Clustering

Outline

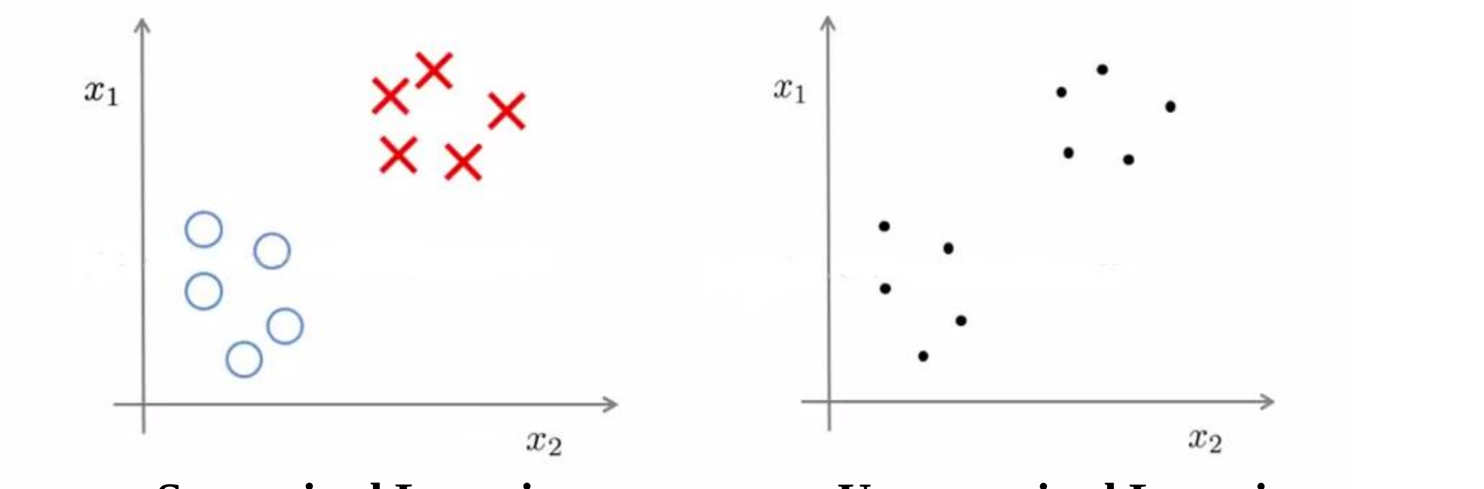
- □ Definition
- □ Distance Calculation
- □ Algorithms
  - K-means
  - Mixture of Gaussian
  - DBSCAN
  - AGNES
- □ Performance Measure

Outline

- □ Definition
- □ Distance Calculation
- □ Algorithms
  - K-means
  - Mixture of Gaussian
  - DBSCAN
  - AGNES
- □ Performance Measure

Definition

- Supervised learning VS. Unsupervised learning
- In supervised learning, we know something(label or value) about data(X={x1,…,xn}, Y={y1,…,yn}known), learn y = f (x)
- In unsupervised learning, we know nothing about the data(X known,Y unknown).

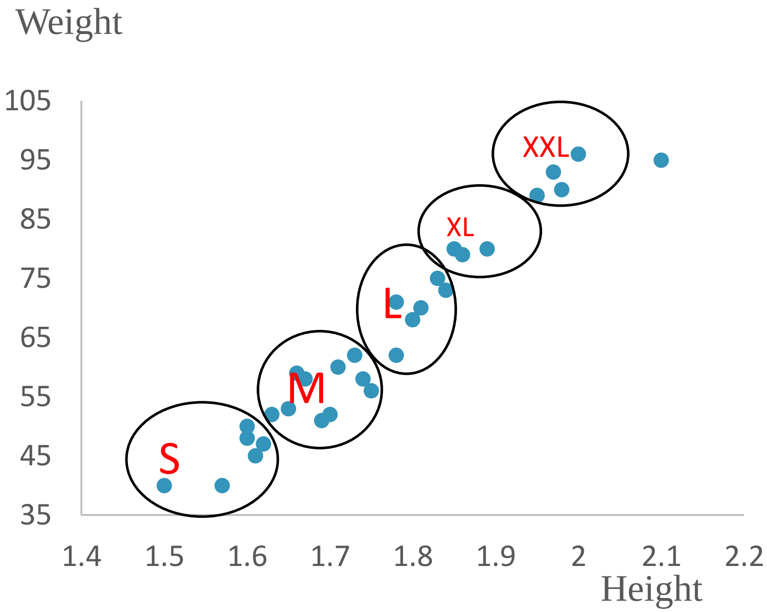


Supervised Learning **Unsupervised Learning**

https://blog.csdn.net/sp_programmer/article/details/42084709

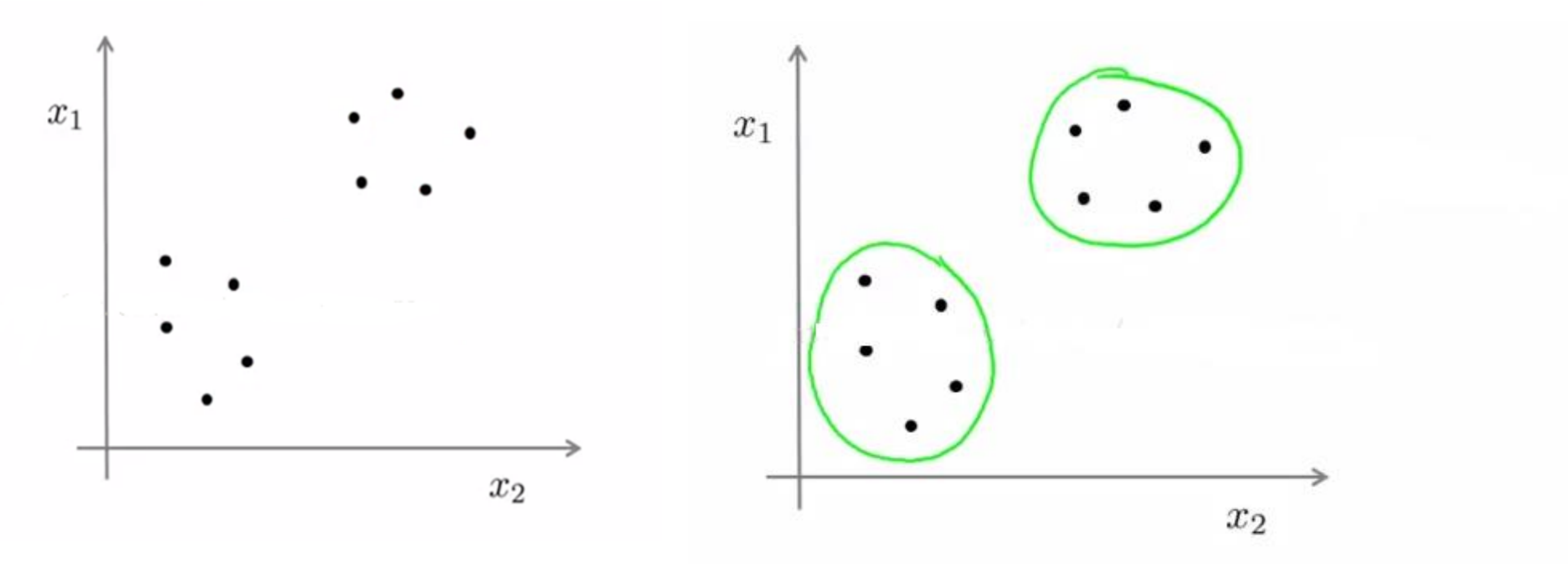
Definition

- □ Clustering is used for unsupervised learning.
- □ T-shirt sizing(S? M? L? XL? XXL…)



Definition

- Clustering is the task of grouping a set of objects in such a way that objects in the same group(inter-group) are more similar to each other than to those in other groups(intra-group) .



https://blog.csdn.net/sp_programmer/article/details/42084709

Outline

□ Definition

□ Distance Calculation

□ Algorithms

- K-means
- Mixture of Gaussian
- DBSCAN
- AGNES

□ Performance Measure

Distance Calculation

□ Similarity measure

- ➢ Similarity measure quantifies the similarity between two objects. The usual method is to calculate distance between samples.

□ Distance Calculation

- ➢ Distance is the inverse of Similarity in some sense.
- ➢ The greater the distance the smaller the similarity.

Distance Calculation

□Minkowski distance

$$dist_{mk}(x_i\,,x_j) = (\sum_{u=1}^{n}\left|x_{iu} - x_{ju}\right|^p)^{\frac{1}{p}}$$

where, $x_i = (x_{i1}; x_{i2}; \dots x_{in}), x_j = (x_{j1}\ x_{j2}; \dots x_{jn})$

□ Euclidean distance (p=2)

□ Manhattan distance (p=1)

x_1=[2,1]

x_2=[3,3]

$\xrightarrow{\text{Manhattan distance}}$ d=(|2-3|^1+|1-3|^1)^1=3

$\xrightarrow{\text{Euclidean distance}}$ d=(|2-3|^2+|1-3|^2)$^{1/2}$=$\sqrt{5}$

Other similarity measurement

- ✓ Chebyshev Distance D=max($|x_1 - x_2|,|y_1 - y_2|$)
- ✓ Cosine $$\cos(\theta) = \frac{\sum_{i=1}^{n}(x_i * y_i)}{\sqrt{\sum_{i=1}^{n}(x_i)^2} * \sqrt{\sum_{i=1}^{n}(y_i)^2}}$$
- ✓ Hamming Distance d(x,y)=$\sum X[i] \oplus Y[i]$
- ✓ Jaccard Distance $$\frac{|A \cup B| - |A \cap B|}{|A \cup B|}$$
- ✓ Correlation coefficient and Correlation distance
- ✓ …

Outline

- □ Definition
- □ Distance Calculation
- □ Algorithms
  - K-means
  - Mixture of Gaussian
  - DBSCAN
  - AGNES
- □ Performance Measure

K-means Algorithm

□ Randomly select K samples as the centroids of clustering.

□ Calculate the distances of each sample from the K centroid points.

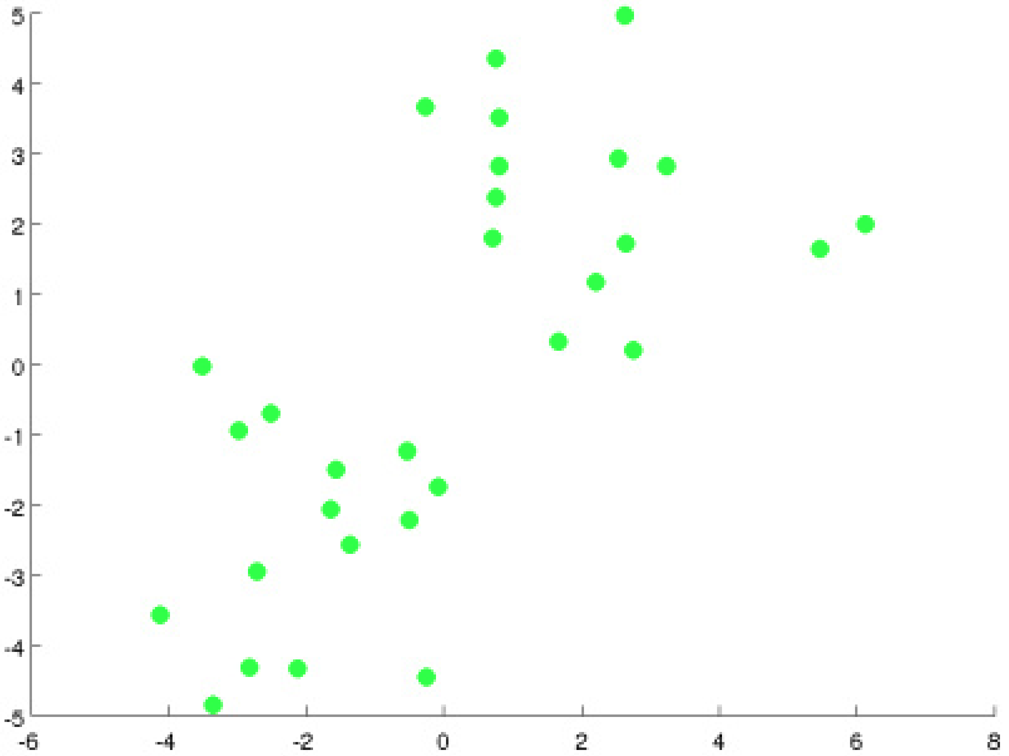
□ Select the cluster centroid $c_i \; (i = 1,2 \ldots K)$ with the smallest distance to divide the cluster.

□ Re-determine the centroids.

□ Iterate until it converges.

