《数据挖掘》实验报告（三）

专　　业　__信息管理与信息系统_____

年　　级　_____2016_____

学　　号　____2016214288_____

学生姓名　_____曾德勤_____

指导老师　_____刘向_____

华中师范大学信息管理学院

I　实验目的和意义

数据挖掘设计实验的目的是培养学生具有初步的 python 程序设计、编程、调试的能力。通过实验使学生进一步熟悉并掌握 python 程序的调试运行环境、程序设计过程、程序的基本结构以及程序设计的基本方法。通过实验，使学生将程序设计的理论知识与实践相结合，为学生学习其他计算机编程语言打下基础。

II　实验内容

实验三　　K-均值

【实验意义】

本章学习的 K-均值聚类算法可以发现 k 个不同的簇，且每个簇的中心采用簇中所含值的均值计算而成。

【实验要求及步骤】

通过迭代寻找 k 个聚类的一种划分方案，使得用这 k 个聚类的均值来代表相应各类样本时所得的总体误差最小。

（1）收集数据：提供文本文件【testSet.txt】

（2）准备数据：解析 tab 键分隔的数据行。

（3）测试算法：使用本章的 kmeans()函数来把相似的（或距离近的）样本聚为同一类，而把不相似的（或距离远的）样本归在其他类。

【实验报告】

实习时间：　　实习地点：　　实习机号：

具体实验内容

（1）收集数据：提供文本文件【testSet.txt】

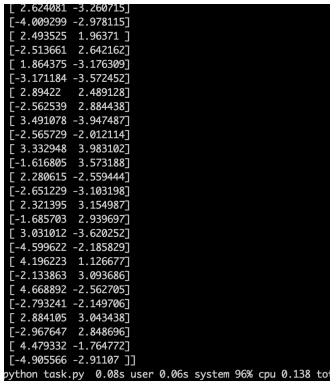
（2）准备数据：解析 tab 键分隔的数据行。

```
# coding=UTF-8
import kMeans
from numpy import *

# 从文本中构建矩阵
dataMat = mat(kMeans.loadDataSet('testSet.txt'))
print dataMat
```

```
Press ENTER or type command to continue
[[ 1.658985  4.285136]
 [-3.453687  3.424321]
 [ 4.838138 -1.151539]
 [-5.379713 -3.362104]
 [ 0.972564  2.924086]
 [-3.567919  1.531611]
 [ 0.450614 -3.302219]
 [-3.487105 -1.724432]
 [ 2.668759  1.594842]
 [-3.156485  3.191137]
 [ 3.165506 -3.999838]
 [-2.786837 -3.099354]
 [ 4.208187  2.984927]
 [-2.123337  2.943366]
 [ 0.704199 -0.479481]
 [-0.39237  -3.963704]
 [ 2.831667  1.574018]
 [-0.790153  3.343144]
 [ 2.943496 -3.357075]
 [-3.195883 -2.283926]
 [ 2.336445  2.875106]
 [-1.786345  2.554248]
 [ 2.190101 -1.90602 ]
 [-3.403367 -2.778288]
 [ 1.778124  3.880832]
 [-1.688346  2.230267]
 [ 2.592976 -2.054368]
 [-4.007257 -3.207066]
 [ 2.257734  3.387564]
 [-2.679011  0.785119]
 [ 0.939512 -4.023563]
 [-3.674424 -2.261084]
 [ 2.046259  2.735279]
 [-3.18947   1.780269]
 [ 4.372646 -0.822248]
 [-2.579316 -3.497576]
 [ 1.889034  5.1904  ]
 [-0.798747  2.185588]
 [ 2.83652  -2.658556]
 [-3.837877 -3.253815]
```

….



