PRML 9 Mixture Models and EM

Takayuki Takaai

ISIR

August 4, 2017

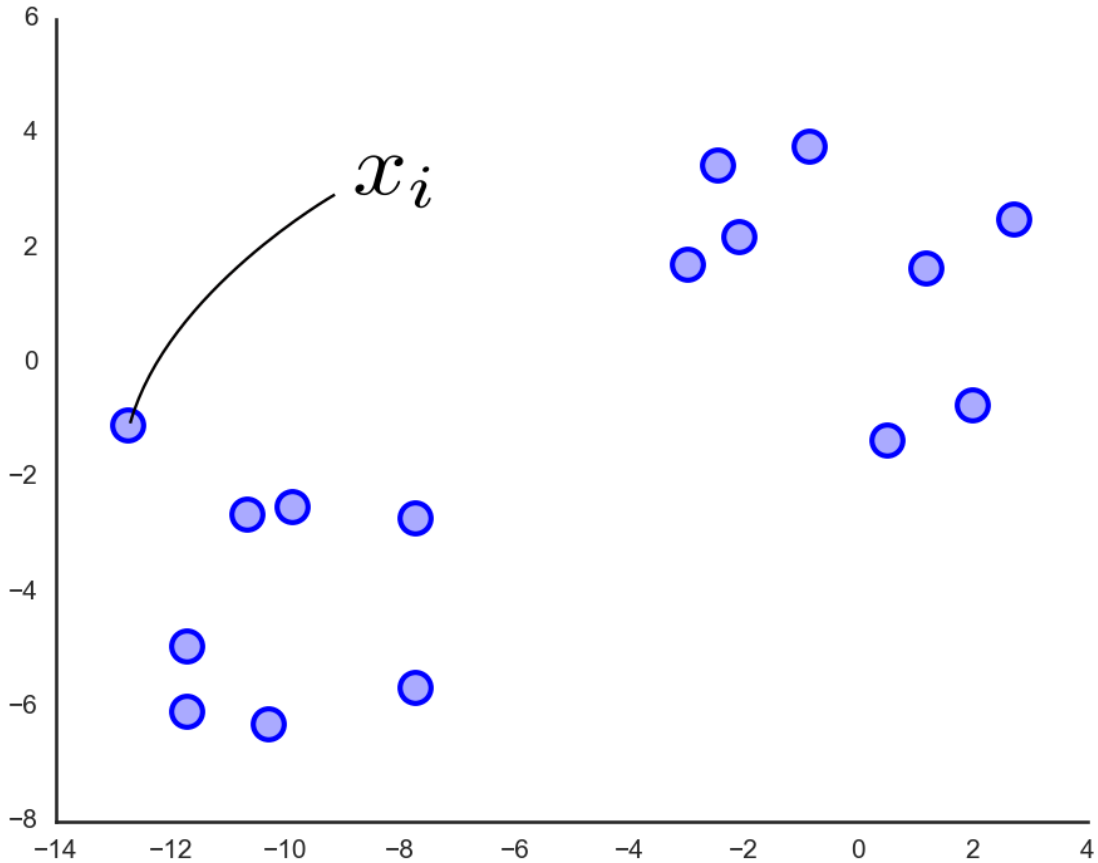
Table of Contents

K-means 法の特徴

- クラスタの『平均（means)』を用い、あらかじめ決められたクラスタ数『k』個に分類する
- 初期値（初期に選択される「核」）はランダムに選ぶ

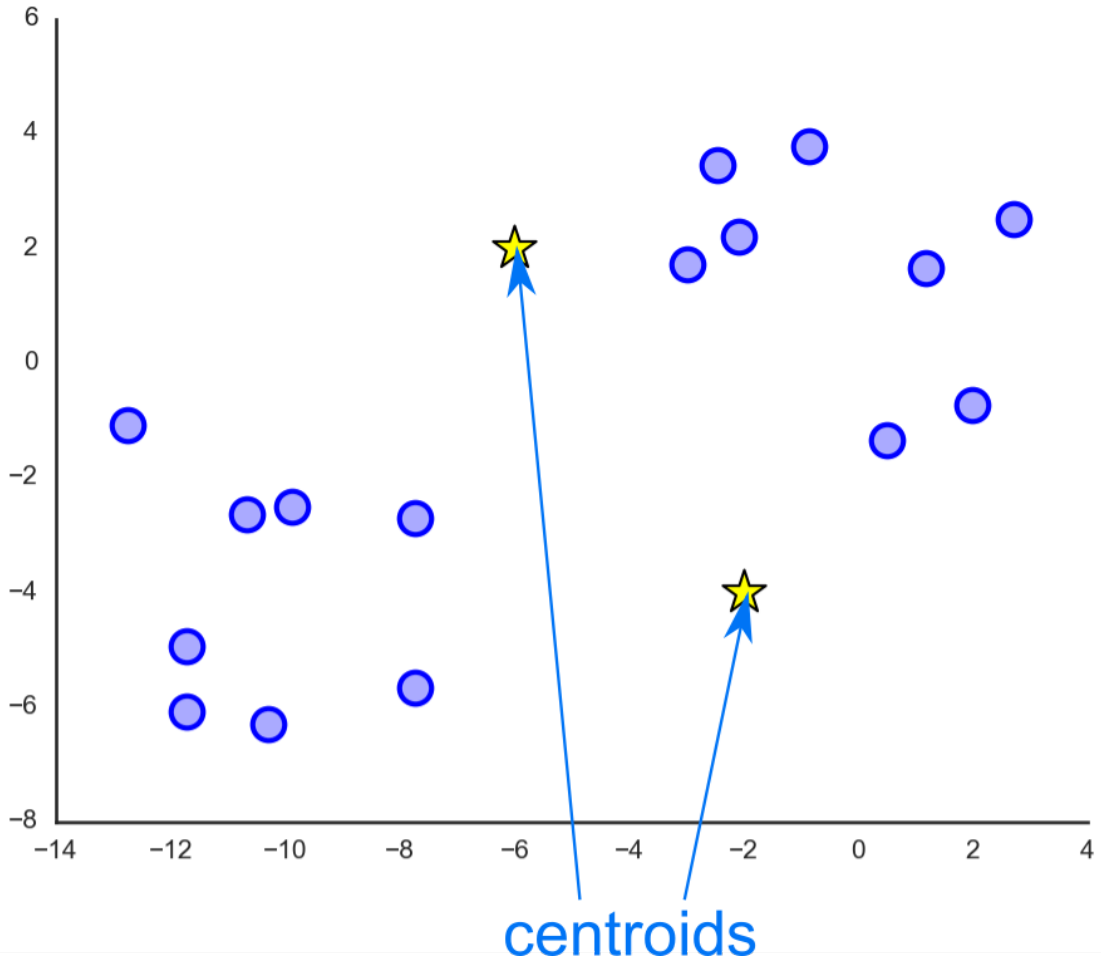
K-means 法のアルゴリズム (1)

