链滴

DBSCAN 聚类

作者：Lonery

原文链接：https://ld246.com/article/1615276421278

来源网站：链滴

DBSCAN

Density-Based Spatial Clustering of Applications with Noise

1. 一种基于密度，对噪声鲁棒的空间聚类算法。

2. DBSCAN算法可以找到样本点的全部密集区域，并把这些密集区域当做一 个一个的聚类簇

3. 通常情形下，密度聚类算法从样本密度的角度来考察样本之间的可连接性， 并基于可连接样本不扩展聚类簇以获得最终的聚类结果。

4. DBSCAN算法基于一组“领域”参数(ε,MinPts)来刻画样本分布的紧密程度。ε：领域半径；MinPt：密度阈值

关于DBSCAN的通俗描述：

<p1 style="text-indent:2em">

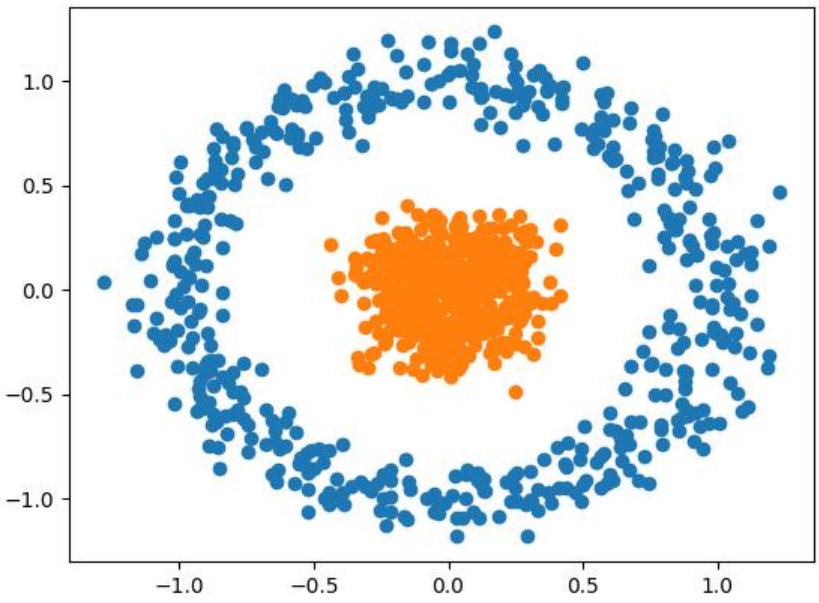
对于DBSCAN算法最为重要的就是核心点。什么是核心点呢？简单来说就是当在以一个点为圆心半径ε的区域内存在不少于MinPts个其他点时，此点即为核心点， 暂称核心点为“大哥”，核心点的邻域为“小弟”，当大哥的小弟作为大哥时，又会存在一批小弟，当第二批小弟又作为大哥，又会找出第批小弟，如此循环直至找不到小弟， 那么这些由第一代大哥衍生出来的小弟也好大哥也好与它们的邻构成一个簇，也就是一类。后来的点按照以上的操作往复循环就能找到一个一个的簇，直到所有的点找过。

</p1>

```
import numpy as np
import matplotlib.pyplot as plt
from sklearn import datasets #加载数据集
X=datasets.make_circles(n_samples=1000, factor=0.2, noise=0.1)[0] #生成环形数据
def distance(X):      #计算个点之间的距离(矩阵，同pdist2)
    n=len(X)
    D=np.zeros((n,n))
    for i in range(n):
        for j in range(0,i):
            D[i][j]=np.linalg.norm(X[i]-X[j],ord=2)
            D[j][i]=D[i][j]
    return D
def dbscan(X,epsilon,MinPts):
    C=0  #以点在源数据中的位置标记类别
    n=len(X)
    idx=[0 for i in range(n)]  #类别信息存储
    visited=[0 for i in range(n)] #巡查判断
    D=distance(X)  #距离矩阵
    def RegionQuery(i):       #列出一个点周围ε内存在的点的下标
        Neighbors=np.where(D[i,:]<=epsilon)
        return list(Neighbors)[0].tolist()  #以列表形式返回
    def ExpandCluster(i,Neighbors,C):  #将邻域点能够成为核心点的点归类
        idx[i]=C  #将点赋给类别
        k=1
        while k<=len(Neighbors):
            j=Neighbors[k]
```

```
            if visited[j]==0:
                visited[j]=1
                Neighbors2=RegionQuery(j)
                if len(Neighbors2)>=MinPts:
                    Neighbors.extend(Neighbors2)
            if idx[j]==0:
                idx[j]=C  #将没有分类的点分类
            k=k+1
            if k>len(Neighbors)-1:
                break
    for i in range(n):  #寻找核心点
        if visited[i]==0:
            visited[i]=1
            Neighbors=RegionQuery(i)
            if len(Neighbors)>=MinPts:
                C=C+1   #类别标志加一
                ExpandCluster(i,Neighbors,C)

    return idx
def plot(a,X):  #绘画板块
    m=max(a)
    for j in range(m):
        index=[i for i,v in enumerate(a) if v==j+1]
        x=[]
        y=[]
        for k in index:
            x.append(X[k][0])
            y.append(X[k][1])
        plt.scatter(x,y)
    x=[]
    y=[]
    for i in range(len(a)): #噪点绘制
        if a[i]==0:
            x.append(X[i][0])
            y.append(X[i][1])
    plt.scatter(x,y,marker='x')
    plt.show()
if __name__=="__main__":  #设置ε为0.2，MinPts为3 进行测
    a=dbscan(X,0.2,3)
    plot(a,X)
```



原文链接：DBSCAN 聚类