什么是聚类分析

       将一群物理对象或者抽象对象的划分成相似的对象类的过程。其中类簇是数据对象的集合，在类簇中所有的对象都彼此相似，而类簇与类簇之间的对象是彼此相异。  
       聚类除了可以用于数据分割（data segmentation）,也可以用于离群点检测（outlier detection）,所谓的离群点指的是与“普通”点相对应的“异常”点，而这些“异常”点往往值的注意。  
       很多人在学习聚类之初，容易将聚类和分类搞混淆。其实聚类属于无监督学习范畴（unsupervised learning），也可称作观察式学习过程，与分类不同，聚类并不依赖已有既定的先验知识。举个例子，我们成年后，很清楚世界是由男人和女人组成的，所以我们在建厕所的时候，会把厕所分为男厕所和女厕所，这就是“分类”；而当我们刚生下来，我们并不知道什么是男人，什么是女人，通过后天对生活的观察，我们发现有一类人他们有胡子，而有一类人她们头发比较长（当然，我的这个举例已经显然不符合当今世界的发展了，你们明白就行），于是我们就把有胡子的人分为一类人，把长头发的分为另一类人，然后“研究”发现，原来有胡子的叫男人，有长头发的叫女人，这个过程就是“聚类”。  
        数据挖掘对聚类的典型要求如下：  
1）  可伸缩性：当聚类对象由几百上升到几百万，我们希望最后的聚类结果的准确度能一致。  
2）  处理不同类型属性的能力：有些聚类算法，其处理对象的属性的数据类型只能为数值类型，但是实际应用场景中，我们往往会遇到其他类型的数据，比如二元数据，分类数据等等。当然，在处理过程我们是可以将这些其他类型的数据预处理成数值型数据的，但是在聚类效率上或者聚类准确度上往往会有折损  
3）  发现任意形状的类簇：因为许多聚类算法是用距离（eg:欧几里得距离或者曼哈顿距离）来量化对象之间的相似度的，基于这种方式，我们往往只能发现相似尺寸和密度的球状类簇或者成为凸形类簇。但是，类簇的形状可能是任意的。  
4）  对聚类算法初始化参数的知识需求的最小化：很多算法在分析过程中需要用户提供一定的初始参数，比如期望的类簇个数，类簇初始质点的设定。聚类结果对这些参数是十分敏感的。这不仅加重了用户的负担，也非常影响聚类结果的准确性  
5）  处理噪声数据的能力：所谓的噪声数据，可以理解为影响聚类结果的干扰数据，这些噪声数据的存在会造成聚类结果的畸变，最终导致低质量的聚类。  
6）  增量聚类和对输入次序的不敏感：一些聚类算法不能将新加入的数据插入到已有的聚类结果；输入次序的敏感是指，对于给定的数据对象集合，以不同的次序提供输入对象时，最终产生的聚类结果的差异会比较大。  
7）  高维性：有些算法只适合处理2维或者3维的数据，而对高维数据的处理能力很弱，因为在高维空间中数据分布可能十分稀疏，而且高度倾斜。  
8）  基于约束的聚类：现实应用中可能需要在各种条件下进行聚类。因为同一个聚类算法，在不同的应用场景中所带来的聚类结果也是各异的，因此找到满足特定约束的具有良好聚类特性的数据分组是十分有挑战性的。  
9）  可解释性和可用性：我们希望得到的聚类结果都能用特定的语义、知识进行解释，和实际的应用场景相联系。