MCMCTree チュートリアル

Mario dos Reis and Ziheng Yang
(訳：井上 潤)

April 9, 2013

MCMCTree はベイズ法によって種の系統関係を推定するソフトウェアです．さまざまな分子時計モデル [4,5,9] のもとで，化石制約 [9] を与えて年代を推定します．分子年代推定の一般的な説明は Yang [7] を参照してください．MCMCTree はインプットファイルとして，配列アライメント (塩基かアミノ酸配列)，化石制約付きの系統樹，および解析全体を制御するコントロールファイル (デフォルトでは mcmctree.ctl という名前になっています) を読み込みます．MCMCTree は PAML パッケージ [8] の一部として配信されています．

このマニュアルは，読者が Windows か Unix のコマンドライン (Linux や MacOS など) を用いて基本的な解析を行うことができると仮定して作成しています．以下から PAML パッケージをダウンロードし，インストールを行ってください :

http://abacus.gene.ucl.ac.uk/software/paml.html

使っているプログラムが最新のバージョンかどうか確認してください．現時点 (2013 年 3 月) では 4.7 です．Windows は *.exe ファイルで解析を直接行うことができます．Unix ユーザはプログラムをコンパイルする必要があります．PAML のウェブサイトにある説明に従って，オペレーティングシステムの PATH を設定してください．この設定を行っておけば，プログラムのあるディレクトリのアドレスをコマンドラインに書かなくても，どのディレクトリからも解析を行うことができます．

チュートリアル 1: 類人猿の分岐年代

このチュートリアルでは，類人猿 7 種から得られたミトコンドリア・タンパク質コーディング遺伝子のデータセットを解析します．このデータセットは Yang and Rannala [9] で解析されており，PAML [8] パッケージ (examples/DatingSoftBound) で配信されています．mtCDNApri123.txt ファイルは塩基配列のアライメントです．このアライメントは 1st, 2nd, 3rd コドン座位ごとに 3 パーティションに分けられています．mtCDNApri.trees ファイルには，化石制約の施された 7 種の系統樹が書かれています :

```
7 1
((((human, (chimpanzee, bonobo)) '>.06<.08', gorilla),
(orangutan, sumatran)) '>.12<.16', gibbon);
```

最初の行には種の数 (7) と系統樹の数 (1) が書かれています．その下に Newick 形式の系統樹が書かれています．系統樹に枝長を書いてはいけません．この系統樹は 2 つの化石制約が施されています．一つは human/ chimp の分岐：'>.06<.08'，もう一方は大型類人猿の MRCA (most recent common ancestor: 最も近縁な共通祖先)：'>.12<.16' にそれぞれ設定されています．時間の単位は 100 Myr (Million Years) です．このため

human/chimp の分岐には 6-8 Mya (Million Years Ago) の制約が設定されています．MCMCTree は SOFT BOUND を採用しています．このため，低い確立 (デフォルトでは 0.025) で，制約が守られない場合があるように設定されています．系統樹のルートに化石制約がないことに注意してください．MCMCTree はルートに制約を必要としていますが，もしない場合には，コントロールファイル内部で別の形 (RootAge 変数) で設定する必要があります．

コントロールファイル mcmctree.ctl には，MCMCTree プログラムを走らせるために必要な説明が書かれています．このファイルをテキストエディタで開いてみると，以下のようになっています :

```
          seed = -1
       seqfile = mtCDNApri123.txt
      treefile = mtCDNApri.trees
       outfile = out
         ndata = 3
       usedata = 1    * 0: no data; 1:seq; 2:approximation; 3:out.BV (in.BV)
         clock = 2    * 1: global clock; 2: independent; and 3: correlated rates
       RootAge = '<1.0'  * safe constraint on root age, used if no fossil for root.
         model = 0    * 0:JC69, 1:K80, 2:F81, 3:F84, 4:HKY85
         alpha = 0    * alpha for gamma rates at sites
         ncatG = 5    * No. categories in discrete gamma
     cleandata = 0    * remove sites with ambiguity data (1:yes, 0:no)?
       BDparas = 1 1 0   * birth, death, sampling
  kappa_gamma = 6 2     * gamma prior for kappa
  alpha_gamma = 1 1     * gamma prior for alpha
  rgene_gamma = 2 2     * gamma prior for rate for genes
sigma2_gamma = 1 10    * gamma prior for sigma^2     (for clock=2 or 3)
      finetune = 1: .1 .1 .1 .1 .1 .1 * auto (0 or 1) : times, rates, etc.
         print = 1
        burnin = 2000
      sampfreq = 2
       nsample = 20000
```

seed: ランダムシードを設定します．-1 に設定すれば，コンピュータの時間から選ばれるので，毎回異なった数字になり得られる結果にも本来の狙い通りずれが生じます．全く同じ解析を行いたい場合は，奇数か偶数を設定します．

seqfile and treefile: 配列アライメントと tree ファイルの名前です．

outfile: 解析の結果が書かれます．

ndata: アライメントファイルに設定したパーティションの数です．例題では 3 パーティション (複数のタンパク質遺伝子から得られた) になっています．

usedata: 1 と設定すると，通常通り尤度関数が算出され MCMC 解析が行われます．usedata=0 の場合は，尤度は計算されず，プライアだけ算出されます．usedata=2 と =3 を用いると，近似尤度計算 (approximate likelihood calculation) と枝長の最尤推定が行われます．後で詳しく説明します．

clock: 分子時計の一定性を仮定したモデルが使われます．ここでは独立速度モデル (independent rates model) (clock=2) を使います．このモデルでは，進化速度は対数正規分布 (速度の対数関数をとると，正規分布になります) に従うと仮定されます．

RootAge: Tree ファイルでルートに制約を設定していない場合は，ここで用いる制約がルートに適用されます．ここでは '< 1.0' と設定しています．これは，この系統樹の最初の分岐 (gibbon/ human) が少なくとも 100 Mya には生じているという制約を意味します．

model, alpha and ncatG: 置換モデルです．この例題では，JC69 を用いています．JC69 はパラメータが少ないため計算が速いのが特徴です．alpha=0 としているので，この例題では速度変異を補正するガンマモデルを用いていません．PAML パッケージの一部として配信されている mcmctree.ctl ファイルでは，model=4, alpha=0.5 (HKY+G5) となっているかもしれません．この場合は JC69 モデルに変更して下さい．

BDparas: 出生死滅過程 (birth-death process) をコントロールするパラメータです．出生死滅過程は，コンピュータが解析を始める際に，まず制約なしの系統樹から時間プライアをつくるのに用いられます．ここではデフォルトの 1 1 0 を用いています．この設定だと，均一な分岐年代を割り振ります．

kappa_gamma and alpha_gamma: 置換モデルパラメータ κ (トランジション / トランスバージョン比) とサイト間の速度変異を補正するガンマ分布の形状パラメータ α です．

rgene_gamma: 平均置換速度のガンマプライアです．ガンマ分布の平均と分散は，それぞれ α/β，α/β^2 と表されます．α は分布の形状を決めます．α =1 あるいは =2 だと，かなり拡散したプライアとなります．平均速度が見て理解できるように，α を 2 つの数字のどちらかに固定すると良いでしょう．この例題では α =2, β =2 としています．これは平均速度が 100 Myr あたり 1 置換であることを示しています．統計解析に R (www.r-project.org) を使っている方は，以下のコマンドでガンマ分布を見ることができます：

```
> curve(dgamma(x, shape=2, rate=2), from=0, to=10)
```

sigma2_gamma: 速度変異パラメータのガンマプライア (i.e. 速度対数の分散 σ^2) です．大きな数を σ^2 に設定することは，速度の変異が大きいと仮定することになります．σ^2 のプライアは推定年代の事後確率に大きな影響を及ぼします [4]．アライメントが短い場合は，とくに影響が大きいです．

finetune: MCMC 解析で行われるステップサイズを設定します．Version 4.4e からは自動的に finetune を選ぶ機能がついたので，それ以前よりもこの数値が問題にならなくなりました．設定については後ほど説明します．

print: 1 の場合は，MCMC 解析のアウトプットと結果の要約がハードディスクに保存されます (MCMC 解析のサンプリングは mcmc.out ファイルに保存されます．アウトプットファイルの要約は上記を参照)．0 の場合は，スクリーンアウトだけでファイルには保存されません．

