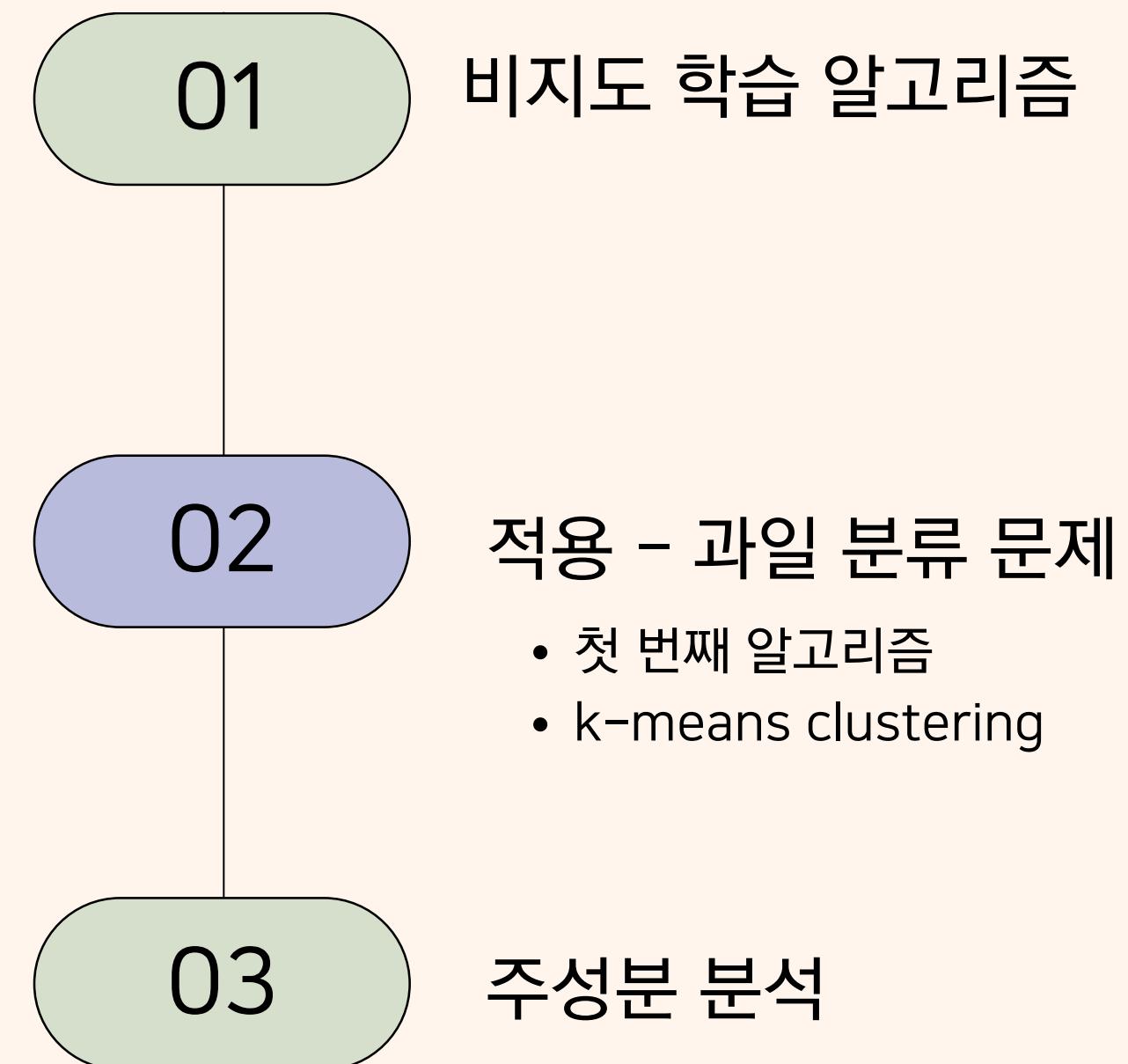


23 하계 학부연구생 프로그램

ch06. 비지도 학습

12223547 박혜민

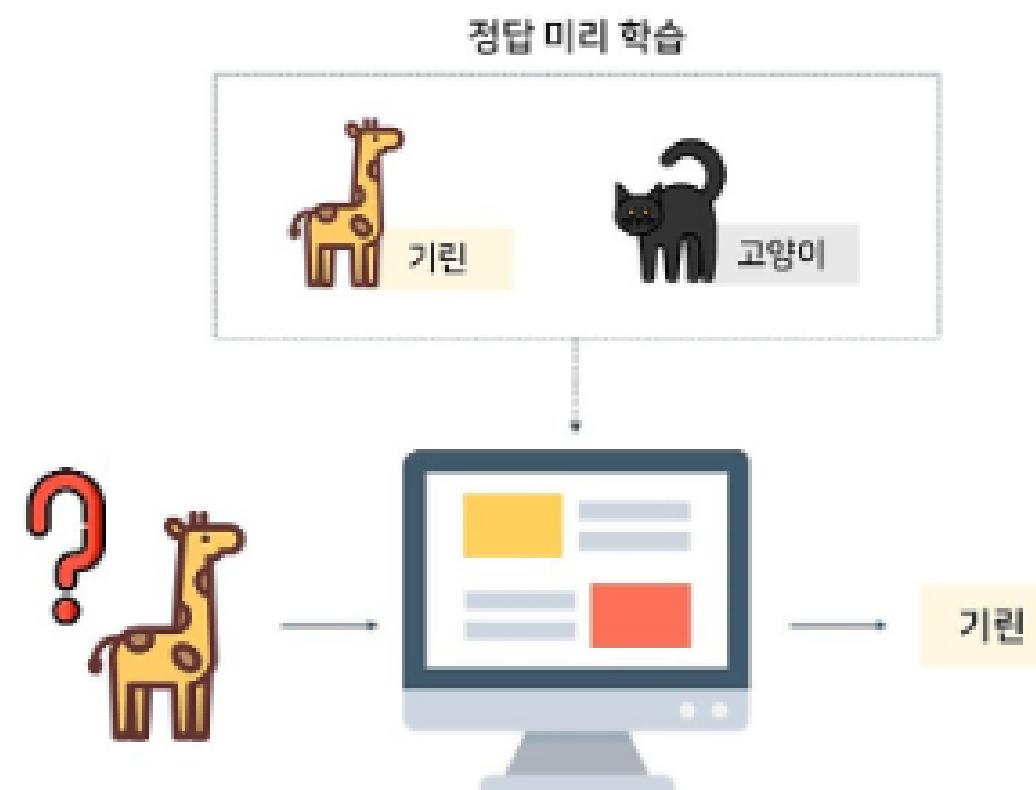
군집 알고리즘으로 과일 분류하기



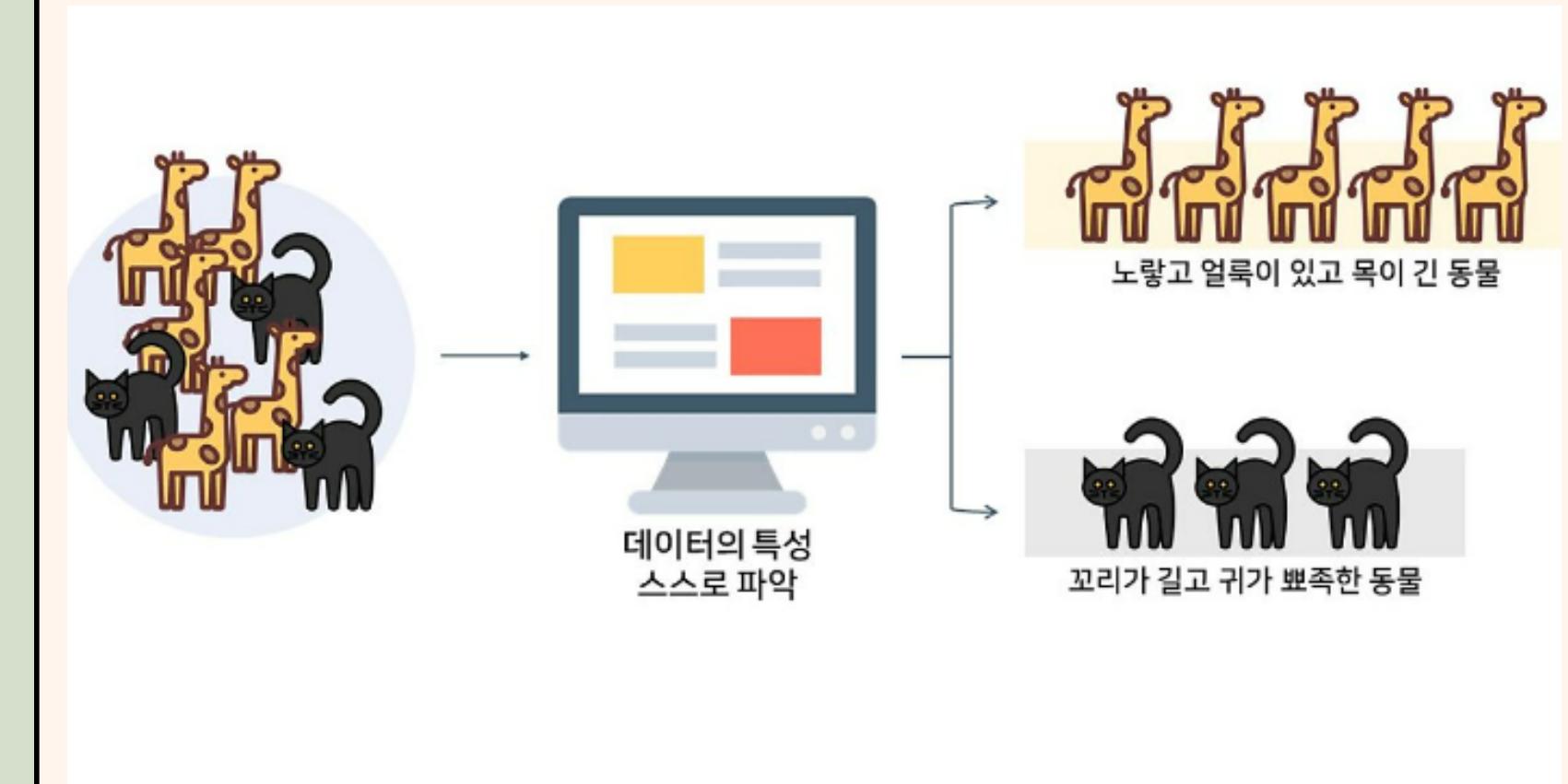
비지도 학습

지도학습 vs 비지도학습

지도학습(supervised learning)



비지도학습(unsupervised learning)



비지도 학습

데이터를 스스로 파악하여 일정한 규칙성을 찾는 방법

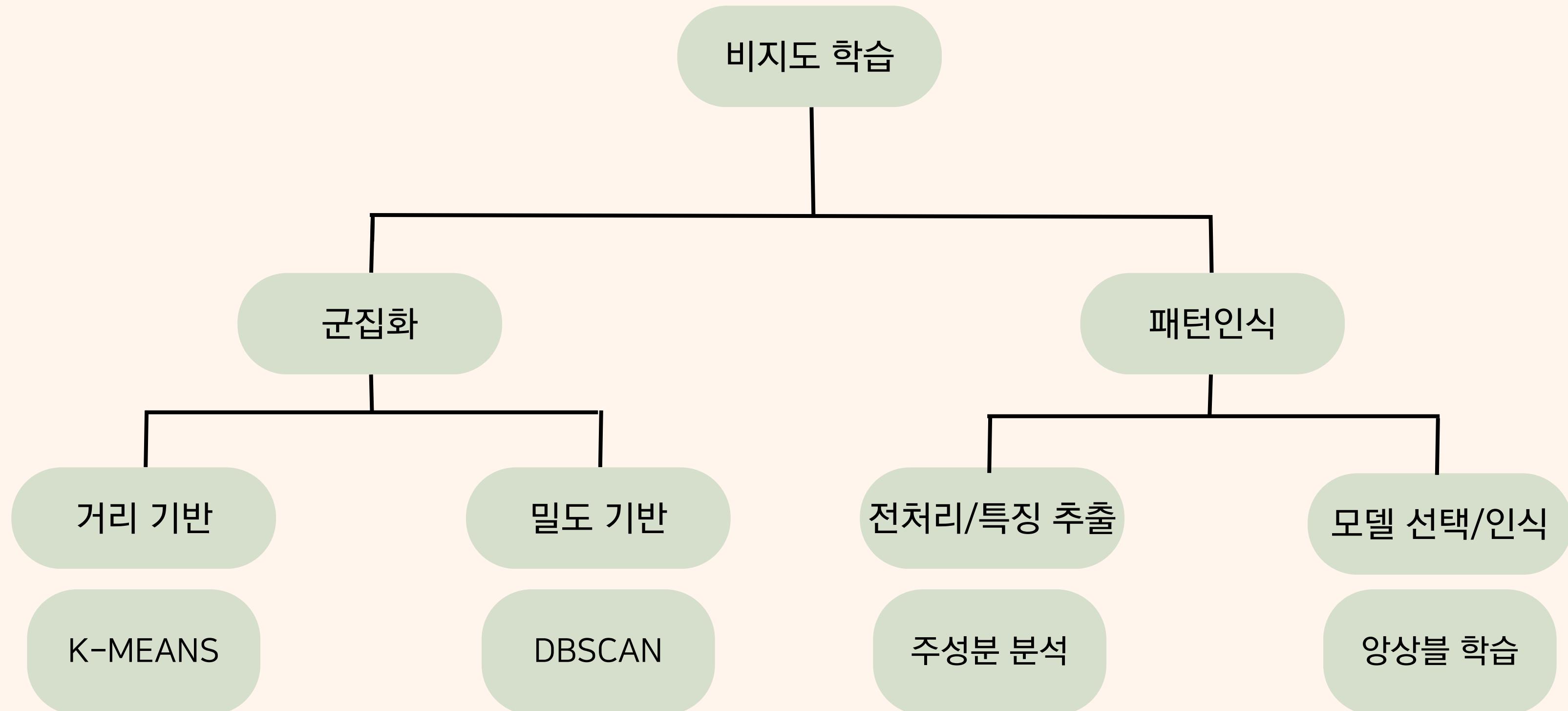
장점

- 사람도 인식하지 못한 본질적인 문제나 데이터에 숨겨진 특징이나 구조 파악 가능
- 목표값을 정해주지 않아도 되고 사전 학습이 필요없으므로 속도가 빠름

