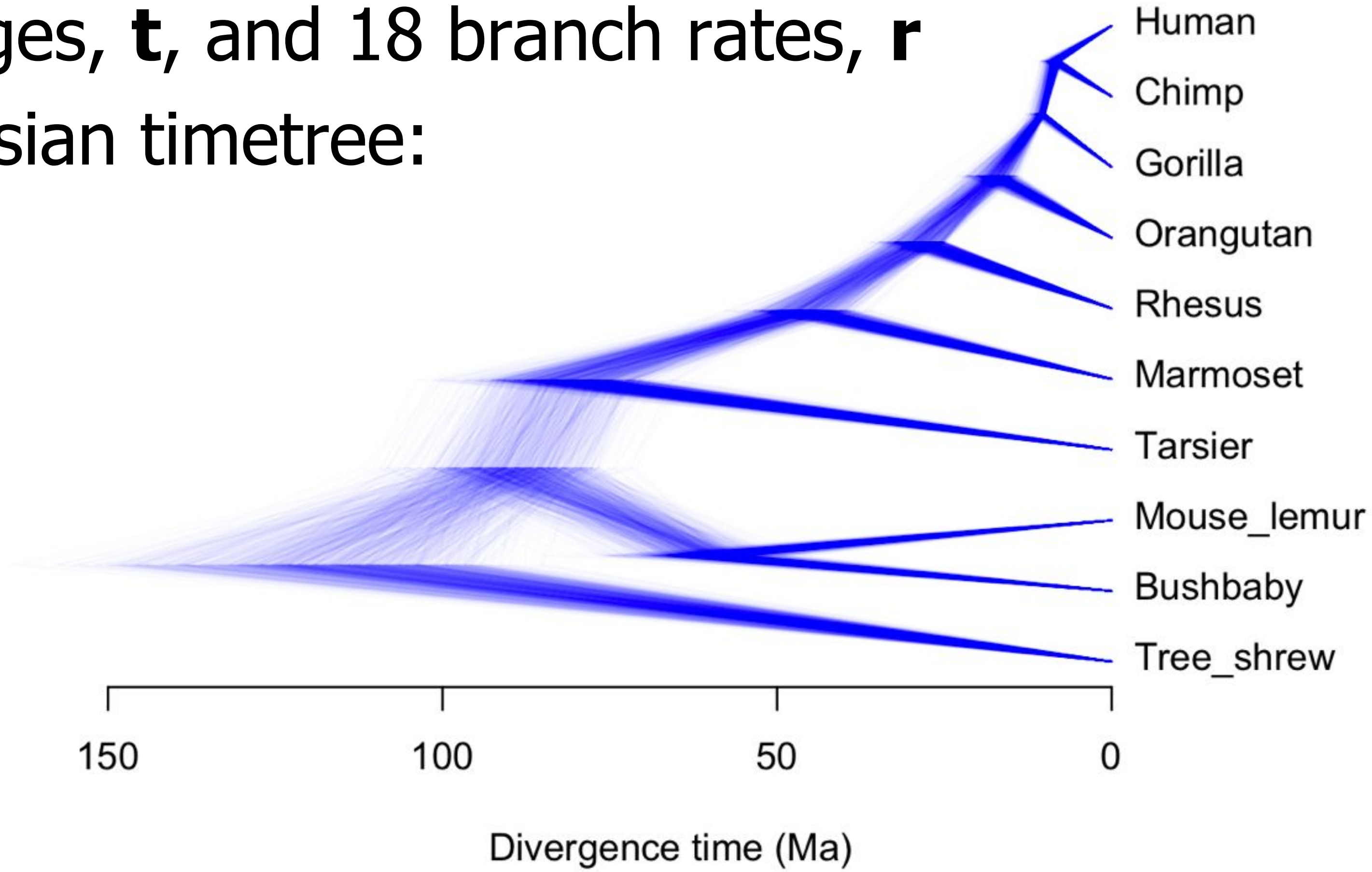
Bayesian molecular-clock dating using genome-scale datasets

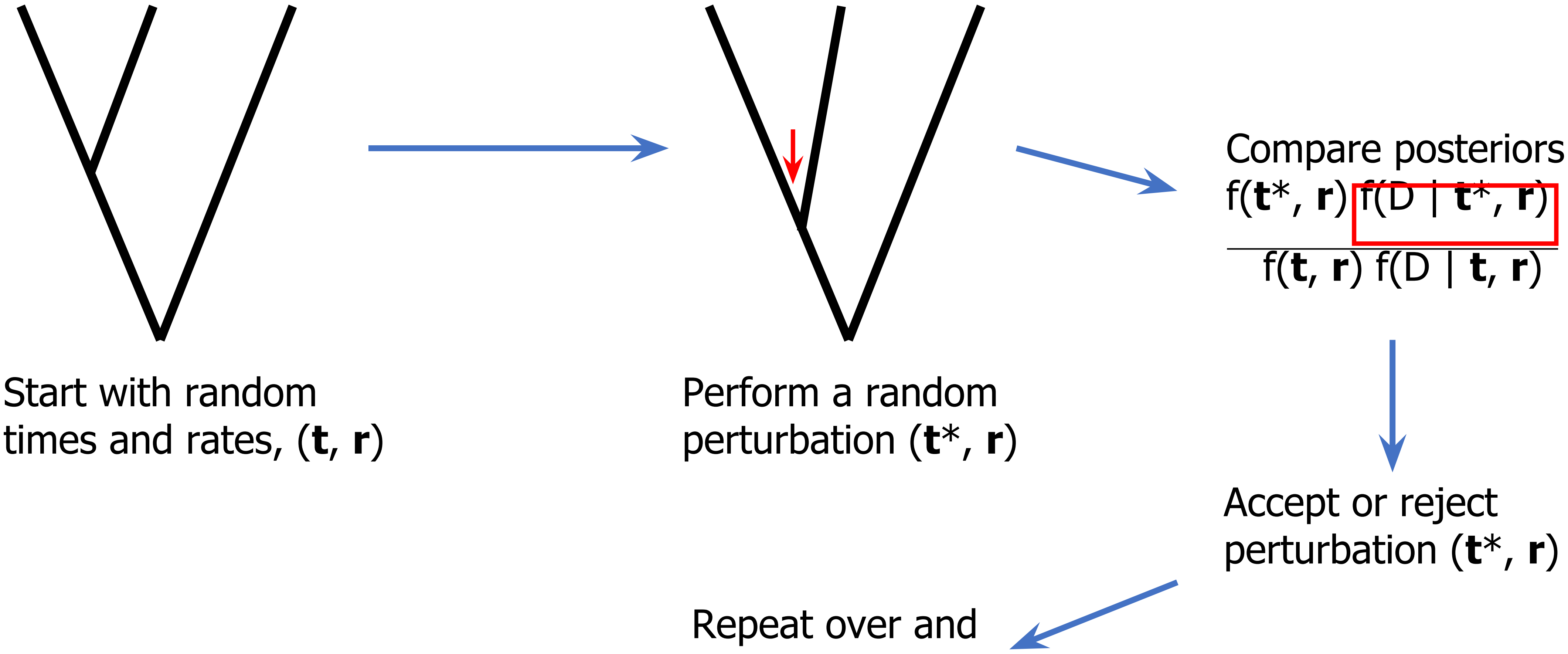
Mario dos Reis – Queen Mary University of London

Bayesian molecular-clock dating using genome-scale datasets

- Today we will use **MCMCtree** to infer the timetree of 10 primates using a genome-scale alignment (3.4 million base pairs)
- 9 node ages, **t**, and 18 branch rates, **r**
- Our Bayesian timetree:



MCMCtree uses the MCMC algorithm



Start with random times and rates, (**t**, **r**)

Perform a random perturbation (**t***, **r**)

Compare posteriors

$$\frac{f(\mathbf{t}^*, \mathbf{r})\, \boxed{f(D \mid \mathbf{t}^*, \mathbf{r})}}{f(\mathbf{t}, \mathbf{r})\, f(D \mid \mathbf{t}, \mathbf{r})}$$

Accept or reject perturbation (**t***, **r**)

Repeat over and over again

Likelihood calculation is expensive

- We must calculate the likelihood for every site in the alignment:
- $f\ (D \mid \mathbf{t}, \mathbf{r}) = \prod_i f\ (D_i \mid \mathbf{t}, \mathbf{r})$, where D_i is the i-th alignment site
- Thus, for genome-scale alignments this multiplication can become very large
- It gets worse if there are many species in the alignment because the algorithm must visit all nodes in the tree to calculate $f\ (D \mid \mathbf{t}, \mathbf{r})$
- MCMC sampling of using genome-alignments would take months or years

