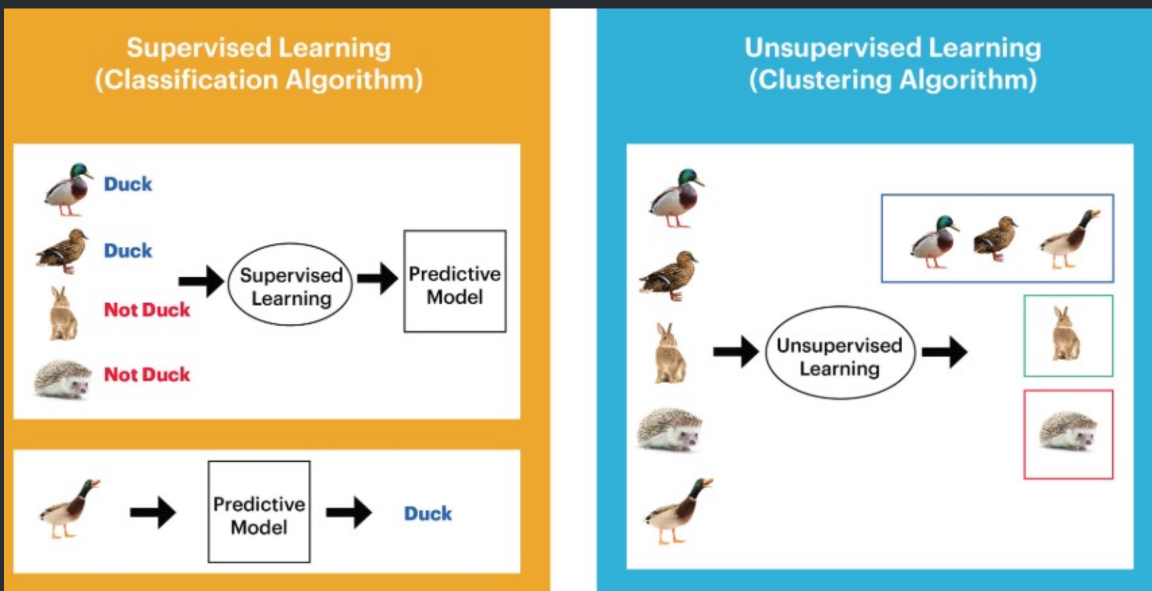
Unsupervised Learning

1



2

Machine Learning

UN-SUPERVISED LEARNING

3

3

Clustering

4

클러스터링은 데이터에서 비슷한 객체들을 하나의 그룹으로 묶는 것

5

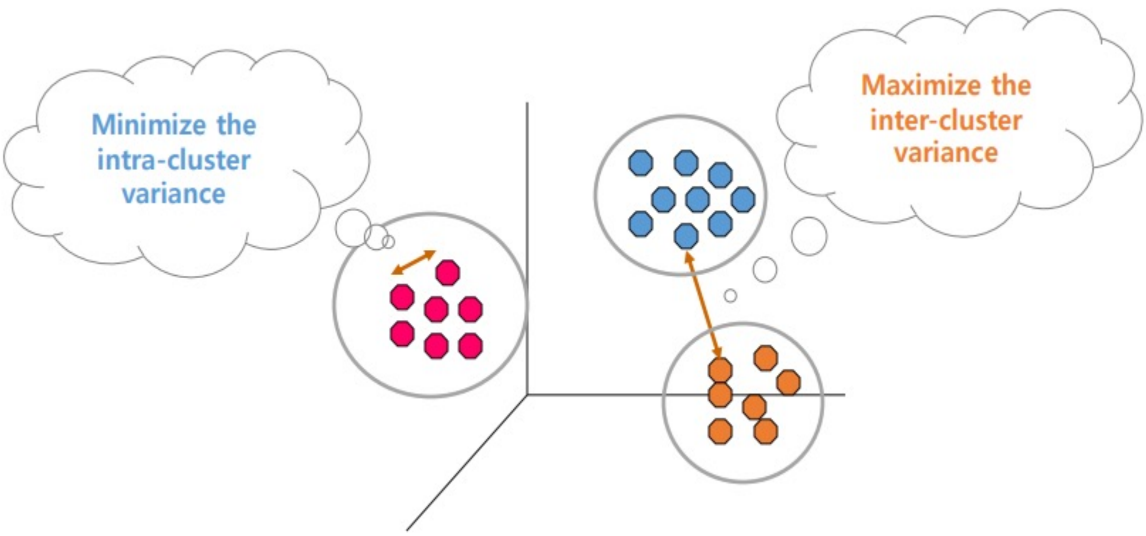
그럼 데이터가 비슷한 기준은?

6

유사도 (거리) 정보 기반

7

CLUSTERING



8

CLUSTERING ALGORITHM

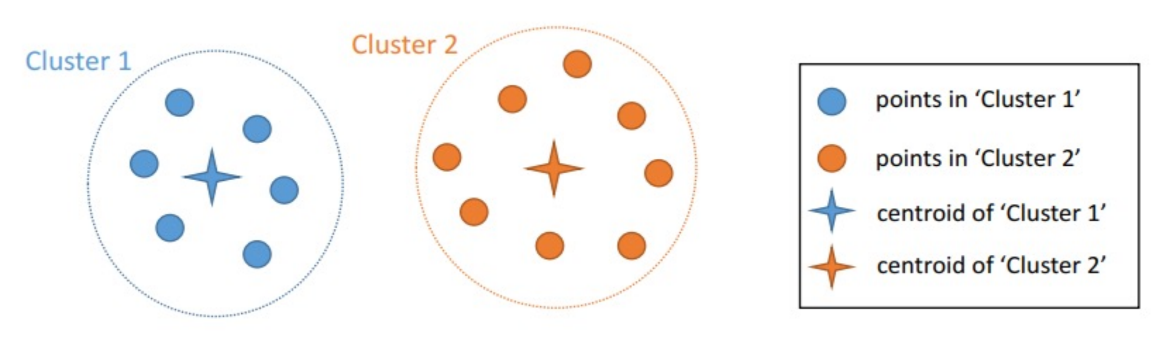
- K-means clustering
- Hierarchical clustering
- Density-based spatial clustering of applications with noise (DBSCAN)
- Gaussian mixture model
- Self-organizing map (SOM)

9

K-means clustering

10

K-MEANS CLUSTERING



11

유사도

- $d(x_i, x_j)$: 데이터 x에 대해 두 데이터 x_i, x_j 간에 정의되는 임의의 거리
- 유클리디언 거리, 코싸인 유사도 등 벡터에서 정의되는 모든 거리 척도

12

유클리디안 거리 (L2 DISTANCE)

- 피타고라스 정의

두 점 P와 Q가 각각 $P=(p_1, p_2, p_3, \ldots, p_n)$와 $Q=(q_1, q_2, q_3, \ldots, q_n)$의 좌표를 갖을 때 두 점 사이의 거리를 계산하는 유클리디안 거리 (Euclidean distance)공식은 다음과 같습니다.

$$\sqrt{(p_1-q_1)^2+(p_2-q_2)^2+\ldots+(p_n-q_n)^2}=\sqrt{\sum_{i=1}^{n}(p_i-q_i)^2}$$

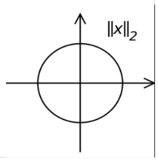
$$\frac{1}{1+Ed}$$

13

예제) 4번과 가장 가까운 데이터는 무엇인가요?

#	A	B	C	D
1	3	2	0	2
2	1	2	3	0
3	2	2	2	2
4	1	5	0	0

$$dist(D1, Q) = \sqrt{(3-1)^2+(2-5)^2+(0-0)^2+(2-0)^2} = \sqrt{17}$$
$$dist(D2, Q) = \sqrt{(1-1)^2+(2-5)^2+(3-0)^2+(0-0)^2} = \sqrt{18}$$
$$dist(D3, Q) = \sqrt{(2-1)^2+(2-5)^2+(2-0)^2+(2-0)^2} = \sqrt{18}$$



14

코사인 유사도

- 두 벡터 사이의 코사인 각도를 구해 서로의 유사도를 구하는 방식
- 텍스트 데이터의 유사도를 구하는 방법 중 하나
- 데이터 셋의 길이 차이가 심한 상황일 때도 데이터들의 유사도를 판단 할 수 있다.

코사인 유사도 : -1　　코사인 유사도 : 0　　코사인 유사도 : 1

15

코사인 유사도

$$\cos.similarity = \frac{\overset{\text{내적 (Dot product)}}{A \cdot B}}{\underset{\text{A 벡터공간}}{\| A \|}\ \underset{\text{B 벡터공간}}{\| B \|}} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} (A_i)^2} \times \sqrt{\sum_{i=1}^{n} (B_i)^2}}$$

16

코사인 유사도

#	A	B	C	D	E	F	D	H
L1	8	8	2	2	0	0	0	0
L2	10	8	3	0	0	0	0	0
L3	0	0	3	2	8	6	6	8
L4	0	0	3	0	8	6	2	8

#	내적	NORM A	NORM B	NORMA*NORM B	Cos.Sim
L1 X L2	8*10+8*8+2*3+2*0	(8*8+8*8+2*2+2*2)^0.5	(10*10+8*8+3*3)^0.5	11.66*13.15	150/153.3878
	150	11.6619	13.1529	153.39	**0.9779**
L3 X L4	3*3+8*8+6*6+6*2+8*8	(3*3+2*2+8*8+6*6+6*6+8*8)^0.5	(3*3+8*8+6*6+2*2+8*8)^0.5	14.59*13.30	185/194.1667
	185	14.5945	13.3041	194.17	**0.9528**
L1 X L3	2*3+2*2	(8*8+8*8+2*2+2*2)^0.5	(3*3+2*2+8*8+6*6+6*6+8*8)^0.5	11.66*14.59	10/170.1996
	10	11.6619	14.5945	170.20	**0.0588**

