมูลที่นำมาใช้ในการทดลอง อัลกอริทึมที่ใช้ในการจำแนกประเภทของข้อมูล วิธีการทำนายความแตกต่างของสถานะการทำงานของเครื่องจักรที่ใช้ในการผลิต วิธีวัดความถูกต้องในการทำนายความแตกต่างของสถานะการทำงานของเครื่องจักร เครื่องมือที่ใช้ในการทดลอง และการออกแบบกระบวนการทดลอง

3.1 ขั้นตอนของการวิจัย

ขั้นตอนของการวิจัยแบ่งออกเป็น 4 ขั้นตอน ได้แก่ 1) ขั้นตอนการเตรียมข้อมูล 2) อัลกอริทึมที่ใช้ในการแบ่งแยกประเภทของข้อมูล 3) วิธีการทำนายความแตกต่างของสถานะการทำงานของเครื่องจักรที่ใช้ในการผลิต 4) การวัดผลการทดลอง (Evaluation) เป็นส่วนที่ใช้ตรวจสอบว่าการทำนายสถานะการทำงานของอุปกรณ์แต่ละชนิด มีความถูกต้องและแม่นยำหรือไม่

3.1.1 ขั้นตอนการเตรียมข้อมูล

ในขั้นตอนการเตรียมข้อมูลของการศึกษาค้นคว้าอิสระนี้เป็นขั้นตอนที่เลือกข้อมูลต่าง ๆ ของเครื่องจักรที่ใช้ในการทดลองที่สนใจ โดยได้ทำการเก็บข้อมูลเกี่ยวกับผลิตภัณฑ์ ชนิดของสินค้า ขนาดสินค้า รวมถึงค่าที่วัดได้ของอุปกรณ์แต่ละชนิดของเครื่องจักรในกระบวนการผลิต โดยชุดข้อมูลที่นำมาทดลองได้จากการจำลองชุดข้อมูล (Simulation Data) จากเครื่องจักรที่ใช้ในกระบวนการผลิต ดังนั้นผู้วิจัยได้ศึกษาและหาชุดข้อมูลที่สามารถนำไปประยุกต์ใช้กับการทดลอง ซึ่งเป็นกระบวนการผลิตสินค้าของโรงงานแห่งหนึ่ง มีข้อมูลสินค้าแบ่งออกเป็น 3 ชนิด ได้แก่ ขนาดเล็ก กลาง และใหญ่ และมีรหัสสินค้าที่แตกต่างกันทั้งหมด 10,000 ข้อมูล นอกจากนี้ยังมีข้อมูลของเครื่องมือวัดอุณหภูมิที่วัดได้ ความดันของเครื่องจักร ความเร็วรอบมอเตอร์ (Rotary Speed) และแรงบิด (Torque) ของมอเตอร์ ทั้งหมดที่เกิดขึ้นในแต่ละช่วงเวลานำมาใช้ในการวิเคราะห์ข้อมูลเกี่ยวกับสถานะการทำงานของเครื่องจักรที่ใช้ในการผลิตเช่นเดียวกัน แหล่งที่มาของชุดข้อมูลกลุ่มตัวอย่างในงานวิจัยเล่มนี้ได้มาจากแหล่งข้อมูลบนเว็บไซต์ Machine Learning Repository ของมหาวิทยาลัยแคลิฟอร์เนียเออร์ไวน์ (University of California Irvine: UCI) เป็นชุดข้อมูลเกี่ยวกับการบำรุงเชิงรักษาของเครื่องจักรในโรงงานแห่งหนึ่ง (AI4I 2020 Predictive Maintenance Dataset) นำมาทดลองและวิเคราะห์ข้อมูลเพื่อหาความผิดปกติสถานะการทำงานของเครื่องจักร

ในชุดข้อมูลกลุ่มตัวอย่าง กำหนดคุณลักษณะสถานะความผิดปกติการทำงานของเครื่องจักร (Failure Type) โดยระบุไว้ มีทั้งหมด 5 ประเภท ได้แก่

1) ความผิดปกติที่เกิดจากความร้อนสะสม (Heat Dissipation Failure) เป็นความล้มเหลวของกระบวนการการผลิตที่เกิดจากความร้อนสะสม โดยขีดจำกัดความร้อนสะสมที่เกิดขึ้นนั้นเป็นความแตกต่างระหว่างอุณหภูมิของอากาศและอุณหภูมิของกระบวนการต่ำกว่า 8.6 องศาเคลวิน (K) และความเร็วในการหมุนของเครื่องมือต่ำกว่า 1380 รอบต่อนาที

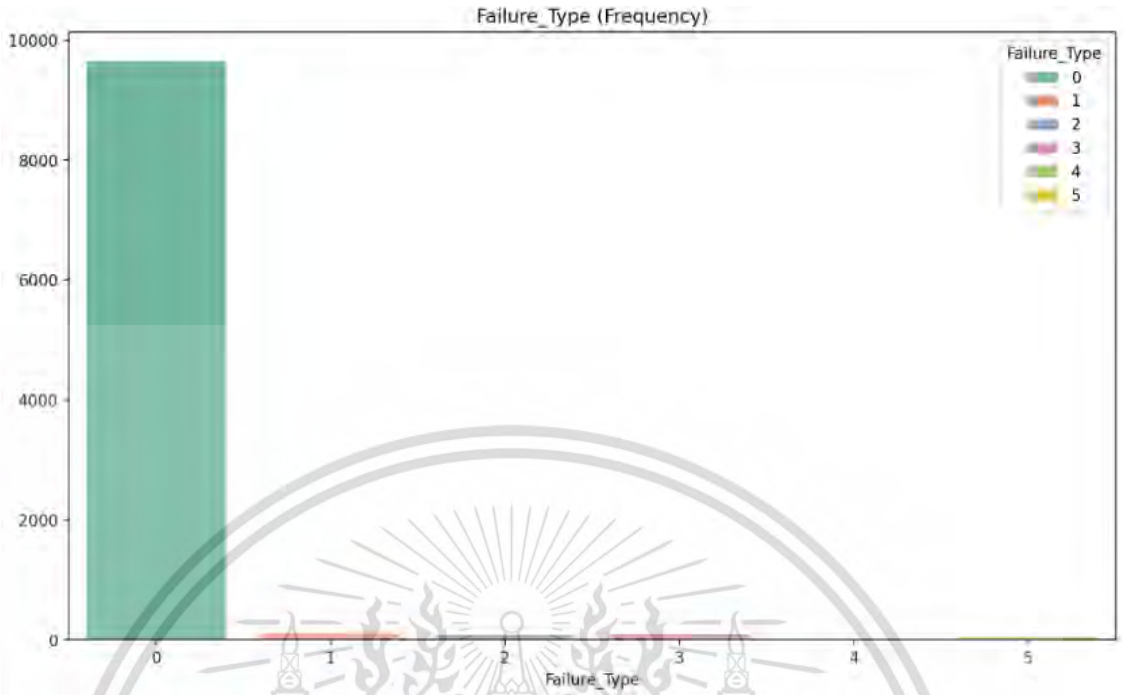
2) ความผิดปกติเนื่องจากเกินกำลัง (Overstrain Failure) เป็นความล้มเหลวที่เกิดจากความสึกหร่อเนื่องจากแรงบิดของเครื่องจักร ซึ่งถ้าหากเครื่องจักรเกิดแรงบิดมากกว่า 11 ,000 minNm ในกระบวนการผลิตสินค้าชนิด L จะทำให้เครื่องจักรทำงานขัดข้องเนื่องจากมีการทำงานเกินกำลังในการผลิต

3) ความผิดปกติที่เกิดจากพลังงานไฟฟ้ากำลัง (Power Failure) ซึ่งคิดจากผลคูณของแรงบิดและความเร็วในการหมุน หน่วย rad/s ถ้าหากพลังงานนี้ต่ำกว่า 3,500 วัตต์หรือสูงกว่า 9,000 วัตต์ ถือว่าเกิดความล้มเหลวการทำงานของเครื่องจักร จากการบันทึกข้อมูลในกรณีนี้มีเหตุการณ์เกิดขึ้นทั้งหมด 95 ครั้ง

4) ความผิดปกติเนื่องจากการความไม่แน่นอน (Random Failure) ในแต่ละกระบวนการผลิตจะมีโอกาสที่เกิดข้อผิดพลาดโดยไม่เกี่ยวข้องกับค่าพารามิเตอร์ที่กำหนดไว้ในเครื่องจักร ซึ่งมีความเป็นไปได้เพียง 0.1% ในชุดข้อมูลทั้งหมดมีความล้มเหลวเนื่องจากความไม่แน่นอนเพียง 5 จุดเท่านั้น

5) ความผิดปกติที่เกิดจากการสึกหร่อของเครื่องจักร (Tool Wear Failure) เมื่อเครื่องจักรกำลังทำงาน เครื่องมือจะถูกแทนที่ด้วยความล้มเหลวในเวลาสึกหร่อของเครื่องมือ โดยจำนวนครั้งในการสุ่มมี 120 ครั้งในระยะเวลา 200 – 240 นาที แต่ละครั้งในการสุ่มตรวจพบว่าเครื่องมือจะถูกแทนที่ 69 ครั้ง และล้มเหลว 51 ครั้ง

จากการศึกษาข้อมูลสถาะนะความผิดปกติการทำงานของเครื่องจักรที่นำมาทดลองได้จาก UCI พบว่า ในชุดข้อมูลกลุ่มตัวอย่างมีข้อจำกัดในด้านการคำนวณหาความผิดปกติการทำงานของเครื่องจักรในกระบวนการผลิต ทำให้ไม่สามารถนำข้อมูลไปวิเคราะห์ได้ทันที เนื่องจากจำนวนของตัวแปรเป้าหมาย (Failure Type) ที่มีไม่เท่ากัน อาจจะส่งผลทำให้เกิดความไม่สมดุลของข้อมูล (Imbalanced Data) ดังรูปที่ 3.1



รูปที่ 3.1 จำนวนครั้งของสถานะความผิดปกติการทำงานของเครื่องจักร

จากรูปที่ 3.1 ในการสำรวจข้อมูลกลุ่มตัวอย่าง ข้อมูลที่ได้จากการจำลองของเครื่องจักรจำนวน 10,000 ข้อมูล พบว่า ข้อมูลสถานะการทำงานของเครื่องจักรที่ปกติ มีจำนวน 9,652 สถานะ คิดเป็นร้อยละ 96.52 ของข้อมูลทั้งหมด ในส่วนจำนวนข้อมูลสถานะความผิดปกติการทำงานของเครื่องจักรที่เกิดขึ้นโดยรวม มีจำนวน 348 ข้อมูล คิดเป็นร้อยละ 3.48 ของจำนวนข้อมูลทั้งหมด และนอกจากนี้ในการสำรวจกลุ่มตัวอย่าง พบว่า จำนวนข้อมูลของสถานะความผิดปกติการทำงานของเครื่องจักรแต่ละประเภท โดยสรุปได้ตามตารางที่ 3.1

ตารางที่ 3.1 จำนวนข้อมูลสถานะความผิดปกติการทำงานของเครื่องจักรแต่ละประเภท

สถานะการทำงานของเครื่องจักร (Machine Failure Type)	**จำนวน (สถานะ)**	**ร้อยละ (%)**
เครื่องจักรทำงานปกติ (No Failure)	9652	96.52
ความผิดปกติที่เกิดจากความร้อนสะสม (Heat Dissipation Failure)	112	1.12
ความผิดปกติที่เกิดจากผลิตเกินกำลัง (Overstrain Failure)	78	0.78
ความผิดปกติที่เกิดจากไฟฟ้ากำลัง (Power Failure)	95	0.95
ความผิดปกติจากความไม่แน่นอนของเครื่องจักร (Random Failure)	18	0.18
ความผิดปกติเนื่องจากการสึกหร่อของเครื่องจักร (Tool Wear Failure)	45	0.45

จากตารางที่ 3.1 ในการสำรวจข้อมูลสถานะความผิดปกติการทำงานของเครื่องจักร แสดงให้เห็นว่า ตัวแปรเป้าหมายในกลุ่มตัวอย่างเป็นข้อมูลที่ไม่สมดุล (Imbalanced Data) ทำให้ความแม่นยำในการวิเคราะห์ข้อมูลน้อยลง และเกิดความคลาดเคลื่อนในการประมวลผลของโมเดลได้ ในการทดลองครั้งนี้ได้เริ่มจากการสำรวจค่าสถิติเชิงบรรยายของชุดข้อมูลก่อน เพื่อดูแนวโน้มการกระจายตัวของข้อมูล (Data Distribution) และรูปร่างของการแจกแจงของชุดข้อมูลทั้งหมด โดยทำการสำรวจชุดข้อมูลเพื่อนำไปสู่การเตรียมชุดข้อมูล ซึ่งมีวิธีการดำเนินการดังต่อไปนี้

1) สำรวจค่าตัวแปรต่าง ๆ ในชุดข้อมูลว่า ค่าตัวแปรใดที่เป็น “Missing Value” ถ้าหากค่าตัวแปรใดค่าหนึ่งในชุดข้อมูลนั้น เป็น “Missing Value” ให้ใส่ค่ามัธยฐาน (Median) แทน เนื่องจากการกระจายตัวของข้อมูลส่วนมากมีลักษณะเบ้ซ้าย หรือลักษณะเบ้ขวา นอกจากนี้ค่าตัวแปรนั้นอาจจะมีค่าผิดปกติ (Outlier)

2) วิธีการปรับช่วงขอบเขตของข้อมูล (Feature Scaling) เนื่องจากในชุดข้อมูลมีส่วนเบี่ยงเบนมาตรฐาน (Standard Deviation) น้อยมาก จึงเลือกวิธีการปรับขนาดข้อมูล (Rescaling) แบบ Min-Max Normalization ก่อนทำการทดลอง ซึ่งเป็นวิธีการปรับข้อมูลให้อยู่ในระหว่าง 0 ถึง 1 โดยสูตรการคำนวณดังสมการที่ (3.1)

$$X' = \frac{(\mathrm{x} - \mathrm{min})}{(\mathrm{max} - \mathrm{min})} \tag{3.1}$$

