

강원지역혁신플랫폼

기계학습

Machine Learning

K-평균 군집 알고리즘



▶ 학습목표

📁 K-평균 군집 알고리즘을 설명할 수 있습니다.





01 | K-평균 군집 알고리즘

K-평균 군집(K-Means Clustering) 알고리즘

⚙ K-평균 군집 알고리즘은 데이터 분석과 기계학습에서 널리 사용되는 비지도 학습 방법임

◆ 이 알고리즘은 주어진 데이터를 K개의 군집(cluster)으로 나누는 작업을 수행함

◆ 각 군집은 유사한 특성을 가진 데이터 포인트로 구성됨

◆ K-평균(means) 이름의 의미는 주어진 데이터를 K개로 군집을 하겠다는 의미임

‣ 즉 K개의 평균값을 사용하여 K개의 군집(Cluster)을 만든다는 것임

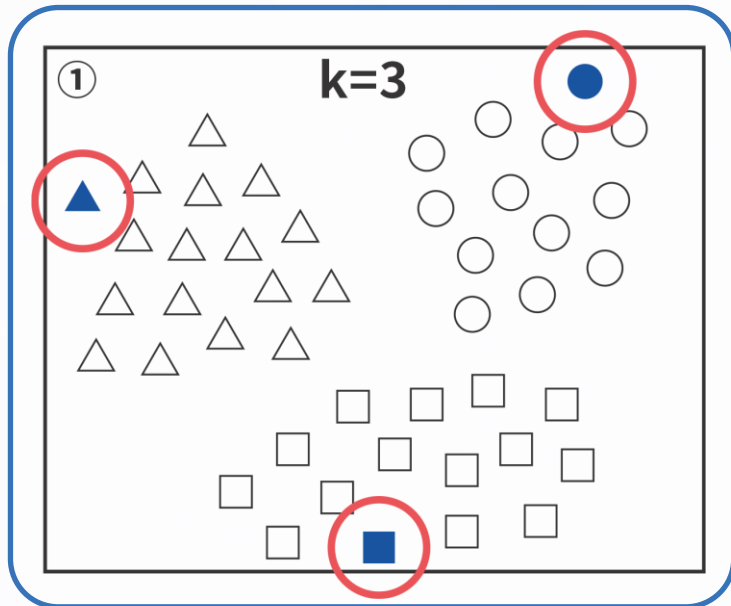
‣ 여기서 K개의 기준은 사용자가 직접 정해주어야 함



01 | K-평균 군집 알고리즘

⚙️ K-평균 군집(K-Means Clustering) 알고리즘

- ⚙️ K-평균 군집에서 **군집 수(K)**는 **사용자가 미리 정해 주어야 함**
- ✦ 여기서 **K개의 초기 중심 값**은 **자료 값 중에서 임의로 선택이 가능함**
 - 하지만, **가급적 멀리 떨어져 있는 것이 바람직함**
 - **초기 중심 값의 선정에 따라 군집 결과가 크게 달라질 수 있음**



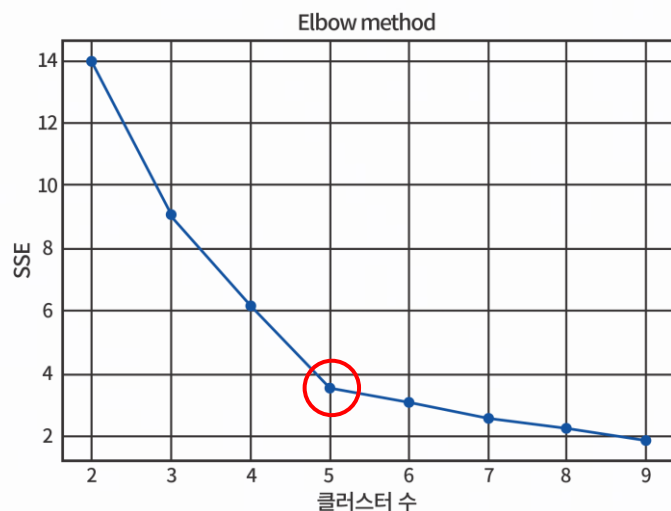


01 | K-평균 군집 알고리즘

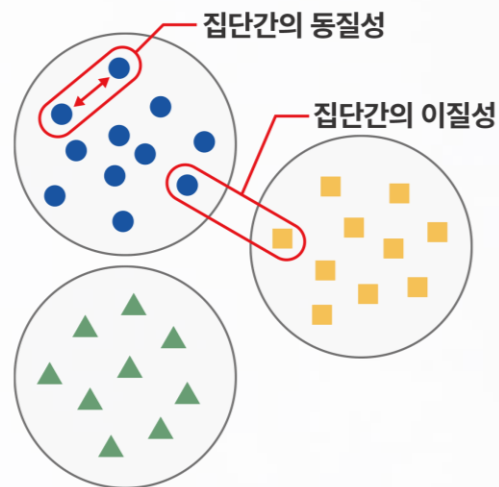
⚙ K-평균 군집(K-Means Clustering) 알고리즘

⚙️ 적정 군집 수(K)를 정하는 한 가지 방법으로 군집 수를 $K=1$ 부터 임의의 K 까지를 지정하고 군집 내 동질성과 이질성을 측정함

◆ 여기서 군집 수(K)를 늘려가면서 동질성의 증가와 이질성의 감소 기울기의 절감 지점인 엘보우(elbow) 값을 찾는 방법을 사용할 수 있음



엘보우 기법으로 K값을 찾는 예



적정 군집 수를 정하는 방법 예



01 | K-평균 군집 알고리즘

K-평균 군집(K-Means Clustering) 알고리즘

- ⚠ K-평균 군집은 군집의 매 단계마다 군집 중심으로부터 오차 제곱합(Sum of Square for Error, SSE) 최소화하는 방향으로 군집을 형성해 나가는 탐욕적(greedy) 알고리즘으로 간주될 수 있음
 - ◆ 하지만, 안정된 군집은 보장하나 전체적으로 최적값을 보장하지 못함