17

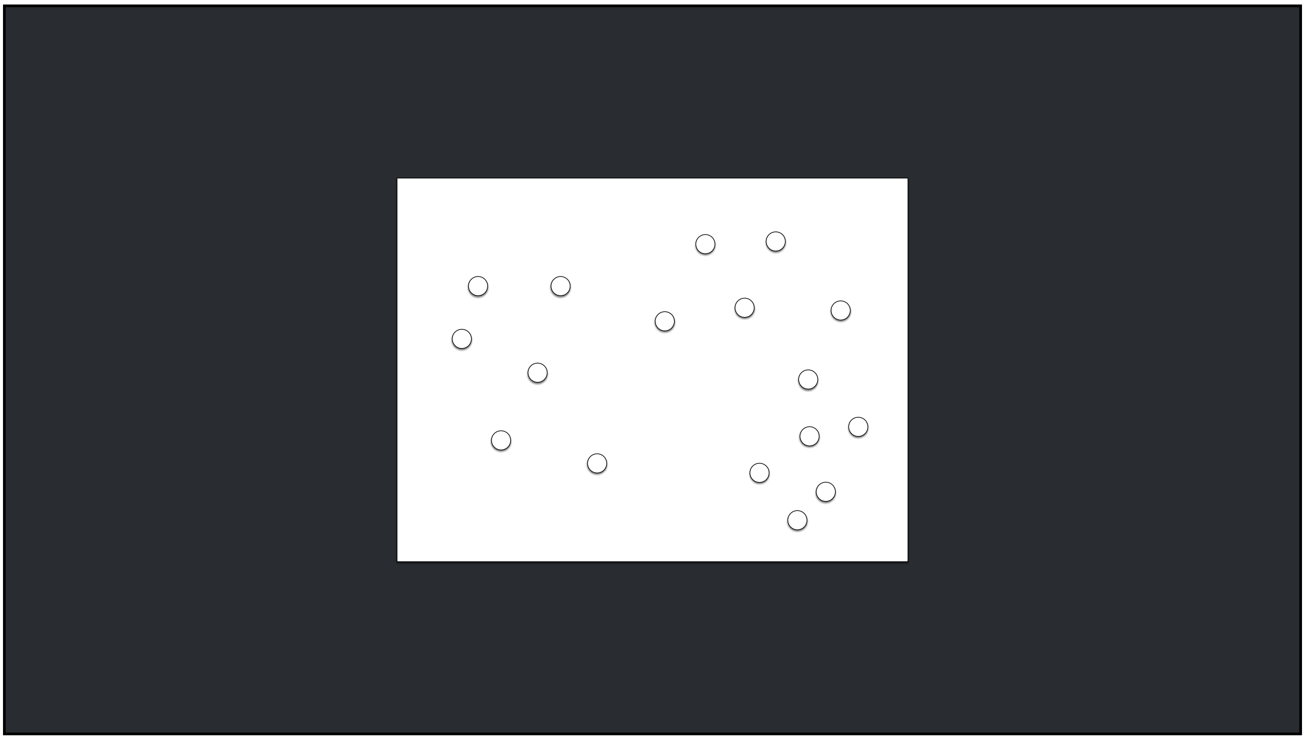
K-MEANS CLUSTERING

$$\arg\min_{\mathbf{C}} \sum_{i=1}^{K} \sum_{\mathbf{x}_j \in C_i} ||\mathbf{x}_j - \mathbf{c}_i||^2$$

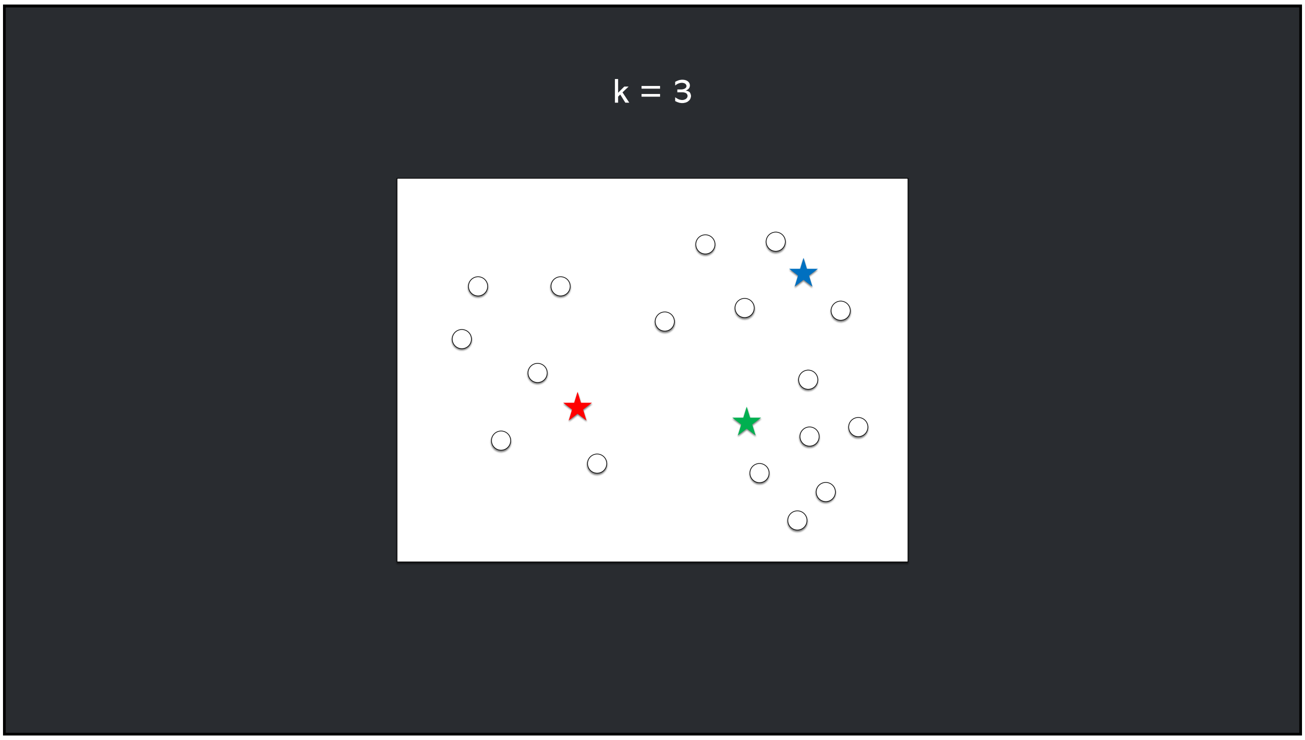
Algorithm 1 Basic K-means Algorithm.

