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 3 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 3 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: March, 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。通常，研究人员通过聚类来挖掘相关的基因集合，然后进行生物学上的整理和分析。然而，由于基因表达数据独特的数据结构和背后的生物意义，倾向于找到全局模式的传统聚类方法并不能很好的找出符合要求的具有局部模式的聚类。于是，更符合基因表达数据特点的双聚类分析被引入进来。

基因表达数据的双聚类分析是指, 找出在某些条件子集下包含一致表达波动的基因子集。双聚类分析已经被证明为 NP 难问题，因此大部分算法都是通过优化策略来尽可能得找到较优解。同时，因为双聚类的指标之间存在一定程度的负相关，所以双聚类分析可以看作是一种多目标优化问题。在优化算法之中，元启发算法中的群智能算法因其高效和简洁，在学术界和工业界都得到了很大的关注和应用。近年来，针对表达数据高维度, 高冗余的特点, 许多群智能优化算法以及多目标优化算法被用于双聚类分析。

当前，对于将群智能算法运用到双聚类分析的研究仍存在或多或少的问题。一方面是群智能算法本身的缺陷所致，如每次只能得到一个最优解，有可能陷入局部最优等；另一方面是没有能将群智能的特点与双聚类分析有机的结合起来，如选取合适的评价指标进行单目标或多目标的寻优。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，旨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）。考虑到布谷鸟算法和萤火虫算法可以看作互补的关系，前者具有较强的全局寻优能力，而后者具有较快的收敛速度，于是本文尝试将两者结合。首先，通过实验确定了有效的结合策略，然后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 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）。因为双聚类分析可以看作多目标优化问题，本文将传统的单目标细菌觅食算法依据基因表达数据双聚类分析的特点进行了改进，主要包括：（1）对于互不支配时，较优解的确定；（2）根据种群中各自的被支配次数排序；（3）引入外部占优解集增加多样性。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。通过对双聚类的质量评价指标和生物富集分析，证明了 MOBFOB 算法能够有效且快速地找到具有显著生物意义的双聚类。

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. Generally, researchers use clustering to mine related sets of genes, and then organize and analyze them biologically.However, due to the unique data structure of the gene expression data and the biological significance behind it, traditional clustering methods that tend to find global patterns are not good at finding clusters with local patterns that meet the requirements.Therefore, biclustering analysis that more suitable for the characteristics of gene expression data was introduced.

Biclustering of gene expression data refers to finding a subset of genes that contain consistent expression fluctuations under certain conditional subsets.Biclustering analysis has proven to be an NP-hard problem, so most algorithms try to find the best solution by optimizing the strategy. At the same time, because there is a certain degree of negative correlation between the indicators of biclustering, biclustering analysis can be regarded as a multi-objective optimization problem.Among the optimization algorithms, the swarm intelligence algorithm in the meta-heuristic algorithm has received great attention and application in academia and industry because of its efficiency and simplicity.In recent years, in view of the high dimensionality and high redundancy of expression data, many swarm intelligence algorithms have been used in biclustering.

At present, the research on the application of swarm intelligence algorithms to biclustering analysis still has more or less problems.On the one hand, it is caused by the shortcomings of the swarm intelligence algorithm. For example, only one optimal solution can be obtained at a time, and it may fall into a local optimal.On the other hand, it is caused by not organically combining the characteristics of swarm intelligence with biclustering analysis, such as selecting appropriate evaluation indicators for single or multi-objective optimization. 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 paper

includes:

1. A hybrid biclustering algorithm CSFAB(Cuckoo Search and Firfly Algorithm hybrid Biclustering) based on cuckoo search algorithm and firefly algorithm is proposed. Considering that the cuckoo algorithm and the firefly algorithm can be regarded as a complementary relationship, the former has a strong global optimization ability, while the latter has a faster convergence speed, so this article attempts to mix the two.First, an effective hybrid strategy was determined through experiments, and then the global search ability of the cuckoo search algorithm and the fast convergence ability of the firefly algorithm were effectively combined. The CSFAB algorithm can significantly improve the search speed and range, and at the same time can jump out of the local optimal solution and find biclusters containing different genes, thereby improving the diversity of biclusters.Compared with 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 analysis can be considered as a multi-objective optimization problem, this paper improves the traditional single-object bacterial foraging algorithm based on the characteristics of bi-clustering analysis of gene expression data,mainly include:(1) Determination of better solution when there is no dominant solution;(2) Sort according to the number of times they are dominated in the population; (3)Introducing externally dominant solution sets to increase diversity.This algorithm uses a multi-object bacterial foraging algorithm to simultaneously optimize the bicluster's quality evaluation indicators such as mean square residual and volume to find the dominant biclustering solution set.The quality evaluation index and bio-enrichment analysis of the biclustering prove that the MOBFOB algorithm can effectively and quickly find the biclusters with significant 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

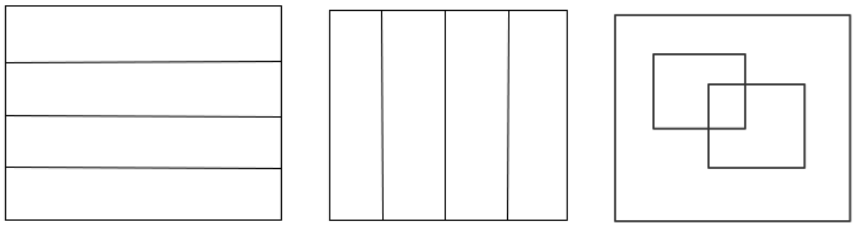
随着科技的进步，人类对自然，以及对生命有了更为深刻的认识。从显微镜的发明到 DNA 双链结构的提出，再到如今的 21 世纪，伴随着科学研究的深入，人们越来越多地发现，许多重大疾病跟基因相关，如某些癌症、一些先天性的心脏病和肥胖症。Maron 等研究发现，大多数肥厚型心肌病、扩张型心肌病的病人中可发现致病基因。对于基因的研究成为了解决或预防这些疾病的关键。在人类的历史上不时出现的大规模传染病，带走了上百万人的生命，对人类的经济甚至文明都产生了巨大的影响。无论是 2003 年的非典（SARS），还是 2019 年底的新型冠状病毒（COVID-19），基因都是研究这些病毒的突破点，解析了其基因的组成和作用，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，已经有公司推出了价格亲民的基因测序产品，人们只需花费数百美元然后在一周之内就可以知道自己的基因序列和表达情况，从而可以提前知道自己有哪些易感基因以及将来容易得哪些疾病，并以此为根据做好预防。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体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。mRNA 的含量越高，则其代表的基因表达水平越高。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，因此被称为基因表达数据。通常情况下，基因表达数据的每一行代表一个基因，每一列代表一个条件或样本。

在生物信息学中，对基因表达数据的挖掘是研究热点。基因表达数据中含有大量有用的信息。例如，在何种条件下，哪些基因是表达相似的或者存在差异？这些基因都共同参与到了哪些功能或者通路？这些基因受到了哪些调节？如何将隐藏在基因表达数据中的价值挖掘出来并利用，需要大量的计算和研究。数据挖掘领

域已经大量成熟的理论和方法供人借鉴，如有监督学习的分类和无监督学习中的聚类。分类技术使得人们更方便地对新产生的数据进行分类，并在医学检测中广泛使用。通过对基因表达数据的聚类分析，可以得到研究人员感兴趣的差异基因。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。比如，Eisen 等为了推断基因的新功能，对人类的 8600 个基因进行了聚类，然后在聚类的结果上利用基因表达谱的相关性进行推断。Tavazoie 等使用 k 均值聚类算法发现了酵母转录调控网络。Tamayo 等利用相似的技术，通过对基因聚类，推断出了新基因的功能和调控网络。

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，正如前面所诉，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。这些特点使得传统的聚类方法无法胜任，于是就引入了双聚类（Bicluster）的概念，如图1-1所示。不同于传统聚类的全局模式（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。



a) 基因方向的传统聚类　b) 条件方向的传统聚类　c) 双聚类

图 1-1 传统聚类与双聚类的区别

对于传统的聚类方法，其可以从基因或条件方向分别聚类，但是任一条件或基因都将被分配到某一类中去，这与现实中的情况是不符合的。而双聚类允许某些条件或基因不在任一类中出现。

