

基于属性区分能力和 AP 聚类的属性粒化方法

朱 红^{1,2} 丁世飞²

(徐州医学院医学信息学院 徐州 221005)¹ (中国矿业大学计算机科学与技术学院 徐州 221116)²

摘 要 提出了一种基于属性区分能力和 AP 聚类的属性粒化方法(Attribute Granulation based on attribute discernibility and AP algorithm, AGAP)。该方法首先依据属性依赖度计算属性的区分能力;然后将所有属性作为潜在的聚类中心,使用 AP 算法聚类,得到若干个属性簇类;最后采取选用代表属性的方法得到较粗的属性粒子,从而达到属性粗粒化的要求。对高维数据的特征降维,这种算法比传统的属性约简算法大大提高了运算效率,在属性粒化精度要求不是很严格的情况下,所提算法优势明显。

关键词 属性区分能力, AP 聚类, 属性粒化

中图分类号 TP181 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2016.2.021

Attribute Granulation Based on Attribute Discernibility and AP Clustering

ZHU Hong^{1,2} DING Shi-fei²

(School of Medical Information, Xuzhou Medical College, Xuzhou 221005, China)¹

(School of Computer Science and Technology, China University of Mining and Technology, Xuzhou 221116, China)²

Abstract This paper put forward a kind of attribute granulation method based on attribute discernibility and AP clustering. The method calculates the similarity of attributes according to attribute discernibility firstly, and then clusters attributes into several groups through affinity propagation clustering algorithm. At last, representative attributes are produced through some algorithms to form a coarser attribute granularity. The method is more efficient than traditional attribute reduction algorithm for large data set. It has obvious advantages under the condition of less strict precision of attribute granularity.

Keywords Attribute discernibility, AP clustering, Attribute granulation

1 引言

属性粒是对属性集从不同角度不同层次的划分,将属性集粒化会产生粒度不同的属性粒子,最粗的属性粒是每个属性,最细的属性粒是整个属性集。所有属性粒化的方法从根本上来讲可以归为两类:(1)属性约简,这种方法在保持分类能力不变的情况下,在较粗的粒度上进行属性的选取,但是它的时间空间复杂度很高,而且求系统的属性集的最小约简问题是一个 NP-hard 问题,因此降低属性约简算法的时间空间复杂度一直是人们追求的目标^[1-5],大数据集下基于粒度计算的属性约简算法显得尤为重要^[6];(2)特征降维,它也是在较粗的粒度上进行属性的选取,但并不强调分类能力不变,其时间空间复杂度远小于属性约简。聚类是一种重要的属性粒化方法^[7]。通过聚类,可以将一类属性聚在一起构成一个属性子集,也就是一个属性粒子。在每个簇类中依据信息熵、属性重要度等指标选取代表属性,构成最后的属性集合,从而完成属性粗粒化的要求。属性之间的相似性度量是属性聚类的关键问题。本文首先计算属性区分能力并将其作为相似性度量依据,然后采用 AP 算法对属性聚类,选取代表属性。AP 聚

类算法把所有的数据点作为潜在的聚类中心,不需要类数、聚类中心等先验知识,而我们对属性的区分能力的类数等信息一般知之甚少,因此算法的准确率高于其它基于聚类的属性粒化方法。

2 属性粒化相关概念

定义 1(属性粒) 设 A 是论域 U 中对象的属性集合, A_1, A_2, \dots, A_n 是 A 的子集,则称 A_1, A_2, \dots, A_n 是论域 U 上的属性粒。

属性粒的大小是对它所包含的信息量的度量。属性粒越大说明它包含的信息量越大。最简单的定义方法是根据属性粒所包含的属性个数来定义,也可以根据属性粒包含的信息熵(属性熵)来定义。

定义 2(属性粒的大小) 设 A_1, A_2, \dots, A_n 是论域 U 上的属性粒,属性粒的大小为 $d(A_i) = \text{Card}(A_i) = |A_i|$,也就是属性粒的大小表示属性粒包含的属性的个数。

定义 3(属性熵) 设 A 是论域 U 中对象的属性集合, $A_i = \{a_1, a_2, \dots, a_n\}$ 是论域 U 上的属性粒, A_i 的属性熵定义为:

