Python을 활용한 데이터 분석 강의

Unsupervised Learning

HYUNDAI MOBIS

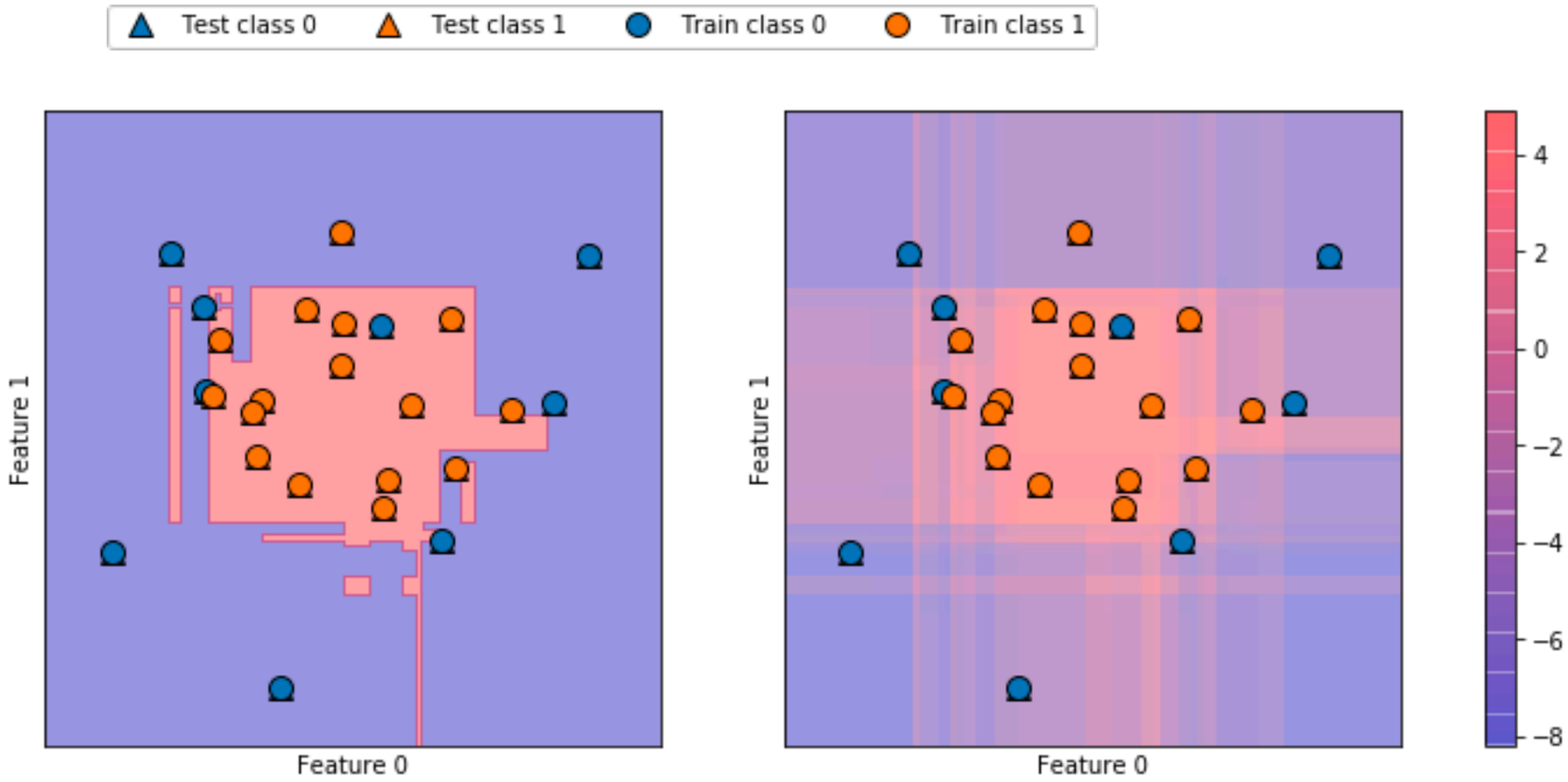
Uncertainty Estimates

- 얼마나 정확하게 분류(classify)를 했을까?

  - decision_function
  - predict_proba

Decision Function

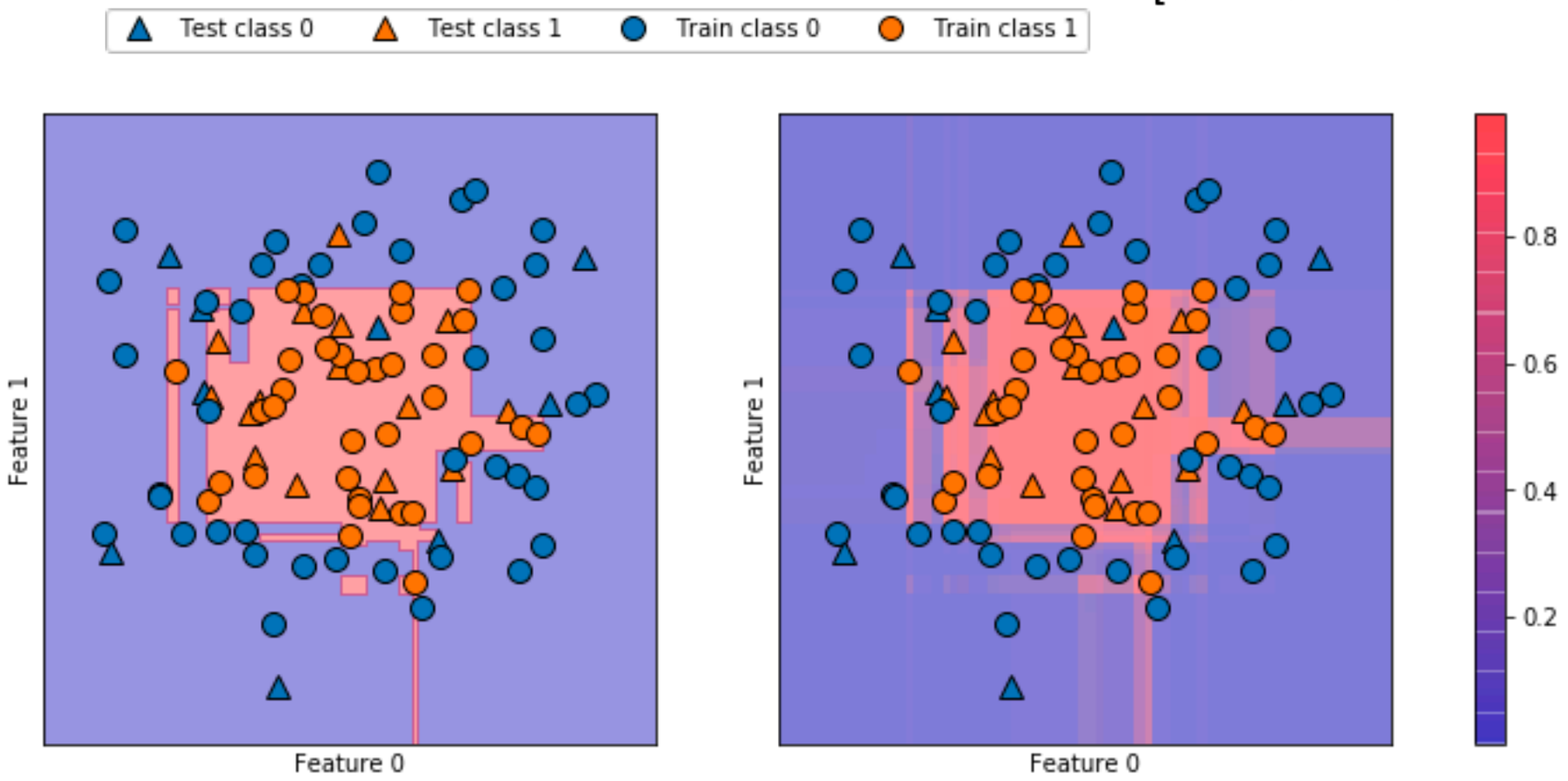
- +값을 가지면 ‘positive’ class에, -값을 가지면 ‘negative’ class에
  - positive: ‘1’ (class 1)
  - negative: ‘0’ (class 0)



Predicted Probabilities

- [class 0에 속할 확률, class 1에 속할 확률]

```
Predicted probabilities:
[[0.01573626 0.98426374]
 [0.84575649 0.15424351]
 [0.98112869 0.01887131]
 [0.97406775 0.02593225]
 [0.01352142 0.98647858]
 [0.02504637 0.97495363]]
```



Unsupervised Learning

- 답을 주지 않고 input data로 leaning
  - transformation, clustering

- Transformation: 데이터를 더 잘 이해하도록 하기 위해 데이터를 변형
  - 차원 축소
    - ex. feature 줄이기
  - 데이터 보완
    - ex. topic extraction
- Clustering: 비슷한 데이터끼리 군집화
  - ex. iphone 사진첩

Unsupervised Learning

- hard to evaluate!
- 답을 모르기 때문에 모델이 잘 설계가 되었는지 판단하기 어려움

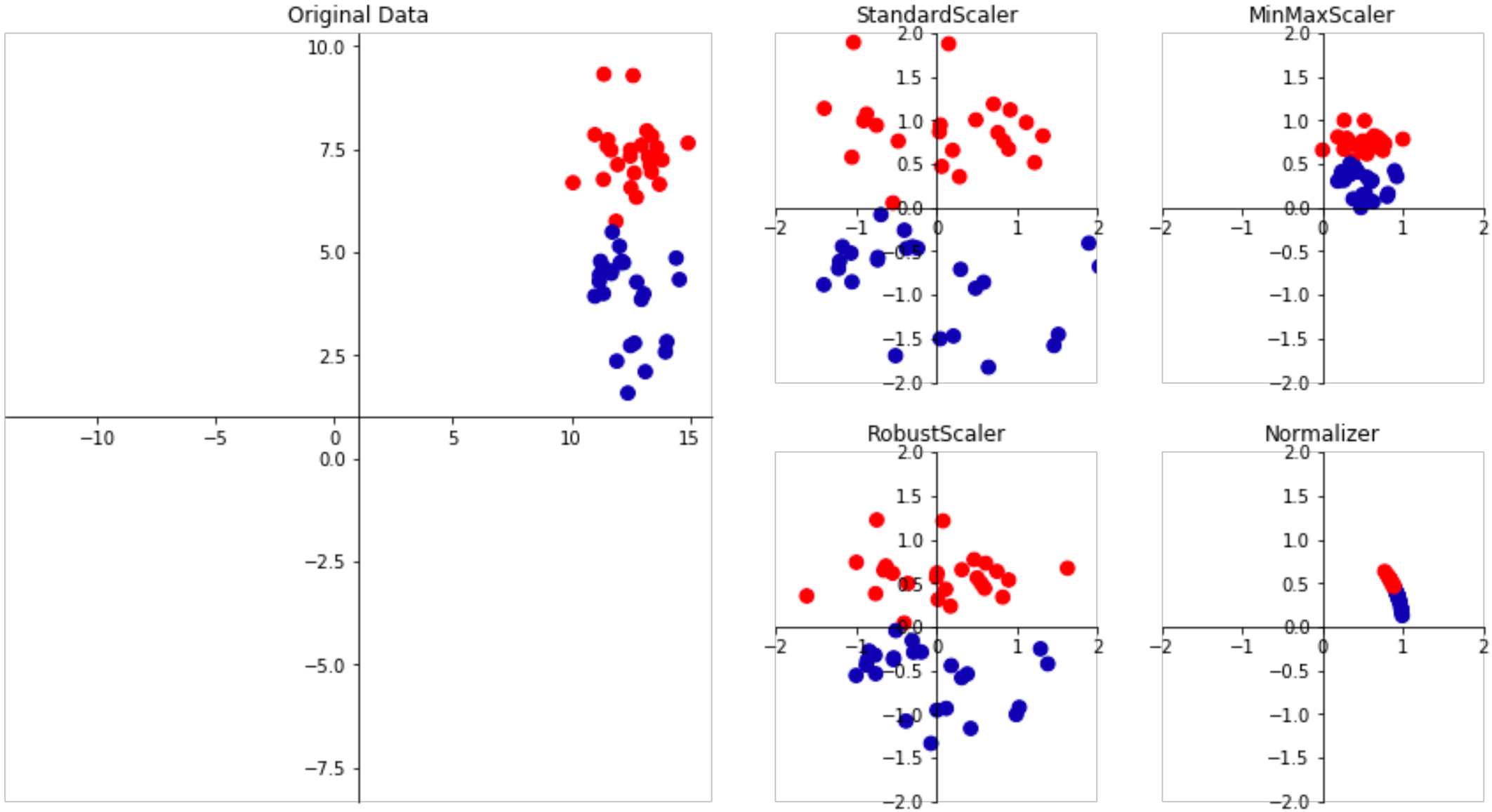
그래서

…

데이터를 더 잘 이해하기 위해

supervised learning의 전처리 개념으로(정확도 향상에 도움)