01 | K-평균 군집 알고리즘

K-평균 군집(K-Means Clustering) 알고리즘

△ K-평균 군집은 주어진 군집 수(K)로 각 개체(데이터)를 가까운 초기값에 할당하여 군집을 형성함

◆ 각 군집의 평균을 계산하여 군집의 중심을 갱신하는 과정을 통해
전체 데이터 세트를 상대적으로 유사한 K개의 최종 군집으로 형성함

◆ K-평균 군집 알고리즘의 절차

- 1 K값 설정
- 2 초기 중심점 선택
- 3 군집 할당
- 4 중심점 업데이트
- 5 반복 : 군집 할당과 중심점 업데이트 단계를 반복



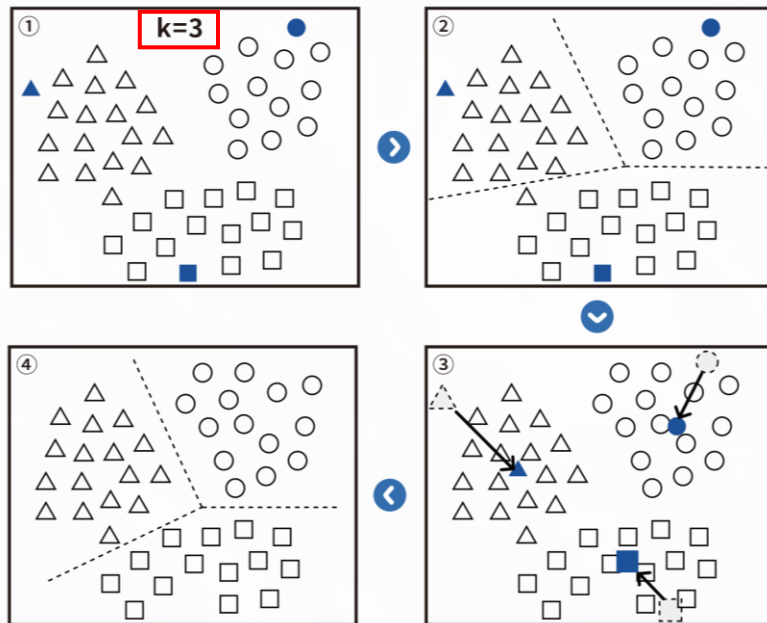
02 | K-평균 군집 알고리즘의 절차

1 K값 설정

◆ 군집의 수 K 를 선택함

➢ 사용자가 결정하며, 데이터와 목적에 따라 달라짐

– 아래 그림의 경우 $K=3$ 으로 설정



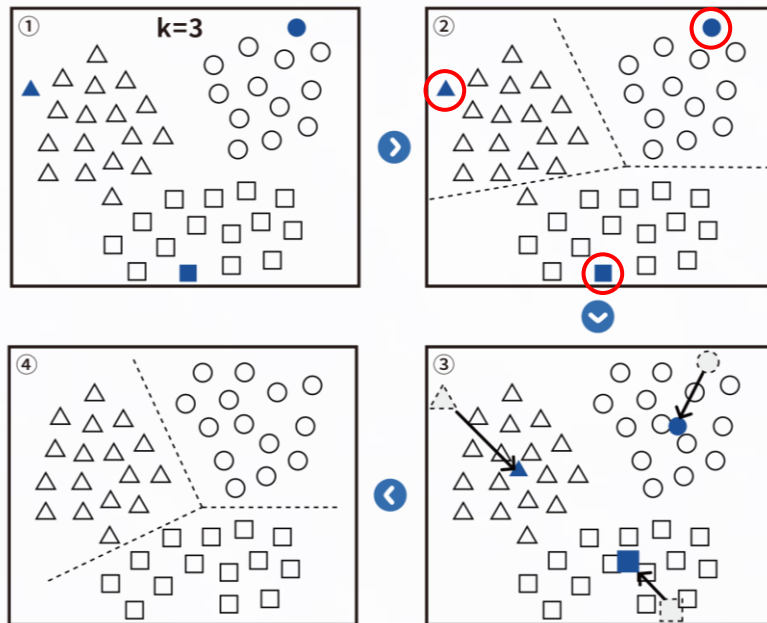
K-means 군집 알고리즘 개념



02 | K-평균 군집 알고리즘의 절차

2 초기 중심점 선택

- ◆ K개의 중심점(centroid)을 임의로 선택함
 - 일반적으로 데이터 포인트 중 K개를 무작위로 선택함



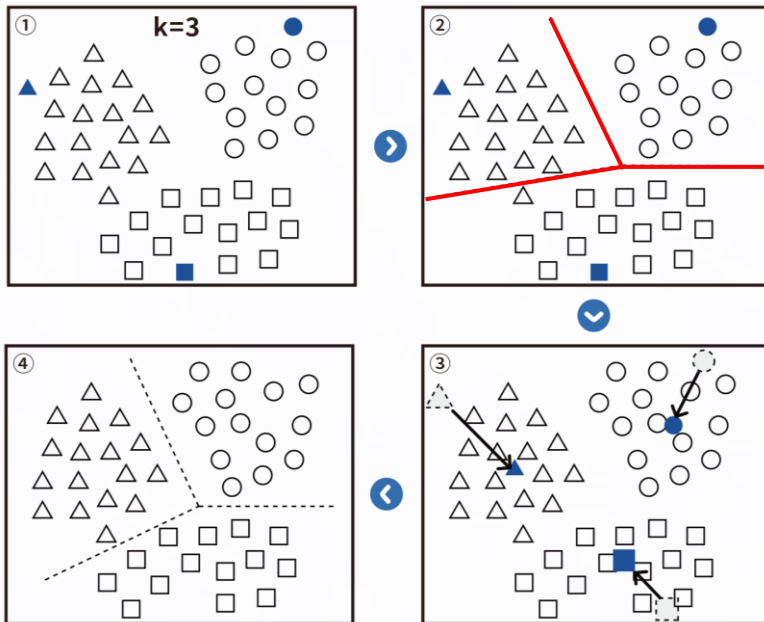
