ISSN 1000-9825, CODEN RUXUEW
Journal of Software, Vol.20, No.4, April 2009, pp.804−814
doi: 10.3724/SP.J.1001.2009.03252

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

基于人工免疫系统的数据简化*

公茂果+, 郝 琳, 焦李成, 王晓华, 孙奕菲

(西安电子科技大学 智能信息处理研究所 智能感知与图像理解教育部重点实验室,陕西 西安 710071)

Data Reduction Based on Artificial Immune System

GONG Mao-Guo+, HAO Lin, JIAO Li-Cheng, WANG Xiao-Hua, SUN Yi-Fei

(Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education of China, Institute of Intelligent Information Processing, Xidian University, Xi'an 710071, China)

+ Corresponding author: E-mail: gong@ieee.org

Gong MG, Hao L, Jiao LC, Wang XH, Sun YF. Data reduction based on artificial immune system. *Journal of Software*, 2009,20(4):804−814. http://www.jos.org.cn/1000-9825/3252.htm

Abstract: Based on the antibody clonal selection theory, an immune clonal data reduction algorithm is proposed for instance selection problems of data reduction. The theory of Markov chain proves that the new algorithm is convergent with probability 1. The experimental studies on seven standard data sets of UCI repository show that the algorithm proposed in this paper is effective. The best domain of the weight parameter λ is determined by analyzing its effect on algorithm's performance. Furthermore, an encoding method based on the stratified strategy is introduced to accelerate the convergence speed when solving large scale data reduction problems. The experimental studies based on seven large scale data sets show that the improved method is superior to the primary one. Finally, the best domain of the number of stratums t is determined by analyzing its effect on algorithm's performance based on the data sets Letter and DNA.

Key words: clonal selection; data reduction; instance selection; artificial immune system; evolutionary computation

摘 要: 针对数据简化中的实例选择问题,基于抗体克隆选择学说提出了一种免疫克隆数据简化算法.利用马尔可夫理论证明了该算法能以概率 1 收敛.通过对 7 个具有代表性的标准 UCI 数据集的简化实验证明了该算法的有效性.通过实验分析了权值参数λ的取值变化对算法性能的影响,确定了其最佳取值区间.针对海量数据集简化时算法收敛较慢的问题,引入分层编码策略.通过对 7 个大规模及海量数据集的简化实验表明了在进化代数不变的情况下,新的编码方式能够极大地提高算法的收敛速度,得到更为理想的结果.通过对Letter和DNA两个数据集的实验给出了分层编码中层数 t 的最佳取值区间.

关键词: 克隆选择;数据简化;实例选择;人工免疫系统;进化计算

* Supported by the National Natural Science Foundation of China under Grant Nos.60703107, 60703108 (国家自然科学基金); the National High-Tech Research and Development Plan of China under Grant No.2009AA12Z210 (国家高技术研究发展计划(863)); the Program for New Century Excellent Talents in University of China (新世纪优秀人才支持计划); the National Basic Research Program of China under Grant No.2006CB705700 (国家重点基础研究发展计划(973))

Received 2007-06-15; Accepted 2008-01-10

中图法分类号: TP18 文献标识码: A

由于计算机和网络的广泛应用,使得我们需要对海量数据进行处理才能得到想要的信息和数据.数据库知识发现(KDD)技术正是基于此问题产生的.对数据集进行有效的预处理是成功构造一个 KDD 系统的前提和关键步骤.数据简化[1]是数据预处理的一项重要内容,其目的主要有以下两点:一是尽量减小数据规模从而减少存储空间和处理时间;二是尽量保留原数据的有用信息从而使数据挖掘的结果更可信,准确性更高.有些研究的目的是基于其中一点[2],而更多的则把两者相结合,考虑两者的折衷最优.经过近 20 年的研究,有关数据简化的算法越来越趋于成熟[3−6].

常见的数据简化方法有特征选择[7]和实例选择[8]两种.实例选择是数据简化的一种典型方法,它直接减少样本的个数,因此可以把数据集的实例简化转化成为一个组合优化问题来处理.基于克隆选择原理提出的克隆选择算法是人工免疫系统算法中的一个典型代表,它继承了生物学抗体克隆选择过程所独有的学习、记忆、抗体多样性等性能.由于其较强的搜索能力,将克隆选择算法用于实例选择有可能是一个很好的选择.

本文将数据简化中的实例选择问题转变为一个组合优化问题,并针对此问题从生物免疫机理出发设计了合适的算子,提出了免疫克隆数据简化算法.文章第 4 节通过对 7 个不同规模的数据集(均源于 UCI 标准数据集)进行测试,证明了此算法的有效性,并用实验说明了权值参数λ对算法性能的影响.针对海量数据简化时带来的空间和时间复杂度过大,算法收敛较慢的问题,引入了分层策略对编码方式进行改进,通过对 7 个大规模及海量数据集的简化实验证明了,在进化代数不变的情况下,相对于原方法,分层策略能够大幅度地提高收敛速度,得到更为理想的结果.最后通过对 Letter 和 DNA 两个数据集的实验给出了分层编码中层数 t 的最佳取值区间.

1 数据简化问题描述

数据简化一般分特征选择和实例选择两种,本文研究的重点是实例选择.图 1 给出了实例选择过程的一个一般模型[9].数据集 D 包含训练样本集 TR 和测试样本集 TE 两部分.通过实例选择,得到 TR 的一个子集 S.最后通过对测试数据集进行测试(本文用最近邻分类算法对其进行分类)来评价所用实例选择算法的性能的优劣.

