

TEXT MINING for PRACTICE

Python을 활용한 비정형 데이터 분석 - WEEK 08

전병진 FINGEREDMAN (fingeredman@gmail.com)

Part 7.

단어 & 문서 구조화

문서 유사도 (Document Similarity)

유사도 (Similarity)

- ▶ 서로 다른 두 객체 사이의 공통점을 통해 서로 공유하는 속성의 수에 따라 증가하는 유사한 정도
- ▶ 서로 공유하는 속성은 그 기능에 따라서 매우 많이 존재할 수도 있으며 없을 수도 있음
- ▶ 각 속성은 서로간에 영향을 주지 않도록 각각 독립적으로 존재해야함

문서
D1 = 텍스트 마이닝은 비정형 데이터에 대해 다룹니다. D2 = 비정형 데이터는 정형 데이터에 비해 복잡하고 어렵습니다. D3 = 오늘은 단어주머니에 대해 배웁니다.

↓

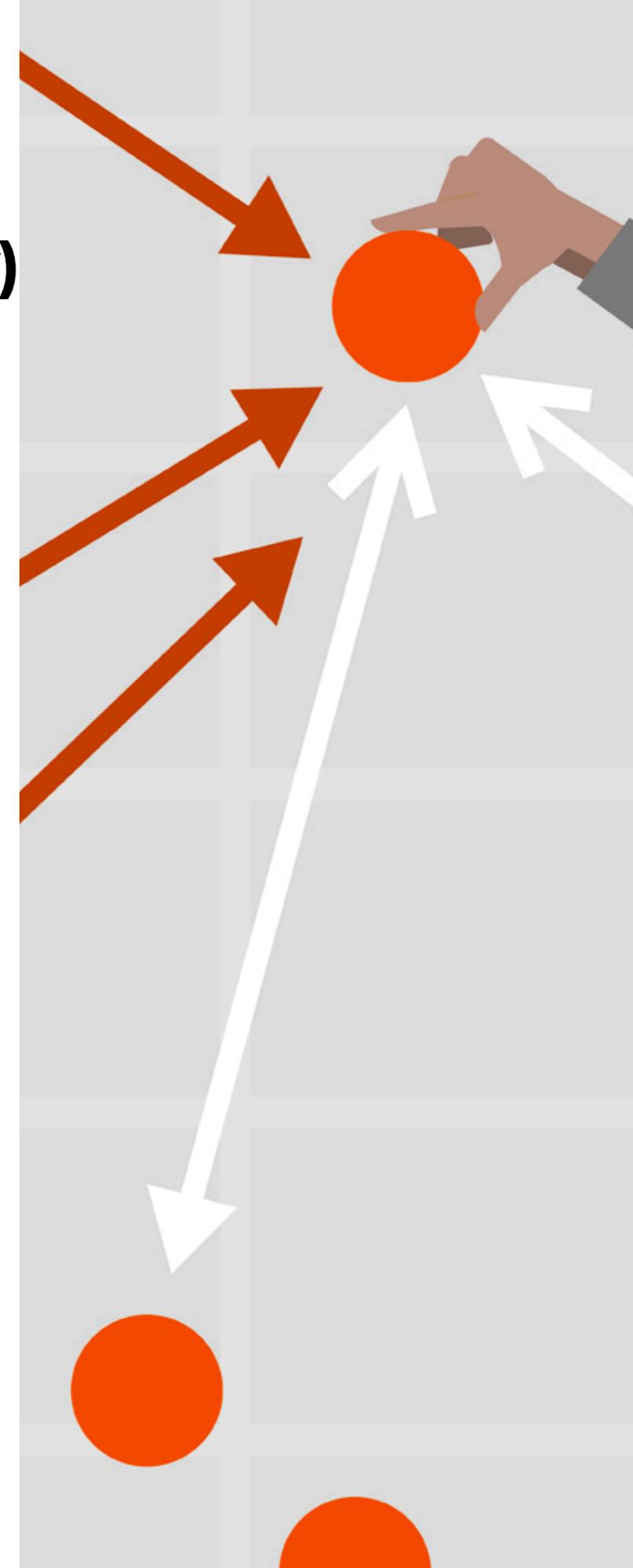
단어주머니
D1 = ["텍스트", "마이닝", "은", "비정형", "데이터", "에", "대해", "다루", "ㅂ니다", "."] D2 = ["비정형", "데이터", "는", "정형", "데이터", "에", "비해", "복잡하", "고", "어렵", "습니다."] D3 = ["오늘", "은", "단어주머니", "에", "대해", "배우", "ㅂ니다", "."]

↓

단어주머니(불용어 제거 후)
D1 = ["텍스트", "마이닝", "비정형", "데이터", "다루"] D2 = ["비정형", "데이터", "정형", "데이터", "복잡하", "어렵"] D3 = ["오늘", "단어주머니", "배우"]

↓

D1, D2, D3 문서는 각각 얼마나 유사한가?



문서 유사도 (Document Similarity)

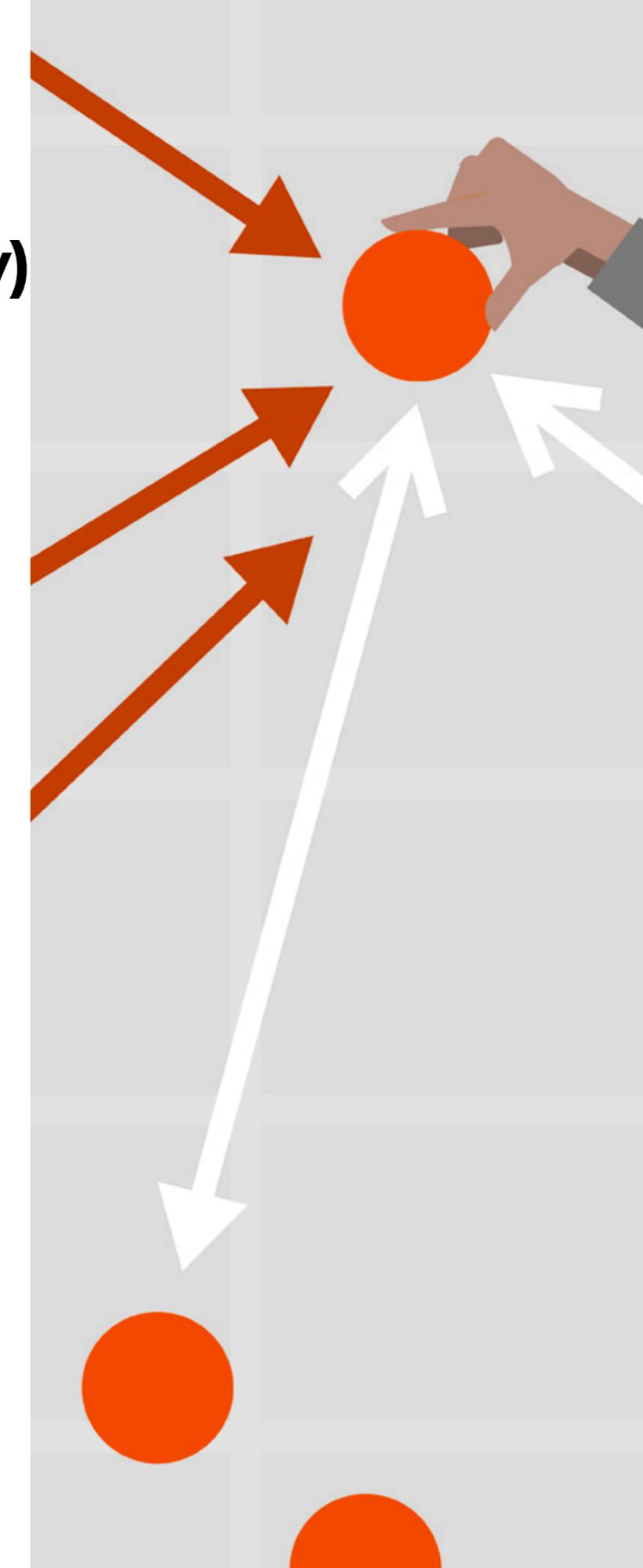
텍스트 문서 간의 유사도

- ▶ 문서 간의 유사도는 텍스트 데이터 검색, 시각화, 필터링, 정렬 등 다양한 분야에 활용됨
- ▶ 문서의 유사도를 측정하기 위한 다양한 방법론이 제시되어 활용되고 있음

유사도 측정방법

- ▶ 유사도 측정 방법은 벡터 사이의 유사도를 측정하는 방식과 동일함 (e.g., cosine similarity)
- ▶ 유사도를 측정하는 척도는 정해진 것이 없으며, 문헌에 따라 가장 적절한 방법을 판단하여 결정함

Similarity Class	Syntax Analysis	Lexical Analysis	Semantic Analysis
Properties			
Similarity Types	Narrow Similarity	Narrow Similarity	Narrow Similarity
	Multi-feature Similarity		Multi-feature Similarity
		Broad Similarity	



문서 유사도 (Document Similarity)

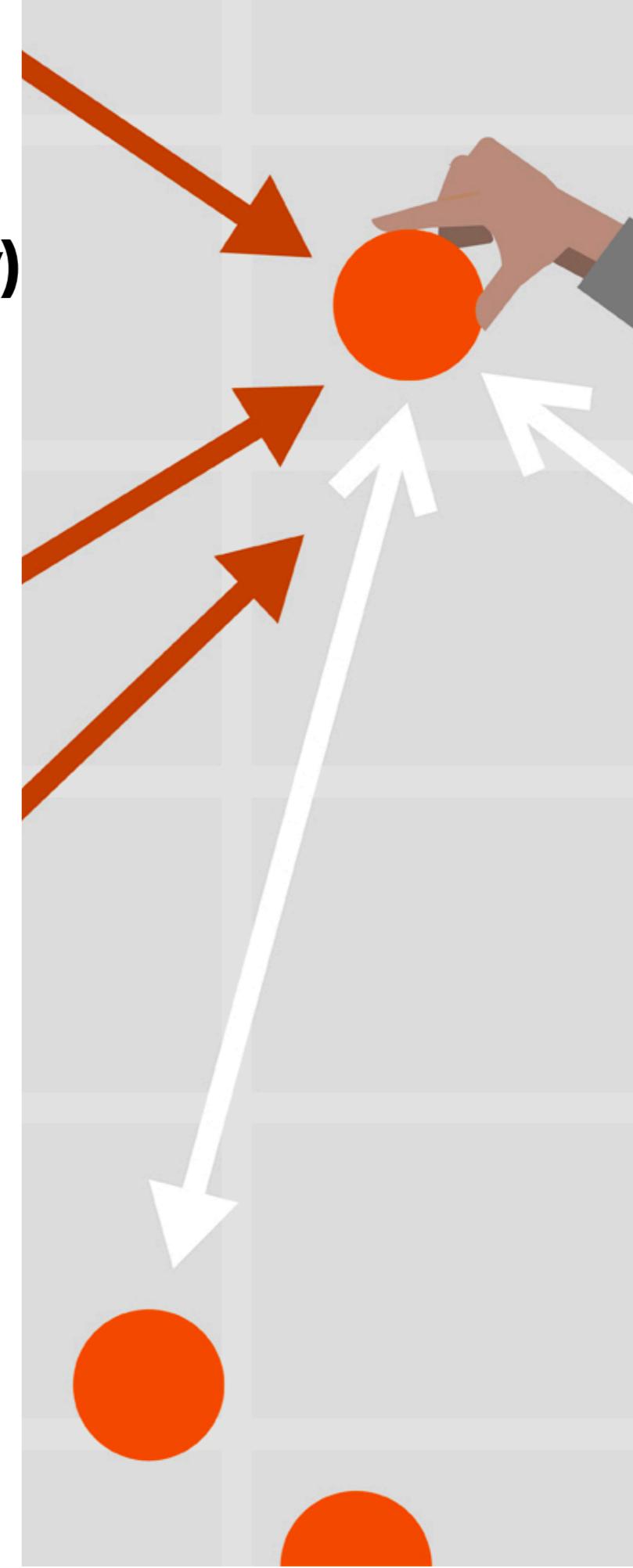
단어를 활용한 유사도 (Term Similarity)

- ▶ 두 개의 문서가 동일한 단어를 포함하는 정도에 따라서 문서의 유사도를 측정하는 방법
- ▶ 또는 그 이상의 특성을 도출해 문서를 표현하는 하나의 피처 (feature)로 활용 가능함
 - 문서의 길이
 - 문서간에 공통된 단어의 수
 - 특정 조건이 공통적으로 출현하는지 여부
 - 문서에 각 단어가 출현하는 횟수

단어	문서 1	문서 2	문서 3	문서 4	문서 5
텍스트	1	0	0	0	1
마이닝	1	0	0	0	1
비정형	1	1	0	1	2
데이터	1	2	0	4	0
다루다	1	0	0	2	1
정형	0	1	0	1	1
복잡하다	0	1	0	0	1
어렵다	0	1	0	0	0
오늘	0	0	1	0	0
단어주머니	0	0	1	2	1
배우다	0	0	1	0	1



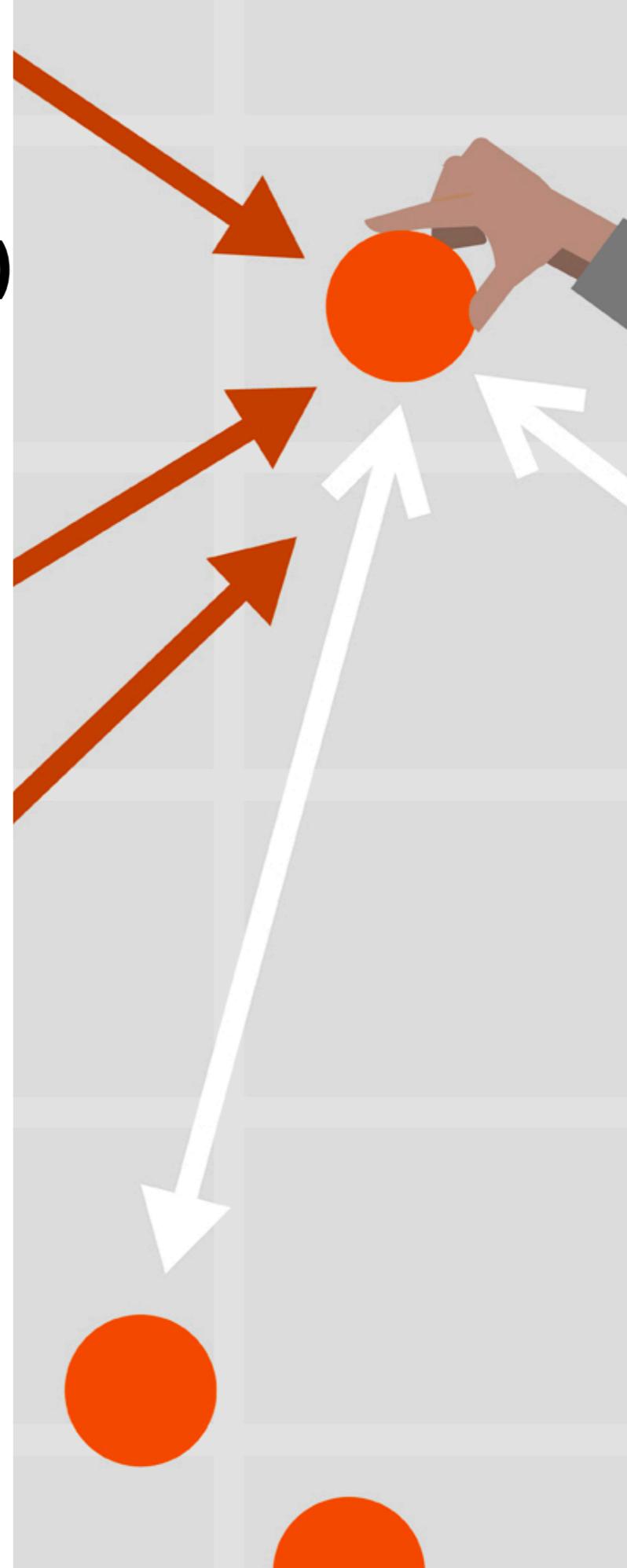
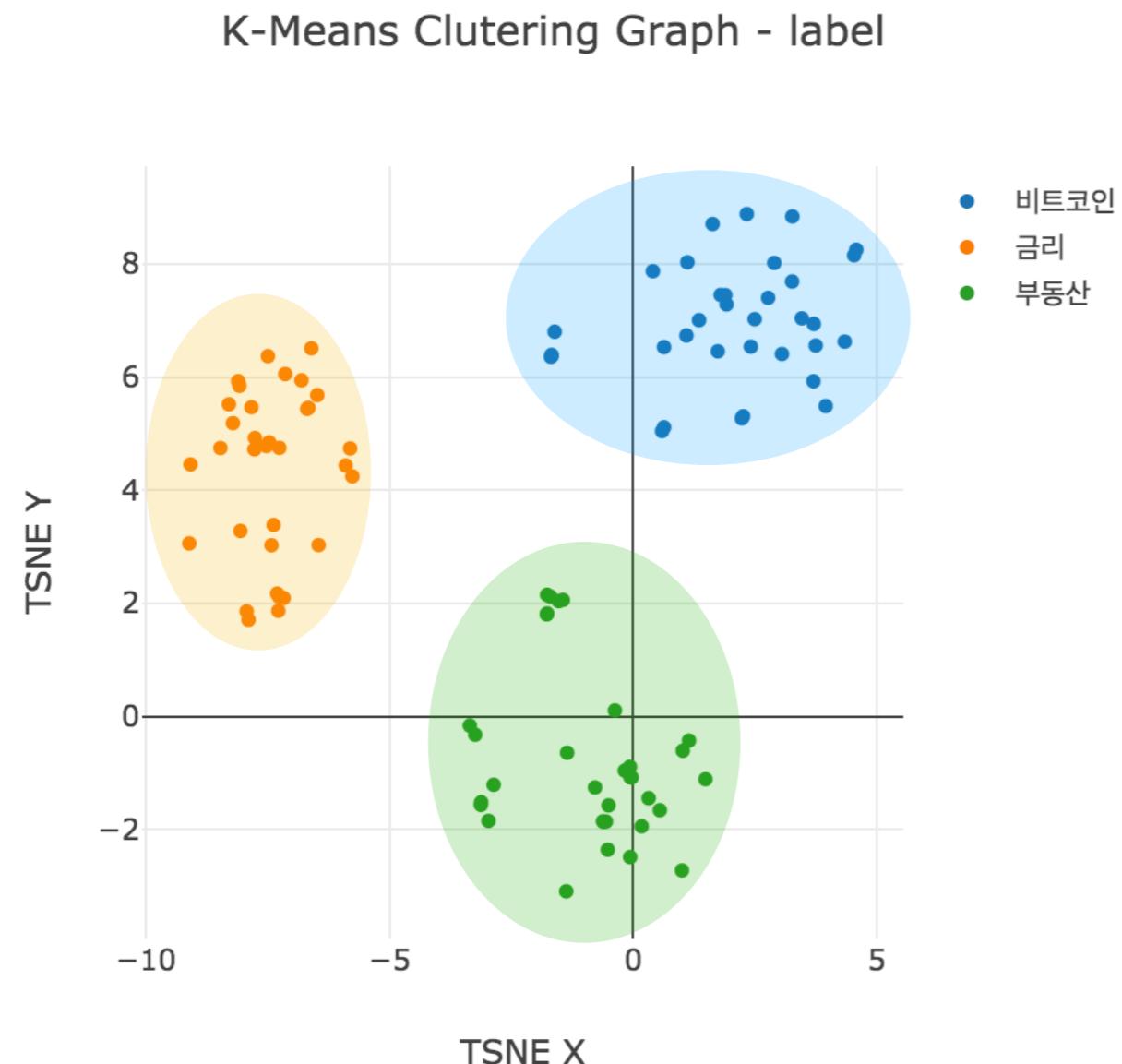
“문서 2”의 벡터 = [0, 0, 1, 2, 0, 1, 1, 1, 0, 0, 0]



문서 유사도 (Document Similarity)

벡터공간 모델 (Vector Space Model)

- ▶ 각 문서를 N 차원 상의 벡터공간에 표현하는 방법
- ▶ N은 문서를 표현하는 특성 (feature)의 수에 따라서 결정됨

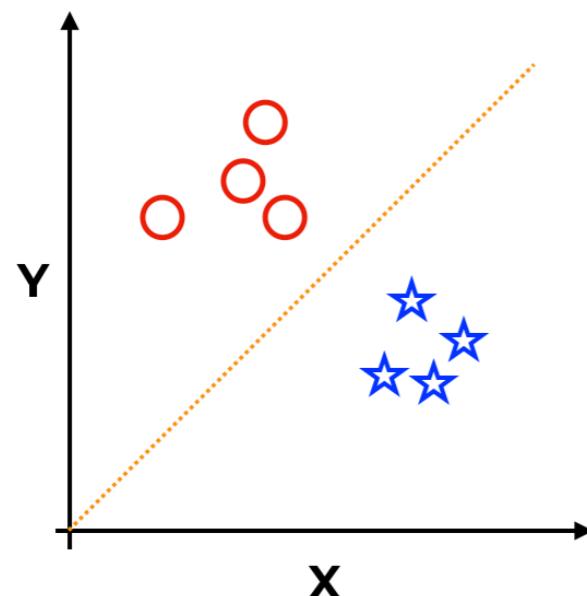


군집화 (Clustering)

