人工智能大作业实验报告

521030910414 张博伦

一、实验原理及设定

采用应用最广泛的K-means作为主体聚类算法，对task1-4进行分类

（一）主要函数：

1. kmeans（data,k）：data表示待分类的ndarray向量组，K表示需要分类的类别数。将数据转为二维后进行二维距离计算，该函数返回最终中心点数组centers（用以优化）与标签数组label
2. dimension_1_to_2(sample):将task1中数据sample升维成二维向量以适应可视化要求
3. pca（X，k）：利用pca原理将高维数据X降维成k维以适应可视化要求

（为了避免多文件可能带来的麻烦，本次实验将所有函数封装在同一文件下）

（二）设计思路：

对于task1、3直接调整维度——聚类——可视化即可

对于task2、4首先（调整维度）可视化，观察确定大致k的范围，再经过调整维度——聚类——可视化多次实验最终确定k值

（三）可视化：

Task1、2直接转为二维散点图即可，无信息损失

Task3、4考虑到降维后可视化观察时有信息损失，降维可视化维度应尽量高，故转为二、三维散点图对比观察（二、（三）中优化kmeans后可转为三维散点图对比观察）

二、实验结果分析与优化

（一）测试模块准备

利用 for 循环多次运行程序，每次测试运行 100 次程序，并进行结果统计

