硕士学位论文

基于群体智能的基因表达数据双聚类的研究

RESEARCH ON BICLUSTERING OF GENE EXPRESSION DATA BASED ON SWARM INTELLIGENCE

凡振豪

西南大学

2020 年 2 月

国内图书分类号：TM301.2　　学校代码：10635
国际图书分类号：62-5　　密级：公开

工学硕士学位论文

基于群体智能的基因表达数据双聚类的研究

硕士研究生：凡振豪

导　　师：欧灵副教授

申请学位：工学硕士

学　　科：计算机软件与理论

所在单位：计算机与信息科学学院

答辩日期：2020 年 2 月

授予学位单位：西南大学

Classified Index: TM301.2
U.D.C: 62-5

Dissertation for the Master's Degree in Engineering

RESEARCH ON BICLUSTERING OF GENE EXPRESSION DATA BASED ON SWARM INTELLIGENCE

Candidate: FAN Zhenhao
Supervisor: OU Ling
Academic Degree Applied for: Master of Engineering
Specialty: Computer Architecture
Affiliation: College of Computer and Information Science
Date of Defence: February, 2020
Degree-Conferring-Institution: Southwest University

摘 要

高通量基因微阵列技术的出现, 产生了大量的基因表达数据。这些数据在追踪生物过程, 基因规则发现以及病理分析中有着至关重要的作用。基因表达数据的双聚类是指, 找出在某些条件子集下包含一致表达波动的基因子集。双聚类可以看作是一种多目标优化问题。针对表达数据高维度, 高冗余的特点, 许多群智能算法被用于双聚类中来。

本文基于布谷鸟搜索算法、萤火虫算法和细菌觅食算法等群智能优化算法，从算法结合以及多目标优化等方面进行基因表达数据双聚类的分析研究，意在解决当前双聚类算法的聚类质量差和生物意义不明显等问题。论文的主要工作包括：

(1) 提出基于布谷鸟搜索算法和萤火虫算法的混合双聚类算法（Cuckoo Search and Firfly Algorithm hybrid Biclustering，CSFAB）。通过将布谷鸟搜索算法的全局搜索能力与萤火虫算法的快速收敛能力有效地结合起来，CSFAB 算法可以显著地提高搜索速度和范围，同时能够跳出局部最优解和找到包含不同基因的双聚类，从而提高双聚类的多样性。与 CC、ISA、CSB、FAB 和 PSOB 等算法比较，实验表明 CSFAB 算法的双聚类质量和生物意义更优。

(2) 提出基于多目标细菌觅食算法的双聚类算法（Multi-Object Bacterial Foraging Algorithm Biclustering，MOBFOB）。因为双聚类可以看作多目标优化问题，该算法使用多目标细菌觅食算法同时优化均方残差和体积等双聚类质量评价指标，找到占优的双聚类解集。与 CC、CSFAB、CSB 和 FAB 相比，MOBFOB 算法在计算效率、双聚类的质量评价指标和生物意义等方面获得提高。

关键词：群智能算法; 基因表达数据; 双聚类; 多目标优化

Abstract

The advent of high-throughput gene microarray technology has generated a large amount of gene expression data. These data play a vital role in tracking biological processes, discovering genetic rules, and analyzing pathology. Biclustering of gene expression data refers to finding a subset of genes that contain consistent expression fluctuations under certain conditional subsets. Biclustering can be considered as a multi-objective optimization problem. In view of the high dimensionality and high redundancy of expression data, many swarm intelligence algorithms have been used in biclustering.

Based on swarm intelligence optimization algorithms such as cuckoo search algorithm, firefly algorithm and bacterial foraging algorithm, this paper conducts analysis and research on the biclustering of gene expression data from the aspects of algorithm combination and multi-objective optimization. Problems such as poor quality and insignificant biological significance. The main work of the thesis includes:

(1) A hybrid biclustering algorithm CSFAB(Cuckoo Search and Firfly Algorithm hybrid Biclustering) based on cuckoo search algorithm and firefly algorithm is proposed. By effectively combining the global search ability of the cuckoo search algorithm with the fast convergence ability of the firefly algorithm, the CSFAB algorithm can significantly improve the search speed and range. And at the same time, the algorithm can jump out of the local optimal solution and find biclusters containing different genes, that increasing the diversity of biclusters. Compared with CC, ISA, CSB, FAB, and PSOB algorithms, experiments show that the quality and biological significance of CSFAB algorithm is better.

(2) MOBFOB (Multi-object Bacterial Foraging Algorithm Biclustering) based on a multi-object bacterial foraging algorithm is proposed. Because biclustering can be considered as a multi-objective optimization problem, the algorithm uses a multi-target bacterial foraging algorithm to simultaneously optimize the bicluster's quality evaluation indicators such as mean square residual and volume, and finds the dominant bicluster solution set. Compared with CC, CSFAB, CSB, and FAB, the MOBFOB algorithm has improved the computational efficiency, the quality evaluation index of bi-clustering, and biological significance.

