14주차 복습세션

LDA & T-SNE

6조
권동구 김은비 조성우 조형주

2023.01.14

목차

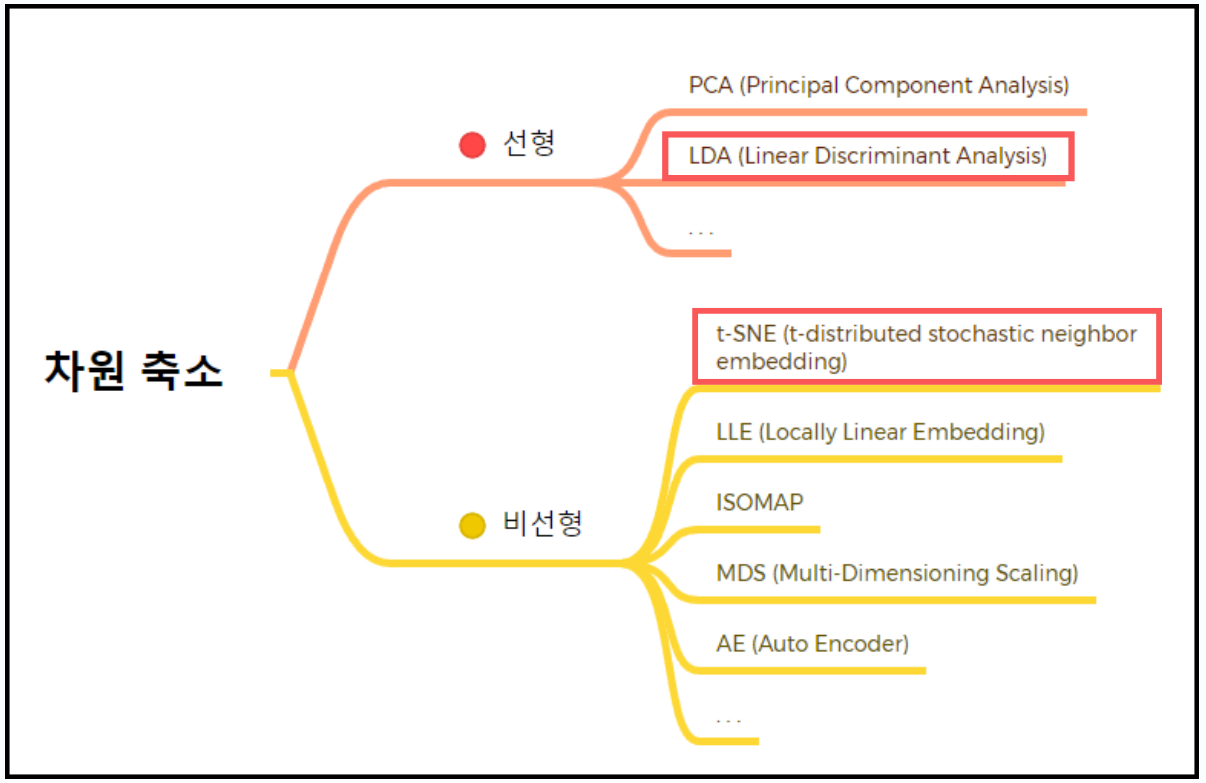
LDA
Linear Discriminatory Analysis

T-SNE
t-distributed stochastic neighbor embedding

EFA
Exploratory Factor Analysis

Manifold Learning

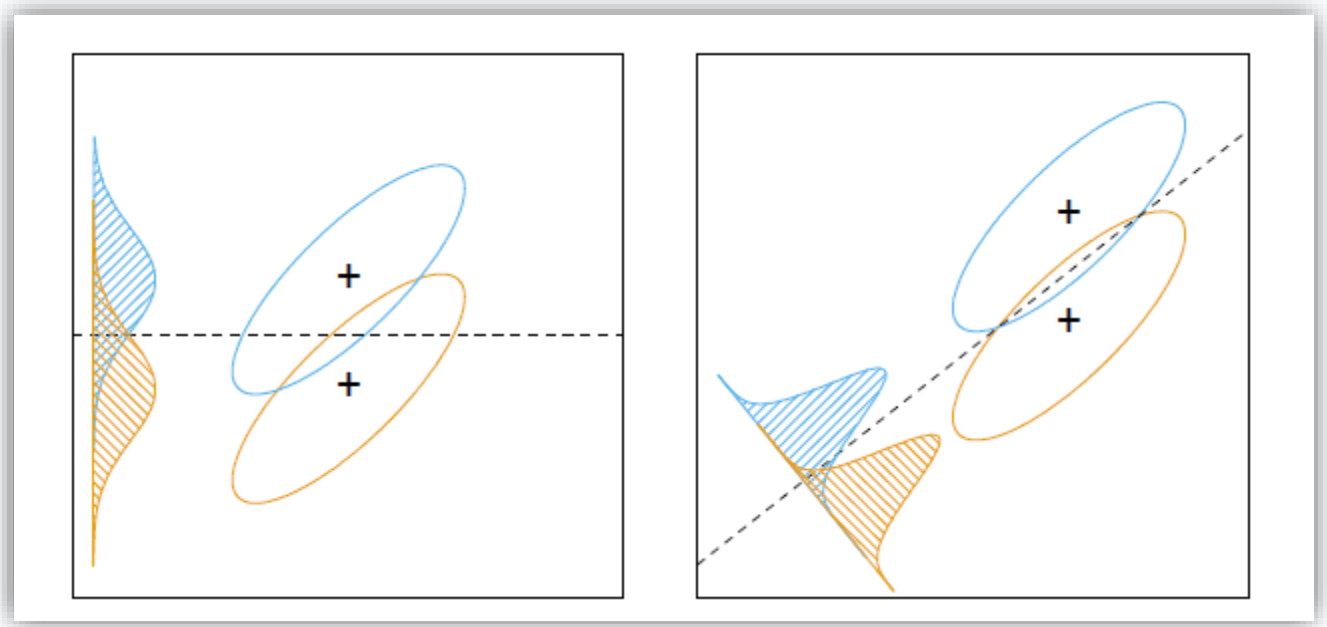
- 고차원 데이터가 있을 때 고차원 데이터를 데이터 공간에 뿌리면 샘플들을 잘 아우르는 subspace가 있을 것이라는 가정에서 학습을 진행
- 차원축소를 위해 사용하며 이를 통해 고차원 데이터를 저차원에서도 잘 표현하는 공간인 manifold를 찾아 차원축소



