



การพยากรณ์การเลื่อนตำแหน่งของพนักงานภายในองค์กร

จัดทำโดย

นางสาวชนิสรา	ทิพย์พันธุ์	64 - 040226 - 1615 - 1
นางสาวธนัชพร	นุชน้อย	64 - 040226 - 3014 - 6
นางสาวณัชญา	กตกุลโสภณ	64 - 040226 - 3043 - 0

เสนอ

ผู้ช่วยศาสตราจารย์ ดร. ศักดิ์ชาย ตั้งประเสริฐ

รายงานฉบับนี้เป็นส่วนหนึ่งของการศึกษาวิชา วิชา BUSINESS INTELLIGENCE (040223406)

ภาคเรียนที่ 1 ปีการศึกษา 2566

มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ

กิตติกรรมประกาศ

รายงานฯ เรื่อง “การพยากรณ์การเลื่อนตำแหน่งของพนักงานภายในองค์กร” นี้ สำเร็จได้ด้วยความเมตตาจากผู้ช่วยศาสตราจารย์ ดร. ศักดิ์ชาย ตังประเสริฐ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ อาจารย์ผู้สอนรายวิชา 040223406 BUSINESS INTELLIGENCE ที่ให้ความกรุณาเป็นอาจารย์ที่ปรึกษา ประสิทธิ์ประสาทความรู้และทักษะด้านการทำ Cross Industry Standard Process for Data Mining (CRISP – DM) Process อีกทั้งได้สละเวลาให้คำแนะนำ ข้อเสนอแนะ ตรวจสอบและแก้ไขข้อบกพร่องต่าง ๆ ด้วยความเอาใจใส่อย่างดียิ่ง ซึ่งทางคณะผู้จัดทำรู้สึกซาบซึ้งในพระคุณ จึงขอกราบขอบพระคุณเป็นอย่างสูงไว้ ณ โอกาสนี้

ขอขอบพระคุณสำนักหอสมุดกลาง มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ ที่ช่วยเหลืออำนวยความสะดวกในการให้สถานที่นั่งทำงานอินเทอร์เน็ตใช้งาน

คุณค่าและประโยชน์ของรายงานเรื่องนี้ทางคณะผู้จัดทำขอมอบให้แก่คณะอาจารย์ที่มีส่วนร่วมในการวางรากฐานทางการศึกษา และประสิทธิ์ประสาทวิชาความรู้แก่คณะผู้จัดทำตลอดมา

คณะผู้จัดทำ

04 กุมภาพันธ์ 2566

คำนำ

รายงานฯ เรื่อง “การพยากรณ์การเลื่อนตำแหน่งของพนักงานภายในองค์กร” นี้จัดทำขึ้นโดยมีจุดประสงค์เพื่อนำเสนอผลปฏิบัติตามกระบวนการ Cross Industry Standard Process for Data Mining (CRISP – DM) Process ที่ได้ศึกษาอันเป็นการฝึกฝนทักษะกระบวนการทำงานเป็นทีม และกระบวนการศึกษาค้นคว้าอย่างเป็นระบบ จนจัดทำเป็นรายงานเรื่องที่คณะผู้จัดทำสนใจ และเสนอให้อาจารย์ที่ปรึกษาได้พิจารณาเพื่อประกอบการเรียนวิชา 040223406 BUSINESS INTELLIGENCE ในภาคเรียนที่ 1 ปีการศึกษา 2566

เนื้อหาในโครงงานนี้ประกอบด้วยหัวข้อสำคัญทั้งสิ้น 5 หัวข้อ คือ หัวข้อที่ 1 บทนำ หัวข้อ ที่ 2 งานวิจัยที่เกี่ยวข้องและทฤษฎี หัวข้อที่ 3 วิธีการศึกษา หัวข้อที่ 4 ผลการศึกษา และหัวข้อที่ 5 สรุปผลและเอกสารอ้างอิง ซึ่งคณะผู้จัดทำได้พยายามเรียบเรียงขึ้นจากการศึกษาค้นคว้าอย่างเป็นระบบ

ประโยชน์ที่คณะผู้จัดทำได้รับจากการจัดทำรายงานเรื่องนี้ คือได้ฝึกฝนกระบวนการทำงานเป็นกลุ่ม การวางแผนการปรับปรุงแก้ไขงาน การศึกษาค้นคว้า เป็นต้น ผู้ที่สนใจอ่านโครงงานเรื่องนี้จะได้ทราบข้อมูล การค้นหาเชิงลึกด้วยกระบวนการ Cross Industry Standard Process for Data Mining (CRISP – DM) Process การนำผลการวิเคราะห์ไปใช้งาน และการเผยแพร่ผลงานด้วยการเล่าเรื่องด้วยการนำเสนอข้อมูล

ปัญหาที่เกิดขึ้นในการจัดทำรายงานเรื่องนี้คือการขาดประสบการณ์การทำ Cross Industry Standard Process for Data Mining (CRISP – DM) Process ไม่ว่าจะเป็นการรวบรวมและศึกษาข้อมูล รวมถึงการสร้างโมเดล และขอขอบคุณสมาชิกผู้จัดทำที่ให้ความร่วมมือกัน แก้ไขปัญหาต่าง ๆ จนงานสำเร็จลุล่วงและสามารถผ่านอุปสรรคไปได้ด้วยดี

คณะผู้จัดทำหวังเป็นอย่างยิ่งว่า โครงงานเรื่องนี้จะเกิดประโยชน์แก่ผู้ที่สนใจได้ตามสมควรและหากยังมีข้อบกพร่องประการใด ทางคณะผู้จัดทำขอน้อมรับคำติชมและข้อเสนอแนะไว้เพื่อปรับปรุงแก้ไขในโอกาสต่อไป

คณะผู้จัดทำ

04 กุมภาพันธ์ 2566

สารบัญ

บทที่	หน้า
กิตติกรรมประกาศ	ก
คำนำ	ข
สารบัญ	ค
สารบัญภาพ	จ
บทที่ 1	1
บทนำ	1
1.1 ที่มาและความสำคัญ	1
1.2 วัตถุประสงค์ของการศึกษา	1
1.3 สมมติฐานการศึกษา	2
บทที่ 2	3
งานวิจัยที่เกี่ยวข้องและทฤษฎี	3
2.1 ทฤษฎีของโมเดล Decision Tree	3
2.2 ทฤษฎีของโมเดล Logistic Regression	3
2.3 ทฤษฎี Machine Learning	4
2.4 ทฤษฎีของโมเดล Neural Network (Deep Learning)	5
2.5 ทฤษฎีของโมเดล Random Forest	5
2.6 ทฤษฎีของโมเดล Support Vector Machine (SVM)	6
บทที่ 3	7
วิธีการศึกษา	7
3.1 รวบรวมและศึกษาชุดข้อมูล (Data Understanding)	7
3.2 การจัดเตรียมข้อมูล (Data Preparation)	8

3.3 การสร้างโมเดล (Modeling)	10
บทที่ 4	15
ผลการศึกษา	15
4.1 การประเมินประสิทธิภาพของโมเดล	15
4.2 เปรียบเทียบประสิทธิภาพของแต่ละโมเดล	17
4.3 การนำผลการวิเคราะห์ไปใช้งานหรือพัฒนาระบบต่อ (Deployment)	18
บทที่ 5	19
สรุปผล	19
5.1 กระบวนการนำข้อมูลไปพัฒนาระบบ	19
5.2 ประโยชน์ที่ได้จากการพัฒนาระบบ	19
5.3 การเผยแพร่ผลงาน (Public) ด้วย การเล่าเรื่องด้วยการนำเสนอข้อมูล (Data Storytelling)	20
บรรณานุกรม	21

สารบัญภาพ

ภาพที่ 1 แสดงการลบ missing value ในคอลัมน์ education	8
ภาพที่ 2 แสดงการลบ missing value ในคอลัมน์ previous_year_rating	8
ภาพที่ 3 แสดงการแปลง m เป็น Male และแปลง f เป็น Female ในคอลัมน์ Gender	9
ภาพที่ 4 แสดงการลบคอลัมน์ employee_id, region, recruitment_channel, no_of_trainings, awards_won? ออก	9
ภาพที่ 5 แสดงการสร้างโมเดล Decision Tree	10
ภาพที่ 6 แสดงการสร้างโมเดล Logistic Regression	11
ภาพที่ 7 แสดงการสร้างโมเดล Neural Network (Deep Learning)	12
ภาพที่ 8 แสดงการสร้างโมเดล Neural Network (Deep Learning)	13
ภาพที่ 9 แสดงการสร้างโมเดล Random Forest	13
ภาพที่ 10 แสดงการสร้างโมเดล Support Vector Machine (SVM)	14
ภาพที่ 11 Data Storytelling	20

