P：聚类听着就像把一堆数据分成几类，这里面有什么奥妙呢？

H：没错，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，其结果满足，类内部相似，类之间不相似。注意它和分类有一定区别，聚类的对象仅仅是一堆数据，这些数据本身是不带标签的，所以聚类可以算作一种无监督学习。

H：就好像我们现在所面临的情况，各种组成物体的数据开始混杂在一起，各种物体的界限不再分明，我们要做的就是将这些逐渐混乱的数据重新分成一个个簇，让物体不再混乱。

P：嗯~听你讲的好像也不难的呀，那我该怎么用聚类呢？总不能我把这些数据一个个用袋子封装好吧？

H：这就要涉及到我们的算法了，为了实现聚类的目的，我们可以使用三种算法：模型聚类算法K-means、密度聚类算法DBSCAN和层次聚类算法Agnes，它们各有各的特点，虽然都能完成聚类的目的，但是聚类出的结果其实各自有所不同。当然，实现聚类可以用的算法远远不止这三个，只不过这三个算法已经可以满足我们的需求，并且也足够你这个大学生领悟了（笑）。

P：（怎么感觉在骂我？？？）好吧，那我速速学习一下吧，来！

H：那么先来看Kmeans算法，这个算法是最常被使用的算法，它认为两个目标的距离越近，相似度越大。

P：有没有什么通俗例子？

H：有啊，把每个样本点比作一个村庄的村民，你是天主教堂的教皇，要选派几个牧师去

传授福音（人啊你在寻找什么？），必须把每个村民覆盖在内。但是牧师们并不知道这个村庄的村民分布情况，于是他们各自选择了任意地点驻扎下来并广告村民。聪明的村民当然会选择最近的牧师去登记，于是每个牧师手里都有了一份名单（簇）。他一想：欸，为了天父我要让我的信众们享受更近的距离，于是他搬到了名单上所有地址的中心地带。所有牧师这样做以后，有的村民发现：离自己最近的牧师换了，于是名单再次更新了，牧师为了信众考虑又搬了家，循环往复不知多少次，终于牧师发现自己可以不用搬家了，于是他们把名单交给了你，对这个村庄的聚类也就此完成了。

P：噢噢噢噢！amen！这样就可以把距离近的数据聚成一类了！！看来我只要设置牧师数量（簇的数量）和迭代次数就可以完成聚类了！那密度聚类DBSCAN算法呢？

H：顾名思义，这是根据密度进行聚类的算法，还是以二位样本举例，坐标系中有很多的点，每个点以自己为圆心，r为半径作圆，形成一个“朋友圈”。

统计一下在这个圆内的点的数量，计它为“好朋友数“。

由你输入一个数n，如果一个点的好朋友数大于等于n，就说这个点是一个”交际花“。

我们知道，交际花总是会相互吸引的，那么一个交际花的好朋友里由另一个交际花也是很正常的，因此我们规定，如果一个交际花A的朋友圈里有令一个交际花B，那我就让他们的朋友圈合并，形成一个簇。

依此规则，这个朋友圈不断扩大，最后直到所有能被包含的交际花都被包含进去了，这个簇就算完事了。最后我们可以得到一个由密度主导的聚类结果，这个结果中，如果一些点的朋友圈太小，就会被淘汰出去，可见它和Kmeans有大不同的。

P：不合群就会被淘汰？？太哈人了！！那我设置参数的时候，可要把n调的小一点了！

H：你确实可以这样做，但是既然我们要聚类，就需要舍弃一些离群太远的点，它们对结果造成的误差可是很大的！我们也因此称它们为噪声。

P：好吧，那层次聚类Agnes呢？

H：这个你就非常熟悉了！数据结构学过吧？二叉树里的哈夫曼树学过吧？对于这么多个点，我先把它们相互之间的距离全部算出来，我每次都取两个最近的点，把他们变成一个簇。对于这个簇，我们仍然把它看成一个点，就像哈夫曼树的构造那样子。于是我们重复这个过程，直到簇的数量聚成一个。至于簇怎么看成一个点来计算距离，我这里给你三种方法：

(1).最小值：就是取两个类中距离最近的两个样本的距离作为这两个集合的距离，

(2).最大值：取两个集合中距离最远的两个点的距离作为两个集合的距离

(3).平均值：把两个集合中的点两两的距离全部放在一起求一个平均值，相对也能得到合适一点的结果。

你看，这样我就介绍完了这些聚类方法了~赶紧干活吧！！

P：我说停停，我是差不多知道它们的工作原理了啦，但是它们在各种情况下孰优孰劣我还没实践过啊！至少，把它们的优缺点也和我一叙呗。

H：呃呃，高级研究员在大学的时候这么差劲啊，好，你听着吧：对于Kmeans来说，如果有一个村民他住在山沟沟里，牧师也要为他一个人搬到山沟沟里，所以Kmeans算法不适合太离散的分类、样本类别不平衡的分类、非凸形状的分类。而DBSCAN因为可以不管离群的点，它的抗噪音性自然强了太多了，而且它可以聚出很有形状的类来，不像Kmeans，总是只给出一片片的聚类结果。不过DBSCAN也有限制的，如果样本集的密度不均匀、聚类间距差相差很大时，聚类质量较差，这时用DBSCAN聚类一般不适合。最后是层次聚类Agnes，它的优点很明显，前两种算法都需要给出很多参数，但我Agnes啥都不用给，也能聚地很有形状，又可以发现簇之间的层次关系，是很省事的方法。不过呢，你省事了，我就苦了，这从头到尾的，得算多少个距离啊，更别提这簇与簇之间的距离，算的我都发烧了！

P：你说得对，呃，我还是想要一些例子，要不你给我几张图，我来看看用三种聚类算法分别聚类以后它们的实际效果？

H：好吧，你操作就是了，快点领悟吧！

给出几个图，每幅图三个按键，代表三种算法，点击就出聚类效果图。

如果能提供自己画点上去的界面就无敌了。