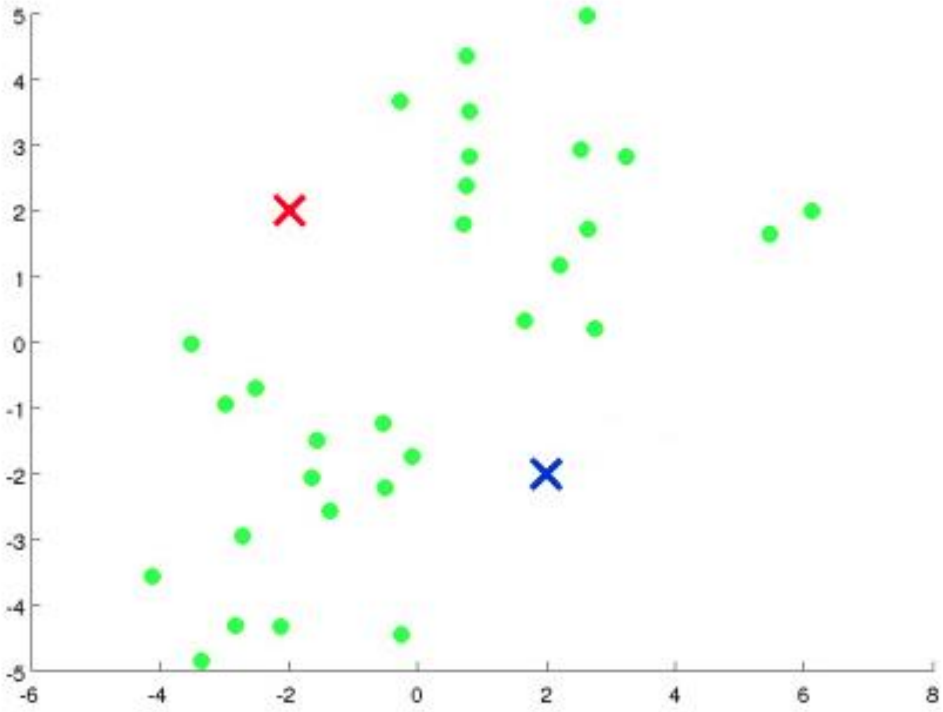
K-means Algorithms

- □ This is unprocessed data without labels.



K-means Algorithms

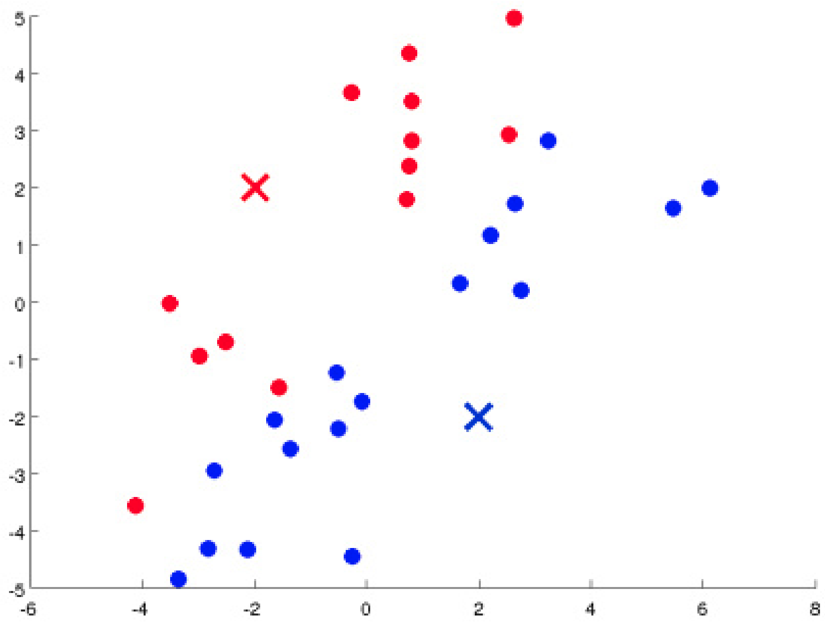
- □ By experience, set k=2 .
- □ Randomly select two points as cluster centroids.



K-means Algorithms

- □ For all green samples, compute distances from the blue point and red point.
- □ Choose the smallest distance and assign it to one cluster.

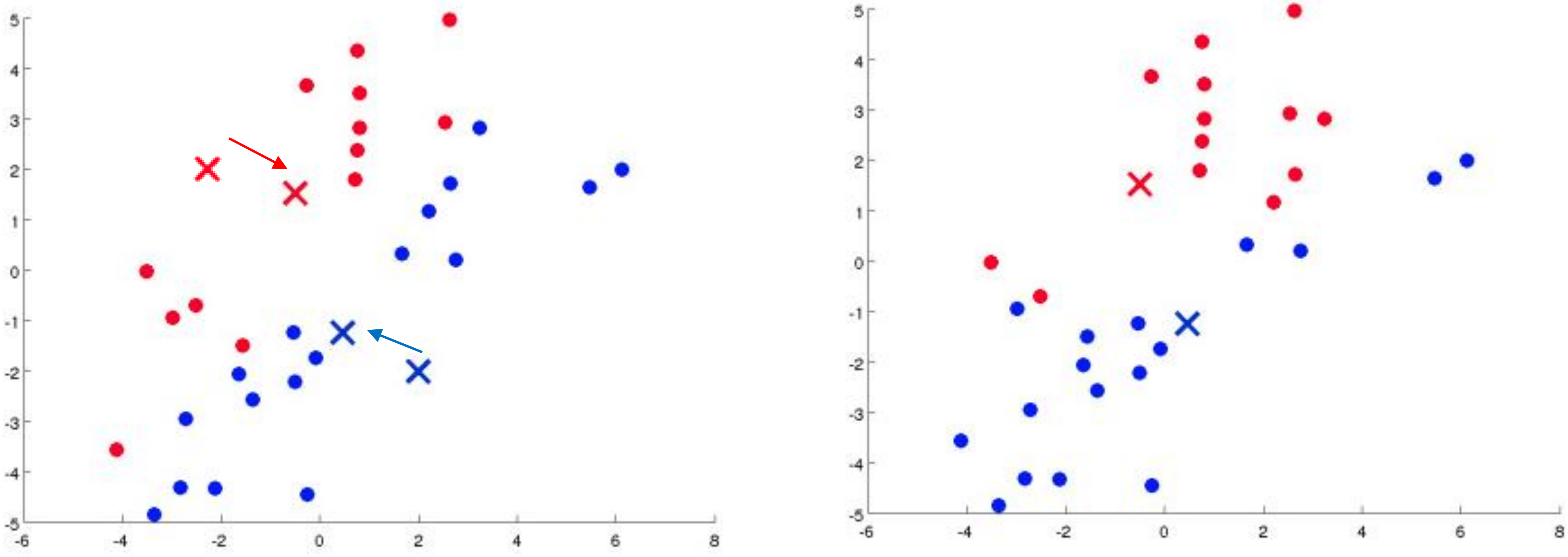
簇划分



K-means Algorithms

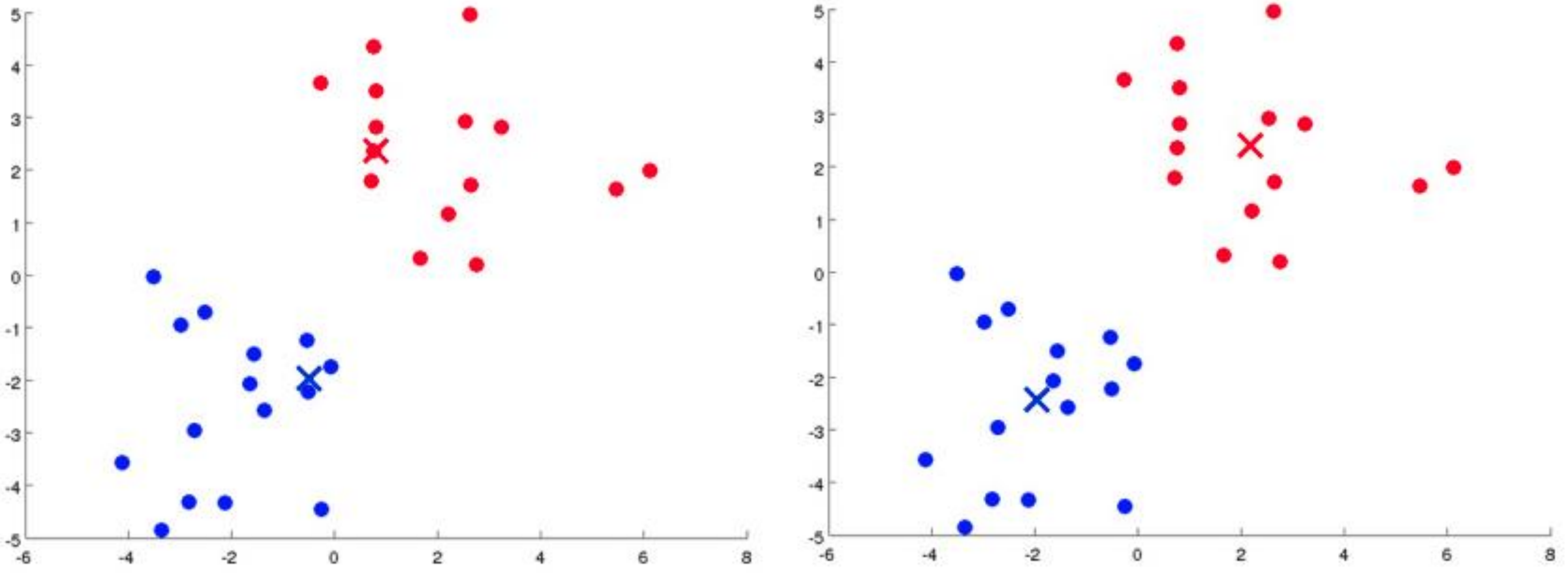
- Compute the average for two clusters.
- Re-determine cluster centroids.

移动聚类中心

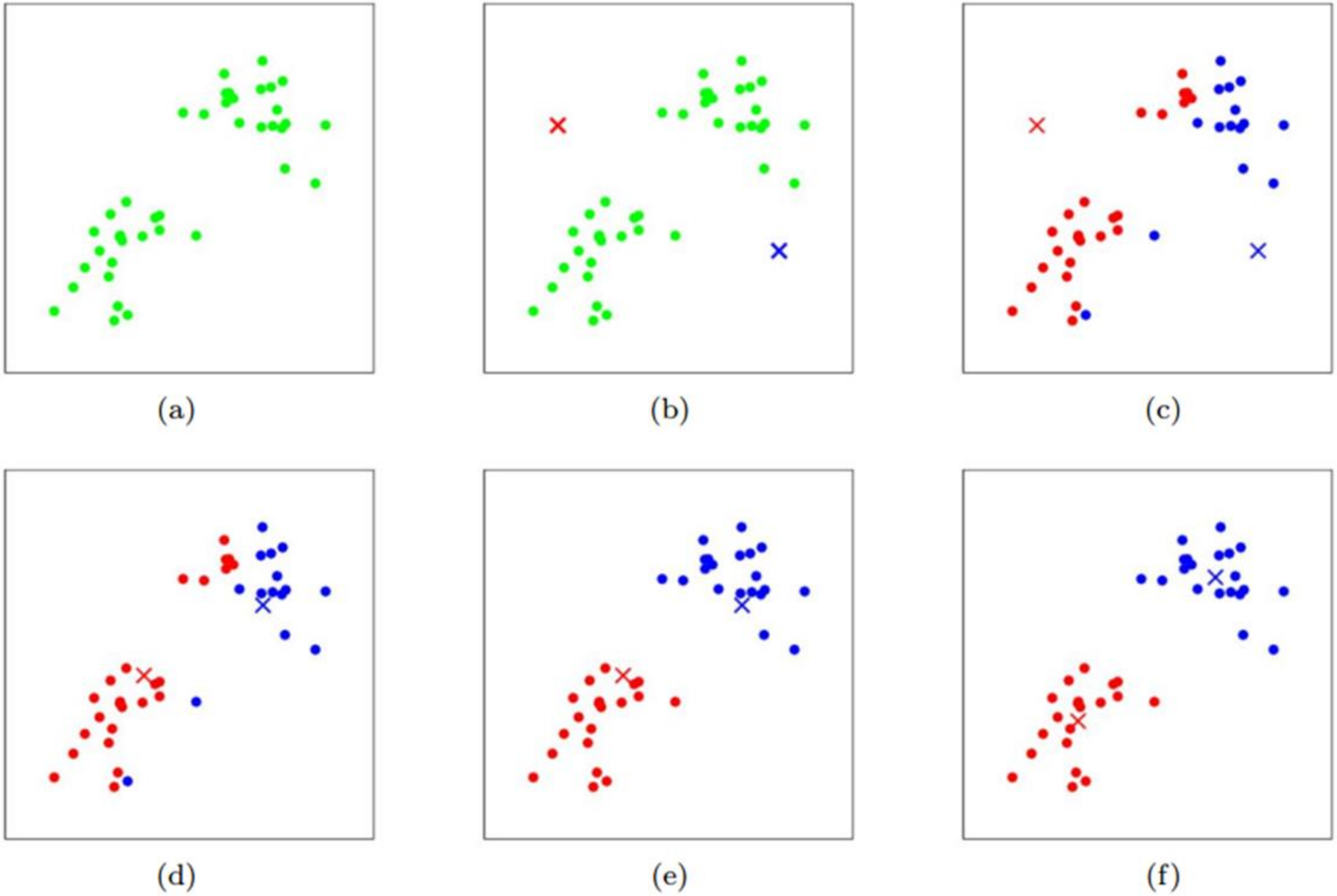


K-means Algorithms

□ Iterate until convergence (the cluster centroids will not change).



