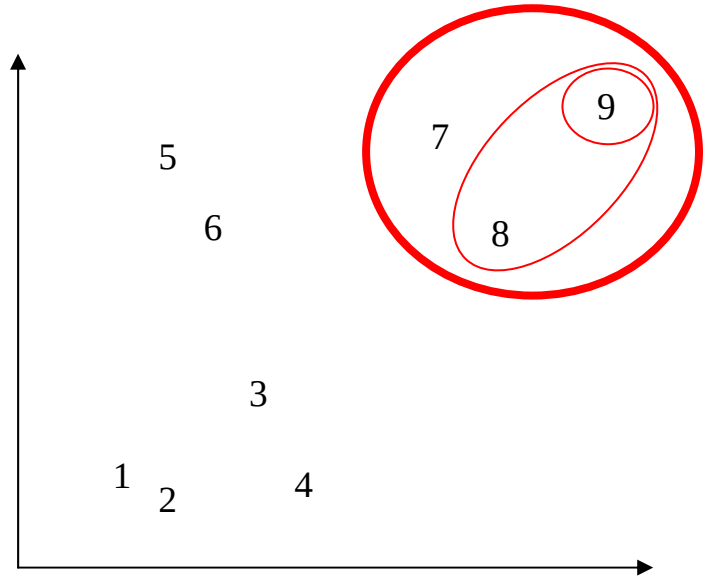
ー　クラスタリング　ー

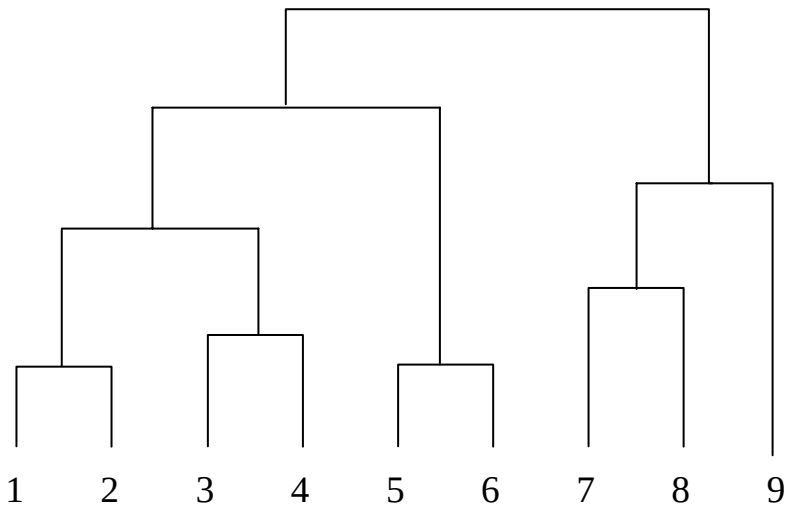
クラスタリングとは、データ間に距離を定義し、

距離が近いデータ同士をグループ（クラスター）にまとめる作業

例

- 顧客のクラスタリング
- 化学物質のクラスタリング
- 遺伝子のクラスタリング





Dendrogram

階層的クラスタリング

ボトムアップ型

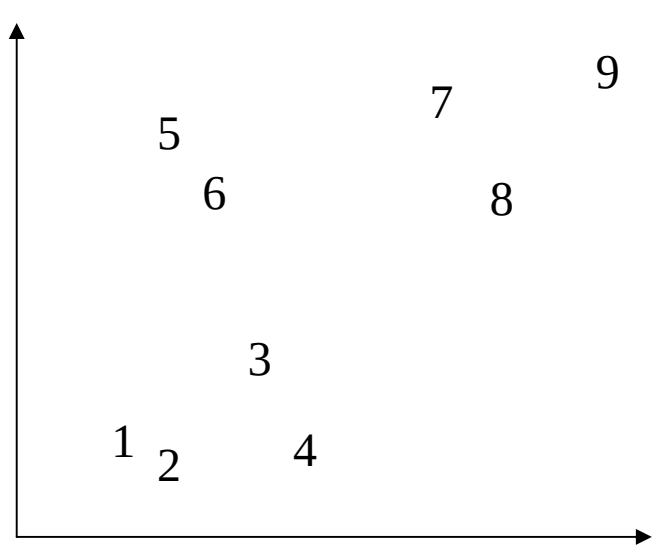
近いクラスター同士を融合するプロセスを繰り返す

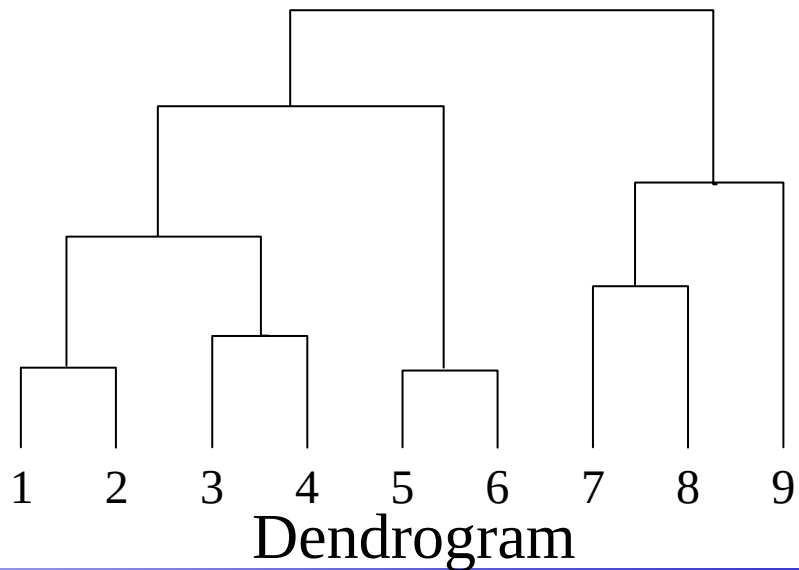
クラスター（点の集合）C_i, C_j 間の距離に結果が依存

距離の例： $d(\vec{x}, \vec{y})$: $\vec{x}, \vec{y}$　間のユークリッド距離

$$D_{\min}(C_i, C_j) = \min\{d(\vec{x}, \vec{y}) \mid \vec{x} \in C_i, \vec{y} \in C_j\}$$

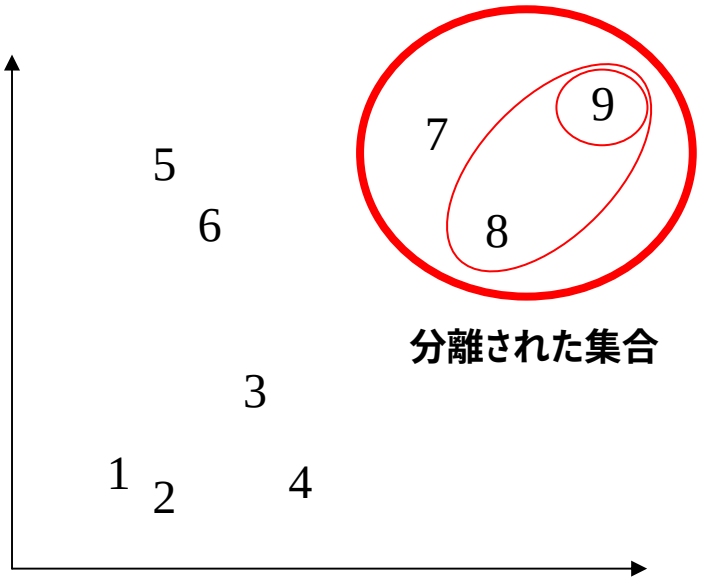
$$D_{\max}(C_i, C_j) = \max\{d(\vec{x}, \vec{y}) \mid \vec{x} \in C_i, \vec{y} \in C_j\}$$