到稿日期:2015-03-22 返修日期:2015-05-19 本文受国家自然科学基金项目(61379101),江苏省自然科学基金项目(BK20130209),江苏省高校自然科学基金项目(14KJB520039),江苏省高校优秀中青年骨干教师和校长境外研修计划资助项目资助。

朱 红(1970—),女,博士,副教授,主要研究方向为数据挖掘、粒度计算, E-mail: zhuhongwin@126.com; 丁世飞(1963—),男,博士,教授,主要研究方向为智能信息处理、数据挖掘, E-mail: dingsf@cumt.edu.cn.

$$H(A_i) = -\sum_{j=1}^k p(a_{ij}) \log p(a_{ij}) \quad (1)$$

其中, $-p(a_i) \log p(a_i)$ 是属性 a_i 的信息熵。熵值越小的属性粒包含的信息量越大, 我们说这个属性粒越大。

所谓关系, 实质上是指论域中元素之间某种相关性。如果关系 R 是自反的、对称的和传递的, 则称 R 为等价关系。一个属性粒对应一个等价关系, 一个等价关系对应一种对论域的分类方法。

设 R 是论域 U 上关系的全体, 且 $R_1, R_2 \in R$, 若对 $\forall x, y \in U, xR_1 y \Rightarrow xR_2 y$, 说明依据等价关系 R_1 对论域划分得到的等价类是包含在依据等价关系 R_2 对论域划分得到的等价类之中的, 则称 R_1 比 R_2 细, 记做 $R_2 < R_1$ 。这样得到一个等价关系序列: $R_1 < \dots < R_{n-1} < R_n$, R_n 是最细的关系, R_1 是最粗的关系。据此, 给出属性粒粗细的定义。

定义 4(属性粒的粗细) 设 A_1, A_2, \dots, A_n 是论域 U 上的属性粒, 它们对应的等价关系 R_1, R_2, \dots, R_n 是论域 U 上的一个等价关系序列, 且 $R_1 < \dots < R_{n-1} < R_n$, 其中 R_n 是最细的关系, R_1 是最粗的关系, 则称属性粒由粗到细的序列为 $A_1 < \dots < A_{n-1} < A_n$, 其中 A_1 为最粗的属性粒, A_n 为最细的属性粒。

一般来说, 较大的属性粒包含的属性个数越多, 它所对应的等价关系越细, 据此等价关系对论域的划分也越细, 所以属性粒也越细, 反之则越粗。

但是属性粒的粗细归根结底与它的区分能力有关, 与它包含的属性个数没有绝对的关系。区分能力越强则属性粒越细。比如 A_1 和 A_2 是属性集 A 上的两个属性粒, 且 $|A_1| > |A_2|$, 但属性粒 A_1 中包含的冗余属性比较多, 区分能力没有 A_2 强, 因此 A_1 比 A_2 粗。

定义 5(属性的依赖度) 给定一个知识库 $K = (U, S)$, $\forall P, Q \in IND(K)$, 定义

$$r_P(Q) = k = \frac{|pos_P(Q)|}{|U|} = \frac{|\bigcup_{x \in U/Q} P(x)|}{|U|} \quad (2)$$

为知识 Q 依赖于 P 的程度。

3 基于聚类的属性粒化方法

聚类算法的对象可以是样本也可以是属性。如果聚类的对象是样本, 则称为 Q 型聚类分析; 如果聚类的对象是属性, 则称为 R 型聚类分析。属性聚类的目的是将区分能力相似的属性放在一个簇类里面, 每个簇类中依据一定的算法选取代表属性, 代表整个簇类的所有属性, 这些代表属性组成新的属性集合, 相当于去掉部分属性, 属性的粒度就变粗了。

属性聚类的关键问题有 3 个: 1) 属性相似度函数的选取, 适合属性聚类的方法有距离法、相关系数法、夹角余弦法等; 2) 聚类算法的选取, 理论上讲, 只要相似度函数选取合理, 所有 Q 型聚类算法都适用于属性聚类; 3) 聚类之后各属性子集的代表属性的选取, 方法有以聚类中心作为代表属性、以信息熵的大小确定代表属性、按权值提取加权属性。