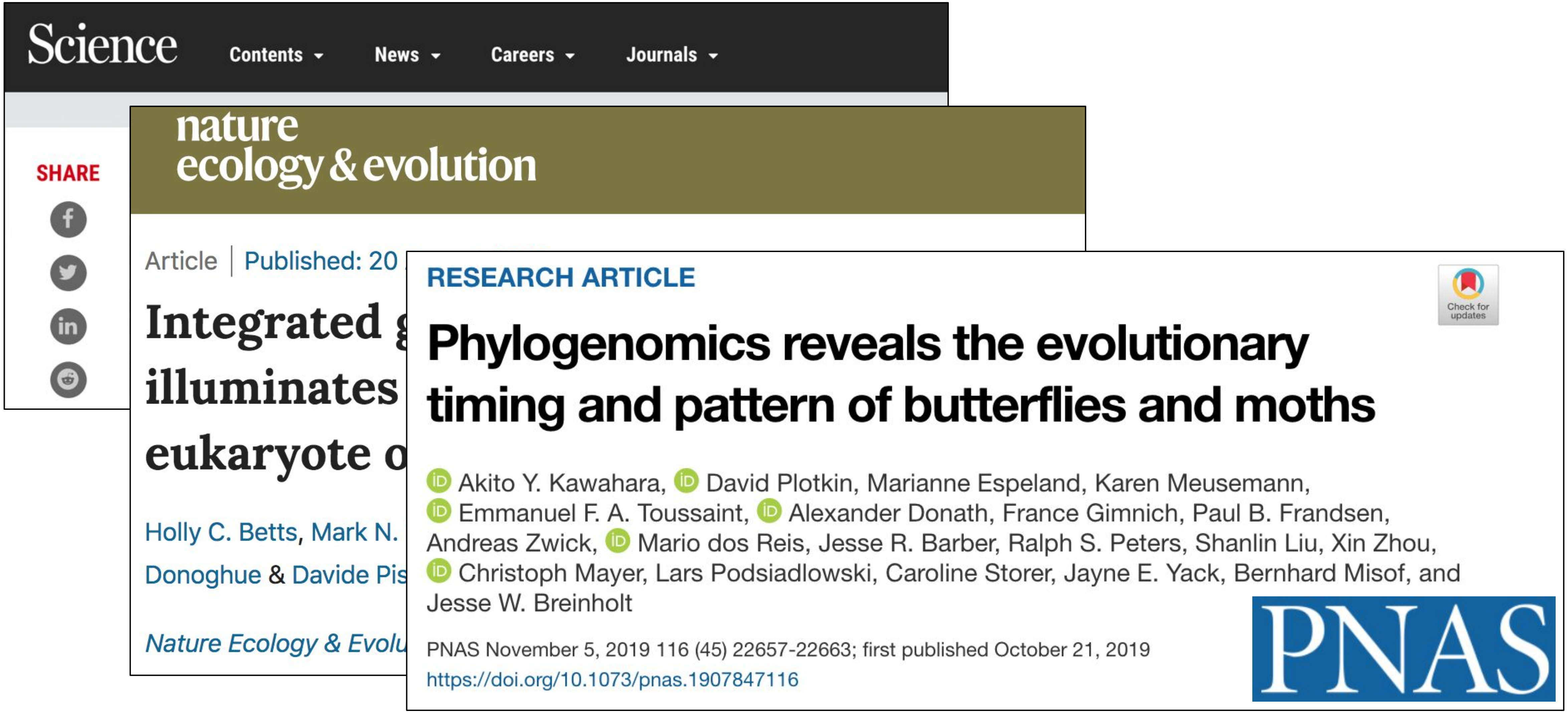
Bayesian molecular-clock dating using genome-scale datasets

- MCMCtree implements a method to approximate the likelihood during MCMC sampling of timetrees
- To use the approximation we need to:
  - Work on a fixed tree topology
  - Estimate the branch lengths, **b**, (in substitution per site) on the tree by ML
  - Estimate the gradient, **g**, and Hessian, **H**, of the likelihood
- Then, by using **b**, **g**, and **H**, MCMCtree can provide a surprisingly accurate approximation of the likelihood during MCMC sampling
- $\Delta \log f\,(D \mid \mathbf{b}) \approx \Delta\mathbf{b}^{T}\,\mathbf{g} + \Delta\mathbf{b}^{T}\,\mathbf{H}\,\Delta\mathbf{b} / 2$
- Cost of approximate calculations depends on the number of species (which determines size of **b**, **g**, **H**) but not on alignment length

Advantages of the approximation

- The approximation gives speedups of over 1,000x compared to MCMC sampling with exact likelihood calculations
- For example, in today’s example, our 3.4 million-base alignment can be sampled in about **2.5 min** with the approximation
- However, it would take ~**15,000 min** (10 days) with exact likelihood
- This is a **6,000x** computation speedup
- MCMCtree is now the de facto standard for timetree inference in large datasets

MCMCtree best for very large data



nature ecology & evolution

Article | Published: 20

Integrated g
illuminates
eukaryote o

Holly C. Betts, Mark N.
Donoghue & Davide Pis

Nature Ecology & Evolu

RESEARCH ARTICLE

Phylogenomics reveals the evolutionary timing and pattern of butterflies and moths

Akito Y. Kawahara, David Plotkin, Marianne Espeland, Karen Meusemann, Emmanuel F. A. Toussaint, Alexander Donath, France Gimnich, Paul B. Frandsen, Andreas Zwick, Mario dos Reis, Jesse R. Barber, Ralph S. Peters, Shanlin Liu, Xin Zhou, Christoph Mayer, Lars Podsiadlowski, Caroline Storer, Jayne E. Yack, Bernhard Misof, and Jesse W. Breinholt

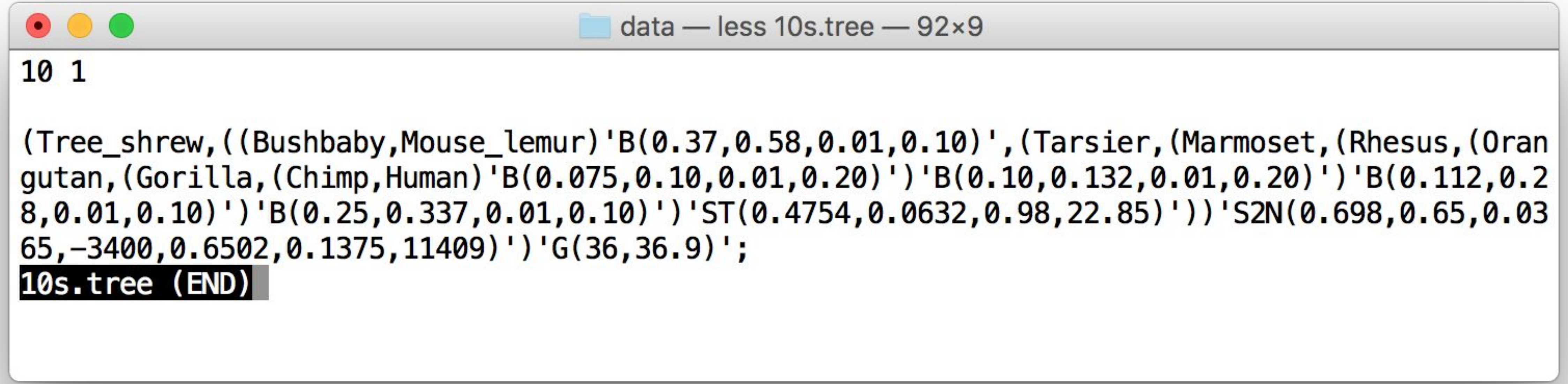
PNAS November 5, 2019 116 (45) 22657-22663; first published October 21, 2019
https://doi.org/10.1073/pnas.1907847116

Tutorial: Primates divergence times

- dos Reis and Yang (2019) **Bayesian Molecular Clock Dating Using Genome-Scale Datasets**. In: Evolutionary Genomics. https://doi.org/10.1007/978-1-4939-9074-0_10
- All data files are available from my GitHub:
- https://github.com/mariodosreis/divtime
- **MCMCtree** can be downloaded from:
- http://abacus.gene.ucl.ac.uk/software/paml4.9j.tgz
- It is recommended to have the following software: **R** and **RStudio**, **Tracer**, and **FigTree**.
- These slides: https://dosreislab.github.io/assets/slides/applnL.pdf

