2. Unsupervised Learning (1)

빅데이터와 금융자료 분석

김아현

Clustering

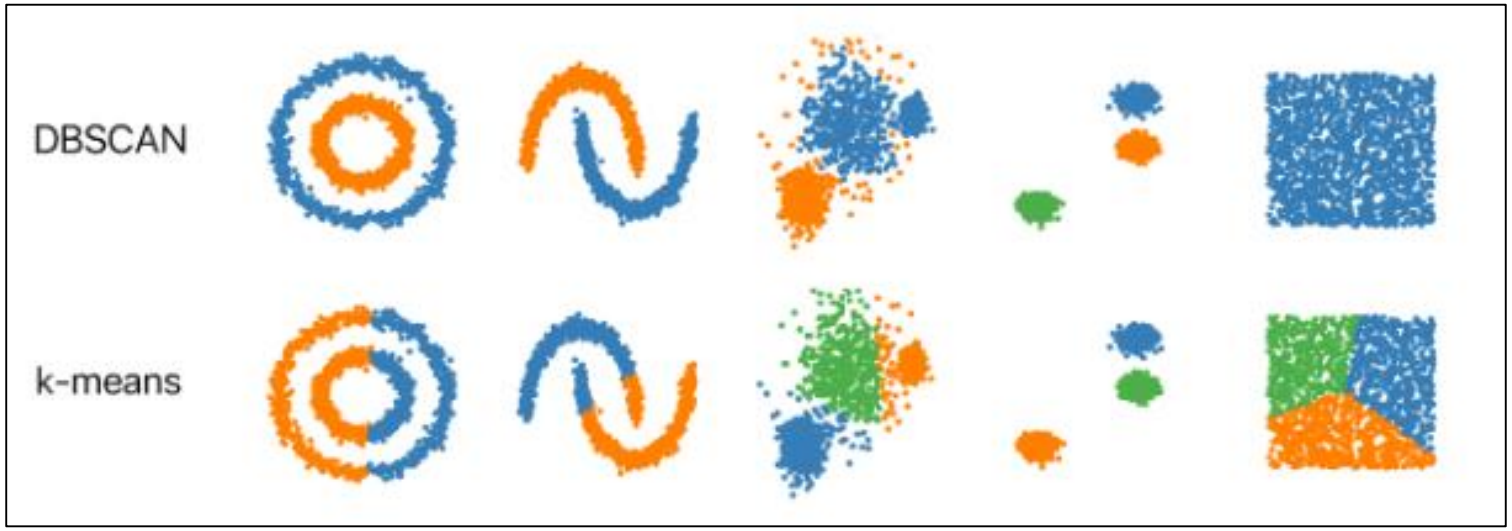
- 군집 분석
  - 군집분석 개요
    - 레이블이 없는 비지도학습용 데이터를 공통적인 패턴을 식별한 뒤 분류하는 방법.
    - 이렇게 얻은 군집 정보는 분류 용도로 사용하거나 다른 학습모델의 특성 변수로 사용할 수 있음.
    - 거리 기반의 모델이 많으며, 분석 전 데이터 표준화가 필요함.
    - 이상치에 민감하므로, 이상치의 제거나 조정이 필요함.
    - 중요하지 않은 특성변수가 모델에 포함되지 않도록 해야 함.

Clustering

- 군집 분석
  - 군집분석 종류
    - 계층적 군집분석 (Hierarchical Clustering)
      - 병합적(agglomerative) 방식, 분리적(divisive) 방식
    - 프로토 타입
      - 연속형 데이터는 K-means, K-median clustering
      - 범주형 데이터는 K-mode clustering 등
    - 분포 기반 : 혼합분포 군집. Gaussian Mixture clustering 등
    - 밀도 기반 : DBSCAN 등

Clustering

- DBSCAN (Density Based Spatial Clustering of Applications with Noise)
  - DBSCAN 개요 및 특징
    - 대규모의 데이터에 적용할 수 있는 밀도 기반의 군집화 알고리즘
    - 간단하고 직관적인 알고리즘임에도 불구하고, 기하학적으로 복잡한 분포를 가지는 데이터에도 효과적인 군집화가 가능함.
    - 군집의 수를 미리 지정할 필요가 없고, 이상치를 효과적으로 제외할 수 있는 것이 특징임.



