ใบความรู้ที่ 5 การเรียนรู้แบบมีไม่มีผู้สอน

วัตถุประสงค์การเรียนรู้

1. รู้ความหมายและหลักการของ Unsupervised Learning (การเรียนรู้แบบไม่มีผู้สอน)
2. เข้าใจความแตกต่างระหว่าง Supervised Learning และ Unsupervised Learning
3. เข้าใจหลักการทำงานของอัลกอริทึม K-Means Clustering ในการจัดกลุ่มข้อมูล
4. รู้ความสำคัญของการทำ Feature Scaling และผลกระทบต่อการคำนวณระยะทางในโมเดล
5. รู้จักวิธีการประเมินผลโมเดลและการหาค่า K ที่เหมาะสม (Elbow Method และ Silhouette Score)
6. เข้าใจแนวทางการประยุกต์ใช้ K-Means เพื่อทำ Customer Segmentation ด้วยหลักการ RFM

1. บทนำปรัชญาการเรียนรู้

ในการทำความเข้าใจแก่นแท้ของการเรียนรู้แบบไม่มีผู้สอน (Unsupervised Learning) จำเป็นต้องย้อนกลับไปเปรียบเทียบกับกระบวนทัศน์ (Paradigm) ของการเรียนรู้แบบมีผู้สอน (Supervised Learning) ที่ผู้เรียนได้ศึกษาผ่านมาในบทเรียนก่อนหน้า (เช่น Linear Regression) การเรียนรู้แบบมีผู้สอนเปรียบเสมือนการสอนเด็กเล็กด้วย "บัตรคำศัพท์" (Flashcards) กระบวนการนี้มีโครงสร้างที่ชัดเจนและเข้มงวด กล่าวคือ ผู้สอนจะแสดงภาพ (Input Data หรือ X) พร้อมกับเฉลยหรือป้ายกำกับ (Label หรือ y) เช่น การชูภาพสุนัขและบอกว่า "นี่คือสุนัข" เป้าหมายของโมเดลคือการจดจำและสร้างฟังก์ชันที่สามารถจับคู่ภาพกับชื่อเรียกได้อย่างถูกต้อง เพื่อให้เมื่อเจอภาพใหม่ในอนาคต (Test Set) จะสามารถระบุคำตอบได้ตามสิ่งที่ถูกสอนมา

ในทางตรงกันข้าม โลกแห่งความเป็นจริงของข้อมูลมักไม่ได้ใจดีและเป็นระเบียบเรียบร้อยเช่นนั้น ข้อมูลส่วนใหญ่ที่องค์กรหรือนักวิจัยเผชิญคือ "กองข้อมูลดิบ" (Raw Data) มหาศาลที่ปราศจากป้ายกำกับ หรือเฉลยใดๆ ในสถานการณ์นี้ ปัญญาประดิษฐ์ไม่สามารถทำหน้าที่เป็นนักเรียนที่รอเฉลยจากครูได้อีกต่อไป แต่ต้องสวมบทบาทเป็น "นักสำรวจ" (The Explorer) ที่ถูกปล่อยลงกลางป่าใหญ่หรือดินแดนลึกลับโดยไม่มีแผนที่ หน้าที่ของ AI คือการสังเกตสภาพแวดล้อม พิจารณาลักษณะร่วมของวัตถุต่างๆ และจัดระเบียบสิ่งที่ค้นพบด้วยตัวเอง หาก AI พบกลุ่มสิ่งมีชีวิตที่มีปีกและบินได้ มันจะจัดให้อยู่ในกลุ่มเดียวกัน และแยกออกจากกลุ่มสิ่งมีชีวิตที่ว่ายน้ำ แม้ว่า AI จะไม่รู้คำศัพท์ว่า "นก" หรือ "ปลา" ก็ตาม กระบวนการนี้สะท้อนถึงปรัชญาญาณวิทยา (Epistemology) ของการเรียนรู้มนุษย์ที่สามารถสร้างมโนทัศน์ (Concept Formation) ได้จากการสังเกตแพทเทิร์นซ้ำๆ โดยไม่ต้องมีผู้ชี้แนะตลอดเวลา

