第 39 卷 第 12 期
Vol.39 No.12

计 算 机 工 程
Computer Engineering

2013 年 12 月
December 2013

• 人工智能及识别技术 •　文章编号：1000—3428(2013)12—0181—05　文献标识码：A　中图分类号：TP18

一种基于哈夫曼判定的蛋白质分类方法

何红洲[1]，周明天[2]

(1. 绵阳师范学院数学与计算机科学学院，四川 绵阳 621000；2. 电子科技大学计算机科学与工程学院，成都 611731)

摘　要：已有的仿射传播聚类算法不能很好地反映复杂蛋白质序列本身的聚类结构。为此，提出一种基于哈夫曼判定的蛋白质分类方法。在计算广义置换式匹配相似度的基础上，使用已有的自适应仿射传播算法聚类蛋白质序列。采用哈夫曼编码方法，通过限制平均码长使聚类结果能反映蛋白质序列家族的聚类结构。在蛋白质同源聚类数据库和蛋白质结构分类数据库的 6 个数据集上进行实验，结果表明，该方法与 adAP、谱聚类、SMS 和 TribeMCL 方法相比，不仅能获得更接近于数据集家族的聚类数目及更紧凑的聚类结构，而且 F-measure 指标平均估值分别高出 19.67%、8.7%、9.5%和 43.51%。

关键词：聚类分析；蛋白质序列；广义置换式匹配相似度；仿射传播聚类；哈夫曼判定；F-measure 指标

A Classification Method of Protein Based on Huffman Decision

HE Hong-zhou[1], ZHOU Ming-tian[2]

(1. College of Mathematics & Computer Science, Mianyang Normal University, Mianyang 621000, China;
2. School of Computer Science and Engineering, University of Electronic Science & Technology, Chengdu 611731, China)

【Abstract】Existed Affinity Propagation(AP) clustering algorithm can not reflect the clustering structure of the complex protein sequences, This paper proposes an adaptive AP classification method based on Generalized SMS and Huffman Decision(adAP/GSHD). Protein sequences are clustered via generalized Substitution Matching Similarity(gSMS) and existed adaptive affinity propagation(adAP) algorithm. It uses Huffman coding and confines the average code length of clustering results to embody the family clustering structure of protein sequences. By experiment of test adAP/GSHD and comparing its performance with other four classic clustering methods on six datasets of Clusters of Orthologous Groups(COG) of proteins database and Structural Classification of Proteins(SCOP) database, results demonstrate that this method not only can acquire number of clusters more approximately to the correct family number of clusters and more compact clustering structure for a given set of proteins, but also the average F-measure is 19.67%, 8.7%, 9.5% and 43.81% better than that of adAP, SMS, Spectral Clustering and TribeMCL respectively.

【Key words】clustering analysis; protein sequence; generalized Substitution Matching Similarity(gSMS); Affinity Propagation(AP) clustering; Huffman decision; F-measure index

DOI: 10.3969/j.issn.1000-3428.2013.12.039

1　概述

蛋白质是生命体赖以生存的营养要素，其种类繁多、结构复杂，是细胞组织的重要组成部分，几乎所有的生命过程都与蛋白质发生某种联系。通过分类对蛋白质进行功能预测不仅可以直接阐明生命体在生理和病理条件上的变化机制，而且对生物制药、农业生物科技等应用领域具有直接的指导作用。由于高通量基因组测序使蛋白质数据库的规模快速膨胀，用实验方法去标注蛋白质序列功能的速度远跟不上新蛋白质序列的产生速度。应用计算手段对蛋白质功能进行分类预测成为当前生物信息学的研究热点之一，其方法就是按照某种计量准则将蛋白质序列家族划分为各个功能上相关的蛋白质构成的类，当新的蛋白质以一定的标准指派到某个类时，就可以以更高的可信度将该类的生物学功能赋予该蛋白质序列。其中，根据蛋白质序列的排列顺序和序列信息确定蛋白质的功能成为其研究的重点。

聚类分析能够从整体中发现数据的结构，揭示个体差异和共性，因而在临床诊断决策数据分析、序列分析、微阵列数据分析[1]及基因表达序列数据分析[2]等生命科学领域的分类中得到了广泛的应用。文献[3]综述了各种聚类算法及它们在生物信息学领域的应用与研究进展。

基金项目：四川省教育厅自然科学研究基金资助项目(12ZB070)

