# EM algorithm

## on mixture of Gaussians

인공지능융합학과

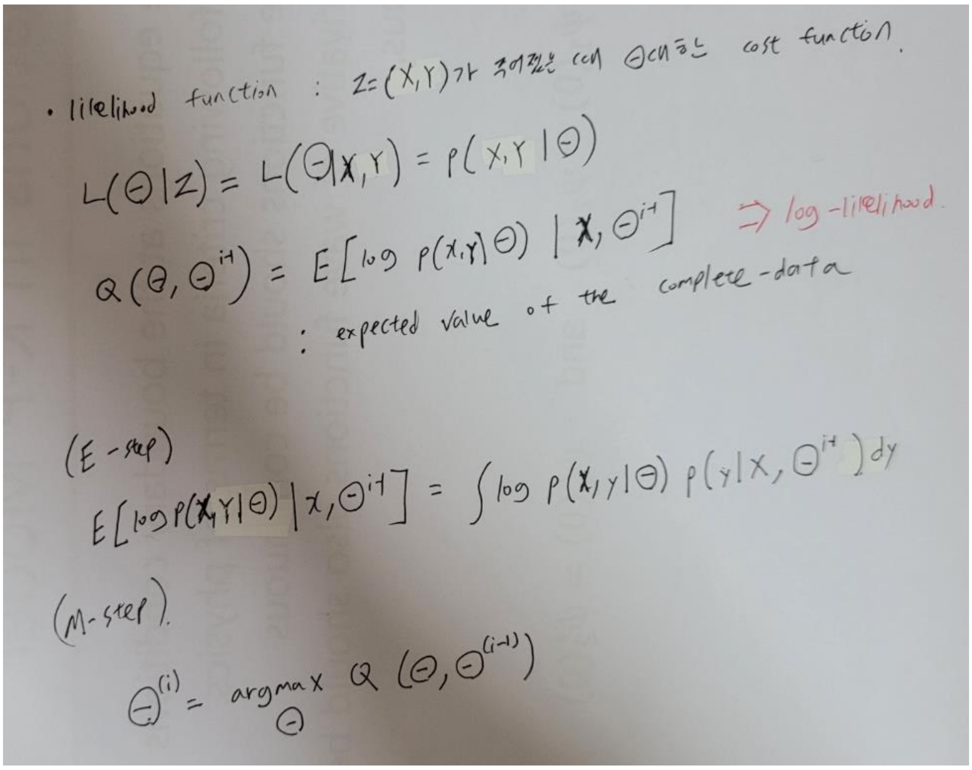
2021123173

송 우 석

## 1. Introduction

### 1.1 EM algorithm

Expectation-maximization 알고리즘은 관측되지 않는 잠재변수에 의존하는 확률 모델에서 "maximum likelihood'를 갖는 모수의 추정값을 찾는 반복적인 알고리즘이다. EM 알고리즘은 모수에 관한 추정값으로 'log likelihood'의 기댓값을 계산하는 (E) 단계와 이 기댓값을 최대화하는 모수 추정값들을 구하는 (M) 단계를 번갈아가면서 적용한다. (M) 단계에서 계산한 변수값은 다음 (E) 단계의 추정값으로 쓰인다. [1]



• likelihood function : $Z=(X,Y)$가 주어졌을 때 $\Theta$에 대한 cost function.

$$L(\Theta|Z) = L(\Theta|X,Y) = p(X,Y|\Theta)$$

$$Q(\Theta,\Theta^{i-1}) = E[\log p(X,Y|\Theta) \mid X,\Theta^{i-1}]$$ ⇒ log-likelihood.

: expected value of the complete-data

(E-step)

$$E[\log p(X,Y|\Theta) \mid X,\Theta^{i-1}] = \int \log p(X,y|\Theta)\, p(y|X,\Theta^{i-1})\, dy$$

(M-step).

$$\Theta^{(i)} = \underset{\Theta}{\operatorname{argmax}}\ Q(\Theta,\Theta^{(i-1)})$$

## 1.2 Mixture of Gaussian

### 1.2.1 Gaussian distribution

정규분포는 2개의 매개 변수 평균과 표준편차 에 대해 모양이 결정되는 연속 확률 분포이다.

Gaussian 분포. $(\theta_j = \mu_j, \Sigma_j)$

$$p_j(x|\theta_j) = p_j(x|\mu_j, \Sigma_j)$$

$$= \frac{1}{(2\pi)^{\frac{d}{2}} |\Sigma_j|^{1/2}} \exp\left[ -\frac{1}{2}(x-\mu_j)^T \Sigma_j^{-1} (x-\mu_j) \right]$$

### 1.2.2 Mixture model : linear combination of m densities

$$p(x|\Theta) = \sum_{j=1}^{m} \alpha_j p_j(x|\theta_j) \quad \cdots\cdots ②$$

where $\Theta = (\alpha_1 \cdots \alpha_m, \theta_1 \cdots \theta_m)$

(s.t $\alpha_j \geq 0$, $\sum_1^m \alpha_j = 1$)

. The loglikelihood function of $\Theta$

$$\log L(\Theta|X,Y) = \log p(X,Y|\Theta) \quad ①$$

$$= \log p(X|Y,\Theta)\, p(Y|\Theta)$$

$$= \sum_{i=1}^{n} \log\left(p(x_i|y_i, \theta_i)\, p(y_i|\theta_i)\right)$$

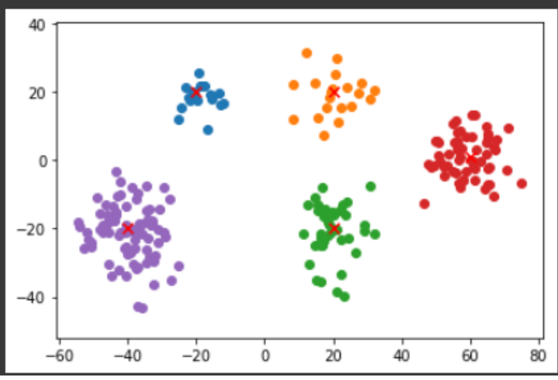
$$= \sum_{i=1}^{n} \log\left(\alpha_{y_i} p_{y_i}(x_i|\theta_{y_i})\right)$$

## 2. Experiment Method

### 2.1 Mixture of Gaussian(MoG) -2차원

밑의 그림과 같이 평균과 분산을 입력해 주면 여러개의 가우시안 분포가 생성되고 , 생성된 데이터를 모두 모아서 Mixture of Gaussian을 만들었다. 그림은 different spherical covariance를 가지는 5개의 가우시안 분포의 mixture 이다.

```
kkk=5 #class 갯수
NUM= [20,20,40,60,80] #각 클래스 별 데이터 수.
Means=[ #각 class의 평균
      [-20, 20],
      [20, 20],
      [20, -20],
      [60, 1],
      [-40, -20]]
COVS=[ #각 클래스 분산
         [[20, 0], [0, 20]],
         [[30, 0], [0, 30]],
         [[40, 0], [0, 40]],
         [[50, 0], [0, 50]],
         [[60, 0], [0, 60]]]

X=Gausion_Mixture(kkk,NUM,Means,COVS)
```




- MoG를 생성할 때 다음과 같이 5가지 분산 형태를 사용해서 실험을 했다.

  Case1. Same spherical covariance

  Case2. Different spherical covariance

  Case3. Different diagonal covariance

  Case4. Arbitrary same covariance

  Case5. Arbitrary covariance

- 각각의 case 마다 가우시안 분포를 5개 , 7개로 설정해서 실험을 했다.