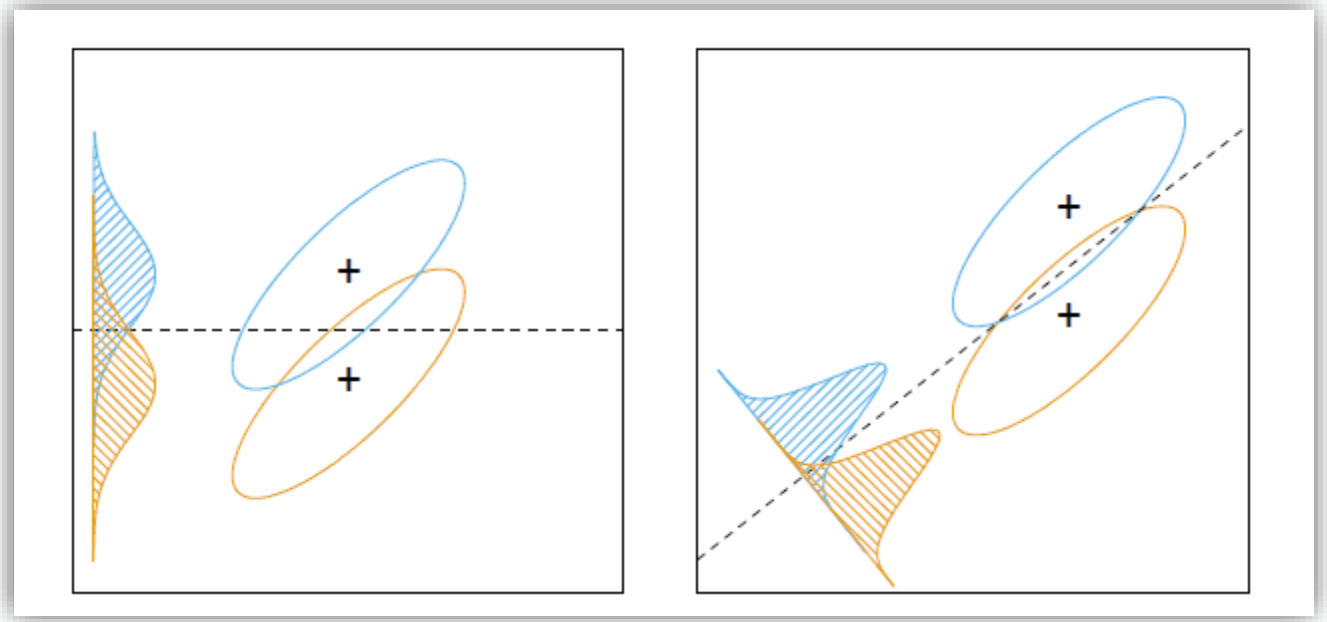
1. LDA

Linear Discriminatory Analysis

LDA란?



LDA란?



선형판별분석

데이터 분포를 학습해 **결정경계(Decision boundary)**를 만들어 데이터를 **분류(classification)**하는 모델

분산대비 평균의 차이를 최대화하는 직선을 찾는 것

LDA와 차원축소

가정

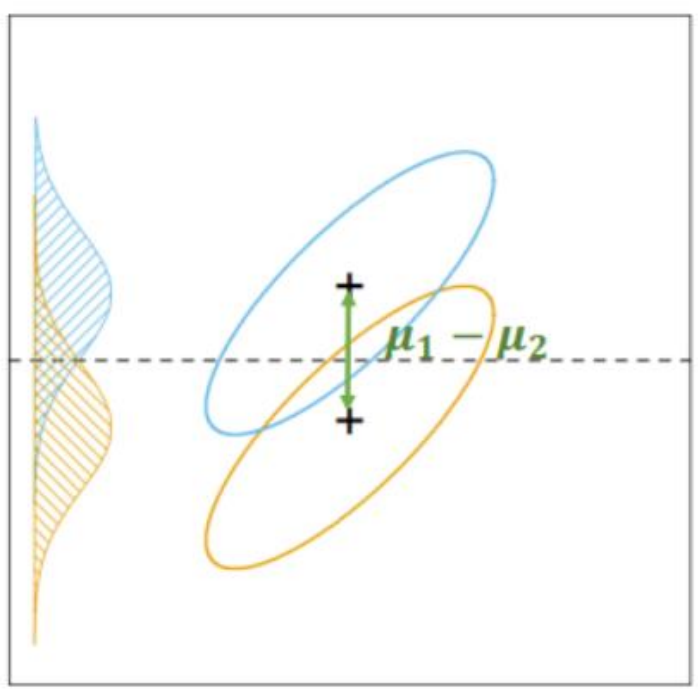
1. 각 집단(클래스)은 정규분포형태의 확률분포를 갖는다
2. 각 집단(클래스)은 비슷한 형태의 공분산 구조를 갖는다 (등분산)

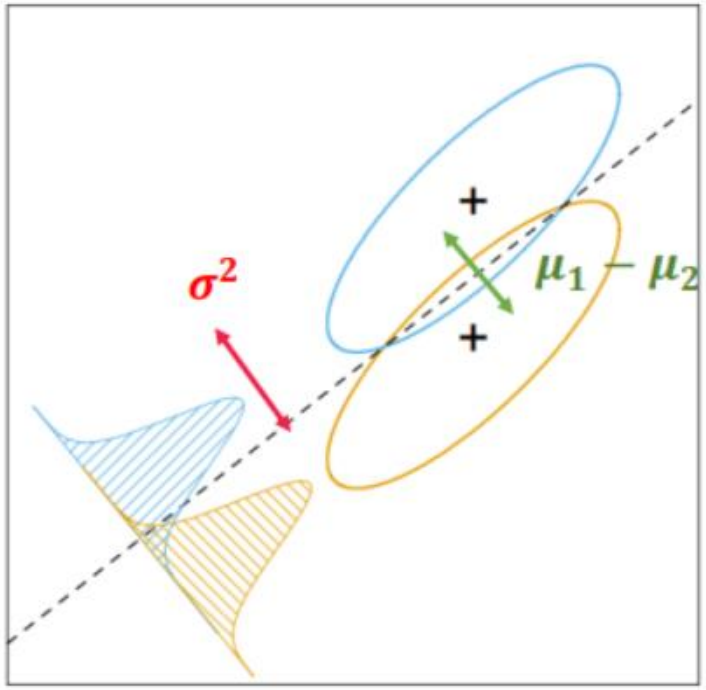
특정 공간상에서 클래스 분리를 최대화하는 축을 찾기 위해 클래스 간 분산과 클래스 내부 분산의 비율을 최대화하는 방식으로 차원을 축소

클래스 간 분산은 최대한 크게, 클래스 내부의 분산은 최대한 작게 projection

LDA와 차원축소

분산대비 평균의 차이를 최대화하는 직선을 찾는 것





평균의 차이를 최대화하는 축에 정사영을 내린 경우

LDA와 차원축소

결정경계 도출

1. 클래스 내부와 클래스 간 분산 행렬을 계산한다.
   두 행렬은 입력 데이터의 결정 값 클래스별로 개별 피처의 평균 벡터를 기반으로 구한다.

$$\sigma^2_{within} = W^T\Sigma_1 W + W^T\Sigma_2 W$$

$$\Sigma_i = \sum_{x \in g_i} W^T(x-\mu_i)(x-\mu_i)^T W$$

Class내 분산

목표 vector
=결정경계

$$\sigma^2_{between} = W^T(\mu_1-\mu_2)(\mu_1-\mu_2)^T W$$

Class간 분산

LDA와 차원축소

결정경계 도출

2. 목표는 σ^2_{within}을 최소화하고 $\sigma^2_{between}$을 최대화하는 것이기 때문에 목표 함수는 다음과 같음

$$J(W) = \frac{\sigma^2_{between}}{\sigma^2_{within}} = \frac{W^T(\sum_i^k(\mu_i - \mu)(\mu_i - \mu)^T)W}{W^T(\sum_i^k \Sigma_i)W} = \frac{W^T S_B W}{W^T S_w W}$$

3. 클래스 간 분산과 클래스 내부 분산의 비율을 최대화해야 하기 때문에 $J(W)$를 미분해서 0이 되게 하는 값을 찾는다.

$$[W^T S_w W]2S_B W - [W^T S_B W]2S_w W = 0$$

$$S_B W - \frac{W^T S_B W}{W^T S_w W} S_w W = 0$$

$$S_B W - J S_w W = 0$$

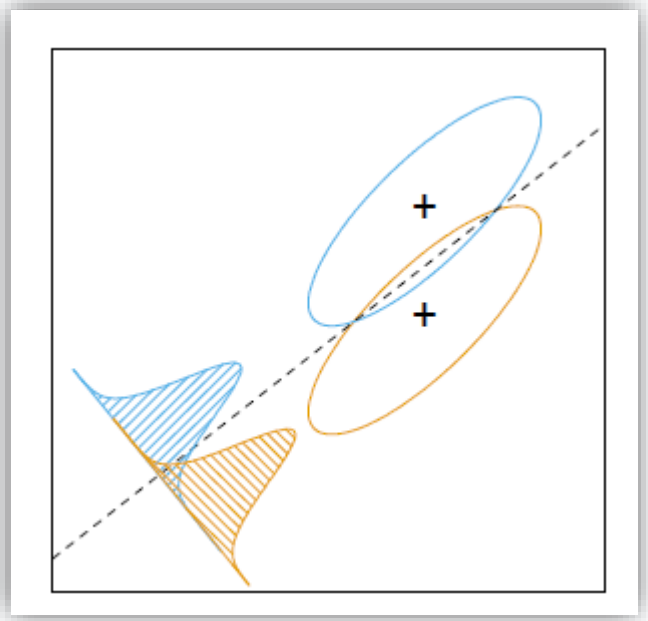
$$S_w^{-1} S_B W = JW$$

LDA와 차원축소

결정경계 도출

$$S_B W - J S_w W = 0$$

$$S_w^{-1} S_B W = J W$$



4. 우리가 찾으려는 vector W는 $S_w^{-1} S_B$의 eigenvector
   Input X와 dot product를 통해 input X를 vector W에 projection을 하면 차원축소