作者简介：何红洲(1967－)，男，副教授、博士研究生，主研方向：生物信息数据挖掘；周明天，教授、博士生导师

收稿日期：2013-01-03　**修回日期：**2013-03-01　**E-mail：**zmoonmoonlhm@aliyun.com

通过测量蛋白质序列之间的相似性对蛋白质序列进行有效的划分，为确定蛋白质序列的家族信息和预测蛋白质序列的功能及对蛋白质序列进行同源检测提供了有力的依据。文献[4-5]分别提出了谱聚类方法和 TribeMCL 方法，采用比对软件 BLAST[6]的结果来计算蛋白质序列的相似度；文献[7]提出了无比对的置换式匹配相似度(Substitution Matching Similarity, SMS)方法计算蛋白质序列的相似度，并通过层次化聚类方法生成系统进化树，对进化树节点赋予协相似度值，最后通过阈值来划分不同的蛋白质家族。文献[8-9]提出了 FORCE 方法和 GeneRAGE 方法，使用一个权值的网络来表示蛋白质序列之间的关系，并通过裁剪网络的边来获得一个特定阈值下的分类结果。

本文针对蛋白质序列的特点，通过回顾和分析经典仿射传播聚类(Affinity Propagation Clustering, APC)算法的缺陷，对其加以改进，提出一种基于广义置换式匹配相似度和哈夫曼判定的自适应仿射传播方法。

2 相关研究

相似度的计算是聚类蛋白质序列的必要环节，其准确性直接关系到蛋白质序列聚类结果的好坏。

2.1 有比对方法和无比对方法

FORCE[8]方法使用比对软件 BLAST 并用 E 值负对数最大值作为距离(不相似度)度量；TribeMCL[5]算法采用 BLAST 结果作为序列之间的相似度并构建相似度矩阵及一个加权的转移概率图，最后通过一种模拟的随机行走来对蛋白质序列进行划分；谱聚类[4]对文献[10]的谱聚类算法进行了改进并使用 BLAST 结果作为序列之间的相似度对蛋白质序列进行聚类。这些方法必须有一个前提：序列中的同源片断是互相邻接的。但是这个假设在分子序列中有时并不可靠，因此，目前存在着一些问题，例如对于一些如多域、循环置换(circular permutation)和 tandem 重复等难以比对的蛋白质序列得不到较理想的聚类结果。

SMS 无需比对软件参与，直接基于序列信息计算相似度：首先找出 2 个序列之间长度超过一个阈值的所有确切匹配的子序列，然后基于这些子序列的得分计算出相似度。SMS 简单并且没有任何假设前提，特别适用于多个结构域的蛋白质，但由于只考虑了 2 个序列之间相同的子序列对而没有考虑不相同甚至不匹配的序列，因此对于完全匹配子序列非常少的序列数据集，会导致计算的较大偏差。

2.2 经典的仿射传播聚类方法

APC[11]是 Frey 等人于 2007 年在《Science》上提出的一种聚类分析方法，近年来广泛应用于对生物信息聚类。

APC 以数据集 $\boldsymbol{X}$ 的 N 个特征向量 $\boldsymbol{x}_i(i=1,2,\cdots,N)$之间欧氏距离平方的相反数(即 $s_{ik}=S(\boldsymbol{x}_i,x_k)=-\|\boldsymbol{x}_i-x_k\|^2$)作为相似度，并引入了每一类“类代表”的概念，为最终选出合适的类代表而不断地从数据集中收集证据(初始时每个特征向量都是潜在的类代表)：为候选类代表 x_k 从每个 x_i 收集证据 $r(i,k)$(称为 x_k 对 x_i 的吸引度)来描述 x_k 适合作为 x_i 的类代表的程度；也为 x_i 从候选代表 x_k 收集证据 $a(i,k)$(称为 x_i 对 x_k 的归属度)来描述 x_i 选择 x_k 作为其类代表的适合度。证据越强(即 $r(i,k)$与 $a(i,k)$越大)，x_k 作为最终聚类中心的可能性就越大。APC 通过一个迭代循环不断地进行证据的收集和传递(也称为消息传递)以产生 K 个高质量的类代表和对应的聚类，同时使聚类的能量函数 $E(C)$最小。将各数据点分配给最近的类代表所属的类，则找到的 K 个类就是最后的聚类结果。下面是 $r(i,k)$、$a(i,k)$和 $E(C)$的计算公式(其中式(3)中的 C 是所有聚类的集合，C_i 是任意一个聚类，而 c_i 是聚类 C_i 的类代表)：

$$r(i,k)=s_{ik}+\max_{js.t.\,j\neq k}\{a(i,j)+s_{ij}\} \tag{1}$$

for $all\ i\neq k$:

$$a(i,k)=\min\{0,r(k,k)+\sum_{js.t.\,j\notin\{i,k\}}\max\{0,r(j,k)\}\} \tag{2}$$

$$a(k,k)=\sum_{js.t.\,j\neq k}\max\{0,r(j,k)\} \tag{2'}$$

$$E(C)=\sum_{C_i\in C}E(C_i)=-\sum_{C_i\in C}\Big(\sum_{x_i\in C_i}S(x_i,c_i)\Big) \tag{3}$$