Keywords: Swarm Intelligence Algorithm，Gene Expression Data, Biclustering, Multiple-optimistic

目　录

第 1 章 绪论

1.1 研究背景及意义

随着科技的进步，人类对自然，以及对生命有了更为深刻的认识。从显微镜的发明到 DNA 双链结构的提出，再到如今的 21 世纪，人们越来越多地发现，许多重大疾病跟基因相关，如癌症、一些先天性的心脏病和肥胖症。对于基因的研究成为了解决这些难题的关键。无论是 2003 年的 SARS，还是 2020 年的新型冠状病毒，基因都是研究这些病毒的突破点，搞懂了其基因的组成和作用，将极大地帮助研究人员研制出对应的抗体。每次病毒的大规模扩散，对所在国家的经济和发展都会带来巨大的损失，因此，基因研究是一项任重而道远的，属于全人类的任务。

随着生物实验技术的进步，基因测序变得越来越方便和便宜。在美国，已经有公司推出了亲民的基因测序产品，人们只需花费数百美元就可以测自己的基因，从而可以提前知道将来容易得哪些疾病，并做好预防。在细胞的生物过程中，基因通过转录成信使核糖核酸（mRNA），在不同酶和氨基酸的参与下合成各种各样的蛋白质，这一过程称之为基因的表达。尽管同一生命体中的各个细胞的基因序列是相同的，但不同的细胞仅会在特定的条件下表达特定的极少数基因。所以，研究基因的表达调控是基因组表达分析的重要内容，也有很大的意义，比如，它有助于确认病毒感染和致癌基因，并有助于确认在细胞的各个生命周期内活性基因等。通过高通量基因表达测量技术如微阵列技术（Microarray），可以测得不同 mRNA 的在细胞中的含量。该数据代表着基因的表达水平，因此被成为基因表达数据。

在生物信息学中，对基因表达数据的挖掘是研究热点。通过对基因表达数据的聚类分析，可以得到感兴趣的差异基因。这些基因在不同的实验条件（如样本，时间）下，存在某种一致的表达模式。通过对这些基因的富集或旁路分析，找到这些基因的功能以及相互的调控关系。

然而，基因表达数据有两个主要特点。一，一般而言，基因的数量在几千到几万，而样本或条件的数量只有几十到几百。二，正如前面所说，基因的表达是条件相关的。基因有可能会在多个条件下表达，也有可能在所有的条件下都没有表达。这些特点使得常规的聚类方法无法胜任，于是就引入了双聚类（Bicluster）的概念。不同与常规聚类的全局模式（Global Pattern），双聚类专注于寻找局部模式（Local Pattern），它不要求同一类基因只有在所有实验条件下才具有相似表达，而只要求在部分实验条件下具有相似的表达。找到的基因子集和条件子集就构成了一个双

聚类。

1.2 相关研究进展

双聚类（bicluster）这一单词最早由 Hartigan 于 1972 年提出。直到 2000 年，Cheng 和 Church 将双聚类引入到基因表达数据挖掘中，提出了 CC 算法，并得到了较好的效果。他们提出了 MSR（Mean Square Residue，均方残差）用于评价双聚类相似性。MSR 越小，则表明行相似性和列相似性越高。该算法先通过不断地删除基因节点和条件节点，找到小于事先给定的 MSR 阈值 δ 的双聚类，然后将其作为初始双聚类，在保证 MSR 不会增大的前提下，不断向其中添加基因节点和条件节点，最终得到一个双聚类结果。如果想要多个结果，算法会把之前找到的双聚类使用随机数覆盖，再重复上述操作，直到获得想要数量的双聚类。随机数的引入，会导致结果不准确，而且无法找到重叠的双聚类。2002 年，Yang 等对 CC 算法进行改进，提出了 FLOC 算法。算法从多个初始双聚类出发，根据最大增益的原则，来执行基因和条件节点的删除或增加。但并没有解决贪心策略带来的陷入局部最优解的问题。

2002 年，Ihmels 等人提出了 ISA 算法（Iterative Signature Algorithm，迭代签名算法）。ISA 并没有使用 MSR，而是将双聚类视为转录模块，然后通过对其进行打分，迭代地修改基因集和条件集，直到无法再继续修改。同年，Tanny 等提出了 SAMBA（Statistical Algorithm Method For Bicluster Analysis）算法。该算法使用了图论和统计学的知识，将基因表达数据看作一个二分图，一边为基因，一边为条件。通过寻找最大稠密子图的方式来找到基因表达数据中的最大子矩阵，即双聚类。

由于双聚类问题是 NP 难问题，通过贪婪或枚举的方法很难高效地得出结果，人们开始使用元启发式算法，如群智能算法来寻找双聚类。2004 年，stanfan 等人设计了一个基因表达数据双聚类分析的进化算法框架。Ken-neth Bryan 等人于 2006 年提出了基于模拟退火算法的双聚类算法，并通过实验证明所提方法获得了质量较优的双聚类。2006 年 Mitra 提出了多目标进化双聚类的框架，应用经典的非支配排序遗传算法（Non-dominated Sorting Genetic Algorithm-II，NSGA-II）算法，整合局部搜索策略，并提出新的定量度量方法估计双聚类的质量。2007 年，Goiraldez 应用最大标准区域（Maximal Standard Area，MSA）作为度量双聚类的标准，并和 MSR 一起应用到多目标进化算法 MOEA 中，提出多目标连续变化双聚类算法（Sequential Multi objective Biclustering，SMOB），有效解决成比例模式的双聚类问题。2013 年 Carlos A. Brizula 提出了一种改进的多目标遗传双聚类算法（Enhanced

Multi-objective Genetic Biclustering，EMOGB），与其它多目标双聚类算法不同的是他采用了一种基因和实验条件组来代表一个双聚类的编码方式，并且在搜索双聚类的过程中减少了局部搜索环节，从而算法的执行效率有了很大的提高。

1.3 本文的研究内容及组织结构

本文首先对基因表达数据和双聚类分析等相关基础知识进行阐述。然后从单目标优化，混合优化和多目标优化等方面，研究了以群智能算法如布谷鸟搜索算法、萤火虫算法和细菌觅食算法为框架的双聚类算法。最后对本文的工作进行了总结和展望。本文各章节内容安排如下:

