智能数据挖掘作业7

1920030004 黄铭瑞

# 实验目的

理解DBSCAN聚类算法的原理，并使用DBSCAN聚类算法对2d4c、long、 moon、 sizes5、 smile、 spiral、 square1、 square4数据集进行聚类。

# 实验原理

## DBSCAN聚类

### 定义

DBSCAN（Density-Based Spatial Clustering of Applications with Noise，具有噪声的基于密度的聚类方法）是一种基于密度的空间聚类算法。该算法将具有足够密度的区域划分为簇，并在具有噪声的空间数据库中发现任意形状的簇，DBSCAN算法将“簇”定义为密度相连的点的最大集合。

### 三类数据点

* **核心点（core point）：**若样本的epsilon邻域内至少包含了MinPts个样本，则称为核心点。
* **边界点（border point）：**若样本的epsilon邻域内包含的样本数目小于MinPts，但他在其他核心点的邻域内，则称为边界点。
* **噪声点（noise）：**既不是核心点也不是边界点的点。噪声点是不被聚类纳入的点。

### 密度相关

* **密度直达（directly density reachable）**：如果满足，那么样本点是由样本点对于参数密度直达。
* **密度可达（density reachable）：**如果存在一系列样本点，，任意两个相邻对象之间是直接密度可达，则是关于参数密度可达。
* **密度相连（density connected）：**如果在样本集中存在一个样本点，使得和均由样本点密度可达，那么称与对于参数密度相连。

### 算法步骤

**Step1:** 任选一个点，找到和它距离小于等于的所有点。如果找到的点数小于，标记为噪声点；如果点的个数大于，这个点被标为核心样本，并分配新的簇标签。

**Step2:** 访问点的所有邻居点，如果他们还未被分配到一个簇，那么将刚创建的簇标签分配给他们，如果他们是核心点，就依次访问他们的邻居，以此进行簇扩张，直到内没有更多核心样本。

**Step3:** 重复**step1**、**step2**，直到没有新的簇添加，结束。

## DBSCAN参数选择

### eps选择

参数设置过小，大部分数据不能聚类；参数设置过大，多个簇会归到一个大簇中。

可以通过绘制k-distance曲线的方法寻找合适值。给定邻域参数，对于数据中的每个点，计算对应的第个最近邻域距离，并将数据集所有点对应的最近邻域距离进行从大到小排序后绘图，称这幅图为排序的距离图，该图中明显拐点位置对应的距离值设定为。

### MinPts选择

的选择由指导性原则，，表示数据的维度。如果设为1，则每个点各自成簇，如果设为2，则与层次聚类最近邻与结果相同。因此。若该值选取过小，则稀疏簇中结果由于密度小于，从而被认为是边界点儿不被用于在类的进一步扩展；若该值过大，则密度较大的两个邻近簇可能被合并为同一簇。因此，该值是否设置适当会对聚类结果造成较大影响。

# 实验过程

## 读取数据

定义loadDataSet()，用于数据集的载入。因为给的数据文件是.mat文件，使用scipy.io.loadmat导入数据文件，但是要读取里面的数据，需要找到对应的key。如2d4c里，key有‘a’, ‘moon’, ‘smile’, ‘b’, 而 ‘a’ 才是对应需要的数据。

## DBSCAN算法部分

### 找出epslion范围内的点

定义dist()，计算欧氏距离。

定义eps\_neighbor()，判断两个数据点之间的距离是否在eps范围内。

定义region\_query()，遍历数据集里所有点，把在eps范围内的点找出。

### 判断是否进行簇扩张

定义expand\_cluster()，dbscan()，不满足MinPts条件的列为噪声点，满足的点划分到该簇，如果是核心点，继续访问他们的邻居进行扩张。遍历所有的点，重复操作，返回聚类簇id和聚类簇数目。

### eps选择

定义select\_eps()用于找出最适eps。对于每个数据点，计算其中某个点与其他数据点的欧氏距离之和，按距离进行排序，找出第k个近邻到该数据点的距离，添加到k\_dist序列中，之后k-distance图需要。

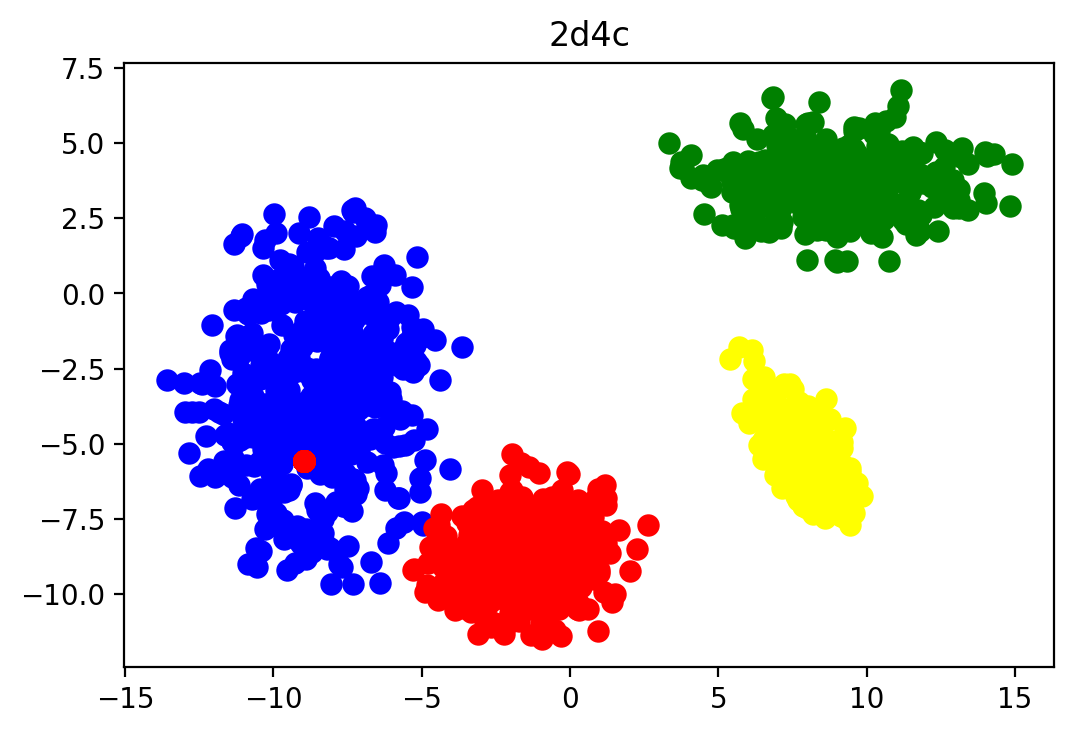
## 绘图部分

定义plotFeature(), 用于对就聚类结果的绘图。每个独特的簇id用同一种颜色标出。有几个不同的聚类簇，就有多少种颜色显示。

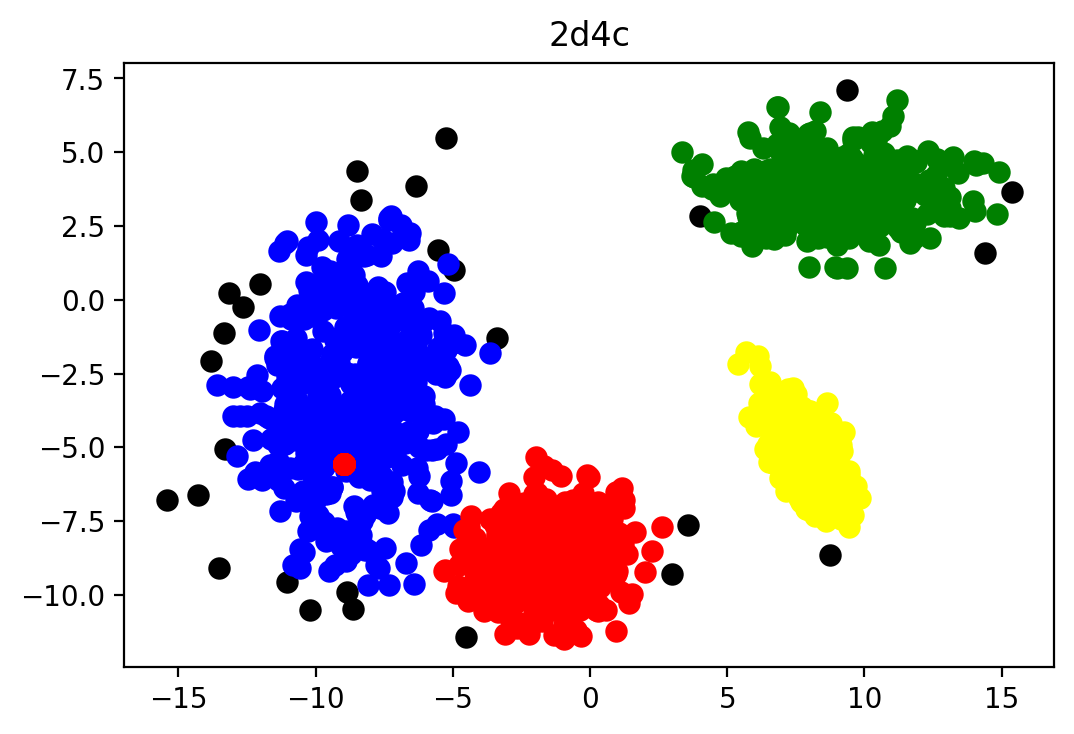
依次对 ‘2d4c’, ‘long’, ‘moon’, ‘sizes5’, ‘smile’, ‘spiral’, ‘square1’, ‘square4’,数据集进行数据读取，DBSCAN聚类，绘制k-distance图，绘制聚类图。

