加权模糊粗糙约简

范星奇[1] 李雪峰[2] 赵素云[1] 陈 红[1] 李翠平[1]
(中国人民大学信息学院 北京 100872)[1] (中国人民大学环境学院 北京 100872)[2]

摘 要 基于模糊粗糙集的传统约简算法的时间代价较高,在处理大规模数据时耗时过长,且在许多实际大规模数据集上存在有限时间内无法收敛等问题。因此将权重引入属性约简的定义中,其中属性权重是属性重要度的数值指标。通过构建优化问题来求解属性权重,证明了属性依赖度即是属性权重的最优解。因此,提出了基于属性权重排序的约简算法,从而大大提升了约简的速度,使得约简算法可以应用于大规模数据集,特别是高维数据集中。

关键词 模糊粗糙集,属性约简,权重,高维数据
中图法分类号 TP311 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2018.01.022

Weighted Attribute Reduction Based on Fuzzy Rough Sets

FAN Xing-qi[1] LI Xue-feng[2] ZHAO Su-yun[1] CHEN Hong[1] LI Cui-ping[1]
(College of Information, Renmin University of China, Beijing 100872, China)[1]
(College of the Environment, Renmin University of China, Beijing 100872, China)[2]

Abstract Now the existing classical reduction algorithms have high time consumption, especially on the large scale datasets. To handle this problem, this paper introduced weights into the concept of attribute reduction, where weight is the measure of attribute significance. By building optimization problem about weights, it is fond that the attribute dependency degree is just the optimal solution of the weights. As a result, this paper proposed a reduction algorithm based on ranked weights, which significantly accelerate attribute reduction. Numerical experiments demonstrate that the proposed algorithm is suitable on large scale datasets, especially on the datasets with high dimension.

Keywords Fuzzy rough sets, Attribute reduction, Weights, High dimension datasets

1 前言

约简又称为降维,是数据挖掘和机器学习领域的一个重要研究方向。在数据挖掘过程中,为了防止维数灾难,采取降维操作,在保持原有数据集信息完整性的基础上,对数据进行维数约简,即通过某种方法,从原来的高维特征集合中选出低维特征集合。约简后的数据在某种评判标准下,最大限度地保留了原始数据的特征[1]。数据降维的方法可以分为以下两种基本类型:

(1)从数据的所有特征中消除无关、弱相关和冗余的特征,得到的特征是原有特征的子集,即在所有特征中选择最能反映数据性质的特征,称为特征选择[1]。

(2)对原始特征进行某种操作以获取有意义的投影,即把 n 个初始特征投影变换为 m 个特征,并在这 m 个特征上进行后续操作,称为特征抽取[1]。

对数据集进行特征选取的方法有很多,基于模糊粗糙集模型的属性约简即是其中之一。

目前应用最为广泛的基于模糊粗糙集的约简方法是基于差别矩阵的约简算法[14]以及基于属性依赖度的启发式算法,例如 QuickReduct 算法[2-3]。传统的启发式算法的时间代价很高,在处理大规模数据时耗时过长。针对这一不足,本文将属性权重引入属性约简的定义中。通过构建优化问题,发现并证明了基于权重排序的属性约简算法在时间复杂度和收敛性上均优于启发式约简算法。属性依赖度是对属性重要程度的一种度量;本文摒弃常用的启发式算法,提出基于属性权重排序的约简算法,大大提升了约简的速度,使得约简算法可以应用于大规模数据集。

本文第2节回顾了模糊粗糙集模型;第3节将权重引入模糊粗糙集和属性约简的定义中,然后设计了一个优化算法,通过该优化算法可以发现,属性权重越大,则属性的权重越大,从而发现属性权重可以作为属性的重要度;第4节设计了基于权重排序的属性约简算法;第5节通过数值实验发现,基

到稿日期:2017-05-08 返修日期:2017-09-15

范星奇(1993—),男,硕士,主要研究方向为机器学习与不确定信息处理;**李雪峰**(1994—),主要研究方向为机器学习与不确定信息处理;**赵素云**(1979—),女,博士,副教授,硕士生导师,主要研究方向为机器学习、基于模糊集、粗糙集理论和概率统计论的不确定信息处理方法研究,E-mail:zhaosuyun@ruc.edu.cn(通信作者);**陈 红**(1965—),女,博士,教授,博士生导师,主要研究方向为数据仓库与数据挖掘、传感器网络数据管理、流数据管理;**李翠平**(1971—),女,博士,教授,博士生导师,主要研究方向为数据仓库和数据挖掘、社会网络分析。

于权重排序的属性约简算法比经典的启发式算法更快。因此，基于权重排序的属性约简算法更适用于属性远远多于样本个数的数据集。

2 模糊粗糙集模型

模糊粗糙集模型由粗糙集理论衍生而来，是一种强有力的处理模糊和不确定性知识的数学工具，其以数据之间的不可区分关系为基础，能够有效地反映数据之间的相互依赖关系，从而去除冗余的属性，对数据进行降维。

在引入模糊逻辑算子后[4-6]，通过表述模糊粗糙集中元素之间的二元关系，可以更有效地定义模糊粗糙集的上下近似。

定义 1 给定论域 U，$R:[0,1]\times[0,1]\to[0,1]$ 是 U 上的一个二元模糊等价关系，并对 $\forall x,y,z\in U$ 均满足：