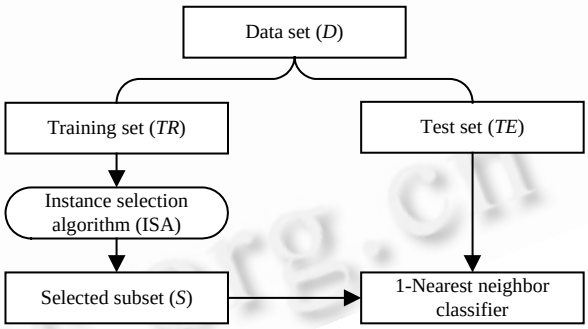

Fig.1 Instance selection model

图 1 实例选择模型

已有的实例选择算法主要是基于最近邻规则提出来的,如 Cnn[10],Ib2[11],Ib3[12],也有基于移动样本或随机取样思想提出来的,如 Drop1[13],Drop2[13],Rmhc[14].它们各有所长,但也存在不可避免的缺点.如 Cnn 能够完全正确地将剩余样本进行分类,但它找不到满足条件的最小的子集,Ib2 和 Ib3 与 Cnn 类似,只是使用了不同的选择策略,Ib2 分类准确率较高,Ib3 的简化率和分类准确率均无法达到最优,但能取得两者的一个较好的折衷.Drop1 随机移出一个样本通过观察交叉验证正确率是否下降来判断是否需要保留该样本.在 Drop2 中首先按照样本与距其最近的异类样本的距离大小来排序,距离最大的样本首先被移出判断.Rmhc 随机选取 TR 的一个子集 S,在每次迭代中,用 TR-S 中的一个样本代替 S 中的一个样本,如果分类准确率提高了,则保留这一替换.以上算法均是根据非进化算法思想提出的,已有文献表明,基于进化计算(如遗传算法)的数据简化算法也能达到较为不错的效果[6].由于免疫克隆选择算法与遗传算法相比,在很多问题上表现出了较好的性能[15,16],因此本文采用免疫克隆选择算法进行数据简化.

2 免疫克隆数据简化算法

克隆选择原理是免疫学中用于解释免疫响应过程的重要理论.基于克隆选择原理,我们提出了一种用于解决数据简化问题的新算法——免疫克隆数据简化算法.

2.1 编 码

假设数据集 D 分为训练样本集 TR 和测试样本集 TE 两部分,TR 中有 m 个实例,则搜索空间即为 TR 的所有子集.一个抗体表示 TR 的一个子集.由于每个基因位所对应的模式只可能存在两种情况,因此采用二进制编码,编码长度为 m(即一个实例对应一个基因),如果某基因位的值是 1,那么该位对应的实例属于 TR 的这个子集,如果值是 0,则正好相反.

2.2 亲和度函数设计

设抗体种群为 $\boldsymbol{A}=\{\boldsymbol{a}_1,\boldsymbol{a}_2,...,\boldsymbol{a}_n\}$, $\boldsymbol{a}_i, 1\le i\le n$ 表示 TR 的一个实例子集 S_i.定义亲和度函数如下:

$$f(\boldsymbol{a}_i)=\lambda\times clas+(1-\lambda)\times per\_redu \tag{1}$$

其中 $clas$ 表示交叉验证时以 S_i 为训练样本的分类正确率,本文采用最近邻分类器进行分类.per_redu 表示与原训练样本集 TR 相比 S_i 的简化率,其定义如下:

$$per\_redu=\frac{|TR|-|S_i|}{|TR|}\times 100\% \tag{2}$$

λ是一个介于 0 和 1 之间的权值参数,λ越大,则 $clas$ 对亲和度的影响越大,反之,则 per_redu 对亲和度的影响越大.

由于该算法的目的是用尽量少的样本代替原来的训练样本集,并且使得分类正确率较之以前不发生较大改变.那么原问题可以转化为式(1)的最大化问题.

2.3 算法流程及主要算子设计

免疫克隆数据简化算法的流程如下:

Step 1. 设置算法初始参数;随机产生初始抗体群 $\boldsymbol{A}(0)$;令当前迭代次数 k:=0;

Step 2. 对抗体群 $\boldsymbol{A}(k)$执行克隆操作;得到新的抗体群 $\boldsymbol{A}^{(1)}(k)$;

Step 3. 对抗体群 $\boldsymbol{A}^{(1)}(k)$执行免疫基因操作,得到新的抗体群 $\boldsymbol{A}^{(2)}(k)$;

Step 4. 根据公式(1)计算所有抗体的亲和度;

Step 5. 对抗体群 $\boldsymbol{A}^{(2)}(k)$和 $\boldsymbol{A}(k)$执行克隆选择操作,得到新的抗体群 $\boldsymbol{A}(k+1)$;

Step 6. 令 k:=k+1;若满足终止条件,算法停止;否则,返回 Step 2.

该算法中主要算子定义如下:

克隆操作 T_c^c:在免疫学中,克隆是指通过无性繁殖(如细胞丝分裂)可连续传代并形成群体.在本文算法中,对抗体种群 $\boldsymbol{A}(k)=\{\boldsymbol{a}_1(k),\boldsymbol{a}_2(k),...,\boldsymbol{a}_n(k)\}$ 的克隆操作 T_c^c 定义为

$$\boldsymbol{A}^{(1)}(k)=T_c^c\left(\boldsymbol{A}(k)\right)=\left\{T_c^c(\boldsymbol{a}_1(k)),T_c^c(\boldsymbol{a}_2(k)),...,T_c^c(\boldsymbol{a}_n(k))\right\} \tag{3}$$

其中,$T_c^c\left(\boldsymbol{a}_i(k)\right)=\left\{\boldsymbol{a}_i^1(k),\boldsymbol{a}_i^2(k),...,\boldsymbol{a}_i^{nc}(k)\right\}$;$\boldsymbol{a}_i^j(k)=\boldsymbol{a}_i(k); i=1,2,...,n; j=1,2,...,nc$;nc 为克隆比例.可见,上述克隆过程与免疫学中的克隆类似,是简单的无性繁殖过程.同一个抗体 $\boldsymbol{a}_i(k)$经过克隆增殖后形成的克隆子种群 $\left\{\boldsymbol{a}_i^1(k),\boldsymbol{a}_i^2(k),...,\boldsymbol{a}_i^{nc}(k)\right\}$ 中的所有抗体与抗体 $\boldsymbol{a}_i(k)$具有完全相同的属性.

免疫基因操作 T_m^c:免疫基因操作有多种形式,本文采用变异操作,抗体种群以如下方式变异:

$$\boldsymbol{A}^{(2)}(k)=T_m^c\left(\boldsymbol{A}^{(1)}(k)\right)=\left\{T_m^c\left(\boldsymbol{a}_i^j(k)\right)\right\}; i=1,2,...,n; j=1,2,...,nc \tag{4}$$

由于算法采用 0,1 二进制编码实现,因此变异操作 $T_m^c\left(\boldsymbol{a}_i^j(k)\right)$ 是指抗体的每一基因位的值以 pm(0<pm<1)的概率进行取反操作.为下文描述简单,$\boldsymbol{a}_i^j(k)$ 经过变异操作后的抗体用 $\boldsymbol{b}_i^j(k)$ 表示,即 $\boldsymbol{b}_i^j(k)=T_m^c\left(\boldsymbol{a}_i^j(k)\right)$,

$\boldsymbol{A}^{(2)}(k)=\left\{\boldsymbol{b}_i^j(k)\right\}, i=1,2,...,n; j=1,2,...,nc$.