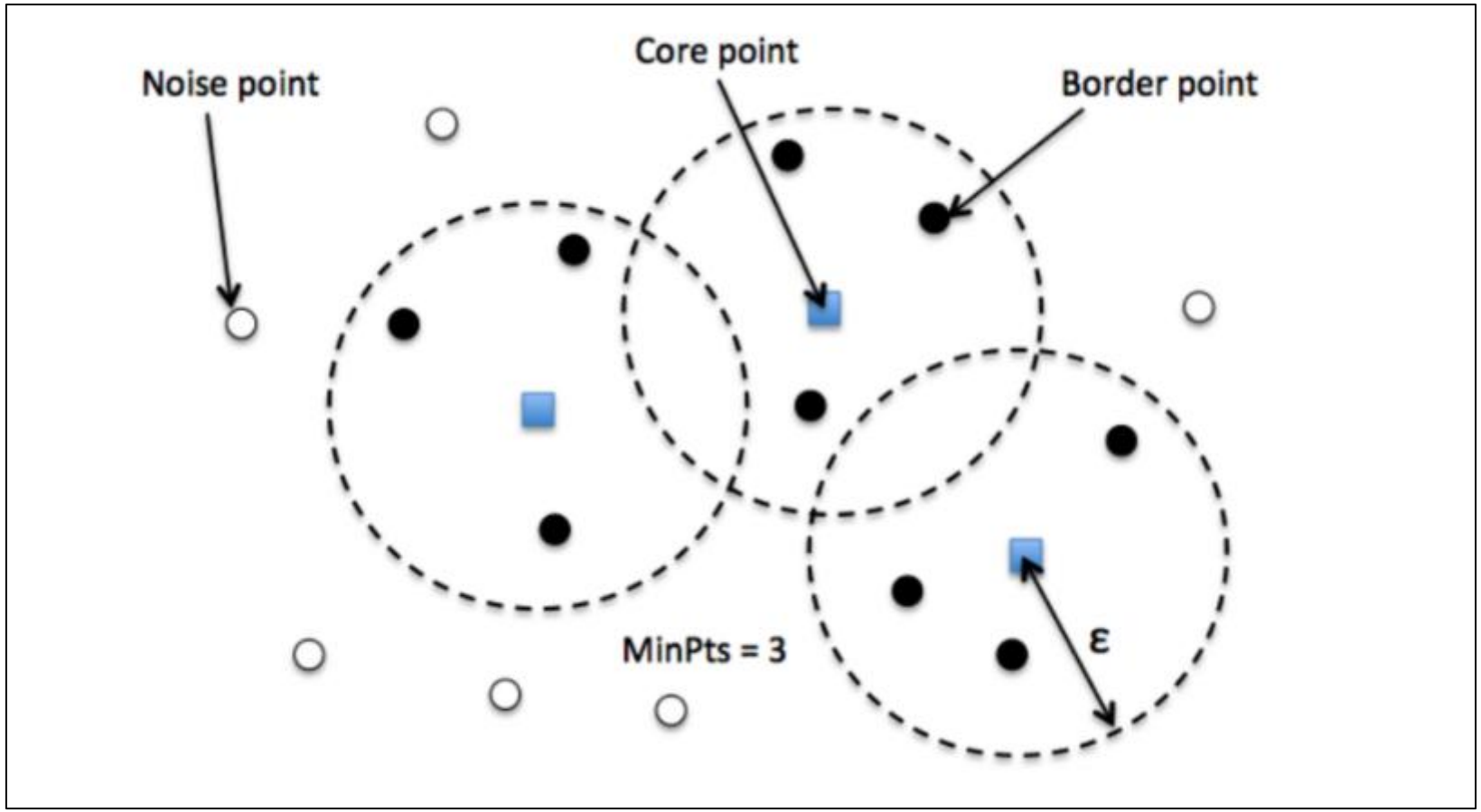
Clustering

- DBSCAN (Density Based Spatial Clustering of Applications with Noise)
  - DBSCAN 알고리즘
    - DBSCAN의 하이퍼파라미터
      - 입실론 (ε) : 개별 데이터를 중심으로 ε의 반경을 가지는 원형의 영역을 주변영역으로 정의함.
      - 최소 데이터개수 ($MinPts$) : 어느 데이터가 핵심포인트가 되기 위해 그 데이터의 입실론 주변영역에 다른 데이터가 몇 개여야 하는지 의미.

    - 데이터 포인트의 분류
      - 이웃포인트 (neighbor point) : 주변영역 내에 위치한 다른 데이터.
      - 핵심포인트 (core point) : 주변영역 내에 $MinPts$개 이상의 이웃포인트가 있는 데이터.
      - 경계포인트 (border point) : 주변영역 내에 $MinPts$개 이상의 이웃포인트를 가지고 있지는 않지만, 핵심포인트를 이웃포인트로 가지고 있는 데이터.
      - 잡음포인트 (noise point) : 주변영역 내에 $MinPts$개 이상의 이웃포인트를 가지고 있지도 않고 핵심포인트를 이웃포인트로 가지고 있지도 않은 데이터.

