5장. EM 알고리즘

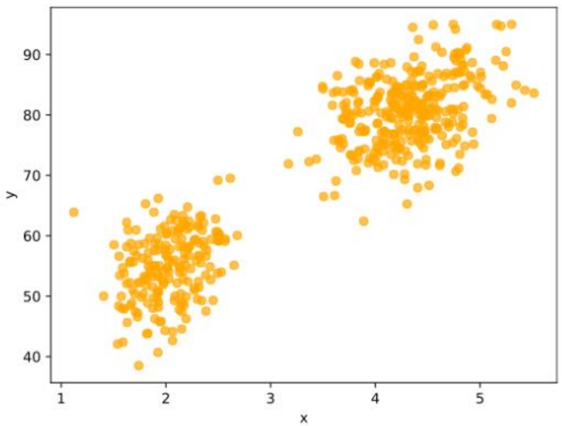
(밑바닥부터 시작하는 딥러닝 5권)

2025. 02. 22

EM 알고리즘

❑ (Remind) 5장에서 해결할 문제

- 주어진 데이터(4장 간헐천 분출)



- 데이터는 GMM 을 따른 다고 가정하고

k=2 라는 것을 안다고 가정

(실제는 k 도 찾아야 함)

$$p(\boldsymbol{x}) = \sum_{k=1}^{K} \phi_k \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

- EM 알고리즘 적용하여

데이터를 가장 잘 설명하는

(로그 가능도를 최대화 하는)

GMM 파라미터 찾기

$$\boldsymbol{\theta} = \{\boldsymbol{\phi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}\}$$

$$\boldsymbol{\phi} = \{\phi_1, \phi_2 \cdots \phi_K\}$$

$$\boldsymbol{\mu} = \{\boldsymbol{\mu}_1, \boldsymbol{\mu}_2 \cdots \boldsymbol{\mu}_K\}$$

$$\boldsymbol{\Sigma} = \{\boldsymbol{\Sigma}_1, \boldsymbol{\Sigma}_2 \cdots \boldsymbol{\Sigma}_K\}$$

Note :

일반적으로 데이터는 벡터

$x^{(i)} = \{ x_1 \ \ x_2 \ \dots \ x_n \}$

EM 알고리즘

❑ EM(Expectation Maximization) 알고리즘이란?

- 통계학에서 **잠재 변수**를 가지는 **확률 모델**의 **파라미터를 반복 과정을 통해 추정**하는 알고리즘
  - 확률 모델 예 : Gaussian Mixture Model,

$$p(x;\theta) = p_\theta(x) = \sum_{k=1}^{K} \phi_k \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

  - 잠재 변수(보통 z 표기 :latent variable) 예 : GMM 에서 관측치 x 가 속한 가우시안 분포(z=k)
  - 파라미터 예: GMM 에서

$$\boldsymbol{\theta} = \{\boldsymbol{\phi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}\}$$

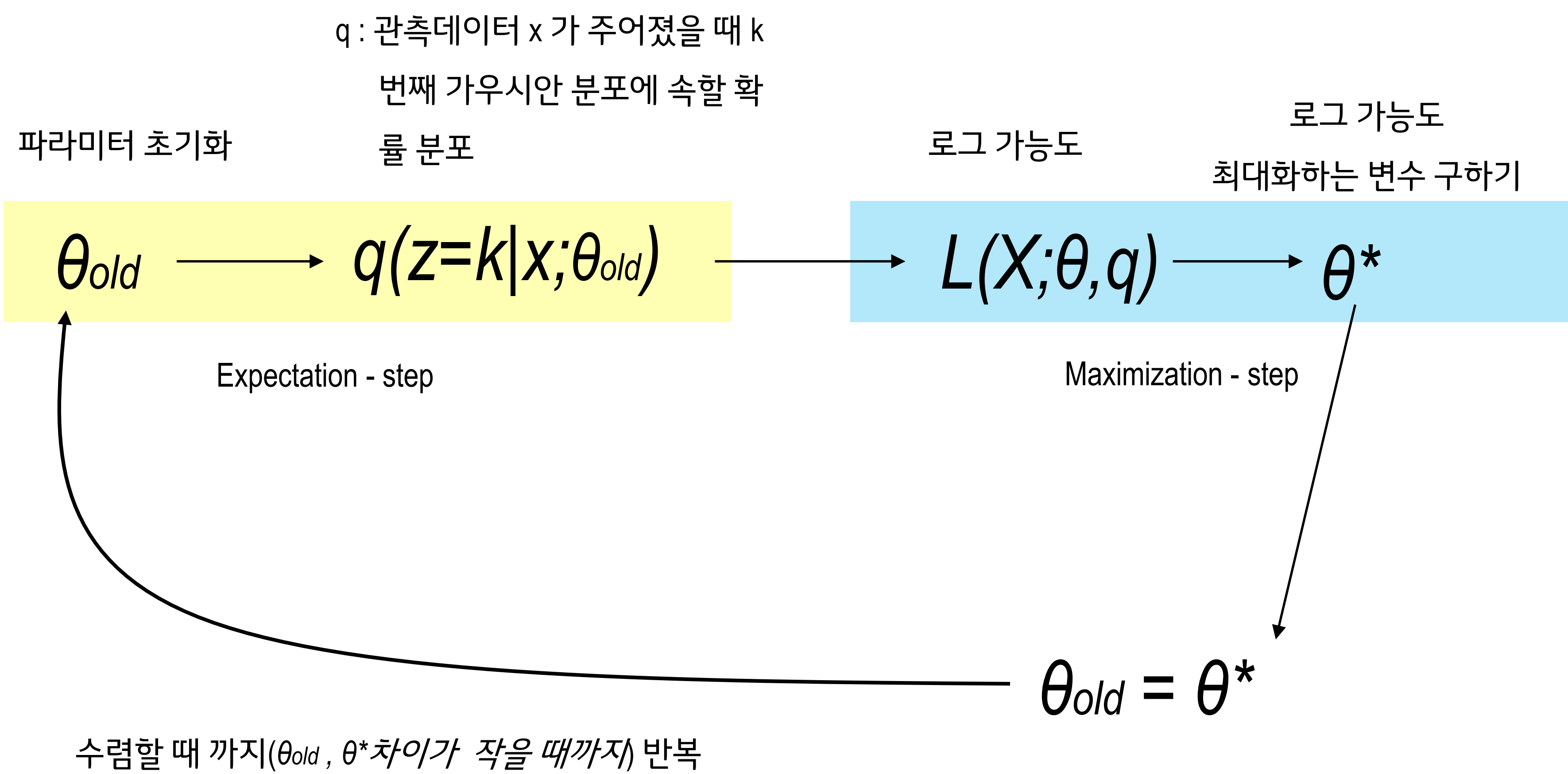
$\boldsymbol{\phi} = \{\phi_1, \phi_2 \cdots \phi_K\}$ GMM 내 특정 가우시안 분포가 선택될 확률