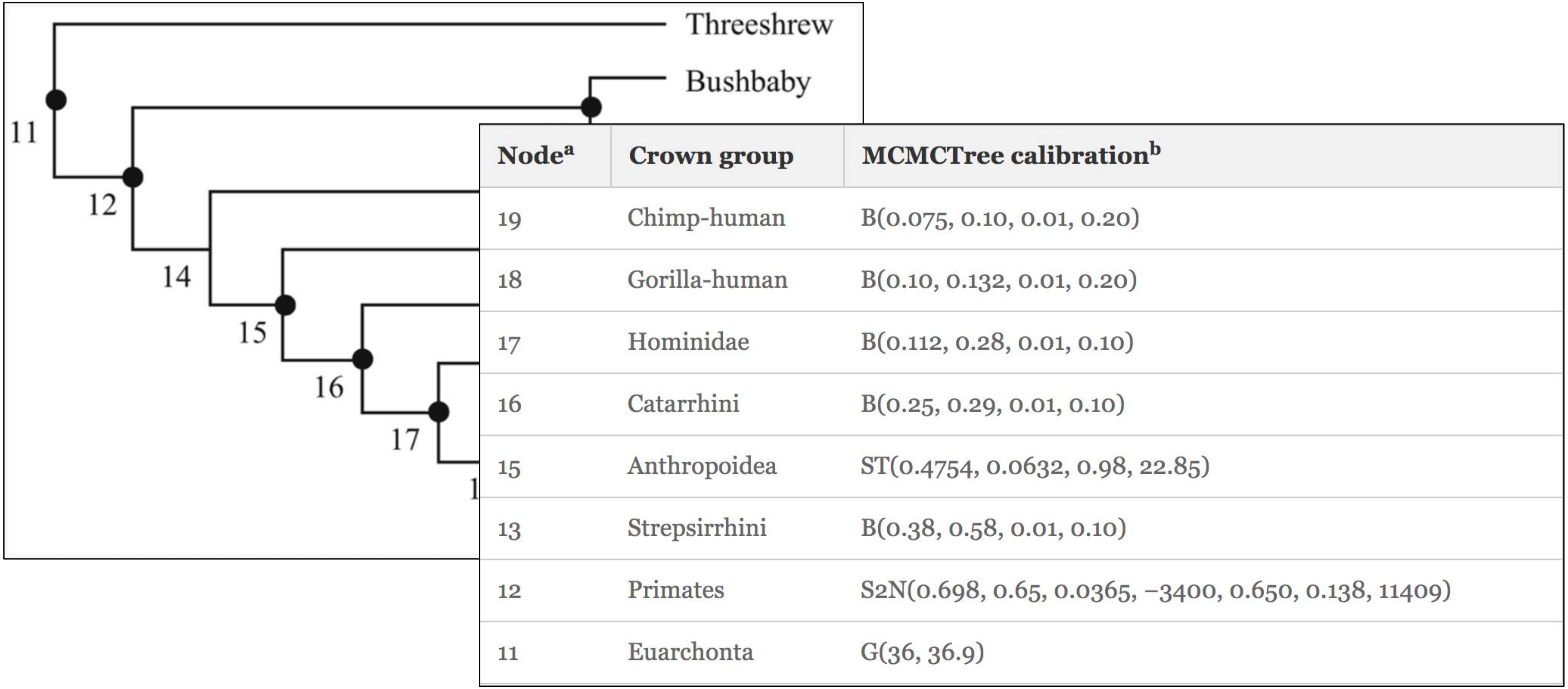
Data files: data/10s.tree

- Tree is in Newick format and has fossil calibrations
- You can open data/10s.tree in a text editor (Notepad, TextEdit, etc.)



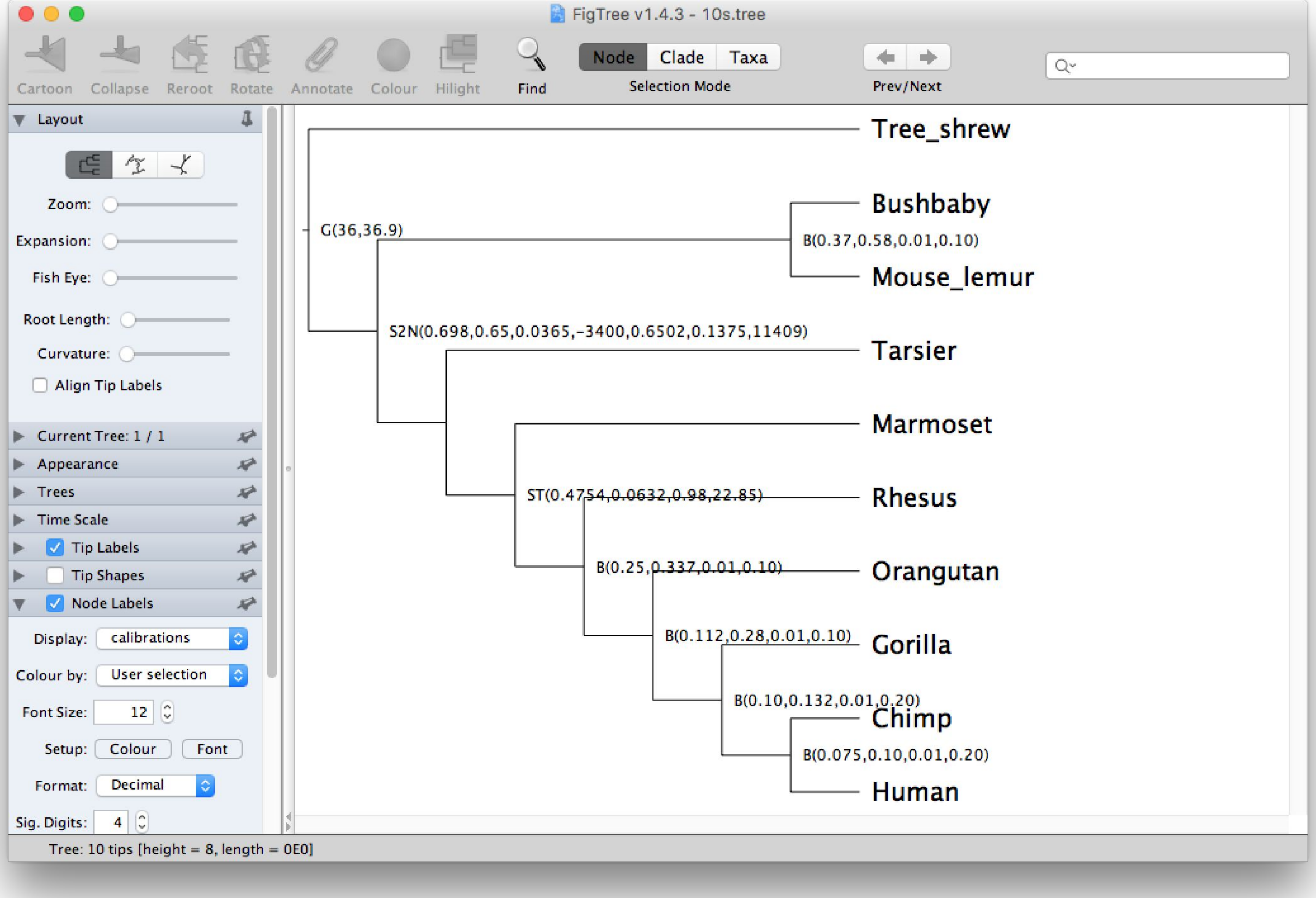
```
10 1

(Tree_shrew,((Bushbaby,Mouse_lemur)'B(0.37,0.58,0.01,0.10)',(Tarsier,(Marmoset,(Rhesus,(Oran
gutan,(Gorilla,(Chimp,Human)'B(0.075,0.10,0.01,0.20)')'B(0.10,0.132,0.01,0.20)')'B(0.112,0.2
8,0.01,0.10)')'B(0.25,0.337,0.01,0.10)')'ST(0.4754,0.0632,0.98,22.85)'))'S2N(0.698,0.65,0.03
65,-3400,0.6502,0.1375,11409)')'G(36,36.9)';
10s.tree (END)
```

Data files: data/10s.tree



Node[a]	Crown group	MCMCTree calibration[b]
19	Chimp-human	B(0.075, 0.10, 0.01, 0.20)
18	Gorilla-human	B(0.10, 0.132, 0.01, 0.20)
17	Hominidae	B(0.112, 0.28, 0.01, 0.10)
16	Catarrhini	B(0.25, 0.29, 0.01, 0.10)
15	Anthropoidea	ST(0.4754, 0.0632, 0.98, 22.85)
13	Strepsirrhini	B(0.38, 0.58, 0.01, 0.10)
12	Primates	S2N(0.698, 0.65, 0.0365, −3400, 0.650, 0.138, 11409)
11	Euarchonta	G(36, 36.9)

