从 DNA 序列到系统发育多样性

进化树的建立和进化相关分析简介

张金龙

嘉道理农场暨植物园

2019 年 9 月 16 日

内容提要

- 进化树相关的一些基本概念
- 建立进化树的基本流程
- 在 R 中进行系统发育多样性分析

目录

目录

什么是进化树?

进化树是表示类群之间系统发育关系的树状结构，是进行系统发育研究（谱系）的基础。

- cladogram: 体现分类单元甚至个体的**等级结构**，不一定能体现系统发育关系。
- phylogram: 体现分类单元甚至个体的**系统发育关系**，以精确的枝长反映类群间分化距离。
- chronogram: 体现分类单元甚至个体的**系统发育关系**，以枝长反映类群间分化的分化时间。

进化树研究的历史

- 1837 年，达尔文提出了符合进化思想的树状图。
- 十九世纪末，德国人海克尔已经对物种之间的亲缘关系，建立了基本的进化树。
- 1964 年 Cavalli-Sforza 和 Edwards 引入了简约法 (parsimony) 和似然法 (likelihood)。
- 1966 年 Hennig 提出了分支系统学理论 (theory of cladistics)。
- 1977 年 Fitch 将简约法应用到 DNA 序列重建系统发育关系中。
- 1978 年 Felsenstein 最早在计算机中实现了最大似然法，并开发出了 PHYLIP 软件。
- 1996 年 Rannala 和杨子恒提出了贝叶斯建树的方法。

进化树: 树状图

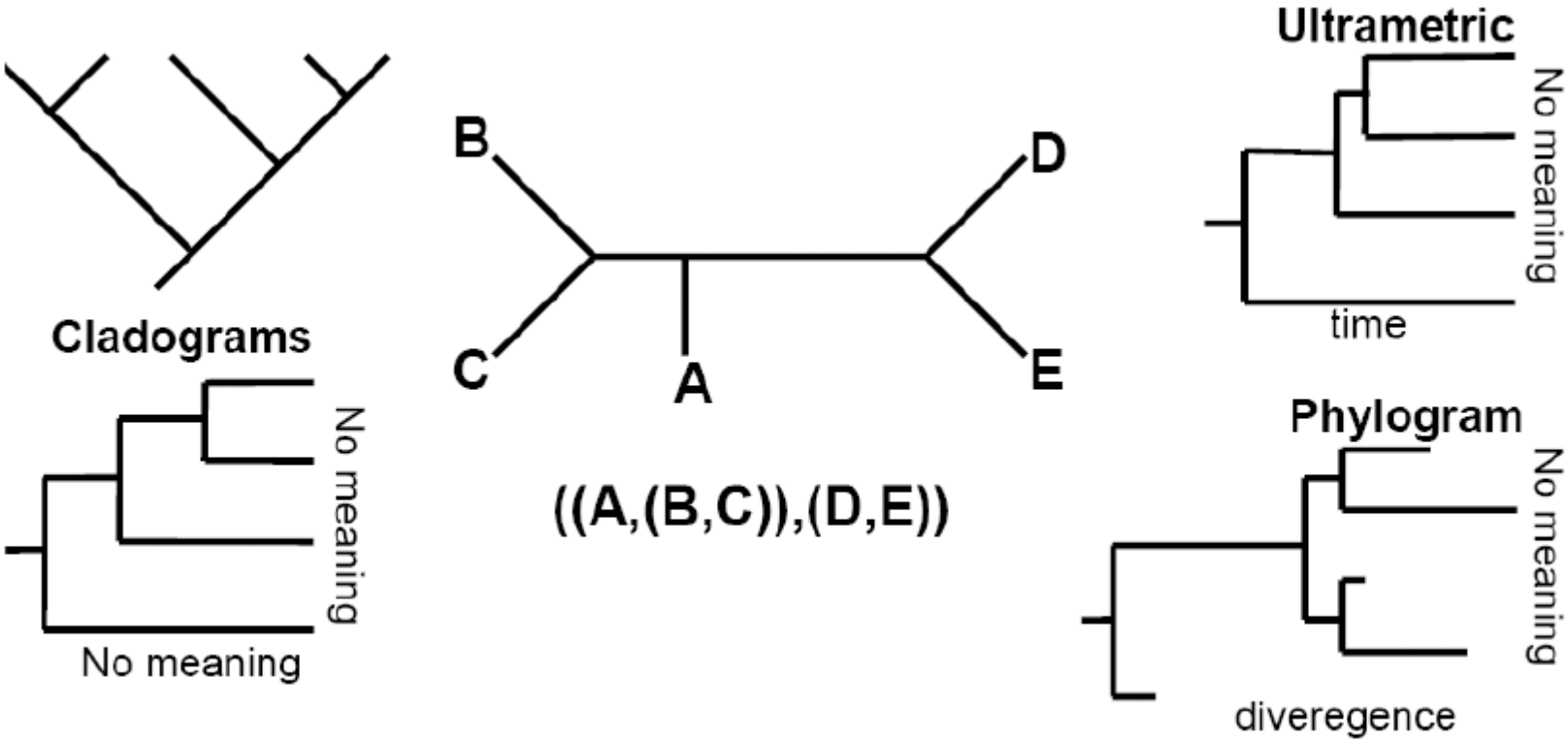
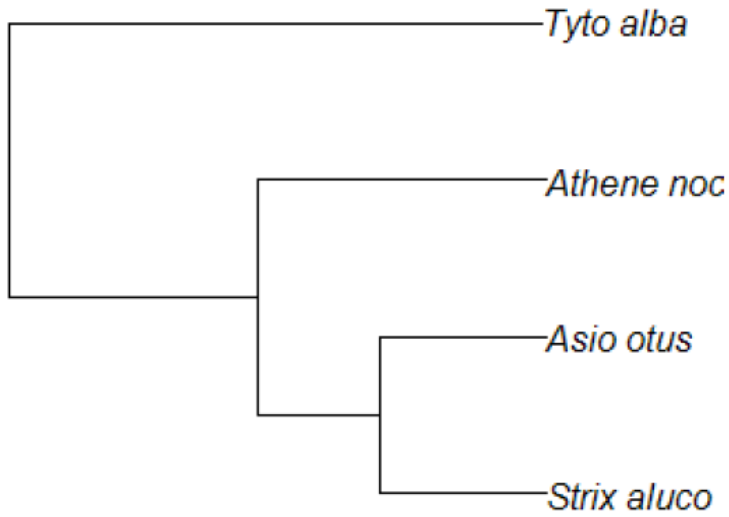


图 1: 各种进化树: 一般从左到右，只计横向枝长。纵向枝长只是为了图形的美观，其长度不计。末端节点（tip）一般为物种名称或者个体名称，内部节点 (internal nodes) 表示分类单元之间的联系

Newick 格式




```
(((Strix_aluco:4.2, Asio_otus:4.2):3.1, Athene_noctua:7.3):
6.3,Tyto_alba:13.5);
```

图 2: Newick 格式，保存在纯文本文件中：用小括号表示拓扑关系，物种之间用逗号间隔开，物种名后往往添加冒号，冒号后为枝长。冒号前面也可有数字，表示是 bootstrap 支持率或后验概率等。内部节点可没有名称，而仅有该节点下的枝长。

NEXUS 格式

```
#NEXUS

begin taxa;
dimensions ntax=4;
taxlabels
Strix_aluco
Asio_otus
Athene_noctua
Tyto_alba ;
end;

begin trees;
tree tree_1 = (((Strix_aluco:4.2,Asio_otus:4.2):3.1,
Athene_noctua:7.3):6.3,Tyto_alba:13.5);
end;
```

扩展的 Nexus 格式、xml 格式和 Json 格式

- nexus 格式可以进一步包括建树用的 DNA 序列、定年用的 r8s 软件、建树用的 Paup* 命名模块、显示 Figtree 的模块，MrBayes 模块等。
- xml 格式，是一种类似于 html 的标记语言，用于保存 BEAST 软件等建树用的数据和相应命令。
- Json 格式 (JavaScript Object Notation)，是基于 JavaScript 脚本的一种语言，用于保存 DNA 序列和某些软件的命令。

余光创博士编写的 ggtree 程序包提供了多种格式的读取和简单处理。

进化树的编辑和打印

以下软件，提供了编辑 Newick 文件或者 Nexus 文件的功能。

- Mesquite：用 Java 开发，有众多插件，能进行 DNA 序列、进化树、性状的编辑并多种系统发育比较分析。
- FigTree：用 JAVA 编写，无需安装，对 Newick 和 Nexus 支持很好，参数较多，容易调整，跨平台。
- TreeView：早期查看进化树的经典软件，但是参数不好控制。
- MEGA：一般针对该软件自己生成的进化树绘图等，支持 Newick 树的可视化编辑。
- Dendroscope：对 Newick 格式和 Nexus 支持较好，类似 FigTree，同时支持 Singletons。
- iTOL：网页版，编辑进化树

进化树出图用的 R 包

- ape: 由 E. Paradis 等编写，其内置的 plot.phylo 函数，功能灵活，拥有绘制进化树的多种参数。
- ggtree: 由余光创博士开发，支持 ggplot2 的语法。支持多种数据格式的读写。

建立进化树的前提: 基因的同源性

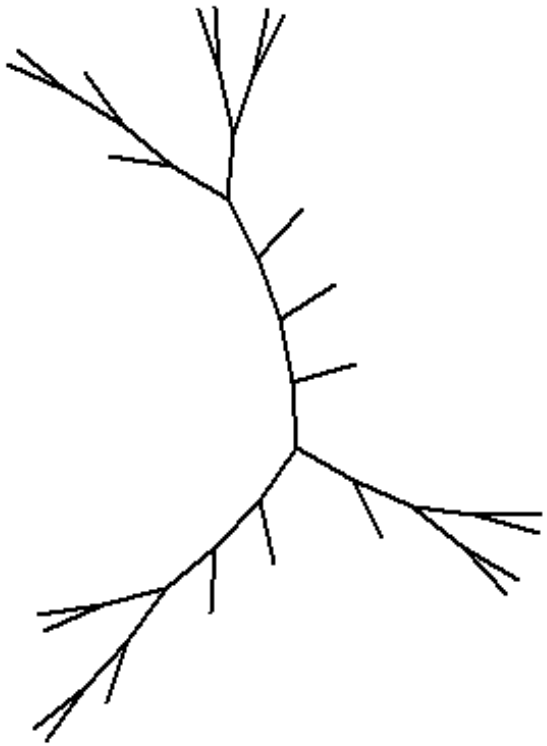
- ortholog: 这个基因出现在所有的后代物种中，由共同祖先的同一个基因演变而来。
- paralog: 该基因出现在一个体内，由共同祖先基因扩增而来
- xenolog: 基因转座

进化树推断中，必须要用 orthological 同源基因

等距树、有根树和外类群及其选择

- 等距树 (Ultrametric Tree)：进化树中，每一个末端分类单元都与根节点的距离相等.
- 有根树 (rooted tree) 和无根树 (unrooted tree)：为了探讨一个类群内部的亲缘关系，需要在进化树中指定一个或几个独立于所研究类群之外的分类单元作为外类群 (outgroup)。指定了外类群的树，称为有根树，否则称为无根树。有根树能够表示类群内部的亲缘关系，而无根树只能显示拓扑结构。
- 外类群: 通常选择研究类群的近缘的单系类群作为外类群，不能位于研究类群中，也不能距离所研究的类群太远。前者无法体现类群的进化关系，后者在序列比对，或者进化树推断时会造成结果不准确。

无根树 Unrooted Tree



从哪里开始？到哪里结束？

图 3: 无根树。在建树时没有指定外类群，则进化树为无根树，在后续操作时，首先应转换为有根树 (rooted tree)

外类群的指定

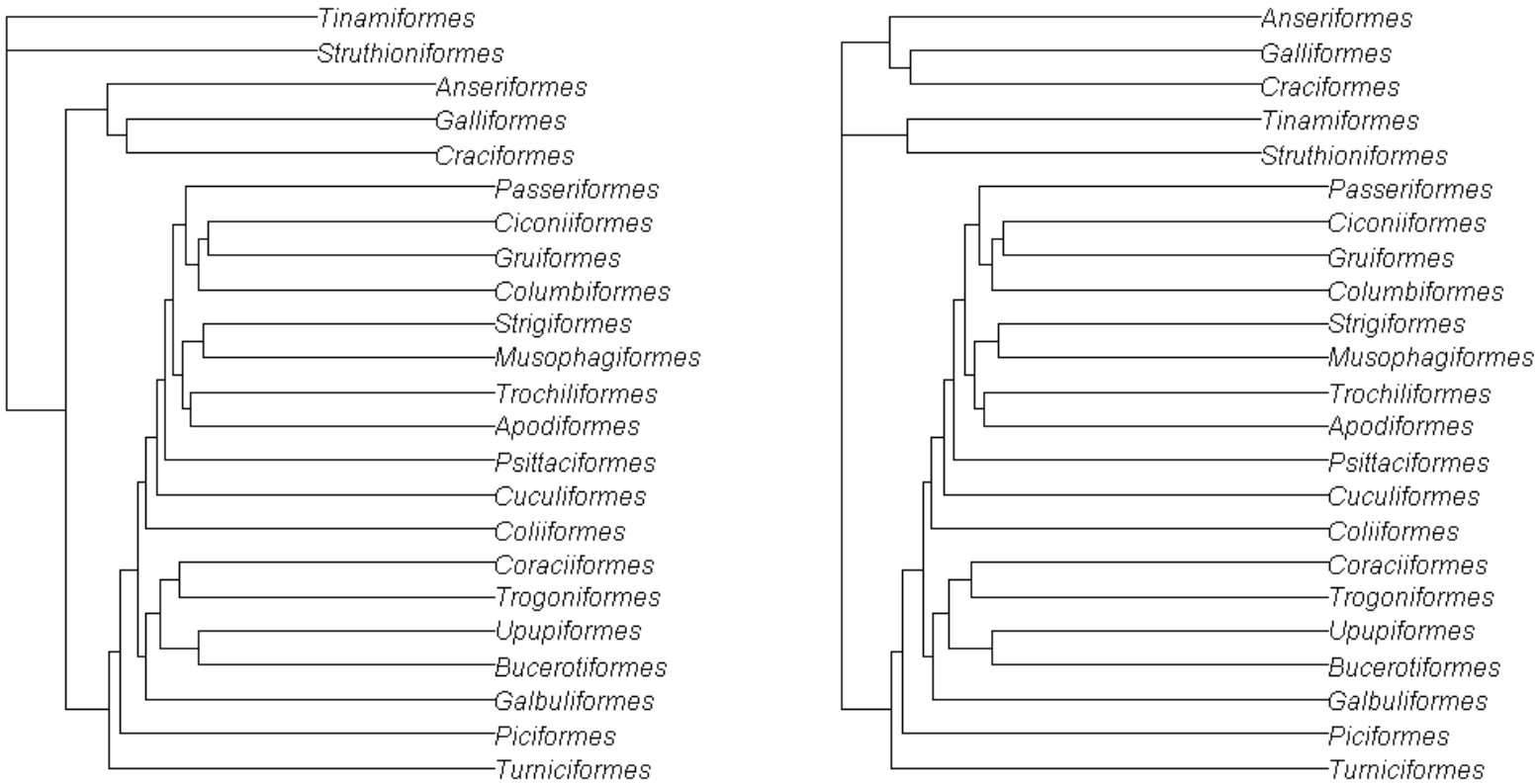


图 4: 外类群的数量，可以根据研究需要进行指定。用 Figtree 和 ape 指定外类群十分方便。

一致性树 Consensus Tree

- 当获得了多个拓扑结构时，需要对拓扑结构的稳定性，即进化树内部的各节点的可靠性进行判断。
- 对进化树内部的节点，进行可靠性检验后，常常会生成大量进化树，需要对结果进行汇总。
- 按照节点一致性原则，选出各进化树都支持的某节点的拓扑结构，并计算支持该节点的百分率，就称为支持率。
- 支持率低的节点，有时候选择合并，这样就形成了多分枝结构 (Polytomies)