`burnin, sampfreq and nsample`: 例題では最初の 2000 回分の反復計算 (iteration) は捨てられます．これをバーンイン (burnin) と呼びます．その後，2 反復ごとにサンプルされ，20,000 回サンプリングが行われます．合計すると MCMC 解析は 2000 + 2×20,000 = 42,000 回行われます．ある程度信頼性の高い結果を得るためには，通常 10,000 から 20,000 サンプル必要です．よりサンプルサイズが大きくなると (例えば 100,000 サンプル)，ハードディスクを無駄に消費するわりに，統計学的に有意な改善がほとんど見られない傾向があります．そして，プログラムが結果を要約するのに長い時間を要します．もし (結果の収束を改善させるために) MCMC 解析の回数を多くする必要があるなら，sampfreq の回数を増やし，nsample は適度な数にとどめた方が良いです．

これでプログラムを走らせる準備ができました．結果を見てみましょう．ターミナルウィンドウを開き (Windows では，`Start > All programs > Accessories > Command prompt`)，チュートリアルの各種ファイルが保存されたディレクトリに移動してください．run01 という名前のディレクトリを作成し，tree ファイル，アライメントファイル，コントロールファイルを run01 ディレクトリにコピーして下さい．私の Windows コンピュータでは，チュートリアルの各種ファイルは `C:\Users\Mario\Tutorial\run01>` に保存しました．この新しいディレクトリ内部に入り，以下のコマンドを入力してください :

```
C:\Users\Mario\Tutorial\run01> mcmctree mcmctree.ctl
```

MCMC プログラムがスタートします．プログラムはアライメントファイル，tree ファイル，コントロールファイルを読み込みます．その後，safety check を行います．MCMC プログラムが走り出したら，以下のようなスクリーンアウトが出ます :

```
lnL0 = -40215.47
Starting MCMC (np = 48) . . .
finetune steps (time rate mixing para RatePara ...):  0.1000  0.1000  0.1000  0.1000  0.1000
  paras: 6 times, 3 mu, 3 sigma2 (& rates, kappa, alpha)
 -4% 0.16 0.64 0.33 0.00 0.69  0.178 0.149 0.091 0.067 0.032 - 1.438 -34999.3
Current Pjump:      0.16033  0.64117  0.32800  0.00000  0.69267
Current finetune:   0.10000  0.10000  0.10000  0.10000  0.10000
New     finetune:   0.05050  0.31051  0.11113  0.00100  0.37446
 -2% 0.38 0.19 0.40 0.00 0.37  0.160 0.147 0.092 0.066 0.029 - 1.313 -34989.9
Current Pjump:      0.38033  0.18628  0.39600  0.00000  0.36633
Current finetune:   0.05050  0.31051  0.11113  0.00100  0.37446
New     finetune:   0.06743  0.18359  0.15638  0.00001  0.47671
 -1% 0.27 0.37 0.27 0.00 0.30  0.154 0.147 0.092 0.066 0.029 - 1.280 -34995.0
Current Pjump:      0.27133  0.36956  0.27400  0.00000  0.30433
Current finetune:   0.06743  0.18359  0.15638  0.00001  0.47671
New     finetune:   0.06009  0.23632  0.14090  0.00000  0.48476
  0% 0.29 0.27 0.32 0.00 0.30  0.155 0.145 0.094 0.067 0.029 - 1.332 -34996.7  0:02
Current Pjump:      0.29367  0.26978  0.32200  0.00000  0.29833
Current finetune:   0.06009  0.23632  0.14090  0.00000  0.48476
New     finetune:   0.05862  0.20922  0.15316  0.00000  0.48163
  5% 0.32 0.31 0.25 0.00 0.31  0.158 0.147 0.092 0.066 0.029 - 1.280 -34997.5  0:04
 10% 0.31 0.32 0.27 0.00 0.30  0.158 0.146 0.093 0.067 0.029 - 1.303 -35000.8  0:06
 15% 0.32 0.32 0.28 0.00 0.30  0.157 0.146 0.093 0.067 0.029 - 1.308 -34994.6  0:08
 20% 0.32 0.32 0.28 0.00 0.31  0.157 0.146 0.093 0.067 0.029 - 1.311 -34993.5  0:10
```

最初に得られる尤度は -40,215.47 です．例題で用いた 7 種の系統樹では分岐は 7–1 = 6 個，枝は 7×2 – 2 =12 本あります．1 回の繰り返し計算で推定されるパタメータを数えてみましょう．3 パーティションの解析を行うため，3 つの分岐年代，3 つの平均置換速度, 3 つの速度浮動パラメータを推定します．12×3=36 個の枝速度も推定されます．これで合計 48 パラメータを推定しています．最初の行を見てみましょう：

```
-4% 0.16 0.64 0.33 0.00 0.69  0.178 0.149 0.091 0.067 0.032 - 1.438 -34999.3
```

マイナスで示されたパーセント (-4%) は MCMC 解析がまだバーイン解析を行っていることを示します．次に示されている 5 つの数字は提案採択率 (acceptance proportions) を示します．順に，時間，速度，mixing, 置換モデルパラメータ，速度パラメータをプリントしています．例えば，MCMC 解析が行われたうち，提案時間 (proposal time) の 16% が採用され (84% は却下されている)，提案速度 (proposed rates) は 64% が採用されています．

適切な MCMC 解析は提案採択率がおよそ 30% である必要があります (20–40% が理想的だが，15–70% でも良いです)．解析が進むに連れて，プログラムは 30% 付近になるまで finetune を改善していることがわかります：

```
0% 0.29 0.27 0.32 0.00 0.30 0.155 0.145 0.094 0.067 0.029 - 1.332 -34996.7 0:02
```