(1)自反性：$R(x,x)=1$；

(2)对称性：$R(x,y)=R(y,x)$；

(3)三角传递性：$R(x,y)\geqslant T(R(x,z),R(z,y))$。

若 A 是 U 上的一个模糊集合，则 A 关于 $\langle U,R\rangle$ 的一对下近似和上近似的定义如下[8-9]：

用 t-模和 t-余模算子定义为：

$$\overline{R_T}A(x)=\sup_{u\in U}T(R(x,u),A(u))$$

$$\underline{R_S}A(x)=\inf_{u\in U}S(N(R(x,u)),A(u))$$

用 t-剩余蕴涵算子及其互补算子定义为：

$$\overline{R_\sigma}A(x)=\sup_{u\in U}\sigma(N(R(x,u)),A(u))$$

$$\underline{R_\vartheta}A(x)=\inf_{u\in U}\vartheta(R(x,u),A(u))$$

可以看出，在 N 为标准否定算子，其他算子为 Lukasiewicz 的算子定义方式下(否定算子与 Lukasiewicz 算子的具体定义参见文献[8])，两种模糊逻辑算子定义的模糊粗糙集的上、下近似是相同的，可表示为如下形式：

$$\underline{A}_R(x)=\inf_{u\in U}(1-R(x,u)+A(u))$$

$$\overline{A}_R(x)=\sup_{u\in U}(R(x,u)+A(u)-1)$$

由对等价关系 R 的定义可知，二元关系 $R(x,u)$ 反映了模糊粗糙集中的两个样本点 x 和 u 的相似程度。样本点 x 和 u 的各个属性的特征值越接近，$R(x,u)$ 越大。而隶属函数 $A(u)$ 则反映了样本点 x 对样本点 u 所属分类的隶属程度。当 $A(u)=0$ 时，x 与 u 为经典意义下的异类点；当 $A(u)=1$ 时，x 与 u 为经典意义下的同类点[7]。

将论域 U 定义为一个标准化的欧几里得空间，则模糊粗糙集 A 可当作欧几里得空间中的一个点集。空间中两个点之间的属性特征值越接近，则它们在空间上的距离越接近；反之，则距离越远。即两个点 x 和 y 之间的距离 $d(x,y)$ 和 $R(x,y)$ 呈负相关关系，两个点之间的不相似程度越大，则距离越大。距离 $d(x,y)$ 可以表示为等价关系 $R(x,y)$ 的否定，即 $d(x,y)=N(R(x,y))$。

因此，模糊粗糙集下近似的几何意义可以表示为寻找距离样本点最近的异类点。也就是说，与样本点最近的异类点的距离即为模糊粗糙集的下近似[13]，如图 1 所示。

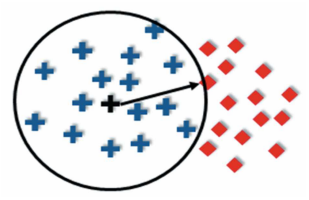

图 1 模糊粗糙集下近似的几何表示

Fig. 1 Geometric representation of lower approximation in fuzzy rough sets

模糊粗糙集在欧几里得空间下的下近似又可以表示为：

$$\underline{A}(x)=\min_{j,D(y)\neq D(x)}d(x,y^{(j)})$$

其中，D 表示样本点所属的决策类。

同理，可以得出上近似的几何表示，即模糊粗糙集的上近似为与样本点距离最近的同类点的相似度。

根据下近似和正域的定义，可以得知模糊粗糙集的正域即为所有样本点下近似的并集，即模糊粗糙集 A 的正域为 $POS_R(A)=\bigcup_{x\in A}\underline{A}_R(x)$。

在粗糙集理论中，属性依赖度可定义如下。

定义 2 对于信息表中的条件属性集 C 和决策属性 D，称 $\delta_C(D)=|POS_C(D)|$ 为条件属性 C 的依赖度，其中 $POS_C(D)$ 为条件属性集 C 在 U 中的正域。

因为论域 U 已经定义为一个标准化的欧几里得空间，所以模糊粗糙集 A 的属性依赖度可以表示为所有下近似向量之和，即：

$$\delta_R(A)=|POS_R(A)|/=\|\sum_{i=1}^{n}\underline{A}(x_i)\|$$

$$=\|\sum_{i=1}^{n}\min_{j,D(y)\neq D(x)}d(x^{(i)},y^{(j)})\|$$

模糊粗糙集依赖度在欧几里得空间的几何意义为模糊粗糙集中所有点与其最近的异类点的距离之和。通过计算每个样本点与其最近的异类点之间的距离，可以求出模糊粗糙集的依赖度，进而通过依赖度对模糊粗糙集进行约简[10-13]。目前，基于依赖度的约简算法基于如下假设：依赖度越大的属性越重要。因此，约简的启发式算法均是通过选择依赖度来设计贪心算法的。但是，没有相关文献明确指出为什么依赖度越大时属性的重要度越高。本文尝试通过引入权重的方法来说明这一问题。

