知能情報システム

第7回 機械学習

教師なし学習 (その2)

秦野 亮

東京理科大学 経営システム工学科

2025

本日の講義内容

クラスタリングについて引き続き学ぶ

1. 前回の問題の解説
2. K-平均法によるクラスタリング
3. 密度法によるクラスタリング

主なクラスタリングの分類 再掲

- 階層的手法
  - 短連結法（最短距離法）
  - 完全連結法（最長距離法）
  - 群平均法
  - 重心法
  - Ward法（ウォード法）
- 非階層的手法
  - K-means法
  - ファジイクラスタリング法
  - 密度法

手を動かしてみよう！ 再掲

- いま、1次元のデータ $D = \{0, 3, 1, 5.5\}$ が与えられている。
- 要素同士の類似度は、**差の逆数**で測る事とする。

(1) このデータに対し、短連結法による凝集型クラスタリングを実行せよ。

(2) 同様に、完全連結法による凝集型クラスタリングを実行せよ。

凝集型階層クラスタリングのアルゴリズム 再掲

入力:データの集合 $D = \{\boldsymbol{x}_1, \cdots, \boldsymbol{x}_{|D|}\}$

クラスタの集合 $C = \{c_1, \cdots, c_{|D|}\}$

$c_1 = \{\boldsymbol{x}_1\}, \cdots, c_{|D|} = \{\boldsymbol{x}_{|D|}\}$

while $|C| \geq 2$ #停止条件

$(c_m, c_n) = \arg\max_{c_i, c_j \in C} sim(c_i, c_j)$

$Merge(c_m, c_n)$

end while

ただし, simは類似度, $Merge$は併合を表す関数

参考: 高村大也, 言語処理のための機械学習入門, コロナ社, 2010.

クラスタ同士の類似度の測り方 再掲

いま、クラスタの要素x_i, x_jに
類似度$sim(x_i, x_j)$が与えられているとする。

- 短連結法

$$sim(c_i, c_j) = \max_{x_k \in c_i, x_l \in c_j} sim(x_k, x_l)$$

  - 2つのクラスタが与えられた時、それらの中で最も似ている要素対の類似度をその2つのクラスタの類似度とする。
  - max をmin にすると、完全連結法の類似度となる。

- 重心法

$$sim(c_i, c_j) = sim(\frac{1}{|c_i|}\sum_{x \in c_i} x, \frac{1}{|c_j|}\sum_{x \in c_j} x)$$

  - 2つのクラスタが与えられた時、それらのクラスタの重心ベクトル間の類似度をその2つのクラスタの類似度とする。

クラスタペア(c_m, c_n)の選び方 類似度版

$\arg\max_{x \in X} f(x)$ ：関数fの出力値が最大となる引数 $x \in X$ を選ぶ

短連結法

$$(c_m, c_n)$$
$$= \arg\max_{c_i, c_j \in C} sim(c_i, c_j)$$
$$= \arg\max_{c_i, c_j \in C} \max_{x_k \in c_i, x_l \in c_j} (sim(x_k, x_l))$$

完全連結法

$$(c_m, c_n)$$
$$= \arg\max_{c_i, c_j \in C} sim(c_i, c_j)$$
$$= \arg\max_{c_i, c_j \in C} \min_{x_k \in c_i, x_l \in c_j} (sim(x_k, x_l))$$

解答例

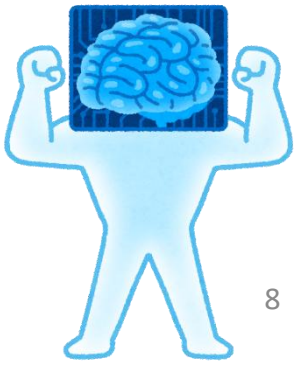
昇順に並べた要素間に｜を置くことで
クラスタを表すとしよう（｜と｜の間が1つのクラスタ）。

