贛南醫科大學
Gannan Medical University

智能医学工程专业课程设计

基于 KMeans 算法的空气质量状况聚类分析

院　　所：医学信息工程学院　　姓　　名：徐浪

班　　级：22 智能医学工程班　　学　　号 113120220461

二〇二四年六月

摘　要

随着社会和工业的高速发展，环境状况越发严重，其中最为关注的就是空气质量，空气质量差可能会引发多种问题，包括但不限于长期暴露在污染物较高的空气中可能导致呼吸道疾病，如哮喘、慢性阻塞性肺病（COPD）和呼吸道感染的风险增加。颗粒物和化学物质如二氧化硫、氮氧化物等对健康影响尤为显著。当大气污染物的浓度在短期内急剧增高，人吸入大量污染物后可造成急性中毒。如人吸入了形成煤烟型烟雾，病人可出现咳嗽、胸闷、呼吸困难，并伴有头疼、呕吐、发绀等症，对于老年人、婴幼儿及患有慢性呼吸道疾病和心血管疾病等病人，影响尤为严重，所以在这样一个环境下，对空气质量状况进行分析就尤为重要。

本文首先介绍了空气状况的背景，随后获取数据并对其分析，之后进行清洗，包括查看空值、异常数据的处理等，使得之后对数据的操作更有可信度，模型预测也更加准确，之后对其进行数据预处理，包括了数据标准化，使得其原始数据转化为无量纲化指标测评值，各指标值处于同一数量级别，可进行综合测评分析。之后通过特征选择对数据进行降维，去除冗余信息，减少过拟合，从而提高算法效率。随后通过使用 sklearn 库的函数，进行模型构建，并通过可视化的方式呈现了分析的数据结果，包括折线图、散点图等。最后通过模型的内的迭代，得出最优参数，并通过图像对其参数进行直观性分析，最终得出本文的研究的空气等级划分。

本文通过 KMeans 模型的可视化分析得出关于空气质量等级划分，这让人们对于空气的污染会有一个比较明显的认知程度，从而根据本文模型得出空气的受污染程度，最终会减少一些因空气而导致的问题。

关键词：KMeans；主成分分析；python

目　录

1. 绪　论

1.1 研究背景和意义

随着工业化和城市化进程的加速，空气质量问题日益受到关注。空气质量的好坏直接影响着人们的健康和生活质量，也对生态环境和社会经济发展产生重要影响。也随着手段的不断进步以及获取数据方式越来越多，我们可以通过传感器技术去采集数据，我们能够获取大量的空气质量相关数据，包括各种污染物的浓度、气象条件等。这些丰富的数据为更深入、更精细地分析空气质量状况提供了可能。还有通过聚类分析，可以将不同地区、不同时间的空气质量状况进行分类，从而更全面地了解空气质量的分布模式和变化趋势，对于改善环境质量、保障公众健康具有重要的理论和实际意义。

1.2 国内外研究现状

漳州职业技术学院建筑工程学院的陈颂[1]教授采用用模糊综合评价的方法，选取 6 项常见的大气污染物的基本项目 SO2，NO2，CO，O3，PM10 和 PM 2 .5 作为评价因素集合的构成元素，通过建立污染物的模糊关系矩阵和权重集，评价近五年石家庄市的环境空气质量。

东南大学机械工程学院的王帅教授利用聚类分析法研究深圳市各区的空气质量问题，就主要污染物 SO2、NO2、PM10、CO 和 O3 等进行分析，得到各污染物含量之间的关系，以及其相关性程度，从中找到污染程度相当的主要地区，结合其地理位置，从而判断其主要污染源，对同一类地区用相同的方法进行集中治理。

1.3 论文结构

本文的论文分为六个部分，其中　第一部分为绪论，主要讲解了本论文的背景、意义、挑战；第二部分为相关理论和主要技术，主要是描述本论文的；

2. 相关理论和主要技术

2. 1KMeans[3]

