

DBSCAN

Density-Based Spatial Clustering
of Applications with Noise

基于密度带有噪声点的聚类方法

目录

CONTENTS

01

概念

02

具体算法

03

聚类过程

04

优缺点



概念

DBSCAN(Density-Based Spatial Clustering of Applications with Noise)是一个比较有代表性的基于密度的聚类算法。与划分和层次聚类方法不同，将簇定义为密度相连的点的最大集合，能够把具有足够高密度的区域划分为簇，并可在噪声的空间数据库中发现任意形状的聚类。



DBSCAN中的基本概念

核心对象:如果给定对象 r 邻域内的样本点数大于等于 MinPts ，则称该对象为核心对象;

r 邻域:设定的半径 r ;

直接密度可达:对于样本集合 D ，如果样本点 q 在 p 的 r 邻域内，并且 p 为核心对象，那么称对象 q 从对象 p 直接密度可达。

密度可达:对于样本集合 D ，给定一串样本点 p_1, p_2, \dots, p_n ， $p = p_1, q = p_n$ ，假如对象 p_i 从 p_{i-1} 直接密度可达，那么对象 q 从对象 p 密度可达，这实际上是直接密度可达的“传播”。

DBSCAN中的基本概念

密度相连:存在样本集合D中的一点 o ，如果对象 o 到对象 p 和对象 q 都是密度可达的，那么 p 和 q 密度相联。

边界点:属于某一个类的非核心点，不能发展下线了。

噪声点:不属于任何一个类簇的点，从任何一个核心点出发都是密度不可达到的。



DBSCAN算法描述:

输入: 包含 n 个对象的数据库, 半径 e , 最少数目MinPts;

输出: 所有生成的簇, 达到密度要求。

(1)Repeat

(2)从数据库中抽出一个未处理的点;

(3)IF抽出的点是核心点 THEN 找出所有从该点密度可达的对象, 形成一个簇;

(4)ELSE 抽出的点是边缘点(非核心对象), 跳出本次循环, 寻找下一个点;

(5)UNTIL 所有的点都被处理。

DBSCAN对用户定义参数很敏感, 细微的不同都可能导致差别很大的结果, 而参数的选择无规律可循, 只能靠经验确定。

DBSCAN具体算法:

具体算法描述如下:

- (1)检测数据库中尚未检查过的对象 p ，如果 p 未被处理(归为某个簇或者标记为噪声)，则检查其邻域，若包含的对象数不小于 minPts ，建立新簇 C ，将其中的所有点加入候选集 N ;
- (2)对候选集 N 中所有尚未被处理的对象 q ，检查其邻域，若至少包含 minPts 个对象，则将这些对象加入 N ;如果 q 未归入任何一个簇，则将 q 加入 C ;
- (3)重复步骤2)，继续检查 N 中未处理的对象，当前候选集 N 为空;
- (4)重复步骤1)~3)，直到所有对象都归入了某个簇或标记为噪声。



算法过程：

1. 标记所有对象为unvisited;
2. Do
3. 随机选择一个unvisited对象p;
4. 标记p为visited ;
5. If p的r领域至少有minpts个对象 ;
6. 创建一个新簇c , 并把添加到c;
7. 令n为p的r领域内的对象集合 ;
8. For n 中每个点p;
9. If p是unvisited ;
10. 标记p为visited.
11. If p的r-领域至少有MinPts个对象 , 把这些对象添加到N ;
12. 如果p还不是任何簇的成员 , 把P添加到C;
13. End for;
14. 输出C;
15. Else标记p为噪声 ;
16. Until没有标记为unvisited的对象。



DBSCAN优点



- 与K-means方法相比，DBSCAN不需要事先知道要形成的簇类的数量。



- 与K-means方法相比，DBSCAN可以发现任意形状的簇类



- DBSCAN能够识别出噪声点。



- DBSCAN对于数据库中样本的顺序不敏感，即Pattern的输入顺序对结果的影响不大。但是，对于处于簇类之间边界样本，可能会根据哪个簇类优先被探测到而其归属有所摆动。



DBSCAN缺点



DBSCAN不能很好反映高维数据。



DBSCAN不能很好反映数据集以变化的密度。



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。