克隆选择操作 T_s^c:克隆选择操作是从抗体经过克隆和变异后的子代中选择优秀的个体,从而形成新的种群.对抗体群 $\boldsymbol{A}^{(2)}(k)=\left\{\boldsymbol{b}_i^j(k)\right\}, i=1,2,...,n; j=1,2,...,nc$,克隆选择操作 T_s^c 定义如下:

$$\begin{aligned}&\boldsymbol{A}(k+1)=T_s^c\left(\boldsymbol{A}^{(2)}(k)\cup\boldsymbol{A}(k)\right)\\&=\left\{T_s^c\left(\boldsymbol{b}_1^1(k),\boldsymbol{b}_1^2(k),...,\boldsymbol{b}_1^{nc}(k),\boldsymbol{a}_1(k)\right),T_s^c\left(\boldsymbol{b}_2^1(k),\boldsymbol{b}_2^2(k),...,\boldsymbol{b}_2^{nc}(k),\boldsymbol{a}_2(k)\right),...,T_s^c\left(\boldsymbol{b}_n^1(k),\boldsymbol{b}_n^2(k),...,\boldsymbol{b}_n^{nc}(k),\boldsymbol{a}_n(k)\right)\right\}\end{aligned}\tag{5}$$

其中, $\boldsymbol{A}(k+1)=\left\{\boldsymbol{a}_1(k+1),\boldsymbol{a}_2(k+1),...,\boldsymbol{a}_n(k+1)\right\}$, $\boldsymbol{a}_i(k+1)=T_s^c\left(\boldsymbol{b}_i^1(k),\boldsymbol{b}_i^2(k),...,\boldsymbol{b}_i^{nc}(k),\boldsymbol{a}_i(k)\right)$.

$\forall i$=1,2,…,n,$\exists j\in$ {1,2,…,nc},使抗体 $\boldsymbol{b}_i^j(k)$ 为子种群 $\left\{\boldsymbol{b}_i^1(k),\boldsymbol{b}_i^2(k),...,\boldsymbol{b}_i^{nc}(k)\right\}$ 中亲和度最高的抗体,则

$$\boldsymbol{a}_i(k+1)=T_s^c\left(\boldsymbol{b}_i^1(k),\boldsymbol{b}_i^2(k),...,\boldsymbol{b}_i^{nc}(k),\boldsymbol{a}_i(k)\right)=\begin{cases}\boldsymbol{a}_i(k) & \text{if } f(\boldsymbol{a}_i(k))>f(\boldsymbol{b}_i^j(k))\\ \boldsymbol{b}_i^j(k) & \text{if } f(\boldsymbol{a}_i(k))\le f(\boldsymbol{b}_i^j(k))\end{cases}\tag{6}$$

2.4 基于分层策略的免疫克隆数据简化算法

当待评估数据集的训练样本逐渐增多时,算法的空间和时间复杂度都会随之增加,并且可能会影响最后的简化结果.因此,大部分实例选择算法都无法对大规模数据集进行有效的简化.采用第 2.1 节所示编码方式的免疫克隆数据简化算法的这些缺点显得尤为突出,这是因为用来表示问题解的抗体的编码长度与训练样本的总量是对应的.基于这一问题,我们引入分层策略[9]对编码方式进行改进.分层策略的示意图如图 2 所示.

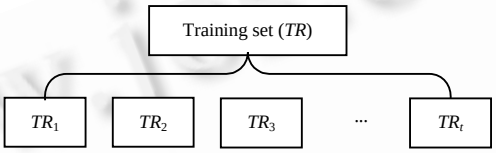

Fig.2 An illustration of the stratified strategy

图 2 分层策略示意图

将原训练集 TR 中的数据均分为互不相交的 t 层,在划分的过程中保持它们的特征属性和类别属性不变.即, $\bigcup_{j\in J}TR_j=TR, J=\{1,2,...,t\}$.此时,将实例选择算法分别作用于 TR_j,进化结束时,选出它的一个最优子集 TRS_j,则最终选出的训练子集 S 就是 t 个 TRS_j 的并集,即 $S=\bigcup_{j\in J}TRS_j, J=\{1,2,...,t\}$.

图 3 给出了基于分层策略的实例选择模型,其中,实例选择算法(instance selection algorithm,简称 ISA)依然采用免疫克隆数据简化算法.设第 j 层有 m_j 个训练样本,则抗体编码长度为 m_j,每个基因位对应 TR_j 中的一个实例.亲和度函数和算子设计与第 2.2 节和第 2.3 节相同.

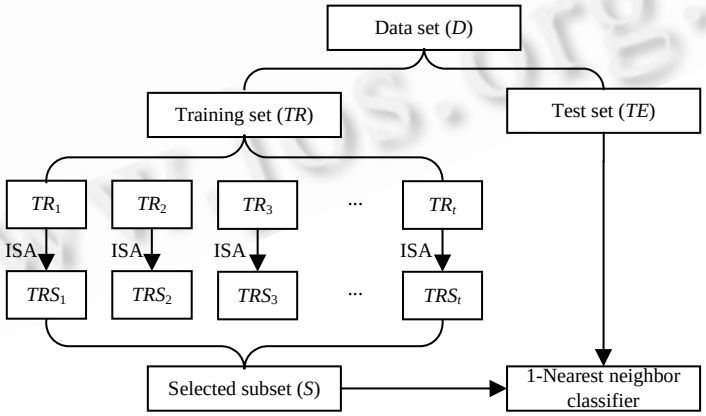

Fig.3 Immune clonal instance selection model based on stratified strategy

图 3 基于分层策略的免疫克隆实例选择模型

3 算法收敛性分析

抗体种群序列{$\boldsymbol{A}(k),k\geq 0$}是一个有限齐次可约马尔可夫链.对于任意初始状态$\boldsymbol{A}_0$,算法以概率1收敛于最优抗体集合$\boldsymbol{A}^*$,证明如下.