第一章从生物背景知识出发，讨论了基因研究的重要性以及双聚类的作用和发展现状。双聚类克服了常规聚类与基因表达数据之间的矛盾，能更好的挖掘出有价值的基因子集和条件子集。群智能算法由于其快速且效果更好，在双聚类分析中得到了很广泛地应用。

第二章先是更具体地讨论的基因表达数据的数学形式，以及双聚类的定义、类型和结构。接着将双聚类算法分为了基于质量评价指标和基于模型两种。然后，对于双聚类结果，讨论了其质量验证指标和生物验证指标。最后分别介绍了本文所关注的群智能算法。

第三章

第四章

第五章对全文工作进行了总结和展望。

第 2 章 基因表达数据的双聚类相关概述

2.1 基因表达数据

数据的好坏在很大程度上决定了结果的好坏。基因表达数据主要通过基因芯片技术和下一代测序技术等这些高通量基因表达测量技术获得。这些技术可以同时在不同的样本或条件下，对成千上万的基因进行高效、精确、定量地测量。这时只是得到了一些原始数据，由于在特定条件下只有很少数的基因会表达，里面会存在很多缺失数据。需要对原始数据进行缺失值和去噪处理，才能得到适合数据挖掘的数据。

基因表达数据极其庞大，以及需要专业的生物知识，导致很多跨领域的研究者很难涉足。为了打破这种学科壁垒，科学家们提出了关于描述和存储基因表达数据的标准，并在标准之上建立了基因表达数据库。最广泛的数据库为 GEO（Gene Expression Omnibus），由美国国家生物技术信息中心于 2000 年开发。该数据库提供了共享基因表达数据的平台，并且有专业的人员进行审核。公共数据库的出现，极大促进了生物信息学的发展。

基因表达数据一般以一个二维矩阵 E 表示，一行代表一个基因，一列代表一个实验条件。实验条件包括不同时期，不同组织，不同个体，不同外部环境等等。矩阵 E 中的每个元素 e_{ij} 表示基因 g_i 在实验条件 c_j 下的表达水平值，其生物含义是该基因在此条件下，细胞中 mRNA 的含量。矩阵 E 中的每一行被称为基因在该基因表达数据上的全局表达模式（Expression Pattern）。矩阵 E 中的每一列被称为条件在该基因表达数据上的全局表达描述（Expression Profile）。

2.2 双聚类的相关概念

2.2.1 双聚类的定义

给定一个大小为 $n \times m$ 基因表达数据 $E(X,Y)$，假定集合 $I \subseteq X, (|I| = k \leq n)$ 是 E 的基因集合 X 的子集；集合 $J \subseteq Y, (|J| = l \leq m)$ 是 E 的基因集合 Y 的子集。如图2-1所示，双聚类是指在条件子集 J 下的基因表达模型表现出同源特性的基因子集 I。因此，双聚类可以定义为一个 $k \times l$ 的子矩阵 $B(I, J)$，也简称为 B。

2.2.2 双聚类的类型

给定一个二维矩阵 $A(I, J)$，a_{ij} 为其中第 i 行第 j 列的值，α_i 是一个与行有关而与列无关的变量，β_j 是一个与列有关而与行无关的变量，$0 \leq h, r, t, d \leq |I|$ 或

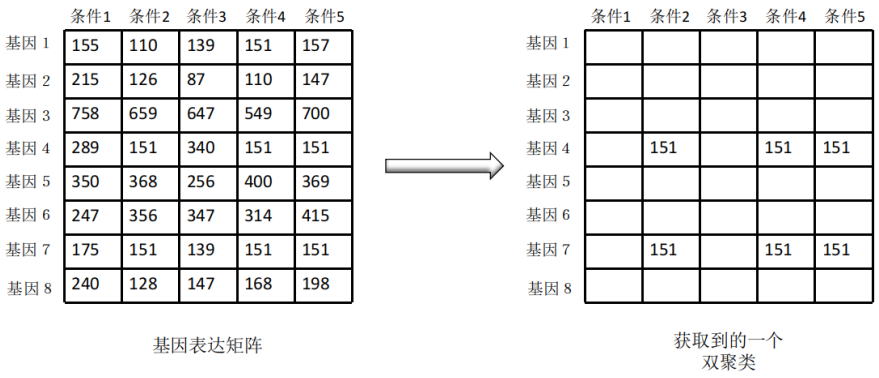


图 2-1 双聚类定义示例

$0 \le h, r, t, d \le |J|$。Madeira 和 Oliveira 提出，在双聚类中主要有以下四种结构：

1. 具有相同常量值的双聚类。该类型的双聚类所有的元素为同一个常量，如图2-2 a)所示，公式如下。

$$a_{ij} = \mu \tag{2-1}$$

2. 列或行具有相同常量值的双聚类，满足公式2-2的双聚类属于行常量值双聚类，如图2-2 b)所示。满足公式2-3的双聚类属于列常量值双聚类，如图2-2 c)所示。

$$a_{ij} = \mu + \alpha_i 或 a_{ij} = \mu * \alpha_i \tag{2-2}$$

$$a_{ij} = \mu + \beta_j 或 a_{ij} = \mu * \beta_j \tag{2-3}$$

3. 数值一致的双聚类，如图2-2 d)和2-2 e)所示，满足的公式如下。

$$a_{ij} = \mu + \alpha_i + \beta_j \tag{2-4}$$

$$a_{ij} = \mu * \alpha_i * \beta_j \tag{2-5}$$

4. 具有连贯演变的双聚类，如图2-2 f)所示，公式如下。

$$a_{ih} \le a_{ir} \le a_{it} \le a_{id} \tag{2-6}$$

$$a_{hj} \le a_{rj} \le a_{tj} \le a_{dj} \tag{2-7}$$

2.2.3 双聚类的结构

双聚类的结构是指，通过算法找到的双聚类之间在原始矩阵时间的相对位置。根据结构，大体可以分为 8 类:

(1) 单一结构。指基因表达数据中只存在一个双聚类，且基因和条件可以不属

1	1	1	1
1	1	1	1
1	1	1	1
1	1	1	1
1	1	1	1

a)

1	1	1	1
2	2	2	2
3	3	3	3
4	4	4	4
5	5	5	5

b)

1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4

c)

1	0	3	5
3	2	5	7
5	4	7	9
2	1	4	6
8	7	10	12

d)

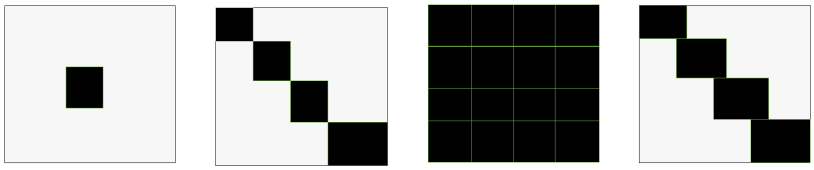
1	2	6	3
3	6	18	9
5	10	30	15
2	4	12	6
7	14	42	21

e)

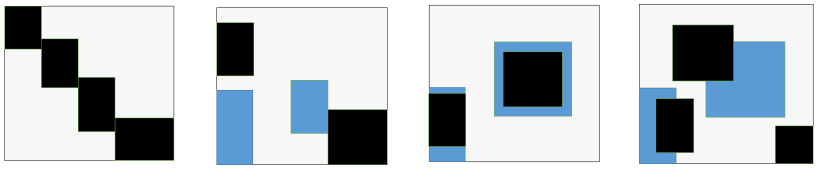
69	12	19	9
47	39	47	34
40	20	28	15
89	18	21	13
50	38	44	28

f)

图 2-2 双聚类的类型



a) 单一结构　b) 对角结构　c) 棋盘结构　d) 行互斥结构



e) 列互斥结构　f) 非互斥且非重叠结构　g) 嵌套重叠结构　h) 任意重叠结构

图 2-3 双聚类的结构

于该双聚类，如图2-3 a)所示。

(2) 对角结构。指任意两个双聚类之间互不共享行和列，且任一行或列只能属于其中一个双聚类，如图2-3 b)所示。这类的双聚类可以通过交换位置，最后呈对角线形状。

(3) 棋盘结构，指通过传统的聚类方法分别对行和列进行聚类，然后组合得到的双聚类，如图2-3 c)所示。

(4) 行互斥结构。指双聚类之间不存在共享的行，可以看作有对角线结构放松对行的限制所得，如图2-3 d)所示。

(5) 列互斥结构。与行互斥相似，指双聚类之间不存在共享的列，如图2-3 e)所示。

(6) 非互斥且非重叠结构。之允许双聚类之间存在相同的行或列，但不能存在重叠和包含关系，如图2-3 f)所示。

(7) 嵌套重叠结构。指双聚类之间可以存在包含关系，但不能出现重叠关系，如图2-3 g)所示。

(8) 任意重叠结构。指双聚类之间即可以存在包含关系，也可以存在重叠关系，如图2-3 h)所示。

传统的聚类只能找到像图2-3 a)所示的单一结构，这对于基因表达数据的挖掘是远远不够的。嵌套和重叠的结构要比非嵌套和非重叠的结构复杂，需要更复杂的算法来找到它们。

2.3 双聚类的评价指标

2.3.1 质量评价指标

由于在基因表达数据进行双聚类分析是NP难问题，目前大部分的算法都是基于优化策略的。为了评价双聚类的质量和知道优化的方向，需要有效的评价指标。指标是否科学可靠直接会体现到双聚类结果上。本节对目前常用的评价指标进行一个总结。

方便起见，先引入一些数学定义。给定一个大小为 $n \times m$ 的基因表达数据 $E(X,Y)$，以及一个大小为 $k \times l$ 的双聚类 $B(I,J)$，b_{Ij} 为 B 中第 j 列的平均值，b_{iJ} 为 B 中第 i 行的平均值，b_{IJ} 为双聚类整体的平均值，Vol_B 表示双聚类的体积，公式定义如下:

$$b_{iJ} = \sum_{j \in J} \frac{b_{ij}}{|J|} \tag{2-8}$$

$$b_{Ij} = \sum_{i \in I} \frac{b_{ij}}{|I|} \tag{2-9}$$

$$b_{IJ} = \sum_{i \in I, j \in I} \frac{b_{ij}}{|I| \times |J|} \tag{2-10}$$

$$Vol_B = |I| \times |J| \tag{2-11}$$

(1) 方差。用 $Var(B)$ 表示双聚类 $B(I,J)$ 的方差，该指标代表了该双聚类的变化幅度，越大则双聚类中的值越不相同，定义如下：

$$Var(B) = \sum_{i \in I, j \in I} \frac{(b_{ij} - b_{IJ})^2}{Vol_B} \tag{2-12}$$