LDA와 차원축소

고유값 (eigenvalue), 고유벡터 (eigenvector) 정의

정방행렬 A에 대하여 다음이 성립하는 0이 아닌 벡터 x가 존재할 때

$Ax = \lambda x$ (상수 λ)

$$\begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = \lambda \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}$$

상수 **λ**를 행렬 A의 **고유값 (eigenvalue)**,
x를 이에 대응하는 **고유벡터 (eigenvector)** 라고 함

$$A \quad x = \lambda x$$

$$\begin{pmatrix} 4 & 2 \\ 3 & 5 \end{pmatrix} \begin{pmatrix} 2 \\ 3 \end{pmatrix} = 7 \begin{pmatrix} 2 \\ 3 \end{pmatrix}$$

$$\begin{pmatrix} 4 & 2 \\ 3 & 5 \end{pmatrix} \begin{pmatrix} 2 \\ 3 \end{pmatrix} = \begin{pmatrix} 4\times 2 + 2\times 3 \\ 3\times 2 + 5\times 3 \end{pmatrix} = \begin{pmatrix} 14 \\ 21 \end{pmatrix} = 7 \begin{pmatrix} 2 \\ 3 \end{pmatrix}$$

square matrix

$$Ax = \lambda x$$

eigenvalue

eigenvector

$$A \quad x = \lambda x$$

$$\begin{pmatrix} 4 & 2 \\ 3 & 5 \end{pmatrix} \begin{pmatrix} -1 \\ 1 \end{pmatrix} = 2 \begin{pmatrix} -1 \\ 1 \end{pmatrix}$$

$$\begin{pmatrix} 4 & 2 \\ 3 & 5 \end{pmatrix} \begin{pmatrix} -1 \\ 1 \end{pmatrix} = \begin{pmatrix} 4\times(-1) + 2\times 1 \\ 3\times(-1) + 5\times 1 \end{pmatrix} = \begin{pmatrix} -2 \\ 2 \end{pmatrix} = 2 \begin{pmatrix} -1 \\ 1 \end{pmatrix}$$

LDA와 차원축소

베이즈 이론

$$P(Y=k|X=x) = \frac{\pi_k f_k(x)}{\sum_{l=1}^{K} \pi_l f_l(x)}$$

사후확률

$\pi_k f_k(x)$

$\pi_k = P(Y_k)$ 사전확률

전체 데이터에서 몇개가 k클래스 라벨을 가지고 있는지

$$f_k(x) = P(X=x|Y=k)$$

$Y=k$ 클래스일 때 특정 x가 나타나는 확률

주로 정규분포임을 가정하고 분석

$$= \frac{1}{\sqrt{2\pi}\sigma_k} \exp\left(-\frac{1}{2\sigma_k^2}(x-\mu_k)^2\right)$$

LDA와 차원축소

분류하는 방법

$$P(Y=k|X=x)=\frac{\pi_k f_k(x)}{\sum_{l=1}^{K}\pi_l f_l(x)}$$

1) p=1일 때

$$\delta_k(x)=x\cdot\frac{\mu_k}{\sigma^2}-\frac{\mu_k^2}{2\sigma^2}+\log(\pi_k)$$

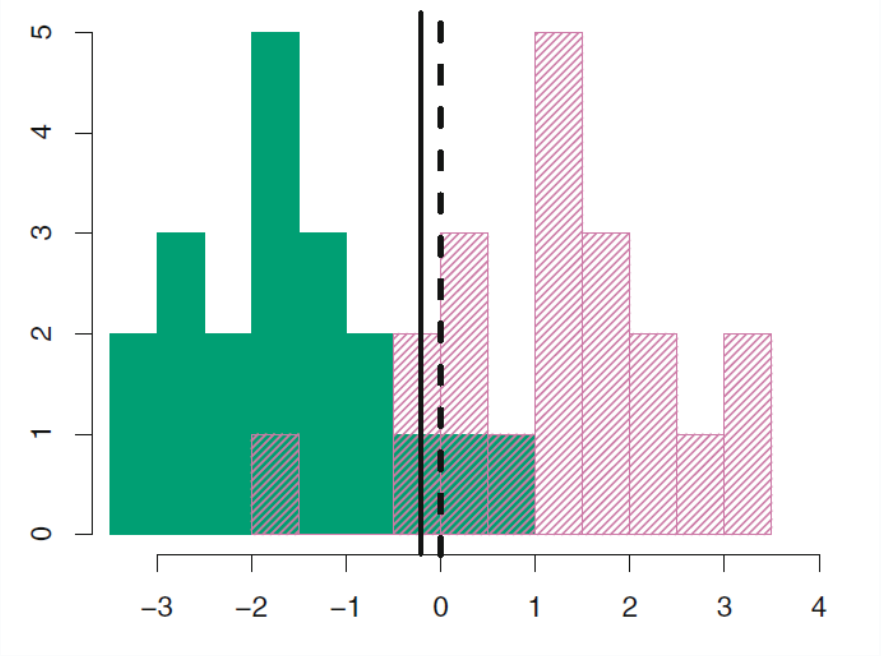
$$\hat{\mu}_k=\frac{1}{n_k}\sum_{i:y_i=k}x_i \qquad \hat{\pi}_k=\frac{n_k}{n}$$