단점

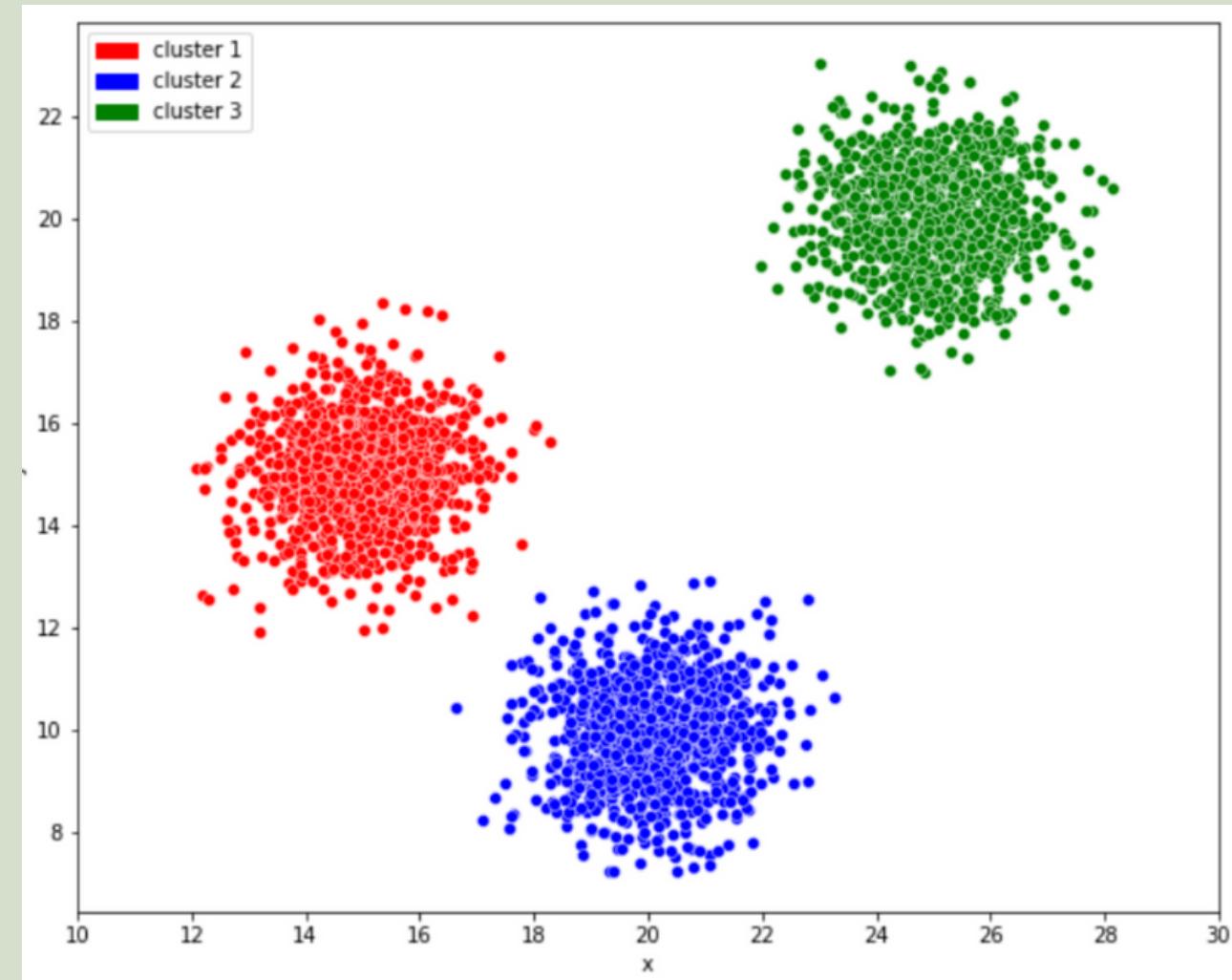
- 지도학습보다는 조금 더 난이도가 있다.
- 학습 결과만으로 분류 기준과 어떤 군집인지 예측 불가

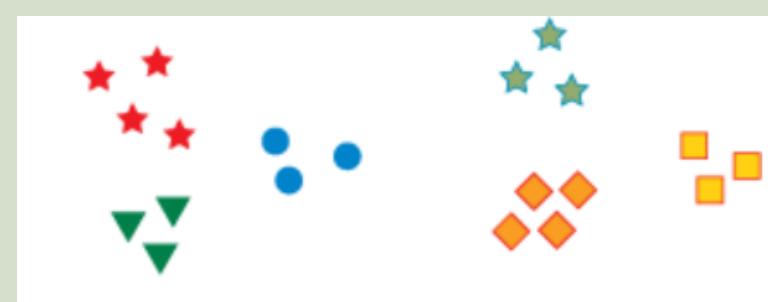
비지도 학습의 기법



군집화(clustering)

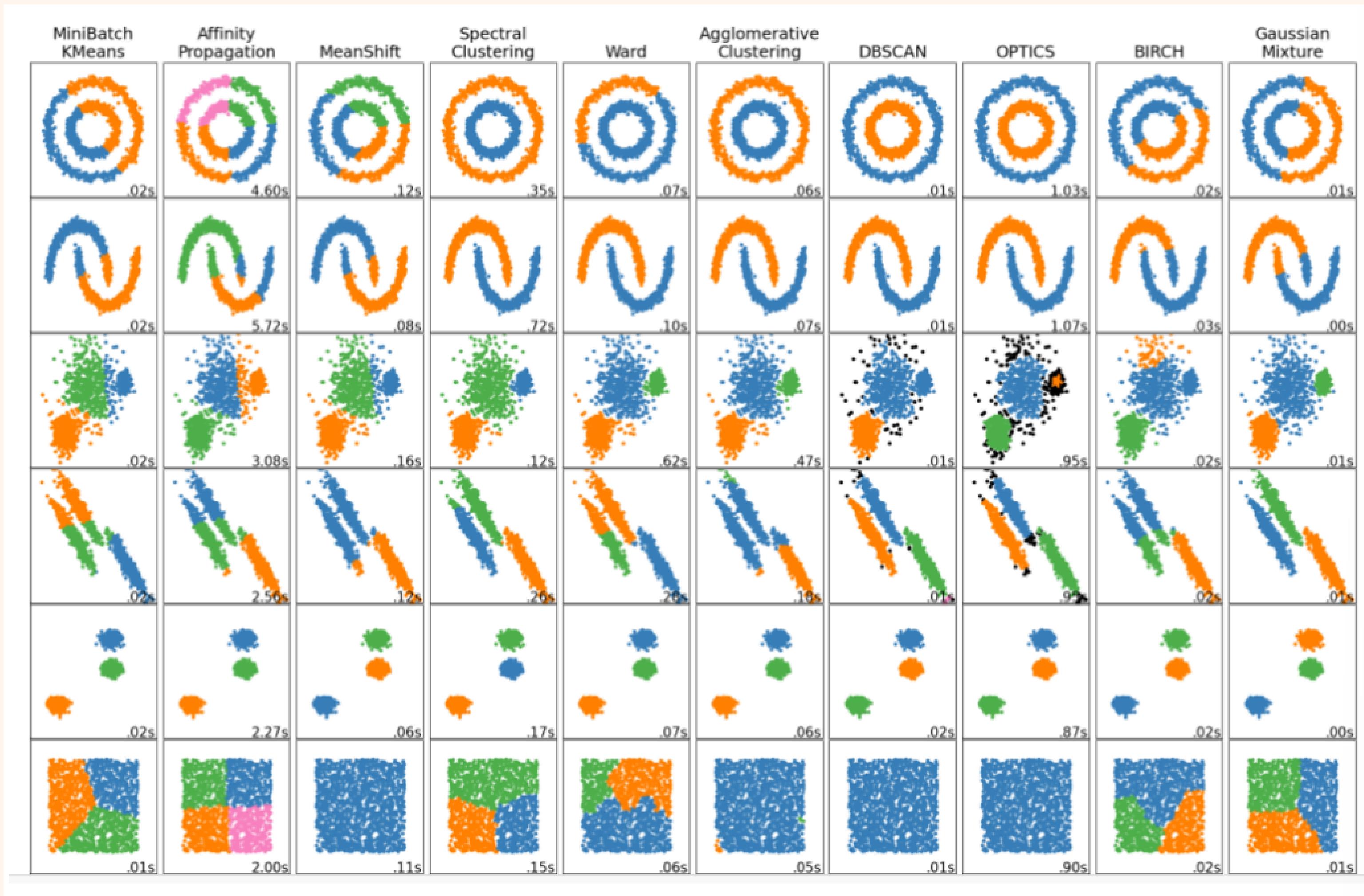
학습 데이터들의 특징(feature)을 분석하여 서로 동일하거나 유사한 특징을 가진 데이터끼리 그룹화하는 모델



- 군집을 나누는 절대적인 기준이 존재하지 않음
- 

6개의 군집
- 

4개의 군집
- 레이블이 없다는 점에서 분류(classification)과 구별



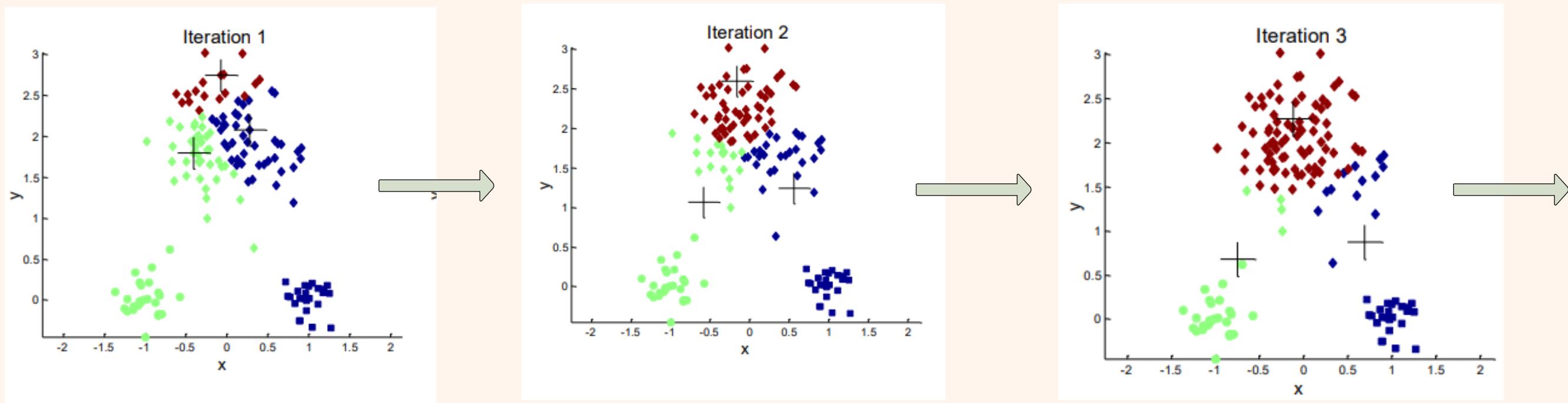
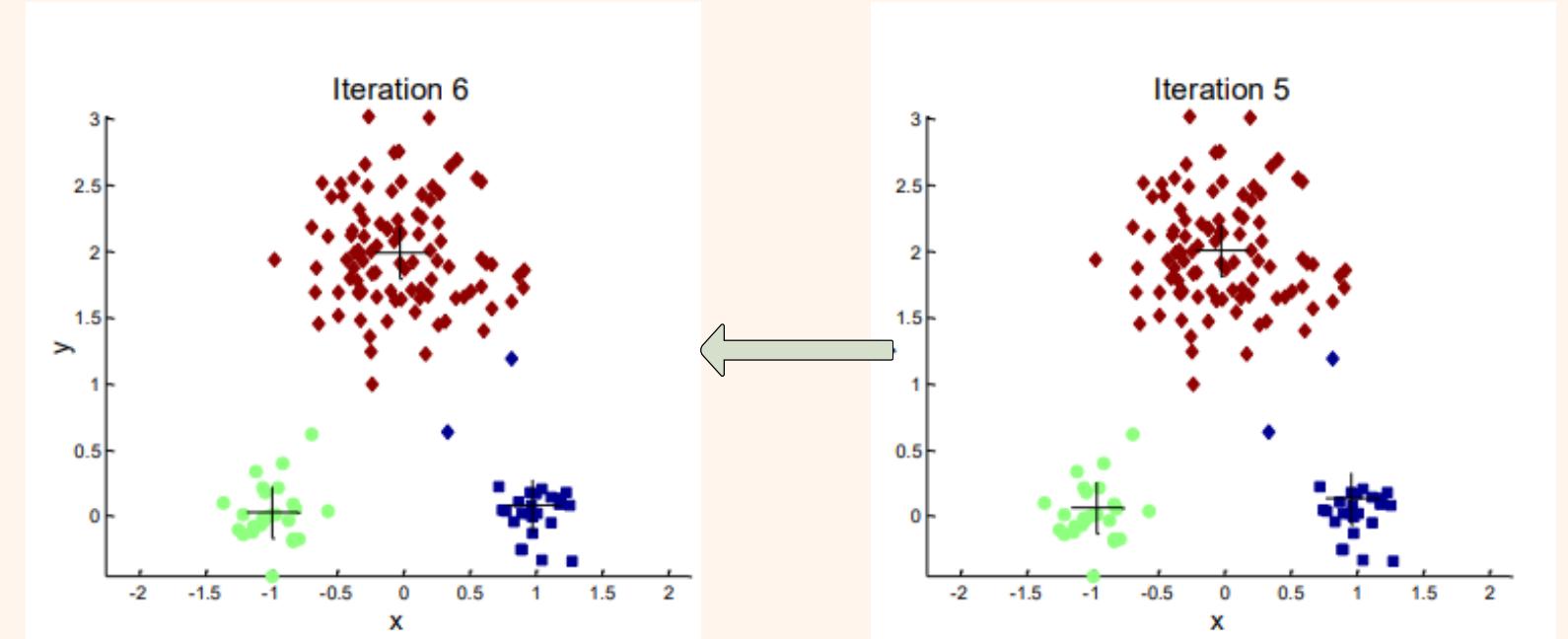
거리 기반 알고리즘

k-means clustering

특정한 임의의 지점을 선택해 해당 중심에 가장 가까운 포인트들을 선택하는 군집화 기법

<알고리즘>

-
- 1: Select K points as the initial centroids.
 - 2: **repeat**
 - 3: Form K clusters by assigning all points to the closest centroid.
 - 4: Recompute the centroid of each cluster.
 - 5: **until** The centroids don't change
-



k-means clustering

특징

- k 사전 지정
- 중심점과 각 데이터간 거리 계산 방법

1. 유클리드 거리

$$d(x_i, \mu_k) = \|x_i - \mu_k\|^2$$

2. 코사인 유사성

$$\text{similarity}=\cos(\theta)=\frac{A \cdot B}{\|A\|\|B\|}$$

3. 상관관계

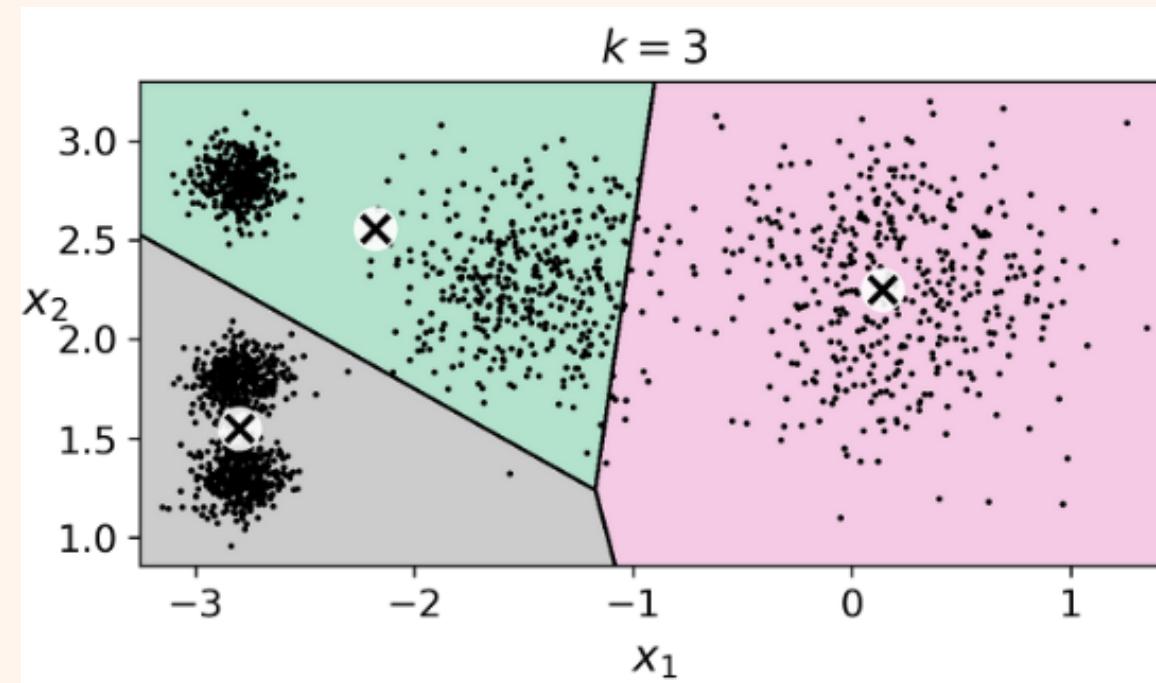
$$d_{corr}(X, Y) = 1 - r, \text{ where } r = \sigma_{XY}$$

