聚类分析

MCM 25组
陈鼎 孟诗涵 洪韫妍
2017.12.14

目录

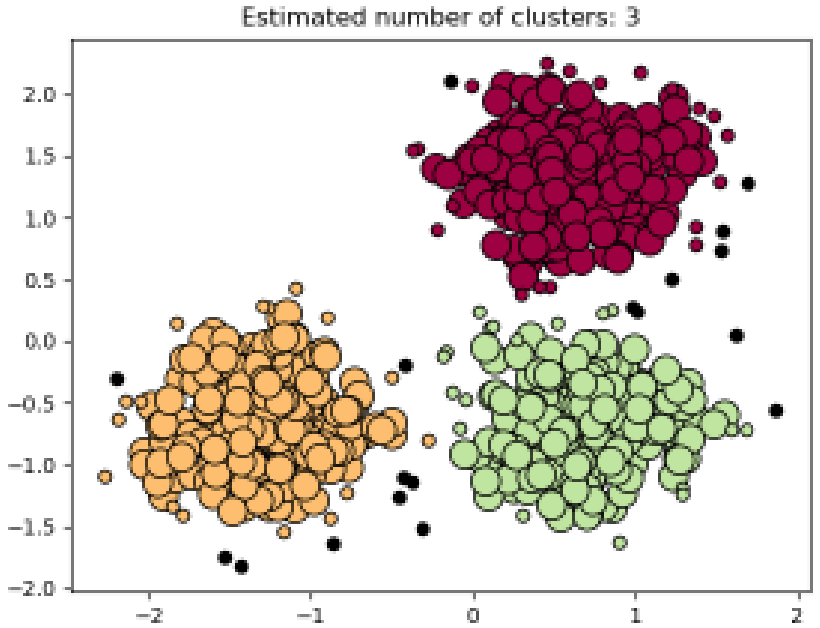
- 聚类预备内容
- K-Means, K-Means++
- 层次聚类
- SOM聚类
- EM算法
- 美赛真题

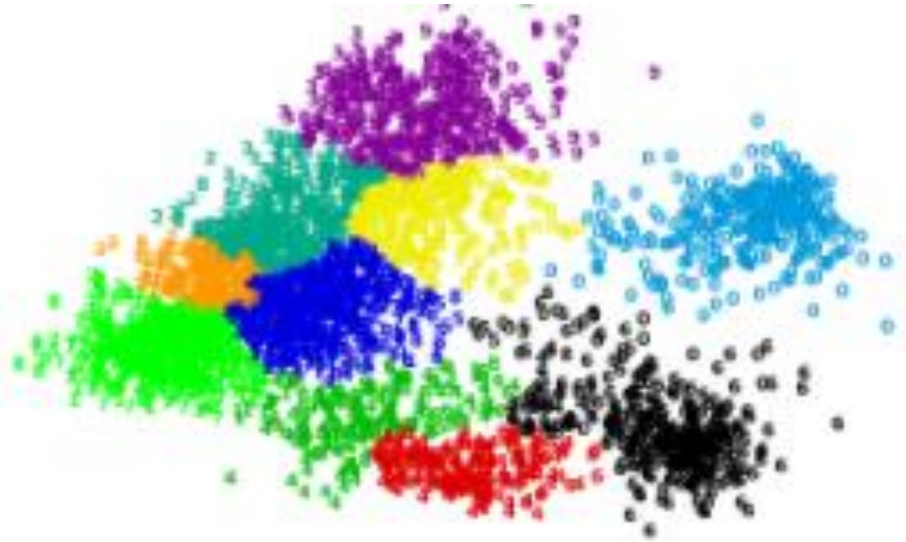
聚类是什么？

- 无监督学习——无标记
- 将相似的事物聚集在一起，而将不相似的事物划分到不同的类别的过程
- 聚类试图将数据集中的样本划分为若干个通常是不相交的子集,每个子集称为一个“簇”。

聚类是什么？

- 假定样本集D=$\{x_1, x_2, \ldots, x_m\}$包含$m$个无标记样本，每个样本$x_i = (x_{i1}; x_{i2}; \ldots; x_{im})$是一个$m$维特征向量，则聚类算法将样本集$D$划分为$k$个不相交的簇$\{C_l \mid l = 1,2, \ldots, k\}$，其中$C_{l'} \cap C_{l\,(l' \neq l)} = \emptyset$且$D = \cup_{l=1}^{k} C_l$。





聚类用来做什么？

- 洞察数据分布的独立工具
- 可以作为数据（算法）的预处理
  - 分类、模式识别、假设生成、测试
  - 数据摘要、数据压缩、数据降维
  - 协同过滤
  - 动态趋势检测
  - 用于多媒体数据、生物数据、社交网络数据的应用

聚类的分类

- 划分聚类法
  - K-Means，K-Means++，CLARA，CLARANS
- 层次聚类法
  - CURE算法，ROCK算法，BIRCH算法
- 基于密度的方法
  - DBSCAN算法，GDBSCAN算法，OPTICS算法，FDC算法
- 基于网格的方法
  - BANG算法，WaveCluster算法，STING算法
- 基于模型的方法
  - 统计学方法（COBWEB算法）和网络神经方法（SOM算法）

聚类性能度量

- 外部指标——与某个参考模型进行比较
- 对数据集$D = \{x_1, x_2, \dots, x_m\}$,假定通过聚类给出的簇划分为$C = \{C_1, C_2, \dots, C_k\}$,参考模型给出的簇的划分为$C^* = \{C_1^*, C_2^*, \dots, C_k^*\}$,相应的$\lambda$与$\lambda^*$分别表示$C$ 和C^*对应的簇标记向量， 我们将样本两两配对考虑， 定义
- $a = |SS|,\ SS = \{(x_i, x_j) |\ \lambda_i = \lambda_j, \lambda_i^* = \lambda_j^*, i < j\}$
- $b = |SD|,\ SD = \{(x_i, x_j) |\ \lambda_i = \lambda_j, \lambda_i^* \neq \lambda_j^*, i < j\}$
- $c = |DS|,\ DS = \{(x_i, x_j) |\ \lambda_i \neq \lambda_j, \lambda_i^* = \lambda_j^*, i < j\}$
- $d = |DD|,\ DD = \{(x_i, x_j) |\ \lambda_i \neq \lambda_j, \lambda_i^* \neq \lambda_j^*, i < j\}$

聚类性能度量

- Jaccard系数（简称JC）

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

- FM指数（简称FMI）

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

- RAND指数（简称RI）

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

上述性能度量的结果均在[0,1]区间，值越大越好！

聚类性能度量

- 内部指标——直接考察聚类结果，而不利用任何参考模型
- 考虑聚类结果 $C = \{C_1, C_2, \dots, C_k\}$，定义
- $avg(C) = \frac{2}{|C|(|C|-1)} \sum_{1 \ll i < j \ll |C|} dist(x_i, x_j)$
- $diam(C) = max_{1 \ll i < j \ll |C|} dist(x_i, x_j)$
- $d_{min}(C_i, C_j) = min_{x_i \in Ci, x_j \in C_j} dist(x_i, x_j)$
- $d_{cen}(C_i, C_j) = dist(\mu_i, \mu_j)$

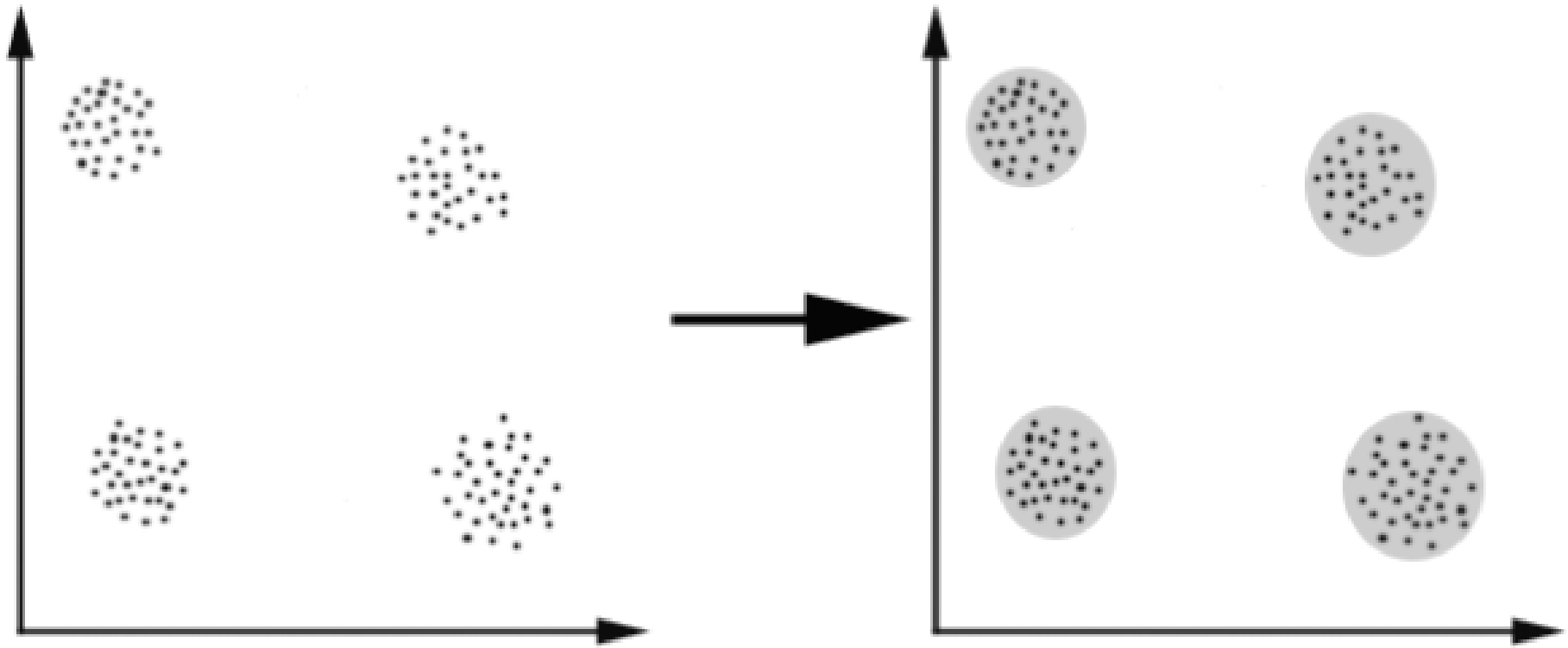
聚类性能度量

- DB指数(简称DBI)
- $DBI = \frac{1}{k}\sum_{i=1}^{k} max_{j \neq i}\left(\frac{avg(C_i)+avg(C_j)}{d_{cen}(\mu_i,\mu_j)}\right)$
- Dunn指数(简称DI)
- $DI = min_{1 \ll i \ll k}\left\{min_{j \neq i}\left(\frac{d_{min}(C_i+C_j)}{max_{1 \ll l \ll k} diam(C_l)}\right)\right\}$
- DBI值越小越好
- DI值越大越好

