Babette practical: solution

Thijs Janzen

2024-09-15

Loading data

We can load data into R using the following functions, where we load an example file from the BEAST2 examples (alternatively, download the file from Brightspace). We then use the package ape to read the example file.

```
nexus_filename <- beastier::get_beast2_example_filename("Primates.nex")
fasta_filename <- "primates.fas"

nexus_data <- ape::read.nexus.data(nexus_filename)
output_data <- ape::as.DNAbin(nexus_data)

ape::write.FASTA(output_data, file = fasta_filename)
str(nexus_data)
```

```
## List of 12
##  $ Tarsius_syrichta: chr [1:898] "a" "a" "g" "t" ...
##  $ Lemur_catta     : chr [1:898] "a" "a" "g" "c" ...
##  $ Homo_sapiens    : chr [1:898] "a" "a" "g" "c" ...
##  $ Pan             : chr [1:898] "a" "a" "g" "c" ...
##  $ Gorilla         : chr [1:898] "a" "a" "g" "c" ...
##  $ Pongo           : chr [1:898] "a" "a" "g" "c" ...
##  $ Hylobates       : chr [1:898] "a" "a" "g" "c" ...
##  $ Macaca_fuscata  : chr [1:898] "a" "a" "g" "c" ...
##  $ M_mulatta       : chr [1:898] "a" "a" "g" "c" ...
##  $ M_fascicularis  : chr [1:898] "a" "a" "g" "c" ...
##  $ M_sylvanus      : chr [1:898] "a" "a" "g" "c" ...
##  $ Saimiri_sciureus: chr [1:898] "a" "a" "g" "c" ...
```

The last command shows us that the dataset we have obtained containes sequences of 12 primate species.

Setting up babette analysis

Again, we choose a birth-death prior:

```
bd_prior <- beautier::create_tree_prior_bd()
```

Clock model Again, we choose a relaxed log-normal model:

```
rln_clock <- beautier::create_clock_model_rln()
```

```
sub_model <- beautier::create_site_model_jc69()
```

Substitution model

Define MCMC I choose a bit longer chain, as this tends to yield better results. Please be aware that this may take quite some time to run!

```
mcmc_settings <- beautier::create_mcmc(chain_length = 3e6, store_every = 5000)
```

MRCA prior According to Finstermeier et al 2013 (doi:10.1371/journal.pone.0069504), the crown age of primates is 66-69 MYA. This corresponds to a normal distribution with mean 67.5 and sd of 0.75 (giving a 95% CI of [66, 69]). This contrasts later findings by Dos Reis et al. 2018 (doi:10.1093/sysbio/syy001), who compare different clock models and obtain an estimate of 70-79 MYA (mean = 74.6, sd = 2.35). Which one do you think will provide better results?

```
mrca_prior <- beautier::create_mrca_prior(
     taxa_names = beautier::get_taxa_names("primates.fas"),
     is_monophyletic = TRUE,
     mrca_distr = beautier::create_normal_distr(mean = 67.5, sigma = 0.75))
```

