텍 스 트 세 미 나

ToBig's 10기 임진혁

클러스터링

Contents

Unit 01 | 비지도학습-클러스터링(군집화)

Unit 02 | 계층적 군집화

Unit 03 | K-Means

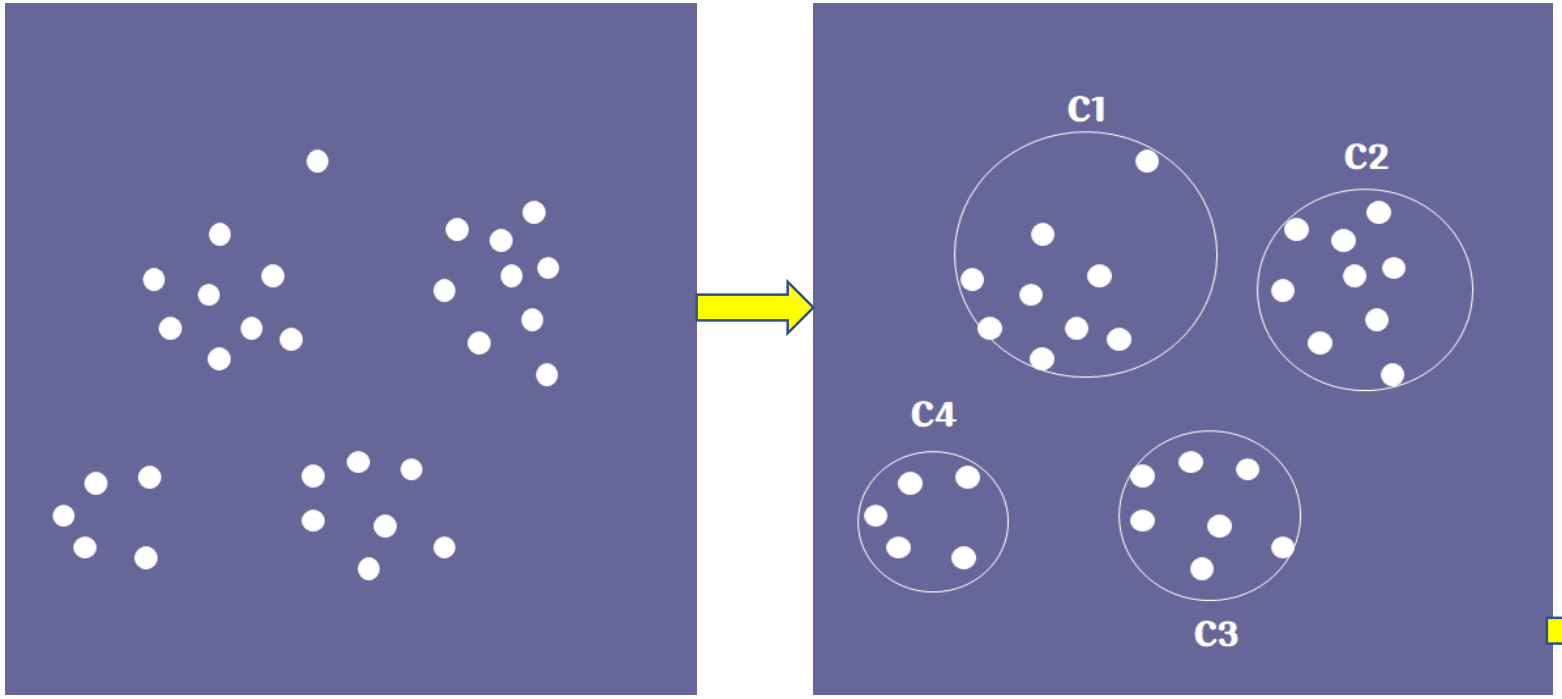
| 그 외 보충 알고리즘

Unit 04 | 모델 평가.

Unit 01 | 비지도학습-클러스터링

클러스터링

유사성 등의 개념에 기초하여 데이터를 몇몇의 그룹으로 분류하는 수법의 총칭. 문헌 검색, 패턴 인식, 경영 과학 등에 폭넓게 응용되고 있다.



클러스터(cluster)는 영어로 '**군집, 모임**' 을 의미합니다.

즉 하나의 **데이터들의 무리**를 말하는거죠.

그렇다면
클러스팅으로 데이터 군집을
왜 만들어야 할까?
클러스터링은 왜 하는건가

데이터에 라벨(LABEL)이 없기 때문에
비지도학습(클러스터링)으로 접근.

Unit 01 | 비지도학습-클러스터링

지도학습(Supervised Learning)

- 분류: 소속집단의 정보를 이미 알고 있는 상태에서, 비슷한 집단으로 묶는 방법
  즉, Label이 있는 Data를 나누는 방법으로, Supervised Learning의 일종

비지도학습(Unsupervised Learning)

- 군집화: 소속집단의 정보가 없고, 모르는 상태에서 비슷한 집단으로 묶는 방법
  즉, Label이 없는 Data를 군집단위로 나누는 것으로, Unsupervised Learning 의 일종

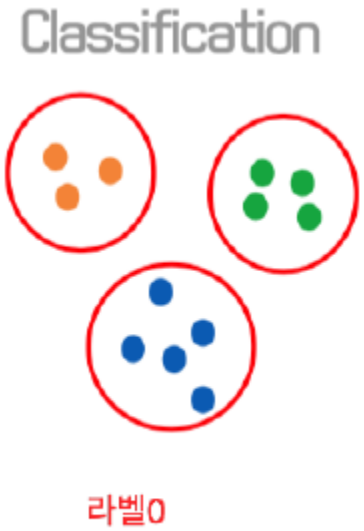
Unit 01 | 비지도학습-클러스터링

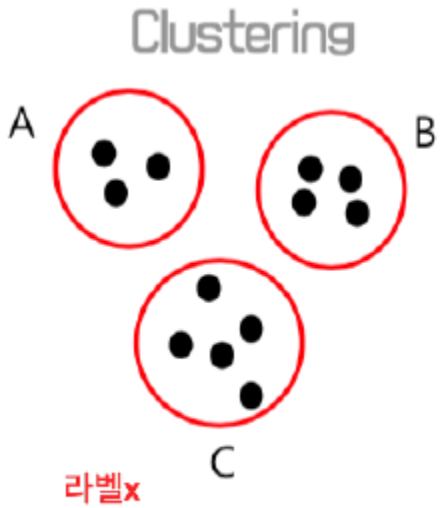
쉽게 말해서
라벨이 있는 데이터들은 편하다.
찾고 싶은 책이 있으면 그 책에 해당하는 라벨에서 찾으면 된다. 즉, 그 책과 비슷한 특징을 가진 책들이 있는 라벨을 찾으면 된다.

하지만 라벨이 없다면? 새로운 책(데이터)를 어디에 배치해야할지 알 수 없다. 그럴 때는, 라벨 정보없이 책들의 특성으로 비슷한 책들끼리 그룹화한 다음에 새로운 책을 가장 비슷한 특성의 그룹에 배치하면 될 것이다.

Unit 01 | 비지도학습-클러스터링

- 즉, 비지도 학습은 라벨링 없는 데이터를 가지고 지도학습과 비슷한 일을 하고자 하는 기법이다. 군집은 데이터의 특성을 가지고 임의대로 만든 라벨링이라고 생각 할 수 있다.





지도학습은 이미 색깔(라벨링)이 있는 데이터들을 가지고 선을 확실하게 그을 수 있다. 새로운 점을 넣을 때 노란 점들이 들어 있는 테두리 안에 있다면 그 새로운 점은 초록점이라고 예측-분류 할 수 있다.

비지도 학습은 색깔(라벨)이 없는 점들을 갖고 각각의 점들의 특성(데이터의 패턴)을 파악하여 라벨 정보가 아닌 데이터 특성들을 이용하여 군집을 만들고 테두리를 만든다.

새로운 점이 왔을 때, B 테두리 안에 들어간다면 군집 B로 분류하고 그 안의 데이터들의 패턴과 가장 비슷하다고(유사) 생각한다.