K-means 군집 알고리즘 개념



02 | K-평균 군집 알고리즘의 절차

3 군집 할당

- ◆ 각 데이터 포인트를 **가장 가까운 중심점**에 할당함
 - 거리 측정 방법으로는 일반적으로 **유클리드 거리**(Euclidean)를 사용함
 - **모든 관측치를 평균**과 **연관시켜 가장 가까운 군집**으로 할당함
 - ─ 아래의 그림에서는 3개 군집으로 나눔



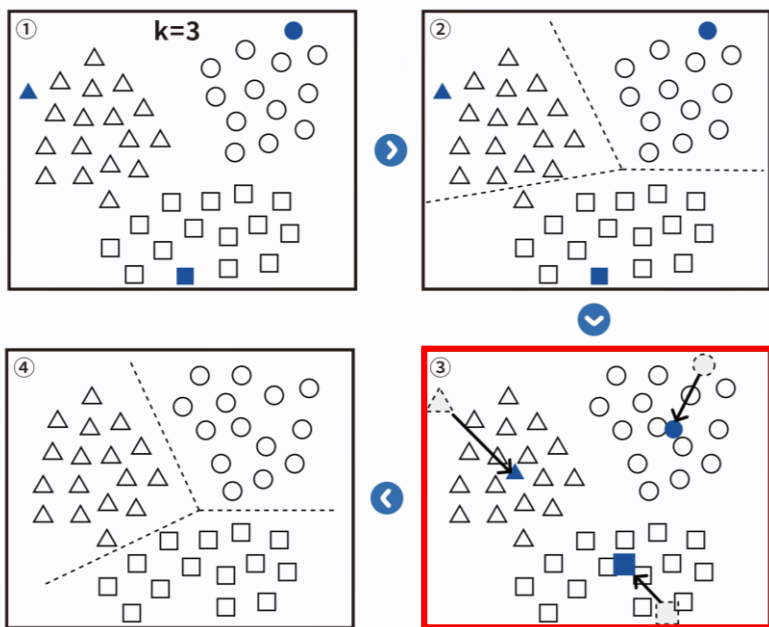
K-means 군집 알고리즘 개념



02 | K-평균 군집 알고리즘의 절차

4 중심점 업데이트

- ◆ 각 군집의 **중심점**을 해당 군집에 속한 **데이터 포인트**들의 **평균값**으로 **재계산**함
 - 즉 각 **군집 내의 새로운 평균**이 **각 K군집의 중심**이 됨



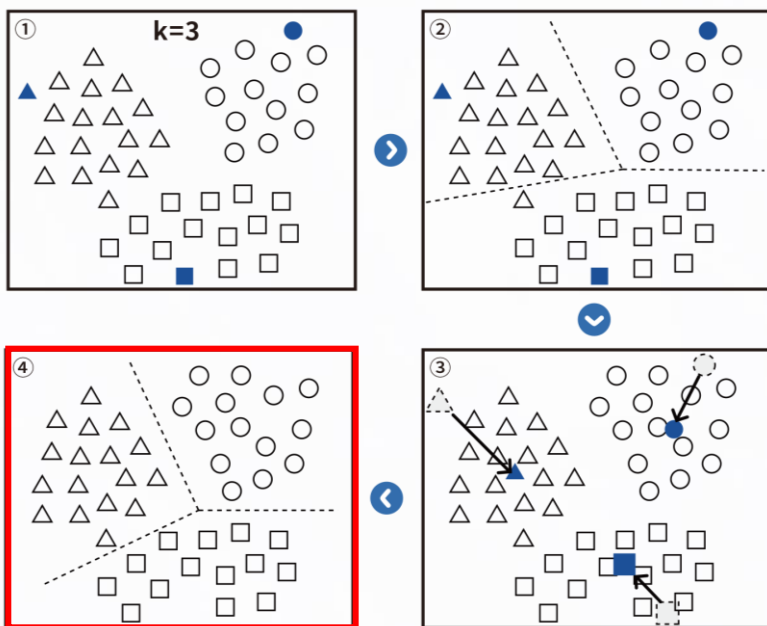
K-means 군집 알고리즘 개념



02 | K-평균 군집 알고리즘의 절차

5 반복 : 군집 할당과 중심점 업데이트 단계를 반복

- ◆ 군집 할당과 중심점 업데이트 단계를 **중심점의 변화가 없을 때까지 반복함**
 - 또는 중심점의 변화가 **일정 임계값 이하일 때까지 반복함**