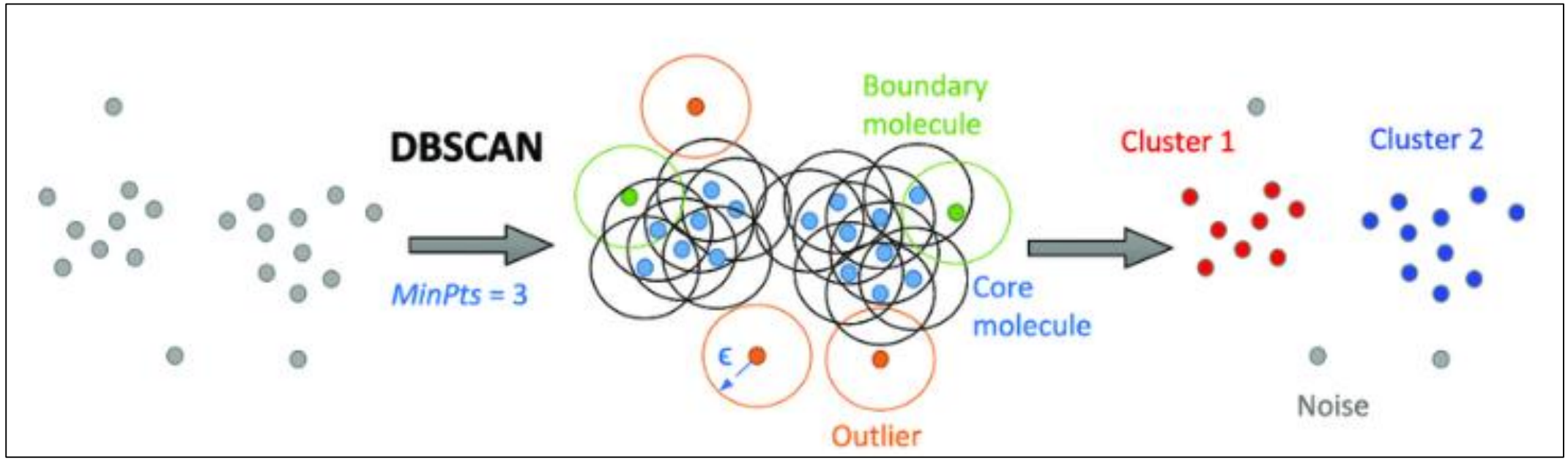
Clustering

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



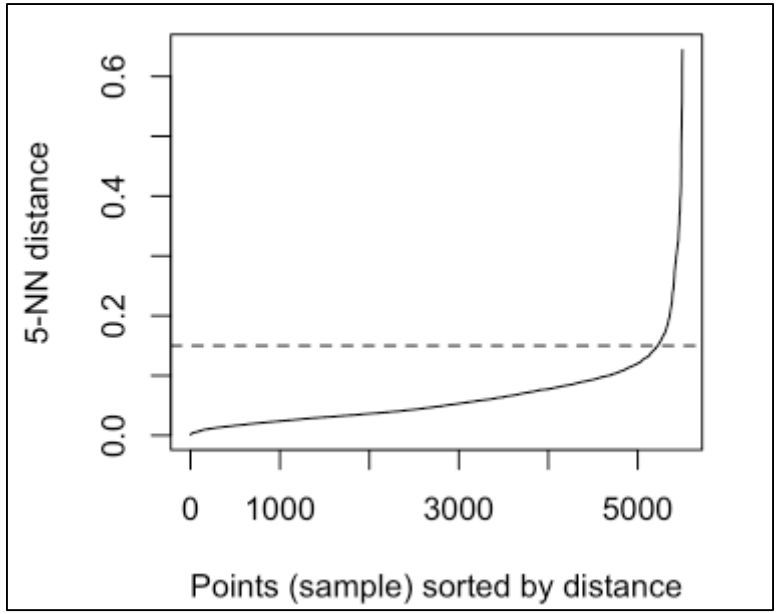
Clustering

- DBSCAN (Density Based Spatial Clustering of Applications with Noise)
  - DBSCAN 알고리즘
    ① 모든 데이터포인트를 세 종류로 분류한다.
    ② 잡음포인트는 버린다.
    ③ ε 반경 이내의 핵심포인트를 연결하여 클러스터를 형성한다.
    ④ 경계포인트는 자신의 ε 반경 내의 핵심포인트 중 하나가 소속된 클러스터에 할당한다.



Clustering

- DBSCAN (Density Based Spatial Clustering of Applications with Noise)
  - DBSCAN 알고리즘
    - $k-distance$ 를 활용한 적정 ε의 결정
      - 각 관찰점 별 $k-distance$를 구했을 때,
        - 군집의 밀도가 높으면 $k-distance$ 는 전반적으로 작아지고,
        - 군집의 밀도가 낮으면 $k-distance$ 는 전반적으로 커짐.
      - 모든 관찰점에 대해 구한 $k-distance$ 를 오름차순 정렬하여 시각화 한 뒤, $k-distance$ 가 급격히 증가하기 시작할 때의 값을 ε으로 둔다.



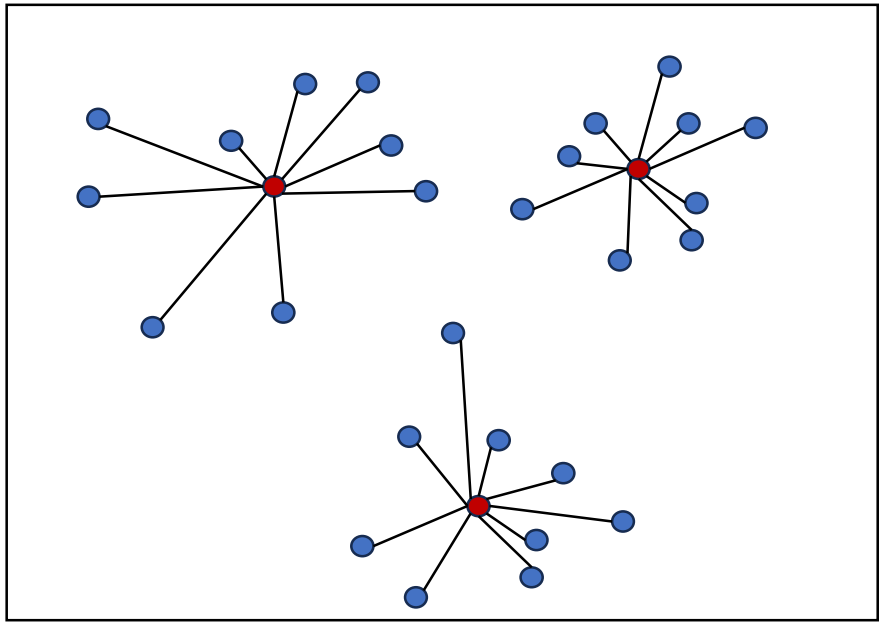
Clustering

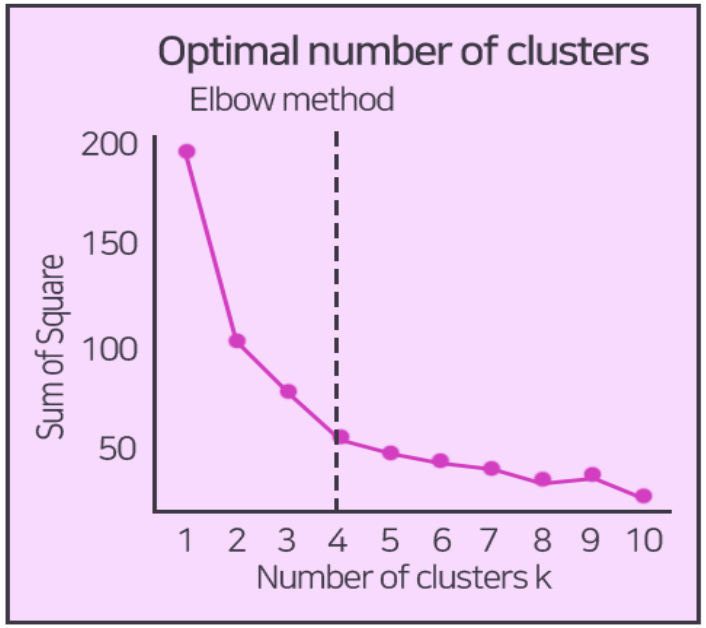
- DBSCAN (Density Based Spatial Clustering of Applications with Noise)
  - DBSCAN 알고리즘
    - 장단점
      - 비선형의 복잡한 형상을 찾을 수 있으며, 어떤 군집에도 속하지 못하는 노이즈를 구분할 수 있음.
      - 밀도가 높은 곳에 집중하기 때문에 밀도가 낮은 곳의 데이터는 하나의 군집으로 인식하지 못하고 노이즈가 될 수 있음.
      - K-평균 또는 병합적 군집분석에 비해 모델링 시간이 오래 걸리는 편임.