データ $X = \{x_1, \cdots, x_N\}$



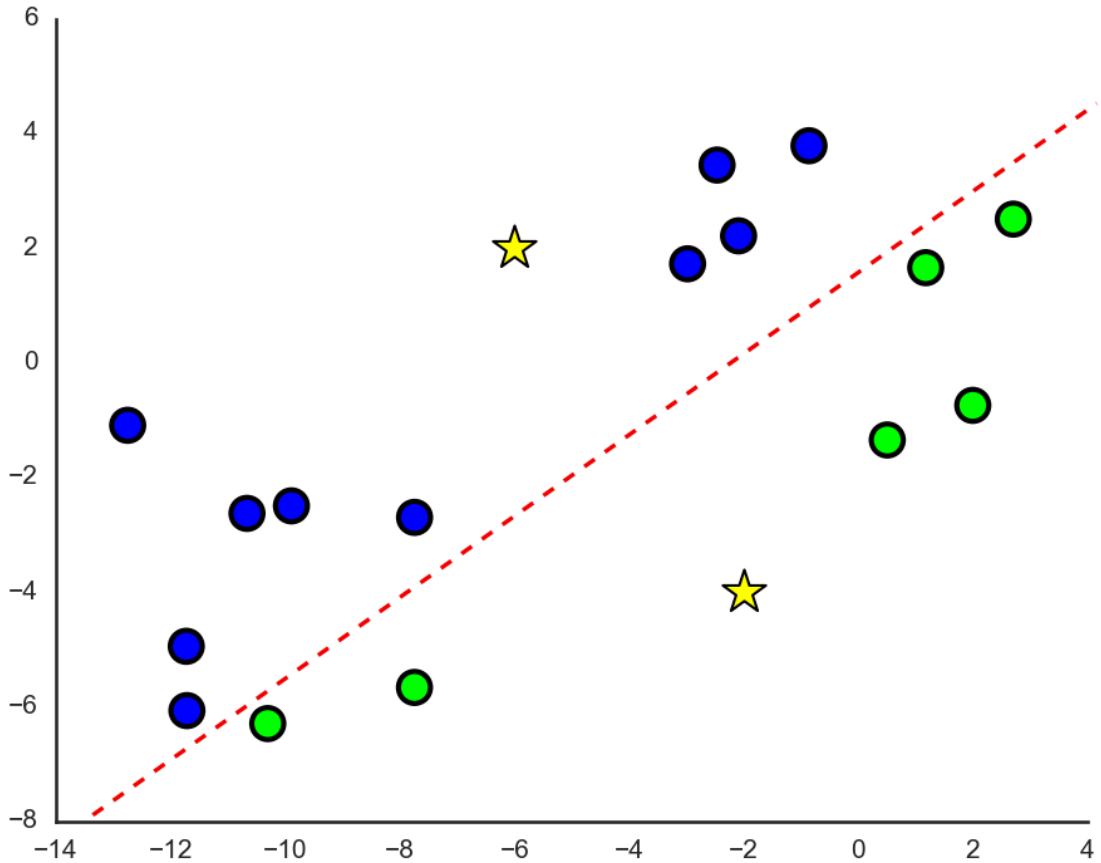
K-means 法のアルゴリズム (2)

任意の k 個の初期クラスタ中心 (セントロイド) を選ぶ



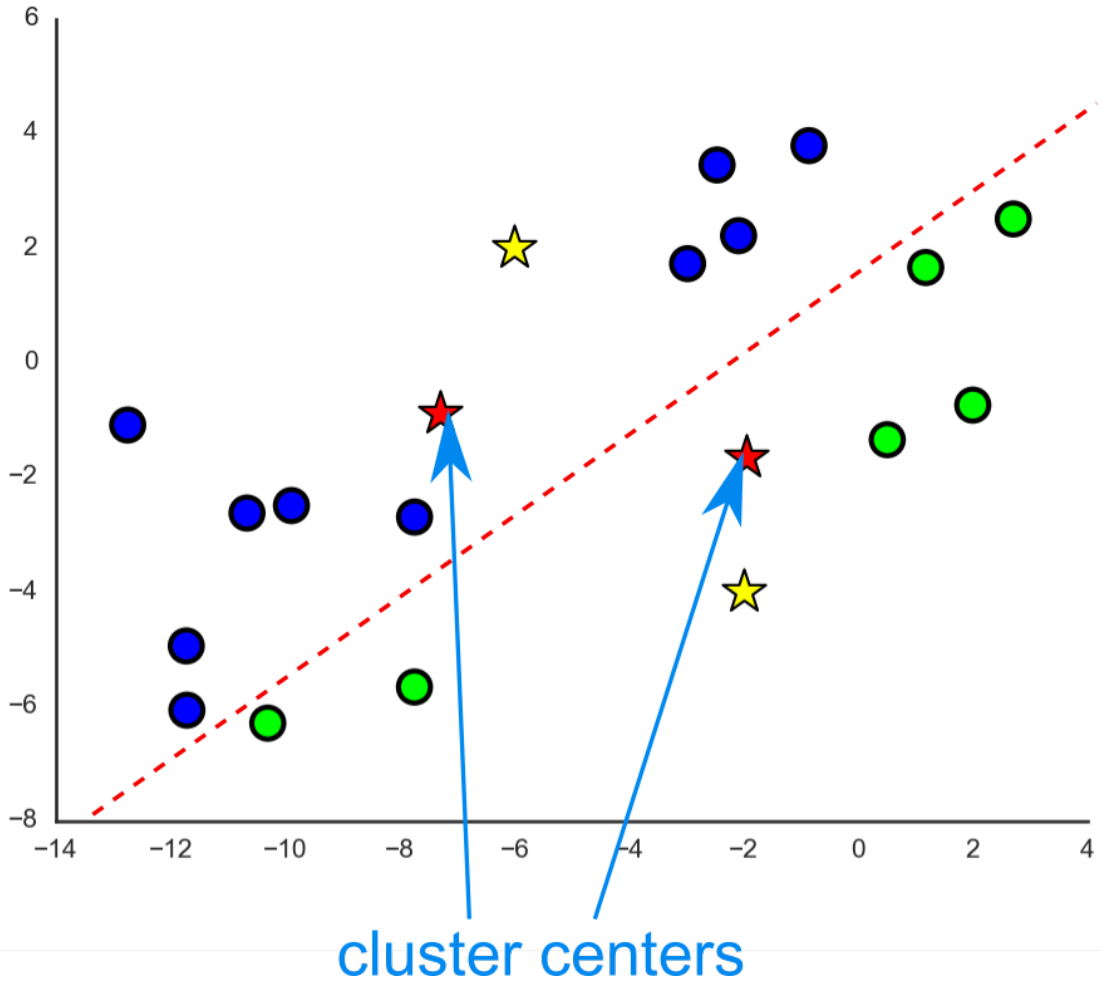
K-means 法のアルゴリズム (3)