JC69 モデルはパラメータがないので, 置換モデルパラメータの提案採択率はゼロです．この場合は提案されるパラメータはないので，何も採択されないだけです．解析としては問題ありません．次の 5 つのパラメータは 5 つのノードで得られた分岐年代の平均です．最初の数字 (0.155) は，ルートの年代です．この段階で，MCMC 解析はルートノードの平均分岐年代を 15.5 Mya と推定したことになります．ダッシュのあとにある数字はそれぞれ, 枝の速度, 尤度 (-34996.7), 要した時間 (2 秒) を示しています．残りのスクリーンアウトは以下のようになっています：

```
 25% 0.32 0.32 0.29 0.00 0.31  0.157 0.146 0.093 0.067 0.029 - 1.313 -34998.0  0:11
 30% 0.31 0.32 0.30 0.00 0.30  0.156 0.145 0.093 0.067 0.029 - 1.318 -34992.6  0:13
 35% 0.32 0.32 0.30 0.00 0.30  0.156 0.146 0.093 0.067 0.029 - 1.316 -34989.3  0:15
 40% 0.32 0.32 0.29 0.00 0.30  0.156 0.146 0.093 0.067 0.029 - 1.318 -35004.4  0:17
 45% 0.32 0.33 0.29 0.00 0.30  0.156 0.146 0.093 0.067 0.029 - 1.315 -34993.0  0:19
 50% 0.32 0.33 0.29 0.00 0.30  0.156 0.146 0.093 0.067 0.029 - 1.315 -34996.8  0:21
 55% 0.32 0.33 0.29 0.00 0.30  0.156 0.145 0.093 0.067 0.029 - 1.316 -34998.6  0:23
 60% 0.32 0.33 0.29 0.00 0.30  0.156 0.145 0.093 0.067 0.029 - 1.317 -34992.6  0:25
 65% 0.32 0.33 0.29 0.00 0.30  0.156 0.145 0.093 0.067 0.029 - 1.318 -34992.3  0:27
 70% 0.32 0.33 0.29 0.00 0.30  0.156 0.145 0.093 0.067 0.029 - 1.318 -34997.5  0:28
 75% 0.32 0.33 0.29 0.00 0.30  0.156 0.145 0.093 0.067 0.029 - 1.316 -34994.6  0:30
 80% 0.32 0.33 0.29 0.00 0.30  0.156 0.145 0.093 0.067 0.029 - 1.317 -34994.3  0:32
 85% 0.32 0.33 0.29 0.00 0.30  0.156 0.145 0.093 0.067 0.029 - 1.318 -34998.6  0:34
 90% 0.32 0.33 0.29 0.00 0.30  0.156 0.145 0.093 0.067 0.029 - 1.317 -34997.2  0:36
 95% 0.32 0.33 0.29 0.00 0.30  0.156 0.145 0.093 0.067 0.029 - 1.317 -34993.1  0:38
100% 0.32 0.33 0.29 0.00 0.30  0.156 0.145 0.093 0.067 0.029 - 1.318 -34998.9  0:40
```

それぞれのコラムにある値 (提案の採択率，時間，速度) をチェックした方が良いです．これらは MCMC 解析を通じて安定している必要があります．もし提案採択率の値が大きく変動する場合は，バーンインの長さが十分でない可能性があります．この場合はコントロールファイルにあるバーンインの値をより大きくして，解析をやり直して下さい．ルートノードの年代が大きく変動する場合は，コントロールファイルのバーンインか sampfreq の値を大きくしてください．同様に他の推定年代，速度，尤度もチェックして，

必要な場合は MCMC 解析の長さを変更してください.

MCMC 解析が収束したかどうかを判断するのは，簡単ではありません．例えば時間や速度などの提案採択率が安定したように見えたからと言って，MCMC 解析が収束していない場合もあります．収束を確認するには，複数の解析を行って結果を比較するしかありません．run02 という新しいディレクトルを作成して，必要なインファイルをコピーしてください．2 つの解析で得られた結果を比較して下さい．この例題では，同じような結果が得られているはずです．しかし，seed number が違うためにまったく同じではありません.

MCMC 解析が終わったら (100% に達したら)，プログラムはサンプリングされた値を結果としてまとめてスクリーンアウトします．プログラムは他のアウトプットファイル (out, SeedUsed, mcmc.out, FigTree.tre) も作ります．アウトファイルは結果の要約が書かれています．ファイルをエディタ (Notepad や Text Edit など) で開いてください. ファイルの最初の方には, 通常は必要でない数字が並んでいます．スクロールダウンして, 6 つの系統樹を見つけてください：

```
Species tree for TreeView.  Branch lengths = posterior mean times; 95% CIs = label
((((1_human, (2_chimpanzee, 3_bonobo) 12 ) 11 , 4_gorilla) 10 , (5_orangutan, 6_su ...
((((human: 0.067071, (chimpanzee: 0.028581, bonobo: 0.028581): 0.038490): 0.025788 ...
((((human: 0.067071, (chimpanzee: 0.028581, bonobo: 0.028581) 0.023-0.035: 0.03849 ...
rategram locus 1:
((((human: 0.389222, (chimpanzee: 0.427481, bonobo: 0.394052): 0.391420): 0.442241 ...
rategram locus 2:
((((human: 0.158087, (chimpanzee: 0.132350, bonobo: 0.145101): 0.129604): 0.164731 ...
rategram locus 3:
((((human: 1.969288, (chimpanzee: 1.926311, bonobo: 1.653482): 2.067122): 1.553286 ...
```

最初の系統樹には，各分岐に付けられたラベルが書かれているだけです．2 番目の tree には，設定した time unit の範囲で推定された枝長が書かれています．3 つめの tree には枝長と推定年代の信頼区間が書かれています．最後に書かれている 3 つの tree は，それぞれのパーティションごとに (ここでは 3 パーティション)，枝長の代わりに置換速度が書かれています．Tree の後には，48 パラメータについて得られた平均と 95% 信頼区間が書かれています．例えば：

```
t_n8 0.1560 (0.1315, 0.1805) (Jeffnode 12)
```

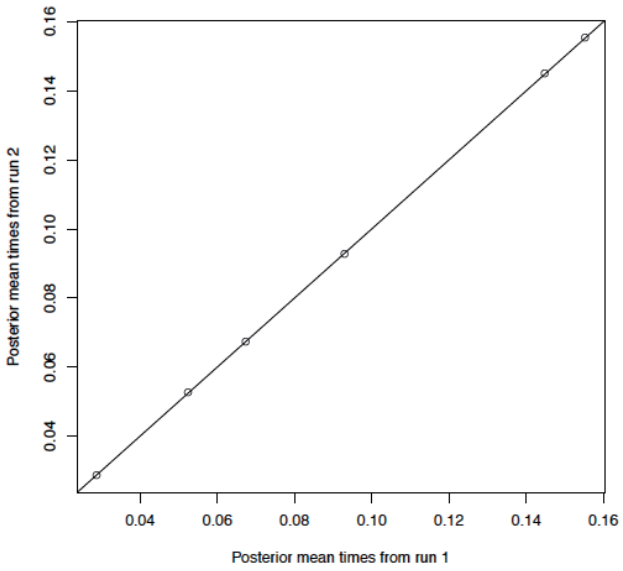
は node 8 の推定年代です．jeffnode は Jeff Thorne 博士が作成した MULTIDIVTIME [6] で解析した場合に得られる node 番号です．比較のために書かれています.

