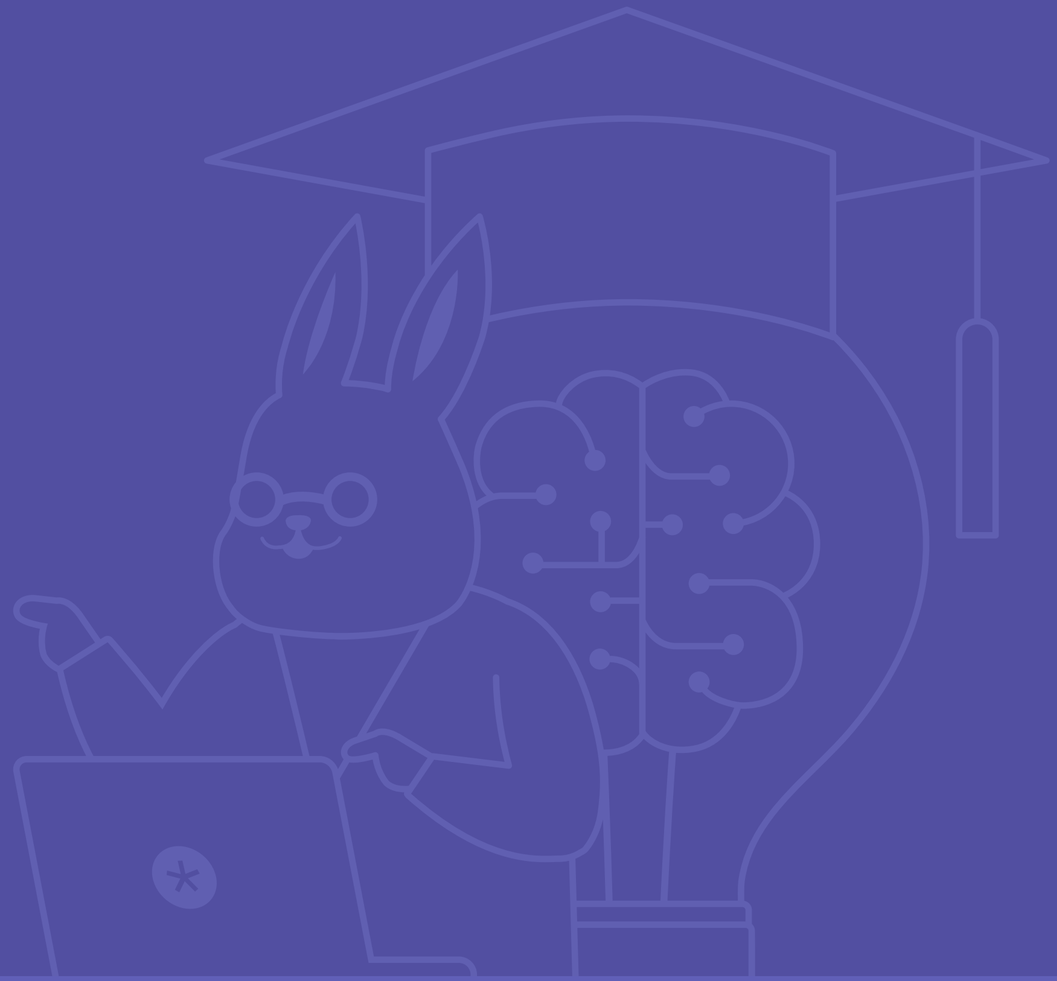


머신러닝 심화

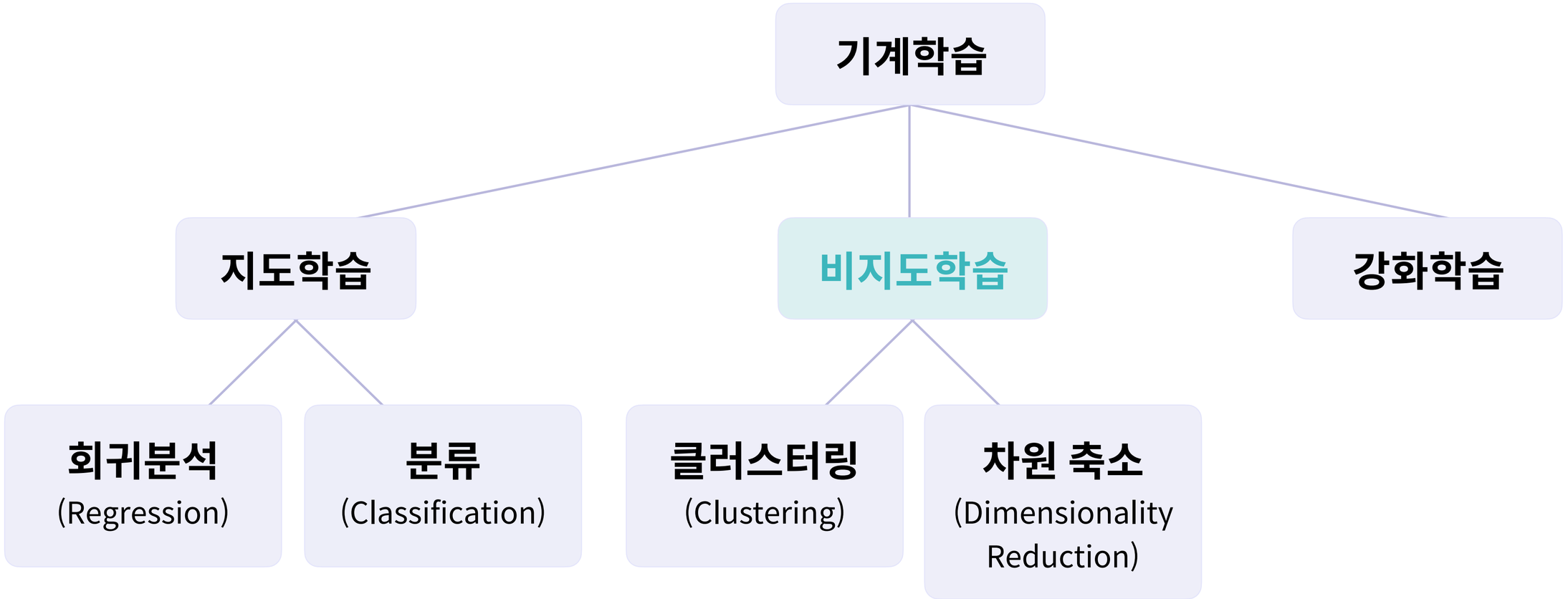
3장 비지도 학습

Contents

01

비지도 학습

머신러닝(기계 학습) 분야


기계학습
지도학습
비지도학습
강화학습
회귀분석
(Regression)
분류
(Classification)
클러스터링
(Clustering)
차원 축소
(Dimensionality Reduction)

지도 학습(Supervised Learning)

얻고자 하는 답(Y)으로 구성된 데이터를 학습

- **회귀**

데이터를 잘 설명하는 선을 찾아 미래 **결과값**을 예측

- **분류**

주어진 데이터가 **어떤 클래스**에 속할 지 여부 예측

X	Y
평균 기온(℃)	아이스크림 판매량(만개)
10	40
13	52.3
20	60.5
25	80

풍속(m/s)	지연 여부
2	No
4	Yes
3	No
1	No

비지도 학습(Unsupervised Learning)

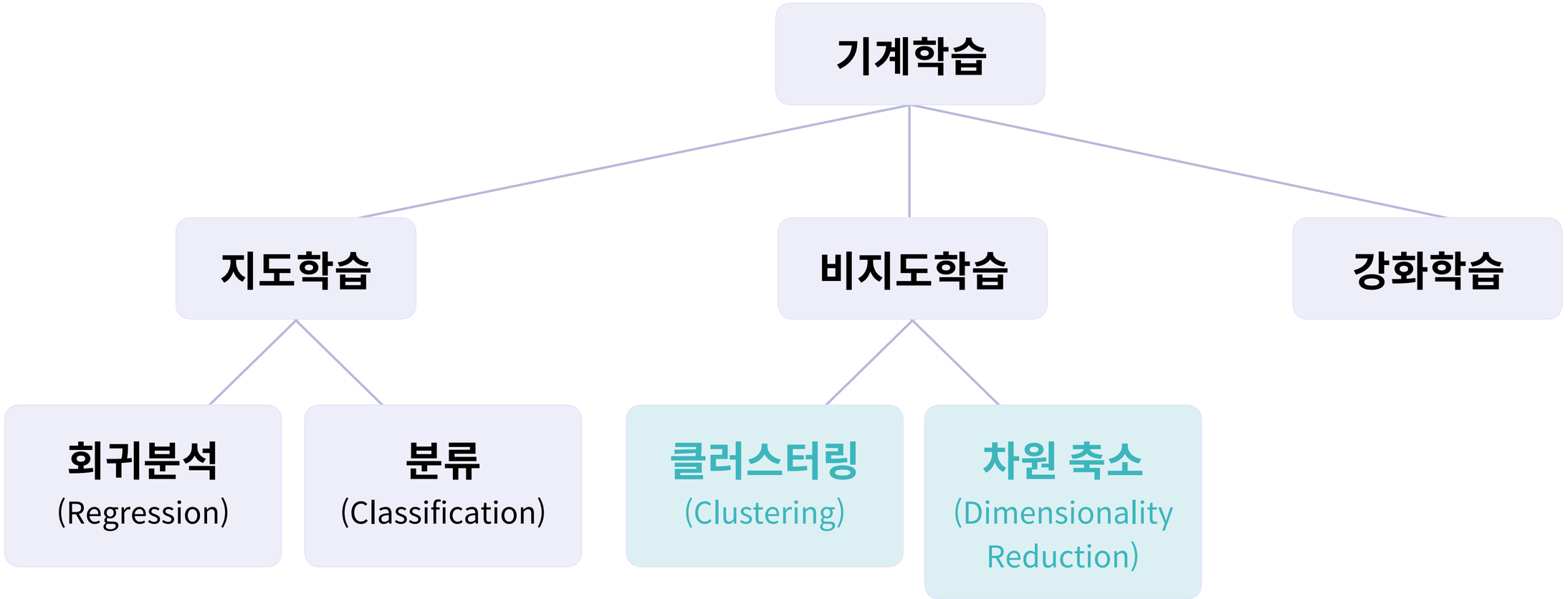
그렇다면 정답이 **없는** 데이터가 주어질 경우엔?

정답이 주어지지 않는 데이터 내에서
숨겨진 구조를 파악하는 **비지도 학습**

• 고객 별 구매 상품 개수 데이터

index	one piece	short skirt
고객1	2	0
고객2	1	1

비지도 학습 대표 분야


기계학습
지도학습
비지도학습
강화학습
회귀분석
(Regression)
분류
(Classification)
클러스터링
(Clustering)
차원 축소
(Dimensionality Reduction)

클러스터링과 차원 축소

클러스터링	차원 축소
각 개체의 그룹 정보(정답) 없이 유사한 특성을 가진 개체끼리 **군집화**하는 것	고차원 데이터의 **차원을 축소**하여 데이터를 더욱 잘 설명할 수 있도록 함

→ 각각의 알고리즘에 대해 더 자세히 알아보자!