K-means Algorithm



(a)

(b)

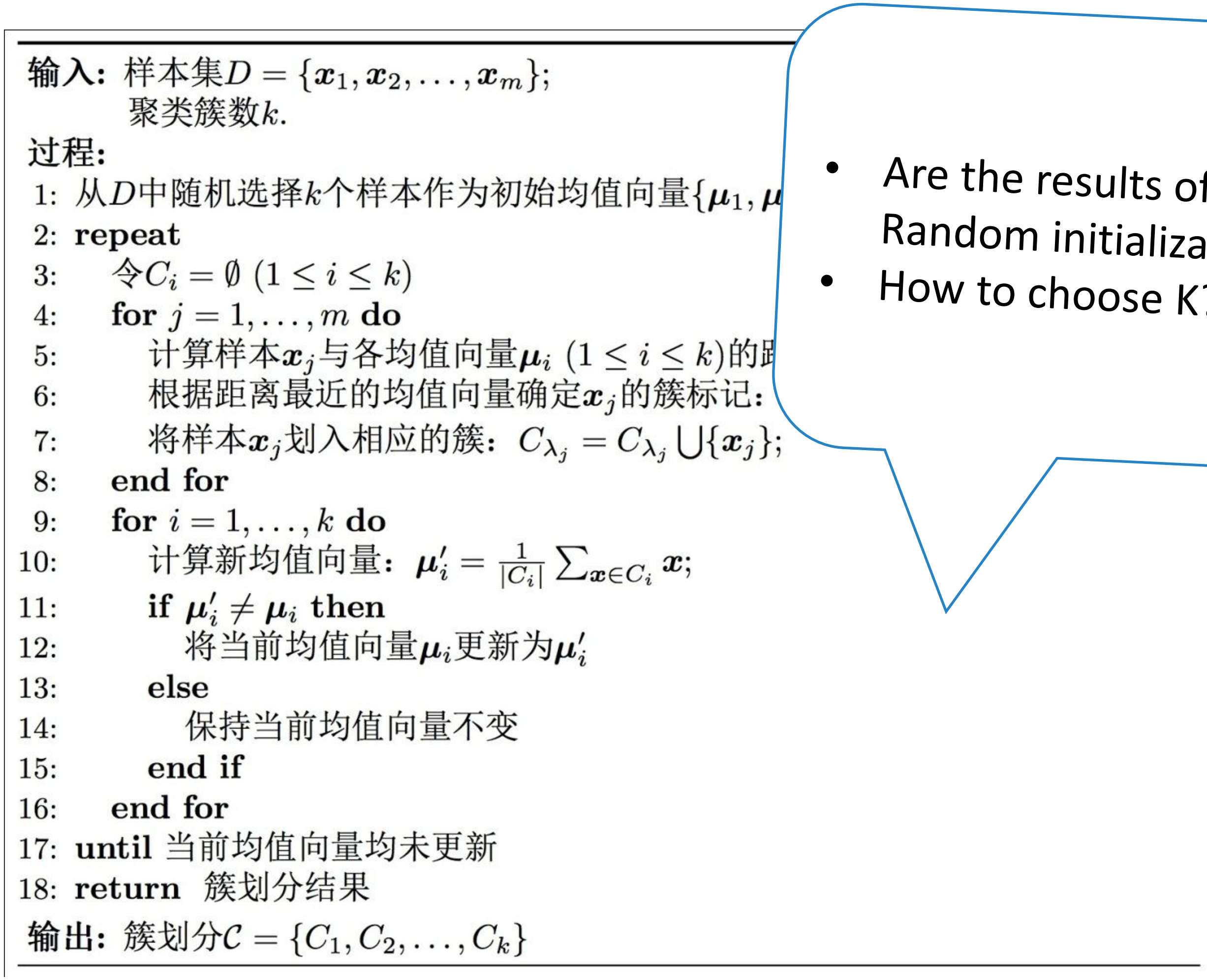
(c)

(d)

(e)

(f)

K-means Algorithm



```
输入：样本集$D = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m\}$；
      聚类簇数$k$.
过程：
从$D$中随机选择$k$个样本作为初始均值向量$\{\boldsymbol{\mu}_1, \boldsymbol{\mu}$
repeat
    令$C_i = \emptyset \ (1 \leq i \leq k)$
    for $j = 1, \ldots, m$ do
        计算样本$\boldsymbol{x}_j$与各均值向量$\boldsymbol{\mu}_i \ (1 \leq i \leq k)$的距
        根据距离最近的均值向量确定$\boldsymbol{x}_j$的簇标记：
        将样本$\boldsymbol{x}_j$划入相应的簇：$C_{\lambda_j} = C_{\lambda_j} \bigcup \{\boldsymbol{x}_j\}$；
    end for
    for $i = 1, \ldots, k$ do
        计算新均值向量：$\boldsymbol{\mu}'_i = \frac{1}{|C_i|} \sum_{\boldsymbol{x} \in C_i} \boldsymbol{x}$；
        if $\boldsymbol{\mu}'_i \neq \boldsymbol{\mu}_i$ then
            将当前均值向量$\boldsymbol{\mu}_i$更新为$\boldsymbol{\mu}'_i$
        else
            保持当前均值向量不变
        end if
    end for
until 当前均值向量均未更新
return 簇划分结果
输出：簇划分$\mathcal{C} = \{C_1, C_2, \ldots, C_k\}$
```

簇划分

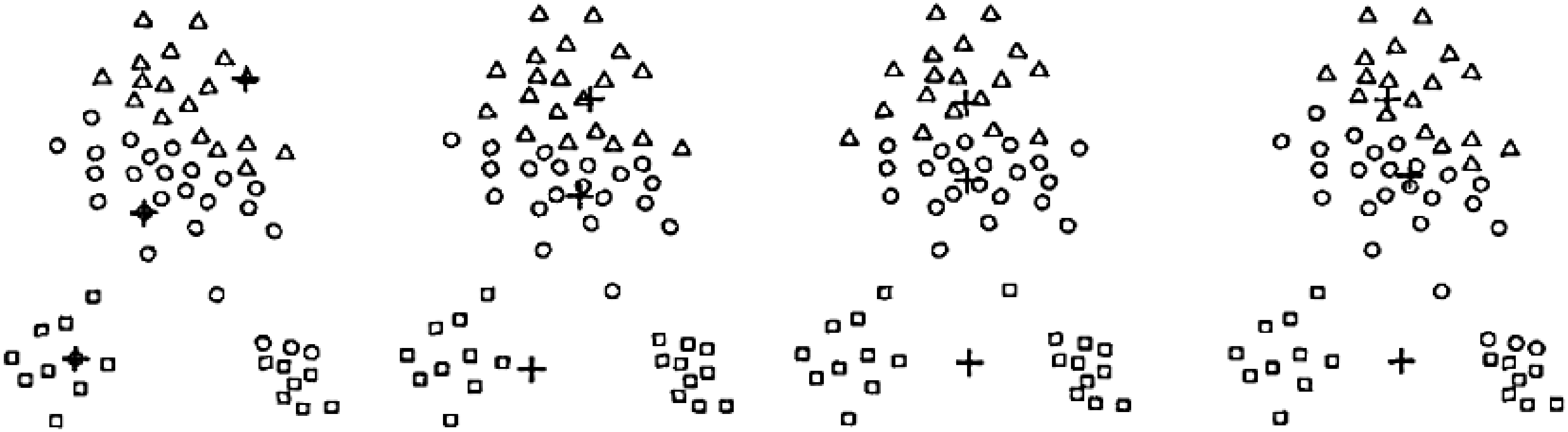
移动
聚类中心

- Are the results of every Random initialization same?
- How to choose K?

K-means

□ Random initialization

The experimental results show that every result is different because of different initializations.



(a)迭代1 (b)迭代2 (c)迭代3 (d)迭代4

https://blog.csdn.net/taoyanqi8932/article/details/53727841

K-means

- □ We need all possible initializations and get the best result.
- □ The measure to find the best result is minimizing square error E(SSE, sum of the Squared Error).

$$\mathrm{E} = \sum_{i=1}^{k} \sum_{x \in C_i} \|x - u_i\|_2^2 \qquad \textbf{or} \qquad \mathrm{E} = \sum_{i=1}^{m} \left\|x_i - u_{\lambda_i}\right\|_2^2$$

$$u_i = \frac{1}{|C_i|} \sum_{x \in C_i} x$$

K-means

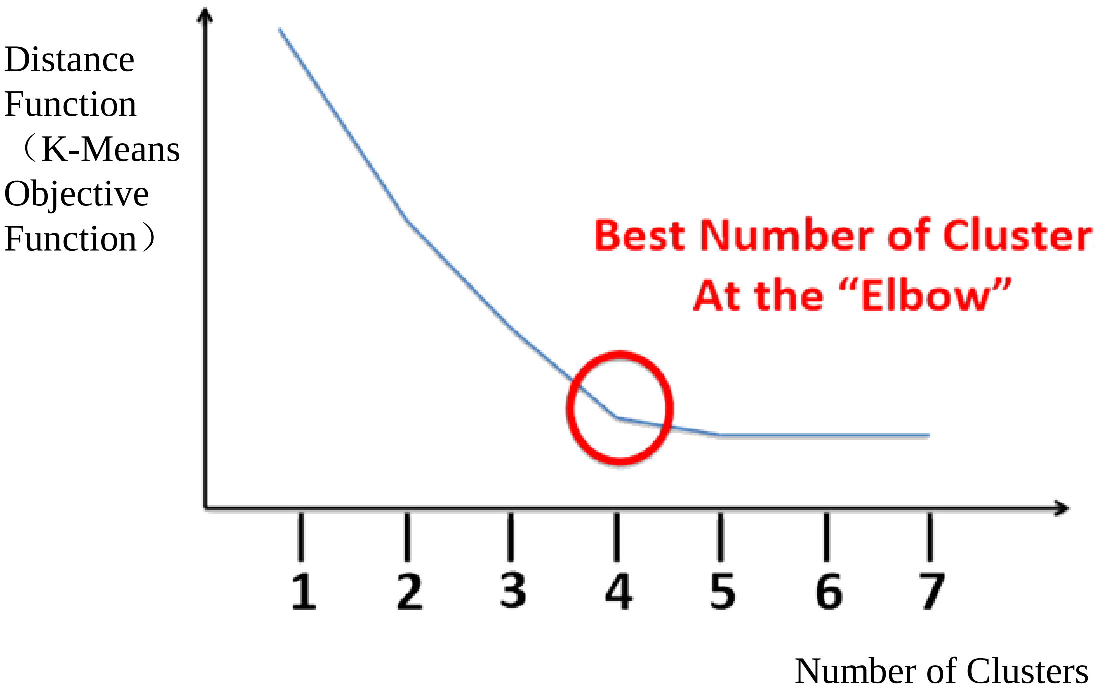
□ How to initialize

- It is NP-Hard to minimizing E.
- K-means uses a interative optimal algorithm. Each step of every iteration is the process of optimizing E.
- We can choose multiple initializations to get the best result(Attention: Whether this measure is effective depends on k).