ดังนั้น นิยามเชิงลึกของการเรียนรู้แบบไม่มีผู้สอน คือกระบวนการทางคณิตศาสตร์และสถิติในการค้นหาโครงสร้างที่ซ่อนอยู่ (Hidden Structures) หรือรูปแบบ (Patterns) ภายในชุดข้อมูลที่ไม่มีฉลากกำกับ (Unlabeled Data) แทนที่จะทำนายผลลัพธ์ที่ระบุไว้ล่วงหน้า อัลกอริทึมจะพยายามอธิบายความสัมพันธ์ระหว่างข้อมูล เพื่อสร้างความเข้าใจใหม่ (Insights) ซึ่งเป็นทักษะที่จำเป็นอย่างยิ่งสำหรับการวิเคราะห์ข้อมูลเชิงลึก (Exploratory Data Analysis) และการทำเหมืองข้อมูล (Data Mining)

2. ทฤษฎีและกลไกคณิตศาสตร์ของอัลกอริทึม K-Means

2.1. ประวัติความเป็นมาและแนวคิดพื้นฐาน

ในบรรดาเทคนิคการจัดกลุ่มข้อมูล (Clustering) ที่มีอยู่มากมาย K-Means Clustering ถือเป็นอัลกอริทึมที่ได้รับความนิยมสูงสุดและเป็นรากฐานสำคัญของ Machine Learning ต้นกำเนิดของแนวคิดนี้ย้อนกลับไปถึงทศวรรษที่ 1950 โดยได้รับการพัฒนาและเสนออย่างเป็นทางการโดย Stuart Lloyd ในปี 1957 ที่ Bell Labs เพื่อใช้ในการทำ Pulse-code modulation ก่อนที่จะได้รับการตีพิมพ์เผยแพร่อย่างเป็นทางการในภายหลัง ความยืนยงของอัลกอริทึมนี้มาจากการทำงานที่เรียบง่ายแต่ทรงพลังในการจัดการกับข้อมูลขนาดใหญ่

หัวใจสำคัญของ K-Means คือการแบ่งข้อมูลจำนวน n ตัวออกเป็น k กลุ่ม (Clusters) โดยที่แต่ละกลุ่มจะมีตัวแทนกลุ่มที่เรียกว่า "จุดศูนย์กลาง" (Centroid) ข้อมูลแต่ละตัวจะถูกจัดให้อยู่ในกลุ่มที่มีจุดศูนย์กลางใกล้ที่สุด ซึ่งในทางคณิตศาสตร์ นี่คือการแบ่งพื้นที่ของข้อมูลออกเป็นส่วนๆ ตามแผนภาพ Voronoi (Voronoi Diagram) โดยที่ขอบเขตของแต่ละกลุ่มจะถูกกำหนดโดยเส้นแบ่งครึ่งตั้งฉากระหว่างคู่ของจุดศูนย์กลาง

2.2. กระบวนการทำงานแบบวนซ้ำ (Iterative Process)

การทำงานของ K-Means ไม่ได้เกิดขึ้นและจบลงในขั้นตอนเดียว แต่เป็นกระบวนการวนซ้ำ (Iterative Method) ที่พยายามปรับปรุงผลลัพธ์ให้ดีขึ้นเรื่อยๆ จนกว่าจะถึงจุดสมดุล ขั้นตอนการทำงานสามารถจำแนกออกได้เป็น 4 ขั้นตอนหลัก ดังนี้ :

2.2.1. การกำหนดค่า K และการเริ่มต้น (Initialization): ผู้ใช้งานต้องกำหนดค่า Hyperparameter ที่สำคัญที่สุดคือ k ซึ่งระบุจำนวนกลุ่มที่ต้องการแบ่ง (เช่น k=3) จากนั้นอัลกอริทึมจะทำการสุ่มวางจุดศูนย์กลางเริ่มต้น (Initial Centroids) จำนวน k จุดลงในพื้นที่ของข้อมูล (Feature Space) การสุ่มจุดเริ่มต้นนี้มีความสำคัญมาก เพราะหากสุ่มได้ไม่ดีอาจทำให้อัลกอริทึมใช้เวลานานในการหาคำตอบหรือติดอยู่ในคำตอบที่ดีที่สุดเฉพาะที่ (Local Optima) ปัจจุบันจึงนิยมใช้วิธี K-Means++ ในการเลือกจุดเริ่มต้น ซึ่งใช้วิธีทางสถิติในการกระจายจุดศูนย์กลางให้ห่างจากกันมากที่สุดเพื่อแก้ปัญหานี้

