基于层次的 K－means 初始化算法

汤九斌,陆建峰,唐振民,杨静宇
(南京理工大学计算机系,南京 210094)

[摘要] K－means 算法是一种常用的聚类算法,但是聚类中心的初始化是其中的一个难点。笔者提出了一个基于层次思想的初始化方法。一般聚类问题均可看作加权聚类,通过层层抽样减少数据量,然后采用自顶向下的方式,从抽样结束层到原始数据层,每层都进行聚类,其中每层初始聚类中心均通过对上层聚类中心进行换算得到,重复该过程直到原始数据层,可得原始数据层的初始聚类中心。模拟数据和真实数据的实验结果均显示基于层次抽样初始化的 K－means 算法不仅收敛速度快、聚类质量高,而且对噪声不敏感,其性能明显优于现有的相关算法。

[关键词] 层次技术; 初始聚类中心; 加权数据; K 平均聚类
[中图分类号] TP311 [文献标识码] A [文章编号] 10009－1742(2008)11－0074－06

1 引言

聚类分析在模式识别、数据挖掘等领域,起着非常重要的作用。近些年来,随着数据挖掘研究的深入,聚类分析日益受到重视[1]。K－means 算法是较常用的聚类算法之一,其中聚类中心的初始化对该算法的性能影响很大。若选取的初始聚类中心不合适,将不仅影响收敛速度,还有可能使算法陷入局部最优,但目前许多参考文献往往绕开该问题。迄今为止,对聚类算法中聚类中心初始化的研究相对较少,且没有简单通用的好方案,重复不同的随机选择似乎成为了事实上的方法[2]。Pena 等对目前常用的初始化算法从聚类质量(有效性)、聚类方法的敏感性(鲁棒性)和收敛速度(即效率)等方面进行了实验比较研究[3],结果表明随机初始化方法和 Kaufman 方法[4](KA)在有效性和鲁棒性方面均优于 Forgy 方法[5](FA)和 Macqueen 方法[6](MA),其中 KA 在收敛速度方面优于随机化方法。

最近, Bradly 和 Fayyad 等人对目前的一些初始点选择的方法进行了评述,并指出"在离散数据上采用 EM 算法和随机的初始化方法相比,并没有什么提高"[7,8]。在文献[7]中,其算法思想:首先对数据进行随机抽样,例如,抽取原数据的 1 %,然后在抽样后的数据上采用 EM 算法进行聚类,得到一组聚类中心,然后,重复上述过程(再重新抽样和进行聚类),假设重复 n 次,那么就得到 n 组聚类中心;然后对上述所有采样的数据,利用 n 组聚类中心采用 K－means 算法再进行聚类,选取性能最佳的一组,作为最终的初始聚类中心。从他们论文所提供的结果来看,该方法能够适用于大规模的数据,并且总体性能较好。

Khan 和 Ahmad 提出了一种基于属性的初始化方法[9]。其方法的主要思想是,数据点的每个属性可以为初始化提供信息,利用这一点,首先获得比类别数多的初始聚类中心,然后再进行合并,使得初始聚类中心的个数等于类别数。实验结果显示,该方法的性能优于随机化的初始化方法。

笔者提出了基于层次方法的聚类中心初始化方法(HIKM),通过分层聚类,找到较好的初始聚类中心。该算法采用误差平方和最小作为聚类测度,首先采用金字塔结构,对原始数据进行层层抽样,在抽样后的数据上进行聚类。尽管抽样后数据有所减少,但对 K－means 算法而言,原始信息却损失较少,抽样后的数据能较好反映抽样前的信息,同时能

[收稿日期] 2006－07－14; 修回日期 2006－09－15
[作者简介] 汤九斌(1969－),男,湖北黄石市人,南京理工大学博士生,主要研究领域为模式识别、数据挖掘、专家系统

够抑制噪声。经过一定次数的抽样后,数据量大大减少,聚类的收敛速度很快。由于上述聚类中心是基于抽样后数据的,而聚类算法所需的是原始数据的初始聚类中心,因此必须在抽样后所得初始聚类中心的基础上,再去求取原始数据的初始聚类中心。

