软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2014,25(5):929−938 [doi: 10.13328/j.cnki.jos.004547]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

邻域种子的启发式 454 序列聚类方法*

陈 伟[1,2], 程咏梅[1], 张绍武[1], 潘 泉[1]

[1](西北工业大学 自动化学院,陕西 西安 710072)

[2](Department of Biostatistics, Yale University, USA)

通讯作者: 张绍武, E-mail: zhangsw@nwpu.edu, http://www.nwpu.edu.cn

摘 要: 随着二代测序技术的发展,产生了海量 16S rRNA 基因序列数据.如何有效地挖掘这些数据中隐藏的基因组学信息,是当前研究的热点与难点.序列聚类研究如何将来源于同一物种的序列合并在一起,其构成了物种多样性、结构及功能多样性研究的基础.针对 454 测序误差的来源特点,提出一种基于邻域种子序列的启发式序列聚类算法(NbHClust).实验结果表明,该算法具有良好的鲁棒性能.与传统启发式序列聚类算法相比,该算法能够降低操作分类单元(operational taxonomy unit,简称 OTU)过估计问题,提高聚类精度,有效地进行操作分类单元计算.

关键词: 二代测序技术;操作分类单元;物种多样性;16S rRNA 基因;序列聚类

中图法分类号: TP181

中文引用格式: 陈伟,程咏梅,张绍武,潘泉.邻域种子的启发式 454 序列聚类方法.软件学报,2014,25(5):929−938. http://www.jos.org.cn/1000-9825/4547.htm

英文引用格式: Chen W, Cheng YM, Zhang SW, Pan Q. Heuristic clustering method based on neighbor-seeds for 454 sequencing data. Ruan Jian Xue Bao/Journal of Software, 2014,25(5):929−938 (in Chinese). http://www.jos.org.cn/1000-9825/4547.htm

Heuristic Clustering Method Based on Neighbor-Seeds for 454 Sequencing Data

CHEN Wei[1,2], CHENG Yong-Mei[1], ZHANG Shao-Wu[1], PAN Quan[1]

[1](College of Automation, Northwestern Polytechnical University, Xi'an 710072, China)

[2](Department of Biostatistics, Yale University, USA)

Corresponding author: ZHANG Shao-Wu, E-mail: zhangsw@nwpu.edu, http://www.nwpu.edu.cn

Abstract: With the development of next-generation sequencing technology, a large number of 16S rRNA gene reads have been collected. A key and important issue is to develop novel methods for mining the hidden information among those data. Sequence clustering aims to find the natural groups of large-scale data which can help us to understand the species, functional and structural diversity of microbial communities. This present work proposes a heuristic clustering method based on Neighbor-seeds, named NbHClust, for 454 sequencing data. The results show that this method can reduce extent of overestimation of operational taxonomy unit (OTU) and have a good robust and high clustering accuracy.

Key words: second-generation sequencing technology; operational taxonomy unit; species diversity; 16S rRNA gene; sequence clustering

聚类分析是一种无监督学习方法,广泛应用于图像处理、数据挖掘、模式识别、生物信息等领域[1,2],已成为一种有效的大规模数据分析手段.

与传统的数值型聚类方法稍有不同,本文的研究对象为 454 测序序列数据.近年来,随着宏基因组学的建立,尤其是二代测序技术的发展(如 454 测序,Illumina 测序),产生了海量的测序数据,使得从整体上研究微生物种群

* 基金项目: 国家自然科学基金(61170134, 61135001); 航空基金(20100853010); 西安市科技计划(CXY1350(2)); 西北工业大学博士创新基金(cx201017)

收稿时间: 2013-07-10; 修改时间: 2013-10-11; 定稿时间: 2013-12-03

的组成与结构成为可能.然而,如何鉴定这些序列的功能并区分序列的差异,是摆在生物学家面前的一个重要问题,亟需研究人员发展高效的计算方法来处理上述序列数据.

由于 16S rRNA 序列含有丰富的进化信息并且具有高度保守性,目前已被广泛用于微生物进化及物种多样性分析.基于 16S rRNA 序列的分析方法主要分为分类学方法与独立分类学方法[3].分类学方法依据现有的注释序列,在一定的分类器准则下预测新序列的结构及功能,通常具有较高准确度,但其受参考数据集完整性约束,难以对大量未知序列进行有效的功能注释[4,5].独立分类学方法则无需任何先验信息,采用聚类策略分析 16S rRNA 序列数据,进行操作分类单元(通常将 16S rRNA 序列聚类单元称为操作分类单元,简称 OTU)估计,从而推断种群的组成及结构[6,7].

近年来,随着宏基因组学发展,国外陆续出现了一些 16S rRNA 序列聚类算法.Schloss 等人基于序列比对建立相似度矩阵,然后采用层次聚类策略估计操作分类单元(Mothur),取得了较好的结果[6].为了降低 Mothur 算法的复杂度,Sun 等人提出了一种基于 kmer 预过滤的层次序列聚类方法,在一定程度上提高了算法效率[7].Wang 等人基于网络社团挖掘思想估计操作分类单元,以节点表示 16S rRNA 序列,以序列相似性表示边权重,取得了较好的聚类精度[8].