3 基于属性权重的模糊粗糙集

3.1 闵可夫斯基距离带有属性权重的距离

由 $d(x,y)=N(R(x,y))$ 可知，可以用两点之间的距离来表示两点之间的等价关系，并用距离表示模糊粗糙集的下近似。

闵可夫斯基距离为欧几里得距离的一个推广，其基本定义如下[12]。

定义 3 对于 m 维空间上的任意两点 x 和 y，都有 $\vec{x}=(x_1,x_2,\cdots,x_m)^{\mathrm{T}}$，$\vec{y}=(y_1,y_2,\cdots,y_m)^{\mathrm{T}}$，则定义 x 与 y 之间的闵可夫斯基距离为：

$$d(x,y)=(\sum_{i=1}^{m}|x_i-y_i|^p)^{\frac{1}{p}},p\in\mathrm{R}^+$$

可以看出，闵可夫斯基距离是一个通用的距离表示形式。参数 p 取不同的值，反映出的是不同几何度量条件下距离的性质。

当 $p=1$ 时，$d(x,y)=\sum_{i=1}^{m}|x_i-y_i|$，即为曼哈顿距离（城市街区距离），反映的是两个点在标准坐标系上的绝对轴距之和，即在欧几里得空间中两点之间的线段在直角坐标轴上的投影距离之和。

当 $p=2$ 时，$d(x,y)=\sqrt{\sum_{i=1}^{m}(x_i-y_i)^2}$，即为欧几里得距离，是经典意义下的直线距离。

当 $p=\infty$ 时，$d(x,y)=\max_i|x_i-y_i|$，即为切比雪夫距离。切比雪夫距离反映的是两点之间各坐标数值差的最大值，即在欧几里得空间中两点之间的线段在直角坐标轴上的投影距离的最大值。

闵可夫斯基距离用一个公式将不同测度条件下的空间距离统一起来，具有直观性。但闵可夫斯基距离将数据点在各个维度方向上的分量都作为相同的量纲来看待，没有考虑数据点在各个维度上的重要程度是有差异的，反映到模糊粗糙集中就表现为闵可夫斯基距离未区分各个属性对距离影响的重要程度。为了解决这一问题，引入了属性权重的概念。

3.2　带有属性权重的闵可夫斯基距离

定义属性权重 $\vec{\theta}$ 为一个 m 维向量：$\vec{\theta}=(\theta_1,\theta_2,\cdots,\theta_m)^{\mathrm{T}}$，其内积为 m，即：

$$\vec{\theta}^{\mathrm{T}}\ \vec{\theta}=\theta_1^2+\theta_2^2+\cdots+\theta_m^2=m$$

其中，θ_i 表示第 i 个属性的权重。属性权重的绝对值越大，对应的属性对距离的影响就越大，即属性重要程度越高。由此可给出带有属性权重的闵可夫斯基距离的定义。

定义 4　定义距离函数：$d(x,y)=(\sum_{i=1}^{m}|\theta_i(x_i-y_i)|^p)^{\frac{1}{p}}$ 为 m 维空间中两个点 x 和 y 带有属性权重的闵可夫斯基距离。

定义了带有属性权重的闵可夫斯基距离后，可推出带有属性权重的曼哈顿距离、欧几里得距离和切比雪夫距离。

当 $p=1$ 时，$d(x,y)=\sum_{i=1}^{m}\theta_i|x_i-y_i|=|\vec{\theta}\cdot(\vec{x}-\vec{y})|$，即为带有属性权重的曼哈顿距离。

当 $p=2$ 时，$d(x,y)=\sqrt{\sum_{i=1}^{m}\theta_i^2(x_i-y_i)^2}$，即为带有属性权重的欧几里得距离。

当 $p=\infty$ 时，$d(x,y)=\max_i\theta_i(x_i-y_i)$，即为带有属性权重的切比雪夫距离。

通过引入属性权重，解决了闵可夫斯基距离中各个维度上量纲相同而无法区分各个属性重要程度的缺点。对于带有属性权重的闵可夫斯基距离而言，两个数据点在不同维度上的绝对距离相同时，该维度的属性权重越大，两点之间的闵可夫斯基距离也就越大。在模糊粗糙集中，这一性质反映了属性的重要程度。属性权重越大的维度，其对应的特征属性对不同分类的数据点的区分能力也就越强，即该维度上不同数据点的不相似程度较大。当各个维度上的属性重要程度均相同时，属性权重向量为 $\vec{\theta}=(1,1,\cdots,1)^{\mathrm{T}}$，此时带有属性权重的闵可夫斯基距离退化为经典意义上的闵可夫斯基距离。

引入带有属性权重的闵可夫斯基距离后，可以更加贴切地用距离表示模糊粗糙集中数据点之间的不相似程度。相比经典意义上的闵可夫斯基距离，带有属性权重的闵可夫斯基距离更加贴近距离函数 $d(x,y)$ 是等价关系 $R(x,y)$ 的否定，即 $d(x,y)=N(R(x,y))$。

