**北京邮电大学软件学院**

**2016－2017学年第一学期实验报告**

**课程名称： 数据挖掘**

**项目名称： K均值算法**

**项目完成人：**

**姓名：\_\_肖逸敏\_\_\_\_\_\_学号：\_\_\_2014211990\_\_\_\_\_**

**姓名：\_\_\_\_\_\_\_\_学号：\_\_\_\_\_\_\_\_**

**姓名：\_\_\_\_\_\_\_\_学号：\_\_\_\_\_\_\_\_**

**姓名：\_\_\_\_\_\_\_\_学号：\_\_\_\_\_\_\_\_**

**姓名：\_\_\_\_\_\_\_\_学号：\_\_\_\_\_\_\_\_**

**指导教师：\_\_\_\_\_\_\_\_牛琨\_\_\_\_\_\_\_\_\_\_**

**日 期： 2016 年 12 月 21 日**

1. **实验目的**

通过利用Weka中提供的simpleKmeans方法对数据进行聚类分析，更深刻的理解k均值算法，并通过对实验结果进行观察分析，找出实验中所存在的问题。

1. **实验内容**
2. 采用K均值算法对给定数据集合进行聚类，给出聚类结果，并与真实数据分布对比，判断聚类准确度。改变K的数值，研究K值改变与聚类准确度之间的关系。改变初始簇心，研究簇心变化与聚类准确度之间的关系。
3. 请在实验报告中给出你的模型结果，并给出解释。
4. **实验环境**

weka

1. **实验结果**

## 格式转换：

1. 打开“data.xls” 另存为CSV类型，得到“data.csv”。

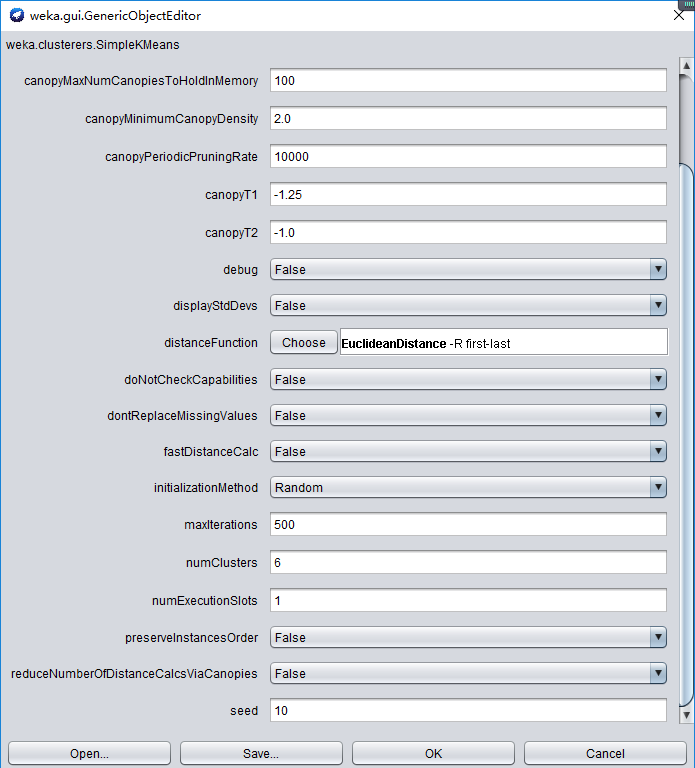


1. 在WEKA中提供了一个“Arff Viewer”模块，打开一个“data.csv”进行浏览，然后另存为ARFF文件，得到“data.arff”