各データを一番近いセントロイドに属させる



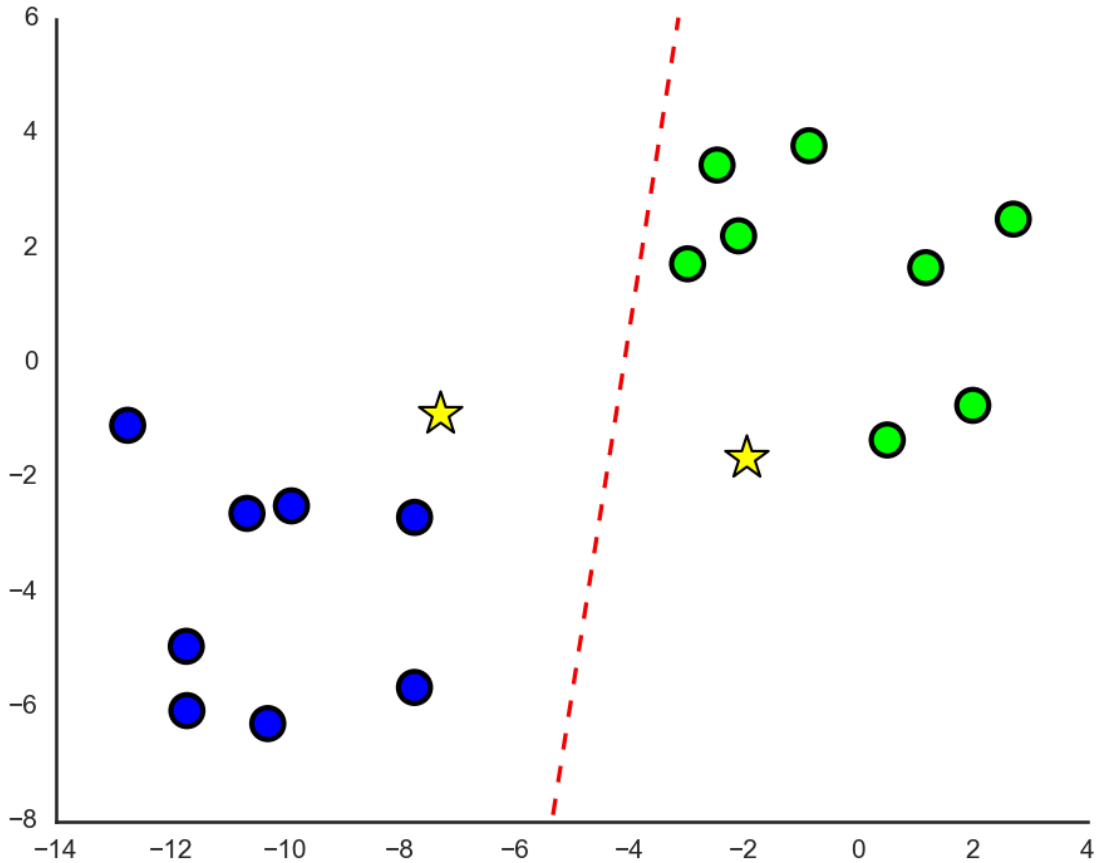
K-means 法のアルゴリズム (4)

各セントロイドの重心を新しいセントロイドとする



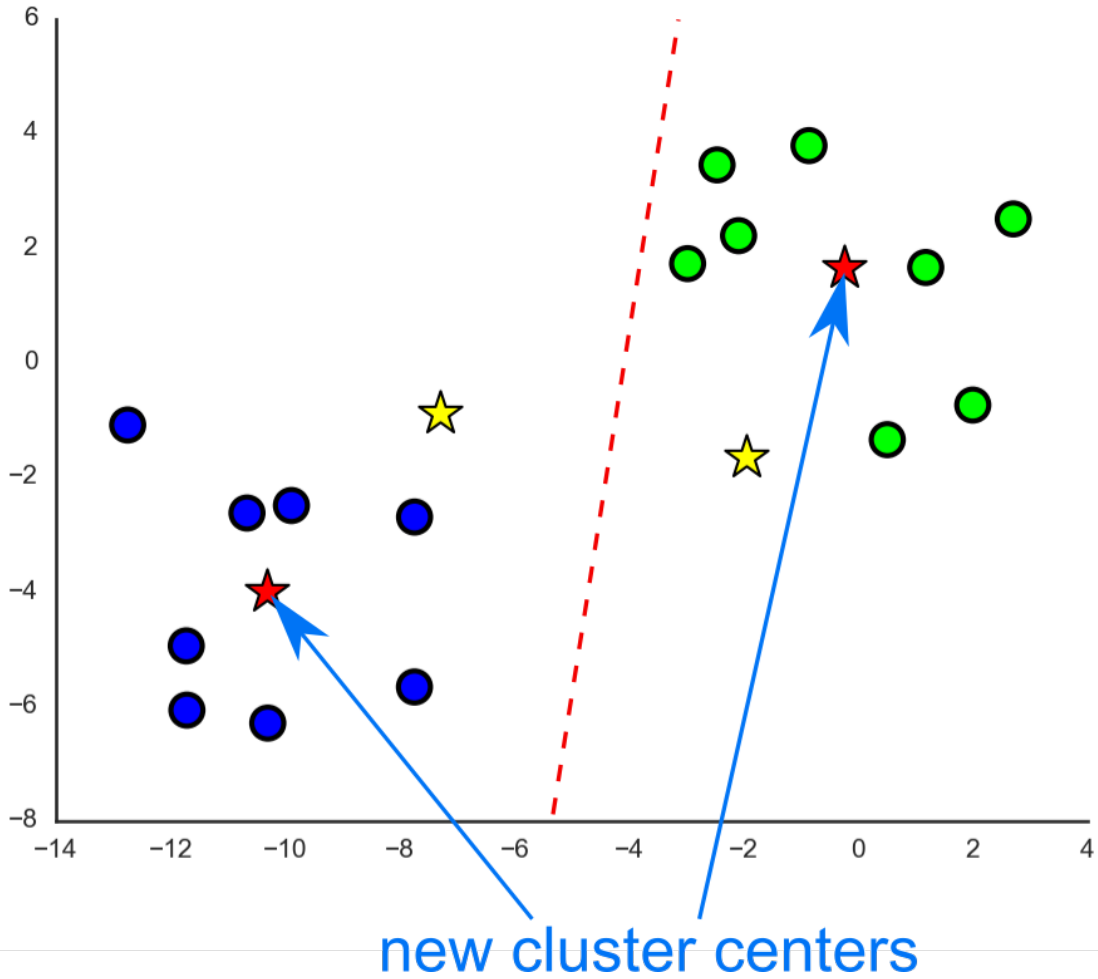
K-means 法のアルゴリズム (5)

新しいセントロイドでクラスタを更新する



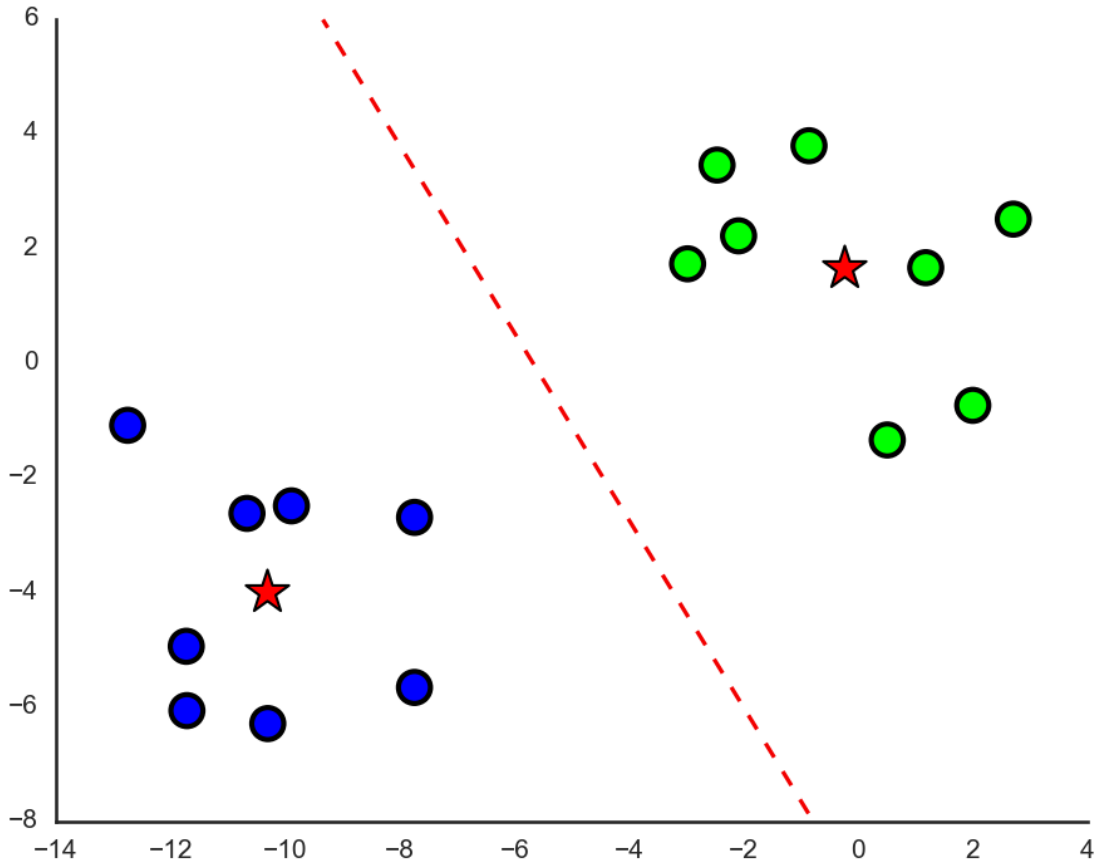
K-means 法のアルゴリズム (6)