Running Babette

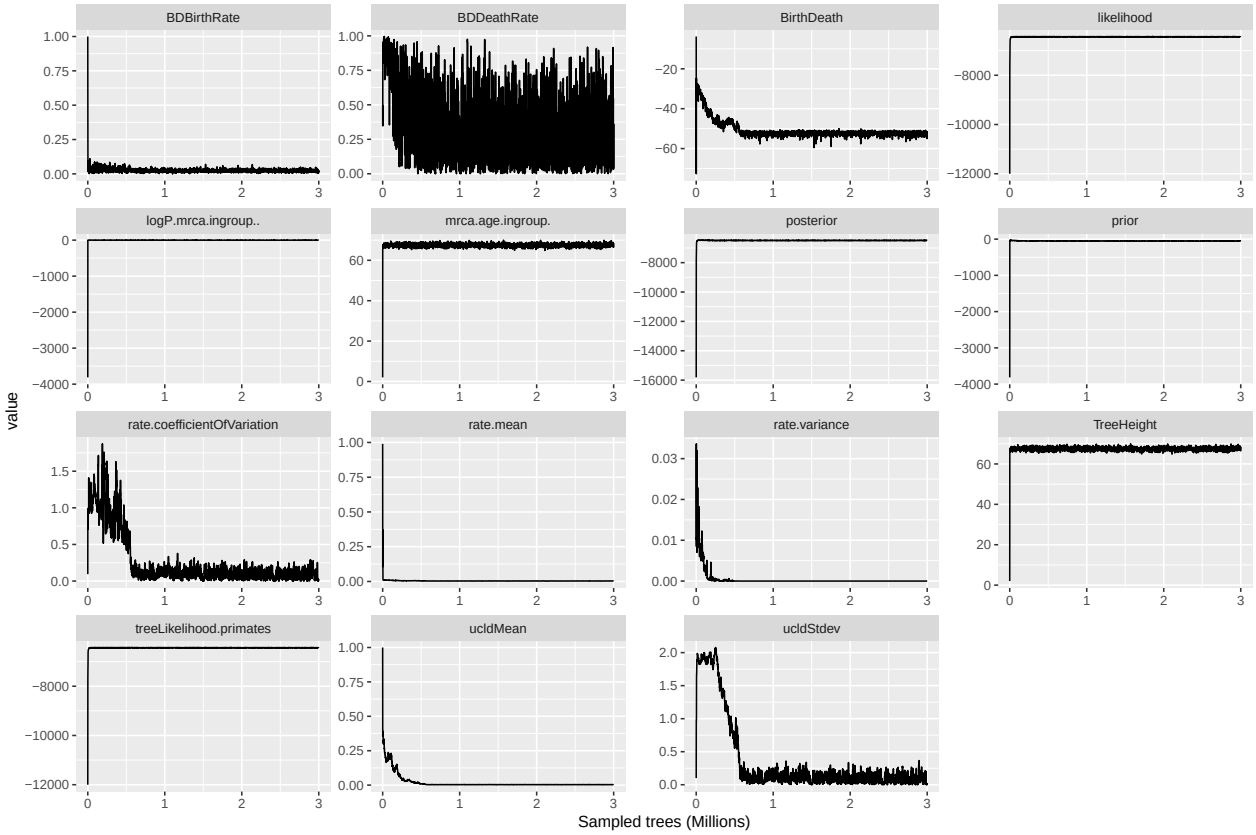
Now we can combine our priors and mcmc settings into an inference model, and pass this inference model to babette to start the inference

```
beauti_options <- beautier::create_beauti_options_v2_6()

inf_model <- beautier::create_inference_model(tree_prior = bd_prior,
                                              clock_model = rln_clock,
                                              site_model = sub_model,
                                              mcmc = mcmc_settings,
                                              mrca_prior = mrca_prior,
                                              beauti_options = beauti_options)

beast_result <- babette::bbt_run_from_model(fasta_filename = "primates.fas",
                                            inference_model = inf_model,
                                            beast2_options =
                                  beastier::create_beast2_options(rng_seed = 421))
```