层次聚类与基于网络的聚类算法在大规模序列聚类时普遍存在着复杂度高及内存开销大的问题,例如,采用网络或者层次聚类算法分析 454 测序仪单次测序序列,内存需求高达 1T,即使采用稀疏存储方式,也高达 >100G.鉴于层次聚类与网络聚类算法在大规模序列数据聚类方面的局限性,Godzik 等人提出了一种基于启发式策略的 OTU 聚类算法,通过反复地贪婪搜索,种子扩增完成聚类过程.由于在聚类过程中不涉及相似度矩阵的计算、存储及读写操作,只需存储有限种子序列,有效降低了时间复杂度与内存需求,尤其适合大规模 rRNA 序列聚类(例如,序列数>106)[9].Russell 等人基于数据压缩原理,采用 Lempel-Ziv 复杂度作为序列相似性度量标准,提出了一种基于语法距离的启发式 OTU 聚类算法 GramCluster[10].类似的算法还包括 DNAClust[11]和 Uclust[12].

尽管国外在序列聚类方面已经取得了一定的研究成果,但从国内的二代测序序列研究来看,主要还是集中于物种多样性估计及种群差异性分析[13,14],序列聚类算法方面的研究较少.

启发式聚类算法以牺牲聚类精度换取算法效率,而且容易产生大量的 singleton 聚类单元,导致 OTU 过估计问题.研究结果表明,由测序误差等因素导致的虚假聚类单元包含的元素较少,即,簇较小,而真实聚类结构一般包含较多元素.

本文在分析 454 测序误差特性的基础上提出了一种新的启发式聚类算法:首先,对每一个种子序列根据其误差特性产生种子邻域序列集合;然后,基于启发式策略完成聚类过程;最后,基于聚类单元大小约束对聚类结果进行属性约减.实验结果表明,该算法具有较好的鲁棒性能,提高了聚类精度.

1 基于邻域种子的启发式序列聚类算法

由于测序误差的存在,目前的大部分启发式聚类算法通常都会导致 OTU 过估计问题,其主要原因在于:定义相似性度量函数时,同等对待测序误差及序列真实差异.针对这一问题,本文首先在分析 454 测序仪误差特性的基础上,基于种子序列的同聚体分布,定义种子序列的邻近序列集合;然后,基于邻域种子序列集合进行启发式搜索、扩增形成聚类单元.其流程如图 1 所示.