K-means

□ How to choose K

Comparing multiple clusters “The Elbow Method”



K-means

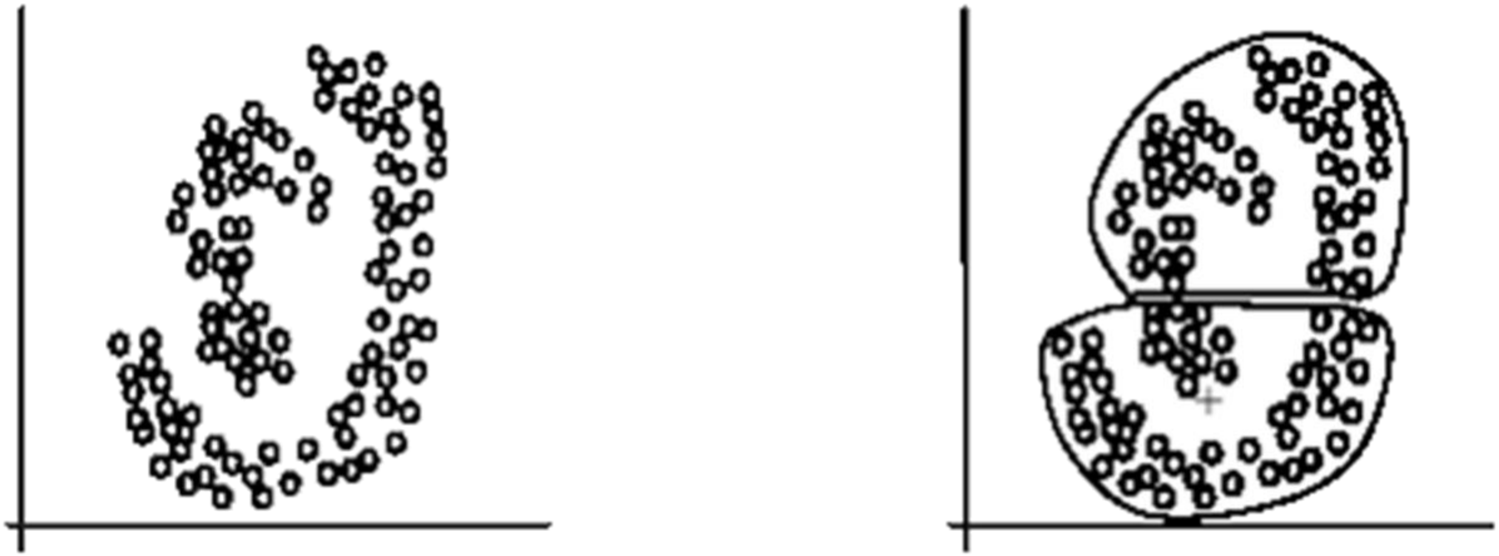
□ Exercise

Suppose you run k-means using k=3 and k=5. You find that the SSE is much higher for k=5 than for k=3.What can you conclude?

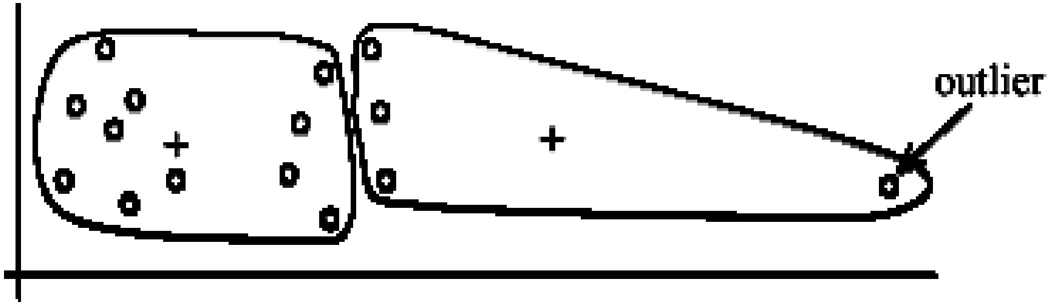
- ➢ This is mathematically impossible. There must be a bug in the code.
- ➢ The correct number of clusters is k=3.
- ➢ In the run with k=5, k-means got stuck in a bad local minimum . You should try re-running k-means with multiple random initializations.
- ➢ In the run with k=3,k-means got lucky. You should try re-running k-means with k=3 and different random initializations until it performs no better than with k=5.

K-means

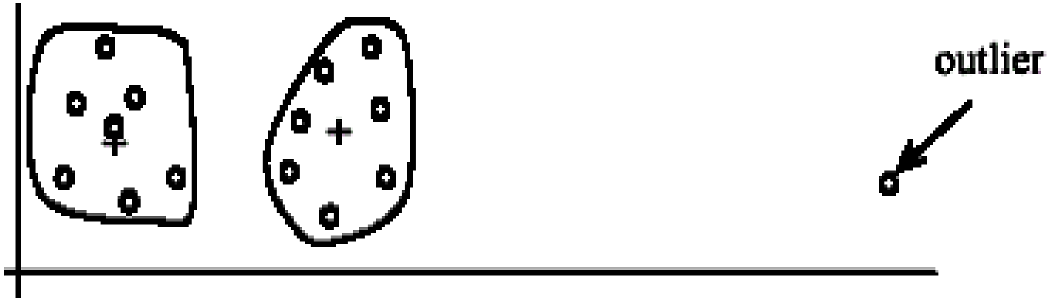
- □ K-means is not always suitable.



K-means



(A): Undesirable clusters



(B): Ideal clusters

Outline

□ Definition

□ Distance Calculation

□ Algorithms

- K-means
- Mixture of Gaussian
- DBSCAN
- AGNES

□ Performance Measure

Mixture of Gaussian

- □ Gaussian mixture distribution

$$p_m(x) = \sum_{i=1}^{k} a_i p(x|\mu_i, \Sigma_i\ ) \qquad \sum_{i=1}^{k} \alpha_i = 1$$

$\boldsymbol{\mu_i}, \boldsymbol{\Sigma_i}$: mean vector and covariance matrix of i_{th} mixed component

$\boldsymbol{\alpha_i}$: corresponding mixture coefficient

Mixture of Gaussian

□ Objective function：

$$p_m\left(z_j = i \middle| x_j\right) = \frac{p_m\left(x_j \middle| z_j = i\right) \cdot p(z_j = i)}{p\left(x_j\right)}$$

$$= \frac{\alpha_i \cdot p_m\left(x_j \middle| \mu_i, \Sigma_i\right)}{\sum_{l=1}^{k} \alpha_i \cdot p_m\left(x_j \middle| \mu_l, \Sigma_l\right)}$$

□ Then， we use MLE(Maximum likelihood estimate) to optimize function.

Mixture of Gaussian

$$\left\{\begin{aligned} \frac{\partial LL(D)}{\partial \mu_i} &= 0 \\ \frac{\partial LL(D)}{\partial \Sigma_i} &= 0 \end{aligned}\right. \longrightarrow \left\{ \boxed{\begin{aligned} u_i &= \frac{\sum_{j=1}^{m} \gamma_{ji} x_j}{\sum_{j=1}^{m} \gamma_{ji}} \\ \Sigma_i &= \frac{\sum_{j=1}^{m} \gamma_{ji} (x_j - \mu_i)(x_j - \mu_i)^T}{\sum_{j=1}^{m} \gamma_{ji}} \end{aligned}} \right.$$

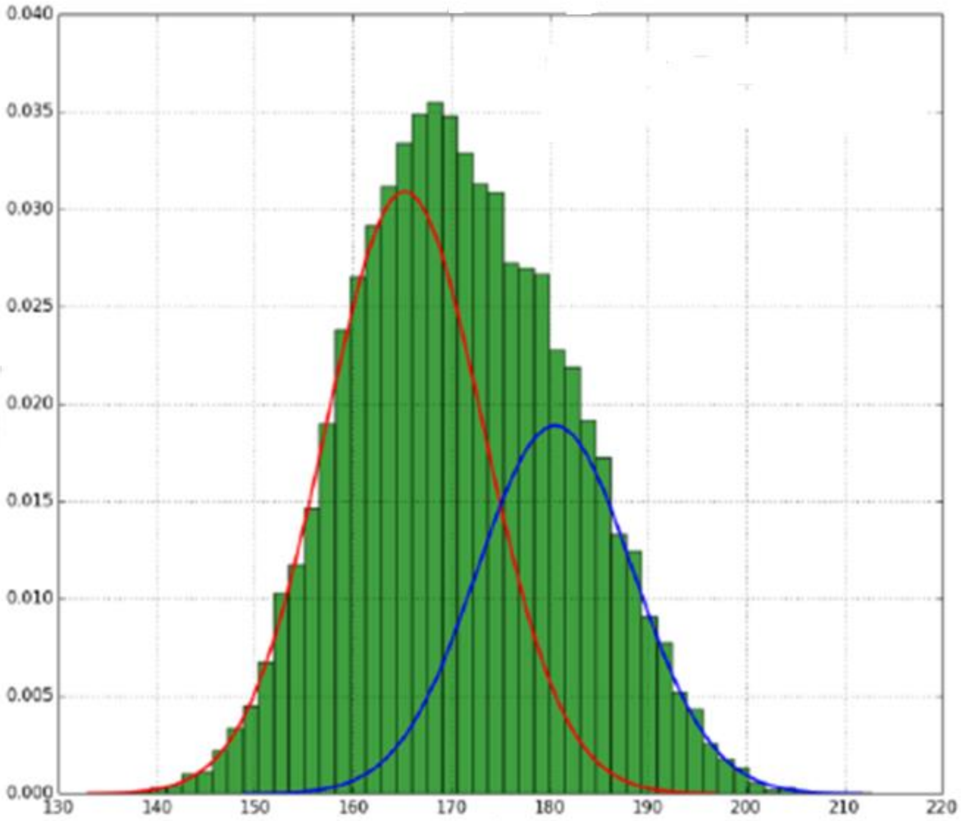
$$\text{s.t.} \quad \sum_{i=1}^{k} \alpha_i = 1$$

Lagrange function:

$$LL(D) + \lambda\left(\sum_{i=1}^{k} \alpha_i - 1\right) \longrightarrow \boxed{\alpha_i = \frac{1}{m} \sum_{j=1}^{k} \gamma_{ji}}$$

Mixture of Gaussian

- □ Suppose we have 100 students and the only data we can get is their height. Try to model the distribution of male and female.
- □ The height obeys Gaussian distribution.

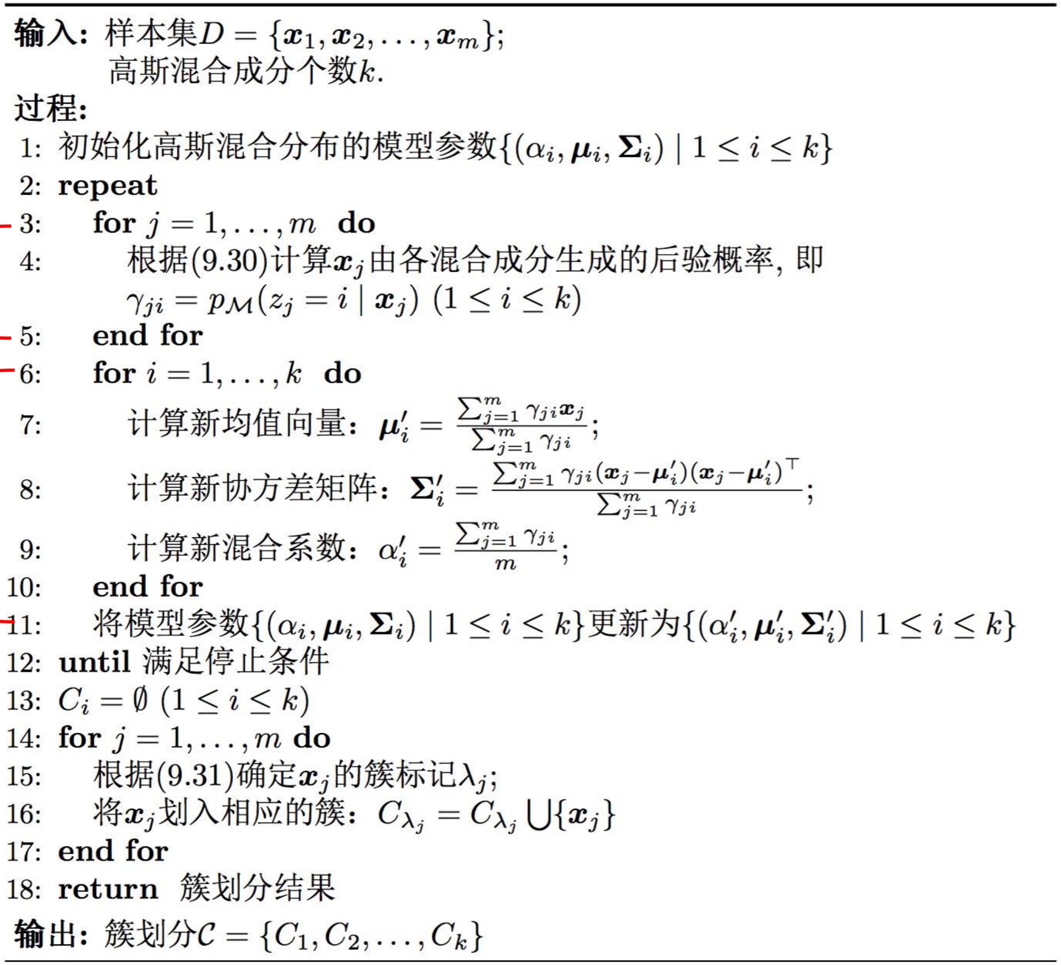


Mixture of Gaussian

先假定男生服从参数为~N(180,10)的高斯分布，女生服从参数为~N(160,8)的高斯分布(Assume a determined Gaussian model for boys and girls)

