

# 用PartitionFinder2 筛选系统发育分析分隔模型

张金龙

嘉道理农场暨植物园植物保育部

[jinlongzhang01@gmail.com](mailto:jinlongzhang01@gmail.com)

核苷酸序列在不同位点有不同的突变速率。核苷酸序列又分为编码基因和非编码基因。编码基因中，密码子第一第二位往往较为稳定，第三位往往变异速率较高。非编码基因因为受到的选择压力一般较小，所以往往可保留更多突变。不同基因以及不同位点的突变速率不同，可能对所推断进化树的稳定性有很大影响。所以，在多基因建立进化树过程中，设置分隔模型就显得很重要。但是分隔模型怎样设置才算合理呢？在PartitionFinder软件研发之前，研究人员一般通过将不同基因分开，例如 gene1、gene2、gene3，再将每个基因的不同位点分开如 gene1\_1、gene1\_2、gene1\_3作为分隔模型，以提高进化树的稳健性。

RAxML、MrBayes、BEAST等常用系统发育软件都支持分隔模型，但并不能帮忙确定最优化的分隔模型设定方案。设置的分隔模型过多，则拟合的参数会过多，造成结果不准确。设置的分隔模型过少，不设置分隔模型，设置的不合理，也会造成进化树不准确。很多学者已经意识到这个问题，但是一直苦于没有很好的应对方法。

模型选择中的简约理论认为，赤迟信息量AIC或者BIC最小的模型是最优的模型。AIC结合模型的精度以及所要估计的参数数量。如果精度已经到达一定程度，进一步增加模型的参数已经不能再显著提高似然值Likelihood的情况下，可以认为已经找到了最优模型。但是这就要对每一种分隔模型所得的结果计算Likelihood，并进行参数估计几乎是不可能的。首先，对碱基比对矩阵计算Likelihood是十分耗费时间的。其次，分隔模型的各种组合的数量呈几何级数增长，如果通过几个基因建树，则可能的分隔模型的数量已经超出了大部分计算机的计算能力。此时就需要引入分隔模型的启发式搜索 Heuristic Search。

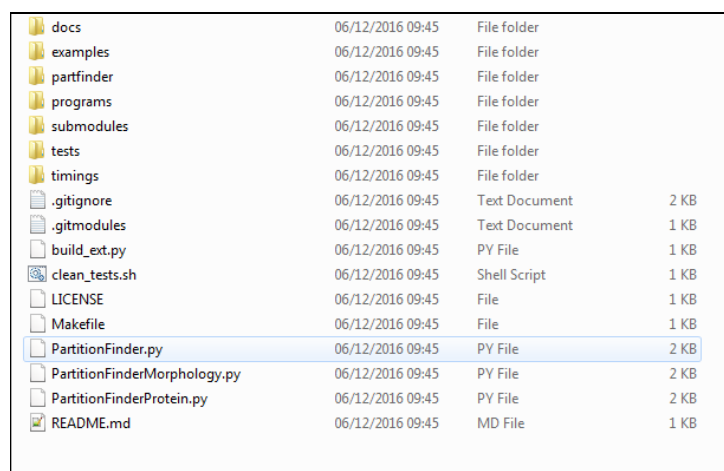
PartitionFinder的作者们从理论上解决了以上的问题，并通过Python语言实现了相应的算法 (Lanfear et al., 2012)。在该研究中，作者进一步证明，PartitionFinder所得的分隔模型比之前的简单处理更加合理。不仅如此，PartitionFinder在获得最优化分隔方案后，

同时会给出每个分隔模块所对应的最优进化模型，所以Modeltest、jModeltest以及ProtTest等软件都已经被PartitionFinder所超越。事实上，2012年，Lanfear介绍PartitionFinder的论文发表后，在google scholar上已经被引用了1748次(截至2017年4月7日)。本文简述PartitionFinder的安装和使用。