3.3　基于属性权重的模糊粗糙集

下面用带有属性权重的闵可夫斯基距离表示模糊粗糙集的下近似和依赖度。

模糊粗糙集的下近似可以表示为：

$$\underline{A}(x)=\min_{j,D(y)\neq D(x)}d(x,y^{(j)})$$

其中，D 表示样本点所属的决策类。

将距离函数 $d(x,y)$ 用带有权重的闵可夫斯基距离中的切比雪夫距离表示，得到模糊粗糙集的下近似为：

$$\underline{A}(x)=\min_{j,D(y)\neq D(x)}\max_i\theta_i(x_i-y_i^{(j)})$$

根据下近似和正域的定义，模糊粗糙集 A 的属性依赖度可以表示为所有下近似之和，即：

$$\delta_R(A)=|POS_R(A)|=\|\sum_{i=1}^{n}\underline{A}(x_i)\|$$

$$=\sum_{k=1}^{n}\min_{j,D(y)\neq D(x)}\max_i\theta_i(x_i^{(k)}-y_i^{(j)})$$

在实际运算中，除了属性权重在每个属性维度上的分量 θ_i 是未知变量，其他所有参数均为已知变量；且所有的未知数 θ_i 均为一次函数，因此最后得到的属性依赖度是一个关于 θ_i 的线性函数，可以表示为：

$$\delta_R(A)=\sum_{i=1}^{n}\delta_i\theta_i=\vec{\delta}^{\mathrm{T}}\cdot\vec{\theta}$$

其中，δ_i 为常数。

由此得到的向量 $\vec{\delta}$ 是一个 m 维向量，称作依赖度向量。因为属性依赖度反映的是模糊粗糙集的区分能力，所以属性依赖度越大，表示异类之间的距离越大，模糊粗糙集对不同分类的区分能力就越强。显然，当属性依赖度取最大值时，对应的属性权重向量 $\vec{\theta}$ 即为要加入闵可夫斯基距离中的属性权重。这样就将求属性权重的问题转化为一个优化问题，即：

$$\begin{cases}\max\ \delta_R(A)=\sum_{i=1}^{n}\delta_i\theta_i=\vec{\delta}^{\mathrm{T}}\cdot\vec{\theta}\\ \text{s.t.}\ \|\vec{\theta}\|^2=\theta_1^2+\theta_2^2+\cdots+\theta_m^2=m\end{cases}$$

显然，在属性权重向量 $\vec{\theta}$ 的内积固定的限制条件下，当 $\vec{\theta}$ 的方向与依赖度向量 $\vec{\delta}$ 的方向相同时，属性依赖度取最大值，该优化问题取得最优解。

该优化问题的最优解为：$\hat{\vec{\theta}}=\frac{\sqrt{m}\cdot\vec{\delta}}{\|\vec{\delta}\|}$，即为所求模糊粗糙集的属性权重。

属性权重反映了模糊粗糙集中各个属性的重要程度，某一属性的属性权重绝对值越大，表示其对模糊粗糙集属性依赖度的影响越大，去掉该属性会显著影响模糊粗糙集的区分能力，即该属性为不可约简的属性。反之，某一属性的属性权重绝对值越小，其对模糊粗糙集属性依赖度的影响越小，去掉

该属性不会显著影响模糊粗糙集的区分能力，即该属性为可约简的属性。

因此，可以根据属性权重的大小判断模糊粗糙集中的冗余属性，进而进行约简，如图 2 所示。

在图 2 中，左图是论域空间在三维空间上的映射，其中存在着隶属于两个不同分类的数据点以及 3 个不同维度上的特征属性 x_1，x_2，x_3。通过计算属性权重向量 $\vec{\delta}$ 可以看出，x_2 方向上属性权重的分量较小，这意味着相比于 x_1 和 x_3，x_2 对属性依赖度的影响较小，不会显著影响模糊粗糙集的区分能力，因此可以将属性 x_2 进行约简，此时论域空间转化为右图所示的二维空间。可以看出，在属性约简之后，模糊粗糙集仍然保持着原有的区分能力。

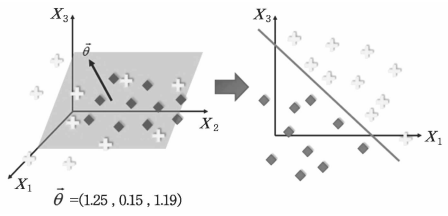

图 2 对属性权重较小的属性进行约简

Fig. 2 Reduction of attributes with smaller attribute weights

因此，通过计算属性权重可以判断某一属性是否更有可能是可约简的冗余属性，从而设计出基于属性权重的模糊粗糙集约简算法。

4 基于属性权重的模糊粗糙集约简算法

本节设计一个约简算法，根据属性依赖度求出属性权重，进而求出相应的约简。

首先计算属性依赖度，其主要思想是利用两个二维数组来储存两个样本点之间各维度的距离，进而求得下近似和依赖度。

根据第 3.2 节所述，在切比雪夫距离定义下，下近似相当于选择出样本点与一个异类点在所有属性维度上距离的最大值，再在所有异类点的距离最大值中间选择一个距离的最小值。反映在该二维数组上，就是先计算每一行距离的最大值并存为一列，再在这一列中选出最小值，即为该样本点的下近似。

求得的每一个样本点的下近似都以 m 维向量的形式存入二维下近似数组 $A[n][m+1]$ 的每一行中，该行只有下近似的距离所对应的属性列为非零值，其他列均为零。这样将所有行的数值相加，即可得到存放在一个一维数组中的依赖度向量 $\vec{\delta}$，在算法中用 $Dep[m]$ 表示。