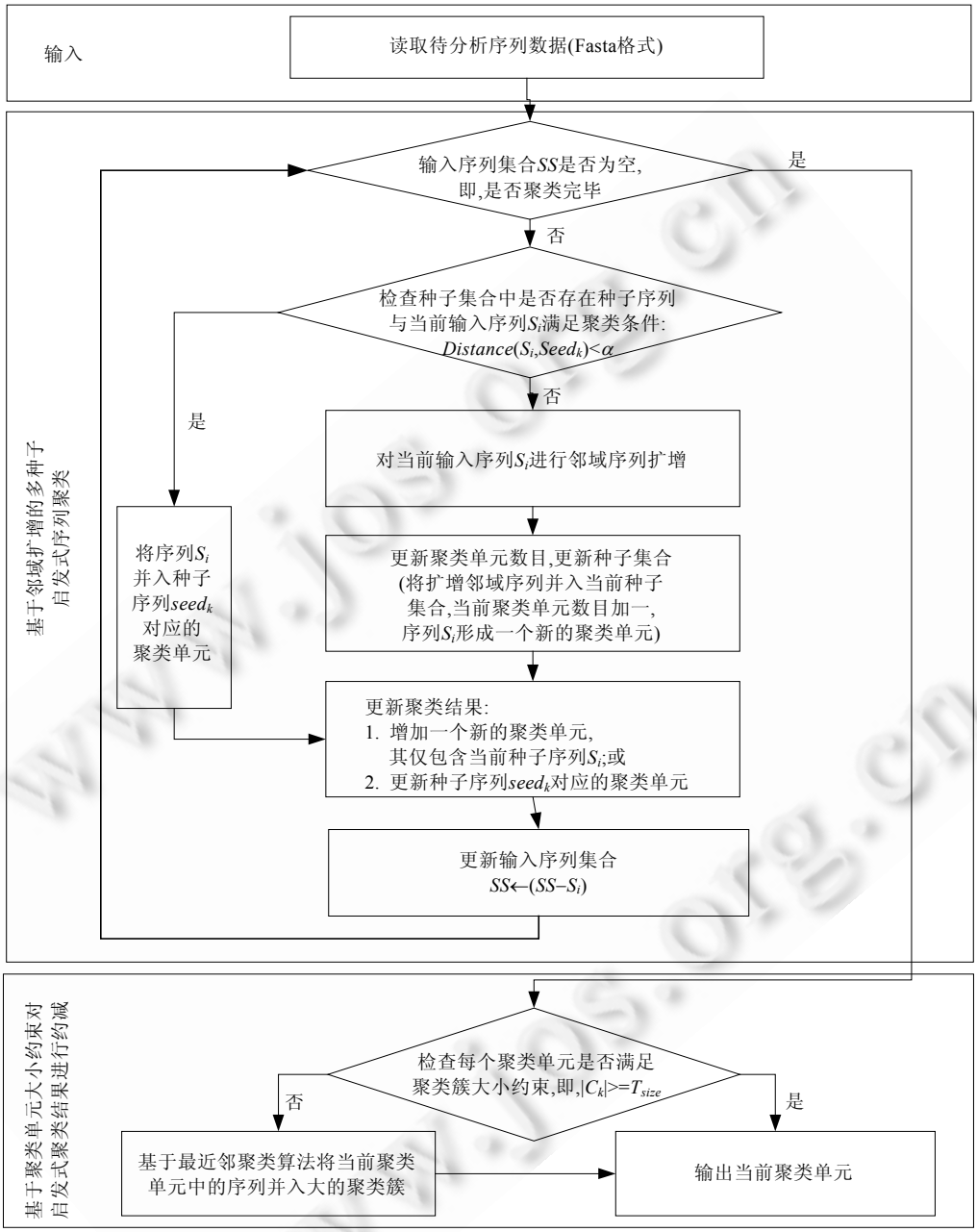


Fig.1 Flowchart for the heuristic clustering method based on neighbor-seeds for 454 sequences

图 1 基于邻域种子的启发式 454 序列聚类流程图

1.1 邻域序列扩增

454 是重要的二代测序技术之一,通过检测荧光信号强度,达到测定同聚物长度的目的.研究结果表明,454 测序误差主要来源于碱基删除/插入错误,同聚体(由连续相同碱基组成的结构)越长,出现删除/插入的概率就越大.针对 454 测序仪这一误差特性,本文首先对每一个种子序列依据其同聚体分布进行扩增,形成种子序列的邻域序列集合.

定义 1. 16S rRNA 序列 S 是定义在字符集合 $CC=\{A,C,G,T\}$ 上的序列,其中,$S(i)$表示序列 S 的第 i 个元素,$|S|$代表序列 S 的长度.

定义 2. $S(i,j)$是 S 中起始于位置 i 终止于位置 j 的子序列,如果 $S(i)=S(i+1)=S(i+2)=\ldots=S(j)$,则称子序列 $S(i,j)$是长度为$|j-i+1|$的同聚体,记为 $S_h(i,j)$.

定义 3. 若序列 S^1 与序列 S 的差异碱基数为δ,并且差异碱基位于同聚体末端,则称 S^1 是 S 的δ-邻域序列.例如,$S_h(i,j)$是序列 S 的一个长度为$|j-i+1|$的同聚体,若 $S^1=S(1,j)S(j)S(j+1,|S|)$,或者 $S^1=S(1,j-1)S(j+1,|S|)$(其中,$S(1,j)S(j)S(j+1,|S|)$表示子序列 $S(1,j)$,$S(j)$与 $S(j+1,|S|)$串联形成的新序列),则称 S^1 是 S 的 1-邻域序列.依此类推 S 的 2-邻域序列、δ-邻域序列.在实际应用中,δ通常取值为 2.

根据上述定义,邻域序列扩增算法可描述为:

算法 1. 邻域序列扩增算法.

输入:种子序列 S,l,δ.

输出:序列 S 的δ-邻域序列集合 $Neighbour_\delta(S)$.

(1) 初始化 $Neighbour_\delta(S)$,$Neighbour_\delta(S)\leftarrow S$;$l$ 为同聚体长度阈值,δ为最大允许差异碱基数.
//根据 454 测序仪特点,实际应用中,l 与δ的取值分别为 3 与 2

(2) While $i\leqslant|S|$, if $\exists j(j\leqslant|S|)$满足$|S_h(i,j)|\geqslant l(j>i)$,记录位置 j,$SH\leftarrow SH+j$.
//遍历 S 中的满足最小长度条件 l 的同聚体子序列

(3) For ($j=1$, $j\leqslant\delta$, i++)
从 SH 中随机选取 j 个元素,$SH_{i1},SH_{i2},\ldots,SH_{ij}$;通过 $Insert(SH_{ij},S)$和 $Delete(SH_{ij},S)$操作获得邻域序列:

$$Neighbour\leftarrow Insert(SH_{ij},S)||Delete(SH_{ij},S).$$

//$Delete(i,S)$表示删除 S 中位置 i 处元素 $S(i)$构成的新序列
//$Insert(i,S)$表示在 S 的第 i 个元素之后添加元素 $S(i)$形成的新序列
if ($Neighbour\notin Neighbour_\delta(S)$) //更新邻域序列集合,直至所有可能组合遍历完毕
$Neighbour_\delta(S)\leftarrow Neighbour_\delta(S)+Neighbour$;
End if
End For

注:$Neighbour_\delta(S)$表示序列 S 的δ-邻域序列集合,SH 是同聚体位置标记集合.