定义 1. 定义数据简化问题的最优抗体集合为

$$\boldsymbol{A}^* \triangleq \left\{\boldsymbol{a}^* : \neg\exists \boldsymbol{a} \neq \boldsymbol{a}^*, f(\boldsymbol{a}) > f(\boldsymbol{a}^*)\right\} \tag{7}$$

对于抗体种群$\boldsymbol{A}$,令$\vartheta(\boldsymbol{A}) \triangleq \left|\boldsymbol{A} \cap \boldsymbol{A}^*\right|$表示抗体种群$\boldsymbol{A}$中包含最优抗体的个数.

定义 2. 如果对于任意的初始状态$\boldsymbol{A}_0$,均有

$$\lim_{k\to\infty} P\left\{\vartheta\left(\boldsymbol{A}(k)\right)\geq 1 \middle| \boldsymbol{A}(0)=\boldsymbol{A}_0\right\}=1 \tag{8}$$

则称算法以概率1收敛.

定理 1. 免疫克隆数据简化算法以概率1收敛.

证明:记$P_0(k)=P\left\{\vartheta\left(\boldsymbol{A}(k)\right)=0\right\}$,由贝叶斯条件概率公式有:

$$\begin{aligned} P_0(k+1)&=P\{\vartheta(\boldsymbol{A}(k+1))=0\}\\ &=P\{\vartheta(\boldsymbol{A}(k+1))=0|\vartheta(\boldsymbol{A}(k))\neq 0\}\times P\{\vartheta(\boldsymbol{A}(k))\neq 0\}+\\ &\quad P\{\vartheta(\boldsymbol{A}(k+1))=0|\vartheta(\boldsymbol{A}(k))=0\}\times P\{\vartheta(\boldsymbol{A}(k))=0\} \end{aligned} \tag{9}$$

由公式(6)可得,$P\{\vartheta(\boldsymbol{A}(k+1))=0|\vartheta(\boldsymbol{A}(k))\neq 0\}$,所以,

$$P_0(k+1)=P\{\vartheta(\boldsymbol{A}(k+1))=0|\vartheta(\boldsymbol{A}(k))=0\}\times P_0(k) \tag{10}$$

又由免疫基因操作的性质可知,

$$P\{\vartheta(\boldsymbol{A}(k+1))>0|\vartheta(\boldsymbol{A}(k))=0\}>0 \tag{11}$$

记:$\zeta=\min_k P\left\{\vartheta\left(\boldsymbol{A}(k+1)\right)>0 \middle| \vartheta\left(\boldsymbol{A}(k)\right)=0\right\}, k=0,1,2,\ldots$,则

$$P\left\{\vartheta\left(\boldsymbol{A}(k+1)\right)>0 \middle| \vartheta\left(\boldsymbol{A}(k)\right)=0\right\}\geq\zeta>0 \tag{12}$$

所以,

$$\begin{aligned} &P\left\{\vartheta\left(\boldsymbol{A}(k+1)\right)=0 \middle| \vartheta\left(\boldsymbol{A}(k)\right)=0\right\}\\ &=1-P\left\{\vartheta\left(\boldsymbol{A}(k+1)\right)\neq 0 \middle| \vartheta\left(\boldsymbol{A}(k)\right)=0\right\}\\ &\leq 1-P\left\{\vartheta\left(\boldsymbol{A}(k+1)\right)>1 \middle| \vartheta\left(\boldsymbol{A}(k)\right)=0\right\}\leq 1-\zeta<1 \end{aligned} \tag{13}$$

因此,

$$0\leq P_0(k+1)\leq(1-\zeta)\times P_0(k)\leq(1-\zeta)^2\times P_0(k-1)\leq\ldots\leq(1-\zeta)^{k+1}\times P_0(0) \tag{14}$$

因为$\lim_{k\to\infty}(1-\zeta)^{k+1}=0$,$1\geq P_0(0)\geq 0$,

所以,

$$0\leq\lim_{k\to\infty} p_0(k)\leq\lim_{k\to\infty}(1-\zeta)^{k+1}P_0(0)=0 \tag{15}$$

故,$\lim_{k\to\infty} p_0(k)=0$,因此,

$$\lim_{k\to\infty} P\left\{\vartheta\left(\boldsymbol{A}(k)\right)\geq 1 \middle| \boldsymbol{A}(0)=\boldsymbol{A}_0\right\}=1-\lim_{k\to\infty} P\left\{\vartheta\left(\boldsymbol{A}(k)\right)=0 \middle| \boldsymbol{A}(0)=\boldsymbol{A}_0\right\}\geq 1-\lim_{t\to\infty} P_0(t)=1 \tag{16}$$

所以,$\lim_{k\to\infty} P\left\{\vartheta\left(A(k)\right)\geq 1 \middle| A(0)=A_0\right\}=1$.

于是定理1得证,免疫克隆数据简化算法以概率1收敛. □

4 对比实验与结果分析

4.1 测试数据及实验过程设计

本文实验中用到的前13个数据集均出自UCI标准数据集.最后一个数据集为著名的Kdd Cuo'99数据.当

样本数小于 10^3 时,称其为小规模数据集,当样本数在 10^3~10^5 之间时称其为大规模数据集,当样本数大于 10^5 时称其为海量数据.表 1 给出了文中所用数据集的部分性质.第 1 列是数据名称,第 2 列表示数据集包含的样本数,第 3 列和第 4 列分别给出了数据集的特征维数和类别数.

Table 1 Data sets used in experiments

表 1 实验中使用的数据集

Data sets	Number of samples	Number of features	Number of classes
Glass	214	9	6
Iris	150	4	3
Lymph	148	18	4
Wine	178	13	3
Winsconsin	683	9	2
Monk	432	6	2
Pima	768	8	2
Australian	690	14	2
DNA	2 000	180	3
Thyroid	7 200	21	3
Pen	10 992	16	10
Letter	15 000	16	26
Adult	30 162	14	2
KddCup'99	494 022	41	23

在实验中,将训练样本集 TR 分为 $TR_1,TR_2,\ldots,TR_{10}$ 这样的 10 等份,采用 10 重交叉验证法,在划分时保持数据原有的类别属性不变.这样就得到了 10 对训练和测试样本集(Tr_i 和 Ts_i,i=1,2,…,10).对每一对组合来说,测试样本 Ts_i 就是 TR_i,训练样本 Tr_i 就是 TR 中剩下的 9 组样本的集合,即 $Tr_i=\bigcup TR_j,\ j\in\{1,2,...,t\},\ j\neq i$.

