“物以类聚，人以群分”，所谓的聚类，就是将样本划分为由类似的对象组成的多个类的过程。聚类后，我们可以更加准确的在每个类中单独使用统计模型进行估计、分析或预测；也可以探究不同类之间的相关性和主要差异。

聚类和上一讲分类的区别：分类是已知类别的，聚类未知。

1.K-means聚类算法

K-means聚类算法

K-means聚类的算法流程：

一、指定需要划分的簇[cù]的个数K值（类的个数）；

二、随机地选择K个数据对象作为初始的聚类中心（不一定要是我们的样本点）；

三、计算其余的各个数据对象到这K个初始聚类中心的距离，把数据对象划归到距离它最近的那个中心所处在的簇类中；

四、调整新类并且重新计算出新类的中心；

五、循环步骤三和四，看中心是否收敛（不变），如果收敛或达到迭代次数则停止循环；

六、结束。

算法流程图

K-means算法的评价

优点：

（1）算法简单、快速。

（2）对处理大数据集，该算法是相对高效率的。

缺点：

（1）要求用户必须事先给出要生成的簇的数目k。

（2）对初值敏感。

（3）对于孤立点数据敏感。

K-means++算法

k-means++算法选择初始聚类中心的基本原则是：**初始的聚类中心之间的相互距离要尽可能的远。**

算法描述如下：

（只对K-means算法“初始化K个聚类中心” 这一步进行了优化）

步骤一：随机选取一个样本作为第一个聚类中心；

步骤二：计算每个样本与当前已有聚类中心的最短距离（即与最近一个聚类中心的距离），这个值越大，表示被选取作为聚类中心的概率较大；最后，用轮盘法（依据概率大小来进行抽选）选出下一个聚类中心；

步骤三：重复步骤二，直到选出K个聚类中心。选出初始点后，就继续使用标准的K-means算法了。

K-means算法的一些讨论

（1）聚类的个数K值怎么定？

答：分几类主要取决于个人的经验与感觉，通常的做法是多尝试几个K值，看分成几类的结果更好解释，更符合分析目的等。

（2）数据的量纲不一致怎么办？

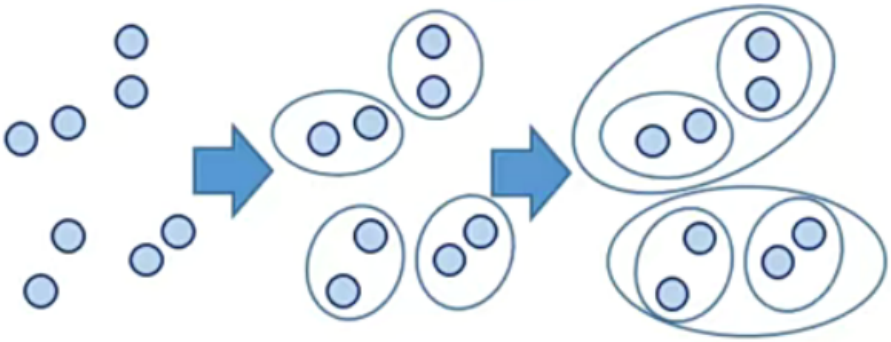
答：如果数据的量纲不一样，那么算距离时就没有意义。例如：如果X1单位是米，X2单位是吨，用距离公式计算就会出现“米的平方”加上“吨的平方”再开平方，最后算出的东西没有数学意义，这就有问题了。

$z_i = \frac{x_i - \overline{x}}{\sigma_x}$ (先减去均值再除以标准差)

系统层次聚类算法

系统（层次）聚类

系统聚类的合并算法通过计算两类数据点间的距离，对最为接近的两类数据点进行组合，并反复迭代这一过程，直到将所有数据点合成一类，并生成聚类谱系图。



聚类分析（物以类聚，人以群分）

LIAONING SHIHUA UNIVERSI

引例1 下表是30个学生的六门课的成绩。根据这30个人的成绩，对这30个学生进行分类。

序号	数学	物理	化学	语文	历史	英语
1	65	61	72	84	81	79
2	77	77	76	64	70	55
3	67	63	49	65	67	57
……	……	……	……	……	……	……
28	77	90	85	68	73	76
29	91	82	84	54	62	60
30	78	84	100	51	60	60

分类准则

距离近的样品聚为一类

数据的一般的格式

	指标 1 X_1	指标 2 X_2	…	指标 j X_j	…	指标 p X_p
样品 1	x_{11}	x_{12}	…	x_{1j}	…	x_{1p}
样品 2	x_{21}	x_{22}	…	x_{2j}	…	x_{2p}
⋮	⋮	⋮	⋱	⋮	…	⋮
样品 i	x_{i1}	x_{i2}	…	x_{ij}	…	x_{ip}
⋮	⋮	⋮	…	⋮	⋱	⋮
样品 n	x_{n1}	x_{n2}	…	x_{nj}	…	x_{np}