## 1. PartitionFinder下载和安装

### 1.1 PartitionFinder下载

PartitionFinder的下载网址 <http://www.robertlanfear.com/partitionfinder/>，用WinRAR或者7zip解压缩。



docs	06/12/2016 09:45	File folder	
examples	06/12/2016 09:45	File folder	
partfinder	06/12/2016 09:45	File folder	
programs	06/12/2016 09:45	File folder	
submodules	06/12/2016 09:45	File folder	
tests	06/12/2016 09:45	File folder	
timings	06/12/2016 09:45	File folder	
.gitignore	06/12/2016 09:45	Text Document	2 KB
.gitmodules	06/12/2016 09:45	Text Document	1 KB
build_ext.py	06/12/2016 09:45	PY File	1 KB
clean_tests.sh	06/12/2016 09:45	Shell Script	1 KB
LICENSE	06/12/2016 09:45	File	1 KB
Makefile	06/12/2016 09:45	File	1 KB
PartitionFinder.py	06/12/2016 09:45	PY File	2 KB
PartitionFinderMorphology.py	06/12/2016 09:45	PY File	2 KB
PartitionFinderProtein.py	06/12/2016 09:45	PY File	2 KB
README.md	06/12/2016 09:45	MD File	1 KB

图1 PartitionFinder的结构

其中docs为说明文档

\Examples为示例文件，包含DNA序列，形态以及蛋白质三个例子

\partfinder文件夹为python脚本

\Programs 文件夹下为phyml可执行文件，用来计算likelihood

\Submodules 文件夹下有raxml文件夹，但是该文件夹为空。作者并未交代这个文件夹的内容。

\tests 为作者开发时的测试文件

PartitionFinder.py 为检测DNA序列的Python脚本

PartitionFinderMorphology.py 为检测形态数据的Python脚本

PartitionFinderProtein.py 为检测氨基酸序列的Python脚本。

三个脚本根据不同的数据类型分别调用，不能混用。

## 1.2. 运行平台Anaconda的安装

Partition Finder是用Python开发的，依赖于 numpy、pandas、pytables、pyparsing、scipy、sklearn 等若干程序包。在Windows下，安装这些程序包往往较为麻烦。作者Lanfear推荐已经预装了这些程序包的Anaconda Python。Anaconda Python下载地址：<https://www.continuum.io/Downloads>。PartitionFinder只能在 Python2.7上运行，不能使用Python3。双击开始安装。以下路径会被直接添加到系统路径: C:\ProgramData\Anaconda2; C:\ProgramData\Anaconda2\Scripts; C:\ProgramData\Anaconda2\Library\bin;

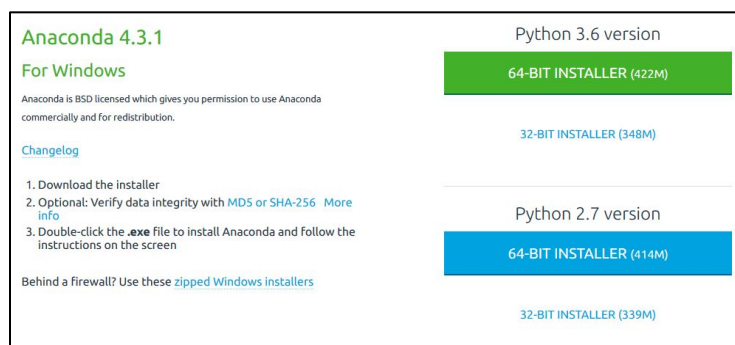


图2 下载Anaconda Python2.7

## 2. PartitionFinder输入文件的格式

Partition Finder运行需要两个文件, phylic文件以及partition\_finder.cfg文件。phylic文件包含所要读取的序列, partition\_finder.cfg包含模型筛选时设定的参数。

### 2.1 PHYLIP文件

phylic文件可以通过MUSCLE, CLUSTAL, MAFFT等比对序列的软件生成。如果是多基因序列，一般对每个基因分别比对，手工调整之后，再通过phylotools (<https://github.com/helixcn/phylotools>)或者Geneious等软件生成supermatrix，以phylic格式导出即可。关于PHYLIP格式的详细定义，可以参考: <http://www.atgc-montpellier.fr/phyml/usersguide.php?type=phylic>。PartitionFinder可以接受relaxed phylic, 其中的物种名最长可以达到100个字符。