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญ

เนื่องจากในปัจจุบันพนักงานบริษัทโดยส่วนใหญ่ต่างทำงานอย่างหนักเพื่อหวังที่จะได้รับการเลื่อนตำแหน่งงานที่สูงขึ้นและได้รับเงินเดือนที่เพิ่มมากขึ้น และในแต่ละปีบริษัทจะคัดเลือกพนักงานที่เหมาะสมเพื่อที่จะทำการเลื่อนตำแหน่งให้ โดยพนักงานที่จะได้เลื่อนตำแหน่งในแต่ละปีนั้นมีจำนวนไม่มาก เมื่อเทียบกับประชากรพนักงานในบริษัท บริษัทจึงจะต้องพิจารณาจากหลักเกณฑ์ต่าง ๆ ว่าพนักงานบุคคลใดมีความเหมาะสมต่อตำแหน่งจะได้รับการเลื่อนตำแหน่ง ทั้งนี้การเลื่อนตำแหน่งก็ไม่ได้แปลว่าจะมีเพียงแค่การเพิ่มเงินเดือนเท่านั้น แต่หมายถึงการรับผิดชอบต่อหน้าที่ก็ต้องมีเพิ่มมากขึ้นอีกด้วย

เพราะฉะนั้นในการคัดเลือกผู้ได้รับการเลื่อนตำแหน่งจึงใช้หลักเกณฑ์และปัจจัยหลายอย่างเพื่อให้ได้บุคคลที่ดีที่สุดที่จะได้เลื่อนตำแหน่ง

ในรายงานเล่มนี้ ผู้จัดทำได้ทำการนำข้อมูลหลักเกณฑ์และปัจจัยต่างๆที่มีผลต่อการคัดเลือกพนักงานที่จะได้รับการเลื่อนตำแหน่ง โดยนำข้อมูลบุคลิกภาพและผลงานของพนักงานในหัวข้อต่าง ๆ มาคำนวณเพื่อหาแนวโน้มของพนักงานที่จะได้รับการคัดเลือกเพื่อเลื่อนตำแหน่ง เพื่อนำมานำเสนอและประกอบการตัดสินใจของแผนกผู้บริหาร และสามารถนำไปใช้ประโยชน์ในปีต่อ ๆ ไปได้

1.2 วัตถุประสงค์ของการศึกษา

1. เพื่อสร้างโมเดลพยากรณ์ความน่าจะเป็นของการเลื่อนตำแหน่งของพนักงาน โดยใช้ข้อมูลที่เรามีอยู่เพื่อช่วยในการตัดสินใจเกี่ยวกับการจัดสรรทรัพยากรและการพัฒนาบุคลากร
2. เพื่อเข้าใจคุณสมบัติหรือตัวแปรที่ส่งผลต่อการเลื่อนตำแหน่งของพนักงาน ที่จะช่วยกำหนดแนวทางในการสร้างโอกาสในการเลื่อนตำแหน่งสำหรับพนักงานในอนาคต
3. เพื่อช่วยวางแผนการจัดการทรัพยากรมนุษย์และการพัฒนาสังคมขององค์กรในอนาคต
4. เพื่อเข้าใจแนวโน้มในการเลื่อนตำแหน่งในองค์กร

1.3 สมมติฐานการศึกษา

H_0 : คะแนนการประเมินประจำปีที่สูงและคะแนนการอบรมที่สูงมีผลต่อโอกาสในการเลื่อนตำแหน่งโดยพนักงานที่มีคะแนนสูงในการประเมินและคะแนนการฝึกสอนสูงมีโอกาสรเลื่อนตำแหน่งสูงขึ้น

$$H_1: \mu_0 = \mu_1$$

μ_0 : คะแนนการประเมินประจำปีที่สูงและคะแนนการอบรมที่สูง

μ_1 : โอกาสเลื่อนตำแหน่งสูงขึ้น

บทที่ 2

งานวิจัยที่เกี่ยวข้องและทฤษฎี

การพยากรณ์การเลื่อนตำแหน่งของพนักงานภายในองค์กร คณะผู้จัดทำได้ศึกษางานวิจัยและทฤษฎีที่เกี่ยวข้องกับการพยากรณ์และในส่วนของเรียนรู้ที่เกี่ยวข้องโดยนำเสนอในลักษณะการเรียบเรียงเชิงสังเคราะห์โดยมีหัวข้อดังต่อไปนี้

2.1 ทฤษฎีของโมเดล Decision Tree

Decision Tree หรือ ต้นไม้ตัดสินใจเป็นโมเดลที่ใช้โครงสร้างต้นไม้เพื่อตัดสินใจในการแบ่งกลุ่มข้อมูล โดยเรียนรู้จากข้อมูลฝึกอบรม (Training Data) โดยเริ่มต้นที่โหนดหลัก (Root Node) และแยกกลุ่มข้อมูลออกไปเรื่อยๆ จนกระทั่งถึงโหนดใบ (Leaf Node) ที่มีคำตอบ ต้นไม้ตัดสินใจนี้มีความชัดเจนและเข้าใจง่าย ทำให้สามารถอธิบายตัวพยากรณ์ได้ง่าย แต่อาจไม่ค่อยมีความแม่นยำในการทำนายกรณีที่มีข้อมูลซับซ้อนมาก สามารถอธิบายหลักการทำงานได้ ดังนี้

1. การแบ่งข้อมูล โดยเริ่มที่โหนดหลัก (Root Node) และตรวจสอบข้อมูลเพื่อค้นหาคุณสมบัติที่ดีที่สุดในการแบ่งข้อมูล โดยใช้คำถามเชิงเปรียบเทียบ เช่น อายุมากกว่า 25 ปีหรือไม่ ?
2. การสร้างโหนดย่อย เมื่อคุณสมบัติที่ถูกค้นพบ จะสร้างโหนดย่อย (child node) สองข้างของโหนดหลัก และแบ่งกลุ่มข้อมูลออกเป็นกลุ่มย่อย ๆ ตามคุณสมบัติที่กำหนด เช่น พนักงานที่มีอายุมากกว่า 30 ปี และกลุ่มที่มีอายุน้อยกว่า 30 ปี
3. การทำซ้ำขั้นตอน ขั้นตอนการแบ่งแบบนี้ทำซ้ำไปเรื่อย ๆ ที่โหนดย่อยทุกๆ ครั้ง โดยค้นหาคุณสมบัติที่ดีที่สุดในการแบ่งกลุ่มข้อมูลย่อย
4. การสร้างโหนดใบ เมื่อไม่สามารถแบ่งกลุ่มข้อมูลได้อีกต่อไป (หรือเงื่อนไขหยุดถูกตัดสินใจ) โมเดลจะสร้างโหนดใบ (leaf node) ที่มีคำตอบหรือการพยากรณ์ คือ ในโหนดใบจะระบุผลลัพธ์ที่แน่นอน เช่น เลื่อนตำแหน่ง หรือ ไม่เลื่อนตำแหน่ง
5. การใช้ Decision Tree เมื่อโมเดล Decision Tree ถูกสร้างขึ้น คุณสามารถนำข้อมูลใหม่เข้ามาและให้โมเดลตัดสินใจว่าควรอยู่ในกลุ่มใดตามคุณสมบัติที่เป็นไปได้ที่ดีที่สุดที่ตามมาจากต้นไม้

2.2 ทฤษฎีของโมเดล Logistic Regression

Logistic Regression เป็นโมเดลที่ใช้ในการทำนายผลลัพธ์ที่เป็นไปได้ 2 ค่า (Binary outcome) โดยใช้ฟังก์ชันที่เรียกว่าฟังก์ชันโลจิสติก (logistic function) เพื่อคำนวณความน่าจะเป็นของผลลัพธ์ ในรายงานนี้

สามารถใช้ Logistic Regression เพื่อพยากรณ์ว่าพนักงานจะได้รับการเลื่อนตำแหน่งหรือไม่ โดยใช้ข้อมูลทางลักษณะของพนักงาน เช่น อายุ, ระดับการศึกษา เป็นต้น สามารถอธิบายหลักการทำงานได้ ดังนี้