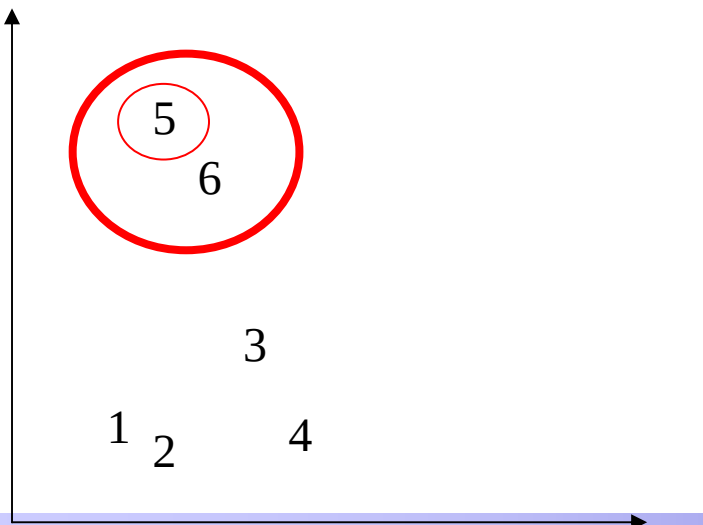




階層的クラスタリング：トップダウン分割型の例

S-plus で使われている diana 法　　L. Kaufman, P. Rousseuw. "*Finding Groups in Data- An Introduction to Cluster Analysis*." Wiley Series in Probability and Mathematical Sciences, 1990.





avgを距離の平均値とするとき、$i \in V - S$について

$V(i,S) \equiv \mathrm{avg}_{j \notin S} d(i,j) - \mathrm{avg}_{j \in S} d(i,j)$

$V(i,S)$を最大にするiは「Sに最も近く$V-S$から遠い」と解釈

初期化ステップ：　Sを空集合{}に初期化

繰り返しステップ：

$V(i,S)$を最大にする$i \in V - S$をhとする。

終了判定：

$V(h,S) > 0$ならばhはSに近いのでSに追加し、

他にもSに近い要素があるか調べるため、

繰り返しステップを実行。

$V(h,S) \leq 0$ならば hはSに近くなく、追加せず終了。

Sを「SplinterGroup」と呼び、Vから分離。

以上のステップを繰り返しSplinterGroupを分離。

他のアプローチ

確率モデル型クラスタリング

複数の (正規) 分布が重なり全データが分布していると仮定し、各分布を予測

k - クラスタリング

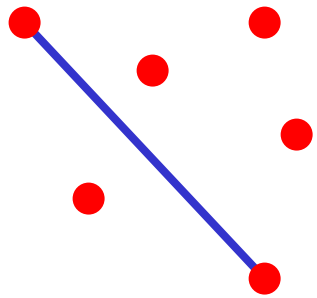
k - クラスタリング

d 次元ユークリッド空間 R^d 内の点集合 S を、S を覆い $\left(S = S_1 \cup S_2 \cup \cdots \cup S_k\right)$ かつ、互いに交わらない k 個の部分集合（クラスターと呼ぶ）$S_1, S_2, \cdots, S_k$ に分解すること。

クラスター S_i の評価： 直径

$$diameter(S_i) = \max\left\{ \left\| \vec{x}_1 - \vec{x}_2 \right\| \mid \vec{x}_1, \vec{x}_2 \in S_i \right\}$$

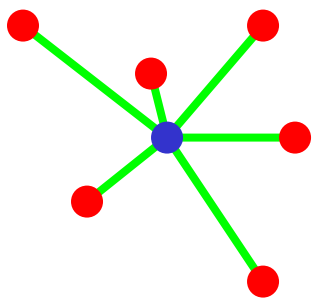
$$\left\| \left(x_1, \cdots, x_d\right) \right\| = \left(\sum_{i=1,\cdots,d} x_i^2 \right)^{1/2}$$

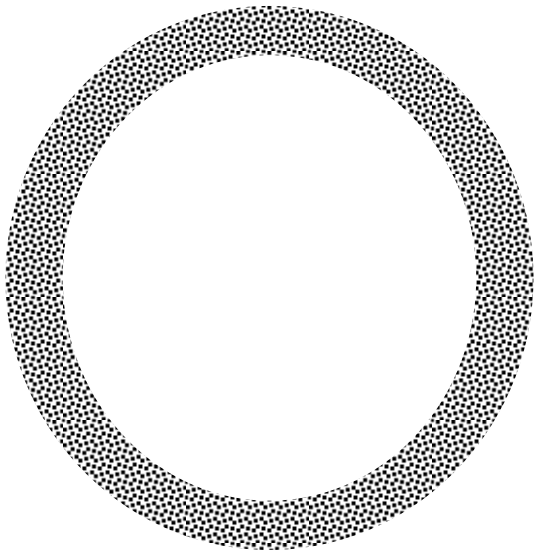


S_i の評価：重心からの距離の分散

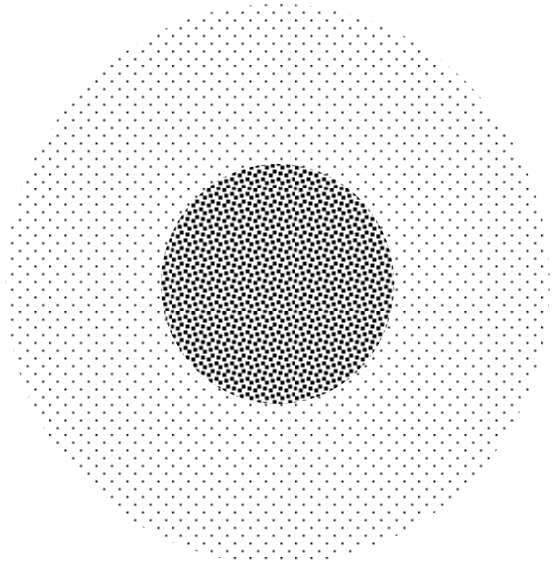
$$c(S_i) = \frac{1}{\left|S_i\right|} \sum_{\vec{x} \in S_i} \vec{x}$$

$$\mathrm{var}(S_i) = \frac{1}{\left|S_i\right|} \sum_{\vec{x} \in S_i} \left\| \vec{x} - c(S_i) \right\|^2$$





