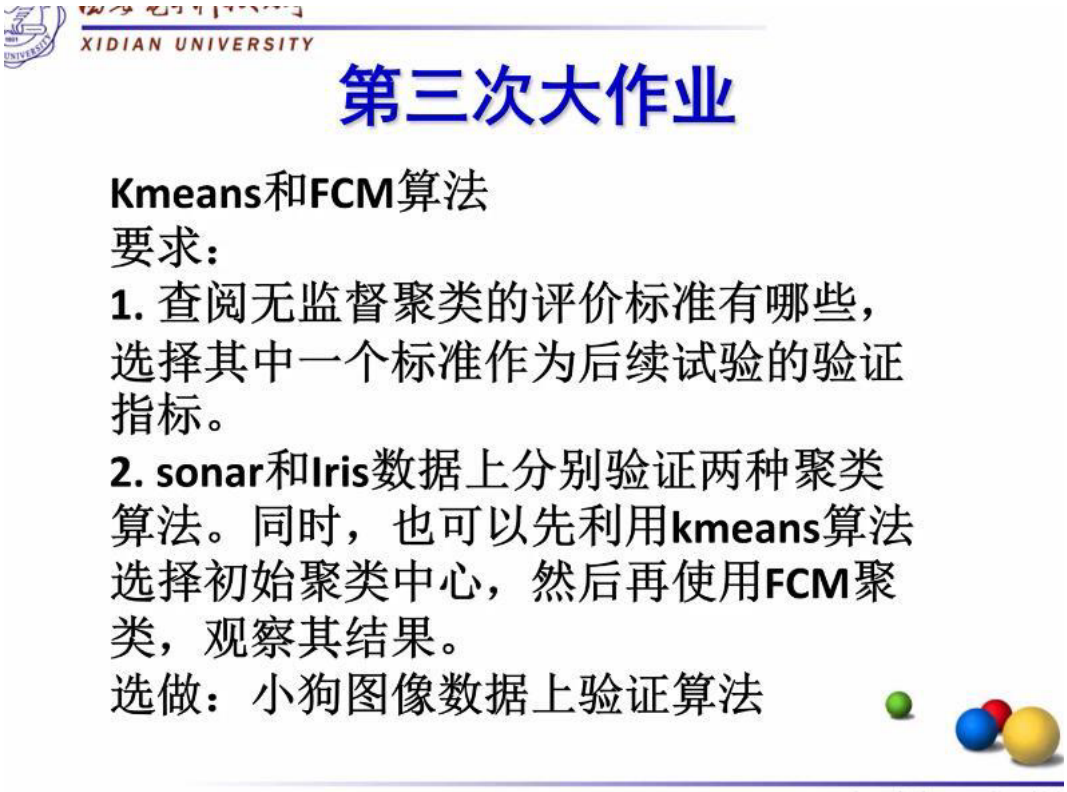
模式识别第三次大作业

姓名：游霄童 学号：21009200158

created: 2023/11/22

一、问题描述



二、数据集简述：

UCI_Iris数据集：

Iris数据集是一组用于分类问题的常用数据集,来自UCI机器学习库。Iris数据集包含了3种类型鸢尾花的4个特征信息。Iris数据集特征数量少,样本 数量适中,适合作为分类算法的「Hello World」示例。

```
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 2 2]
```

```
[[5.1 3.5 1.4 0.2]
 [4.9 3.  1.4 0.2]
 [4.7 3.2 1.3 0.2]
 [4.6 3.1 1.5 0.2]
 [5.  3.6 1.4 0.2]
 [5.4 3.9 1.7 0.4]
 [4.6 3.4 1.4 0.3]
 [5.  3.4 1.5 0.2]
 [4.4 2.9 1.4 0.2]
```

基本信息

- 样本数量:150个
- 类别:3类
    - Iris Setosa(山鸢尾)
    - Iris Versicolour(杂色鸢尾)
    - Iris Virginica(维吉尼亚鸢尾)
- 每类样本数量:50个
- 特征数:4个
    - 花萼长度
    - 花萼宽度
    - 花瓣长度
    - 花瓣宽度

