군집분석

- 권수태 교수

JEONJU UNIVERSITY
The Place for Superstars

1. 군집화

❖K-평균 알고리즘의 이해

- 군집 중심점이라는 특정한 임의 지점을 선택해 해당 중심에 가장 가까운 포인트들을 선택하는 군집화 기법
- 군집화에서 가장 일반적으로 사용되는 알고리즘

1) 군집화의 기준이 되는 중심을 구성하려는 군집화 개수만큼 임의의 위치에 가져다 놓음

2) 각 데이터는 가장 가까운 곳에 위치한 중심점에 소속

3) 소속이 결정되면 군집 중심점을 소속된 데이터의 평균 중심으로 이동

4) 중심점이 이동했기 때문에 각 데이터는 기존에 속한 중심점보다 더 가까운 중심점이 있다면 해당 중심점으로 다시 소속 변경

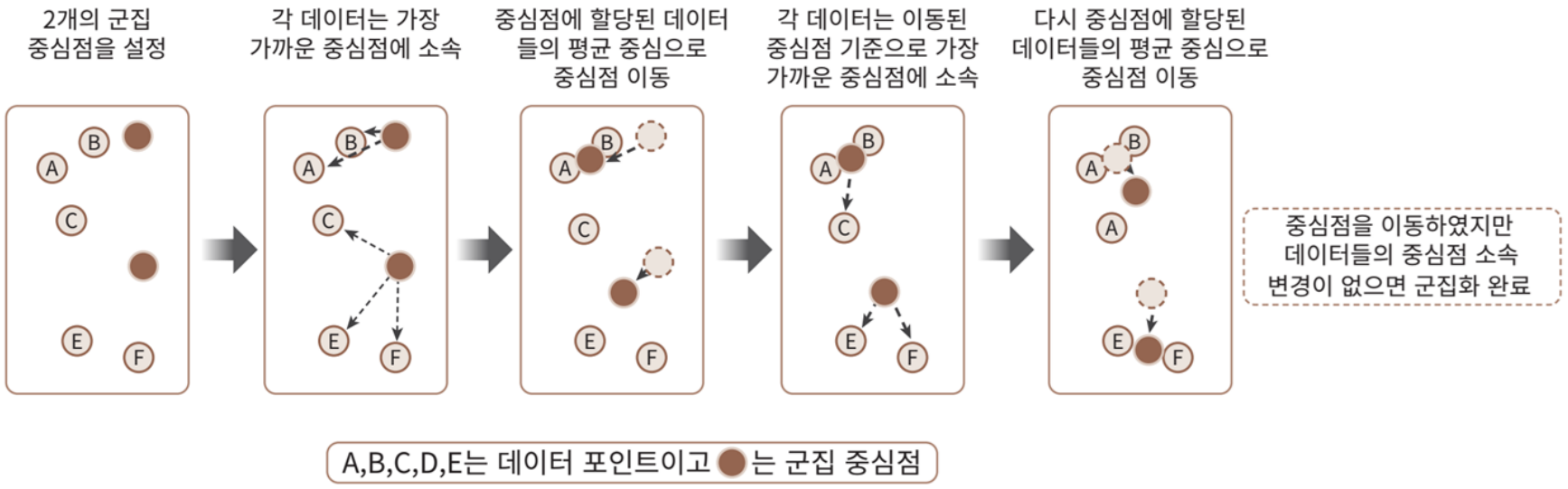
5) 다시 중심을 소속된 데이터의 평균 중심으로 이동

6) 중심점을 이동했는데 데이터들의 중심점 소속 변경이 없으면 군집화 종료

1. 군집화

❖K-평균 알고리즘의 이해

- ➢ 장점
  - ✓ 알고리즘이 쉽고 간결
- ➢ 단점
  - ✓ 거리 기반 알고리즘으로 속성의 개수가 매우 많을 경우 군집화 정확도가 떨어짐 (※ PCA로 차원 감소 적용하기도 함)
  - ✓ 반복을 수행하는데, 반복 횟수가 많을 경우 수행 시간이 매우 느려짐
  - ✓ 몇 개의 군집을 선택해야 할지 가이드가 어려움



JEONJU UNIVERSITY

1. 군집화

❖K-평균 알고리즘의 이해

➢ 사이킷런 KMeans 클래스 소개

- ✓ n_clusters: 군집 중심점의 개수
- ✓ max_iter: 최대 반복 횟수
- ✓ init: 초기에 군집 중심점의 좌표를 설정할 방식
- ✓ 주요 속성 정보
  - labels_ : 각 데이터 포인트가 속한 군집 중심점 레이블
  - cluster_centers_ : 각 군집 중심점 좌표

```
kmeans = KMeans(n_clusters=3, init='k-means++', max_iter=300,random_state=0)
kmeans.fit(irisDF)
```


1. 군집화

❖K-평균 알고리즘의 이해

➢ 군집화 알고리즘 테스트를 위한 데이터 생성

✓ 대표적인 군집화용 데이터 생성기 : make_bolbs()와 make_classification() API

- make_bolbs(): 개별 군집의 중심점과 표준편차 제어기능이 추가
- make_classification(): 노이즈를 포함한 데이터를 만드는데 유용

✓ Make_bolbs()를 호출하면 피처 데이터 세트와 타깃 데이터 세트가 튜플로 반환됨

- n_samples:생성할 총 데이터의 개수 (디폴트는 100)
- n_features: 데이터의 피처 개수 (시각화를 목표를 할 경우 2개로 설정함)
- centers: int 값, 군집의 개수
- cluster_std: 생성될 군집 데이터의 표준 편차

X, y = make_blobs(n_samples=200, n_features=2, centers=3, cluster_std=0.8, random_state=0)

