

Phylogenetic tree

2020.12.04

目录

- ▶ 系统发育树的基础
- ▶ 系统发育树的构建
- ▶ 系统发育树的解读
- ▶ 如何使用本地工具构建系统发育树（mega）
- ▶ 如何使用在线工具构建系统发育树（cipres）
- ▶ 如何使用本地工具编辑展示树文件（figtree）
- ▶ 如何使用在线工具编辑展示树文件（itol）
- ▶ 作业题

1.系统发育树基础

1.基础知识

▶ 系统发育与进化

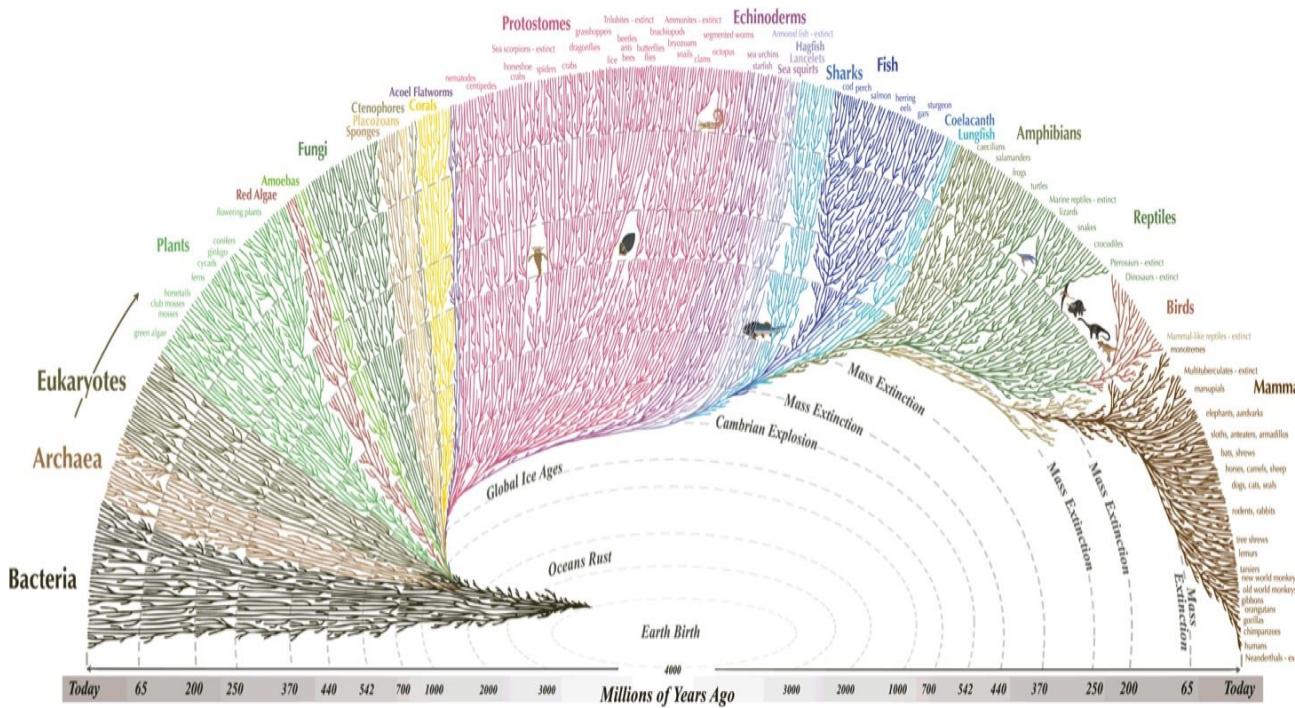
- ▶ **Phylogeny**, the history of the evolution of a species or group, especially in reference to lines of descent and relationships among broad groups of organisms.
- ▶ **Phylogenetics** is a part of systematics that addresses the inference of the evolutionary history and relationships among or within groups of organisms. These relationships are hypothesized by phylogenetic inference methods that evaluate observed heritable traits, such as DNA sequences or morphology, often under a specified model of evolution of these traits.
- ▶ 系统发生学是研究生物个体或群体（例如物种或种群）之间的演化历史和关系。这些关系是通过系统发育推理方法被发现，这些方法评估被观察到的可遗传性状，例如在这些性状的进化模型下的DNA序列或形态。

1. 基础知识

▶ 什么是系统发育树

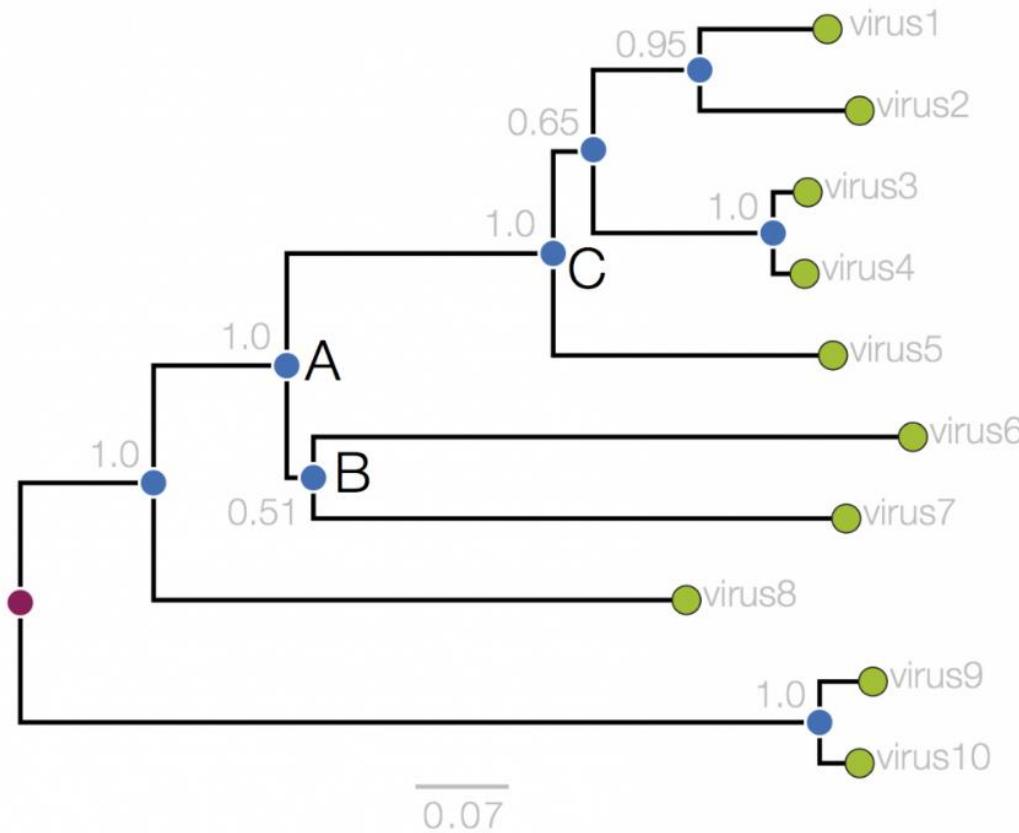
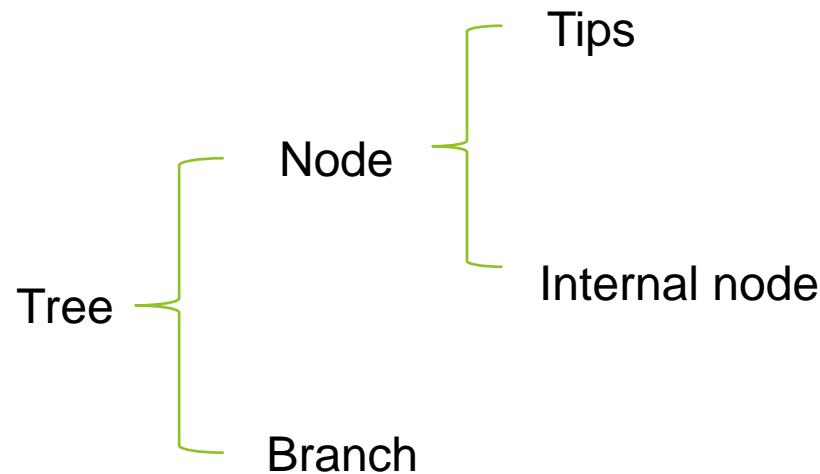
系统发育树又称为进化树，演化树，是表明被认为具有共同祖先的各物种间演化关系的树。在树中每个节点代表其各个分支的最近共同祖先，而节点的线段长度对应了其演化的距离。

http://en.wikipedia.org/wiki/Phylogenetic_tree



1. 基础知识

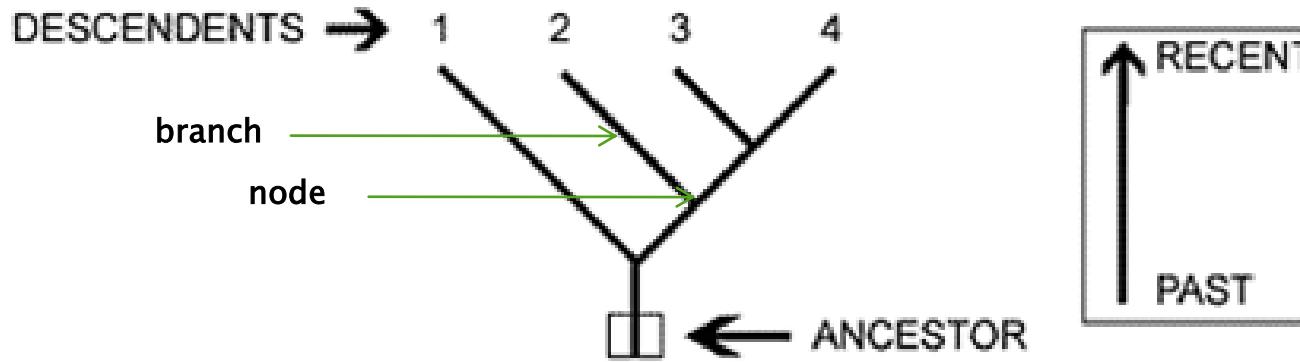
► 1.1 拓扑结构



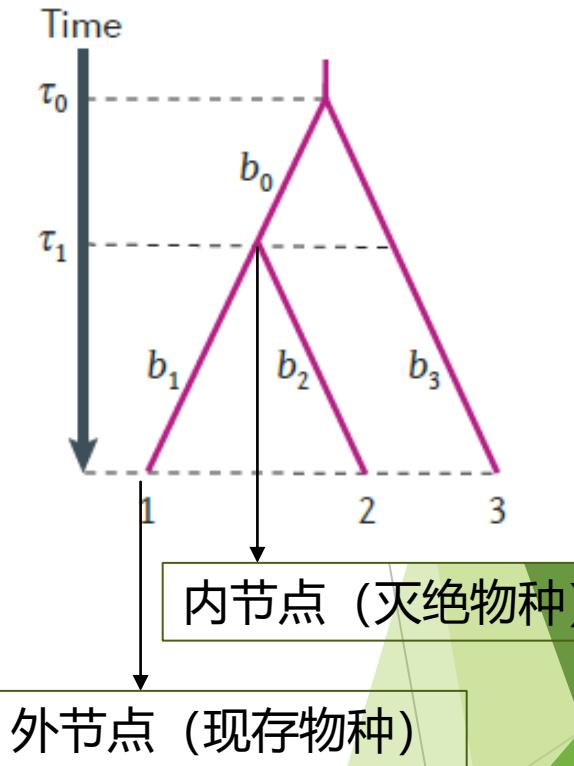
- 树结构本身可以分解为节点(在树中表示为圆圈)和分支(连接它们的线)。
- 节点分为两种类型：Tips(图中绿色圆圈)和非Tips(分支节点，图中蓝色圆圈)。
- Tips代表实际的生物样本和它们的序列，是现实存在的数据，除了实际的序列还有其他相关信息，比如时间、地点、物种等。
- 非Tips的Nodes代表现有序列的假定祖先。

1. 基础知识

► 1.1 拓扑结构



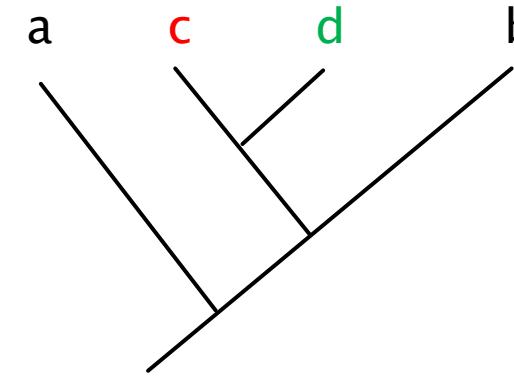
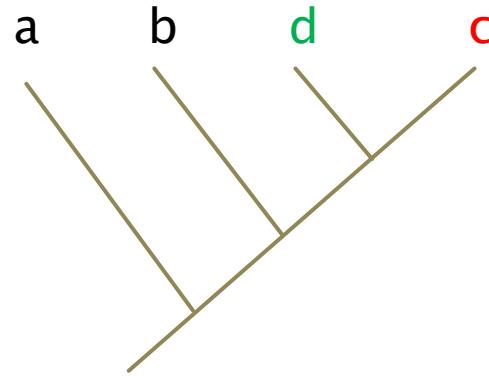
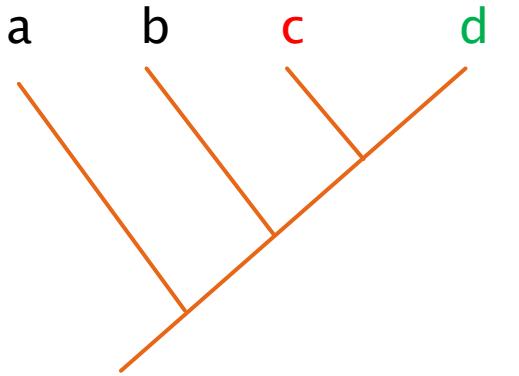
a Rooted tree



- 其中我们把从同一个节点上分出的两个分支叫做sister group.
- Sister group 从结构上可以理解为从进化史上看两者非常接近，其次两者拥有唯一的共同的祖先。
- 进化树的子树称clade

1.基础知识

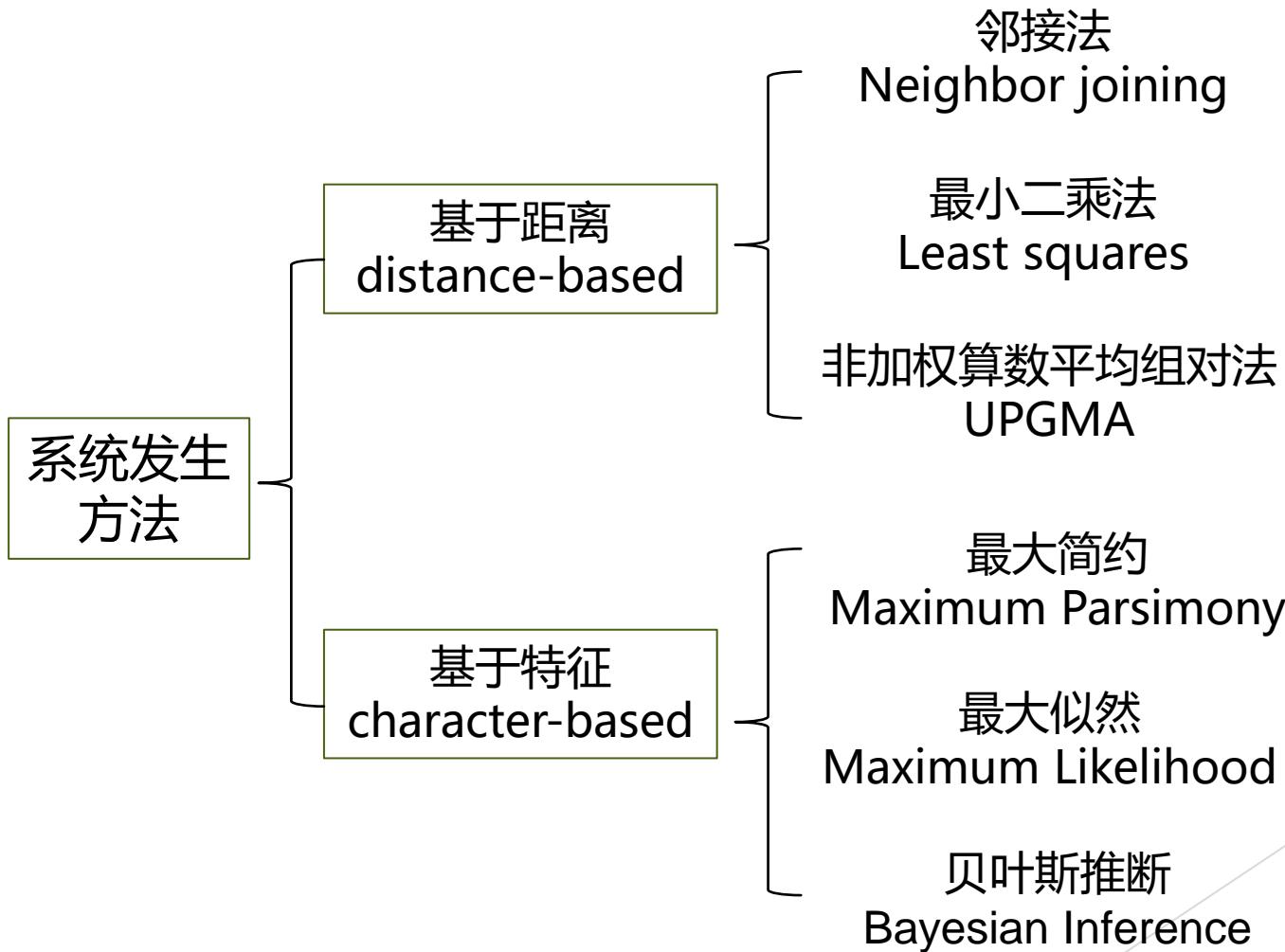
▶ 1.1 拓扑结构



•从结构上看，我们认为这三个树是等价的

1. 基础知识

► 1.2 系统发育树构建方法



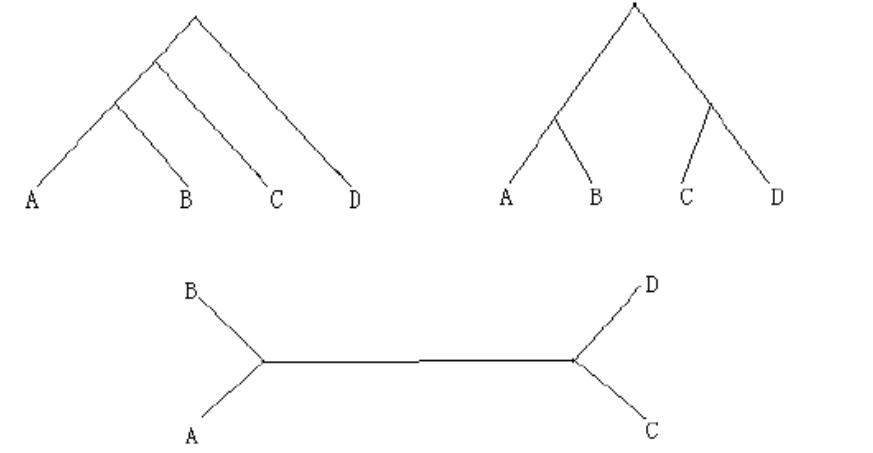
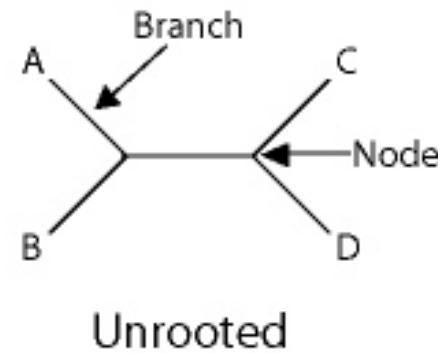
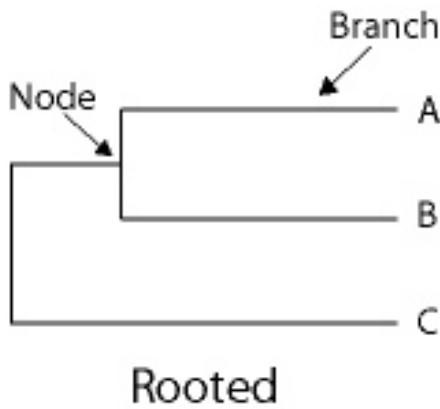
▶ 各个方法比较