  -총 10가지(5가지 variance에 대해서 5개 섞었을 때, 7개 분포 섞었을 때 ->5*2=10 ) 분포의 결과를 명시했습니다. 7개의 분포를 섞을 때는 분포들 끼리 겹치게 설정을 하기도 했습니다.

## 2.2 EM algorithm on MoG ( 코드는 ipynb 파일로 따로 첨부하겠습니다)

### 2.2.1 Expectation step

(E-step)

$\Theta^{(g)}$ : initial parameters $= (\alpha_1^{(g)} \dots \alpha_m^{(g)}, \theta_1^{(g)} \dots \theta_m^{(g)})$

① $p(y_i | x_i, \Theta^{(g)}) = \frac{p(y_i, x_i | \Theta^{(g)})}{p(x_i | \Theta^{(g)})}$ ① $= \frac{\alpha_{y_i}^{(g)} p_{y_i}(x_i | \theta_{y_i}^{(g)})}{p(x_i | \Theta^{(g)})}$

② $= \frac{\alpha_{y_i}^{(g)} p_{y_i}(x_i | \theta_{y_i}^{(g)})}{\sum_{j=1}^{m} \alpha_j^{(g)} p_j(x_i | \theta_j^{(g)})}$ $\Rightarrow p(Y | X, \Theta^{(g)}) = \prod_{i=1}^{n} p(y_i | x_i, \Theta^{(g)})$

$Q(\Theta, \Theta^{(g)}) = \sum_{y} \log p(X, y | \Theta) p(y | X, \Theta^{(g)})$

$\sum_{i=1}^{n} \sum_{j=1}^{m} p(j | x_i, \Theta^{(g)})$

$\|$

$\log p(X, Y | \Theta)$

$\|$

$\sum_{i=1}^{n} \log(\alpha_{y_i} p_{y_i}(x_i | \theta_{y_i})) = \sum_{i=1}^{n} \sum_{j=1}^{m} \log(\alpha_j p_j(x_i | \theta_j))$

$Q = \sum_{j=1}^{m} \sum_{i=1}^{n} \left[ \log(\alpha_j) p(j | x_i, \Theta^{(g)}) + \log(p_j(x_i | \theta_j)) p(j | x_i, \Theta^{(g)}) \right]$

### 2.2.2 Maximization step

#### 2.2.2.1 알파(mixing parameters) 업데이트

$$\max_{\alpha} Q \quad \left(s.t \quad \alpha_j \geq 0, \ \sum_j \alpha_j = 1\right)$$

$$\Rightarrow \frac{\partial\left(Q + \lambda\left(\sum_j \alpha_j - 1\right)\right)}{\partial \alpha_j} = \frac{\partial Q}{\partial \alpha_j} + \lambda = 0 \quad \Rightarrow \quad \frac{\partial Q}{\partial \alpha_j} = -\lambda$$

$$\frac{1}{\alpha_j} \underbrace{\sum_{i=1}^{n} p(j|x_i, \Theta^{(g)})}_{b_j} = -\lambda \quad \Rightarrow \quad \alpha_j = \frac{b_j}{-\lambda}$$

$\left(\sum_j \alpha_j = 1\right.$ 이므로

$$\Downarrow$$

$$-\frac{\sum b_j}{\lambda} = 1$$

$$\therefore \ -\lambda = \underbrace{\sum_j b_j}$$

$$\alpha_j = \frac{b_j}{\sum_j b_j}$$

$$\parallel$$

$$\frac{\sum_{i=1}^{n} p(j|x_i, \Theta^{(g)})}{\sum_i \underbrace{\sum_j p(j|x_i, \Theta^{(g)})}_{=1}} = \frac{\sum_{i=1}^{n} p(j|x_i, \Theta^{(g)})}{n}$$

### 2.2.2.2 평균 업데이트

ii) $\theta_j$ 찾기. $- \mu_j$

Gaussian 분포. $(\theta_j = \mu_j, \Sigma_j)$

$$p_j(x|\theta_j) = p_j(x|\mu_j, \Sigma_j)$$

$$= \frac{1}{(2\pi)^{\frac{d}{2}} |\Sigma_j|^{1/2}} \exp\left[-\frac{1}{2}(x-\mu_j)^T \Sigma_j^{-1} (x-\mu_j)\right]$$

$$\frac{\partial Q}{\partial \mu_j} = \frac{\partial\left[\sim + \sum\sum \log(p_j(x_i|\theta_j))\, p(j|x_i, \Theta^{(g)})\right]}{\partial \mu_j} = 0$$

$$= \frac{\partial\left[\sum_i \sum_j \left[-\frac{d}{2}\log 2\pi - \frac{1}{2}\log|\Sigma_j| - \frac{1}{2}(x_i-\mu_j)^T \Sigma_j^{-1}(x_i-\mu_j)\right] p_{ji}\right]}{\partial \mu_j}$$

$$= -\cancel{\frac{1}{2}} \sum_i \cancel{\sum_j} \Sigma^{-1}(x_i-\mu_j)\, p_{ji} \qquad \therefore \hat{\mu}_j = \frac{\sum_i x_i p_{ji}}{\sum_i p_{ji}}$$

### 2.2.2.3 분산 업데이트

iii) $\Sigma_j$ 찾기.

$$\frac{\partial Q}{\partial \Sigma_j} = \sum_i \left(-\frac{1}{2}\Sigma_j^{-1} + \frac{1}{2}\Sigma_j^{-1}(x_i-\mu_j)(x_i-\mu_j)^T \Sigma_j^{-1}\right) p_{ji} = 0$$

각항 양쪽에 $\Sigma_j$ 곱해.

$$\sum_i \left(-\Sigma_j + (x_i-\mu_j)(x_i-\mu_j)^T\right) p_{ji} = 0$$

$$\Rightarrow \sum_i p_{ji} \Sigma_j = \sum_i (x_i-\mu_j)(x_i-\mu_j)^T p_{ji}$$

$$\Sigma_j = \frac{\sum_i p_{ji}(x_i-\mu_j)(x_i-\mu_j)^T}{\sum_i p_{ji}}$$

### 2.2.3 Log-likelihood

Log-likelihood를 최대화 시키는 parameter을 찾기위해 E-step과 M-step의 iteration을 반복했습니다.

iteration 마다 계산된 Log-likelihood를 그래프로 나타내었고,

Epsilon= 1e-5라는 threshold를 설정하여 iteration을 마친 후 최대화된 log-likelihood를 적었습니다.

### 2.2.4 (Optimized) Parameters

원래 분포의 평균,분산, mixing parameter(알파) 와 EM 알고리즘 적용 후의 평균,분산, mixing parameter를 적었습니다. (반올림해서 적어놓았습니다.)

### 2.2.5 Clustering 정확도

새로 구한 parameter들과 E-step에서 구한 확률로 클러스터링을 하였고, (그 결과를) 색깔을 다르게 하여 시각화 했고, 클러스터링된 가우시안 분포를 잘 보여주는 타원을 그렸습니다.