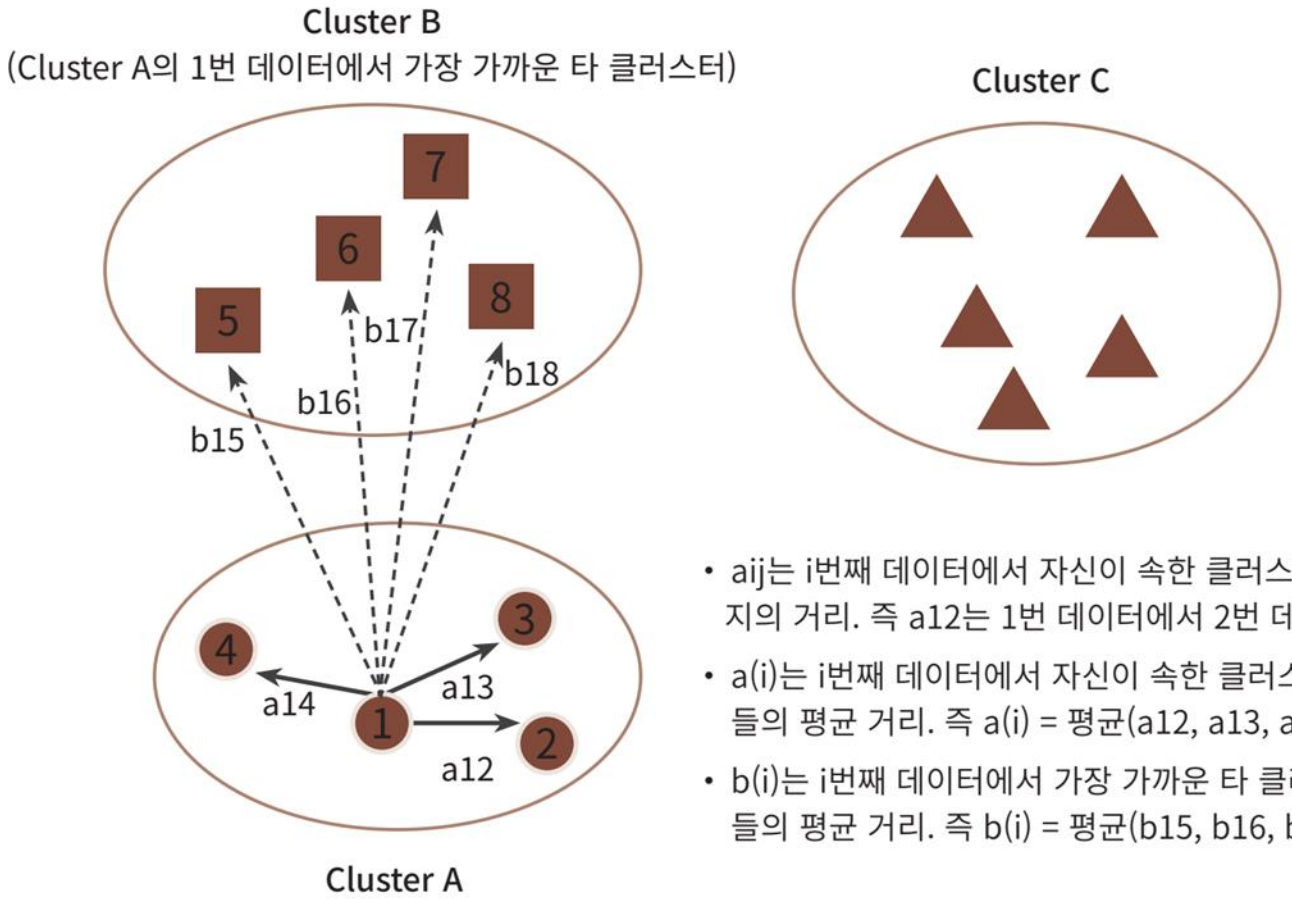
1. 군집화

❖군집평가

- 대부분의 군집화 데이터 세트는 비교할 만한 타깃 레이블을 갖고 있지 않음
- 군집화는 분류와 유사해 보일 수 있으나 성격이 많이 다름
- 데이터 내에 숨어있는 별도의 그룹을 찾아서 의미를 부여하거나 동일한 분류 값에 속하더라도 그 안에서 더 세분화된 군집화를 추구하거나 서로 다른 분류 값의 데이터도 더 넓은 군집화 레벨화 등의 영역을 갖고 있음
- 군집화의 성능을 평가하는 대표적인 방법으로는 실루엣 분석
- 실루엣 분석
  - ✓ 각 군집 간의 거리가 얼마나 효율적으로 분리돼 있는지 나타냄
  - ✓ 효율적으로 잘 분리됨 = 다른 군집과의 거리는 떨어져 있고 동일 군집끼리의 데이터는 서로 가깝게 잘 뭉쳐짐
  - ✓ 실루엣 계수 : 개별 데이터가 가지는 군집화 지표
  - ✓ 해당 데이터가 같은 군집 내의 데이터와 얼마나 가깝게 군집화돼 있고, 다른 군집에 있는 데이터와는 얼마나 멀리 분리돼 있는지를 나타내는 지표

1. 군집화

❖군집평가



- aij는 i번째 데이터에서 자신이 속한 클러스터내의 다른 데이터 포인트까지의 거리. 즉 a12는 1번 데이터에서 2번 데이터까지의 거리
- a(i)는 i번째 데이터에서 자신이 속한 클러스터내의 다른 데이터 포인트들의 평균 거리. 즉 a(i) = 평균(a12, a13, a13)
- b(i)는 i번째 데이터에서 가장 가까운 타 클러스내의 다른 데이터 포인트들의 평균 거리. 즉 b(i) = 평균(b15, b16, b17, b18)

1. 군집화

❖군집평가

➢실루엣 계수

✓i번째 데이터 포인트의 실루엣 계수 값 s(i)

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

✓Cluster A 내의 데이터 포인트들 끼리 평균 거리 : a(i)

✓Cluster B 내의 데이터 포인트들 끼리 평균 거리 : b(i)

✓실루엣 계수는 -1에서 1 사이의 값을 가짐

- 1로 가까워질수록 근처의 군집과 더 멀리 떨어져 있다는 것
- 0에 가까울수록 근처의 군집과 가까워진다는 것
- -값은 아예 다른 군집

1. 군집화

❖군집평가

➢ 사이킷런의 메서드

- ✓ silhouette_samples : 각 데이터 포인트의 실루엣 계수 반환
- ✓ silhouette_score : 전체 데이터의 실루엣 계수 값을 평균해 반환 (=np.mean(silhouette_samples())) 보통 값이 높으면 군집화가 어느정도 잘 됐다고 판단 가능하지만 무조건은 아님

➢ 좋은 군집화가 되기 위한 조건

1) 전체 실루엣 계수의 평균값은 0~1 사이의 값을 가지며 , 1에 가까울수록 좋음

2) 개별 군집의 평균값의 편차가 크지 않아야 함. 즉, 개별군집의 실루엣 계수 평균값이 전체 실루엣 계수의 평균값에서 크게 벗어나지 않는 것이 중요

