电子科技大学

计算机科学与工程学院

实 验 指 导 书

（实验）课程名称 人工智能综合实验 I

电子科技大学教务处制表

实验三　利用 K-均值聚类算法对未标注数据分组

一、　实验目的和任务

学习并掌握 K-均值聚类算法的算法原理和代码实现，学会使用该算法对未标注数据进行聚类。

二、　实验原理

K-均值聚类是机器学习算法两大分支之一的无监督学习中的重点算法，使用该算法对未标注数据进行聚类应用很广泛。

2.1 算法原理

1. 算法介绍

聚类是一种无监督学习，它将相似的对象归到同一个簇中，簇中的对象越相似，聚类的效果越好。聚类的目标是簇内数据相似度高，簇间拉大距离。K-means 是聚类算法中的一种，能够将样本按照距离远近划分成不同个簇。其中 K 是指簇的数量，means 是指均值。

K-means 算法的工作流程是这样的。首先，随机确定 k 个初始点作为质心，然后将数据集中的每个点分配到一个簇中，具体来讲，为每个点距离其最近的质心，将其分配给该质心所对应的簇。之后每个簇的质心更新为该簇所有点的平均值。之后不断重复以上流程，直到质心位置不再改变（样本簇的分配不再改变）。

K-means 算法：

①随机选取 k 个样本作为初始的聚类中心

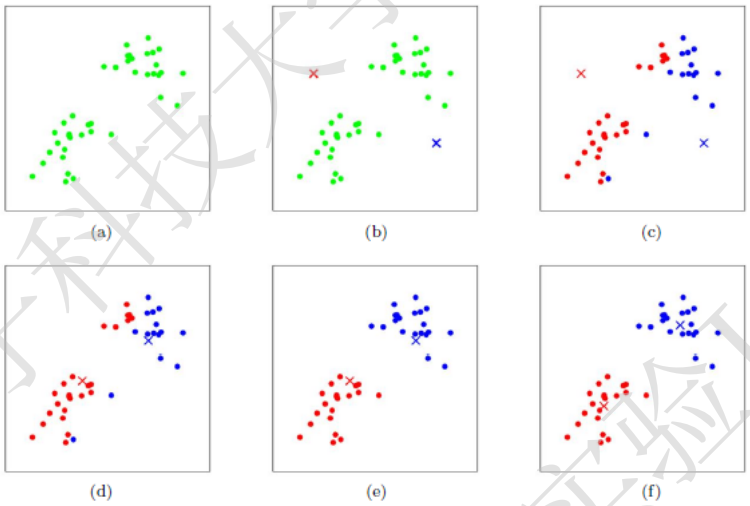
②计算每个样本到各个聚类中心之间的距离，将每个样本分配给

距离它最近的聚类中心，此时全部样本已划分为 k 组

③更新聚类中心，将每组中样本的均值作为该组新的聚类中心；

④重复进行第二、三步，直到聚类中心趋于稳定，或者到达最大迭代次数。

2. 图解



上图是 k-means 聚类算法的简单图解。假设 k=2，在图 b 中，我们随机选择了两个类别质心，即图中的红色质心和蓝色质心，然后分别求样本中所有点到这两个质心的距离，并标记每个样本的类别为和该样本距离最小的质心的类别，如图 c 所示，我们得到了所有样本点的第一轮迭代后的类别。此时对当前标记为红色和蓝色的点分别求其新的质心，如图 d 所示，新的红色质心和蓝色质心的位置已经发生了变动。图 e 和图 f 重复了我们在图 c 和图 d 的过程，即将所有点的类别标记为距离最近的质心的类别并更新质心。经过多次迭代后，最终得到的两个类别如图 f 所示。

