SOON CHUN HYANG UNIVERSITY

Machine Learning

-KNN, Cluster Analysis-

SCH Univ.
Dept. of AI and Bigdata
Kim JinSeong

Contents

K-Nearest Neighbor (KNN)

Model for Classification and Prediction

- Model-Based Learning
  - 선형/비선형모델 (e.g., linear regression, logistic regression)
  - Neural Network
  - Decision Tree
  - Support Vector Machine

  → 데이터로부터 모델을 생성하여 분류/예측 진행

- Instance-Based Learning
  - K-Nearest Neighbor (KNN)
  - Locally weighted regression

  → 별도의 모델 생성 없이 인접 데이터를 분류/예측에 사용

Model for Classification and Prediction

- Model-Based Learning
  - 선형/비선형모델 (e.g., linear regression, logistic regression)
  - Neural Network
  - Decision Tree
  - Support Vector Machine

  → 데이터로부터 모델을 생성하여 분류/예측 진행

(1)

$$f(X)$$

분류/예측 모델

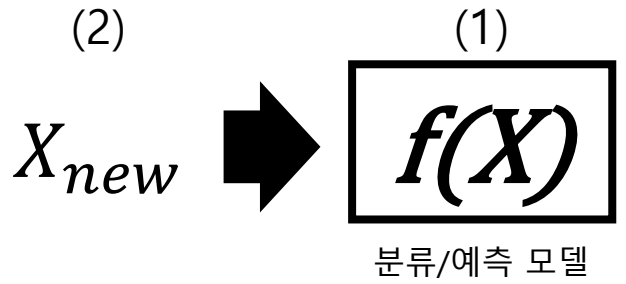
- Instance-Based Learning
  - K-Nearest Neighbor (KNN)
  - Locally weighted regression

  → 별도의 모델 생성 없이 인접 데이터를 분류/예측에 사용

Model for Classification and Prediction

- Model-Based Learning
  - 선형/비선형모델 (e.g., linear regression, logistic regression)
  - Neural Network
  - Decision Tree
  - Support Vector Machine

  → 데이터로부터 모델을 생성하여 분류/예측 진행



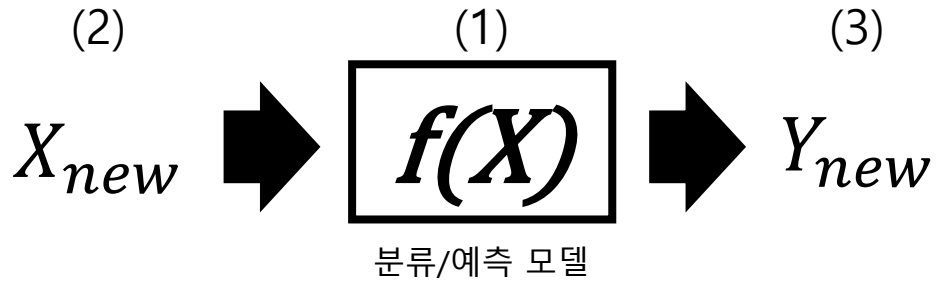
- Instance-Based Learning
  - K-Nearest Neighbor (KNN)
  - Locally weighted regression

  → 별도의 모델 생성 없이 인접 데이터를 분류/예측에 사용

Model for Classification and Prediction

- Model-Based Learning
  - 선형/비선형모델 (e.g., linear regression, logistic regression)
  - Neural Network
  - Decision Tree
  - Support Vector Machine

  → 데이터로부터 모델을 생성하여 분류/예측 진행



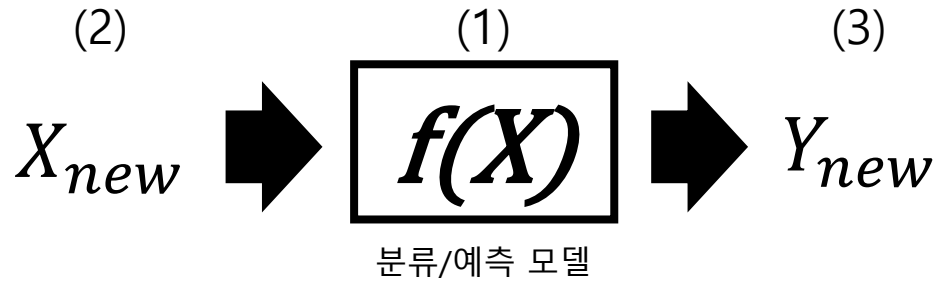
- Instance-Based Learning
  - K-Nearest Neighbor (KNN)
  - Locally weighted regression

  → 별도의 모델 생성 없이 인접 데이터를 분류/예측에 사용

Model for Classification and Prediction

- Model-Based Learning
  - 선형/비선형모델 (e.g., linear regression, logistic regression)
  - Neural Network
  - Decision Tree
  - Support Vector Machine

  → 데이터로부터 모델을 생성하여 분류/예측 진행



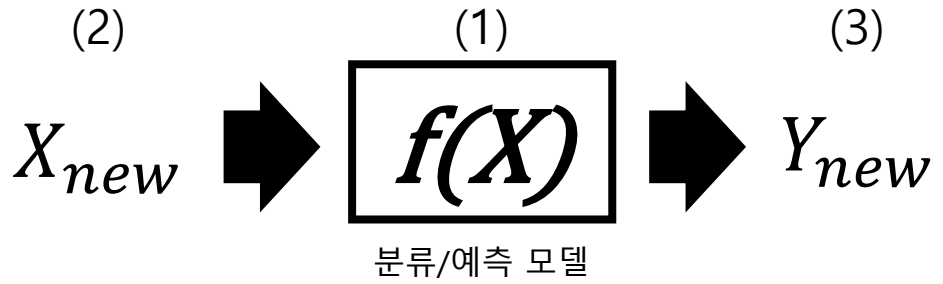
- Instance-Based Learning
  - K-Nearest Neighbor (KNN)
  - Locally weighted regression

  → 별도의 모델 생성 없이 인접 데이터를 분류/예측에 사용

Model for Classification and Prediction

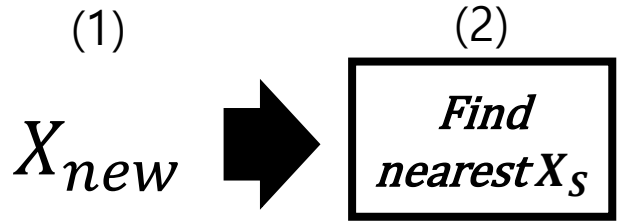
- Model-Based Learning
  - 선형/비선형모델 (e.g., linear regression, logistic regression)
  - Neural Network
  - Decision Tree
  - Support Vector Machine

  → 데이터로부터 모델을 생성하여 분류/예측 진행



- Instance-Based Learning
  - K-Nearest Neighbor (KNN)
  - Locally weighted regression

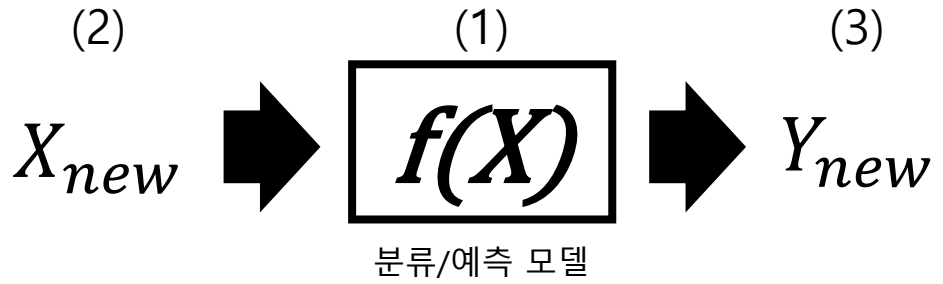
  → 별도의 모델 생성 없이 인접 데이터를 분류/예측에 사용



Model for Classification and Prediction

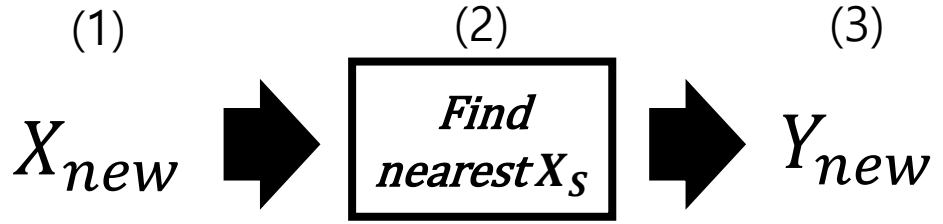
- Model-Based Learning
  - 선형/비선형모델 (e.g., linear regression, logistic regression)
  - Neural Network
  - Decision Tree
  - Support Vector Machine

  → 데이터로부터 모델을 생성하여 분류/예측 진행



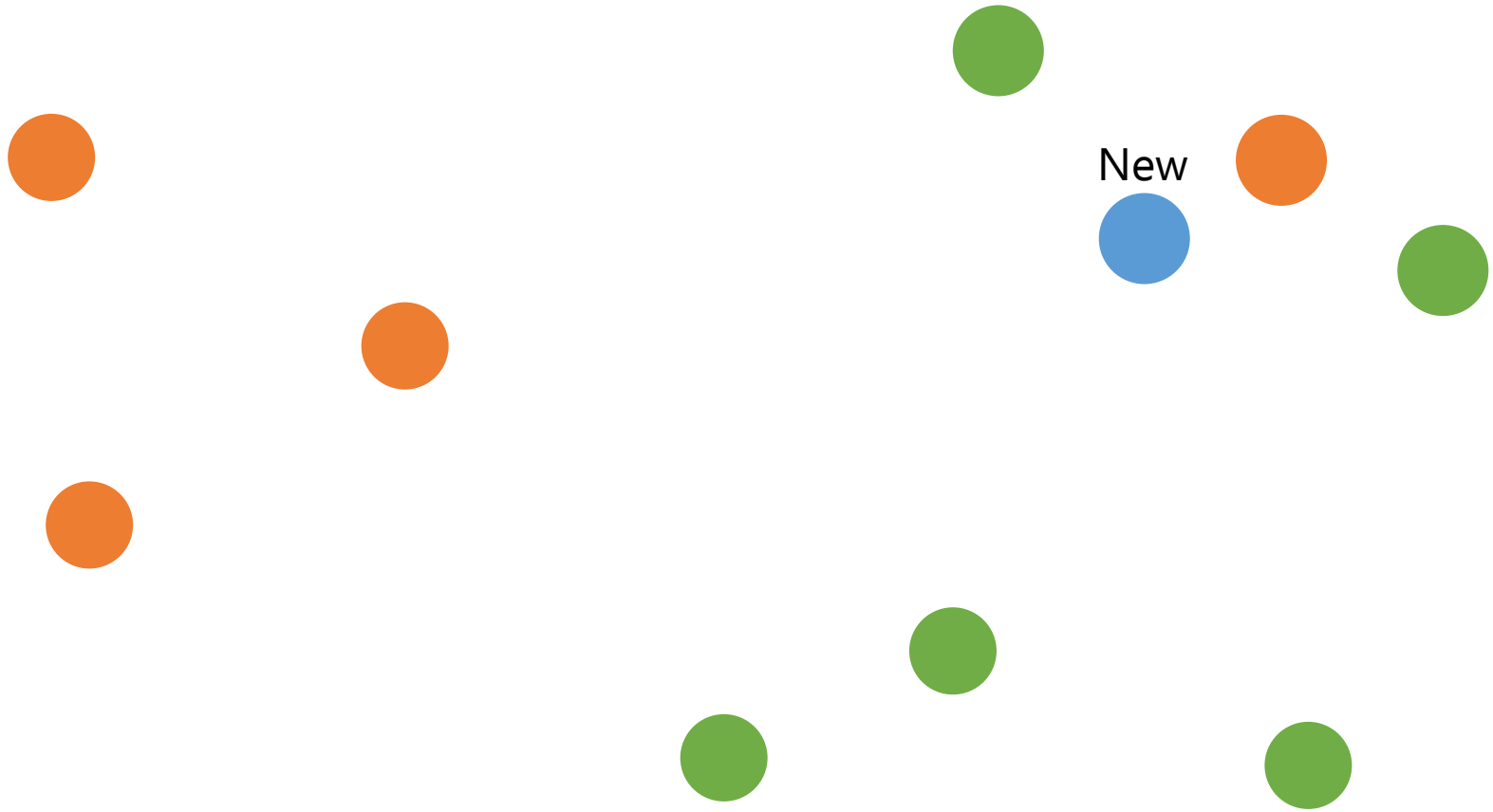
- Instance-Based Learning
  - K-Nearest Neighbor (KNN)
  - Locally weighted regression

  → 별도의 모델 생성 없이 인접 데이터를 분류/예측에 사용



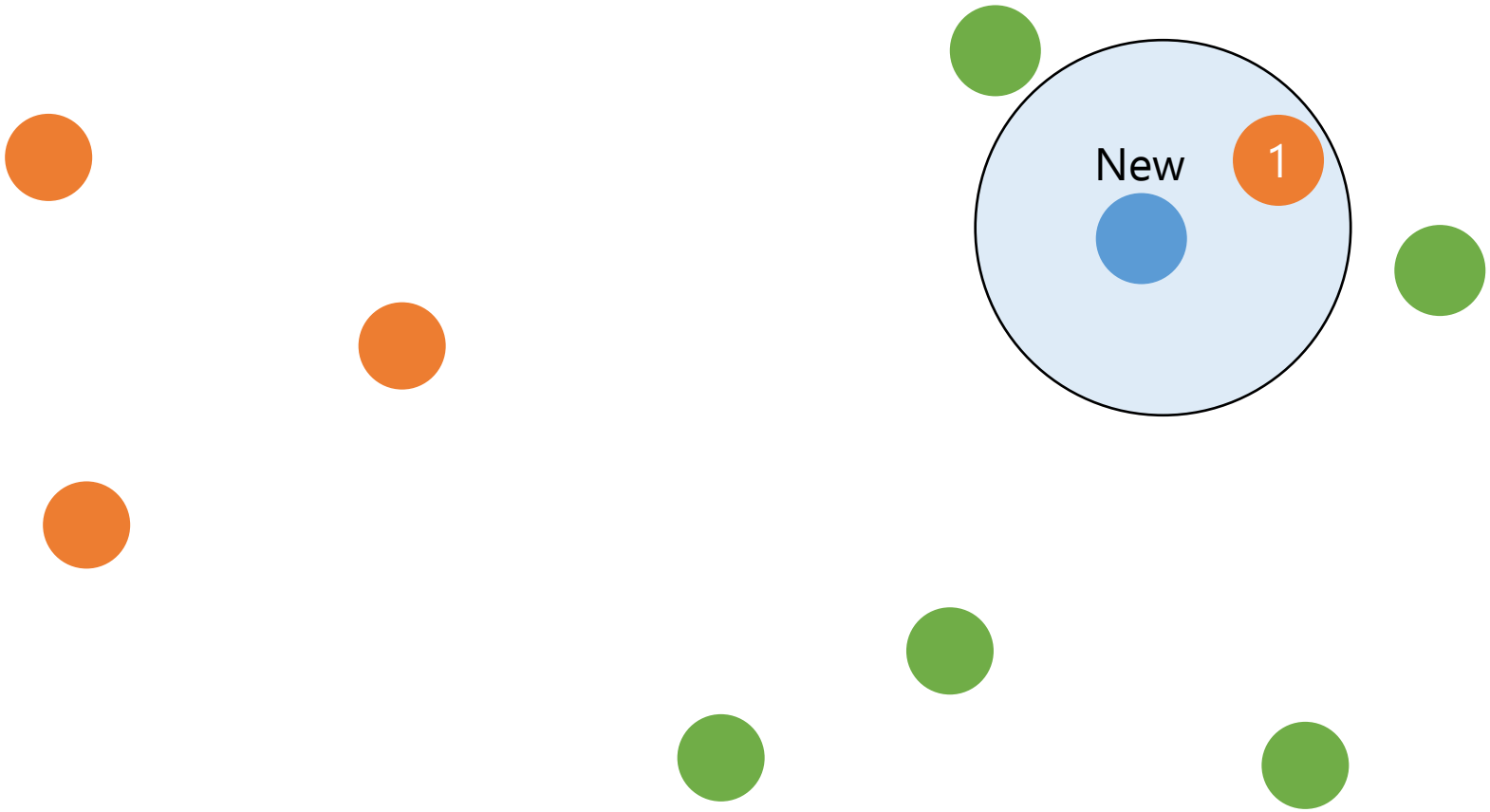
Nearest Neighbor

- 1-nearest neighbor



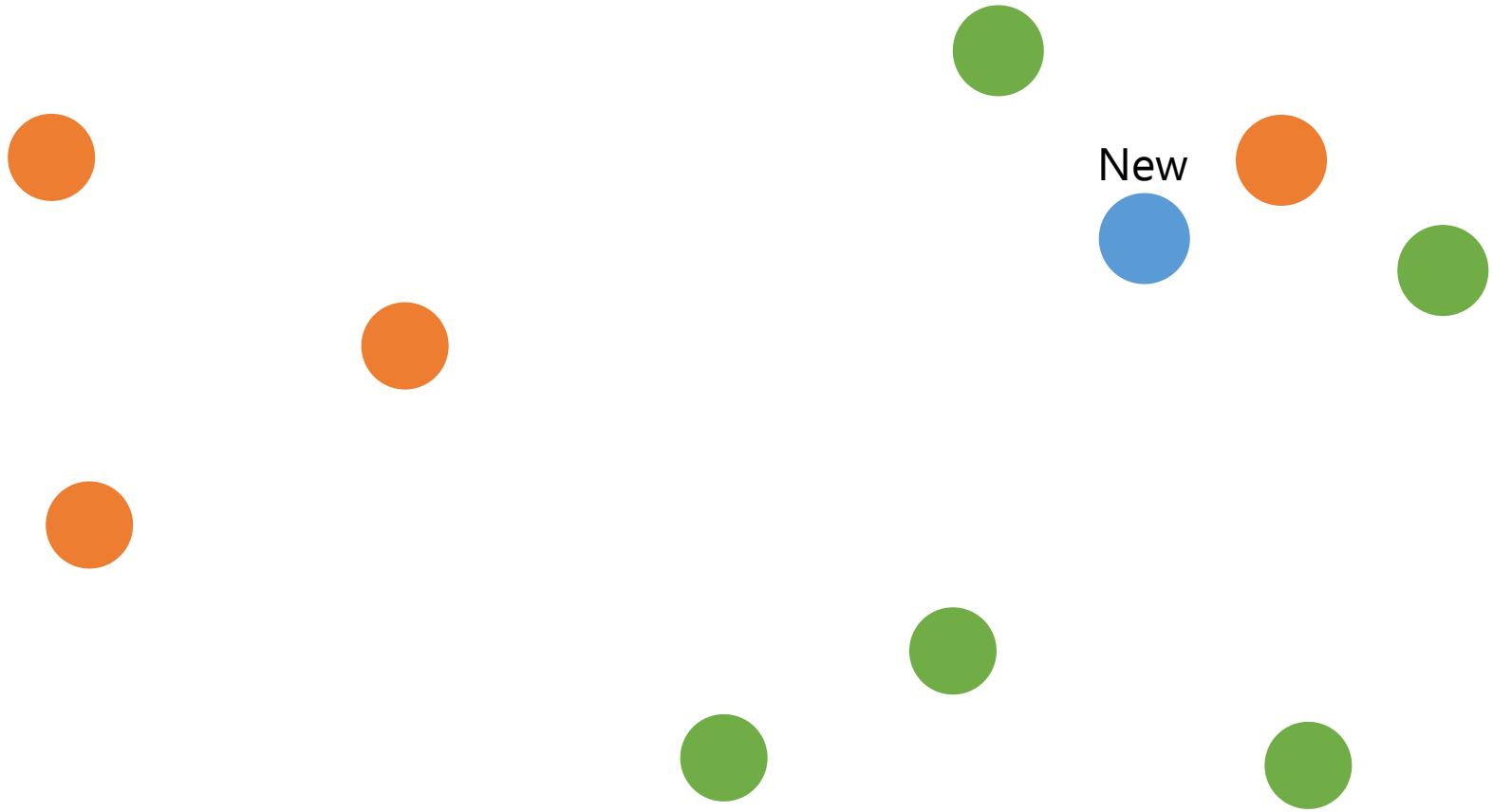
Nearest Neighbor

- 1-nearest neighbor



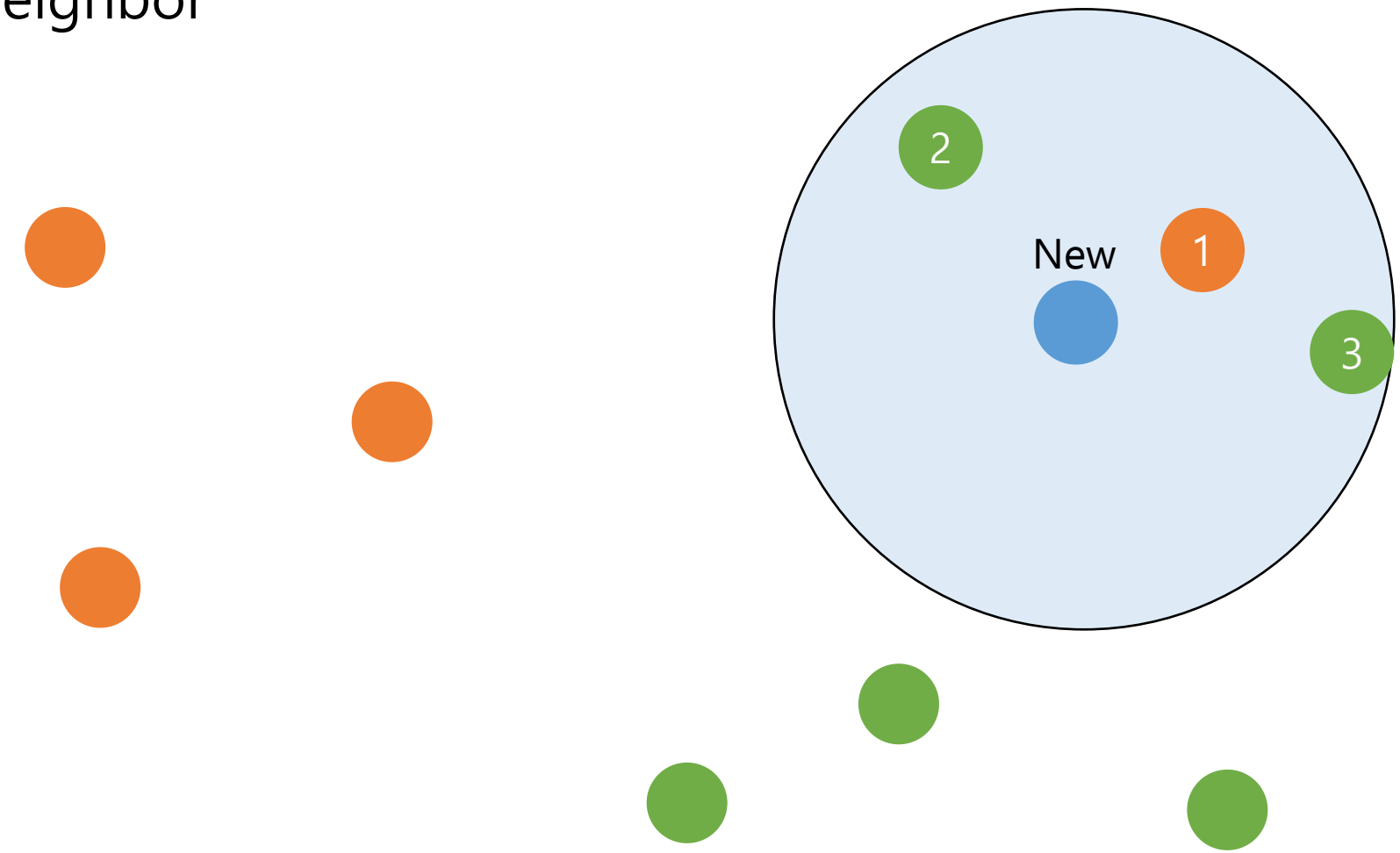
Nearest Neighbor

- 3-nearest neighbor



Nearest Neighbor

- 3-nearest neighbor



K-Nearest Neighbor (KNN) Algorithm

X1	X2	Class
3	3.5	1
4	5	1
2	4.5	1
1	1	2
3.5	7	2
6	5.5	2

K-Nearest Neighbor (KNN) Algorithm

X1	X2	Class
3	3.5	1
4	5	1
2	4.5	1
1	1	2
3.5	7	2
6	5.5	2

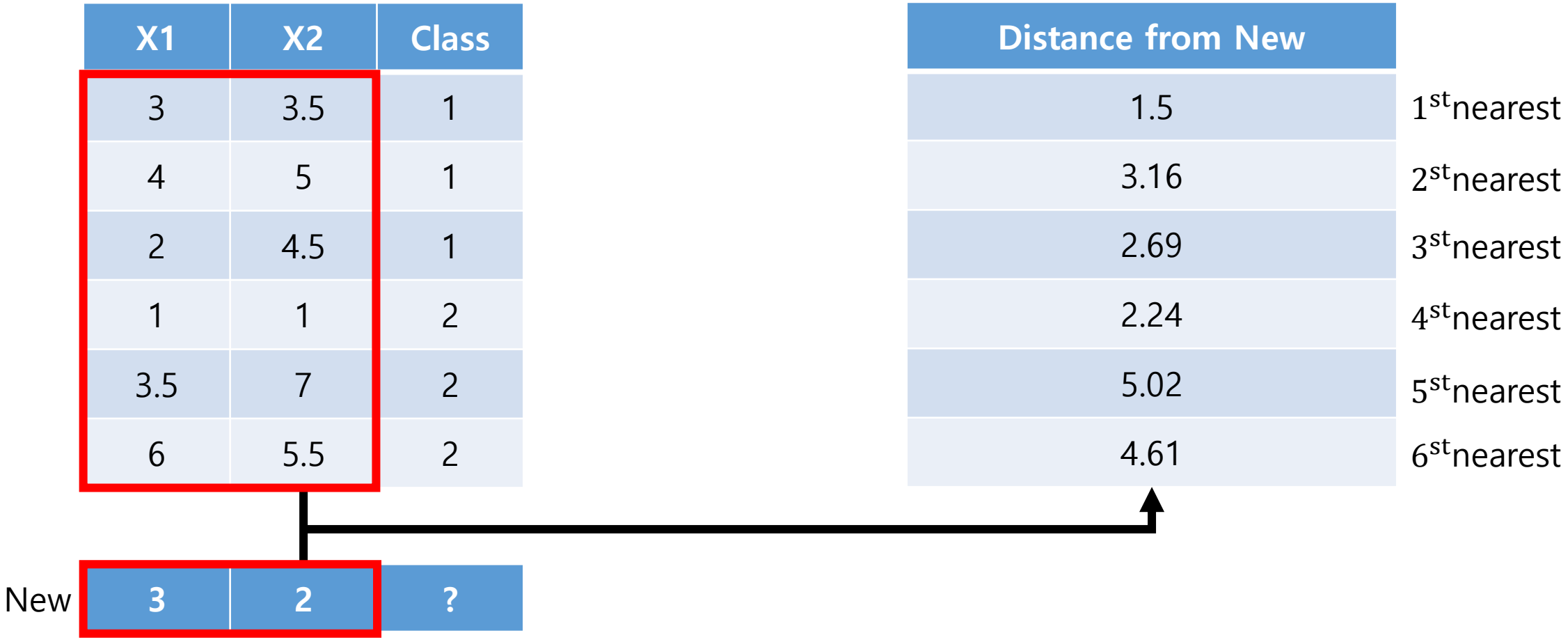
New	**3**	**2**	**?**

K-Nearest Neighbor (KNN) Algorithm

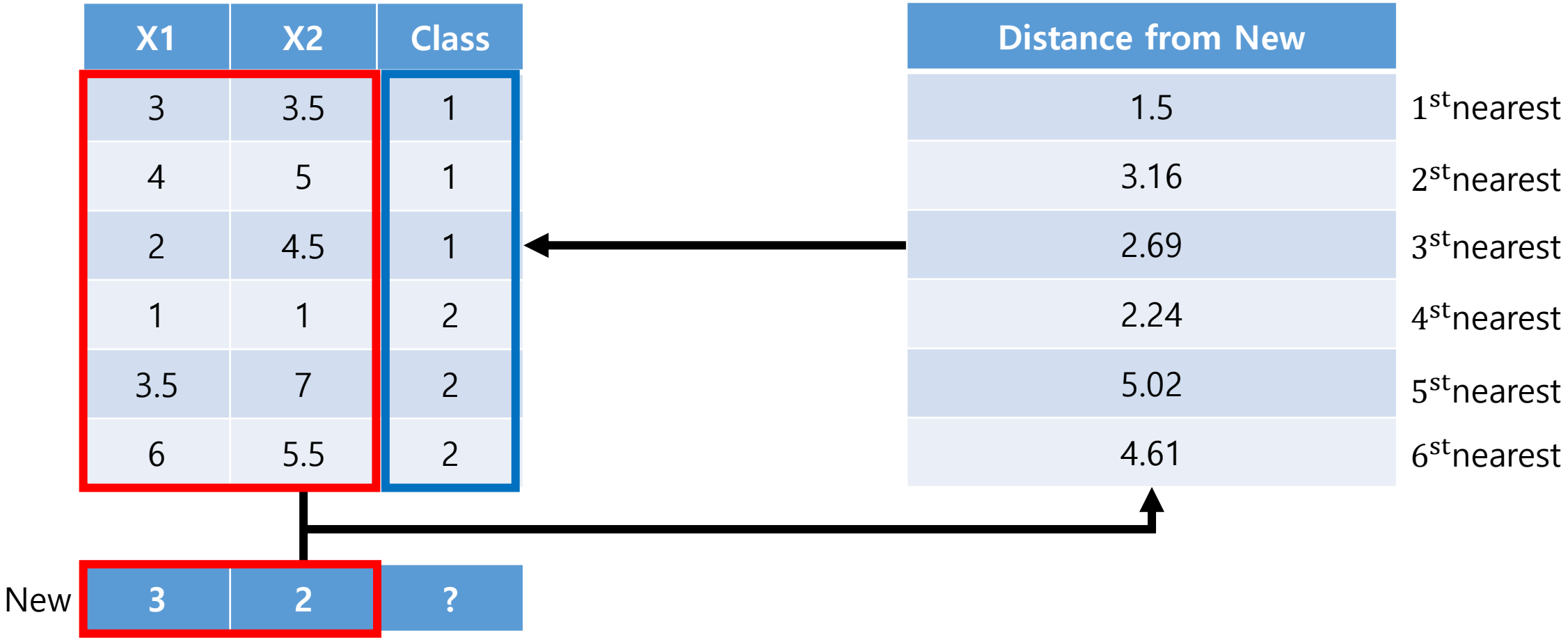
X1	X2	Class
3	3.5	1
4	5	1
2	4.5	1
1	1	2
3.5	7	2
6	5.5	2