方法	基本特征	适用范围	优点	缺点
NJ	不需要分子钟假设，是基于最小进化原理，进行类的合并时，不仅要求待合并的类是相近的，而且要求待合并的类远离其他的类。	远缘序列，但进化距离不大，信息位点少的短序列。	假设少，树的构建相对准确，计算速度快，只得一颗树，可以分析较多的序列，运行速度优于最大简约法	序列上的所有位点等同对待，且所分析的序列的进化距离不能太大
MP	基于进化过程中碱基替代数目最少这一假说，不需要替代模型，对所有可能的拓扑结构进行计算，并计算出所需替代数最小的那个拓扑结构，作为最优树	近缘序列物种序列的数目 ≤ 12 。残基差别少，具有近似的具有近似的变异率，包含信息位点比较多的长序列。	善于分析某些特殊的分子数据如插入、缺失等序列有用。	只适于序列数目 $N \leq 12$ 。存在较多回复突变或平行突变时，结果较差。推测的树不是唯一的，变异大的序列会出现长枝吸引而导致建树错误。
ML	依赖于某一个特定的替代模型来分析给定的一组序列数据，使得获得的每一个拓扑结构的似然率都为最大值，然后再挑出其中似然率最大的拓扑结构作为最优树。	特定的替代的模型，有模型有模型的情况下ML是与进化事实最吻合的树。	很好的统计学基础，大样本时似然法可以获得参数统计的最小方差，在进化模型确定的情况下，ML法是与进化事实吻合最好的建树算法。	所有可能的系统发育树都计算似然函数，计算量大，耗时时间长。依赖于合适的替代模型。
BI	基因进化模型的统计推论法，通过后验概率直观反映出各分支的可靠性，而不需要自举法检验。	大而复杂的数据集	具有坚实的数学和统计学基础，可以处理复杂和接近实际情况的进化模型	对进化模型敏感，BI法中指定的每个氨基酸的后验概率建立在许多假设条件下，现实中可能不成立。

1.基础知识

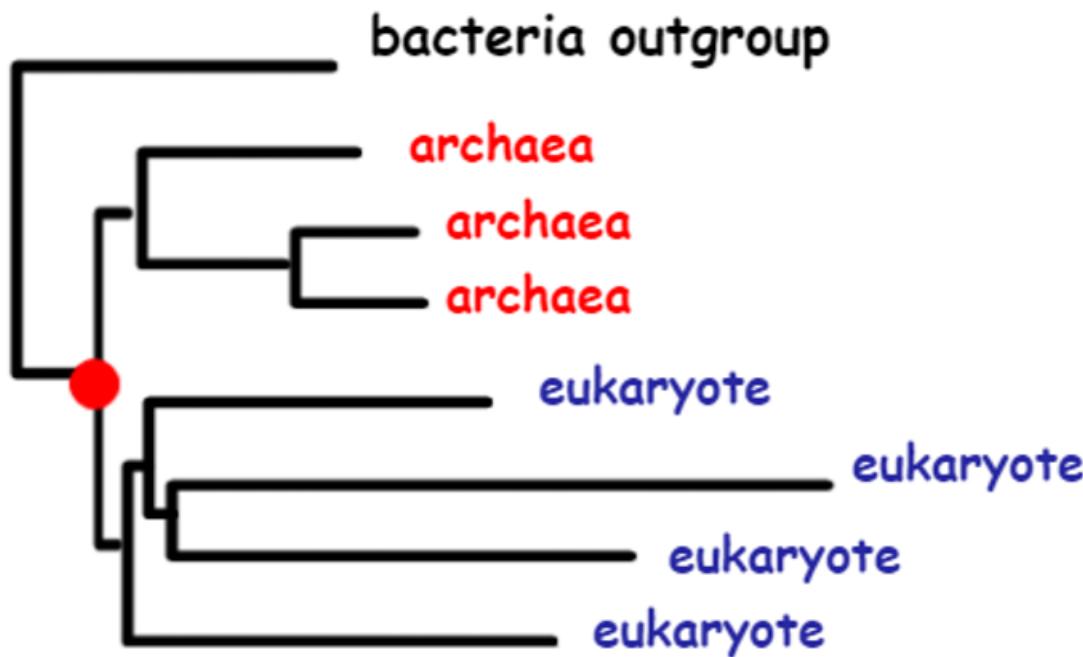
▶ 1.2 系统发育树的分类

- ▶ 系统发育树分有根(rooted)和无根(unrooted)树
- ▶ 有根树反映了树上物种或者基因进化的时间顺序,通过分析有根树的长度,可以了解不同的物种或者基因以什么方式和速率进化。
- ▶ 无根树只反映分类单元之间的距离,而不涉及谁是谁的祖先问题



1. 基础知识

▶ 1.2 系统发育树的分类



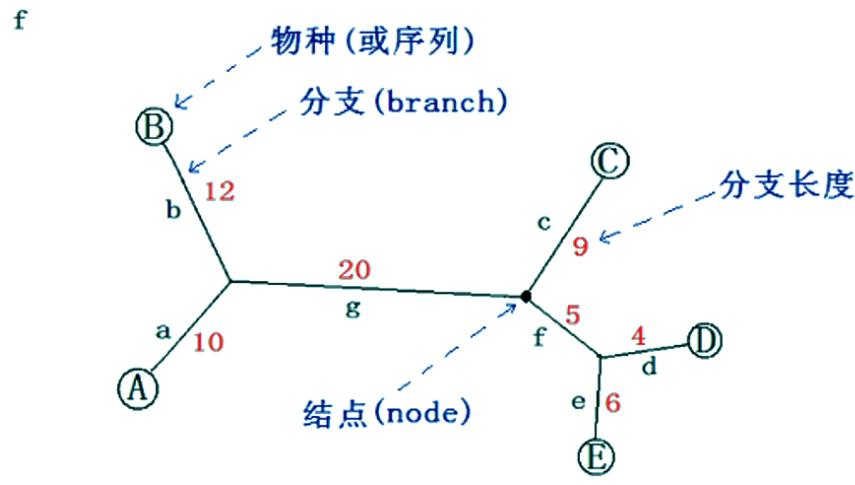
产生有根树的条件：

分子钟置根法：如果在所有时间内进化速率是恒定的，即假定存在分子钟

外类群置根法：在树重建中引入关系较远的物种，同时在对所有物种重建的无根树中，将树根置于连接外类群的枝，使得内类群的子树有根

1.基础知识

▶ 1.2 系统发育树的分类



5个物种的无根(unrooted)树

No. of taxa	No. of rooted trees	No. of unrooted trees
3	3	1
4	15	3
5	105	15
6	945	105
7	10395	945
...

- N个物种的无根树，有 $[(2N-5)*(2N-7)*\dots*1]$ 种结构；
- 每种结构的树都有 $(2N-3)$ 条分支， $(N-2)$ 个结点；
- 任何分支都可以看成是根；
- 连接两个物种的所有分枝长度之和为它们的距离。

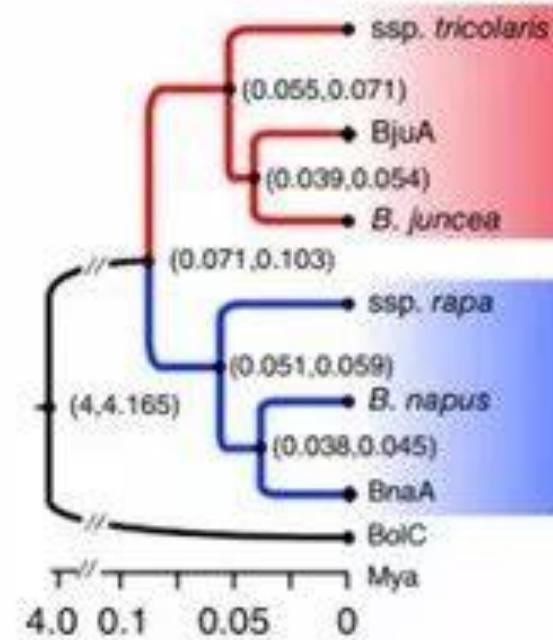
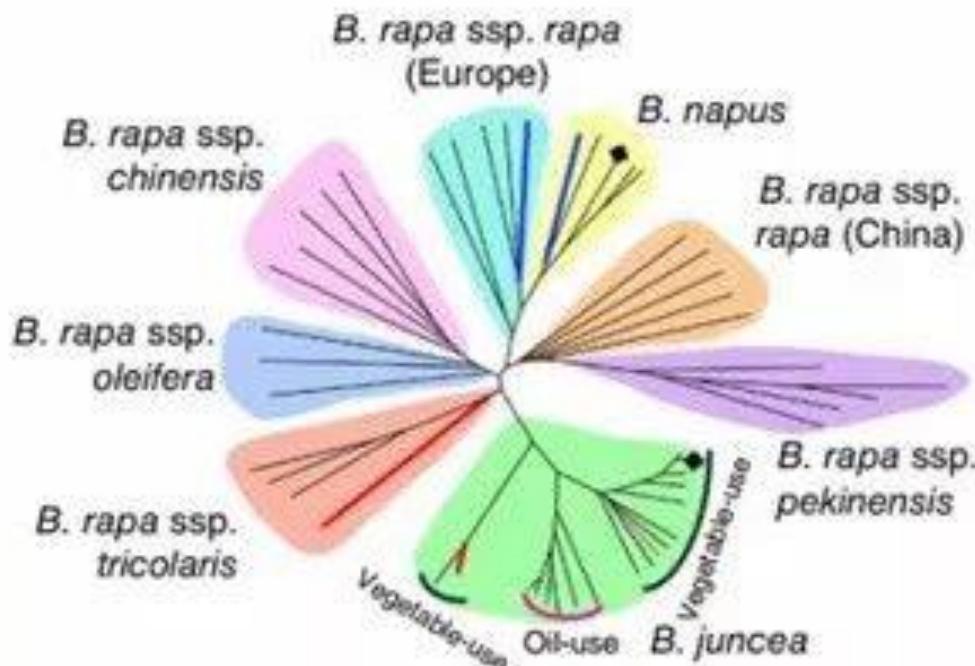
1. 基础知识

► 1.2 系统发育树的分类

► 标度树和非标度树

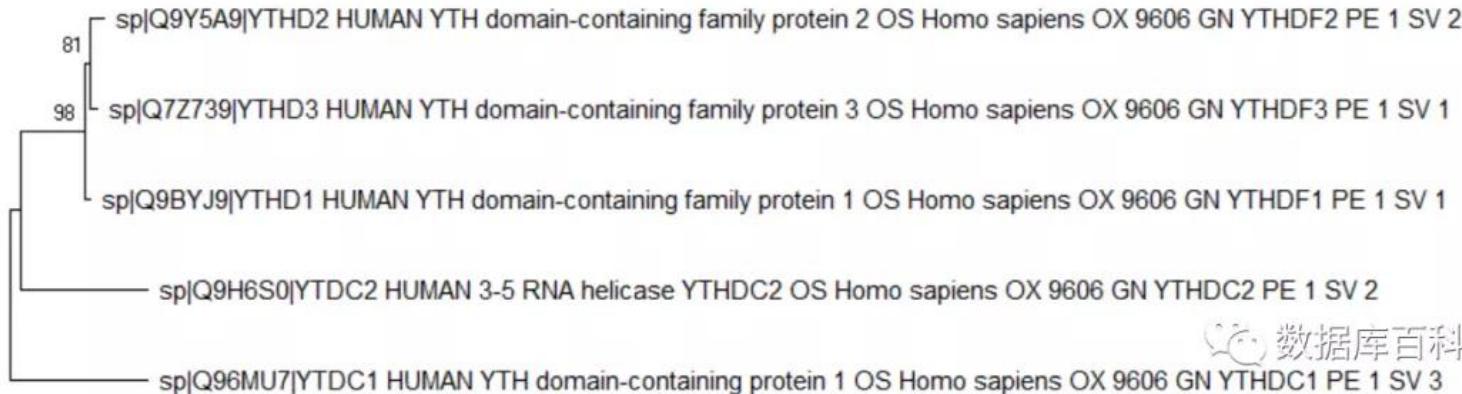
系统进化树还可以根据分支长度是否具有意义分为标度树和非标度树。

标度树的分支长度表示变化的程度，而非标度树的分支只表示进化关系，支长无意义。

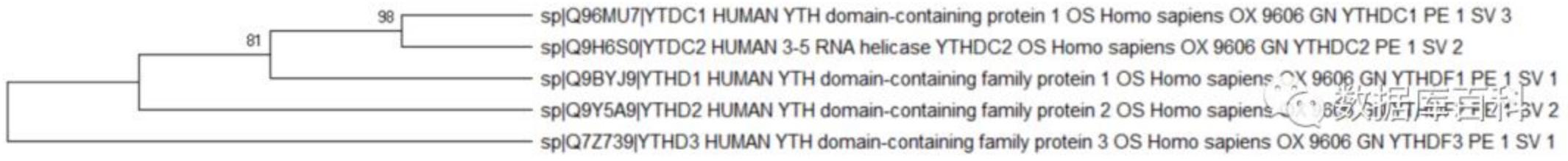


1.基础知识

► 1.2 系统发育树的分类



- Original Tree (原始树)，是步长检验构建的 1000 株树中的一株，未经过多棵
树合并，所以树枝的长短可以精确代表遗传距离，即进化的距离远近。



- Bootstrap consensus tree (步长检验合并出来的树)，只反映进化关系，树枝
的长短与遗传距离无关。

2. 系统发育树构建

2.构建步骤

▶ 2.1 系统发育树文件类型与结构

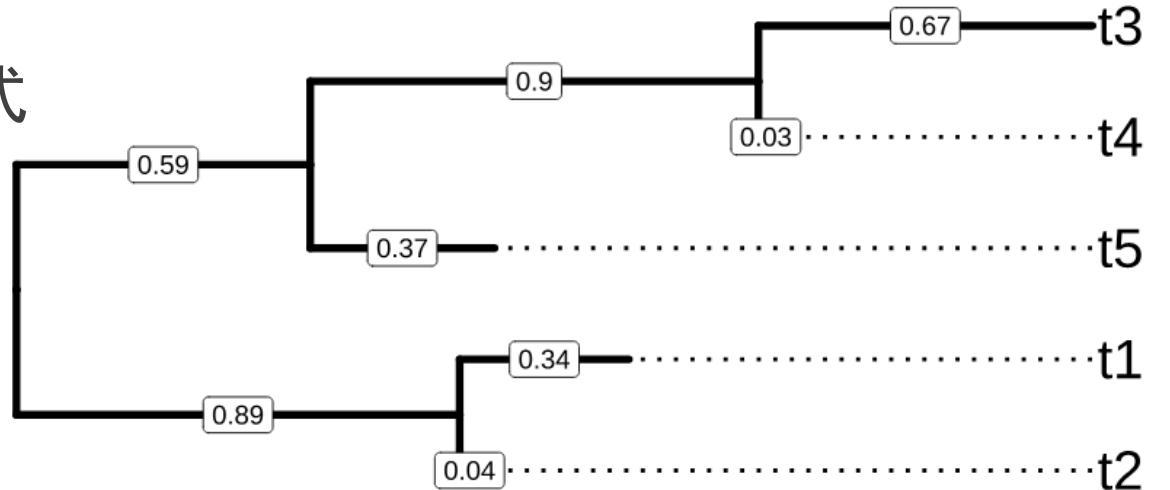
- ▶ Newick/Phylip (Felsenstein [1989](#))
 - ▶ NEXUS (Maddison et al. [1997](#))
 - ▶ NHX
 - ▶ Jplace
-
- 用于存储系统发育树节点和分支的数据的格式中，常用的两种格式是Newick(Phylip)和NEXUS，一些格式(例如NHX)是从Newick格式扩展而来的。
 - 大多数进化生物学软件支持Newick和Nexus格式作为输入，而一些软件工具输出更新的标准文件(例如BEAST)通过引入新的规则/数据块来存储进化树的数据。
 - 其他软件(例如PAML和r8s)，输出文件只能由它们自己的软件识别。

<https://yulab-smu.top/treedata-book/chapter1.html>

https://evolution.genetics.washington.edu/phylip/newick_doc.html

2. 构建步骤

▶ 2.1.1 Newick 格式



A sample tree for demonstrating Newick text to encode tree structure. Tips were aligned to right hand side and branch lengths were labelled on the middle of each branch.

```
((t2:0.04,t1:0.34):0.89,(t5:0.37,(t4:0.03,t3:0.67):0.9):0.59);
```

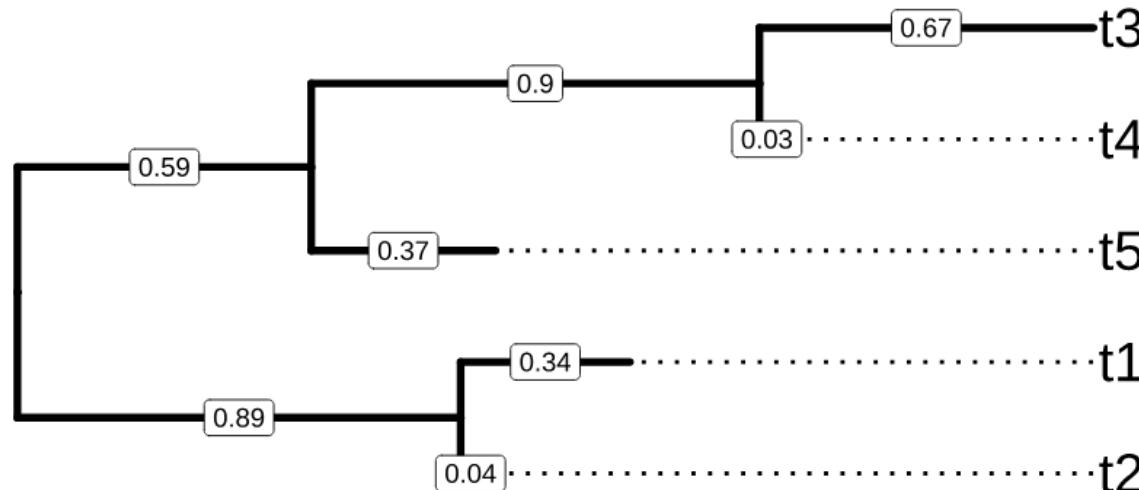
- ▶ 树文本以分号结尾。内部节点由一对匹配的括号表示。括号之间是该节点的后代节点。例如(t4:0.03,t3:0.67)表示t4和t3他们是直系后代。同级节点用逗号分隔，提示用名称表示。分支长度(从父节点到子节点)由子节点后面的实数表示，前面是冒号。注释数据与内部节点或分支相关联，可以编码为节点标签，并由冒号之前的简单文本/数字表示。

2.构建步骤

▶ 2.1.2 NEXUS 格式

- ▶ 1997年， Maddison 等人将Newick树中的文本等相关信息重组为单独的区块，形成了NEXUS格式。

Nexus文件以#NEXUS开头，文件采用“块”的方式来组织信息。每一个块的语法结构如右侧红框所示：



A sample tree for demonstrating Newick text to encode tree structure. Tips were aligned to right hand side and branch lengths were labelled on the middle of each branch.

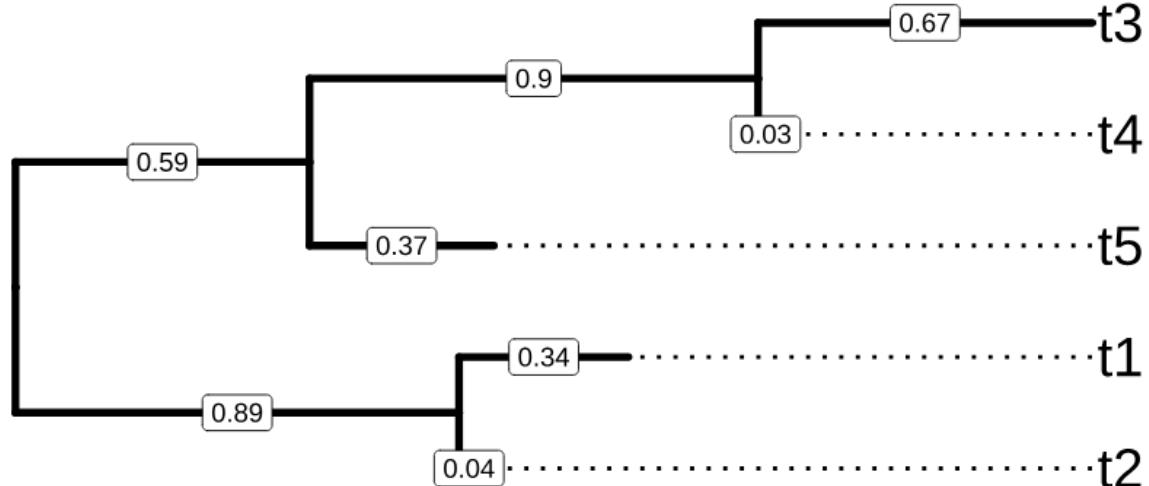
#NEXUS

...
BEGIN characters;
...
END;

<https://academic.oup.com/sysbio/article-lookup/doi/10.1093/sysbio/46.4.590>

2. 构建步骤

▶ 2.1.2 NEXUS 格式



A sample tree for demonstrating Newick text to encode tree structure. Tips were aligned to right hand side and branch lengths were labelled on the middle of each branch.

