Big Data Analysis -- Homework 5

201250182 郑义

一、方法理解

本次分类采用的是 KMeans(SimpleKMeans)的方法，数据集为 2（iris.arff），处理顺序为：

1. 对数据进行预处理
2. 设置类别数量（cluster numbers）
3. 使用数据进行聚类
4. 查看结果

对 KMeans 聚类方法的理解

1. 什么是聚类：聚类是一种无监督的机器学习方法（可以把它看成无监督的分类问题，不需要通过打好标签的数据进行模型训练），通过找到数据中的 label 将数据分为不同的类别，这些数据一般都有着一定的“相似性”，因此会被分在同一类中，这种行为称为聚类。
2. Kmeans 算法思想：Kmeans 算法是一种非常经典的聚类算法，其核心思想就是通过计算数据到类别中心点的距离来判断该数据究竟是属于哪一个类别，在 Kmeans 中，类别的个数（cluster number）是需要人工指定的（一般可以通过肘部法则得出较佳的类别个数）。
3. Kmeans 算法步骤：
   a) 指定类别个数
   b) 随机初始化类别的中心点
   c) 计算各个数据到这些中心点的距离，距离哪个中心点较近则该数据归属到哪个类别中
   d) 重新计算类别的中心点，方法是通过计算当前 batch 之后该类别中的数据的重心
   e) 重复 c、d 步骤，直到中心点不再发生变化，此时聚类结束

二、数据集处理思路

对于聚类来说，较为合适的数据预处理办法为规范化（Normalize），下面代码展示了使用 weka 来进行规范化的步骤：

三、实验结果

1. 聚类方法代码：

```java
/**
 * @author zhengyi
 */
2 usages  ZhengYi
public class ClusterServiceImpl implements ClusterService {

    2 usages
    private Instances data;
    5 usages
    private final SimpleKMeans simpleKMeans;
    3 usages
    private int clusterNum;

    1 usage  ZhengYi
    public ClusterServiceImpl() { simpleKMeans = new SimpleKMeans(); }

    1 usage  ZhengYi
    @Override
    public void setTrainSetAndTestSet(String fileName) throws Exception {
        Instances instances = DataSetUtil.loadDataSet(fileName);
        data = DataPreProcessingUtil.preProcessingData(instances);
    }

    1 usage  ZhengYi
    @Override
    public void setNumClusters(int num) throws Exception {
        clusterNum = num;
        simpleKMeans.setNumClusters(clusterNum);
    }

    1 usage  ZhengYi
    @Override
    public void trainModel() throws Exception {
        // Build Classifier
        simpleKMeans.buildClusterer(data);
    }
```

2. 运行结果

```
Cluster 0 size: 50.0 Centroid: 0.454444,0.320833,0.552542,0.510833,Iris-versicolor
Cluster 1 size: 50.0 Centroid: 0.196111,0.590833,0.078644,0.06,Iris-setosa
Cluster 2 size: 50.0 Centroid: 0.635556,0.405833,0.771525,0.8025,Iris-virginica
```

3. 结果评估：运行结果的 cluster size 和数据集中的分类是相同的，可以看出这次聚类是成功的。