S_1



S_2

$$\text{diameter}(S_1) = \text{diameter}(S_2)$$

$$\text{var}(S_1) >> \text{var}(S_2)$$

誤差二乗平均によるクラスターの評価

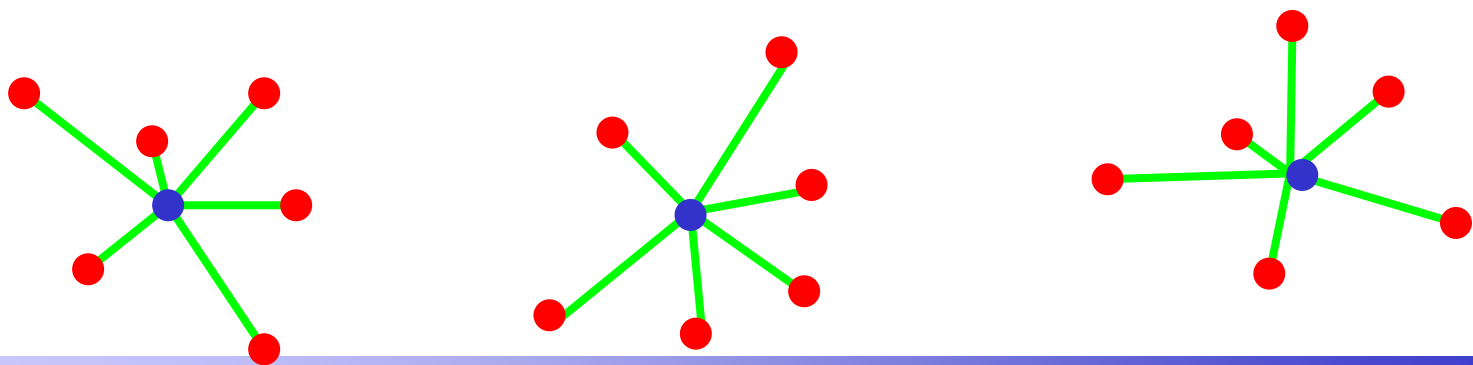
S の k –クラスタリングを $\{S_1, \ldots, S_k\}$

- 各クラスターの重心は $c(S_i) = \frac{1}{|S_i|} \sum_{\vec{y} \in S_i} \vec{y}$

- Sの各点 $\vec{x}$と、$\vec{x}$が属するクラスターの重心間の距離の分散

$$\frac{1}{|S|} \sum_{i=\{1,\ldots,k\}} \sum_{\vec{x} \in S_i} \left\| \vec{x} - c(S_i) \right\|^2$$

を*mean squared error* (誤差二乗平均) と呼ぶ



誤差二乗平均を最小化する k-クラスタリングを計算する
問題はＮＰ困難(現実的な時間で解けない)

できるだけ小さくすると言われているアルゴリズムとして
k-means 法がある

k-means 法の様々な変形が広く使われている

$k-means$法

クラスターの代表点の集合をT、
Tの点$\vec{y}$を代表点とするクラスターを$S_{\vec{y}}$と表現.

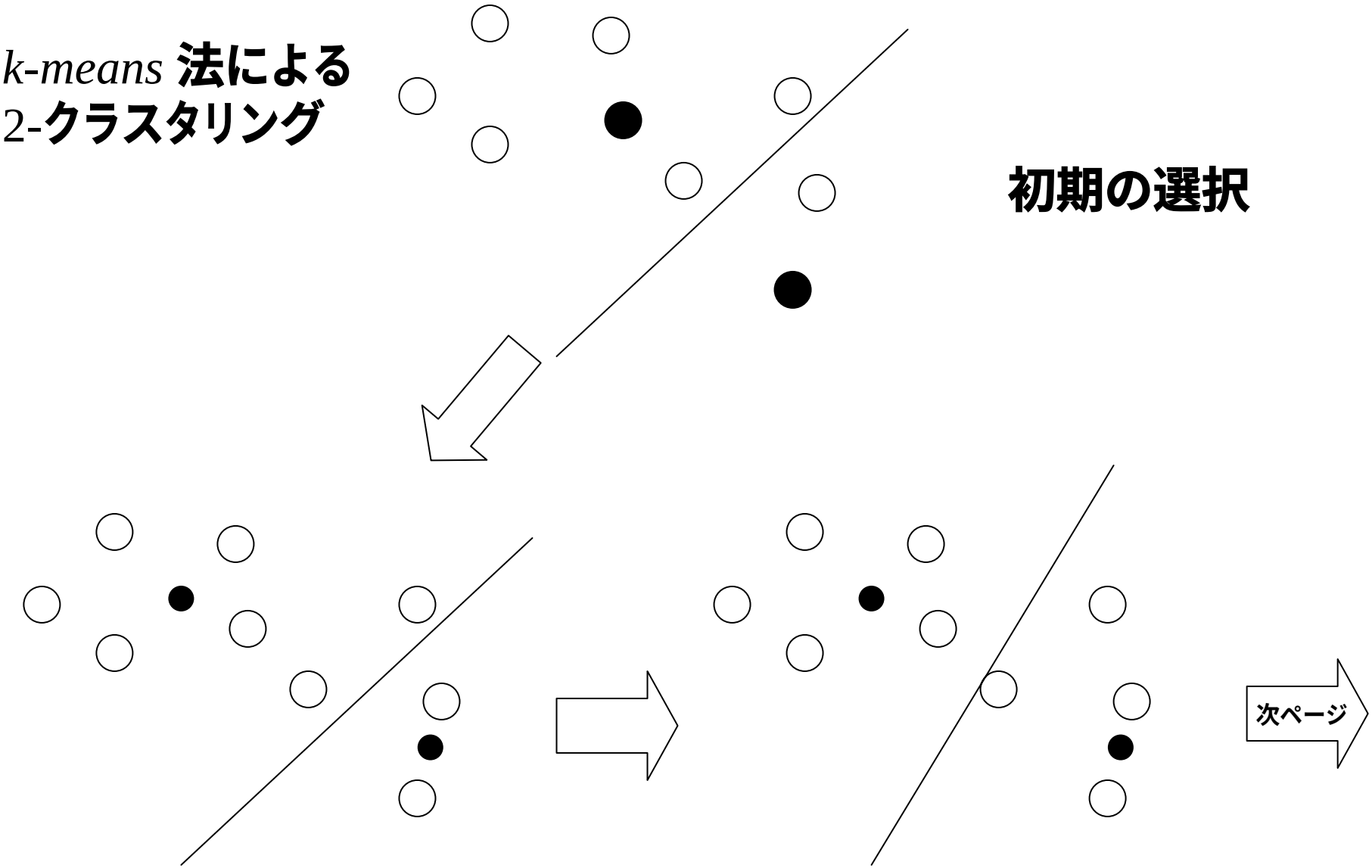
1.（初期化）Sからk個の点を選択し、Tの初期集合とする.

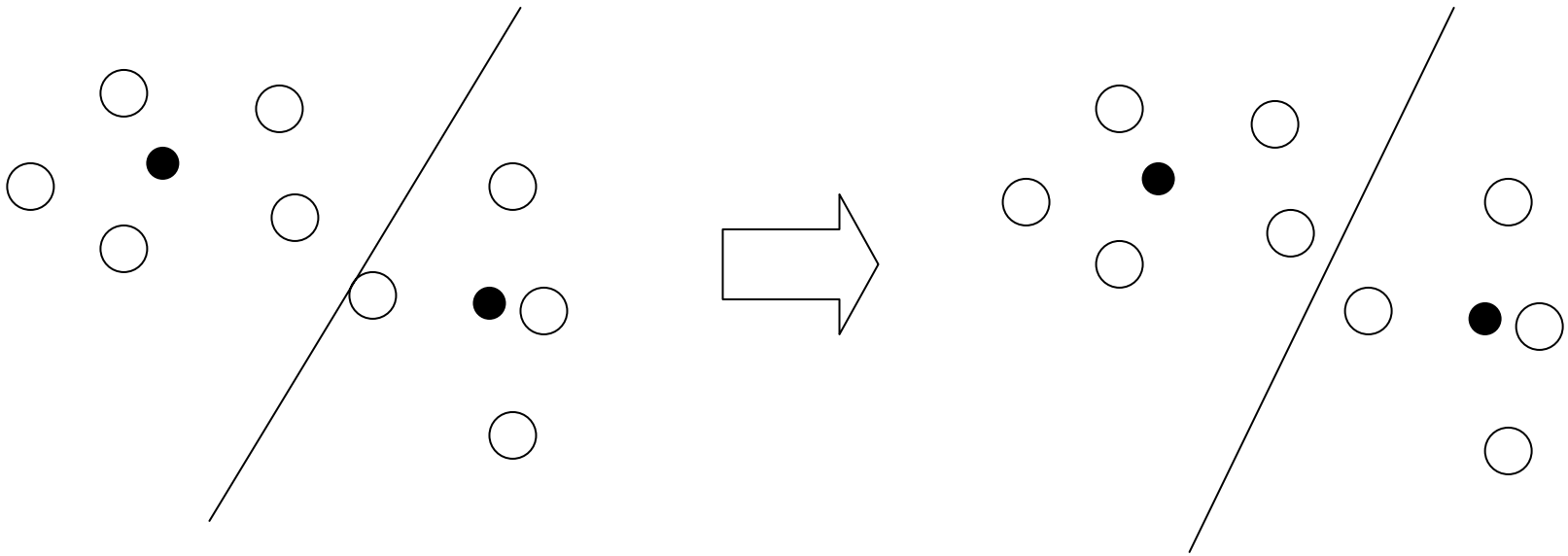
2.（再クラスタリング）各代表点$\vec{y} \in T$について$S_{\vec{y}}$を空集合にリセット.
Sの各点$\vec{x}$に最も近いTの点$\vec{y}$ $\left(\left\|\vec{x}-\vec{y}\right\| = \min_{\vec{z} \in T}\left\|\vec{x}-\vec{z}\right\|\right)$を計算し、$\vec{x}$を$S_{\vec{y}}$に追加.