Clustering

- 군집에 관한 타당성 지표 (Validity Measure)
  - SSE (Sum of Squared Error)
    - 관찰치를 x, i번째 군집을 C_i, 군집의 개수를 K라고 할 때,

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

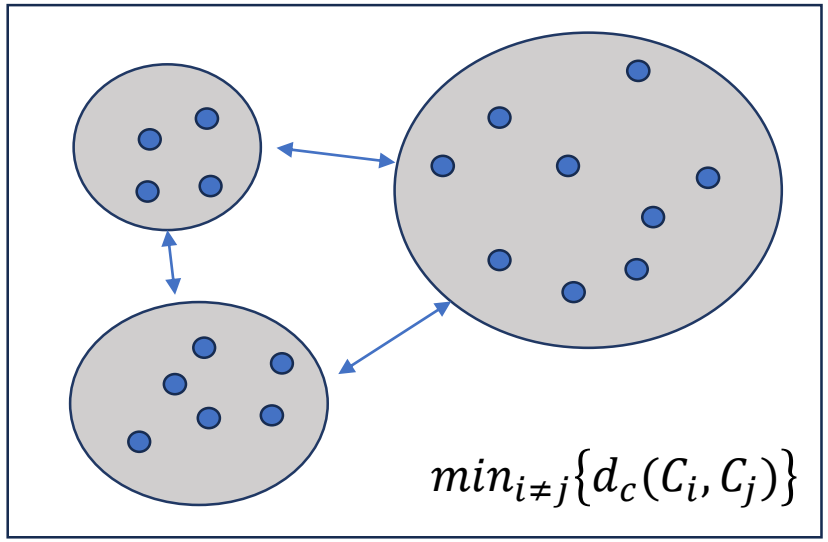


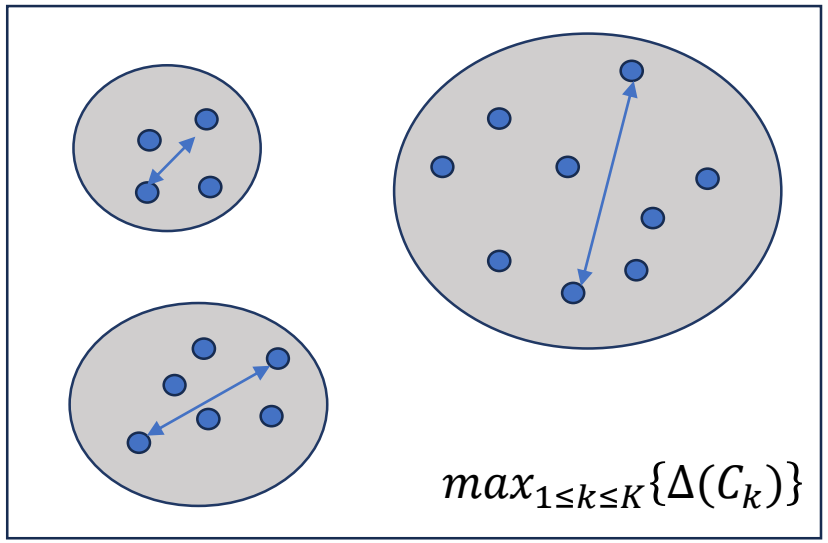


Clustering

- 군집에 관한 타당성 지표 (Validity Measure)
  - Dunn Index
    - i번째 군집을 C_i, 군집의 개수를 K라고 하고,
    - 두 군집 C_i 와 C_j간 거리를 $d_c(C_i, C_j)$, 어느 군집 C_k 내 거리를 $\Delta(C_k)$라고 할 때,

$$I(C) = \frac{min_{i \neq j}\{d_c(C_i, C_j)\}}{max_{1 \leq k \leq K}\{\Delta(C_k)\}}$$





Clustering

- 군집에 관한 타당성 지표 (Validity Measure)
  - Dunn Index
    - 군집 내 거리 $\Delta(C_k)$

      - Complete diameter distance
        - $\Delta(C) = \max_{x_i, x_j \in C} d(x_i, x_j)$

      - Average diameter distance
        - $\Delta(C) = \frac{2}{N_C(N_C-1)} \sum_{x_i, x_j \in C, i \neq j} d(x_i, x_j)$

      - Centroid diameter distance
        - $\Delta(C) = \frac{1}{N_C} \sum_{x_i \in C} d(x_i, \mu)$, $\mu = \frac{1}{N_C} \sum_{x_i \in C} x_i$

Clustering

- 군집에 관한 타당성 지표 (Validity Measure)
  - Dunn Index
    - 군집 간 거리 $d_c(C_i, C_j)$

      - Single linkage
        - $d_c(C_1, C_2) = \min d(a, b)\,, \quad a \in C_1, b \in C_2$

      - Complete linkage
        - $d_c(C_1, C_2) = \max\ d(a, b)\,, \quad a \in C_1, b \in C_2$

      - Average linkage
        - $d_c(C_1, C_2) = \frac{1}{N_{C_1} N_{C_2}} \sum_{i=1}^{N_{C1}} \sum_{j=1}^{N_{C2}} d(a_i, b_j) \quad , \ a_i \in C_1, b_j \in C_2$

