

基于网格的量子博弈聚类算法

黄德才 汤胜龙

(浙江工业大学计算机科学与技术学院 杭州 310023)

摘 要 量子博弈是对经典博弈的量子模拟,利用量子的纠缠态,可以使博弈参与人在博弈策略的选择过程中相互影响,从而得到与经典博弈不同的结果。将量子博弈运用于聚类问题,并提出一种基于网格的量子博弈聚类算法。算法将数据点看作是博弈的参与人,通过在收益矩阵中内嵌距离函数,使相似的数据点能够获得更大的收益,从而形成聚类。此外,通过设定网格合并规则,使博弈过程得到了简化。仿真实验表明,算法在聚类质量上优于传统的 K-means 等算法。最后,就算法中的几个参数对算法性能的影响进行了讨论,并给出了参数选择的建议。

关键词 博弈论,量子博弈,网格,聚类

中图分类号 TP18 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2014.10.055

Clustering Algorithm Based on Quantum Game and Grid

HUANG De-cai TANG Sheng-long

(College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310023, China)

Abstract Quantum game is an analogy of classical game. Using the quantum entanglement, game players interact with each other implicitly, and the game will result in a different way. Quantum game was applied to clustering. A clustering algorithm based on quantum game and grid was proposed where data points are regarded as players. By embedding distance function into payoff matrix, similar data points can get more payoff, and clusters will be formed in that way. In addition, a rule about merging grid was designed to simplify the game. Simulations show the clustering quality of this algorithm is superior to K-means etc. At last, several parameters in this algorithm were discussed and some recommendations about parameters selection were provided.

Keywords Game theory, Quantum game, Grid, Clustering

1 引言

聚类分析是一个非常活跃的研究领域,其基本思想是在数据集中根据数据间的相似度,将相似的归为一类,不相似的则不归类。聚类分析在数据挖掘、机器学习、模式识别等领域有着广泛的应用。常见的聚类算法可以分为以下几类:划分方法、层次方法、基于密度的方法、基于网格的方法等,它们各自的代表算法分别有 k-means 算法、BIRCH 算法、DBSCAN 算法、STING 算法等^[1]。但随着大规模复杂数据集的出现,传统的聚类算法受到了挑战。处理这些数据集,则需要能够提供更高的可靠性、有效性以及处理高维数据能力的新聚类算法。

另一方面,量子博弈是量子计算中一个很有吸引力的计算模型,它是用量子化的方式模拟博弈论的过程。经典博弈论中,要求博弈参与人必须是完全理性甚至是超理性的。然而,有限的信息与局限的意识常常使得理性的决策是不可能的。此外,完全理性还可能导致无限重复博弈中 backward induction 悖论的出现^[2]。而在实际中,博弈参与人在考虑收益时总是需考虑到周边人的收益,因为自身的收益与其他人

的收益是相互关联而非独立的,这与量子力学中的纠缠态一致。

量子博弈利用量子纠缠态,使博弈双方在使用量子策略时隐含地相互影响,从而获得与经典博弈不同的结果。量子博弈扩大了经典博弈的策略集并引入了纠缠态。策略集的扩大使得寻找满足条件的策略更加容易^[3]。纠缠态的引入从一定程度修正了“个体都是理性的”这一基本经典假设。互相纠缠的状态成为博弈个体之间相互“关照”程度的度量,正如人群中的亲友之间进行博弈时,会把对方的收益部分当作自己的收益一样。

基于量子博弈理论设计聚类算法是一个非常前沿的研究领域,目前仍处于起步阶段。当前的研究已经显现出了量子博弈聚类算法较传统聚类算法有巨大优势。2009年,浙江大学李强博士在其学位论文中建立了一种基于有向图的量子博弈聚类模型,此模型将每个待聚类的数据点看作是博弈的参与人,得到的聚类精度明显高于传统 K-means 等聚类算法^[4]。但是基于图的量子博弈聚类模型,在每次博弈的过程中需要各个参与人都参与博弈,此过程需要不断地进行断边重连函数的计算,过程繁琐,并且由于数据点邻域的固定导致

到稿日期:2013-12-09 返修日期:2014-04-12

黄德才(1958—),男,教授,主要研究方向为数据仓库与数据挖掘、人工智能、量子计算、网格调度等,E-mail: hdc@zjut.edu.cn; 汤胜龙(1988—),男,硕士生,主要研究方向为数据挖掘、量子计算等。

