文章编号:1001 - 9081(2009)06 - 1563 - 03

一种新的用于候选基因排序的数据融合方法

卜令超,王士同
(江南大学 信息工程学院, 江苏 无锡 214122)
(lingchaobu@ yahoo. com. cn)

摘 要:从成百上千的候选基因中确定关键基因是寻找致病基因(或参与某个生物过程的基因)的重要步骤,而根据多种数据源对候选基因进行综合排序则成为该领域新的挑战。提出一种新的基于单类支持向量机的数据融合方法用于候选基因排序。实验表明该方法可以有效地利用多种异构的生物数据源对候选基因排序,其准确率和鲁棒性均优于根据单数据源进行排序。

关键词:候选基因排序; 数据融合; 单类支持向量机
中图分类号: TP181 **文献标志码**:A

Novel data fusion method for candidate gene prioritization

BU Ling-chao, WANG Shi-tong
(*School of Information Technology, Jiangnan University, Wuxi Jiangsu* 214122, *China*)

Abstract: Identifying key candidates in the thousands of genes in a genome is an important step in hunting genes playing roles in a disease phenotype or a complex biological process, and candidate gene prioritization integrating kinds of data sources is becoming a new challenge in this field. A new data fusion method based on one-class Support Vector Machine (SVM) was proposed for candidate gene prioritization. Experimental results indicate that the proposed method is valid in gene prioritization integrating kinds of heterogeneous data sources and its accuracy and robustness are better than that of the method with single data source.

Key words: candidate gene prioritization; data fusion; one-class support vector machine

0 引言

寻找致病基因(或参与某个生物过程的基因)时,通常会先用传统的位置克隆或高通量基因组技术来确定一些候选基因,这些基因的数量大概在几百到几千个;然后再通过生物学实验对这些候选基因进行逐一验证。在进行逐一验证之前,为了节约时间和资金,需要对这些候选基因按其相关性的高低进行排序。随着基因组学的发展,如何有效地确定验证这些基因优先顺序成为制约寻找致病基因速度的瓶颈。

传统上,生物学家们主要依靠学术文献、各种生物学数据库和自己的直觉来确定各个候选基因的优先级。现在,随着各种生物数据的日益丰富,如何高效地利用各种类型的数据对候选基因进行综合排序成为研究的热点。文献[1]先根据序列、科技文献、生物实验数据等对候选基因进行单数据源排序,获得单数据源下的排名(rank),然后用顺序统计(order statistics)的方法计算综合排名(overall rank),并得到了良好的结果;文献[2]中作者则引入了支持向量机数据融合的方法,提高了速度。

支持向量机的方法由于适合处理生物信息学中各种不同类型的带噪声的高维数据,而被广泛地应用于蛋白质同源性检测、微阵列基因表的数据分析、翻译起始位点识别、蛋白质和基因分类等生物信息学领域的各种问题。随着近十年的发展,支持向量机在生物信息学中的应用研究越来越多地转向了数据融合,以集成各种异构的生物数据源。文献[3]研究了相加核矩阵的数据融合方法,文献[4 - 6]基于半定规划(Semidefinite Programming)提出核矩阵加权求和的融合方法,文献[2]的方法可以看作为基于半定规划的方法在解决候选基因排序问题时的变种。

文献[3]根据数据融合与求取核矩阵的先后顺序,将支持向量机的数据融合方法分为三类:早融合(early integration),在数据融合之后求取一个综合的核矩阵;中融合(intermediate integration),先分别根据单个数据源求取核矩阵,然后将核矩阵相加,得到最终的核矩阵;后融合(late integration),先分别根据单个数据源求取核矩阵并计算结果,然后对各个单数据源得到的结果进行融合。文献[4 - 5]和文献[2]的方法可以看作是中融合,即对核矩阵的融合,其关键是确定各核矩阵相加的权重;而文献[1]的方法则类似于后融合,是对各个单数据源求得的排名(rank)结果进行融合,只是在根据单数据源排序时使用的方法不同。