Clustering

- 군집에 관한 타당성 지표 (Validity Measure)
  - Dunn Index
    - 군집 간 거리 $d_c(C_i, C_j)$

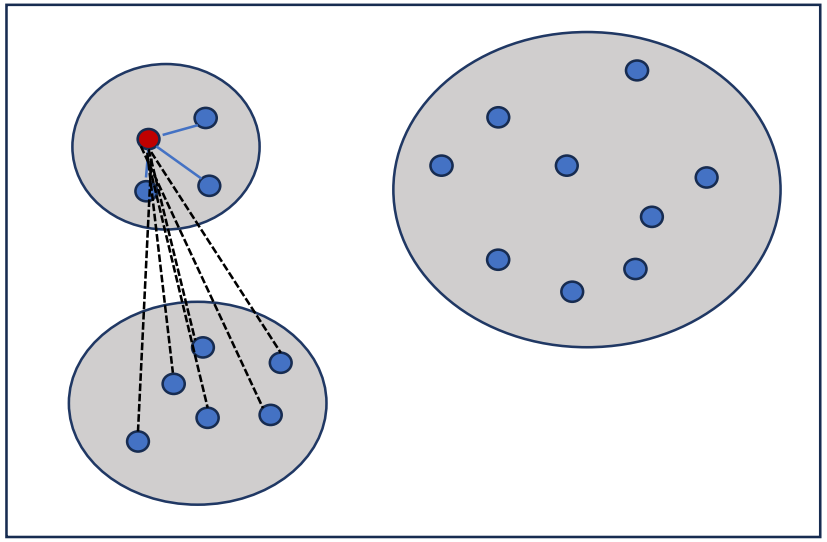
      - Centroid linkage
        - $d_c(C_1, C_2) = d(\mu_1, \mu_2)$, $\mu_k = \frac{1}{N_{C_k}} \sum_{x_i \in C_k} x_i$, $k = 1, 2$

      - Average of Centroid linkage
        - $d_c(C_1, C_2) = \frac{1}{N_{C_1} + N_{C_2}} \left(\sum_{i=1}^{N_{C1}} d(a_i, \mu_2) + \sum_{j=1}^{N_{C2}} d(b_j, \mu_1) \right)$

Clustering

- 군집에 관한 타당성 지표 (Validity Measure)
  - Silhouette Score
    - 관찰치 i의 Silhouette Value : $s(i)$
      - $a(i)$: 관찰치 i와 같은 군집 내에 있는 다른 관찰치들과의 거리의 평균
      - $b(i)$: 관찰치 i의 군집에서 가장 가까운 다른 군집 내에 있는 관찰치들 과의 거리의 평균

$$s(i) = \frac{b(i)-a(i)}{max\{a(i),b(i)\}} = \begin{cases} 1 - \frac{a(i)}{b(i)}, & if\ a(i) < b(i) \\ 0 & if\ a(i) = b(i) \\ \frac{b(i)}{a(i)} - 1, & if\ a(i) > b(i) \end{cases}$$

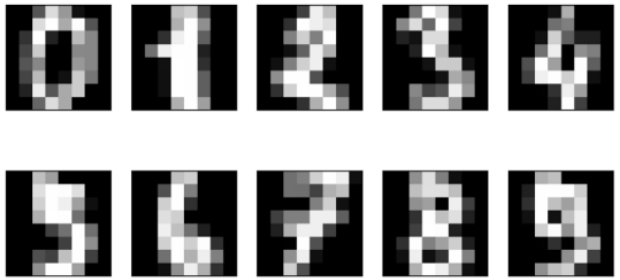


    - Silhouette Score
      - $\bar{s}(J)$: J번째 군집 내 관찰치들의 평균 Silhouette Value

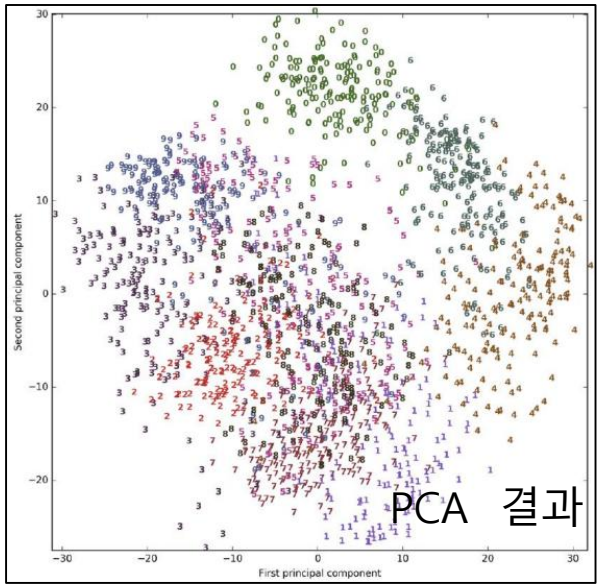
$$SC = \max_{1 \le J \le K} \bar{s}(J)$$

Dimension Reduction

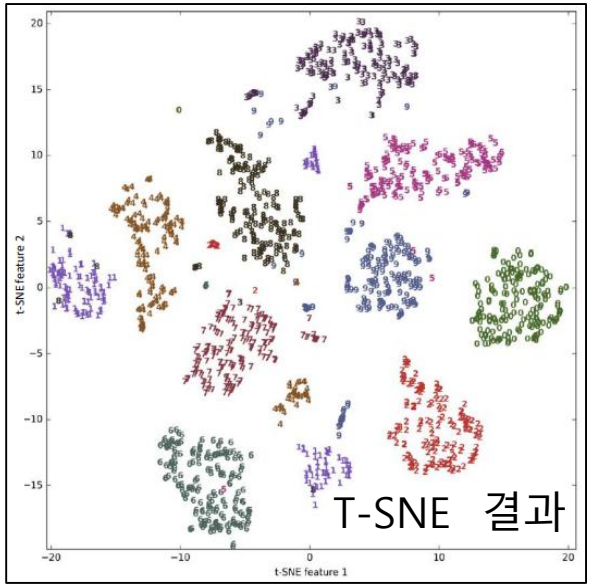
- t-SNE (t-distributed Stochastic Neighbor Embedding, t-확률적 이웃 임베딩)
  - t-SNE 개요
    - 비선형적 차원축소 기법으로 주로 데이터 시각화에 활용됨.
    - 특히 고차원 공간에서의 데이터 클러스터의 시각화에 유용함.
    - 원래의 공간에서 데이터 포인트들 사이의 거리를 최대한 유지하는 저차원 공간에서의 표현을 찾고자 함.