多分枝结构

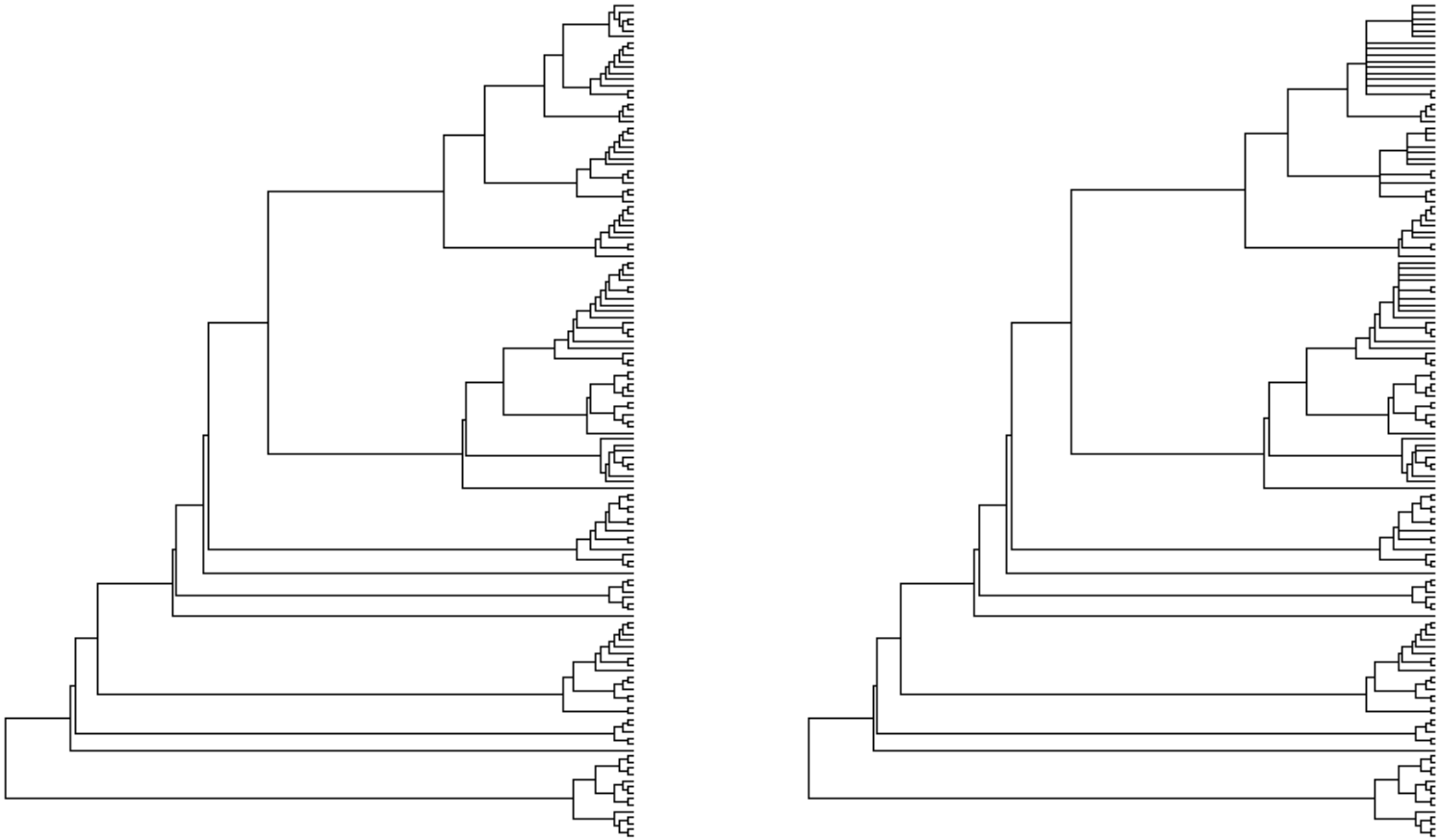


图 5: 多分枝结构

多分枝结构的成因

若某一个节点支持率很低，表明现有数据不能将该节点下的各分类单元分开
多分枝结构的成因:

- 数据缺失，或所选的建树基因变异很低，无法体现足够的系统发育信息。
- 物种的快速辐射进化，如环境对表型的强烈筛选，近缘种在短时间内形成了多个物种。这些亲缘种之间，用某一基因常常难以区分。

二叉树和孤立节点

- 一般情况下，每个节点下，应该直接分成两个姊妹节点，这种进化树称为二歧树（divaricate），也称二叉树。
- 若一个节点下，只有一个节点，则称其为孤立节点（Singleton）。
- 群落系统发育分析的 Phylomatic 软件在建立进化树时，会生成很多多分枝结构和孤立节点。Phylocom 软件则能处理含有孤立节点的 newick 进化树
- 若进化树含有多分枝结构，而数据分析软件要求输入二叉树，则可使用 ape 包的 multi2di 函数转换，为多分枝结构的节点，随机添加一个很小的枝长，从而转换为二叉树。

FASTA 文件

- FASTA 文件为纯文本格式。
- 物种的 DNA 序列常以该文件格式保存。
- 每条序列以 $>$ 开头。
- 换行后为该条核苷酸序列'ATCG'
- FASTA 文件也可以保存比对后的序列。
- FASTA 数据与换行 (interleaved) 和不换行的区别。

FASTA 文件示意图

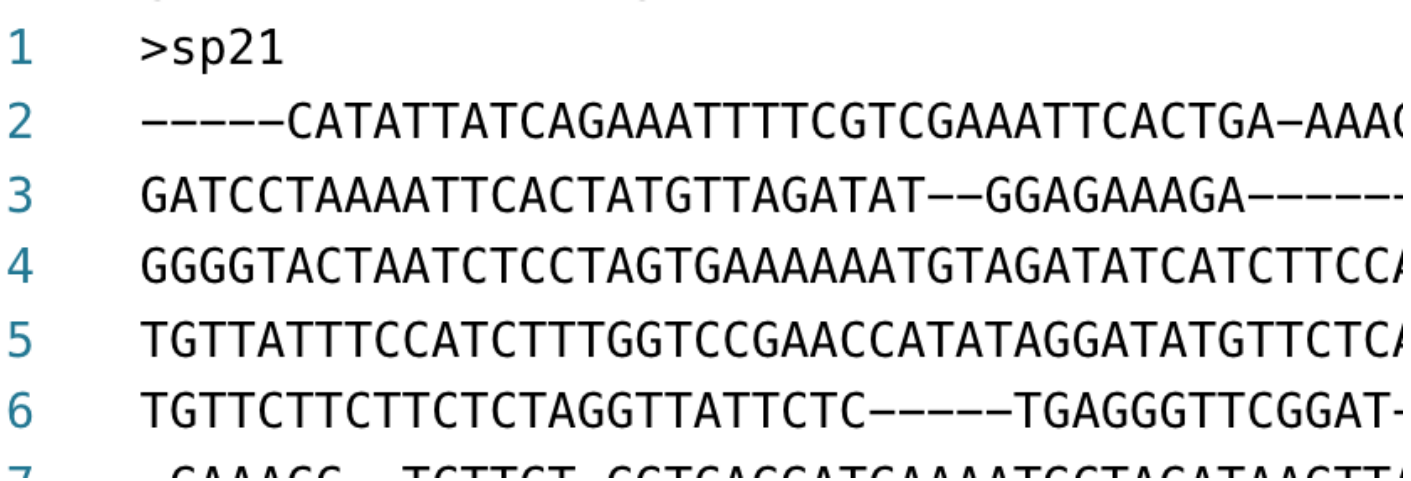


图 6: FASTA 文件

DNA 序列的获取

DNA 数据，按照来源可分成三类：

- 测序数据，包含桑格（一代）测序，高通量（二代、三代）测序，单分子测序等。
- 从数据库下载，GenBank、EBI、DDBJ 等综合数据库，以及各类专业数据库。
- 模拟数据：用软件按照一定规则，随机生成的数据

DNA 条形码，就是一种快速获取 DNA 序列的方法。

DNA 条形码建立进化树的基本技术流程

- 采样：注意一定要有凭证标本
- 总 DNA 提取和琼脂糖电泳检测
- 目的序列的 PCR 和琼脂糖电泳检测
- 送公司测序，获得 ab1 峰图文件和 fasta 文件，测序质量检查 (ChromasPro 软件等)，正反测序结果合并和校对（如 DNAman 软件，ContigExpress 软件，VectorNTI 等)
- 通过和 BLAST 数据库比对，检查是否存在污染和测序错误，确认和保存为 Fasta 文件
- 用 MUSCLE 比对，进一步检查，在 BioEdit 或者 AliView 中检查
- 每个基因，分别建树检查各分类单元的系统发育关系。用多个基因，建立 Supermatrix 检查

序列比对

目的：将具有变异信息的位点对齐

- ClustalX：图形用户界面，但是比对速度较慢 (过去主流)
- MUSCLE：命令行，适用于较大大规模数据 (当前主流)
- Mafft：命令行，能处理 30000 条以内的 DNA 序列比对（当前主流）
- PRANK：DNA 和氨基酸序列的比对（当前主流）
- TranslatorX, MASCE：主要用于编码区的比对