属性聚类分析的目的是: 通过聚类得到的结果有明确的实际意义, 有较强的分辨力和代表性。

3.1 决策表情况

1. 决策表中属性区分能力

决策表中选用式(2)中的 $r_P(Q)$ 作为属性区分能力的度

量方法, 因为这个度量易于计算、理解, 也适用于各种应用^[6]。系数 $r_P(Q)$ 可以看作是属性 Q 对属性 P 的依赖度。如果 Q 选取的是决策属性 D , 而 P 选取的是条件属性 C , 则 $r_C(D)$ 表示的是决策属性 D 对条件属性 C 的依赖度。通过 $r_C(D)$ 系数的计算, 可以得到决策属性对每个条件属性的依赖度。 $r_C(D)$ 相似的条件属性具有相似的区分能力。

值得注意的是 $r_C(D)$ 反映的仅仅是单个条件属性的区分能力, 并不能反映属性的重要度。而且一个条件属性的 $r_C(D)$ 的大小也不能表示它的区分能力的大小, 因为它与其它条件属性组合而成的属性集合可能区分能力非常强, 也就是即使某一条件属性的 $r_C(D)$ 很小, 或者为 0, 但它对决策表的贡献却不能忽略。

例如表 1 是来自文献[6]的一个决策表。其中仅有一个决策属性 d , 据此表计算出的决策属性对条件属性的依赖度如下。

$$\gamma_a(d) = 3/6$$

$$\gamma_b(d) = 0$$

$$\gamma_c(d) = 0$$

表 1 决策表

U	a	b	c	d
1	2	2	0	1
2	1	2	0	0
3	1	2	0	1
4	0	0	0	0
5	1	0	1	0
6	2	0	1	1

决策属性对单个条件属性 a, b, c 的依赖度分别为 0.5, 0 和 0。这说明属性 b 和属性 c 的区分能力相似, 并不是说明它们对决策表没有贡献, 也不说明它们不重要, 更不能直接作为冗余属性舍去。

2. 属性相似性度量及粒化方法

得到决策属性对单个条件属性的依赖度后, 可以得出单个属性的区分能力(尽管并不能表示区分能力的大小)。如果在此基础之上选用一定的聚类算法, 就可以将区分能力相似的属性聚在一个簇类里面, 在每个簇类中选取代表属性, 组成新的属性集合, 就可以达到属性粒化的目的。

如上文通过计算得出决策属性对单个条件属性 a, b, c 的依赖度分别为 0.5, 0 和 0, 对这 3 个属性依赖度聚类可以得到两类: $\{a\}, \{b, c\}$ 。这表明属性 b 和 c 具有相同的区分能力, 可以在 $\{b, c\}$ 中选取代表属性和属性 a 一起构成整个属性集的粒化后的集合。如果通过最简单的任选其一的方法选择属性 b , 那么属性粒化后的集合就是 $\{a, b\}$; 如果选择 c , 属性粒化后的集合就是 $\{a, c\}$ 。下面通过差别矩阵方法来验证其正确性。

依据决策表 1 可以得到对应的差别矩阵, 如表 2 所列。

表 2 差别矩阵

	1	2	3	4	5	6
1						
2	a					
3	a					
4	a, b	a, b	a, b			
5	a, b, c	b, c	b, c			
6	a, b, c	a, b, c	a, b, c	a, c	a	

表 2 对应的送别函数为:

$$f = a(a \vee b)(a \vee b \vee c)(a \vee b)(a \vee b \vee c)(a \vee b)(b \vee c)(a \vee c)a = ab \vee ac$$

可以看出通过差别矩阵得到最后的约简集是 $\{a, b\}$ 或 $\{a, c\}$, 本节提出的决策表属性粒化方法得到了验证。

3.2 知识表达系统情况

知识表达系统中没有决策属性, 属性粒化方法可以依据属性之间的距离聚类, 然后选取代表属性。这里属性相似度函数的选取至关重要。因为属性相对依赖度易于计算、理解, 也适用于各种应用, 选用它作为属性之间区分能力的度量依据。具体方法有 3 步:

(1) 计算属性之间的相对依赖度, 得到属性相对依赖度关系矩阵;

(2) 依据属性相对依赖度关系矩阵计算每两个属性之间的距离;