(2) 均方残差。该指标首先在 CC 算法中提出，并被广泛地应用在基因表达矩阵的分析中。数学定义如公式2-13所示。该指标适合寻找像式2-4这样的加法模型双聚类，且值越小越符合该模型。

$$MSR(B) = \frac{1}{Vol_B}\sum_{i=1}^{k}\sum_{j=1}^{l}(b_{ij} - b_{iJ} - b_{Ij} + b_{IJ})^2 \tag{2-13}$$

(3) 扩展均方残差。因为 $MSR(B)$ 只能发现加法模型，Mukhopadhyay 等提出了扩展均方残差（Scaling Mean Squared Residue，SMSR）。该指标适合寻找如2-5这类的乘法模型双聚类，且值越小越符合该模型。双聚类 $B(I,J)$ 的 SMSR(B) 定义如下：

$$SMSR(B) = \frac{1}{Vol_B}\sum_{i=1}^{k}\sum_{j=1}^{l}(\frac{b_{iJ}\times b_{Ij} - b_{ij}\times b_{IJ}}{b_{iJ}\times b_{Ij}})^2 \tag{2-14}$$

(4) 相关指数。该指标用来寻找列值常量类型的双聚类，定义如下：

$$RI(B) = \sum_{j=1}^{l} R_{Ij}/l = \sum_{j=1}(1 - \frac{\sigma_{Ij}^2}{\sigma_j^2})/l \tag{2-15}$$

其中，R_{Ij} 为双聚类中第 j 列的相关指数，σ_{Ij}^2 是双聚类中第 j 列所有元素的局部方差,σ_j^2 是基因表达数据第 j 列所有元素的全局方差。该指标越大则越符合列值常量类型。类似的，稍加改造则可以寻找行值常量类型的双聚类。

(5) 最大标准化区域。该指标由 Giraldez 等提出，并用于寻找趋势一致的双聚类。计算过程：首先要对双聚类 $B(I,J)$ 进行标准化，得到 $\hat{B}(I,J)$，计算公式为：

$$\hat{b}_{ij} = \frac{b_{ij} - b_{iJ}}{\sigma_i} \tag{2-16}$$

其中，b_{iJ} 和 σ_i 为双聚类 B 中第 i 行元素的平均值和标准差。根据 $\hat{B}(I,J)$，双聚类 $B(I,J)$ 的最大标准化区域的定义如下：

$$MSA(B) = \sum_{j=1}^{l} | \frac{M_j - m_j - M_{j+1} + m_{j+1}}{2} | \tag{2-17}$$

其中，$M_j = max_{i\in[i,k]}\hat{b}_{ij}$，$m_j = min_{i\in[i,k]}\hat{b}_{ij}$。当双聚类中基因表达模型完全一致时，$MSA(B) = 0$。

(6) 覆盖率。该指标是双聚类集合的多样性指标，因为大多数算法找到的双聚类都不止一个，如果双聚类过度重合则意义不大。我们总是希望找到互相重叠小且能覆盖到更多的基因表达数据的双聚类集合。假设双聚类集合 $\Pi\{B_1, B_2, ..., B_r\}$，$\phi_k(E)$ 为判断 $E(X,Y)$ 中每个元素 e_{ij} 是否在双聚类 B_k 的函数。定义公式如下：

$$\phi_k(a_{ij}) = \begin{cases} 1 = & if a_{ij} \in B_k \\ 0 = & otherwise \end{cases} \tag{2-18}$$

则双聚类集合 Π 的覆盖率 $covRate(\Pi)$ 定义如下：

$$covRate(\Pi) = \frac{\sum_{i=1}^{k}\sum_{j=1}^{l}\cup_{k=1}^{r}\phi_k(a_i)}{n \times m} \tag{2-19}$$

从定义可以看出，覆盖率的含义是集合 Π 中 r 个子矩阵并集所占基因表达数据 E 的比例。

2.3.2 生物评价指标

为了评价通过双聚类获得基因集合的生物意义，需要对其进行基因解释。生物技术的发展积累了很多关于基因的描述，这些信息可以为我们提供参考。目前，最好的基因注释数据库当属 KEGG 数据库和 GO 数据库。前者主要用来做旁路分析，所谓旁路分析，就是获得基因之间的调控关系，后者主要用来做富集分析，对基因功能进行注释。目前对双聚类结果的生物验证主要还是 GO 的富集分析。

GO 数据库中保存了各种物种的基因的注释信息，包括基因的功能和之间的关系。GO 将基因的功能分为分子功能（Molecular Funciotn，MF），细胞组成（Cell Compose，CC）和生物过程（Biological Process，BP）。GO 将一项功能称为一个 GO 项（term），并通过一个有向无环图表示项与项之间的关系。如果两个 GO 项之间有连线，则表示之间存在联系。如果双聚类中关于某一 GO 项的基因个数大于该项随机概率出现的次数，则称该双聚类的基因集合富集在这一 GO 项，并用通过统计学方法，得到统计值 P-value 来表示富集的程度。P-value 越小，则富集程度越大，一般只关注 P-value 小于 0.01 的 GO 项。