# 实验结果与分析

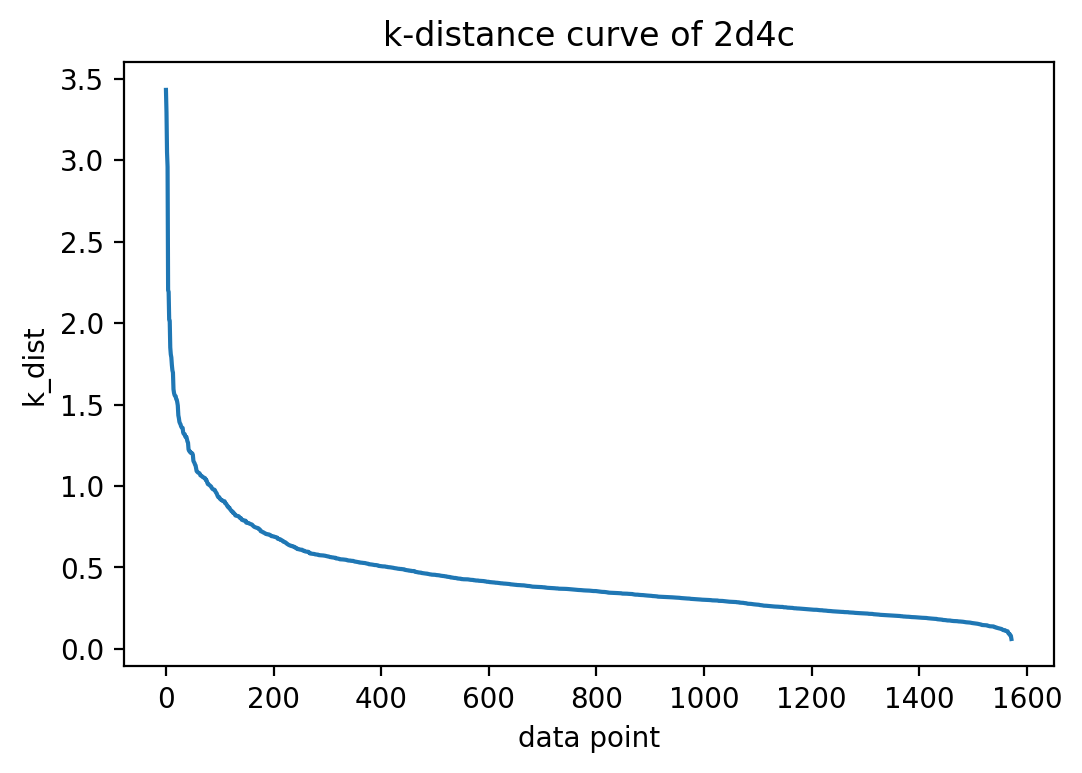
## 2d4c



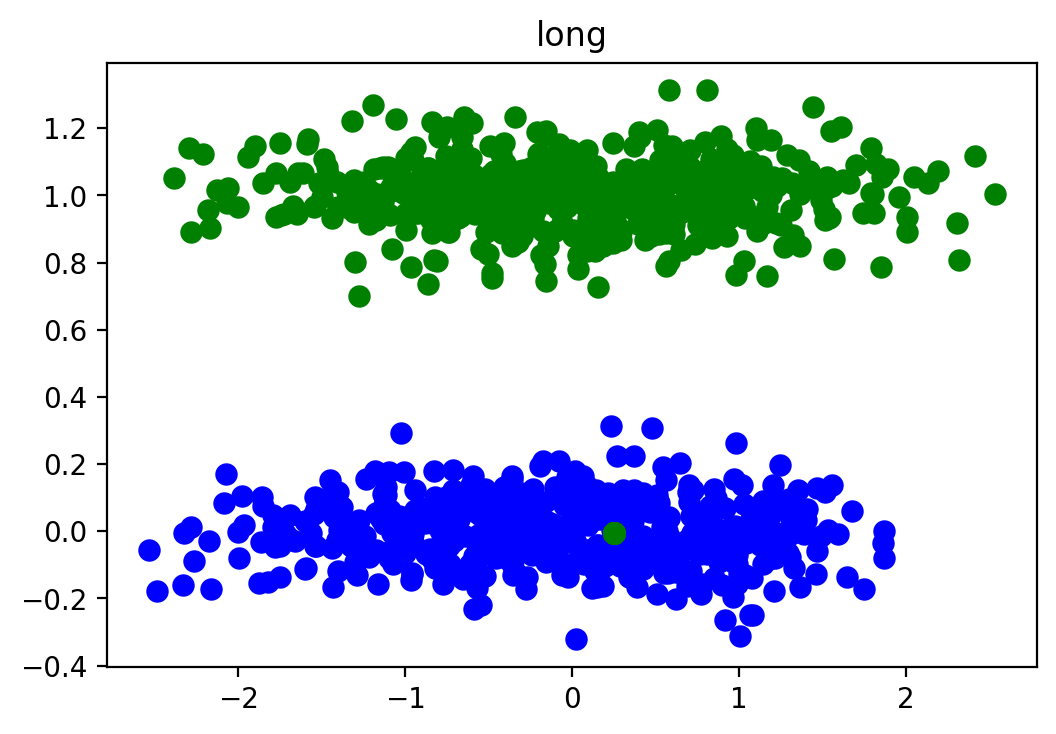
(noise = 0.1)



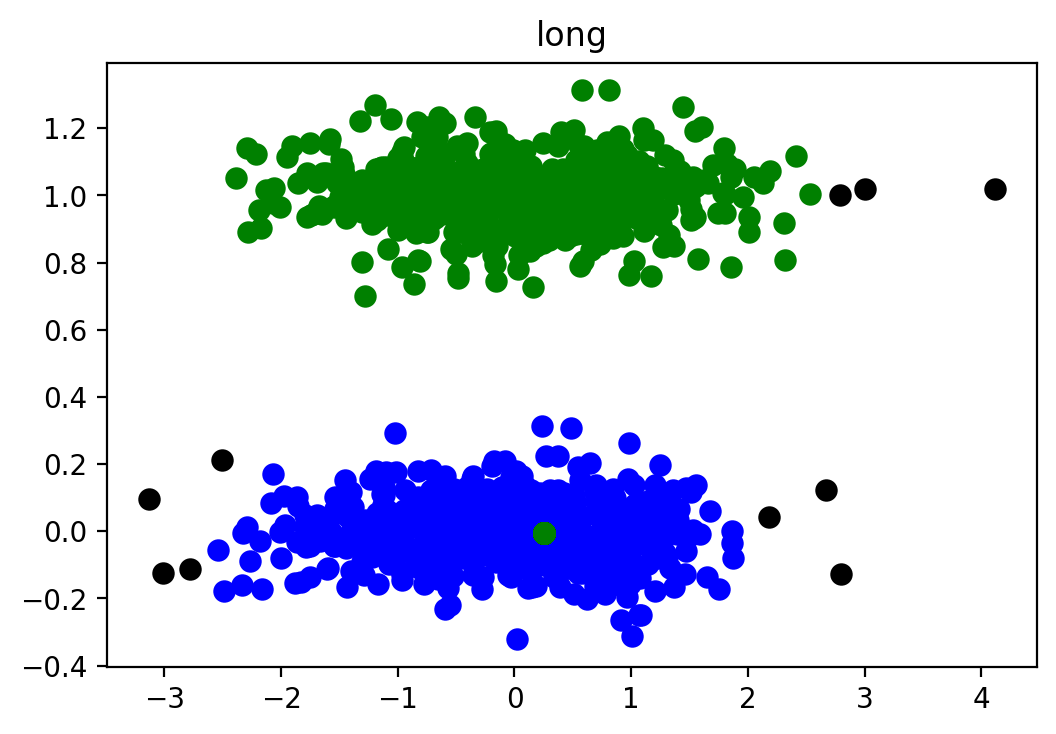
(noise = 0)