1. การเริ่มต้น โมเดลเริ่มต้นด้วยการเก็บข้อมูลคุณสมบัติ (predictor variables) และผลลัพธ์ของเหตุการณ์ที่ต้องการพยากรณ์ (binary outcome) จากชุดข้อมูลการฝึกอบรม (Training Data)
2. การสร้างโมเดล จะใช้ข้อมูลคุณสมบัติในการสร้างสมการทายค่าความน่าจะเป็นในการเป็นไปได้ของผลลัพธ์ เหตุการณ์ที่สนใจในรูปแบบของสมการ Logistic Regression คือ $P(Y = 1 | X) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_k x_k)}}$
3. ที่ $P(Y = 1 | X)$ คือความน่าจะเป็นที่ Y มีค่าเป็น 1 ในขณะที่ X คือคุณสมบัติแต่ละอัน $\beta_0, \beta_1, \beta_2, \dots, \beta_k$ คือพารามิเตอร์ที่จะถูกปรับให้เข้ากับข้อมูลการฝึกอบรม (Training Data)
4. การปรับค่าพารามิเตอร์ (Parameter Estimation) การปรับค่าพารามิเตอร์ $\beta_0, \beta_1, \beta_2, \dots, \beta_k$ ให้เหมาะสมโดยใช้การจับคู่ข้อมูลคุณสมบัติและผลลัพธ์จากชุดข้อมูลการฝึกอบรม (Training Data) เพื่อให้โมเดลมีความสามารถในการทำนายและวิเคราะห์ในรูปแบบที่ดีที่สุด
5. การพยากรณ์ พารามิเตอร์ที่ได้จากขั้นตอนก่อนหน้านี้จะถูกนำไปใช้ในการพยากรณ์ความน่าจะเป็นที่เหตุการณ์ที่สนใจจะเป็นไปได้ในชุดข้อมูลทดสอบ (Test Data) หรือชุดข้อมูลใหม่

2.3 ทฤษฎี Machine Learning

Machine Learning เป็นเครื่องมือในการวิเคราะห์และการพยากรณ์ข้อมูล ในรายงานนี้คือการสร้างโมเดลที่สามารถพยากรณ์การเลื่อนตำแหน่งของพนักงาน การทดสอบและประเมินผลโมเดล การใช้งานและปรับปรุง โดยใช้ข้อมูลทางลักษณะและประวัติการทำงาน สามารถอธิบายหลักการทำงานได้ ดังนี้

1. การรวบรวมข้อมูล เริ่มต้นด้วยการรวบรวมข้อมูลที่เกี่ยวข้อง โดยรวบรวมข้อมูลจากแหล่งต่าง ๆ เช่น อายุ, ระดับการศึกษา, คะแนนการฝึกอบรมเฉลี่ย, ปีที่แล้วได้รับการประเมิน, ระยะเวลาการบริการ, และอื่น ๆ ที่อาจมีผลต่อการเลื่อนตำแหน่งของพนักงาน
2. การแบ่งข้อมูล ข้อมูลถูกแบ่งเป็นชุดการฝึกอบรม (training data) และชุดทดสอบ (test data) เพื่อใช้ในการสร้างและทดสอบโมเดล
3. การสร้างโมเดล โมเดลถูกสร้างขึ้นโดยใช้ข้อมูลการฝึกอบรม (Training Data) เป็นหลัก โมเดลนี้มีหน้าที่เรียนรู้ความสัมพันธ์ระหว่างคุณสมบัติและผลลัพธ์ที่เกี่ยวข้อง
4. การทดสอบและประเมินโมเดล หลังจากโมเดลถูกสร้างขึ้นจะถูกทดสอบด้วยชุดข้อมูลทดสอบเพื่อประเมินความแม่นยำและประสิทธิภาพของโมเดลในการพยากรณ์การเลื่อนตำแหน่งของพนักงาน

5. การใช้งานและการปรับปรุง เมื่อโมเดลถูกตรวจสอบและพบว่ามีประสิทธิภาพที่น่าพอใจ จะสามารถนำไปใช้งานในการพยากรณ์การเลื่อนตำแหน่งของพนักงานในองค์กร และสามารถปรับปรุงโมเดลเมื่อมีข้อมูลใหม่เพื่อทำให้มันมีประสิทธิภาพมากขึ้น

2.4 ทฤษฎีของโมเดล Neural Network (Deep Learning)

Neural Network (Deep Learning) เป็นโมเดลที่อธิบายความซับซ้อนของ Neural Network และวิธีการทำงานของ Deep Learning ในการเรียนรู้ทักษะที่ซับซ้อนของข้อมูล ในรายงานนี้ใช้ Neural Network (Deep Learning) เพื่อพยากรณ์การเลื่อนตำแหน่งของพนักงานโดยใช้ข้อมูลหลายประเภท สามารถอธิบายหลักการทำงานได้ ดังนี้

1. เซลล์ประสาท (Neurons) Neural Network ประกอบด้วยเซลล์ประสาทหลายตัว แต่ละเซลล์ประสาทมีหน้าที่รับข้อมูล, ประมวลผล, และส่งข้อมูลไปยังเซลล์ประสาทอื่น ๆ
2. การเรียนรู้ Neural Network สามารถเรียนรู้โดยปรับค่าพารามิเตอร์ภายในโมเดลเพื่อให้สามารถพยากรณ์ผลลัพธ์ที่ถูกต้อง การเรียนรู้สามารถเกิดขึ้นผ่านกระบวนการการคำนวณและปรับค่าพารามิเตอร์ตลอดเวลา
3. เลเยอร์ (Layers) Neural Network ประกอบด้วยเลเยอร์หลายชั้นที่ประกอบด้วยเซลล์ประสาท แต่ละเลเยอร์มีหน้าที่แตกต่างกัน โดยเลเยอร์แรกเรียกว่าเลเยอร์นำเข้า (input layer) ที่รับข้อมูลเข้า และเลเยอร์สุดท้ายเรียกว่าเลเยอร์เอาต์พุต (output layer) ที่ส่งผลลัพธ์ออก
4. การส่งสัญญาณ (Forward Propagation) ในกระบวนการ Forward Propagation ข้อมูลถูกส่งผ่านเลเยอร์ต่าง ๆ โดยทำการคำนวณและประมวลผลทางคณิตศาสตร์ เพื่อให้ได้ผลลัพธ์ที่ถูกต้อง
5. การคำนวณค่าคลาดเคลื่อน (Error Calculation) หลังจาก Forward Propagation ค่าคลาดเคลื่อนระหว่างผลลัพธ์ที่ได้จากโมเดลและผลลัพธ์ที่ควรจะได้ถูกคำนวณ เพื่อใช้ในการปรับค่าพารามิเตอร์.
6. การปรับค่าพารามิเตอร์ (Backpropagation) ในกระบวนการ Backpropagation ค่าคลาดเคลื่อนถูกถ่ายย้อนกลับผ่านเลเยอร์ทั้งหมดเพื่อปรับค่าพารามิเตอร์ในโมเดล ที่ช่วยให้โมเดลเรียนรู้จากข้อมูลการฝึกอบรม (Training Data)
7. การสร้างโมเดล Deep Learning คือการใช้ Neural Network ที่มีหลายเลเยอร์ (deep neural network) ซึ่งช่วยให้โมเดลเรียนรู้คุณสมบัติที่มีความซับซ้อนและมีการเชื่อมต่อกัน

2.5 ทฤษฎีของโมเดล Random Forest

Random Forest เป็นโมเดลที่ใช้หลายโมเดล Decision Tree เพื่อเพิ่มความแม่นยำและป้องกันการเกิดโมเดลเชิงเจียง (Overfitting) โดยการสร้างหลายโมเดล Decision Tree และรวมผลลัพธ์จากทุกโมเดลเพื่อคำนวณ

ผลลัพธ์สุดท้าย โมเดล Random Forest มีความสามารถในการจัดกลุ่มข้อมูลที่ซับซ้อนและมีความแม่นยำ สามารถอธิบายหลักการทำงานได้ ดังนี้

1. การสร้างชุดข้อมูลสุ่ม (Bootstrap) โมเดลจะใช้กระบวนการสร้างชุดข้อมูลสุ่มจากชุดข้อมูลการฝึกอบรม (Training Data) โดยการสุ่มข้อมูลอย่างสุ่มหรือแบบมีทับซ้อน (with replacement) ทำให้สร้างชุดข้อมูลสุ่มที่มีข้อมูลตัวอย่างหลาย ๆ ชุด
2. การสร้าง Decision Trees สำหรับแต่ละชุดข้อมูลสุ่ม โมเดลจะสร้าง Decision Tree เป็นโมเดลเฉพาะสำหรับชุดข้อมูลนั้น ๆ
3. การพยากรณ์ผลลัพธ์ เมื่อมี Decision Trees หลาย ๆ ต้นสร้างขึ้น โมเดลจะให้แต่ละต้นสร้างการพยากรณ์ของผลลัพธ์แยกต่างหาก ผลลัพธ์ที่ได้จากแต่ละต้นจะมีน้ำหนักเท่ากัน
4. การรวมผลลัพธ์ ผลลัพธ์ที่ได้จากทุก ๆ Decision Trees ในโมเดลถูกรวมรวบเข้าด้วยกัน (ensemble) เพื่อให้ได้ผลลัพธ์ที่ถูกต้องและแม่นยำมากขึ้น โดยใช้วิธีโหวตหรือการคำนวณค่าเฉลี่ยของผลลัพธ์
5. ความสำคัญของคุณสมบัติ (Feature Importance) โมเดลสามารถประเมินความสำคัญของแต่ละคุณสมบัติในการพยากรณ์ผลลัพธ์ ซึ่งช่วยในการวิเคราะห์ข้อมูลและการพยากรณ์การเลื่อนตำแหน่งของพนักงาน