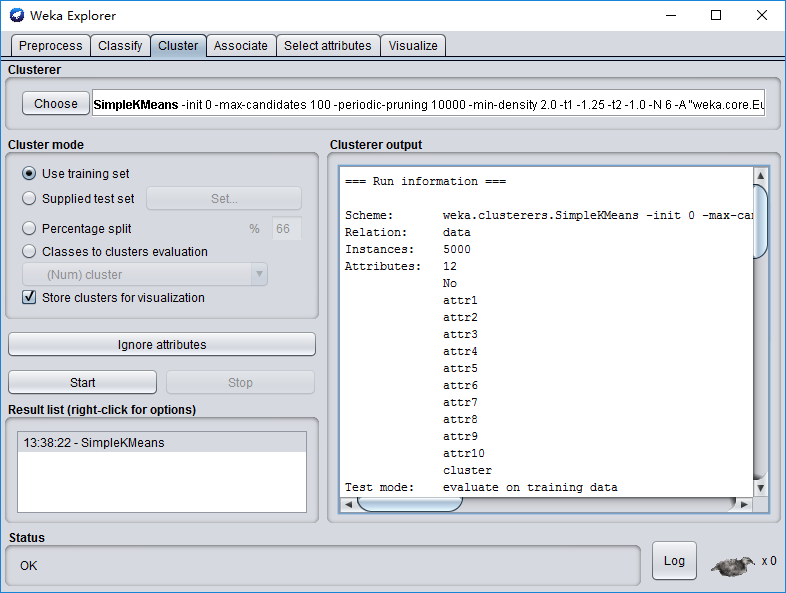


### 实验过程：

1. 用“Explorer”打开刚才得到的“data.arff”，并切换到“Cluster”。点“Choose”按钮选择“SimpleKMeans”，这是WEKA中实现K均值的算法。点击旁边的文本框，修改“numClusters”为6，说明我们希望把这些实例聚成6类，即K=6。下面的“seed”参数是要设置一个随机种子，依此产生一个随机数，用来得到K均值算法中第一次给出的K个簇中心的位置。我们不妨暂时让它就为10。



选中“ClusterMode”的“Usetrainingset”，点击“Start”按钮，观察右边“Clustereroutput”给出的聚类结果如下：



右侧结果为：

=== Run information ===

Scheme: weka.clusterers.SimpleKMeans -init 0 -max-candidates 100 -periodic-pruning 10000 -min-density 2.0 -t1 -1.25 -t2 -1.0 -N 6 -A "weka.core.EuclideanDistance -R first-last" -I 500 -num-slots 1 -S 10

Relation: data

Instances: 5000

Attributes: 12

No

attr1

attr2

attr3

attr4

attr5

attr6

attr7

attr8

attr9

attr10

cluster

Test mode: evaluate on training data

=== Clustering model (full training set) ===

kMeans

======

Number of iterations: 6

Within cluster sum of squared errors: 1914.5761410118812

Initial starting points (random):

Cluster 0: 2114,89,31,98,68,92,96,26,67,62,19,3

Cluster 1: 3817,93,100,103,121,43,89,45,37,45,26,9

Cluster 2: 3832,72,80,116,107,51,121,73,38,37,87,8

Cluster 3: 644,73,86,115,115,60,110,64,61,28,82,8

Cluster 4: 2035,65,73,114,102,59,104,63,37,40,97,8

Cluster 5: 1352,77,67,92,113,50,107,59,34,33,86,8

Missing values globally replaced with mean/mode

Final cluster centroids:

Cluster#

Attribute Full Data 0 1 2 3 4 5

(5000.0) (2497.0) (500.0) (500.0) (503.0) (500.0) (500.0)

=================================================================================