KMeans 算法是最常用的聚类算法，主要思想是:在给定 K 值和 K 个初始类簇中心点的情况下，把每个点(亦即数据记录)分到离其最近的类簇中心点所代表的类簇中，所有点分配完毕之后，根据一个类簇内的所有点重新计算该类簇的中心点(取平均值)，然后再迭代的进行分配点和更新类簇中心点的步骤，直至类簇中心点的变化很小，或者达到指定的迭代次数（如下图）。并使得聚类结果对应的损失函数最小化，损失函数公式如下：

（**簇**内平方和）

其中第 x_i 代表第 i 个样本，c_i 是 x_i 所属的簇，μ_{ci} 代表簇类中心，是 M 样本总数

（**CH 指数**）

其中 B(k)：簇间的方差；

W(k)：簇内的方差；

N：样本总数；

K：簇的数量

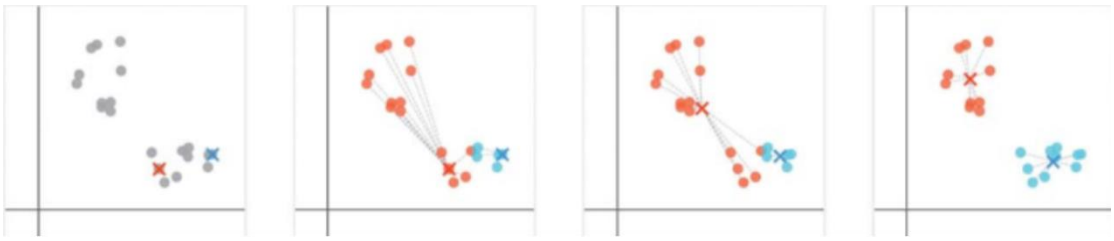

图 1

2. 2 主成分分析

PCA[4]算法是一种常用的数据降维算法，它将 n 维特征映射到 k 维上去，并且还使得这 k 维特征尽可能的去保留原来 n 维数据的信息，它通过如下步骤降维：

- 对矩阵去中心化矩阵 X，如下：

- 求矩阵 X 的协方差矩阵，公示如下：

- 对协方差矩阵特征分解，得到特征向量和特征值
- 计算降维到 k 维后的样本特征，计算公式如下：

其中 W 是根据特征向量对应的特征值降序排序前 k 列矩阵

PCA 适用于数据维度过高，使用 PCA 可以减少数据的维度，从而减少计算量和存储空间；也适用于数据中存在高度相关的变量，使用 PCA 可以将其转换为互不相关的变量，减少冗余信息；适用于数据分布不均匀，使用 PCA 可以将其转换为新的坐标系，使得数据更易于分析和处理。

2.3 python

Python 是一种高级编程语言，具有简单易读、面向对象、解释型等特性，具有广泛的应用领域，包括但不限于 Web 开发、数据分析、人工智能、科学计算、网络爬虫、系统运维等方面。它的语法简洁明了，支持多种数据类型，包括整数、浮点数、字符串、布尔值等，并且是一种动态类型语言，意味着在定义变量时不需要显式声明数据类型。Python 还支持面向对象编程，具有丰富的库和强大的社区支持，使得学习和使用 Python 相对容易。

2.4 pycharm

PyCharm 是一种 Python IDE（Integrated Development Environment，集成开发环境），带有一整套可以帮助用户在使用 Python 语言开发时提高其效率的工具，比如调试、语法高亮、项目管理、代码跳转、智能提示、自动完成、单元测试、版本控制。此外，该 IDE 提供了一些高级功能，以用于支持 Django 框架下的专业 Web 开发。

2.5 sklearn 库

sklearn，全称 Scikit-learn，是一个基于 Python 的开源机器学习库，提供了从数据预处理、特征工程到各种机器学习算法的各种工具函数和类。它的设计哲学是简单、高效和易于使用。它专注于实现常见的机器学习算法和分析工具，并且提供了简单一致的 API，让使用者可以方便地构建模型、进行训练和预测。无论是初学者还是经验丰富的专家，都可以轻松上手并充分利用 sklearn 的功能。