2 基于层次初始化的相关理论

在给出算法之前,先介绍该算法所用到的相关理论。以二维情况为例,原始数据定义为0层,经过一次抽样后为1层,依此类推。此外,假设聚类数据每一维的坐标起始值为1,权值为非负,采用误差平方和最小为聚类准则。抽样方法如下:$k+1$层的点P,对应k层的4个点,即采用2(2的抽样方式,点P的权值等于其对应4个点权值的平均。设点P所对应k层4个点的坐标及权值均可用三元组表示,即$(i, j, w_{k,i,j})$,$(i+1, j, w_{k\,i+1,j})$,$(i, j+1, w_{k,i,j+1})$,$(i+1, j+1, w_{k,i+1,j+1})$,其中前二元代表坐标,$w_{k,i,j}$代表第$k$层,横坐标为$i$纵坐标为$j$的点的权值,$i$,$j$均为奇数。则点$P$的坐标及权值为$((i+1)/2, (j+1)/2, (w_{k,i,j} + w_{k,i+1,j} + w_{k,i,j+1} + w_{k,i+1,j+1})/4)$。

性质1 对于加权数据,可把每点的权值视为密度函数,则对于密度聚类,若采用误差平方和最小准则,其聚类中心即为其重心[10]。

性质2 基于前述抽样算法,层次抽样(或金字塔结构)相邻两层(k层和$k+1$层)聚类中心的任一维之间存在如下关系:

$$2X_{k+1} \geqslant X_k,$$
$$2X_{k+1} - X_k \leqslant 1 \qquad (1)$$

其中X_{k+1}为$k+1$层的某聚类中心的某一维坐标,X_k为k层对应聚类中心的对应维坐标,1为单位坐标长度。式(1)表明,若将第$k+1$层聚类中心的坐标投射到第k层(均换算为k层坐标),其大于等于k层相对应聚类中心的对应维坐标,但二者之差小于1,也即相邻两层经抽样后,同一聚类中心的任一维所引起的误差小于1。

证明 假设第$k+1$层第j个聚类中心为向量$CC_{k+1,j}$,可认为第$k+1$层聚类于该中心的点所对应的第k层的点也将聚类于相对应中心$CC_{k,j}$(实际可能会有少数不满足该特性的点,不会对结论产生本质的影响),根据性质1,该聚类中心即为重心。为不失一般性,以X坐标为例,分别计算第$k+1$层和k层的相应重心。

对于$k+1$层重心,

$$X_{k+1,j} = \sum_{m=1}^{n} w_{k+1,m}x_{k+1,m} \Big/ \sum_{m=1}^{n} w_{k+1,m} \qquad (2)$$

抽样后第$k+1$层点的权值等于其所对应第k层的4个点权值的平均,故$w_{k+1,m}$和k层对应点的x坐标权值的关系为

$$w_{k+1,m} = (w_{k,i,j} + w_{k,i+1,j} + w_{k,i,j+1} + w_{k,i+1,j+1})/4 \qquad (3)$$

则k层相对应聚类中心的X坐标为

$$X_{k,j} = \sum_{l=1}^{4n} w_{k,l}x_{k,l} \Big/ \sum_{l=1}^{4n} w_{k,l} \qquad (4)$$

令$T = \sum_{l=1}^{4n} w_{k,l}$,$S = \sum_{m=1}^{n} w_{k+1,m}$,且$T = 4S$,可得

$$2X_{k+1,j} - X_{k,j} =$$
$$2\sum_{m=1}^{n} w_{k+1,m}x_{k+1,m} \Big/ \sum_{m=1}^{n} w_{k+1,m} - \sum_{l=1}^{4n} w_{k,l}x_{k,l} \Big/ \sum_{l=1}^{4n} w_{k,l} =$$
$$\sum_{m=1}^{n} 8w_{k+1,m}x_{k+1,m} \Big/ T - \sum_{l=1}^{4n} w_{k,l}x_{k,l} \Big/ T \qquad (5)$$

抽样后$k+1$层的重心可用其对应的k层数据表示,现仅以$k+1$层某点和其所对应k层的4个点为例,可得

$$8w_{k+1,m}x_{k+1,m} =$$
$$(w_{k,i,j} + w_{k,i+1,j} + w_{k,i,j+1} + w_{k,i+1,j+1})(i+1) \qquad (6)$$
$$w_{k,l}x_{k,l} =$$
$$w_{k,i,j}i + w_{k,i+1,j}(i+1) + w_{k,i,j+1}i + w_{k,i+1,j+1}(i+1) \qquad (7)$$

其中i为奇数。

联立方程式(5)至式(7),得

$$2X_{k+1,j} - X_{k,j} = \left[\sum_{m=1}^{n} w_{k,2m-1,j} + w_{k,2m-1,j+1}\right] \Big/ T \qquad (8)$$