K-means 군집 알고리즘 개념



03 | K값의 중요성



K값의 중요성

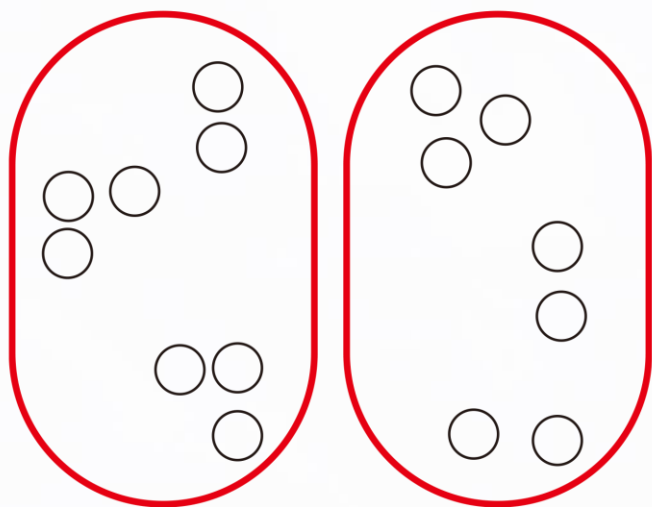
△ 주어진 데이터들을 K 를 바꾸어가면서 군집(Cluster)을 생성한다고 생각해 보자.

◆ 평균값의 기준은 적절하게 맞춰졌다고 가정함

➤ 먼저 $K=2$ 일 때를 확인해보자.

— 군집(Cluster)안의 일부분이 좀 떨어져 있어서 약간 적절해 보이지는 않음

$K=2$





03 | K값의 중요성

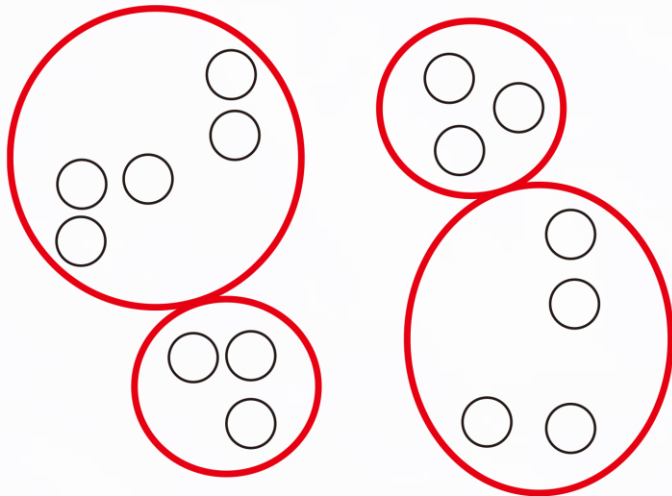


K값의 중요성

> 이번에는 K를 늘려 $K=4$ 로 조정해보자.

— $K=2$ 일 때 좀 떨어져 있었던 데이터들이 따로 클러스터로 분류되었음

$K=4$





03 | K값의 중요성

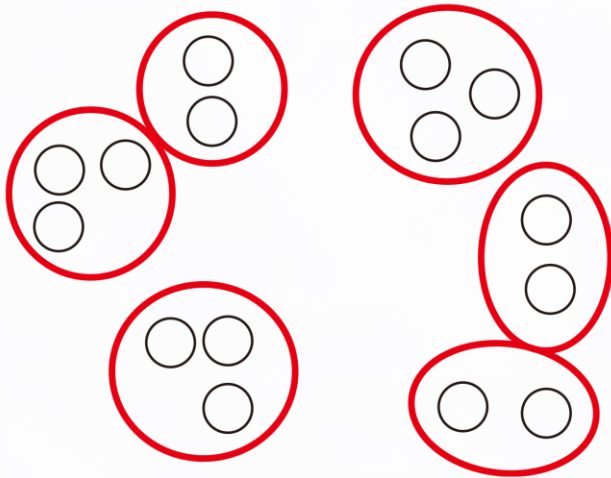


K값의 중요성

> 이번에는 K를 늘려 **K=6**로 **조정**해보자.

