การทำนายการปรับตำแหน่งพนักงานโดยใช้การเรียนรู้ของเครื่อง

EMPLOYEE PROMOTION PREDICTION USING MACHINE LEARNING

ณัชชา ชินนาพันธ์

บัณฑิตวิทยาลัย มหาวิทยาลัยศรีนครินทรวิโรฒ

2565

การทำนายการปรับตำแหน่งพนักงานโดยใช้การเรียนรู้ของเครื่อง

ณัชชา ชินนาพันธ์

สารนิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
วิทยาศาสตรมหาบัณฑิต สาขาวิชาวิทยาการข้อมูล
คณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ
ปีการศึกษา 2565
ลิขสิทธิ์ของมหาวิทยาลัยศรีนครินทรวิโรฒ

EMPLOYEE PROMOTION PREDICTION USING MACHINE LEARNING

NATCHA CHINNAPAN

A Master's Project Submitted in Partial Fulfillment of the Requirements

for the Degree of MASTER OF SCIENCE

(Data Science)

Faculty of Science, Srinakharinwirot University

2022

Copyright of Srinakharinwirot University

สารนิพนธ์

เรื่อง

การทำนายการปรับตำแหน่งพนักงานโดยใช้การเรียนรู้ของเครื่อง

ของ

ณัชชา ชินนาพันธ์

ได้รับอนุมัติจากบัณฑิตวิทยาลัยให้นับเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิชาวิทยาการข้อมูล

ของมหาวิทยาลัยศรีนครินทรวิโรฒ

(รองศาสตราจารย์ นายแพทย์ฉัตรชัย เอกปัญญาสกุล)

คณบดีบัณฑิตวิทยาลัย

คณะกรรมการสอบปากเปล่าสารนิพนธ์

..... ที่ปรึกษาหลัก

(ผู้ช่วยศาสตราจารย์ ดร.วราภรณ์ วิยานนท์)

..... ประธาน

(ผู้ช่วยศาสตราจารย์ ดร.อัครา ประโยชน์)

..... กรรมการ

(ผู้ช่วยศาสตราจารย์ ดร.ศิริสรรพ เหล่าหะเกียรติ)

ชื่อเรื่อง	การทำนายการปรับตำแหน่งพนักงานโดยใช้การเรียนรู้ของเครื่อง
ผู้วิจัย	ณัชชา ชินนาพันธ์
ปริญญา	วิทยาศาสตรมหาบัณฑิต
ปีการศึกษา	2565
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ ดร. วราภรณ์ วิยานนท์

งานวิจัยนี้มีวัตถุประสงค์เพื่อศึกษาการทำนายการปรับตำแหน่งพนักงานโดยใช้การเรียนรู้ของเครื่อง เพื่อสร้างแบบจำลองการทำนายการปรับตำแหน่งพนักงาน โดยใช้ชุดข้อมูลการปรับตำแหน่งพนักงานปี 2562 - 2565 จำนวน 212 คน ของฝ่ายบริหารทรัพยากรบุคคล บริษัท ไทยสมุทรประกันชีวิต จำกัด (มหาชน) ผู้วิจัยได้ศึกษาและเปรียบเทียบประสิทธิภาพของแบบจำลอง ได้แก่ Support Vector Machine, Decision Tree, Random Forest และ XGBoost ร่วมกับการคัดเลือกฟีเจอร์ (Feature Selection) โดยใช้เทคนิค Recursive Feature Elimination (RFE) และการจัดความไม่สมดุลของข้อมูลด้วยวิธีการสุ่มตัวอย่างเพิ่มกลุ่มน้อยด้วยการสร้างตัวอย่างสังเคราะห์ (Synthetic Minority Over-sampling Technique หรือ SMOTE) โดยมีการวัดประสิทธิภาพของแบบจำลองจากค่า Accuracy, Precision, Recall และ F1 Score จากการทดสอบแบบจำลองที่ได้ประสิทธิภาพสูงสุด คือ แบบจำลอง Support Vector Machines และ แบบจำลอง Random Forest ที่แบ่งข้อมูล 80:20 ร่วมกับการคัดเลือกฟีเจอร์ ได้ค่า Accuracy = 93.02%, Precision = 94.74%, Recall = 97.30% และ F1-Score = 96.00% ที่เท่ากัน ซึ่งการคัดเลือกฟีเจอร์ก่อนการสร้างแบบจำลองให้ประสิทธิภาพดีที่สุดเมื่อเปรียบเทียบประสิทธิภาพของแบบจำลองที่ใช้เทคนิคอื่นร่วมด้วย

คำสำคัญ : การปรับตำแหน่งพนักงาน, การทำนาย, การเรียนรู้ของเครื่อง

Title	EMPLOYEE PROMOTION PREDICTION USING MACHINE LEARNING
Author	NATCHA CHINNAPAN
Degree	MASTER OF SCIENCE
Academic Year	2022
Thesis Advisor	Assistant Professor Waraporn Viyanon , Ph.D.

This research aimed to develop a machine learning based predictive model for employee promotion using data from the Human Resources Management Division of Ocean Life Insurance Public Company Limited. The study focused on 212 employees positions from 2019 to 2022. To achieve the research goal, the researcher investigated the efficacy of four models: Support Vector Machine, Decision Tree, Random Forest and XGBoost. Feature selection was done through Recursive Feature Elimination (RE) and Synthetic Minority Over-sampling Technique (SMOTE) was used to address imbalanced datasets. The study employed an 80:20 data split for training and testing the models. The Support Vector Machines model and the Random Forest model, with feature selection, produced the best results with 93.02% accuracy, 94.74% precision, 97.30% recall, and 96.00% F1-Score. The study found that the Support Vector Machines model and the Random Forest model, with feature selection, outperformed the other models, and feature selection significantly improved the models performance.

Keyword : Employee promotion, Predictive model, Support Vector Machine, Decision Tree, Random Forest, XGBoost, Feature selection

กิตติกรรมประกาศ

การจัดทำวิจัยฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี จากการได้รับความช่วยเหลือ และแนะนำของ ผศ.ดร.วราภรณ์ วิยานนท์ ซึ่งเป็นอาจารย์ที่ปรึกษาที่ได้กรุณาให้คำแนะนำ ข้อคิดเห็น ตรวจสอบ และแก้ไขร่างสารนิพนธ์มาโดยตลอด

ขอขอบคุณคณาจารย์ทุกท่านในสาขาวิชาวิทยาการข้อมูล คณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ ที่ให้ความรู้ และแนะนำแนวทางในการทำสารนิพนธ์นี้

ขอขอบคุณคณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ สำหรับทุนพัฒนาศักยภาพทางวิชาการนิสิตระดับบัณฑิตศึกษา ในการจัดทำสารนิพนธ์นี้ รวมถึงการสนับสนุนจากบัณฑิตวิทยาลัย มหาวิทยาลัยศรีนครินทรวิโรฒ ในการนำเสนอผลงานวิจัย

ขอขอบคุณฝ่ายบริหารทรัพยากรบุคคล บริษัท ไทยสมุทรประกันชีวิต จำกัด (มหาชน) ที่ให้ความอนุเคราะห์ข้อมูลที่ใช้ในการดำเนินการวิจัย

ขอขอบคุณทีม Career Development และทุกท่านในฝ่ายบริหารทรัพยากรบุคคล บริษัท ไทยสมุทรประกันชีวิต จำกัด (มหาชน) ที่ให้ความช่วยเหลือ และให้คำแนะนำในการทำการวิจัย

ขอขอบคุณ นายพรภวิษย์ สารบูรณ์ และนางสาวอมรกานต์ ใจดี ที่ให้ความช่วยเหลือ ให้คำปรึกษา ทั้งการเรียน การจัดทำสารนิพนธ์ และร่วมทำกิจกรรมการเรียนตลอดระยะเวลาการศึกษา ทำให้การเรียน และการจัดทำสารนิพนธ์

ขอขอบคุณเพื่อนในสาขาวิชาวิทยาการข้อมูล รุ่น 2564 ทุกท่าน ที่ให้ความช่วยเหลือ และให้คำแนะนำในการเรียน

และที่สำคัญขอขอบคุณครอบครัวที่เป็นกำลังใจ แรงผลักดัน ให้คำปรึกษา และสนับสนุนตลอดระยะเวลาการศึกษาและการจัดทำสารนิพนธ์จนสำเร็จลุล่วง

ณัชชา ชินนาพันธ์

สารบัญ

หน้า

บทคัดย่อภาษาไทย ..... ง

บทคัดย่อภาษาอังกฤษ ..... จ

กิตติกรรมประกาศ ..... ฉ

สารบัญ ..... ช

สารบัญตาราง ..... ฌ

สารบัญรูปภาพ ..... ญ

บทที่ 1 บทนำ 1

ภูมิหลัง 1

ความมุ่งหมายของงานวิจัย 3

ความสำคัญของการวิจัย 3

ขอบเขตของงานวิจัย 3

ประชากรที่ใช้ในการวิจัย 3

กลุ่มตัวอย่างที่ใช้ในการวิจัย 3

ตัวแปรที่ศึกษา 3

กรอบแนวคิดในงานวิจัย 4

สมมุติฐานในการวิจัย 5

บทที่ 2 ทบทวนวรรณกรรม 6

เทคนิคการเรียนรู้ของเครื่อง (Machine Learning Techniques) 6

แบบจำลองอัลกอริธึมที่ใช้ในเทคนิคการเรียนรู้ของเครื่องที่ใช้ในงานวิจัย 8

การปรับช่วงขอบเขตฟีเจอร์ (Feature Scaling) 10

การคัดเลือกฟีเจอร์ (Feature Selection) 11

การจัดการความไม่สมดุลของข้อมูล (Imbalanced data) 12
การปรับค่าพารามิเตอร์ (Hyperparameter Tuning) 13
งานวิจัยที่เกี่ยวข้อง 13
บทที่ 3 วิธีดำเนินการวิจัย 20
การกำหนดประชากรและกลุ่มตัวอย่าง 20
การออกแบบโครงสร้างและวิธีการดำเนินงานวิจัย 24
การเตรียมข้อมูลและการวิเคราะห์ข้อมูล 25
การแบ่งชุดข้อมูลและการปรับช่วงขอบเขตฟีเจอร์ 42
การคัดเลือกฟีเจอร์และการจัดการความไม่สมดุลของข้อมูล 43
การสร้างแบบจำลอง และการประเมินประสิทธิภาพแบบจำลอง 45
บทที่ 4 ผลการศึกษา 49
ผลลัพธ์ของการวิเคราะห์ข้อมูล 49
ผลลัพธ์ของการวิเคราะห์ฟีเจอร์ 49
ผลลัพธ์ของการปรับจูนพารามิเตอร์ 51
ผลลัพธ์ของการสร้างแบบจำลอง 54
บทที่ 5 สรุป อภิปรายผล และข้อเสนอแนะ 63
สรุปผลการวิจัย 63
อภิปรายผลการวิจัย 63
ข้อเสนอแนะ 65
บรรณานุกรม 67
ประวัติผู้เขียน 70

สารบัญตาราง

หน้า

ตาราง 1 แสดงผลการทดลองของงานวิจัยที่เกี่ยวข้อง..... 17

ตาราง 2 รายละเอียดของชุดข้อมูลการปรับตำแหน่ง..... 20

ตาราง 3 การกำหนดพารามิเตอร์ของแต่ละอัลกอริธึม..... 46

ตาราง 4 พารามิเตอร์หลังการปรับจูนของแต่ละแบบจำลอง..... 51

ตาราง 5 แสดงประสิทธิภาพการทดสอบจากแบบจำลอง Support Vector Machine..... 55

ตาราง 6 แสดงประสิทธิภาพการทดสอบจากแบบจำลอง Decision Tree..... 56

ตาราง 7 แสดงประสิทธิภาพการทดสอบจากแบบจำลอง Random Forest..... 58

ตาราง 8 แสดงประสิทธิภาพการทดสอบจากแบบจำลอง XGBoost..... 59

ตาราง 9 แสดงการเปรียบเทียบความแม่นยำจากการทดสอบแบบจำลอง..... 61

สารบัญรูปภาพ

หน้า

ภาพประกอบ 1 แผนผังความแตกต่างของเทคนิคการเรียนรู้ของเครื่อง..... 6
ภาพประกอบ 2 การแบ่งกลุ่มข้อมูล 2 ประเภท ด้วยเส้นแบ่ง maximum margin 8
ภาพประกอบ 3 ขั้นตอนของ Decision Tree 9
ภาพประกอบ 4 ขั้นตอนของ Random Forest..... 9
ภาพประกอบ 5 การทำงานของการ Boosting..... 10
ภาพประกอบ 6 วิธีการ Feature Selection 12
ภาพประกอบ 7 ขั้นตอนการสร้างแบบจำลอง และการดำเนินการวิจัย..... 24
ภาพประกอบ 8 ตัวอย่างข้อมูลในตารางของชุดข้อมูลดั้งเดิม 26
ภาพประกอบ 9 รายละเอียดของชุดข้อมูลดั้งเดิม 26
ภาพประกอบ 10 รายละเอียดชุดข้อมูลที่ได้รับการเปลี่ยนชื่อคอลัมน์ 27
ภาพประกอบ 11 รายละเอียดของชุดข้อมูลที่ได้รับเปลี่ยนชนิดข้อมูลและปรับแต่งให้เหมาะสมแล้ว 27
ภาพประกอบ 12 ตรวจสอบจำนวนค่าว่าง และจำนวนแถวข้อมูลที่ซ้ำกัน 28
ภาพประกอบ 13 ตรวจสอบพนักงานที่มีอายุงานในองค์กร 1 ปี และการแทนที่ค่าว่างด้วย 0 28
ภาพประกอบ 14 รายละเอียดค่าทางสถิติเบื้องต้น..... 29
ภาพประกอบ 15 จำนวนข้อมูลผลการปรับตำแหน่ง 29
ภาพประกอบ 16 กราฟแท่งแสดงจำนวนข้อมูลผลการปรับตำแหน่ง 30
ภาพประกอบ 17 จำนวนข้อมูลของแต่ละตำแหน่ง..... 30
ภาพประกอบ 18 กราฟแท่งแสดงจำนวนของแต่ละตำแหน่งกับผลการปรับตำแหน่ง 31
ภาพประกอบ 19 จำนวนข้อมูลของแต่ละสายงาน 31
ภาพประกอบ 20 กราฟแท่งแสดงจำนวนของแต่ละสายงานกับผลการปรับตำแหน่ง..... 32

ภาพประกอบ 21 จำนวนข้อมูลของแต่ละระดับการศึกษา 32

ภาพประกอบ 22 กราฟแท่งแสดงจำนวนของแต่ละระดับการศึกษากับผลการปรับตำแหน่ง..... 33

ภาพประกอบ 23 จำนวนข้อมูลของแต่ละเพศ..... 33

ภาพประกอบ 24 กราฟวงกลมแสดงสัดส่วนของเพศกับผลการปรับตำแหน่ง 34

ภาพประกอบ 25 การจัดกลุ่มช่วงอายุและจำนวนข้อมูลของแต่ละช่วงอายุ 34

ภาพประกอบ 26 กราฟแท่งแสดงจำนวนของแต่ละช่วงอายุกับผลการปรับตำแหน่ง 35

ภาพประกอบ 27 การจัดกลุ่มอายุงานจำนวนข้อมูลของแต่ละกลุ่มอายุงาน 35

ภาพประกอบ 28 กราฟแท่งแสดงจำนวนของแต่ละกลุ่มอายุงานกับผลการปรับตำแหน่ง 36

ภาพประกอบ 29 การจัดกลุ่มคะแนนการปฏิบัติงานของปีก่อนหน้าและจำนวนข้อมูลของแต่ละกลุ่มคะแนนการปฏิบัติงานของปีก่อนหน้า 36

ภาพประกอบ 30 กราฟแท่งแสดงจำนวนของแต่ละกลุ่มคะแนนการปฏิบัติงานของปีก่อนหน้ากับผลการปรับตำแหน่ง 37

ภาพประกอบ 31 การจัดกลุ่มคะแนนการประเมินสมรรถนะและทักษะของปีก่อนหน้าและจำนวนข้อมูลของแต่ละกลุ่มคะแนนการประเมินสมรรถนะและทักษะของปีก่อนหน้า 37

ภาพประกอบ 32 กราฟแท่งแสดงจำนวนของกลุ่มคะแนนการประเมินสมรรถนะและทักษะ 38

ภาพประกอบ 33 การจัดกลุ่มคะแนนการประเมินรอบทิศแบบ 360 องศา และจำนวนข้อมูลของแต่ละกลุ่มคะแนนการประเมินรอบทิศแบบ 360 องศา..... 38

ภาพประกอบ 34 กราฟแท่งแสดงจำนวนของแต่ละกลุ่มคะแนนการประเมินรอบทิศ..... 39

ภาพประกอบ 35 การจัดกลุ่มคะแนนผลสัมภาษณ์จากคณะกรรมการและจำนวนข้อมูลของแต่ละกลุ่มคะแนนผลสัมภาษณ์จากคณะกรรมการ..... 39

ภาพประกอบ 36 กราฟแท่งแสดงจำนวนของแต่ละกลุ่มคะแนนผลสัมภาษณ์จากคณะกรรมการกับผลการปรับตำแหน่ง..... 40

ภาพประกอบ 37 คำสั่งการแปลงข้อมูลให้เป็นตัวเลข โดยใช้Label encoder..... 41

ภาพประกอบ 38 ตัวอย่างข้อมูลในตารางของชุดข้อมูลที่ทำการแปลงข้อมูลให้เป็นตัวเลข 41

ภาพประกอบ 39 คำสั่งการหาค่าความสัมพันธ์ของข้อมูล 41
ภาพประกอบ 40 ค่าความสัมพันธ์ของข้อมูล (Correlation) 42
ภาพประกอบ 41 ตัวอย่างข้อมูลที่ทำการปรับช่วงขอบเขตของฟีเจอร์ 43
ภาพประกอบ 42 กราฟแท่งแสดงจำนวนข้อมูลผลการปรับตำแหน่งก่อนการทำ SMOTE 44
ภาพประกอบ 43 กราฟแท่งแสดงจำนวนข้อมูลผลการปรับตำแหน่งหลังการทำ SMOTE 44
ภาพประกอบ 44 กรอบการทำงานของ confusion matrix 47
ภาพประกอบ 45 คะแนนความสำคัญของฟีเจอร์ของแบบจำลอง Decision Tree 50
ภาพประกอบ 46 คะแนนความสำคัญของฟีเจอร์ของแบบจำลอง Random Forest 50
ภาพประกอบ 47 คะแนนความสำคัญของฟีเจอร์ของแบบจำลอง XGBoost 51
ภาพประกอบ 48 กราฟแท่งแสดงประสิทธิภาพการทดสอบจากแบบจำลอง Support Vector Machine 55
ภาพประกอบ 49 กราฟแท่งแสดงประสิทธิภาพการทดสอบจากแบบจำลอง Decision Tree 57
ภาพประกอบ 50 กราฟแท่งแสดงแสดงประสิทธิภาพการทดสอบจากแบบจำลอง Random Forest 58
ภาพประกอบ 51 กราฟแท่งแสดงประสิทธิภาพการทดสอบจากแบบจำลอง XGBoost 60
ภาพประกอบ 52 กราฟแท่งแสดงการเปรียบเทียบความแม่นยำจากการทดสอบแบบจำลอง 62

บทที่ 1
บทนำ

ภูมิหลัง

ทรัพยากรบุคคลถือเป็นปัจจัยหลักปัจจัยหนึ่งที่มีความสำคัญอย่างยิ่งต่อองค์กร เป็นปัจจัยที่ขับเคลื่อนองค์กรให้ดำเนินงานต่างๆ ให้เป็นไปตามเป้าหมายที่ตั้งไว้ให้ประสบความสำเร็จ รวมถึงยังเป็นส่วนสำคัญในการผลักดันกลยุทธ์ขององค์กร และวิสัยทัศน์ของผู้บริหารให้เกิดผลสำเร็จอีกด้วย

ในภาวการณ์การแข่งขันเพื่อได้มาซึ่งทรัพยากรบุคคลที่มีความสามารถของแต่ละองค์กร ทำให้การบริหารทรัพยากรบุคคลในองค์กรจึงเป็นเรื่องที่สำคัญ และมีความท้าทายที่เพิ่มขึ้นเป็นอย่างมากของฝ่ายบริหารทรัพยากรบุคคล ตั้งแต่การสรรหา การคัดเลือกบุคลากรเข้ามาร่วมงาน การรักษาพนักงานให้อยู่กับองค์กรให้ยาวนาน การฝึกอบรมและพัฒนาในด้านต่างๆ เพื่อให้บุคลากรที่รับเข้าเป็นพนักงานเกิดทักษะ ความรู้ ความชำนาญในตำแหน่งงานนั้นๆ เพื่อให้สามารถปฏิบัติงานได้อย่างมีคุณภาพและเกิดประสิทธิผลตามที่องค์กรคาดหวัง รวมถึงการสร้างความพึงพอใจให้แก่พนักงาน เพื่อดึงดูดให้พนักงานอยู่กับองค์กรต่อไปให้ยาวนานที่สุด แต่หากไม่สามารถรักษาพนักงานที่มีประสิทธิภาพเอาไว้ได้ ก็จะส่งผลกระทบต่อองค์กรในหลายๆ ทาง อาทิเช่น การดำเนินงานที่ไม่มีความต่อเนื่อง การสะสมของงานในตำแหน่งงานนั้นๆ และย้อนกลับมาส่งผลกระทบต่อฝ่ายทรัพยากรบุคคลโดยตรงในการสรรหาบุคลากรใหม่มาทดแทน ทำให้เสียทั้งกำลังคนและเงินในกระบวนการสรรหาบุคคลการตลอดจนการฝึกอบรมและพัฒนาพนักงานใหม่

การที่องค์กรจะรักษาพนักงาน สร้างความพึงพอใจ และดึงดูดให้พนักงานอยู่กับองค์กรไปอย่างยาวนาน เพื่อให้สามารถสร้างผลงานที่มีคุณภาพและทำงานได้ตรงตามประสิทธิภาพขององค์กรนั้น ขึ้นอยู่กับปัจจัยหลายอย่าง อาทิเช่น การได้รับมอบหมายงานตามความรู้ ความสามารถ การได้รับมอบหมายงานที่ได้พัฒนาตนเอง การได้รับการอบรมและพัฒนาความสามารถ บรรยากาศในการทำงานที่ดี เพื่อนร่วมงานที่ดี การเป็นที่ยอมรับในองค์กร และที่สำคัญคือการมองเห็นความก้าวหน้าในอาชีพการทำงานของตนเอง และได้รับผลตอบแทนที่ยุติธรรม

ความก้าวหน้าและการเติบโตในอาชีพการทำงาน เป็นสิ่งที่พนักงานควรได้รับ องค์กรที่มีโครงสร้างการปฏิบัติงานที่ชัดเจน จะมีรายละเอียดของตำแหน่งงานแต่ละตำแหน่งงาน มีแนวทางการปรับตำแหน่งพนักงาน และการประเมินศักยภาพและพฤติกรรมของพนักงานด้านต่างๆ ในแต่ละปี เพื่อเก็บเป็นข้อมูลในการพิจารณาปรับตำแหน่งงาน และประเมินการปรับผลตอบแทน

ประจำปี ซึ่งข้อมูลที่นำมาใช้พิจารณาจะมาจากผลการปฏิบัติงาน ความสามารถในการพัฒนาความรู้ ความสามารถของตนเอง พฤติกรรมที่แสดงออกตามวัฒนธรรมขององค์กร และอาจรวมไปถึงการเป็นที่ยอมรับของเพื่อนร่วมงาน และหัวหน้างาน