1.2 序列相似性

相似性度量函数是聚类问题的重要组成部分,与数值型对象不同,本文聚类对象是字符序列数据,因此,如何设计针对字符序列的相似性函数是算法成功的关键之一.根据是否采用序列比对,序列相似性定义函数可分为两类:一类是基于序列比对,另一类不依赖序列比对.通常,基于比对的方法能够更好地逼近序列间的真实差异,而非序列比对的方法则具有更小的复杂度.为了更好地逼近真实的聚类结果,本文提出了一种基于序列比对的相似性度量函数.假定种子序列为 $S^x(S^x\in Neighbour_\delta(S))$,其中,$Neighbour_\delta(S)$表示序列 S 的δ-邻域序列集合,当前输入序列为 S^y.S^x 与 S^y 的比对序列分别为 $X=x_1x_2x_3\ldots x_n$ 与 $Y=y_1y_2y_3\ldots y_n$,其中,$\{x_i,y_i\in(A,C,G,T,-)\}$,则 S^x 与 S^y 相似度定义为

$$Sim(S^x,S^y)=Sim(X,Y)=1-\frac{\omega_{S^x\to S}\sum_{i=1}^{n}s(x_i,y_i)}{n} \tag{1}$$

$$s(x_i,y_i)=\begin{cases}1, & \text{if } x_i\neq y_i\\ 0, & \text{else}\end{cases} \tag{2}$$

$$\omega_{S^x\to S}=\exp\left\{-\frac{\delta}{|S|}\right\} \tag{3}$$

其中,$s(x_i,y_i)$与 $\omega_{S^x\to S}$ 分别为指示函数与权重因子,$\omega_{S^x\to S}$ 表示序列 S 的δ-邻域序列 S^x 相对于序列 S 的权重,$|S|$表

示序列 S 的长度.

1.3 启发式序列聚类

由于层次聚类和基于网络的序列聚类算法通常需要存储、读写序列相似性矩阵,难以处理大规模序列数据,因此,本文采用启发式策略进行序列动态聚类,通过引入序列邻域扩增策略,适当地增加聚类单元的种子规模,基于聚类单元大小约束条件优化聚类结果,在保证较小的复杂度的同时,取得了较高的聚类精度.详细过程描述如算法 2 所示.

算法 2. 基于邻域序列的启发式聚类算法.

输入:序列 $X=\{S^1,S^2,\ldots,S^N\}$、聚类阈值α、聚类单元大小约束参数 *MinClusterSize*;

输出:序列集 X 的聚类结果 C.

Step 1. 初始化,种子集合 $Seeds=\{\varnothing\}$,$l=3$,$\delta=2$.

Step 2. 计算输入序列 S^i 与种子集合中序列的距离,如果$\exists k$,满足 $Sim(S^i,Seeds^k)\geq\alpha$,则将 S^i 赋给 $Seeds^k$ 对应的聚类单元;否则,转 Step 3. //$Seeds^k$ 表示种子集合 *Seeds* 中的第 k 条序列

$OTU^{Seed^k}=OTU^{Seeds^k}+S^i$, // OTU^{Seed^k} 表示序列 $Seeds^k$ 对应的聚类单元

$label(S^i)=\arg_{label}(Seeds^k)$. //$S^i$ 与 $Seeds^k$ 具有相同的聚类单元标号

Step 3. 根据邻域序列扩增算法(算法 1)产生序列 S^i 的δ-邻域序列 $Neighbour_\delta(S^i)$.

Step 4. 更新种子序列集合:

$Seeds\leftarrow Seeds+Neighbour_\delta(S^i)$,

$ClusterNum=ClusterNum+1$; //*ClusterNum* 表示聚类单元数目

$OTU^{s^i}=\{S^i\}$.

Step 5. 输入序列集合 X 是否处理完毕:若是,则转 Step 6;否则,转 Step 2.

Step 6. 根据参数 *MinClusterSize* 对初始聚类结果进行约减;

对于$\forall n$,如果$|OTU_n|<MinClusterSize$,则基于最近邻聚类算法对聚类单元 $n(OTU_n)$中的序列重新聚类,即,将其赋值为与其距离最近的聚类单元;否则,转 Step 7.

If $\forall n,|OTU_n|<MinClusterSize$ //$|OTU_n|$表示聚类单元$|OTU_n|$的势,即聚类单元大小

For (i=1:$|OTU_n|$)

$label(OTU_{n,i})=\arg\min_{k'}(d(OTU_{n,i},OTU_{k'}))$

subject to: $|OTU_{k'}|>MinClusterSize$ //$OTU_{n,i}$ 表示聚类单元 OTU_n 中的第 i 条序列

$k'\neq n$

End For

End if

Step 7. 输出当前聚类结果:$C=C+OTU_n$.

注:$d(OTU,OTU)=\min d(OTU,y),y\in OTU$,$d(X,Y)=1-Sim(X,Y)$.