— 위의 두 경우와 비교하면 **각 데이터들이 좀 더 독립적**이게 **분류된 것**을 볼 수 있음

K=6





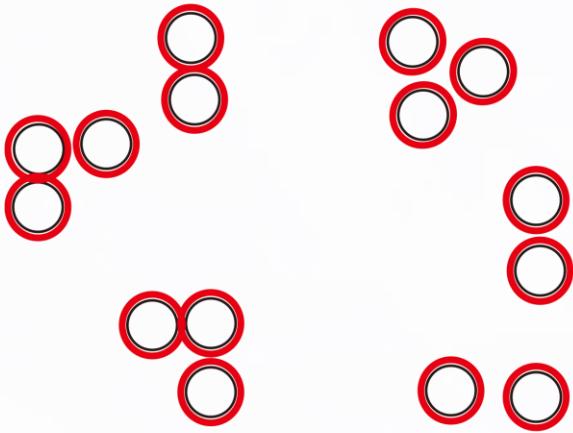
03 | K값의 중요성



K값의 중요성

- > 이번에는 (지나치게) K를 늘려 $K=15$ 로 조정해보자.
- 앞의 세 가지 경우와 비교해보면, $K=15$ 인 경우 너무 극단적으로 군집된 것을 볼 수 있음

$K=15$





03 | K값의 중요성



K값의 중요성

- ▶ 앞에서 K가 2, 4, 6, 15 일 때의 경우를 확인해 보았음
 - ─ 여기서 K의 값에 따라서 군집(Cluster)들이 변화한다는 것을 느낄 수 있었을 것임
 - ─ K=15와 같이 극단적인 경우를 제외하고, 나면 나머지 3가지 경우는 납득할 수 있는 정도로 군집(Cluster)을 생성 하였음
 - ─ 적절한 K의 선택이 군집(Cluster)의 개수 및 데이터 분류에 매우 큰 영향을 미치는 것을 알 수 있음
 - ❖ 따라서, K값이 군집(Cluster)의 신뢰도를 높일 수 있다는 점을 확인할 수 있음



04 | 군집의 기준 데이터 선정



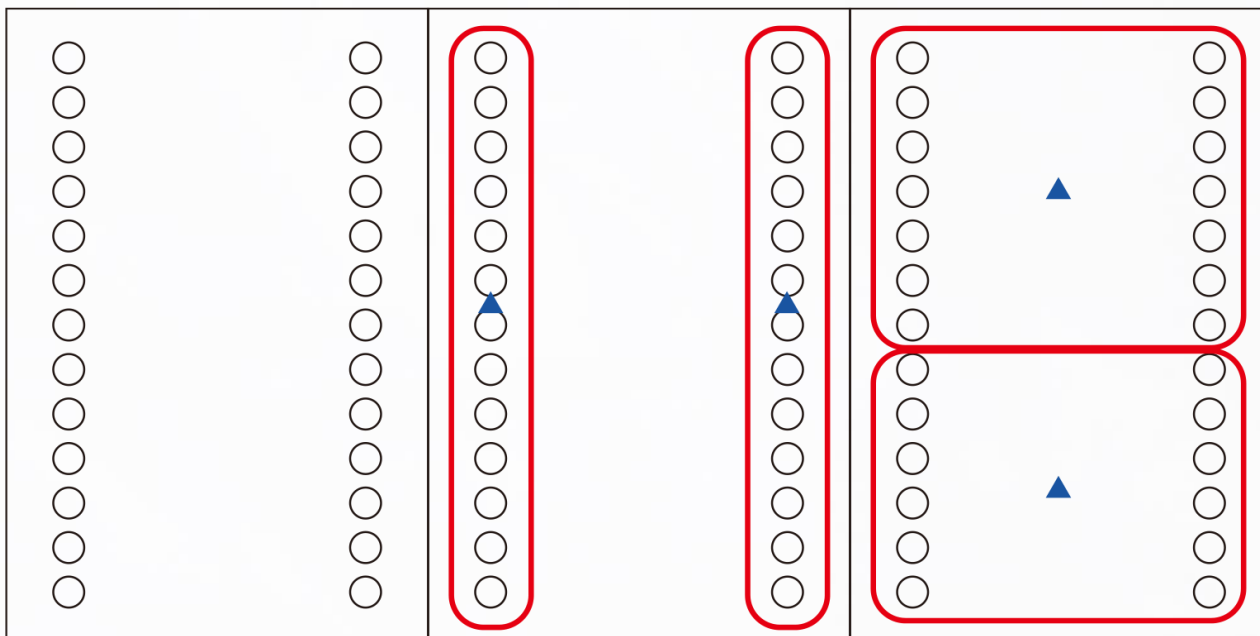
군집의 기준 데이터 선정

△ 아래의 그림에서 **왼쪽 데이터**들은 **어떻게 군집을 해야 할지** 생각해보자.