3.（代表点を再計算）$S_{\vec{y}}$に登録された点全体の重心は代表点$\vec{y}$から
ずれている可能性がある.各代表点$\vec{y} \in T$を重心$c\left(S_{\vec{y}}\right) = \frac{1}{\left|S_{\vec{y}}\right|}\sum_{\vec{u} \in S_{\vec{y}}}\vec{u}$に更新.

4. 誤差二乗平均が改善しなくなるまで、ステップ２と３を繰返す.

k-means 法による 2-クラスタリング





重心を再計算

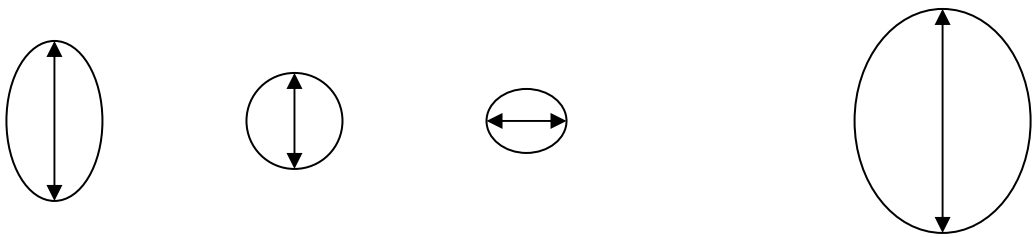
再クラスタリング

誤差二乗平均は収束

クラスターの評価に直径を使う場合

S の k –クラスタリング $C=\{S_1, S_2, \cdots, S_k\}$ に対して

$q(C)=\max\{\, diameter(S_i) \mid i=1,\cdots,k \,\}$　と定義



$q(C)$ を最小化する k –クラスタリング C を
効率的に計算できるか？

与えられた B に対して $q(C)\le B$ となる k –クラスタリングが
存在するか否かを決定する問題はＮＰ完全

近似的解法

$opt = \min\{q(C) \mid C \text{は} S \text{の} k\text{–クラスタリング}\}$ とおく

$q(C) \leq 2 \bullet opt$ となる k–クラスタリング C を生成する

アルゴリズムが存在する

Gonzalez’s farthest point heuristics

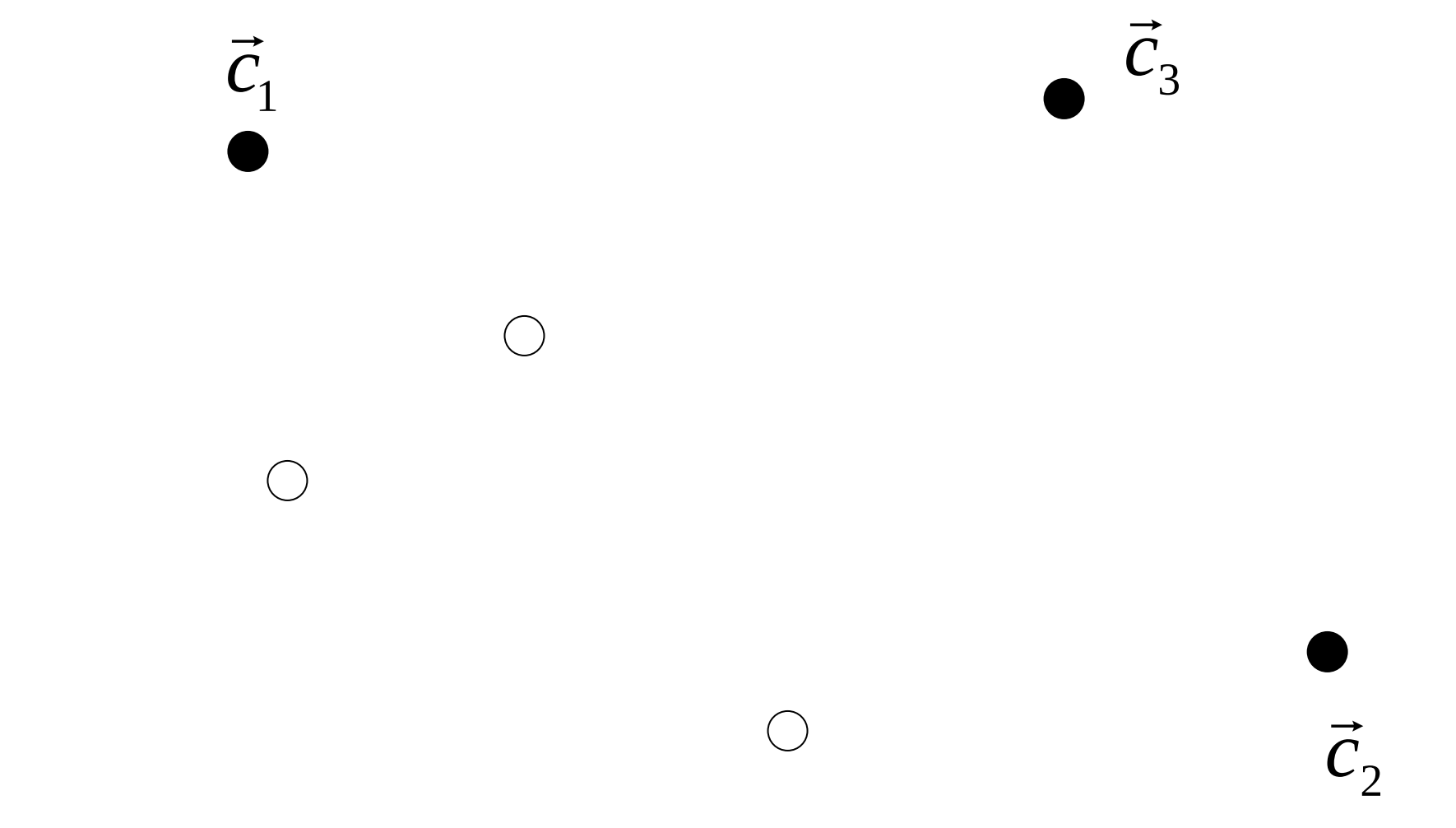
- T を S のクラスターの代表点の集合とし、空集合で初期化
- S から１点 $\vec{c}_1$ を選択し T に追加
- 各 $j = 2, \ldots, k$ について以下のステップを実行

1. $\vec{x} \in S - T$ に最も近い T の点を $neighbor(\vec{x})$ と記述

$\vec{x}$ は $neighbor(\vec{x})$ を代表点とするクラスターに属すると定義

2. 属するクラスターの代表点との距離が最大の点 $\vec{c}_j \in S - T$ (farthest point) を T に追加

$$\left\| \vec{c}_j - neighbor(\vec{c}_j) \right\| = \max\left\{ \left\| \vec{x} - neighbor(\vec{x}) \right\| \mid \vec{x} \in S - T \right\}$$