2.2.2. การวัดระยะและจัดกลุ่ม (Assignment Step): ขั้นตอนนี้เปรียบเสมือนการทำงานของแม่เหล็ก โดยอัลกอริทึมจะคำนวณระยะห่างระหว่างจุดข้อมูลแต่ละจุด (Data Point) กับจุดศูนย์กลางทั้ง k จุด ข้อมูลตัวใดอยู่ใกล้จุดศูนย์กลางของกลุ่มไหนที่สุด ก็จะถูกดึงดูดเข้าไปเป็นสมาชิกของกลุ่มนั้น ในทางทฤษฎี ขั้นตอนนี้คือการลดค่าความแปรปรวนภายในกลุ่ม (Within-Cluster Variance) ให้เหลือน้อยที่สุด

2.2.3. การปรับตำแหน่งจุดศูนย์กลาง (Update Step): หลังจากข้อมูลทุกจุดถูกจัดเข้ากลุ่มแล้ว จุดศูนย์กลางเดิมที่ถูกสุ่มขึ้นมาอาจจะไม่ได้อยู่ที่ "ใจกลาง" ของกลุ่มจริงๆ อัลกอริทึมจึงต้องคำนวณหาจุดศูนย์กลางใหม่ โดยการหาค่าเฉลี่ย (Mean) ของพิกัดข้อมูลทุกจุดที่เป็นสมาชิกในกลุ่มนั้น แล้วย้ายจุดศูนย์กลางไปยังตำแหน่งใหม่ที่คำนวณได้ นี่คือที่มาของชื่อ "K-Means" (K-ค่าเฉลี่ย)

2.2.4. การตรวจสอบการลู่เข้า (Convergence Check): ทำซ้ำขั้นตอนที่ 2 และ 3 ไปเรื่อยๆ คือวัดระยะใหม่ จัดกลุ่มใหม่ และหาค่าเฉลี่ยใหม่ กระบวนการนี้จะหยุดลงเมื่อถึงเงื่อนไขที่กำหนด เช่น จุดศูนย์กลางไม่มีย้ายตำแหน่งอีกต่อไป (Stable Centroids) หรือค่าความคลาดเคลื่อนรวมลดลงจนถึงเกณฑ์ที่ยอมรับได้ ซึ่งแสดงว่าอัลกอริทึมได้เรียนรู้โครงสร้างข้อมูลจนสมบูรณ์แล้ว

2.3. คณิตศาสตร์แห่งระยะทาง: Euclidean Distance

กลไกหลักที่ขับเคลื่อนการตัดสินใจของ K-Means ในขั้นตอน Assignment คือการวัด "ความเหมือน" หรือ "ความใกล้ชิด" ซึ่งในทางเรขาคณิตวิเคราะห์ จะใช้สมการ ระยะทางยูคลิเดียน (Euclidean Distance) เป็นมาตรฐาน

สมมติว่าเรามีข้อมูลที่มี 2 ตัวแปร (Features) คือ รายได้ (x_1) และ อายุ ((x_2) ระยะทางระหว่างจุดข้อมูลลูกค้า $A(x_{A1}, x_{A2})$ และจุดศูนย์กลาง $C(c_1, c_2)$ จะถูกคำนวณด้วยสูตรพีทาโกรัส:

$$d(A, C) = \sqrt{(x_{A1} - c_1)^2 + (x_{A^2} - c_2)^2}$$

และสามารถขยายสูตรนี้ไปสู่ n มิติ (กรณีมีหลายตัวแปร เช่น มีทั้ง อายุ, รายได้, ความถี่ในการซื้อ, ยอดซื้อรวม) ได้ดังนี้:

$$d(p, Q) = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}$$

สมการนี้ชี้ให้เห็นประเด็นสำคัญที่มักถูกมองข้าม: อัลกอริทึมนี้อ่อนไหวต่อ "ขนาด" (Magnitude) ของตัวเลขอย่างยิ่ง หากตัวแปรใดมีค่าตัวเลขที่สูงกว่าตัวแปรอื่นมาก ผลต่างยกกำลังสองของตัวแปรนั้นจะมีค่ามหาศาลและครอบงำผลการคำนวณระยะทางทั้งหมด ทำให้ AI "มองเห็น" แค่ตัวแปรที่มีค่ามากและ "เพิกเฉย" ตัวแปรที่มีค่าน้อย ซึ่งนำไปสู่ความจำเป็นของการทำ Feature Scaling

2.4. ฟังก์ชันวัตถุประสงค์ (Objective Function): Inertia

ในเชิงคณิตศาสตร์ K-Means พยายามแก้ปัญหาการหาค่าเหมาะสมที่สุด (Optimization Problem) โดยมีเป้าหมายเพื่อลดค่า Inertia หรือผลรวมความคลาดเคลื่อนกำลังสองภายในกลุ่ม (Within-Cluster Sum of Squares: WCSS) ให้มีค่าน้อยที่สุด สมการของ Inertia คือ:

$$WCSS = \sum_{i=1}^{k} \sum_{x \in C_i} \|x - \mu_i\|^2$$

โดยที่

- k คือจำนวนของกลุ่ม
- c_i แทนกลุ่มลำดับที่ i
- x คือจุดข้อมูลที่อยู่ในกลุ่ม c_i
- μ_i คือจุดศูนย์กลาง (Centroid) ของกลุ่ม c_i
- $\|x - \mu_i\|^2$ คือระยะห่างแบบยูคลิดยกกำลังสอง ระหว่างจุด x กับจุดศูนย์กลาง μ_i

ค่า Inertia นี้สะท้อนถึง "ความหนาแน่น" หรือ "ความเกาะกลุ่ม" ของข้อมูล ยิ่งค่าต่ำแสดงว่าข้อมูลเกาะกลุ่มกันแน่นรอบจุดศูนย์กลาง ซึ่งเป็นสิ่งที่เราต้องการ อย่างไรก็ตาม เราไม่สามารถใช้ Inertia เพียงอย่างเดียวในการตัดสินว่าโมเดลดีที่สุด เพราะหากเราเพิ่มจำนวนกลุ่ม k ไปเรื่อยๆ จนเท่ากับจำนวนข้อมูล ($k = n$) ค่า Inertia จะกลายเป็น 0 เสมอ (เพราะจุดศูนย์กลางทับจุดข้อมูลพอดี) แต่โมเดลนั้นจะไม่มีประโยชน์ในการสรุปข้อมูล จึงต้องใช้วิธีอื่นเช่น Elbow Method มาช่วยพิจารณาความเหมาะสมร่วมด้วย

3. วิศวกรรมข้อมูล: ความสำคัญของการปรับสเกล (Feature Scaling)

3.1. ปัญหา "อายุ vs รายได้" และความบิดเบือนของมิติ

หนึ่งในอุปสรรคที่ใหญ่ที่สุดของการนำ K-Means ไปประยุกต์ใช้จริง และเป็นหัวข้อที่ถูกเน้นย้ำอย่างหนักในบทเรียนวิดีโอ คือปัญหาความแตกต่างของหน่วยวัดและช่วงข้อมูล (Scale) เพื่อให้เห็นภาพชัดเจน เราสามารถใช้อุปมา (Analogy) เรื่อง "อายุ vs รายได้"

จินตนาการว่าเรากำลังวิเคราะห์ข้อมูลลูกค้าสองคน:

- ลูกค้า ก: อายุ 30 ปี, รายได้ 20,000 บาท
- ลูกค้า ข: อายุ 80 ปี, รายได้ 20,000 บาท ความแตกต่างของอายุคือ 50 ปี ซึ่งในทางประชากรศาสตร์ถือว่าแตกต่างกันอย่างสิ้นเชิง (คนหนุ่มสาว vs ผู้สูงอายุ) แต่ระยะห่างทางคณิตศาสตร์คือ 50 หน่วย