You may open data/10s.tree with FigTree



☐ Align Tip Labels

☑ Tip Labels

☐ Tip Shapes

☑ Node Labels

Display: calibrations

Colour by: User selection

Font Size: 12

Format: Decimal

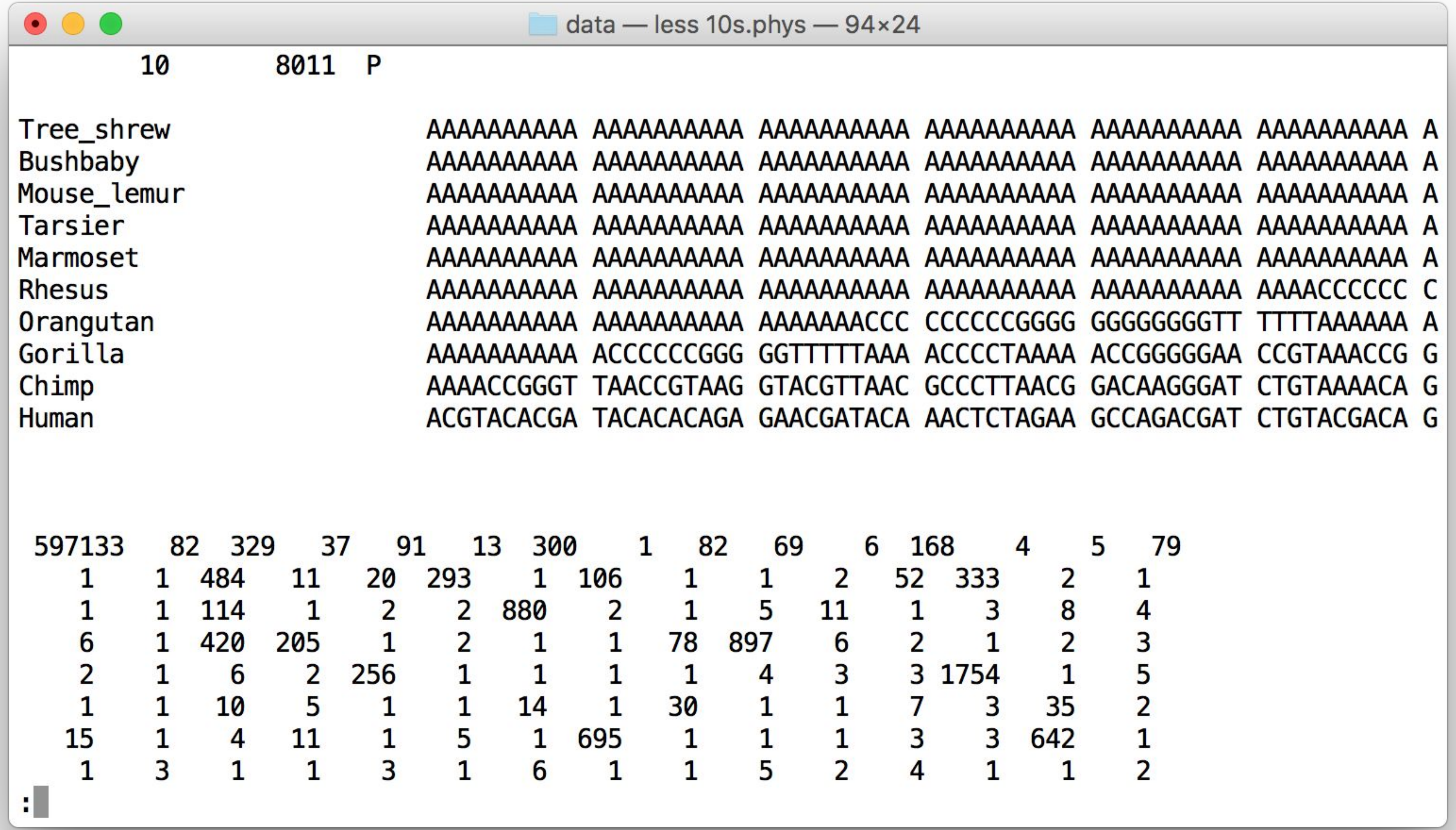
Sig. Digits: 4

Tree: 10 tips [height = 8, length = 0E0]

Data files: data/10s.phys

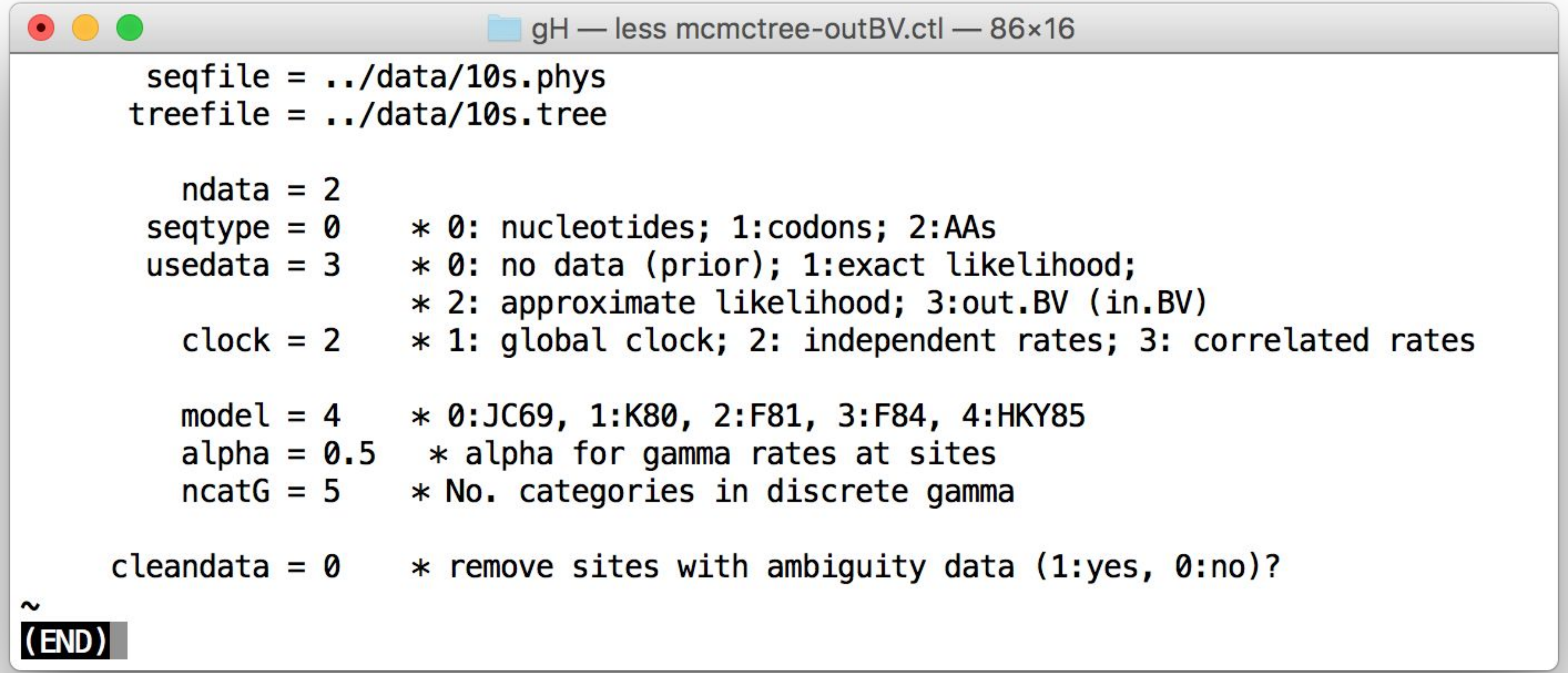
- Alignment is in Phylip sequential format with site-patterns
- Alignment is divided into two partitions:
  - First and second codon positions (8,011 site-patterns)
  - Third codon positions (14,507 site-patterns)
- You can also open the alignment using a text editor

Data files: data/10s.phys



```
        10       8011  P

Tree_shrew                 AAAAAAAAAA AAAAAAAAAA AAAAAAAAAA AAAAAAAAAA AAAAAAAAAA AAAAAAAAAA A
Bushbaby                   AAAAAAAAAA AAAAAAAAAA AAAAAAAAAA AAAAAAAAAA AAAAAAAAAA AAAAAAAAAA A
Mouse_lemur                AAAAAAAAAA AAAAAAAAAA AAAAAAAAAA AAAAAAAAAA AAAAAAAAAA AAAAAAAAAA A
Tarsier                    AAAAAAAAAA AAAAAAAAAA AAAAAAAAAA AAAAAAAAAA AAAAAAAAAA AAAAAAAAAA A
Marmoset                   AAAAAAAAAA AAAAAAAAAA AAAAAAAAAA AAAAAAAAAA AAAAAAAAAA AAAAAAAAAA A
Rhesus                     AAAAAAAAAA AAAAAAAAAA AAAAAAAAAA AAAAAAAAAA AAAAAAAAAA AAAACCCCCC C
Orangutan                  AAAAAAAAAA AAAAAAAAAA AAAAAAACCC CCCCCCGGGG GGGGGGGGTT TTTTAAAAAA A
Gorilla                    AAAAAAAAAA ACCCCCCGGG GGTTTTTAAA ACCCCTAAAA ACCGGGGGAA CCGTAAACCG G
Chimp                      AAAACCGGGT TAACCGTAAG GTACGTTAAC GCCCTTAACG GACAAGGGAT CTGTAAAACA G
Human                      ACGTACACGA TACACACAGA GAACGATACA AACTCTAGAA GCCAGACGAT CTGTACGACA G



 597133   82  329   37   91   13  300    1   82   69    6  168    4    5   79
    1    1  484   11   20  293    1  106    1    1    2   52  333    2    1
    1    1  114    1    2    2  880    2    1    5   11    1    3    8    4
    6    1  420  205    1    2    1    1   78  897    6    2    1    2    3
    2    1    6    2  256    1    1    1    1    4    3    3 1754    1    5
    1    1   10    5    1    1   14    1   30    1    1    7    3   35    2
   15    1    4   11    1    5    1  695    1    1    1    3    3  642    1
    1    3    1    1    3    1    6    1    1    5    2    4    1    1    2
:
```