02

클러스터링(Clustering)

02 클러스터링(Clustering)

가정해보기

인터넷 쇼핑몰 마케터라고 가정하기

고객 별 구매 상품 개수 데이터를 활용하여
유사한 고객 집단으로 세분화하고자 한다면?

- **고객 별 구매 상품 개수 데이터**

index	one piece	short skirt
고객1	2	0
고객2	1	1

02 클러스터링(Clustering)

문제 정의와 해결 방안

• 문제 정의

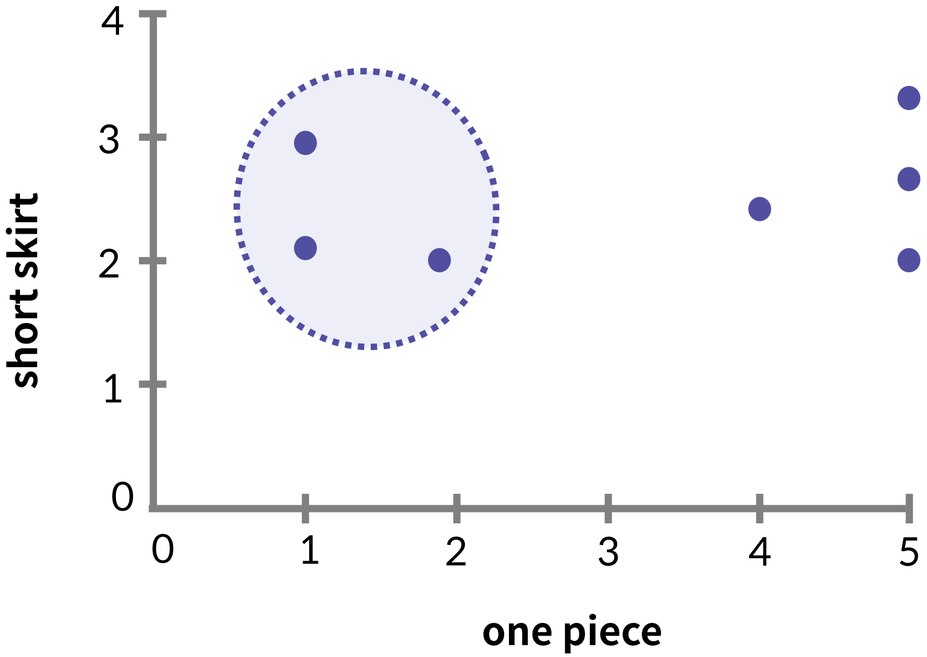
유사한 특성을 지닌 고객을
동일한 그룹으로 묶으면 어떨까?

데이터 : 고객 별 구매 상품 개수 데이터

목표 : 유사한 특성을 지닌 고객 **그룹화**하기

• 해결 방안

클러스터링(Clustering) 알고리즘



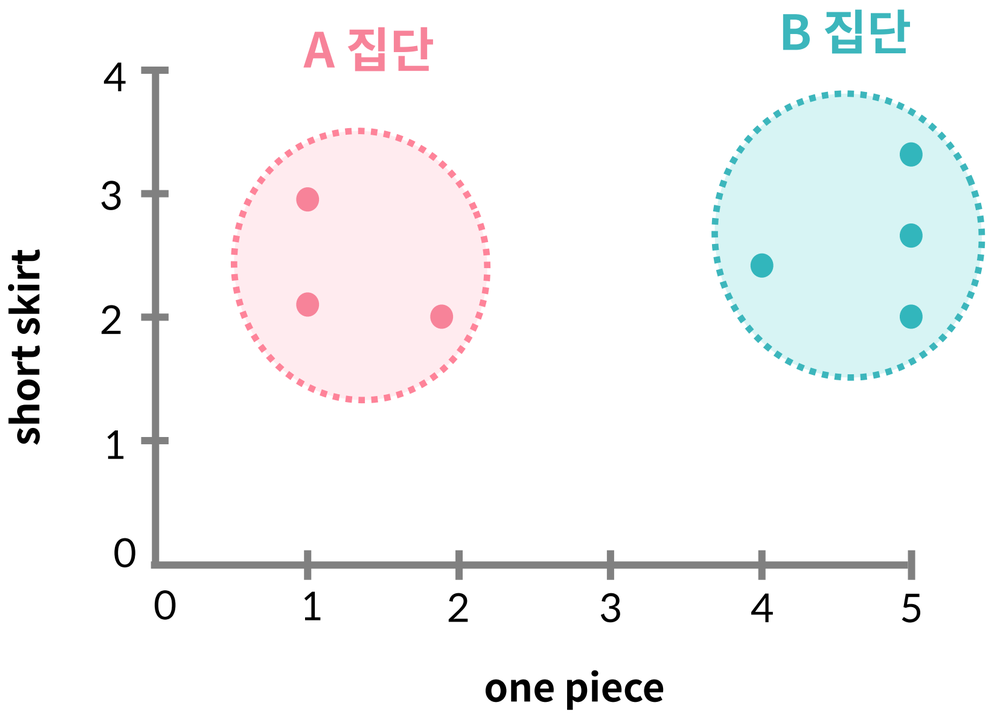
02 클러스터링(Clustering)

클러스터링(Clustering)이란?

각 개체의 그룹 정보(정답) 없이
유사한 특성을 가진 개체끼리 **군집화**하는 것

클러스터링 종류

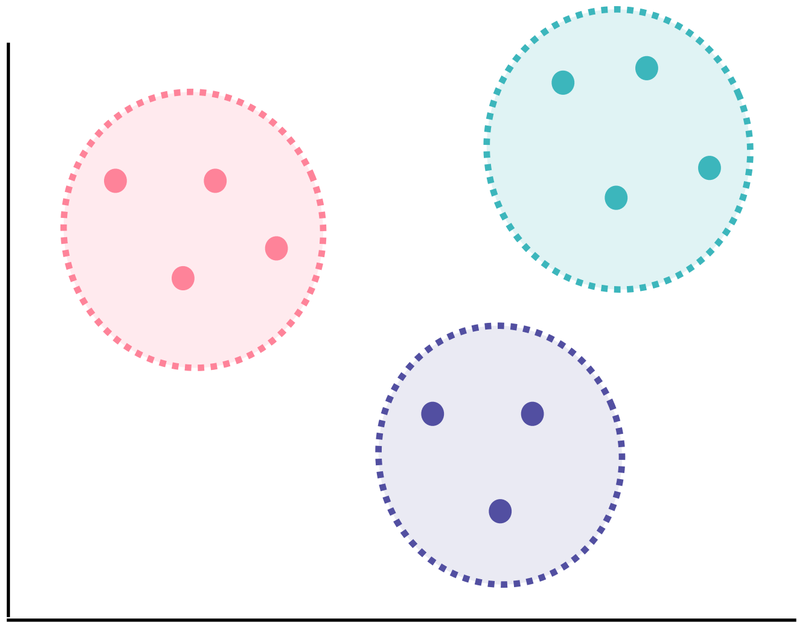
1. Hard Clustering
2. Soft Clustering



Hard Clustering

특정 개체가 집단에 **포함되는지 여부**
클러스터에 **속한다(1), 속하지 않는다(0)**으로 표현

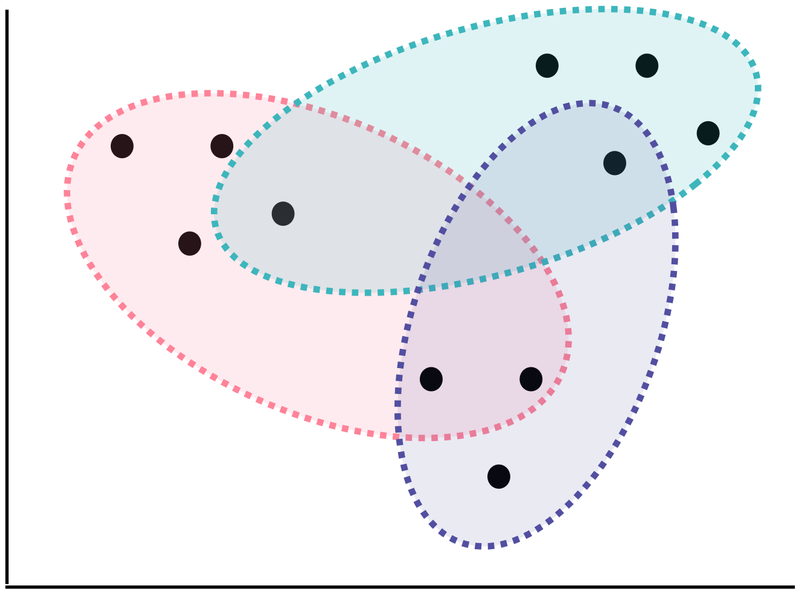
K-means Clustering 알고리즘이 이에 해당함



Soft Clustering

특정 개체가 집단에 얼마나 포함되는지 정도
클러스터에 **속하는 정도**로 표현

Gaussian Mixture Model 알고리즘이 이에 해당



02 클러스터링(Clustering)

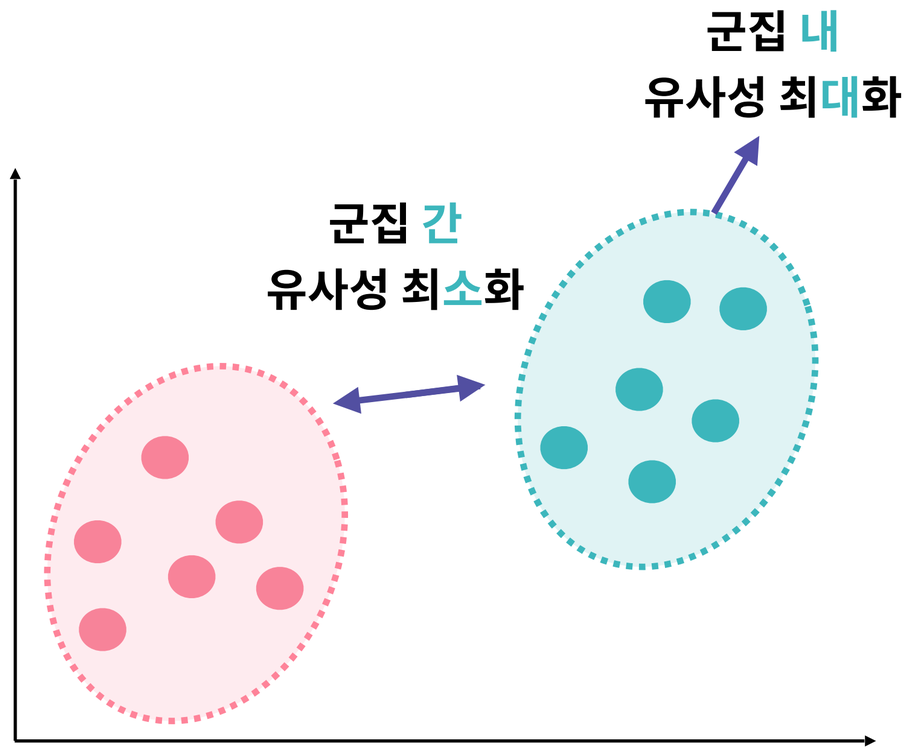
클러스터링 목표

(1)군집 간 유사성 최소화

다른 군집 간 데이터 간에는 서로 비슷하지 않게

(2) 군집 내 유사성 최대화

동일 군집 내 데이터 간에는 서로 비슷하게



03

K-means Clustering

문제 정의와 해결 방안

- 문제 정의

100만 명 이상인 고객의 구매 상품 데이터를 활용하여 고객을 군집화 하고자 한다면?