Analysing results

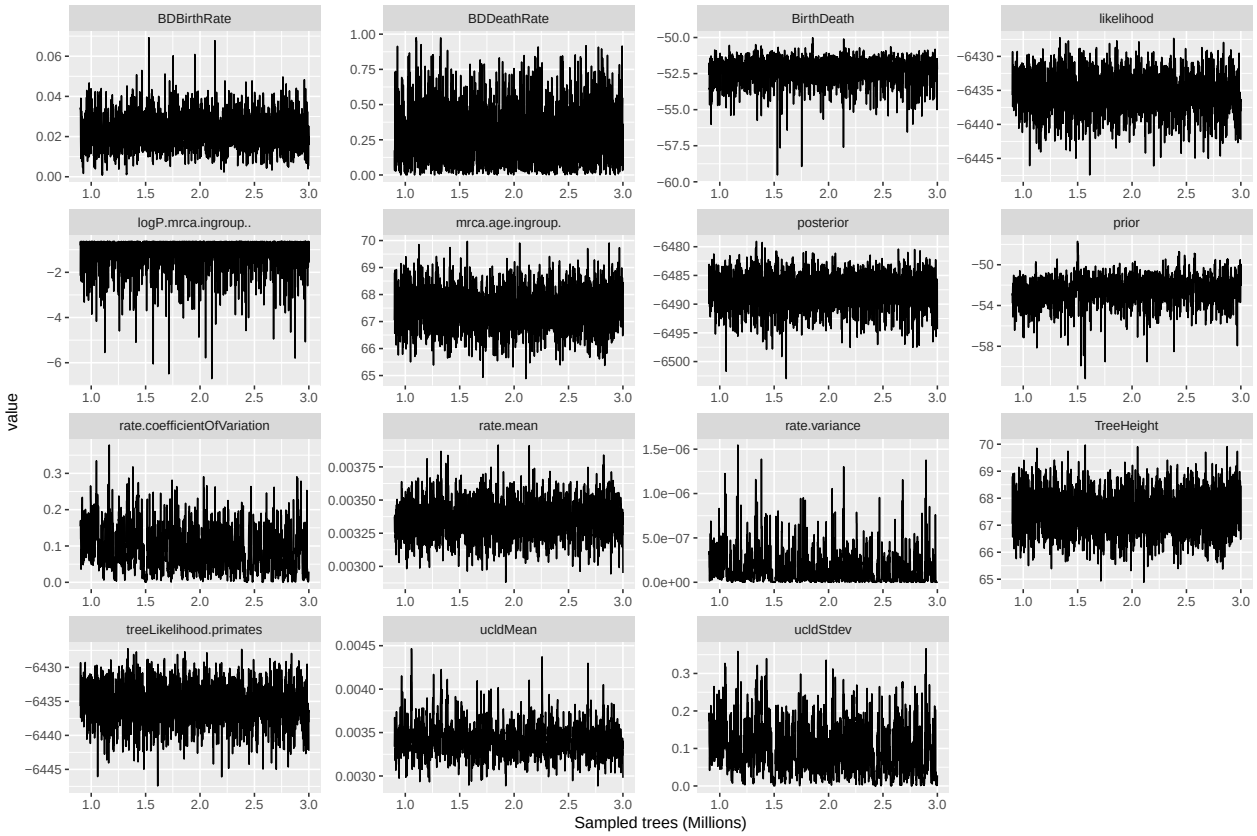
Now that we have our results, we can plot them and see how we did:

```
beast_result$estimates %>%
  mutate("Sample" = Sample / 1000000) %>%
  gather(key = "statistic", val = "value", -c(Sample)) %>%
  ggplot(aes(x = Sample, y = value)) +
  geom_line() +
  facet_wrap(~statistic, scales = "free") +
  xlab("Sampled trees (Millions)")
```



Clearly, burnin is an issue, let's first remove 30%:

```
estimates <- tracerer::remove_burn_ins(beast_result$estimates, 0.3)
estimates %>%
  mutate("Sample" = Sample / 1000000) %>%
  gather(key = "statistic", val = "value", -c(Sample)) %>%
  ggplot(aes(x = Sample, y = value)) +
  geom_line() +
  facet_wrap(~statistic, scales = "free") +
  xlab("Sampled trees (Millions)")
```




```
tracerer::calc_esses(estimates, sample_interval = 5000)
```

```
##     posterior likelihood prior treeLikelihood.primates TreeHeight ucldMean
## 901      1388        948   623                     948       2028       456
##     ucldStdev rate.mean rate.variance rate.coefficientOfVariation BirthDeath
## 901       246       798           435                         272       1337
##     BDBirthRate BDDeathRate logP.mrca.ingroup.. mrca.age.ingroup.
## 901        1352        1024                2101              2028
```