APC 算法中有 2 个重要参数：置于相似度矩阵 $\boldsymbol{S}$ 对角线上的编好参数 $p(k)=s_{kk}=S(x_k,x_k)(k=1,2,\cdots,N)$和迭代中针对 $r(i,k)$和 $a(i,k)$更新的阻尼因子 l_{damp}，$p(k)$(通常小于 0)表示 x_k 被选为聚类中心的先验知识，并对最终作为类代表的聚类中心产生重要的影响。由式(1)可知，当 $p(k)$较大使得 $r(k,k)$较大时，所有的 $a(i,k)$也都较大，即其他 x_i 对数据点 x_k 归属感就越强，从而 x_k 作为最终类代表的可能性就较大。同样，这样的 $p(k)$较多时，最终的聚类数也会较多。由此，对所有的 $p(k)$作适当的调整可以增加或减少 APC 输出的类的数目。在通常情况下，没有哪些点应更倾向于作为聚类中心的先验知识，因此，文献[11]认为初始时取所有的 $p(k)$都相同是可行的，并认为取相似度矩阵 $\boldsymbol{S}$ 所有非对角线元素的中值会达到较为理想的效果；带阻力因子 l_{damp} 的更新规则由式(4)、式(5)给出(式中最左边的 $r_l(i,k)$和 $a_l(i,k)$分别表示 $r(i,k)$和 $a(i,k)$在第 $l(l=1,2,\cdots)$轮迭代的最终结果)，引入阻尼因子是为了适当减慢迭代的步伐以减少振荡从而改进算法的收敛性。

$$r_l(i,k)\leftarrow(1-l_{\text{damp}})r_l(i,k)+l_{\text{damp}}r_{l-1}(i,k) \tag{4}$$

$$a_l(i,k)\leftarrow(1-l_{\text{damp}})a_l(i,k)+l_{\text{damp}}a_{l-1}(i,k) \tag{5}$$

2.3 自适应仿射传播聚类

APC 算法在处理数据集的类数很多时其运算速度较快[12]，这正是聚类蛋白质序列所需要的。但使用时还存在 2 个重要问题：(1)偏好参数的设定与最终的聚类个数没有直接关系，因而找出最优的聚类结果(聚类个数)是一个尚待解决的问题；(2)当振荡发生时如何自动调节阻尼因子而消除振荡的问题。自适应仿射传播聚类[13](Adaptive Affinity Propagation Clustering, adAP)在 APC 算法的执行中引入了自适应调节机制来优化偏好参数及阻尼因子，以及当阻尼

因子效果不佳时使用自适应逃离技术来逃离振荡。

3 adAP/GSHD 方法

将偏好参数设为相似度矩阵非对角线元素的中值是 APC 算法的主要特点，自适应搜索偏好参数空间和自适应调节阻尼因子也体现了 adAP 较 APC 的优越性，但它们不能反映聚类结构比较松散(如蛋白质序列)的数据集的特征。因此，如何从样本数据集中找出更能体现出其家族聚类结构的聚类结果是 APC 和 adAP 算法都未解决的问题。本节给出了基于广义置换式匹配相似度和哈夫曼判定的自适应仿射传播方法(adAP based on Generalized SMS and Huffman Decision, adAP/GSHD)，首先给出 SMS 方法一个变体，称为 gSMS(generalized SMS)来计算蛋白质序列的相似度，以便更能反映各种蛋白质序列数据集的共性和特点；其次使用 adAP 方法对蛋白质序列进行初始聚类；最后通过二元哈夫曼编码的平均编码长度对初始聚类的个数进行限制。