โดยที่ X' คือ ค่าที่ทำการปรับขนาดข้อมูล ให้อยู่ในระหว่าง 0 ถึง 1

3) ทำการตรวจสอบความสัมพันธ์เชิงเส้นระหว่างกันของตัวแปรอิสระ (Multicollinearity) โดยพิจารณาจากค่าสหสัมพันธ์ (Correlation Coefficient) จากตารางเมทริกซ์สหสัมพันธ์ (Correlation Matrix) โดยหลักเกณฑ์การพิจารณาค่าสัมประสิทธิ์สหสัมพันธ์นี้จะมีช่วงอยู่ระหว่าง -1.0 ถึง +1.0 โดยที่หากตัวแปรใดมีค่าใกล้ -1.0 นั้นหมายความว่าตัวแปรทั้งสองตัวมีความสัมพันธ์กันอย่างมากในเชิงตรงกันข้าม และถ้าหากมีค่าใกล้ +1.0 นั้นหมายความว่า ตัวแปรทั้งสองมีความสัมพันธ์กันโดยตรงอย่างมาก

3.1.2 อัลกอริทึมที่ใช้ในการจำแนกประเภท (Classification) ของข้อมูล

ในการศึกษาค้นคว้าอิสระนี้เพื่อต้องการเพิ่มประสิทธิภาพของกำลังในการผลิต โดยนำข้อมูลของเครื่องจักรที่รวบรวมเก็บได้ มาวิเคราะห์เพื่อลดปัญหาเครื่องจักรทำงานผิดพลาดและปรับปรุงส่งเสริมกระบวนการผลิต ทำให้เครื่องจักรสามารถผลิตปริมาณสินค้าได้มากยิ่งขึ้น ผู้วิจัยได้เลือกอัลกอริทึมสำหรับใช้ในการจำแนกประเภท เพื่อจำแนกและแยกสถานะความผิดพลาดจากการทำงานของเครื่องจักรที่ใช้ในกระบวนการผลิต

สถานการณ์ทำงานที่ผิดปกติของเครื่องจักร แบ่งออกเป็นทั้งหมด 5 ประเภท ได้แก่ ความผิดปกติที่เกิดจากการสึกหร่อของเครื่องจักร (Tool Wear Failure) ความผิดปกติที่เกิดจากความร้อนสะสม (Heat Dissipation Failure) ความผิดปกติเกิดจากพลังงานไฟฟ้ากำลัง (Power Failure) ความ-ผิดปกติเนื่องจากเกินกำลัง (Overstrain Failure) รวมถึงความผิดปกติเนื่องจากความไม่แน่นอน (Random Failure) เป็นต้น

3.1.3 วิธีการทำนายสถานะความผิดพลาดการทำงานของเครื่องจักร

ในการศึกษาค้นคว้าอิสระนี้ได้เลือกอัลกอริทึมสำหรับใช้ในการจำแนกประเภท (Classification) เพื่อจำแนกชนิดความผิดพลาดการทำงานของเครื่องจักรในกระบวนการผลิต และวัดประสิทธิภาพของกลุ่มตัวอย่าง เพื่อเปรียบเทียบการทำงานของโมเดลที่มีผลต่อประสิทธิภาพการทำงานของเครื่องจักร ที่ส่งผลให้ปริมาณการผลิตเพิ่มมากยิ่งขึ้น ซึ่งจำนวนอัลกอริทึมที่นำมาใช้ในการทดลองมีทั้งหมด 3 ประเภท ได้แก่ อัลกอริทึม Gradient Boosting Machine อัลกอริทึม k-Nearest Neighbors และอัลกอริทึม Random Forest

3.1.4 การวัดผลการทดลอง (Evaluation)

ในการวัดผลสถานะความผิดพลาดการทำงานของเครื่องจักรในกลุ่มตัวอย่างที่นำมาทดลอง เลือกใช้วิธีการวัดความถูกต้องของเครื่องจักรแบบโดยรวม (Accuracy) ค่าความแม่นยำ (Precision) ค่าความระลึกได้ (Recall) และค่า F1-Score ของสถานการณ์ทำงานเครื่องจักรทั้งหมด รวมถึงวิธีการวัดประสิทธิภาพโมเดลของอัลกอริทึม

3.2 การออกแบบการทดลอง

3.2.1 ขั้นตอนการเลือกสถานการณ์ทำงานของเครื่องจักรจากชุดข้อมูล

ชุดข้อมูลที่นำมาทดลอง ได้จาก open source ของ UCI เป็นชุดข้อมูลเกี่ยวกับการบำรุงเชิงรักษาของเครื่องจักรในโรงงาน (AI4I 2020 Predictive Maintenance Dataset) งานวิจัยนี้สนใจสถานการณ์ทำงานต่าง ๆ ของเครื่องจักรเท่านั้น ซึ่งภายในชุดข้อมูลมีทั้งหมด 10,000 ข้อมูล ประกอบด้วย พารามิเตอร์ต่าง ๆ ทั้งหมด 10 แบบ รายละเอียดตามตารางที่ 3.2

ตารางที่ 3.2 รายละเอียดคุณลักษณะที่นำมาสร้างตัวแบบ

ลำดับ	คุณลักษณะ	คำอธิบาย
1	UDI	หมายเลขอ้างอิง
2	Product_ID	รหัสสินค้า
3	Type	ขนาดของสินค้า มีทั้งหมด 3 ขนาด คือ ชนิด H, L และ M
4	Air_temperature_[K]	อุณหภูมิอากาศ หน่วยเคลวิน

ตารางที่ 3.2 (ต่อ) รายละเอียดคุณลักษณะที่นำมาสร้างตัวแบบ

ลำดับ	คุณลักษณะ	คำอธิบาย
5	Process_temperature_[K]	อุณหภูมิที่เครื่องจักรวัดได้ หน่วยเคลวิน
6	Rotational_speed_[rpm]	ความเร็วรอบของมอเตอร์ หน่วย RPM
7	Torque_[Nm]	แรงบิดมอเตอร์ (Nm)
8	Tool wear_[min]	เวลาที่ชดเชยของเครื่องจักรขณะที่กำลังผลิต (นาที)
9	Target	โหมดแสดงสถานะเครื่องจักรทำงาน (ปกติ/ไม่ปกติ)
10	Failure Type	ชนิดสถานะเครื่องจักรทำงานผิดปกติ

3.2.2 วิธีการทดลอง

3.2.2.1 ขั้นตอนการเตรียมชุดข้อมูล

ขั้นตอนการเตรียมชุดข้อมูล ทำการเลือกชุดข้อมูลของสินค้าที่ผลิตได้ทั้งหมด 10,000 แถว และทำการแบ่งข้อมูลออกเป็น 2 ส่วน ได้แก่ ชุดข้อมูลสำหรับการสอนโมเดล (Training Dataset) เพื่อให้โมเดลได้เรียนรู้และทำการรู้จักข้อมูลทั้งหมด และชุดข้อมูลสำหรับทำการทดสอบ (Test Dataset) โดยแบ่งจำนวนข้อมูลสำหรับการสอนโมเดลออกเป็นร้อยละ 70 ของจำนวนข้อมูลทั้งหมด และแบ่งจำนวนข้อมูลสำหรับทดสอบความถูกต้องของโมเดลออกเป็นร้อยละ 30 ของจำนวนข้อมูลทั้งหมด เพื่อนำมาสร้างแบบจำลองในการวิเคราะห์สถานะความผิดพลาดการทำงานของเครื่องจักรในกระบวนการผลิต จากนั้นขั้นตอนเตรียมชุดข้อมูลสำหรับการสอน และข้อมูลสำหรับการทดสอบเริ่มจากการตรวจสอบลักษณะข้อมูลว่า เกิดความไม่สมดุลกันในกลุ่มชุดข้อมูลเป้าหมาย (Target Dataset) หรือไม่ ถ้าหากชุดข้อมูลเกิดความไม่สมดุล (Imbalanced Data) จะทำการสังเคราะห์ข้อมูลใหม่จากกลุ่มตัวอย่างเดิมให้มีจำนวนตัวอย่างมากขึ้น เพื่อช่วยทำให้เกิดการกระจายของกลุ่มข้อมูลมีความสมดุลมากยิ่งขึ้น ก่อนที่จะนำชุดข้อมูลเหล่านี้ไปใช้กับอัลกอริทึมเพื่อทำนายและวิเคราะห์ได้ในขั้นตอนต่อไป

3.2.2.2 ขั้นตอนการเลือกคุณลักษณะ (Feature Selection)

ชุดข้อมูลที่ได้หลังจากการทำ SMOTE ประกอบไปด้วยพารามิเตอร์หรือคุณลักษณะทั้งหมด 10 แบบ ในจำนวนทั้งหมด 10,000 ข้อมูล เพื่อนำมาสร้างเป็นตัวแบบโมเดลอัลกอริทึม ซึ่งเป็นขนาดข้อมูลที่มีจำนวนมิติขนาดใหญ่ขนาดใหญ่ ในการใช้ข้อมูลที่มิติของข้อมูลขนาดใหญ่ในการเรียนรู้ของเครื่องทำให้เสียเวลา และอาจจะส่งผลให้ประสิทธิภาพของโมเดลที่ได้ไม่ดีเท่าที่ควร ดังนั้นการเลือกคุณลักษณะจึงมีความจำเป็นเพื่อลดประสิทธิภาพของโมเดลที่ได้ไม่ดีเท่าที่ควร

ในงานวิจัยได้เลือกใช้วิธี Embedded Method สำหรับการคัดเลือกคุณลักษณะ โดยใช้ Extra Tree Classifier มาแยกคุณลักษณะทั้งหมด 10 แบบของชุดข้อมูล กำหนดให้ตัวแปรเป้าหมายเป็นสถานะการทำงานของเครื่องจักร แล้วทำการเรียงลำดับความสำคัญของพารามิเตอร์แต่ละตัว เมื่อเรียงลำดับความสำคัญของพารามิเตอร์เรียบร้อยแล้ว ให้พิจารณาเลือกจากความสำคัญของพารามิเตอร์โดยการรวมค่าความสำคัญ (Feature Importance) และตัดตัวแปรที่ไม่เกี่ยวข้องออก เนื่องจากไม่มีความสำคัญในการเรียนรู้ของอัลกอริทึม

3.2.2.3 ขั้นตอนการสร้างโมเดลแบบจำลอง

ส่วนอัลกอริทึมที่นำมาใช้ในการทำนายเป็นแบบการจำแนกประเภท (Classification) มีทั้งหมด 3 ประเภท ได้แก่ อัลกอริทึม Gradient Boosting Machine อัลกอริทึม k-Nearest Neighbors และอัลกอริทึม Random Forest เพราะอัลกอริทึมทั้งหมดเป็นลักษณะแบบ Supervised Learning และเลือกใช้วิธีสุ่มเลือกแบ่งข้อมูลแบบความเที่ยงตรง k กลุ่ม (K-Fold Cross Validation) เพื่อให้การเรียนรู้ของเครื่องมีประสิทธิภาพเพิ่มมากยิ่งขึ้น โดยเริ่มจากการแบ่งชุดข้อมูลออกเป็นส่วน ๆ ให้เท่ากัน กำหนดค่า K =10 จากนั้นให้นำข้อมูลบางส่วนมาทำการเรียนรู้ และนำข้อมูลบางส่วนมาทำการทดสอบแบบจำลองที่ได้จากการเรียนรู้ เพื่อช่วยลดการเกิดปัญหาโมเดลชำนาญเกินไปในการเรียนรู้ชุดข้อมูลสำหรับการสอน รวมถึงทำการเปรียบเทียบประสิทธิภาพของแบบจำลองทั้งหมด 3 โมเดลที่กล่าวข้างต้น

3.2.2.3 เกณฑ์การวัดประสิทธิภาพของโมเดล

หลังจากสร้างโมเดลการเรียนรู้ของเครื่องด้วยข้อมูลเรียนรู้แล้ว ทำการทดสอบโมเดลที่สร้างขึ้นทั้งหมด 3 โมเดล ด้วยข้อมูลทดสอบ และทำการวัดประสิทธิภาพของแต่ละโมเดลจากผลการทำนายจากข้อมูลทดสอบ โดยในการวัดประสิทธิภาพของโมเดลจะพิจารณาและเปรียบเทียบค่าความถูกต้อง (Accuracy) ค่าความแม่นยำ (Precision) ค่าความระลึกได้ (Recall) และค่า F1-Score

3.3 เครื่องมือที่ใช้ในการทดลอง

การทดลองแบ่งออกเป็น 3 ส่วน ได้แก่ ส่วนที่นำชุดข้อมูลเกี่ยวกับการบำรุงเชิงรักษาของเครื่องจักรในโรงงานมาทำความสะอาด (Data Cleaning) และปรับให้อยู่ในรูปแบบที่ต้องการนำไปใช้ส่วนแบบจำลองอัลกอริทึมเพื่อการทำนาย และส่วนที่ใช้ในการคำนวณความถูกต้องและความแม่นยำของอัลกอริทึมที่ใช้ในแบบจำลองเพื่อการทำนายสถานะการทำงานของเครื่องจักรในกระบวนการผลิต สำหรับชุดข้อมูลที่ผ่านการทำความสะอาดแล้ว จะถูกเก็บในรูปแบบไฟล์ .csv ส่วนการทำงานของแบบจำลองอัลกอริทึมทั้งหมดจะถูกคำนวณโดย Google Collaboratory และเลือกใช้ภาษา Python ในการเขียนโปรแกรม

บทที่ 4

ผลการวิจัยและอภิปรายผล

เนื้อหาบทนี้นำเสนอผลการทดลองและการคำนวณเพื่อประเมินประสิทธิภาพของอัลกอริทึมแบบจำแนกประเภทที่ใช้ในการทำนายสถานะการทำงานของเครื่องจักรในโรงงาน โดยทำการวัดค่าความแม่นยำ ค่าความระลึก ค่าความถ่วงดุล และค่าความถูกต้อง ในการทำนายความผิดพลาดของสถานะการทำงานของเครื่องจักร รวมถึงการวิเคราะห์ผลการทดลอง