1.4 评价指标

为了验证本文算法的有效性,分别采用真实数据集 Clone43[15]与模拟数据集 Gum_V6 进行算法评估,采用归一化最小距离(NID)与 F-value[16]作为聚类精度衡量标准.其中,NID 是一个整体性指标,与其他整体性聚类评价指标相比,NID 具有更严格的边界范围,并且满足归一化条件;而 F-value 则整合了准确率与召回率,度量了每一个聚类单元的细节信息.NID 与 F-value 的定义如下:

$$NID=1-\frac{I(S,C)}{\max(H(S),H(C))} \tag{4}$$

$$F_{i\bullet}=\max_{j=1}^{n}F_{ij},i=1,2,\ldots,m;F_{ave}=\frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}F_{ij} \tag{5}$$

$$F_{ij}=2p_{ij}r_{ij}/(p_{ij}+r_{ij}),\ \ p_{ij}=\frac{a_{ij}}{|C_j|},\ r_{ij}=\frac{a_{ij}}{|S_i|} \tag{6}$$

$$I(S,C)=\sum_{i=1}^{m}\sum_{j=1}^{n}\frac{a_{ij}}{N}\log\frac{a_{ij}/N}{|S_i||C_j|/N^2},\ i=1,2,...,m;\ j=1,2,...,n \tag{7}$$

$$H(S)=-\sum_{i=1}^{m}\frac{S_i}{N}\log\frac{S_i}{N},\ H(C)=-\sum_{j=1}^{n}\frac{C_j}{N}\log\frac{C_j}{N} \tag{8}$$

其中,$(S_1,S_2,\ldots,S_m)$为基准划分,$(C_1,C_2,\ldots,C_n)$代表聚类结果,α_{ij} 表示属于第 i 个物种但被划至第 j 个聚类单元的序列数量.NID 值越小,表明算法聚类结果越逼近基准划分结构,当算法聚类结果与真实结构完全一致时,*NID*=0; *F*-value 值越大,表明该聚类单元越接近真实类单元组成.

2 结果与分析

面向数值型的传统聚类,数据真实划分结构通常保持不变;而在 16S rRNA 序列分析中,聚类结构是动态变化的,通常,不同的相似性水平对应不同的分类学层次(种、属、科、目等),进而,真实聚类单元也是不同的.如真实数据集 Clone43,在种(species)层次,其对应的真实聚类单元是 43;而在属(genus)层次,其数目<43[15].因此,在 16S rRNA 序列数据分析中,聚类结果随聚类水平变化而变化.为了分析本文算法的有效性,基于默认参数设置(*MinClusterSize*=1),在一列系相似性水平(99%~95%,因为 16S rRNA 序列聚类分析中,97%的序列相似性通常被定义为 species-level,95%对应 genus-level[17,18])对实验数据集进行聚类分析.

2.1 真实数据集Clone43聚类结果

采用 Schloss 等人设计的真实数据集 Clone43 来验证本文算法的有效性.该数据集是由 Schloss 等人[15]基于 43 个真实物种的 16S rRNA 序列经 PCR 扩增,454 测序构成.剔除低质量序列之后,一共含有 202 340 条序列,平均长度为 65nt,在 97%的序列相似度下,理想的聚类单元数量为 43.其中,NbHClust,CD-HIT,Uclust 与 DNAClust 聚类结果如下图 2 所示.由于基准数据集 Clone43 的真实划分结构未知,也就是说,每个序列所属的真实簇单元(即所属物种)是未知的,因此我们采用聚类单元数目作为评价指标.

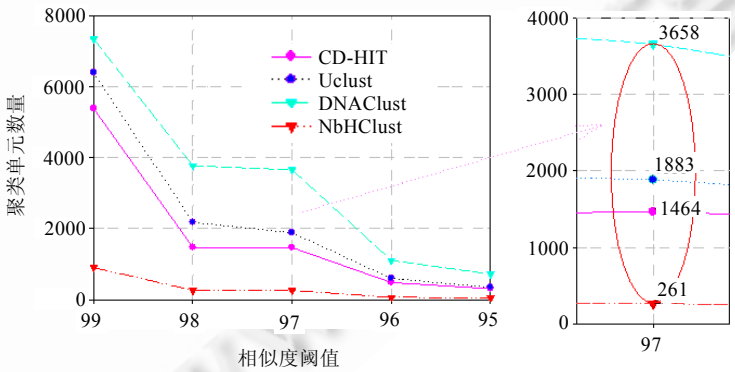


Fig.2 Clustering results for the dataset Clone43

图 2 数据集 Clone43 的聚类结果

从图 2 中可以看出,聚类单元数目随着聚类水平变化而变化.这主要是因为随着聚类阈值的增大,一些小的聚类单元将会逐次合并[5].在 97%的聚类水平下,NbHClust,CD-HIT,Uclust 与 DNAClust 的聚类单元数目都大于真实物种数量(43),其中,DNAClust 算法产生了~3700 个聚类单元,CD-HIT 算法产生了~1900 个聚类单元,Uclust 产生了~1400 个聚类划分,而 NbHClust 产生了~260 个聚类单元.NbHClust 算法的聚类单元数量明显小于其他几种算法,是 DNAClust 算法聚类数目的 7%,是 CD-HIT 的 19%,是 Uclust 的 14%.可见,NbHClust 算法明显降低了 OTU 数量,其返回的 OTU 值也更接近真实聚类单元数量,明显改善了过估计问题,提高了聚类性能.从图 2 中可

以看出,在 99%~95%的聚类阈值区间,NbHClust 算法的聚类单元数量变化范围相对最小,说明该算法具有更好的鲁棒性能.

本文算法中,我们根据最小聚类单元参数(*MinClusterSize*)对聚类结果进行了约减.