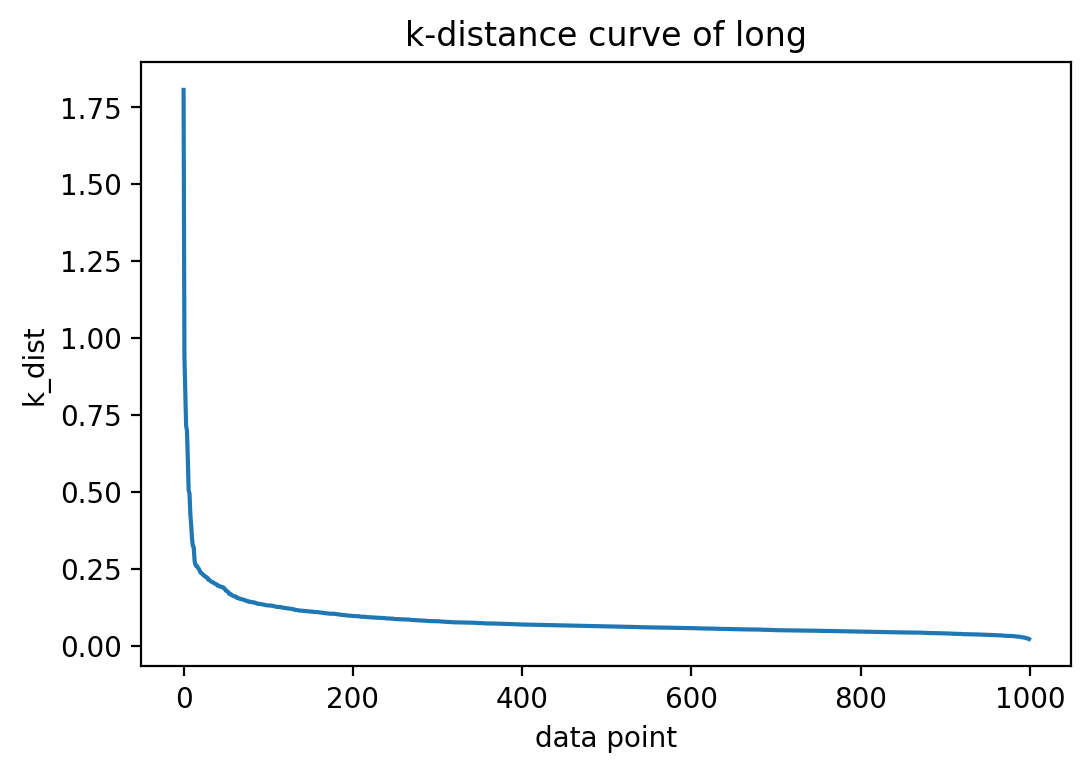
## Long



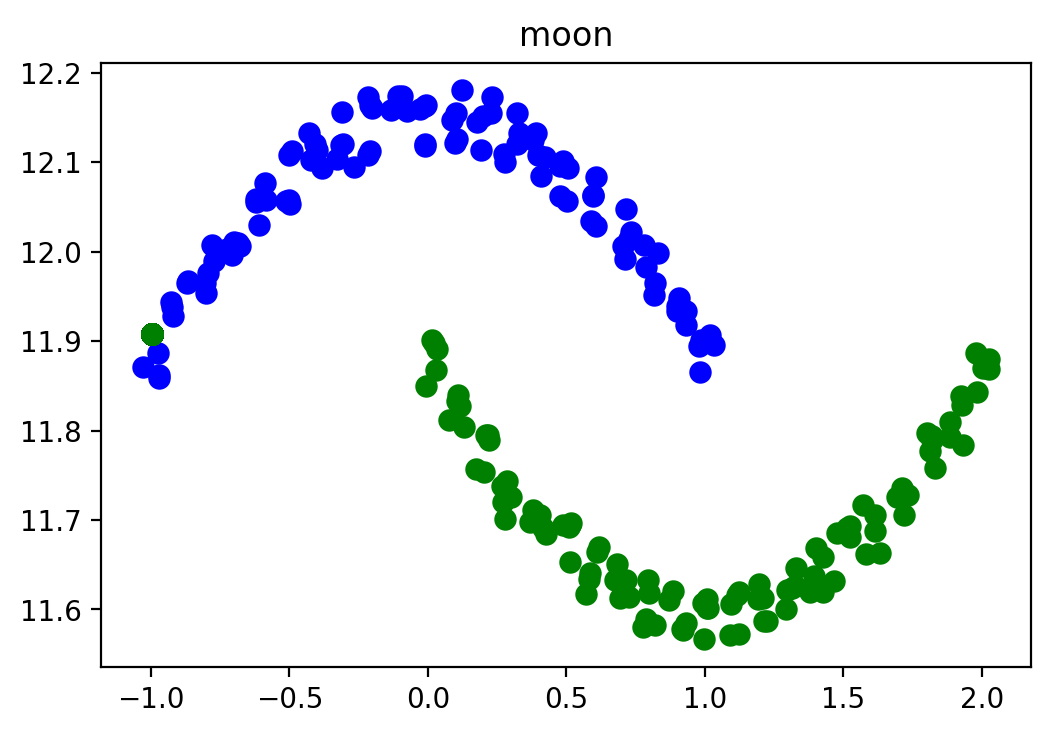
(noise = 0.1)



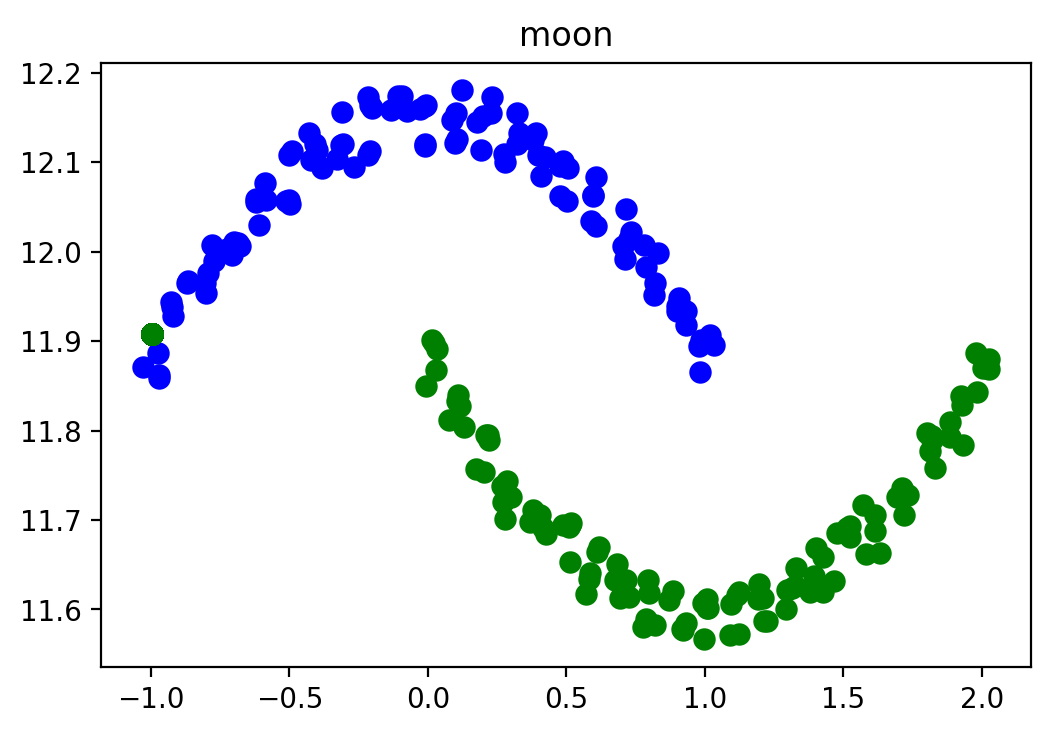
(noise = 0)