数据描述

Iris数据集包含了3种类型鸢尾花的4个特征信息。这是一个多分类问题,常被用来测试分类算法的性能。数据集中的每个样本都属于其中一类,类别是均衡的,每个类型50个样本。4个特征表示花萼和花瓣的长度和宽度,都是正实数。

使用场景

Iris数据集特征数量少,样本数量适中,适合作为分类算法的「Hello World」示例。常用来测试分类算法的效果,如KNN、SVM、决策树等。也可用来比较不同算法的分类性能。由于样本量小,可视化分类结果。

UCI _Sonar数据集

> *UCI 机器学习库（UCI Machine Learning Repository）是一个广泛用于机器学习和数据挖掘研究的资源，其中包含了许多开源数据集，Sonar数据集就是其中之一。Sonar数据集（也称为声纳数据集）是一个经典的二分类问题数据集，用于声纳信号处理和目标检测领域的研究。*
>
> *网址：https://archive.ics.uci.edu/ml/machine-learning-databases/undocumented/connectionist-bench/sonar/sonar.all-data*

	attribute_1	attribute_2	...	attribute_58	attribute_59	attribute_60	Class
1	0.02	0.0371	...	0.0084	0.009	0.0032	Rock
2	0.0453	0.0523	...	0.0049	0.0052	0.0044	Rock
3	0.0262	0.0582	...	0.0164	0.0095	0.0078	Rock
4	0.01	0.0171	...	0.0044	0.004	0.0117	Rock
...	...	...	...	...	...	...	...
208	0.026	0.0363	...	0.0036	0.0061	0.0115	Mine

基本信息

- 样本数量:208
- 类别:2类
    - Rock(矿石)
    - Metal(金属)
- 特征数:60
- 特征信息:声纳返回信号在60个频带中的能量强度

数据描述

这是一个二分类问题,根据声纳返回信号区分矿石和金属。数据来自实际的声纳信号采集。相对Iris数据集,它有更多的特征数,样本不均衡,是一个更难的二分类问题。

使用场景

Sonar数据集可以用来测试二分类算法在高维特征上的表现。样本量适中,可用于比较不同二分类算法的效果。也可研究特征工程技术在高维数据上的应用。因为样本不均衡,还可研究分类算法处理样本不均衡的技巧。

三、无监督聚类的评价标准

1、误差平方和 (SSE - Sum of Squares of Errors):

定义：
在聚类分析中，误差平方和（SSE）是一种度量聚类效果的指标。它表示每个数据点到其簇内中心的距离的平方和。

计算方法：

1. 对于每个簇，计算该簇内每个数据点到簇中心的距离的平方。
2. 将所有簇的这些平方距离相加，得到总的误差平方和。

数学表达式：

$$SSE = \sum_{i=1}^{k} \sum_{j=1}^{n_i} \|x_{ij} - c_i\|^2$$

解释：
SSE越小，表示数据点越接近它们所属的簇中心，聚类效果越好。但SSE也有一个问题，即它会随着簇的数量增加而减小，因此对于不同簇数的聚类结果的比较可能并不直观。在实际应用中，SSE通常用于帮助确定合适的簇的数量。

2、轮廓系数（Silhouette Coefficient）：

定义：
轮廓系数是一种用于衡量聚类结果质量的指标，考虑了簇内的紧密度和簇间的分离度。

计算方法：

1. 对于每个数据点，计算它与同簇内其他点的平均距离（a）以及与最近其他簇内点的平均距离（b）。
2. 对于每个数据点，轮廓系数计算为

$$\frac{b - a}{\max(a, b)}$$

。

3. 对所有数据点的轮廓系数求平均，得到整个数据集的轮廓系数。

数学表达式：

$$\text{轮廓系数}(i) = \frac{b_i - a_i}{\max(a_i, b_i)}$$

$$\text{轮廓系数} = \frac{\sum_{i=1}^{n} \text{轮廓系数}(i)}{n}$$

其中，

- (a_i) 是数据点 (i) 与同簇内其他点的平均距离，
- (b_i) 是数据点 (i) 与最近其他簇内点的平均距离。

解释：
轮廓系数范围在 -1 到 1 之间，越接近1表示聚类结果越合理，越接近-1表示聚类结果越不合理。轮廓系数对于评估聚类效果的优劣性是一种有效的指标，特别是当真实的簇结构未知时。

