R编程与进化分析

第三部分进化树及系统发育比较分析

张金龙

jinlongzhang01@gmail.com

2016年5月9日北京

目录

目录

什么是进化树

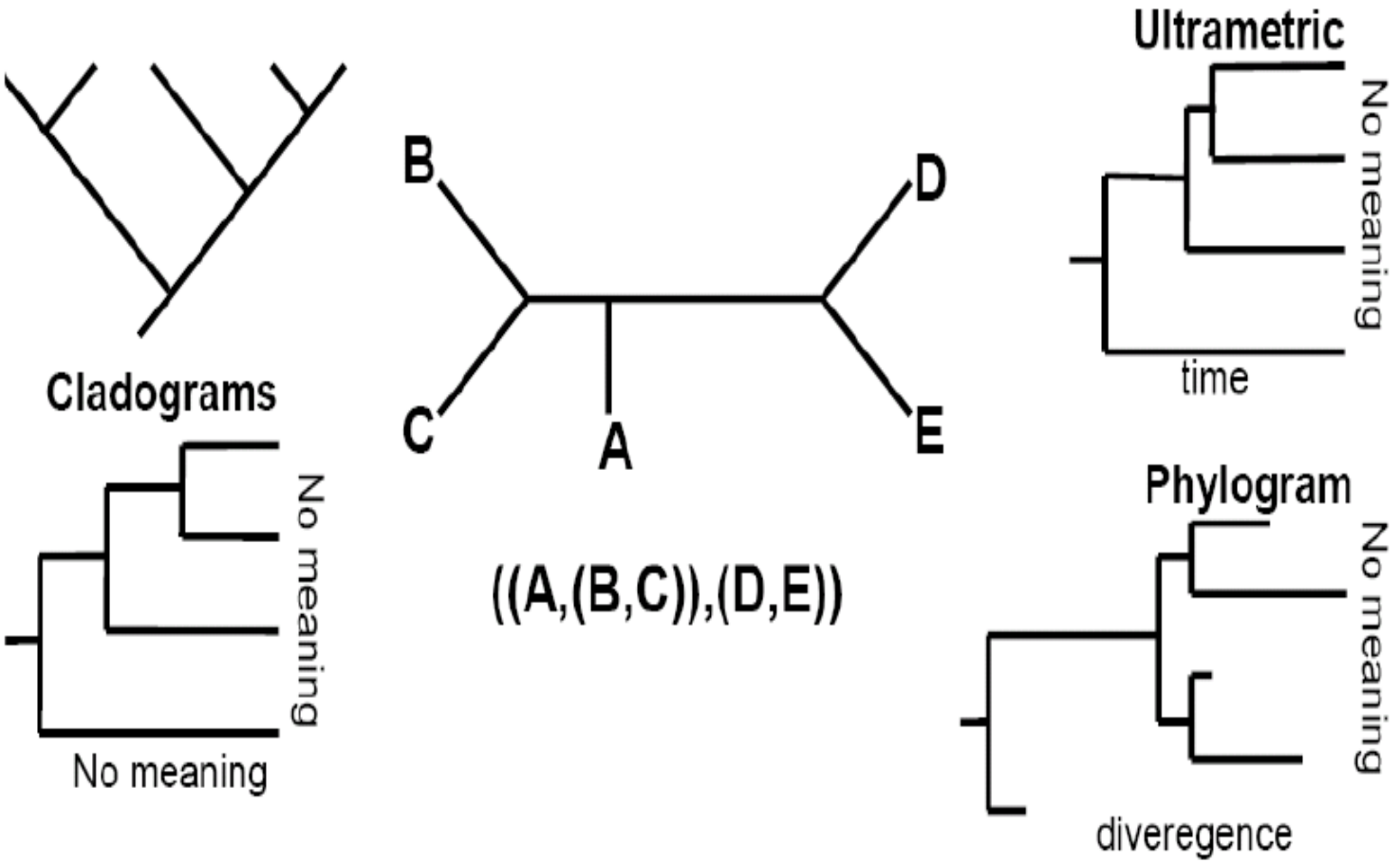
从DNA序列等推断出的物种系统发育关系，用树状图表示，即为进化树。进化树是进行系统发育比较分析 Phylogenetic Comparative Methods的基础。

- 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。
- 体现分类单元甚至个体的系统发育关系，具有精确的枝长。
- 体现分类单元之间的系统发育关系，枝长以时间为单位。

进化树的里程碑

- 1859年，达尔文引入了亲缘关系的树状图
- 十九世纪末，海克尔用进化树表示物种之间的亲缘关系。
- 1964年， Cavalli-Sforza 和 Edwards提出了用简约法和极大似然法构建进化树。
- Hennig 提出了分支理论（Theory of cladistics）
- 1977年， Fitch将简约法应用到构建进化树中。
- 1978年， Felsenstein首次用极大似然法构建进化树
- 1996年， Rannala和杨子恒将贝叶斯推断引入到进化树的构建中。

各种进化树


Cladograms
No meaning
No meaning
B
D
C
A
E
((A,(B,C)),(D,E))
Ultrametric
No meaning
time
Phylogram
No meaning
diveregence

Newick Format

```
1 a <- c(24,43,58,67,61,44,67,49,59,52,62,50,42,43,65,26,33,41
  ,19,54,42,20,17,60,37,42,55,28)
2 group <- factor(c(rep("A",12),rep("B",16)))
3 data <- data.frame(group,a)
4 find.mean <- function(x){ mean(x[group=="A",2])-mean(x[group
  =="B",2]) }
5 results <- replicate(999,find.mean(data.frame(group,sample(
  data[,2]))))
6 p.value <- length(results[results>mean(data[group=="A",2])-
  mean(data[group=="B",2])])/1000
7 hist(results,breaks=20,prob=TRUE)
8 lines(density(results))
```

dopted by James Archie, William H. E. Day, Joseph Felsenstein, Wayne Maddison, Christopher Meacham, F. James Rohlf, and David Swofford, at two meetings in 1986, the second of which was at Newick's restaurant in Dover, New Hampshire, US. PHYLIP,RAxML等就用这种格式。

Nexus格式

```
#NEXUS
Begin data;
Dimensions ntax=4 nchar=15;
Format datatype=dna missing=? gap=-;
Matrix
Species1   atgctagctagctcg
Species2   atgcta??tag-tag
Species3   atgttagctag-tgg
Species4   atgttagctag-tag
;
End;
```

是Newick格式的升级版，可以增加多种blocks。BEAST，PAUP*, MrBayes, Mesquite，r8s等都使用该格式。

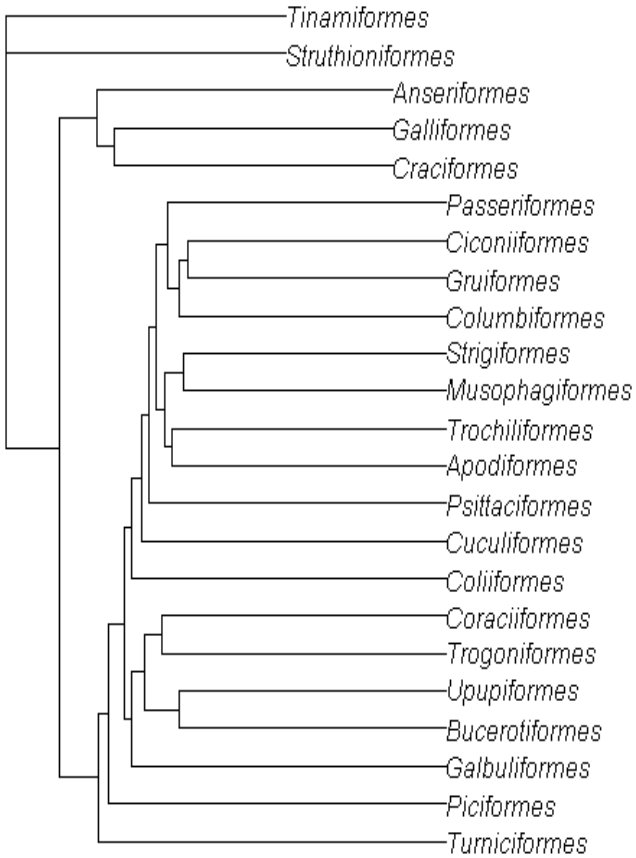
其他格式

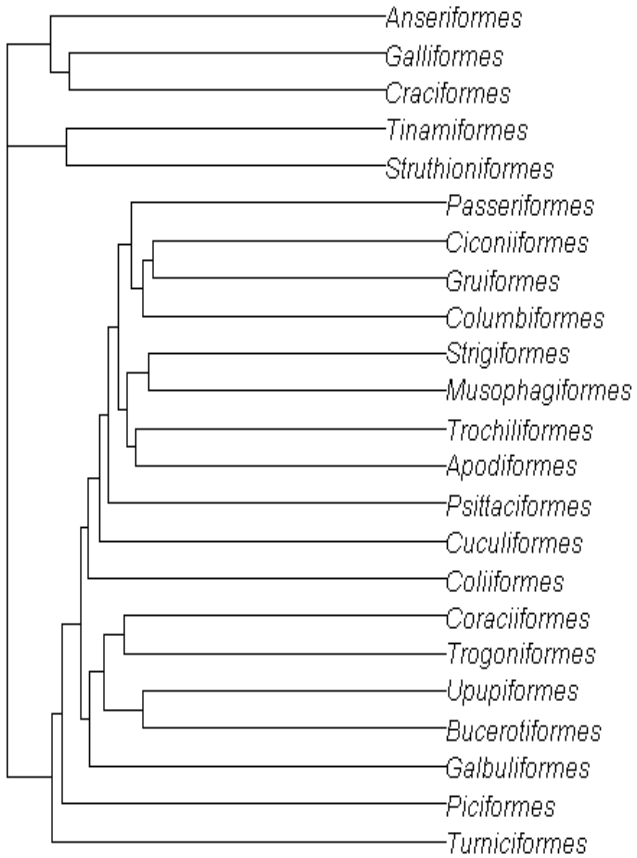
New Hampshire eXtended format (NHX) （http://home.cc.umanitoba.ca/psgendb/doc/atv/NHX.pdf）
Jplace
（http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0031009）

重要概念: 等距树、有根树和外类群

- 等距树 Ultrametric Tree: 每个末端分类单元都与根节点距离相同。
- 有根树 Rooted Tree: 探讨某类群系统发育关系时，在该类群之外寻找的一个或者几个外类群。经过调整顺序后，便于分析内部的系统发育关系。这样的进化树称为有根树。
- 外类群 Outgroup：与研究的类群系统发育举例适中，不能处在研究的类群中，也不能太远。