Unit 01 | 비지도학습-클러스터링

세 가지 일반 과업

- 군집화clustering: 유사한 샘플을 모아 같은 그룹으로 묶는 일
- 밀도 추정density (probability mass function) estimation: 데이터로부터 확률분포를 추정하는 일
- 공간 변환space transformation: 원래 특징 공간을 저차원 또는 고차원 공간으로 변환하는 일

데이터에 내재한 구조를 잘 파악하여 새로운 정보를 발견해야 함

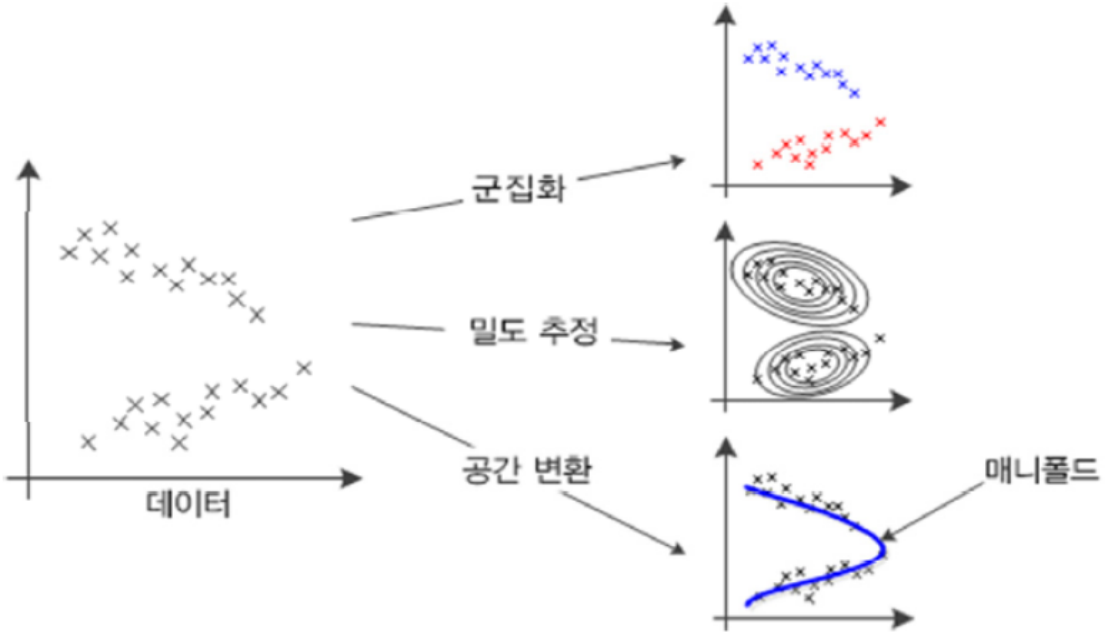


그림 6-2 비지도 학습의 군집화, 밀도 추정, 공간 변환 과업이 발견하는 정보

Unit 01 | 비지도학습-클러스터링

- 군집분석의 예시.

-고객들을 구매태도,구매성향 등의 특성-패턴을 분석하여 비슷한 고객들끼리 군집화하고 각각의 군집에 대해서 다른 마케팅 전략을 적용. -상품에 대한 리뷰-텍스트 분석을 할 때, 비슷한 텍스트들끼리 군집분석을 하면

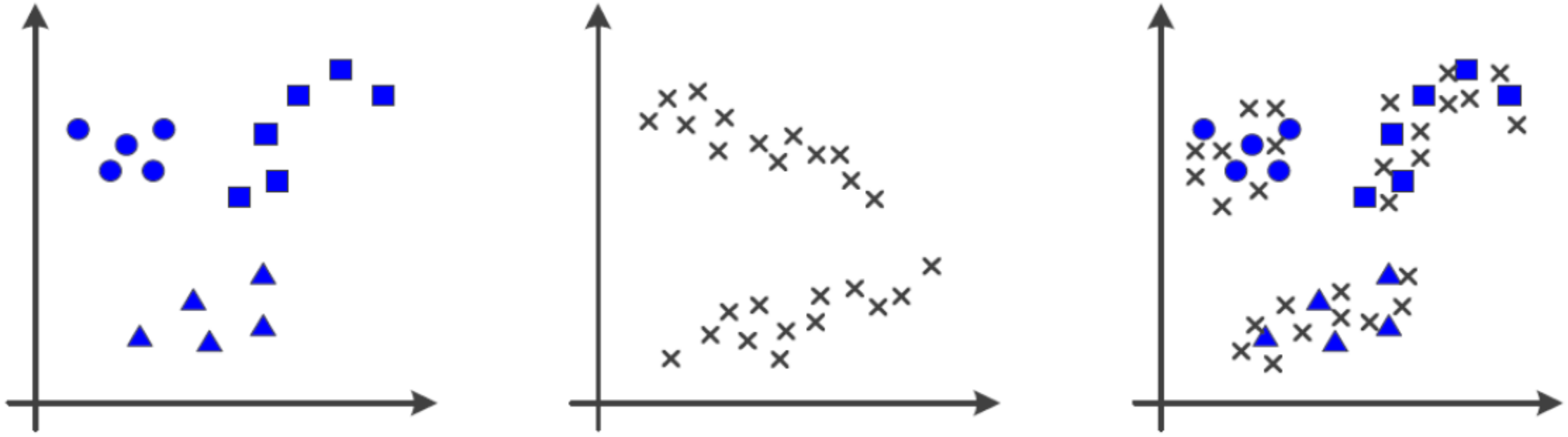
상품에 대한 의견을 쉽게 파악할 수 있다. (불만 30 %, 만족 70% 등)

-또한, 시각화를 할 때, 보다 쉽게 데이터 간의 유사함과 특성을 파악할 수 있다.

-준 지도학습 : 라벨링이 일부 되어있는 데이터에 (현실 데이터) 지도학습을 적용할 수 있다.

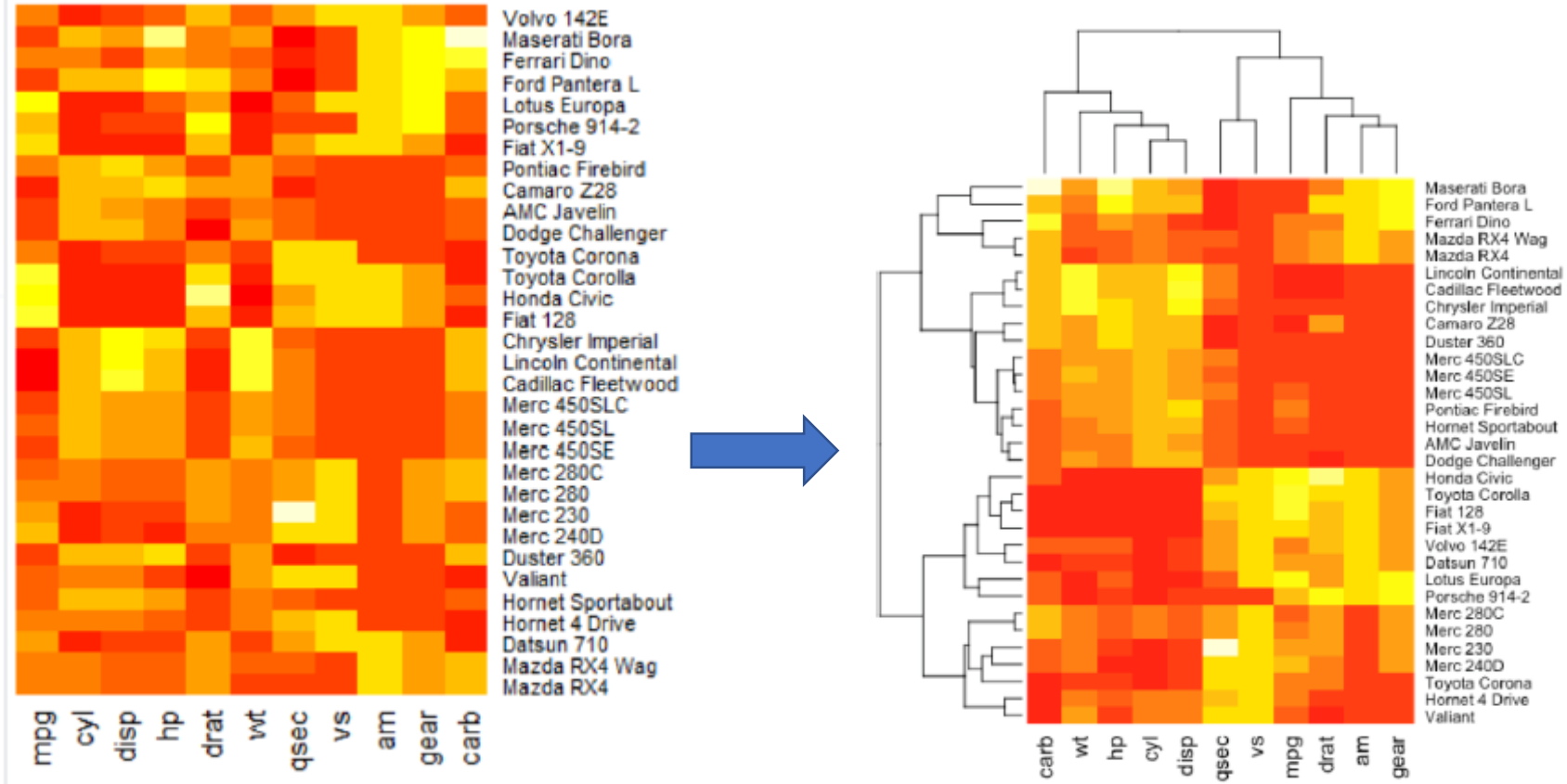
Unit 01 | 비지도학습-클러스터링

- 지도 학습supervised learning: 모든 훈련 샘플이 레이블 정보를 가짐
- 비지도 학습unsupervised learning: 모든 훈련 샘플이 레이블 정보를 가지지 않음 ←
- 준지도 학습semi-supervised learning: 레이블을 가진 샘플과 가지지 않은 샘플이 섞여 있음



