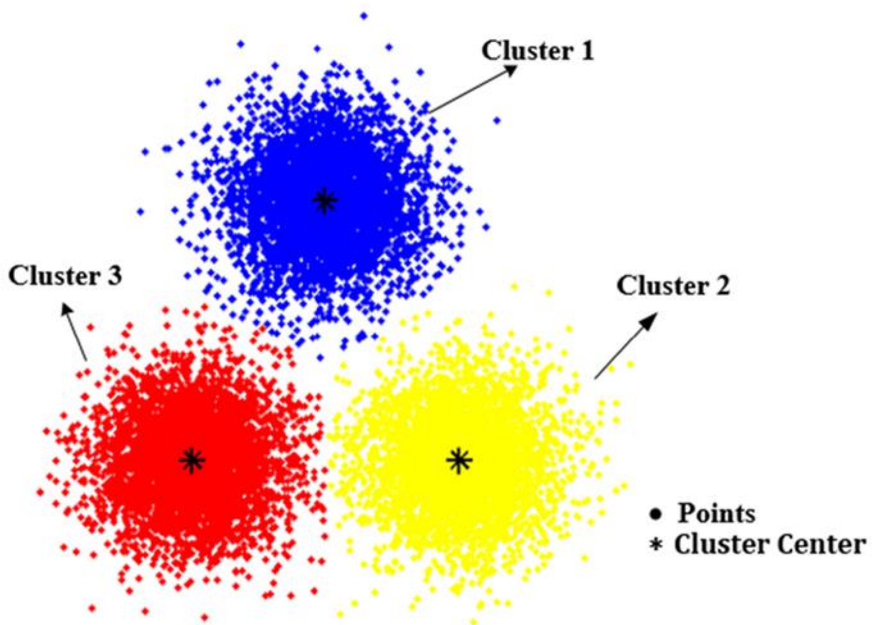
第八章：聚类方法
Ch8: Clustering



概述

物以类聚，人以群分

Birds of a feather flock together

概述

聚类是针对给定的样本，依据它们特征的相似度或距离，将其归并到若干个“类”或“簇”的数据分析问题。一个类是样本的一个子集。直观上，**相似的样本聚集在相同的类，**不相似的样本分散在不同的类。这里，样本之间的相似度或距离起着重要作用。

聚类的目的是通过得到的**类**或**簇**来发现数据的特点或对数据进行处理，在**数据挖掘、模式识别**等领域有着广泛的应用。**聚类属于无监督学习**，因为只是根据样本的相似度或距离将其进行归类，而类或簇事先并不知道。

聚类算法很多，本章介绍两种最常用的聚类算法：**层次聚类**（hierarchical clustering）和***k*-均值聚类**（*k*-means clustering）。

相似度或距离

假设有n个样本，每个样本由m个属性的特征向量组成，样本合集可以用矩阵X表示，

$$X = [x_{ij}]_{m\times n} = \begin{array}{cc} \text{第}j\text{个样品} \longrightarrow & \\ \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{bmatrix} & \Big\downarrow \text{第}i\text{个属性} \end{array}$$

聚类的核心概念是**相似度**（similarity）或**距离**（distance），有多种相似度或距离定义。因为相似度直接影响聚类的结果，所以**相似度选择是聚类的根本问题。**

闵可夫斯基距离

闵可夫斯基距离越大相似度越小，距离越小相似度越大

给定样本集合X, X是m维实数向量空间R^m中点的集合，其中，

$$x_i, x_j \in X,\ \ x_i = (x_{1i}, x_{2i}, \cdots, x_{mi})^{\mathrm{T}},\ \ x_j = (x_{1j}, x_{2j}, \cdots, x_{mj})^{\mathrm{T}}$$

样本x_i与样本x_j 的闵可夫斯基距离（Minkowski distance）定义为，

$$d_{ij} = \left(\sum_{k=1}^{m} |x_{ki} - x_{kj}|^p \right)^{\frac{1}{p}} \qquad p \geqslant 1$$

闵可夫斯基距离

当p=2时，称为欧氏距离（Euclidean distance)

$$d_{ij} = \left(\sum_{k=1}^{m} |x_{ki} - x_{kj}|^2 \right)^{\frac{1}{2}}$$

当p=1时，称为曼哈顿距离（Manhattan distance)

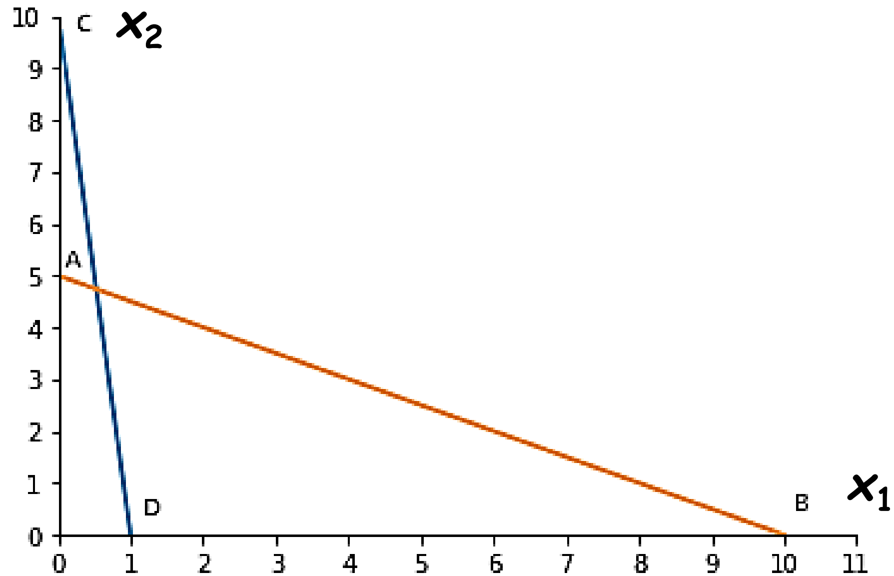
$$d_{ij} = \sum_{k=1}^{m} |x_{ki} - x_{kj}|$$

