領域分割(segmentation)

藤田 一寿

領域分割とは

- 画像を領域に分割する
- 手法
  - クラスタリング
  - ミーンシフト
  - グラフカット
    - 難しいのでこの講義では取り扱わない.

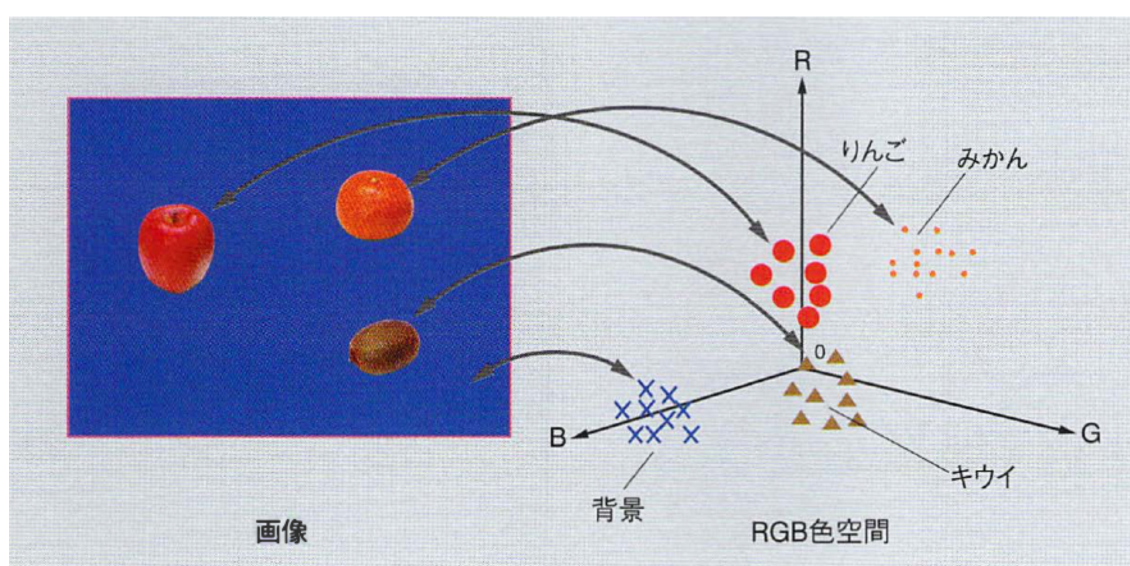
[a] 原画像

[b] 処理結果

クラスタリングによる領域分割

- 画像を色の類似性により領域を分割する
- 類似した色をまとめるときにクラスタリングを用いる

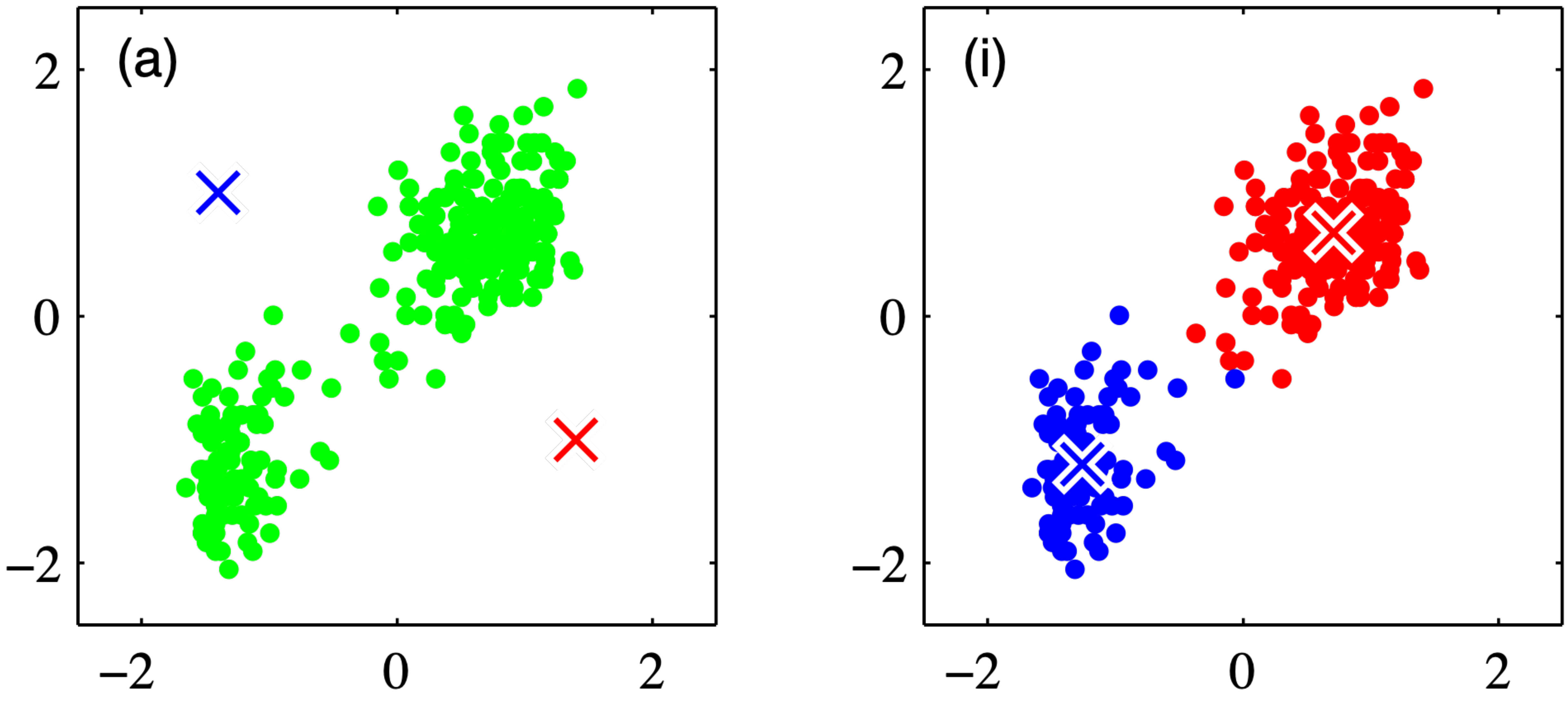
似た色をまとめる



クラスタリングとは

- データをクラスタに自動に分ける
  - クラスタとはデータを何かの基準で集めてできるグループ
  - クラスタの意味は後で考える
  - 教師なし学習
    - 答えはない
  - 基準の例
    - ユークリッド距離，コサイン距離など

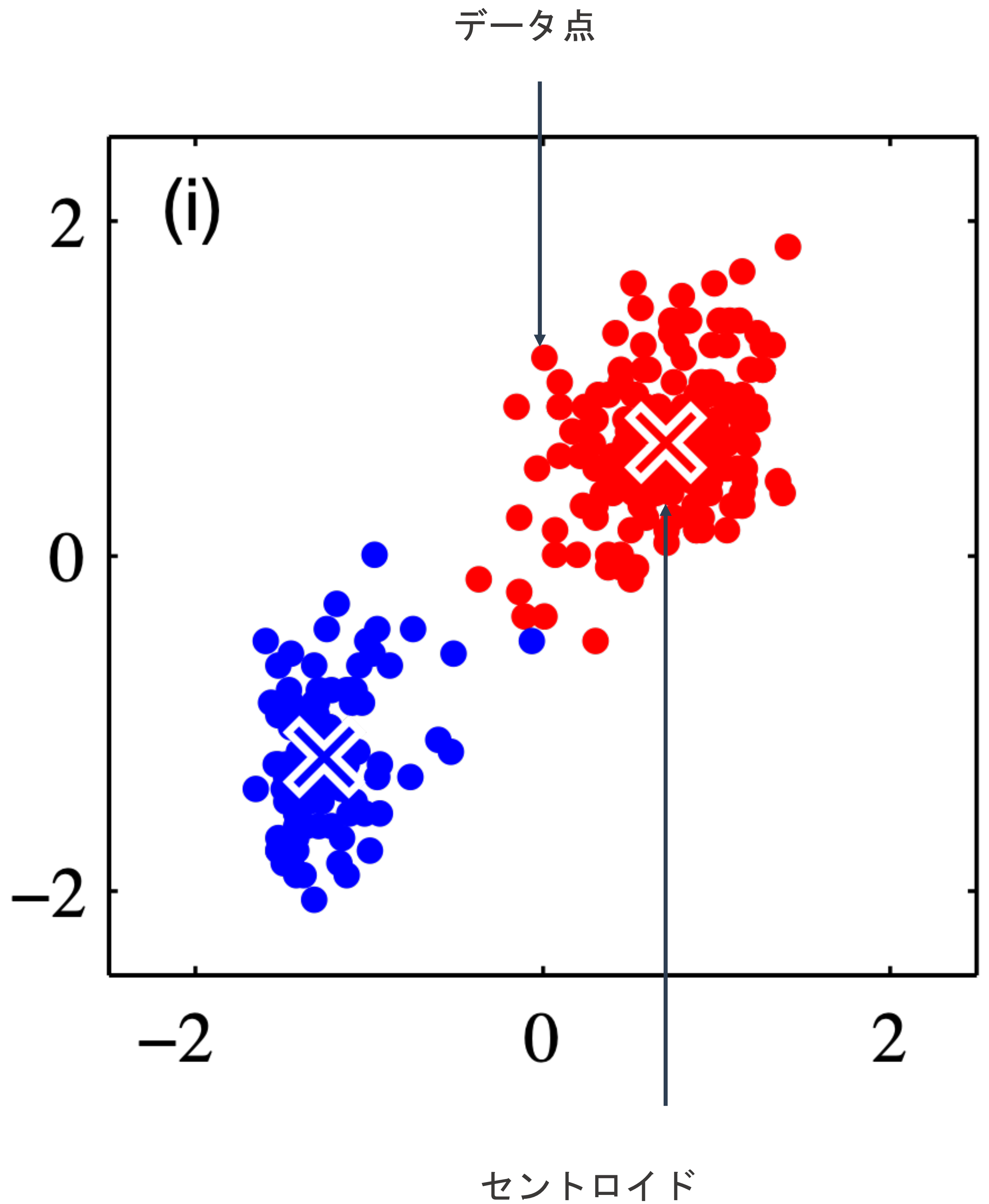
クラスタリング結果の例



自動で2つの塊に分ける

用語

- データ点
- クラスタ
- セントロイド
  - クラスタの中心



k-means

- ユークリッド距離を基準としてクラスタリングする手法
- 目的関数Jを最小化

データ点

セントロイド

$$J = \sum_{i=1}^{N}\sum_{k=1}^{k}\gamma_{ik}||\boldsymbol{x}_i - \boldsymbol{m}_k||^2$$