一次博弈探索深度不够,从而影响聚类效果。本文在研究量子博弈内在机理的情况下,提出一种基于网格的量子博弈聚类算法,亦即将数据点自然分布在网格中,计算数据点从它所有邻居中获得的收益,通过设定一定的网格合并条件,使得在聚类的过程中对外表现相同的网格能够合并。算法在简化聚类过程的情况下进一步优化了聚类效果。

2 博弈论及其量子化

2.1 经典博弈论

博弈论,又称对策论,或游戏理论,是应用数学的一个分支。它在经济学、社会学、生物科学以及计算机科学等领域都有很广泛的应用^[2]。

一个博弈可以设为 $Q=\{X,S,U\}$, 它的 3 个组成部分是:

(1)能够进行决策的参与人集合 $X=\{X_1, X_2, \dots, X_n\}$ 。

(2)参与人的策略集 $S=\{s(i), i=1, 2, \dots, n\}$ 。参与人的策略就是一个与参与人的行动相关的规则或函数,这是就他当时所掌握的信息而言,最好的行动选择;

(3)参与人的收益集 $U=\{u(i)=\pi_i(s(1), \dots, s(n)), i=1, 2, \dots, n\}$ 。收益就是在参与人本身与其他参与人所选择的策略都确定的情况下,他所得到的博弈的结果。通常将收益通过收益矩阵的形式列出。

囚徒困境(Prisoner's Dilemma)是一个著名的博弈案例,它描述的情形是:两个嫌疑犯 A 和 B 被分别带到独立的房间里问话,如果他们都坦白,那么每个人被判三年徒刑;如果其中一个人坦白,另一人抵赖,那么坦白者将被判五年徒刑,而抵赖者将无罪释放;如果两人都抵赖,那么每人都判一年徒刑。

在这个博弈中,参与人集合是{嫌疑犯 A, 嫌疑犯 B}, 他们所拥有的策略集是{坦白 C, 抵赖 D}, 对于所选择的策略, 双方的收益矩阵可以用表 1 表示。

表 1 囚徒困境收益矩阵

	嫌疑犯 B, 选择 C	嫌疑犯 B, 选择 D
嫌疑犯 A, 选择 C	$U(A)=U(B)=3$	$U(A)=0, U(B)=5$
嫌疑犯 A, 选择 D	$U(A)=5, U(B)=0$	$U(A)=U(B)=1$

可以发现,在这个博弈中,不管对方选择什么策略,个人的最优选择都是坦白 C。所以双方都选择坦白就是一个 Nash 平衡点。Nash 平衡点是一个策略对,即游戏的任何一方单方面调整自己的策略后,都不能获得更大的收益。但事实上,双方都选择抵赖是对整体更有利的策略,所以两个囚徒就陷入了“为集体考虑”和“为自己考虑”的困境中^[5]。

2.2 量子囚徒困境模型

J. Eisert 最早提出囚徒困境的量子化形式,他的模型主要有 3 个组成部分^[6],如图 1 所示。

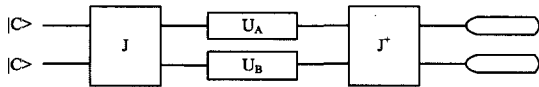


图 1 囚徒困境的量子化模型

(1)两位量子比特(qubit),每一位属于一个博弈参与人;

(2)参与人的操作装置,参与人可以操作属于他自己的那一个比特,这些操作实际上就是参与人的策略;

(3)一套物理测量装置,用以从博弈的终止态中测量出每个人的收益。

设定两个博弈的参与人都知道上述 3 点。

用 $|C\rangle=(1,0)^T$ 和 $|D\rangle=(0,1)^T$ 分别表示合作和抵赖。首先,将两个 qubit 都制备成合作态,即 $|CC\rangle$ (前一个字母为 A 的 qubit 状态,后一个字母为 B 的 qubit 状态)。通过一个二位量子门 J 使得状态 $|CC\rangle$ 纠缠在一起形成初始状态 $|\psi_i\rangle=J|CC\rangle$ 。A 和 B 的博弈策略是酉矩阵 U_A 和 U_B , 它们分别代表 A 和 B 对各自 qubit 的操作。 J^+ 是 J 的共轭算符,在 J^+ 之后是输出装置,用来计算 A 和 B 博弈后的收益。当 A 和 B 进行一次博弈之后,最终状态变成了^[7]:

$$|\psi_f\rangle=J^+U_A\otimes U_BJ|CC\rangle$$

其中, $\otimes$ 为直积符号。

根据末态的观察结果,可以得到博弈的最终收益。例如,若末态为 $|\psi_f\rangle=|CD\rangle$, 则 $U(A)=0, U(B)=5$ 。对于一般的末态, A 收益的期望值为 $U(A)=rP_{CC}+pP_{DD}+tP_{DC}+sP_{CD}$, B 的收益期望值为 $U(B)=rP_{CC}+pP_{DD}+sP_{DC}+tP_{CD}$, 其中 $r=3, p=1, t=5, s=0$, 而 $P_{\sigma\sigma'}=|\langle\sigma\sigma'|\psi_f\rangle|^2$, 其中 $\sigma, \sigma'=C, D$ 。

如果限定 A 和 B 的策略是 2×2 的酉矩阵, 形如

$$U(\theta, \phi)=\begin{bmatrix} e^{i\phi}\cos\frac{\theta}{2} & \sin\frac{\theta}{2} \\ -\sin\frac{\theta}{2} & e^{-i\phi}\cos\frac{\theta}{2} \end{bmatrix}$$

其中, $0\leq\theta\leq\pi, 0\leq\phi\leq\pi/2$ 。

所以,坦白和抵赖的算符分别为:

$$\hat{C}=U(\pi, 0)=\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}, \hat{D}=U(0, 0)=\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

$J=\exp\{i\gamma D\otimes D/2\}$, $0\leq\gamma\leq\pi/2$ 。 γ 标识了 A 和 B 纠缠度的大小,当 $\gamma=0$ 时, A 和 B 之间无纠缠;当 $\gamma=\pi/2$ 时, A 和 B 之间的纠缠度达到最大^[6]。此时, DD 不再是 Nash 平衡点, 双方都选择 $U(0, \pi/2)=\begin{pmatrix} i & 0 \\ 0 & -i \end{pmatrix}$ 成为新的 Nash 平衡点。

3 基于网格的量子博弈聚类算法

文献[4]提出了一种基于图的量子博弈聚类算法,该算法将数据点看作有一定决策能力的博弈参与人。每个参与人分别与邻域中的每个邻居进行 2×2 纠缠量子博弈,并且参与人都希望最大化自己的收益。在博弈的过程中,参与人不断观察邻居的收益,断掉与低收益邻居的链接,同时建立与高收益邻居的链接。每轮博弈过后,它利用量子搜索算法中的 Grover 迭代对与邻居间的链接强度进行更新。在不断进行量子博弈的过程中,参与人网络的结构和他与邻居的链接强度都将逐渐趋于稳定。此时,如果让每个参与人只与有最大链接强度的邻居相连,那么参与人的网络将自然地分成几个分离的部分,每一部分就对应了一个聚类^[4]。

但是基于图的量子博弈聚类算法需要每次更新链接强度,并且在博弈的过程中,每一次都需要全体数据点的参与,过程复杂,且参数难以控制。基于以上考虑,本文提出一种基于网格的量子博弈聚类算法。

3.1 网格的构造以及量子邻域

首先定义网格空间以及量子邻域。考虑二维的情况,设有一批二维数据点 (X_i, Y_i) , 令 $d_X=X_{\max}-X_{\min}$ ($X_{\max}$ 表示 X 方向的最大坐标, $X_{\min}$ 表示 X 方向的最小坐标)。所以 d_X 即为 X 方向的最大坐标差,同样可以定义出 d_Y 。采用某种策略

对 d_X, d_Y 做一定的扩张得到 d_X', d_Y' 。扩张方式视具体问题和数据集而定。如当划分的网格数为 k 时, 可以用 $d_X' = d_X(1+1/k)$ 来进行扩张。这样扩张以后, 以 d_X', d_Y' 为长宽建立的网格将包含所有的数据点。再按需求对这个网格进行划分, 使每个数据点都有各自的归属网格。其中, 需注意的是处于网格线上的数据点归属于共线的两个网格, 处于网格点上的数据点归属于共点的 4 个网格。按此规则可以将构成网格的方法推广至 n 维的情况。