3、 Calinski Harabasz Index

将容量为 N 的数据集合 X 聚成 K 类，通过计算类内各点与类中心的距离平方和来度量类内的紧密度（类内距离），各个类中心点与数据集中心点距离平方和来度量数据集的分离度（类间距离）。

$$cal = \frac{tr(B_K)(N-K)}{tr(W_K)(K-1)}$$

这里 B_K 为类间的协方差矩阵， W_K 类内数据的协方差矩阵，表示为

$$W_K = \sum_{k=1}^{K} \sum_{x \in C_k} (x - c_k)(x - c_k)^T$$
$$B_K = \sum_{k=1}^{K} n_k (c_k - c_X)(c_k - c_X)^T$$

C_k 表示以为 c_k 为中心的簇集合， n_k 表示集合 C_k 中的样本数，c_X 表示数据集中心， tr 表示矩阵的迹。

4、兰德系数&调整兰德指数

兰德系数（Rand Index）

是一种用于衡量两个聚类结果之间的相似性的指标。兰德系数的计算基于四个值:
真正例（$TruePositives$，TP）、真负例（$TrueNegatives$，TN）、假正例（$FalsePositives$，FP）和假负例（$FalseN$

在以下的描述中，我们假设有两个聚类结果，一个是“真实”聚类结果，另一个是某个聚类算法的结果。

定义:

- TP（$TruePositives$）： 表示在真实聚类结果中属于同一簇且在聚类算法的结果中也属于同一簇的数据对数。
- TN（$TrueNegatives$）： 表示在真实聚类结果中不属于同一簇且在聚类算法的结果中也不属于同一簇的数据对数。
- FP（$FalsePositives$）： 表示在真实聚类结果中不属于同一簇但在聚类算法的结果中属于同一簇的数据对数。
- FN（$FalseNegatives$）： 表示在真实聚类结果中属于同一簇但在聚类算法的结果中不属于同一簇的数据对数。

兰德系数的计算公式如下:

$$textRandIndex = \frac{TP + TN}{TP + FP + FN + TN}$$

解释:

- **兰德系数取值范围：** 0 到 1 之间。值越接近 1，表示两个聚类结果越相似。
- **随机兰德系数：** 如果完全随机地进行聚类，兰德系数的期望值为 0。所以，一个兰德系数大于 0 的聚类结果相对于完全随机的情况更好。

兰德系数是一种用于比较两个聚类结果的相似性的常用指标。它对簇的数量不敏感，但对于聚类结果的大小和形状有一定的影响。

调整兰德指数（Adjusted Rand Index，ARI）

是一种用于衡量两个聚类结果之间相似性的指标。它是兰德指数（Rand Index）的一个调整版本，用于解决兰德指数对于随机聚类的敏感性问题。

对于两个聚类结果，ARI将它们的相似性评分标准化，使其值介于 -1 和 1 之间。ARI 的计算基于以下四个值:

- **a（同簇内的样本对数）：**
  - a 是两个聚类结果中，被同时分到相同簇的样本对数。也就是说，这是同时在两个聚类结果中都被正确划分到同一个簇的样本对数。
- **b（不同簇内的样本对数）：**
  - b 是两个聚类结果中，被分到不同簇的样本对数。也就是说，这是在一个聚类结果中的样本在另一个聚类结果中被错误划分到不同簇的样本对数。
- **c（第一个聚类结果中同簇内的样本对数）：**
  - c 是在第一个聚类结果中，同簇内的样本对数。即第一个聚类结果中同一个簇内的所有样本两两之间的组合数。
- **d（第二个聚类结果中同簇内的样本对数）：**
  - d 是在第二个聚类结果中，同簇内的样本对数。即第二个聚类结果中同一个簇内的所有样本两两之间的组合数。