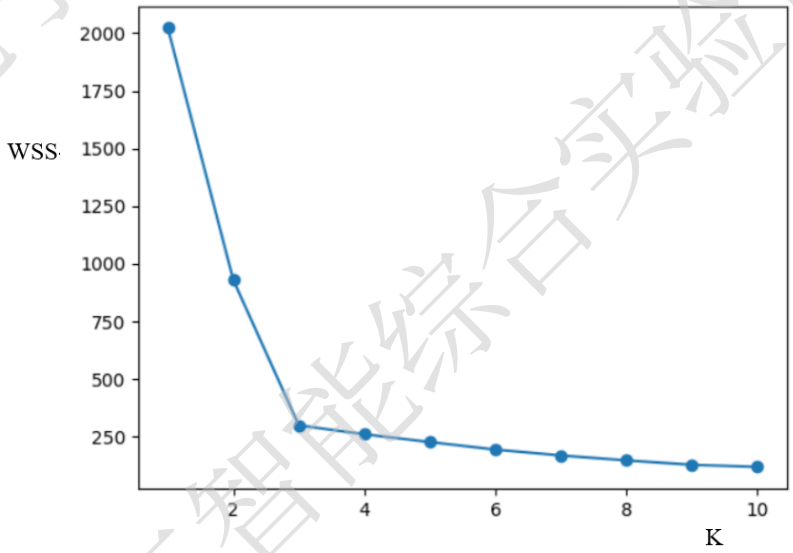
3.寻找最优 K 值

如何寻找最优 K 值呢？

簇内误差平方和是衡量聚类效果的常用方法，是指所有样本到对应质心的距离的平方和，如下面的公式，其中 k 为簇数，Ck 为第 k 簇的质心，p 是样本。

$$WSS(K) = \sum_{k=1}^{K} \sum_{p \in C_k} |p - c_k|^2.$$

那什么时候 k 值最优呢？显然不是 WSS 最小的时候，因为质心越多，WSS 越少，质心和样本数量相同时，WSS 能达到零，我们显然不需要这么多质心。因此有一种方法 Elbow method（手肘法），随着 K 值的变大，WSS 值下降的幅度越来越小，WSS 下降开始不明显的那个点，就是我们需要的 K 值。下面是 WSS-K 的一般性关系。



可以看到随着 K 值的增大，WSS 值在变小，当在 K=3 时，出现了明显的偏折，好像是人的肘部，此时的 K 即为最优值。

2.2 算法 demo

```
def loadDataSet(fileName):          #general function to parse tab -delimited floats
    dataMat = []                    #assume last column is target value
    fr = open(fileName)
    for line in fr.readlines():
        curLine = line.strip().split('\t')
        fltLine = list(map(float,curLine)) #map all elements to float()
        dataMat.append(fltLine)
    return dataMat
```

可参考的距离计算 demo，可以根据具体任务选择使用其他距离度量函数

```
def distEclud(vecA, vecB):
    return sqrt(sum(power(vecA - vecB, 2))) #la.norm(vecA-vecB)
```

下面这个函数是生成随机质心的 demo。完成 K-means 聚类方法中最开始的生成初始化质心的任务。注意随机生成质心，质心坐标要求在整个数据集的边界之内，下面这个函数就保证了随机点在整个数据集的边界之内。

```
def randCent(dataSet, k):
    n = shape(dataSet)[1]
    centroids = mat(zeros((k,n)))#create centroid mat
    for j in range(n):#create random cluster centers, within bounds of each dimension
        minJ = min(dataSet[:,j])
        rangeJ = float(max(dataSet[:,j]) - minJ)
        centroids[:,j] = mat(minJ + rangeJ * random.rand(k,1))
    return centroids
```

K-means 算法由大家完成

```
def kMeans(dataSet, k):
    # **********************
    return centroids, clusterAssment    # 返回质心  聚类结果
```

三、 实验内容

编程实现 K-均值聚类计算代码，并在西瓜数据集 3.0α 上实现聚类任务。

验证　（1）K 不同取值时聚类实验，分析实验结果。

（2）利用手肘法，绘制 K-WSS 的关系折线图，根据关系图指出最优的 K 值。（画图可参考 matplotlib.pyplot 库）

表 4.5　西瓜数据集 3.0α

编号	密度	含糖率	好瓜
1	0.697	0.460	是
2	0.774	0.376	是
3	0.634	0.264	是
4	0.608	0.318	是
5	0.556	0.215	是
6	0.403	0.237	是
7	0.481	0.149	是
8	0.437	0.211	是
9	0.666	0.091	否
10	0.243	0.267	否
11	0.245	0.057	否
12	0.343	0.099	否
13	0.639	0.161	否
14	0.657	0.198	否
15	0.360	0.370	否
16	0.593	0.042	否
17	0.719	0.103	否

四、　实验报告要求

根据实验要求完成实验内容，要求有实验代码，并给出每小问的实验结果、讨论分析。

五、　实验仪器设备

机房电脑一台，编程平台为 Anaconda 下 Spyder 编辑器。