$\boldsymbol{\mu} = \{\boldsymbol{\mu}_1, \boldsymbol{\mu}_2 \cdots \boldsymbol{\mu}_K\}$ GMM 내 각 가우시안 분포의 평균(일반적으로 벡터)

$\boldsymbol{\Sigma} = \{\boldsymbol{\Sigma}_1, \boldsymbol{\Sigma}_2 \cdots \boldsymbol{\Sigma}_K\}$ GMM 내 각 가우시안 분포의 공분산 행렬

EM 알고리즘

❑ 요약 : EM(Expectation Maximization) 알고리즘을 사용하여 GMM 파라미터 θ 추정 방법



EM 알고리즘

- 5.1 KL 발산 : 쿨백-라이블러 발산 (Kullback-Leibler Divergence)
  - 5.1.1 수식 표기법

$$\mathbb{E}_{p(x)}[f(x)] = \int f(x)p(x)dx$$

그림 5-1 변경된 수식 표기법

Before		After
$\mathbb{E}[f(x)]$	$\longrightarrow$	$\mathbb{E}_{p(x)}[f(x)]$
$p(x;\theta)$	$\longrightarrow$	$p_\theta(x)$

EM 알고리즘

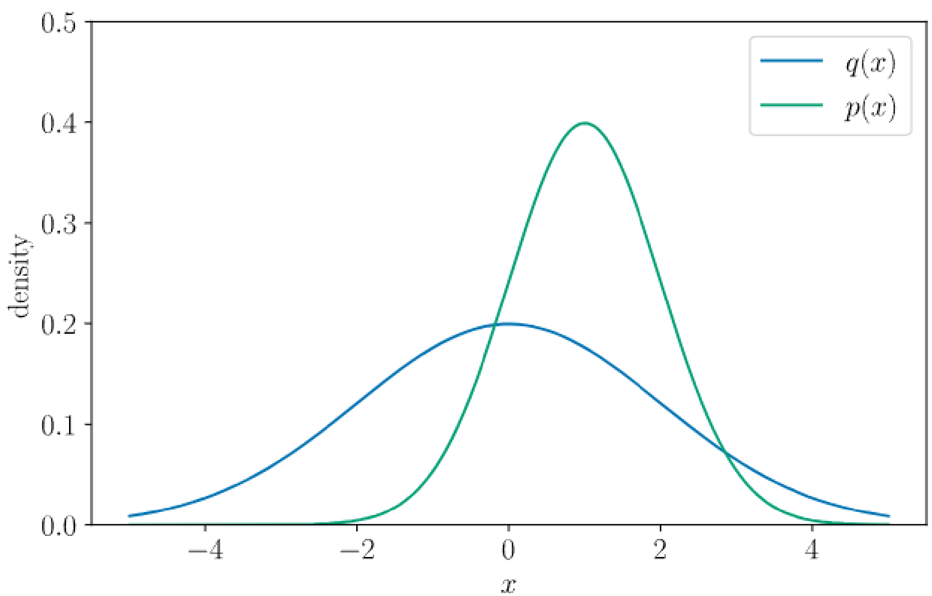
- 5.1.2 KL 발산 정의식
  - 두 확률 분포 p(x) , q(x) 가 어느 정도 유사하는지 비교하는 지표
  - 정의식

$$D_{\mathrm{KL}}(p \parallel q) = \int p(x) \log \frac{p(x)}{q(x)} dx$$

<x 연속 확률 변수>

$$D_{\mathrm{KL}}(p \parallel q) = \sum_x p(x) \log \frac{p(x)}{q(x)}$$

<x 이산 확률 변수>



특징

p,q 가 일치 하면 0

p,q 가 다를 수록 값이 커짐

$D_{KL}(p \parallel q) >= 0$ (Gibb's Inequality)