2.6 ทฤษฎีของโมเดล Support Vector Machine (SVM)

Support Vector Machine (SVM) เป็นโมเดลที่ใช้จัดกลุ่มข้อมูลโครงสร้างแผนตัดเส้น (Hyperplane) ที่เลือกโมเดลที่มีขอบความผิดพลาดน้อยที่สุด แผนตัดเส้นนี้จะหยิบข้อมูลจากกลุ่มต่าง ๆ และใช้ในการแบ่งกลุ่มข้อมูลเป็นกลุ่ม SVM ที่มีความแม่นยำในการจัดกลุ่มข้อมูลและยังสามารถใช้กับข้อมูลที่มีลักษณะเชิงเส้นและไม่เชิงเส้น ในรายงานนี้อาจใช้เพื่อจัดกลุ่มพนักงานเป็นกลุ่มที่มีโอกาสในการเลื่อนตำแหน่งและกลุ่มที่ไม่มีโอกาสสามารถอธิบายหลักการทำงานได้ ดังนี้

1. การสร้างแบบโมเดล การสร้างโมเดลจะเริ่มต้นด้วยการเก็บข้อมูลคุณสมบัติและผลลัพธ์จากชุดข้อมูลการฝึกอบรม (Training Data)
2. การหาโมเดลแบ่งแยก จะหาแบบโมเดลที่สามารถแบ่งแยกข้อมูลอย่างชัดเจนระหว่างกลุ่มข้อมูลต่าง ๆ โดยใช้แผนการแบ่งที่ดีที่สุด (decision boundary) ที่มีระยะห่างที่มากที่สุดระหว่างข้อมูลแต่ละจุดและแผนการแบ่งนี้บ่งบอกถึงข้อมูลของแต่ละกลุ่ม
3. การหาเวกเตอร์สนับสนุน จากแผนการแบ่งที่ดีที่สุดนี้จะค้นหาเวกเตอร์สนับสนุน (support vectors) ที่เป็นข้อมูลที่อยู่ใกล้กับแผนการแบ่งและมีส่วนร่วมในการกำหนดแผนการแบ่งนี้
4. การพยากรณ์ เมื่อได้โมเดล SVM และเวกเตอร์สนับสนุน สามารถนำไปใช้ในการพยากรณ์ผลลัพธ์ในชุดข้อมูลทดสอบ (Test Data) หรือชุดข้อมูลใหม่

บทที่ 3

วิธีการศึกษา

การพยากรณ์การเลื่อนตำแหน่งของพนักงานภายในองค์กร ในการดำเนินการพยากรณ์ในครั้งนี้เป็นการศึกษาด้วยกระบวนการ Cross Industry Standard Process for Data Mining (CRISP – DM) Process เพื่อศึกษาประสิทธิภาพการทำงานของโมเดลต่าง ๆ ในการพยากรณ์ ดังนั้นเพื่อให้การดำเนินการพยากรณ์ดำเนินการวิจัยไปตามขั้นตอนจึงดำเนินการดังนี้

3.1 รวบรวมและศึกษาชุดข้อมูล (Data Understanding)

1. แหล่งข้อมูล ข้อมูลถูกรวบรวมโดยเว็บไซต์ Kaggle ในหัวข้อ HR Analysis Case Study

2. ลักษณะของชุดข้อมูล ประกอบด้วยคอลัมน์ 14 คอลัมน์ ดังนี้

1. employee_id
2. department
3. region
4. education
5. gender
6. recruitment_channel
7. no_of_trainings
8. age
9. previous_year_rating
10. length_of_service
11. KPIs_met >80%
12. awards_won?
13. avg_training_score
14. is_promoted

3. ปริมาณของชุดข้อมูล ประกอบด้วย 54809 แถว และ 14 คอลัมน์

4. ความเหมาะสมต่อการวิเคราะห์ ข้อมูลมีข้อมูลที่จำเป็นต่อการพยากรณ์

3.2 การจัดเตรียมข้อมูล (Data Preparation)

1. จำนวน Transactions ก่อนจัดเตรียมข้อมูล มี 54809 แถว และ 14 คอลัมน์
2. ทำความสะอาดข้อมูล (Data Cleansing) โดยตรวจสอบและแก้ไขข้อมูลที่ผิดพลาด ในข้อมูลนี้พบ missing value ในคอลัมน์ education, previous_year_rating จึงทำการลบ missing value ออก



employee_id Number	department Category	region Category	education Category	gender Category	recruitment_c... Category	no_of_trainings Number	age Number	previous_year... Number	length_of_ser... Number	KPIs_met >80% Number
65141	Operations	region_22	Bachelor's	m	other	1	30	5	4	0
7513	Sales & Marketing	region_19	Bachelor's	m	sourcing	1	34	3	7	0
2542	Sales & Marketing	region_23	Bachelor's	m	other	2	39	1	10	0
48945	Technology	region_26	Bachelor's	m	other	1	45	3	2	0
58896	Analytics	region_2	Bachelor's	m	sourcing	2	31	3	7	0
20379	Operations	region_20	Bachelor's	f	other	1	31	3	5	0
16290	Operations	region_34	Master's & above	m	sourcing	1	33	3	6	0
73202	Analytics	region_20	Bachelor's	m	other	1	28	4	5	0
28911	Sales & Marketing	region_1	Master's & above	m	sourcing	1	32	5	5	1
29934	Technology	region_23	?	m	sourcing	1	30	?	1	0
49017	Sales & Marketing	region_7	Bachelor's	f	sourcing	1	35	5	3	1
60051	Sales & Marketing	region_4	Bachelor's	m	sourcing	1	49	5	5	1

54,808 rows - 14 columns (5 nominal, 9 numerical)

ภาพที่ 1 แสดงการลบ missing value ในคอลัมน์ education



employee_id Number	department Category	region Category	education Category	gender Category	recruitment_c... Category	no_of_trainings Number	age Number	previous_year... Number	length_of_ser... Number	KPIs_met >80% Number
65141	Operations	region_22	Bachelor's	m	other	1	30	5	4	0
7513	Sales & Marketing	region_19	Bachelor's	m	sourcing	1	34	3	7	0
2542	Sales & Marketing	region_23	Bachelor's	m	other	2	39	1	10	0
48945	Technology	region_26	Bachelor's	m	other	1	45	3	2	0
58896	Analytics	region_2	Bachelor's	m	sourcing	2	31	3	7	0
20379	Operations	region_20	Bachelor's	f	other	1	31	3	5	0
16290	Operations	region_34	Master's & above	m	sourcing	1	33	3	6	0
73202	Analytics	region_20	Bachelor's	m	other	1	28	4	5	0
28911	Sales & Marketing	region_1	Master's & above	m	sourcing	1	32	5	5	1
49017	Sales & Marketing	region_7	Bachelor's	f	sourcing	1	35	5	3	1
60051	Sales & Marketing	region_4	Bachelor's	m	sourcing	1	49	5	5	1
38401	Technology	region_29	Master's & above	m	other	2	39	3	16	0

52,399 rows - 14 columns (5 nominal, 9 numerical)

ภาพที่ 2 แสดงการลบ missing value ในคอลัมน์ previous_year_rating

3. การแปลงข้อมูล (Data Transformation) โดย แปลง m เป็น Male และแปลง f เป็น Female ในคอลัมน์ Gender ลบคอลัมน์ employee_id, region, recruitment_channel, no_of_trainings, awards_won? ออก เนื่องจากไม่มีผลในการพยากรณ์



employee_id Number	department Category	region Category	education Category	gender Category	recruitment_c... Category	no_of_trainings Number	age Number	previous_year... Number	length_of_ser... Number	KPIs_met>80% Number
1	Analytics	region_7	Bachelor's	Male	sourcing	2	29	3	5	0
2	Finance	region_2	Master's & above	Female	sourcing	1	35	1	2	0
4	Sales & Marketing	region_2	Bachelor's	Male	other	1	25	3	2	0
5	Analytics	region_7	Master's & above	Male	other	2	46	3	7	0
7	Operations	region_32	Bachelor's	Male	other	1	31	3	7	0
8	Sales & Marketing	region_7	Bachelor's	Male	other	1	42	5	6	1
9	Sales & Marketing	region_27	Bachelor's	Male	sourcing	1	29	5	2	1
10	Operations	region_24	Master's & above	Male	other	1	35	4	9	0
12	Technology	region_31	Bachelor's	Male	other	1	24	3	2	1
14	Operations	region_2	Bachelor's	Female	sourcing	2	38	5	8	1
15	Sales & Marketing	region_2	Bachelor's	Male	sourcing	2	39	3	12	0
18	Operations	region_24	Bachelor's	Female	sourcing	1	33	3	4	0