ในทางกลับกัน:

- ลูกค้า ค: อายุ 30 ปี, รายได้ 20,050 บาท เมื่อเทียบลูกค้า ก กับ ค ความแตกต่างของรายได้คือ 50 บาท ซึ่งในทางเศรษฐกิจถือว่าแทบไม่ต่างกันเลย

แต่สำหรับ K-Means ที่คำนวณระยะทางแบบทื่อๆ มันจะมองว่า "ความต่างของอายุ 50 ปี" มีค่าเท่ากับ "ความต่างของเงิน 50 บาท" ยิ่งไปกว่านั้น หากรายได้เป็นหลักแสนหรือหลักล้าน ความต่างเพียงเล็กน้อยของรายได้ (เช่น 1,000 บาท) จะมีค่าทางตัวเลขมากกว่าความต่างของอายุทั้งช่วงชีวิต (เช่น 100 ปี) อย่างมหาศาล ส่งผลให้ K-Means ตัดสินใจแบ่งกลุ่มโดยดูแค่ "รายได้" เป็นหลัก และมองข้าม "อายุ" ไปโดยปริยาย การวัดระยะทางจึงเกิดความ "เพี้ยน" (Distortion) และไม่สะท้อนความเหมือนที่แท้จริง

3.2. การแก้ปัญหาด้วย Standardization (Z-Score)

เพื่อป้องกันปัญหานี้ นักวิทยาศาสตร์ข้อมูลจำเป็นต้องทำ Feature Scaling หรือการปรับสเกลข้อมูลให้เป็นมาตรฐานเดียวกันก่อนนำเข้าสู่โมเดล วิธีการที่นิยมและมีประสิทธิภาพสูงสำหรับ K-Means คือ Standardization หรือการแปลงเป็นค่ามาตรฐาน Z-Score

กระบวนการนี้จะเปลี่ยนข้อมูลดิบ (x) ของแต่ละฟีเจอร์ ให้กลายเป็นคะแนนมาตรฐาน (z) โดยใช้สูตรทางสถิติ

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

โดยที่:

- μ (Mu) คือค่าเฉลี่ย (Mean) ของฟีเจอร์นั้น ๆ
- σ (Sigma) คือส่วนเบี่ยงเบนมาตรฐาน (Standard Deviation)

ผลลัพธ์ที่ได้คือ ข้อมูลทุกตัวแปรไม่ว่าจะมีหน่วยเป็น "ปี" หรือ "บาท" จะถูกแปลงให้อยู่ในสเกลเดียวกัน คือมีค่าเฉลี่ยเท่ากับ 0 และมีความแปรปรวน (Variance) เท่ากับ 1 การทำเช่นนี้ทำให้การเปลี่ยนแปลง 1 หน่วยของค่า z ในตัวแปรอายุ มีความหมายทางสถิติเท่ากับการเปลี่ยนแปลง 1 หน่วยในตัวแปรรายได้ (คือเปลี่ยนไป 1 ส่วนเบี่ยงเบนมาตรฐานเท่ากัน) ทำให้ K-Means สามารถชั่งน้ำหนักความสำคัญของทุกตัวแปรได้อย่างยุติธรรมและเท่าเทียมกัน

4. การประเมินผลและการหาจำนวนกลุ่มที่เหมาะสม (Optimizing K)

เนื่องจากการเรียนรู้แบบไม่มีผู้สอนไม่มี "เฉลย" (Ground Truth) ให้ตรวจคำตอบ การวัดความแม่นยำ (Accuracy) จึงทำไม่ได้เหมือนใน Supervised Learning เราจึงต้องใช้เทคนิคพิเศษเพื่อประเมินว่าการแบ่งกลุ่มนั้น "ดี" หรือไม่ และจำนวนกลุ่ม k เท่าไหร่ถึงจะเหมาะสมที่สุด