- 평균으로 각 군집의 중심위치 계산

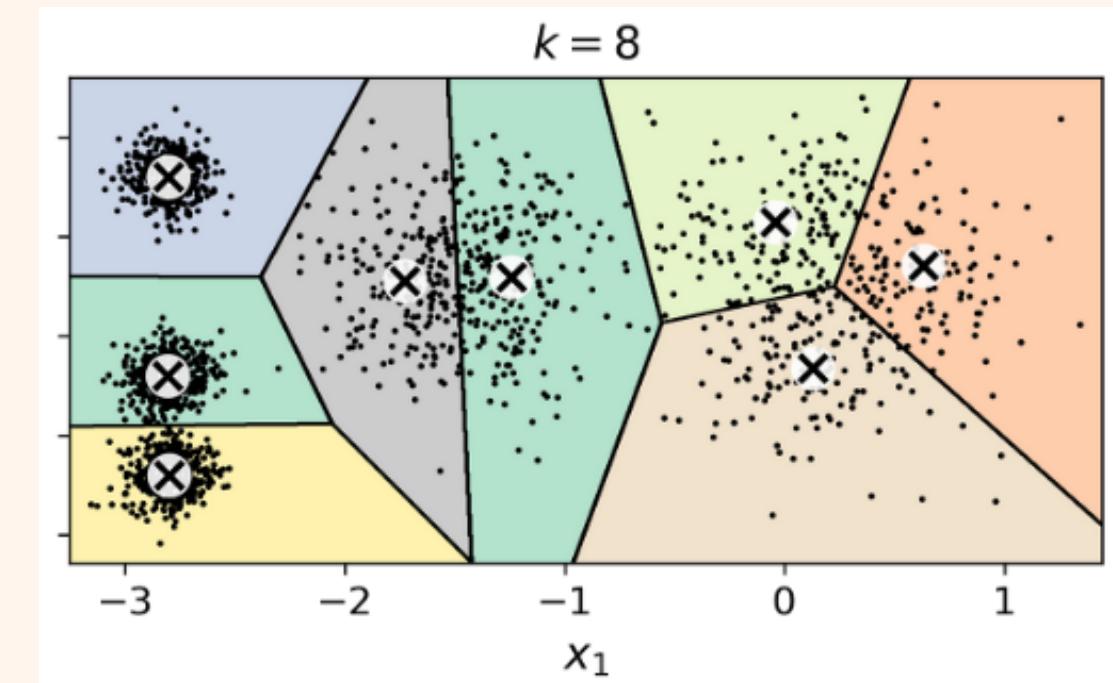
k-means clustering

엘보우 방법 – 사전 지식이 없는 경우 적절한 k 값 찾기

이너셔(inertia) : 클러스터 중심과 클러스터에 속한 샘플 사이의 거리 제곱합 (=응집도)



k
<
이너셔
>

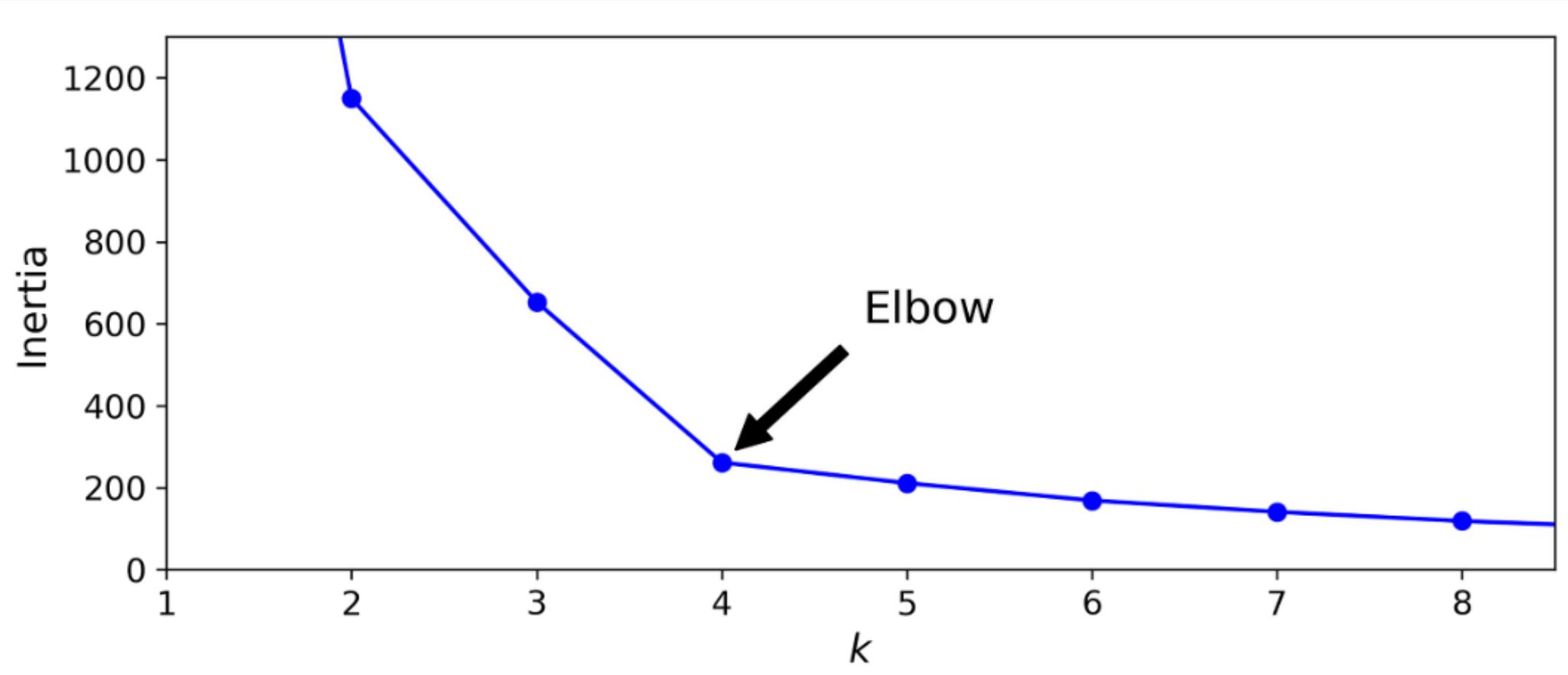


Q. 이너셔 값이 작을수록 좋은 모델인가? X

k-means clustering

엘보우 방법 - 사전 지식이 없는 경우 적절한 k 값 찾기

밀집도와 과대적합 사이의 균형점을 잘 찾는 것이 중요 -> 엘보우 지점



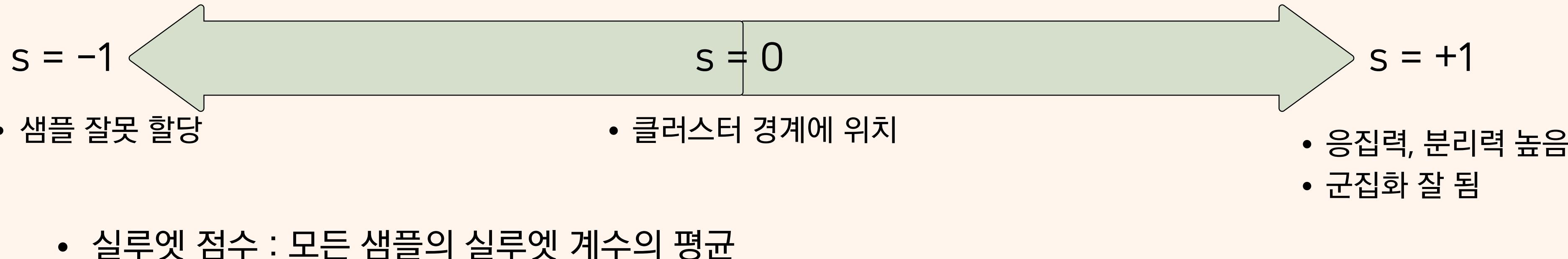
"그래프가 꺾이는 지점 관찰"

단점 1. 엘보우 지점이 명확하지 않을 수 있음 2. 계산비용이 많이 듬

k-means clustering

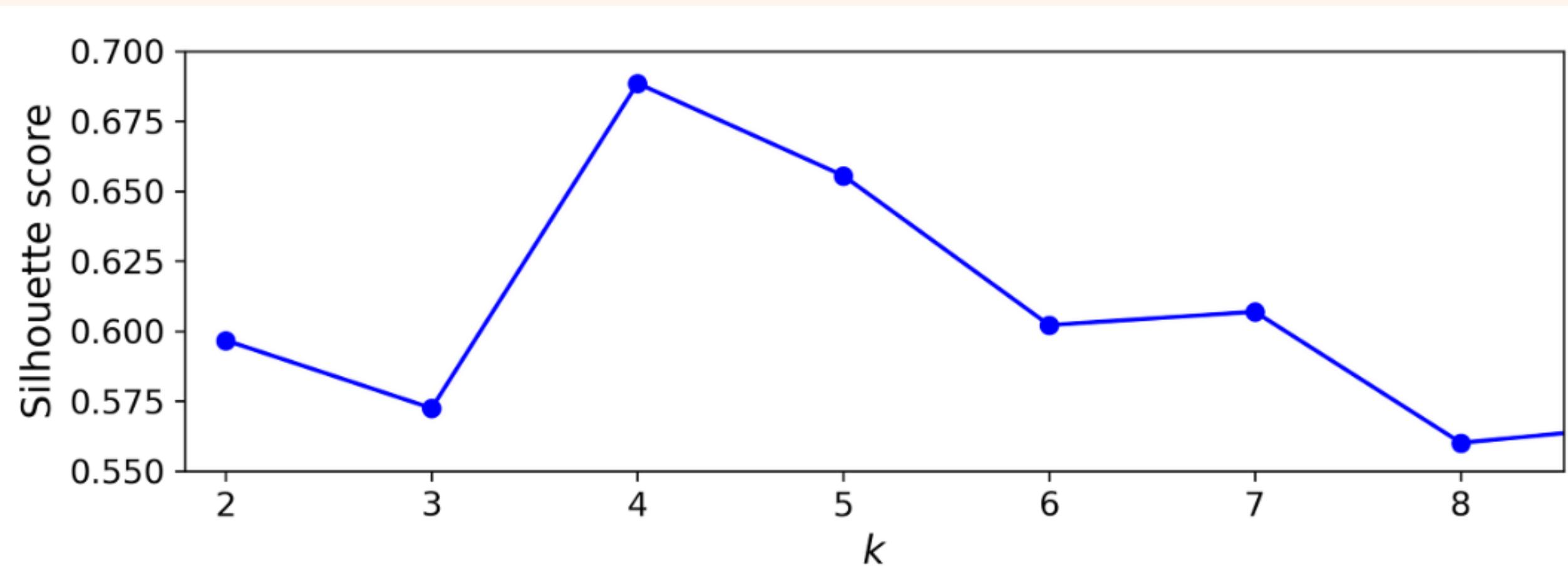
실루엣 점수 - 사전 지식이 없는 경우 적절한 k 값 찾기

- a : 동일한 클러스터에 있는 다른 샘플까지 평균 거리
- b : 현재 속한 클러스터를 제외한 가장 가까운 클러스터까지 평균 거리
- 실루엣 계수 : $s = (b - a) / \max(a, b)$



k-means clustering

실루엣 점수 – 사전 지식이 없는 경우 적절한 k 값 찾기



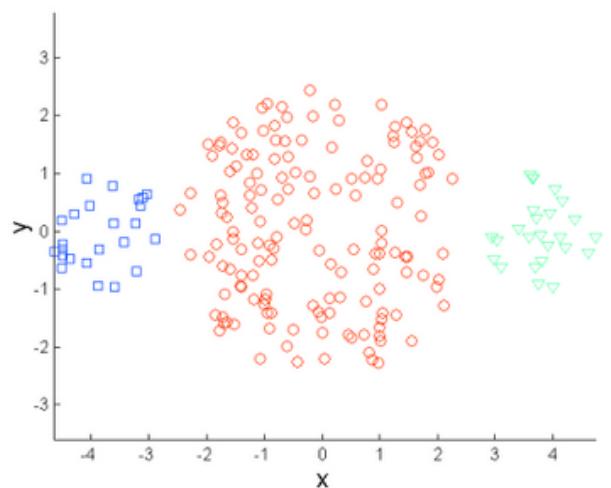
- $k = 4$ 일 때 실루엣 점수 최대, 그 이후 감소
- 이너셔보다 더 명확하게 파악 가능

k-means clustering

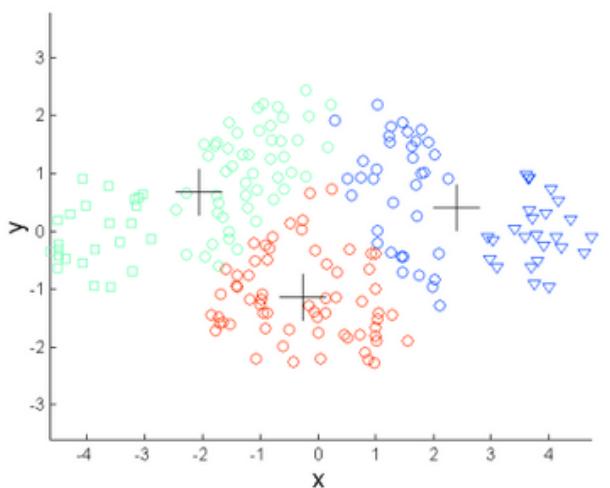
한계

1. 데이터의 형태적 한계

- 다른 크기

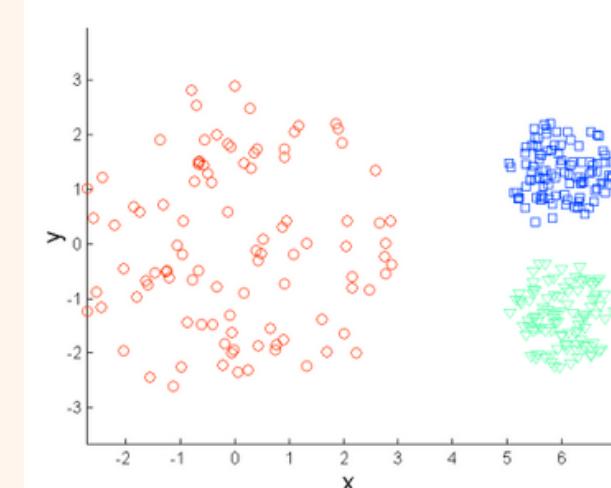


원본 포인트

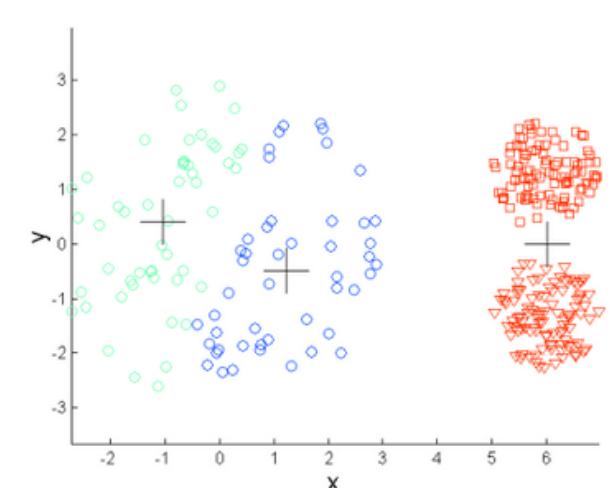


K-means (3개의 군집)