1.2 相关研究进展

从 2000 年双聚类分析被 Cheng 和 Church 引入到基因表达数据挖掘中到现在，双聚类分析经过了二十年的发展，许多优秀的算法被提出。由于双聚类分析是一个非常困难的问题，这一方向的研究一直在不断地进行着。殷路的研究将双聚类

的发展过程大致分为了三个阶段。

1.2.1 早期阶段

该阶段的工作主要集中在模型和质量评价上。双聚类（Bicluster）这一单词最早由 Hartigan 于 1972 年提出，但 Hartigan 只是在行和列两个方向上分别聚类，没有全面地阐述双聚类的概念。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，Bergmann 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。Lazzeroni 等把基因表达数据看作为背景模型与多个双聚类的叠加，并以此为基础提出了 Plaid 模型。Ben-dor 等将双聚类建模为 OPSM（Order Preserving Sub-Matrices），一个在基因表达水平上由连续保持的基因和条件组成的子矩阵，然后使用贪婪启发式算法在基因表达数据中搜索双聚类。Murai 等将双聚类建模为基因表达的保守模式 xMotifs（Conserved Gene Expression Motifs），然后将随机选择的样本作为种子进行扩展，得到满足条件的双聚类，重复该过程，直到所有样本都包含在多个双聚类中。

Segal 等利用贝叶斯公式，在基因表达数据上结合先验信息建立了概率模型并称之为 RPM （Rich Probabilistic Models）模型。该模型通过期望最大化 EM (Expectation Maximization) 算法来确定双聚类。紧接着，Segal 等又推广了 RPM 模型，使其能够允许双聚类出现重叠。Wang 等提出了 pCluster 指标来描述基因之

间的相似距离，并将找到的双聚类为 p-Cluster，然后通过枚举和减枝技术搜索基因表达数据中的双聚类。

1.2.2 壮大阶段

该阶段相比早期阶段增加了在元启发式算法和多目标优化算法的研究。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，Bleuler 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，Divina 等提出多目标连续变化双聚类算法（Sequential Multi objective Biclustering，SMOB）。Giraldez 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，有效解决了成比例模式的双聚类问题。2013 年 Brizula 等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。

1.2.3 当前阶段

随着大数据，集成学习，深度学习等技术的发展和普及，许多新型的双聚类分析算法涌现出来。当前阶段除了双聚类模型和质量评价指标，元启发式算法和多目标优化算法之外，增加了以这些新型技术为基础的双聚类研究。基于 Bayesian 理论和双聚类的 Plaid 模型，Zhang 等提出了 Bayesian Plaid 模型。Liu 等提出了基于 GPU 的统一计算设备体系结构 (CUDA) 的并行 GBC 算法。另外，集成学习因其在聚类方向的广泛应用，也被引入到双聚类分析中。Hanczar 等应用 bagging 技术提高 CC 和 Plaid 算法的结果质量。基于谱技术，Huang 等提出了 SCCE（Spectral Co-Clustering Ensemble）双聚类算法。近年来的研究使得神经网络这种深度学习技术取得了突飞猛进的发展，其强大的能力不断地让人们感到惊叹。Sun 等基于深度学习技术提出了自动解码机（AutoDecoder）算法用来寻找双聚类。

1.3 论文主要工作

因为基因表达数据上双聚类分析的困难性和重要性，过去提出的大量算法都有各自的侧重点。有关基于元启发式的双聚类分析算法，缺少一个较为完整的完

整的分析工作。本文以提高双聚类结果的质量指标和生物意义为目标，结合混合元启发式方法和多目标优化方法，对双聚类分析这一问题进行了综合比较研究。本文主要工作如下：

1 考虑到布谷鸟搜索算法和萤火虫算法的优势互补，本文提出了一种基于布谷鸟搜索和萤火虫算法的混合双聚类分析算法（Cuckoo Search and Firfly Algorithm hybrid Biclustering，CSFAB）。首先，将均方残差，基因容量和样本容量的权重之和作为适应值函数。然后将连续的解转换成比特串来表示双聚类，经过几种混合方式的比较之后，选择了最佳的混合方案。最后在四个常用且数据规模大小不一的基因表达数据集上，通过实验对算法的有效性进行了分析和验证。

2 双聚类分析其实是多目标优化问题。本文提出了一种多目标细菌觅食算法的双聚类分析算法（Multi-object Bacterial Foraging Algorithm Biclustering，MOBFOB）。算法将均方残差，基因容量和样本容量看作待优化指标，并按照被支配次数对种群排序。以细菌觅食算法为指导，不断地寻找占优的双聚类，并将其保存在外部集合中。最后在四个基因表达数据集上，通过实验对算法的有效性进行了验证。

1.4 论文组织结构

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。全文共有五章，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。

第三章为了解决普通双聚类算法的质量评价指标不足和生物意义不明显的缺点，本文将布谷鸟算法与萤火虫算法进行了融合，并提出了 CSFAB 双聚类算法。通过实验，确定了融合的最佳方案，并从质量评价指标和生物评价指标的角度进行了比较分析。

第四章首先介绍了多目标优化的基本概念，接着将细菌觅食算法按照多目标优化和双聚类的特点进行了改进，包括适应值函数，以及互不支配时的比较规则，并提出了 MOBFOB 算法。最后根据适应值曲线和生物指标对算法进行了分析。

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上限，而算法只是去逼近这个上限，因此了解数据本身是很有必要的。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。大致的流程为，制备芯片、制备样本、标记和杂交、洗涤和绘制图像，获取结构化数据。首先，准备一个具有成千上万个网格的微阵列芯片，该芯片的每个网格里面都放置着一个 DNA 探针。接着准备两条 mRNA，一条用于测试，另一条作为控制样本。对这两条 mRNA 进行逆转录，得到互补脱氧核糖核酸（cDNA），并使用荧光染料或放射性同位素对 cDNA 进行标记。将标记后的 cDNA 与微阵列芯片中的 DNA 进行杂交，然后清洗掉多余的杂交液。最后，通过芯片扫描仪获得芯片在杂交后各个 DNA 探针的荧光染料或放射性同位素强度，并绘制成信号强度图。通过专业的软件将图像转化为结构化的数字信息，得到基因表达数据。

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和噪声。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。一般来说，有两种方法处理缺失值：一是直接丢弃存在缺失值的行和列；二是根据已知的数据对缺失值进行填充，该种方法有可能会对原始数据产生不利的影响，改变真实数据的分布，需要根据数据特点选择不同的填充策略。

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）。

基因表达数据极其庞大，以及需要非常专业的生物学知识，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，如序列、平台和数据集，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。

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。

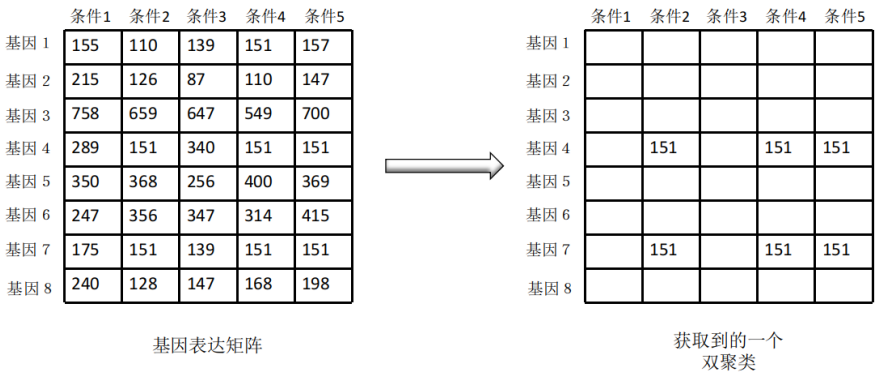


图 2-1 双聚类定义示例

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

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。

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

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

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

2.2.3 双聚类的结构

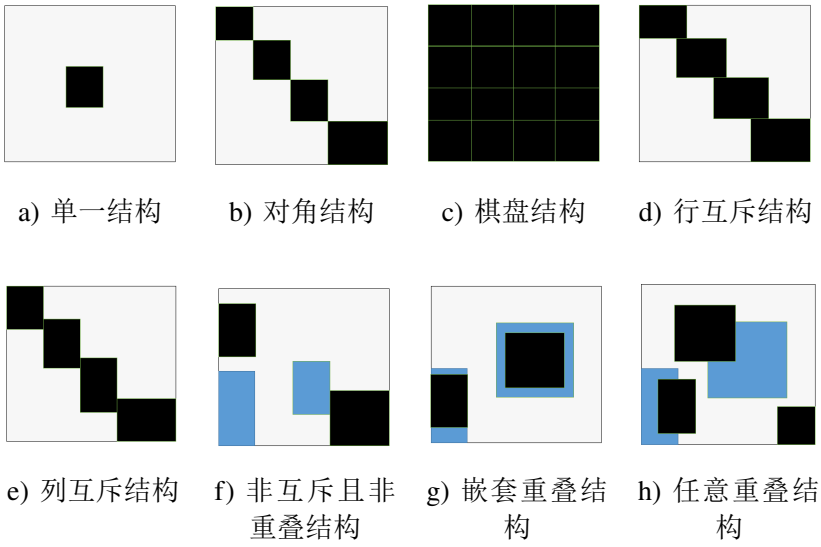
双聚类的结构是指，通过算法找到的双聚类之间在原始矩阵时间的相对位置。根据结构，大体可以分为 8 类:

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。



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

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：

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

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. HV-Score。一个双聚类 $B(I,J)$ 的 Hv-Score 定义如下：

$$Hv(B) = \frac{\sum_{i=1}^{k}\sum_{j=1}^{l}(b_{ij} - b_{iJ} - b_{Ij} + b_{IJ})^2}{\sum_{i=1}^{k}\sum_{j=1}^{l}(b_{ij} - b_{iJ})^2} \tag{2-18}$$

该指标是对 MSR 的改进，改善了 MSR 偏向找到常量类型双聚类的不足，由 Bryan 等提出。值越小则双聚类的质量越好。

7. 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：

$$\phi_k(a_{ij}) = \begin{cases} 1 = & if a_{ij} \in B_k \\ 0 = & otherwise \end{cases} \tag{2-19}$$

则双聚类集合 Π 的覆盖率 $covRate(\Pi)$ 定义如下：

$$covRate(\Pi) = \frac{\sum_{i=1}^{k}\sum_{j=1}^{l}\cup_{k=1}^{r}\phi_k(a_i)}{n \times m} \tag{2-20}$$

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，最流行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 $r_{sig} \leq r$ 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：

$$proSigEnriched = \frac{r_{sig}}{r} \times 100\% \tag{2-21}$$

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：

$$WEScore = \sum_{i=1}^{t} x_i s_i / k \tag{2-22}$$

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，k 是双聚类中基因的个数。$WEScore(B)$ 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：

$$meanPValue = \sum_{i=1}^{t} s_i / t \tag{2-23}$$

其中，t 与 s_i 与公式2-22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：

$$rateGeneTerm(B) = k / t \tag{2-24}$$

其中，k 和 t 与公式2-22中含义一致。

2.4 双聚类算法的分类

为了解决双聚类这一难题，大量的算法被提出。有的算法使用质量评价指标来指引着双聚类搜索过程，有的则使用其他策略解决问题。基于此，本文将双聚类分析算法大致分为基于质量评价的双聚类算法和基于模型的双聚类算法。

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。跟前一种不同的是，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

在有些双聚类分析算法中，并没有使用质量评价指标，该类算法被称为基于模型的双聚类算法。根据算法使用的数学模型或结构的不同，将这些算法分为以下几类：

1. 基于图论的双聚类分析方法。计算机科学中的图可以用二维矩阵表示，将图论中的知识迁移到基因表达数据的双聚类分析中，是该类算法的显著特点。例如，SAMBA 和 Bi-Force 算法将基因表达数据转换为带权重的二分图，双聚类则被视为其中的二分团（biclique）。MicroCluster 算法将基因表达数据转换为带权有向多重图，双聚类分子转化为深度搜索树。

2. 基于概率模型的双聚类分析方法。概率论和数理统计在挖掘海量数据方面有着强大的理论支持。这类方法通过在基因表达数据上建立统计模型，通过优化模型参数来找到优质的双聚类。这类算法有，基于 Gibbs 采样理论的 QDB 算法以及对 Plaid 模型改进的 PPM 算法。

3. 基于矩阵论的双聚类分析方法。基于矩阵论和线性代数，通过线性变换和矩阵分解等理论来寻找双聚类是这类算法的主要手段。比如，ISA 算法中就间接用到了奇异值分解（SVD）。nsNMF 算法使用非负矩阵分解（NMF）来寻找双聚类。

4. 基于关联规则挖掘的双聚类分析方法。将在业务数据上发现最大频繁项的问题，与在基因表达数据中双聚类分析问题联系起来，是该类算法的主要特点。基于关联规则的挖掘算法在商业上有着广泛地应用，通过类比的思想，这类的双聚类分析算法取得了不错的效果。具有代表性的算法有 BiModule 算法和 Fdcluster 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。这些方法帮助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 粒子群算法

粒子群（Particle Swarm Optimization，PSO）算法是 Kennedy 和 Eberhart 于 1995 年提出的一种群体智能优化算法，其流程图如图2-4所示。该算法是受到了鸟群觅食过程中的集体行为的启发。PSO 算法将群体中的粒子看作可行域中的一个点，这些点既没有质量也没有体积，只有一定的初始速度，在可行域中飞行。粒子可以通过飞行过程中自身的最优值和群体的最优值不断地修正自己的前进方向和速度大小，从而形成群体寻优的正反馈机制。其粒子更新方式为：

$$v_i = wv_i + c_1r_1(pbest_i - x_i) + c_2r_2(gbest - x_i) \tag{2-25}$$

$$x_i = x_i + v_i \tag{2-26}$$

其中，v_i 是粒子 i 的速度矢量，x_i 是粒子 i 的位置矢量，$pbest$ 是粒子 i 的历史中最优的位置。$gbest$ 是所有粒子的历史中最优的位置。c_1, c_2 是加速度常数，调节学习最大步长。r_1, r_2 是两个随机函数，取值范围［0，1］，以增加搜索随机性。w 是惯性权重，非负数，调节对解空间的搜索范围。

粒子速度更新公式包含三部分：第一部分为“惯性部分”，即对粒子先前速度

的记忆；第二部分为“自我认知”部分，可理解为粒子 i 当前位置与自己最好位置之间的距离；第三部分为“社会经验”部分，表示粒子间的信息共享与合作，可理解为粒子 i 当前位置与群体最好位置之间的距离。

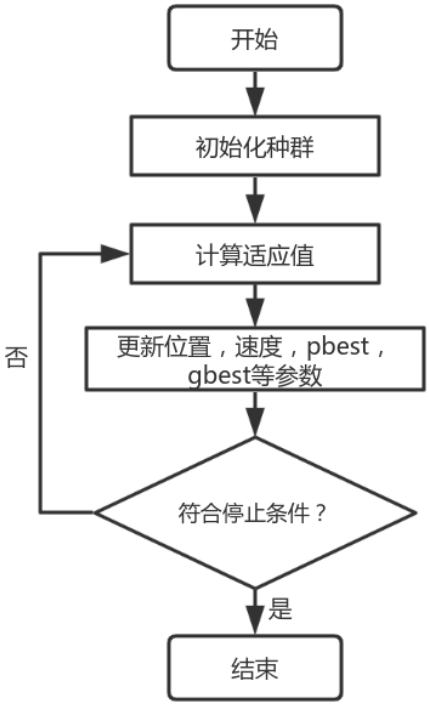


图 2-4 粒子群算法流程图

2.5.2 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，其流程图如图2-5所示。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：

$$x_{i+1} = x_i + \alpha \otimes Levy(\beta) \tag{2-27}$$

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:

$$x_{i+1} = x_i + r \otimes Heaviside(P_a - \varepsilon) \otimes (x_t - x_k) \tag{2-28}$$