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

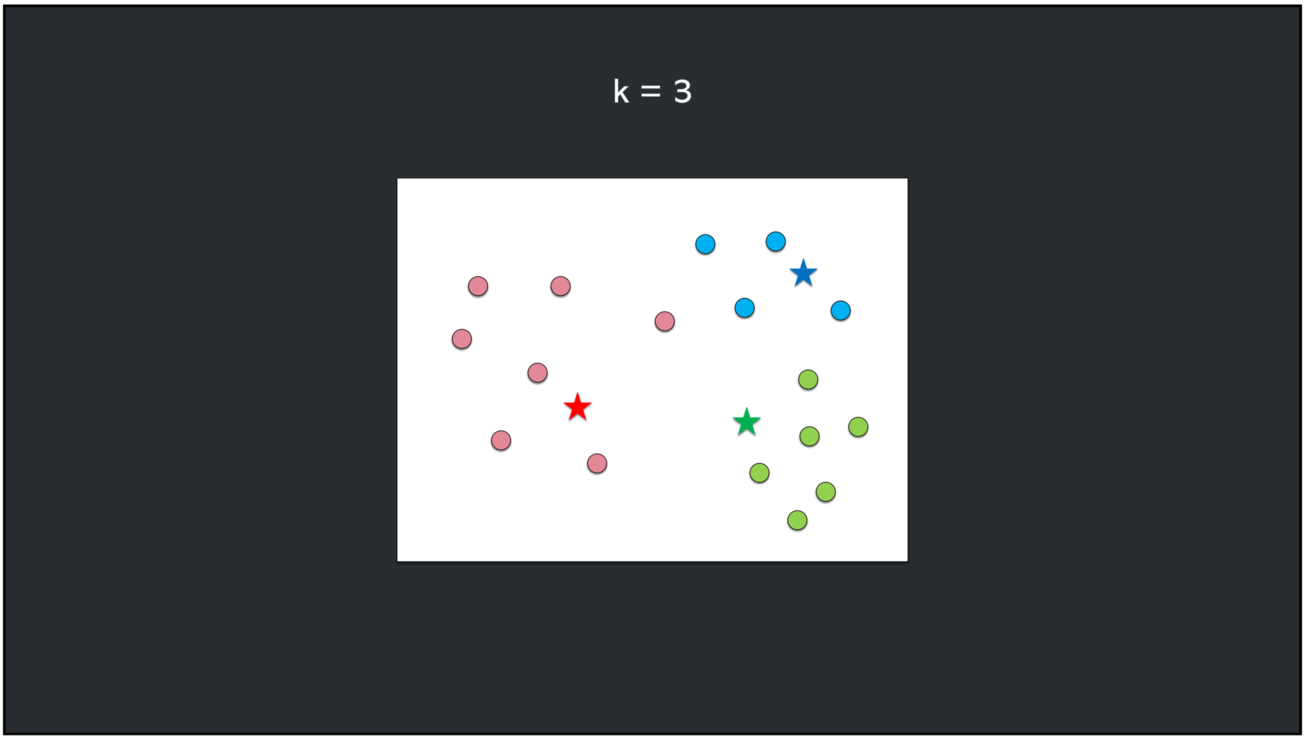
18



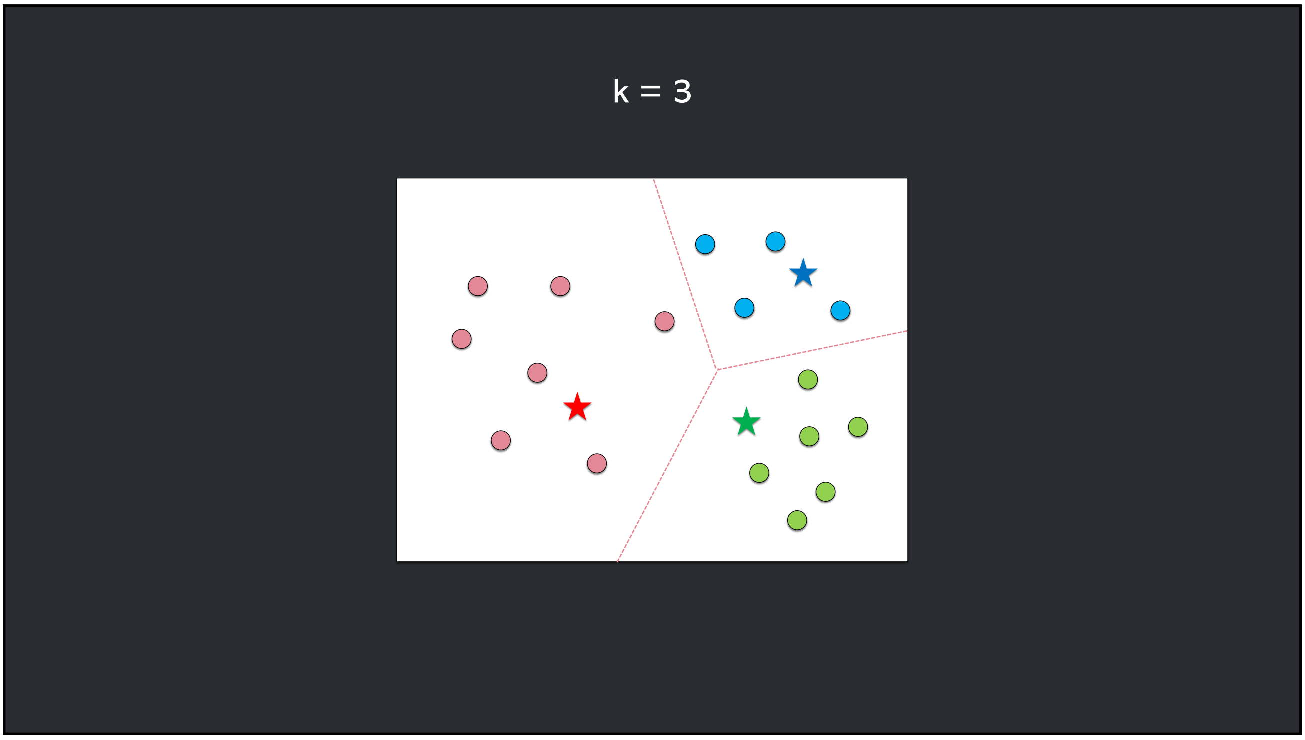
19



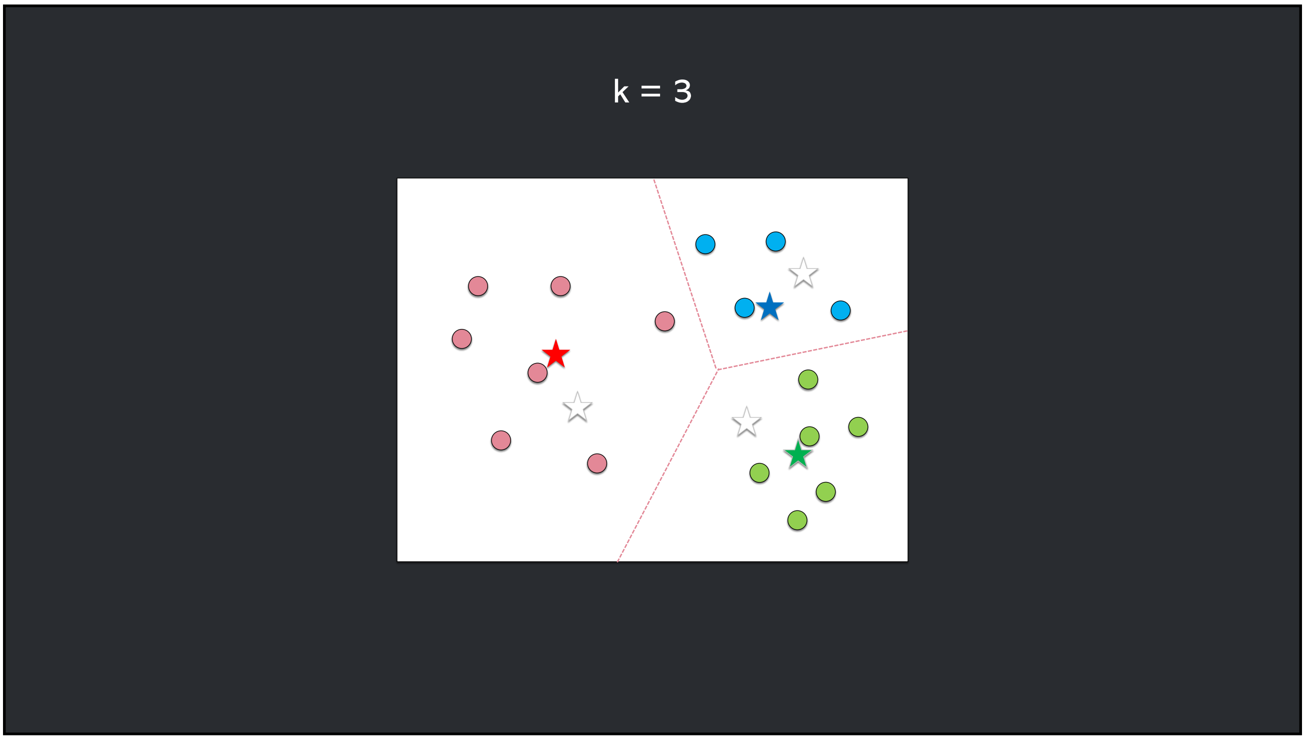
20



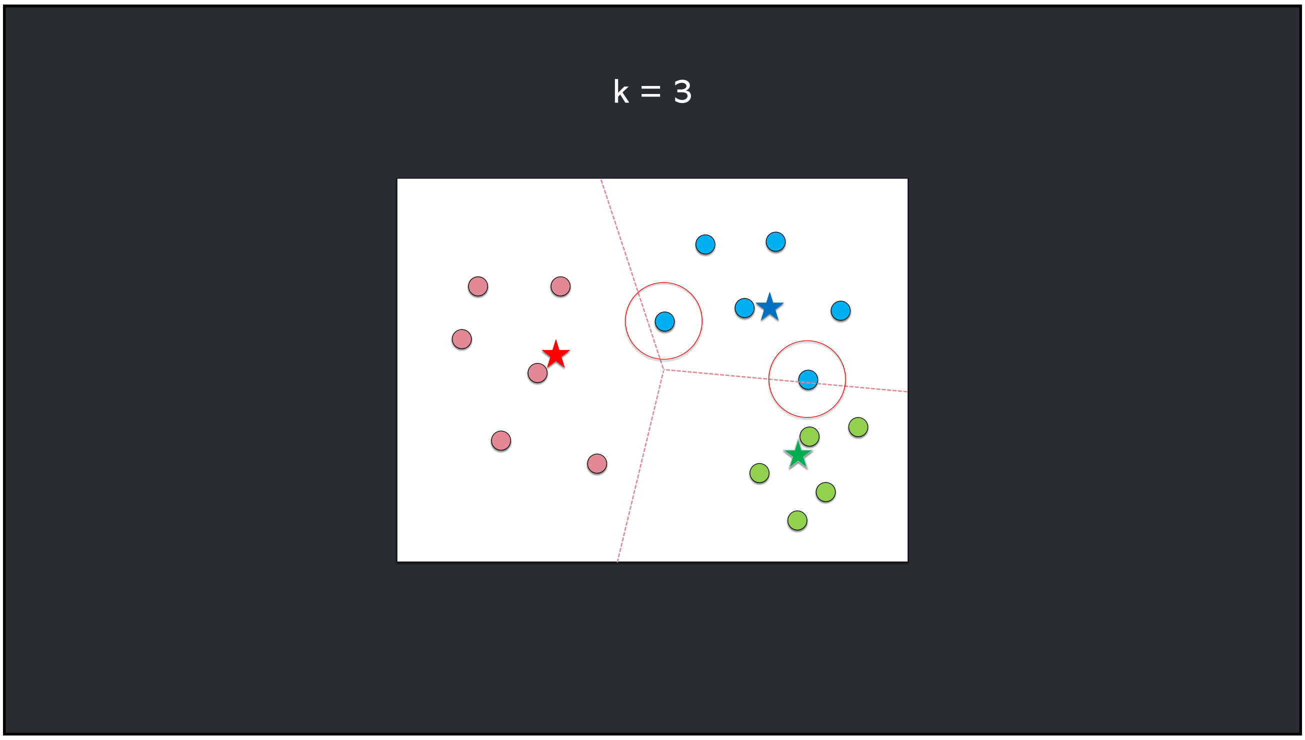
21



22

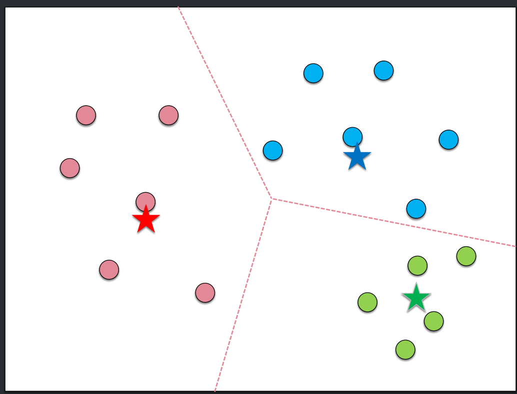


23



24

k = 3



epoch =2

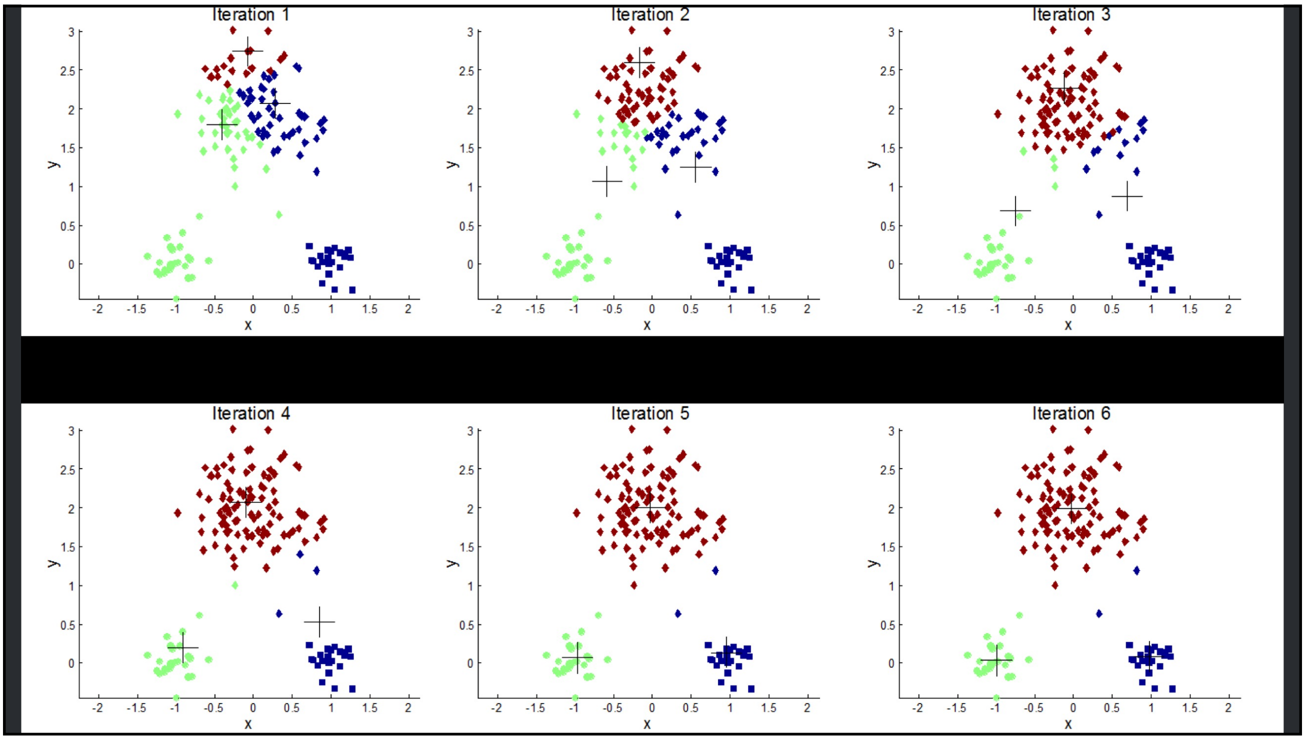
25

Weak points	
1	Sensitive results from Initial points
2	Ball-shaped clusters
3	Sensitive to noise points