本文考虑到文献[1]中在求取综合排名(overall rank)时可以不必显式地计算,而可以利用支持向量机的方法通过机器学习得到,并且在单数据源候选基因排序时本文也使用基因支持向量机的方法,以统一形式。即先通过单类支持向量机的方法求得各个已知基因和候选基因的得分,然后将已知基因在各个数据源下的得分作为新的训练样本输入,求得候选基因的总得分,从而进行数据融合。

1 单数据源候选基因排序

本文中单数据源候选基因排序使用基于单类支持向量机(One-Class Support Vector Machine, OCSVM)的方法,与文献[2]中的方法类似。在利用单类支持向量机对候选基因进行排序时,基本的策略是根据各种数据将候选基因(测试基因)

收稿日期:2009 - 01 - 04;**修回日期**:2009 - 03 - 02。 **基金项目**:国家自然科学基金资助项目(60773206;60704047)。

作者简介:卜令超(1985 -),男,山东沂源人,硕士研究生,主要研究方向:机器学习、模式识别、生物信息学; 王士同(1964 -),男,江苏扬州人,教授,博士生导师, CCF 会员,主要研究方向:人工智能、模式识别、模糊系统、生物信息学。

与已知的致病基因或参与该病理过程的相关基因(训练基因)进行比较,越"相似",则该候选基因越有可能与该疾病有关,其得分就越高。随着高通量技术的进步,现在有许多数据源可以用来定义"相似",例如基因表达数据:一个基因对应的基因表达数据可以用一个向量 $\boldsymbol{x}$ 来表示,则 $\{\boldsymbol{x}_i\}$ 表示一个训练基因集,将 $\boldsymbol{x}$ 在核空间中到原点的间距定义为得分函数 f,则对任意一个候选基因 $\boldsymbol{x}$,可以用得分 $f(\boldsymbol{x})$ 来定义候选基因 $\boldsymbol{x}$ 与训练基因集 $\{\boldsymbol{x}_i\}$ 的相似度。

单类支持向量机最初由文献[7]中提出,该方法在将训练样本集 $\{\boldsymbol{x}_i\}$ 映射到特征空间后,巧妙的将原点作为负类的代表,通过最大化原点和训练样本之间的最小欧氏距离,来寻找最优超平面将训练样本所对应的点尽量与原点分开。

假设每个基因有一个相关的向量表示 $\boldsymbol{x}$,训练基因集为 $\{\boldsymbol{x}_i \mid \boldsymbol{x}_i \in \mathbf{R}^d, i=1,2,\cdots,l\}$, $d,l \in \mathbf{N}$。该方法寻找一个以单位标准权向量 $\boldsymbol{w}$ 为参数,由等式 $\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}=M$ 定义的超平面,将所有的训练基因与原点分开。对任意的基因 $\boldsymbol{x}$ 定义得分函数 $f(\boldsymbol{x})=\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}$,函数 $f(\boldsymbol{x})$ 度量了在该超平面的法线方向上基因 $\boldsymbol{x}$ 到原点的距离,可以用来对候选基因进行打分:$f(\boldsymbol{x})$ 值越大,说明基因 $\boldsymbol{x}$ 与训练基因越相似,得分越高,排名(rank)越靠前。

如图1所示,实心点表示训练基因,空心点表示候选基因,超平面 P 将训练基因与原点分隔开。

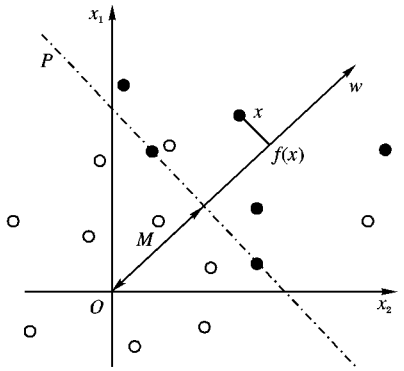

图1 基于OCSVM的候选基因排序二维示意图

下面我们寻找一个最优的超平面使其到原点的间隔 M 最大,寻找该最优超平面需要解决下面的二次线性规划问题:

$$\min\left(\frac{1}{2}\|\boldsymbol{w}\|^2-\rho\right)$$

$$\text{s.t.}\quad \boldsymbol{w}^{\mathrm{T}}\cdot\boldsymbol{x}_i-\rho\geqslant 0 \tag{1}$$