複数 (ここでは 2 つ) の解析結果が収束しているかどうかを見るには，アウトファイルに出力された推定年代をコピーして，エクセルなどで解析する必要があります．なお Unix を用いると，特定の文字を含んだ行だけを簡単に抽出できます：

```
[DatingSoftBound]$ grep t_n out
t_n8        0.1565 (0.1316, 0.1801)  (Jeffnode 12)
t_n9        0.1455 (0.1237, 0.1605)  (Jeffnode 11)
t_n10       0.0929 (0.0807, 0.1063)  (Jeffnode 10)
t_n11       0.0671 (0.0598, 0.0782)  (Jeffnode  9)
t_n12       0.0287 (0.0230, 0.0352)  (Jeffnode  8)
t_n13       0.0527 (0.0425, 0.0637)  (Jeffnode  7)
```

Excel かまたは R を用いて，推定された年代を比較します．プロットは y=x 線の周辺に分布する必要があります．うまく収束していない場合は，MCMC 解析の回数 (nsamp, バーンインなど) を増やす必要があります．この比較は，MCMC 解析で最も重要なので，毎回行ってください．以下はうまく収束した例です：

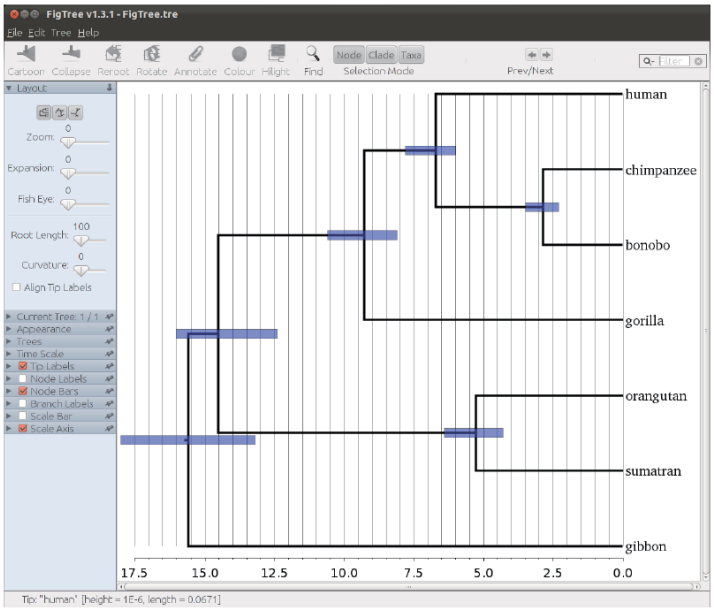
```
> # This is R code
> # Posterior mean times previously uploaded into R
> plot(t1$t, t2$t, xlab="Posterior mean times from run 1",
+ ylab="Posterior mean times from run 2")
> abline(0, 1) # this adds the x = y line.
```



例題では，50 列，20,002 行です．最初の列はサンプリングの回数を示します．その右側に続く 48 列は解析によって得られた 48 個のパラメータです. 最後の列は尤度です. mcmc.out ファイルは Tracer (http://beast.bio.ed.ac.uk/Tracer) で解析できるように作成されています.

SeedUsed ファイルには，MCMC 解析を開始するのに用いられた乱数が書かれています. この数値 (例: 949119895) を seed 変数としてコントロールファイルに記載すれば，まったく同じ結果が得られます.

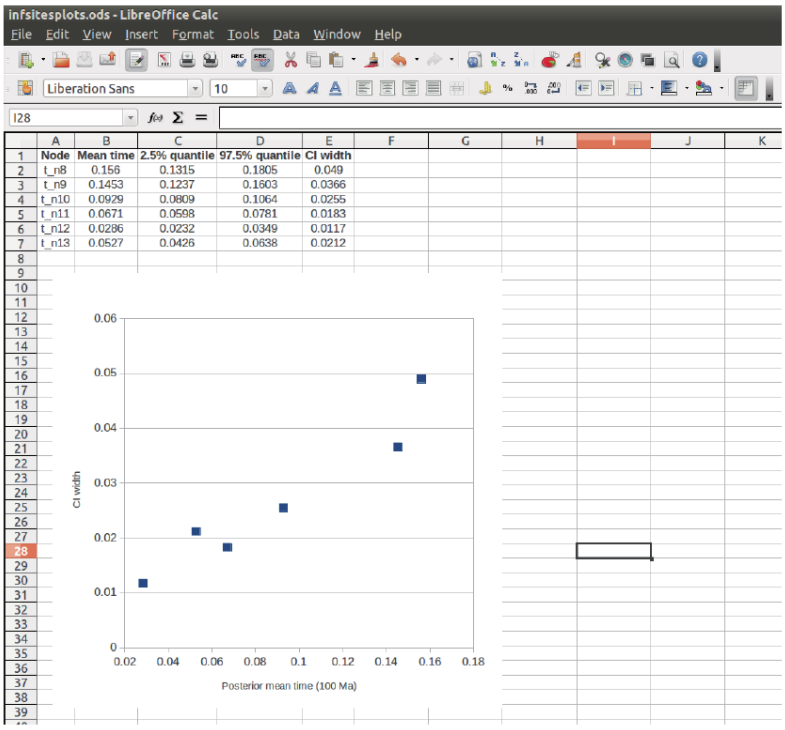
FigTree.tre ファイルには，解析によって得られた tree が Nexus 形式で保存されています．このファイルは FigTree プログラム (http://tree.bio.ed.ac.uk/software/figtree/) で画像にできます：



解析に用いる配列が長くなるほど (あるいは遺伝子座の数が多くなるほど)，事後年代 (解析によって得られた年代値) と信頼区間の幅は直線状になる傾向があります．この解析を行うことで，解析に用いる配列を増やした場合に解析精度が増すか検討することができます．得られた分岐年代をエディタで見てみましょう：

```
t_n8  0.1560 (0.1315, 0.1805) (Jeffnode 12)
t_n9  0.1453 (0.1237, 0.1603) (Jeffnode 11)
t_n10 0.0929 (0.0809, 0.1064) (Jeffnode 10)
t_n11 0.0671 (0.0598, 0.0781) (Jeffnode 9)
t_n12 0.0286 (0.0232, 0.0349) (Jeffnode 8)
t_n13 0.0527 (0.0426, 0.0638) (Jeffnode 7)
```

例えばルート (node 8) では，信頼区間は 0.1805 - 0.1315 = 0.049 です．Excel を用いて得られた結果 (上記) をプロットしてみます：



ここでは E 列 (CI width: 信頼区間) vs. B 列 (Mean time: 推定年代の平均値) のグラフを示しています．原点を通る近似曲線を引くことで，プロットが直線に乗っているかどうかより詳細に検討できます．このプロットは無限サイトプロット (infinite sites plot) として知られています [5,9]．ここで mcmctree.ctl ファイルを開いて，usedata 変数を変更しましょう：

```
seed = -1
seqfile = mtCDNApri123.txt
treefile = mtCDNApri.trees
outfile = out
ndata = 3
usedata = 0 * 0: no data; 1:seq; 2:approximation; 3:out.BV (in.BV)
```

解析をもう一度行います．この解析は配列データを用いない解析なので，事前分布を見ることができます．アウトファイルを開いて，上記で行ったように Excel を用いて事前年代と事後年代を比較してみましょう．事後年代は配列を解析に用いたため，両者は異なっているはずです：

```
t_n8  0.5849 (0.1604, 1.0043) (Jeffnode 12)
t_n9  0.1400 (0.1201, 0.1599) (Jeffnode 11)
t_n10 0.1052 (0.0684, 0.1481) (Jeffnode 10)
t_n11 0.0702 (0.0601, 0.0800) (Jeffnode 9)
t_n12 0.0353 (0.0018, 0.0721) (Jeffnode 8)
t_n13 0.0696 (0.0033, 0.1418) (Jeffnode 7)
```

化石制約を施した分岐では，事前年代と事後年代が大きく異なることがあります．これは，配列データを解析に入れると得られた年代が Soft bound を通り越してしまったことを意味しています．常に usedata=0 の解析を行って，事前年代と事後年代を比較した方が良いです．この値を使って，上に示した無限サイトプロットを作成してみましょう．usedata=1 のときと解析結果はどのように違うのか検討してください．