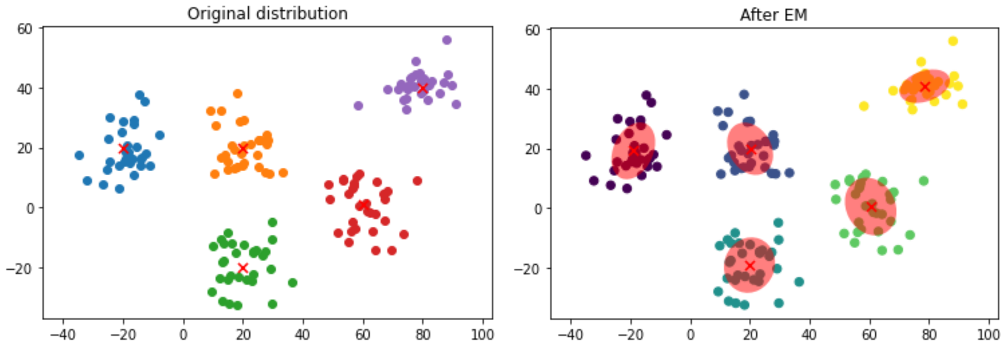
또한 원래 분포의 라벨(클러스터 라벨)과 비교하여 클러스터링 정확도를 계산했습니다.

# 3. Result

## 3.1 5개의 Gaussian mixture

- Case1. Same spherical covariance

  ■ 원래분포 vs 새로 구한 parameter와 확률로 클러스터링 한 분포




  ■ Clustering 정확도: 100%

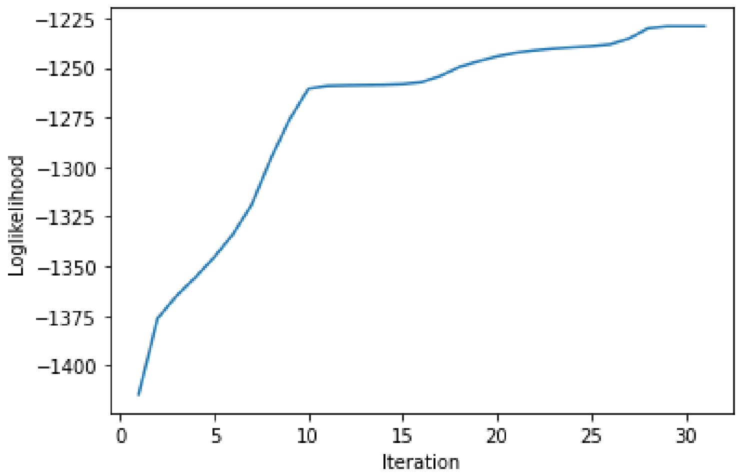
  ■ 원래 평균,분산,mixing parameter vs 새로 구한 평균,분산,mixing parameter

```
원래분포
mean
[[-20  20]
 [ 20  20]
 [ 20 -20]
 [ 60   1]
 [ 80  40]]

coVariance
[[[50  0]
  [ 0 50]]

 [[50  0]
  [ 0 50]]

 [[50  0]
  [ 0 50]]

 [[50  0]
  [ 0 50]]

 [[50  0]
  [ 0 50]]]

alpha
[0.2, 0.2, 0.2, 0.2, 0.2]
```

```
After EM
mean
[[ 20.1   19.8]
 [ 78.6   40.8]
 [ 19.9  -19.2]
 [-19.    19.1]
 [ 60.5    0.5]]

coVariance
[[[ 37.9 -10.6]
  [-10.6  49.4]]

 [[ 45.5    9.9]
  [  9.9   20.1]]

 [[ 46.9    3.8]
  [  3.8   55.4]]

 [[ 34.4   13.5]
  [ 13.5   58.2]]

 [[ 47.8   -9.1]
  [ -9.1   60.1]]]

alpha
[0.2 0.2 0.2 0.2 0.2]
```

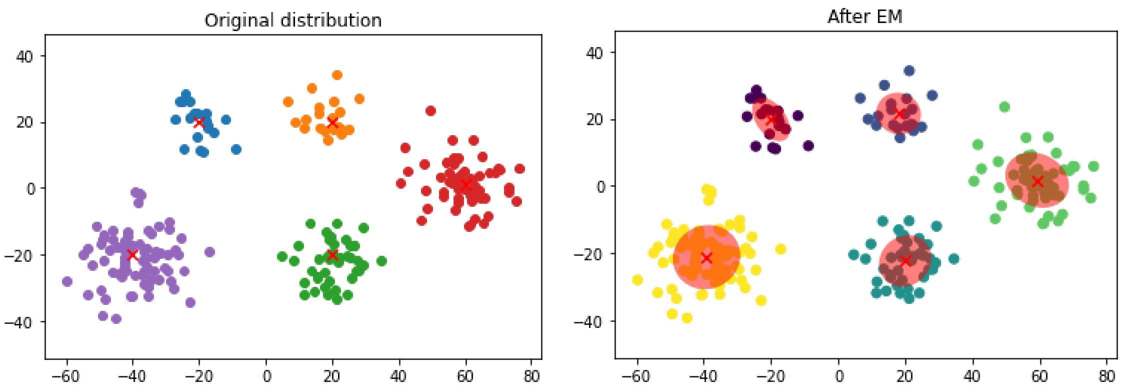
■ Loglikelihood 변화 (iteration 31번)

Maximized loglikelihood = -1229.114379




● Case2. Different spherical covariance

■ 원래분포 vs 새로 구한 parameter와 확률로 클러스터링 한 분포




■ Clustering 정확도: 100%

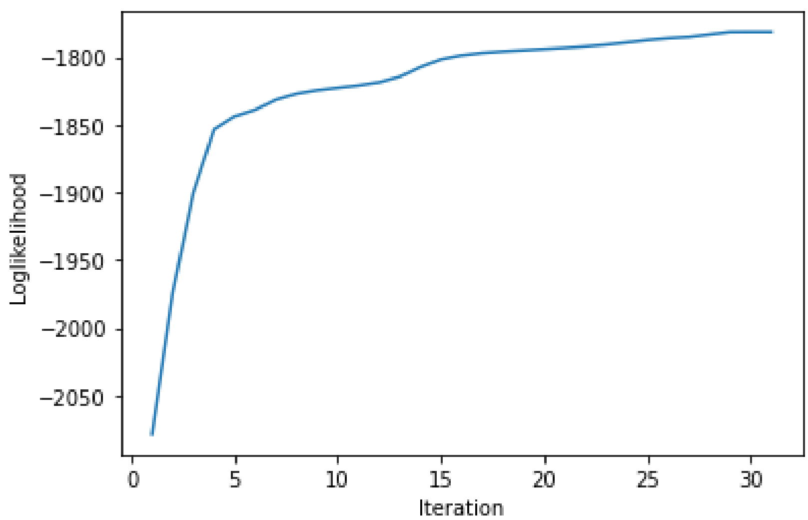
■ 원래 평균과 분산 vs 새로구한 평균과 분산

```
원래분포
mean
[[-20  20]
 [ 20  20]
 [ 20 -20]
 [ 60   1]
 [-40 -20]]

coVariance
[[[20  0]
  [ 0 20]]

 [[30  0]
  [ 0 30]]

 [[40  0]
  [ 0 40]]

 [[50  0]
  [ 0 50]]

 [[60  0]
  [ 0 60]]]

alpha
[0.09 0.09 0.18 0.27 0.36]
```

```
After EM
mean
[[-39.1 -21.1]
 [ 18.2  21.5]
 [ 59.4   1.7]
 [-19.9  19.7]
 [ 20.1 -22.3]]

coVariance
[[[ 64.3   3.5]
  [  3.5  58.2]]

 [[ 28.3  -1.4]
  [ -1.4  24.5]]

 [[ 57.6  -9. ]
  [ -9.   43.2]]

 [[ 20.2 -10.5]
  [-10.5  26.5]]

 [[ 39.1   4.6]
  [  4.6  37.9]]]

alpha
[0.4 0.1 0.3 0.1 0.2]
```