无根树和有根树以及外类群





重要概念：一致性树 Consensus Tree

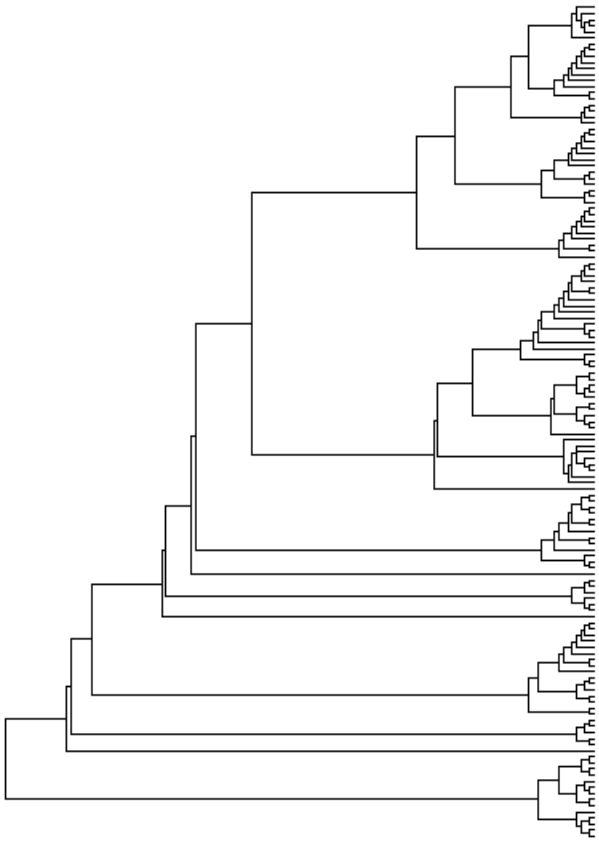
当获得多个进化树时，可以对进化树内部节点的支持率进行汇总。按照节点一致性的原则，选出各进化树都支持的某节点的拓扑结构，以及所支持的百分率，称为节点支持率。

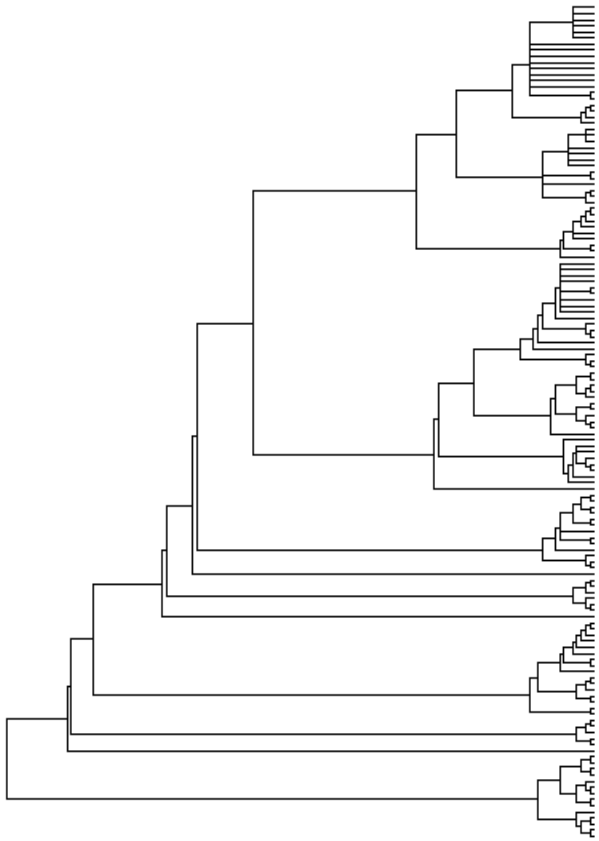
重要概念：二叉树， Singleton和Polytomies

- 正常情况下，每个节点下，应该有两个子代。这种进化树称为二叉树。
- 若一个节点下，只有一个节点，称为孤立节点 Singleton。
- 若某一节点下，有多个子代，则生成如降落伞一样的多分枝结构。

形成多分枝结构的原因：物种数据缺失，如建树的基因突变率很低，无法反应出足够的系统发育距离。
用Phylomatic软件等生成的进化树，常常包含 Singletons以及多分枝结构。

多分枝结构





DNA序列

DNA序列的获取

- 1. DNA barcoding 测序 AB1 file > Fasta File
- 2. Genebank Fasta File
- 3. Next Generation Sequencing (Genome)

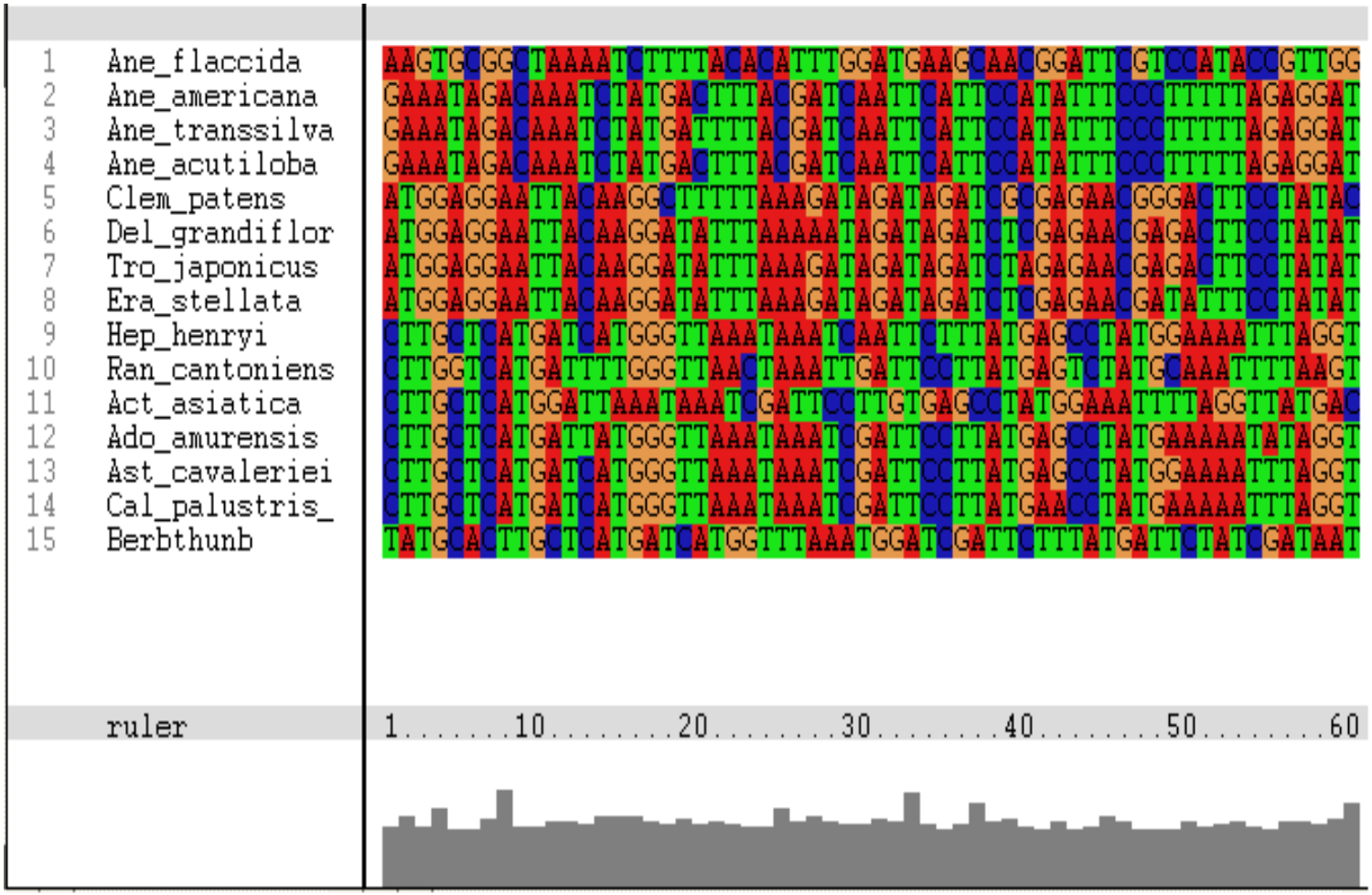
FASTA文件

DNA序列的纯文本文件，默认为5‘到 3’的顺序

- 1. 每个物种的名称以>开始
- 2. 下一行为核苷酸序列。
- 3. 序列与序列之间，用一个空格分开。
- 3. 用减号表示缺失的位点，可以保存比对好的序列。