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。

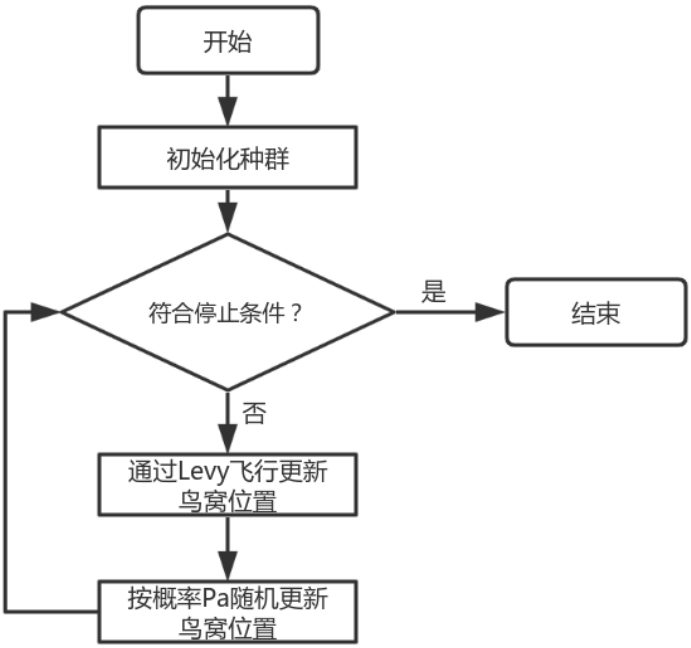


图 2-5 布谷鸟算法流程图

2.5.3 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，其流程图如图2-6所示。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:

$$I = I_0 e^{-\gamma r_{ij}} \tag{2-29}$$

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：

$$\beta(r) = \beta_0 e^{-\gamma r_{ij}^2} \tag{2-30}$$

$$x_i = x_i + \beta(x_j - x_i) + \alpha(rand - 1/2) \tag{2-31}$$

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。

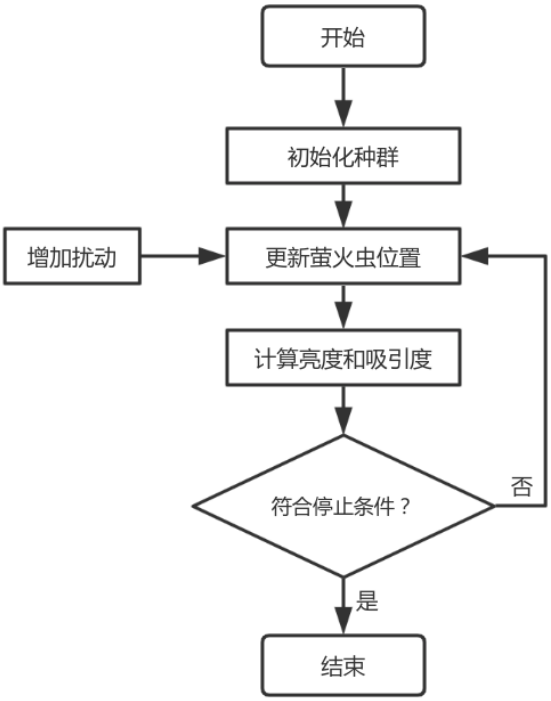


图 2-6 萤火虫算法流程图

2.5.4 细菌觅食算法

细菌觅食算法（Bacterial Foraging Optimization，BFO）由 Passino 于 2002 年提出，其流程图如图2-7所示。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2-8所示。算法分为趋向性操作（趋化操作）、复制操作和迁徙操作。

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：

$$\theta(i, j+1, k, l) = \theta(i, j, k, l) + C(i) \times \phi(i, j) \tag{2-32}$$

$$\phi(i, j) = \frac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}} \tag{2-33}$$

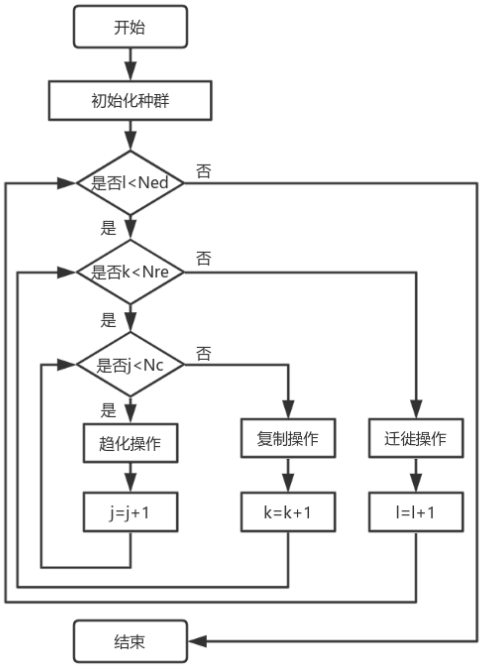


图 2-7 细菌觅食算法流程图

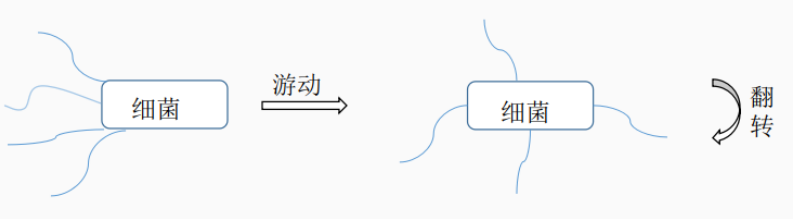


图 2-8 细菌的游动和翻转

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性以及特点；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（Cuckoo Search and Firfly Algorithm hybrid Biclustering，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：

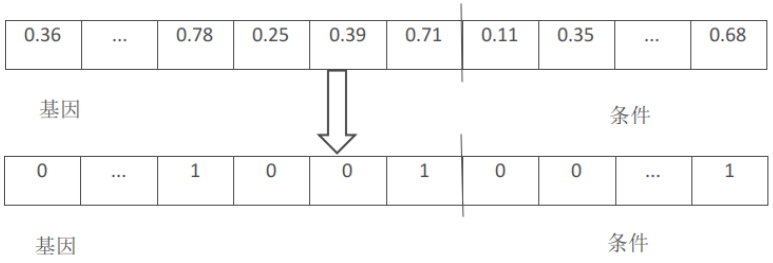


图 3-1 将连续的解映射为双聚类

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：

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

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

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

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。例如，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：该方案在每一次迭代都会执行 CS 操作和 FA 操作，并且采用竞标策略保留两次操作中最优的个体。

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
```

3.1.4 停止条件

算法的停止条件是达到最大的迭代次数或者种群中最优双聚类的质量已经有

一定的时间不再提升，后一种情况称为 early stopping。通过 EarlyStop 函数来实现，当新的最优适应值相比较上一代的最优适应值几乎没有变化时，将 earlyStopCnt 加一，否则置零。

算法 3-4 EarlyStop 函数

Input: earlyStopCnt, 新的最优适应值 f_{new}，旧的最优适应值 f_{old}

Output: 新的 earlyStopCnt