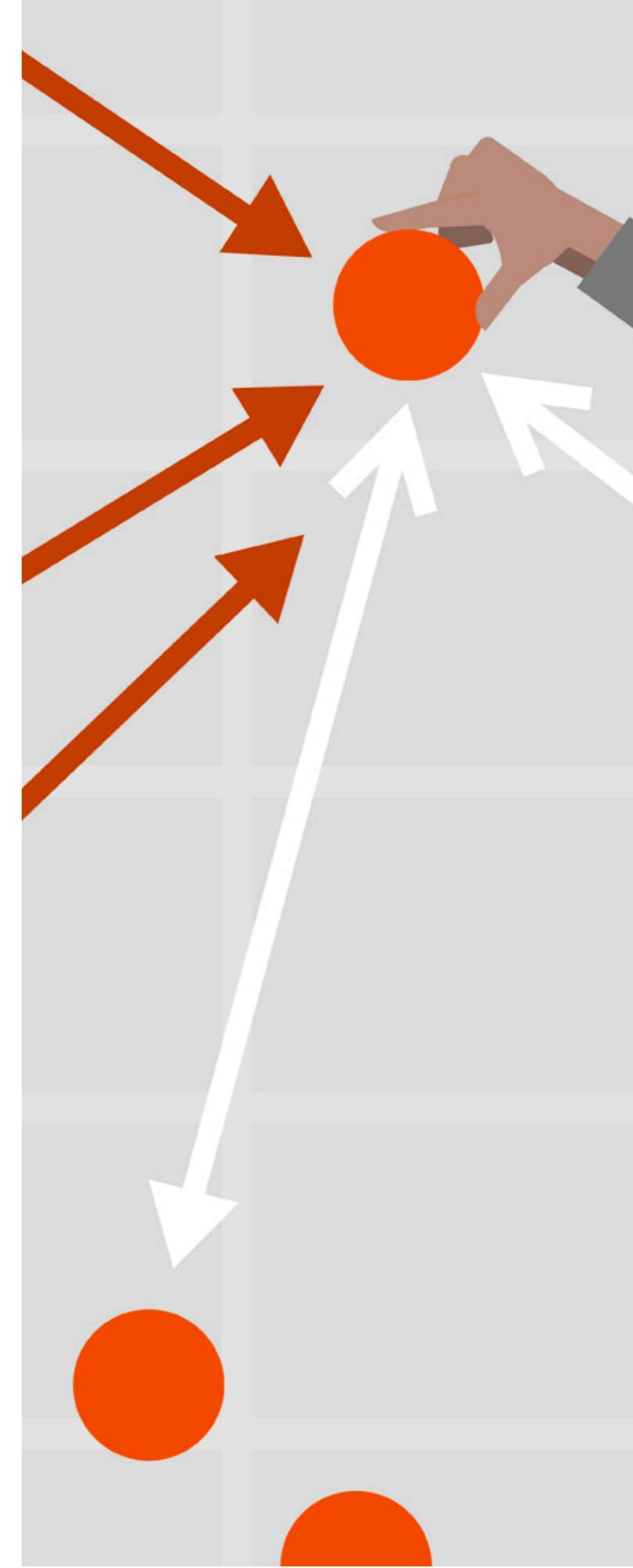
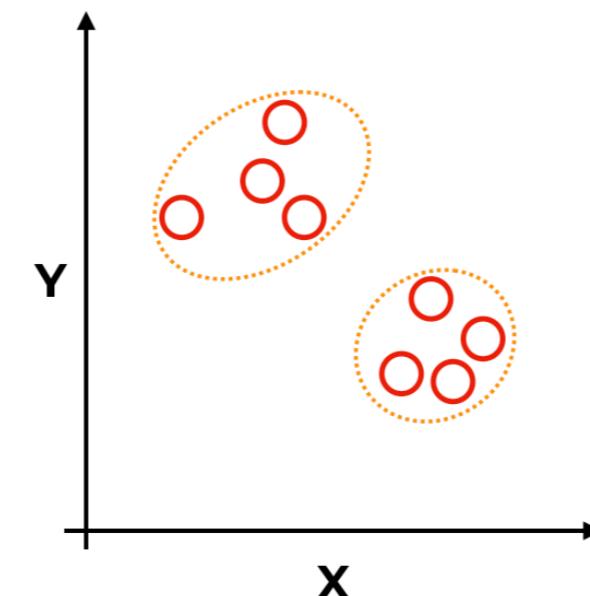
군집화란?

- ▶ 비지도 학습 (unsupervised learning)으로 데이터에서 자연스럽게 분류되는 그룹을 찾아내는 방법
- ▶ 지도 학습 (supervised learning) vs 비지도 학습
 - 지도학습 :
 y 값(타깃 값)이 사전에 정해진 경우 지도학습이며 ($y=O$ or $y=X$), 타깃값을 알고 있는 데이터를 이용해 새로운 객체의 타깃값을 예측할 수 있는 패턴을 찾아내는 것
 - 비지도학습 :
 y 값이 정해지지 않고 자율적으로 학습하는 경우 비지도 학습이라 하며, 타깃 변수에 신경쓰지 않고 데이터 집합에 있는 어떤 규칙성을 찾으려는 것

지도학습 (Supervised Learning)



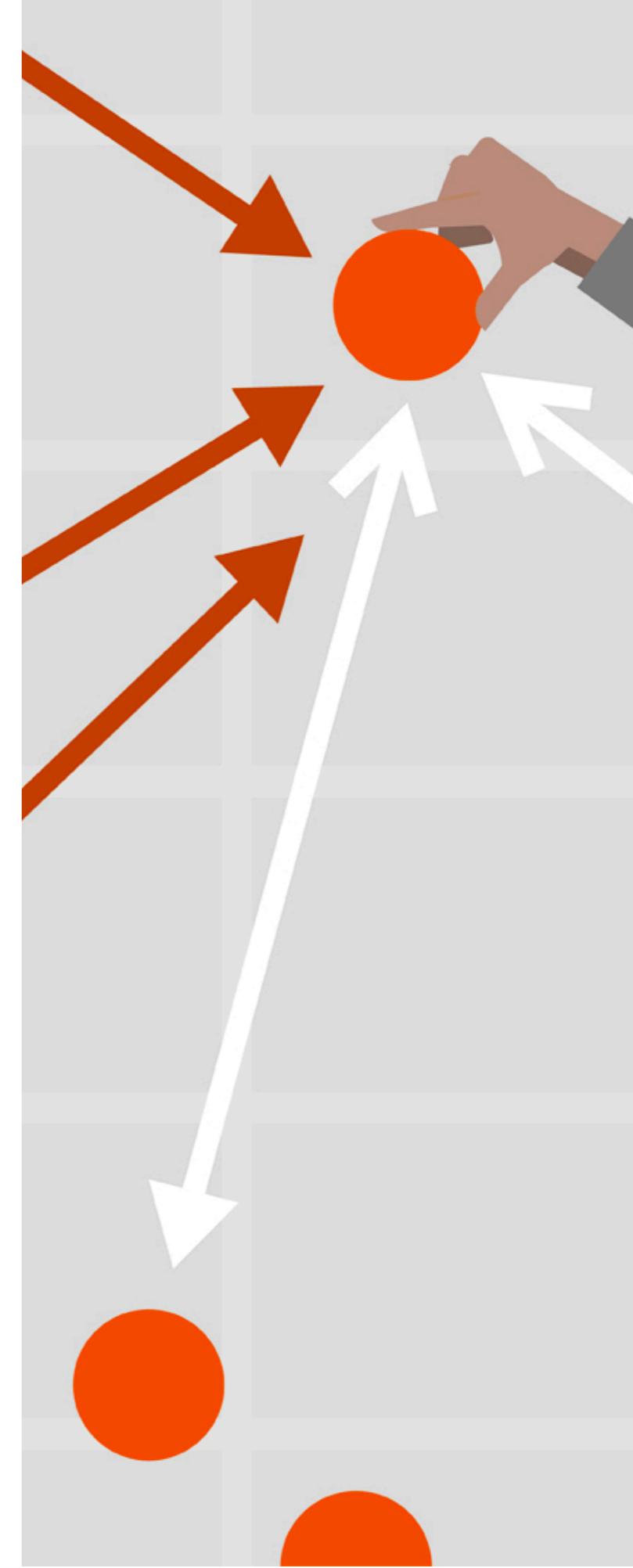
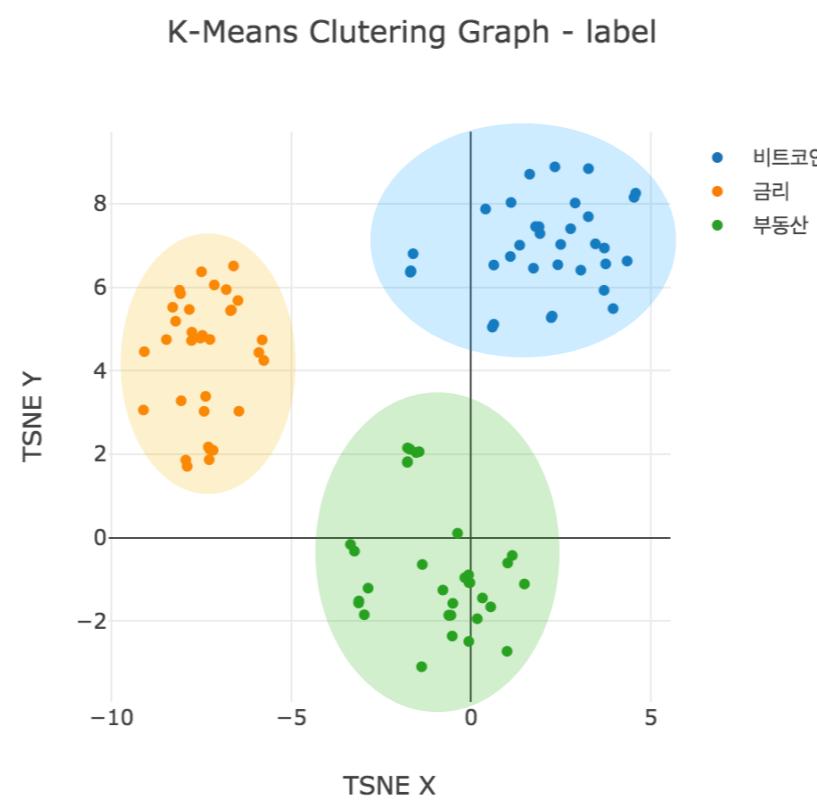
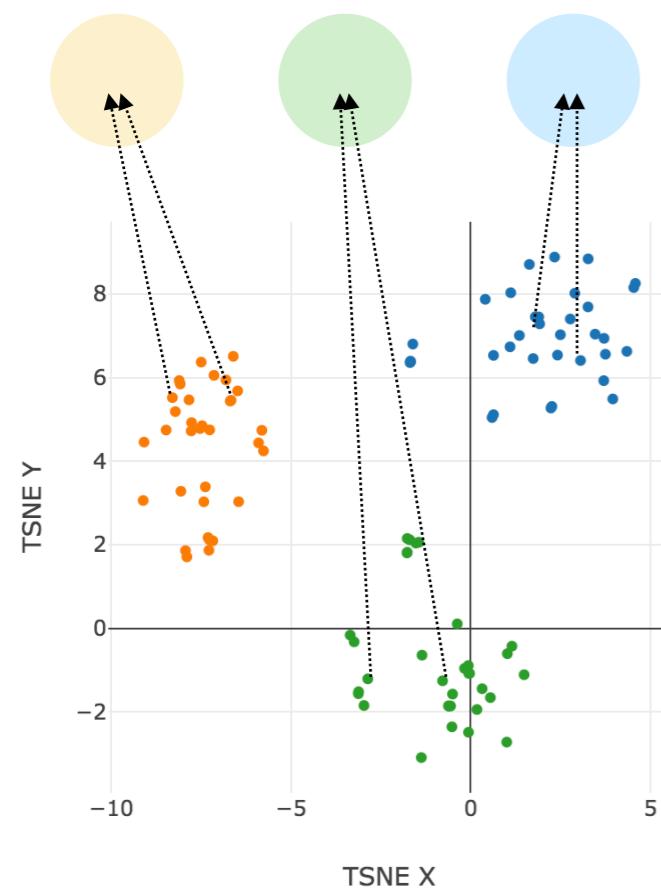
비지도학습 (Unsupervised Learning)



군집화 (Clustering)

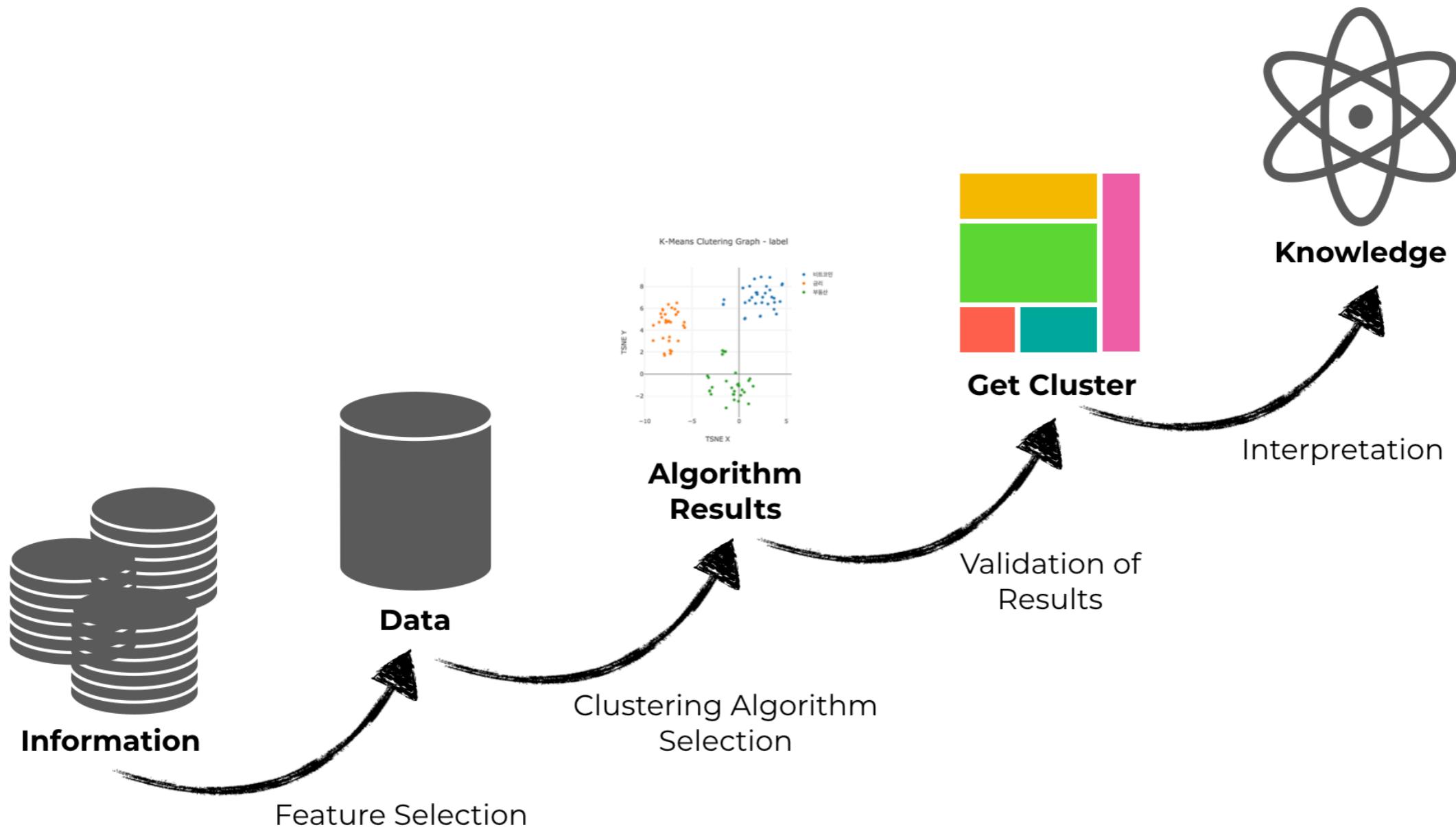
문서 군집화란?

- ▶ 문서들 사이에서 발견되는 자연스러운 그룹(군집)을 발견하고 주제를 제시하는 방법
- ▶ 분류 vs 군집화
 - 분류 : 문서의 집합을 이미 정해진 유형의 개수와 속성에 따라서 분류하는 방법
 - 군집화 : 사전에 유형의 개수와 속성이 알려지지 않은 상태로 그룹을 발견하는 방법



군집화 (Clustering)

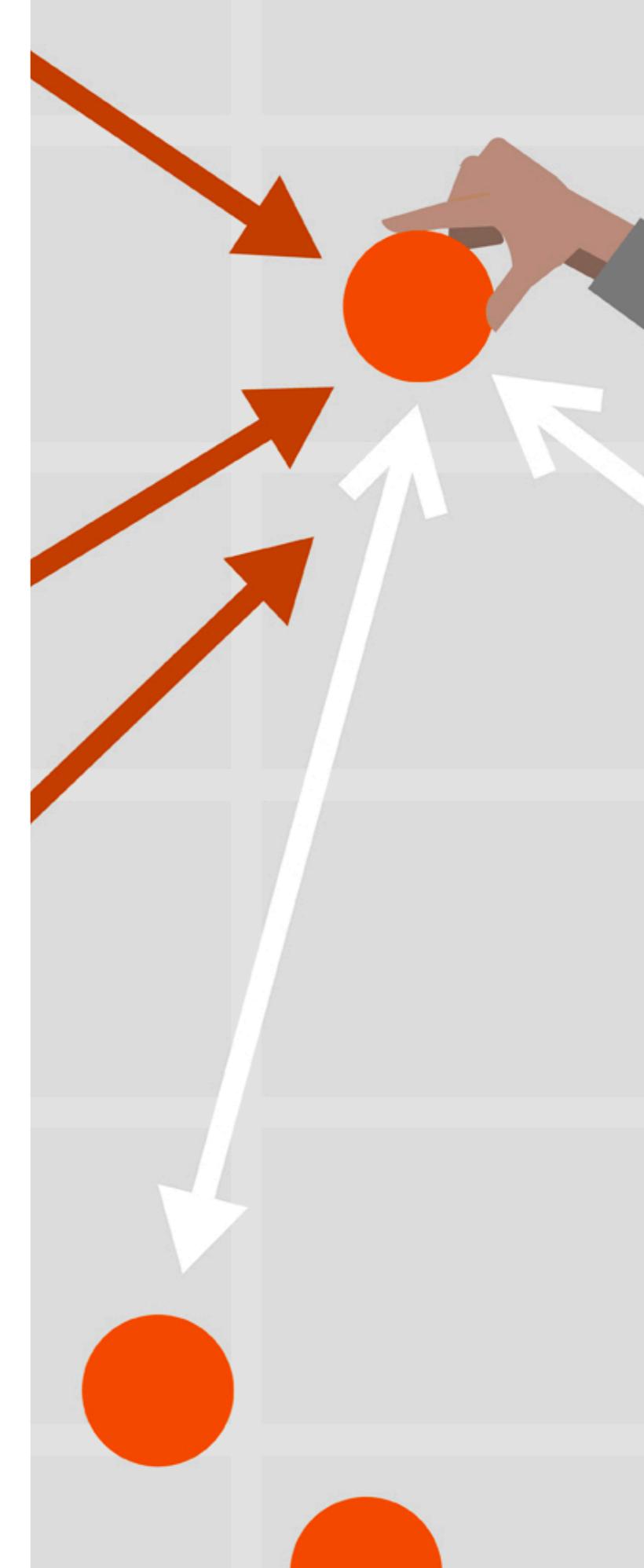
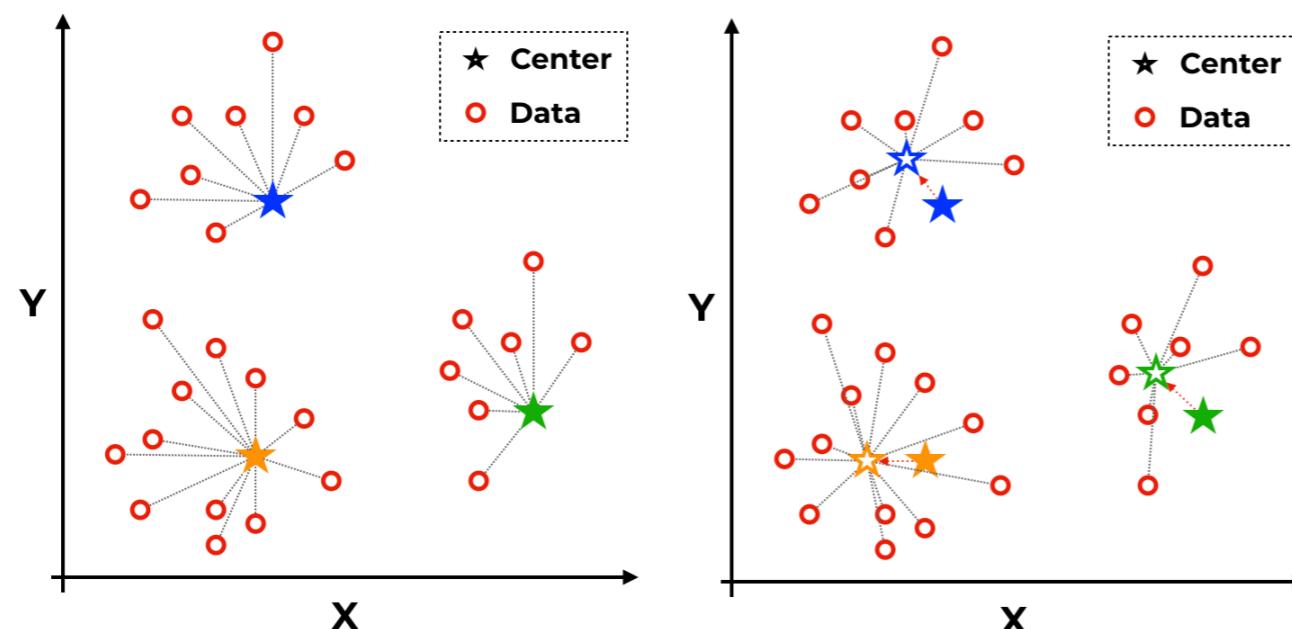
문서를 이해하기 위한 과정