#NEXUS
[R-package APE, Wed Nov 9 11:46:32 2016]

```
BEGIN TAXA;  
DIMENSIONS NTAX = 5;  
TAXLABELS  
t5  
t4  
t1  
t2  
t3  
;  
END;  
BEGIN TREES;  
TRANSLATE  
1 t5,  
2 t4,  
3 t1,  
4 t2,  
5 t3  
;  
TREE * UNTITLED = [&R] (1:0.89,((2:0.59,3:0.37):0.34,  
(4:0.03,5:0.67):0.9):0.04);  
END;
```

2.构建步骤

▶ 2.1.2 NEXUS 格式

- ▶ 使用方括号[] 标识注释。大多数程序可以识别以下三类：TAXA(分类学数据), DATA(数据矩阵, 如MSA)和TREE(系统发育树, 如Newick)。
- ▶ 区块可以非常多样, 其中一些只被一个特定的程序识别。例如, 由PAUP导出的文件有一个PAUP块, 其中包含PAUP命令, 而Figtree导出Nexus文件有Figtree块, 其中包含可视化设置。Nexus将不同类型的数据组织到不同的块中, 而支持读取Nexus的程序可以解析它们识别的一些块, 而忽略它们无法识别的块。这种机制允许不同的程序在不受支持的数据类型出现时不会崩溃。
- ▶ DATA块常被用于存储MSA。为此, 用户可以以Phylip格式存储树和序列数据, 这两种数据实质上分别是Phylip多序列对齐和Newick树文本。

```
begin data;
dimensions ntax=5 nchar=600;
format interleave datatype=DNA missing=N gap=-;
end;
```

- ntax表示分类单位的数量, nchar表示数据的长度, 上面这个例子里的含义是DNA有600个碱基长度。数据类型, datatype是DNA。另外可以定义缺失信息和缺口用什么字符表示。

2. 构建步骤

Others

```
+-----Gibbon  
+---2  
|  
|     +-----Orang  
|  
|     +---4  
|         |     +-----Gorilla  
|         +--6  
|             |     +-----Chimp  
|             +---5  
|                 +----Human  
--1  
|  
|     +-----Mouse  
|  
+-----Bovine
```

```
{
  "root": {
    "children": [
      {
        "children": [
          {
            "name": "t2",
            "length": 0.04,
            "height": 1.23
          },
          {
            "name": "t1",
            "length": 0.34,
            "height": 0.9300000000000002
          }
        ],
        "length": 0.89,
        "height": 1.27
      },
      {
        "children": [
          {
            "name": "t5",
            "length": 0.37,
            "height": 1.2000000000000002
          },
          {
            "children": [
              {
                "name": "t4",
                "length": 0.03,
                "height": 0.6400000000000001
              },
              {
                "name": "t3",
                "length": 0.67,
                "height": 0.0
              }
            ],
            "length": 0.9,
            "height": 0.6700000000000002
          }
        ],
        "length": 0.9,
        "height": 0.6700000000000002
      }
    ],
    "length": 0.9,
    "height": 0.6700000000000002
  }
}
```

2.构建步骤

- ▶ 2.2 构建系统进化树的主要步骤
 - ▶ 1.选择序列
 - ▶ 2.序列比对（Sequences Alignment）
 - ▶ 3.模型预测
 - ▶ 4.进化树构建
 - ▶ 5.进化树展示与美化

2. 构建步骤

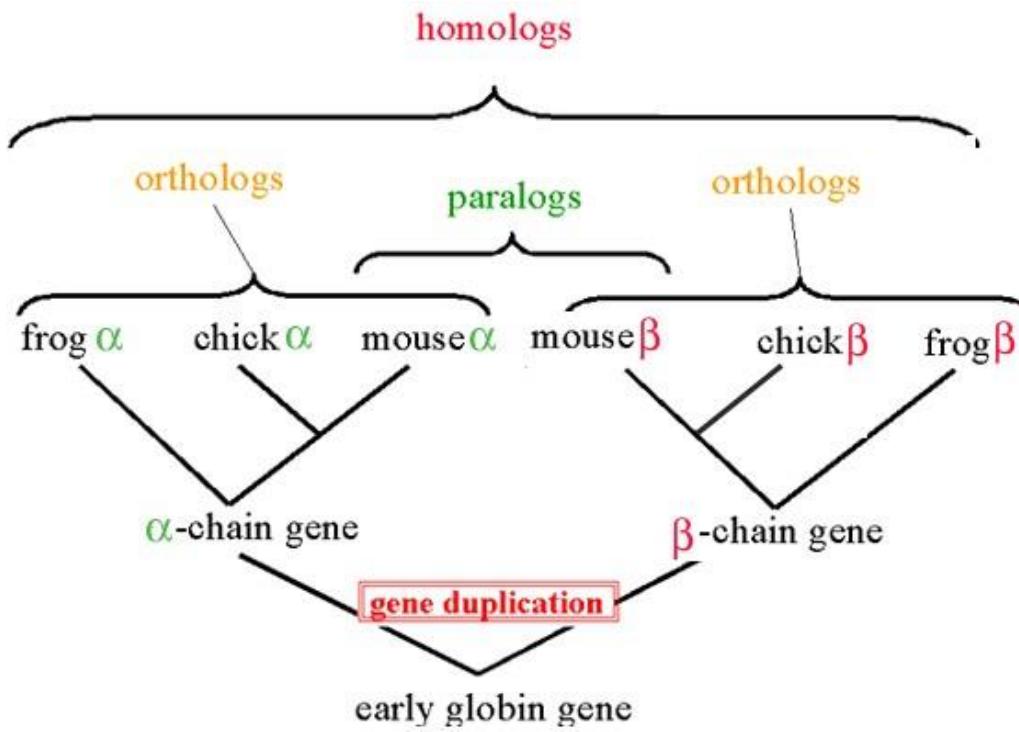
▶ 2.2.1. 序列选择

▶ 直系同源(Orthologs)与旁系同源(Paralogs)

直系同源：同源的基因是由共同的祖先基因进化而产生的。

旁系同源：同源的基因是由于基因复制产生的。

用于分子进化分析中的序列必须是直系同源的才可以真实的反映其进化的过程。



2. 构建步骤

► 2.2.2序列比对

- ▶ 序列比对指将两个或多个序列排列在一起，标明其相似之处。序列中可以插入间隔（通常用短横线“-”表示）。对应的相同或相似的符号（在核酸中是A, T(或U), C, G，在蛋白质中是氨基酸残基的单字母表示）排列在同一列上。
 - ▶ 在比对中，错配与突变相应，而空位与插入或缺失对应。序列比对还可用于语言进化或文本间相似性之类的研究。

tcctctgcctctgccatcat---caaccccaaagt
||||| ||||| ||||| ||||| |||||
tcctgtgcatctgcaatcatggcaaccccaaagt

2.构建步骤

▶ 2.2.2序列比对

- ▶ 序列比对的计算方法一般分为两类：全局性比对 (global alignments) 和局部比对 (local alignments)。计算一个全局性的路线，是一个全局优化的形式，其强制按照整个长度的所有查询序列对齐。与此相反，局部比对只确定局部的相似而整个长序列却往往大相径庭。局部比对往往是可取的，但可能更难以计算的，因为还有来自确定其他相似区域的挑战。
- ▶ 序列比对按照比对数量可以分为成对比对和多序列比对。成对序列比对方法用于寻找两个序列的最佳匹配。成对比对一次只能在两个序列之间使用，但它们在计算时效率很高，通常用于不需要极高精度的方法（例如搜索与查询高度相似的序列的数据库）。多序列比对可以比对两个以上的序列，也可以比对成对序列。多序列比对方法尝试对齐所有给定查询的序列。多序列比对是通常用于在一组假设为进化相关的序列中识别保守区域，并进而构造系统发育树。多序列比对大多数属于NP问题。
- ▶ 产生成对比对的三种主要方法是点阵法、动态规划法和关键词法 (k-tuple)；尽管每种方法都有各自的优点缺点，但是这三种方法都难以处理高重复性的序列-尤其是在要对齐的两个序列中重复次数不同的情况下。

2.构建步骤

▶ 2.2.2序列比对

- 比对后进行人工检查，并选取所需区域，如有必要也可以转换格式（BioEdit or ClustalX2）



ACGTTCGAATCCGTTGCCATGCTGTAGTTATGT
CCGTTGCCATGCTGTAGTT
TTTCGAATCCGTTGCCATGCTG

- 也可使用工具如Gblocks进行选取
- 序列比对本身就是一个随机问题，我们要对这种随机进行检验，比对完后，会有P值。
- **P值的解读：**
- 如果 $P < 10^{-100}$ ，表明两条序列是精确匹配的；
- 如果 $10^{-100} < P < 10^{-50}$ ，表明两条序列近似匹配；
- 如果 $10^{-50} < P < 10^{-5}$ ，表明两条序列有较近的同源关系；
- 如果 $10^{-5} < P < 10^{-1}$ ，表明两条序列可能存在较远的同源关系；
- 如果 $P > 10^{-1}$ ，那么这两条序列能匹配上可能是由于随机的关系；
- 可以使用软件PRSS，用来评价序列两两比对在统计学上显著性。

2.构建步骤

► 2.2.3模型预测

(A) Substitution

Thr	Tyr	Leu	Leu
ACC	TAT	TTG	CTG
	↓		
ACC	TCT	TTG	CTG

(B) Deletion

Thr	Tyr	Leu	Leu
ACC	TAT	TTG	CTG
	↓		
ACC	TAT	TGC	TG-

(C) Insertion

Thr	Tyr	Leu	Leu	
ACC	TAT	TTG	CTG	
	↓			
ACC	TAC	TTT	GCT	G--

(D) Inversion

Thr	Tyr	Leu	Leu
ACC	TAT	TTG	CTG
	↓		
ACC	TTT	ATG	CTG

2.构建步骤

▶ 2.2.3模型预测

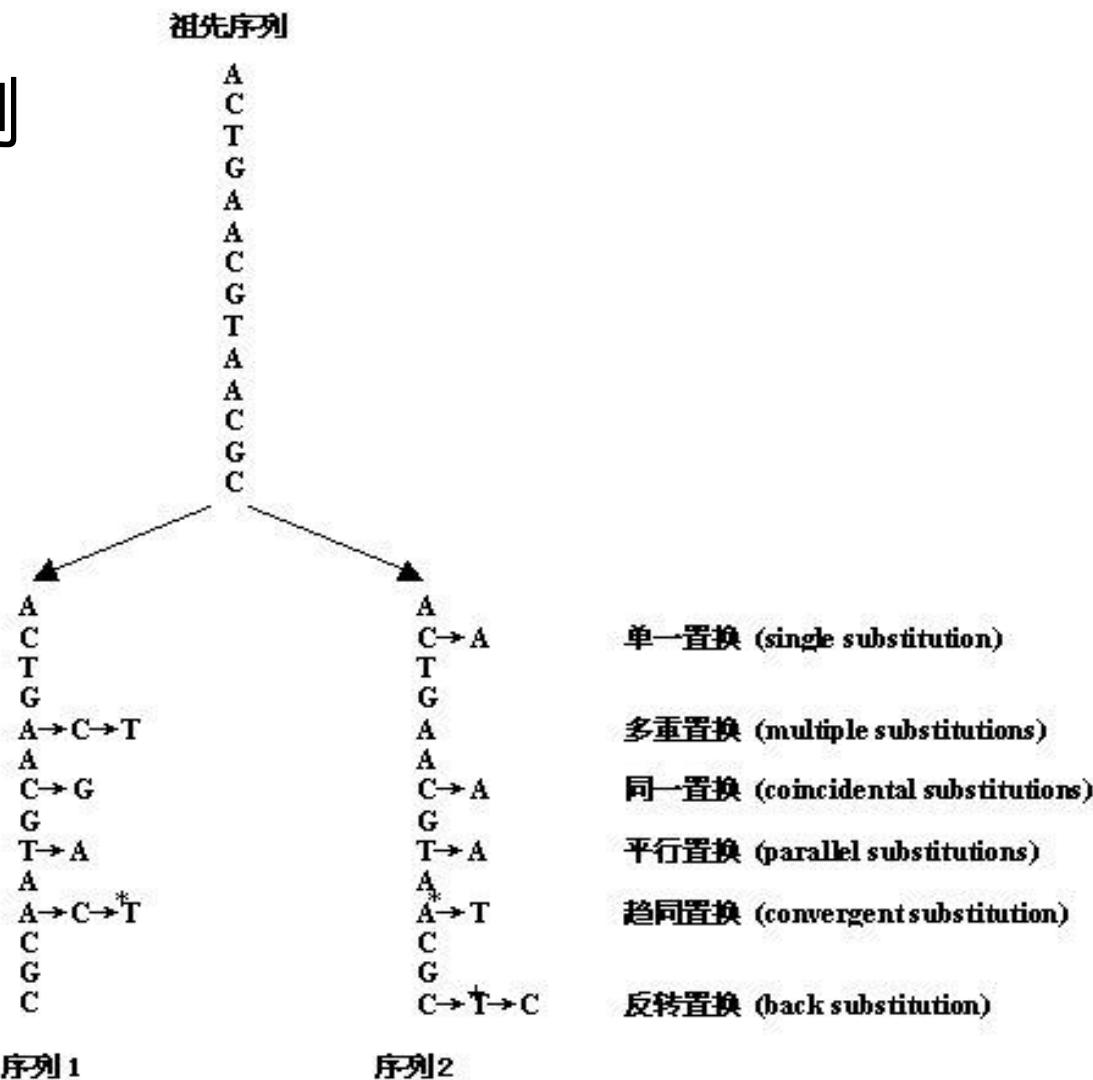
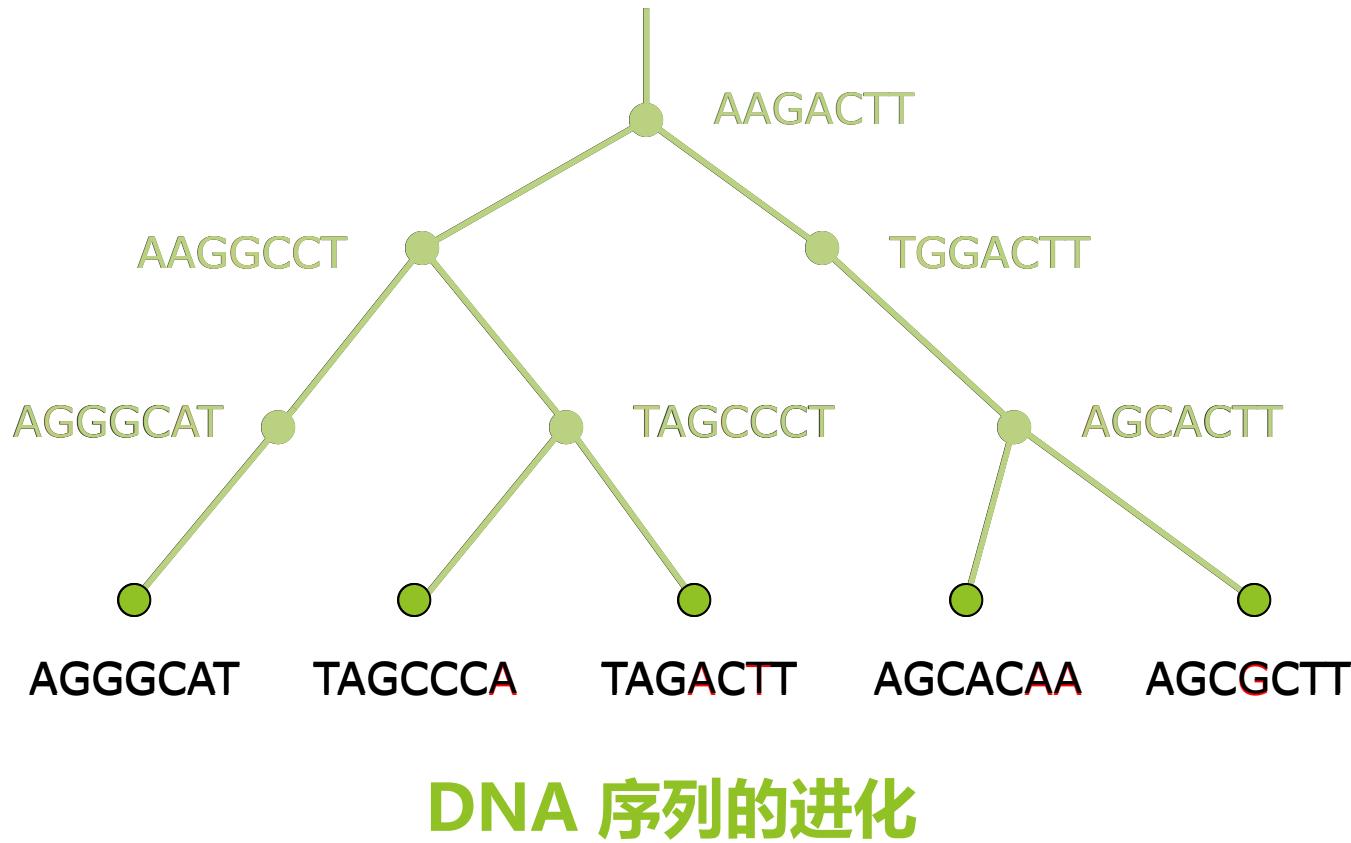


图 8.4 同源序列间的核苷酸置换(自 Li & Graur, 1991)

2.构建步骤

► 2.2.3模型预测

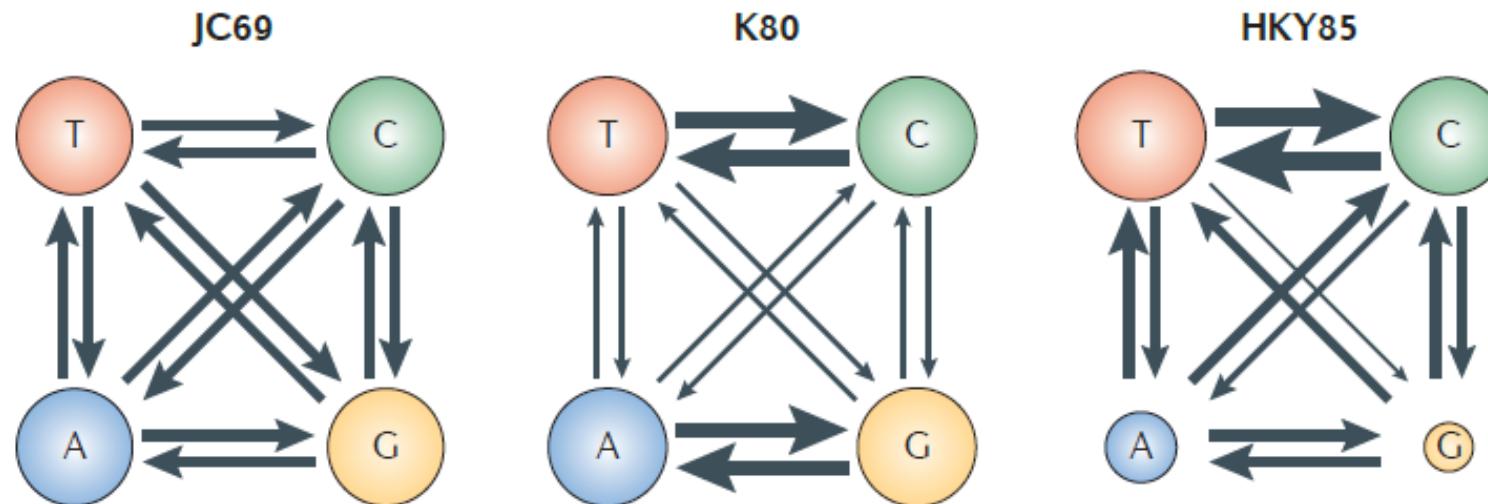


2.构建步骤

▶ 2.2.3模型预测

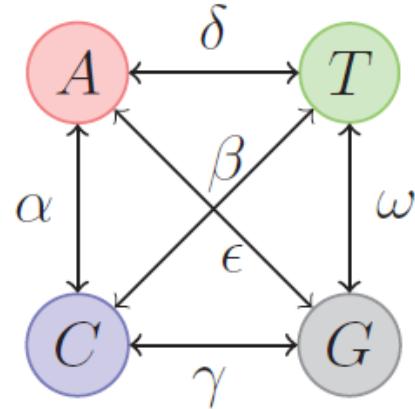
两条序列间的**距离**被定义为平均每个位点核苷酸置换的期望数。

例：如果进化速率是恒定的，距离将随分歧时间线性增长。一种简化的距离测度就是差异位点比例，有时称为 p 距离。如果同为100个核苷酸长度的两条序列间有10个位点差异，则 $p = 10\% = 0.1$ 。