式(8)表明,由于分母T是k层所有属于该类点的权值之和,而分子$\sum_{m=1}^{n} w_{k,2m-1,j} + w_{k,2m-1,j+1}$仅为其中部分点的权值之和($X$坐标为奇数所对应的所有点的权值之和),因此$2X_{k+1,j} - X_{k,j}$不大于1,但一定大于0。性质2证毕。

3 基于层次初始化的聚类算法(HIKM)

3.1 算法描述

为简单起见,假设数据为二维数据,每一维大小相等且为2的整数次幂,clusternum为聚类类别数。HIKM算法主要包括自底向上和自顶向下两个过程,首先通过层层抽样减少数据,然后进行聚类,得

到一组聚类中心,再根据该聚类中心,推算原始数据的初始聚类中心,进行聚类。预处理将原始数据处理为所需要的形式。

HIKM 算法步骤如下:

Step 0:对于原始数据进行预处理,通过线性变换把原始数据的坐标变换为[1 … 2^N]范围内的整数。

Step 1:对原始数据进行层层抽样,直到某层数据个数为大于 20 * clusternum 的最小整数为止。

Step 2:对抽样结束层的数据,采用 K - means 迭代算法进行聚类,得到该层的聚类中心。具体做法:首先,对所有的数据点按照权值进行排序;然后,选择前 clusternum 个点作为初始聚类中心进行迭代,找到该层数据的聚类中心。

Step 3:基于相邻两层聚类中心的位置关系(性质 2),从高层向低层逐层进行聚类,即将 $k+1$ 层的聚类中心投射到 k 层,作为 k 层的初始聚类中心进行聚类运算,得到 k 层的聚类中心,再把 k 层的聚类中心投射到 $k-1$ 层,作为 $k-1$ 层的初始聚类中心,依此类推,直到原始数据层为止。然后,依据此初始聚类中心,对原始数据进行聚类。

Step 4:对 Step 3 得到的聚类中心,采用 Step 0 中的逆变换,得到真正的聚类中心。

3.2 HIKM 算法的分析及说明

3.2.1 HIKM 算法说明

1)假设某坐标值为 x,Step 0 所采用的线性变换为$(2^N-1)(x-\mathrm{mi})/(\mathrm{ma}-\mathrm{mi})+1$,其中 ma 和 mi 分别为所有数据点坐标的最大值和最小值。由于聚类算法的测度是基于欧氏距离的,而上述线性变换只改变距离的绝对大小,不改变相对距离,因此原始数据的误差平方和与变换数据误差平方和之间只相差一个系数,对于最终的聚类结果没有本质的影响。

2)若数据每一维的大小不是 2 的整数次幂,可将其扩充为 2 的整数次幂,扩充部分的权值为 0。

3)Step 1 抽样结束条件为某层剩余数据量大于 20 * clusternum 的最小整数(clusternum 大小可根据需要进行调整),使得最终用于找到初始聚类中心的数据点不要太少,否则聚类无意义。假设数据大小为 256(256,clusternum = 3,则抽样过程应该多在 8×8 层结束,这样可以保证数据点多于 20×3 个。

4)对于 Step 2,尽管数据较少,但若随机选取初始聚类中心,则很容易收敛到局部最优,故采用排序策略,即取权值最大的前 clusternum 个点作为该层聚类迭代的初始聚类中心。该策略若直接用于原始数据,有可能选取噪声点,而经过多层抽样后,如果是噪声点的话,其周围数据点应该很少或者周围数据点的权值都应该很小,按照 HIKM 抽样方法,在二维情形下,噪声点的权值按照$(1/4)^k$进行衰减,其中 k 代表抽样的次数,衰减的速度很快,噪声会被严重削弱,因此权值大的点也就几乎不可能为噪声点了。

5)HIKM 算法处理完所有数据点后再更新聚类中心,使算法结果与数据顺序无关。

3.2.2 HIKM 算法分析

1)存储空间分析 假设数据维数为 D,每维大小为 S,则原始数据的存储空间 $M=S^D$,采用层层抽样算法,抽样后每层数据量以 0.5^D 的速度减少,极限情况下(即抽样的次数为无穷)所需的存储空间为 $M/(1-0.5^D)$。$D=1$ 时总的存储空间为 $2M$,其中一半用于存储原始数据,另一半存储抽样后数据;$D=2$ 时所需存储空间为 1.3333 M,其中 0.3333 M的空间存储抽样数据;依此类推,随着 D 的增加,抽样后数据的存储空间越小。