군집화 (Clustering)

k-평균 군집 (K-means Clustering)

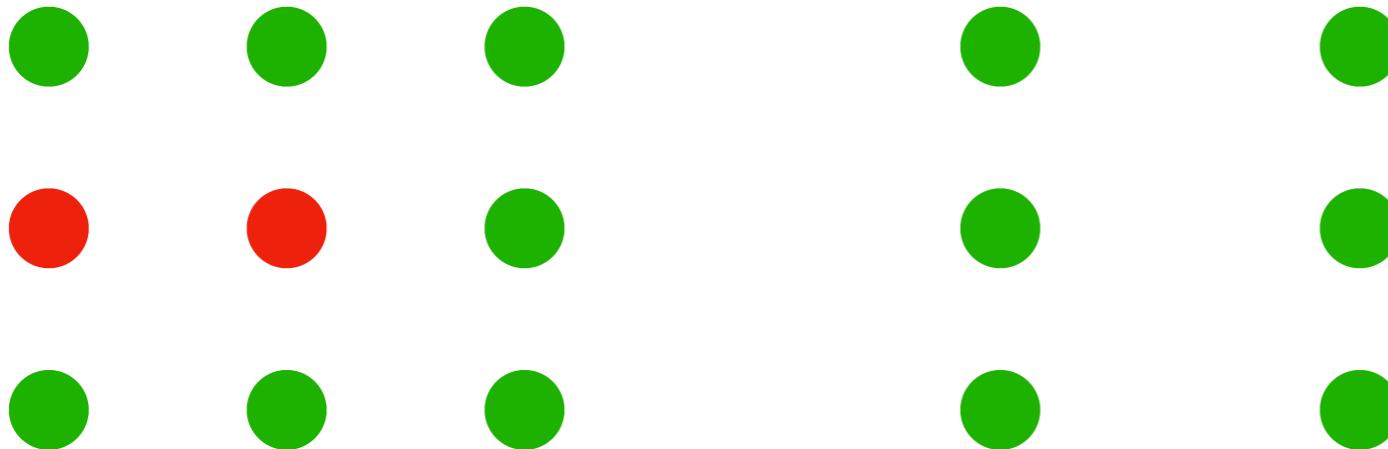
- ▶ 중점 주변 군집화 중 가장 널리 사용되며, 찾아내려는 군집 k개를 파악 (k =군집수)
- ▶ 평균은 각 군집의 중점이며, 군집에 들어 있는 객체들의 각 차원별 산술 평균값으로 표현
- ▶ k-평균 알고리즘의 3 단계
 - Step 1 : 임의로 k 개의 데이터 포인트를 시드로 선택
 - Step 2 : 각 레코드를 가장 가까운 시드에 배정
 - Step 3 : 군집의 중심점 찾기 (다시 계산)
- ▶ 군집 중점이 이동하면 각 점마다 어느 군집에 속하는지 다시 계산해야 하고, 각 점이 속한 군집을 다시 계산한 후에는 군집의 중점을 다시 계산
- ▶ 더 이상 군집에 변화가 생기지 않을 때까지 Step 2와 Step 3 과정을 반복



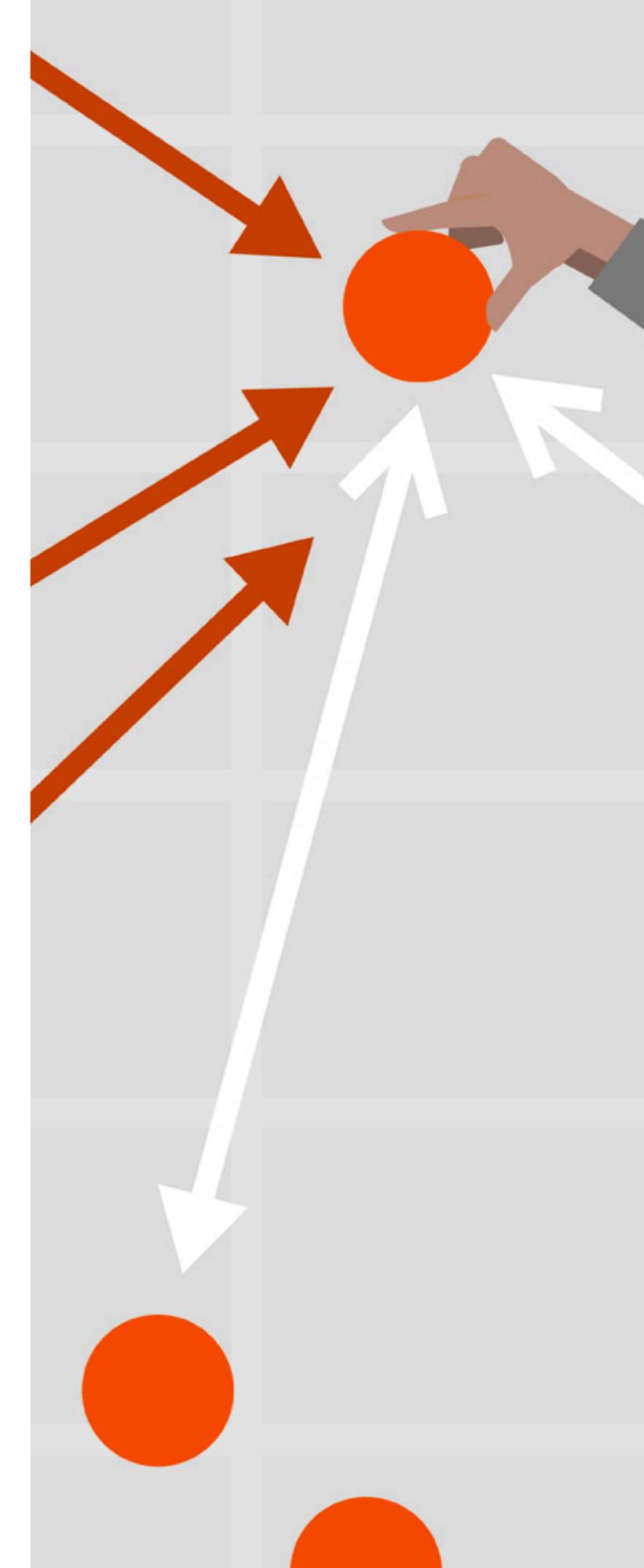
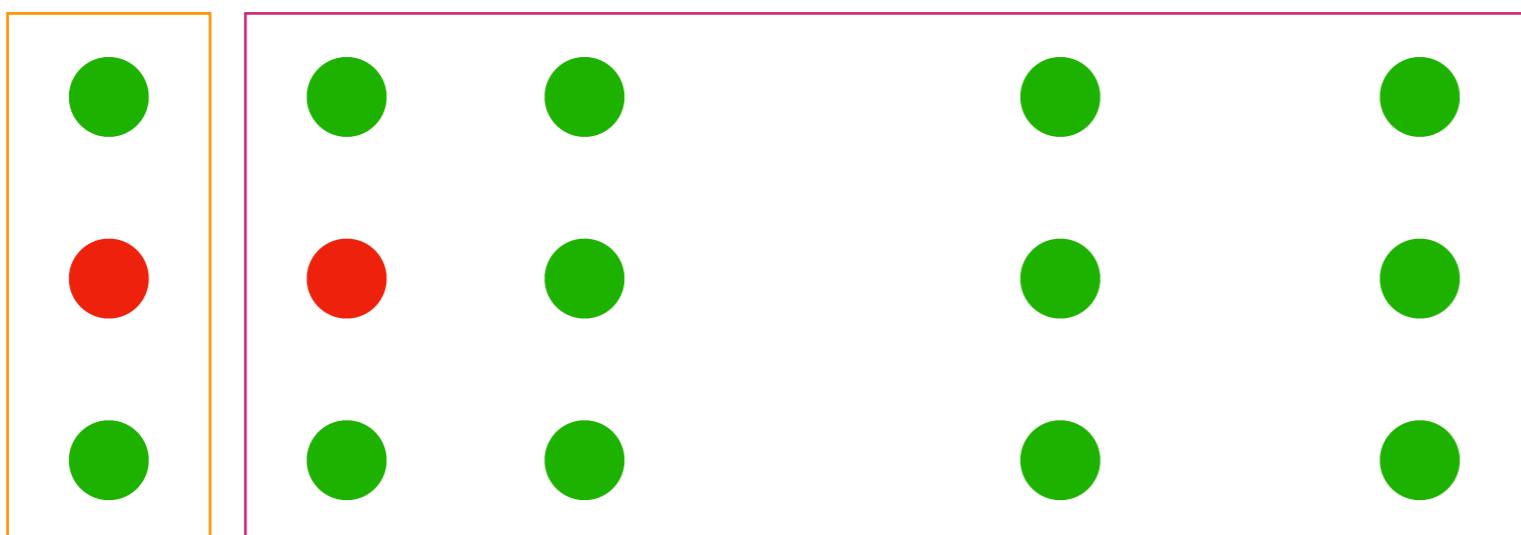
군집화 (Clustering)

k-평균 군집 (K-means Clustering)

- ▶ Step 1 : 임의로 k개의 데이터 포인트를 시드로 선택



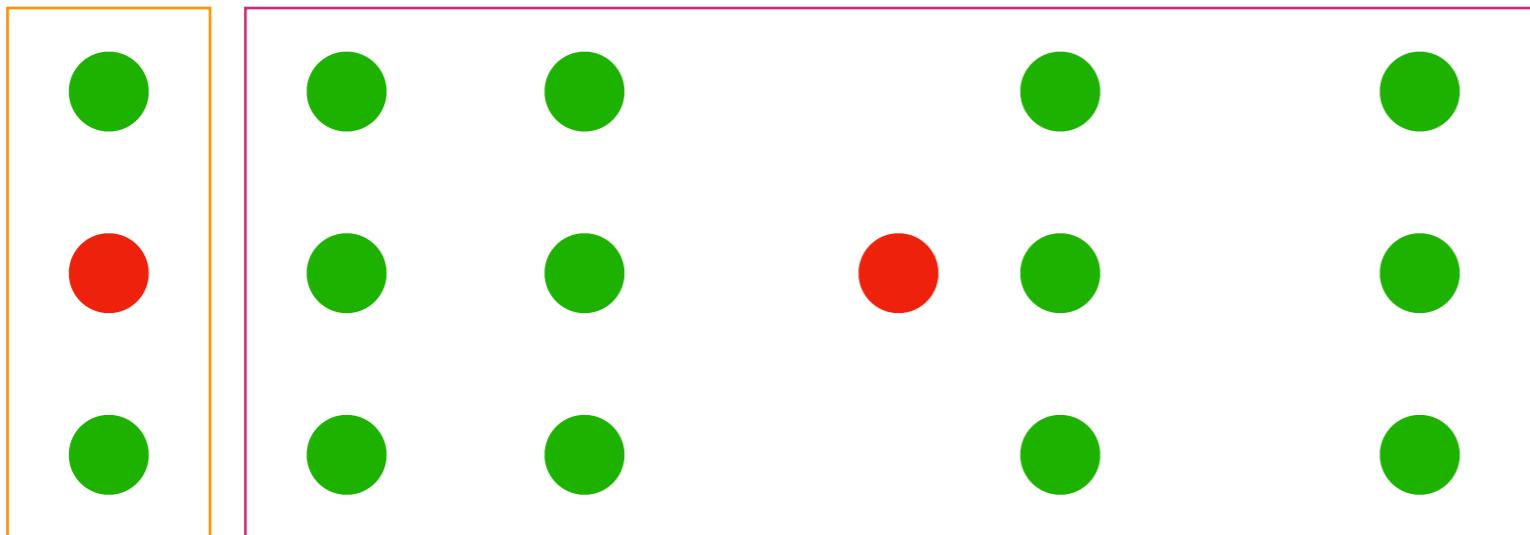
- ▶ Step 2 : 각 레코드를 가장 가까운 시드에 배정



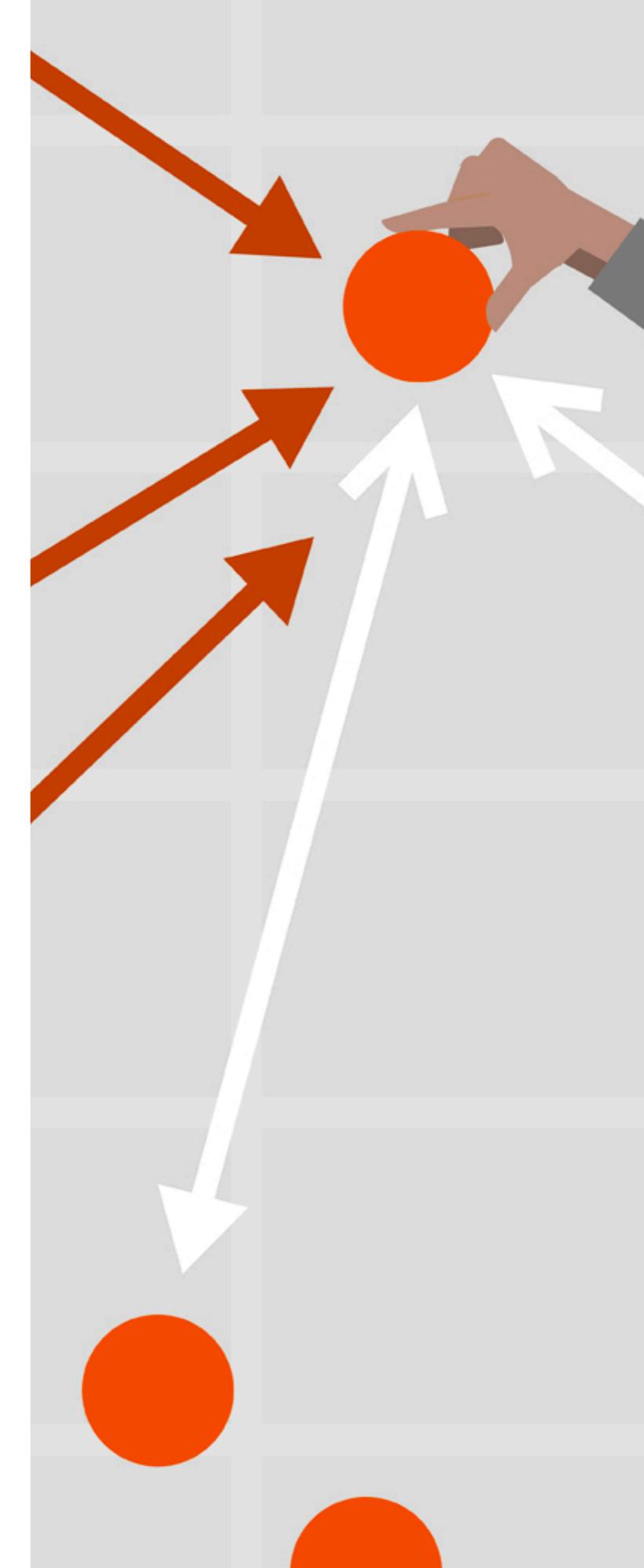
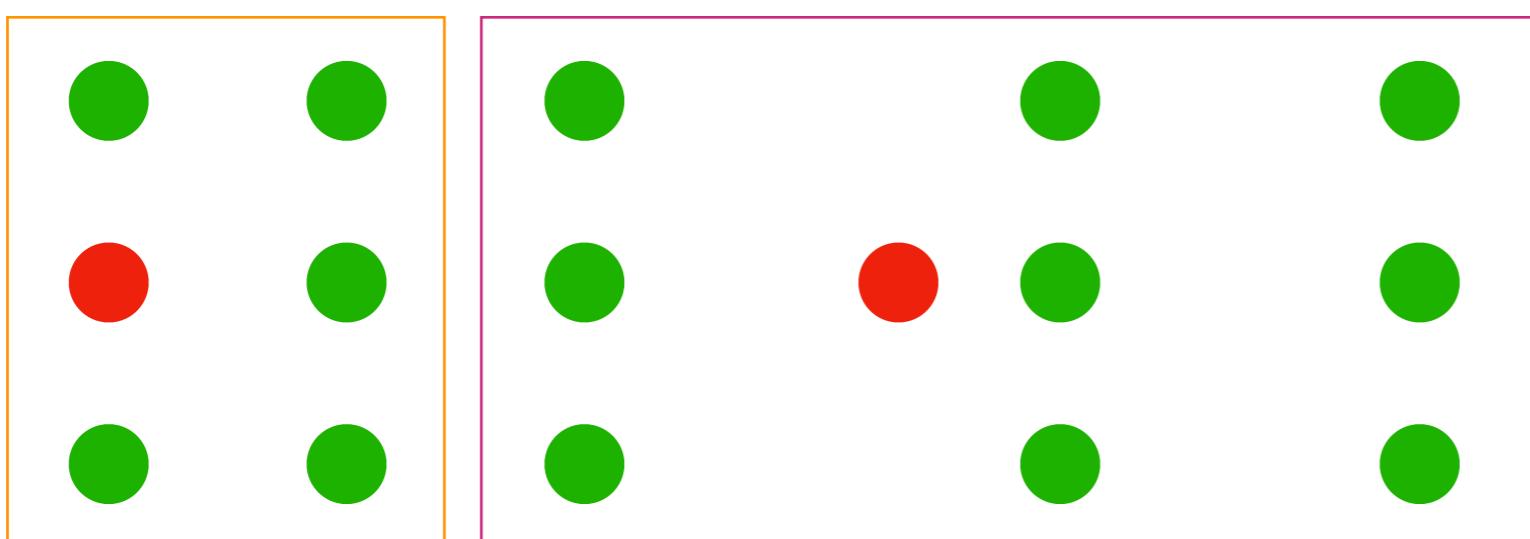
군집화 (Clustering)

k-평균 군집 (K-means Clustering)

- ▶ Step 3 : 군집의 중심점 찾기 (다시 계산)



