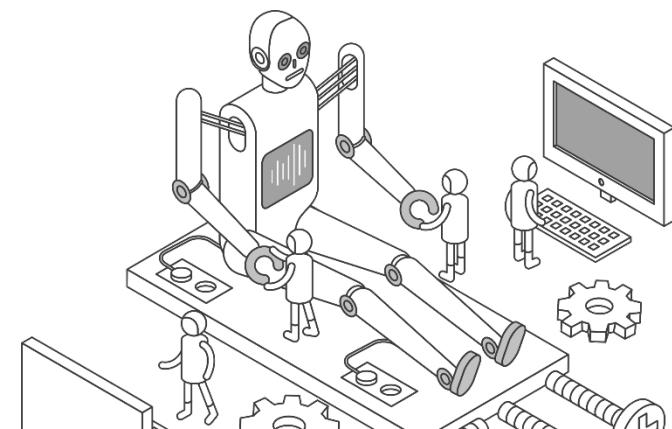


# 2022 디지털 전환을 위한 AI 전문가 과정

ARTIFICIAL INTELLIGENCE  
BIG DATA  
SMART FACTORY

AI·빅데이터 심화과정

## Unsupervised Learning



## K-Means

- Clustering은 unsupervised learning으로 target value가 없는 dataset이 주어집니다.
- K-means는 machine learning의 가장 기초적인 clustering 알고리즘입니다.
- K개의 중심(centroid)을 찾고, 데이터들은 가장 가까운 중심에 해당하는 cluster에 대응시킵니다.
- K개의 centroid들은 각 cluster들의 평균 지점(mean)에 해당하기 때문에 K-means라는 이름이 붙었습니다.
- Dataset  $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$ 에 대해  $K$ 개로 clustering한다고 합시다. 각 cluster의 mean이  $\mu_k$ 이고 data point  $\mathbf{x}_n$ 이  $k$ 번째 cluster에 속함을 indicator variable  $r_{nk} \in \{0,1\}$ 로 표현하여 다음과 같이 objective function을 쓸 수 있습니다.
- 

$$J = \sum_{n=1}^N \sum_{k=1}^K r_{nk} \ \| \mathbf{x}_n - \boldsymbol{\mu}_k \ |^2$$

# K-Means

- Objective function  $J$ 를 최소화하기 위한  $r_{nk}$ 는 각 data point index  $n$ 에 대해  $\| \mathbf{x}_n - \boldsymbol{\mu}_k \| ^2$  항이 최소가 되는 cluster index  $k$ 를 대응시키는 방법으로 구할 수 있습니다.

•

$$r_{nk} = \begin{cases} 1 & \text{if } k = \arg \min_j \| \mathbf{x}_n - \boldsymbol{\mu}_j \| ^2 \\ 0 & \text{otherwise} \end{cases}$$