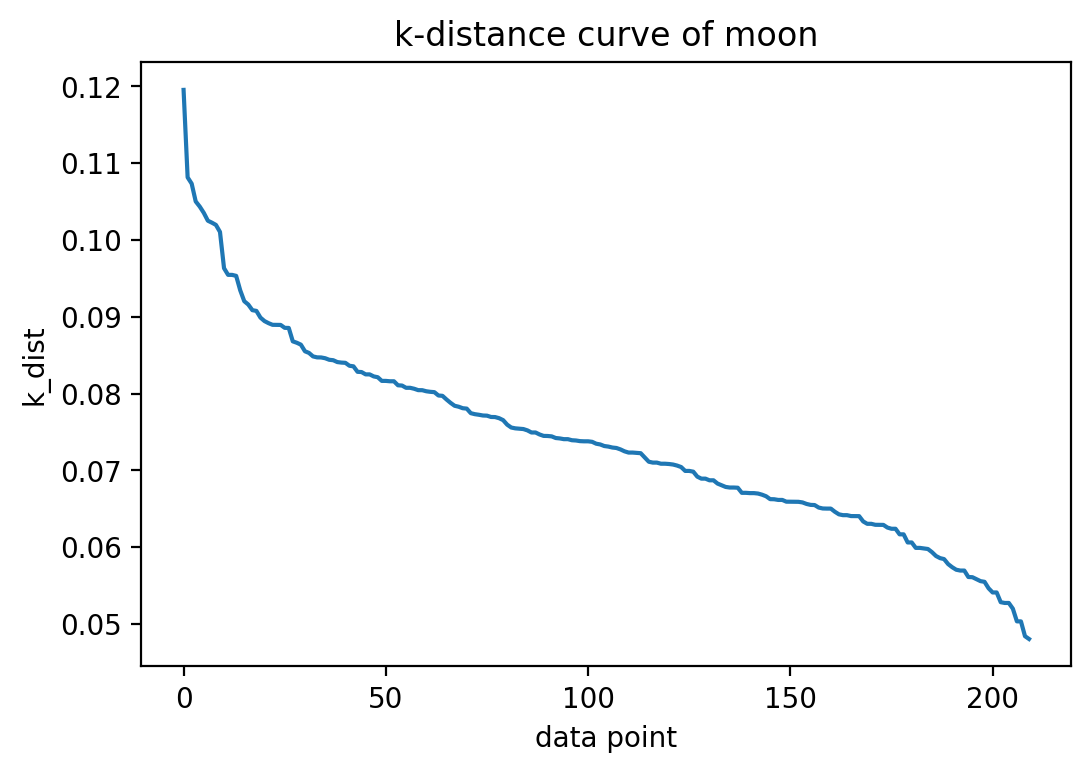
## Moon



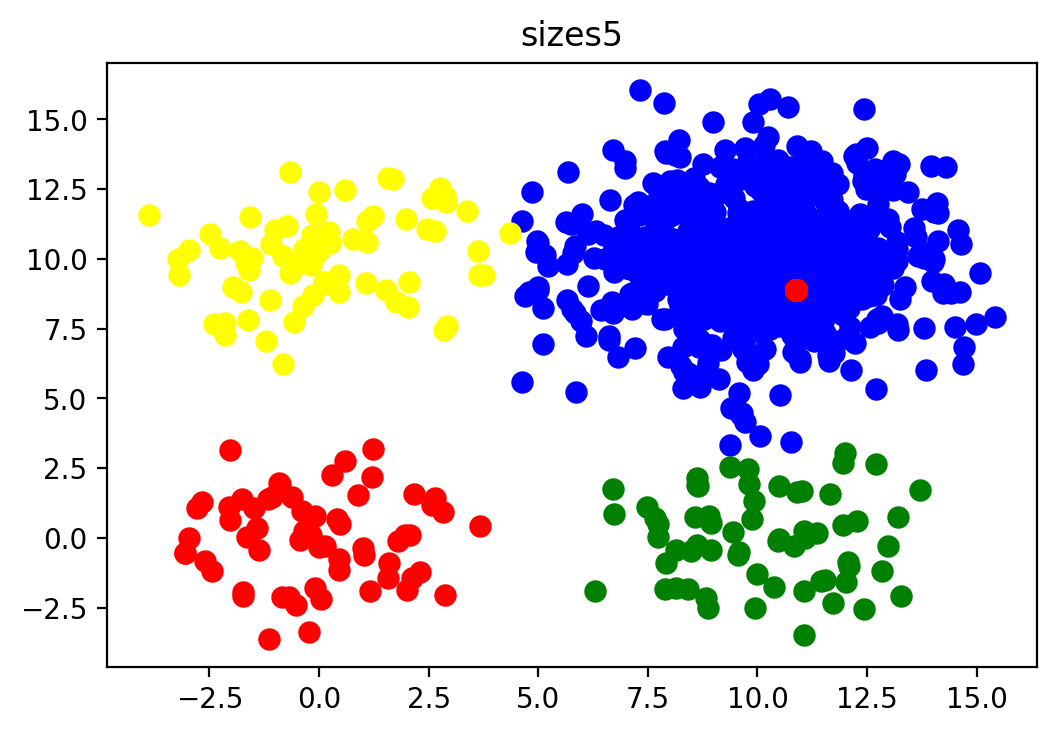
(noise = 0.1)



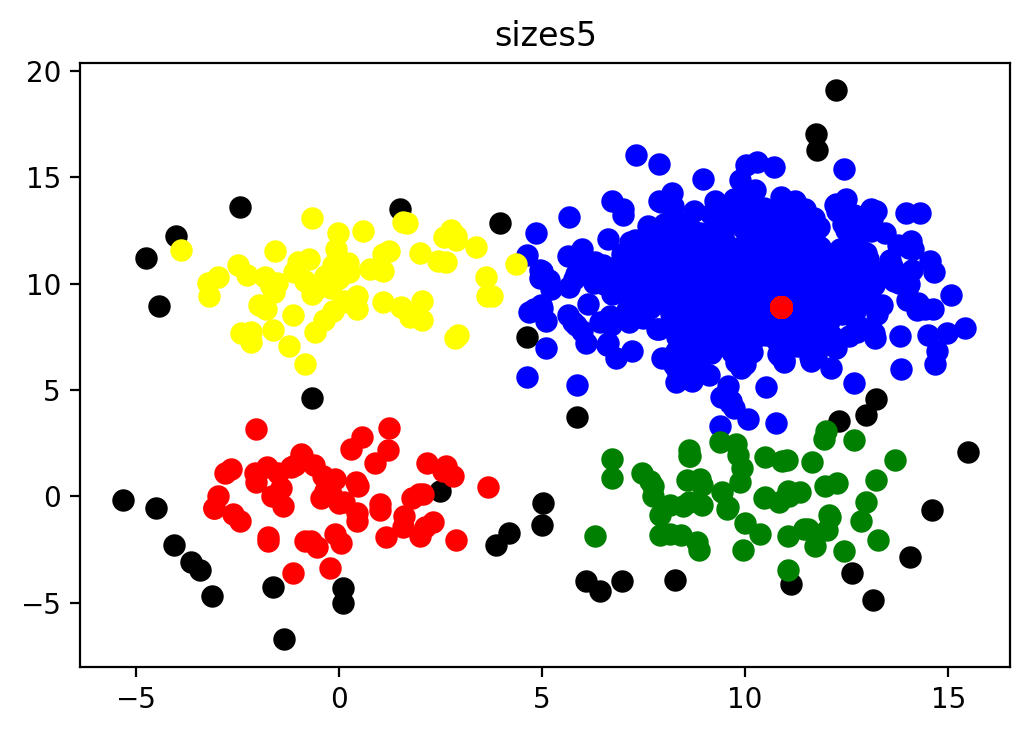
(noise = 0)