序列比对Alignment

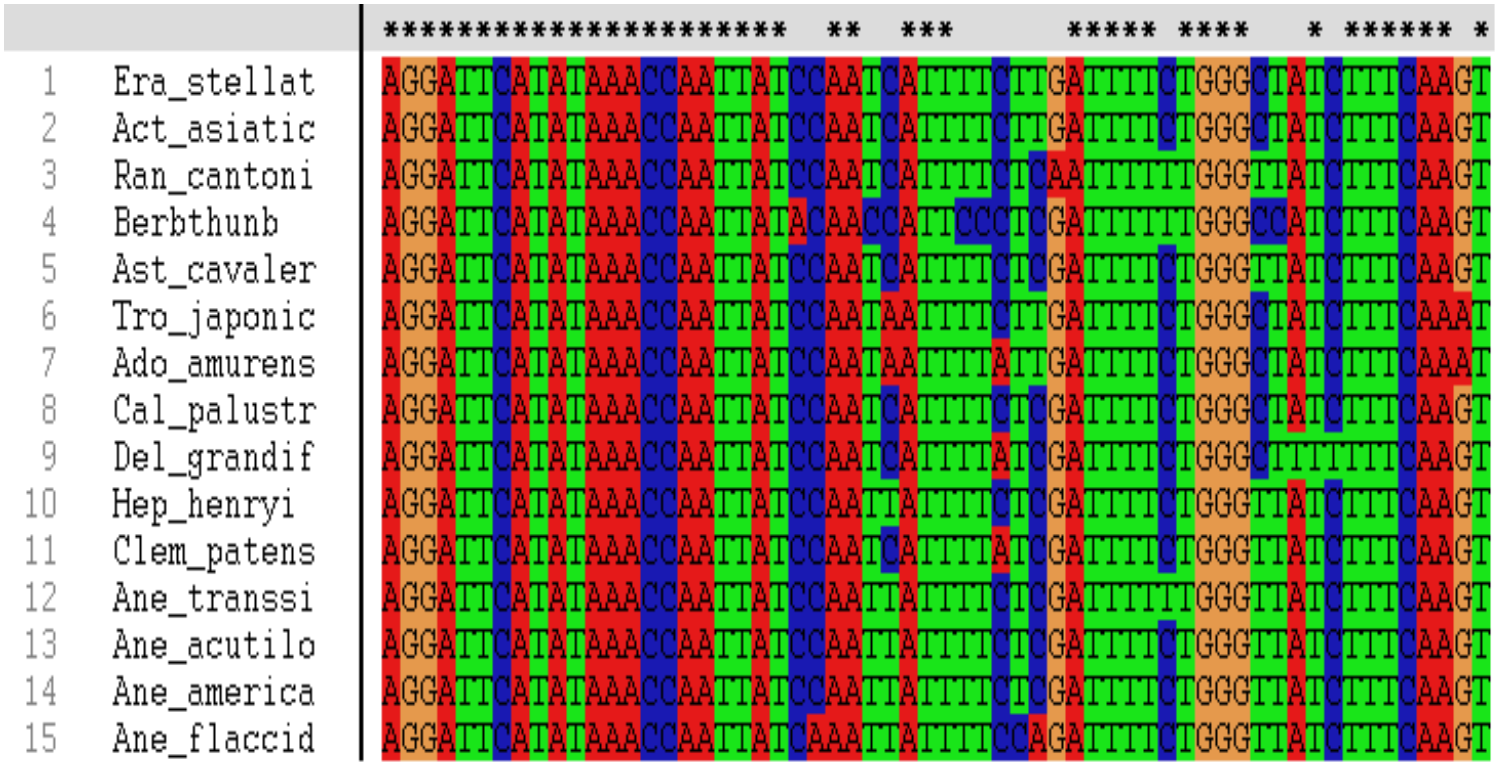
将DNA序列对齐，以便寻找突变的位点，以推断系统发育关系



DNA序列在比对前

序列比对Alignment

将DNA序列对齐，以便寻找突变的位点，以推断系统发育关系



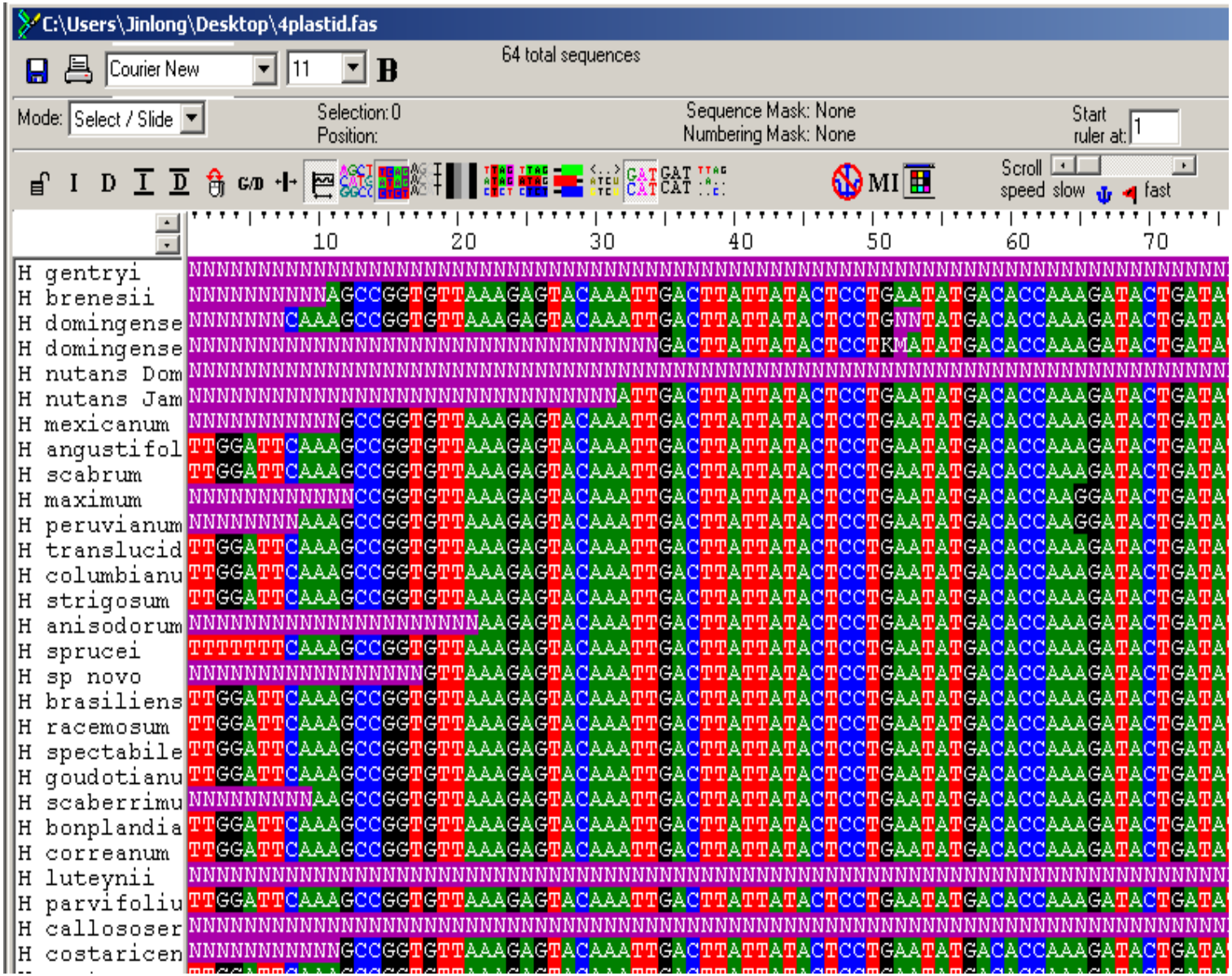
DNA序列在比对后

序列比对之后的检查和校对

为什么要检查和校对？测序的错误，如ab1峰图文件，在序列的起始以及结束20bp左右，质量不佳，容易误判。主要用到的软件：

- Bioedit
- ChromasLite
- ContigExpress
- DNAman等

Bioedit检查和校对比对好的DNA序列



碱基替换模型

ATCG之间的替换，发生的概率

$$Prob[kevents] = \frac{\mu t^k e^{-\mu t}}{k!}$$

考虑ATCG之间的碱基替换率，可以写成transition matrix 即转移概率矩阵。

转移概率矩阵

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

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

JC69模型 Jukes-Cantor 1969

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

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

Kimura 81模型和F81模型

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

F84模型

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

HKY85模型和Tamura-Nei93

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

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

HKY85模型和Tamura-Nei93

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

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

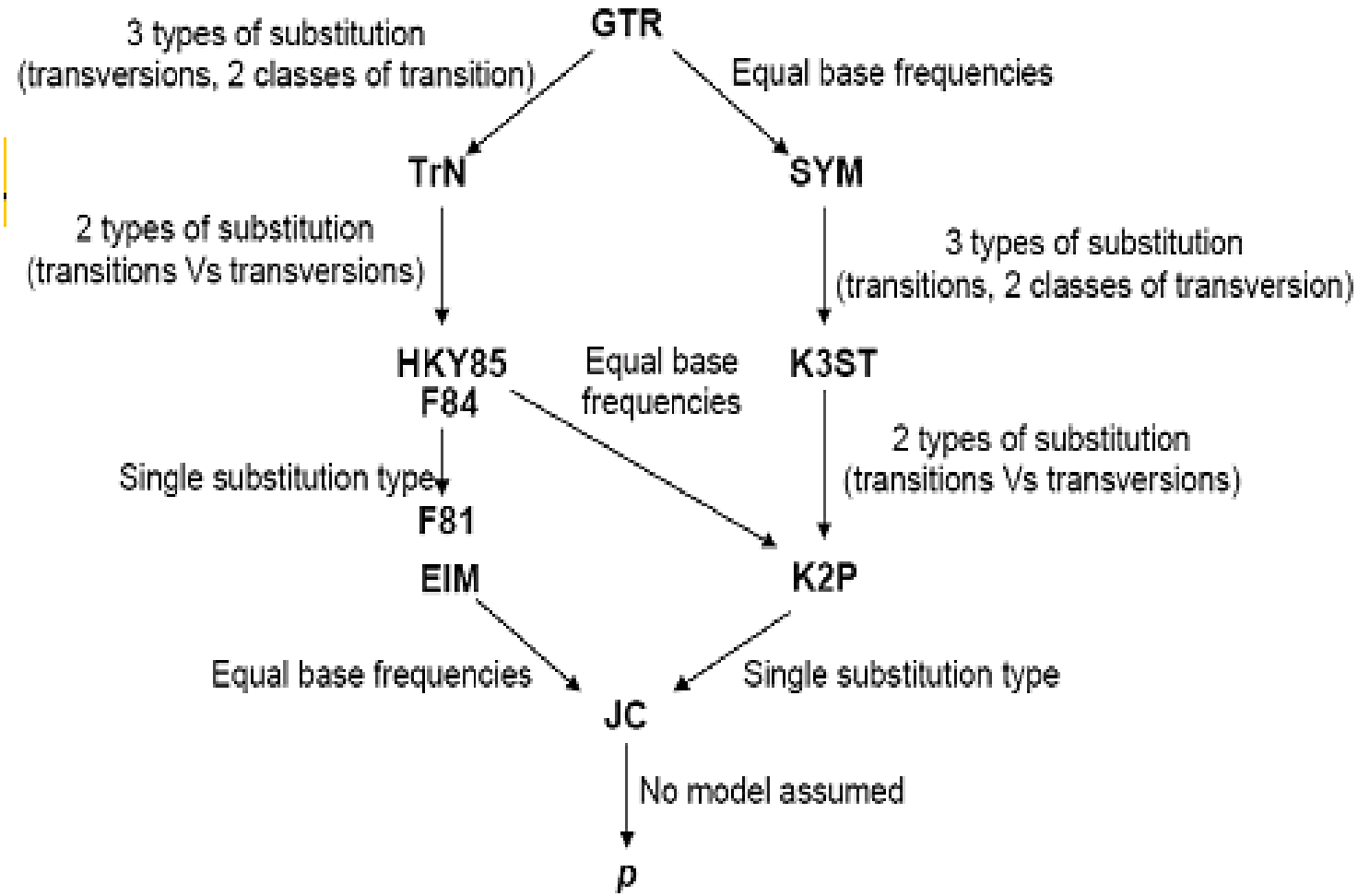
GTR: General Time Reversible Model

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

DNA碱基替换模型之间的关系



Gamma 分布拟合不同碱基的不同替换速率

不同位置的核苷酸具有不同的编译速率，这种变异一般用Gamma分布进行拟合。

进化模型的筛选

模型的筛选：用 Modeltest， Jmodeltest等，依据AIC或者LRT进行筛选。

如果数据较小，如出现的信息位点较少，对于GTR模型+Gamma容易出现模型的过度拟合，此时就需要进行模型筛选。对于大数据，建议用GTR+Gamma。

基于遗传距离的建树方法

- 距离法
- 简约法
- 极大似然法
- 贝叶斯法

基于遗传距离的建树方法

- 计算序列之间的遗传距离，基于RAW或者Transition或者Tranversion，或者其他DNA碱基替换模型。
- 进行聚类，一般用UPGMA聚类。
- Neighbour Joining方法
- Minumum Evolution
- 基于遗传距离聚类的进化树已经很少使用!