更新したクラスタの重心を新しいセントロイドとする



K-means 法のアルゴリズム (7)

収束するまで繰り返す



K-means cost function

- Input : データセット $X \subseteq \mathbb{R}^d$, クラスタ数 k
- Output : セントロイドセット $Z \subseteq \mathbb{R}^d$, $|Z| = k$

Cost function

$$\mathrm{cost}(Z) \stackrel{def}{=} \sum_{x \in X} \min_{z \in Z} ||x - z||^2$$

一番近いセントロイドとの距離

- Goal : Cost function を最小にする set Z を見つける

K-means Algorithm

data set $X \subseteq \mathbb{R}^d$, integer k

K-means Algorithm

1: $z_1, \cdots, z_k \in \mathbb{R} \leftarrow randomly$
2: **while** Cost function still improves
3: $\quad S_1, \cdots, S_k \leftarrow \phi$
4: $\quad$ **for** $x \in X$
5: $\quad\quad$ **for** $i \in \{1, \cdots, k\}$
6: $\quad\quad\quad j \leftarrow \arg\min_i ||x - z_i||^2$
7: $\quad\quad\quad$ add x to S_j
8: $\quad$ **for** $j \in \{1, \cdots, k\}$
9: $\quad\quad z_j = \frac{1}{|S_j|} \sum_{x \in S_j} x$

K-means 法の収束性 (1)

センタロイド z のクラスタを

$$C_z = \{x \in S : x \text{ から一番近いのが } z\}$$

と定義すると，Cost function は

$$\mathrm{cost}(Z) = \sum_{z \in Z} \sum_{x \in C_z} ||x - z||^2$$

のように書くことができて，これをクラスタ $C_{z_1} \cdots C_{z_k}$ とセントロイド $z_1 \cdots z_k$ の関数とみると

$$\mathrm{cost}(C_{z_1} \cdots C_{z_k}; z_1 \cdots z_k) = \sum_{z_i \in Z} \sum_{x \in C_{z_i}} ||x - z_i||^2$$

と書ける.

K-means 法の収束性 (2)

補題 1

1 クラスタの Cost function を

$$\mathrm{cost}(C, z) = \sum_{x \in C} ||x - z||^2$$

とすると，$\forall C \subset \mathbb{R}^d, \forall z \in \mathbb{R}^d$ に対し，

$$\mathrm{cost}(C, z) = \mathrm{cost}(C, \mathrm{mean}(C)) + |C| \cdot ||z - \mathrm{mean}(C)||.$$

$\mathrm{cost}(C, z)$ が最小値をとるのは $z = \mathrm{mean}(C)$ のとき.

K-means 法の収束性 (3)

補題 2

Cost function

$$\mathrm{cost}(C, z) = \sum_{x \in C} ||x - z||^2$$

は，K-means アルゴリズムの一連の繰り返し操作において単調減少.

[証明] t 回目の反復におけるセントロイドとクラスタを $z^{(t)}, C^{(t)}$ とする.

1. 各データを一番近いセントロイドに割り当てる操作では,

$$\mathrm{cost}(C^{(t+1)}, z^{(t)}) \leq \mathrm{cost}(C^{(t)}, z^{(t)}).$$

2. セントロイドの更新操作では，補題 1 によって,

$$\mathrm{cost}(C^{(t+1)}, z^{(t+1)}) \leq \mathrm{cost}(C^{(t)}, z^{(t)}).$$

K-means 法の収束性 (4)

定理 1

K-means の Cost function は K-means アルゴリズムの一連の繰り返し操作に対して収束する.

[証明] Cost function

$$\mathrm{cost}(C_{z_1} \cdots C_{z_k}; z_1 \cdots z_k) = \sum_{z_i \in Z} \sum_{x \in C_{z_i}} ||x - z_i||^2 = \sum_{z_i \in Z} \mathrm{cost}(C_{z_i}, z_i)$$

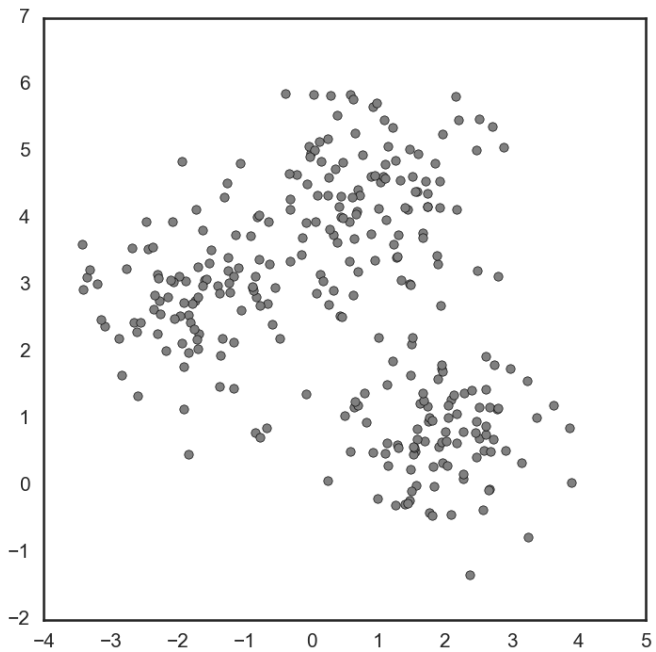
は一連の操作によって単調減少 (補題 2).
また, $\mathrm{cost}(C_{z_1} \cdots C_{z_k}; z_1 \cdots z_k)$ は定義により下に有界.
したがって $\mathrm{cost}(C_{z_1} \cdots C_{z_k}; z_1 \cdots z_k)$ は収束する.