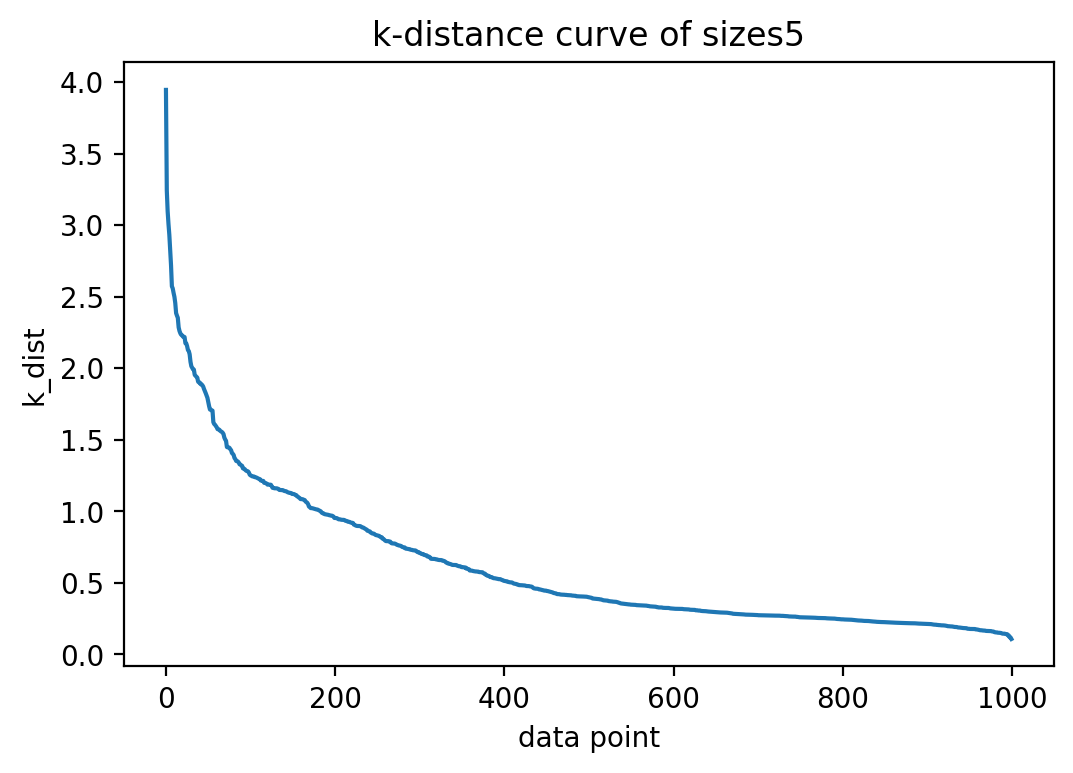
## Sizes5



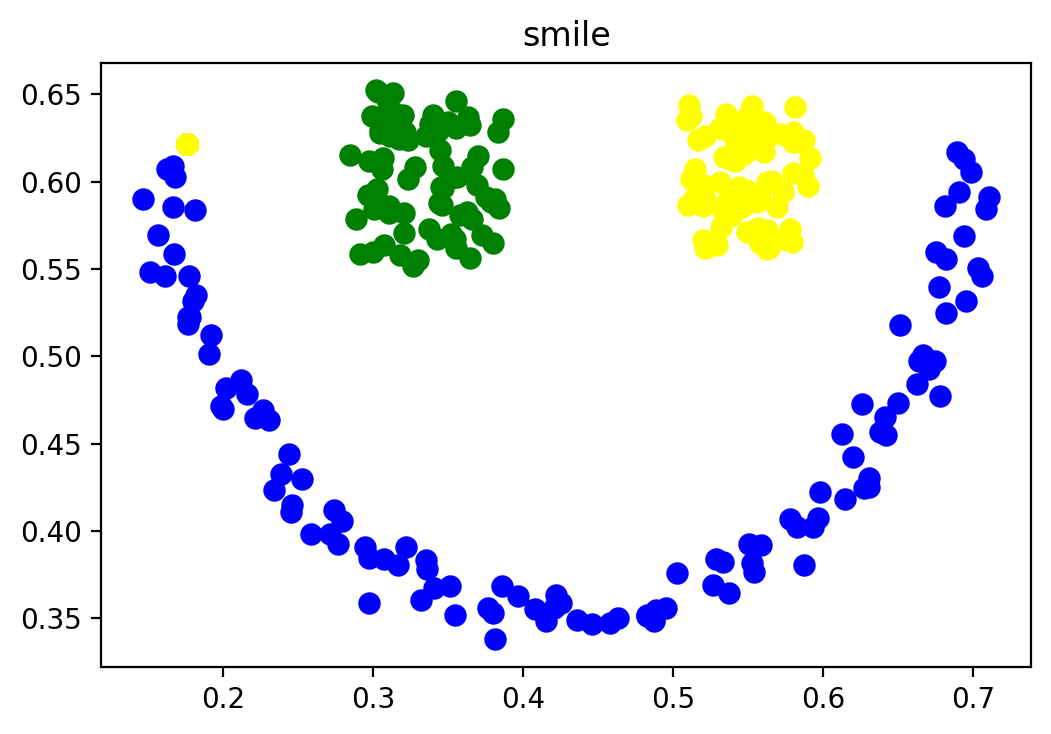
(noise = 0.1)



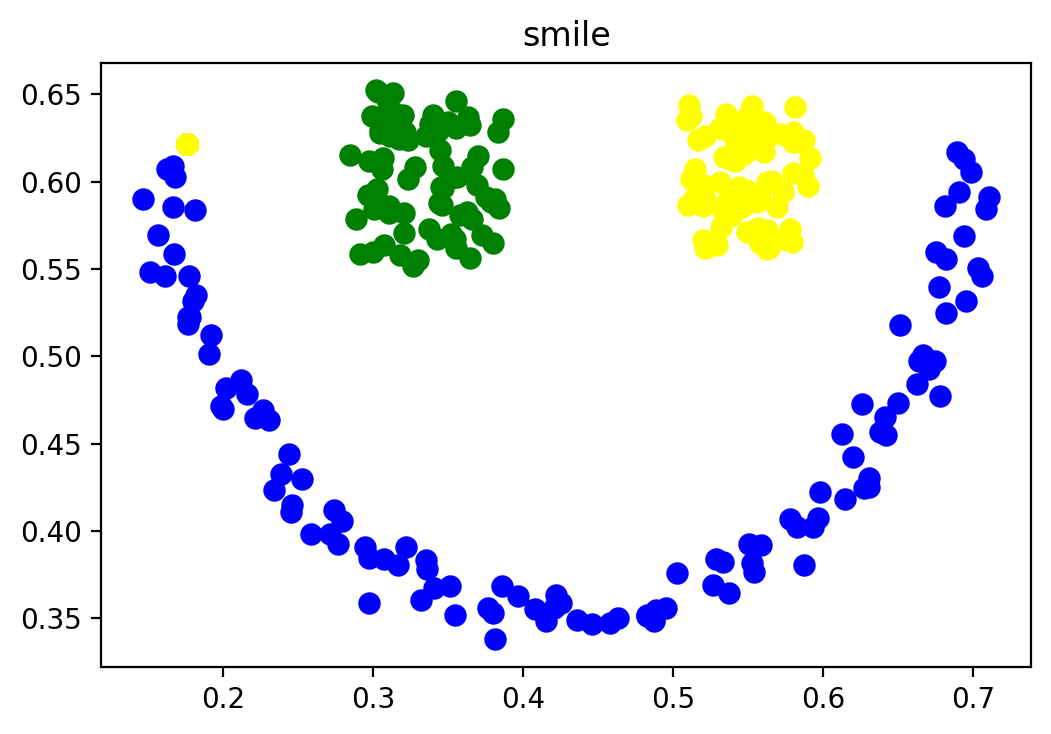
(noise = 0)