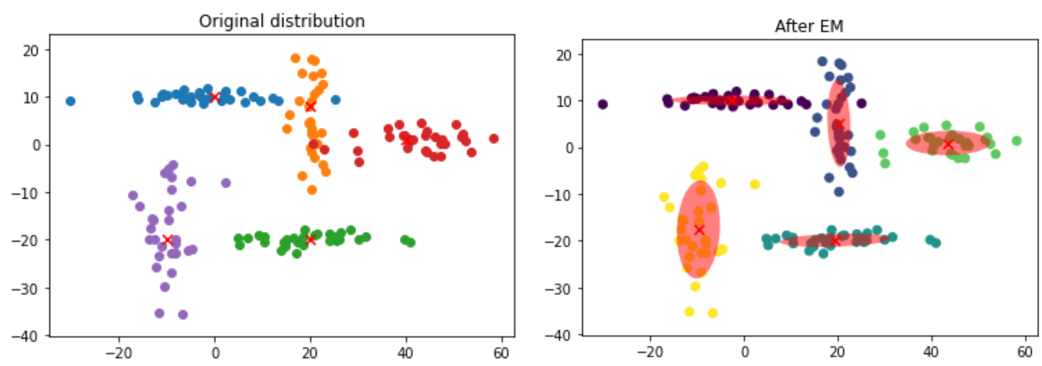
■ Loglikelihood 변화 (iteration 31번)

Maximized loglikelihood = -1781.499086



- Case3. Different diagonal covariance

  - 원래분포 vs 새로 구한 parameter와 확률로 클러스터링 한 분포




  - Clustering 정확도: 98.7%

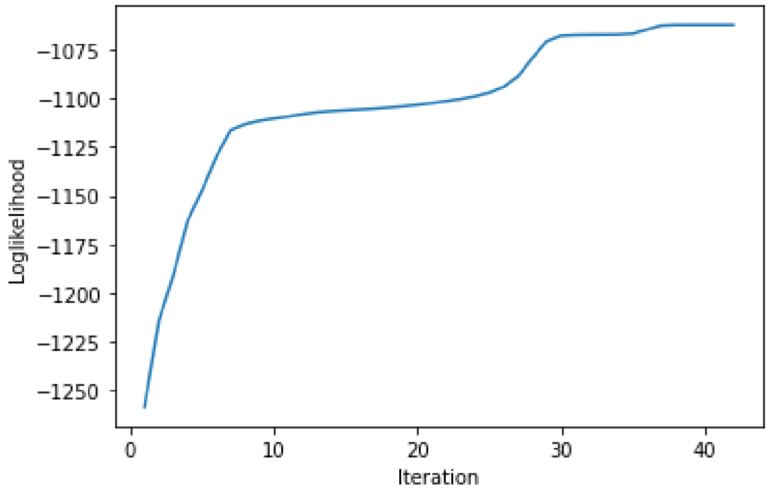
  - 원래 평균,분산,mixing parameter vs 새로 구한 평균,분산,mixing parameter

```
원래분포
mean
[[  0  10]
 [ 20   8]
 [ 20 -20]
 [ 40   1]
 [-10 -20]]

coVariance
[[[100   0]
  [  0   1]]

 [[  5   0]
  [  0  80]]

 [[120   0]
  [  0   2]]

 [[100   0]
  [  0   5]]

 [[ 10   0]
  [  0  70]]]

alpha
[0.2 0.2 0.2 0.2 0.2]
```

```
After EM
mean
[[ 19.4 -19.9]
 [ 20.3   5.1]
 [ -2.7  10. ]
 [ 43.6   0.9]
 [ -9.7 -17.6]]

coVariance
[[[ 88.3   2.6]
  [  2.6   1.3]]

 [[  3.9  -1.8]
  [ -1.8  56. ]]

 [[107.4  -0.7]
  [ -0.7   0.7]]

 [[ 52.3   0.7]
  [  0.7   4.4]]

 [[ 13.5   2.7]
  [  2.7  70.2]]]

alpha
[0.2 0.2 0.2 0.2 0.2]
```

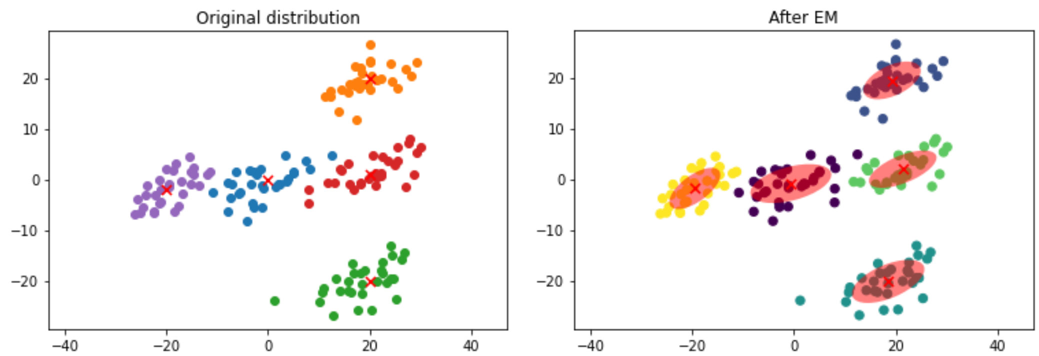
■ Loglikelihood 변화 (iteration 43번)

Maximized loglikelihood = -1062.542528




● Case4. Arbitrary same covariance

■ 원래분포 vs 새로 구한 parameter와 확률로 클러스터링 한 분포




■ Clustering 정확도: 98.7%

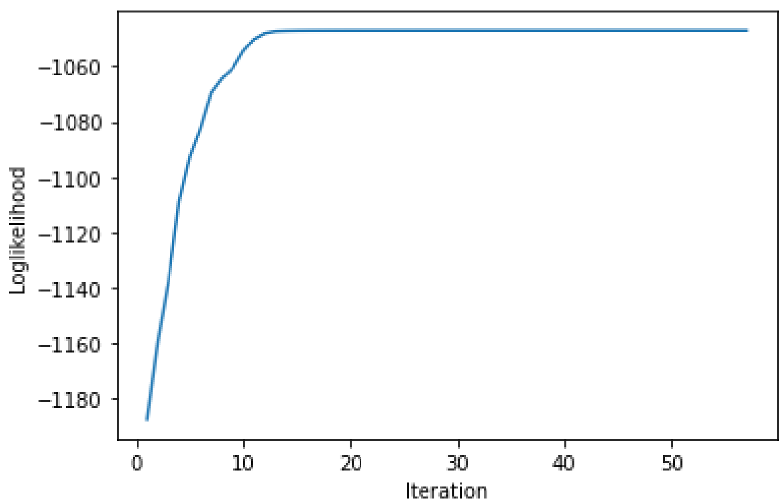
■ 원래 평균,분산,mixing parameter vs 새로 구한 평균,분산,mixing parameter

```
원래분포
mean
[[  0   0]
 [ 20  20]
 [ 20 -20]
 [ 20   1]
 [-20  -2]]

coVariance
[[[10 10]
  [30 10]]

 [[10 10]
  [30 10]]

 [[10 10]
  [30 10]]

 [[10 10]
  [30 10]]

 [[10 10]
  [30 10]]]

alpha
[0.2 0.2 0.2 0.2 0.2]
```

```
After EM
mean
[[-19.5  -1.9]
 [ 21.4   1.9]
 [ 18.5 -20.2]
 [ -0.6  -0.9]
 [ 19.3  19.4]]

coVariance
[[[16.1  8.5]
  [ 8.5 10.6]]

 [[28.4  9.7]
  [ 9.7  8.7]]

 [[32.8 10.9]
  [10.9 11.6]]

 [[40.7  8.9]
  [ 8.9  9.4]]

 [[20.4  6.9]
  [ 6.9  8.9]]]

alpha
[0.2 0.2 0.2 0.2 0.2]
```