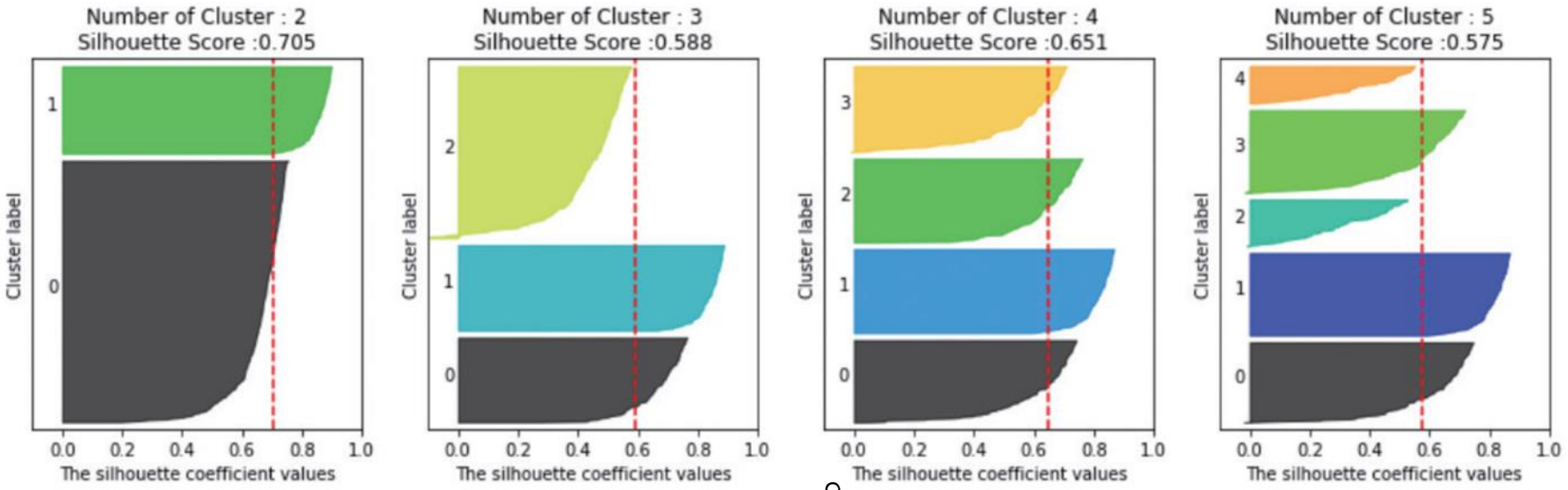
1. 군집화

❖군집평가

➢군집별 평균 실루엣 계수의 시각화 통한 군집 개수 최적화 방법

✓전체 데이터의 평균 실루엣 계수값이 높다고 해서 반드시 최적의 군집 개수로 군집화가 잘 되었다고 볼 수 없음

✓개별 군집별로 적당히 분리된 거리를 유지하면서도 군집 내의 데이터가 서로 뭉쳐있는 경우에 k-평균의 적절한 군집 개수가 설정됐다고 판단 가능

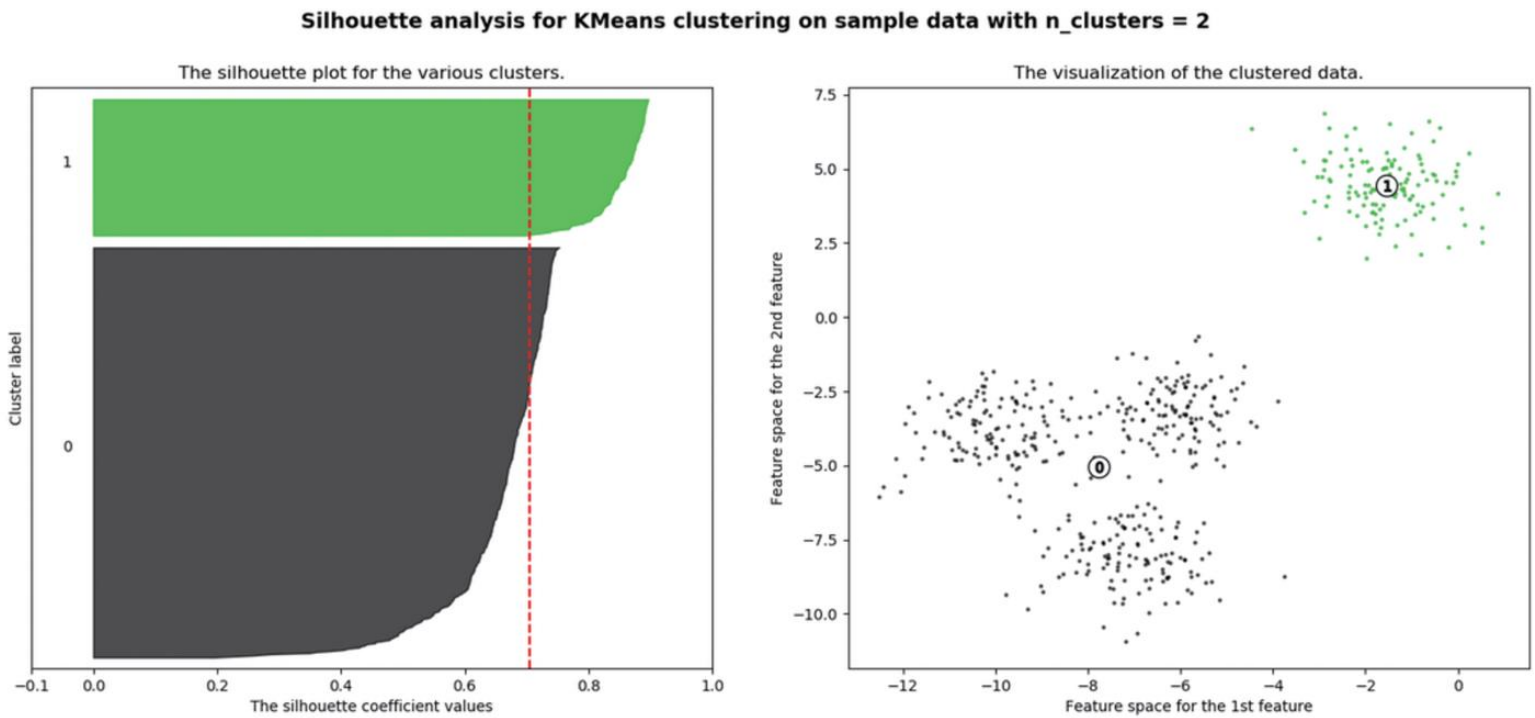


JEONJU UNIVERSITY

1. 군집화

❖군집평가

➢군집별 평균 실루엣 계수의 시각화 통한 군집 개수 최적화 방법



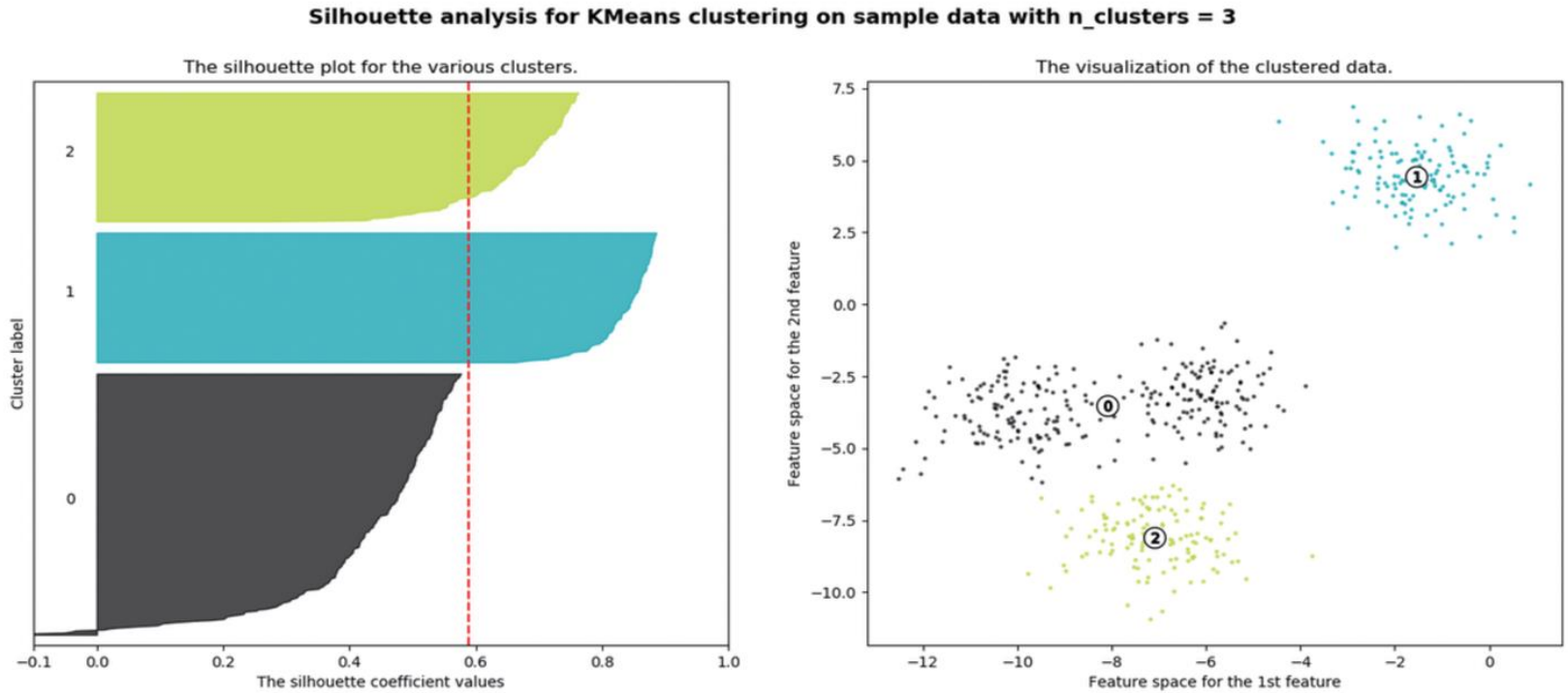
군집이 2개일 경우 평균 실루엣 계수 값: 0.704

JEONJU UNIVERSITY

1. 군집화

❖군집평가

➢군집별 평균 실루엣 계수의 시각화 통한 군집 개수 최적화 방법



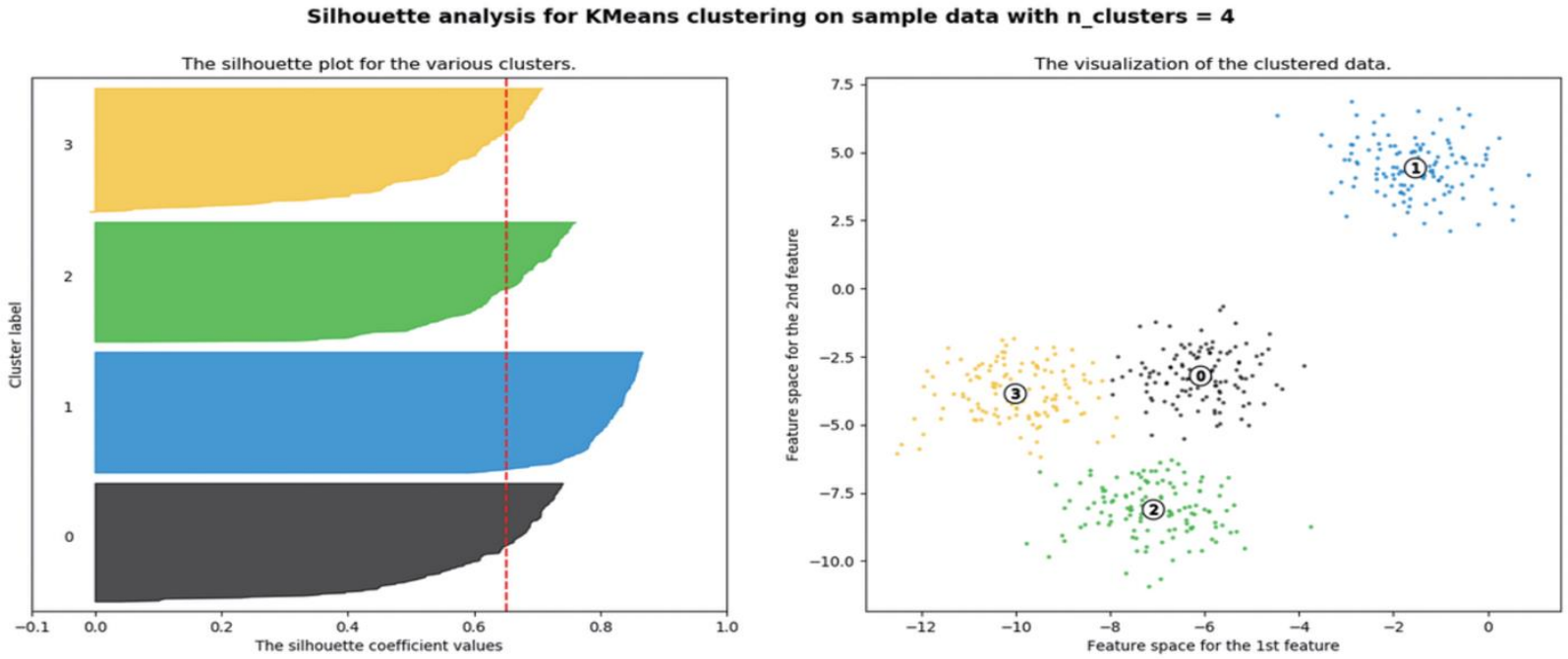
군집이 3개일 경우 평균 실루엣 계수 값: 0.588

JEONJU UNIVERSITY

1. 군집화

❖군집평가

➢군집별 평균 실루엣 계수의 시각화 통한 군집 개수 최적화 방법



군집이 4개일 경우 평균 실루엣 계수 값: 0.65

JEONJU UNIVERSITY

1. 군집화

❖평균이동

➢ Mean Shift 개요

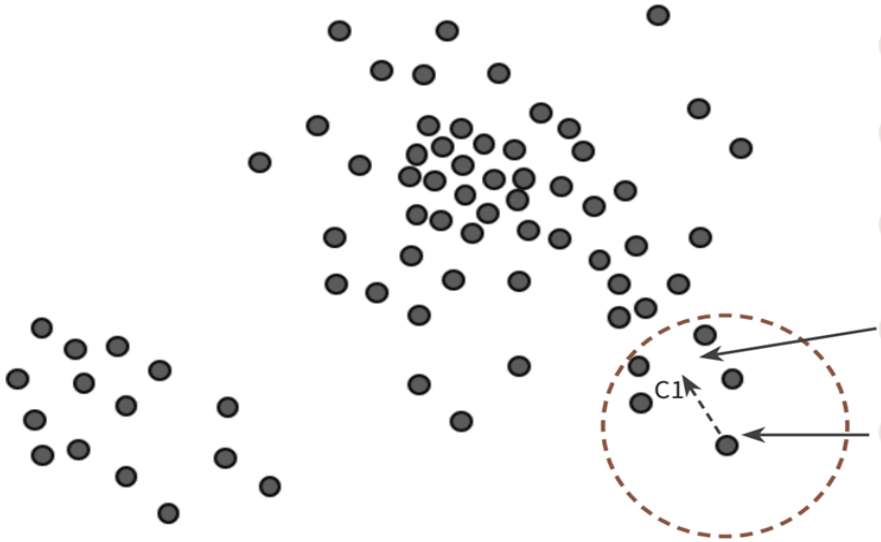
- ✓ k-평균과 유사하게 중심을 군집의 중심으로 지속적으로 움직이면서 군집화를 수행
- ✓ k-평균이 중심에 소속된 데이터의 평균 거리 중심으로 이동하는데 반해, 평균 이동은 중심을 데이터가 모여 있는 밀도가 가장 높은 곳으로 이동 시킴
- ✓ 데이터의 분포도를 이용해 군집 중심점을 찾음
- ✓ 별도의 군집화 개수를 정하지는 않음
- ✓ 군집 중심점 : 데이터 포인트가 모여있는 곳이라는 생각에서 착안한 것 / 확률 밀도 함수 이용
- ✓ 일반적으로 주어진 모델의 확률 밀도 함수를 찾기 위해 KDE이용

1. 군집화