$\vec{c}_1$
$\vec{c}_3$
$\vec{c}_2$
代表点
代表点以外の点

補題　$T = \{\vec{c}_1, \vec{c}_2, \ldots, \vec{c}_{j-1}\}$のとき、

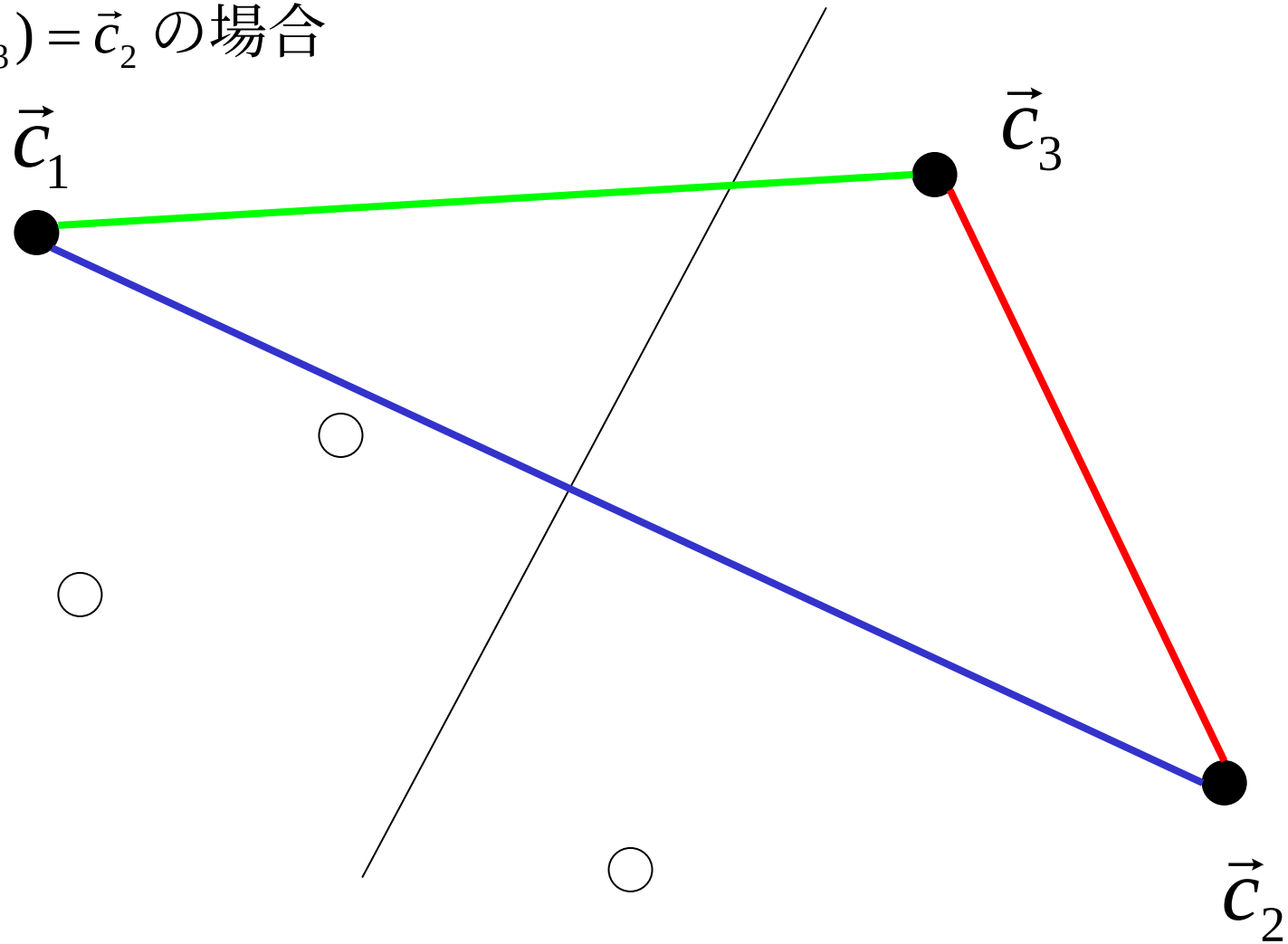
$T \cup \{\vec{c}_j\}$の任意の２点は

$\left\| neighbor(\vec{c}_j) - \vec{c}_j \right\|$以上離れている

つまり$1 \leq h < i \leq j$について

$$\left\| neighbor(\vec{c}_j) - \vec{c}_j \right\| \leq \left\| \vec{c}_h - \vec{c}_i \right\|$$