การวัดผลการปฏิบัติงาน (KPI) เป็นการวัดผลความสามารถของพนักงานในการปฏิบัติงานที่ตนเองได้รับมอบหมายไปจนถึงงานที่ได้รับมอบหมายให้เป็นความท้าทายใหม่ตามแผนการปฏิบัติงานประจำปี โดยมีเกณฑ์การวัดผล คือ ความชำนาญในการปฏิบัติงาน การดำเนินงานตามที่ได้รับมอบหมายให้สำเร็จตามวัตถุประสงค์ และเป็นไปตามระยะเวลาที่กำหนด รวมถึงความสามารถในการพัฒนางานที่ปฏิบัติอยู่ให้ดียิ่งขึ้น หากพนักงานมีผลการปฏิบัติงานที่สูงจะสะท้อนให้เห็นถึงความชำนาญในงานที่ปฏิบัติ ความรับผิดชอบต่องานที่ได้รับมอบหมาย การพัฒนาการปฏิบัติงานของตนเอง และสะท้อนให้เห็นถึงความพร้อมในการรับมอบหมายงานใหม่ๆ ที่จะเป็นงานจากตำแหน่งงานที่สูงขึ้นในอนาคต ตลอดจนสะท้อนให้เห็นว่าองค์กรมีความได้เปรียบในเชิงการแข่งขันมากกว่าองค์กรอื่น

สมรรถนะและทักษะ (Competency) เป็นการวัดความรู้ ทักษะ และทัศนคติที่จำเป็นต่อการทำงานให้มีประสิทธิภาพ โดยต้องสอดคล้องกับวิสัยทัศน์ พันธกิจ ค่านิยม และกลยุทธ์ขององค์กร รวมถึงต้องสอดคล้องกับหน้าที่และความรับผิดชอบที่กำหนดไว้ของแต่ละตำแหน่งงาน ซึ่งสามารถแบ่งตัวชี้วัดการประเมินสมรรถนะและทักษะ ได้ 3 ประเภท

1. ศักยภาพหลัก (core competency) ซึ่งเป็นทักษะ ความรู้ ความสามารถที่ทุกคนในองค์กรต้องมี

2. ศักยภาพสำหรับผู้บริหาร (leadership competency)

3. ศักยภาพสำหรับฝ่ายงานหรือหน่วยงาน (functional competency)

ตัวชี้วัดการประเมินสมรรถนะและทักษะจะใช้เป็นเกณฑ์ในการสรรหาบุคลากรที่จะเข้ามาร่วมงานกับองค์กร รวมถึงใช้ประเมินสมรรถนะและทักษะของพนักงานในองค์กร เพื่อใช้ในการวางแผนการพัฒนาของแต่ละบุคคลได้ตรงกับความต้องการของฝ่ายงานและองค์กร หากพนักงานมีผลการประเมินสมรรถนะและทักษะที่สูงจะสะท้อนให้เห็นถึงความพร้อมในการเข้ารับการประเมินเพื่อปรับตำแหน่งการทำงานที่สูงขึ้นในอนาคต

จากสิ่งที่กล่าวมาข้างต้นจะเห็นได้ว่าทุกๆ ปีพนักงานในองค์กรจะได้รับการประเมินในด้านต่างๆ ตั้งแต่การประเมินผลการปฏิบัติงาน การประเมินสมรรถนะและทักษะ หรืออาจรวมไปถึงการประเมินพฤติกรรมที่แสดงออกกับเพื่อนร่วมงาน และหัวหน้างาน เพื่อใช้เป็นข้อมูลการพิจารณาปรับตำแหน่งงาน ซึ่งฝ่ายบริหารทรัพยากรบุคคลต้องพิจารณาข้อมูลของพนักงานทั้งองค์กรเพื่อคัด

กรอง และคัดเลือกพนักงานที่มีความเหมาะสมสำหรับปรับตำแหน่งงาน ดังนั้น ผู้วิจัยจึงมีความสนใจที่จะสร้างเทคนิคการเรียนรู้ของเครื่อง เพื่อช่วยในการตัดสินใจเลือกพนักงานที่จะปรับตำแหน่ง

ความมุ่งหมายของงานวิจัย

ในการวิจัยครั้งนี้ผู้วิจัยได้ตั้งความมุ่งหมายไว้ ดังนี้

1. เพื่อสร้างแบบจำลองการทำนายการปรับตำแหน่งพนักงาน โดยใช้เทคนิคการเรียนรู้ของเครื่อง

2. เพื่อหาฟีเจอร์ที่มีส่งผลต่อการทำนายการปรับตำแหน่ง โดยใช้เทคนิคการคัดเลือกฟีเจอร์ (Feature Selection)

3. เพื่อจัดการความไม่สมดุลของข้อมูลด้วยวิธีการสุ่มตัวอย่างเพิ่มกลุ่มน้อยด้วยการสร้างตัวอย่างสังเคราะห์ (Synthetic Minority Oversampling Technique: SMOTE)

4. เพื่อหาแบบจำลองการทำการปรับตำแหน่งที่มีประสิทธิภาพที่สุดจากการเปรียบเทียบประสิทธิภาพของแบบจำลองแต่ละแบบ

ความสำคัญของการวิจัย

งานวิจัยนี้ศึกษาการทำนายผลการปรับตำแหน่งงานล่วงหน้า โดยใช้เทคนิคการเรียนรู้ของเครื่องเป็นเครื่องมือสำหรับสร้างแบบจำลองในงานวิจัย โดยคาดหวังว่าจะใช้ในการประกอบการตัดสินใจเลือกพนักงานที่จะปรับตำแหน่งได้

ขอบเขตของงานวิจัย

ประชากรที่ใช้ในการวิจัย

ใช้ข้อมูลการปรับตำแหน่งพนักงานปี 2562 - 2565 ของฝ่ายบริหารทรัพยากรบุคคล บริษัท ไทยสมุทรประกันชีวิต จำกัด (มหาชน) ที่มีการระบุว่าผ่านการปรับตำแหน่ง หรือไม่ผ่านการปรับตำแหน่ง

กลุ่มตัวอย่างที่ใช้ในการวิจัย

ใช้ข้อมูลการปรับตำแหน่งพนักงานปี 2562 - 2565 จำนวน 212 คน ที่มีการระบุว่าผ่านการปรับตำแหน่ง หรือไม่ผ่านการปรับตำแหน่ง และมีข้อมูลคุณลักษณะทั้งหมด 14 คอลัมน์

ตัวแปรที่ศึกษา

1. ตัวแปรอิสระ แบ่งเป็นดังนี้

1.1 ข้อมูลพนักงาน

- รหัสพนักงาน
- ตำแหน่งงานปัจจุบัน
- ตำแหน่งงานที่ปรับขึ้น
- สังกัดสายงาน
- การศึกษา
- เพศ
- อายุพนักงาน
- อายุการทำงานในองค์กร

1.2 ข้อมูลคะแนนการประเมินผลงานและพฤติกรรม

- ผลการปฏิบัติงาน
- ผลการประเมินสมรรถนะและทักษะ
- ผลการประเมินรอบทิศแบบ 360 องศา
- ผลสัมภาษณ์จากคณะกรรมการ

2. ตัวแปรตาม ได้แก่ ผลการปรับตำแหน่ง ซึ่งในงานวิจัยนี้มี 2 ประเภท คือ ผ่านการปรับตำแหน่ง หรือไม่ผ่านการปรับตำแหน่ง

กรอบแนวคิดในงานวิจัย

งานวิจัยนี้พัฒนาแบบจำลองการทำนายการปรับตำแหน่งของพนักงานระหว่างผ่านการปรับตำแหน่ง และไม่ผ่านการปรับตำแหน่ง โดยใช้เทคนิคการเรียนรู้ของเครื่องเป็นเครื่องมือสำหรับการสร้างแบบจำลอง โดยใช้ข้อมูลการปรับตำแหน่งพนักงานปี 2562 - 2565 จำนวน 212 คน ของฝ่ายบริหารทรัพยากรบุคคล บริษัท ไทยสมุทรประกันชีวิต จำกัด (มหาชน) ที่มีการระบุว่าผ่านการปรับตำแหน่ง หรือไม่ผ่านการปรับตำแหน่ง โดยมีข้อมูลคุณลักษณะทั้งหมด 14 คอลัมน์

โดยข้อมูลที่ได้รับมาจะอยู่ในรูปแบบ csv file หลังจากนั้นจึงนำไปเก็บในรูปแบบตาราง (DataFrame) และจะใช้เทคนิคการเรียนรู้ของเครื่องในการสร้างแบบจำลองเพื่อทำนายการปรับตำแหน่งพนักงาน ซึ่งเขียนด้วยภาษาโปรแกรม Python โดยใช้เทคนิคคัดเลือกฟีเจอร์ (Feature Selection) และเทคนิคการจัดการความไม่สมดุลของข้อมูลด้วยวิธีการสุ่มตัวอย่างเพิ่มกลุ่มน้อยด้วยการสร้างตัวอย่างสังเคราะห์ (Synthetic Minority Oversampling Technique: SMOTE) โดยใช้เทคนิคการเรียนรู้ของเครื่อง ได้แก่ Support Vector Machine, Decision Tree, Random

Forest และ XGBoost ในการแก้ปัญหาประเภท Classification การประเมินประสิทธิภาพของแบบจำลองจะใช้ค่า Precision, Accuracy, Recall, และ F1-score เพื่อเปรียบเทียบประสิทธิภาพของแบบจำลอง เพื่อหาแบบจำลองที่มีประสิทธิภาพในการทำนายสูงที่สุด

สมมุติฐานในการวิจัย

แบบจำลองที่ใช้เทคนิคการคัดเลือกฟีเจอร์ (Feature Selection) ร่วมกับเทคนิคการจัดการความไม่สมดุลของข้อมูล (SMOTE) จะเป็นแบบจำลองมีประสิทธิภาพมากที่สุดสำหรับการทำนายผลการปรับตำแหน่ง

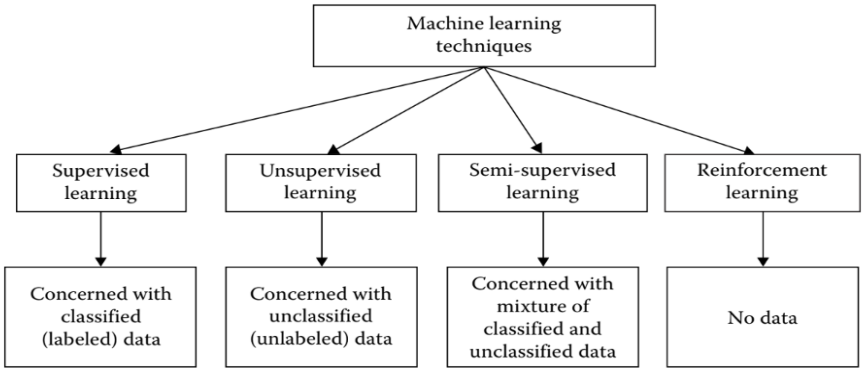
บทที่ 2
ทบทวนวรรณกรรม

ในการวิจัยครั้งนี้ ผู้วิจัยได้ศึกษาเอกสารและงานวิจัยที่เกี่ยวข้อง และได้นำเสนอตามหัวข้อต่อไปนี้

1. เทคนิคการเรียนรู้ของเครื่อง (Machine Learning Techniques)
2. แบบจำลองอัลกอริธึมที่ใช้ในเทคนิคการเรียนรู้ของเครื่องที่ใช้ในงานวิจัย
3. การปรับช่วงขอบเขตฟีเจอร์ (Feature Scaling)
4. การคัดเลือกฟีเจอร์ (Feature Selection)
5. การจัดการความไม่สมดุลของข้อมูล (Imbalanced Data)
6. การปรับค่าพารามิเตอร์ (Hyperparameter Tuning)
7. งานวิจัยที่เกี่ยวข้อง

เทคนิคการเรียนรู้ของเครื่อง (Machine Learning Techniques)

การเรียนรู้ของเครื่อง (Machine Learning) คือ การศึกษาทางวิทยาศาสตร์ที่เกี่ยวข้องกับอัลกอริธึมและแบบจำลองทางสถิติที่ใช้ระบบคอมพิวเตอร์ในการดำเนินการ ข้อได้เปรียบหลักของการใช้การเรียนรู้ของเครื่อง คือ เมื่ออัลกอริธึมเรียนรู้ว่าต้องทำอย่างไรกับข้อมูลจะสามารถทำงานกับข้อมูลได้โดยอัตโนมัติ แต่ไม่มีอัลกอริธึมแบบใดแบบหนึ่งที่ดีที่สุด เนื่องจากแต่ละอัลกอริธึมจะใช้ในการทำนายหรือช่วยในการตัดสินโดยอาศัยการใช้ข้อมูลที่แตกต่างกัน (Mahesh, 2020) ดังภาพประกอบ 1



ภาพประกอบ 1 แผนผังความแตกต่างของเทคนิคการเรียนรู้ของเครื่อง
ที่มา: (Mohammed et al., 2016)

1. การเรียนรู้แบบมีผู้สอน (Supervised learning) เทคนิคการเรียนรู้ของเครื่องประเภทนี้มีหลักการ คือ การทำนายผลลัพธ์ (Target) โดยเรียนรู้จากข้อมูลที่มีการระบุประเภทข้อมูล (Labeled data) เรียบร้อยแล้ว แบ่งออกเป็น 2 ประเภท คือ

1.1 การจำแนกประเภท (Classification) ใช้เพื่อแก้ปัญหาที่ต้องการผลลัพธ์ออกมาเป็นกลุ่มหรือประเภทข้อมูล อาทิเช่น การทำนายผลการเรียนของนักศึกษา, การจำแนกเนื้อร้าย เป็นต้น

1.2 การวิเคราะห์การถดถอย (Regression) ใช้เพื่อแก้ปัญหาที่ต้องการผลลัพธ์ออกมาเป็นค่าตัวเลขที่ต่อเนื่องกัน อาทิเช่น การทำนายราคาที่ดิน, การทำนายปริมาณออกซิเจนในอากาศ เป็นต้น

2. การเรียนรู้แบบไม่มีผู้สอน (Unsupervised learning) เทคนิคการเรียนรู้ของเครื่องประเภทนี้ มีหลักการ คือ การทำนายผลลัพธ์ (Target) โดยเรียนรู้จากข้อมูลที่ไม่มีการระบุประเภทข้อมูล (Unlabeled data) ของข้อมูลไว้ แบ่งออกได้เป็น 2 ประเภท ดังนี้

2.1 การแบ่งกลุ่มข้อมูล (Clustering) เป็นการจัดกลุ่มข้อมูลที่มีความคล้ายกันให้อยู่ในกลุ่มเดียวกัน อาทิเช่น การจัดกลุ่มของคลิปวิดีโอ

2.2 การหาความสัมพันธ์ของข้อมูล (Association) เป็นการค้นหาความสัมพันธ์ของตัวแปรในชุดข้อมูล (Dataset) อาทิเช่น การหาสินค้าที่ลูกค้ามักจะซื้อพร้อมกัน

3. การเรียนรู้แบบกึ่งมีผู้สอน (Semi-supervised learning) เทคนิคการเรียนรู้ของเครื่องประเภทนี้ มีหลักการ คือ เรียนรู้โดยใช้ข้อมูลทั้งที่มีการระบุประเภทข้อมูล (labeled data) และไม่มีการระบุประเภทข้อมูล (Unlabeled data) ใช้เพื่อสร้างรูปแบบที่เหมาะสมในการจำแนกข้อมูล ซึ่งส่วนใหญ่ข้อมูลที่ใช้เป็นข้อมูลที่ไม่มีการระบุประเภทข้อมูล

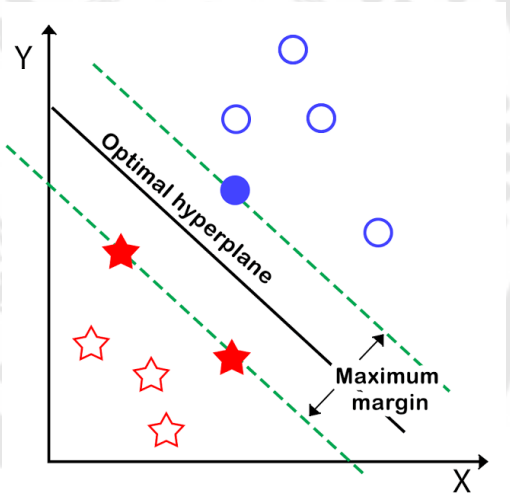
4. การเรียนรู้แบบเสริมแรง (Reinforcement Learning) เทคนิคการเรียนรู้ของเครื่องประเภทนี้ มีหลักการ คือ เรียนรู้โดยไม่ใช้ข้อมูล ซึ่งใช้ตัวแทน (Agent) กระทำการตัดสินใจเพื่อแก้ปัญหาสถานการณ์ต่างๆ โดยใช้ประสบการณ์หรือดูจากสภาพแวดล้อมนั้นๆ ซึ่งแต่ละการกระทำจะมีการให้คะแนน (Reward) เพื่อหาผลลัพธ์ที่ดีที่สุดในการแก้ปัญหา อาทิเช่น หุ่นยนต์แตะบอลเข้า Goal เป็นต้น

แบบจำลองอัลกอริธึมที่ใช้ในเทคนิคการเรียนรู้ของเครื่องที่ใช้ในงานวิจัย

แบบจำลองอัลกอริธึมที่ใช้เทคนิคการเรียนรู้ของที่ใช้ในงานวิจัยนี้ เป็นเทคนิคการเรียนรู้ของเครื่องแบบมีผู้สอน (Supervised learning) ซึ่งเป็นการจำแนกประเภท (Classification) 4 แบบจำลอง ได้แก่ Support Vector Machine, Decision Tree, Random Forest และ XGBoost

1. Support Vector Machine (SVM) เป็นแบบจำลองที่มีหลักการ คือ สร้างเส้น (Hyperplane) เพื่อแบ่งข้อมูลที่กระจายอยู่บน feature space ออกจากกัน โดยดูจากระยะขอบ (margin) ยิ่งเส้นแบ่งมีระยะขอบมาก (Maximum margin) ข้อผิดพลาดในการจัดประเภทข้อมูลก็ยิ่งจะน้อยลง (Singh et al., 2016) ดังภาพประกอบ 2 โดยใช้สมการที่กำหนดตัวแปรต่างๆ ได้แก่ w^T คือ ค่าน้ำหนัก, x_i คือ คุณลักษณะของรายการที่ i, b คือ ค่าไบแอส (Bias) และ $i = 1, 2, 3 \ldots, N$ เป็นของรายการที่ i ของชุดข้อมูล ดังสมการที่ 1

$$w^T x_i + b = 0 \qquad [1]$$

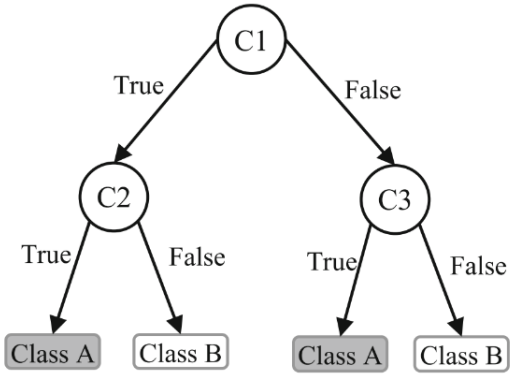


ภาพประกอบ 2 การแบ่งกลุ่มข้อมูล 2 ประเภท ด้วยเส้นแบ่ง maximum margin

ที่มา: https://medium.com/@cdabakoglu/what-is-support-vector-machine-svm-fd0e9e39514f

2. Decision Tree เป็นแบบจำลองที่เรียกว่าต้นไม้การตัดสินใจมีหลักการ คือ สร้างกฎการตัดสินใจ โดยจำแนกข้อมูลเป็นโครงสร้างต้นไม้ แบบลำดับชั้นการตัดสินใจ โดยมีโหนดแรกหรือโหนดบนสุดเรียกว่า Root node เป็นตัวแปรต้นหรือเงื่อนไขแรก และแตกแขนง (branching) ทำซ้ำ

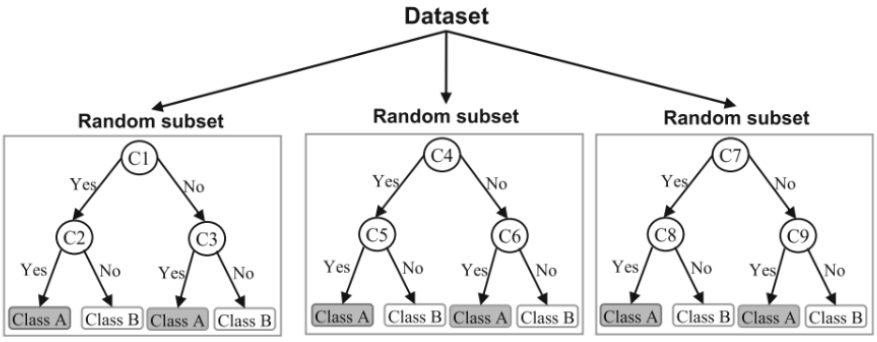
การตัดสินใจไปยังเงื่อนไขลำดับถัดไปจนถึงโหนดใบ เรียกว่า Leaf node ซึ่งเป็นผลลัพธ์สุดท้ายที่ทำนายออกมาได้ ดังภาพประกอบ 3



ภาพประกอบ 3 ขั้นตอนของ Decision Tree

ที่มา: (Uddin et al., 2019)

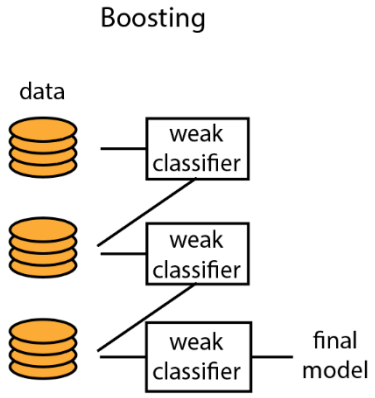
3. Random Forest เป็นแบบจำลองที่มีหลักการ คือ สร้าง Decision Tree ขึ้นมาหลายๆ ต้น โดยชุดข้อมูล (dataset) จะผ่านการทำ bootstrapping หรือการสุ่มข้อมูลแบบเลือกซ้ำได้ และฟีเจอร์ (feature) ที่นำมาใช้ในการแบ่งจะเกิดการทำ random feature projecting หรือการสุ่มชุดของฟีเจอร์ คือ ในการแบ่งแต่ละครั้งจะมีแค่ subset ของฟีเจอร์ เท่านั้น ที่มาใช้ในการทำนาย จึงทำให้ได้ต้นไม้ที่ไม่เหมือนกัน โดยผลลัพธ์ของการทำนายจะได้จากผลโหวตที่มากที่สุด (Majority Vote) จากต้นไม้หลายๆ ต้น ดังภาพประกอบ 4



ภาพประกอบ 4 ขั้นตอนของ Random Forest

ที่มา: (Uddin et al., 2019)

4. Extreme Gradient Boosting (XGBoost) เป็นแบบจำลองที่มีหลักการที่พัฒนามาจากเทคนิค Gradient boosting โดยใช้หลักการของ Ensemble Learning Method ในการ Boosting ดังภาพประกอบ 5 เพื่อสร้างการเรียนรู้หลายๆ ตัว (Multi Learner) เข้าด้วยกัน คือ การนำ Decision Tree หลายๆ ต้น มาเรียนรู้ต่อๆ กัน โดยที่แต่ละ Decision Tree จะเรียนรู้จากข้อผิดพลาดของ Decision Tree ก่อนหน้า ทำให้ความแม่นยำในการทำนายมีมากขึ้นเรื่อย ๆ ซึ่งแบบจำลองจะหยุดเรียนรู้เมื่อ Decision Tree มีความต่อเนื่องกันจนมีความลึกมากพอและไม่เหลือรูปแบบ (pattern) ของข้อผิดพลาดจาก Decision Tree ก่อนหน้าให้เรียนรู้แล้ว ดังภาพประกอบ 5



ภาพประกอบ 5 การทำงานของการ Boosting

ที่มา: (TITIPATA, 2018)

การปรับช่วงขอบเขตฟีเจอร์ (Feature Scaling)

การปรับช่วงขอบเขตฟีเจอร์เป็นหนึ่งในขั้นตอนของการเตรียมข้อมูลให้เหมาะสม เพื่อนำไปสร้างแบบจำลองหรือประมวลผล โดยจะแปลงฟีเจอร์ที่เป็นตัวเลขที่มีขอบเขตต่างกัน ให้เป็นตัวเลขภายในขอบเขตเดียวกัน ข้อดีของการปรับช่วงขอบเขตฟีเจอร์จะทำให้ลดการเกิดอคติ (bias) ที่อาจจะทำให้แบบจำลองที่สร้างขึ้นเกิดการทำนายผิดพลาดได้และช่วยลดเวลาในการสร้างแบบจำลอง เนื่องจากชุดข้อมูลจะมีขนาดเล็กลง ทำให้คำนวณได้เร็วขึ้น ซึ่งการปรับช่วงขอบเขตฟีเจอร์มีเทคนิค (Zheng & Casari, 2018) ได้แก่