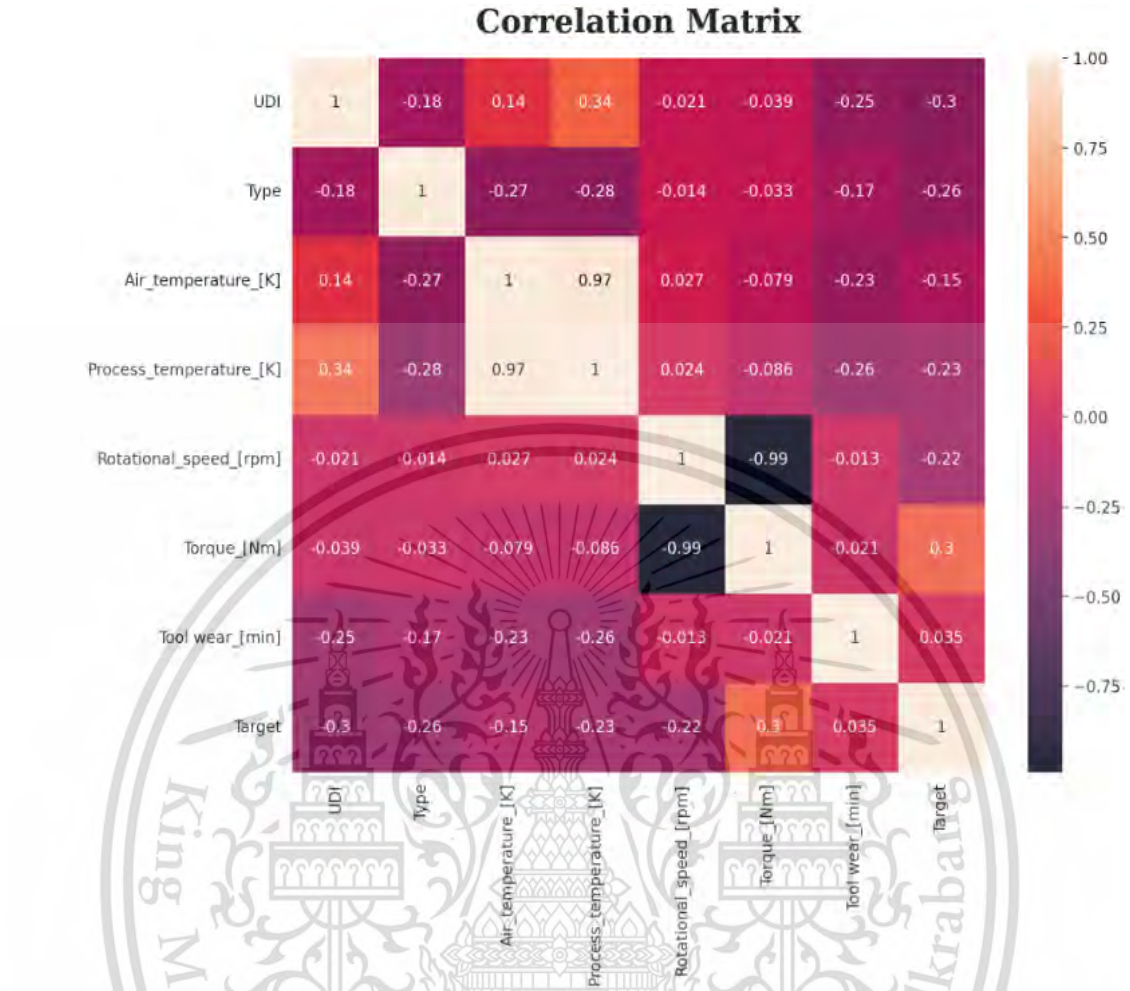
4.1 การเตรียมชุดข้อมูลและการจัดการชุดข้อมูลเบื้องต้น

ปริมาณข้อมูลที่ทำการทดลองมีจำนวน 10,000 ข้อมูล นำมาตรวจสอบวัดค่าสถิติเบื้องต้นของตัวแปรที่เป็น Numerical จะได้ดังตารางที่ 4.1

ตารางที่ 4.1 ค่าทางสถิติเบื้องต้นสำหรับข้อมูลทั้งหมด

Feature	count	mean	std	min	25%	50%	75%	max
UDI	10,000.00	5000.5	2886.8957	1.0000	2500.0000	5000.5000	7500.2500	10000.0000
Type	10,000.00	1.5003	0.6713	1.0000	1.0000	1.0000	2.0000	3.0000
Air_temperature_[K]	10,000.00	300.0049	2.0002	295.3000	298.3000	300.1000	301.5000	304.5000
Process_temperature_[K]	10,000.00	2.0002	1.4837	305.7000	308.8000	310.1000	311.1000	313.8000
Rotational_speed_[rpm]	10,000.00	1538.7761	179.2841	1168.0000	1423.0000	1503.0000	1612.0000	2886.0000
Torque_[Nm]	10,000.00	39.9869	9.9689	3.8000	33.2000	40.1000	46.8000	76.6000
Tool wear_[min]	10,000.00	107.9510	63.6515	0	53.0000	108.0000	162.0000	253.0000
Target	10,000.00	0.0339	0.1809	0	0	0	0	1.0000
Failure_Type	10,000.00	0.0850	0.51186	0	0	0	0	5.0000

จากตารางที่ 4.1 ผลลัพธ์ของสถิติเบื้องต้นของชุดข้อมูลพิจารณาจากค่าเฉลี่ย (Mean) และความแปรปวนของข้อมูลเป็นหลัก ค่าเฉลี่ยของตัวแปรแต่ละตัวมีค่าไม่เท่ากัน คิดจากค่าเฉลี่ยผลรวมของค่าทั้งหมด และความแปรปวนของข้อมูลใช้เพื่อวัดการกระจายของข้อมูล คิดจากค่าเบี่ยงเบนมาตรฐาน (Standard Deviation) จากนั้นทำการตรวจสอบความสัมพันธ์ระหว่างตัวแปรอิสระของชุดข้อมูล โดยใช้ตารางเมทริกซ์สหสัมพันธ์ (Correlation Matrix) ของชุดข้อมูล ใช้สถิติเพื่อหาค่าความสัมพันธ์ระหว่างตัวแปร ผลลัพธ์ที่ได้จากการหาความสัมพันธ์ระหว่างตัวแปรอิสระของชุดข้อมูลดังรูปที่ 4.1



รูปที่ 4.1 ตารางเมทริกซ์สหสัมพันธ์ (Correlation Matrix) ของชุดข้อมูล

จากรูปที่ 4.1 การตรวจสอบความสัมพันธ์ระหว่างตัวแปรอิสระของชุดข้อมูล โดยใช้ตารางเมทริกซ์สหสัมพันธ์ (Correlation Matrix) ของชุดข้อมูล โดยพิจารณาจากตัวแปรโหมดแสดงสถานะเครื่องจักรทำงาน (Target) ซึ่งเป็นตัวบ่งชี้ว่าเครื่องจักรทำงานปกติหรือไม่ปกติ พบว่า ค่าสหสัมพันธ์ที่มีค่ามากที่สุดกับตัวแปรโหมดแสดงสถานะเครื่องจักรทำงาน (Target) คือ แรงบิดมอเตอร์ (Torque) มีค่าสหสัมพันธ์เท่ากับ 0.3 แสดงให้เห็นว่าความสัมพันธ์ทั้งสองตัวแปรเชิงบวก ในระดับปานกลาง มีผลต่อตัวแปรโหมดแสดงสถานะเครื่องจักรทำงาน (Target) และเมื่อนำเซลล์จุดตัดระหว่างตัวแปรโหมดแสดงสถานะเครื่องจักรทำงาน (Target) และหมายเลขอ้างอิง (UDI) มาพิจารณา พบว่า ค่าสหสัมพันธ์มีค่าเท่ากับ – 0.3 บ่งบอกถึงความสัมพันธ์เชิงลบ หมายถึงความสัมพันธ์ระหว่างตัวแปรไม่แน่นแฟ้นมากนัก ทำให้ไม่มีผลต่อสถานการณ์ทำงานของเครื่องจักรเท่าที่ควร ในการทดลองครั้งนี้ไม่พบตัวแปรใดที่มีค่าสหสัมพันธ์ที่มีค่าเป็นศูนย์ เราสามารถนำเมทริกซ์สหสัมพันธ์ที่ได้จากการทดลองกับชุดข้อมูลนี้ไปใช้ในการพิจารณาความสำคัญของตัวแปรที่มีผลต่ออัลกอริทึมในขั้นตอนต่อไป

4.2 การคัดเลือกคุณลักษณะของชุดข้อมูล

ในขั้นตอนการคัดเลือกคุณลักษณะสำคัญของการทดลอง ใช้การ Import ไลบรารีของ skimage เพื่อคัดแยกคุณลักษณะสำคัญด้วยเทคนิค Embedded มาคัดแยกคุณสมบัติ 10 แบบของชุดข้อมูล โดยกำหนดให้ตัวแปรเป้าหมายเป็นสถานะการทำงานของเครื่องจักร เมื่อทำการคัดแยกคุณลักษณะสำคัญของชุดข้อมูลเสร็จเรียบร้อยแล้ว จะได้ค่าความสำคัญของตัวแปรแต่ละตัว มีรายละเอียดดังต่อไปนี้

ตารางที่ 4.2 ผลการคัดแยกคุณลักษณะสำคัญ

ลำดับ	คุณลักษณะ	คำอธิบาย	Feature Importance
1	Torque_[Nm]	แรงบิดมอเตอร์ (Nm)	0.29997
2	Rotational_speed_[rpm]	ความเร็วรอบของมอเตอร์ (RPM)	0.2149
3	Tool wear_[min]	เวลาที่ชดเชยของเครื่องจักรขณะที่กำลังผลิต (นาที)	0.1789
4	Air_temperature_[K]	อุณหภูมิอากาศ (K)	0.1469
5	Process_temperature_[K]	อุณหภูมิที่เครื่องจักรวัดได้ (K)	0.1299
6	Type	ขนาดของสินค้า	0.029382

จากตารางที่ 4.2 ผลลัพธ์ที่ได้จากการคัดแยกคุณลักษณะสำคัญของชุดข้อมูล พบว่า ค่าความสำคัญของตัวแปรที่มีผลต่อโมเดลมากที่สุด ได้แก่ พบว่า ตัวแปรที่มีความสำคัญมากที่สุดคือ แรงบิดมอเตอร์ (Torque) มีค่าเท่ากับ 0.2977 อันดับที่สองคือ ความเร็วรอบของมอเตอร์ (Rotational Speed) มีค่าเท่ากับ 0.2149 อันดับที่สามคือ เวลาที่ชดเชยของเครื่องจักรขณะที่กำลังผลิต (Tool wear) มีค่าเท่ากับ 0.1789 อันดับที่สี่คือ อุณหภูมิอากาศ (Air Temperature) มีค่าเท่ากับ 0.1469 อันดับที่ห้าคือ อุณหภูมิที่เครื่องจักรวัดได้ (Process Temperature) มีค่าเท่ากับ 0.1299 ส่วนอันดับสุดท้าย ได้แก่ ตัวแปรชนิดของสินค้า (Type) มีค่าเท่ากับ 0.02938

ในการทดลอง พิจารณาและคัดเลือกคุณลักษณะจากตัวแปรต้นในชุดข้อมูลทั้งหมด 4 ตัวแปร ได้แก่ แรงบิดมอเตอร์ (Nm) ความเร็วรอบของมอเตอร์ (RPM) เวลาที่ชดเชยของเครื่องจักรขณะที่กำลังผลิต (นาที) อุณหภูมิอากาศ (K) เนื่องจากค่าความสำคัญของตัวแปรมีค่าใกล้เคียงกัน

4.3 การจัดการข้อมูลที่ไม่สมดุล (Imbalanced Data)

จากชุดข้อมูลสำหรับการสอน และข้อมูลสำหรับการทดสอบ พบว่า ข้อมูลเกิดความไม่สมดุลกันในกลุ่มชุดข้อมูลเป้าหมาย (Target Dataset) ดังนั้นในการทดลองจึงได้เลือกวิธีการสุ่มเพิ่มตัวอย่างกลุ่มน้อย (Synthetic Minority Oversampling Technique: SMOTE) เพื่อสังเคราะห์ข้อมูลใหม่จากกลุ่มตัวอย่างเดิมให้มีจำนวนตัวอย่างมากขึ้น ช่วยทำให้เกิดการกระจายของกลุ่มข้อมูลมีความสมดุลมากยิ่งขึ้น รายละเอียดดังตารางที่ 4.3

ตารางที่ 4.3 จำนวนข้อมูลก่อนและหลังทำ SMOTE

สถานะการทำงานของเครื่องจักร (Machine Failure Type)	จำนวนข้อมูล ก่อนทำ SMOTE	จำนวนข้อมูล หลังทำ SMOTE
เครื่องจักรทำงานปกติ (No Failure)	6,756	6,756
ความผิดปกติที่เกิดจากความร้อนสะสม (Heat Dissipation Failure)	78	6,756
ความผิดปกติที่เกิดจากผลิตเกินกำลัง (Overstrain Failure)	55	6,756
ความผิดปกติที่เกิดจากไฟฟ้ากำลัง (Power Failure)	67	6,756
ความผิดปกติจากความไม่แน่นอนของเครื่องจักร (Random Failure)	13	6,756
ความผิดปกติเนื่องจากการสึกหร่อของเครื่องจักร (Tool Wear Failure)	31	6,756

จากตารางที่ 4.3 ชุดข้อมูลสำหรับการทดสอบจำนวน 7,000 ข้อมูล ประกอบด้วยข้อมูลเป้าหมายประเภท ปกติ จำนวน 6,756 ข้อมูล และผิดปกติที่เกิดจากความร้อน จำนวน 348 ข้อมูล หลังจากทำการสุ่มเพิ่มตัวอย่างกลุ่มน้อย พบว่า ชุดข้อมูลสำหรับการทดสอบมีจำนวนเพิ่มขึ้น ได้แก่ ข้อมูลสถานะความผิดปกติที่เกิดจากความร้อนสะสม (Heat Dissipation Failure) จาก 78 ข้อมูล เป็นจำนวน 6,756 ข้อมูล ข้อมูลสถานะความผิดปกติที่เกิดจากผลิตเกินกำลัง (Overstrain Failure) จาก 55 ข้อมูล เป็นจำนวน 6,756 ข้อมูล ข้อมูลสถานะความผิดปกติที่เกิดจากไฟฟ้ากำลัง (Power Failure) จาก 67 ข้อมูล เป็นจำนวน 6,756 ข้อมูล ข้อมูลสถานะความผิดปกติจากความไม่แน่นอนของเครื่องจักร (Random Failure) จาก 13 ข้อมูล เป็นจำนวน 6,756 ข้อมูล รวมถึงข้อมูลสถานะความผิดปกติเนื่องจากการสึกหร่อของเครื่องจักร (Tool Wear Failure) จาก 31 ข้อมูล เป็นจำนวน 6,756 ข้อมูล ส่วนข้อมูลสถานะเครื่องจักรทำงานปกติ มีจำนวนข้อมูล 6,756 ข้อมูล เท่าเดิม

หลังจากทำการสุ่มเพิ่มตัวอย่างกลุ่มน้อยเรียบร้อยแล้ว นำข้อมูลชุดการทดลองไปทำการสอนและทดสอบโมเดล พร้อมทั้งปรับแต่งค่าพารามิเตอร์เพื่อทำให้ประสิทธิภาพในการทำนายที่ดี และเพิ่มความแม่นยำในการทำนายมากขึ้น