4.1. วิธี Elbow Method

Elbow Method เป็นเทคนิคฮิวริสติก (Heuristic) ที่ใช้กราฟเพื่อหาจุดสมดุลระหว่างความละเอียดของการแบ่งกลุ่มกับความซับซ้อนของโมเดล วิธีการคือการรันอัลกอริทึม K-Means หลายๆ รอบ โดยเปลี่ยนค่า k ไปเรื่อยๆ (เช่น จาก 1 ถึง 10) และคำนวณค่า Inertia (WCSS) ของแต่ละ k มาพล็อตลงกราฟ

- เมื่อ k=1 ค่า Inertia จะสูงมาก เพราะข้อมูลทุกตัวอยู่ห่างจากจุดศูนย์กลางเดียว
- เมื่อเพิ่ม k ค่า Inertia จะลดลง เพราะจุดศูนย์กลางมีมากขึ้นและอยู่ใกล้ข้อมูลมากขึ้น
- ณ จุดหนึ่ง อัตราการลดลงของ Inertia จะเริ่มชะลอตัวลงอย่างชัดเจนจนกราฟมีลักษณะหักงอเหมือน "ข้อศอก" (Elbow)

จุดข้อศอกนี้คือนัยสำคัญที่บ่งบอกว่า การเพิ่มจำนวนกลุ่มมากกว่านี้ไม่ได้ช่วยให้ข้อมูลเกาะกลุ่มกันแน่นขึ้นอย่างมีนัยสำคัญอีกแล้ว จึงถือเป็นค่า k ที่เหมาะสมที่สุด ในทางปฏิบัติ การเขียนโค้ดเพื่อหาค่านี้มักจะวนลูปค่า k และเก็บค่า kmeans.inertia_ มาพล็อตกราฟ

4.2. การวิเคราะห์ด้วย Silhouette Score

นอกเหนือจาก Elbow Method แล้ว อีกวิธีที่มีความแม่นยำและเป็นปรนัยมากกว่าคือการใช้ Silhouette Score ซึ่งเป็นค่าดัชนีชี้วัดคุณภาพของการจัดกลุ่มโดยพิจารณาสองปัจจัยพร้อมกัน

4.2.1. Cohesion (a): ข้อมูลในกลุ่มเดียวกันเกาะกันแน่นแค่ไหน (ยิ่งใกล้ยิ่งดี)

4.2.2. Separation (b): ข้อมูลกลุ่มนี้อยู่ห่างจากกลุ่มอื่นแค่ไหน (ยิ่งไกลยิ่งดี)

สูตรการคำนวณสำหรับข้อมูลแต่ละจุดคือ:

$$S = \frac{b - a}{\max(a, b)}$$

ค่า Silhouette Score จะอยู่ระหว่าง -1 ถึง +1

- ค่าเข้าใกล้ +1: ดีมาก ข้อมูลอยู่ถูกกลุ่ม เกาะกันแน่นและห่างจากกลุ่มอื่น
- ค่าใกล้ 0: ข้อมูลอยู่ก้ำกึ่งระหว่างรอยต่อของกลุ่ม (Overlapping)
- ค่าติดลบ: ข้อมูลถูกจัดผิดกลุ่ม

การใช้ Silhouette Analysis ช่วยให้นักพัฒนา AI มั่นใจได้ว่าค่า k ที่เลือกนั้นทำให้เกิดกลุ่มที่แยกจากกันอย่างชัดเจนจริง ไม่ใช่แค่การแบ่งซอยพื้นที่ทางคณิตศาสตร์

5. กรณีศึกษา: การประยุกต์ใช้ทางธุรกิจ (Customer Segmentation & RFM)

เพื่อให้เห็นภาพการนำทฤษฎีไปใช้จริง บทเรียนนี้ได้ยกตัวอย่างการทำ Customer Segmentation ซึ่งเป็น Use Case ที่คลาสสิกที่สุดของ K-Means โดยประยุกต์ใช้ร่วมกับทฤษฎีการตลาด RFM Analysis