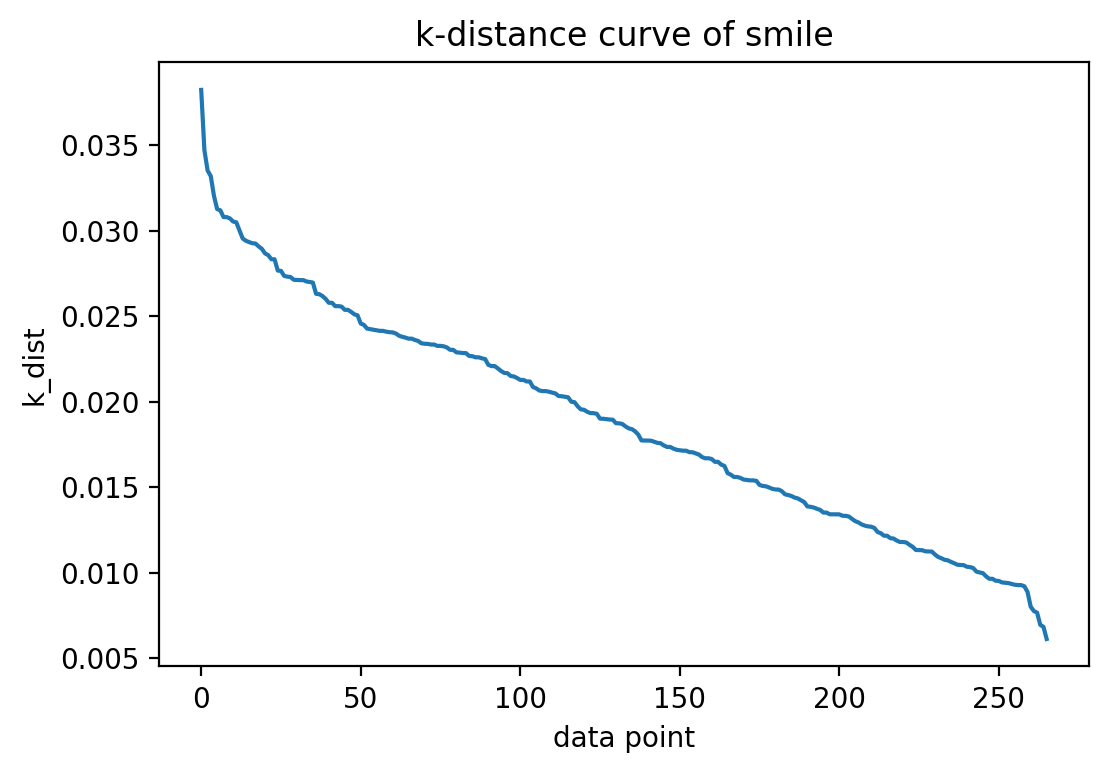
## Smile



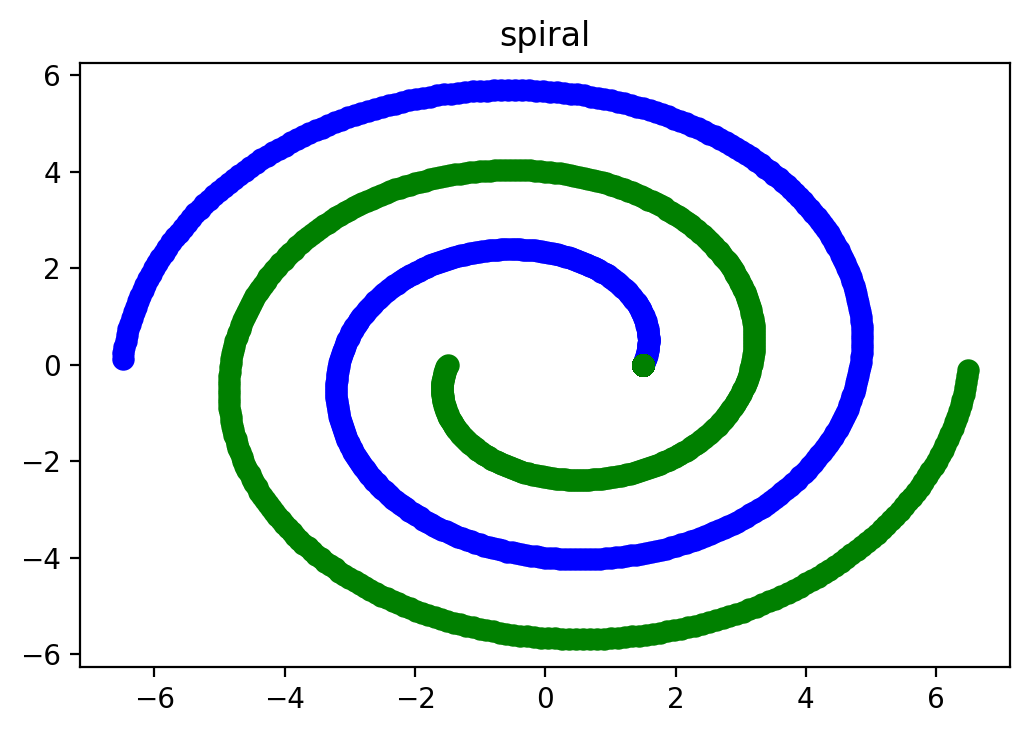
(noise = 0.1)



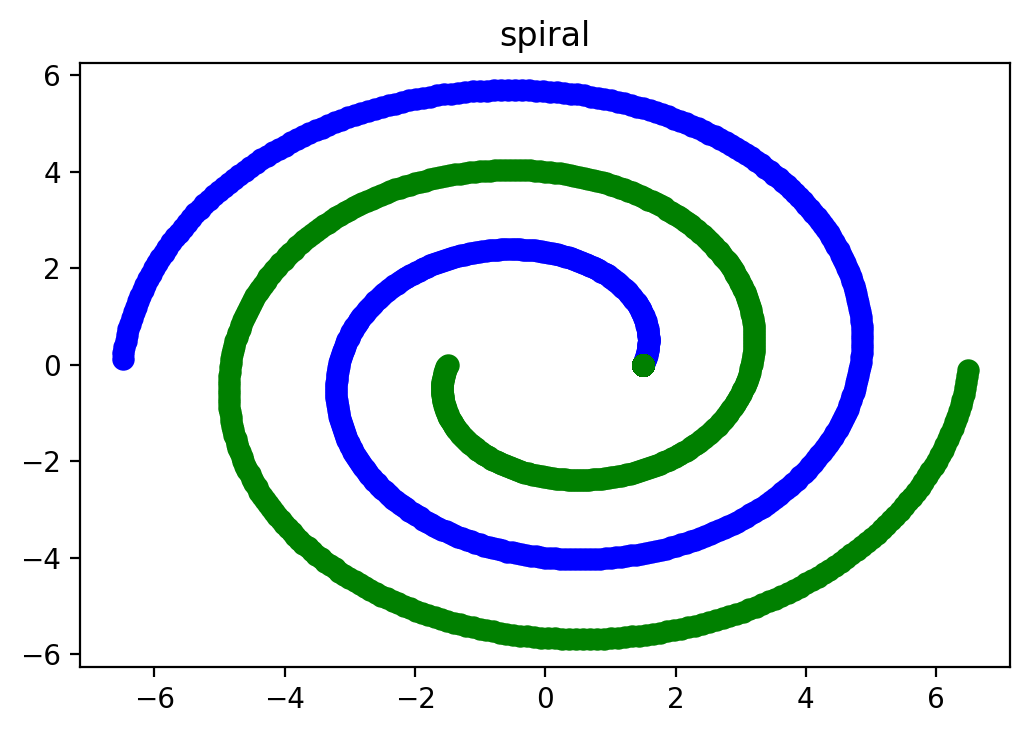
(noise = 0)