Step 1: Estimation of b, g and H control file: gH/mcmctree-outBV.ctl



```
       seqfile = ../data/10s.phys
      treefile = ../data/10s.tree

         ndata = 2
       seqtype = 0    * 0: nucleotides; 1:codons; 2:AAs
       usedata = 3    * 0: no data (prior); 1:exact likelihood;
                      * 2: approximate likelihood; 3:out.BV (in.BV)
         clock = 2    * 1: global clock; 2: independent rates; 3: correlated rates

         model = 4    * 0:JC69, 1:K80, 2:F81, 3:F84, 4:HKY85
         alpha = 0.5   * alpha for gamma rates at sites
         ncatG = 5    * No. categories in discrete gamma

     cleandata = 0    * remove sites with ambiguity data (1:yes, 0:no)?
~
(END)
```

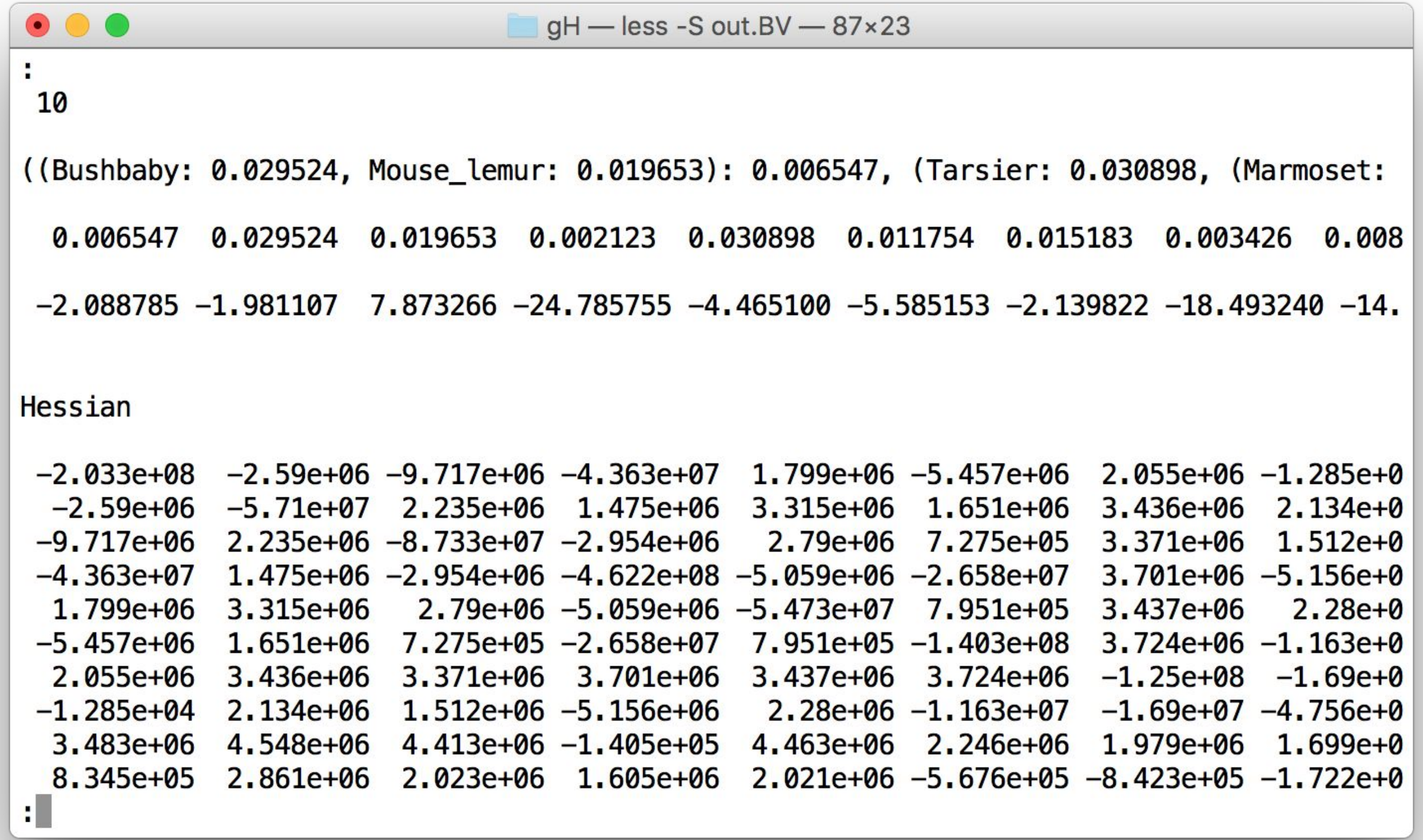
Step 1: Estimation of b, g and H

- In a terminal window, go to the gH/ directory and type

```
$ mcmctree mcmctree-outBV.ctl
```

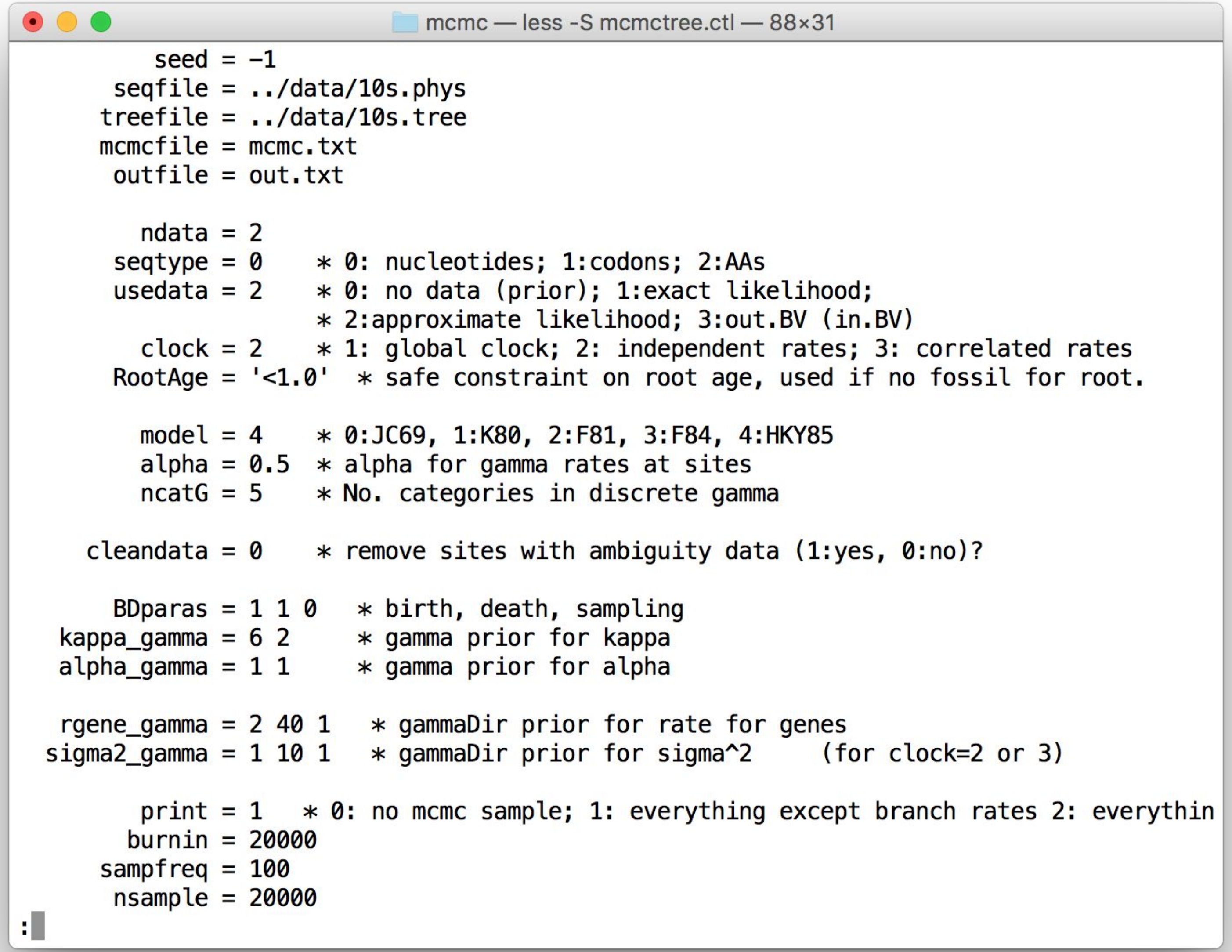
- MCMCtree will prepare and unrooted tree, a baseml.ctl file, and then call BASEML to calculate **b**, **g**, and **H**
- These parameters will be saved to a file called gH/out.BV
- Copy gH/out.BV as mcmc/in.BV