48,660 rows - 14 columns (5 nominal, 9 numerical)

ภาพที่ 3 แสดงการแปลง m เป็น Male และแปลง f เป็น Female ในคอลัมน์ Gender



department Category	education Category	gender Category	age Number	previous_year_rati... Number	length_of_service Number	KPIs_met>80% Number	avg_training_score Number	is_promoted Number
Analytics	Bachelor's	Male	29	3	5	0	85	0
Finance	Master's & above	Female	35	1	2	0	63	0
Sales & Marketing	Bachelor's	Male	25	3	2	0	53	0
Analytics	Master's & above	Male	46	3	7	0	86	0
Operations	Bachelor's	Male	31	3	7	0	59	0
Sales & Marketing	Bachelor's	Male	42	5	6	1	52	0
Sales & Marketing	Bachelor's	Male	29	5	2	1	50	0
Operations	Master's & above	Male	35	4	9	0	60	0
Technology	Bachelor's	Male	24	3	2	1	77	0
Operations	Bachelor's	Female	38	5	8	1	60	0
Sales & Marketing	Bachelor's	Male	39	3	12	0	50	0
Operations	Bachelor's	Female	33	3	4	0	57	0

48,660 rows - 9 columns (3 nominal, 6 numerical)

ภาพที่ 4 แสดงการลบคอลัมน์ employee_id, region, recruitment_channel, no_of_trainings, awards_won? ออก

4. จำนวน Transactions ที่เหลือหลังจัดเตรียมข้อมูล มี 48660 แถว และ 9 คอลัมน์

3.3 การสร้างโมเดล (Modeling)

1. สร้างโมเดล Decision Tree

- เริ่มต้นโดยการแบ่งชุดข้อมูลเป็นชุดฝึก (Training Set) และชุดทดสอบ (Test Set) โดยใช้ `train_test_split`
- สร้างโมเดล Decision Tree โดยใช้ `DecisionTreeClassifier` จาก `scikit-learn`
- Training Data โดยใช้ฟังก์ชัน `fit`
- ผลลัพธ์ที่ได้จากต้นไม้จะแบ่งแยกข้อมูลตามเงื่อนไขที่อยู่ในแต่ละโหนด (Nodes) และโหนดใบ (Leaf Node) ที่แทนค่าผลลัพธ์ที่พยากรณ์
- และทำการประเมินผลโมเดล เพื่อดูว่าโมเดลมีประสิทธิภาพในการพยากรณ์

```

1 import pandas as pd
2 from sklearn.tree import DecisionTreeClassifier
3 from sklearn.model_selection import train_test_split
4 from sklearn import metrics
5 from sklearn.metrics import classification_report, accuracy_score
6
7 data = pd.read_csv('Employee_Clean.csv')
8
9 feature_cols = ["department", "education", "gender", "age",
10                 "previous_year_rating", "length_of_service", "KPIs_met >80%", "avg_training_score"]
11 #แยกคอลัมน์ feature และ target
12 X = data[feature_cols]
13 y = data.is_promoted
14
15 #แบ่งข้อมูลเป็น train, test
16 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=1)
17
18 #แปลง ข้อความ เป็น ตัวเลข
19 X_train = pd.get_dummies(X_train, columns=['department', 'education', 'gender'], drop_first=True)
20 X_test = pd.get_dummies(X_test, columns=['department', 'education', 'gender'], drop_first=True)
21
22 clf = DecisionTreeClassifier() #สร้าง Decision Tree Classifier
23 clf = clf.fit(X_train, y_train) #วิเคราะห์ด้วย training data
24 y_pred = clf.predict(X_test) #พยากรณ์ด้วย test data
25
26 # ประเมินผลโมเดล
27 print(classification_report(y_test, y_pred))
28 print("Accuracy:", accuracy_score(y_test, y_pred))

```

ภาพที่ 5 แสดงการสร้างโมเดล Decision Tree

2. สร้างโมเดล Logistic Regression

- เริ่มต้นโดยการแบ่งชุดข้อมูลเป็นชุดฝึก (Training Set) และชุดทดสอบ (Test Set)
- สร้างโมเดล Model Logistic Regression โดยใช้ LogisticRegression จาก scikit-learn
- ทำการ Training Data และ Test Data
- และทำการประเมินผลโมเดล เพื่อดูว่าโมเดลมีประสิทธิภาพในการพยากรณ์

```

1 import numpy as np
2 import pandas as pd
3 from sklearn.model_selection import train_test_split
4 from sklearn.preprocessing import StandardScaler
5 from sklearn.linear_model import LogisticRegression
6 from sklearn.metrics import classification_report, accuracy_score
7
8 # โหลดข้อมูล
9 data = pd.read_csv('Employee_Clean.csv')
10 # แปลงข้อมูลและแบ่งชุดข้อมูล
11 X = data.drop(columns=['is_promoted'])
12 y = data['is_promoted']
13
14 X = pd.get_dummies(X, columns=['department', 'education', 'gender'])
15
16 # แบ่งชุดข้อมูลเป็นชุดฝึกและชุดทดสอบ
17 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
18
19 # มาตรฐานข้อมูล
20 scaler = StandardScaler()
21 X_train = scaler.fit_transform(X_train)
22 X_test = scaler.transform(X_test)
23
24 # สร้างโมเดล Logistic Regression
25 model = LogisticRegression()
26
27 # ฝึกโมเดล
28 model.fit(X_train, y_train)
29
30 # ทดสอบโมเดล
31 y_pred = model.predict(X_test)
32
33 # ประเมินผลโมเดล
34 print(classification_report(y_test, y_pred))
35 print("Accuracy:", accuracy_score(y_test, y_pred))

```

ภาพที่ 6 แสดงการสร้างโมเดล Logistic Regression

3. สร้างโมเดล Neural Network (Deep Learning)

- เริ่มต้นโดยการแบ่งชุดข้อมูลชุดฝึก (Training Set) และชุดทดสอบ (Test Set)
- สร้างโมเดล Neural Network โดยใช้ไลบรารี deep learning เช่น TensorFlow, Keras Sequential
- กำหนดโครงสร้างของโมเดล Neural Network, จำนวนชั้น (layers), จำนวนโหนด (neurons) ในแต่ละชั้น, ฟังก์ชันกระตุ้น (activation functions)
- วิเคราะห์ค่าความน่าจะเป็น (Probability) ที่โมเดลพยากรณ์ผลลัพธ์ว่าอยู่ในกลุ่มที่ถูกเลื่อนตำแหน่งหรือไม่ได้เลื่อนตำแหน่ง
- และทำการประเมินผลโมเดล เพื่อดูว่าโมเดลมีประสิทธิภาพในการพยากรณ์

```

1 import numpy as np
2 import pandas as pd
3 from sklearn.model_selection import train_test_split
4 from sklearn.preprocessing import StandardScaler
5 from keras.models import Sequential
6 from keras.layers import Dense
7 from keras.optimizers import Adam
8 from sklearn.metrics import classification_report, accuracy_score
9
10 # โหลดข้อมูล
11 data = pd.read_csv('Employee_Clean.csv')
12
13 # แปลงข้อมูลและแบ่งชุดข้อมูล
14 X = data.drop(columns=['is_promoted']) # ลบคอลัมน์ 'is_promoted' จากตัวแปรอิสระ
15 y = data['is_promoted']
16 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=1)
17
18 X_train = pd.get_dummies(X_train, columns=['department', 'education', 'gender'])
19 X_test = pd.get_dummies(X_test, columns=['department', 'education', 'gender'])
20
21 scaler = StandardScaler()
22 X_train = scaler.fit_transform(X_train)
23 X_test = scaler.transform(X_test)

```

ภาพที่ 7 แสดงการสร้างโมเดล Neural Network (Deep Learning)

```

25 # สร้างโมเดล Neural Network
26 model = Sequential()
27 model.add(Dense(units=64, activation='relu', input_dim=X_train.shape[1]))
28 model.add(Dense(units=32, activation='relu'))
29 model.add(Dense(units=1, activation='sigmoid'))
30
31 model.compile(loss='binary_crossentropy', optimizer=Adam(learning_rate=0.001), metrics=['accuracy'])
32
33 # ฝึกโมเดล
34 model.fit(X_train, y_train, epochs=10, batch_size=64)
35
36 # ทดสอบโมเดล
37 y_pred = model.predict(X_test)
38 y_pred = (y_pred > 0.5) # ตัดสินใจค่าส่งเสริมหรือไม่ด้วยค่าความน่าจะเป็น
39
40 # ประเมินผลโมเดล
41 print(classification_report(y_test, y_pred))
42 print("Accuracy:", accuracy_score(y_test, y_pred))

```

ภาพที่ 8 แสดงการสร้างโมเดล Neural Network (Deep Learning)