网格划分好后, 每个数据点都可以确定各自的归属网格。设一个数据点的邻域为与其归属网格共面、共线或共点的所有网格, 这个邻域也可以称作是该归属网格的邻域。还是考虑二维的情况, 如图 2 所示, X_i 是一个数据点, 它的邻域即为其所在网格共线的网格 2、4、5、7 以及共点的网格 1、3、6、8。

1	2	3
4	X_i	5
6	7	8

图 2 二维情况下一个数据点的邻域

为了使任意网格中的数据点具有相同的博弈环境, 假设网格空间的上下边界相连, 左右边界相连。高维情况下, 每一维的边界相连。我们可以用这种方法来模拟无限的网格环境^[8]。

3.2 博弈方式

我们可以用一个四元组来定义网格 G 上的一个量子博弈 $QG = \{G, X, S, U\}$: 除 G 表示划分好的网格外, 其中 $X = \{X_i, i=1, 2, \dots, N\}$ 表示参与人的集合; $S = \{S(i), i=1, 2, \dots, N\}$ 表示参与人的策略集; $U = \{U(i), i=1, 2, \dots, N\}$ 表示参与人的收益集。

在每轮博弈开始时, 参与人从策略集中选取博弈策略, 一次博弈结束后, 每个参与人都计算各自的收益。并且每个参与人可以观察到邻域内各个邻居的收益。

在经典的 2×2 博弈中, 参与人的策略通常为两个纯策略, 但在量子博弈中, 不同的酉矩阵均可以作为参与人的策略, 即每个参与人的策略集可以为 2 阶酉矩阵集中的某个子集^[9]。本文为简化计算, 将参与人的策略集限定在 $S = \{H, \hat{D}\}$ 中。参与人从策略集中选择一个策略作为博弈的策略, 与其邻域内的邻居进行 2×2 纠缠量子博弈。假设博弈的初始态为 $|\phi_0\rangle = J|00\rangle$, 在一次博弈过后的博弈终止态为:

$$|\psi_{f,j}\rangle = \begin{cases} \frac{1}{2}(|00\rangle - i|01\rangle - i|10\rangle + |11\rangle), & H \otimes H \\ \frac{1}{\sqrt{2}}(i|00\rangle - |01\rangle), & H \otimes D \end{cases}$$

3.3 收益的计算

雪堆博弈模型: 在一个风雨交加的夜晚, 两人相向而行, 被一个雪堆挡住。假设铲除雪堆所需的劳动力为 c , 铲除后每个人能够获得回家的收益为 b 。所以雪堆博弈的收益矩阵如表 2 所列。

表 2 采用类雪堆模型时数据点 X_i 在 X_j 处的收益矩阵

C=0		D=1	
C=0	$R = w_{t-1}(X_i, X_j) - c/2$	S=	$w_{t-1}(X_i, X_j) - c$
D=1	$T = w_{t-1}(X_i, X_j)$	P=	$0.01w_{t-1}(X_i, X_j)$

本文采用类雪堆模型的收益矩阵来计算最后的收益。在类雪堆模型的收益矩阵中, $w_{t-1}(X_i, X_j) = \rho(X_j)/d(X_i, X_j)$, 表示在一次博弈后, 参与人 X_i 从参与人 X_j 处获得的收益。 $\rho(X_j)$ 表示 X_j 所处网格的规模, 即所含数据点的总数; $d(X_i, X_j)$ 是定义在网格空间内的一个距离函数, 它满足两个数据点越接近, 函数值越小。

收益之间满足关系 $T > R > S > P$ 。雪堆模型中有一个很重要的概念: 损失因子 $\beta = c/w_{t-1}(X_i, X_j)$, 它表示铲除雪堆所需的劳动力与回家获得的收益之间的比值。 β 的大小对最后博弈的收益会产生影响^[10]。在本文的聚类算法中, 调整 β 值可以使聚类得到不同的精度, 这在今后将会进一步讨论。

3.4 算法步骤

算法的输入: 聚类数目 K , 数据集。输出: 聚类结果 ($C_1, C_2, \dots, C_K$)。

算法步骤为:

Step 1 初始化 m 维网格 (m 为数据点的维度), 并将数据点自然映射在网格上。在此模型中, 为了使落在任意网格中的数据点具有相同博弈环境, 假设量子博弈空间的上下、左右边界相连(此描述为 2 维的情况, 也可推广至更高维)。