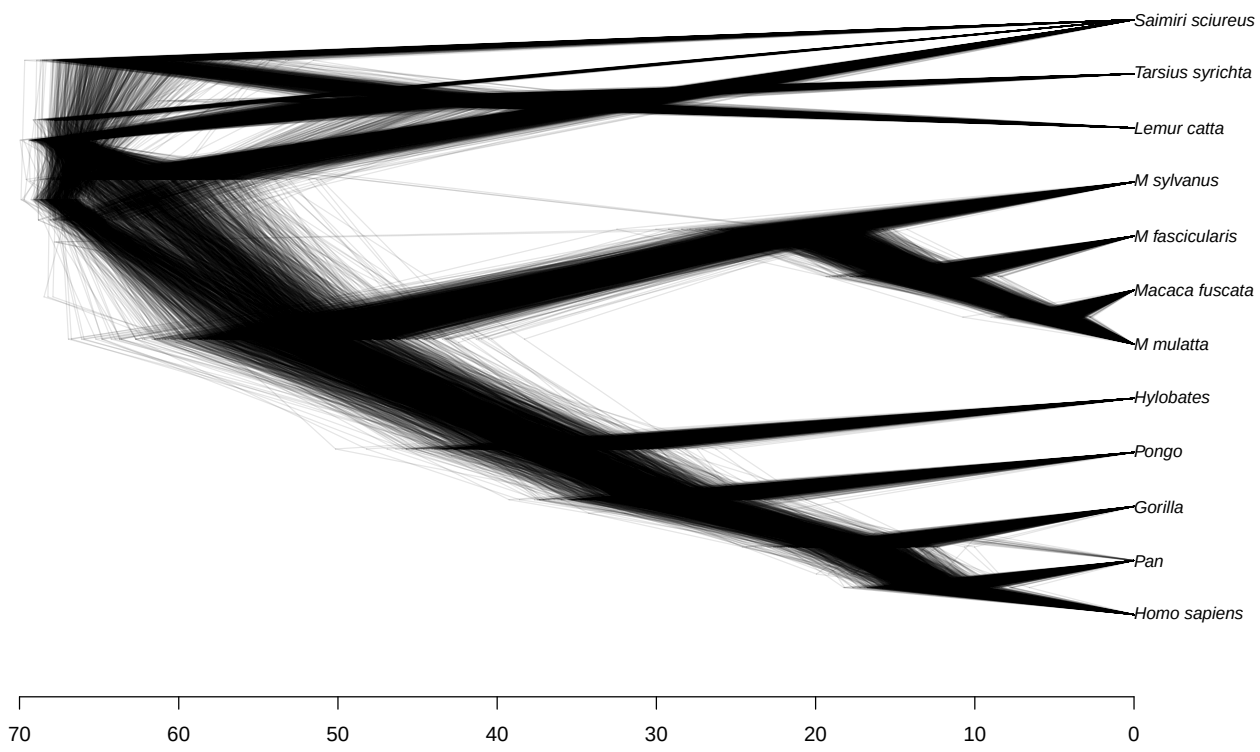
All ESS values are » 200, which is very satisfying, so 10% is enough burnin. We now also have to remove the accompanying trees of those 10% by hand:

```
all_trees <- beast_result$primates_trees
start <- floor(0.3 * length(all_trees))
end <- length(all_trees)
all_trees <- all_trees[start:end]
```

