PPI 网络的改进谱聚类算法*

吴 爽[a], 雷秀娟[a], 郭 玲[b]
(陕西师范大学 a. 计算机科学学院; b. 生命科学学院, 西安 710062)

摘 要: 蛋白质相互作用(PPI)网络是生物信息学的一个新的研究领域。近年来谱聚类算法在未知蛋白质的功能预测方面发挥了重要作用,但是它要求事先确定聚类数目,为此提出了一种基于边的得分搜索的谱聚类算法。该算法采用谱聚类方法对数据进行预处理,并通过构造蛋白质节点之间的边的得分矩阵找到数据样本之间的相关性,同时融入粒子群算法来确定边的得分的最佳选择阈值,最后用广度优先遍历结点的方法得到聚类结果。算法在 PPI 网络数据集上进行了测试,结果表明该算法不但可以自动确定聚类数目,而且聚类结果的正确率和 F-measure 值都得到了提高。

关键词: 谱聚类算法; 粒子群优化算法; 蛋白质相互作用网络
中图分类号: TP391.4;TP301.6 **文献标志码**: A **文章编号**: 1001-3695(2012)07-2442-05
doi:10.3969/j.issn.1001-3695.2012.07.011

Clustering PPI networks based on improved spectral clustering method

WU Shuang[a], LEI Xiu-juan[a], GUO Ling[b]
(*a. School of Computer Science, b. School of Life Science, Shaanxi Normal University, Xi'an* 710062, *China*)

Abstract: Protein-protein interaction (PPI) network is a new research field in the bioinformatics. Recently spectral clustering algorithm has played an important role in the field of predicting the function of unknown proteins. However, the cluster number must be predefined. With regard to this problem, this paper proposed a spectral clustering algorithm combining with edge-based scoring searching method. Firstly, the algorithm preprocessed the PPI data via spectral clustering, then constructed the scoring matrix of edges connecting protein nodes with each other to find the relationship of dataset, and adopted particle swarm optimization algorithm to determine optimal threshold of the score of edge. Finally, it obtained the clustering results by means of breadth first traversing the protein nodes. Tested this algorithm on the PPI dataset, and the results prove that the algorithm can not only automatically determine the cluster number, but also improve both the precision value and F-measure value.

Key words: spectral clustering algorithm; particle swarm optimization algorithm; protein-protein interaction network

0 引言

随着人类基因组计划的完成以及基因序列和海量数据的增长,由生物体动态产生并执行遗传程序的蛋白质逐渐进入人们的视线[1]。由于生物体的大多数生命活动都是由多个蛋白质通过相互作用进行调控的,单纯地研究单个蛋白质的作用已经无法从整体上把握这种调控结果。因此,目前的研究重点就从单个蛋白质转向由多个蛋白质构成的复杂网络,也就是蛋白质相互作用网络。对 PPI 网络进行聚类研究,可以帮助人们预测功能未知的蛋白质,以此来探究生命活动的机制,为药物设计和病理学的研究提供理论基础。

Watts 等人[2]在 1998 年提出了小世界网络模型,它既具有与规则网络类似的聚类特性,又具有与随机网络类似的较小的平均路径长度。PPI 网络就体现了小世界网络的特性。Barabasi 等人[3]在研究复杂网络特性的时候,发现大多数现实网络的节点的度服从幂率分布规律,即节点的度为 k 的概率大概是 k^{-r}。由于这种幂率分布没有明显的特征长度,所以这样的网络叫做无尺度网络。PPI 网络就是一个无尺度网络。

目前研究 PPI 网络聚类的方法有基于划分的方法、层次聚类算法、基于密度的算法以及基于模型的方法[4]。但是基于层次的方法对于逃逸点很敏感,基于密度的方法会淘汰掉大量稀疏连接的节点。为此产生了一些新颖的算法来分析和研究 PPI 网络。2005 年,Nabieva 等人[5]首先提出了功能流模型,他们认为每一个节点(蛋白质)就像一个蓄水池,它容纳了每次迭代时由该节点流向它的邻接节点的流量,每一条边的权值就代表了它的流量限度。但是这个模型并没有考虑到距离的作用效果。Cho 等人[6]提出了一种基于流的模块化方法来识别重叠的功能模块,该方法可以确定在 PPI 网络中,一个蛋白质 s 在功能上对其他蛋白质产生多少影响。结果表明算法的效率是比较高的,但是聚类结果的查全率和正确率[7]偏低。