- 다른 밀도



원본 포인트



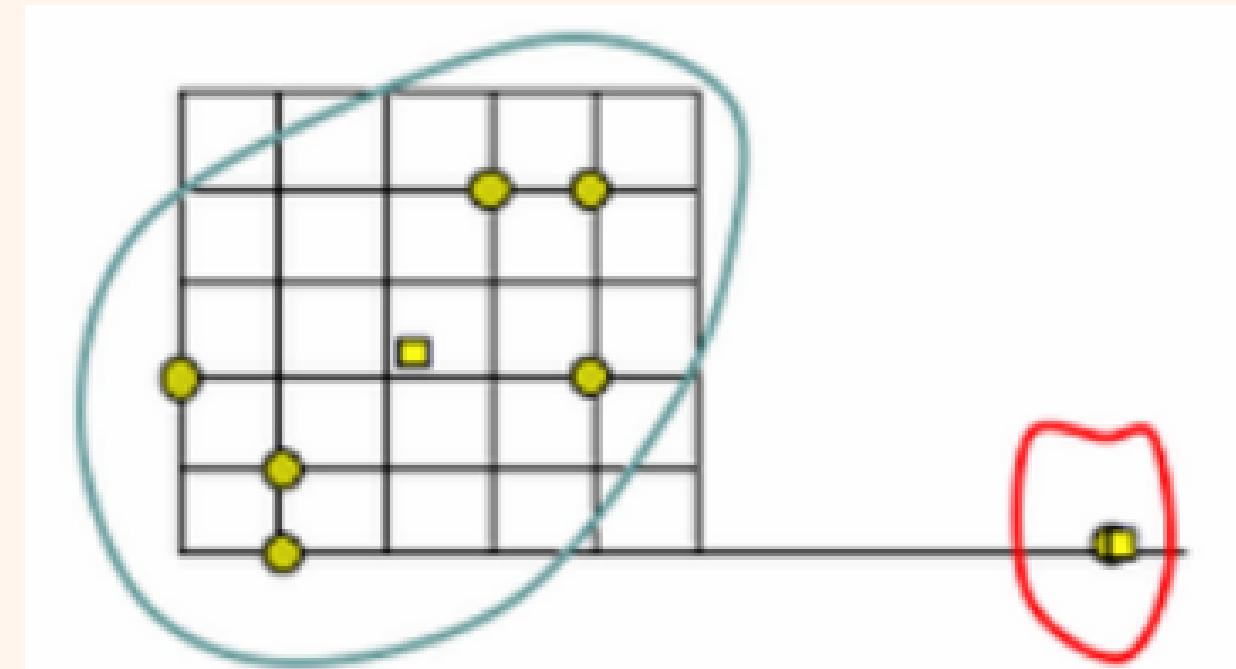
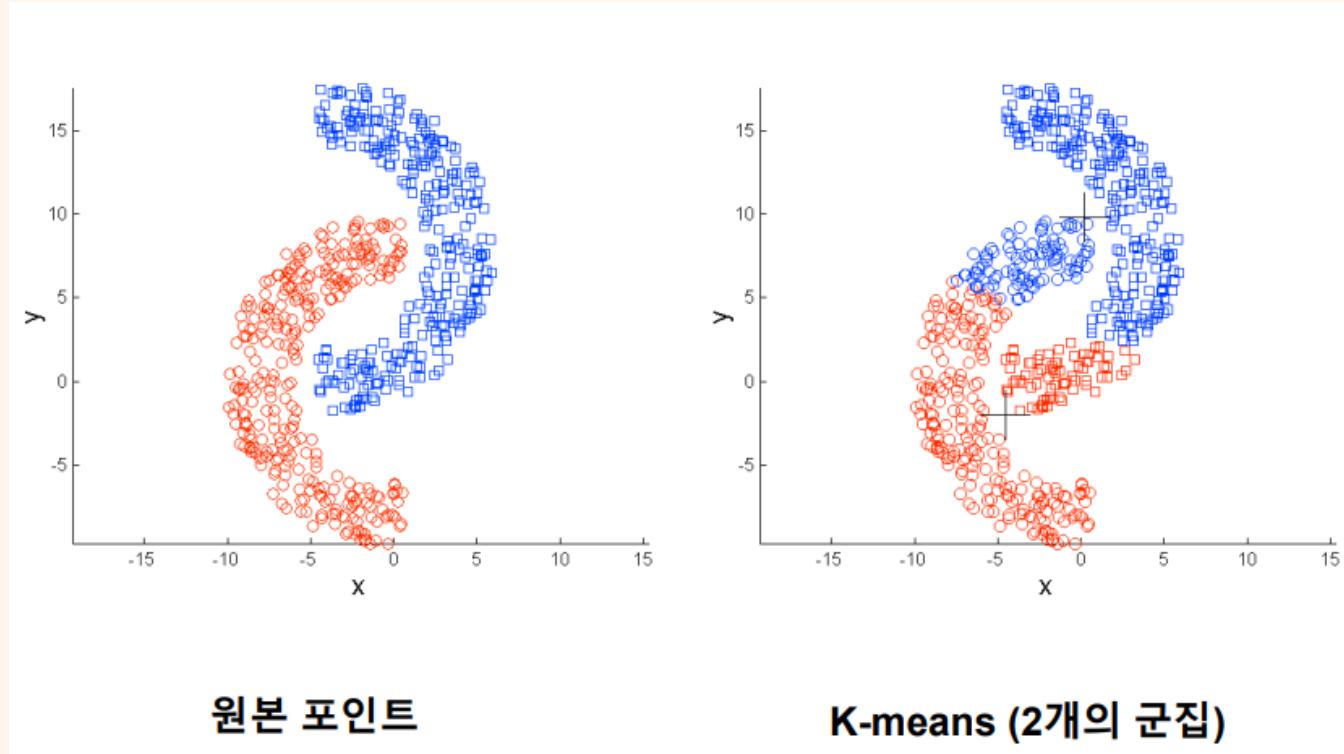
K-means (3개의 군집)

k-means clustering

한계

- 비구 모양

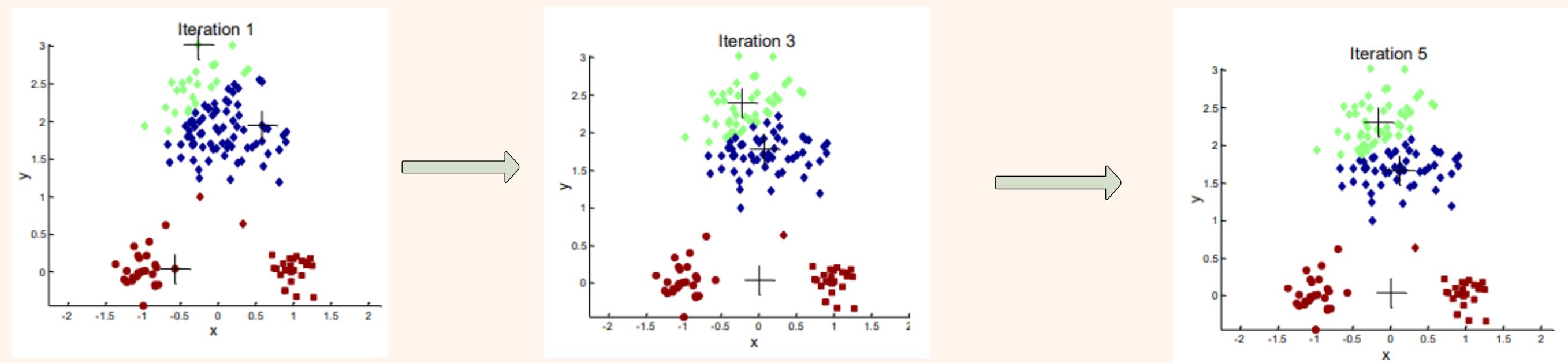
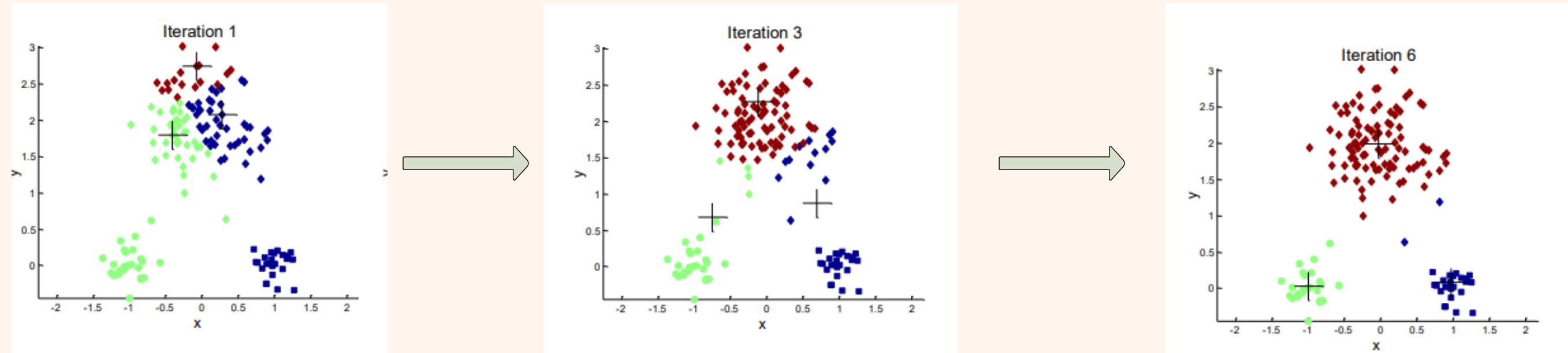
2. 데이터의 이상치를 잘 극복하지 못한다.



k-means clustering

한계

3. 초기 중심점에 따라 군집 결과가 달라진다.



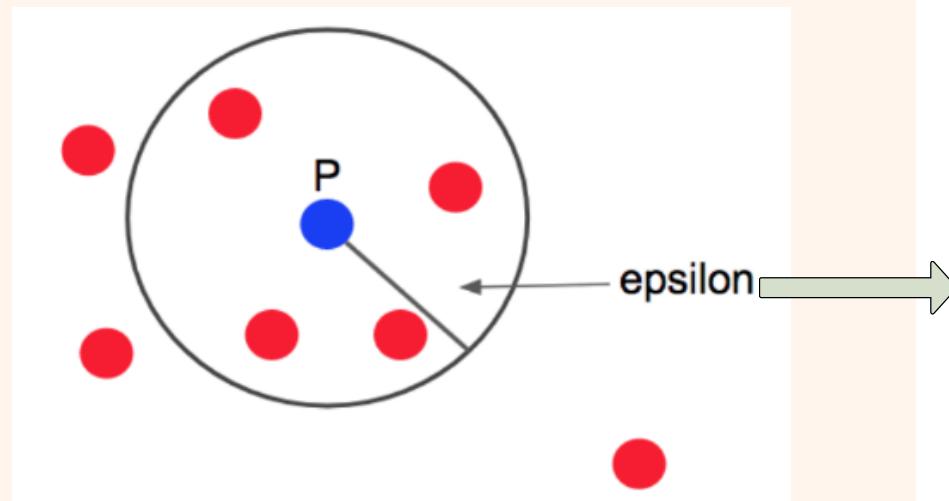
밀도 기반 알고리즘

DBSCAN

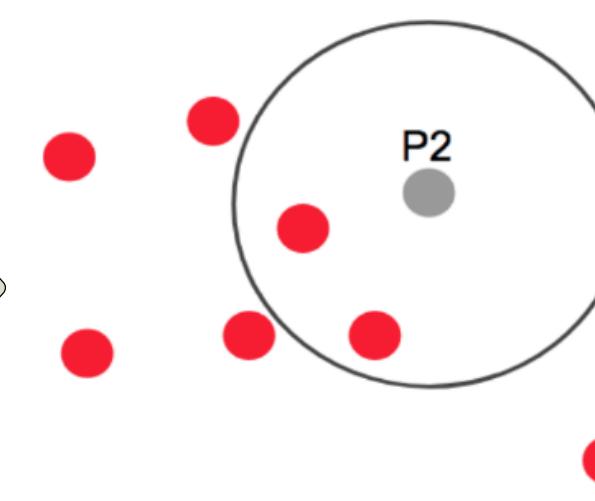
특정 공간 내에 데이터 밀도 차이에 기반한 알고리즘

<알고리즘>

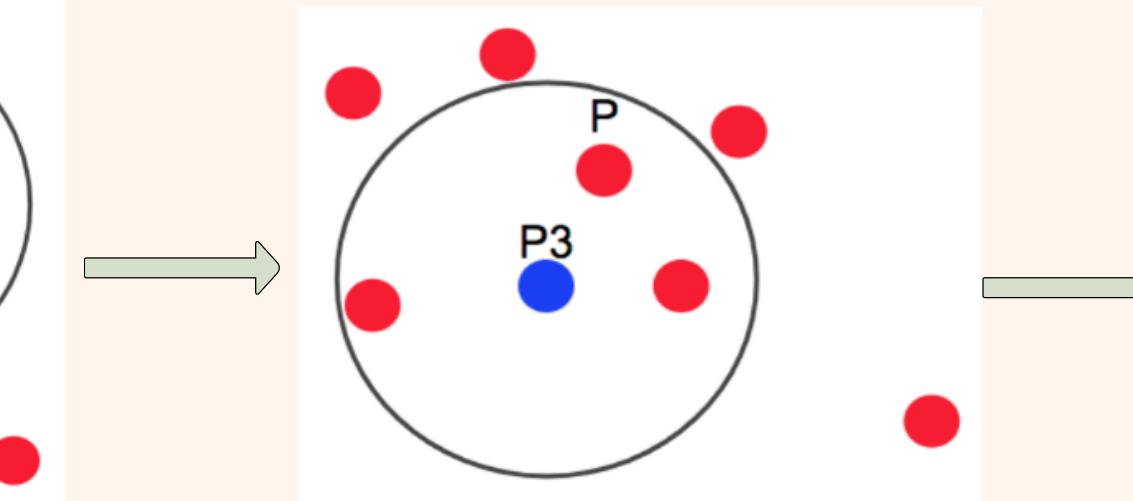
- 최소 데이터 개수(min points) = 4
- 영역 내 샘플 수 \geq min points \rightarrow 핵심 포인트(core point)
- 핵심포인트의 영역 내 점 and 영역 내 샘플 수 $<$ min points \rightarrow 경계점(border point)
- 핵심포인트의 영역 외 점 and 영역 내 샘플 수 $<$ min points \rightarrow 이상치(noise point)