- 对每个样本计算出分别属于男生和女生的概率(Compute the probability that clustering each sample to boys and girls)
- 认定：每个样本分属于男生和女生的部分（即概率，用$\gamma(i,k)$表示，即第i个样本属于第k个类别的概率）同样服从高斯分布，且具有更好的拟合属性。(如一个样本身高175，我们可以通过设定的参数计算出他有80%的概率为男生，20%的概率为女生，那可以把这个样本看作由80%的男生和20%的女生组成，并将这个样本看作是一个80%的男生样本和一个20%的女生样本。)(Divide each sample into two components according to the probability in step 2)
- 根据每个样本的男生组分和女生组分拟合出新的高斯模型。(Update the parameters according to the components in step 3)
- 迭代直到收敛。(Iterating these steps until convergence)

Mixture of Gaussian

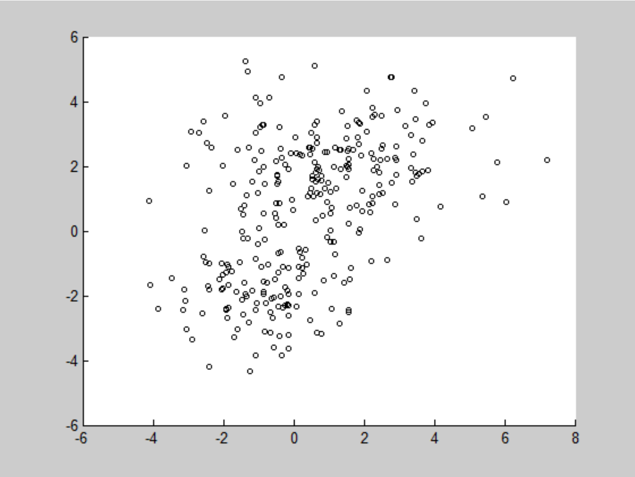


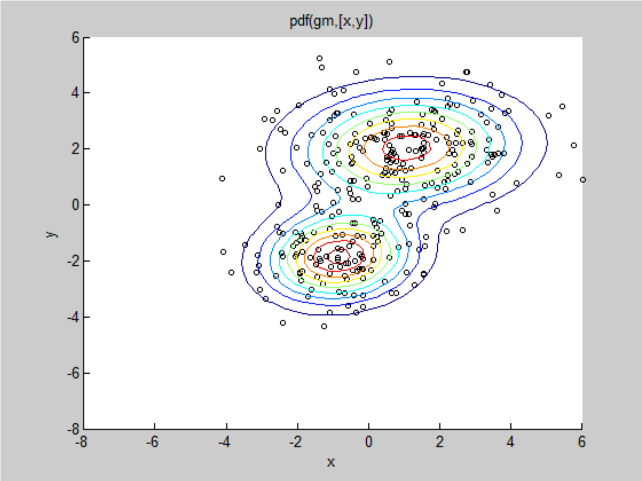
```
输入：样本集$D = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m\}$;
      高斯混合成分个数$k$.
过程：
初始化高斯混合分布的模型参数$\{(\alpha_i, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) \mid 1 \leq i \leq k\}$
repeat
    for $j = 1, \ldots, m$ do
        根据(9.30)计算$\boldsymbol{x}_j$由各混合成分生成的后验概率，即
        $\gamma_{ji} = p_{\mathcal{M}}(z_j = i \mid \boldsymbol{x}_j)\ (1 \leq i \leq k)$
    end for
    for $i = 1, \ldots, k$ do
        计算新均值向量：$\boldsymbol{\mu}_i' = \frac{\sum_{j=1}^m \gamma_{ji}\boldsymbol{x}_j}{\sum_{j=1}^m \gamma_{ji}}$;
        计算新协方差矩阵：$\boldsymbol{\Sigma}_i' = \frac{\sum_{j=1}^m \gamma_{ji}(\boldsymbol{x}_j-\boldsymbol{\mu}_i')(\boldsymbol{x}_j-\boldsymbol{\mu}_i')^\top}{\sum_{j=1}^m \gamma_{ji}}$;
        计算新混合系数：$\alpha_i' = \frac{\sum_{j=1}^m \gamma_{ji}}{m}$;
    end for
    将模型参数$\{(\alpha_i, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) \mid 1 \leq i \leq k\}$更新为$\{(\alpha_i', \boldsymbol{\mu}_i', \boldsymbol{\Sigma}_i') \mid 1 \leq i \leq k\}$
until 满足停止条件
$C_i = \emptyset\ (1 \leq i \leq k)$
for $j = 1, \ldots, m$ do
    根据(9.31)确定$\boldsymbol{x}_j$的簇标记$\lambda_j$;
    将$\boldsymbol{x}_j$划入相应的簇：$C_{\lambda_j} = C_{\lambda_j} \bigcup \{\boldsymbol{x}_j\}$
end for
return 簇划分结果
输出：簇划分$\mathcal{C} = \{C_1, C_2, \ldots, C_k\}$
```

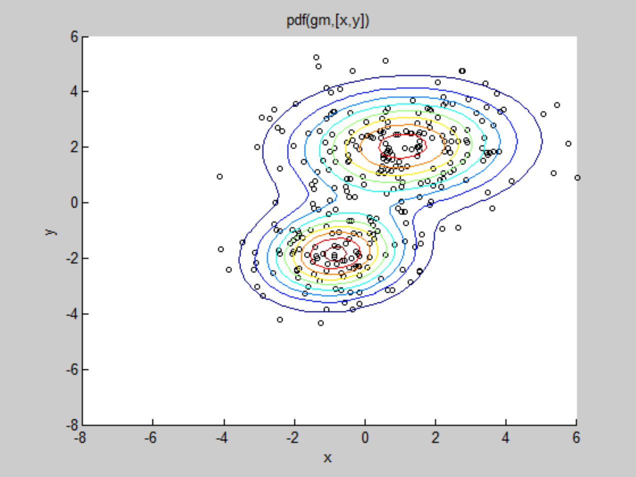
固定模型参数
更新后验概率

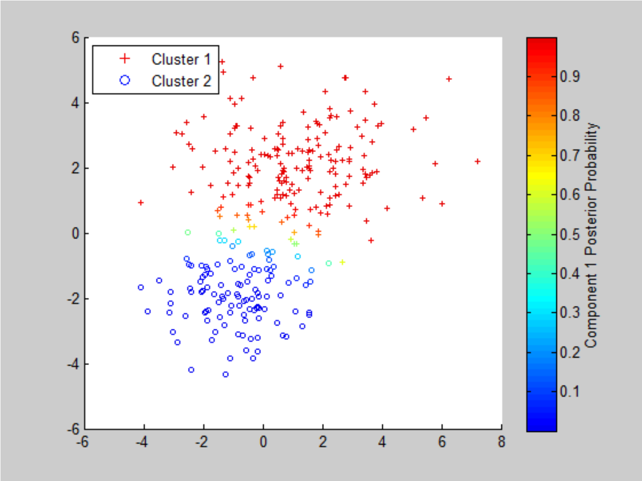
固定后验概率
更新模型参数

Mixture of Gaussian









https://www.cnblogs.com/wt869054461/p/6066042.html

Outline

□ Definition

□ Distance Calculation

□ Algorithms

- K-means
- Mixture of Gaussian
- DBSCAN
- AGNES

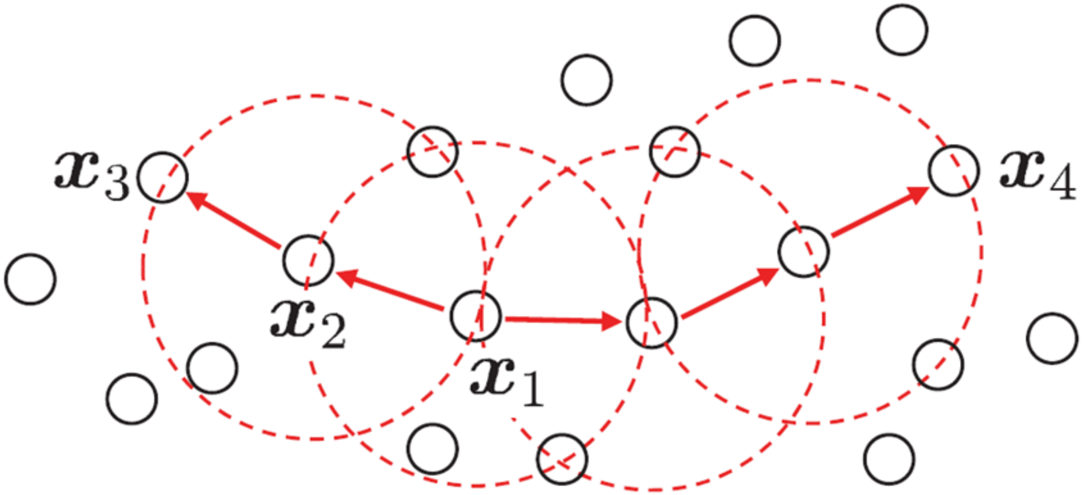
□ Performance Measure

DBSCAN

- It is a famous density-based-clustering.
- ϵ-neighborhood:$N_\epsilon\left(x_j\right) = \{x_i \in D | dist(x_i, x_j\ ) \leq \epsilon)$ of the point x_j
- core object :a point with a $\left|N_\epsilon\left(x_j\right)\right| \geq MinPts$
- Directly density-reached: x_j is density-reachable from a core object x_i if x_j is in $N_\epsilon(x_i)$
- Density-reached: x_j is density-reachable from a core object x_i if a sequence of core objects $p_1, p_2, \dots\ p_n$ between x_i and x_j exists and $p_{i+1}$is directly density-reached from p_i.
- Density-connected: x_i and x_j are density-connected if they are density-reachable from a common core object x_k.

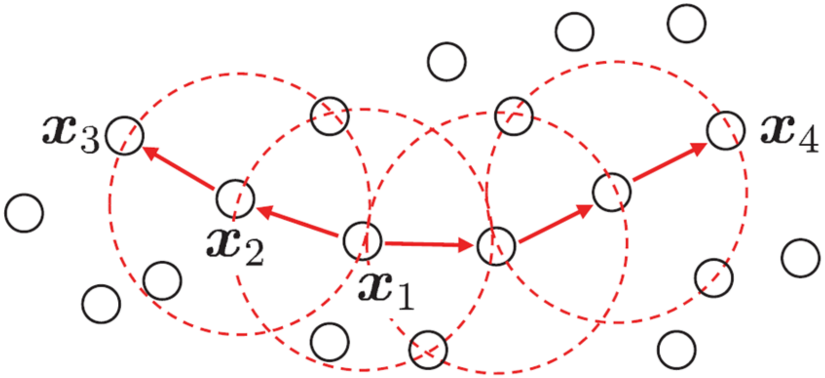
DBSCAN

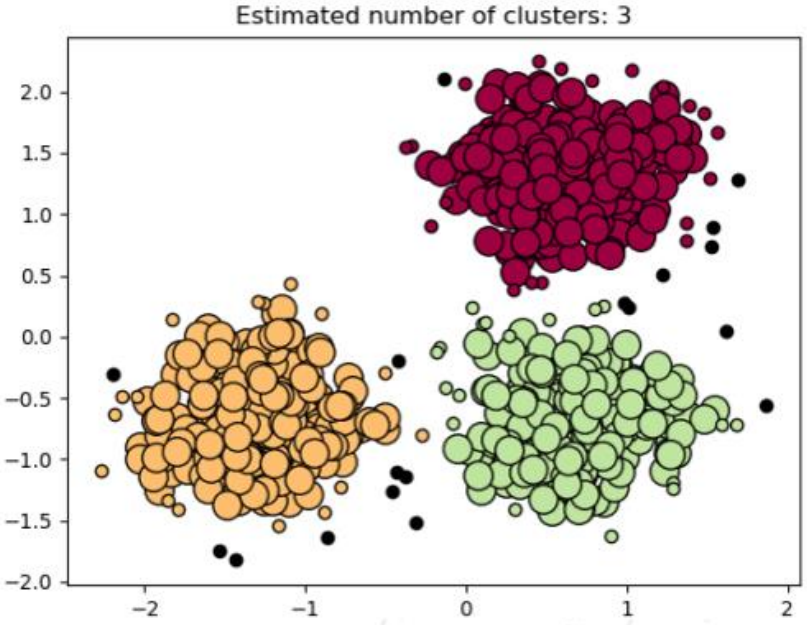
- □ Directly density-reached： X2 from X1
- □ Density-reached: X3 from X1
- □ Density-connected： X4 from X3



DBSCAN

- DBSCAN defines cluster as such sample set which is most density-connected.