(1) 显著富集双聚类的比例。对于双聚类集合 $\Pi\{B_1, B_2, ..., B_r\}$，假设其中存在 $r_{sig} \le r$ 个双聚类存在富集。显著富集双聚类的比例（Proportion of the biclustersSignificantly Enriched, proSigEnriched）定义如下：

$$proSigEnriched = \frac{r_{sig}}{r} \times 100\% \tag{2-20}$$

(2) 带权重的富集分数。$proSigEnriched$ 只是在双聚类层面的验证指标，不仅没有精确到功能项而且对于基因集合很大的双聚类很难区分。所以，带权重的富集分数（Weight Enrichment Score，WEScore）被引入进来，定义如下：

$$WEScore = \sum_{i=1}^{T} x_i s_i / k \tag{2-21}$$

其中，T 是双聚类 B 经过 GO 分析后得到的 GO 项的个数，x_i 是对应第 i 个

GO 项的基因个数，s_i 是对应 GO 项经过负对数变换后的 P 值。$WEScore(B)$ 越大则该双聚类生物意义越大。

(3) 平均 P 值。与 $WEScore$ 类似，平均 P 值（Mean of P Values, meanPValue）的定义如下：

$$meanPValue = \sum_{i=1}^{T} s_i / T \tag{2-22}$$

其中，T 与 s_i 与2-21含义一样，且 $meanPValue(B)$ 越大则该双聚类生物意义越大。

(4) 基因与 GO 项的比值。由于一个双聚类一般会在很多个 GO 项出现富集，如果 GO 项的个数越少，则说明双聚类中的基因之间越相关。因此，引入了基因与 GO 项的比值（Ratio of number of gene to number of significant terms, rateGeneTerm），定义如下：

$$rateGeneTerm(B) = k / T \tag{2-23}$$

其中，k 是双聚类中基因的个数，T 与2-21中含义一致。

2.4 双聚类算法的分类

2.4.1 基于质量评价的双聚类算法

因为在基因表达数据上进行双聚类分析是 NP 难问题，所以无法通过穷举的方式来搜索双聚类。上一节给出了一些质量评价指标，大部分算法都是使用不同的策略，找到在一种或多种评价指标最优的双聚类。根据搜索策略的不同，可以将算法分为以下几类。

(1) 基于贪婪迭代搜索策略。这类算法，一般是从一个初始的双聚类出发，然后根据质量指标迭代地添加或移除基因或条件节点。该类算法的有点是速度快，但通常质量欠佳。MSB（Maximum Similarity Biclusters）算法以及 CC 算法就其中的典型算法。

(2) 基于随机贪婪搜索。如前一种不同的是，此类算法在迭代过程中并不只考虑最优的操作，而是一定概率采取次优的操作，保证了搜索的多样性。FLOC 算法就属于此类。

(3) 基于聚类算法。该类算法的特点是，先使用传统的聚类方法分别对行和列进行聚类，并将其组合起来，然后在组合得到的双聚类中挑取质量评价较好的结果。例如，PSB 算法先使用 IPC 聚类算法在行和列两个方向聚类分析，然后使用 MSR(B) 筛选组合后得到的双聚类。

(4) 基于元启发式算法。元启发算法的特点是模拟大自然中生物的高效的搜索行为，例如例子群算法，蚁群算法。这类算法通过使用双聚类的质量评价指标组合成适应度函数，从而找到质量评价指标高的双聚类。

2.4.2 基于模型的双聚类算法

在有些双聚类算法中，并没有使用质量评价指标，该类算法被成为基于模型的双聚类算法。如基于图论的 SAMBA 算法，基于概率模型的 Plaid 算法，基于矩阵论的 ISA 算法，基于排序的 OPSM 算法，基于关联规则挖掘的 BiModule 算法。

2.5 群智能算法

目前, 有很多优秀的优化算法，有确定性方法如线性规划、二次规划，动态规划和梯度下降；以及随机性方法如群体智能。这些方法让我们能够在一定的时间内解决某些问题。然而，处理大量高维数据时，确定性方法太过复杂导致需要大量的计算成本。元启发式的群体智能算法因其高效率越来越受到关注。

2.5.1 布谷鸟搜索算法

布谷鸟搜索算法（Cuckoo Search，CS）是 Yang 和 Deb 于 2009 年提出的新兴启发算法。该算法通过模拟布谷鸟寄生育雏行为，在可行域中通过 Levy 飞行寻找合适的鸟巢，来找到较优解。该算法有三条理想化的规则：

1. 每只布谷鸟每次下一个蛋，并将其放入随机选择的巢中。

2. 具有优质蛋的最佳巢会被带到下一代。

3. 可用的寄主巢数量是固定的，且寄主以概率 $P_a \in (0,1)$ 发现布谷鸟放的蛋。在这种情况下，寄主可以消灭该蛋或放弃旧巢另建新巢。

CS 中有两种更新方式，一种是布谷鸟寻找宿主鸟巢的 Levy 飞行：

$$x_{i+1} = x_i + \alpha \otimes Levy(\beta) \tag{2-24}$$

其中，α 是步长缩放因子，$Levy(\beta)$ 是 Levy 飞行路径。

另一种是寄主以概率 P_a 发现外来鸟蛋后，采用随机方式重新建巢:

$$x_{i+1} = x_i + r \otimes Heaviside(P_a - \varepsilon) \otimes (x_t - x_k) \tag{2-25}$$

其中，r, ε 是服从均匀分布的随机数，$Heaviside()$ 是跳跃函数，x_t, x_k 是其他任意的两个鸟巢。

2.5.2 萤火虫算法

萤火虫算法 (Firefly Algorithm，FA) 是 Yang 于 2008 年提出的一种启发算法。把空间各点看成萤火虫，利用发光强的萤火虫会吸引发光弱的萤火虫的特点，在