試しに model=4, alpha=0.5 (mcmctree.ctl ファイル) で解析してみてください．これは HKY+G モデルです．JC69 モデルで得られた結果と比較してみましょう．

チュートリアル 2: 近似尤度計算を用いた類人猿の分岐年代

大きなアライメントでは，MCMC 解析の間に行われる尤度計算はコンピュータに負担がかかるので, 年代推定が非常に遅くなります．Thorne ら [6] は尤度を概算する方法によって MCMC 解析の速度が劇的に速くなることを示唆しています．詳細は dos Reis and Yang [2] でも説明しています．MCMCtree v4.5 から Thorne らの方法が導入され，現在ではアークサインに基づく近似方法がデフォルトになっています [2]．dos Reis et al. [1] では，この近似方法によって 2 千万サイトのデータを解析しています．

この近似方法による分岐年代の推定は，以下 2 ステップからなります．最初のステップでは，最尤推定値での勾配とヘッセ行列 (i.e. 一次導関数のベクトルと二次導関数の行列) とともに，枝長が最尤推定されます．勾配とヘッセには，尤度表面の曲率に関する情

報が含まれています．第二のステップでは，MCMC 解析を用いて分岐年代が推定されます．その過程では，勾配とヘッセを用いて尤度関数をテイラー展開することで，近似計算を行っています [2].

このチュートリアルで解析を行ったディレクトリに移動してください．新しいディレクトリを作成し Hessian と言う名前にしてください．tree，配列アライメント，コントロールファイルをこちらにコピーしてください．コントロールファイル (mcmctree.ctl) をエディタで開いて，usedata を 3 にしてください：

```
usedata = 3 * 0: no data; 1:seq; 2:approximation; 3:out.BV (in.BV)
```

MCMCTree を走らせます：

```
C:\Users\Mario\Tutorial\Hessian> mcmctree mcmctree.ctl
```

MCMCTree は，BASEML プログラムで使われる 3 つのファイル (tmp*.txt : 配列アライメントファイル ; tmp*.tree：tree ファイル ; tmp*.ctl：コントロールファイル) を自動的に作成します．3 つのファイルはパーティションごとに作成されます．MCMCTree は BASEML を自動的に呼び出し，枝長，勾配，ヘッセをパーティションごとに推定させます (BASEML に PATH を通しておく必要があります).

解析の結果得られた out.BV ファイルを見てみましょう．このファイルには，パーティションごとに推定された枝長，勾配，ヘッセが保存されています．ファイルの最初は以下のようになっています：

```
7
(((human: 0.025136, (chimpanzee: 0.013241, bonobo: 0.010461): 0.014365): 0.013406, ...
0.025520 0.013406 0.025136 0.014365 0.013241 0.010461 0.029460 0.041605 0.022252 ...
0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 ...
Hessian
-96520.86 -3713.45 -10369.47 -11757.41 -13729.28 -20652.92 -8997.32 -4832.46 ...
-3713.45 -176562.71 -4315.44 -143.38 -13722.26 -9111.27 -21294.71 -7017.87 ...
```

最初の行には，種数 (7) が書かれています．その下には , 枝長付きの無根樹，2x7-3 = 11 個の枝長，勾配 (通常すべて 0)，ヘッセ行列 (11x11 = 121) が書かれています．より下には，他のパーティションについて得られた同様の推定値が書かれています．BASEML は rst2 ファイルに tree，勾配，ヘッセを書き込みます．MCMCTree は rst2 ファイルの情報をあつめて，out.BV ファイルにまとめます．

上位のディレクトリ (C:\Users\Mario\Tutorial) に戻って，approx01 という名前のディレクトリを作成しましょう．このディレクトリに tree, 配列アライメント，out.BV ファイルをコピーしてください．out.BV ファイルを in.BV という名前に変更してください．mcmctree.ctl ファイルを開いて usedata 変数を変更します：

```
usedata = 2 * 0: no data; 1:seq; 2:approximation; 3:out.BV (in.BV)
C:\Users\Mario\Tutorial\approx01> mcmctree mcmctree.ctl
```

解析を始めます：

```
C:\Users\Mario\Tutorial\approx01> mcmctree mcmctree.ctl
```

この解析で MCMCTree は年代推定を行いますが，勾配とヘッセを用いて尤度を近似します．他の MCMC 解析同様，解析を何度か行って結果が収束するか確認してください．approx02 というディレクトリを作成して，MCMC 解析を行ってください．しかし最初のステップ (枝長，勾配，ヘッセの推定) は繰り返さなくてよいです．近似尤度計算を用いて得られた結果と，Exact 法 (usedata=1) の結果を比較してください．ほぼ同じはずです．アウトファイルには計算時間も書かれています．

今回のケースでは，分子時計を仮定したモデル (clock=1) で近似尤度計算による解析を行わないでください．正しい結果が得られない傾向があります [2].

チュートリアル 3: 時間スケールの変更

独立速度モデル (clock=2) では，速度 (r) は以下の対数正規分布に従います：

$$f(r \mid \mu, \sigma^2) = \frac{1}{r\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{1}{2\sigma^2}\left[\log(r/\mu) + \sigma^2/2\right]^2\right\}$$

with mean

$$E(r) = \mu$$

and variance

$$Var(r) = \left(e^{\sigma^2} - 1\right)\mu^2.$$

分布は μ と σ^2 だけで決まります．σ^2 パラメータは $\log(r)$ の分散です．

時間を t とします．時間スケールを変更すると，変更後の時間は $t' = kt$ となり，これに従って置換率のスケールも $r' = r/k$ のように変更する必要があります．定数 a は , $E(aX) = aE(X)$ および $Var(aX) = a^2Var(X)$ と変化することに注意しましょう．このため，時間スケール変更後の速度 r' の平均と分散は以下のように表すことができます：

mean

$$E(r') = E(r/k) = \frac{1}{k}E(r) = \frac{\mu}{k}$$

and variance

$$Var(r') = Var(r/k) = \frac{1}{k^2}Var(r) = \frac{1}{k^2}\left(e^{\sigma^2} - 1\right)\mu^2 = \left(e^{\sigma^2} - 1\right)\left(\frac{\mu}{k}\right)^2.$$

上記の式を見ると，r' はパラメータ σ^2 と $\mu' = \mu/k$ からなる対数正規分布であることがわかります．

時間スケールを変更した場合は，速度プライアも変更する必要があります．もし r がガンマ分布

$$f(r) = Gamma(r \mid \alpha, \beta),$$

に従う場合，r' はこれに相当するガンマ分布

$$f(r') = Gamma(r' \mid \alpha, k\beta)$$

に従います．

時間スケールを変化させても σ^2 は影響を受けないので，変更する必要はありません．出生死滅過程で用いられているパラメータは，変更する必要があります．例えば，例題で用いている霊長類の系統樹で時間スケールを 100 Myr から 1 Myr (i.e. $t' = 100t$ および $r' = r/100$) に変更すると, tree ファイルの制約年代は以下のように変更する必要があります：

```
7 1
((((human, (chimpanzee, bonobo)) '>6<8', gorilla),
(orangutan, sumatran)) '>12<16', gibbon);
```