(1)｜0｜1｜3｜5.5｜
　　｜0 1｜3｜5.5｜
　　｜0 1 3｜5.5｜
　　｜0 1 3 5.5｜

(2)｜0｜1｜3｜5.5｜
　　｜0 1｜3｜5.5｜
　　｜0 1｜3 5.5｜
　　｜0 1 3 5.5｜

赤字の要素が、クラスタ間の
距離計算に用いられる要素。

離れた要素同士が併合されることは無いので
この様な簡潔な表現ができました。



クラスタリングにおける距離と類似度

- 類似度$sim(x, y)$
  - 実数値（$[0,1]$や$[-1,1]$に正規化される事が多い）
  - 似ているほど値が大きい
  - $\arg\max_{i,j \in X} sim(i, j)$ はi, jの類似度が最大のペア(i, j)を選ぶ
- 距離$d(x, y)$
  - 実数値（大抵$[0, \infty)$だが、$[0,1]$に正規化される事もある）
  - 距離が近いほど値が小さい
  - 距離$:= 1 -$ 類似度 となるものもある（コサイン距離など）
  - $\arg\min_{i,j \in X} d(i, j)$ は距離が最小のペア(i, j)を選ぶ

クラスタリングでは、一般に、
「よく似ているもの（類似度大）」・「距離が近いもの（距離小）」
のデータ同士をひとまとめ（マージ）にする点に要注意！！

クラスタペア(c_m, c_n)の選び方 距離版

$\arg\min_{x \in X} f(x)$ ：関数fの出力値が最小となる引数 $x \in X$ を選ぶ

短連結法

$$(c_m, c_n)$$
$$= \arg\min_{c_i, c_j \in C} d(c_i, c_j)$$
$$= \arg\min_{c_i, c_j \in C} \min_{x_k \in c_i, x_l \in c_j} (d(x_k, x_l))$$

完全連結法

$$(c_m, c_n)$$
$$= \arg\min_{c_i, c_j \in C} d(c_i, c_j)$$
$$= \arg\min_{c_i, c_j \in C} \max_{x_k \in c_i, x_l \in c_j} (d(x_k, x_l))$$

手を動かしてみよう！

いま、2次元の特徴 x, y により構成される
以下のデータセットがある。

データ	特徴 x	特徴 y
d_1	2	4
d_2	5	5
d_3	3	1
d_4	1	5
d_5	5	4

要素同士の類似度を平方ユークリッド距離で測る事
とした時、重心法を用いてクラスタリングしなさい。

ヒント: 距離行列を作るとよい。

クラスタ同士の類似度の測り方 再掲

いま、クラスタの要素x_i, x_jに
類似度$sim(x_i, x_j)$が与えられているとする。

- 短連結法

$$sim(c_i, c_j) = \max_{x_k \in c_i, x_l \in c_j} sim(x_k, x_l)$$

  - 2つのクラスタが与えられた時、それらの中で最も似ている要素対の類似度をその2つのクラスタの類似度とする。
  - max をmin にすると、完全連結法の類似度となる。

- 重心法

$$sim(c_i, c_j) = sim(\frac{1}{|c_i|}\sum_{x \in c_i} x, \frac{1}{|c_j|}\sum_{x \in c_j} x)$$

  - 2つのクラスタが与えられた時、それらのクラスタの重心ベクトル間の類似度をその2つのクラスタの類似度とする。

様々な距離関数 再掲

いま，v, u をN次元のベクトルとする。

名称	定義
ユークリッド距離	$d_{\mathrm{Euc}}(v,u) = \sqrt{\Sigma_{i=1}^{N}(v_i - u_i)^2}$
平方ユークリッド距離	$d_{\mathrm{quad}}(v,u) = d_{\mathrm{Euc}}^2(v,u)$
コサイン距離	$d_{\cos}(v,u) = 1 - \mathrm{sim}_{\cos}(v,u) = 1 - \frac{v \cdot u}{\lvert v \rvert \lvert u \rvert}$

なお，平方ユークリッド距離，コサイン距離は半距離。

解答例

1. 各クラスタにデータを割り当てる：
$$c_1 = \{d_1\}, c_2 = \{d_2\}, \cdots, c_5 = \{d_5\}.$$
2. 以下のように距離行列を計算し、
最も値の小さい組み合わせのクラスタを併合する。

	d_1	d_2	d_3	d_4	d_5
d_1	0	10	10	2	9
d_2	10	0	20	16	1
d_3	10	20	0	20	13
d_4	2	16	20	0	17
d_5	9	1	13	17	0

d_2とd_5が属するクラスタを併合し、新しいクラスタを生成する、i.e.,
$$c_6 = c_2 \cup c_5 = \{d_2, d_5\}$$

※対象律: $sim(x, y) = sim(y, x)$が成り立つので、
行列と言っても上半分だけ計算すればよい。

解答例

3. 併合したクラスタの重心を求める:

$$c_6\text{の重心} = \left(\frac{5+5}{2}, \frac{5+4}{2}\right) = (5, 4.5)$$

4. 以下のように距離行列を計算する:

$$sim(c_6, c_1) = (5-2)^2 + (4.5-4)^2 = 9.25$$
$$sim(c_6, c_3) = (5-3)^2 + (4.5-1)^2 = 16.25$$
$$sim(c_6, c_4) = (5-1)^2 + (4.5-5)^2 = 16.25$$

	d_1	d_3	d_4	c_6
d_1	0	10	2	9.25
d_3	10	0	20	16.25
d_4	2	20	0	16.25
c_6	9.25	16.25	16.25	0

d_1とd_4が属するクラスタを併合し、新しいクラスタを生成する、i.e.,

$$c_7 = c_1 \cup c_4 = \{d_1, d_4\}$$

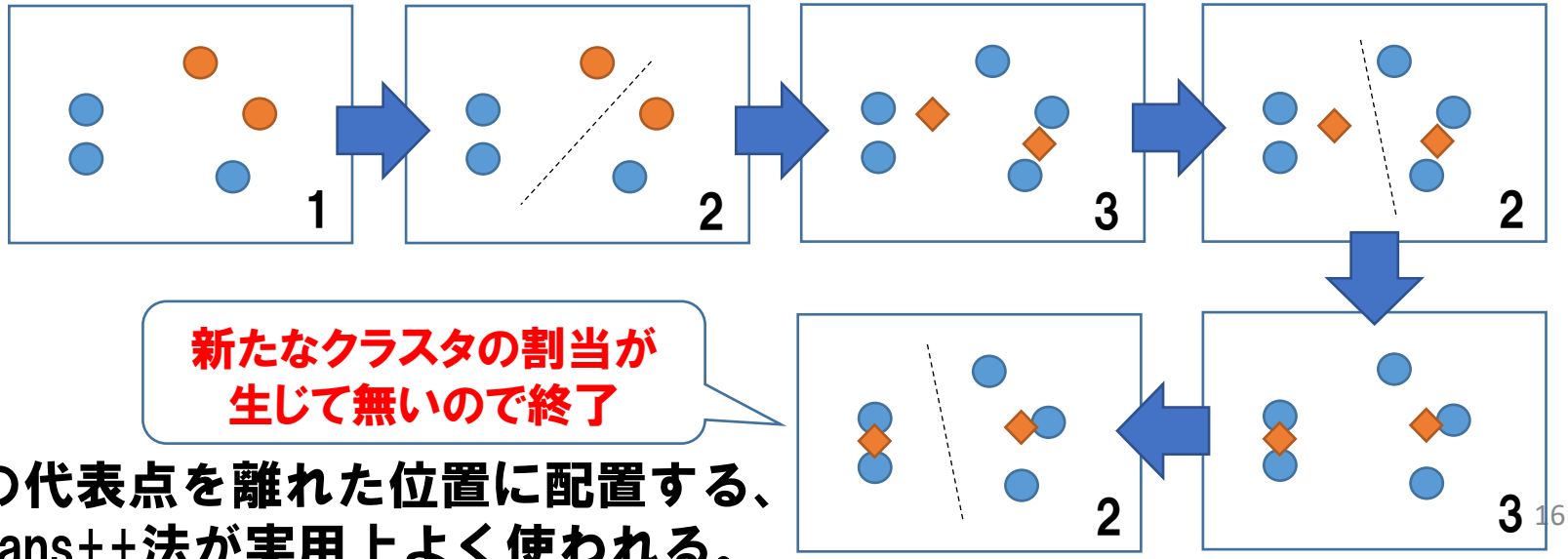
(以下略)

まず適当に分けて、後で
よりうまく分かれるように調整…

K-means法

最終的なクラスタ数

1. データセットからk個の代表点（セントロイド）を決める。
2. データセットの各要素をどちらか近い方の
   代表点を要素とするクラスタに割り当てる。
3. 各クラスタの重心を代表点として再計算し、
   新たなクラスタ割当がなくなるまで2，3を繰り返す。