比对前的 DNA 序列

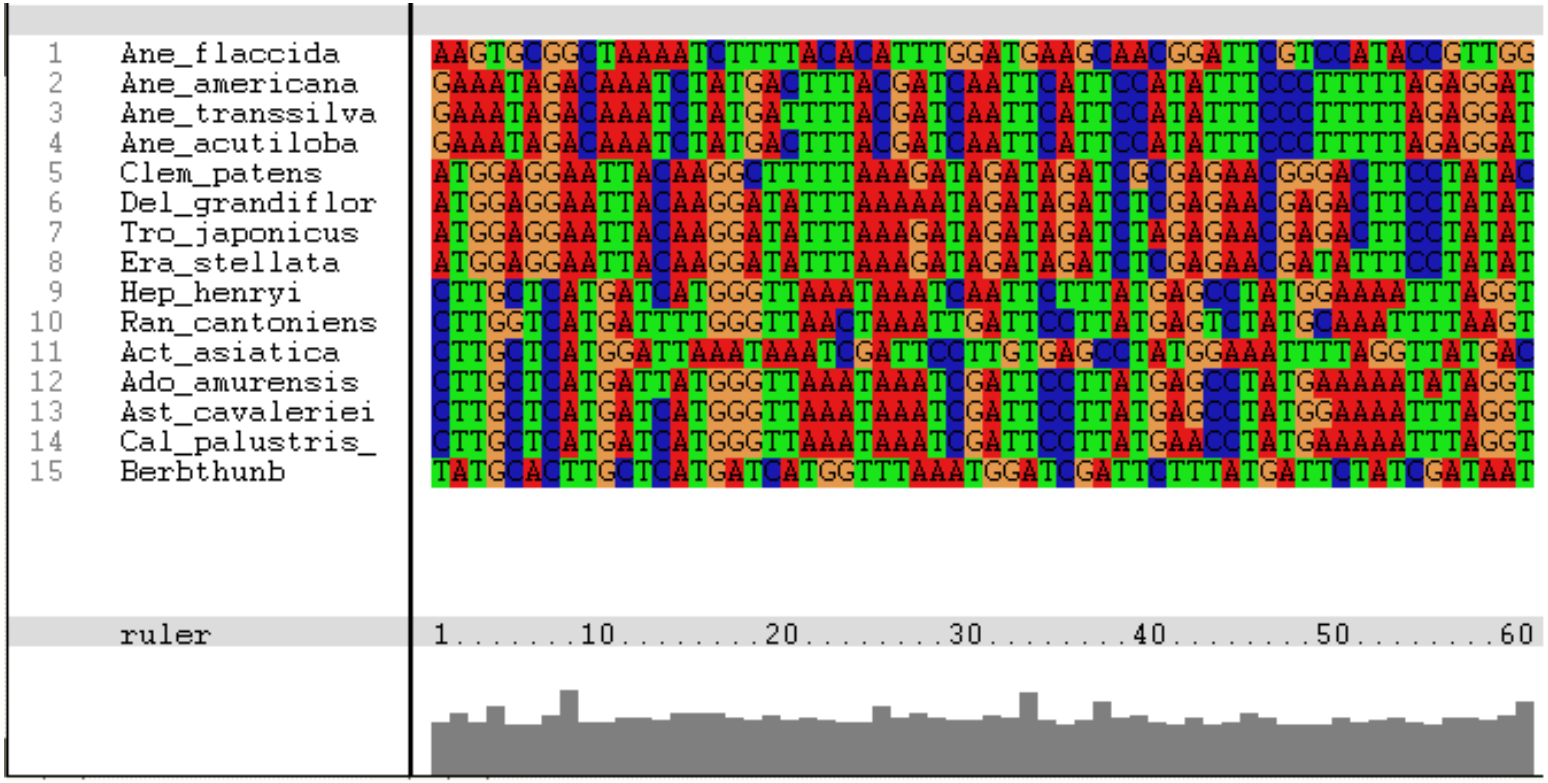


图 7: 比对前的序列

比对后的 DNA 序列

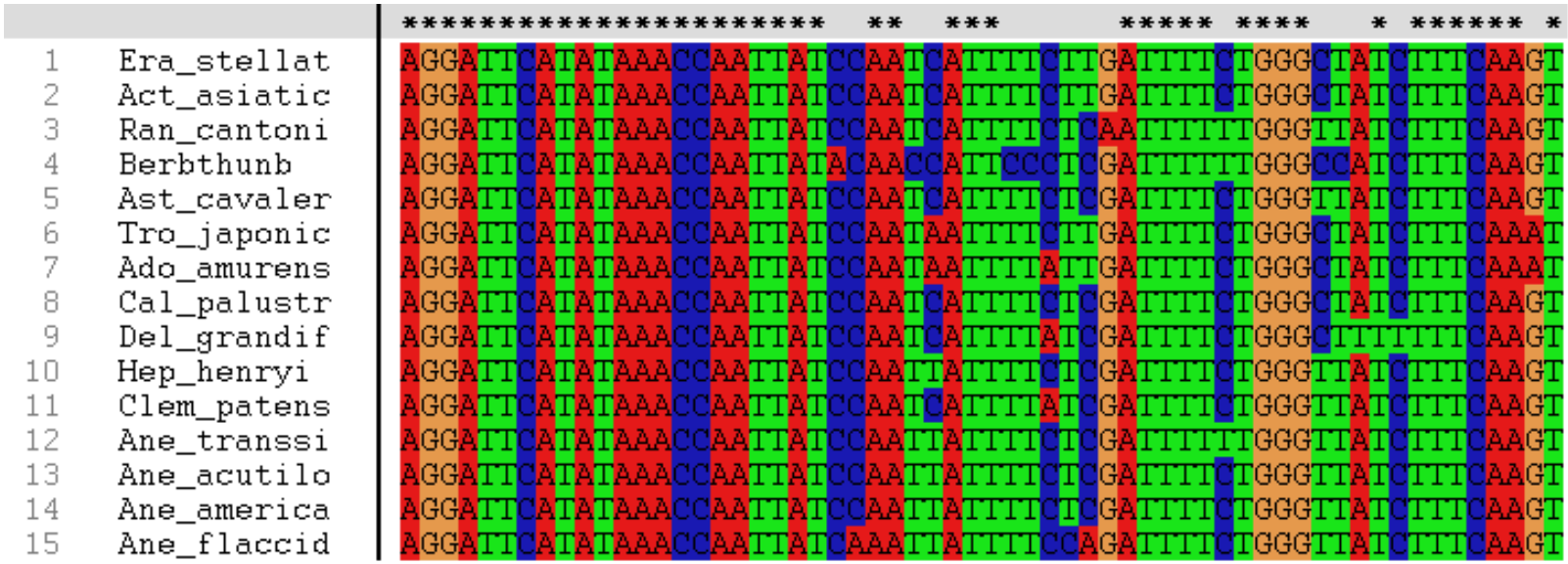


图 8: 比对后的序列

比对后的手工校正

测序的错误，可能在比对后才能体现出来，因此需要核对测序的荧光染料强度峰图，进行人工判读和修改。
对于编码基因，三联体密码子，缺失的位置和长度必须是 3 的倍数
软件:

- AliView 较新，跨平台
- BioEdit 传统，仅 Windows
- Geneious 功能强大的集成分析软件

BioEdit

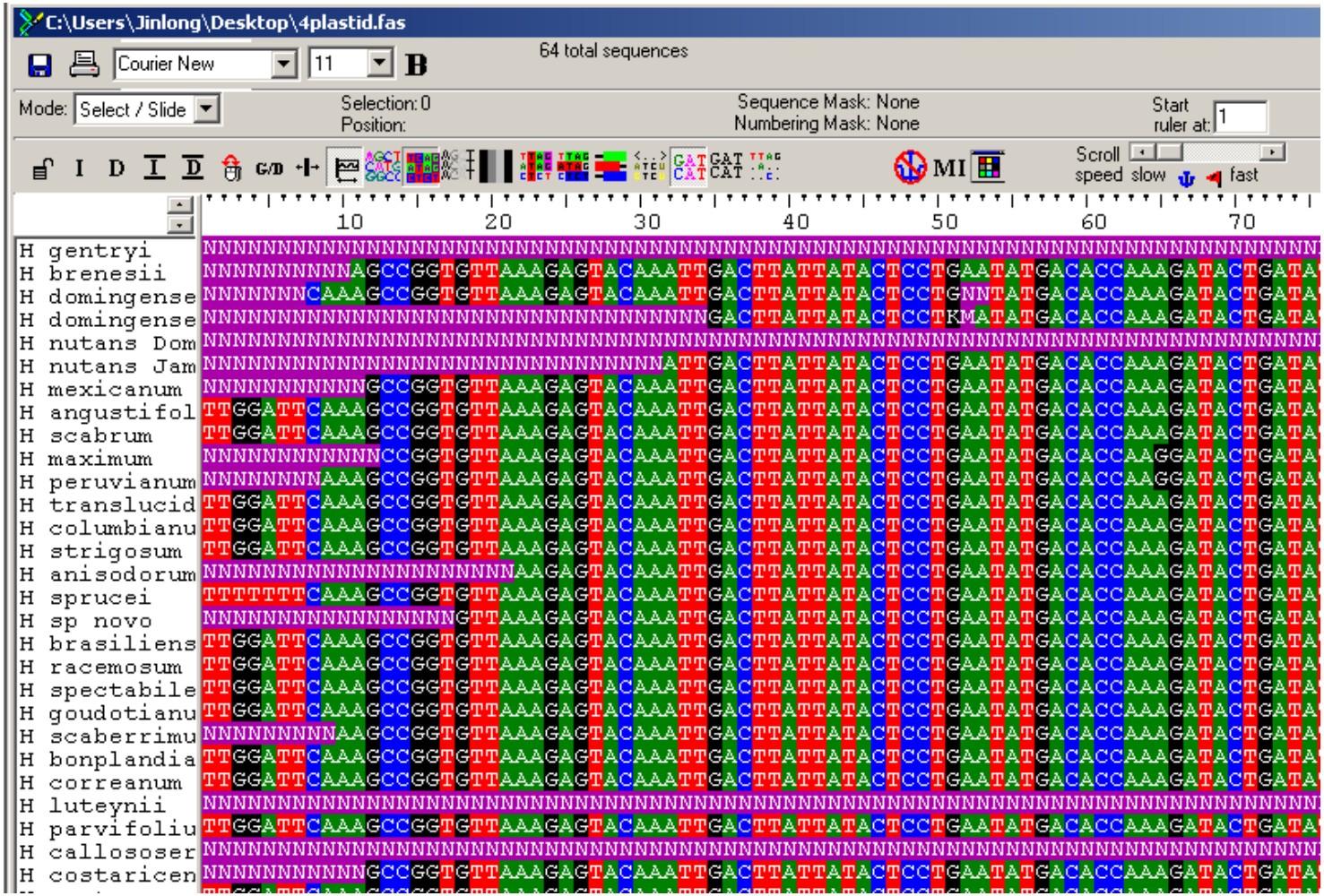


图 9: 用 Bioedit 编辑序列

目录

进化模型

A A→ A A → G G→ T

$$Prob[k\,events] = \frac{(\mu t)^k e^{-\mu t}}{k!}$$

符合统计学中的泊松分布 (Poisson Distribution)
μ 为碱基替换率
R_{xy} 为位点从 x, 转变为 y, 的概率。

$$Prob(t) = \sum_{k=0}^{\infty} (R^k) \frac{(\mu t)^k e^{-\mu t}}{k!}$$

$$Prob(t) = \sum_{k=0}^{\infty} \frac{(R-1)^k (\mu t)^k}{k!} = \sum_{k=0}^{\infty} \frac{(Q\mu t)^k}{k!} = e^{Q\mu t}$$

进化模型

$$Q = \begin{bmatrix} - & r_{AC} & r_{AG} & r_{AT} \\ r_{CA} & - & r_{CG} & r_{CT} \\ r_{GA} & r_{GC} & - & r_{GT} \\ r_{TA} & r_{TC} & r_{TG} & - \end{bmatrix}$$

行表示初始状态，列表示转变后的状态。

$$q_{ii} = -\sum_{j=0, j\neq i}^{4} q_{ij}$$

Q 为转移概率矩阵

转移概率矩阵模型 JC69 模型

由 Jukes and Cantor (1969) 提出。
转移概率矩阵模型

$$Q = \begin{bmatrix} -\mu(a\pi_C + b\pi_G + c\pi_T) & a\mu\pi_C & b\mu\pi_G & c\mu\pi_T \\ g\mu\pi_A & -\mu(g\pi_A + d\pi_G + e\pi_T) & d\mu\pi_G & e\mu\pi_T \\ h\mu\pi_A & i\mu\pi_C & -\mu(h\pi_A + j\pi_C + f\pi_T) & f\mu\pi_T \\ j\mu\pi_A & k\mu\pi_C & l\mu\pi_G & -\mu(i\pi_A + k\pi_C + l\pi_G) \end{bmatrix}$$

JC69 模型

$$Q = \begin{bmatrix} - & 1 & 1 & 1 \\ 1 & - & 1 & 1 \\ 1 & 1 & - & 1 \\ 1 & 1 & 1 & - \end{bmatrix}$$

参考：?dist.dna

转移概率矩阵模型 Kimura 81 和 F81

Kimura 81: 考虑 transition 或 transversion
例如由 G 变成 A 称为 transition
A 转变成 T，或 T 变成 A，称为 transerversion。

$$Q = \begin{bmatrix} - & 1 & \kappa & 1 \\ 1 & - & 1 & \kappa \\ \kappa & 1 & - & 1 \\ 1 & \kappa & 1 & - \end{bmatrix}$$

F81 (Felsenstein, 1981)

$$Q = \begin{bmatrix} - & \pi_C & \pi_G & \pi_T \\ \pi_A & - & \pi_G & \pi_T \\ \pi_A & \pi_C & - & \pi_T \\ \pi_A & \pi_C & \pi_G & - \end{bmatrix}$$

π_i is the equilibrium frequency of nucleotide i

转移概率矩阵模型 F84

F84 (Felsenstein, 1984)

$$Q = \begin{bmatrix} - & \pi_C & (1+\kappa/\pi_R)\pi_G & \pi_T \\ \pi_A & - & \pi_G & (1+\kappa/\pi_Y)\pi_T \\ (1+\kappa/\pi_R)\pi_A & \pi_C & - & \pi_T \\ \pi_A & (1+\kappa/\pi_Y)\pi_C & \pi_G & - \end{bmatrix}$$

π_i : the equalibrium frequency of nucleotide i

$\kappa = \alpha/\beta$

α : transition matrix

β : the transversion rate

更多转移概率矩阵模型二

HKY85 (Hasegawa-Kishino-Yano, 1985)

$$Q = \begin{bmatrix} - & \pi_C & \kappa\pi_G & \pi_T \\ \pi_A & - & \pi_G & \kappa\pi_T \\ \kappa\pi_A & \pi_C & - & \pi_T \\ \pi_A & \kappa\pi_C & \pi_G & - \end{bmatrix}$$

Tamura-Nei, 1993

$$Q = \begin{bmatrix} - & \pi_C & \kappa_2\pi_G & \pi_T \\ \pi_A & - & \pi_G & \kappa_1\pi_T \\ \kappa_2\pi_A & \pi_C & - & \pi_T \\ \pi_A & \kappa_1\pi_C & \pi_G & - \end{bmatrix}$$

其中 $\kappa_1 = \alpha_Y/\beta, \quad \kappa_2 = \alpha_R/\beta$

α_R： the purine transition rate α_Y ： the pyrimidine transition rate β ： the transversion rate

统一模型 GTR

General Time Reversible Model

$$Q = \begin{bmatrix} - & \alpha\pi_C & \beta\pi_G & \gamma\pi_T \\ \alpha\pi_A & - & \delta\pi_G & \epsilon\pi_T \\ \beta\pi_A & \delta\pi_C & - & \eta\pi_T \\ \gamma\pi_A & \epsilon\pi_C & \eta\pi_G & - \end{bmatrix}$$

$\alpha...\eta$：从一种碱基变换为另一种碱基的速率 π 碱基频率将参数 $\alpha...\eta$ 做一定的限制，GTR 模型可以简化为以上任何一种转移概率模型