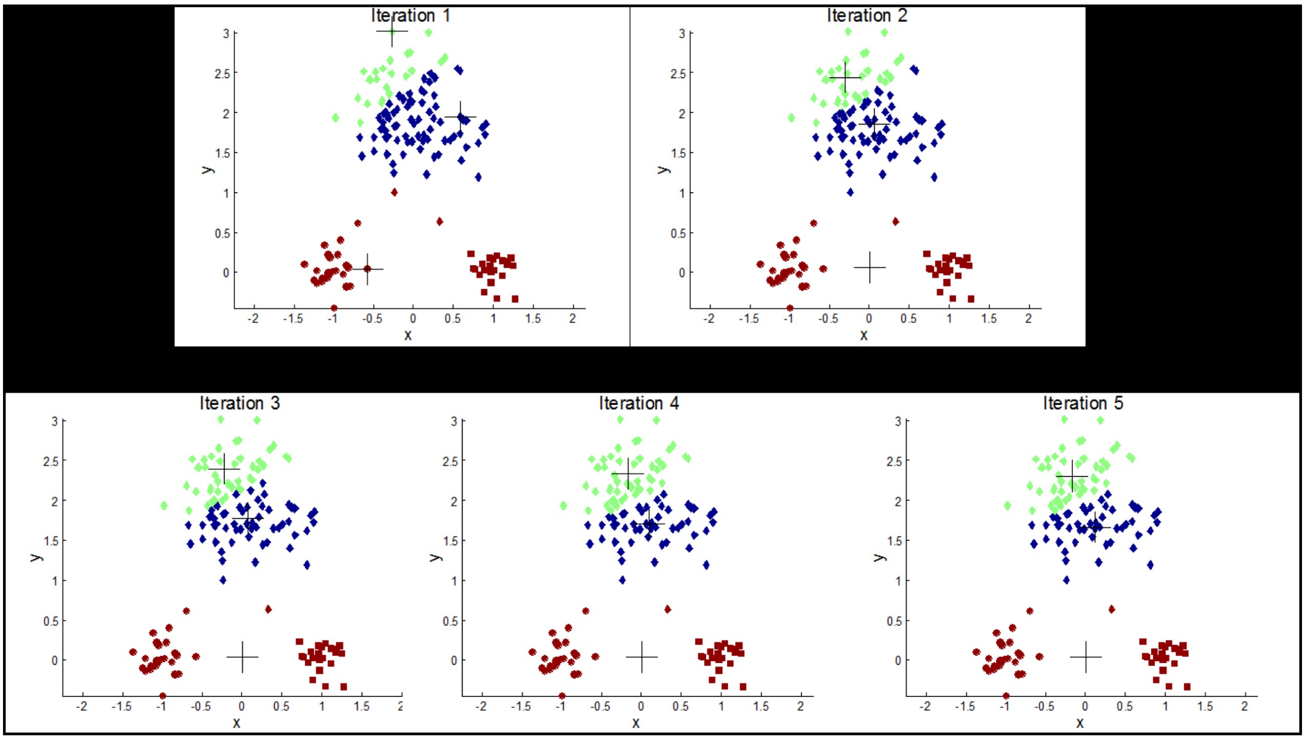
26

1. Sensitive results from Initial points

27



28


Iteration 1
Iteration 2
Iteration 3
Iteration 4
Iteration 5

29

1. 해결법

30

1	n_init
2	init='k-means++'

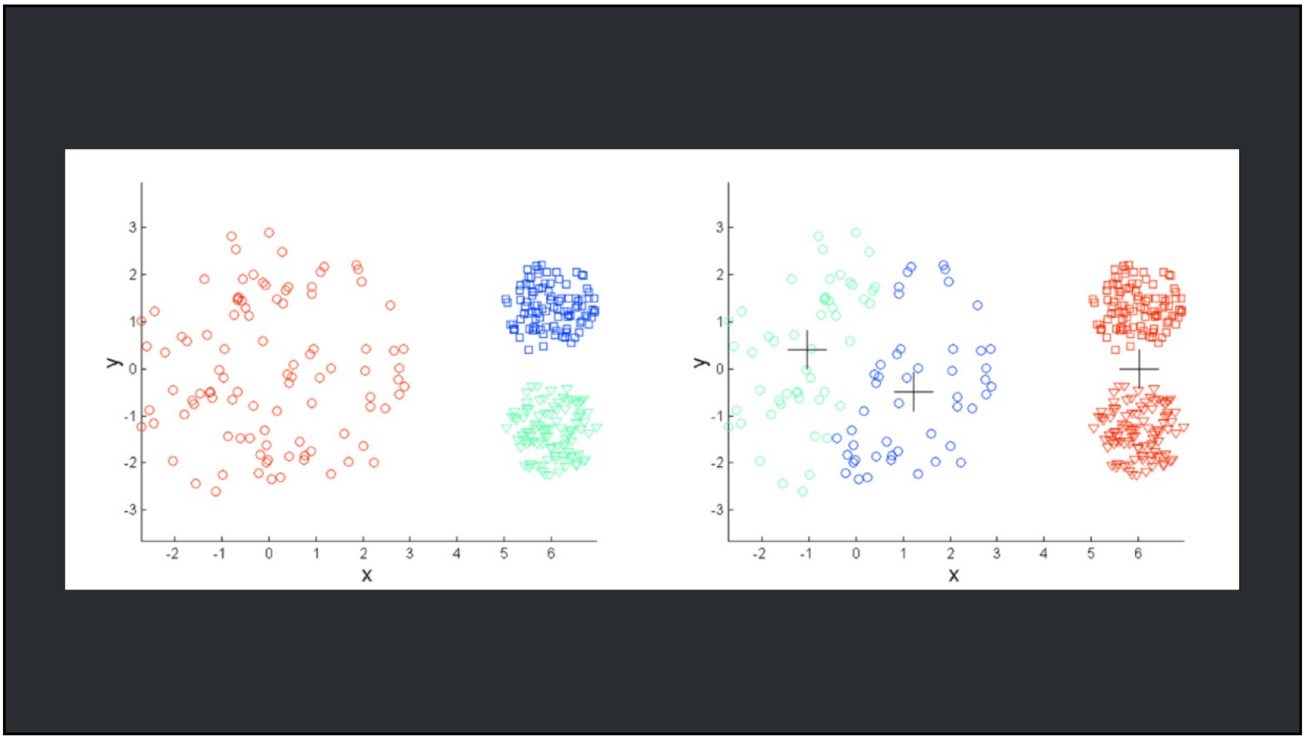
sklearn.cluster.KMeans

class sklearn.cluster. KMeans (n_clusters=8, init='k-means++', n_init=10, max_iter=300, tol=0.0001, precompute_distances='auto', verbose=0, random_state=None, copy_x=True, n_jobs=1, algorithm='auto') [source]