近年来产生了谱聚类算法,最初应用在图像分割领域[8]。Shi 等人[9]将像素作为顶点,根据像素的亮度和空间位置确定连接像素点间边的权值,提出了一种新型的评价图像分割优劣的标准——normalized cut,算法取得了比较满意的结果。近年

收稿日期: 2011-12-05; **修回日期**: 2012-01-04 **基金项目**: 国家自然科学基金资助项目(61100164、61173190);陕西省自然科学基础研究计划资助项目(2010JQ8034);中央高校基本科研业务费专项资金资助项目(GK200902016);陕西师范大学研究生创新基金资助项目(2011CXS030)

作者简介:吴爽(1987-),女,河南南阳人,硕士研究生,主要研究方向为数据挖掘、生物信息计算(persistencewu@126.com);雷秀娟(1975-),女,副教授,硕导,博士,CCF 会员,主要研究方向为智能计算与智能优化、生物信息计算等;郭玲(1980-),女,博士研究生,主要研究方向为生物信息学.

来,谱聚类算法已成为了机器学习研究领域的热点之一,其中比较经典的就是 Ng[10] 等人提出的 NJW (Ng-Jordan-Weiss)算法。该算法主要利用拉普拉斯矩阵的最大特征值对应的特征向量得到与数据样本对应的一个低维矩阵,然后通过 K-means 算法得到聚类结果。在相似性矩阵的构造过程中涉及一个关键参数[11],它的选取对最后的聚类结果影响较大。而且 K-means 算法需要事先确定聚类数目,初始聚类中心的选择也会对结果产生影响。

结合 PPI 网络的数据属性,本文在谱聚类算法的基础上,提出了一种改进的基于边的得分搜索的 PPI 网络谱聚类算法。本文简单阐述了带收缩因子的粒子群优化算法和 PPI 网络聚类分析方法中的功能流算法模型,详细叙述了改进的谱聚类算法在 PPI 网络聚类问题上的应用以及算法的具体操作步骤。最后在 PPI 网络数据集上进行了仿真,并与功能流算法进行了比较,结果表明该算法可以自动确定数目,克服了 K-means 算法的缺陷,而且还提高了聚类结果的正确率和查全率。

1　相关介绍

1.1　带收缩因子的粒子群优化算法

粒子群优化算法[12]是一种基于迭代的启发式智能优化算法,虽然它需要调整的参数比较少,但是算法易于陷入局部最优解,因此采用了带收缩因子的粒子群优化算法[13],可以确保粒子能有效搜索到不同的区域,得到高质量的解。首先随机初始化若干个粒子,并用如下公式更新粒子的速度和位置:

$$V_i^{k+1} = \chi \times (V_i^k + c_1 r_1 (P_i^k - X_i^k) + c_2 r_2 (P_g^k - X_i^k)) \tag{1}$$

$$X_i^{k+1} = X_i^k + V_i^{k+1} \tag{2}$$

其中:c_1、c_2 是学习因子;r_1、r_2 是(0,1)上均匀分布的随机数;P_i^k 表示粒子 i 在第 k 次迭代时获得的个体最佳位置;P_g^k 表示第 k 次迭代过程中整个种群的最佳位置;X_i^k 表示粒子 i 在第 k 次迭代中的当前位置;V_i^k 表示粒子 i 在第 k 次迭代中的速度;参数 χ 代表收缩因子,它可以保证算法最终收敛。

1.2　功能流算法

功能流算法[6]是对 PPI 网络聚类分析的一种相对有效的算法,该算法首先对每一个携带信息的蛋白质 s 赋权值,并把它作为最初的流量 $\inf_s(s)$,同时给其他没有携带信息的蛋白质全都赋值为 0。对于每一个携带信息的蛋白质 s 来说,在向它的邻接节点 y 传递流量的过程中,$\inf_s(s)$ 的值会随着对应边的权重的值的改变而逐渐减小:

$$f_s(x \to y) = w_e(s,y) \times \inf_s(s) \tag{3}$$

其中:y 表示蛋白质 s 的邻接节点并且满足条件 $0 \leqslant w_e(s, y) \leqslant 1$,$s$ 在蛋白质 y 上的流量 $\inf_s(y)$ 可以通过计算 y 的所有的邻接节点流入 y 的流量 f_s 的和来进行更新:

$$\inf_s(y) = \sum_{x \in N(y)} f_s(x \to y) \tag{4}$$

蛋白质 s 对于 y 的影响将继续流经网络中所有与 y 连接的边,定义如下:

$$f_s(y \to z) = w_e(y,z) \times \frac{\inf_s(y)}{|N(y)|} \tag{5}$$

算法通过式(4)反复地将所有流入 s 的节点的流量加起来,再通过式(5)把流量传递到所有与之相互连接的节点。最后,两个模块 M_s 和 M_t 之间的相似度 $S(M_s, M_t)$ 可以通过式(6)计算得到:

$$S(M_s, M_t) = \frac{\sum_{x \in M_s, y \in M_t} c(x,y)}{\min(|M_s|, |M_t|)} \tag{6}$$

其中,

$$c(x,y) = \begin{cases} 1 & \text{if } x = y \\ w_e(x,y) & \text{if } x \neq y \text{ and } \langle x,y \rangle \in E \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

重复合并具有最大相似度的两个模块,直到最大相似度低于设定的合并阈值。虽然 Cho 等人的方法是相对高效的,但是它的正确率和查全率是很低的。

2　改进的谱聚类算法

2.1　PPI 网络数据的预处理

2.1.1　谱聚类算法的基本原理

假设给定样本的数量是 n,样本维数是 l,则样本集可以表示为 $S = \{s_1, s_2, \cdots, s_n\} \in \mathrm{R}^l$。首先要构造一个相似性矩阵 $\boldsymbol{A}$,矩阵中的元素 A_{ij} 通过下面的公式来计算[11]:

$$A_{ij} = \begin{cases} \exp(-d^2(s_i, s_j)/\delta^2) & i \neq j \\ 0 & i = j \end{cases} \tag{8}$$

其中:$d(s_i, s_j) = \| s_i - s_j \|$;尺度参数 δ 是用来控制两个样本点 s_i 和 s_j 之间的距离对相似性矩阵 $\boldsymbol{A}$ 的影响。在相似性矩阵的基础上,构造矩阵 $\boldsymbol{D}$ 为度矩阵,其主对角线上的元素 $D(i,i)$ 为相似性矩阵 $\boldsymbol{A}$ 的第 i 行元素之和,其他元素均为 0。接下来构造拉普拉斯矩阵 $\boldsymbol{L}$:

$$\boldsymbol{L} = \boldsymbol{D}^{-1/2} \boldsymbol{A} \boldsymbol{D}^{-1/2} \tag{9}$$

对拉普拉斯矩阵 $\boldsymbol{L}$ 进行特征值分解,并找到它的前 k 个最大特征值所对应的特征向量 $x_1, x_2, \cdots, x_k$,得到矩阵 $\boldsymbol{X} = [x_1, x_2, \cdots, x_k] \in \mathrm{R}^{n \times k}$。然后按照如下公式对特征向量矩阵进行归一化:

$$Y_{ij} = X_{ij} / (\sum_{j=1}^{k} X_{ij}^2)^{1/2} \tag{10}$$

最后把矩阵 $\boldsymbol{Y}$ 的每一行看做是空间中的样本,其中样本数量仍然是 n,样本维数是 k,然后对这些样本用 K-means 方法进行聚类。如果矩阵 $\boldsymbol{Y}$ 的第 i 行被划分到第 j 个聚类,则最初的样本 s_i 就被划分到第 j 个聚类中。

2.1.2　采用谱聚类算法处理 PPI 数据

在 PPI 网络数据集中,每个蛋白质可以看做是一个节点,蛋白质之间的相互作用看做是边的权重。在 PPI 网络数据聚类的过程中,仅仅考虑边的权重信息是不够的,可以采用谱聚类算法得到蛋白质节点之间的一些相关信息。因此首先构造蛋白质节点之间的相似度矩阵。假定在一个 PPI 网络中,节点 i 的邻接结点的集合用 θ_i 来表示,则节点 i 和节点 j 的相似度计算公式[14]如下:

$$\mathrm{sim}(i,j) = \frac{|\theta_i \cap \theta_j|}{|\theta_i \cup \theta_j|} \tag{11}$$