2.构建步骤

▶ 2.2.3模型预测



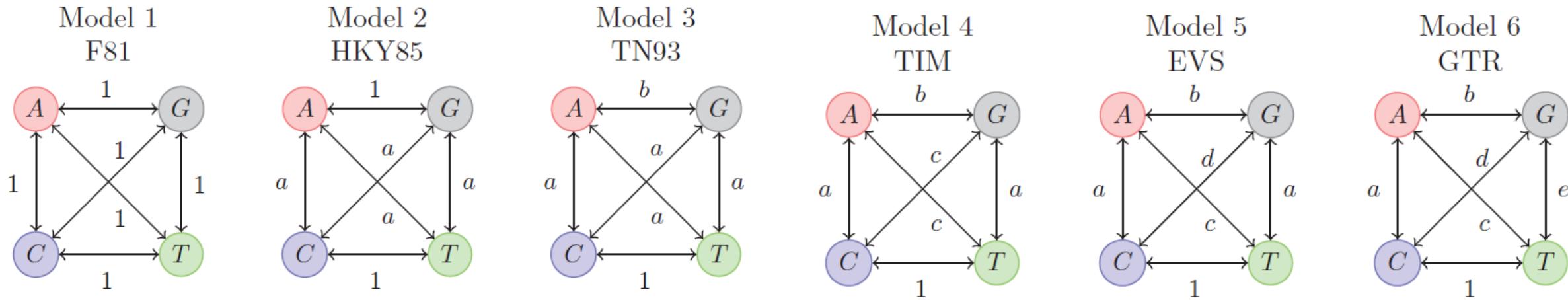
Model	α	β	γ	δ	ϵ	ω	# Dimensions
F81 (JC69)	1	1	1	1	1	1	0
HKY85 (K80)	a	1	a	a	1	a	1
TN93	a	b	a	a	1	a	2
TIM	a	b	c	c	1	a	3
GTR (SYM)	a	b	c	d	1	e	5

DNA置换模型的必要性

- ▶ 必须准确和无偏见地估计分歧度和速率
- ▶ 数学模型可以考虑回复和平行突变等情况，尤其是在p值较大时

2.构建步骤

▶ 2.2.3模型预测



常见模型的局限性

- ▶ 所有位点的替代速率并不是一致的
- ▶ 一些位点的进化并不是独立的, 如相互作用位点可能需要互补突变 (例: 发卡结构)

2.构建步骤

► 2.2.3模型预测

<http://www.iqtree.org/doc/Substitution-Models>

DNA models

Model	df	Explanation	Code
JC or JC69	0	Equal substitution rates and equal base frequencies (Jukes and Cantor, 1969).	000000
F81	3	Equal rates but unequal base freq. (Felsenstein, 1981).	000000
K80 or K2P	1	Unequal transition/transversion rates and equal base freq. (Kimura, 1980).	010010
HKY or HKY85	4	Unequal transition/transversion rates and unequal base freq. (Hasegawa, Kishino and Yano, 1985).	010010
TN or TN93	5	Like HKY but unequal purine/pyrimidine rates (Tamura and Nei, 1993).	010020
TNe	2	Like TN but equal base freq.	010020
K81 or K3P	2	Three substitution types model and equal base freq. (Kimura, 1981).	012210
K81u	5	Like K81 but unequal base freq.	012210
TPM2	2	AC=AT, AG=CT, CG=GT and equal base freq.	010212
TPM2u	5	Like TPM2 but unequal base freq.	010212
TPM3	2	AC=CG, AG=CT, AT=GT and equal base freq.	012012
TPM3u	5	Like TPM3 but unequal base freq.	012012
TIM	6	Transition model, AC=GT, AT=CG and unequal base freq.	012230
TIMe	3	Like TIM but equal base freq.	012230
TIM2	6	AC=AT, CG=GT and unequal base freq.	010232
TIM2e	3	Like TIM2 but equal base freq.	010232
TIM3	6	AC=CG, AT=GT and unequal base freq.	012032
TIM3e	3	Like TIM3 but equal base freq.	012032
TVM	7	Transversion model, AG=CT and unequal base freq.	012314
TVMe	4	Like TVM but equal base freq.	012314
SYM	5	Symmetric model with unequal rates but equal base freq. (Zharkikh, 1994).	012345
GTR	8	General time reversible model with unequal rates and unequal base freq. (Tavaré, 1986).	012345

2.构建步骤

► 2.2.3模型预测

Protein models

<http://www.iqtree.org/doc/Substitution-Models>

Model	Region	Explanation
Blosum62	nuclear	BLOcks SUbsitution Matrix (Henikoff and Henikoff, 1992). Note that BLOSUM62 is not recommended for phylogenetic analysis as it was designed mainly for sequence alignments.
cpREV	chloroplast	chloroplast matrix (Adachi et al., 2000).
Dayhoff	nuclear	General matrix (Dayhoff et al., 1978).
DCMut	nuclear	Revised Dayhoff matrix (Kosiol and Goldman, 2005).
FLU	viral	Influenza virus (Dang et al., 2010).
HIVb	viral	HIV between-patient matrix HIV-B _m (Nickle et al., 2007).
HIVw	viral	HIV within-patient matrix HIV-W _m (Nickle et al., 2007).
JTT	nuclear	General matrix (Jones et al., 1992).
JTTDCMut	nuclear	Revised JTT matrix (Kosiol and Goldman, 2005).
LG	nuclear	General matrix (Le and Gascuel, 2008).
mtART	mitochondrial	Mitochondrial Arthropoda (Abascal et al., 2007).
mtMAM	mitochondrial	Mitochondrial Mammalia (Yang et al., 1998).
mtREV	mitochondrial	Mitochondrial Vertebrate (Adachi and Hasegawa, 1996).
mtZOA	mitochondrial	Mitochondrial Metazoa (Animals) (Rota-Stabelli et al., 2009).
mtMet	mitochondrial	Mitochondrial Metazoa (Vinh et al., 2017).
mtVer	mitochondrial	Mitochondrial Vertebrate (Vinh et al., 2017).
mtInv	mitochondrial	Mitochondrial Invertebrate (Vinh et al., 2017).
Poisson	none	Equal amino-acid exchange rates and frequencies.
PMB	nuclear	Probability Matrix from Blocks, revised BLOSUM matrix (Veerassamy et al., 2004).
rtREV	viral	Retrovirus (Dimmic et al., 2002).
VT	nuclear	General ‘Variable Time’ matrix (Mueller and Vingron, 2000).
WAG	nuclear	General matrix (Whelan and Goldman, 2001).
GTR20	general	General time reversible models with 190 rate parameters. <i>WARNING: Be careful when using this parameter-rich model as parameter estimates might not be stable, especially when not having enough phylogenetic information (e.g. not long enough alignments).</i>

2.构建步骤

▶ 2.2.3模型预测

AICc Selection Results

Model	TrNef+G
partition	010020
-lnL	2940.5106
K	103
freqA	- R(a) 1.0000
freqC	- R(b) 13.1860
freqG	- R(c) 1.0000
freqT	- R(d) 1.0000
ti/tv	- R(e) 20.1118
	R(f) 1.0000
p-inv	- gamma 0.2900

结果解读

- ①**核苷酸替换模型**: TrNef
- ②**替换模型代号**: 010020
- ③**变异速率模型**: Gamma(G)
- ④R(a)即Rate [AC]; R(b)即Rate [AG]; R(c)即Rate [AT];
R(d)即Rate [CG]; R(e)即Rate [CT]; R(f)即Rate [GT]
- ⑤Gamma变异速率的**shape值** (或叫alpha值) : 0.2900

这里需要注意的是，在不同软件里，模型的名字有时会不一样，但是其实是同一个模型，需要互相比对不同软件里的模型叫法，最好的方法是去了解该模型的“别称”，然后读你所使用的建树软件的说明书，找到对应模型的代号。

2.构建步骤

▶ 2.2.4进化树构建 (以raxml为例)

- ▶ RaxML对序列格式和序列内容要求非常严格，建树前先检查序列格式是否错误。
- ▶ 1. 序列的名称禁止出现空格，制表符，换行符，：，（）或者[]。
- ▶ 2. 不能有相同名称的序列。这个可能是我们将其他格式的序列变成phylip格式的时候，只保留序列名称前8个或者10个字符造成的。
- ▶ 3. 不能出现不同名称但序列一样的这种序列。
- ▶ 4. 序列不能完全由未知符号组成，即序列里不能全是这样的字符：如氨基酸序列里不能全是X、？、*；如DNA序列里不能全是N、O、X、？。（红字部分）
- ▶ 注意：①RaxML会自动移除你序列文件里的相同序列和不确定的序列。
- ▶ ②如果是一个分区模型的分析，则相应的模型文件modelFileName.reduced也会被读写。RAxML遇到相同的序列名称或未确定的序列或非法字符，它将退出并报错。（②这种情况是多基因联合建树）

IUPAC amino acid code	Three letter code	Amino acid	IUPAC nucleotide code	Base
A	Ala	Alanine	A	Adenine
C	Cys	Cysteine	C	Cytosine
D	Asp	Aspartic Acid	G	Guanine
E	Glu	Glutamic Acid	T (or U)	Thymine (or Uracil)
F	Phe	Phenylalanine	R	A or G
G	Gly	Glycine	Y	C or T
H	His	Histidine	S	G or C
I	Ile	Isoleucine	W	A or T
K	Lys	Lysine	K	G or T
L	Leu	Leucine	M	A or C
M	Met	Methionine	B	C or G or T
N	Asn	Asparagine	D	A or G or T
P	Pro	Proline	H	A or C or T
Q	Gln	Glutamine	V	A or C or G
R	Arg	Arginine	N	any base
S	Ser	Serine	.	gap
T	Thr	Threonine	or -	
V	Val	Valine		
W	Trp	Tryptophan		
Y	Tyr	Tyrosine		

2.构建步骤

► 2.2.4进化树构建 (以raxml为例)

raxmlHPC

```
-s sequenceFileName -n outputFileName -m substitutionModel
[-a weightFileName] [-A secondaryStructureSubstModel]
[-b bootstrapRandomNumberSeed] [-B wcCriterionThreshold]
[-c numberOfCategories] [-C] [-d] [-D]
[-e likelihoodEpsilon] [-E excludeFileName]
[-f a|A|b|B|c|C|d|D|e|E|F|g|G|h|H|i|I|j|J|k|m|n|N|o|p|q|r|R|s|s|t|T|u|
v|V|w|W|x|y]
[-F]
[-g groupingFileName] [-G placementThreshold] [-h] [-H]
[-i initialRearrangementSetting]
[-I autoFC|autoMR|autoMRE|autoMRE_IGN]
[-j] [-J MR|MR_DROP|MRE|STRICT|STRICT_DROP|T_<PERCENT>] [-k] [-K]
[-L MR|MRE|T_<PERCENT>] [-M]
[-o outGroupName1[,outGroupName2[,...]]][-O]
[-p parsimonyRandomSeed] [-P proteinModel]
[-q multipleModelFileName] [-r binaryConstraintTree]
[-R binaryModelParamFile] [-S secondaryStructureFile]
[-t userStartingTree]
[-T numberOfWorkThreads] [-u] [-U] [-v] [-V] [-w outputDirectory]
[-W slidingWindowSize]
[-x rapidBootstrapRandomNumberSeed][-X][-Y]
[-Y quartetGroupingFileName|ancestralSequenceCandidatesFileName]
[-z multipleTreesFile]
[-#|-N numberOfWorkRuns|autoFC|autoMR|autoMRE|autoMRE_IGN]
[--mesquite][--silent][--no-seq-check][--no-bfgs]
[--asc-corr=stamatakis|felsenstein|lewis]
[--flag-check]
[--auto-prot=ml|bic|aic|aicc ]
[--epa-keep-placements=number]
[--epa-accumulated-threshold=threshold]
[--epa-prob-threshold=threshold]
```

Simple Parameters

Maximum Hours to Run (click here for help setting this correctly) * 1

Set a name for output files test_haha

Enable ML searches under CAT (-F)

Enter the number of patterns in your dataset

Please select the Data Type * Nucleotide

Outgroup (one or more comma-separated outgroups, see comment for syntax)

Specify the number of distinct rate categories (-c) * 25

Disable Rate Heterogeneity (-V)

Supply a tree (Not available when doing rapid bootstrapping, -x) (-t)

Specify a random seed value for parsimony inferences (-p) *

Enter a random seed value for parsimony inferences (gives reproducible results from random starting tree) * 12345

Specify an initial rearrangement setting (-i) *

Specify the distance from original pruning point (-i) * 10

Constraint (-g)

Binary Backbone (-r)

Use a mixed/partitioned model? (-q)

Estimate individual per-partition branch lengths (-M) *

Correct for Ascertainment bias (ASC_) no yes

Ascertainment bias correction type (-asc-corr) [Not Mandatory] Lewis Felsenstein Stamatakis

Estimate proportion of invariable sites (GTRGAMMA + I) no yes

Choose an input file that excludes the range of positions specified in this file (-E)

Weight characters as specified in this file (-a)

Disable checking for sequences with no values (-O)

Print output files that can be parsed by Mesquite. (-mesquite)

2.构建步骤

► 2.2.4进化树构建 (以raxml为例)

Simple Parameters

Maximum Hours to Run (click here for help setting this correctly) * 1

Set a name for output files test_haha

Enable ML searches under CAT (-F)

Enter the number of patterns in your dataset

Please select the Data Type * Nucleotide

Outgroup (one or more comma-separated outgroups, see comment for syntax)

Specify the number of distinct rate categories (-c) * 25

Disable Rate Heterogeneity (-V)

Supply a tree (Not available when doing rapid bootstrapping, -x) (-t)

Specify a random seed value for parsimony inferences (-p) *

Enter a random seed value for parsimony inferences (gives reproducible results from random starting tree) * 12345

Specify an initial rearrangement setting (-i)

Specify the distance from original pruning point (-i) * 10

Constraint (-g)

Binary Backbone (-r)

Use a mixed/partitioned model? (-q)

Estimate individual per-partition branch lengths (-M)

Correct for Ascertainment bias (ASC_) no yes

Ascertainment bias correction type (-asc-corr) [Not Mandatory] Lewis Felsenstein Stamatakis

Estimate proportion of invariable sites (GTRGAMMA + I) no yes

Choose an input file that excludes the range of positions specified in this file (-E)

Weight characters as specified in this file (-a)

Disable checking for sequences with no values (-O)

Print output files that can be parsed by Mesquite. (-mesquite)

- 输入文件 → MSA file (多序列比对文件)
- 序列类型 → 核酸or氨基酸?
- 替换模型 → GTR or HKY or ... ?
- Bootstrap → 1000? 10000?
- 输出文件 → 名称?

2.构建步骤

► 2.2.4 进化树构建 (以raxml为例)

```

raxmlHPC
-s sequenceFileName -n outputFileName -m substitutionModel
[-a weightFileName] [-A secondaryStructureSubstModel]
[-b bootstrapRandomNumberSeed] [-B wcCriterionThreshold]
[-c numberOfCategories] [-C] [-d] [-D]
[-e likelihoodEpsilon] [-E excludeFileName]
[-f a|A|b|B|c|C|d|D|e|E|F|g|G|h|H|i|I|j|J|k|m|n|N|o|p|q|r|R|s|S|t|T|u|
v|V|w|W|x|y]
[-F]
[-g groupingFileName] [-G placementThreshold] [-h] [-H]
[-i initialRearrangementSetting]
[-I autoFC|autoMR|autoMRE|autoMRE_IGN]
[-j] [-J MR|MR_DROP|MRE|STRICT|STRICT_DROP|T_<PERCENT>] [-k] [-K]
[-L MR|MRE|T_<PERCENT>] [-M]
[-o outGroupName1[,outGroupName2,...]]] [-O]
[-p parsimonyRandomSeed] [-P proteinModel]
[-q multipleModelFileName] [-r binaryConstraintTree]
[-R binaryModelParamFile] [-S secondaryStructureFile]
[-t userStartingTree]
[-T numberOfWorkThreads] [-u] [-U] [-v] [-V] [-w outputDirectory]
[-W slidingWindowSize]
[-x rapidBootstrapRandomNumberSeed] [-X] [-y]
[-Y quartetGroupingFileName|ancestralSequenceCandidatesFileName]
[-z multipleTreesFile]
[-#|-N numberOfWorkRuns|autoFC|autoMR|autoMRE|autoMRE_IGN]
[--mesquite][--silent][--no-seq-check][--no-bfgs]
[--asc-corr=stamatakis|felsenstein|lewis]
[--flag-check]
[--auto-prot=ml|bic|aic|aicc ]
[--epa-keep-placements=number]
[--epa-accumulated-threshold=threshold]
[--epa-prob-threshold=threshold]

```

- ▶ -f a
此参数用于选择 RAxML 运算的算法。可以设定的值非常之多。 a 表示执行快速 Bootstrap 分析并搜索最佳得分的 ML 树。
- ▶ -x 12345
指定一个 int 数作为随机种子，以启用快速 Bootstrap 算法。
- ▶ -p 12345
指定一个随机数作为 parsimony inferences 的种子。
- ▶ -# 100
指定 bootstrap 的次数。
- ▶ -m PROTGAMMALGX
指定核苷酸或氨基酸替代模型。PROTGAMMALGX 的解释：“PROT”表示氨基酸替代模型； GAMMA 表示使用 GAMMA 模型； X 表示使用最大似然法估计碱基频率。
- ▶ -s ex.phy
指定输入文件。phy 格式的多序列比对结果。软件包中包含一个程序来将 fasta 格式转换为 phy 格式。
- ▶ -n ex
输出文件的后缀为 .ex 。
- ▶ -T 20
指定多线程运行的 CPUs 。

2. 构建步骤

► 2.2.4 进化树构建 (以raxml为例)

▶ 结果文件

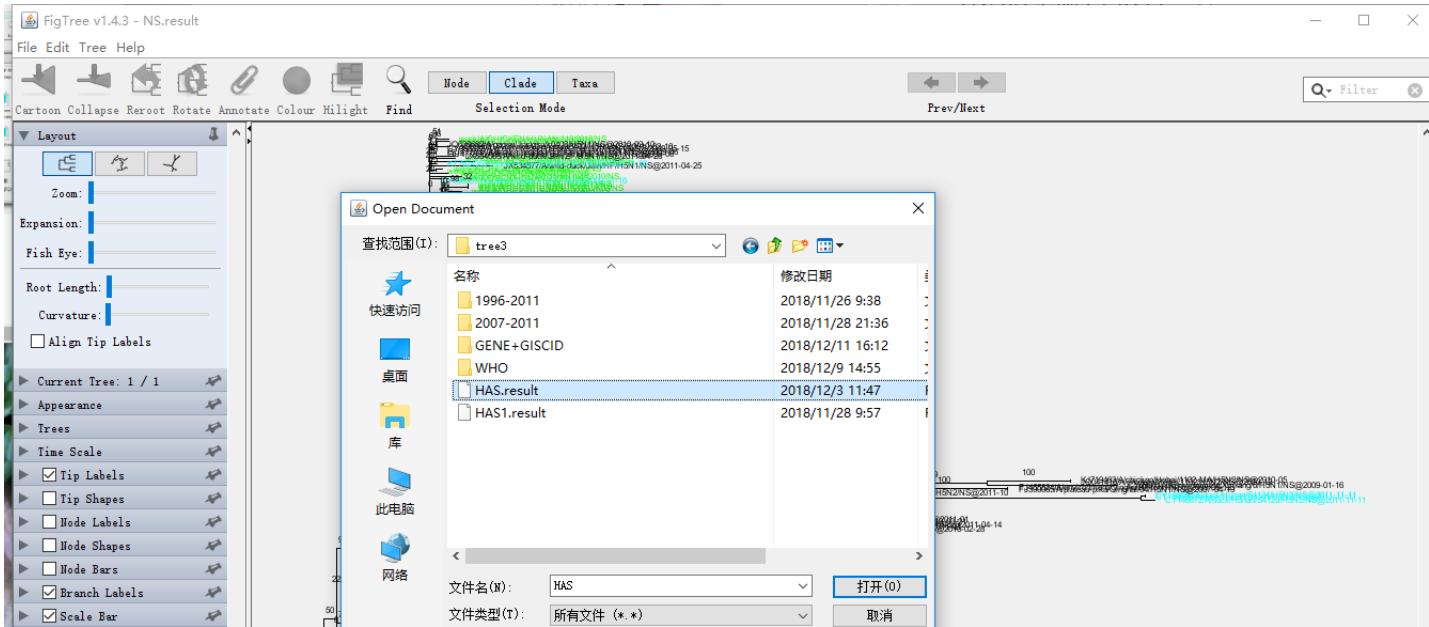
- RAxML_bootstrap.ex bootstrapped trees
 - RAxML_bestTree.ex 最佳得分 ML 树
 - RAxML_bipartitions.ex 有 bootstrap 分值支持的最佳得分树，分值在 node 上。
 - RAxML_bipartitionsBranchLabels.ex 有 bootstrap 分值支持的最佳得分树，分值在 branch 上
FigTree不能识别此文件。