(a) 지도 학습 (b) 비지도 학습 (c) 준지도 학습

Unit 01 | 비지도학습-클러스터링



데이터의 시각화에서도 군집화를 통해 보다 가독성 좋게 표현할 수 있다.

- [오늘 배울 내용]

- 군집화에는 어떤 기법들이 있는가?
- 군집으로 나누는 기준은?
- 군집화가 잘됐는지 확인할 수 있는가?

Unit 02 | 계층적 군집화

- 데이터의 패턴이 비슷하면 같은 군집(유사성), 데이터의 특성이 다를 수록 다른 군집(상이성)
- 즉, 유사하면 같은 군집으로 나눈다!
- 군집의 기준 → 유사도

군집분석 방법

1. Hierarchical agglomerative clustering(계층적 군집화):

   모든 데이터가 하나의 군집으로 병합될 때까지 군집들을 자연적인 계층 구조로 정렬하는 것
   ex) single linkage, complete linkage, average linkage, controid, Ward의 방법 등

2. Partitioning clustering(비계층적 군집화):

   구하고자 하는 군집의 수를 정한 상태에서 설정된 군집의 중심에 가장 가까운 개체를 하나씩 포함해 가는 방식
   ex) k-means, PAM(partitioning around medoids)

Unit 02 | 계층적 군집화

군집분석 단계

1. **알맞은 속성 선택** - 데이터를 군집화하는데 중요하다고 판단되는 속성들을 선택

2. **데이터 표준화** - 분석에 사용되는 변수들의 범위에 차이가 있는 경우 가장 큰 범위를 갖는 변수가 결과에 가장 큰 영향을 미치게 됨

3. **이상치 선별** - 많은 군집분석 방법은 이상치에 민감하기 때문에 군집 분석 결과가 왜곡됨

4. **군집 알고리즘 선택**

5. **군집의 개수 결정**

Unit 02 | 계층적 군집화

계층적 군집화는 "거리"를 유사도로 판단.
→거리가 가까울 수록 같은 특성을 가졌다 = 같은 군집이다.
→특성이 다를수록 거리가 먼 군집에 속해 있다.

주로 사용되는 거리의 정의

1 유클리드 거리 (Euclidean) : $d(x,y) = \left(\sum_{i=1}^{p}(x_i - y_i)^2\right)^{1/2}$

2 맨하탄 거리 (Manhattan) : $d(x,y) = \sum_{i=1}^{p}|x_i - y_i|$

3 표준화 거리 (Standardized) : $d(x,y) = \left(\sum_{i=1}^{p}(x_i - y_i)^2/s_i^2\right)^{1/2}$

4 민콥스키 거리 (Minkowski) : $d(x,y) = \left(\sum_{i=1}^{p}(x_i - y_i)^m\right)^{1/m}$

군집-군집 or 군집-개체 간 거리 측정 방법

- Single link: 두 군집에서 가장 가까운 멤버들의 거리를 잰다. 긴 체인(chain)을 만드는 경향이 있다.
- Complete link: 두 군집에서 가장 먼 멤버의 거리를 잰다. 구형(spherical)으로 뭉치는 경향이 있다.
- Average link: 모든 멤버들 사이의 거리를 평균낸다.
- Centroids: 군집의 중심과 중심의 거리를 잰다.
- Ward's method: 두 군집을 합쳤을 때 군집 내 거리 분산의 변화를 거리의 척도로 삼는다.

- 먼저, 최단 연결법으로 “거리 " 를 이용해서 군집화하는 과정을 보겠습니다.
  이 때, 거리는 “유클리드” 거리 정의를 사용.

데이터	(x1, x2)
A	(1, 5)
B	(2, 4)
C	(4, 6)
D	(4, 3)
E	(5, 3)

Dist(데이터)

유클리드 제곱거리	A	B	C	D	E
A	0				
B	2	0			
C	10	8	0		
D	13	5	9	0	
E	20	10	10	1	0

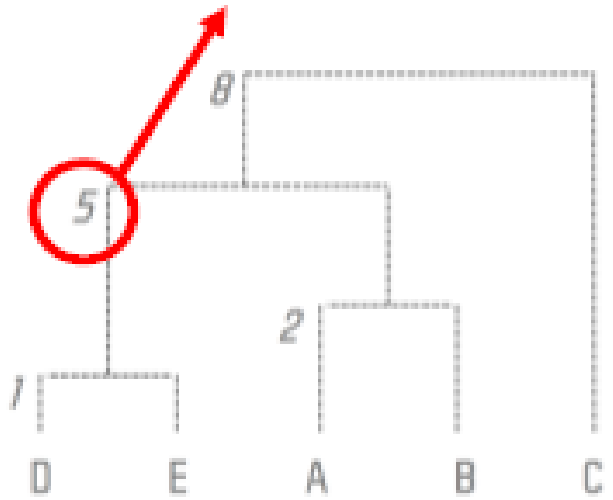
데이터	(x1, x2)
A	(1, 5)
B	(2, 4)
C	(4, 6)
D	(4, 3)
E	(5, 3)

Dist(데이터)

유클리드 제곱거리	A	B	C	(D, E)
A	0			
B	2	0		
C	10	8	0	
(D, E)	13	5	9	0

유클리드 제곱거리	(A, B)	C	(D, E)
(A, B)	0		
C	8	0	
(D, E)	5	9	0

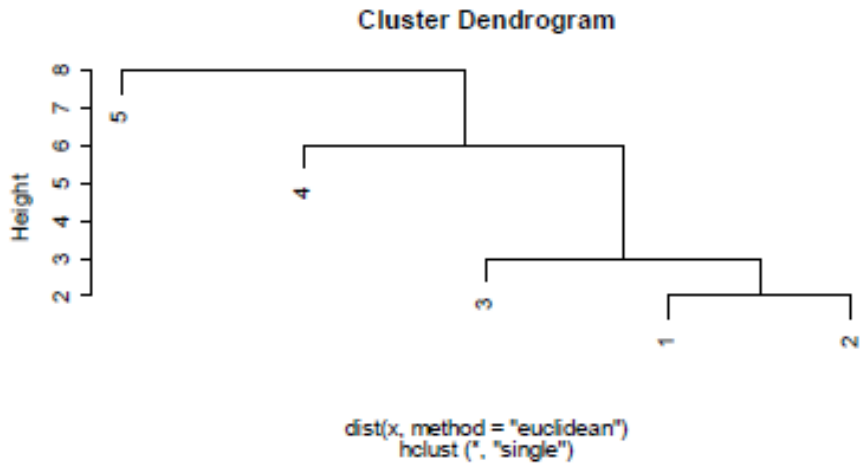
덴드로그램의 높이 = 관측치 간의 거리



계층적 군집분석: 최단연결법 (Single Linkage)

- $d_{C_iC_j} = \min\{d(x,y)|x \in C_i, y \in C_j\}$

```
> x=c(1,3,6,12,20)
> dist(x,method="euclidean")
   1  2  3  4
2  2
3  5  3
4 11  9  6
5 19 17 14  8
> hc1=hclust(dist(x,method="euclidean"),method="single") # single linkage
> plot(hc1)
```