式(1)等价于:

$$\max\rho$$

$$\text{s.t.}\quad \boldsymbol{w}^{\mathrm{T}}\cdot\boldsymbol{x}_i-\rho\geqslant 0;\ \boldsymbol{w}^{\mathrm{T}}\cdot\boldsymbol{w}\leqslant 1 \tag{2}$$

通过拉格朗日方法可得到对偶形式:

$$\min\sqrt{\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{X}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{\alpha}}$$

$$\text{s.t.}\quad \sum_i\alpha_i=1;\alpha_i\geqslant 0 \tag{3}$$

这是一个二次线性规划问题,可以用通用程序求解得到 $\boldsymbol{\alpha}$ 且

$$\boldsymbol{w}^*=\frac{\boldsymbol{X}^{\mathrm{T}}\boldsymbol{\alpha}}{\sqrt{\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{X}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{\alpha}}} \tag{4}$$

记 $\boldsymbol{X}\boldsymbol{X}^{\mathrm{T}}=\boldsymbol{K}$,于是得分函数

$$f(\boldsymbol{x})=\boldsymbol{w}^{*\mathrm{T}}\boldsymbol{x}=\frac{1}{\sqrt{\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{K}\boldsymbol{\alpha}}}\sum_i^n\alpha_i k(\boldsymbol{x},\boldsymbol{x}_i) \tag{5}$$

其中:$\boldsymbol{W}^*$ 表示问题(2)的最优解,$\boldsymbol{X}$ 是训练基因的矩阵表示,$\boldsymbol{X}$ 的第 i 行即训练基因 $\boldsymbol{x}_i$,矩阵 $\boldsymbol{X}\boldsymbol{X}^{\mathrm{T}}$ 的第 i 行、第 j 列位置上是内积 $\langle\boldsymbol{x}_i^{\mathrm{T}}\cdot\boldsymbol{x}_j\rangle$,$\boldsymbol{K}$ 即所谓的核矩阵,$k(\boldsymbol{x},\boldsymbol{x}_i)$ 为核函数。

该得分函数 $f(\boldsymbol{x})$ 将用于根据各单数据源对候选基因进行打分,根据得分可以得到候选基因的排序;同时该函数也可用于对已知基因进行打分,以用于数据融合。

2 基于数据融合的候选基因排序

2.1 算法描述

设有 m 个数据源,n 个训练基因,基因 $\boldsymbol{x}_i$ 在数据源 $\boldsymbol{S}_k$ 下的数据为 $\boldsymbol{x}_i^k$,$\boldsymbol{w}_k$ 表示根据数据源 $\boldsymbol{S}_k$ 训练得到的权重向量。则基因 $\boldsymbol{x}_i$ 的得分向量 $\boldsymbol{s}_i$ 为:

$$(\boldsymbol{w}_1^{\mathrm{T}}\cdot\boldsymbol{x}_i^1,\boldsymbol{w}_2^{\mathrm{T}}\cdot\boldsymbol{x}_i^2,\cdots,\boldsymbol{w}_k^{\mathrm{T}}\cdot\boldsymbol{x}_i^k,\cdots,\boldsymbol{w}_m^{\mathrm{T}}\cdot\boldsymbol{x}_i^m)$$

n 个训练基因可以得到 n 行 m 列的得分矩阵 $\boldsymbol{S}$:

$$\boldsymbol{s}_1:(\boldsymbol{w}_1^{\mathrm{T}}\cdot\boldsymbol{x}_1^1,\boldsymbol{w}_2^{\mathrm{T}}\cdot\boldsymbol{x}_1^2,\cdots,\boldsymbol{w}_k^{\mathrm{T}}\cdot\boldsymbol{x}_1^k,\cdots,\boldsymbol{w}_m^{\mathrm{T}}\cdot\boldsymbol{x}_1^m)$$