◆ 가운데 그림처럼 직관적인 군집이 가장 먼저 생각날 것임

➢ 그런데, **오른쪽 그림**과 같이 **군집**을 했다고 **틀렸다고 말할 수 있을까?**

➤ 이것 또한 **기준 데이터**로부터 **일정 거리 안에 있는 원소들**로만 **구성 되어** 있는 **정상적인 군집**임



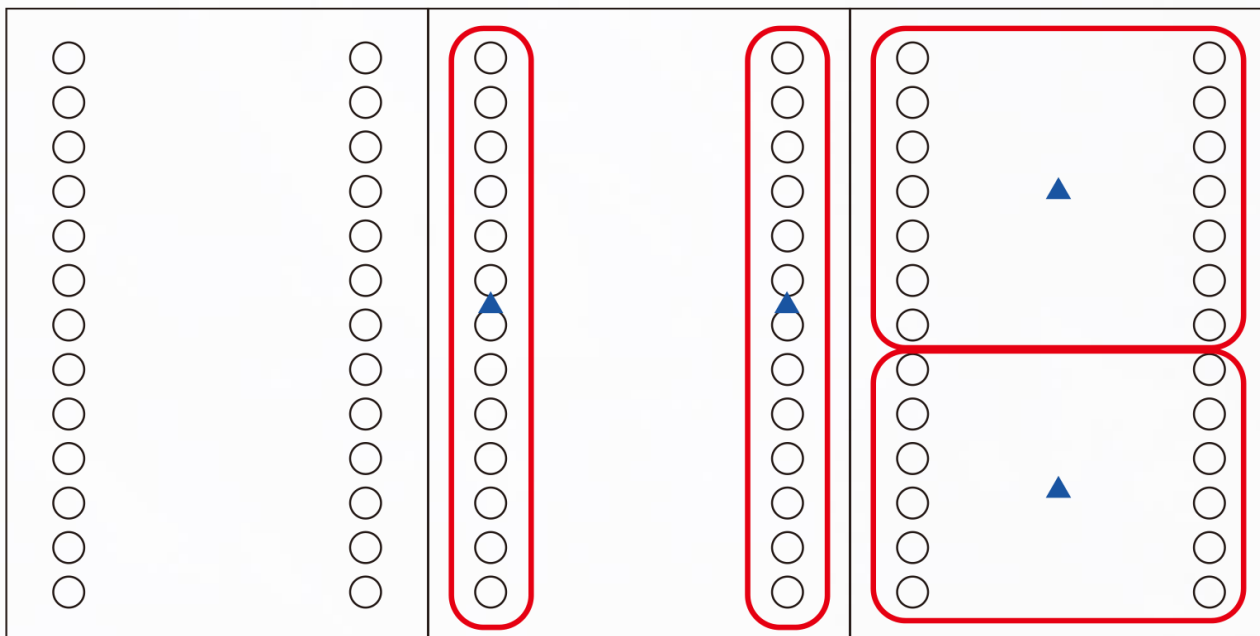


04 | 군집의 기준 데이터 선정



군집의 기준 데이터 선정

- △ 아래 그림과 같이 데이터에 따라서 어떻게 기준 데이터를 선정할지에 따라서 원소가 속하는 군집이 바뀌게 됨
- ◆ 이에 따라 군집된 모델도 확연히 달라지게 됨
 - 따라서, 기준 데이터를 적절하게 설정하는 작업은 매우 중요함
 - K-평균 군집 알고리즘을 통하여 적절한 값을 선정함





