面向层次类型变量的相异度量及其聚类分析*)

杨培颖　王大玲　于　戈　陈冬玲
(东北大学信息科学与工程学院　沈阳 110004)

摘　要　本文在分析传统类型变量相异度量的基础上,定义了"层次类型"的概念,提出了层次类型变量的相异度量计算方法。引入层次类型变量,并结合传统类型变量,设计了具有包括层次类型在内的混合数据类型描述的对象之间的相异度量方法,并基于此实现了此类对象的聚类分析。

关键词　聚类,层次变量,相异度,k-medoids 算法

Dissimilarity Metric and Clustering for Hierarchy Variable

YANG Pei-Ying WANG Da-Ling YU Ge　CHEN Dong-Ling
(School of Information Science and Engineering, Northeastern University, Shenyang 110004)

Abstract　Based on the analysis for the dissimilarity metric of traditional type variables, the hierarchy type is defined, and the dissimilarity metric for the type is proposed in the paper. Moreover, the dissimilarity metric of the hybrid type including traditional types and the hierarchy type is designed, and a clustering algorithm based on the metric is implemented.

Keywords　Clustering, Hierarchy variable, Dissimilarity, k-medoids

1　引言

聚类分析是指将物理或抽象对象的集合分组成为由类似的对象组成的多个类的过程。由聚类所生成的簇是一组数据对象的集合,这些对象与同一个簇中的对象彼此相似,与其它簇中的对象相异,因此,聚类分析的基础是对象间相似性或相异度的计算。对象一般由不同类型的属性(变量)所描述,而不同类型的变量之间的相异度计算方法不同。目前已知的数据类型包括:区间标度变量、二元变量、标称变量、序数型变量和比例标度型变量。

但是,在实际的聚类应用中,存在这样一类对象,它们需要不同于上述传统类型的变量来描述。例如,对高等学校的描述,除学校等级、招生规模、教师人数这些传统数据类型外,专业设置则很难用上述类型来准确描述,对于综合性大学,将包括学院、系、专业等,而它们之间的关系构成了图 1 所示的层次结构。

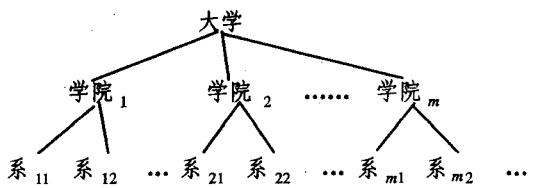

图 1　层次结构

对于这样的类型所描述对象的相异度,也很难用上述传统的方法来度量。如,系$_{11}$、系$_{12}$、系$_{21}$、系$_{22}$之间的相异度,按照传统的度量方法,由于它们的名字不同,则不相似。但实际上,由于系$_{11}$与系$_{12}$同属于学院$_1$,而系$_{21}$与系$_{22}$同属于学院$_2$,因此,有理由认为,系$_{11}$与系$_{12}$的相似度大于系$_{11}$与系$_{21}$之间的相似度,同样系$_{21}$与系$_{22}$之间的相似度大于系$_{21}$与系$_{11}$之间的相似度。

基于此,本文定义了"层次类型"的概念,提出了层次类型变量的相异度量计算方法。设计了具有包括层次类型在内的混合数据类型描述的对象之间的相异度量方法,并基于此实现了此类对象的聚类分析。

2　相关工作

聚类分析的基础是对象间相似性或相异度的计算,不同的数据类型有不同的相异度量方法。现有的相异度量方法主要包括以下几种[1]:

设 $I=\{i_1, i_2, \cdots, i_p\}$ 是 p 个不同属性的集合,对象 $T_i \subseteq I, D=\{T_1, T_2, \cdots, T_n\}$ 是关于 T_i 的集合,希望把 D 中的对象分为 k 个类 $C_k=\{C_1, C_2, \cdots, C_k\}$。$I$ 中的各个属性可能是不同的数据类型,分别讨论如下:

① 区间标度变量:区间标度变量是个粗略线性标度的连续变量。传统的处理方法是对变量进行标准化处理后,对象间(i 和 j)的相异度基于对象间的距离来计算。最常用的如欧几里得距离,定义如下式:

$$d(i,j)=\sqrt{|x_{i1}-x_{j1}|^2+|x_{i2}-x_{j2}|^2+\cdots+|x_{ip}-x_{jp}|^2}$$

② 二元变量:只有两个状态(0 或 1)的变量称为二元变量。对这类变量,传统的度量方法是以 0(或 1)在所有记录中所占的比例作为相异度的计算标准[2]。

③ 标称变量:具有多于两个离散状态值的变量称为标称变量。两个由标称变量描述的对象 i 和 j 之间的相异可以用简单匹配方法来计算:$d(i,j)=\frac{p-m}{p}$。其中 p 为全部标量的