$$\boldsymbol{s}_2:(\boldsymbol{w}_1^{\mathrm{T}}\cdot\boldsymbol{x}_2^1,\boldsymbol{w}_2^{\mathrm{T}}\cdot\boldsymbol{x}_2^2,\cdots,\boldsymbol{w}_k^{\mathrm{T}}\cdot\boldsymbol{x}_2^k,\cdots,\boldsymbol{w}_m^{\mathrm{T}}\cdot\boldsymbol{x}_2^m)$$

$$\vdots$$

$$\boldsymbol{s}_i:(\boldsymbol{w}_1^{\mathrm{T}}\cdot\boldsymbol{x}_i^1,\boldsymbol{w}_2^{\mathrm{T}}\cdot\boldsymbol{x}_i^2,\cdots,\boldsymbol{w}_k^{\mathrm{T}}\cdot\boldsymbol{x}_i^k,\cdots,\boldsymbol{w}_m^{\mathrm{T}}\cdot\boldsymbol{x}_i^m)$$

$$\vdots$$

$$\boldsymbol{s}_n:(\boldsymbol{w}_1^{\mathrm{T}}\cdot\boldsymbol{x}_n^1,\boldsymbol{w}_2^{\mathrm{T}}\cdot\boldsymbol{x}_n^2,\cdots,\boldsymbol{w}_k^{\mathrm{T}}\cdot\boldsymbol{x}_n^k,\cdots,\boldsymbol{w}_m^{\mathrm{T}}\cdot\boldsymbol{x}_n^m)$$

对于候选基因 $\boldsymbol{x}$,其得分向量 $\boldsymbol{s}$:

$$(\boldsymbol{w}_1^{\mathrm{T}}\cdot\boldsymbol{x}^1,\boldsymbol{w}_2^{\mathrm{T}}\cdot\boldsymbol{x}^2,\cdots,\boldsymbol{w}_k^{\mathrm{T}}\cdot\boldsymbol{x}^k,\cdots,\boldsymbol{w}_m^{\mathrm{T}}\cdot\boldsymbol{x}^m)$$

带入第1章中介绍的得分函数得到候选基因 $\boldsymbol{x}$ 的最终得分:

$$f(\boldsymbol{s})=\langle\boldsymbol{w}^*\cdot\boldsymbol{s}\rangle=\frac{1}{\sqrt{\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{K}_S\boldsymbol{\alpha}}}\sum_i^m\alpha_i k(\boldsymbol{s},\boldsymbol{s}_i) \tag{5}$$

其中 $\boldsymbol{\alpha}$ 通过优化二次线性规划问题(式(6))求解。

$$\min\sqrt{\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{K}_S\boldsymbol{\alpha}}$$

$$\text{s.t.}\quad \sum_i\alpha_i=1,\alpha_i\geqslant 0 \tag{6}$$

核矩阵 $\boldsymbol{K}_S=\boldsymbol{S}\cdot\boldsymbol{S}^{\mathrm{T}}$,$\boldsymbol{K}_S$ 的第 i 行、第 j 列上的元素为:

$$\begin{aligned}k_{ij}=\boldsymbol{s}_i\cdot\boldsymbol{s}_j^{\mathrm{T}}=&\\ &\boldsymbol{w}_1^{\mathrm{T}}\cdot\boldsymbol{x}_i^1\cdot\boldsymbol{w}_1^{\mathrm{T}}\cdot\boldsymbol{x}_j^1+\boldsymbol{w}_2^{\mathrm{T}}\cdot\boldsymbol{x}_i^2\cdot\boldsymbol{w}_2^{\mathrm{T}}\cdot\boldsymbol{x}_j^2+\cdots+\\ &\boldsymbol{w}_k^{\mathrm{T}}\cdot\boldsymbol{x}_i^k\cdot\boldsymbol{w}_k^{\mathrm{T}}\cdot\boldsymbol{x}_j^k+\cdots+\boldsymbol{w}_m\cdot\boldsymbol{x}_i^m\cdot\boldsymbol{w}_m^{\mathrm{T}}\cdot\boldsymbol{x}_j^m=\\ &\boldsymbol{w}_1^{\mathrm{T}}\cdot(\boldsymbol{x}_i^1\cdot\boldsymbol{x}_j^{1\mathrm{T}})\cdot\boldsymbol{w}_1+\boldsymbol{w}_2^{\mathrm{T}}\cdot(\boldsymbol{x}_i^2\cdot\boldsymbol{x}_j^{2\mathrm{T}})\cdot\\ &\boldsymbol{w}_2+\cdots+\boldsymbol{w}_k^{\mathrm{T}}\cdot(\boldsymbol{x}_i^k\cdot\boldsymbol{x}_j^{k\mathrm{T}})\cdot\boldsymbol{w}_k+\cdots+\\ &\boldsymbol{w}_m^{\mathrm{T}}\cdot(\boldsymbol{x}_i^m\cdot\boldsymbol{x}_j^{m\mathrm{T}})\cdot\boldsymbol{w}_m\end{aligned} \tag{7}$$

本文在数据融合时核函数使用高斯核函数:

$$\begin{cases}k(\boldsymbol{x},\boldsymbol{z})=\mathrm{e}^{-\frac{(\boldsymbol{x}-\boldsymbol{z})^2}{2\sigma^2}}\\ \sigma=10\end{cases} \tag{8}$$

按照文献[3]中的分类,本文提出的 *OCSVM* 数据融合方法是对各分数据源的得分进行融合,可以看作是"后融合";观察 k_{ij} 的表达式最终的核矩阵 $\boldsymbol{K}_S$ 实际上是原始数据进行复杂运算求得的,因此又可以看作是"早融合";而 $(\boldsymbol{x}_i^k\cdot\boldsymbol{x}_j^{k\mathrm{T}})$ 则是由单数据源 $\boldsymbol{S}_k$ 得到的核矩阵 $\boldsymbol{K}_k$ 第 i 行、第 j 列上的元素,即 $\boldsymbol{S}_k$ 是通过对各个单数据源核矩阵进行融合得到,因此该方法又可看作是"中融合"。本文提出的数据融合方法,将文献[3]中三类支持向量数据融合的方法统一起来。

2.2 算法扩展

当训练基因过多(大于数据源个数的10倍)时,该算法性能会出现明显下降,因此提出对该算法的补充。此时将训练基因随机分为两部分:*set*1 和 *set*2。*set*2 中的基因个数可以选为数据源个数的1~2倍,剩余的构成 *set*1。将 *set*1 作为训练样本,根据各单数据源对 *set*2 中的基因以及候选基因进行打分;然后将 *set*2 中基因的得分作为新的训练样本对候选基因

进行综合打分、排序。

*set*1 用于单数据源训练获得各单数据源下的权重 $\boldsymbol{w}$,*set*2 用于求取在各单数据源下的得分矩阵进行数据融合。*set*2 中基因的得分情况反映了在对候选基因排序时各个数据源应该占有的比重,可以作为识别基因的模板,即候选基因中的相关基因的得分情况应该和 *set*2 中基因的得分情况相似。

3　实验及分析

3.1　实验数据来源

实验使用文献[5]中的数据与核矩阵,根据酵母基因的序列、蛋白质交互数据、基因表达数据来寻找编码酵母核糖体蛋白的基因。其中,序列数据使用 Smith-sWaterman、blast 和 Pfam HMM 得分,蛋白质交互数据使用线性(linear)核和扩散(diffusion)核,基因表达数据来自斯坦福大学微阵列数据库(Stanford Microarray Database),对数据的标记来自 MIPS Comprehensive Yeast Genome Database (CYGD)。以上数据可以在 http://noble. gs. washington. edu/proj/sdp-svm/下载,具体情况见文献[5]。