05 | K-평균 군집 기준값 조정 규칙

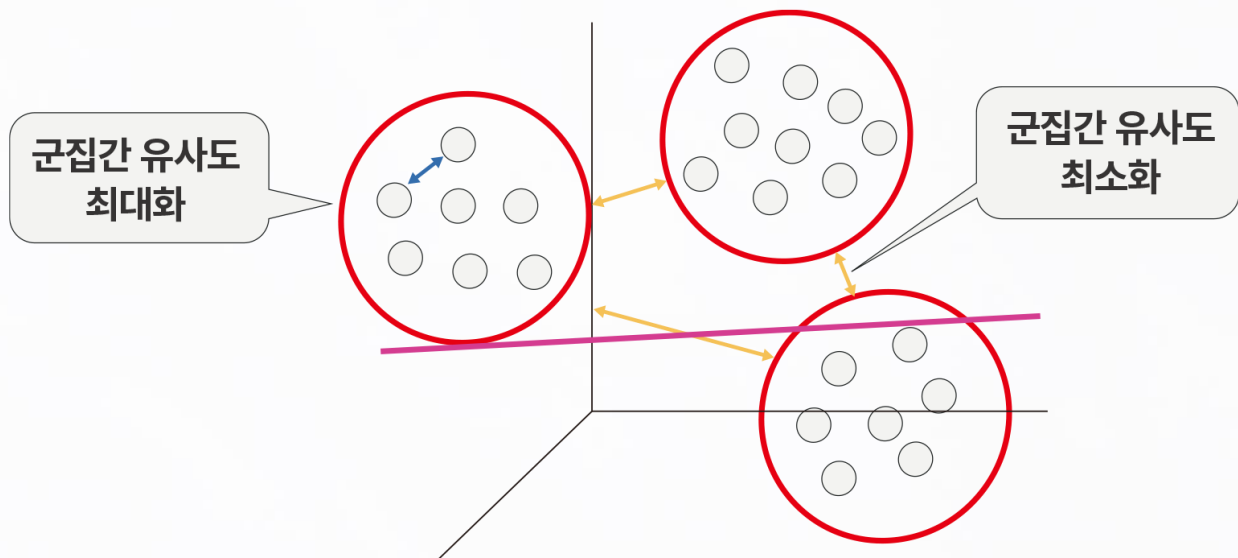
⚙️ K-평균 군집 기준값 조정 규칙

△ 기본적으로 K-평균 군집은 **두 가지 기준**으로 **기준 값을 조정**함

◆ 두 가지 기준은 아래 그림과 같음

1 군집 내의 기준과 데이터들의 거리합 최소

2 각각의 군집 간 거리 최대





05 | K-평균 군집 기준값 조정 규칙

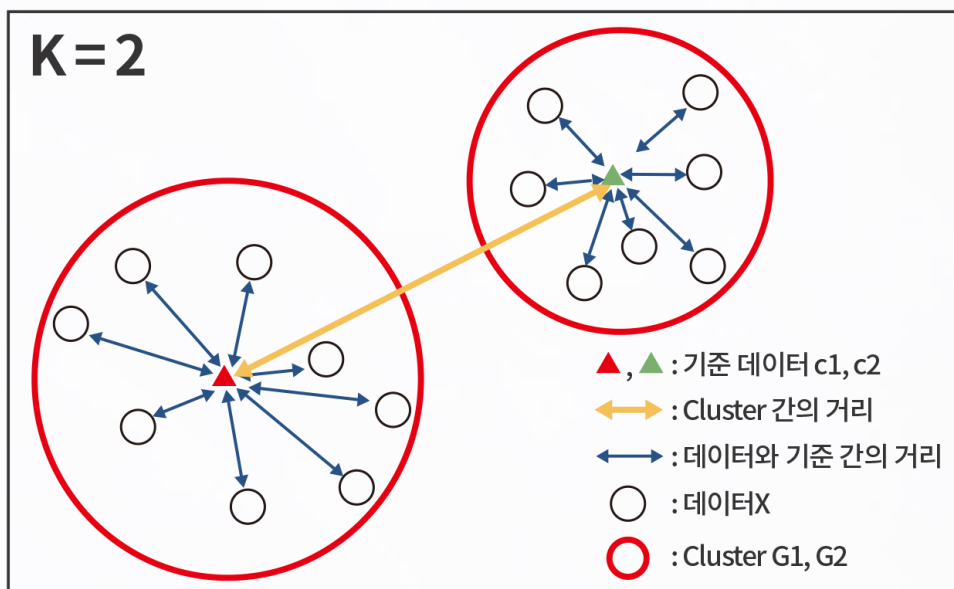
⚙️ K-평균 군집 기준값 조정 규칙

1 군집 내의 기준과 데이터들의 거리합 최소

◆ 아래 그림에서 **파란색** 화살표의 **길이의 합**을 **최소화하는 것이 목표임**

➢ 이를 **최소화**하면, **데이터의 밀도**가 **높아질 것임**(이를 응집도가 높다고 함)