즉, **대용량 데이터**를 군집화하고자 한다면?

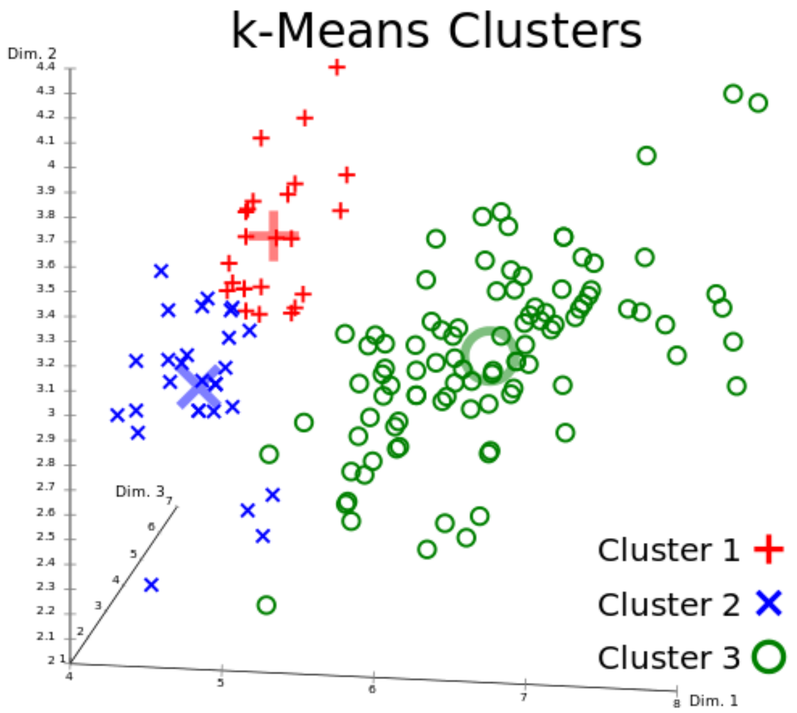
- 해결 방안

K-means 클러스터링

K-means Clustering 이란?

제공된 데이터를 **K개로** 군집화하는 알고리즘

군집화할 개수 K는 직접 설정해야 하는
하이퍼 파라미터

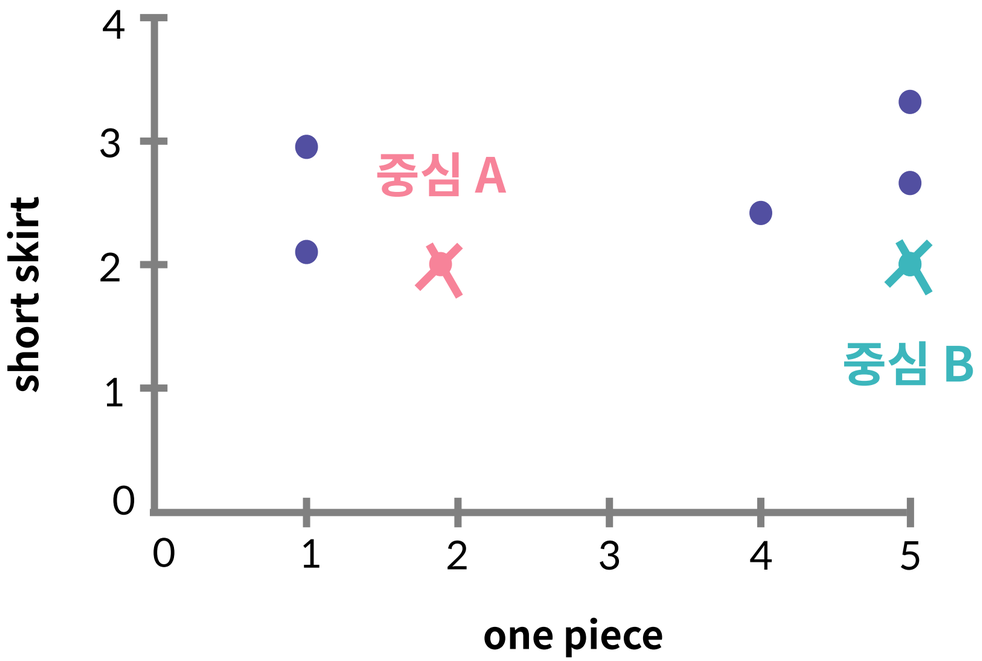


K-means Clustering 원리 -(1)

K-means Clustering 의 군집화 과정

1. 데이터셋 중에서 K개를 랜덤하게 뽑아 해당 데이터를 중심으로 함

초기화 단계에서만 진행

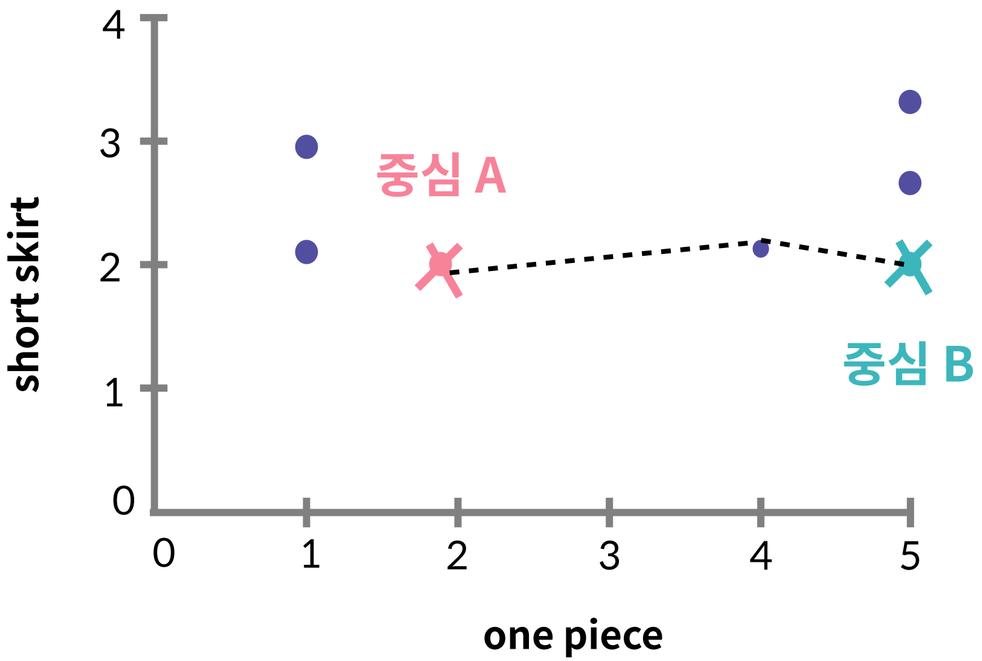


K-means Clustering 원리 -(2)

K-means Clustering 의 군집화 과정

2. 모든 **데이터**에 대해서 아래 과정 반복

각 클러스터의 중심과 자신(해당 데이터)을 비교하고, 가장 **가까운 클러스터**를 저장함

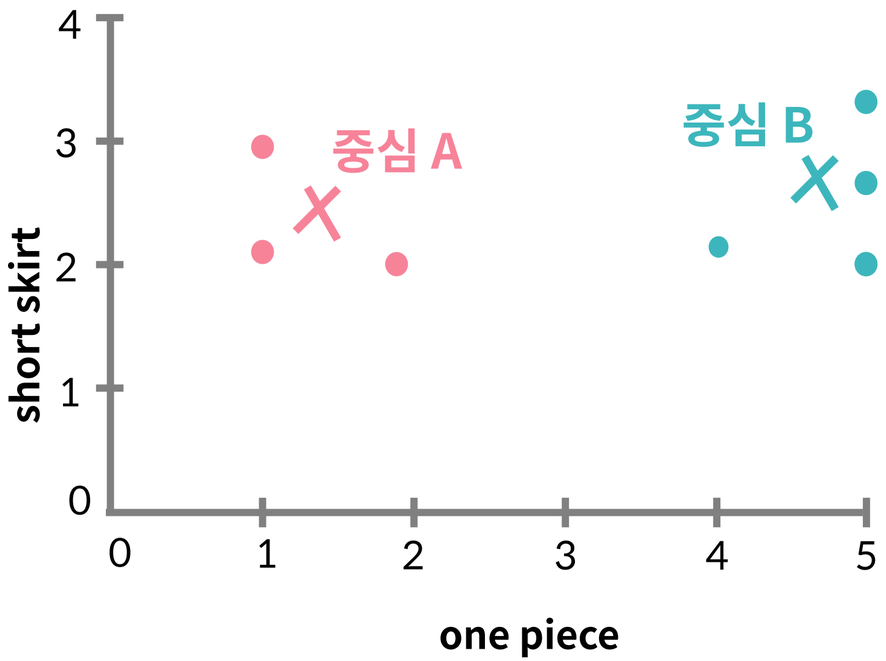


K-means Clustering 원리 -(3)