3.2　实验样本集设计

该数据集共包含 1 150 个已标记的基因,其中有 134 个编码参与核糖体组成的蛋白质,标记为 +1,1 016 个与编码核糖体蛋白无关,标记为 -1。本文采用已标记的 1 150 个基因来进行验证性实验:

1)从 1 016 个 -1 基因中随机选取 99 个作为负例训练基因。

2)从 134 个 +1 基因中选取 34 个作为正例测试基因,剩余的 100 个 +1 基因构成训练基因集。

3)从 34 个正例测试基因中每次选出 1 个与第 1)步中选出的 99 个 -1 基因构成测试基因集。

3.3　评价指标

理想情况下,测试基因集中正例测试基因 $\boldsymbol{x}$ 的得分函数值 $f(\boldsymbol{x})$ 应该最大,根据 $f(\boldsymbol{x}_i)$ 由大到小进行排序,则正例测试基因排在第一位。从 34 个正例测试基因中依次选取每个基因进行实验,得到各自的排名(rank),可以构造如图 2 所示的类 ROC(Receiver Operating Characteristic)曲线。曲线的纵轴灵敏度表示 34 次实验中正例测试基因排名在某个阈值之前的频率。

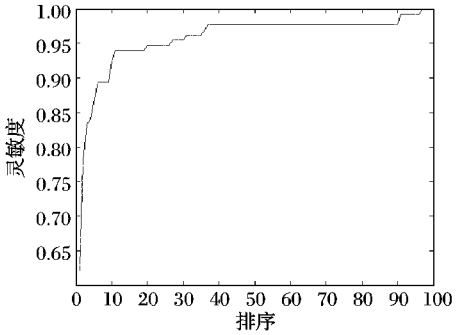

图 2　类 ROC 曲线

类 ROC 曲线下区域的面积(Area Under Curve, AUC)可以作为结果好坏的度量:*AUC* 越接近 1 越好,如果 *AUC* = 1,则表明每个正例测试基因都排在第一位。

3.4　与单数据源排序的对比实验

图 3 为 7 种排序方法的对比实验结果。实验中的数据融合方法为 3.2 节介绍的算法,| *set*1 | = 90,| *set*2 | = 10。实验中各种方法的 *AUC* 如表 1 所示。

表 1　各种排序方法的 *AUC*

候选基因排序方法		*AUC*
单数据源方法	基因表达数据	0.973 5
	Blast	0.857 7
	Smith-Waterman	0.765 9
	Pfam HMM	0.906 2
	蛋白质交互数据 Diff	0.708 2
	蛋白质交互数据 Lin	0.701 2
数据融合方法	核矩阵加和求平均法	0.712 9
	核矩阵加权求和法	0.913 8
	OCSVM 数据融合	0.974 7

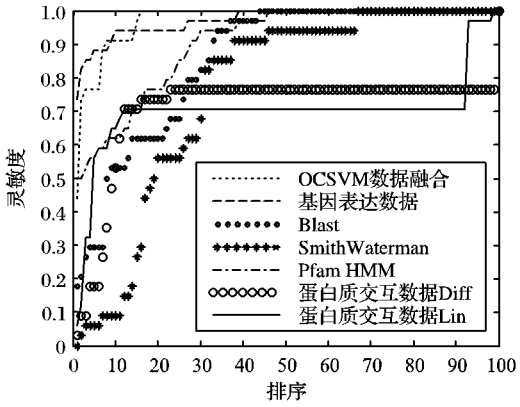

图 3　与单数据源排序的对比

通过实验结果可以看出:该方法可以有效地融合多种数据源对候选基因进行排序,并且较好地排除了较差数据源对较好数据源的影响,使对候选基因排序的鲁棒性增强。

3.5　与其他数据融合方法的对比实验

如图 4 为本文方法与文献[2]中的两种数据融合方法的对比。核矩阵加和求平均法(使用除 Blast 数据外的其他 5 种数据)、核矩阵加权求和法进行对比,各数据融合方法的 *AUC* 见表 1。可以看出,本文提出的 OCSVM 数据融合方法比核矩阵加和的数据融合方法在解决候选基因排序问题方面有优势。