3.1 蛋白质序列相似度计算

本节给出 SMS 方法的一个变体，即广义 SMS(gSMS)方法，具体如下：

设序列 $X=x_1x_2\cdots x_P$ 为长度 P 的蛋白质序列，其中，符号 $x_i(i=1,2,\cdots,P)$取值为任意一种残基，$X_{j\cdots k}=x_jx_{j+1}\cdots x_k$ 表示序列 X 中的第 j 个到第 k 个残基的子序列，$|X|$表示序列 X 的长度。显然对于 2 个区间$[j,k]$和$[u,v]$，如果满足 $j,k,u,v\leqslant|X|$ 且$[j,k]$包含于$[u,v]$，则意味着子序列 $X_{u\cdots v}$ 覆盖了 $X_{j\cdots k}$，反之亦然。

若 A 和 B 为 2 个待计算相似度的蛋白质序列，$\sigma_{a,b}$ 表示 A 和 B 的子序列 a 和 b 形成的序列对，而 $l(a,b)=|\sigma_{a,b}|=\max\{|a|,|b|\}$为 $\sigma_{a,b}$ 的长度。为了区分每个子序列对的相似程度，给每个子序列对定义一个权值 $W(\sigma_{a,b})=\boldsymbol{M}(a_1,b_1)+\boldsymbol{M}(a_2,b_2)+\cdots+\boldsymbol{M}(a_{l(a,b)},b_{l(a,b)})$，其中，$\boldsymbol{M}$ 为按 BLOSUM250 方法给出的记分矩阵，a_i 和 b_i 分别表示序列 a 和 b 的第 i 个残基。$W(\sigma_{a,b})$评估了子序列 a 和 b 之间的相似度，它也可以看成是对这对蛋白质片断的保守性估计。设 $D=\{\sigma_{a,b}||\sigma_{a,b}|\geqslant l\wedge W(\sigma_{a,b})\geqslant e\}$为序列 A 和 B 所有长度不低于 l，相似度得分不低于 e 的最大匹配的子序列对 $\sigma_{a,b}$ 的集合，则 A 和 B 的相似度得分为：

$$S(A,B)=\Big(\sum_{\sigma\in D}W(\sigma)\Big)/\max\{|A|,|B|\} \tag{6}$$

对于 A、B 中剩余的不匹配的子序列 $\boldsymbol{\alpha}$ 和 $\boldsymbol{\beta}$，采用下面的广义欧氏方法来度量其相似度。广义欧氏方法基于序列的特征列向量。为此，首先给出求一个蛋白质序列 $\boldsymbol{Z}$ 特征列向量的方法：称由 L 个残基组成的蛋白质子序列为 L 元组。令 $\boldsymbol{Z}_L=\{Z_1,Z_2,\cdots,Z_K\}$包含 $\boldsymbol{Z}$ 中所有可能的 L 元组($K=20^L$)，使用序列 $\boldsymbol{Z}$ 的 K 个 L 元组重复出现的计数值 $cZ_k(k=1,2,\cdots,K)$ 可将 $\boldsymbol{Z}$ 映射到 Euclidean 空间中的一个 K 维列向量 $\boldsymbol{cZ}_L=(cZ_1,cZ_2,\cdots,cZ_K)^{\mathrm{T}}$，称其为 $\boldsymbol{Z}$ 的特征列向量。

为了度量 $\boldsymbol{\alpha}$ 和 $\boldsymbol{\beta}$ 的相异度，定义 $dE(\boldsymbol{\alpha},\boldsymbol{\beta})=(\boldsymbol{c\alpha}_L-\boldsymbol{c\beta}_L)^{\mathrm{T}}diag(d_1,d_2,\cdots,d_K)^{-1}(\boldsymbol{c\alpha}_L-\boldsymbol{c\beta}_L)$为 $\boldsymbol{\alpha}$ 和 $\boldsymbol{\beta}$ 的广义欧氏距离。其中，$d_1,d_2,\cdots,d_K$为计数向量 $\boldsymbol{c\alpha}_L-\boldsymbol{c\beta}_L$ 的协方差矩阵的对角线元素，引入 $diag(d_1,d_2,\cdots,d_K)^{-1}$ 是为了充分考虑 2 个子序列的相关性。为了将距离转换成相似度，使用负指数函数：$f(d)=e^{-\lambda d}$，其中，d 表示序列 $\boldsymbol{\alpha}$ 和 $\boldsymbol{\beta}$ 之间的广义欧氏距离，而 λ 是一个正的可调参数。

综上所述，序列 A 和 B 的广义 SMS 相似度可以定义为：$s_{AB}=S(A,B)+e^{-\lambda dE(\boldsymbol{\alpha},\boldsymbol{\beta})}$，其中，$S(A,B)$由式(6)给出。