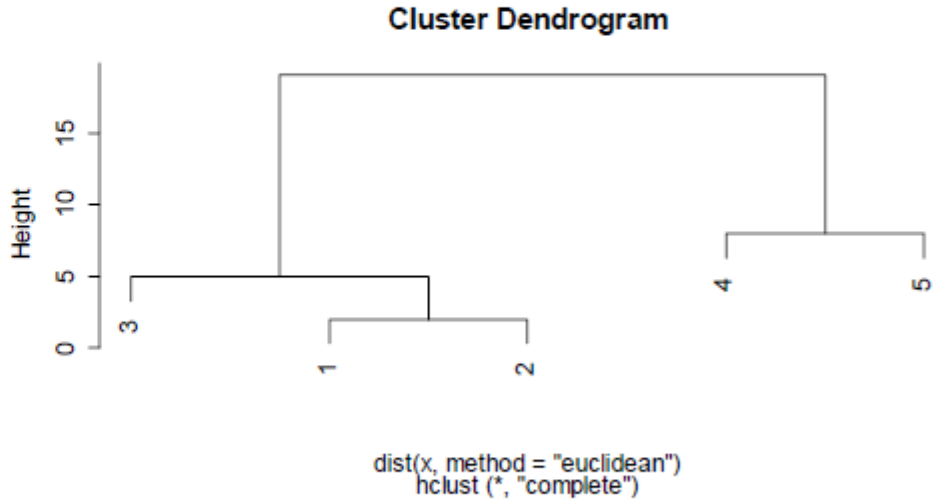


- 계산이 효율적
- 근시안적: 길다란 형태의 군집 형성 가능

계층적 군집분석: 최장연결법 (Complete Linkage)

- $d_{C_iC_j} = \max\{d(x,y)|x \in C_i, y \in$

```
> hc2=hclust(dist(x,method="euclidean"),method="complete") # complete linkage
> plot(hc2)
```

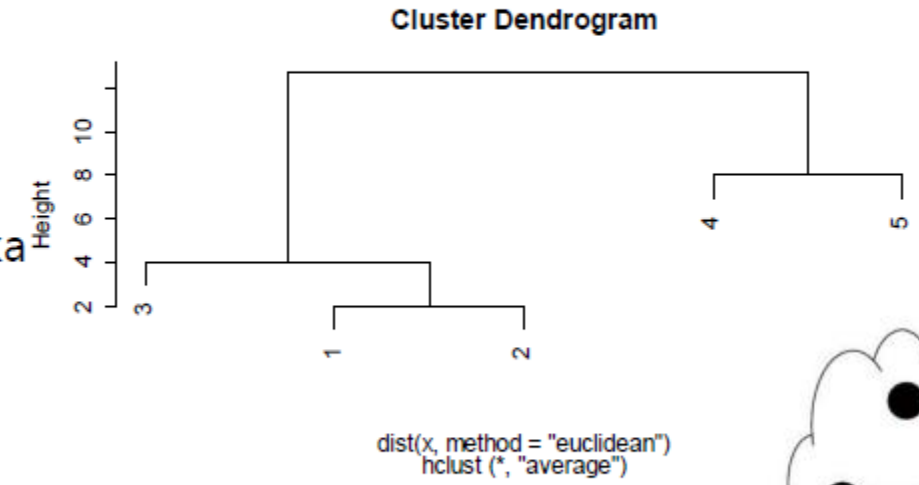


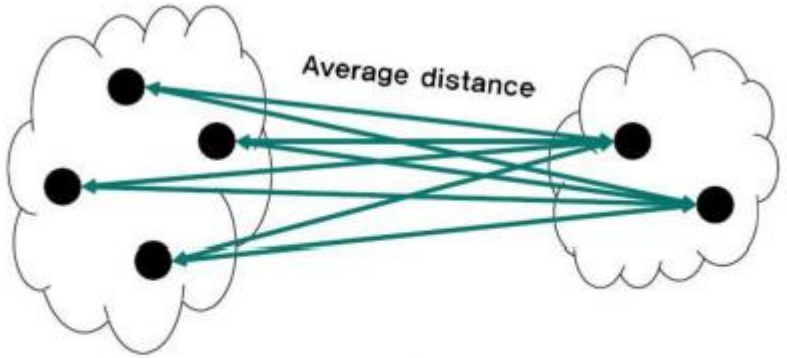
계층적 군집분석: 평균연결법 (Average Linkage)

- d

```
> hc3=hclust(dist(x,method="euclidean"),method="average") # average linkage
> plot(hc3)
```

- single linkage와 complete linka





$$d_{(UV)W} = \frac{\sum_{x_i \in (U,V)} \sum_{x_j \in W} d(x_i, x_j)}{n_{(UV)} n_W}$$

요약

1. 단일 데이터 간 거리를 정의하고

   - 맨하탄 거리, 유클리드 거리 등

2. 군집-군집 or 군집-개체 간 거리를 정의하고

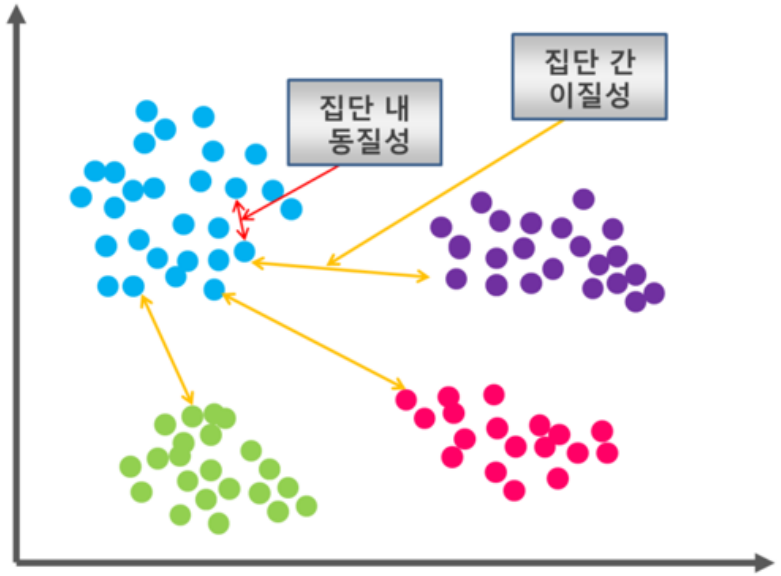
   - 최단 연결법, 평균 연결법, 최장 연결법 등

3. 돌리자!

[단점]

1. 일단 한 군집에 속하게 되면 그 데이터는 다른 군집으로 이동할 수 없다.
2. 자료의 개수 n이 많아지면 그룹의 개수 y에 따라 계산량이 많아진다.
3. 역시 자료의 개수가 많아지면 덴드로그램을 통한 해석이 어려워진다.
4. 최적의 군집 수를 알 수 없다. 사용자가 그 때마다 해석하기 나름

그렇다면 계층적 군집화에서 적절한 군집의 개수는 몇 개 일까?



군집 간 분산 최대화 !!

군집 내 분산 최소화 !!

기본적으로 군집화의 평가기준은

“

같은 군집에서의 데이터는 최대한 유사하고 (가깝고)

다른 군집끼리는 최대한 달라야 한다. (멀다)

”