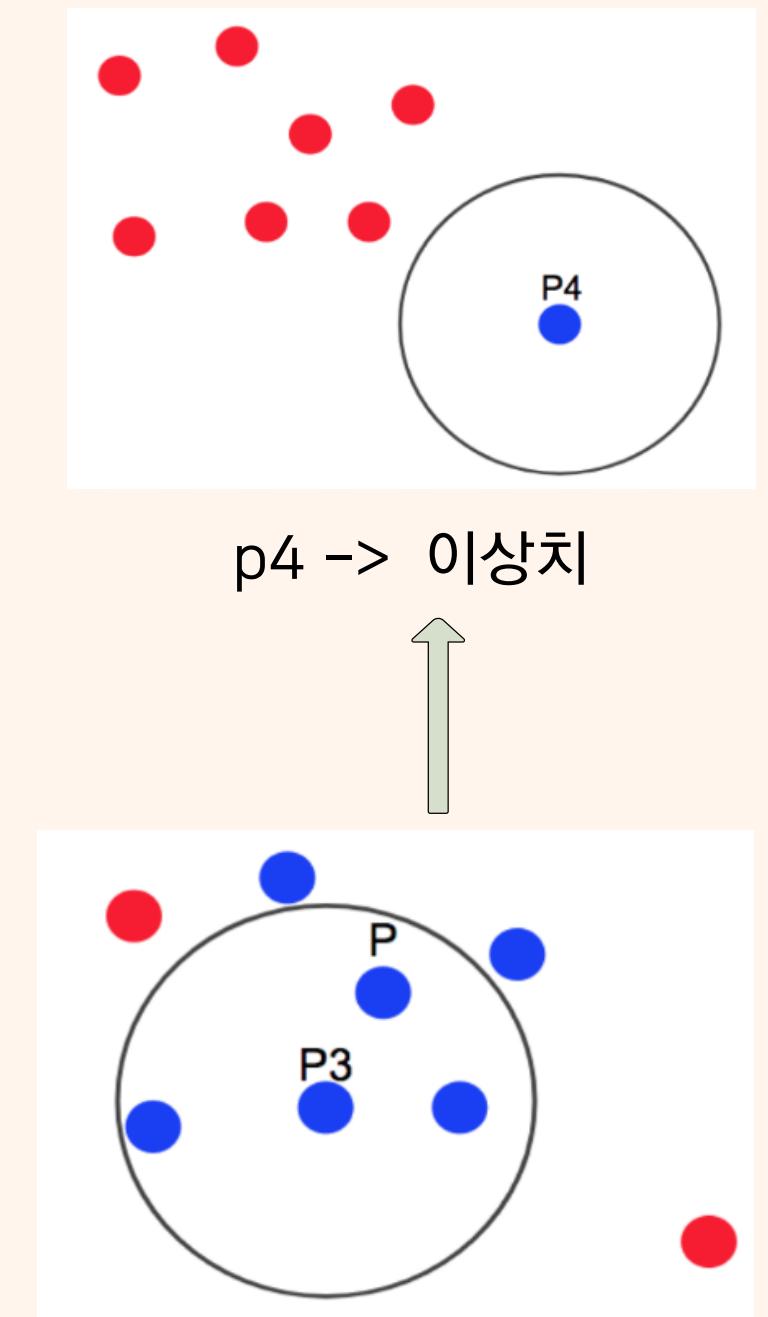
p \rightarrow 핵심 포인트



P2 \rightarrow 경계점



P3 \rightarrow 핵심 포인트



군집 연결

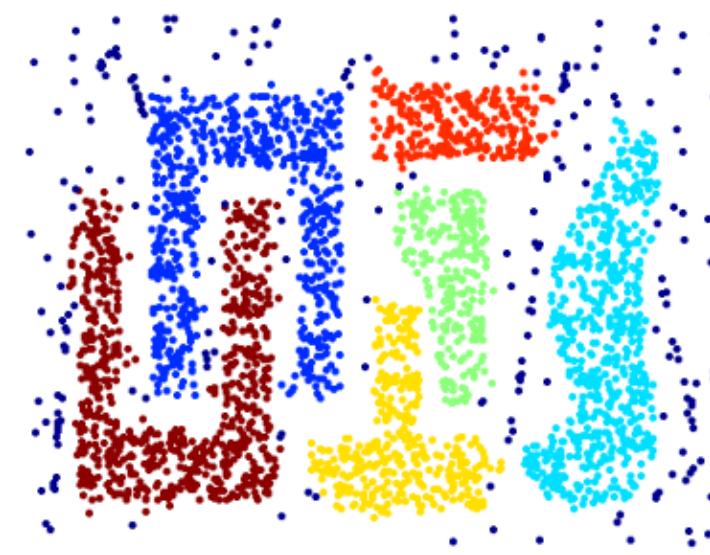
DBSCAN

장점

- 클러스터의 수를 정하지 않아도 됨
- 노이즈 검출/제거 가능
- 데이터의 분포가 기하학적으로 복잡한 데이터에 효과

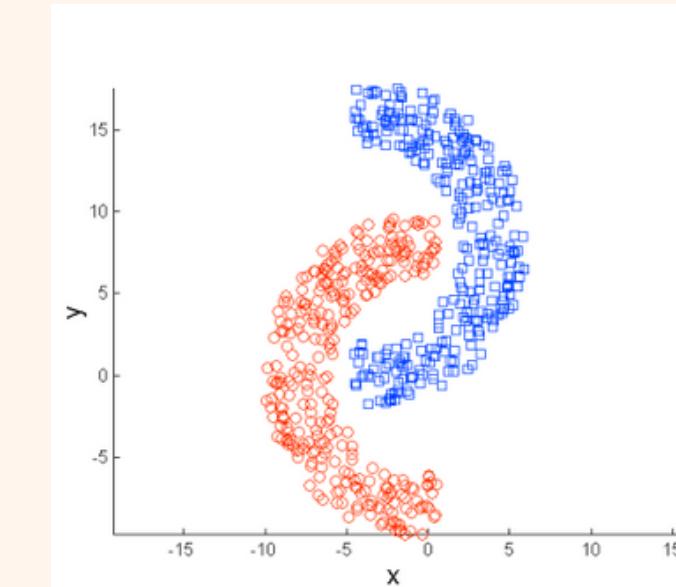


원본 포인트

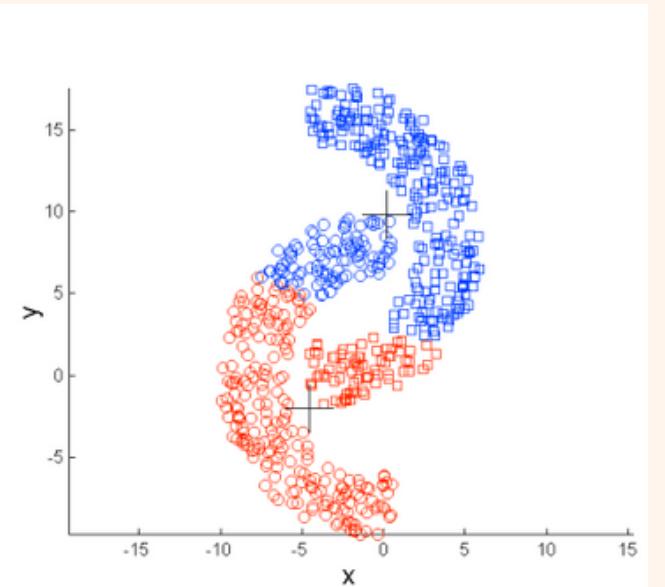


군집

VS



원본 포인트



K-means (2개의 군집)

<DBSCAN>

<비구 형태 데이터의 k-means>

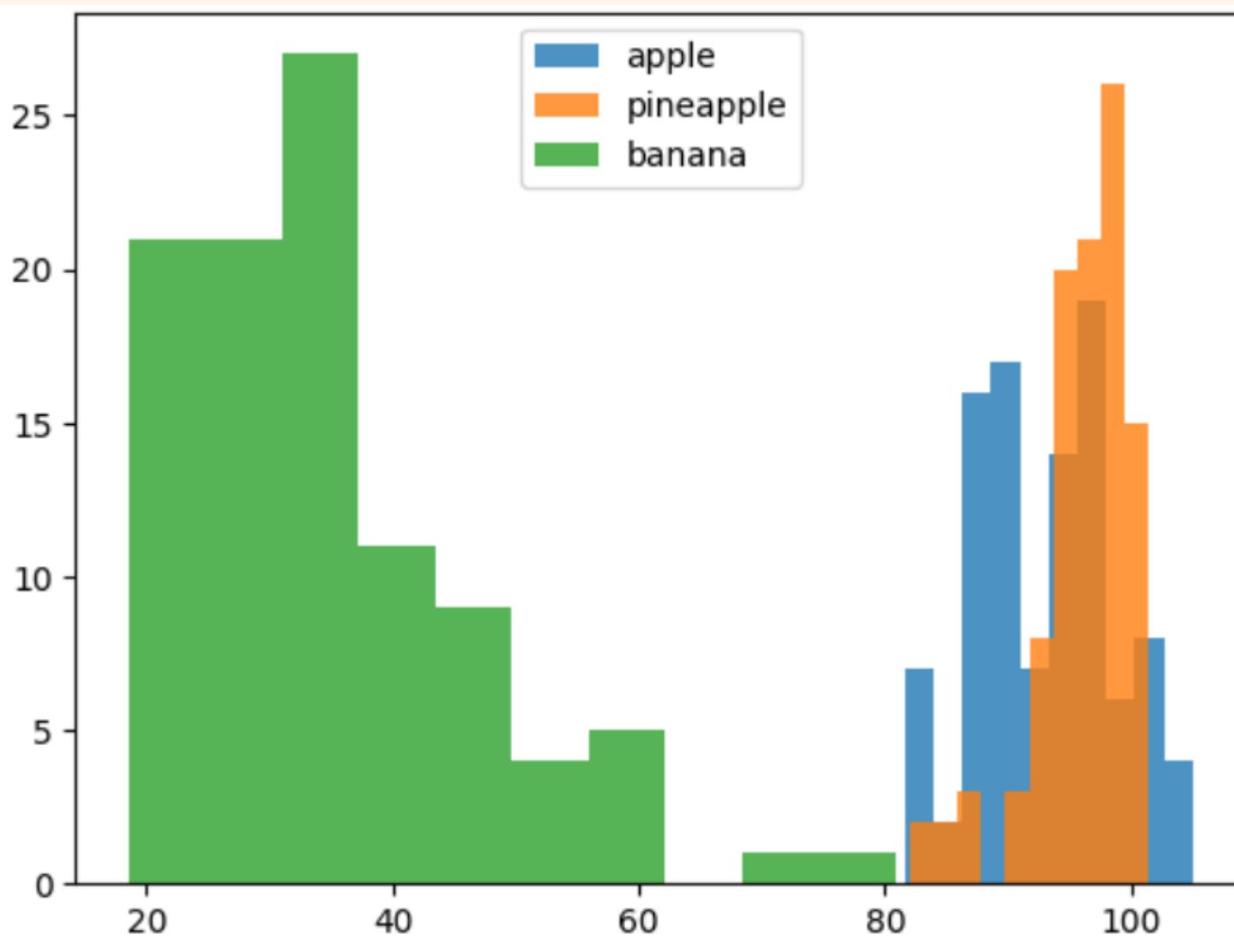
적용 - 과일 분류 문제

'이미지의 픽셀값을 모두 평균 내면 비슷한 과일끼리 모일 것이다.'

첫 번째 알고리즘

타깃값을 아는 경우

방법1. 샘플별 평균 픽셀값 이용



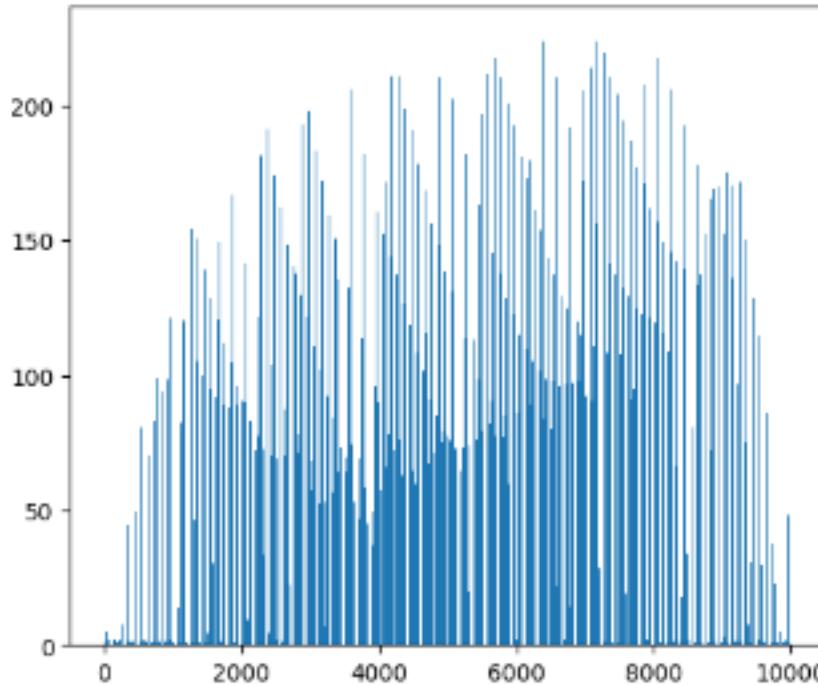
- 바나나 : 평균값이 40아래에 집중되어 있음
- 사과, 파인애플 : 90~100 사이에 모여 있음
- 문제점 : 파인애플과 사과의 구별이 어려움