最初の代表点を離れた位置に配置する、
K-means++法が実用上よく使われる。

K-means法のアルゴリズム

入力:データの集合 $D = \{\boldsymbol{x}_1, \cdots, \boldsymbol{x}_{|D|}\}$

クラスタ数k

Dの要素から、ランダムに代表点$\boldsymbol{m}_1, \cdots, \boldsymbol{m}_k$ を選ぶ。

Until 収束

foreach $\boldsymbol{x}_i \in D$

$c_{\max} = \arg\max_{c} sim(\boldsymbol{x}_i, \boldsymbol{m}_c)$

insert $\boldsymbol{x}_i$ into $c_{\max}$

end foreach

$\forall c \left(m_c = \frac{1}{|c|}\sum_{x_i \in c} \boldsymbol{x}_i\right)$ #各クラスタの代表点≒重心に設定

end until

参考: 高村大也, 言語処理のための機械学習入門, コロナ社, 2010.

手を動かしてみよう！

- いま、1次元のデータ $D = \{0, 3, 1, 5.5\}$
  が与えられている。
- $k = 2$ としてK-means法を用いて
  クラスタリングを実行せよ。
- ただし、初期状態において各クラスタの代表点は
  以下の様に設定されているとする：
  クラスタc_1の代表点$m_1 = -1$
  クラスタc_2の代表点$m_2 = 6$

解答例

1. データセットの各要素を最寄りの
   代表点を要素とするクラスタに割り当てる:
   $$c_1 = \{0,1\}, c_2 = \{3,5.5\}$$
2. 代表点の再計算
   $$m_1 = \frac{0+1}{2} = 0.5, \qquad m_2 = \frac{3+5.5}{2} = 4.25$$
3. データセットの各要素を最寄りの
   代表点を要素とするクラスタに割り当てる:
   $$c_1 = \{0,1\}, c_2 = \{3, 5.5\}$$

   新たなクラスタの割当が無いので終了。

Sum of Squared Error

クラスタ内誤差平方和 (SSE)

Cluster Inertia

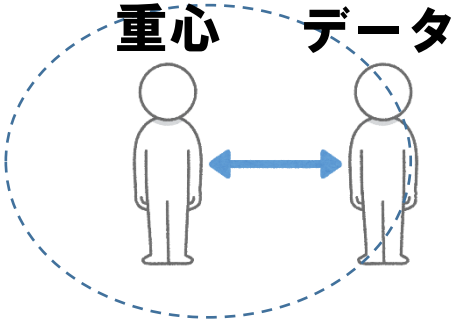
クラスタの慣性とも呼ばれる、
クラスタリングの結果の「歪み」の程度を表す指標。

$$SSE = \sum_{i=1}^{n} \sum_{j=1}^{k} w(x_i, c_j) d_{\text{quad}}(\boldsymbol{x}_i, \boldsymbol{m}_j)$$

平方ユークリッド距離

ただし、m_jはクラスタc_jの代表点であり，関数wは
以下の様に定義される：

$$w(x_i, c_j) = \begin{cases} 1 & \text{if } x_i \in c_j, \\ 0 & \text{o.w.} \end{cases}$$