$$\hat{\sigma}^2=\frac{1}{n-K}\sum_{k=1}^{K}\sum_{i:y_i=k}(x_i-\hat{\mu}_k)^2$$

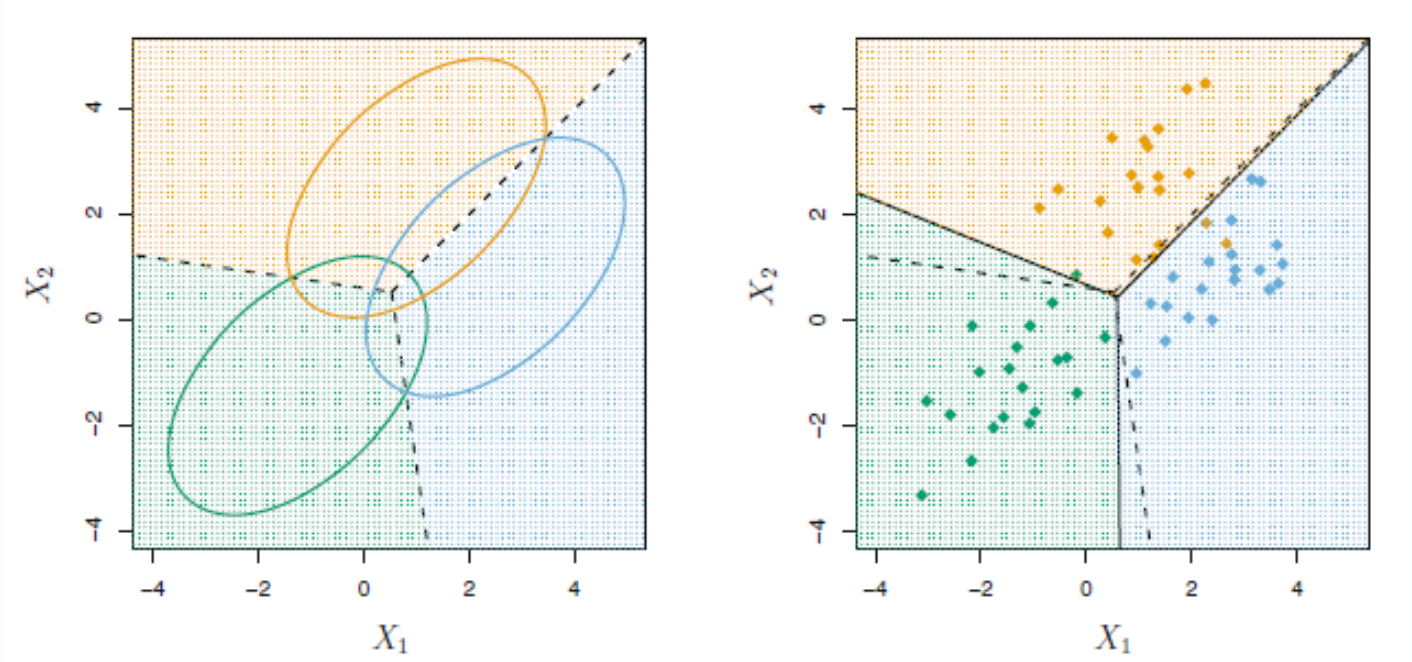
2) p>1일 때

$$\delta_k(x)=x^T\Sigma^{-1}\mu_k-\frac{1}{2}\mu_k^T\Sigma^{-1}\mu_k+\log\pi_k$$

LDA와 차원축소



p=1, k=2



p=2, k=3

LDA와 차원축소

4차원 → 2차원

LDA와 PCA

	PCA	LDA
공통점	입력 데이터 세트를 저차원 공간에 투영해 차원을 축소하는 기법	
차이점	비지도학습	지도학습
	변동성이 가장 큰 축	결정 값 클래스를 최대한 분리할 수 있는 축
	공분산 행렬을 사용	클래스 간 분산, 클래스 내 분산 행렬 사용

LDA는 분류에서 사용하기 쉽도록 개별 클래스를 분별할 수 있는 기준을 최대한 유지하면서 차원 축소

QDA란?

→ Quadratic Discriminant Analysis (비선형)

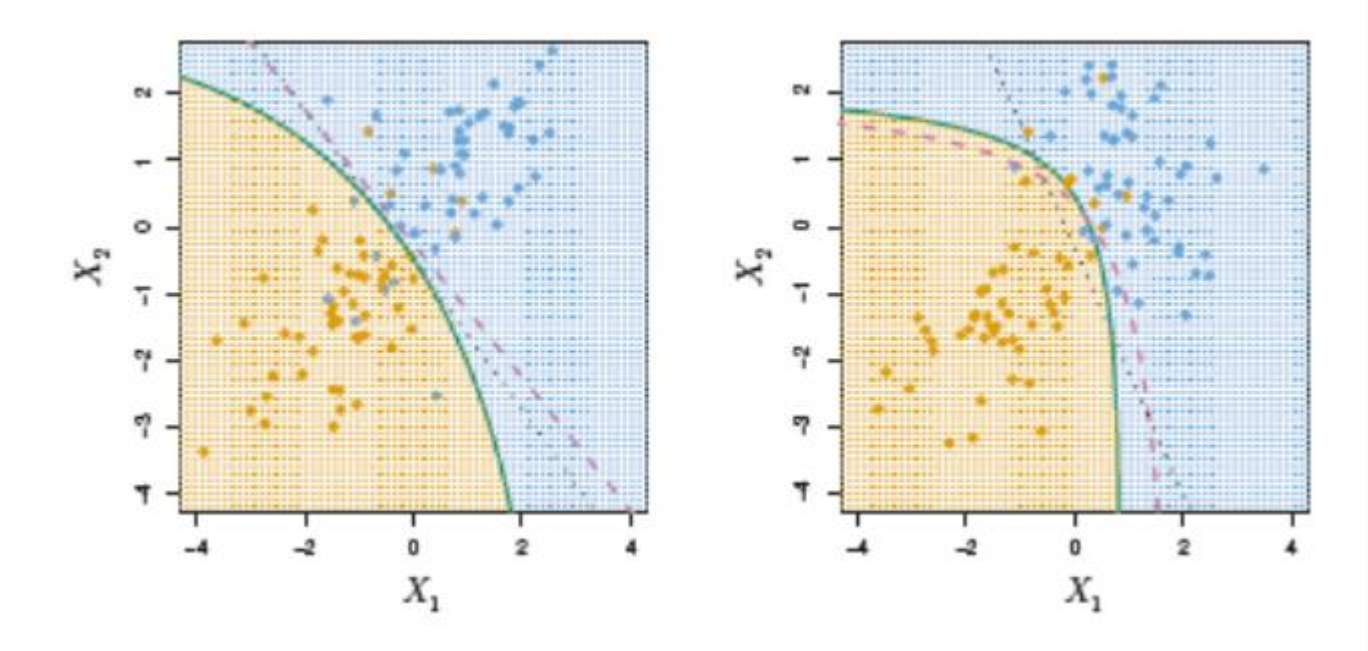
가정

1. 각 집단(클래스)은 정규분포형태의 확률분포를 갖는다
2. ~~각 집단(클래스)은 비슷한 형태의 공분산 구조를 갖는다~~ → 각 집단(클래스)은 다른 형태의 공분산 구조를 갖는다

$$X \sim N(\boldsymbol{\mu_k}, \sum_{\boldsymbol{k}})$$

$$\begin{aligned} \delta_k(x) &= -\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k) - \frac{1}{2}\log|\Sigma_k| + \log\pi_k \\ &= -\frac{1}{2}x^T\Sigma_k^{-1}x + x^T\Sigma_k^{-1}\mu_k - \frac{1}{2}\mu_k^T\Sigma_k^{-1}\mu_k - \frac{1}{2}\log|\Sigma_k| + \log\pi_k \end{aligned}$$

QDA란?

