第七章 聚类分析

7.5 基于密度的聚类法

1. DBSCAN简介

DBSCAN(Density-Based Spatial Clustering of Applications with noise，具有噪声的基于密度的聚类方法）是典型的基于密度的聚类算法。与K均值算法相比，DBSCAN算法在执行之初不需要预先制定聚类组的个数。当然，最终的聚类组个数在结果出来之前也就不得而知了。

DBSCAN算法的优点：聚类速度快、能够有效处理噪声点，以及聚类组的形状没有偏倚。

缺点：当数据量增大时，要求较大的内存支持，同时当空间聚类的密度不均匀、聚类间距相差很大时聚类质量较差。

第七章 聚类分析

7.5 基于密度的聚类法

2. 相关定义

既然是基于密度的聚类，下面给出与密度相关的定义。

ε-邻域：给定对象O，半径ε内的区域称为该对象O的ε -邻域。

核心对象：如果给定对象O的ε- 邻域内的样本点数大于或等于MinPts，则称该对象O为核心对象。

MinPts 为人为预先指定的最小点数，阈值参数。

第七章 聚类分析

7.5 基于密度的聚类法

2. 相关定义

直接密度可达：给定一个对象集合D，如果p在q的ε -邻域内，且q是一个核心对象，则称**对象p**从**对象q**出发是直接密度可达的。

密度可达：对于样本集合D，如果存在一个对象链p1,p2,...,pn，使得p1=q，pn=p，并且pi属于D（i=1,2,...,n），p(i+1)是pi关于 ε 和 MinPts 直接密度可达的，则称**对象p**从**对象q**出发是密度可达的。

第七章 聚类分析

7.5 基于密度的聚类法

2. 相关定义

密度相连：如果存在对象q属于D，使对象p1和p2都是从q关于 ε 和MinPts密度可达的，那么对象p1、p2是关于ε 和MinPts 密度相连的。

DBSCAN聚类：由**密度可达**关系导出的**最大密度相连**的**样本集合**，即为最终聚类簇（一簇 即为 一类）。

第七章 聚类分析

7.5 基于密度的聚类法

3. DBSCAN基本原理

这个DBSCAN的簇（类）里面可以有一个或多个**核心对象**。

如果只有**一个核心对象**，则簇里其它的非核心对象都在这个核心对象的 ε 里；

如果有**多个核心对象**，则簇里任意一个核心对象的 ε 邻域中一定有一个其它的核心对象，否则这两个核心对象无法密度可达。这些核心对象的 ε 邻域里**所有样本的集合**组成一个**DBSCAN聚类簇**。

7.5 基于密度的聚类法

3. DBSCAN基本原理

那么怎么才能找到这样的簇样本集合呢？DBSCAN使用的方法流程很简单：

(1) 任意选择一个没有类别的核心对象作为种子，

(2) 然后找到所有这个核心对象能够密度可达的样本集合，即为一个聚类簇。

(3) 接着继续选择另一个没有类别的核心对象去寻找密度可达的样本集合，这样就得到另一个聚类簇。

(4) 一直运行到所有核心对象都有类别为止。

第七章 聚类分析

7.5 基于密度的聚类法

3. DBSCAN基本原理

此外，对于DBSCAN算法有3个问题需要注意：

第一个是一些异常样本点或者说少量游离于簇外的样本点。这些点不在任何一个核心对象的周围，在DBSCAN中，一般将这些样本点标记为噪音点。

第二个是距离的度量问题，即如何计算某样本到核心对象样本的距离。

第七章 聚类分析

7.5 基于密度的聚类法

3. DBSCAN基本原理

第三个问题比较特殊，某些样本到两个核心对象的距离可能都小于ε ，但是这两个核心对象由于不是密度可达，又不属于同一个聚类簇，那么如何界定这个样本的类别？一般来说，DBSCAN采用先来后到，先进行聚类的类别簇会标记这个样本为它的类别。也就是说，DBSCAN算法不是完全稳定的算法。

7.5 基于密度的聚类法

4. DBSCAN的Python实现

在Python中，sklearn库提供了DBSCAN函数

DBSCAN使用方法如下：

```
from sklearn.cluster import DBSCAN

model = DBSCAN(eps=1.5,min_samples=4)          #输入参数建立模型

model.fit(Data)                    #将数据集提供给模型进行聚类
```

7.5 基于密度的聚类法

4. DBSCAN的Python实现

例 7.4　Iris 数据集由 Fisher 于 1936 收集整理。Iris 也称鸢尾花卉数据集，是一类多重变量分析的数据集。数据集包含 150 个数据集，分为 3 类，每类 50 个数据，每个数据包含 4 个属性，数据格式如表所示。可通过花萼长度，花萼宽度，花瓣长度，花瓣宽度 4 个属性预测鸢尾花卉属于（Setosa，Versicolour，Virginica）三个种类中的哪一类。

7.5 基于密度的聚类法

4. DBSCAN的Python实现

```
#程序文件Pex79.py
import numpy as np; import pandas as pd
from sklearn.cluster import DBSCAN
import matplotlib.pyplot as plt
a=pd.read_csv("iris.csv")
b=a.iloc[:,:-1]
md=DBSCAN(eps=1.5,min_samples=4);
md.fit(b)  #构建模型并求解模型
labels=md.labels_ ;
b['cluster']=labels  #数据框b添加一个列变量cluster
c=b.cluster.value_counts()  #各类频数统计
```

7.5 基于密度的聚类法

4. DBSCAN的Python实现

```
plt.rc('font',family='SimHei'); plt.rc('font',size=16)
str1=['^r','.k','*b']; plt.subplot(121)
for i in range(3):
    plt.plot(b['Petal_Length'][labels==i],b['Petal_Width']
             [labels==i], str1[i],markersize=3,label=str(i))
    plt.legend(); plt.xlabel("(a)KMeans聚类结果")
plt.subplot(122); str2=['setosa','versicolour','virginica']
ind=np.hstack([np.zeros(50),np.ones(50),2*np.ones(50)])
for i in range(3):
    plt.plot(b['Petal_Length'][ind==i],b['Petal_Width'][ind==i],
             str1[i],markersize=3,label=str2[i])
    plt.legend(loc='lower right'); plt.xlabel("(b)原数据的类别")
plt.show()
```