聚类：将数据分组成多个类，同一个类对象之间具有较高的相似度，不同类之间的对象差别较大；

统计方法：一般称为聚类分析；机器学习：一般称为无监督学习

相似度尺度的获取：1直接从对象获取；2根据描述每个队形的测量或特征向量，间接地得到相似性尺度；

数据转换：通常包括平方根，倒数，对数等；特别是当我们使用正式的统计推理时，数据的分布是很重要的（比如关系到正态假设），则可以通过转换数据以便使得他们更加接近所需要的分布，如对正向倾斜数据（银行账户里的金额）取对数使其分布更加均衡（使其更好地逼近一个正态分布）

分类：层次聚类（凝聚、分裂）、划分式聚类、基于密度和网格的聚类算法

凝聚的层次聚类：该算法由树状结构的底部开始逐层向上进行聚合，假定样本集

{O1，O2，．．．，On)共有n个样本。

1、初始化：置每个样本Oi为一个类；

/\* 共形成n个类O1，O2，．．．，On\*/

2、寻找最近的两个类。

/\*从现有的所有类中找出距离最近的两个类Or, Ok\*/

3、合并Or, Ok并形成一个新类Ork；

/\*现有类的数量将减少1\*/

4、若所有的样本属于同一个类则终止本算法。

层次聚类的缺点：

* AGNES和DIANA算法比较简单，但一旦一组对象被合并或撤销，下一步的处理将在新生成的簇上进行。已做处理不能撤消，因此，如果合并或分裂选择不当，则可能导致低质量的簇。
* 假定在开始的时候有n个簇，在结束的时候有1个簇，因此在主循环中有n次迭代，另外算法必须计算所有对象两两之间的距离，因此这个算法的复杂度为 O(n2)，该算法对于n很大的情况是不适用的

两个簇之间的距离度量方法



划分式聚类算法（如K均值聚类）

在基于划分的聚类中，任务就是将数据划分成K个不相交的点集，使每个子集中的点尽可能同质。也就是给定n个数据点的集合D={X(1)，X(2), …… ,X(n)}，任务是找到K个聚类C={C1,C2,….Ck }，使每一个点x(i)被分配到唯一一个聚类Ck 中。

同质性（homogeneity）是这样实现的：选取适当的评分函数，并使每一点到它所属的聚类矩心（centroid）的距离最小化。

聚类内差异的一个简单尺度是看聚类内每个点到它所属聚类中心距离的平方和：



当的d(x, rk)被定义为欧式距离时，wc(C)被称为聚类内平方和。

聚类间差异定义为聚类中心间的距离：



聚类C的总体质量可被定义为wc(C)与bc(c)的单调组合。比如bc(c)/wc(c)。即目标函数

计算wc(C)至少需要次操作。而计算bc(c)需要次操作。因此，为一个聚类计算评分函数需要遍历

数据一次。

K均值算法：给定一个n个对象的数据集，构建数据的K个划分，每个划分表示一个类，满足：每个组至少包含一个对象；每个对象必须且只属于一个组。

K均值算法的优缺点：

K均值是一种典型的爬山式的搜索算法。这种算法简单，快速。对初值敏感。对不同的初值，可能会导致不同的聚类结果。另外，该算法是基于梯度下降的算法，由于目标函数局部极小点的存在，以及算法的贪心性，可能会陷入局部最优，而无法达到全局最优。也就是说K均值算法的搜索过程局限于全部可能划分空间的一个很小部分。因此有可能因为算法收敛到评分函数的局部而非全局最小值而错过一个更好的解。缓解该问题的方法是从不同的随机选择的初始点进行多次搜索。甚至可以更进一步的采用模拟退火策略来尽可能避免陷入评分函数的局部最小值。

K均值（K- means）算法优点：

能够对大型数据集进行高效分类，其计算复杂度为O（KmnI）

其中n为待分类的数据对象数量，m为特征属性数。在对大型数据

集合聚类时，该方法比层次聚类快很多。

K均值（K- means）算法缺点：

通常会在获得一个局部最优值时中止，仅适合对数值数据聚类，只适用于聚类结果为凸（即类簇位凸形）的数据集。或者说K-means方法这种贪婪算法的优点是简单而且保证至少得到评分函数的局部最大值（最小值）。也存在贪婪算法的普遍缺陷，即无法知道收敛到的聚类C与最佳的可能聚类相比的好坏程度。

主要问题：初始点随机选取的合理性；选择点数多少的合理性；聚类中心的计算方法；