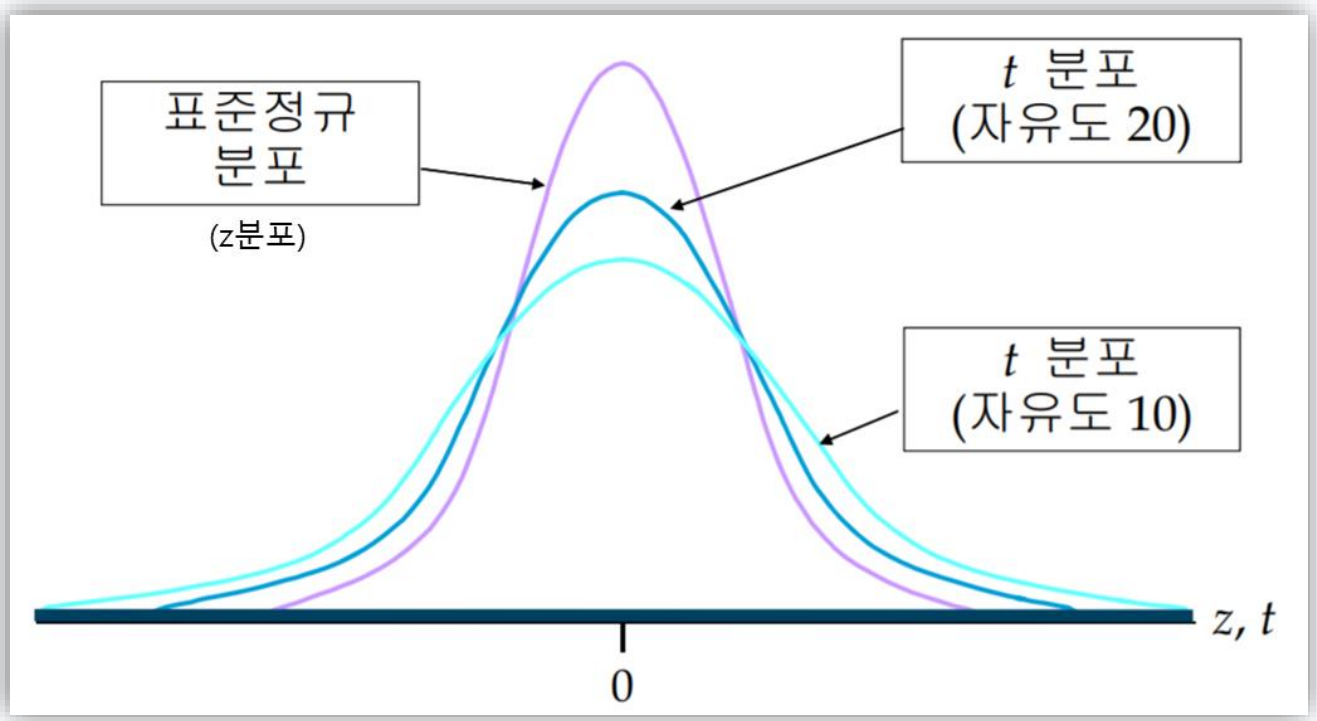


훈련 데이터 수가 적어서 variance를 줄이는 것이 중요할 경우 **LDA를**,
데이터 수가 많아서 variance(데이터 셋이 달라지는 것에 따른 모델의 변동성)에 대한 우려가 적을 때, 혹은 공분산에 대한 가정이 비현실적으로 판단될 때에는 **QDA를** 사용

2. T-SNE

t-distributed stochastic neighbor embedding

t-분포



- 표준정규분포보다 분산이 크기 때문에 양쪽 꼬리가 두터운 형태
- 평균, 분산에 따라 형태가 결정되는 표준정규분포와 달리 자유도에 따라 다른 모양
  - 자유도가 커질수록 표준정규분포와 유사해짐
- 소 표본 (n 〈 30)일 때 모 평균 추정하고 모 표준편차를 모를 때 주로 사용

T-SNE란?

T-SNE t-distributed stochastic neighbor embedding

고차원 데이터를 저차원 데이터로 변환하는 차원 축소 (dimensionality reduction) 기법

목적

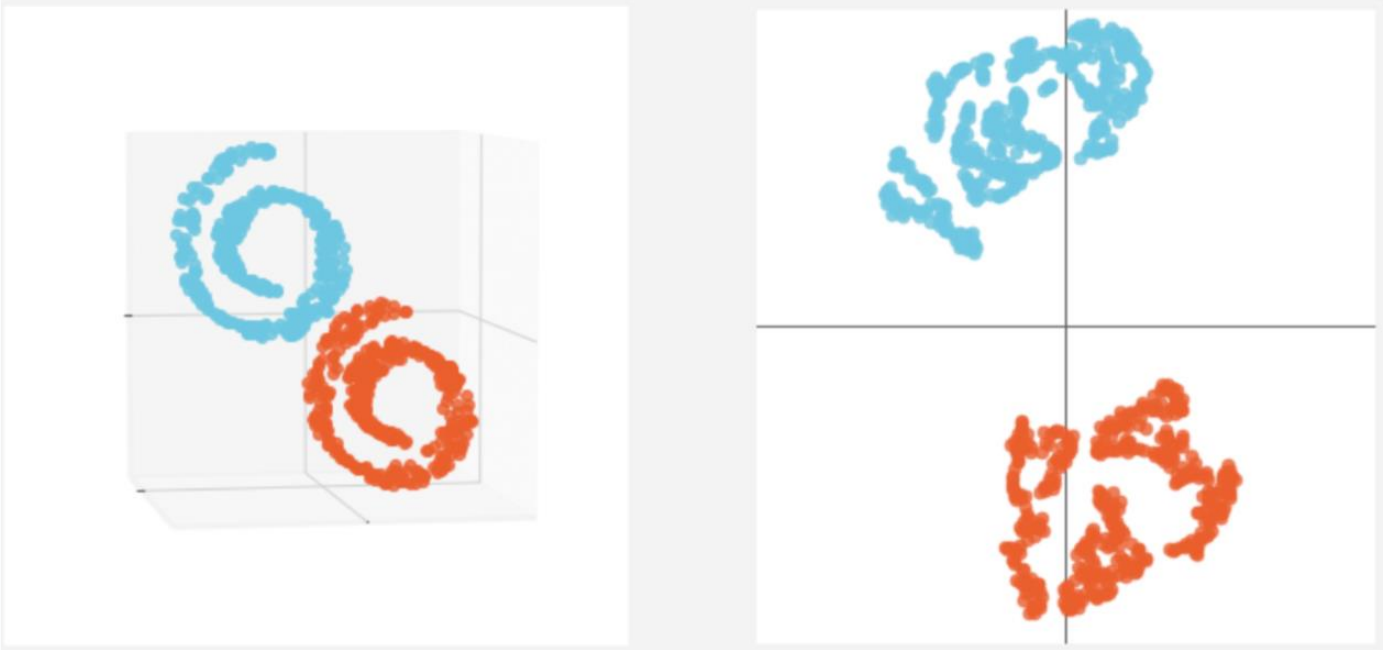
낮은 차원 공간의 시각화에 주로 사용 (매니폴드 학습)

비슷한 구조끼리 데이터를 정리한 상태이므로 데이터 구조를 이해하는 데 용이

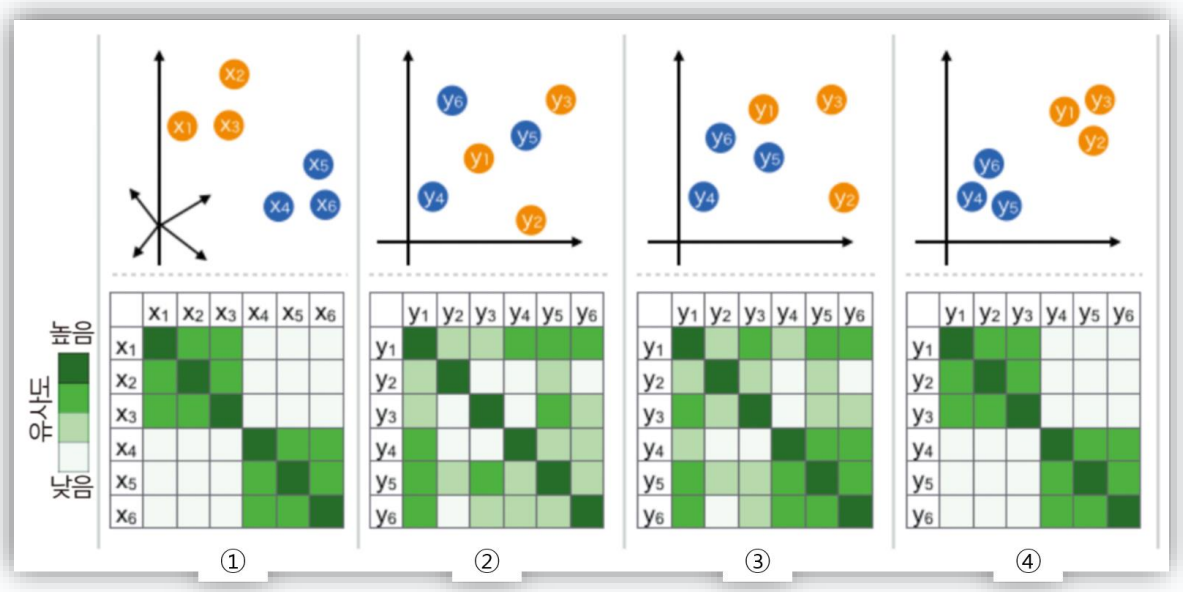
☞ 높은 차원 공간에서 비슷한 데이터 구조는 낮은 차원 공간에서 가깝게 대응하며,
비슷하지 않은 데이터 구조는 멀리 떨어져 대응

T-SNE란?