模型之间的关系

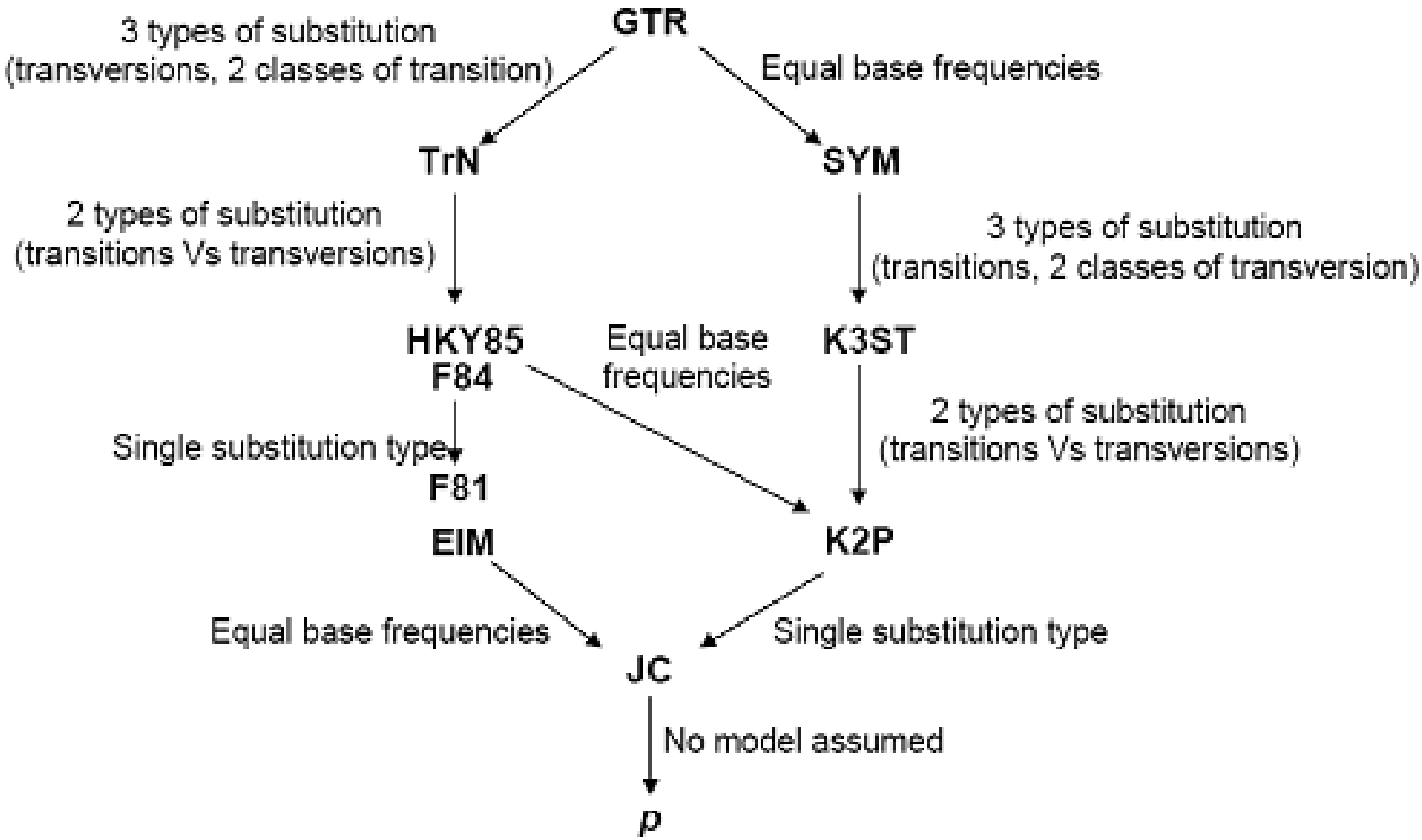


图 10: DNA 碱基替换模型间的关系

碱基位点变异速率的差异

密码子的第三个碱基比前两个碱基变异更快；
不同区域的功能不同，变异速率存在差异。
为此，人们引入 Gamma 分布拟合不同位点进化速率的差异.

$$\Gamma(a) = \int_0^\infty x^a e^{-x} \frac{dx}{x}, \ a > 0$$

碱基替换模型的筛选

什么样的模型是合适的?
对于巢式模型（nested models）的筛选，采用似然比检验 (Hierarchical Likelihood Ratio Test)
即：求每个进化树的最大似然值，即给定进化模型 M，模型的各参数 θ 进化树拓扑结构 τ 以及枝长 ν 当前序列比对格局出现的概率。

$$L = P(D \mid M, \theta, \tau, \nu)$$

为了使该序列比对格局出现的可能性最大，对 Likelihood function 取最大值

$$\hat{\theta}, \hat{\tau}, \hat{\nu} = \max L(\theta, \tau, \nu)$$

碱基替换模型的筛选

取对数

$$\ell = \ln P(D|M, \hat{\theta}, \hat{\tau}, \hat{\mu})$$

$$LRT = 2(\ell_1 - \ell_2)$$

LRT(*LogLikelihoodRatio*)，符合 χ^2 分布。因此可检验模型复杂程度的变化是否对模型的精度有显著的影响。
赤池信息量 AIC(Akaike Information Criterion)
一般作为模型的准确性 (Likelihood 部分) 和参数数量 (k) 之间的权衡。

$$AIC = -2\ell + 2k$$

软件：ModelTest、jModelTest、PartitionFinder、IQ-TREE 等

目录

构建进化树的主要方法 I

碱基概率转移模型保证我们获得建立进化树所需的枝长，但是如何获得进化树的拓扑结构?

- 1. 距离法 (Distance based method)：根据序列两两之间的进化距离，进行聚类
- 2. 简约法 (Most Parsimony)：根据序列之间变化的关系，找到一棵树，使该进化树上所发生的进化事件（碱基替换发生的次数）最少。

构建进化树的主要方法 II

- 3. 极大似然法 (Maximum Likelihood)：在所有拓扑结构中找出最有可能形成当前序列比对格局的拓扑结构。
- 4. 贝叶斯法 (Bayesian)：从一系列可能取值中，随机给出参数的初始值，让进化树的参数在参数空间按照后验概率密度移动，生成成千上万棵进化树，对这些进化树进行信息汇总，从而得到进化树拓扑结构和枝长分布的估计。

进化树建立：遗传距离法

第一步：计算序列之间的遗传距离矩阵
遗传距离可以基于 RAW，即碱基差异的数量，或者 Transition，transversion，也可以基于各种转移概率模型。计算可参考 ape 软件的 dist.dna()
第二步：基于距离矩阵进行聚类

- UPGMA 法 (Unweighted Pair Group Method with Arithmetric Mean)
- NJ 法: Neighbour Joining
- ME 法: 最少进化法 Minimum Evolution

目前，由 NJ 法或 ME 法创建的进化树，一般只用于构建极大似然进化树的初始树，极少能再作为论文的最终结果。

进化树建立：最大简约法

进化速率低的序列，回复突变很少，此时可以用最大简约法。

简约原理

真实的进化历史的经历的进化事件最少。

优点：

- 结果容易解释
- 计算速度快
- 进化树结构良好

什么时候用简约法建树?

- 作为其它方法的初始进化树
- 基因家族进化历史的推断

PAUP* 适用于建立最大简约树

进化树建立：最大似然法

最大似然估计一种参数估计的方法。按照当前的概率，数据所出现概率的乘积称为似然 (Likelihood)。

$$L(H) = \prod Prob(D|H) = Prob(D_1|H)Prob(D_2|H)\cdots Prob(D_n|H)$$

$$Prob[D_j|\tau, M, \rho_j,],\ j = 1, 2, ...n$$

$$L(\tau, M, \rho|D) \equiv Prob[D|\tau, M, \rho] = \prod_{j=1}^{I} Prob[D_j|\tau, M, \rho_j]$$

因此，Likelihood 为联合概率密度，当所有发生某一事件的概率密度（假设的概率密度）相乘，其结果取最大值时，就发生了当前的事件，这就是极大似然的思想。

应用于进化树推断时，问题变为：**假设已经有一棵进化树 T 以及碱基替换模型 Q，则获得当前 DNA 比对格局的可能性有多大?**

进化树建立：最大似然法

$$L(T, Q) = Prob(D|T, Q)$$

一些常用假设：

- 不同位点的进化彼此独立
- 不同分支的进化彼此独立
- 各位点的进化速率相同

$$L(\tau, M, \rho|D) \equiv Prob[D|\tau, M, \rho] = \prod_{j=1}^{l} Prob[D_j|\tau, M, \rho_j)$$

给定进化树，比对格局的总体似然值无法直接用表达式计算，进化树内部的每个节点与两个子代之间似然函数的关系为：

$$L_j^i(s) = \left[\sum_{x\in\{A,C,G,T\}} P_{sx}(d_{o1}) L_j^{o1}(x)\right] \cdot \left[\sum_{x\in\{A,C,G,T\}} P_{sx}(d_{o2}) L_j^{o2}(x)\right]$$

进化树建立：计算 Likelihood

对于末端节点, 如果 $s = s_i^j$, 则

$$L_j^i(s) = 1$$

否则

$$L_j^i(s) = 0$$

$$Prob[D_j, \tau, Q, 1] = \sum_{s \in A,C,G,T} \pi_s L_j^{2n-2}(s)$$

此时整个碱基比对格局的似然值为：

$$log[L(\tau, M, 1)] = log\left[\prod_{j=1}^{l} Prob[D_j, \tau, Q, 1]\right] = \sum_{j=1}^{l} log\left[Prob[D_j, \tau, Q, 1]\right]$$

计算机是如何计算极大似然法进化树的?

- 对一系列参数给出随机值或基于简约法计算出的数值，作为初始值。
- 每次对之前所得的参数数值做一定程度的改变，每次计算 LogLikelihood，直到 LogLikelihood 不再改变为止。
- 所得参数就是进化树的极大似然估计。

哪些参数需要改变?

- 1. 进化树的拓扑结构和枝长
- 2. 碱基替换之间的转移概率矩阵 Q
- 3. 不同区段的碱基替换频率的差异 (Gamma distribution)

目录

进化树的数量

需要在多少种拓扑结构中搜寻?
答：随着进化树中物种数的增加，进化树的拓扑结构数量迅速增加。
每新增加一个分类单元，都能够在之前的 2n-3 个位置放置进化树
进化树的数量，因此，进化树的数量与分类单元的数量的关系为:

$$t_n = \frac{(2n-5)!}{2^{n-3}(n-3)!} = \prod_{i=1}^{n}(2i-5)$$

进化树的数量

3: 1
4: 3
5: 15
6: 105
7: 945
8: 10,395
9: 135,135
10: 2,027,025
11: 34,459,425
12: 654,729,075
13: 13,749,310,575
14: 316,234,143,225
15: 7,905,853,580,625

进化树的数量

16: 213,458,046,676,875
17: 6,190,283,353,629,374
18: 191,898,783,962,510,624
19: 6,332,659,870,762,850,304
20: 2.21643095476 7e+20
30: 8.687364e+36
50: 2.838063e+74

Bang!!!

通常，在所有的拓扑结构中搜索是不可能的

因此在分类单元数较多时，需要用到启发式搜索 (heuristic search).

启发式搜索: 物种逐步添加

以一棵随机的、只包含三个种的进化树开始，在剩余数据中，随机找出一个物种，加入到进化树中。随着加入位置的不同，所得的比对格局的 Likelihood 不同，取能够使 Likelihood 最高的加入位置，继续加入剩余的节点，直到所有物种加入到进化树中。这种添加物种的方式，称为**物种逐步添加 (stepwise addition)**，是一种启发式搜索。

减少局域最大似然的影响

但是对于 n 个物种的进化树来说，我们只计算了 $(n-2)^2$ 个进化树，所得的极大似然值，并不一定为所有可能的进化树中最大的，因此，结果很可能为**局域最大似然值（regional maximum likelihood）**。为了尽量减少局域最大似然的影响，生物信息学家采用以下方法：

- 1. 多次从随机的进化树开始，随机添加物种
- 2. 进化树的剪接和重排
- 3. 对于接近最优树的进化树进行重排，以便寻找全局最优树

进化树的重排

进化树重排，是对现有进化树的分支结构进行剪接后，再按照一定规则连接起来的一系列方法，包括

- NNI: Nearest neibourgh interchange $2(n-3)$
- SPR: sub-tree pruning and regrafting $4(n-3)(n-2)$
- TBR: tree bisection and reconstruction $(2n_1-3)(2n_2-3)$

TBR 是最复杂的进化树剪接方式，NNI 和 SPR 都可以看做 TBR 的特例。

进化树重排

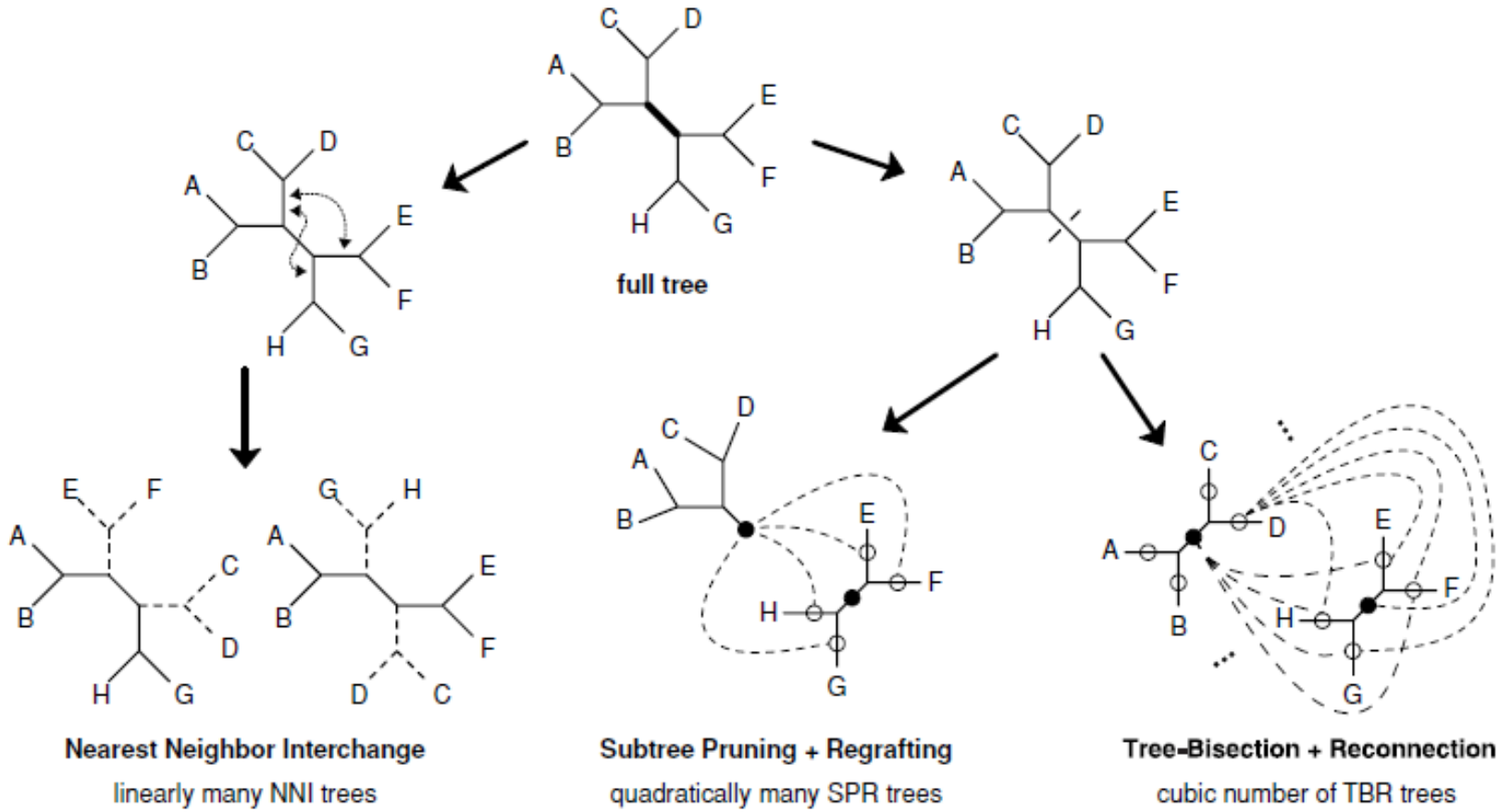


图 11: 进化树重排的方法：NNI,SPR 和 TBR

启发式搜索中的问题

启发式搜索，只保留每一步的最优树，作为下一步的起始值。在进化树的空间中，进化树可能最终到达局部最优，而无法达到全局最优。
为了进一步接近全局最优，人们开发了多种方法：