4.4 การทดลองโดยใช้อัลกอริทึม Gradient Boosting Machine

ผลการทดลองจาก Classification Report โดยเลือกใช้อัลกอริทึม Gradient Boosting Machine ในการสร้างโมเดลและวัดประสิทธิภาพ และเลือกใช้วิธีสุ่มเลือกแบ่งข้อมูลแบบความเที่ยงตรง k กลุ่ม (K-Fold Cross Validation) กำหนดค่า K = 10 ซึ่งมีรายละเอียดดังตารางที่ 4.4

ตารางที่ 4.4 ผลการทดลองโดยเลือกใช้อัลกอริทึม Gradient Boosting Machine

สถานะการทำงานของเครื่องจักร (Machine Failure Type)	Precision	Recall	F1-score	Support
เครื่องจักรทำงานปกติ (No Failure)	0.99	0.94	0.97	2896
ความผิดปกติที่เกิดจากความร้อนสะสม (Heat Dissipation Failure)	0.38	0.62	0.47	34
ความผิดปกติที่เกิดจากผลิตเกินกำลัง (Overstrain Failure)	0.63	0.74	0.68	23
ความผิดปกติที่เกิดจากไฟฟ้ากำลัง (Power Failure)	0.83	0.68	0.75	28
ความผิดปกติจากความไม่แน่นอนของเครื่องจักร (Random Failure)	0.00	0.00	0.00	5
ความผิดปกติเนื่องจากการสึกหร่อของเครื่องจักร (Tool Wear Failure)	0.11	0.43	0.18	14
Accuracy	-	-	0.93	3000
Macro avg	0.49	0.57	0.51	3000
Weighted avg	0.97	0.93	0.95	3000

จากตารางที่ 4.4 เป็นการแสดงผลลัพธ์จาก Classification Report ของการวัดประสิทธิภาพการทำนายโมเดล Gradient Boosting Machine แสดงค่า Precision, Recall และ F1-score เมื่อพิจารณาจากสถานะเครื่องจักรทำงานปกติ จะได้ค่า Precision = 99%, Recall = 94% และ F1-score = 97% สถานะความผิดปกติที่เกิดจากความร้อนสะสม จะได้ค่า Precision = 38%, Recall = 62% และ F1-score = 47% สถานะความผิดปกติที่เกิดจากผลิตเกินกำลัง จะได้ค่า Precision = 63%, Recall = 74% และ F1-score 68% สถานะความผิดปกติที่เกิดจากไฟฟ้ากำลัง จะได้ค่า Precision = 83%, Recall = 68% และ F1-score = 75% สถานะความผิดปกติจากความไม่แน่นอนของเครื่องจักร จะได้ค่า Precision = 0%, Recall = 0% และ F1-score = 0% สถานะความผิดปกติเนื่องจากการสึกหร่อของเครื่องจักร จะได้ค่า Precision = 11%, Recall = 43% และ F1-score

เท่ากับ 18% และเมื่อพิจารณาค่าเฉลี่ยน้ำหนักของสถานะทั้งหมด จะได้ค่า Precision = 97%, Recall = 93% และ F1-Score = 95% ซึ่งให้ค่าประสิทธิโดยรวมของโมเดล Gradient Boosting Machine เท่ากับ 93% เมื่อเราพิจารณาร่วมกับตาราง Confusion Matrix จะได้ผลลัพธ์การทำนายรูปแบบ Confusion Matrix ของโมเดล Gradient Boosting Machine เป็นดังตารางที่ 4.5

ตารางที่ 4.5 ผลลัพธ์รูปแบบ Confusion Matrix ของโมเดล Gradient Boosting Machine

Gradient Boosting Machine		Predicted					
		No Failure	Heat Dissipation Failure	Overstrain Failure	Power Failure	Random Failure	Tool Wear Failure
Actual	No Failure	2736	33	8	3	70	46
	Heat Dissipation Failure	9	21	1	1	2	0
	Overstrain Failure	3	2	17	0	0	0
	Power Failure	8	0	1	19	0	0
	Random Failure	4	0	0	0	0	1
	Tool Wear Failure	8	0	0	0	0	6

จากตารางที่ 4.5 แสดงการทำนาย Confusion Matrix ของการวัดประสิทธิภาพของโมเดล Gradient Boosting Machine พบว่า การทำนาย สถานะปกติ (No Failure) ถูกต้อง จำนวน 2,736 ครั้ง ทำนาย ปกติ (No Failure) ผิดพลาด จำนวน 100 ครั้ง สถานะความผิดปกติที่เกิดจากความร้อนสะสม (Heat Dissipation Failure) ถูกต้อง จำนวน 21 ครั้ง ทำนาย สถานะความผิดปกติที่เกิดจากความร้อนสะสม (Heat Dissipation Failure) ผิดพลาด จำนวน 13 ครั้ง สถานะความผิดปกติที่เกิดจากการผลิตเกินกำลัง (Overstrain Failure) ถูกต้อง จำนวน 17 ครั้ง สถานะความผิดปกติที่เกิดจากการผลิตเกินกำลัง (Overstrain Failure) ผิดพลาด จำนวน 5 ครั้ง สถานะความผิดปกติที่เกิดจากไฟฟ้ากำลัง (Power Failure) ถูกต้อง จำนวน 19 ครั้ง สถานะความผิดปกติที่เกิดจากไฟฟ้ากำลัง (Power Failure) ผิดพลาด จำนวน 9 ครั้ง สถานะความผิดปกติจากความไม่แน่นอนของเครื่องจักร (Random Failure) ถูกต้อง จำนวน 0 ครั้ง สถานะความผิดปกติจากความไม่แน่นอนของเครื่องจักร (Random Failure) ผิดพลาด จำนวน 5 ครั้ง ในส่วนสถานะความผิดปกติเนื่องจากการสึกหร่อของเครื่องจักร (Tool Wear Failure) ถูกต้อง จำนวน 6 ครั้ง สถานะความผิดปกติเนื่องจากการสึกหร่อของเครื่องจักร (Tool Wear Failure) ผิดพลาด จำนวน 8 ครั้งตามลำดับ

4.5 การทดลองโดยใช้อัลกอริทึม k-Nearest Neighbors

ผลการทดลองจาก Classification Report โดยเลือกใช้อัลกอริทึม k-Nearest Neighbors ในการสร้างโมเดลและวัดประสิทธิภาพ และเลือกใช้วิธีสุ่มเลือกแบ่งข้อมูลแบบความเที่ยงตรง k กลุ่ม (K-Fold Cross Validation) กำหนดค่า K = 10 รายละเอียดดังตารางที่ 4.6

ตารางที่ 4.6 ผลการทดลองโดยเลือกใช้อัลกอริทึม k-Nearest Neighbors

สถานะการทำงานของเครื่องจักร (Machine Failure Type)	Precision	Recall	F1-score	Support
เครื่องจักรทำงานปกติ (No Failure)	0.99	0.73	0.84	2896
ความผิดปกติที่เกิดจากความร้อนสะสม (Heat Dissipation Failure)	0.10	0.56	0.17	34
ความผิดปกติที่เกิดจากผลิตเกินกำลัง (Overstrain Failure)	0.42	0.70	0.52	23
ความผิดปกติที่เกิดจากไฟฟ้ากำลัง (Power Failure)	0.44	0.64	0.52	28
ความผิดปกติจากความไม่แน่นอนของเครื่องจักร (Random Failure)	0.00	0.20	0.00	5
ความผิดปกติเนื่องจากการสึกหรอของเครื่องจักร (Tool Wear Failure)	0.08	0.64	0.14	14
Accuracy	-	-	0.72	3000
Macro avg	0.34	0.58	0.37	3000
Weighted avg	0.96	0.72	0.82	3000

จากตารางที่ 4.6 ผลลัพธ์จาก Classification Report ของการวัดประสิทธิภาพการทำนายโมเดล k-Nearest Neighbors แสดงค่า Precision, Recall และ F1-score เมื่อพิจารณาจากสถานะเครื่องจักรทำงานปกติ (No Failure) จะได้ค่า Precision = 99%, Recall = 73% และ F1-score = 84% สถานะความผิดปกติที่เกิดจากความร้อนสะสม (Heat Dissipation Failure) จะได้ค่า Precision = 10%, Recall = 56% และ F1-score = 17% สถานะความผิดปกติที่เกิดจากผลิตเกินกำลัง (Overstrain Failure) จะได้ค่า Precision = 42%, Recall = 70% และ F1-score = 52% สถานะความผิดปกติที่เกิดจากไฟฟ้ากำลัง (Power Failure) จะได้ค่า Precision = 44%, Recall = 64% และ F1-score = 52% สถานะความผิดปกติจากความไม่แน่นอนของเครื่องจักร (Random Failure) จะได้ค่า Precision = 0%, Recall = 20% และ F1-score = 0% สถานะความผิดปกติ

เนื่องจากการสึกหร่อของเครื่องจักร (Tool Wear Failure) จะได้ค่า Precision = 8%, Recall = 64% และ F1-score = 14% และเมื่อพิจารณาค่าเฉลี่ยน้ำหนักของสถานะทั้งหมด จะได้ค่า Precision = 96%, Recall = 72% และ F1-Score = 82% ซึ่งให้ค่าประสิทธิภาพโดยรวมของโมเดล k-Nearest Neighbors ที่ค่อนข้างปานกลาง

ตารางที่ 4.7 ผลลัพธ์รูปแบบ Confusion Matrix ของโมเดล k-Nearest Neighbors

k-Nearest Neighbors		Predicted					
		No Failure	Heat Dissipation Failure	Overstrain Failure	Power Failure	Random Failure	Tool Wear Failure
Actual	No Failure	2102	160	21	22	485	106
	Heat Dissipation Failure	10	19	0	1	4	0
	Overstrain Failure	0	4	16	0	0	3
	Power Failure	4	4	1	18	1	0
	Random Failure	3	0	0	0	1	1
	Tool Wear Failure	5	0	0	0	0	9

จากตารางที่ 4.7 แสดงการทำนาย Confusion Matrix ของการวัดประสิทธิภาพของโมเดล k-Nearest Neighbors พบว่า การทำนาย สถานะปกติ (No Failure) ถูกต้อง จำนวน 2,102 ครั้ง ทำนาย ปกติ (No Failure) ผิดพลาด จำนวน 794 ครั้ง สถานะความผิดปกติที่เกิดจากความร้อนสะสม (Heat Dissipation Failure) ถูกต้อง จำนวน 19 ครั้ง ทำนาย สถานะความผิดปกติที่เกิดจากความร้อนสะสม (Heat Dissipation Failure) ผิดพลาด จำนวน 15 ครั้ง สถานะความผิดปกติที่เกิดจากการผลิตเกินกำลัง (Overstrain Failure) ถูกต้อง จำนวน 16 ครั้ง สถานะความผิดปกติที่เกิดจากการผลิตเกินกำลัง (Overstrain Failure) ผิดพลาด จำนวน 7 ครั้ง สถานะความผิดปกติที่เกิดจากไฟฟ้ากำลัง (Power Failure) ถูกต้อง จำนวน 18 ครั้ง สถานะความผิดปกติที่เกิดจากไฟฟ้ากำลัง (Power Failure) ผิดพลาด จำนวน 10 ครั้ง สถานะความผิดปกติจากความไม่แน่นอนของเครื่องจักร (Random Failure) ถูกต้อง จำนวน 1 ครั้ง สถานะความผิดปกติจากความไม่แน่นอนของเครื่องจักร (Random Failure) ผิดพลาด จำนวน 4 ครั้ง สถานะความผิดปกติเนื่องจากการสึกหร่อของเครื่องจักร (Tool Wear Failure) ถูกต้อง จำนวน 9 ครั้ง สถานะความผิดปกติเนื่องจากการสึกหร่อของเครื่องจักร (Tool Wear Failure) ผิดพลาด จำนวน 5 ครั้งตามลำดับ

4.6 การทดลองโดยใช้อัลกอริทึม Random Forest

ผลการทดลองจาก Classification Report โดยเลือกใช้อัลกอริทึม Random Forest ในการสร้างโมเดลและวัดประสิทธิภาพโดยรวม และเลือกใช้วิธีสุ่มเลือกแบ่งข้อมูลแบบความเที่ยงตรง k กลุ่ม (K-Fold Cross Validation) กำหนดค่า K = 10 รายละเอียดดังตารางที่ 4.8

ตารางที่ 4.8 ผลการทดลองโดยเลือกใช้อัลกอริทึม Random Forest

สถานะการทำงานของเครื่องจักร (Machine Failure Type)	Precision	Recall	F1-score	Support
เครื่องจักรทำงานปกติ (No Failure)	0.99	0.82	0.90	2896
ความผิดปกติที่เกิดจากความร้อนสะสม (Heat Dissipation Failure)	0.36	0.79	0.50	34
ความผิดปกติที่เกิดจากผลิตเกินกำลัง (Overstrain Failure)	0.55	0.70	0.62	23
ความผิดปกติที่เกิดจากไฟฟ้ากำลัง (Power Failure)	0.81	0.75	0.78	28
ความผิดปกติจากความไม่แน่นอนของเครื่องจักร (Random Failure)	0.00	0.00	0.00	5
ความผิดปกติเนื่องจากการสึกหรอของเครื่องจักร (Tool Wear Failure)	0.08	0.64	0.14	14
Accuracy	-	-	0.82	3000
Macro avg	0.46	0.62	0.49	3000
Weighted avg	0.97	0.82	0.89	3000

จากตารางที่ 4.8 จากการทดลอง ผลลัพธ์ที่ได้ Classification Report ของการวัดประสิทธิภาพการทำนายโมเดล Random Forest Classifier แสดงค่า Precision, Recall และ F1-score เมื่อพิจารณาจากสถานะเครื่องจักรทำงานปกติ (No Failure) จะได้ค่า Precision = 99% Recall = 82% และ F1-score = 90% สถานะความผิดปกติที่เกิดจากความร้อนสะสม (Heat Dissipation Failure) จะได้ค่า Precision = 36%, Recall = 79% และ F1-score = 50% สถานะความผิดปกติที่เกิดจากการผลิตเกินกำลัง (Overstrain Failure) จะได้ค่า Precision = 55%, Recall = 70% และ F1-score = 62%, สถานะความผิดปกติที่เกิดจากไฟฟ้ากำลัง (Power Failure) จะได้ค่า Precision = 81%, Recall = 75% และ F1-score = 78%, สถานะความผิดปกติจากความไม่แน่นอนของเครื่องจักร (Random Failure) จะได้ค่า Precision = 0%, Recall = 0%และ F1-score