由上述公式计算 PPI 数据集中任意两个蛋白质节点之间的相似度,构成矩阵 $\boldsymbol{A}_{ij}$,其中对角线上的元素均为 0;然后根据式(9)计算得到拉普拉斯矩阵 $\boldsymbol{L}$,并选取 $\boldsymbol{L}$ 中的前 k 个最大特征值对应的特征向量构成矩阵 $\boldsymbol{X}$,这里 k 取 3,是一个经验值;接下来对矩阵 $\boldsymbol{X}$ 归一化之后得到矩阵 $\boldsymbol{Y}$,$\boldsymbol{Y}$ 中的第 i 行代表了第 i 个蛋白质节点的相关信息。

为了得到 PPI 网络中每两个蛋白质节点之间相互作用的信

息,可以根据皮尔逊相关系数和样本的标准偏差来找到节点之间的相互关系,也就是边的得分[15]。得分值越高,节点之间的相关度越高,那么这两个节点成为一个类的可能性就越大;反之亦然。用矩阵 Y 中的第 i 行代表第 i 个蛋白质节点的相关信息,则节点 i 和节点 j 之间的边的得分用如下公式进行计算:

$$\text{score}(i,j) = \text{corr}(Y(i,:),Y(j,:)) \times \text{std}(Y(i,:)) \times \text{std}(Y(j,:)) \tag{12}$$

其中,$\text{corr}(Y(i,:),Y(j,:))$ 是节点 i 和节点 j 之间的皮尔逊相关系数的值,而 $\text{std}(Y(i,:))$ 是节点 i 所对应的那一行数据的标准偏差,根据式(12)构造一个边的得分的邻接矩阵 score。

采用谱聚类算法对 PPI 数据预处理后,得到蛋白质节点之间的相互作用的边的得分。PPI 数据集中包含了每两个蛋白质节点间相互作用的边的权重信息,所以要综合蛋白质相互作用的边的得分和边的权重这两方面信息,将每两个蛋白质之间的边的得分与权重相加,得到一个新的得分矩阵 score。接下来通过广度优先搜索遍历得到聚类结果,这里采用带收缩因子的 PSO 算法来确定遍历 PPI 网络时的选择阈值 T_0。

2.2 算法的编码与解空间设计

2.2.1 PSO 算法的目标函数

在优化选择阈值的过程中,要采用一个目标函数以获得最优解,也就是聚类模块内部的节点之间的相似度越高,聚类模块间的相似度越低,则聚类效果越好。因此 PSO 算法的目标函数就是使最终得到的聚类内的节点之间的边的得分值最高。对于粒子 x,假设最后得到的聚类是 cluster,规模是 $m \times p$,聚类结果的每一行记为 $c_i(i=1,2,\cdots,m)$,每一行的非零元素个数是 $n_i(i=1,2,\cdots,m)$,即 n_1 就是聚类 c_1 中的元素的个数。接下来,按照如下公式计算每个粒子 x 所对应的目标函数:

$$\text{fun}(x) = \sum_{i=1}^{m}\sum_{j=1}^{n_i-1}\sum_{k=j+1}^{n_i}\text{score}(c_i(j),c_i(k)) \tag{13}$$

fun 值越大的粒子的值,就是最佳边的选择阈值 T_0。

2.2.2 编码与解空间设计

在谱聚类算法中,由于 K-means 方法需要事先确定聚类个数,并且对初始的聚类中心的选择也很敏感,因此算法的聚类效果在一定程度上受到了局限。本文在谱聚类算法的基础上,提出了一种改进的基于边的得分搜索的谱聚类(edge based scoring searching spectral clustering,SSC)算法。算法主要根据谱聚类算法的基本原理以及边的得分的计算公式,构造一个蛋白质相互作用的边的得分的邻接矩阵,并采用带收缩因子的 PSO 算法优化边的得分的最佳选择阈值。

由于带收缩因子的 PSO 算法具有快速搜索和全局收敛性能,把边的得分的选择阈值映射成 PSO 算法中的一个粒子,按照该选择阈值进行遍历获得的聚类模块内的蛋白质节点之间的得分的和(即 fun 值)作为算法的优化目标。首先在得分矩阵的最大值和最小值之间初始化若干个粒子,每一个粒子根据式(1)(2)来调整自己的速度和位置,在规定的迭代次数内,不断更新种群个体的局部最优值和全局最优值,最后把 fun 值最高的个体粒子作为边的最佳选择阈值 T_0。