New	**3**	**2**	**?**

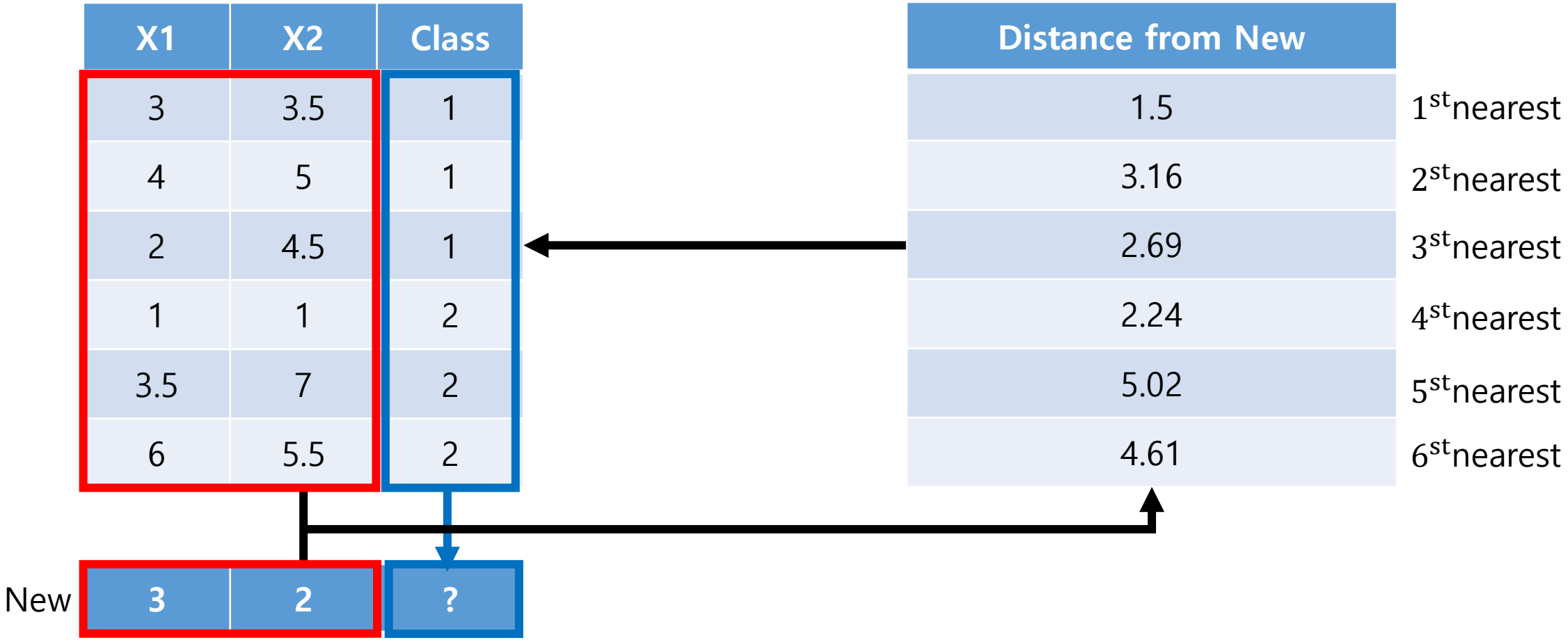
K-Nearest Neighbor (KNN) Algorithm



K-Nearest Neighbor (KNN) Algorithm



K-Nearest Neighbor (KNN) Algorithm

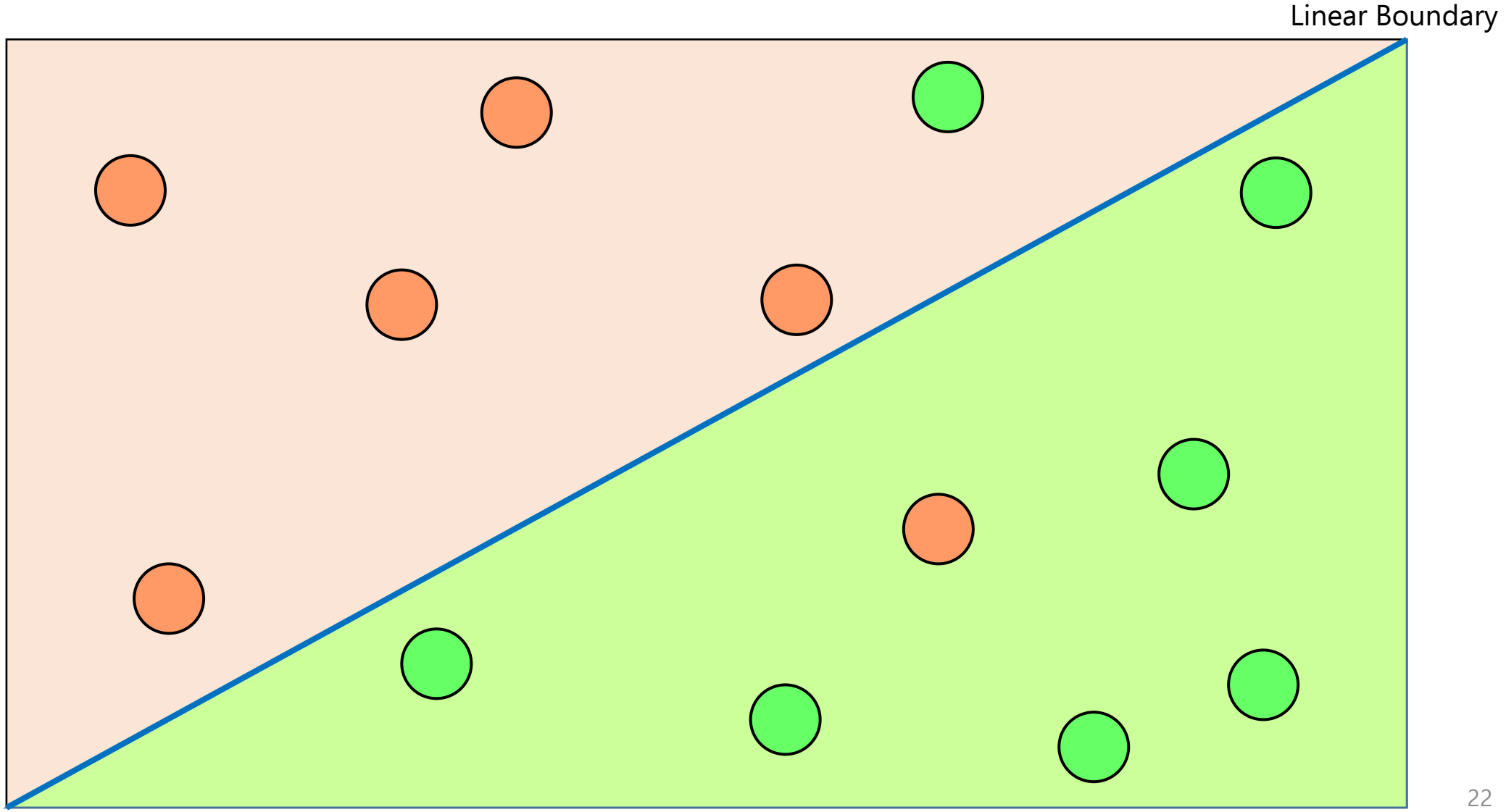


- 새로운 데이터가 발생한 이후 예측 수행

Property of KNN Algorithm

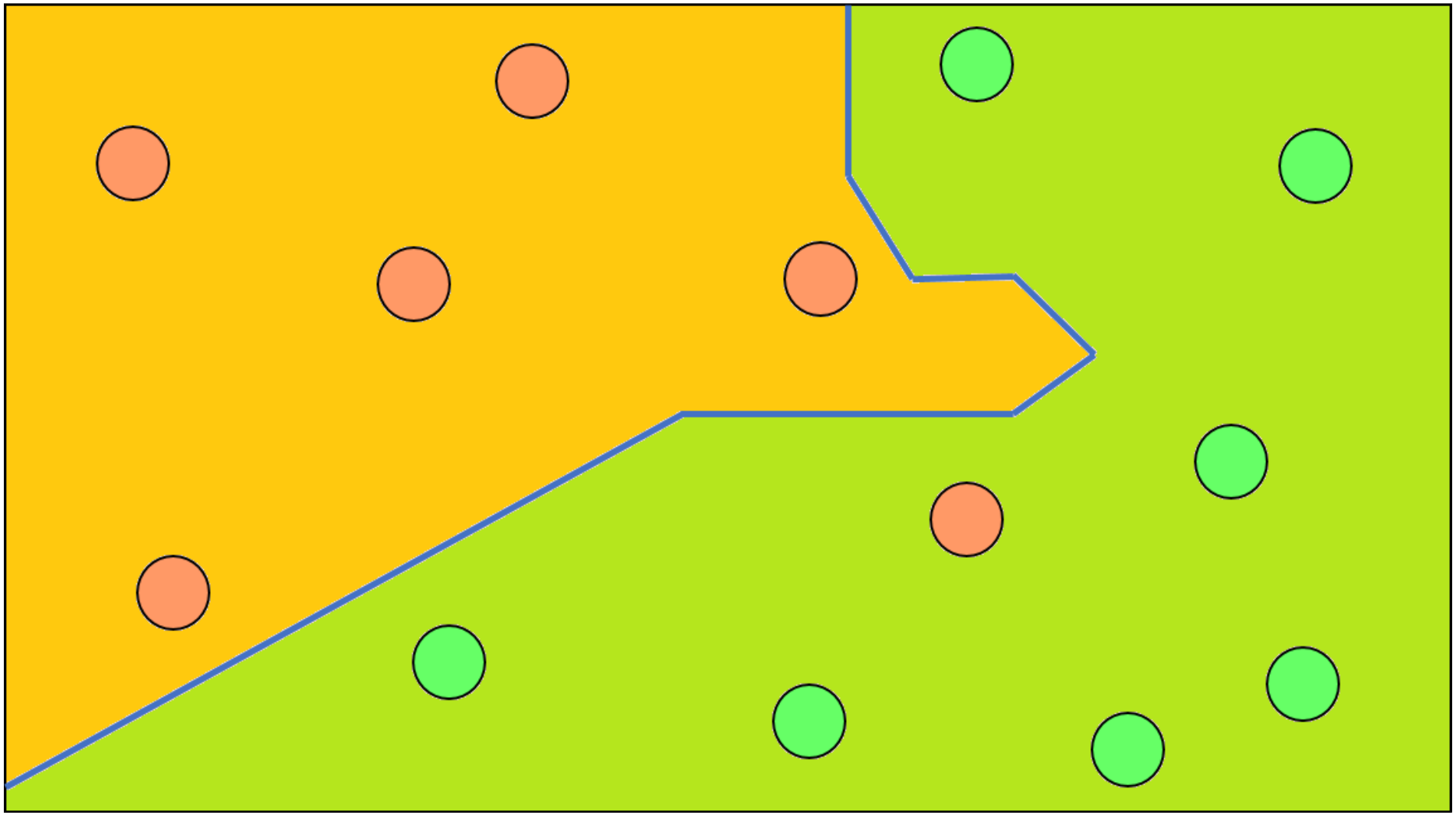
- Instance-Based Learning
  - 각각의 관측치(Instance) 만을 이용해 새로운 data에 대한 예측 진행
- Memory-Based Learning
  - 모든 학습 data를 메모리에 저장한 후, 이를 바탕으로 예측 시도
- Lazy Learning
  - 모델을 별도로 학습하지 않는 testing data가 들어와야 비로소 작동하는 게으른(lazy) 알고리즘

Comparison of Linear Regression and KNN

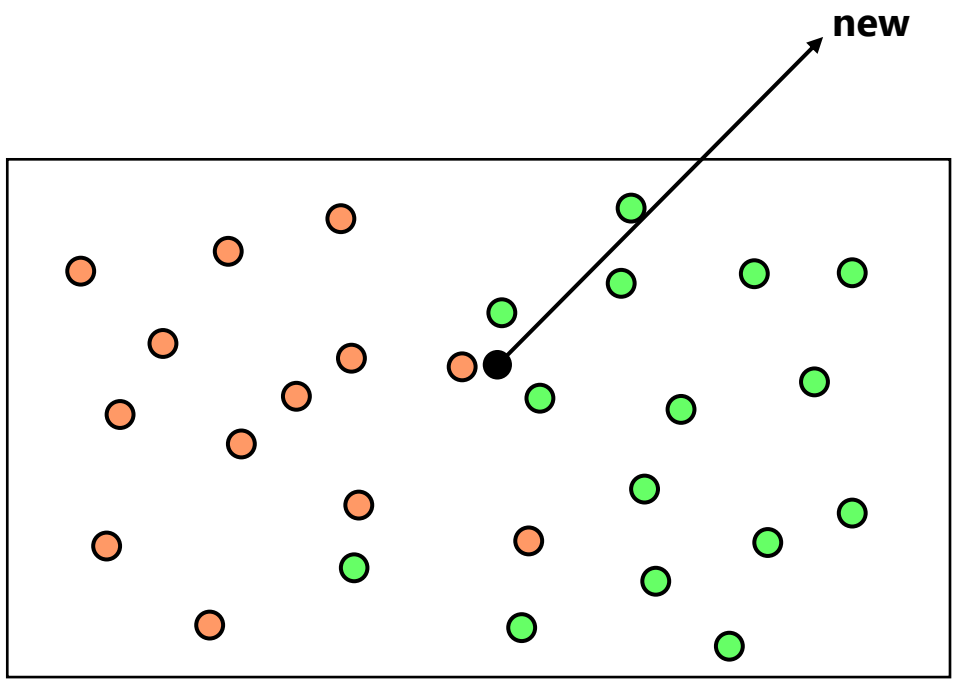


Comparison of Linear Regression and KNN

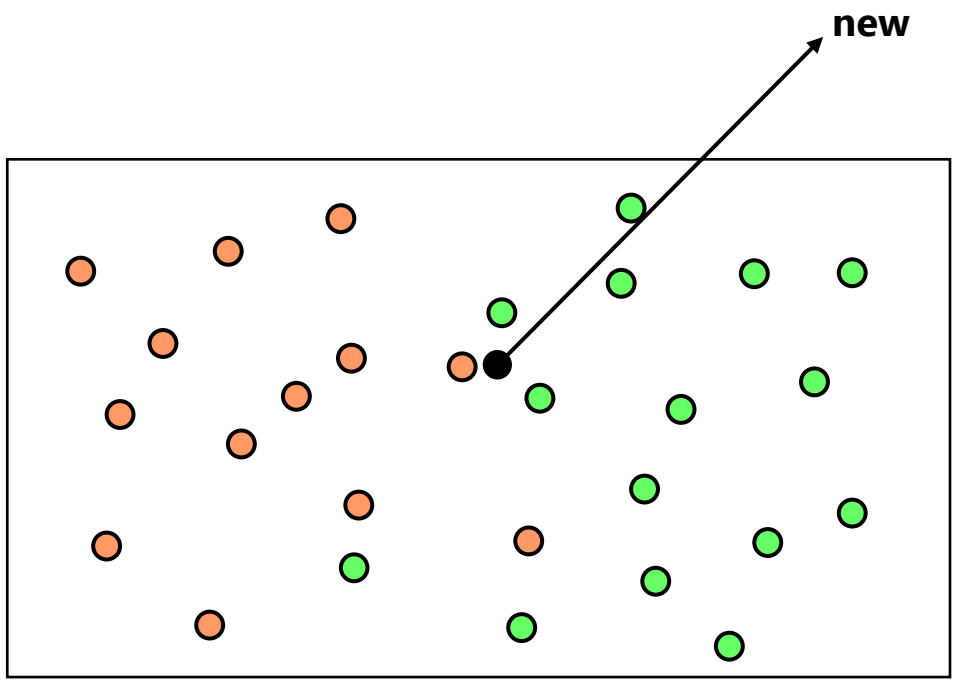
KNN (k=3) Boundary



KNN Classification Model



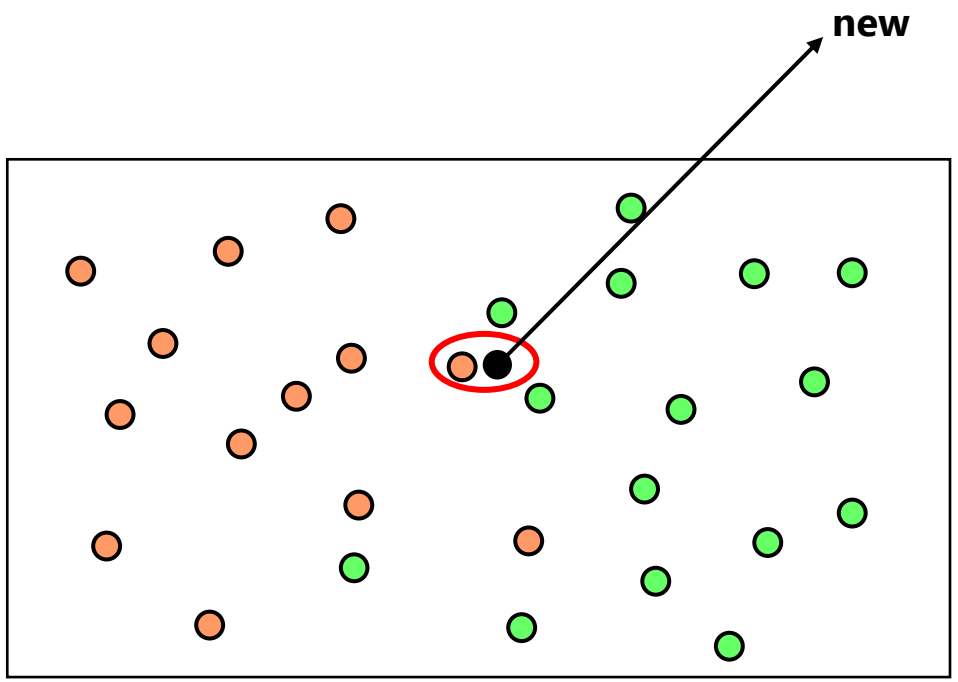
KNN Classification Model



K = # of nearest neighbors

K = 1 :

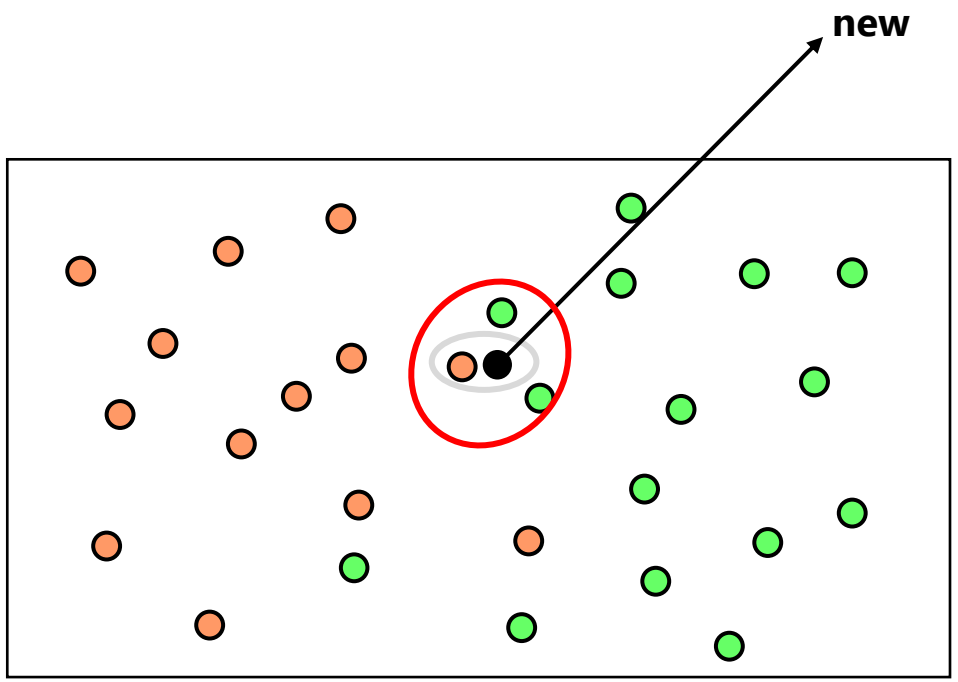
KNN Classification Model



K = # of nearest neighbors

K = 1 : **Orange**

KNN Classification Model

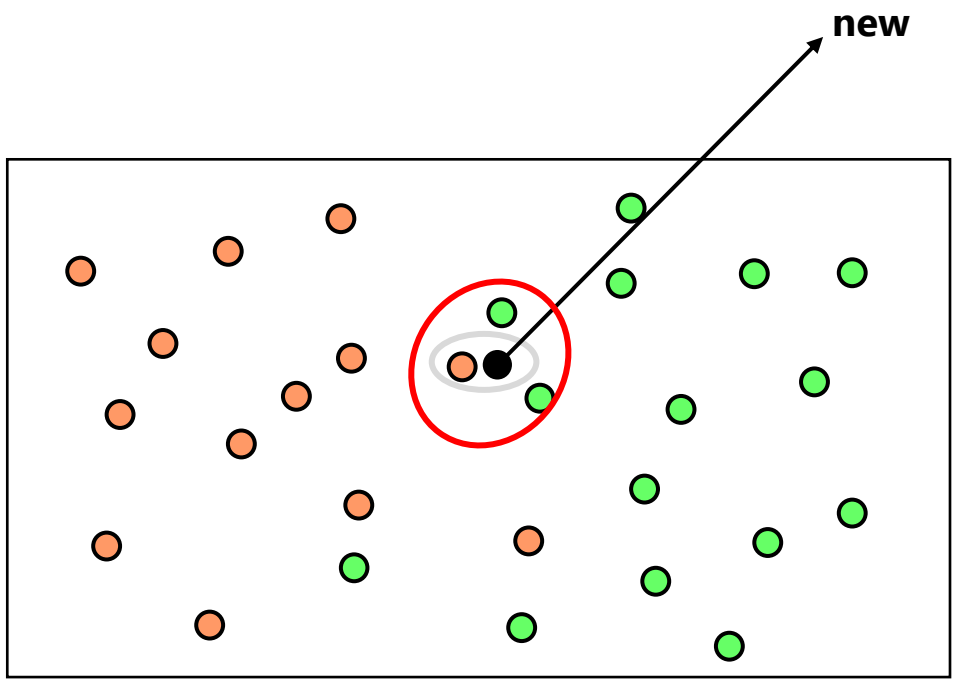


K = # of nearest neighbors

K = 1 : **Orange**

K = 3 :

KNN Classification Model

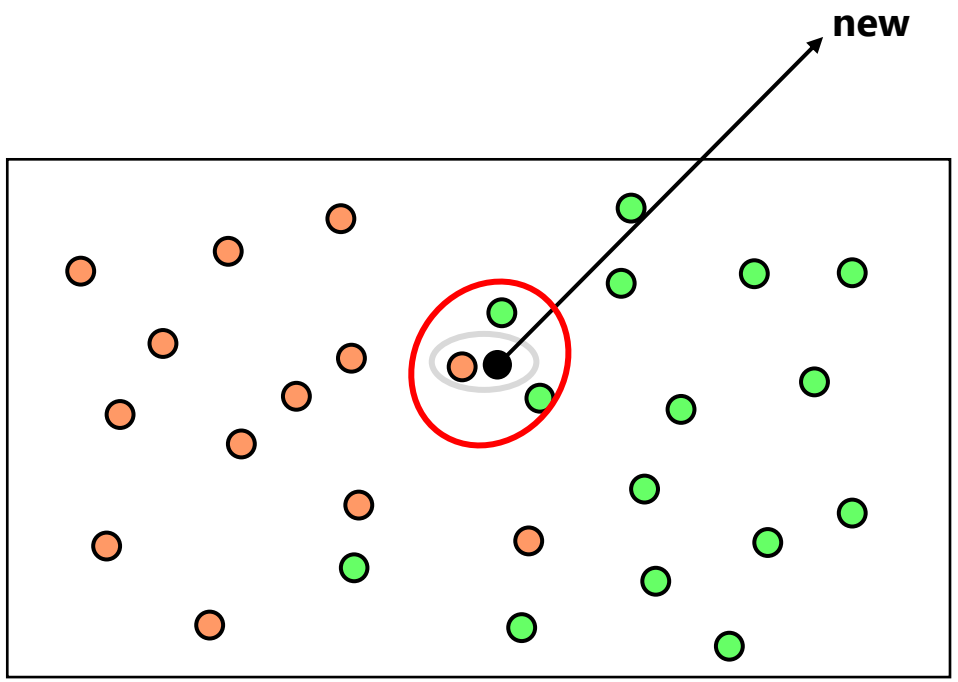


K = # of nearest neighbors

K = 1 : **Orange**

K = 3 : **Green**

KNN Classification Model



K = # of nearest neighbors

K = 1 : **Orange**

K = 3 : **Green**

- 인접한 k개의 데이터로부터 majority voting 시행

Example of KNN Classification Model

유전자 정보

사람	유전자 A	유전자 B	유전자 C	유전자 D
A	2.54	4.33	3.99	2.57
B	3.12	3.87	3.84	3.04
C	2.76	4.17	5.63	3.28
D	3.87	3.56	4.25	3.65
E	3.55	3.91	2.68	4.22
F	4.12	2.86	3.30	3.71

환자 상태

질병유뮤
정상
정상
정상
질병
질병
질병

Example of KNN Classification Model

유전자 정보

환자 상태

사람	유전자 A	유전자 B	유전자 C	유전자 D	질병유무
A	2.54	4.33	3.99	2.57	정상
B	3.12	3.87	3.84	3.04	정상
C	2.76	4.17	5.63	3.28	정상
D	3.87	3.56	4.25	3.65	질병
E	3.55	3.91	2.68	4.22	질병
F	4.12	2.86	3.30	3.71	질병
G	**3.24**	**3.68**	**3.82**	**3.77**	**?**

Example of KNN Classification Model

유전자 정보

환자 상태

사람	유전자 A	유전자 B	유전자 C	유전자 D	질병유무	새로운 관측치와의 거리
A	2.54	4.33	3.99	2.57	정상	1.54
B	3.12	3.87	3.84	3.04	정상	0.76
C	2.76	4.17	5.63	3.28	정상	2.00
D	3.87	3.56	4.25	3.65	질병	0.78
E	3.55	3.91	2.68	4.22	질병	1.28
F	4.12	2.86	3.30	3.71	질병	1.31
G	**3.24**	**3.68**	**3.82**	**3.77**	**?**	

Example of KNN Classification Model

유전자 정보

사람	유전자 A	유전자 B	유전자 C	유전자 D
A	2.54	4.33	3.99	2.57
B	3.12	3.87	3.84	3.04
C	2.76	4.17	5.63	3.28
D	3.87	3.56	4.25	3.65
E	3.55	3.91	2.68	4.22
F	4.12	2.86	3.30	3.71
G	**3.24**	**3.68**	**3.82**	**3.77**

환자 상태

질병유뮤	새로운 관측치와의 거리
정상	1.54
정상	0.76
정상	2.00
질병	0.78
질병	1.28
질병	1.31
?	

K = 1 :

Example of KNN Classification Model

유전자 정보

환자 상태

사람	유전자 A	유전자 B	유전자 C	유전자 D	질병유무	새로운 관측치와의 거리
A	2.54	4.33	3.99	2.57	정상	1.54
B	3.12	3.87	3.84	3.04	정상	0.76
C	2.76	4.17	5.63	3.28	정상	2.00
D	3.87	3.56	4.25	3.65	질병	0.78
E	3.55	3.91	2.68	4.22	질병	1.28
F	4.12	2.86	3.30	3.71	질병	1.31
G	**3.24**	**3.68**	**3.82**	**3.77**	**정상**	

K = 1 : 정상

Example of KNN Classification Model

유전자 정보

환자 상태

사람	유전자 A	유전자 B	유전자 C	유전자 D	질병유무	새로운 관측치와의 거리
A	2.54	4.33	3.99	2.57	정상	1.54
B	3.12	3.87	3.84	3.04	정상	0.76
C	2.76	4.17	5.63	3.28	정상	2.00
D	3.87	3.56	4.25	3.65	질병	0.78
E	3.55	3.91	2.68	4.22	질병	1.28
F	4.12	2.86	3.30	3.71	질병	1.31
G	**3.24**	**3.68**	**3.82**	**3.77**	**정상**	

K = 1 : 정상
K = 3 :

Example of KNN Classification Model

유전자 정보

환자 상태

사람	유전자 A	유전자 B	유전자 C	유전자 D	질병유무	새로운 관측치와의 거리
A	2.54	4.33	3.99	2.57	정상	1.54
B	3.12	3.87	3.84	3.04	정상	0.76
C	2.76	4.17	5.63	3.28	정상	2.00
D	3.87	3.56	4.25	3.65	질병	0.78
E	3.55	3.91	2.68	4.22	질병	1.28
F	4.12	2.86	3.30	3.71	질병	1.31
G	**3.24**	**3.68**	**3.82**	**3.77**	**질병**	

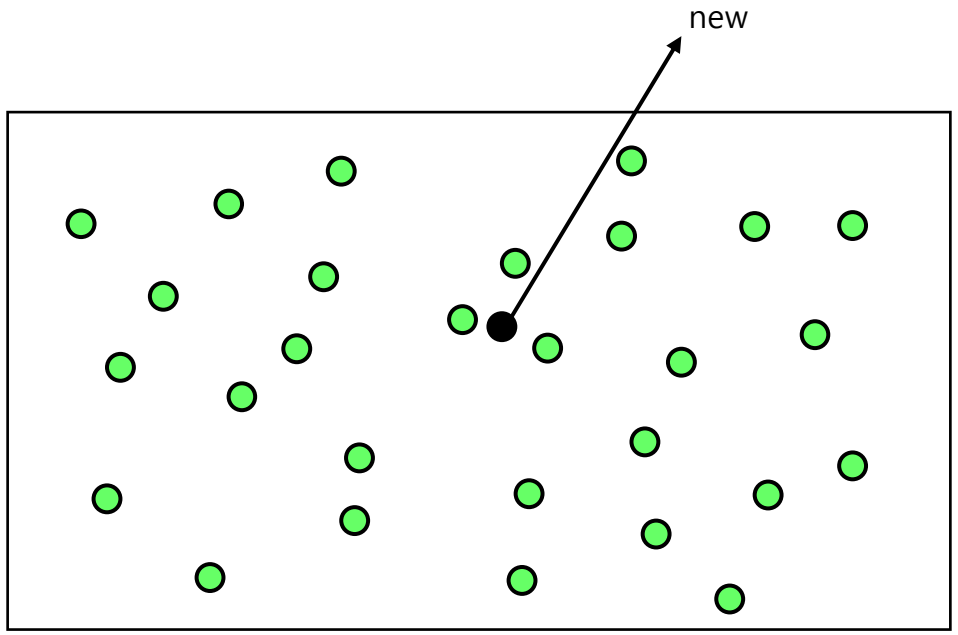
K = 1 : 정상
K = 3 : 질병

KNN Classification Algorithm

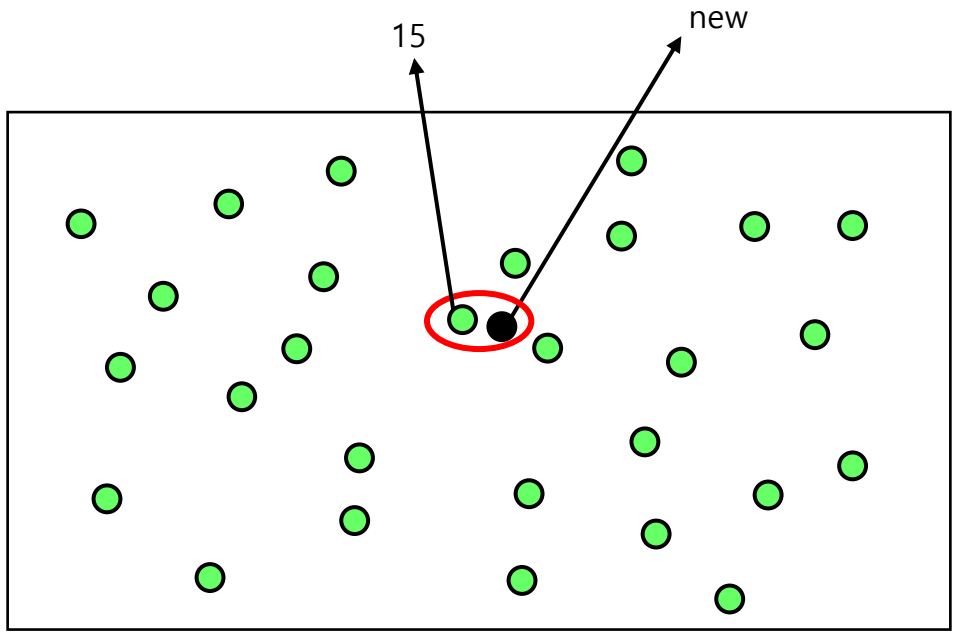
- 분류 알고리즘

1. 분류할 관측치 x를 선택
2. X로부터 인접한 k개의 학습 데이터를 탐색
3. 탐색된 k개 학습 데이터의 majority class c를 정의
4. c를 x의 분류 결과로 반환

KNN Prediction Model



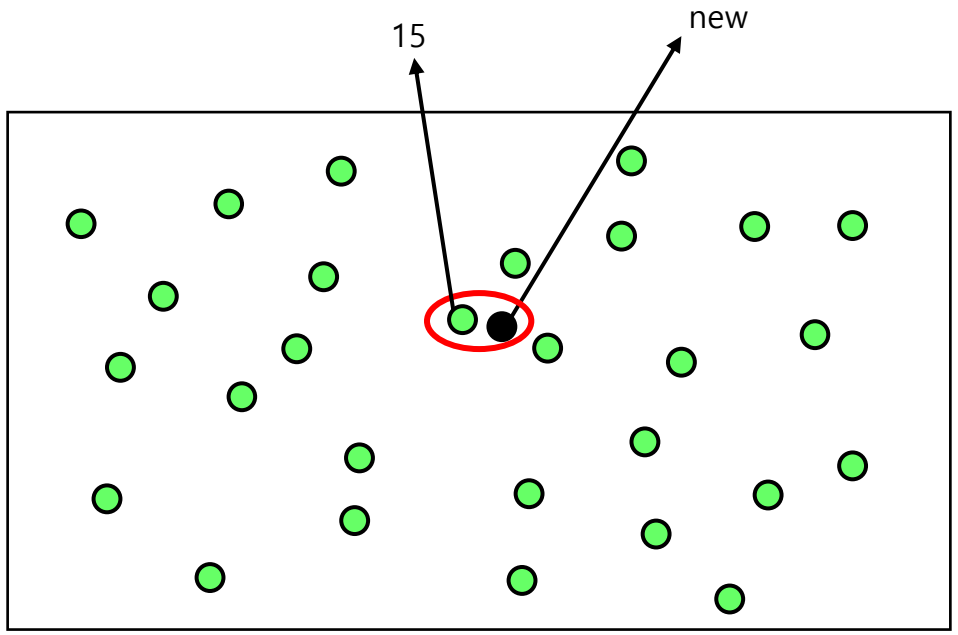
KNN Prediction Model



k = number of nearest neighbors

k = 1 :

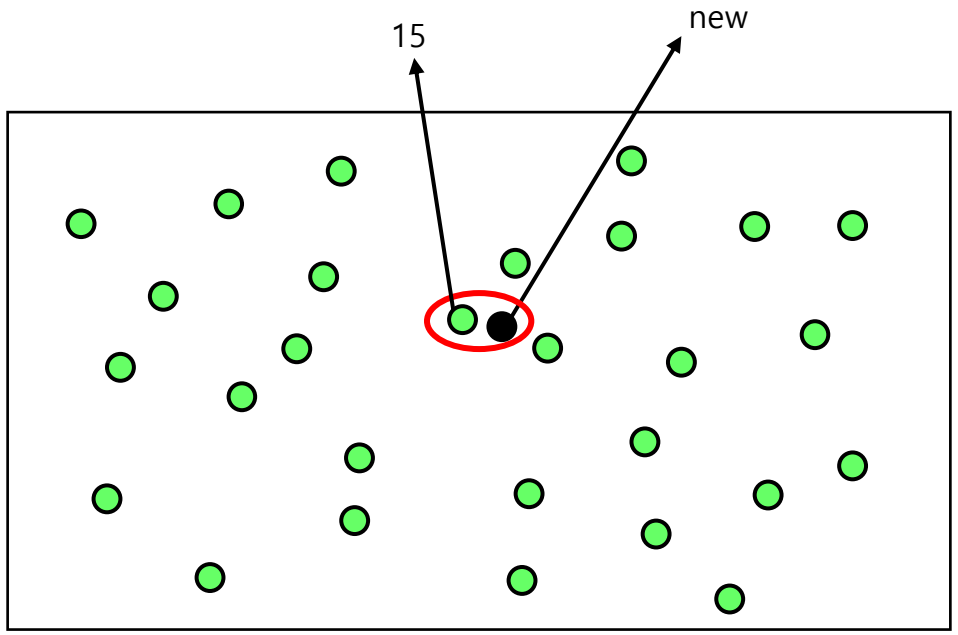
KNN Prediction Model



k = number of nearest neighbors

k = 1 : new = 15

KNN Prediction Model

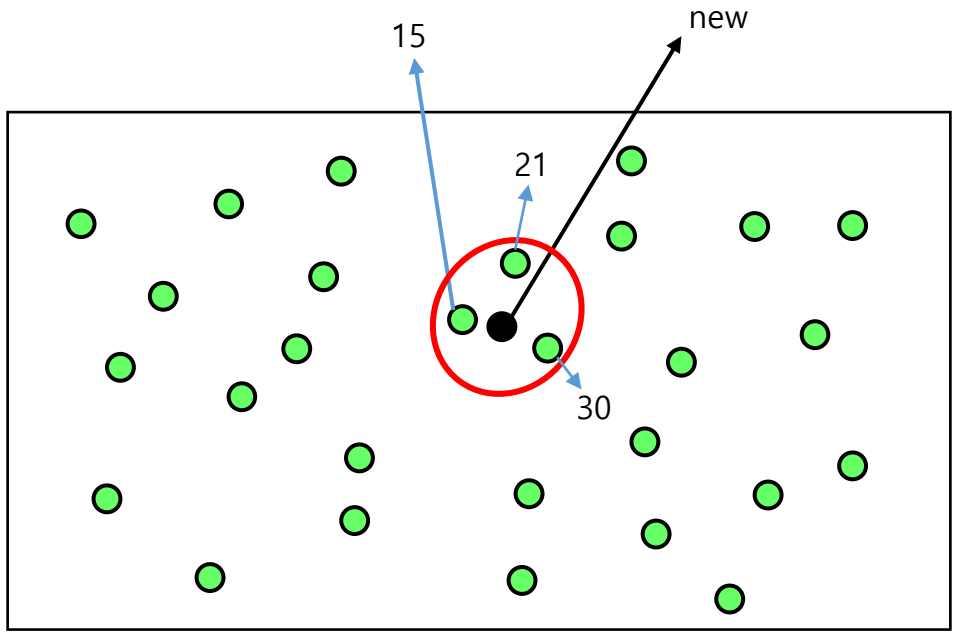


k = number of nearest neighbors

k = 1 : new = 15

k = 3 :