2.构建步骤

► 2.2.5 进化树展示

File I/O

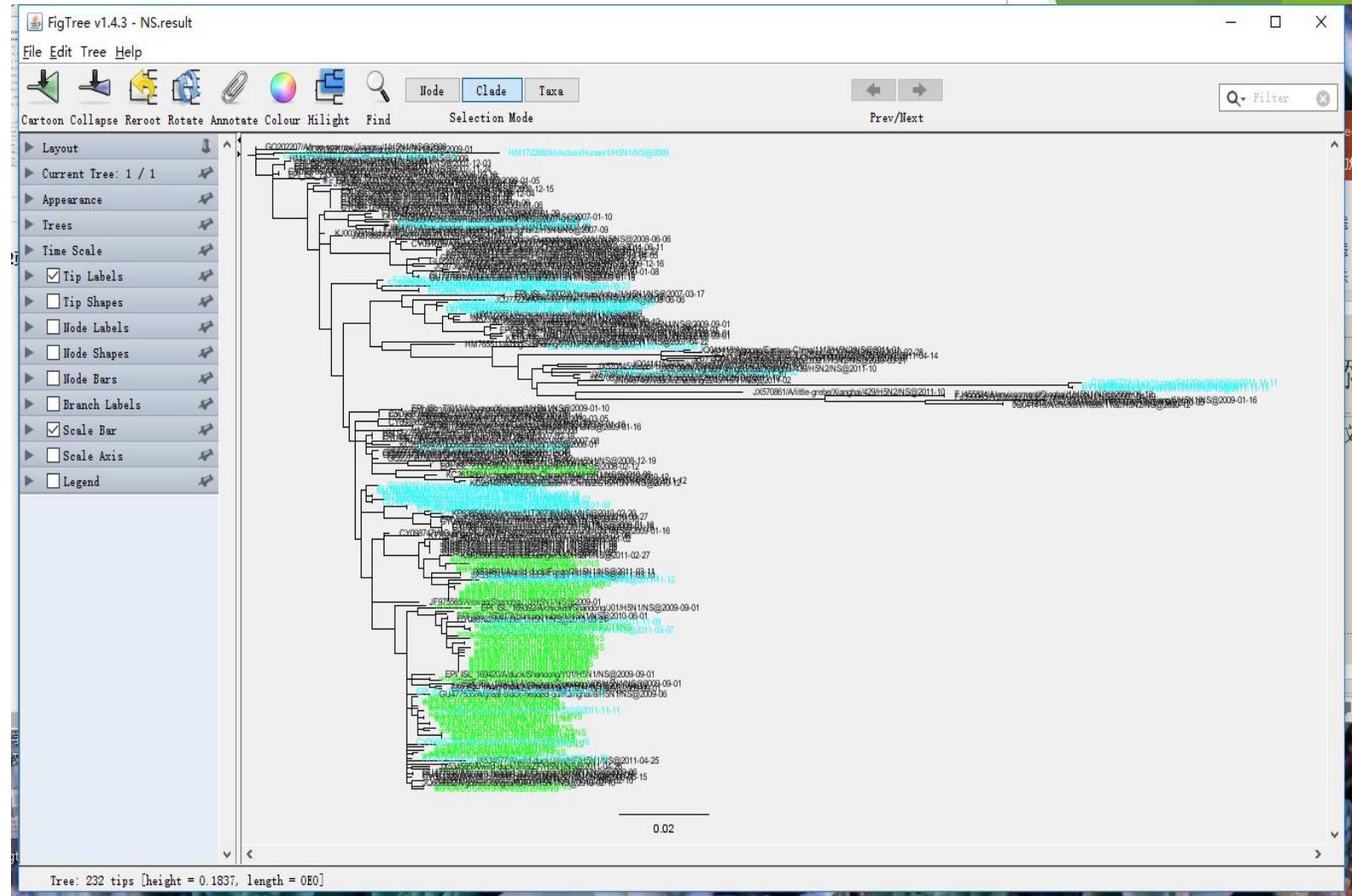
- file-open 可以打开*.tre, *.tree, *.result文件； file-export tree/PDF 等多种输出格式。



2.构建步骤

► 2.2.5 进化树展示

Interface



2.构建步骤

► 2.2.5 进化树展示

Menu

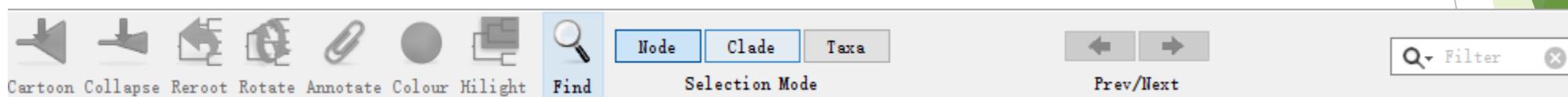
- File 一般是输入、输出，新建、保存，打印等功能
- Edit 复制、粘贴、选择、查找等功能
- Tree 和工具栏功能基本一样，但是有清除功能
- Help 帮助~

2. 构建步骤

► 2.2.5 进化树展示

Toolbar

- 没选择树的时候是灰的



- ▶ 选择树的node、clade、taxa等都可以，就亮了，就可以操作你所选择的区域了

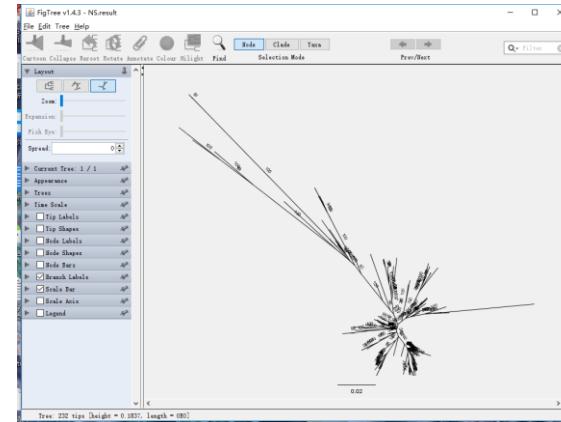
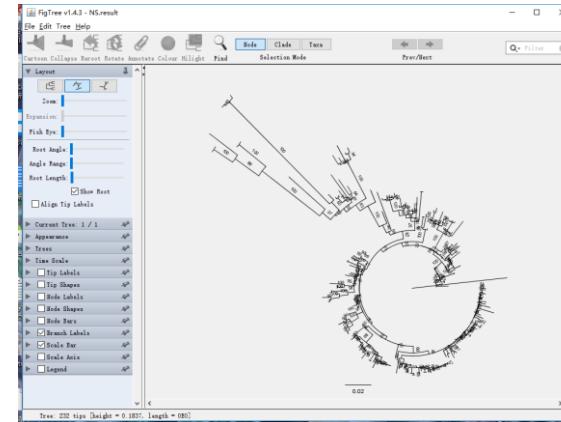
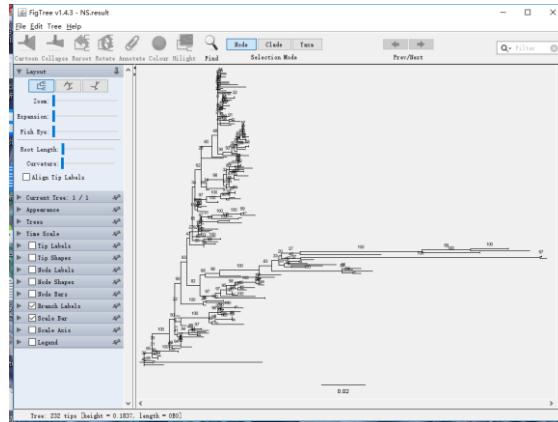


2. 构建步骤

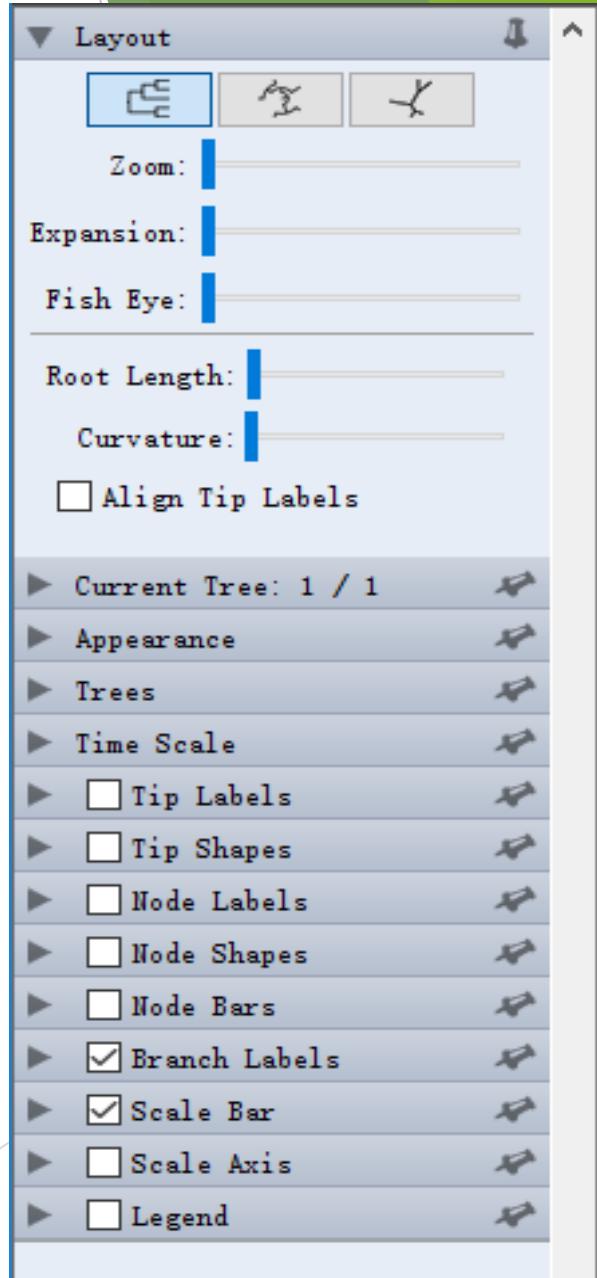
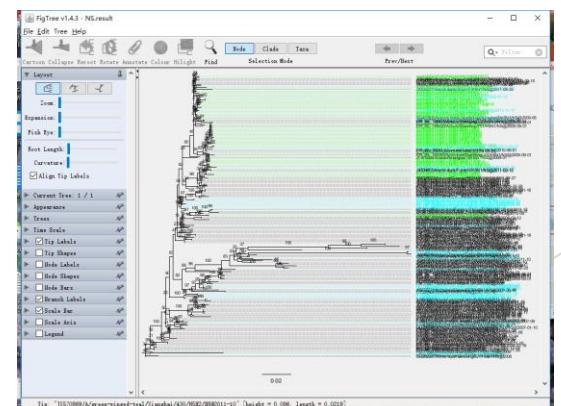
▶ 2.2.5 进化树展示

Layout

- ▶ 可以选择不同的树型，这些都是无根树



- ▶ Align tip labels和tip labels一起选择的效果（左）

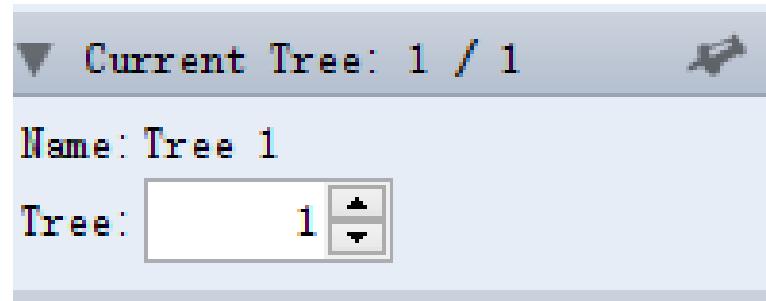


2.构建步骤

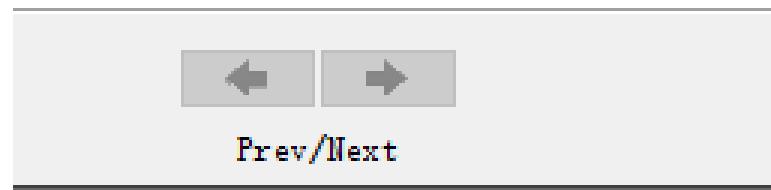
► 2.2.5 进化树展示

Current tree

- 现在只有一个树，超过一个可以在这里切换



- 还可以在这里，工具栏里面进行切换

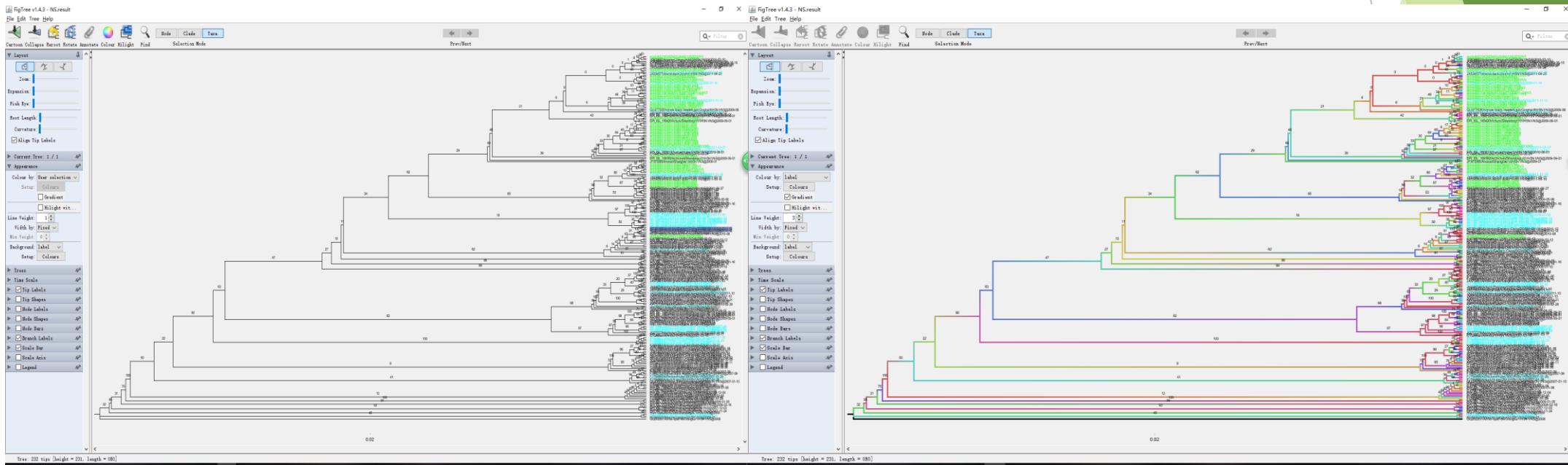


2.构建步骤

► 2.2.5 进化树展示

Appearance

- 这里可以改变整体（除了taxa）的颜色，（左图）默认为黑色，（右图）根据自己选择做出为结果。

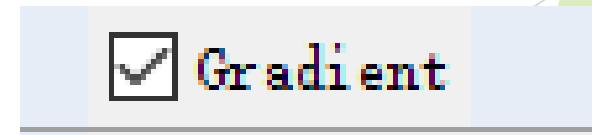
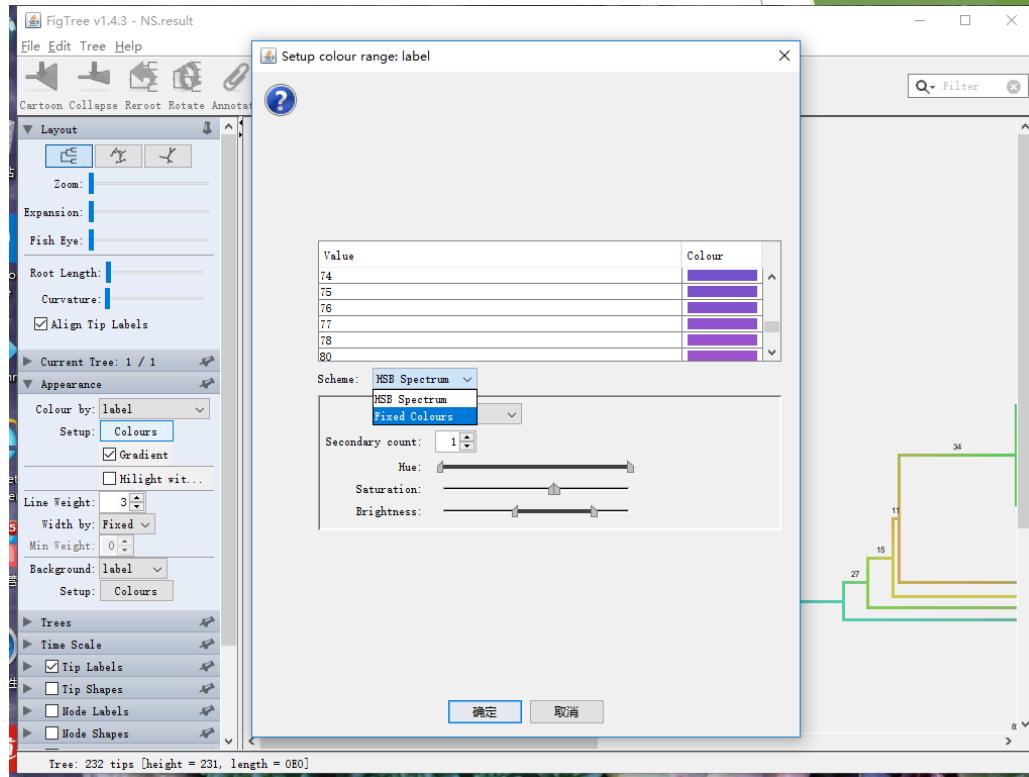


2.构建步骤

▶ 2.2.5 进化树展示

Appearance

- ▶ 里面有很多可以选择的
- ▶ Setup: colours点进(右图)
- ▶ 可以选择scheme, 还可以调整
- ▶ 色调、饱和度、亮度等等
- ▶ 其他的还可以勾选一下gradient
- ▶ 色彩柔和均匀一点

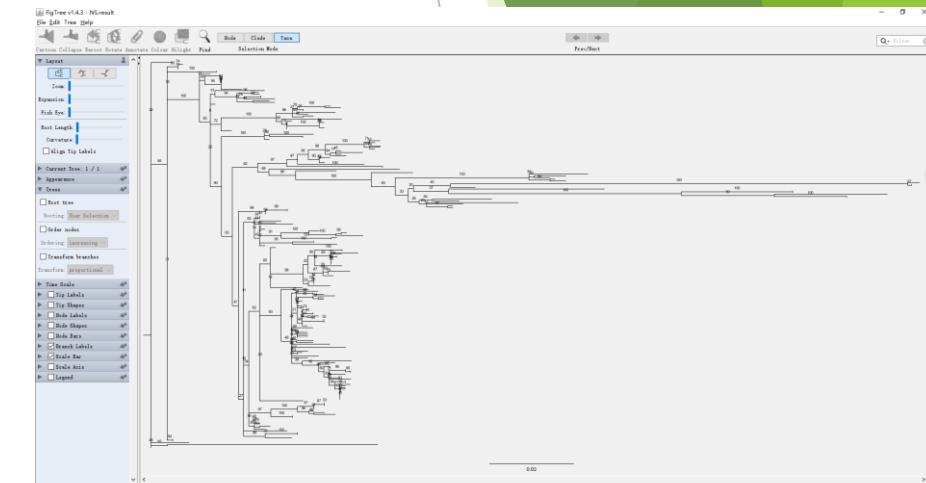


2.构建步骤

► 2.2.5 进化树展示

Trees

- 三个选择一个都没选（右上）
- Order tree（左下），可以升序和降序，这个是升序
- 在上面的基础上，选择了一个root（中间）
- 也可以改变一下branch的样子，选择transform branches