(3) 计算每个属性与其它属性距离之和, 将其作为属性区分能力的度量依据。

1. 属性相对依赖度关系矩阵

例如来自文献[8]的知识系统, 如表 3 所列。

表 3 知识系统

U	a	b	c	d
1	0	1	2	0
2	1	2	0	2
3	1	0	1	0
4	2	1	0	1
5	1	1	0	2

通过计算得出:

$$\gamma_b(a) = 3/5, \gamma_c(a) = 0, \gamma_d(a) = 3/5$$

$$\gamma_a(b) = 2/5, \gamma_c(b) = 0, \gamma_d(b) = 1/5$$

$$\gamma_a(c) = 2/5, \gamma_b(c) = 2/5, \gamma_d(c) = 5/5$$

$$\gamma_a(d) = 4/5, \gamma_b(d) = 2/5, \gamma_c(d) = 4/5$$

从而得到属性相对依赖度关系矩阵, 如表 4 所列。

表 4 属性相对依赖度关系矩阵

	a	b	c	d
a	1	3/5	0	3/5
b	2/5	1	0	1/5
c	2/5	2/5	1	1
d	4/5	2/5	4/5	1

2. 属性相似性度量及粒化方法

(1) 依据属性相对依赖度关系矩阵, 可以计算出属性之间的距离:

$$|ab| = \sqrt{(3/5)^2 + (2/5)^2 + (2/5)^2} = \sqrt{17}/5$$

$$|ac| = \sqrt{(3/5)^2 + (1/5)^2 + 1 + (2/5)^2} = \sqrt{39}/5$$

$$|ad| = \sqrt{(1/5)^2 + (1/5)^2 + (4/5)^2 + (2/5)^2} = \sqrt{22}/5$$

$$|bc| = \sqrt{(3/5)^2 + 1 + (4/5)^2} = \sqrt{50}/5$$

$$|bd| = \sqrt{(2/5)^2 + (3/5)^2 + (4/5)^2 + (4/5)^2} = \sqrt{45}/5$$

$$|cd| = \sqrt{(2/5)^2 + (1/5)^2} = \sqrt{5}/5$$

(2) 属性距离聚类

属性 a 与其它属性距离之和为 $|ab| + |ac| + |ad| \approx 3.0115$, 同理属性 b 与其它属性距离之和为 3.5804, 属性 c 与其它属性距离之和为 4.0048, 属性 d 与其它属性距离之和为

2.7268。通过聚类可得到这样的簇类: $\{a, d\}, \{c, b\}$ 。

(3) 依据信息熵的代表属性的选取

选取(2)中得到的属性簇类的代表属性的方法有很多, 如果一个属性簇类中只有一个属性, 则它自然成为代表属性; 如果有两个以上, 则可以根据属性重要度、信息熵等来选取。

这里选用信息熵的方法选取代表属性, 通过计算得到属性 a 的信息熵为 -0.412, 属性 b 的信息熵为 -0.46, 属性 c 的信息熵为 -0.412, 属性 d 的信息熵为 -0.46, 因此选取 d 为属性集 $\{a, d\}$ 的代表属性, 选取 b 为属性集 $\{b, c\}$ 的代表属性。这样得到的最后的属性粒化结果为 $\{b, d\}$ 。

通过差别矩阵方法得到的属性约简集合为 $\{a, b\}$ 或 $\{b, d\}$, 进而算法得到了验证。

4 基于属性区分能力和 AP 聚类的属性粒化方法

AP 算法^[9]是由 Frey 等人于 2007 年在《Science》杂志上提出的一种新的聚类算法, 在人脸图像的识别、手写邮政编码的识别、基因外显子发现以及最优航线的搜索等方面得到了广泛的应用^[10-13]。AGAP 在计算出属性区分能力之后选用 AP 算法对属性的区分能力聚类, 主要原因是 AP 算法把所有的数据点作为潜在的聚类中心, 不需要类数、聚类中心等先验知识, 而我们对属性的区分能力的类数等信息一般知之甚少。

基于属性区分能力和 AP 聚类的属性粒化方法如下。

输入: 知识表示系统 $S = (U, C, V, f)$

输出: 属性集合的约简 RED(C)

Step 1 数据离散化预处理

Step 2 计算属性区分能力

决策表情况, 直接计算决策属性对每个条件属性的依赖度, 转到 Step 3

知识系统情况计算:

(1) 计算属性之间的相对依赖度, 得到属性相对依赖度关系矩阵

(2) 计算属性之间的距离

(3) 计算属性与其它所有属性距离之和

Step 3 使用 AP 算法对属性区分能力聚类, 将区分能力相似的属性聚在一个簇类

Step 4 代表属性的选取

在聚类后的每个属性簇类中选用下列方法之一选取代表属性:

(1) 任选其一作为代表属性

(2) 选取属性重要度大的作为代表属性

(3) 选取信息熵小的作为代表属性

Step 5 每个属性簇类的代表属性构成最后的属性粒化集合

5 实验

下面使用 AGAP 算法与传统的属性约简算法 AR 做比较, 验证算法的正确性和效率。我们使用 UCI 的 4 个数据集验证算法, 数据集特征如表 5 所列, 实验对比结果如表 6 所列。

表 5 UCI 4 个数据集特征

Data set	Number of samples	Number of attributes
Iris	150	4
Glass Identification	213	10
Ionosphere	351	34
Mushroom	5642	22

[11] Xie Zhou, Wang Xian-fan. A fast algorithm for detecting local community structure in complex networks [J]. Computer Simulation, 2007, 24 (11): 82-85

[12] Kernen A, Ott J, Krkkinen T. The ONE simulator for DTN protocol evaluation[C]//The 2nd International Conference on Simulation Tools and Techniques. Rome, Italy, 2009: 1-10

[13] Oh S. An advanced taxi movement model in the working day movement for delay-tolerant networks[J]. Cluster Computing,

2014, 17(3): 751-756

[14] Gewali L, Roman V. Generalization of Shortest Path Map[C]//2010 Seventh International Conference on Information Technology; New Generations (ITNG). IEEE, 2010: 296-300

[15] Rosvall M, Bergstrom C T. Maps of random walks on complex networks reveal community structure[C]//Proceedings of the National Academy of Sciences of the United States of America. 2008: 1119-1123

(上接第 97 页)

首先,我们用 Iris 数据集验证算法。它有 150 个对象,包含 4 个条件属性和一个决策属性,决策属性对条件属性 a_1, a_2, a_3, a_4 的依赖度为 0.2133, 0.1267, 0.7800, 0.6533。使用 AP 算法对属性依赖度聚类,得到 3 类: $\{a_1, a_2\}, \{a_3\}, \{a_4\}$ 。通过选取代表属性, Iris 数据集较粗的属性粒子产生了 $\{a_2, a_3, a_4\}$ 。由此可以看出, AGAP 算法可以产生正确的属性粒化结果,并且其效率大大提高,如表 6 所列。

表 6 AR 算法与 AGAP 算法比较

Data set	Attribute granulation results		Runtime (s)	
	AR	AGAP	AR	AGAP
Iris	$\{a_2, a_3, a_4\}$	$\{a_2, a_3, a_4\}$	0.6875	0.2832
Glass	$\{a_1, a_3, a_5\}$	$\{a_1, a_3\}$	8.0756	2.0312
Identification	$\{a_6, a_7\}$	$\{a_6, a_7\}$	5.2116E+	5.9639
Ionosphere	$\{a_{14}, a_{16}, a_{28}\}$	$\{a_1, a_4, a_{14}, a_{16}, a_{28}\}$	002	
Mushroom	$\{a_2, a_3, a_4, a_6, a_8, a_{10}, a_{13}, a_{14}, a_{15}, a_{16}, a_{21}, a_{22}\}$	$\{a_2, a_3, a_4, a_5, a_6, a_8, a_{10}, a_{13}, a_{14}, a_{15}, a_{21}\}$	7.2308E+	87.3126
			003	

实验表明,对 Iris 数据集, AGAP 算法和 AR 算法有相同的属性粒化结果 $\{a_2, a_3, a_4\}$, 但是 AGAP 算法的效率显然高于传统的属性约简算法 AR。对于 Glass Identification 数据集, AGAP 算法与传统的属性约简算法相比丢失了属性 a_5 , 但是算法效率明显提高。对于 Ionosphere 数据集, AGAP 算法比 AR 算法多了两个属性 a_1, a_4 , 算法结果有了冗余。对于 Mushroom 数据集, 与 AR 算法相比, AGAP 算法丢失了两个属性 a_{16}, a_{22} , 又多了一个属性 a_5 , 虽然算法的结果没达到 100% 的精度, 但是算法运行时间只有原来算法的 1%。对于大数据集来说, 大数据时代降低算法的精度来换取算法的时效性具有重要意义。