KNN Prediction Model



k = number of nearest neighbors

k = 1 : new = 15

k = 3 : new = (15+21+30)/3 = 22

Example of KNN Prediction Model

	기존 영화 평점					영화 평점
사람	미녀와 야수	그린북	라라랜드	극한직업	명량	항거
A	7.5	7.5	7.0	9.5	8.5	5.0
B	7.5	7,0	7,5	8.0	8.0	6.0
C	8.0	7.0	8.0	8.0	8.5	8.5
D	8.5	8.0	9.5	7.5	6.0	7.0
E	10.0	9.5	9.0	7.5	7.5	10.0
F	9.0	9.0	8.0	8.0	8.0	9.0

Example of KNN Prediction Model

	기존 영화 평점					영화 평점
사람	미녀와 야수	그린북	라라랜드	극한직업	명량	항거
A	7.5	7.5	7.0	9.5	8.5	5.0
B	7.5	7,0	7,5	8.0	8.0	6.0
C	8.0	7.0	8.0	8.0	8.5	8.5
D	8.5	8.0	9.5	7.5	6.0	7.0
E	10.0	9.5	9.0	7.5	7.5	10.0
F	9.0	9.0	8.0	8.0	8.0	9.0
G	**9.0**	**8.5**	**8.0**	**7.0**	**8.0**	**?**

Example of KNN Prediction Model

	기존 영화 평점					영화 평점	
사람	미녀와 야수	그린북	라라랜드	극한직업	명량	항거	새로운 관측치와의 거리
A	7.5	7.5	7.0	9.5	8.5	5.0	3.28
B	7.5	7,0	7,5	8.0	8.0	6.0	2.40
C	8.0	7.0	8.0	8.0	8.5	8.5	2.12
D	8.5	8.0	9.5	7.5	6.0	7.0	2.65
E	10.0	9.5	9.0	7.5	7.5	10.0	1.87
F	9.0	9.0	8.0	8.0	8.0	9.0	1.12
G	**9.0**	**8.5**	**8.0**	**7.0**	**8.0**	**?**	

Example of KNN Prediction Model

기존 영화 평점

사람	미녀와 야수	그린북	라라랜드	극한직업	명량
A	7.5	7.5	7.0	9.5	8.5
B	7.5	7,0	7,5	8.0	8.0
C	8.0	7.0	8.0	8.0	8.5
D	8.5	8.0	9.5	7.5	6.0
E	10.0	9.5	9.0	7.5	7.5
F	9.0	9.0	8.0	8.0	8.0
G	**9.0**	**8.5**	**8.0**	**7.0**	**8.0**

영화 평점

항거	새로운 관측치와의 거리
5.0	3.28
6.0	2.40
8.5	2.12
7.0	2.65
10.0	1.87
9.0	1.12
9.0	

K = 1 : 9.0

Example of KNN Prediction Model

기존 영화 평점

사람	미녀와 야수	그린북	라라랜드	극한직업	명량
A	7.5	7.5	7.0	9.5	8.5
B	7.5	7,0	7,5	8.0	8.0
C	8.0	7.0	8.0	8.0	8.5
D	8.5	8.0	9.5	7.5	6.0
E	10.0	9.5	9.0	7.5	7.5
F	9.0	9.0	8.0	8.0	8.0
G	**9.0**	**8.5**	**8.0**	**7.0**	**8.0**

영화 평점

항거	새로운 관측치와의 거리
5.0	3.28
6.0	2.40
8.5	2.12
7.0	2.65
10.0	1.87
9.0	1.12
9.17	

K = 1 : 9.0
K = 3 : (9.0+10.0+8.5)/3 = 9.17

KNN Prediction Algorithm

- 예측 알고리즘

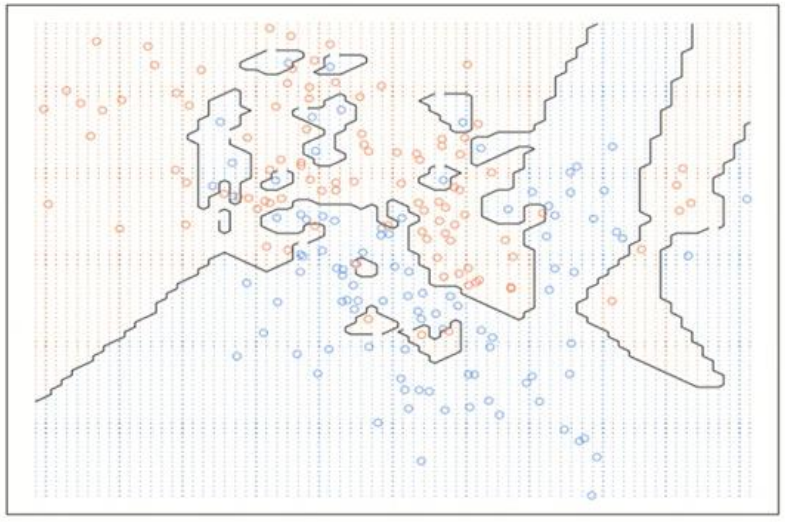
1. 예측할 관측치 x를 선택
2. X로부터 인접한 k개의 학습 데이터를 탐색
3. 탐색된 k개 학습 데이터의 평균을 x의 예측 값으로 반환

Hyperparameter of KNN

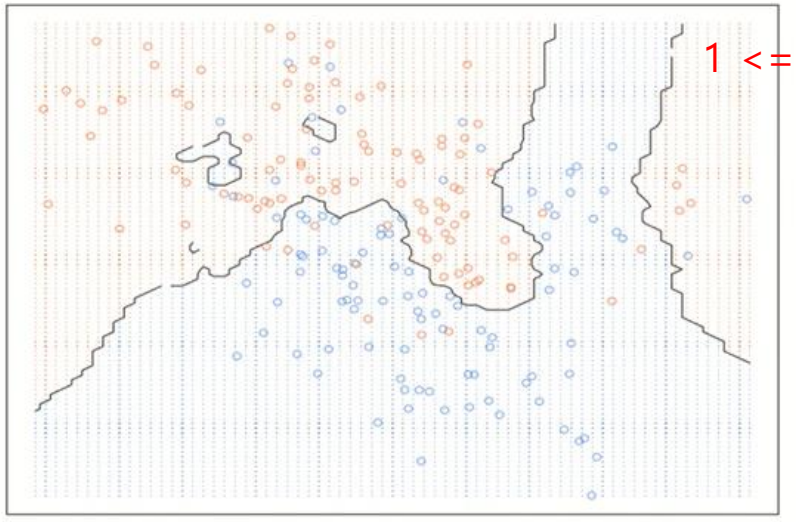
1. K
   - 인접한 학습 데이터를 몇 개까지 탐색할 것인가?

2. Distance Measures
   - 데이터 간 거리는 어떻게 측정할 것인가?

The result according to K

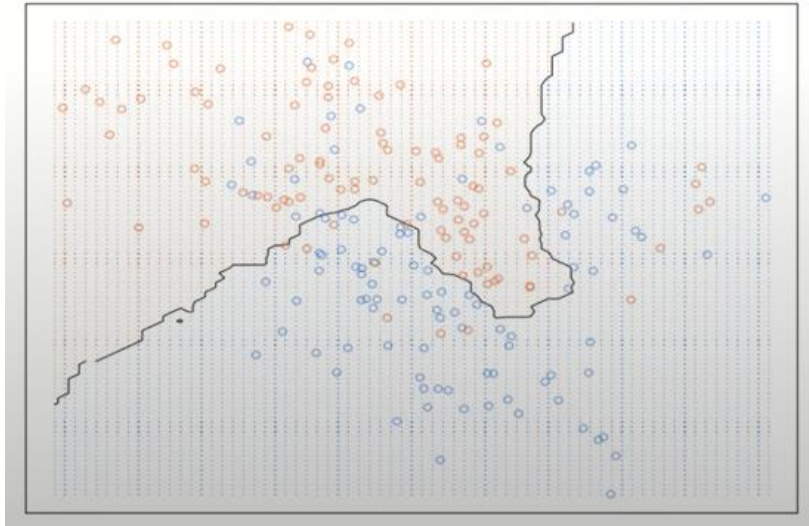


1-nearest neighbor

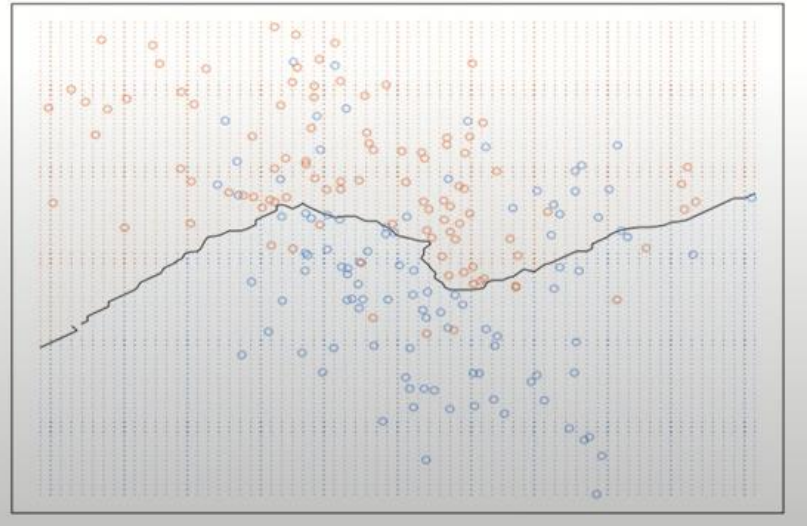


5-nearest neighbor

1 <= k <= 데이터의 개수



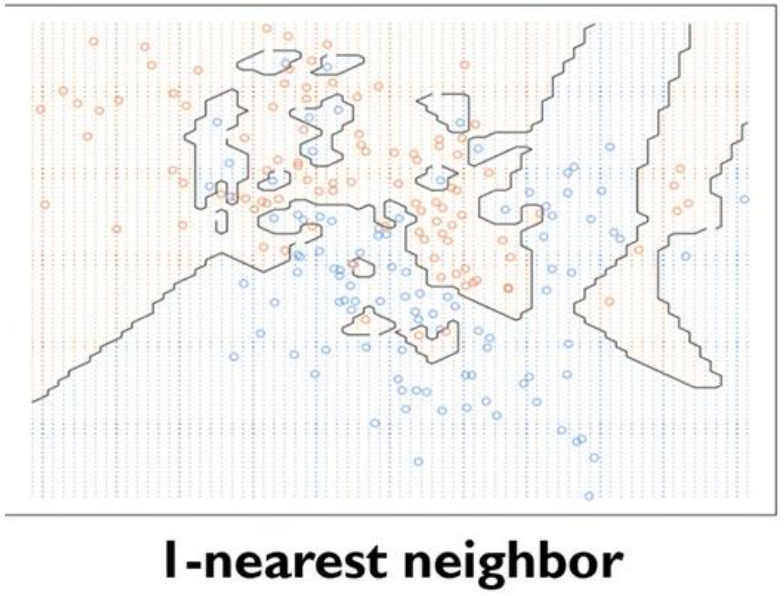
15-nearest neighbor



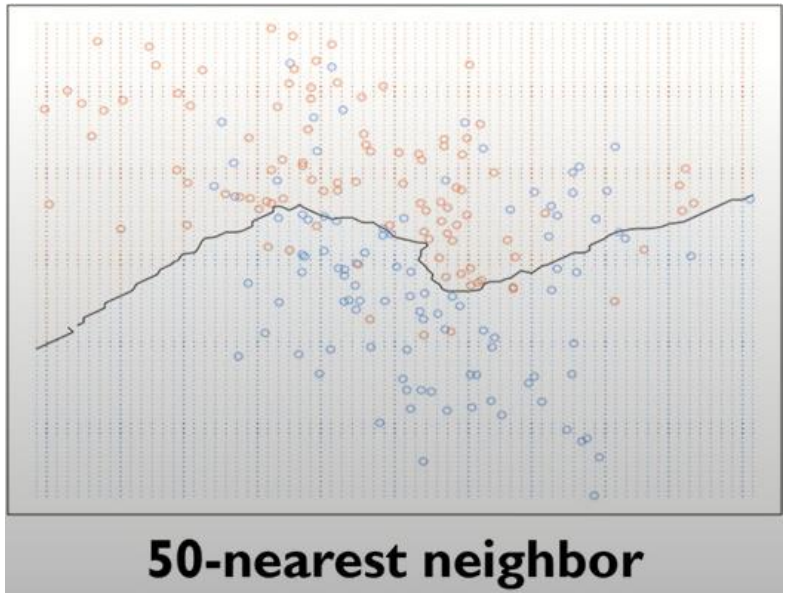
50-nearest neighbor

The result according to K

- K가 매우 작을 경우 : 데이터의 지역적 특성을 지나치게 반영함 (overfitting)
- K가 매우 클 경우 : 다른 범주의 개체를 너무 많이 포함하여 오분류할 위험 (underfitting)



1-nearest neighbor



50-nearest neighbor

How to select K

- 일정 범위 내로 k를 조정하여 (1~k*), 가장 좋은 예측 결과를 보이는 k값을 선정

- 분류 모델

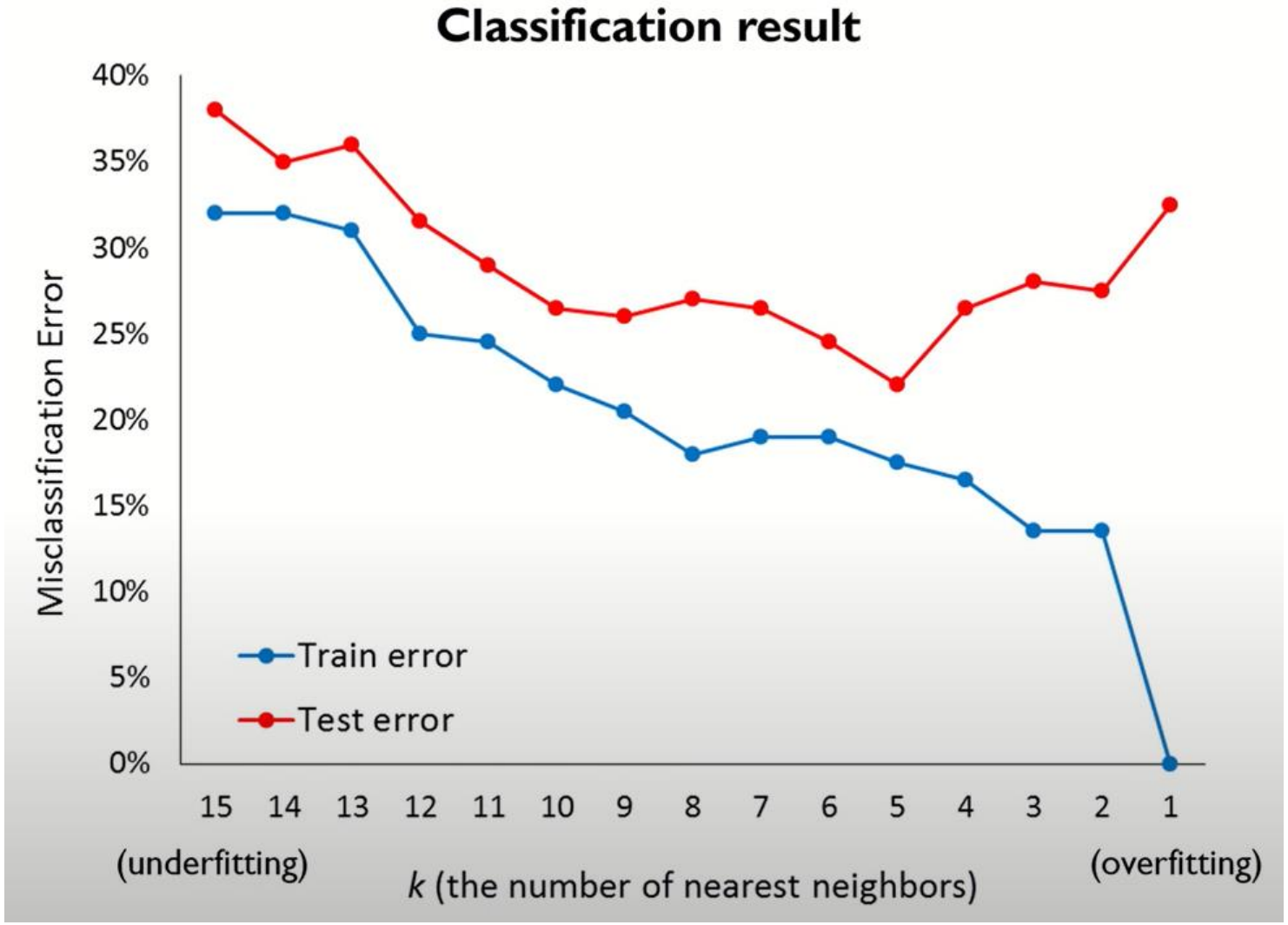
$$\text{MisclassError}_k = \frac{1}{k}\sum_{i=1}^{k} I(c_i \neq \hat{c}_i) \text{for } k = 1,2,\ldots,k*$$

$I(\,)$: Indicator Function

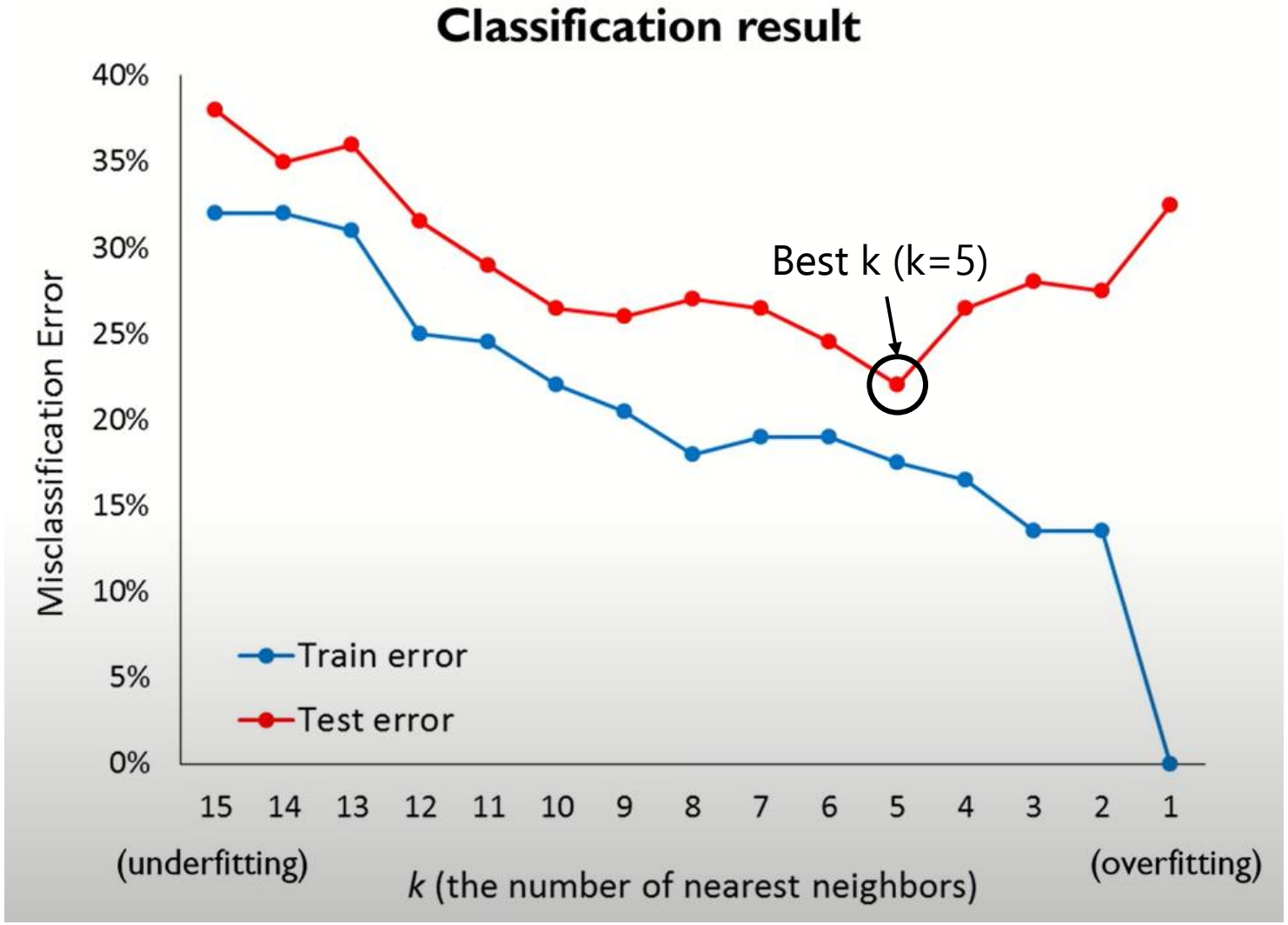
- 예측 모델

$$\text{SSE}_k = \sum_{i=1}^{k} (y_i - \hat{y}_i)^2 \text{ for } k = 1,2,\ldots,k*$$

How to select K



How to select K



Distance measurement (1-similarity)

- 다양한 거리측도 (Distance measure) 존재
  (e.g., Euclidean Distance, Correlation Distance, …)

- 데이터 내 변수들이 각기 다른 데이터 범위, 분산 등을 가질 수 있으므로, 데이터 정규화 (혹은 표준화)를 통해 이를 맞추는 것이 중요
  - 거리를 계산할 때, 단위가 큰 특정 변수(들)가 거리를 결정하는 것 방지
  - ex) 키(1.5m~1.8m), 몸무게(90lb~300lb), 연봉(20,000,000원~100,000,000원)

Types of Distance measurement

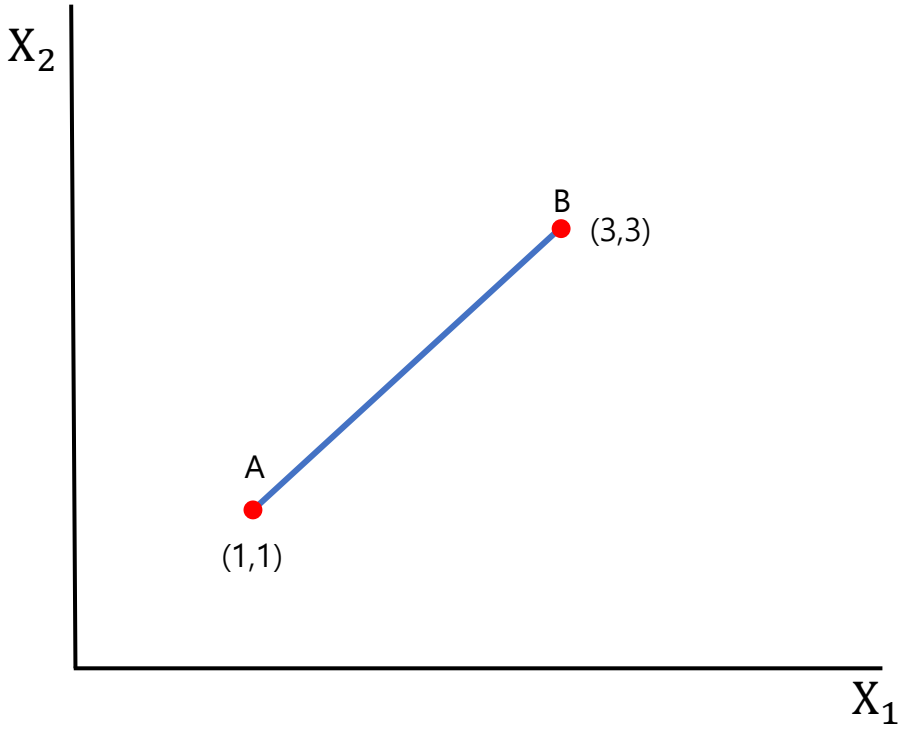
- Euclidean Distance
- Manhattan Distance
- Mahalanobis Distance
- Correlation Distance
  - Pearson Correlation
  - Spearman Rank Correlation

Euclidean Distance

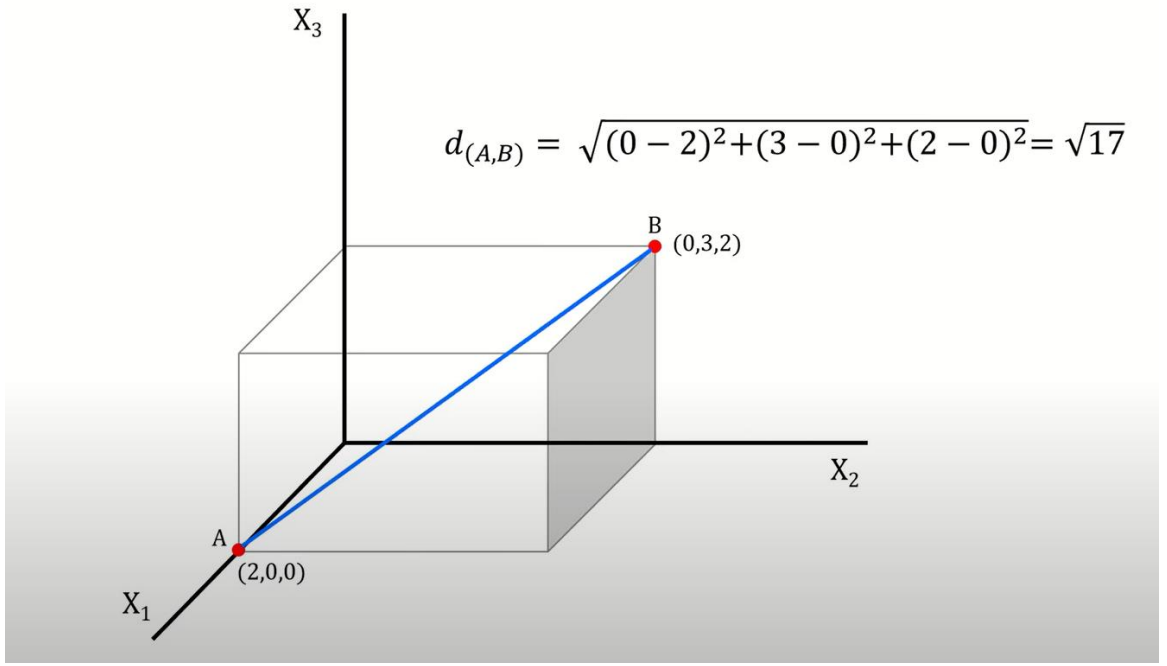
$$d_{(X,Y)} = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

- 가장 흔히 사용하는 거리측도
- 대응되는 X,Y값 간 차이 제곱합의 제곱근으로써, 두 관측치 사이의 직선거리 의미

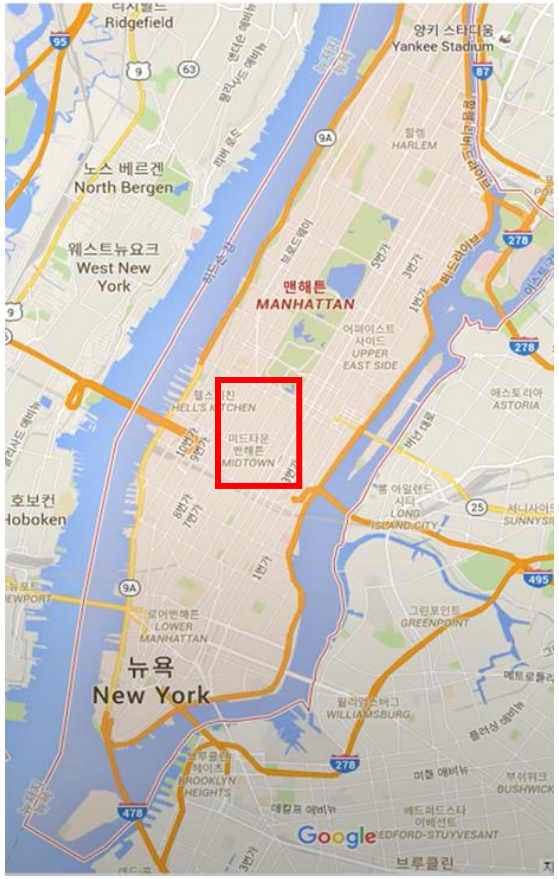
Euclidean Distance



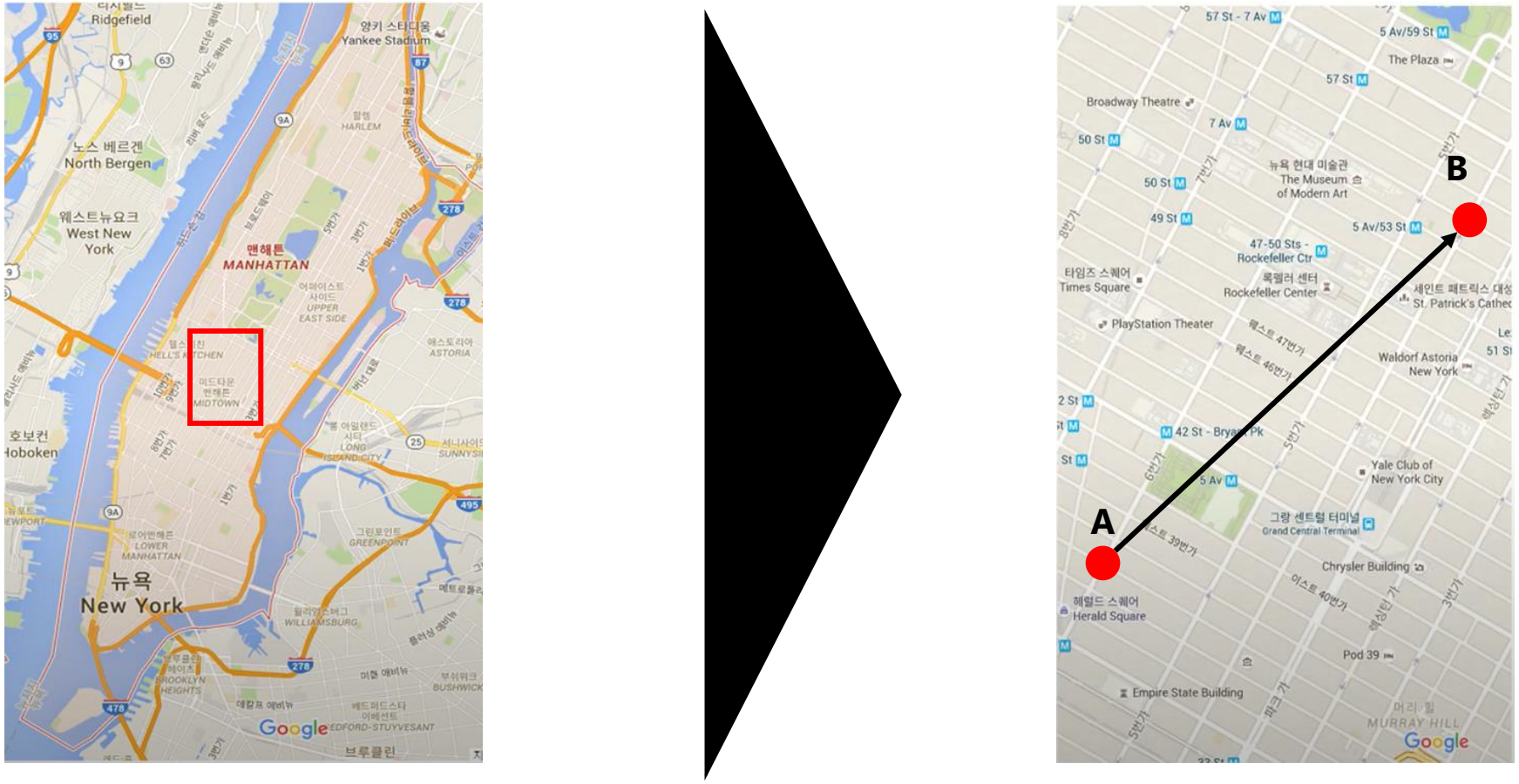
$$d_{(A,B)} = \sqrt{(3-1)^2+(3-1)^2} = \sqrt{8}$$