1. Min-Max Scaling เป็นเทคนิคที่ปรับช่วงขอบเขตฟีเจอร์ให้อยู่ในช่วง [0,1] เหมาะกับฟีเจอร์ที่ไม่มีค่าผิดปกติ (outlier) หรือมีค่าผิดปกติ (outlier) น้อย โดยใช้สมการที่กำหนดตัวแปรต่างๆ ได้แก่ $\min(x)$ คือ ค่าที่น้อยที่สุดในฟีเจอร์ และ $\max(x)$ คือ ค่าที่มากที่สุดในฟีเจอร์ ดังสมการที่ 2

$$\tilde{x} = \frac{x - \min(x)}{\max(x) - \min(x)} \quad [2]$$

2. Standardization หรือ Variance Scaling เป็นเทคนิคที่ปรับช่วงขอบเขตฟีเจอร์ให้อยู่ในช่วงค่าเฉลี่ย (mean) เท่ากับ 0 และ ส่วนเบี่ยงเบนมาตรฐาน (standard deviation) เท่ากับ 1 ซึ่งจะช่วยให้ข้อมูลมีความเปรียบเทียบได้ง่ายขึ้น เหมาะกับฟีเจอร์ที่มีการกระจายตัวปกติ (Normal distribution) โดยใช้สมการที่กำหนดตัวแปรต่างๆ ได้แก่ $\bar{x}$ คือ ค่าเฉลี่ยของฟีเจอร์ และ σ คือ ส่วนเบี่ยงเบนมาตรฐานของฟีเจอร์ ดังสมการที่ 3

$$\tilde{x} = \frac{x - \bar{x}}{\sigma} \quad [3]$$

การคัดเลือกฟีเจอร์ (Feature Selection)

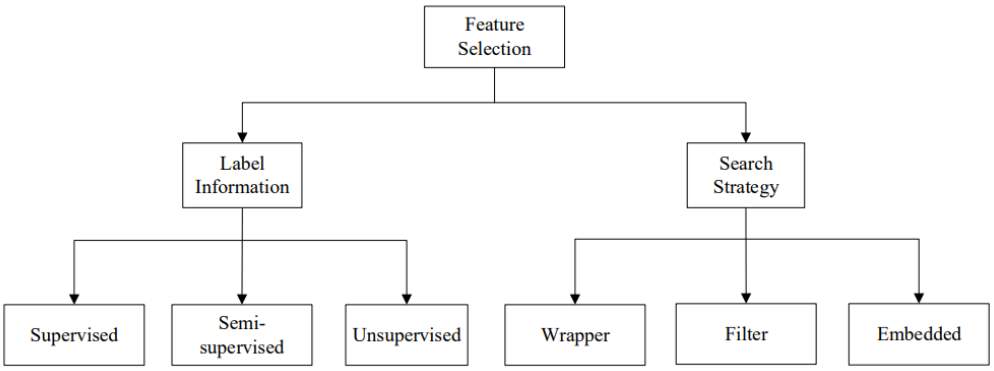
การคัดเลือกฟีเจอร์เป็นเทคนิคการลดมิติ (dimensionality reduction technique) มีจุดมุ่งหมายเพื่อกำจัดฟีเจอร์ที่ไม่เกี่ยวข้อง ซ้ำซ้อน ที่ไม่มีส่วนช่วยในการสร้างแบบจำลองออก โดยจะคัดเลือกฟีเจอร์ที่มีความสัมพันธ์ (Correlation) สูงกับ label และตัดฟีเจอร์ที่มีความสัมพันธ์น้อยๆ ออก เพื่อทำให้แบบจำลองมีประสิทธิภาพมากขึ้น ให้ความแม่นยำที่สูงขึ้น ใช้เวลาในการคำนวณลดลง ป้องกันการ Overfitting และช่วยให้แบบจำลองอธิบายได้ดีขึ้น สามารถแบ่งได้ 3 วิธีหลักๆ ดังภาพประกอบ 6 ได้แก่

1. Filter Methods เป็นการคัดเลือกฟีเจอร์โดยดูความเกี่ยวข้องระหว่างฟีเจอร์และผลลัพธ์ โดยฟีเจอร์ที่มีความเกี่ยวข้องน้อยจะถูกคัดออกไม่นำมาใช้สร้างแบบจำลอง ข้อดี คือ เป็นวิธีที่มีความรวดเร็ว เหมาะกับข้อมูลที่มีฟีเจอร์จำนวนมาก และทำการคำนวณหาความเกี่ยวข้องของฟีเจอร์เพียงครั้งเดียว จึงทำให้ไม่ต้องใช้ทรัพยากรในการคำนวณ ข้อเสีย คือ ฟีเจอร์แต่ละตัวจะถูกวัดคะแนนความเกี่ยวข้องแยกกันกับสร้างแบบจำลอง ดังนั้นฟีเจอร์ที่ได้อาจไม่เหมาะที่จะนำไปใช้ในการสร้างแบบจำลองได้ เช่น Correlation, Chi-square, Mutual information เป็นต้น

2. Wrapper Methods เป็นการคัดเลือกฟีเจอร์โดยคำนวนณจากการเรียนรู้และทดสอบจากการสร้างแบบจำลองที่เลือกไว้ ข้อดี คือ เป็นวิธีที่หาฟีเจอร์ที่เหมาะสมกับแบบจำลองโดยตรง

ข้อเสีย คือ มีความเสี่ยงที่จะ Overfitting มากกว่า Filter Methods เพราะมีการคำนวณหลายครั้ง และใช้เวลาในการคำนวณมากกว่า Filter Methods เช่น Forward Selection, Recursive Feature Elimination (RFE)

3. Embedded Methods เป็นการคัดเลือกฟีเจอร์ที่เหมาะสมที่สามารถสร้างแบบจำลองให้ได้ประสิทธิภาพมากที่สุดขณะที่กำลังสร้างแบบจำลอง เช่น Decision Tree, L1 Regularization ข้อดี คือ ได้ฟีเจอร์ที่มีความเกี่ยวข้องกับการสร้างแบบจำลองสูง และใช้เวลาคำนวณน้อยกว่าวิธี Wrapper Methods



ภาพประกอบ 6 วิธีการ Feature Selection

ที่มา: (Miao & Niu, 2016)

การจัดการความไม่สมดุลของข้อมูล (Imbalanced data)

ปัญหาชุดข้อมูลที่ไม่สมดุลเกิด คือ จำนวน Class หนึ่งมีจำนวนมากกว่าอีก Class หนึ่ง เมื่อเกิดความไม่สมดุลของชุดข้อมูลจะทำแบบจำลองเรียนรู้ข้อมูลที่มีจำนวน Class มากกว่า ทำให้เกิดการอคติ (bias) และผลลัพธ์ของการทำนายจะออกมาเป็น Class ที่มีมากกว่า ซึ่งจะทำให้ประสิทธิภาพของแบบจำลองลดน้อยลง โดยเทคนิคการจัดการกับข้อมูลที่ไม่สมดุลมีทั้งการทำ Oversampling คือ เทคนิคการสุ่มเพิ่มจำนวนตัวอย่างข้อมูลของ Class ที่มีจำนวนน้อย โดยจะเพิ่มข้อมูลให้เท่ากับ Class ที่มีจำนวนมากกว่า เพื่อให้มีจำนวนเท่ากัน หรือ เทคนิคการทำ Undersampling คือ การการสุ่มลดจำนวนตัวอย่างข้อมูลของ Class ที่มีจำนวนมากกว่า โดยจะลดข้อมูลให้เท่ากับ Class ที่มีจำนวนน้อย เพื่อให้มีจำนวนเท่ากัน

งานวิจัยนี้จะใช้เทคนิค Synthetic Minority Over-sampling Technique (SMOTE) เป็นเทคนิคการสุ่มตัวอย่างแบบ Oversampling แบบหนึ่ง โดยทำการสุ่มตัวอย่างแบบสังเคราะห์ข้อมูลขึ้นมาใหม่จากข้อมูลเดิมที่มีอยู่ของ Class ที่มีจำนวนน้อย ใช้หลักการ K-neighbors หรือการหา

เพื่อนบ้านที่มีระยะใกล้ที่สุด โดยข้อมูลสังเคราะห์สร้างขึ้นระหว่างข้อมูลสุ่มและเพื่อนบ้านที่มีระยะใกล้ที่สุดจากจำนวน k ที่ได้เลือกไว้ ซึ่งขั้นตอนนี้จะมีการทำซ้ำหลายครั้งจนกว่าจะได้จำนวนข้อมูลใน Class ที่มีน้อยกว่าเท่ากับจำนวนข้อมูลใน Class ที่มีมากกว่า (Ramyachitra & Manikandan, 2014)

การปรับค่าพารามิเตอร์ (Hyperparameter Tuning)

เป็นกระบวนการในการสร้างและปรับปรุงแบบจำลอง เพื่อหาค่าพารามิเตอร์ (parameter) ที่ดีที่สุดให้กับแบบจำลองนั้นๆ ทำให้เพิ่มประสิทธิภาพของแบบจำลอง (Owen, 2022) โดยมีเทคนิคหลักๆ 3 เทคนิค ได้แก่

1. Manual search เป็นวิธีการปรับค่า Hyperparameter ที่ใช้ความชำนาญและประสบการณ์ของผู้ที่สร้างแบบจำลอง โดยทำทดสอบแบบจำลองด้วยค่า Hyperparameter ที่กำหนด และปรับค่า Hyperparameter ไปเรื่อย ๆ จนกว่าจะได้ค่าที่พึงพอใจ

2. Grid Search เป็นวิธีการปรับค่า Hyperparameter จากการกำหนดค่า Hyperparameter ไว้แล้ว โดยจะทำการค้นหาค่า Hyperparameter จากค่าที่กำหนดไว้ทุกค่ากับแบบจำลอง และเลือกค่าที่ทำให้แบบจำลองมีประสิทธิภาพสูงที่สุด ซึ่งจะใช้เวลานานกว่าแบบ Random Search

3. Random Search เป็นวิธีการปรับค่า Hyperparameter แบบการสุ่มค่า Hyperparameter จากค่าที่กำหนดไว้ ทดสอบกับแบบจำลองถ้าพบว่าค่า Hyperparameter ที่สุ่มขึ้นทำให้แบบจำลองมีประสิทธิภาพดีกว่าค่า Hyperparameter ปัจจุบัน จะใช้ค่า Hyperparameter นั้น ข้อดี คือ สามารถค้นหาค่า Hyperparameter ที่ดีได้อย่างรวดเร็ว ข้อเสีย คือ อาจไม่ได้ค่า Hyperparameter ที่ดีที่สุด เนื่องจากเป็นการสุ่มค่า Hyperparameter

งานวิจัยที่เกี่ยวข้อง

การทบทวนวรรณกรรมของงานวิจัยนี้ได้ทำการศึกษาค้นคว้างานวิจัยที่เกี่ยวข้องกับการทำนายปรับตำแหน่งพนักงานโดยงานวิจัยที่เกี่ยวข้องมีรายละเอียด ดังนี้

1. บทความเรื่อง Employee Classification for Personalized Professional Training Using Machine Learning Techniques and SMOTE (Kaewwiset et al., 2021)

งานวิจัยนี้ทำการศึกษาวิธีการทำนายการปรับตำแหน่งพนักงาน โดยใช้เทคนิคการเรียนรู้ของเครื่อง 3 แบบ คือ Decision tree, Random Forest และ Support Vector Machines และ

แบ่งชุดข้อมูล training set และ test set ที่แตกต่างกัน 3 กลุ่ม คือ 80:20, 70:30 และ 60:40 โดยในแต่ละกลุ่มจะทำ SMOTE กับชุดข้อมูล และไม่ทำ SMOTE

จากผลการศึกษาพบว่าการทำ SMOTE กับชุดข้อมูลให้ความแม่นยำมากกว่าการไม่ทำ SMOTE และแบ่งชุดข้อมูล training set และ test set ที่ 80:20 ให้ความแม่นยำสูงสุด โดยแบบจำลอง Random Forest เป็นแบบจำลองที่มีความแม่นยำสูงสุดอยู่ที่ 96.32% ซึ่งการทำ SMOTE กับชุดข้อมูล สามารถปรับปรุงความแม่นยำได้สูงสุดถึง 4.80% เมื่อเทียบกับการไม่ทำ SMOTE

2. บทความเรื่อง Employee Promotion Prediction by using Machine Learning Algorithms for Imbalanced Dataset (Şahinbaş, 2022)

งานวิจัยนี้ทำการศึกษาวิธีการทำนายปรับตำแหน่งพนักงานโดยใช้เกณฑ์การประเมินพนักงานในการปรับตำแหน่ง โดยใช้เทคนิคการเรียนรู้ของเครื่อง 3 แบบ คือ Support Vector Machines, Artificial Neural Network และ Random Forest โดยมีการจัดการกับชุดข้อมูลที่ไม่สมดุล 2 แบบ คือ การทำ SMOTE และการทำ Random Over-Sampling

จากผลการศึกษาพบว่าแบบจำลอง Random Forest เป็นแบบจำลองที่มีประสิทธิภาพสูงสุดด้วย accuracy อยู่ที่ 98%, precision อยู่ที่ 96%, recall อยู่ที่ 100% และ f1-score อยู่ที่ 98% พร้อมกับการทำ Random Over-Sampling

3. บทความเรื่อง Prediction of Employee Promotion Based on Personal Basic Features and Post Features (Long et al., 2018)

งานวิจัยนี้ทำการศึกษาวิธีการทำนายปรับตำแหน่งพนักงาน โดยใช้ข้อมูลของพนักงานองค์กรรัฐวิสาหกิจในประเทศจีน ใช้เทคนิคการเรียนรู้ของเครื่อง 6 แบบ คือ K-Nearest Neighbor (KNN), Logistic Regression (LR), Support Vector Classifier (SVC), Decision Tree (DT), Random Forest (RF) และ Adaboost โดยจะมีการดำเนินการ 3 หลัก คือ

ขั้นตอนที่ 1 สร้างฟีเจอร์ที่เป็นข้อมูลพื้นฐานส่วนบุคคล และข้อมูลตามกลยุทธ์ 5 ประการ ได้แก่ กลยุทธ์ 1 คือ เลือกค่าที่ไม่ซ้ำ, กลยุทธ์ 2 คือ เลือกโหมด, กลยุทธ์ 3 คือ เลือกค่าสูงสุดหรือต่ำสุด, กลยุทธ์ 4 คือ นับจำนวนค่าต่างๆ และกลยุทธ์ 5 คือ คำนวณความแตกต่างระหว่างสองวัน

ขั้นตอนที่ 2 ตรวจสอบฟีเจอร์ และวิเคราะห์ความสัมพันธ์ (correlation) ระหว่างฟีเจอร์

ขั้นตอนที่ 3 สร้างแบบจำลอง ทดสอบแบบจำลอง และวิเคราะห์ความสำคัญของฟีเจอร์

จากผลการศึกษาพบว่าแบบจำลอง Random Forest เป็นแบบจำลองที่มีความแม่นยำสูงสุดอยู่ที่ 90% จาก 6 แบบจำลอง เมื่อการตรวจสอบความถูกต้องของฟีเจอร์ต่างๆ คำนวณค่า

Gini ของแต่ละฟีเจอร์เพื่อวิเคราะห์เพิ่มเติม พบว่าฟีเจอร์ข้อมูลตามกลยุทธ์ 5 ประการมีผลกระทบกับการปรับตำแหน่งมากกว่าเมื่อเทียบกับฟีเจอร์ข้อมูลพื้นฐานส่วนบุคคล และจากฟีเจอร์ทั้งหมด จำนวนปีการทำงาน ตำแหน่งที่แตกต่างกัน และระดับแผนกสูงสุดส่งผลกระทบอย่างมากกับการปรับตำแหน่ง

4. บทความเรื่อง Promotion Classification Using Decision Tree and Principal Component Analysis (Kaewwiset & Temdee, 2022)

งานวิจัยนี้ทำการศึกษาวิธีการจำแนกการปรับตำแหน่งพนักงาน โดยใช้เทคนิคการเรียนรู้ของเครื่อง 1 แบบ คือ Decision Tree และทำ Principle Component Analysis (PCA) โดยใช้ชุดข้อมูลเปิด Human Resource Analysis Case Study dataset จาก Kaggle เพื่อเปรียบเทียบประสิทธิภาพของแบบจำลอง เมื่อมีการใช้ฟีเจอร์ที่แตกต่างกัน 3 แบบ คือ ใช้ฟีเจอร์ทั้งหมด, ใช้เฉพาะฟีเจอร์ข้อมูลส่วนบุคคล และใช้เฉพาะฟีเจอร์ประสิทธิภาพการทำงาน โดยทั้ง 3 แบบ มีการทำ PCA และไม่ทำ PCA

จากผลการศึกษาพบว่าการใช้เฉพาะฟีเจอร์ประสิทธิภาพการทำงานมีผลกระทบกับการปรับตำแหน่งมากกว่าเมื่อเทียบกับการใช้เฉพาะฟีเจอร์ข้อมูลพื้นฐานส่วนบุคคล และการใช้ฟีเจอร์ทั้งหมด โดยแบบจำลอง Decision Tree เมื่อใช้เฉพาะฟีเจอร์ประสิทธิภาพการทำงานพร้อมกับทำ PCA ให้ความแม่นยำสูงสุดอยู่ที่ 91.25% และแบบจำลอง Decision Tree เมื่อใช้เฉพาะฟีเจอร์ประสิทธิภาพการทำงานแต่ไม่มีการทำ PCA ให้ความแม่นยำสูงสุดที่ 91.22%

5. บทความเรื่อง Analysis and Prediction of Employee Promotions Using Machine Learning (Alqahtani & Almaleh, 2022)

งานวิจัยนี้ทำการศึกษาวิธีการทำนายการปรับตำแหน่งพนักงาน และหาฟีเจอร์ที่สำคัญที่ส่งผลต่อการปรับตำแหน่ง โดยใช้ชุดข้อมูลเปิดจาก Kaggle ปี 2020 ที่ประกอบข้อมูลบริษัทข้ามชาติ 54,808 แถว และ 13 คอลัมน์ ซึ่งเป็นชุดข้อมูลที่ครอบคลุมธุรกิจ 9 ประเภท ใช้เทคนิคการเรียนรู้ของเครื่อง 6 แบบ คือ K-Nearest Neighbor, Logistic Regression, Decision Tree, Random Forest, Support Vector Machine และ Ensemble models (Adaboosting and Gradient Boosting models)

จากผลการศึกษาพบว่าแบบจำลอง Gradient Boosting มีประสิทธิภาพดีกว่าแบบจำลองอื่น ให้ความแม่นยำสูงสุดอยู่ที่ 93.90% และฟีเจอร์ที่สำคัญที่สุดในการทำนายการปรับตำแหน่ง คือ Previous Year Ratings Feature

6. บทความเรื่อง KNN and SVM Machine learning to Predict Staff Due for Promotions and Training (Emmanuel-Okereke & Anigbogu, 2022)

งานวิจัยนี้ทำการศึกษาวิธีการทำนายการปรับตำแหน่งพนักงานและการฝึกอบรมพนักงาน โดยใช้เทคนิคการเรียนรู้ของเครื่อง 2 แบบ คือ K-Nearest Neighbor และ Support Vector Machine ซึ่งแบบจำลองทั้ง 2 แบบ มีการใช้ grid search cross validation เพื่อเลือก kernel ที่เหมาะสมที่สุดร่วมกับค่า support vector

จากผลการศึกษาพบว่าแบบจำลอง Support Vector Machine มีความแม่นยำสูงสุดอยู่ที่ 91% เมื่อมีการปรับค่า hyper-parameter อย่างละเอียดหลายค่า ซึ่งสูงกว่าแบบจำลอง K-Nearest Neighbor ที่มีความแม่นยำอยู่ที่ 79%

จากการศึกษางานวิจัยที่เกี่ยวข้องกับการทำนายปรับตำแหน่งพนักงานทั้งหมดจำนวน 6 เรื่อง ผู้วิจัยจึงทำการสรุปวิธีการ และผลลัพธ์ของแต่ละงานวิจัย ดังตาราง 1

ตาราง 1 แสดงผลการทดลองของงานวิจัยที่เกี่ยวข้อง

ลำดับ	งานวิจัย	แบบจำลอง	วิธีการ	ผลการทดสอบ
1.	Employee Classification for Personalized Professional Training Using Machine Learning Techniques and SMOTE (Kaewwiset et al., 2021)	3 แบบจำลอง คือ Decision tree, Random Forest และ Support Vector Machines	แบ่ง training set และ test set ที่แตกต่างกัน 3 กลุ่ม คือ 80:20, 70:30 และ 60:40 แต่ละกลุ่มทำ SMOTE กับชุดข้อมูล และไม่ทำ SMOTE	Random Forest แบ่ง training set และ test set ที่ 80:20 ร่วมกับการทำ SMOTE ให้ accuracy สูงสุดอยู่ที่ 96.32% และไม่ทำ SMOTE อยู่ที่ 93.52%
2.	Employee Promotion Prediction by using Machine Learning Algorithms for Imbalanced Dataset (Şahinbaş, 2022)	3 แบบจำลอง คือ Support Vector Machines, Artificial Neural Network และ Random Forest	จัดการกับชุดข้อมูลที่ไม่สมดุล 2 แบบ คือ การทำ SMOTE และการทำ Random Over-Sampling	Random Forest ร่วมกับการทำ Random Over-Sampling มีประสิทธิภาพสูงสุดด้วย accuracy ที่ 98%, precision ที่ 96%, recall ที่ 100% และ f1-score ที่ 98%

ตาราง 1 (ต่อ)

ลำดับ	งานวิจัย	แบบจำลอง	วิธีการ	ผลการทดสอบ
3.	Prediction of Employee Promotion Based on Personal Basic Features and Post Features (Long et al., 2018)	6 แบบ คือ K-Nearest Neighbor (KNN), Logistic Regression (LR), Support Vector Classifier (SVC), Decision Tree (DT), Random Forest (RF) และ Adaboost	สร้างฟีเจอร์ที่เป็นข้อมูลพื้นฐานส่วนบุคคล และข้อมูลตามกลยุทธ์ 5 ประการเพิ่มเติม และวิเคราะห์ความสำคัญของฟีเจอร์	Random Forest ให้ accuracy สูงสุดอยู่ที่ 90% ฟีเจอร์ข้อมูลตามกลยุทธ์ 5 ประการ เป็นกลุ่มฟีเจอร์ที่มีความสำคัญที่สุดจากฟีเจอร์ทั้งหมด จำนวนปีการทำงาน, ตำแหน่งที่แตกต่างกัน และระดับแผนกสูงสุด มีความสำคัญอย่างมากกับการปรับตำแหน่ง
4.	Promotion Classification Using Decision Tree and Principal Component Analysis (Kaewwiset & Temdee, 2022)	1 แบบจำลอง คือ Decision Tree	เปรียบเทียบประสิทธิภาพ เมื่อทำ PCA และไม่ทำ PCA เมื่อใช้ฟีเจอร์ที่แตกต่างกัน 3 แบบ คือ ฟีเจอร์ทั้งหมด, ฟีเจอร์ข้อมูลส่วนบุคคล และฟีเจอร์ประสิทธิภาพการทำงาน	Decision Tree เมื่อใช้เฉพาะฟีเจอร์ประสิทธิภาพการทำงานร่วมกับการทำ PCA ให้ accuracy สูงสุดอยู่ที่ 91.25% และ ไม่ทำ PCA ให้ accuracy สูงสุดอยู่ที่ 91.22%

ตาราง 1 (ต่อ)