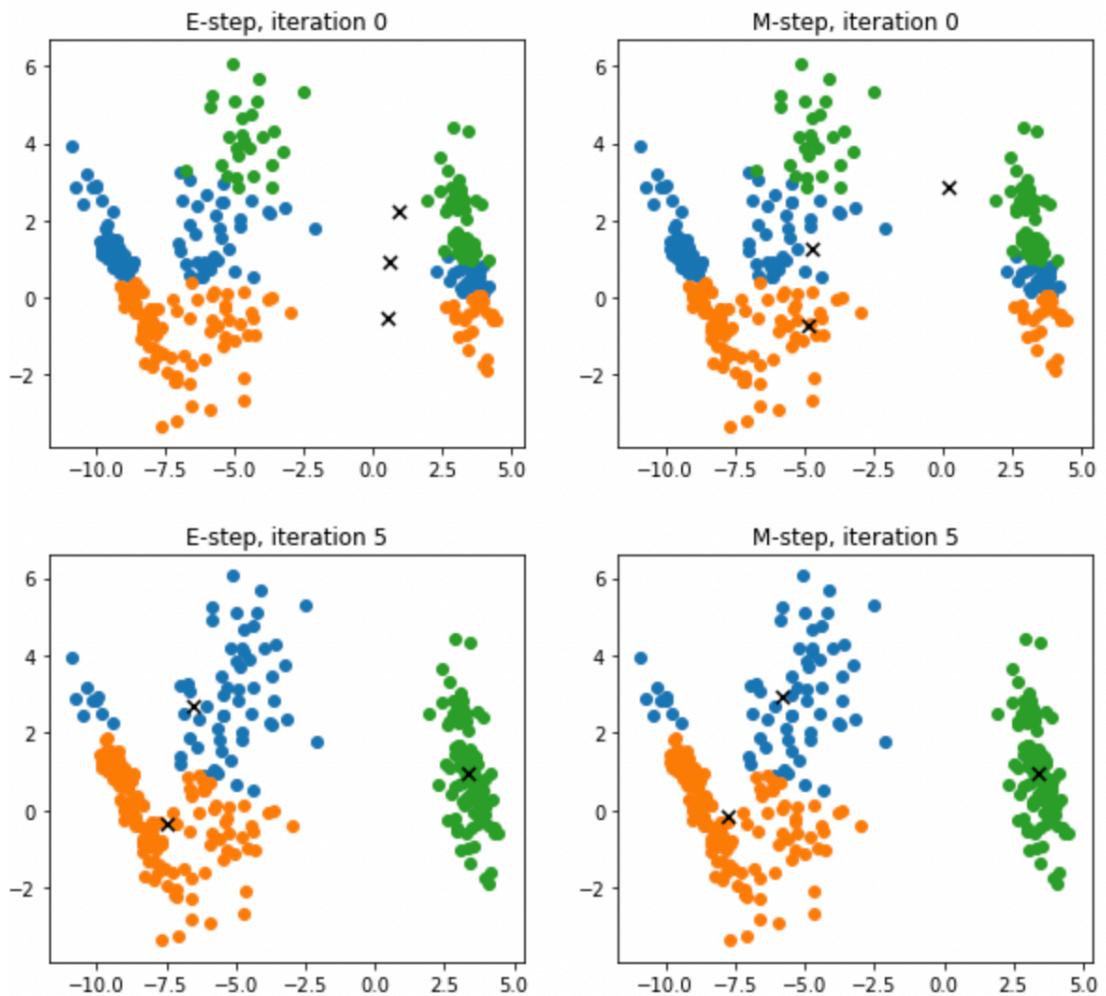
- data point  $\mathbf{x}_n$ 에 대응하는 cluster를 추정한다, 예상한다라는 맥락에서 위 과정을 expectation step이라 부릅니다.
- Objective function  $J$ 를 최소화하는  $\boldsymbol{\mu}_k$ 는  $J$ 를 미분한 식을 0으로 두고  $\boldsymbol{\mu}_k$ 에 대해 식을 전개하여 얻을 수 있습니다.

•

$$\boldsymbol{\mu}_k = \frac{\sum_n r_{nk} \mathbf{x}_n}{\sum_n r_{nk}}$$

- Expectation step에서 추정한  $r_{nk}$ 값들을 근거로  $J$ 를 최대화하기 위한 parameter  $\boldsymbol{\mu}_k$ 를 구한다는 맥락에서 위 과정을 maximization step이라고 부릅니다.
- Expectation과 maximization step을 번갈아 가면서 작동시키는 것을 EM 알고리즘이라 부릅니다.

# K-Means



- GMM은 K-means와 달리 확률적으로 clustering합니다.
- 어떤 데이터가 주어졌을 때, 특정 cluster에 속할지를 확률적으로 표현합니다.
- 예를 들어 데이터  $x=(1.0, -0.5)$ 라는 값이 주어졌을 때, cluster 1에 속할 확률 70%, cluster 2에 속할 확률 20%, cluster 3에 속할 확률 10%과 같은 식으로 표현합니다.
- 이를 위해 Gaussian 분포를 조합하여 사용합니다.
- 하나의 Gaussian 분포는 평균 지점을 뜻하는 mean 값  $\mu$ 와, covariance matrix를 뜻하는  $\Sigma$ 의 파라메터를 갖습니다.
- 총  $K$ 개의 Gaussian component가 있고, index  $k$ 번째의 Gaussian component의 mean 을  $\mu_k$ , covariance matrix를  $\Sigma_k$ 라고 나타냅니다.
- 그리고 여러개의 Gaussian 분포의 가중합(weighted sum)을 나타내기 위해 벡터  $\pi = (\pi_1, \pi_2, \pi_3, \dots, \pi_K)$ 를 사용합니다.