在每次交叉验证时,免疫克隆算法作为实例选择的方法作用于 Tr_i,选出它的一个子集 S_i.然后用 S_i 代替原来的训练样本集 Tr_i,通过最近邻分类器对测试样本 Ts_i 进行分类.我们最后得到的分类正确率是 10 重交叉验证的平均分类正确率.

实验中所用到的参数主要是免疫克隆选择算法中涉及到的,其取值按照经验取值设置如下:抗体规模设为 7,克隆比例大小为 6,变异概率取 0.1.为叙述方便,下文中将采用第 2.1 节所示编码方式的免疫克隆数据简化算法标记为 ICDRA,将采用第 2.4 节所示分层策略的免疫克隆数据简化算法标记为 SICDRA.

4.2 测试结果

4.2.1 基于 ICDRA 的实验

4.2.1.1 小规模数据集测试结果

为了验证本文提出的算法的有效性,首先对 7 个小规模数据集进行简化实验,并与基于遗传算法(GA)的数据简化方法进行对比.为了便于比较,两者的停止准则均设为迭代次数达到 300 代,遗传算法中变异概率取 0.1,交叉概率取 0.8,种群规模为 50.

表 2 中给出的是对原数据集进行归一化处理后的测试结果,使用了 10 重交叉验证策略.由于训练集规模不大,直接用 ICDRA 进行简化.当迭代次数达到 300 代时停止进化,亲和度函数中的参数 λ 取 0.5.表中 *accur* 表示使用最近邻分类器对测试样本分类得到的分类正确率;*clas* 表示交叉验证分类正确率;*per_redu* 表示对原训练样本集的简化率.很明显,对于 Glass,Iris,Lymph 和 Wine 这 4 个数据集使用 ICDRA 简化后的分类正确率虽然比用遗传算法要差,但其简化率均达到了 80%以上,尤其是 Iris 达到了 92%,而用遗传算法得到的简化率均不足 60%.对于 Winsconsin,Monk,Pima 这 3 个数据集,无论是分类正确率还是简化率,ICDRA 都要优于使用遗传算法简化得到的结果.可见,对于数据简化问题,ICDRA 表现出比遗传算法更优的搜索能力,当数据规模增大时,这一优势就更加明显.但是随着数据集规模的逐渐增大,ICDRA 的收敛速度也迅速下降,主要表现为在相同停止准则下简化率有大幅度的下降趋势.针对该问题,我们将在第 4.2.2 节中采用 SICDRA 对大规模数据集进行简化实验,并与 ICDRA 的简化结果进行对比.

Table 2 Test results of small scale data sets

表 2 小规模数据集测试结果

Data sets	*accur*			*clas*		*per_redu*	
	Without reduction	ICDRA	GA	ICDRA	GA	ICDRA	GA
Glass	0.695 6	0.604 73	0.628 38	0.967 86	0.803 57	0.819 05	0.555 95
Iris	0.954 0	0.905	0.917	1	1	0.915 56	0.597 78
Lymph	0.865 8	0.655 21	0.706 25	0.99	0.905	0.89	0.576 11
Wine	0.953 3	0.895 69	0.918 97	1	0.987 5	0.897 22	0.593 06
Winsconsin	0.960 7	0.967 71	0.966 82	0.998 91	0.979 35	0.707 25	0.540 7
Monk	–	0.550 45	0.545 56	0.989 06	0.957 14	0.709 55	0.550 22
Pima	–	0.681 57	0.674 35	0.960 53	0.942 79	0.644 74	0.532 07

4.2.1.2 参数λ对算法性能的影响

由式(1)可以看出,亲和度函数中参数λ的取值直接影响到分类正确率和简化率这两者哪个对亲和度函数贡献较大.很明显,当λ较小时,简化率对亲和度影响较大,反之,当λ取值较大时,分类正确率对亲和度的影响较大.以下是λ取不同值时对算法性能影响的数值实验.在实验中,λ分别取 0.1,0.2,…,0.9,当迭代次数达到 300 代时,停止进化.

图 4(a)~图 4(f)分别代表了数据集 Glass,Iris,Lymph,Wine,Wisconsin 和 Thyroid 的简化实验结果.其中实线代表 *per_redu* 随参数λ而变化的曲线,虚线代表 *clas* 随参数λ的变化规律,而点划线则给出λ增大时 *accur* 的变化规律.从以上实验可以看出,随着λ的增大,*clas* 的值有上升趋势,而 *per_redu* 则不断下降.当λ取值在 0.5 左右时,*accur* 可以达到一个较好的折衷.当数据集规模增大时,这个规律就表现得更加明显.