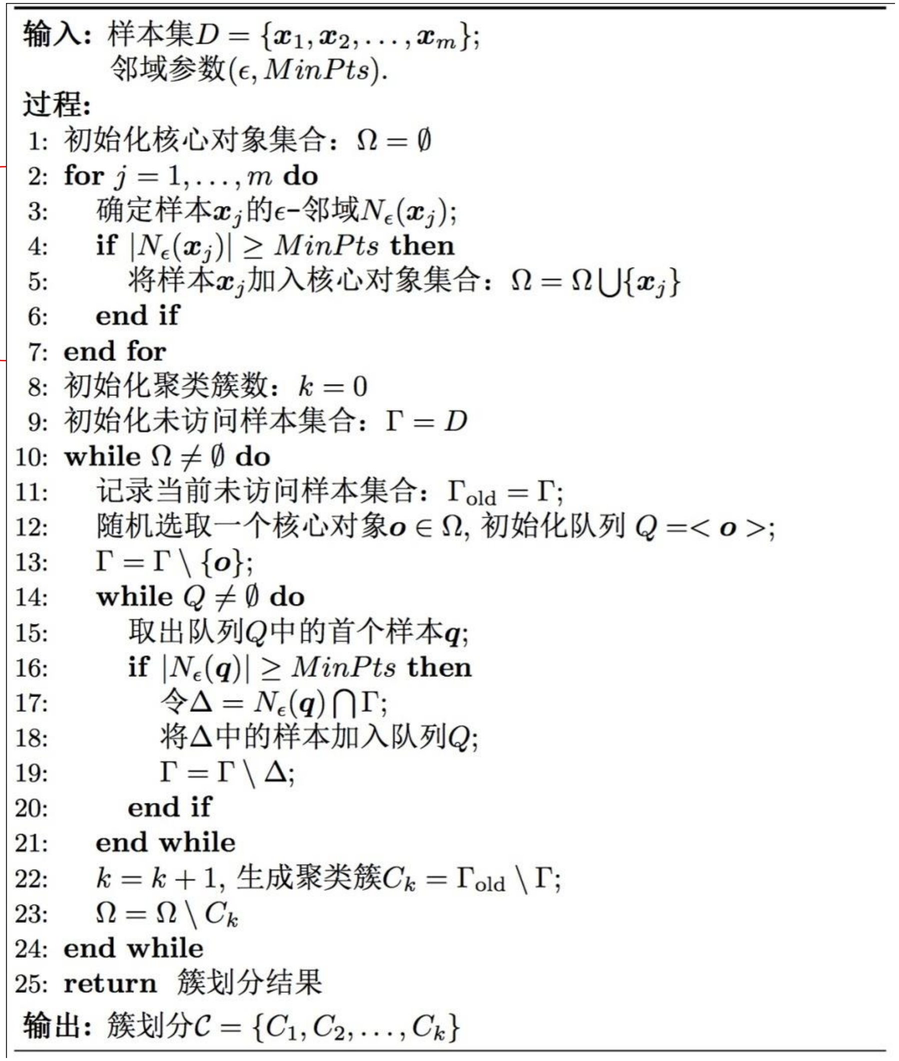


https://blog.csdn.net/leonliu1995/article/details/78944798

DBSCAN

找出所有核心对象

随机选一个核心对象生长出一个簇，并在核心对象集合里删去该核心对象

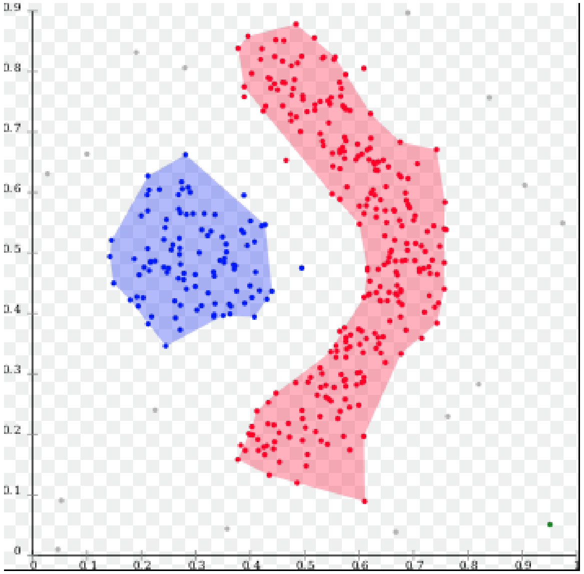


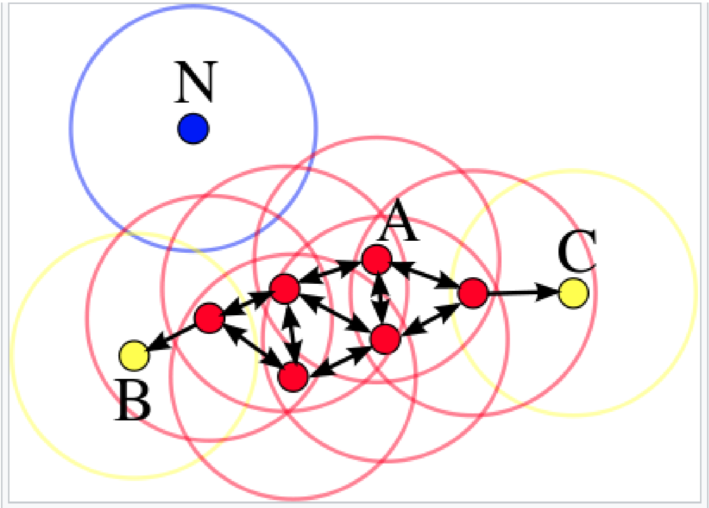
```
输入：样本集$D = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m\}$;
      邻域参数$(\epsilon, MinPts)$.
过程：
初始化核心对象集合：$\Omega = \emptyset$
for $j = 1, \ldots, m$ do
    确定样本$\boldsymbol{x}_j$的$\epsilon$-邻域$N_\epsilon(\boldsymbol{x}_j)$;
    if $|N_\epsilon(\boldsymbol{x}_j)| \geq MinPts$ then
        将样本$\boldsymbol{x}_j$加入核心对象集合：$\Omega = \Omega \bigcup \{\boldsymbol{x}_j\}$
    end if
end for
初始化聚类簇数：$k = 0$
初始化未访问样本集合：$\Gamma = D$
while $\Omega \neq \emptyset$ do
    记录当前未访问样本集合：$\Gamma_{\text{old}} = \Gamma$;
    随机选取一个核心对象$\boldsymbol{o} \in \Omega$, 初始化队列 $Q =< \boldsymbol{o} >$;
    $\Gamma = \Gamma \setminus \{\boldsymbol{o}\}$;
    while $Q \neq \emptyset$ do
        取出队列$Q$中的首个样本$\boldsymbol{q}$;
        if $|N_\epsilon(\boldsymbol{q})| \geq MinPts$ then
            令$\Delta = N_\epsilon(\boldsymbol{q}) \bigcap \Gamma$;
            将$\Delta$中的样本加入队列$Q$;
            $\Gamma = \Gamma \setminus \Delta$;
        end if
    end while
    $k = k + 1$, 生成聚类簇$C_k = \Gamma_{\text{old}} \setminus \Gamma$;
    $\Omega = \Omega \setminus C_k$
end while
return 簇划分结果
输出：簇划分$\mathcal{C} = \{C_1, C_2, \ldots, C_k\}$
```

DBSCAN

□ Strengths

- ➢ There is no K.
- ➢ It can discover any shape of spatial clustering.
- ➢ It can discard the remote point.



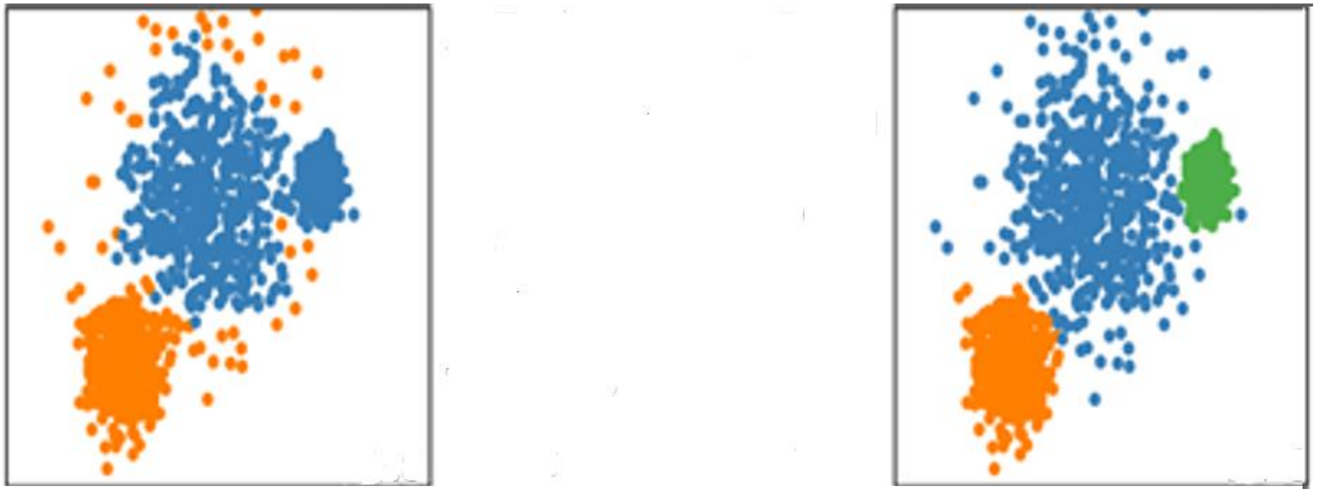


https://en.wikipedia.org/wiki/DBSCAN

DBSCAN

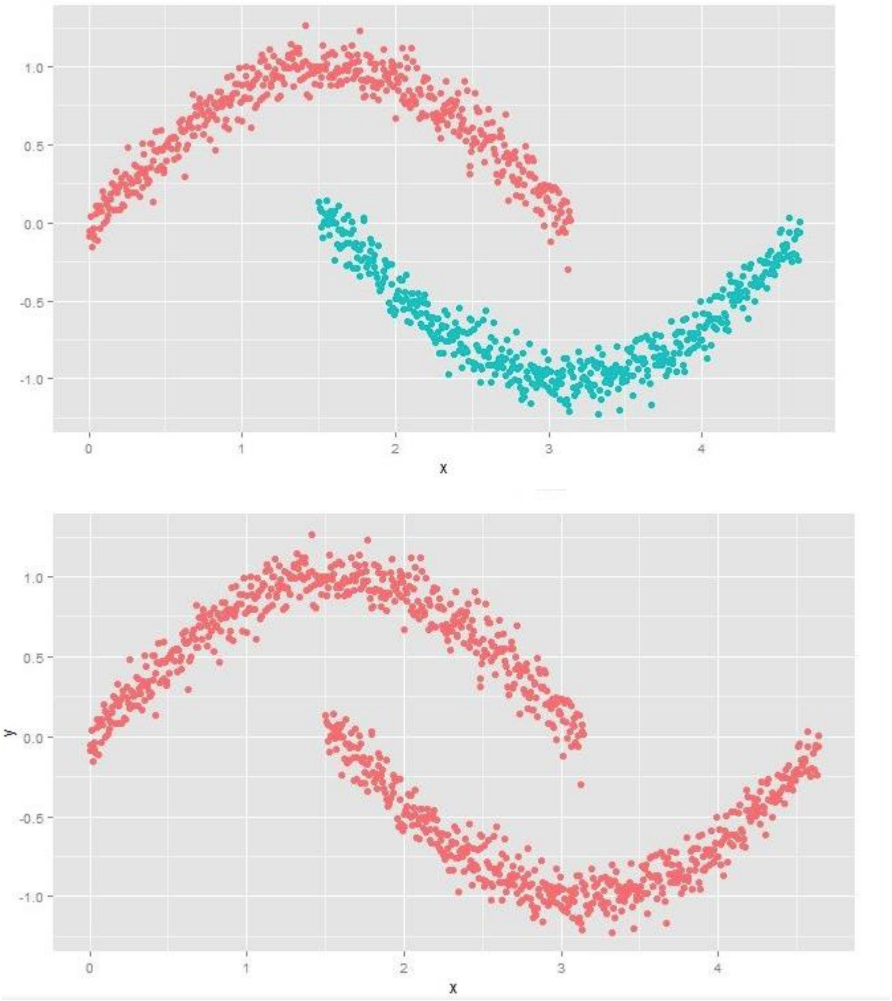
□ Weaknesses

- ➢ It is not suitable when the cluster spacing difference is very different.
- ➢ Parameters adjustment are more complicated.



http://scikitlearn.org/stable/auto_examples/cluster/plot_cluster_comparison.html#sphx-glr-auto-examples-cluster-plot-cluster-comparison-py