첫 번째 알고리즘

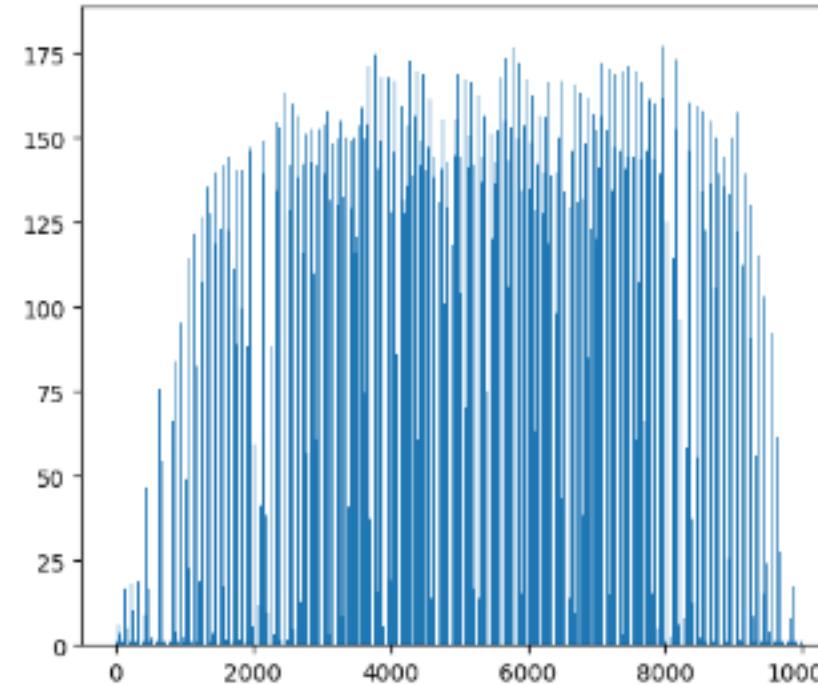
타깃값을 아는 경우

방법2. 픽셀별 평균값 이용

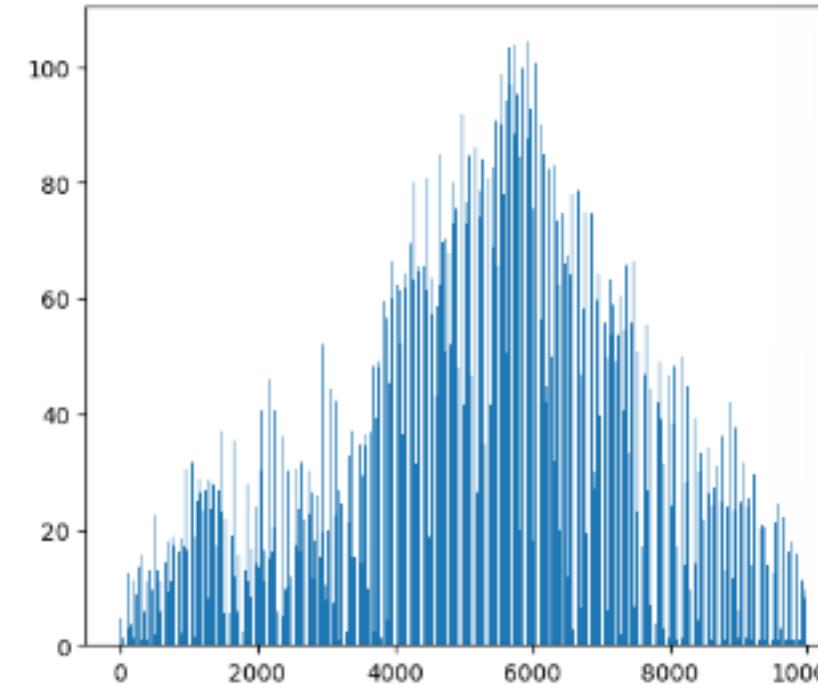
사과



파인애플



바나나



- 사과 : 사진 아래쪽으로 갈수록 값이 높아짐
- 파인애플 : 비교적 고르면서 높음
- 바나나 : 중앙 픽셀값이 높음

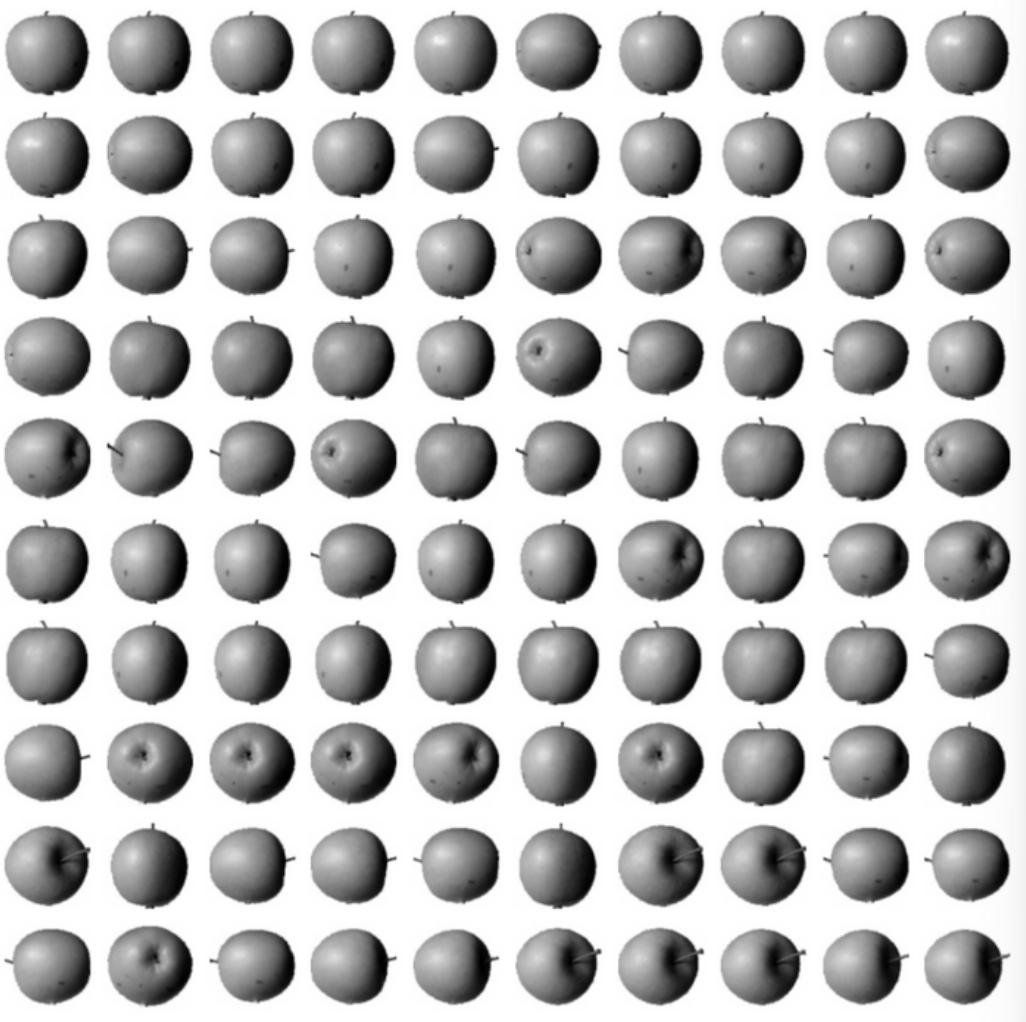
첫 번째 알고리즘

타깃값을 아는 경우

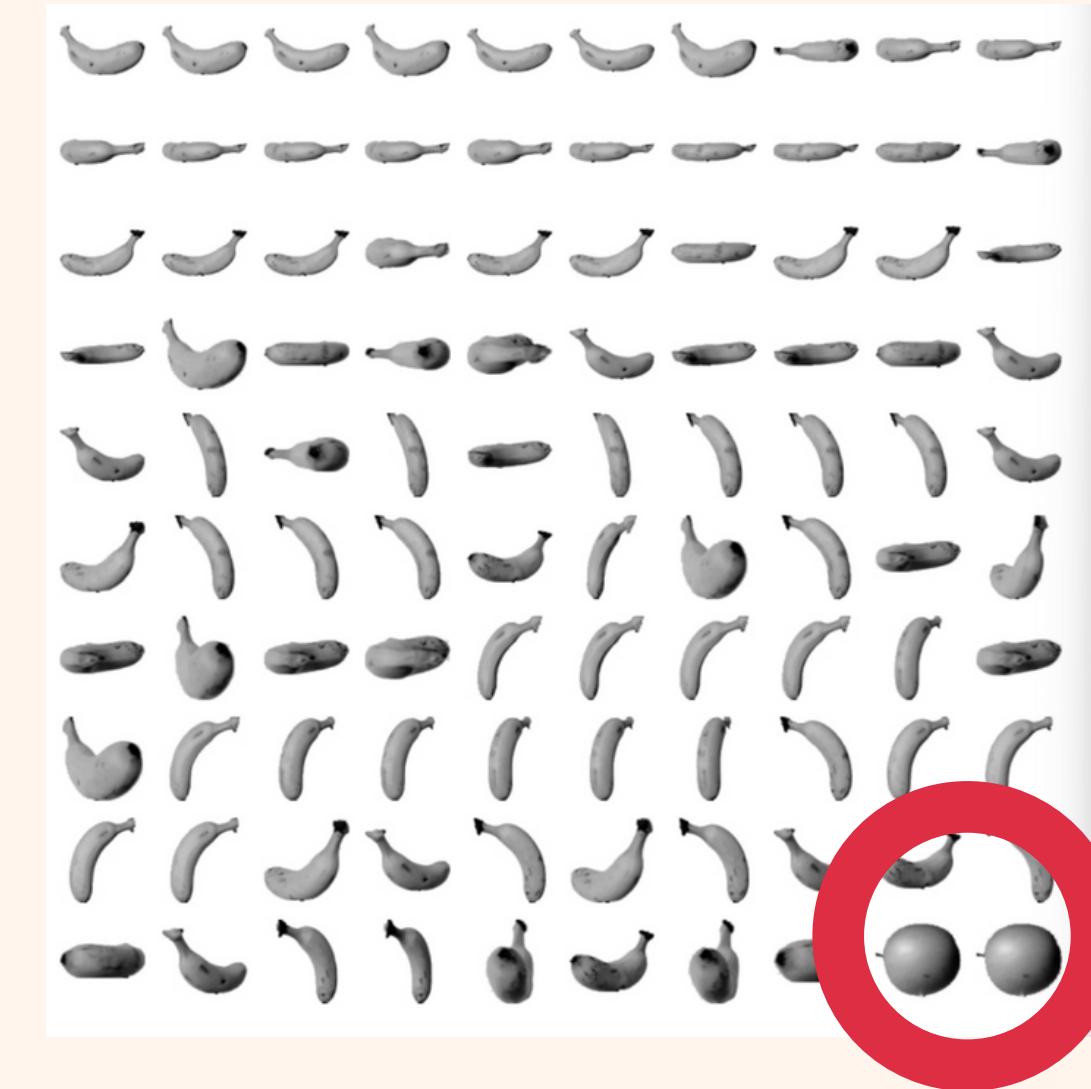
방법2. 픽셀별 평균값 이용

- 오차 절댓값 평균을 사용하여 군집 구성

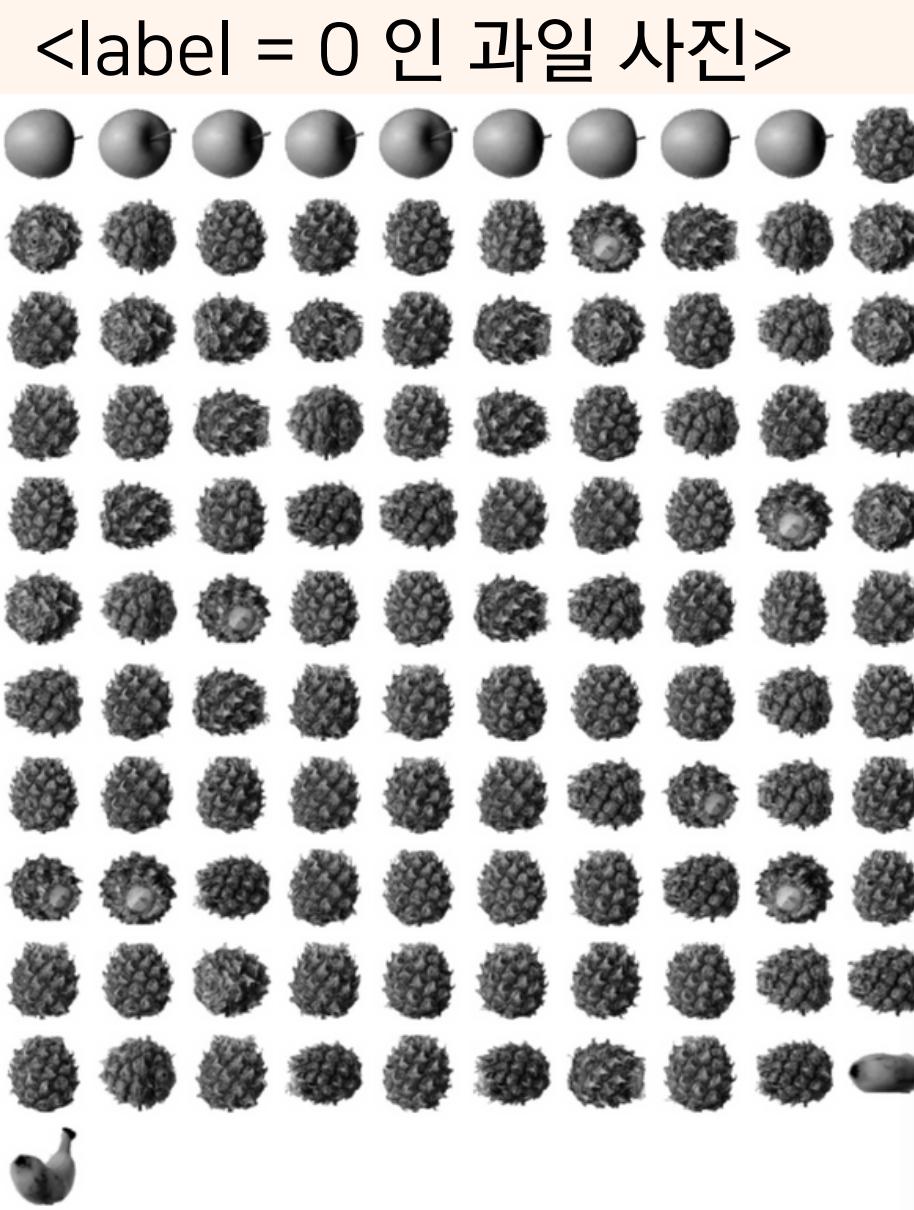
<사과와 가까운 사진>



<바나나와 가까운 사진>



두 번째 알고리즘 - k-means

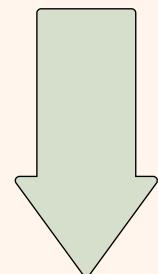


두 번째 알고리즘 - k-means

- 각 클러스터 중심까지 거리를 특성으로 사용하여 데이터셋을 저차원으로 줄일 수 있음

```
[55] np.shape(fruits_2d[100:101])
```

```
(1, 10000)
```



#훈련 데이터 샘플에서 각 클러스터 중심까지 거리로 변환

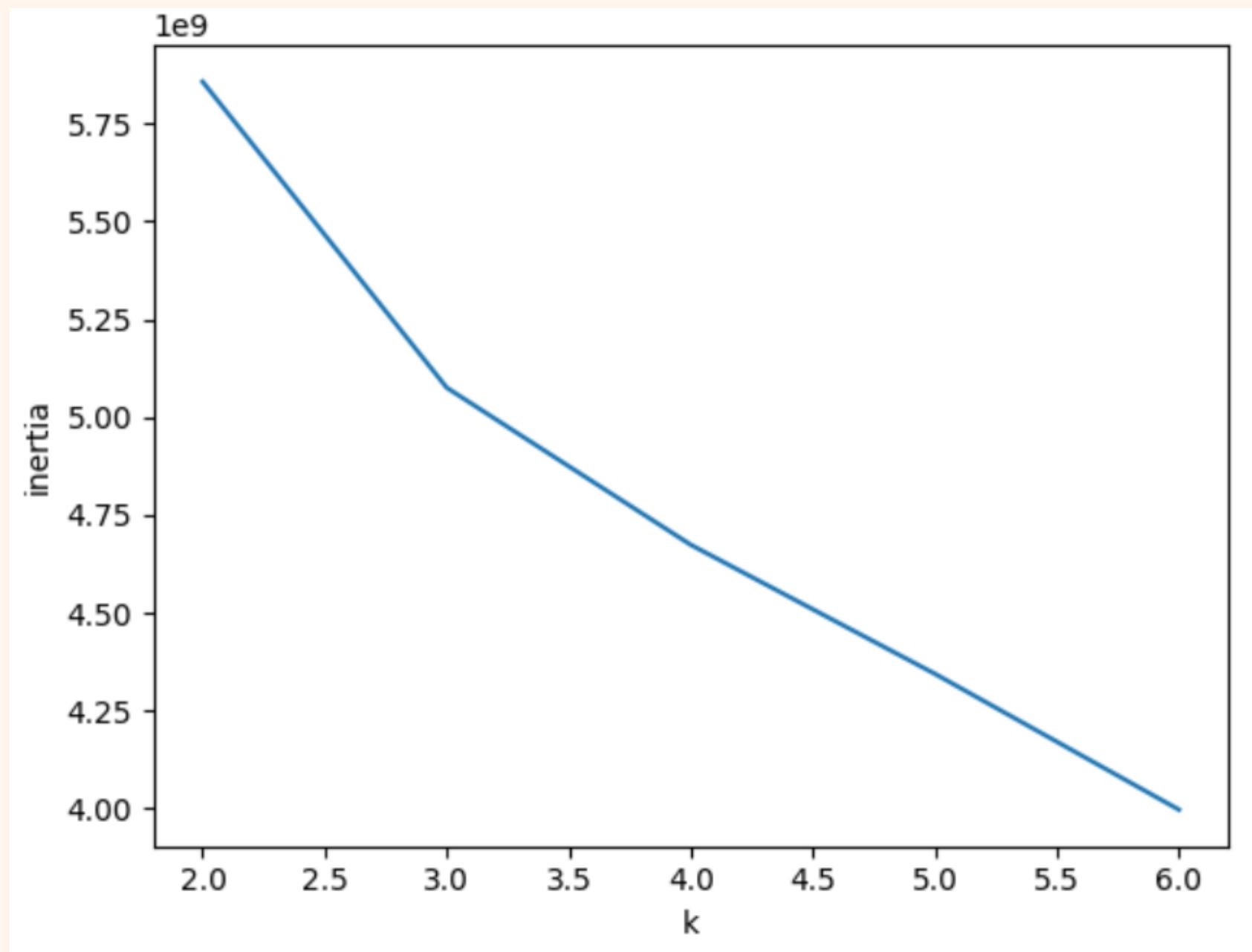
```
print(km.transform(fruits_2d[100:101])) #레이어 2에 속하는 샘플
```



```
[[3393.8136117 8837.37750892 5267.70439881]]
```