データ点がどのクラスタに所属しているかどうか

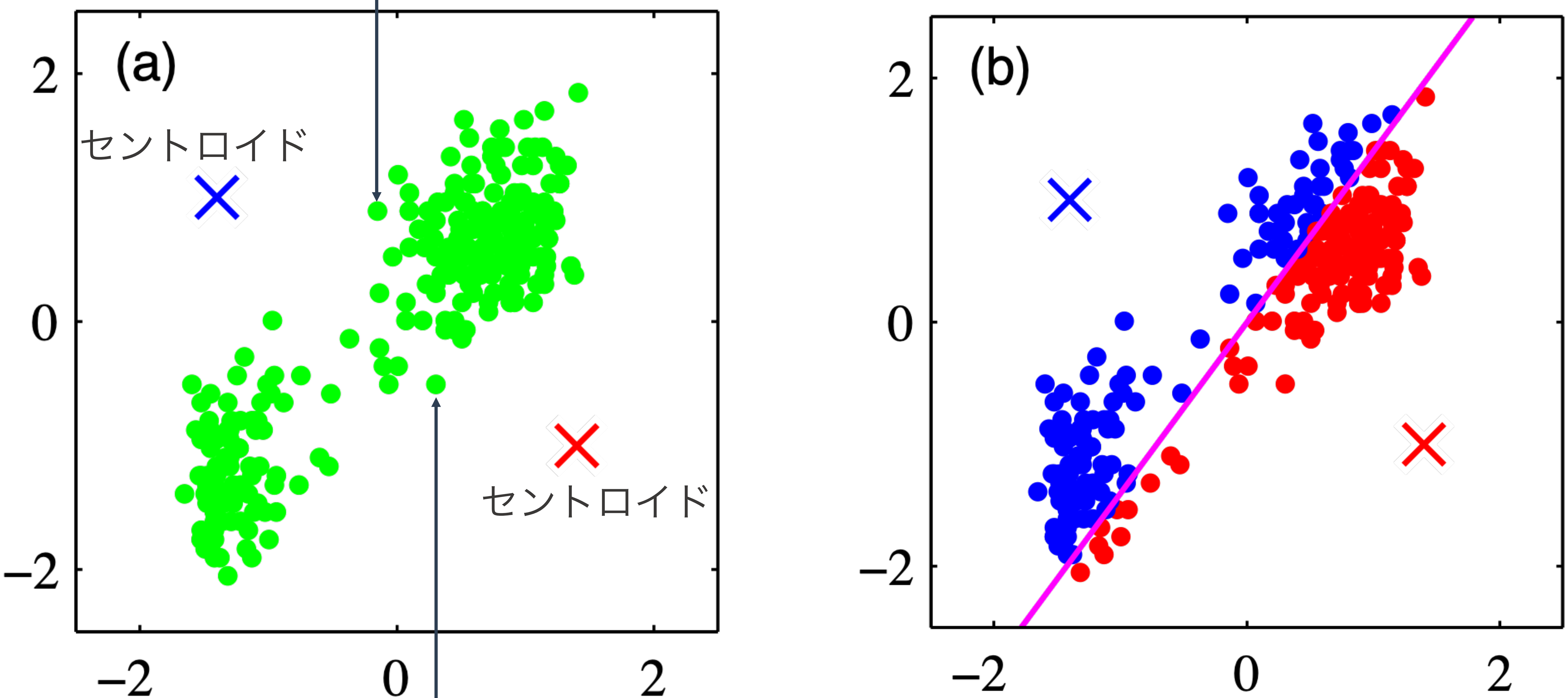
- データ分布が等方ガウス分布であることを想定

目的関数の最小化

1. クラスタ数k，セントロイドmkを初期化
2. すべてのデータ点について，最も近いセントロイドを持つクラスタに所属させる
3. セントロイドを計算する
4. 目的関数の値が収束した場合終了，収束しなかった場合2に戻る