当p= ∞ 时，称为切比雪夫距离（Chebyshev distance)

$$d_{ij} = \max_{k} |x_{ki} - x_{kj}|$$

数据指标的单位对距离度量的影响

横轴x_1代表重量（以kg为单位），纵轴x_2代表长度（以cm为单位）。有四个点A，B，C，D，如图所示：



$$d(A,B)=\sqrt{5^2+10^2}=\sqrt{125}$$
$$d(C,D)=\sqrt{10^2+1^2}=\sqrt{101}$$

→ AB>CD

如果x_2用毫米(mm)做单位，x_1不变，此时A坐标(0，50)，C坐标为(0，100)，则，

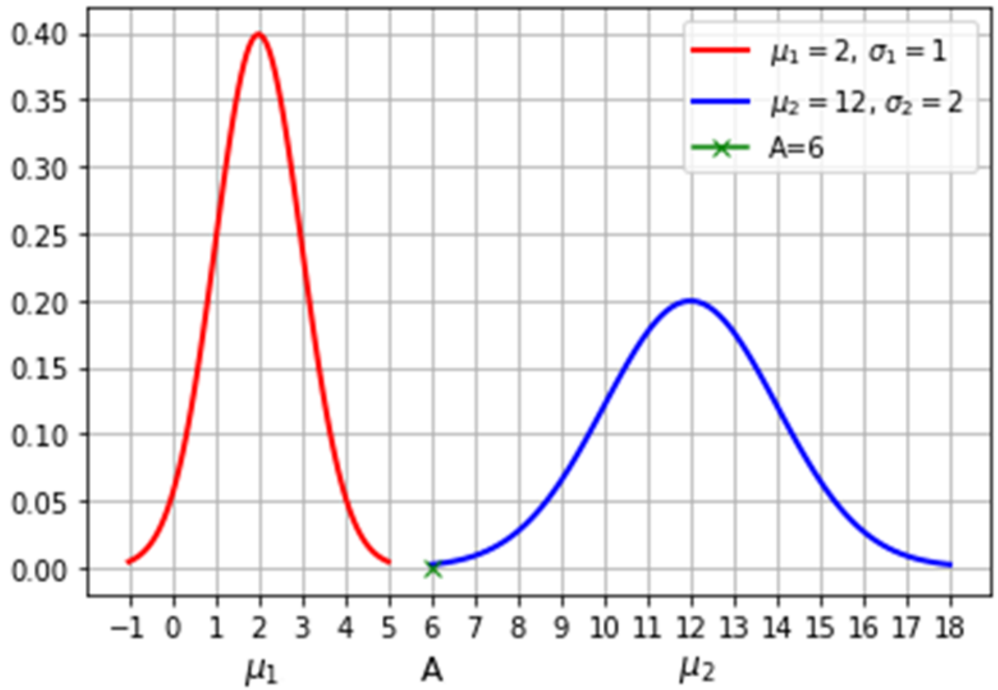
$$d(A,B)=\sqrt{50^2+10^2}=\sqrt{2600}$$
$$d(C,D)=\sqrt{100^2+1^2}=\sqrt{10001}$$

→ AB<CD

样本分布对距离度量的影响

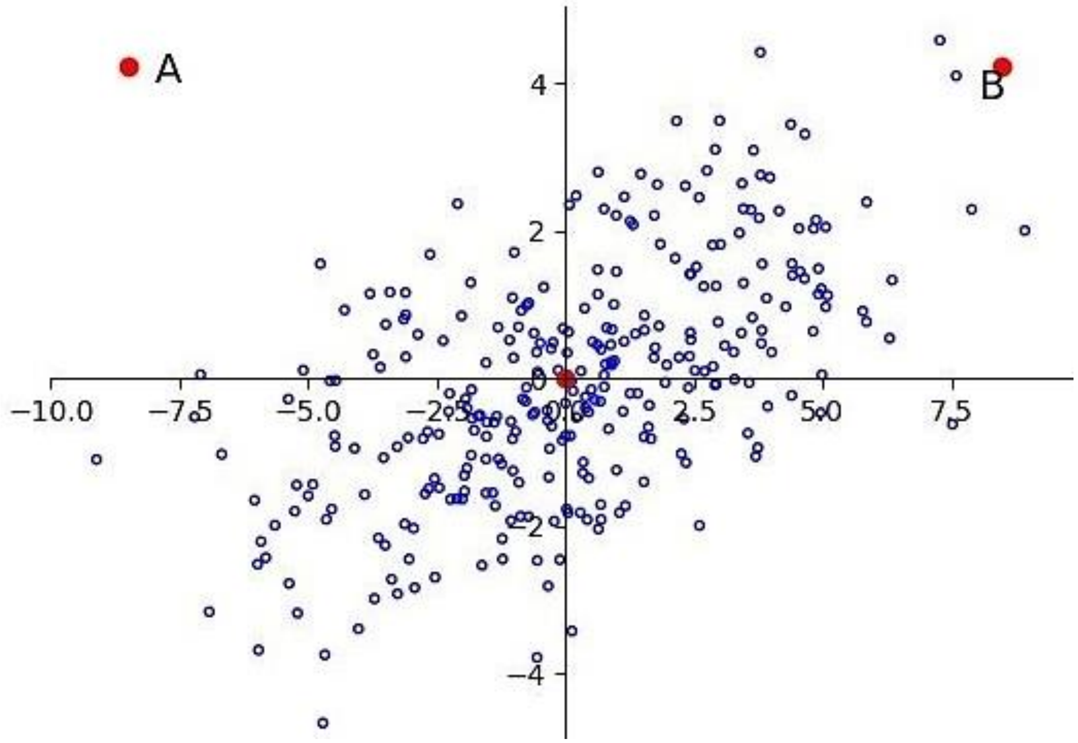
现实问题中，当坐标轴表示观测值时，它们往往带有大小不等的随机波动，即具有**不同的方差**。

设有两个正态分布总体 $G_1: N(\mu_1, \sigma_1^2)$和$G_2: N(\mu_2, \sigma_2^2)$。若有一个样本，其值在点*A*处，那么，*A*距离哪个总体近些呢?