5.1. ทฤษฎี RFM Analysis

RFM เป็นกรอบแนวคิดในการวิเคราะห์พฤติกรรมลูกค้าโดยดูจากตัวแปรสำคัญ 3 ตัว :

5.1.1. Recency (R): ความสดใหม่ในการซื้อ - ลูกค้ามาซื้อครั้งล่าสุดเมื่อไหร่? (ค่ายิ่งน้อยยิ่งดี แสดงว่าเพิ่งมาซื้อ)

5.1.2. Frequency (F): ความถี่ - ลูกค้ามาซื้อบ่อยแค่ไหน? (ค่ายิ่งมากยิ่งดี)

5.1.3. Monetary (M): ยอดเงิน - ลูกค้าใช้จ่ายไปเท่าไหร่? (ค่ายิ่งมากยิ่งดี)

5.2. การตีความผลลัพธ์: บทบาทของมนุษย์ในวงจร AI

สิ่งที่ต้องเน้นย้ำคือ AI ไม่สามารถตั้งชื่อกลุ่มได้ เมื่อ K-Means แบ่งกลุ่มเสร็จ มันจะให้ผลลัพธ์เป็น Group 0, Group 1, Group 2 เท่านั้น การจะบอกว่ากลุ่มไหนคือ "ลูกค้าชั้นดี" เป็นหน้าที่ของมนุษย์ที่ต้องเข้าไปดูค่าเฉลี่ย (Cluster Centroids) ของแต่ละกลุ่ม

จากโค้ดตัวอย่าง มีการใช้ตรรกะ (Logic) ทางธุรกิจในการตีความค่าที่ถูกปรับสเกลแล้ว (Scaled Values) เพื่อติดป้ายกำกับ (Labeling) ดังนี้ :

5.2.1. กลุ่ม VIP: คือกลุ่มที่มีค่าเฉลี่ย Monetary > 0.8 (สูงกว่าค่าเฉลี่ยปกติมาก) และ Frequency > 0.7 (ซื้อบ่อย)

5.2.2. กลุ่ม General (ลูกค้าทั่วไป): คือกลุ่มที่มี Frequency > 0.5 (ซื้อบ่อย) แต่ Monetary < 0.4 (ยอดซื้อต่อครั้งต่ำ)

5.2.3. กลุ่ม Uncertain (ลูกค้าไม่แน่นอน): คือกลุ่มที่ไม่เข้าเกณฑ์ข้างต้น อาจเป็นลูกค้าใหม่หรือขาจร

5.3. กลยุทธ์การตลาดจากข้อมูล (Data-Driven Marketing Strategy)

เมื่อทราบ Label ของลูกค้าแล้ว ธุรกิจสามารถสร้างกลยุทธ์ที่แม่นยำ (Personalized Marketing) ได้ตามตัวอย่างในโค้ด :

5.3.1. VIP: เสนอ "สินค้าพรีเมียม" และ "บริการส่งฟรี" เพื่อรักษาความพึงพอใจและกระตุ้นยอดขายสินค้าราคาสูง

5.3.2. General: เสนอโปรโมชั่น "ซื้อ 2 แถม 1" หรือ "Flash Sale" เพื่อกระตุ้นให้ซื้อในปริมาณที่มากขึ้น (เพิ่ม Basket size)

5.3.3. Uncertain: ใช้กลยุทธ์ดึงกลับ (Win-back) เช่น "คูปองส่วนลด 30% ที่มีอายุสั้น (7 วัน)" เพื่อสร้างความเร่งด่วนให้กลับมาใช้งาน

อ้างอิง

Amazon Web Services. (n.d.). *Difference between supervised and unsupervised learning*. https://aws.amazon.com/th/compare/the-difference-between-machine-learning-supervised-and-unsupervised/

Analytics Vidhya. (2021, January 20). *In-depth intuition of K-means clustering algorithm in machine learning*. https://www.analyticsvidhya.com/blog/2021/01/in-depth-intuition-of-k-means-clustering-algorithm-in-machine-learning/

Analytics Vidhya. (2021, May 24). *K-Mean: Getting the optimal number of clusters*. https://www.analyticsvidhya.com/blog/2021/05/k-mean-getting-the-optimal-number-of-clusters/