4. สร้างโมเดล Random Forest

- เริ่มต้นโดยการแบ่งชุดข้อมูลเป็นชุดฝึก (Training Set) และชุดทดสอบ (Test Set) โดยใช้ `train_test_split`
- สร้างโมเดล Random Forest โดยใช้ `RandomForestClassifier` จาก `scikit-learn`
- กำหนดพารามิเตอร์ จำนวนต้นไม้ (`n_estimators`) ใน Random Forest
- Training Data โดยใช้ฟังก์ชัน `fit`
- และทำการประเมินผลโมเดล เพื่อดูว่าโมเดลมีประสิทธิภาพในการพยากรณ์

```

1 import pandas as pd
2 from sklearn.model_selection import train_test_split
3 from sklearn.ensemble import RandomForestClassifier
4 from sklearn.metrics import accuracy_score, classification_report
5
6 data = pd.read_csv("Employee_Clean.csv")
7
8 # การเตรียมข้อมูล: แปลงตัวแปรแบบอักษรเป็นตัวเลข
9 data = pd.get_dummies(data, columns=["department", "education", "gender"])
10
11 # เป้าหมาย
12 X = data.drop("is_promoted", axis=1)
13 y = data["is_promoted"]
14
15 # แบ่งชุดข้อมูลเป็นชุดฝึกและชุดทดสอบ
16 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)
17
18 # สร้างโมเดล Random Forest
19 rf_model = RandomForestClassifier(n_estimators=100, random_state=42) # โมเดล Random Forest ใช้ 100 ต้นไม้ในการพยากรณ์การเลื่อนตำแหน่ง (is_promoted)
20 # ฝึก
21 rf_model.fit(X_train, y_train)
22 # ทดสอบ
23 y_pred = rf_model.predict(X_test)
24
25 # ประเมิน
26 accuracy = accuracy_score(y_test, y_pred)
27 print(classification_report(y_test, y_pred))
28 print(f"Accuracy: {accuracy}")

```

ภาพที่ 9 แสดงการสร้างโมเดล Random Forest

4. สร้างโมเดล Support Vector Machine (SVM)

- เริ่มต้นโดยการแบ่งชุดข้อมูลเป็นชุดฝึก (Training Set) และชุดทดสอบ (Test Set)
- สร้างโมเดล SVM โดยใช้ SVC (Support Vector Classification) หรือ SVR (Support Vector Regression) จาก scikit-learn
- ผลการพยากรณ์จะแสดงแนวตั้งและแนวนอน (Support Vectors) ที่กำหนดเส้นแบ่งระหว่างกลุ่มข้อมูลแต่ละกลุ่ม และแสดงถึงค่าของพารามิเตอร์ C ที่กำหนดความเข้มของการละเมิด (penalty) ต่อการกำหนดค่าผิดพลาด (misclassification) ของข้อมูล
- ทำการประเมินผลโมเดล เพื่อดูว่าโมเดลมีประสิทธิภาพในการพยากรณ์

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.svm import SVC # ใช้ SVC แทน SVM
from sklearn.metrics import classification_report, accuracy_score

data = pd.read_csv("Employee_Clean.csv") #โหลดข้อมูล

#แบ่งข้อมูลเป็น train, test
X = data.drop("is_promoted", axis=1)
y = data["is_promoted"]
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=1)

#แปลง department
X_train = pd.get_dummies(X_train, columns=['department'], drop_first=True)
X_test = pd.get_dummies(X_test, columns=['department'], drop_first=True)

#แปลง education เป็นตัวเลข
education_mapping = {
    "Master's & above": 3,
    "Bachelor's": 2,
    "Below Secondary": 1
}
X_train['education'] = X_train['education'].map(education_mapping)
X_test['education'] = X_test['education'].map(education_mapping)

#แปลง gender เป็น ตัวเลข
gender_mapping = {
    "Male": 0,
    "Female": 1
}
X_train['gender'] = X_train['gender'].map(gender_mapping)
X_test['gender'] = X_test['gender'].map(gender_mapping)

# ทำการมาตรฐานข้อมูล Standardization
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)

svm_model = SVC(kernel='linear', C=1.0) # ในกรณีนี้เราใช้ kernel เป็นเส้นตรง (linear)
# train model
svm_model.fit(X_train, y_train)
# test model
y_pred = svm_model.predict(X_test)