维度间具有相关性对距离度量的影响

如果**维度间具有相关性**，度量距离会怎样变化呢？



要针对主成分分析中的主成分（PCA）来进行标准化

马哈拉诺比斯距离

马哈拉诺比斯距离（Mahalanobis distance），简称马氏距离，考虑**各个分量（特征）之间的相关性并与各个分量的尺度无关。**

方差：方差是标准差的平方，而标准差的意义是数据集中各个点到均值点距离的平均值。反应的是数据的离散程度。

$$\boxed{\sigma^2 = \frac{\sum (X-\mu)^2}{N}}$$

协方差：协方差是衡量多维数据集中变量之间相关性的统计量。比如，一个人的身高与体重的关系。如果两个变量之间的协方差为正值，则这两个变量正相关；若为负值，则负相关。

$$\begin{aligned} Cov(X,Y) &= E\left[(X-E[X])(Y-E[Y])\right] \\ &= E[XY] - 2E[Y]E[X] + E[X]E[Y] \\ &= E[XY] - E[X]E[Y] \end{aligned}$$

从直观上来看，协方差表示的是两个变量总体误差的期望。

马哈拉诺比斯距离

协方差矩阵：协方差矩阵用来衡量多变量之间的相关性。假设X是以n个随机变数（其中的每个随机变数是一个行向量）组成的列向量。

样本协方差矩阵S_G

$$X = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_n \end{bmatrix}$$

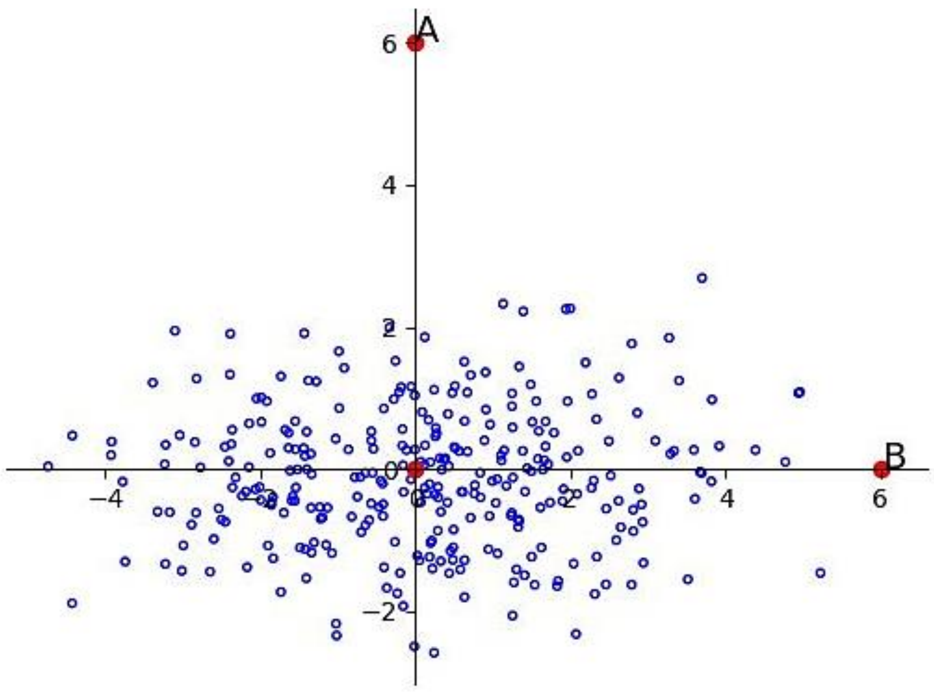
$$S_G = \frac{1}{n_G - 1} \sum_{i=1}^{n_G} (x_i - \bar{x}_G)(x_i - \bar{x}_G)^{\mathrm{T}}$$

给定一个样本集合X, $X: [x_{ij}]_{m\times n}$，其协方差矩阵记作S。样本x_i与样本x_j之间的马哈拉诺比斯距离d_{ij}定义为:

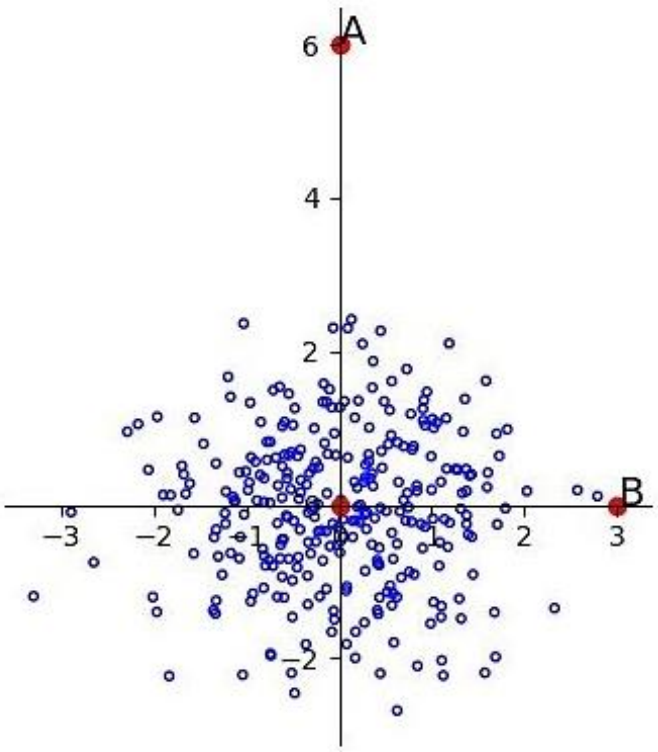
$$d_{ij} = \left[(x_i - x_j)^T S^{-1} (x_i - x_j)\right]^{\frac{1}{2}} \tag{14.6}$$

$$x_i = (x_{1i}, x_{2i}, \cdots, x_{mi})^{\mathrm{T}}, \quad x_j = (x_{1j}, x_{2j}, \cdots, x_{mj})^{\mathrm{T}} \tag{14.7}$$