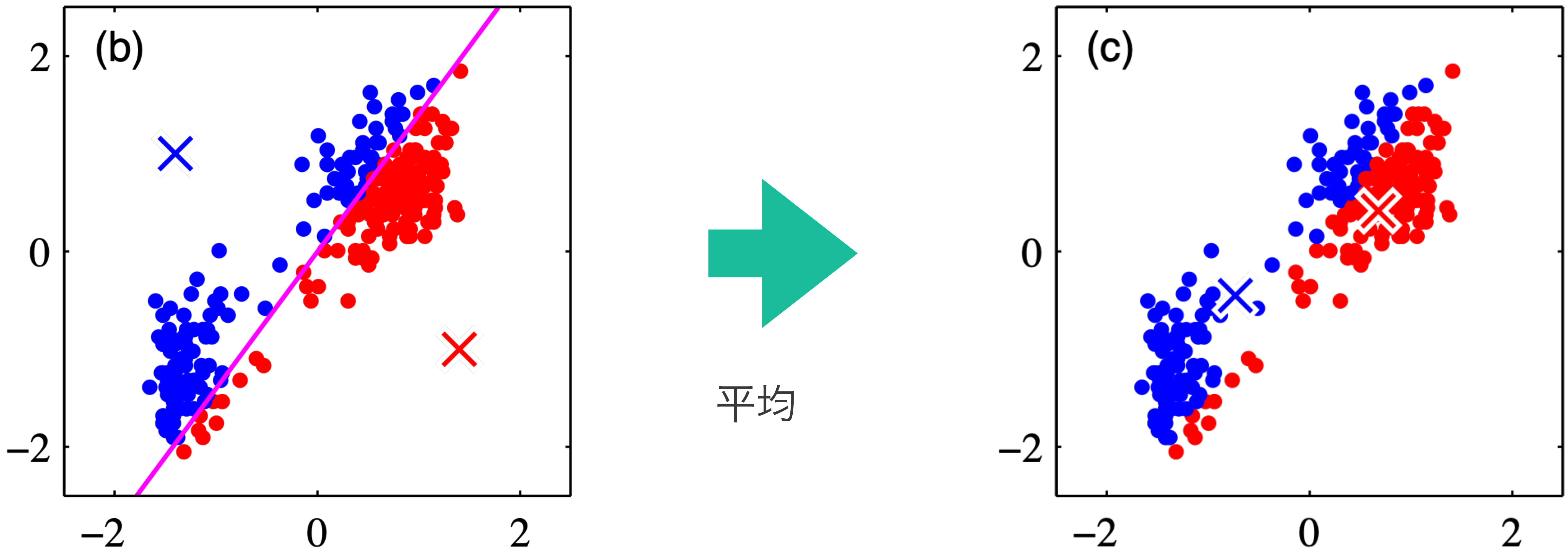
手順2について

このデータ点は青xに近いので青のクラスタに所属

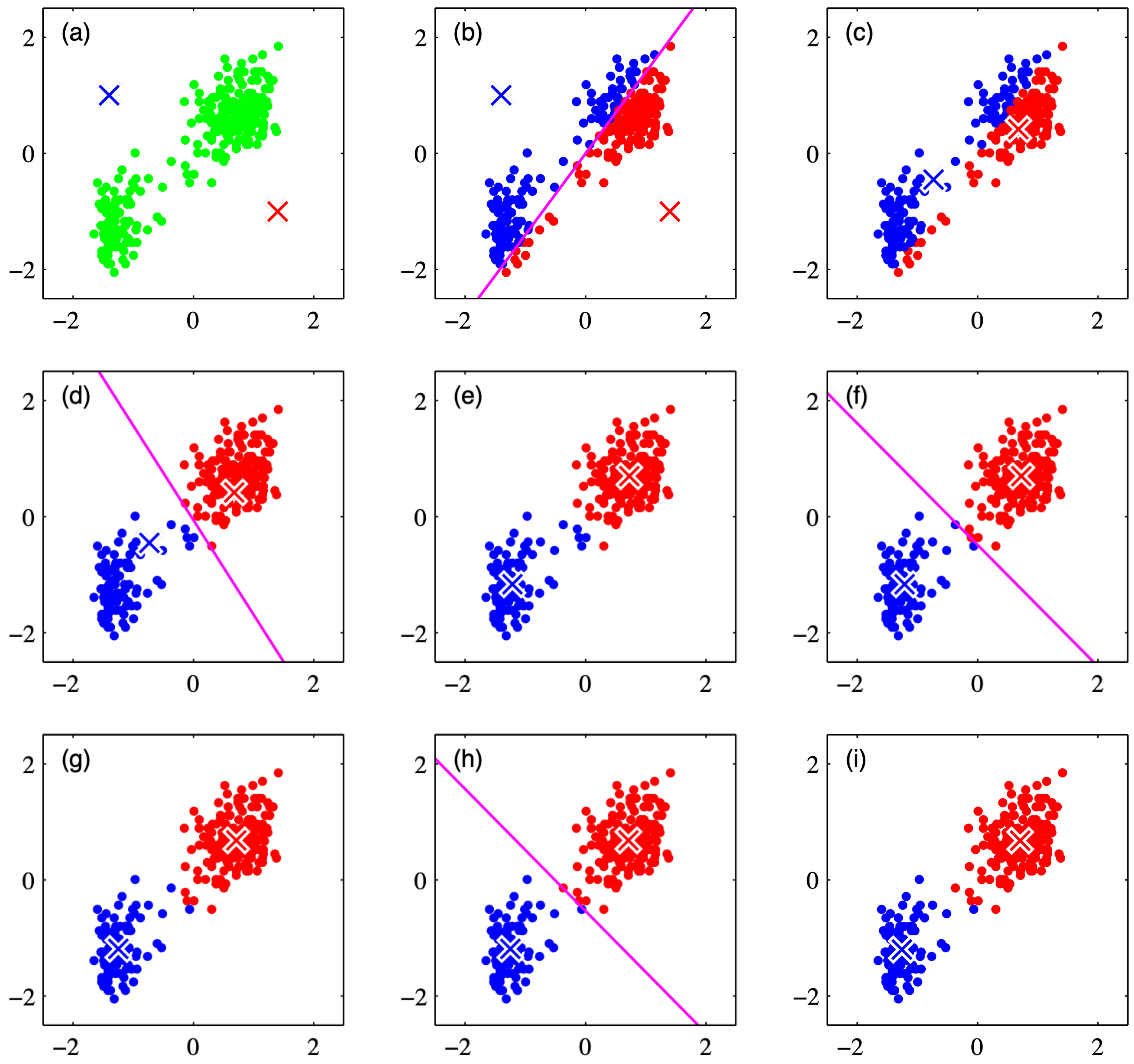


このデータ点は赤xに近いので赤のクラスタに所属

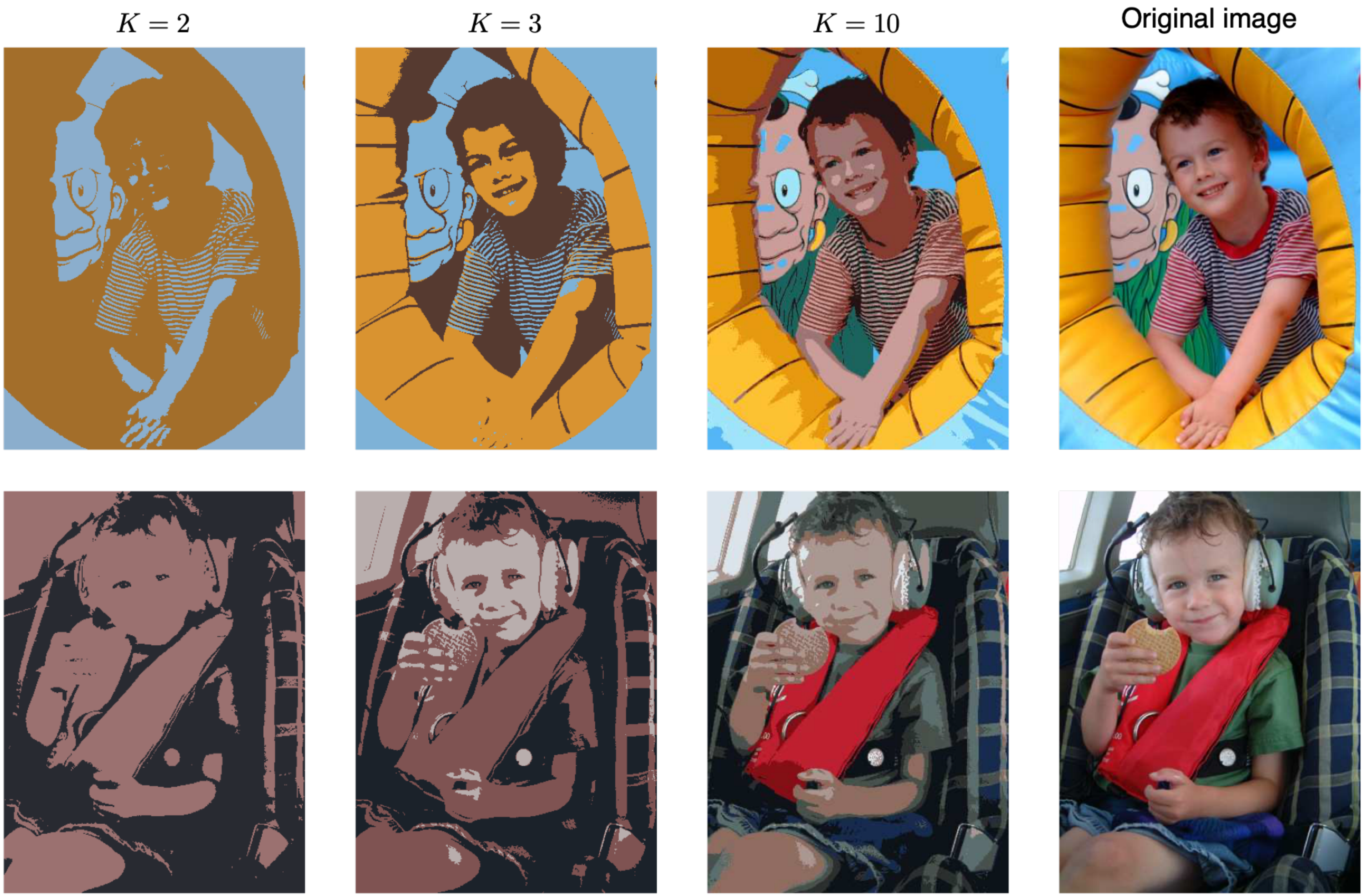
手順３について



新たなクラスタごとに平均を求め，それが新しいセントロイドとなる．


(a)
(b)
(c)
(d)
(e)
(f)
(g)
(h)
(i)
2
0
−2
−2
0
2

k-meansによる領域分割の例



k-meansの利点と欠点

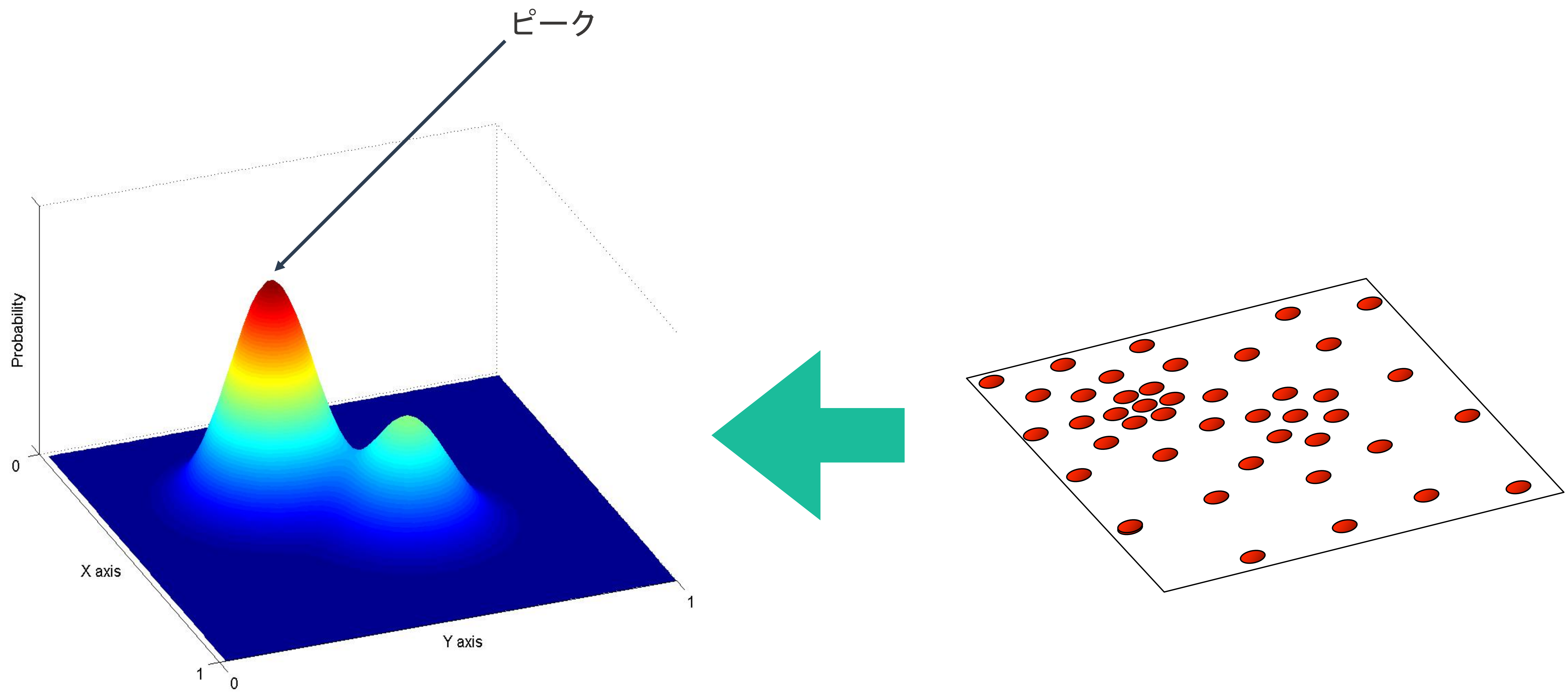
- 利点
  - 実装が簡単
  - アルゴリズムが分かりやすい
- 欠点
  - 外れ値に弱い
  - データの分布が等方ガウス分布のときにのみ適切にクラスタリングできる（何が適切なのかは難しい問題だが）
  - 領域分割に用いる場合は空間情報を用いないため，良い結果を得られない場合がある（何が良いかは難しい）