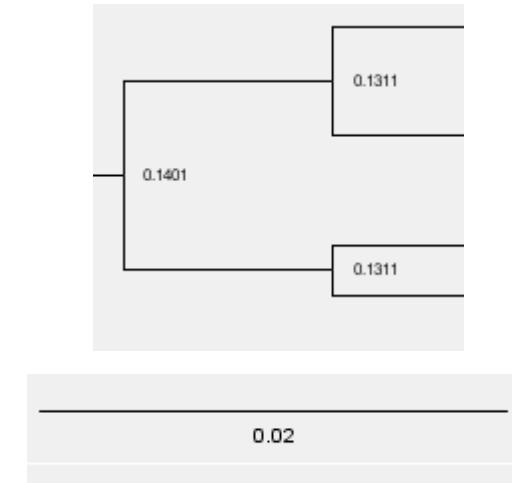
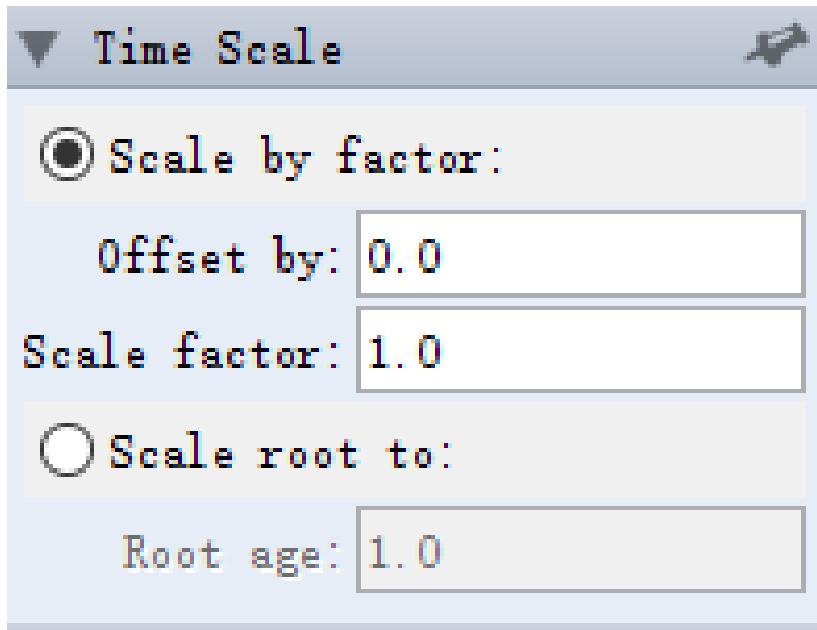


2.构建步骤

► 2.2.5 进化树展示

Time Scale

- 这个试一试就可以，主要是改变scale和node age等选择的，不常用

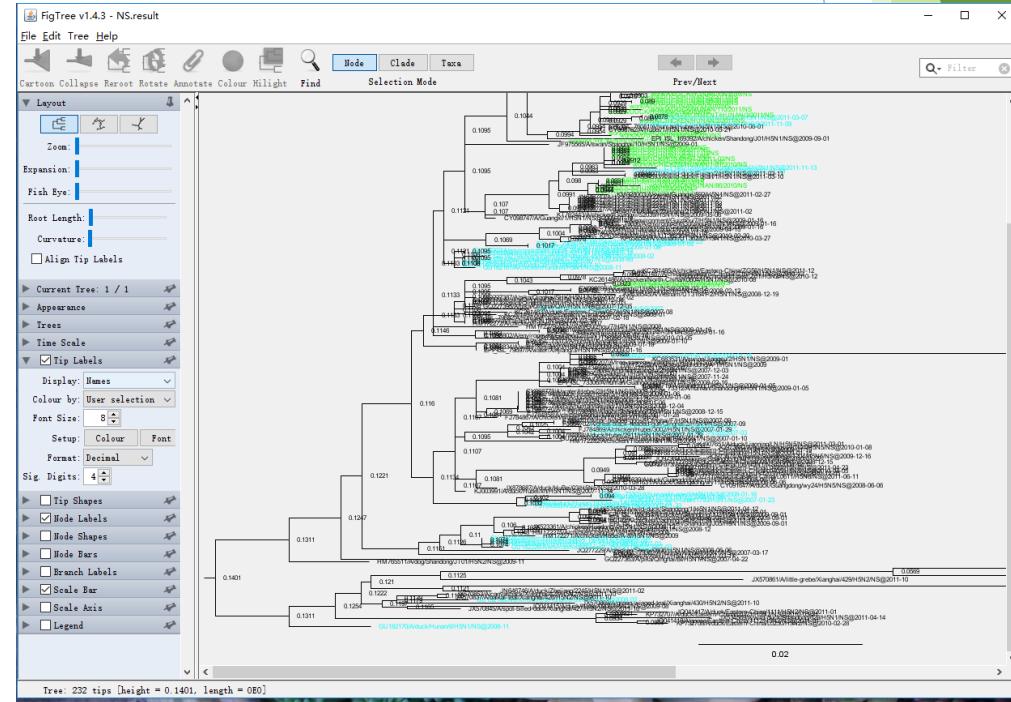
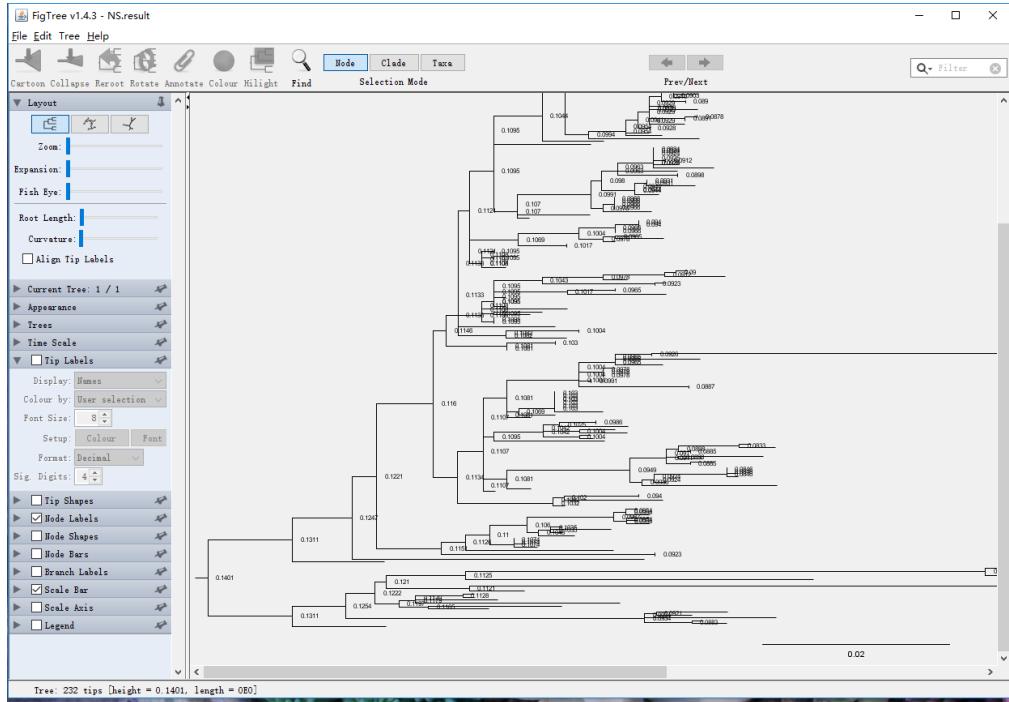


2.构建步骤

► 2.2.5 进化树展示

Tip labels

- 这个是在枝的末端显示，一般显示names，还可以改变一下字体，字号，颜色等。

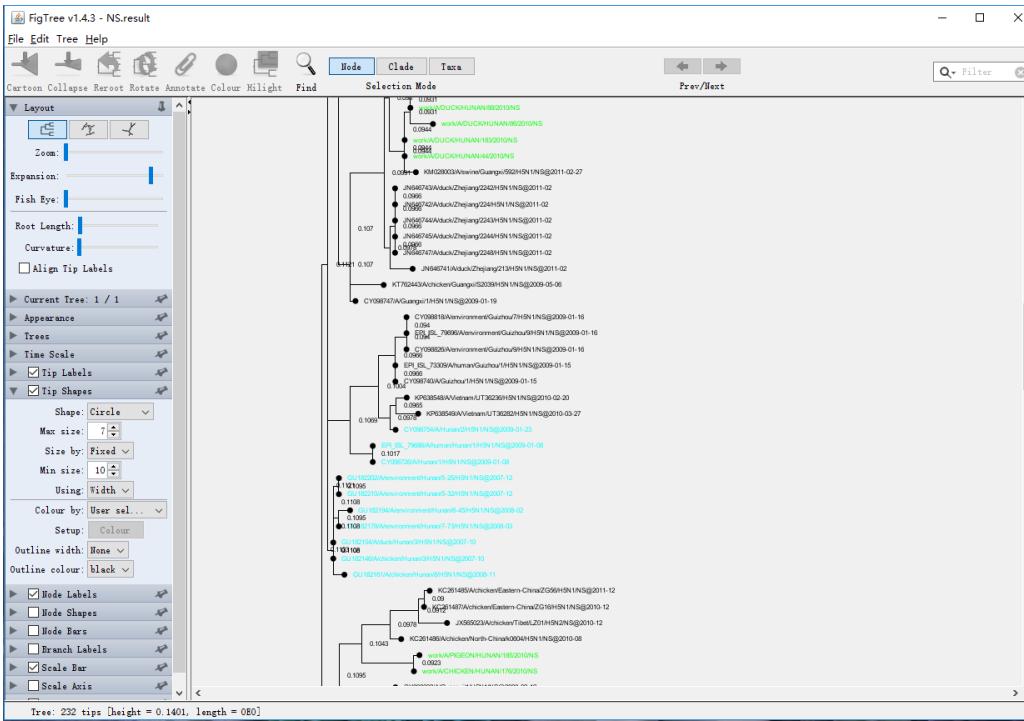


2.构建步骤

► 2.2.5 进化树展示

Tip shapes

- 改变tip前的节点的形状（二叉树叶的形状），有圆形、方形和菱形三种形状，可以改变大小，颜色，outline等等

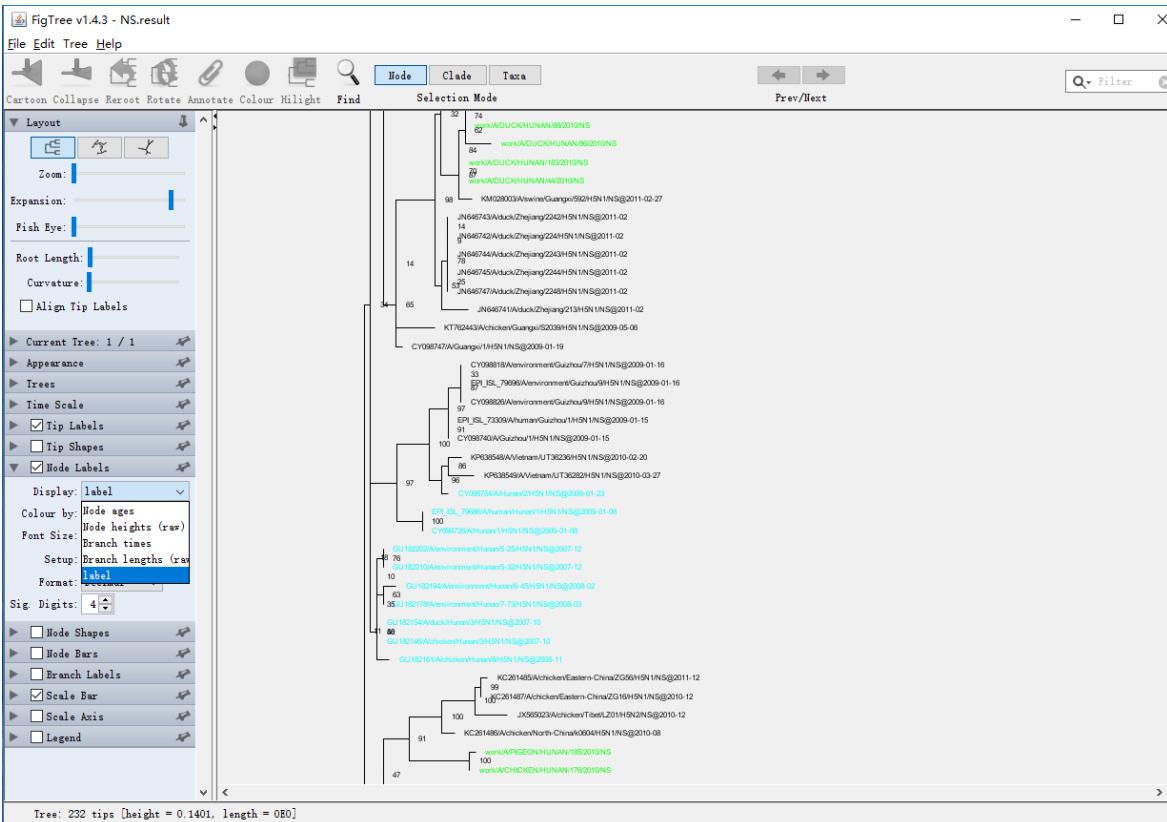


2.构建步骤

► 2.2.5 进化树展示

Node labels

- Display有多种选择，一般选择label，显示支持度

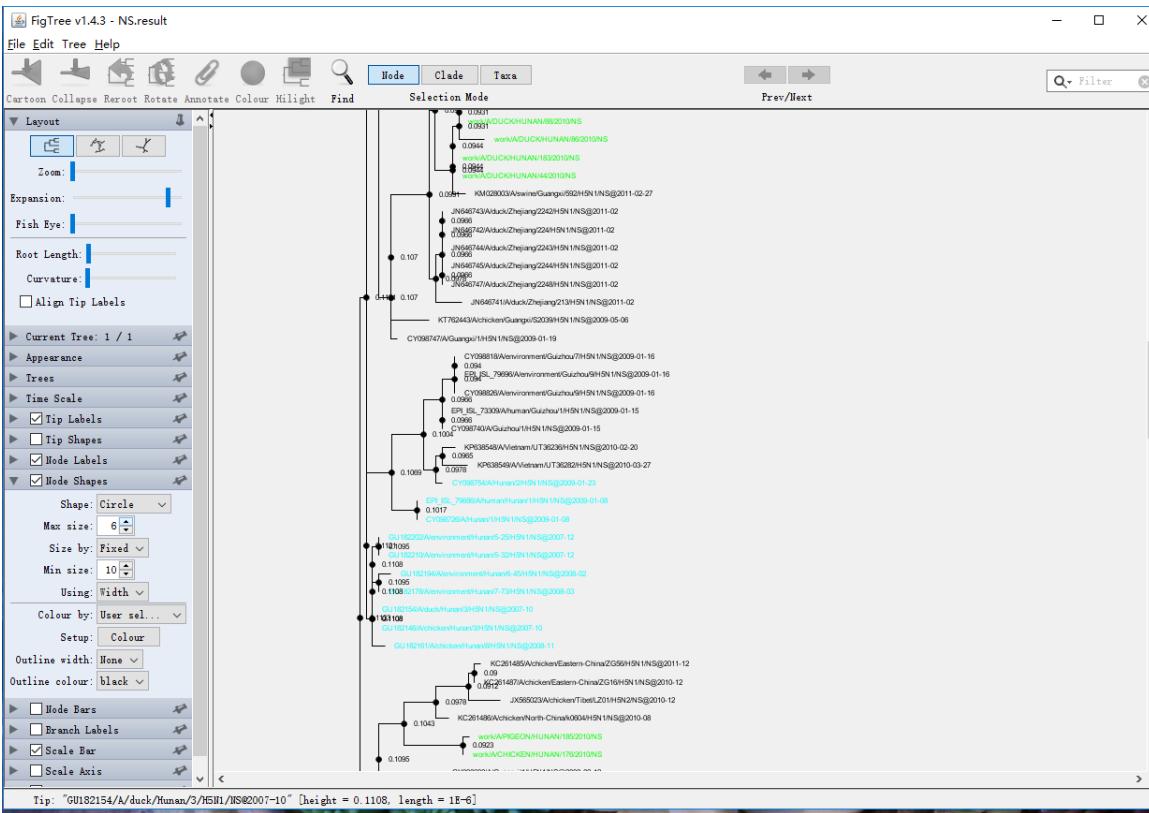


2.构建步骤

► 2.2.5 进化树展示

Node shapes

- 与tip shapes相似，只是位置在node



2.构建步骤

► 2.2.5 进化树展示

Node bars

- 可以选择display和bar width

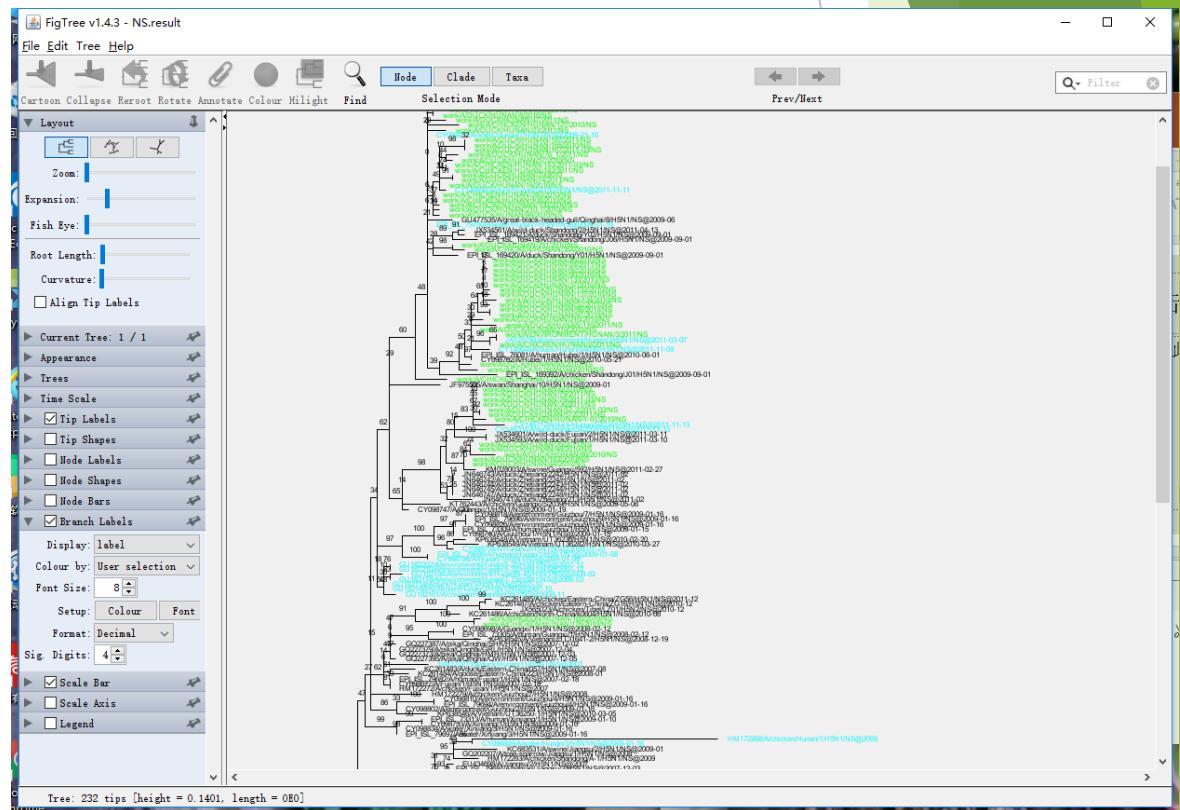


2.构建步骤

► 2.2.5 进化树展示

Branch labels

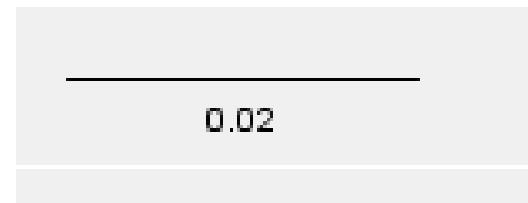
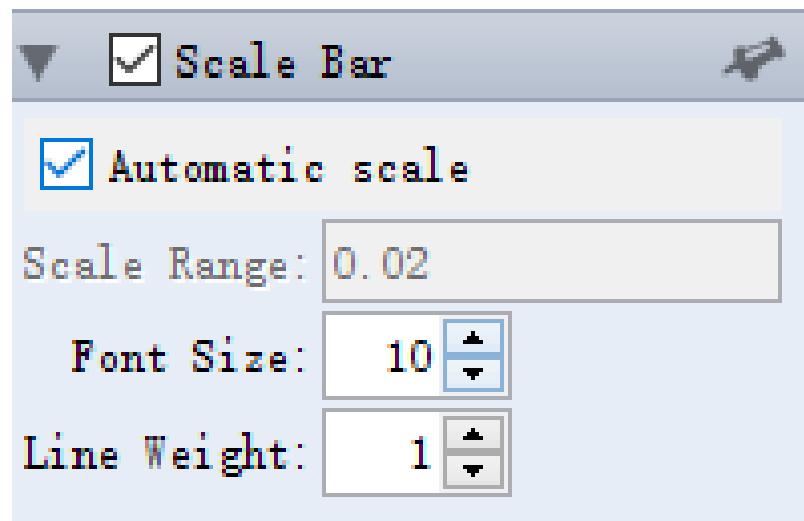
- 和node labels相似，改变的是branch