发光弱的萤火虫向发光强的萤火虫移动的过程中，完成位置的迭代，从而找出最优位置。算法有以下三条假设：

1. 萤火虫不分性别, 这样一个萤火虫将会吸引到所有其他的萤火虫。

2. 吸引力与它们的亮度成正比，对于任何两个萤火虫，不那么明亮的萤火虫被吸引，因此移动到更亮的一个，然而，亮度又随着其距离的增加而减少。

3. 如果没有比一个给定的萤火虫更亮的萤火虫，它会随机移动。

萤火虫的相对荧光亮度计算方式:

$$I = I_0 e^{-\gamma r_{ij}} \tag{2-26}$$

其中，I_0 表示最亮萤火虫的亮度，即自身（$r = 0$ 处）荧光亮度，与目标函数值相关，目标函数值越优，自身亮度越高；γ 表示光吸收系数，因为荧光会随着距离的增加和传播媒介的吸收逐渐减弱，所以设置光强吸收系数以体现此特性，可设置为常数；r_{ij} 表示萤火虫 i 与 j 之间的距离。

当萤火虫 i 的相对亮度小于萤火虫 j 时，向萤火虫 j 靠拢。位置的更新方式为：

$$\beta(r) = \beta_0 e^{-\gamma r_{ij}^2} \tag{2-27}$$

$$x_i = x_i + \beta(x_j - x_i) + \alpha(rand - 1/2) \tag{2-28}$$

其中，β_0 表示最大吸引度，即光源处（$r = 0$ 处）的吸引度。α 为步长因子，$rand$ 为 [0，1] 上服从均匀分布的随机因子。

2.5.3 细菌觅食算法算法

细菌觅食算法算法（Bacterial Foraging Optimization，BFO）由 Passino 于 2002 年提出。通过模拟大肠杆菌菌落的觅食行为，不断地使用鞭毛游动和翻转，最终躲开有毒的地方并找到营养度高的位置，如图2-4所示。算法分为趋向性操作、复制操作和迁徙操作。

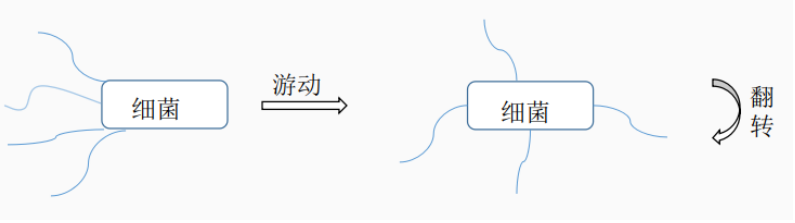


图 2-4 细菌的游动和翻转

1. 趋向性操作。这一操作模拟得是大肠杆菌的游动和翻转。在营养度高的地区，细菌会更多地游动，在营养度低的地区，细菌会更多地翻转，以逃出该地区。设细菌的种群规模为 S，维度为 n。细菌的觅食行为可以用以下公式表示：

$$\theta(i, j+1, k, l) = \theta(i, j, k, l) + C(i) \times \phi(i, j) \tag{2-29}$$

$$\phi(i, j) = \frac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}} \tag{2-30}$$

其中，$\theta(i, j, k, l)$ 表示细菌在第 j 次趋向性操作，第 k 次复制操作和第 l 次迁徙操作时的位置。$C(i)$ 是细菌 i 的趋向性步长。$\phi(i, j)$ 表示细菌在第 j 次趋向性操作时的随机方向的单位向量。$\Delta(i)$ 为随机向量。

2. 复制操作。复制操作的目的是将表现不好的细菌淘汰掉。首先，对种群按适应度排序，然后，前一半的细菌会复制一份覆盖后一半的细菌。保持种群数量不变的同时，实现优胜劣汰的机制。

3. 迁徙操作。在生物观察中发现，随着某些条件的改变，可能会使该地区的细菌突然死亡或迁移。算法通过迁徙操作模拟这一现象，提高种群的多样性。细菌会以一定的概率被清除，并随机生成一个新的细菌。

2.6 本章小结

生物信息学中，通过双聚类分析对基因表达数据进行挖掘，希望找到在对应条件下紧密相关的基因集合。本章主要对基因表达数据上的双聚类的相关知识进行了阐述，先是介绍了基因表达数据的重要以及特点；然后对双聚类的定义、类型和结构进行了描述；接下来对常用的质量评价指标以及生物评价指标进行了简要介绍，以及把双聚类算法分为了基于质量评价指标的和基于模型的两种；最后简要说明了本文用到的几种群智能算法。

第 3 章 基于 CS 和 FA 的混合双聚类算法

元启发式算法在双聚类领域的应用取得了很不错的效果，但元启发式算法本身的缺陷也会影响着双聚类的质量。一般来说，不同的算法有不同的使用范围，一个算法很难做到兼顾全局寻优与快速收敛。比如，布谷鸟算法具有较强的全局搜索能力，而在局部搜索却表现欠佳；萤火虫算法跟布谷鸟算法却刚好相反。全局寻优能力使得算法在寻在双聚类中能够找到更多样的结果，提高了覆盖率；局部寻优能力能够指导算法找到生物意义更加明确的双聚类结果。本文结合布谷鸟算法和萤火虫算法，提出一种混合的元启发式双聚类算法（CS-FA Biclustring，CSFAB），并将 CSFAB 算法在四个基因表达数据与其他常用的双聚类算法进行了质量验证指标和生物验证指标的比较。

3.1 混合双聚类算法分析

3.1.1 编码设计

给定基因表达矩阵 $E(X,Y)$，比特串 $x_p = (g_1,\ldots,g_i,\ldots,g_m,s_1,\ldots,s_j,\ldots,s_n)$，$p = 1,\ldots,N$, 被用来表示一个双聚类或子矩阵 $E(I,J)$。其中，N,m,n 分别是种群数量,E 的基因数目和样本数目。当 E 中第 i 个基因或第 j 个样本被选为 $E(I,J)$ 时，$g_i = 1$ 或 $s_j = 1$，否则，$g_i = 0$ 或 $s_j = 0$，$1 \le i \le m$ 且 $1 \le j \le n$。