コントロールファイルにある RootAge，BDparas，rgene_gamma パラメータも，変更する必要があります：

```
          seed = -1
      seqfile = mtCDNApri123.txt
     treefile = mtCDNApri.trees
      outfile = out
        ndata = 3
      usedata = 1    * 0: no data; 1:seq; 2:approximation; 3:out.BV (in.BV)
        clock = 2    * 1: global clock; 2: independent; and 3: correlated rates
      RootAge = '<100.0'  * safe constraint on root age, used if no fossil for root.
        model = 0    * 0:JC69, 1:K80, 2:F81, 3:F84, 4:HKY85
        alpha = 0    * alpha for gamma rates at sites
        ncatG = 5    * No. categories in discrete gamma
    cleandata = 0    * remove sites with ambiguity data (1:yes, 0:no)?
      BDparas = .01  .01 0   * birth, death, sampling
  kappa_gamma = 6 2      * gamma prior for kappa
  alpha_gamma = 1 1      * gamma prior for alpha
  rgene_gamma = 2 200     * gamma prior for rate for genes
sigma2_gamma = 1 10    * gamma prior for sigma^2     (for clock=2 or 3)
     finetune = 1: .1 .1 .1 .1 .1 .1 * auto (0 or 1) : times, rates, etc.
        print = 1
       burnin = 2000
     sampfreq = 2
      nsample = 20000
```

finetune を自動的に推定する設定を行わない場合は，finetune パラメータを変更する必要が出てきます．時間スケールを変化させた解析でも，まったく同じ結果が得られます．この場合，事後年代は k 倍，事後速度は $1/k$ 倍の値が得られます．

自己相関速度モデル (clock=3: correlated rates model) では，速度はパラメータ μ と $t\sigma^2$ で表される対数正規分布に従います．$\log(r)$ の分散が時間 t の関数となることに注意してください．時間スケールを変更すると，この変数は $t'\sigma^2/k$ と変化するので，形状パラメータを修正する必要があります．例えば，100 Myr の時間スケールで相関速度モデルを用いている場合，コントロールファイルで clock を 3 にしてください：

```
        seed = -1
    seqfile = mtCDNApri123.txt
   treefile = mtCDNApri.trees
    outfile = out
      ndata = 3
    usedata = 1    * 0: no data; 1:seq; 2:approximation; 3:out.BV (in.BV)
      clock = 3    * 1: global clock; 2: independent; and 3: correlated rates
    RootAge = '<1.0'  * safe constraint on root age, used if no fossil for root.
      model = 0    * 0:JC69, 1:K80, 2:F81, 3:F84, 4:HKY85
      alpha = 0    * alpha for gamma rates at sites
      ncatG = 5    * No. categories in discrete gamma
  cleandata = 0    * remove sites with ambiguity data (1:yes, 0:no)?
    BDparas = 1 1 0   * birth, death, sampling
kappa_gamma = 6 2      * gamma prior for kappa
alpha_gamma = 1 1      * gamma prior for alpha
rgene_gamma = 2 2     * gamma prior for rate for genes
sigma2_gamma = 1 10    * gamma prior for sigma^2     (for clock=2 or 3)
   finetune = 1: .1 .1 .1 .1 .1 .1 * auto (0 or 1) : times, rates, etc.
      print = 1
     burnin = 2000
   sampfreq = 2
    nsample = 20000
```

1 Myr の時間スケールで解析する場合は，コントロールファイルの該当部分を変更してください：

```
        seed = -1
    seqfile = mtCDNApri123.txt
   treefile = mtCDNApri.trees
    outfile = out
      ndata = 3
    usedata = 1    * 0: no data; 1:seq; 2:approximation; 3:out.BV (in.BV)
      clock = 3    * 1: global clock; 2: independent; and 3: correlated rates
    RootAge = '<100.0'  * safe constraint on root age, used if no fossil for root.
      model = 0    * 0:JC69, 1:K80, 2:F81, 3:F84, 4:HKY85
      alpha = 0    * alpha for gamma rates at sites
      ncatG = 5    * No. categories in discrete gamma
  cleandata = 0    * remove sites with ambiguity data (1:yes, 0:no)?
    BDparas = .01 .01 0   * birth, death, sampling
kappa_gamma = 6 2      * gamma prior for kappa
alpha_gamma = 1 1      * gamma prior for alpha
rgene_gamma = 2 200     * gamma prior for rate for genes
sigma2_gamma = 1 1000    * gamma prior for sigma^2     (for clock=2 or 3)
   finetune = 1: .1 .1 .1 .1 .1 .1 * auto (0 or 1) : times, rates, etc.
      print = 1
     burnin = 2000
   sampfreq = 2
    nsample = 20000
```

チュートリアル 4: アミノ酸配列データを用いた近似尤度計算

アミノ酸配列の解析で近似尤度計算を行う場合は，他にも作業が必要です．例題として，examples ディレクトリにある abglobin.aa ファイルを使います．このファイルには哺乳類 5 種から得られたグロビンのアミノ酸配列が保存されています．mcmctree-globin という名前のディレクトリを作成し，ここに abglobin.aa ファイルをコピーしてください．エディタで abglobin.tree ファイルを作成し，以下を保存してください：

```
5 1
((((rabbit, rat), human), goat-cow), marsupial)'B(1.7,1.9)';
```

ここでは 100 Myr を時間スケールとして用います．marsupial/human の分岐を 170-190 Mya と設定します．mcmctree.ctl ファイル (チュートリアル 2 と同じです) をコピーして mcmctree-globin ディレクトリに保存し，これをエディタで開いて以下のように該当部分を編集してください：

```
      seed = -1
   seqfile = abglobin.aa
  treefile = abglobin.tree
   outfile = out
     ndata = 1
   seqtype = 2    * 0: nucleotides; 1:codons; 2:AAs
   usedata = 3    * 0: no data; 1:seq like; 2:normal approximation; 3:out.BV (in.BV)
     clock = 2    * 1: global clock; 2: independent rates; 3: correlated rates
   RootAge = '<1.0'  * safe constraint on root age, used if no fossil for root.
     model = 0    * 0:JC69, 1:K80, 2:F81, 3:F84, 4:HKY85
     alpha = 0    * alpha for gamma rates at sites
     ncatG = 5    * No. categories in discrete gamma
 cleandata = 0    * remove sites with ambiguity data (1:yes, 0:no)?
   BDparas = 1 1 0.1   * birth, death, sampling
kappa_gamma = 6 2      * gamma prior for kappa
alpha_gamma = 1 1      * gamma prior for alpha
rgene_gamma = 2 2     * gamma prior for rate for genes
sigma2_gamma = 1 10    * gamma prior for sigma^2     (for clock=2 or 3)
   finetune = 1: .1 .1 .1 .1 .1 .1 * auto (0 or 1): times, rates, mixing, paras, RateParas, FossilErr
      print = 1
     burnin = 2000
   sampfreq = 2
    nsample = 20000
```

以下のコマンドを入力してください：

```
C:\Users\Mario\Tutorial\mcmctree-globin> mcmctree mcmctree.ctl
```

MCMCTree は tmp1.ctl，tmp1.tree，tmp1.txt ファイルを作り，ヘッセ行列を作成するために CODEML を自動的に起動します．しかし，MCMCTree は最も単純なアミノ酸置換モデルを用いるので，実際のデータ解析向きではありません．得られた out.BV と rst ファイルを削除してください．dat ディレクトリから wag.dat ファイルをコピーし，mcmctreeglobin ディレクトリに保存してください．tmp1.ctl をエディタで開いて以下のように編集してください：

```
seqfile = tmp1.txt
treefile = tmp1.trees
outfile = tmp1.out
noisy = 3
seqtype = 2
model = 2 * 2: Empirical
aaRatefile = wag.dat
fix_alpha = 0
alpha = .5
ncatG = 4
Small_Diff = 0.1e-6
getSE = 2
method = 1
```