2.2.3 广度优先搜索遍历产生聚类结果

在图 1 所示的 PPI 网络中,用数字 1 ~ 11 编码蛋白质节点,节点之间的连线表示两个蛋白质节点间存在相互作用。

对于得到的边的得分矩阵 score,首先对所有的节点设置访问标记 flag = 0,如图 2 中的第一行列表所示,每个方框中的值代表该节点的访问标记的值;然后从节点 1 开始进行广度优先搜索遍历,如果 $\text{score}(1,i) > T_0$,$(i=2、3、7)$,则把节点 2、3、7 归并到节点 1 所在的聚类中,并修改节点 1、2、3、7 的访问标记 flag = 1,而其他节点的访问标记不作修改,依然是 flag = 0。依此类推,直到 PPI 网络中的所有节点都判断过了,得到第一个聚类模块;然后依次找到第二个访问标记的值为 0 的节点,并从该节点的下一级邻接节点开始搜索,把满足条件的节点都归并到该节点所属的类中,并修改其对应的访问标记。算法一直持续搜索到 PPI 网络中的最后一个节点为止,最终得到若干个聚类模块。

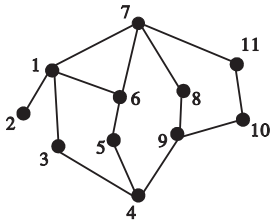

图1 一个PPI网络示意图

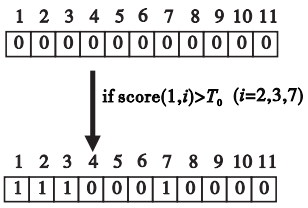

图2 PPI网络的节点遍历方法

2.3 算法的具体实现步骤

a)根据式(11)得到 PPI 网络中的蛋白质节点间的相似度,并采用谱聚类算法来处理相似度矩阵,得到数据维度较低的矩阵 Y。

b)根据式(12)计算蛋白质节点之间的相互作用关系,构造一个边的得分的邻接矩阵 score。

c)初始化带收缩因子的 PSO 算法中的一些参数:最大迭代次数;学习因子 c_1、c_2 以及粒子的运动范围,并随机生成若干个粒子。

d)用式(13)得到当前粒子的适应度,并根据粒子群算法的式(1)(2)不断更新迭代粒子,直到达到最大迭代次数,最后得到的最佳粒子的值就是最佳的边的选择阈值。

e)根据得到的阈值遍历所有的节点,得到聚类结果。

2.4 聚类结果的评价方法

在聚类算法中,一般采用查全率和正确率来评价聚类结果。聚类结果的正确率 precision 和查全率 recall 的计算公式[7]如下:

$$\text{precision}(C|R) = \frac{\text{MMS}(C,R)}{|C|} \tag{14}$$

$$\text{recall}(C|R) = \frac{\text{MMS}(C,R)}{|R|} \tag{15}$$

其中:MMS 表示的是最大匹配的节点的个数;C 代表了匹配上的聚类 C_i 中节点的个数;R 表示的是对应的标准数据集中聚类模块 F_j 的节点个数。一般来说,当聚类模块比较大时,聚类 C_i 中的节点与 F_j 中的节点的交集个数比较多,查全率比较高;反之,如果聚类模块比较小,那么正确率就会非常高。因此,本文改进了这两种评价方法,用 F-measure 值来评价聚类方法的优劣,它是正确率 precision、查全率 recall 以及聚类正确的蛋白质节点的数目 Pnum 的调和平均值:

$$\text{F-measure} = \frac{3}{\frac{1}{\text{precision}} + \frac{1}{\text{recall}} + \frac{1}{\text{Pnum}}} \tag{16}$$

3 仿真结果及分析

3.1 参数分析

本文所采用的 PPI 数据集来自 MIPS 数据库[16]。MIPS 数

据库提供了一些关于 RNA 基因和其他遗传因素的信息,可以作为衡量基准评价算法中得到的蛋白质复合物。实验参数设置如下:在 PSO 算法中,学习因子 c_1、c_2 取值 2.5,收缩因子 $\gamma = 2/|2 - c - \sqrt{c^2 - 4c}|$,其中 $c = c_1 + c_2$。在算法运行过程中,有一些参数的取值是不确定的,可能会影响最终的聚类结果,因此首先讨论这些参数的取值对于实验结果的影响。在确定边的选择阈值过程中采用了带收缩因子的 PSO 算法,其中涉及到的不确定参数有粒子的运动范围的下界 lowbound 以及 PSO 算法的最大迭代次数 maxiter。