ลำดับ	งานวิจัย	แบบจำลอง	วิธีการ	ผลการทดสอบ
5.	Analysis and Prediction of Employee Promotions Using Machine Learning (Alqahtani & Almaleh, 2022)	6 แบบจำลอง คือ K-Nearest Neighbor, Logistic Regression, Decision Tree, Random Forest, Support Vector Machine และ Ensemble models (Adaboosting and Gradient Boosting models)	หาแบบจำลองที่ดีที่สุด และหาฟีเจอร์ที่สำคัญที่ส่งผลต่อการปรับตำแหน่ง	Gradient Boosting ให้ accuracy สูงสุดอยู่ที่ 93.90% และฟีเจอร์ที่สำคัญที่สุดในการทำนาย คือ Previous Year Ratings Feature
6.	KNN and SVM Machine learning to Predict Staff Due for Promotions and Training (Emmanuel-Okereke & Anigbogu, 2022)	2 แบบจำลอง คือ K-Nearest Neighbor และ Support Vector Machine	ใช้ grid search cross validation เพื่อเลือก kernel ที่เหมาะสมที่สุดร่วมกับค่า support vector ฟีเจอร์ประสิทธิภาพการทำงาน มีการทำ PCA และไม่ทำ PCA	Support Vector Machine ให้ accuracy สูงสุดอยู่ที่ 91% เมื่อมีการปรับค่า hyper-parameter อย่างละเอียด

บทที่ 3
วิธีดำเนินการวิจัย

ในการวิจัยครั้งนี้ ผู้วิจัยได้ดำเนินการตามขั้นตอน ดังนี้

1. การกำหนดประชากรและกลุ่มตัวอย่าง
2. การออกแบบโครงสร้างและวิธีการดำเนินงานวิจัย
3. การเตรียมข้อมูลและการวิเคราะห์ข้อมูล
4. การแบ่งชุดข้อมูลและการปรับช่วงขอบเขตฟีเจอร์
5. การคัดเลือกฟีเจอร์และการจัดการความไม่สมดุลของข้อมูล
6. การสร้างแบบจำลอง และการประเมินประสิทธิภาพแบบจำลอง

การกำหนดประชากรและกลุ่มตัวอย่าง

1. ประชากร

งานวิจัยนี้ใช้ข้อมูลการปรับตำแหน่งพนักงานปี 2562 - 2565 จำนวน 212 คน ของฝ่ายบริหารทรัพยากรบุคคล บริษัท ไทยสมุทรประกันชีวิต จำกัด (มหาชน) ที่มีการระบุว่าผ่านการปรับตำแหน่ง หรือไม่ผ่านการปรับตำแหน่ง ซึ่งภายในชุดข้อมูลประกอบด้วยข้อมูลตัวแปรทั้งหมด 14 คอลัมน์ และมีจำนวนแถวทั้งหมด 212 แถว ดังตาราง 2

ตาราง 2 รายละเอียดของชุดข้อมูลการปรับตำแหน่ง

ลำดับ	**ข้อมูลตัวแปร** (Variable)	**คำอธิบายข้อมูล** (Description)
1	employee_id (รหัสพนักงาน)	รหัสพนักงาน
2	position (ตำแหน่งงานปัจจุบัน)	Officer คือ เจ้าหน้าที่ Senior Officer คือ เจ้าหน้าที่อาวุโส Senior Lecturer คือ วิทยากรอาวุโส Specialist คือ ผู้เชี่ยวชาญ Senior Specialist คือ ผู้เชี่ยวชาญอาวุโส Branch Area Manager คือ ผู้จัดการสาขา

ตาราง 2 (ต่อ)

ลำดับ	ข้อมูลตัวแปร (Variable)	คำอธิบายข้อมูล (Description)
		Manager คือ ผู้จัดการ Senior Manager คือ ผู้จัดการอาวุโส
3	position next (ตำแหน่งงานที่ปรับขึ้น)	Specialist Lecturer คือ วิทยากรผู้เชี่ยวชาญ Team Leader of Lecturer คือ หัวหน้าทีมวิทยากร Team Leader of Lecturer Region คือ หัวหน้าทีมวิทยากรภูมิภาค Specialist คือ ผู้เชี่ยวชาญ Senior Specialist คือ ผู้เชี่ยวชาญอาวุโส Elite Specialist คือ ผู้เชี่ยวชาญระดับสูง Manager คือ ผู้จัดการ Senior Manager คือ ผู้จัดการอาวุโส Associate Vice President คือ รองผู้อำนวยการฝ่าย
4	group (สังกัดสายงาน)	Agent Channels คือ สายงานประกันชีวิตด้านช่องทางตัวแทน Alternative Channels คือ สายงานประกันชีวิตด้านช่องทางการขายสถาบัน Assets & Administration Group คือ สายงานทรัพย์สินและธุรการ Customer Service Group คือ สายงานบริการลูกค้าธุรกิจประกันชีวิต Direct Report President คือ สายงานกลาง Financial Group คือ สายงานการเงิน Information Technology Group คือ สายงานเทคโนโลยีสารสนเทศ

ตาราง 2 (ต่อ)

ลำดับ	ข้อมูลตัวแปร (Variable)	คำอธิบายข้อมูล (Description)
5	education (การศึกษาสูงสุด)	Senior High School คือ ระดับชั้นมัธยมศึกษาตอนปลาย Diploma คือ ประกาศนียบัตรวิชาชีพชั้นสูง Bachelor's คือ ปริญญาตรี Master's คือ ปริญญาโท
6	gender (เพศ)	F (Female) คือ เพศหญิง M (Male) คือ เพศชาย
7	age (อายุ)	ตั้งแต่ 25 - 54 ปี
8	years experience (อายุการทำงานในองค์กร)	ตั้งแต่ 1 - 33 ปี
9	KPI (คะแนนผลการปฏิบัติงานของปีก่อนหน้า)	ตั้งแต่ 1.38 - 5 คะแนน
10	competency (คะแนนผลการประเมินสมรรถนะและทักษะของปีก่อนหน้า)	ตั้งแต่ 2.87 - 4.86 คะแนน
11	score 360 (ผลการประเมินรอบทิศแบบ 360 องศา)	ตั้งแต่ 40.8 - 97.78 คะแนน
12	score interview (ผลสัมภาษณ์จากคณะกรรมการ)	ตั้งแต่ 58.04 - 100 คะแนน
13	year promoted (ปีที่ได้รับการปรับตำแหน่ง)	ตั้งแต่ปี 2019 - ปี 2022

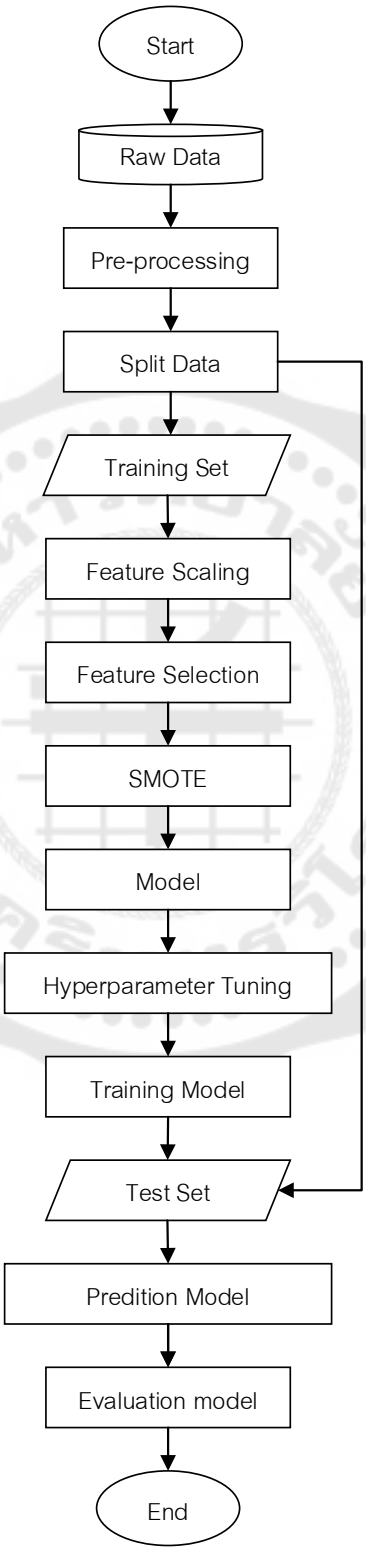
ตาราง 2 (ต่อ)

ลำดับ	ข้อมูลตัวแปร (Variable)	คำอธิบายข้อมูล (Description)
14	is_promoted (ผลการปรับตำแหน่ง)	pass = ผ่าน not pass = ไม่ผ่าน

2. การเลือกกลุ่มตัวอย่าง

ชุดข้อมูลจะถูกแบ่งเป็น 2 ชุด คือ ชุดข้อมูลสำหรับสร้างแบบจำลอง (Training Set) และชุดข้อมูลสำหรับทดลองแบบจำลอง (Test Set) ในอัตราส่วน 70:30 และ 80:20

การออกแบบโครงสร้างและวิธีการดำเนินงานวิจัย



ภาพประกอบ 7 ขั้นตอนการสร้างแบบจำลอง และการดำเนินการวิจัย

จากภาพประกอบ 7 แสดงถึงขั้นตอนการสร้างแบบจำลอง และการดำเนินการของงานวิจัยนี้ เริ่มต้นจากการนำเข้าข้อมูลการปรับตำแหน่งพนักงาน จากนั้นทำการเตรียมข้อมูล หรือ Data Preprocessing เพื่อทำให้ข้อมูลอยู่ในรูปแบบที่ง่ายต่อการนำไปใช้งาน และวิเคราะห์ข้อมูล จากนั้นทำการแบ่งชุดข้อมูลออกเป็น 2 ส่วน คือ ชุดข้อมูลสำหรับสร้างแบบจำลอง หรือ Training Set และ ชุดข้อมูลสำหรับทดสอบแบบจำลอง หรือ Test Set ในอัตราส่วน 2 อัตราส่วน คือ 70:30 และ 80:20 จากนั้นนำชุดข้อมูลสำหรับสร้างแบบจำลองมาทำการปรับช่วงขอบเขตของฟีเจอร์ หรือ Feature scaling ด้วยเทคนิค StandardScaler เพื่อปรับให้ข้อมูลอยู่ในช่วงเดียวกัน จากนั้นทำการคัดเลือกฟีเจอร์ หรือ Feature Selection โดยใช้เทคนิค Recursive Feature Elimination (RFE) เพื่อคัดเลือกฟีเจอร์ที่มีความสำคัญกับประสิทธิภาพของแต่ละแบบจำลอง และหาค่า Feature Importance ของแต่ฟีเจอร์ จากนั้นทำการจัดการความไม่สมดุลของข้อมูลด้วยเทคนิค Synthetic Minority Over-sampling Technique หรือ SMOTE เพื่อสังเคราะห์ข้อมูลในกลุ่มที่มีน้อยให้เพิ่มขึ้น จากนั้นทำการสร้างแบบจำลอง หรือ Model ทั้งหมด 4 แบบจำลอง ได้แก่ Support Vector Machine, Decision Tree, Random Forest และ XGBoost และปรับจูนพารามิเตอร์ หรือ Hyperparameter Tuning ร่วมด้วย ด้วยเทคนิค GridSearchCV เมื่อแบบจำลองทำการเรียนรู้จากชุดข้อมูลสำหรับสร้างแบบจำลองเรียบร้อยแล้ว จึงนำชุดข้อมูลสำหรับทดสอบแบบจำลองมาใช้ในการวัดประสิทธิภาพของแบบจำลองโดยใช้ตาราง Confusion Matrix ได้แก่ Accuracy, Precision, Recall และ F1 Score และทำการเปรียบเทียบประสิทธิภาพของแบบจำลองที่ใช้ฟีเจอร์ทั้งหมด แบบจำลองที่มีการคัดเลือกฟีเจอร์ แบบจำลองที่มีการจัดการความไม่สมดุลของข้อมูล และแบบจำลองที่มีการคัดเลือกฟีเจอร์และการจัดการความไม่สมดุลของข้อมูลร่วมด้วย

การเตรียมข้อมูลและการวิเคราะห์ข้อมูล

1. การนำเข้าข้อมูล

ขั้นตอนการดำเนินงานจะเริ่มต้นจากการนำชุดข้อมูลเพื่อมาทำการแสดงรายละเอียด โดยรายละเอียดของชุดข้อมูลดั้งเดิม ดังภาพประกอบ 8 และ 9

data.head()

	employee_id	position	position_next	group	education	gender	age	years_experience	kpi	competency	score360	score_interview	year_promoted	is_promoted
0	10001	Manager	Senior Manager	Direct Report Precident	Master's	M	40	5	3.95	3.79	82.22	88.57	2019	pass
1	10002	Manager	Senior Manager	Direct Report Precident	Master's	F	39	4	3.43	3.50	71.11	89.29	2019	pass
2	10003	Specialist	Senior Specialist	Information Technology Group	Diploma	M	48	14	3.40	3.17	77.78	83.93	2019	not pass
3	10004	Manager	Associate Vice President	Information Technology Group	Master's	M	39	7	3.68	3.55	57.78	98.21	2019	pass
4	10005	Senior Officer	Specialist	Information Technology Group	Bachelor's	M	36	7	3.30	3.15	60.00	65.71	2019	not pass

ภาพประกอบ 8 ตัวอย่างข้อมูลในตารางของชุดข้อมูลดั้งเดิม

```
data.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 212 entries, 0 to 211
Data columns (total 14 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   employee_id      212 non-null    int64
 1   position         212 non-null    object
 2   position_next    212 non-null    object
 3   group            212 non-null    object
 4   education        212 non-null    object
 5   gender           212 non-null    object
 6   age              212 non-null    int64
 7   years_experience 212 non-null    int64
 8   kpi              208 non-null    float64
 9   competency       209 non-null    float64
 10  score360         212 non-null    float64
 11  score_interview  212 non-null    float64
 12  year_promoted    212 non-null    int64
 13  is_promoted      212 non-null    object
dtypes: float64(4), int64(4), object(6)
memory usage: 23.3+ KB
```

ภาพประกอบ 9 รายละเอียดของชุดข้อมูลดั้งเดิม

2. เตรียมข้อมูล

เมื่อนำเข้าข้อมูลแล้ว จึงมาเตรียมข้อมูลให้เหมาะสมเพื่อให้สามารถอ่านและทำความเข้าใจได้ง่ายขึ้น ดังนี้

1. เปลี่ยนชื่อคอลัมน์ที่มีการเว้นวรรคด้านหลัง เพื่อป้องกันการเรียกใช้คอลัมน์ไม่พบ ดังภาพประกอบ 10

```
data.rename(columns={'years_experience ':'years_experience'},inplace=True)
data.rename(columns={'score360 ':'score360'},inplace=True)
data.rename(columns={'score_interview ':'score_interview'},inplace=True)
data.columns
```

```
Index(['employee_id', 'position', 'position_next', 'group', 'education',
       'gender', 'age', 'years_experience', 'kpi', 'competency', 'score360',
       'score_interview', 'year_promoted', 'is_promoted'],
      dtype='object')
```

ภาพประกอบ 10 รายละเอียดชุดข้อมูลที่ได้รับการเปลี่ยนชื่อคอลัมน์

2. เปลี่ยนชนิดข้อมูล 2 คอลัมน์ คือ employee_id และ year_promoted จาก int64 เป็น category เนื่องจากไม่ใช่ค่าตัวเลขที่จะใช้ในการคำนวณหาค่าต่างๆ แต่เป็นรหัสพนักงานที่ระบุตัวบุคคล และเป็นปีที่ได้รับการปรับตำแหน่ง ดังภาพประกอบ 11

```
data['employee_id'] = data['employee_id'].astype('category')
data['year_promoted'] = data['year_promoted'].astype('category')
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 212 entries, 0 to 211
Data columns (total 14 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   employee_id      212 non-null    category
 1   position         212 non-null    object
 2   position_next    212 non-null    object
 3   group            212 non-null    object
 4   education        212 non-null    object
 5   gender           212 non-null    object
 6   age              212 non-null    int64
 7   years_experience 212 non-null    int64
 8   kpi              208 non-null    float64
 9   competency       209 non-null    float64
 10  score360         212 non-null    float64
 11  score_interview  212 non-null    float64
 12  year_promoted    212 non-null    category
 13  is_promoted      212 non-null    object
dtypes: category(2), float64(4), int64(2), object(6)
memory usage: 30.6+ KB
```

ภาพประกอบ 11 รายละเอียดของชุดข้อมูลที่ได้รับเปลี่ยนชนิดข้อมูลและปรับแต่งให้เหมาะสมแล้ว

3. ตรวจสอบค่าว่าง และข้อมูลที่ซ้ำกันในแต่ละแถว ดังภาพประกอบ 12

```
data.isnull().sum()
```

```
employee_id         0
position            0
position_next       0
group               0
education           0
gender              0
age                 0
years_experience    0
kpi                 4
competency          3
score360            0
score_Interview     0
year_promoted       0
is_promoted         0
dtype: int64
```

```
data.duplicated(keep=False).sum()
```

```
0
```

ภาพประกอบ 12 ตรวจสอบจำนวนค่าว่าง และจำนวนแถวข้อมูลที่ซ้ำกัน

พบค่าว่างคอลัมน์ kpi และ competency จึงทำการตรวจสอบว่าเป็นพนักงานที่มีอายุงานในองค์กร 1 ปี เนื่องจากหากมีอายุงาน 1 ปี อาจจะไม่มีคะแนนในส่วนนี้ จึงทำแทนที่ด้วย 0 ดังภาพประกอบ 13

```
null_filter = data[data.is_promoted == 'pass']

print("ค่าว่างในคอลัมน์ kpi = ", null_filter.kpi.isnull().sum())

print("ค่าว่างในคอลัมน์ competency = ", null_filter.competency.isnull().sum())

print("ค่าว่างในคอลัมน์ kpi ของพนักงานที่มีอายุงาน 1 ปี = ", null_filter[null_filter.years_experience == 1].kpi.isnull().sum())

print("ค่าว่างในคอลัมน์ competency ของพนักงานที่มีอายุงาน 1 ปี = ", null_filter[null_filter.years_experience == 1].competency.isnull().sum())

data.kpi = data.kpi.fillna(0)
data.competency = data.competency.fillna(0)
```

```
ค่าว่างในคอลัมน์ kpi =  4
ค่าว่างในคอลัมน์ competency =  3
ค่าว่างในคอลัมน์ kpi ของพนักงานที่มีอายุงาน 1 ปี =  2
ค่าว่างในคอลัมน์ competency ของพนักงานที่มีอายุงาน 1 ปี =  2
```

ภาพประกอบ 13 ตรวจสอบพนักงานที่มีอายุงานในองค์กร 1 ปี และการแทนที่ค่าว่างด้วย 0

3. การวิเคราะห์ข้อมูล

เมื่อเตรียมข้อมูลให้เหมาะสมแล้ว จะทำการวิเคราะห์ข้อมูลต่างๆ เพื่อค้นหาข้อมูลที่สำคัญที่แฝงอยู่ในกลุ่มข้อมูลนั้น โดยจะทำการหารูปแบบความเชื่อมโยงระหว่างกัน เริ่มจากการหาค่าทางสถิติเบื้องต้น ดังภาพประกอบ 14

```
data.describe()
```

	age	years_experience	kpi	competency	score360	score_interview
count	212.000000	212.000000	212.000000	212.000000	212.000000	212.000000
mean	37.905660	8.150943	3.679104	3.728679	77.525849	85.993349
std	6.611104	7.669466	0.837425	0.616734	19.944584	13.779962
min	25.000000	1.000000	0.000000	0.000000	0.000000	0.000000
25%	33.000000	3.000000	3.300000	3.467500	75.500000	81.385000
50%	37.000000	6.000000	3.785000	3.690000	82.950000	87.550000
75%	42.000000	9.000000	4.157500	4.030000	87.862500	95.000000
max	54.000000	33.000000	5.000000	4.860000	97.780000	100.000000

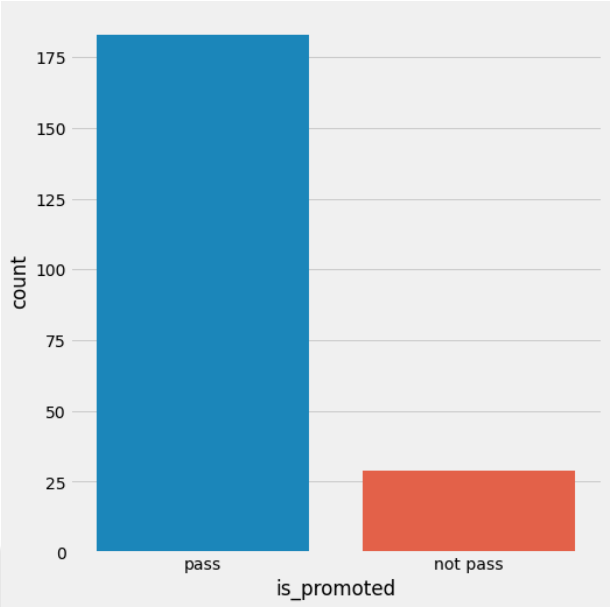
ภาพประกอบ 14 รายละเอียดค่าทางสถิติเบื้องต้น

ทำการเรียกดูข้อมูลจากคอลัมน์ต่างๆ ว่ามีตัวแปรแต่ละตัวแปรเป็นจำนวนเท่าไหร่ และสัดส่วนของแต่ละตัวแปรกับผลการปรับตำแหน่งว่าเป็นจำนวนเท่าใด เพื่อหาว่าข้อมูลมีความเป็นไปอย่างไร โดยแสดงเป็นกราฟแท่ง ซึ่งคอลัมน์แรกที่จะเรียกดูข้อมูล คือ คอลัมน์ is_promoted หรือ ผลการปรับตำแหน่ง ดังภาพประกอบ 15

```
print(data.is_promoted.value_counts())
```

```
pass       183
not pass    29
Name: is_promoted, dtype: int64
```

ภาพประกอบ 15 จำนวนข้อมูลผลการปรับตำแหน่ง



ภาพประกอบ 16 กราฟแท่งแสดงจำนวนข้อมูลผลการปรับตำแหน่ง

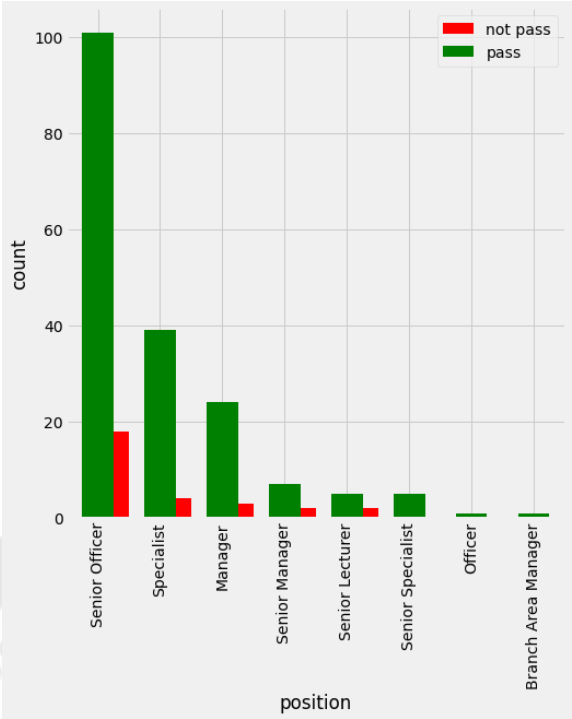
จากภาพประกอบ 16 แสดงให้เห็นว่าในชุดข้อมูลมีจำนวนผู้ที่ได้รับการปรับตำแหน่งมากกว่าผู้ที่ไม่ได้รับการปรับตำแหน่งเป็นอย่างมาก สะท้อนให้เห็นว่าชุดข้อมูลมีความไม่สมดุลกัน

ต่อมาจะเริ่มทำการเรียกดูสัดส่วนของแต่ละตัวแปรกับผลการปรับตำแหน่งว่าเป็นจำนวนเท่าใด โดยแสดงเป็นกราฟในรูปแบบต่างๆ แต่จะไม่ทำการเรียกดูสัดส่วนข้อมูล 3 คอลัมน์ คือ ข้อมูลรหัสพนักงาน, ข้อมูลตำแหน่งที่ปรับขึ้น และปีที่ปรับ

```
print(data.position.value_counts())
```

```
Senior Officer        119
Specialist             42
Manager                28
Senior Manager          9
Senior Lecturer         5
Senior Specialist       5
Branch Area Manager     3
Officer                 1
Name: position, dtype: int64
```