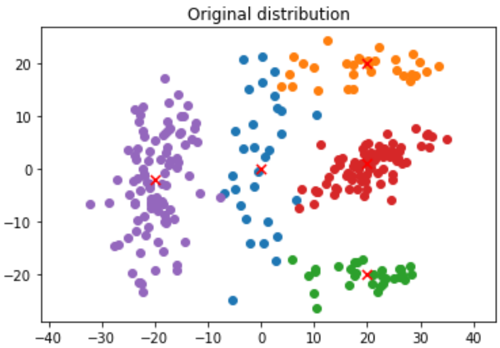
■ Loglikelihood 변화 (iteration 57번)

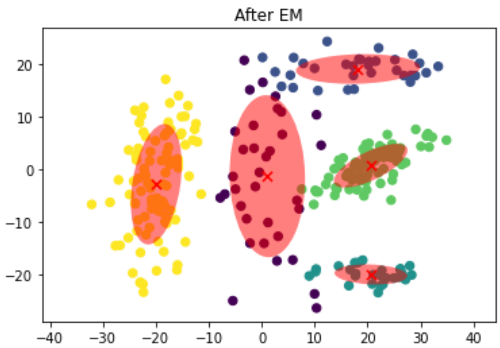
Maximized loglikelihood = -1047.057826



- Case5. Arbitrary covariance

  - 원래분포 vs 새로 구한 parameter와 확률로 클러스터링 한 분포




  - Clustering 정확도: 97%

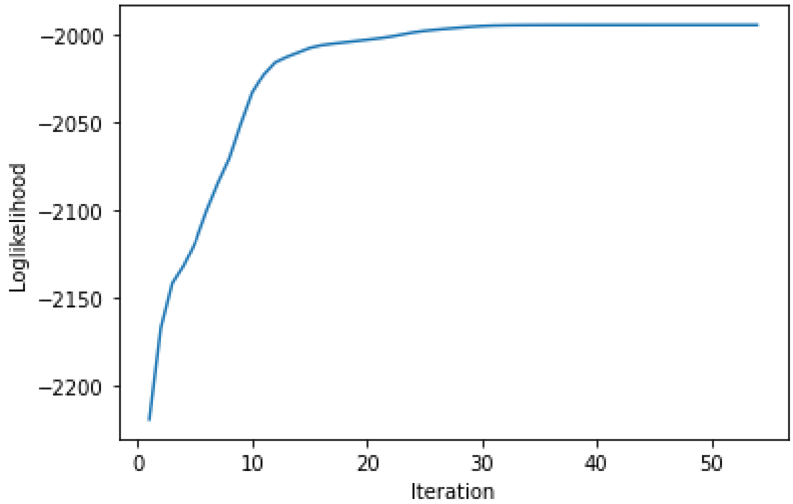
  - 원래 평균,분산,mixing parameter vs 새로 구한 평균,분산,mixing parameter

```
원래분포
mean
[[  0   0]
 [ 20  20]
 [ 20 -20]
 [ 20   1]
 [-20  -2]]

coVariance
[[[ 16  10]
  [ 10 100]]

 [[ 50   7]
  [  1   7]]

 [[ 10   4]
  [ 70   6]]

 [[ 10  10]
  [ 30  10]]

 [[ 10  70]
  [ 20  26]]]

alpha
[0.11 0.11 0.11 0.3  0.37]
```

```
After EM
mean
[[ 20.6   0.7]
 [  1.1  -1.3]
 [ 20.6 -19.9]
 [ 18.2  19. ]
 [-19.9  -2.8]]

coVariance
[[[ 31.   13. ]
  [ 13.   11. ]]

 [[ 32.4  -2.5]
  [ -2.5 150.7]]

 [[ 30.1  -1.2]
  [ -1.2   2.3]]

 [[ 88.    2.3]
  [  2.3   5.1]]

 [[ 14.8  11.2]
  [ 11.2  83.4]]]

alpha
[0.3 0.1 0.1 0.1 0.4]
```

■ Loglikelihood 변화 (iteration 54번)

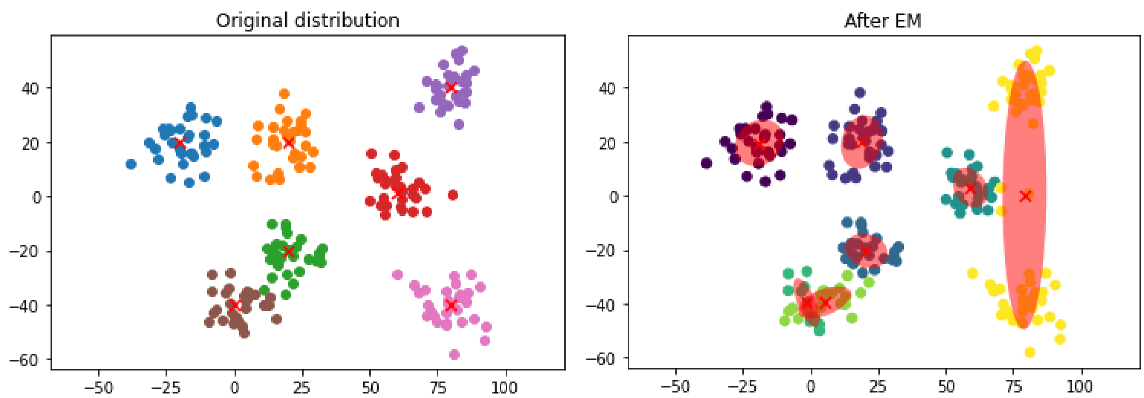
Maximized loglikelihood = -1994.509828




### 3.2 7개의 Gaussian mixture

- Case1. Same spherical covariance

■ 원래분포 vs 새로 구한 parameter와 확률로 클러스터링 한 분포




■ Clustering 정확도: 76.7%

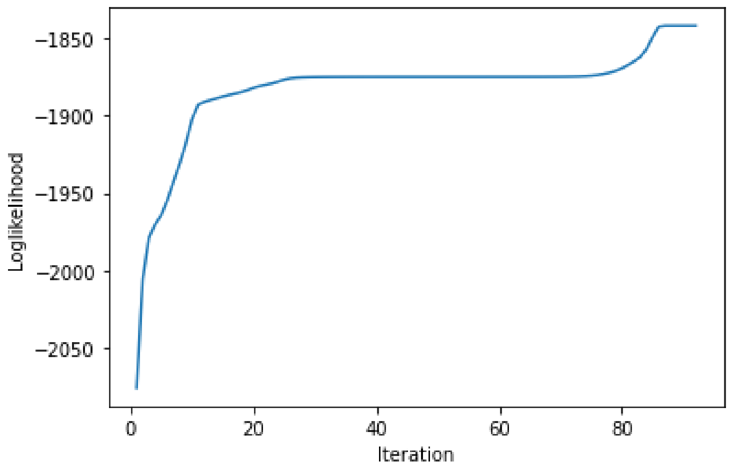
■ 원래 평균,분산,mixing parameter vs 새로 구한 평균,분산,mixing parameter

```
원래분포
mean
[[-20  20]
 [ 20  20]
 [ 20 -20]
 [ 60   1]
 [ 80  40]
 [  0 -40]
 [ 80 -40]]

coVariance
[[[50  0]
  [ 0 50]]

 [[50  0]
  [ 0 50]]

 [[50  0]
  [ 0 50]]

 [[50  0]
  [ 0 50]]

 [[50  0]
  [ 0 50]]

 [[50  0]
  [ 0 50]]

 [[50  0]
  [ 0 50]]]

alpha
[0.14 0.14 0.14 0.14 0.14 0.14 0.14]
```

```
After EM
mean
[[ -1.5 -39.6]
 [  5.1 -39.5]
 [-19.   19.5]
 [ 19.   19.8]
 [ 59.1   2.9]
 [ 79.2   0.2]
 [ 20.8 -20.4]]

coVariance
[[[  15.6   -19.2]
  [ -19.2    52.7]]

 [[  62.1    19.3]
  [  19.3    22.5]]

 [[  52.7     8.3]
  [   8.3    47.1]]

 [[  37.2     7.6]
  [   7.6    62.7]]

 [[  27.3    -8. ]
  [  -8.     35.2]]

 [[  42.7     8.9]
  [   8.9  1584.2]]

 [[  38.9    -5.5]
  [  -5.5    26.7]]]

alpha
[0.1 0.1 0.1 0.1 0.1 0.3 0.1]
```