3. 모든 **클러스터**에 대해서 아래 과정 반복

> 자신(해당 클러스터)에 할당된
> 데이터들의 중심을 계산하고,
> 계산된 중심을 새로운 중심으로 설정

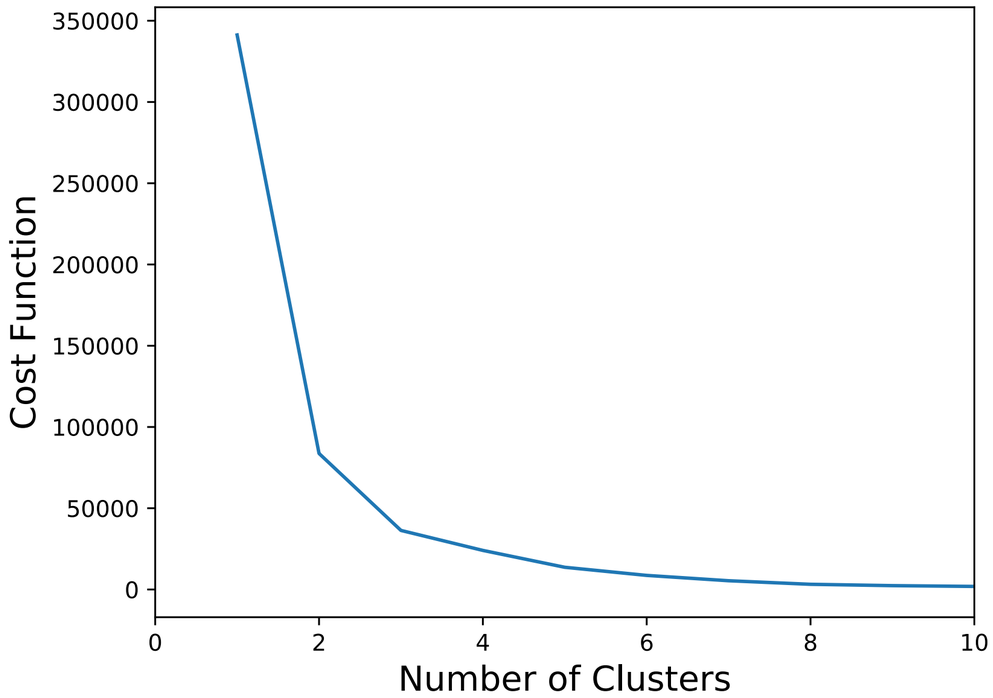
설정되는 중심의 **변화가 없을 때까지**
2,3번 과정 반복



최적의 군집 개수 K 구하기

Elbow Method

다양한 K값을 시도해보고,
비용 함수 그래프가 꺾이는 부분
즉, 클러스터 수를 증가시켜도
별 효과가 없는 지점의 K 선택



K-means Clustering 특징 및 활용

- 랜덤 초기값 설정으로 인해 데이터의 분포가 독특한 경우 원하는 결과 나오지 않을 가능성
- 시간 복잡도가 가벼워 많은 계산량이 필요한 대용량 데이터에 적합
- 실제 문제에 적용할 때는 여러 번 클러스터링을 수행해 가장 빈번히 등장하는 군집에 할당

04

Gaussian Mixture Model(GMM)

04 Gaussian Mixture Model(GMM)

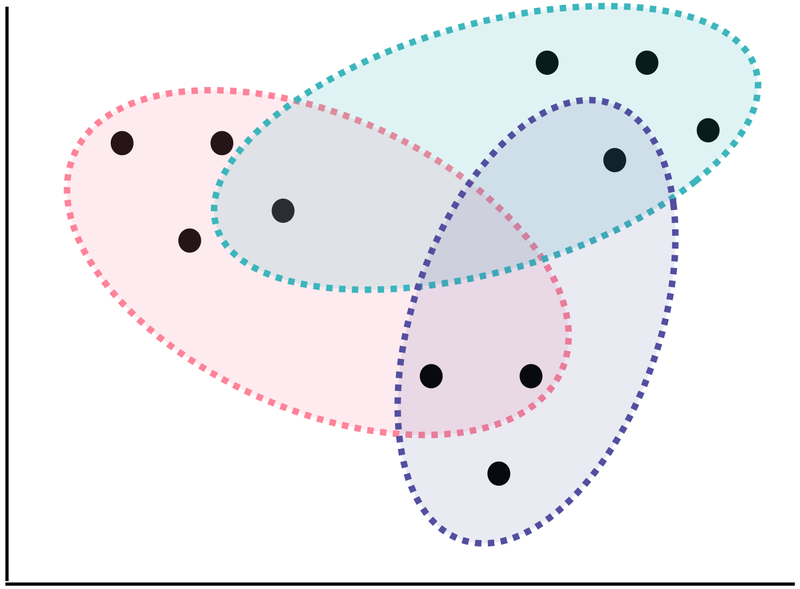
문제 정의와 해결 방안

- 문제 정의

클러스터에 속하는 정도를
표현하는 클러스터링 알고리즘은 없을까?

- 해결 방안

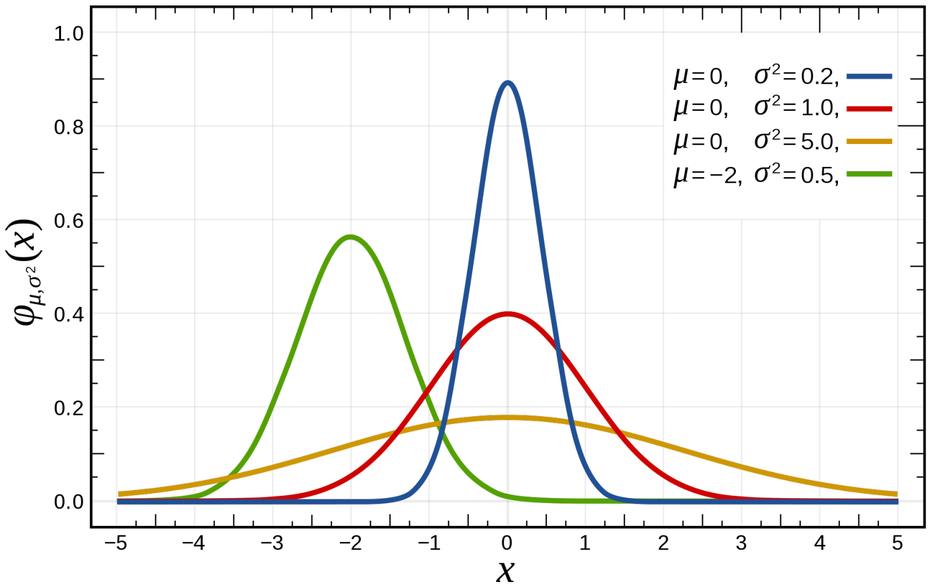
확률을 통해 데이터 유사성을 측정하는
Gaussian Mixture Model(GMM) 알고리즘



Gaussian Mixture Model(GMM)

전체 데이터의 **확률분포**가 여러 개의 **정규분포 (Normal Distribution)**의 조합으로 이루어져 있다고 가정하고,

각 분포에 속할 확률이 높은 데이터끼리 클러스터링 하는 방법



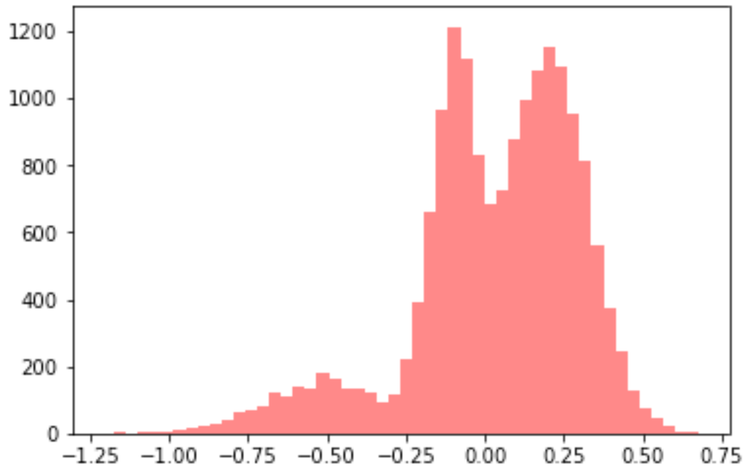
04 Gaussian Mixture Model(GMM)

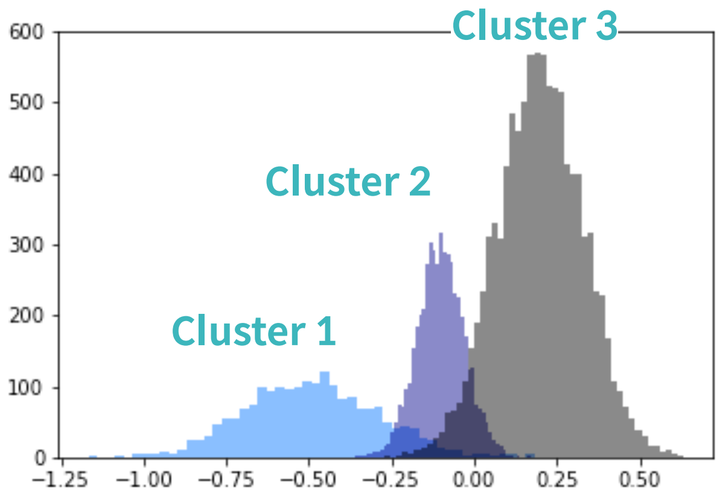
GMM 원리

(1) 학습 데이터의 분포와 유사한
k개의 정규 분포를 추출

(2) 개별 데이터가 **어떤 정규 분포에 속하는지** 결정

(k개의 정규 분포는 k개의 클러스터에 해당됨)





/* elice */

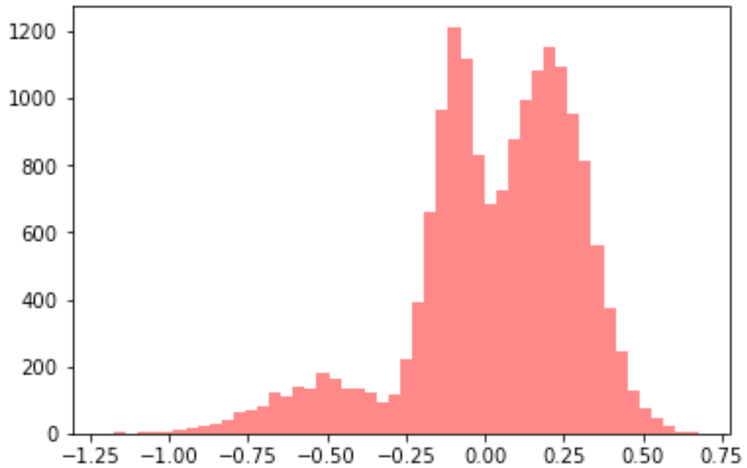
04 Gaussian Mixture Model(GMM)

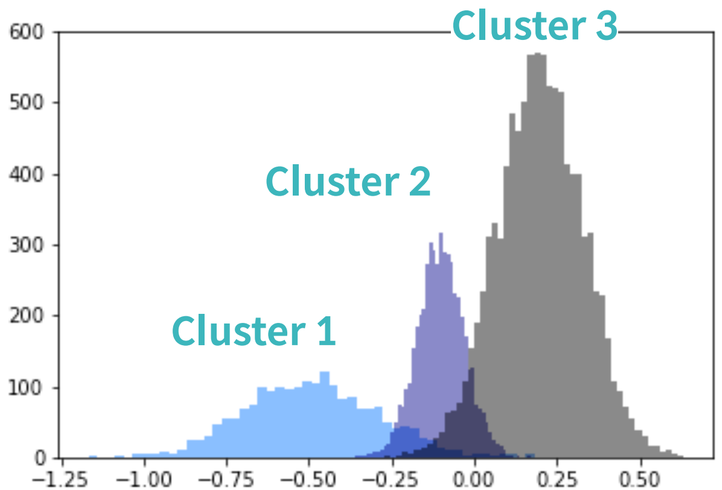
GMM 원리

구하고자 하는 것

1. **클러스터의 형태**
즉, 각 클러스터의 평균과 분산
2. 데이터가 **어떤 클러스터에 속하는지**

동시에 구하는 것은 **불가능**함!
두 가지 과정으로 분리해서 수행





GMM 진행 과정 - (1)