2.构建步骤

► 2.2.5 进化树展示 **Scale bar**

- ▶ 可以选择是否显示scale， scale大小可以自定，还可以改变scale的字体大小的线的粗细。

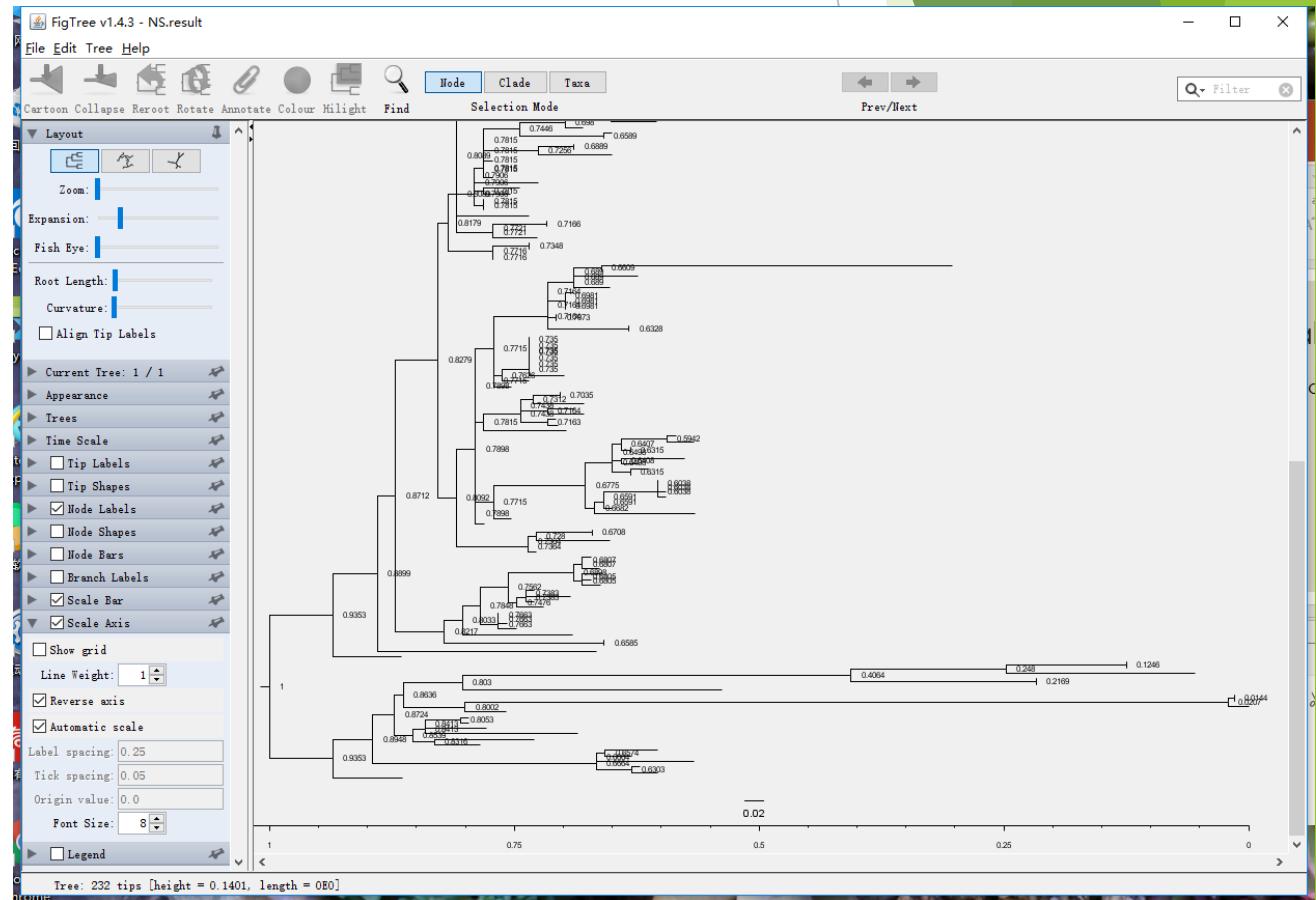


2.构建步骤

► 2.2.5 进化树展示

Scale axis

- 给scale加了一个坐标轴
- 这里面还是有很多可以调整的
- 可以和scale bar、time scale结合起来调整

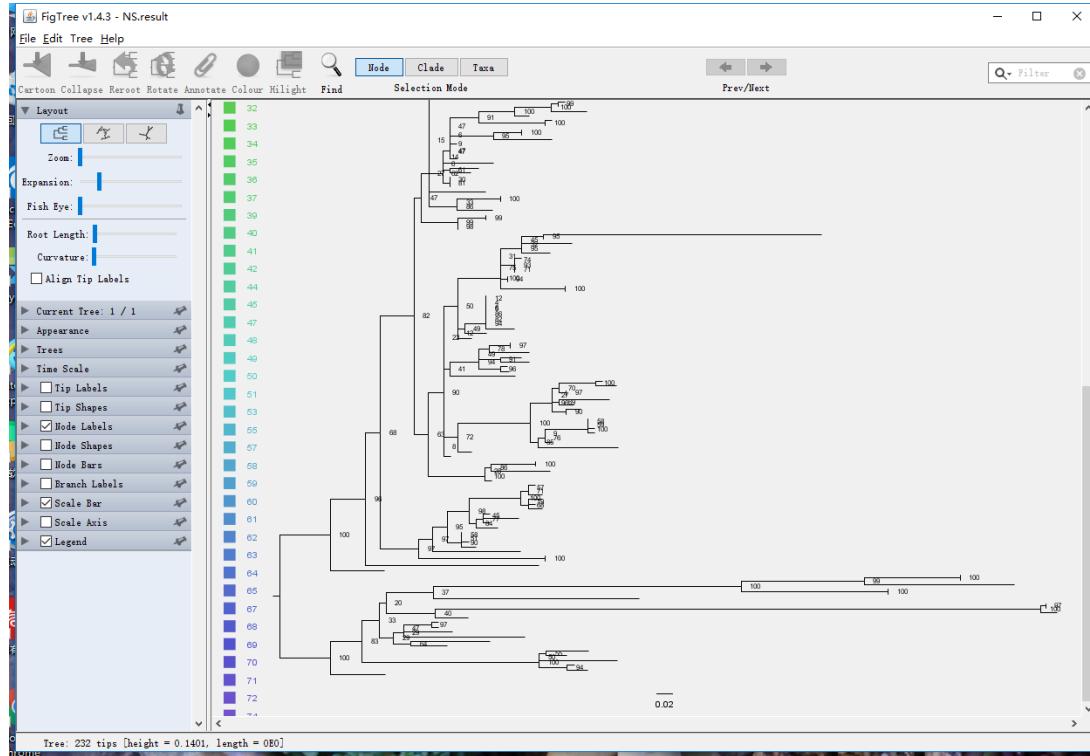


2.构建步骤

► 2.2.5 进化树展示

Legend

► 前面提到的appearance



► 显示一个标尺

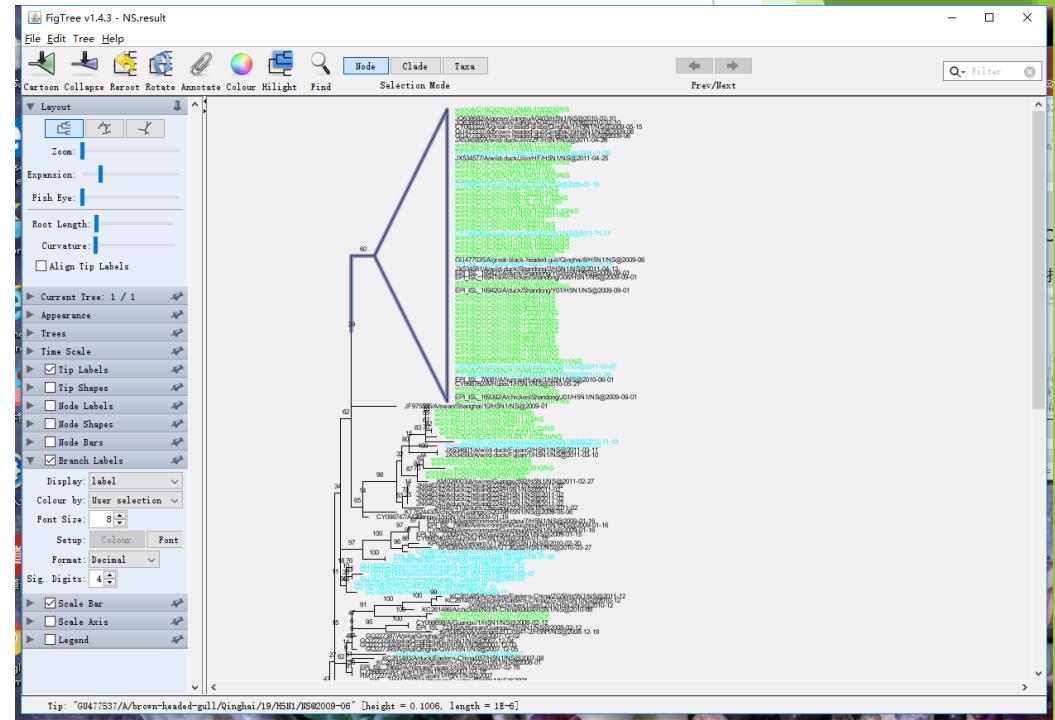
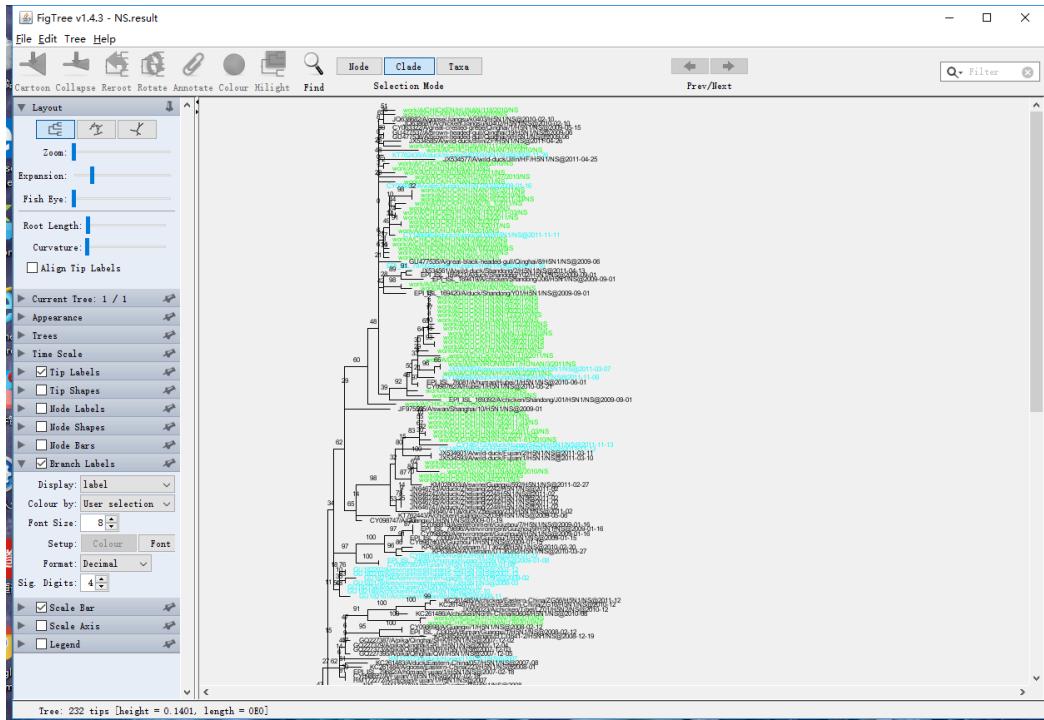


2.构建步骤

► 2.2.5 进化树展示

Cartoon

► 根据选择的枝来改变

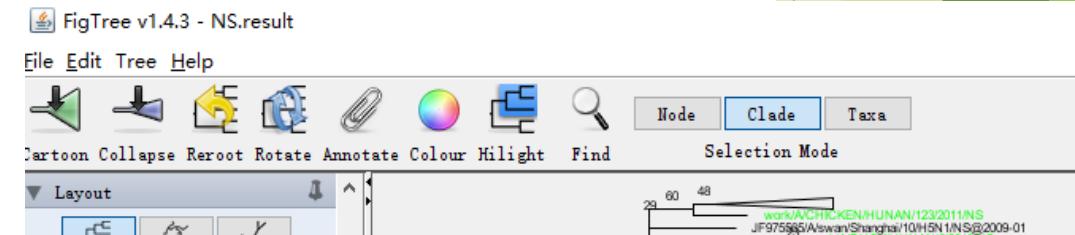
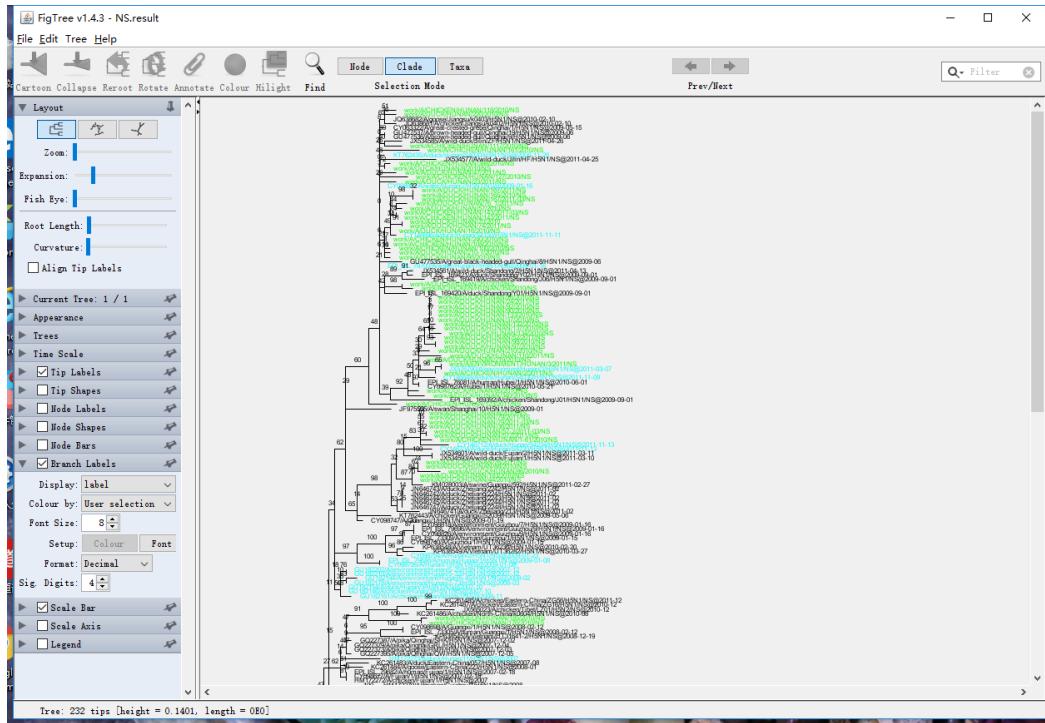


2.构建步骤

► 2.2.5 进化树展示

Collapse

- 可以把不想展示的枝隐藏

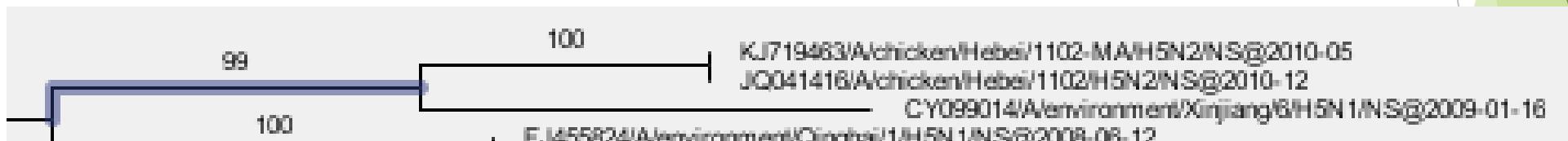


2.构建步骤

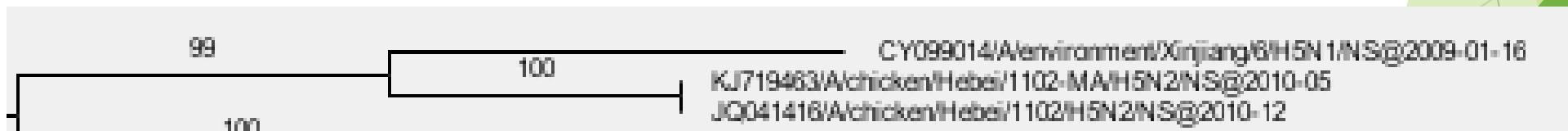
► 2.2.5 进化树展示

Reroot / Rotate

- Reroot可以在树上随意选择，来作为树的根
- Rotate可以选择的二叉树枝的上下顺序颠倒
- 例如：



- 结果：

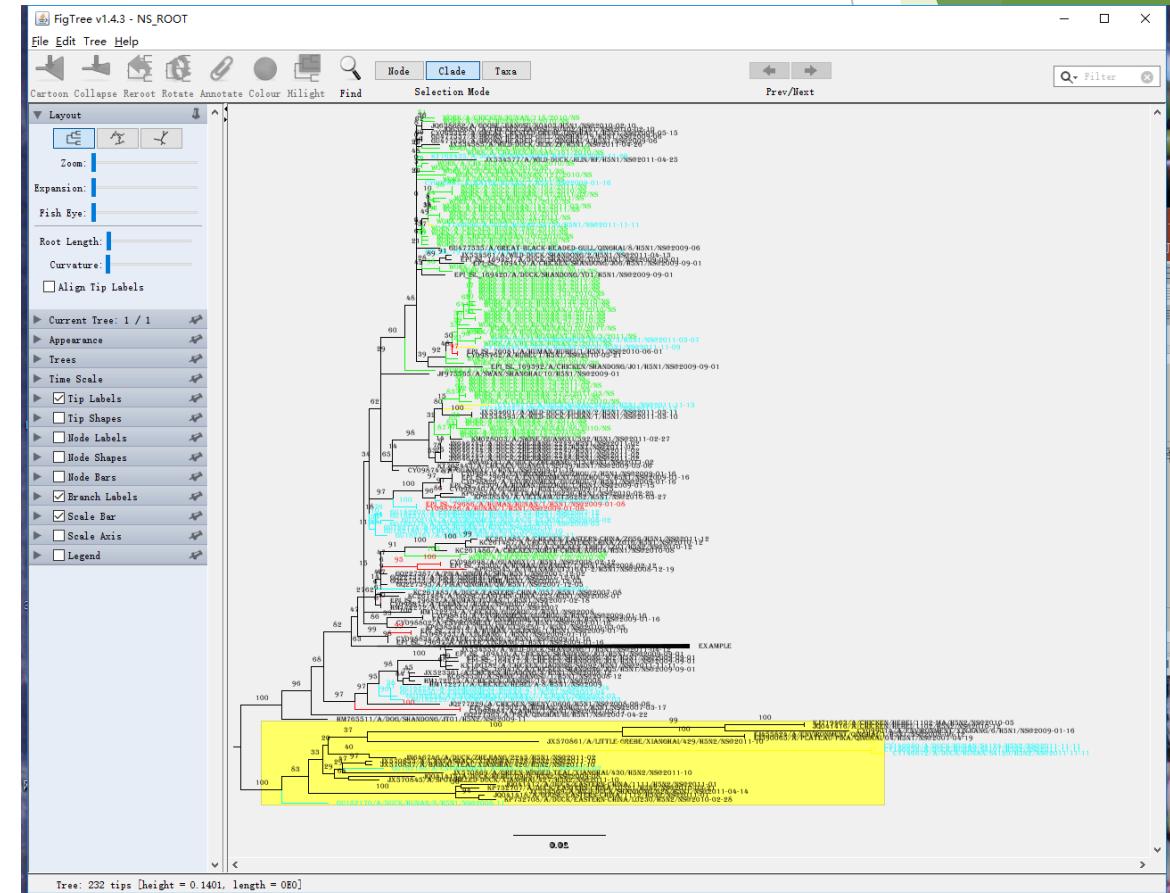


2.构建步骤

► 2.2.5 进化树展示

Annotate / color / highlight

- 可以分别为选择内容添加注释，改变颜色等。



2.构建步骤

► 2.3 系统发育树构建工具

PHYLIP	由美国华盛顿大学Felsenstein开发，可以免费下载，适用于绝大多数操作系统
PAUP	由美国simthsonion institute开发，仅适用于Apple-Macintosh和UNIX操作系统
MEGA	美国宾夕法尼亚州立大学MasatoshiNei开发的分子进化遗传学软件，图形化，集成的进行分析工具，不包括ML
MOLPHY	日本国立统计数理研究所开发，最大似然法构树
PAML	英国University college London开发，最大似然法构树和分子进化模型

2.构建步骤

► 2.3 系统发育树构建工具

PUZZLE	应用quarter puzzling方法（一种最大简约法）构建系统树
RaxML	用极大似然法建立 进化树 的软件之一，可以处理超大规模的序列数据，包括上千至上万个物种，几百至上万个已经比对好的碱基序列。
Phylogeny	欧洲生物信息研究所（EBI）的系统发育分析软件
PHYML	快速的ML建树工具
MrBayes	基于贝叶斯方法的建树工具
MAC5	基于贝叶斯方法的建树工具

2.构建步骤

▶ 2.3 系统发育树构建工具

- ▶ 构建NJ树，可以用PHYLIP或者MEGA。
- ▶ 构建MP树，可以使用PHYLIP或者MEGA。
- ▶ 构建ML树可以使用PHYML，速度快，同时构建ML树还可以用PHYLIP；RaxML则更为准确。
- ▶ 贝叶斯的算法以MrBayes为代表，不过速度比较慢。

Phylogeny / Alignment Tools

If there is a tool or a feature you need, please [let us know](#).