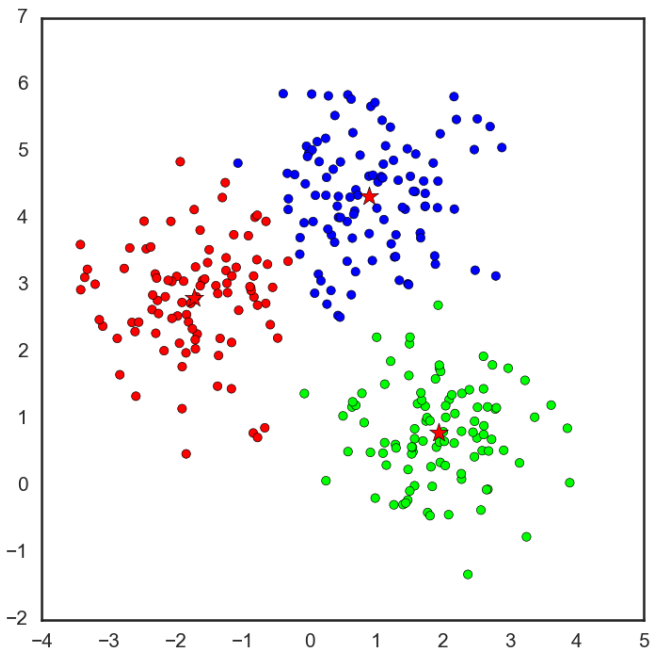
注意

収束先は極小値であり, 最小値とは限らない. (初期値依存性)

K-means 法の実装 -染みクラスタ-

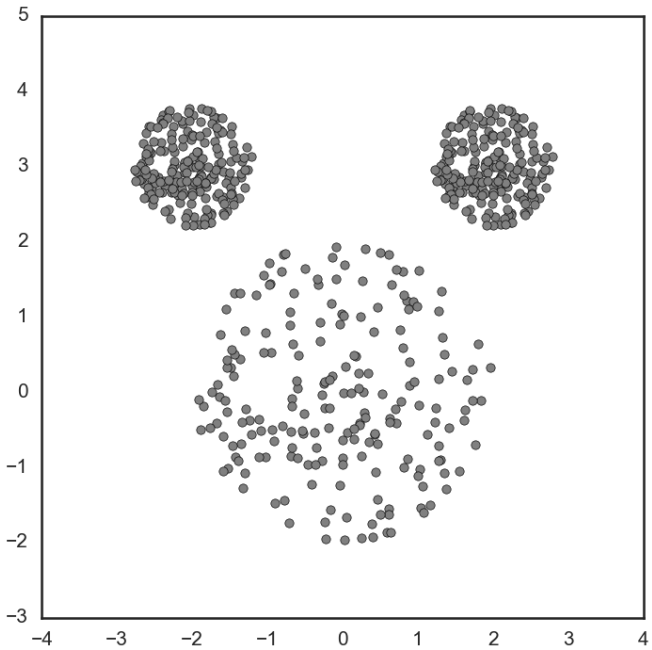
クラスタ内のデータ数が均等な場合

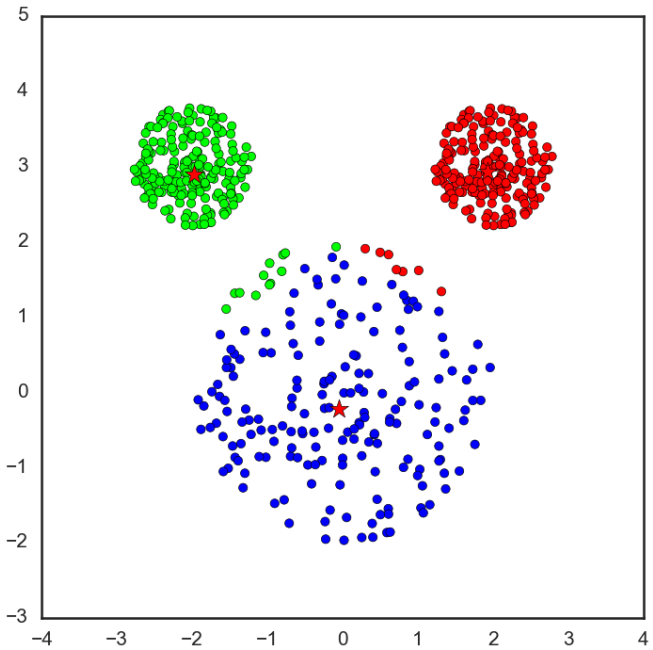




K-means 法の実装 -Mickey 1-

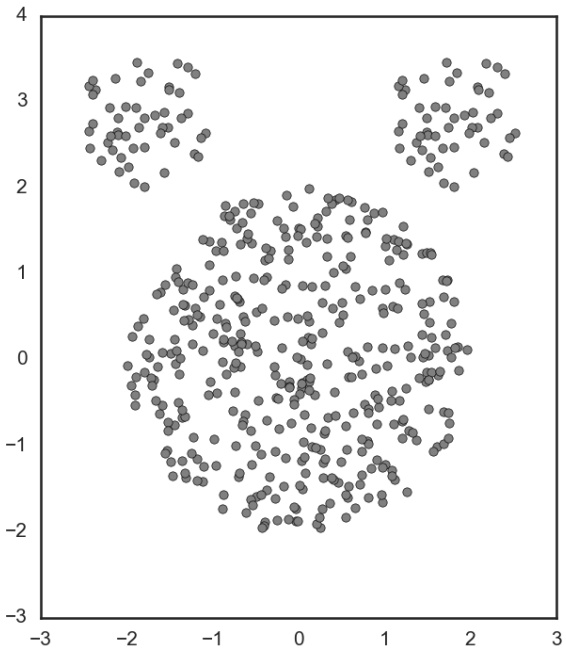
クラスタ内のデータ数が均等な場合

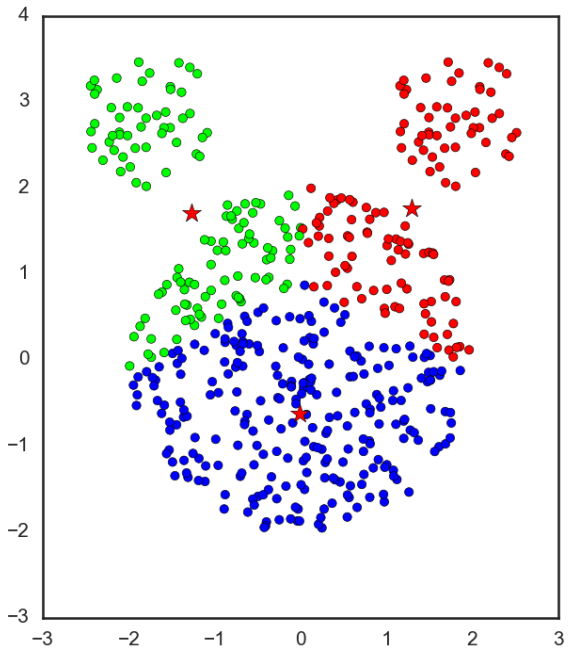




K-means 法の実装 -Mickey 2-

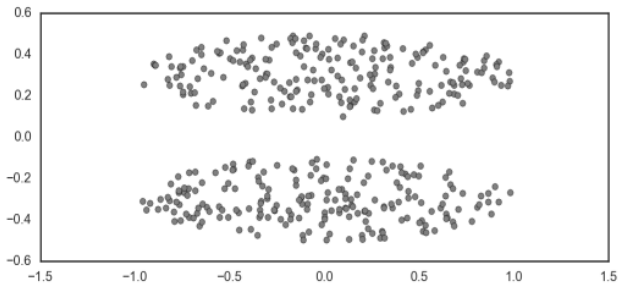
クラスタ内のデータ数が均等でない場合
データ数の多い所にセントロイドが引き寄せられる