2)时间复杂度分析 HIKM 算法的主要时间开销在于 Step 3 中的迭代聚类,Step 2 中抽样聚类时间却很少,抽样过程需处理的总数据量(极限情况下)仅 $M/(1-0.5^D)$。原始数据层迭代一次所处理的数据量为 clusternum * M,且涉及大量的浮点乘法运算,而抽样过程仅涉及加法运算(实际处理过程无须采用除法运算求平均)。与原始数据量相比,Step 2 中聚类的高层数据很少,抽样后第一层的数据量仅为原始数据的 $1/2^D$,第 k 层迭代相同次数的运算量仅为原始层的$(1/2^D)^k$。若 D 和 k 均为 2,迭代相同次数的运算量约为原始层的 0.063;若 D 和 k 均大于 2,迭代相同次数的运算量更小,与原始数据层的运算时间相比,几乎可以忽略。可见,与 Step 3 相比,Step 2 的时间开销几乎可以忽略。

对于 Step 3,每层初始聚类中心距真正聚类中心很近,在每一维上均小于一个单位长度,因而每层的聚类速度都很快,通常几次迭代即可收敛。因此,采用基于层次的方法进行分层迭代,可以得到较好的初始聚类中心,再通过层层映射直到原始数据层,可显著减少原始数据层的迭代次数,从而节省了聚类算法总的运算时间,使算法收敛速度大大提高。

4 实验结果及分析

为验证 HIKM 算法的有效性,聚类数据采用

UCI 机器学习数据库中的真实数据集:Iris 数据集、Ruspini 数据集和 Glass 数据集[3]。上述数据集通常有 100 ~ 200 实例,每个实例最多 9 个属性,聚类的类别数为 3 至 4 个,主要和以下初始化算法进行比较:随机化算法、Kaufman 等的 KA 方法[4]和 Fayyad 等的方法[7],对比实验研究分为两组,其中一组根据 Pena 的结论[3],KA 方法和随机化方法的总体性能最好,因此,采用方差作为聚类质量的评判指标,采用聚类开始到结束所用时间(包括聚类中心初始化)评价收敛速度,将 HIKM 方法与 KA 方法及随机化方法进行比较。另一组则将 HIKM 方法和 Fayyad 方法及随机化方法进行对比,主要基于 UCI 数据库的加权数据,但评价聚类质量的指标为算法收敛到全局最优的平均迭代次数。上述实验的运行环境:PIII800 笔记本电脑,128M 内存,WindowsXP 和 MATLAB 6.5。

4.1 与 KA 方法及随机化方法的对比

采用 UCI 机器学习数据库中的 Iris 和 Glass 数据集,将随机化方法及 Kaufman 的 KA 方法与 HIKM 方法进行比较,Iris 数据集的线性变换区间为 1 ~ 256,Glass 数据集的线性变换区间为 1 ~ 4096,基本上不产生舍入误差,不影响最终聚类结果。算法聚类质量数据如表 1、表 2 所示,均为算法运行 100 次后的平方误差,K 表示聚类类别数。采用 HIKM 方法和 KA 方法,聚类结果均具有唯一性,但随机初始化方法的聚类结果具有不确定性,故分别列出最好和最差结果。可以看出,对于 Iris 数据集和 Glass 数据集聚类,HIKM 方法和 KA 方法及随机初始化方法的最佳结果基本一致,尤其表 2 中的 Glass 数据集聚类,$k=2$ 时层次抽样方法的平方误差最小。

表 1 Iris 数据集的聚类质量

Table 1 Square-error for Iris data set

	$K=3$	$K=4$
随机初始化方法	78.94	57.32
(分别代表最好和最差的结果)	78.95	71.66
KA 方法	78.94	57.32
HIKM 方法	78.94	57.47

表 2 Glass 数据集的聚类质量

Table 2 Square-error for Glass data set

	$K=2$	$K=7$	$K=10$
随机初始化方法	838.78	298.16	230.07
(分别代表最好和最差的结果)	840.19	571.60	321.31
KA 方法	838.78	298.16	230.07
HIKM 方法	819.65	315.49	243.48