Preprocessing & Scaling



Preprocessing & Scaling

Scaling

자료에 선형 변환을 적용하여 자료의 overflow나 underflow를 방지

- **StandardScaler**
  - 각 feature의 평균을 0, 분산을 1로 맞춰줌
- **RobustScaler**
  - median과 quartiles를 사용
- **MinMaxScaler**
  - 최대, 최소값이 0, 1이 되도록 스케일링
- **Normalizer**
  - 모든 데이터를 Euclidean 거리 1을 가진 벡터로 변환

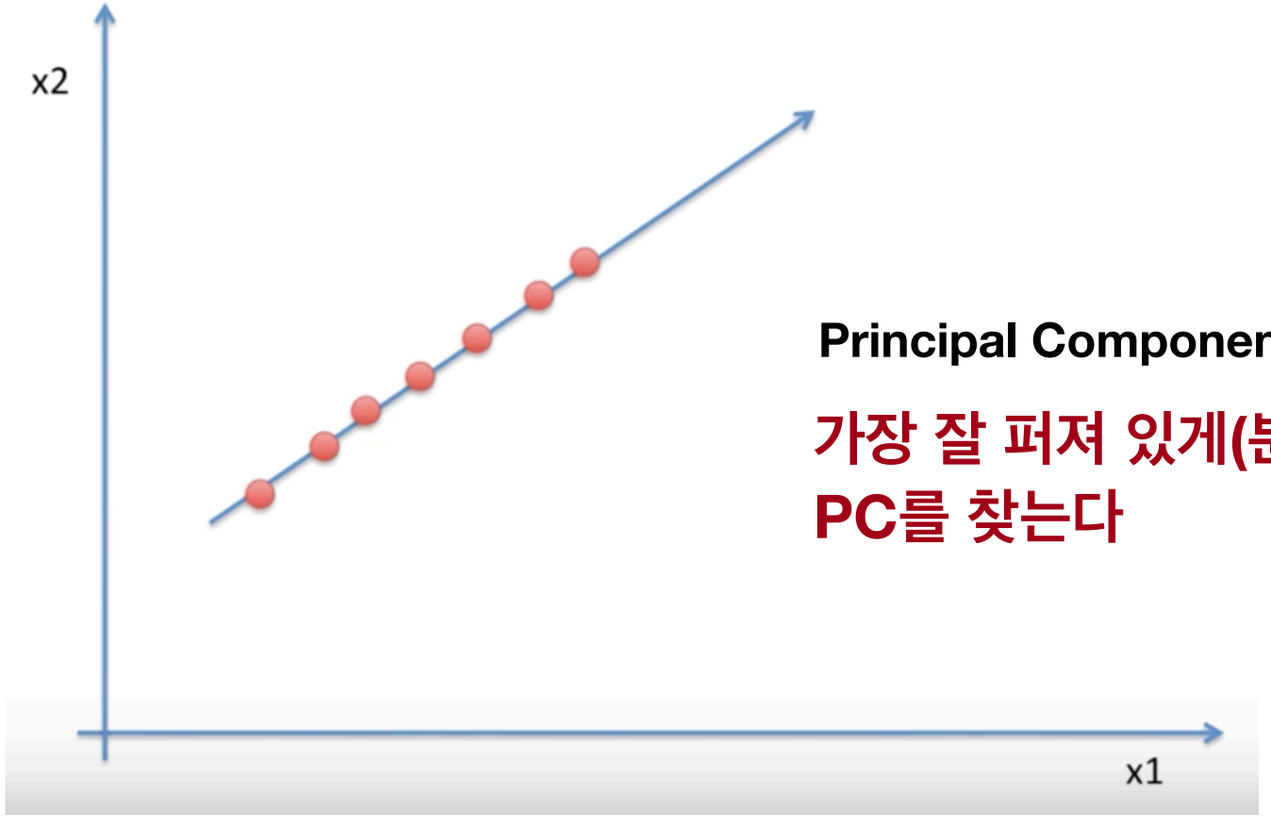
PCA
(Principal Component Analysis)

- feature들이 통계적으로 상관관계가 없도록 dataset을 회전
- 회전한 뒤 중요도에 따라 일부 feature만 선택

HYUNDAI MOBIS

PCA
(Principal Component Analysis)

- feature들이 통계적으로 상관관계가 없도록 dataset을 회전
- 회전한 뒤 중요도에 따라 일부 feature만 선택



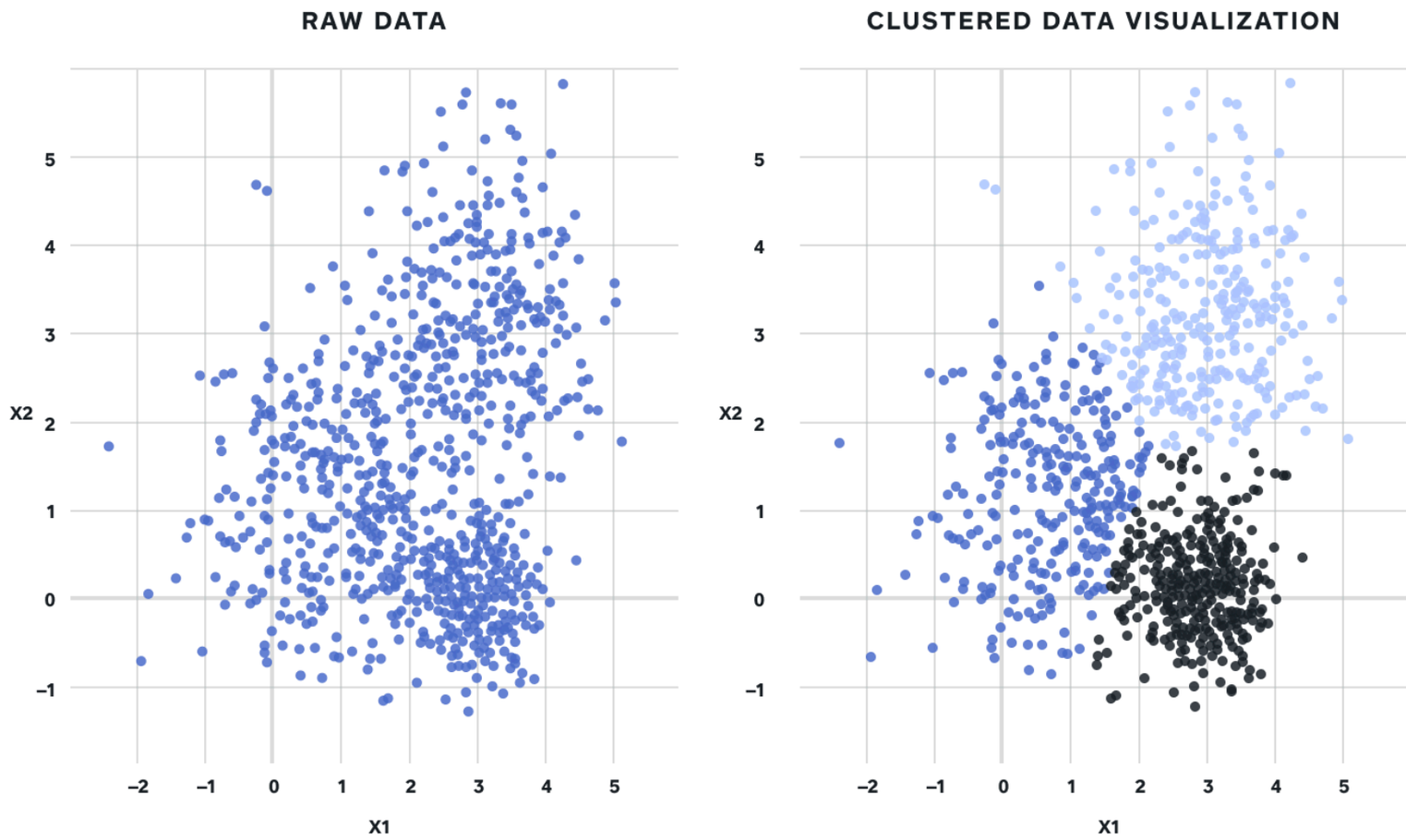
Principal Component(PC)