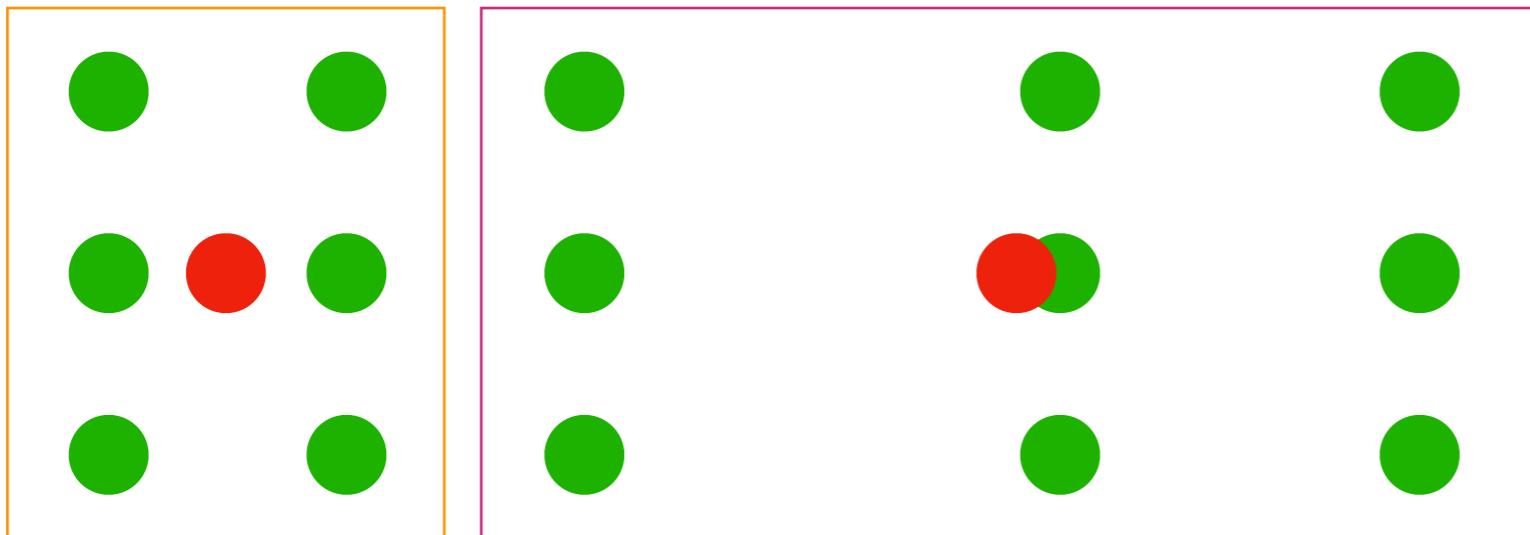
- ▶ Step 2 : 각 레코드를 가장 가까운 시드에 배정



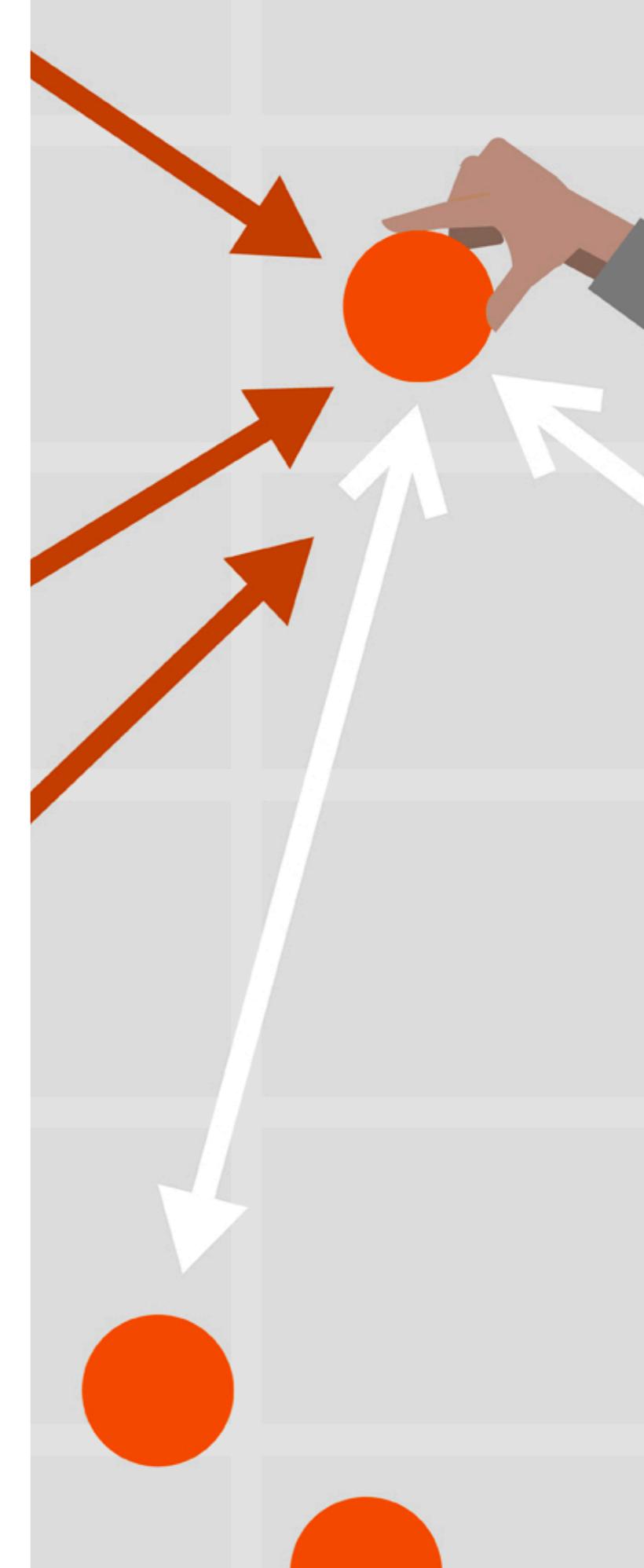
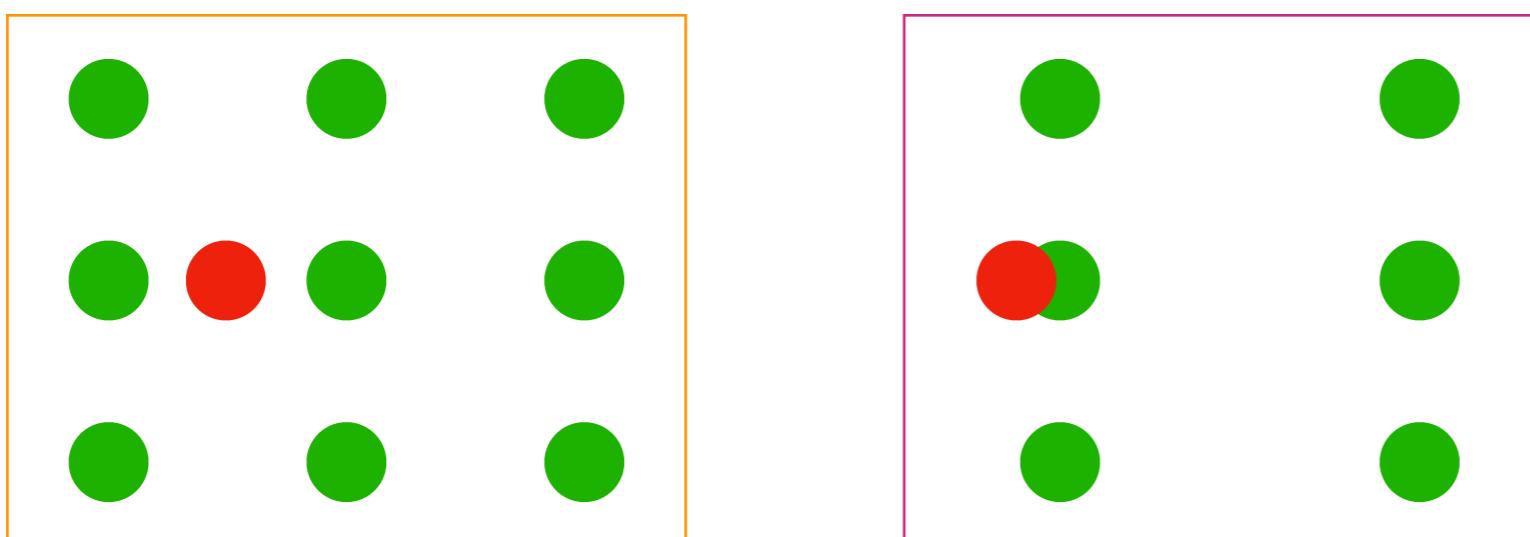
군집화 (Clustering)

k-평균 군집 (K-means Clustering)

- ▶ Step 3 : 군집의 중심점 찾기 (다시 계산)



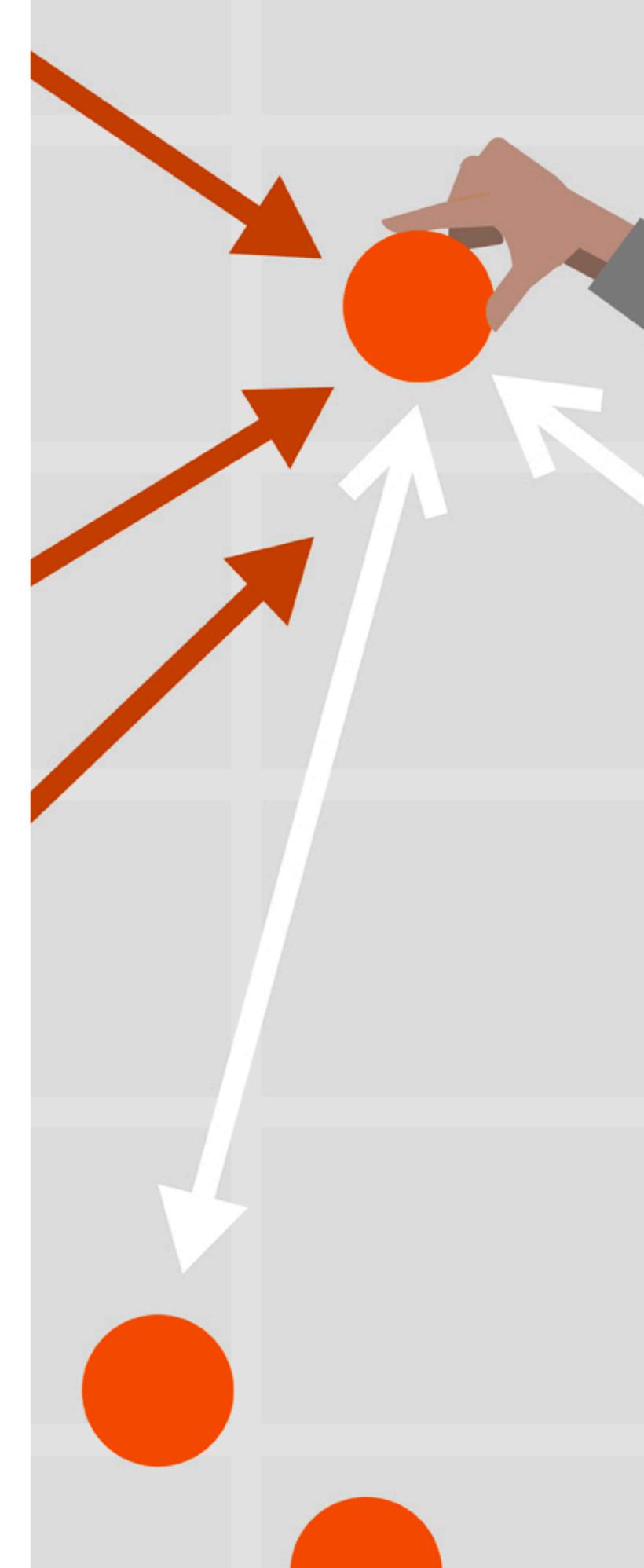
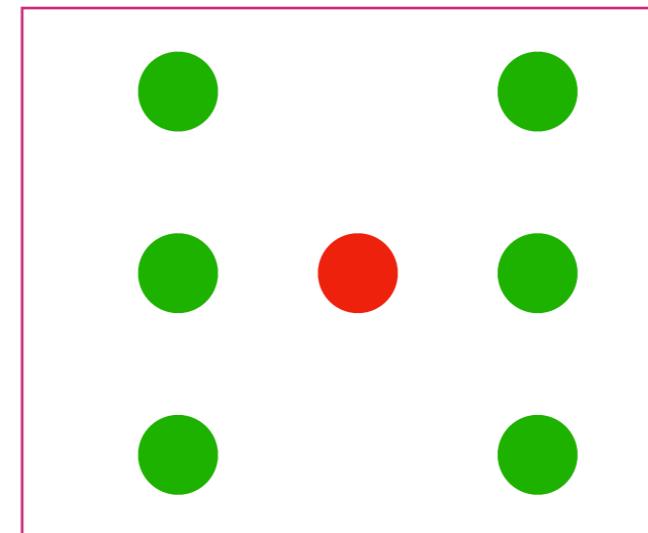
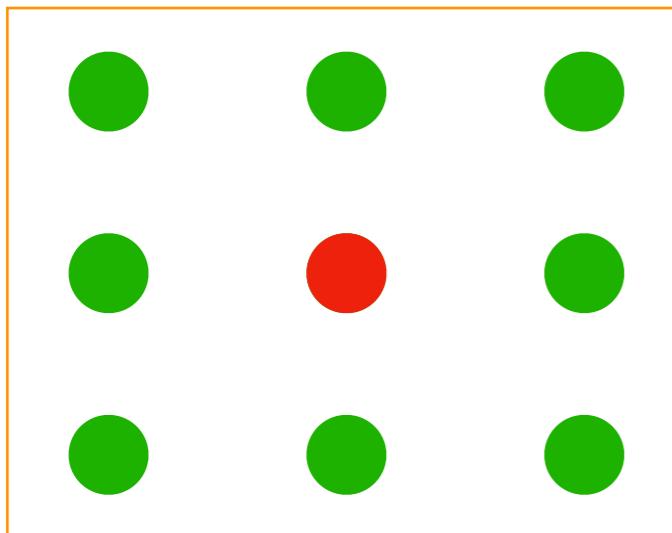
- ▶ Step 2 : 각 레코드를 가장 가까운 시드에 배정



군집화 (Clustering)

k-평균 군집 (K-means Clustering)

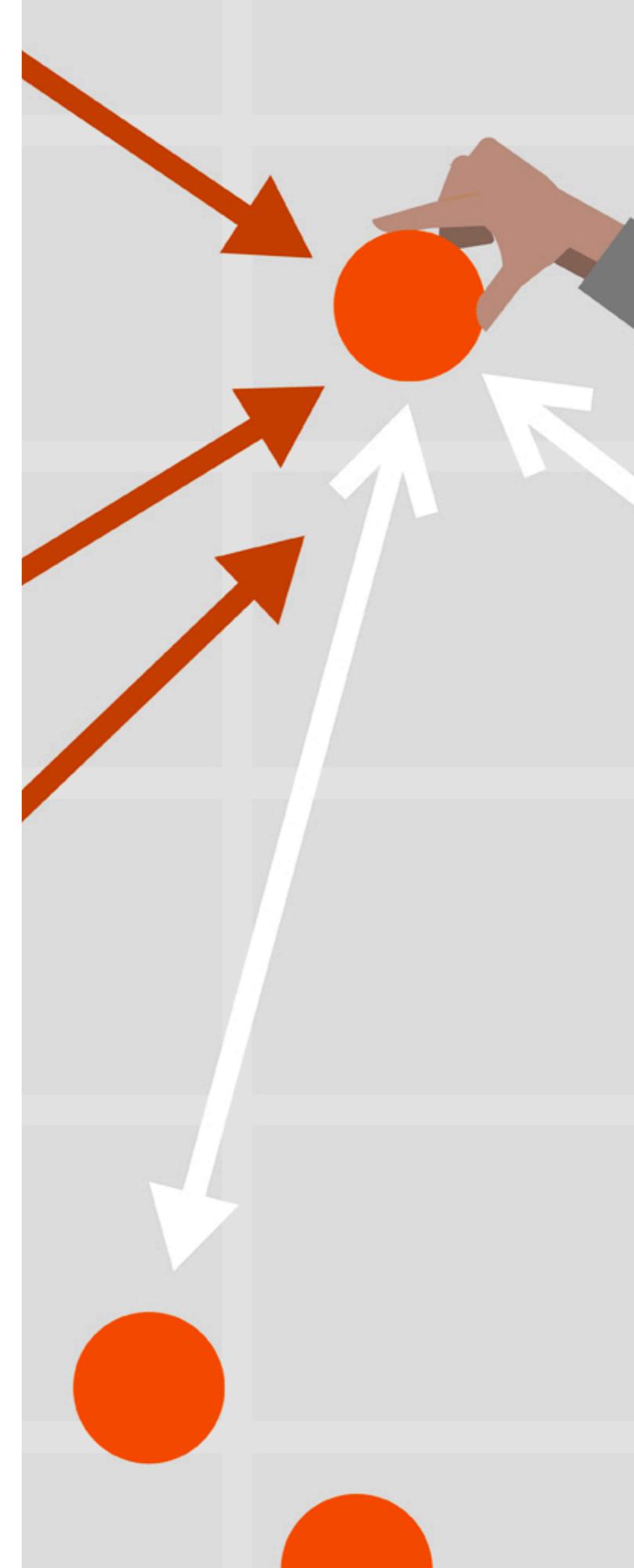
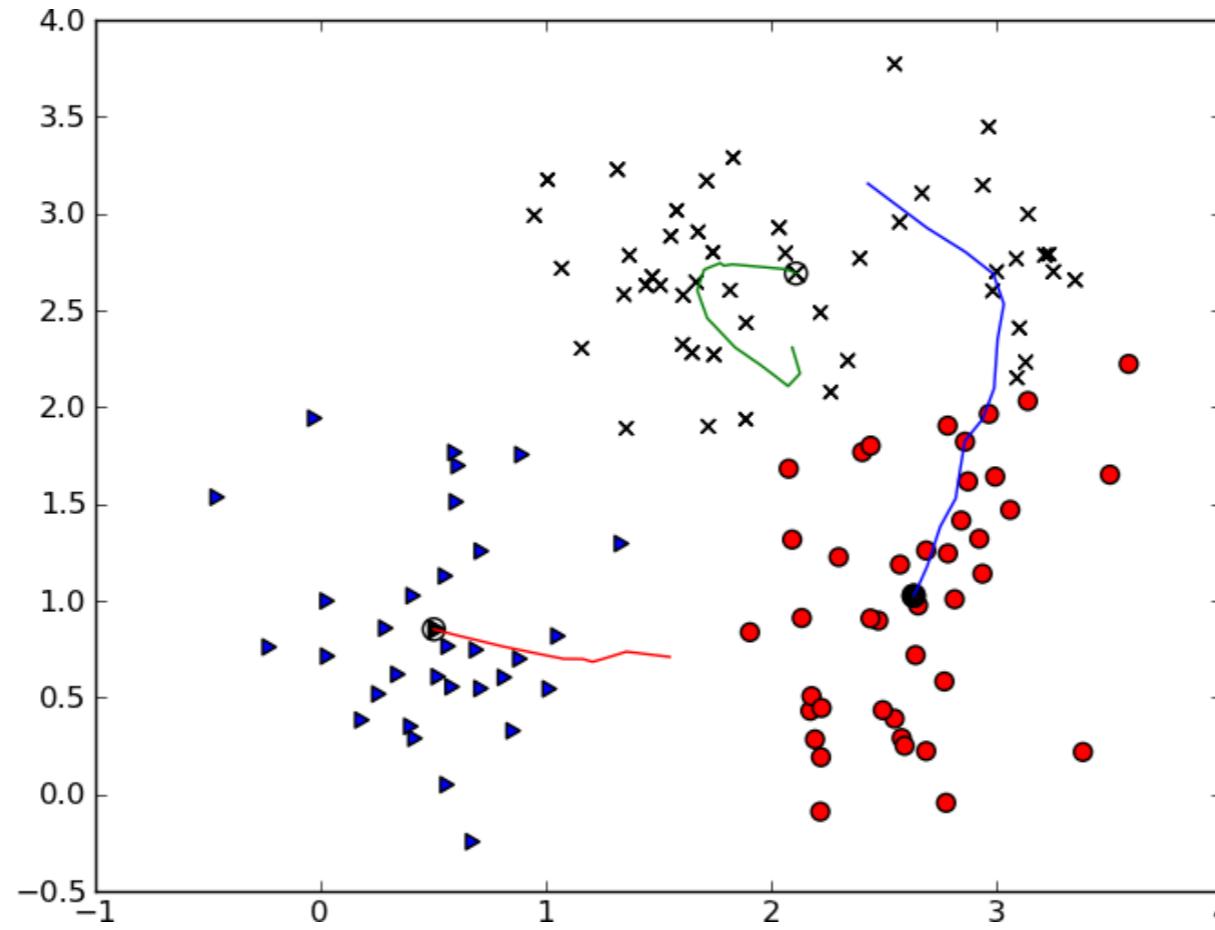
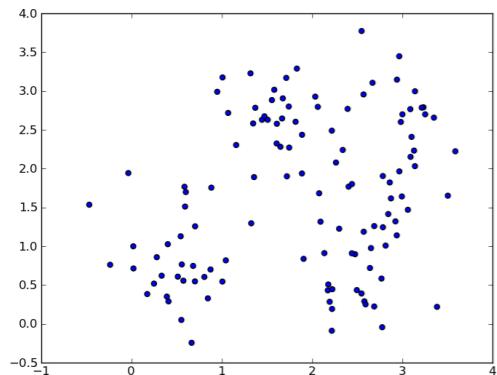
- ▶ Step 3 : 군집의 중심점 찾기 (다시 계산)



군집화 (Clustering)

k-평균 군집 (K-means Clustering)