編集を行ったコントロールファイルは，WAG+G モデルで解析を行います．CODEML を以下のコマンドで走らせて，WAG+G モデルでヘッセ行列を推定してください：

```
C:\Users\Mario\Tutorial\mcmctree-globin> codeml tmp1.ctl
```

rst2 ファイルを in.BV と言う名前に変更してください．mcmctree.ctl ファイルを以下のように変更して, MCMCTree を近似尤度計算方法で走らせてください．やり方はステップ 2 と同じです：

```
usedata = 2
```

コドンモデルでも同様の解析が可能です．さらには，RNA 遺伝子やアミノ酸配列の解析を BASEML と CODEML で別々に走らせて，rst ファイルに保存された値を in.BV ファイルにまとめて解析を行うことも可能です．[詳しくはこちら http://abacus.gene.ucl.ac.uk/software/MCMCtreeStepByStepManual.JPN.pdf を参照してください．]

チュートリアル 5: 長大な配列データを用いた分岐年代の MCMC 推定

チュートリアル 5 はやや難しいです．ここでは，読者がベイズ系統学と分岐年代推定の理論を知っていて，PAML やコードコンパイラなどに詳しいと仮定しています．無限サイト理論については文献 3, 5, 9 を読んでください．プログラム Infinitesites は，配列が無限に長いと仮定して年代を推定します. Windows ユーザは PAML パッケージの bin ディレクトリからプログラムを見つけてください. Unix 系システム (Mac, Linux など) のユーザはプログラムをコンパイルする必要があります．[コンパイルのヒントは，mcmctree.c の上の方に書いてあります．]

(1) 分子進化速度の一定性を仮定した解析，(2) 仮定しない解析，2 種類の解析が可能です [5].

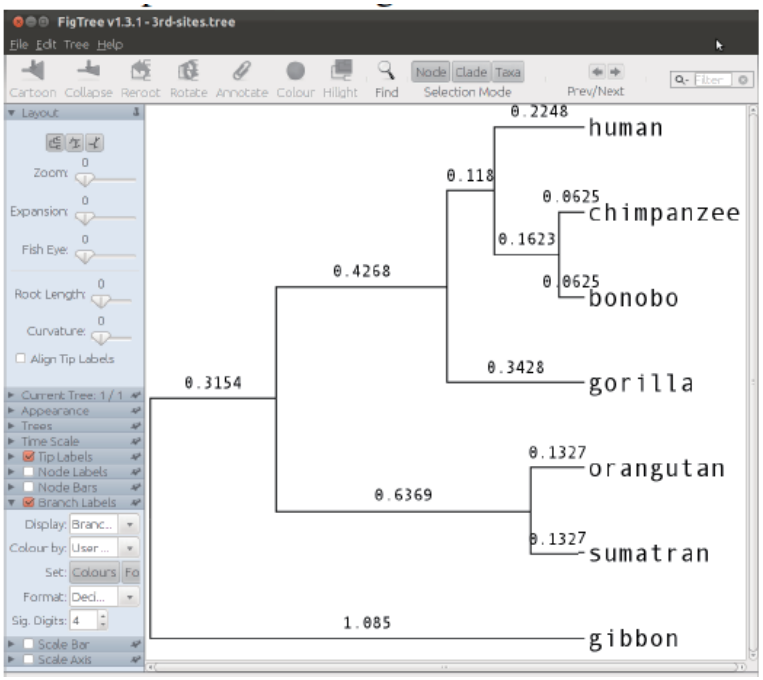
(1) 分子進化速度の一定性を仮定した解析: 有根樹の枝長 (端から分岐点までの枝長 [いわゆる node height のことです]) すべての距離を記載したファイルを作成する必要があります．距離は通常，BASEML か CODEML を用いて，分子進化速度の一定性を仮定した最尤法によって種の系統樹に沿って算出します．ここでは，チュートリアル 1 で用いた霊長類の系統樹について，BASEML を用いて枝長を計算しました．分子進化速度の一定性を仮定して，HKY+G5 モデルを用いて 3rd コドン座位を解析します．枝長付きの系統樹は mlb ファイルに保存されています：

```
((((human: 0.224767, (chimpanzee: 0.062487, bonobo: 0.062487): 0.162279):
0.118038, gorilla: 0.342804): 0.426831, (orangutan: 0.132715, sumatran:
0.132715): 0.636921): 0.315408, gibbon: 1.085044);
```

Node 番号付きの系統樹は以下のようになっています：

```
((((1_human, (2_chimpanzee, 3_bonobo) 12 ) 11 , 4_gorilla) 10 , (5_
orangutan, 6_sumatran) 13 ) 9 , 7_gibbon) 8 ;
```

FigTree で系統樹を描きます：



Inf という名前のディレクトリを作成し，以下のような距離を書いて FixedDsClock1.txt というファイルに保存します：

```
7
1.085  0.7696  0.3428  0.2248  0.0625  0.1327
```

s 種からなる系統樹 (分子進化速度の一定性を仮定) では，s-1 個の距離があります．最初の行 (7) は系統樹に含まれる種の数を示します．最初の数字 1.085 はルート (node 8) から末端まで，それ以外は各分岐 (順に node 9 から 13) から末端までの距離，をそれぞれ示します．例えば，orang から node 9 までの距離は 0.1327+0.6369 = 0.7696 です．clock=1 という条件で BASEML か CODEML を用いる場合は，アウトファイルに (デフォルトでは mlb か mlc という名前です) 保存されます．対数尤度 (lnL) の下に，距離は番号順に並んでいます．チュートリアル 1 から tree と配列アライメント，コントロールファイルをコピーし，inf ディレクトリに保存します．配列アライメントファイル mtCDNApri123.txt から，1st と 2nd コドン在位からなる部分を削除します．その後コントロールファイルを編集してください：

```
seed = -1
seqfile = mtCDNApri123.txt
treefile = mtCDNApri.trees
outfile = out
ndata = 1
seqtype = 0 * 0: nucleotides; 1:codons; 2:AAs
usedata = 1 * 0: no data; 1:seq like; 2:normal approximation; 3:out.BV (in.
BV)
clock = 1 * 1: global clock; 2: independent rates; 3: correlated rates
```

プログラムを走らせます：

```
C:\Users\Mario\Tutorial\inf> Infinitesites
```

プログラムは, 距離が, 無限に長い配列アライメントから推定された完全な最尤推定値 (分

散が 0) である，と 仮定します．そして，速度と時間のプライアとともに最尤推定値を用いて root の事後年代 (t_8) を推定します．推定方法に関しては文献 9 の方程式を参照してください．ここで，mcmctree.ctl ファイル内部で置換モデルに使われているパラメータ (カッパやアルファなど) は，結果に影響しないことに注意してください．これらのモデルは BASML で枝長を推定するときに用いられるだけです．

解析が終了すると，プログラムは事後確率をスクリーンアウトします：

```
mean (95% CI) CI-width for times
Node  8: 0.230088 ( 0.221717, 0.246060) 0.024343
Node  9: 0.163204 ( 0.157266, 0.174532) 0.017267
Node 10: 0.072695 ( 0.070050, 0.077741) 0.007691
Node 11: 0.047672 ( 0.045937, 0.050981) 0.005044
Node 12: 0.013254 ( 0.012772, 0.014174) 0.001402
Node 13: 0.028141 ( 0.027117, 0.030094) 0.002977
mean & 95% CI for rates
gene 1: 4.715580 ( 4.409499, 4.893628)
```