가장 잘 퍼져 있게(분산이 제일 크게)
PC를 찾는다

HYUNDAI MOBIS

Clustering

- 비슷한 특징을 가진 데이터끼리 군집화
  - k-Means
  - agglomerative
  - DBSCAN

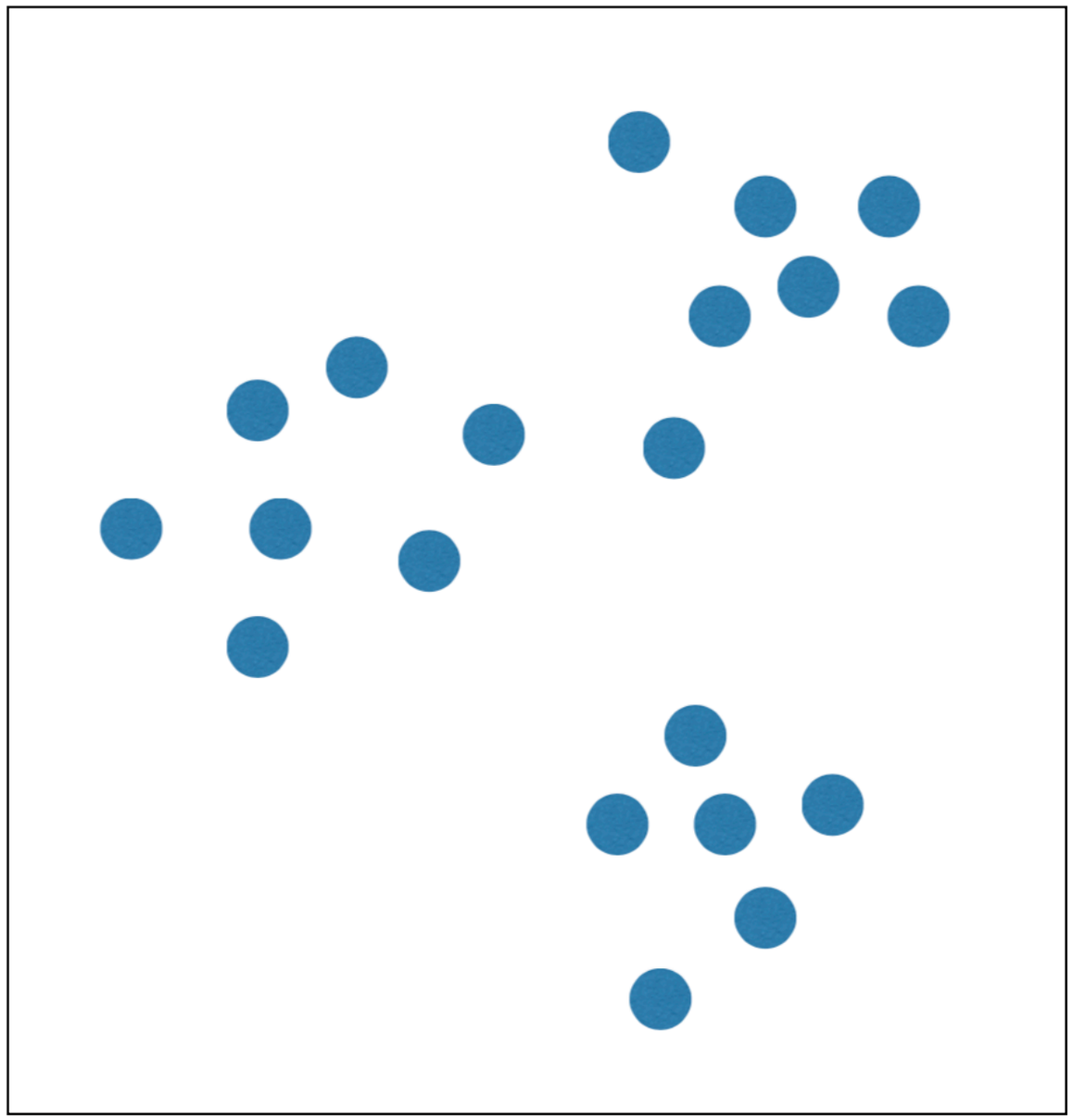


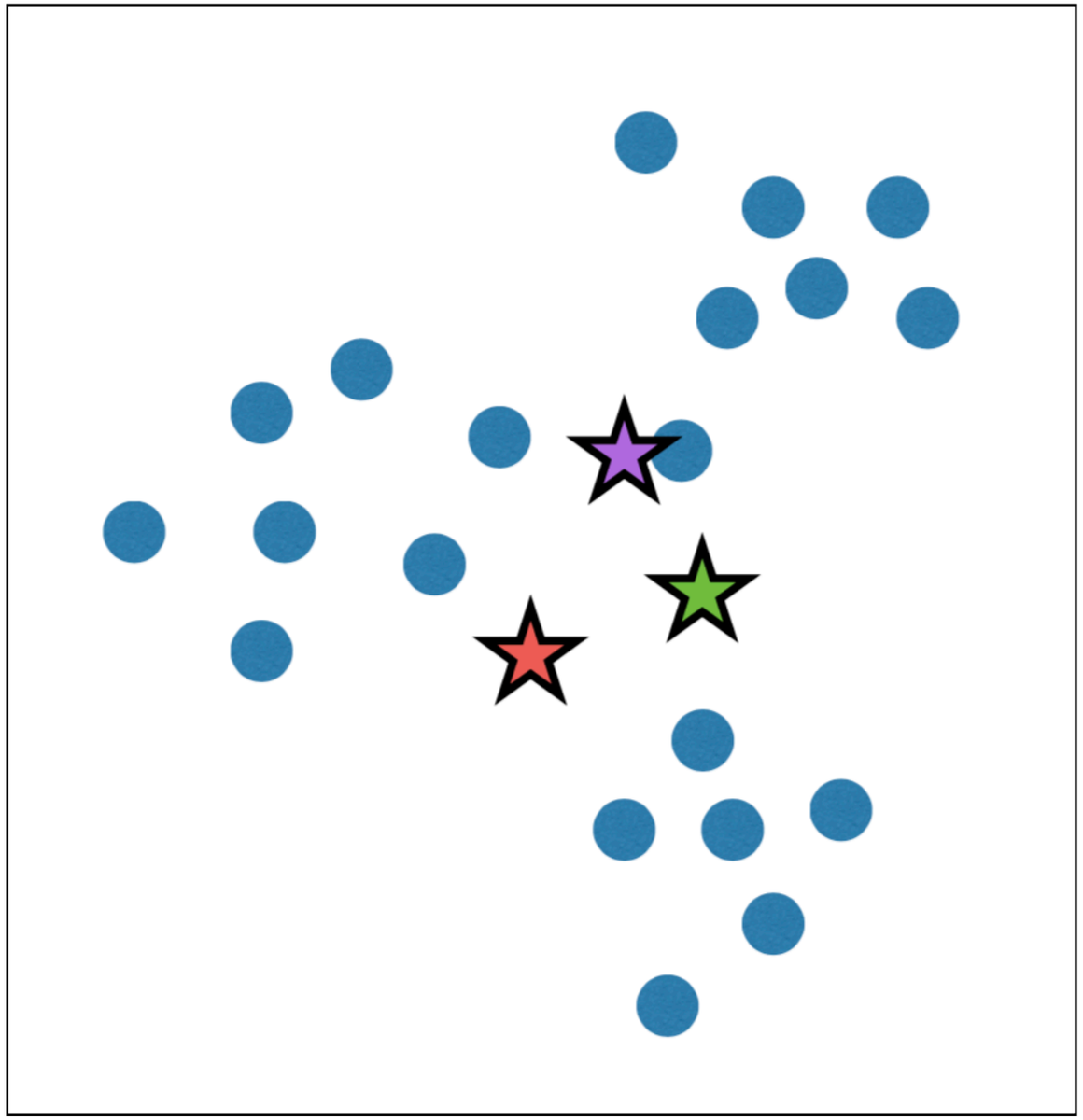
k-Means

- 간단하고 가장 보편적으로 사용되는 clustering 알고리즘
- k개의 cluster를 찾는 알고리즘

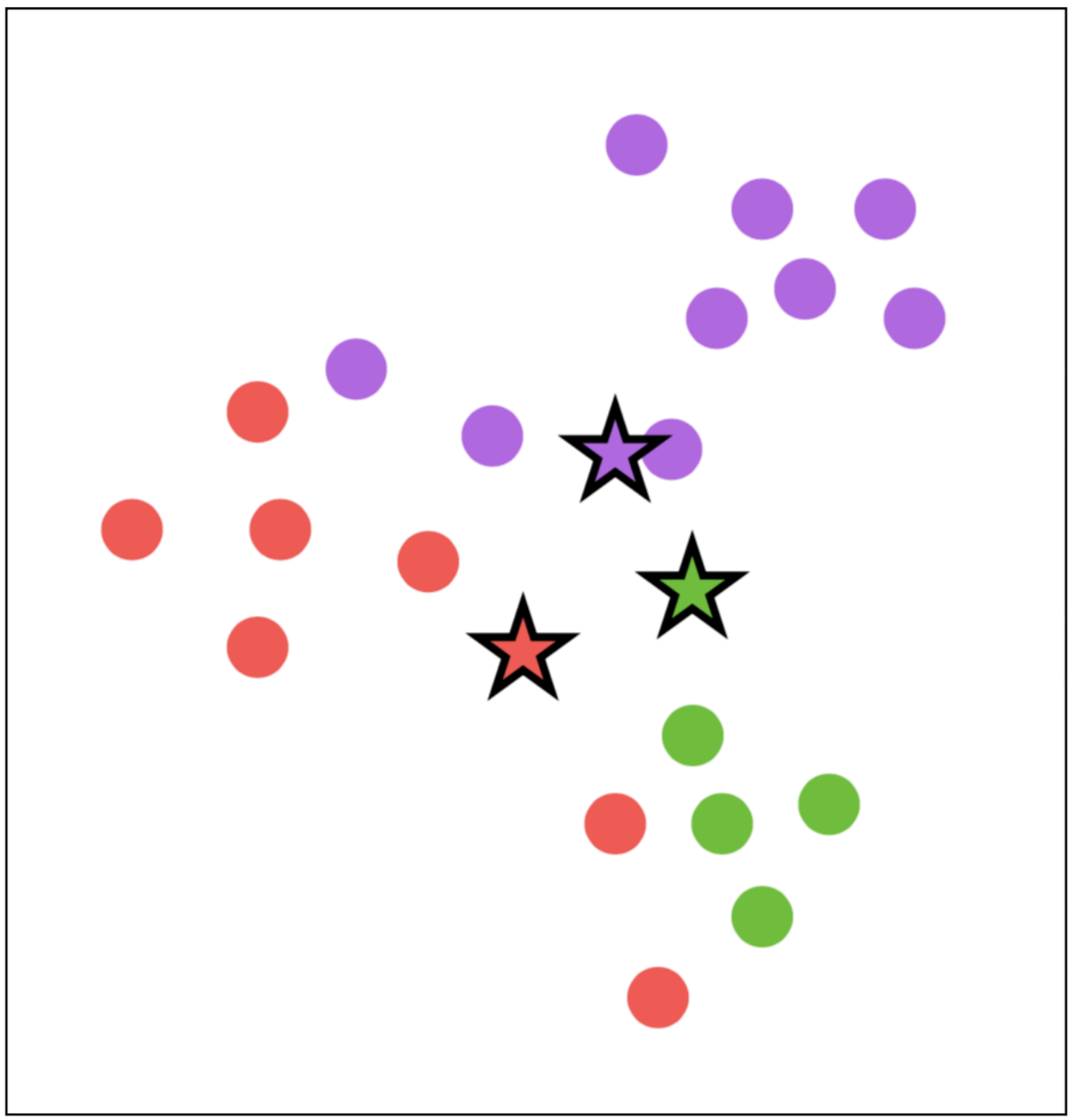
- k개의 '클러스터 중심점(cluster center)'을 데이터 공간에 뿌리고
- 각 center과 가까운 데이터 점들을 해당 center의 cluster로 할당
- 각 cluster의 데이터 점들을 각각 평균내서 새로운 center를 찾음
- 변화가 없을 때까지 반복