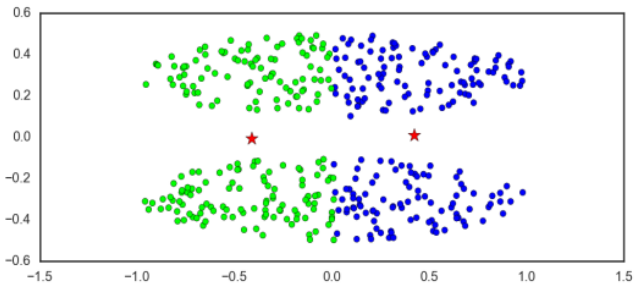


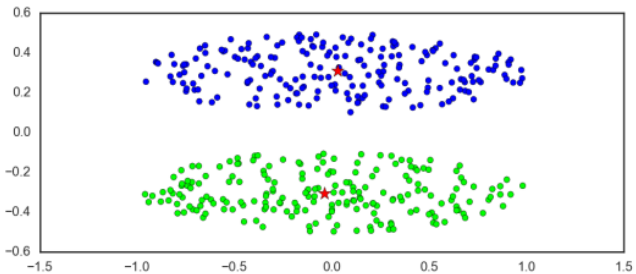


K-means 法の実装 -初期値依存性-

初期セントロイドによってクラスタリング結果が異なる

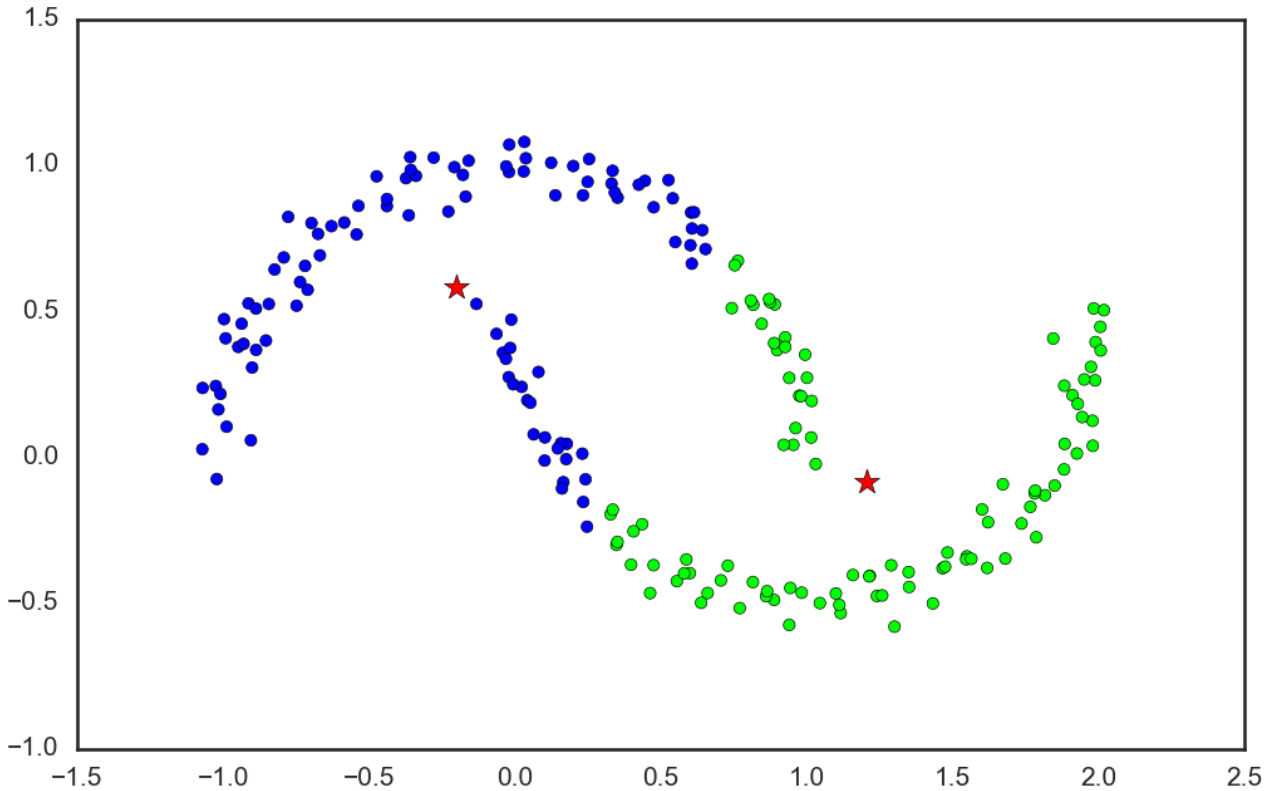






K-means 法の実装 -円形でないクラスタ-

円形でないものは苦手



K-means 法の実装 -画像の減色-

元の画像

$k = 2$

$k = 3$

$k = 8$

K-means 法の実装 -画像の減色-

初期セントロイドを変えると結果が変ってしまう

$k = 4$

$k = 4$

K-means 法の欠点

1. **初期値依存性**が大きい
2. 最悪計算時間は入力サイズに対して超多項式 (**NP 困難**)
3. ノイズや外れ値の影響を受けやすい
4. クラスタは超球状の形状ということを暗黙のうちに仮定している
5. クラスタ中の対象数はどれも等しいということを暗黙のうちに仮定している
6. クラスタ個数を事前に人間が指定する必要がある

- 1 に対する改善法が K-means++ (2 や3 にもある程度対応)
- 6 に対する改善法は決定的なものはなさそう．(最適なクラスター数の目安を得るにはエルボー図，シルエット法，x-means がある)

K-means++ algorithm

- k-means の初期値の選び方にひと工夫を加えたもの
- **初期の k 個のクラスタ中心はなるべく離れている方が良い**
- 手順
  1. データセット X からランダムに最初のクラスタ中心 z_1 を一つ選ぶ.
  2. それぞれのデータ点 x に対して最近傍中心との距離 $D(x)$ を計算する.
  3. 重みつき確率分布 $\frac{D(x)^2}{\sum_{x \in X} D(x)^2}$ を用いて，データ点 x の中から新しいクラスタ中心をランダムに選ぶ．(クラスタ中心から遠いほど確率が高くなる)
  4. 2 , 3 をクラスタ数が k に達するまで繰り返す．(但しクラスタ中心は重複させない)
  5. 選ばれたクラスタ中心を初期値として k-means を行う.

K-means++ -画像の減色-

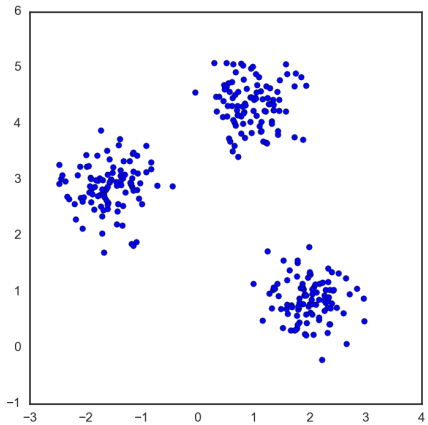
k-means++の場合，初期セントロイドを変えても結果は変らない

k-means

k-means++

混合ガウス分布

- **目的** :データをクラスタに" 分類する" 代わりに，データ x に対し，各クラスタ c に属する**確率** $P(c|x)$ を与える.
- 決定すべき事柄：
  1. クラスタごとの**確率モデル**
  2. クラスタごとの**パラメータ**
  3. **クラスタ数**
  4. **各クラスタの重み**
- 混合ガウスモデル
  1. 1 はすべて ガウス分布 ,
     3 は クラスタ数 K を与える
  2. 2 は 平均 μ_k , 共分散行列 Σ_k ,
     4 は 重み π_k $(k = 1, 2, \cdots K)$ を推定する