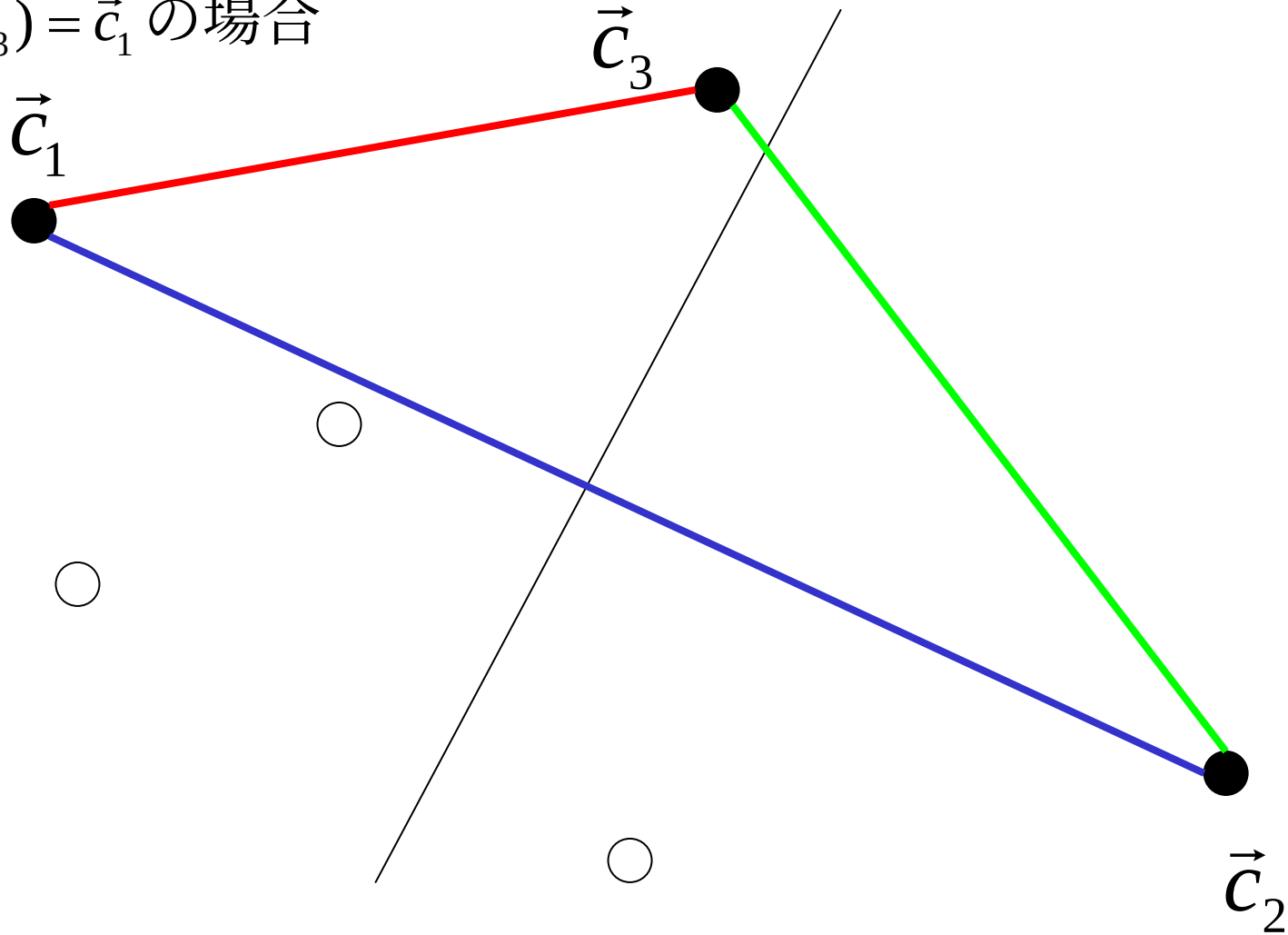
$neighbor(\vec{c}_3) = \vec{c}_2$ の場合



$neighbor(\vec{c}_3) = \vec{c}_2$なので　$\|\vec{c}_2 - \vec{c}_3\| \leq \|\vec{c}_1 - \vec{c}_3\|$

$\vec{c}_2$ が $\vec{c}_3$ より先に選択されたので　$\|\vec{c}_1 - \vec{c}_3\| \leq \|\vec{c}_1 - \vec{c}_2\|$

$neighbor(\vec{c}_3) = \vec{c}_1$ の場合



$neighbor(\vec{c}_3) = \vec{c}_1$ なので　$\|\vec{c}_1 - \vec{c}_3\| \leq \|\vec{c}_2 - \vec{c}_3\|$

$\vec{c}_2$ が $\vec{c}_3$ より先に選択されたので　$\|\vec{c}_1 - \vec{c}_3\| \leq \|\vec{c}_1 - \vec{c}_2\|$

補題　$T = \{\vec{c}_1, \vec{c}_2, \ldots, \vec{c}_{j-1}\}$のとき、$1 \le h < i \le j$について

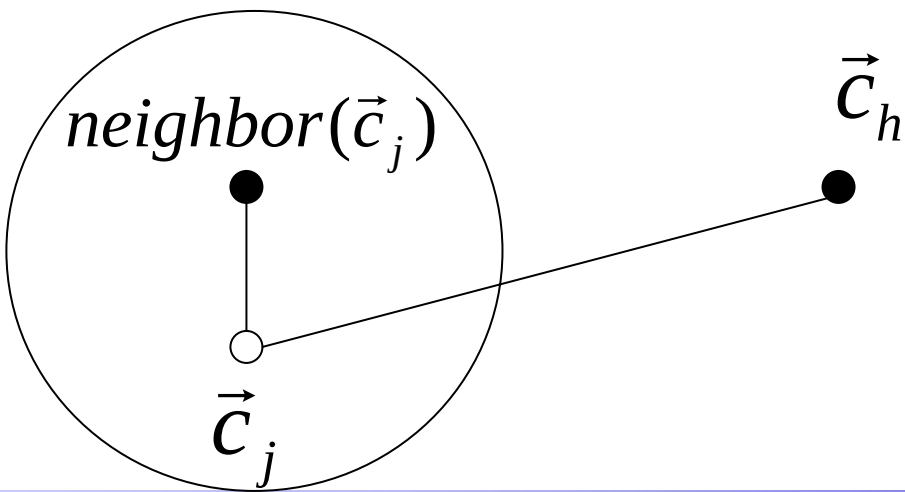
$$\left\| neighbor(\vec{c}_j) - \vec{c}_j \right\| \le \left\| \vec{c}_h - \vec{c}_i \right\|$$

つまり、$T \cup \{\vec{c}_j\}$の任意の２点間距離は$\left\| neighbor(\vec{c}_j) - \vec{c}_j \right\|$以上

jに関する帰納法.　　一般の場合を証明

- $i = j$　のとき、Tの中で最も$\vec{c}_j$に近いのは$neighbor(\vec{c}_j)$なので

$$\left\| neighbor(\vec{c}_j) - \vec{c}_j \right\| \le \left\| \vec{c}_h - \vec{c}_j \right\|$$



● 代表点

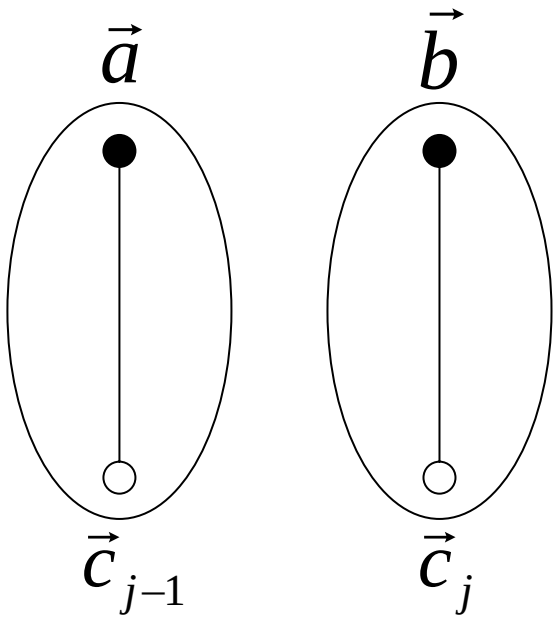
○ 代表点以外の点

$i < j$ のとき：

- $j-2$個の代表点を選択した一つ前のステップの状態、

つまり$T = \left\{ \vec{c}_1, \vec{c}_2, \ldots, \vec{c}_{\underline{\underline{j-2}}} \right\}$を考える。

この時点での$\vec{c}_{j-1}$の属するクラスターの代表点 $neighbor(\vec{c}_{j-1})$を$\vec{a}$とすれば、帰納法の仮定より $\left\| \vec{a} - \vec{c}_{j-1} \right\| \leq \left\| \vec{c}_h - \vec{c}_i \right\|$