- 进化树合并 (tree-fusing)
- 遗传算法 (genetic algorithms)
- 进化树局部全搜索 (tree windowing)
- 加权搜索 (search by reweighting)
- 模拟退火 (simulated annealing)

贪婪算法可靠吗?

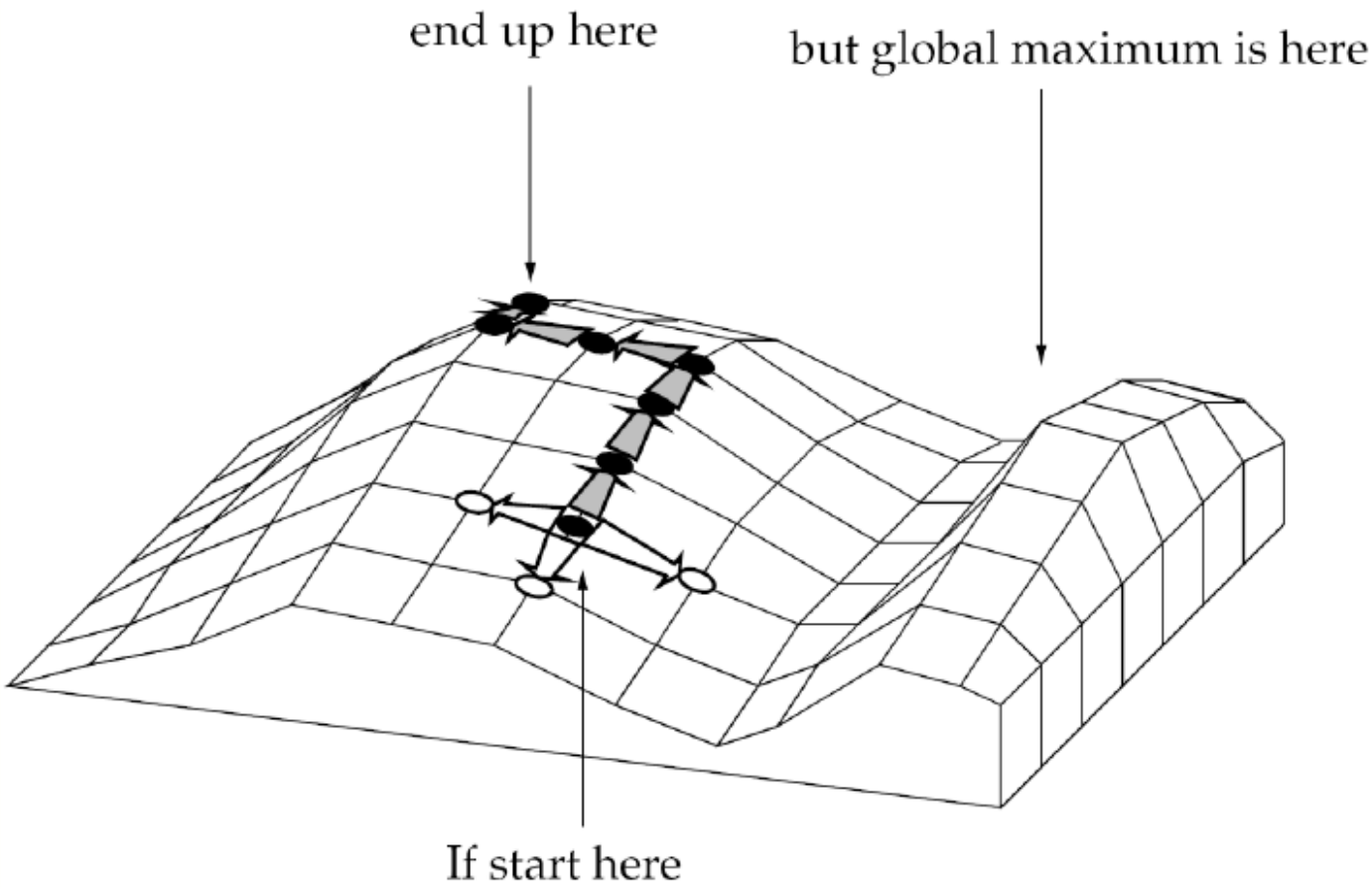


图 12: 贪婪算法为什么容易找到局域最优进化树?

进化树可信度评估 Bootstrap

已知样本数据，如何求某一统计量在总体中的分布?
Bootstrap 是利用样本数据进行可放回的抽样，形成大量 Boostrap 拟样本后，每个样本进行和源数据相同的分析，从而得到某一统计量的分布。如：从 Boostrap 的 1000 个 bootstrap 样本可获得 1000 棵进化树，对这 1000 棵进化树和真实数据形成的进化树进行比较，即可获得每个节点的支持率。

进化树的 Bootstrap 支持率

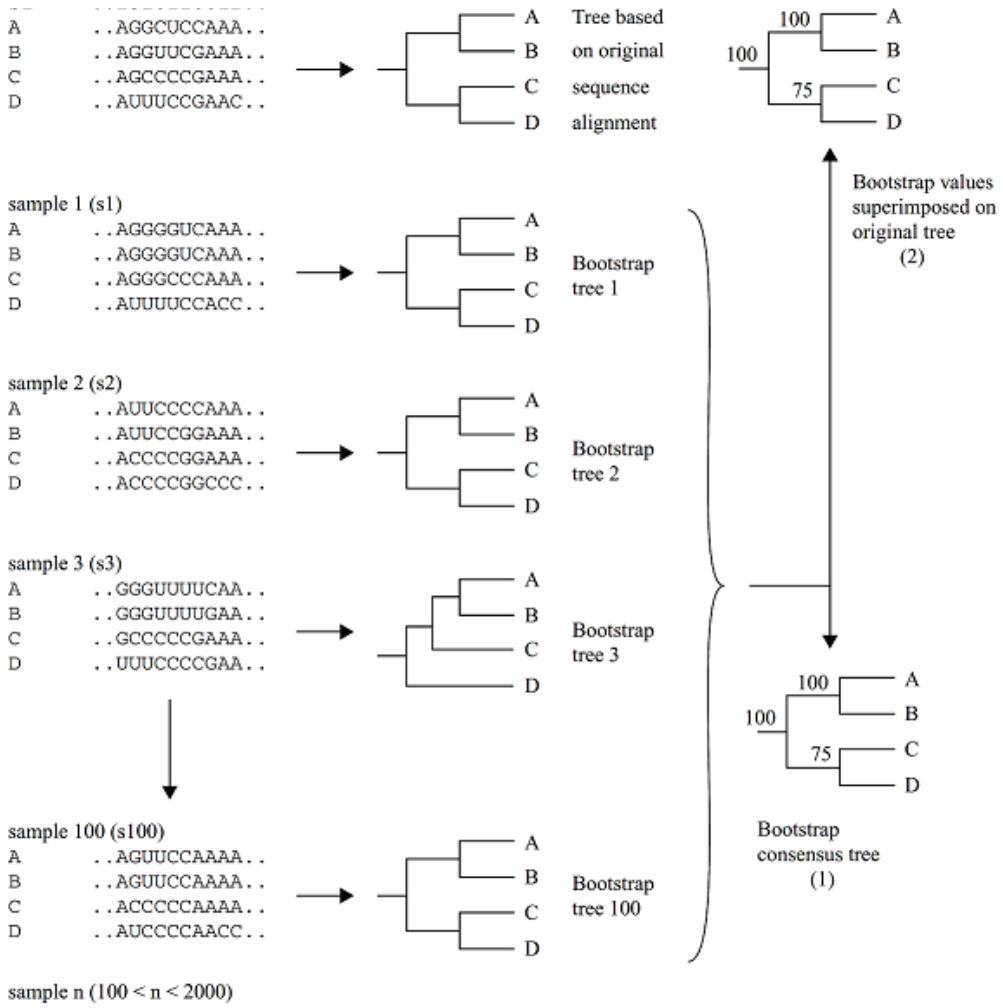


图 13: 进化树可靠性的 Bootstrap 检验

Maximum Likelihood 建树软件 I：往日的辉煌

- PHYLIP: 由美国 Joe Felsenstein 编写，开源软件，也是第一个用极大似然法搜索进化树的软件，适用于进化树入门学习和小规模数据的进化树建立
- PAUP*: 由美国 David Swoford 编写，特别适用于最大简约法，但是也包括极大似然法以及各种搜索以及进化树剪接，Bootstrap 等。还可以结合 ModelTest 进行进化模型筛选。但是需要购买版权。

Maximum Likelihood 建树软件 II：当今的王者

- PHYML：由法国的 Guidon 编写，适用于大规模数据（大量分类单元或大量信息位点）。支持多种模型的筛选。
- RAxML：由瑞士 Stamatakis 编写只支持 DNA 的 GTR 模型以及蛋白质的模型，适用于分类单元数较多，或者位点数较多时，建立极大似然进化树，运用快速 bootstrap，甚至可以基于整个基因组建立进化树。
- IQTREE：软件较新，由越南的 Bui Quang Minh 等开发。支持模型筛选、快速 bootstrap。效率和准确度甚至高于 RAxML，目前是建立进化树的最佳选择之一。

目录

频率学派

频率学派：给定当前的假设，观察到当前的样本的概率是多少?
求 P 值
通过样本数据，根据某一假设的统计分布假设，计算出的统计量，在统计量的分布中，出现在哪个区间，从而求得 P 值。

- 1 被估计的参数未知，但是有一个确定值。
- 2 需要根据重复取样对概率分布进行客观取样
- 3 样本量越大，参数的估计就越准确
- 4 进行极大似然估计

贝叶斯学派

贝叶斯推断是参数估计的一种方法，但是贝叶斯推断不是估计出参数的固定值，而是生成估计对象的分布。
贝叶斯学派：给定样本数据，待估计参数的概率分布如何?
主要特征：
1 待估计的参数是统计分布
2 主观得假设参数服从某种分布（先验分布）
3 无需大样本
4 基于统计模拟，对参数进行估计

贝叶斯法则

$$p(\theta|y) = \frac{p(y|\theta)p(\theta)}{p(y)}$$

θ 是待估计的参数 y 是样本数据 $p(y|\theta)$ 是样本的概率密度 (Likelihood)
$p(\theta)$ 是主观假定的参数概率密度 $p(y)$ 是 Normalizing constant

$$p(y) = \int P(y|\theta)P(\theta)d\theta$$

即所有生成当前数据的概率相乘。很多情况下，$p(y)$ 是无法用表达式求出的。而是要用到 MCMCMC 的方法。

贝叶斯方法的优缺点

优点

- 能够计算出待估计参数的概率分布
- 能够解决常规方法不能解决的复杂问题

缺点

- 需要高深的统计知识
- 需要强大的计算能力
- 在指定参数的先验概率分布时，需要进行足够的解释

一般用蒙特卡洛马尔科夫链 Metropolis coupling(Monte Carlo Markov Chain Metropolis Coupling, MCMCMC, MC^3) 的方法生成参数的分布，但是结果是否收敛，没有很好的评判标准。

蒙特卡罗马尔科夫链 Metropolis coupling (MCMCMC)

$$p(y) = \int P(y|\theta)P(\theta)d\theta$$

贝叶斯积分一般难以获得解析解。在无法获得参数联合概率密度的情况下，蒙特卡罗马尔科夫链可以保证人们在后验分布的参数空间中取样，当获得大量后验分布的样本后，即可获得后验概率的近似分布。

事件的马尔科夫性：某一事件在 t 时刻状态 B 的概率只与之前一个时刻 t_0 的状态 A 有关，而与 t_0 时刻之前的状态以及 t 之后的状态无关。

对于进化树

$$prob(Tree, \theta|Sequence) = \frac{prob(Tree, \theta)prob(Sequence|Tree, \theta)}{prob(Sequence)}$$

