**作业9.10**

1. **实验要求**

实现一种能自动确定聚类数的改进k均值算法，编程实现并在西瓜数据集上运行。

1. **实验原理**

如果单纯的从最小化类间均方误差来求解，当K的值为样本数，即每个样本分为一类时，得到均方误差为0，显然这是没有意义的。所以需要对k进行适当的惩罚。这里对分类后的混乱程度(熵)进行惩罚，取适当的系数t，使得



其中|Di|是第i类包含的样本数，|D|为样本总数，t为适当的参数，这里设为0.5。

由于熵为负数，所以在最小化作为惩罚时用减号。

然后从K=2起迭代，当求出的Ek增大时，认为k-1是最优的k值。

1. **实验步骤**
2. 初始化均方误差
3. 从2开始，对划分类进行遍历。
4. 对所有样本遍历，选择最近的集合。
5. 计算两次均值差异，并更新均值。
6. 如果ts比前一轮大则停止,否则更新。
7. 绘制凸包与点。

代码如下：

x = xlsread('w4.xlsx', 'sheet1', 'A1:B30');

[m,n]=size(x);

%当前最低的平方误差，初始值设为一个很大的数

old\_ts=100;

%对K值遍历，至少2类

for k=2:10

%随机均值

u=x(randperm(m,k),:);

while 1

%将各类集合清空

c=zeros(k,30);

nums=zeros(k,1);

%对所有样本遍历，选择最近的集合

for i=1:m

mind=100000;

minl=0;

for j=1:k

d=norm(x(i,:)-u(j,:));

if(d<mind)

mind=d;

minl=j;

end

end

nums(minl)=nums(minl)+1;

c(minl,nums(minl))=i;

end

%计算两次均值差异，并更新均值

ut=zeros(k,2);

for i=1:k

for j=1:nums(i)

ut(i,:)=ut(i,:)+x(c(i,j),:);

end

ut(i,:)=ut(i,:)/nums(i);

end

du=norm(ut-u);

if(du<0.001)

break;

else

u=ut;

end

end

%计算当前的均方误差

ts=0;

for i=1:k

for j=1:nums(i)

ts=ts+norm(x(c(i,j),:)-u(i,:))^2;

end

%惩罚项

ts=ts-(nums(i)/m)\*log(nums(i)/m)\*0.5;

end

%如果ts比前一轮大则停止,否则更新

if(ts<old\_ts)

old\_ts=ts;

old\_c=c;

old\_nums=nums;

else

break;

end

end

ch='o\*+>.';

%取前一轮的k为最佳的k值

nums=old\_nums;

c=old\_c;

k=k-1;

%绘制凸包与点

for i=1:k

plot(x(c(i,1:nums(i)),1),x(c(i,1:nums(i)),2),ch(i));

hold on;

tc=x(c(i,1:nums(i)),:);

chl=convhull(tc);

line(tc(chl,1),tc(chl,2))

hold on;

end

xlabel('密度');

ylabel('含糖率');

title('K-means');

1. **实验结果**

得出聚类的数目为3或4，图分别如下：