为了研究参数 *MinClusterSize* 对聚类结果的影响,在 97%的聚类水平下,选择不同的 *MinClusterSize* 值,其实验结果如图 3 所示.从图中可以看出,随着 *MinClusterSize* 取值的增大,聚类单元数目越来越少.这主要是因为大量小的聚类单元被重新赋值到了大的聚类单元当中.

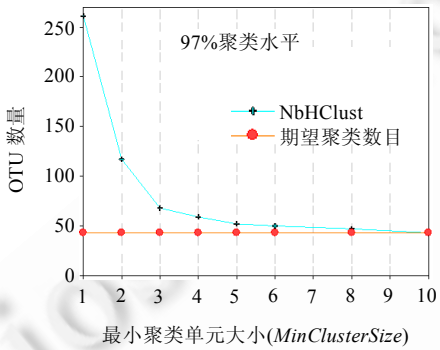
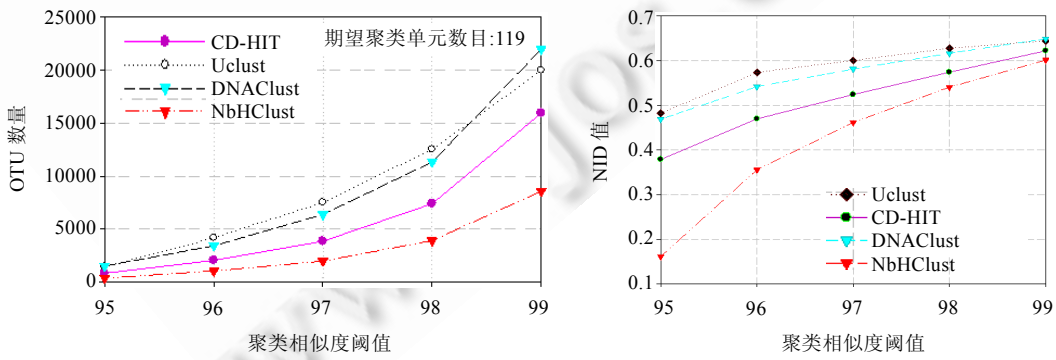


Fig.3 Effect of Parameter MinClusterSize (size of cluster) on clustering results

图 3 不同聚类单元大小约束下的聚类结果

2.2 模拟数据集实验结果

采用模拟数据集 Gum_V6 进行算法验证.在 16S rRNA 序列聚类分析中,由于缺乏基准数据集,因而很难评价各种算法的性能.为了解决这一问题,本文基于 RDP[4]数据库注释信息对测序数据集进行分类学注释,保留满足注释条件的序列.基于 Blast 局部比对方法对 Turnbaugh 等人[18]测序的肠道微生物数据集(gum microbiota)注释,采用 97%的序列相似性过滤序列,去掉被多个物种(species)注释的序列以及小于 10 个序列的物种.经过上述步骤,产生了一个包含~310k 条序列基准数据集,属于 119 个物种.理想情况下,97%的聚类阈值对应 119 个聚类单元,与 Clone43 相比,Gum_V6 的真实划分结构是已知的.对于模拟数据集 Gum_V6,聚类单元数目及相应 *NID* 值如图 4 所示.



(a) 不同聚类阈值下的聚类单元数量 (b) 不同聚类阈值下的 NID 值

Fig.4 Clustering results for simulated dataset Gum_V6

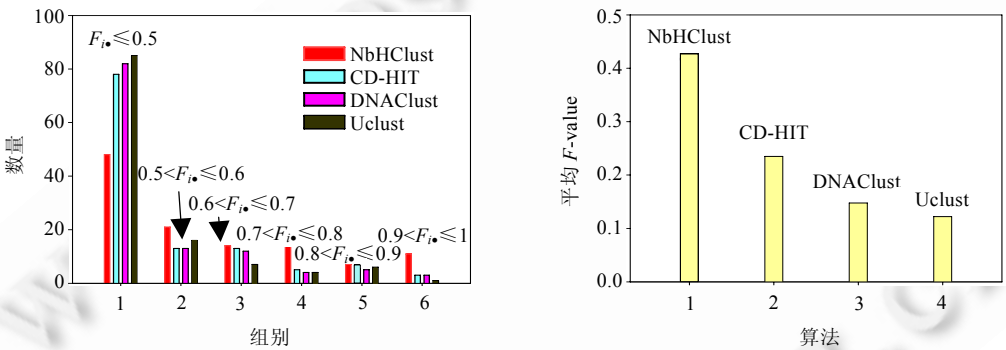
图 4 基于模拟数据集 Gum_V6 的聚类结果

从图 4 中可以看出,聚类相似度阈值对聚类结果影响较大.如,在 97%的聚类阈值下,对于 NbHClust 算法,其聚类单元数目~1 900,相应的 NID 值是 0.46;CD-HIT 的聚类单元数目是~3 800,对应的 NID 值是 0.52;对于 Uclust

算法,其产生了~7 500 个聚类单元,相应的 NID 值为 0.60;对于 DNAClust 算法,其产生了~6 300 个聚类单元,相应的 NID 值是 0.58.当聚类阈值为 96%时,4 种算法的 NID 值分别为 0.47,0.57,0.54,0.16.