构建进化树思路的比较

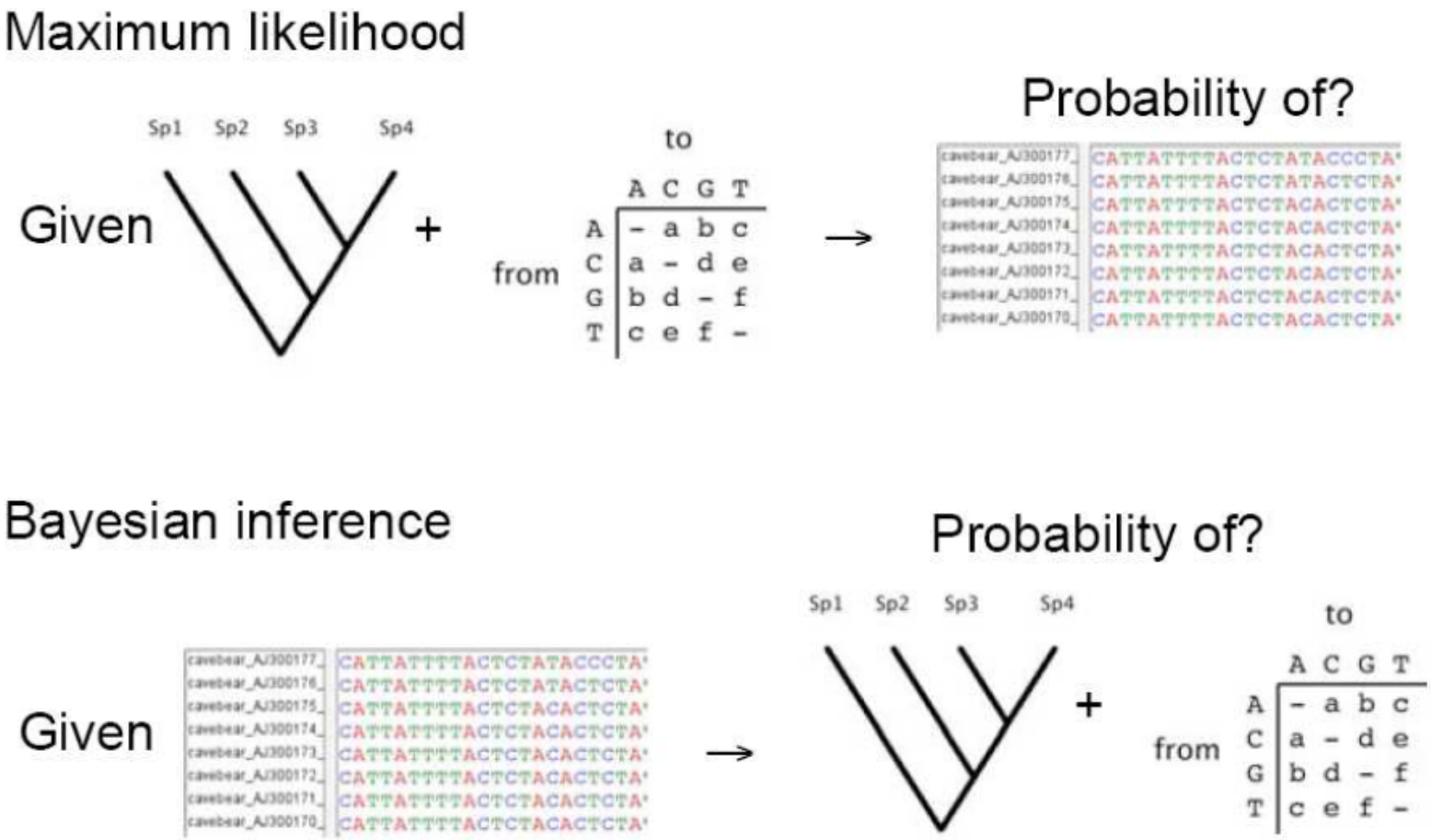


图 14: 极大似然与贝叶斯方法建树思路的比较

贝叶斯方法建立进化树的 MCMCMC

1. 以任意一棵进化树 T_i 开始，计算该进化树的似然值 L_i
2. 将进化树的参数 (即拓扑结构和枝长) 作随机变化，生成新的进化树 T_j，并计算 L_j
3. 计算两棵进化树的 (似然率 * 先验概率) 的比值 $R = \frac{f(T_j)}{f(T_i)}$
4. 如果 $R \geqslant 1$, 则接受 T_j
   如果 $R < 1$, 则在 $[0, 1]$ 之间取随机数 k, 若 $k < R$, 则保留 T_j，否则拒绝接受 T_j
5. 回到第 2 步

MCMC 在参数空间的随机游走（Random Walk）

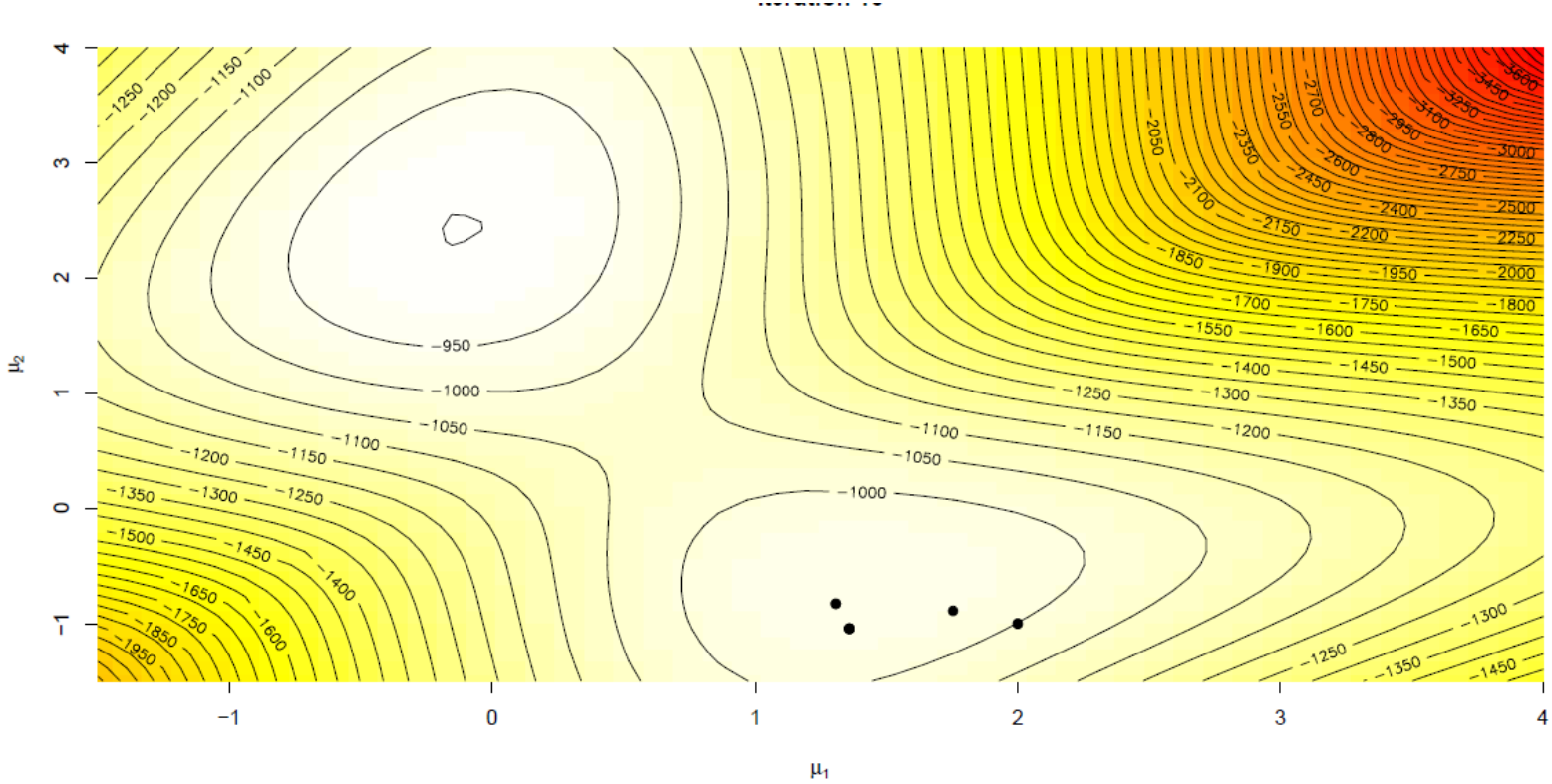


图 15: Random Walk 示意图，假设只有两个参数，数据在两个参数组成的空间中变化。通过 MCMCMC 的方法，获得后验分布。

冷链 Cold Chain 和热链 Hot Chain

- 参数随机改变的快慢程度，称为马尔科夫链的“温度”。变化越激烈，马尔科夫链越热。
- 在实际情况中，往往设置若干条温度不同的马尔科夫链，在参数空间中不断游走。
- 参数变化较快的马尔科夫链，称为热链，相反则称为冷链。
- 热链变化迅速，便于发现概率密度更高的区域，一旦发现概率密度更高的区域，则冷热链交换。
- 冷链在概率密度更高的区域内走动，热链则继续寻找概率密度更高的区域。
- 热链用于帮助马尔科夫链收敛，冷链用于精确样本的积累。

冷链和热链

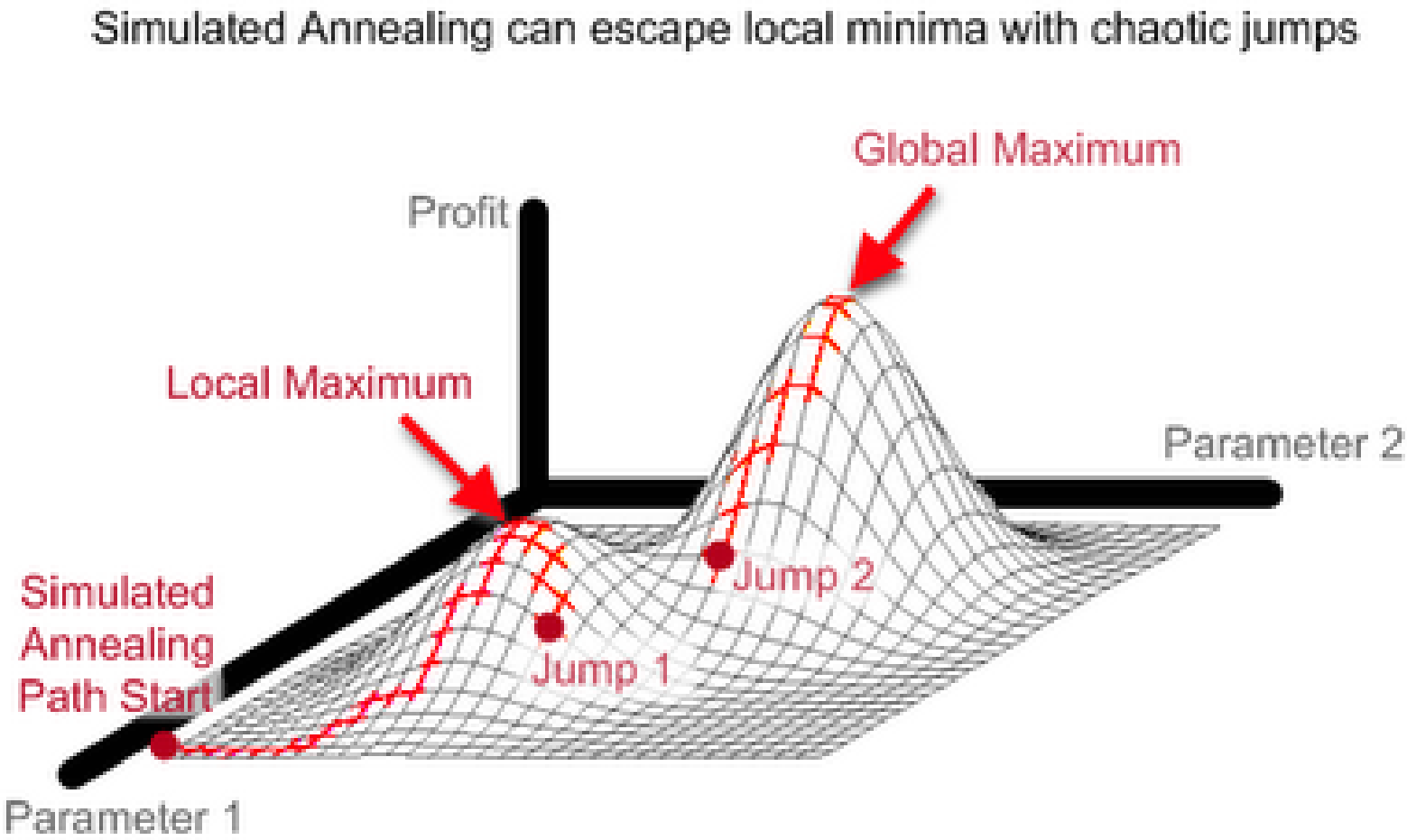


图 16: 冷链和热链示意图

贝叶斯方法建树的主要软件

- MrBayes：处理 nexus 文件，添加 MrBayes 命令模块。
- BEAST：处理 xml 文件，由 Beauti 生成。
- PhyloBayes：用于基因组水平矩阵的计算