马哈拉诺比斯距离变换



欧氏距离



马氏距离

相关系数

样本之间的相似度也可以用**相关系数**（correlation coefficient）来表示

相关系数的绝对值越接近于1，表示样本越相似；越接近于0，表示样本越不相似

样本x_i与样本x_j之间的相关系数定义为：

$$r_{ij} = \frac{\sum_{k=1}^{m}(x_{ki}-\bar{x}_i)(x_{kj}-\bar{x}_j)}{\left[\sum_{k=1}^{m}(x_{ki}-\bar{x}_i)^2 \sum_{k=1}^{m}(x_{kj}-\bar{x}_j)^2\right]^{\frac{1}{2}}} \quad (14.8)$$

$$\text{其中}, \bar{x}_i = \frac{1}{m}\sum_{k=1}^{m} x_{ki}, \quad \bar{x}_j = \frac{1}{m}\sum_{k=1}^{m} x_{kj}$$

夹角余弦

样本之间的相似度也可以用**夹角余弦**（cosine）来表示。

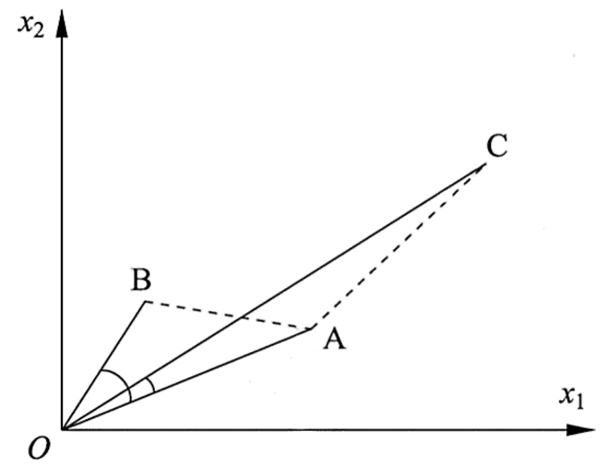
夹角余弦越接近于1，表示样本越相似；越接近于0，表示样本越不相似。

样本x_i与样本x_j之间的夹角余弦定义为:

$$s_{ij}=\frac{\sum_{k=1}^{m}x_{ki}x_{kj}}{\left[\sum_{k=1}^{m}x_{ki}^{2}\sum_{k=1}^{m}x_{kj}^{2}\right]^{\frac{1}{2}}}$$

相似度

- 用距离度量相似度时，**距离越小样本越相似**
- 用相关系数时，**相关系数越大样本越相似**
- 注意不同相似度度量得到的结果并不一定一致
- 从右图可以看出，如果从距离的角度看，*A*和*B*比*A*和*C*更相似
- 但从相关系数的角度看，*A*和*C*比*A*和*B*更相似



类或簇

通过聚类得到的类或簇，本质是样本的**子集**

如果一个聚类方法假定一个样本只能属于一个类，或**类的交集为空集**，那么该方法称为**硬聚类**（hard clustering）方法

如果一个样本可以属于多个类，或**类的交集不为空集**，那么该方法称为**软聚类**（soft clustering）方法

类或簇

- 用G表示类或簇（cluster），用x_i, x_j表示类中的样本，用n_G表示G中样本的个数，用d_{ij}表示样本x_i与样本x_j之间的距离

- 类或簇有多种定义，下面给出几个常见的定义

类或簇

定义 14.5 设 T 为给定的正数，若集合 G 中任意两个样本 x_i, x_j，有

$$d_{ij} \leqslant T$$

则称 G 为一个类或簇。

定义 14.6 设 T 为给定的正数，若对集合 G 的任意样本 x_i，一定存在 G 中的另一个样本 x_j，使得

$$d_{ij} \leqslant T$$

则称 G 为一个类或簇。

定义 14.7 设 T 为给定的正数，若对集合 G 中任意一个样本 x_i，G 中的另一个样本 x_j 满足

$$\frac{1}{n_G - 1} \sum_{x_j \in G} d_{ij} \leqslant T$$

其中 n_G 为 G 中样本的个数，则称 G 为一个类或簇。

定义 14.8 设 T 和 V 为给定的两个正数，如果集合 G 中任意两个样本 x_i, x_j 的距离 d_{ij} 满足

$$\frac{1}{n_G(n_G - 1)} \sum_{x_i \in G} \sum_{x_j \in G} d_{ij} \leqslant T$$

$$d_{ij} \leqslant V$$

则称 G 为一个类或簇。

类或簇

- **类的特征**可以通过不同角度来刻画，常用的特征有下面三种：

（1）类的均值 $\bar{x}_G$，又称为类的中心

$$\bar{x}_G = \frac{1}{n_G} \sum_{i=1}^{n_G} x_i \tag{14.10}$$

式中 n_G 是类 G 的样本个数。

（2）类的直径（diameter）D_G

类的直径 D_G 是类中任意两个样本之间的最大距离，即

$$D_G = \max_{x_i, x_j \in G} d_{ij} \tag{14.11}$$

类或簇

- 类的特征可以通过不同角度来刻画，常用的特征有下面三种：

（3）类的样本散布矩阵（scatter matrix）A_G 与样本协方差矩阵（covariance matrix）S_G

类的样本散布矩阵 A_G 为

$$A_G = \sum_{i=1}^{n_G} (x_i - \bar{x}_G)(x_i - \bar{x}_G)^{\mathrm{T}} \tag{14.12}$$

样本协方差矩阵 S_G 为

$$\begin{aligned} S_G &= \frac{1}{\mathrm{n_G} - 1} A_G \\ &= \frac{1}{\mathrm{n_G} - 1} \sum_{i=1}^{n_G} (x_i - \bar{x}_G)(x_i - \bar{x}_G)^{\mathrm{T}} \end{aligned} \tag{14.13}$$