Step 1: file gH/out.BV



```
:
 10

((Bushbaby: 0.029524, Mouse_lemur: 0.019653): 0.006547, (Tarsier: 0.030898, (Marmoset:

  0.006547  0.029524  0.019653  0.002123  0.030898  0.011754  0.015183  0.003426  0.008

 -2.088785 -1.981107  7.873266 -24.785755 -4.465100 -5.585153 -2.139822 -18.493240 -14.


Hessian

 -2.033e+08  -2.59e+06 -9.717e+06 -4.363e+07  1.799e+06 -5.457e+06  2.055e+06 -1.285e+0
  -2.59e+06  -5.71e+07  2.235e+06  1.475e+06  3.315e+06  1.651e+06  3.436e+06  2.134e+0
 -9.717e+06  2.235e+06 -8.733e+07 -2.954e+06   2.79e+06  7.275e+05  3.371e+06  1.512e+0
 -4.363e+07  1.475e+06 -2.954e+06 -4.622e+08 -5.059e+06 -2.658e+07  3.701e+06 -5.156e+0
  1.799e+06  3.315e+06   2.79e+06 -5.059e+06 -5.473e+07  7.951e+05  3.437e+06   2.28e+0
 -5.457e+06  1.651e+06  7.275e+05 -2.658e+07  7.951e+05 -1.403e+08  3.724e+06 -1.163e+0
  2.055e+06  3.436e+06  3.371e+06  3.701e+06  3.437e+06  3.724e+06  -1.25e+08  -1.69e+0
 -1.285e+04  2.134e+06  1.512e+06 -5.156e+06   2.28e+06 -1.163e+07  -1.69e+07 -4.756e+0
  3.483e+06  4.548e+06  4.413e+06 -1.405e+05  4.463e+06  2.246e+06  1.979e+06  1.699e+0
  8.345e+05  2.861e+06  2.023e+06  1.605e+06  2.021e+06 -5.676e+05 -8.423e+05 -1.722e+0
:
```

Step 2: Posterior MCMC sampling control file: mcmc/mcmctree.ctl



```
          seed = -1
       seqfile = ../data/10s.phys
      treefile = ../data/10s.tree
      mcmcfile = mcmc.txt
       outfile = out.txt

         ndata = 2
       seqtype = 0    * 0: nucleotides; 1:codons; 2:AAs
       usedata = 2    * 0: no data (prior); 1:exact likelihood;
                      * 2:approximate likelihood; 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 = 4    * 0:JC69, 1:K80, 2:F81, 3:F84, 4:HKY85
         alpha = 0.5  * 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 40 1   * gammaDir prior for rate for genes
  sigma2_gamma = 1 10 1   * gammaDir prior for sigma^2     (for clock=2 or 3)

         print = 1   * 0: no mcmc sample; 1: everything except branch rates 2: everythin
        burnin = 20000
      sampfreq = 100
       nsample = 20000
:
```

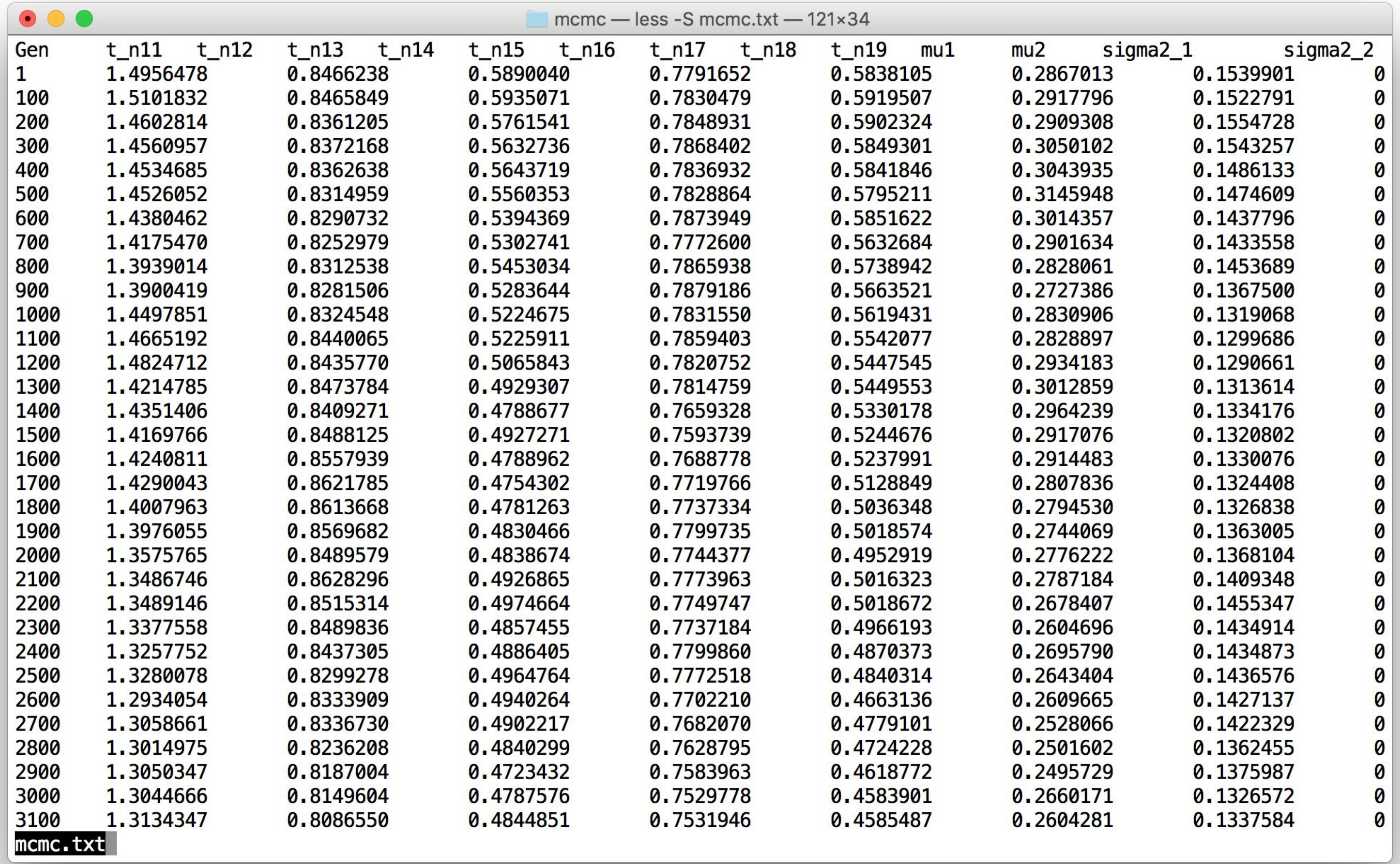
Step 2: Posterior MCMC sampling

- In a terminal window, go to the mcmc/ directory and type

```
$ mcmctree mcmctree.ctl
```

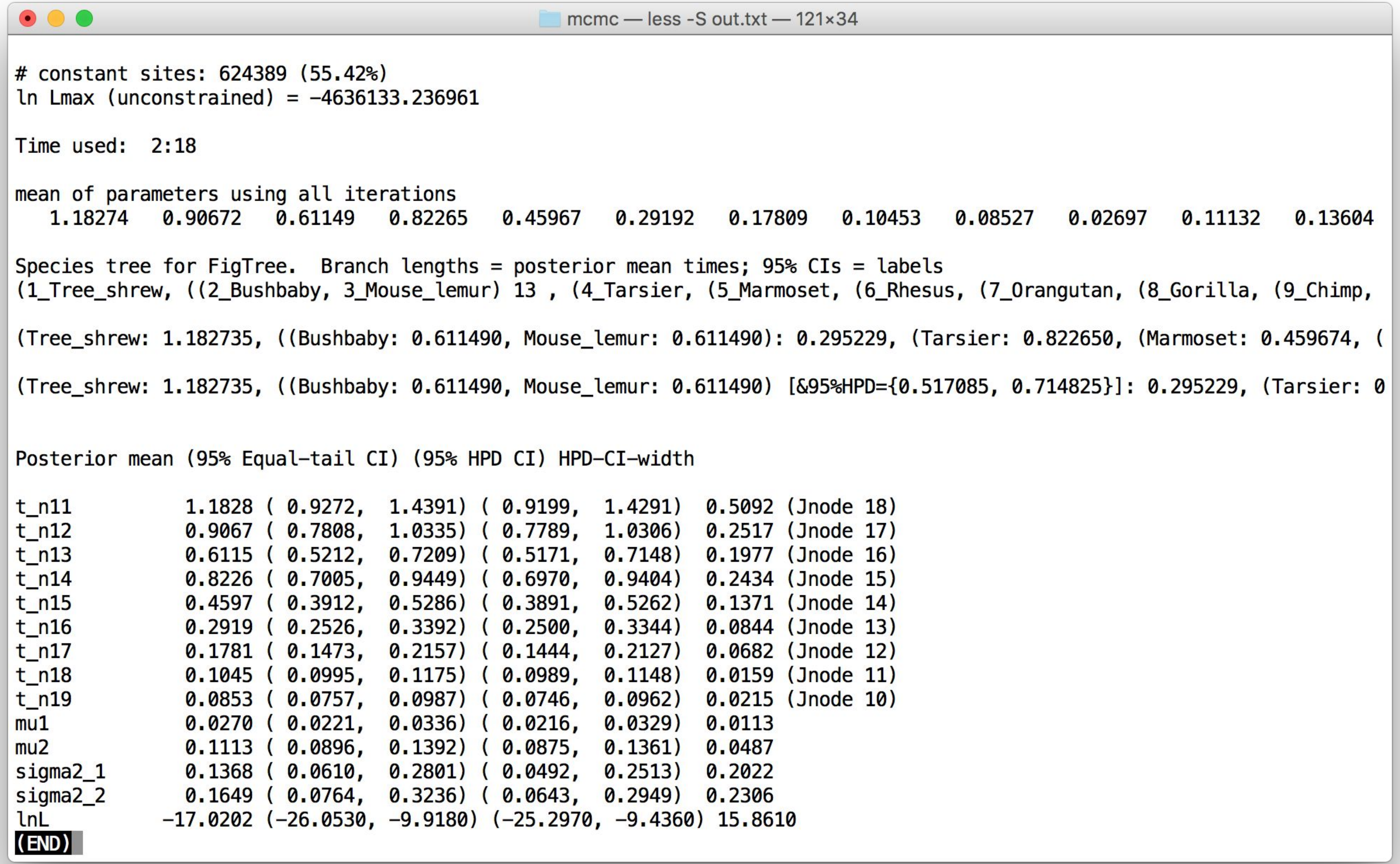
- MCMCtree will start the MCMC chain to obtain a posterior sample of primate timetrees
- Once the run is finished, MCMCtree will have created mcmc/mcmc.txt, mcmc/out.txt, and mcmc/FigTree.tre files
- Rename the files as mcmc/mcmc1.txt, mcmc/out1.txt, and mcmc/FigTree1.tre and run MCMCtree again
- This is necessary to assess convergence