（3）测试算法：使用本章的 kmeans()函数来把相似的（或距离近的）样本聚为同一类，而把不相似的（或距离远的）样本归在其他类。

```
10 # 创建4个质心的聚类算法
11
12
13 c1, c2 = kMeans.kMeans(dataMat, 4)
14
15 print '四个质心的坐标: ', c1
16
17 print '每个点属于哪个类以及与质心的欧式距离', c2
18
```

```
Press ENTER or type command to continue
四个质心的坐标:  [[-3.53973889 -2.89384326]
 [-2.46154315  2.78737555]
 [ 2.65077367 -2.79019029]
 [ 2.6265299   3.10868015]]
```

```
每个点属于哪个类以及与质心的欧式距离 [[  3.00000000e+00   1.87526709e+01]
 [  1.00000000e+00   2.30407721e+00]
 [  2.00000000e+00   1.78710249e+01]
 [  0.00000000e+00   0.00000000e+00]
 [  3.00000000e+00   1.43931113e+01]
 [  1.00000000e+00   3.44569387e+00]
 [  2.00000000e+00   1.11511194e+00]
 [  0.00000000e+00   1.61100003e+01]
 [  3.00000000e+00   1.59040631e+01]
 [  1.00000000e+00   3.36312746e+00]
 [  2.00000000e+00   3.08320659e+00]
 [  0.00000000e+00   4.93255492e+00]
 [  3.00000000e+00   3.64375409e+01]
 [  1.00000000e+00   1.42590338e+00]
 [  2.00000000e+00   1.00642047e+01]
 [  2.00000000e+00   3.44966411e+00]
 [  3.00000000e+00   1.69602084e+01]
 [  1.00000000e+00   3.87631675e+00]
 [  2.00000000e+00   2.35335749e+00]
 [  0.00000000e+00   1.56660075e+01]
 [  3.00000000e+00   2.06289448e+01]
 [  1.00000000e+00   1.35263362e+00]
 [  2.00000000e+00   3.66508780e+00]
 [  0.00000000e+00   2.01088676e+01]
 [  3.00000000e+00   2.52019725e+01]
 [  1.00000000e+00   2.07510476e+00]
 [  2.00000000e+00   3.93003510e+00]
 [  0.00000000e+00   2.55784000e+01]
 [  3.00000000e+00   2.38247377e+01]
 [  1.00000000e+00   4.49454134e+00]
 [  2.00000000e+00   3.78222689e-01]
 [  0.00000000e+00   1.64431138e+01]
 [  3.00000000e+00   1.80089340e+01]
 [  1.00000000e+00   2.20490033e+00]
 [  2.00000000e+00   1.66909444e+01]
 [  0.00000000e+00   7.04039112e+00]
 [  3.00000000e+00   2.53714279e+01]
 [  1.00000000e+00   4.85071237e+00]
 [  2.00000000e+00   2.96559512e+00]
 [  0.00000000e+00   2.55273167e+01]
 [  3.00000000e+00   2.69604666e+01]
```

```
[  2.00000000e+00   1.56489945e+00]
[  0.00000000e+00   2.34356402e+01]
[  3.00000000e+00   1.63842698e+01]
[  1.00000000e+00   1.07303298e+00]
[  2.00000000e+00   4.20719169e-01]
[  0.00000000e+00   7.06283005e+00]
[  3.00000000e+00   2.19636171e+01]
[  1.00000000e+00   7.01885367e-01]
[  2.00000000e+00   4.27992839e+00]
[  0.00000000e+00   1.30703090e+01]
[  3.00000000e+00   3.63846754e+01]
[  1.00000000e+00   2.63933885e+00]
[  2.00000000e+00   1.93326060e+00]
[  0.00000000e+00   5.07648961e+00]
[  3.00000000e+00   2.24786627e+01]
[  1.00000000e+00   1.20840666e+00]
[  2.00000000e+00   2.52960059e+00]
[  0.00000000e+00   0.00000000e+00]
[  3.00000000e+00   2.69872390e+01]
[  1.00000000e+00   1.29117424e+00]
[  2.00000000e+00   1.16391576e+01]
[  0.00000000e+00   1.41932320e+01]
[  3.00000000e+00   2.54612423e+01]
[  1.00000000e+00   7.08256025e-01]
[  2.00000000e+00   1.28519126e+01]
[  0.00000000e+00   0.00000000e+00]]
python task.py  0.16s user 0.09s system 91% cpu 0.273 total
```

实习小结	我通过本次实验尝试了聚类分析当中比较简单的 k 均值算法：首先，选择 k 个初始质心（随机的），每个数据点指派到最近的质心，指派到一个质点的点集为一个簇，然后根据指派到簇的点，更新每个簇的质心，重复指派和更新步骤，直到质心近似不变。