เท่ากับ 0% สถานะความผิดปกติเนื่องจากการสึกหร่อของเครื่องจักร (Tool Wear Failure) จะได้ค่า Precision = 8%, Recall = 64% และ F1-score = 14% และเมื่อพิจารณาค่าเฉลี่ยน้ำหนักของสถานะทั้งหมด จะได้ค่า Precision = 97%, Recall = 82% และ F1-Score = 89% ซึ่งให้ค่าประสิทธิภาพของโมเดล Random Forest ที่ค่อนข้างดี

ตารางที่ 4.9 ผลลัพธ์รูปแบบ Confusion Matrix ของโมเดล Random Forest

Random Forest Classifier		Predicted					
		No Failure	Heat Dissipation Failure	Overstrain Failure	Power Failure	Random Failure	Tool Wear Failure
Actual	No Failure	2381	46	12	3	346	108
	Heat Dissipation Failure	3	27	0	1	3	0
	Overstrain Failure	3	2	16	1	0	1
	Power Failure	6	0	1	21	0	0
	Random Failure	4	0	0	0	0	1
	Tool Wear Failure	5	0	0	0	0	9

จากตารางที่ 4.9 แสดงการทำนาย Confusion Matrix ของการวัดประสิทธิภาพของโมเดล Random Forest พบว่า การทำนาย สถานะปกติ (No Failure) ถูกต้อง จำนวน 2,381 ครั้ง ทำนายปกติ (No Failure) ผิดพลาด จำนวน 515 ครั้ง สถานะความผิดปกติที่เกิดจากความร้อนสะสม (Heat Dissipation Failure) ถูกต้อง จำนวน 27 ครั้ง ทำนาย สถานะความผิดปกติที่เกิดจากความร้อนสะสม (Heat Dissipation Failure) ผิดพลาด จำนวน 7 ครั้ง สถานะความผิดปกติที่เกิดจากการผลิตเกินกำลัง (Overstrain Failure) ถูกต้อง จำนวน 16 ครั้ง สถานะความผิดปกติที่เกิดจากการผลิตเกินกำลัง (Overstrain Failure) ผิดพลาด จำนวน 7 ครั้ง สถานะความผิดปกติที่เกิดจากไฟฟ้ากำลัง (Power Failure) ถูกต้อง จำนวน 21 ครั้ง สถานะความผิดปกติที่เกิดจากไฟฟ้ากำลัง (Power Failure) ผิดพลาด จำนวน 7 ครั้ง สถานะความผิดปกติจากความไม่แน่นอนของเครื่องจักร (Random Failure) ถูกต้อง จำนวน 0 ครั้ง สถานะความผิดปกติจากความไม่แน่นอนของเครื่องจักร (Random Failure) ผิดพลาด จำนวน 5 ครั้ง สถานะความผิดปกติเนื่องจากการสึกหร่อของเครื่องจักร (Tool Wear Failure) ถูกต้อง จำนวน 9 ครั้ง สถานะความผิดปกติเนื่องจากการสึกหร่อของเครื่องจักร (Tool Wear Failure) ผิดพลาด จำนวน 5 ครั้งตามลำดับ

4.7 ความแม่นยำของอัลกอริทึม

จากผลการทดลองเปรียบเทียบค่าความแม่นยำ ค่าความระลึกได้ ค่าความถ่วงดุล และค่าความถูกต้องของโมเดลอัลกอริทึมทั้งหมด ในการทำนายความผิดพลาดของสถานะการทำงานของเครื่องจักรที่เกี่ยวข้องในกระบวนการผลิต มีรายละเอียดดังตารางที่ 4.10

ตารางที่ 4.10 ผลความแม่นยำของอัลกอริทึม

Algorithm Model	Accuracy	Precision	Recall	F1-score
Gradient Boosting Machine	0.9330	0.4884	0.5681	0.5057
k-Nearest Neighbors	0.7217	0.4610	0.6363	0.4827
Random Forest	0.8103	0.3381	0.5777	0.3658

จากตารางที่ 4.10 เป็นตารางแสดงความแม่นยำของการทำนายของอัลกอริทึมทั้ง 3 แบบ ได้แก่ อัลกอริทึม Gradient Boosting Machine, อัลกอริทึม k-Nearest Neighbors อัลกอริทึม Random Forest หลังจากการทำความสะอาดข้อมูล พร้อมทั้งปรับค่าพารามิเตอร์ (Turning Parameter) ด้วยการค้นหาแบบกริด (Grid Search) และทำการวิเคราะห์ความแม่นตรงของตัวแบบ (K-Fold Cross Validation) โดยการแบ่งชุดข้อมูลสำหรับการสอนออกเป็นจำนวน K ส่วน แบบสุ่มจำนวนเท่า ๆ กัน กำหนดค่า K = 10 เพื่อสร้างและสอนโมเดลก่อนนำไปทำนายประสิทธิภาพด้วยชุดข้อมูลสำหรับการทดสอบ (Test Data) ซึ่งในการทดลองนี้ได้ทำการแบ่งชุดข้อมูลออกเป็นจำนวน 10 ส่วน

ประสิทธิภาพการทำนายความผิดปกติของเครื่องจักร พบว่า โมเดลที่มีความแม่นยำมากที่สุดคืออัลกอริทึม Gradient Boosting Machine ซึ่งมีค่าเท่ากับ 93.30% ที่ Precision = 48.84% Recall = 56.81% และ F1-Score = 50.57% อันดับที่ 2 คืออัลกอริทึม Random Forest มีความแม่นยำเท่ากับ 81.03% ที่ Precision = 46.10%, Recall = 63.63% และ F1-Score = 48.27% และอัลกอริทึมที่มีค่าแม่นยำต่ำที่สุดคือ อัลกอริทึม k-Nearest Neighbors มีค่าแม่นยำเท่ากับ 72.17% ที่ Precision = 33.81%, Recall = 57.77% และ F1-Score = 36.58%

บทที่ 5

สรุปผลการวิจัยและข้อเสนอแนะ

5.1 สรุปผลการวิจัย

งานวิจัยนี้มีแนวคิดในการปรับปรุงกระบวนการผลิตให้ดียิ่งขึ้น โดยนำระบบการบำรุงรักษาเชิงพยากรณ์สำหรับกระบวนการผลิตเข้ามาใช้ร่วมเพื่อช่วยวิเคราะห์สถานะการทำงานของเครื่องจักรและเพิ่มประสิทธิภาพในการผลิตมากยิ่งขึ้น โดยใช้ข้อมูลสถานะการทำงานและพารามิเตอร์ที่เกี่ยวข้องของเครื่องจักรทั้งหมด มาวิเคราะห์เพื่อลดปัญหาเครื่องจักรหยุดทำงานในการผลิต โดยจัดอัลกอริทึมมาทำการทดสอบกับชุดข้อมูลของเครื่องจักร จำนวน 10,000 ข้อมูล และแบ่งชุดข้อมูลเป็น 2 ส่วนและเปรียบเทียบเพื่อวัดประสิทธิภาพโมเดลของอัลกอริทึมในแต่ละประเภท ในกระบวนการทดสอบด้วยอัลกอริทึมที่ใช้ในการจำแนกประเภทของข้อมูล รวมถึงการนำเสนอวิธีการในการเลือกคุณลักษณะ การจัดการเก็บข้อมูลที่ไม่สมดุล และทำการทดลองด้วยชุดข้อมูล

ในการเลือกคุณลักษณะใช้วิธีการ Embedded ทำให้สามารถลดจำนวนพารามิเตอร์ที่ไม่เกี่ยวข้องออกจากโมเดล เพื่อลดเวลาในการประมวลผลของอัลกอริทึม และใช้วิธีจัดการข้อมูลที่ไม่สมดุลด้วยเทคนิควิธีการสุ่มเพิ่มตัวอย่างกลุ่มน้อย (SMOTE) เพื่อสังเคราะห์ข้อมูลใหม่จากกลุ่มตัวอย่างเดิมให้มีจำนวนตัวอย่างมากขึ้น ช่วยทำให้เกิดการกระจายของกลุ่มข้อมูลมีความสมดุลมากยิ่งขึ้น และทำการทดสอบด้วยชุดข้อมูลทดสอบจำนวน 6,756 ข้อมูล

หลังจากได้ทำความสะอาดข้อมูล ทำการแบ่งชุดข้อมูลก่อนเข้าโมเดลด้วยวิธีการวิเคราะห์ความแม่นตรงของตัวแบบ (K-Fold Cross Validation) โดยการแบ่งชุดข้อมูลสำหรับการสอนออกเป็นจำนวน K ส่วน แบบสุ่มจำนวนเท่า ๆ กัน กำหนดค่า K = 10 ก่อนนำข้อมูลเข้าโมเดลจะช่วยทำให้ประสิทธิภาพของโมเดลในการวิเคราะห์ข้อมูลดีขึ้นและลดเวลาในการประมวลผลได้ดีขึ้นอีกด้วย

การวัดประสิทธิภาพของโมเดลด้วยค่าความถูกต้องของเครื่องจักรแบบโดยรวม (Accuracy) ค่าความแม่นยำ (Precision) ความระลึกได้ (Recall) และ F1-Score แสดงให้เห็นว่าประสิทธิภาพของอัลกอริทึม Gradient Boosting Machine มีประสิทธิภาพที่ดีมาก มีค่าความความถูกต้องของเครื่องจักรแบบโดยรวม (Accuracy) มากกว่า 90% หลังจากที่ได้จัดการเรื่องข้อมูลที่ไม่สมดุล นอกจากนี้ยังมีอัลกอริทึม Random Forest และอัลกอริทึม k-Nearest Neighbors ที่มีค่าความแม่นยำอยู่ในช่วง 70 – 80% ผลจากการทำนายโมเดลมีจำนวนข้อมูลค่าจริง กับจำนวนข้อมูลจากการทำนายมีผลลัพธ์ตรงกันอยู่ในระดับค่อนข้างสูง ถือว่ายังให้ประสิทธิภาพในการวิเคราะห์ข้อมูลได้ค่อนข้างดี สามารถนำอัลกอริทึมเหล่านี้ไปใช้งานกับชุดข้อมูลจำลองนี้ได้เช่นกัน

5.2 ข้อจำกัด

หลังจากที่ได้ทำการสรุปผลของการศึกษาวิธีการจำแนกประเภทความผิดปกติของเครื่องจักรที่ใช้ในกระบวนการผลิตพบข้อจำกัดดังนี้

1) ข้อมูลที่ได้มาเป็นชุดข้อมูลสังเคราะห์ที่ได้จากการสร้างแบบจำลอง จึงไม่มีข้อมูลจริงที่ได้จากการเก็บค่าพารามิเตอร์ของเครื่องจักรที่ใช้งานจริงในการผลิต

2) มีข้อจำกัดทางด้านข้อมูลเนื่องจากข้อมูลบางส่วนถือเป็นความลับทางธุรกิจของโรงงาน หรือไม่มีข้อมูลที่ถูกเก็บไว้ในระบบฐานข้อมูลของเครื่องจักร ผู้วิจัยจึงต้องใช้ข้อมูลสังเคราะห์ที่ใกล้เคียงกันในการทดลอง

5.3 ข้อเสนอแนะ

ข้อเสนอแนะในการศึกษาและพัฒนาวิธีการจำแนกประเภทความผิดปกติของเครื่องจักรจากข้อมูลการผลิต เป็นแนวคิดเบื้องต้นในการนำเอาตัวแบบพัฒนาต่อในอนาคต ซึ่งมีรายละเอียดดังต่อไปนี้

1. เนื่องจากชุดข้อมูลที่ได้มาจากระบบฐานข้อมูลการบำรุงรักษาเชิงคาดการณ์ที่พบในโรงงานอุตสาหกรรม ซึ่งข้อมูลที่ได้เป็นชุดข้อมูลสังเคราะห์ ถ้าหากมีการสุ่มตัวอย่างเพิ่มเติมจากเครื่องจักรที่ใช้ในกระบวนการผลิตใช้งานจริงมาร่วมในการทดลอง และนำพารามิเตอร์ที่อยู่ในเครื่องจักรมาวิเคราะห์ข้อมูล ทำให้ทราบถึงแนวโน้มวิธีการวิเคราะห์ข้อมูลของโมเดลให้กว้างขึ้น จะทำให้ผลลัพธ์ของตัวอัลกอริทึมมีความแม่นยำมากขึ้น

2. ในส่วนของวิธีการทำนายพยากรณ์สถานการณ์ทำงานของเครื่องจักร อาจจะเลือกใช้อัลกอริทึมประเภทอื่นที่สามารถทำการพยากรณ์ได้ เช่น อัลกอริทึมประเภทแบ่งกลุ่มข้อมูล เพื่อให้ได้การพยากรณ์ที่แม่นยำและตรงกับสถานการณ์ในกระบวนการผลิตมากยิ่งขึ้น

3. ในการเลือกใช้โปรแกรมเพื่อวิเคราะห์และทำนาย ควรเลือกใช้โปรแกรมที่ใช้ภาษาในการประมวลผลที่ง่ายต่อผู้พัฒนาระบบ เพื่อให้ง่ายต่อการต่อยอดและพัฒนาเทคโนโลยีให้เข้ากับกระบวนการผลิต

เอกสารอ้างอิง

พิภพ ลลิตาภรณ์. 2553. **การวางแผนและควบคุมการผลิต.** กรุงเทพฯ : สมาคมส่งเสริม เทคโนโลยี (ไทย – ญี่ปุ่น).

เศรษฐภูมิ เถาชารี. 2559. **การบริหารการผลิตด้วยกลยุทธ์ 5P.** [Online]. Available : http://www.thailandindustry.com/indust_newweb/onlinemag_preview.php?cid=550

Emily Himes. 2023. **Understanding the 6 Big Losses in Manufacturing.** [Online]. Available : https://www.ptc.com/en/blogs/iiot/understanding-the-six-big-losses-in-manufacturing

สมาคมโปรแกรมเมอร์ไทย. 2561. **การเรียนรู้ของเครื่อง (Machine Learning) และการเรียนรู้เชิงลึก (Deep Learning).** [Online]. Available : https://www.thaiprogrammer.org/2018/12/การเรียนรู้ของเครื่องmachine-le/

สมาคมโปรแกรมเมอร์ไทย. 2018. **อะไรคือการเรียนรู้ของเครื่อง(Machine Learning)? ฉบับมือใหม่.** [Online]. Available : https://www.thaiprogrammer.org/2018/12/อะไรคือ-การเรียนรู้ของเครื่อง/

ณัฐโชติ พรหมฤทธิ์, สัจจาภรณ์ ไวจรรยา. 2564. **Fundamental of Deep Learning in Practive.** กรุงเทพฯ : ไอดีซี พรีเมียร์.