ARI 的计算公式如下:

$$ARI = \frac{\mathrm{a} + \mathrm{b} - \mathrm{random\_index}}{\mathrm{a} + \mathrm{b} + \mathrm{c} + \mathrm{d} - \mathrm{random\_index}}$$

其中，

$$\text{random\_index}$$

表示随机情况下的兰德指数。

ARI 的取值范围在 -1 到 1 之间：

- ARI 接近 1 表示两个聚类结果高度一致；
- ARI 接近 0 表示聚类结果之间的相似性与随机聚类没有太大差异；
- ARI 接近 -1 表示两个聚类结果之间的差异大于随机聚类。

ARI 是一种调整后的指标，更适合于评估聚类的性能，尤其是在聚类结果中簇的数量不同的情况下。

异同点：

相同点：

- 都用于评估聚类结果的相似性。
- 兰德系数和调整兰德系数都在不同聚类结果之间进行比较。

不同点：

- **对簇数和样本量的处理：**
  - 兰德系数对于不同的簇数和样本量比较敏感，因此在比较不同算法或者不同数据集时，可能不够公平。
  - 调整兰德系数通过对随机情况下的期望值进行修正，使得对于不同簇数和样本量更加鲁棒。
- **取值范围：**
  - 兰德系数的取值范围是 0 到 1。
  - 调整兰德系数的取值范围是 -1 到 1。

在实际应用中，通常更倾向于使用调整兰德系数，因为它对于簇的数量和样本量的变化更具有稳健性，能够更好地反映聚类结果的相似性。

四、算法介绍

K-Means 聚类算法：

目标：
K-Means 算法的目标是将数据集划分为 K 个簇，使得每个数据点到其所属簇的中心的距离平方和最小。

步骤：

1. **初始化：** 随机选择 K 个数据点作为初始簇中心。
2. **分配：** 将每个数据点分配到离它最近的簇中心。
3. **更新中心：** 对每个簇，计算其所有成员的均值，作为新的簇中心。
4. **迭代：** 重复步骤2和步骤3，直到簇中心不再改变或者达到最大迭代次数。

数学公式：

- **簇中心 c_i 的更新公式：**

$$c_i = \frac{1}{|S_i|} \sum_{x \in S_i} x$$

其中，S_i是第i个簇的成员集合，c_i是第i个簇的中心。

- **数据点x_j到簇中心c_i的距离平方：**

$$\|x_j - c_i\|^2$$

- **目标函数（损失函数）：**

$$J = \sum_{i=1}^{K} \sum_{j=1}^{n_i} \|x_{ij} - c_i\|^2$$

其中，K是簇的数量，n_i是第i个簇的成员数量。

模糊C均值（FCM）聚类算法：

目标：
FCM 算法同样旨在将数据集划分为 K 个簇，但相比于 K-Means，FCM 使用了模糊集合的概念，允许数据点属于多个簇的可能性。

步骤：

1. **初始化：** 随机初始化每个数据点属于每个簇的隶属度。

2. **更新隶属度：** 根据数据点与簇中心的距离重新计算隶属度。
3. **更新中心：** 根据重新计算的隶属度，更新每个簇的中心。
4. **迭代：** 重复步骤2和步骤3，直到隶属度和簇中心稳定或者达到最大迭代次数。

数学公式：

- 隶属度u_{ij}的更新公式：

$$u_{ij} = \frac{1}{\sum_{k=1}^{K} \left(\frac{\|x_j - c_i\|}{\|x_j - c_k\|} \right)^{\frac{2}{m-1}}}$$