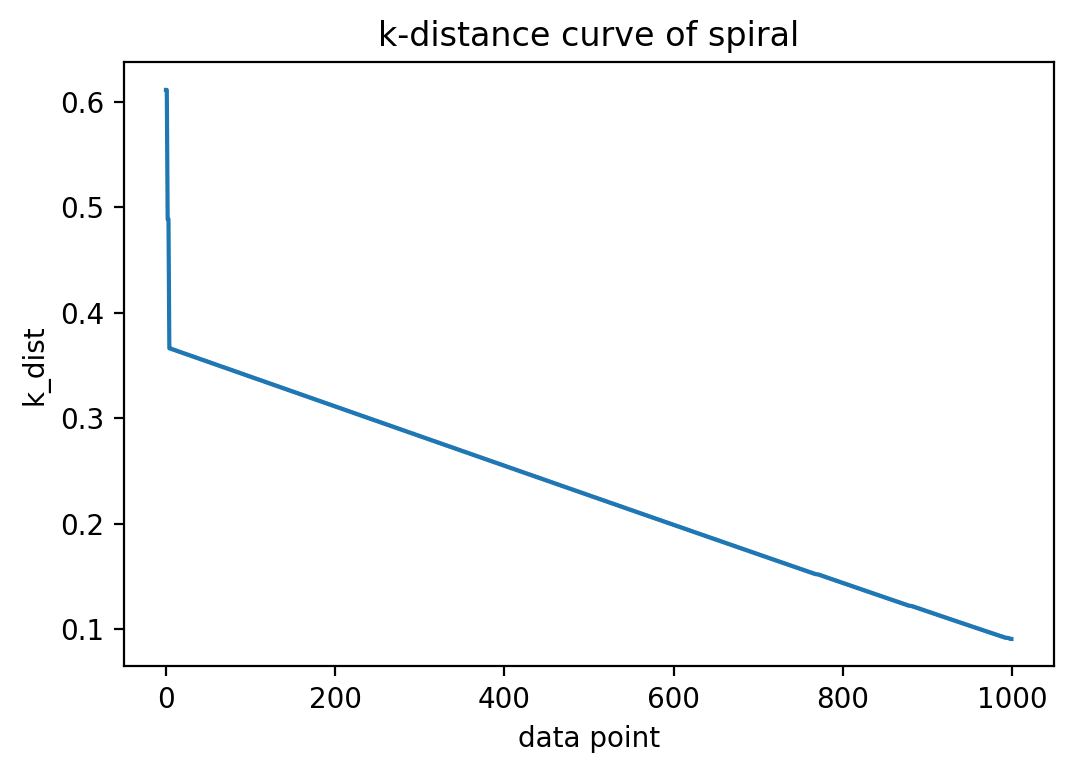
## Spiral



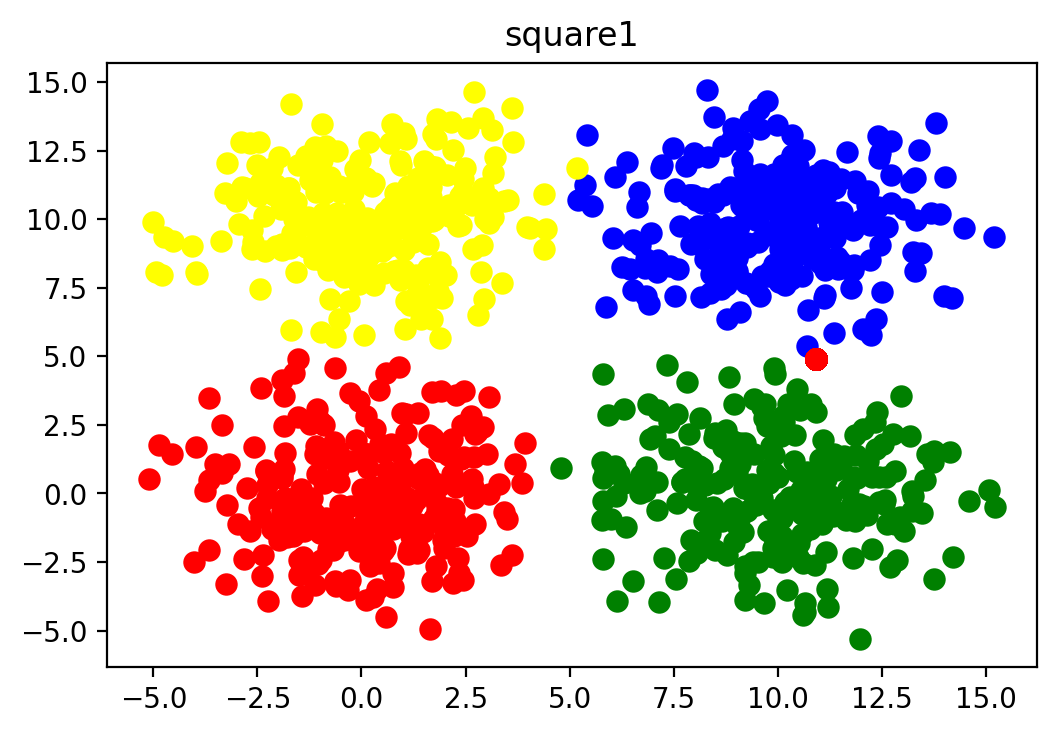
(noise = 0.1)



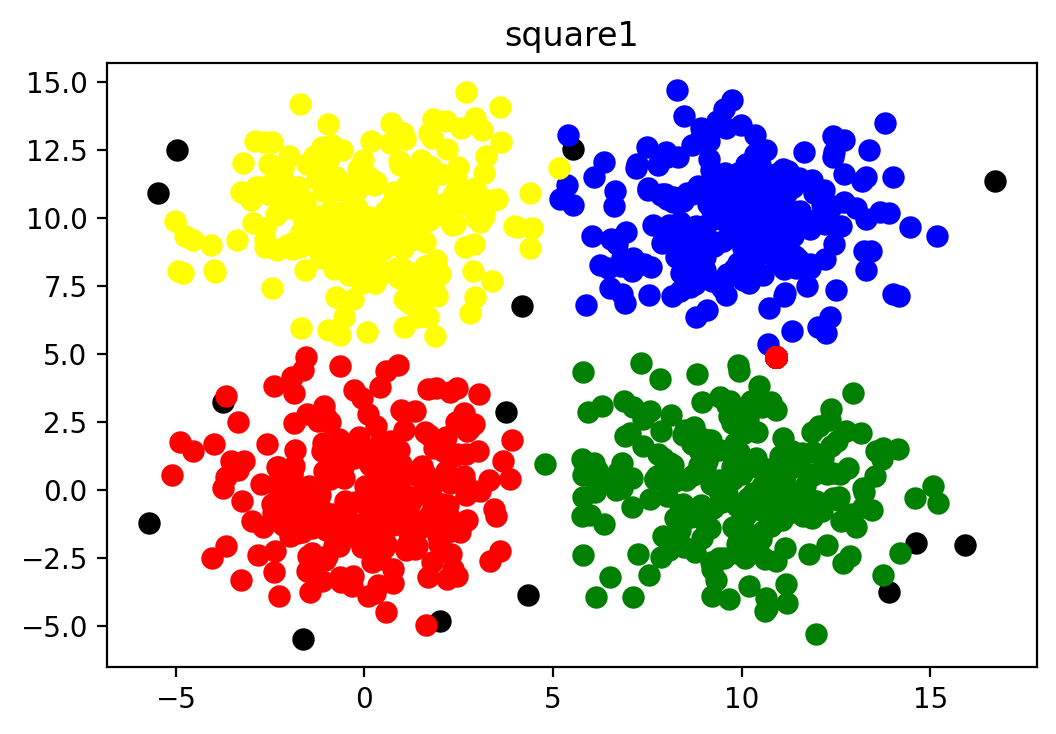
(noise = 0)