3.2 偏好参数 p 的设定及聚类有效性指标的选择

对于待聚类的包含 N 个蛋白质序列的数据集，按 3.1 节的方法计算它们每个序列之间的相似度(包括自相似度)，取这 $N\times N$ 个相似度的中值作为偏好参数 p，按 adAP 聚类划分的 q 个结果为 Cluster=$\{\boldsymbol{C}_1,\boldsymbol{C}_2,\cdots,\boldsymbol{C}_q\}$。为了评估聚类结果的质量，通常采用定量的有效性指标，其中，Silhouette 有效性指标[14]以其性能好、简单实用而得到了广泛的应用，方法如下：

首先用 3.1 节的方法计算的相似度的倒数来表示 2 个蛋白质序列的距离。对任意 $\boldsymbol{C}\in$Cluster，记 $\boldsymbol{C}=\{C_1,C_2,\cdots,C_R\}$ 表示该划分中的 R 个类。设 N_i 表示第 i 个聚类 C_i 的成员个数，$a_j(s)$为 C_j 中样本点 s 与该类中其他样本点的平均距离，$d(s,C_i)$为该 s 到其余类 C_i 的所有样本的平均距离且 $b_j(s)=\min\{d(s,C_i)\}(i=1,2,\cdots,R$ 且 $i\neq j)$，则 s 的 Silhouettes 指标 $Sil_j(s)$ 及 $\boldsymbol{C}$ 中所有样本的 Silhouettes 指标 $Sil(\boldsymbol{C})$分别为：

$$Sil_j(s)=(b_j(s)-a_j(s))/\max\{a_j(s),b_j(s)\} \tag{7}$$

$$Sil(\boldsymbol{C})=\Big(\sum_{c_j\in C}\sum_{j=1}^{N_j}Sil_j(s)\Big)/N \tag{8}$$

$Sil(\boldsymbol{C})$揭示了聚类结果 $\boldsymbol{C}$ 结构的紧密性(类内密度)和可分性(类间密度)，因而反映了聚类结果的质量。针对不同的聚类结果，其最大值对应类数则为最优的聚类个数。记：$Sil(\boldsymbol{C}_M)=\max\{Sil(\boldsymbol{C}_1),Sil(\boldsymbol{C}_2),\cdots,Sil(\boldsymbol{C}_q)\}$。并令 InitCluster=$\boldsymbol{C}_M$。

3.3 哈夫曼判定

本节通过限制 InitCluster 的二元哈夫曼编码的平均编码长度来组合 InitCluster 中的类，使得聚类的结果更接近于家族的类。哈夫曼编码是通信中对信源进行编码的一种最佳方法，它一方面保证了编码的唯一可译性，另一方面又保证了信源的平均码长最短。这对于优化蛋白质序列的聚类(将每一个聚类当成一个信源)是非常适用的一种方法。一般而言，只要优化后各类的平均二元编码长度不超过 lb$F$(F 表示该蛋白质序列数据集家族类数)，结果就满足要求。哈夫曼判定方法如下(其中，对 InitCluster：K 为其类数；N_k 为其第 k 个聚类 C_k 的序列个数，$p_k=N_k/N=N_k/(N_1+N_2+\cdots+N_K)$为该类序列数占总序列数的比率；$S_k=\sum_{AB\in Ck}s_{AB}$ 为 InitCluster 第 k 个聚类的总相似度；OpCluster=$\{OC_1,OC_2,\cdots,OC_L\}(L\leqslant K)$为优化后的最终聚类结果)：(1)用算法 Huffman(InitCluster)求出 InitCluster 中每一个聚类的哈夫曼编码。过程如下：显然仅有 2 个聚类(K=2 时)的哈夫曼

编码分别应为 c_1=0 和 c_2=1；当 $K\geqslant 3$ 时，若 SortCluster=$\{SC_1,SC_2,\cdots,SC_K\}$ 为对 InitCluster 同时非递增排序 $p_k(k=1,2,\cdots,K)$和非递减排序 $S_k(k=1,2,\cdots,K)$的结果，令 $SC=SC_{K-1}\cup SC_K$ 且 $p=p_{K-1}+p_K$ 从而得到一个聚类个数为 $K-1$ 的聚类结果$\{SC_1,SC_2,\cdots,SC_{K-2},SC\}$，递归调用 Huffman 算法可得到 $SC_1,SC_2,\cdots,SC_{K-2},SC$ 的哈夫曼编码如 $c_1,c_2,\cdots,c_{K-2},c$；在 c 后面分别补 0 和 1 就得到聚类 SC_{K-1} 和 SC_K 的编码 $c_{K-1}=c0$, $c_K=c1$。(2)令 $l_k=length(c_k)(k=1,2,\cdots,K)$表示 c_k 的码长，则 $l_a=p_1l_1+p_2l_2\cdots+p_Kl_K$ 为 SortCluster 的平均码长，若 $l_a\leqslant \mathrm{lb}F$ (F 为聚类数据集家族的类个数)，则算法结束，OpCluster←SortCluster，否则 InitCluster←$\{SC_1,SC_2,\cdots,SC_{K-2},SC\}$，转方法(1)。