Step 2: Posterior MCMC sampling
File mcmc/mcmc.txt



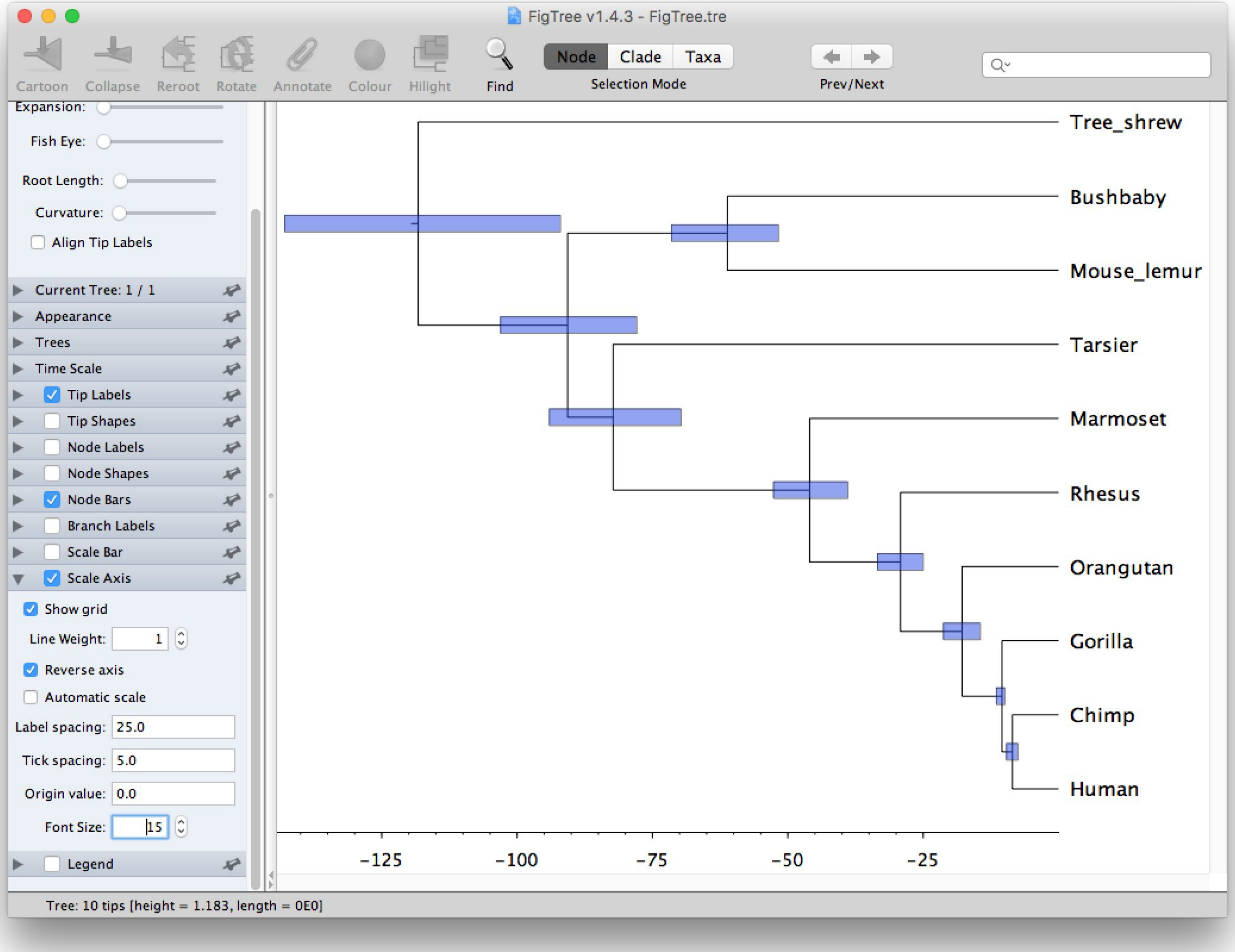
```
Gen     t_n11   t_n12   t_n13   t_n14   t_n15   t_n16   t_n17   t_n18   t_n19   mu1     mu2     sigma2_1        sigma2_2
1       1.4956478       0.8466238       0.5890040       0.7791652       0.5838105       0.2867013       0.1539901       0
100     1.5101832       0.8465849       0.5935071       0.7830479       0.5919507       0.2917796       0.1522791       0
200     1.4602814       0.8361205       0.5761541       0.7848931       0.5902324       0.2909308       0.1554728       0
300     1.4560957       0.8372168       0.5632736       0.7868402       0.5849301       0.3050102       0.1543257       0
400     1.4534685       0.8362638       0.5643719       0.7836932       0.5841846       0.3043935       0.1486133       0
500     1.4526052       0.8314959       0.5560353       0.7828864       0.5795211       0.3145948       0.1474609       0
600     1.4380462       0.8290732       0.5394369       0.7873949       0.5851622       0.3014357       0.1437796       0
700     1.4175470       0.8252979       0.5302741       0.7772600       0.5632684       0.2901634       0.1433558       0
800     1.3939014       0.8312538       0.5453034       0.7865938       0.5738942       0.2828061       0.1453689       0
900     1.3900419       0.8281506       0.5283644       0.7879186       0.5663521       0.2727386       0.1367500       0
1000    1.4497851       0.8324548       0.5224675       0.7831550       0.5619431       0.2830906       0.1319068       0
1100    1.4665192       0.8440065       0.5225911       0.7859403       0.5542077       0.2828897       0.1299686       0
1200    1.4824712       0.8435770       0.5065843       0.7820752       0.5447545       0.2934183       0.1290661       0
1300    1.4214785       0.8473784       0.4929307       0.7814759       0.5449553       0.3012859       0.1313614       0
1400    1.4351406       0.8409271       0.4788677       0.7659328       0.5330178       0.2964239       0.1334176       0
1500    1.4169766       0.8488125       0.4927271       0.7593739       0.5244676       0.2917076       0.1320802       0
1600    1.4240811       0.8557939       0.4788962       0.7688778       0.5237991       0.2914483       0.1330076       0
1700    1.4290043       0.8621785       0.4754302       0.7719766       0.5128849       0.2807836       0.1324408       0
1800    1.4007963       0.8613668       0.4781263       0.7737334       0.5036348       0.2794530       0.1326838       0
1900    1.3976055       0.8569682       0.4830466       0.7799735       0.5018574       0.2744069       0.1363005       0
2000    1.3575765       0.8489579       0.4838674       0.7744377       0.4952919       0.2776222       0.1368104       0
2100    1.3486746       0.8628296       0.4926865       0.7773963       0.5016323       0.2787184       0.1409348       0
2200    1.3489146       0.8515314       0.4974664       0.7749747       0.5018672       0.2678407       0.1455347       0
2300    1.3377558       0.8489836       0.4857455       0.7737184       0.4966193       0.2604696       0.1434914       0
2400    1.3257752       0.8437305       0.4886405       0.7799860       0.4870373       0.2695790       0.1434873       0
2500    1.3280078       0.8299278       0.4964764       0.7772518       0.4840314       0.2643404       0.1436576       0
2600    1.2934054       0.8333909       0.4940264       0.7702210       0.4663136       0.2609665       0.1427137       0
2700    1.3058661       0.8336730       0.4902217       0.7682070       0.4779101       0.2528066       0.1422329       0
2800    1.3014975       0.8236208       0.4840299       0.7628795       0.4724228       0.2501602       0.1362455       0
2900    1.3050347       0.8187004       0.4723432       0.7583963       0.4618772       0.2495729       0.1375987       0
3000    1.3044666       0.8149604       0.4787576       0.7529778       0.4583901       0.2660171       0.1326572       0
3100    1.3134347       0.8086550       0.4844851       0.7531946       0.4585487       0.2604281       0.1337584       0
mcmc.txt
```