No 2500.5 2501.7761 2494.862 2488.126 2568.3976 2514.756 2429.578

attr1 70.2764 75.1774 89.336 65.974 39.4513 25.42 106.91

attr2 62.4154 32.9624 95.956 73.956 73.2763 111.298 104.614

attr3 74.4678 76.6296 107.356 101.414 104.3519 14.632 33.61

attr4 58.653 32.6468 114.478 110.254 94.5229 72.42 31.25

attr5 57.7228 65.9491 34.672 55.04 22.1491 50.63 85.254

attr6 80.194 75.8454 103.226 109.028 64.495 36.332 109.7

attr7 60.4024 52.2119 49.462 69.784 14.9125 102.012 107.018

attr8 63.5642 64.2827 46.24 47.946 107.8151 68.922 43.044

attr9 47.998 62.5471 33.22 39.63 63.8052 15.59 14.992

attr10 63.6096 63.9652 29.592 87.398 67.002 109.542 22.718

cluster 5.5 3.7986 9 8 6.9881 2 10

Time taken to build model (full training data) : 0.17 seconds

=== Model and evaluation on training set ===

Clustered Instances

0 2497 ( 50%)

1 500 ( 10%)

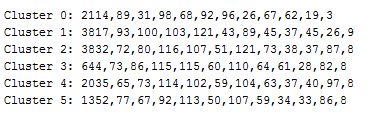
2 500 ( 10%)

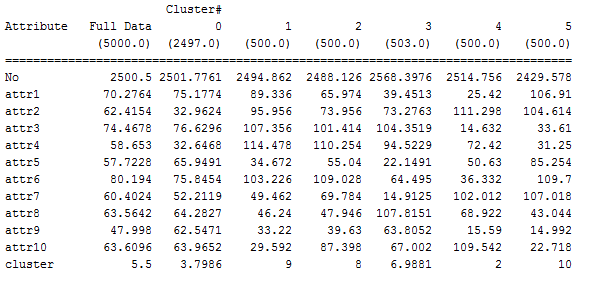
3 503 ( 10%)

4 500 ( 10%)

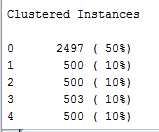
5 500 ( 10%)

下面对实验结果进行解释。观察图1，首先我们注意到结果中有这么一行：Withinclustersumofsquarederrors: 1914.5761410118812这是评价聚类好坏的标准，数值越小说明同一簇实例之间的距离越小。也许你得到的数值会不一样；实际上如果把“seed”参数改一下，得到的这个数值就可能会不一样。我们应该多尝试几个“seed”，并采纳这个数值最小的那个结果。例如我让“seed”取100，就得到：Withinclustersumofsquarederrors：1635.8546541601272

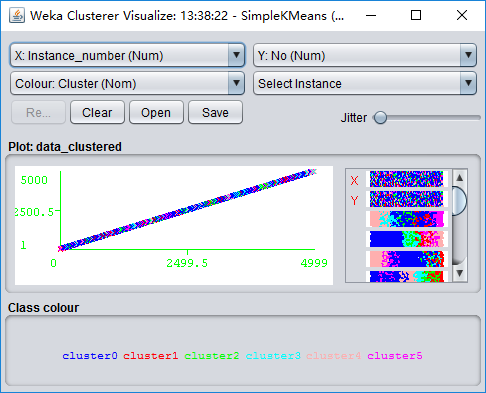
接下来“Clustercentroids:”之后列出了各个簇中心的位置。对于数值型的属性，簇中心就是它的均值（Mean），对于数值型的属性，还给出了它在各个簇里的标准差（StdDevs）。



最后的“ClusteredInstances”是各个簇中实例的数目及百分比。



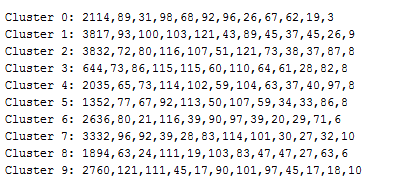
为了观察可视化的聚类结果，我们在左下方“Resultlist”列出的结果上右击，点“Visualizeclusterassignments”。弹出的窗口给出了各实例的散点图。最上方的两个框是选择横坐标和纵坐标，第二行的”color”是散点图着色的依据，默认是根据不同的簇“Cluster”给实例标上不同的颜色。