ミーンシフトによる領域分割

ミーンシフト

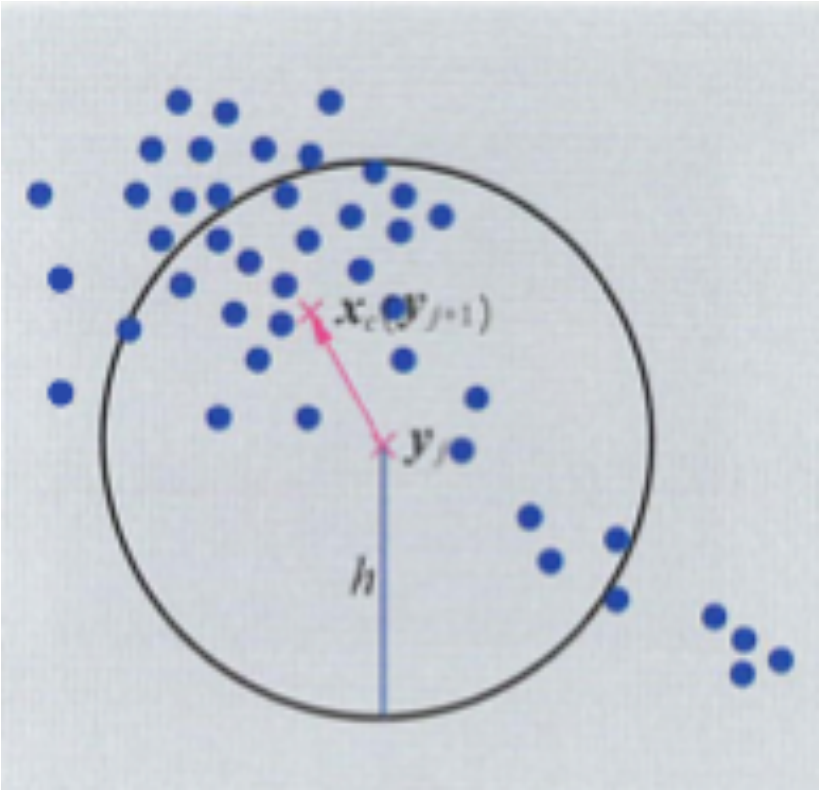
ミーンシフトとは

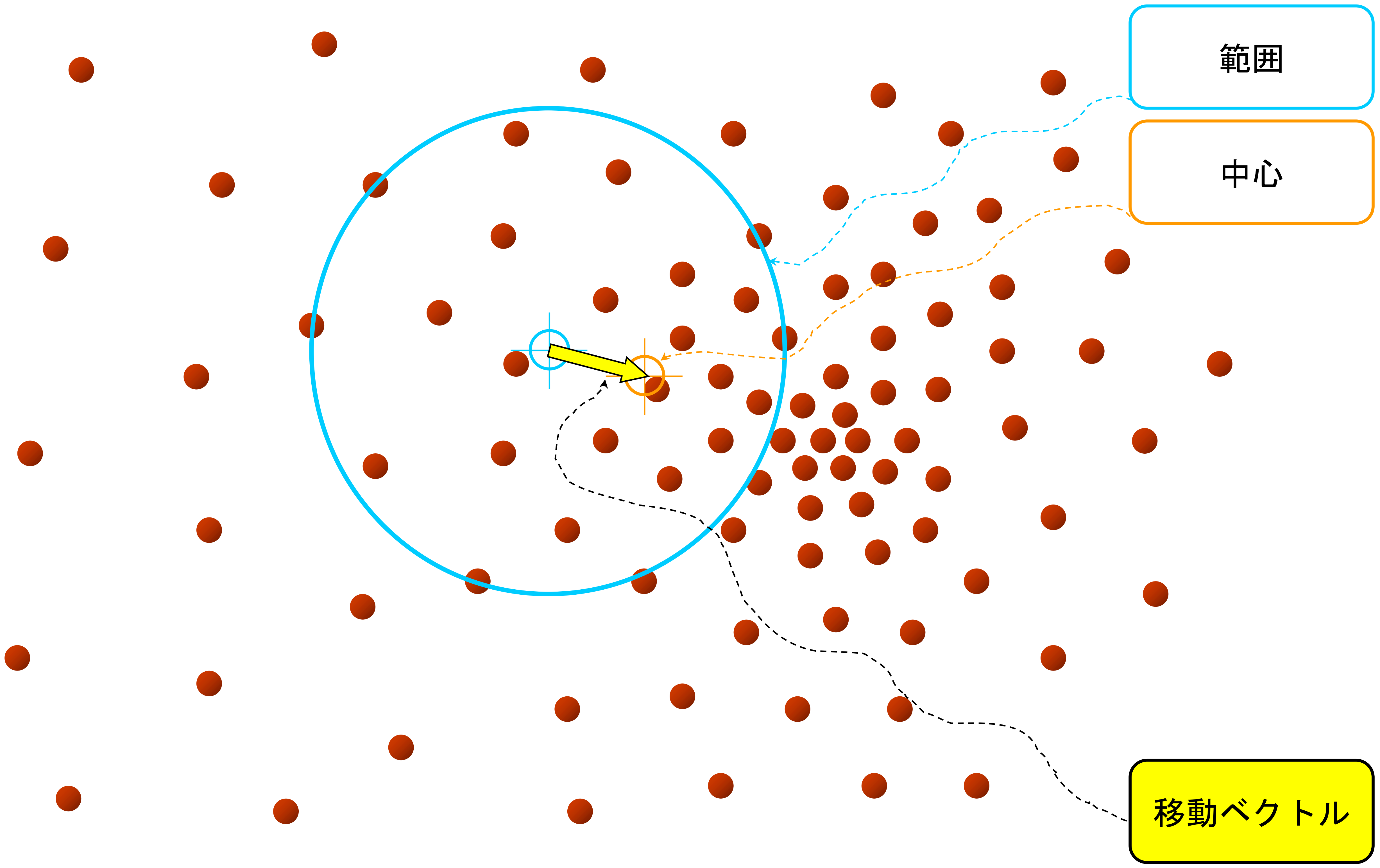
- 分布のピークの場所を求める手法
- クラスタリングに用いる事ができる

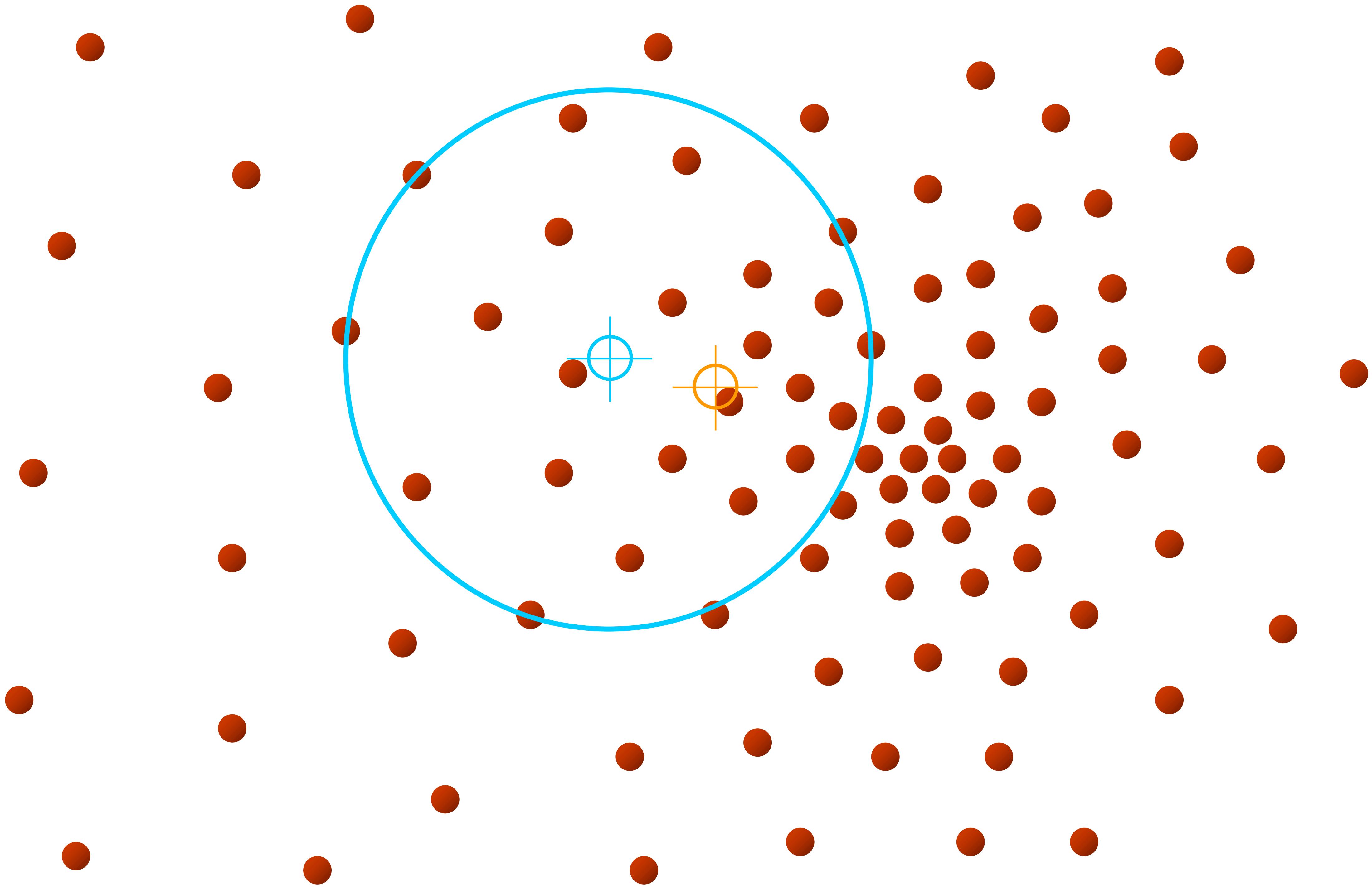