距离度量

- 点与点之间
- 有序属性最常用的是欧式距离
- $dist_{ed}(x_i, x_j) = ||x_i, x_j||_2 = \sqrt{\sum_{u=1}^{n} |x_{iu}, x_{ju}|^2}$
- 簇和簇之间
- Single Linkage — 两个簇中距离最近的两个数据点的距离
- Complete Linkage — 两个簇中距离最远的两个数据点的距离
- Average Linkage —两个簇中所有点之间的距离的平均

K-Means算法



K-Means算法

- 以k为参数，将n个对象分为k个簇

- 给定样本集D=$\{x_1, x_2, \ldots, x_m\}$，聚类算法将样本集$D$划分为$k$个不相交的簇 $\{C_l \mid l = 1,2, \ldots, k\}$
- 并且最小化平方误差$E = \sum_{i=1}^{k} \sum_{x \in C_i} ||x - \mu_i||_2^2$
- 其中$\mu_i = \frac{1}{|C_i|} \sum_{x \in C_i} x$，是簇$C_i$的均值向量

K-Means算法

- 算法：

  1、首先随机选择了k个对象，每个对象初始地代表了一个簇的平均值或者中心

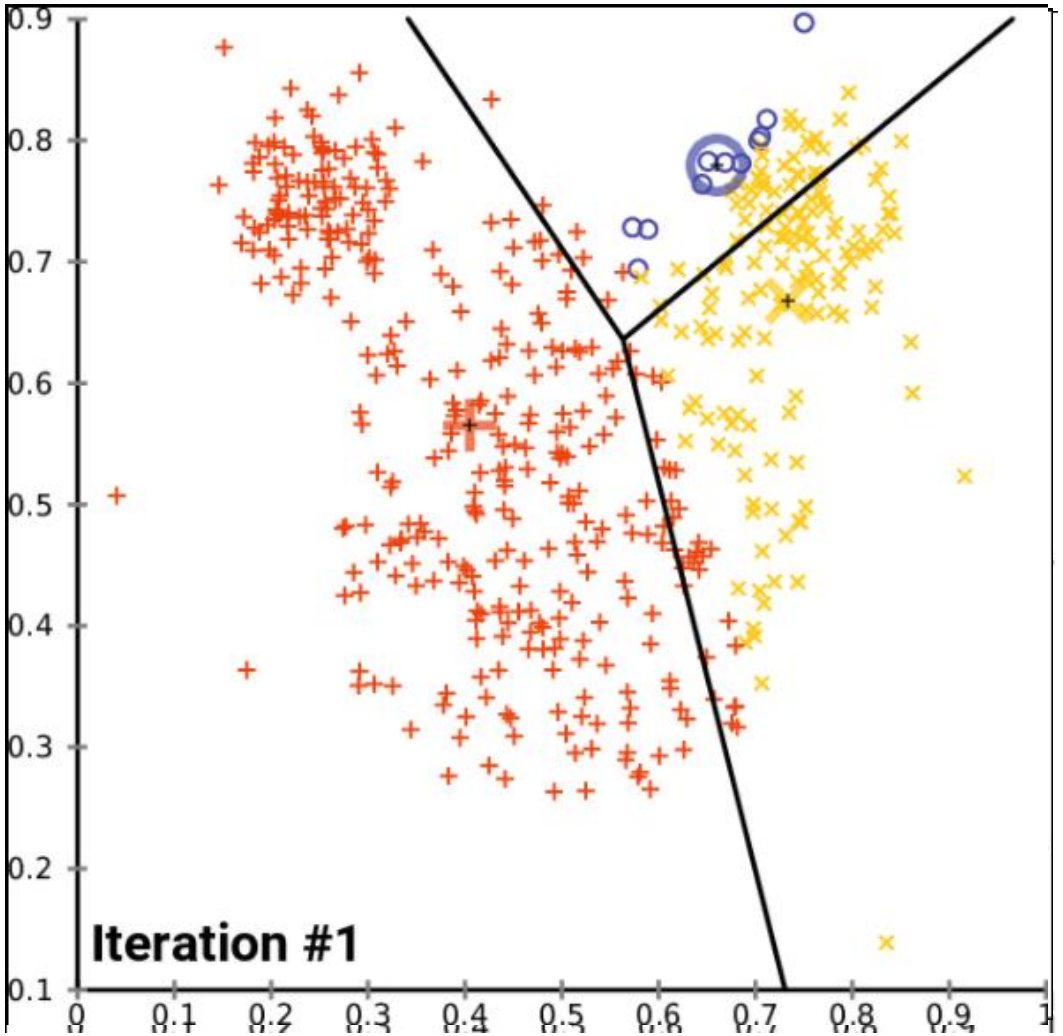
  2、对剩余的每个对象，根据其与各簇中心的距离，将它赋给最近的簇

  3、更新计算每个簇均值向量（μ_i）

  4、重复2,3，直到达到停止条件（达到了指定的最大迭代次数，或者是算法已经收敛，即各个簇的质心不再发生变化。）

- K-Means通常使用随机初始化，所以不会产生相同的结果。
  - 即初始化对结果影响大

K-Means算法



K-Means++

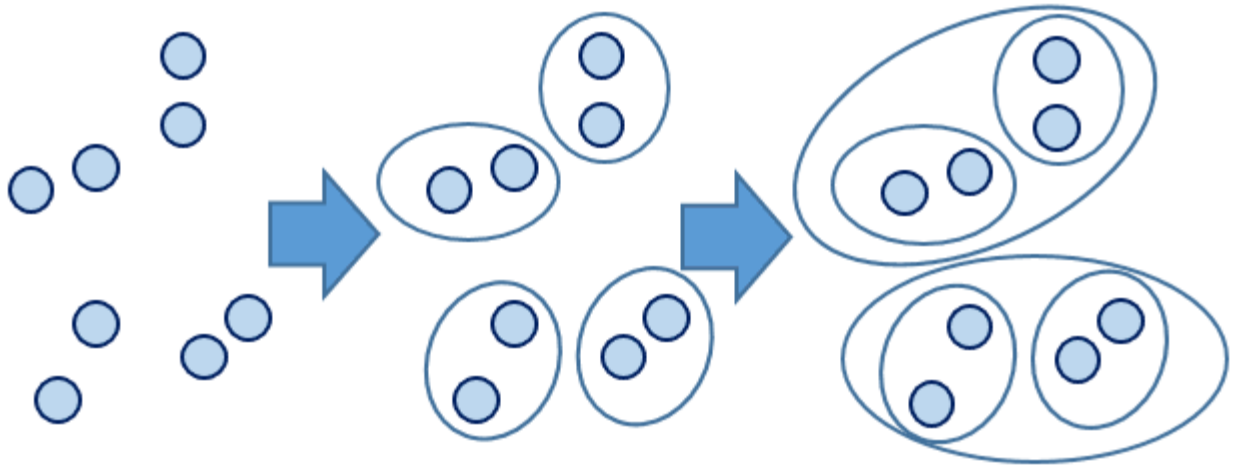
- K-Means算法中，初始点的位置（seeds）对结果有很大的影响。
- K-Means++就是一种初始化选seeds的算法
- 基本思想：初始的聚类中心之间的相互距离要尽可能的远。

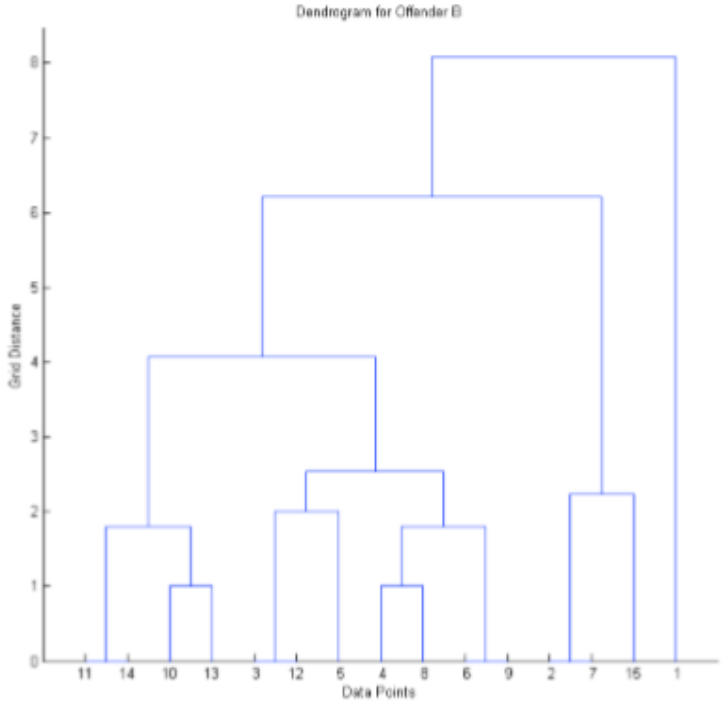
K-Means++

- 算法：
  - 从输入的数据点集合中随机选择一个点作为第一个聚类中心
  - 对于数据集中的每一个点x，计算它与最近聚类中心(指已选择的聚类中心)的距离D(x)
  - 选择一个新的数据点作为新的聚类中心，选择的原则是：D(x)较大的点，被选取作为聚类中心的概率较大
  - 重复2和3直到k个聚类中心被选出来
  - 利用这k个初始的聚类中心来运行标准的k-means算法

层次聚类

- 层次聚类(hierarchical clustering)通过计算不同类别数据点间的相似度来创建一棵有层次的嵌套聚类树。在聚类树中，不同类别的原始数据点是树的最低层，树的顶层是一个聚类的根节点。创建聚类树有自下而上合并和自上而下分裂两种方法。





层次聚类

- **Bisecting k-means二分k均值聚类算法（自上而下）**
- 基本思想是，通过引入局部二分试验，每次试验都通过二分 具有最大SSE值的一个簇，二分这个簇以后得到的2个子簇，选择2个子簇的总SSE最小的划分方法，这样能够保证每次二分得到的2个簇是比较优的（也可能是最优的）

$$\mathrm{SSE}=\sum_{m=1}^{k}\sum_{p_i\in C_i} dist(p_i,c_i)^2=\sum_{m=1}^{k}\sum_{p_i\in C_i}\sum_{\mathrm{j}=1}^{\mathrm{n}_{C_i}}(p_{ij}-c_{ij})^2$$