- GMM 알고리즘 또한 EM 알고리즘으로 트레이닝 합니다.
- E-step에서는 각 데이터들이 각 cluster에 속할 확률 값을 계산합니다. 확률값은 Bayes' rule에 의해 계산됩니다. 이 확률값은 responsibility라 부르며  $\gamma(z_{nk})$ 는  $n$ 번째 데이터  $x_n$ 가  $k$ 번째 cluster에 속할 확률을 나타냅니다.

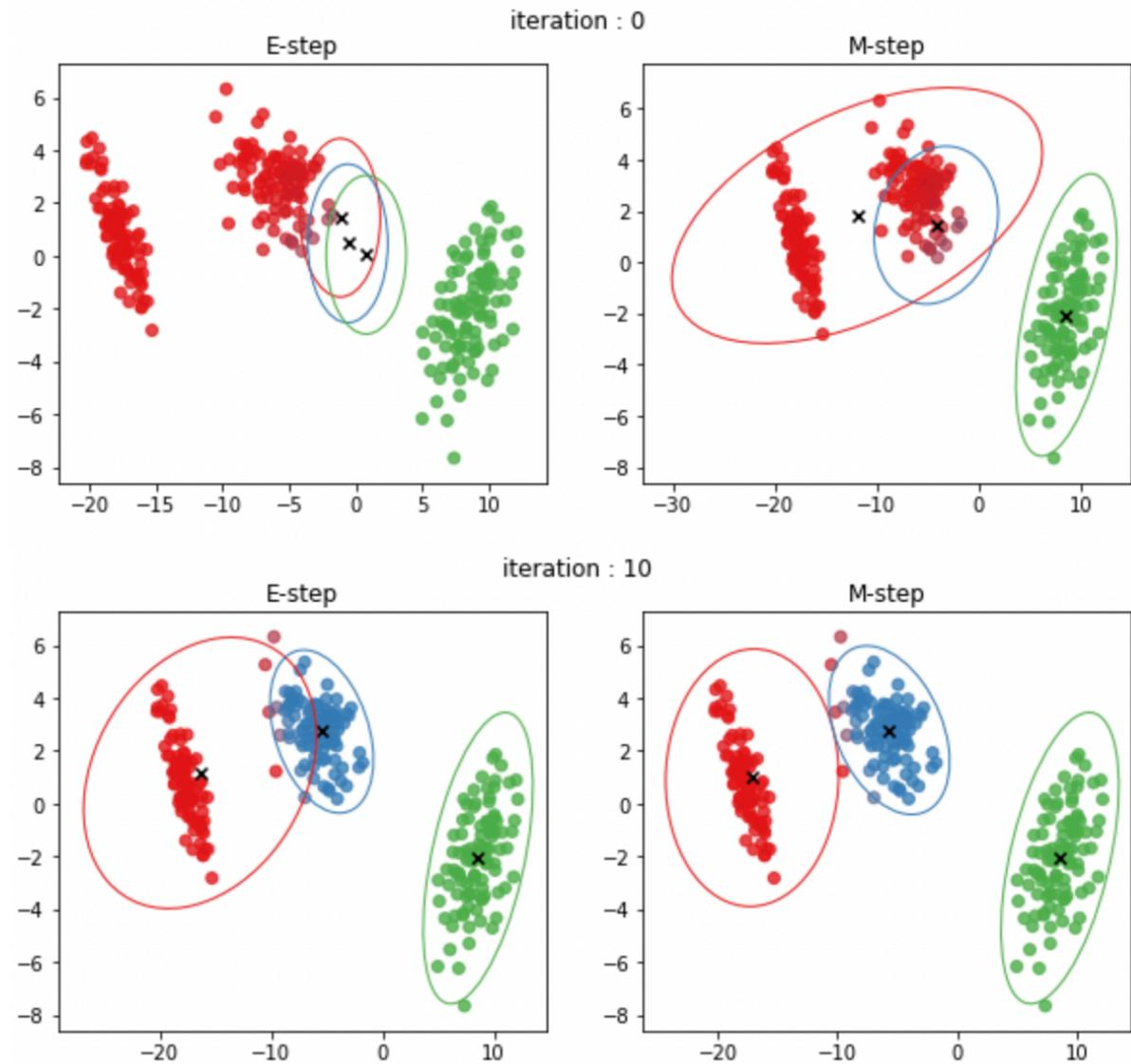
$$\gamma(z_{nk}) = \frac{\pi_k N(x_n | \mu_k, \Sigma_k)}{\sum_{j=1}^K \pi_j N(x_n | \mu_j, \Sigma_j)}$$