10 949	
AD00P068	SLMLLISSSIVENGAGTGWTVYPPLSSNIAHSGSSVDLAIFSLHLA
SY03A501	SLMMLISSMIVENGAGTGWTVYPPLSSNIAHSGSSVDLTIFSLHLA
SY03A503	SLMLLISSSIVENGAGTGWTVYPPLSSNIAHSGSSVDLAIFSLHLA
TDA99Q985	SLMLLISSSIVENGAGTGWTVYPPLSSNIAHSGSSVDLAIFSLHLA
TDA99Q986	SLMLLISSSIVENGAGTGWTVYPPLSSNIAHSGSSVDLAIFSLHLA
UK99W804	SLMLLISSSIVENGAGTGWTVYPPLSSNIAHSGSSVDLAIFSLHLA
UK99W805	SLMLLISSSIVENGAGTGWTVYPPLSSNIAHSGSSVDLAIFSLHLA
YJ02Q703	SLMLLISSSIVENGAGTGWTVYPPLSSNIAHSGSSVDLAIFSLHLA
ZD99S303	SLMLLISSSIVENGAGTGWTVYPPLSSNIAHSGSSVDLAIFSLHLA
ZD99S305	SLMLLISSSIVENGAGTGWTVYPPLSSNIAHSGSSVDLAIFSLHLA

图3 Phylip文件

## 2.2 partition\_finder.cfg 文件

partition\_finder.cfg配置文件为纯文本文件。PartitionFinder可以处理三种类型的数据，分别为(1)编码区DNA序列(2)氨基酸序列或非编码区DNA序列(3)形态数据，配置时各有不同。其中编码区DNA序列的配置要提供每个片段的(1)起始和终止位置;(2) 编码位置(1, 2, 3)。氨基酸序列和非编码区DNA序列，只需要提供每个基因片段的起始和终止位置。形态数据，无需直接指定模块，而在计算时采用k-means聚类，以划分成不同的组别。

```
## ALIGNMENT FILE ##
alignment = test.phy;

## BRANCHLENGTHS: linked | unlinked ##
branchlengths = linked;

## MODELS OF EVOLUTION: all | allx | mrbayes | beast | gamma | gamma1 | <list> ##
models = GTR, GTR+G, GTR+I+G;

# MODEL SELECTION: AIC | AICc | BIC #
model_selection = aicc;

## DATA BLOCKS: see manual for how to define ##
[data_blocks]
Gene1_pos1 = 1-789\3;
Gene1_pos2 = 2-789\3;
Gene1_pos3 = 3-789\3;
Gene2_pos1 = 790-1449\3;
Gene2_pos2 = 791-1449\3;
Gene2_pos3 = 792-1449\3;
Gene3_pos1 = 1450-2208\3;
Gene3_pos2 = 1451-2208\3;
Gene3_pos3 = 1452-2208\3;

## SCHEMES, search: all | user | greedy | rcluster | rclusterf | kmeans ##
[schemes]
search = greedy;
```

图4 partition\_finder.cfg 文件的内容

### 3. 主要运行步骤

运行partitionFinder时，需要将phylip文件以及配置文件放在同一文件夹下。然后直接调用对应的python脚本(PartitionFinder.py, PartitionFinderMorphology.py , 或 PartitionFinderProtein.py )即可。生成的结果会自己保存在相应的文件夹下，并生成若干新文件以及文件夹。根据分区的数目，运行PartitionFinder的策略需要做相应调整

#### 3.1对10个分区进行筛选

对于10个DNA片段以下的DNA编码DNA序列，主要的运行步骤为：

(1) 在.cfg文件中指明基因片段的起始和终止位置。

(2) 设定分隔模型筛选的主要参数

```
branchlengths = linked;
models = all;
model_selection = aicc;
search=greedy;
```