(1) 각 클러스터마다 해당 클러스터가 선택될 **확률**과
평균, 그리고 **분산** 을 랜덤하게 초기화

클러스터가 선택될 확률 = 데이터가 **어떤 클래스에 속하는지**
평균, 그리고 분산 = **클러스터의 형태**

GMM 진행 과정 - (2)

(2) 변화가 없을 때까지 모든 **데이터**에 대해서 아래 과정 반복

- 클러스터가 선택될 확률, 평균, 분산이 주어짐
- 각 데이터가 어느 클러스터에 들어가는지 계산

→ 클러스터의 형태(평균, 분산)가 주어졌을 때 **데이터가 어떤 클러스터에 들어갈지** 계산

GMM 진행 과정 - (3)

(3) 변화가 없을 때까지 모든 **클러스터**에 대해서 아래 과정 반복

- 각 데이터가 어느 클러스터에 들어가는지가 주어짐
- 각 클러스터마다 선택될 확률, 평균, 분산 계산

→ 데이터가 어떤 클러스터에 들어갈지 주어졌을 때 **클러스터의 형태(평균, 분산)** 계산

04 Gaussian Mixture Model(GMM)

클러스터링 타당성 평가

정답이 없기 때문에 실제값과 예측값의 오차 혹은 단순 정확도 지표로 평가할 수 없음
군집 간 거리, 군집의 지름, 군집의 분산을 고려하여 클러스터링 목표 달성 여부 확인

대표적 평가 지표

Dunn Index, 실루엣(Silhouette) 지표

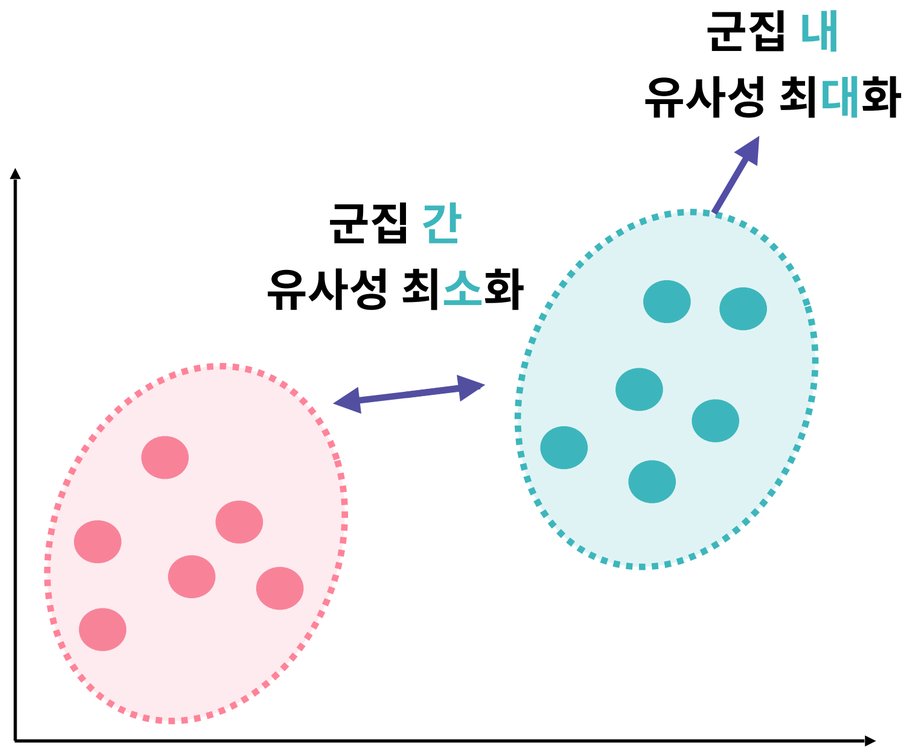
클러스터링 목표 복습하기

(1)군집 간 유사성 최소화

다른 군집 간 데이터 간에는 서로 비슷하지 않게

(2) 군집 내 유사성 최대화

동일 군집 내 데이터 간에는 서로 비슷하게

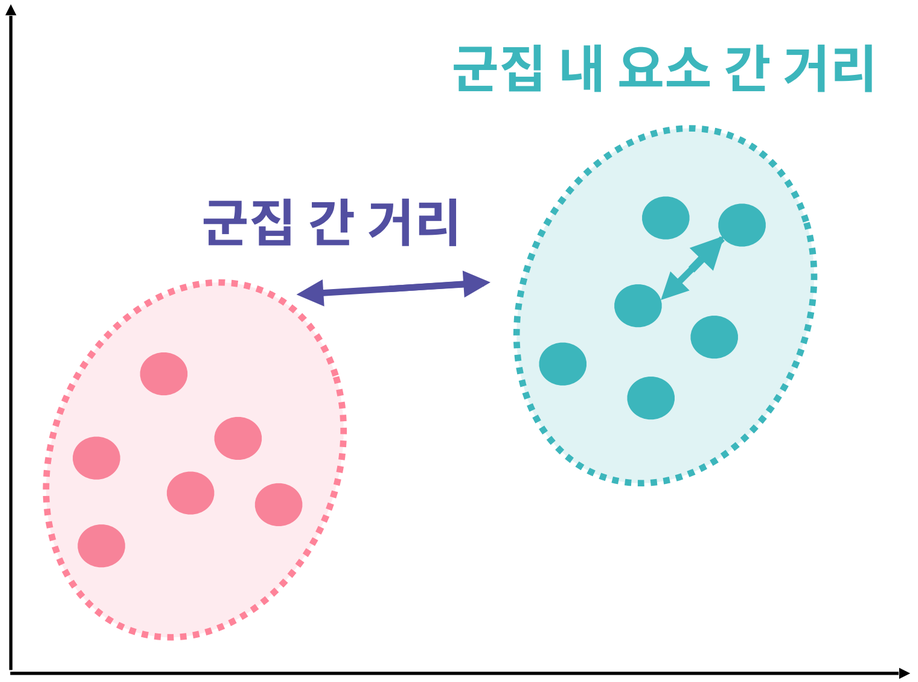


Dunn Index

분자값이 클수록 군집 간 거리가 크고,
분모값이 작을수록 군집내의 데이터들이 모여있다는 뜻

해당 지표는 **클수록 높은 성능**을 의미함

$$\frac{\text{군집 간 거리의 최소값}}{\text{군집 내 요소 간 거리의 최대값}}$$



04 Gaussian Mixture Model(GMM)

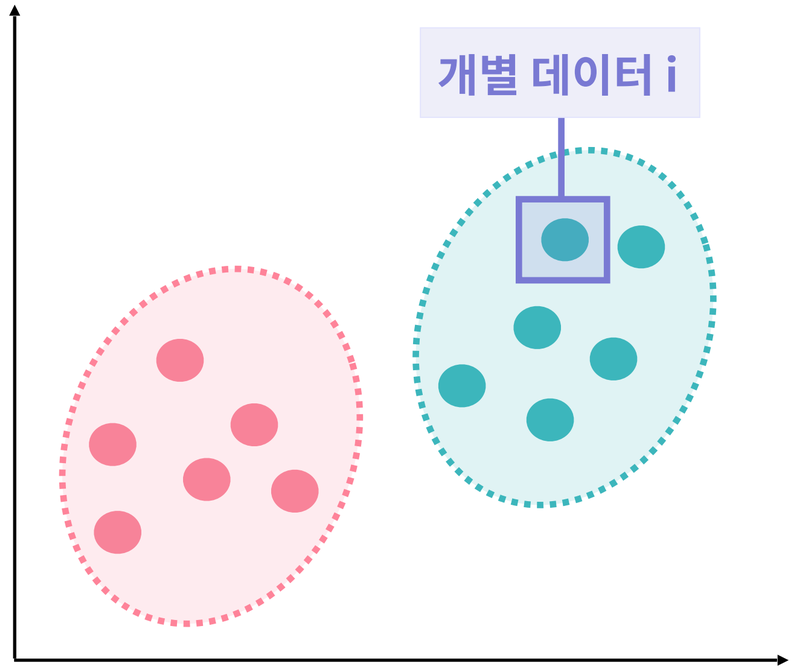
실루엣(Silhouette) 지표

얼마나 잘 군집화 되었는 지에 대한 정량적 평가

클러스터의 밀집 정도 계산

-1부터 1 사이의 값을 가짐, 1에 가까울 수록 높은 성능

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



$a(i)$: i번째 데이터가 속한 군집과 가장 가까운 이웃군집을 택해서 계산한 값

$b(i)$: i번째 데이터와 같은 군집에 속한 요소들 간 거리의 평균

05

차원 축소(Dimensionality Reduction)

05 차원 축소(Diemensionality Reduction)

가정해보기

현재 보유하고 있는 학습 데이터의 **변수가 100만개**라고 가정하기
만약, 학습 속도 개선과 데이터 압축을 하고자 한다면?

05 차원 축소(Diemensionality Reduction)

문제 정의와 해결방안

- 문제 정의

데이터 변수의 개수를 줄여 데이터의 차원을 줄이면 어떨까?

데이터 : 100만개 변수를 가진 데이터
목표 : 학습 속도 개선 및 데이터 압축하기

- 해결 방안

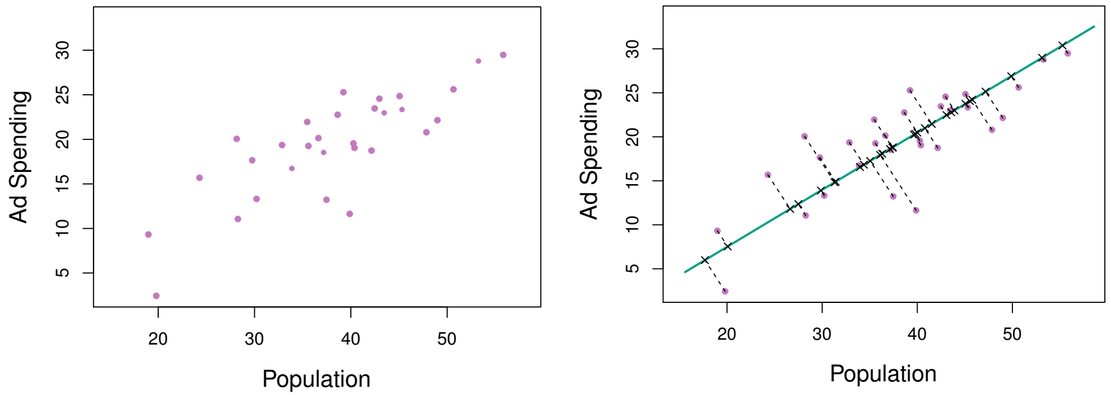
차원 축소(Dimensionality Reduction) 알고리즘

05 차원 축소(Diemensionality Reduction)

차원 축소(Dimensionality Reduction)란?