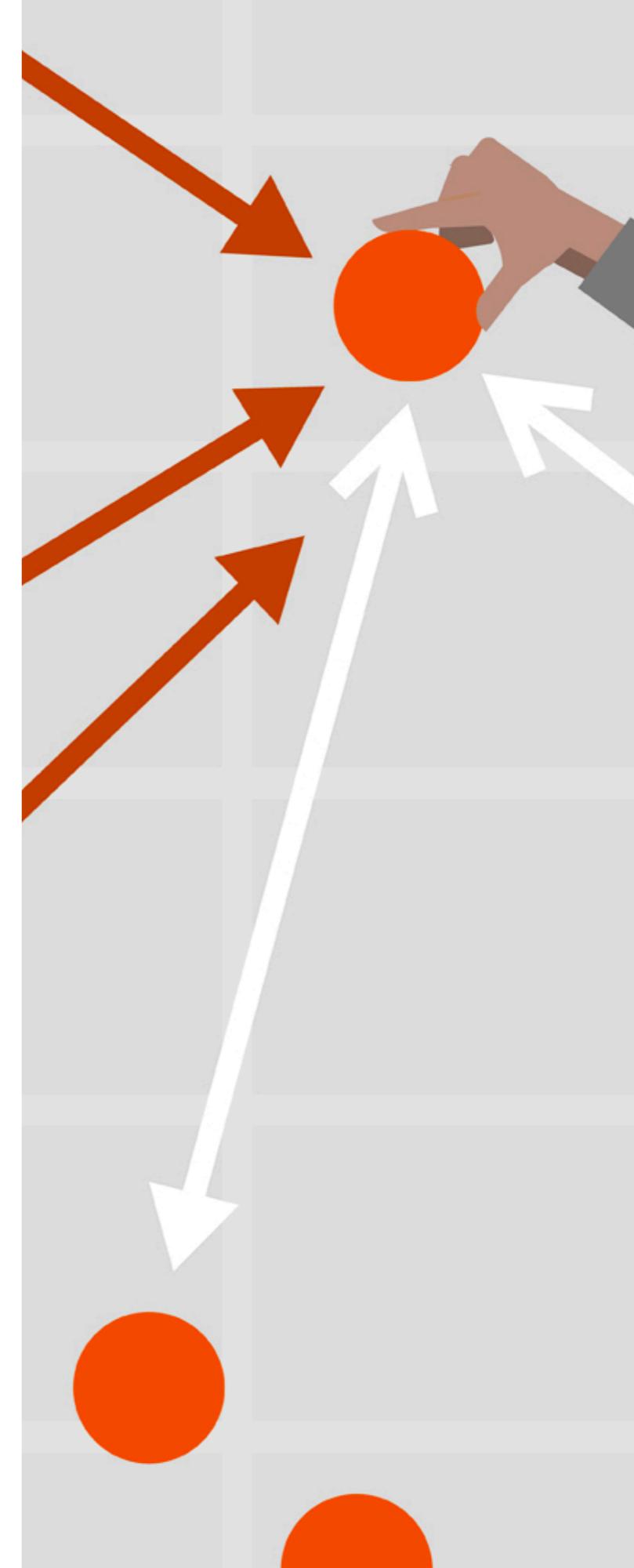
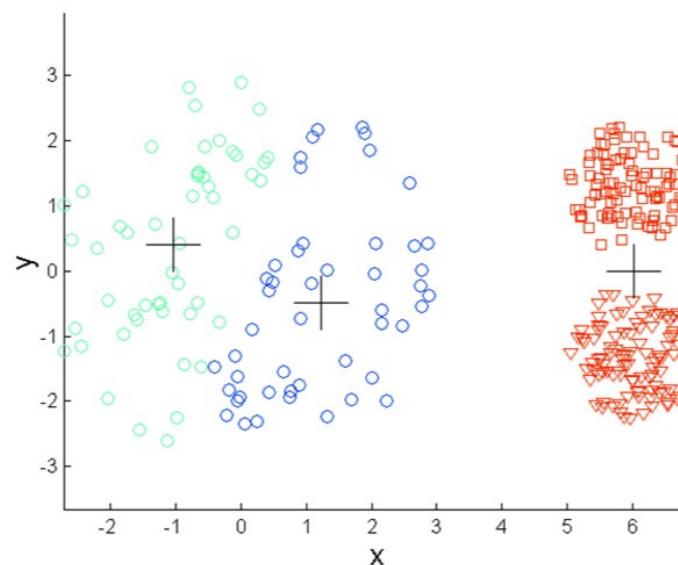
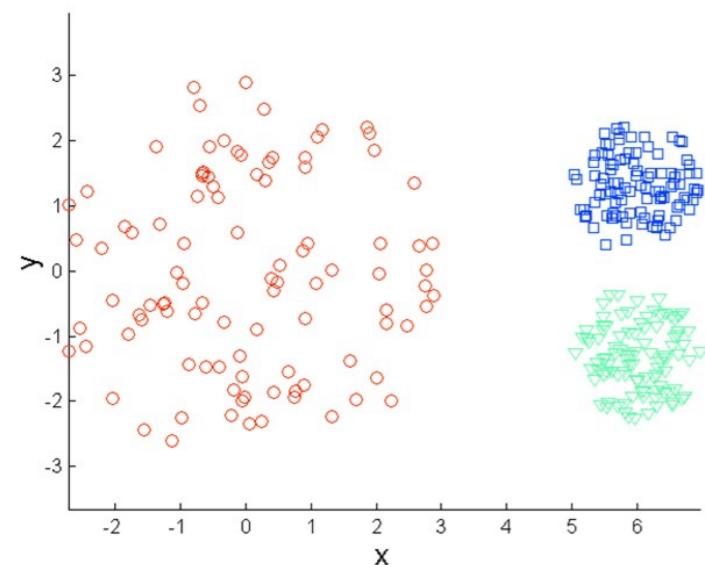
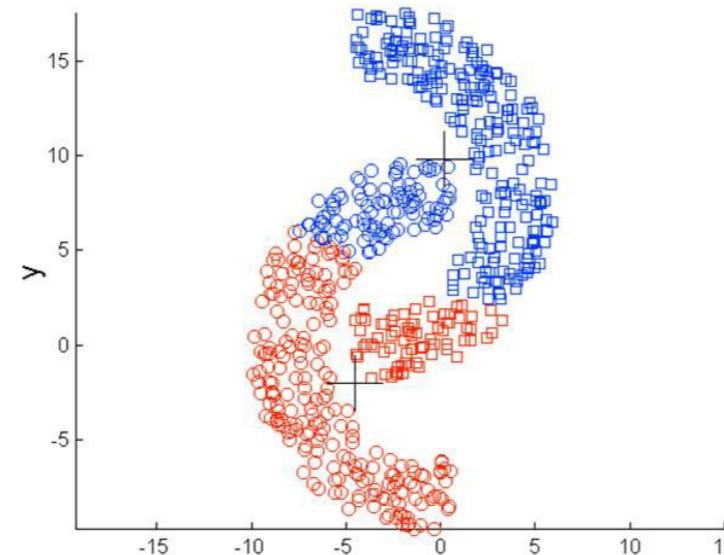
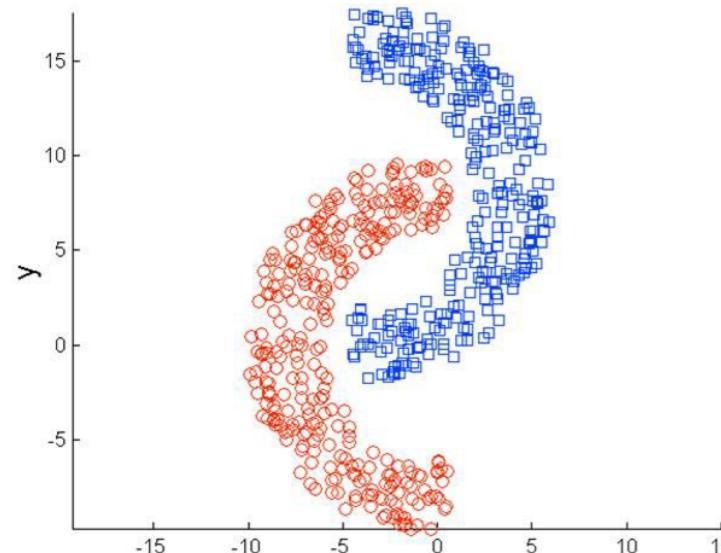
- ▶ K-평균 알고리즘은 각 단계마다 중점을 변경해가면 여러 번 반복
- ▶ 군집의 왜곡 (distortion)은 각 데이터에서 해당 중점까지의 거리의 제곱을 모두 합한 값으로, 이를 이용할 경우 왜곡값이 가장 작은 군집이 제일 좋음
- ▶ 실행 시간 측면에서 보면 k-평균 알고리즘은 효율이 좋음 (비교적 실행속도가 빠름)



군집화 (Clustering)

k-평균 군집의 한계점

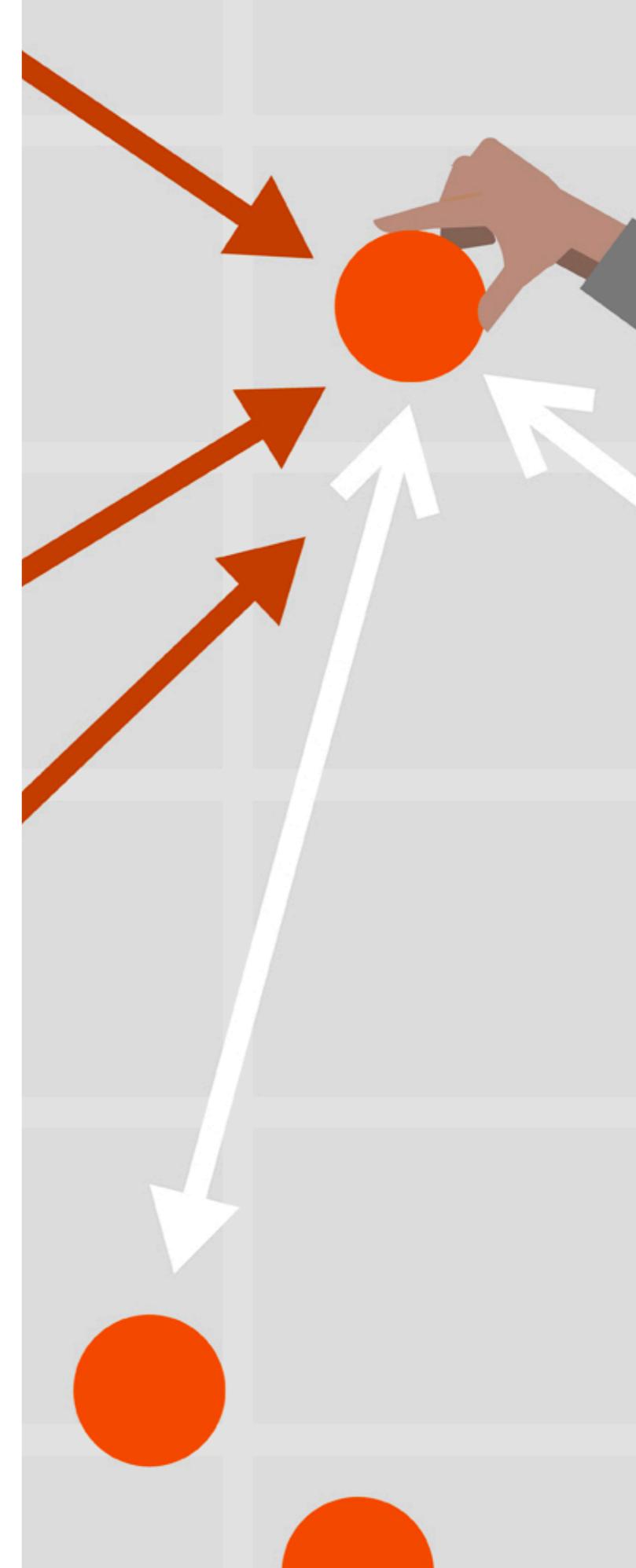
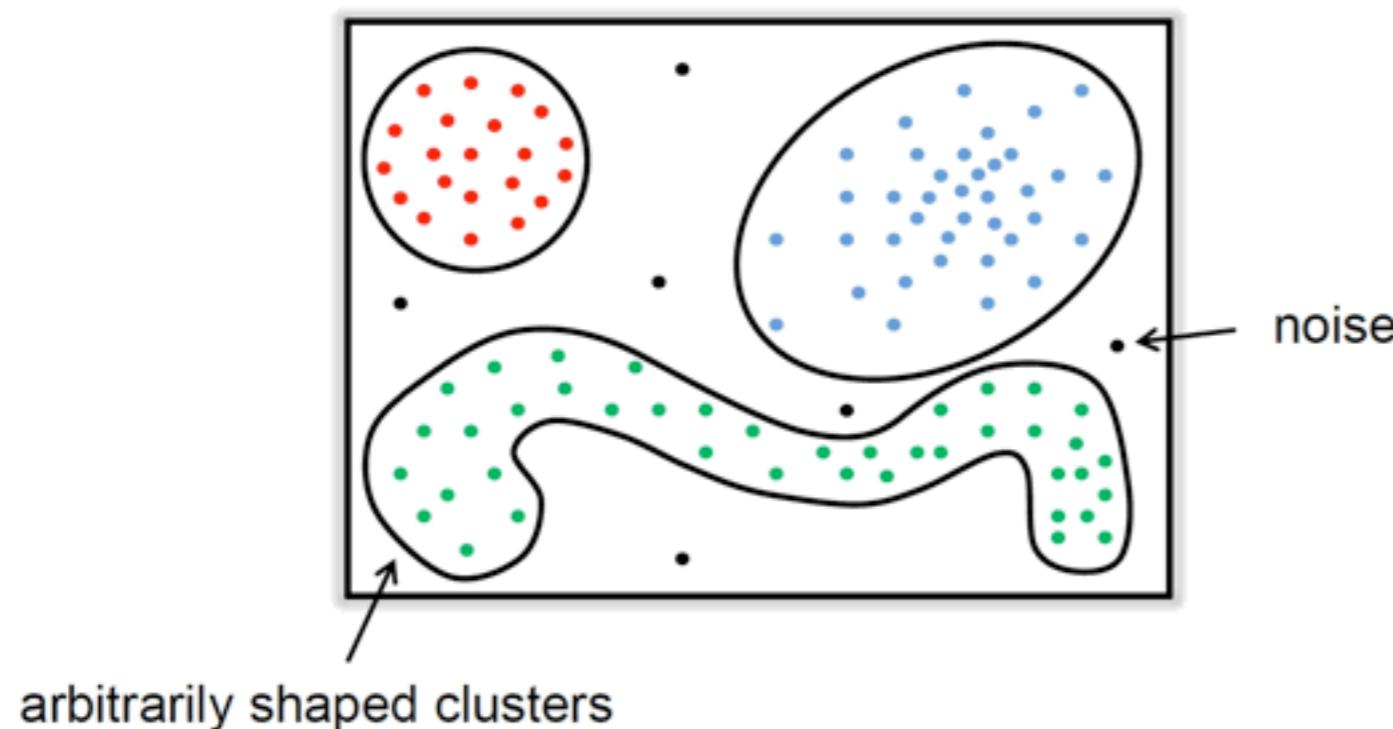
- ▶ 구형이 아닌 데이터 분포에 대해서 잘 구분할 수 있을까?
- ▶ 밀도가 다른 군집을 분할 수 있을까?



군집화 (Clustering)

Density-Based Clustering

- ▶ 데이터의 분포와 밀도 (density)를 고려하여 클러스터를 구성하는 방법
- ▶ 구형이 아닌 임의의 모양으로 생긴 클러스터도 잘 찾을 수 있음
- ▶ 클러스터링 과정에서 노이즈를 제거하는 것이 가능함

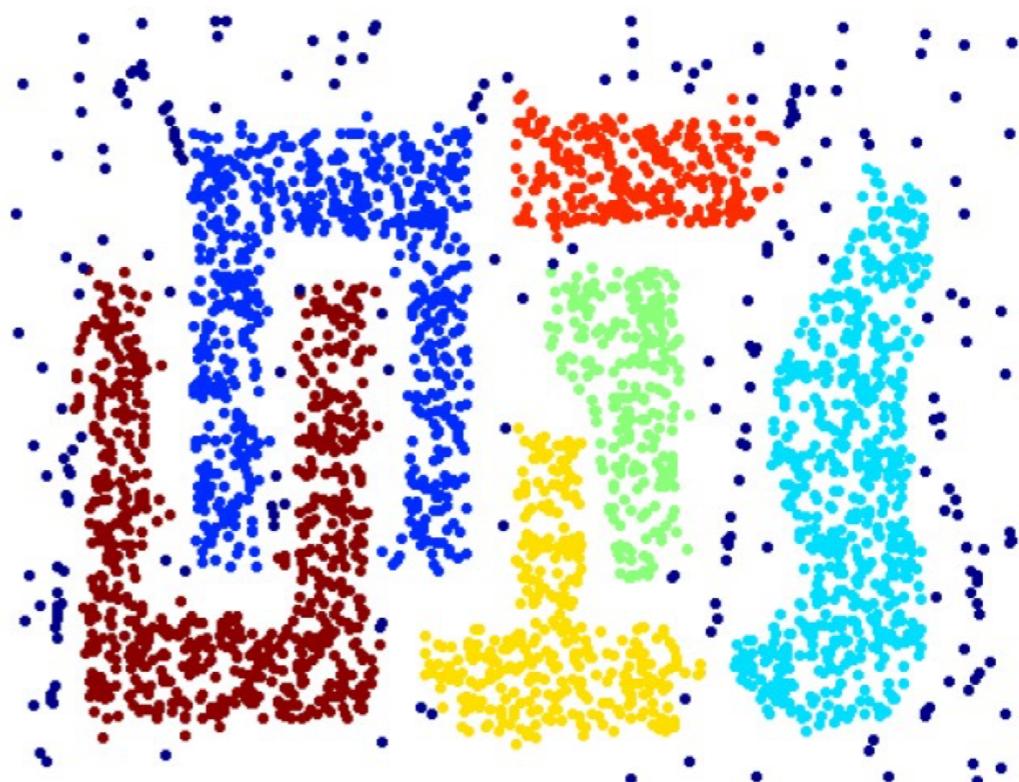


군집화 (Clustering)