สมาคมโปรแกรมเมอร์ไทย. 2561. **การเรียนรู้ของเครื่อง (Machine Learning) และการเรียนรู้เชิงลึก (Deep Learning).** [Online]. Available : https://www.thaiprogrammer.org/2018/12/การเรียนรู้ของเครื่องmachine-le/

รัสรินทร์ เมธาเฉลิมพัฒน์. 2022. **การประยุกต์ใช้ Machine Learning กับงานภาคอุตสาหกรรม (ตอนที่1).** [Online]. Available : https://www.nectec.or.th/news/news-public-document/machine-learning-manufact-1.html

อัศวิน สุรวัชโยธิน,วรภัทร ไพรีเกรง. 2021 “การสร้างตัวแบบการทำนายในการเลือกศึกษาต่อในระดับอุดมศึกษาโดยใช้เทคนิคแบบบูรณาการในการแก้ปัญหาการจำแนกข้อมูลที่ไม่สมดุลของกลุ่มผู้เรียน.” *Journal of Information Science and Technology.* 11(1) : 65-79

Jaime Lynn Speiser, Michael E. Miller, Janet Tooze, Edward Ip. 2019. "A comparison of random forest variable selection methods for classification prediction modeling." *Expert systems with application Journal.* 134(1) : 93–101.

Abid Ali Awan.2023. **What is Bagging in Machine Learning?.** [Online]. Available : https://www.datacamp.com/tutorial/what-bagging-in-machine-learning-a-guide-with-examples

Juan de Dios Sanz Bob, Pablo Garrido Martinez-Llop, Pablo Rubio Marcos, Alvaro Solano Jimanez, Javier Gomez Fernandez. 2024. "Prodiction of Degraded Infrastructure Conditions for Railway Operation." *MDPI Journals.* 24(8) : 5.

Dinesh Varma. 2019. **[Overview]: Ensemble Learning made simple.** [Online]. Available : https://towardsdatascience.com/overview-ensemble-learning-made-simple-d4ac0d13cb96

Sumit Saha. 2022. **LightGBM.** [Online]. Available : https://neptune.ai/blog/xgboost-vs-lightgbm

Yiheng Li, Weidong Chen. 2020. "A Comparative Performance Assessment of Ensemble Learning for Credit Scoring" *Mathematics Journals.* 2(1) : 8.

Siddharth Nandakumar Chikalkar. 2020. "K -NEAREST NEIGHBORS MACHINE LEARNING ALGORITHM" *International Journal of Creative Research Thoughts.* 8(12) : 1-5.

Sebastian Raschka. 2020. **Intro to Machine Learning.** [Online]. Available : https://sebastianraschka.com/pdf/lecture-notes/stat451fs20/02-knn__notes.pdf

สถาบันนวัตกรรมและธรรมาภิบาลข้อมูล. 2022. **Supervised, Unsupervised, Reinforcement Learning ต่างกันอย่างไร?.** [Online]. Available : https://digi.data.go.th/blog/supervised-unsupervised-reinforcement-learning/

Big Data Institute. 2020. **อีกขั้นของ k-means algorithm ที่สามารถแบ่งกลุ่มข้อมูลได้ทุกประเภท.** [Online]. Available : https://bdi.or.th/big-data-101/k-means-algorithm-for-clustering-large-data-sets-with-categorical-values/

Big Data Institute. 2020. **มาทำความรู้จักกับ การเรียนรู้แบบเสริมกำลัง (Reinforcement Learning).** [Online]. Available : https://bdi.or.th/big-data- 101/introduction-to-reinforcement-learning/

Sandeep Srikonda, William Robert Norris, Dustin Nottage, Ahmet Soylemezoglu. "Deep Reinforcement Learning for Autonomous Dynamic Skid Steer Vehicle Trajectory Tracking." *MDPI Journals.* 11(5) : 3. 2022.

He, H. and Garcia, E.A., 2009, Learning from Imbalanced Data, IEEE T. Knowl. Data. En. 21: pp1263-12824.

Qinge Xiao, Congbo Li, Ying Tang, Xingzheng Chen. 2021. "Energy Efficiency Modeling for Configuration-Dependent Machining via Machine Learning: A Comparative Study." *IEEE Trans. Automation Science and Engineering.* Vol. 18, No. 2, pp. 717-730, 2021.

Musagil Musabayli, Mohd Hafeez Osman, Michael Dirix. 2020. "Classification Model for predictive Maintenance of Small Steam Sterilizers." *The Institution of Engineering and Technology Journals.* 2(1) : 3-4.

I. EL HASSANI, C. EL MAZGUALDI, T. MASROUR. 2019. "Artificial Intelligence and Machine Learning to Predict and Improve Efficiency in Manufacturing Industry." Ph.D. dissertation, Moulay Ismail University.

Stephan Matzka. 2020. **AI4I 2020 Predictive Maintenance.** [Online]. Available : https://archive.ics.uci.edu/dataset/601/ai4i+2020+predictive+maintenance+dataset

ภาคผนวก

ภาคผนวก ก

ขั้นตอนการจัดเตรียมข้อมูลเพื่อใช้ในการสอนและทดสอบโมเดล

ในส่วนนี้เป็นขั้นตอนในการศึกษาข้อมูล ทำความสะอาดข้อมูล พร้อมทั้งจัดตรียมชุดข้อมูลเพื่อสามารถนำข้อมูลใช้ในการสอน และทดสอบโมเดล การศึกษาค้นคว้าอิสระได้จากแหล่งข้อมูลเว็บไซต์ Machine Learning Repository ของมหาวิทยาลัยแคลิฟอร์เนียเออร์ไวน์ มีดังต่อไปนี้

UDI	Product_ID	Type	Air_temperature_[K]	Process_temperature_[K]	Rotational_speed_[rpm]	Torque_[Nm]	Tool wear_[min]	Target	Failure_Type
1	M14860	M	298.1	308.6	1551	42.8	0	0	No Failure
2	L47181	L	298.2	308.7	1408	46.3	3	0	No Failure
3	L47182	L	298.1	308.5	1498	49.4	5	0	No Failure
4	L47183	L	298.2	308.6	1433	39.5	7	0	No Failure
5	L47184	L	298.2	308.7	1408	40	9	0	No Failure
6	M14865	M	298.1	308.6	1425	41.9	11	0	No Failure
7	L47186	L	298.1	308.6	1558	42.4	14	0	No Failure
8	L47187	L	298.1	308.6	1527	40.2	16	0	No Failure
9	M14868	M	298.3	308.7	1667	28.6	18	0	No Failure
10	M14869	M	298.5	309	1741	28	21	0	No Failure
11	H29424	H	298.4	308.9	1782	23.9	24	0	No Failure
12	H29425	H	298.6	309.1	1423	44.3	29	0	No Failure
13	M14872	M	298.6	309.1	1339	51.1	34	0	No Failure
14	M14873	M	298.6	309.2	1742	30	37	0	No Failure
15	L47194	L	298.6	309.2	2035	19.6	40	0	No Failure
16	L47195	L	298.6	309.2	1542	48.4	42	0	No Failure
17	M14876	M	298.6	309.2	1311	46.6	44	0	No Failure
18	M14877	M	298.7	309.2	1410	45.6	47	0	No Failure
19	H29432	H	298.8	309.2	1306	54.5	50	0	No Failure
20	M14879	M	298.9	309.3	1632	32.5	55	0	No Failure
21	H29434	H	298.9	309.3	1375	42.7	58	0	No Failure
22	L47201	L	298.8	309.3	1450	44.8	63	0	No Failure
23	M14882	M	298.9	309.3	1581	30.7	65	0	No Failure
24	L47203	L	299	309.4	1758	25.7	68	0	No Failure

รูปที่ ก.1 ข้อมูลจาก .csv ไฟล์

จากรูปที่ ก.1 แสดงตัวอย่างประเภทของข้อมูลแบบคร่าว ๆ พบว่ามีการเก็บข้อมูลโดยแบ่งตามรหัสของผลิตภัณฑ์ ชนิดของผลิตภัณฑ์ อุณหภูมิความร้อนที่ใช้ในการผลิตชิ้นงาน ความดัน ความเร็วรอบของมอเตอร์ และตัวแปรเป้าหมายมี 6 ประเภท คือ เครื่องจักรทำงานปกติ (No Failure) ความผิดปกติที่เกิดจากความร้อนสะสม (Heat Dissipation Failure) ความผิดปกติที่เกิดจากผลิตเกินกำลัง (Overstrain Failure) ความผิดปกติที่เกิดจากไฟฟ้ากำลัง (Power Failure) ความผิดปกติจากความไม่แน่นอนของเครื่องจักร (Random Failure) ความผิดปกติเนื่องจากการสึกหร่อของเครื่องจักร (Tool Wear Failure)

ก.1 ชุดข้อมูลการทดลอง

ก่อนทำความสะอาดข้อมูล จำนวนข้อมูลมีทั้งหมด 10,000 ข้อมูล ในขั้นตอนการทำความสะอาดข้อมูล เริ่มจากการศึกษาลักษณะข้อมูลดังรูปที่ ก.2 และทำความสะอาดชุดข้อมูลทั้งหมด

ในขั้นตอนของการทำความสะอาดข้อมูล เริ่มจากการเช็คจากชุดข้อมูลว่า มีข้อมูลส่วนใดว่างหรือไม่ ซึ่งในชุดข้อมูลที่นำมาทดลองไม่ปรากฏ จากนั้นทำการปรับขนาดข้อมูลให้อยู่ระหว่าง 0 ถึง 1 ด้วยการ Rescaling หรือ Min-Max Normalization และหาความสัมพันธ์ของตัวแปรต้นทั้ง 2 ตัว โดยใช้วิธี Pair plot ดังรูปที่ ก.3 พร้อมทั้งศึกษา Heatmap เพื่อดูความสัมพันธ์ของตัวแปรอิสระ ดังรูปที่ ก.4

```
[13] df.describe()
```

	UDI	Air_temperature_[K]	Process_temperature_[K]	Rotational_speed_[rpm]	Torque_[Nm]	Tool wear_[min]	Target
count	10000.00000	10000.000000	10000.000000	10000.000000	10000.000000	10000.000000	10000.000000
mean	5000.50000	300.004930	310.005560	1538.776100	39.986910	107.951000	0.033900
std	2886.89568	2.000259	1.483734	179.284096	9.968934	63.654147	0.180981
min	1.00000	295.300000	305.700000	1168.000000	3.800000	0.000000	0.000000
25%	2500.75000	298.300000	308.800000	1423.000000	33.200000	53.000000	0.000000
50%	5000.50000	300.100000	310.100000	1503.000000	40.100000	108.000000	0.000000
75%	7500.25000	301.500000	311.100000	1612.000000	46.800000	162.000000	0.000000
max	10000.00000	304.500000	313.800000	2886.000000	76.600000	253.000000	1.000000

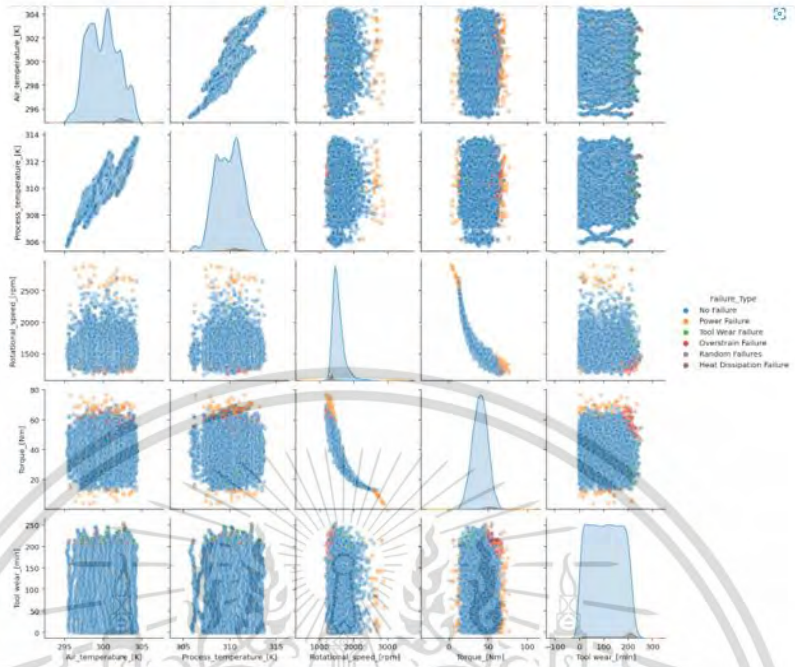
```
[7]  df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 10 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   UDI                     10000 non-null  int64
 1   Product_ID              10000 non-null  object
 2   Type                    10000 non-null  object
 3   Air_temperature_[K]     10000 non-null  float64
 4   Process_temperature_[K] 10000 non-null  float64
 5   Rotational_speed_[rpm]  10000 non-null  int64
 6   Torque_[Nm]             10000 non-null  float64
 7   Tool wear_[min]         10000 non-null  int64
 8   Target                  10000 non-null  int64
 9   Failure_Type            10000 non-null  object
dtypes: float64(3), int64(4), object(3)
memory usage: 781.4+ KB
```

```
[12] (df.shape)

(10000, 10)
```

```
[8]  df.Failure_Type.value_counts()

Failure_Type
No Failure                9652
Heat Dissipation Failure   112
Power Failure               95
Overstrain Failure          78
Tool Wear Failure           45
Random Failures             18
Name: count, dtype: int64
```

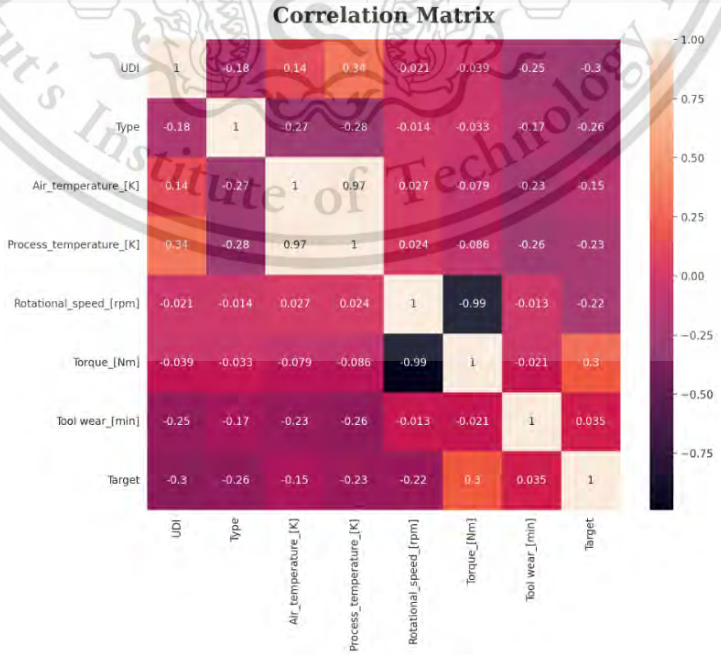
รูปที่ ก.2 ลักษณะข้อมูลแต่ละตัวแปร

```
sns.pairplot(df, vars=['Air_temperature_[K]',
       'Process_temperature_[K]', 'Rotational_speed_[rpm]', 'Torque_[Nm]','Tool wear_[min]'],
             hue='Failure_Type',
             plot_kws={'alpha': .4});
```