고차원의 데이터를 **저차원으로 줄이는** 알고리즘

엄청나게 많은 변수를 가지고 있는 고차원의 데이터에서는 **차원의 저주**가 발생할 가능성이 높아짐

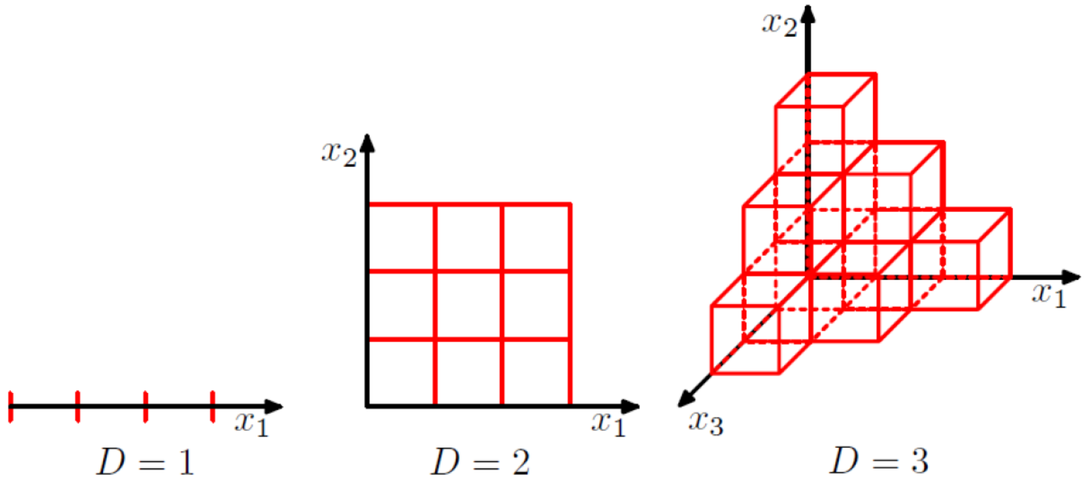


05 차원 축소(Diemensionality Reduction)

차원의 저주

차원이 높을 수록
학습에 요구되는 **데이터의 개수도 증가**함

고차원일 때 적은 개수의 데이터로만
차원을 표현하는 경우 과적합(Overfitting)
발생 가능



05 차원 축소(Diemensionality Reduction)

차원 축소 필요성

차원의 저주 발생 방지와
모델 학습 속도 및 성능 향상을 위한 차원 축소 알고리즘

차원 축소 알고리즘

1. 주성분 분석(Principle Component Analysis)
2. t-SNE(t-Stochastic Neighborhood Embedding)

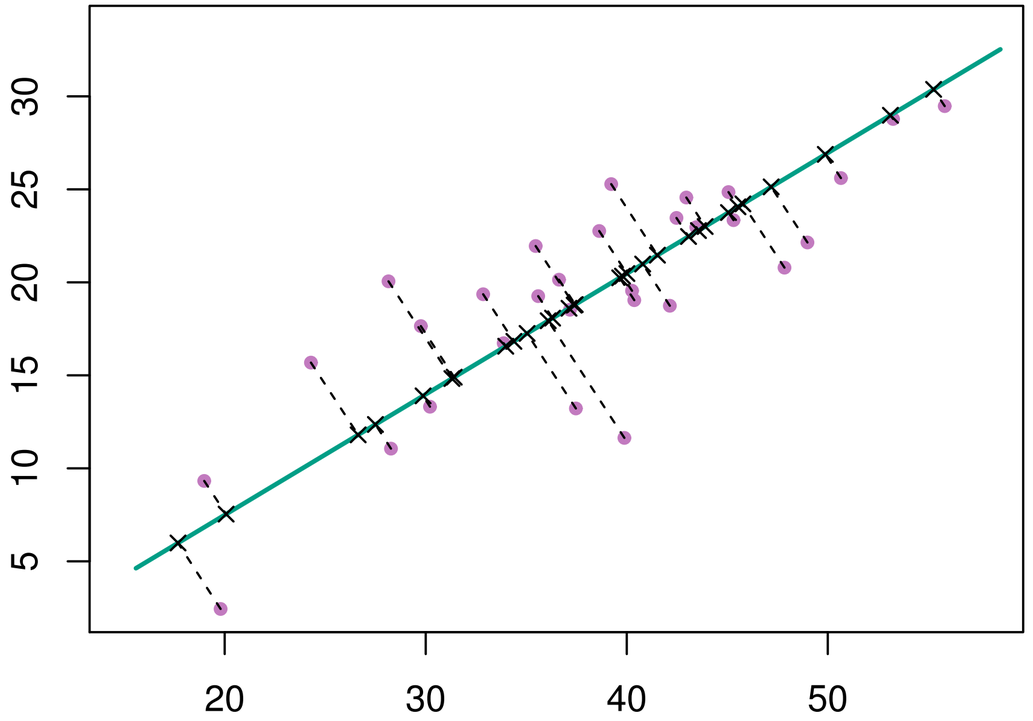
06

주성분 분석(PCA)

주성분 분석(Principle Component Analysis)

고차원 데이터를 가장 잘 설명할 수 있는
주성분(Principle Component)를 찾는 방법

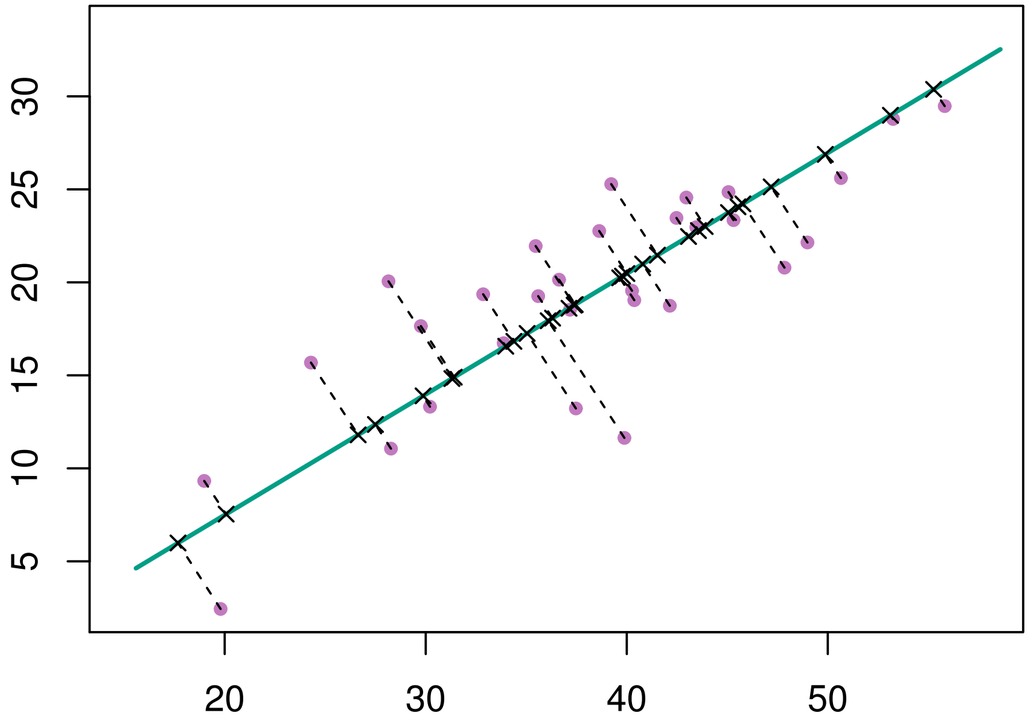
차원을 축소하면서도 원본 데이터(original data)의 특징을 가지고 있을 수 있도록 함



주성분 분석(Principle Component Analysis)

원본 데이터의 특징을 가지고 있다

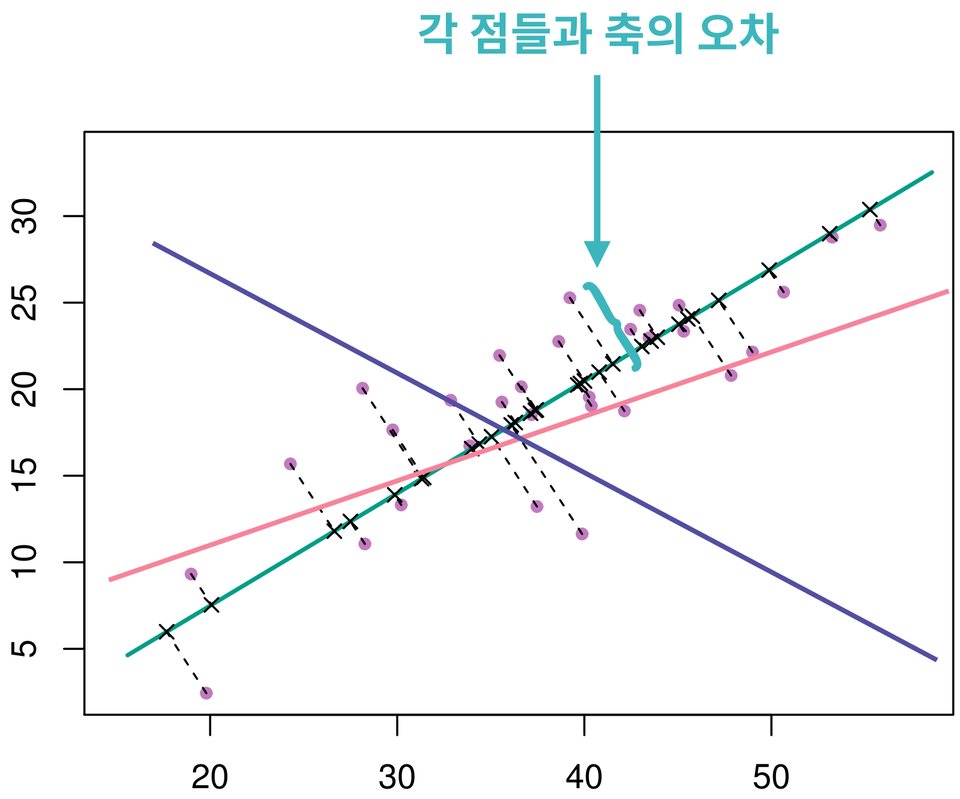
= 원본 데이터와의 **차이를 최소화 해야 함**
= 원본 데이터와의 **차이를 최소화 하는 축 (주성분)을 찾아야 함**



주성분 분석 원리

2차원 데이터를 1차원으로 차원 축소할 경우

여러 갈래의 축을 확인해보며 **각 점들과 축의 오차가 가장 작은 축**을 중심으로 데이터를 모은다.