对 savefig 函数中文件命名进行微调，使每次运行生成图片命名不同从而保存每一次运行的结果以便于统计（例：某次测试结果如下）

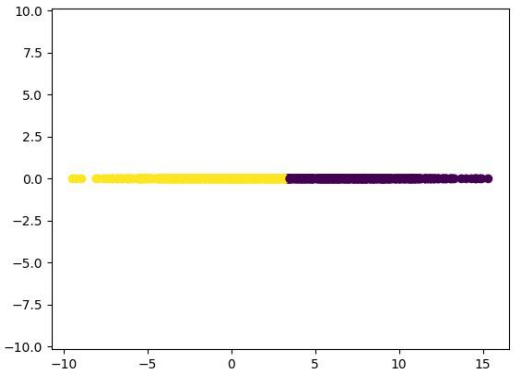
（二）理想聚类结果判定标准

1. 根据可视化散点图，观察得最优聚类

2. 根据百次运行结果，取其中重复出现次数最多的聚类作为最优

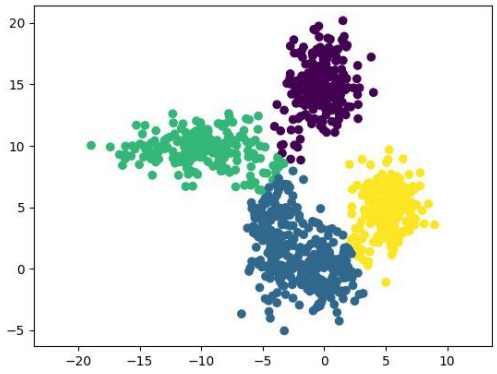
二者综合作为判定标准：

Task1 标准结果：

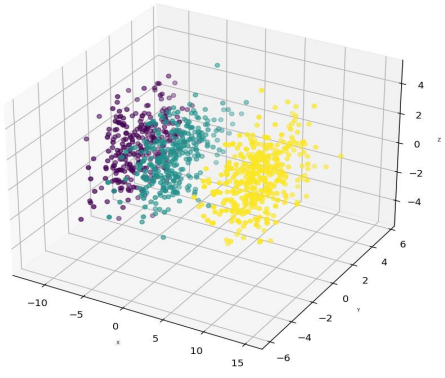


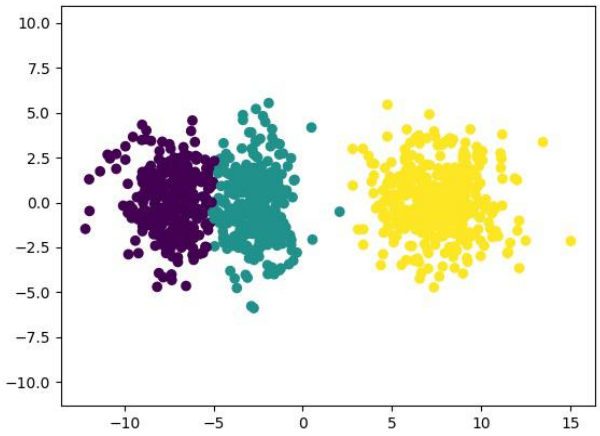
Task2 标准结果：

经过散点图观察——调整维度——聚类——可视化多次实验确定 k=4



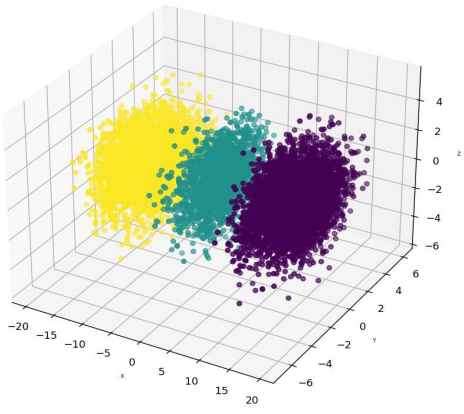
Task3 标准结果：

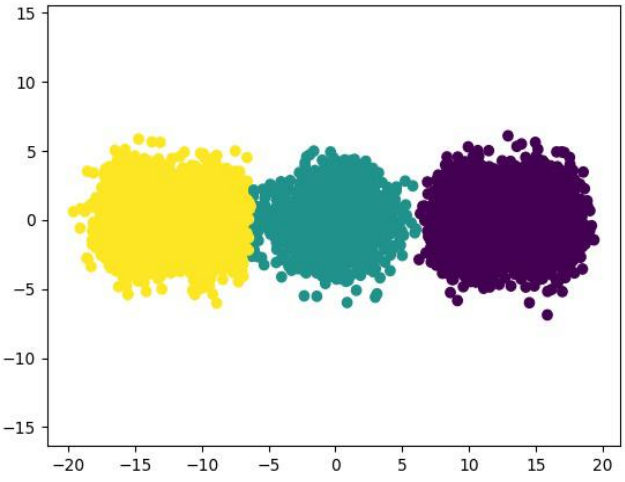




Task4 标准结果：

经过散点图观察——调整维度——聚类——可视化多次实验确定 k=3





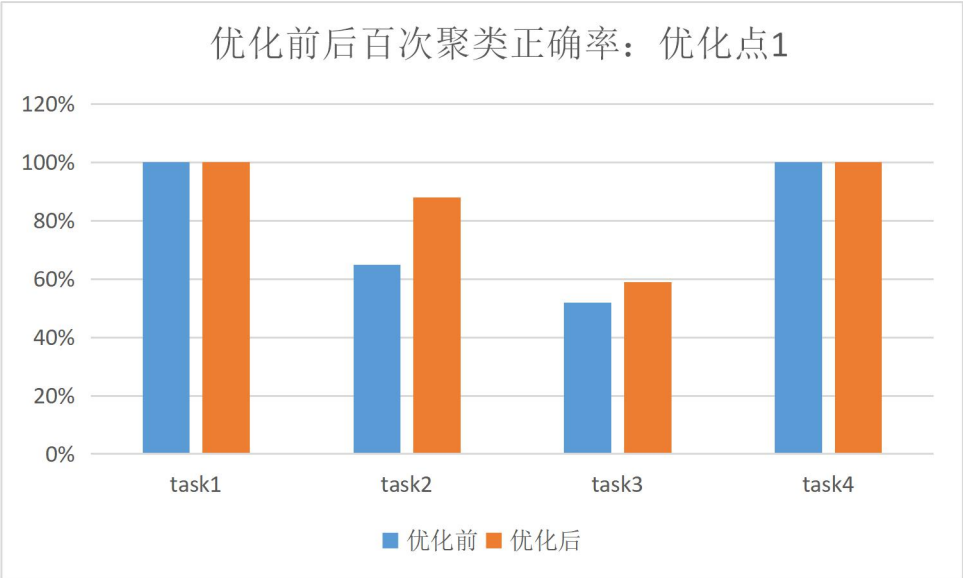
（三）对比实验及优化

优化点 1：随机初始点→样本中随机初始点

——优化前初始点范围任意，易产生异常结果；优化后范围必属于样本范围内，异常结果相对较少