$D_{KL}(p \parallel q) \neq D_{KL}(q \parallel p)$

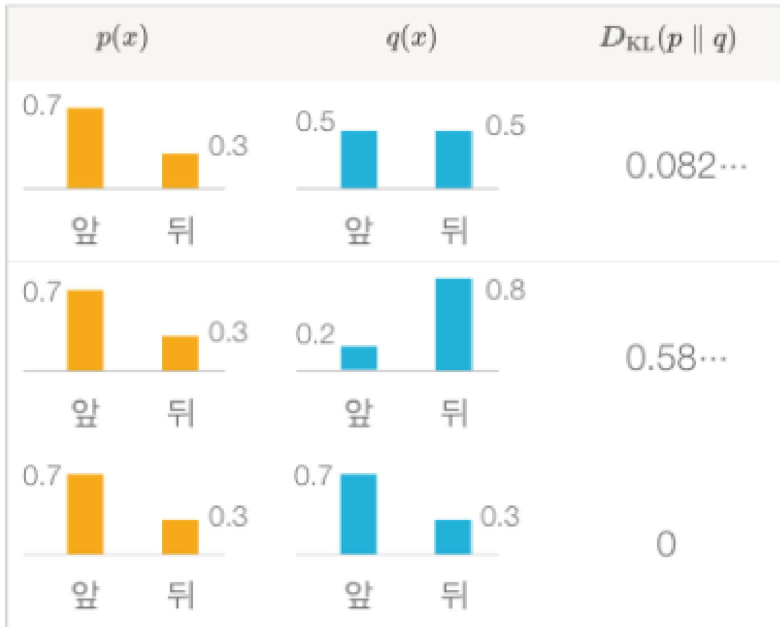
$D_{KL}(p \parallel q)$: p 기준에서 q 가 얼마나 유사한지 계산

$D_{KL}(q \parallel p)$: q 기준에서 p 가 얼마나 유사한지 계산

EM 알고리즘

- KL 발산 계산 예제 – 뒷 페이지의 계산 결과 미리 보기

그림 5-2 세 가지 KL 발산값 비교



EM 알고리즘

- KL 발산 계산 예제

$$D_{\mathrm{KL}}(p \parallel q) = \sum_x p(x) \log \frac{p(x)}{q(x)}$$

case 1 : p, q 가 조금 다를 때

p(x)

p(x=H)	앞면이 나올 확률	70%
p(x=T)	뒷면이 나올 확률	30%

q(x)

q(x=H)	앞면이 나올 확률	50%
q(x=T)	뒷면이 나올 확률	50%

p(H)log{p(H)/q(H)} + p(T)log{p(T)/q(T)}

$$\begin{aligned} D_{\mathrm{KL}}(p \parallel q) &= 0.7 \log \frac{0.7}{0.5} + 0.3 \log \frac{0.3}{0.5} \\ &= 0.082 \cdots \end{aligned}$$

case 2 : p, q 가 많 다를 때

p(x)

앞면이 나올 확률	70%
뒷면이 나올 확률	30%

q(x)

앞면이 나올 확률	20%
뒷면이 나올 확률	80%

$$\begin{aligned} D_{\mathrm{KL}}(p \parallel q) &= 0.7 \log \frac{0.7}{0.2} + 0.3 \log \frac{0.3}{0.8} \\ &= 0.58 \cdots \end{aligned}$$

case 3 : p, q 가 일치

p(x)

앞면이 나올 확률	70%
뒷면이 나올 확률	30%

q(x)

앞면이 나올 확률	70%
뒷면이 나올 확률	30%

$$\begin{aligned} D_{\mathrm{KL}}(p \parallel q) &= 0.7 \log \frac{0.7}{0.7} + 0.3 \log \frac{0.3}{0.3} \\ &= 0.7 \log 1 + 0.3 \log 1 \qquad (\log 1 = 0) \\ &= 0 \end{aligned}$$

EM 알고리즘

- 5.1.3 KL 발산과 최대 가능도 추정(MLE)의 관계
  - 최대 가능도(Maximum Likelihood) 수식을 KL발산식을 이용하여 표현할 수 있음
  - 존재는 하지만 우리가 알 수 없는 모집단 분포 p*(x) 에서 N 개의 샘플 데이터 D 추출 됬다고 가정

$$| \{x^{(1)}, x^{(2)} \cdots x^{(N)}\}$$

  - 우리는 현재 이 데이터만 볼 수 있음. 이 데이터들을 보고 최대한 모집단의 확률 분포 p*(x) 을 추정하려고 함
  - 이제 우리는 모집단이 p_θ (x) 라는 확률 분포는 따른 다고 가정하고(GMM 같은 확률 모델을 설정하고) 로그 가능도를 계산해본다
    (여기서 θ 은 조절 가능 파라미터 , GMM 의 경우 각 가우시안 분포들의 평균 , 공분산 행렬 등)

$$L(\theta) = \log \prod_{n=1}^{N} p_\theta(x^{(n)}) = \sum_{n=1}^{N} \log p_\theta(x^{(n)})$$