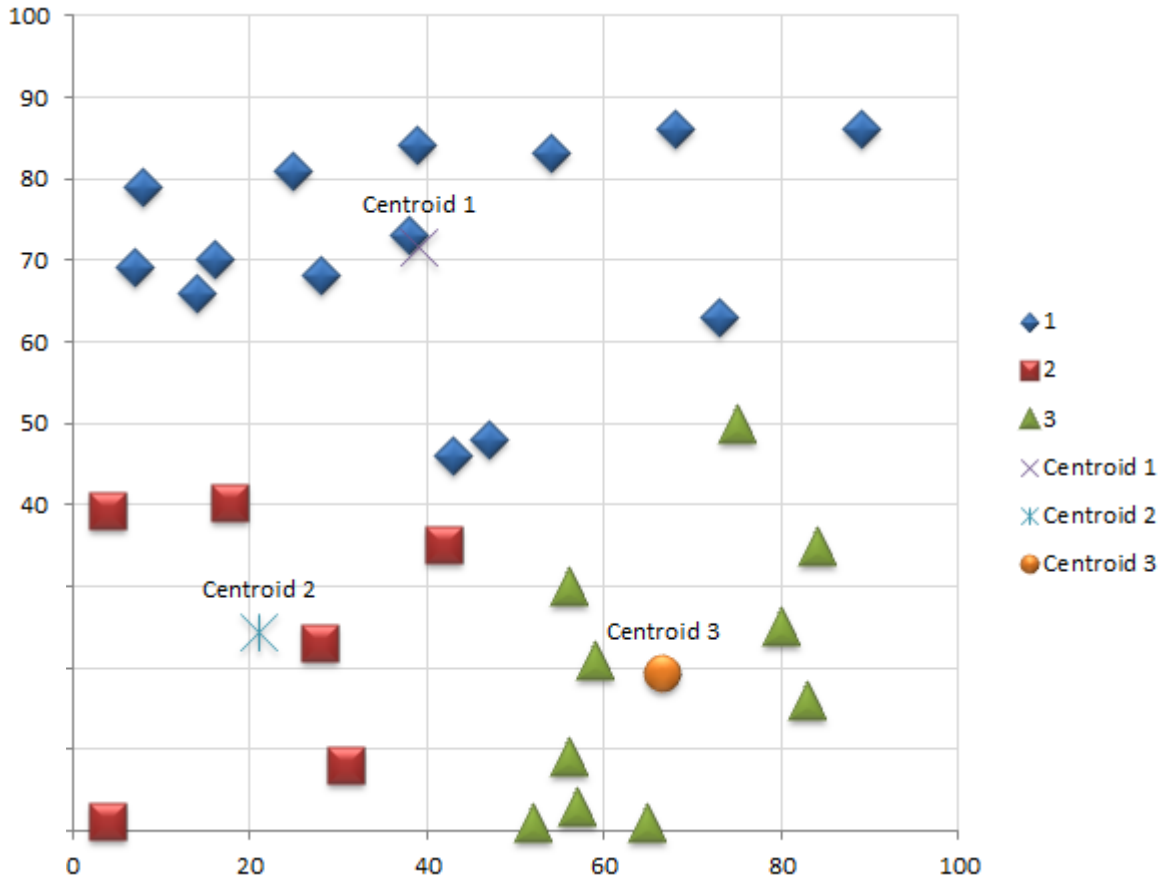
라는 개념을 반영한다.

뒷부분에서 더 자세하게 다루겠습니다.

K-MEANS

K (군집 개수- 지정 가능)

MEANS (군집의 CENTROID는 그 군집의 평균)



k-평균 군집화*k*-mean clustering 알고리즘의 특성

- 원리 단순하지만 성능이 좋아 인기 좋음
- 직관적으로 이해하기 쉽고 구현 쉬움
- 군집 개수 *k*, 거리측정 방법을 설정해야 함

Unit 03 | K-Means

K-Means(비계층적 군집화)

1. 데이터 내 객체 중 임의로 K개의 군집 중심점(Centroid) 설정

2. 모든 객체에 대해 각 군집 중심점까지의 거리 계산

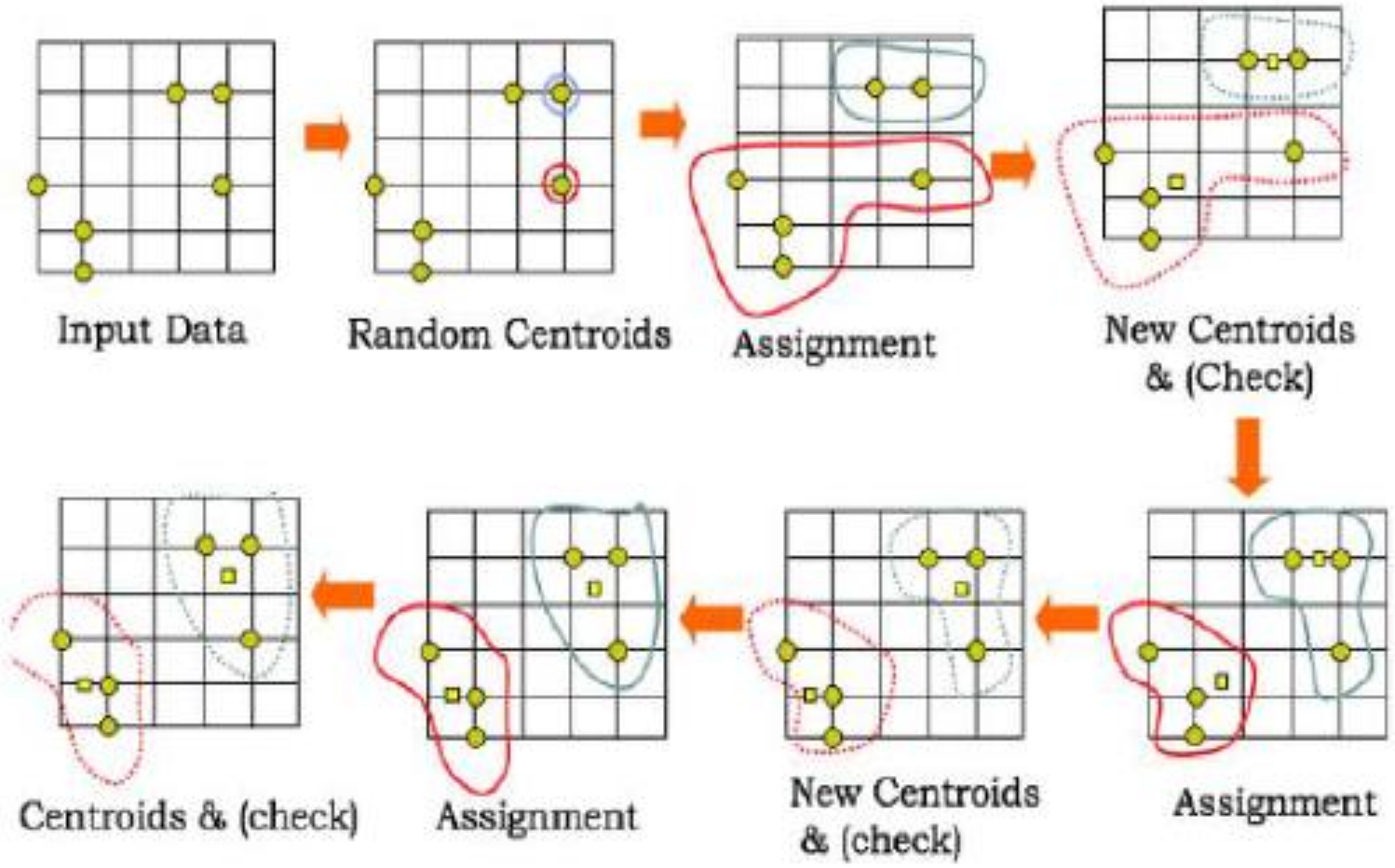
3. 모든 객체를 가장 가까운 군집 중심점이 속한 군집으로 할당

4. 각 군집의 중심점 재설정

5. 군집의 중심점이 변경되지 않을 때까지 1~4 반복

   (또는 적당한 범위 내로 수렴하거나 적당한 반복회수에 도달할 때까지 반복)

Unit 03 | K-Means



https://media.giphy.com/media/12vVAGkaqHUqCQ/giphy.gif

K_MEANS의 구현이 과제이므로

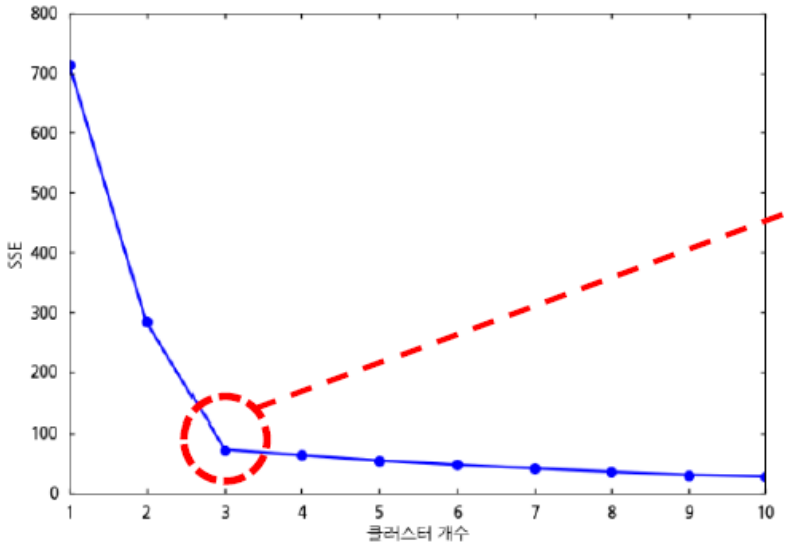
잘이해하면 빠르게 과제 끄읕!