在 sklearn 中，我们可以找到众多机器学习算法的实现，包括分类、回归、降维和聚类等四大类。此外，它还包括了特征提取、数据处理和模型评估等模块。这些模块之间相互协同，可以大大提高机器学习的效率。

本文所用到 sklearn 库函数有 decomposition.PCA，cluster.KMeans，metrics.silhouette_score，preprocessing.StandardScaler。下面是一些对于 sklearn 用于机器学习的一些流程：

i. 获取数据：爬虫爬取，sklearn 内置的数据集，数据库
ii. 数据处理：使用 sklearn 里的 preprocessing.StandardScaler 或者 preprocessing.MinMaxScaler 进行处理
iii. 数据集的划分：使用 model_selection.train_test_split 进行划分
iv. 构建机器学习模型：根据数据选择合适的模型，如分类，回归或聚类的模型
v. 模型评估：使用 sklearn. Metrics 下的方法进行评估，类别如下：
如果是分类问题，计算准确率，精确率以及召回率；
如果是回归问题，计算均分误差，R2 等；
如果是聚类问题，计算轮廓系数等

3. 数据收集与处理

3.1 数据来源

数据来源于和鲸社区的数据集下载，对其信息进行分析汇总如下

特征	数量	类型
AQI	2376	Int64
PM2.5	2376	Int64
PM10	2376	Int64
SO2	2376	Int64
CO	2376	Float64
NO2	2376	Int64
O3_8h	2376	Int64

3.2 数据预处理

添加总的数据预处理结构图。

```
print(df.isnull().sum())
df['CO'] = df['CO'].astype(int)
print(df.info())
print(df[(df['AQI'] == 0) & (df['PM2.5'] == 0) & (df['PM10'] == 0)])
df = df.drop(df[(df['AQI'] == 0) & (df['PM2.5'] == 0) & (df['PM10'] == 0)].index)
tran = StandardScaler()
x=tran.fit_transform(df.iloc[0:, 1:])
```

```
日期       0
AQI      0
PM2.5    0
PM10     0
SO2      0
CO       0
NO2      0
O3_8h    0
dtype: int64
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2376 entries, 0 to 2375
Data columns (total 8 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   日期      2376 non-null   object
 1   AQI     2376 non-null   int64
 2   PM2.5   2376 non-null   int64
 3   PM10    2376 non-null   int64
 4   SO2     2376 non-null   int64
 5   CO      2376 non-null   int32
 6   NO2     2376 non-null   int64
 7   O3_8h   2376 non-null   int64
dtypes: int32(1), int64(6), object(1)
memory usage: 139.3+ KB
```

	日期	AQI	PM2.5	PM10	SO2	CO	NO2	O3_8h
1008	2016-09-06	0	0	0	0	0	0	0
1576	2018-03-28	0	0	0	6	0	52	75
1614	2018-05-05	0	0	0	4	0	50	86
1635	2018-05-26	0	0	0	4	0	36	139
1637	2018-05-28	0	0	0	2	0	25	95

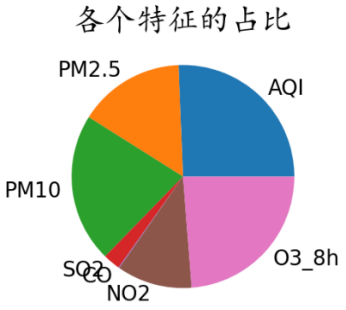

图 2

3.2.1 空值处理

```
日期        0
AQI       0
PM2.5     0
PM10      0
SO2       0
CO        0
NO2       0
O3_8h     0
dtype: int64
```

```
print(df.isnull().sum())
```