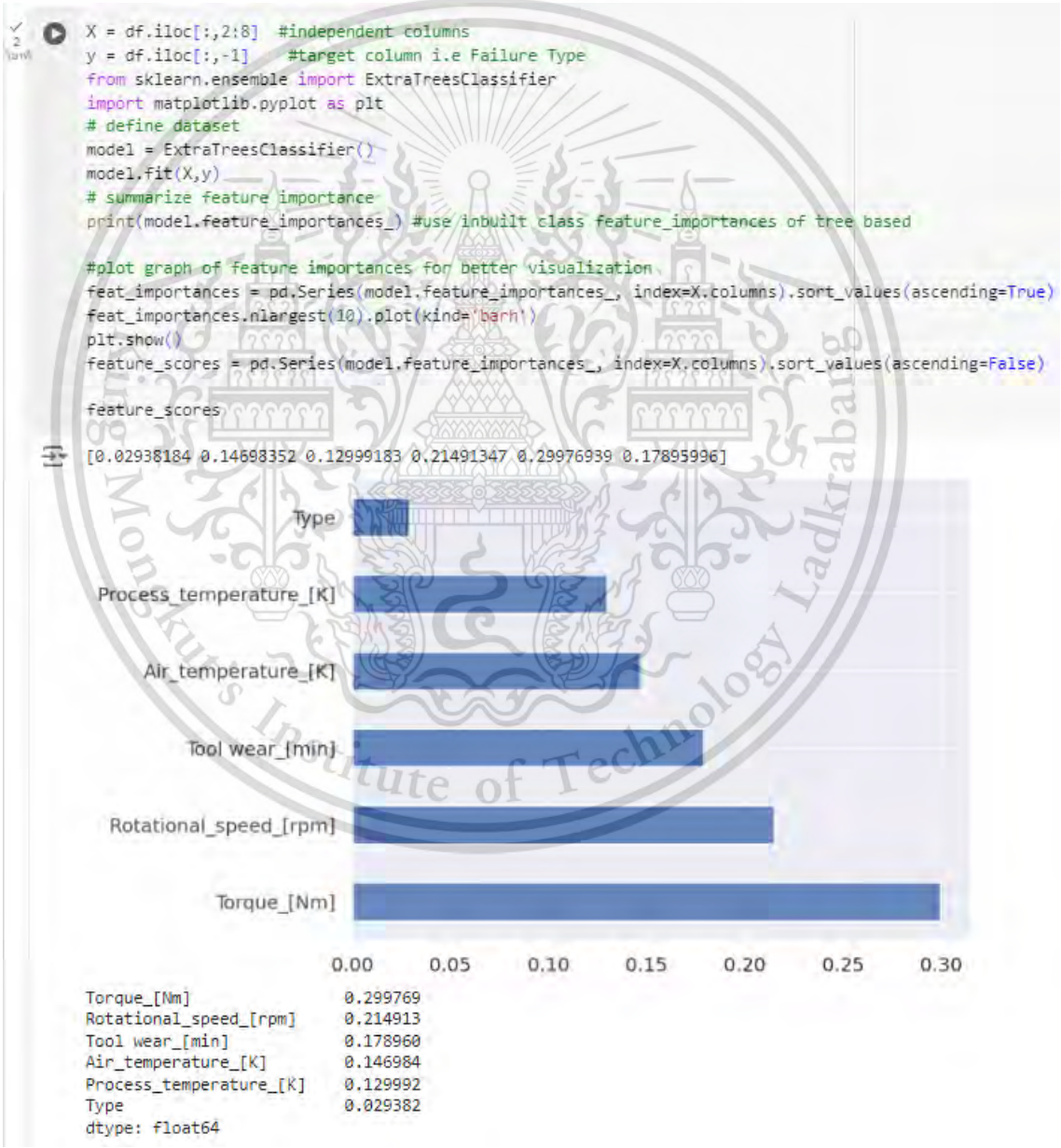
รูปที่ ก.3 หาความสัมพันธ์ระหว่างตัวแปร โดยใช้ pair plot

```
# Create correlation matrix
corr_matrix = df[["UDI","Type","Air_temperature_[K]", "Process_temperature_[K]","Rotational_speed_[rpm]", "Torque_[Nm]" , "Tool wear_[min]" , "Target"]].corr()

# Farbliche Abgrenzung hinzufügen
corr_matrix.style.background_gradient(cmap='coolwarm')
plt.figure(figsize=(10,8))
pl = sns.heatmap(corr_matrix.corr(), annot=True )
pl.set_title('Correlation Matrix', fontdict={'fontsize':20, 'fontfamily' : 'serif', 'fontweight' : 'bold'},
             pad=16)
plt.show()
```



รูปที่ ก.4 Heatmap ของชุดข้อมูล

จากนั้นทำการคัดเลือกคุณลักษณะของชุดข้อมูล เป็นวิธีการสำหรับเลือกตัวแปรที่สำคัญต่อโมเดล ว่าตัวแปรใดมีอิทธิพลต่อโมเดลมากที่สุด โดยใช้การ Import ไลบรารีของ skimage เพื่อคัดแยกคุณลักษณะสำคัญด้วยเทคนิคของอัลกอริทึม Extra Tree Classifier มาแยกคุณสมบัติในแต่ละตัวแปรสำคัญ จำนวนทั้งหมด 10 แบบ ซึ่งทำการแบ่งให้เป็นตัวแปรเป้าหมายเป็นสถานะการทำงานของเครื่องจักร ดังนั้นจะเหลือตัวแปรที่ใช้ในการพิจารณาเพียง 9 แบบเท่านั้น เมื่อทำการคัดแยกคุณลักษณะสำคัญของชุดข้อมูลเสร็จเรียบร้อยแล้ว จะได้ค่าความสำคัญของตัวแปรแต่ละตัว ดังรูปที่ ก.5 และตัดตัวแปรที่ไม่เกี่ยวข้องออกจากโมเดล เพื่อนำข้อมูลเหล่านี้ไปสร้างตัวแบบในขั้นตอนต่อไป



รูปที่ ก.5 การคัดแยกคุณลักษณะของข้อมูล

ขั้นตอนการเตรียมข้อมูลเพื่อใช้ในการสอน และทดสอบโมเดล เริ่มจากนำข้อมูลที่ทำความสะอาดเรียบร้อยแล้วมาแบ่งข้อมูลเป็นชุดข้อมูลสำหรับการสอนโมเดล และชุดข้อมูลสำหรับทดสอบโมเดล ในอัตราส่วน 70 และ 30 ตามลำดับ ดังรูปที่ ก.6 จากการทดลอง พบว่าข้อมูลตัวแปรเป้าหมายไม่มีความสมดุลกัน จึงเลือกใช้เทคนิคการสุ่มเพิ่มตัวอย่างกลุ่มน้อย (SMOTE) เพื่อแก้ปัญหาดังรูปที่ ก.7 และจากนั้นนำข้อมูลเหล่านี้ไปสร้างตัวแบบในขั้นตอนต่อไป

```
[35] # train-test split
     from sklearn.model_selection import train_test_split

     X_train, X_test, y_train, y_test = train_test_split(
         X, y, stratify=y, test_size=0.3, random_state=7)

     print(X_train.shape, X_test.shape, y_train.shape, y_test.shape)

     (7000, 9) (3000, 9) (7000,) (3000,)

[36] # Number of instances for each class
     from collections import Counter
     print(sorted(Counter(y).items()))

     [(0, 9652), (1, 112), (2, 78), (3, 95), (4, 18), (5, 45)]
```

รูปที่ ก.6 ชุดข้อมูลสำหรับการสอน และทดสอบโมเดลของข้อมูล

```
print("Before OverSampling status 'No Failure', counts of label '0': {} \n".format(sum(y_train==0)))
print("Before OverSampling status 'Heat Dissipation Failure', counts of label '1': {} \n".format(sum(y_train==1)))
print("Before OverSampling status 'Overstrain Failure', counts of label '2': {} \n".format(sum(y_train==2)))
print("Before OverSampling status 'Power Failure', counts of label '3': {} \n".format(sum(y_train==3)))
print("Before OverSampling status 'Random Failures', counts of label '4': {} \n".format(sum(y_train==4)))
print("Before OverSampling status 'Tool Wear Failure', counts of label '5': {} \n".format(sum(y_train==5)))

SEED = 42
# We now use SMOTE technique because of the imabalance in training data
from imblearn.over_sampling import SMOTE
oversample = SMOTE(random_state=SEED)
X_train_upsampled, y_train_upsampled = oversample.fit_resample(X_train, y_train.ravel())

print('After OverSampling, the shape of train_X: {}'.format(X_train_upsampled.shape))
print('After OverSampling, the shape of train_y: {} \n'.format(y_train_upsampled.shape))

print("After OverSampling status 'No Failure', counts of label '0': {}".format(sum(y_train_upsampled==0)))
print("After OverSampling status 'Heat Dissipation Failure', counts of label '1': {}".format(sum(y_train_upsampled==1)))
print("After OverSampling status 'Overstrain Failure', counts of label '2': {}".format(sum(y_train_upsampled==2)))
print("After OverSampling status 'Power Failure', counts of label '3': {}".format(sum(y_train_upsampled==3)))
print("After OverSampling status 'Random Failures', counts of label '4': {}".format(sum(y_train_upsampled==4)))
print("After OverSampling status 'Tool Wear Failure', counts of label '5': {}".format(sum(y_train_upsampled==5)))
```

```
Before OverSampling status 'No Failure', counts of label '0': 6756

Before OverSampling status 'Heat Dissipation Failure', counts of label '1': 78

Before OverSampling status 'Overstrain Failure', counts of label '2': 55

Before OverSampling status 'Power Failure', counts of label '3': 67

Before OverSampling status 'Random Failures', counts of label '4': 13

Before OverSampling status 'Tool Wear Failure', counts of label '5': 31

After OverSampling, the shape of train_X: (40536, 3)
After OverSampling, the shape of train_y: (40536,)

After OverSampling status 'No Failure', counts of label '0': 6756
After OverSampling status 'Heat Dissipation Failure', counts of label '1': 6756
After OverSampling status 'Overstrain Failure', counts of label '2': 6756
After OverSampling status 'Power Failure', counts of label '3': 6756
After OverSampling status 'Random Failures', counts of label '4': 6756
After OverSampling status 'Tool Wear Failure', counts of label '5': 6756
```

รูปที่ ก.7 เทคนิคการสุ่มเพิ่มตัวอย่างกลุ่มน้อยของข้อมูล

ภาคผนวก ข

ขั้นตอนการสอน ทดสอบ และปรับแต่งพารามิเตอร์ให้เหมาะสมกับโมเดล

ในขั้นตอนนี้เป็นส่วนการสอน ทดสอบ และปรับแต่งค่าพารามิเตอร์ให้เหมาะสมกับโมเดล แต่ละชุดการทดลองแบ่งออกเป็น 2 ส่วน คือส่วนที่ทำการสอน - ทดสอบโมเดล และส่วนที่ปรับแต่งค่าพารามิเตอร์มีดังต่อไปนี้

1) ศึกษาภาพรวมของข้อมูลที่ทำความสะอาดเรียบร้อยแล้วจากชุดข้อมูลการสอน และชุดข้อมูลการทดสอบเพื่อนำไปใช้กับโมเดล

2) ศึกษาผลการทดสอบประสิทธิภาพของโมเดลแต่ละชนิด ได้แก่ ค่า Accuracy, Precision, Recall และ F1-Scores พร้อมทั้งปรับแต่งค่าพารามิเตอร์ให้เหมาะสม

ข.1 ชุดข้อมูลการทดลอง

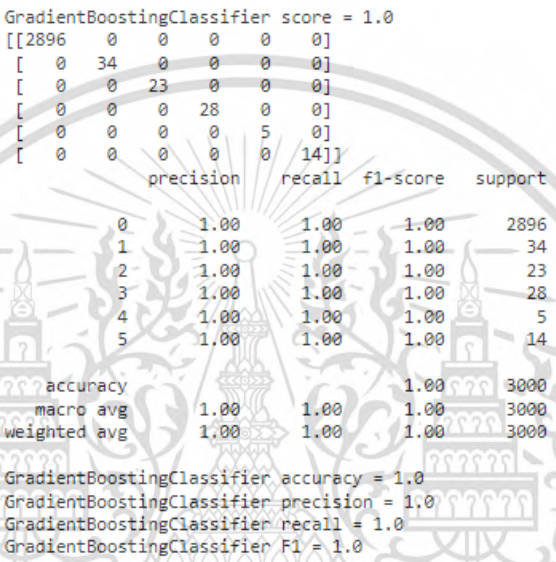
จากการศึกษาภาพรวมของข้อมูลที่ทำที่ทำความสะอาดเรียบร้อยแล้วจากชุดข้อมูลการสอน และชุดข้อมูลการทดสอบ นำไปสู่การสร้างโมเดล ดังรูปที่ ข.1 โดยทำไปทั้งหมด 7 โมเดล และเลือกมา 3 โมเดล ได้แก่ อัลกอริทึม Gradient Boosting Machine อัลกอริทึม k-Nearest Neighbors และอัลกอริทึม Random Forest เพื่อเปรียบเทียบประสิทธิภาพวิธีจำแนกกลุ่มข้อมูล โดยมีการกำหนดวิธีการวิเคราะห์ความแม่นตรงของตัวแบบ K=10

```
from sklearn.model_selection import KFold
kf = KFold(n_splits=10, random_state=42, shuffle=True)
```

```
algo = [[SVC(random_state=0),'SVC'],
    [AdaBoostClassifier(random_state=0),'AdaBoostClassifier'],
    [LogisticRegression(random_state=0),'LogisticRegression'],
    [KNeighborsClassifier(n_neighbors=10), 'KNeighborsClassifier'],
    [GradientBoostingClassifier(), 'GradientBoostingClassifier'],
    [RandomForestClassifier(), 'RandomForestClassifier'],
    [BaggingClassifier(), 'BaggingClassifier']
]
model_score=[]
for a in algo:
    model=a[0]
    model.fit(X_train_upsampled, y_train_upsampled) # step 2: fit
    y_pred=model.predict(X_test) # step 3: predict
    score=model.score(X_test, y_pred)
    model_cross_val = cross_val_score(estimator = model, X = X_train_upsampled, y = y_train_upsampled, cv=10 ) #K-Fold Validation
    model_accuracy = accuracy_score(y_test, y_pred)
    #model_roc = roc_auc_score(y_test, y_pred, average="macro")              # ROC AUC Score
    model_precision = precision_score(y_test, y_pred, average="macro")     # Precision Score
    model_recall = recall_score(y_test, y_pred, average="macro")           # Recall Score
    model_F1 = f1_score(y_test, y_pred, average="macro")                   # F1 Score
    model_score.append([score, a[1]])
    model_score.append([model_accuracy, a[1]])
    model_score.append([model_precision, a[1]])
    model_score.append([model_recall, a[1]])
    model_score.append([model_F1, a[1]])
    print(f'{a[1]} score = {score}') # step 4: score
    print(metrics.confusion_matrix(y_test, y_pred))
    print(metrics.classification_report(y_test, y_pred))
    print(f'{a[1]} accuracy = {model_accuracy}')
    print(f'{a[1]} precision = {model_precision}')
    print(f'{a[1]} recall = {model_recall}')
    print(f'{a[1]} F1 = {model_F1}')
    print('')
print(model_score)
print('-' * 100)
```