두 번째 알고리즘 - k-means

- $k = 30$ | 엘보우 지점

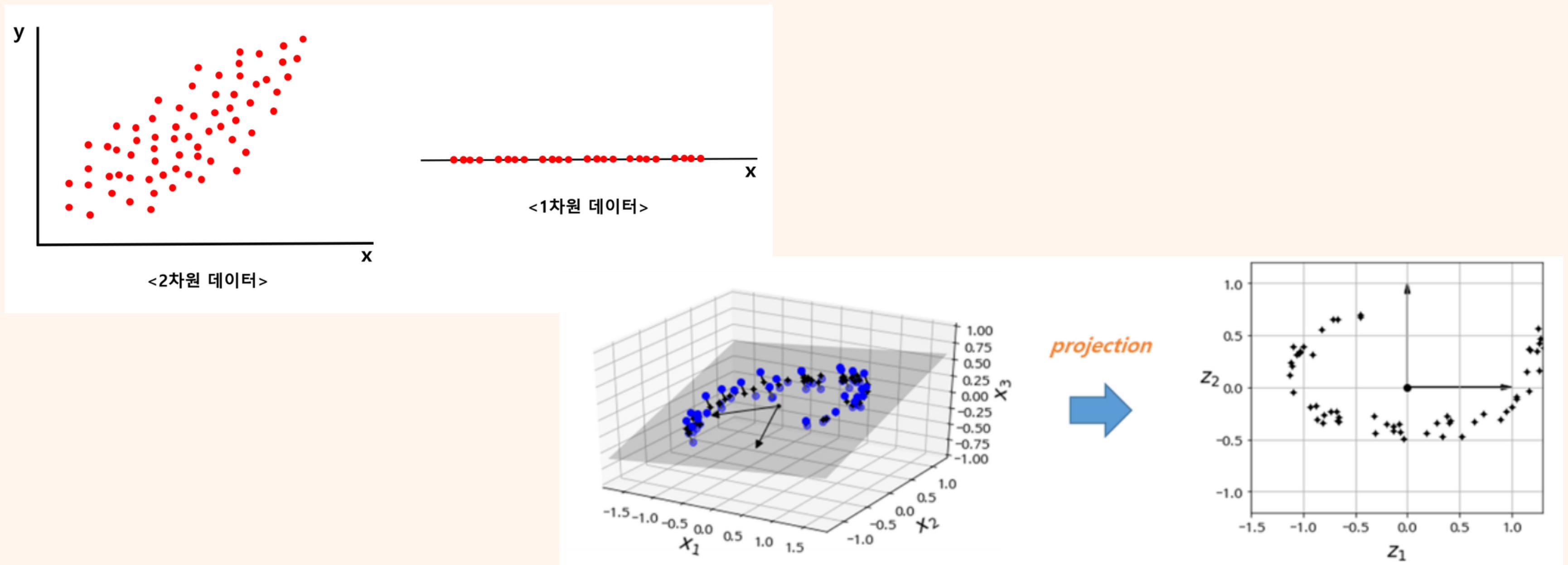


주성분 분석(PCA)

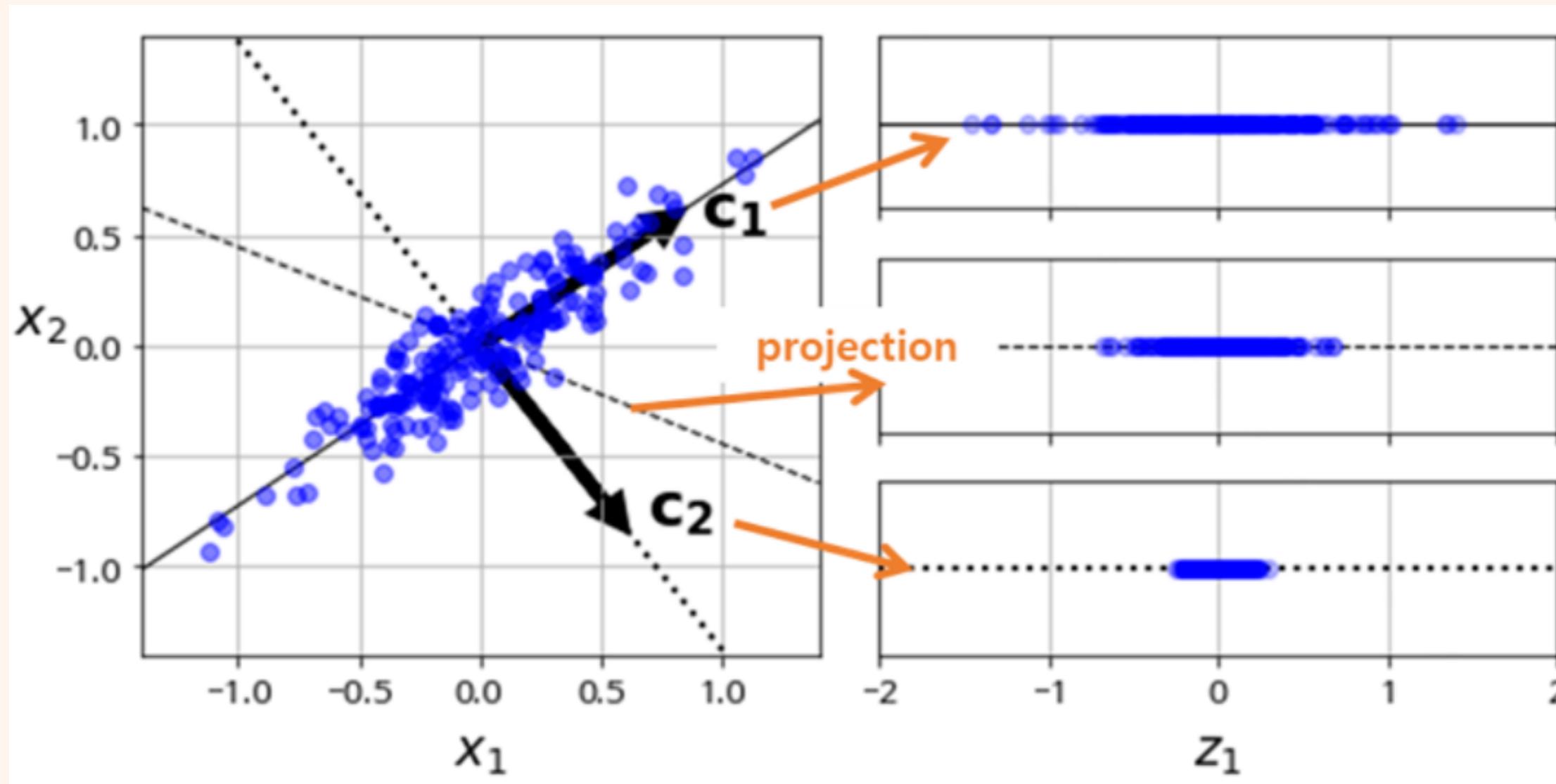
: 효율적인 저장, 관리

PCA의 목표

- 데이터를 정사영 시켜 차원을 낮추는 것



어떤 벡터에 정사영 시켜야 데이터 특성을 잘 유지할 수 있을까?