对比试验：

	task1	task2	task3	task4
优化前	100%	65%	52%	100%
优化后	100%	88%	59%	100%



优化点 2:kmeans 计算距离只适用于二维→kmeans 利用向量矩阵计算‘距离’，适用于 2~n 维

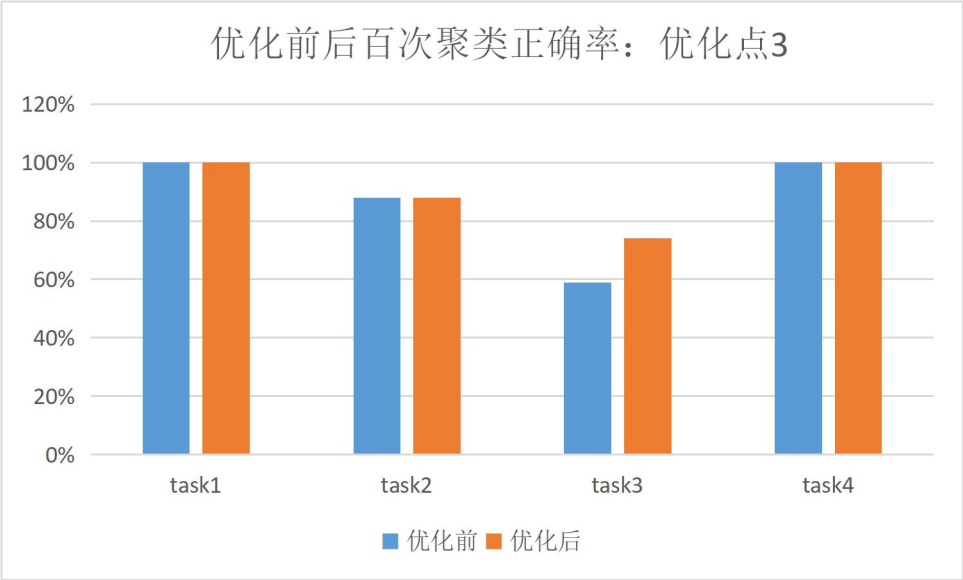
——优化后 kmeans 实用性大幅提升，为之后的进一步优化提供了极大便利

优化点 3：先 pca 降维，再聚类→先聚类，再 pca 降维

——在优化点 2 的基础上，成功避免了 pca 造成的信息损失，同时聚类后的降维操作仅用于可视化，task3、4 可转化为三维散点图观察，便于进一步确定标准结果

对比试验：

	task1	task2	task3	task4
优化前	100%	88%	59%	100%
优化后	100%	88%	74%	100%

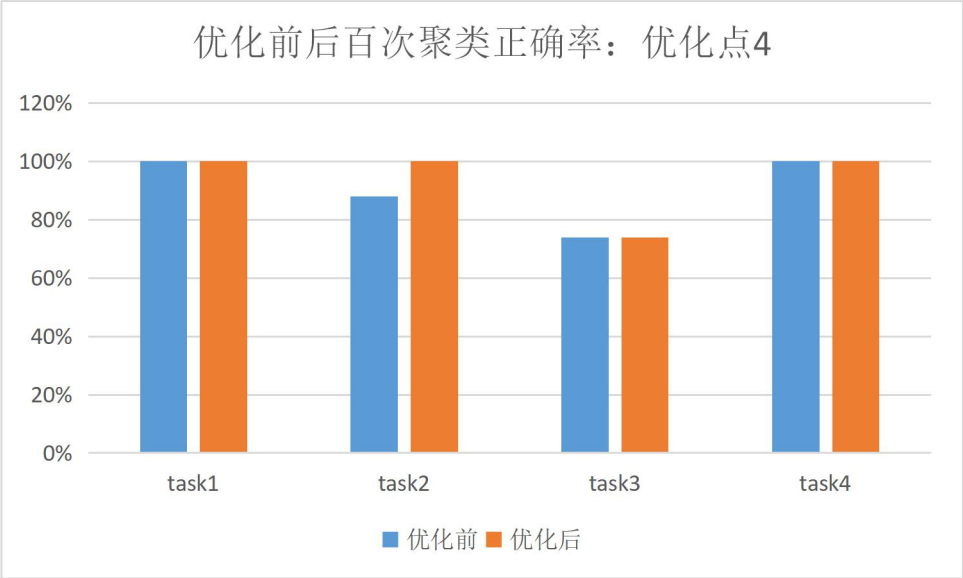


优化点 4：kmeans 算法→kmeans++算法

——将聚类初始点在数据中随机选择的过程优化为 kmeans++聚类中心的初始化过程，其基本原则是使得初始的聚类中心之间的相互距离尽可能远，使得算法准确性大幅提升

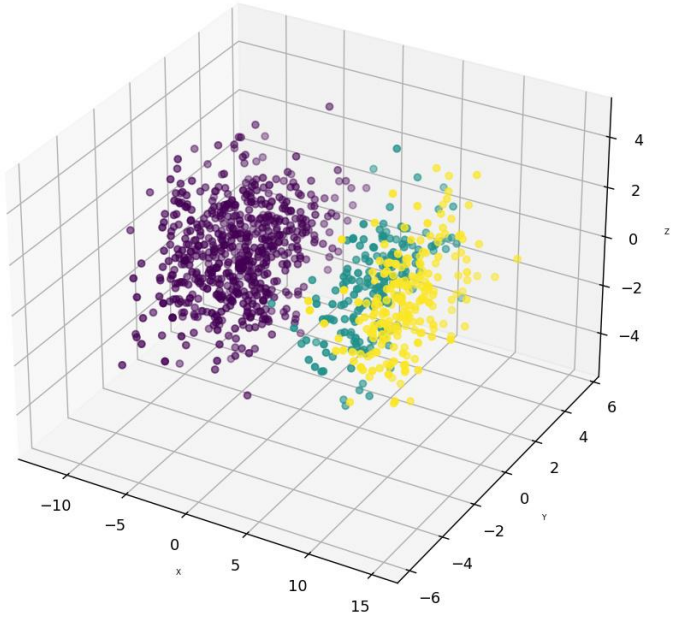
对比试验：

	task1	task2	task3	task4