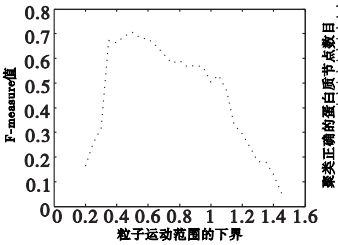

图3　粒子运动范围对 F-measure值的影响

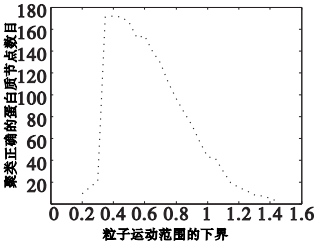

图4　粒子运动范围对聚类正确的蛋白质节点数目的影响

由于粒子的运动范围是在边的得分的邻接矩阵的最小值 0 和最大值 1.47 之间,因此实验过程中粒子运动范围的上界取值为 1.47,下界的取值从 0.2 到 1.45 之间不断发生变化。从图 3 中可以看到,随着粒子运动范围下界的值从 0.2 变化到 0.5 时,聚类结果的 F-measure 值不断变大,达到相对较高的点。而当粒子运动范围的值从 0.5 开始增加时,F-measure 值不断降低。这是由于在确定选择阈值的过程中,如果粒子运动范围的下界的值偏大,则得到的阈值 T_0 也相应地偏高,属于同一个聚类模块的蛋白质数目就会越来越少,可能会有大量的单个蛋白质节点作为一个聚类模块,则聚类结果的正确率和查全率就会大大降低,F-measure 值也会逐渐降低,搜索到的聚类正确的蛋白质节点的数目也会越来越少,从图 4 中也可以看出这一点。结合图 3、4,当粒子运动范围的下界取值 0.5 时,PPI 网络的聚类效果最好。

表 1 给出了聚类结果与粒子运动范围的下界之间的关系。在表 1 中,参数 lowbound 表示粒子运动范围的下界,蛋白质数目代表最后得到的聚类模块中所有的蛋白质节点的数目。从表 1 中可以看出,随着粒子运动范围的下界的值在 0.35 ~ 1.45 变化的过程中,正确率的值基本保持上升趋势,这是由于当粒子运动范围的下界的值变大,在蛋白质节点的遍历过程中,一些得分偏低的边就会被舍弃掉,算法搜索到的蛋白质节点的数目变得越来越少,而且每个聚类模块内部的节点数目也会偏少,如果某个聚类模块 m 与标准数据库的某个模块 n 最大匹配成功,则该模块 m 内的所有节点很有可能都存在于标准数据库中相对应的聚类模块 n 中,因此正确率就会越来越高。但是聚类目标是尽可能多地在短时间内找到聚类正确的未知蛋白质,因此不能单纯地追求正确率高这个目标。从表 1 中可以看出,当 lowbound 的值取 0.5 时,F-measure 值最高,而且仅仅丢失了 37.5% 的蛋白质结点,运行时间也不长。

图 5 反映了粒子群算法的最大迭代次数对聚类效果的影响。从图中可以看出,随着最大迭代次数的增加,聚类结果的正确率和查全率的变化相对平缓,因此该参数对聚类结果的影响不大。为了提高算法的运行效率,实验过程中最大迭代次数设置为 20。