T-SNE t-distributed stochastic neighbor embedding



T-SNE 알고리즘



①의 x_i는 기존 데이터로 고차원에 분포되어 있음
y_i는 t-SNE를 통하여 저차원으로 매핑된 데이터
기존 데이터는 3차원이고 저차원은 2차원

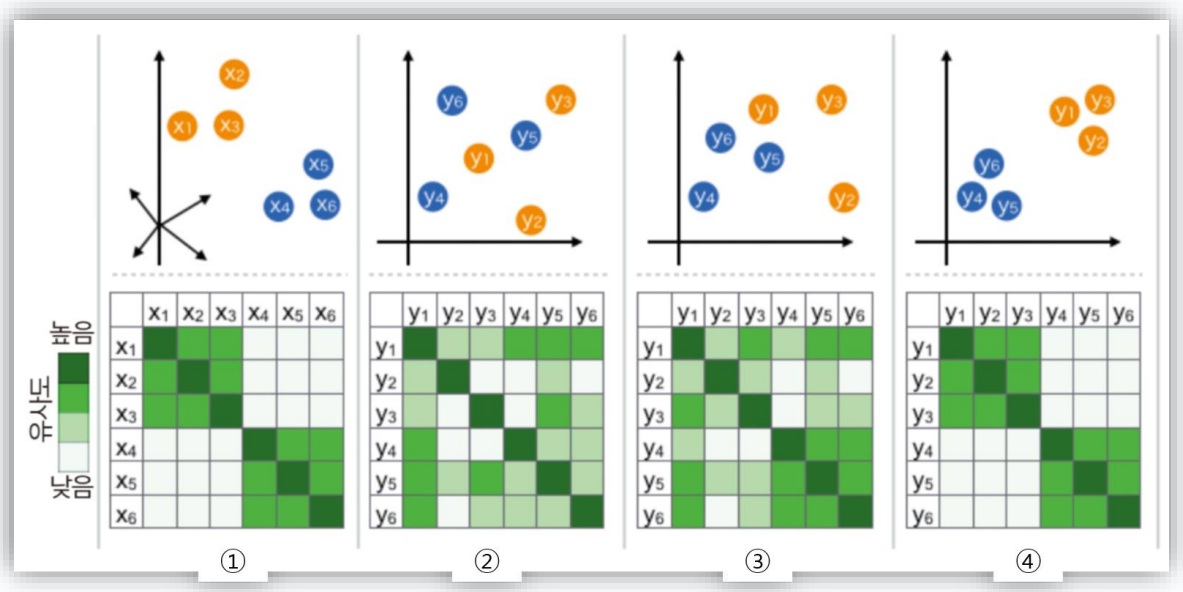
① 모든 i, j쌍에 대하여 x_i, x_j의 유사도를 가우시안 분포를 이용하여 나타낸다.

② x_i와 같은 개수의 점 y_i를 낮은 차원 공간에 무작위로 배치하고, 모든 i, j쌍에 관하여 y_i, y_j의 유사도를 t-sne를 이용하여 나타낸다.

③ 앞의 ①, ②에서 정의한 유사도 분포가 가능하면 같아지도록 데이터 포인트 y_i를 갱신한다.

④ 수렴 조건까지 과정 ③을 반복한다.

T-SNE 알고리즘



①의 x_i는 기존 데이터로 고차원에 분포되어 있음
y_i는 t-SNE를 통하여 저차원으로 매핑된 데이터
기존 데이터는 3차원이고 저차원은 2차원

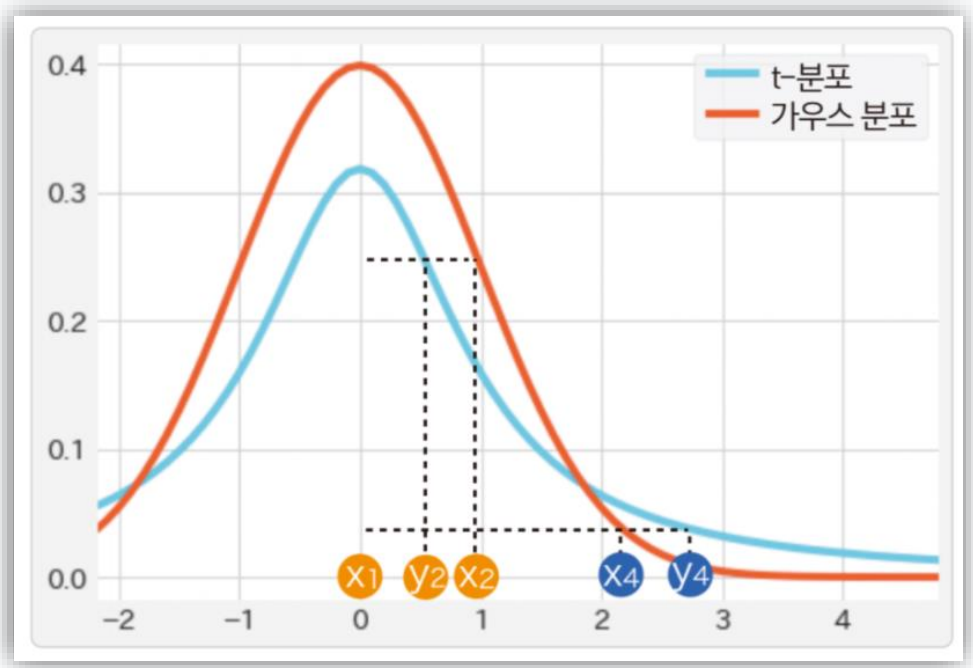
① 모든 i, j쌍에 대하여 x_i, x_j의 유사도를 가우시안 분포를 이용하여 나타낸다.

② x_i와 같은 개수의 점 y_i를 낮은 차원 공간에 무작위로 배치하고, 모든 i, j쌍에 관하여 y_i, y_j의 유사도를 t-sne를 이용하여 나타낸다.

↓

①, ②의 유사도는 데이터 포인트들의 비슷한 정도를 의미
단순히 데이터 사이의 거리가 아니라 확률 분포를 이용

T-SNE 알고리즘



가로축은 거리, 세로축은 유사도
데이터 사이의 거리가 가까울수록 유사도가 크고,
멀수록 유사도가 작아진다

높은 차원 데이터들(x_i)은 정규 분포로 유사도 측정

☞ p_{ij}는 데이터 포인트 x_i, x_j의 유사도를 나타냄

$$p_{j|i} = \frac{\exp(-\frac{\|x_i - x_j\|^2}{2\sigma_i^2})}{\sum_{k \neq i} \exp(-\frac{\|x_i - x_k\|^2}{2\sigma_i^2})}$$

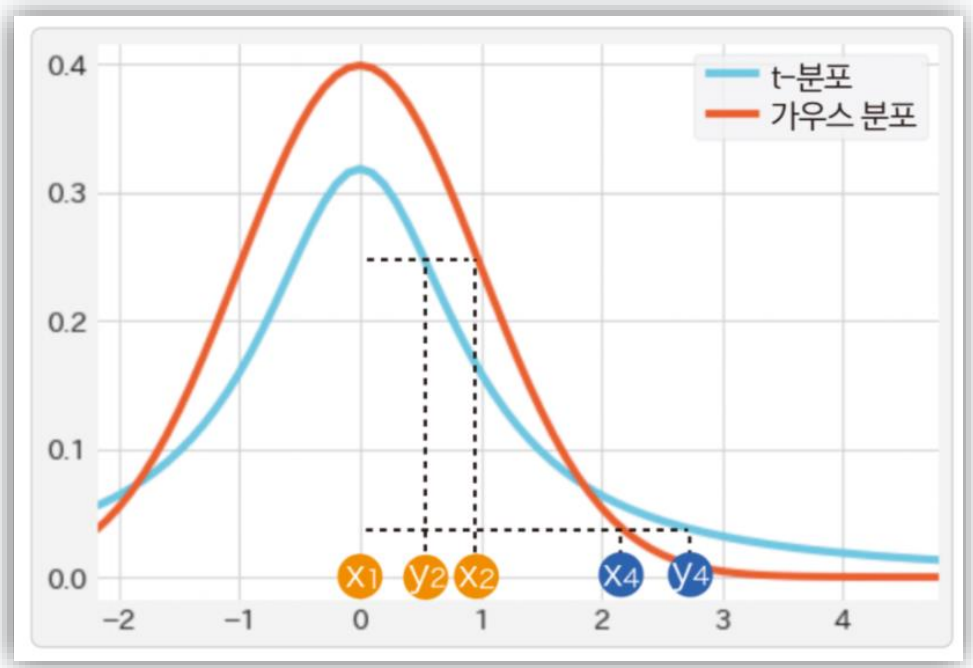
$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2n} \quad (n = \#of\ datapoints)$$

x_i에 대응하는 데이터 포인트 y_i를 낮은 차원 공간에 무작위로 배치

☞ y_i는 t-분포로 유사도를 나타내는 q_{ij} 구하기

$$q_{ij} = \frac{\mathrm{f}(\|x_i - x_j\|)}{\sum_{k \neq i} \mathrm{f}(\|x_i - x_k\|)} \qquad f(z) = \frac{1}{1 + z^2}$$

T-SNE 알고리즘



③ 앞의 ①, ②에서 정의한 유사도 분포가 가능하면 같아지도록 데이터 포인트 y_i를 갱신한다.