*)国家自然科学基金资助项目(No. 60573090),辽宁省自然科学基金资助项目(No. 20060232)。杨培颖　硕士,主要从事文本挖掘技术研究。王大玲　博士,教授,CCF 高级会员,主要从事 Web 挖掘技术研究。于　戈　博士,教授,CCF 高级会员,主要从事数据库及相关技术研究。

数目，m 为 i 和 j 取值相同的数目(匹配的数目)。

④ 序数型变量：序数型变量分为离散的序数型变量和连续的序数型变量两种。序数型变量类似于标称变量，不过序数型变量的各个状态是以有意义的序列排序的。通常，可以将序数型变量的值映射为秩。计算两个对象 i 和 j 相异度时，用相应的秩代替实际取值，并标准化到[0,1]之间，之后利用距离度量方法进行计算。

⑤ 比例标度型变量：比例标度型变量在非线性的标度上取正的度量值，例如指数标度。常用的方法有对其进行对数变换，作为区间标度变量来处理，或是作为序数型变量来处理。

⑥ 混合型变量：一种方法就是将变量按类型分组，对每种类型的变量进行单独的聚类分析。但在实际应用中，根据每种类型的变量单独进行聚类分析不太可能获得满意的结果。

一个更好的方法就是将所有类型的变量统一处理，一次完成整个聚类分析。一种技术将不同类型的变量组合在单个相异度矩阵中，把所有有意义的变量转换到共同的值域区间[0.0,1.0]。

3 问题分析和求解

3.1 相异度计算

在传统的区间标度变量的相异度量方法中(以欧几里得距离为例)，我们注意到，往往并非必须得到参与计算的每一个属性的具体数值，而只须得到对应属性的差值就可以计算距离。我们将该方法引入层次类型数据的距离度量中。我们将图 1 实例中的层次类型抽象成为概念层次树(如图 2)，将每个变量抽象为一个概念，基于概念层次树定义两个概念之间的距离。

定义 1 对于一个概念层次树上的两个概念 d_1、d_2，我们定义它们的距离是从 d_1 沿概念层次树到 d_2 的最短路径的长度。

以图 2 所示的概念层次树为例。

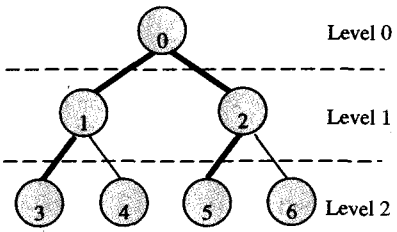

图 2 概念层次树

图中，节点 3、节点 4、节点 5、节点 6 表示处在最细节概念层次中的各个概念，如果我们计算节点 3 和节点 5 之间的距离，按照定义 1，它们之间的距离即加粗的黑线的路径的长度。层次越高，概念越粗化，有理由认为，层次越高，不同层次间节点之间的距离就越大。如图 2 中，节点 3 到节点 1 的距离小于节点 1 到节点 0 的距离。对于不同层次间概念距离的度量，我们可以采用等差的方法，也可以采用等比的方法度量垂直相邻的两个概念层次的距离。假设节点 3 到节点 1 的距离我们定义为 1，如果采用等差的方法，节点 1 到节点 0 的距离就可以设为 $1+k$，而采用等比的方法，节点 1 到节点 0 的距离就可以设为 $1\times k$。所以当 $k=1$ 时(等差)，上边的例子中 $d(3,5)=d(3,1)+d(1,0)+d(0,2)+d(2,5)=1+2+2+1=6$。等比法($k=3$)，距离则为 $1+3+3+1=8$。

这样计算层次型分类数据距离的优点是，两个概念的最小公共祖先所处的概念层次越低，则这两个概念的距离越小——这符合一般的数学逻辑模型思维。

3.2 数据预处理

实际的对象常常使用不同类型的多个变量描述。除层次类型变量外，还包括前述的各种传统类型变量。

如果描述对象的属性是多个数值类型的，并且这些属性之间数值量级差别较大，在数据预处理时，将采用归一化的方法，以统一量级，使这些属性对最终距离的度量有近似相等的贡献。例如：一个二维的点集，经过归一化，假设点与点之间的平均距离是 d，那么每一个维对这个平均距离的贡献将是 $\sqrt{d^2/2}$。推广之，对于一个 n 维的数据集，可以抽象成 n 维空间中的一个点集，经过归一化，假设它们的平均距离是 d，那么每一个维对这个平均距离的贡献就是 $\sqrt{d^2/n}$(这里我们假设采用的是欧几里得距离)。