类与类之间的距离

- 下面考虑类G_p与类G_q之间的距离$D(p, q)$，也称为**连接 (linkage)**。类与类之间的距离也有多种定义。

- 设类G_p包含 n_p个样本，G_q包含 n_q个样本，分别用 $\bar{x}_p$ 和 $\bar{x}_q$ 表示G_p和G_q的均值，即类的中心 。

类与类之间的距离

- **最短距离或单连接（single linkage）**

- 定义类G_p的样本与G_q的样本之间的**最短距离**为两类之间的距离

$$D_{pq} = \min\left\{d_{ij} \middle| x_i \in G_p, x_j \in G_q\right\}$$

- **最长距离或完全连接（complete linkage）**

- 定义类G_p的样本与G_q的样本之间的**最长距离**为两类之间的距离

$$D_{pq} = \max\left\{d_{ij} \middle| x_i \in G_p, x_j \in G_q\right\}$$

类与类之间的距离

- **中心距离**

- 定义类G_p与G_q的中心 $\bar{x}_p$ 与 $\bar{x}_q$ 之间的距离为两类之间的距离

$$D_{pq} = d_{\bar{x}_p \bar{x}_q}$$

先平均，后距离

- **平均距离**

- 定义类G_p与G_q任意两个样本之间距离的平均值为两类之间的距离

$$D_{pq} = \frac{1}{n_p n_q} \sum_{x_i \in G_p} \sum_{x_j \in G_q} d_{ij}$$

先距离，后平均

层次聚类

- 层次聚类假设类别之间存在**层次结构**，将样本聚到层次化的类中

- 层次聚类又有**聚合**（agglomerative）或自下而上（bottom-up）聚类、**分裂**（divisive）或自上而下（top-down）聚类两种方法

- 因为每个样本只属于一个类，所以**层次聚类属于硬聚类**

层次聚类

- **聚合（agglomerative）聚类**开始将每个样本各自分到一个类
- 之后将相距最近的两类合并，建立一个新的类
- 重复此操作直到满足停止条件
- 得到层次化的类别

- **分裂（divisive）聚类**开始将所有样本分到一个类
- 之后将已有类中相距最远的样本分到两个新的类
- 重复此操作直到满足停止条件
- 得到层次化的类别

聚合聚类的具体过程

- 对于给定的样本集合，开始将**每个样本分到一个类**。
- 然后按照一定规则，例如类间距离最小，将**最满足规则条件的两个类进行合并**。
- 如此反复进行，**每次减少一个类，直到满足停止条件**，如所有样本聚为一类。

聚合聚类

聚合聚类需要预先确定下面**三个要素**

- **距离或相似度**
  - 闵可夫斯基距离
  - 马哈拉诺比斯距离
  - 相关系数
  - 夹角余弦
- **合并规则**
  - 类间距离最小
  - 类间距离可以是最短距离、最长距离、中心距离、平均距离
- **停止条件**
  - 类的个数达到阈值（极端情况类的个数是1)
  - 类的直径超过阈值

聚合聚类算法

输入：n 个样本组成的样本集合及样本之间的距离；

输出：对样本集合的一个层次化聚类。

（1）计算 n 个样本两两之间的欧氏距离 $\{d_{ij}\}$，记作矩阵 $D=[d_{ij}]_{n\times n}$。

（2）构造 n 个类，每个类只包含一个样本。

（3）合并类间距离最小的两个类，其中最短距离为类间距离，构建一个新类。

（4）计算新类与当前各类的距离。若类的个数为 1，终止计算，否则回到步 (3)。■

可以看出聚合层次聚类算法的复杂度是 $O(n^3m)$，其中 m 是样本的维数，n 是样本个数。

例子

- 给定5个样本的集合，样本之间的欧氏距离由如下矩阵D表示

$$D = [d_{ij}]_{5\times 5} = \begin{bmatrix} 0 & 7 & 2 & 9 & 3 \\ 7 & 0 & 5 & 4 & 6 \\ 2 & 5 & 0 & 8 & 1 \\ 9 & 4 & 8 & 0 & 5 \\ 3 & 6 & 1 & 5 & 0 \end{bmatrix}$$

- 其中d_{ij}表示第i个样本与第j个样本之间的欧氏距离。
- D为对称矩阵。应用聚合层次聚类法对这5个样本进行聚类。

例子

(1)

- 首先用5个样本构建5个类,
- 这样，样本之间的距离也就变成类之间的距离，所以5个类之间的距离矩阵亦为D

$$G_i = \{x_i\},\ i = 1, 2, \cdots, 5,$$

(2)

- 由矩阵D可以看出, $D_{35} = D_{53} = 1$ 为最小，所以把G_3和G_5合并为一个新类，记作$G_6 = \{x_3, x_5\}$。

$$D = [d_{ij}]_{5\times5} = \begin{bmatrix} 0 & 7 & 2 & 9 & 3 \\ 7 & 0 & 5 & 4 & 6 \\ 2 & 5 & 0 & 8 & 1 \\ 9 & 4 & 8 & 0 & 5 \\ 3 & 6 & 1 & 5 & 0 \end{bmatrix}$$

(3)

- 计算G_6与G_1, G_2, G_4之间的最短距离，有

$$D_{61} = 2, \quad D_{62} = 5, \quad D_{64} = 5$$

- 又注意到其余两类之间的距离是

$$D_{12} = 7, \quad D_{14} = 9, \quad D_{24} = 4$$

- 显然，D_{61}=2最小，所以将G_1与G_6合并成一个新类，记作

$$G_7 = \{x_1, x_3, x_5\}$$

例子

(4) $G_7=\{x_1,x_3,x_5\}$

$$D=[d_{ij}]_{5\times 5}=\begin{bmatrix} 0 & 7 & 2 & 9 & 3 \\ 7 & 0 & 5 & 4 & 6 \\ 2 & 5 & 0 & 8 & 1 \\ 9 & 4 & 8 & 0 & 5 \\ 3 & 6 & 1 & 5 & 0 \end{bmatrix}$$