ภาพประกอบ 17 จำนวนข้อมูลของแต่ละตำแหน่ง



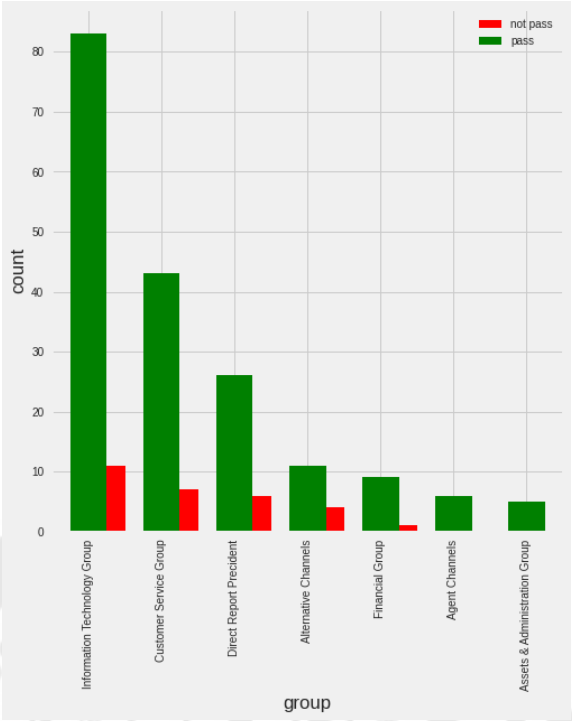
ภาพประกอบ 18 กราฟแท่งแสดงจำนวนของแต่ละตำแหน่งกับผลการปรับตำแหน่ง

จากภาพประกอบ 17 และ 18 แสดงให้เห็นว่าตำแหน่ง Senior Officer หรือเจ้าหน้าที่อาวุโส เป็นตำแหน่งที่มีจำนวนมากที่สุดในชุดข้อมูล และผู้ที่ผ่านการปรับตำแหน่งมากที่สุดเช่นกัน โดยรองลงมา คือ ตำแหน่งSpecialist หรือผู้เชี่ยวชาญ และ Manager หรือผู้จัดการ ซึ่งเมื่อเทียบกับจำนวนของตำแหน่งอื่นๆ จะมีจำนวนมากกว่าอย่างเห็นได้ชัด

```
print(data.group.value_counts())
```

```
Information Technology Group     89
Customer Service Group           54
Direct Report Precident          30
Agent Channels                   13
Alternative Channels             12
Financial Group                   9
Assets & Administration Group     5
Name: group, dtype: int64
```

ภาพประกอบ 19 จำนวนข้อมูลของแต่ละสายงาน



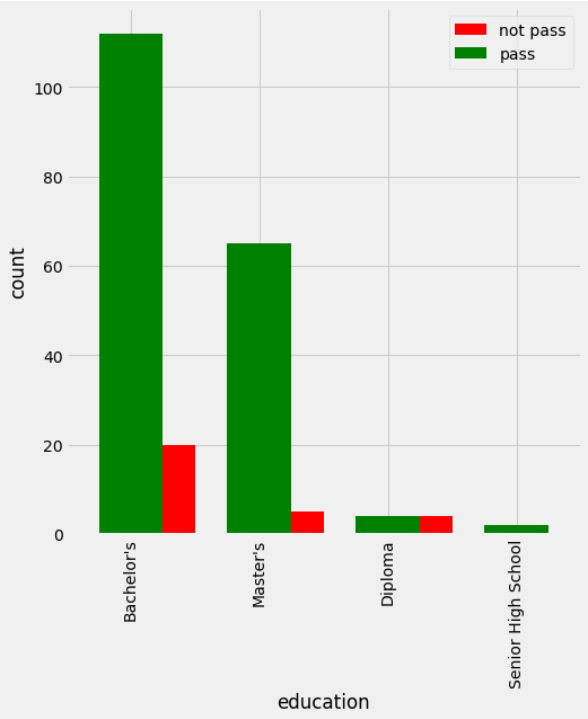
ภาพประกอบ 20 กราฟแท่งแสดงจำนวนของแต่ละสายงานกับผลการปรับตำแหน่ง

จากภาพประกอบ 19 และ 20 แสดงให้เห็นว่าสายงาน Information Technology Group คือ สายงานเทคโนโลยีสารสนเทศ เป็นสายงานที่มีจำนวนคนมากที่สุดในชุดข้อมูล และมีผู้ที่ผ่านการปรับตำแหน่งมากที่สุดเช่นกัน โดยรองลงมา คือ สายงาน Customer Service Group หรือสายงานบริการลูกค้าธุรกิจประกันชีวิต และสายงาน Direct Report President หรือสายงานกลาง ซึ่งเมื่อเทียบกับจำนวนของตำแหน่งอื่นๆ จะมีจำนวนมากกว่าอย่างเห็นได้ชัด ซึ่งสายงาน Agent Channels คือ สายงานประกันชีวิตด้านช่องทางตัวแทน และสายงาน Assets & Administration Group หรือสายงานทรัพย์สินและธุรการ เป็นสายงานที่ไม่มีผู้ไม่ผ่านการปรับตำแหน่งเลย

```
print(data.education.value_counts())
```

```
Bachelor's          132
Master's             70
Diploma               8
Senior High School    2
Name: education, dtype: int64
```

ภาพประกอบ 21 จำนวนข้อมูลของแต่ละระดับการศึกษา



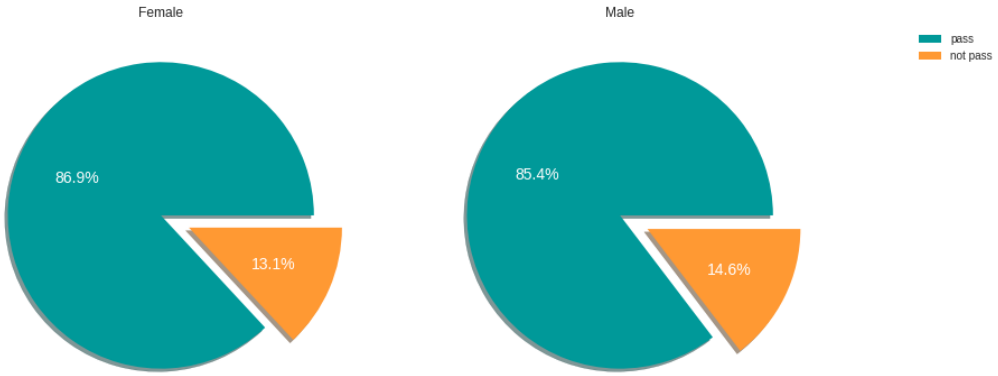
ภาพประกอบ 22 กราฟแท่งแสดงจำนวนของแต่ละระดับการศึกษากับผลการปรับตำแหน่ง

จากภาพประกอบ 21 และ 22 แสดงให้เห็นว่าการศึกษาระดับ Bachelor's หรือปริญญาตรี เป็นระดับการศึกษาที่มีจำนวนคนมากที่สุดในชุดข้อมูล และมีผู้ที่ผ่านการปรับตำแหน่งมากที่สุดเช่นกัน โดยรองลงมา คือ การศึกษาระดับ Master's หรือปริญญาโท ซึ่งเมื่อเทียบกับจำนวนของระดับการศึกษาอีก 2 ระดับ จะมีจำนวนมากกว่าอย่างเห็นได้ชัด

```
print(data.gender.value_counts())
```

```
F    130
M     82
Name: gender, dtype: int64
```

ภาพประกอบ 23 จำนวนข้อมูลของแต่ละเพศ



ภาพประกอบ 24 กราฟวงกลมแสดงสัดส่วนของเพศกับผลการปรับตำแหน่ง

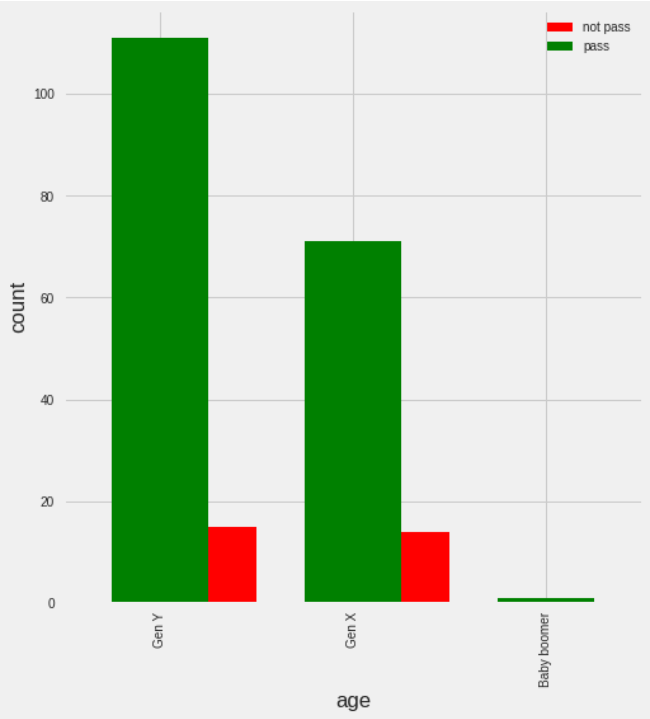
จากภาพประกอบ 23 และ 24 แสดงให้เห็นว่าเพศหญิง หรือ Female มีจำนวนมากว่าเพศชาย แต่สัดส่วนของผู้ที่ผ่านการปรับตำแหน่งและไม่ผ่านการปรับตำแหน่งมีสัดส่วนใกล้เคียงกัน

ซึ่งในเรียกดูข้อมูลหลังจากนี้จะมีการจัดกลุ่มของข้อมูลตามค่าแต่ละช่วงตามความเหมาะสม และสร้างคอลัมน์ใหม่ เพื่อรองรับข้อมูลที่ถูกจัดกลุ่มใหม่ เพื่อไม่ให้กระทบกับข้อมูลจริง ทำให้สามารถเรียกดูข้อมูลได้ง่ายขึ้น

```
data['age_label'] = pd.cut(data.age,bins=[22,38,53,72],labels=['Gen Y','Gen X','Baby boomer'])
print(data.age_label.value_counts())
```

```
Gen Y         125
Gen X          86
Baby boomer     1
Name: age_label, dtype: int64
```

ภาพประกอบ 25 การจัดกลุ่มช่วงอายุและจำนวนข้อมูลของแต่ละช่วงอายุ



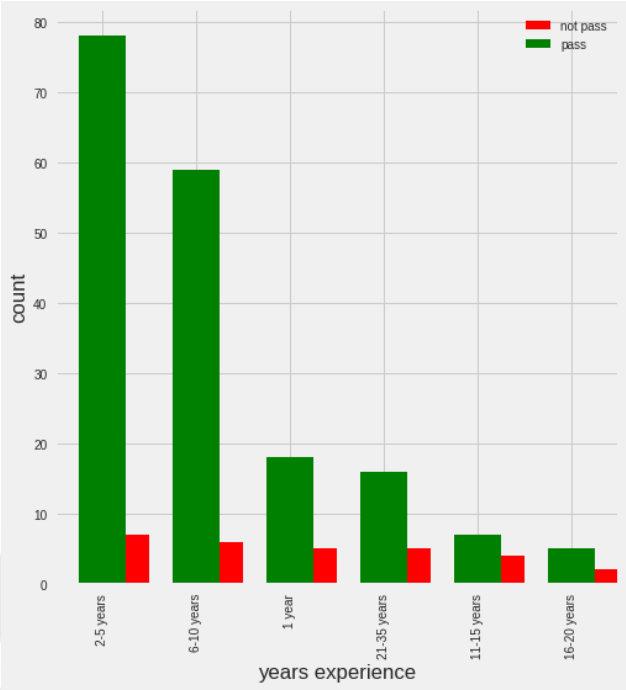
ภาพประกอบ 26 กราฟแท่งแสดงจำนวนของแต่ละช่วงอายุกับผลการปรับตำแหน่ง

จากภาพประกอบ 25 และ 26 แสดงให้เห็นว่ากลุ่ม Gen Y หรือกลุ่มที่มีช่วงอายุ 22-38 ปี เป็นกลุ่มที่มีจำนวนคนมากที่สุดในชุดข้อมูล และมีผู้ที่ผ่านการปรับตำแหน่งมากที่สุดเช่นกัน โดยรองลงมา คือ กลุ่ม Gen Y หรือกลุ่มที่มีช่วงอายุ 39-53 ปี และกลุ่ม Baby boomer มีจำนวนน้อยที่สุด เนื่องจากองค์กรมีเกษียณอายุหลังจาก 55 ปี เป็นต้นไป

```
data['experience_label'] = pd.cut(data.years_experience,bins=[0,1,5,10,15,20,35],
                                  labels=['1 year','2-5 years','6-10 years','11-15 years','16-20 years','21-35 years'])
print(data.experience_label.value_counts())
```

```
2-5 years      83
6-10 years     65
21-35 years    23
1 year         20
11-15 years    11
16-20 years    10
Name: experience_label, dtype: int64
```

ภาพประกอบ 27 การจัดกลุ่มอายุงานจำนวนข้อมูลของแต่ละกลุ่มอายุงาน



ภาพประกอบ 28 กราฟแท่งแสดงจำนวนของแต่ละกลุ่มอายุงานกับผลการปรับตำแหน่ง

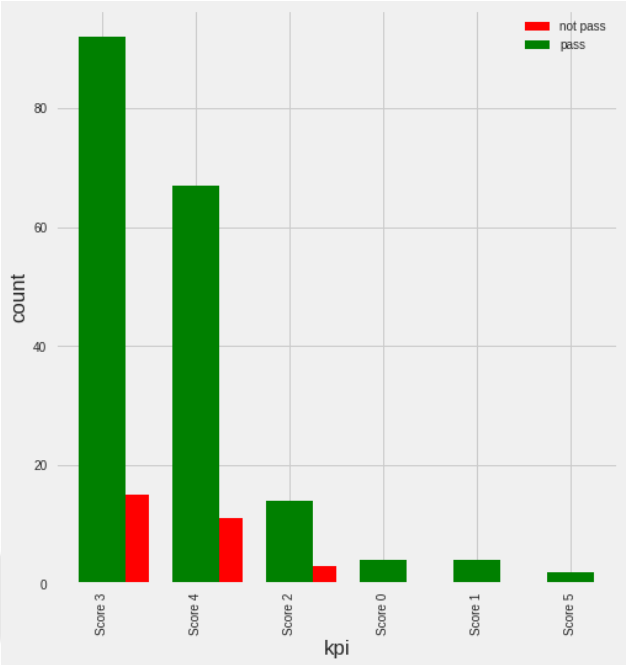
จากภาพประกอบที่ 27 และ 28 แสดงให้เห็นว่ากลุ่มอายุงาน 2-5 ปี เป็นกลุ่มอายุงานที่มีจำนวนคนมากที่สุดในชุดข้อมูล และมีผู้ที่ผ่านการปรับตำแหน่งมากที่สุดเช่นกัน โดยรองลงมา คือ กลุ่มอายุงาน 6-10 ปี ซึ่งเมื่อเทียบกับจำนวนของกลุ่มอายุงานอื่นๆ จะมีจำนวนมากกว่าอย่างเห็นได้ชัด

```
def decode(val):
    if  val == 0 : return 'Score 0'
    elif val <= 1.99 : return 'Score 1'
    elif val <= 2.99 : return 'Score 2'
    elif val <= 3.99 : return 'Score 3'
    elif val <= 4.99 : return 'Score 4'
    else : return 'Score 5'

data['kpi_label'] = data.kpi.apply(decode)
print(data.kpi_label.value_counts())
```

```
Score 3    107
Score 4     78
Score 2     17
Score 0      4
Score 1      4
Score 5      2
Name: kpi_label, dtype: int64
```

ภาพประกอบ 29 การจัดกลุ่มคะแนนการปฏิบัติงานของปีก่อนหน้าและจำนวนข้อมูลของแต่ละกลุ่มคะแนนการปฏิบัติงานของปีก่อนหน้า



ภาพประกอบ 30 กราฟแท่งแสดงจำนวนของแต่ละกลุ่มคะแนนการปฏิบัติงานของปีก่อนหน้ากับผลการปรับตำแหน่ง

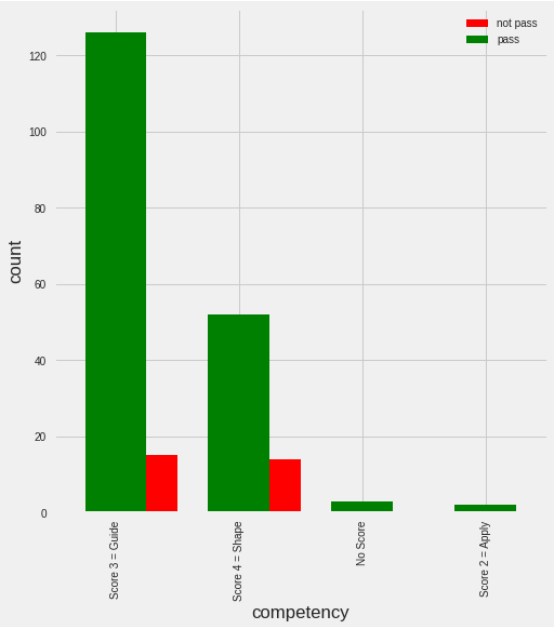
จากภาพประกอบที่ 29 และ 30 แสดงให้เห็นว่ากลุ่มคะแนนการปฏิบัติงาน 3 คะแนน เป็นกลุ่มคะแนนการปฏิบัติงานที่มีจำนวนคนมากที่สุดในชุดข้อมูล และมีผู้ที่ผ่านการปรับตำแหน่งมากที่สุดเช่นกัน โดยรองลงมา คือ กลุ่มคะแนนการปฏิบัติงาน 4 คะแนน ซึ่งเมื่อเทียบกับจำนวนของกลุ่มคะแนนการปฏิบัติงานกลุ่มอื่นๆ จะมีจำนวนมากกว่าอย่างเห็นได้ชัด

```
def decode(val):
    if  val == 0 : return 'No Score'
    elif val <= 1.99 : return 'Score 1 = Learn'
    elif val <= 2.99 : return 'Score 2 = Apply'
    elif val <= 3.99 : return 'Score 3 = Guide'
    elif val <= 4.99 : return 'Score 4 = Shape'
    else : return 'Score 5 = Envision'

data['competency_label'] = data.competency.apply(decode)
print(data.competency_label.value_counts())
```

```
Score 3 = Guide    141
Score 4 = Shape     66
No Score             3
Score 2 = Apply      2
Name: competency_label, dtype: int64
```

ภาพประกอบ 31 การจัดกลุ่มคะแนนการประเมินสมรรถนะและทักษะของปีก่อนหน้าและจำนวนข้อมูลของแต่ละกลุ่มคะแนนการประเมินสมรรถนะและทักษะของปีก่อนหน้า



ภาพประกอบ 32 กราฟแท่งแสดงจำนวนของกลุ่มคะแนนการประเมินสมรรถนะและทักษะของปีก่อนหน้ากับผลการปรับตำแหน่ง

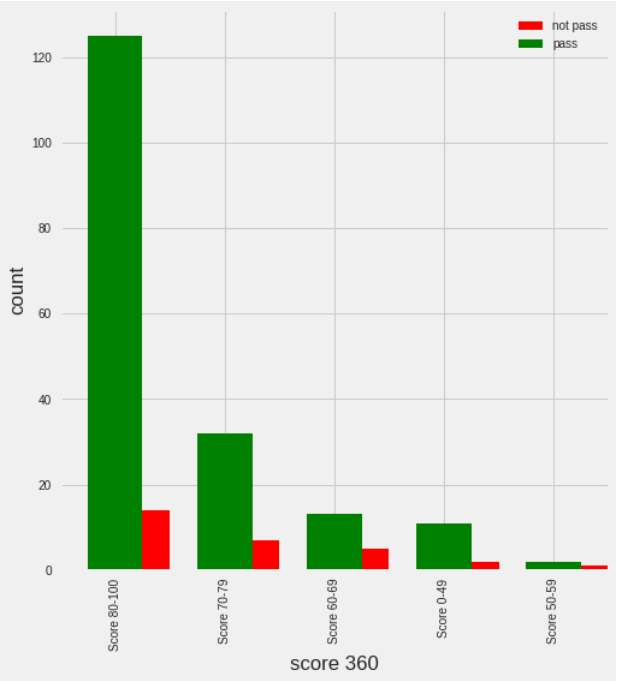
จากภาพประกอบที่ 31 และ 32 แสดงให้เห็นว่ากลุ่มคะแนนการประเมินสมรรถนะและทักษะ 3 คะแนน เป็นกลุ่มคะแนนการประเมินสมรรถนะและทักษะที่มีจำนวนคนมากที่สุดในชุดข้อมูล และมีผู้ที่ผ่านการปรับตำแหน่งมากที่สุดเช่นกัน โดยรองลงมา คือ กลุ่มคะแนนการประเมินสมรรถนะและทักษะ 4 คะแนน ซึ่งเมื่อเทียบกับจำนวนของกลุ่มคะแนนการประเมินสมรรถนะและทักษะกลุ่มอื่นๆ จะมีจำนวนมากกว่าอย่างเห็นได้ชัด

```
def decode(val):
    if val <= 49 : return 'Score 0-49'
    elif val <= 59 : return 'Score 50-59'
    elif val <= 69 : return 'Score 60-69'
    elif val <= 79 : return 'Score 70-79'
    else : return 'Score 80-100'

data['score360_label'] = data.score360.apply(decode)
print(data.score360_label.value_counts())
```

```
Score 80-100    139
Score 70-79      37
Score 60-69      20
Score 0-49       13
Score 50-59       3
Name: score360_label, dtype: int64
```

ภาพประกอบ 33 การจัดกลุ่มคะแนนการประเมินรอบทิศแบบ 360 องศา และจำนวนข้อมูลของแต่ละกลุ่มคะแนนการประเมินรอบทิศแบบ 360 องศา



ภาพประกอบ 34 กราฟแท่งแสดงจำนวนของแต่ละกลุ่มคะแนนการประเมินรอบทิศแบบ 360 องศากับผลการปรับตำแหน่ง

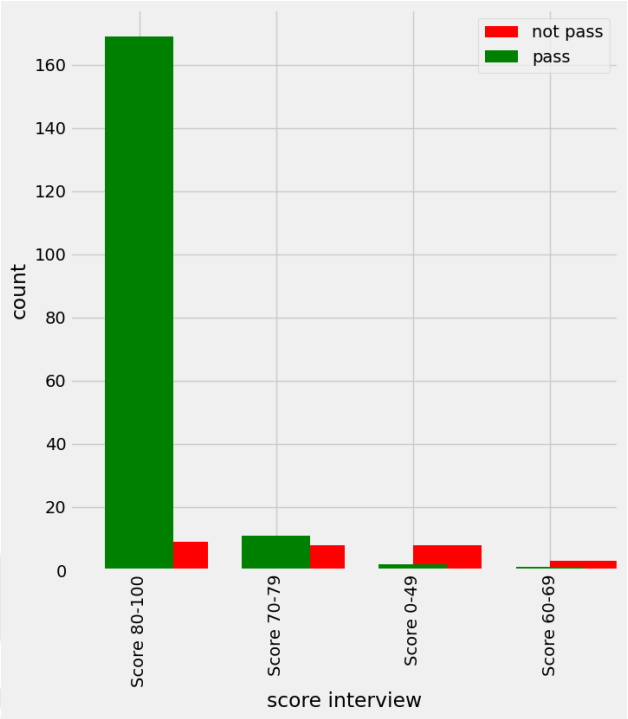
จากภาพประกอบที่ 33 และ 34 แสดงให้เห็นว่ากลุ่มคะแนนการประเมินรอบทิศแบบ 360 องศา 80-100 คะแนน เป็นกลุ่มคะแนนการประเมินสมรรถนะและทักษะที่มีจำนวนคนมากที่สุดในชุดข้อมูล และมีผู้ที่ผ่านการปรับตำแหน่งมากที่สุดเช่นกัน และมากกว่ากลุ่มคะแนนการประเมินรอบทิศแบบ 360 องศากลุ่มอื่นๆ อย่างเห็นได้ชัด

```
def decode(val):
    if val <= 49 : return 'Score 0-49'
    elif val <= 59 : return 'Score 50-59'
    elif val <= 69 : return 'Score 60-69'
    elif val <= 79 : return 'Score 70-79'
    else : return 'Score 80-100'

data['score_interview_label'] = data.score_interview.apply(decode)
print(data.score_interview_label.value_counts())
```

```
Score 80-100    178
Score 70-79      19
Score 60-69       9
Score 50-59       3
Score 0-49        3
Name: score_interview_label, dtype: int64
```