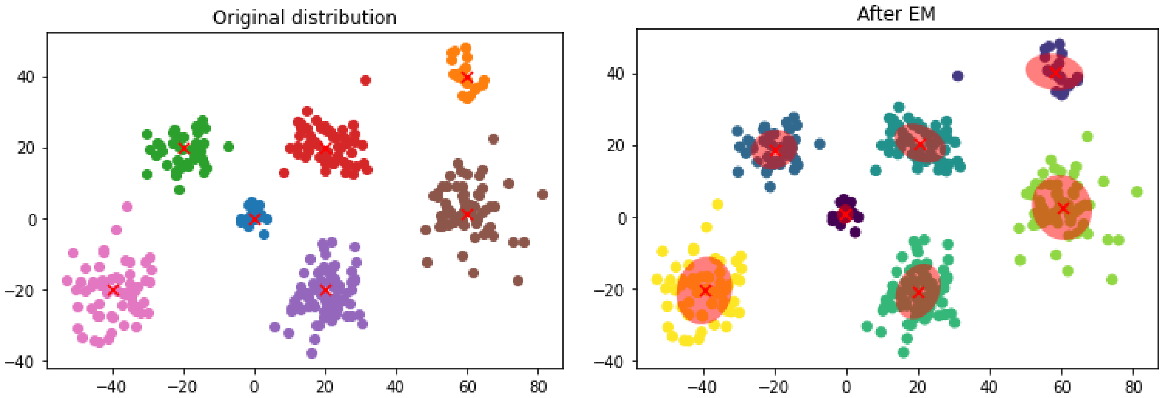
■ Loglikelihood 변화 (iteration 92번)

Maximized loglikelihood = -1842.365787



- Case2. Different spherical covariance

  - 원래분포 vs 새로 구한 parameter와 확률로 클러스터링 한 분포




  - Clustering 정확도: 99.7%

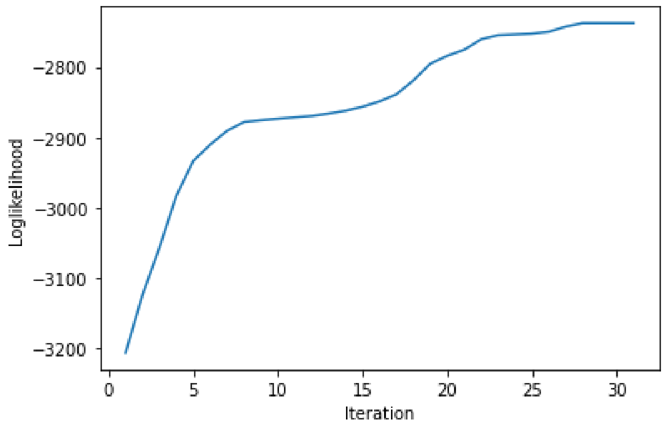
  - 원래 평균과 분산 vs 새로구한 평균과 분산

```
원래분포
mean
[[  0   0]
 [ 60  40]
 [-20  20]
 [ 20  20]
 [ 20 -20]
 [ 60   1]
 [-40 -20]]

coVariance
[[[ 5  0]
  [ 0  5]]

 [[10  0]
  [ 0 10]]

 [[20  0]
  [ 0 20]]

 [[30  0]
  [ 0 30]]

 [[40  0]
  [ 0 40]]

 [[50  0]
  [ 0 50]]

 [[60  0]
  [ 0 60]]]

alpha
[0.06 0.06 0.12 0.18 0.24 0.18 0.18]
```

```
After EM
mean
[[ 20.6  20.4]
 [ 58.   40.2]
 [-39.6 -20.7]
 [ 20.1 -20.9]
 [ 60.2   2.5]
 [-20.1  18.7]
 [ -0.3   0.6]]

coVariance
[[[33.  -7.6]
  [-7.6 19.1]]

 [[42.1 -3.7]
  [-3.7 16.6]]

 [[38.9  5. ]
  [ 5.  57. ]]

 [[25.8  7.7]
  [ 7.7 38.3]]

 [[45.6 -3.8]
  [-3.8 52.6]]

 [[26.4  1.9]
  [ 1.9 19.1]]

 [[ 3.3  0.1]
  [ 0.1  4.7]]]

alpha
[0.2 0.1 0.2 0.2 0.2 0.1 0.1]
```

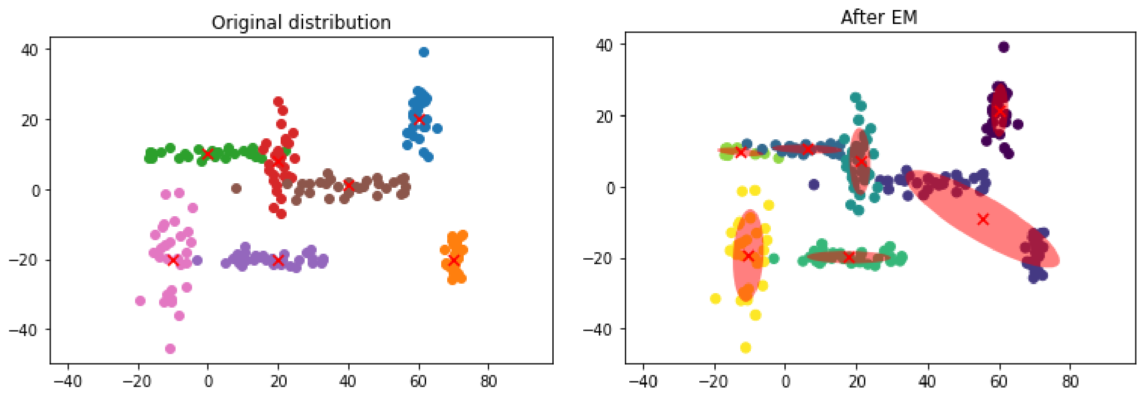
■ Loglikelihood 변화 (iteration 31번)