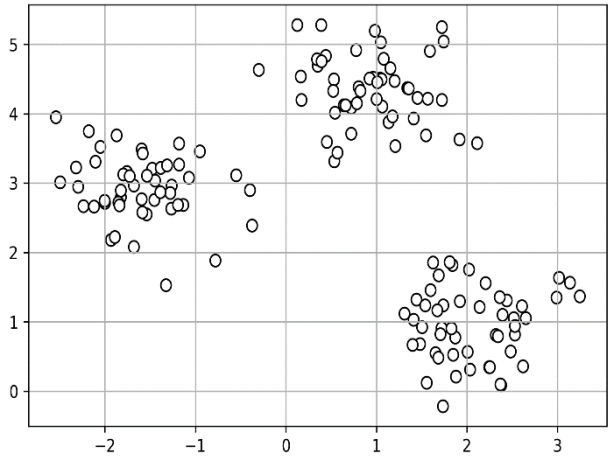


要するに各クラスタの重心から、同クラスタ内の各データまでの距離の総和。

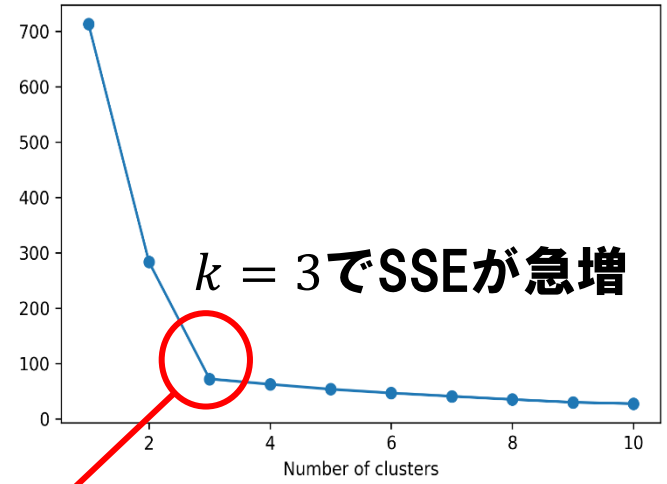
Elbow Method

エルボー法によるクラスタ数の決定

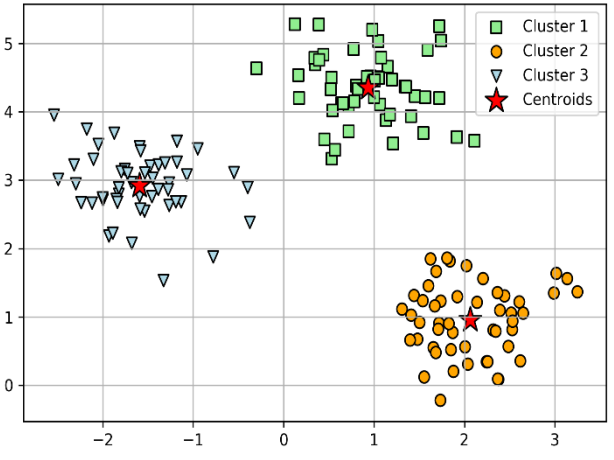
クラスタ内誤差平方和 (SSE) に基づいて、
タスクに最適なクラスタ数kを推定する方法。



(1) データセット



(2) 各kでのSSEの状態



(3) $k = 3$での結果

SSEが急速に増え始めるkの値を特定する。

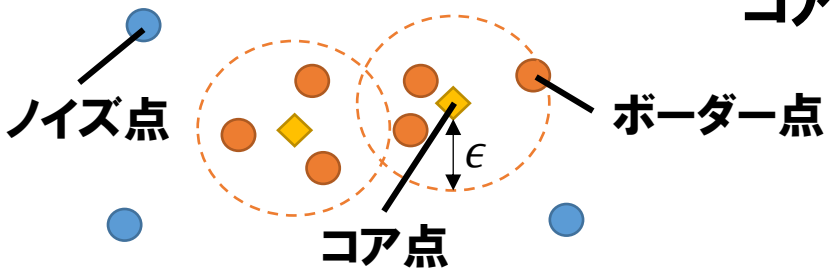
参考: S. Raschka, V. Mirjalili, Python Machine Learning 3rd Ed., Impress, 2020.

Density-based Spatial Clustering of Applications with Noise

密度法 (DB SCAN)

指定された半径ϵ以内に存在するデータを
クラスタの要素とする。

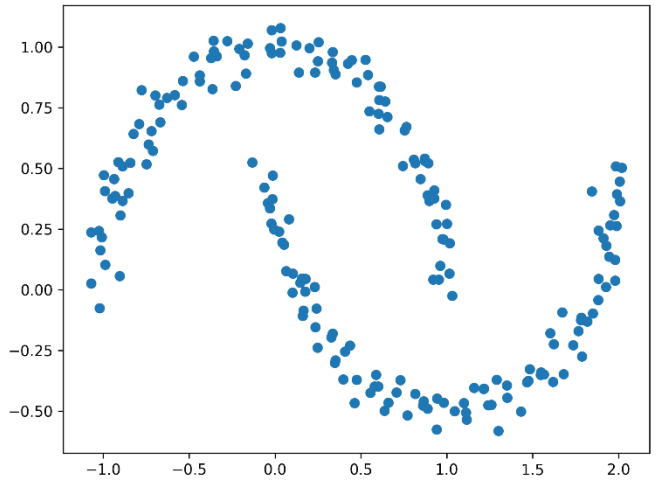
1. **コア点、ボーダー点、ノイズ点を以下のように設定する:**
   コア点:半径ϵ以内に指定数以上の
   隣接データを持つデータ。
   ボーダー点:隣接データの個数が指定数未満で
   あるものの、コア点の半径ϵ以内に位置するデータ。
   ノイズ点:上記意外のデータ。
2. **コア点ごとにクラスタを形成する。**
3. **コア点同士が半径ϵ以内のクラスタ同士を併合する。**
4. **各ボーダー点を、それと対になっている**
   コア点のクラスタに割り当てる。