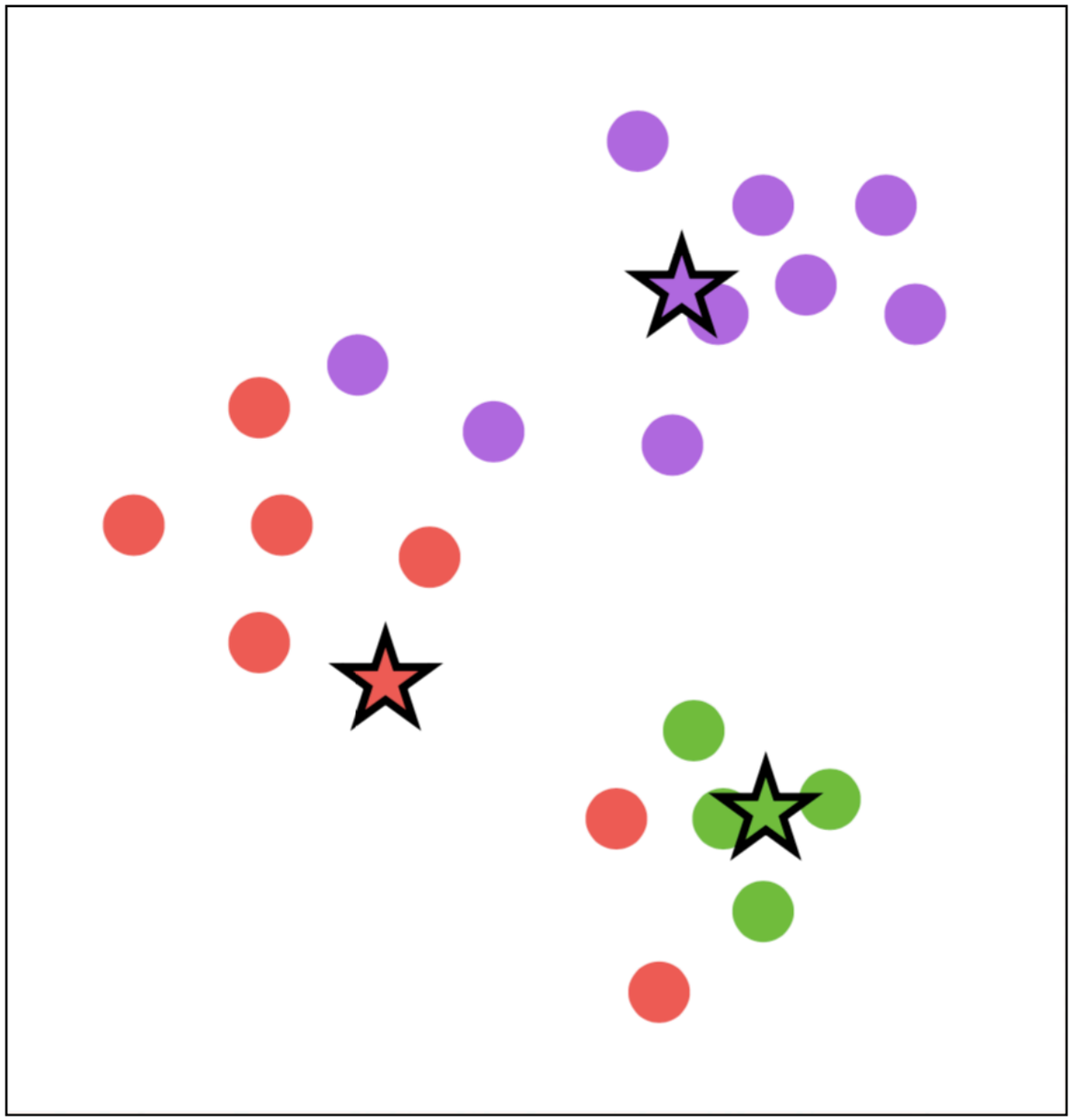
k-Means



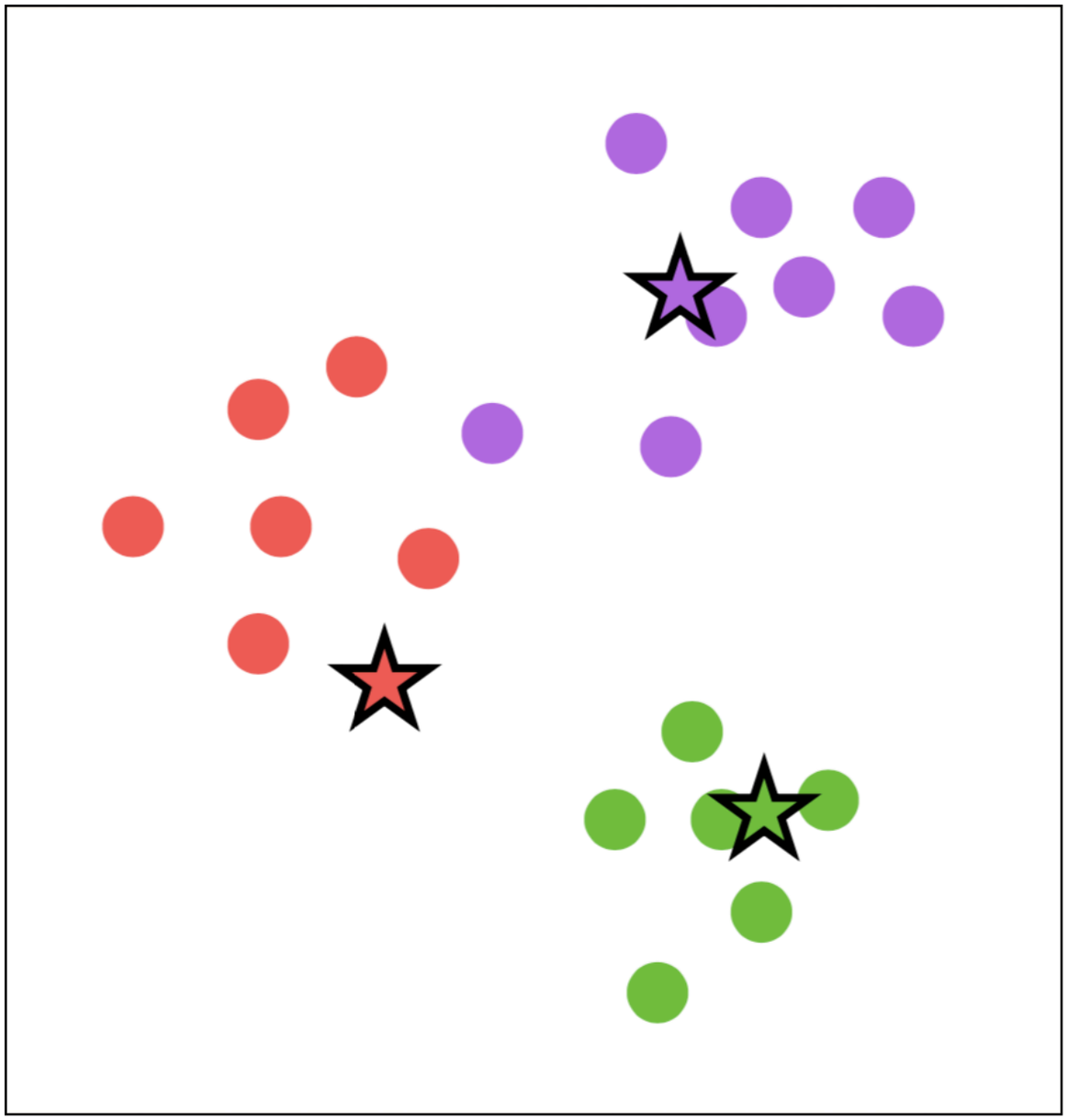


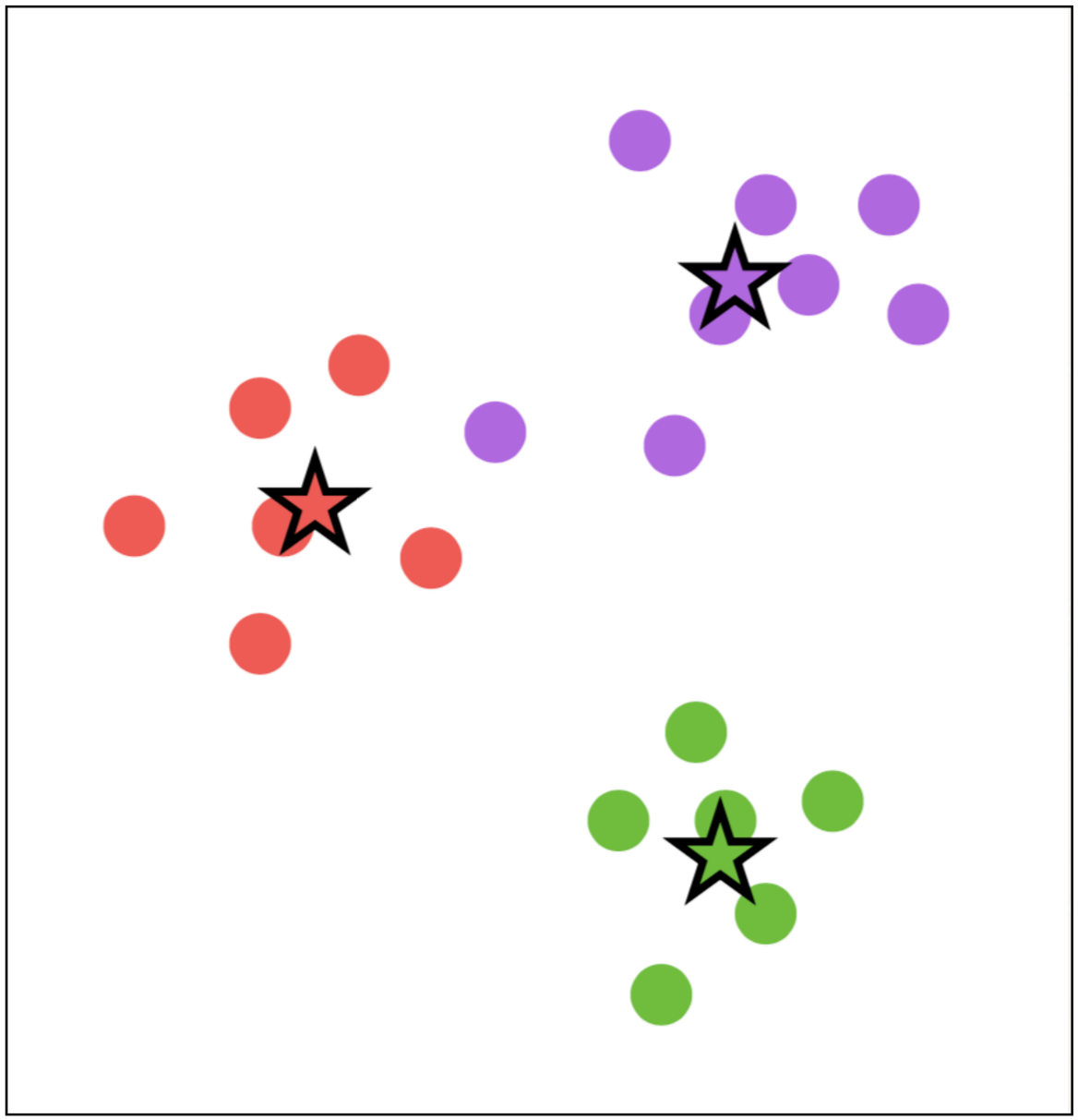
k-Means



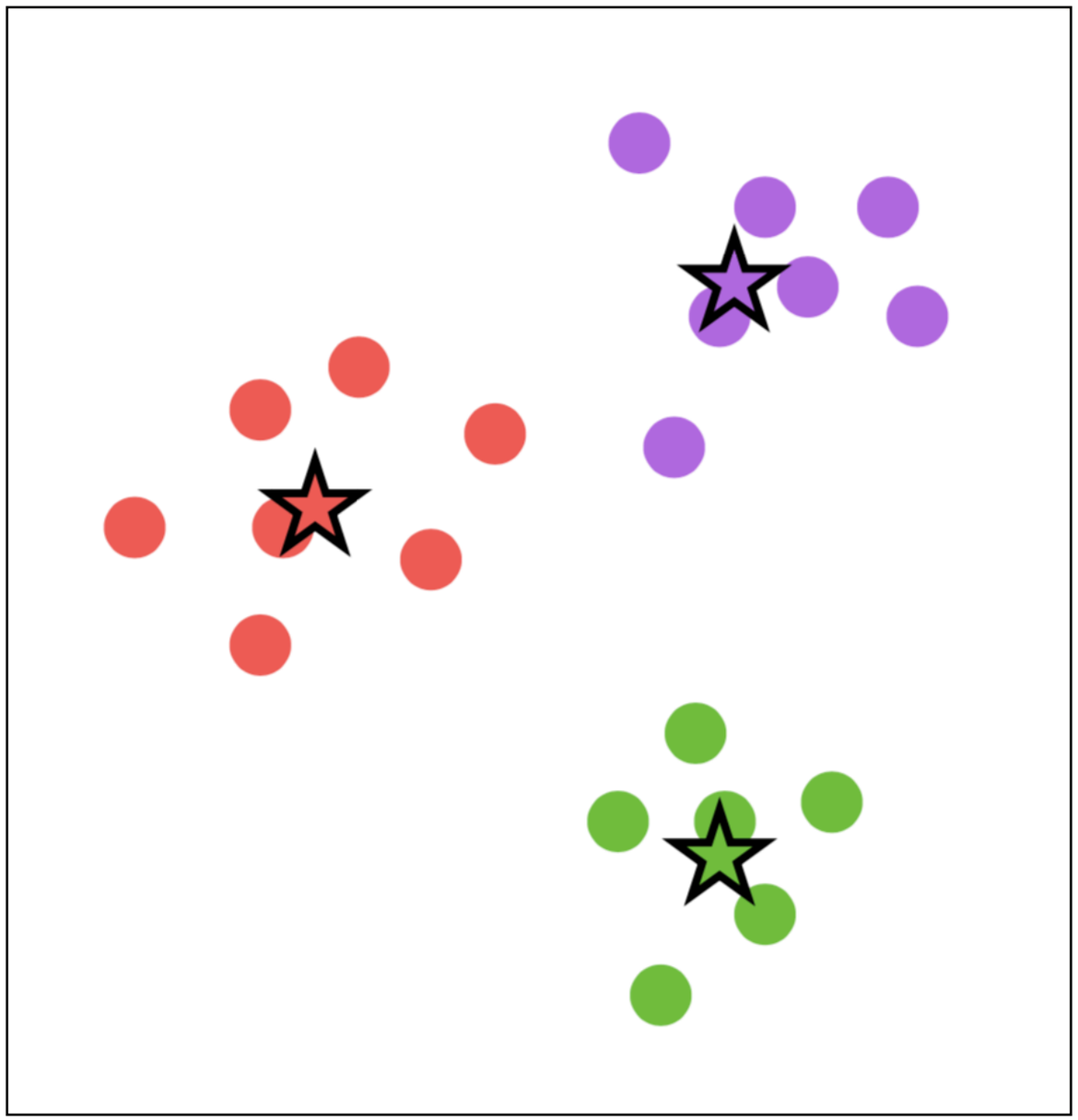


k-Means

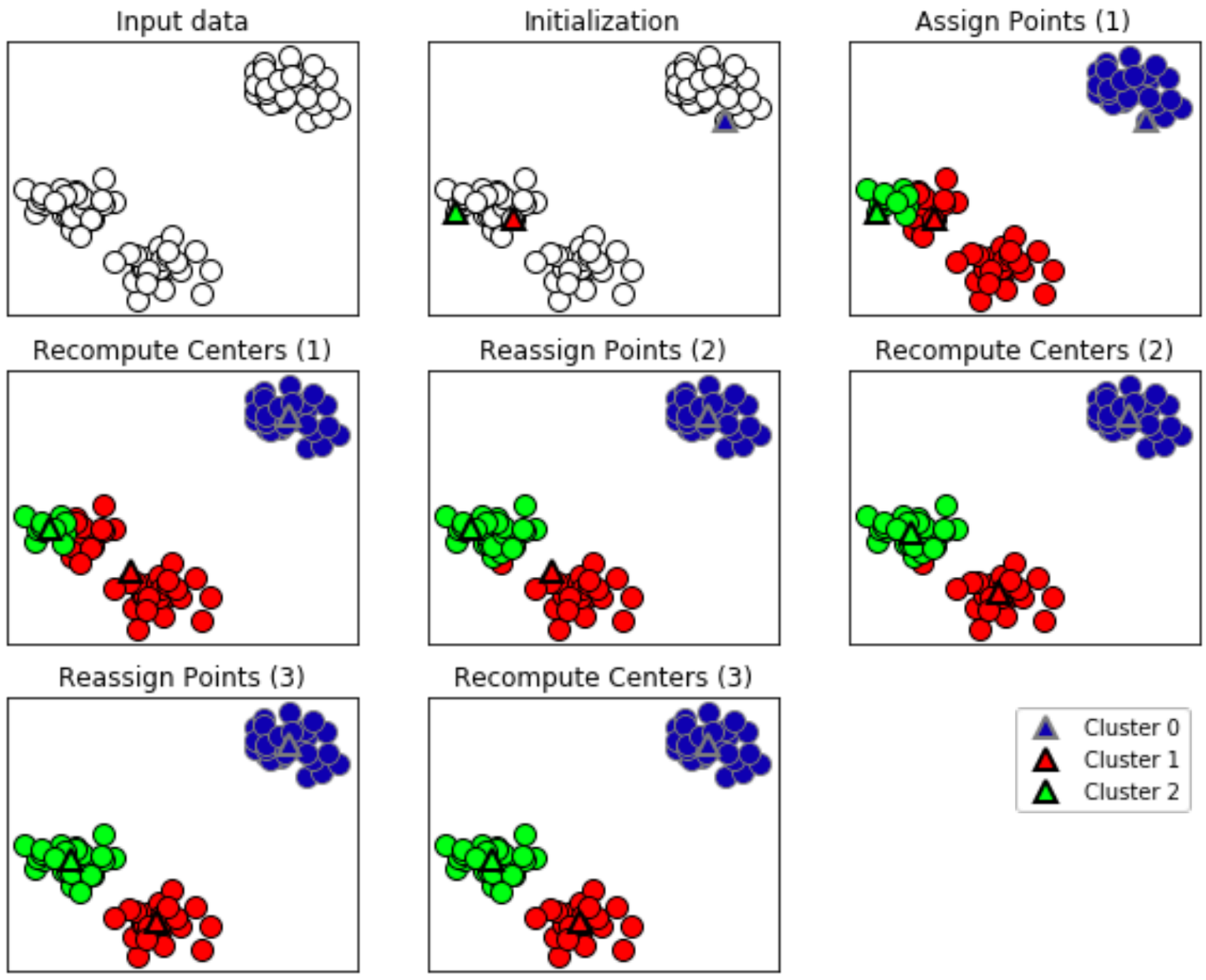