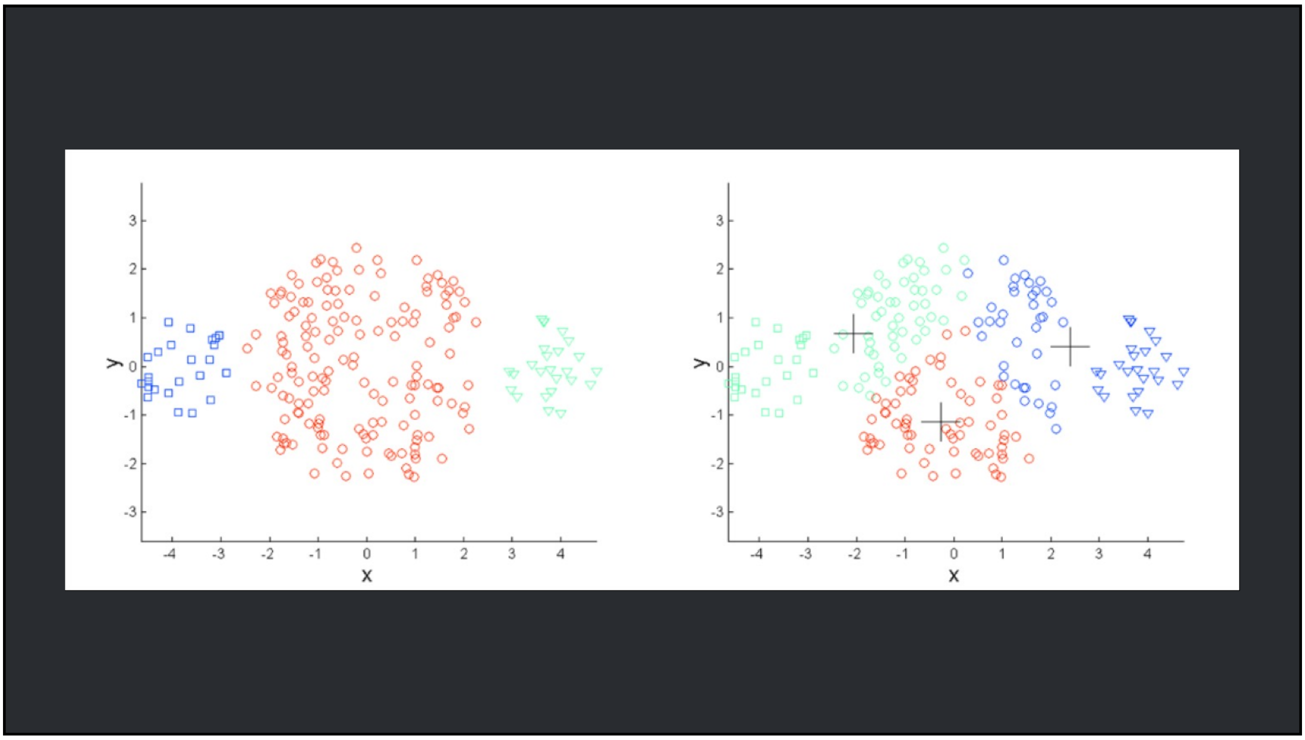
31

2. Ball-shaped clusters

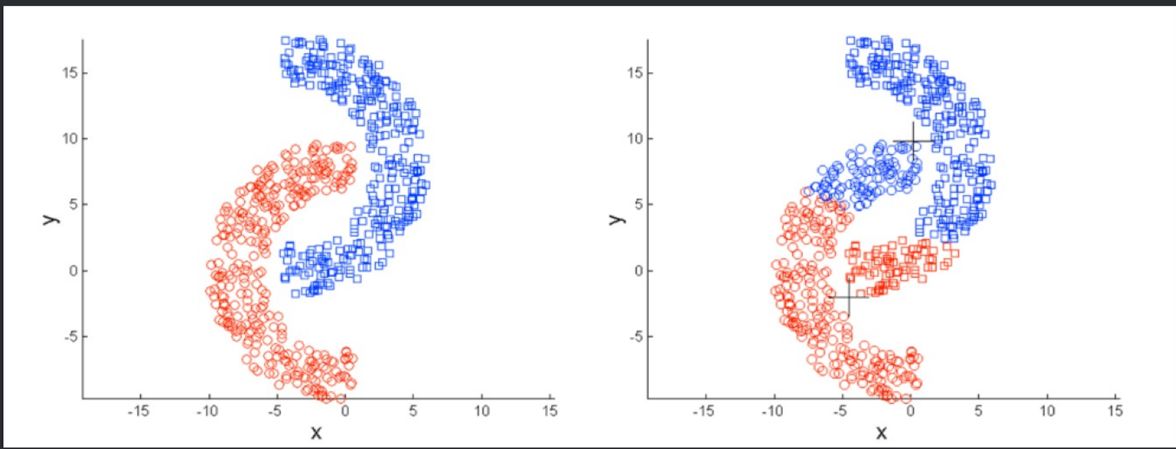
32



33



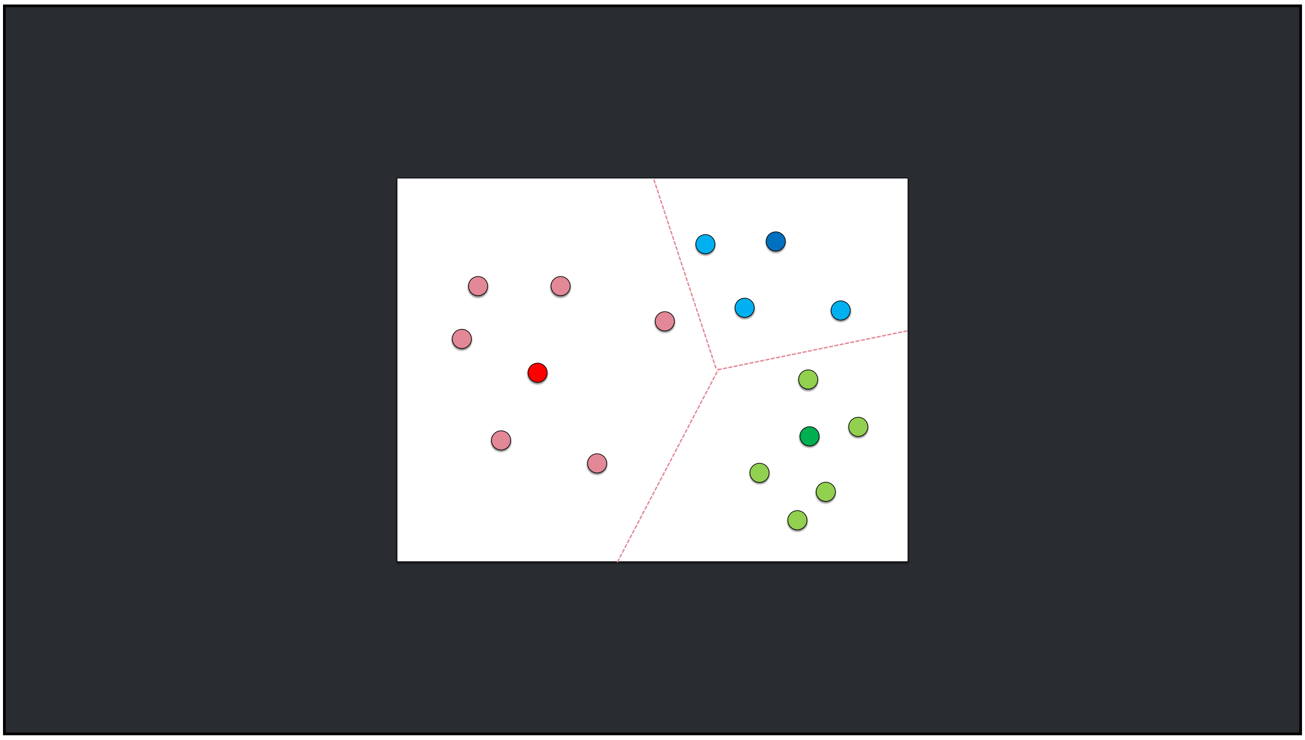
34



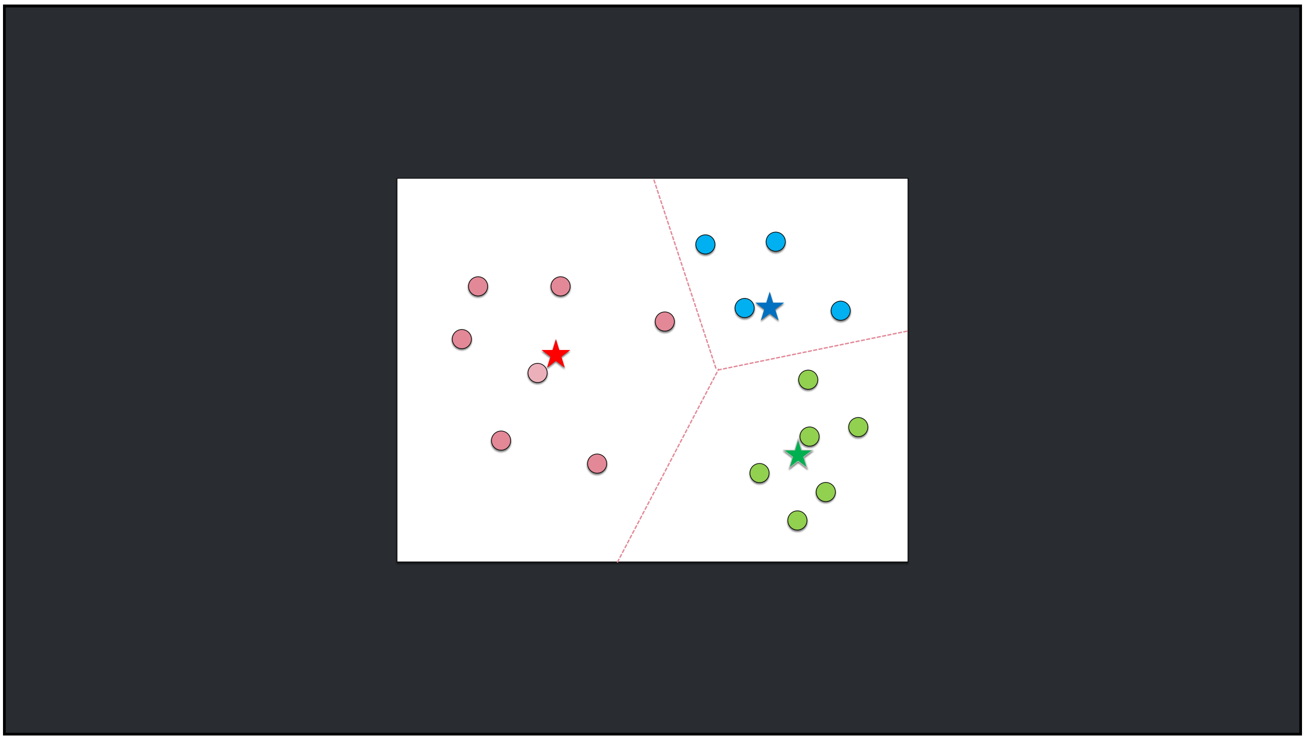
35

3. sensitive to noise points

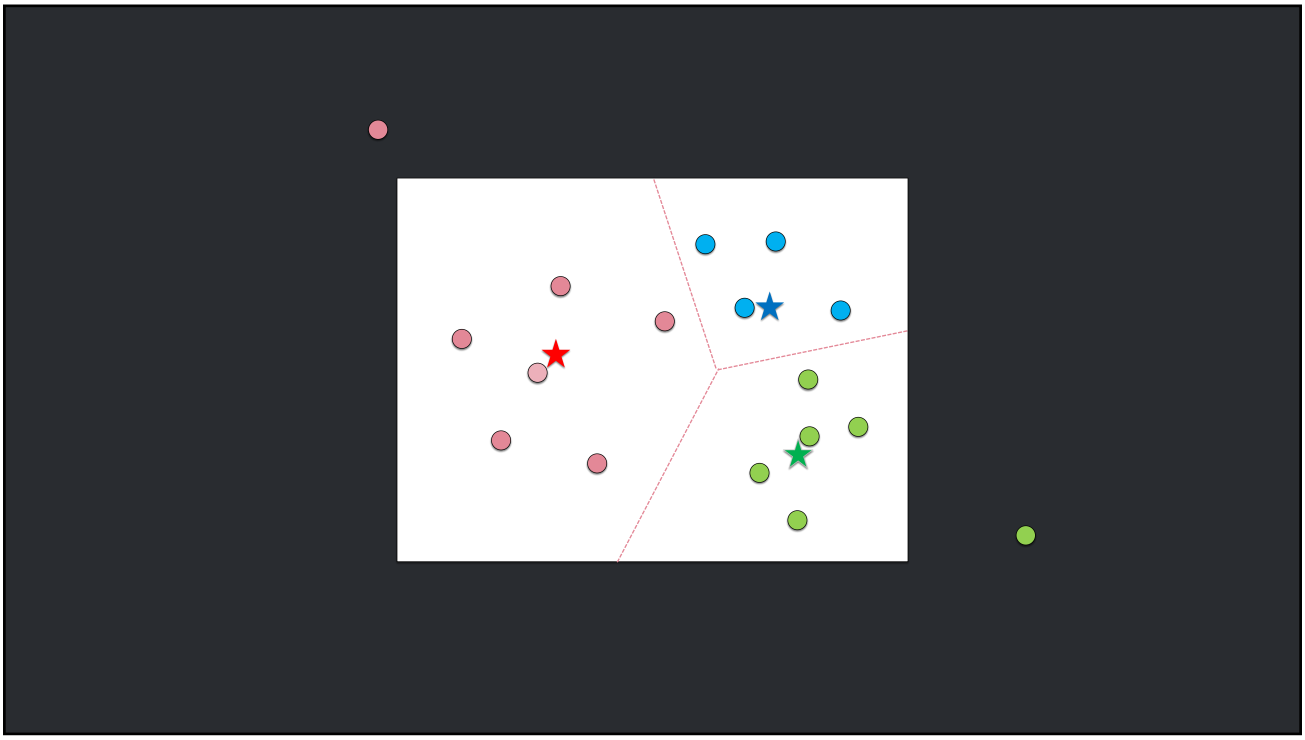
37



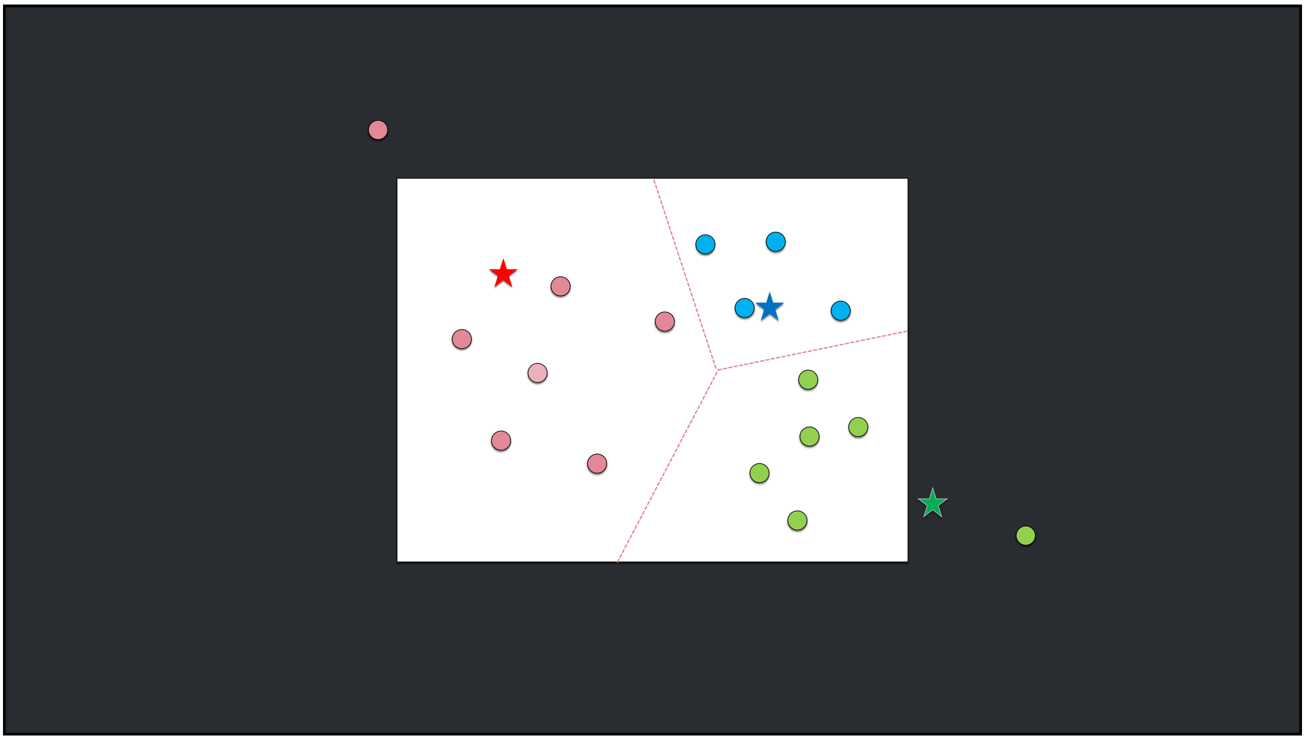