  - L(θ) 을 최대화하는 θ 을 구한다

$$\hat{\theta} = \arg\max_\theta \sum_{n=1}^{N} \log p_\theta(x^{(n)})$$

[식 5.1]

EM 알고리즘

- 이 과정을 KL 발산을 사용하여 다르게 표현한다면

$$D_{\mathrm{KL}}(p_* \parallel p_\theta) = \int p_*(x) \log \frac{p_*(x)}{p_\theta(x)} dx \qquad \text{[식 5.2]}$$

'$p_\theta(x)$를 $p_*(x)$에 최대한 가깝게 만든다'

'$p_*(x)$와 $p_\theta(x)$의 KL 발산을 최소화한다'.

- KL발산을 최소화하면 왜 로그 가능도가 최대화되는지는 뒤에서 증명

- 하지만 문제점은
  - 모집단의 분포 p*(x) 은 알 수 없다.(단지 p*(x) 에서 생성된 샘플 데이터만 확인 가능)
  - 모든 x (모든 데이터)에 대해서 적분할 수 없다.

  → [식5.2] 을 몬테카를로 방법으로 근사해보자

EM 알고리즘

- 현재 우리에게는 아래 데이터가 주어져 있다

확률 분포 $p_*(x)$에 따라 샘플 $\{x^{(1)}, x^{(2)} \cdots x^{(N)}\}$

- 이 데이터들을 하나씩 대입해서 결과를 구하고 평균을 내서 $D_{KL}(p^* \| p_\theta)$ 을 근사해보자
- (p*(x) 은 여전히 모르지만 수식으로 표현은 가능)

$$D_{\mathrm{KL}}(p_* \parallel p_\theta) = \int p_*(x) \underbrace{\log \frac{p_*(x)}{p_\theta(x)}}_{f(x)} dx \qquad \text{[식 5.2]}$$

몬테카를로 근사 →
$$\approx \frac{1}{N} \sum_{n=1}^{N} \underbrace{\log \frac{p_*(x^{(n)})}{p_\theta(x^{(n)})}}_{f(x)} \qquad (x^{(n)} \sim p_*(x))$$

$$= \frac{1}{N} \sum_{n=1}^{N} \left(\log p_*(x^{(n)}) - \log p_\theta(x^{(n)}) \right) \qquad \text{[식 5.3]}$$

- $D_{KL}(p^* \| p_\theta)$ 값은 θ 의 함수임 , 우리는 $D_{KL}(p^* \| p_\theta)$ 을 최소화하는 θ 값을 구하고 싶음

$$- \log p_\theta(x^{(n)}) \Big)$$

- $D_{KL}(p^* \| p_\theta)$ 값을 최소화하는 θ 을 구할 때 첫째은 θ 무관함으로 제거하고 두번째 항만 고려하면됨

EM 알고리즘

- 아래와 같이 성립되고 모집단의 분포 p*(x) 을 몰라도 θ 을 구할 수 있음!!!

$$\begin{aligned}\arg\min_{\theta} D_{\mathrm{KL}}(p_* \parallel p_\theta) &\approx \arg\min_{\theta}\left(-\frac{1}{N}\sum_{n=1}^{N}\log p_\theta(x_n)\right)\\ &= \arg\min_{\theta}\left(-\sum_{n=1}^{N}\log p_\theta(x_n)\right)\\ &= \arg\max_{\theta}\sum_{n=1}^{N}\log p_\theta(x_n)\end{aligned}$$

- 결론
  - (우리가 알 수 없는) 모집단의 분포 $p^*(x)$ 와 최대한 유사한 확률 모형 $p_\theta(x)$ 을 만들고 싶으면
    1. 주어진 데이터들과
    2. 조절 가능한 파라미터 θ 가 있는 확률 모형 $p_\theta(x)$ 을 가정해서
    3. 로그 가능도 L(θ) 을 계산하고

$$L(\theta) = \log\prod_{n=1}^{N} p_\theta(x^{(n)}) = \sum_{n=1}^{N}\log p_\theta(x^{(n)})$$

    4. 로그 가능도를 최대화 시키는 θ* 을 찾아서 $p_{\theta^*}(x)$ 을 만들면 된다.(θ* 을 찾을 때 EM 알고리즘 사용 예정)

$$\arg\min_{\theta} D_{\mathrm{KL}}(p_* \parallel p_\theta) \approx \arg\max_{\theta}\sum_{n=1}^{N}\log p_\theta(x_n)$$

EM 알고리즘

- 5.2 EM 알고리즘 도출(유도) – 1
  - 앞으로 하려고 하는 것은
  - $p_\theta(x)$: GMM 으로 모델링
    - 여기서 θ 은

ϕ_k **는** k **번째 정규 분포가 선택될 확률.**

$$\mu = \{\mu_1, \mu_2, \ldots, \mu_K\}$$

$$\Sigma = \{\Sigma_1, \Sigma_2, \ldots, \Sigma_K\}$$

- 여기서 μ_k **는** k **번째 정규 분포의 평균 벡터,**
- Σ_k **는 공분산 행렬**을 의미.