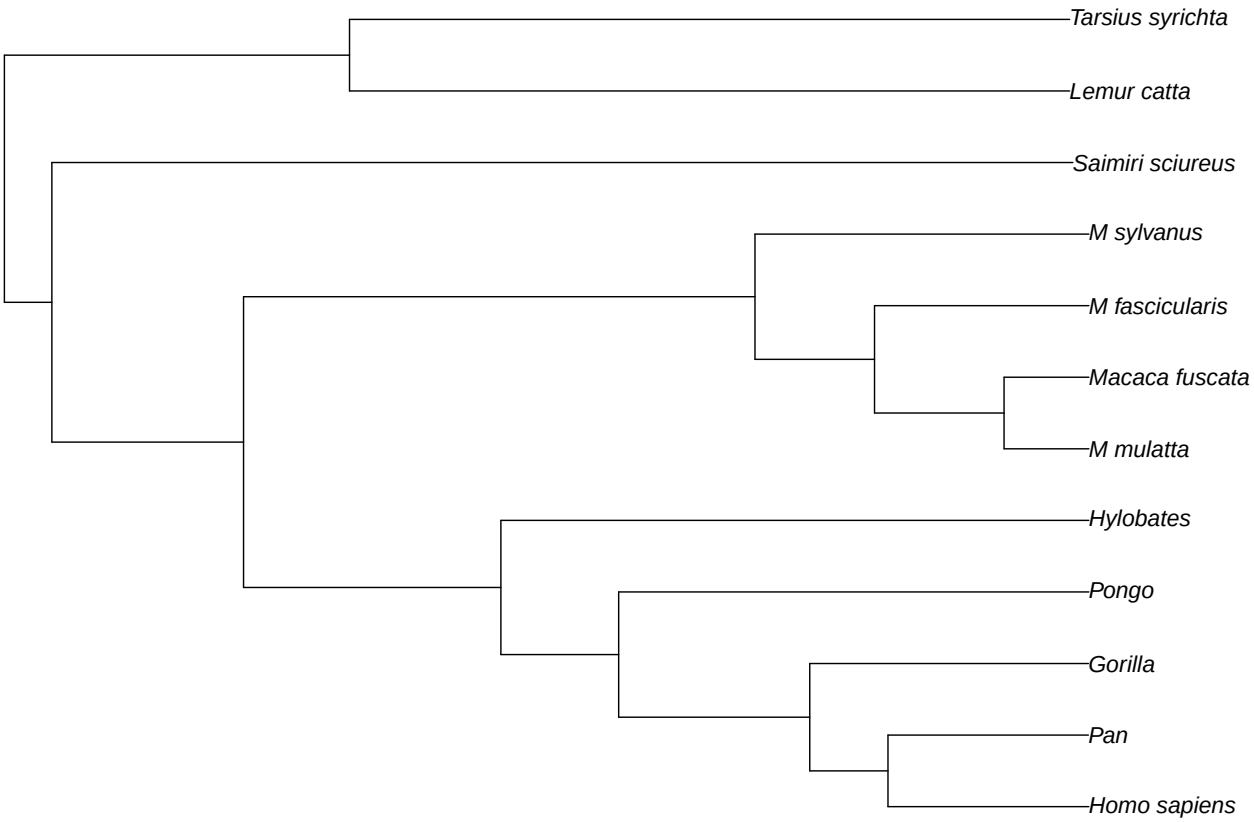
Resulting tree(s)

Having removed the burnin, we can now plot our results in two ways: 1) using a densi-tree, which superimposes all trees on top of each other and 2) using a consensus tree, which calculates the 'average' resulting tree from our distribution of trees.

```
babette::plot_densitree(all_trees, alpha = 0.1)
```




```
cons_tree <- phytools::consensus.edges(trees = all_trees)
plot(cons_tree)
```



Does this tree make sense to you?

Age of Homo Sapiens An interesting tidbit would be to infer the age of Homo Sapiens. Let's take the consensus tree and find this. In here, we have to combine some information. First, we have to know which index is used to index Homo sapiens. In the tip.labels, we find the index of 2:

```
cons_tree$tip.label
```

```
##  [1] "Gorilla"          "Homo_sapiens"     "Hylobates"        "Lemur_catta"
##  [5] "M_fascicularis"   "M_mulatta"        "M_sylvanus"       "Macaca_fuscata"
##  [9] "Pan"              "Pongo"            "Saimiri_sciureus" "Tarsius_syrichta"
```

```
human_index <- which(cons_tree$tip.label == "Homo_sapiens")
```

Phylogenies in R are stored with two information objects: one edge table, and one edge-length vector. The edge table indicates which nodes are connected with each other and with tips. The edge-length vector indicates the lengths of these connections (e.g. branch lengths). We now know to look for an edge leading towards tip '2':

```
cons_tree$edge
```

```
##       [,1] [,2]
##  [1,]   13   14
##  [2,]   14   15
##  [3,]   15   16
```

```
##  [4,]   16   17
##  [5,]   17   18
##  [6,]   18   19
##  [7,]   19    2
##  [8,]   19    9
##  [9,]   18    1
## [10,]   17   10
## [11,]   16    3
## [12,]   15   20
## [13,]   20   21
## [14,]   21   22
## [15,]   22    6
## [16,]   22    8
## [17,]   21    5
## [18,]   20    7
## [19,]   14   11
## [20,]   13   23
## [21,]   23    4
## [22,]   23   12
```

```
tip_index <- which(cons_tree$edge[, 2] == human_index)
tip_index
```

```
## [1] 7
```