- M-step에서는 E-step에서 구한 responsibility를 기반으로, GMM 모델에 필요한 파라미터들 즉, weight  $\pi_k$  mean  $\mu_k$ , covariance  $\Sigma_k$ 를 구합니다.

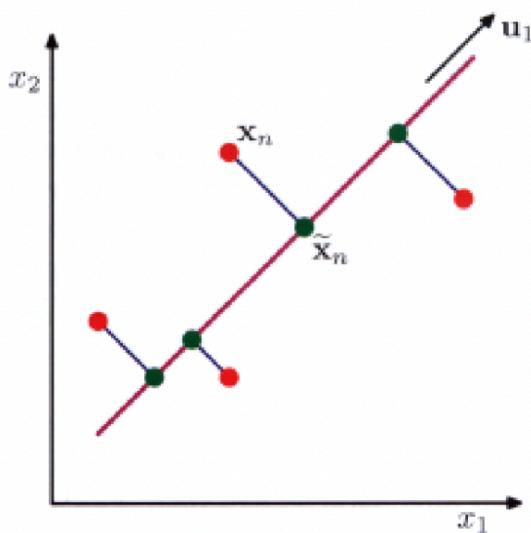
$$\pi_k^{new} = \frac{N_k}{N}, \mu_k^{new} = \frac{1}{N_k} \sum_{n=1}^N \gamma(z_{nk}) x_n, \Sigma_k^{new} = \frac{1}{N_k} \sum_{n=1}^N \gamma(z_{nk}) (x_n - \mu_k^{new}) (x_n - \mu_k^{new})^T$$

$$\text{, where } N_k = \sum_{n=1}^N \gamma(z_{nk})$$

# GMM

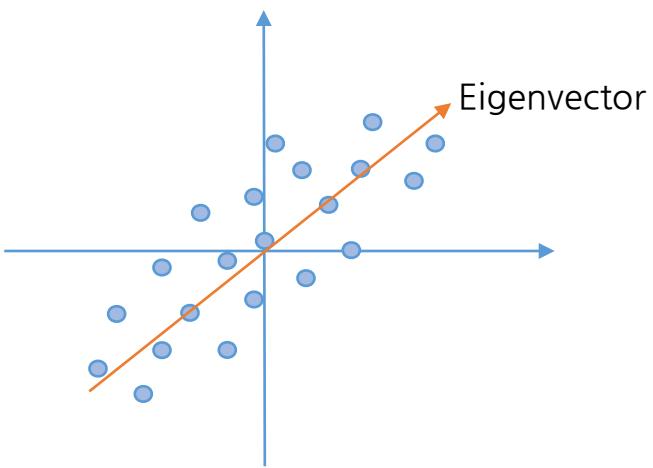


- PCA (Principal Component Analysis)는 dimension reduction의 한 방법입니다.
- 이는 다차원을 가진 data에서 중요한 속성만을 남겨 lossy data compression에 이용되거나, feature extraction으로써 이미지 인식이나 음성 인식 등에 사용될 수 있습니다.
- 또는 2차원이나 3차원에 매핑하여 visualization을 하는데 사용될 수 있습니다.