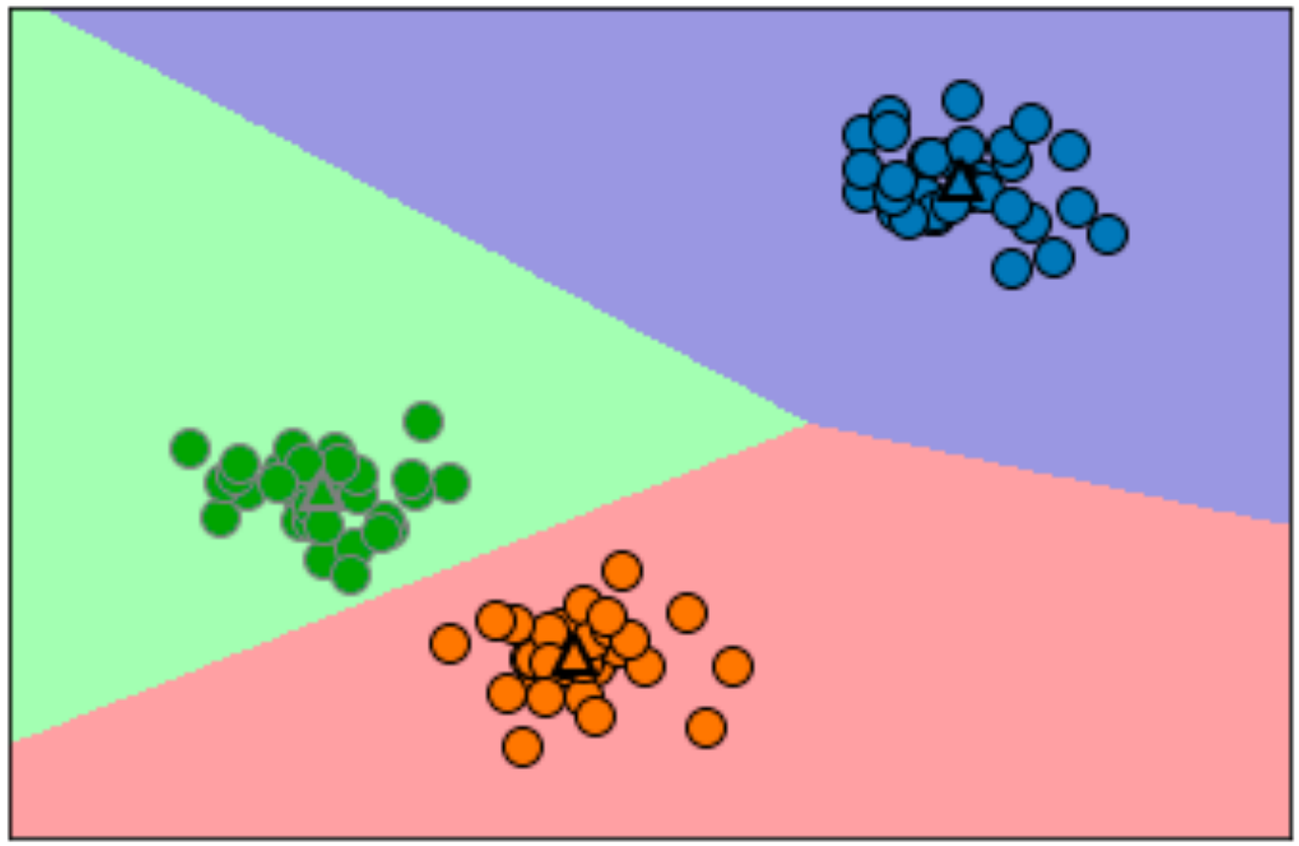
k-Means



k-Means

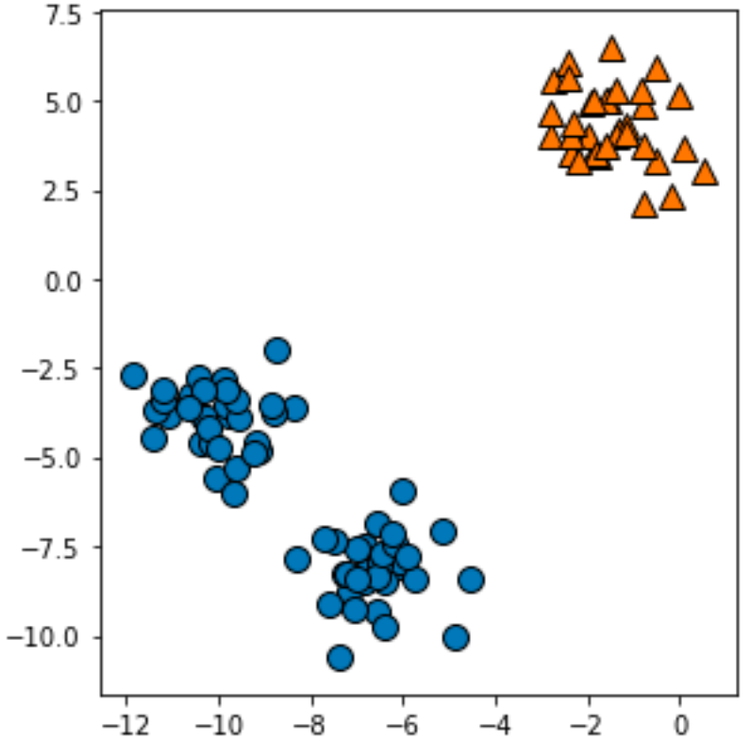


k-Means

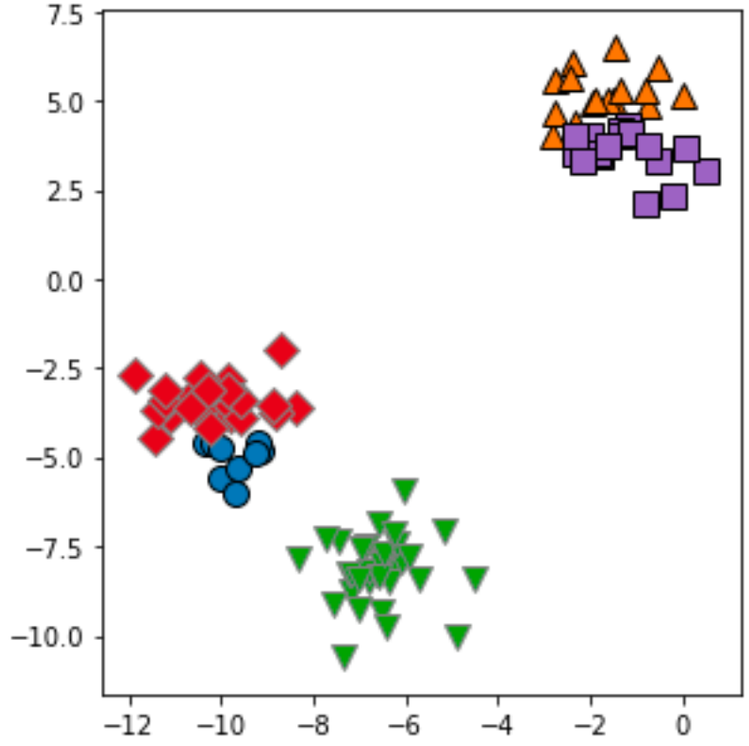


k-Means

- n_clusters: cluster의 개수
- kmeans.labels_: 각 데이터가 할당된 cluster
- cluster_centers_: cluster의 중심