ภาพประกอบ 35 การจัดกลุ่มคะแนนผลสัมภาษณ์จากคณะกรรมการและจำนวนข้อมูลของแต่ละกลุ่มคะแนนผลสัมภาษณ์จากคณะกรรมการ



ภาพประกอบ 36 กราฟแท่งแสดงจำนวนของแต่ละกลุ่มคะแนนผลสัมภาษณ์จากคณะกรรมการกับผลการปรับตำแหน่ง

จากภาพประกอบที่ 35 และ 36 แสดงให้เห็นว่ากลุ่มคะแนนผลสัมภาษณ์จากคณะกรรมการ 80-100 คะแนน เป็นกลุ่มคะแนนผลสัมภาษณ์จากคณะกรรมการที่มีจำนวนคนมากที่สุดในชุดข้อมูล และมีผู้ที่ผ่านการปรับตำแหน่งมากที่สุดเช่นกัน และมากกว่ากลุ่มคะแนนผลสัมภาษณ์จากคณะกรรมการกลุ่มอื่นๆ อย่างเห็นได้ชัด และกลุ่มคะแนนผลสัมภาษณ์จากคณะกรรมการ 0-49 คะแนน เป็นกลุ่มที่มีจำนวนผู้ที่ไม่ผ่านการปรับตำแหน่งมากกว่าผู้ที่ได้รับการปรับตำแหน่ง

4. แปลงข้อมูลเป็นตัวเลข

ทำการแปลงข้อมูลโดยใช้เทคนิค Label encoder เพื่อแปลงข้อมูลให้เป็นตัวเลขกับ 5 ฟีเจอร์ คือ is_promoted หรือผลการปรับตำแหน่ง, position หรือตำแหน่งงานปัจจุบัน, group หรือสังกัดสายงาน, education หรือการศึกษาสูงสุด และ gender หรือเพศ โดย is_promoted หรือผลการปรับตำแหน่ง จะมีค่า 1 = pass หรือผ่านการปรับตำแหน่ง และ 0 คือ not pass หรือไม่ผ่านการปรับตำแหน่ง ดังภาพประกอบ 37 และ 38

```
from sklearn.preprocessing import LabelEncoder

labelencoder = LabelEncoder()
data["is_promoted"]=labelencoder.fit_transform(data["is_promoted"])
data["position"]=labelencoder.fit_transform(data["position"])
data["group"]=labelencoder.fit_transform(data["group"])
data["education"]=labelencoder.fit_transform(data["education"])
data["gender"]=labelencoder.fit_transform(data["gender"])
```

ภาพประกอบ 37 คำสั่งการแปลงข้อมูลให้เป็นตัวเลข โดยใช้Label encoder

	employee_id	position	position_next	group	education	gender	age	years_experience	kpi	competency	score360	score_interview	year_promoted	is_promoted
0	10001	1	Senior Manager	4	2	1	40	5	3.95	3.79	82.22	88.57	2019	1
1	10002	1	Senior Manager	4	2	0	39	4	3.43	3.50	71.11	89.29	2019	1
2	10003	7	Senior Specialist	6	1	1	48	14	3.40	3.17	77.78	83.93	2019	0
3	10004	1	Associate Vice President	6	2	1	39	7	3.68	3.55	57.78	98.21	2019	1
4	10005	5	Specialist	6	0	1	36	7	3.30	3.15	60.00	65.71	2019	0

ภาพประกอบ 38 ตัวอย่างข้อมูลในตารางของชุดข้อมูลที่ทำการแปลงข้อมูลให้เป็นตัวเลข

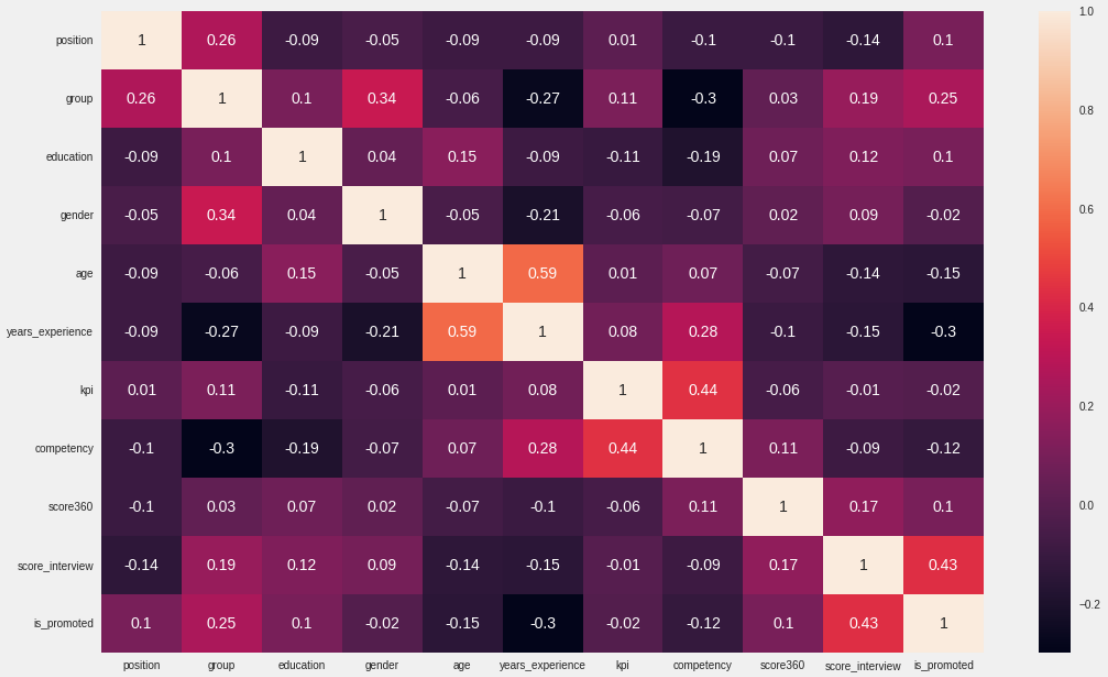
5. ค่าความสัมพันธ์ของข้อมูล (Correlation)

เมื่อแปลงข้อมูลให้เป็นตัวเลขแล้ว จึงจะสามารถหาค่าความสัมพันธ์ของข้อมูล (Correlation) ได้ เพื่อตรวจสอบว่าฟีเจอร์ใดมีความสัมพันธ์กับจัดกลุ่มการปรับตำแหน่งมากที่สุด ดังภาพประกอบ 39 และ 40

```
plt.figure(figsize=(16, 10))
correlation_matrix = data.corr().round(2)
sns.heatmap(data=correlation_matrix, annot=True)

plt.show()
plt.close()
```

ภาพประกอบ 39 คำสั่งการหาค่าความสัมพันธ์ของข้อมูล



ภาพประกอบ 40 ค่าความสัมพันธ์ของข้อมูล (Correlation)

โดยค่าความสัมพันธ์ระหว่างข้อมูลที่ได้ คือ ฟีเจอร์ score interview หรือผลสัมภาษณ์จากคณะกรรมการ มีความสัมพันธ์มากที่สุดกับจัดกลุ่มการปรับตำแหน่ง โดยให้ค่า Correlation เท่ากับ 0.43 รองลงมา คือ ฟีเจอร์ position หรือ ตำแหน่งงานปัจจุบัน, ฟีเจอร์ education หรือการศึกษา และฟีเจอร์ score 360 หรือผลการประเมินรอบทิศ แบบ 360 องศา โดยให้ค่า Correlation เท่ากัน คือ 0.1

การแบ่งชุดข้อมูลและการปรับช่วงขอบเขตฟีเจอร์

1. การแบ่งชุดข้อมูล

นำชุดข้อมูลไปแบ่งเป็น 2 ส่วน คือ ชุดข้อมูลสำหรับสร้างแบบจำลอง (Training Set) และชุดข้อมูลสำหรับทดสอบแบบจำลอง (Test Set) ด้วยเทคนิค Train-test split ในอัตราส่วน 2 อัตราส่วน คือ 70:30 และ 80:20 โดยอัตราส่วน 70:30 จะมีข้อมูลสำหรับสร้างแบบจำลองทั้งหมด 148 ข้อมูล และข้อมูลสำหรับทดสอบแบบจำลองทั้งหมด 64 ข้อมูล อัตราส่วน 80 : 20 จะมีข้อมูลสำหรับสร้างแบบจำลองทั้งหมด 169 ข้อมูล และข้อมูลสำหรับทดสอบแบบจำลองทั้งหมด 23 ข้อมูล โดยการเตรียมข้อมูลต่อไปจะใช้กับชุดข้อมูลสำหรับสร้างแบบจำลองเท่านั้น

2. ปรับช่วงขอบเขตของฟีเจอร์ (Feature Scaling)

ต่อมาจะทำการปรับช่วงขอบเขตของฟีเจอร์ เนื่องจากช่วงข้อมูลของฟีเจอร์มีความหลากหลาย ซึ่งจะใช้เทคนิค StandardScaler ที่เป็นการปรับค่าข้อมูลตัวเลขโดยคำนวณจากค่าเฉลี่ยและค่าส่วนเบี่ยงเบนมาตรฐาน เพื่อให้ข้อมูลตัวเลขอยู่ภายในขอบเขตเดียวกัน ดังภาพประกอบ 41

	0	1	2	3	4	5	6	7	8	9
0	0.236153	-0.628458	-0.752474	-0.8142	-1.076136	-0.404245	-0.404810	0.290848	0.454998	0.150511
1	0.236153	-0.628458	0.298150	-0.8142	2.010455	2.536030	-1.491237	1.043443	0.324317	-0.561218
2	0.236153	-0.087692	1.348773	-0.8142	-0.767477	-0.659921	-0.020034	-0.535531	-0.008326	-0.261543
3	0.236153	-0.628458	-0.752474	-0.8142	0.930148	2.152516	0.828737	0.645010	0.469254	-0.321478
4	0.236153	0.993840	-0.752474	1.2282	-0.304488	-0.148569	-0.472712	-0.653585	-3.679273	-1.053435

ภาพประกอบ 41 ตัวอย่างข้อมูลที่ทำการปรับช่วงขอบเขตของฟีเจอร์

การคัดเลือกฟีเจอร์และการจัดการความไม่สมดุลของข้อมูล

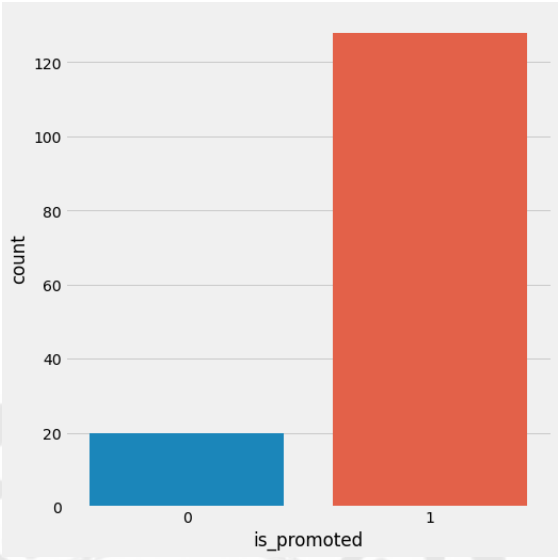
1. การคัดเลือกฟีเจอร์ (Feature Selection)

ในงานวิจัยนี้ใช้การคัดเลือกฟีเจอร์แบบ Wrapper Methods โดยใช้เทคนิค Recursive Feature Elimination (RFE) กับแบบจำลองทั้ง 4 แบบจำลอง ได้แก่ Support Vector Machine, Decision Tree, Random Forest และ XGBoost เพื่อหาฟีเจอร์ที่มีความสำคัญกับประสิทธิภาพของแต่ละแบบจำลอง โดยจะมีการจัดให้อันดับให้กับฟีเจอร์แต่ละตัว และทำการหาค่า Feature Importance ของแต่ละฟีเจอร์

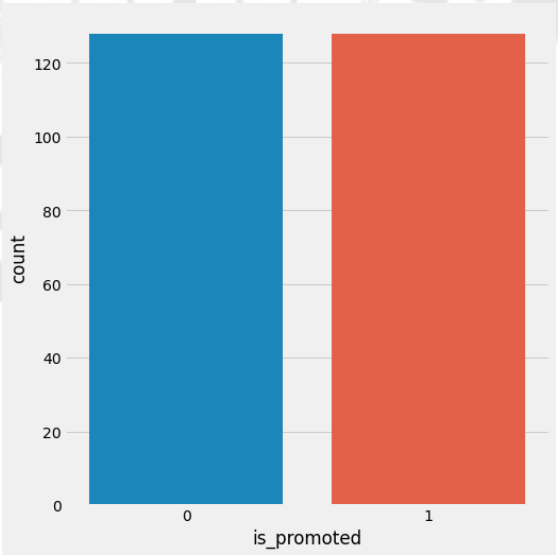
2. การจัดการความไม่สมดุลของข้อมูล

จากการสำรวจข้อมูลและแบ่งชุดข้อมูลพบว่าข้อมูลผลการปรับตำแหน่ง ในชุดข้อมูลสำหรับสร้างแบบจำลอง เมื่อแบ่งชุดข้อมูลในอัตราส่วน 70:30 มีข้อมูลพนักงานผ่านการปรับตำแหน่งจำนวน 128 คน และพนักงานที่ไม่ผ่านการปรับตำแหน่งจำนวน 20 คน และอัตราส่วน 80:20 มีข้อมูลพนักงานผ่านการปรับตำแหน่งจำนวน 146 คน และพนักงานที่ไม่ผ่านการปรับตำแหน่งจำนวน 23 คน เห็นได้ว่าชุดข้อมูลนี้มีลักษณะของข้อมูลที่ไม่สมดุลกัน อาจส่งผลต่อประสิทธิภาพของการทำนาย จึงทำการปรับความสมดุลของข้อมูล โดยทำให้ข้อมูลพนักงานผ่าน

และไม่ผ่านการปรับตำแหน่งมีจำนวนเท่ากัน ซึ่งจะใช้เทคนิค Synthetic Minority Over-sampling Technique หรือ SMOTE โดยกำหนดค่า K-neighbors = 5 ดังภาพประกอบ 42 และ43



ภาพประกอบ 42 กราฟแท่งแสดงจำนวนข้อมูลผลการปรับตำแหน่งก่อนการทำ SMOTE



ภาพประกอบ 43 กราฟแท่งแสดงจำนวนข้อมูลผลการปรับตำแหน่งหลังการทำ SMOTE

หลังการทำ SMOTE ทำให้ข้อมูลผลการปรับตำแหน่งมีจำนวนคนผ่านการปรับตำแหน่ง คือ 1 และไม่ผ่านการปรับตำแหน่ง คือ 0 เท่ากัน โดยชุดข้อมูลในอัตราส่วน 70:30 มีข้อมูลทั้ง 2

กลุ่มเท่ากันที่ 128 คน รวม 256 ข้อมูล และอัตราส่วน 80:20 มีข้อมูลทั้ง 2 กลุ่มเท่ากันที่ 146 คน รวม 292 ข้อมูล ที่จะใช้เป็นชุดข้อมูลสำหรับสร้างแบบจำลอง

การสร้างแบบจำลอง และการประเมินประสิทธิภาพแบบจำลอง

1. การสร้างแบบจำลอง

แบบจำลองที่ใช้ในงานวิจัยนี้ มี 4 แบบจำลอง ได้แก่ Support Vector Machine, Decision Tree, Random Forest และ XGBoost ในการสร้างแบบจำลองจะทำการปรับจูนพารามิเตอร์ (Hyperparameter Tuning) ร่วมด้วย โดยจะใช้เทคนิค GridSearchCV เพื่อค้นหาค่า Hyperparameter จากค่าที่กำหนดไว้แล้ว และเลือกค่าที่ทำให้แบบจำลองมีประสิทธิภาพสูงที่สุด ดังตาราง 3

1.1 Support Vector Machine

สำหรับแบบจำลอง Support Vector Machine ได้ทำการปรับจูนพารามิเตอร์ (Hyperparameter Tuning) ทั้งหมด 3 ตัว ดังนี้

1. C คือ ขนาดของการทำ Regularization สำหรับค่าที่กำหนด คือ 0.01, 0.1, 1, 10, 100

2. kernel คือ ฟังก์ชันที่ใช้แปลงมิติข้อมูลให้อยู่ในแนวรอบของ Hyperplane สำหรับค่าที่กำหนด คือ linear, rbf, poly

3. gamma คือ สัมประสิทธิ์ของ kernel สำหรับค่าที่กำหนด คือ 0.1, 1, 10, 100

1.2 Decision Tree

สำหรับแบบจำลอง Decision Tree ได้ทำการปรับจูนพารามิเตอร์ (Hyperparameter Tuning) ทั้งหมด 4 ตัว ดังนี้

1. max_depth คือ จำนวนความลึกของต้นไม้ สำหรับค่าที่กำหนด คือ 2 ถึง 10

2. min_samples_split คือ จำนวนขั้นต่ำที่จำเป็นในโหนดในการแยกโหนด สำหรับค่าที่กำหนด คือ 2 ถึง 10

3. min_samples_leaf คือ จำนวนข้อมูลขั้นต่ำใน Leaf Node สำหรับค่าที่กำหนด คือ 1 ถึง 10

4. max_leaf_nodes คือ กำหนดจำนวน leaf node ที่ สำหรับค่าที่กำหนด คือ 5, 10, 15, 20

1.3 Random Forest

สำหรับแบบจำลอง Random Forest ได้ทำการปรับจูนพารามิเตอร์ (Hyperparameter Tuning) ทั้งหมด 4 ตัว ดังนี้

1. n_estimators คือ จำนวนต้นไม้ สำหรับค่าที่กำหนด คือ 100, 200, 300

2. max_depth คือ จำนวนความลึกของต้นไม้ สำหรับค่าที่กำหนด คือ 2, 5, 7, 10

3. min_samples_split คือ จำนวนขั้นต่ำที่จำเป็นในโหนดในการแยกโหนด สำหรับค่าที่กำหนด คือ 2, 5, 10

4. min_samples_leaf คือ จำนวนข้อมูลขั้นต่ำใน Leaf Node สำหรับค่าที่กำหนด คือ 1, 2, 4

1.4 XGBoost

สำหรับแบบจำลอง XGBoost ได้ทำการปรับจูนพารามิเตอร์ (Hyperparameter Tuning) ทั้งหมด 3 ตัว ดังนี้

1. n_estimators คือ จำนวนต้นไม้ สำหรับค่าที่กำหนด คือ 100, 200, 300, 400, 500

2. max_depth คือ จำนวนความลึกของต้นไม้ สำหรับค่าที่กำหนด คือ 3 ถึง 10

3. learning_rate คือ ค่าน้ำหนักของการเปลี่ยนแปลงแบบจำลองใน 1 รอบ สำหรับค่าที่กำหนด คือ 0.01, 0.1, 0.2, 0.3

ตาราง 3 การกำหนดพารามิเตอร์ของแต่ละอัลกอริธึม

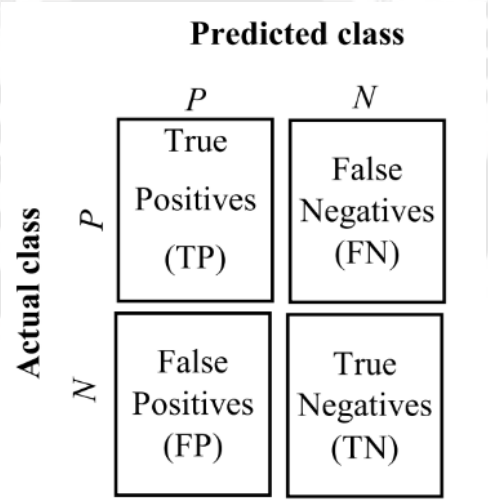
แบบจำลอง	พารามิเตอร์
Support Vector Machine	C: [0.01, 0.1, 1, 10, 100] kernel: ['linear', 'rbf', 'poly'] gamma: [0.1, 1, 10, 100]
Decision Tree	max_depth: np.arange(2, 11) min_samples_split: np.arange(2, 11) min_samples_leaf: np.arange(1, 11) max_leaf_nodes: [5, 10, 15, 20]

ตาราง 3 (ต่อ)

แบบจำลอง	พารามิเตอร์
Random Forest	n_estimators: [100, 200, 300]
	max_depth: [2, 5, 7, 10]
	min_samples_split: [2, 5, 10]
	min_samples_leaf: [1, 2, 4]
XGBoost	n_estimators: [100, 200, 300, 400, 500]
	max_depth: np.arange(3, 11)
	learning_rate: [0.01, 0.1, 0.2, 0.3]

2. การประเมินประสิทธิภาพแบบจำลอง

ในงานวิจัยนี้ใช้เทคนิคการประเมินประสิทธิภาพแบบจำลอง โดยใช้ตาราง Confusion Matrix ได้แก่ Accuracy, Precision, Recall และ F1 Score แสดงตามภาพประกอบ 44



ภาพประกอบ 44 กรอบการทำงานของ confusion matrix

ที่มา: (Uddin et al., 2019)

True Positive (TP) คือ แบบจำลองทำนายว่าเป็น positive และความจริงเป็น positive

True Negative (TN) คือ แบบจำลองทำนายว่าเป็น negative และความจริงเป็น negative

False Positive (FP) คือ แบบจำลองทำนายว่าเป็น positive แต่ความจริงเป็น negative

False Negative (FN) คือ จำลองทำนายว่าเป็น negative แต่ความจริงเป็น positive

Accuracy หรือ ความแม่นยำ เป็นอัตราส่วนการจำแนกประเภทที่ถูกต้อง ต่อจำนวนข้อมูลที่นำมาจำแนกประเภททั้งหมด ดังสมการที่ 4

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad [4]$$

Precision หรือ ความเที่ยง เป็นอัตราส่วนของกลุ่ม Positive ที่จำแนกประเภทได้ถูกต้องต่อจำนวนกลุ่ม Positive ทั้งหมดที่จำแนกประเภทออกมา ดังสมการที่ 5

$$Precision = \frac{TP}{TP+FP} \quad [5]$$

Recall หรือ True Positive Rate (TPR) เป็นอัตราส่วนของกลุ่มที่เป็น Positive จริง ต่อจำนวนกลุ่ม Positive ทั้งหมด ดังสมการที่ 6

$$Recall = \frac{TP}{TP+FN} \quad [6]$$

F1 Score เป็นการรวมค่า Precision และ Recall เข้ามาเป็นค่าเดียว เพื่อใช้เป็นต้นแบบในการเปรียบเทียบประสิทธิภาพของแบบจำลอง ดังสมการที่ 7

$$F1\ Score = \frac{2 \cdot TP}{2 \cdot TP+FN+FP} \quad [7]$$

บทที่ 4
ผลการศึกษา

ในงานวิจัยเรื่องการทำนายการปรับตำแหน่งพนักงานโดยใช้การเรียนรู้ของเครื่อง ผู้วิจัยได้ดำเนินการวิจัยโดยศึกษาตามขั้นต่อนต่างๆ ตลอดจนถึงการประเมินประสิทธิภาพของแบบจำลอง เพื่อให้บรรลุสำเร็จจุดประสงค์ของการวิจัยที่กำหนดไว้ ดังนี้

1. ผลลัพธ์ของการวิเคราะห์ข้อมูล
2. ผลลัพธ์ของการวิเคราะห์ฟีเจอร์
3. ผลลัพธ์ของการปรับจูนพารามิเตอร์
4. ผลลัพธ์ของการสร้างแบบจำลอง

ผลลัพธ์ของการวิเคราะห์ข้อมูล

จากการสำรวจข้อมูลผลการปรับตำแหน่งจำนวน 212 คน มีผู้ที่ผ่านการปรับตำแหน่งจำนวน 183 คน และผู้ที่ไม่ผ่านการปรับตำแหน่งจำนวน 29 คน เมื่อทำการวิเคราะห์ลักษณะของพนักงานที่ผ่านการปรับตำแหน่ง และพนักงานที่ไม่ผ่านปรับตำแหน่ง มีผลลัพธ์ดังต่อไปนี้