# ประเมินผลโมเดล
print(classification_report(y_test, y_pred))
```

บทที่ 4

ผลการศึกษา

4.1 การประเมินประสิทธิภาพของโมเดล

1. โมเดล Decision Tree สามารถแสดงผลลัพธ์ของการประเมินประสิทธิภาพของโมเดลได้ดังนี้

- ค่า Precision สำหรับคลาส 0 (ไม่เลื่อนตำแหน่ง) คือ 0.95 ซึ่งหมายความว่าพยากรณ์ว่าไม่เลื่อนตำแหน่งมี 95 % ที่จริง ๆ ไม่เลื่อนตำแหน่ง ส่วนคลาส 1 (เลื่อนตำแหน่ง) คือ 0.44 ซึ่งหมายความว่าพยากรณ์ว่าเลื่อนมี 44 % ที่จริง ๆ เลื่อนตำแหน่ง
- ค่า Recall สำหรับคลาส 0 (ไม่เลื่อนตำแหน่ง) คือ 0.94 ซึ่งหมายความว่าโมเดลครอบคลุม 94 % ของกรณีไม่เลื่อนตำแหน่ง สำหรับคลาส 1 (เลื่อนตำแหน่ง) คือ 0.45 ซึ่งหมายความว่าครอบคลุม 45 % ของกรณีที่เลื่อนตำแหน่ง
- ค่า F1 – Score เป็นค่าเฉลี่ยระหว่างค่า Precision และ Recall โดยสำหรับคลาส 0 คือ 0.95 และสำหรับคลาส 1 คือ 0.44
- ค่า Accuracy คือ 0.90 ซึ่งหมายความว่าโมเดลมีความแม่นยำประมาณ 90 % ในการพยากรณ์ผลการเลื่อนตำแหน่ง

ค่า Precision, Recall สำหรับคลาส 1 ค่อนข้างต่ำ ซึ่งอาจส่งผลให้โมเดลมีความแม่นยำสูง แต่ความครอบคลุมต่ำ จะทำให้โมเดลที่มีความครอบคลุมน้อยและพบปัญหาในการพยากรณ์คลาส 1 โดยที่มีความแม่นยำสูงไม่พอในบางกรณี

2. โมเดล Logistic Regression สามารถแสดงผลลัพธ์ของการประเมินประสิทธิภาพของโมเดลได้ดังนี้

- ค่า Precision สำหรับคลาส 0 (ไม่เลื่อนตำแหน่ง) คือ 0.93 ซึ่งหมายความว่าพยากรณ์ว่าไม่เลื่อนตำแหน่งมี 93 % ที่จริง ๆ ไม่เลื่อนตำแหน่ง ส่วนคลาส 1 (เลื่อนตำแหน่ง) คือ 0.86 ซึ่งหมายความว่าพยากรณ์ว่าเลื่อนมี 86 % ที่จริง ๆ เลื่อนตำแหน่ง
- ค่า Recall สำหรับคลาส 0 (ไม่เลื่อนตำแหน่ง) คือ 1.00 ซึ่งหมายความว่าโมเดลครอบคลุมทุกกรณีของการไม่เลื่อนตำแหน่ง สำหรับคลาส 1 (เลื่อนตำแหน่ง) คือ 0.26 ซึ่งหมายความว่าครอบคลุมเพียง 26 % ของกรณีที่เลื่อนตำแหน่ง
- ค่า F1 – Score เป็นค่าเฉลี่ยระหว่างค่า Precision และ Recall โดยสำหรับคลาส 0 คือ 0.96 และสำหรับคลาส 1 คือ 0.40
- ค่า Accuracy คือ 0.93 ซึ่งหมายความว่าโมเดลมีความแม่นยำประมาณ 93 % ในการพยากรณ์ผลการเลื่อนตำแหน่ง

ผลลัพธ์นี้แสดงว่าโมเดลมีความแม่นยำสูงสำหรับคลาส 0 (ไม่เลื่อนตำแหน่ง) แต่ความครอบคลุมสำหรับคลาส 1 (เลื่อนตำแหน่ง) ซึ่งหมายความว่าโมเดลมีความแม่นยำสูงในการพยากรณ์คลาส 0 แต่ไม่ค่อยดีในการทำนายคลาส 1

3. โมเดล Neural Network (Deep Learning) สามารถแสดงผลลัพธ์ของการประเมินประสิทธิภาพของโมเดลได้ดังนี้

- ค่า Precision สำหรับคลาส 0 (ไม่เลื่อนตำแหน่ง) คือ 0.94 ซึ่งหมายความว่าพยากรณ์ว่าไม่เลื่อนตำแหน่งมี 94 % ที่จริง ๆ ไม่เลื่อนตำแหน่ง ส่วนคลาส 1 (เลื่อนตำแหน่ง) คือ 0.96 ซึ่งหมายความว่าพยากรณ์ว่าเลื่อนมี 96 % ที่จริง ๆ เลื่อนตำแหน่ง
- ค่า Recall สำหรับคลาส 0 (ไม่เลื่อนตำแหน่ง) คือ 1.00 ซึ่งหมายความว่าโมเดลครอบคลุมทุกกรณีของการไม่เลื่อนตำแหน่ง สำหรับคลาส 1 (เลื่อนตำแหน่ง) คือ 0.34 ซึ่งหมายความว่าครอบคลุมเพียง 34 % ของกรณีที่เลื่อนตำแหน่ง
- ค่า F1 – Score เป็นค่าเฉลี่ยระหว่างค่า Precision และ Recall โดยสำหรับคลาส 0 คือ 0.97 และสำหรับคลาส 1 คือ 0.50
- ค่า Accuracy คือ 0.94 ซึ่งหมายความว่าโมเดลมีความแม่นยำประมาณ 94 % ในการพยากรณ์ผลการเลื่อนตำแหน่ง

ผลลัพธ์นี้แสดงว่าโมเดลมีความแม่นยำสูงสำหรับคลาส 0 (ไม่เลื่อนตำแหน่ง) และคลาส 1 (เลื่อนตำแหน่ง) และครอบคลุมทุกกรณีของการไม่เลื่อนตำแหน่ง ค่า F1 – Score ของคลาส 1 (เลื่อนตำแหน่ง) ค่อนข้างสูง ซึ่งแสดงให้เห็นว่าโมเดลมีความแม่นยำและครอบคลุมที่ดีสำหรับทั้งคลาส 0 และ คลาส 1

4. โมเดล Random Forest สามารถแสดงผลลัพธ์ของการประเมินประสิทธิภาพของโมเดลได้ดังนี้

- ค่า Precision สำหรับคลาส 0 (ไม่เลื่อนตำแหน่ง) คือ 0.94 ซึ่งหมายความว่าพยากรณ์ว่าไม่เลื่อนตำแหน่งมี 94 % ที่จริง ๆ ไม่เลื่อนตำแหน่ง ส่วนคลาส 1 (เลื่อนตำแหน่ง) คือ 0.69 ซึ่งหมายความว่าพยากรณ์ว่าเลื่อนมี 69 % ที่จริง ๆ เลื่อนตำแหน่ง
- ค่า Recall สำหรับคลาส 0 (ไม่เลื่อนตำแหน่ง) คือ 0.98 ซึ่งหมายความว่าโมเดลครอบคลุม 98 % ของกรณีที่ ไม่เลื่อนตำแหน่ง สำหรับคลาส 1 (เลื่อนตำแหน่ง) คือ 0.35 ซึ่งหมายความว่าครอบคลุมเพียง 35 % ของกรณีที่เลื่อนตำแหน่ง
- ค่า F1 – Score เป็นค่าเฉลี่ยระหว่างค่า Precision และ Recall โดยสำหรับคลาส 0 คือ 0.96 และสำหรับคลาส 1 คือ 0.46
- ค่า Accuracy คือ 0.93 ซึ่งหมายความว่าโมเดลมีความแม่นยำประมาณ 93 % ในการพยากรณ์ผลการเลื่อนตำแหน่ง

ผลลัพธ์นี้แสดงว่าโมเดลมีความแม่นยำสูงสำหรับคลาส 0 (ไม่เลื่อนตำแหน่ง) และครอบคลุมสูง ส่วนในคลาส 1 (เลื่อนตำแหน่ง) ความแม่นยำค่อนข้างต่ำและความครอบคลุมก็ไม่สูงมาก

5. โมเดล Support Vector Machine (SVM) สามารถแสดงผลลัพธ์ของการประเมินประสิทธิภาพของโมเดลได้ดังนี้

- ค่า Precision สำหรับคลาส 0 (ไม่เลื่อนตำแหน่ง) คือ 0.92 ซึ่งหมายความว่าพยากรณ์ว่าไม่เลื่อนตำแหน่งมี 92 % ที่จริง ๆ ไม่เลื่อนตำแหน่ง ส่วนคลาส 1 (เลื่อนตำแหน่ง) คือ 1.00 ซึ่งหมายความว่าพยากรณ์ว่าเลื่อนมี 100 % ที่จริง ๆ เลื่อนตำแหน่ง
- ค่า Recall สำหรับคลาส 0 (ไม่เลื่อนตำแหน่ง) คือ 1.00 ซึ่งหมายความว่าโมเดลครอบคลุมทุกกรณีของการไม่เลื่อนตำแหน่ง สำหรับคลาส 1 (เลื่อนตำแหน่ง) คือ 0.12 ซึ่งหมายความว่าครอบคลุมเพียง 12 % ของกรณีที่เลื่อนตำแหน่ง
- ค่า F1 – Score เป็นค่าเฉลี่ยระหว่างค่า Precision และ Recall โดยสำหรับคลาส 0 คือ 0.96 และสำหรับคลาส 1 คือ 0.21
- ค่า Accuracy คือ 0.92 ซึ่งหมายความว่าโมเดลมีความแม่นยำประมาณ 92 % ในการพยากรณ์ผลการเลื่อนตำแหน่ง

ผลลัพธ์นี้แสดงว่าโมเดลมีความแม่นยำสูงสำหรับคลาส 0 (ไม่เลื่อนตำแหน่ง) และความครอบคลุมสูง ส่วนคลาส 1 (เลื่อนตำแหน่ง) ความแม่นยำค่อนข้างสูงในการพยากรณ์คลาส 1 แต่ไม่ค่อยดีในการครอบคลุมทุกกรณีของคลาสนี้

4.2 เปรียบเทียบประสิทธิภาพของแต่ละโมเดล