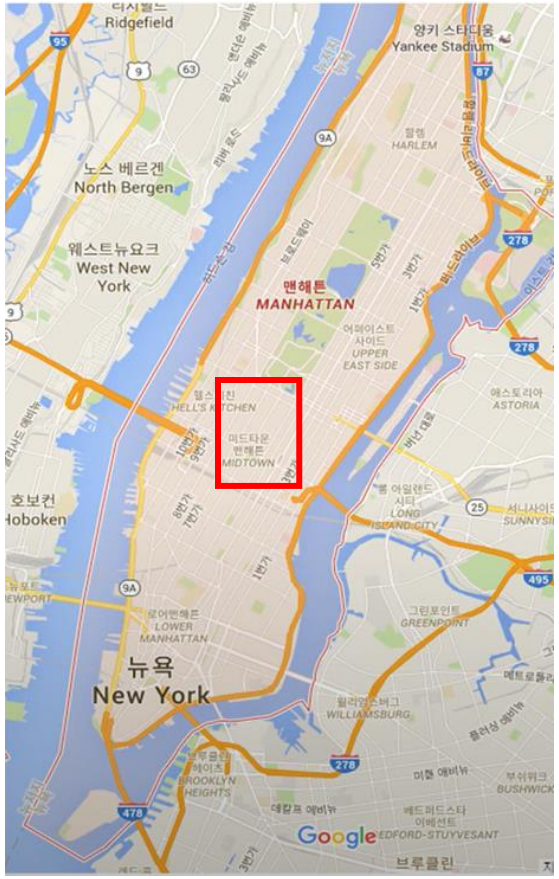
Manhattan Distance

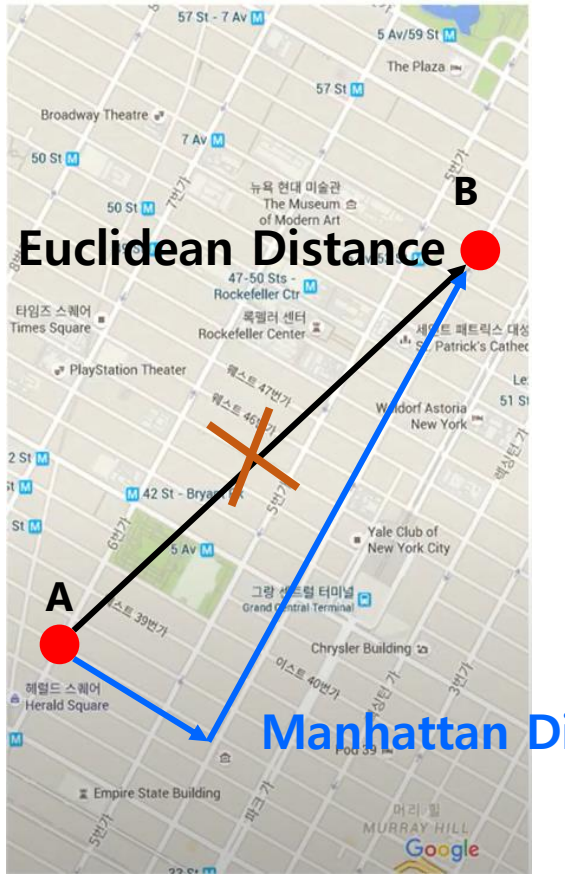


Manhattan Distance



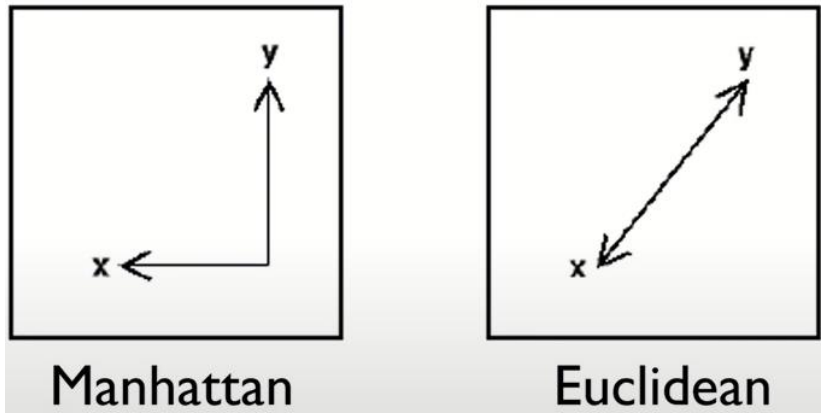
Manhattan Distance





Manhattan Distance

$$d_{Manhattan}(X, Y) = \sum_{i=1}^{n} |x_i - y_i|$$



- X에서 Y로 이동 시 각 좌표축 방향으로만 이동할 경우에 계산되는 거리

Mahalanobis Distance

$$\mathrm{d}_{\mathrm{Mahalanobis}}(\mathrm{X},\mathrm{Y}) = \sqrt{(X-Y)^{\mathrm{T}}\sum^{-1}(\mathrm{X}-\mathrm{Y})}$$

$\sum^{-1}$: inverse of covariance matrix

- 변수 내 분산, 변수 간 공분산을 모두 반영하여 X, Y 간 거리를 계산하는 방식
- 데이터의 covariance matrix가 identity matrix인 경우는 Euclidean Distance와 동일함

Mahalanobis Distance

$$\sqrt{(X-Y)^{\mathrm{T}}\sum^{-1}(\mathrm{X}-\mathrm{Y})}=\mathrm{c}\quad(\mathrm{c}\text{ is Mahalanobis Distance})$$

$$\Rightarrow\quad(X-Y)^{\mathrm{T}}\sum^{-1}(\mathrm{X}-\mathrm{Y})=\mathrm{c}^2$$

$$\text{Let X}=\begin{pmatrix}\mathrm{x}_1\\ \mathrm{x}_2\end{pmatrix},\mathrm{Y}=\begin{pmatrix}\mathrm{y}_1\\ \mathrm{y}_2\end{pmatrix},\sum^{-1}=\begin{pmatrix}{\mathrm{s}_{11}}^{-1} & {\mathrm{s}_{12}}^{-1}\\ {\mathrm{s}_{21}}^{-1} & {\mathrm{s}_{22}}^{-1}\end{pmatrix},\text{then}$$

$$\Rightarrow\quad(X-Y)^2{\mathrm{s}_{11}}^{-1}+2(\mathrm{x}_1\text{-}\mathrm{y}_1)(\mathrm{x}_2\text{-}\mathrm{y}_2){\mathrm{s}_{12}}^{-1}+(\mathrm{x}_2-\mathrm{y}_2)^2{\mathrm{s}_{22}}^{-1}=\mathrm{c}^2(\because {\mathrm{s}_{12}}^{-1}={\mathrm{s}_{21}}^{-1})$$

It can be considered as the squared Mahalanobis distance between a certain point X, and the fixed point Y.

$$\text{Let Y}=\begin{pmatrix}0\\ 0\end{pmatrix},\text{then}$$

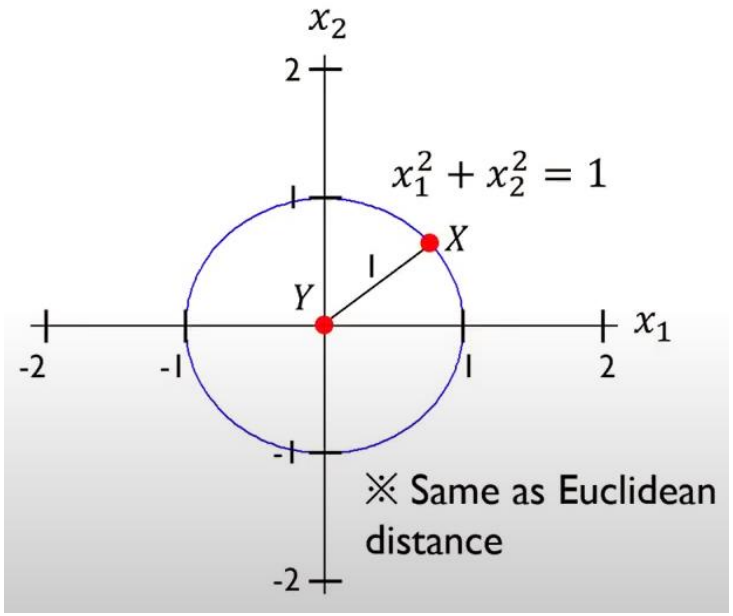
$$\Rightarrow\quad{\mathrm{x}_1}^2{\mathrm{s}_{11}}^{-1}+2\mathrm{x}_1\mathrm{x}_2{\mathrm{s}_{12}}^{-1}+{\mathrm{x}_2}^2{\mathrm{s}_{22}}^{-1}=\mathrm{c}^2$$

which is a general equation of the ellipse

Mahalanobis Distance

$$\Sigma \quad = \Sigma^{-1} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

(identity matrix)

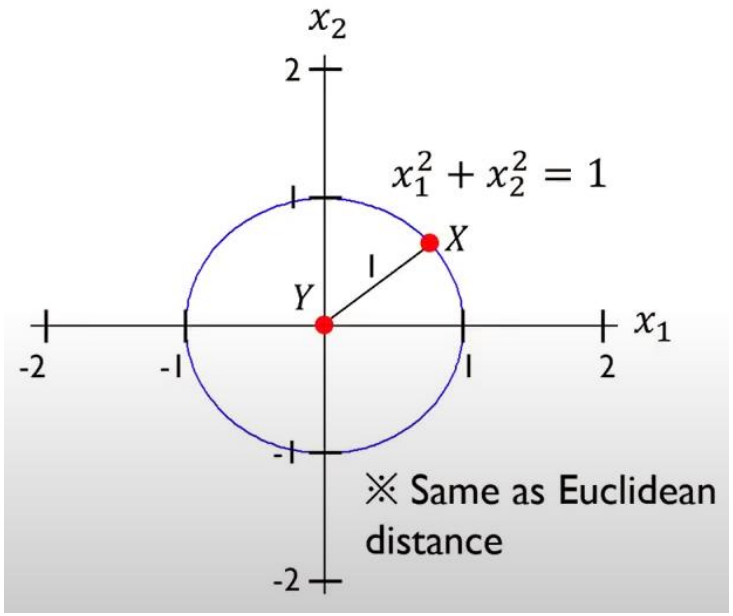


$$\Sigma = \begin{pmatrix} 4 & 0 \\ 0 & 1 \end{pmatrix},\ \Sigma^{-1} = \begin{pmatrix} \frac{1}{4} & 0 \\ 0 & 1 \end{pmatrix}$$

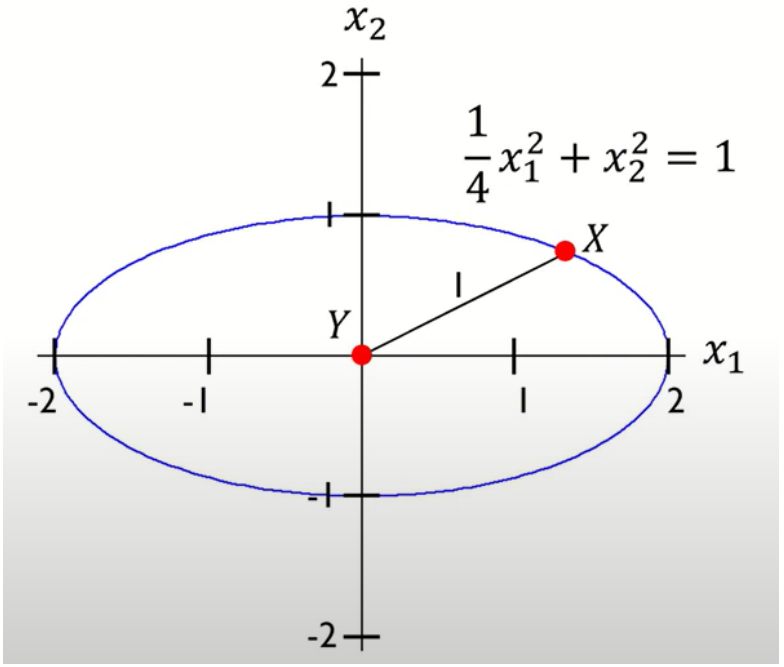
Mahalanobis Distance

$$\Sigma \quad = \Sigma^{-1} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

(identity matrix)



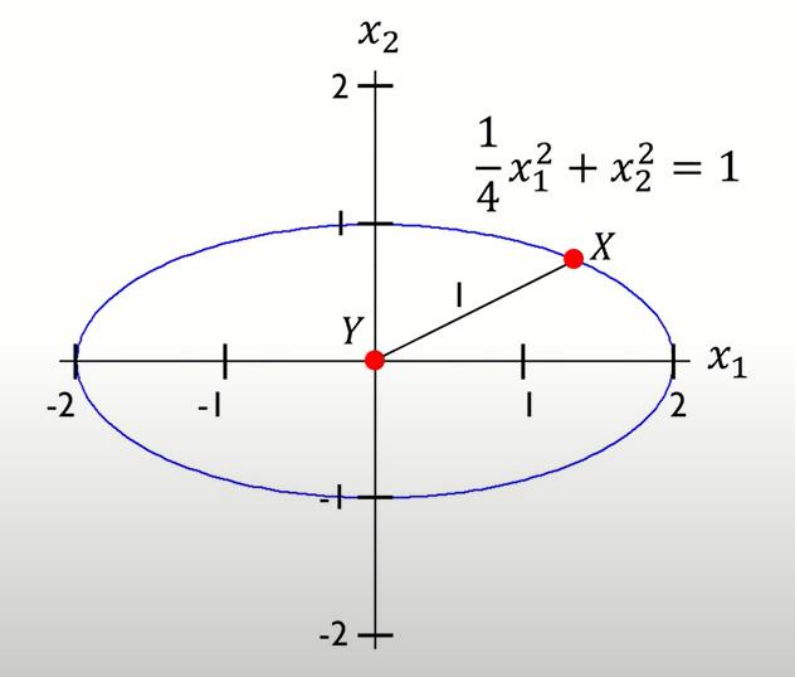
$$\Sigma = \begin{pmatrix} 4 & 0 \\ 0 & 1 \end{pmatrix},\ \Sigma^{-1} = \begin{pmatrix} \frac{1}{4} & 0 \\ 0 & 1 \end{pmatrix}$$



Mahalanobis Distance

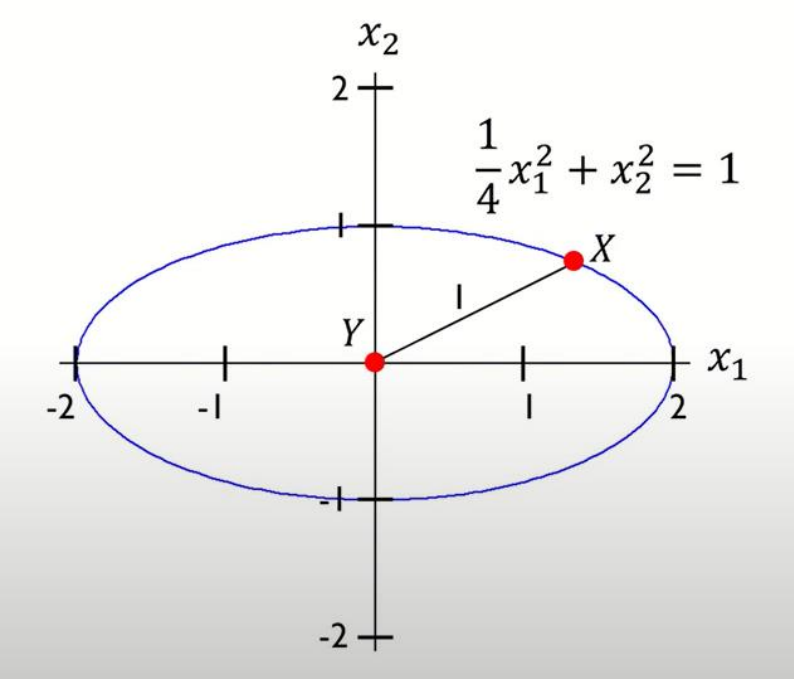
$$\Sigma = \begin{pmatrix} 4 & 0 \\ 0 & 1 \end{pmatrix},\ \Sigma^{-1} = \begin{pmatrix} \frac{1}{4} & 0 \\ 0 & 1 \end{pmatrix}$$

$$\Sigma = \begin{pmatrix} 4 & \sqrt{2} \\ \sqrt{2} & 1 \end{pmatrix},\ \Sigma^{-1} = \begin{pmatrix} \frac{1}{2} & -\sqrt{\frac{1}{2}} \\ \sqrt{\frac{1}{2}} & 2 \end{pmatrix}$$

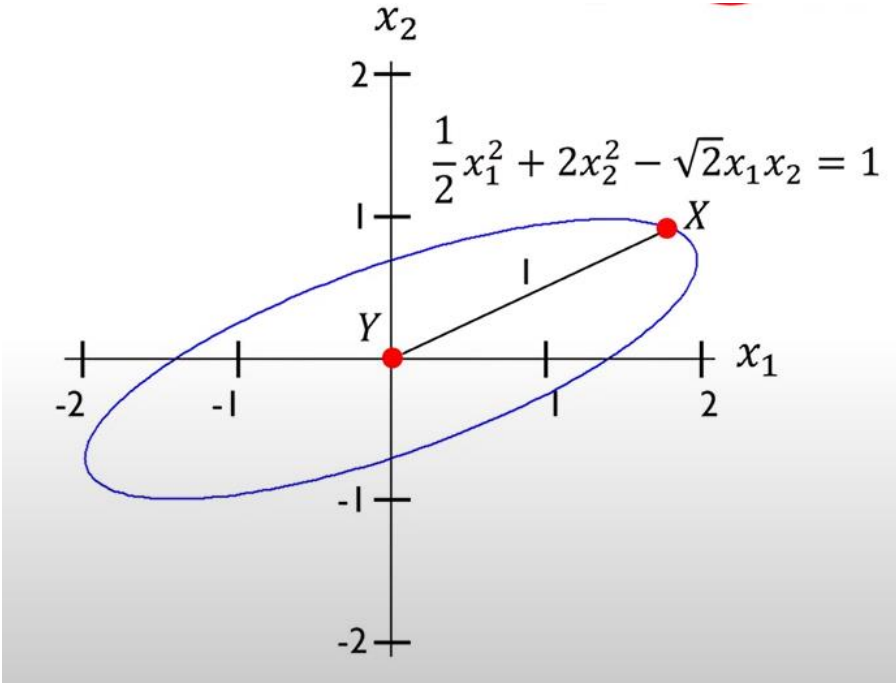


Mahalanobis Distance

$$\Sigma = \begin{pmatrix} 4 & 0 \\ 0 & 1 \end{pmatrix},\ \Sigma^{-1} = \begin{pmatrix} \frac{1}{4} & 0 \\ 0 & 1 \end{pmatrix}$$



$$\Sigma = \begin{pmatrix} 4 & \sqrt{2} \\ \sqrt{2} & 1 \end{pmatrix},\ \Sigma^{-1} = \begin{pmatrix} \frac{1}{2} & -\sqrt{\frac{1}{2}} \\ \sqrt{\frac{1}{2}} & 2 \end{pmatrix}$$

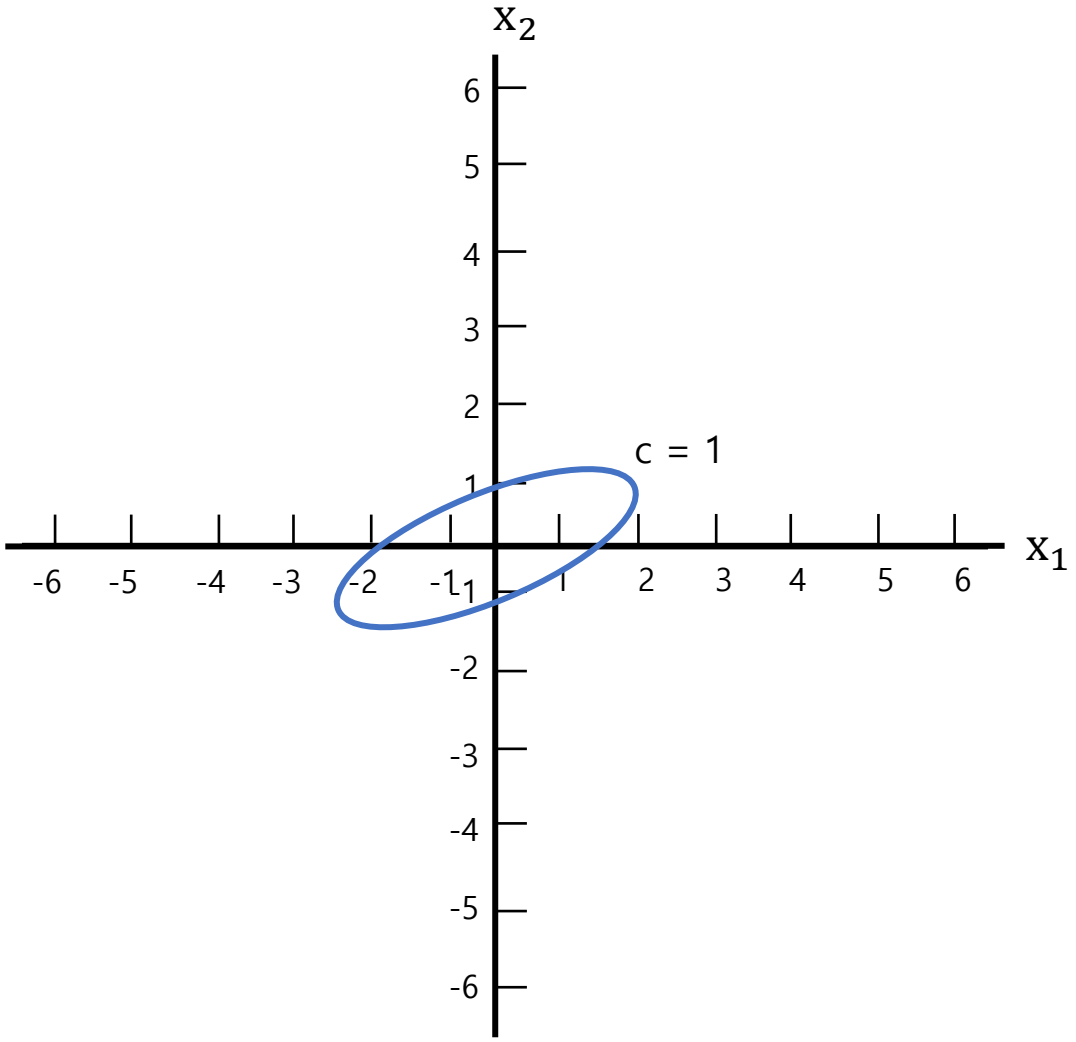


Mahalanobis Distance

$$\Sigma = \begin{pmatrix} 4 & \sqrt{2} \\ \sqrt{2} & 1 \end{pmatrix}, \quad \Sigma^{-1} = \begin{pmatrix} \frac{1}{2} & -\sqrt{\frac{1}{2}} \\ \sqrt{\frac{1}{2}} & 2 \end{pmatrix}$$

Equation of Ellipse

$$\frac{1}{2}x_1{}^2 + 2x_2{}^2 - \sqrt{2}x_1x_2 = c^2$$

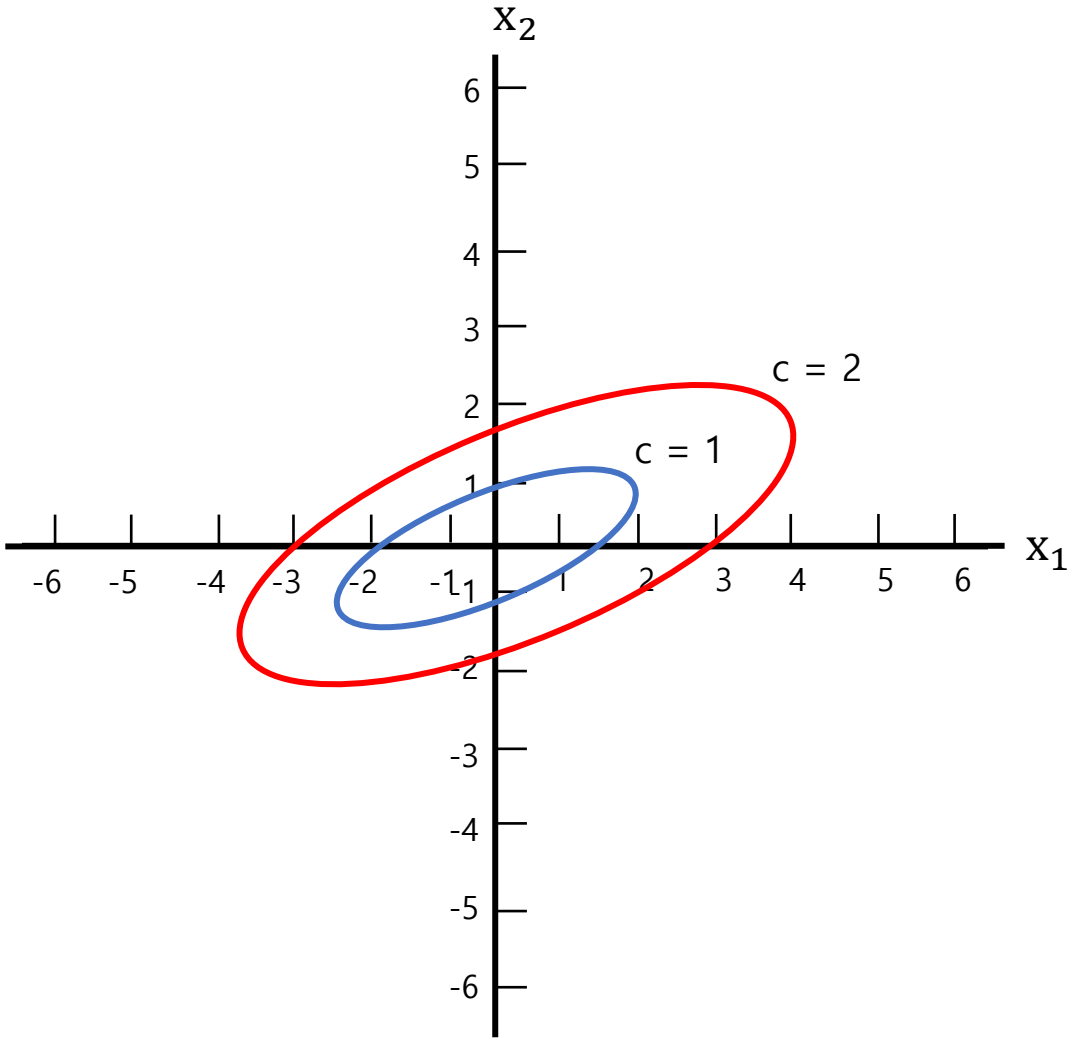


Mahalanobis Distance

$$\Sigma = \begin{pmatrix} 4 & \sqrt{2} \\ \sqrt{2} & 1 \end{pmatrix}, \quad \Sigma^{-1} = \begin{pmatrix} \frac{1}{2} & -\sqrt{\frac{1}{2}} \\ \sqrt{\frac{1}{2}} & 2 \end{pmatrix}$$

Equation of Ellipse

$$\frac{1}{2}x_1{}^2 + 2x_2{}^2 - \sqrt{2}x_1x_2 = c^2$$

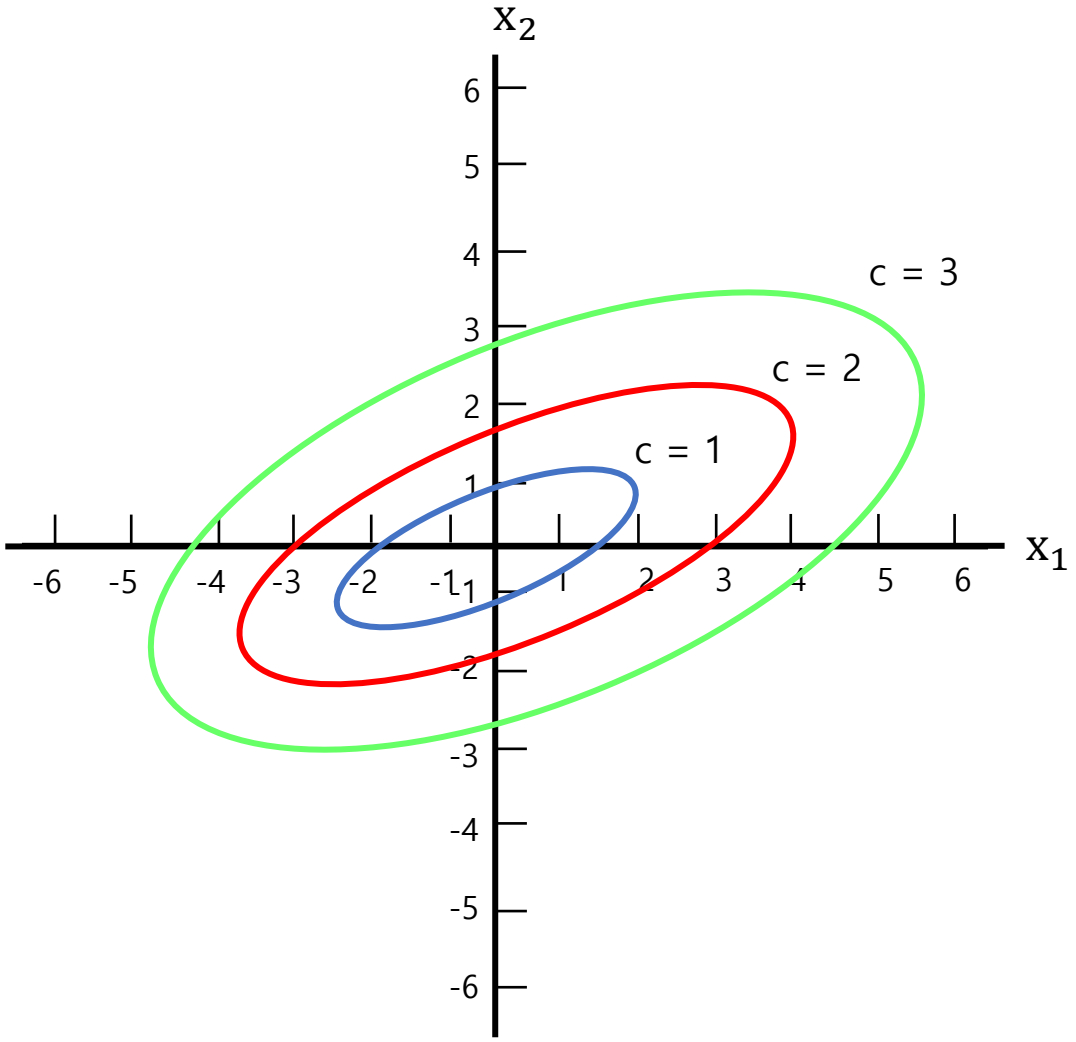


Mahalanobis Distance

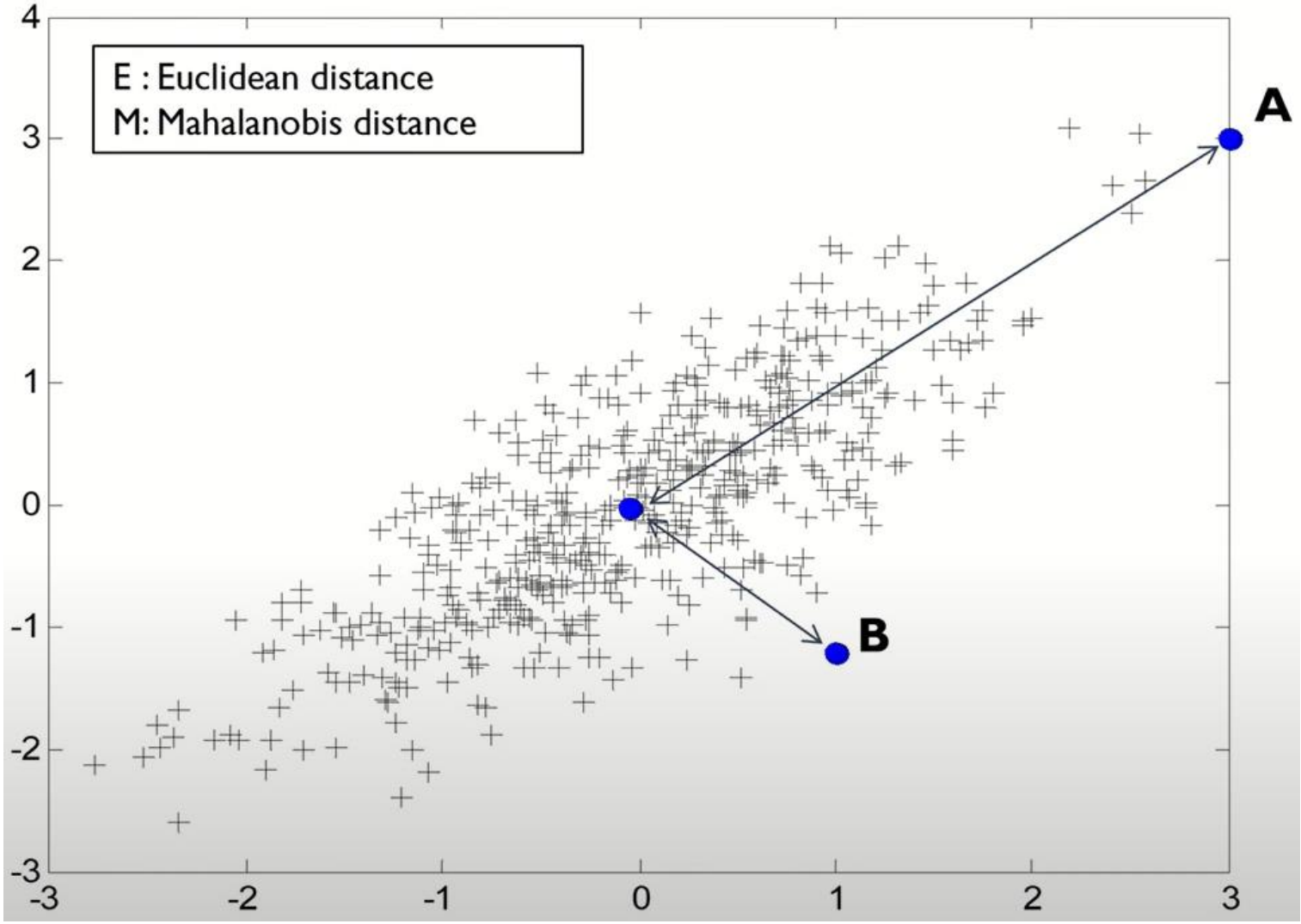
$$\Sigma = \begin{pmatrix} 4 & \sqrt{2} \\ \sqrt{2} & 1 \end{pmatrix}, \quad \Sigma^{-1} = \begin{pmatrix} \frac{1}{2} & -\sqrt{\frac{1}{2}} \\ \sqrt{\frac{1}{2}} & 2 \end{pmatrix}$$