❖평균이동

➢ Mean Shift 개요

✓ 특정 데이터를 반경 내의 데이터 분포 확률 밀도가 가장 높은 곳으로 이동하기 위해 주변 데이터와의 거리 값을 KDE 함수 값으로 입력한 뒤 그 반환 값을 현재 위치에서 업데이터 하면서 이동하는 방식



5 개별 데이터들이 모인 중심점을 군집 중심점으로 설정

4 지정된 반복(Iteration) 횟수만큼 전체 데이터에 대해서 KDE 기반으로 데이터를 이동시키면서 군집화 수행

3 모든 데이터를 1~2까지 수행하면서 데이터를 이동. 개별 데이터들이 군집중심점으로 모임.

2 KDE로 계산된 데이터 분포도가 높은 방향으로 데이터 이동

1 개별 데이터의 특정 반경 내에 주변 데이터를 포함한 데이터 분포도를 KDE 기반의 Mean Shift 알고리즘으로 계산

JEONJU UNIVERSITY

1. 군집화

❖평균이동

➢ KDE(Kernel Density Estimation)

✓ 커널 함수를 통해 어떤 변수의 확률 밀도 함수를 측정하는 대표적인 방법

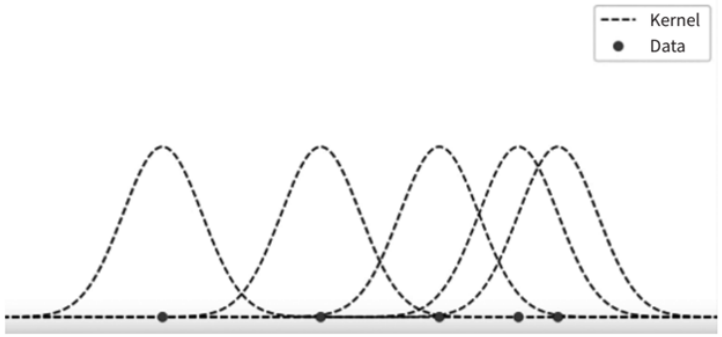
✓ 확률 밀도 함수를 알면 특정 변수가 어떤 값을 갖게 될지에 대한 확률을 알게 되므로 이를 통해 변수의 특성, 확률 분포 등 많은 요소를 알 수 있음

※확률 밀도 함수 특정방법 :모수적 or 비모수적

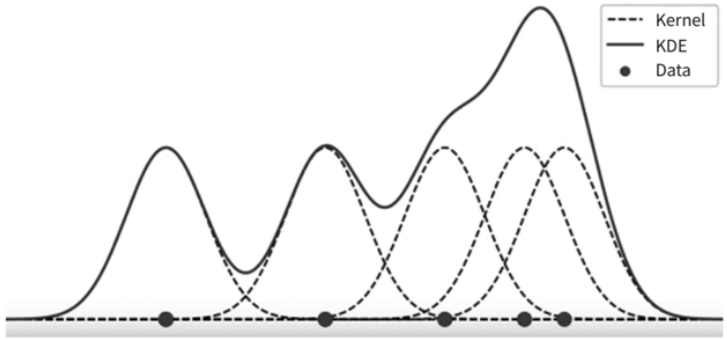
✓ 커널 함수를 적용한 뒤 , 이 적용값을 모두 더한 후 개별 관측 데이터의 건수로 나눠 확률 밀도 함수 추정

==> 대표적인 커널 함수 : 가우시안 분포 함수

개별 관측 데이터에 가우시안 커널 함수 적용



가우시안 커널 함수 적용 후 합산



1. 군집화

❖평균이동

➢ KDE(Kernel Density Estimation)

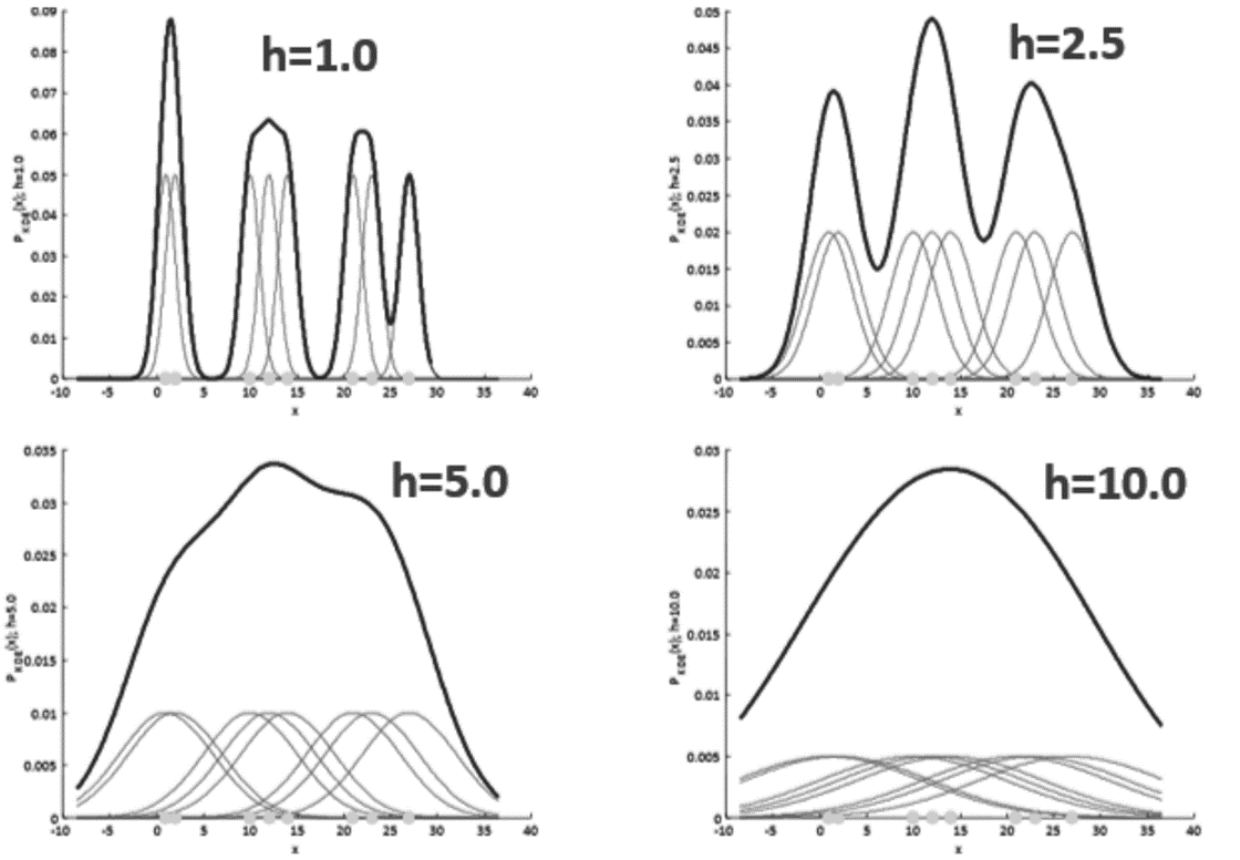
$$\mathrm{KDE} = \frac{1}{n}\sum_{i=1}^{n} K_h(x - x_i) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right)$$