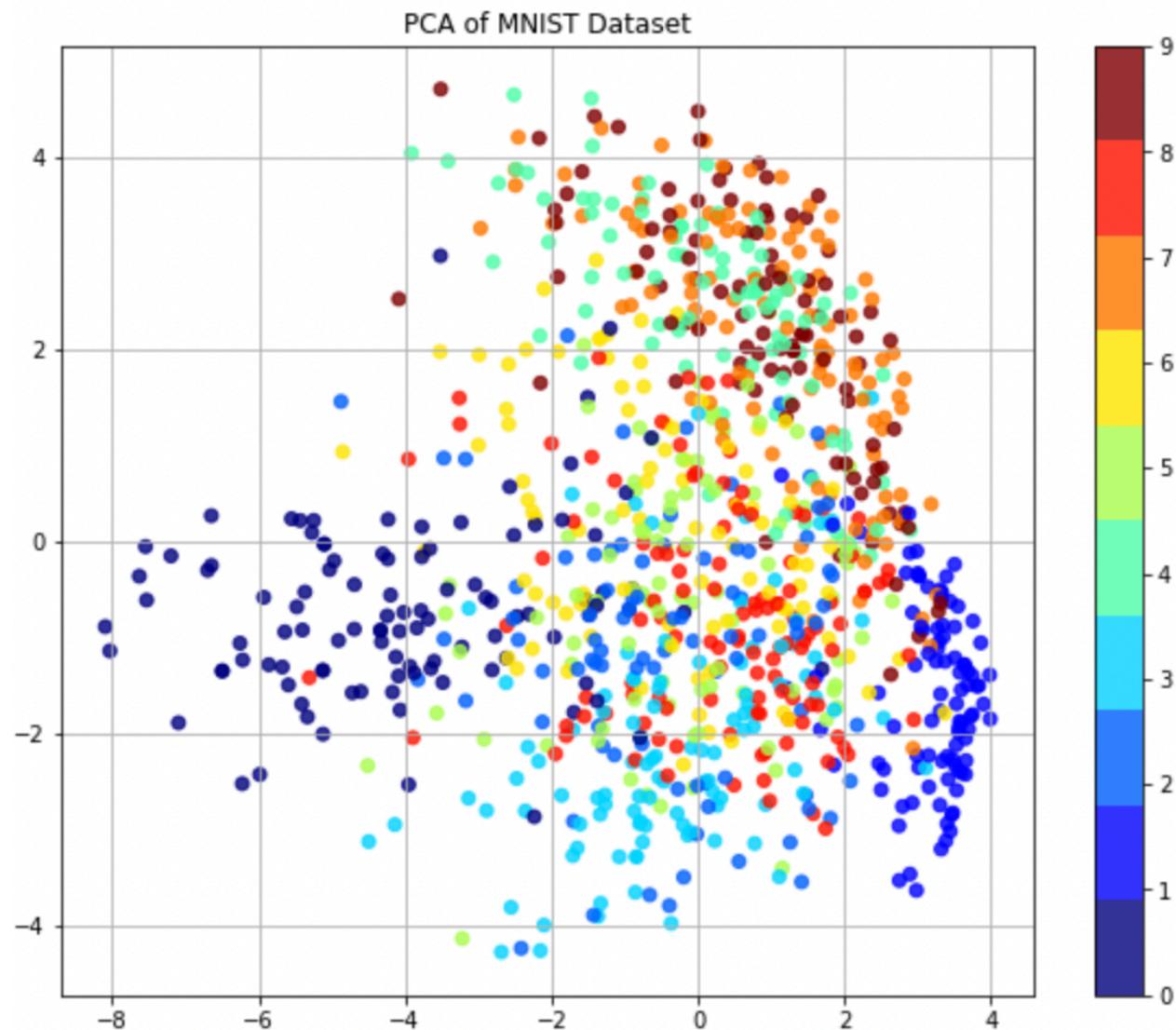


Christopher M. Bishop, Pattern Recognition And Machine Learning 2006, p.561

- PCA는 저차원으로 매핑된 후에도 최대한 정보를 유지하는 방향으로 매핑 함수를 설정합니다.
- 최대한 정보를 유지한다는 조건은 매핑 후에도 각 정보를 최대한 분별 가능하다는 것으로 구체화 합니다.
- 이는 다시, 매핑 후의 데이터들의 분산을 최대화 한다는 것으로 수식화 할 수 있습니다.
- 선형대수학적으로 계산해보면 매핑 후의 데이터들의 분산을 최대화하기 위해서는 data matrix를 eigen decomposition하고, 가장 큰 eigenvalue에 해당하는 eigenvector를 매핑된 space의 basis로 잡아야함을 알 수 있습니다.



# PCA



모두 모두  
파이!!!