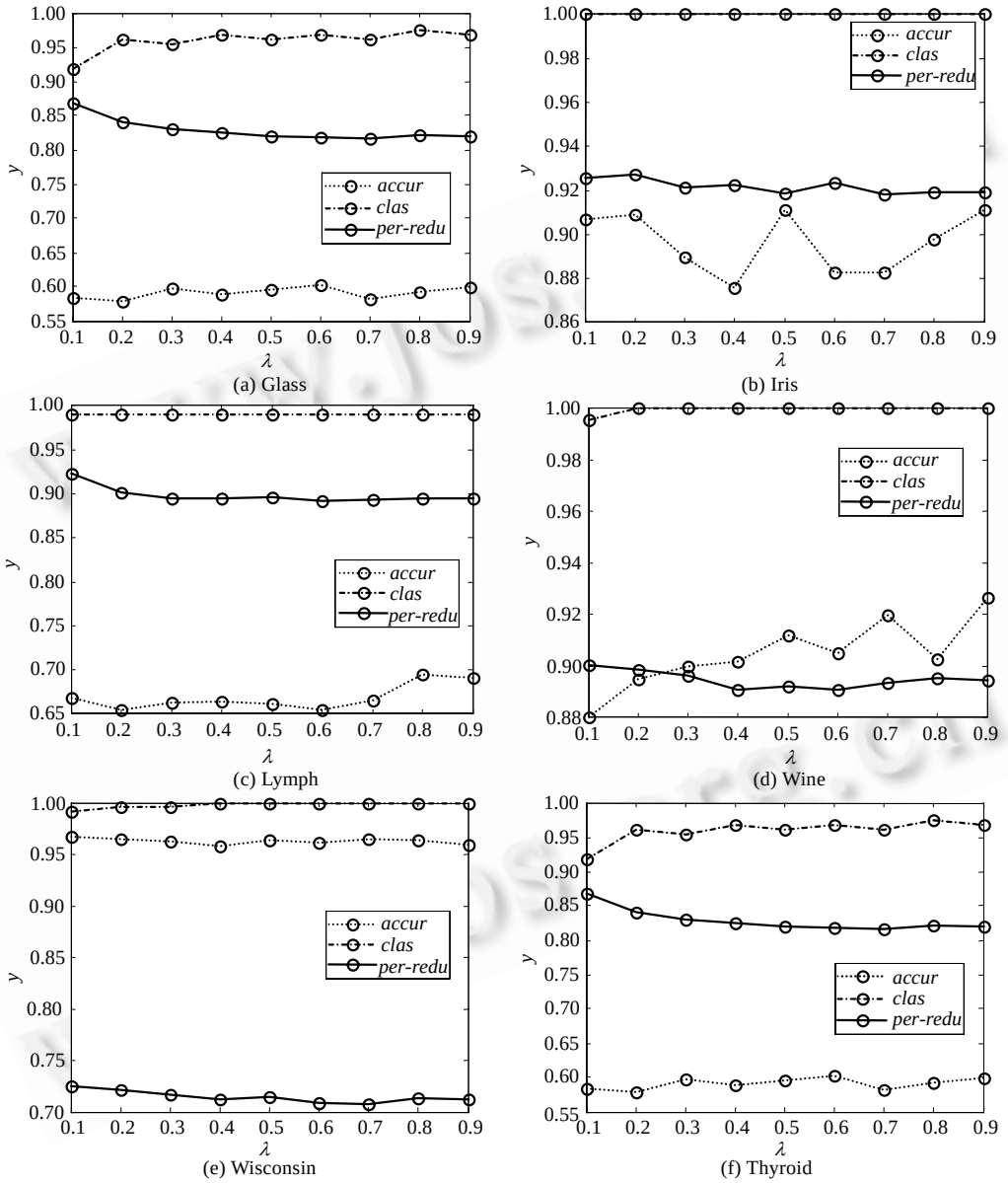

Fig.4 Variation of *accur*, *clas* and *per_redu* with λ

图 4 *accur*,*clas* 和 *per_redu* 随λ发生变化的曲线

4.2.2 基于 SICDRA 的简化实验

4.2.2.1 SICDRA 和 ICDRA 两种算法的对比实验

从第 4.2.1 节中的实验结果可以看出,当数据集规模增大时,最终得到的简化率逐渐减小.主要原因是由于抗体编码长度随数据集规模不断增大,导致收敛速度下降,而此时的停止准则不变,当进化结束时,得到的结果与最优解相差较大.因此在接下来的实验中,我们采用 SICDRA 对几个大规模数据集进行简化实验.对于各不同的数据集层数 t 的取值见表 3.

表 4 给出了两种简化算法的对比实验结果,训练样本规模从 690 增大到 494 022,表中最后一列 Time 指的是算法执行的总时间,它仅从量上给出两种算法的一个对比结果.实验中,对最后 3 个数据量较大的数据集,ICDRA 在规定时间内没有运行完,因此没有得到最后结果,SICDRA 的运行时间也就不再给出了.从前 4 个数据集的简化实验结果很明显地可以看出,两者的 *accur* 值不相上下,但是相对于 ICDRA,SICDRA 得到的 *per_redu* 均提高了 20%以上.此外,后者所用的时间也比前者有大幅度缩减.对于样本规模为 690 的 Australian 数据集,SICDRA 的执行时间约为 ICDRA 的 1/6,而对于样本规模为 10 992 的 Pen 数据集,前者的执行时间约为后者的 1/80.可见,相比 ICDRA,SICDRA 的时间复杂度要低得多,且收敛速度快.当数据规模增大时,这些优势就表现得更加明显.当数据量达到 10^6(Kdd Cup'99)时,SICDRA 的简化率依然能够保持在 75%以上,用选出的训练样本子集对 311 029 个测试样本进行分类,得到的分类正确率为 96.95%.

Table 3 Setting of parameter t

表 3 参数 t 的设置

	Australian	DNA	Thyroid	Pen	Letter	Adult	Kdd Cup'99
t	3	10	15	35	50	50	500

Table 4 Comparison of the two algorithms

表 4 两种方法的对比结果

Data sets	*accur*		*clas*		*per_redu*		Time (s)	
	ICDRA	SICDRA	ICDRA	SICDRA	ICDRA	SICDRA	ICDRA	SICDRA
Australian	0.773 68	0.815 79	0.96	0.979 17	0.666 67	0.822 98	1 854.8	321.25
Thyroid	0.917 01	0.903 44	0.982 76	0.96	0.562 44	0.796 3	1.1428e+05	17 751
DNA	0.702 36	0.709 95	0.845	0.99	0.550 6	0.758 89	1.3095e+05	5 255.3
Pen	0.973 41	0.965 12	0.995 99	0.990 48	0.550 78	0.793 91	4.5052e+05	5 885.2
Letter	–	0.896 4	–	0.846	–	0.676 37	–	–
Adult	–	0.741 43	–	0.877 44	–	0.652 43	–	–
Kdd Cup'99	–	0.969 5	–	0.977 25	–	0.750 71	–	–

4.2.2.2 SICDRA 与几种经典方法的对比结果

在接下来的实验中,将 SICDRA 与其他几种用于数据简化的经典算法进行比较,以 Adult 数据集为例.表 5 所示为实验结果.从该表可以看出,Cnn,Ib2 和 Drop1 得到的简化率都在 90%以上,但是由于简化力度过大,得到的 *accur* 值都在 40%以下,*clas* 的值也都很低.Drop2 算法的 *accur* 和 *clas* 分别达到 83%和 85%以上,但是 *per_redu* 比 SICDRA 要差,只有 60%.可以看出,相对于其他几种方法,本文提出的算法 SICDRA 能够得到简化率和分类正确率两者之间一个很好的折衷,是一种有效的数据简化方法.

