

## 实验 5 密度聚类

### 一、实验目的

1. 熟悉 Python 的基础使用
2. 掌握 `sklearn.datasets` 数据生成
3. 理解和掌握 DBSCAN 聚类的基本原理和实现过程
4. 体会 DBSCAN 聚类时 `eps`, `min_Pts` 参数对结果的影响

### 二、实验内容

1. 了解 DBSCAN 聚类原理及实现
2. 自制数据集
3. 用 DBSCAN 聚类解决实际问题
4. 体会 k-均值聚类和 DBSCAN 聚类的区别

### 三、实验原理：

DBSCAN 是一种基于密度的聚类算法，这类密度聚类算法一般假定类别可以通过样本分布的紧密程度决定。同一类别的样本，他们之间的紧密相连的，也就是说，在该类别任意样本周围不远处一定有同类别的样本存在。

通过将紧密相连的样本划为一类，这样就得到了一个聚类类别。通过将所有各组紧密相连的样本划为各个不同的类别，就得到了最终的所有聚类类别结果。

具体分析见课本。

### 任务 1: 运行 DBSCAN 程序

#### 任务要求:

(1) 打开原文链接，看懂并运行程序，运行结果以“班级姓名”为名

原文链接：<https://blog.csdn.net/xyisv/article/details/88918448>

(2) 程序中的数据是由下面两条语句得到的。

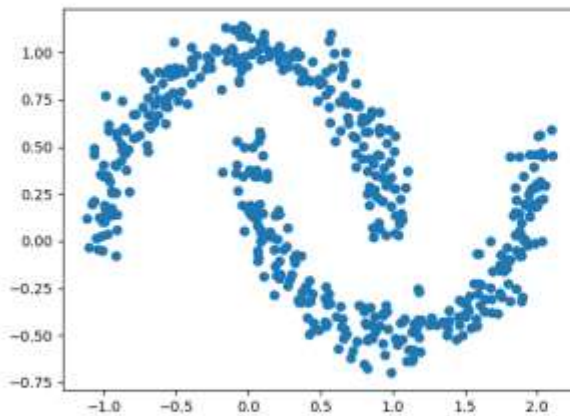
```
X1, y1 = datasets.make_circles(n_samples=2000, factor=.6, noise=.02)
X2, y2 = datasets.make_blobs(n_samples=400, n_features=2, centers=[[1.2, 1.2]],
cluster_std=[[.1]], random_state=9)
```

- ① 利用这两条语句，编写一个数据集可视化的程序；
- ② 分析两条语句的作用；
- ③ 改变语句中的参数，分析参数作用；
- ④ 保存运行结果（多图，对比展示，用图名标注参数数值）

## 任务 2: `datasets.make_moons` 函数

### 任务要求:

(1) 查阅资料, 利用 `datasets.make_moons` 函数, 生成类似如下形式的数据集

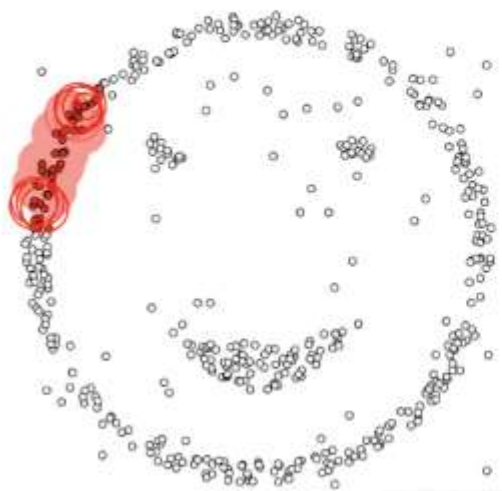


(2) 利用任务 1 中的代码, 对该数据进行 DBSCAN 聚类, 保存结果, 图名为  
“eps=\*, min\_Pts=\*, 班级姓名”

## 任务 3: 自定义数据集

### 任务要求:

(1) 查阅资料, 生成类似如下形式的“笑脸”数据集



(2) 利用任务 1 中的代码, 对该数据进行 DBSCAN 聚类, 保存结果