3.2.2 标准化

```
pd.set_option('display.max_columns',None)
x = df.iloc[0:,1:]
from sklearn.preprocessing import StandardScaler
tran = StandardScaler()
df=tran.fit_transform(x)
print(df)
```

```
[[ 0.55928786  0.78238112  0.74722173 ...  1.89217411  1.91443765
  -1.3704303 ]
 [-0.28422742  0.01521137 -0.03697836 ...  0.68738108  0.40900671
  -0.82977267]
 [ 0.06221635  0.32207927  0.18923321 ...  1.16929829  0.76322576
  -1.17960996]
 ...
 [ 0.43878567 -0.35984939  1.95368341 ... -0.87884987 -0.56509566
   0.69679006]
 [-0.55535733 -0.93948876 -0.70053228 ... -0.87884987 -1.09642423
   0.37875616]
 [-0.05828583 -0.87129589 -0.76085536 ... -0.63789126 -1.18497899
   1.01482396]]
```

3.3 特征提取

特征抽取是指从原始数据中提取出对于任务有用的、更高级别的信息或特征的过程。在机器学习和数据分析中，原始数据可能包含大量的维度和信息，其中很多信息可能是冗余的、无用的或嘈杂的。特征抽取的目标是通过一系列变换和处理，将原始数据转化为更有信息量、更有区分性的特征，从而改善模型的性能、泛化能力和效率。

利用 PCA 对数据集进行降维，通过最大似然估计选择最优降维特征数量，最后可视化

分析。

```
pca = PCA(n_components='mle')#mle 最大似然估计
x_pca = pca.fit_transform(x)
good_com=len(x_pca[0])
#保留前几个主成分
pca = PCA().fit(x)
act=pca.explained_variance_ratio_
act_sum=np.cumsum(act)
plt.plot( *args: [1,2,3,4,5,6,7],act_sum)
plt.xticks( ticks: [1,2,3,4,5,6,7],fontsize = 20)
plt.yticks(fontsize = 15)
plt.title( label: '特征数目占比',fontsize = 20,family = 'KaiTi')
plt.xlabel( xlabel: '特征选取数量',family = 'KaiTi',fontsize = 25)
plt.ylabel( ylabel: '特征在原始信息的占比',family = 'KaiTi',fontsize = 25)
plt.show()
```