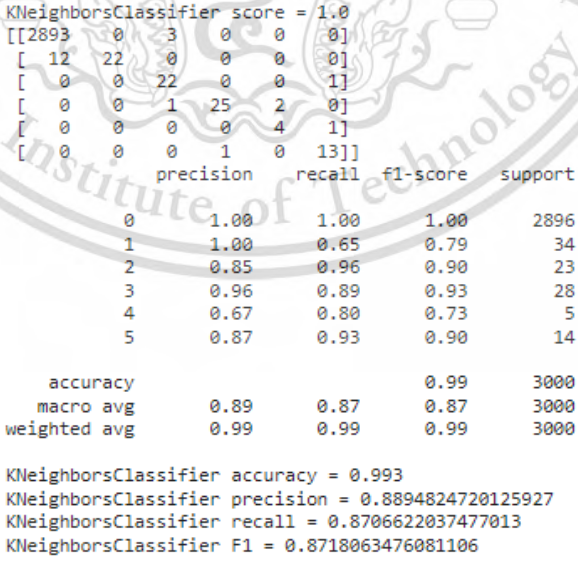
รูปที่ ข.1 ตัวอย่างการสอน และการทดสอบโมเดลเบื้องต้น

ผลการทดสอบก่อนปรับค่าพารามิเตอร์ ใน Classification Report พบว่า อัลกอริทึม Gradient Boosting Machine มีค่าความถูกต้อง 100% ที่ Precision = 100% Recall = 100% F1-Score = 100% ดังรูปที่ ข.2 อัลกอริทึม k-Nearest Neighbors มีค่าความถูกต้อง 99.30% ที่ Precision = 88.94% Recall = 87.06% F1-Score = 87.18% ดังรูปที่ ข.3 และอัลกอริทึม Random Forest มีค่าความถูกต้อง 99.93% ที่ Precision = 98.94% Recall = 95.94% และ F1-Score = 97.24% ดังรูปที่ ข.4



```
GradientBoostingClassifier score = 1.0
[[2896    0    0    0    0    0]
 [   0   34    0    0    0    0]
 [   0    0   23    0    0    0]
 [   0    0    0   28    0    0]
 [   0    0    0    0    5    0]
 [   0    0    0    0    0   14]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      2896
           1       1.00      1.00      1.00        34
           2       1.00      1.00      1.00        23
           3       1.00      1.00      1.00        28
           4       1.00      1.00      1.00         5
           5       1.00      1.00      1.00        14

    accuracy                           1.00      3000
   macro avg       1.00      1.00      1.00      3000
weighted avg       1.00      1.00      1.00      3000

GradientBoostingClassifier accuracy = 1.0
GradientBoostingClassifier precision = 1.0
GradientBoostingClassifier recall = 1.0
GradientBoostingClassifier F1 = 1.0
```

รูปที่ ข.2 ผลลัพธ์การทดสอบของโมเดล Gradient Boosting Machine ก่อนปรับพารามิเตอร์



```
KNeighborsClassifier score = 1.0
[[2893    0    3    0    0    0]
 [  12   22    0    0    0    0]
 [   0    0   22    0    0    1]
 [   0    0    1   25    2    0]
 [   0    0    0    0    4    1]
 [   0    0    0    1    0   13]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      2896
           1       1.00      0.65      0.79        34
           2       0.85      0.96      0.90        23
           3       0.96      0.89      0.93        28
           4       0.67      0.80      0.73         5
           5       0.87      0.93      0.90        14

    accuracy                           0.99      3000
   macro avg       0.89      0.87      0.87      3000
weighted avg       0.99      0.99      0.99      3000

KNeighborsClassifier accuracy = 0.993
KNeighborsClassifier precision = 0.8894824720125927
KNeighborsClassifier recall = 0.8706622037477013
KNeighborsClassifier F1 = 0.8718063476081106
```

รูปที่ ข.3 ผลลัพธ์การทดสอบของโมเดล k-Nearest Neighbors ก่อนปรับพารามิเตอร์

```
RandomForestClassifier score = 1.0
[[2896    0    0    0    0    0]
 [   0   34    0    0    0    0]
 [   0    1   22    0    0    0]
 [   0    0    0   28    0    0]
 [   0    0    0    1    4    0]
 [   0    0    0    0    0   14]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      2896
           1       0.97      1.00      0.99        34
           2       1.00      0.96      0.98        23
           3       0.97      1.00      0.98        28
           4       1.00      0.80      0.89         5
           5       1.00      1.00      1.00        14

    accuracy                           1.00      3000
   macro avg       0.99      0.96      0.97      3000
weighted avg       1.00      1.00      1.00      3000

RandomForestClassifier accuracy = 0.9993333333333333
RandomForestClassifier precision = 0.9894909688013137
RandomForestClassifier recall = 0.9594202898550724
RandomForestClassifier F1 = 0.9724383422323926
```

รูปที่ ข.4 ผลลัพธ์การทดสอบโมเดล Random Forest Classifier ก่อนปรับพารามิเตอร์

จากนั้นทำการปรับแต่งค่าพารามิเตอร์ในแต่ละโมเดล โดยใช้ GridSearchCV พบว่าค่าพารามิเตอร์ที่ดีที่สุดของโมเดล Gradient Boosting Machine มีเกณฑ์ใช้พารามิเตอร์ โดยกำหนดค่า Learning Rate = 0.01 จำนวน max_depth = 3 รายละเอียดดังรูปที่ ข.5 ส่วนโมเดล k-Nearest Neighbors กำหนด n_neighbors = 5 และmetric เป็นแบบ “Euclidean” ดังรูปที่ ข.6 การปรับแต่งค่าพารามิเตอร์ของโมเดล Random Forest มีเกณฑ์ใช้ค่า Max_depth = 6 จำนวน Estimate = 150 ดังรูปที่ ข.7 ผลการทดสอบหลังจากที่ได้มีการปรับค่าพารามิเตอร์ พบว่า โมเดล Gradient Boosting Machine มีค่าความถูกต้อง 93.30% ดังรูปที่ ข.8 และโมเดล k-Nearest Neighbors มีค่าความถูกต้อง 72.16% ดังรูปที่ ข.9 โมเดล Random Forest มีค่าความถูกต้อง 81.03% ดังรูปที่ ข.10

```
from sklearn.ensemble import GradientBoostingClassifier
from imblearn.under_sampling import RandomUnderSampler
from sklearn.model_selection import cross_validate
from imblearn.pipeline import Pipeline, make_pipeline
from imblearn.over_sampling import SMOTE
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import GridSearchCV
from sklearn.metrics import accuracy_score, classification_report, precision_score, f1_score, roc_auc_score, recall_score, confusion_matrix
from sklearn.metrics import precision_recall_fscore_support, roc_curve, precision_recall_curve
from sklearn import metrics
import pickle
sns.set()
from sklearn.model_selection import KFold
kf = KFold(n_splits=10, random_state=42, shuffle=True)
GBT_pipeline = make_pipeline(RandomUnderSampler(sampling_strategy='majority'),SMOTE(random_state=42),
                             GradientBoostingClassifier(random_state=13))

# use recall_weighted instead of recall because this is multiclass
grid_GBT = GridSearchCV(GBT_pipeline, param_grid=pipe_GBT_params, cv=kf,scoring='recall_weighted',return_train_score=True)
grid_GBT.fit(X_train_upsampled, y_train_upsampled)
```

```
grid_GBT.best_params_
```

```
{'gradientboostingclassifier__max_depth': 3,
 'gradientboostingclassifier__tol': 0.01}
```

รูปที่ ข.5 พารามิเตอร์ของโมเดล Gradient Boosting Machine

```
from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import cross_validate
from imblearn.pipeline import Pipeline, make_pipeline
from imblearn.over_sampling import SMOTE
from imblearn.under_sampling import RandomUnderSampler
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import GridSearchCV
from sklearn.metrics import accuracy_score, classification_report, precision_score, f1_score, roc_auc_score, recall_score, confusion_matrix
from sklearn.metrics import precision_recall_fscore_support, roc_curve, precision_recall_curve
import pickle
sns.set()
from sklearn.model_selection import KFold
kf = KFold(n_splits=10, random_state=42, shuffle=True)

knn = KNeighborsClassifier(n_neighbors=5, metric= 'euclidean')
knn.fit(X_train_upsampled,y_train_upsampled)
y_pred_KNN = knn.predict(X_test)
y_pred_probs_KNN = knn.predict_proba(X_test)
knn.score(X_train_upsampled, y_train_upsampled)
```

รูปที่ ข.6 พารามิเตอร์ของโมเดล k-Nearest Neighbors

```
from sklearn.model_selection import cross_validate
from imblearn.pipeline import Pipeline, make_pipeline
from imblearn.over_sampling import SMOTE
from imblearn.under_sampling import RandomUnderSampler
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import GridSearchCV
from sklearn.metrics import accuracy_score, classification_report, precision_score, f1_score, roc_auc_score, recall_score, confusion_matrix
from sklearn import metrics
from sklearn.metrics import precision_recall_fscore_support, roc_curve, precision_recall_curve
import pickle
sns.set()
from sklearn.model_selection import KFold
kf = KFold(n_splits=10, random_state=42, shuffle=True)
rf_pipeline = make_pipeline(RandomUnderSampler(sampling_strategy='majority'),SMOTE(random_state=42),
                            RandomForestClassifier(random_state=13))
# Cross validation metrics test data
# use recall_weighted instead of recall because this is multiclass
grid_rf = GridSearchCV(rf_pipeline, param_grid=pipe_rf_params, cv=kf,scoring='recall_weighted',
                       return_train_score=True)
grid_rf.fit(X_train_upsampled, y_train_upsampled)
```

```
grid_rf.best_params_

{'randomforestclassifier__max_depth': 12,
 'randomforestclassifier__n_estimators': 100,
 'randomforestclassifier__random_state': 13}
```

รูปที่ ข.7 พารามิเตอร์ของโมเดล Random Forest

```
Results for Gradient Boosting Classifier
              precision    recall  f1-score   support

           0       0.99      0.94      0.97      2896
           1       0.38      0.62      0.47        34
           2       0.63      0.74      0.68        23
           3       0.83      0.68      0.75        28
           4       0.00      0.00      0.00         5
           5       0.11      0.43      0.18        14

    accuracy                           0.93      3000
   macro avg       0.49      0.57      0.51      3000
weighted avg       0.97      0.93      0.95      3000

Macro roc auc (OvR): 0.8541
Macro f1 test set: 0.5057
Cross val macro precision score test data: 0.4341
Cross val macro recall score test data: 0.3795
```

รูปที่ ข.8 ผลลัพธ์การทดสอบของโมเดล Gradient Boosting Machine หลังปรับพารามิเตอร์

Results for Nearest Neighbors Classifier

	precision	recall	f1-score	support
0	0.99	0.73	0.84	2896
1	0.10	0.56	0.17	34
2	0.42	0.70	0.52	23
3	0.44	0.64	0.52	28
4	0.00	0.20	0.00	5
5	0.08	0.64	0.14	14
accuracy			0.72	3000
macro avg	0.34	0.58	0.37	3000
weighted avg	0.96	0.72	0.82	3000

Macro roc auc (OvR): 0.7744
Macro f1 test set: 0.3658
Cross val macro precision score test data: 0.4341
Cross val macro recall score test data: 0.3795

รูปที่ ข.9 ผลลัพธ์การทดสอบของโมเดล k-Nearest Neighbors หลังปรับพารามิเตอร์

Results for Random Forest by K-Fold Cross-Validation = 10]

	precision	recall	f1-score	support
0	0.99	0.82	0.90	2896
1	0.36	0.79	0.50	34
2	0.55	0.70	0.62	23
3	0.81	0.75	0.78	28
4	0.00	0.00	0.00	5
5	0.08	0.64	0.14	14
accuracy			0.82	3000
macro avg	0.46	0.62	0.49	3000
weighted avg	0.97	0.82	0.89	3000

Macro roc auc (OvR): 0.8601
Macro f1 test set: 0.4871
Cross val macro precision score test data: 0.4341
Cross val macro recall score test data: 0.3795

รูปที่ ข.10 ผลลัพธ์การทดสอบของโมเดล Random Forest หลังปรับพารามิเตอร์

หลังจากปรับแต่งพารามิเตอร์ให้เหมาะสมกับแต่ละโมเดลเรียบร้อยแล้ว จึงได้ทำการสอนและทดสอบโมเดลอีกรอบ ได้ผลลัพธ์ดังรูปที่ ข.8 ถึงรูปที่ ข.10 พบว่า โมเดล Gradient Boosting Machine ให้ค่าความถูกต้องสูงที่สุด และค่าความคลาดเคลื่อนต่ำที่สุด ตามด้วย Random Forest และ k-Nearest Neighbors ตามลำดับ

ประวัติผู้เขียน

ชื่อ	นางสาวอารีรัตน์ ชื่นชม
วัน เดือน ปีเกิด	20 กันยายน พ.ศ. 2535
ที่อยู่ปัจจุบัน	11/29 ซอยเสรีไทย 43 แยก 3 ถนนเสรีไทย แขวงคลองกุ่ม เขตบึงกุ่ม กทม.
ประวัติการศึกษา	(2558) วิศวกรรมศาสตรบัณฑิต สาขา วิศวกรรมอัตโนมัติ เกรดเฉลี่ย 2.59 สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ทุนการศึกษา	-
ผลงานทางวิชาการ	-