Maximized loglikelihood = -2737.723874




● Case3. Different diagonal covariance

■ 원래분포 vs 새로 구한 parameter와 확률로 클러스터링 한 분포




■ Clustering 정확도: 80%

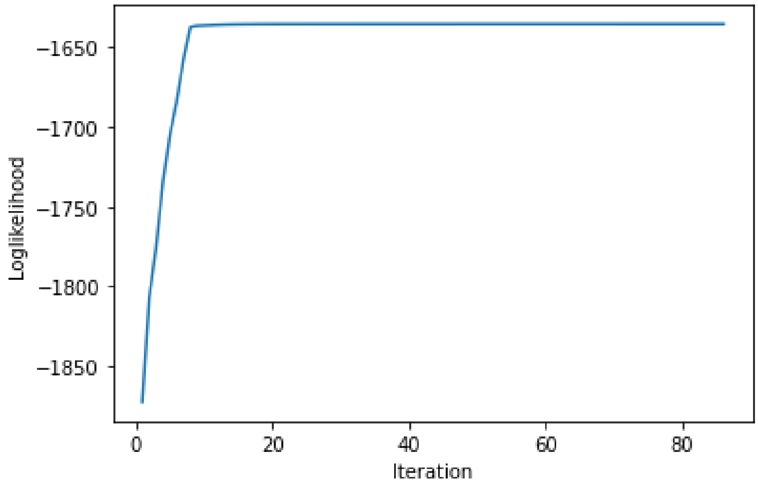
■ 원래 평균,분산,mixing parameter vs 새로 구한 평균,분산,mixing parameter

```
원래분포
mean
[[ 60  20]
 [ 70 -20]
 [  0  10]
 [ 20   8]
 [ 20 -20]
 [ 40   1]
 [-10 -20]]

coVariance
[[[  4   0]
  [  0  50]]

 [[  1   0]
  [  0  20]]

 [[100   0]
  [  0   1]]

 [[  5   0]
  [  0  80]]

 [[120   0]
  [  0   2]]

 [[100   0]
  [  0   5]]

 [[ 10   0]
  [  0  70]]]

alpha
[0.14 0.14 0.14 0.14 0.14 0.14 0.14]
```

```
After EM
mean
[[ 21.    7. ]
 [-10.4 -19.5]
 [ 55.6  -9.2]
 [-12.6   9.7]
 [ 60.3  21.4]
 [  6.2  10.5]
 [ 17.9 -19.9]]

coVariance
[[[   5.6   -0.9]
  [  -0.9   57.5]]

 [[  12.     5.3]
  [   5.3  108.9]]

 [[ 302.  -163.1]
  [-163.1  120.7]]

 [[  26.5   -2.2]
  [  -2.2    0.9]]

 [[   3.6    1.1]
  [   1.1   36.6]]

 [[  62.4   -1.9]
  [  -1.9    0.9]]

 [[  89.9   -1.2]
  [  -1.2    2. ]]]

alpha
[0.1 0.1 0.3 0.  0.1 0.1 0.1]
```

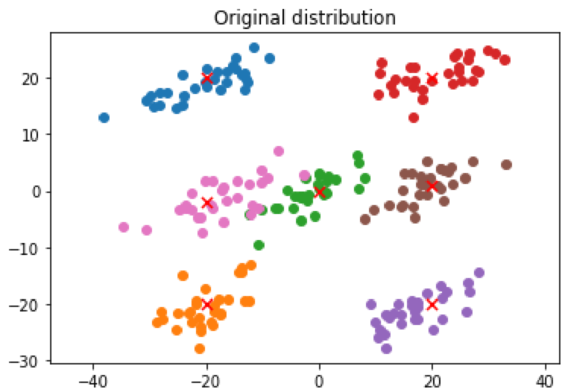
■ Loglikelihood 변화 (iteration 86번)

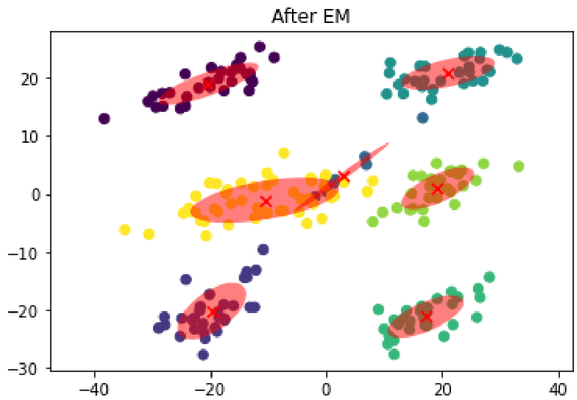
Maximized loglikelihood = -1635.832905




● Case4. Arbitrary same covariance

■ 원래분포 vs 새로 구한 parameter와 확률로 클러스터링 한 분포




■ Clustering 정확도: 88.1%

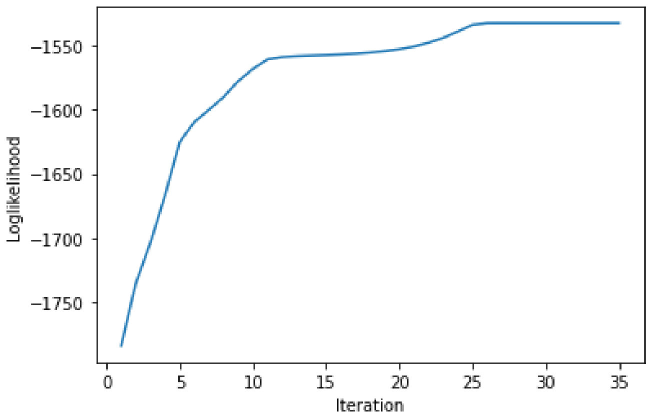
■ 원래 평균,분산,mixing parameter vs 새로 구한 평균,분산,mixing parameter

```
원래분포
mean
[[-20  20]
 [-20 -20]
 [  0   0]
 [ 20  20]
 [ 20 -20]
 [ 20   1]
 [-20  -2]]

coVariance
[[[10 10]
  [30 10]]

 [[10 10]
  [30 10]]

 [[10 10]
  [30 10]]

 [[10 10]
  [30 10]]

 [[10 10]
  [30 10]]

 [[10 10]
  [30 10]]

 [[10 10]
  [30 10]]]

alpha
[0.14 0.14 0.14 0.14 0.14 0.14 0.14]
```

```
After EM
mean
[[-19.7 -20.3]
 [ 21.1  20.9]
 [ 17.2 -21.2]
 [-10.6  -1.1]
 [-20.3  19. ]
 [  2.9   3. ]
 [ 19.3   0.9]]

coVariance
[[[ 22.5  11.1]
  [ 11.1  15.3]]

 [[ 42.2   8.8]
  [  8.8   5.6]]

 [[ 27.8  10.5]
  [ 10.5   8.7]]

 [[105.5  16.3]
  [ 16.3   9.7]]

 [[ 48.1  16.5]
  [ 16.5   8.5]]

 [[ 41.4  29.9]
  [ 29.9  21.9]]

 [[ 25.2  10.1]
  [ 10.1   8.5]]]

alpha
[0.1 0.1 0.1 0.3 0.1 0.  0.1]
```

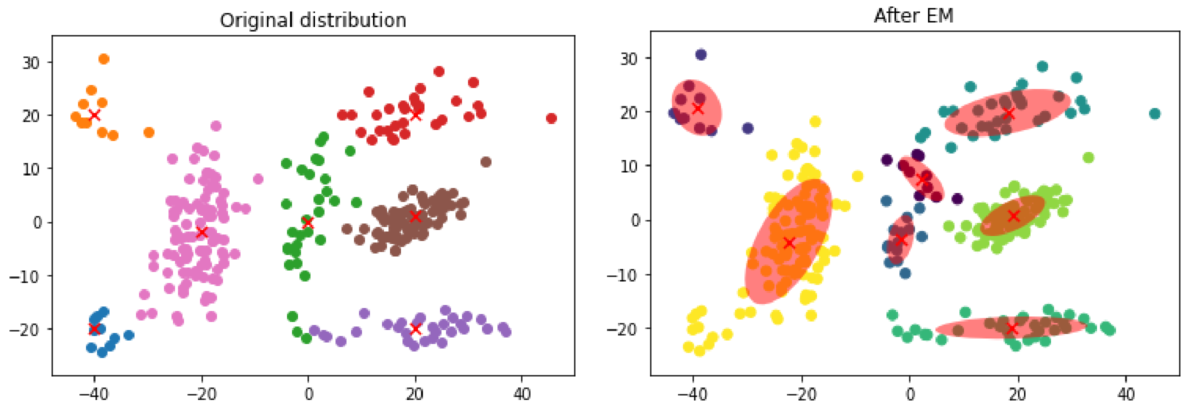
- Loglikelihood 변화 (iteration 35번)