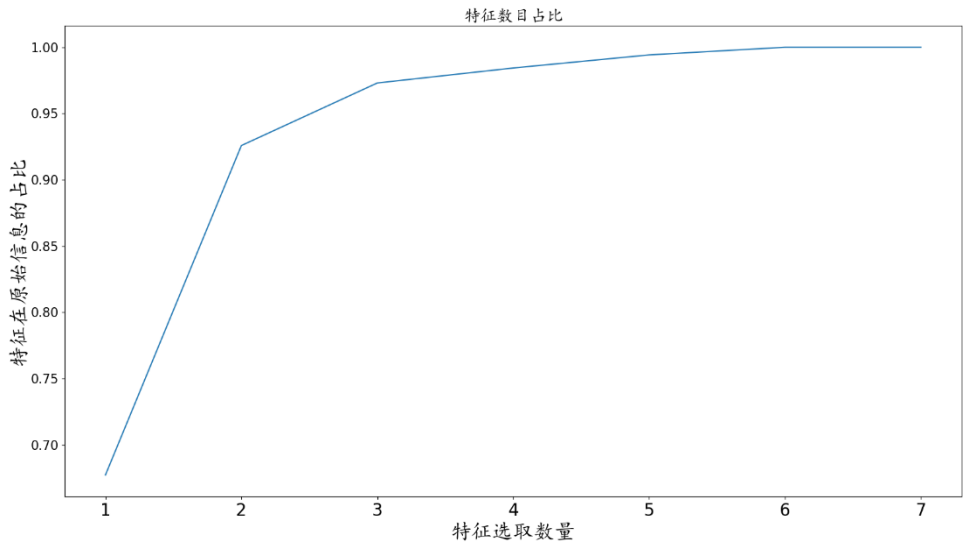

图 3

通过对上图的分析，可以看出随着选取特征数量的增加，其保留原始信息的占比也越高，所以 KMeans 模型构建时的数据降维可以选取 6 个特征。

4. 空气质量模型的构建

4.1 KMeans 模型的构建

通过前面对数据的处理以及特征的降维后，随机选取 K 个中心点，通过计算每个数据点与中心点的距离，来得出中心点的集合点，当数据点归好集合后，再计算每个集合的质心，如果质心与原来的中心点相差太大的话，就重新迭代；如果相差不大，则得到聚类的最终预期结果，计算也就结束，过程如下图：

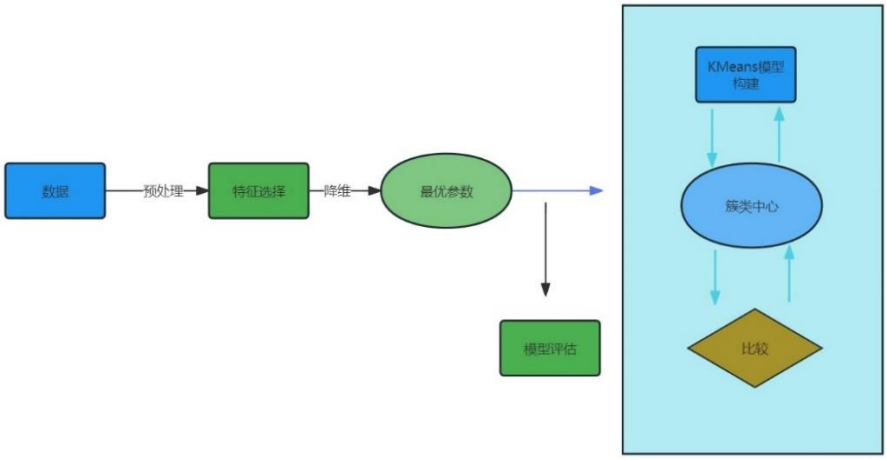

图 4

4.2 模型的评估指标

介绍轮廓系数：

轮廓系数是一种用于评估聚类质量的指标，它结合聚类的内聚度和分离度两种因素来计算样本与与其所属簇内的相似度及最近簇内的不相似度，计算公式如下：

其中 $a(i)$: i 所属簇内其它样本的平均距离;

$b(i)$: i 与其它簇的样本平均距离的最小值;

s (i)的值越接近 1 表示样本聚类合理；越接近 0 表示样本簇类重叠，越接近-1 表示错误分配到相邻簇。

空 1 行

5. 空气质量分析模型的实现

5.1 模型的构建与参数调优

5.1.1 KMeans 模型

```
n_com ,n_clu,sil,temp=0,0,[],[]
plt.figure(figsize= (30,15))
k = [3,4,5,6,7]
pca = PCA(n_components=6).fit_transform(x)
for j in range(len(k)):
        estamtor=KMeans(n_clusters=k[j])
        estamtor.fit(pca)
        pred=estamtor.predict(pca)
        temp.append(silhouette_score(pca, pred))
        sil.append(k[j])
        print('聚类为'+str(j+3)+'的轮廓系数:',silhouette_score(pca,pred))
plt.plot( *args: k,temp)
plt.title( label: 'kmeans簇类中心数目参数比较',family='KaiTi',fontsize = 25)
plt.xticks([3,4,5,6,7])
plt.xlabel( xlabel: '簇类数目',family='KaiTi',fontsize = 25)
plt.ylabel( ylabel: '轮廓系数',family='KaiTi',fontsize = 25)
plt.show()
```