군집 개수(K)의 선택

1. 경험적 방법 : $\sqrt{n}$, n = 데이터의 수

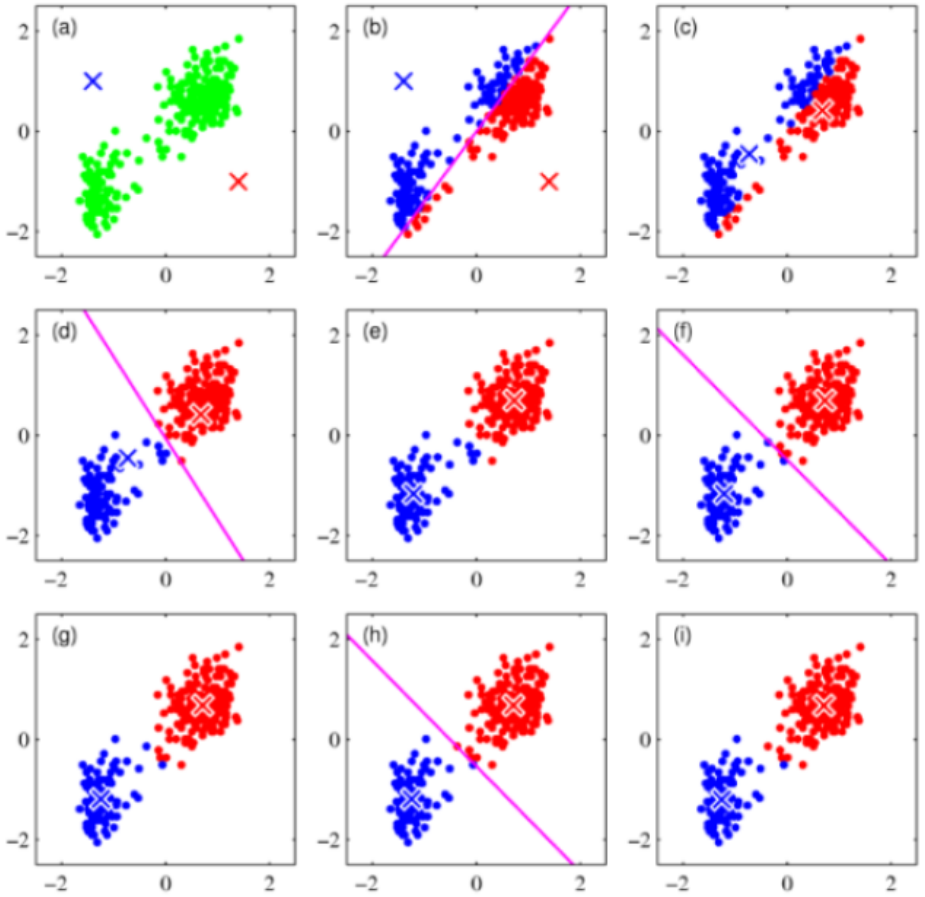
2. Elbow Point 기법

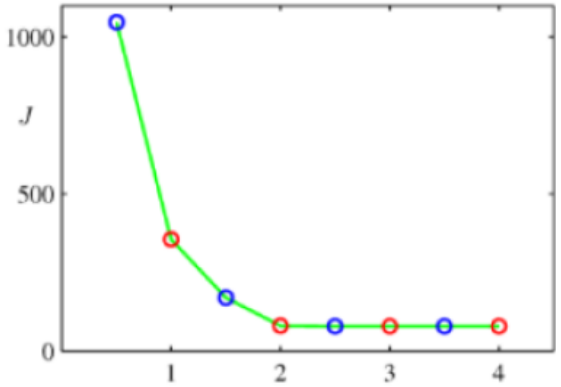
군집 내 분산을 기준으로!!



최적의 군집 개수!

Unit 03 | K-Means





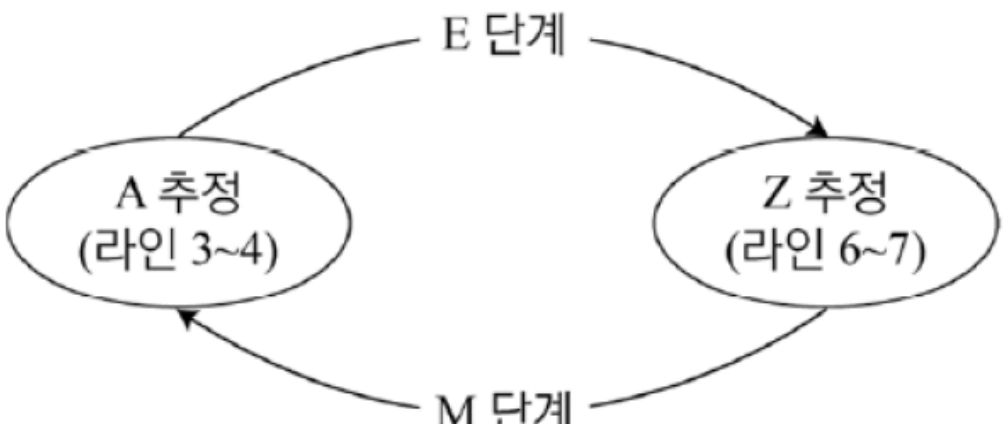
과정을 반복할수록 J값이 줄어든다

- 관측치가 군집에 영구히 할당되는 것이 아니라 최종 결과를 개선시키는 방향으로 이동한다

EM (expectation maximization) 기초

- k-평균에서 훈련집합 $\mathbb{X}$와 군집집합 C (행렬 $\mathbf{A}$)는 각각 입력단과 출력단에서 관찰 가능
- 중간 단계의 임시 변수 Z (입출력단에서 보이지 않기 때문에 은닉변수latent variable라 부름)
- k-평균의 할당assignment과 갱신update 과정은

  Z의 추정 (E 단계)과 $\mathbf{A}$의 추정 (M 단계)을 번갈아 가면 수행하는 EM 알고리즘과 유사

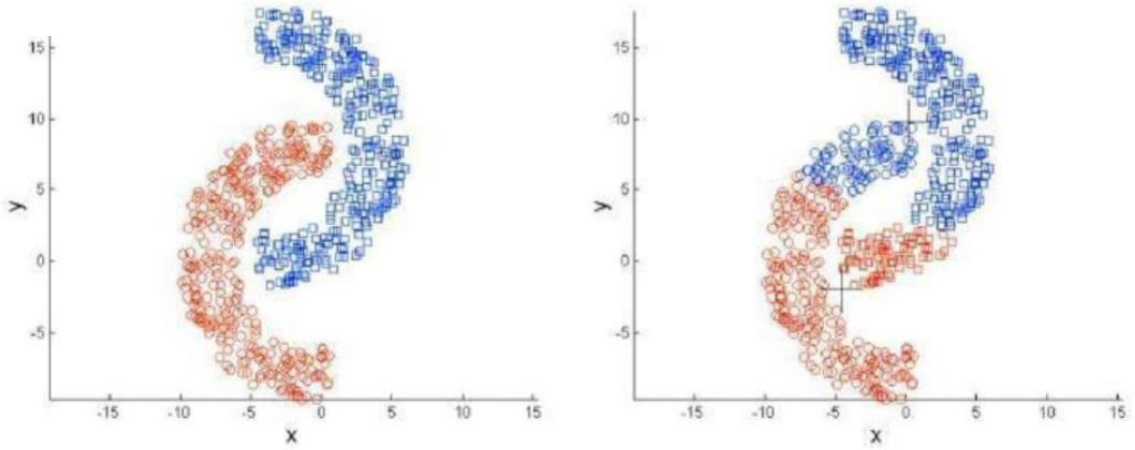


Unit 03 | K-Means

단점

1. 중심점을 기반으로 할당하기에 이상치에 굉장히 민감하다.
2. 초기화에 따라 다른 결과가 나타난다. 일관성 X
3. 수치화된 자료에만 사용할 수 있다. (거리 기반)

구형이 아닌 형태의 군집을 판별하기 어려움



이를 보완한 알고리즘들이 존재 !!