Step 2 在此空间中定义一个距离函数 d , 它满足两个数据点越靠近, 函数值越小。

Step 3 检查含数据点的网格数目 M 与聚类数目 K 的关系, 若 $M > K$, 则执行 Step 4; 若 $M = K$, 则执行 Step 7。

Step 4 寻找含数据点的网格中数据点数目最少的网格, 设为 a 。若 a 的邻域中没有数据点, 则将其邻域都并入 a 。此后, 再判断, 若合并后的 a 所占面积占总面积达到一定比例(一般为 10% 左右), 则将该网格视为潜在孤立点网格。否则, 执行 Step 5。

Step 5 若 a 的邻域中有数据点, 则 a 中的所有数据点分别与邻域内的所有数据点进行一次博弈, 博弈后分别计算 a 中数据点相对邻域中每个网格的总收益。

Step 6 网格合并: 一次博弈之后, a 中的每个数据点根据 Step 5 中计算出的收益决定归属, 整个网格最后按少数服从多数的原则决定合并到某个邻域网格中。合并后跳回 Step 3。

Step 7 当含数据的网格数达到聚类数时, 若网格相邻, 则相邻网格中的数据点再与邻域网格中的数据点进行一次博弈以确定最终的归属网格; 若不相邻, 则算法结束, 得到相应的聚类结果。

4 仿真实验及参数分析

4.1 实验条件

为了评估上述算法, 本文使用 weka 工具, 并从 UCI 数据集中选择了 2 个典型的数据集 Iris (4 维) 和 Ionosphere (32 维) 来进行仿真研究, 它们分别代表较低维和较高维的数据。这两个数据集的情况如表 3 所列。

表 3 Iris 和 Ionosphere 数据集的情况

数据集	数据个数	维度	标准类数
Iris	150	4	3
Ionosphere	351	32	2

博弈方式: 2×2 纠缠量子博弈。参与人的策略集合 S 限定在集合 $S = \{H, \hat{D}\}$ 中; 每一维的网格数设为 150; 收益模型

为类雪堆模型,雪堆模型中损失因子 β 设为0.25。在距离函数的选择上,我们考虑广义加权 Minkovski 距离^[11]:

$$d_G(x, y) = [\sum_j (\frac{1}{d_i f_j} |x_j - y_j|^p)]^{\frac{1}{p}}$$

该距离函数满足本算法对距离函数的要求,并且克服了欧氏距离对高维数据的距离度量不准确的弊端。取范数 $p=1$,属性值分位数 $q=0.125$ 。

本文采用聚类精度(PE)、正确率(AC)以及召回率(RE)3个指标来评估算法的聚类质量,它们的计算式分别为:

$$PE = \frac{\sum_{i=1}^k \frac{a_i}{|a_i + b_i|}}{k}, AC = \frac{\sum_{i=1}^k a_i}{n}, RE = \frac{\sum_{i=1}^k \frac{a_i}{|a_i + c_i|}}{k}$$

其中, a_i 表示应该分到 i 类,且正确分到 i 类的数据点数目; b_i 表示不应该分到 i 类但是被分到了 i 类的数据点数目; c_i 表示应该分到 i 类却没有分到 i 类的数据点数目; n 表示数据集所有的数据点数目; k 为聚类的数目^[12]。

4.2 实验结果分析

在上述实验设定下,分别对本文算法与经典的 K-means 算法以及基于图的量子博弈算法进行20次运算,并计算聚类精度(PE)、正确率(AC)、召回率(RE)的平均值,结果如表4—表6所列。

表4 几种算法的聚类精度(PE)比较

实验数据集	K-means	基于图的量子博弈算法	本文算法
Iris	86.51%	90.13%	92.36%
Ionosphere	72.22%	74.03%	75.61%

表5 几种算法的正确率(AC)比较

实验数据集	K-means	基于图的量子博弈算法	本文算法
Iris	84.73%	88.65%	91.97%
Ionosphere	71.36%	74.62%	76.21%

表6 几种算法的召回率(RE)比较

实验数据集	K-means	基于图的量子博弈算法	本文算法
Iris	83.45%	88.19%	90.68%
Ionosphere	68.33%	72.60%	75.42%