4 实验结果与分析

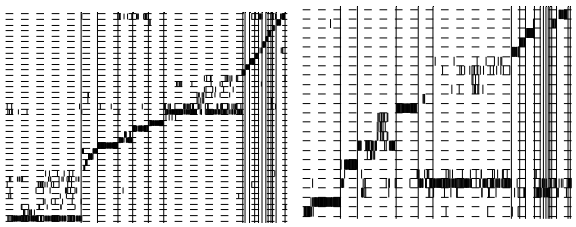
本节分为 2 个部分。首先描述 6 个待测试的蛋白质序列数据集，接着将该方法与 adAP[13]、谱聚类算法[4]、TribeMCL[5]及 SMS[7]算法进行对比实验，并结合 F-measure[4]指标来评估聚类结果与真实分类结果之间的差异。

4.1 蛋白质序列数据集

数据集 C_423[20]、C_487[19]、C_584[19]和 C_765[19]从蛋白质直系同源簇 COG(Clusters of Orthologous Groups of proteins)[15]数据库提取并使用其家族信息作为“真实”的分类结果。COG 数据库是由对细菌、藻类和真核生物的 21 个完整基因组的编码蛋白，根据系统进化关系分类构建而成，对预测单个蛋白质的功能和整个新基因组中蛋白质的功能都很有用；数据集 S_200[7]和 S_512[20]从蛋白质结构分类数据库(Structural Classification of Proteins, SCOP)[16]中随机提取并使用其超家族信息作为“真实”的分类结果。SCOP 数据库由英国医学研究委员会(Medical Research Council, MRC)的分子生物学实验室和蛋白质工程研究中心开发和维护，其目标是提供关于已知结构蛋白质之间的结构和进化关系的信息，对已知三维结构的蛋白质进行分类，并描述它们之间的结构和进化关系，所涉及的蛋白质包括结构数据库 PDB 中的所有条目。各数据集下画线后面的数值表示抽取的蛋白质序列的总数，而后面括号里的数值表示其家族或超家族信息的分类数。

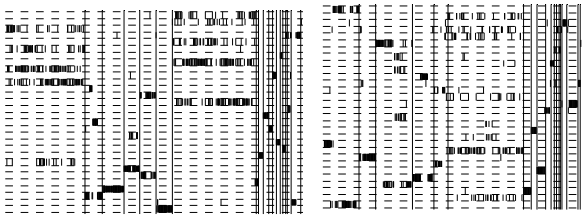
4.2 性能分析

为了展示算法的性能，首先选择 2 个其家族聚类结构最为松散(即分布极不均匀)的数据集 C_765[19]和 C_584[19]，并采用如图 1、图 2 和图 3 对比 adAP/GSHD、谱聚类及 SMS 3 种算法的聚类结果，图中 2 条横虚线之间的部分表示算法获得的一个分类；2 条竖的长实线之间的部分表示参照该家族信息的一个“真实”的类(它们都是 19 类)；2 条虚线之间出现的“短竖线”表示一条蛋白质序列，其具体位置是一个坐标点，横坐标表示其归属的家族，而纵坐标表示算法对它的分类。



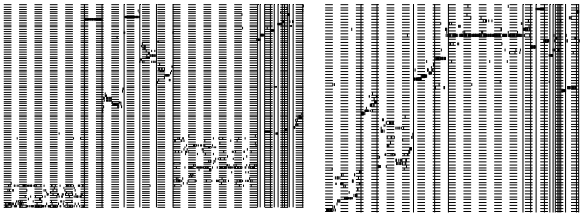
(a)数据集 C_765[19] (b)数据集 C_584[19]

图 1 adAP/GSHD 方法在 2 个蛋白质序列数据集上的聚类结果



(a)数据集 C_765[19] (b)数据集 C_584[19]