Maximized loglikelihood = -1532.796690




- Case5. Arbitrary covariance

■ 원래분포 vs 새로 구한 parameter와 확률로 클러스터링 한 분포




■ Clustering 정확도: 91%

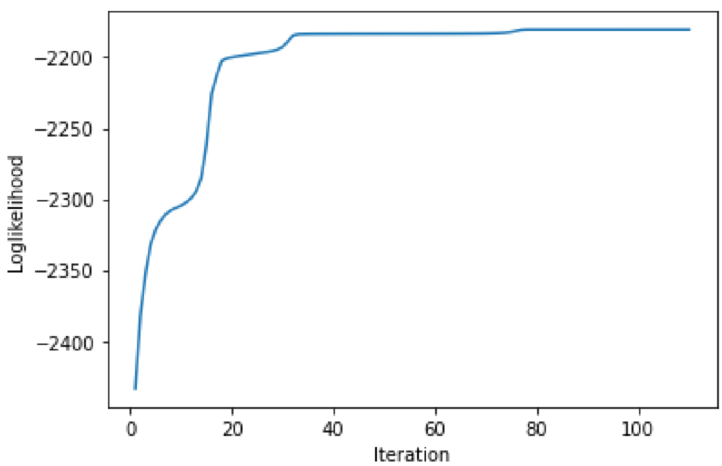
■ 원래 평균,분산,mixing parameter vs 새로 구한 평균,분산,mixing parameter

```
원래분포
mean
[[-40 -20]
 [-40  20]
 [  0   0]
 [ 20  20]
 [ 20 -20]
 [ 20   1]
 [-20  -2]]

coVariance
[[[  3   4]
  [-10  11]]

 [[ -1  19]
  [ 20   9]]

 [[ 16  10]
  [ 10 100]]

 [[ 50   7]
  [  1   7]]

 [[ 10   4]
  [ 70   6]]

 [[ 10  10]
  [ 30  10]]

 [[ 10  70]
  [ 20  26]]]

alpha
[0.03 0.03 0.1  0.1  0.1  0.28 0.34]
```

```
After EM
mean
[[-22.3  -4.1]
 [ -1.6  -3.7]
 [ 18.9 -20. ]
 [ 18.2  19.7]
 [  2.3   7.5]
 [ 19.1   0.7]
 [-39.4  20.7]]

coVariance
[[[ 41.7  36.5]
  [ 36.5  86.1]]

 [[  4.1   2.6]
  [  2.6  13.3]]

 [[126.    3.4]
  [  3.4   2.7]]

 [[ 87.4  16. ]
  [ 16.   12.6]]

 [[ 11.2  -7.9]
  [ -7.9  11.2]]

 [[ 23.3   9.2]
  [  9.2   9.1]]

 [[ 13.8  -3.5]
  [ -3.5  17.5]]]

alpha
[0.4 0.  0.1 0.1 0.  0.3 0. ]
```

- Loglikelihood 변화 (iteration 110번)

Maximized loglikelihood = -2180.551871



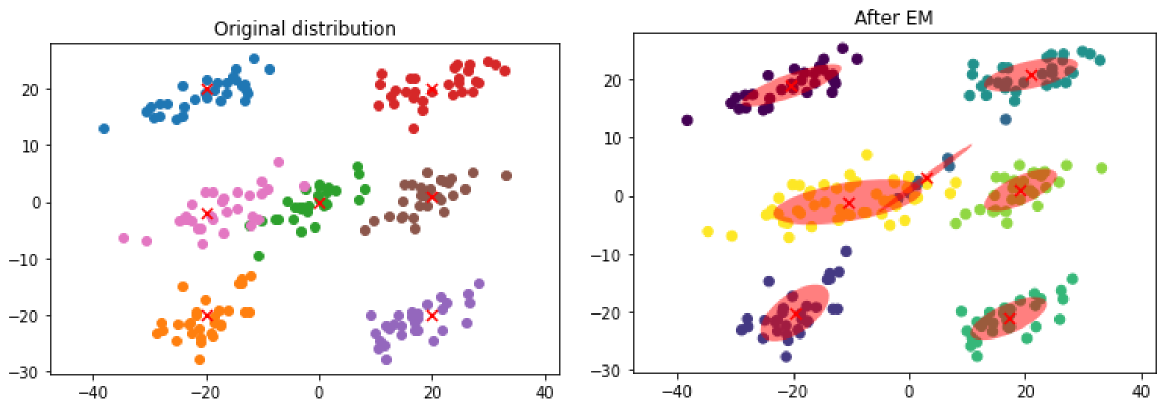
## 4. Discussion

**4.1** Result 부분에 명시한 부분 말고도 다양한 초깃값, 다양한 분포 수에 대해서도 실험을 해보았다. Mixing parameter, 평균,분산 , 즉 몇 개의 파라미터 만으로 클러스터를 특정지을 수 있다는 장점을 잘 확인했다. 가우시안 분포끼리 서로 겹치지 않을 때 클러스터링 정확도가 100%에 가깝게 매우 높았다.

**4.2** 하지만, EM 알고리즘은 Likelihood의 global maximization을 찾는게 아니라 local한 maximization을 하는 것이라서 초기값에 따라서 지역해에 빠질 수가 있다는 것을 확인했다. 따라서 초기값에 따라서 클러스터링을 잘 하지 못하는 경우을 확인했고, 그런 분포들은 다른 초기값을 주고 iteration을 돌리면 클러스터링이 잘 되고 loglikelihood도 더 크게 나왔다.

**4.3** 또한 데이터 분포가 서로 겹치게 생기는 부분이 있으면 밑의 그림과 같이 클러스터링을 일부 잘 못하는 경우가 발생한다.

- Ex) 원래 분포에서 분홍색, 초록색은 다른 가우시안 분포로부터 나온 데이터인데, EM알고리즘을 적용한 노란색 부분처럼 하나의 가우시안 분포로 클러스터링된다.
- 초기값을 적절하게 주고 다시 iteration을 돌리면 클러스터링이 잘 된다.




**4.4** 분포가 여러 개일수록 , 계산량(소요시간)이 많다는 것을 확인했다. (computationally expensive)

**4.5** 나는 원래 분포가 몇 개의 (클러스터)가우시안 분포로 이루어져 있는지를 알려주고 클러스터링을 시켰지만, ex) "이렇게 생긴 분포를 7 개의 가우시안 분포로 클러스터링해 ! " 원래 분포의 클러스터 개수를 미리 알지 못 한다면 육안으로 클러스터 개수를 추정하기 힘들다는 단점이 있다고 느꼈다.

## 5. Appendix

코드 첨부는 ipynb 파일로 따로 첨부하겠습니다. 코드에 주석처리로 부가 설명을 해 놓았습니다.

## 6. Reference

[1]위키피디아
https://ko.wikipedia.org/wiki/%EA%B8%B0%EB%8C%93%EA%B0%92_%EC%B5%9C%EB%8C%80%ED%99%94_%EC%95%8C%EA%B3%A0%EB%A6%AC%EC%A6%98