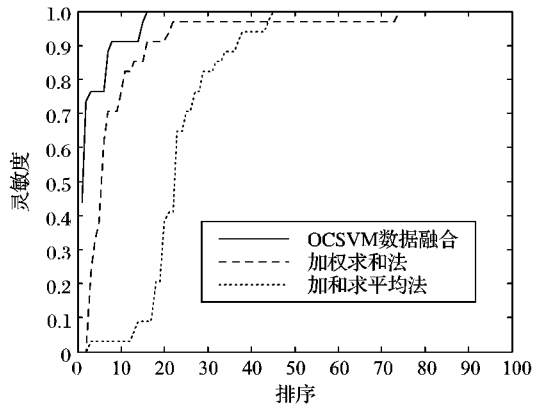

图 4　与其他数据融合方法的对比

4　结语

对候选基因排序是寻找致病基因(或参与某个生物过程的基因)的重要步骤,本文提出一种新的基于单类支持向量机的数据融合方法,有效地利用多种异构的生物数据源对候选基因排序,并通过实验验证了该方法的有效性。进一步的工作将使用更多的数据进行实验,并与其他数据融合方法相比较。

(下转第 1571 页)

该测试数据集样本个数为 178，条件属性个数为 13，聚类类别为 3。类别 1、2 和 3 所包括的样本数分别为 59、71 和 48。初始时，样本的随机分布情况如图 1 所示。

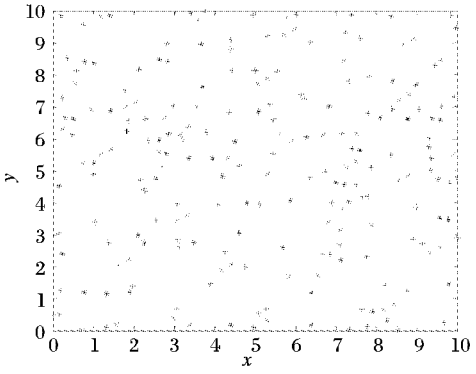

图 1　样本在二维平面随机分布

采用提出的新方法，经过 30 次迭代，移动步长为 1，可视域范围为 2，则动态模糊聚类结果如图 2 所示。

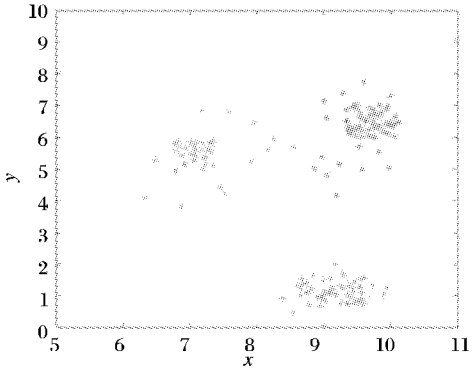

图 2　动态模糊聚类结果

与 FCM 算法、文献[13－14]相比，本文方法其聚类正确率高于 FCM 算法和文献[13－14]的结果，收敛速度也比 FCM 算法和文献[13－14]快。比较结果如表 2 所示。

表 2　几种算法的比较结果

方法	迭代次数	正确率/%		
		类别 1	类别 2	类别 3
FCM	150	90	92	100
文献[13]方法	80	93	98	100
文献[14]方法	80	90	95	100
本文方法	30	95	98	100

3　结语

传统聚类方法的有效性依赖于样本的分布情况，若样本界限分明，则聚类效果好。但是实际情况往往是样本分布呈任意形状簇。对于这类情形，已有的方法效果不佳，本文提出的基于人工鱼群算法的动态模糊聚类方法，通过人工鱼群算法和模糊等价关系矩阵将高维样本映射到二维平面，迭代优化各样本的坐标值，使样本之间的欧氏距离逐步趋近于它们之间的模糊等价矩阵，最终得到全局最优解，动态实现模糊聚类。仿真结果表明，该方法在性能上较经典的模糊聚类算法有一定改进，不依赖于样本特征空间的分布。

参考文献：

[1] JIA WEI-HAN, KAMBER M. 数据挖掘概念与技术 [M]. 范明, 孟小峰, 译. 北京: 机械工业出版社, 2001.