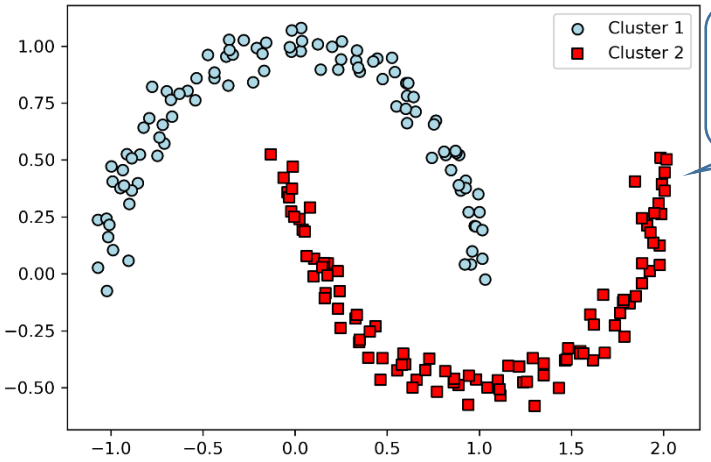
1

DB SCANと他のクラスタリングとの違い

データセット



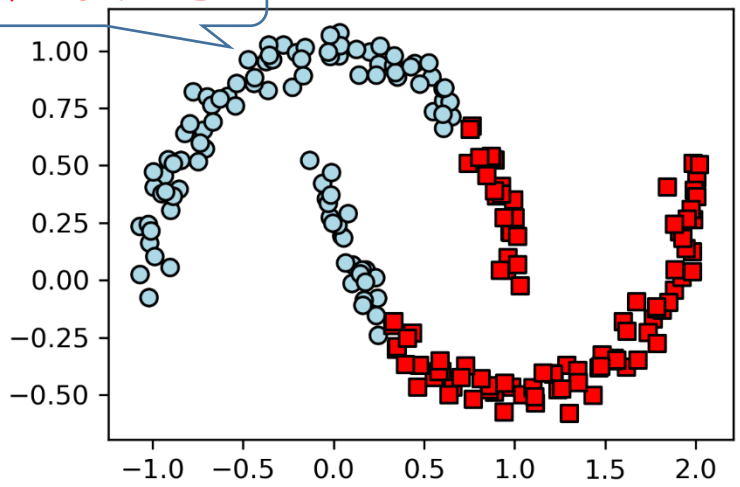
DB SCAN



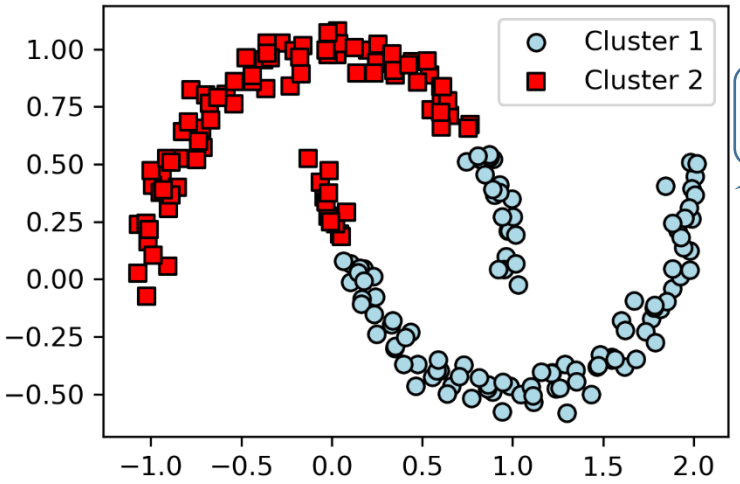
任意の形状OK

球状になりがち

K-means法



凝集型階層クラスタリング



階層？