```
if |f_new - f_old| < f_new/1000 then
    earlyStopCnt++;
else
    earlyStopCnt = 0
end
return earlyStopCnt
```

3.2 实验环境及所用数据集

本节主要介绍实验所用到的软硬件环境和数据，后面章节的实验所用的环境和数据与本节相同，将不再赘述。

3.2.1 实验环境

本文提出的 CSFAB 双聚类算法与其他用来比较的算法均是用 Matlab 语言实现，并运行在 Matlab R2018b 环境中。对于双聚类结果的生物验证，是用 R 语言的 clusterProfiler 包得到的。所有代码都运行在 64 位的 Ubuntu 18.04 操作系统上，CPU 是 intel i7-9700K，内存大小为 16G。

3.2.2 实验所用数据集

不同的双聚类算法采取不同的搜索策略，有不同的侧重。为了全面的评估本文所提出的算法，本文选择了四个数据量各不同的数据集，如表3-1所示。这些基因表达数据集都是来自 GEO 数据库，编号分别为 GDS2350，GSE2403，GSE952，GSE2034。λ 和 μ 均为经验值。本文使用 Python 的 GEOParse 包和 Pandas 包在基

表 3-1 本文所用的基因表达数据集的相关信息

数据名缩写	数据全名	基因数量	条件数量	λ	μ
Yeast Cell	Yeast cell cycle	5847	50	2.0E05	2.0E03
BCLL	B-cell chronic lymphocytic leukemia	12815	21	1.0E04	1.0E03
RatStrain	Rat multiple tissue in strain	7751	122	6.0E05	1.0E04
PBC	Primary breast cancer	21225	286	3.0E06	4.0E04

因维度上对数据进行了 Min-max Normalization，缩放到 [0，1] 区间并乘以 100。计算公式如下。

$$x' = \frac{x - min(x)}{max(x) - min(x)} \times 100 \tag{3-4}$$

其中，x 为基因表达数据的某一行，也就是该基因在所有条件下的表达水平。

3.3 实验结果及分析

本节主要内容为三种混合方案的性能分析比较，然后在四个基因表达数据上对各双聚类算法进行测试，最后在质量验证指标和生物验证指标对各算法进行讨论分析。

3.3.1 混合方案比较

为了确定3.1.3节中提到的三种方案中，哪种更适合双聚类分析，本文在 BCLL 数据集上，使用3.1.2节中定义的适应值函数，分别对三种方案进行了 100 次实验，得到相应的双聚类。因为在算法中使用到了三个指标，为了减少偏向性，同时对相关指数 RI 和 HV-Score 共五个指标计算其平均值和标准差，如表3-2所示。由表3-2可知，前两种混合方案能够使双聚类的体积增加一些，在 HV-Score 和样本个数指标上，三种策略的差别并不大，但是嵌套执行方案在 MSR 和 RI 上均有明显的优势。所以，本文抛弃了顺序执行方案，并将嵌套执行方案命名为 CSFAB。

表 3-2 三种混合方案在 BCLL 数据集上的质量评价指标

	基因个数	样本个数	MSR	RI	HV-Score
CS-FA	**6093.88 ± 39.12**	**8.03 ± 0.99**	296.212 ± 7.26	0.019 ± 0.003	0.994 ± 0.001
FA-CS	6044.54 ± 51.53	7.54 ± 0.65	307.616 ± 14.17	0.006 ± 0.005	0.995 ± 0.001
CSFA	5992.93 ± 61.52	8.0 ± 0.0	**277.049 ± 2.97**	**0.031 ± 0.003**	**0.994 ± 0.0007**

3.3.2 CSFAB 的质量验证指标比较分析

为了较公平和全面地衡量本文提出的 CSFAB 算法的有效性，本文选择了采用相同的编码方案和适应值函数的萤火虫算法，布谷鸟搜索算法和粒子群算法作为对比算法，并分别命名为 FAB，CSB，PSOB。表3-3和表3-4分别为各算法在各数据集上的基因容量和样本容量的平均值和标准差。同时，为了评价算法的多样性，图3-2展示了各算法在不同基因表达数据集上的覆盖率。图3-3和图3-4给出了各算法在在四个数据集上 MSR 以及 RI 的箱线图。

从表3-3可以看出，PSOB 在 BCLL 数据集上表现最优，而 CSFAB 在其他三个数据集上均取得了最大的基因容量。而表3-4表明，在样本容量方面，CSFAB 表现平平，仅在量级最大的 PBC 中比另外三个算法更优，其他三个数据集均为 FAB 算

法最优。但在 BCLL 中，各算法的表现相差不大。实验数据说明，在数量级比较大时，CSFAB 算法更占优势。图3-2可知，无论是在什么量级的数据中，相比其他算法，CSFAB 算法总能找到更多样化的双聚类。

表 3-3 CSFAB 等四个算法的基因容量平均值与标准差

	Yeast Cell	BCLL	RatStrain	PBC
CSB	2974.42 ± 41.57	6075.06 ± 50.18	3929.93 ± 47.22	10781.59 ± 80.67
FAB	3034.14 ± 40.78	6053.57 ± 58.17	4126.25 ± 39.74	11268.66 ± 61.56
CSFAB	**3093.02 ± 37.56**	5992.93 ± 61.52	**4193.04 ± 48.38**	**11539.04 ± 82.86**
PSOB	3030.74 ± 41.14	**6093.41 ± 49.89**	4094.83 ± 43.77	11052.69 ± 78.39

表 3-4 CSFAB 等四个算法的样本容量平均值与标准差

	Yeast Cell	BCLL	RatStrain	PBC
CSB	18.82 ± 0.71	7.86 ± 0.40	71.96 ± 3.96	202.79 ± 5.31
FAB	**23.01 ± 2.67**	**8.03 ± 0.30**	**80.59 ± 2.82**	259.41 ± 5.09
CSFAB	18.35 ± 0.47	8.0 ± 0.0	76.01 ± 1.79	**267.79 ± 2.65**
PSOB	22.66 ± 3.23	7.98 ± 0.58	77.08 ± 3.44	220.94 ± 6.76

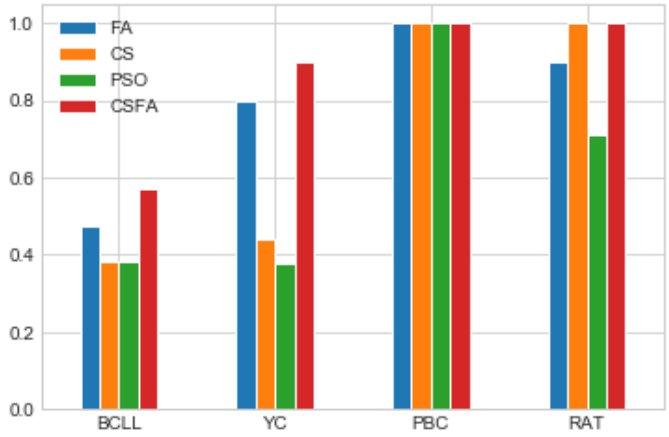
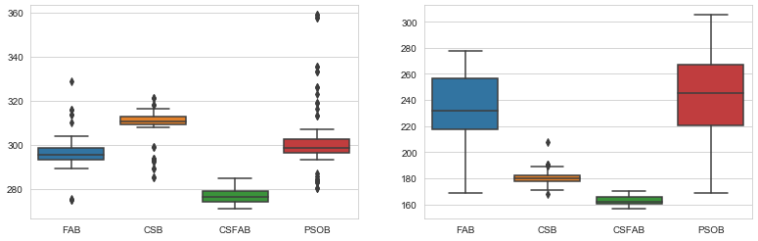


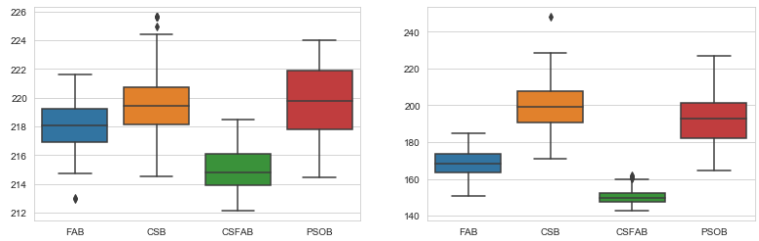
图 3-2 CSFAB 等四个算法的覆盖率

从图3-3可以看出，CSFAB 在全部的数据集上都找得了 MSR 最小的双聚类，并且优势明显。一方面，这是由于适应值函数中 MSR 所占的比重是相对较大的，这也解释了在样本容量上，CSFAB 表现并不突出；另一方面，结合 Lavy 飞行和萤火虫，使得算法有了更强的寻优能力。值得一提的是，CSFAB 算法的四分位间距是最小的，而且其他算法出现了较多的异常点，这一现象说明了 CSFAB 算法的稳定性。