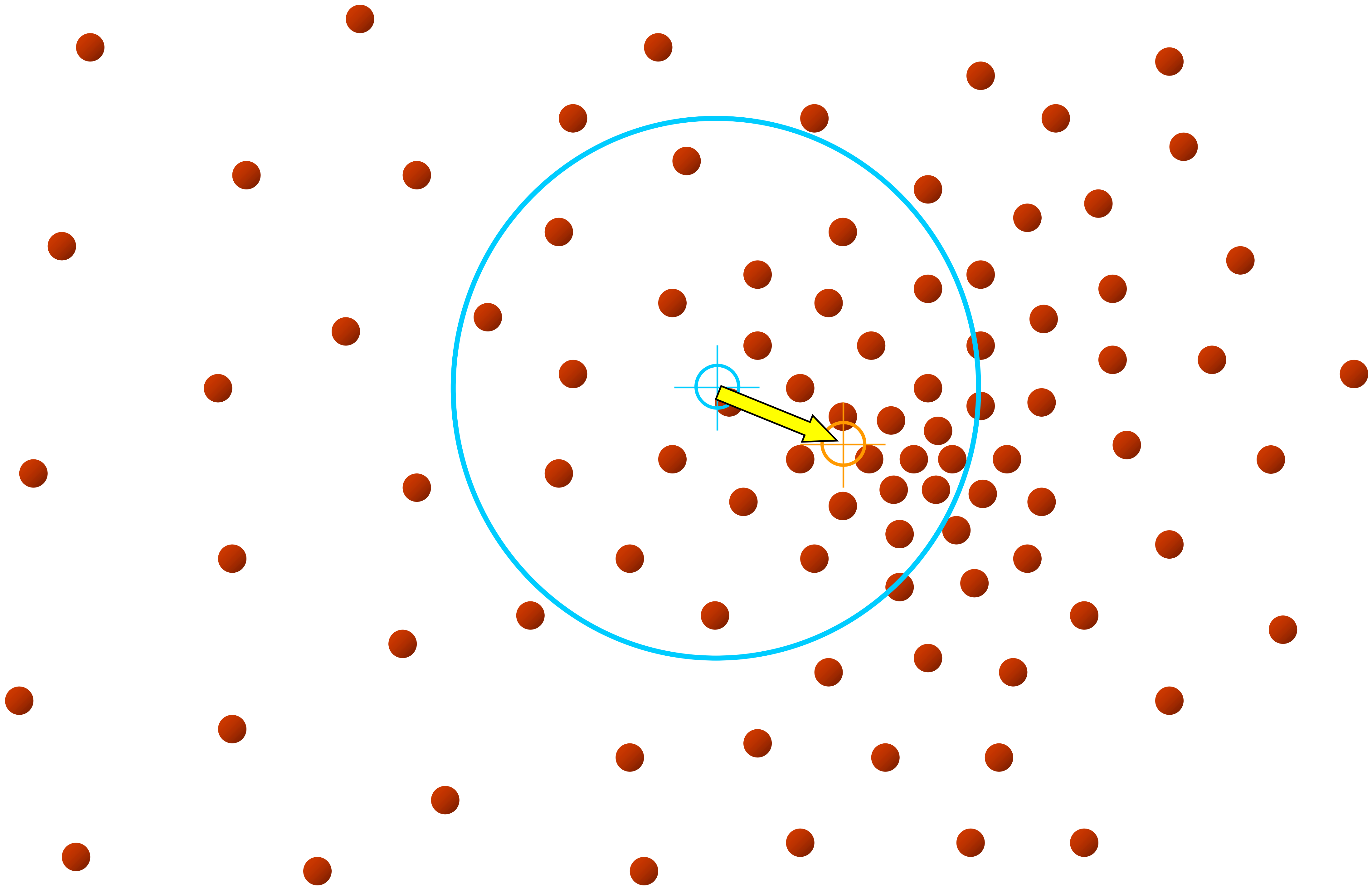
ミーンシフトの計算方法

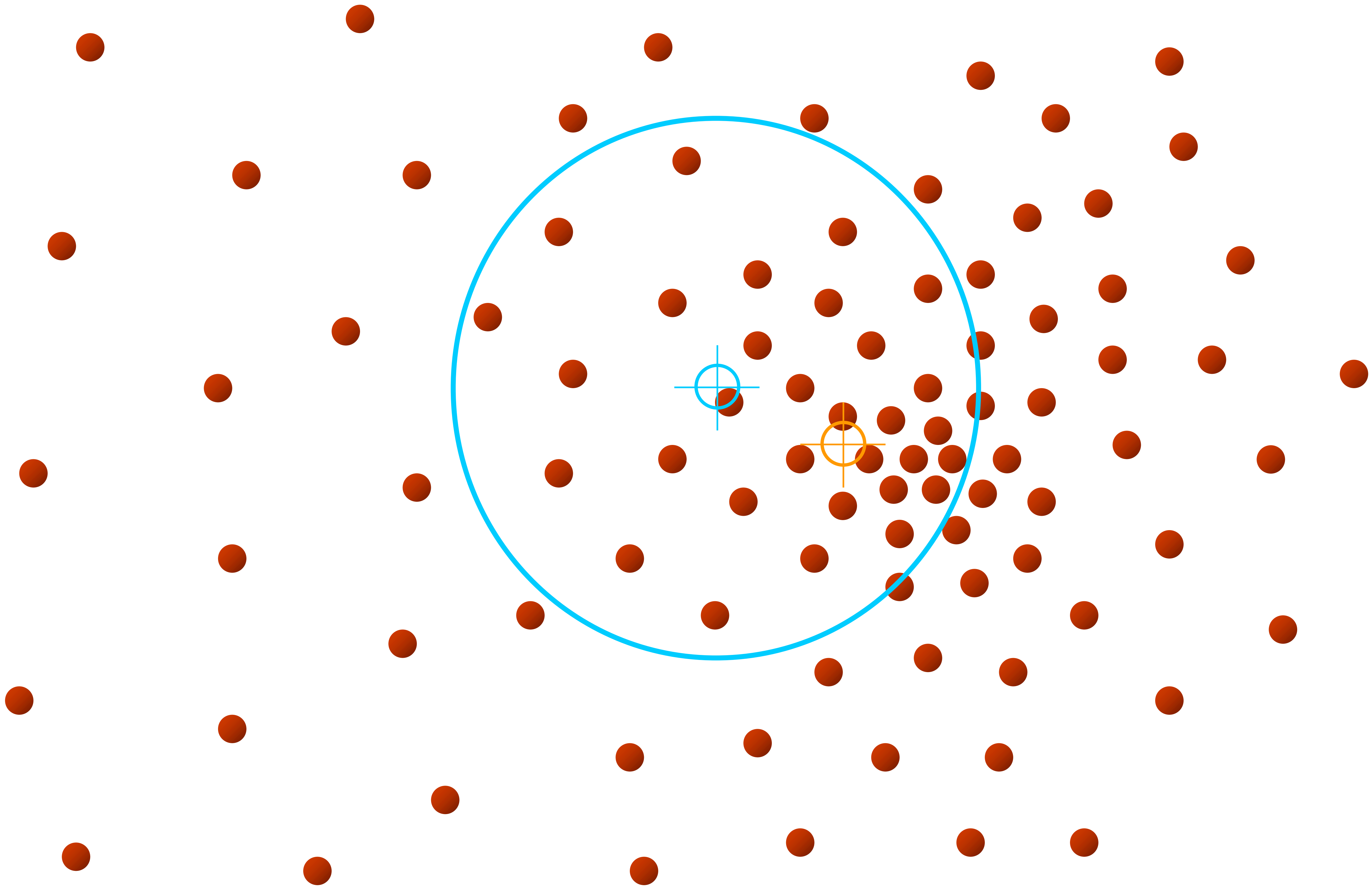
1. 中心x0, y0の半径hの円を考える.
2. 円の中に含まれる点の平均を求め，それを円の中心にする.
3. 平均の変化がある値より小さくなれば終了する．そうでない場合は2の作業を行う.