贝叶斯实例

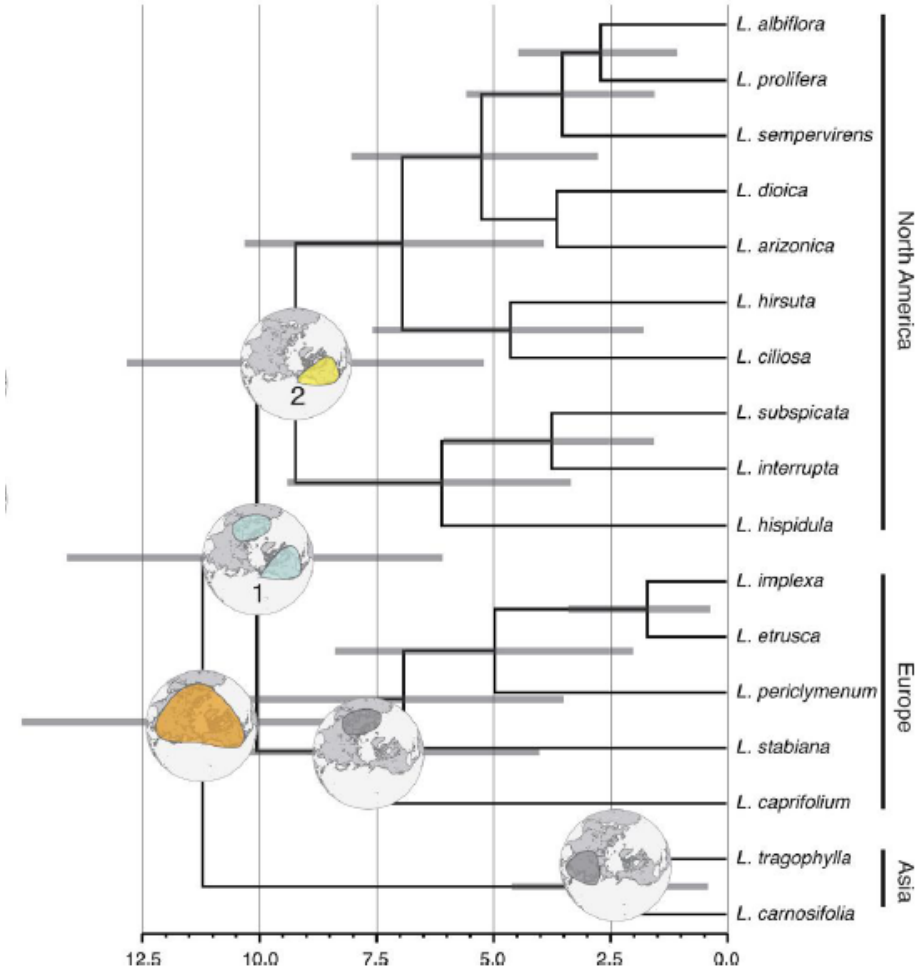


图 17: 北美忍冬的分化时间 Smith et al. 2010

目录

进化树枝长的意义

- 距离法: 枝长表示遗传距离。
- 简约法: 枝长表示变异的位点数。
- 最大似然法和贝叶斯法: 枝长表示连接这段枝长的两个节点之间每个位点的期望碱基替换数。

Chronogram: 枝长表示时间。
因此，一段枝长较长时，既可能是由于分化的时间足够长，也可能是由于碱基替换速率高，或者两者共同的决定的。
分子钟校正的目的，是将遗传距离、变异的位点数等转换为 chronogram。

分子钟常用软件

- r8s
  - 非参数速率平滑 NPRS：Nonparametric rate smoothing
  - 似然罚分法：PL (Penalized likelihood)
- BEAST 贝叶斯法，可以在建立进化树时，直接给出经过时间校正的等距树。
- PATHd8，用于较大进化树的定年
- ape 程序包也提供几种方法，如 PL

个人经验：定年要用没有多分枝结构的进化树

分子钟示意图

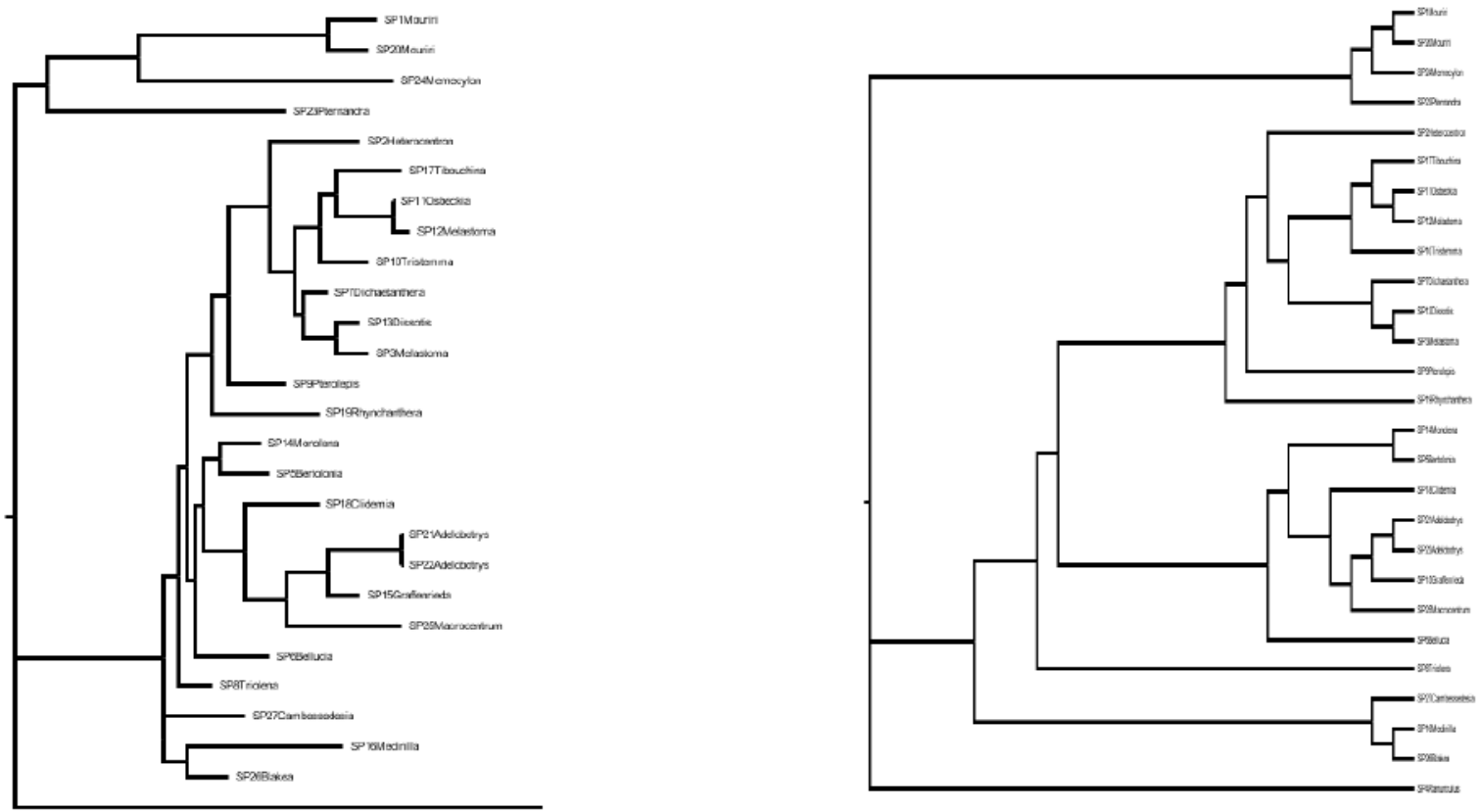


图 18: 分子钟校正前与校正后

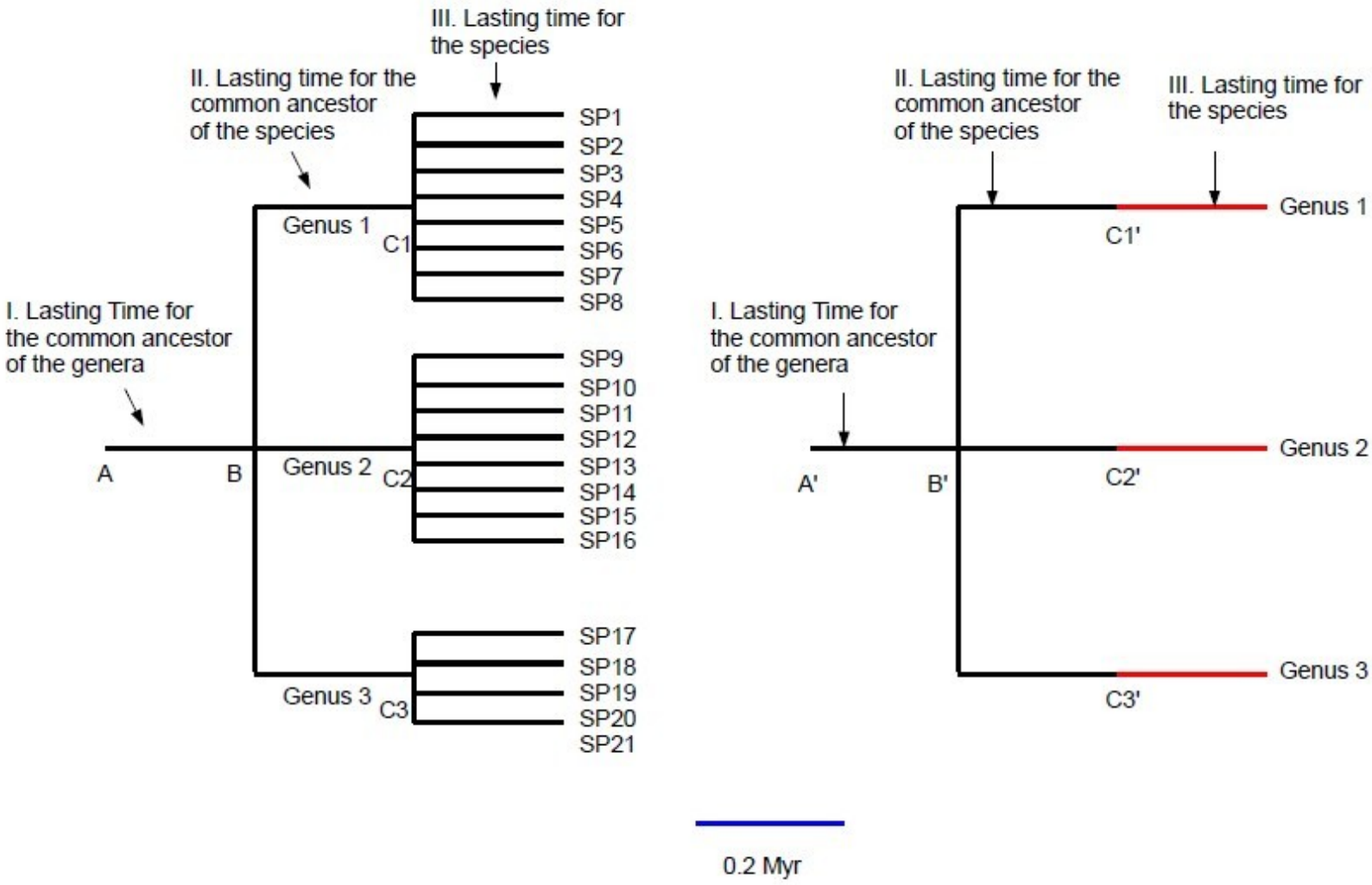
经过分子钟校正后的进化树，是一切系统发育多样性分析所要求输入的数据格式。

目录

通过物种名录建立进化树

- 在 DNA 条形码无法获得时，人们希望通过植物名录，以现有的各大类群，如科的进化关系为基础，建立粗略的进化树。
- 2004 年，Cam Webb 博士发布了 Phylomatic 软件。用户输入植物名录，就可以按照指定的进化关系，生成物种水平的 Newick 进化树。
- 进化时间可通过 Webb 博士提出的 BLADJ 算法，并根据 Wikstroem 对被子植物科的分化时间进行校正。
- 在科以下 BLADJ 算法假设，科的节点形成后，科、属、种，各自占据三分之一的分化时间。

BLADJ 算法示意图


III. Lasting time for the species
II. Lasting time for the common ancestor of the species
I. Lasting Time for the common ancestor of the genera
Genus 1
Genus 2
Genus 3
C1
C2
C3
A
B
SP1
SP2
SP3
SP4
SP5
SP6
SP7
SP8
SP9
SP10
SP11
SP12
SP13
SP14
SP15
SP16
SP17
SP18
SP19
SP20
SP21
0.2 Myr
(a) Species Level
II. Lasting time for the common ancestor of the species
III. Lasting time for the species
I. Lasting Time for the common ancestor of the genera
Genus 1
Genus 2
Genus 3
C1'
C2'
C3'
A'
B'
(b) Genus Level

Phylomatic 的不足

- 物种较多时，难以直接查询建树
- 植物之外的类群，Phylomatic 未收录
- 不能直接输入学名，获得进化树，而是先要查询所有种的科、属、种等信息

当分类单元很多：S.PhyloMaker 脚本

EDITOR'S CHOICE

An updated megaphylogeny of plants, a tool for generating plant phylogenies and an analysis of phylogenetic community structure FREE

Hong Qian, Yi Jin

Journal of Plant Ecology, Volume 9, Issue 2, April 2016, Pages 233–239,
https://doi.org/10.1093/jpe/rtv047
Published: 15 June 2015 **Article history** ▾

S.PhyloMaker 是一个 R 脚本，基于 Zanne 2014 的进化树，和给出的科、属、种列表，可建立包含几千种至几万分类单元的 newick 进化树。

当分类单元很多：V.PhyloMaker 程序包

Software note

V.PhyloMaker: an R package that can generate very large phylogenies for vascular plants

Yi Jin and Hong Qian

V.PhyloMaker 基于 Smith and Brown (2018) (i.e. GBOTB) 的进化树，提供几种查询模式。

批量查询植物的科、属、种

- Taxonomic Name Resolution Service
  `http://tnrs.iplantcollaborative.org/`
- Taxonstand R 包：批量查询植物的科属
- taxize R 包: 集成查询多种数据库
- plantlist R 包: 批量查询科属和制作植物名录等

用 plantlist 批量查询科名

```
# devtools::install_github("helixcn/plantlist")
library(plantlist)
species <- c(
"Ranunculus japonicus",
"Anemone udensis",
"Ranunculus repens",
"Ranunculus chinensis",
"Solanum nigrum",
"Punica sp."
)
col <- TPL(species)
taxa.table(col) # 生成科属种列表
```