<digits 데이터>
0~9 사이의 숫자에 대한 손글씨 데이터
8×8 흑백이미지. 1797개



PCA 결과



T-SNE 결과

Dimension Reduction

- t-SNE (t-distributed Stochastic Neighbor Embedding, t-확률적 이웃 임베딩)
  - t-SNE 알고리즘
    - 고차원 및 저차원 공간에서의 유사성의 분포
      - 원 공간에서의 유사도 분포
        - $p_{j|i}$: 어떤 관찰치 $\boldsymbol{x}_i$에 대하여, $\boldsymbol{x}_i$가 다른 관찰치 $\boldsymbol{x}_j$를 이웃으로 선택할 확률.

$$p_{j|i} = \frac{exp\left(-\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2 / 2\sigma_i^2\right)}{\sum_{k \neq i} exp\left(-\|\boldsymbol{x}_i - \boldsymbol{x}_k\|^2 / 2\sigma_i^2\right)}$$

          - $\boldsymbol{x}_i$를 중심으로 하는 가우시안 분포를 이용함.
          - $\boldsymbol{x}_i$와 가까울수록 이웃(Nearest Neighbor)으로 선택될 확률이 높아짐.
          - σ_i는 이웃의 범위를 결정함.
            - σ_i가 작으면 가까운 이웃만 0보다 큰 확률을 가짐.
            - σ_i가 크면 대부분의 관찰치들이 비슷한 확률을 가짐.
          - σ_i는 <u>각 관찰치 별로 다른 값을 가지도록</u> 정의됨!
        - $\boldsymbol{x}_i$와 $\boldsymbol{x}_j$의 확률이 대칭이 되도록 보정

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2n}$$

Dimension Reduction

- t-SNE (t-distributed Stochastic Neighbor Embedding, t-확률적 이웃 임베딩)
  - t-SNE 알고리즘
    - 고차원 및 저차원 공간에서의 유사성의 분포
      - Perplexity
        - 어떤 관찰치 $\boldsymbol{x}_i$가 주어졌을 때, 다른 모든 관찰치들에 대한 조건부 확률분포를 P_i 라고 할 때,

$$Perplexity(P_i) = 2^{H(P_i)}$$

        - $H(P_i)$는 P_i에 대한 entropy

$$H(P_i) = -\sum_j p_{j|i} \log_2(p_{j|i})$$

Dimension Reduction

- t-SNE (t-distributed Stochastic Neighbor Embedding, t-확률적 이웃 임베딩)
  - t-SNE 알고리즘
    - 고차원 및 저차원 공간에서의 유사성의 분포
      - Perplexity
        - Perplexity는 $\boldsymbol{x}_i$에 대한 유효한 이웃의 수를 결정함. KNN의 smooth 버전.
        - 어떤 고정된 perplexity를 설정하면, 이를 만족하도록 bisection 방식으로 각 관찰치 $\boldsymbol{x}_i$의 σ_i를 구함.
        - Perplexity는 5-50 사이의 값으로 설정하였을 때, 일반적으로 잘 작동함.
          - Low perplexity = small σ_i
          - High perplexity = large σ_i

Dimension Reduction

- t-SNE (t-distributed Stochastic Neighbor Embedding, t-확률적 이웃 임베딩)
  - t-SNE 알고리즘
    - 고차원 및 저차원 공간에서의 유사성의 분포
      - 임베딩 공간에서의 유사도 분포
        - $\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_n$ $(\in \mathbb{R}^D)$에 대한 저차원의 임베딩을 $\boldsymbol{z}_1, \boldsymbol{z}_2, \dots, \boldsymbol{z}_n$ $(\in \mathbb{R}^d,\ d < D)$라고 할 때, $\boldsymbol{z}_i$를 기준으로 $\boldsymbol{z}_j$를 이웃으로 선택할 확률 q_{ij} 은 아래와 같이 정의함.

$$q_{ij} = \frac{\left(1+\|\boldsymbol{z}_i - \boldsymbol{z}_j\|^2\right)^{-1}}{\sum_k \sum_{l \neq k} (1+\|\boldsymbol{z}_k - \boldsymbol{z}_l\|^2)^{-1}}$$

        - 자유도가 1인 t분포에 해당함.