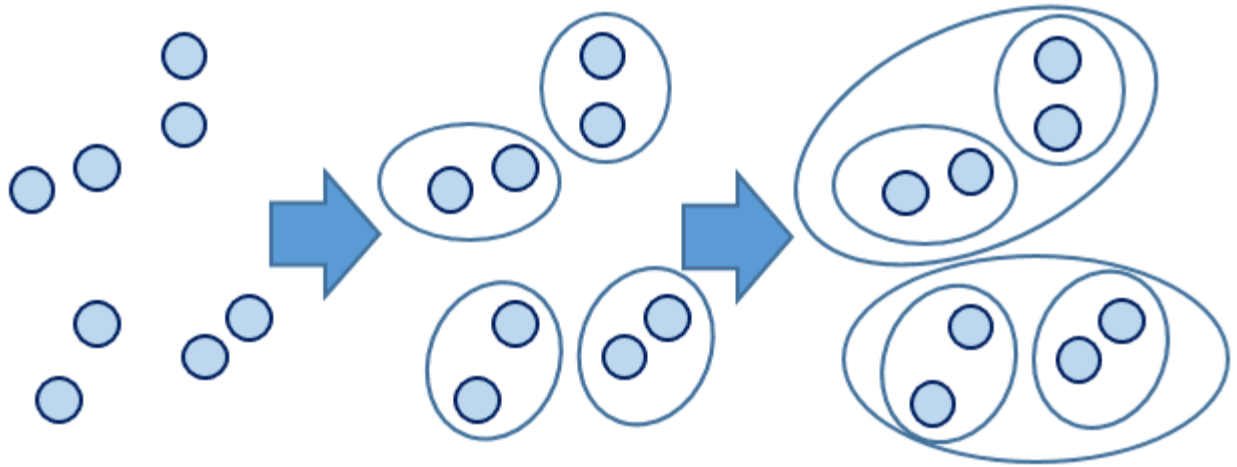
误差平方和(SSE)计算公式

层次聚类

1. 初始时，将待聚类数据集D作为一个簇C_0，即$C = \{C_0\}$，输入参数为：二分试验次数m、k-means聚类的基本参数；
2. 取C中具有最大SSE的簇C_p，进行二分试验m次：调用k-means聚类算法，取$k = 2$，将C_p分为2个簇：C_{i1}, C_{i2}，一共得到m个二分结果集合$B = \{B_1, B_2, \dots, B_m\}$，其中，$B_i = \{C_{i1}, C_{i2}\}$，这里$C_{i1}$和$C_{i2}$为每一次二分试验得到的2个簇；
3. 计算上一步二分结果集合B中，每一个划分方法得到的2个簇的总SSE值，选择具有最小总SSE的二分方法得到的结果：$B_j = \{C_{j1}, C_{j2}\}$，并将簇C_{j1}, C_{j2}加入到集合C，并将C_p从C中移除；
4. 重复步骤2和3，直到得到k个簇，即集合C中有k个簇。

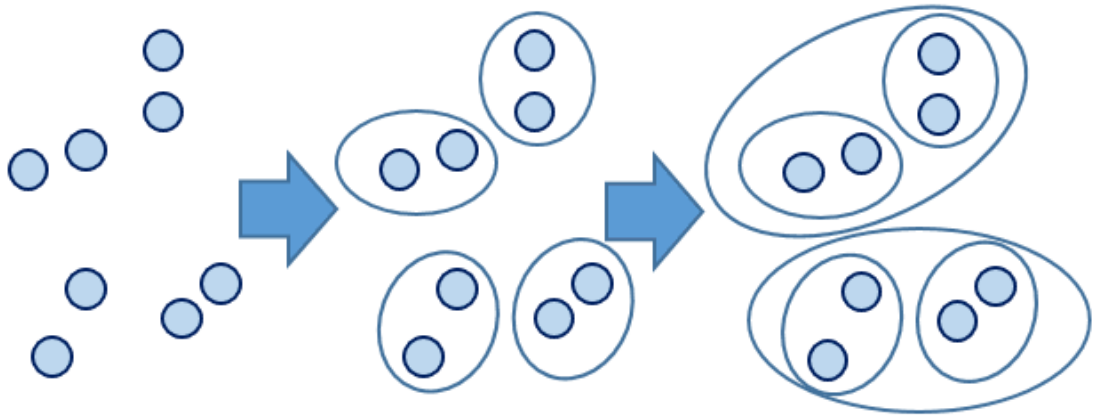
层次聚类

- **Agglomerative Hierarchical Clustering (AHC)合成聚类算法（自下而上）**
- 基本思想是，通过计算两类数据点间的相似性，对所有数据点中最为相似的两个数据点进行组合，并反复迭代这一过程



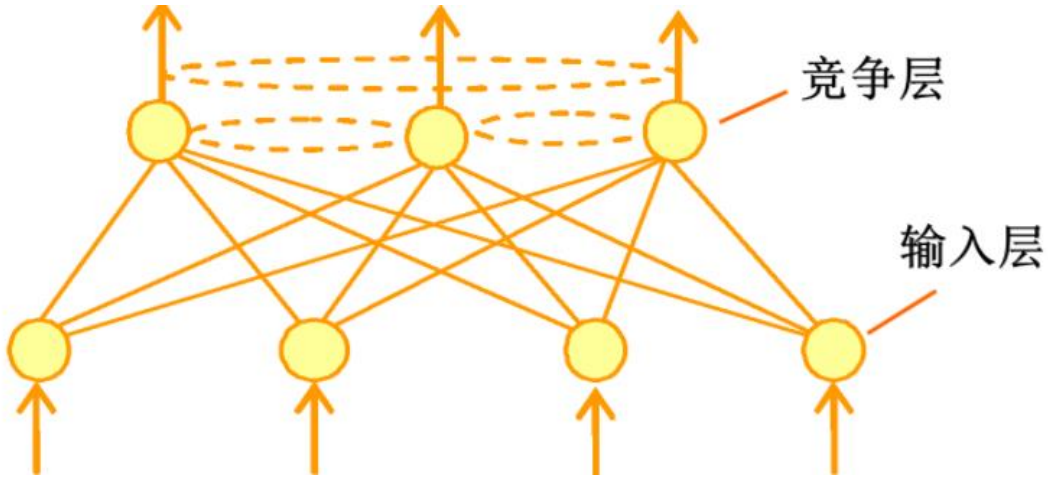
层次聚类

1. 初始化,把每个样本归为一类，计算每两个类之间的距离，也就是样本与样本之间的相似度
2. 寻找各个类之间最近的两个类，把他们归为一类——这样类的总数就少了一个
3. 重新计算新生成的这个类与各个旧类之间的相似度
4. 重复2和3直到所有样本点都归为一类，结束。



SOM聚类

- 自组织特征映射SOM网络
- 一般的用法是将高维的input数据在低维的空间表示，因此SOM天然是一种降维方法。除了降维，SOM还可以用于数据可视化，以及聚类等应用中。
- 自组织神经网络的典型结构

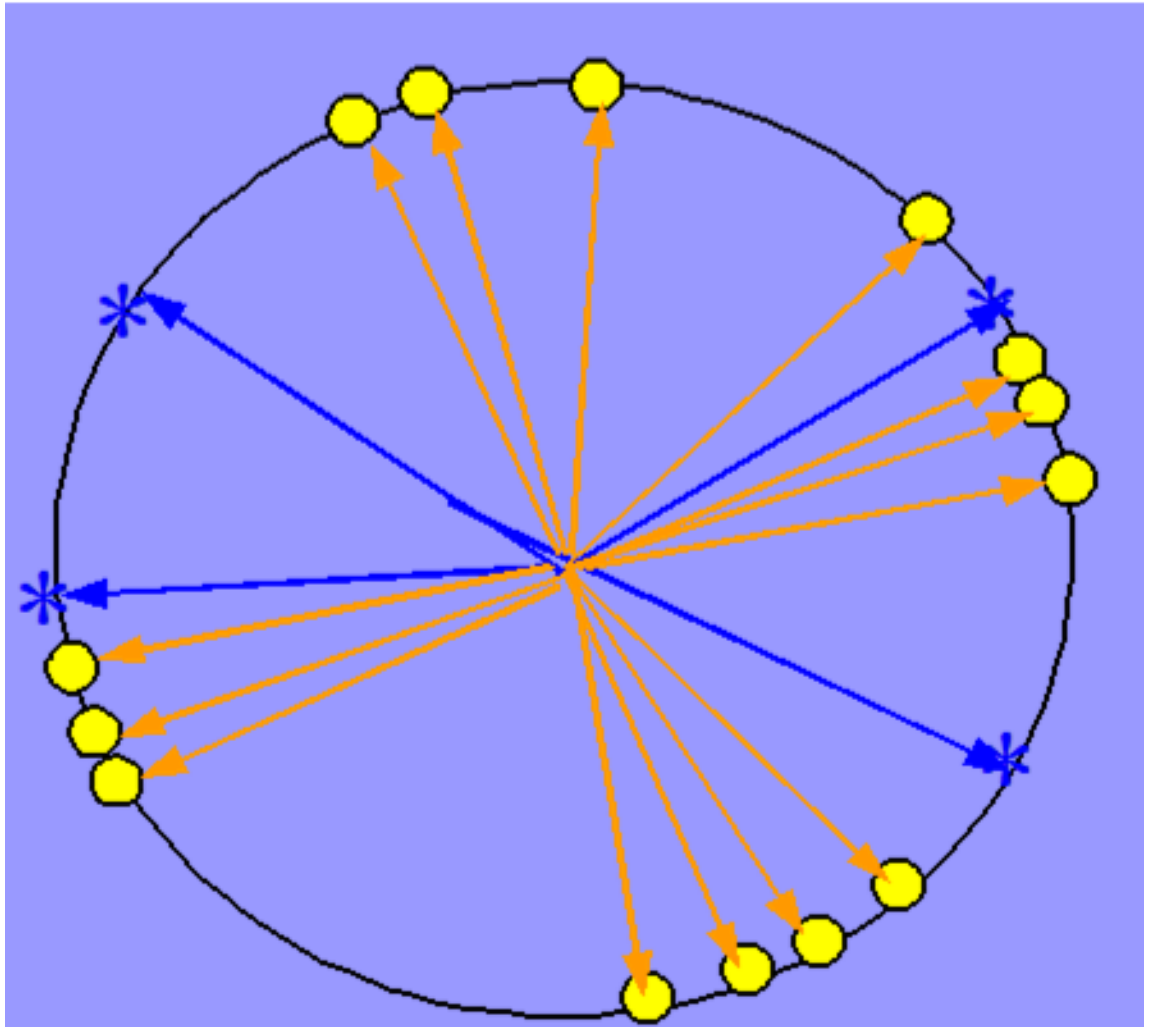


- 聚类的时候也可以看成将目标样本分类，只是是没有任何先验知识的，目的是将相似的样本聚合在一起，而不相似的样本分离。

SOM聚类

- **竞争学习规则——Winner-Take-All**
  - 网络的输出神经元之间相互竞争以求被激活，结果在每一时刻只有一个输出神经元被激活。这个被激活的神经元称为竞争获胜神经元，而其它神经元的状态被抑制
- 首先，对网络当前输入模式向量X和竞争层中各神经元对应的权重向量Wj（对应j神经元）全部进行归一化，使得X和Wj模为1；当网络得到一个输入模式向量X时，竞争层的所有神经元对应的权重向量均与其进行相似性比较，并将最相似的权重向量判为竞争获胜神经元。
- 竞争学习的步骤是：
  - 向量归一化 + 寻找获胜神经元 + 网络输出与权值调整