Braze. (n.d.). *RFM segmentation*. https://www.braze.com/resources/articles/rfm-segmentation

Built In. (n.d.). *Elbow method*. https://builtin.com/data-science/elbow-method

Codecademy. (n.d.). *Clustering cheatsheet*. https://www.codecademy.com/learn/dspath-unsupervised/modules/dspath-clustering/cheatsheet

Draj0718. (2019, August 5). The math behind K-Means clustering. *Medium*. https://medium.com/@draj0718/the-math-behind-k-means-clustering-4aa85532085e

Galletti, L. (2019, April 14). K-Means from scratch. *Medium*. https://medium.com/@gallettilance/kmeans-from-scratch-24be6bee8021

GeeksforGeeks. (n.d.). *What is silhouette score?* https://www.geeksforgeeks.org/machine-learning/what-is-silhouette-score/

Google Cloud. (n.d.). *Supervised vs. unsupervised learning*. https://cloud.google.com/discover/supervised-vs-unsupervised-learning

Gultekin, H. (2020, December 1). What is silhouette score? *Medium*. https://medium.com/@hazallgultekin/what-is-silhouette-score-f428fb39bf9a

Hafeez, N. (2022, November 8). Understanding K-Means clustering. *Medium*. https://medium.com/@nerminhafeez2002/understanding-k-means-clustering-3716731cc648

IBM. (n.d.). *Supervised vs. unsupervised learning*. https://www.ibm.com/think/topics/supervised-vs-unsupervised-learning

K-means clustering. (2024, May 15). In *Wikipedia*. https://en.wikipedia.org/wiki/K-means_clustering

Lloyd, S. (1982). Least squares quantization in PCM. *IEEE Transactions on Information Theory, 28*(2), 129–137. http://www.stat.yale.edu/~hz68/Lloyd.pdf

MathWorks. (n.d.). *Kmeans*. https://www.mathworks.com/help/stats/kmeans.html

Misterfour. (2019). *K-means clustering: Optimal number of clusters*. Kaggle. https://www.kaggle.com/code/misterfour/k-means-clustering-optimal-number-of-clusters

National Institutes of Health. (2023). *On the philosophy of unsupervised learning*. PubMed Central. https://pmc.ncbi.nlm.nih.gov/articles/PMC10435961/

Reddit. (2013). *I am trying to figure out the difference between* [Online forum post]. https://www.reddit.com/r/MachineLearning/comments/1lkpz1/i_am_trying_to_figure_out_the_difference_between/

ResearchGate. (2023). *On the philosophy of unsupervised learning*. https://www.researchgate.net/publication/370181796_On_the_Philosophy_of_Unsupervised_Learning

Scikit-learn. (n.d.). *Clustering*. https://scikit-learn.org/stable/modules/clustering.html

Scikit-learn. (n.d.). *Preprocessing data*. https://scikit-learn.org/stable/modules/preprocessing.html

Scikit-learn. (n.d.). *sklearn.preprocessing.StandardScaler*. https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html

Shah, T. (2020, May 14). Determining the optimal number of clusters using the elbow method. *Medium*. https://medium.com/@tarangds/determining-the-optimal-number-of-clusters-using-the-elbow-method-how-when-where-what-and-why-cd1e94b82c1e

Stack Exchange (Mathematics). (2023). *What is the correct formula for within-cluster sum of squares?* [Online Q&A]. https://math.stackexchange.com/questions/4785701/what-is-the-correct-formula-for-within-cluster-sum-of-squares

Stack Overflow. (2021). *MinMaxScaler vs StandardScaler for scaling features* [Online Q&A]. https://stackoverflow.com/questions/68972590/minmaxscaler-vs-standardscaler-for-scaling-features

Towards AI. (n.d.). *Z-score standardization (StandardScaler)*. https://towardsai.net/p/machine-learning/z-score-standardization-standardscaler

Towards Data Science. (2020, November 23). *Clustering: How to find hyperparameters using inertia*. https://towardsdatascience.com/clustering-how-to-find-hyperparameters-using-inertia-b0343c6fe819/