4.1 模糊粗糙集约简的启发式算法

基于属性依赖度的启发式约简算法是目前常用的一种约简算法，其大致思想是向一个空子集中不断添加属性元素并计算依赖度，最终找到一个属性约简。

模糊粗糙集的启发式算法开始于一个空的属性集，向空属性集中迭代地添加属性，并根据属性依赖度的计算方法计算每一个属性下模糊粗糙集的依赖度，选择并添加一个具有最大区分能力即使依赖度最大的属性，然后重复以上步骤。当添加某一个属性后依赖度与属性全集 A 的依赖度相等或与添加前相比属性依赖度不变时，表明再加入属性不能提供更精确的分类，此时算法终止，输出属性约简结果。

模糊粗糙集的启发式约简算法的详细描述如算法 1 所示[6]。

算法 1 模糊粗糙集求约简的启发式算法

输入：模糊粗糙集 FRS=〈U，A，D〉，约简属性集合 Red←{}

输出：Red

Step1 计算模糊粗糙集 FRS 在属性全集 A={a_1，a_2，…，a_m}下的依赖度向量 Dep[m]，并计算依赖度向量的模：

$$Dep(A)=\|Dep[m]\|=\sqrt{\sum_{i=0}^{m-1}Dep[i]^2}$$

Step2 令集合 B=A−Red(K)；

Step3 任取一个属性 a∈B，计算 Dep(Red∪{a})并存入数组，B←B−{a}。

Step4 重复 Step3 直至 B=∅，从数组中取出使 Dep(Red∪{a})最大的属性 a，Red←Red∪{a}。

Step5 若 Dep(Red)<Dep(A)，则重复 Step2−Step4，直至 Dep(Red)=Dep(A)。

Step6 输出 Red。

可以看出，启发式算法的基本工作原理是依据当前属性集合的局部最优，不断地将能够使当前属性集合辨识度提升最高的属性加入到目标子集中，最终找到一个满足要求的属性子集作为约简。该启发式约简算法虽然不一定能够保证得到的约简是最优的约简结果，但该算法在最坏的情况下也只需要进行 $m(m+1)/2$ 次迭代便可得出约简结果，将模糊粗糙集关于属性的时间复杂度降低到了平方量级，即 $O(m^2)$。

当属性个数较多，且数据集的元组规模庞大时，从空属性集开始逐步添加属性会大量重复执行算法的 Step3，进而浪费大量的时间，造成算法性能下降。因此，本文将属性权重的概念引入模糊粗糙集的约简中，通过属性权重来标记属性重要程度，从而有针对性地对属性进行约简，不再需要大量重复的迭代操作。

4.2 基于属性权重的模糊粗糙集约简算法

基于属性权重的模糊粗糙集约简算法的主要思想是：首先计算属性全集的依赖度，得到以向量形式表示的依赖度之后再计算属性权重；然后根据属性权重的绝对值对属性集中的每个属性按从小到大的顺序进行排序，从属性权重最小的属性开始逐个删除属性并计算依赖度；若删除属性之后依赖度没有变化，则该属性可约简，若依赖度减小，则该属性不可约简，再将其加回属性集中，直至将所有属性全部遍历，算法终止并输出约简结果。

基于属性权重的模糊粗糙集约简算法的详细描述如算法 2 所示。

算法 2 基于属性权重的模糊粗糙集约简算法

输入：模糊粗糙集 FRS=〈U，A，D〉，约简属性集合 Red←{}

输出：Red

Step1　计算模糊粗糙集 FRS 在属性全集 $A=\{a_1,a_2,\cdots,a_m\}$ 下的依赖度向量 Dep[m]，并计算依赖度向量的模：$Dep(A)=\|Dep[m]\|=\sqrt{\sum_{i=0}^{m-1}Dep[i]^2}$。

Step2　建立属性权重向量数组 θ[m]，其长度为 $\sqrt{m}$，方向与依赖度向量一致。即：

$$\theta[i]=\frac{Dep[i]\cdot\sqrt{m}}{Dep},i=0,1,\cdots,m-1$$

Step3　对 θ[i]按绝对值从小到大进行排序，取出绝对值最小的 θ[i]，Red←Red−$\{a_{i+1}\}$。

Step4　计算 Dep(Red)，若 Dep(Red)=Dep(A)，则 Red 保持不变；若 Dep(Red)<Dep(A)，则 Red←Red∪$\{a_{i+1}\}$。

Step5　取出下一个 θ[i]，重复 Step3 和 Step4。待 θ[i]全部遍历之后，输出 Red。

由于属性代表了属性的重要程度，因此按照属性权重由小到大排列，可约简的冗余属性基本都排在前面，在前几次循环中便可删除，故只需迭代一次即可得到最终的约简结果。通过引入属性权重对模糊粗糙集进行约简，将模糊粗糙集关于属性的时间复杂度降低到线性量级，即 $O(m)$，进一步降低了时间复杂度，提高了约简计算的效率。