Equation of Ellipse

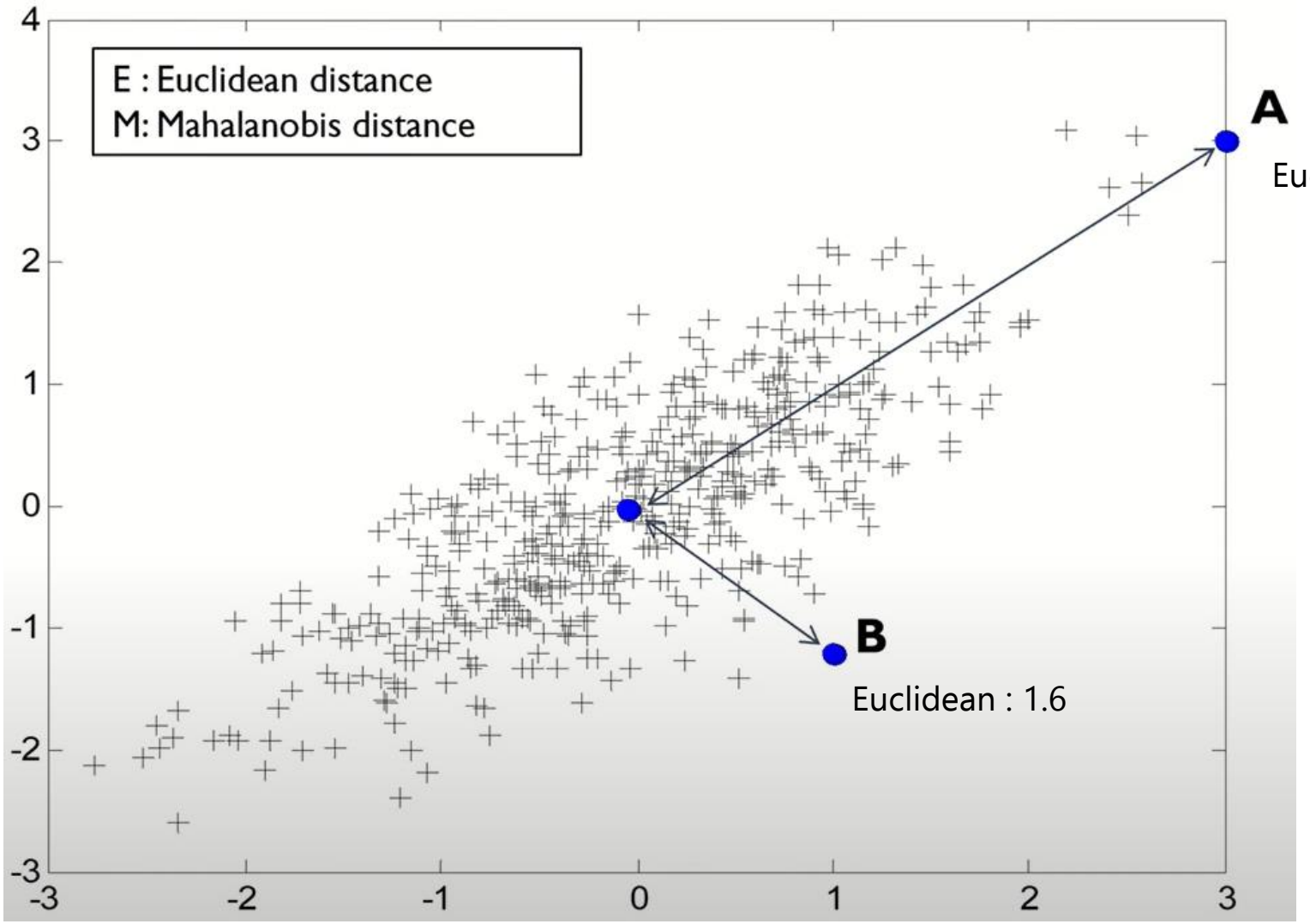
$$\frac{1}{2}x_1{}^2 + 2x_2{}^2 - \sqrt{2}x_1x_2 = c^2$$



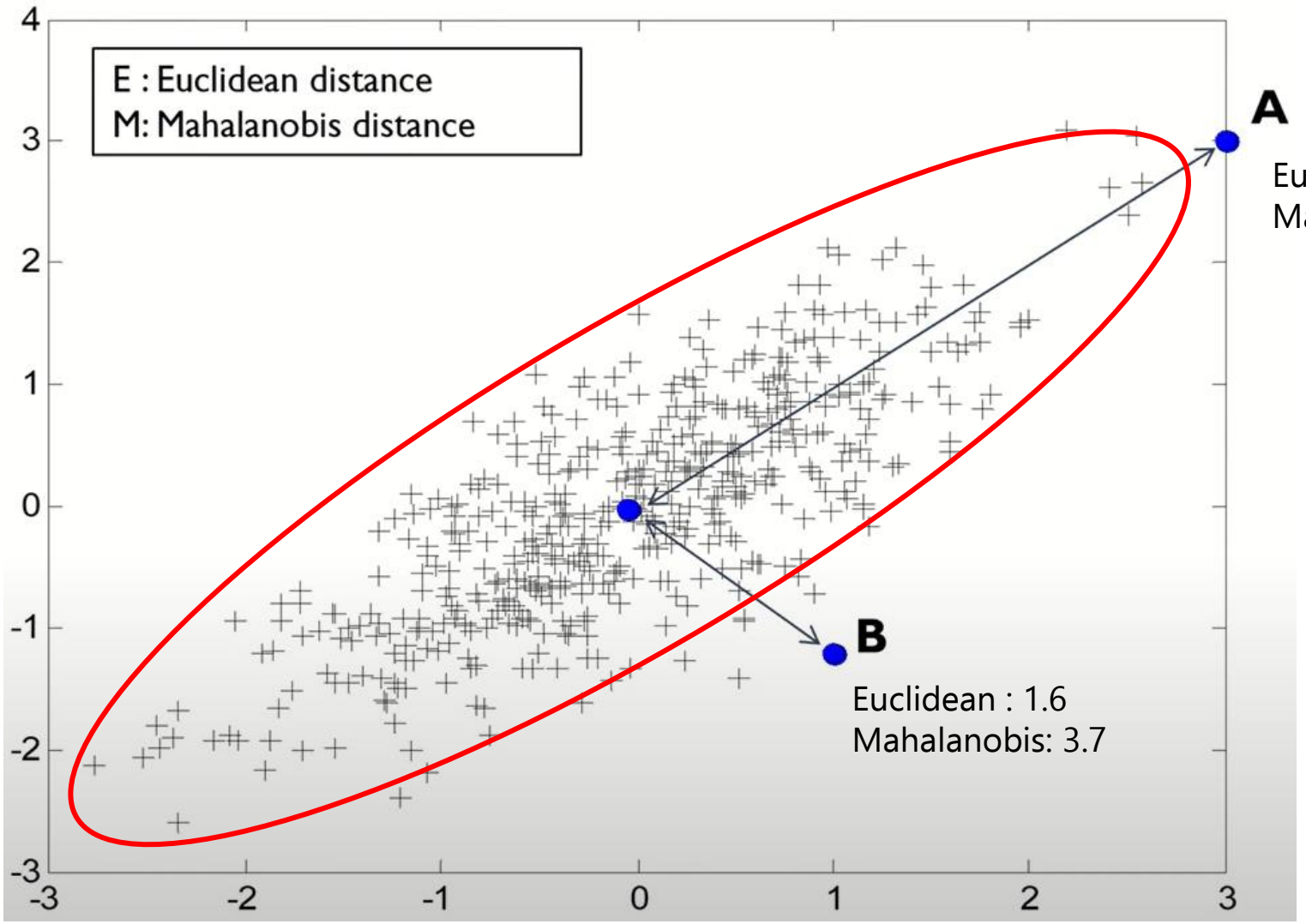
Mahalanobis Distance



Mahalanobis Distance



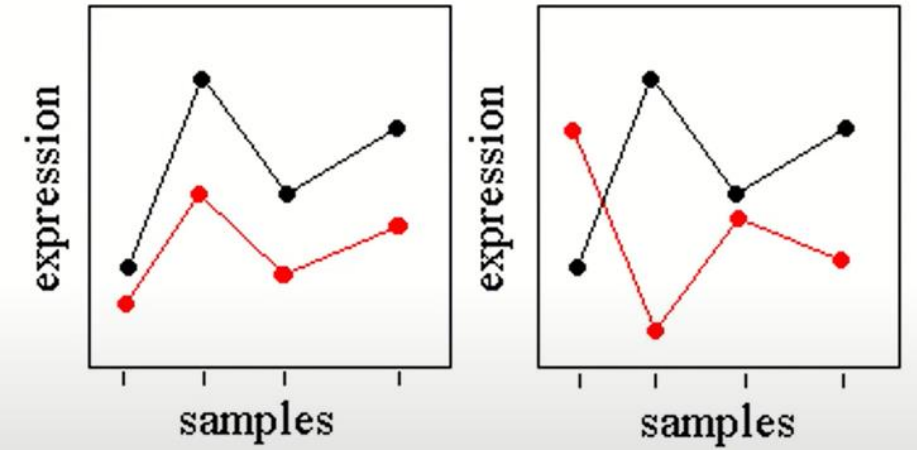
Mahalanobis Distance



Correlation Distance

$$d_{Corr}(X,Y) = 1 - r$$

$$\text{Where } r = \sigma_{XY}$$



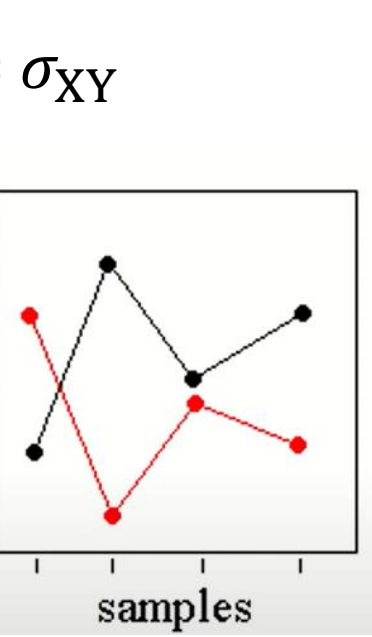
- 데이터 간 Pearson correlation을 거리측도로 사용하는 방식으로, 데이터 패턴의 유사도를 반영할 수 있음

Correlation Distance

$$d_{Corr}(X,Y) = 1 - r$$

Where $r = \sigma_{XY}$

r = correlation, -1 <= r <= 1



0 <= d <= 2

- 데이터 간 Pearson correlation을 거리측도로 사용하는 방식으로, 데이터 패턴의 유사도를 반영할 수 있음

Spearman Rank Correlation Distance

$$d_{Spearman}(X,Y) = 1 - p$$

$$\text{Where } p = 1 - \frac{6\sum_{i=1}^{n}(\text{rank}(x_i) - \text{rank}(y_i))^2}{n(n^2-1)}$$

- P를 Spearman Correlation이라 하며, 이는 데이터의 rank를 이용하여 correlation distance를 계산하는 방식임
- P의 범위는 -1부터 1로, Pearson Correlation과 동일

Spearman Rank Correlation Distance

$$1 - \frac{6 \sum_{i=1}^{n} (\mathrm{rank}(x_i) - \mathrm{rank}(y_i))^2}{n(n^2 - 1)}$$

계절 평균 낮 최고 기온

지역	봄	여름	가을	겨울
서울	17.06	28.43	19.07	3.50
뉴욕	16.32	28.22	18.37	5.43
시드니	22.23	17.03	21.90	25.63

지역 별 계절 기온 순위

지역	봄	여름	가을	겨울
서울	3	1	2	4
뉴욕	3	1	2	4
시드니	2	4	3	1

Spearman Rank Correlation Distance

$$1 - \frac{6\sum_{i=1}^{n}(\mathrm{rank}(x_i)-\mathrm{rank}(y_i))^2}{n(n^2-1)}$$

계절 평균 낮 최고 기온

지역	봄	여름	가을	겨울
서울	17.06	28.43	19.07	3.50
뉴욕	16.32	28.22	18.37	5.43
시드니	22.23	17.03	21.90	25.63

지역 별 계절 기온 순위

지역	봄	여름	가을	겨울
서울	3	1	2	4
뉴욕	3	1	2	4
시드니	2	4	3	1

Spearman correlation distance between Seoul - New York

$$p = 1 - \frac{6\{(3-3)^2 + (1-1)^2 + (2-2)^2 + (4-4)^2\}}{4(4^2-1)} = 1 \rightarrow d_{(\mathrm{Seuol,NewYork})} = 1 - 1 = 0$$

Spearman Rank Correlation Distance

$$1 - \frac{6\sum_{i=1}^{n}(\mathrm{rank}(x_i)-\mathrm{rank}(y_i))^2}{n(n^2-1)}$$

계절 평균 낮 최고 기온

지역	봄	여름	가을	겨울
서울	17.06	28.43	19.07	3.50
뉴욕	16.32	28.22	18.37	5.43
시드니	22.23	17.03	21.90	25.63

지역 별 계절 기온 순위

지역	봄	여름	가을	겨울
서울	3	1	2	4
뉴욕	3	1	2	4
시드니	2	4	3	1

Spearman correlation distance between Seoul - New York

$$p = 1 - \frac{6\{(3-3)^2 + (1-1)^2 + (2-2)^2 + (4-4)^2\}}{4(4^2-1)} = 1 \rightarrow d_{(Seuol,NewYork)} = 1 - 1 = 0$$

Spearman correlation distance between Seoul - Sydney

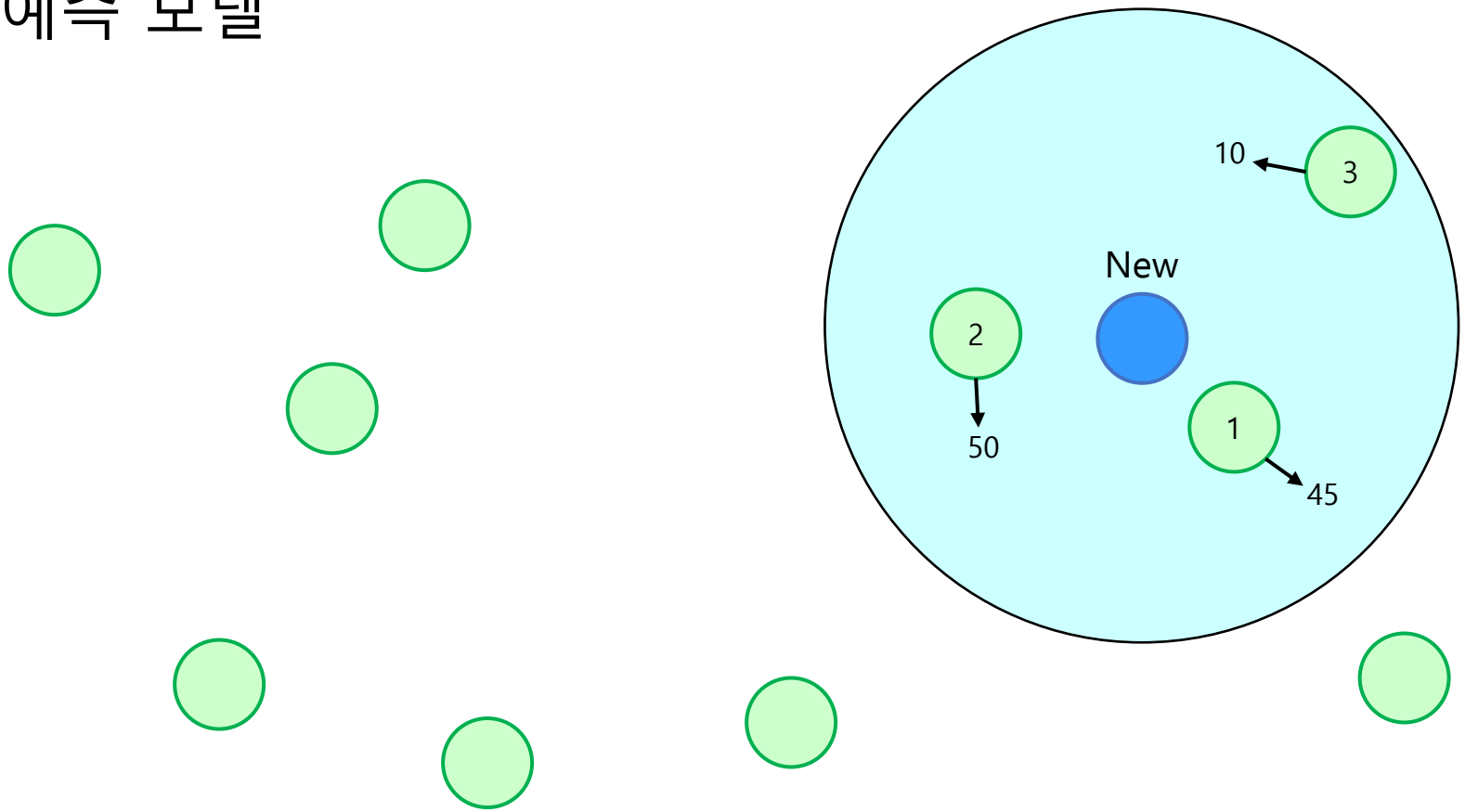
$$p = 1 - \frac{6\{(3-2)^2 + (1-4)^2 + (2-3)^2 + (4-1)^2\}}{4(4^2-1)} = 1 \rightarrow d_{(Seuol,Sydney)} = 1 - (-1) = 2$$

Advantages and Limitations of KNN

- 장점
  - 데이터 내 노이즈에 영향을 크게 받지 않으며, 특히 Mahalanobis Distance와 같이 데이터의 분산을 고려할 경우 강건함
  - 학습 데이터의 수가 많을 경우 효과적임
- 한계점
  - 파라미터 k의 값을 설정해야 함
  - 어떤 거리 척도가 분석에 적합한 지 불분명하며, 따라서 데이터의 특성에 맞는 거리측도를 임의로 선정해야 함
  - 새로운 관측치와 각각의 학습 데이터 간 거리를 전부 측정해야 하므로, 계산시간이 오래 걸리는 단점이 있음

Weighted KNN

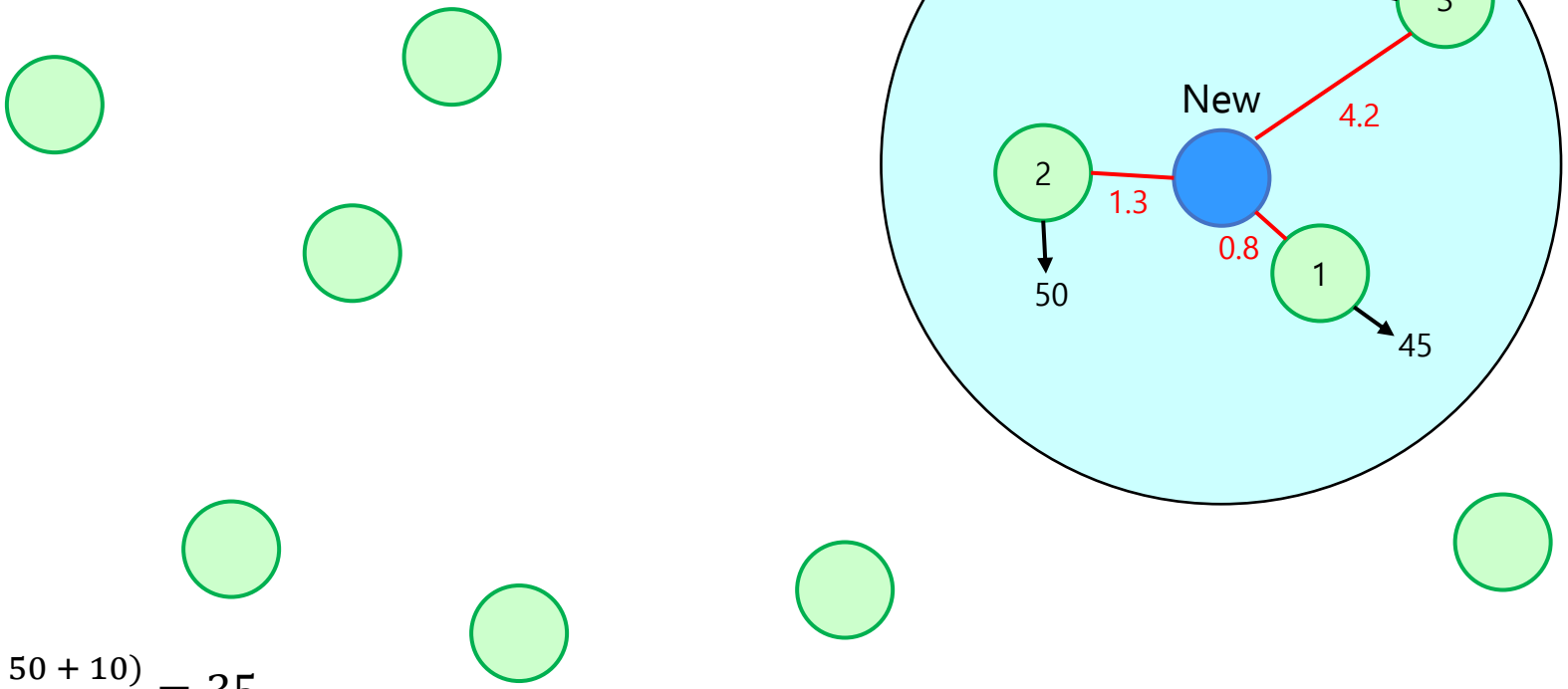
- 3NN 예측 모델



Q. 관측치 1,2,3,이 전부 같은 가중치를 가져야 하는가?

Weighted KNN

- Weighted 3NN 예측 모델

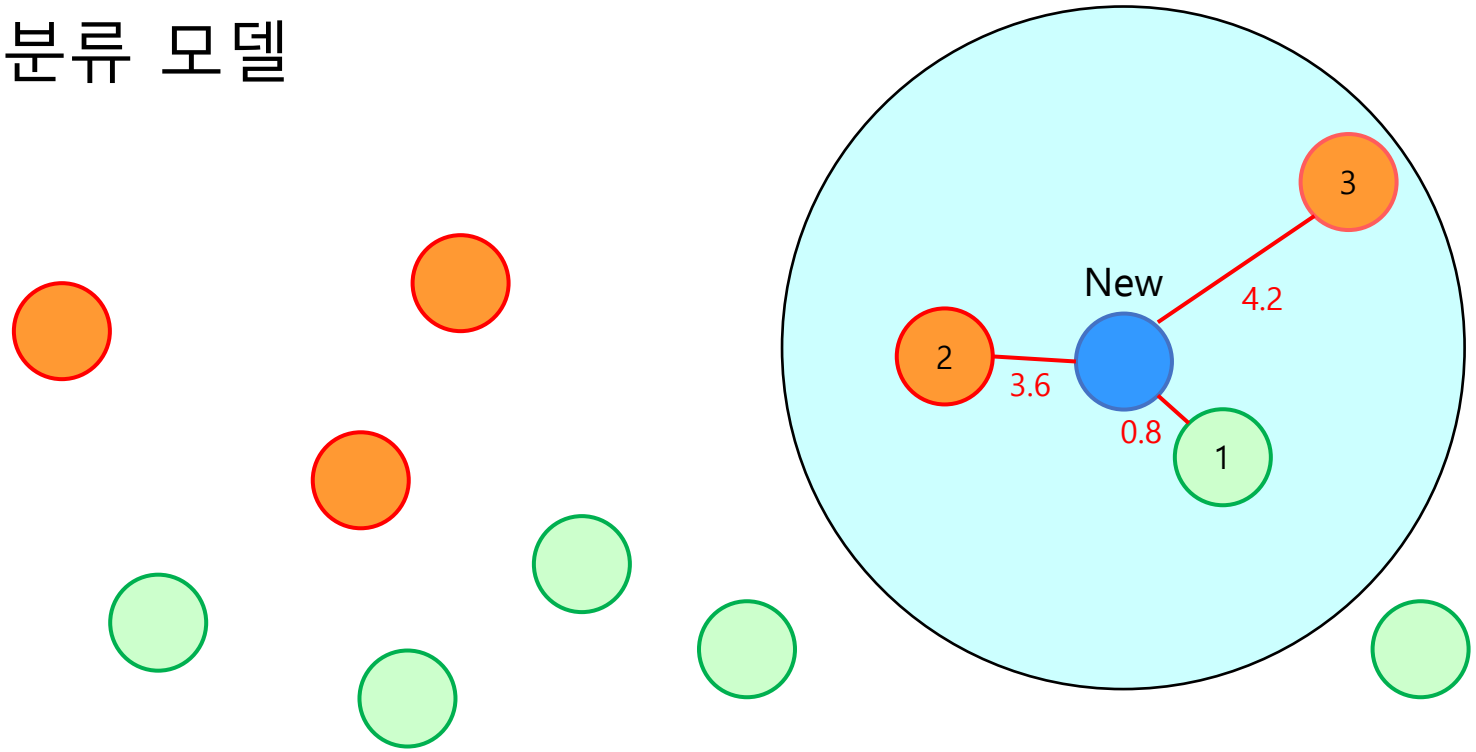


$$New = \frac{(45 + 50 + 10)}{3} = 35$$

$$New_{weighted} = \left(\frac{1}{0.8^2} \cdot 45 + \frac{1}{1.3^2} \cdot 50 + \frac{1}{4.2^2} \cdot 10\right) / \left(\frac{1}{0.8^2} + \frac{1}{1.3^2} + \frac{1}{4.2^2}\right) = 45.4$$

Example of Weighted KNN

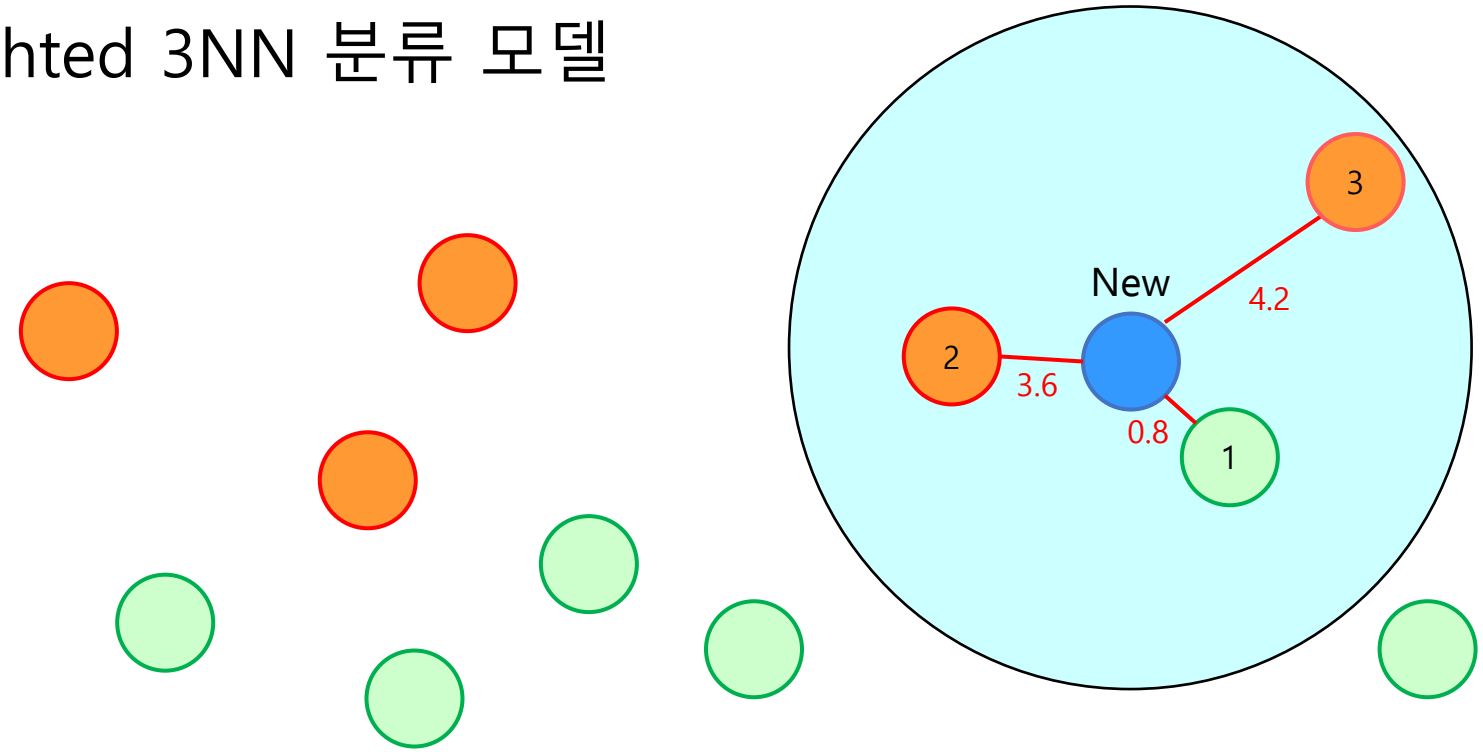
- 3NN 분류 모델



New = Orange

Example of Weighted KNN

- Weighted 3NN 분류 모델



$New = Orange$

$$New_{weighted} = \begin{cases} Orange = \frac{1}{3.6^2} + \frac{1}{4.2^2} \cong 0.13 \\ Green = \frac{1}{0.8^2} \cong 1.56 \end{cases} = Green$$

Example of Weighted KNN

- 새 data와 기존 학습 관측치 간의 거리를 가중치로 하여 예측 결과를 도출함

- 예측 모델

$$\hat{y}_{new} = \frac{\sum_{i=1}^{k} w_i y_i}{\sum_{i=1}^{k} w_i} \quad where\ w_i = \frac{1}{d_{(new,x_i)}{}^2}$$

- 분류 모델

$$\hat{c}_{new} = \max_{c} \sum_{i=1}^{k} w_i I(w_i \in c)$$

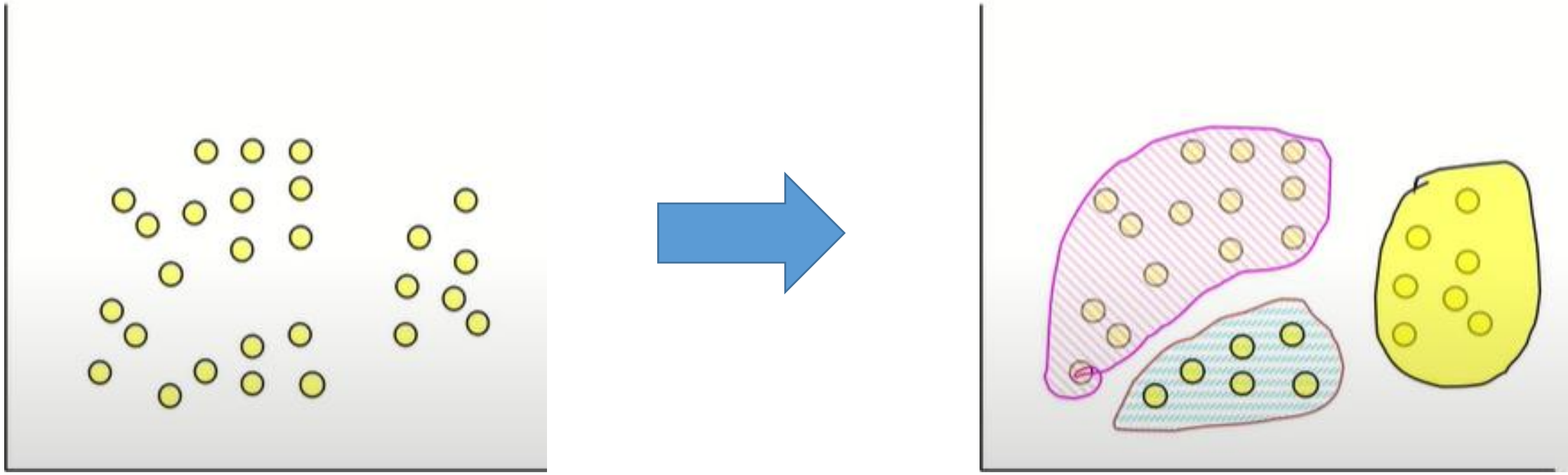
Summary of KNN

- KNN은 매우 단순한 접근방식으로 새로운 관측치 분류, 예측할 수 있는 방법
- 선형모델과 같이 학습 데이터로부터 특정 형태의 모델 제시하는 것 아니라,
  학습 데이터 내 유사한 관측치들 만을 토대로 새로운 데이터 예측 수행
- 일부 유사한 관측치의 반응변수의 조합 (e.g., average, majority voting)을 통해
  예상되는 반응변수 값 제공
- Weighted KNN Algorithm으로 데이터의 가중치를 고려할 수 있으며,
  이를 통해 보다 정확한 모델 구축 가능

Clustering Analysis

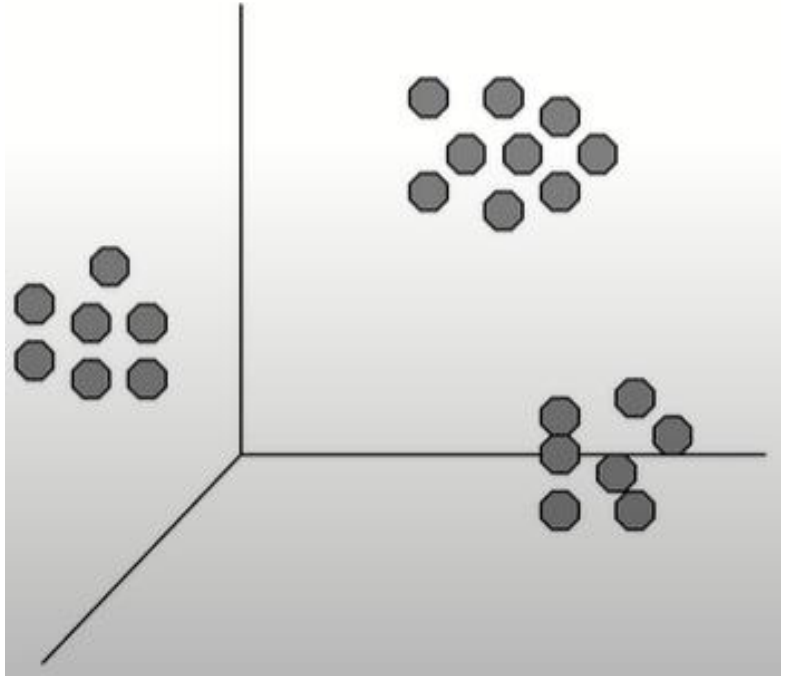
What is Clustering?

- 유사한 속성들을 갖는 관측치들을 묶어 전체 data를 몇 개의 군집(그룹)으로 나누는 것



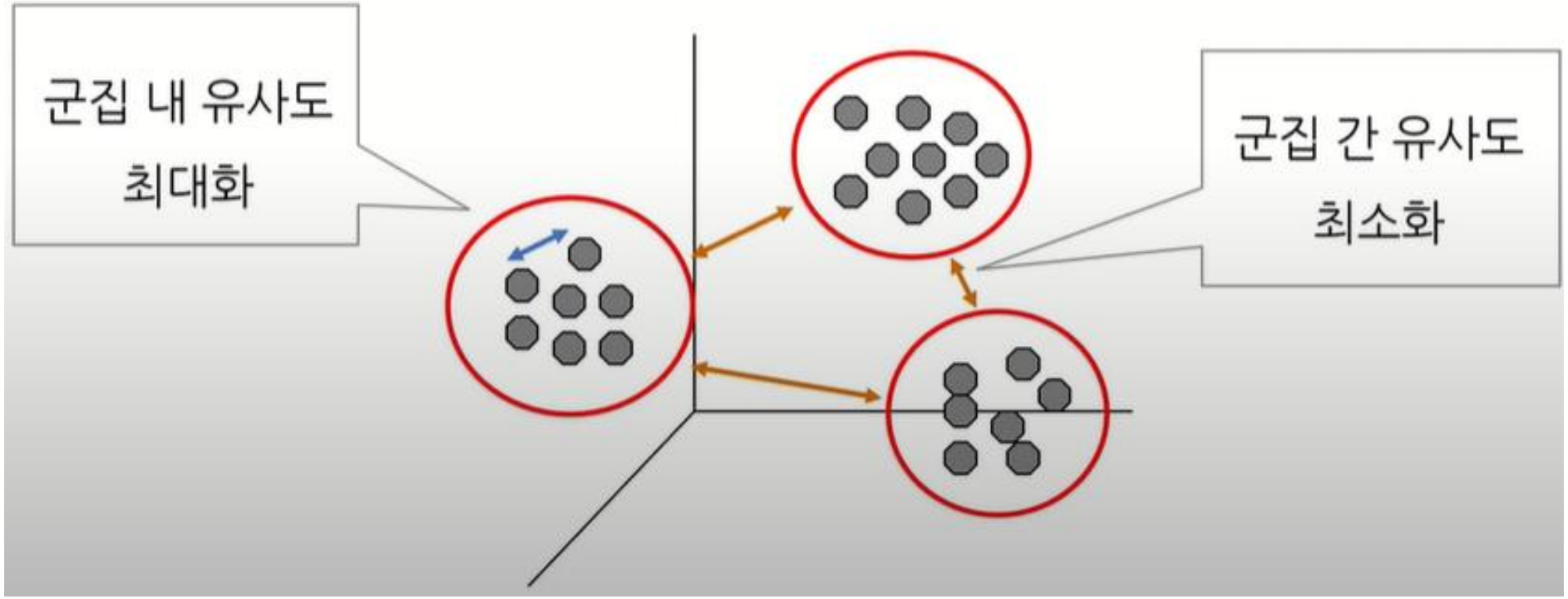
What is Clustering?