38



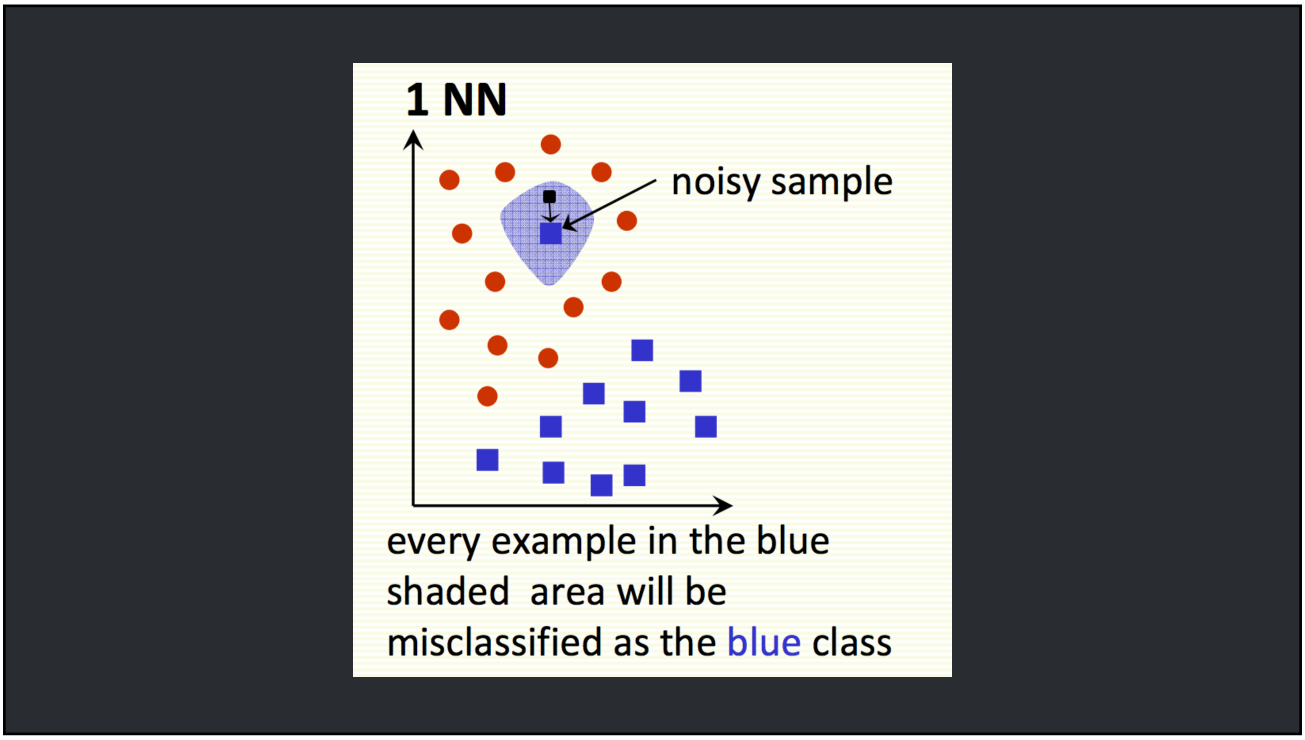
39



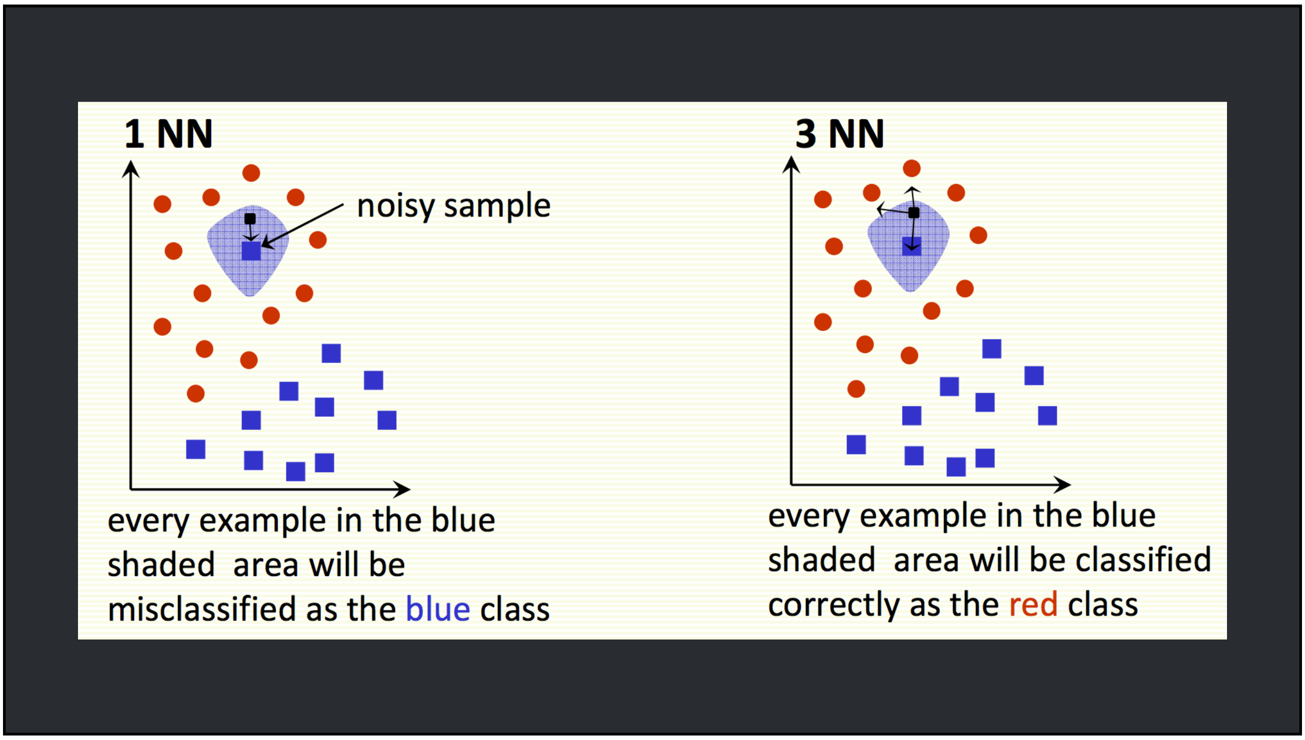
40



41


1 NN
noisy sample
every example in the blue
shaded area will be
misclassified as the blue class

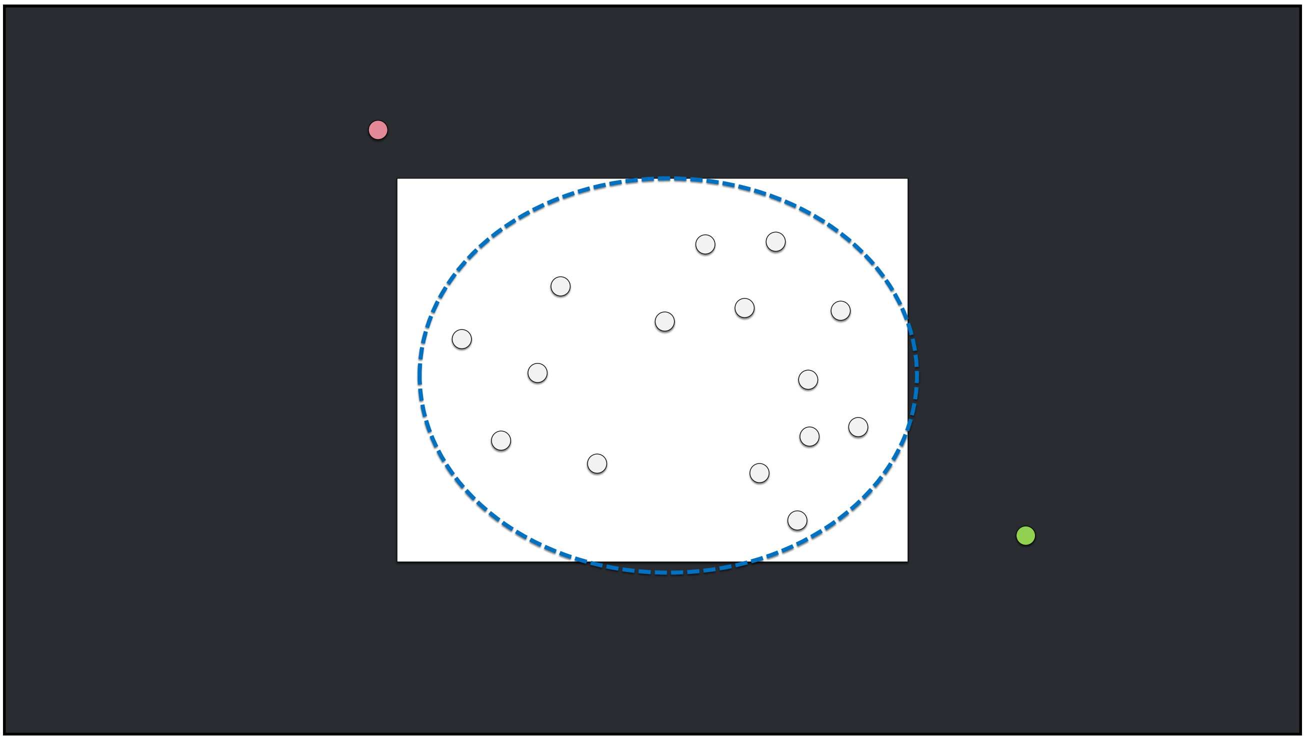
42


1 NN
noisy sample
every example in the blue
shaded area will be
misclassified as the blue class
3 NN
every example in the blue
shaded area will be classified
correctly as the red class

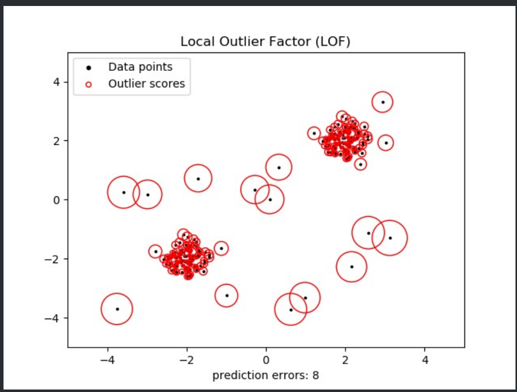
43

3. 해결법

44



45



46

k-means clustering 장단점

47

장점	계산이 쉽다. 다른 군집화 알고리즘에 비해 복잡도가 낮다
	구현이 쉽고 다양한 언어와 플랫폼에서 제공되는 알고리즘
단점	노이즈에 매우 민감
	군집 개수를 사전에 지정
	앞의 몇 가지 상황에서는 최적의 군집 구조를 찾기 어려움