从上述实验结果可以看出,本文算法在两个数据集上的3个聚类指标值都高于传统的 K-means 算法和基于图的量子博弈聚类算法。Iris 数据集数据维度较低,全局性较为简单,本文算法在此数据集上的聚类质量较传统的 K-means 算法有明显的提升。基于图的量子博弈算法在 Iris 数据集上的聚类效果已经较好,每个指标值都在90%左右,本文算法在此基础上还能将各个指标值都提升2个百分点以上。

高维数据的聚类一直是聚类问题中的难点。Ionosphere 数据集有32个属性,属于较高维的数据。本文算法在此数据集上也取得了不错的聚类结果,每项指标值都是3个算法中最高的,其中召回率高出基于图的量子博弈聚类算法将近3个百分点,说明本文算法在处理高维数据时具有明显优势。

4.3 算法参数改变的影响

本节主要讨论算法若干参数改变时对最后聚类结果的影响。

在本文算法中,网格的分辨率改变时,每个网格中数据点的个数会改变,从而影响整个数据集在网格中的分布。此外,当采用类 SD 模型的收益矩阵时,由于损失因子 c 的存在,算法的整体性能也会受到影响。本节将采用控制变量法对以上两个参数对算法的影响进行讨论,并由实验结果对两个参数的选值给出建议。

4.3.1 网格分辨率对聚类精度的影响

实验数据集:Iris;数据个数为150;数据维度为4;标准类数为3。博弈方式: 2×2 纠缠量子博弈。参与人的策略集合 S 限定在集合 $S = \{H, D\}$ 中。收益模型:使用类雪堆模型计算收益,雪堆模型中损失因子设为0.25。实验结果如图3所示。

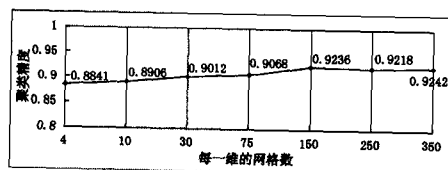


图3 不同网格划分下的聚类精度

实验结论:通过对比多种网格模式下的聚类精度可以发现,在一定的区间范围内,随着网格分辨率的提高,聚类精度也有所上升;但在达到一定程度后,精度趋于稳定。并且从实验中发现,将每一维网格的个数用数据点的数目来确定是合适的。

4.3.2 类 SD 收益矩阵中损失因子 c 对聚类精度的影响

实验数据集:Iris;数据个数为150;数据维度为4;标准类数为3。博弈方式: 2×2 纠缠量子博弈。参与人的策略集合 S 限定在集合 $S = \{H, D\}$ 中。网格分辨率:每一维网格数为150。实验结果如图4所示。

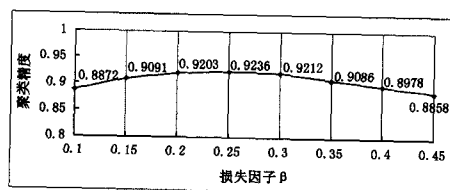


图4 损失因子 β 取不同值时的聚类精度

实验结论:当收益矩阵中的损失因子太大时,类 SD 收益矩阵将退化为类 PD 收益矩阵^[10],所以算法中将损失因子限定在区间(0,0.5)。从实验结果来看,将损失因子的取值限定在(0.25,0.3)能够得到较好的结果。

结束语 量子计算是目前科研领域的研究热点,运用量子计算能够对传统算法进行加速。量子计算与博弈理论相似,都是以概率论作为基础的。鉴于这两种研究课题的特点,此前曾有将二者结合的研究,并提出图上的量子博弈聚类算法。本文借鉴图上量子博弈的部分思想,提出一种基于网格的量子博弈模型,使博弈的过程在得到简化的同时也提升了算法的聚类精度。

本文还对算法中出现的两个参数(网格分辨率以及雪堆收益模型中的损失因子)对聚类精度的影响进行了讨论,给出了这两个参数选值的建议。

从最后的实验结果来看,量子博弈在聚类上是很有优势的。将量子博弈运用到聚类中是一个非常有价值的研究方向。此后的研究可以考虑从以下几个方向开展:

(1)本文算法中,采用的是类雪堆模型的收益矩阵,此模型中需要考虑损失因子对整个算法性能的影响,并且这种影响还会随数据集的变化而变化。虽然可以通过参数的调整来增强算法的灵活性,但这也在一定程度上增加了工作量。后续的研究可以寻找其它收益模型来替代,如类囚徒困境模型、类性别大战模型等。