结束语 本文提出的 AGAP 算法首先依据属性区分能力得到属性相似性度量, 然后使用 AP 算法对属性聚类, 最后从各属性簇类中依据信息熵方法提取代表属性, 从而达到对属性集合粗粒化的目的。算法效率相比传统属性粒化方法大大提高, 但有时精确度不高, 需要进一步寻找度量属性相似度的合适方法, 以提高算法的准确率。

参 考 文 献

[1] Wang Guo-yin, Yu Hong, Yang Da-chun. Decision table reduction based on conditional information entropy[J]. Chinese Journal of Computers, 2002, 25(7): 759-766 (in Chinese)

王国胤, 于洪, 杨大春. 基于条件信息熵的决策表约简[J]. 计算机学报, 2002, 25(7): 759-766

[2] Xu Zhang-yan, Liu Zuo-peng, Yang Bing-ru, et al. A quick at-

tribute reduction algorithm with complexity of $\max\{O(|C||U|), O(|C|^2|U/C|)\}$ [J]. Chinese Journal of Computers, 2006, 29(3): 391-399 (in Chinese)

徐章艳, 刘作鹏, 杨炳儒, 等. 一个复杂度为 $\max\{O(|C||U|), O(|C|^2|U/C|)\}$ 的快速属性约简算法[J]. 计算机学报, 2006, 29(3): 391-399

[3] Tong C, Ding S F, Zhu H, et al. A Granularity Attribute Reduction Method Based on Binary Discernibility Matrix[J]. IJACT; International Journal Advancements in Computing Technology, 2012, 4(12): 213-221

[4] Qian Y H. Positive approximation: an accelerator for attribute reduction in rough set theory[J]. Artificial Intelligence, 2010, 174(9-10): 597-618

[5] Wang Sa, Zheng Lian. Feature selection method based on fisher criterion and feature clustering [J]. Computer Applications, 2007, 27(11): 2812-2840 (in Chinese)

王飒, 郑链. 基于 Fisher 准则和特征聚类的特征选择[J]. 计算机应用, 2007, 27(11): 2812-2840

[6] Ma Xi-ao, Wang Guo-yin, Yu Hong. Heuristic method to attribute reduction for decision region distribution preservation [J]. Journal of software, 2014, 25(8): 1761-1780 (in Chinese)

马希霄, 王国胤, 于洪. 决策域分布保持的启发式属性约简方法[J]. 软件学报, 2014, 25(8): 1761-1780

[7] Ji Su-qin, Shi Hong-bo, Lv Ya-li. An attribute reduction algorithm based on granular computing and discernibility [J]. PR&AI, 2015, 28(4): 327-334 (in Chinese)

冀素琴, 石洪波, 吕亚丽. 基于粒计算与区分能力的属性约简算法[J]. 模式识别与人工智能, 2015, 28(4): 327-334

[8] Zhang Wen-xiu, Wu Wei-zhi, Liang Ji-ye, et al. Rough set theory and method[M]. Beijing: Science Press, 2001 (in Chinese)

张文修, 吴伟志, 梁吉业, 等. 粗糙集理论与方法[M]. 北京: 科学出版社, 2001

[9] Frey J, Dueck D. Clustering by passing messages between data points[J]. Science, 2007, 315(5814): 972-976

[10] Jia S, Qian Y T, Ji Z. Band selection for hyperspectral imagery using affinity Propagation[C]//Proceedings of the 2008 Digital Image Computing: Techniques and Applications. Canberra, ACT: IEEE, 2008: 137-141

[11] Li G, Guo L, Liu T M, et al. Grouping of brain MR images via affinity propagation [C]// IEEE International Symposium on Circuits and Systems, 2009 (ISCAS 2009). Taipei: IEEE, 2009: 2425-2428

[12] Dueck D, Frey B J, Jojic N, et al. Constructing treatment portfolios using affinity propagation[C]//Proceedings of 12th Annual International Conference (RECOMB 2008). Berlin: Springer, 2008: 360-371

[13] Kelly K. Afinity Program Slashes Computing Times[EB/OL]. (2007-02-15). <http://www.news.utoronto.ca/bin6/070215-2952.asp>