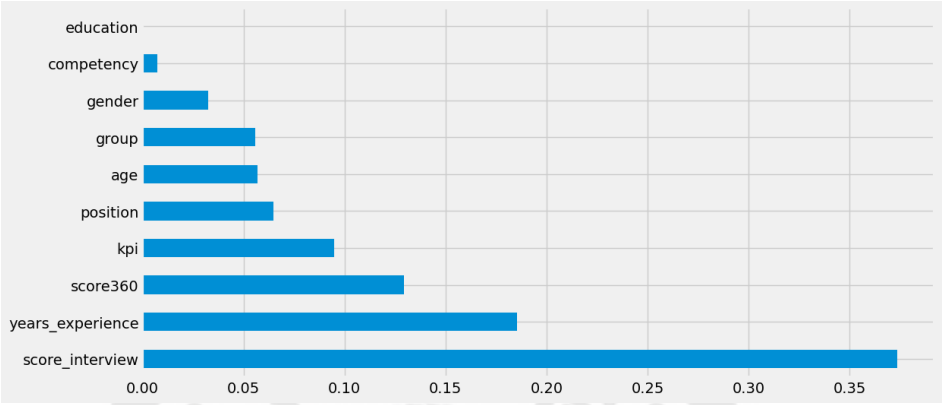
พนักงานที่ผ่านการปรับตำแหน่งจากกลุ่มข้อมูลพนักงานพบว่าพนักงานที่มีอายุ 22-38 ปี หรือกลุ่ม Gen Y และมีอายุงาน 2-10 ปี จะเป็นกลุ่มพนักงานที่ผ่านการปรับตำแหน่งมากที่สุด และจากกลุ่มข้อมูลคะแนนการประเมินผลงานและพฤติกรรมพบว่า พนักงานที่มีคะแนน 3-4 คะแนน จากคะแนนการปฏิบัติงานของปีก่อนหน้าและคะแนนการประเมินสมรรถนะและทักษะของปีก่อนหน้า และมีคะแนน 80-100 คะแนน จากคะแนนการประเมินรอบทิศ แบบ 360 องศา และคะแนนผลสัมภาษณ์จากคณะกรรมการ จะเป็นกลุ่มพนักงานที่ผ่านการปรับตำแหน่งมากที่สุด

พนักงานที่ไม่ผ่านปรับตำแหน่ง จากกลุ่มข้อมูลพนักงานพบว่าพนักงานที่มีระดับการศึกษาต่ำกว่าปริญญาตรี จะเป็นกลุ่มพนักงานที่ไม่ผ่านการปรับตำแหน่งมากที่สุด และจากกลุ่มข้อมูลคะแนนการประเมินผลงานและพฤติกรรมพบว่าพนักงานที่มีคะแนน 50-59 คะแนน จากคะแนนการประเมินรอบทิศ แบบ 360 องศา และมีคะแนน 0-69 คะแนน จากคะแนนผลสัมภาษณ์จากคณะกรรมการ จะเป็นกลุ่มพนักงานที่ไม่ผ่านการปรับตำแหน่งมากที่สุด

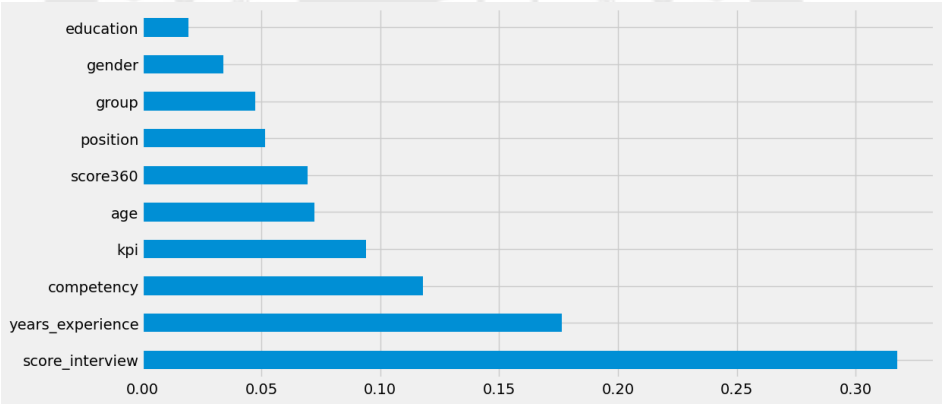
ผลลัพธ์ของการวิเคราะห์ฟีเจอร์

ก่อนการสร้างแบบจำลองการทำนายการปรับตำแหน่งพนักงาน ได้ทำการคัดเลือกฟีเจอร์ ผู้วิจัยจึงทำการหาค่า Feature Importance เพื่อหาคะแนนความสำคัญของฟีเจอร์เพิ่มเติม โดยใช้

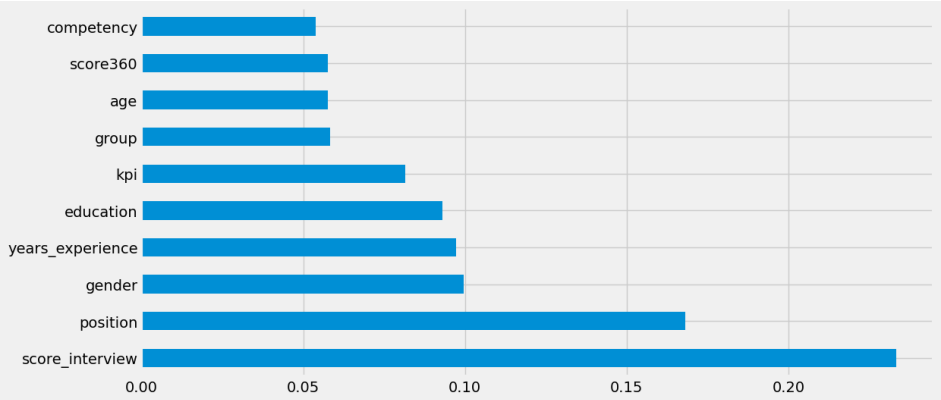
แบบจำลอง 3 จำลอง ได้แก่ Decision Tree, Random Forest และ XGBoost ดังภาพประกอบ 45-47 ซึ่งแบบจำลอง Support Vector Machine ไม่สามารถหา Feature Importance ได้ เนื่องจาก kernel ไม่ได้เป็น “linear”



ภาพประกอบ 45 คะแนนความสำคัญของฟีเจอร์ของแบบจำลอง Decision Tree



ภาพประกอบ 46 คะแนนความสำคัญของฟีเจอร์ของแบบจำลอง Random Forest



ภาพประกอบ 47 คะแนนความสำคัญของฟีเจอร์ของแบบจำลอง XGBoost

จากค่า Feature Importance ของแบบจำลอง พบว่า score interview หรือ ผลสัมภาษณ์จากคณะกรรมการ เป็นฟีเจอร์ที่สำคัญมากที่สุดของทุกแบบจำลอง โดยฟีเจอร์ในกลุ่มข้อมูลคะแนนการประเมินผลงานและพฤติกรรมมีความสำคัญกับแบบจำลอง Decision Tree และ Random Forest ซึ่งแตกต่างกับแบบจำลอง XGBoost ที่ข้อมูลในกลุ่มข้อมูลพนักงานมีความสำคัญมากกว่า

ผลลัพธ์ของการปรับจูนพารามิเตอร์

จากการสร้างแบบจำลองการทำนายการปรับตำแหน่งพนักงาน มีการปรับจูนพารามิเตอร์ (Hyperparameter Tuning) ของแต่ละแบบจำลองร่วมด้วย โดยมีรายละเอียดของผลลัพธ์พารามิเตอร์ของแต่ละแบบจำลอง ดังตาราง 4

ตาราง 4 พารามิเตอร์หลังการปรับจูนของแต่ละแบบจำลอง

แบบจำลอง	อัตราส่วนข้อมูล	พารามิเตอร์
Support Vector Machine without SMOTE and Feature Selection	70:30	C: 10, gamma: 0.1, kernel: 'rbf'
	80:20	C: 10, gamma: 0.1, kernel: 'rbf'

ตาราง 4 (ต่อ)

แบบจำลอง	อัตราส่วนข้อมูล	พารามิเตอร์
Support Vector Machine with SMOTE	70:30	C: 1, gamma: 1, kernel: 'rbf'
	80:20	C: 1, gamma: 1, kernel: 'rbf'
Support Vector Machine with Feature Selection	70:30	C: 10, gamma: 0.1, kernel: 'rbf'
	80:20	C: 10, gamma: 0.1, kernel: 'rbf'
Support Vector Machine with SMOTE and Feature Selection	70:30	C: 100, gamma: 0.1, kernel: 'rbf'
	80:20	C: 10, gamma: 1, kernel: 'rbf'
Decision Tree without SMOTE and Feature Selection	70:30	max_depth: 7, max_leaf_nodes: 10, min_samples_leaf:1, min_samples_split: 6
	80:20	max_depth: 4, max_leaf_nodes: 10, min_samples_leaf': 8, min_samples_split: 6
Decision Tree with SMOTE	70:30	max_depth: 6, max_leaf_nodes: 15, min_samples_leaf: 1, min_samples_split: 4
	80:20	max_depth: 8, max_leaf_nodes: 20, min_samples_leaf: 2, min_samples_split: 5
Decision Tree with Feature Selection	70:30	max_depth: 8, max_leaf_nodes: 10, min_samples_leaf: 1, min_samples_split: 6
	80:20	max_depth: 4, max_leaf_nodes: 10, min_samples_leaf: 8, min_samples_split: 2
Decision Tree with SMOTE and Feature Selection	70:30	max_depth: 9, max_leaf_nodes: 10, min_samples_leaf: 1, min_samples_split: 2
	80:20	max_depth: 9, max_leaf_nodes: 20, min_samples_leaf: 1, min_samples_split: 6

ตาราง 4 (ต่อ)

แบบจำลอง	อัตราส่วนข้อมูล	พารามิเตอร์
Random Forest without SMOTE and Feature Selection	70:30	max_depth: 7, min_samples_leaf: 2, min_samples_split: 10, n_estimators: 200
	80:20	max_depth: 4, max_leaf_nodes: 10, min_samples_leaf: 8, min_samples_split: 2
Random Forest with SMOTE	70:30	max_depth: 10, min_samples_leaf: 1, min_samples_split: 5, n_estimators: 100
	80:20	max_depth: 7, min_samples_leaf: 1, min_samples_split: 2, n_estimators: 200
Random Forest with Feature Selection	70:30	max_depth: 7, min_samples_leaf: 1, min_samples_split: 5, n_estimators: 300
	80:20	max_depth: 5, min_samples_leaf: 2, min_samples_split: 10, n_estimators: 300
Random Forest with SMOTE and Feature Selection	70:30	max_depth: 7, min_samples_leaf: 1, min_samples_split: 5, n_estimators: 100
	80:20	max_depth: 10, min_samples_leaf: 1, min_samples_split: 2, n_estimators: 200
XGBoost without SMOTE and Feature Selection	70:30	learning_rate: 0.01, max_depth: 4, n_estimators: 300
	80:20	learning_rate: 0.1, max_depth: 3, n_estimators: 300
XGBoost with SMOTE	70:30	learning_rate: 0.3, max_depth: 5, n_estimators: 400
	80:20	learning_rate: 0.1, max_depth: 7, n_estimators: 200

ตาราง 4 (ต่อ)

แบบจำลอง	อัตราส่วนข้อมูล	พารามิเตอร์
XGBoost with Feature Selection	70:30	learning_rate: 0.01, max_depth: 3, n_estimators: 500
	80:20	learning_rate: 0.3, max_depth: 5, n_estimators: 500
XGBoost with SMOTE and Feature Selection	70:30	learning_rate: 0.01, max_depth: 7, n_estimators: 500
	80:20	learning_rate': 0.1, max_depth: 5, n_estimators: 300

ผลลัพธ์ของการสร้างแบบจำลอง

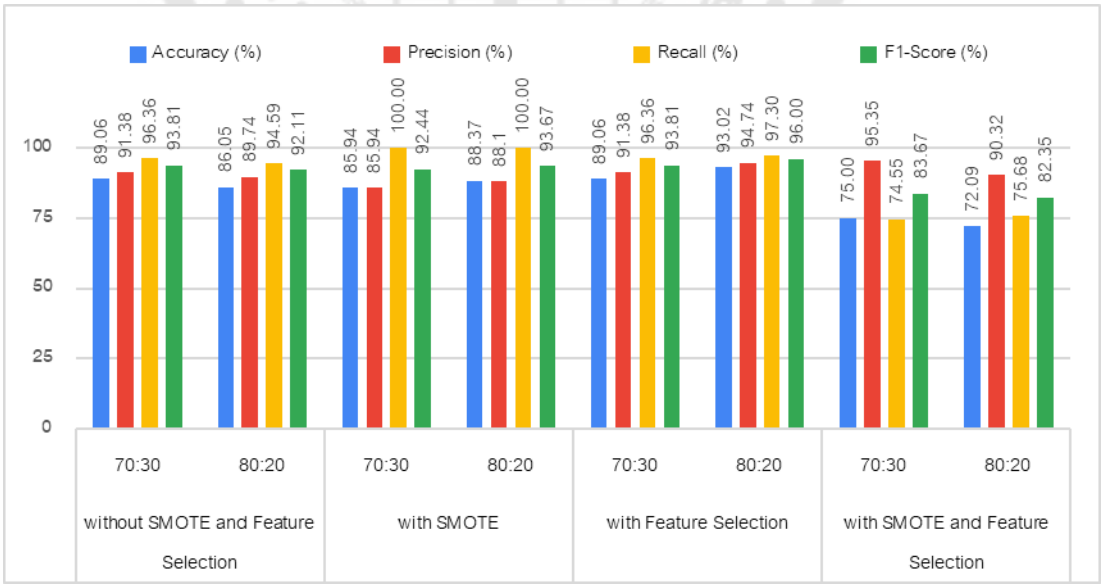
จากการสร้างแบบจำลองการทำนายการปรับตำแหน่งพนักงาน จะวัดประสิทธิภาพจากการทดสอบแบบจำลองโดยใช้ชุดข้อมูลสำหรับทดสอบแบบจำลอง (Test Set) โดยแสดงเป็นตารางค่า Accuracy, Precision, Recall และ F1 Score ของแต่ละแบบจำลอง ดังตาราง 5-8 และภาพประกอบ 48-51

1. Support Vector Machine

เมื่อการทำคัดเลือกฟีเจอร์ก่อนการสร้างแบบจำลอง พบว่าจำนวนฟีเจอร์ที่ทำให้แบบจำลองมีประสิทธิภาพดีที่สุด คือ 5 ฟีเจอร์ ได้แก่ score_interview, years_experience, kpi, group และ score360 โดยแสดงประสิทธิภาพการทดสอบจากการใช้เทคนิคต่างๆ ร่วมด้วย ดังตาราง 5 และภาพประกอบ 48

ตาราง 5 แสดงประสิทธิภาพการทดสอบจากแบบจำลอง Support Vector Machine

แบบจำลอง	อัตราส่วนข้อมูล	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Support Vector Machine without SMOTE and Feature Selection	70:30	89.06	91.38	96.36	93.81
	80:20	86.05	89.74	94.59	92.11
Support Vector Machine with SMOTE	70:30	85.94	85.94	100	92.44
	80:20	88.37	88.10	100	93.67
Support Vector Machine with Feature Selection	70:30	89.06	91.38	96.36	93.81
	80:20	93.02	94.74	97.30	96.00
Support Vector Machine with SMOTE and Feature Selection	70:30	75.00	95.35	74.55	83.67
	80:20	72.09	90.32	75.68	82.35



ภาพประกอบ 48 กราฟแท่งแสดงประสิทธิภาพการทดสอบจากแบบจำลอง Support Vector Machine

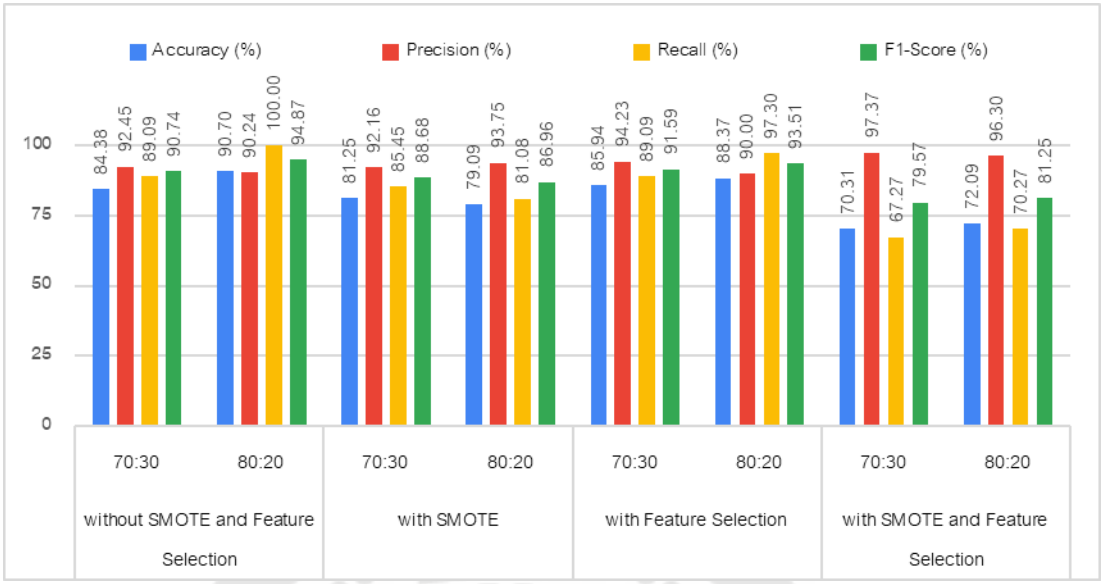
จากตาราง 5 และภาพประกอบ 48 พบว่าประสิทธิภาพของแบบจำลอง Support Vector Machine ที่แบ่งชุดข้อมูล 80:20 ร่วมกับการทำ Feature Selection ให้ประสิทธิภาพที่ดีที่สุดเมื่อวัดด้วยค่า Accuracy, Precision, Recall และ F1-Score รองลงมา คือ แบบจำลอง Support Vector Machine ที่แบ่งชุดข้อมูล 70:30 ร่วมกับการทำ Feature Selection และแบบจำลอง Support Vector Machine ที่ไม่ได้ใช้เทคนิคใดร่วมด้วย แบ่งชุดข้อมูล 70:30 มีค่า Accuracy, Precision, Recall และ F1-Score เท่ากัน

2. Decision Tree

เมื่อการทำคัดเลือกฟีเจอร์ก่อนการสร้างแบบจำลอง พบว่าจำนวนฟีเจอร์ที่ทำให้แบบจำลองมีประสิทธิภาพดีที่สุด คือ 5 ฟีเจอร์ ได้แก่ score_interview, years_experience, kpi, position และ score360 โดยแสดงประสิทธิภาพการทดสอบจากการใช้เทคนิคต่างๆ ร่วมด้วย ดังตาราง 6 และภาพประกอบ 49

ตาราง 6 แสดงประสิทธิภาพการทดสอบจากแบบจำลอง Decision Tree

แบบจำลอง	**อัตราส่วนข้อมูล**	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Decision Tree without SMOTE and Feature Selection	70:30	84.38	92.45	89.09	90.74
	80:20	90.70	90.24	100	94.87
Decision Tree with SMOTE	70:30	81.25	92.16	85.45	88.68
	80:20	79.09	93.75	81.08	86.96
Decision Tree with Feature Selection	70:30	85.94	94.23	89.09	91.59
	80:20	88.37	90.00	97.30	93.51
Decision Tree with SMOTE and Feature Selection	70:30	70.31	97.37	67.27	79.57
	80:20	72.09	96.30	70.27	81.25



ภาพประกอบ 49 กราฟแท่งแสดงประสิทธิภาพการทดสอบจากแบบจำลอง Decision Tree

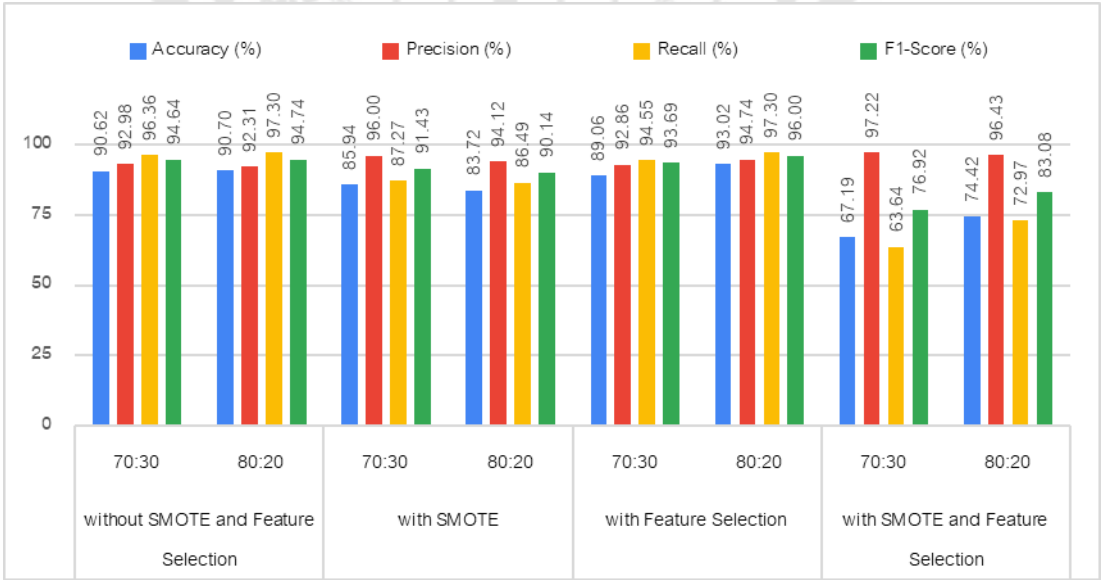
จากตาราง 6 และภาพประกอบ 49 พบว่าประสิทธิภาพของแบบจำลอง Decision Tree ที่ไม่ได้ใช้เทคนิคใดร่วมด้วย ที่แบ่งชุดข้อมูล 80:20 ให้ประสิทธิภาพที่ดีที่สุดเมื่อวัดด้วยค่า Accuracy, Precision, Recall และ F1-Score รองลงมา คือ แบบจำลอง Decision Tree ที่แบ่งชุดข้อมูล 80:20 และ 70:30 ร่วมกับการทำ Feature Selection มีค่า Accuracy, Precision, Recall และ F1-Score รองลงมาตามลำดับ

3. Random Forest

เมื่อการทำคัดเลือกฟีเจอร์ก่อนการสร้างแบบจำลอง พบว่าจำนวนฟีเจอร์ที่ทำให้แบบจำลองมีประสิทธิภาพดีที่สุด คือ 5 ฟีเจอร์ ได้แก่ score_interview, years_experience, competency, age และ kpi โดยแสดงประสิทธิภาพการทดสอบจากการใช้เทคนิคต่างๆ ร่วมด้วยดังตาราง 7 และภาพประกอบ 50

ตาราง 7 แสดงประสิทธิภาพการทดสอบจากแบบจำลอง Random Forest

แบบจำลอง	อัตราส่วนข้อมูล	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Random Forest without SMOTE and Feature Selection	70:30	90.62	92.98	96.36	94.64
	80:20	90.70	92.31	97.30	94.74
Random Forest with SMOTE	70:30	85.94	96.00	87.27	91.43
	80:20	83.72	94.12	86.49	90.14
Random Forest with Feature Selection	70:30	89.06	92.86	94.55	93.69
	80:20	93.02	94.74	97.30	96.00
Random Forest with SMOTE and Feature Selection	70:30	67.19	97.22	63.64	76.92
	80:20	74.42	96.43	72.97	83.08