优化前	100%	88%	74%	100%
优化后	100%	100%	74%	100%



优化点 5：针对 task3 的专门优化

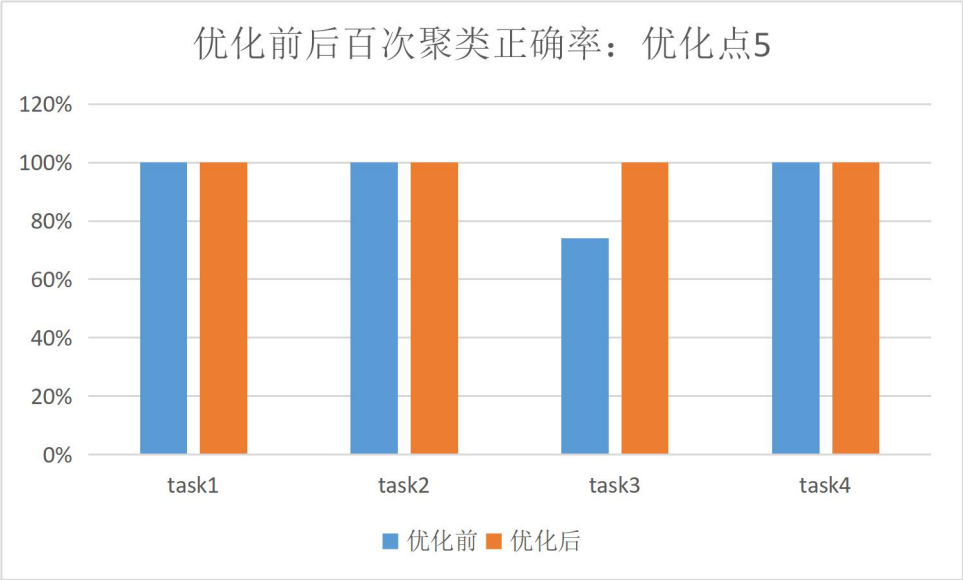
——我们观察到，task3 的所有错误聚类均趋向于同一个结果：



（事实上，该结果亦有其合理性，但根据我们定义的判定标准，以及三维图观察，其合理性低于我们在（二）中列出的结果）

故我们修改 task3 部分代码，将该结果的 label 记录下来，“学习错误”，在修改后的运行中，task3 会将运行结果与 label 逐项比较，相似度大于某个值时（该值为两数组间相似元素个数阈值，通过实验确定），重新聚类，直至结果合理。考虑到 task3 仅有 1000 个向量，时间复杂度并不高，这样以时间复杂度换结果准确性的修改，显然是一种优化

	task1	task2	task3	task4
优化前	100%	100%	74%	100%
优化后	100%	100%	100%	100%



三、总结与反思

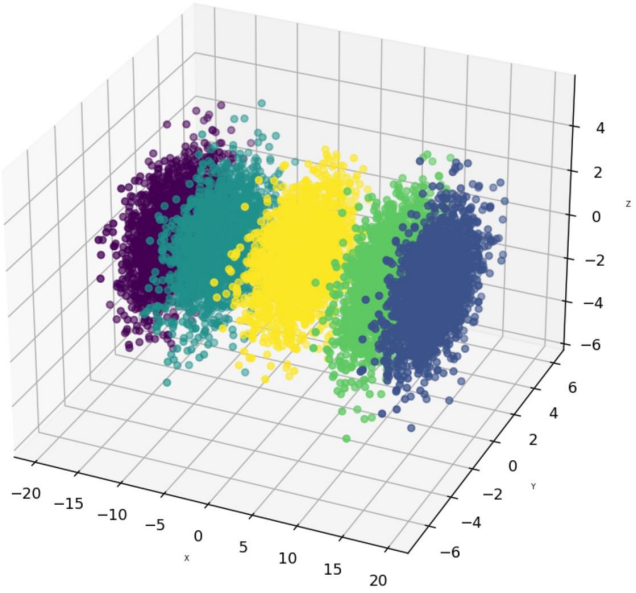
本次实验利用 kmeans（++）成功完成了四个任务，并通过 5 次优化成功使程序聚类稳定性大幅提升，四项任务聚类结果成功率稳定在 100%。同时，在全部优化结束后的百次聚类结果比较中，本实验算法与标准库 sklearn.kmeans 编写程序运行结果基本一致，令人欣慰。对四项任务结果的总结如下：