简约法

简约法背后的哲学：真实的进化历史背后，所经历的进化事件最少。即碱基突变的位点最少
简约法的优点

- 容易解释
- 速度快
- 一般情况下，进化树结构准确。

简约树常用于构建其他进化树的起始树。

极大似然法 1

假设已经有一棵进化树T，以及碱基替换模型Q，获得当前DNA比对格局的可能性有多大？ Likelihood

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

计算中需要用到如下假设：

- 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：

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

极大似然法 2

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:

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

为什么构建进化树极为耗时？

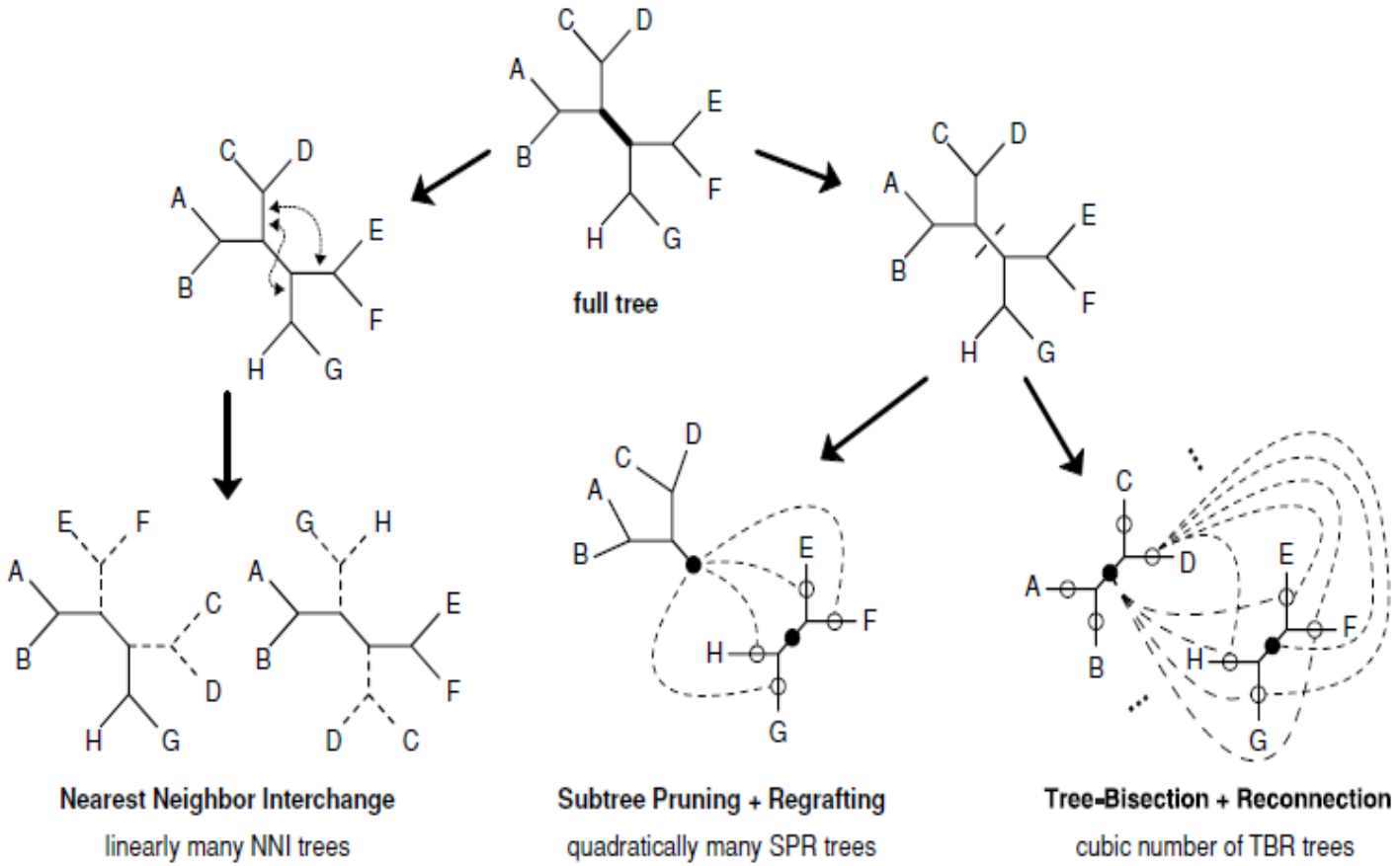
分类单元数：可能的进化树的数量

3: 1
5: 15
10: 2027025
15: 7905853580625
16: 213458046676875
17: 6190283353629374
18: 191898783962510624
19: 6332659870762850304
20: 2.216430954767e+20
30: 8.687364e+36
50: 2.838063e+74

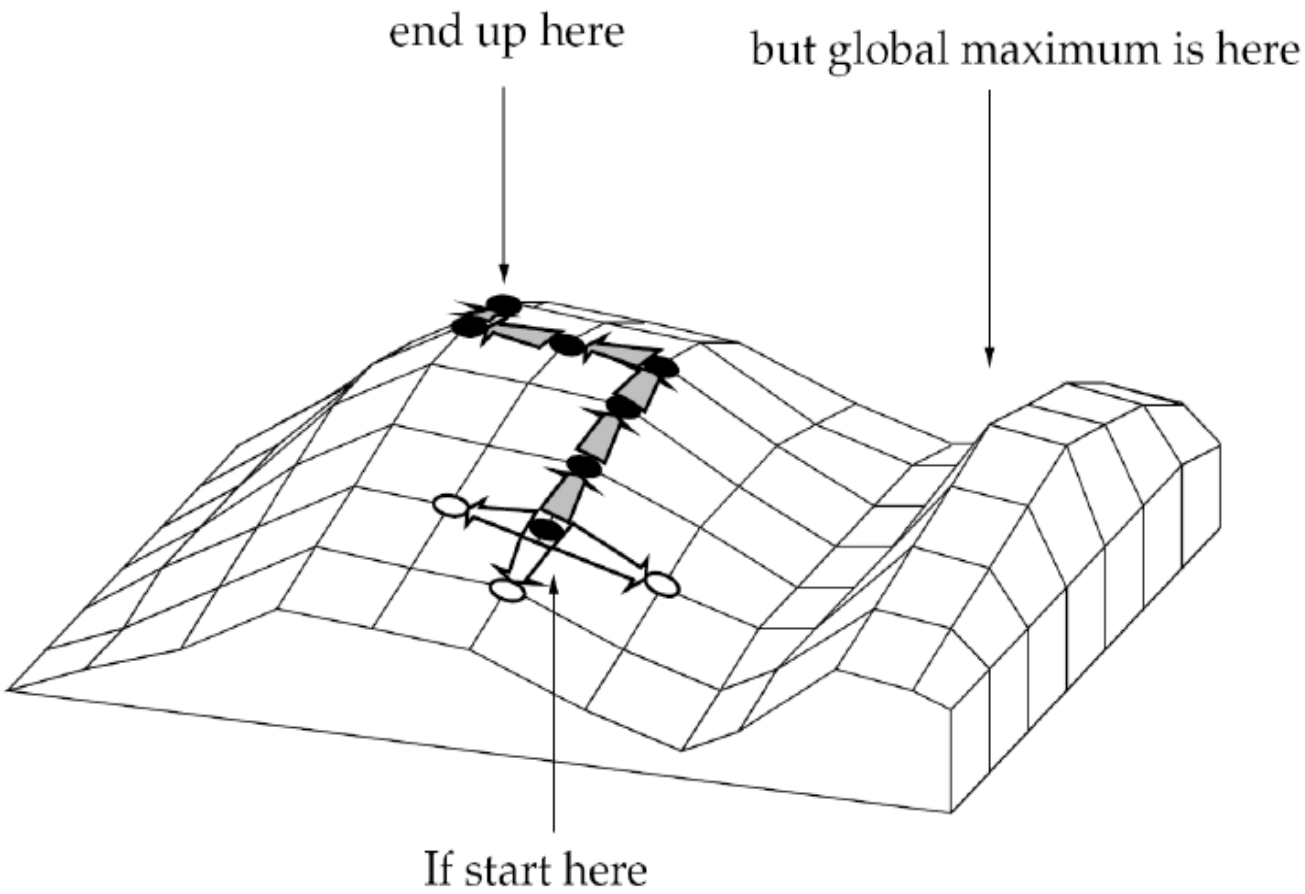
启发式搜索

从某一个随机的进化树开始，寻找Maximum Likelihood。进行物种逐步添加，以及进行进化树拓扑结构的变换（剪接和重排）
从多个随机的进化树开始，有助于寻找到全局最优进化树。

进化树的剪接和重排


full tree
Nearest Neighbor Interchange
linearly many NNI trees
Subtree Pruning + Regrafting
quadratically many SPR trees
Tree-Bisection + Reconnection
cubic number of TBR trees