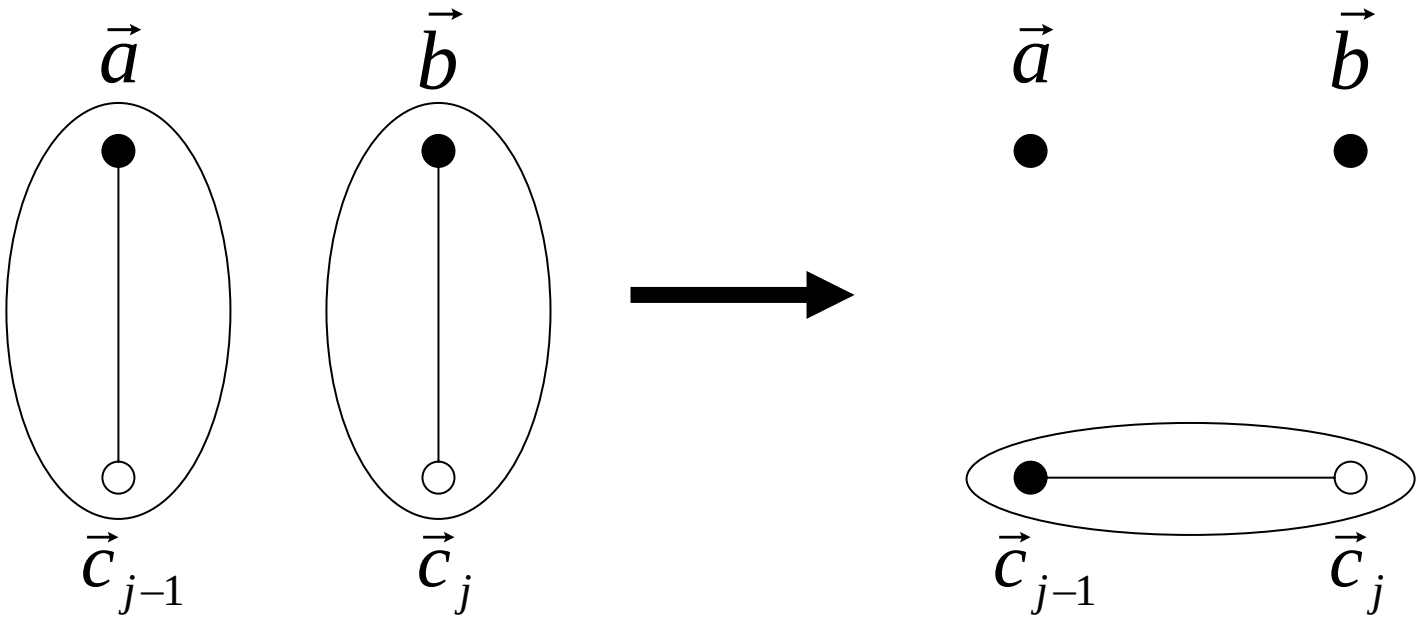


この時点で$\vec{c}_j$の属する
クラスターの代表点を$\vec{b}$

- $T = \left\{ \vec{c}_1, \vec{c}_2, \ldots, \vec{c}_{\underline{\underline{j-1}}} \right\}$ のとき、つまり$j-1$個代表点を選択後に

$\left\| neighbor(\vec{c}_j) - \vec{c}_j \right\| \leq \left\| \vec{a} - \vec{c}_{j-1} \right\|$ となることを示せば十分

$neighbor(\vec{c}_j) = \vec{c}_{j-1}$のとき



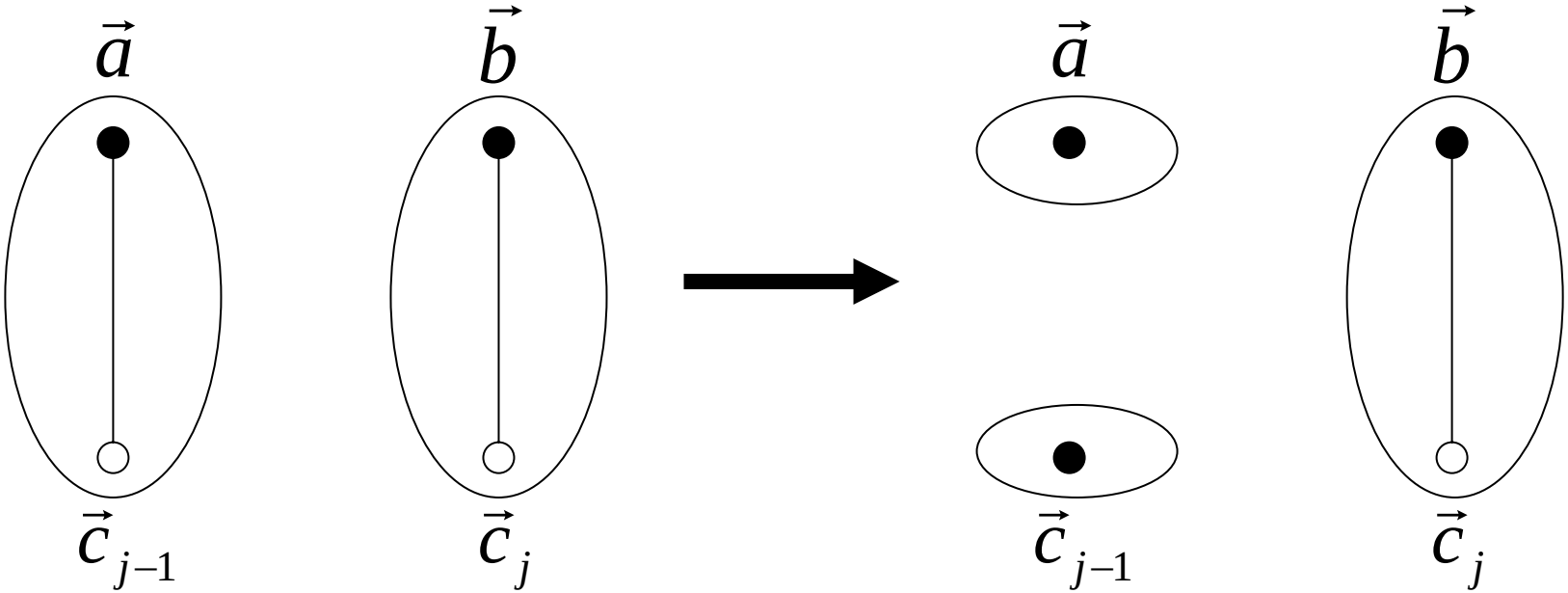
$\vec{c}_{j-1}$が代表点として選択されたので

$\left\| \vec{b} - \vec{c}_j \right\| \leq \left\| \vec{a} - \vec{c}_{j-1} \right\|$

$\vec{c}_j$ がより近い代表点$\vec{c}_{j-1}$ の
クラスターに移動したので

$\left\| neighbor(\vec{c}_j) - \vec{c}_j \right\| \leq \left\| \vec{b} - \vec{c}_j \right\|$

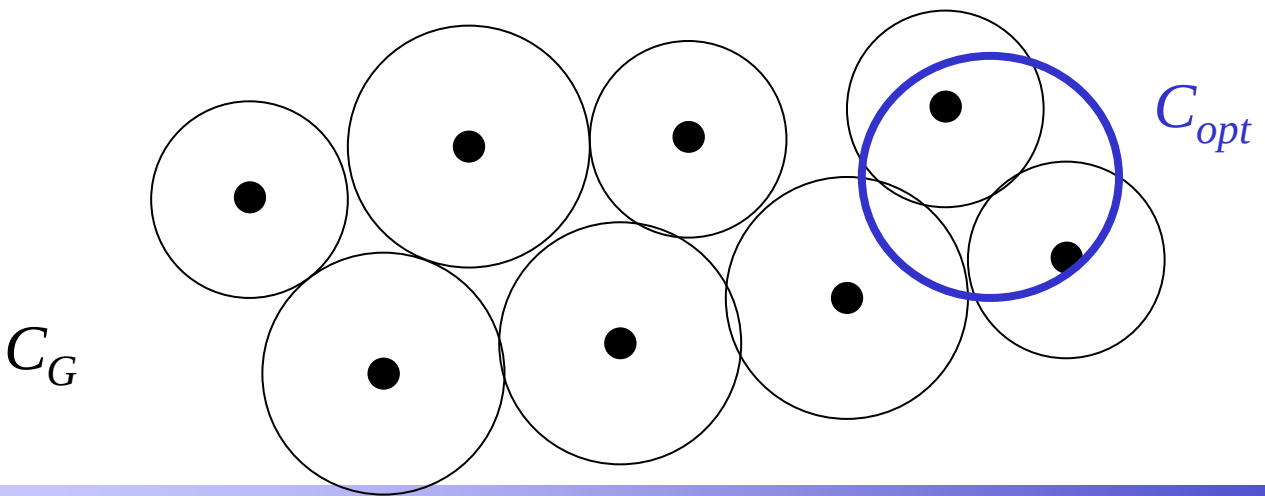
$neighbor(\vec{c}_j) = \vec{b}$のとき、つまりクラスターを移動しないとき



$\vec{c}_{j-1}$が代表点として選択されたので　$\left\| neighbor(\vec{c}_j) - \vec{c}_j \right\| \leq \left\| \vec{a} - \vec{c}_{j-1} \right\|$