SOM聚类



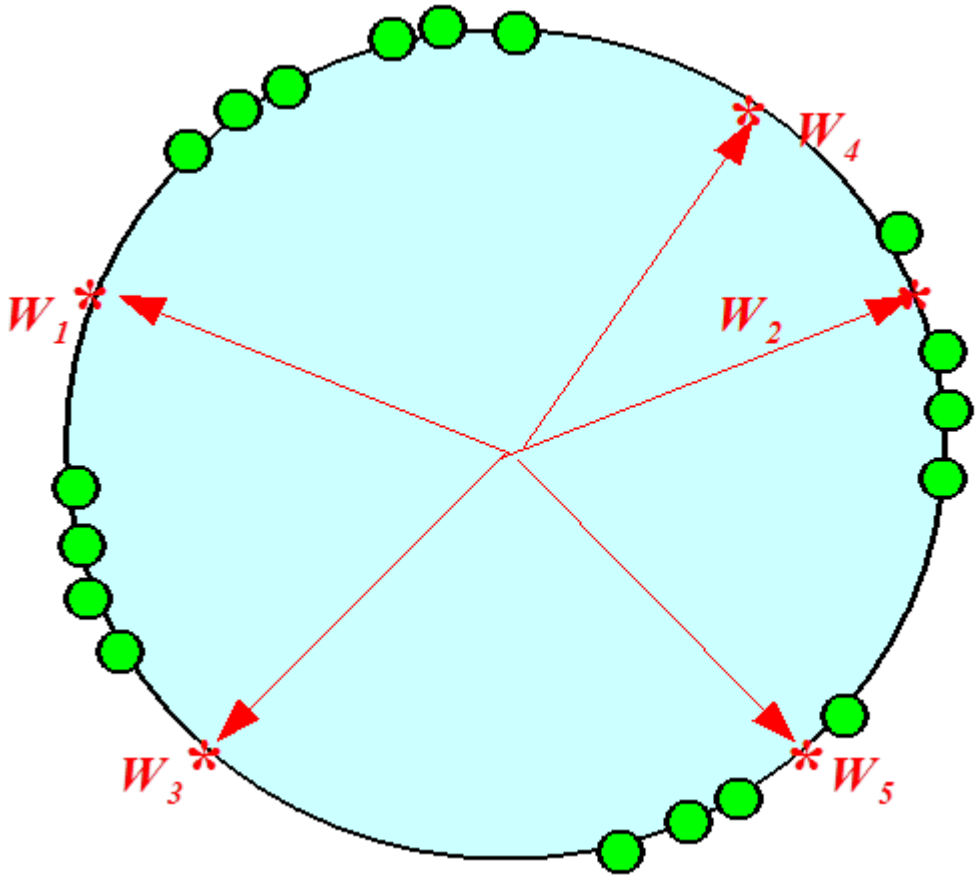
SOM聚类

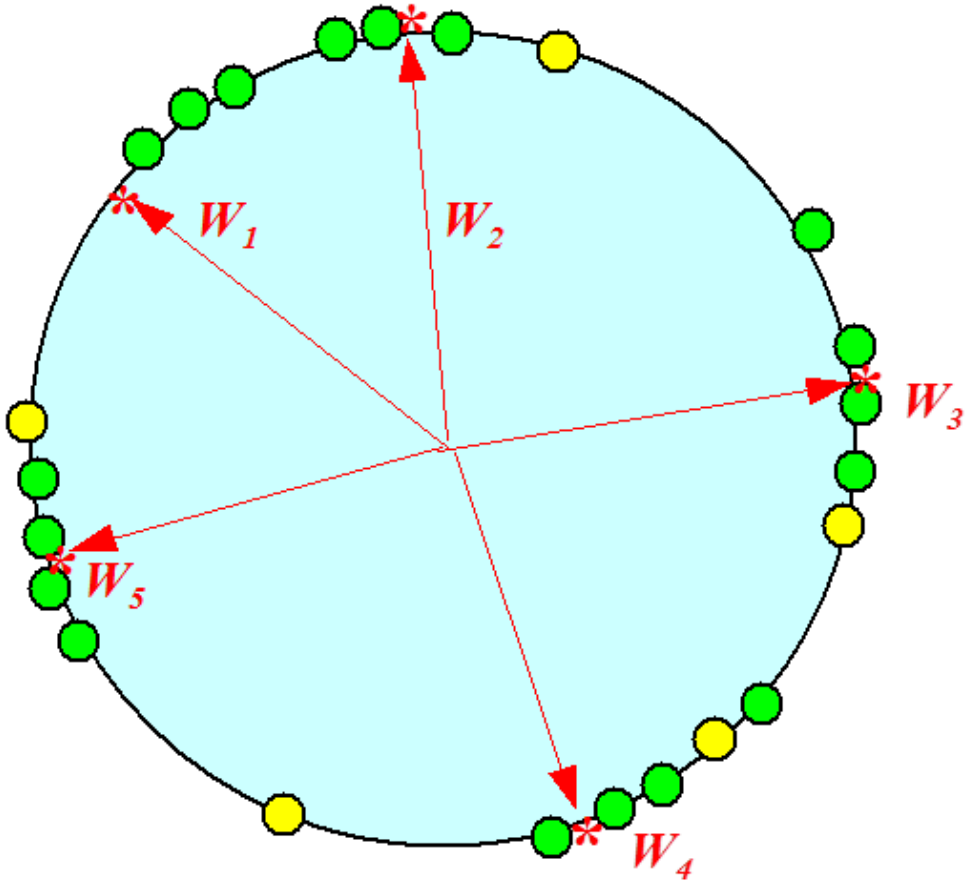
1. 初始化：每个节点随机初始化自己的参数。每个节点的参数个数与Input的维度相同。
2. 对于每一个输入数据，找到与它最相配的节点。假设输入时D维的，即 $X_{\overline{D}} = \{x_i, i = 1, \ldots, D\}$，那么判别函数可以为欧几里得距离：$d_j(\mathbf{x}) = \sum_{i=1}(x_i - w_{ji})^2$
3. 找到激活节点$I(x)$之后，我们也希望更新和它临近的节点。令S_{ij}表示节点i和j之间的距离，对于$I(x)$临近的节点，分配给它们一个更新权重 $T_{j,I(\mathbf{x})} = \exp(-S_{j,I(\mathbf{x})}^2 / 2\sigma^2)$ (简单地说，临近的节点根据距离的远近，更新程度要打折扣。)
4. 接着就是更新节点的参数了。按照梯度下降法更新：
5. 迭代，直到收敛。