(3) 在cmd中运行PartitionFinder

```
cd "PartitionFinder.py 所在的文件夹"
```

然后输入

```
python "<PartitionFinder.py>" "<InputFoldername>"
```

程序会自动运行，并且输出结果：

例如 `python PartitionFinder.py -v "examples/nucleotide"`

```
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Users\j1zhang>cd C:\Users\j1zhang\Desktop\partitionfinder-2.1.1

C:\Users\j1zhang\Desktop\partitionfinder-2.1.1>python PartitionFinder.py -v "examples/nucleotide"
INFO | 2017-04-10 10:26:25,223 | main | ----- PartitionFinder 2.1.1 -----
INFO | 2017-04-10 10:26:25,224 | main | You have Python version 2.7
INFO | 2017-04-10 10:26:25,224 | main | Command-line arguments used: PartitionFinder.py -v examples/nucleotide

INFO | 2017-04-10 10:26:25,226 | config | ----- Configuring Parameters -----
INFO | 2017-04-10 10:26:25,226 | config | Setting datatype to 'DNA'
INFO | 2017-04-10 10:26:25,226 | config | Setting phylogeny program to 'phym1'
INFO | 2017-04-10 10:26:25,226 | config | Program path is here C:\Users\j1zhang\Desktop\partitionfinder-2.1.1\programs
DEBUG | 2017-04-10 10:26:25,226 | config | Setting rcluster-percent to 10.00
DEBUG | 2017-04-10 10:26:25,226 | config | Setting rcluster-max to -987654321
INFO | 2017-04-10 10:26:25,226 | config | Setting working folder to: 'C:\Users\j1zhang\Desktop\partitionfinder-2.1.1\examples\nucleotide'
DEBUG | 2017-04-10 10:26:25,227 | config | About to search for partition_finder.cfg file...
INFO | 2017-04-10 10:26:25,227 | config | Loading configuration at '.\partition_finder.cfg'
INFO | 2017-04-10 10:26:25,233 | config | Setting 'alignment' to 'test.phy'
INFO | 2017-04-10 10:26:25,236 | config | Setting 'branchlengths' to 'linked'
INFO | 2017-04-10 10:26:25,236 | parser | You set 'models' to: GTR, GTR+G, GTR+I+G
INFO | 2017-04-10 10:26:25,244 | model_load | This analysis will use the following 3 models of molecular evolution
INFO | 2017-04-10 10:26:25,244 | model_load | GTR, GTR+G, GTR+I+G
INFO | 2017-04-10 10:26:25,246 | config | Setting 'model_selection' to 'aicc'
DEBUG | 2017-04-10 10:26:25,246 | subset | Created Subset(..)
DEBUG | 2017-04-10 10:26:25,247 | subset | Created Subset(..)
DEBUG | 2017-04-10 10:26:25,249 | subset | Created Subset(..)
```

图5 运行partitionFinder

## 3.2 对10-100个分区进行筛选

超过100个等位基因的序列，需要用贪婪算法，基本步骤如下

- (1) 设定每个基因的起始位点以及结束位点，编码位置(1, 2, 3)等
- (2) 设定.cfg文件
- (3) 用raxml计算Likelihood

对于DNA为

```
python "<PartitionFinder.py>" "<InputFoldername>" --raxml
```

对于氨基酸为

```
python "<PartitionFinderProtein.py>" "<InputFoldername>" -raxml
```

## 3.3 超大数据(1000个)的分区筛选

超大型数据的位点 (1000个以上)，贪婪算法已经太慢，推荐使用其他算法，以猜测最优分隔模型设置方案(Lanfear et al

2014)，基本步骤如下

- (1) 设定每个基因的起始位点以及结束位点，编码位置(1, 2, 3)等
- (2) 设定.cfg文件

特别是 `search = rcluster`

(3)在cmd中运行PartitionFinder

```
python "<PartitionFinder.py>" "<InputFoldername>" --raxml
```

如仍然过慢， 则更改recluster-max 参数

```
python "<PartitionFinder.py>" "<InputFoldername>" --raxml --  
recluster-max 100
```

## 4各参数的意义和设定

### 4.1 partition\_finder.cfg文件的基本结构

partition\_finder.cfg配置文件的文件名不要更改。 配置文件的基本格式如下， 其中# 为注释。

```
# ALIGNMENT FILE # 指出比对数据phylip的文件名  
alignment = test.phy;
```

```
# BRANCHLENGTHS # 在优化进化树枝长用于Likelihood计算时，是否各枝长一起优化。  
branchlengths = linked;
```

```
# MODELS OF EVOLUTION # 所筛选模型的范围， 模型筛选的标准，一般用aicc  
models = all;  
model_selection = aicc;
```

```
# DATA BLOCKS # 设定数据分区  
[data_blocks]  
Gene1_pos1 = 1-789\3;  
Gene1_pos2 = 2-789\3;  
Gene1_pos3 = 3-789\3;
```

```
# SCHEMES # [schemes] 模型筛选的算法  
search = greedy;  
# user schemes (see manual) 其他设定的参数
```

其他选项用竖线 | 分隔

## 4.2 各参数的意义

4.2.1 alignment : phylip 文件名

4.2.2 branchlengths: linked | unlinked 枝长是否相互关联

即对每个分隔模式是否独立估计进化树的枝长(目前只有BEAST, MrBayes, RAxML支持独立估算枝长)