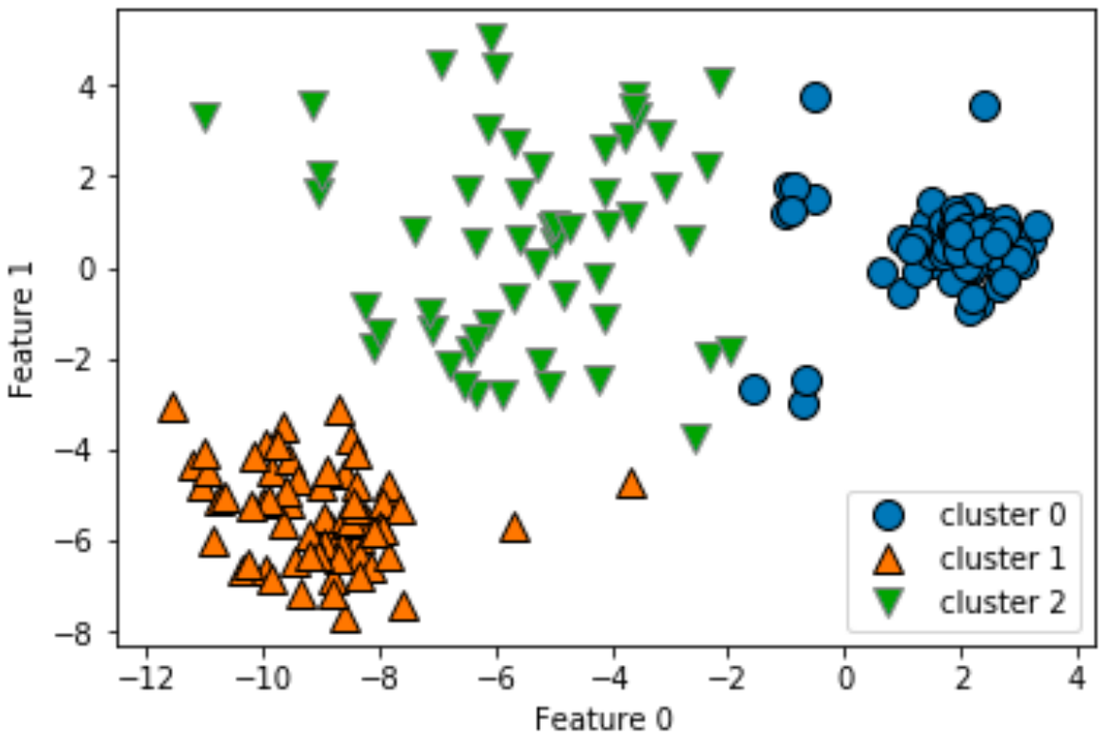
n_clusters=2

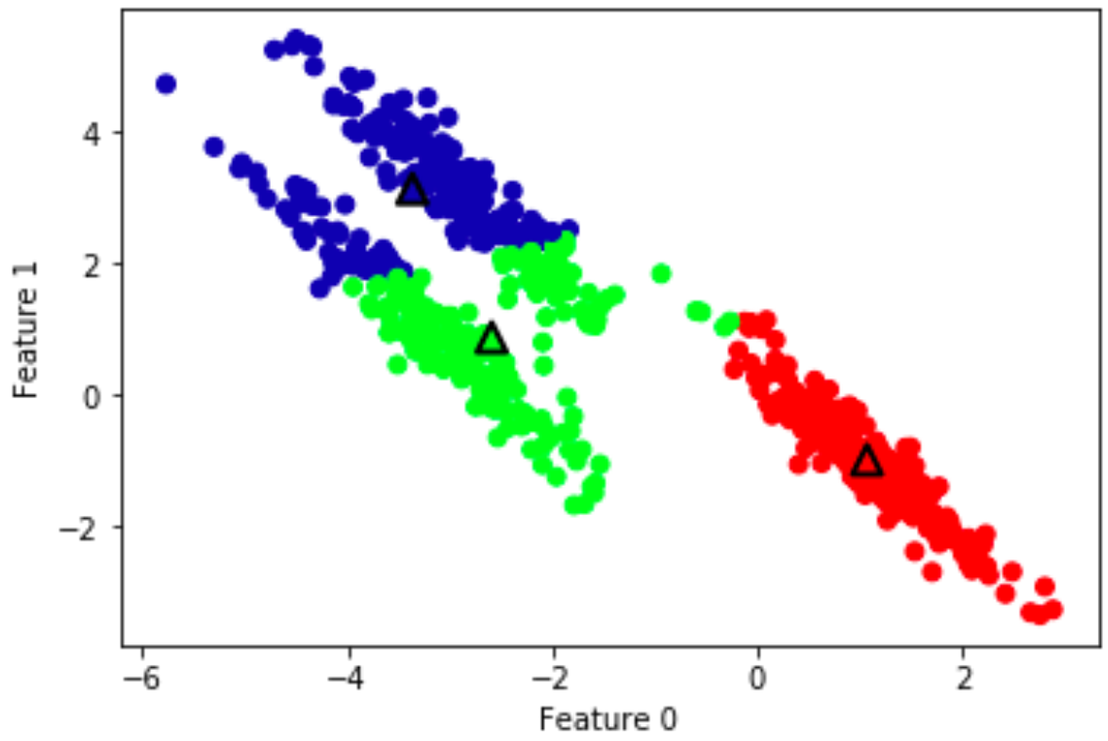


n_clusters=5

k-Means

- failure cases of k-means

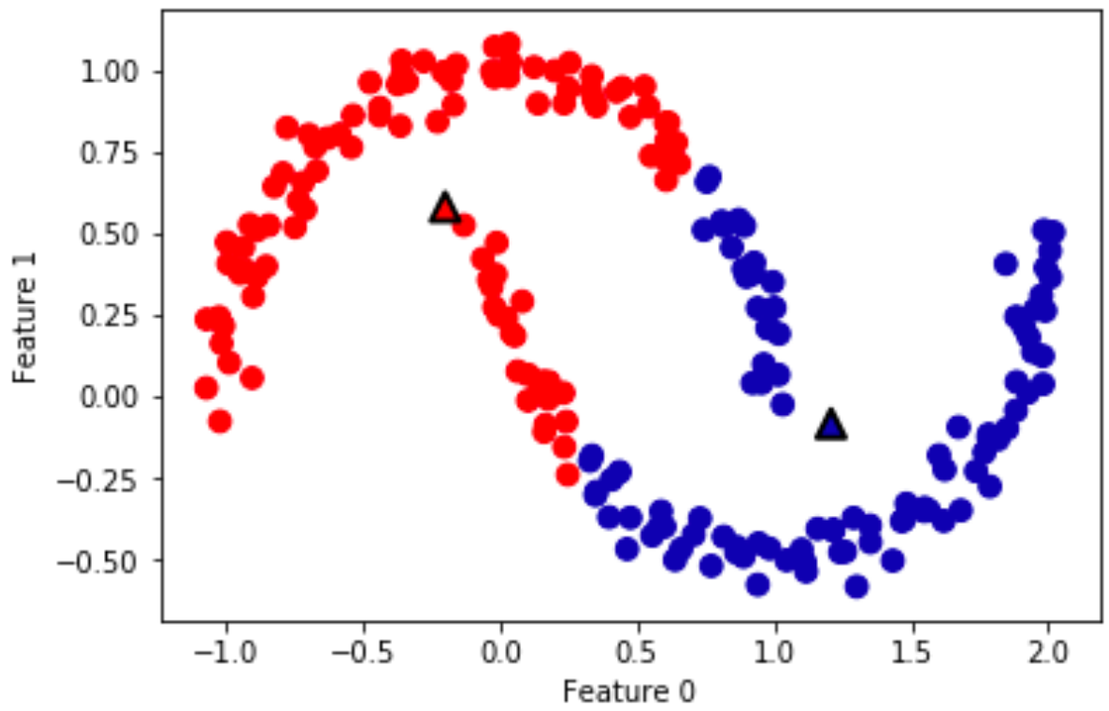




HYUNDAI MOBIS

k-Means

- failure cases of k-means

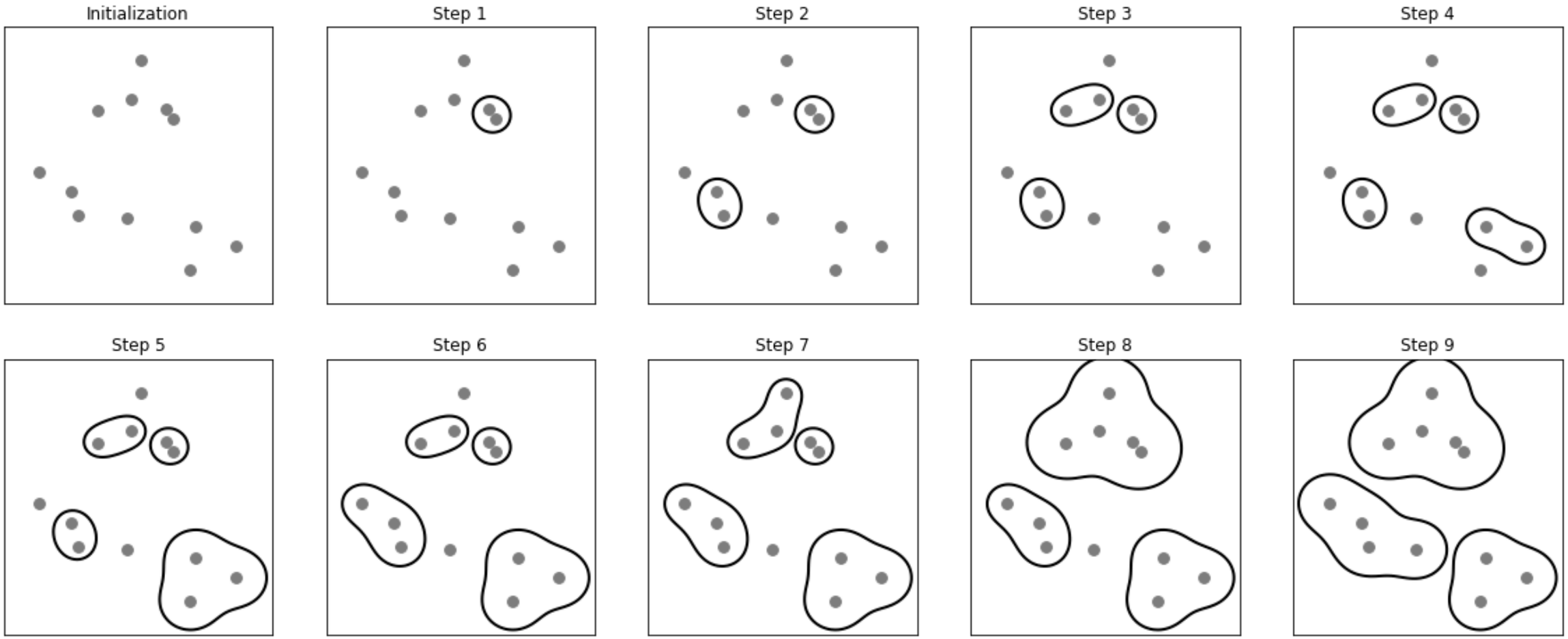


k-Means

- Pros
  - runs quickly
  - 비교적 이해하기 쉽고 구현도 쉬움
  - large dataset도 쉽게 scaling할 수 있음
- Cons
  - random한 초기 설정에 의존
    - scikit-learn은 default로 알고리즘을 randomly 10번 돌리고 가장 좋은 결과를 보여줌
  - cluster의 모양을 가정하고 있어서 활용 범위가 제한적

Agglomerative Clustering

- 시작할 때 각 포인트를 하나의 클러스터로 지정하고, 어떠한 종료 조건을 만족할 때까지 가장 비슷한 두 클러스터를 합쳐나가는 방식