Step 2: Posterior MCMC sampling
File mcmc/out.txt



```
# constant sites: 624389 (55.42%)
ln Lmax (unconstrained) = -4636133.236961

Time used:  2:18

mean of parameters using all iterations
   1.18274   0.90672   0.61149   0.82265   0.45967   0.29192   0.17809   0.10453   0.08527   0.02697   0.11132   0.13604

Species tree for FigTree.  Branch lengths = posterior mean times; 95% CIs = labels
(1_Tree_shrew, ((2_Bushbaby, 3_Mouse_lemur) 13 , (4_Tarsier, (5_Marmoset, (6_Rhesus, (7_Orangutan, (8_Gorilla, (9_Chimp,

(Tree_shrew: 1.182735, ((Bushbaby: 0.611490, Mouse_lemur: 0.611490): 0.295229, (Tarsier: 0.822650, (Marmoset: 0.459674, (

(Tree_shrew: 1.182735, ((Bushbaby: 0.611490, Mouse_lemur: 0.611490) [&95%HPD={0.517085, 0.714825}]: 0.295229, (Tarsier: 0


Posterior mean (95% Equal-tail CI) (95% HPD CI) HPD-CI-width

t_n11          1.1828 ( 0.9272,  1.4391) ( 0.9199,  1.4291)  0.5092 (Jnode 18)
t_n12          0.9067 ( 0.7808,  1.0335) ( 0.7789,  1.0306)  0.2517 (Jnode 17)
t_n13          0.6115 ( 0.5212,  0.7209) ( 0.5171,  0.7148)  0.1977 (Jnode 16)
t_n14          0.8226 ( 0.7005,  0.9449) ( 0.6970,  0.9404)  0.2434 (Jnode 15)
t_n15          0.4597 ( 0.3912,  0.5286) ( 0.3891,  0.5262)  0.1371 (Jnode 14)
t_n16          0.2919 ( 0.2526,  0.3392) ( 0.2500,  0.3344)  0.0844 (Jnode 13)
t_n17          0.1781 ( 0.1473,  0.2157) ( 0.1444,  0.2127)  0.0682 (Jnode 12)
t_n18          0.1045 ( 0.0995,  0.1175) ( 0.0989,  0.1148)  0.0159 (Jnode 11)
t_n19          0.0853 ( 0.0757,  0.0987) ( 0.0746,  0.0962)  0.0215 (Jnode 10)
mu1            0.0270 ( 0.0221,  0.0336) ( 0.0216,  0.0329)  0.0113
mu2            0.1113 ( 0.0896,  0.1392) ( 0.0875,  0.1361)  0.0487
sigma2_1       0.1368 ( 0.0610,  0.2801) ( 0.0492,  0.2513)  0.2022
sigma2_2       0.1649 ( 0.0764,  0.3236) ( 0.0643,  0.2949)  0.2306
lnL          -17.0202 (-26.0530, -9.9180) (-25.2970, -9.4360) 15.8610
(END)
```

Step 2: Visualization with FigTree
File mcmc/FigTree.tre



Expansion:

Fish Eye:

Root Length:

Curvature:

☐ Align Tip Labels

- Current Tree: 1 / 1
- Appearance
- Trees
- Time Scale
- ☑ Tip Labels
- ☐ Tip Shapes
- ☐ Node Labels
- ☐ Node Shapes
- ☑ Node Bars
- ☐ Branch Labels
- ☐ Scale Bar
- ☑ Scale Axis

☑ Show grid

Line Weight: 1

☑ Reverse axis

☐ Automatic scale

Label spacing: 25.0

Tick spacing: 5.0

Origin value: 0.0

Font Size: 15

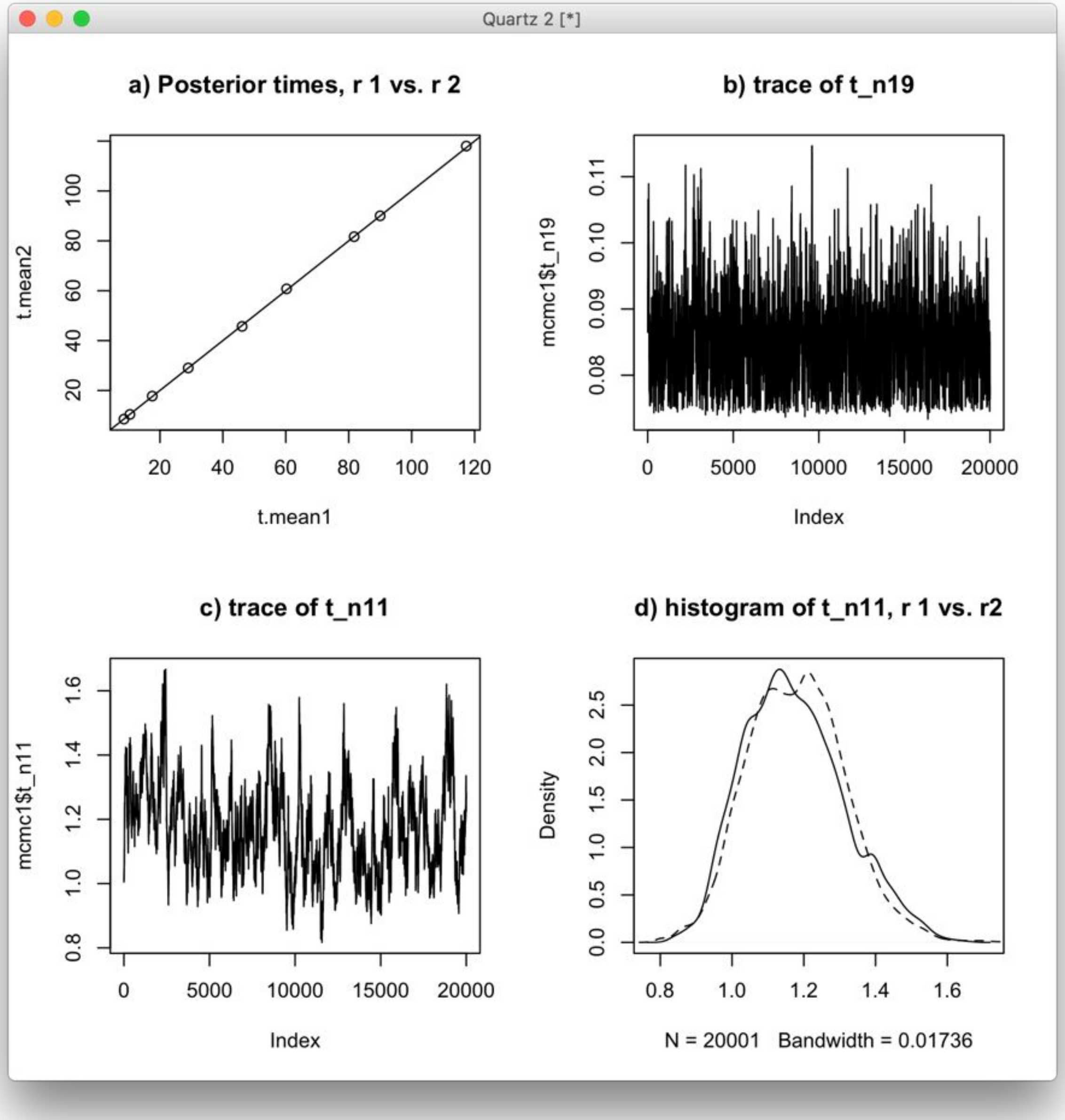
- ☐ Legend

Tree: 10 tips [height = 1.183, length = 0E0]

Step 3: Convergence diagnostics

- MCMC is a stochastic algorithm
- Thus, MCMC analyses must be run several times to verify results are consistent
- For example, you can carry out convergence diagnostics using Tracer or R

Step 3: Convergence plots in R



Exercises

- **Exercise 1:** Following the instructions in the book chapter, carry out MCMC sampling from the prior (**Hint:** usedata = 0)
- **Exercise 2:** The analysis we just carried out used the independent log-normal rates model. Repeat the analysis using auto-correlated rates (**Hint:** clock = 3)
- **Exercise 3:** Using a text editor, open the treefile, data/10s.tree, and modify the fossil calibrations. Use B(0.057, 0.10, 0.01, 0.2) for the human-chimp divergence, and use B(0.615, 1.315, 0.01, 0.05) for the root. Run the analysis again
- **Exercise 4:** Repeat the analysis using exact likelihood (**Hint:** usedata = 1, you need to reduce the number of iterations)

Further Exercises

- **Exercise 5:** Date the 330 species primate dataset (data/330s.phy and data/330s.tree). MCMC sampling of this datasets takes several hours.
- **Exercise 6:** MCMCtree now implements marginal likelihood calculation for Bayesian model selection of tree topology, substitution model and relaxed-clock model. You can follow the tutorial to choose the clock model here: https://dosreislab.github.io/2017/10/24/marginal-likelihood-mcmc3r.html