$$\Delta w_{ji} = \eta(t) \cdot T_{j,I(\mathbf{x})}(t) \cdot (x_i - w_{ji})$$

SOM聚类

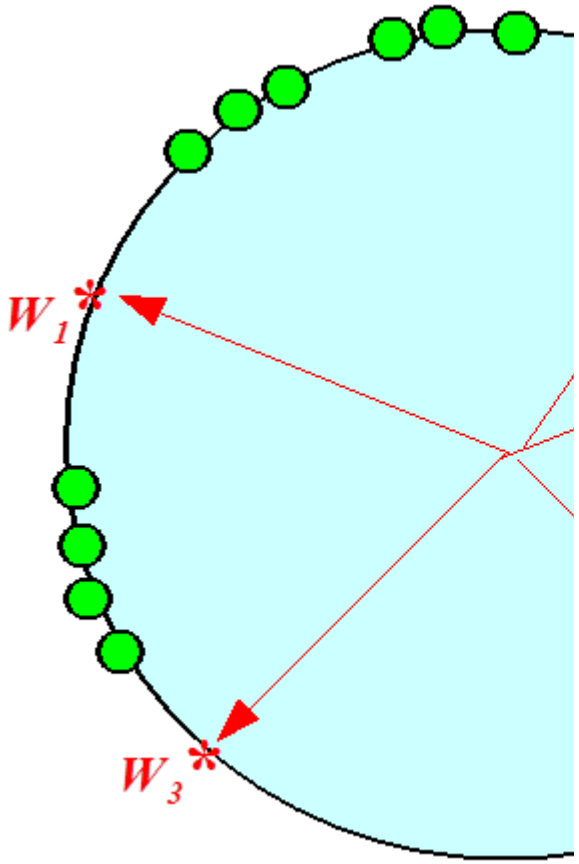
训练数据（绿点）和神经元权重初始值（红花）



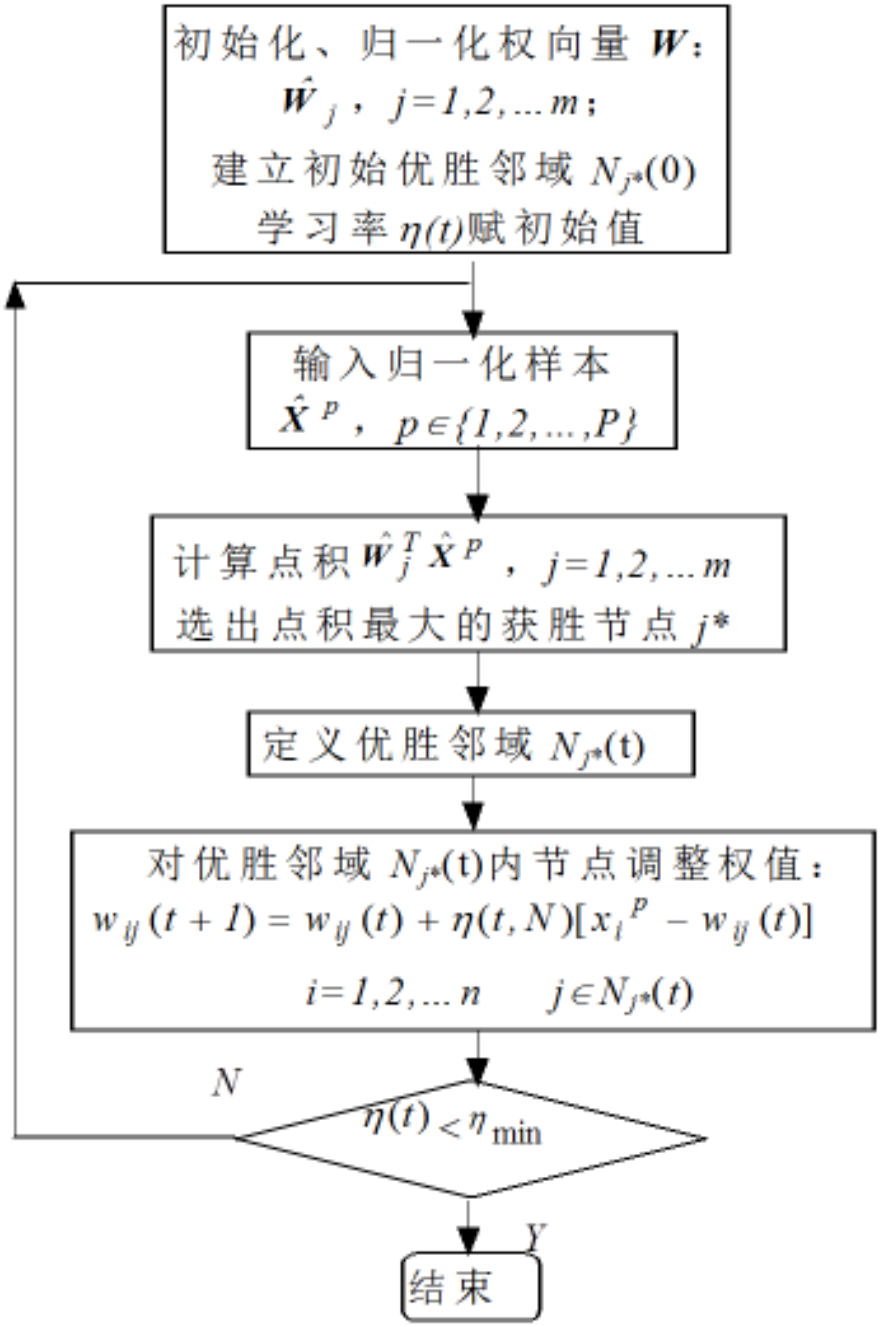


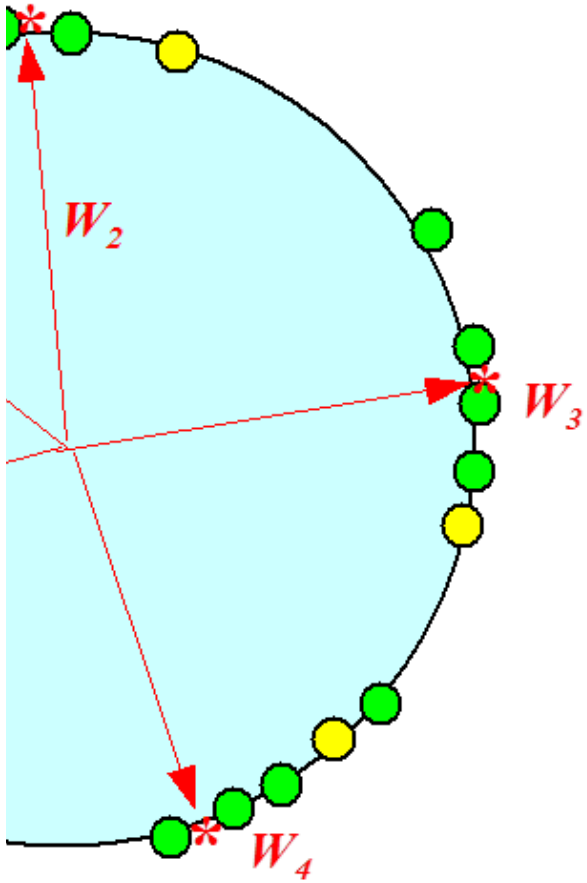
SOM聚类

训练数据



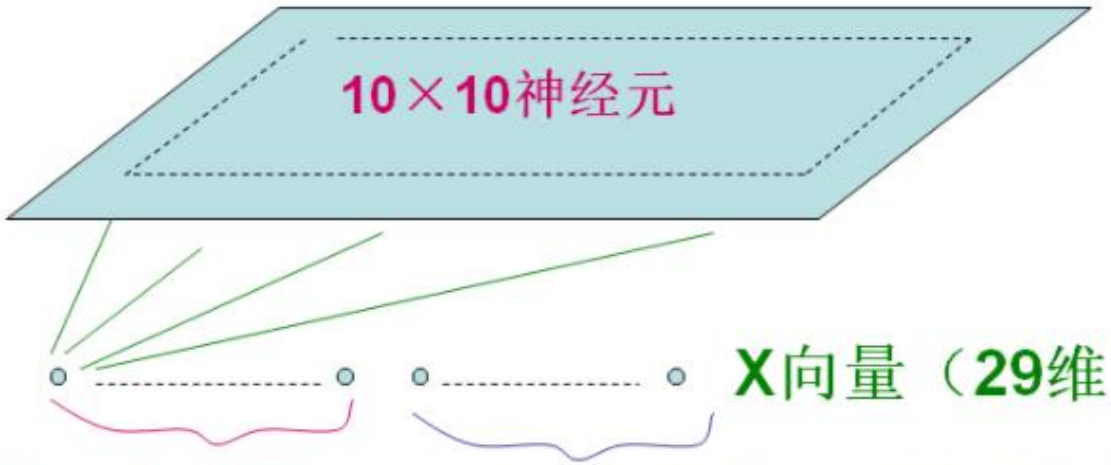
算法流程总结





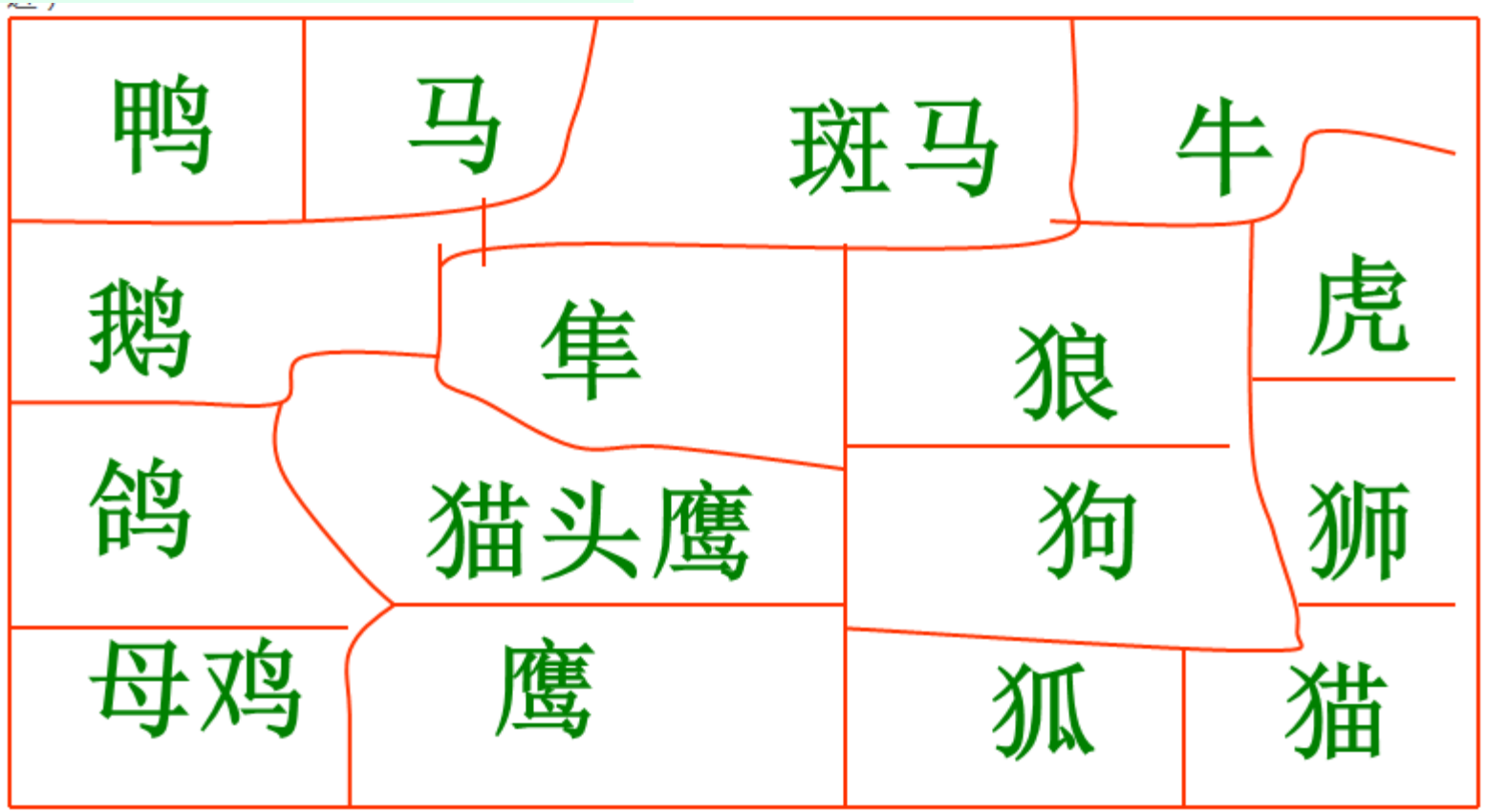
SOM聚类

动物 / 属性	鸽子	母鸡	鸭	鹅	猫头鹰	隼	鹰	狐狸	狗	狼	猫	虎	狮	马	斑马	牛
小	1	1	1	1	1	1	0	0	0	0	1	0	0	0	0	0
中	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0
大	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
2只腿	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
4只腿	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
毛	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
蹄	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
鬃毛	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
羽毛	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
猎	1	0	0	0	1	1	1	1	0	1	1	1	1	0	0	0
跑	0	0	0	0	1	1	0	1	0	1	0	1	1	1	1	0
飞	1	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0
泳	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0



SOM聚类

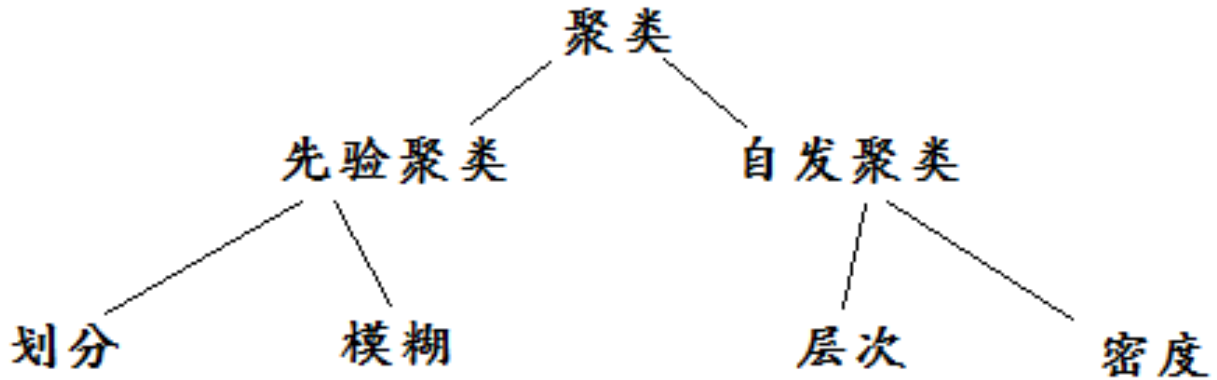
动物 / 属性	鸽子	母鸡	鸭	鹅	猫头鹰	隼
小	1	1	1	1	1	1
中	0	0	0	0	0	0
大	0	0	0	0	0	0
2只腿	1	1	1	1	1	1
4只腿	0	0	0	0	0	0
毛	0	0	0	0	0	0
蹄	0	0	0	0	0	0
鬃毛	0	0	0	0	0	0
羽毛	1	1	1	1	1	1
猎	0	0	0	0	1	1
跑	0	0	0	0	0	0
飞	1	0	0	1	1	1
泳	0	0	1	1	0	0



量（29维）

（表达动物属性）

聚类算法对比



先验聚类的优点：

1. 精确度高
2. 有明确优化目标

先验的缺点：

1. 速度慢
2. 噪音敏感
3. 难以确定K值

自发聚类和先验聚类正好相异

EM算法

- EM算法，全称Expectation Maximization Algorithm，即最大期望化算法或期望最大算法，它是一种迭代算法，用于含有隐变量（hidden variable）的概率参数模型的最大似然估计或极大后验概率估计。

举个例子

- 极大似然估计
  - 有一个硬币A，我要求抛A得到正面的概率。已知我随手抛了10次，3次正面，7次反面。
  - $L(p) = p^3 * (1-p)^7$
  - p指的是抛A为正面的概率
- EM算法
  - 有一个硬币A和一个硬币B，我要求抛A得到正面的概率以及抛B得到的概率（A,B正面的概率相互独立）。已知我做了五次实验，每次实验过程为随机在A,B中选一个硬币，抛了10次，记录相应的结果。
  - 1 0 1 0 1 0 1 0 1 0
  - 1 1 1 1 1 1 1 1 1 1
  - $L(\theta) = (p(\text{第一次抛的是}A\text{硬币}) * p_A^5 * (1-p_A)^5 + p(\text{第一次抛的是}B\text{硬币}) * p_B^5 * (1-p_B)^5)$*第二次* .....