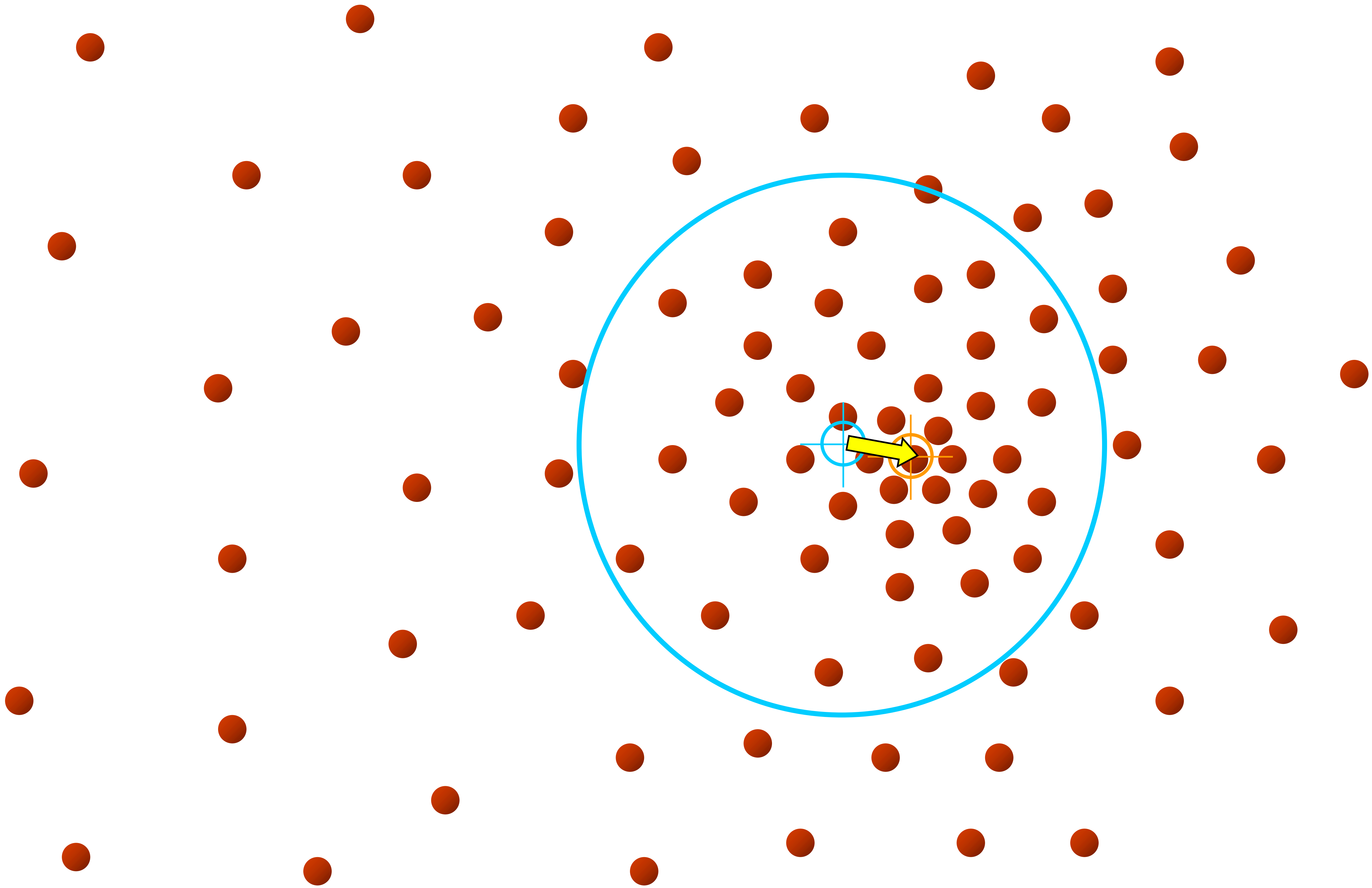


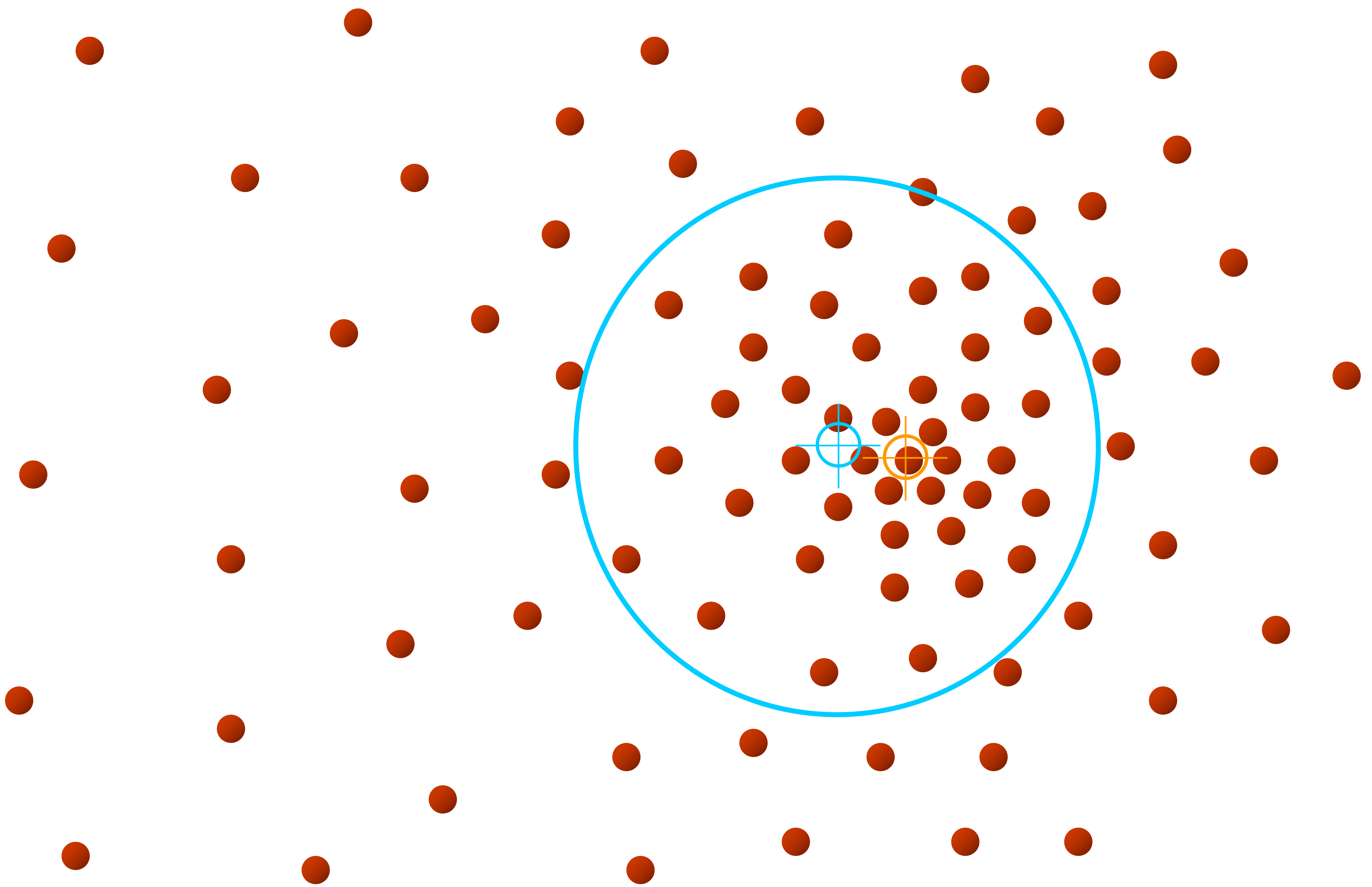

範囲
中心
移動ベクトル

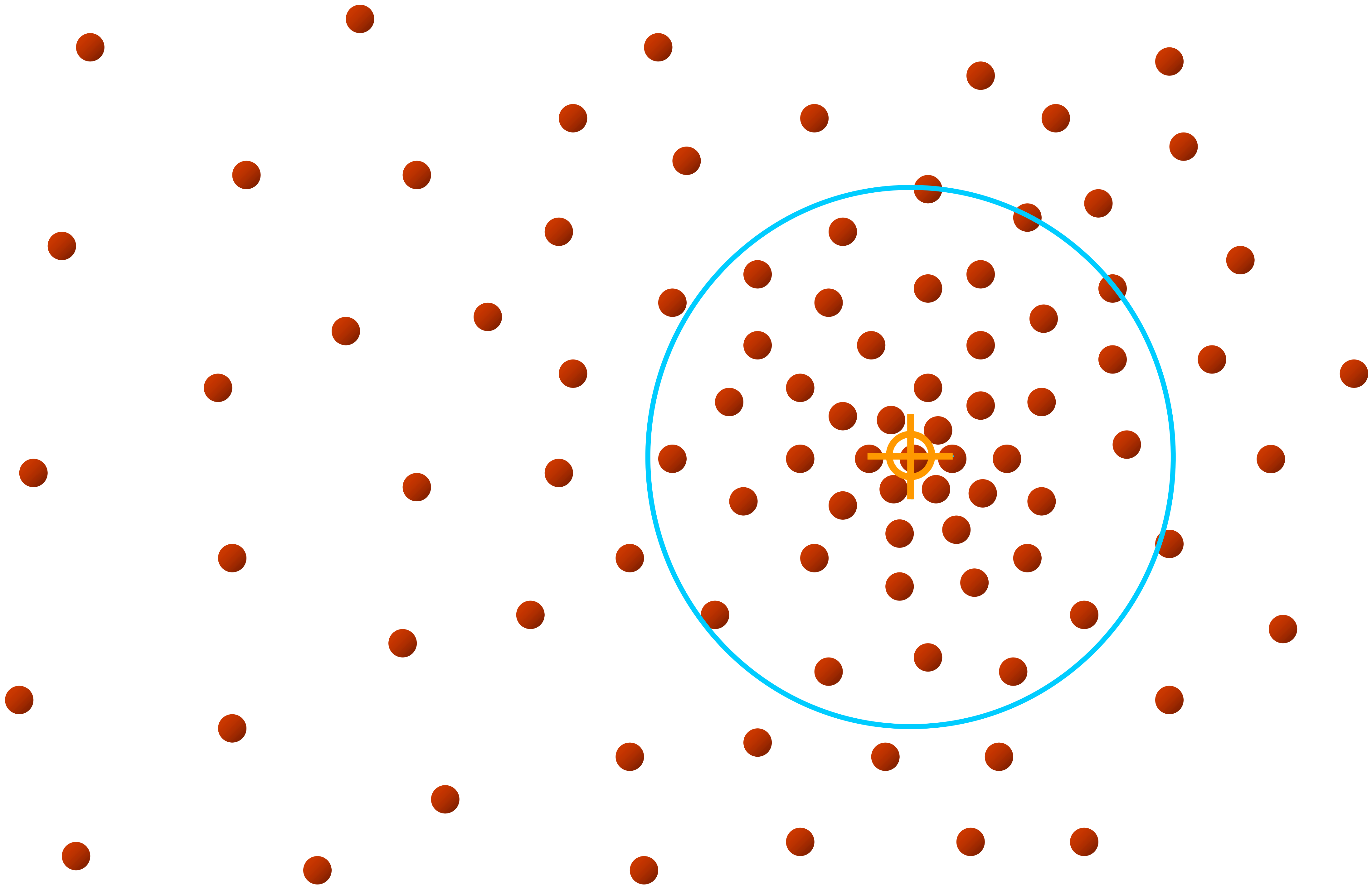












ミーンシフトによるクラスタリング

- データ点の数がM個あるデータのクラスタリングを行う.

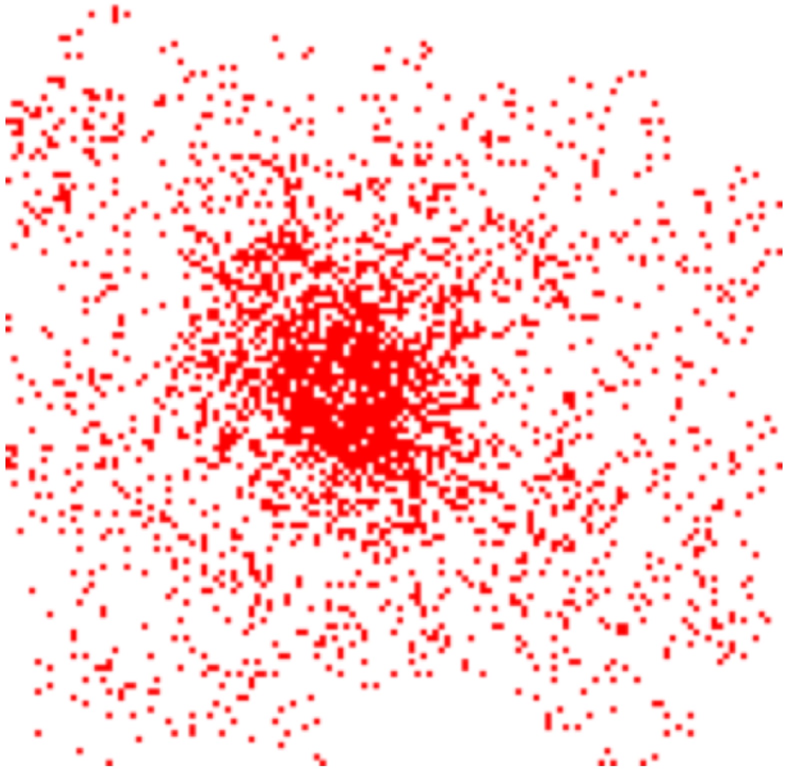
1. M個のxmからミーンシフトを実行し，M個の収束点zmを得る.
2. 距離がh以下の近傍にある収束点zmを同一のグループとする.
3. zmと同じグループにデータ点を所属させる．そのグループがクラスタとなる.

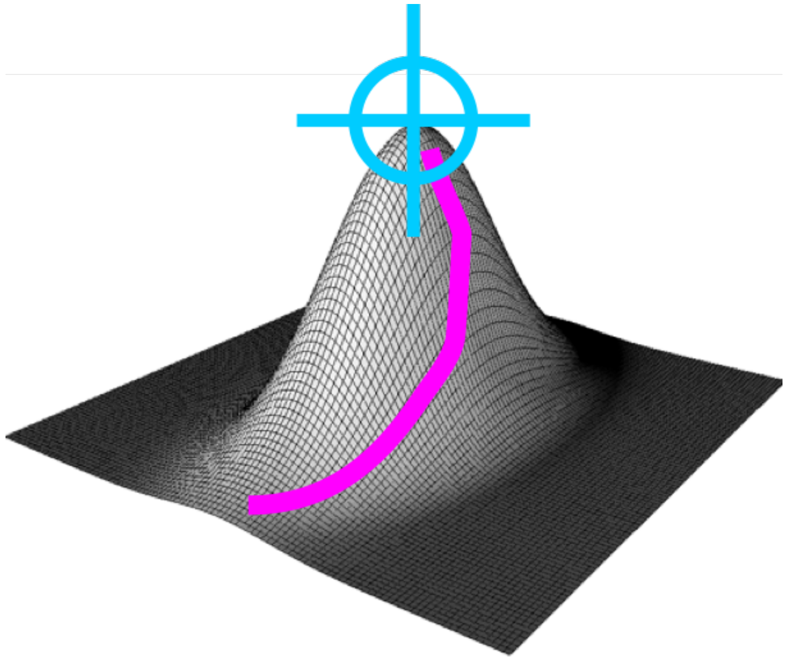
- 利点：クラスタ数を設定する必要はない（近傍もしくは閾値を設定する必要あり）.
- 欠点：計算コストが大きい.