贪婪算法(Greedy Algorithm)的误区



可能滞留在局域最优进化树上

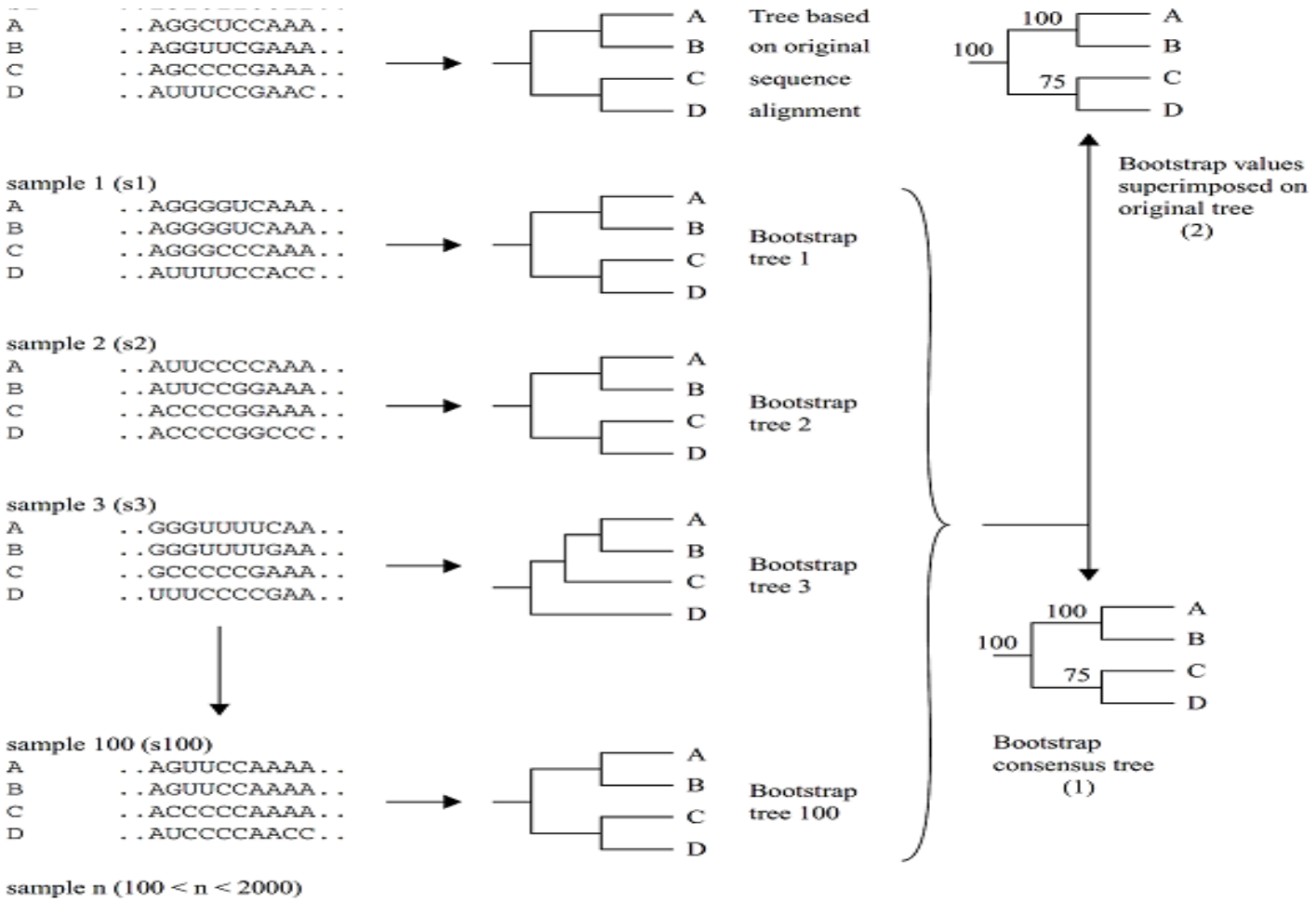
基于启发式搜索寻找全局最优树的方法

- 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

用ML法构建进化树的软件

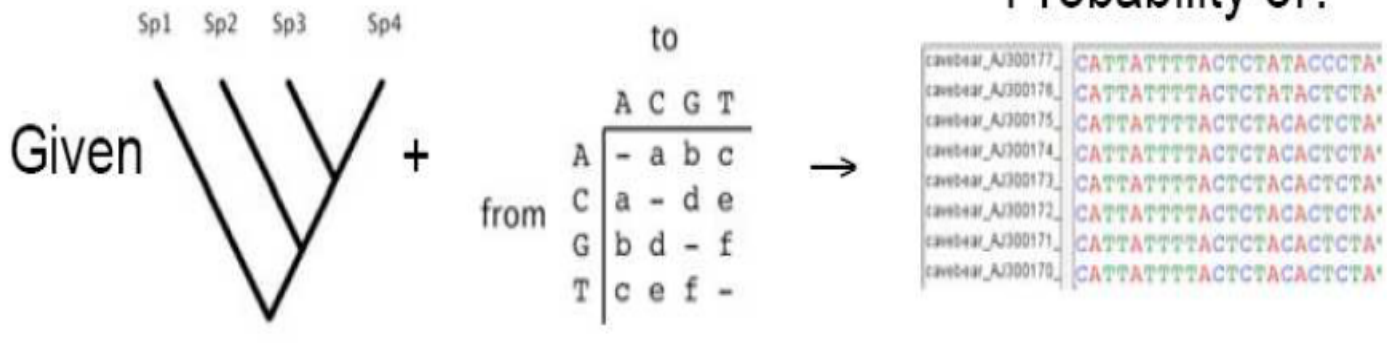
- PHYLIP: 由美国Joe Felsenstein编写，开源软件，也是第一个用极大似然法搜索进化树的软件，适用于进化树入门学习和小规模数据的进化树建立
- PAUP*: 由美国David Swoford编写，特别适用于最大简约法，但是也包括极大似然法以及各种搜索以及进化树剪接，Bootstrap等。还可以结合ModelTest进行进化模型筛选。但是需要购买版权。
- PHYML：由法国的Guidon编写，适用于大规模数据（大量分类单元或大量信息位点）。支持多种模型的筛选。
- RAxML：由瑞士Stamatakis编写只支持DNA的GTR模型以及蛋白质的模型，适用于分类单元数较多，或者位点数较多时，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。