  其中，m是模糊度指数，c_i是第i个簇的中心。

- 目标函数（损失函数）：

$$J = \sum_{i=1}^{K} \sum_{j=1}^{n} u_{ij}^{m} \|x_j - c_i\|^2$$

  其中，K是簇的数量，n是数据点的数量，u_{ij}是第i个簇对数据点j的隶属度。

五、结果分析

1、K-Means

```
from sklearn.metrics import adjusted_rand_score
ari = adjusted_rand_score(labels, res)
print('Adjusted Rand Index (ARI) in iris: ', ari)
```
在 2023.11.22 19:10:00 于 57ms内执行

```
Adjusted Rand Index (ARI) in iris:  0.7302382722834697
```

```
print('iris-SC指标: ' + str(metrics.silhouette_score(data, res, metric='euclidean')))
```
在 2023.11.22 19:10:00 于 42ms内执行

```
iris-SC指标: 0.5528190123564095
```

```
ari2 = adjusted_rand_score(labels2, res2)
print('Adjusted Rand Index (ARI) in sonar: ', ari2)
```
在 2023.11.22 19:10:00 于 64ms内执行

```
Adjusted Rand Index (ARI) in sonar:  0.010869872549936507
```

```
# 计算并打印 SC（轮廓系数）指标，使用欧氏距离作为度量方式
print('sonar-SC指标: ' + str(metrics.silhouette_score(data2, res2, metric='euclidean')))
```
在 2023.11.22 19:10:00 于 46ms内执行

```
sonar-SC指标: 0.19896316739419573
```

2、FCM

```
ari = adjusted_rand_score(labels, res)
print('Adjusted Rand Index (ARI) in iris: ', ari)
```
在 2023.11.22 19:09:38 于 56ms内执行

```
Adjusted Rand Index (ARI) in iris:  0.7683058726537342
```

```
print('iris-SC指标: ' + str(metrics.silhouette_score(data, res, metric='euclidean')))
```
在 2023.11.22 19:09:38 于 39ms内执行

```
iris-SC指标: 0.5209590632921908
```

```
    ari2 = adjusted_rand_score(labels2, res2)
    print('Adjusted Rand Index (ARI) in sonar: ', ari2)
```

在 2023.11.22 19:09:38 于 86ms内执行

```
Adjusted Rand Index (ARI) in sonar:  0.008545699314580773
```

```
    print('sonar-SC指标: ' + str(metrics.silhouette_score(data2, res2, metric='euclidean')))
```

在 2023.11.22 19:09:38 于 72ms内执行

```
sonar-SC指标: 0.19554404245764484
```

结果分析：

首先，针对 Iris 数据集和 Sonar 数据集，K-Means 和 FCM 算法都表现出了一般的聚类效果，尽管在 Iris 数据集上表现相对较好。以下是对这一结果的分析：

1. Iris 数据集的较好表现原因：

- **简单性和分布形状：** Iris 数据集相对简单，特征数较少，而且更接近于球形分布。这使得使用欧氏距离等度量是合理的。
- **前期模式识别表现：** 先前对 Iris 数据集进行的模式识别任务取得了较好的结果，这可能反映了数据集的相对易处理性。

2. Sonar 数据集的较差表现原因：

- **复杂性和噪声：** Sonar 数据集相较于 Iris 数据集更为复杂，且包含大量噪声和孤立点。这增加了聚类的难度，尤其是对于 K-Means 和 FCM 这样的传统聚类算法。
- **特征数量：** Sonar 数据集的特征数量相对较多，这增加了数据集的维度，使得欧氏距离等度量可能无法有效地捕捉数据间的相似性。
- **与球形分布不符：** 数据集的形状可能与球形分布的假设相去甚远，这导致 K-Means 和 FCM 在该数据集上的聚类效果不理想。
- **ARI 结果较差：** 调整兰德系数（ARI）的结果也不理想，这表明聚类结果与真实标签的一致性相对较差。

总体评价：

- 对于简单、相对规则的数据集，K-Means 和 FCM 可能表现较好。
- 对于复杂、包含噪声或异类的数据集，传统聚类算法的表现可能受到限制，考虑使用更复杂的聚类算法或特征工程来改进效果。