다중 시작 k-평균

- k-평균은 [알고리즘 6-1]의 라인 1에서 초기 군집 중심이 달라지면 최종 결과가 달라짐
- 다중 시작은 서로 다른 초기 군집 중심을 가지고 여러 번 수행한 다음,

  가장 좋은 품질의 해를 취함

Unit 03 | K-Means

평균법(MEANS)이 이상치에 약한 것을 보완→ K-MEDOIDS

k-평균과 *k*-중앙객체medoids 군집

- *K*-평균은 [알고리즘 6-1]의 라인 7에서 샘플의 평균으로 군집 중심을 갱신 (잡음에 민감)
- *K*-중앙객체는 실제 존재하는 객체들 중 하나를 뽑아 대표 객체로 선정하고
  이 객체를 중심으로 군집 중심을 갱신 (*k*-평균에 비해 잡음에 둔감)
  - 중앙객체medoids: 객체 집합에서 수학적으로 대표적인 객체

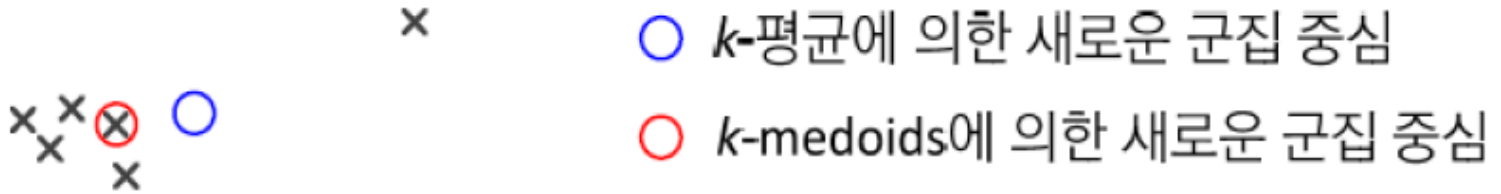
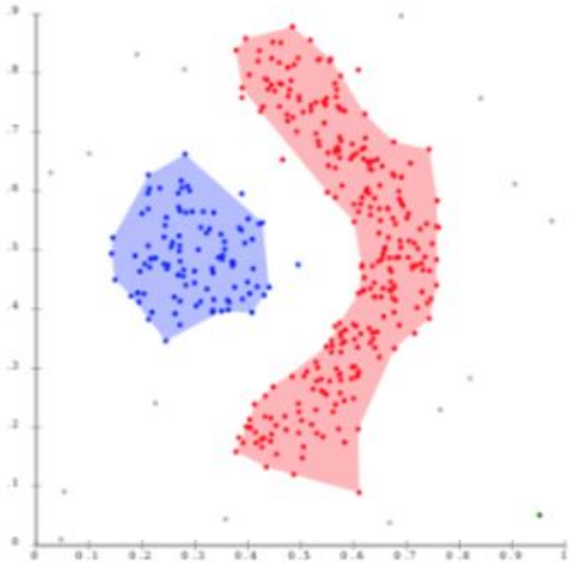


그림 6-4 *k*-평균과 *k*-medoids가 군집 중심을 갱신하는 과정

DBSCAN 방법

어느 점을 기준으로 반경 x내에 점이 N개 이상 있으면 하나의 군집으로 인식

K-means와 달리 최적의 개수를 설정하지 않아도 되며,
클러스터의 밀도에 따라서 클러스터를 서로 연결하기에 군집을 잘찾는다.



군집 개수를 자동으로 찾아줌

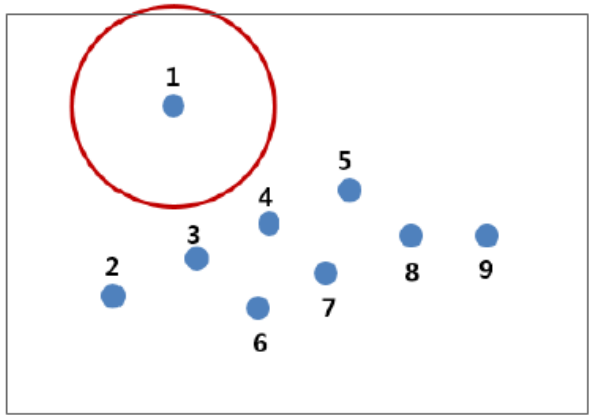
→ 군집 개수의 주관성 보완

노이즈 데이터를 따로 처리

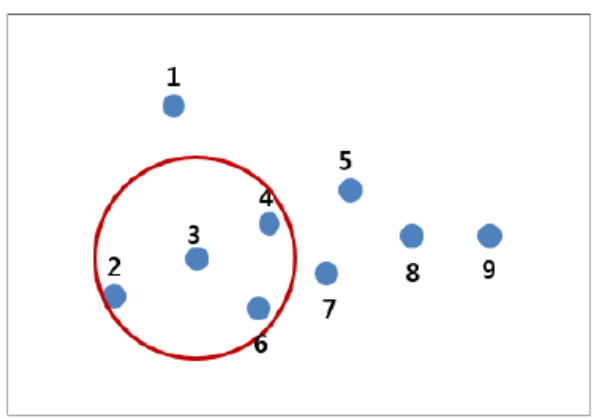
→ 이상치에 둔감.

점 P에서부터 거리 e(epsilon)내에 점이 m(minPts)개 있으면 하나의 군집으로 인식!