进化树的Bootstrap与节点支持率

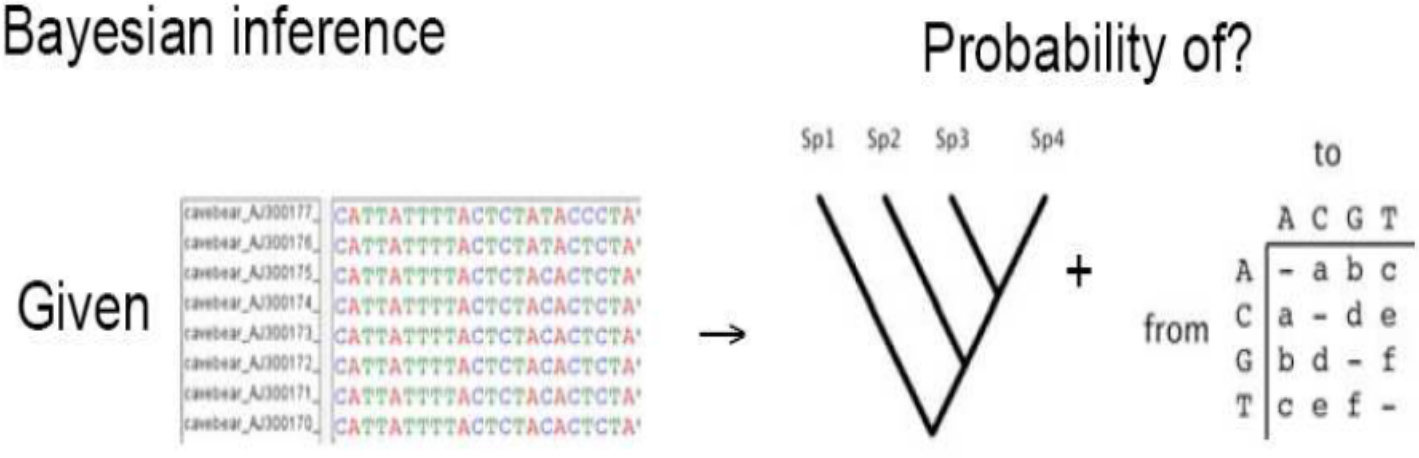


贝叶斯方法推断进化树

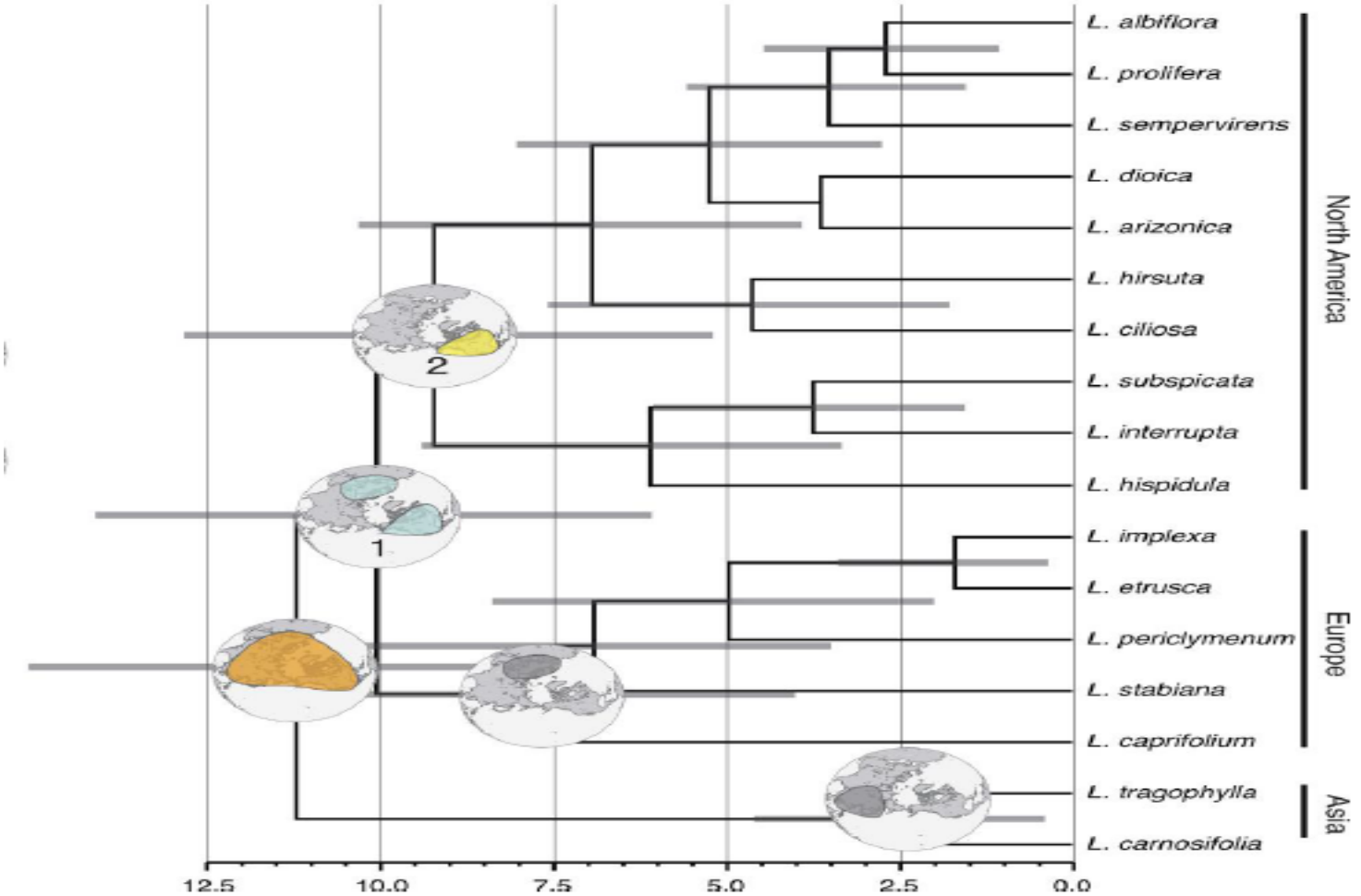
Maximum likelihood



Bayesian inference



贝叶斯方法建树



BEAST以及MrBayes是用贝叶斯方法建立进化树的最重要软件。
BEAST建立的进化树同时进行了分子钟校对。

分子钟：枝长的意义

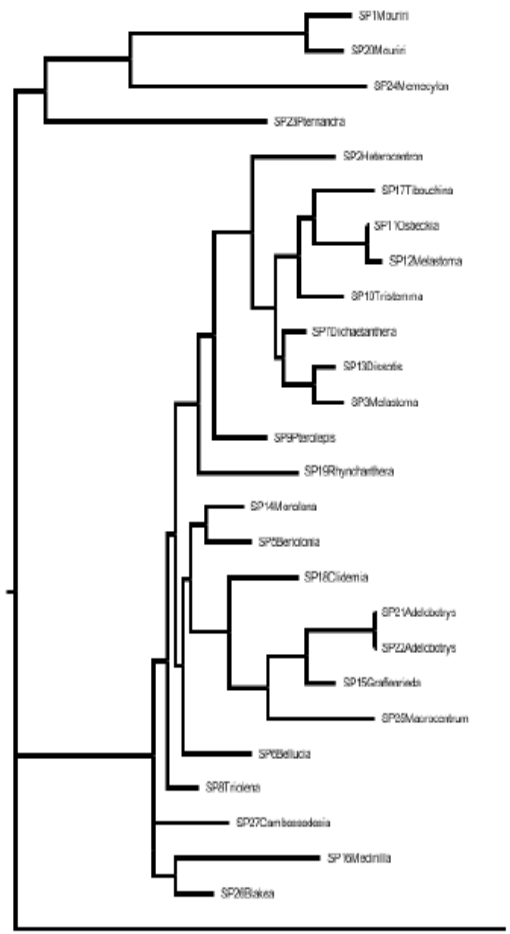
- 距离法: 枝长表示遗传距离
- 简约法: 枝长表示发生变异的位点数
- ML和贝叶斯法：枝长表示位点的变异。

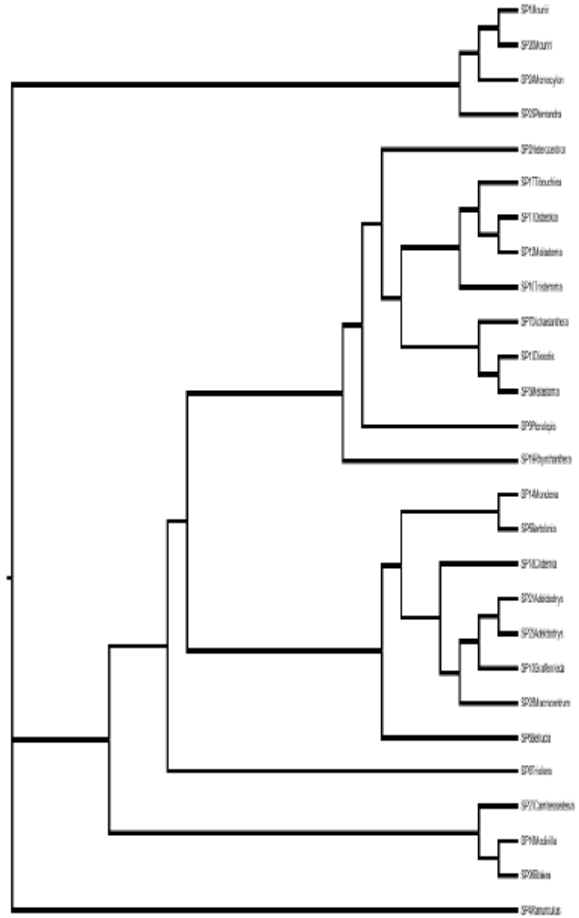
对于ML进化树来讲，如果枝长较长，则可能是由于碱基替换速率高，或者分化的时间足够长。为了让枝长表示分化时间，需要进行分子钟校订。

分子钟校订的软件

- r8s: 非参数速率平滑 NPRS；似然罚分法： Penalized Likelihood
- BEAST：构建进化树时直接进行了分子钟校订。
- ape程序包： PL方法等
- multidivide time: 基于贝叶斯的方法

分子钟校订前后





左图：校订之前
右图：校订之后

练习

- 用MUSCLE比对序列
- 用phylotools转换成Relaxed Phylip
- 用RAxML构建ML进化树
- 用Figtree将RAxML Best Tree转换成Nexus格式
- 用r8s进行分析钟校订

目录

APE: Analysis of Phylogenetics and Evolution

APE 是法国 E. Paradis 等人编写的用于进化分析的程序包
http://ape-package.ird.fr/ape_authors.html
该程序包定义了存储进化树信息的phylo类，成为进行系统发育比较分析的基础。
ape provides functions for:

- reading and manipulating phylogenetic trees and DNA sequences,
- computing DNA distances,
- estimating trees with distance-based methods,
- a range of methods for comparative analyses and analysis of diversification.
- Functionalities are also provided for programming new phylogenetic methods.

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

读取进化树 read.tree

该函数用来读取newick格式的进化树，并生成phylo类型的对象。
phylo对象的本质是一个列表 list 带有四个向量，分别为 edge: 表示分支之间的拓扑关系; Nnode：内部的节点数量; tip.label：分类单元的名称; edge.length：枝长

```
> library(ape)
> s <- "owls(((Strix_aluco:4.2,Asio_otus:4.2):3.1,Athene_noctua:7.3):6.3,Tyto_alba:13.5);"
> cat(s, file = "ex.tre", sep = "\n")
> tree.owls <- read.tree("ex.tre")
> 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" "Athene_noctua" "Tyto_alba"
 $ 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"
```

绘制进化树 plot.phylo

由于读取的进化树以phylo的格式保存，因此，基于phylo格式开发的plot.phylo函数，可以用于绘制进化树。该函数可以直接用 plot()（泛型函数）来调用。

```
## S3 method for class 'phylo'
plot(x, type = "phylogram", use.edge.length = TRUE,
node.pos = NULL, show.tip.label = TRUE, show.node.label =
FALSE, edge.color = "black", edge.width = 1, edge.lty = 1,
font = 3, cex = par("cex"), adj = NULL, srt = 0, no.margin
= FALSE, root.edge = FALSE, label.offset = 0, underscore =
FALSE, x.lim = NULL, y.lim = NULL, direction =
"rightwards", lab4ut = NULL, tip.color = "black", plot =
TRUE, rotate.tree = 0, open.angle = 0, node.depth = 1,
align.tip.label = FALSE, ...)
```

保存绘制的进化树

```
### An extract from Sibley and Ahlquist (1990)
tiff(filename = "tree.tiff", width = 2400, height = 2400,
units = "px", pointsize = 12, res = 600, compression
="lzw")
cat("(((Strix_aluco:4.2,Asio_otus:4.2):3.1,",
"Athene_noctua:7.3):6.3,Tyto_alba:13.5);", file = "ex.tre",
sep = "
n")
tree.owls <- read.tree("ex.tre")
plot(tree.owls)
dev.off()
```

ape中的重要函数 I

- ace() Ancestral Character Estimation
- ladderize() Ladderize a Tree
- bd.ext() Extended Version of the Birth-Death Models to Estimate Speciation and Extinction Rates
- chronopl() Molecular Dating With Penalized Likelihood
- consensus() Concensus Trees
- di2multi() Collapse and Resolve Multichotomies
- drop.tip() Remove Tips in a Phylogenetic Tree

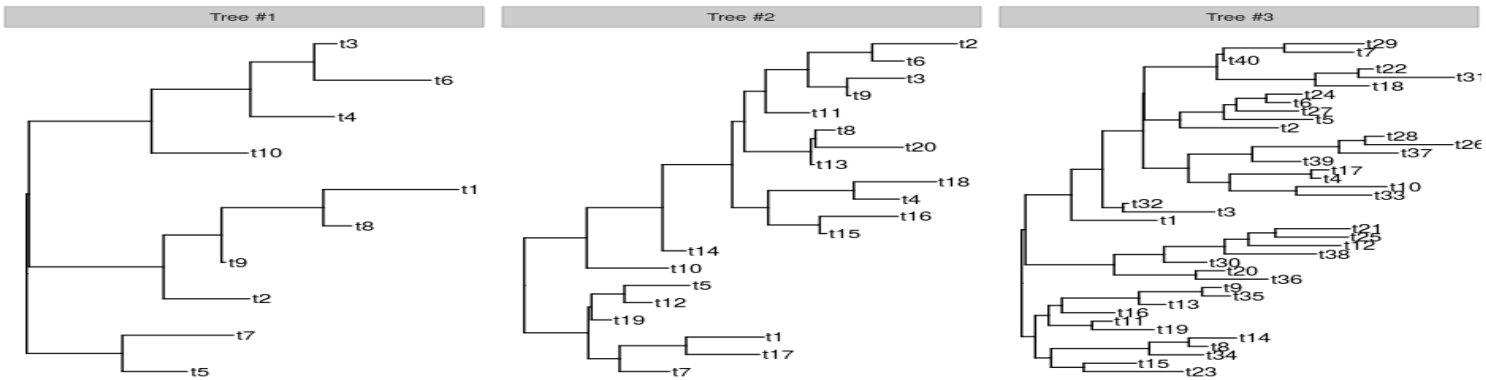
ape中的重要函数 II

- gammaStat() Gamma-Statistic of Pybus and Harvey
- ladderize() Ladderize a Tree
- ltt.plot() Lineages Through Time Plot
- mrca() Find Most Recent Common Ancestors Between Pairs
- multi2di() Collapse and Resolve Multichotomies
- read.nexus() Read Tree File in Nexus Format
- root() Roots Phylogenetic Trees
- rtree() Generates Random Trees
- write.nexus() Write Tree File in Nexus Format
- write.tree() Write Tree File in Parenthetic Format

ggtree是绘制进化树的程序包

ggtree是香港大学公共卫生学院在读博士生余光创先生编写的R程序包。ggtree将进化树看做点拓扑关系的集合，绘制进化树的线段是后期处理获得。 ggtree是ggplot2程序包的良好扩展，能够很好得用图层方式的语法对对象修改，因此，通过较为简单的参数，即可完成十分复杂的进化树绘制。

ggtree提供了多种进化树格式的读取，并且能够读取并转换BEAST, r8s, RAxML等软件所生成的数据。用于也可自定义数据，轻松实现R实现基于进化树的高级绘图。