为了减少对 MSR 的倾向性，图3-4展示了没有参与适应值函数的评价指标相关系数 RI，值得注意的是，该指标越大则质量越好。从图中可以看出，CSFAB 仅



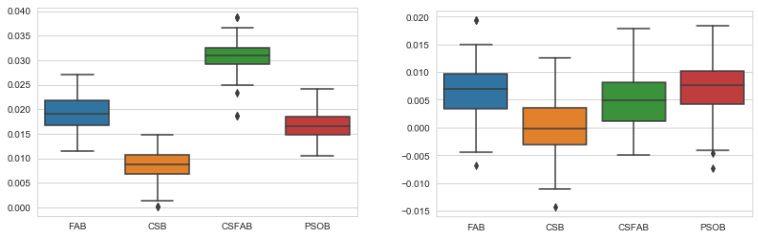
a) BCLL　　b) Yeast Cell



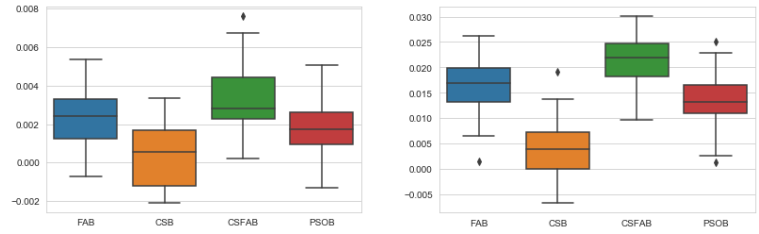
c) PBC　　d) RatStrain

图 3-3 CSFAB 等四个算法的 MSR

在 Yeast Cell 数据集紧跟在 PSOB，FAB 之后，另外三个数据集都取得了最优的结果。



a) BCLL　　b) Yeast Cell



c) PBC　　d) RatStrain

图 3-4 CSFAB 等四个算法的 RI

以上分析表明，尽管 CSFAB 在样本容量方面没有取得足够的优势，但是在基因容量和 MSR 以及 RI 上都取得了显著的成绩。不过，对于基因表达数据的一个

双聚类，最终还需要以生物质量指标来判断优劣。

3.3.3 CSFAB 的生物验证指标比较分析

虽然 CSFAB 在质量评价指标上表现不俗，但是对于双聚类来说，还是要找到具有生物意义的双聚类才行。因此，我们对实验中得到的双聚类进行了 GO 富集分析，并通过图表的方式展示。表3-5给出了 CSFAB 等四个算法的 WEScore 平均值与标准差。

由表3-5可知，在 RatStrain 和 PBC 这两个大数据集上，CSFAB 算法是最优的，而且在 Yeast Cell 数据集上是次优的，仅比最优的 FAB 低了 2.5 个百分点。这充分说明 CSFAB 算法更适合在大规模的数据上搜索相对其它算法难以搜索到的双聚类。

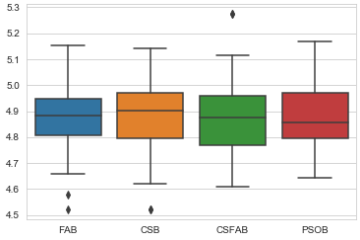
表 3-5 CSFAB 等四个算法的 WEScore 平均值与标准差

	Yeast Cell	BCLL	RatStrain	PBC
CSB	43.59 ± 7.45	**231.93 ± 14.10**	124.20 ± 44.52	137.90 ± 10.54
FAB	**45.76 ± 8.34**	230.55 ± 14.91	119.85 ± 41.61	149.24 ± 9.22
CSFAB	44.59 ± 7.17	228.71 ± 14.55	**124.91 ± 42.06**	**153.49 ± 12.73**
PSOB	43.27 ± 8.03	230.89 ± 14.92	115.30 ± 38.60	143.24 ± 8.30

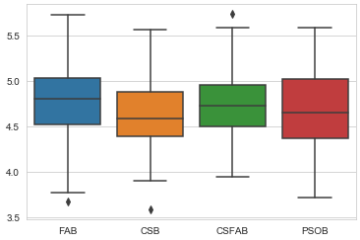
图3-5为 CSFAB 等四个算法的 meanPValue。与 WEScore 吻合的是，数据集越大，则 CSFAB 算法的表现越好。在规模最大的 PBC 数据集上，CSFAB 的优势最为明显。同样在 Yeast Cell 数据集上是次优的，仅比最优的 FAB 低了 0.7 个百分点。

3.4 本章小结

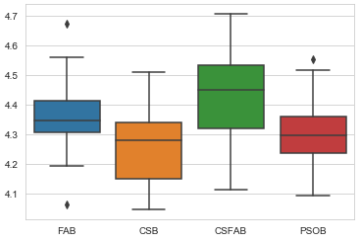
本文为了解决元启发式算法在双聚类时覆盖率不高和生物意义不明显的问题，提出了将 CS 算法和 FA 算法结合的 CSFAB 算法。首先提出了三种混合的方案，经实验比较后选择了嵌套执行的方案。然后，在四个数据集上对 CSB，FAB 和 PSOB 等四个算法进行了讨论分析。最后，实验证明，CSFAB 算法表现稳定，不仅提高了双聚类的多样性，而且保证了其生物意义，在规模大的数据集上体现得更加明显。



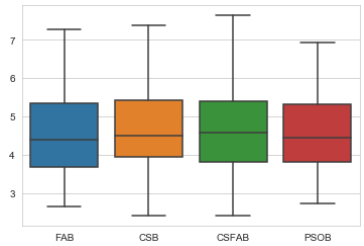
a) BCLL



b) Yeast Cell



c) PBC



d) RatStrain

图 3-5 CSFAB 等四个算法的 meanPValue

第 4 章 基于多目标 BFO 优化的双聚类算法

多目标优化问题在工业界和生活中广泛存在，如著名背包问题和旅行商问题。前面提到，双聚类有多个质量评价指标，其中一些是存在竞争的关系。基因表达数据的双聚类分析本质上就是一个多目标优化问题, 而且已经有研究将多目标优化算法引入到基因表达数据的双聚类分析中。同时，因为单目标优化每次仅能找到一个最优解，会存在效率问题，所以本章拟对细菌觅食算法进行改造，使其适合进行双聚类分析。本章先简要介绍多目标优化的基本知识，然后在根据细菌觅食算法的特点进行多目标的改进，最后通过实验进行算法的分析。

4.1 多目标优化问题的基本概念

假设 $S \subset \mathbb{R}^n$ 为一个 n 维的搜索空间，$f_i(x), i = 1, ..., k$ 为定义在 S 上的 k 个目标函数，并且定义向量函数 $f(x)$ 和 m 个不同的限制函数如下。

$$f(x) = [f_1(x), f_2(x), ..., f_k(x)] \tag{4-1}$$

$$g(x) \leq 0, i = 1, ..., m \tag{4-2}$$

然后，我们想要找到一个解 $x^* = (x_1^*, x_2^*, ..., x_n^*)$ 使得 $f(x)$ 最小。但是，目标函数 $f_i(x)$ 之间可能是互相冲突的，这使得不可能在 S 上找到一个全局的最优解。由于这个缘故，我们需要恰当地定义在多目标问题上的优化问题。

给定 $u = (u_1, ..., u_n)$ 和 $v = (v_1, ..., v_n)$ 为搜索空间 S 上的两个向量，当且仅当对于所有的 $i = 1, 2, ..., n$，$u_i \leq v_i$ 都成立且至少有一维 $u_i < v_i$ 成立，我们称 u 支配 v。这一性质也称为帕雷托支配（Pareto Dominance）。当且仅当 S 中没有任何一个解 y 支配 x，那么解 x 是该多目标问题的帕雷托最优解。也就是说 x 在 S 中是非支配的。S 中也许会存在多个非支配解，它们的集合被称为帕雷托最优解集，并用 P^* 表示。

$$PF^* = \{f(x) : x \in P^*\} \tag{4-3}$$

被称为帕雷托前沿。帕雷托前沿可以是不连续的，并且部分是凸而部分是非凸的。这种性质可以视为多目标优化问题的难点所在。

基于帕雷托最优解的定义，多目标优化问题的主要目标可以看作对帕雷托最优解的寻找。然而，帕雷托最优解可能是无穷的，受限于计算时间和空间，我们只能追求一个更加实际的目标。因此，我们只能尽可能地寻找帕雷托最优解，使其

帕雷托前沿经可能的扩张，与真实的帕雷托前沿的误差尽量小。

4.2 基于多目标 BFO 搜索双聚类算法

将单目标的 BFO 优化算法改进成多目标，需要对其趋向性操作和复制操作作出相应修改。本节结合 Lavy 飞行和多目标优化问题，提出基于多目标 BFO 搜索双聚类算法，并命名为 MOBFOB。

4.2.1 编码设计