ภาพประกอบ 50 กราฟแท่งแสดงแสดงประสิทธิภาพการทดสอบจากแบบจำลอง Random Forest

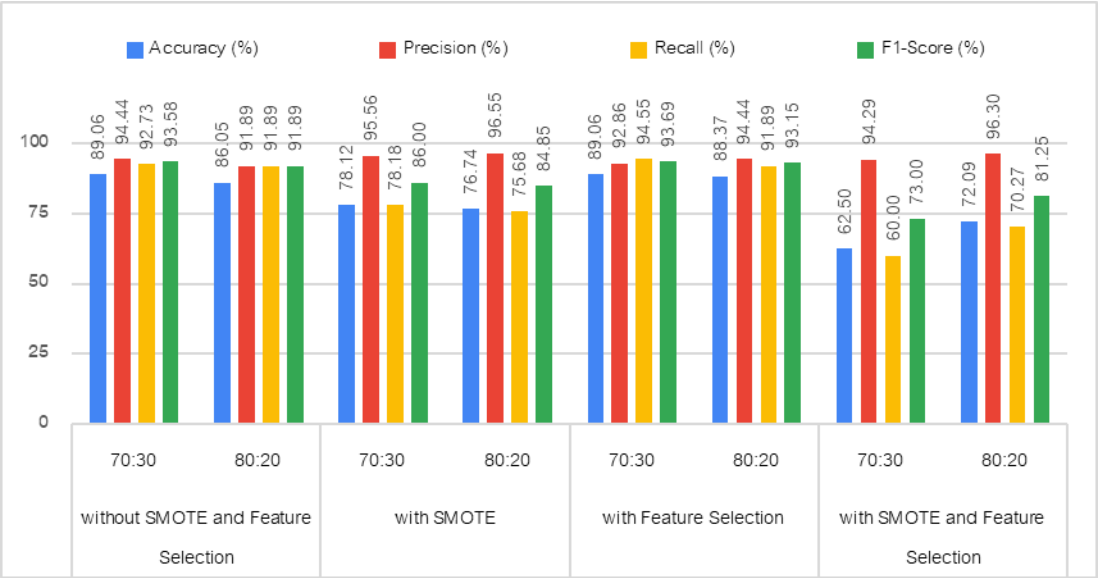
จากตาราง 7 และภาพประกอบ 50 พบว่าประสิทธิภาพของแบบจำลอง Random Forest แบ่งชุดข้อมูล 80:20 ร่วมกับการทำ Feature Selection ให้ประสิทธิภาพที่ดีที่สุด เมื่อวัดด้วยค่า Accuracy, Precision, Recall และ F1-Score รองลงมา คือ แบบจำลอง Random Forest ที่แบ่งชุดข้อมูล 80:20 และ 70:30 ที่ไม่ได้ใช้เทคนิคใดร่วมด้วย มีค่า Accuracy, Precision, Recall และ F1-Score รองลงมาตามลำดับ

4. XGBoost

เมื่อการทำคัดเลือกฟีเจอร์ก่อนการสร้างแบบจำลอง พบว่าจำนวนฟีเจอร์ที่ทำให้แบบจำลองมีประสิทธิภาพดีที่สุด คือ 3 ฟีเจอร์ ได้แก่ score_interview, gender และ position โดยแสดงประสิทธิภาพการทดสอบจากการใช้เทคนิคต่างๆ ร่วมด้วย ดังตาราง 8 และภาพประกอบ 51

ตาราง 8 แสดงประสิทธิภาพการทดสอบจากแบบจำลอง XGBoost

แบบจำลอง	**อัตราส่วนข้อมูล**	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
XGBoost without SMOTE and Feature Selection	70:30	89.06	94.44	92.73	93.58
	80:20	86.05	91.89	91.89	91.89
XGBoost with SMOTE	70:30	78.12	95.56	78.18	86.00
	80:20	76.74	96.55	75.68	84.85
XGBoost with Feature Selection	70:30	89.06	92.86	94.55	93.69
	80:20	88.37	94.44	91.89	93.15
XGBoost with SMOTE and Feature Selection	70:30	62.50	94.29	60.00	73.00
	80:20	72.09	96.30	70.27	81.25



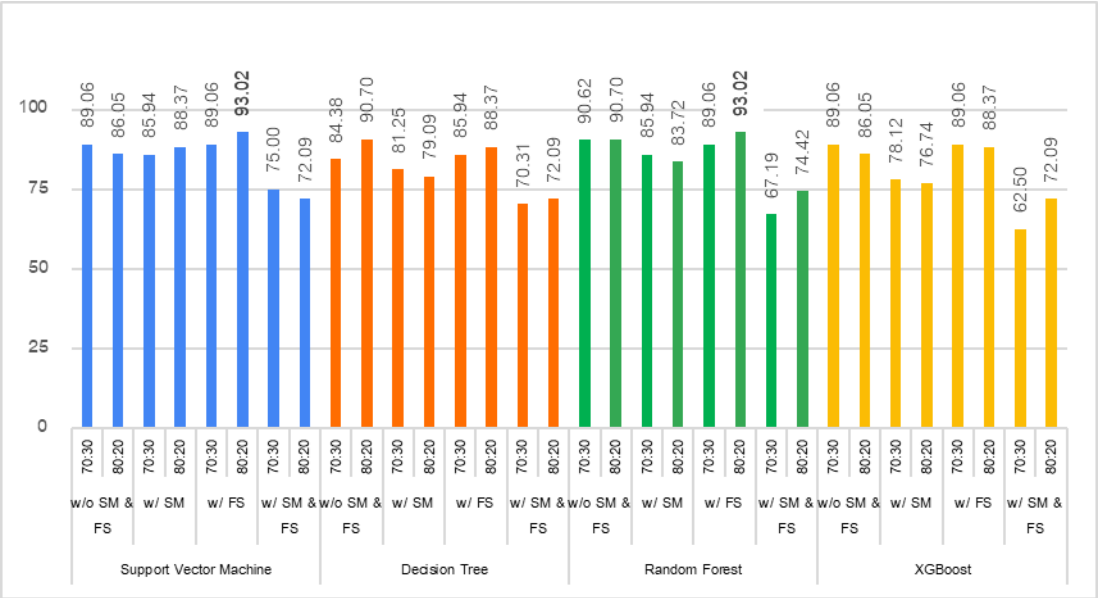
ภาพประกอบ 51 กราฟแท่งแสดงประสิทธิภาพการทดสอบจากแบบจำลอง XGBoost

จากตาราง 8 และภาพประกอบ 51 พบว่าประสิทธิภาพของแบบจำลอง XGBoost ที่แบ่งชุดข้อมูล 70:30 ร่วมกับการทำ Feature Selection ให้ประสิทธิภาพที่ดีที่สุดเมื่อวัดด้วยค่า Accuracy, Precision, Recall และ F1-Score รองลงมา คือ แบบจำลอง XGBoost ที่แบ่งชุดข้อมูล 70:30 ที่ไม่ได้ใช้เทคนิคใดร่วมด้วย และแบบจำลอง XGBoost ที่แบ่งชุดข้อมูล 80:20 ร่วมกับการทำ Feature Selection มีค่า Accuracy, Precision, Recall และ F1-Score ตามลำดับ

ผลลัพธ์ของการเปรียบเทียบแบบจำลอง โดยแสดงเป็นตารางค่าความแม่นยำ หรือ Accuracy ของแต่ละแบบจำลองที่มีการใช้เทคนิคต่างๆ ร่วมด้วย ได้แก่ ไม่ได้ใช้เทคนิคใดร่วมด้วย (w/o SM & FS), ใช้เทคนิค SMOTE ร่วมด้วย (w/ SM), ใช้เทคนิค Feature Selection ร่วมด้วย (w/ FS) และใช้เทคนิค SMOTE และ Feature Selection ร่วมด้วย (w/ SM & FS) ดังตาราง 9 และภาพประกอบ 52

ตาราง 9 แสดงการเปรียบเทียบความแม่นยำจากการทดสอบแบบจำลอง

แบบจำลอง		without SMOTE	SMOTE
	อัตราส่วนข้อมูล	Accuracy (%)	Accuracy (%)
Support Vector Machines without Feature Selection	70:30	89.06	85.94
	80:20	86.05	88.37
Support Vector Machines with Feature Selection	70:30	89.06	75.00
	80:20	93.02	72.09
Decision Tree without Feature Selection	70:30	84.38	81.25
	80:20	90.70	79.07
Decision Tree with Feature Selection	70:30	85.94	70.31
	80:20	88.37	72.09
Random Forest without Feature Selection	70:30	90.62	85.94
	80:20	90.70	83.72
Random Forest with Feature Selection	70:30	89.06	67.19
	80:20	93.02	74.42
XGBoost without Feature Selection	70:30	89.06	78.12
	80:20	86.05	76.74
XGBoost with Feature Selection	70:30	89.06	63.50
	80:20	88.37	72.09



ภาพประกอบ 52 กราฟแท่งแสดงการเปรียบเทียบความแม่นยำจากการทดสอบแบบจำลอง

จากตาราง 9 และภาพประกอบ 52 ทำการเปรียบเทียบค่า Accuracy ของทุกแบบจำลอง พบว่าแบบจำลอง Support Vector Machines ที่แบ่งข้อมูล 80:20 ร่วมกับการทำ Feature Selection และแบบจำลอง Random Forest ที่แบ่งข้อมูล 80:20 ร่วมกับการทำ Feature Selection ได้ประสิทธิภาพสูงสุด มีค่า Accuracy เท่ากัน คือ 93.02%

บทที่ 5
สรุป อภิปรายผล และข้อเสนอแนะ

ในงานวิจัยเรื่องการทำนายการปรับตำแหน่งด้วยการเรียนรู้ของเครื่อง ผู้วิจัยได้วัดประสิทธิภาพของแบบจำลองในแต่ละอัลกอริธึม เพื่อนำมาเปรียบเทียบและสรุปผล โดยสามารถแบ่งหัวข้อการสรุปผลได้ ดังต่อไปนี้

1. สรุปผลการวิจัย
2. อภิปรายผลการวิจัย
3. ข้อเสนอแนะ

สรุปผลการวิจัย

งานวิจัยนี้ศึกษาการทำนายผลการปรับตำแหน่งงาน โดยใช้ข้อมูลการปรับตำแหน่งพนักงาน ของฝ่ายบริหารทรัพยากรบุคคล บริษัท ไทยสมุทรประกันชีวิต จำกัด (มหาชน) มาใช้ในการสร้างแบบจำลองทั้งหมด 4 แบบจำลอง ได้แก่ Support Vector Machine, Decision Tree, Random Forest และ XGBoost ร่วมกับการคัดเลือกฟีเจอร์ด้วยเทคนิค RFE และการจัดการความไม่สมดุลของข้อมูลด้วยเทคนิค SMOTE โดยในการสร้างแบบจำลองได้ทำการปรับค่าพารามิเตอร์ด้วยเทคนิค GridSearchCV ร่วมด้วย จากตาราง 9 และภาพประกอบ 52 พบว่าแบบจำลองที่ได้ประสิทธิภาพสูงสุด คือ แบบจำลอง Support Vector Machines และ แบบจำลอง Random Forest ที่แบ่งข้อมูล 80:20 ร่วมกับการคัดเลือกฟีเจอร์ ได้ค่า Accuracy 93.02%, Precision 94.74%, Recall 97.30% และ F1-Score 96.00% เท่ากัน ดังนั้น แบบจำลองที่ทำการคัดเลือกฟีเจอร์ก่อนการสร้างแบบจำลองเป็นแบบจำลองที่มีประสิทธิภาพสูงสุด รองลงมา คือ แบบจำลองที่ไม่ได้ใช้เทคนิคร่วมด้วย และแบบจำลองที่ใช้เทคนิค SMOTE ตามลำดับ ซึ่งแบบจำลองที่ใช้เทคนิคการคัดเลือกฟีเจอร์ ร่วมกับเทคนิคการจัดการความไม่สมดุลของข้อมูล (SMOTE) เป็นแบบจำลองที่มีประสิทธิภาพต่ำที่สุดในการทำนายผลการปรับตำแหน่ง เมื่อเปรียบเทียบประสิทธิภาพของแบบจำลองทั้งหมด

อภิปรายผลการวิจัย

จากการสร้างแบบจำลองการทำนายผลการปรับตำแหน่งงานที่ทดสอบด้วยชุดข้อมูลสำหรับทดสอบแบบจำลอง โดยใช้แบบจำลองทั้งหมด 4 แบบจำลอง ได้แก่ Support Vector Machine, Decision Tree, Random Forest และ XGBoost แบบจำลองที่มีประสิทธิภาพสูงสุด

คือ แบบจำลอง Support Vector Machines และ Random Forest ที่แบ่งข้อมูล 80:20 ร่วมกับการคัดเลือกฟีเจอร์ให้ประสิทธิภาพสูงสุดเท่ากัน คือ Accuracy 93.02%, Precision 94.74%, Recall 97.30% และ F1-Score 96.00%

เมื่อเปรียบเทียบประสิทธิภาพของแบบจำลองที่ใช้เทคนิคการคัดเลือกฟีเจอร์ ร่วมกับเทคนิคการจัดการความไม่สมดุลของข้อมูล (SMOTE) กับประสิทธิภาพของแบบจำลองที่ไม่ใช้เทคนิคใดร่วมด้วย ผลลัพธ์ คือ แบบจำลองที่ใช้เทคนิคการคัดเลือกฟีเจอร์ ร่วมกับเทคนิค SMOTE ที่มีประสิทธิภาพสูงสุด คือ แบบจำลอง Support Vector Machines ที่แบ่งข้อมูล 70:30 ได้ค่า Accuracy 75.00, Precision 95.35%, Recall 74.55% และ F1-Score 84.67% และแบบจำลองที่ไม่ใช้เทคนิคใดร่วมด้วยที่มีประสิทธิภาพสูงสุด คือ แบบจำลอง Random Forest ที่แบ่งข้อมูล 80:20 ได้ค่า Accuracy 90.70, Precision 92.31, Recall 97.30% และ F1-Score 94.74% ดังนั้น แบบจำลองที่ใช้เทคนิคการคัดเลือกฟีเจอร์ ร่วมกับเทคนิคการจัดการความไม่สมดุลของข้อมูล (SMOTE) ได้ประสิทธิภาพที่ต่ำกว่าการไม่ใช้เทคนิคใดร่วมด้วยในทุกแบบจำลอง และได้ประสิทธิภาพต่ำที่สุดเมื่อเปรียบเทียบประสิทธิภาพของแบบจำลองที่ใช้เทคนิคอื่นๆ ร่วมด้วย

สำหรับการเปรียบเทียบประสิทธิภาพของแบบจำลองที่ทำการคัดเลือกฟีเจอร์ก่อนการสร้างแบบจำลองกับประสิทธิภาพของแบบจำลองที่ใช้ฟีเจอร์ทั้งหมด ผลลัพธ์ คือ แบบจำลองที่ทำการคัดเลือกฟีเจอร์ที่มีประสิทธิภาพสูงสุด คือ แบบจำลอง Support Vector Machines และ Random Forest ที่แบ่งข้อมูล 80:20 มีประสิทธิภาพสูงสุดเท่ากัน คือ Accuracy 93.02%, Precision 94.74%, Recall 97.30% และ F1-Score 96.00% ซึ่งเป็นแบบจำลองที่มีประสิทธิภาพสูงสุดในการทดสอบ และแบบจำลองที่ใช้ฟีเจอร์ทั้งหมดที่มีประสิทธิภาพสูงสุด คือ Decision Tree ที่แบ่งข้อมูล 80:20 ได้ค่า Accuracy 90.70%, Precision 90.24 %, Recall 100% และ F1-Score 94.87% ดังนั้น การคัดเลือกฟีเจอร์ก่อนการสร้างแบบจำลองให้ประสิทธิภาพที่ดีกว่าการใช้ฟีเจอร์ทั้งหมดในการสร้างแบบจำลอง ยกเว้นแบบจำลอง Decision Tree ที่แบ่งข้อมูล 80:20 เพียงแบบจำลองเดียวที่ใช้ฟีเจอร์ทั้งหมดแต่ได้ประสิทธิภาพดีกว่าแบบจำลอง Decision Tree ที่แบ่งข้อมูล 80:20 ร่วมกับการคัดเลือกฟีเจอร์

แบบจำลองที่ทำการจัดการความสมดุลของข้อมูลด้วยเทคนิค SMOTE เมื่อเปรียบเทียบประสิทธิภาพของการใช้เทคนิค SMOTE และ ไม่ใช้เทคนิค SMOTE ผลลัพธ์ คือ แบบจำลองที่ใช้เทคนิค SMOTE ที่มีประสิทธิภาพสูงสุด คือ แบบจำลอง Support Vector Machines ที่แบ่งข้อมูล 80:20 ได้ค่า Accuracy 88.37%, Precision 88.10%, Recall 100% และ F1-Score 93.67% ซึ่งมีประสิทธิภาพมากกว่าเมื่อเทียบแบบจำลอง Support Vector Machines ที่แบ่งข้อมูล 80:20 ที่

ไม่ใช้เทคนิค SMOTE แต่เมื่อเปรียบเทียบประสิทธิภาพของทุกแบบจำลองผลลัพธ์ คือ แบบจำลองที่ใช้การจัดการความสมดุลของข้อมูลด้วยเทคนิค SMOTE ได้ประสิทธิภาพที่ต่ำกว่าการไม่ใช้เทคนิค SMOTE ร่วมด้วยในทุกแบบจำลอง โดยเฉพาะแบบจำลองที่มีความซับซ้อนสูงอย่าง XGBoost

จากการแบ่งข้อมูลในอัตราส่วนที่แตกต่างกัน 2 อัตราส่วน คือ 80:20 และ 70:30 เมื่อเปรียบเทียบประสิทธิภาพของแบบจำลอง ผลลัพธ์ คือ แบบจำลองที่มีประสิทธิภาพสูงสุด 9 แบบจำลอง เป็นแบบจำลองที่มาจากการแบ่งอัตราส่วนข้อมูล 80:20 และแบบจำลองที่มีประสิทธิภาพสูงสุด 7 แบบจำลอง เป็นแบบจำลองที่มาจากการแบ่งอัตราส่วนข้อมูล 70:30 สรุป คือ การแบ่งข้อมูลในอัตราส่วน 80:20 ทำให้ประสิทธิภาพของแบบจำลองที่ดีกว่าการแบ่งข้อมูลด้วยอัตราส่วน 70:30

พิจารณาฟีเจอร์ที่สำคัญของแบบจำลองแต่ละแบบ ผลลัพธ์ คือ ฟีเจอร์ score_interview เป็นฟีเจอร์ที่ได้อันดับฟีเจอร์ที่มีความสำคัญมากที่สุด และได้ค่าคะแนนฟีเจอร์มากที่สุดจากทุกแบบจำลอง ซึ่งอยู่ในกลุ่มข้อมูลคะแนนการประเมินผลงานและพฤติกรรม รองลงมา คือ ฟีเจอร์ years_experience ซึ่งอยู่ในกลุ่มข้อมูลพนักงาน โดยเมื่อทำการคัดเลือกฟีเจอร์จะได้ 3 ฟีเจอร์ ที่อยู่ในกลุ่มข้อมูลคะแนนการประเมินผลงานและพฤติกรรม ที่นำมาใช้ในการสร้างแบบจำลอง ยกเว้นแบบจำลอง XGBoost จะใช้ฟีเจอร์ที่อยู่กลุ่มข้อมูลพนักงานมากกว่าฟีเจอร์ที่อยู่ในกลุ่มข้อมูลคะแนนการประเมินผลงานและพฤติกรรม ทำให้ฟีเจอร์ที่อยู่ในกลุ่มข้อมูลคะแนนการประเมินผลงานและพฤติกรรม เป็นฟีเจอร์ที่มีความสำคัญมากกว่าฟีเจอร์ที่อยู่กลุ่มข้อมูลพนักงาน

ข้อเสนอแนะ

1. ข้อมูลที่นำมาใช้ในงานวิจัยนี้ มีจำนวนค่อนข้างน้อย และมีบางรายการที่ไม่มีข้อมูลของฟีเจอร์สำคัญ คือ ฟีเจอร์ในกลุ่มของข้อมูลคะแนนการประเมินผลงานและพฤติกรรม ทำให้ต้องแทนค่าด้วย 0 ดังนั้น ในอนาคตมีจำนวนข้อมูลที่เพิ่มขึ้น และครบถ้วน อาจช่วยเพิ่มประสิทธิภาพของการทำนายการปรับตำแหน่งงานให้ดียิ่งขึ้น

2. ขั้นตอนการเตรียมข้อมูล เปลี่ยนการแปลงข้อมูลเป็นตัวเลขโดยใช้เทคนิค One Hot Encoder ซึ่งเป็นเทคนิคที่เหมาะกับข้อมูลหมวดหมู่หรือประเภทที่ไม่สามารถเรียงลำดับได้อย่างชัดเจน เพื่อให้เหมาะกับข้อมูลที่ใช้ในงานวิจัยมากยิ่งขึ้น

3. จัดการความไม่สมดุลของข้อมูลโดยใช้เทคนิค Random Over-sampling เพิ่มเติม อาจช่วยเพิ่มประสิทธิภาพของการทำนายการปรับตำแหน่งงานให้ดียิ่งขึ้น

4. เนื่องจากงานวิจัยนี้ได้สร้างและทดสอบแบบจำลอง ทั้งหมด 4 แบบจำลอง ได้แก่ Support Vector Machine, Decision Tree, Random Forest และ XGBoost ดังนั้น อาจมีแบบจำลองอื่นที่สามารถเรียนรู้และทำนายออกมาได้มีความแม่นยำและมีประสิทธิภาพที่ดีกว่า เช่น AdaBoost, Gradient Boosting

5. ในอนาคตอาจมีการทดลองใช้แบบจำลองร่วมกับโปรแกรมจัดการทรัพยากรบุคคลเพื่อความสะดวกและรวดเร็วในการวิเคราะห์ ทำนายผลการปรับตำแหน่งงานล่วงหน้า

บรรณานุกรม

Alqahtani, F. A., & Almaleh, A. (2022). Analysis and Prediction of Employee Promotions Using Machine Learning. 2022 5th International Conference on Data Science and Information Technology (DSIT),

Emmanuel-Okereke, I. L., & Anigbogu, S. O. (2022). KNN and SVM Machine learning to Predict Staff Due for Promotions and Training.

Kaewwiset, T., & Temdee, P. (2022). Promotion Classification Using DecisionTree and Principal Component Analysis. 2022 Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering (ECTI DAMT & NCON),

Kaewwiset, T., Temdee, P., & Yooyativong, T. (2021). Employee classification for personalized professional training using machine learning techniques and SMOTE. 2021 Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunication Engineering,

Long, Y., Liu, J., Fang, M., Wang, T., & Jiang, W. (2018). Prediction of employee promotion based on personal basic features and post features. Proceedings of the International Conference on Data Processing and Applications,

Mahesh, B. (2020). Machine learning algorithms-a review. *International Journal of Science and Research (IJSR).[Internet]*, *9*, 381-386.

Miao, J., & Niu, L. (2016). A survey on feature selection. *Procedia Computer Science*, *91*, 919-926.

Mohammed, M., Khan, M. B., & Bashier, E. B. M. (2016). *Machine learning: algorithms and applications*. Crc Press.

Owen, L. (2022). *Hyperparameter Tuning with Python: Boost your machine learning model's performance via hyperparameter tuning*. Packt Publishing. https://books.google.co.th/books?id=CqF-EAAAQBAJ

Ramyachitra, D., & Manikandan, P. (2014). Imbalanced dataset classification and solutions: a review. *International Journal of Computing and Business Research (IJCBR)*, *5*(4), 1-29.

Şahinbaş, K. (2022). Employee Promotion Prediction by using Machine Learning Algorithms for Imbalanced Dataset. 2022 2nd International Conference on Computing and Machine Intelligence (ICMI),

Singh, A., Thakur, N., & Sharma, A. (2016). A review of supervised machine learning algorithms. 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom),

TITIPATA. (2018). *[ML] Bagging หรือ Boosting คืออะไร ทำงานอย่างไร?* https://tupleblog.github.io/bagging-boosting/

Uddin, S., Khan, A., Hossain, M. E., & Moni, M. A. (2019). Comparing different supervised machine learning algorithms for disease prediction. *BMC medical informatics and decision making*, *19*(1), 1-16.

Zheng, A., & Casari, A. (2018). *Feature engineering for machine learning: principles and techniques for data scientists*. " O'Reilly Media, Inc.".

ประวัติผู้เขียน