5　数值实验

5.1　实验环境及所用的数据集

本文实验均是在 linux 系统下，由 python 编码完成。实验所使用的硬件参数为：CPU 为 Intel(R) Core(TM) i7-4510U CPU @ 2.00GHz 2.60GHz，内存为 8GB。

本文实验应用 10 个 UCI 数据集[4]，均提取其中两类，提取后的具体数据如表 1 所列。

表 1　实验数据集

Table 1　Experimental datasets

数据集	属性	记录条数
gene1	800	33
gene15	700	102
gene6	500	52
sonar	60	208
spectF	44	212
QSAR	41	500
Wdbc	30	569
EEG Eye State	14	12000
Poker Hand	11	12000
Abalone	8	2400
iris	3	100

本文的数值实验主要验证以下几方面的内容：

(1)用基于权重的模糊粗糙集约简算法与传统的启发式约简算法对数据集进行模糊粗糙集约简，以约简所需的时间为指标来比较约简操作的效率。

(2)基于 KNN 分类算法，利用约简前以及两种算法约简后的属性集分别对相应的数据集进行分类，通过比较查准率、查全率、分类正确率等验证两种约简算法的可行性。

5.2　两种约简算法的运行时间

分别采用两种算法对 10 个不同的数据集进行约简，约简所需时间如表 2 所列。

表 2　在不同数据集下执行两种算法所需的时间/s

Table 2　Execution time based on different datasets of two algorithms/s

数据集	启发式运行时间	属性权重运行时间
gene1	30808.83	110.33
gene15	20505.87	813.09
gene6	13600.42	81.76
sonar	292.198	17.43
spectF	189.304	13.992
QSAR	1101.178	79.094
Wdbc	663.66	76.94
EEG Eye State	19015.02	8752.68
Poker Hand	17532.29	4858.52
Abalone	437.224	123.468
iris	0.15422	0.05336

从表 2 中可以看出，相比传统的启发式约简算法，基于属性权重的模糊粗糙集约简算法可以显著地提高算法的时间效率，减少算法执行消耗的时间，从而证明了基于属性权重的约简算法的可行性和有效性。

同时从表 2 可以发现，一个数据集中所含的属性个数越多，基于属性权重的约简算法相对传统的启发式约简算法在时间性能上的优越性就越大。诸如 EEG Eye State，Poker Hand，Abalone，iris 这类属性个数在 15 个以下的数据集，使用快速约简算法仅能将时间性能提高 1～2 倍左右。而对于 sonar，spectF，QSAR biodegradation 这类具有数十个属性的数据集，使用基于属性权重的约简算法可以将算法的时间性能提高近十倍或十几倍；而对于 gene1 和 gene15 这两个具有数百个属性的数据集，属性权重的约简算法可以将算法的时间性能分别提高 280 倍和 25 倍左右。

由于在模糊粗糙集中，影响约简算法时间复杂度的两个最主要的因素是模糊粗糙集的数据集大小 n 以及属性个数 m，因此我们将通过实验来分别研究数据集中样本点的个数(数据集大小)以及数据集中的属性个数(属性集大小)对于快速约简算法和启发式约简算法时间性能的影响。

5.3　两种约简算法执行时间和数据集大小的关系

本节实验采用的数据集是 poker hand。在实验中将随机抽取一部分数据组成集合的子集，来检验两种算法运行的时间效率。具体操作是：首先抽取 100 个数据分别运行启发式约简算法和基于属性权重的约简算法，并观察其时间性能；然后以 100 为增量逐步递增到 2000 个数据，共测试 20 组，并根据测试出的运行时间绘制出程序运行时间与数据集的大小关系，如图 3 所示。

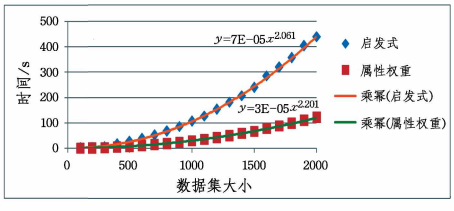

图 3　两种算法的执行时间与数据集大小的关系(poker hand)

Fig. 3　Relationship between execution time and the size of dataset of two algorithms(poker hand)

对图 3 中的数据进行幂函数拟合的结果如下：基于属性权重的模糊粗糙集约简算法的拟合函数为 $y=3E-05x^{2.0215}$，

传统启发式约简算法的拟合函数为 $y=7E-05x^{2.0614}$，可认为两种算法的执行时间和数据集大小的关系均满足二次函数 $y=mx^2$。

这说明两种算法在数据集上的时间复杂度相同，即基于属性权重的约简算法和启发式约简算法对数据集大小的时间复杂度均为 $O(n^2)$，说明相比于启发式约简算法，基于属性权重的约简算法虽然大大提高了约简的效率，但并未降低数据集大小上的时间复杂度，基于属性权重的约简算法相对启发式约简算法在时间复杂度上并无优势。

5.4 两种约简算法执行时间和属性集大小的关系

下面验证两种约简算法执行的时间和数据集的属性规模之间的关系，以探究属性集的大小对算法性能的影响。

本部分实验采用的数据集是 sonar 和 QSAR biodegradation。在实验中将随机抽取一部分属性组成数据集中属性集合的子集，来检验两种算法运行的时间效率。具体操作是：首先抽取 5 个属性分别运行启发式约简算法和基于属性权重的约简算法，并观察其时间性能；然后以 5 为增量逐步递增到属性全集，并根据测试出的运行时间绘制出程序运行时间与属性集大小的关系，如图 4 和图 5 所示。