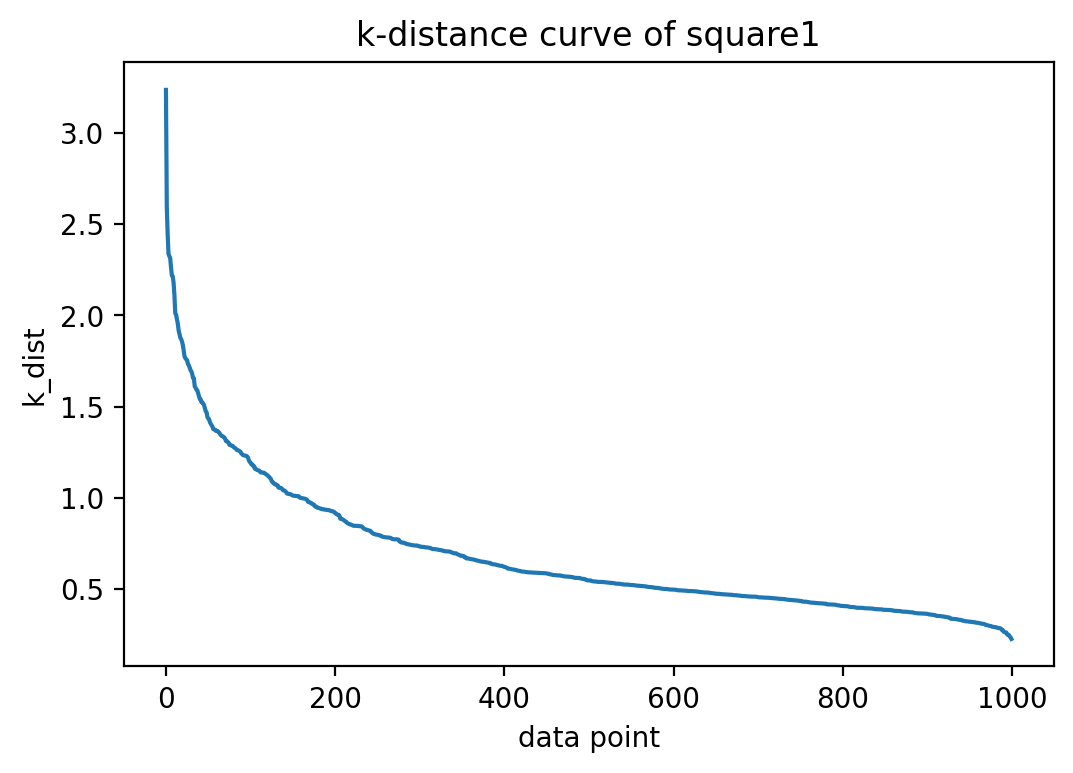
## Square1



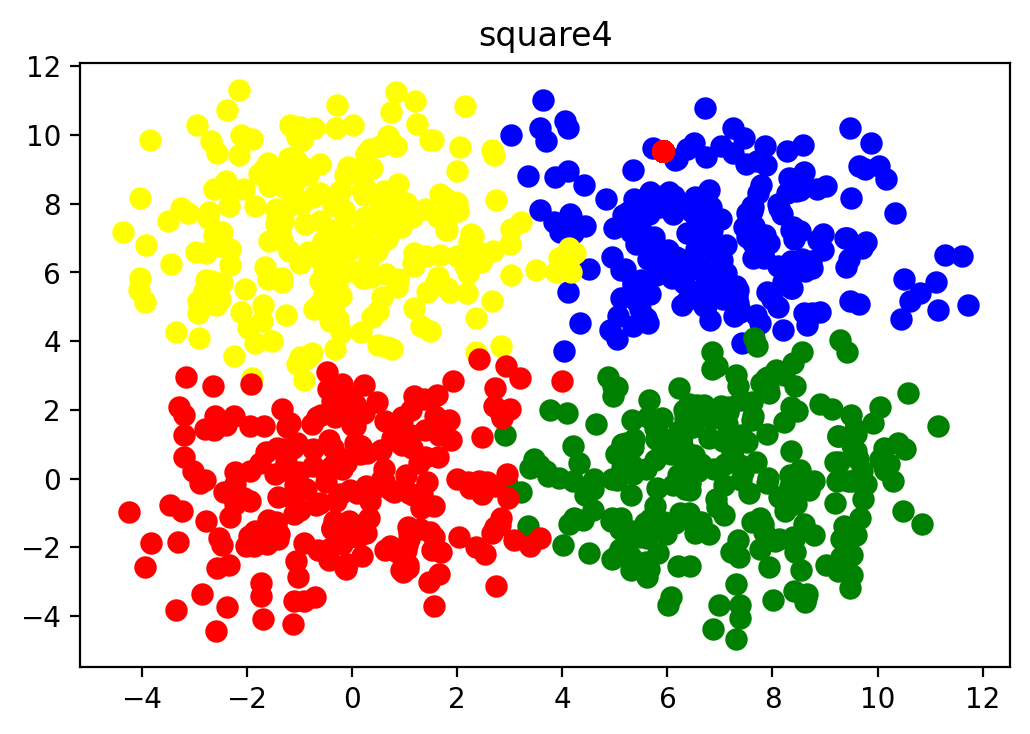
(noise = 0.1)



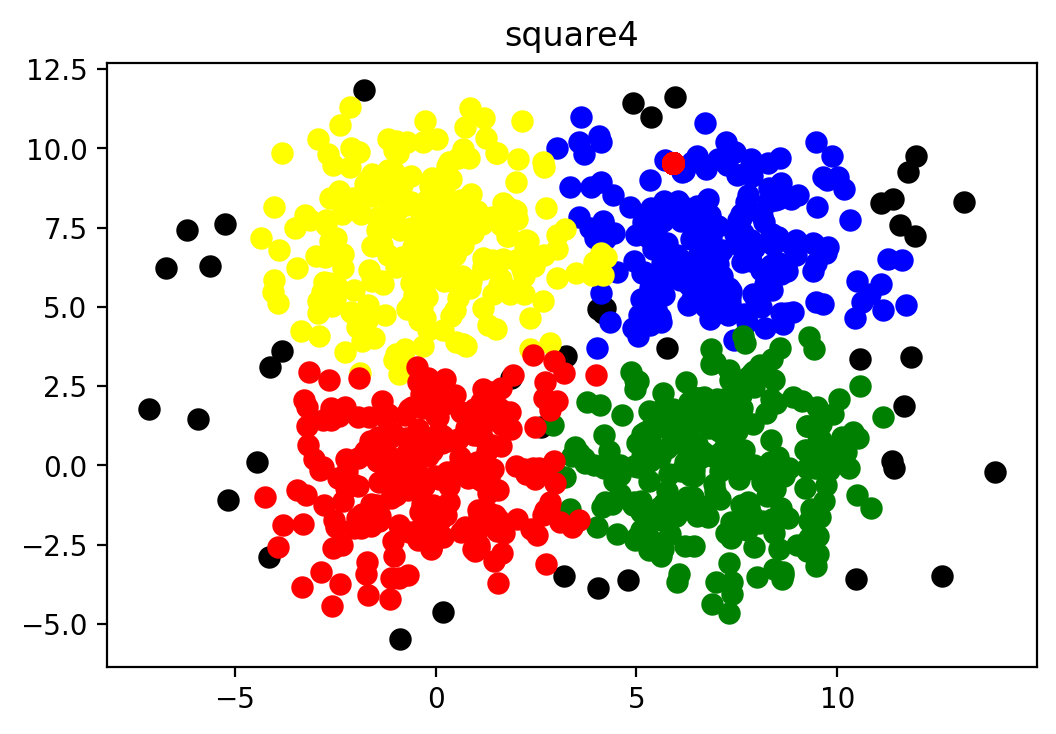
(noise= 0)



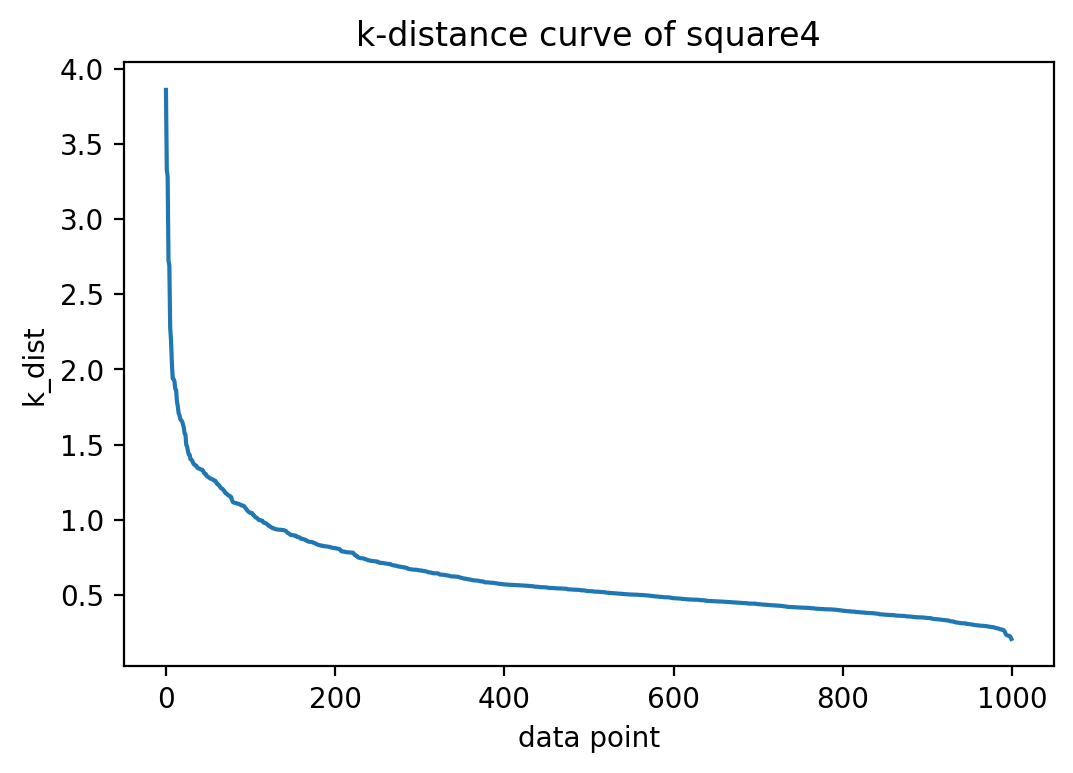
## Square4



(noise = 0.1)



(noise = 0)



## 结果分析

1. 对于‘moon’，‘smile’，‘spiral’这三个分离明显，聚类间差距较大的数据集，不管添不添加噪声，所得到的聚类结果都很好。但是对于‘2d4c’，‘long’，‘sizes5’，‘square1’，‘square4’这五个数据点分布较复杂，密度不均匀的数据集，不添加噪声点的情况下，会遗留有有未归类的噪点，但是对原始数据加了0.1的噪声后，反而聚类结果变得更好。
2. 对于2d4c’，‘sizes5’，‘square1’，‘square4’这四个密度分布不均匀，簇间界限模糊的数据集，eps与MinPts的微调都会对聚类结果造成较大的影响，尤其是未加噪声的原始数据集。
3. 比较奇怪的是，几乎所有的聚类图里，都会有本不属于这一簇的点出现在该簇中，而且是仅有一个点，如月亮的月牙尖角处，笑脸的嘴角处，螺旋曲线最内部，都发现有突兀点，暂时找不到是哪出了问题。

# 附录

<DBSCAN.html>