- ✓ K :커널 함수, x : 확률 변숫값, xi : 관측값, h : 대역폭
- ✓ 대역폭 h는 KDE 형태를 부드러운 형태로 평활화하는데 적용됨
- ✓ h가 너무 작으면 과적합되기 쉬우며(많은 수의 군집 중심점을 가짐), h가 너무 크면 과소 적합 되기 쉬움 (적은 수의 군집 중심점을 가짐)
- ✓ 또한, 평균이동 군집화는 군집의 개수를 지정하지 않으며, 오직 대역폭의 크기에 따라 군집화를 수행

1. 군집화

❖평균이동

➢ KDE(Kernel Density Estimation)



JEONJU UNIVERSITY

1. 군집화

❖평균이동

- 사이킷런 MeanShift 클래스 제공
- 최적의 대역폭 계산을 위해 estimate_bandwith() 제공

- 평균이동의 장점은 데이터 세트의 형태를 특정형태로 가정한다든가, 특정 분포도 기반의 모델로 가정하지 않기 때문에 좀 더 유연한 군집화가 가능
- 또한, 이상치의 영향력도 크지 않으며, 미리 군집의 개수를 정할 필요도 없음
- 평균 이동은 알고리즘 수행 시간이 오래 걸리고 bandwidth의 크기에 따른 군집화 영향도가 매우 크다
- 이런 특징 때문에 이미지나 영상 데이터에서 특정 개체를 구분하거나 움직임을 추적하는데 뛰어난 역할을 수행하는 알고리즘

1. 군집화

❖GMM(Gaussian Mixture Model)

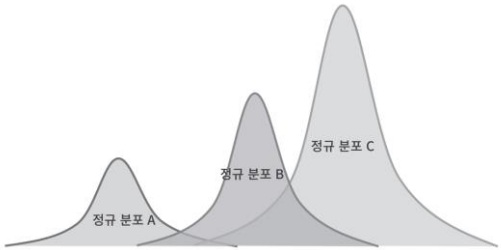
➢ GMM(Gaussian Mixture Model) 소개

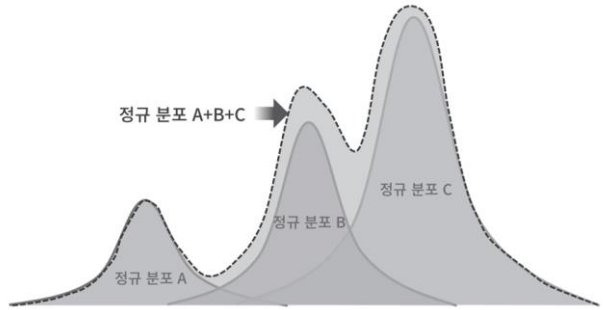
- ✓ 군집화를 적용하고자 하는 데이터가 여러 개의 가우시안 분포를 가진 데이터 집합들이 섞여서 생성된 것이라는 가정 하에 군집화를 수행하는 방식
- ✓ 섞인 데이터 분포에서 개별 유형의 가우시안 분포 추출
- ✓ 개별 데이터가 이 중 어떤 정규분포에 속하는지 찾고 데이터가 특정 정규 분포에 해당될 확률 구함
- ✓ 이를 모수 추정이라 함

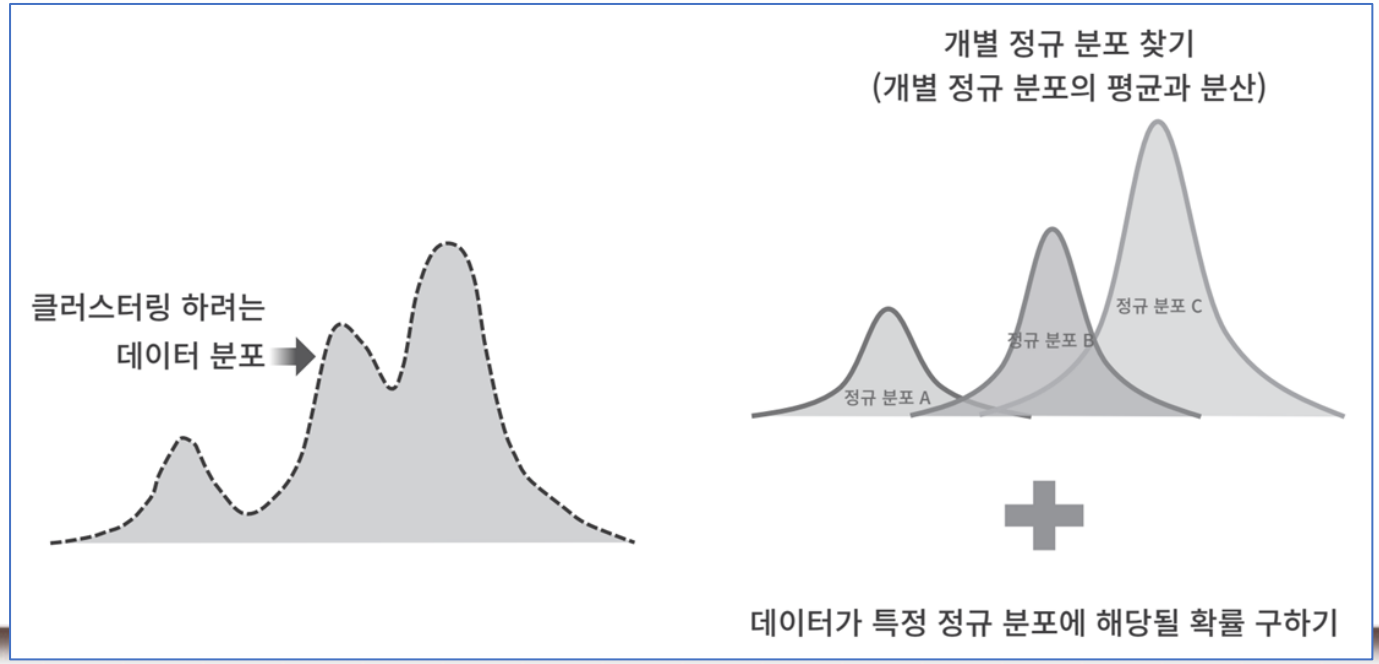
1. 군집화

❖GMM(Gaussian Mixture Model)

➢ GMM(Gaussian Mixture Model) 소개







JEONJU UNIVERSITY

1. 군집화

❖GMM(Gaussian Mixture Model)

➢GMM(Gaussian Mixture Model) 소개

✓ 모수 추정은 대표적으로 2가지를 추정

1) 개별 정규 분포의 평균과 분산

2) 각 데이터가 어떤 정규 분포에 해당되는지의 확률

✓ 모수 추정 위해 GMM은 EM 방법 사용

✓ EM(Expectation and Maximization) : 개별 정규 분포의 모수인 평균과 분산이 더이상 변경되지 않고 각 개별 데이터들이 이전 정규 분포 소속이 더 이상 변경되지 않으면 그것을 최종 군집화로 결정(변경되면 계속 EM)

✓ 사이킷런 GaussianMixture 클래스 지원

1. 군집화

❖GMM(Gaussian Mixture Model)

➢GMM과 K-평균의 비교

- ✓ KMeans 는 원형의 범위에서 군집화를 수행
- ✓ 데이터세트가 원형의 범위를 가질수록 Kmeans 의 군집화 효율은 더욱 높아짐

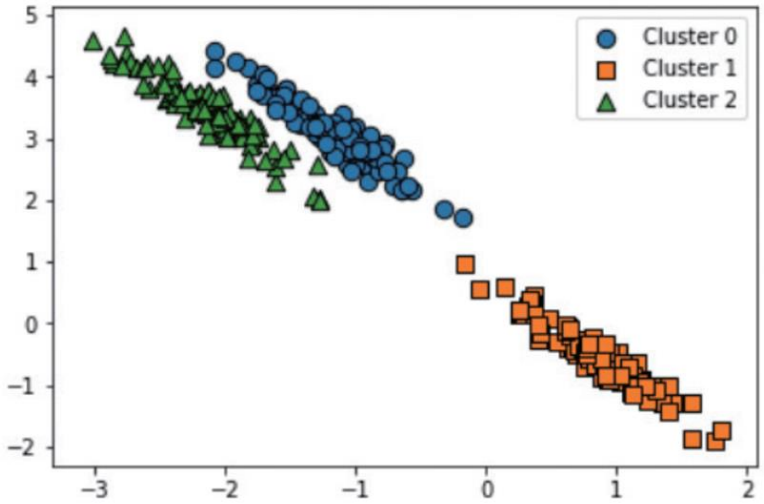
➢군집 시각화 함수 visualize_cluster_plot()