- 计算G_7与G_2, G_4之间的最短距离，

$$D_{72}=5, \quad D_{74}=5$$

- 又注意到

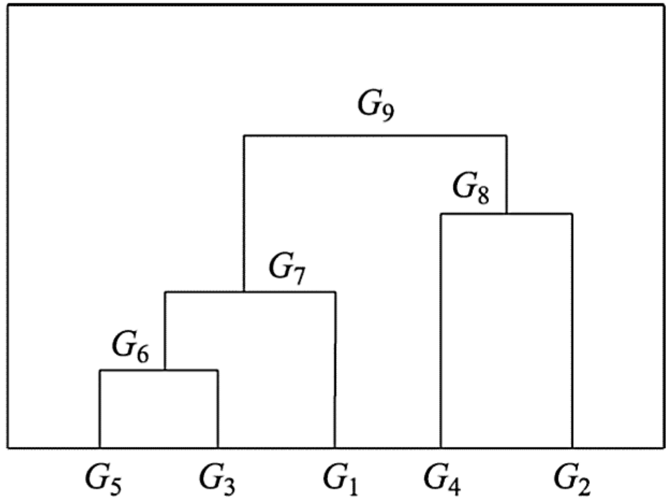
$$D_{24}=4$$

- 显然，其中D_{24}=4最小，所以将G_2与G_4合并成一个新类，记作

$$G_8=\{x_2,x_4\}$$

(5)

- 将G_7与G_8合并成一个新类，记作$G_9=\{x_1,x_2,x_3,x_4,x_5\}$
- 即将全部样本聚成1类，聚类终止



k均值（k-means）聚类

- 1967年MacQueen提出。
- k均值聚类是基于**样本集合划分**的聚类算法。
- k均值聚类将样本集合划分为***k*个子集**，构成k个类，将n个样本分到k个类中，**每个样本到其所属类的中心的距离最小**。
- 每个样本只能属于一个类，所以k均值聚类是**硬聚类**。

模型

- 给定n个样本的集合 $X=\{x_1,x_2,\cdots,x_n\}$
- 每个样本由一个特征向量表示，特征向量的维数是m。
- k均值聚类的目标是将n个样本分到k个不同的类或簇中，这里假设$k<n$。
- k个类$G_1, G_2, \ldots, G_k$形成对样本集合X的划分，其中，

$$G_i \cap G_j = \varnothing,\ \bigcup_{i=1}^{k} G_i = X$$

- 用C表示划分，一个划分对应着一个聚类结果。

模型

- 划分C是一个多对一的函数
- k均值聚类的模型是**一个从样本到类的函数。**
- 划分或者聚类可以用函数 $l=C(i)$ 表示，其中样本用一个整数 $i \in \{1,2,\cdots,n\}$ 表示，类用一个整数 $l \in \{1,2,\cdots,k\}$ 表示。

策略

- k均值聚类归结为样本集合X的划分，或者从**样本到类的函数的选择问题**。

- k均值聚类的策略是通过**损失函数的最小化**选取最优的划分或函数C^*。

- 首先，采用欧氏距离平方（squared Euclidean distance）作为样本之间的距离 $d(x_i, x_j)$。

$$
\begin{aligned}
d(x_i, x_j) &= \sum_{k=1}^{m} (x_{ki} - x_{kj})^2 \\
&= \|x_i - x_j\|^2
\end{aligned}
$$

策略

- 然后，定义**样本与其所属类的中心之间的距离的总和为损失函数**，即

$$W(C)=\sum_{l=1}^{k}\sum_{C(i)=l}\|x_i-\bar{x}_l\|^2$$

- $\bar{x}_l=(\bar{x}_{1l},\bar{x}_{2l},\cdots,\bar{x}_{ml})^{\mathrm{T}}$ 是第l个类的均值或中心，$n_l=\sum_{i=1}^{n}I(C(i)=l)$
- $I(C(i)=l)$ 是指示函数，取值1或0
- 函数$W(C)$ 也称为能量，表示相同类中的样本相似的程度。

策略

- k均值聚类就是求解最优化问题：

$$
\begin{aligned}
C^* &= \arg\min_C W(C) \\
&= \arg\min_C \sum_{l=1}^{k} \sum_{C(i)=l} \|x_i - \bar{x}_l\|^2
\end{aligned}
$$

- 相似的样本被聚到同类时，损失函数值最小，这个目标函数的最优化能达到聚类的目的。

- 但是，这是一个**组合优化**问题，n个样本分到k类，所有可能分法的数目是：

$$
S(n,k) = \frac{1}{k!} \sum_{l=1}^{k} (-1)^{k-l} \binom{k}{l} k^n
$$

- 事实上，k均值聚类的最优解求解问题是NP困难问题。现实中采用迭代的方法求解。

算法

- k均值聚类的算法是一个迭代的过程，每次迭代包括**两个**步骤：
- **首先选择k个类的中心**，将样本逐个指派到与其最近的中心的类中，得到一个聚类结果
- 然后**更新每个类的样本的均值**，作为类的**新的中心**
- **重复**以上步骤，直到收敛为止。

策略

- 首先，对于给定的中心值（$m_1, m_2, \dots, m_k$），求一个划分C，使得目标函数极小化：

$$\min_C \sum_{l=1}^{k} \sum_{C(i)=l} \|x_i - m_l\|^2$$

- 就是说在类中心确定的情况下，将每个样本分到一个类中，使**样本和其所属类的中心之间的距离总和最小**。

- 求解结果，将每个样本指派到与其最近的中心m_l的类G_l中。

策略

- 然后，对给定的划分C，再求各个类的中心（m_1，m_2,… ，m_k），使得目标函数极小化：

$$\min_{m_1,\cdots,m_k} \sum_{l=1}^{k} \sum_{C(i)=l} \|x_i - m_l\|^2$$

- 在划分确定的情况下，使样本和其所属类的中心之间的距离总和最小

- 求解结果，对于每个包含n_l个样本的类G_l，更新其均值m_l

$$m_l = \frac{1}{n_l} \sum_{C(i)=l} x_i, \quad l = 1, \cdots, k$$

- 重复以上两个步骤， 直到**划分不再改变**，得到聚类结果

k均值聚类算法

算法 14.2 (k均值聚类算法)

输 入： n个样本的集合X;

输 出： 样 本 集 合 的 聚 类 C。

(1) 初 始 化 。 令 $t=0$, 随 机 选 择 k 个 样 本 点 作 为 初 始 聚 类 中 心 $m^{(0)}=\left(m_1^{(0)},\cdots,m_l^{(0)},\cdots,m_k^{(0)}\right)$。