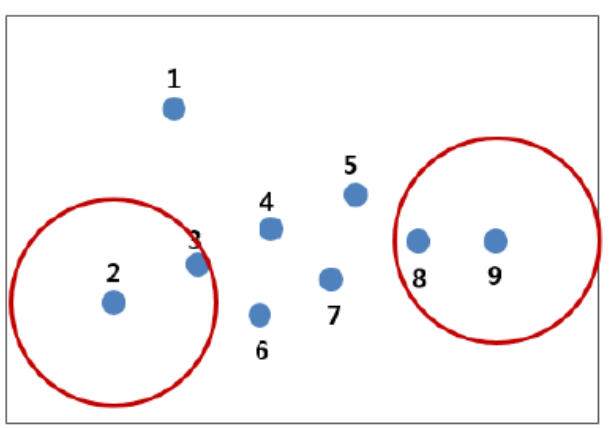
m=4 일때



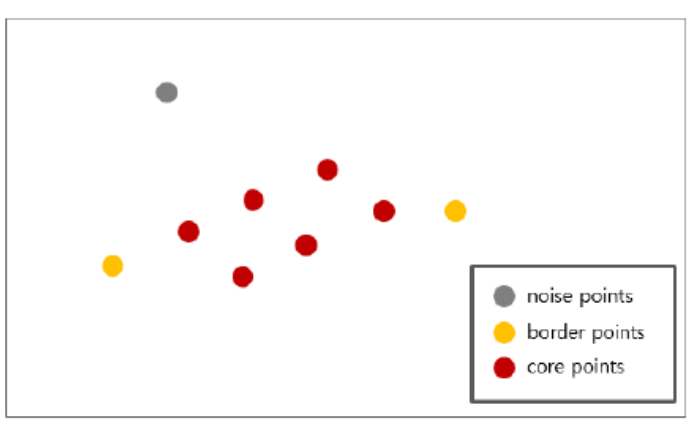
노이즈 데이터

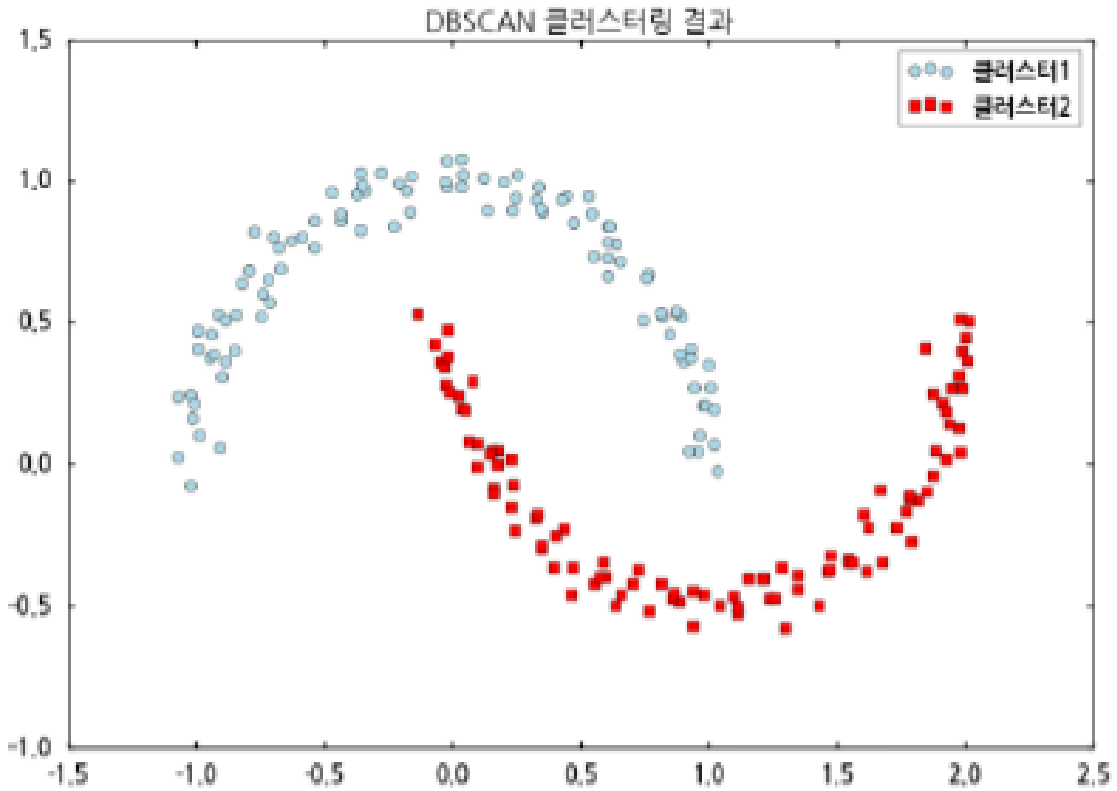


코어 데이터



경계 데이터





<특징>

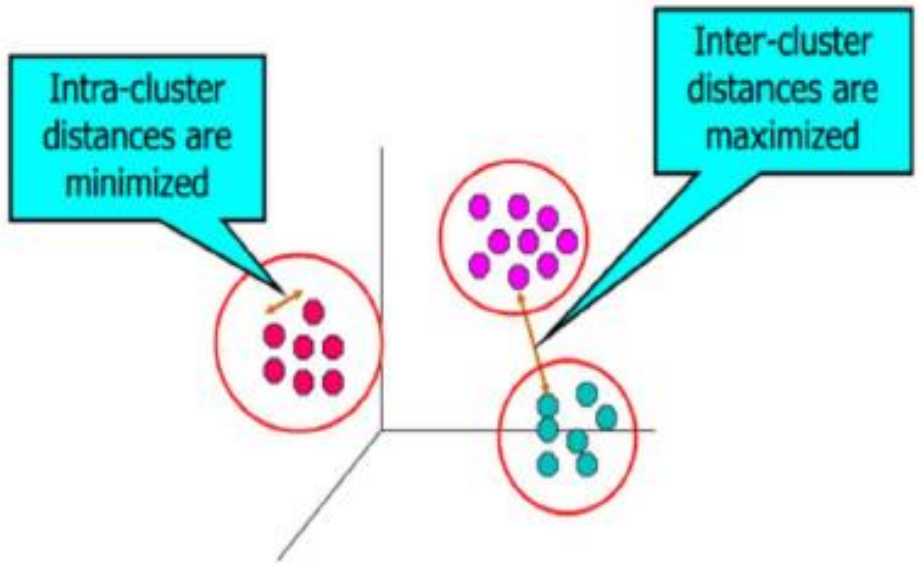
- K- means와 같이 클러스터의 수를 정하지 않아도 됨
- 비선형 경계의 군집을 구하는 것도 가능

  (밀도에 따라 클러스터를 서로 연결하기 때문)
- 노이즈 데이터를 따로 분류하여 노이즈 데이터들이

  군집에 영향을 주지 않음

Unit 04 | 모델 평가.

모델 평가



- Inter-cluster distance
  클러스터 간의 거리를 최대로

- Intra-cluster distance
  각 클러스터 내 데이터의 거리는 작게

클러스터의 평가 척도

내부 평가

- 스스로 클러스터링 된 데이터를 기반으로 평가

1. Dunn Index
2. 실루엣 (Silhouette)

- 외부 평가법도 존재! (라벨링값을 이용!)

< 1타깃값으로 군집 평가하기 >

- **군집 알고리즘의 결과를 실제 정답 클러스터와 비교하여 평가할 수 있는 지표**

1. ARI (adjusted rand index)
2. NMI (normalized mutual information)

- ARI : 1(최적일 때)와 0(무작위로 분류될 때)

  하지만, 대부분의 상황에서는 라벨링값이 없음.

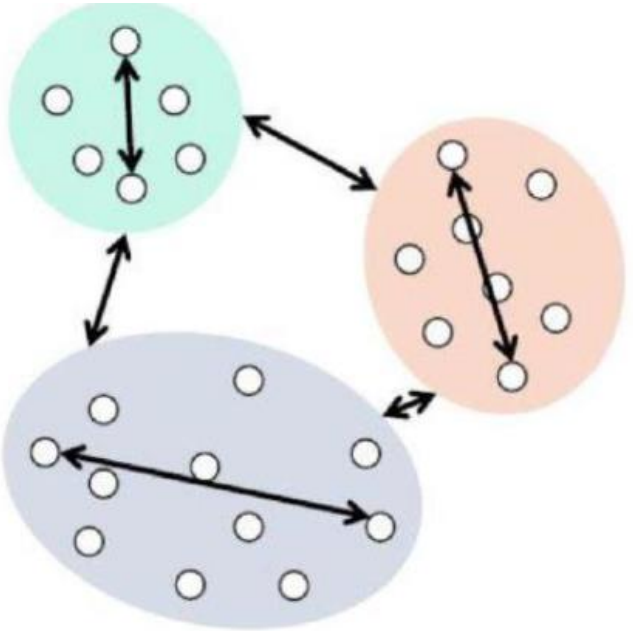
  따라서, 라벨링값 없이 군집의 응집도를 보는게 가장 보편적 방법.

1. Dunn Index

$$DI = \frac{\text{군집과 군집 사이의 거리 중 최소값}}{\text{군집 내 객체 간 거리 중 최대값}}$$

군집과 군집 사이의 거리가 클수록,
군집 내 객체 간 거리가 작을수록
군집화가 잘 되었군!

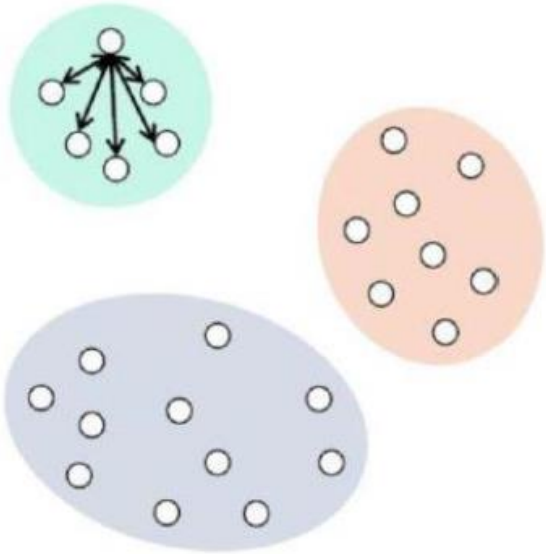
= DI가 큰 모델



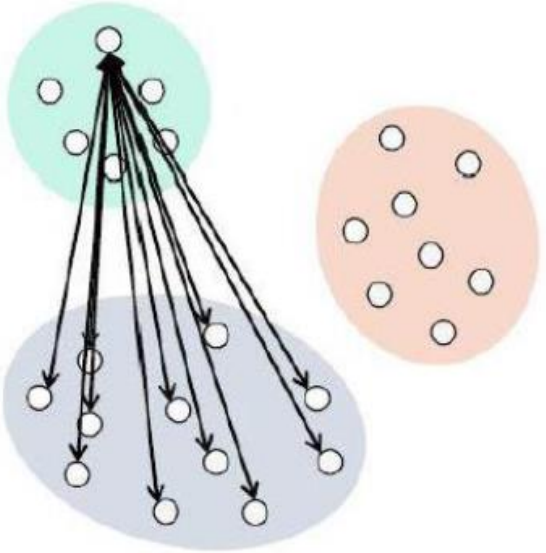
2. Silhouette

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

S(i)가 1에 가까울 수록 좋은 모델



$a(i)$: 객체 i로부터 같은 군집 내 모든 다른 객체들 사이 평균 거리 (클러스터 내 데이터 응집도)



$b(i)$: 객체 i로부터 다른 군집 내 객체들 사이의 평균 거리 중 최소값 (클러스터 간 분리도)

실루엣 점수는 클러스터의 밀집 정도를 계산하는 것으로, 높을수록 좋으며, 최대 점수는 1이다.

- 평가지표 , 알고리짐들의 공통적 특성
- → 결과를 확인해야지 알 수 있다.

비지도학습을 포함한 머신 러닝을 잘하는 방법

→할 수 있는 것은 전부 해보고, 가장 좋은 결과를 선택한다.

결국 머신 러닝의 본질은 노가다!

들어주셔서 감사합니다.