ggtree 举例

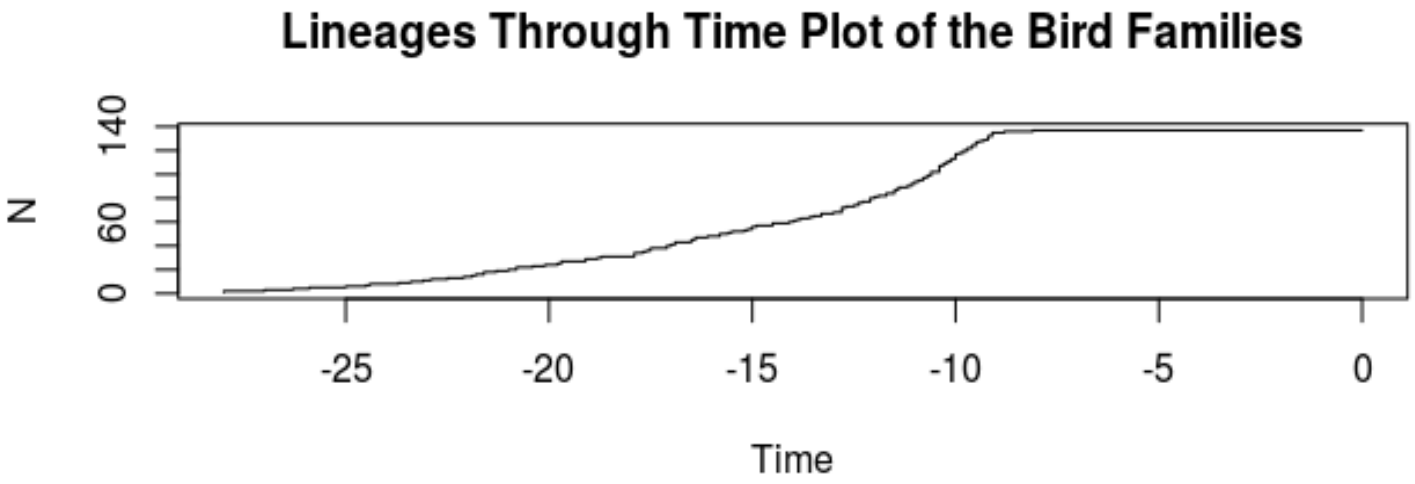
```
trees <- lapply(c(10, 20, 40), rtree)
class(trees) <- "multiPhylo"
ggtree(trees) + facet_wrap(~id, scale="free") +
geom_tiplab()
```

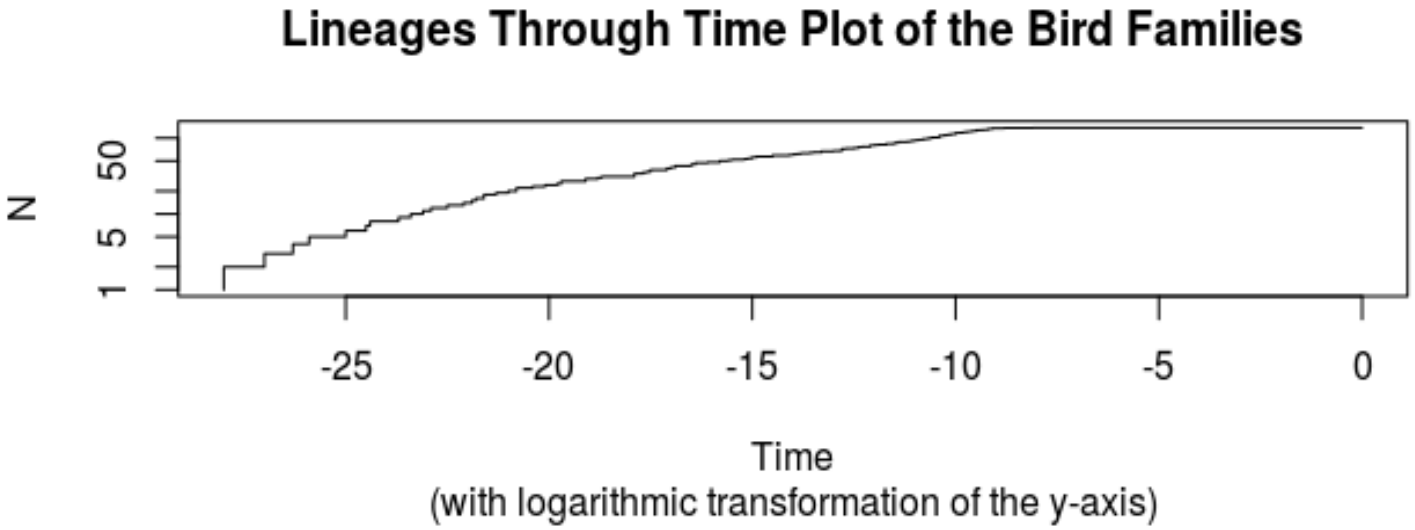
可参考 http://blog.sciencenet.cn/home.php?mod=space&uid=255662&do=blog&id=969228

目录

Lineages Through Time Plot

ltt plot





Laser：基于ltt plot估计物种形成和灭绝速率

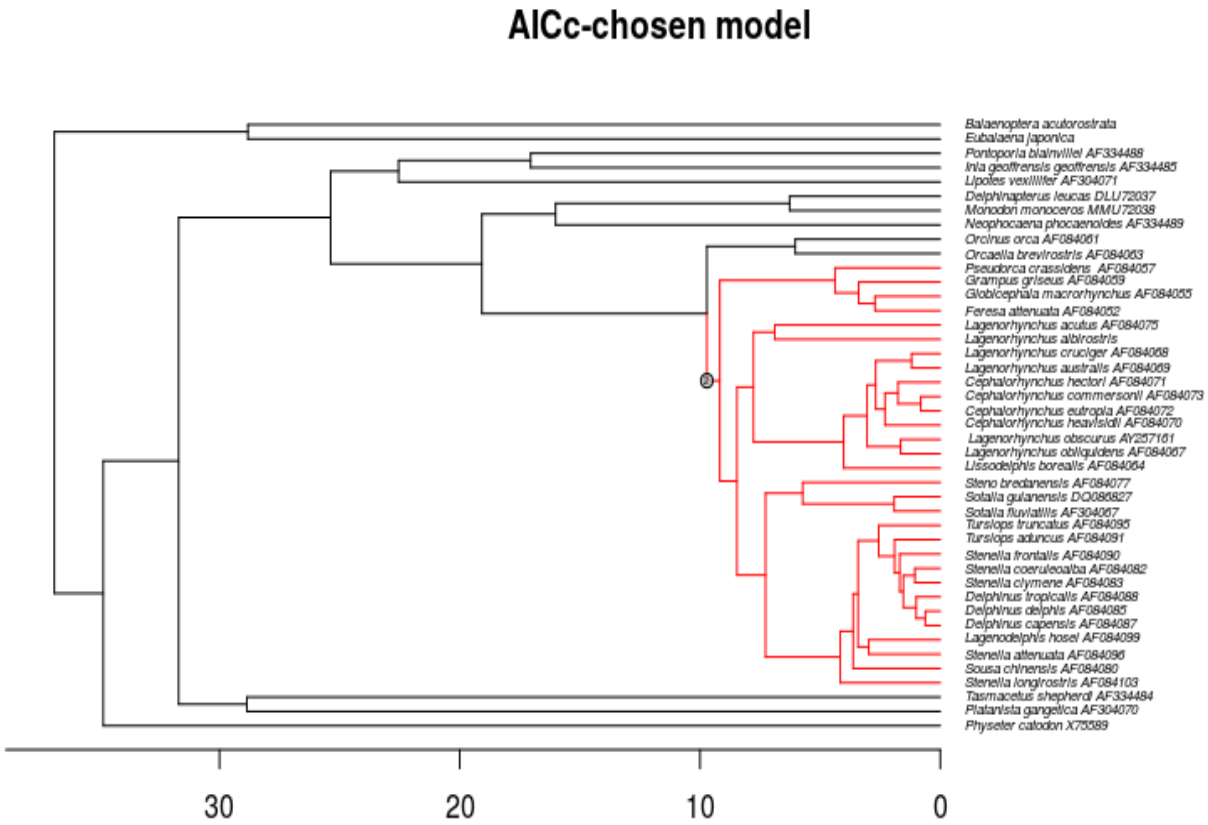
Laser 程序包由 Dan Rabosky开发。是用极大似然法进行物种形成和灭绝速率进行估计。
包括三种模型:
1. 物种形成速率随时间的变化而降低，但灭绝速率恒定（SPVAR）
2. 物种形成速率恒定，但是物种灭绝速率随时间上升（EXVAR）
3. 物种形成随时间下降，同时物种的灭绝速率上升（BOTHVAR）

Laser: Likelihood Analysis of Speciation/Extinction Rates from Phylogenies

```r
fitSPVAR <- function (bt, init = c(2, 0.2, 0.1))
{
    STIMES <- getSpeciationtimes(bt)
    TMAX <- max(bt)
    getLam0.SPVAR <- function(x) {
        k <- x[2]
        mu0 <- x[3]
        lam0 <- (x[1] + mu0) * exp(TMAX * k)
        return(lam0)
    }
    optimLH.SPVAR <- function(init) {
        k <- init[2]
        mu0 <- init[3]
        lam0 <- (init[1] + mu0) * exp(TMAX * k)
        LH <- getLikelihood.SPVAR(lam0, k, mu0, STIMES, TMAX)
        return(-LH)
    }
    low <- c(0.001, 0.001, 0.001)
    up <- c(Inf, Inf, Inf)
    temp <- optim(init, optimLH.SPVAR, method = "L-BFGS-B", lower = low,
        upper = up)
    res <- list(model = "SPVAR", LH = -temp$value, aic = (2 *
        temp$value) + 6, lam0 = getLam0.SPVAR(temp$par), k = temp$par[2],
        mu0 = temp$par[3])
    return(res)
}
```