Table 5 Comparison results on data set Adult

表 5 数据集 Adult 的对比实验结果

Algorithms	Cnn	Ib2	Drop1	Drop2	SICDRA
accur	0.364 5	0.363 7	0.263 1	0.830 9	0.741 4
clas	0.521 7	0.494 2	0.249 2	0.856 1	0.877 4
per_redu	0.973 4	0.995 7	0.950 9	0.603 3	0.652 4

4.2.2.3 参数 t 对算法性能的影响

在第 4.2.2.1 节的实验中,针对不同规模的数据集分别设定了一个比较合适的 t 值,但并不是最优的 t 值.事实上,t 的取值对算法最终的简化结果和执行时间影响很大,在本节中,以 Letter 和 DNA 数据集作为测试对象进

行实验分析.

图 5 给出了 t 从 30 变化到 120 时对数据集 Letter 的测试结果.图 6 给出了 t 从 2 变化到 30 时对数据集 DNA 的测试结果.其中,图 5(a)中 x 轴和 y 轴分别表示层数 t 和进化代数,z 轴表示亲和度值,图 5(b)的 x 轴和 y 轴与图 5(a)相同,z 轴表示简化率;图 5(c)的横坐标表示层数 t,纵坐标表示算法执行时间;图 5(d)的横坐标与图 5(c)相同,点划线代表测试样本的分类正确率,实线代表简化率.

由图 5(a)和图 5(b)可以看出,随着 t 的增大,算法收敛速度加快,且得到的最终的亲和度值和简化率都有不同程度的增大.由图 5(c)、图 5(d)我们可以看到,随着 t 的增大,算法执行时间 Time 先快速下降,后缓慢下降;分类正确率在 t 小于某一阈值时变化不大,而当 t 超过某一范围后开始下降.在实际应用中,我们应当两者兼顾.设待简化训练样本集数量为 Num,将其分为 t 层,则抗体编码长度 $Length=Num/t$.由上面两图可知,$Length$ 取[100,200]较为合适,即,$t=Num/Length$(100≤$Length$≤200).

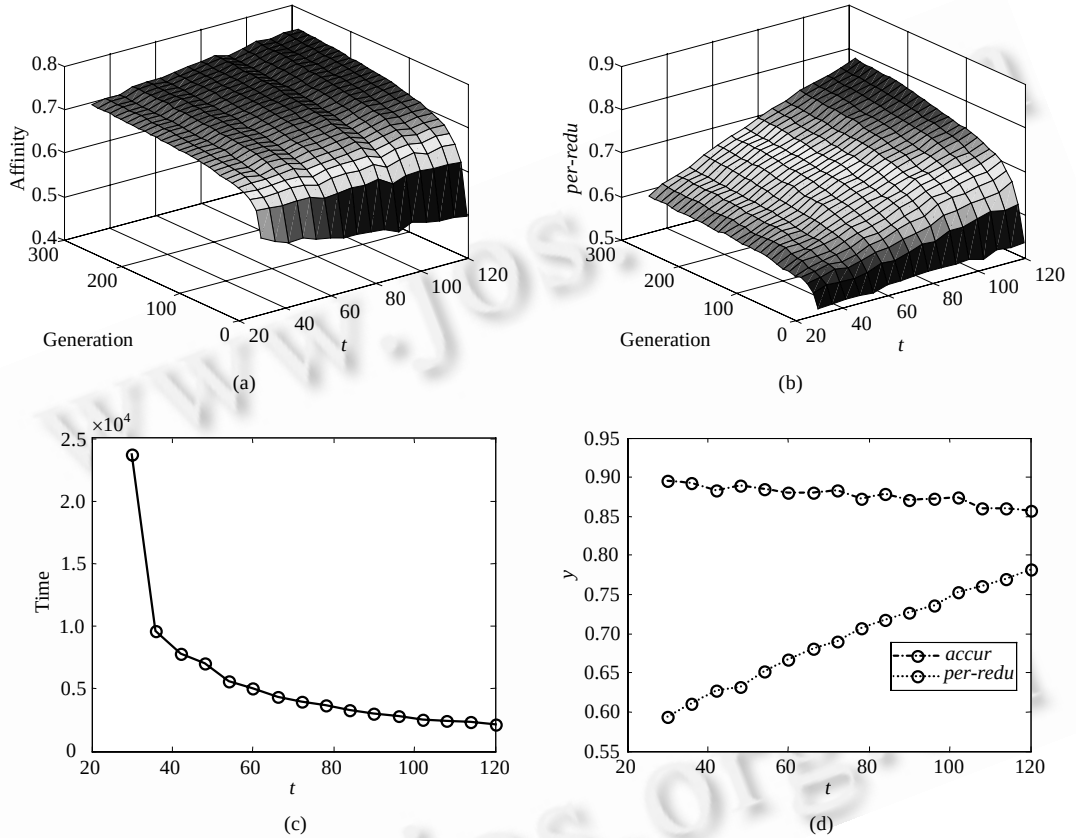

Fig.5 Sensitivity in relation to parameter t (Data set Letter)

图 5 t 参数对算法性能的影响(Letter 数据集)

5 结 论

本文将基于生物克隆选择机理提出的免疫克隆选择算法用于大规模数据简化,提出了一种新的数据简化方法——免疫克隆数据简化算法.并通过对一些具有代表性的小规模标准 UCI 数据集的简化实验证明了该算法的有效性.同时,本文还通过实验分析了权值参数λ的取值变化对算法性能的影响,确定了其最佳取值区间.针对海量数据集简化时算法收敛较慢的问题,引入分层策略对编码方式进行改进,实验结果表明,在对大规模数据进行简化时,相对于原方法,该方法能够大幅度地提高收敛速度,得到更优的结果,最后通过实验给出了层数 t 的

最佳取值区间.