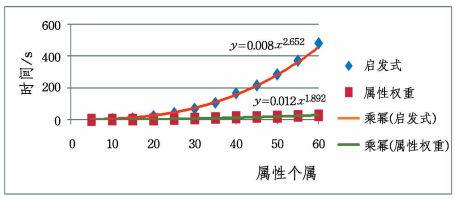

图 4 两种算法的执行时间和属性个数的关系(sonar)

Fig. 4 Relationship between execution time and the number of attributes of two algorithms (sonar)

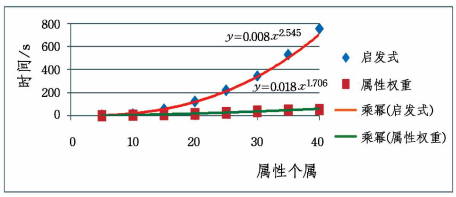

图 5 两种算法的执行时间和属性个数的关系(QSAR biodegradation)

Fig. 5 Relationship between execution time and the number of attributes of two algorithms (QSAR biodegradation)

对图 4、图 5 中的数据进行幂函数拟合的结果如下：sonar 数据集基于属性权重的算法的拟合函数为 $y=0.0126x^{1.8926}$，传统启发式约简算法的拟合函数为 $y=0.0087x^{2.6527}$；QSAR biodegradation 数据集基于属性权重的算法的拟合函数为 $y=0.1082x^{1.7063}$，传统启发式约简算法的拟合函数为 $y=0.0587x^{2.5455}$。

这说明基于属性权重的约简算法相对启发式约简算法在属性集的时间复杂度上有巨大的优势，基于属性权重的约简算法比启发式算法在属性个数上的复杂度降低了接近一次方量级，从而大大提升了算法的时间性能。可以看出，数据集中所包含的属性规模越大，包含的属性个数越多，基于属性权重的约简算法也就越有优势。

5.5 两种约简算法约简结果的比较

本节的实验采用 EEG Eye State，gene15，gene1，sonar，Wdbc，spectF，QSAR biodegradation，gene6 等 8 个数据集，分别用两种约简算法对数据集进行属性约简，约减后计算两种算法约简结果的 Jaccard 相似度，及两者的交集与并集之比，结果如表 3 所列。

表 3 两种算法约简后的属性个数及结果的相似度

Table 3 The number of attributes and the similarity of results after reduction of two algorithms

数据集	属性	启发式约简结果	属性权重约简结果	Jaccard 相似度
EEG Eye State	14	8	5	0.625
gene15	700	149	225	0.238
gene1	800	797	34	0.043
sonar	60	60	32	0.53
Wdbc	30	30	21	0.7
spectF	44	44	22	0.5
QSAR biodegradation	41	41	16	0.39
gene6	500	500	49	0.096

从表 3 中可以看出，对于 EEG Eye State，gene15，gene1 这 3 个数据集，两种算法均进行了约简；而对 sonar，Wdbc，spectF，QSAR biodegradation，gene6 这 5 个数据集，启发式算法的结果仍然是属性全集，并没有约简，而属性权重算法则进行了约简，两种算法的约简结果相似度没有明显的规律，无论是属性个数较多的数据集还是较大的数据集，两者的结果都有相似度高和相似度低的情况，在所有的数据集上，属性权重算法约简后的属性集都小于启发式算法约简的结果。

为了进一步比较两种算法约减后的结果，用 KNN 算法分别对两种算法约简前后的数据集进行分类，分类前按照两种算法的结果对数据集进行约简，分类时抽取 70%的数据作为训练集，其余 30%的数据作为测试集。针对原数据集和两种算法约简后的数据集的分类结果，计算查准率 P、查全率 R 和分类正确率 3 个指标，结果如表 4 所列。

表 4 两种算法约简前后的 KNN 分类结果

Table 4 Results of KNN classification before and after reduction of two algorithms

	查准率 P			查全率 R			正确率		
数据集	未约简	启发式	属性权重	未约简	启发式	属性权重	未约简	启发式	属性权重
EEG Eye State	0.9234	0.9038	0.8663	0.9275	0.9078	0.8745	0.9253	0.9056	0.8698
gene15	0.83	0.7858	0.75	0.75	0.825	0.825	0.8	0.8	0.775
gene1	1	1	1	1	1	1	1	1	1
sonar	0.942	0.942	0.9559	0.9286	0.9286	0.9286	0.9357	0.9357	0.9429
Wdbc	0.9781	0.9781	0.978	1	1	0.9972	0.9859	0.9859	0.9842
spectF	0.9162	0.9162	0.8895	0.9107	0.9107	0.9107	0.8632	0.8632	0.8396
QSAR biodegradation	0.8926	0.8926	0.8113	0.8867	0.8867	0.86	0.934	0.934	0.898
gene6	0.88	0.88	0.9583	1	1	0.9583	0.9423	0.9423	0.9615