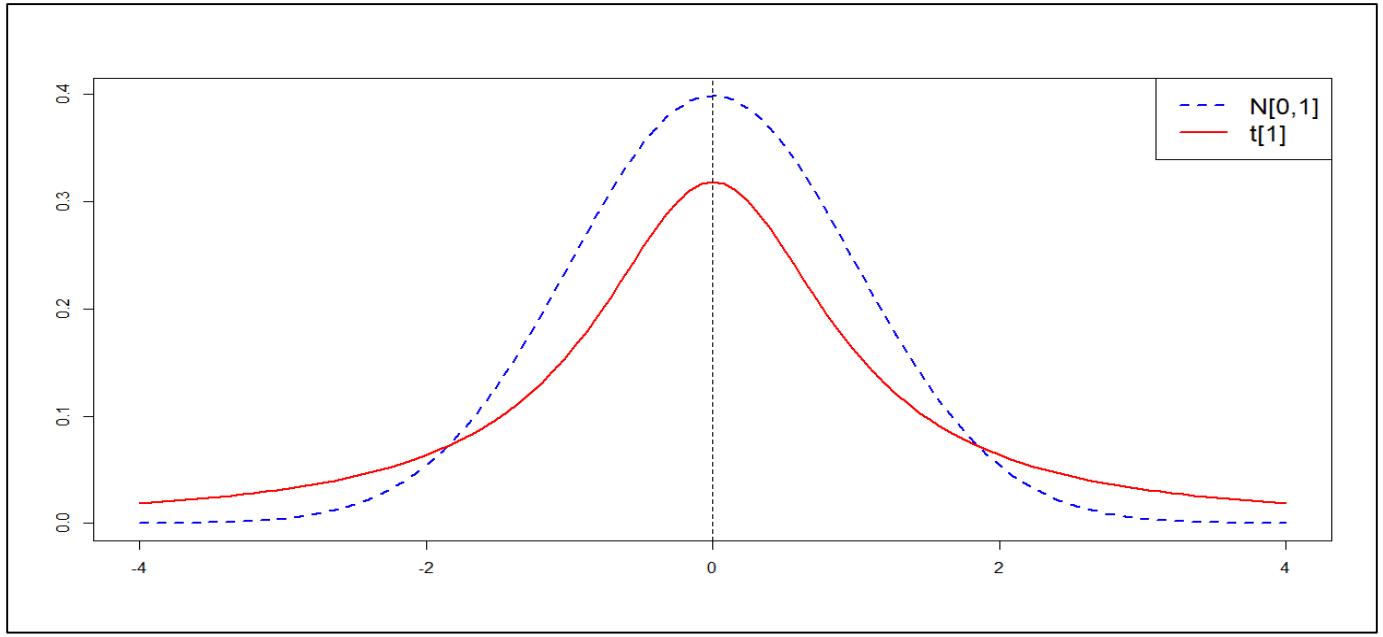
Dimension Reduction

- t-SNE (t-distributed Stochastic Neighbor Embedding, t-확률적 이웃 임베딩)
  - t-SNE 알고리즘
    - Optimization
      - 고차원의 원 공간과 저차원 공간에서의 유사성 분포가 가능한 같아지도록 함.
      - KL divergence 를 이용하여 고차원 및 저차원 공간에서의 유사성 확률 분포의 차이를 측정함.
      - KL divergenc를 비용함수도 두고, 이를 최소로 만드는 저차원 공간 상의 데이터 포인트들의 위치 $\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_n$ 를 경사하강법을 적용하여 갱신함.

        - $Cost = \sum_i \sum_j p_{ij} log\left(\frac{p_{ij}}{q_{ij}}\right)$
        - $\frac{\partial Cost}{\partial \boldsymbol{z}_i} = 4\sum_j (\boldsymbol{z_i} - \boldsymbol{z}_j)(p_{ij} - q_{ij})\left(1 + \|\boldsymbol{z}_i - \boldsymbol{z}_j\|^2\right)^{-1}$

Dimension Reduction

- t-SNE (t-distributed Stochastic Neighbor Embedding, t-확률적 이웃 임베딩)
  - 특징
    - 새로운 데이터의 변환을 허용하지 않음.
    - 데이터 크기에 따라 상당한 시간이 소요될 수 있음.
    - 유용한 비선형 변환
      - 꼬리가 두꺼운 t-분포는 crowding 문제를 해결.
      - 가까운 점은 더 가까워지고, 먼 점은 더 멀어지도록 이동시킴.