从聚类单元数目来看,在 97%的聚类阈值下,4 种算法都存在 OTU 过估计问题,但 NbHClust 明显降低了过估计程度:在相同的聚类水平下,NbHClust 的聚类单元数量明显小于其他几种算法,表明其有效降低了聚类单元(OTUs)过估计问题;对应的 NID 值也小于其他几种算法,表明 NbHClust 具有更高的聚类精度,聚类性能优于其他几种算法.

为了进一步验证聚类结果,在 97%的聚类阈值下,我们基于 *F*-value 对聚类单元进行细节分析.由于每一个真实类(物种)单元可能被分为多个聚类单元,因此定义真实类的 *F*-value 为与该物种相关聚类单元的最大 *F*-value($F_{i\bullet}=\max_{j=1}^{n}F_{ij}$, i 为物种数量,j 为聚类单元数量),计算结果如图 5 所示.在 NbHClust,CD-HIT,DNAClust 和 Uclust 这 4 种算法的聚类结果中,分别有 71,42,37,34 个真实类单元的 *F*-value 大于 0.5.也就是说,在 NbHClust 的结果中,聚类单元具有更高的准确率及召回率,聚类单元的同源性与完整性更好.例如,NbHClust 聚类结果中,有 11 个真实类的 *F*-value 满足 $0.9<F_{i\bullet}\leqslant 1$,明显高于 CD-HIT(3),DNAClust(3)与 Uclust(1).从聚类单元平均 *F*-value 来看, NbHClust 算法也明显高于其他几种算法,表明 NbHClust 算法具有更高的聚类精度.



(a) 真实物种 $F_{i\bullet}$值分布 (b) 不同算法的平均 *F*-value

Fig.5 *F*-value results for the simulated dataset Gum_V6

图 5 模拟数据集 Gum_V6 的 *F*-value 结果

算法效率是衡量聚类算法的一个重要内容.为了进一步分析本文算法的有效性,我们从模拟数据集 Gum_V6 中随机挑选了 500 条序列(Gum_V6_R500)在相同的阈值(90%)下聚类,其时间统计结果见表 1.

Table 1 Time costs for the compared methods

表 1 聚类算法时间消耗(Gum_V6_R500)

算法	消耗时间(s)
NbHClust	2.1
Uclust	0.2
CD-HIT	0.2
DNAClust	0.4

从表 1 中可以看出,NbHClust 算法消耗的时间~2.1s,稍大于 Uclust 等算法(但远小于标准的层次聚类算法,如 Mothur 算法消耗时间>40s).实际上,假定序列平均长度为 L,每个序列所需的存储空间为 $O(L)$,总的聚类单元数为 C,每个聚类单元包括 k 个 δ-邻域序列,则本文算法 NbHClust 的空间复杂度为 $O(kLC)$,时间复杂度为 $O(kN)$(当 $k<<N$ 时,$O(kN)\approx O(N)$).由此可见,NbHClust 算法具有线性复杂度,保留了 CD-HIT 等低复杂度的优点,适合处理大规模测序序列数据.

3 结 论

随着下一代测序技术的发展,发展高效的计算方法分析序列数据尤为重要.基于 454 测序数据特点,本文提出了一种基于邻域种子序列的启发式序列聚类算法.由于采用启发式策略,有效降低了时间与空间复杂度,略高于 DNAClust,CD-HIT 等算法,但明显低于层次聚类算法,如 Mothur.通过邻域扩增策略引入容许测序误差因素,使得 NbHClust 具有更好的鲁棒性.最后,通过聚类单元大小属性约减小聚类单元,有效减少了虚假聚类单元数量,降低了测序误差对聚类结果的影响,提高了聚类精度.实验结果表明,NbHClust 能够较好地处理大规模序列数据聚类分析.

References:

[1] Huang X, Lü Q, Qian PD. An examplear selection algorithm for protein structure clustering. Acta Automatica Sinica, 2011,37(6): 682−692 (in Chinese with English abstract).

[2] Sun JG, Liu J, Zhao LY. Clustering algorithms research. Ruan Jian Xue Bao/Journal of Software, 2008,19(1):48−61 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/19/48.htm [doi: 10.3724/SP.J.1001.2008.00048]

[3] Sun Y, Cai Y, Mai V, Farmerie W, Yu F, Li J, Goodison S. Advanced computational algorithms for microbial community analysis using massive 16S rRNA sequence data. Nucleic Acids Research, 2010,38(22):e205. [doi: 10.1093/nar/gkq872]

[4] Cole JR, Wang Q, Cardenas E, Fish J, Chai B, Farris RJ, Kulam-Syed-Mohideen AS, McGarrell DM, Marsh T, Garrity GM, Tiedje JM. The ribosomal database project: Improved alignments and new tools for rRNA analysis. Nucleic Acid Research, 2009,37: D141−D145. [doi: 10.1093/nar/gkn879]

[5] Chen W, Zhang CK, Cheng Y, Zhang S, Zhao H. A comparison of methods for clustering 16S rRNA sequences into OTUs. PLOS ONE, 2013,8(8):e70837. [doi: 10.1371/journal.pone.0070837]