Pena 等的研究结果[3]表明 KA 的收敛速度最快,因此仅与 KA 方法进行时间性能比较。实验仿真结果如表 3、表 4 所示,均为算法运行 100 次时间的平均值。可以看出,HIKM 方法能显著减少算法时间。随着类别数的增加,HIKM 方法运行时间的增加幅度远远低于 KA 方法,对于 Glass 数据集,当类别数 k 从 2 增加到 7 时,KA 方法运行时间从 3 s 多增加到 31 s 多,而 HIKM 方法的运行时间仅增加了零点几秒。此外,线性变换范围对 HIKM 方法运行时间的影响较小,表 3 中线性变换范围增加了 64 倍,而 HIKM 算法所增加的运行时间不超过 10 %,表 4 的线性变换范围增加了 4 倍,但增加的运行时间不超过 30 %(0.5 s)。

表 3 Iris 数据集的算法运行时间

Table 3 Running time for Iris data set s

	$K=2$	$K=7$	$K=10$
KA 方法	3.04	31.7	64.6
HIKM 方法 (4096)	1.26	1.53	1.55
HIKM 方法 (16384)	1.38	1.871	2.066

表 4 Glass 数据集的算法运行时间

Table 4 Running time for Glass data set s

	$K=3$	$K=4$
KA 方法	3.45	6.03
HIKM 方法 (256)	0.55	0.759
HIKM 方法 (16384)	0.561	0.795

4.2 与 Fayyad 方法及随机化方法的对比

模拟数据是二维数据,大小为 256 × 256,分别采用混合高斯分布和平均分布对每一数据点赋权值,且加入正态分布噪声,旨在测试算法对噪声的敏感程度。

4.2.1 正态分布 + 噪声的情形 生成数据大小为 256 × 256,3 个中心分别取为(128,64),(64,192),(192,192)。权值采用正态分布函数

$$f(x,y)=(1/2\pi\sigma^2)\exp[-((x-x_0)^2+(y-y_0)^2)/\sigma^2] \quad (9)$$

其中 $\sigma=10$。为避免权值太小,仿真实验中实际采用函数

$$f(x,y)=\exp[-((x-x_0)^2+(y-y_0)^2)/\sigma^2] \quad (10)$$

每个点的权值等于 3 个不同中心的正态分布函数值之和,即对以(128,64),(64,192),(192,192)为中心的正态分布函数值求和,在此基础上叠加平均分布的随机噪声,即得每个点的最终权值。

对于 Fayyad 方法[7],每次抽样 1 % 数据,可得一组聚类中心,共抽样 5 次,共得 5 组聚类中心,在此基础上采用 K - means 算法得到最终初始聚类中心。算法仿真结果如表 5 所示,均为 100 次实验的平均,Case 0 为未加噪声的数据,Case 1 对应(0.05, 0.05)区间上均匀分布的随机噪声(若该值小于 0 则取 0),Case 2 对应(0 ~ 0.1)范围内均匀分布的随机噪声。

由表 5 可以看出,HIKM 方法能收敛到全局最优,且迭代聚类任一层迭代次数一般小于 10,为 6 ~ 7 次。Case 1 最终聚类中心在(62, 187),(128, 59),(194, 186)附近,Case 2 最终聚类中心在(61, 181),(128, 55),(196, 181)附近。

表 5 正态分布加权数据实验结果

Table 5 Experimental results for normal distribution weighted data

Case	比较项目	HIKM 算法	随机初始化方法	Fayyad 方法
0	平均时间/s	14.2	26.8	28.9
	收敛到局部最优的次数	0	21	26
1	平均时间/s	21.7	46.7	38.8
	收敛到局部最优的次数	0	2	1
2	平均时间/s	44.1	109.3	110.3
	收敛到局部最优的次数	0	21	24

4.2.2 均匀分布 + 噪声的情形 实验数据大小仍为 256 × 256,每点权值为(0, 1)区间均匀分布,并叠加中心分别为(100, 100), (100, 200), (200, 200)正态分布的噪声,聚类类别数为 4。算法仿真结果如表 6 所示,均为 100 次实验的平均结果,其中 Case 0 表示未加噪声,Case 1 则对应加噪声数据。

表 6 均匀分布加权数据实验结果

Table 6 Experimental results for uniform distribution weighted data

Case	比较项目	HIKM 算法	随机初始化方法	Fayyad 方法
0	平均时间/s	47.9	82.6	81.4
	收敛到局部最优的次数	0	0	0
1	平均时间/s	61.8	94.1	87.4
	收敛到局部最优的次数	0	0	2