確率密度関数からミーンシフト考える

ミーンシフトとは何なのか

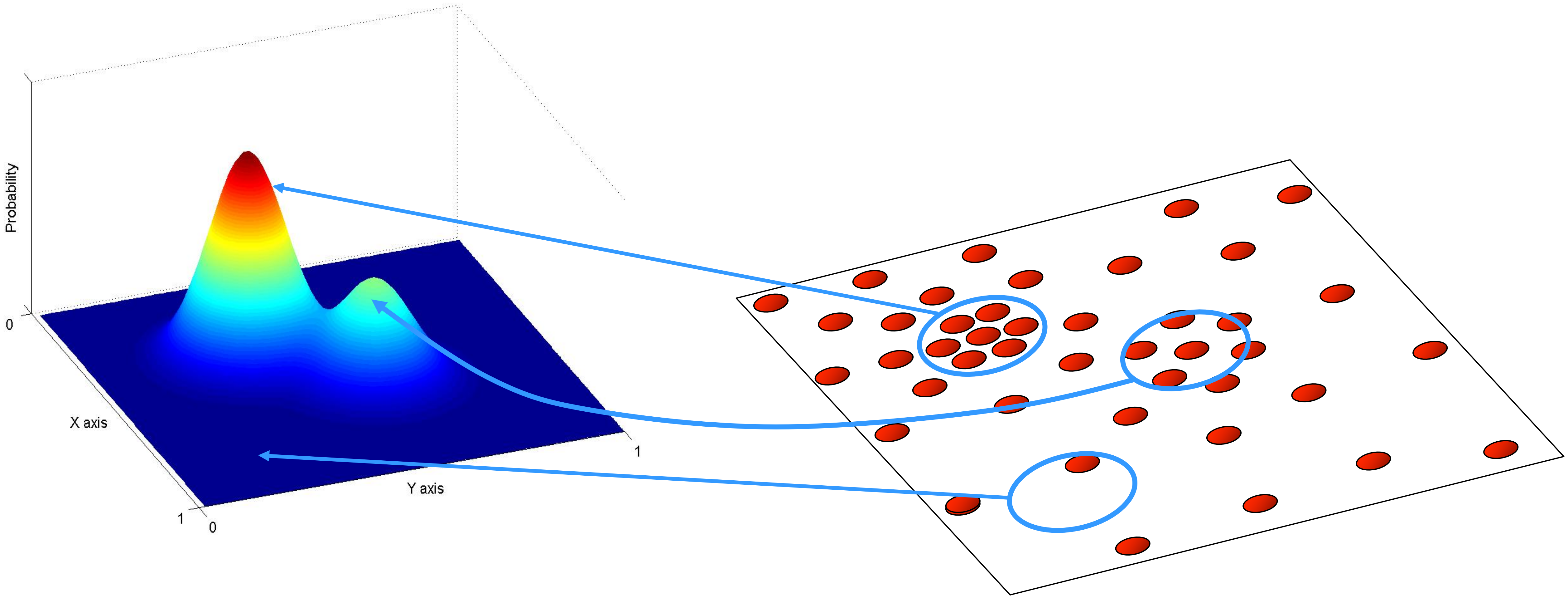
- ある点xを推定された確率密度関数の勾配にもとづいて移動させることで，確率密度関数の極大値を取る場所を求める手法





確率密度関数の推定

確率密度関数をどのように数式で表現するのか

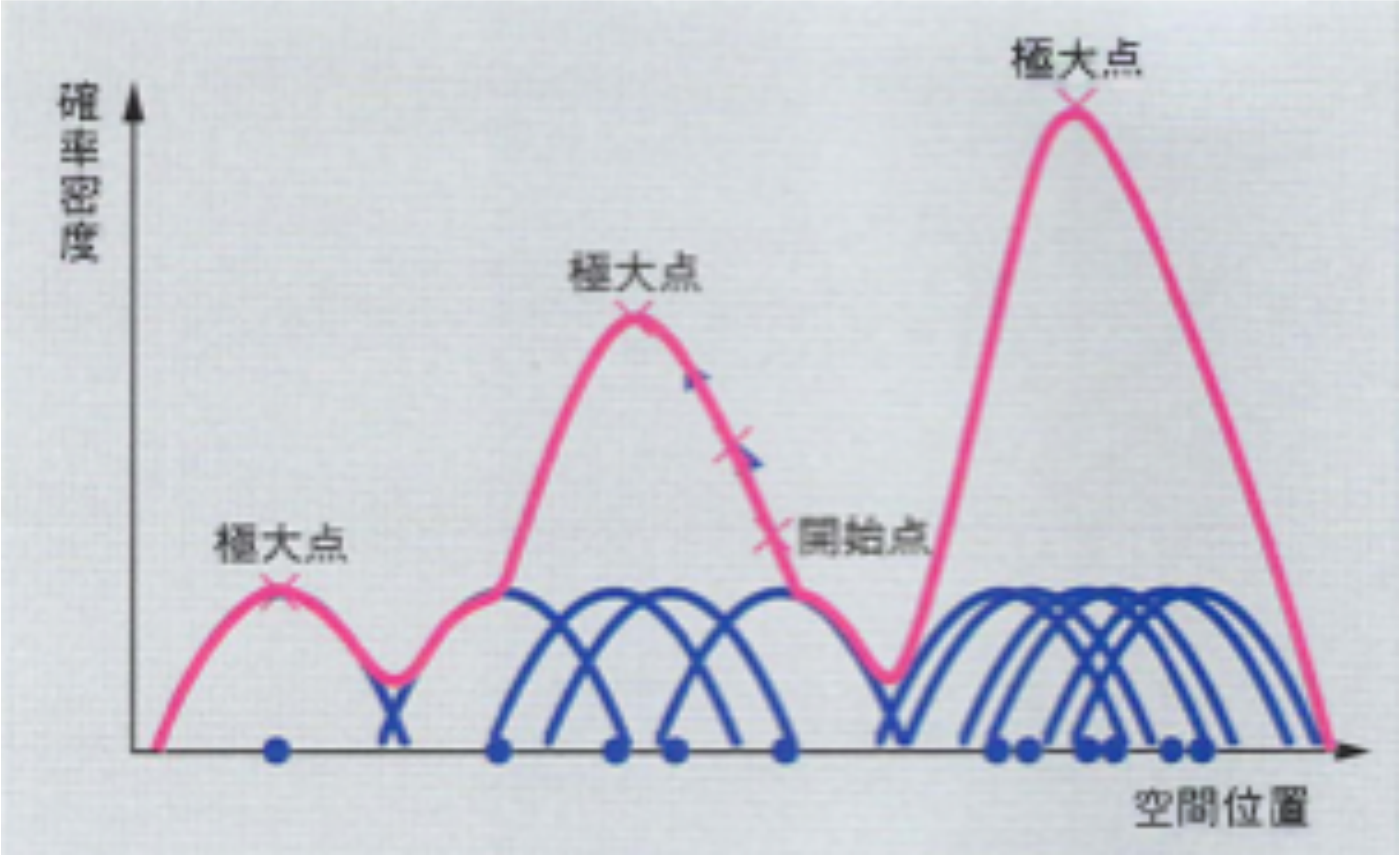


想定される確率密度関数

データの分布

カーネル密度推定

- 何らかの関数（カーネル関数）を足して確率密度関数を作れば良いのでは



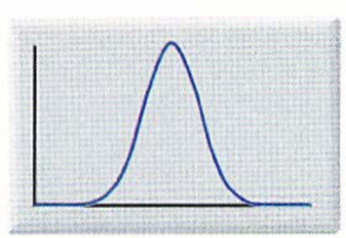
カーネル関数を用いた確率密度関数の推定

$$f(\boldsymbol{x}) = \frac{c_{kd}}{Nh^d}\sum_{i=1}^{N} k\left(\left|\frac{\boldsymbol{x}-\boldsymbol{x}_i}{h}\right|^2\right)$$

- 上記の式で確率密度関数を推定する．c_{kd}は正則化項，Nはデータ点の数，hはパラメタ，dは次元，kはカーネル関数Kのプロファイル．$K(t)=k(|t|^2)$.

ガウシアン（Gaussian）カーネル

$$K(t) = \exp\left(-\frac{|t|^2}{2}\right)$$

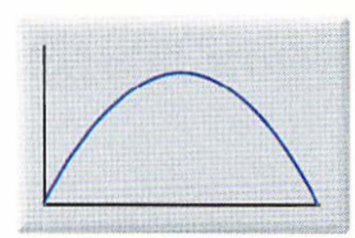


プロファイル

$$k(t) = \exp\left(-\frac{t}{2}\right)$$

エパネックニコフ（Epanechnikov）カーネル

$$K(t) = 1-|t|^2 \quad (|t| \leqq 1),\ 0(other)$$



プロファイル

$$k(t) = 1-t \qquad (0 \leqq t \leqq 1),\ 0(t>1)$$

確率密度関数の勾配を求める

$$f(\boldsymbol{x}) = \frac{c_{kd}}{Nh^d}\sum_{i=1}^{N} k\left(\left\|\frac{\boldsymbol{x}-\boldsymbol{x}_i}{h}\right\|^2\right)$$

- 勾配を計算する

$$\nabla f(\boldsymbol{x}) = \frac{2c_{kd}}{Nh^{d+2}}\sum_{i=1}^{N} k'\left(\left\|\frac{\boldsymbol{x}-\boldsymbol{x}_i}{h}\right\|^2\right)(\boldsymbol{x}-\boldsymbol{x}_i)$$

$g(t) = -k'(t),\ g_i = g\left(\left\|\frac{\boldsymbol{x}-\boldsymbol{x}_i}{h}\right\|^2\right)$ とおくと

$$\nabla f(\boldsymbol{x}) = \frac{2c_{kd}}{Nh^{d+2}}\sum_{i=1}^{N} (g_i\boldsymbol{x}_i - g_i\boldsymbol{x})$$

- さらに変形すると

$$\nabla f(\boldsymbol{x}) = \frac{2c_{kd}}{Nh^{d+2}}\left\{\sum_{i=1}^{N} g_i\right\}\left\{\frac{\sum_{i=1}^{N} g_i\boldsymbol{x}_i}{\sum_{i=1}^{N} g_i} - \boldsymbol{x}\right\}$$

勾配法によるミーンシフト

1. 点yを初期化する.
2. 勾配法によりyを更新する.

$$\boldsymbol{y}_{j+1} \leftarrow \boldsymbol{y}_j + \left(\frac{\sum_{i=1}^{N} g_i \boldsymbol{x}_i}{\sum_{i=1}^{N} g_i} - \boldsymbol{y}_j \right)$$

3. yの変化がある値より小さくなれば終了する．そうでない場合は2の作業を行う.