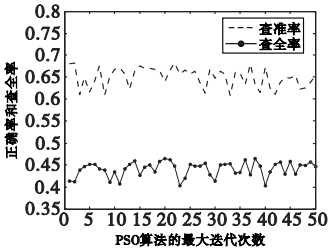

图5　PSO算法最大迭代次数对聚类效果的影响

表 1　聚类结果与粒子运动范围的下界之间的关系表

lowbound	正确率	查全率	聚类数目	蛋白质数目	运行时间/s
0.2	0.389 0	0.211 4	10	73	1 705.3
0.25	0.559 3	0.300 5	13	55	2 046.8
0.3	0.644 1	0.314 6	19	93	1 312.23
0.35	0.563 3	0.477 6	147	860	503.219
0.4	0.547 4	0.473 1	142	874	493.938
0.45	0.597 6	0.480 2	160	828	494.047
0.5	0.646 6	0.473 7	166	758	509.25
0.55	0.673 5	0.444 0	126	563	466.625
0.6	0.665 1	0.441 2	132	602	484.438
0.65	0.674 1	0.415 3	99	438	463.156
0.7	0.651 9	0.395 0	86	412	454.375
0.75	0.662 9	0.364 2	69	321	450.141
0.8	0.721 4	0.376 5	48	224	447.406
0.85	0.678 3	0.379 4	42	187	446.563
0.9	0.701 5	0.411 4	28	138	450.297
0.95	0.706 2	0.466 1	18	98	447.594
1.0	0.699 6	0.456 2	17	79	447.422
1.05	0.787 0	0.501 0	12	61	445.984
1.1	0.886 9	0.517 1	8	41	446.688
1.15	0.733 3	0.409 4	5	26	445.797
1.2	0.888 8	0.405 7	3	16	447.891
1.25	0.8	0.324 1	4	18	447.516
1.3	1	0.335 6	3	11	446.188
1.35	1	0.636 3	1	7	442.406
1.4	1	0.454 5	1	5	441.875
1.45	1	0.181 8	1	2	440.594

3.2　算法性能测试

为了更充分地说明 SSC 算法的优势所在,本文提到的功能流 Flow 算法,它是针对 PPI 网络的拓扑结构特征提出来的一种算法,对这两种算法进行了对比,结果如图 6 所示。

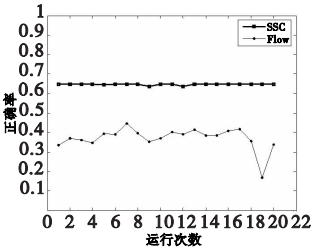

图6　SSC算法与功能流算法的聚类结果在正确率上的比较

图 6 中给出了本文提出的 SSC 算法与功能流 Flow 算法各自运行 20 次的正确率的比较结果。从图 6 中可以看出,SSC 算法的聚类结果的正确率基本保持在 0.65 左右,比较稳定,而 Flow 算法的聚类结果的正确率在 0.15 ~ 0.45 上下波动,因此

Flow 算法的正确率远远低于 SSC 算法的正确率。

为了综合考虑两种算法在正确率、查全率以及聚类数目上的综合性能,图 7 给出了两种算法在聚类结果的 F-measure 值上的比较。在图 7 中,SSC 算法的聚类结果的 F-measure 值依然比较稳定,保持在 0.7 左右,而 Flow 算法的 F-measure 值在 0.2 ~0.45 之间波动。因此,本文中提出的 SSC 算法在聚类结果的正确率上远远高于 Flow 算法;更重要的一点在于 Flow 需要事先确定聚类数目。而 SSC 算法不需要事先确定聚类数目,在对蛋白质节点遍历的过程中,随着聚类模块的不断增加,聚类数目最终自动获得。因此本文提出的 SSC 算法的性能要优于 Flow 算法的性能。

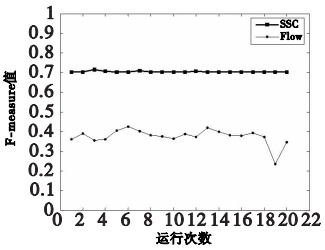

图7 SSC算法与功能流算法的聚类结果在F-measure值上的比较

表 2 给出了采用该算法得到的聚类模块,由于篇幅限制,表 2 只列出了前 10 个聚类模块中聚类正确的蛋白质和聚类错误的蛋白质。从表中可以看出,在聚类模块 1、6、7、8、10 中,实验得到的每一个模块中的所有的蛋白质节点都存在于标准结果数据库对应的聚类模块中,因此每一个模块内的蛋白质都具有相同的功能。而在实验得到的其他聚类模块中,有一部分聚类错误的蛋白质,也就是说这些节点没有出现在标准数据库中与之相对应的聚类模块中。从表 2 的聚类结果中可以有效地识别出一些具有相同功能的蛋白质以及蛋白质复合物,以此来预测未知蛋白质的功能。