4.2.3 models 所要筛选的模型

指定要筛选的模型的种类。在PartitionFinder2运行结束后, 将会获得每个分隔模型的对应最优模型。一般选择all

4.2.3.1 models模型筛选的范围

#####

默认情况下, 推荐设定models=all; 如果用户需要使用更多模型, 可设定models=allx, 但是最好不要这么做, 这是因为对少数几个模型进行优化要耗费很多时间, 多数情况下得不偿失。

(1)对于离散性状, 例如01性状(BINARY), 以及无序质量性状(MULTISTATE)BINARY+G, BINARY+G+A, MULTISTATE+G, MULTISTATE+G+A, 模型详情需要参考RAxML的说明书。注意: 性状数据模型不能计算AIC。(2)对于DNA序列, models=all 比较内置的56种模型。由14种基本模型为基础JC, K80, TrNef, K81, TVMef, TIMef, SYM, F81, HKY, TrN, K81uf, TVM, TIM, and GTR, 同时参考是否用Gamma或者I来表征不同位点进化速率的差异。但是在分析DNA数据时, 如果添加了 --raxml commandline option 选项, 则只计算GTR, GTR+G, GTR+I+G三种模型的AIC的值。(3)对于氨基酸序列, 默认情况下(使用PHYML计算), 要比较的基本模型为 LG, WAG, MTREV, DAYHOFF, DCMUT, JTT, VT, BLOSUM62, CPREV, RTREV, MTMAM, MTART, HIVB, HIVW, 每种又分成8种情况 LG, LG+F, LG+G, LG+G+F, LG+I, LG+I+F, LG+I+G, LG+I+G+F, 其中F为是否使用经验氨基酸频率(with or without empirical amino acid frequencies)

如果开启了RAxML选项则会比较 (LG, WAG, MTREV, DAYHOFF, DCMUT, JTT, VT, BLOSUM62, CPREV, RTREV, MTMAM, MTART, HIVB, HIVW, MTZOA, PMB, JTTDCMUT, FLU, STMTREV, DUMMY, DUMMY2)等21种模型, 每个模型考虑6种情况。



所筛选模型范围进一步扩大 `models = allx`后，对于DNA序列，使用PHYML时，会比较84种模型；使用RAxML会比较6种模型。对于氨基酸序列，使用PHYML不能获得结果，所以只能使用RAxML，比较的模型数量为195个。这种情况下，所有的碱基频率或者氨基酸频率都是用极大似然估计获得的。

#### 4.2.3.2从指定的模型中筛选

如果后续用mrBayes或者beast做分析，则需要设定models为`models = mrbayes`; `models = beast`;以便只从这两个软件中有的模型中筛选。

```
#####
```

```
models = gamma; models = gammai 告诉Partition Finder只考虑有Gamma以及I 的情况
```

```
##### 指定DNA模型的筛选范围
```

```
models = JC, JC+G, HKY, HKY+G, GTR, GTR+G;
```

```
##### 指定核苷酸模型的筛选范围
```

```
models = LG, LG+G, LG+G+F, WAG, WAG+G, WAG+G+F;
```

```
#####模型选择的标准，作者推荐 AICc
```

```
model_selection: AIC | AICc | BIC
```

### 4.2.3 设定分区

```
[data_blocks]
```

```
Gene1_codon1 = 1-1000\3;
```

```
### 1-1000位点， 从第1个碱基开始，步长为3， 所有的碱基
```

```
Gene1_codon2 = 2-1000\3;
```

```
### 2-1000位点， 从第2个碱基开始，步长为3， 所有的碱基
```

```
Gene1_codon3 = 3-1000\3;
```

```
### 3-1000位点， 从第3个碱基开始，步长为3， 所有的碱基
```

```
intron = 1001-2000;
```

如果是为MrBayes筛选分区模型，可在每行前增加 `charset`。

```
charset Gene1_codon1 = 1-1000\3;
charset Gene1_codon2 = 2-1000\3;
charset Gene1_codon3 = 3-1000\3;
charset intron = 1001-2000;
```

#### 4.2.4 搜寻策略[schemes]

```
search: all | greedy | rcluster | rclusterf | hcluster | kmeans
| user
```

其中 search = all 一般用于 12 个分区以下

search = greedy 10-100个分区

search = rcluster 100个以上的分区

search = rclusterf 只适用于RAxML选项开启时，尤其适用于有若干分析需要数据很大，同时ML需要优化很长时间时，可以减少等待的时间。

search = hcluster 在研究中不推荐使用，仅为了研究的目的而保留。

search = user

同一对括号中的基因表示将在一起分析

```
together      = (Gene1_codon1, Gene1_codon2, Gene1_codon3,
intron);
intron_123    = (Gene1_codon1, Gene1_codon2, Gene1_codon3)
(intron);
intron_12_3   = (Gene1_codon1, Gene1_codon2) (Gene1_codon3)
(intron);
separate      = (Gene1_codon1) (Gene1_codon2) (Gene1_codon3)
(intron);
```