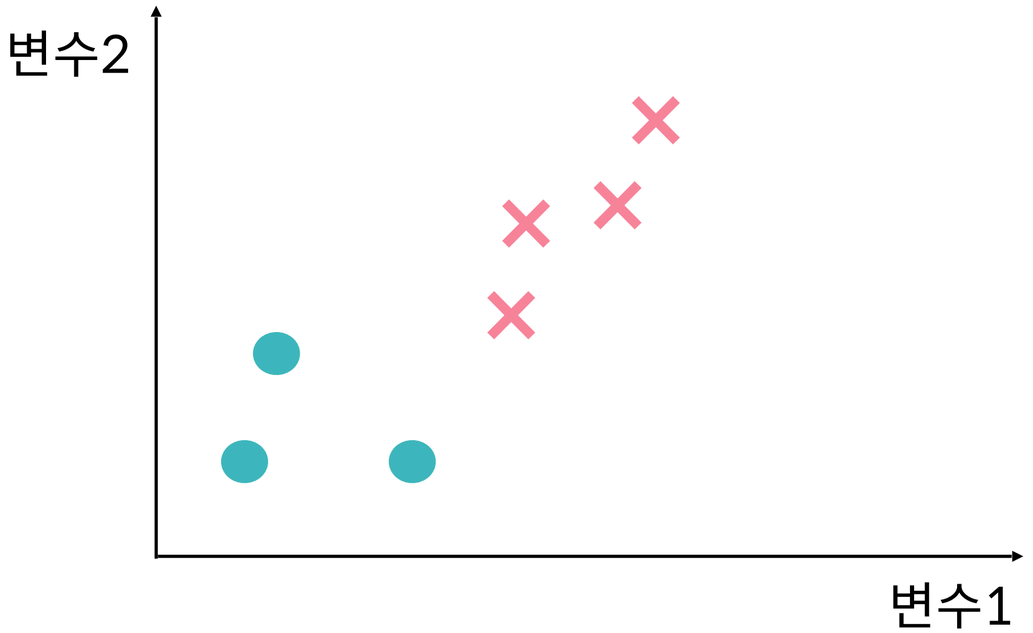


주성분 분석 원리

2차원 데이터를 1차원으로 차원 축소할 경우

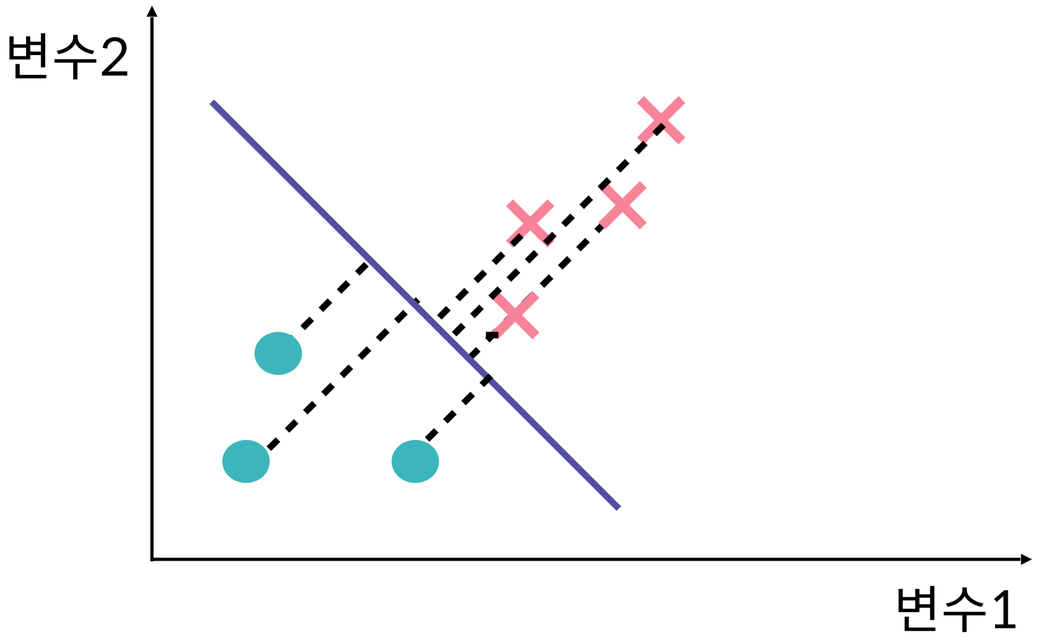
예시 데이터 분포가 우측과 같을 때,
2차원 데이터를 1차원으로 축소하는 과정

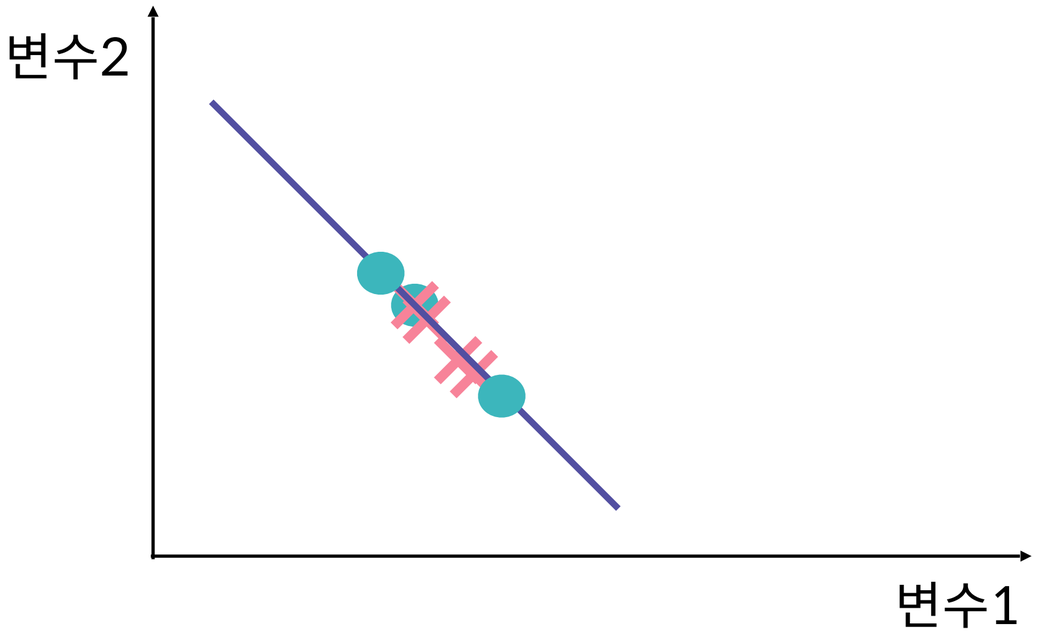
- 데이터 산점도



주성분 분석 원리

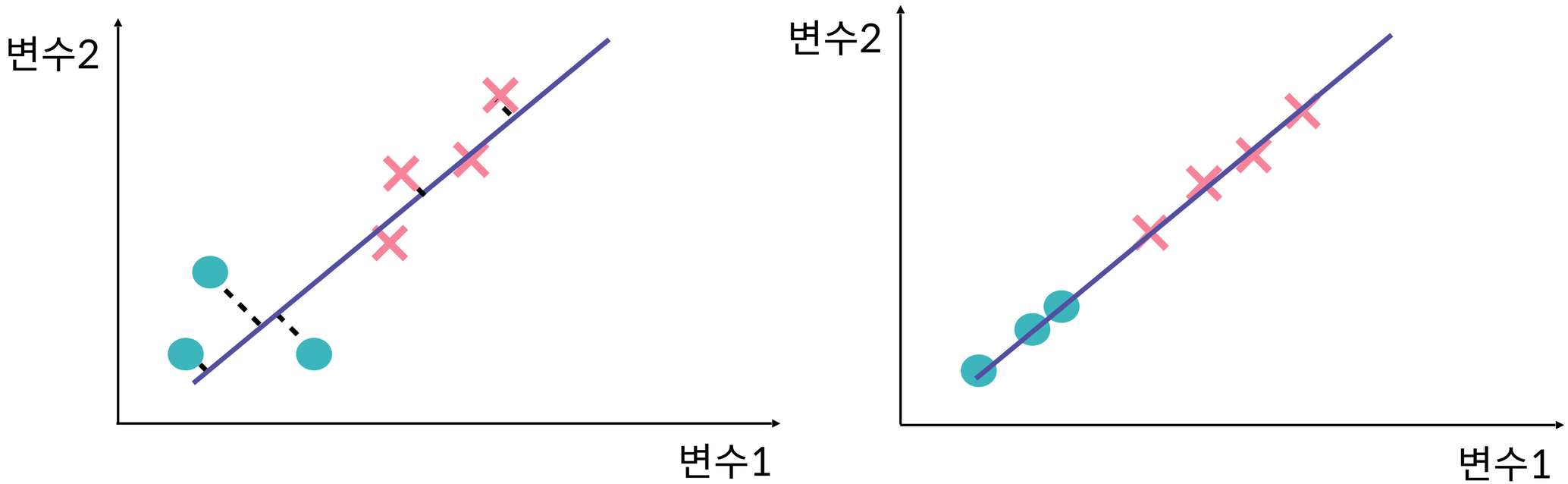
Case 1: **각 점들과 축의 오차가 큰 축**





주성분 분석 원리

Case 2: **각 점들과 축의 오차가 작은 축**



06 주성분 분석(PCA)

주성분 분석 특징 및 활용

- 고차원의 데이터를 **함축적으로 표현**하기 때문에 직관적 해석이 어려울 수 있음
- 대용량 고차원 데이터 압축 시 유용하게 활용됨

07

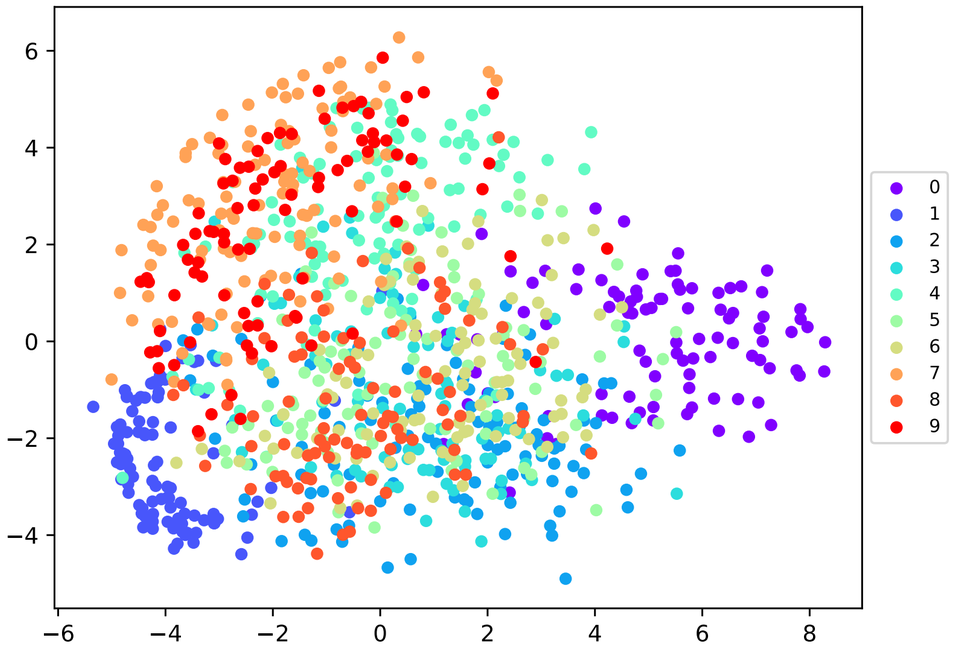
t-SNE

문제 정의와 해결 방안

- 문제 정의

고차원 공간에서 데이터들의 군집을
구별하여 시각화하고 싶다면?