原始的群智能算法的解（粒子, 鸟巢, 萤火虫）都是多维的连续值, 需要映射成对应的比特串后才能用来表示双聚类。通常的做法就是设置上限为 1, 下限为 0, 然后判断是否大于 0.5, 将实数值映射成比特值。如下图所示：

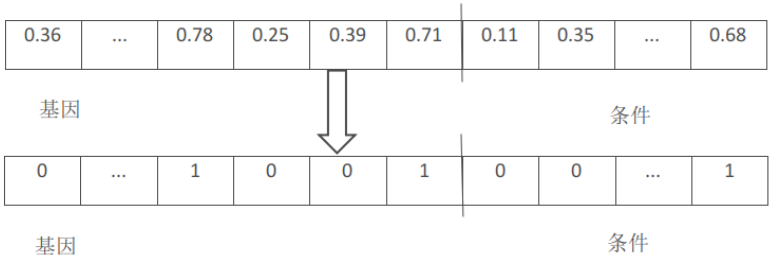


图 3-1 将连续的解映射为双聚类

3.1.2 适应值函数设计

优化算法需要知道优劣的评价标准，在群智能算法中一般称之为适应值。我们需要设计一个适应值函数，用来得到一个解的质量，从而在解与解之间以及算法之间进行比较。正如2.3.1小节提到的，MSR 是最主要也是最直观的质量评价指

标，同时双聚类的体积也是衡量好坏的标准之一。一般来说，体积越大的双聚类MSR 会相应的变大，而我们希望找到体积大但是 MSR 较小的双聚类。所以，需要在保持两者之间平衡的同时，能够引导双聚类算法找到更优的解。对于双聚类 $B(I,J)$，其适应值为：

$$f(B) = MSR(B) + \frac{\lambda}{GV(B)} + \frac{\mu}{CV(B)} + \frac{\omega}{Var(B)} \tag{3-1}$$

$$GV(B) = |I| \tag{3-2}$$

$$CV(B) = |J| \tag{3-3}$$

其中，$GV(B)$，$CV(B)$ 分别是 $B(I,J)$ 中基因和实验条件的容量。λ, μ 是针对量纲不同问题，λ, μ 越大则 GV 和 CV 对适应值的影响越大，其值视数据集的情况而定。

3.1.3 混合方案设计

大致有两种策略将两个算法混合，顺序执行策略和嵌套策略。第一种策略是将一个算法的结果作为另一个算法的输入，特点是两个算法前后互不影响。Nepomuceno 等将 SEBI 的结果输入到 SSB 算法中，进一步提高双聚类的质量。第二种策略是将两个算法的揉合到一起，将某一个算法作为局部功能嵌入到另一个算法中，这时两种算法前后不是独立的。例如，Bryan 等将 CC 算法的局部搜索功能作为 SAB 算法的一步，以提高双聚类的容量。基于上述二种策略, 可有如下三种方案:

(1) 顺序执行 CS-FA：这种方案可以看作将 FAB 算法的随机初始化替换成 CSB 算法，先使用 CSB 算法生成双聚类，然后使用 FAB 算法进一步提高双聚类的质量，如算法3-1所示。

算法 3-1 CS-FA 混合方案

Input: $n \times m$ 的基因表达矩阵 E，弃巢比例 p，种群大小 N，光吸收系数 γ，最大吸引度 β_0，步长因子 α，最大迭代次数 Iter

Output: 一个满足条件的双聚类 B

```
P = Initialization(E, N)    //初始化种群
P_CS = CSB(P,E,p,Iter)
P_CS-FA = FAB(P,E,γ,β0,α,Iter)
B = Best(P_CS-FA)
return B
```

(2) 顺序执行 FA-CS：与第一种方案刚好相反，将 CSB 算法的随机初始化改为

FAB 算法，如算法3-2所示。

算法 3-2 FA-CS 混合方案

Input: $n \times m$ 的基因表达矩阵 E，弃巢比例 p，种群大小 N，光吸收系数 γ，最大吸引度 β_0，步长因子 α，最大迭代次数 Iter

Output: 一个满足条件的双聚类 B

```
P = Initialization(E, N)    //初始化种群
P_FA = FAB(P,E,γ,β0,α,Iter)
P_FA−CS = CSB(P,E,p,Iter)
B = Best(P_FA−CS)
return B
```

(3) 嵌套执行 CS-FA

算法 3-3 CSFA 混合方案

Input: $n \times m$ 的基因表达矩阵 E，弃巢比例 p，种群大小 N，光吸收系数 γ，最大吸引度 β_0，步长因子 α，最大迭代次数 Iter，最大早熟次数 maxEarlyStopCnt

Output: 一个满足条件的双聚类 B

```
i = 1
earlyStopCnt = 0
B_old = INF
P_fa = Initialization(E, N)    //初始化种群
do
    P_cs, best_cs = csIter(P_fa, E, p)
    P_fa, best_fa = faIter(P_cs, E, γ,β0,α)
    B = Best( best_cs, best_fa)
    earlyStopCnt = EarlyStop(earlyStopCnt, B, B_old) B_old = B i++
while (i< = Iter AND earlyStopCnt< maxEarlyStopCnt);
return B
```

第 4 章 基于多目标 BFO 优化的双聚类算法

4.1 多目标优化问题的基本概念

4.2 基于多目标 BFO 搜索双聚类算法

4.3 实验结果及分析

4.4 本章小结

第 5 章 总结与展望