目录

系统发育比较分析

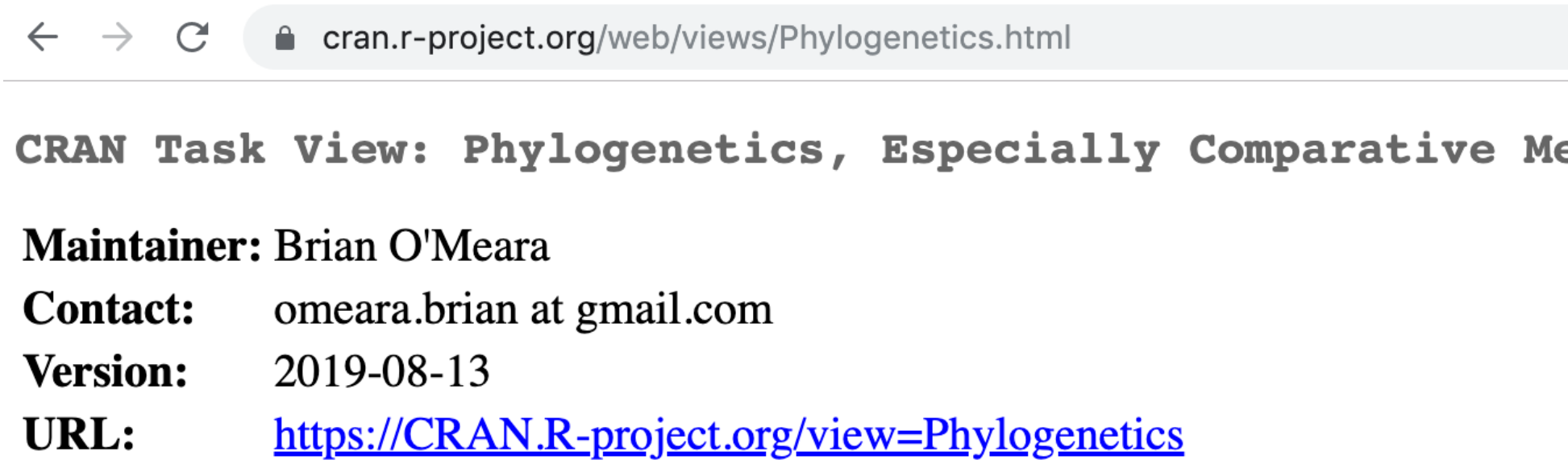


图 19: CRAN Task View: Phylogenetics

进化树能帮我们回答哪些问题?

- 分类单元之间的系统发育关系
- 物种的形成速率受什么影响? 如: 类群的古老程度, 类群的丰富度, 类群所处的纬度, 类群的特殊生境
- 物种形成速率和灭绝速率在历史上发生过哪些变化?
- 物种的进化历史越独特, 越应该受到保护吗?
- 相近的物种有相似的性状吗? 有相似的习性吗?

进化树能帮我们回答哪些问题?

- 如果已知某一分支的性状，是否能够了解其祖先的性状?
- 物种的适应性是如何进化的?
- 已知物种的当前分布区，如何获得其祖先分布区?
- 群落内物种的组成是随机的，还是由于对生境的偏好造成的?

系统发育比较分析的主要方向

- 性状进化 Trait evolution
- 性状进化模拟 Trait Simulations
- 群落系统发育 Community/Microbial Ecology
- 祖先状态重建和气候适应性进化 Phyloclimatic Modeling
- 祖先分布区重建 Phylogeography/Biogeography
- 物种与种群的界定与模拟 Species/Population Delimitation

ape 程序包

ape 是 Analysis of Phylogenetics and Evolution 的缩写，作者是法国进化生物学家 E. Paradis 博士。

ape 程序包的主要功能

- 调整、读取和绘制进化树
- DNA 序列的读取以及遗传距离计算
- 建立小型进化树
- 进行分子钟校正并估计进化速率
- 模拟生成随机进化树

参考：Paradis, E. (2012) Analysis of Phylogenetics and Evolution with R (Second Edition). New York: Springer.

进化树的格式：Newick 进化树转换成 phylo 类

```
# Newick进化树
owls(((Strix_aluco:4.2,Asio_otus:4.2):3.1,Athene_noctua:
7.3):6.3,Tyto_alba:13.5)

### 读取进化树
tree.owls <- read.tree("ex.tre")
```

显示的结果

```
Phylogenetic tree with 4 tips and 3 internal nodes.
Tip labels:
[1] "Strix_aluco" "Asio_otus" "Athene_noctua" "Tyto_alba"
Rooted; includes branch lengths.
```

进化树的格式：Newick 进化树转换成 List

```
> str(tree.owls)
List of 4
 $ edge    : int [1:6, 1:2] 5 6 7 7 6 5 6 7 1 2 ...
 $ Nnode   : int 3
 $ tip.label  : chr [1:4] "Strix_aluco" "Asio_otus" "Ather
 $ edge.length: num $[1:6]$ 6.3 3.1 4.2 4.2 7.3 13.5
 - attr(*, "class")= chr "phylo"
 - attr(*, "order")= chr "cladewise"
```

群落数据的准备和转换

野外记录的格式

	plot	species	abundance
1	plot1	sp1	3
2	plot1	sp2	6
3	plot1	sp3	1
4	plot1	sp4	2
5	plot1	sp5	1
6	plot2	sp1	8
7	plot2	sp3	30
8	plot3	sp4	2
9	plot3	sp2	1
10	plot3	sp6	1
11	plot3	sp7	3

spaa::data2mat(testdata)

转换为物种矩阵

- 行表示样方，第一列作为行名。
- 列表示物种，第一行作为物种名。
- 行列交叉处表示物种的个体数，若没有出现，则用 0 来表示。

R 中物种多样性指数计算、排序、系统发育多样性分析、种间联结计算和生态位重叠分析，都使用这种矩阵。

	sp1	sp2	sp3	sp4	sp5	sp6	sp7
plot1	3	6	1	2	1	0	0
plot2	8	0	30	0	0	0	0
plot3	0	1	0	2	0	1	3

vegan 程序包：群落数据分析

Community Ecology Package: Ordination, Diversity and Dissimilarities
作者是芬兰生态学家 Jari Oksanen.

- Alpha 多样性计算: Shannon, Simpson, Pielou 多样性指数等
- 排序 CCA、DCA、CA、NMDS
- 种面积曲线
- 物种多度分布曲线
- 方差分解
- beta 多样性的计算等

alpha 多样性：每个样方的多样性特征

vegan 程序包下计算多样性相关的几个指数：

`diversity()`	多样性计算
`rarefy()`	面积不等的两个群落比较时，进行随机抽样
`fisher.alpha()`	计算多样性
`specnumber()`	物种数的累计

```
data(BCI)
H <- diversity(BCI)
simp <- diversity(BCI, "simpson")
invsimp <- diversity(BCI, "inv")
S <- specnumber(BCI)
J <- H/log(S)
```

beta 多样性：群落之间的相似性或相异性

```
vegdist(x, method="bray", binary=FALSE, diag=FALSE,
upper=FALSE, na.rm = FALSE, ...)
```

相似性指数的计算，一般都将转换为 disimmilarity 才能进行后续计算。

- **Jaccard** 指数：适用于 01 数据，在计算 beta 多样性中，
- **Bray-Curtis** 指数：适用于物种多度数据

环境距离的计算：欧几里得距离

设想二维平面直角坐标系，有 A,B 两点，A 坐标为 (x_1, y_1)，B 坐标为 (x_2, y_2)
用 dist() 用来求欧几里得距离

$$d(A, B) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

推广到高维：

$$d(p, q) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \cdots + (q_n - p_n)^2}$$

$$d(p, q) = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}$$

欧几里得距离的计算

```
x <- matrix(rnorm(100), nrow = 5)
dist(x)
dist(x, diag = TRUE)
dist(x, upper = TRUE)
m <- as.matrix(dist(x))
d <- as.dist(m)
```

获得的距离矩阵可以进一步用于进行聚类分析等。

地理距离的计算

地球可看做近似椭球，经纬度是球面坐标，此时可用 sp 程序包等计算地理距离。假设 A 地为 L1、Phi1, B 地为 L2,Phi2, 求两地的地理距离

```
library(sp)
longitude <- as.numeric(c(L1, L2))
latitude  <- as.numeric(c(phi1, phi2))
cities <- c("Paris", "Washington")
location <- cbind(longitude = -longitude, latitude)
row.names(location) <- cities
location <- data.frame(location)
coordinates(location) <- ~longitude+latitude
proj4string(location) <- CRS("+proj=longlat +datum=WGS84")
spDists(location)
```

Mantel 检验：距离矩阵之间的相关性

两个距离矩阵的相关性不能直接用 Pearson 相关性来表示，而是需要用 Mantel 检验

```
ape::mantel.test(m1, m2, nperm = 999, graph = FALSE,
alternative = "two.sided", ...)
```

示例：

```
data(varespec)
data(varechem)
veg.dist <- vegdist(varespec) # Bray-Curtis
env.dist <- vegdist(scale(varechem), "euclid")
mantel(veg.dist, env.dist)
```

beta 多样性是由哪些因素决定的?

adonis: Analysis of variance using distance matrices
—for partitioning distance matrices among sources of variation and fitting linear models (e.g., factors, polynomial regression) to distance matrices; uses a permutation test with pseudo-F ratios.
mrpp: Multi Response Permutation Procedure and Mean Dissimilarity Matrix
anosim: Analysis of Similarities
MRM: 基于 Mantel test 对 beta 多样性进行方差分解

系统发育信号 (Phylogenetic signal)

- **系统发育信号 (Phylogenetic signal)**用于检验系统发育关系相近的物种是否具有相似性状。
- Blomberg's K(Blomberg and Garland 2002, Blomberg et al. 2003, Butler and King 2004) 是最常用的系统发育信号指数，K 值是类群间性状差异偏离于随机过程的度量。
- `picante::phylosignal()`

系统发育多样性

系统发育多样性 PD：Phylogenetic Diversity

- Faith(Faith, 1992) 提出的系统发育多样性 (Phylogenetic diversity, PD) 给出的是某一地点所有物种在进化树上的枝长之和。
- `picante::pd()`

为什么要提出系统发育多样性和系统发育信号?

- 系统发育多样性：保护历史悠久而独特的分支类群
- 系统发育信号：探讨性状和系统发育之间的关系

因此：系统发育多样性，常用于优先保护地的归化。系统发育信号，常用于检验近缘种，是否拥有相近的性状特征。

群落系统发育的理论框架

- 相似的环境，往往会选择拥有相近性状的物种，这种作用称为生境过滤作用（Environmental Filtering）。
- 如果物种的性状越相似，系统发育关系越接近，也就是有显著的系统发育信号，那么通过分析群落内物种之间平均或者最近的系统发育距离，并将该距离和随机生成的距离相比，就可以判断生境过滤作用是否显著存在。
- 如何实现？Cam Webb 博士最早提出了基于 PD 的四个指数：MPD、MNTD、NRI 和 NTI

MPD、MNTD、NRI 和 NTI

- NRI 和 NTI 是最早提出的群落系统发育指数，表示群落内物种系统发育的距离高于或者低于零模型给出的距离。
- 计算群落内物种系统发育的距离常用指数有 MPD 和 MNTD。
- MPD 是计算群落内物种两两之间的系统发育距离的平均值；
- MNTD 是寻找群落内每个物种系统发育关系最近的物种的系统发育距离，并计算所有最近物种的系统发育平均距离。
  MPD 和 MNTD 和与零模型进行比较之后，分别得到 NRI 和 NTI 指数。

NRI 和 NTI 的计算

$$NRI_{sample} = -1 \times \frac{MPD_{sample} - MPD_{rndsample}}{sd(MPD_{rndsample})}$$

$$NTI_{sample} = -1 \times \frac{MNTD_{sample} - MNTD_{rndsample}}{sd(MNTD_{rndsample})}$$

- 四个指数最早通过 Phylocom 软件计算，后来多通过 picante 软件包计算。
- `ses.mpd()` 等价于 NRI
- `ses.mntd()` 等价于 NTI

群落的零模型

为什么要有零模型?
要用真实群落计算的某一指数和若干个随机化之后的群落所得指数进行比较，以观察真实值是否落在 95% 的置信区间里，从而对生态学机制做出进一步的推断。

- 群落系统发育分析中的零模型是将群落内物种组成的关系进行随机化的一系列方法。
- 按照中性理论，物种与物种之间是等同的，群落中物种应该是随机组合的。
- 因此可按照一定的规则，将物种在群落内出现的方式进行随机化。

常用的几种零模型

群落系统发育分析中，一般通过以下四种方式进行随机化。

- **Null 0** 群落数据不变，但是物种在进化树末端随机排列。
- **Null 1** 进化树不变，物种在样方中随机排列，物种从所有样方中随机选取。
- **Null 2** 进化树不变，物种在样方中随机排列，物种从指定的物种库中选取。
- **Null 3** 进化树不变，与此同时，物种在样方中成对的关系保持不变。这种随机化的方法称为独立交换法 (Independent swap)。

系统发育 beta 多样性

系统发育 beta 多样性是群落或地点之间系统发育距离的度量 (Fine and Kembel 2010)

comdist()& comdistnn()	:MPD 和 MNTD (Webb 2000)
phylosor()	:Phylosor (Bryant et al. 2008)
unifrac()	:Unifrac(Lozupone et al. 2006)
rao()	:Rao 1982, Jost 2007, Webb et al. 2008
pcd()	:PCD (Ives and Helmus 2010)

系统发育多样性研究还活跃吗?

- 1 地理尺度系统发育 beta 多样性的成因探讨，如纬度变化，解释宏生态学的一些假说（如 Out of Tropics 假说）；
- 2 群落尺度内，系统发育结构、种面积曲线，系统发育多样性面积曲线和点格局分析进一步整合，以探讨群落组成的理论问题。
- 3 群落尺度的数据和物种分布信息和适应性进一步整合，了解物种在局域尺度的适应性。

系统发育多样性研究还活跃吗？II

- 4 物种的功能性状数据和物种在不同尺度的分布信息，以了解物种分布和适应性的一些机理。
- 5 用功能基因组学和转录组的手段，探讨物种的适应性和分布。
- 6 借助群体遗传学，亲本分析等手段进行物种群落分布格局的研究，进一步分析群落系统发育结构等的成因。
- ……

谢　谢！
敬请批评指正！