BAli-Phy on XSEDE (3.4.1-3.5.0) - BAli-Phy estimates multiple sequence alignments and evolutionary trees.

BEAST2 on XSEDE (2.1 - 2.6.3) - Bayesian Evolutionary Analysis by Sampling Trees - run on XSEDE

BEAST on XSEDE (1.8.0 - 1.10.4) - Bayesian Evolutionary Analysis by Sampling Trees - run on XSEDE

BEAST (OLD) on XSEDE (1.8.0 - 1.8.4) - Bayesian Evolutionary Analysis by Sampling Trees - run on XSEDE

BlastN (2.2.1) - Search DBs for Nucleotide Sequence similarity

Clearcut (1.0.9) - Fast Implementation of Relaxed Neighbor Joining

ClustalW (2.1) - Create Multiple Alignments from Sequences

Consense (Phylib 3.66) - Find A Consensus Tree

Cutadapt on XSEDE (2.10) - Remove adapter sequences from high-throughput sequencing reads

DPPDIV on XSEDE (1.0) - Estimating species divergence times and lineage-specific substitution rates on a fixed topology run on XSEDE

EPA-NG on XSEDE (0.3.7) - Bayesian Evolutionary Analysis by Sampling Trees - run on XSEDE

ExaBayes on XSEDE (1.5) - Bayesian Evolutionary Analysis by Sampling Trees - run on XSEDE

FastML on XSEDE (3.1) - Fast (Approximate) Maximum Likelihood tree construction - run on XSEDE

FastTreeMP on XSEDE (2.1.10) - Fast (Approximate) Maximum Likelihood tree construction - run on XSEDE

2. 构建步骤

Tool	Latest Version	Language	Compiler	Computer	Cores charged	MPI library	Threading	Other Software
BEAST	1.10.5 pre	Java/C++		Comet	2-84		Pthreads/CUDA	BEAGLE 3.1.2
BEAST2	2.6.3	Java/C++		Comet	1-84		Java/CUDA	BEAGLE 3.1.2
DPPDiv	1.0	C++	GNU	Comet	12		Pthreads	
EPA-NG	0.3.7	C++	GNU	Comet	24		OpenMP	
ExaBayes	1.5	C++	GNU	Comet	24,48, or 72	OpenMPI		
FastTree	2.1.10	C	Intel	Comet	3		OpenMP	
GARLI	2.01	C++	Intel	Comet	1 to 24	OpenMPI		
G-PhoCS	1.3.2	C	Intel	Comet	6 to 24		OpenMP	
Ima3	1.11	C++	Intel	Comet	1 to 48	MVAPICH2		
IQ-TREE	1.6.10/2.1.1	C++	Clang	Comet	6,12, or 24		OpenMP	SPRNG; Eigen3

<http://www.phylo.org/index.php/portal/codes/>

2. 构建步骤

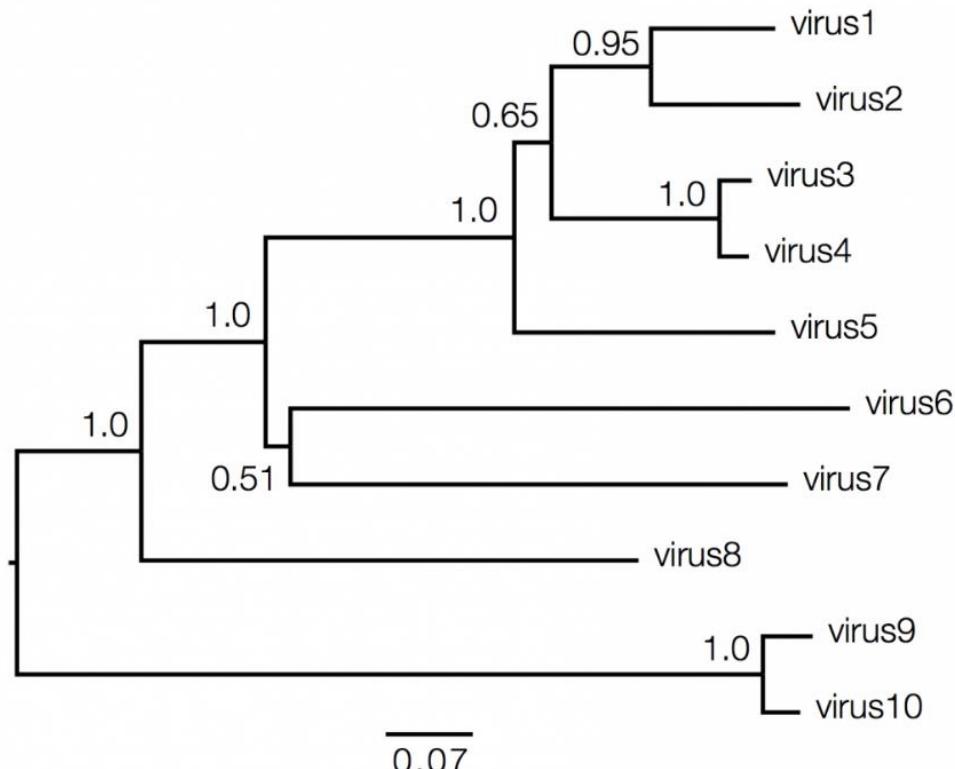
Tool	Latest Version	Language	Compiler	Computer	Cores charged	MPI library	Threading	Other Software
jModelTest2	2.1.10	Java		Comet	24		Java	PhyML 3.0
MAFFT	7.427	C	Intel	Comet	12		Pthreads	
ModelTest-NG	0.1.5	C++	GNU	Comet	24		Pthreads	PLL
MrBayes	3.2.7a	C/C++	Intel	Comet	2 to 24	OpenMPI		BEAGLE 3.1.2
Migrate-N	4.2.14	C	Intel	Comet	1 to 72	MVAPICH2		
ParallelStructure	2.3.4	C + R	Intel	Comet	12		R	
PartitionFinder2	2.1.1	Python	Intel	Comet	8		Python	RAxML, PhyML
PAUP	4.a165	C	Intel	Comet	1 or 4		Pthreads	
PhyloBayes	1.7b	C++	Intel	Comet	48	MVAPICH2		
RAxML	8.2.12	C	Intel	Comet	12, 24, or 48	OpenMPI	Pthreads	
RAxML-NG	0.9.0	C	Intel	Comet	6-48	OpenMPI	Pthreads	

3.系统发育树解读

3.理解进化树

► 支长

分支长度的单位通常是每个位点的核苷酸替换--即变化的数量或“替换”除以序列的长度(尽管它们可以被表示为%变化，即每100个核苷酸位点的变化数)。



<https://courses.lumenlearning.com/suny-wmopen-biology1/chapter/phylogenetic-trees/>
<https://artic.network/how-to-read-a-tree.html>

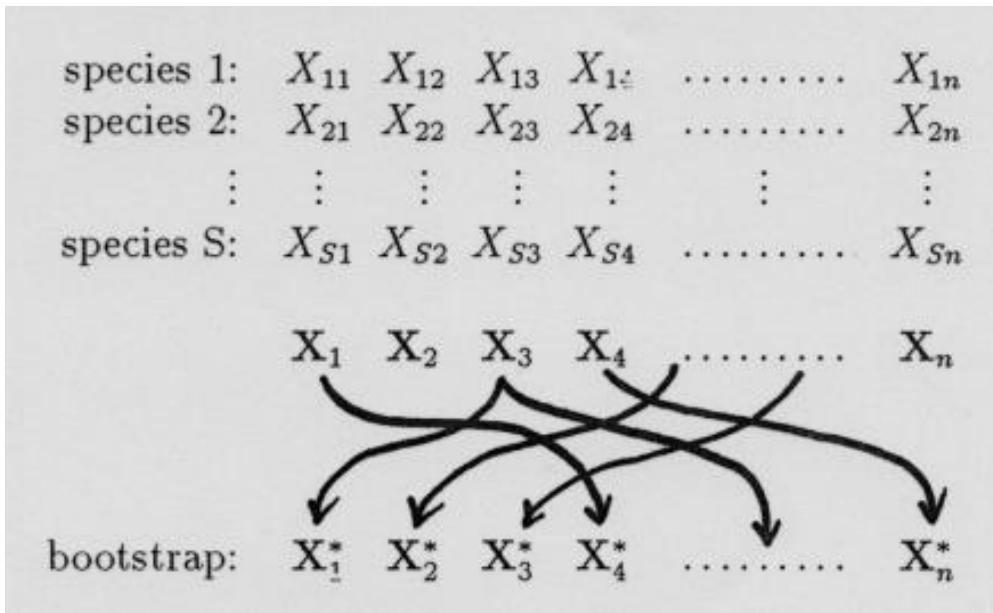
3.理解进化树

▶ 置信度

- ▶ Jack-knifing resampling
- ▶ Bootstrap
- ▶ Permutation

3.理解进化树

▶ 置信度——bootstrap (自展值)



初始对位

位点
人类
黑猩猩
倭黑猩猩
大猩猩
猩猩
苏门答腊猩猩
长臂猿

自展样本

位点
人类
黑猩猩
倭黑猩猩
大猩猩
猩猩
苏门答腊猩猩
长臂猿

1 2 3 4 5 6 7 8 9 10

N E N L F A S F I A

N E N L F A S F A A

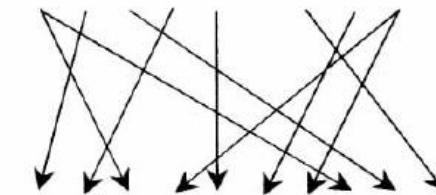
N E N L F A S F A A

N E N L F A S F I A

N E D L F T P F T T

N E S L F T P F I T

N E N L F T S F A T



2 4 1 9 5 8 9 1 3 7

E L N I F F I N N S

E L N A F F A N N S

E L N A F F A N N S

E L N I F F I N N S

E L N T F F T N D P

E L N I F F I N S P

E L N A F F A N N S

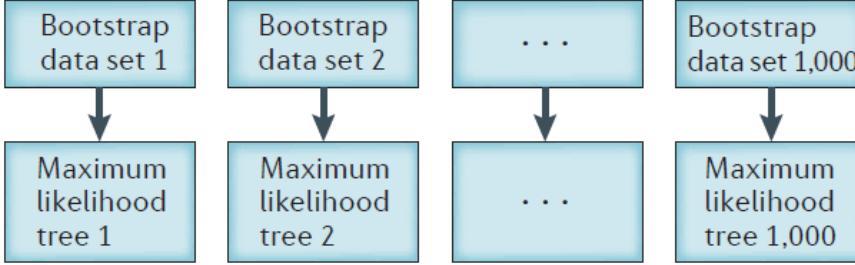
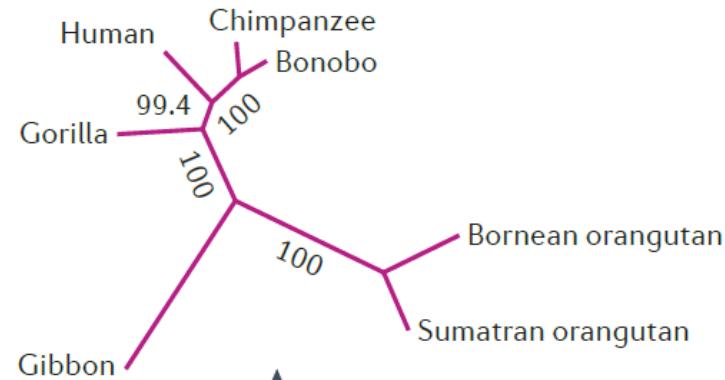
3.理解进化树

Box 3 | Sampling error in the estimated tree and bootstrap analysis

Sequence alignment

Human	NENLFASFIA	PTVLGLPAAV	...
Chimpanzee	NENLFASFAA	PTILGLPAAV	...
Bonobo	NENLFASFAA	PTILGLPAAV	...
Gorilla	NENLFASFIA	PTILGLPAAV	...
Bornean orangutan	NEDLFTPFTT	PTVLGLPAAI	...
Sumatran orangutan	NESLFTPFIT	PTVLGLPAAV	...
Gibbon	NENLFTSFAT	PTILGLPAAV	...

Maximum likelihood tree inferred from original data



Use maximum likelihood trees from the bootstrap data sets to place support values on the original maximum likelihood tree

图 6.7 用于计算 ML 树上分支支持值的自展方法。在 MTMAM 模型下分析 7 个猿类物种的线粒体蛋白质。所示为从原始数据出发,进行穷举式树搜索,从而鉴定 ML 树。自展则用于 ML 树分支上附加的支持值(百分率)。通过对原始对位排列中位点的重复抽样,产生了一批(如 1000 个)自展数据集(见图 6.6)。每个自展数据集均采取与原始数据集相同的方法分析,对每个自展样本产生一棵 ML 树。计算原始 ML 树上含有某一分支的自展树的比例并将其置于原始的 ML 树上。依 Whelan 等(2001)绘制。

3.理解进化树

▶ 外群

大多数的种系发生重建方法会产生无根树，但是观察树的拓扑结构无法识别树根应在哪一分支上。实际上，对于要证实哪一个分类单元的分支先于其他的分类单元，树根必须确定。

在无根树中设定一个根，最简单的方法是在数据集中增加一个外群（outgroup）。

外群是一种分类操作单元，且有外部信息表明外群在所有分类分类群之前就已分化。合适的外群与待分析的分类群关系不能相距太远，因为在比较关系较远的物种时，系统发出的信号会降低，这是核苷酸替换饱和的结果。**使用一个以上的外群通常可以进一步改善推导的树状拓扑的准确度。**

所谓的外类群就是与你研究的序列关系极为密切的序列，且外类群能很好的聚为一支（若外类群不止一条序列），**若研究的是演化历史，一般应选择比目标序列具有较早进化历史的序列作为外类群。**

另一种可选的引入外群的方法是，使用两套相同的、同时存在于待分析的所有分类操作单元中的**并系同源基因**。在这种方法中，第一个并系同源基因群中的基因可以成为第二个并系同源基因群中基因的外群。这种确定的系统已用于确定tree of life的第一层分支，树根可以置于通向生命树中细菌、古细菌以及真核细胞中任一分枝上。当使用单一外群时，根可以置于通向外群的分支上。另外，**若使用多个外群，根必须置于连接外群和内群的分支上。**

如果是鉴定物种，最好选一个外群。

在缺少一个合适的外群时，根大约可以置于两个分类操作单元间最长支的中点上。这种确定根的方法叫做中点定根（midpoint rooting），当在树中所有分支的进化速度大致相同而且实际的外群与其它分类群间的支的长度不太短时，这种方法相当准确，但是**中点生根这种方法慎用**，它有一个假设前提：假设两个最不同的谱系以相同的速度进化。显然，这个假设现实中很可能不成立。

Appendix: 学习指南

<http://evomics.org/>

https://evolution.genetics.washington.edu/phylip/newick_doc.html

<https://www.britannica.com/science/phylogeny/Taxonomic-systems>

<https://artic.network/how-to-read-a-tree.html>

<https://www.yuque.com/wusheng/gw7a9p/ke1mdu>

https://en.wikipedia.org/wiki/Models_of_DNA_evolution

<https://yulab-smu.top/treedata-book/chapter1.html>

<http://www.iqtree.org/doc/Substitution-Models>

<http://www.chenlianfu.com/?p=2225>

<https://www.plob.org/article/1107.html>

Appendix: 作业

1. 建立DNA系统发育树

使用Maximum Likelihood方法建立流感病毒NS基因的系统发生树。

文件: NS.fasta

推荐网站:

EMBL-EBI: <https://www.ebi.ac.uk/>

CIPRES: <http://www.phylo.org/>

Bootstrap 100次, 展示bipartitionLabel tree

在Figtree中尝试调整各种参数, 观察结果。

2. 蛋白质分析

- 序列获取 自行在相关数据库中检索
- 要求

- 用在线工具预测其分子量, 并与数据库中的记录做对比;
- 预测蛋白质的等电点, 不同工具预测的结果是否有差异;
- 预测蛋白质是否存在信号肽, 并与真实情况做对比;
- 预测蛋白质是否具有跨膜结构, 记录跨膜结构的信息, 与真实情况做对比;
- 预测蛋白质的二级结构, 并与真实情况做对比;
- 利用同源模建, 预测蛋白质的三级结构。

说明

1. 课后作业共两道题, 将操作步骤按顺序截图粘贴到word文件中, 上传至课程网站
2. 进化树相关序列用老师提供的, 蛋白质的分析自行检索序列

Thanks

Folders

Total Storage: 4.10 MB

- test1
 - Data (1)
 - Tasks (0)

All Data

There is currently 1 data item in this folder.

[Upload Data](#)

« ‹ Page 1 of 1 › »

20 ▾ records on each page

User Data

<input type="checkbox"/> Select all	User Data ID	Label	Size	Data Format	Date Created	Parse Status
<input type="checkbox"/>	2656061	Italy_5-9_aln.fasta	4.10 MB	Unknown	11/30/20, 01:15	Queued

[Move](#) ▾ selected to [test1](#) ▾ [GO](#)

[Delete Selected](#)

[Transfer Selected to CIPRES Share](#)

Folders

Total Storage: 4.10 MB

- test1
 - Data (1)
 - Tasks (0)

Create new task

Task Summary Select Data Select Tool Set Parameters

You may edit your task using the tabs above.

Current CPU Hr Usage: 0 [Explain this?](#)

Description

Description

Input

Select Input Data

Tool

Select Tool

Input Parameters

Set Parameters

[Save Task](#)

[Save and Run Task](#)

[Discard Task](#)

Saved tasks can be run later from the task list

XSEDE tasks are limited to 168 hours. Non-XSEDE tasks are limited to 72 hours.