48

Evaluation metrics for clustering

49

Sadly, there is no good way

50

So,

51

Sum of squared distance for each point to it's assigned centroid

52

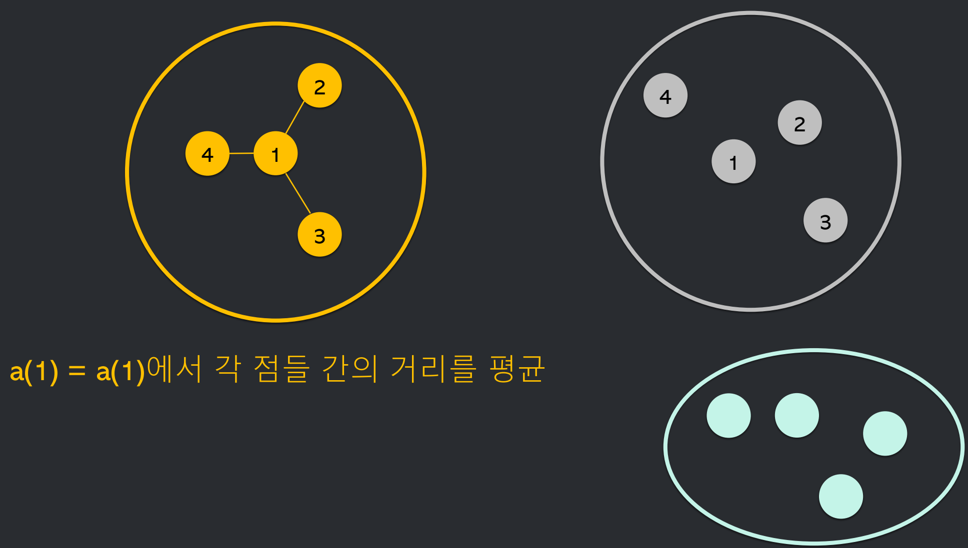
Silhouette score

53

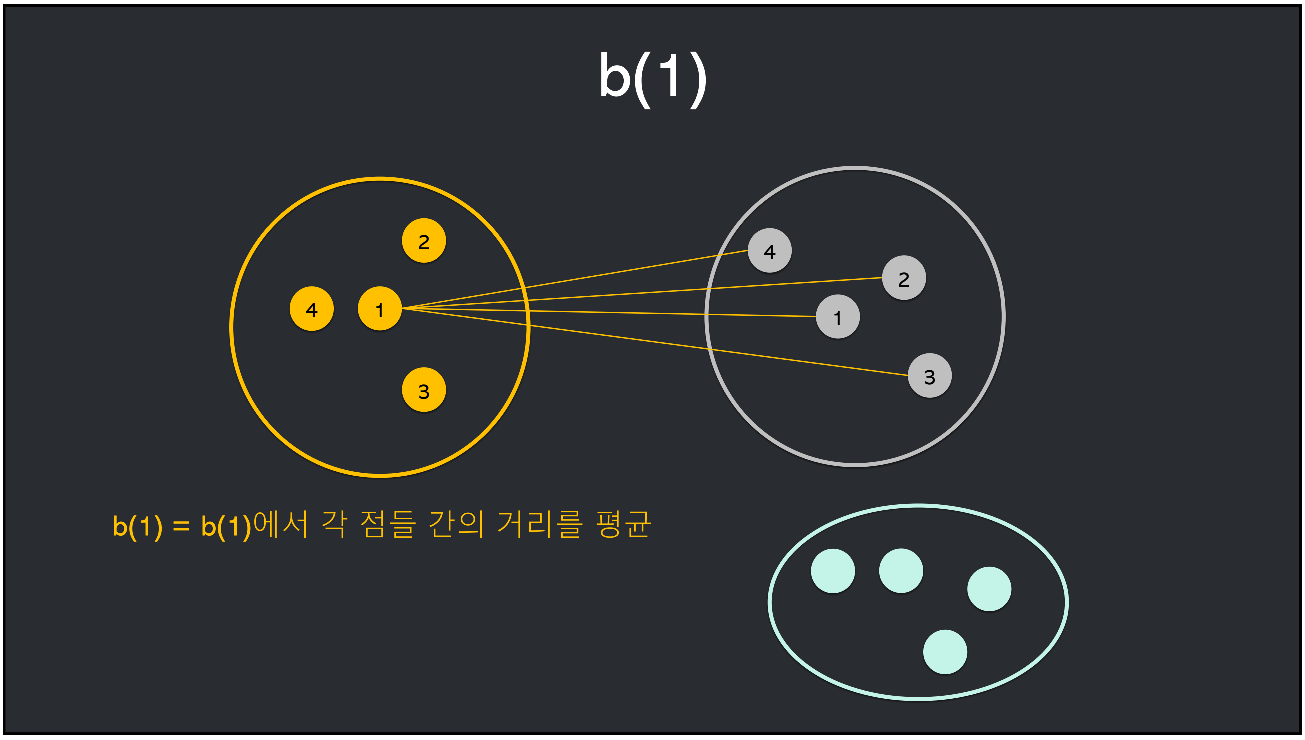
$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$$

54

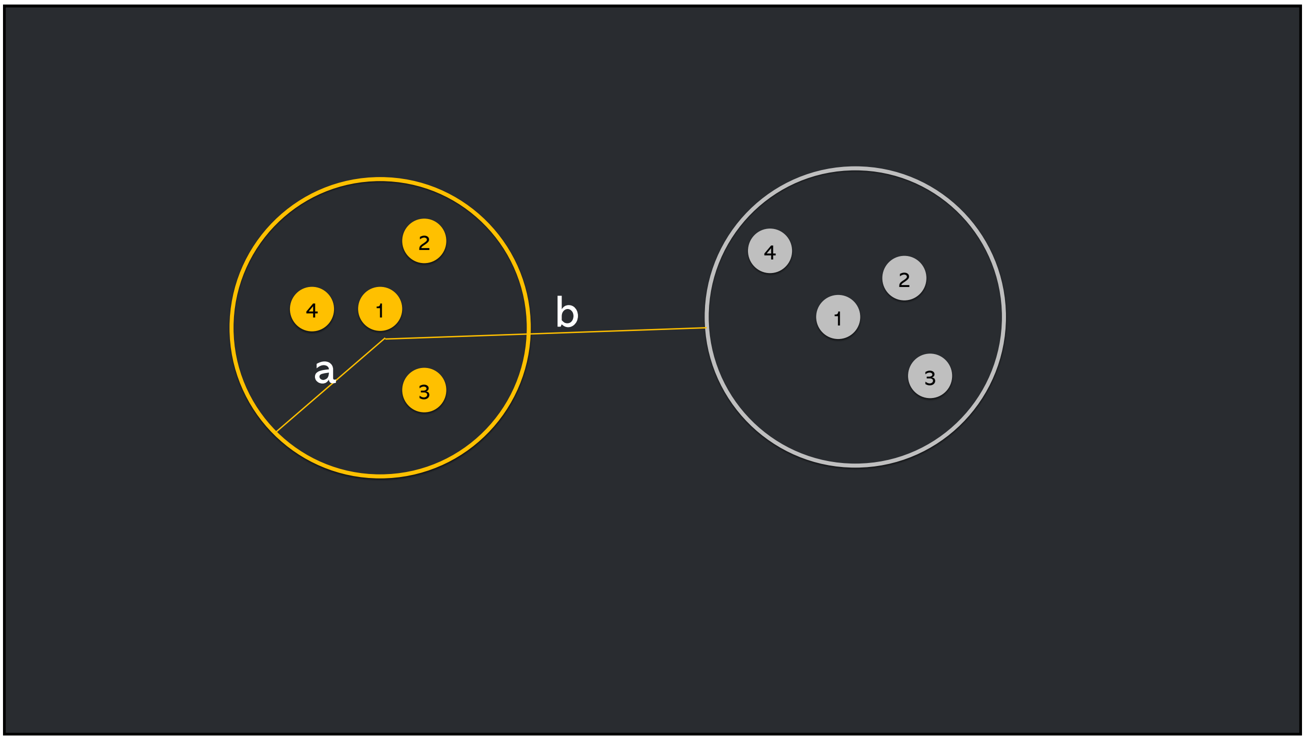
a(1)



55



56

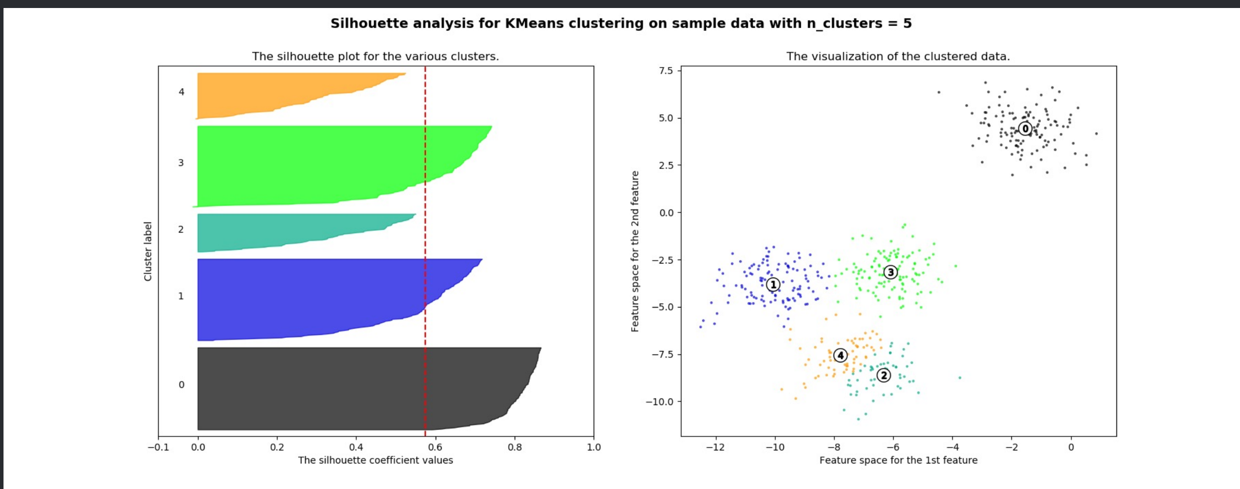


57

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

$$-1 \leq s \leq 1$$

58



59