(2)区别于传统的 2×2 博弈,在量子博弈中,所有2阶的

西矩阵都可以作为参与人的策略,本文为了简化运算,将参与人的策略做了限定。策略矩阵不同,博弈终止态也会不同,从而影响收益和最后的聚类精度。所以可以尝试使用其它西矩阵作为参与人的策略,以观察结果。

(3)本文只考虑了两人两策略的最简博弈模型,后续研究可以考虑多人同时参与博弈的情形^[13],如3个邻居同时参与。这样,在量子情形下,每个人的收益都将受到多个邻居的影响,博弈的收益矩阵需要重新设计,博弈终止态也将发生变化,从而得到不同的聚类结果。

参考文献

- [1] 孙吉贵,刘杰,赵连宇. 聚类算法研究[J]. 软件学报,2008,19(1):48-61
- [2] 焦宝聪,陈兰平,方海光. 博弈论-思想方法及应用[M]. 北京:中国人民大学出版社,2013:118-121
- [3] Meyer D A. Quantum strategies[J]. Physical Review Letters, 1998,82(5):1052-1055
- [4] 李强. 动点聚类算法及其量子化研究[D]. 杭州:浙江大学电气工程学院,2009

- [5] 王龙,王靖,武斌. 量子博弈:新方法与新策略[J]. 智能系统学报,2008,3(4):294-304
- [6] Eisert J, Wilkens M, Lewenstein M. Quantum games and quantum strategies[J]. Physical Review Letters A, 1999, 83(15): 3077-3080
- [7] 李威,赵红敏,林家逖. 量子博弈论及其应用[J]. 大学物理, 2003,22(12):3-8
- [8] 钱国红. 量子算法及其在数据挖掘中的应用[D]. 杭州:浙江工业大学,2012
- [9] Guo Hong, Zhang Ju-heng, Koehler G J. A survey of quantum games[J]. Decision Support Systems, 2008, 46(1): 318-332
- [10] Hauert C, Doebeli M. Spatial structure often inhibits the evolution of cooperation in snowdrift game[J]. Nature, 2004, 428: 643-646
- [11] 钱国红,黄德才,陆亿红. 广义加权 Minkovski 距离及量子遗传聚类算法[J]. 计算机科学,2013,40(5):224-228
- [12] 赵恒,杨万海. 模糊 K-Modes 聚类精确度分析[J]. 计算机工程, 2003,29(12):27-28
- [13] 周杰,李筠,马雷. 任意多人的量子博弈[J]. 量子电子学报, 2006,23(2):173-177

(上接第251页)

故A的约简为 $\{a_1, a_3\}$ 与 $\{a_2, a_3\}$ 。

由定义4和定义5求得的约简与例3的结论完全一样。

一般地,我们有以下定理。

定理9 当 $C_R=C_r$ 时,由定义1、定义4和定义5所定义的3种属性约简等价。

证明:我们只需证明下面3个条件等价:

(1) $B=\{e_1, \dots, e_n\}, e_i \in E_i, E_i \in A/r(i=1, \dots, n)$ 。

(2) B 是 A 所在的 R 等价类的极小元。

(3) B 是 $M=\{D_1 \subseteq A, D_1 \cap D \neq \emptyset, \forall D \in \overline{C_R}\}$ 的极小元。

其中 $\overline{C_R}=\{D^c | D \in C_R, D \neq A\}$ 。

(1) $\Rightarrow$ (2):若 $B=\{e_1, \dots, e_n\}$,则 $R_B=R_{\bigcup_{i=1}^n E_i}=\bigcap_{i=1}^n R_{E_i}=R_{\bigcup_{i=1}^n E_i}=R_A \subseteq R_d$ 。所以 $(A, B) \in R$ 。如果 B 不是 A 所在的 R 等价类 $B \in B(A)/R$ 的极小元,则存在 $B_1 \subseteq B$ 且 $B_1 \neq B$,使 B_1 为 B 的极小元。不妨设 $B_1=\{e_{i_1}, \dots, e_{i_m}\}$ 。则 $R_{B_1}=R_{\bigcup_{j=1}^m E_{e_{ij}}}=R_{\bigcap_{j=1}^m R_{E_{ij}}}=R_{\bigcup_{j=1}^m E_{ij}}$ 。又 $R_A=R_{B_1} \subseteq R_d$,所以 $(\bigcup_{j=1}^m E_{ij}, A) \in R$ 。