对于层次类型属性，采用定义 1 计算两个变量的距离，运用这个距离参与混合属性的距离计算，需要对分类型属性的距离进行转换。我们同样借用归一化的思路，假设参与混合属性距离计算后的各个属性对最终距离的贡献是近似相等的，这样我们就可以通过计算数值型属性对距离的贡献，间接计算出分类型属性对距离的贡献，将这个值作为“单位 1”。为此，我们只保留数值型数据，然后按照开始的方法($\sqrt{d^2/n}$)度量每一个数值型属性对距离的贡献，以此作为“单位 1”。

4 实验评估

4.1 算法

这里我们采用 k-medoids 聚类算法[3]，该算法的基本策略是：首先为每个簇随意选择一个代表对象，剩余的对象根据其与代表对象的距离分配给最近的一个簇，然后反复地用非代表对象来代替代表对象，以改进聚类的质量。

一个典型的 k-medoids 算法描述如下：

算法：k-medoids；
输入：聚类数 k，以及包含 n 个数据对象的数据库；
输出：k 个聚类；
处理流程：

(1)从 n 个数据对象任意选择 k 个对象作为初始聚类(中心)代表；
(2)循环(3)到(5)直到每个聚类不再发生变化为止；
(3)依据每个聚类的中心代表对象，以及各对象与这些中心对象间距离，并根据最小距离重新对相应对象进行划分；
(4)任意选择一个非中心对象 O_{random}；计算其与中心对象 O_j 交换的整个成本 S；
(5)若 S 为负值，则交换 O_{random}，与 O_j 以构成新聚类的 k 个中心对象。

PAM[4](Partitioning around Medoid，围绕中心点的划分)是最早提出的 k-medoids 算法之一。它试图对 n 个对象给出 k 个划分。最初随机选择 k 个中心点后，该算法反复地试图找出更好的中心点。所有可能的对象对被分析，每个对中的一个对象被作是中心点，而另一个不是。对可能的各种组合，估算聚类结果的质量。一个对象 O_j 被可以产生最大平方误差值减少的对象代替。在一次迭代中产生的最佳对象的集合成为下次迭代的中心点。

4.2 实验及分析

由于目前可供实验的聚类数据集中不包含层次类型属性，因此我们的实验数据是由程序合成的具有层次类型变量的分类数据集(设定了类标签)。

生成的数据集共有 n 个属性，前 $n-1$ 个属性是数值型属性，最后一个属性是层次类型属性。我们按照一定的规律不

断变化各属性的值，以确定 k 个类及每个类的记录数 i，因此最后生成的数据集的规模将是 $i \times k$。

就第 i 个类的空间形状而言，它是以 $(4 \times i, 0, 0 \cdots)$ 为中心，1 为边长的一个超立方体。类内的每一条记录，它们的层次属性分布且仅分布在概念层次树树根的一棵子树上——即对分别属于不同类的任意两条记录，它们的层次属性的公共祖先只会是概念层次树的根(如图 3 所示)。

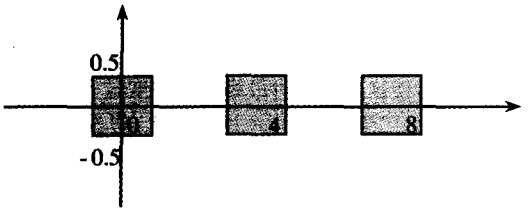

图 3　数据集分布

采用此方法合成的数据，有如下特点：数值型属性有着较小的类内距离和较大的类间距离；在同一个类内的层次型数据，概念距离更接近。综上所述，合成的类有着比较低的类内相异度，和较高的类间相异度。

我们的测试环境为 P3 866、128M 内存、4G 硬盘、Slackware Liunx 11、内核版本 2.4.33、gcc 版本 3.4.6、编译选项"-O3"。

我们应用 k-medoids 聚类算法测试我们的相异度计算方法与传统方法对于聚类准确性的影响。两种度量方法的主要区别在于层次类型变量的相异度计算，在传统方法中，只比较属性的值而不管其所在层次和位置，而我们的方法则根据定义 1 的方法进行计算。