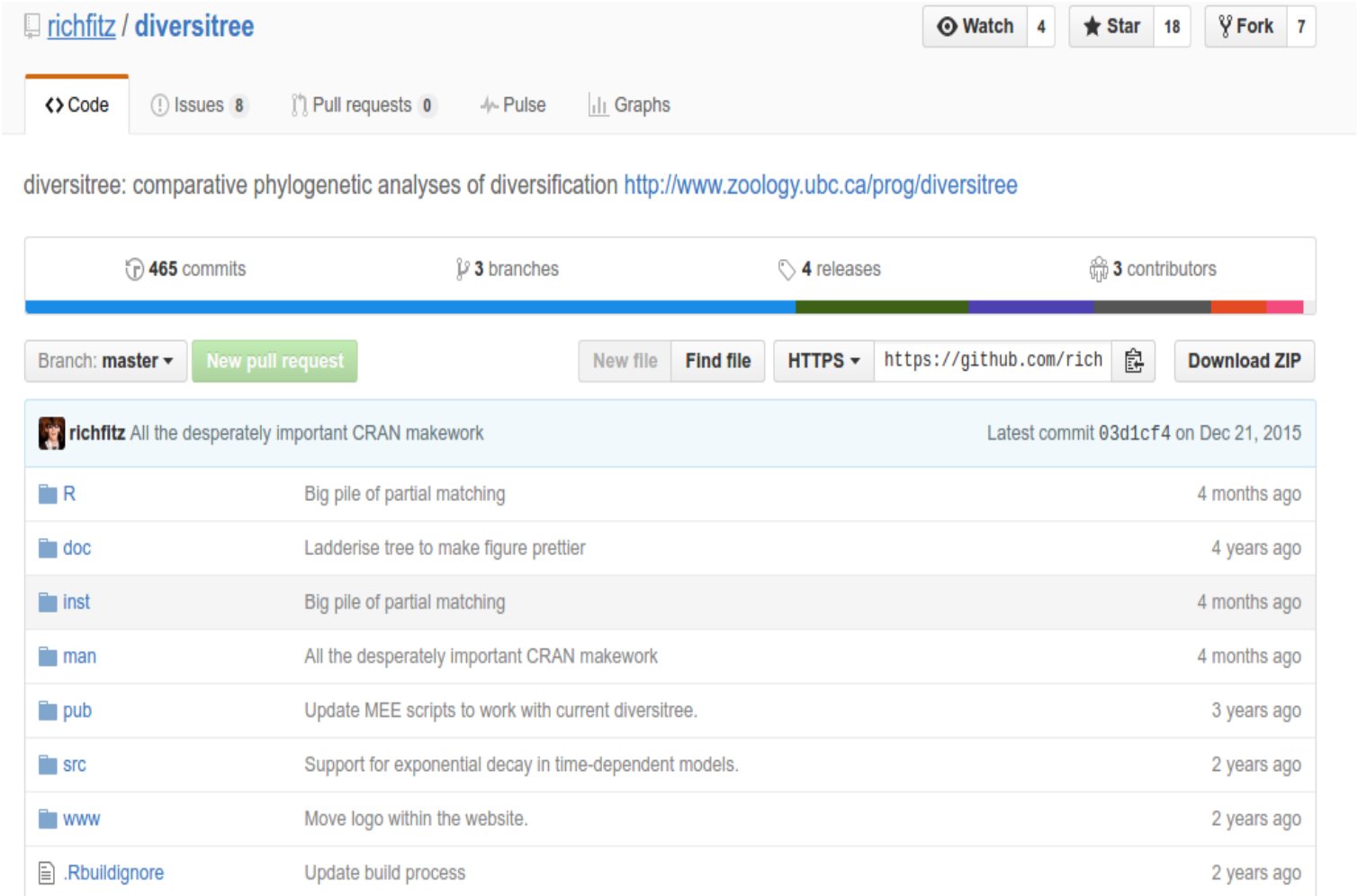
Geiger分析物种分化速率 MEDUSA

Modeling Evolutionary Diversification Using Stepwise AIC



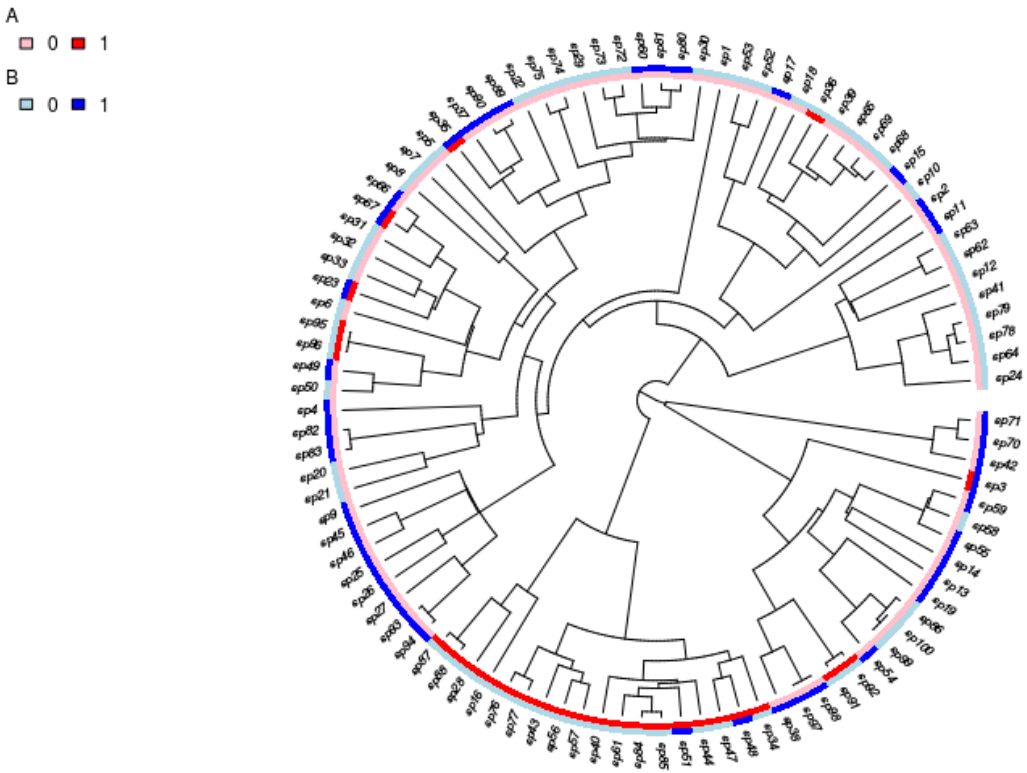
geiger::medusa

diversitree 安装

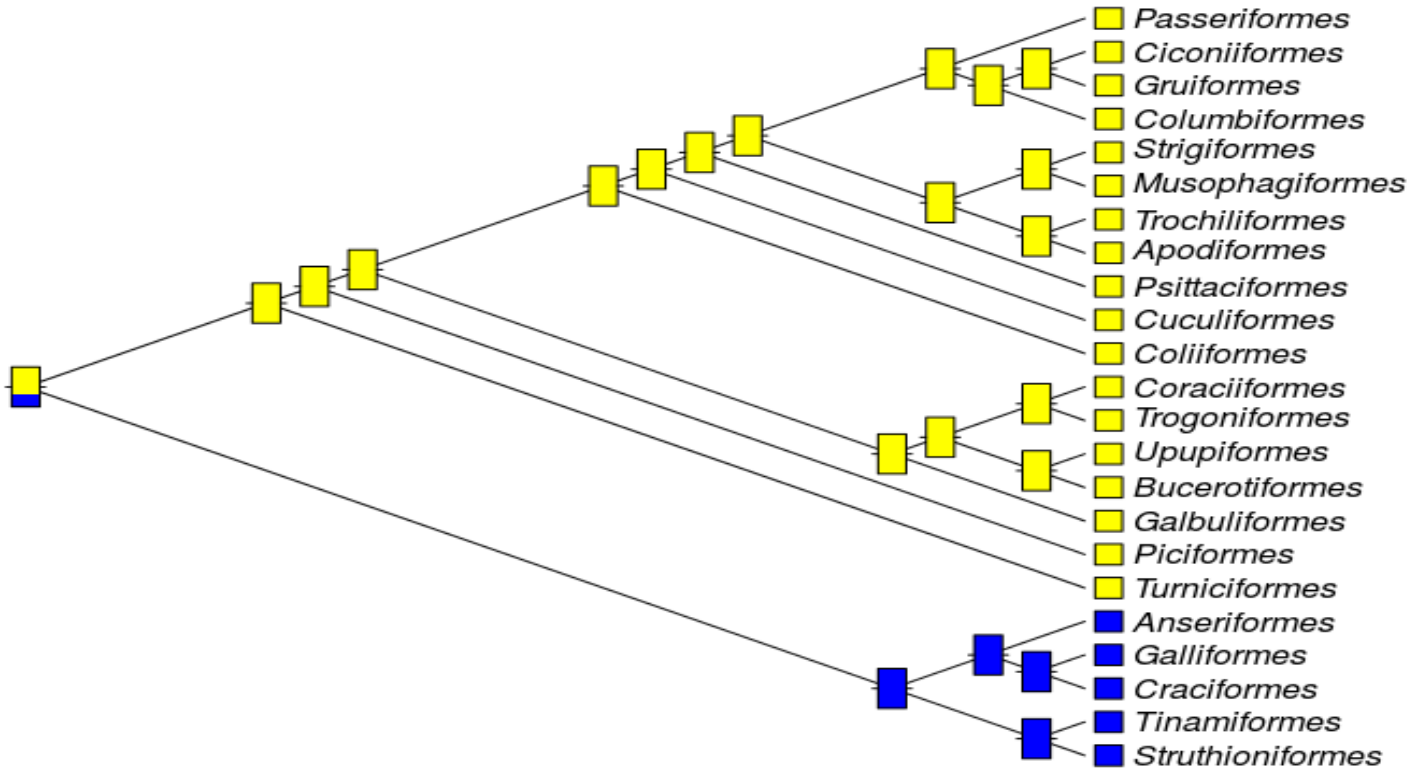


install.packages("diversitree")

diversitree



重建祖先状态 Ancestral Character Estimation



ape::ace

目录

系统发育多样性

- Faith's PD Phylogenetic Diversity (Faith 1992) 连接某地点所有出现物种的枝长总和。 picante::pd()
- 系统发育信号（Blomberg's K）：检验系统发育相近的分类单元是否具有相近的性状。picante::phylosignal()
- 进化的独特性 Evolutionary Distinctiveness: 每个进化树中，每个分类单元所独占的系统发育信息。 picante::evol.distinct()

系统发育beta多样性

系统发育beta多样性： Phylogenetic Beta diversity
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)

群落系统结构中的零模型 Null Models

群落系统发育分析中零模型，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，因此，群落中物种应该是随机组合的。
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)。

目录

分类位置查询

程序包主要为 taxize, taxonstand, plantlist

- status(): Looking for the taxonomic status of species
- data(acc_dat): The accepted plant names from the Plant List
- data(syn_dat): Synonyms database from The Plant List
- taxa.table(): Making a Taxa Table based on the result of function TPL for Phylomatic
- TPL(): Looking up Family, Family Number and Order in Modern Classification Systems.

用plantlist程序包查询植物科属

```
> library(plantlist)
This is plantlist 0.2.5.
> ?TPL
> TPL("Carex")
  YOUR_SEARCH POSSIBLE_GENUS     FAMILY  ORDER FAMILY_NUMBER
1       Carex          Carex Cyperaceae Poales    APGIII_099
> TPL("Apple")
  YOUR_SEARCH POSSIBLE_GENUS FAMILY ORDER FAMILY_NUMBER
1       Apple          Apple   <NA>  <NA>          <NA>
> splist <- c( "Ranunculus japonicus",
+              "Solanum nigrum",
+              "Punica sp.",
+              "Machilus", "Today", "####" )
> res <- TPL(splist)
> res
```

plantlist::TPL

参考材料

https://cran.r-project.org/web/views/Phylogenetics.html

问题？

谢谢!
敬请批评指正