(2) 对样本进行聚类。对固定的类中心$m^{(t)}=\left(m_1^{(t)},\cdots,m_l^{(t)},\cdots,m_k^{(t)}\right)$,其中$m_l^{(t)}$ 为类G_l 的中心，计算每个样本到类中心的距离，将每个样本指派到与其最近的中心的类中，构成聚类结果$C^{(t)}$。

(3) 计算新的类中心。对聚类结果$C^{(t)}$,计算当前各个类中的样本的均值，作为新 的类中心$m^{(t+1)}=\left(m_1^{(t+1)},\cdots,m_l^{(t+1)},\cdots,m_k^{(t+1)}\right)$。

(4) 如果迭代收敛或符合停止条件，输出$C^*=C^{(t)}$。否则，令$t=t+1$,返回步 (2)。
k均值聚类算法的复杂度是$O(mnk)$,其中m是样本维数，n是样本个数，k是类别个数。

例子

- 给定含有5个样本的集合

$$X = \begin{bmatrix} 0 & 0 & 1 & 5 & 5 \\ 2 & 0 & 0 & 0 & 2 \end{bmatrix}$$

- 试用*k*均值聚类算法将样本聚到2个类中。

解： 按照算法14.2，

（1）选择两个样本点作为类的中心。假设选择 $m_1^{(0)} = x_1 = (0,2)^{\mathrm{T}}$，$m_2^{(0)} = x_2 = (0,0)^{\mathrm{T}}$。

（2）以 $m_1^{(0)}$，$m_2^{(0)}$ 为类 $G_1^{(0)}$，$G_2^{(0)}$ 的中心，计算 $x_3 = (1,0)^{\mathrm{T}}$，$x_4 = (5,0)^{\mathrm{T}}$，$x_5 = (5,2)^{\mathrm{T}}$ 与 $m_1^{(0)} = (0,2)^{\mathrm{T}}$，$m_2^{(0)} = (0,0)^{\mathrm{T}}$ 的欧氏距离平方。

对 $x_3 = (1,0)^{\mathrm{T}}$，$d(x_3, m_1^{(0)}) = 5$，$d(x_3, m_2^{(0)}) = 1$，将 x_3 分到类 $G_2^{(0)}$。

对 $x_4 = (5,0)^{\mathrm{T}}$，$d(x_4, m_1^{(0)}) = 29$，$d(x_4, m_2^{(0)}) = 25$，将 x_4 分到类 $G_2^{(0)}$。

对 $x_5 = (5,2)^{\mathrm{T}}$，$d(x_5, m_1^{(0)}) = 25$，$d(x_5, m_2^{(0)}) = 29$，将 x_5 分到类 $G_1^{(0)}$。

例子

（3）得到新的类 $G_1^{(1)}=\{x_1,x_5\}$，$G_2^{(1)}=\{x_2,x_3,x_4\}$，计算类的中心 $m_1^{(1)}$，$m_2^{(1)}$：

$$m_1^{(1)}=(2.5,2.0)^{\mathrm{T}},\quad m_2^{(1)}=(2,0)^{\mathrm{T}}$$

$$X=\begin{bmatrix}0&0&1&5&5\\2&0&0&0&2\end{bmatrix}$$

（4）重复步骤 (2) 和步骤 (3)。

具体如何操作呢？

将 x_1 分到类 $G_1^{(1)}$，将 x_2 分到类 $G_2^{(1)}$，x_3 分到类 $G_2^{(1)}$，x_4 分到类 $G_2^{(1)}$，x_5 分到类 $G_1^{(1)}$。

得到新的类 $G_1^{(2)}=\{x_1,x_5\}$，$G_2^{(2)}=\{x_2,x_3,x_4\}$。

由于得到的新的类没有改变，聚类停止。得到聚类结果：

$$G_1^*=\{x_1,x_5\},\quad G_2^*=\{x_2,x_3,x_4\}$$

算法特性

总体特点：

- **基于划分**的聚类方法
- 类别数k事先指定（**超参数！**）
- 以欧氏距离平方表示样本之间的距离
- 以样本和其所属类的中心之间的距离的总和为最优化的目标函数
- 得到的类别是**平坦的、非层次化的**
- **算法是迭代算法，不能保证得到全局最优**

算法特性

收敛性：

- k均值聚类属于启发式方法，不能保证收敛到全局最优

- 类中心在聚类的过程中会发生移动，但是往往不会移动太大，因为在每一步，样本被分到与其最近的中心的类中

算法特性

初始类的选择：

- 选择不同的初始中心，会得到不同的聚类结果

- 初始中心的选择，比如可以用层次聚类对样本进行聚类，得到k个类时停止，然后从每个类中选取一个与中心距离最近的点

算法特性

类别数*k*的选择：

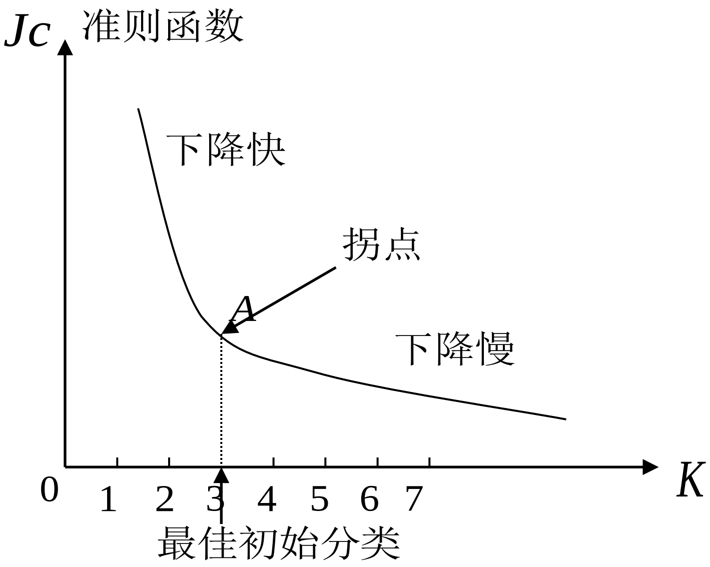
- *k*均值聚类中的类别数***k*值需要预先指定**，而在实际应用中最优的*k*值是不知道的
- 尝试用不同的*k*值聚类，检验得到聚类结果的质量，推测最优的*k*值
- 聚类结果的**质量可以用类的平均直径来衡量**
- 一般地，类别**数变小时，平均直径会增加**