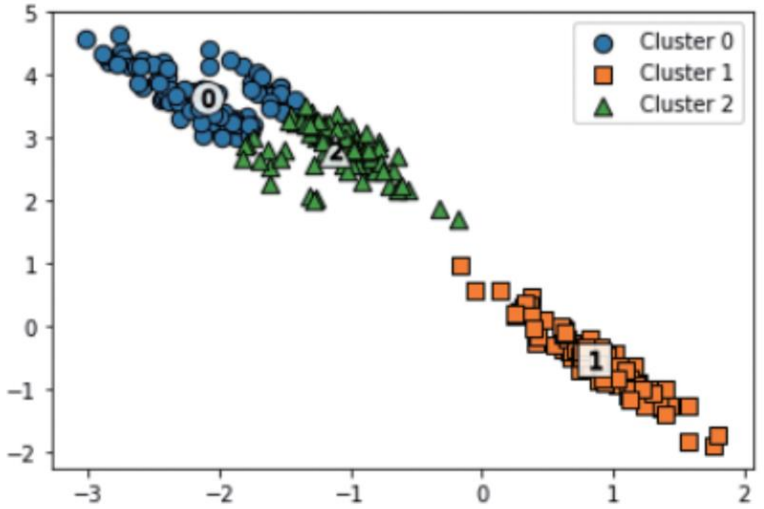
- ✓ clusterobj: 사이킷런의 군집 수행 객체. KMeans 나 GaussianMixture 의 fit()와 predict()로 군집화를 완료한 객체
- ✓ dataframe: 피처 데이터세트와 label 값을 가진 DataFrame
- ✓ label_name: dataframe 내의 군집화 label 칼럼명
- ✓ iscenter: 객체가 군집 중심좌표를 제공하면 true

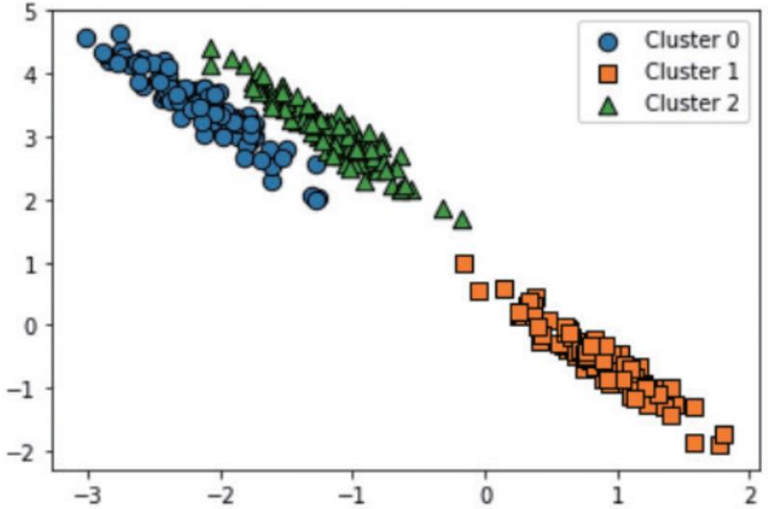
1. 군집화

❖GMM(Gaussian Mixture Model)

➢GMM과 K-평균의 비교



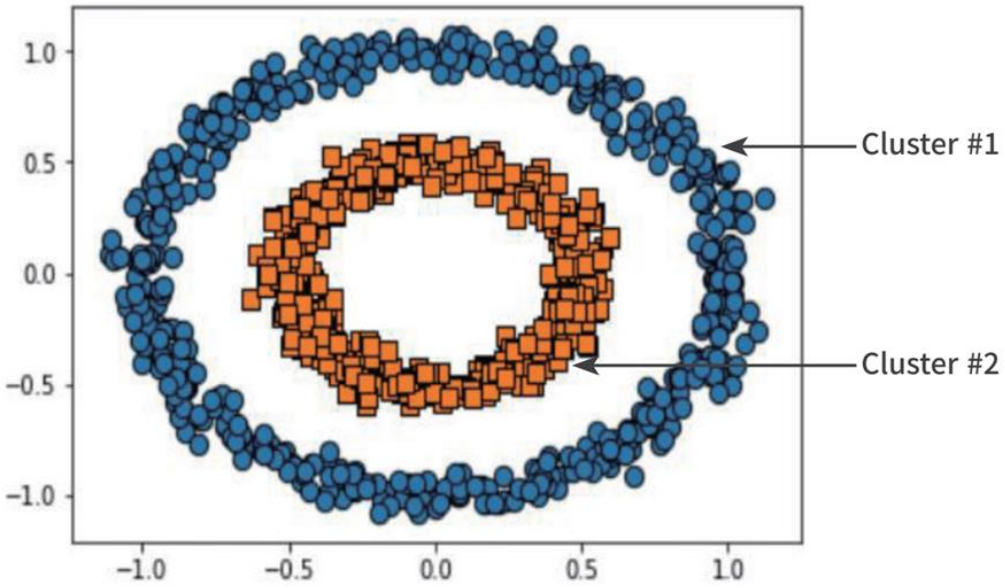




1. 군집화

❖DBSCAN (Density Based Spatial Clustering of Applications with Noise)

- 특정 공간 내에 데이터 밀도 차이를 기반 알고리즘으로 하고 있어서 복잡한 기하학적 분포도를 가진 데이터 세트에 대해서도 군집화를 잘 수행함
- 단점: 데이터 밀도가 자주 변하거나 아예 변화하지 않는 것, 피처 개수가 많은 데이터인 경우에는 군집화 성능이 떨어짐



JEONJU UNIVERSITY

1. 군집화

❖ DBSCAN (Density Based Spatial Clustering of Applications with Noise)

➢ 가장 중요한 두가지 파라미터

✓ 입실론 주변 영역 : 개별 데이터를 중심으로 입실론 반경을 가지는 원형의 영역

✓ 최소 데이터 개수 : 개별 데이터의 입실론 주변 영역에 포함되는 타 데이터의 개수

✓ 사이킷런 DBSCAN 클래스의 주요 파라미터 (eps, min_samples)

➢ 주변 영역 내에 포함되는 최소 데이터 개수를 총족시키는가 아닌가에 따른 데이터 포인트 정의

1) 핵심 포인트 : 주변 영역 내에 최소 데이터 개수 이상의 타 데이터를 가지고 있을 경우

2) 이웃 포인트 : 주변 영역 내에 위치한 타 데이터

3) 경계 포인트 : 주변 영역 내에 최소 데이터 개수 이상의 이웃 포인트를 가지고 있지 않지만 핵심 포인트를 이웃 포인트로 가지고 있는 데이터/ 군집의 외곽 형성

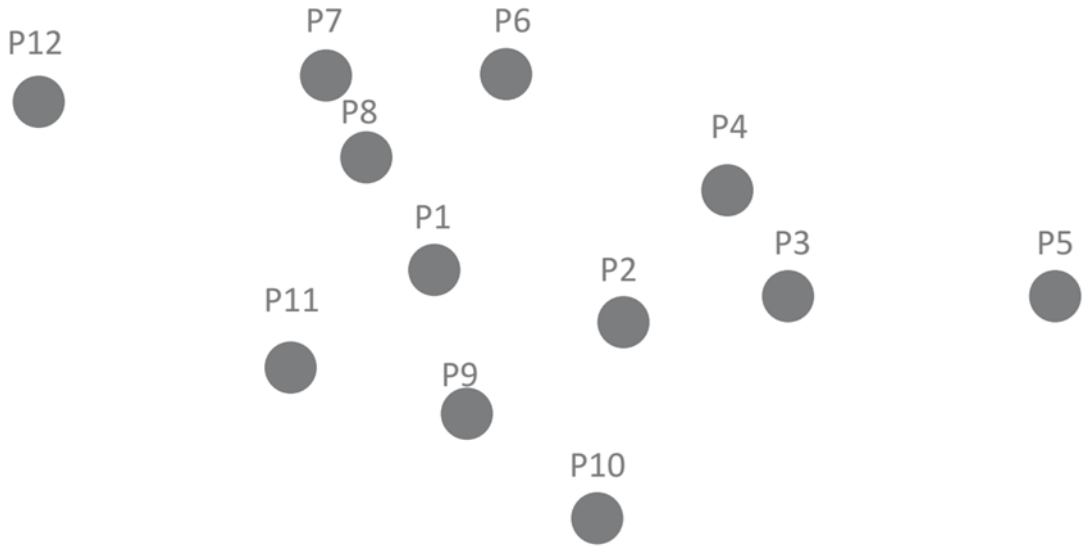
4) 잡음 포인트 : 최소 데이터 개수 이상의 이웃 포인트를 가지고 있지 않으며, 핵심 포인트도 이웃 포인트로 가지고 있지 않는 데이터

1. 군집화

❖DBSCAN (Density Based Spatial Clustering of Applications with Noise)