图 2 谱聚类算法在 2 个蛋白质序列数据集上的聚类结果



(a)数据集 C_765[19] (b)数据集 C_584[19]

图 3 SMS 算法在 2 个蛋白质序列数据集上的聚类结果

其次采用 F-measure[4]来评估聚类结果与“真实”分类结果之间的差异。对一个给定的蛋白质序列数据集，设 $\boldsymbol{EC}=\{EC_1,EC_2,\cdots,EC_L\}$ 为实验中获得的聚类结果，而 $\boldsymbol{TC}=\{TC_1,TC_2,\cdots,TC_K\}$ 为蛋白质序列集的家族或超家族分类结果，N 为蛋白质序列集中的序列总数，N_i 和 N^j 分别为类 EC_i 和 TC_j 的序列个数，$N_{ij}=|EC_i\cap TC_j|$，则 F-measure 如式(9)所示。显然 F-measure 取值为 0~1 之间，值越大说明聚类的结果越好，值为 1 则表明聚类结果和真实家族分类完全相同。

$$F(\boldsymbol{EC},\boldsymbol{TC})=\frac{1}{N}\sum_{j=1}^{K}\left(N^j\times\max_{1\leqslant i\leqslant L}\frac{2N_{ij}}{N_i+N^j}\right) \tag{9}$$

图 4 对比了 adAP/GSHD 方法、adAP[13]方法、谱聚类算法[4]、TribeMCL[5]及 SMS[7]在 F-measure 上的得分。

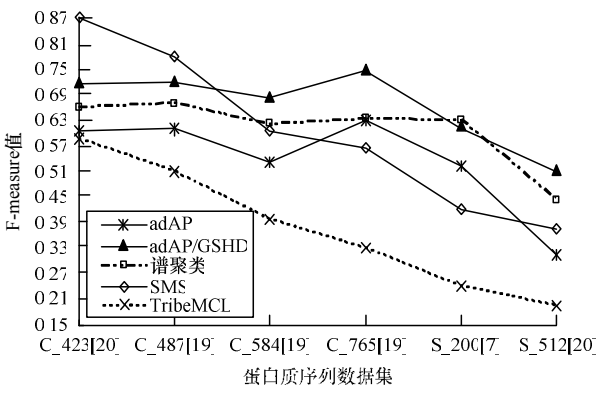


图 4 5 种算法在 6 个数据集上的 F-measure 值比较

其中，TribeMCL 和谱聚类算法的 E-value cut-off 均设置为 0.001，其他参数均采用默认参数。图 5 进一步说明了 adAP/GSHD 算法在揭示蛋白质家族的分类方面相比于其他 4 种算法的优越性及平均性能。

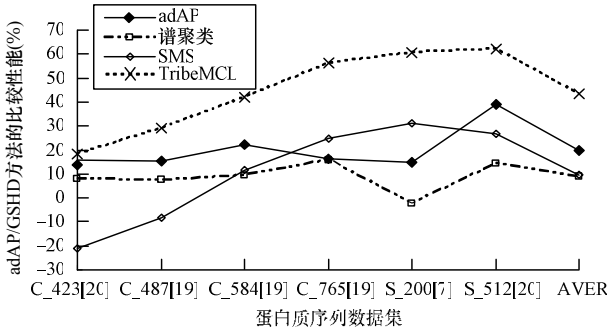


图 5　adAP/GSHD 方法相比于其他方法的优势

对以上结果作如下分析：

(1)聚类结构：比较图 1、图 2 和图 3 可以看出，adAP/GSHD 方法获得的聚类最为紧凑，对 2 个数据集的聚类结果形成了一个清晰的阶梯型，既体现了隶属于同一家族的蛋白质序列的紧密程度，又体现了隶属于不同家庭的蛋白质序列的可分离程度，尤其是对几个大的家族和一些较小的家族聚类结果都是最好的；SMS 算法对 2 个数据集产生了非常多的小类，并且包含一些“孤类”(单个蛋白质序列构成的一个类)，因此整体效果不佳，但是其对第 2、第 3 和第 4 个家族及后面的一些小的家族划分效果较好；adAP/GSHD 对几个大家族除了少量的数据之外在大体上都划分到了一起(参见图 1(a)的第 1 和第 7 个家庭及图 1(b)的第 1、第 3 和第 6 个家族)，而谱聚类算法将几个大家族的数据基本上平分成了几个类(参见图 2(a)的第 1 和第 7 个家庭及图 2(b)的第 6 个家族)，因此聚类结果不够紧凑。