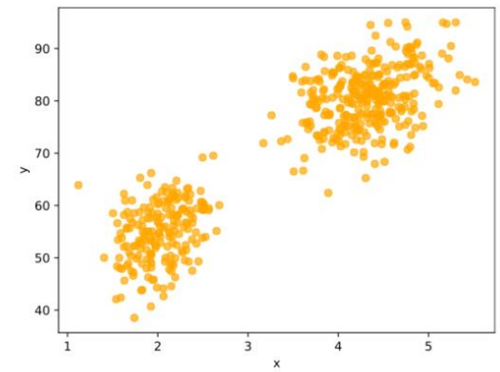
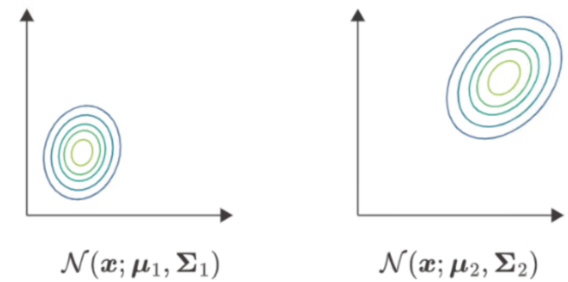


그림 4-8 두 가지 정규 분포의 예

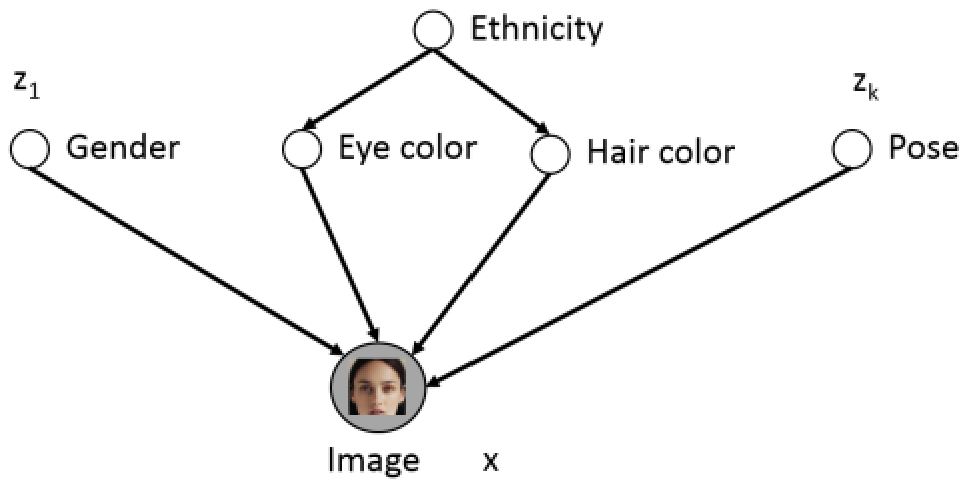


- EM 알고리즘을 사용하여 로그 가능도(L(θ))를 최대화하는 θ* 을 구한다
- GMM 가정한 $p_{\theta^*}(x)$ 으로 주어진 데이터들의 모집단 분포 p*(x) 을 근사하고 신규 데이터를 생성한다.

EM 알고리즘

- 5.2.1 잠재 변수(latent variable)가 있는 모델
  - EM 알고리즘은 잠재 변수가 있는 확률 모델을 대상으로 매개변수 θ 을 갱신하는 방식

  - 잠재 변수에 대하여 (https://wikidocs.net/232062)
    - https://wikidocs.net/229463
      https://wikidocs.net/232062



- 관측 변수 x : 우리가 관찰한 데이터(사람 얼굴 사진 , 녹음한 소리 ...)
- 잠재 변수 z : 실질적으로 x 를 생성하게하는 본질적인 값 , 우리가 관측할 수 없음
  예) 사람 사진에서 사람의 눈색깔 , 머리 모양 , 눈모양 ...
  GMM : 관측 데이터 X 를 생성한 특정 가우시안 분포 z = k

EM 알고리즘

- 잠재 변수를 사용해서 확률 모델 표현하기
  - 확률의 주변화(marginalization)

$$\log p_\theta(x) = \log \sum_z p_\theta(x, z)$$

- 잠재 변수(z)가 포함된 확률 모델로 로그 가능도 표현하기
- 관찰된 데이터 샘플

$$\mathcal{D} = \{x^{(1)}, x^{(2)}, \dots, x^{(N)}\}$$

  - 로그 가능도

$$\begin{aligned} \log p_\theta(\mathcal{D}) &= \log\left(p_\theta(x^{(1)})\, p_\theta(x^{(2)}) \cdots p_\theta(x^{(N)})\right) \\ &= \sum_{n=1}^{N} \log p_\theta(x^{(n)}) \\ &= \sum_{n=1}^{N} \log \sum_{z^{(n)}} p_\theta(x^{(n)}, z^{(n)}) \end{aligned}$$