PCA 사용 시 함축적 표현 –
시각화 진행 시 구별하기 어려움



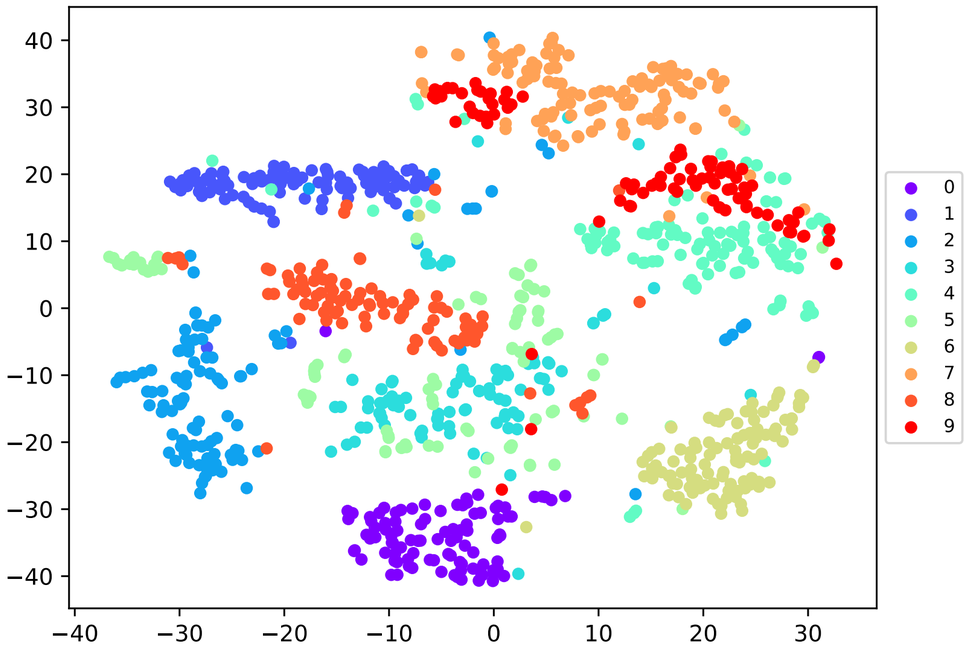
- 해결 방안

t-SNE(t-Stochastic Neighborhood Embedding) 알고리즘 활용

t-SNE(t-Stochastic Neighborhood Embedding)

고차원의 공간에 존재하는 **데이터 간의 거리를 최대한 유지**하며 차원을 축소하는 방법

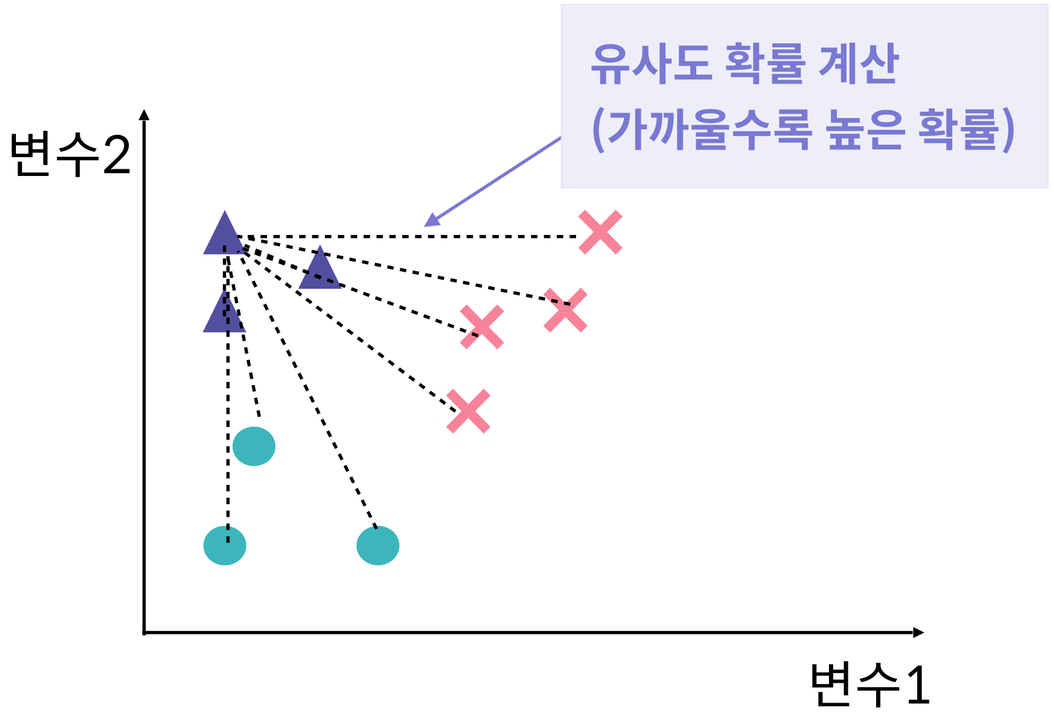
기존 데이터 공간에서 서로 가까이 있는 데이터는 차원이 줄어든 공간에서도 가까이 있어야 함



t-SNE 원리

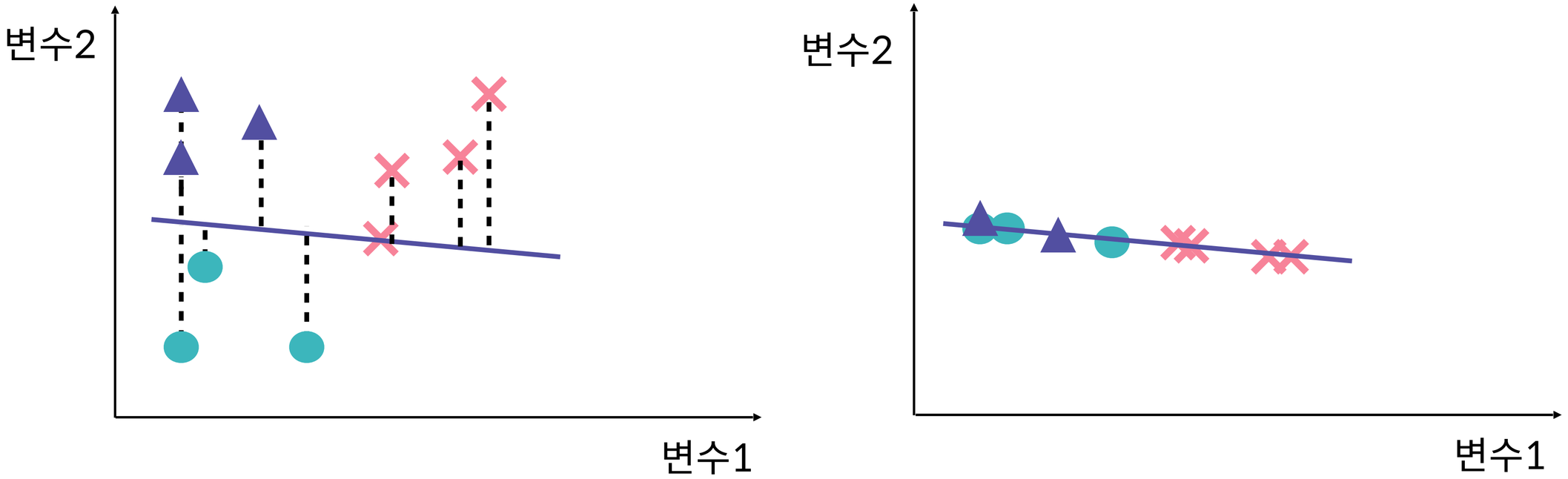
각 데이터마다 자기 이외의
데이터와의 **유사도 확률 계산**

- 데이터 산점도



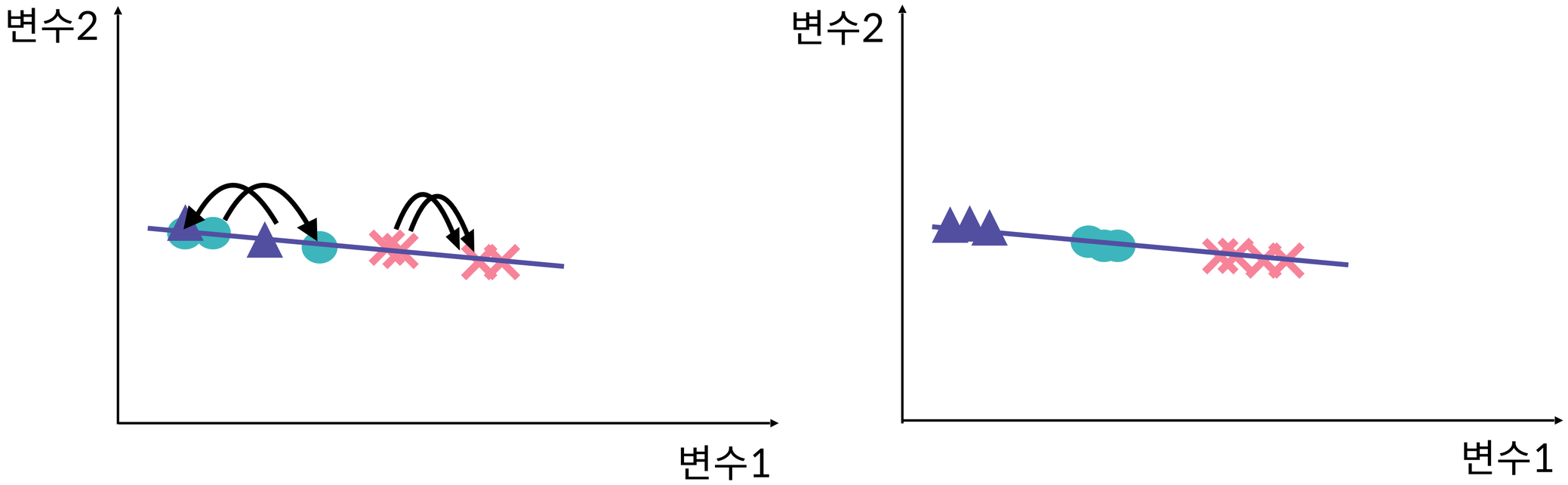
t-SNE 원리

저차원으로 **축소**



t-SNE 원리

초기에 계산했던 유사도 확률분포를 바탕으로 각 데이터를 **이동**



t-SNE 특징 및 활용

- 데이터 간 거리 유지를 통해 차원 축소 이후에도 객체 간 구별이 가능함
- 계산 시 마다 값이 지속적으로 변경되어 예측을 위한 학습 데이터로는 사용 불가
- 고차원 데이터의 시각화를 위해 활용됨

Contact

TEL

070-4633-2015

WEB

https://elice.io

E-MAIL

contact@elice.io