混合ガウスモデル Gaussian Mixture Models (GMM)

定義

混合ガウスモデルの確率分布:

$$
\begin{aligned}
p(x) &= \sum_{k=1}^{K} \pi_k p_k(x) \\
p_k(x) &= \mathcal{N}(x|\mu_k, \Sigma_k) \\
&= \frac{1}{(2\pi)^{\frac{d}{2}} |\Sigma_k|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(x_n - \mu_k)^T \Sigma_k^{-1} (x_n - \mu_k) \right)
\end{aligned}
$$

$$
\sum_{k=1}^{K} \pi_k = 1, \ \pi_k \leq 0 \ (k = 1, 2, \cdots K)
$$

単一ガウス分布 (1 次元) の最尤推定

- $x_1, x_2, \cdots, x_N$ が独立で，それぞれ平均が μ，分散が σ^2 である正規分布に従うとすると，尤度関数 (確率密度) は

$$L(\mu, \sigma^2) = \prod_{k=1}^{N} \left\{ \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x_k - \mu)^2}{2\sigma^2} \right) \right\}$$

- 対数尤度関数は

$$\log L(\mu, \sigma^2) = -\frac{n}{2} \log(2\pi\sigma^2) - \frac{1}{2\sigma^2} \sum_{k=1}^{N} (x_k - \mu)^2.$$

- 最大値をとる必要条件は

$$\begin{cases} \dfrac{\partial}{\partial \mu} \log L(\mu, \sigma^2) = \dfrac{1}{\sigma^2} \displaystyle\sum_{k=1}^{N} (x_k - \mu) = 0 \\ \dfrac{\partial}{\partial \sigma^2} \log L(\mu, \sigma^2) = -\dfrac{n}{2\sigma^2} + \dfrac{1}{2\sigma^4} \displaystyle\sum_{k=1}^{N} (x_k - \mu)^2 = 0 \end{cases}$$

単一ガウス分布 (1 次元) の最尤推定

- 最大値をとる必要条件はすなわち

$$\begin{cases} \mu = \dfrac{x_1 + x_2 + \cdots + x_n}{n} \\ \sigma^2 = \dfrac{1}{n} \displaystyle\sum_{k=1}^{n} (x_k - \mu)^2 \end{cases}$$

- ガウス分布 1 つの場合は (多次元でも) 簡単に解析的に解ける.
- 尤度関数の **exp に log が直接かかる**から

混合ガウス分布の最尤推定

- データ $X = \{x_1, \cdots, x_N\}$ に対する尤度関数は,

$$L(\pi, \mu, \Sigma) = \prod_{n=1}^{N} p(x_k)$$
$$= \prod_{n=1}^{N} \left\{ \sum_{k=1}^{K} \pi_k \mathcal{N}(x_n | \mu_k, \Sigma_k) \right\}$$

- 対数尤度関数は,

$$J(\pi, \mu, \Sigma) = \log L(\pi, \mu, \Sigma) = \sum_{n=1}^{N} \log \left\{ \sum_{k=1}^{K} \pi_k \mathcal{N}(x_n | \mu_k, \Sigma_k) \right\}$$

log の中に和があるので, 扱いが困難

混合ガウス分布の最尤推定

- 対数尤度関数

$$J(\pi, \mu, \Sigma) = \sum_{n=1}^{N} \ln \left\{ \sum_{k=1}^{K} \pi_k \mathcal{N}(x_n | \mu_k, \Sigma_k) \right\}$$

$$= \sum_{n=1}^{N} \ln \left\{ \sum_{k=1}^{K} \pi_k \frac{1}{(2\pi)^{\frac{d}{2}} |\Sigma_k|^{\frac{1}{2}}} \exp \left(-\frac{1}{2} (x_n - \mu_k)^T \Sigma_k^{-1} (x_n - \mu_k) \right) \right\}$$

- 最尤推定解の必要条件：

$$\frac{\partial J}{\partial \mu_k} = 0, \quad \frac{\partial J}{\partial \Sigma_k} = 0$$

π_k については，束縛条件：$\pi_k \leq 1, \sum_{i=1}^{K} \pi_k = 1$ を考慮してラグランジュの未定係数法を使う.

最尤推定解の必要条件 (その 1)：$\dfrac{\partial J}{\partial \mu_k} = 0$

$$\frac{\partial J}{\partial \mu_k} = \sum_{n=1}^{N} \frac{\pi_k \mathcal{N}(x_n|\mu_k, \Sigma_k)}{\sum_{i=1}^{K} \pi_i \mathcal{N}(x_n|\mu_i, \Sigma_i)} \Sigma_k^{-1}(x_n - \mu_k) = 0$$

γ_n^k とおく

両辺の左から Σ_k をかけて整理すると

$$\mu_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma_n^k x_n$$

$$N_k = \sum_{n=1}^{N} \gamma_n^k$$

k 番目のクラスターに割り当てられる点の数

最尤推定解の必要条件 (その 2)：$\dfrac{\partial J}{\partial \Sigma_k} = 0$

$$\frac{\partial J}{\partial \Sigma_k} = 0 \Longleftrightarrow \Sigma_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma_n^k (x_n - \mu_k)(x_n - \mu_k)^T$$

最尤推定解の必要条件 (その 3)：π_k

$$L = J + \lambda \left(\sum_{k=1}^{K} \pi_k - 1 \right)$$

とおくと,

$$\frac{\partial L}{\partial \pi_k} = \sum_{n=1}^{N} \frac{\mathcal{N}(x_n|\mu_k, \Sigma_k)}{\sum_{i=1}^{K} \pi_k \mathcal{N}(x_n|\mu_i, \Sigma_i)} + \lambda = \sum_{n=1}^{N} \frac{\gamma_n^k}{\pi_k} + \lambda = \frac{N_k}{\pi_k} + \lambda = 0$$

$N_k = -\lambda \pi_k$ の両辺を 1 から N まで加えると $N = -\lambda$ なので

$$\pi_k = \frac{N_k}{-\lambda} = \frac{N_k}{N}$$

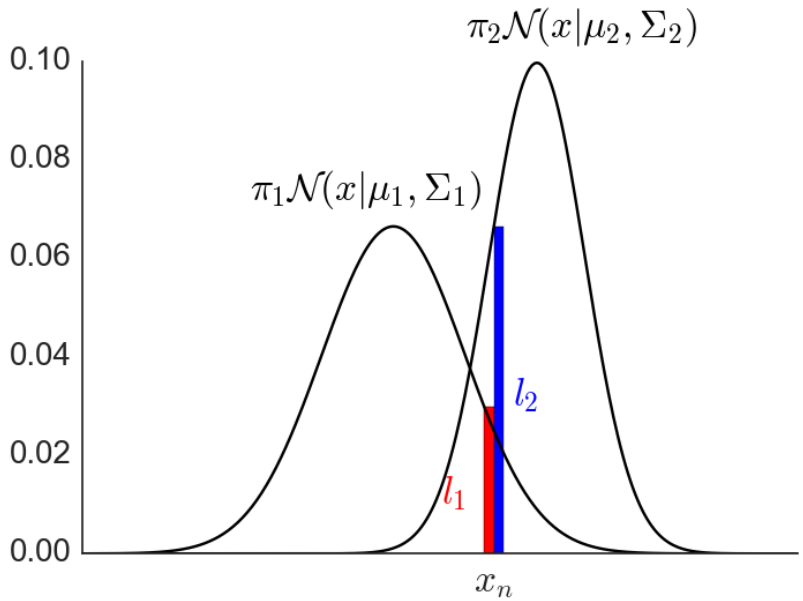
負担率

$$\gamma_n^k = \frac{\pi_k \mathcal{N}(x_n|\mu_k, \Sigma_k)}{\sum_{i=1}^{K} \pi_i \mathcal{N}(x_n|\mu_i, \Sigma_i)}$$ って何？