1、Task1 由于是一维数据的原因,聚类较为容易,初始成功率即稳定在 100%

2、Task2 两次成功率的提升均与 kmeans 初始点的选择优化有关,可以看出 kmeans 算法成功与否与其初始点选择有极大关系

3、Task3 由于是高维数据，受 pca 降维影响较大，故经过优化点 3 后成功率提升较明显。但 kmeans 到 kmeans++的优化对 task3 无影响，此处原因仍存疑

4、Task4 可能由于数据量较大，k=3 时向量相对较集中，聚类效果始终较好，成功率稳定在 100%，同时实验发现，task4 聚类成 k=5 同样有其合理性：



实验的局限性：

1、本实验想要达到的效果并非多次运行取最好结果，而是保证每次运行的质量与稳定性，故几次优化主要着眼于算法稳定性，同时综合时间复杂度，对其它方面的优化相对较少。

2、task3 的优化（优化点 5）专一性较高而普适性较低

3、对实验标准结果优劣的评判含有人为观察的成分，相对而言不够严谨

4、由于 4 项任务向量数都不算高，对时间复杂度优化的力度相对较低，对于巨量数据下该算法时间复杂度的表现尚未进行对应优化

5、本次实验数据分布相对而言比较适合 kmeans，故 kmeans 在本次实验中表现出色，若数据按环状或其他分布，仍需优化 kmeans 或采取新的聚类算法