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



Original Points

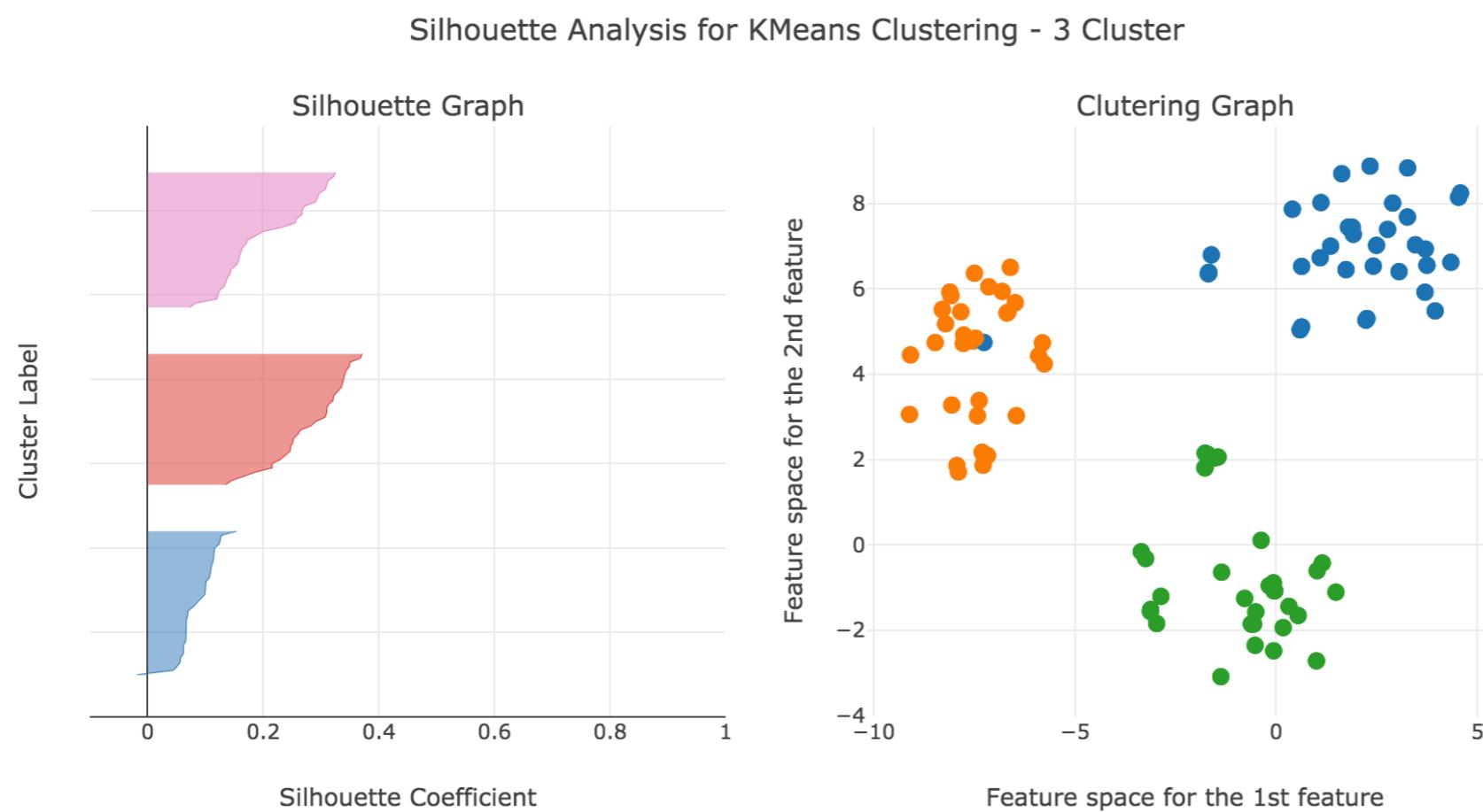
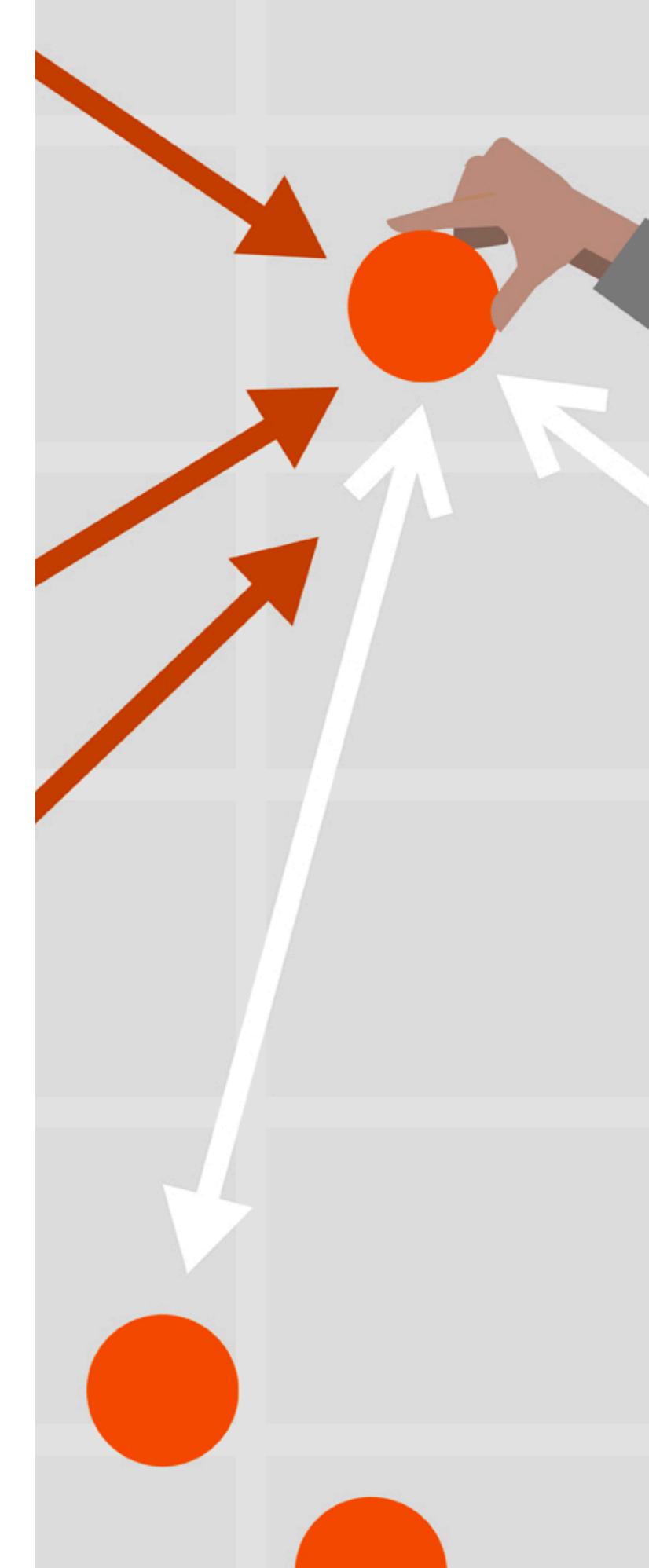


Clusters

군집의 평가 (Validation)

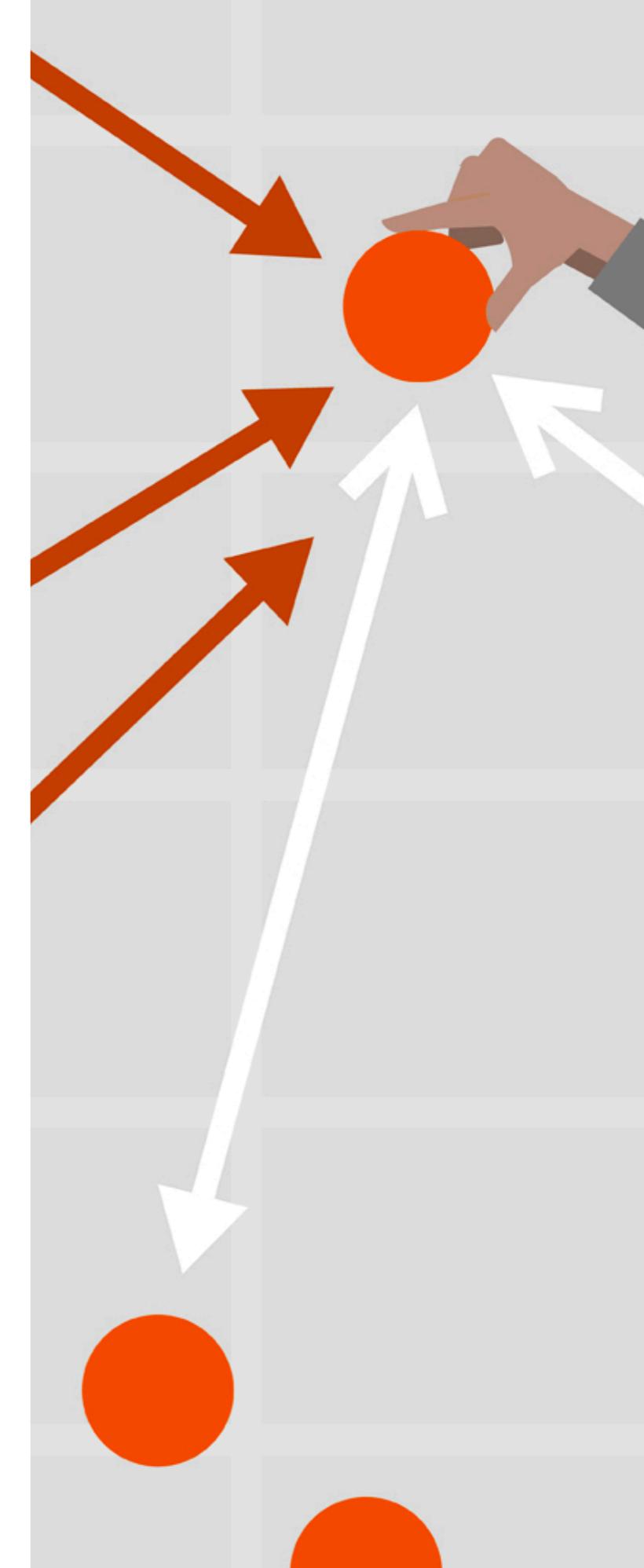
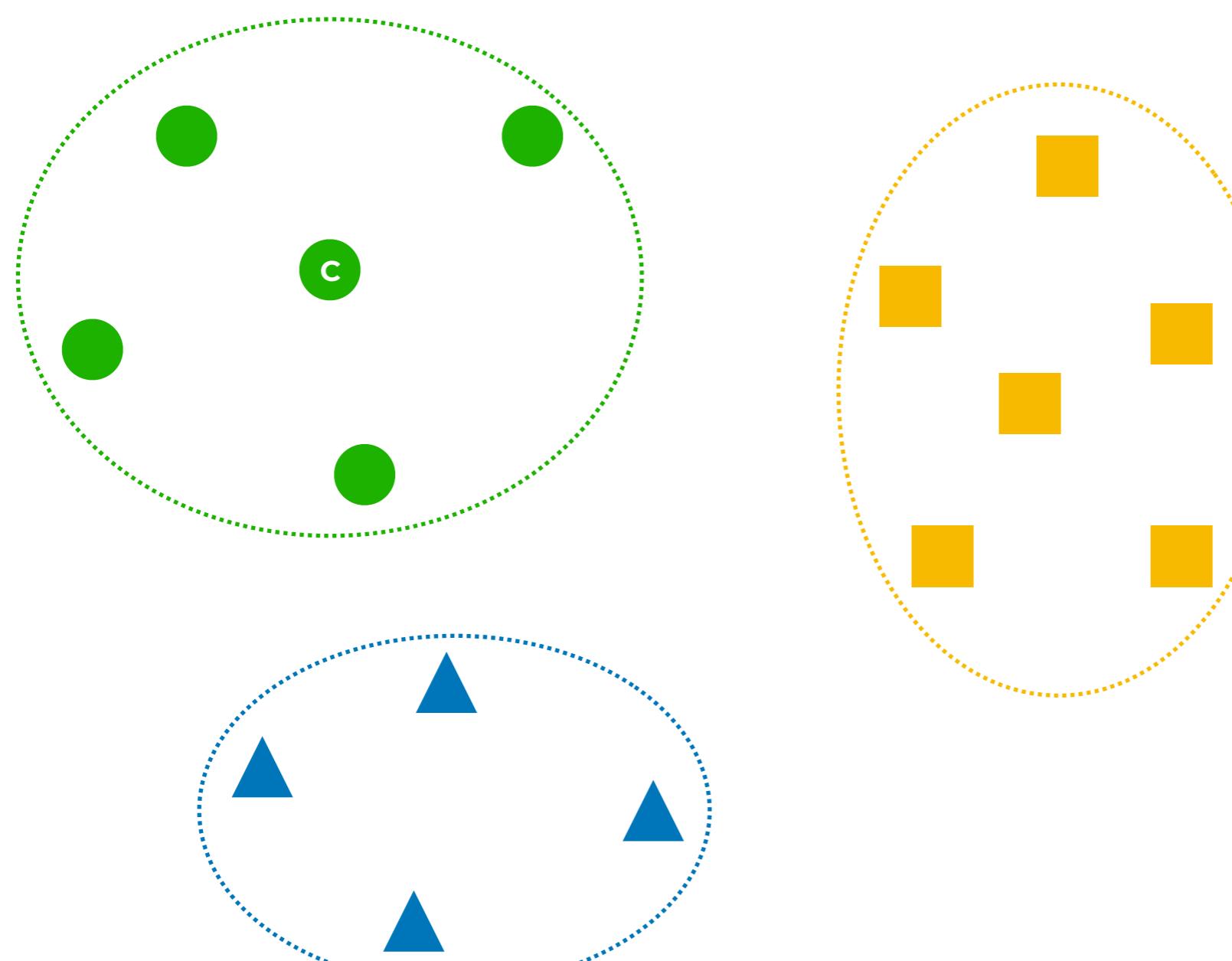
Silhouette Score

- ▶ 클러스터 내의 일관성과 유효성을 검증하는 척도 중 하나로, 각 객체가 얼마나 잘 분류되었는가를 판단하고 시각화하기 위해 활용됨
- ▶ 다른 클러스터와 비교하여 객체가 자체 클러스터와 얼마나 비슷한지 (cohesion) 또는 분리되어 있는지 (separation)에 대한 척도



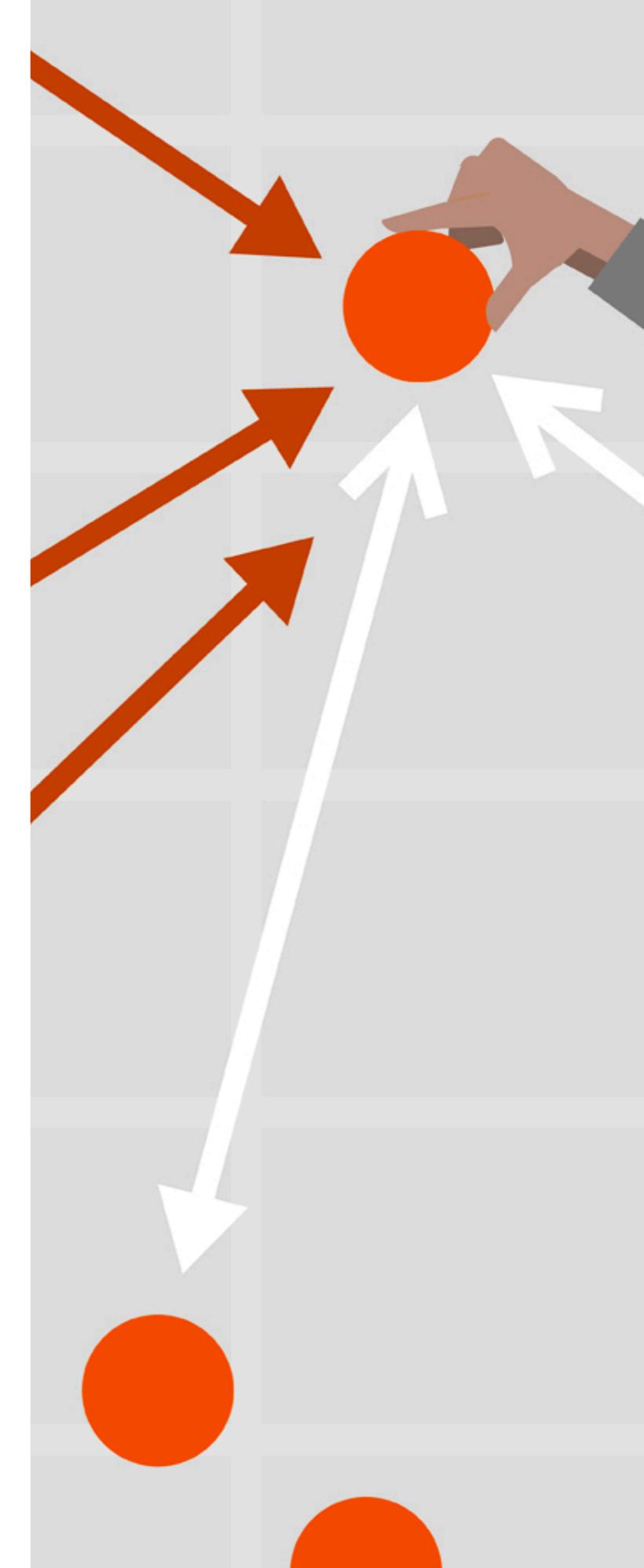
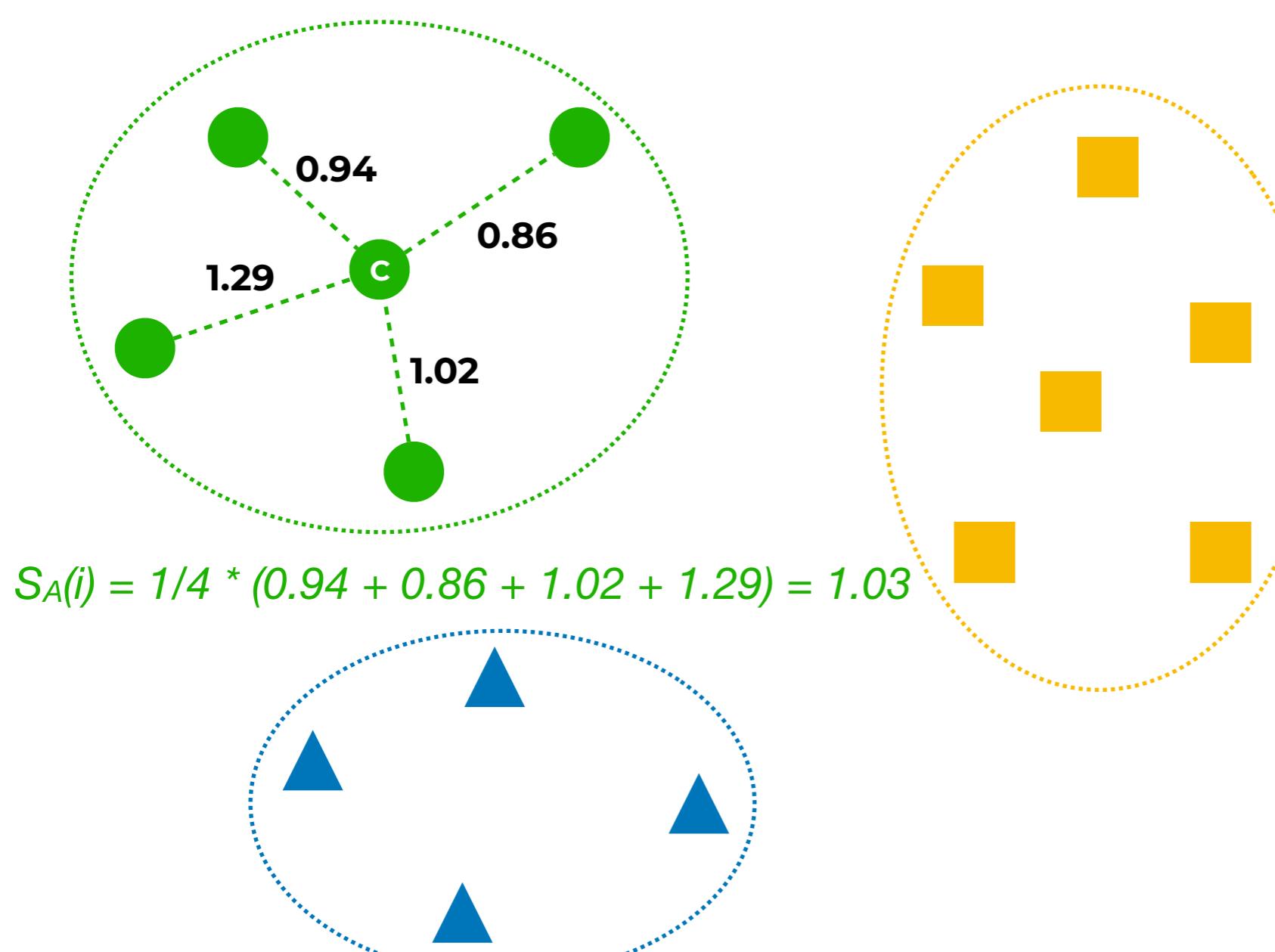
군집의 평가 (Validation)