- 군집화 기준
  - 동일한 군집에 소속된 관측치들은 서로 유사할수록 좋음
  - 상이한 군집에 소속된 관측치들은 서로 다를수록 좋음



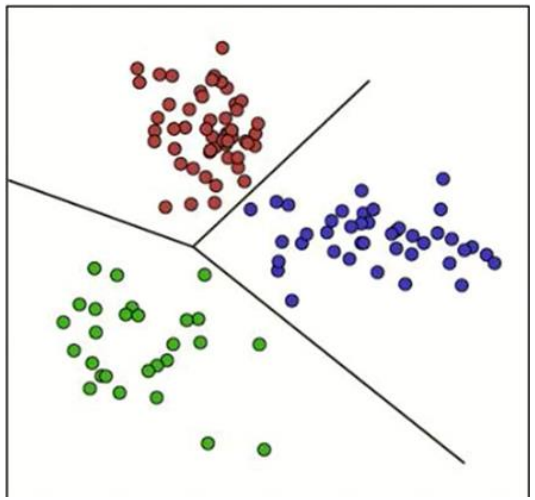
What is Clustering?

- 군집화 기준
  - 동일한 군집에 소속된 관측치들은 서로 유사할수록 좋음
  - 상이한 군집에 소속된 관측치들은 서로 다를수록 좋음

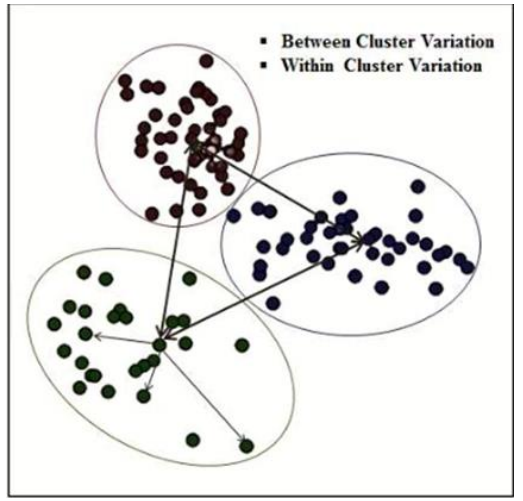


What is Clustering?

- 분류(Classification) vs 군집화(Clustering)
  - 분류: 사전 정의된 범주가 있는(labeled) 데이터로부터 예측 모델을 학습하는 문제
    (지도학습: Supervised Learning)
  - 군집화: 사전 정의된 범주가 없는(unlabeled) 데이터에서 최적의 그룹을 찾아나가는 문제
    (비지도학습: Unsupervised Learning)

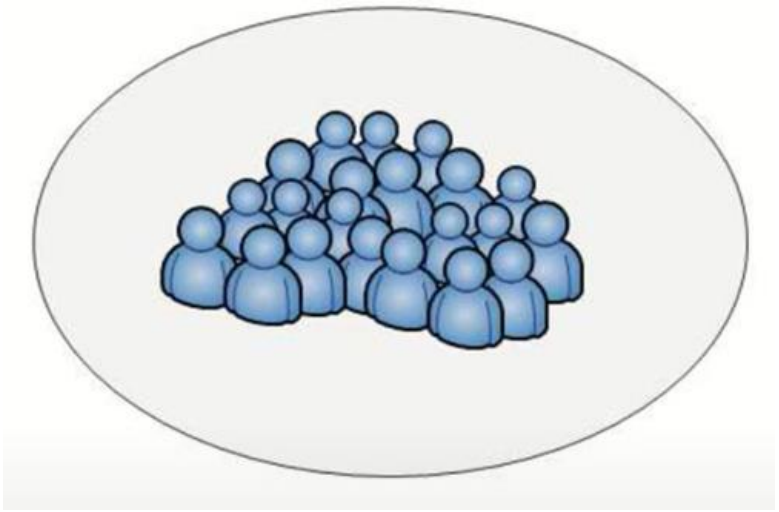


분류

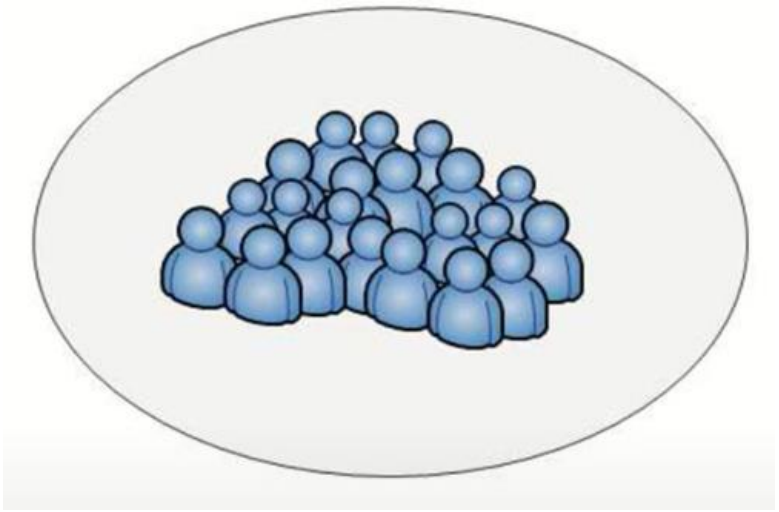


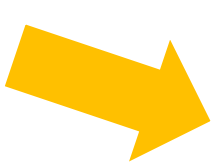
군집

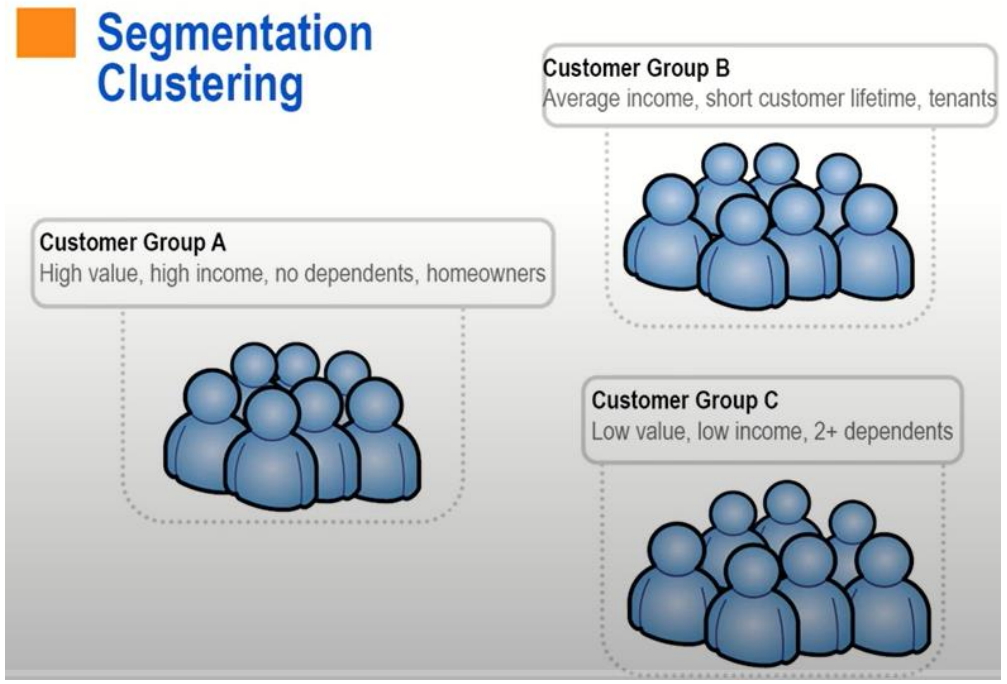
Example of Clustering



Example of Clustering

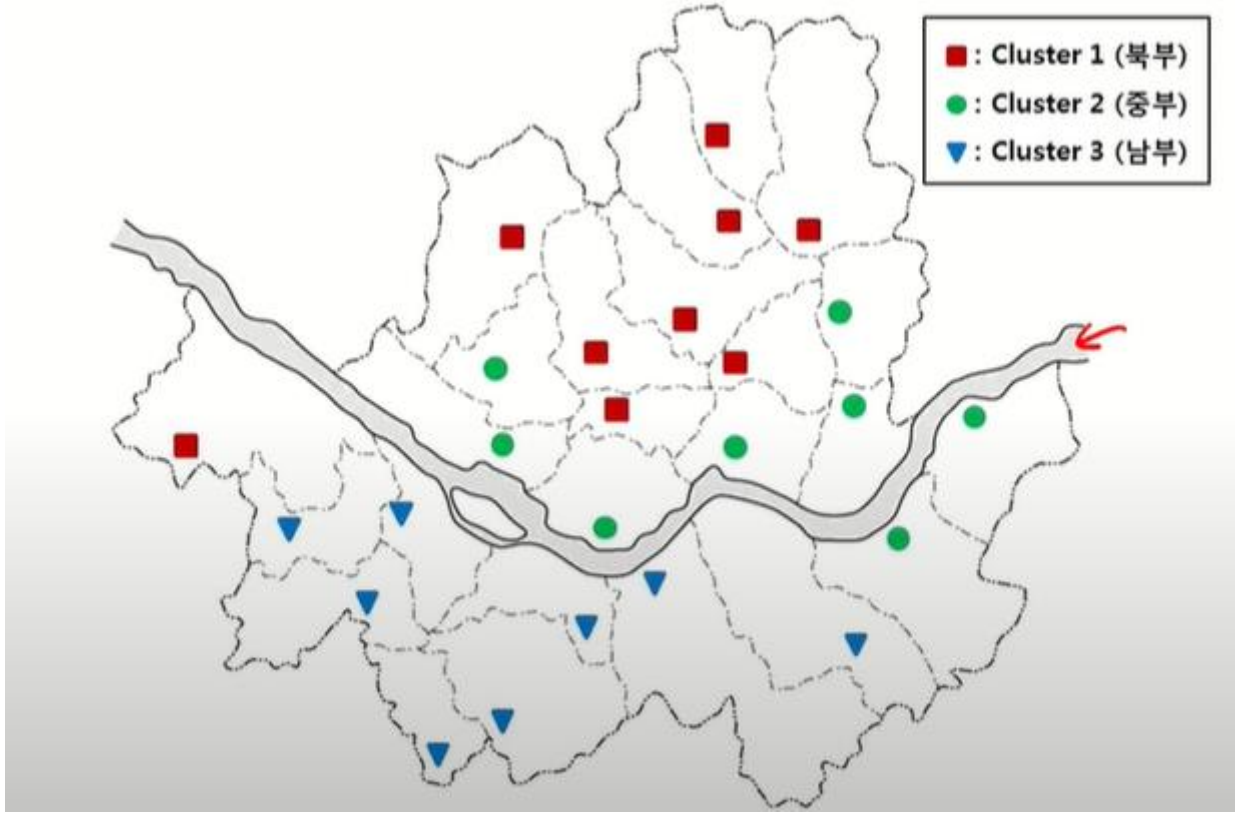






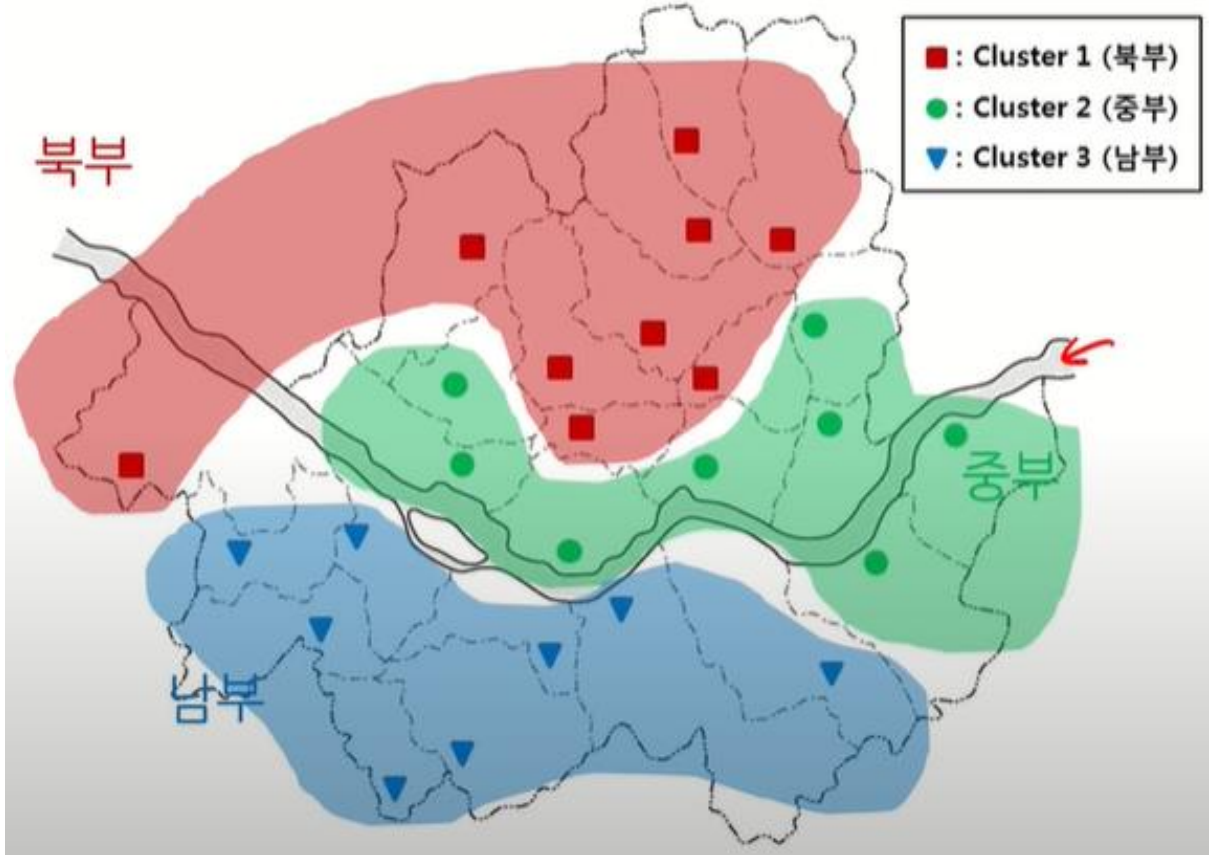
Example of Clustering

- 군집화 적용 사례
  - 서울시 오존농도 패턴 군집화 (25개 구)



Example of Clustering

- 군집화 적용 사례
  - 서울시 오존농도 패턴 군집화 (25개 구)



Considerations of Clustering

Q1. 어떤 거리 척도를 사용하여 유사도를 측정할 것인가?

Q2. 어떤 군집화 알고리즘을 사용할 것인가?

Q3. 어떻게 최적의 군집 수를 결정할 것인가?

Q4. 어떻게 군집화 결과를 측정/평가할 것인가?

Clustering: Similarity Scale

Q1. 어떤 거리 척도를 사용하여 유사도를 측정할 것인가?

- Euclidean Distance
- Manhattan Distance
- Mahalanobis Distance
- Correlation Distance

Clustering: Similarity Scale

Q2. 어떤 군집화 알고리즘을 사용할 것인가?

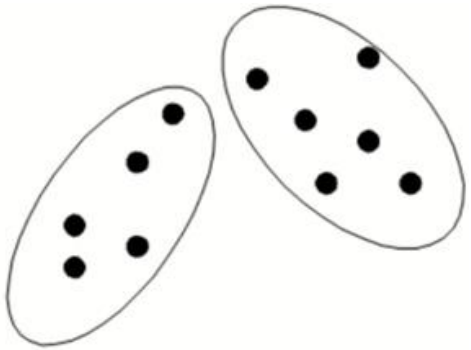
- 군집화 알고리즘의 종류

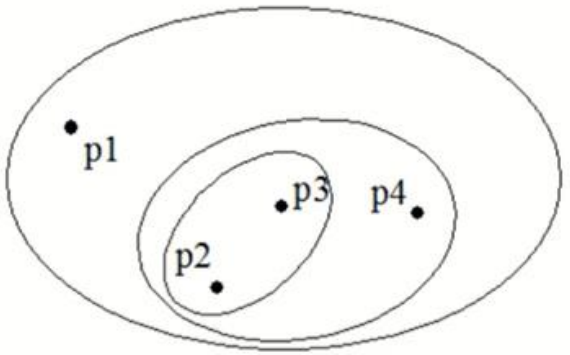
✓ 계층적 군집화

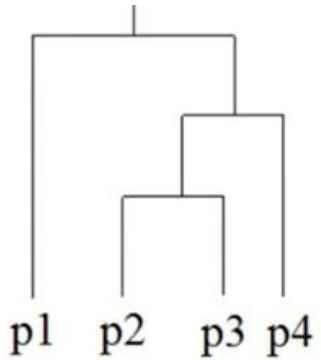
- 개체들을 가까운 집단부터 차근차근 묶어나가는 방식
- 군집화 결과 뿐만 아니라 유사한 개체들이 결합되는 dendrogram 생성

✓ 분리형 군집화

- 전체 데이터의 영역을 특정 기준에 의해 동시에 구분
- 각 개체들은 사전에 정의된 개수의 군집 중 하나에 속하게 됨







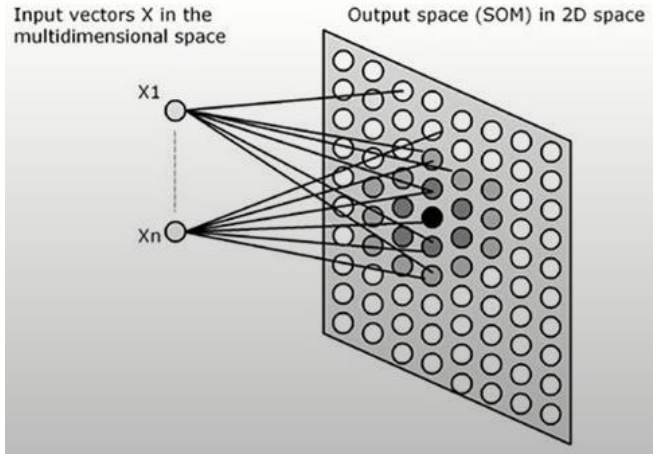
Clustering: Similarity Scale

Q2. 어떤 군집화 알고리즘을 사용할 것인가?

- 군집화 알고리즘의 종류

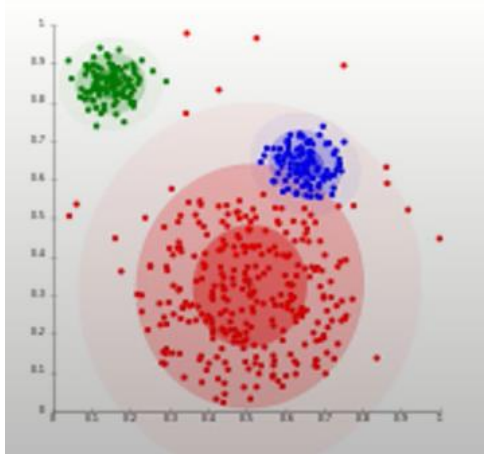
✓ 자기조직화 지도

- 2차원 격자에 각 개체들이 대응하도록 인경신경망과 유사한 학습을 통해 군집 도출



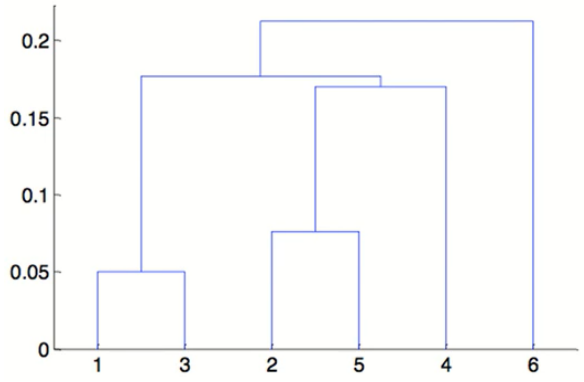
✓ 분포 기반 군집화

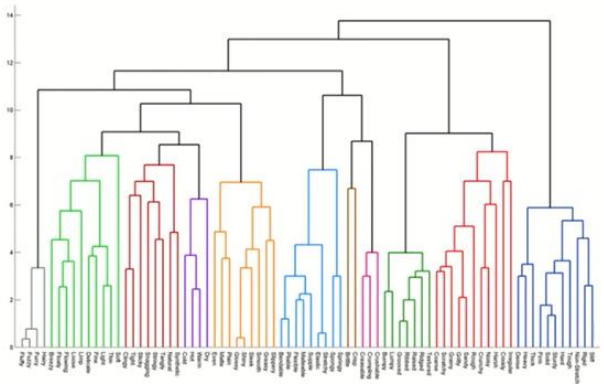
- 데이터의 분포를 기반으로 높은 밀도를 갖는 세부 영역들로 전체 영역을 구분



Hierarchical Clustering

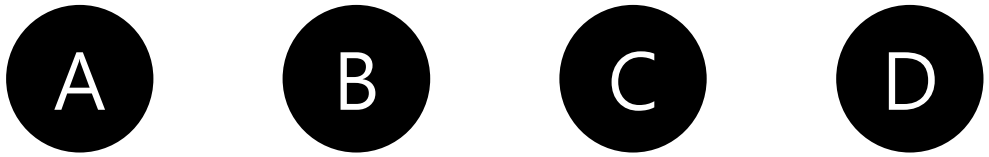
- 계층적 군집화
  - 계층적 트리모형을 이용해 개별 개체들을 순차적/계층적으로 유사한 개체/군집과 통합
  - Dendrogram을 통해 시각화 가능
    - ✓ Dendrogram: 개체들이 결합되는 순서를 나타내는 트리형태의 구조
  - 사전에 군집의 수를 정하지 않아도 수행 가능
    - ✓ Dendrogram 생성 후 적절한 수준에서 자르면 그에 해당하는 군집화 결과 생성





Hierarchical Clustering

- 계층적 군집화 수행 예시
  - 모든 개체들 사이의 거리에 대한 유사도 행렬 계산

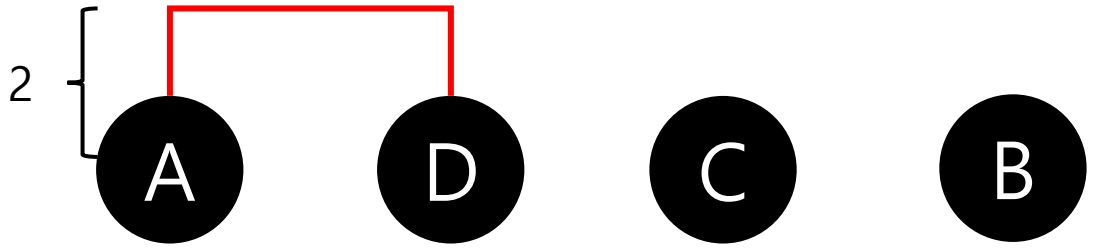


	A	B	C	D
A		20	7	2
B			10	25
C				3
D				

Hierarchical Clustering

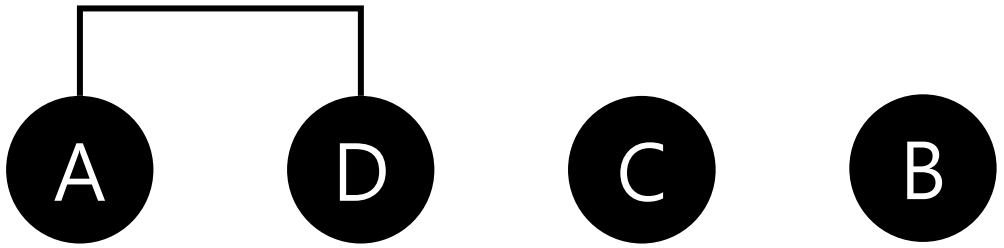
- 계층적 군집화 수행 예시
  - 모든 개체들 사이의 거리에 대한 유사도 행렬 계산

	A	B	C	D
A		20	7	2
B			10	25
C				3
D				



Hierarchical Clustering

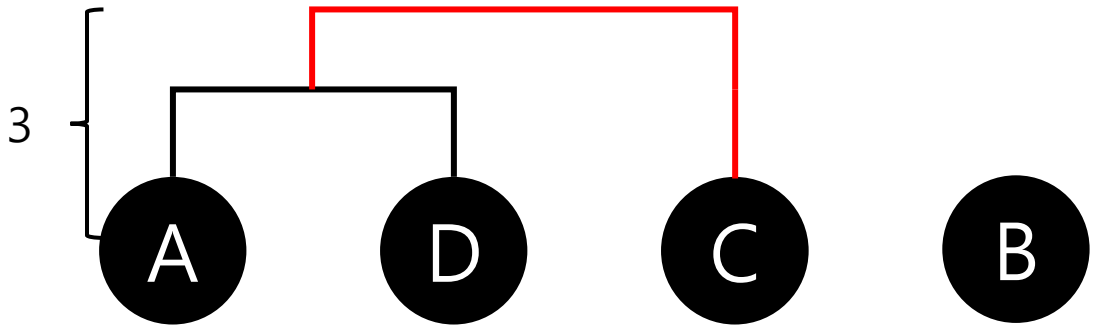
- 계층적 군집화 수행 예시
  - 모든 개체들 사이의 거리에 대한 유사도 행렬 계산
  - 거리가 인접한 관측치끼리 군집 형성
  - 유사도 행렬 업데이트



	AD	B	C	
AD		20	3	
B			10	
C				

Hierarchical Clustering

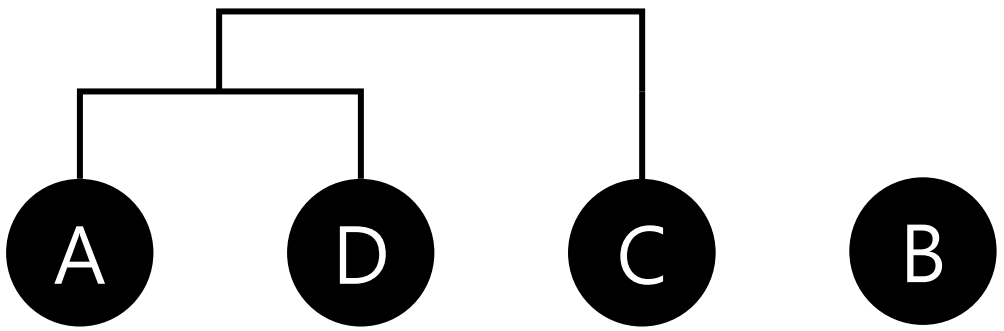
- 계층적 군집화 수행 예시
  - 모든 개체들 사이의 거리에 대한 유사도 행렬 계산
  - 거리가 인접한 관측치끼리 군집 형성
  - 유사도 행렬 업데이트



	AD	B	C	
AD		20	3	
B			10	
C				

Hierarchical Clustering

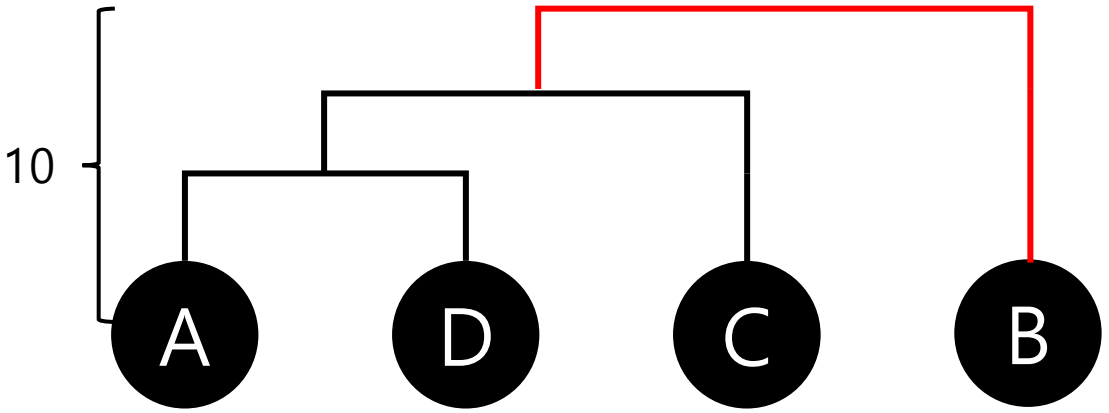
- 계층적 군집화 수행 예시
  - 모든 개체들 사이의 거리에 대한 유사도 행렬 계산
  - 거리가 인접한 관측치끼리 군집 형성
  - 유사도 행렬 업데이트
  - 위의 과정 반복



	AD C	B		
AD C		10		
B				

Hierarchical Clustering

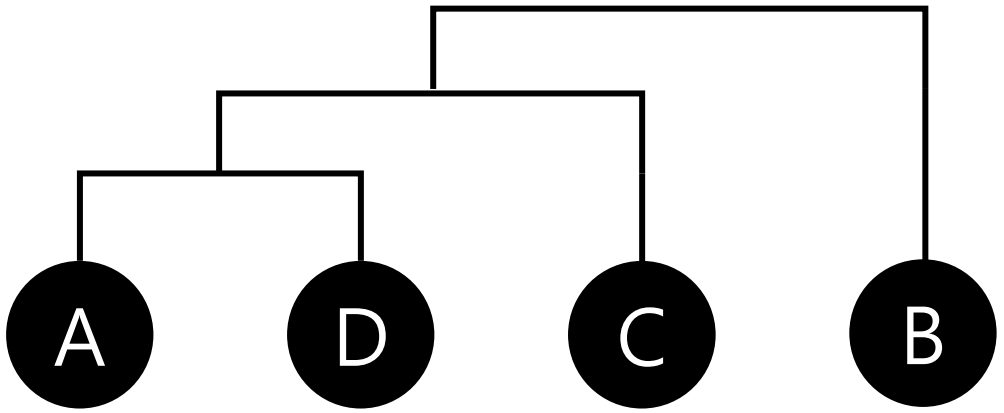
- 계층적 군집화 수행 예시
  - 모든 개체들 사이의 거리에 대한 유사도 행렬 계산
  - 거리가 인접한 관측치끼리 군집 형성
  - 유사도 행렬 업데이트
  - 위의 과정 반복



	AD C	B		
AD C		10		
B				

Hierarchical Clustering

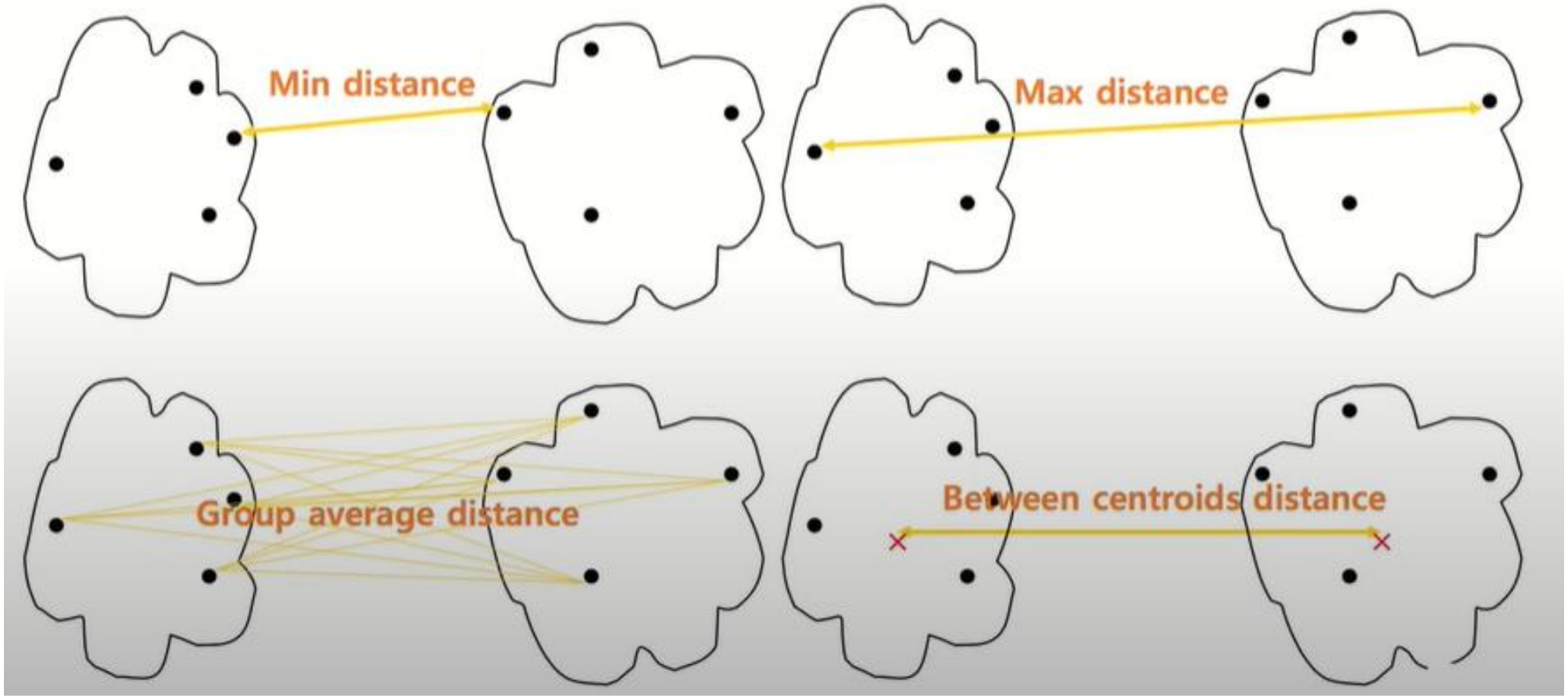
- 계층적 군집화 수행 예시
  - 최종 결과



	AD CB			
AD CB				

Hierarchical Clustering

- 핵심 수행 절차: 두 군집 사이의 유사성/거리 측정
  - ✓ Min(단일 연결), max(완전 연결), group average(평균 연결), between centroid, Ward's, …