データ x_n がクラスタ k から発生した可能性

$$\gamma_n^1 = \frac{l_1}{l_1 + l_2}$$

$$\gamma_n^2 = \frac{l_2}{l_1 + l_2}$$



混合ガウス分布の最尤推定

最尤推定解の必要条件 (まとめ):

$$\mu_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma_n^k x_n$$

$$\Sigma_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma_n^k (x_n - \mu_k)(x_n - \mu_k)^T$$

$$\pi_k = \frac{N_k}{N}$$

$$\gamma_n^k = \frac{\pi_k \mathcal{N}(x_n | \mu_k, \Sigma_k)}{\sum_{i=1}^{K} \pi_i \mathcal{N}(x_n | \mu_i, \Sigma_i)}, \quad N_k = \sum_{n=1}^{N} \gamma_n^k$$

解析的には解けないので一工夫する

EM アルゴリズム

負担率はパラメータの値が決まれば求まる

$$\gamma_n^k = \frac{\pi_k \mathcal{N}(x_n | \mu_k, \Sigma_k)}{\sum_{i=1}^{K} \pi_i \mathcal{N}(x_n | \mu_i, \Sigma_i)}$$

パラメータの値は負担率が決まれば求まる

$$\mu_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma_n^k x_n$$

$$\Sigma_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma_n^k (x_n - \mu_k)(x_n - \mu_k)^T$$

$$\pi_k = \frac{N_k}{N}, \quad N_k = \sum_{n=1}^{N} \gamma_n^k$$

交互に一方を固定し，他方を求めればよい

EM アルゴリズム

data set $X \subseteq \mathbb{R}^d$, integer K

EM Algorithm

1: Initialize parameters $\mu_k, \Sigma_k, \pi_k \in \mathbb{R}^d (k = 1, 2, \cdots, K)$
2: **while** log-likelihood doesn't converge
3: **for** $n \in \{1, \cdots, N\}$
4: **for** $k \in \{1, \cdots, K\}$
5: "**E step**". Evaluate the responsibilities

$$\gamma_n^k = \frac{\pi_k \mathcal{N}(x_n | \mu_k, \Sigma_k)}{\sum_{i=1}^K \pi_i \mathcal{N}(x_n | \mu_i, \Sigma_i)}$$

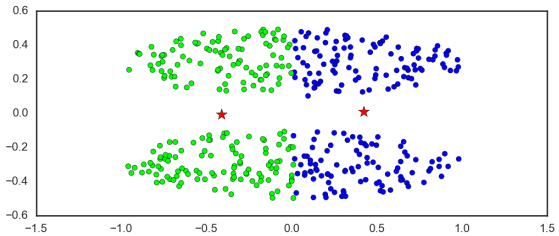
6: **for** $k \in \{1, \cdots, K\}$
7: "**M step**". Re-estimate the parameters

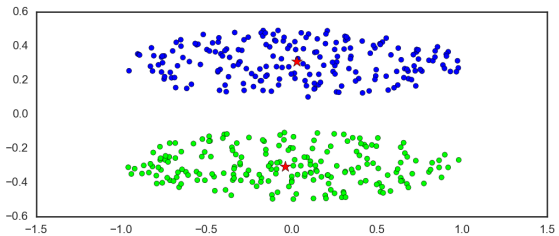
$$\mu_k = \frac{1}{N_k} \sum_{n=1}^N \gamma_n^k x_n, \Sigma_k = \frac{1}{N_k} \sum_{n=1}^N \gamma_n^k (x_n - \mu_k)(x_n - \mu_k)^T, \pi_k = \frac{N_k}{N}$$

混合ガウス -初期値依存性-

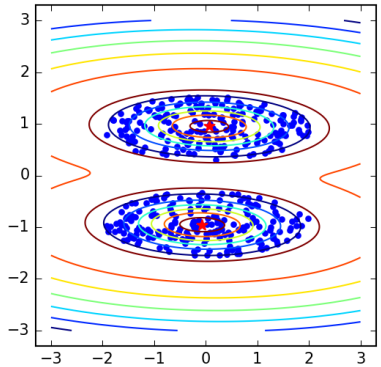
混合ガウスは初期値の影響を受けにくい

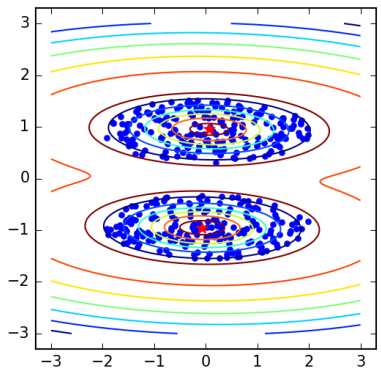
k-means





混合ガウス





Thank you for your attention !!

潜在変数モデル

- 観測データ X が与えられたとき、実はそれらは潜在的にグループ C_1, C_2 に分かれていて，**データはその潜在的なグループに依存して生成される**と考える
- データ x がどのクラスタに属するかを表す**潜在変数**

$$\boldsymbol{z} = (z_1, \cdots, z_K)^T$$

(どれか 1 つの z_k だけが 1 で，他は 0) を導入する.

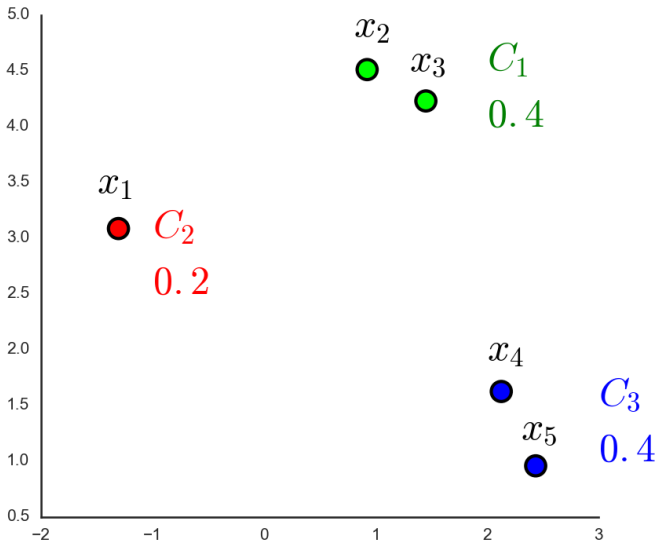
$$p(z_k = 1) = \pi_k$$

$z_k = 1$ になるというのは k 番目のガウス分布から生まれたことを表し，この確率を π_k だとすると,

$$p(\boldsymbol{z}) = \prod_{k=1}^{K} \pi_k^{z_k}$$

潜在変数モデル

$$\boldsymbol{z}^{\mathbf{1}} = (0, 1, 0)$$



	$k=1$	$k=2$	$k=3$
x_1	0	1	0
x_2	1	0	0
x_3	1	0	0
x_4	0	0	1
x_5	0	0	1
π_k	0.4	0.2	0.4