 

**智能信息系统综合实践**

**实验报告**

|  |  |
| --- | --- |
| **题 目：** |  |
| **年 级：** | **2021** |
| **专 业：** | **软件工程** |
| **学 号：** | **2021117405** |
| **姓 名：** | **孙潇桐** |

目录

[1. 题目 2](#_Toc163252110)

[2. 解题步骤 3](#_Toc163252111)

[2.1. 题目一： 3](#_Toc163252112)

[2.1.1. 3](#_Toc163252113)

[2.2. 题目二： 3](#_Toc163252114)

[2.2.1. 3](#_Toc163252115)

[2.3. 题目三： 3](#_Toc163252116)

[2.3.1. 3](#_Toc163252117)

[3. 总结 3](#_Toc163252118)

[4. 附件 4](#_Toc163252119)

[4.1. 完整代码 4](#_Toc163252120)

[4.1.1. .py 4](#_Toc163252121)

[4.1.2. .py 4](#_Toc163252122)

[4.1.3. .py 4](#_Toc163252123)

# 题目

# 解题步骤

## 题目一：

### 

## 题目二：

### 

## 题目三：

### 

# 总结

详细的结论我直接写在了第二节（[2.3.4节](#_聚类结果的展示和算法间的比较)）中的每一次的结果后面，这里再对之前的结果进行一次简单的概括。这次我一共尝试了五种聚类的方案，分别是K-Means（曼哈顿距离），K-Means（欧氏距离），K-Means（余弦相似度），DBSCAN（密度聚类），GMM（混合高斯模型）。但是由于曼哈顿距离和欧式距离都属于闵氏距离，最后对比的时候只是用了曼哈顿模型。对于像 blobs.txt中的泾渭分明的团状数据，不论是使用哪种分类器都可以有非常好的效果。闵氏距离的K-Means，由于需要类似圆形的数据集，所以在上面多个数据集中表现不佳，但是类似圆形的数据集DBSCAN和GMM的表现同样不错，所以可能K-Means的优势只剩速度稍快了吧。而余弦相似度对数据集的要求更是苛刻，只扫从上面6个数据集来看感觉不适合用于聚类。DBSCAN由于更贴合“聚类”这个概念，所以在多数数据集上表现不错。GMM使用了多个正态分布来替代了K-Means中各种距离函数的计算，对正态分布的样本表现最佳。

综上所述，上面我提到的5个聚类方案没有银弹，都需要先对样本进行观察，才能选择出最适合的聚类方式，但是相对来说DBSCAN的适用性是最广的，就是需要多次调节参数。

这次实验花费了我非常多的时间和精力，从复习上个学习学过的各种聚类方式的原理到可视化模型的方法，到思考模型的测试策略。每一个环节都十分重要，我也都遇到了一些困难：

这次我花费最多时间解决的问题就是在亲手余弦相似度的时候遇到的运行效率问题。一开始我使用普通的循环来完成这个过程，但是感觉速度非常慢。通过思考，我想到了通过矩阵运算完成计算余弦相似度的方法。Numpy中的矩阵运算是经过高度优化的，在替换之后效率果然大幅提升，为我节省了很多等待的时间。

这次的实验让我受益匪浅，使我对numpy库和sklearn库的理解更进一步的同时也锻炼了我解决问题的能力。上面提到的两个库都是在机器学习领域非常重要的工具，只有用好他们才能更好的完成实验，并理解这些模型的底层原理。我将继续保持学习，多学，多写，争取有朝一日能成为机器学习大师。

# 附件

## 完整代码

### .py

### .py

### .py