Silhouette Score



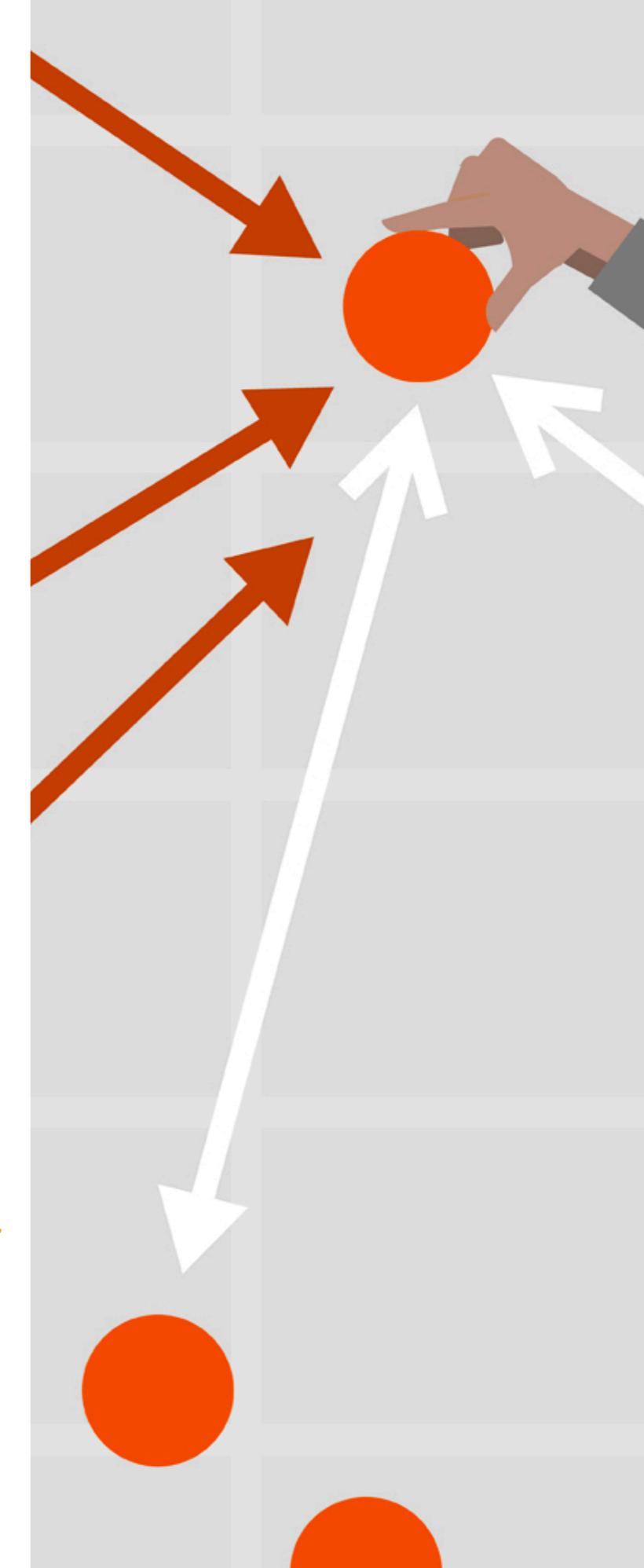
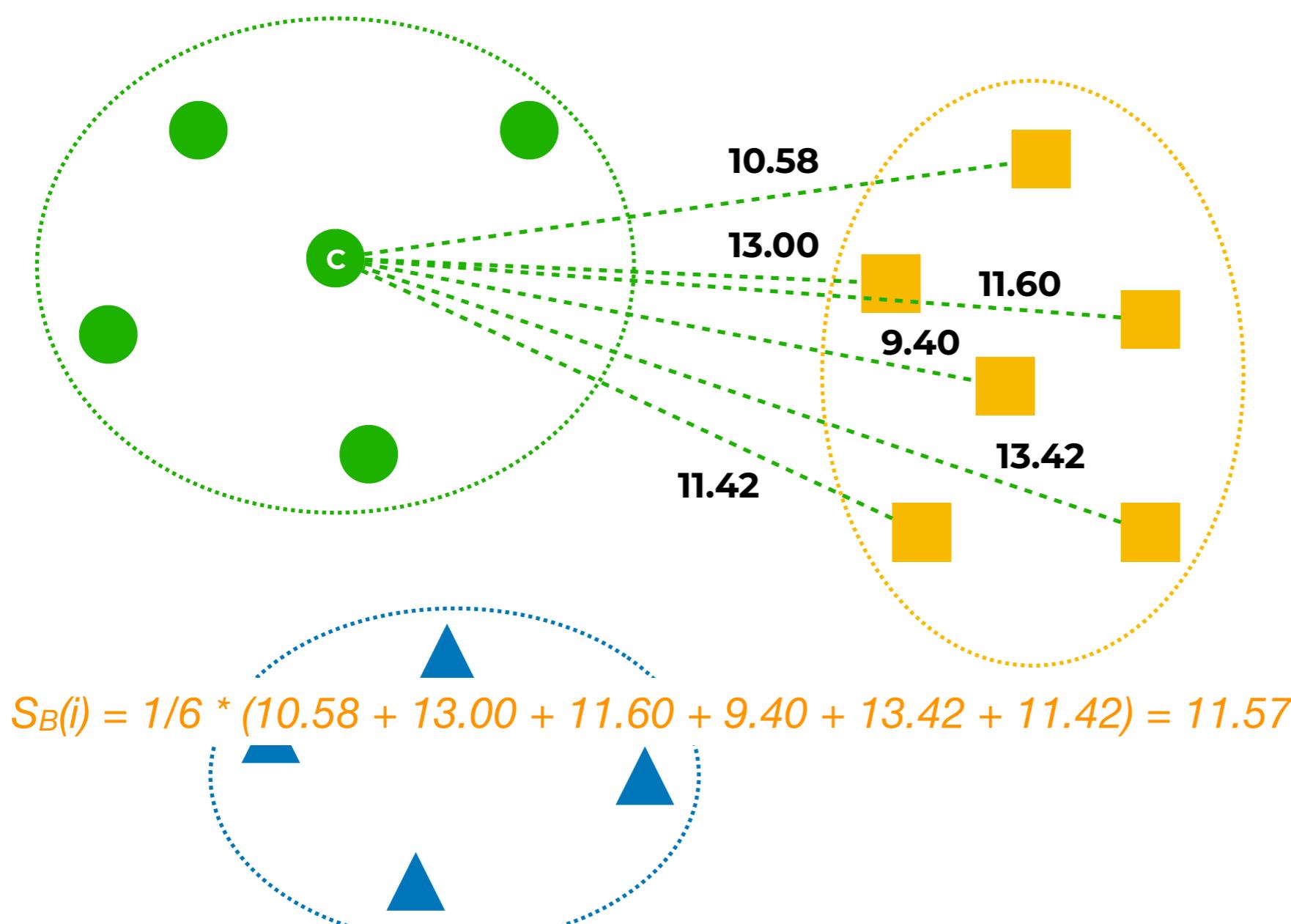
군집의 평가 (Validation)

Silhouette Score



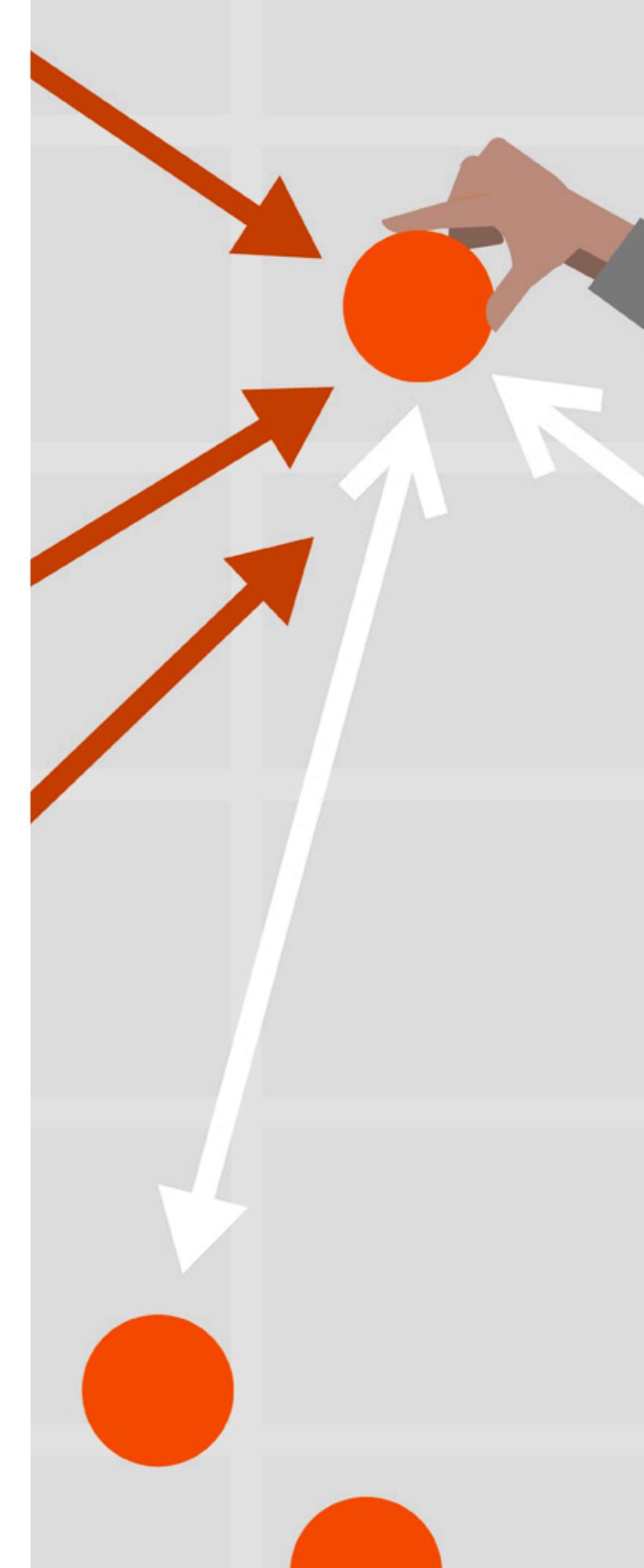
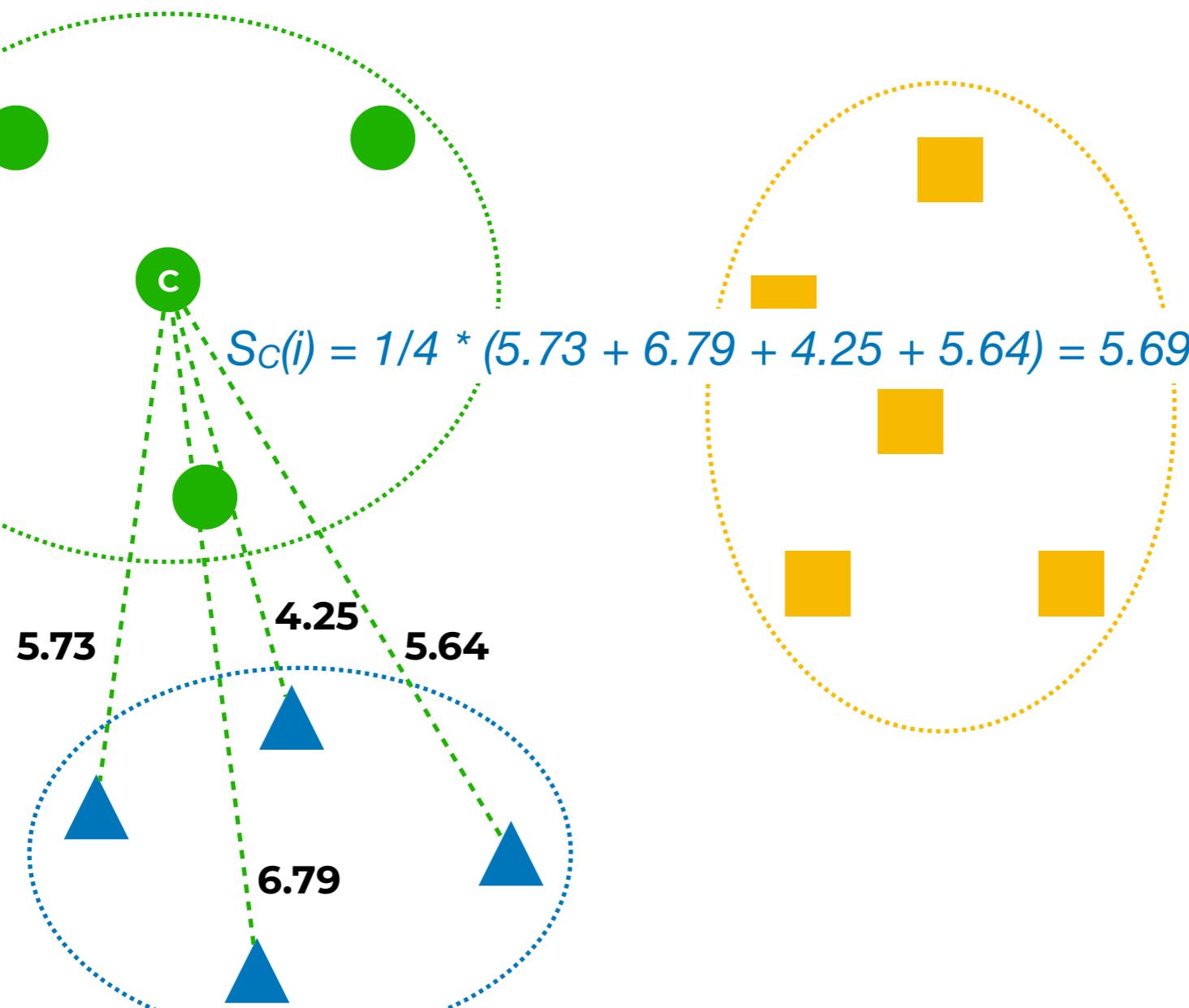
군집의 평가 (Validation)

Silhouette Score



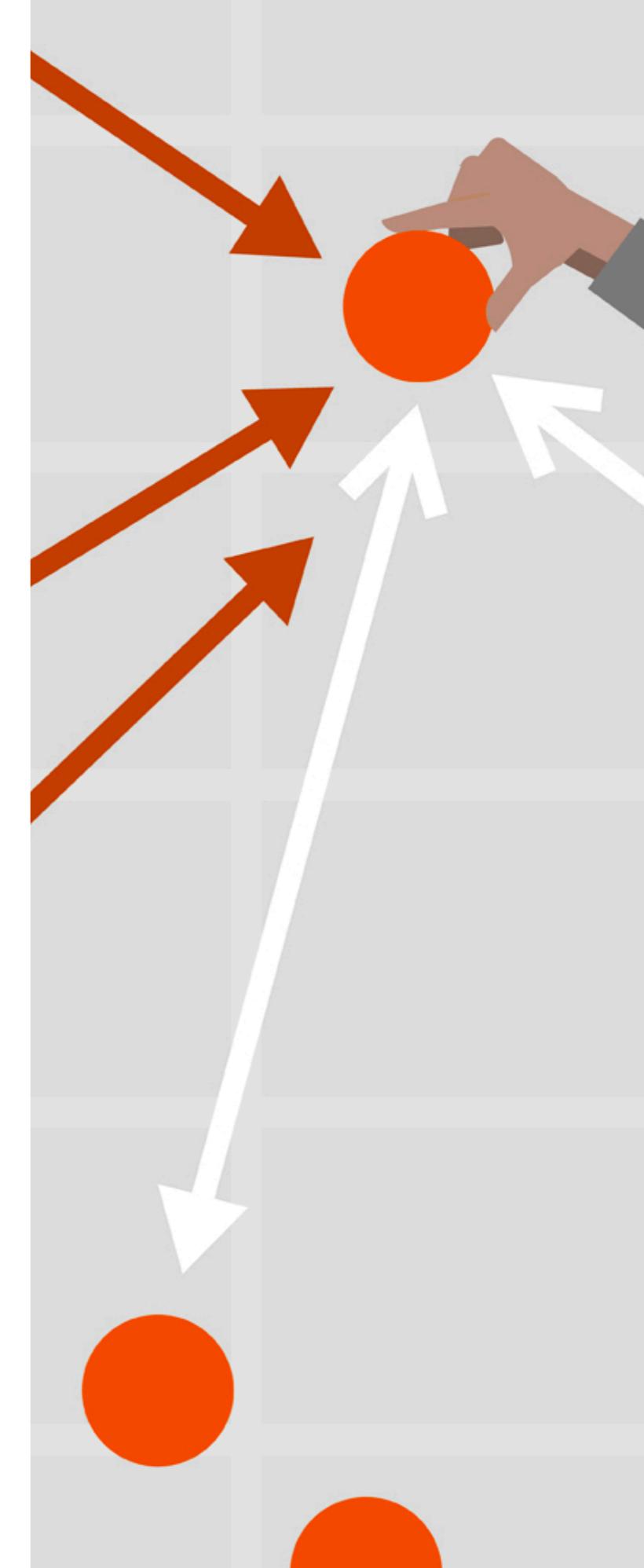
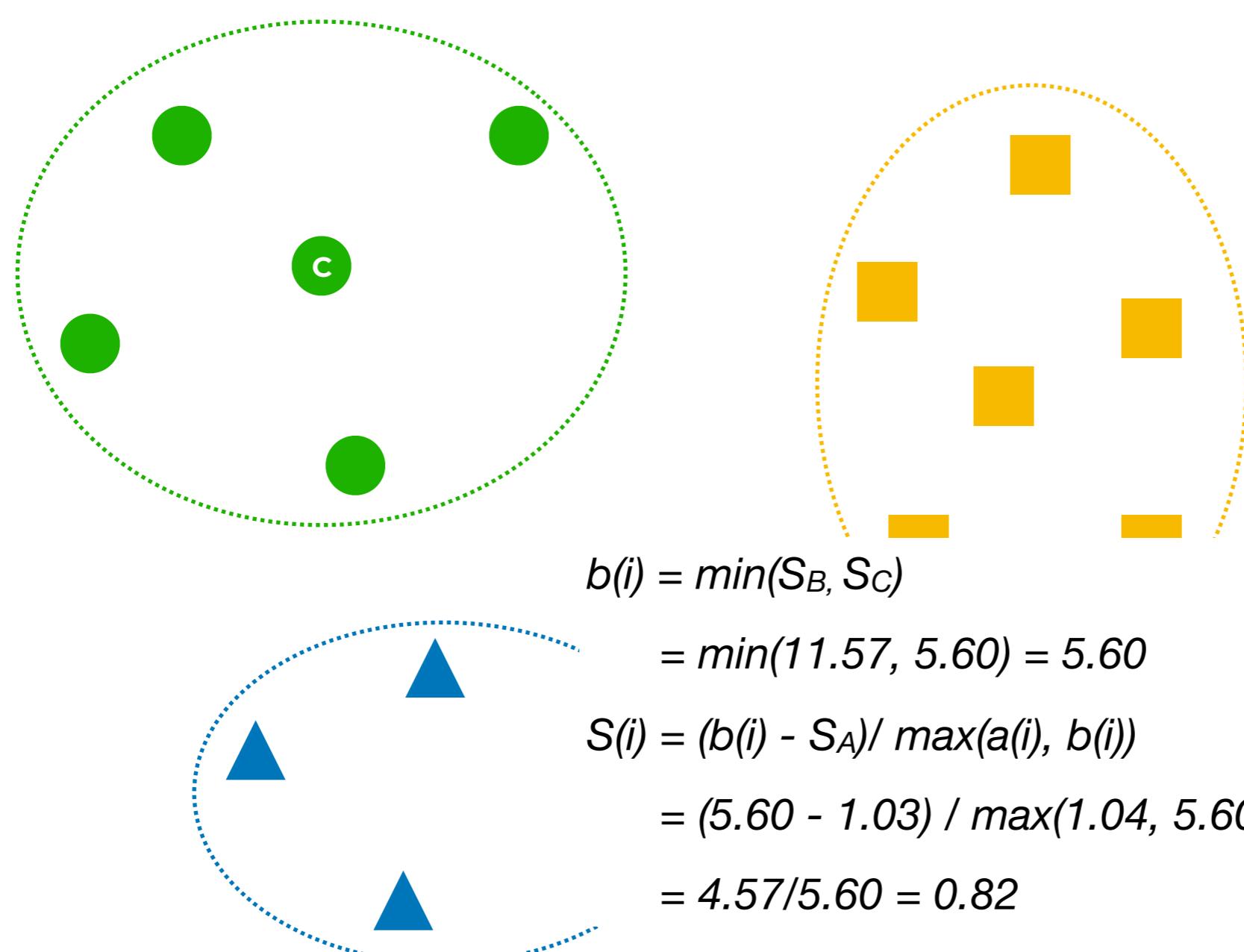
군집의 평가 (Validation)

Silhouette Score



군집의 평가 (Validation)

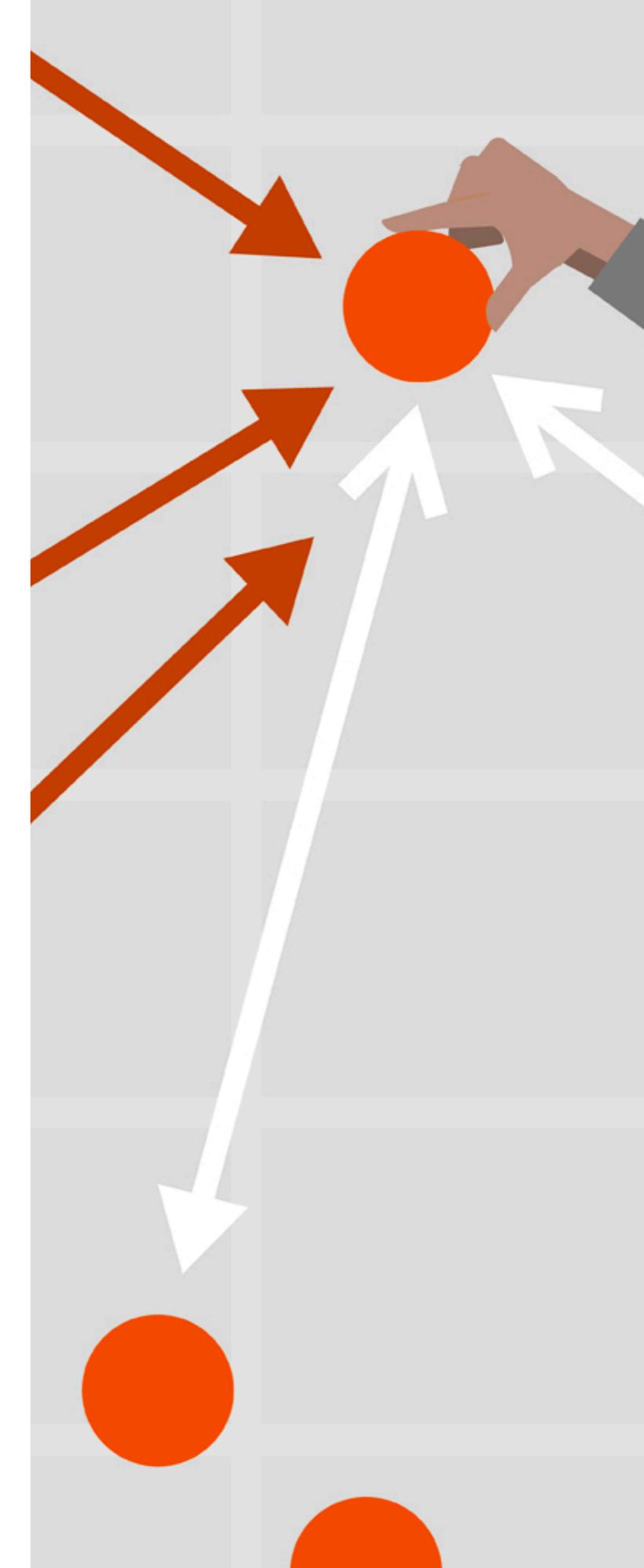
Silhouette Score



토픽모델링 (Topic Modeling)

구조화되지 않은 방대한 문헌집단에서 주제를 찾아내기 위한 방법

- ▶ 뉴스, 블로그, 웹페이지, 기사 등의 형태로 온라인 상에서 방대한 양의 문서가 생성되고 저장되면서 사람들이 찾고자 하는 것을 발견하는 것이 어려워짐
- ▶ 방대한 양의 문서를 관리하고, 검색하고, 이해를 돋기 위한 새로운 도구로 텍스트 마이닝 기법이 주목받기 시작했으며, 그 중 문서 요약과 검색을 위해 토픽모델링 기법이 제안됨
- ▶ 맥락과 관련된 단어들을 이용하여 유사한 의미를 가진 단어들을 클러스터링하는 방식으로 주제를 추론하며, 같은 맥락에서 나타날 가능성이 있는 단어들을 그룹화함



Seeking Life's Bare (Genetic) Necessities

COLD SPRING HARBOR, NEW YORK—How many genes does an organism need to survive? Last week at the genome meeting here,³ two genome researchers with radically different approaches presented complementary views of the basic genes needed for life. One research team, using computer analyses to compare known genomes, concluded that today's organisms can be sustained with just 250 genes, and that the earliest life forms required a mere 128 genes. The other researcher mapped genes in a simple parasite and estimated that for this organism, 800 genes are plenty to do the job—but that anything short of 100 wouldn't be enough.