定理　Gonzalez’s farthest point heuristicsが生成するk–クラスタリングをC_G
$q(C)$が最小のk–クラスタリングをC_{opt}とすれば　$q(C_G) \le 2 \cdot q(C_{opt})$

- C_Gのk個の代表点を $T = \{\vec{c}_1, \ldots, \vec{c}_k\}$
- ステップ２をもう一度実行して得られる代表点を$\vec{c}_{k+1}$，
  $D = \|\vec{c}_{k+1} - neighbor(\vec{c}_{k+1})\|$　とおく
- 補題より$k+1$個の代表点の任意の２点間の距離はD以上
- 最適なクラスタリングC_{opt}のk個のクラスターのどれかは
  $k+1$個の代表点$T \cup \{\vec{c}_{k+1}\}$のうち２点を含むので　$D \le q(C_{opt})$

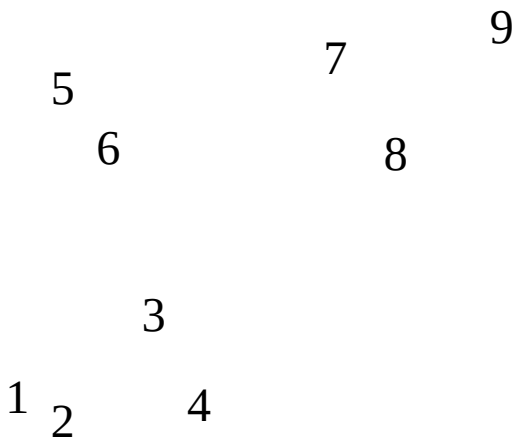


- $k+1$個の代表点$T \cup \{\vec{c}_{k+1}\}$の任意の２点間の距離はD以上（補題より）

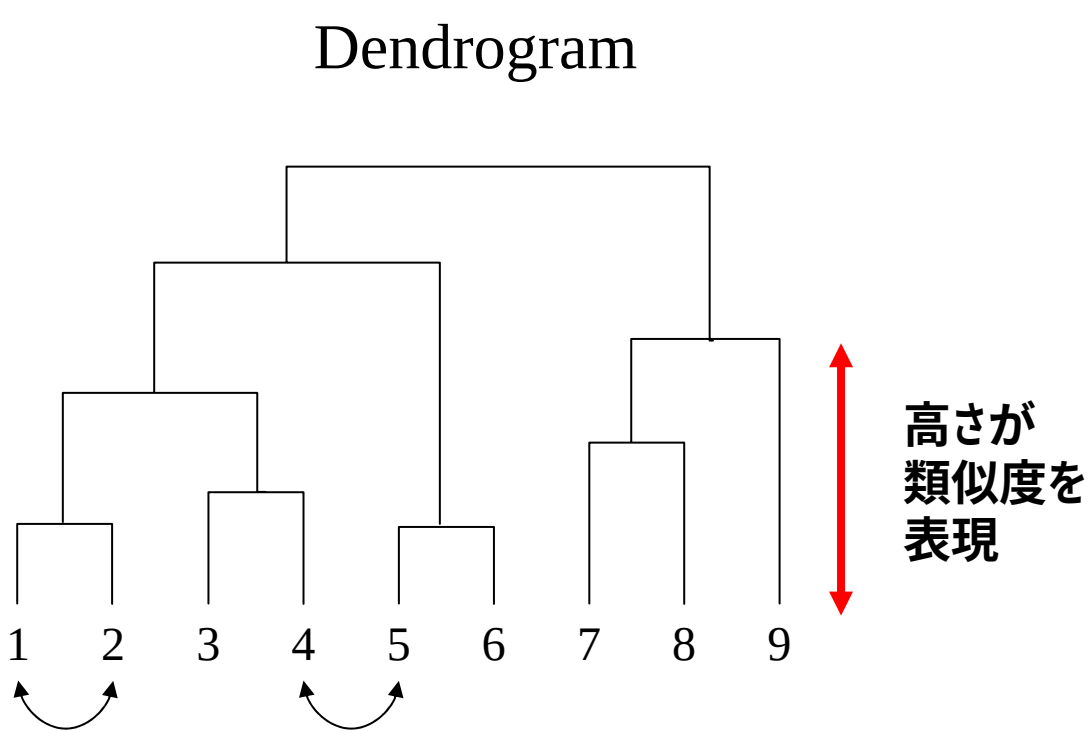
- 最適なクラスタリングC_{opt}のk個のクラスターのどれかは$k+1$個の代表点$T \cup \{\vec{c}_{k+1}\}$のうち２点を含むので　$\underline{D \leq q(C_{opt})}$

- $q(C_G) \leq 2D$より、$q(C_G) \leq 2 \cdot q(C_{opt})$

クラスタリング結果や類似性の表現



高次元の
データと仮定



各ノードで部分木の左右を交換しても構わない

点間の距離をできるだけ保存して高次元を低次元に埋め込む

Multi-dimensional Scaling

Latent Semantic Indexing

Self-Organizing Maps (SOM)

参考文献 Soumen Chakrabarti. Mining the Web: Discovering Knowledge from Hypertext Data. Morgan-Kaufmann Publishers, 352 pages, cloth/hard-bound, ISBN 1-55860-754-4 4**章**

Multi-dimensional Scaling

2点（たとえば文書）i,j 間の距離 $d_{i,j}$

点 i を低次元への写像した結果 $f(i)$

点間の距離をできるだけ保つ写像 f が望ましい

最小化したい指標 $\dfrac{\sum_{i,j}(d_{f(i),f(j)} - d_{i,j})^2}{\sum_{i,j} d_{i,j}^{\ 2}}$

最急勾配山登り法（hill climbing） による最適化
FastMap 発見的に射影する次元を見つける方法

Latent Semantic Indexing

行列 A_{mn}

たとえば

文書

トークン

0	1	0	0	0	0	0	0	1	0	0	0	1	0	0
1	0	0	0	0	1	0	0	0	0	0	1	0	0	0
0	0	0	1	0	0	0	0	0	1	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	1	0	0	1
0	1	0	0	1	0	0	0	1	0	0	0	1	0	0
1	0	0	1	0	0	0	0	0	1	0	0	0	1	0

特異値分解 (Singular Value Decomposition)

$$A_{m,n} = U_{m,r}\,\Sigma\,V_{r,n}^{T}$$

$$\Sigma = \begin{pmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_r \end{pmatrix}$$

$r(\le \min(m,n))$ は　$A_{m,n}$　のランク

Σ は対角行列　$\sigma_1 \ge \sigma_2 \ge \ldots \ge \sigma_r > 0$

U, V　直交行列（$U^T U = V^T V = E_{r,r}$，　$E_{r,r}$は単位行列）

文書間の類似性　$A_{m,n}^T A_{m,n} = \left(V \Sigma^2 V^T\right)_{n,n}$

トークン間の類似性　$A_{m,n} A_{m,n}^T = \left(U \Sigma^2 U^T\right)_{m,m}$

問合せをトークンのベクトル $q_{m,1}$ で表現

r次元空間への埋め込み　$\Sigma^{-1} U^T q_{m,1}$

Latent Semantic Indexing

Aの近似　$A_k = \sum_{1 \le i \le k} \vec{u}_i \sigma_i \vec{v}_i^T$

Frobenius norm

$$|A|_F = \sqrt{\sum_{t,d} A[t,d]^2}$$

すると

$$|A|_F^{\,2} = \sigma_1^2 + \ldots + \sigma_r^2$$

$$\min_{rank(B)=k} |A - B|_F^2 = |A - A_k|_F^2 = \sigma_{k+1}^2 + \ldots + \sigma_r^2$$