➢DBSCAN 군집화 적용

✓최소 데이터세트 6개 가정

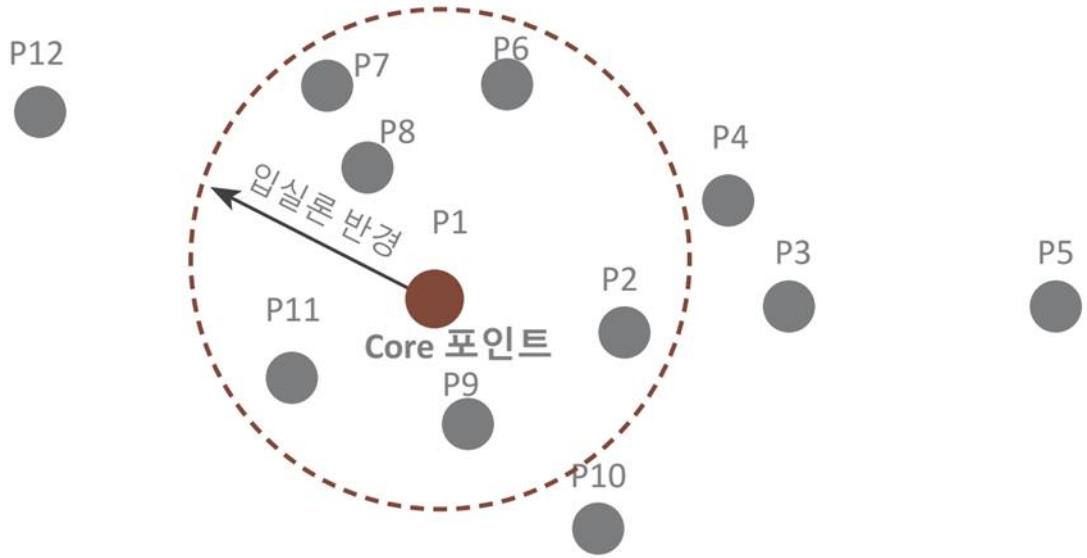


1. 군집화

❖DBSCAN (Density Based Spatial Clustering of Applications with Noise)

➢DBSCAN 군집화 적용

✓P1 데이터를 기준으로 입실론 반경 내에 포함된 데이터가 7개(자신은 P1, 이웃데이터 P2, P6, P7, P8, P9, P11)로 최소 데이터 5개 이상을 만족하므로 P1 데이터는 핵심 포인트



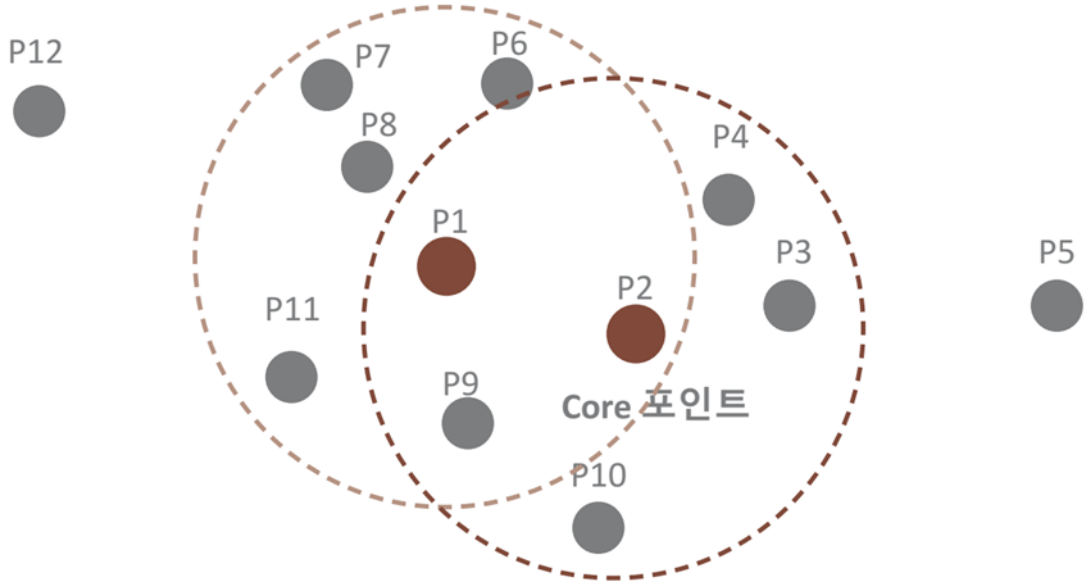
JEONJU UNIVERSITY

1. 군집화

❖DBSCAN (Density Based Spatial Clustering of Applications with Noise)

➢DBSCAN 군집화 적용

✓P2 데이터 기준 (최소 데이터 6개로 핵심 포인트)

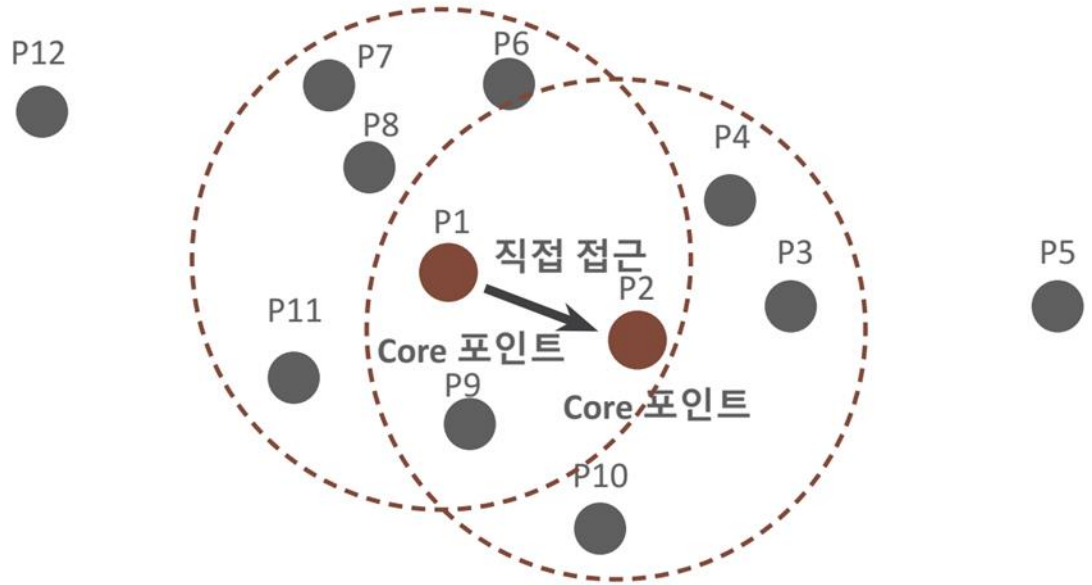


1. 군집화

❖DBSCAN (Density Based Spatial Clustering of Applications with Noise)

➢DBSCAN 군집화 적용

✓핵심포인트 P1의 이웃 데이터 포인트 P2 역시 핵심 포인트일 경우 P1에서 P2 로 연결해 직접 접근이 가능



JEONJU UNIVERSITY

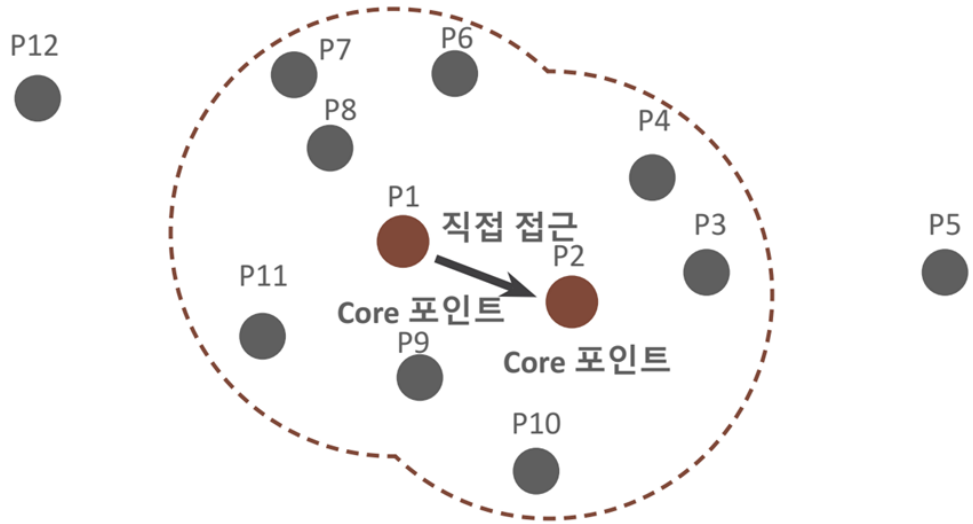
1. 군집화

❖DBSCAN (Density Based Spatial Clustering of Applications with Noise)