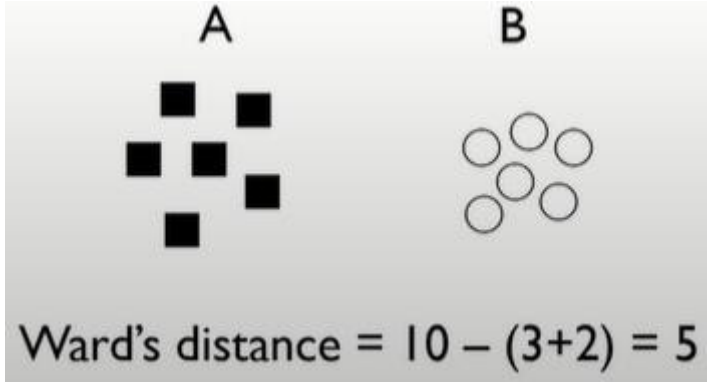
Hierarchical Clustering

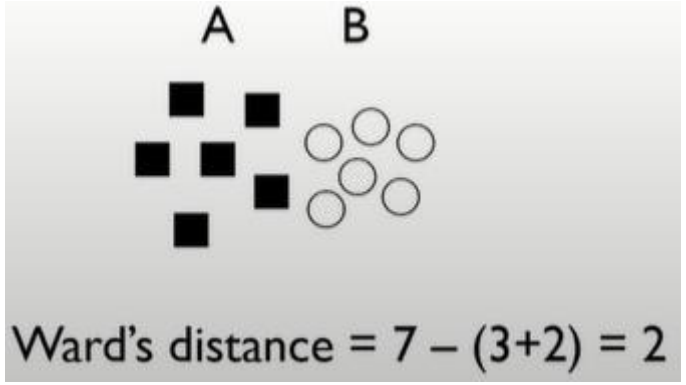
- Ward's method: Distance between two clusters, A and B, is how much the sum of squares will increase when they are merged

$$\text{Ward Distance} = \sum_{i \in A \cup B} ||x_i - m_{A \cup B}||^2 - \left\{ \sum_{i \in A} \|x_i - m_A\|^2 + \sum_{i \in B} \|x_i - m_B\|^2 \right\}$$

m_A is the center of cluster

Ward's distance can be considered as the merging cost of combining the clusters A and B





K-Means Clustering

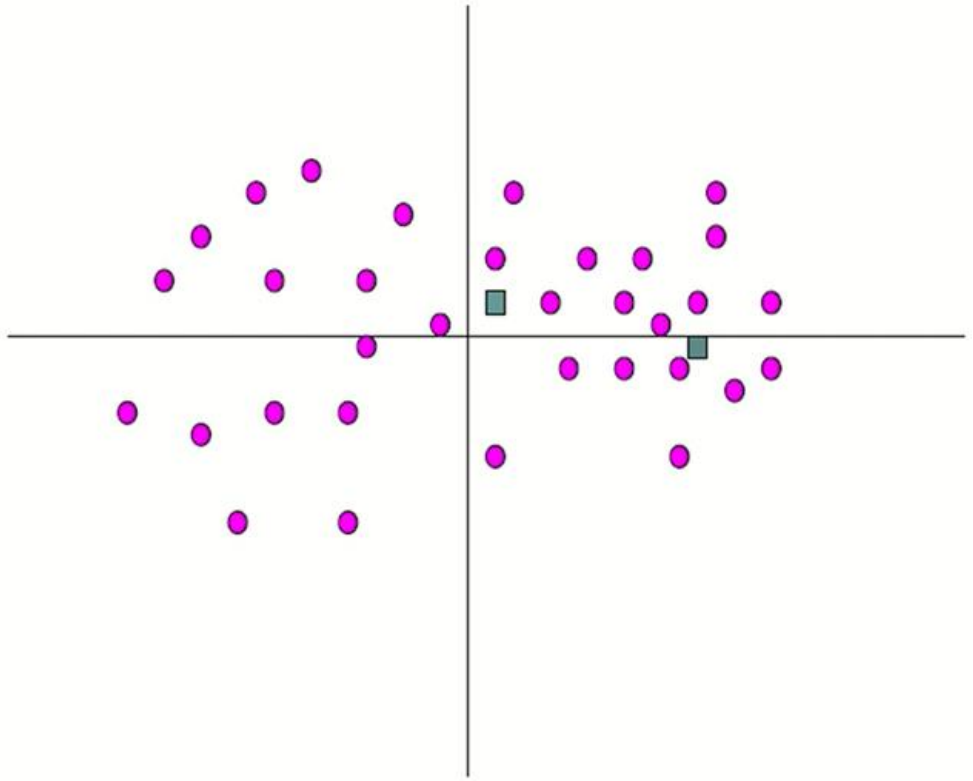
- K-Means Clustering
  - 대표적인 분리형 군집화 알고리즘
    - ✓ 각 군집은 하나의 중심(centroid)을 가짐
    - ✓ 각 개체는 가장 가까운 중심에 할당되며, 같은 중심에 할당된 개체들이 모여 하나의 군집을 형성
    - ✓ 사전에 군집의 수 K가 정해져야 알고리즘을 실행할 수 있음

$$X = C_1 \cup C_2 \cdots C_k, C_i \cap C_j = \emptyset, \qquad i \neq j$$

$$\underset{c}{\operatorname{argmax}} \sum_{i=1}^{K} \sum_{x_j \in C_i} \|x_j - c_i\|^2$$

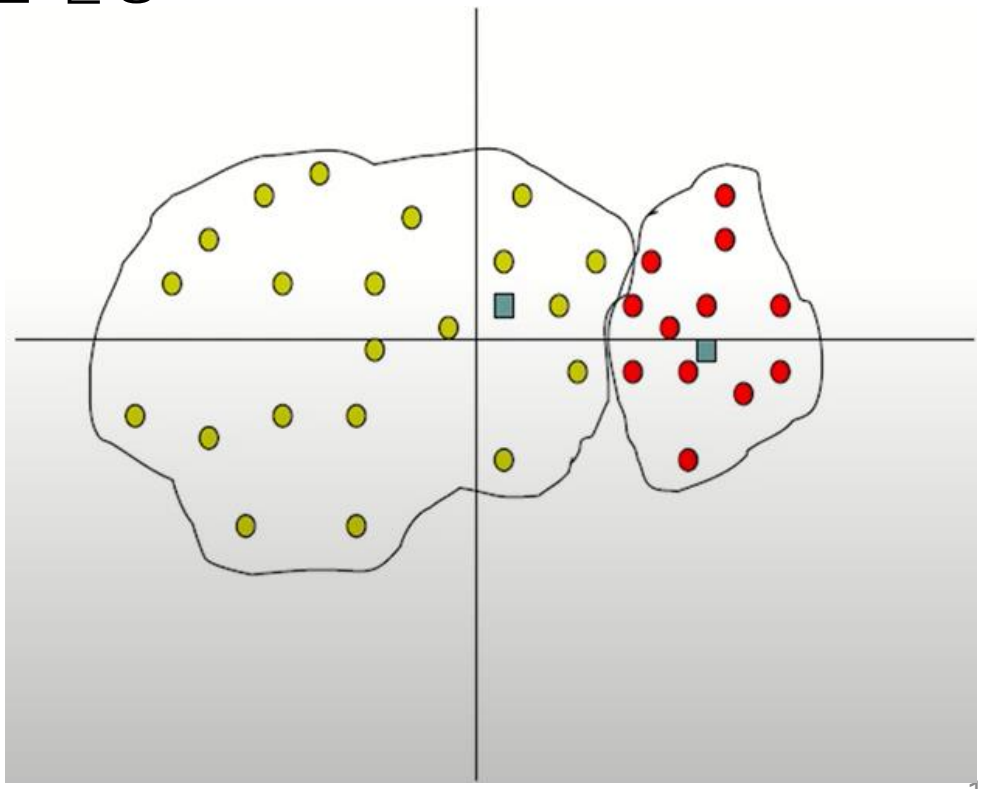
Example of K-Means Clustering

- Example (K=2)
  1. 2개의 중심을 임의로 생성



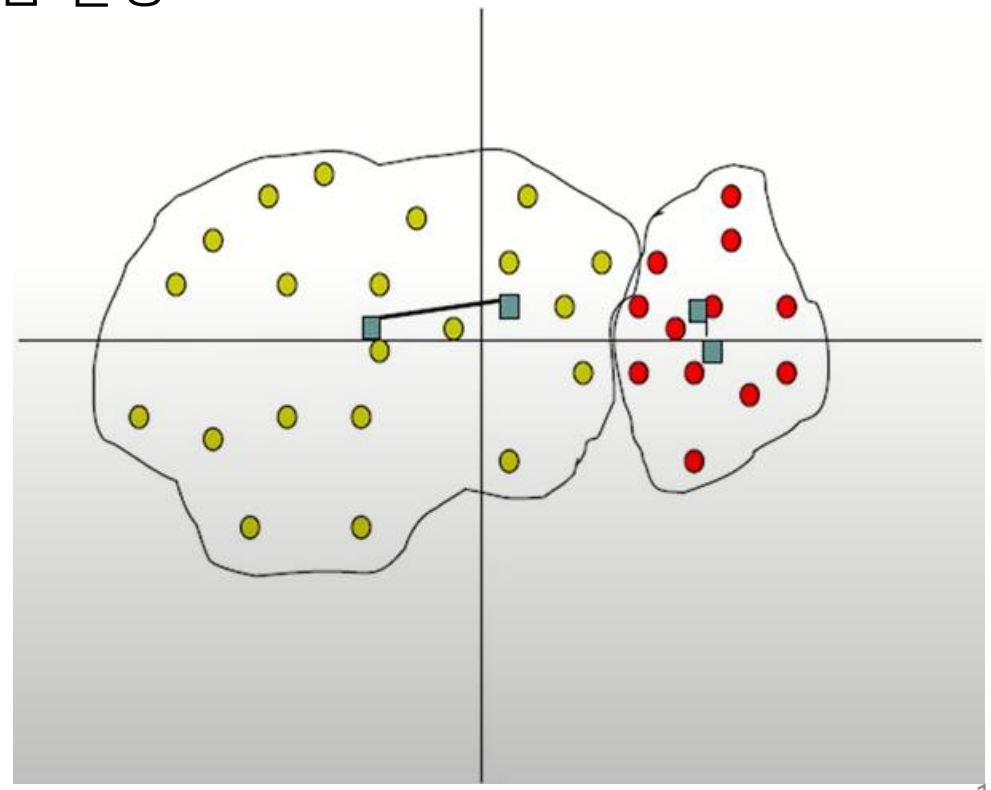
Example of K-Means Clustering

- Example (K=2)
  1. 2개의 중심을 임의로 생성
  2. 생성된 중심을 기준으로 모든 관측치에 군집 할당



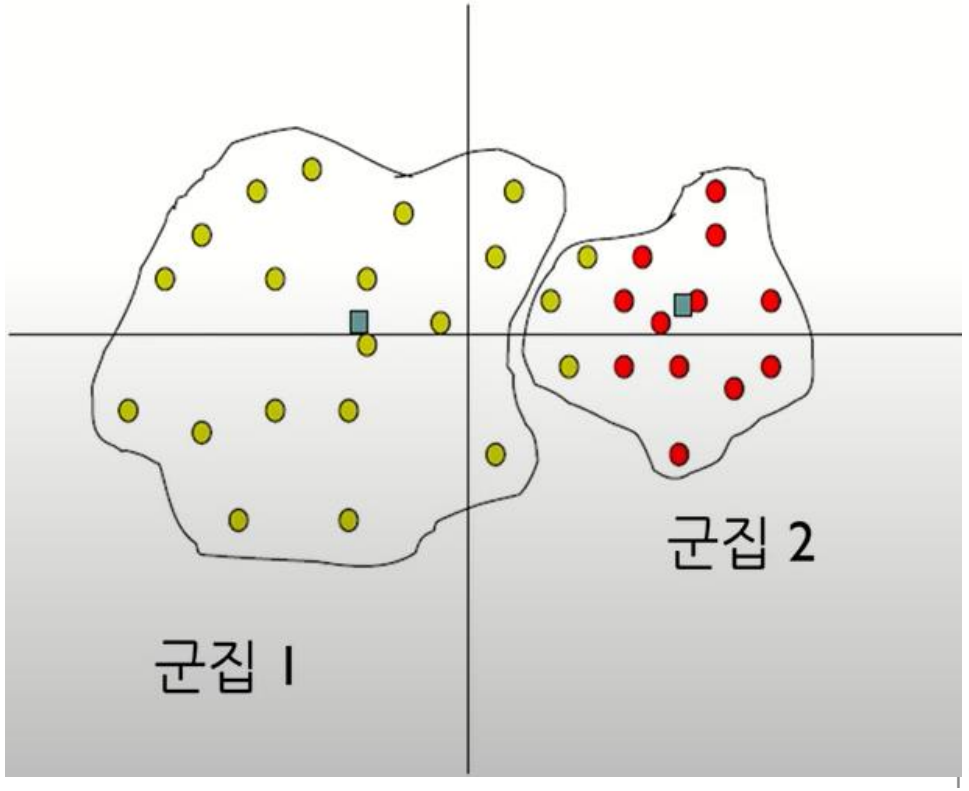
Example of K-Means Clustering

- Example (K=2)
  1. 2개의 중심을 임의로 생성
  2. 생성된 중심을 기준으로 모든 관측치에 군집 할당
  3. 각 군집의 중심을 다시 계산



Example of K-Means Clustering

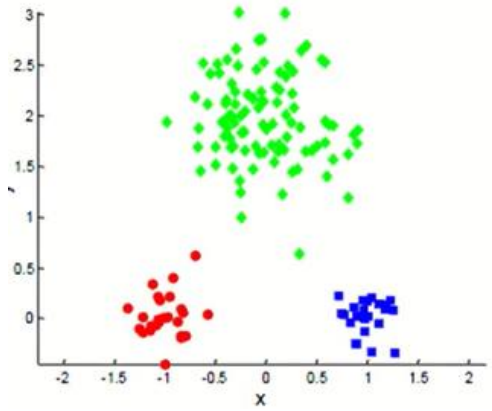
- Example (K=2)
  1. 2개의 중심을 임의로 생성
  2. 생성된 중심을 기준으로 모든 관측치에 군집 할당
  3. 각 군집의 중심을 다시 계산
  4. 중심이 변하지 않을 때까지
     위의 과정을 반복

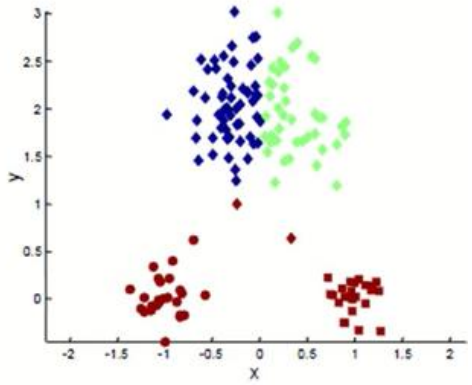


Procedure of K-Means Clustering

- K-Means Clustering
  1. 초기 중심을 K개 임의로 생성
  2. 개별 관측치로부터 각 중심까지의 거리를 계산 후, 가장 가까운 중심이 이루는 군집에 관측치 할당
  3. 각 군집의 중심을 다시 계산
  4. 중심이 변하지 않을 때까지 2,3의 과정을 반복

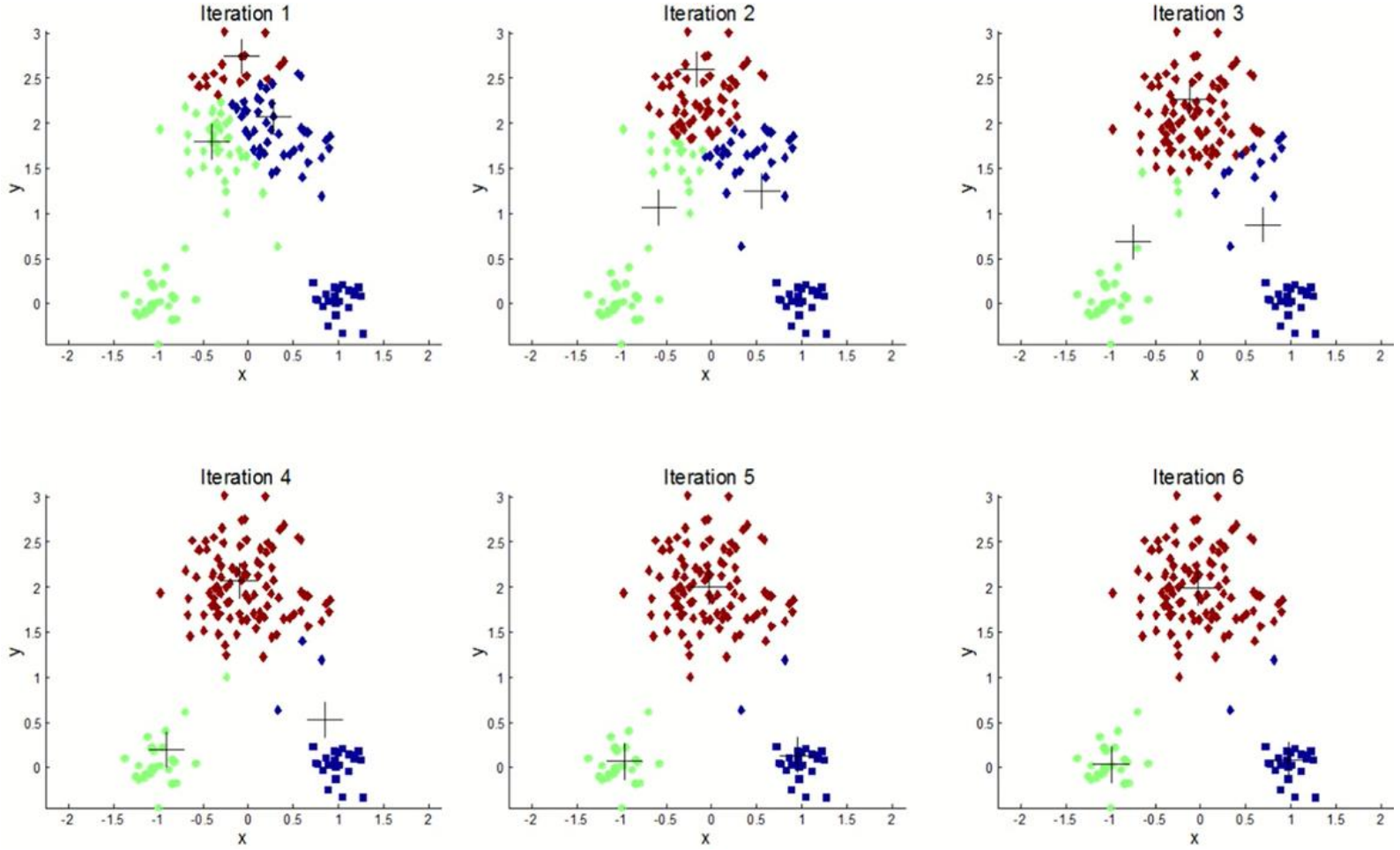
  ✓ 초기 중심은 종종 무작위로 설정됨: 군집화 결과가 초기 중심 설정에 따라 다르게 나타나는 경우가 발생할 수 있음





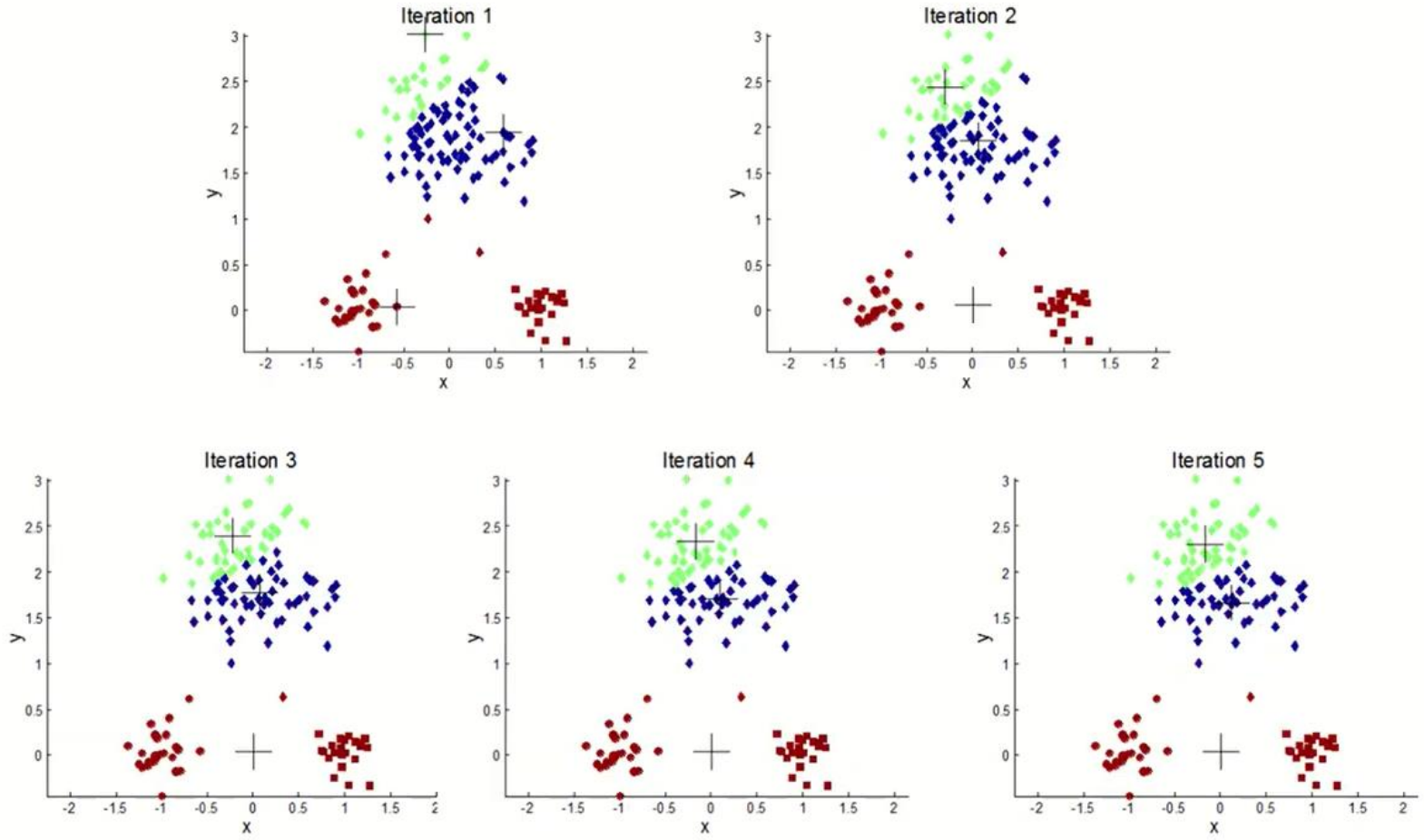
K-Means Clustering

- 초기 중심 설정이 최종 결과에 어떤 영향을 미치는가?
  - ➢ **바람직한 결과**



K-Means Clustering

- 초기 중심 설정이 최종 결과에 어떤 영향을 미치는가?
  - ➢ **바람직하지 않은 결과**

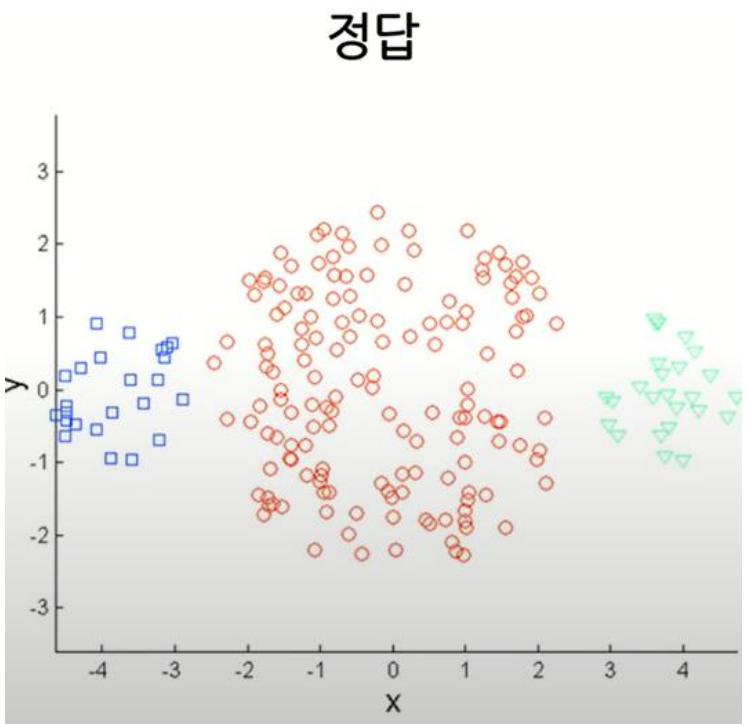


K-Means Clustering

- 무작위 초기 중심 설정의 위험을 피하고자 다양한 연구 존재
  - 반복적으로 수행하여 가장 여러 번 나타나는 군집을 사용
  - 전체 데이터 중 일부만 샘플링하여 계층적 군집화를 수행한 뒤 초기 군집 중심 설정
  - 데이터 분포의 정보를 사용하여 초기 중심 설정
  - 하지만 많은 경우 초기 중심 설정이 최종 결과에 큰 영향을 미치지는 않음

K-Means Clustering

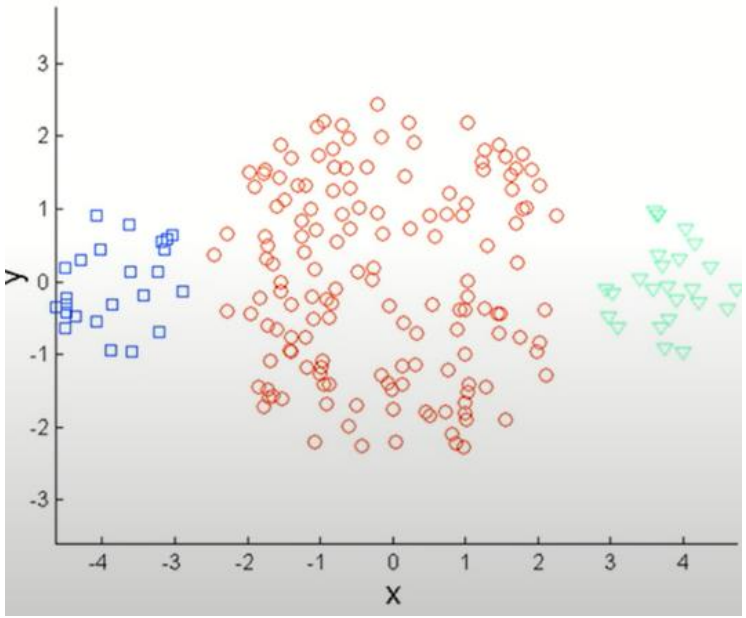
- K-Means Clustering의 문제점
  - 문제점 I : **서로 다른 크기의** 군집을 찾아내지 못함



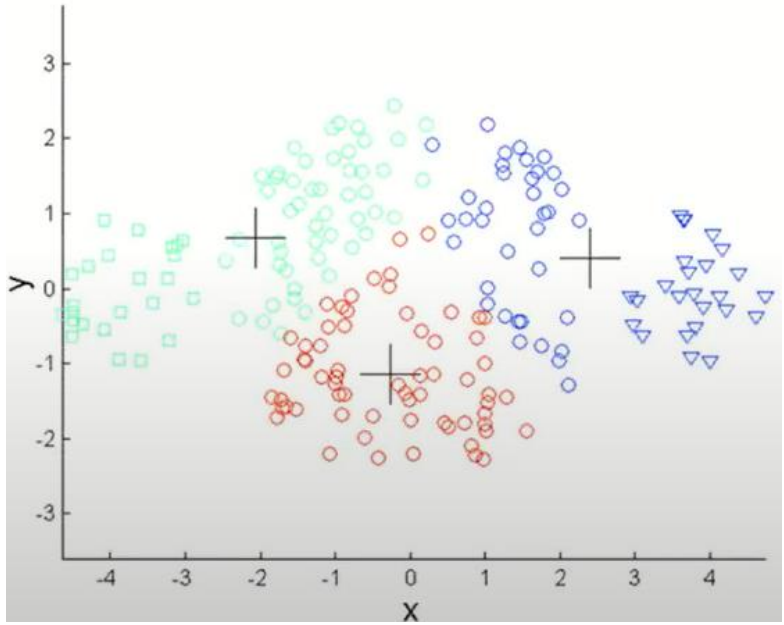
K-Means Clustering

- K-Means Clustering의 문제점
  - 문제점Ⅰ : **서로 다른 크기의** 군집을 찾아내지 못함

정답

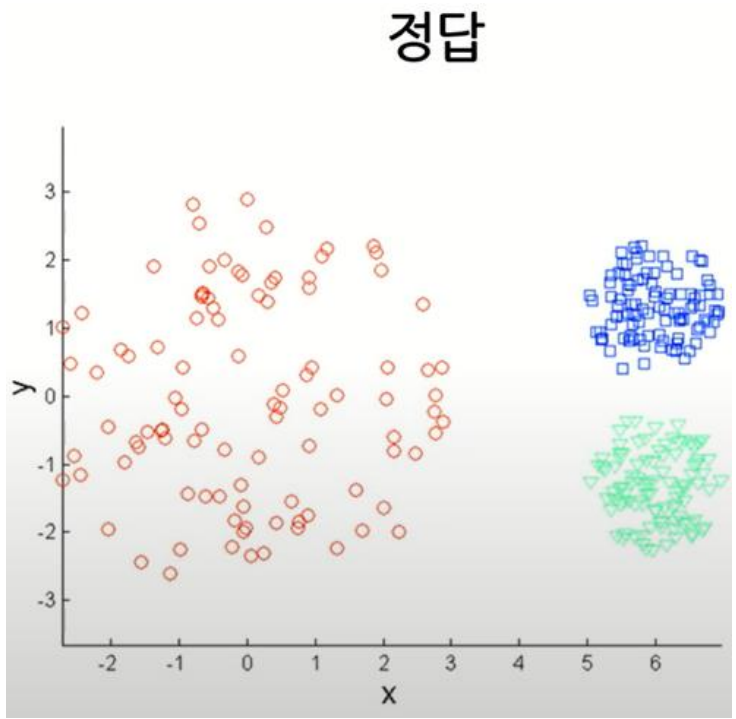


K-평균 군집화 결과



K-Means Clustering

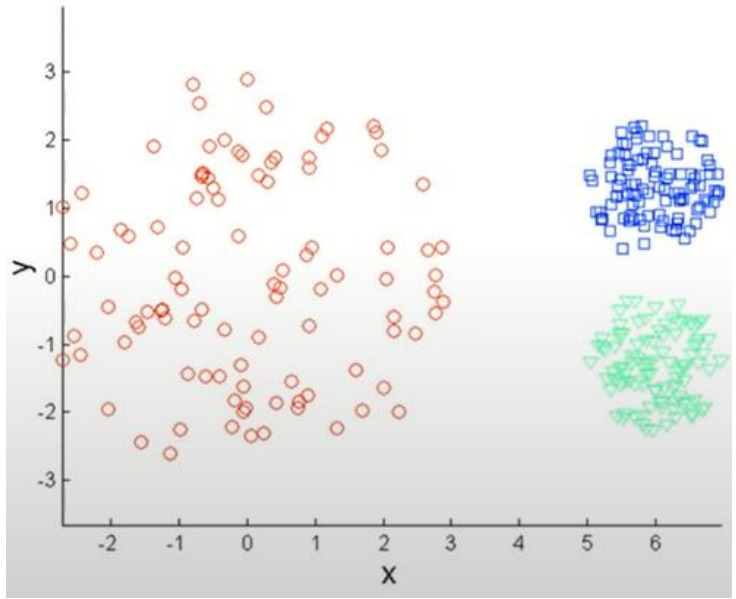
- K-Means Clustering의 문제점
  - 문제점 II: **서로 다른 밀도의** 군집을 찾아내지 못함