DBSCAN



https://zhuanlan.zhihu.com/p/23504573

Outline

□ Definition

□ Distance Calculation

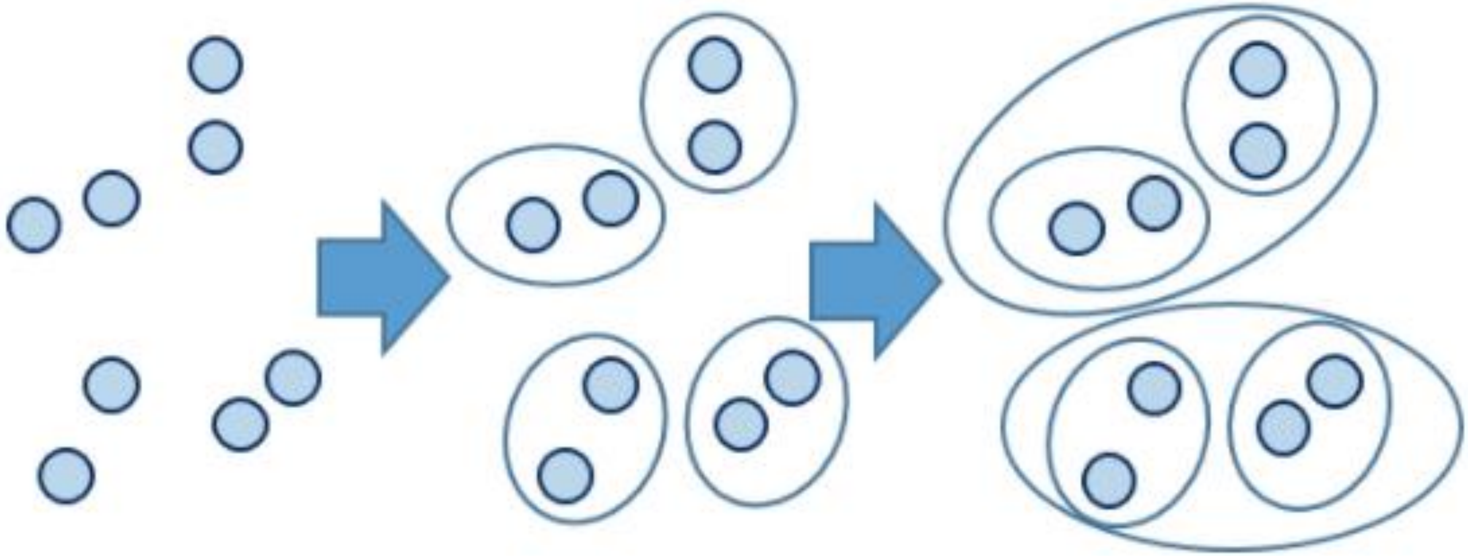
□ Algorithms

- K-means
- Mixture of Gaussian
- DBSCAN
- AGNES

□ Performance Measure

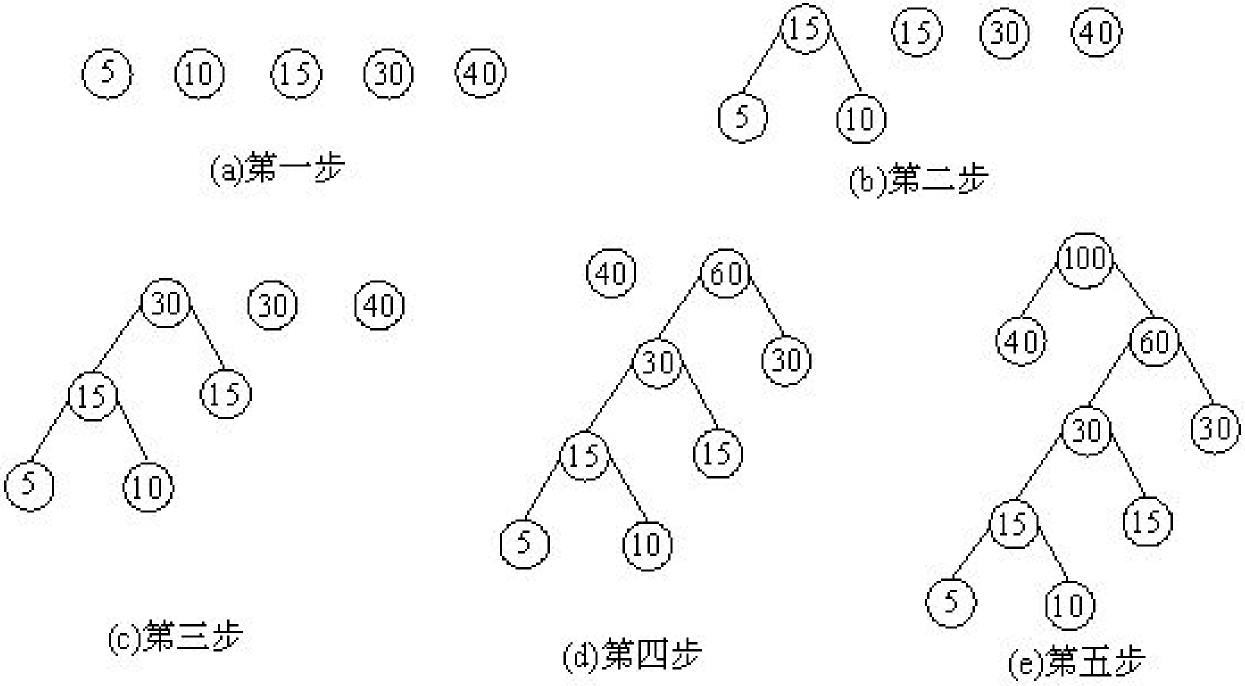
AGNES

- AGNES is a kind of Hierarchical clustering(层次聚类)



http://bluewhale.cc/2016-04-19/hierarchical-clustering.html

AGNES



(a)第一步

(b)第二步

(c)第三步

(d)第四步

(e)第五步

AGNES

- □ Given cluster C_i , C_j, Usually the distance between two clusters is one of the following
  - ➢ Maximum distance(also called complete-linkage clustering)

$$dist_{max}\left(C_i\,,C_j\right)=\max_{x\in C_i\,,z\in C_j} dist(x,z)$$

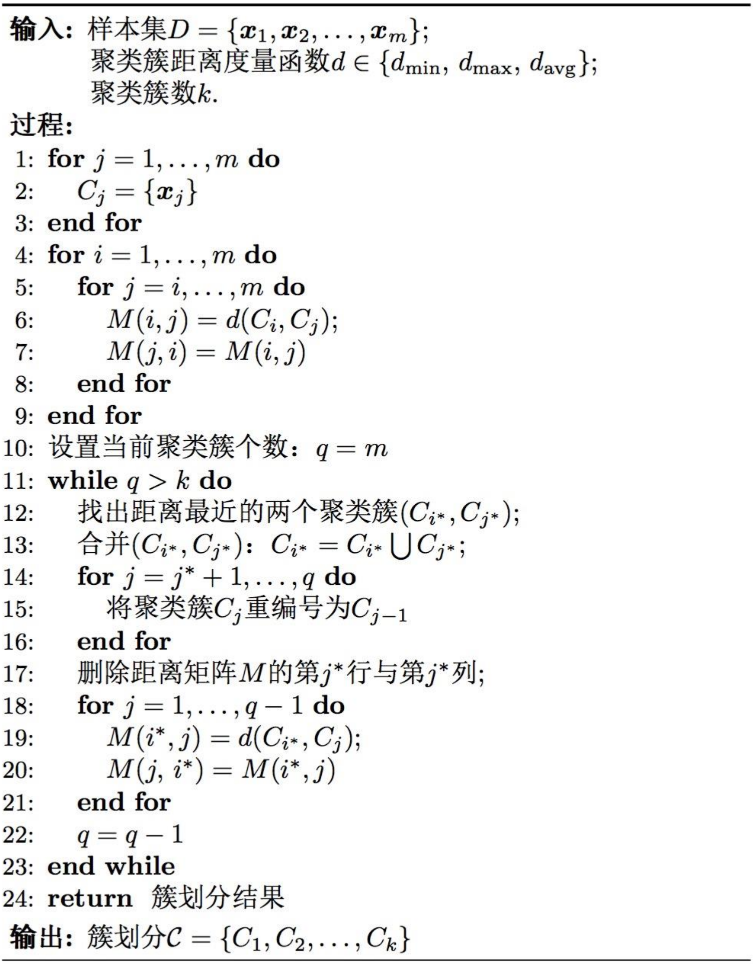
  - ➢ Minimum distance(also called single-linkage clustering)

$$dist_{\min}\left(C_i\,,C_j\right)=\min_{x\in C_i\,,z\in C_j} dist(x,z)$$

  - ➢ Average distance (also called average-linkage clustering)

$$dist_{avg}\left(C_i\,,C_j\right)=\frac{1}{|C_i||C_j|}\sum_{x\in C_i}\sum_{z\in C_j} dist(x,z)$$

AGNES

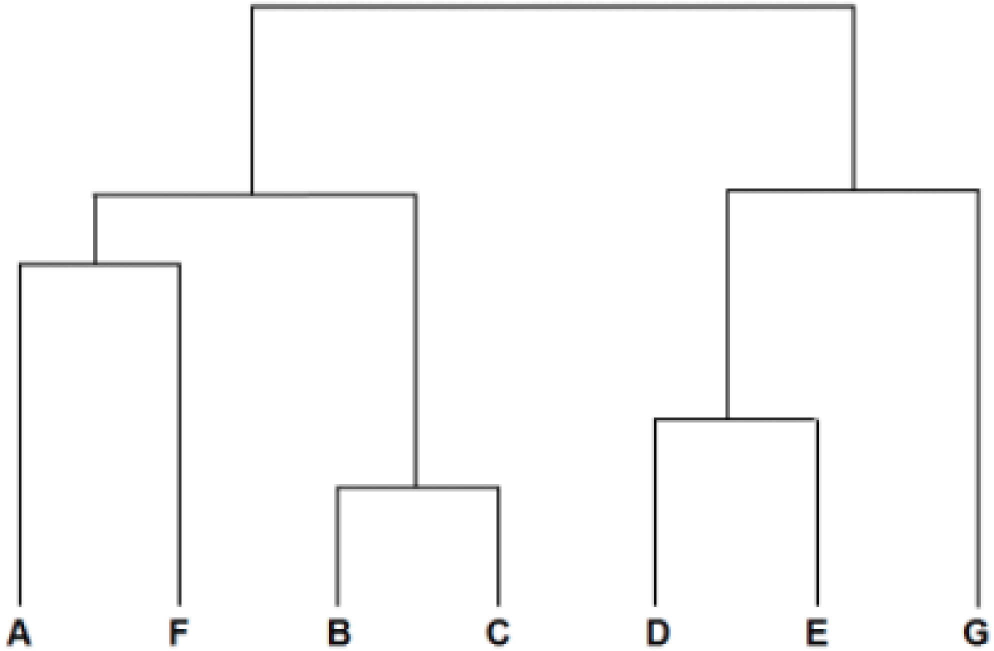


```
输入：样本集$D=\{\boldsymbol{x}_1,\boldsymbol{x}_2,\ldots,\boldsymbol{x}_m\}$;
      聚类簇距离度量函数$d\in\{d_{\min},d_{\max},d_{\text{avg}}\}$;
      聚类簇数$k$.
过程：
for $j=1,\ldots,m$ do
    $C_j=\{\boldsymbol{x}_j\}$
end for
for $i=1,\ldots,m$ do
    for $j=i,\ldots,m$ do
        $M(i,j)=d(C_i,C_j)$;
        $M(j,i)=M(i,j)$
    end for
end for
设置当前聚类簇个数：$q=m$
while $q>k$ do
    找出距离最近的两个聚类簇$(C_{i^*},C_{j^*})$;
    合并$(C_{i^*},C_{j^*})$：$C_{i^*}=C_{i^*}\bigcup C_{j^*}$;
    for $j=j^*+1,\ldots,q$ do
        将聚类簇$C_j$重编号为$C_{j-1}$
    end for
    删除距离矩阵$M$的第$j^*$行与第$j^*$列;
    for $j=1,\ldots,q-1$ do
        $M(i^*,j)=d(C_{i^*},C_j)$;
        $M(j,i^*)=M(i^*,j)$
    end for
    $q=q-1$
end while
return 簇划分结果
输出：簇划分$\mathcal{C}=\{C_1,C_2,\ldots,C_k\}$
```