结果绘图

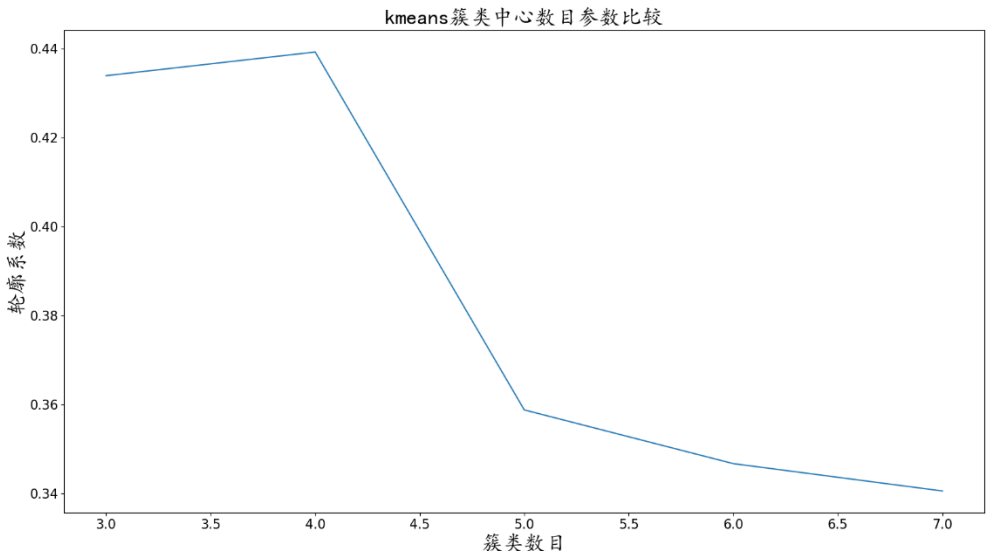

图 5

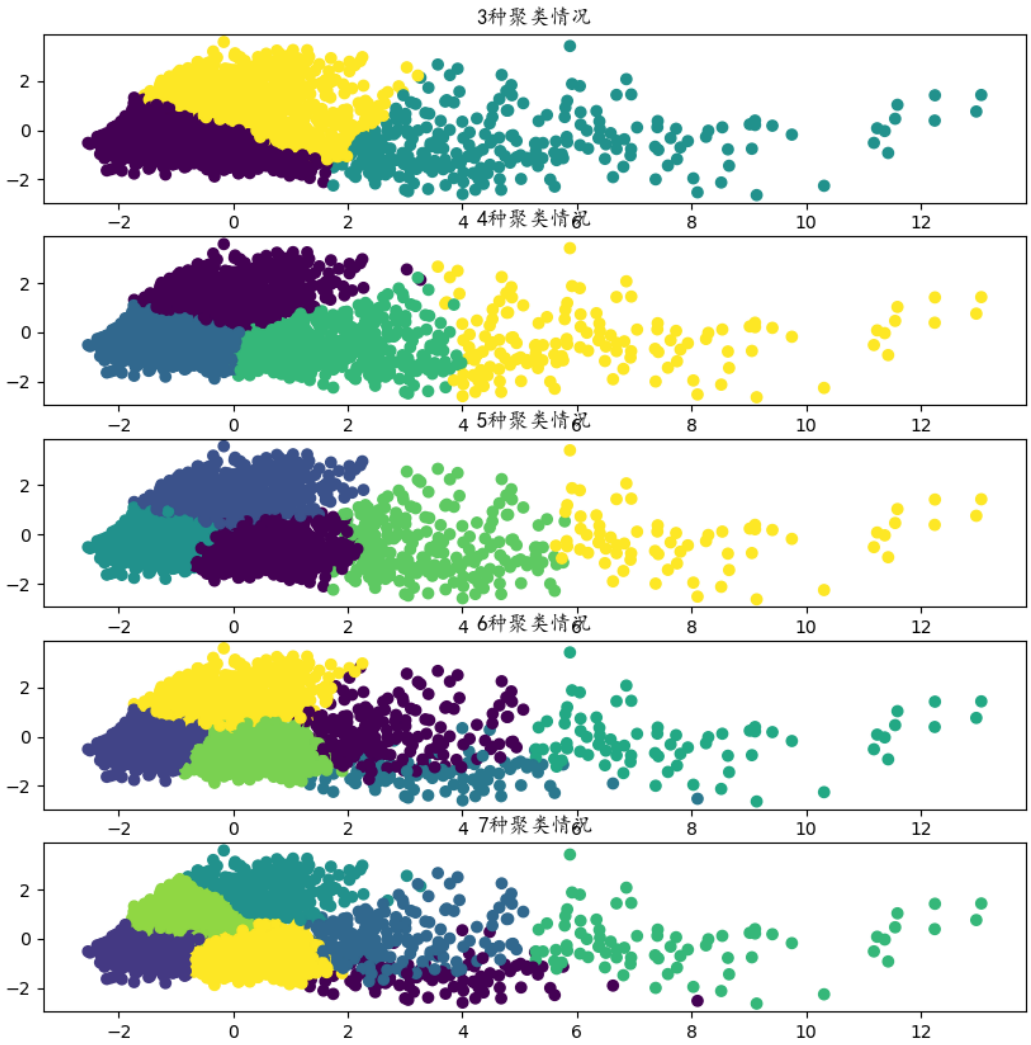

图 6

训练的最优参数下的评估指标表格

表 1 xxx 表

簇类数目	轮廓指数
3	0.434
4	0.439
5	0.358
6	0.347
7	0.330

6. 总结与展望

6.1 模型结果分析

6.11 结论

通过最后模型得出的折线图以及散点图和损失函数的比较，可以看出当簇类数目为 4 时，聚类效果是最好的 ，当簇类数目为 7 时，聚类效果是最差的，由此得出在这个空气质量数据下，当簇类数目变大时，其样本的聚类越合理，而当达到一个界值时，随着聚类数目的增加，其样本间的聚类重叠也就更大。

6.12 模型优缺点

通过 KMeans 模型划分聚类，并令其可视化展示，使得聚类结果更加直观，在数据的处理上，通过 PCA 对多特征进行降维，使得保留的特征更容易被模型所学习，可视化分析也更容易理解，另外为克服少量样本聚类的不准确性，该算法本身具有优化迭代功能，在已经求得的聚类上再次进行迭代修正剪枝确定部分样本的聚类，优化了初始监督学习样本分类不合理的地方；簇间重叠严重，聚合效果不合理，还有 SO2 以及 CO 特征数据与其他特征数据相差太大，另外该算法需要不断的进行样本分类调整，不断的计算调整最新的聚类中心，使得时间复杂度大大增加。

6.13 改进方法

对于 SO2 以及 CO 特征数据与其他特征相差太大，可以把这两个特征合并为一个特征，或者调整样本数据点的距离阈值；时间复杂度问题可以在初始时给定一个 k 值，通过算法得到一次聚类中心，对于这个聚类中心，根据得到的 k 个 聚类的距离情况，合并距离最近的类，因此聚类中心数减少小，当将其用于下次聚类时，相依聚类的数目也就减小，最终得到合适数目的聚类数

参考文献

[1]陈颂. 2018-2022 年石家庄市环境空气质量模糊综合评价[J].石家庄职业技术学院学报,2024,36(2):21-25. DOI:10.3969/j.issn.1009-4873.2024.02.005.
[2] 王帅. 基于聚类分析法的空气质量分析[J]. 城市建设理论研究（电子版）,2012(36).
[3]岳珊,雍巧玲.基于确定初始簇心的优化 K-means 算法[J].数字技术与应用,2023, 41(11):140-142.
[4]朱梦滢，杨思悦，罗冰洁，吴晓东.基于 Kraljic 矩阵和 PCA 算法的手术器械 分类管理研究[J].中国医疗设备, 2024, 39(4): 116-121.