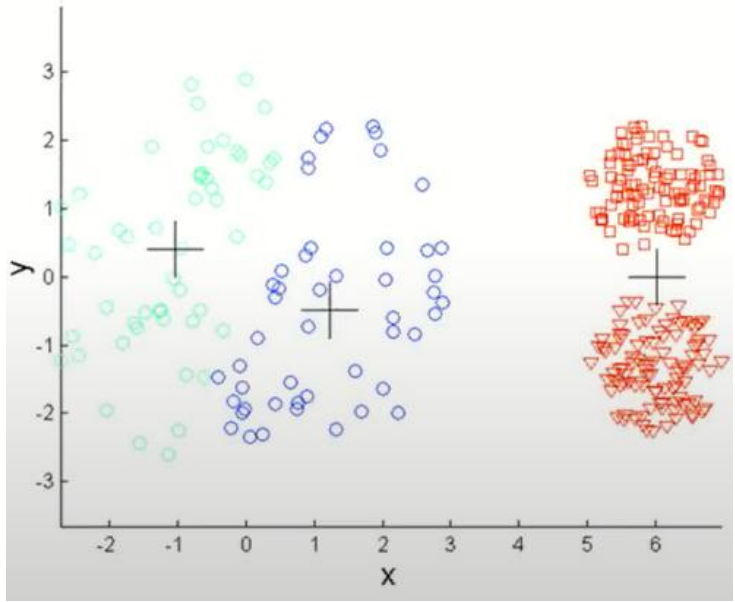
K-Means Clustering

- K-Means Clustering의 문제점
  - 문제점 II : **서로 다른 밀도의** 군집을 찾아내지 못함

정답

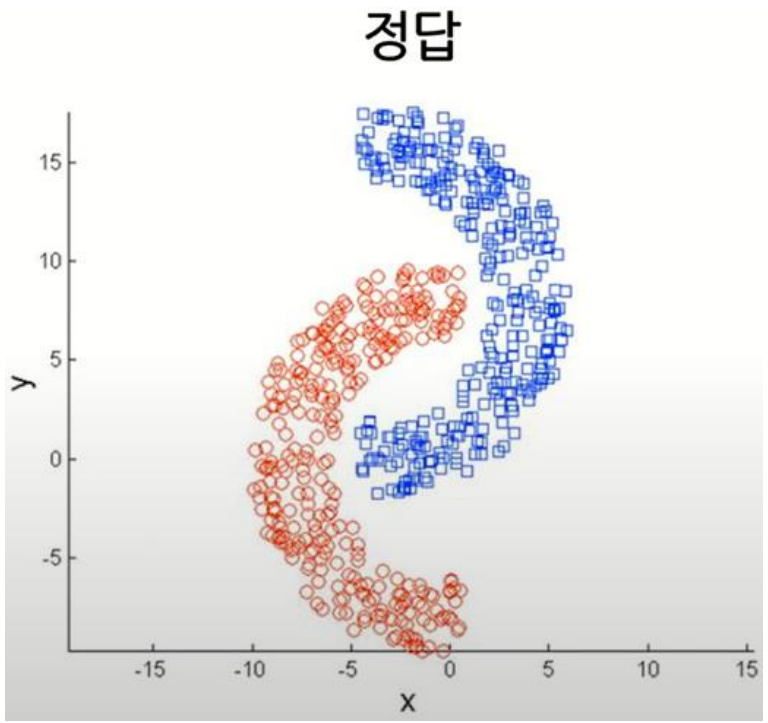


K-평균 군집화 결과



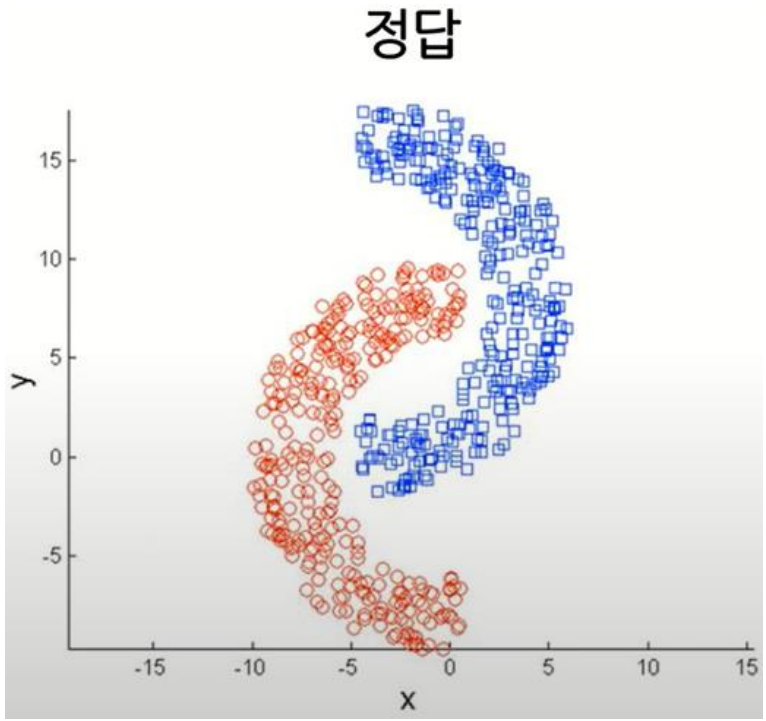
K-Means Clustering

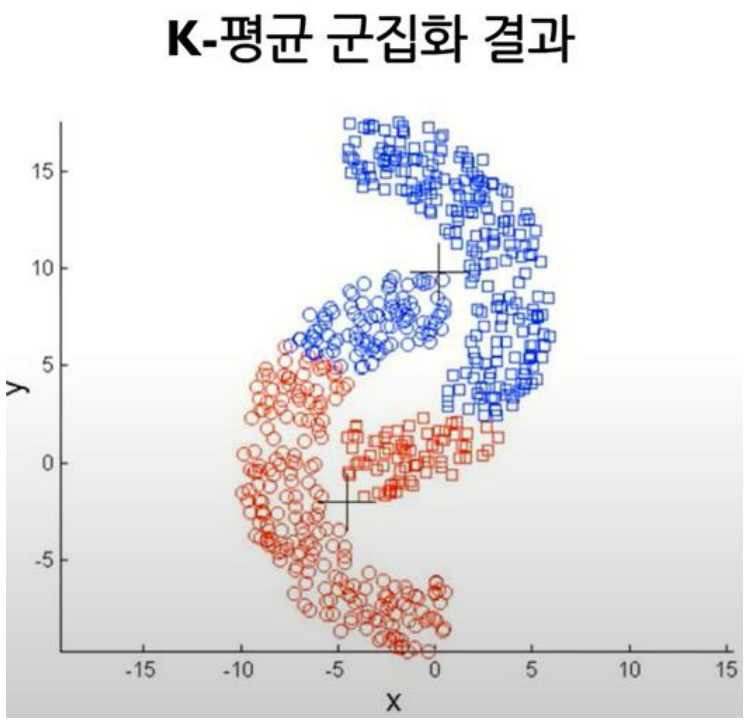
- K-Means Clustering의 문제점
  - 문제점 Ⅲ: **지역적 패턴이 존재**하는 군집을 판별하기 어려움



K-Means Clustering

- K-Means Clustering의 문제점
  - 문제점 Ⅲ : **지역적 패턴이 존재**하는 군집을 판별하기 어려움





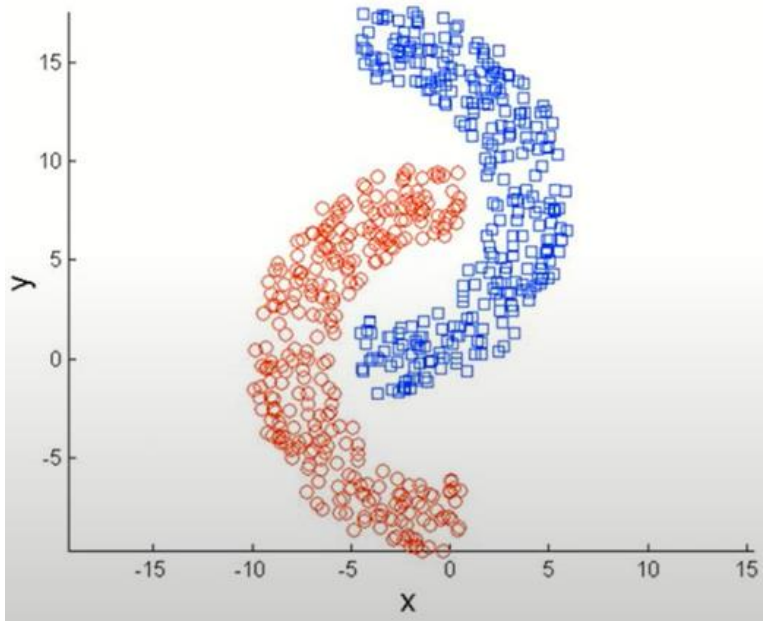
K-Means Clustering

- K-Means Clustering의 문제점

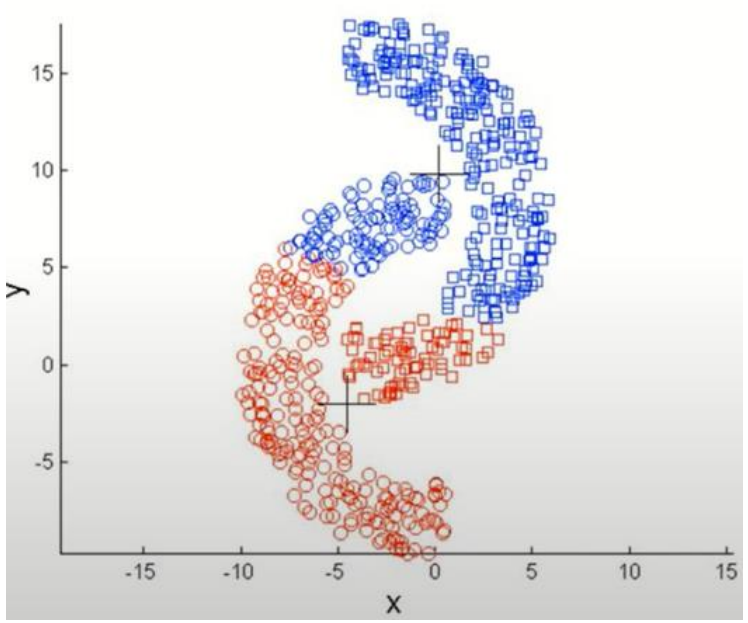
※ **Geodesic Distance**

  - 문제점 Ⅲ : **지역적 패턴이 존재**하는 군집을 판별하기 어려움

정답

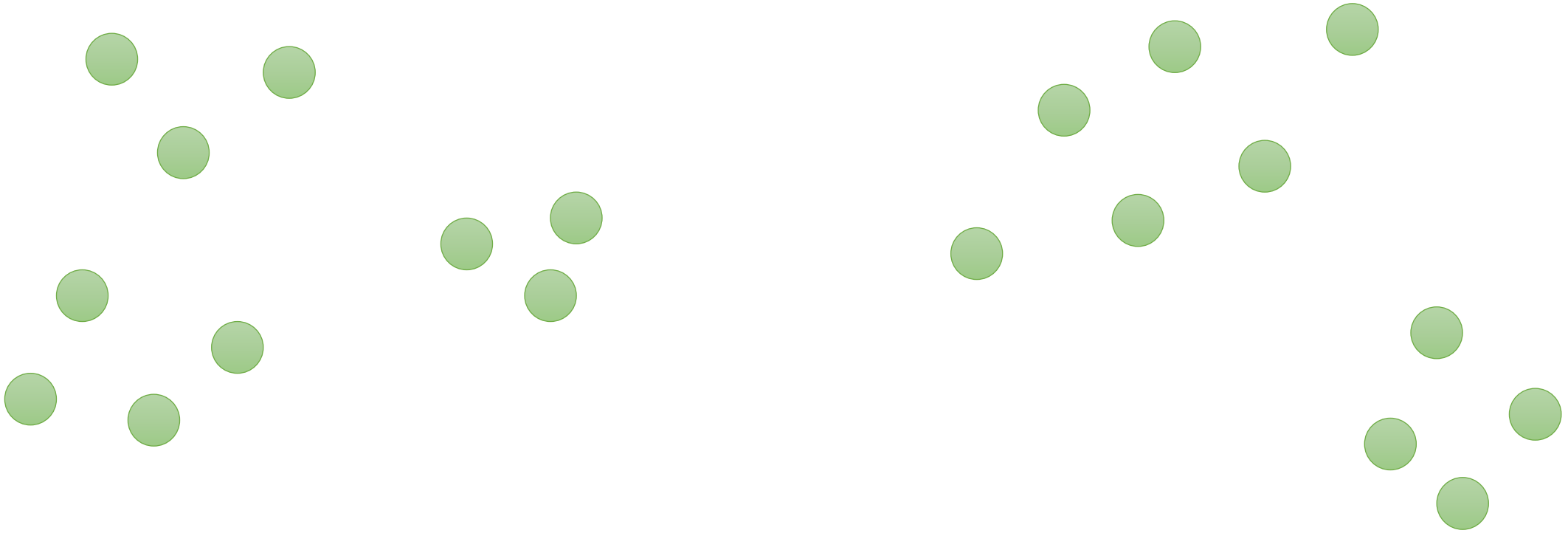


K-평균 군집화 결과



Clustering: Determination the optional number of cluster

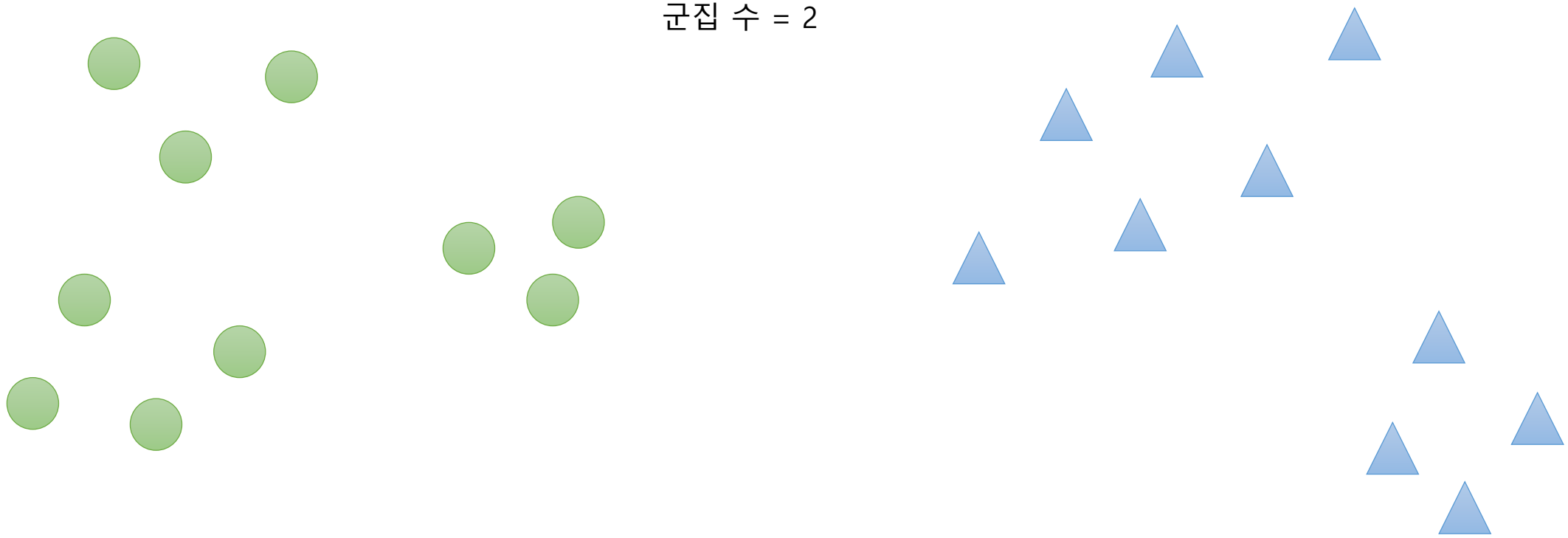
- 어떻게 최적의 군집 수를 결정?
  - ex) 20개의 관측치가 존재할 때, 최적의 군집 수는?



Clustering: Determination the optional number of cluster

- 어떻게 최적의 군집 수를 결정?
  - ex) 20개의 관측치가 존재할 때, 최적의 군집 수는?

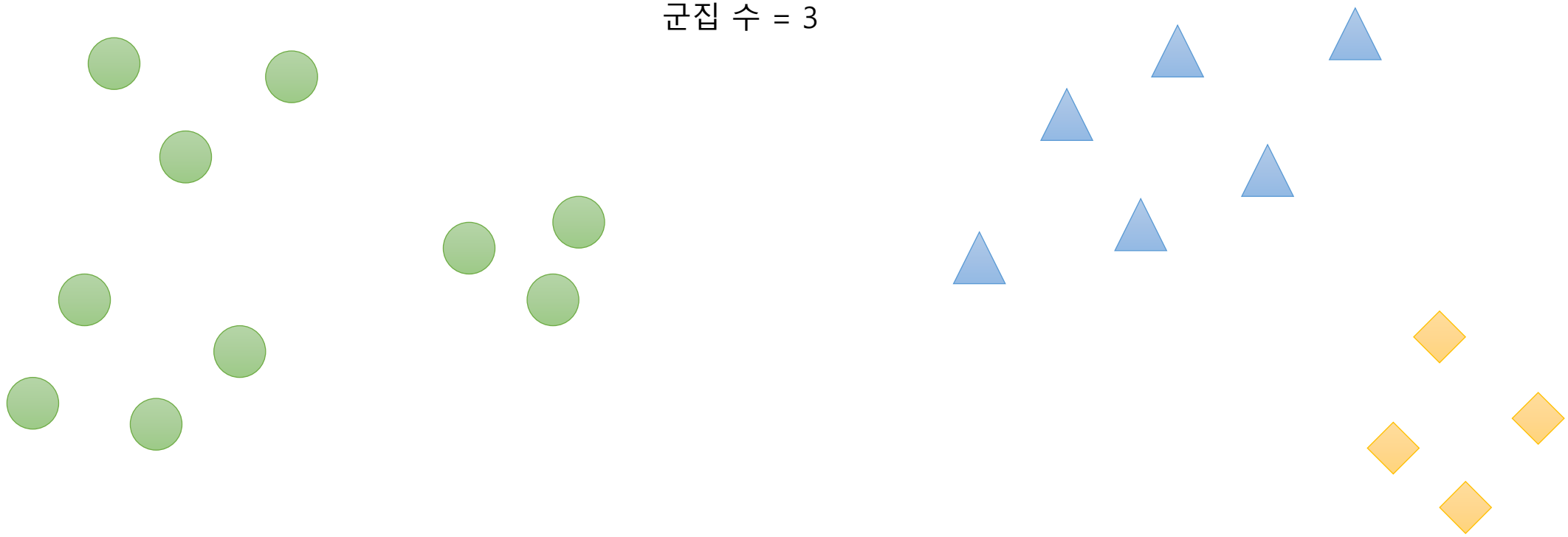
군집 수 = 2



Clustering: Determination the optional number of cluster

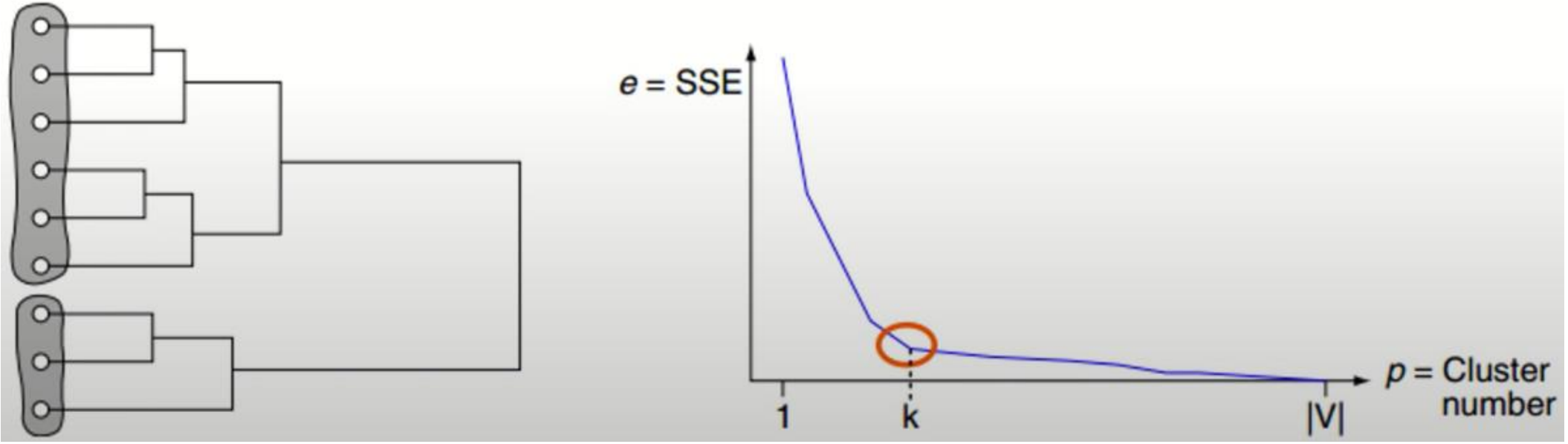
- 어떻게 최적의 군집 수를 결정?
  - ex) 20개의 관측치가 존재할 때, 최적의 군집 수는?

군집 수 = 3



Clustering: Determination the optional number of cluster

- 어떻게 최적의 군집 수를 결정?
  - 다양한 군집 수에 대해 성능 평가 지표를 도식화하여 최적의 군집 수 선택
  - Elbow point에서 최적 군집 수가 결정되는 경우가 일반적



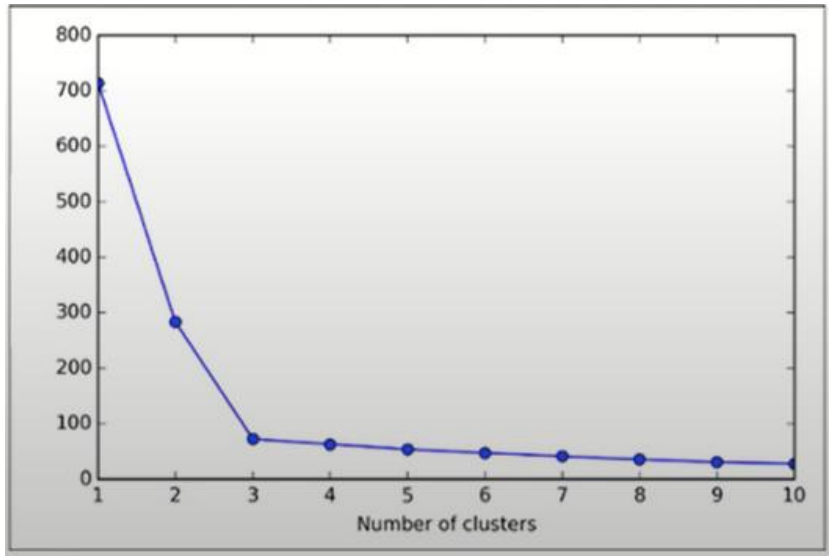
Clustering: measure and evaluate the result

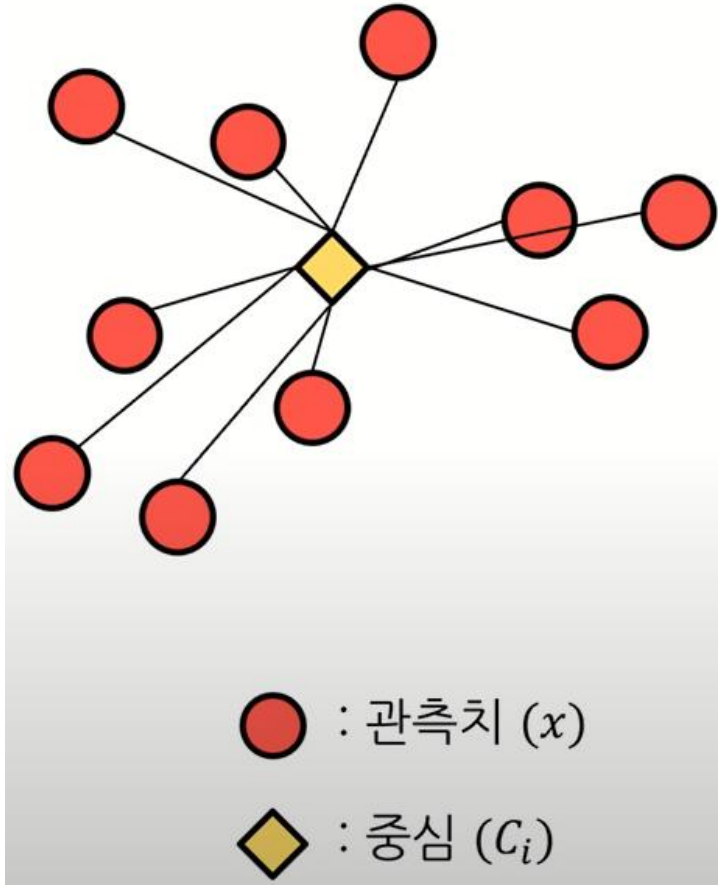
- 어떻게 군집화 결과를 측정/평가할 것인가?
- 분류 알고리즘처럼 모든 상황에 적용 가능한 평가 지표 부재
  - 내부 평가지표
    - ✓ Dunn Index, Silhouette, Sum of Squared Error, …
  - 외부 평가 지표
    - ✓ Rand index, Jaccard Coefficient, Folks and Mallows Index, …

Clustering: measure and evaluate the result

- 군집화 평가 지표 Ⅰ: Sum of Squared Error (SSE)

$$SSE = \sum_{i=1}^{K} \sum_{x \in c_i} dist(x, c_i)^2$$



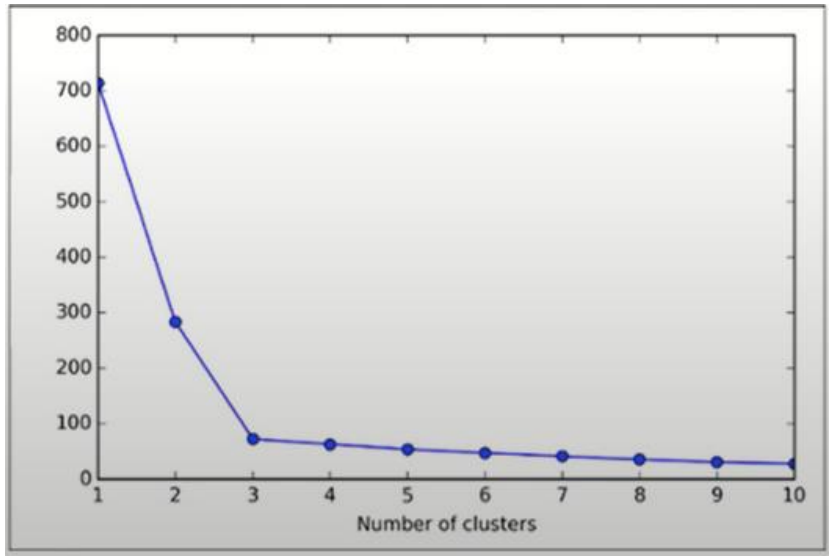


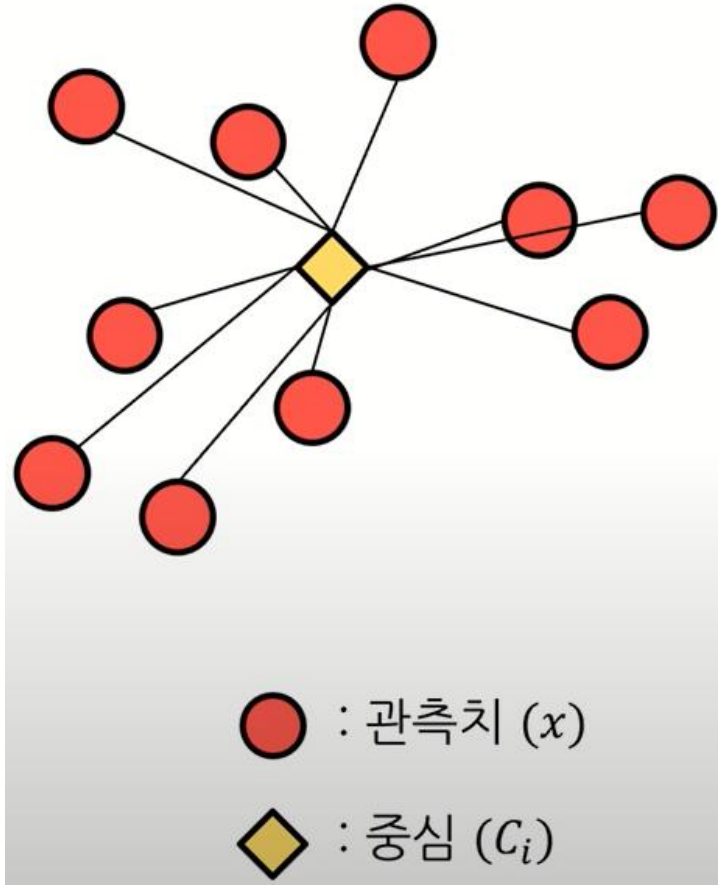
Clustering: measure and evaluate the result

- 군집화 평가 지표 Ⅰ: Sum of Squared Error (SSE)

But, 군집 간의 거리는 고려 X

$$SSE = \sum_{i=1}^{K} \sum_{x \in c_i} dist(x, c_i)^2$$





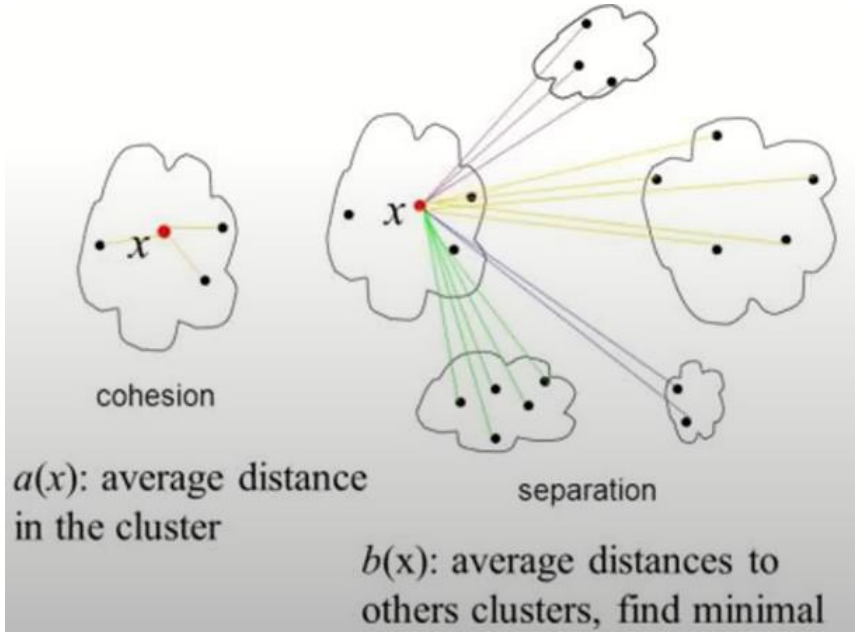
Clustering: measure and evaluate the result

- 군집화 평가 지표 II: Silhouette 통계량
  - a(i): 관측치 i로부터 같은 군집 내에 있는 모든 다른 개체들 사이의 평균 거리
  - b(i): 관측치 i로부터 다른 군집 내에 있는 개체들 사이의 평균 거리 중 최솟값
  - 일반적으로 $\bar{S}$의 값 0.5보다 크면 군집 결과가 타당하다고 볼 수 있음
  - -1에 가까우면 군집이 전혀 되지 않음

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

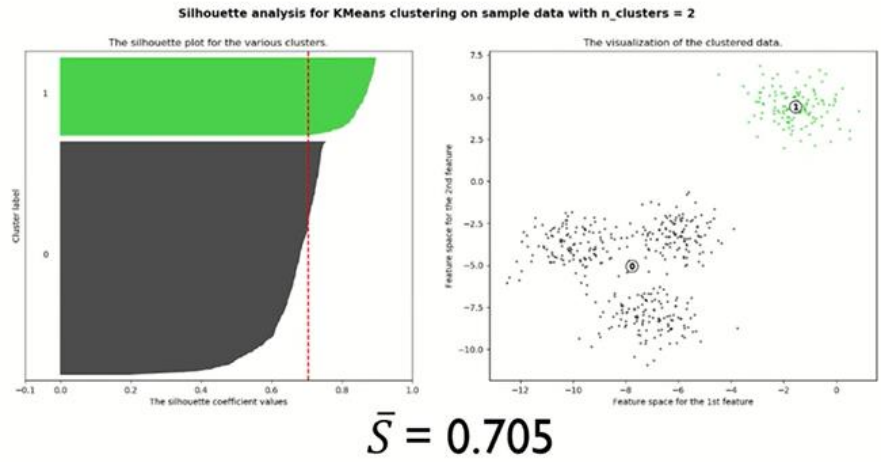
$$-1 \leq s(\mathrm{i}) \leq 1$$

$$\bar{S} = \frac{1}{n}\sum_{i=1}^{n} S(i)$$

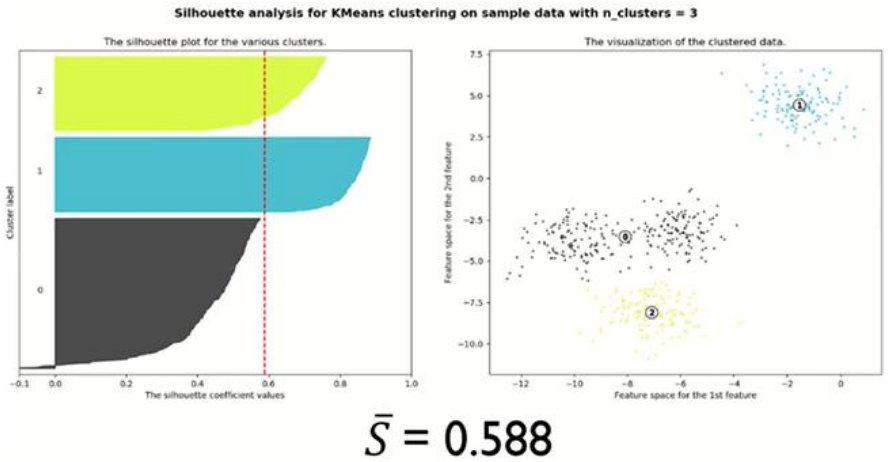


Clustering: measure and evaluate the result

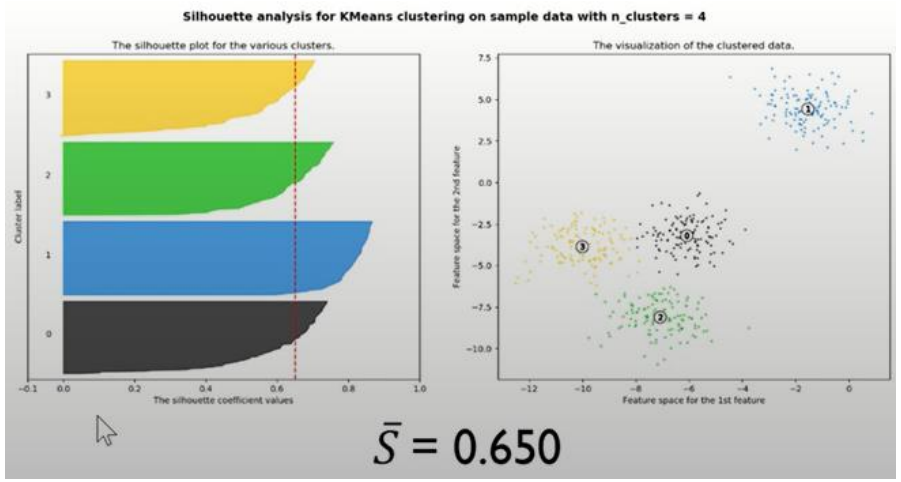
$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}'}$$



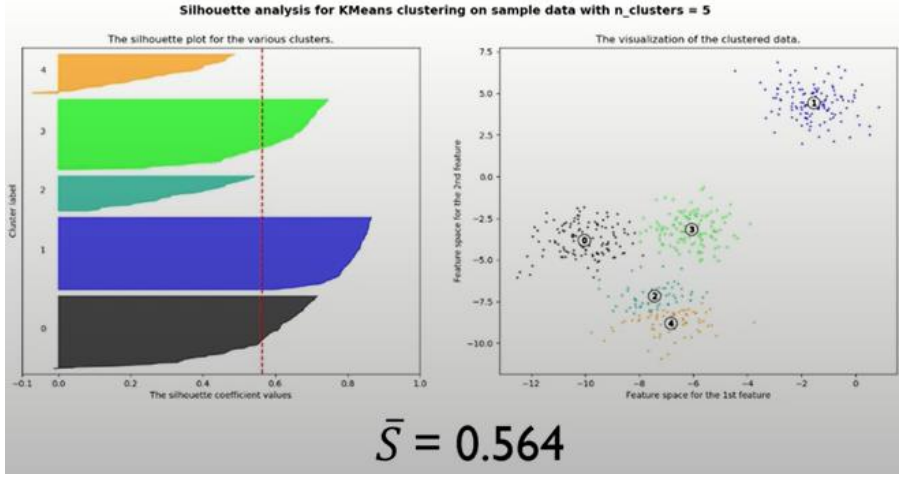
$\bar{S} = 0.705$



$\bar{S} = 0.588$



$\bar{S} = 0.650$



$\bar{S} = 0.564$

Thank you