对于 EEG Eye State，gene15，gene1 这 3 个数据集，两种算法都进行了约简，EEG Eye State 属于数据个数较多的数据集，gene15 和 gene1 属于属性个数较多的数据集。两种算法约简后与约简前相比，KNN 分类的正确率、查准率以及查全率都没有显著下降，在数据集 gene15 的结果中约减后的查全率还要高于原数据集，而在数据集 gene1 的结果中两种算法约简后与原数据集的分类效果是完全相同的。

对于 sonar，Wdbc，spectF，QSAR biodegradation，gene6 这 5 个数据集，启发式算法的结果是不约简，属性权重算法则进行了约简。约简后的数据集和原数据集相比，总体上 KNN 分类的查全率、查准率和正确率都没有显著下降，在大部分情况下分类结果几乎没有差别甚至部分约简后的分类结果优于原数据集。

从以上的几个实验结果可以看出，无论在运行时间上还是在约简的结果上，属性权重算法相比启发式算法有着巨大的优势，属性权重算法可以用更少的时间找到不显著降低分类效果的属性集子集。在运行时间上，属性权重算法将属性个数上的时间复杂度降低了；而且在传统的启发式算法得出不约简原属性集结果的情况下，属性权重算法仍然可以得出约简的结果，证明属性权重是一个效率与可行性兼备的算法。

结束语　本文通过引入属性权重的概念，提出了基于属性权重的模糊粗糙集约简算法，其主要优点和创新点在于引入属性权重的概念，通过求得属性权重对模糊粗糙集进行约简，不需要反复增删属性进行迭代，提高了约简的效率。本文的主要贡献如下：

(1)通过引入带有属性权重的闵可夫斯基距离，更精确地描述了模糊粗糙集中的等价关系，从而引入了属性权重的概念，利用属性权重的概念刻画模糊粗糙集中下近似和属性依赖度的定义。

(2)通过属性权重刻画的属性依赖度，定义了模糊粗糙集的各个属性在数据集中的重要程度，并根据属性权重对模糊粗糙集进行了属性约简，相比原有的启发式约简算法，降低了属性约简的时间复杂度。

(3)通过数值实验，验证了基于属性权重的模糊粗糙集约简算法相比传统的启发式约简算法有着良好的时间性能，尤其是处理属性维数较多的集合时性能更好，在约简的结果上也优于传统的启发式算法。

参 考 文 献

[1] HU J. Survey on feature dimension reduction for high-dimensional data[J]. Application Research of Computers, 2008(9): 2601-2606. (in Chinese)
胡洁. 高维数据特征降维研究综述[J]. 计算机应用研究，2008(9)：2601-2606.

[2] JENSON R, SHEN Q. Fuzzy-rough attribute reduction with application to web categorization[J]. Fuzzy Sets and Systems, 2004, 141(3): 469-485.

[3] JENSON R, SHEN Q. Fuzzy-rough sets for descriptive dimensionality reduction[C]// IEEE International Comference on Plasma Science. 2002: 29-34.

[4] MENGER K. Statistical metrics[J]. Proceedings of the National Academy of Sciences of the United States of America, 1942, 28(12): 535-537.

[5] DUBOIS D, PRADE H. Fuzzy sets in approximate reasoning, Part 1: Inference with possibility distributions[J]. Fuzzy Sets & Systems, 1999, 40(1): 73-132.

[6] ZHAO Y L, WANG Y L. Generalized fuzzy rough set approach to fuzzy information reduct[J]. Computer Engineering and Applications, 2008, 44(4): 169-171. (in Chinese)
赵越岭，王英丽. 广义模糊粗糙集在模糊信息约简中的应用[J]. 计算机工程与应用，2008，44(4)：169-171.

[7] 胡清华，于达仁. 应用粗糙计算[M]. 北京：科学出版社，2012.

[8] ZHAO S Y, TSANG C C, CHEN D G. The model of fuzzy variable precision rough sets[J]. IEEE Trans. Fuzzy Systems, 2009, 17(2): 451-467.

[9] ZHAO S Y, TSANG C C. On Fuzzy approximation Operators in Attribute Reduction with Fuzzy Rough Sets[J]. Information Sciences, 2008, 178(16): 3162-3176.

[10] YAO Y Y, ZHAO Y. Attribute reduction in decision-theoretic rough set models[J]. Information Sciences, 2008, 178(17): 3356-3373.

[11] UCI. Machine Learning Repository[OL]. http://archive.ics.uci.edu/ml.

[12] JMMM C. A New Minkowski Distance Based on Induced Aggregation Operators[J]. International Journal of Computational Intelligence Systems, 2011(2): 123-133.

[13] ZHANG Z X, FAN X Q, ZHAO S Y, et al. Fast reduction algorithm research based on k-nearest neighbor fuzzy rough set[J]. Journal of Frontiers of Computer Science and Technology, 2015, 9(1): 14-23. (in Chinese)
张照星，范星奇，赵素云，等. k-近邻模糊粗糙集的快速约简算法研究[J]. 计算机科学与探索，2015，9(1)：14-23.

[14] TSANG E C C, CHEN D G, YEUNG D S, et al. Attributes reduction using fuzzy rough sets[J]. IEEE Transactions on fuzzy system, 2008, 16(5): 1130-1141.