    - 로그-합 형태라 해석적으로 풀 수 없음 → 합-로그 형태로 변형

EM 알고리즘

- 잠재변수(z) 사용해서 로그 가능도 수식 변형
  - 확률 곱셈 정리

$$\log p_\theta(x) = \log \frac{p_\theta(x, z)}{p_\theta(z \mid x)}$$

$p_\theta(x)$ GMM 이라면
관측 데이터 x 가 z=k 번째 가우시안 분포에서 나왔을 확률 계산할 수 있음!!!

x 가 관측 되었는데 z=k 번째 가우시안 분포에서 나왔을 확률
GMM 예제에서는 계산 가능

참고: 베이즈 정리

$$p_\theta(z \mid x) = \frac{p_\theta(x, z)}{\sum_z p_\theta(x, z)}$$

z 가 연속이거나 수가 많은 경우에 계산이 복잡/불가능할 수 있음

EM 알고리즘

- 5.2.2 임의의 확률 분포 q(z)
  - 잠재 변수가 임의의 확률 분포 q(z)을 따른다고 가정하고 로그 가능도 식에 적용하여 로그-합 형태에서 합-로그 형태로 변형한다

$$
\begin{aligned}
\log p_\theta(x) &= \log \frac{p_\theta(x, z)}{p_\theta(z \mid x)} \\
&= \log \frac{p_\theta(x, z)}{p_\theta(z \mid x)} \frac{q(z)}{q(z)} && \left(\frac{q(z)}{q(z)} = 1\text{을 곱한다.}\right) \\
&= \log \frac{p_\theta(x, z)}{q(z)} + \log \frac{q(z)}{p_\theta(z \mid x)} && \text{[식 5.4]}
\end{aligned}
$$

- 두 번째항에 $p_{\theta^*}(z|x)$ 가 여전히 존재

EM 알고리즘

- *KL 발산식을 활용하여 다시 변형*
  - *하나의 데이터 포인트에 대하여 로그 가능도를 계산하는 식(전체 데이터에 대해서는 뒤에서)*

$$
\begin{aligned}
&\log p_\theta(x) \\
&= \log p_\theta(x) \sum_z q(z) && \left(\sum_z q(z) = 1\text{을 곱한다.}\right) \\
&= \sum_z q(z) \log p_\theta(x) && \left(\log p_\theta(x)\text{를 } \sum \text{의 안으로}\right) \\
&= \sum_z q(z) \left(\log \frac{p_\theta(x,z)}{q(z)} + \log \frac{q(z)}{p_\theta(z \mid x)} \right) && \left([\text{식 } 5.4] \text{ 대입}\right) \\
&= \sum_z q(z) \log \frac{p_\theta(x,z)}{q(z)} + \underbrace{\sum_z q(z) \log \frac{q(z)}{p_\theta(z|x)}}_{\text{KL 발산}} \\
&= \sum_z q(z) \log \frac{p_\theta(x,z)}{q(z)} + D_{\mathrm{KL}}(q(z) \parallel p_\theta(z \mid x)) && [\text{식 } 5.5]
\end{aligned}
$$

EM 알고리즘

- 5.3 EM 알고리즘 도출 - 2

$$\log p_\theta(x) = \sum_z q(z) \log \frac{p_\theta(x,z)}{q(z)} + D_{\mathrm{KL}}(q(z) \parallel p_\theta(z \mid x))$$ [식 5.5]

  - 5.3.1 ELBO(Evidence Lower Bound) , 증거 하한

$$\begin{aligned} \log p_\theta(x) &= \sum_z q(z) \log \frac{p_\theta(x,z)}{q(z)} + \underbrace{D_{\mathrm{KL}}(q(z) \parallel p_\theta(z|x))}_{\geq 0} \\ &\geq \sum_z q(z) \log \frac{p_\theta(x,z)}{q(z)} \end{aligned}$$ [식 5.5]

    - KL발산은 >= 0 이기 때문에

- ELBO

$$\log p_\theta(x) \geq \boxed{\sum_z q(z) \log \frac{p_\theta(x,z)}{q(z)}}$$

EM 알고리즘

- *ELBO 정의*

$$\mathrm{ELBO}(x; q, \theta) = \sum_{z} q(z) \log \frac{p_\theta(x, z)}{q(z)}$$ [식 5.6]

- ELBO 특징
  - 로그 가능도 >= ELBO
  - 식 5.6 , 합-로그 형식이라 해석적으로 계산 용이

NOTE_ **증거 하한**에서 증거는 로그 가능도 $\log p_\theta(x)$의 별칭으로 쓰입니다. 증거라는 이름은 '로그 가능도 값이 커질수록 구하고자 하는 q나 θ가 올바른 방향을 가리키고 있다는 증거다'라는 말에서 유래했습니다. 하한이란 $x \geq a$를 만족할 때 'x의 하한은 a다'라는 뜻입니다. 정리하면 '증거 하한 = 로그 가능도의 최솟값'입니다.