= q_{ij}를 p_{ij}와 같은 분포가 되도록 데이터 포인트 y_i를 갱신

[직관적]

☞ 높은 차원 공간에서 x_i 유사도 각각의 관계를 낮은 차원 공간의 y_i에서 재현

유사도가 큰 상태의 관계를 재현할 때

≫ 낮은 차원 공간에서 데이터 포인트를 더 가까이 배치

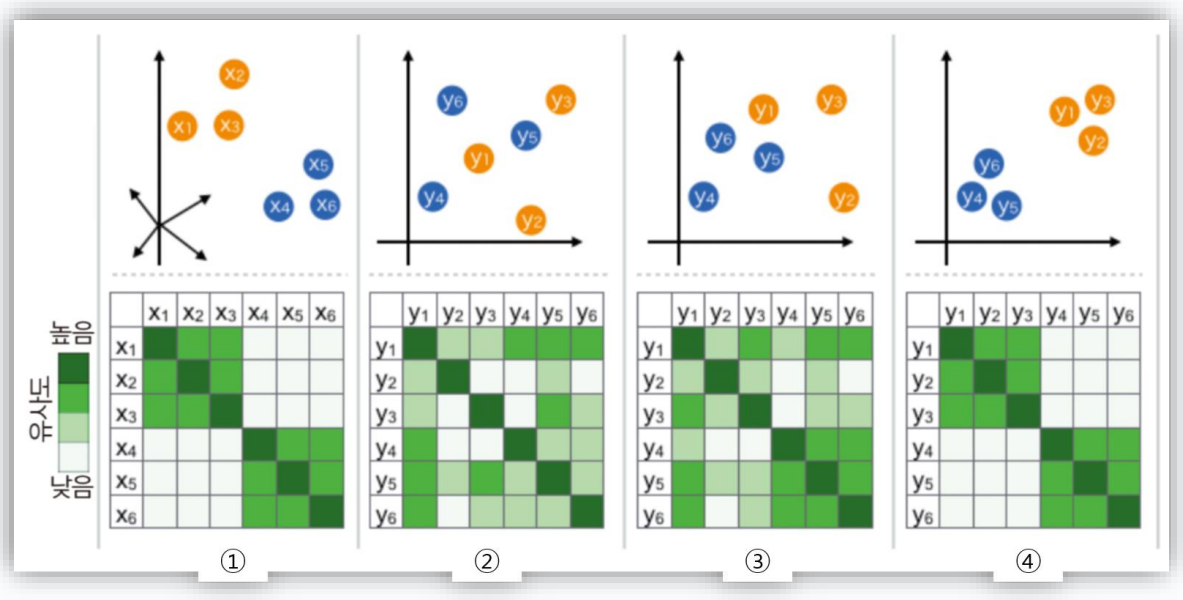
유사도가 작은 상태의 관계를 재현할 때

≫ 낮은 차원 공간에서 데이터 포인트를 더 멀리 배치

왜 t분포인가?

- 일반적인 정규분포보다 끝단의 값이 두터운 분포
- 정규 분포는 꼬리가 두텁지 않아서 i번째 개체에서 적당히 떨어져 있는 이웃 j와 아주 많이 떨어져 있는 이웃 k가 선택될 확률이 크게 차이가 나지 않음
- 구분을 좀 더 잘하기 위해 낮은 차원에서는 꼬리가 두터운 t분포를 씀

T-SNE 알고리즘



[수학적]

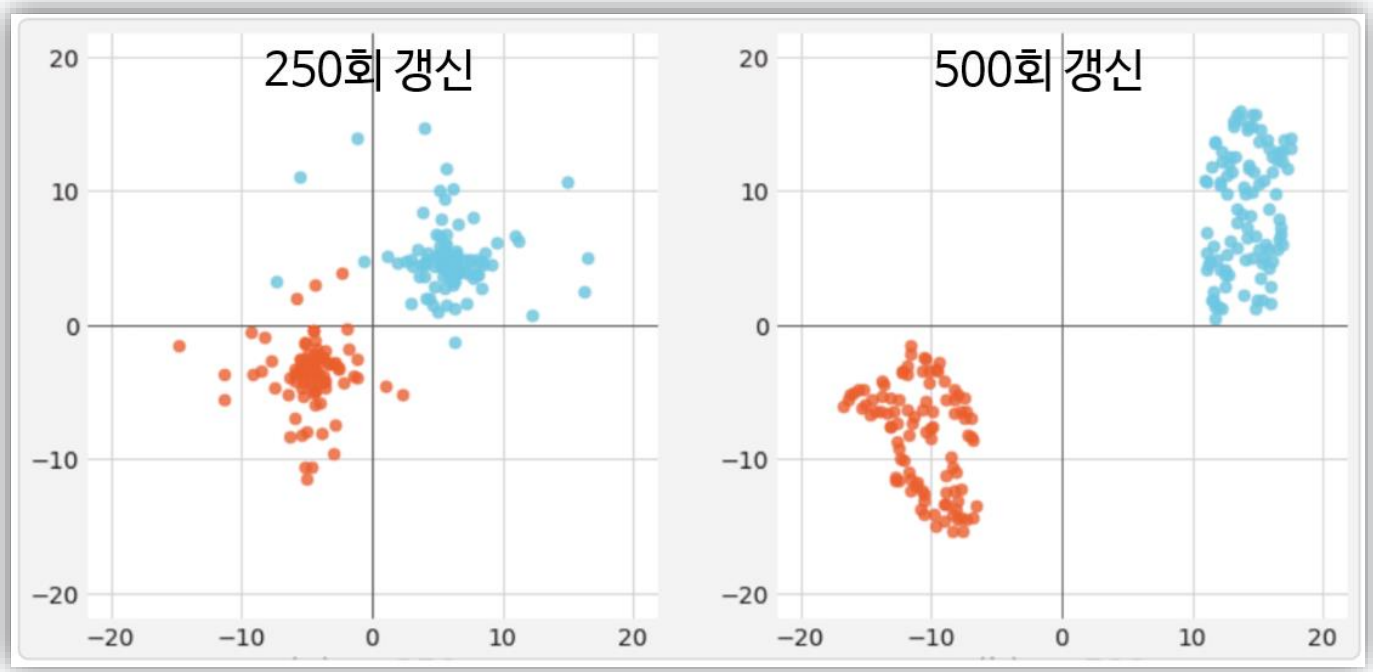
☞ 높은 차원에서의 유사도(p_{ij})와 낮은 차원에서 유사도(q_{ij})의 KL-divergence를 최소화