EM算法

- EM算法是常用的估计参数隐变量的利器。
- 对于上述情况，由于存在隐含变量，不能直接最大化$l(\theta)$，所以只能不断地建立l的下界（E-step），再优化下界（M-step），依次迭代，直至算法收敛到局部最优解。

- E-Step:通过observed data和现有模型估计参数估计值missing data
- M-Step:假设missing data已知的情况下，最大化似然函数。

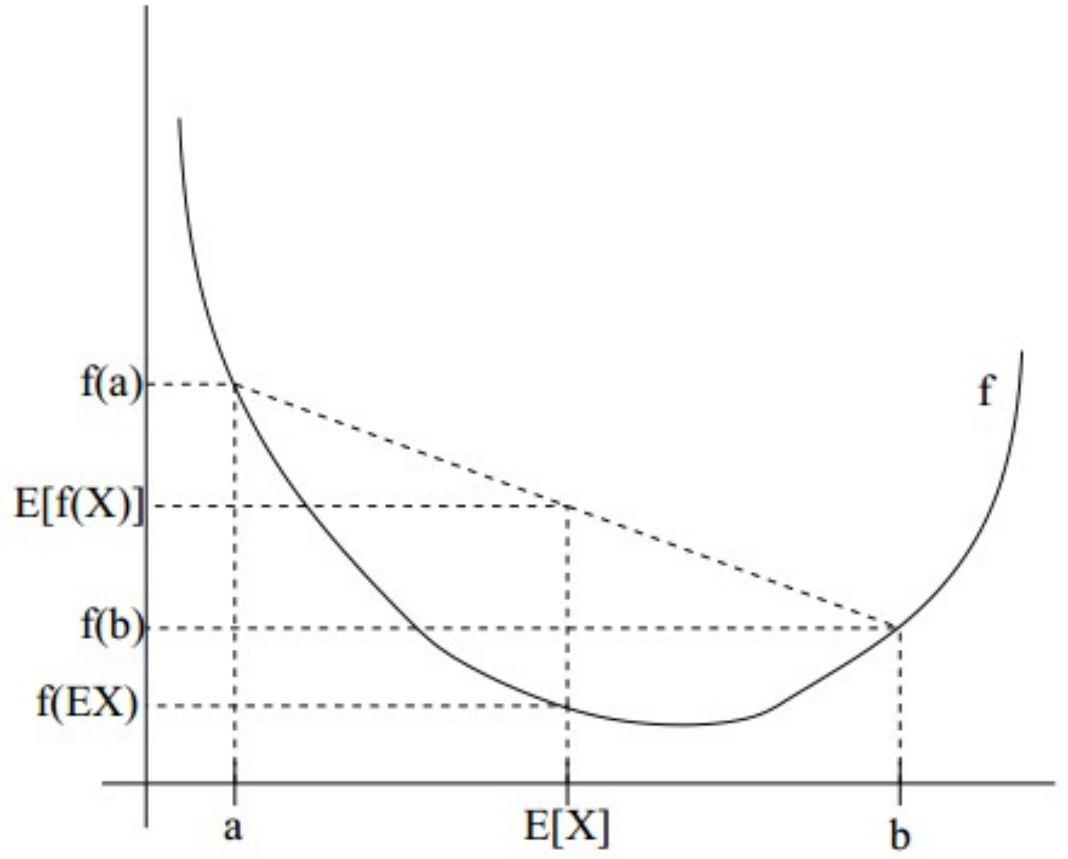
Jensen不等式

- 假设f是定义域为实数的函数
- 如果对于所有的x，$f(x)$的二阶导数大于等于0，那么f是凸函数。
- 当x是向量时，如果hessian矩阵H是半正定（即$H \geq 0$），那么f是凸函数。
- 如果，$f(x)$的二阶导数小于0或者$H > 0$，那么f就是凹函数。

Jensen不等式

Jensen不等式：

- 如果f是凸函数，X是随机变量，则$E[f(X)] \geq f(E[X])$
- 特别地，如果f是严格凸函数，$E[f(X)] \geq f(E[X])$，那么当且仅当$p(x = E[X]) = 1$时（也就是说X是常量），$E[f(x)] = f(E[X])$；
- 如果f是凹函数，X是随机变量，则$f(E[X]) \leq E[f(X)]$.
- 当f是（严格）凹函数当且仅当f是（严格）凸函数。



EM算法

- 最大似然函数估计值的一般步骤：
- （1）写出似然函数；
- （2）对似然函数取对数，并整理；
- （3）求导数，令导数为0，得到似然方程；
- （4）解似然方程，得到的参数即为所求；

EM算法

- 给定m个训练样本$\{x^{(1)},\ldots,x^{(m)}\}$。假设样本间相互独立，我们想要拟合模型$p(x,z)$ 到数据的参数。根据分布，我们可以得到如下这个似然函数

$$\ell(\theta) = \sum_{i=1}^{m} \log p(x;\theta)$$

对极大似然函数取对数

$$= \sum_{i=1}^{m} \log \sum_{z} p(x,z;\theta).$$

每个样本实例的每个可能的类别z求联合分布概率之和

EM算法

$$\sum_i \log p(x^{(i)};\theta) = \sum_i \log \sum_{z^{(i)}} p(x^{(i)}, z^{(i)};\theta) \quad (1)$$

$$= \sum_i \log \sum_{z^{(i)}} Q_i(z^{(i)}) \frac{p(x^{(i)}, z^{(i)};\theta)}{Q_i(z^{(i)})} \quad (2)$$

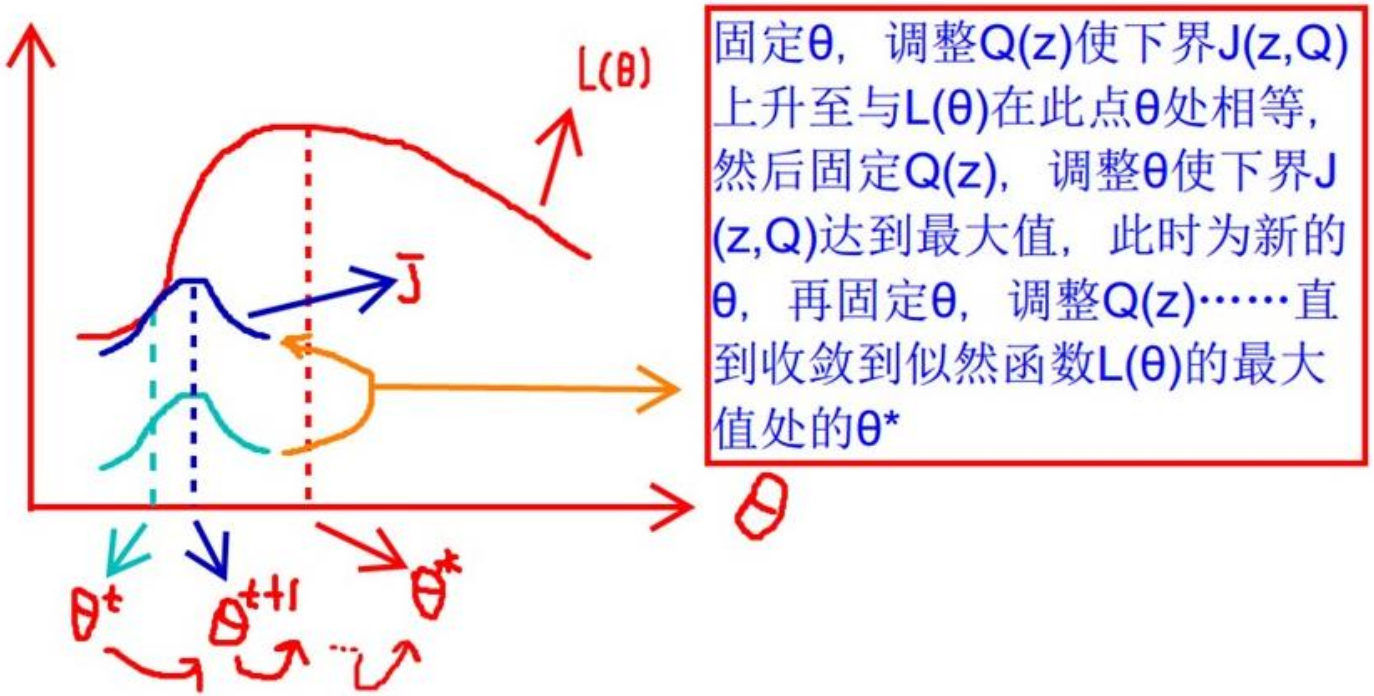
Jensen不等式

$$\geq \sum_i \sum_{z^{(i)}} Q_i(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)};\theta)}{Q_i(z^{(i)})} \quad (3)$$

对于每个实例i,用Q_i表示样本实例隐含变量z的某种分布

EM算法

- 似然函数$L(\theta) \geq J(z, Q)$ 的形式（z为隐含变量），那么我们可以通过不断的最大化J的下界，来使得$L(\theta)$ 不断提高，最终达到它的最大值。使用下图会比较形象：



EM算法

E-step

$$
\begin{aligned}
Q_i(z^{(i)}) &= \frac{p(x^{(i)}, z^{(i)}; \theta)}{\sum_z p(x^{(i)}, z; \theta)} \\
&= \frac{p(x^{(i)}, z^{(i)}; \theta)}{p(x^{(i)}; \theta)} \\
&= p(z^{(i)} | x^{(i)}; \theta)
\end{aligned}
$$