Although the numbers don't match precisely, those predictions

"are not all that far apart," especially in comparison to the 75,000 genes in the human genome, notes Siv Andersson of Uppsala University in Sweden, who arrived at the 800 number. But coming up with a consensus answer may be more than just a genetic numbers game, particularly as more and more genomes are completely mapped and sequenced. "It may be a way of organizing any newly sequenced genome," explains

Arcady Mushegian, a computational molecular biologist at the National Center for Biotechnology Information (NCBI) in Bethesda, Maryland. Comparing an

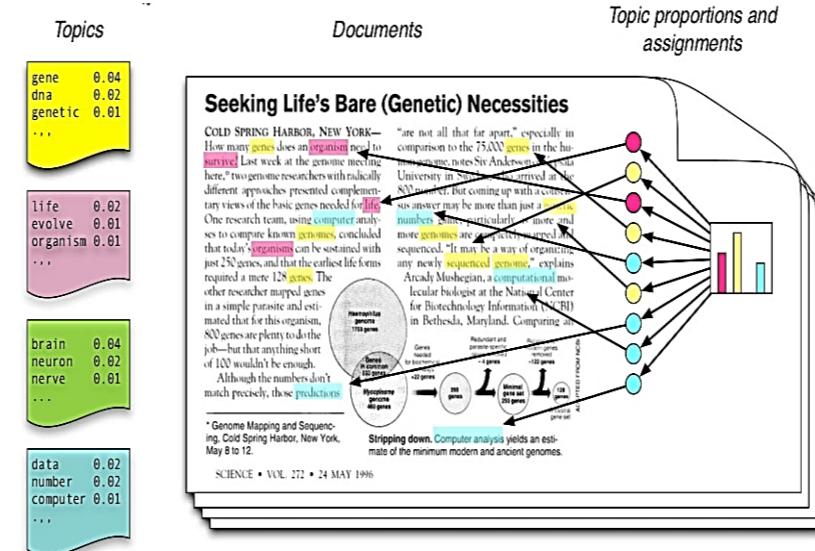
ancestor genome to a modern one, he found that the two share 233 genes in common. By subtracting 22 genes needed for biochemical pathways, he arrived at a minimum gene set of 250 genes.

Redundant and parasite-specific genes removed –122 genes

Related and modern genes removed –4 genes

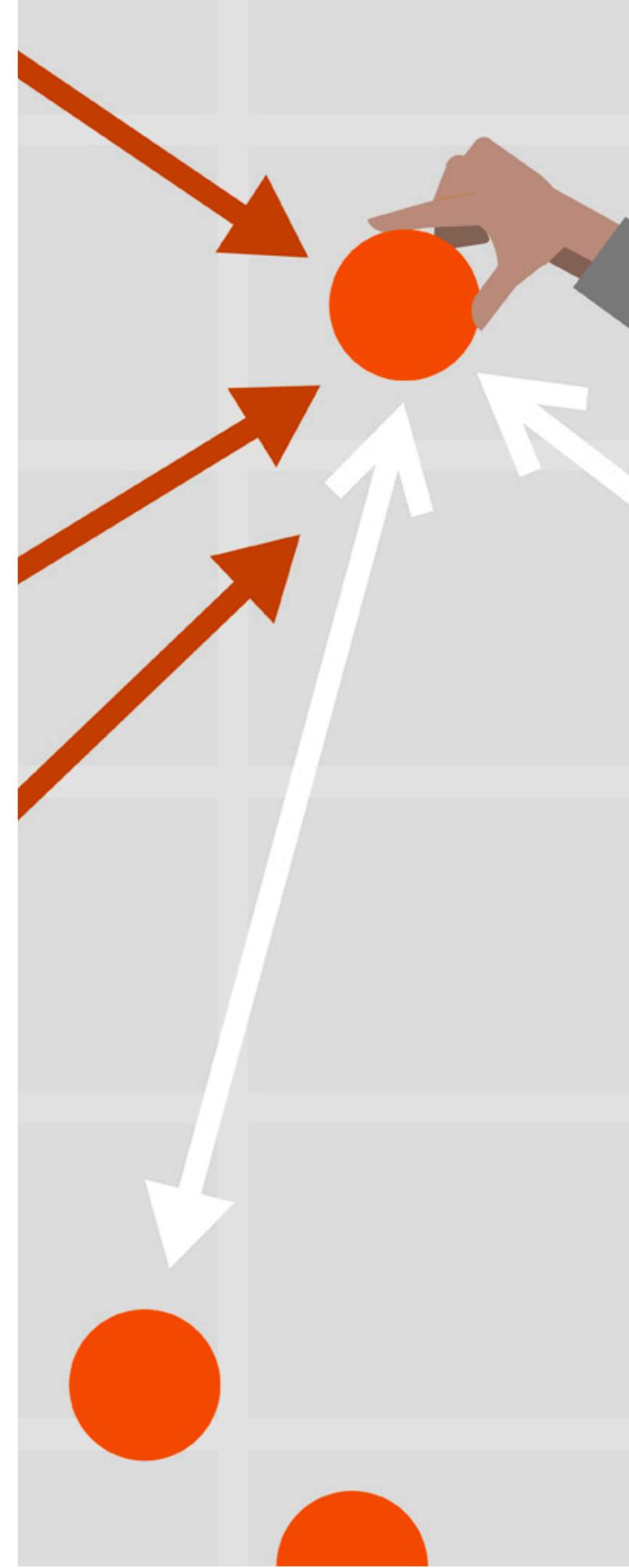
Minimal gene set 250 genes

Ancestral gene set



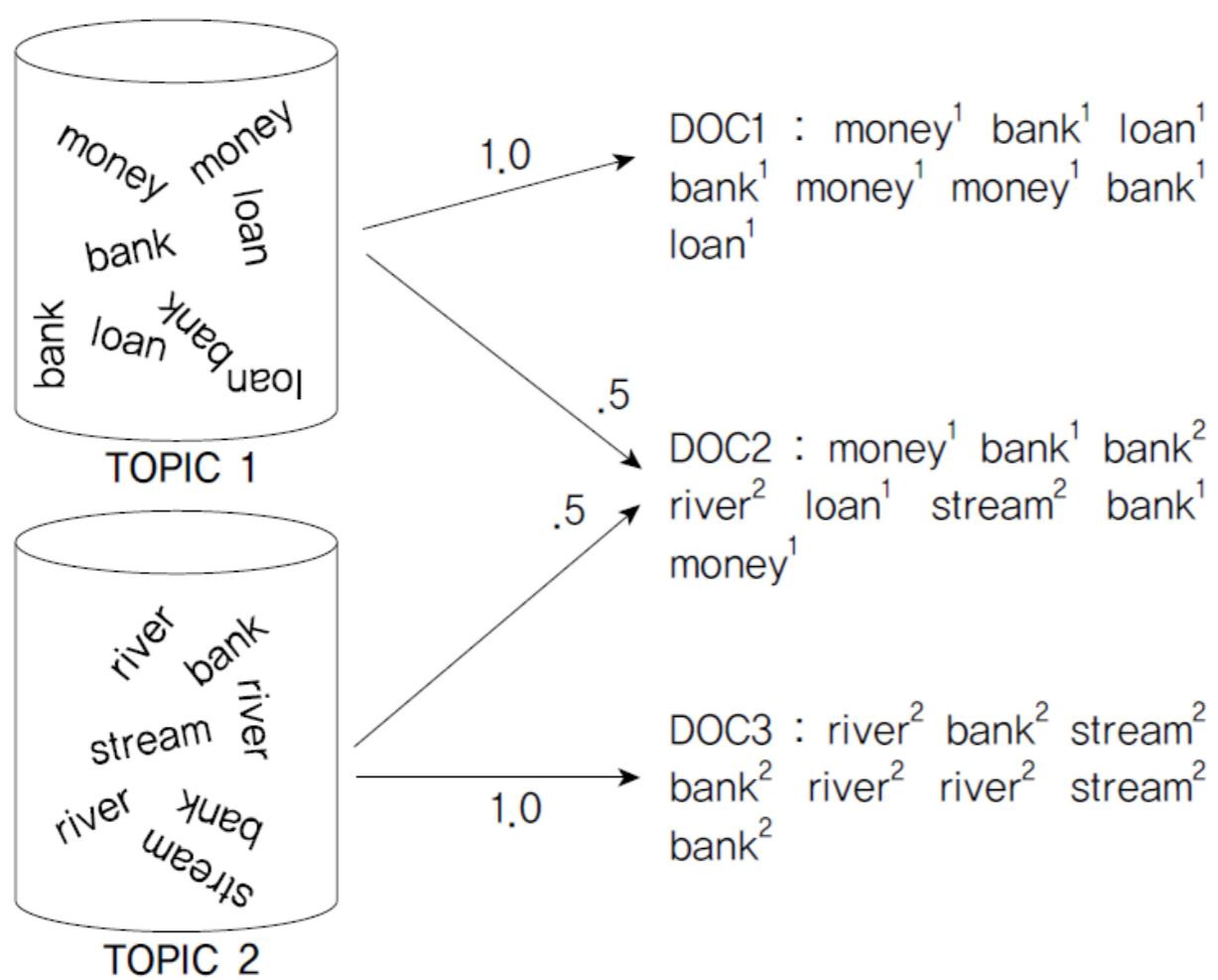
* Genome Mapping and Sequencing, Cold Spring Harbor, New York, May 8 to 12.

토픽모델링 (Topic Modeling)



토픽 모델링: 문헌 생성 모델

- ▶ 토픽 모델링의 문제점
 - 문헌 내의 용어 분포는 알 수 있지만 주제들 (Topic 1, Topic 2)의 용어분포를 사전에 알 수 없음
 - 직접 관찰할 수 있는 문헌 내 용어분포로부터 주제의 용어분포를 추정하는 과정이 필요
 - 잠재 디리클레 할당 (Latent Dirichlet Allocation, LDA) 기법 제안



*Source : 송민, 텍스트 마이닝, 2018., 도서출판 청람.

토픽모델링 (Topic Modeling)

LDA 토픽 모델링

- ▶ 토픽 모델링 기법 중 텍스트 마이닝 분석에서 가장 많이 활용되고 있는 문헌 생성 모델 (generative probabilistic model)
- ▶ LDA (Latent Dirichlet Allocation)
 - 이미 관찰된 변수를 통해 각각의 확률을 계산하여 토픽을 생성하는 사후 추론방법
 - 특이값 분해 (singular value decomposition)를 활용한 LSI에서 발전됨
- ▶ LSI (Latent Semantic Index) : 용어-문헌 행렬의 차원을 축소하는 방법으로 문헌을 표현
- ▶ LDA 기법에서는 문헌 단위에서 각 주제들의 분포로 문헌을 표현

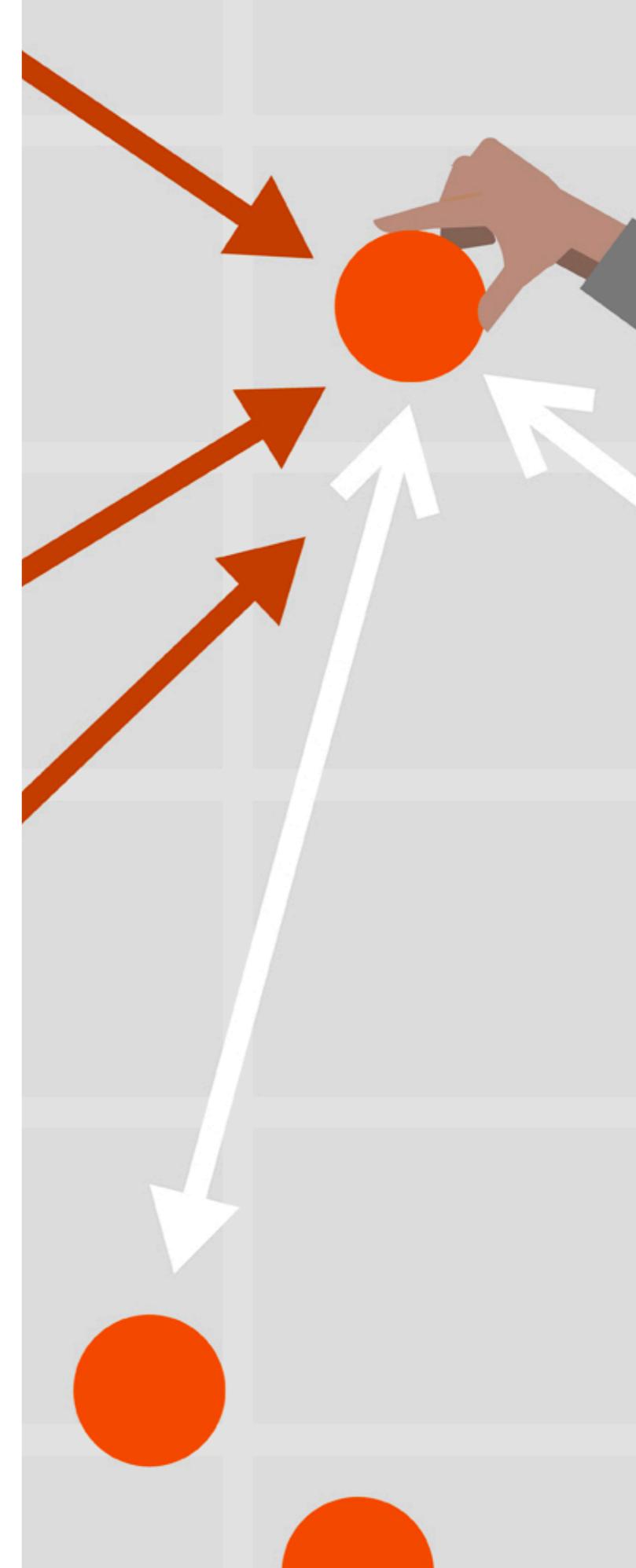
$$\text{LSI} \quad C = U \times \Sigma_{\text{dims} \times \text{dims}} \times V^T_{\text{dims} \times \text{docs}}$$

$$\text{LDA} \quad C = \Phi_{\text{words} \times \text{topics}} \times \Theta_{\text{topics} \times \text{docs}}$$

normalized co-occurrence matrix

mixture component

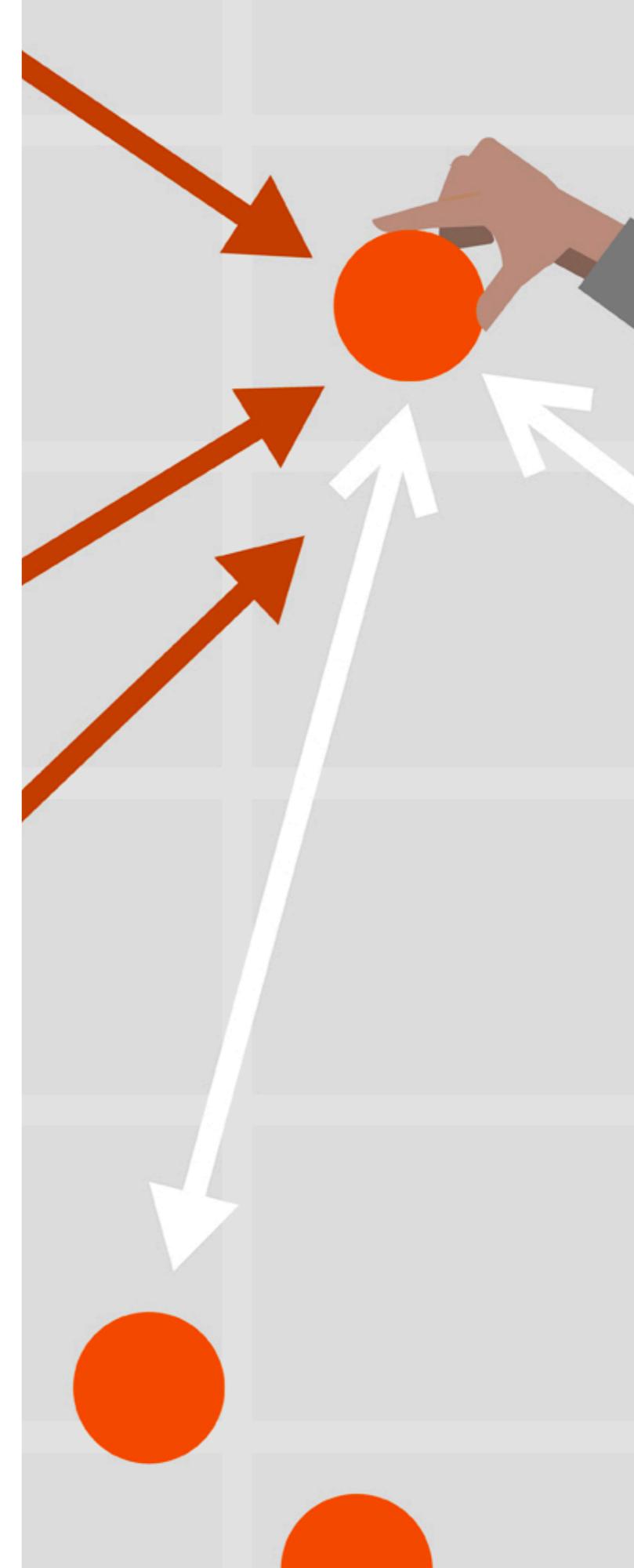
mixture weights



토픽모델링 (Topic Modeling)

LDA 토픽 모델링 수행과정

- ▶ 포아송분포로부터 임의의 문헌 길이 N 을 선택
 α 를 매개변수로 한 디리클레분포로부터 주제분포 θ 를 선택
(α : 문헌집단 내에서 정해지는 변수로 학습을 통해 추정됨)
 1. choose $N \sim Poisson(\xi)$
 - ▶ 어떤 문헌에 대해 주제 벡터인 θ 가 매개변수일 때, 앞에서부터 단어를 하나씩 채울 때마다 θ 로부터 하나의 주제($_z_n$)를 선택
 2. choose $\theta \sim Dir(\alpha)$
 - ▶ 다시 그 주제로부터 단어를 선택
 3. For each word in document:
 - (i) choose a topic $z_n \sim Multinomial(\theta)$
 - (ii) choose a word $w_n \sim Multinomial(\beta)$
- α : 문헌 내 주제분포 θ 를 추정하기 위한 매개변수
- β : 각 용어가 특정 주제에 할당될 사전 확률
- θ : 문헌 내에서 특정 주제가 할당될 사전 확률(prior probability)



E.O.D