$$\text{Min}(\sum_{i=1}^K \sum_{x \in G_i} \text{distance}(c_i, x))$$





05 | K-평균 군집 기준값 조정 규칙

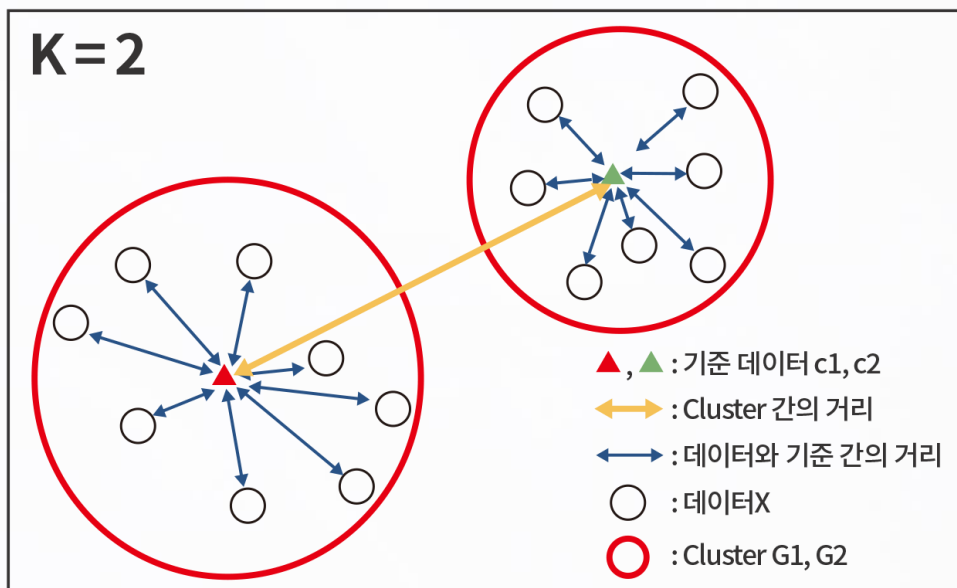
⚙️ K-평균 군집 기준값 조정 규칙

2 각각의 군집 간 거리 최대

◆ 아래 그림의 노랑색 화살표의 길이를 최대화 하는 것이 목표임

➢ 이 거리를 최대로 하면, 군집 간의 거리가 멀어져 추후 데이터를 새로이 추가할 때 오판 할 가능성을 낮춰줌

$$\text{Max}(\sum_{i=1}^K \text{distance}(c_i, c_j), i \neq j)$$





06 | 엘보우 기법



엘보우(elbow) 기법

△ K-평균 군집은 군집 내 오차 제곱합(SSE)의 값이 최소가 되도록 군집의 중심을 결정해 나가는 방법임

◆ 예를 들어 군집의 개수를 2, 3로 두고 SSE 값을 비교할 수 있을 것임

➤ 이런 식으로 군집의 개수를 늘려나가면서 계산한 SSE를 그래프로 그려봄

군집의 개수 = 2

군집의 개수 = 3

SSE = 0.1243



SSE = 0.5231

3개의 군집보다 2개의 군집이 더 적합

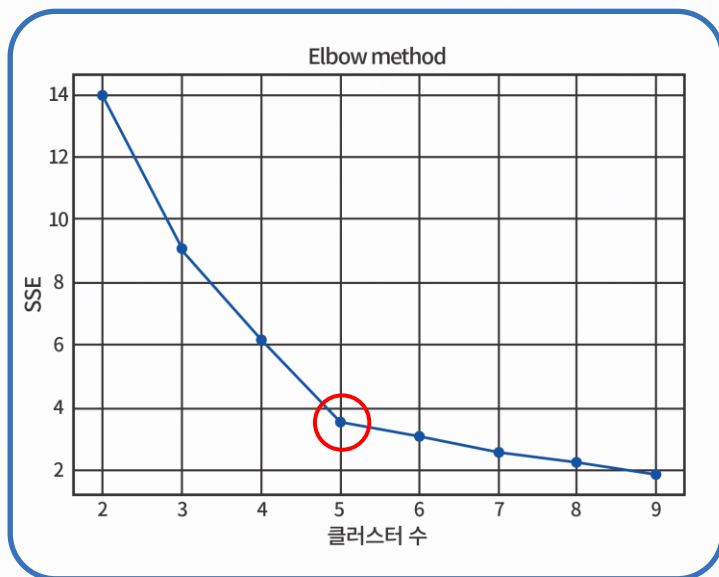


06 | 엘보우 기법



엘보우(elbow) 기법

- △ 아래의 그림처럼, **SSE의 값이 점점 줄어**들다가 **어느 순간 줄어드는 비율이 급격하게 작아지는 부분**이 생김
- ◆ 그래프 모양을 보면 **팔꿈치(elbow)**에 해당하는 바로 그 **부분이 최적의 군집 개수**가 됨
 - 아래의 그래프에서는 K=5가 됨





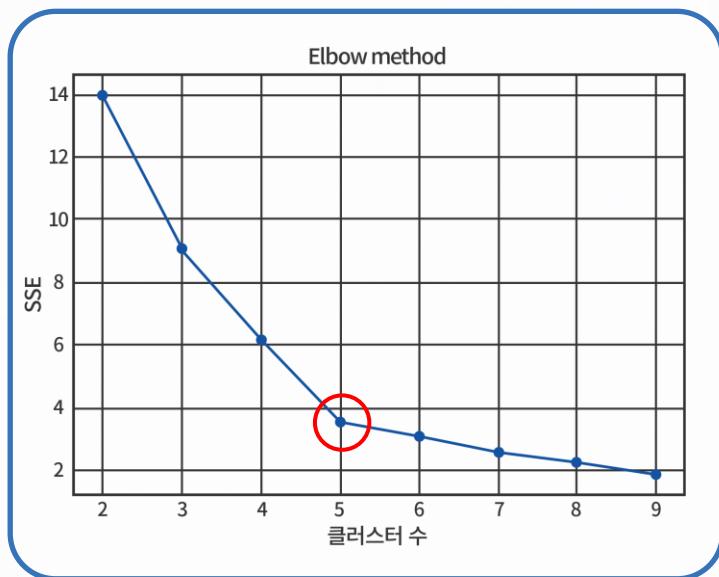
06 | 엘보우 기법



엘보우(elbow) 기법

△ 엘보우 기법은 **SSE 값을 군집의 개수**를 두고 **비교**를 한 그래프를 통해 **급격한 경사도**를 보이다가 **완만한 경사**를 보이는 **SSE값**을 보이는 부분(**팔꿈치**)에 해당하는 **군집**을 **선택하는 기법**

◆ 즉 엘보우 기법은 **SSE의 값이 최소가 되도록 K값**을 **결정하는 방법**임





06 | 엘보우 기법



[참고] 오차 제곱합 (Sum of squares error, SSE)

$$SSE = \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

◆ 여기서 y 는 실제값, \hat{y} 는 예측값이다.



07 | K-평균 군집 알고리즘의 장단점



K-평균 군집(K-means clustering) 알고리즘의 장·단점 비교

구분	설 명
장 점	<ul style="list-style-type: none">❖ 알고리즘이 단순하며, 빠르게 수행되어 기법 적용이 용이하다.❖ 계층적 군집보다 많은 양의 자료를 다룰 수 있다.❖ 개체들 간의 거리측정과 군집 수(K), 초기 중심점만 주어지면 바로 분석을 적용할 수 있다.❖ 기법의 역사가 길어서 다양한 프로그래밍 언어에서 사용될 수 있다.❖ 다양한 형태의 데이터에 적용이 가능하다.
단 점	<ul style="list-style-type: none">❖ 임의의 초기점(중심점) 할당으로 인해 최적의 군집을 찾지 못할 수도 있다.❖ 초기 군집 수(K)에 대한 임의의 판단이 필요하다.❖ 연속형 변수의 거리 측도만 다룰 수 있다.❖ 잡음(노이즈)이나 이상값에 영향을 많이 받는다.❖ 볼록한 형태가 아닌(non-convex) 군집(예를 들어, U-형태)이 존재할 경우에는 성능이 떨어진다.❖ 사전에 주어진 목적이 없으므로 결과 해석이 어렵다.