4.3 结果分析

对于采用层次初始化的 K - means 算法,由上述实验结果可得如下结论:

1)算法具有较快的收敛速度。模拟数据和真实数据均显示新算法的收敛速度较传统的 K - means 算法有显著提高;

2)算法的聚类质量较高。对于模拟数据,能收敛到全局最优;对于真实数据,尽管最终所得的平方误差不一定能达到最小值,但接近最小值;

3) 算法的结果具有唯一性,即给定数据顺序后,新算法的最终聚类结果唯一,而随机初始化算法和 Fayyad 方法对于同样数据每次的聚类结果则不唯一;

4)算法具有较好的抗噪性能,在有噪声的情况下,也能获得较好的聚类结果。

5 结语

针对 K - Means 算法,提出了一种新的基于层次思想的初始聚类中心选取方法。该方法具有如下特点:

1)对于 K - means 算法而言,在抽样过程中信息丢失很少,一般认为高层信息基本上完全包含了低层信息;

2)新算法在抽样过程中具有更强的抑制噪声能力;

3)对于相邻两层间的聚类中心,高层聚类中心投射到相邻低层后,与低层真正聚类中心相比,其每维之差均小于 1,即初始中心和实际中心很接近,从而使得新算法具有较快的收敛速度;

4)对于 HIKM 算法,初始化结果和数据顺序无关,最终聚类结果具有唯一性,即不会产生多种聚类结果。模拟数据和实际数据的实验结果验证了新算法的正确性和有效性,表明其性能不仅优于文献[4]和文献[7]的方法,而且优于传统的随机初始化方法。

参考文献

[1] Keim D A, Hinneburg A. Clustering techniques for large data sets—from the past to the future [A]. In: Proc Tutorial Notes 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining [C]. San Diego, 1999. 141 ~ 181

[2] Duda R O, Hart P E. Pattern Classification and Scene Analysis [M]. New York: John Wiley and Sons, 1973

[3] Pena J M, Lozano J A, Larranaga P. An empirical comparison of four initialization methods for the K - means algorithm [J]. Pattern Recognition Letters, 1999, 20(10): 27 ~ 40

[4] Kaufman L, Rousseeuw P J. Finding Groups in Data: An Introduction to Cluster Analysis [M]. Wiley, Canada, 1990

[5] Forgy E. Cluster analysis of multivariate data: efficiency vs interpretability of classifications [J]. Biometrics 1965, 21: 768

[6] MacQueen J B. Some methods for classification and analysis of multivariate observations [A]. In: Proc Symposium on Mathematics and Probability, 5th, Berkely, Vol 1, AD 669871 [M]. University of California Press, Berkeley, CA, 1967. 281 ~ 297

[7] Fayyad U M, Renia C A, Bradley P S. Initialization of iterative refinement clustering algorithm [A]. In: Proceedings of the International Conference on Knowledge Discovery and Data Mining (KDD98) [C]. New York, 1998. 94 ~ 98

[8] Bradley P S, Fayyad U. Refining INITIAL Points for K - means clustering [A]. In: Proc 5th Int Conf Machine Learning [C]. Morgan Kaumann, 1998

[9] Khan S S, Ahmad A. Cluster center initialization algorithm for K - means clustering [J]. Pattern Recognition Letters, 2004, 25 (11): 1293 ~ 1302

[10] Du Q, Faber V, Gunzburger M. Centroidal voronoi tessellations, theory, applications and algorithms [J]. SIAM Review, 1999, 41(4): 637 ~ 676

A Hierarchical-Based Initialization Method for K - Means Algorithm

Tang Jiubin, Lu Jianfeng, Tang Zhenmin, Yang Jingyu

(*Department of Computer, Nanjing University of Science and Technology, Nanjing 210094, China*)

[**Abstract**] K - means algorithm is one of common clustering algorithms, but the cluster center initialization is a hard problem. In this paper, a hierarchical-based initialization approach is proposed for K - Means algorithm. The general clustering problem is treated as weighted clustering problem, the original data is sampled level by level to reduce the data amount. Then clustering is carried out at each level by top-down. The initial center of each level is mapped from the clustering center of upper level and this procedure is repeated until the original data level is reached. As a result, the initial center for the original data is obtained. Both the experimental results on simulated data and real data show that the proposed method has high converging speed, high quality of clustering and is insensitive to noise, which is superior to some existing clustering algorithms.

[**Key words**] hierarchical technique; initial cluster centers; weighted data, K - means clustering