该算法的编码设计依然采用连续值转为二进制的方案，基因表达矩阵 $E(X,Y)$ 上的一个双聚类用比特串 $x_p = (g_1, \ldots, g_i, \ldots, g_m, s_1, \ldots, s_j, \ldots, s_n)$，$p = 1, \ldots, N$, 来表示。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 $1 \le j \le n$。

4.2.2 适应值函数设计

不同于上一章将 MSR，GV 和 CV 通过权重系数直接相加的做法，结合多目标的情况，设计适应值函数如下。

$$f(x) = [MSR(x), -GV(x), -CV(x)] \tag{4-4}$$

这里一共有三个目标函数，根据帕雷托支配的定义，如果 $f(x_a) \le f(x_b)$ 严格成立且至少存在某个目标函数 $f_i(x)$，使得 $f_i(x_a) < f_i(x_b)$, 那么 x_a 支配 x_b。

4.2.3 多目标趋向性操作

趋向性操作是指细菌在搜索区间中的随机搜索，每走一步都要和之前的解进行比较，如果新位置优于旧位置，则进行跟新，这是算法收敛的保证。同时，为了增加种群的多样性，本文为 BFO 增加了 Levy 飞行。而在多目标问题中，需要基于 Pareto 支配关系来决定是否更新位置。对于旧位置 x_{old} 和新位置 x_{new}，如果之间相互支配，则选择支配位置而淘汰被支配位置；否则，进行归一化之后依据权重系数进行比较，如算法4-1所示。

4.2.4 多目标复制操作

在标准的单目标的细菌觅食算法中，根据种群的适应值进行排序，并淘汰排在后面的细菌，同时将排在前面的细菌复制，保持种群的大小不变。但对于多个目标函数，需要自己定义一个排序规则。本文采取的是根据被被支配的次数来排序，被支配的次数越少则该个体越优。首先进行两两支配判断，然后统计出每个个体被支配的次数，最后根据次数排序。

算法 4-1 归一化比较

Input: 新旧位置 x_{new}，x_{old}，目标函数的权重 $weight$

Output: 较优的位置

$f_{new} = f(x_{new})$ //计算新位置的适应值

$f_{old} = f(x_{old})$ //计算旧位置的适应值

$f_{total} = f_{new} + f_{old}$

$percent_{new} = f_{new} \div f_{total}$ //分别计算新旧个体中相同

$percent_{new} = f_{new} \div f_{total}$ //目标的函数值所占的比例

$sub = percent_{new} - percent_{old}$ //计算比例的差值

$rate = sub * weight$ //矩阵相乘各目标函数的权重

if $rate > 0$ **then**

  return x_{new} //若 rate 大于 0，则新位置更优

else

  return x_{old} //否则，旧位置更优

end

4.2.5 外部集存放策略

为了维护种群的多样性，保存搜索过程中的非支配解，本文引入了外部集。每经过一次复制操作后，对于新产生的非支配解集，按照下面三个步骤加入到外部集中。

1. 将新产生的非支配解集加入到外部集中并去重，如果外部集的大小没有变化，则说明都是重复的，直接返回，否则执行第（2）步

2. 计算外部集每个个体的被支配次数，然后淘汰支配次数非零的被支配解，如果剩余个数小于事先给定的阈值，则直接返回，否则执行（3）步

3. 剩余的都是非支配解，为了维持外部集的大小需要进行择优，具体过程如算法4-2所示。

4.3 实验结果及分析

为了验证 MOBFOB 算法的有效性，本文分别在每个数据集上执行了五次，每次执行生成 20 个帕雷托解，每个数据集生成 100 个双聚类。其中，种群规模 $nPop = 100$，趋化次数 $Nc = 50$，复制操作次数 $Nre = 5$，驱散次数 $Ned = 2$，各目标函数的权重 $weight = [-0.7, 0.2, 0.1]$。数据集的详细信息之前已经介绍过，这里不再赘述。

算法 4-2 更新外部集

Input: 帕雷托最优解集 P^*，外部集最大个数 outPSize, 目标函数权重 weight

Output: 更新后的外部集

$PF^* = \{f(x) : x \in P^*\}$ //计算帕雷托前沿

$f_{total} = sum(PF^*)$ //按目标函数相加

$percent = PF^* \div f_{total}$ //计算个体在各目标函数所占的比例

$ave = 1 \div |P^*|$ //计算平均的比例

$sub = percent - ave$ //个体比例减去平均比例

$rate = sub * weight$ //矩阵相乘各目标函数的权重

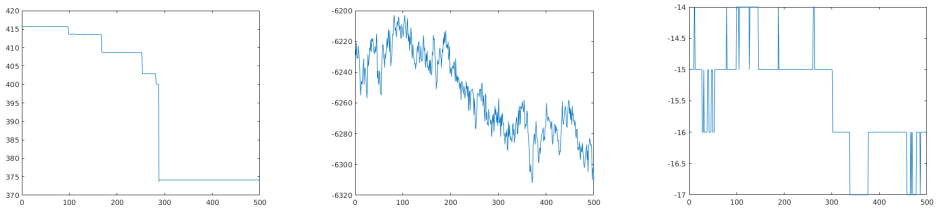
$sort(P^*, rate)$ //排序

$outPareto = P^*[1 : outPSize, :]$ //淘汰排在 outPSize 之后的解

return outPareto

4.3.1 质量验证指标

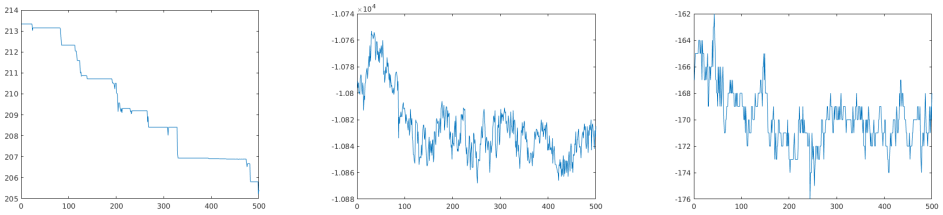
图4-1到图4-4分别是 MOBFO 算法在各数据集上，每次趋化操作后种群最小适应值的变化曲线。从图中可以看出，尽管在 GV 和 CV 指标上会出现明显的起伏，但是总体的趋势还是下降的。需要额外提示的是，纵坐标为 GV 和 CV 的负数，越低则容量越大。而 MSR 一直在稳步下降，这是因为在进行趋化操作时，对于 MSR 的权重是最大的，所以算法会在两个解互不支配时，牺牲一部分容量来换取 MSR 的下降，这也解释了 GV 和 CV 的起伏。



a) MSR b) GV c) CV

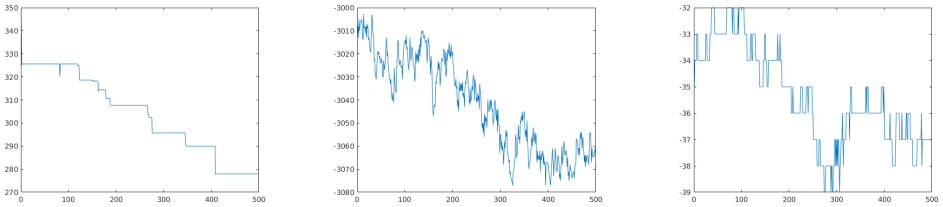
图 4-1 MOBFO 在 BCLL 数据集上适应值变化曲线

从适应值的变化曲线可以看出，算法能够在可行域上找到 MSR 较低但容量高的双聚类，为了更准确的表示算法双聚类的质量，表4-1为 MOBFOB 算法在各数据集上的各质量指标的平均值和标准差。跟3.3节中的实验数据相比，MOBFOB 并没有很突出的表现，但是，MOBFOB 由于外部集保存了多个帕雷托解，所以在执行效率上有着比单目标优化算法足够多的优势。



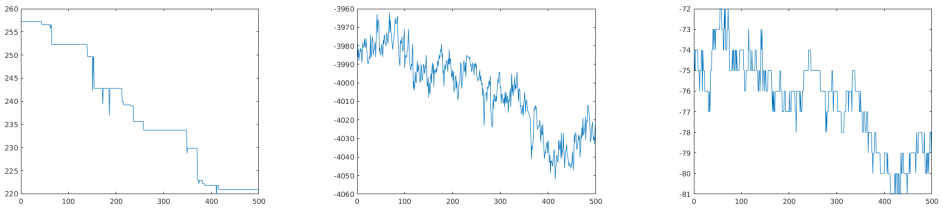
a) MSR　　b) GV　　c) CV

图 4-2 MOBFO 在 PBC 数据集上适应值变化曲线



a) MSR　　b) GV　　c) CV

图 4-3 MOBFO 在 Yeast Cell 数据集上适应值变化曲线



a) MSR　　b) GV　　c) CV

图 4-4 MOBFO 在 RatStrain 数据集上适应值变化曲线

表 4-1 MOBFOB 在各数据集上的质量指标的平均值和标准差

	BCLL	Yeast Cell	RatStrain	PBC
MSR	408.103 ± 54.16	344.99 ± 18.57	253.95 ± 13.15	215.49 ± 4.27