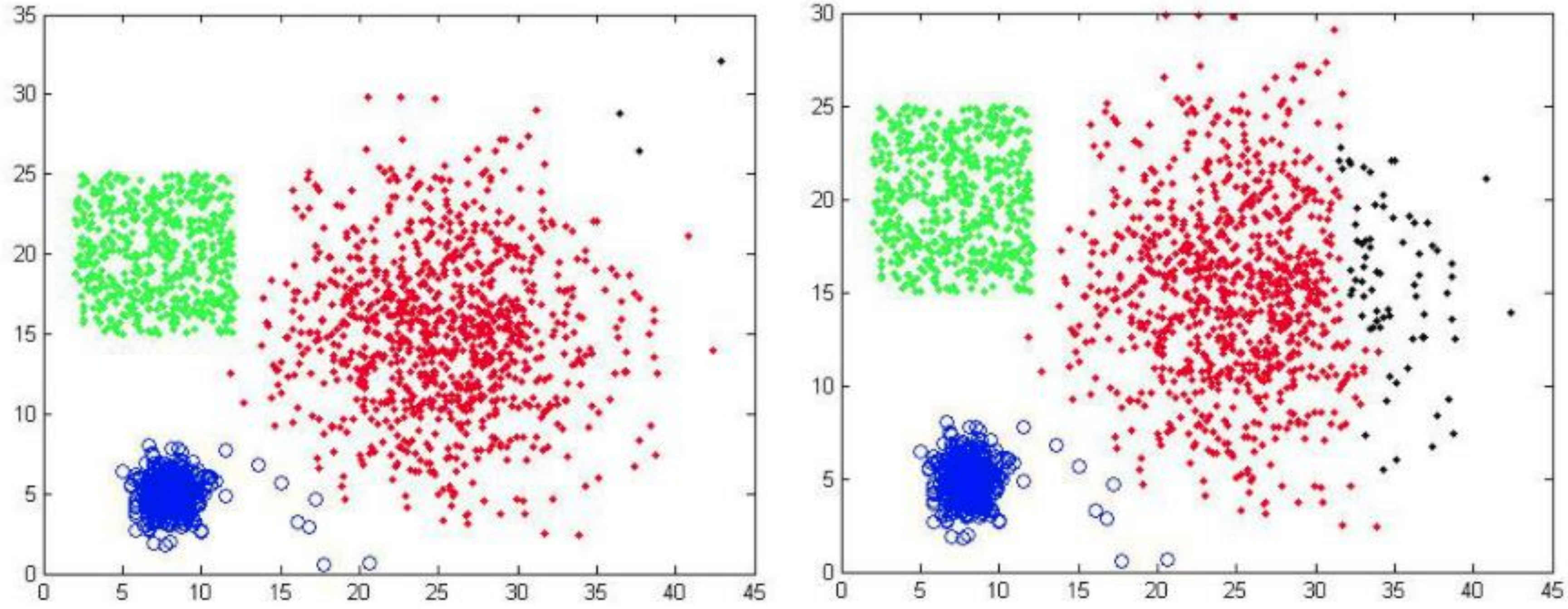
计算距离矩阵

每次循环合并两个簇并更新距离矩阵

AGNES



AGNES



https://blog.csdn.net/xiaokang06/article/details/51441534

Summary

- □ Partitioning Methods

  k-means, k-medoids, CLARANS, FCM

- □ Hierarchical Methods

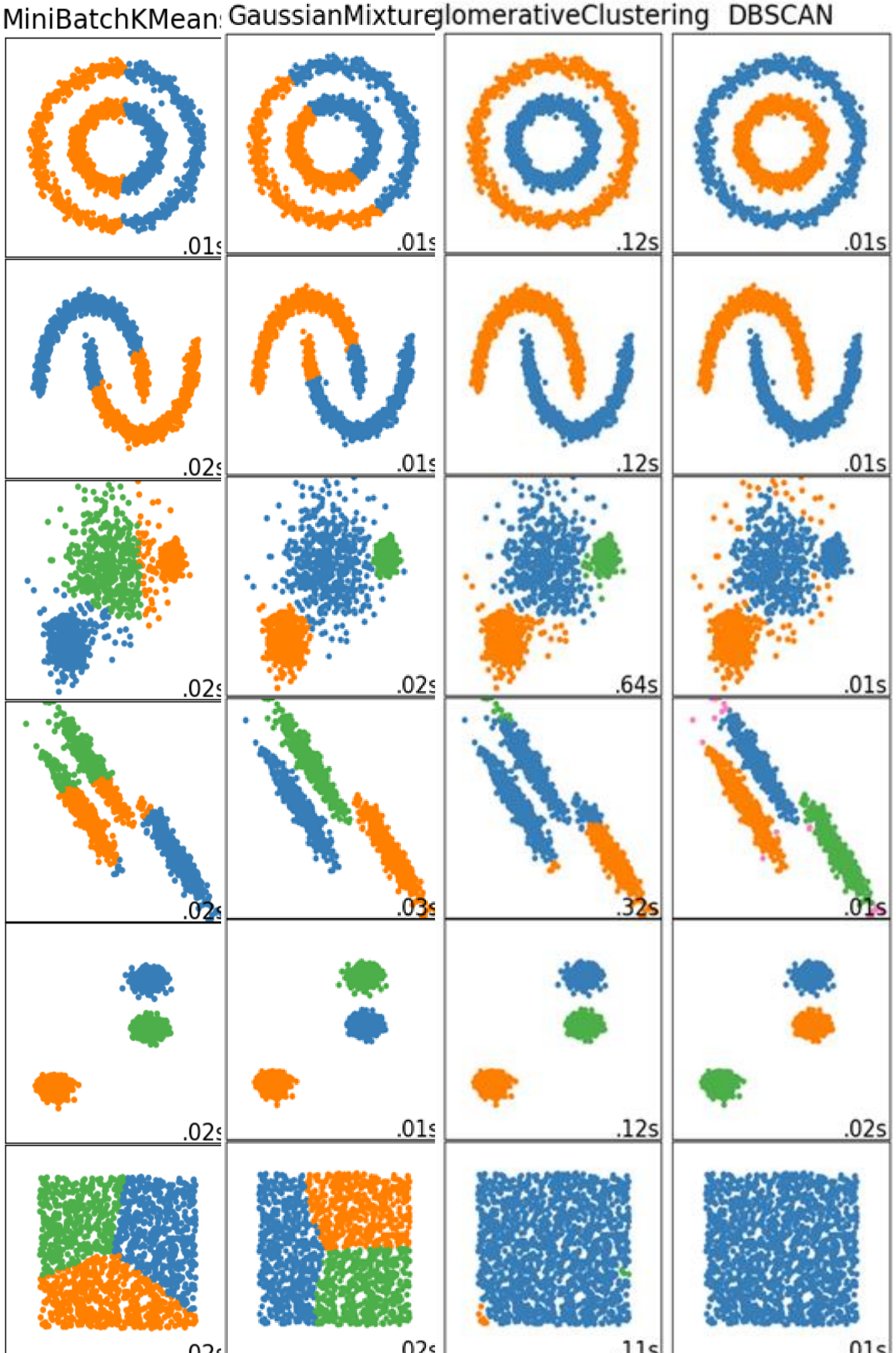
  AGNES, Birch, Cure, Rock, CHEMALOEN

- □ Density-based Methods

  DBSCAN,OPTICS

- □ Grid-based Methods
- □ Model-Based Methods

Transitive closure, Boolean matrix, direct clustering, correlation analysis clustering, clustering method based on statistics……


MiniBatchKMean:
GaussianMixture
glomerativeClustering
DBSCAN
.01s
.01s
.12s
.01s
.02s
.01s
.12s
.01s
.02s
.02s
.64s
.01s
.02s
.03s
.32s
.01s
.02s
.01s
.12s
.02s
.02s
.02s
.11s
.01s

Summary

- https://www.cnblogs.com/taojakeML/p/6144266.html
- http://blog.csdn.net/abcjennifer/article/details/8170687
- http://blog.csdn.net/zhoubl668/article/details/6639801?locationNum=3&fps=1
- http://blog.csdn.net/zhoubl668/article/details/7945678?locationNum=5&fps=1
- http://blog.csdn.net/zhoubl668/article/details/7881313?locationNum=13&fps=1
- http://blog.pluskid.org/?page_id=78

Outline

- Definition
- Distance Calculation
- Algorithms
  - K-means
  - Mixture of Gaussian
  - DBSCAN
  - AGNES
- Performance Measure

Performance Measure

Good clustering should be:

Intra-cluster similarity ⇨ maximized

Inter-cluster similarity ⇨ minimized

Intra distance & Inter distance

Intra distance

$$agv(C) = \frac{2}{|C|(|C|-1)} \sum_{1 \le i \le j \le |C|} dist(x_i, x_j)$$

$$diam(C) = \max_{1 \le i \le j \le |C|} dist(x_i, x_j)$$

Inter distance

$$d_{min}(C_i, C_j) = \min_{x_i \in C_i, x_j \in C_j} dist(x_i, x_j)$$

$$d_{cen}(C_i, C_j) = dis(\mu_i, \mu_j)$$

Performance Measure

- □ Internal Index

  Evaluate clustering results directly without using reference model.

- □ External Index

  Compare clustering results with reference model, for example, partitioning results given by domain expert.

Internal Index

□ Davies-Boukdin Index:

The smaller the better

$$DBI = \frac{1}{k}\sum_{i=1}^{k} \boxed{\max_{j\neq i}\boxed{\left(\frac{avg(C_i)+avg(C_j)}{d_{cen}(C_i,C_j)}\right)}}$$

□ Dunn Index:

The bigger the better

$$DI = \min_{1\le i\le k} \boxed{\left\{\min_{j\neq i} \boxed{\frac{d_{min}(C_i,C_j)}{\max\limits_{1\le l\le k} diam(C_l)}}\right\}}$$

External Index

- Assume our cluster partition is

$$C = \{C_1, C_2, \ldots C_k\}$$

- The partition given by reference model is

$$C^* = \{C_1^*, C_2^*, \ldots C_s^*\}$$

- let λ and λ^* be clustering label vectors corresponding to C and C^*. Consider C_m^2 sample pairs

$$a = |SS|, SS = \{(x_i, x_j) | \lambda_i = \lambda_j, \lambda_i{}^* = \lambda_j{}^*, i < j\}$$

$$b = |SD|, SS = \{(x_i, x_j) | \lambda_i = \lambda_j, \lambda_i{}^* \neq \lambda_j{}^*, i < j\}$$

$$c = |DS|, SS = \{(x_i, x_j) | \lambda_i \neq \lambda_j, \lambda_i{}^* = \lambda_j{}^*, i < j\}$$

$$d = |DD|, SS = \{(x_i, x_j) | \lambda_i \neq \lambda_j, \lambda_i{}^* \neq \lambda_j{}^*, i < j\}$$

$$a + b + c + d = C_m^2 = m(m-1)/2$$

External Index

- JC: Jaccard Coefficient

$$JC = \frac{a}{a+b+c}$$

- FMI: Fowlkes and Mallows Index

$$\text{FMI} = \sqrt{\frac{a}{a+b} \cdot \frac{a}{a+c}}$$

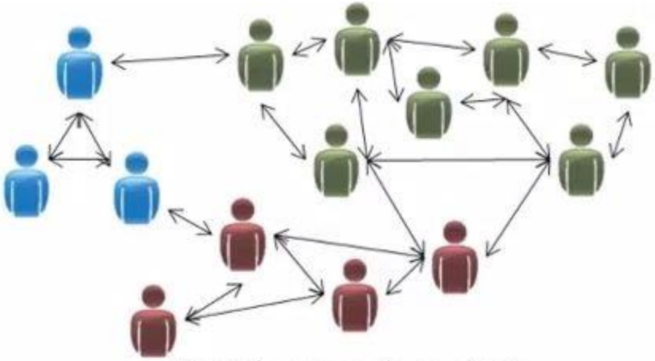
- RI: Rand Index

$$RI = \frac{2(a+d)}{m(m-1)}$$

[0,1] interval
the bigger
the better

Applications

Organize computing clusters



Social network analysis

Market segmentation

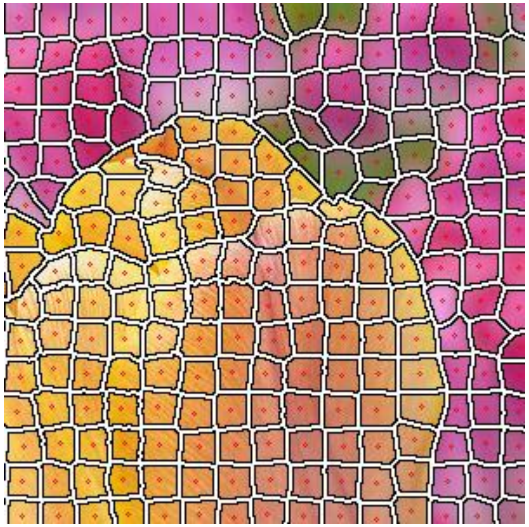
Image credit: NASA/JPL-Caltech/E. Churchwell (Univ. of Wisconsin, Madison)

Astronomical data analysis

https://mf.mbd.baidu.com/govil5e?f=cp

An Example

- Superpixel segmentation uses the similarity of features between pixels to group pixels, and replaces a large number of pixels with a small number of superpixels to express image features.



http://www.cnblogs.com/Imageshop/p/6193433.html

Thanks !