算法特性

类别数k的选择-拐点法：

$$J_C = \sum_{j=1}^{c}\sum_{k=1}^{n_j}\left\| x_k - m_j \right\|^2$$

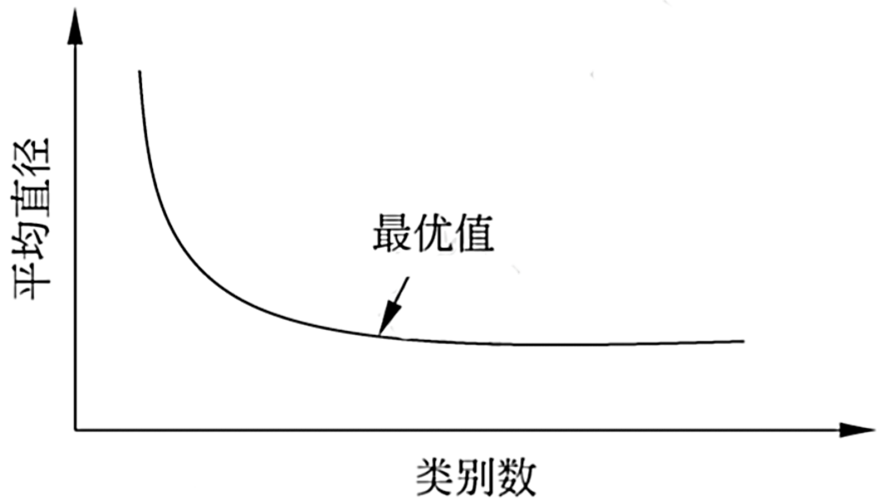
对于K未知时，可以令K逐渐增加，如：K=1,2….，使用K-means算法，误差平方和J_c随K的增加而单调减少。最初，由于K较小，类型的分裂会使J_c迅速减小，但当K增加到一定数值时，J_c的减小速度会减慢，即：随着初始分类k的增大，准则函数下降很快，经过拐点后，下降速度减慢。拐点处的K值就是最佳初始分类。



算法特性

类别数k的选择-临界平均直径

类别数变大超过某个值以后，平均直径会不变，而这个值（**临界平均直径**）正是**最优的*k*值**。实验时，可以采用二分查找，快速找到最优的*k*值。



K-Means算法的优缺点

优点：

（1）特征数很大，k均值比层次聚类的计算速度更快

（2）与层次聚类相比，k均值可以得到更紧密的簇，尤其是对于球状簇

（3）数据样本很大，效率比较高

（4）当结果簇是密集的，而簇与簇之间区别明显的时候，效果较好

K-Means算法的优缺点

缺点：

（1）没有指明初始化均值的方法，常用的方法是随机的选取k个样本作为均值

（2）产生的结果依赖于均值的初始值，经常发生得到次优划分的情况。解决方法是多次尝试不同的初始值

（3）可能发生距离簇中心m_j最近的样本集为空的情况，因此，m_j将得不到更新

（4）不适合发现非凸面形状的簇，并且对噪声和离群点数据是比较敏感的，因为少量的这类数据能够对均值产生极大的影响

K-Means算法的十大用例

1.文档分类器

根据标签、主题和文档内容将文档分为多个不同的类别。首先，需要对文档进行初始化处理，将每个文档都用矢量来表示，并使用术语频率来识别常用术语进行文档分类，这一步很有必要。然后对文档向量进行聚类，识别文档组中的相似性。

2.物品传输优化

使用K-means算法的组合找到无人机最佳发射位置和遗传算法来解决旅行商的行车路线问题，优化无人机物品传输过程。

3.识别犯罪地点

使用城市中特定地区的相关犯罪数据，分析犯罪类别、犯罪地点以及两者之间的关联，可以对城市或区域中容易犯罪的地区做高质量的勘察。

4.客户分类

聚类能过帮助营销人员改善他们的客户群（在其目标区域内工作），并根据客户的购买历史、兴趣或活动监控来对客户类别做进一步细分。

5.球队状态分析

分析球员的状态一直都是体育界的一个关键要素。随着竞争越来愈激烈，机器学习在这个领域也扮演着至关重要的角色。如果你想创建一个优秀的队伍并且喜欢根据球员状态来识别类似的球员，那么K-means算法是一个很好的选择。

K-Means算法的十大用例

6.保险欺诈检测

机器学习在欺诈检测中也扮演着一个至关重要的角色，在汽车、医疗保险和保险欺诈检测领域中广泛应用。利用以往欺诈性索赔的历史数据，根据它和欺诈性模式聚类的相似性来识别新的索赔。

7.乘车数据分析

比如滴滴乘车信息的数据集，为我们提供了大量关于交通、运输时间、高峰乘车地点等有价值的数据集。分析这些数据不仅对Uber大有好处，而且有助于我们对城市的交通模式进行深入的了解，来帮助我们做城市未来规划。

8.网络分析犯罪分子

网络分析是从个人和团体中收集数据来识别二者之间的重要关系的过程。网络分析源自于犯罪档案，该档案提供了调查部门的信息，以对犯罪现场的罪犯进行分类。

9.呼叫记录详细分析

通话详细记录是电信公司在对用户的通话、短信和网络活动信息的收集。将通话详细记录与客户个人资料结合在一起，这能够帮助电信公司对客户需求做更多的预测。

10.IT警报的自动化聚类

大型企业IT基础架构技术组件（如网络，存储或数据库）会生成大量的警报消息。由于警报消息可以指向具体的操作，因此必须对警报信息进行手动筛选，确保后续过程的优先级。

The end