EM 알고리즘

- 5.3.2 드디어 EM 알고리즘으로

$$\mathrm{ELBO}(x; q, \theta) = \sum_z q(z) \log \frac{p_\theta(x, z)}{q(z)}$$ [식 5.6]

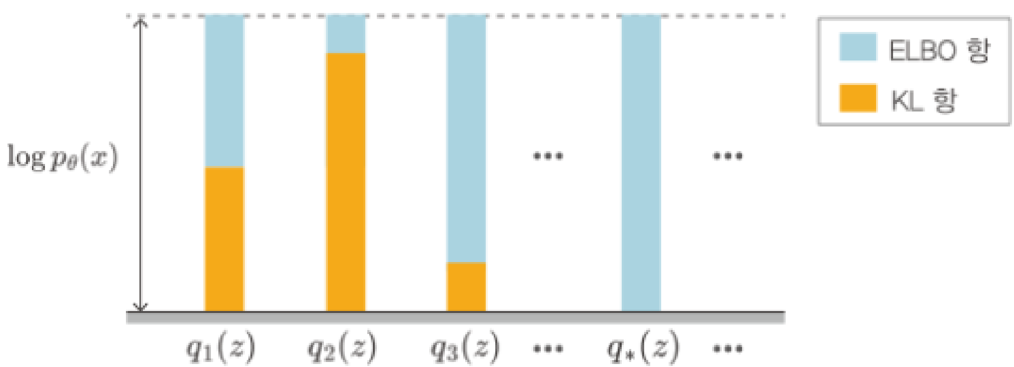
- ELBO 을 사용하여 로그 가능도 $\log p_\theta(x)$ 을 최대화하는 θ 찾기
  - 로그 가능도 $\log p_\theta(x)$ 을 최대화하는 θ 을 직접 찾기는 어려움
  - 로그 가능도 >= ELBO
  - ELBO 을 최대화 하도록 $\theta, q(z)$ 갱신 → 로그 가능도도 최대화
  - 하지만 동시에 θ , $q(z)$ 을 갱신하기는 어려움 → EM 알고리즘

EM 알고리즘

$$\log p_\theta(x) = \mathrm{ELBO}(x; q, \theta) + D_{\mathrm{KL}}(q(z) \parallel p_\theta(z|x)) \qquad \text{[식 5.7]}$$

상수 ELBO 최대화 → KL발산 최소화 → $q(z) = p_\theta(z|x)$ 으로 설정 → $D_{KL} = 0$
→ $\log p_\theta(x) = ELBO$

그림 5-7 $q(z)$가 변해도 ELBO 항과 KL 항의 합은 일정



EM 알고리즘

❑ EM(Expectation Maximization) 알고리즘 적용하여 확률 모델의 파라미터 추정하기

파라미터 초기화

q 를 구함

ELBO 최대화(=로그 가능도 최대화) 하는

θ 구하기 – 교재에서는 미분을 활용해서 구함

$$\theta_{old} \longrightarrow q(z) = p_{\theta_{old}}(z|x) \longrightarrow ELBO(X;\theta,q) \longrightarrow \theta^*$$

Expectation - step

Maximization - step

$$\theta_{old} = \theta^*$$

수렴할 때 까지(θ_{old} , θ^**차이가 작을 때까지*) 반복

EM 알고리즘

❑ GMM 파라미터 추정에 EM(Expectation Maximization) 알고리즘 적용 - 1

- 요약(이 내용이 교재의 코드로 구현)

$$\theta_{(k\text{-}1)}$$

E−스텝

각 n, k에 대해 다음 값을 계산합니다.

$$q^{(n)}(k) = \frac{\phi_k \mathcal{N}(\boldsymbol{x}^{(n)}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} \phi_j \mathcal{N}(\boldsymbol{x}^{(n)}; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}$$

M−스텝

각 k에 대해 다음 값을 계산합니다.

$$\phi_k = \frac{1}{N} \sum_{n=1}^{N} q^{(n)}(k)$$

$$\boldsymbol{\mu}_k = \frac{\sum_{n=1}^{N} q^{(n)}(k)\ \boldsymbol{x}^{(n)}}{\sum_{n=1}^{N} q^{(n)}(k)}$$

$$\boldsymbol{\Sigma}_k = \frac{\sum_{n=1}^{N} q^{(n)}(k)\ (\boldsymbol{x}^{(n)} - \boldsymbol{\mu}_k)(\boldsymbol{x}^{(n)} - \boldsymbol{\mu}_k)^{\top}}{\sum_{n=1}^{N} q^{(n)}(k)}$$

종료 판정

다음 로그 가능도를 계산하여 이전 로그 가능도와 비교합니다.

$$\frac{1}{N} \sum_{n=1}^{N} \log \sum_{j=1}^{K} \phi_j \mathcal{N}(\boldsymbol{x}^{(n)}; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)$$

EM 알고리즘

❑ GMM 파라미터 추정에 EM(Expectation Maximization) 알고리즘 적용 - 2

$\theta_{old} \longrightarrow q(z) = p_{\theta old}(z|x) \longrightarrow ELBO(X;\theta,q) \longrightarrow \theta^*$

1) $\boldsymbol{\phi} = \{\phi_1, \phi_2 \cdots \phi_K\}$

$\boldsymbol{\mu} = \{\boldsymbol{\mu}_1, \boldsymbol{\mu}_2 \cdots \boldsymbol{\mu}_K\}$