调整$Q(z)$使下界$J(z,Q)$上升至与$L(\theta)$在此点θ处相等

M-step

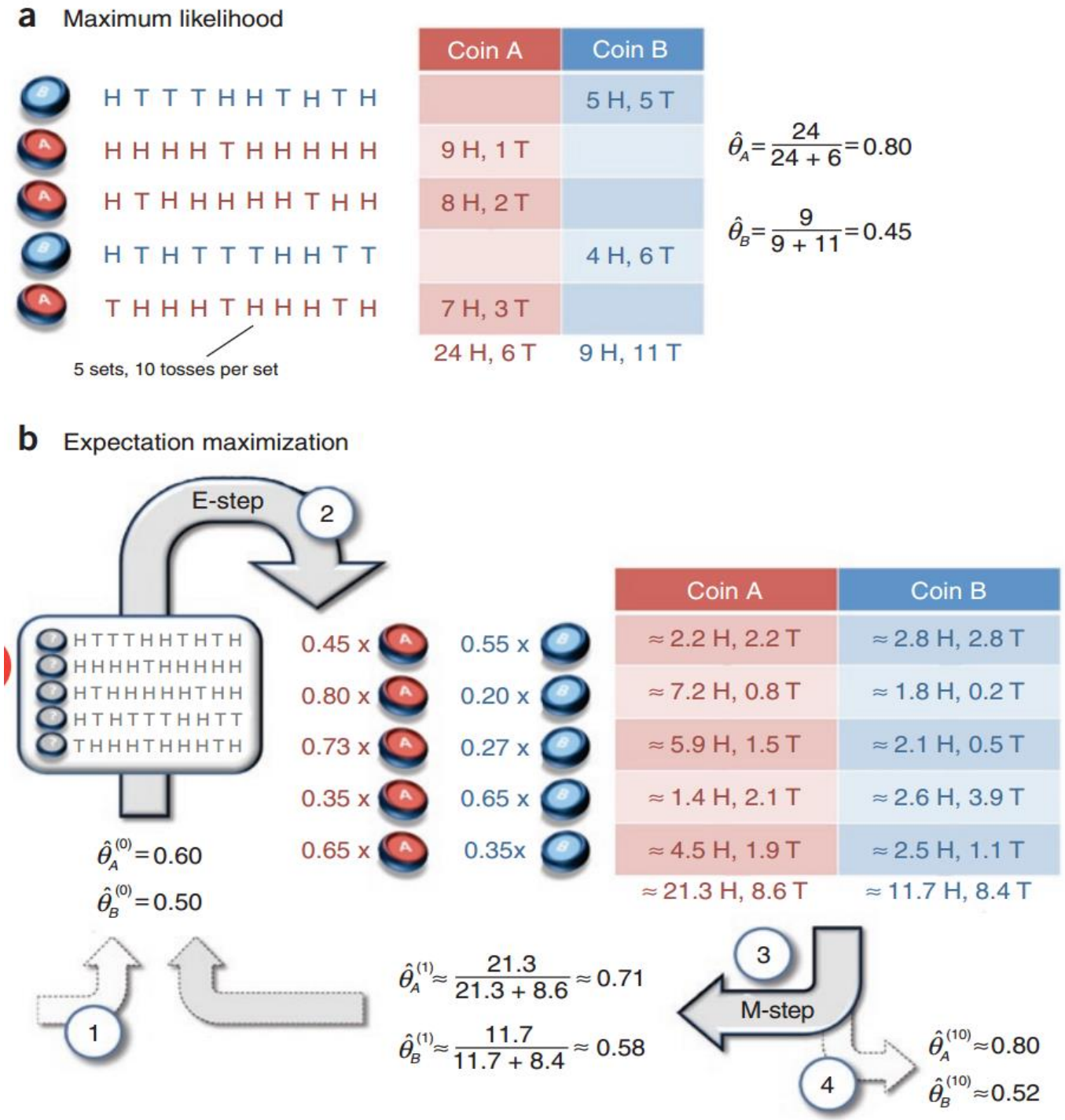
$$
\theta := \arg\max_\theta \sum_i \sum_{z^{(i)}} Q_i(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)}; \theta)}{Q_i(z^{(i)})}.
$$

调整θ求极值

EM算法应用

- 假设有两枚硬币A、B，以相同的概率随机选择一个硬币，进行如下的掷硬币实验：共做5次实验，每次实验独立的掷十次，结果如图中a所示，例如某次实验产生了H、T、T、T、H、H、T、H、T、H，H代表证明朝上。a是在知道每次选择的是A还是B的情况下进行，b是在不知道选择的硬币情况下进行，问如何估计两个硬币正面出现的概率？

EM算法应用


a Maximum likelihood
H T T T H H T H T H
H H H H T H H H H H
H T H H H H H T H H
H T H T T T H H T T
T H H H T H H H T H
5 sets, 10 tosses per set
Coin A
Coin B
5 H, 5 T
9 H, 1 T
8 H, 2 T
4 H, 6 T
7 H, 3 T
24 H, 6 T
9 H, 11 T
$\hat{\theta}_A = \frac{24}{24+6} = 0.80$
$\hat{\theta}_B = \frac{9}{9+11} = 0.45$
b Expectation maximization
E-step
2
HTTTHHTHTH
HHHHTHHHHH
HTHHHHHTHH
HTHTTTHHTT
THHHTHHHTH
0.45 x
0.80 x
0.73 x
0.35 x
0.65 x
0.55 x
0.20 x
0.27 x
0.65 x
0.35x
Coin A
Coin B
≈ 2.2 H, 2.2 T
≈ 2.8 H, 2.8 T
≈ 7.2 H, 0.8 T
≈ 1.8 H, 0.2 T
≈ 5.9 H, 1.5 T
≈ 2.1 H, 0.5 T
≈ 1.4 H, 2.1 T
≈ 2.6 H, 3.9 T
≈ 4.5 H, 1.9 T
≈ 2.5 H, 1.1 T
≈ 21.3 H, 8.6 T
≈ 11.7 H, 8.4 T
$\hat{\theta}_A^{(0)} = 0.60$
$\hat{\theta}_B^{(0)} = 0.50$
1
$\hat{\theta}_A^{(1)} \approx \frac{21.3}{21.3+8.6} \approx 0.71$
$\hat{\theta}_B^{(1)} \approx \frac{11.7}{11.7+8.4} \approx 0.58$
3
M-step
4
$\hat{\theta}_A^{(10)} \approx 0.80$
$\hat{\theta}_B^{(10)} \approx 0.52$

EM算法应用

- K-means聚类算法
- 混合高斯模型（Mixtures of Gaussians）

K-Means算法

- 算法：

1、首先随机选择了k个对象，每个对象初始地代表了一个簇的平均值或者中心

E-step 2、对剩余的每个对象，根据其与各簇中心的距离，将它赋给最近的簇

M-step 3、更新计算每个簇均值向量

4、重复2,3，直到达到停止条件（达到了指定的最大迭代次数，或者是算法已经收敛，即各个簇的质心不再发生变化。）

聚类实例

- **2017年国赛B题——“拍照赚钱”的任务定价**
- 拍照赚钱”是移动互联网下的一种自助式服务模式。用户下载APP，注册成为APP的会员，然后从APP上领取需要拍照的任务（比如上超市去检查某种商品的上架情况），赚取APP对任务所标定的酬金。
- 问题三：实际情况下，多个任务可能因为位置比较集中，导致用户会争相选择，一种考虑是将这些任务联合在一起打包发布。在这种考虑下，如何修改前面的定价模型，对最终的任务完成情况又有什么影响？

聚类实例

- 通过聚类来生成不同的打包情况，来计算不同的任务价格，进行比较，挑选合适的打包方式。

1、聚类分析的结果

我们利用 DBSCAN 算法实现聚类，最终得到了 82 个类，图5中展示了用 MATLAB 绘制的聚类分布图，其中横纵坐标分别为任务聚点的经纬度坐标，不同颜色的十字记号代表了不同的类别，没有聚成类的点在图中没有展示，可以看出，在广东和佛山可打包的任务比较多。

美赛实例

- **2010年美赛B题——犯罪学问题**
- 1981年Peter Sutcliffe(萨克利夫)被判刑因为他参与了十三起谋杀和对其他人的恶毒攻击。缩小搜索Sutcliffe的方法之一是发现一个攻击位置的“质心”.最终犯罪嫌疑人恰好生活在该方法预测的同一个小镇。从那时起，已经发展出一系列更加复杂的技术用来预测基于犯罪地点的具有地理效应（地理轮廓）的系列犯罪行为。
- 你的团队被一个当地警察局要求发展出一种方法用来帮助他们的系列犯罪调查。你们的方法应该至少需要利用两种不同的情景以生成地理效应（地理轮廓），进而根据不同情况下的分析结果对执法人员提供有效的预测。基于以往犯罪的时间和位置，预测信息应该提供一些估计或指导下次可能的犯罪地点。如果在预测中用到了其它的信息，必须提供特别的细节说明告诉我们这些信息是如何被整合的。你们的方法中也应该包括在给定条件下（包括适当警告信息）下预测的可靠性估计。

美赛实例

- Our second method explicitly assumes at least two anchor points (for example, a home and a workplace) and treats each as the centroid of its own local cluster of crimes. This method requires determining an appropriate number of clusters, which we derive from the locations of the previous crimes.
- 这个是针对不知道聚类数目的情况，通过生成聚类树以后，计算聚类评价指标，来检测效果，选择合适的聚类数目。