LIAONING SHIHUA UNIVERSITY

样品与样品之间的常用距离（样品*i*与样品*j*）

绝对值距离：$d(\vec{x}_i,\vec{x}_j)=\sum_{k=1}^{p}|x_{ik}-x_{jk}|$

欧氏距离：$d(\vec{x}_i,\vec{x}_j)=\sqrt{\sum_{k=1}^{p}(x_{ik}-x_{jk})^2}$

Minkowski距离：$d(\vec{x}_i,\vec{x}_j)=\left[\sum_{k=1}^{p}(x_{ik}-x_{jk})^q\right]^{\frac{1}{q}}$

Chebyshev距离：$d(\vec{x}_i,\vec{x}_j)=\max_{1\le k\le p}|x_{ik}-x_{jk}|$

马氏距离：$d(\vec{x}_i,\vec{x}_j)=(\vec{x}_i-\vec{x}_j)'\Sigma^{-1}(\vec{x}_i-\vec{x}_j)$

其中：$\vec{x}_i=(x_{i1},x_{i2},\cdots,x_{ip})'$　$\vec{x}_j=(x_{j1},x_{j2},\cdots,x_{jp})'$

Σ为样本的协方差矩阵

聚类分析需要注意的问题

1. 对于一个实际问题要根据分类的目的来选取指标，指标选取的不同分类结果一般也不同。

2. 样品间距离定义方式的不同，聚类结果一般也不同。

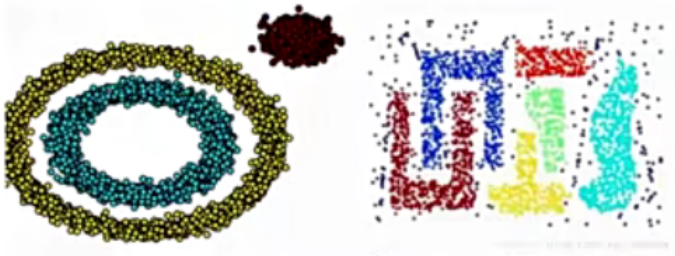
3. 聚类方法的不同，聚类结果一般也不同（尤其是样品特别多的时候）。最好能通过各种方法找出其中的共性。

4. 要注意指标的量纲，量纲差别太大会导致聚类结果不合理。

5. 聚类分析的结果可能不令人满意，因为我们所做的是一个数学的处理，对于结果我们要找到一个合理的解释。

基于密度的DBSCAN算法

DBSCAN(Density-based spatial clustering of applications with noise)是Martin Ester, Hans-PeterKriegel等人于1996年提出的一种基于密度的聚类方法，聚类前不需要预先指定聚类的个数，生成的簇的个数不定（和数据有关）。该算法利用基于密度的聚类的概念，即要求聚类空间中的一定区域内所包含对象（点或其他空间对象）的数目不小于某一给定阈值。该方法能在具有噪声的空间数据库中发现任意形状的簇，可将密度足够大的相邻区域连接，能有效处理异常数据。



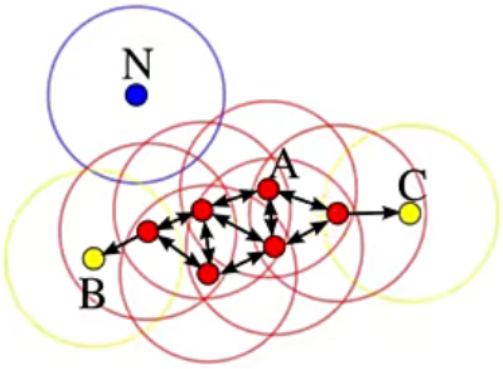
谁和我挨的近，我就是谁兄弟
兄弟的兄弟，也是我的兄弟

DBSCAN：具有噪声的基于密度的聚类方法

基本概念

DBSCAN算法将数据点分为三类：

- 核心点：在半径Eps内含有不少于MinPts数目的点
- 边界点：在半径Eps内点的数量小于MinPts，但是落在核心点的邻域内
- 噪音点：既不是核心点也不是边界点的点



在这幅图里，MinPts = 4，点 A 和其他红色点是核心点，因为它们的 ε-邻域（图中红色圆圈）里包含最少 4 个点（包括自己），由于它们之间相互相可达，它们形成了一个聚类。点 B 和点 C 不是核心点，但它们可由 A 经其他核心点可达，所以也和A属于同一个聚类。点 N 是局外点，它既不是核心点，又不由其他点可达。

优缺点

优点：

1. 基于密度定义，能处理任意形状和大小的簇；
2. 可在聚类的同时发现异常点；
3. 与K-means比较起来，不需要输入要划分的聚类个数。

缺点：

1. 对输入参数 ε 和Minpts敏感，确定参数困难；
2. 由于DBSCAN算法中，变量 ε 和Minpts是全局唯一的，当聚类的密度不均匀时，聚类距离相差很大时，聚类质量差；
3. 当数据量大时，计算密度单元的计算复杂度大。

我的建议：

只有两个指标，且你做出散点图后发现数据表现得很"DBSCAN"，这时候你再用DNSCAN进行聚类。

其他情况下，全部使用系统聚类吧。

K-means也可以用，不过用了的话你论文上可写的东西比较少。