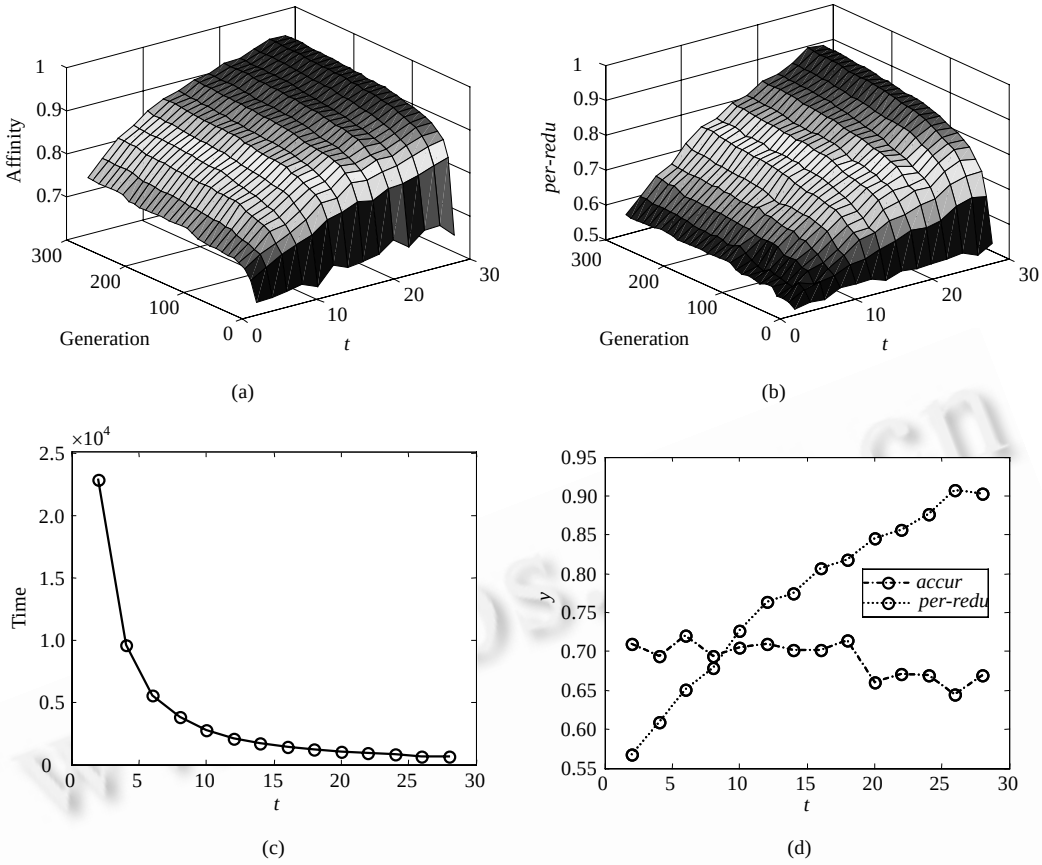

Fig.6 Sensitivity in relation to parameter t (Data set DNA)

图 6 t 参数对算法性能的影响(DNA 数据集)

References:

[1] Liu H, Motoda H. Instance Selection and Construction for Data Mining. New York: Kluwer Academic Publishers, 2001. 3−20.

[2] Takashi F, Akio D. A Study of data reduction method with data accuracy for triangle data. In: Barolli L, ed. Proc. of the 11th Int'l Conf. on Parallel and Distributed Systems. Washington: IEEE Computer Society, 2005. 210−213.

[3] Charu CA. An efficient subspace sampling framework for high-dimensional data reduction, selectivity estimation, and nearest-neighbor search. IEEE Trans. on Knowledge and Data Engineering, 2004,16(10):1247−1262.

[4] Lynch RS, Willett P K. A theoretical performance analysis of the Bayesian data reduction algorithm. In: Proc. of the 2005 IEEE Int'l Symposium on Systems, Man, and Cybernetics. Piscataway: IEEE Systems, Man, and Cybernetics Society, 2005. 330−335.

[5] Tahani H, Plummer B, Hemamalini NS. A new data reduction algorithm for pattern classification. In: Proc. of the 1996 IEEE Int'l Conf. on Acoustics, Speech and Signal Processing. Piscataway: IEEE Signal Processing Society, 1996. 3446−3449.

[6] Cano JR, Herrera F, Lozano M. Using evolutionary algorithms as instance selection for data reduction in KDD: An experimental study. IEEE Trans. on Evolutionary Computation, 2003,7(6):561−575.

[7] Liu H, Motoda H. Feature Selection for Knowledge Discovery and Data Mining. New York: Kluwer Academic Publishers, 1998.

[8] Liu H, Motoda H. On issues of instance selection. Data Mining and Knowledge Discovery, 2002,6(2):115−130.

[9] Cano JR, Herrera F, Lozano M. On the combination of evolutionary algorithm and stratified strategies for training set selection in data mining. Applied Soft Computation, 2006,6(3):323−332.

[10] Hart PE. The condensed nearest neighbor rule. IEEE Trans. on Information Theory, 1968,IT-14(3):515−516.

[11] Kibbler D, Aha DW. Learning representative exemplars of concepts: An initial case of study. In: Shavlik JW, Dietterich TG, eds. Readings in Machine Learning. San Fransisco: Morgan Kaufmann Publishers, 1990. 108−114.

[12] Aha DW, Kibbler D, Albert MK. Instance-Based learning algorithms. Machine Learning, 1991,6(1):37−66.

[13] Wilson DR, Martinez TR. Instance pruning techniques. In: Fisher DH, ed. Proc. of the 14th Int'l Conf. on Machine Learning. San Fransisco: Morgan Kaufmann Publishers, 1997. 403−411.

[14] Skalak DB. Prototype and feature selection by sampling and random mutation hill climbing algorithms. In: Cohen WC, Hirsh H, eds. Proc. of the 11th Int'l Conf. on Machine Learning. San Fransisco: Morgan Kaufmann Publishers, 1994. 293−301.

[15] De Castro LN, Von Zuben FJ. Learning and optimization using the clonal selection principle. IEEE Trans. on Evolutionary Computation, 2002,6(3):239−251.

[16] Jiao LC, Du HF, Liu F, Gong MG. Immunological Computation for Optimization, Learning and Recognition. Beijing: Science Press, 2006 (in Chinese).

附中文参考文献:

[16] 焦李成,杜海峰,刘芳,公茂果.免疫优化计算、学习与识别.北京:科学出版社,2006.

公茂果(1979－),男,山东蒙阴人,博士,副教授,主要研究领域为人工免疫系统,进化算法,数据挖掘,工程优化.

王晓华(1979－),男,博士生,主要研究领域为复杂网络,智能计算.

郝琳(1982－),女,硕士生,主要研究领域为人工免疫系统,数据挖掘.

孙奕菲(1983－),女,博士生,主要研究领域为人工免疫系统,复杂网络.

焦李成(1959－),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为自然计算,智能信息处理.