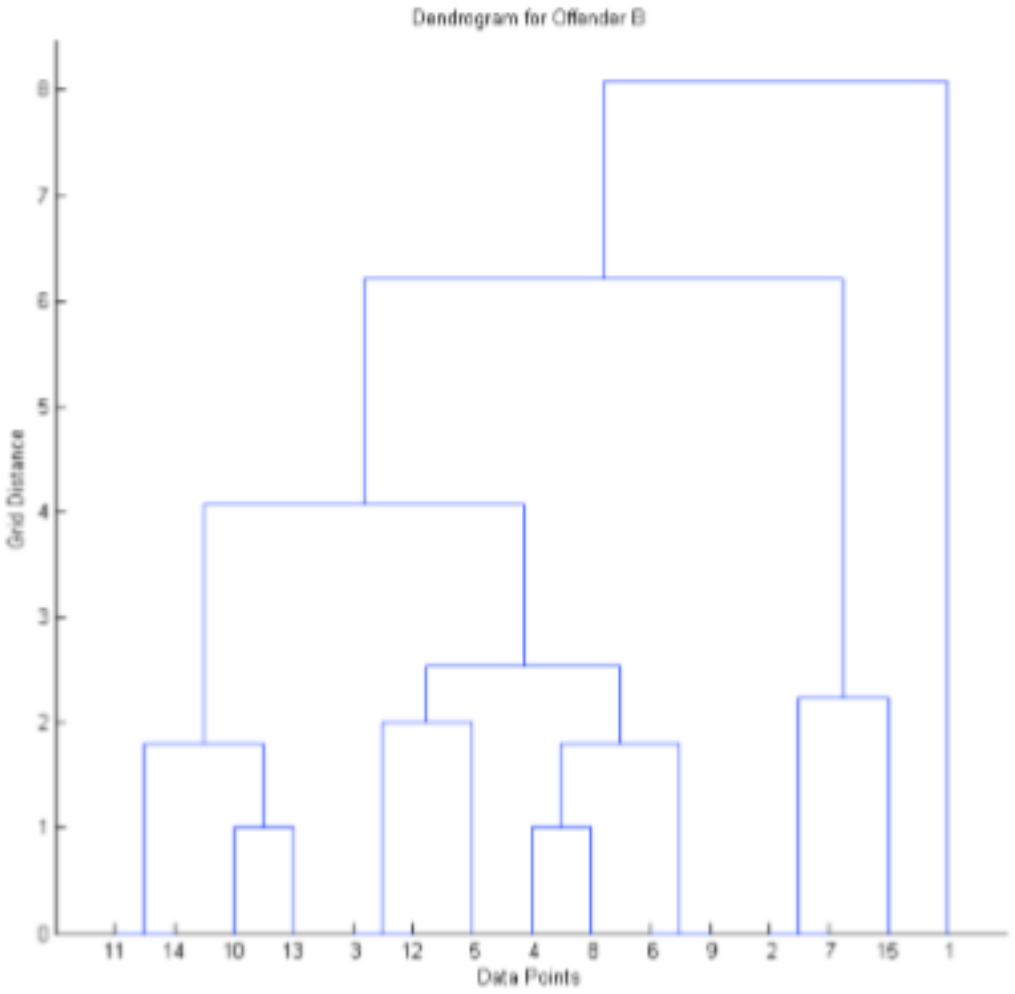
$$s(P_i) = \frac{\left[\min_{k|P_i \notin C_k} b(P_i,k)\right] - a(P_i)}{\max\left(a(P_i), \min_{k|P_i \notin C_k} b(P_i,k).\right)}$$

美赛实例

- We force a minimum of 2 clusters and a maximum of 4. The clustering algorithm is accomplished in a 3-step process。
  - Compute the distances between all crime locations, using the Euclidean distance.
  - Organize the distances into a hierarchical cluster tree, represented by a dendrogram. The cluster tree of data points P1,...,PN is built up by first assuming that each data point is its own cluster.
  - Merge the two clusters that are the closest (in distance between their centroids), and continue such merging until the desired number of clusters is reached. These cluster merges are plotted as the horizontal lines in the dendrogram, and their height is based on the distance between merged clusters at the time of merging.

美赛实例

- We force a mi…um of 4. The clustering algo… process。
  - Compute the… using the Euclidean distance.
  - Organize the…e, represented by a dendrogram… is built up by first assuming tha…
  - Merge the tw… ce between their centroids), ar… red number of clusters is rea… as the horizontal lines in the d… the distance between me…



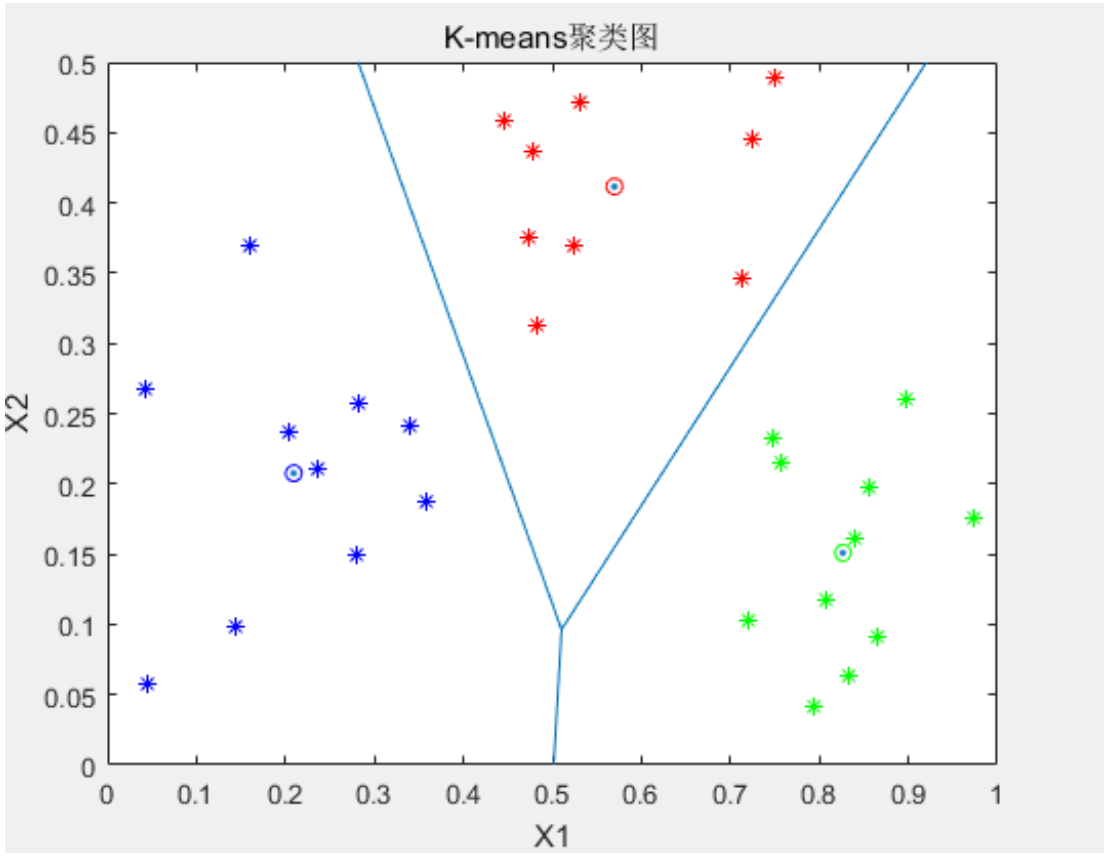
美赛实例

- **2016年美赛C题——优质基金挑战**
- goodgrant基金会是一个慈善组织，其目的是提高高校就读的美国本科生教育绩效。为此，基金会计划从2016年7月开始的五年中每年捐赠1亿美元到符合条件的学校。在这样做时，他们不想重复投资和关注其他大型的捐赠组织如盖茨基金会和基金会所。
- goodgrant基金会要求你的团队建立一个模型来确定最优投资策略，以确定需要投资的学校、每个学校的投资额、这项投资的回报、对学生成绩有显著的正向影响所需要持续的投资时间。这一策略应该包含一个1到N的最优化，以及优先推荐学校的列表，而这些学校的选择是基于每个候选学校所表明的能有效利用私人资金投资、有潜力的学校候选名单。此外，你的策略还应包括适合诸如goodgrant基金组织投资的预估的投资回报。

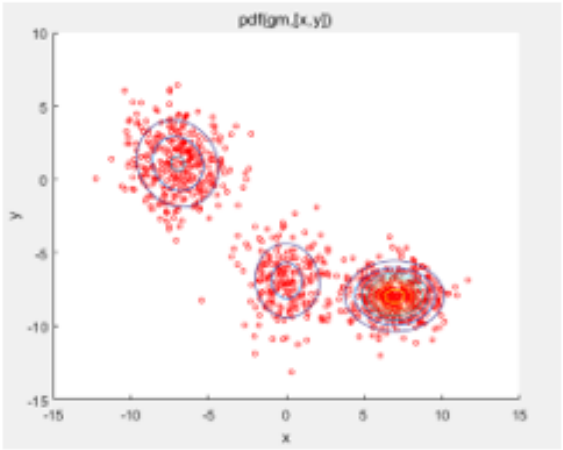
美赛实例

- For the reserved attributes and schools that are kept, we do data imputation to fill in missing data, based on k-means clustering. Then we normalize all the data to make them comparable in the following analysis.
- 即用k-Means来聚类，补充缺失的数据
- group similar schools, and then use the mean of schools in the group with complete data to fill in the missing data for others in the group.
- 先聚类，然后在group中，用完整的数据取平均等填补到空缺中。
- 作者也尝试了不同的K值，最后选择了一个效果比较好的k值来计算。

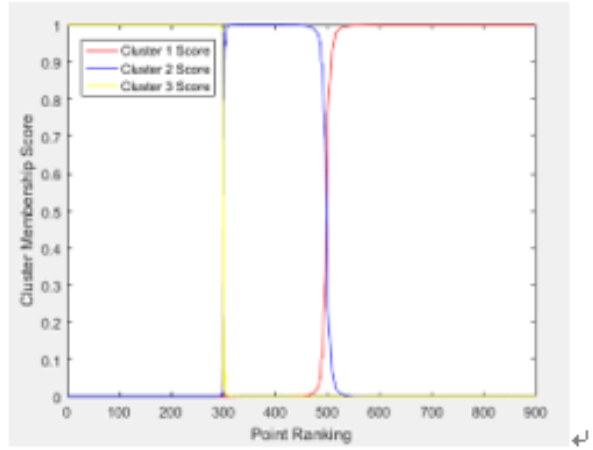
程序示例



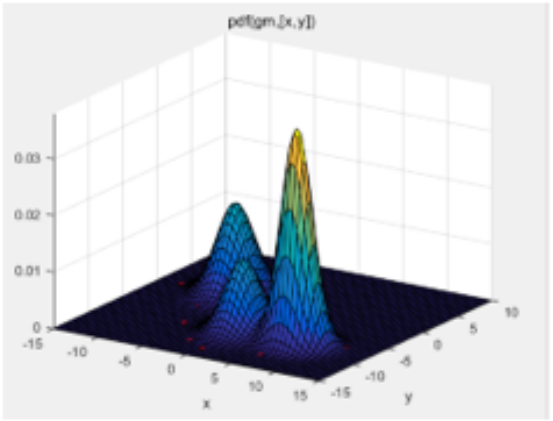
等高线



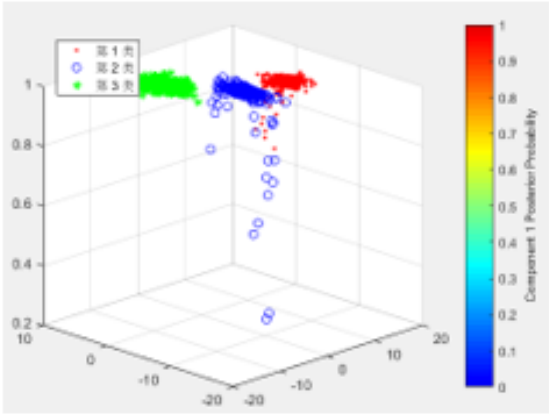
概率图



立体网格（等高线）



热能图



附录

- K-Means算法:
- https://coolshell.cn/articles/7779.html
- 层次算法：
- http://www.jianshu.com/p/785bb19386db
- SOM算法：
  http://blog.csdn.net/App_12062011/article/details/53462563?locationNum=7&fps=1
- EM算法：
- http://blog.csdn.net/zouxy09/article/details/8537620
- http://www.csuldw.com/2015/12/02/2015-12-02-EM-algorithms/
- https://chenrudan.github.io/chenrudan.github.io//blog/2015/12/02/emexample.html