Agglomerative Clustering

- linkage

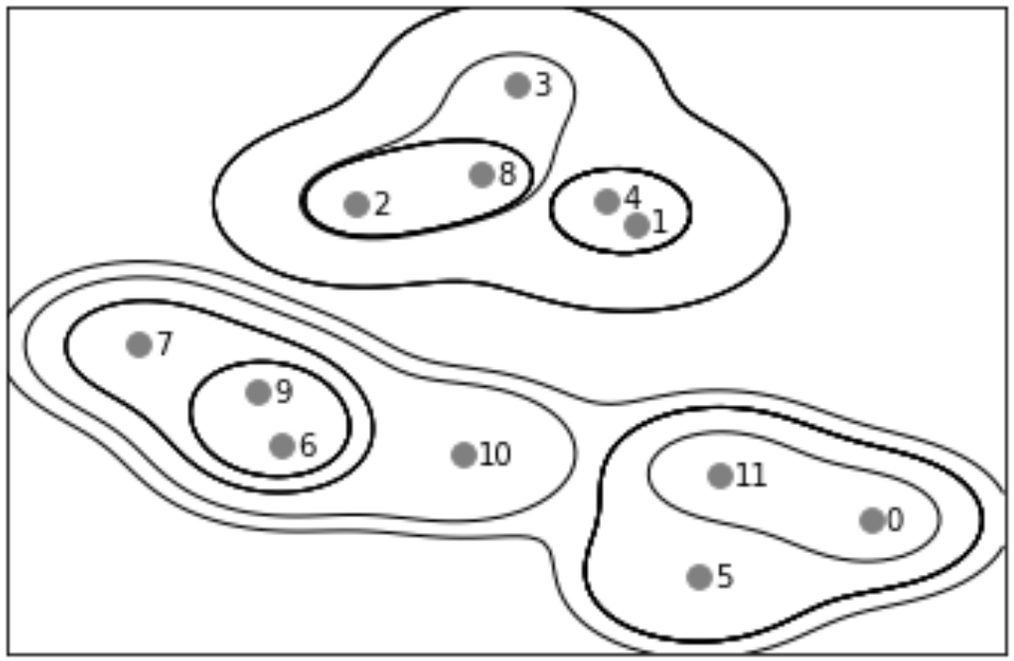
  - ward(default)
    - cluster 내의 분산을 가장 작게 증가시키는 두 클러스터를 합침
    - 그래서 크기가 비교적 비슷한 클러스터가 만들어짐

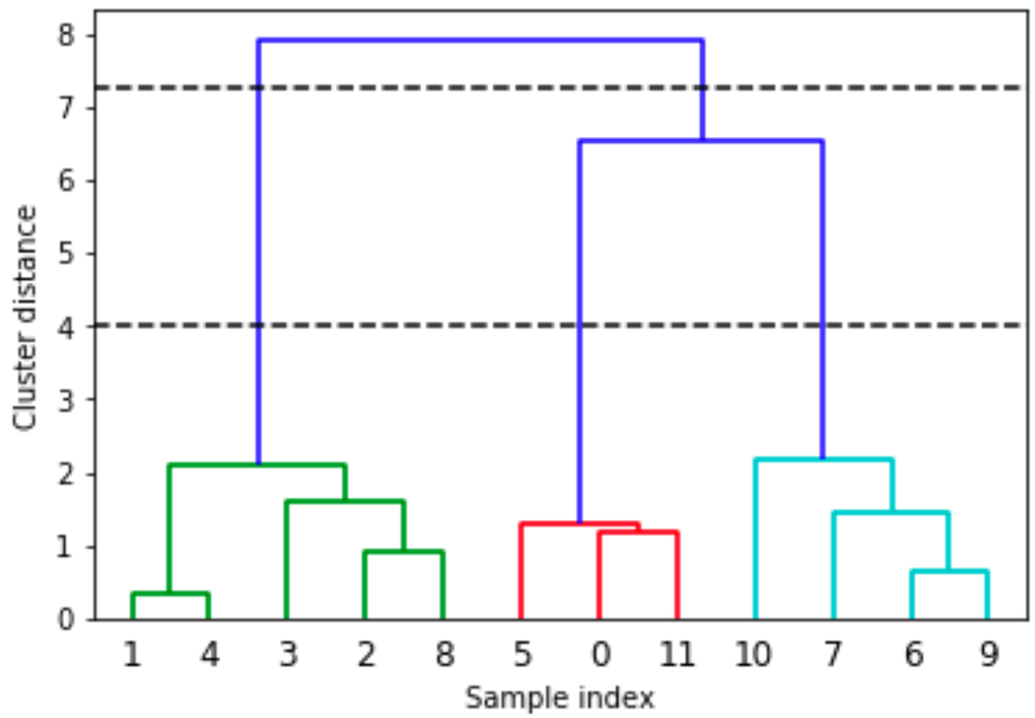
  - average
    - cluster 사이의 평균 거리가 가장 짧은 두 클러스터를 합침

  - complete(maximum)
    - cluster 사이의 최대 거리가 가장 작은 두 클러스터를 합침

Agglomerative Clustering

- 계층적 군집





DBSCAN

“density-based spatial clustering of applications with noise”

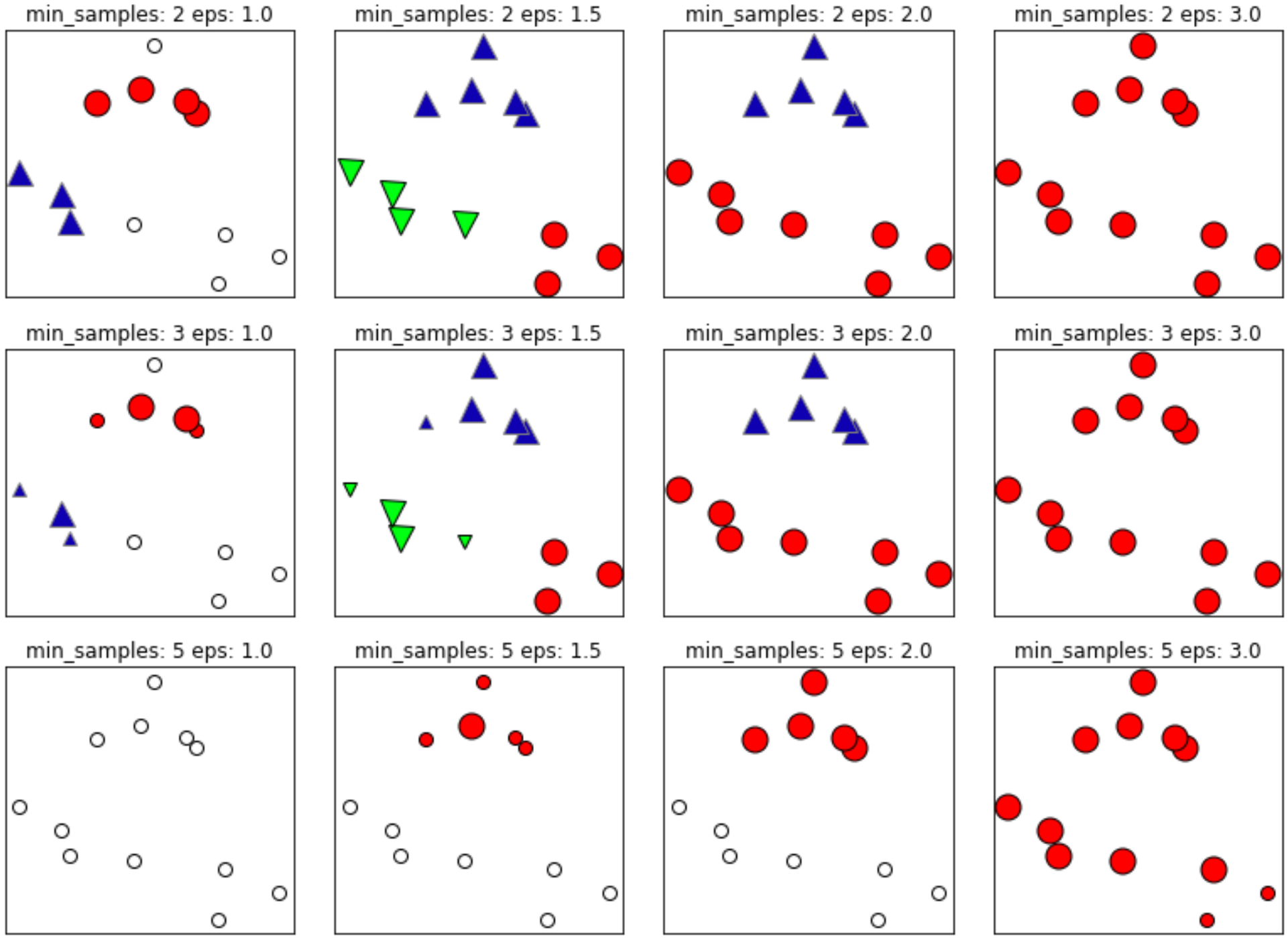
- 특성 공간에서 데이터가 붐비는 지역의 포인트를 찾음(특성 공간의 밀집 지역(dense region))
- 데이터의 밀집 지역이 한 클러스터를 구성하며 비교적 비어있는 지역을 경계로 다른 클러스터와 구분