-> '주성분' : 데이터의 분산이 최대가 되는 벡터

주성분 구하는 방법

공분산 행렬과 선형변환

- 공분산 행렬은 데이터의 특징 쌍들의 변동이 얼마나 닮아있는지를 표현

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

a : x축 방향으로 퍼진 정도

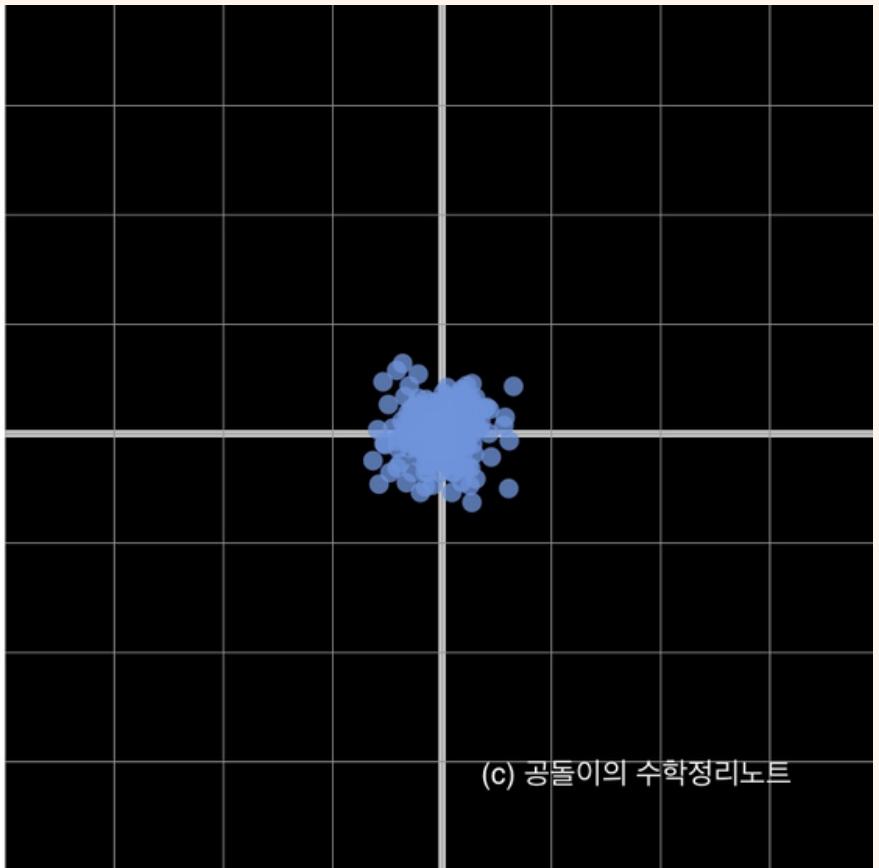
b, c : x, y축으로 함께 퍼진 정도

d : y축 방향으로 퍼진 정도

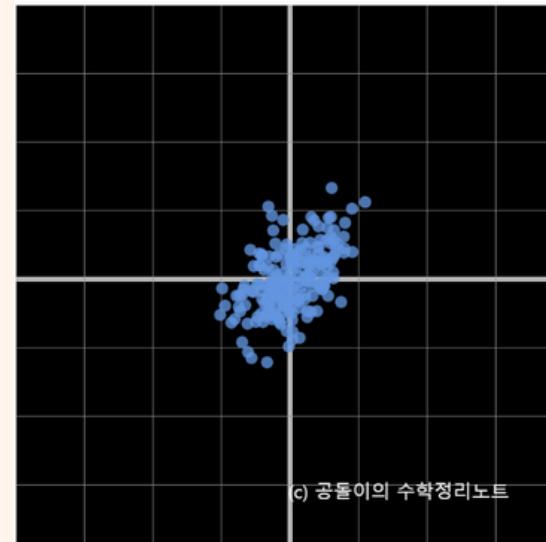
주성분 구하는 방법

공분산 행렬과 선형변환

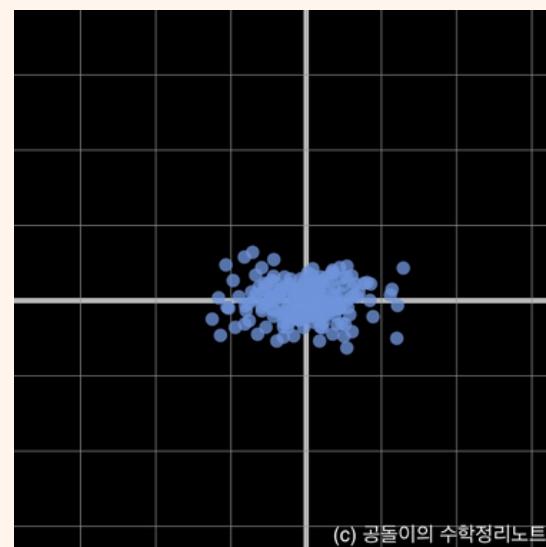
<선형변환>



$$\begin{bmatrix} 3 & 2 \\ 2 & 4 \end{bmatrix} \rightarrow$$



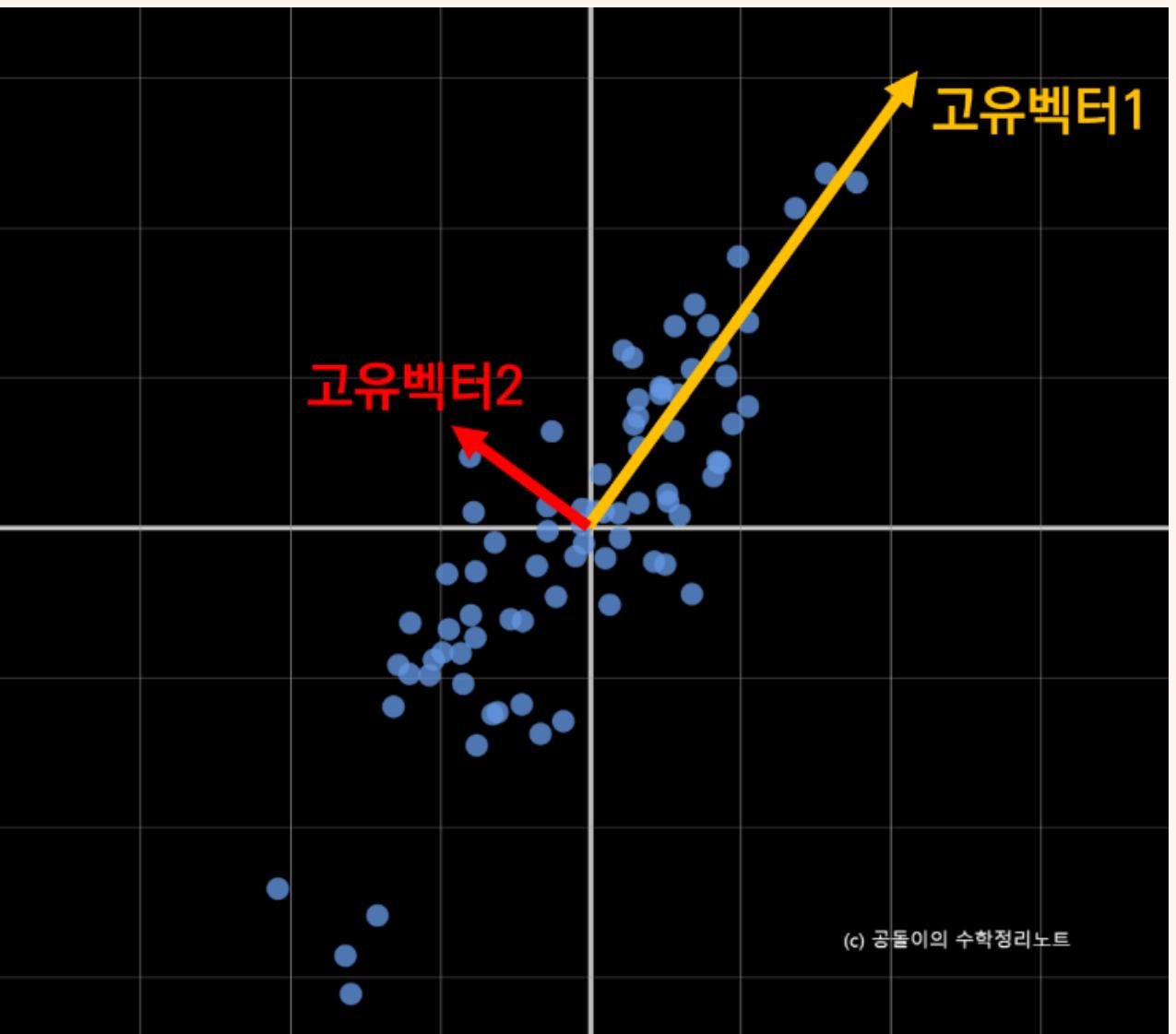
$$\begin{bmatrix} 5 & 0 \\ 0 & 1 \end{bmatrix} \rightarrow$$



주성분 구하는 방법

공분산 행렬의 고유값과 고유벡터

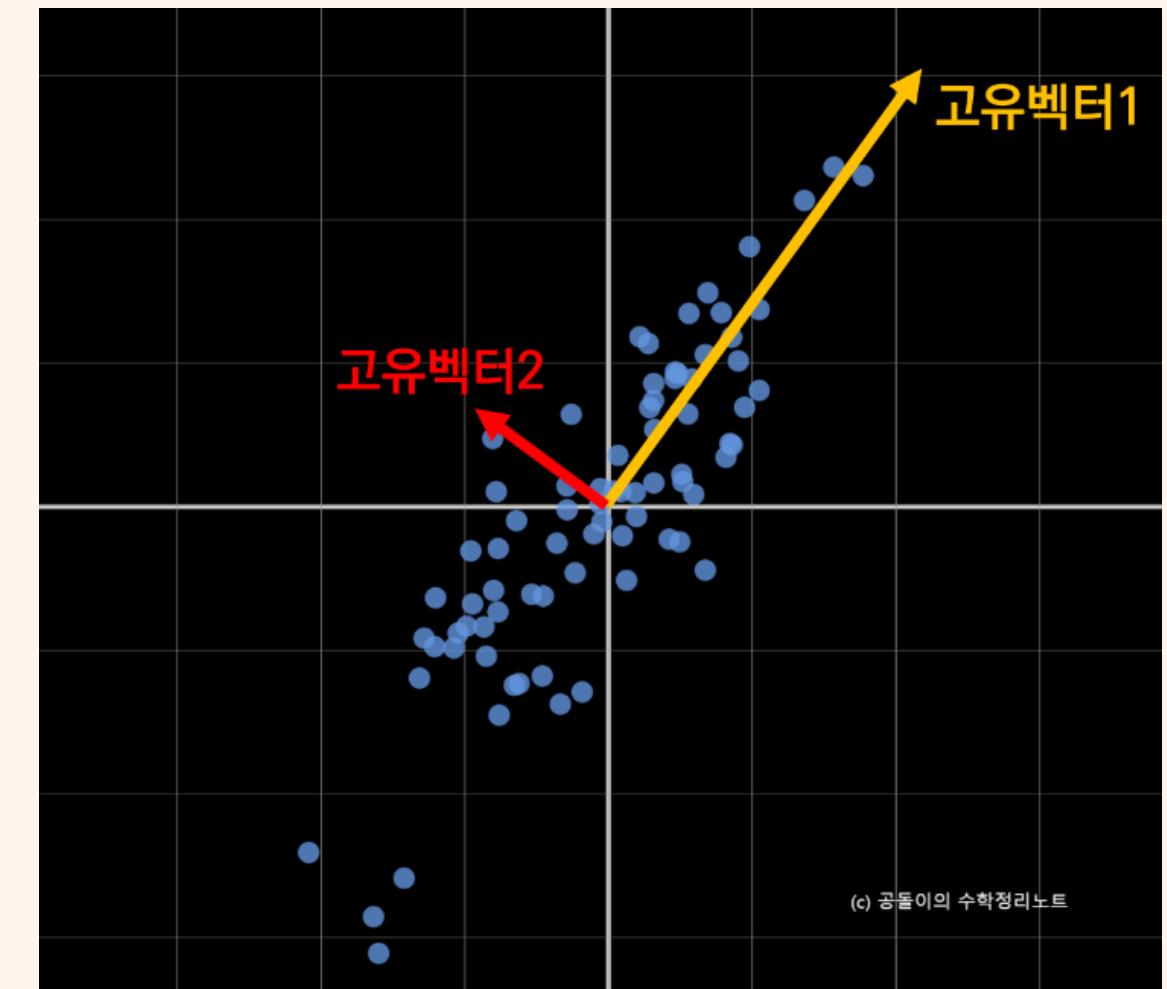
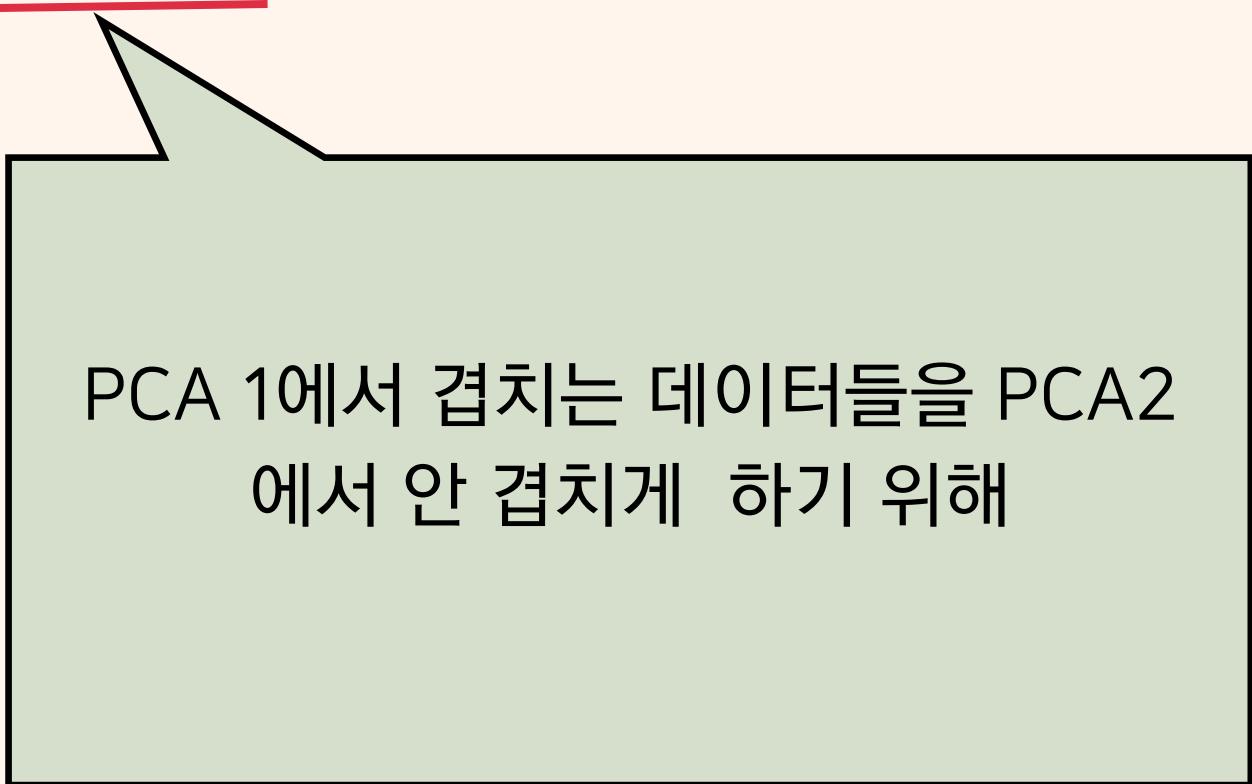
- 고유벡터는 행렬이 벡터에 작용하는 주축의 방향을 나타냄
- 고유값은 고유벡터 방향으로 얼마만큼 크게 늘려지는 지 나타냄



"공분산 행렬의 고윳값과 고유벡터를 이용해 분산을 최대로하는 벡터를 구할 수 있음"

주성분 분석 과정

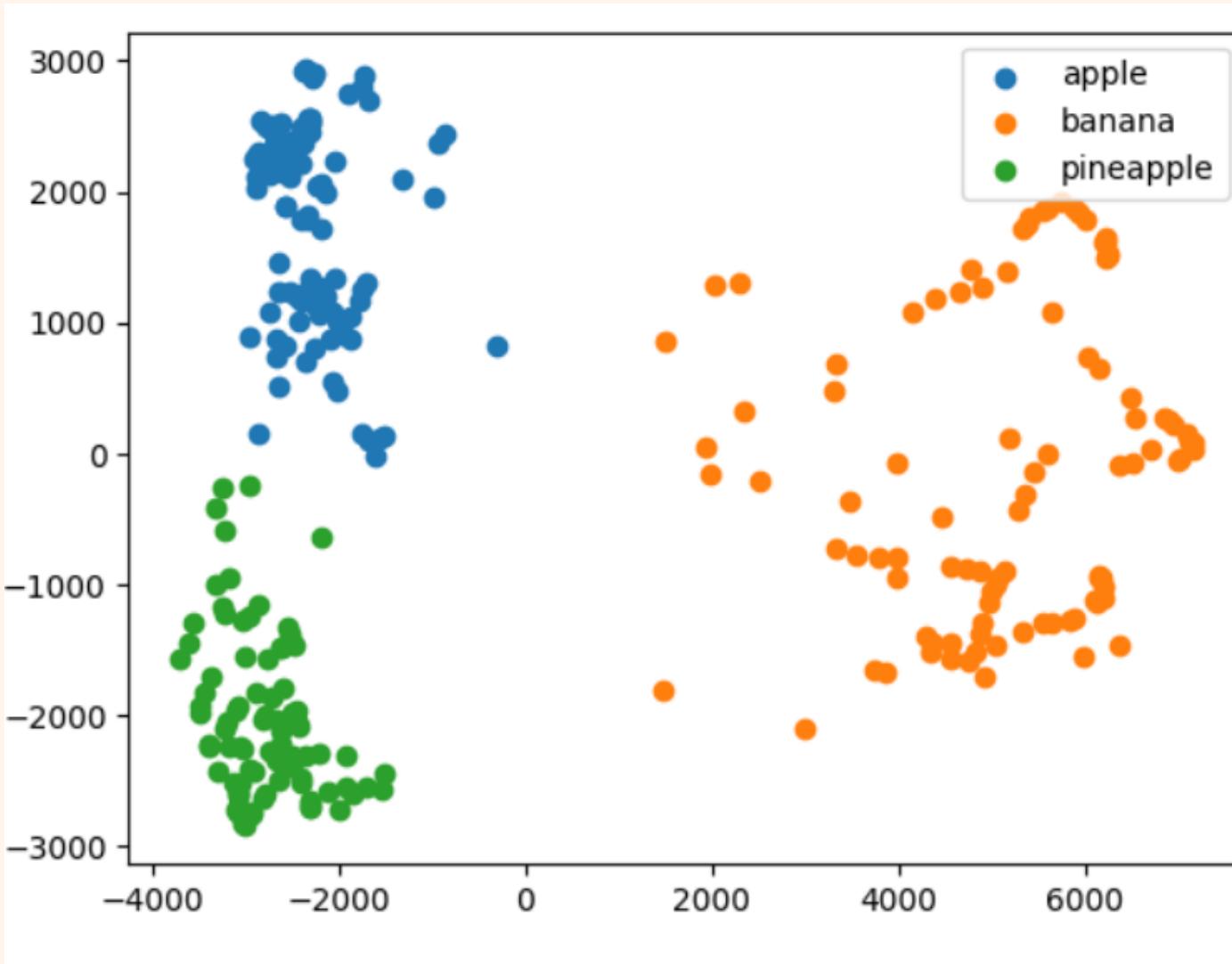
1. 데이터의 중심을 원점으로 이동한다.
2. 분산이 가장 큰 방향을 찾는다
3. 샘플 데이터를 주성분에 정사영 하여 차원을 줄인다.
3. 첫 번째 주성분에 수직이고 분산이 가장 큰 다음 방향을 두번째 주성분으로 한다.



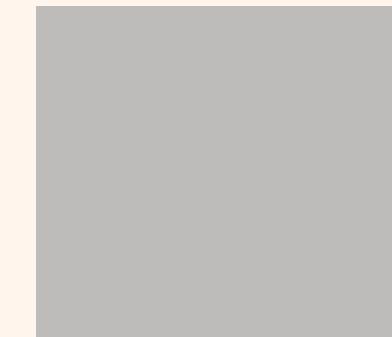
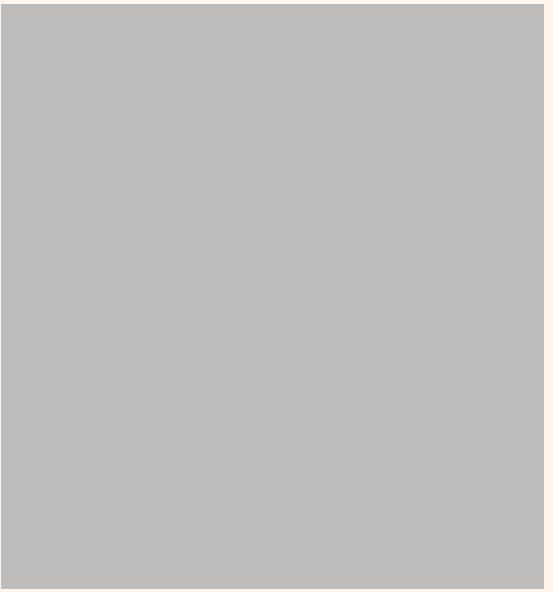
(c) 공동이의 수학정리노트

주성분 분석 장점

- 고차원의 데이터를 큰 정보손실없이 변환해 준다. 시각화 편리



- 데이터셋의 크기를 줄여 성능을 높이거나 훈련 속도를 빠르게 만들 수 있다.



Q & A