➢DBSCAN 군집화 적용

✓특정 핵심포인트에서 직접 접근이 가능한 다른 핵심 포인트를 서로 연결하면서 군집화를 구성

✓이러한 방식으로 점차적으로 군집 영역을 확장해 나가는 것이 DBSCAN 군집화 방식



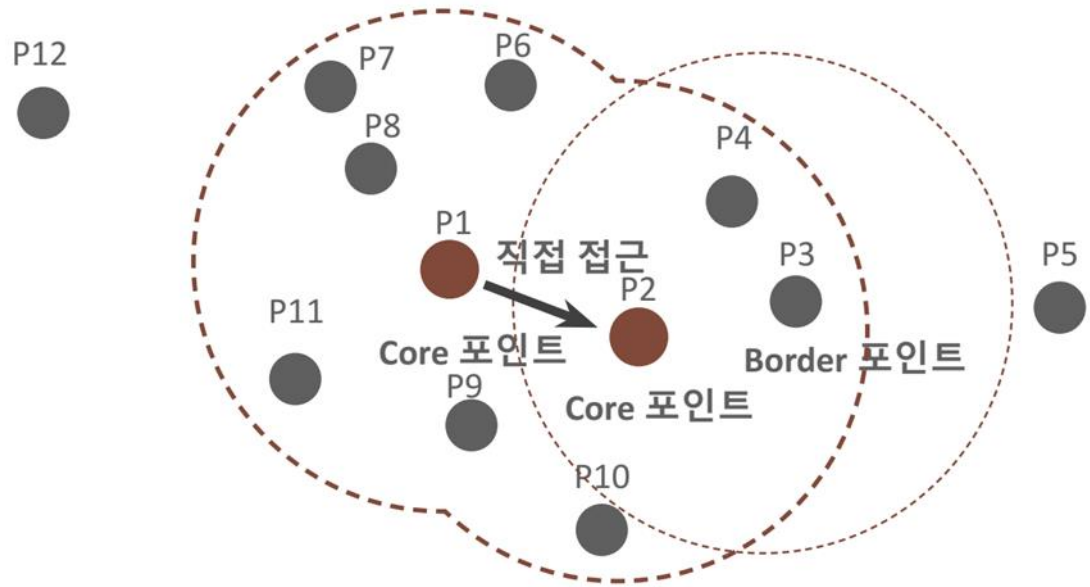
JEONJU UNIVERSITY

1. 군집화

❖DBSCAN (Density Based Spatial Clustering of Applications with Noise)

➢DBSCAN 군집화 적용

✓P3는 반경내에 P2, p4로 2개 이므로 핵심포인트가 아님. 이웃데이터 중에 핵심포인트 P2 를 가지고 있어 경계 포인트(군집의 외곽을 형성)

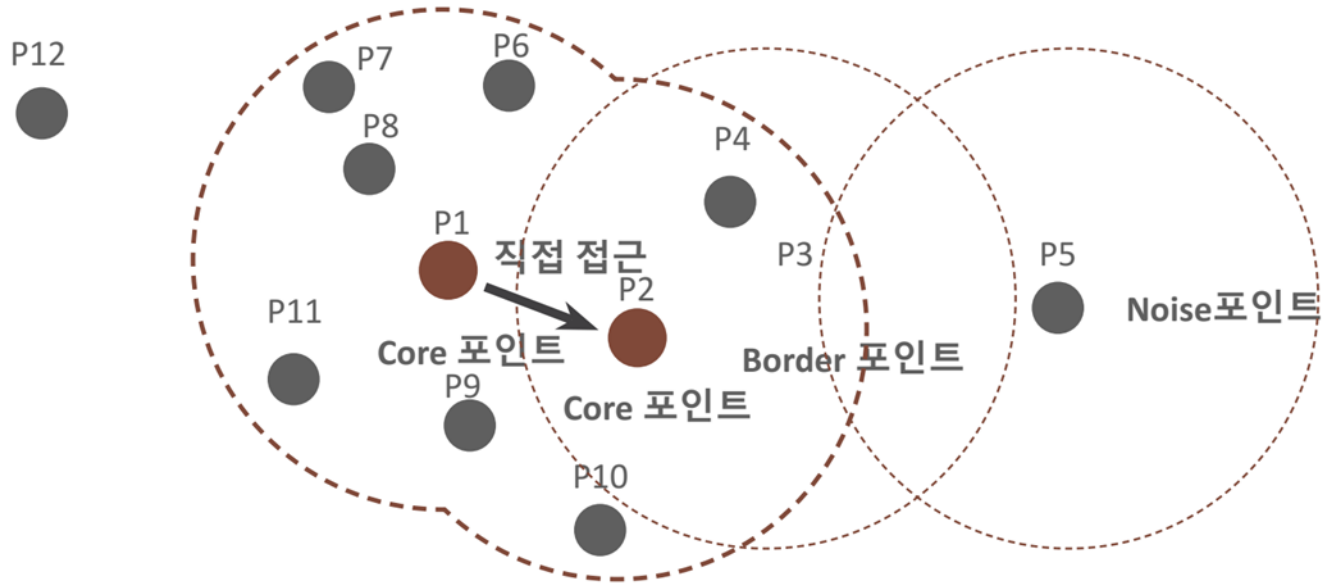


1. 군집화

❖DBSCAN (Density Based Spatial Clustering of Applications with Noise)

➢DBSCAN 군집화 적용

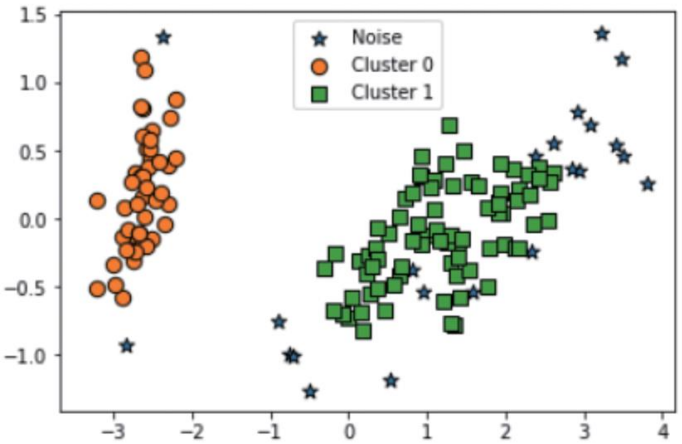
✓P5는 잡음포인트

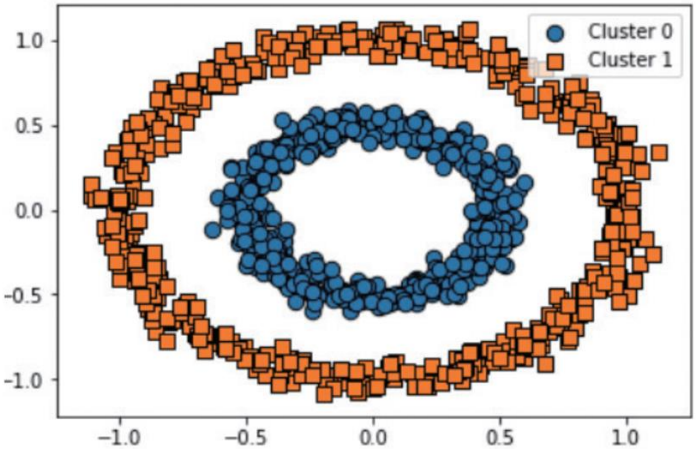


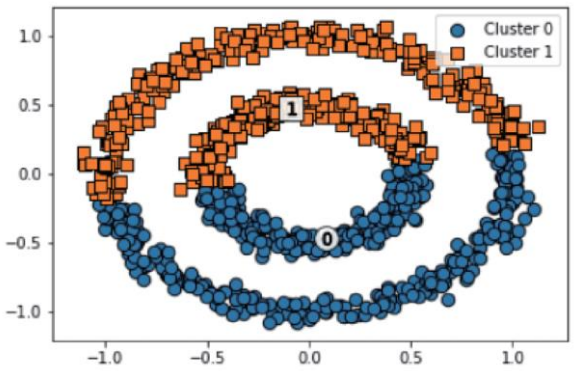
1. 군집화

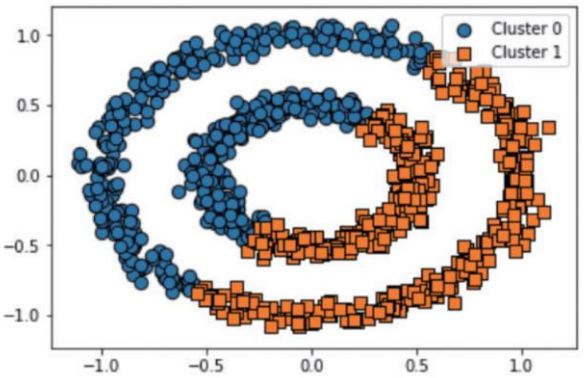
❖DBSCAN (Density Based Spatial Clustering of Applications with Noise)

➢DBSCAN 군집화 적용









JEONJU UNIVERSITY

Thank You !

JEONJU UNIVERSITY
The Place for Superstars