ここで事後分布は一次元であることに注意しましょう．ルートの分岐年代の分布がわかれば，他の分岐年代の分布もわかります．時間と距離は比例関係にあります：$t_9/t_8 = d_9/d_8$. 平均推定年代と信頼区間をプロットすると，プロットは直線状に並びます．

上記の解析は，1 遺伝子座を仮定しています．1 つ以上の遺伝子座を用いる場合は，枝長 (分岐年代) は遺伝子座間で比例関係にある必要があります．FixedDsClock1.txt ファイルには，ルートの分岐年代 (距離) を含んだ各遺伝子座ごとの行が追加されている必要があります．

(2) 分子進化速度の一定性を仮定しない解析：この解析はより複雑です．解析に用いる配列が無限に長くても，推定年代の分布が一次元になることはありません．無限サイト解析では，有限数の遺伝子座を用いて分岐年代を推定しますが，それぞれの遺伝子座の配列は無限に長いと仮定します．解析プログラム infinitesites を走らせるには，遺伝子座ごとに tree のリスト (最尤推定された枝長付き) が必要です．理論的には，tree は無根で，分子時計の一定性を仮定しないで枝長が推定されている必要があります．
しかし，今のところプログラム Infinitesites は tree は有根と仮定し，枝長は分子時計の一定性を仮定しないで推定します．このためプログラムはルート付近の枝長を合計して用います．

ここでは BASEML を用いて枝長を計算しました．チュートリアル 1 の霊長類データから 3 つのコドン座位を別々のパーティションとして解析します．系統樹は有根とし，分子時計の一定性を仮定しません．HKY+G5 モデルを使います．制約付き tree ファイルを BASEML のインファイルとして使います．inf-loci という名前のディレクトリを作成してください．そこに制約付きの tree，mcmctre.ctl ファイル，配列アライメントファイルを保存してください．FixedDsClock23.txt というファイルに BASEML の解析で得られた ML tree を保存してください：

```
7
((((human: 0.029043, (chimpanzee: 0.014557, bonobo: 0.010908): 0.016729):
0.015344,
gorilla: 0.033888): 0.033816, (orangutan: 0.026872, sumatran: 0.022437):
0.069648):
0.073309, gibbon: 0.024637);
((((human: 0.012463, (chimpanzee: 0.002782, bonobo: 0.003835): 0.003331):
0.004490,
gorilla: 0.014278): 0.006308, (orangutan: 0.010818, sumatran: 0.008845):
0.030551):
0.004363, gibbon: 0.029246);
((((human: 0.270862, (chimpanzee: 0.066698, bonobo: 0.056883): 0.124104):
0.139082,
gorilla: 0.310797): 0.391342, (orangutan: 0.152555, sumatran: 0.114176):
0.696518):
0.017607, gibbon: 1.394718);
```

次のコマンドでプログラムを走らせてください：

```
C:\Users\Mario\Tutorial\inf-loci> Infinitesites
```

アウトプットは以下のようになっています：

```
Posterior mean (95% Equal-tail CI) (95% HPD CI) HPD-CI-width
t_n8      0.1942 (0.1610, 0.2364) (0.1602, 0.2306) 0.0704 (Jeffnode 12)
t_n9      0.1557 (0.1438, 0.1623) (0.1456, 0.1633) 0.0178 (Jeffnode 11)
t_n10     0.0921 (0.0827, 0.1039) (0.0820, 0.1031) 0.0210 (Jeffnode 10)
t_n11     0.0622 (0.0588, 0.0695) (0.0582, 0.0679) 0.0097 (Jeffnode 9)
t_n12     0.0242 (0.0185, 0.0302) (0.0185, 0.0302) 0.0117 (Jeffnode 8)
t_n13     0.0453 (0.0348, 0.0563) (0.0347, 0.0561) 0.0214 (Jeffnode 7)
r_left_L1 0.4834 (0.2852, 0.7186) (0.2584, 0.6878) 0.4294
r_left_L2 0.1727 (0.0795, 0.3591) (0.0604, 0.3111) 0.2507
r_left_L3 4.3565 (1.7766, 8.8623) (1.5023, 8.1197) 6.6173
mu_L1     0.5126 (0.4376, 0.6053) (0.4334, 0.5976) 0.1642
mu_L2     0.1736 (0.1366, 0.2302) (0.1338, 0.2213) 0.0875
mu_L3     3.9529 (3.1518, 4.8803) (3.1001, 4.7768) 1.6767
sigma2_L1 0.0542 (0.0167, 0.1446) (0.0107, 0.1194) 0.1087
sigma2_L2 0.1460 (0.0670, 0.3218) (0.0512, 0.2736) 0.2224
sigma2_L3 0.1476 (0.0604, 0.3035) (0.0552, 0.2760) 0.2208
```

この場合 3 パーティションの解析を行っているため，分子時計の一定性を仮定した解析とは異なり，事後年代は一次元ではありません．しかし，用いる遺伝子座の数を極端に多くできるのであれば, 事後推定年代は一次元に近くなり, 平均値と信頼区間は直線状になってゆくでしょう．Infinitesites は配列アライメントファイルを読み込みますが，データ自体は無視されることに注意してください．このため，もし mcmctree.ctl ファイルの ndata を ndata=3 として解析した場合は，配列アライメントファイルに 3 つのパーティションが存在し，FixedDsClock23.txt ファイルには枝長付きの系統樹が 3 つあることになります．

コメントと質問はこちらまで：
mariodosreis@gmail.com
Dep. Genetics, Evolution and Environment, University College London, London, UK, WC1E 6BT.

References

[1] M. dos Reis, J. Inoue, M. Hasegawa, R. J. Asher, P. C. Donoghue, and Z. Yang. Phylogenomic datasets provide both precision and accuracy in estimating the timescale of placental mammal phylogeny. Proc Biol Sci, 279(1742):3491–500, 2012.
[2] M. dos Reis and Z. Yang. Approximate likelihood calculation on a phylogeny for Bayesian estimation of divergence times. Mol Biol Evol, 28(7):2161–72, 2011.
[3] M dos Reis and Z Yang. The unbearable uncertainty of Bayesian divergence time estimation. Journal of Systematics and Evolution, 51(1):30–43, 2013.
[4] J. Inoue, P. C. Donoghue, and Z. Yang. The impact of the representation of fossil calibrations on Bayesian estimation of species divergence times. Syst Biol, 59(1):74–89, 2010.
[5] B. Rannala and Z. Yang. Inferring speciation times under an episodic molecular clock. Syst Biol, 56(3):453–66, 2007.
[6] J. L. Thorne, H. Kishino, and I. S. Painter. Estimating the rate of evolution of the rate of molecular evolution. Mol Biol Evol, 15(12):1647–57, 1998.
[7] Z Yang. Computational Molecular Evolution. Oxford University Press, Oxford, 2006.
[8] Z. Yang. PAML 4: phylogenetic analysis by maximum likelihood. Mol Biol Evol, 24(8):1586–91, 2007.
[9] Z. Yang and B. Rannala. Bayesian estimation of species divergence times under a molecular clock using multiple fossil calibrations with soft bounds. Mol Biol Evol, 23(1):212–26, 2006.