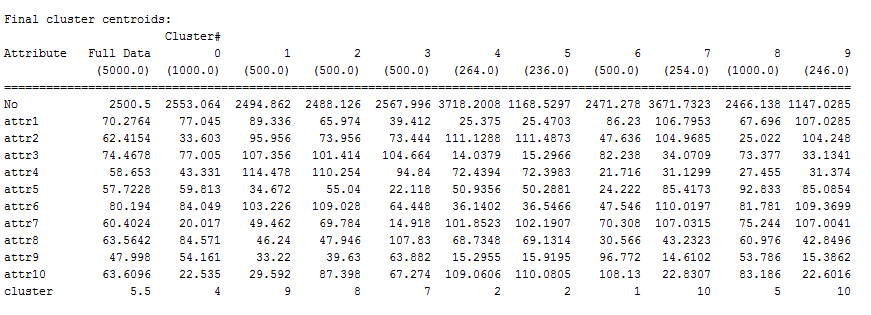


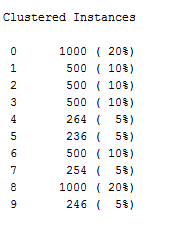
##### K值修改结果：

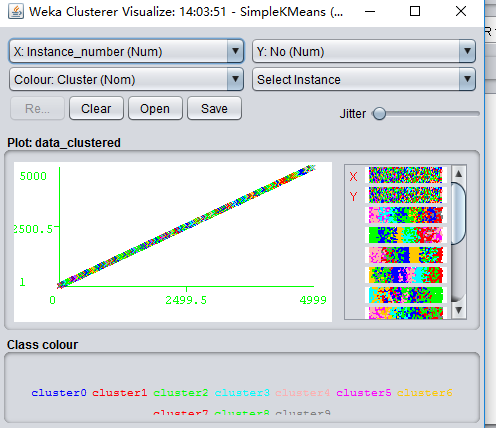
K=10：





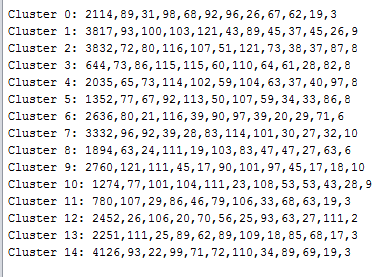


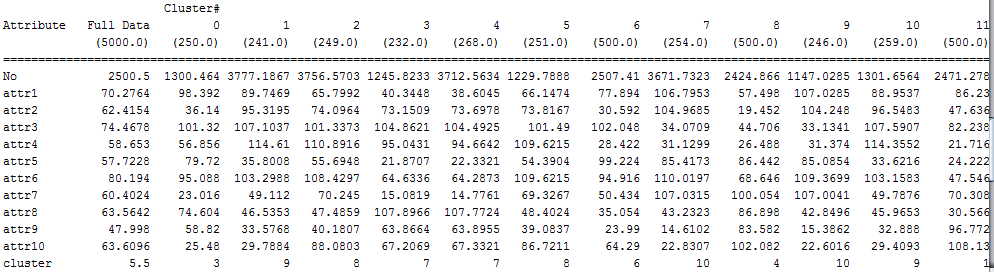


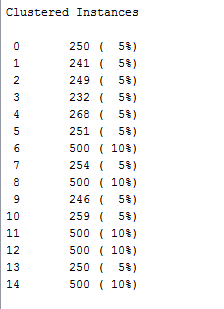


K=15：







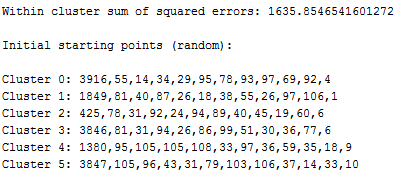


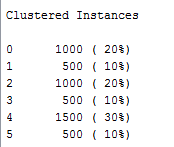
可以看出，聚类一项的均方值随着K的增大基本呈现为逐渐减小的趋势，即簇间平均距离趋于减小，单就此项来看，K值越大，准确度越低；

簇心修改结果：

Seed=100：







Seed：200：