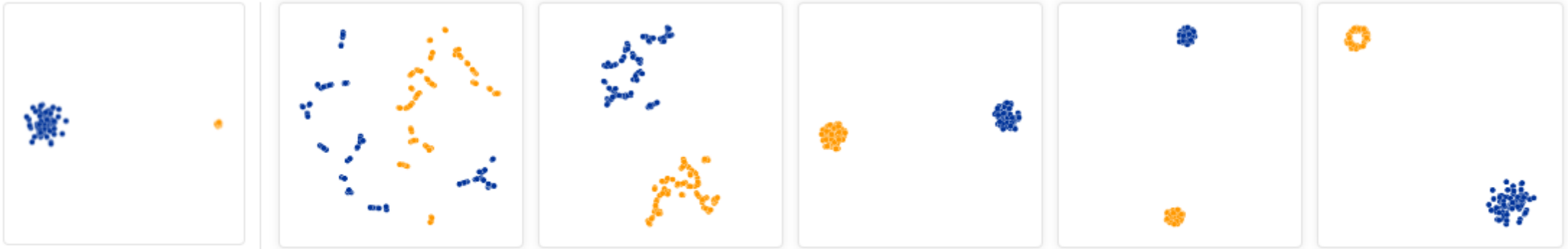


Dimension Reduction

- t-SNE (t-distributed Stochastic Neighbor Embedding, t-확률적 이웃 임베딩)
  - t-SNE 적용

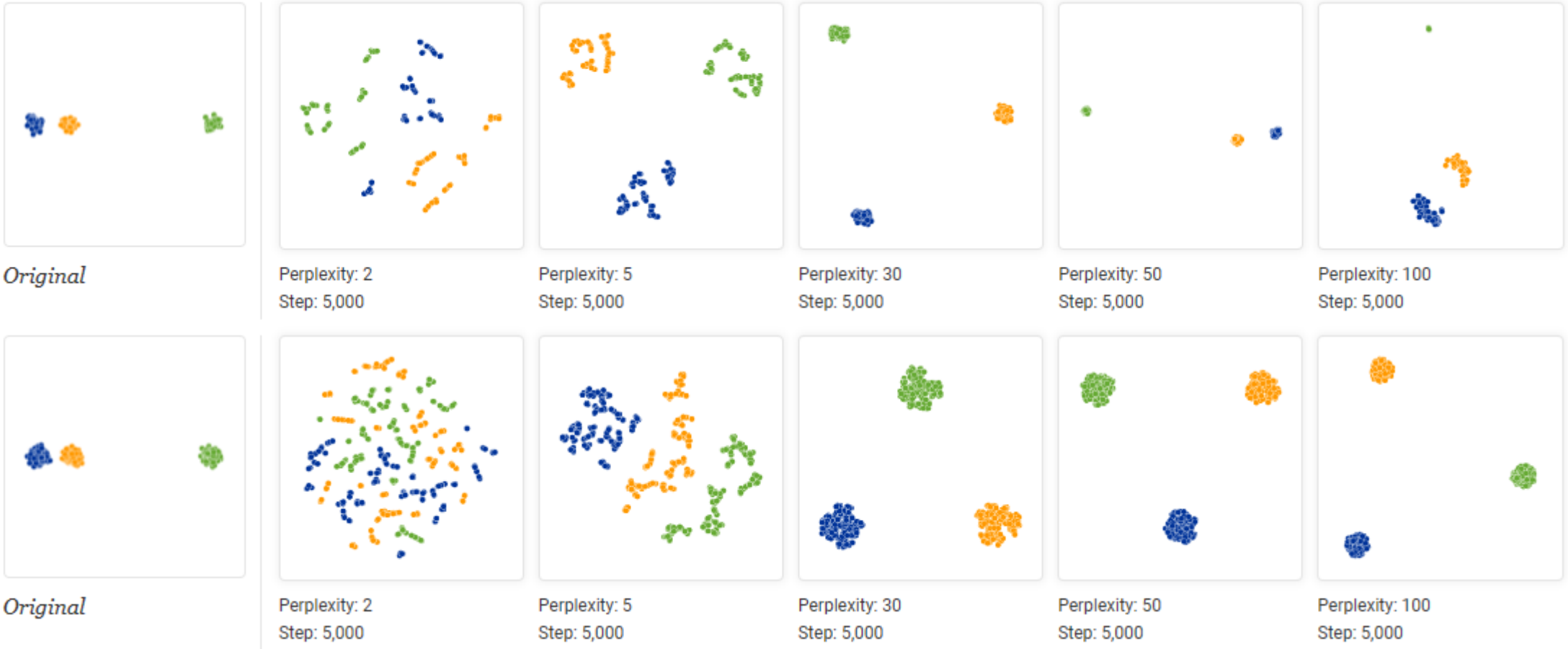
Dimension Reduction

- t-SNE (t-distributed Stochastic Neighbor Embedding, t-확률적 이웃 임베딩)
  - t-SNE 적용



Dimension Reduction

- t-SNE (t-distributed Stochastic Neighbor Embedding, t-확률적 이웃 임베딩)
  - t-SNE 적용



Dimension Reduction

- t-SNE (t-distributed Stochastic Neighbor Embedding, t-확률적 이웃 임베딩)
  - t-SNE 적용

Original

Perplexity: 2
Step: 5,000

Perplexity: 5
Step: 5,000

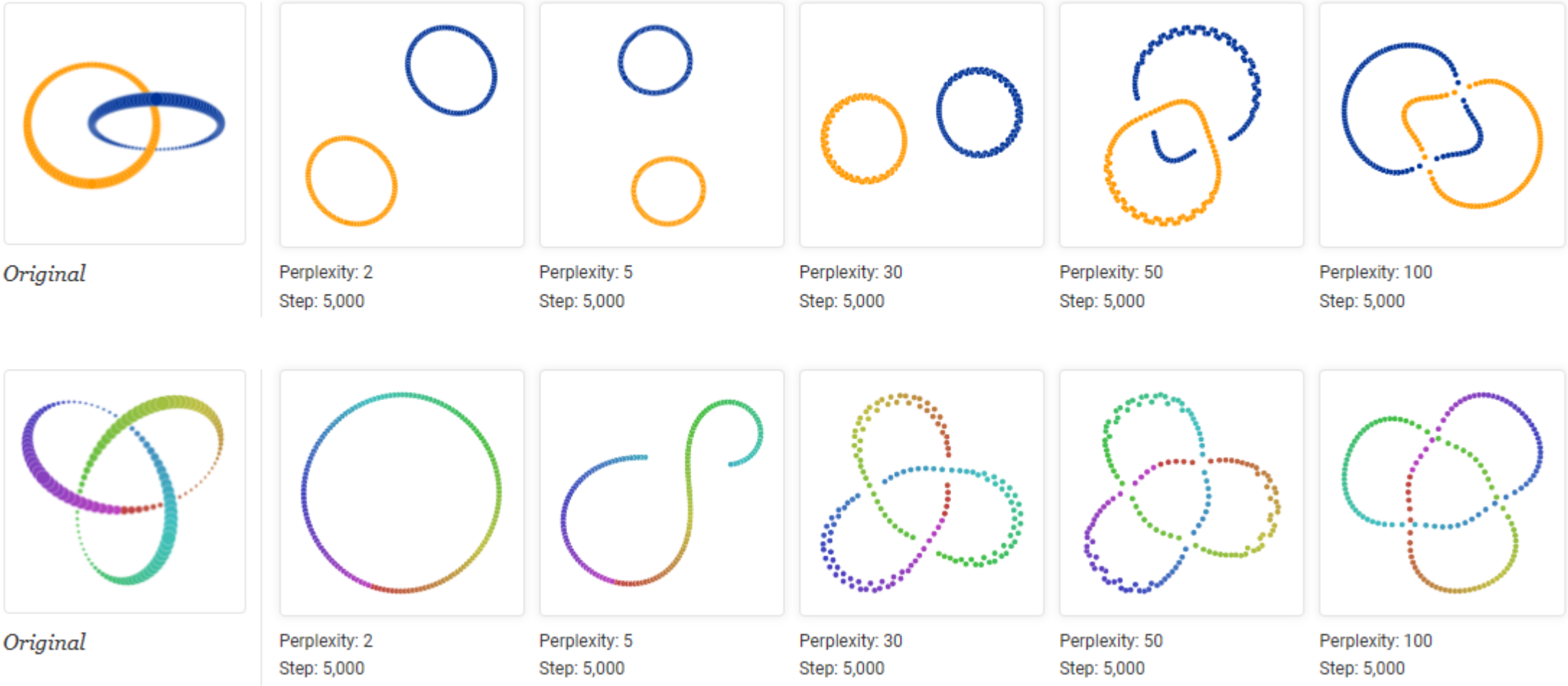
Perplexity: 30
Step: 5,000

Perplexity: 50
Step: 5,000

Perplexity: 100
Step: 5,000

Dimension Reduction

- t-SNE (t-distributed Stochastic Neighbor Embedding, t-확률적 이웃 임베딩)
  - t-SNE 적용



Dimension Reduction

- t-SNE (t-distributed Stochastic Neighbor Embedding, t-확률적 이웃 임베딩)
  - t-SNE 적용