GV	6154.86 ± 65.08	2954.80 ± 31.75	3912.67 ± 38.76	10651.02 ± 80.24
CV	9.41 ± 2.82	27.05 ± 4.38	63.55 ± 7.86	156.78 ± 10.98
RI	−0.00093 ± 0.00309	0.00053 ± 0.00432	0.00101 ± 0.00559	0.00071 ± 0.00143
Var	468.17 ± 44.06	360.074 ± 17.46	263.75 ± 13.07	221.19 ± 4.42
HV	0.98 ± 0.004	0.99 ± 0.001	0.97 ± 0.004	0.98 ± 0.001

4.3.2 生物验证指标

为了验证 MOBFOB 算法能够找到具有生物意义的双聚类，本文对得到的双聚类进行了详细的 GO 分析，生物验证指标的平均值和标准差如表4-2所示。图4-5给出了 MOBFOB 在 BCLL 数据集上的某个双聚类（编号 1）GO 分析后部分较为显著的 GO 项，其中包括了 15 个生物过程（Biological Process），10 个细胞组成（Cellualr Component），15 个分子功能（Molecular Function）。该双聚类由 4529 个基因和 7 个样本组成。由图4-5可知，该双聚类对于生物过程的富集程度要高于其他两种, 而且所富集的 GO 项和相关的基因个数都表明了其具有一定的生物意义。

表 4-2 MOBFOB 在各数据集上的生物验证指标的平均值和标准差

	BCLL	Yeast Cell	RatStrain	PBC
WEScore	231.37 ± 13.98	46.15 ± 8.38	137.07 ± 48.68	138.07 ± 11.49
meanPValue	4.89 ± 0.13	4.67 ± 0.40	4.87 ± 1.18	4.24 ± 0.12
rateGeneTerm	1.71 ± 0.06	1.57 ± 0.18	0.16 ± 0.05	3.32 ± 0.22

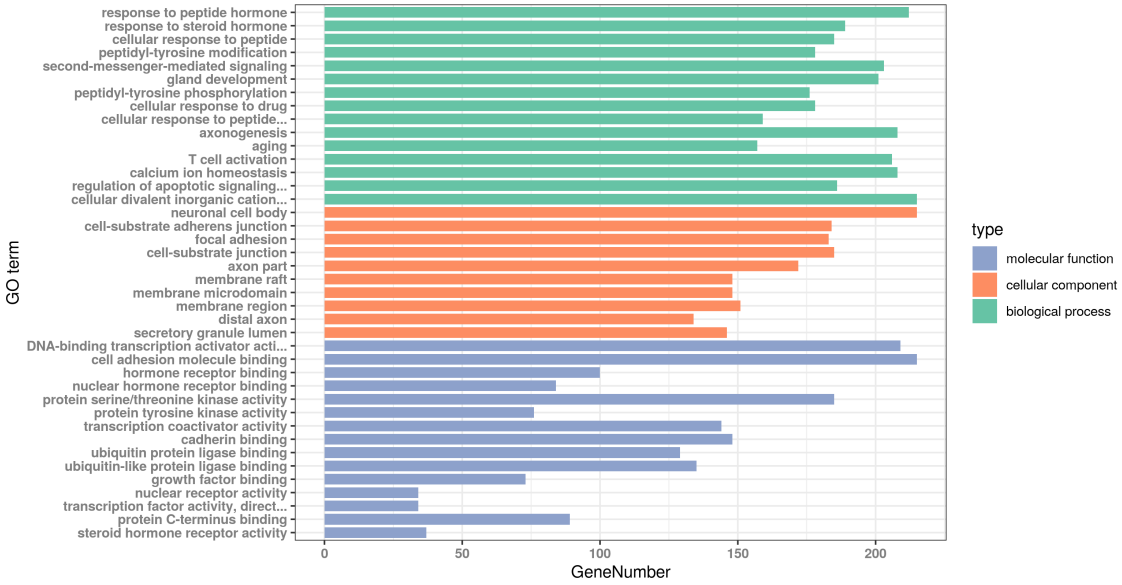


图 4-5 MOBFOB 算法得到的双聚类（编号 1）主要的 GO 项

表4-3为双聚类（编号 1）相关 GO 项的详细信息。例如，表中第二行对应的 ID 为 GO:0048545，该 GO 项代表的是对类固醇激素的反应，4529 个基因中有 189 个与该 GO 项相关，从 BCLL 数据集上随机选择 4529 个基因至少包含这 189 个基因的概率是 4.52E-23。这说明了该双聚类的基因集合中存在功能紧密相关的子集合，形成了一种明显的基因表达模式。

表 4-3 双聚类（编号 1）相关 GO 项的详细信息

GOID	调整 P 值	基因个数	描述
GO:0043434	3.74E-25	212	response to peptide hormone
GO:0048545	4.52E-23	189	response to steroid hormone
GO:0019932	3.53E-21	203	second-messenger-mediated signaling
GO:0043025	1.37E-20	215	neuronal cell body
GO:0059240	2.68E-20	184	cell-substrate adherens junction
GO:0004674	2.23E-12	185	protein serine/threonine kinase activity
GO:0050839	1.77E-15	215	cell adhesion molecule binding

4.4 本章小结

本章为了解决单目标优化的效率问题，将细菌觅食算法的多目标版本用于双聚类分析。首先简要介绍了多目标优化中的相关概念，然后结合多目标优化和细菌觅食算法的特点，提出了 MOBFOB 算法。实验数据证明了算法的有效性，即能够在迭代中优化多个目标，并最终得到有显著生物意义的双聚类，达到预期目标。

第 5 章 总结与展望

5.1 论文的工作总结

科学技术的进步使得人类有了更多的途径来认知事物，而数据则是人类认知事物的特殊媒介。随着高新科技的迅猛发展，越来越多的数据被产生出来，挖掘出蕴藏在数据中的价值，将带来巨大的经济和社会价值。而基因表达数据作为一种特殊的数据，保存着生命密码，能够帮助人们更加了解自身和自然，对医疗以及生态保护都有长远的意义。因为生物体中的细胞种类繁多和基因之间互相调控等方面的原因，基因表达数据有着较为复杂，体量大和增长速度快的特点。因此对于基因表达数据的研究一直是生物信息学领域的难点和重点。

聚类一直是人们分析数据时常用的手段之一。而传统的聚类方法无法找到基因表达数据中的局部模式，所以双聚类分析成为了最主要的挖掘工具。在有的双聚类分析算法中，基于元启发式的优化算法和多目标优化被应用到寻找双聚类。因此，本文以解决双聚类算法中常见的问题，如质量评价不高和生物意义不明显，为出发点，运用算法融合和多目标优化的手段，做了一些总结和探索性的工作，主要包括：

1. 介绍了基因表达数据双聚类分析的研究背景和意义。然后，对基因表达数据双聚类分析的基础概念和和数学定义进行了阐述，如基因表达数据的数学模型，双聚类的相关概念，常见的双聚类分析方法的分类以及常用的双聚类验证指标。最后，对于本文所涉及的群智能算法也进行了说明

2. 本文基于布谷鸟算法中的 levy 飞行和萤火虫算法中的亮度吸引作用，将两个算法嵌套融合，提出了 CSFAB 双聚类算法。然后讨论了适应值函数，混合方案和停止条件，并通过实验证明了所提出算法具有更高的全局搜索能力和收敛能力，达到预期目标。

3. 本文从多目标优化的角度，对细菌觅食算法进行了相应的设计，如种群的趋化操作和迁徙操作。基于多目标细菌觅食算法，提出 MOBFOB 双聚类算法。算法对双聚类的均方残差和容量同时优化，使得成对立关系的指标都能得到优化，最终得到占优的双聚类。实验证明，MOBFOB 算法能都有效地找到高质量的双聚类。

5.2 后续工作展望

基因表达数据的双聚类分析是一个很大且复杂的研究方向。由于时间和条件

有限，本文仅仅是对群智能算法做了一些工作。本工作仍有不少问题需要进一步的研究，包括：

1. 由于性能的关系，质量评价指标和生物意义被分别考虑。优化算法也只是根据质量评价指标来寻优，这与基因表达数据双聚类分析的目的是存在一定的偏差的。最近，Nepomuceno 等使用 SSB 双聚类算法直接以 GO 注释信息来指导搜索，取得了一定的效果，但仍有很多不足。如何更好地直接利用 GO 信息来需找双聚类，还需要进一步的研究。

2. 二维基因表达数据无法全面的记录所需要的信息。目前所用的基因表达数据都是二维矩阵形式的，而现实中，基因的表达是有时序关系的。只有将时间序列这一维度引入进来，才能够更完全更科学地找到双聚类。

3. 将算法优势互补的方法不止融合一种，还可以通过集成的方法来提高双聚类结果的质量。在机器学习领域，集成能够将多个相对较弱的模型组合成一个效果更好的模型，如随机森林算法，XGBoost 算法等。如何将不同的双聚类算法集成为一个效果更好的算法，仍处于研究初始阶段，这为下一步的研究指明了明确的方向。