表 2 PPI 网络聚类结果的主要聚类模块

模块	聚类正确的蛋白质	聚类错误的蛋白质
1	YGL240w, YFR036w, YHR166w, YDR118w, YBL084c, YKL022c, YOR249c, YNL172w, YDL008w, YLR102c, YLR127c	——
2	YDR176w, YLR055c, YPL254w, YMR236w, YGR252w, YBR198c, YDR392w, YDR448w, YOL148c, YGL112c, YDR167w	YDR145w, YBR081c, YHR009w
3	YML077w, YGR166w, YBR254c, YOR115c, YMR218c, YDR472w, YDR246w, YDR108w, YDR407c	YLR078c
4	YOL139c, YKR059w, YER165w, YOR276w, YJL138c, YGL049c	YGR162w, YOR204w
5	YOR039w, YOR061w, YGL019w, YIL035c	YNL008w, YOR303w
6	YJL014w, YIL142w, YDL143w, YDR212w	——
7	YLR396c, YLR148w, YMR231w, YPL045w	——
8	YNL121c, YMR203w, YGR082w, YNL070w	——
9	YLR277c, YJR093c, YAL043c, YLR115w, YDR301w	YPR107c, YKR002w, YNL317w
10	YGR195w, YOL021c, YDL111c, YCR035c, YHR069c	——

4 结束语

本文在谱聚类算法的基本原理以及蛋白质节点之间的边的得分计算公式的基础上,提出了一种基于边的得分搜索的谱聚类算法,并在 MIPS 数据库中的 PPI 网络数据集上进行了效果测试,结果表明该算法不但可以自动确定聚类数目,而且聚类结果的正确率和查全率得到了很大提高。除此之外,算法的性能也比较稳定,运行时间相对较短。但是该算法的缺陷就是没有完全搜索到 PPI 网络中的所有蛋白质节点并进行聚类,而且算法在处理大量的数据时,时间复杂度可能会比较高,因此下一步的研究工作就是在 PPI 网络数据集的规模比较大的情况下,如何在保持聚类结果正确率的前提下,提高算法的运行效率。

参考文献:

[1] 王正华, 董蕴源, 王勇献. 蛋白质相互作用网络的几种聚类算法综述[J]. 国防科技大学学报,2009,31(4):81-86.

[2] WATTS D J. STROGATZ S H. Collective dynamics of 'small-world' networks[J]. Nature,1998,393:440-442.

[3] BARABÁSI A L, OLTVAI Z N. Network biology: understanding the cell's functional organization[J]. Nature Reviews: Genetics, 2004,5(2):101-113.

[4] 刘昊, 廖波, 彭利红. 基于蛋白质相互作用网络的聚类算法研究[J]. 计算机工程与应用, 2008,44(30):142-144.

[5] NABIEVA E, JIM K, AGARWAL A, *et al*. Whole-proteome prediction of protein function via graph - theoretic analysis of interaction maps[J]. Bioinformatics,2005,21(1):302-310.

[6] CHO Young-Rae, HWANG Woochang, RAMANATHAN M, *et al*. Semantic integration to identify overlapping functional modules in protein interaction networks[J]. BMC Bioinformatics,2007(8):265.

[7] ZHANG Ai-dong. Protein interaction networks[M]. New York: Cambridge University Press,2009: 44-46.

[8] 高琰, 谷士文, 唐琎,等. 机器学习中谱聚类方法的研究[J]. 计算机科学, 2007,34(2):201-203.

[9] SHI J, MALIK J. Normalized cuts and image segmentation[J]. IEEE Trans on Pattern Analysis and Machine Intelligence,2000,22(8):888-905.

[10] NG A Y, JORDAN M, WEISS Y. On spectral clustering: analysis and an algorithm[C]//Advances in Neural Information Processing Systems. Cambridge:MIT Press,2001.

[11] 卜德云,张道强. 自适应谱聚类算法研究[J]. 山东大学学报, 2009,10(5):22-26.

[12] KENNEDY J, EBERHART R. Particle Swarm Optimization[C]// Proc of IEEE International Conference on Neural Networks. 1995: 1942-1948.

[13] 王俊伟. 粒子群优化算法的改进及应用[D]. 沈阳:东北大学, 2006.

[14] QIN Gui-min, GAO Lin. Spectral clustering for detecting protein complexes in PPI networks[J]. Mathematical and Computer Modelling,2010,52(11-12):2006-2074.

[15] GUO Z, LI Yong-jin, GONG Xue, *et al*. Edge-based scoring and searching method for identifying condition-responsive protein-protein interaction sub-network[J]. Bioinformatics, 2007, 23(16): 2121-2128.

[16] GÜLDENER U, MÜNSTERKÖTTER M, KASTENMÜLER G, *et al*. CYGD: the comprehensive yeast genome database[J]. Nucleic Acids Research,2005,33(database issue):D364-D368.