เมื่อเปรียบเทียบโมเดลทั้ง 5 โมเดล ตามค่า Accuracy จะได้โมเดล Neural Network มีความแม่นยำสูงสุดที่ประมาณ 94% และมีค่า F1-Score สำหรับคลาส 1 (เลื่อนตำแหน่ง) ที่สูงสุด คือ 0.50 ส่วน Decision Tree และ Random Forest มีความแม่นยำสำหรับคลาส 0 (ไม่เลื่อนตำแหน่ง) สูง แต่ความแม่นยำสำหรับคลาส 1 (เลื่อนตำแหน่ง) ต่ำกว่า โมเดลอื่น ๆ ส่วน SVM มีความแม่นยำสูงสำหรับคลาส 0 แต่ความแม่นยำสำหรับคลาส 1 ต่ำมาก

4.3 การนำผลการวิเคราะห์ไปใช้งานหรือพัฒนาระบบต่อ (Deployment)

การนำโมเดลที่ได้จากการประเมินผลไปใช้งานต่อได้ดังนี้

1. การนำโมเดลไปใช้ในการพยากรณ์ โมเดลที่ได้จากการประเมินผลสามารถใช้ในการพยากรณ์ข้อมูลเพื่อเข้าใจสถานการณ์หรือความสัมพันธ์ต่าง ๆ โดยใช้โมเดลสร้างกราฟ หรือผลลัพธ์ในรูปแบบอื่น ๆ เพื่อใช้ในการตัดสินใจหรือพยากรณ์ข้อมูลเพิ่มเติม
2. การแก้ไขปรับปรุงโมเดล โมเดลที่ได้จากการประเมินผลไม่ใช่โมเดลสุดท้ายและสมบูรณ์เสมอไป การนำโมเดลไปแก้ไขปรับปรุงโมเดล เพื่อปรับความแม่นยำหรือประสิทธิภาพของโมเดล เช่น การเพิ่ม Features ใหม่, การเปลี่ยนค่าพารามิเตอร์
3. การพัฒนาระบบต่อ ผลลัพธ์และโมเดลที่ได้จากการประเมินผลสามารถนำไปพัฒนาระบบหรือแอปพลิเคชันต่อไปได้ เพื่อทำให้ระบบมีฟังก์ชันมากขึ้น หรือแก้ไขประสิทธิภาพการทำงาน ในกรณีการพยากรณ์การเลื่อนตำแหน่ง สามารถนำโมเดลไปใช้ในระบบ HRM (Human Resource Management) เพื่อช่วยจัดการและควบคุมการเลื่อนตำแหน่งของพนักงานได้
4. การช่วยตัดสินใจและดำเนินการ ผลลัพธ์และโมเดลที่ได้จากการประเมินผลสามารถช่วยตัดสินใจและดำเนินการได้ ในกรณีการพยากรณ์การเลื่อนตำแหน่ง จะสามารถกำหนดเกณฑ์การเลื่อนตำแหน่งจากโมเดล และตัดสินใจเกี่ยวกับการเลื่อนตำแหน่งของพนักงานได้อีกด้วย
5. การดูแลและควบคุมโมเดล การรักษาความแม่นยำและประสิทธิภาพของโมเดลและระบบการเลื่อนตำแหน่ง จะต้องดูแลและควบคุมโมเดลสม่ำเสมอ เพื่อสามารถแก้ไขหรือตรวจสอบความถูกต้องของข้อมูลที่นำมาใช้ในการพยากรณ์ได้
6. การรวบรวมข้อมูลการใช้งานและผลการพยากรณ์ เพื่อให้สามารถแก้ไขและปรับปรุงระบบเลื่อนตำแหน่งในอนาคตได้ และยังสามารถสร้างฐานข้อมูลของการเลื่อนตำแหน่งที่เกิดขึ้นอีกด้วย

บทที่ 5

สรุปผล

5.1 กระบวนการนำข้อมูลไปพัฒนาระบบ

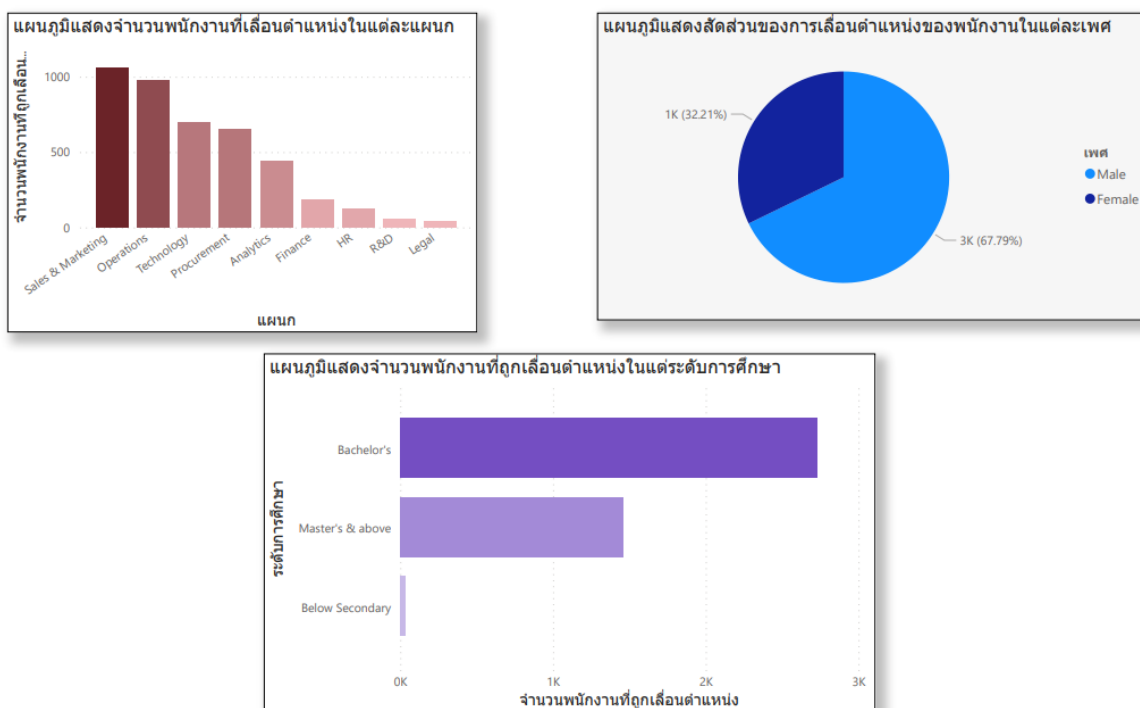
การพยากรณ์การเลื่อนตำแหน่งของพนักงานภายในองค์กรสามารถนำไปพัฒนาระบบได้ดังนี้

1. เลือกโมเดลที่เหมาะสมกับระบบ ในรายงานจะประกอบด้วยโมเดล Decision Tree, โมเดล Logistic Regression, โมเดล Neural Network (Deep Learning), โมเดล Random Forest และโมเดล Support Vector Machine (SVM) ซึ่งขึ้นอยู่กับลักษณะข้อมูลและการพยากรณ์
2. นำเข้าข้อมูล นำข้อมูลที่ได้จากกระบวนการพยากรณ์และสรุปผลมาใช้ในระบบหรือแอปพลิเคชัน
3. แปลงข้อมูล ข้อมูลที่นำเข้ามาต้องอยู่ในรูปแบบที่สามารถนำไปใช้งานต่อได้ รวมถึงการจัดการข้อมูลให้ถูกต้องและเข้ากันได้กับโมเดลที่นำมาใช้งาน
4. ใช้พัฒนาระบบหรือแอปพลิเคชัน การใช้พัฒนาต้องคำนึงถึงการจัดการข้อมูล, แสดงผล, ประสิทธิภาพของการประมวลผล
5. ทดสอบระบบ หลังจากพัฒนาระบบควรทดสอบระบบเพื่อตรวจสอบความถูกต้องและความเสถียรของระบบ ว่าทำงานเป็นอย่างไร รวมถึงการทดสอบการพยากรณ์ข้อมูล, ประสิทธิภาพ, การใช้งาน
6. ดูแลและอัปเดตระบบ ควรดูแลและอัปเดตระบบอย่างสม่ำเสมอ เพื่อให้ระบบสามารถใช้งานได้มีประสิทธิภาพตลอด และป้องกันปัญหาที่อาจเกิดขึ้นได้

5.2 ประโยชน์ที่ได้จากการพัฒนาระบบ

1. ช่วยพยากรณ์เหตุการณ์หรือแนวโน้มในอนาคต เพื่อค้นหาข้อสรุปมาช่วยตัดสินใจ
2. ช่วยเลือกพนักงานที่เหมาะสมสำหรับตำแหน่งต่าง ๆ โดยลดความผิดพลาดในการคัดเลือก
3. ช่วยสร้างความพึงพอใจและความพร้อมใจให้กับพนักงาน เนื่องจากเห็นว่ากระบวนการเลือกมีความยุติธรรม
4. ช่วยเพิ่มความเข้าใจของระบบบริหารจัดการทรัพยากรมนุษย์ และการจัดทำแผนยุทธศาสตร์

5.3 การเผยแพร่ผลงาน (Public) ด้วย การเล่าเรื่องด้วยการนำเสนอข้อมูล (Data Storytelling)



ภาพที่ 11 Data Storytelling

บรรณานุกรม

- รัสรินทร์ เมธาเฉลิมพัฒน์. (2565). การประยุกต์ใช้ Machine Learning กับงานในภาคอุตสาหกรรม. สืบค้น 25 ตุลาคม 2566, จาก <https://www.nectec.or.th/news-public-document/machine-learning>
- อภิชาติ ศุภรณี. (2566). Build the first neural network. ใน **Machine Learning for Mathematics**. กรุงเทพฯ. มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ
- อภิชาติ ศุภรณี. (2566). Decision Tree Algorithm for Classification Problems. ใน **Machine Learning for Mathematics**. กรุงเทพฯ. มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ
- อภิชาติ ศุภรณี. (2566). Performance Metrics for Classification Problems. ใน **Machine Learning for Mathematics**. กรุงเทพฯ. มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ
- BenzTech. (2564). วิธีใส่เลขหน้ารายงาน Word (เลขหน้าต่างกัน). สืบค้น 6 พฤศจิกายน 2566, จาก <https://youtu.be/Loo9zqX5j7k?si=hjRz7miITG-QCscZ>
- Datacamp. (2562). Support Vector Machines with Scikit-learn Tutorial. สืบค้น 25 ตุลาคม 2566, จาก <https://www.datacamp.com/tutorial/svm-classification-scikit-learn-python>
- Datacamp. (2562). Understanding Logistic Regression in Python Tutorial. สืบค้น 25 ตุลาคม 2566, จาก <https://www.datacamp.com/tutorial/svm-classification-scikit-learn-python>
- Natthanan Bhukan. (2563). Deploy model machine learning ฉบับมือใหม่. สืบค้น 27 ตุลาคม 2566, จาก <https://medium.com/@RTae/deploy-model-machine-learning>
- Pumi Boonyatud. (2565). HR Forecasting มาพยากรณ์และทำนายอนาคตของ HR กันเถอะ!. สืบค้น 20 ตุลาคม 2566, จาก <https://th.hrnote.asia/orgdevelopment/221216-hr-forecasting/>
- SHIVAN KUMAR. (2563). HR Analysis Case Study. สืบค้น 20 ตุลาคม 2566, จาก <https://www.kaggle.com/datasets/shivan118/hranalysis/data>
- scikit-learn. (2550). Support Vector Machines. สืบค้น 31 ตุลาคม 2566, จาก <https://scikit-learn.org/stable/modules/svm.html>