*KL-divergence = Kullback-Leiblur divergence
=한 확률 분포가 두 번째 예상 확률 분포와 어떻게 다른지 측정하는 척도

$$KL(P||Q) = \sum_{i,j} p_{ij} \log(\frac{p_{ij}}{q_{ij}})$$

t-SNE는 거리 KL을 최소화 시킴으로써 작동
KL 발산 값은 P와 Q가 유사할수록 0이 됨

T-SNE 알고리즘



갱신 횟수가 늘수록 데이터 포인트의 차이가 명확하게 나타남

T-SNE의 특징

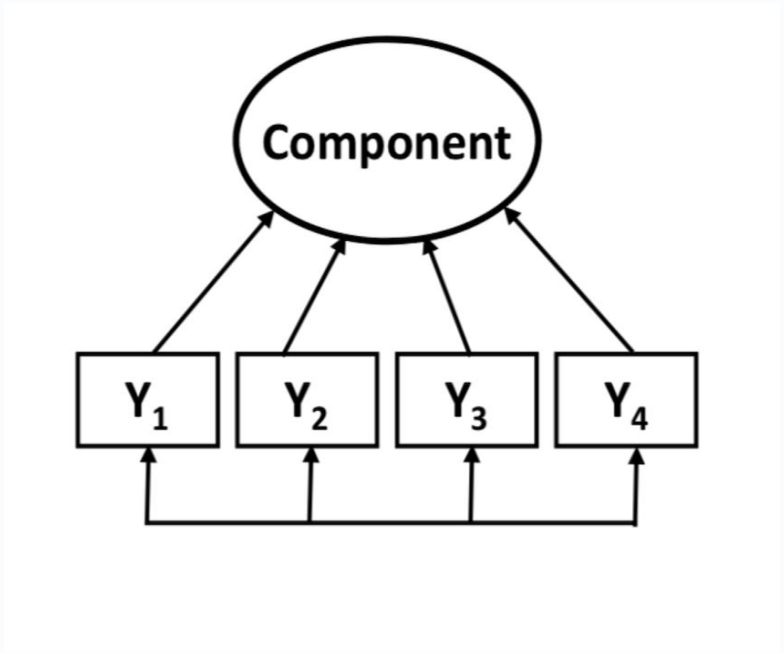
- 비선형적인 방법의 차원 축소 방법
- 고차원 공간에서의 점들의 유사성과 그에 해당하는 저차원 공간에서의 점들의 유사성을 계산
- 점들의 유사도는 A를 중심으로 한 정규 분포에서 확률 밀도에 비례하여 이웃을 선택하면 포인트 A가 포인트 B를 이웃으로 선택한다는 조건부 확률(=유사도)로 계산됨
- 본래 공간(고차원)에서 거리가 가까운 포인트들은 보통 줄어든 차원에서도 함께 있으며 본래 공간에서 떨어져 있는 포인트들은 줄어든 공간에서 떨어지기도, 같이 있기도 함
- 이러한 방식으로, t-SNE는 다차원 데이터를 보다 낮은 차원 공간으로 매핑하고, 다수의 특징을 갖는 데이터 포인트의 유사성을 기반으로 점들의 클러스터를 식별함으로써 데이터에서 패턴을 발견
- 하지만 t-SNE 과정이 끝나면 input feature를 확인하기가 어렵고 t-SNE 결과만 가지고 무언가를 추론 하기는 어려워 주로 시각화 툴로 사용

3. EFA

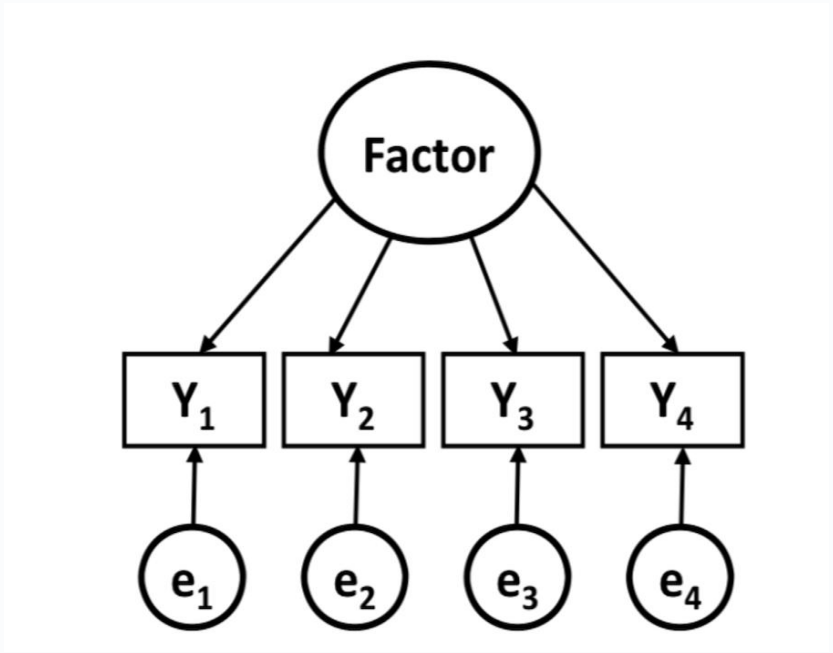
Exploratory Factor Analysis

EFA & PCA

PCA



EFA



여러 개의 변수로부터 하나의 요인/성분을 얻는 것

차원 축소 기법 | 차원 축소 기법 X

Factor Analysis

요인분석

여러 변수들 사이의 공분산 구조를 밝히는 것
= 여러 변수들 사이의 어떤 관계가 있어 특정 변수가 다른 변수와 함께 변함
☞ 그들을 공통적으로 설명하는 숨겨진 요인 (factor)가 있기 때문이라고 가정

EFA **탐색적 요인분석** Exploratory Factor Analysis

어떤 이론이나 가설 없이 응답 데이터를 분석해 설문 문항들이 몇 개의 요인으로 구성되는지 살펴봄
☞ 요인의 수를 직접 결정

CFA **확인적 요인분석** Confirmatory Factor Analysis

수집한 데이터가 이미 가설로 설정된 모델 구조에 적합한지 검증

EFA란?

❖ data

```
> head(cereal.data, 4)
  Filling Natural Fiber Sweet Easy Salt Satisfying Energy Fun Kids Soggy
1       5       5     5     1    2    1          5      4   1    4     5
2       1       2     2     1    5    2          5      1   1    5     3
3       5       4     5     5    5    3          5      5   5    5     3
4       5       5     5     3    5    2          5      5   5    5     3
  Economical Health Family Calories Plain Crisp Regular Sugar Fruit Process
1          5      5      5        1     3     1       4     1     1       3
2          5      2      5        1     5     5       1     2     1       5
3          3      5      5        1     1     5       4     3     1       2
4          3      5      5        1     1     1       4     2     5       2
  Quality Treat Boring Nutritious
1       5     1      1          5
2       2     1      1          3
3       5     4      1          5
4       5     5      1          5
```

```
> fa(cereal_answer, nfactors = 5, fm = 'ml', rotate='varimax')
```

Factor2 = Sweet, Salt, Calories, Sugar = 당도 요인
Factor3 = Kids, Economical, Family = 가족 적합성
Factor4 = Fun, Crisp, Fruit, Treat = 식감 요인

❖ EFA factor loadings

```
> loadings(fa_result)

Loadings:
            ML1    ML2    ML3    ML4    ML5
Filling      0.647         0.190  0.144  0.487
Natural      0.731 -0.215                0.153
Fibre        0.816
Sweet               0.696         0.351  0.166
Easy         0.230         0.307
Salt                0.689
Satisfying   0.570         0.387  0.199  0.333
Energy       0.611         0.168  0.225  0.339
Fun          0.125  0.155  0.377  0.538
Kids                       0.867
Soggy                      0.130 -0.454
Economical         -0.258  0.409 -0.197 -0.110
Health       0.840 -0.271
Family                     0.794  0.122
Calories    -0.155  0.592         0.122  0.179
Plain       -0.115                -0.638 -0.150
Crisp               0.157  0.335  0.459
Regular      0.657
Sugar       -0.177  0.852         0.170
Fruit        0.341  0.161 -0.284  0.439  0.152
Process     -0.214  0.387        -0.101 -0.184
Quality      0.681 -0.222  0.200  0.218 -0.102
Treat        0.234  0.216  0.299  0.650
Boring      -0.150         -0.198 -0.508
Nutritious   0.849 -0.154
```

감사합니다