[2] 黄凤岗, 宋克欧. 模式识别[M]. 哈尔滨: 哈尔滨工程大学出版社, 1998.

[3] ZHANG YUAN-QUAN, RUEDA L. A geometric framework to visualize fuzzy clustered data [C]// Proceedings of the XXV International Conference on the Chilean Computer Science Society. Washington, DC: IEEE Computer Society, 2005: 13－17.

[4] 张莉, 周伟达, 焦李成. 核聚类算法[J]. 计算机学报, 2002, 25(6): 587－590.

[5] 高新波, 谢维信. 模糊聚类理论发展及应用的研究进展[J]. 科学通报, 1999, 44(21): 2241－2251.

[6] HATHAWAY R J, BEZDEK J C. Fuzzy c-means clustering of incomplete data [J]. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 2001, 31(5): 735－744.

[7] AGGARWAL C, YU P. Finding generalized projected clusters in high dimensional spaces[C]// SIGMOD'00: Proceedings of ACM SIGMOD International Conference on Management of Data. New York: ACM Press, 2000: 70－81.

[8] 李晓磊, 邵之江, 钱积新. 一种基于动物自治体的寻优模式: 鱼群算法[J]. 系统工程理论与实践, 2002, 22(11): 76－82.

[9] KAMEL S M. New algorithms for solving the fuzzy C-means clustering problem [J]. Pattern Recognition, 1994, 27(4): 14－21.

[10] 赵艳厂, 谢帆, 宋俊德. 一种新的聚类算法: 等密度线算法[J]. 北京邮电大学学报, 2002, 25(2): 8－13.

[11] 李晓磊. 一种新型的智能优化方法——人工鱼群算法[D]. 杭州: 浙江大学, 2003.

[12] 高新波. 模糊聚类分析及其应用[M]. 西安: 西安电子科技大学出版社, 2004.

[13] 郑岩, 黄荣怀, 战晓苏, 等. 基于遗传算法的动态模糊聚类 [J]. 北京邮电大学学报, 2005, 28 (1): 75－78.

[14] 张利彪, 周春光, 马铭, 等. 基于粒子群优化算法的模糊 C-均值聚类[J]. 吉林大学学报: 理学版, 2006, 44(1): 217－222.

(上接第 1565 页)

参考文献：

[1] AERTS S, LAMBRECHTS D, MAITY S, *et al.* Gene prioritization through genomic data fusion [J]. Nature Biotechnology, 2006, 24(5): 537－544.

[2] de BIE T, TRANCHEVENT L C, van OEFFELEN L M M, *et al.* Kernel-based data fusion for gene prioritization [J]. Bioinformatics, 2007, 23(13): 125－132.

[3] PAVLIDIS P, WESTON J, CAI J S, *et al.* Learning gene functional classifications from multiple data types [J]. Journal of Computational Biology, 2002, 9(2): 401－411.

[4] LANCKRIET G R G, DENG M, CRISTIANINI N, *et al.* Kernel-based data fusion and its application to protein function prediction in yeast [C]// Pacific Symposium on Biocomputing. Hawaii, USA: World Scientific, 2004: 300－311.

[5] LANCKRIET G R G, de BIE T, CRISTIANINI N, *et al.* A statistical framework for genomic data fusion [J]. Bioinformatics, 2004, 20(16): 2626－2635.

[6] LANCKRIET G R G, CRISTIANINI N, BARTLETT P, *et al.* Learning the kernel matrix with semidefinite programming [J]. Journal of Machine Learning Research, 2004, 5: 27－72.

[7] SCHOLKOPF B, PLATT J C, SHAWE-TAYLOR J, *et al.* Estimating the support of a high-dimensional distribution [J]. Neural Computation, 2001, 13(7): 1443－1471.