DBSCAN

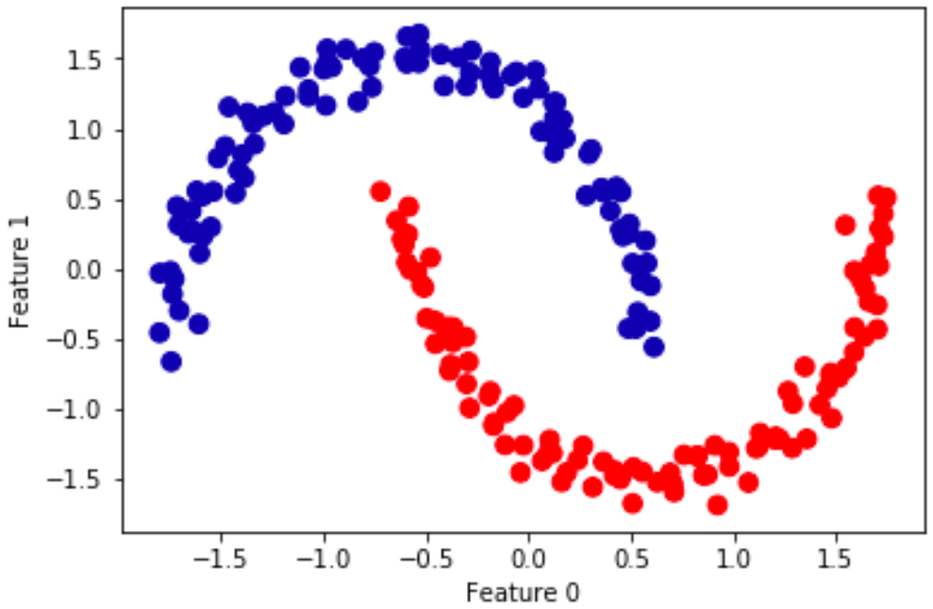
- min_samples

- eps
  - 한 데이터 포인트에서 eps 거리 안에 데이터가 min_samples 개수만큼 들어 있으면 이 데이터 포인트를 core sample(핵심 샘플)로 분류
  - eps 거리 안에 있는 포인트 수가 min_samples보다 적다면 그 포인트는 어떤 클래스에도 속하지 않는 잡음(noise)으로 분류

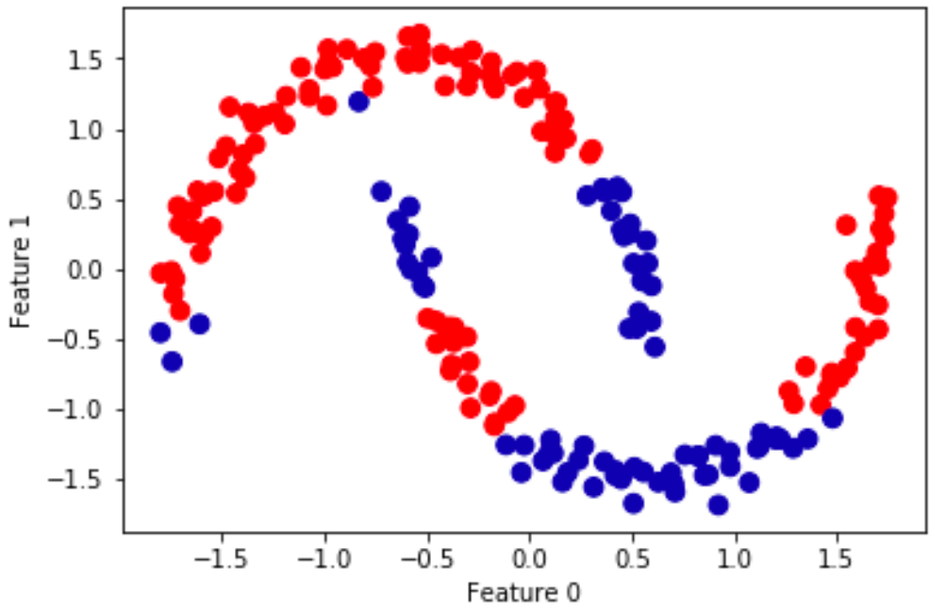
DBSCAN


min_samples: 2 eps: 1.0
min_samples: 2 eps: 1.5
min_samples: 2 eps: 2.0
min_samples: 2 eps: 3.0
min_samples: 3 eps: 1.0
min_samples: 3 eps: 1.5
min_samples: 3 eps: 2.0
min_samples: 3 eps: 3.0
min_samples: 5 eps: 1.0
min_samples: 5 eps: 1.5
min_samples: 5 eps: 2.0
min_samples: 5 eps: 3.0

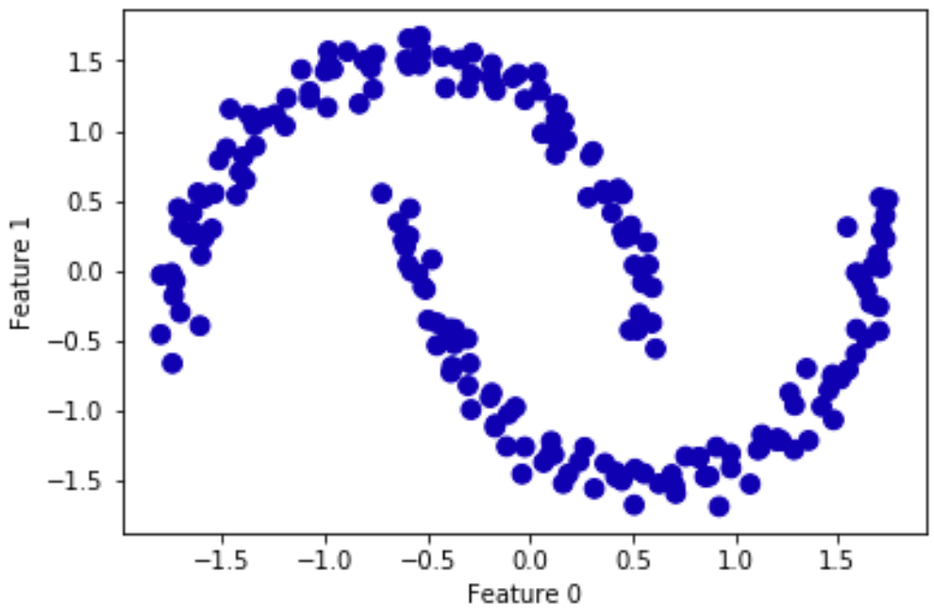
DBSCAN



eps=0.5(default)



eps=0.7



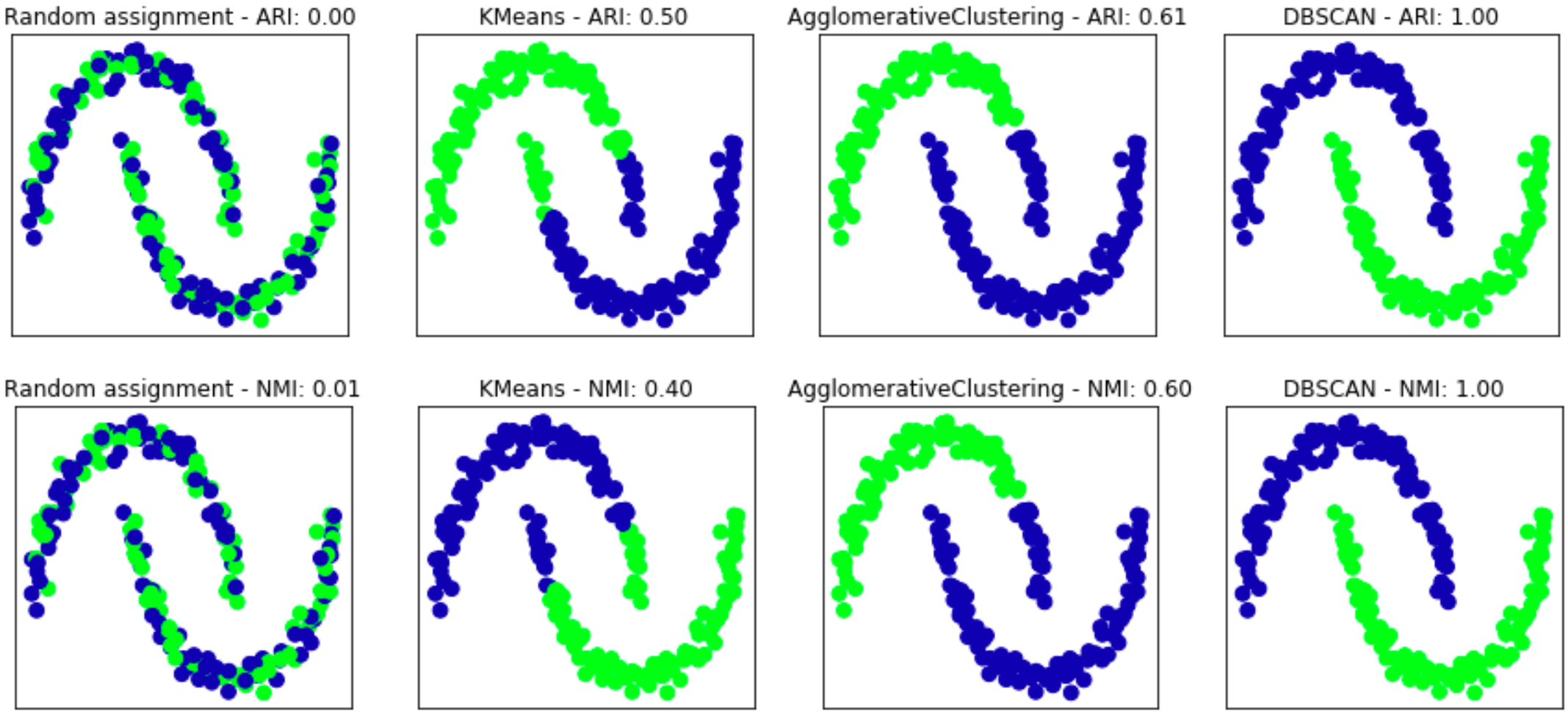
eps=0.2

DBSCAN

- Pros
  - cluster의 개수를 미리 지정할 필요가 없음
  - 복잡한 형상도 찾을 수 있으며, 어떤 클래스에도 속하지 않는 포인트를 구분할 수 있음
  - 비교적 큰 dataset에도 적용할 수 있음
- Cons
  - 분포가 고르지 않은(밀도의 차이가 큰) 데이터의 경우 clustering이 어려움
  - k-means나 agglomerative clustering보다는 다소 느림

Comparing & Evaluating

- ARI(adjusted rand index)
- NMI(normalized mutual information)



accuracy_score를 쓰면 안된다!

HYUNDAI MOBIS

Comparing & Evaluating

- silhouette coefficient
  - cluster의 밀집 정도를 계산
  - 높을 수록 좋음(max=1)