但 $\bigcup_{j=1}^m E_{ij} \neq \bigcup_{i=1}^n E_i=A$,这与定理7矛盾。故 B 是 A 所在的 R 等价类 $B \in B(A)/R$ 的极小元。

(2) $\Rightarrow$ (1):若 B 是 A 所在的 R 等价类 $B \in B(A)/R$ 的极小元, $B=\{e_1, \dots, e_n\}$,则 $R_{B_1}=R_{\bigcup_{i=1}^n E_i}=R_A$ 。所以 $(B_1, A) \in R$,从而 $B \subseteq B_1$ 。假设 $B \neq B_1$,则 B 为 B_1 的真子集。不妨设 $B=\{e_{i_1}, \dots, e_{i_m}\}$,则 $R_B=R_{\bigcup_{j=1}^m E_{e_{ij}}}=R_{\bigcap_{j=1}^m R_{E_{ij}}}$ 。所以 $(\bigcup_{j=1}^m E_{ij}, B) \in R$ 。又 $\bigcup_{j=1}^m E_{ij} \neq \bigcup_{i=1}^n E_i$,由定理7知 $(\bigcup_{j=1}^m E_{ij}, A) \notin R$,所以 $(B, A) \notin R$ 。这与已知条件矛盾,因此 $B=\overline{B_1}=\{e_1, \dots, e_n\}$ 。

(1) $\Leftrightarrow$ (3):当 $C_R=C_r$ 时, $\overline{C_R}=\overline{C_r}=\{D^c | D \in C_r, D \neq A\}$,而 $C_r=\{\bigcup_i E_i | E_i \in A/r\}$ 。于是 $B=\{e_1, \dots, e_n\} \Leftrightarrow B$ 是 $\{D_1 \subseteq A | D \cap E_i \neq \emptyset\}$ 的极小元 $\Leftrightarrow B$ 是 $\{D_1 \subseteq A | D \cap D_1 \neq \emptyset, \forall D \in C_r, D \neq \emptyset\}$ 的极小元 $\Leftrightarrow B$ 是 $M=\{D_1 \subseteq A | D \cap E_i \neq \emptyset\}$ 的极小元。

结束语 本文另辟新径,对于协调信息系统,在其条件属性集而非对象集的幂集上定义了两个闭算子,讨论了由它们所生成的两个闭集族的性质,提出了属性约简的一种新方法。给出了 $C_r=C_R$ 的充要条件,并在此条件下证明了本文所提出的新方法已有结论的等价性。利用这些结果,可以把对信息系统的对象集的幂集的研究转移到条件属性集上来,从而在一定条件下简化了信息系统的知识发现。

参考文献

- [1] 张文修,吴伟志,梁吉业,等. 粗糙集理论与方法. 北京:科学出版社,2001
- [2] 王珏,王任,苗夺谦,等. 基于 Rough set 理论的“数据浓缩”[J]. 计算机学报,1998,21(5):393-400
- [3] 王国胤. 粗糙集理论与知识获取[M]. 西安:西安交通大学出版社,2001
- [4] 张文修,梁怡,吴伟志. 信息系统与知识发现[M]. 北京:科学出版社,2003
- [5] Kryzkievich M. Comparative study of alternative types of knowledge reduction in inconsistent systems[J]. International Journal of Intelligent Systems, 2001, 16: 105-120
- [6] 张文修,米据生,吴伟志. 不协调目标信息系统的知识约简[J]. 计算机学报,2003,26(1):12-18
- [7] Novotny M. Dependence spaces of information systems[C]// Ortowska E, et al. Incomplete informations: Rough set analysis. Physica-Verlag, 1998: 193-246
- [8] Novotny M, Pawlak Z. On a problem concerning dependence spaces[J]. Fundamenta Informaticae, 1992, 16: 275-287
- [9] Ling Wei, Zhang Wen-xiu. Attribute reduction based on equivalence relation defined on attribute set and its power set[C]// International workshop on monitoring, security and rescue techniques in multiagent systems MSRAS, 2004: 317-326
- [10] Wei Ling, Li Hong-ru, Zhang Wen-xiu. Knowledge reduction based on the equivalence relations defined on attribute set and its power set[J]. Information Sciences, 2007, 177: 3178-3185