ミーンシフトによる領域分割

ミーンシフトを用いた領域分割

- N画素あるカラー画像を考える．各画素は場所xと色vの要素を持つと考えると，各画素は5次元のデータ(x, v)であると考えられる．

- この時，カーネル密度推定を次のように定義する．バンド幅を位置と色(hs, hr)それぞれについて持つとする．また，位置をxs，色をxrとする．

$$f(x) = \frac{c}{N h_s^2 h_r^3} \sum_{i=1}^{N} k\left(\left\| \frac{x^s - x_i^s}{h_s} \right\|^2 \right) k\left(\left\| \frac{x^r - x_i^r}{h_r} \right\|^2 \right)$$

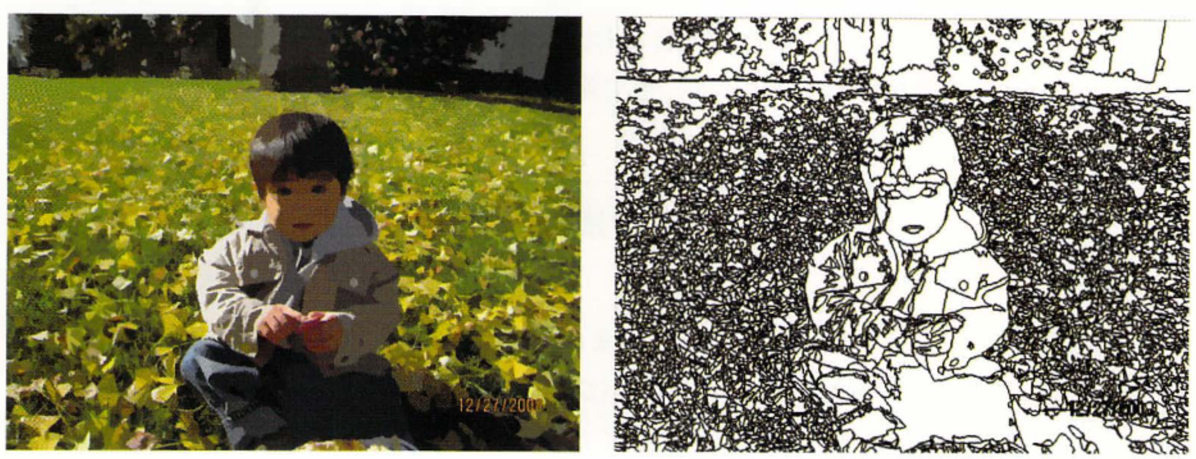
- 密度推定からミーンシフトに用いる更新式が得られる．

$$y_{j+1}^s = \frac{\sum_{i=1}^{N} g_i^s x_i^s}{\sum_{i=1}^{N} g_i^s}, \quad y_{j+1}^r = \frac{\sum_{i=1}^{N} g_i^r x_i^r}{\sum_{i=1}^{N} g_i^r}$$

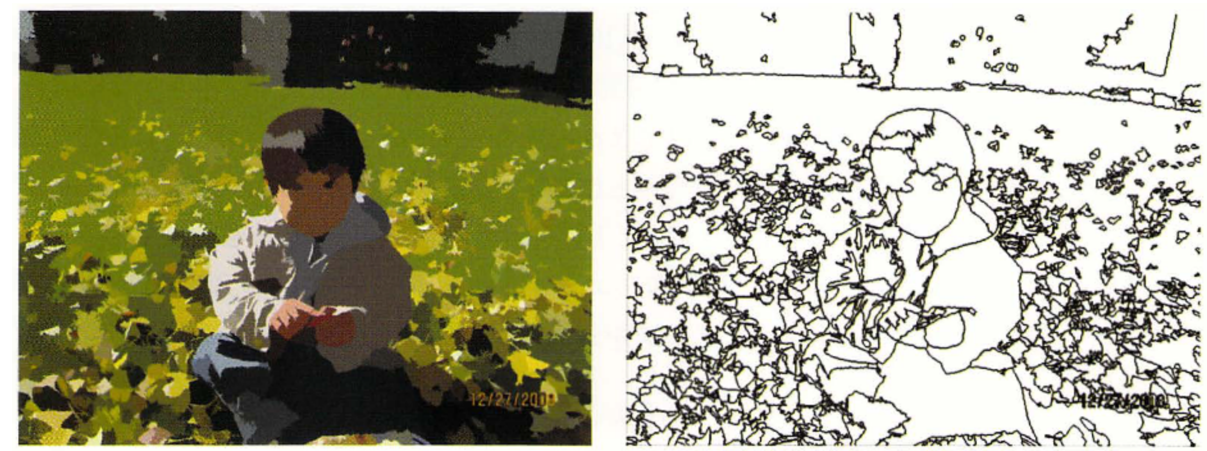
- 上記の更新式を用い，ミーンシフトによるクラスタリングを行う．

例

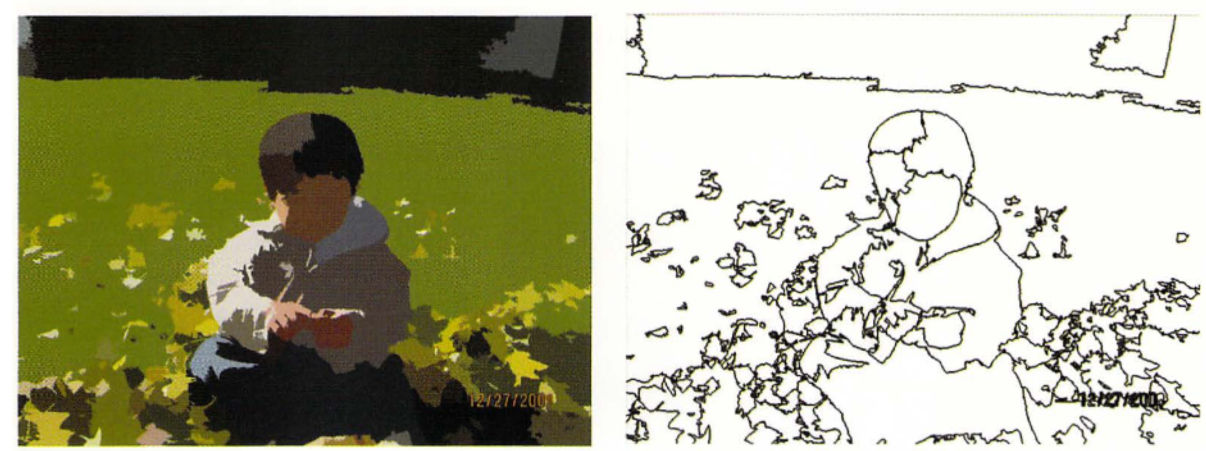
[a] 原画像



[b] $(h_s, h_r)=(7, 6.5)$

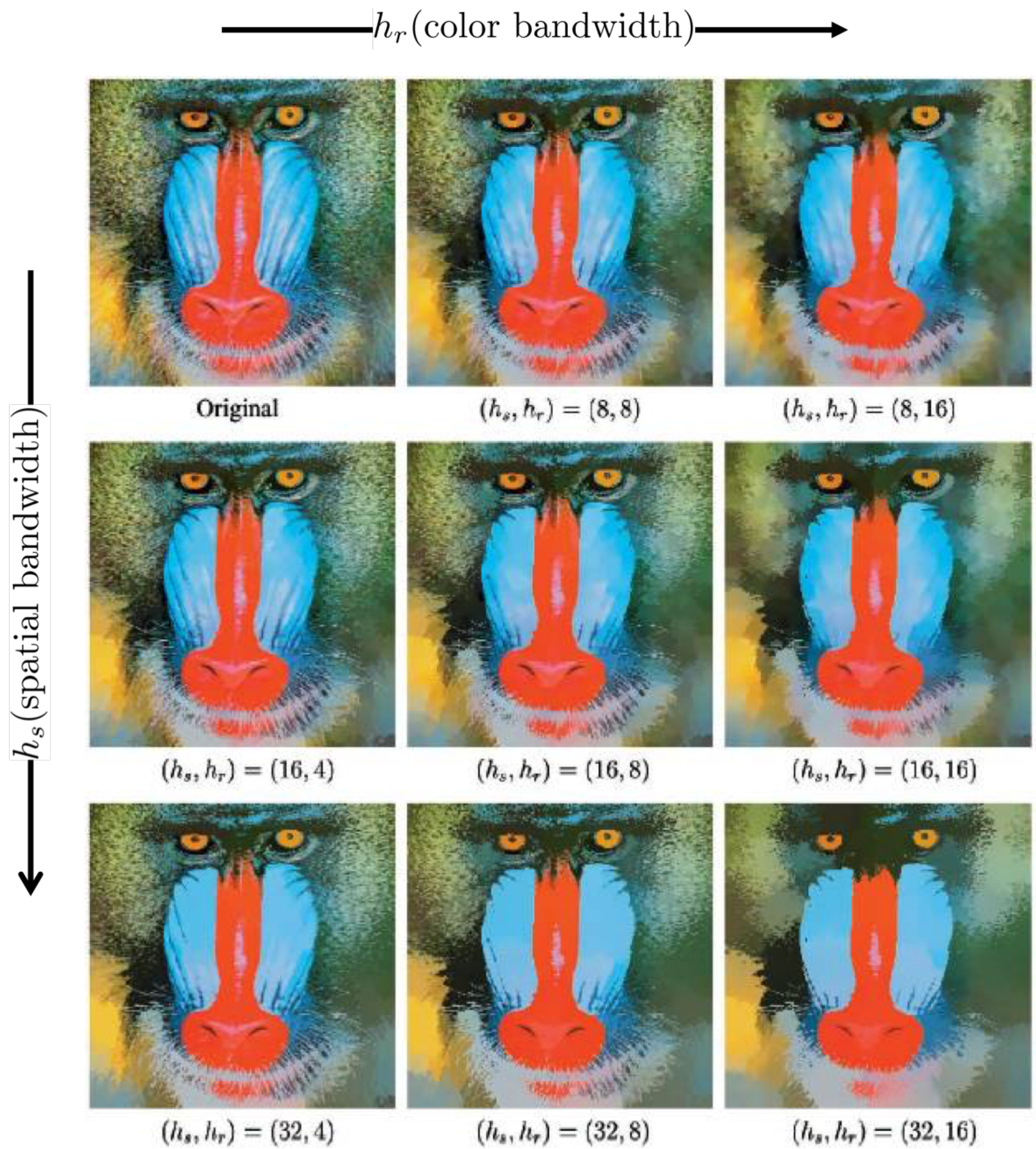


[c] $(h_s, h_r)=(14, 13)$



[d] $(h_s, h_r)=(21, 19.5)$

例


h_r(color bandwidth)
h_s(spatial bandwidth)
Original
$(h_s, h_r) = (8, 8)$
$(h_s, h_r) = (8, 16)$
$(h_s, h_r) = (16, 4)$
$(h_s, h_r) = (16, 8)$
$(h_s, h_r) = (16, 16)$
$(h_s, h_r) = (32, 4)$
$(h_s, h_r) = (32, 8)$
$(h_s, h_r) = (32, 16)$