#### 4.2.5用户自定义进化树

在simulation研究中，可能假设已知进化树的结构，用来做进一步计算。此时可以指定进化树。在之后的分析中，进化树的拓扑结构被保留，但是枝长会根据GTR+I+G模型重新估计。

```
# ALIGNMENT FILE #
alignment = test.phy;
user_tree_topology = tree.phy
```

## 5.输出结果

生成的结果将位于同一文件夹下的analysis文件夹中，该文件夹包含以下文件

best\_schemes.txt

纯文本文件，包括所筛选到的最优化分隔模型划分方案，以及每个区块对应的最优模型。同时给出RAxML和Nexus的分隔模型文件。

subsets文件夹: 每个区块所对应的AICc值以及相应的筛选过程

schemes文件夹: 所分析的分隔模型划分方案，详细的分析过程

## 6.命令行选项

在运行PartitionFinder时，可以在命令行设定一些参数，具体如下：

`--all-states`

仅用于k-means选项开启的情况，用于限定状态的数量。

`--force-restart`

删除之前的分析，完全重新开始

`--min-subset-size`

设定子集的大小。只用于k-means算法(这个选项在说明书中交待得不是很清楚)

`--no-ml-tree`

设定此选项后，PartitionFinder将从Neighbour Joining(PHYML)或者Parsimony树(RAxML)开始。

`--processors N, -p N`

多线程计算，用多个CPU核。默认情况下，PF会使用多个核，如果要控制使用一定数量的核，则应该设定这个选项。

`--quick, -q`

让PF停止写一些记录文件，如果处理的数据特别大，有可能会提升速度。特别是在使用贪婪算法的情况下。

`--raxml`

告诉PF使用RAxML帮忙寻找最优分隔模型

`--rcluster-max N`

设定cluster的数量

--rcluster-percent N

默认值--rcluster-max to 1000

默认值--rcluster-percent to 10

一般已经可以保证搜寻到正确的分隔模型

--save-phylofiles

保存数据分析过程中生成的进化树， 主要用来纠错

--weights "W rate , W base , W model , W alpha "

只用于--raxml 或 the hcluster or rcluster , 分别表示几个参数的权重。(1)the overall rate for a subset, (2)the base/amino acid frequencies, (3)the model parameters, and (4)the alpha parameter (which describes gamma distributed rates across sites)

## 参考文献

Lanfear, R., Calcott, B., Ho, S. Y., & Guindon, S. (2012). PartitionFinder: combined selection of partitioning schemes and substitution models for phylogenetic analyses. *Molecular biology and evolution*, 29(6), 1695-1701.

## 附录

### DNA序列碱基替换模型的三大要素

#### (1)碱基频率的估算

模型需要提供ATCG频率的估计方法， 目前有四种方法：

equal 直接假定ATCG的频率相同

model 是从其他数据读取不同氨基酸的频率 (仅用于氨基酸序列)

empirical 是直接从数据计算ATCG的频率

ML是用极大似然法估计ATCG的频率

注意： 在模型后添加 +X， 同样开启ML方法估计ATCG频率， 例如‘F81+X’， 不过一般用户都是用empirical的方法估计频率， 而且其结果与ML的结果相差无几。

(2)替换相对速率矩阵 (6个参数)用来表征碱基之间的转换速率 Relative rates of substitution, 其中GTR模型最为复杂， 需要估计所有6个参数

#### (3)不同位点进化速率的差异

分成以下几种情形

[1] 各位点速率恒定

[2] 部分位点进化树进化树速率恒定不变 (I 用来表示进化速率恒定位点的比例) GTR +

I

[3] 假设各位点的进化速率符合Gamma分布(因为Gamma分布要估计的参数少, 同时曲线有足够的变化, 能够较好地拟合多数情况, 所以用Gamma分布) GTR + GAMMA

[4] 同时考虑各位点的进化树速率的差异以及恒定位点的比例。GTR + GAMMA + I