[6] Schloss PD, Westcott SL, Ryabin T, Hall JR, Hartmann M, Hollister EB, Lesniewski RA, Oakley BB, Parks DH, Robinson CJ, Sahl JW, Stres B, Thallinger GG,Van Horn DJ, Weber CF. Introducing mothur: Open-Source, platform-independent, community-supported software for describing and comparing microbial communities. Appl Environ Microbial, 2009,75(23):7537−7541. [doi: 10.1128/AEM.01541-09]

[7] Sun Y, Cai Y, Liu L, Yu F, Farrell ML, McKendree W, Farmerie W. ESPRIT: Estimating species richness using large collections of 16S rRNA pyrosequences. Nucleic Acids Research, 2009,37(10):e76. [doi: 10.1093/nar/gkp285]

[8] Wang XY, Yao J, Sun YJ, Mai V. M-Pick, a modularity-based method for OTU picking of 16S rRNA sequences. BMC Bioinformatics, 2013,14:43. [doi: 10.1186/1471-2105-14-43]

[9] Li W, Godzik A. Cd-Hit: A fast program for clustering and comparing large sets of protein or nucleotide sequences. Bioinformatics, 2006,22(13):1658−1659. [doi: 10.1093/bioinformatics/btl158]

[10] Russell DJ, Way SF, Benson AK, Sayood K. A grammar-based distance metric enables fast and accurate clustering of large sets of 16S sequences. BMC Bioinformatics, 2010,11:601. [doi: 10.1186/1471-2105-11-601]

[11] Ghodsi M, Liu B, Pop M. DNACLUST: Accurate and efficient clustering of phylogenetic marker genes. BMC Bioinformatics, 2011,12:271. [doi: 10.1186/1471-2105-12-271]

[12] Edgar RC. Search and clustering orders of magnitude faster than BLAST. Bioinformatics, 2010,26(19):2460−2461. [doi: 10.1093/bioinformatics/btq461]

[13] Lü ZM, Xu YT, Wu CW, Fan ZJ, Zhang JS. Genetic variation in different populations of ilisha elongate in China coastal water based on 16S rRNA gene analysis. Journal of Fishery Science of China, 2010,3(17):463−469 (in Chinese with English abstract).

[14] Zhuang L, Xie QY, Ling HP, Hong K. Selectively isolated deep-sea streptomyces and analysised 16S rRNA phylogenetic tree of acivity strains. Biotechnology Bulletin, 2009,(z1):398−401 (in Chinese with English abstract). [doi: 10.1093/nar/gkn879]

[15] Huse SM, Huber JA, Morrison HG, Sogin ML, Welch DM. Accuracy and quality of massively parallel DNA pyrosequencing. Genome Biology, 2007,8(7):R143. [doi: 10.1186/gb-2007-8-7-r143]

[16] van Rijsbergen CJ. Information retrieval [Ph.D. Thesis]. Boston: University of Glasgo, 1979.

[17] Schloss PD. The effects of alignment quality, distance calculation method, sequence filtering, and region on the analysis of 16S rRNA gene-based studies. Plos computational Biology, 2010,6(7):e1000844. [doi: 10.1371/journal.pcbi.1000844]

[18] Turnbaugh PJ, Hamady M, Yatsunenko T, Cantarel BL, Duncan A, Ley RE, Sogin ML, Jones WJ, Roe BA, Affourtit JP, Egholm M, Henrissat B, Heath AC, Knight R, Gordon JI. A core gutmicrobiome in obese and lean twins. Nature, 2009,457(7228):480−484. [doi: 10.1038/nature07540]

附中文参考文献:

[1] 黄旭,吕强,钱培德.一种用于蛋白质结构聚类的聚类中心选择算法.自动化学报,2011,37(6):682−692.

[2] 孙吉贵,刘杰,赵连宇.聚类算法研究.软件学报,2008,19(1):48−61. http://www.jos.org.cn/1000-9825/19/48.htm [doi: 10.3724/SP.J.1001.2008.00048]

[13] 吕振明,许逸天,吴常文,樊甄姣,张建设.中国沿海鳗不同地理群体 16S rRNA 基因的遗传变异分析.中国水产科学,2010,3(17):463−469.

[14] 庄令,谢晴宜,林海鹏,洪葵.深海链霉菌选择性分离及活性菌株 16S rRNA 聚类分析.生物技术通报,2009,(z1):398−401.

陈伟(1984－),男,湖北黄冈人,博士,主要研究领域为模式识别,数据挖掘,信息融合.
E-mail: chenwei903@gmail.com

张绍武(1964－),男,博士,教授,博士生导师,主要研究领域为模式识别理论方法及应用,机器学习,复杂网络,计算生物学.
E-mail: zhangsw@nwpu.edu

程咏梅(1960－),女,博士,教授,博士生导师,主要研究领域为信息融合,证据推理,动态系统建模,组合导航和相对导航中的应用.
E-mail: chengym@nwpu.edu.cn

潘泉(1961－),男,博士,教授,博士生导师,主要研究领域为估计辨识和信息融合理论及应用,无人机导航,避撞及对地探测技术,计算生物学.
E-mail: quanpan@nwpu.edu.cn