(2)聚类个数：除谱聚类算法对 C_765[19]的聚类外(参见图 2(a))，adAP/GSHD 获得的聚类个数最接近于正确的分类个数，其总体结果要明显优于谱聚类；SMS 算法获得的聚类个数较多且形成了很多成员个数很少的类，只对 C_584[19]最大的一个家族及其后面的几个小的家族聚类效果较好(参见图 3(b))，从整体上看，其聚类效果不佳。

(3)F-measure 值：从图 4 可以看出 adAP/GSHD 算法和谱聚类算法表现较为稳定，对 6 个数据集均取得了较好的聚类效果。SMS 算法除了对 C_423[20]和 C_487[19]取得最高的 2 个 F-measure 值外，对其他几个数据集的聚类效果并不理想，原因是 SMS 算法对数据集的聚类结构较为敏感，表现起伏较大。TribeMCL 的聚类结果产生了大量的“孤类”，与真实的分类差异很大。对比图 4 中第 3、第 4 和第 6 列标记，可以看出 adAP/GSHD 方法对蛋白质序列数据集(特别是对大的非均衡蛋白质序列数据集 C_584[19]、C-765[19]和 S_512[20])的聚类更体现出较大的优势。从图 5 可以看出，adAP/GSHD 除了比谱聚类算法聚类 S_200[7]，SMS 算法聚类 C_423[20]及 C_487[19]有较微弱的劣势外，在绝大多数数据集上比其它方法都有着明显的聚类优势。从 F-measure 值在 6 个数据集上的平均性能上(参照图 5 的最后 1 列标记)，adAP/GSHD 优于 adAP 19.67%，优于谱聚类 8.7%，优于 SMS 9.5%及优于 TribeMCL 43.51%。

5　结束语

本文提出一种基于哈夫曼判定的复杂蛋白质序列分类方法。首先对传统的聚类算法聚类复杂的蛋白质序列进行分析，并指出它们所面临的问题，然后对仿射传播聚类算法进行了改进，即用广义置换式匹配相似度测度蛋白质序列，使用已有的自适应仿射传播算法，并结合 Silhouettes 聚类有效性度量指标和二元哈夫曼编码的平均编码长度去限制和优化聚类结果。实验结果表明，该方法能有效地克服传统方法在聚类蛋白质序列时的缺陷，从而较真实地反映了蛋白质家族的聚类结构。另一方面，在 6 个蛋白质序列数据集上的平均 F-measure 指标也显示出本文方法相对于 adAP、SMS、谱聚类和 TribeMCL 方法的优势。

参考文献

[1] Wang Yanfei, Yu Zuguo, Anh V. Fuzzy C-means Method with Empirical Mode Decomposition for Clustering Microarray Data[C]//Proc. of IEEE International Conference on BIBM. Hong Kong, China: IEEE Computer Society, 2010: 192-197.

[2] Wang Haiying, Zheng Huiru. Clustering-based Approaches to SAGE Data Mining[J]. BioData Mining, 2008, 1(1): 5-17.

[3] Xu Rui, Wunsch D C. Clustering Algorithms in Biomedical Research: A Review[J]. IEEE Review in Biomedical Engineering, 2010, 3: 120-154.

[4] Paccanaro A, Casbon J A, Saqi M A. Spectral Clustering of Protein Sequences[J]. Nucleic Acids Research, 2006, 34(5): 1571-1580.

[5] Enright A J, Van Dongen S, Ouzounis C A. An Efficient Algorithm for Large-scale Detection of Protein Families[J]. Nucleic Acids Research, 2002, 30(7): 1575-1584.

[6] Altschul S F, Gish W, Miller W, et al. Basic Local Alignment Search Tool[J]. Journal of Molecular Biology, 1990, 215(3): 403-410.

[7] Kelil A, Wang S, Brzezinski R, et al. CLUSS: Clustering of Protein Sequences Based on A New Similarity Measure[J]. BMC Bioinformatics, 2007, 8(8): 286-305.

[8] Wittkop T, Baumbach J, Lobo F P, et al. Large Scale Clustering of Protein Sequences with FORCE——A Layout Based Heuristic for Weighted Cluster Editing[J]. BMC Bioinformatics, 2007, 8(10): 396-408.

[9] Enright A J, Ouzounis C A. GeneRAGE: A Robust Algorithm for Sequence Clustering and Domain Detection[J]. Bioinformatics, 2000, 16(5): 451-457.

(下转第 190 页)