$\boldsymbol{\Sigma} = \{\boldsymbol{\Sigma}_1, \boldsymbol{\Sigma}_2 \cdots \boldsymbol{\Sigma}_K\}$

Expectation - step

Maximization - step

3) $$q^{(n)}(z=k) = p(z=k \mid \boldsymbol{x}^{(n)}; \boldsymbol{\theta}) = \frac{p(\boldsymbol{x}^{(n)}, z=k; \boldsymbol{\theta})}{p(\boldsymbol{x}^{(n)}; \boldsymbol{\theta})} = \frac{\phi_k \mathcal{N}(\boldsymbol{x}^{(n)}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} \phi_j \mathcal{N}(\boldsymbol{x}^{(n)}; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}$$

2) $$p(\boldsymbol{x}; \boldsymbol{\theta}) = \sum_{j=1}^{K} p(\boldsymbol{x}, z=j; \boldsymbol{\theta}) = \sum_{j=1}^{K} p(z=j; \boldsymbol{\theta})\, p(\boldsymbol{x} \mid z=j; \boldsymbol{\theta}) = \sum_{j=1}^{K} \phi_j \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)$$

❑ GMM 파라미터 추정에 EM(Expectation Maximization) 알고리즘 - 3

$$\theta_{old} \longrightarrow q(z) = p_{\theta_{old}}(z|x) \longrightarrow ELBO(X;\theta,q) \longrightarrow \theta^*$$

Expectation - step

Maximization - step

1) $\boldsymbol{\phi} = \{\phi_1, \phi_2 \cdots \phi_K\}$

$\boldsymbol{\mu} = \{\boldsymbol{\mu}_1, \boldsymbol{\mu}_2 \cdots \boldsymbol{\mu}_K\}$

$\boldsymbol{\Sigma} = \{\boldsymbol{\Sigma}_1, \boldsymbol{\Sigma}_2 \cdots \boldsymbol{\Sigma}_K\}$

4) $\mathrm{ELBO}(x; q = p_{\theta_{\mathrm{old}}}(z \mid x), \theta) = \sum_z p_{\theta_{\mathrm{old}}}(z \mid x) \log \frac{p_\theta(x,z)}{p_{\theta_{\mathrm{old}}}(z \mid x)}$

$$= \mathbb{E}_{p_{\theta_{\mathrm{old}}}(z|x)}\left[\log \frac{p_\theta(x,z)}{p_{\theta_{\mathrm{old}}}(z \mid x)}\right]$$ [식 5.8]

5) $\dfrac{\partial J}{\partial \boldsymbol{\mu}_k} = 0 \qquad \therefore \ \boldsymbol{\mu}_k = \dfrac{\sum_{n=1}^N q^{(n)}(k)\, \boldsymbol{x}^{(n)}}{\sum_{n=1}^N q^{(n)}(k)}$

$$\boldsymbol{\Sigma}_k = \frac{\sum_{n=1}^N q^{(n)}(k)(\boldsymbol{x}^{(n)} - \boldsymbol{\mu}_k)(\boldsymbol{x}^{(n)} - \boldsymbol{\mu}_k)^\top}{\sum_{n=1}^N q^{(n)}(k)}$$ [식 5

$$\phi_k = \frac{1}{N}\sum_{n=1}^N q^{(n)}(k)$$

- Code Review

EM vs K-means clustering

❑ K-means 클러스터링이 EM 방식의 한 형태인가?

K-means와 EM 알고리즘의 유사점

-K-means 할당 단계 (Assignment) : 데이터 포인트를 가장 가까운 클러스터에 할당
-EM 알고리즘 E-Step : 데이터 포인트를 가장 속할 확률이 높은 클러스터의 확률값 할당

-K-means 업데이트 단계 (Update) : 클러스터 중심을 업데이트하는 과정
-EM 알고리즘의 M-Step: 확률 모델의 매개변수를 최대화하는 과정

차이점

- 확률 모델 사용 여부: K-means는 비확률적(non-probabilistic) 방법인 반면, EM 알고리즘은 확률 모델 기반
- 목표 함수: K-means는 유클리드 거리 제곱합(sum of squared Euclidean distances)을 최소화하려고 하며, EM 알고리즘은 로그 가능도 함수를 최대화하려 함.
- 수렴 조건: K-means는 클러스터 중심이 수렴할 때까지 반복하지만, EM 알고리즘은 로그 가능도 함수가 수렴할 때까지 반복.

결론

K-means 클러스터링은 EM 방식의 한 형태라고 말하기에는 부족한 부분이 존재하지만, 두 알고리즘이 기본적으로 비슷한 반복적 접근 방식을 사용하고 있다는 점에서 유사성을 인정할 수 있습니다. K-means를 EM 알고리즘의 특수한 경우로 볼 수 있다고 주장하는 시각도 있을 수 있습니다. 예를 들어, K-means는 잠재 변수가 클러스터 중심이며 각 데이터가 클러스터에 속할 확률이 단순히 "0 또는 1"로 결정되는 경계 케이스로 볼 수 있습니다. 더 정확히 말하면, K-means 클러스터링은 EM 알고리즘의 비확률적(non-probabilistic) 변형에 가깝다고 할 수 있습니다.