实验 1：聚类数 $k=3$，属性数 $d=10$(其中 9 个数值属性，1 个层次属性)，结果如表 1。

表 1　$d=10$ 时实验结果的对比表

每类记录数	本文的度量方法		传统的度量方法	
	运算时间(秒)	准确度(%)	运算时间(秒)	准确度(%)
1000	1	99.3	1	89.7
2000	4	99.5	4	86.7
4000	38	99.9	18	83.9
8000	225	99.8	249	75.4
10000	278	99.7	251	79.2

实验 2：聚类数 $k=3$，属性数 $d=6$(其中 5 个数值属性，1 个层次属性)，结果如表 2。

由于这两种方法结果都不很稳定，准确度与开始时随机选择的中心点有一定关系，因此我们对每一个样本数据都进行了 5 次实验，取平均值。

与传统的相异度计算方法相比，我们的算法度量更具合理性和可解释性，所以准确度更高。当然，由于层次类型属性的相异度计算需要比较概念层次树中不同层次的值，因此通常比传统的计算方法需要更多的时间。

表 2　$d=6$ 时实验结果的对比表

每类记录数	本文的度量方法		传统的度量方法	
	运算时间(秒)	准确度(%)	运算时间(秒)	准确度(%)
1000	1	99.9	1	99.7
2000	4	99.9	4	96.7
4000	38	99.9	18	93.9
8000	225	96.9	249	95.4
10000	278	97.4	251	89.2

结束语　本文在分析传统类型变量相异度量的基础上，定义了"层次类型"的概念，提出了层次类型变量的相异度量计算方法。引入层次类型变量，并结合传统类型变量，设计了具有包括层次类型在内的混合数据类型描述的对象之间的相异度量方法，并基于此实现了此类对象的聚类分析。实验表明，对于具有层次型属性的数据集，其聚类准确度高于传统的相异度度量方法。

本文所提出的算法对于层次型和数据型的混和类型数据比较有效，而其它类型数据处理还需要采用传统的方法度量。另外，两个层次类型属性值的相异度的计算方法本身也有待于进一步改进，例如将节点的位置与节点值综合考虑进行相异度计算。因此在今后的工作中，我们将在以下两个方面进行更深入的研究。

(1) 对于层次类型属性，进一步探索更加准确、合理的相异度计算方法，包括两个层次类型变量对应的概念层次树中各节点本身的值、其祖先和子孙节点的值以及层次关系等；

(2) 研究包括层次类型属性在内的多种类型混合的数据类型所描述对象的相异度计算方法。

参 考 文 献

1　李桂林，陈云晓. 关于聚类分析中相似度的讨论[J]. 计算机工程与应用，2004，31：64～82

2　Han J，Kamber M. Data Mining：Concepts and Techniques. Higher Education Press，Morgan Kaufmann Publishers，2001，5

3　郭金华. 数据挖掘中聚类分析的研究[D]. 武汉理工大学管理学院，2003

4　Jain A K，Murty M N，Flynn P J. Data Clustering：A Review. ACM Computing Surveys，1999，31(3)：264～323

(上接第 161 页)

理，因此本文所提出的实现方案在满足多个请求时尚有一定的困难，访问控制的安全性还需进一步的认识和解决。

参 考 文 献

1　Park N，Moon K，Sohn S. XML key management system for Web-based business application. In：Network Operations and Management Symposium 2004，NOMS 2004 on April 2004，1：903～904

2　Park N，Moon K，Sohn S. A study on the XKMS-based key management system for secure global XML web services. Advanced Communication Technology，2004，1：492～495

3　Lorch M，Kafura D. An XACML-based policy management and authorization service for globus resources. In：Grid Computing 2003. Proceedings on Nov 2003. 208～210

4　Anderson. An introduction to the Web Services Policy Language (WSPL). In：Policies for Distributed Systems and Networks on June 2004. 189～192

5　Matheus A. How to Declare Access Control Policies for XML Structured Information Objects using OASIS' eXtensible Access Control Markup Language (XACML). In：System Sciences 2005. HICSS '05 on 03 Jan 2005. 168a～168a

6　Marin L G. A Network Access Control Approach Based on the AAA Architecture and Authorization Attributes. In：Parallel and Distributed Processing Symposium 2005 on April 2005. 287a～287a

7　Jeong J，Dongkyoo S，Dongil S. An XML-based single sign-on scheme supporting OSGi framework. In：Consumer Electronics 2005 on Jan. 2005. 31～32