This is on the 7th position. We can now use this to look up our branch length:

```
cons_tree$edge.length[tip_index]
```

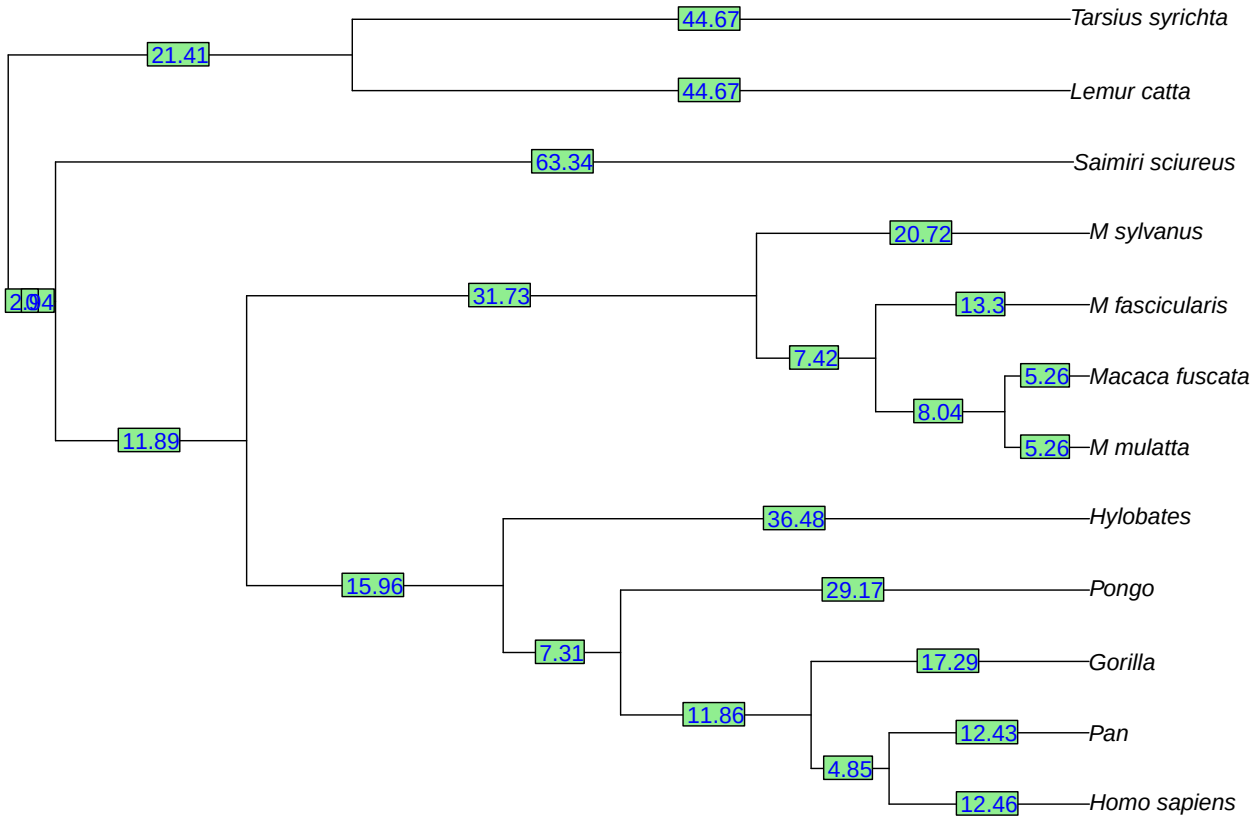
```
## [1] 12.46185
```

Alternatively, we can also plot the phylogeny with edge labels!

```
plot(cons_tree)
ape::edgelabels(round(cons_tree$edge.length, 2), col = "blue")
```

```
## Warning in XX - width * adj[1]: longer object length is not a multiple of
## shorter object length
```

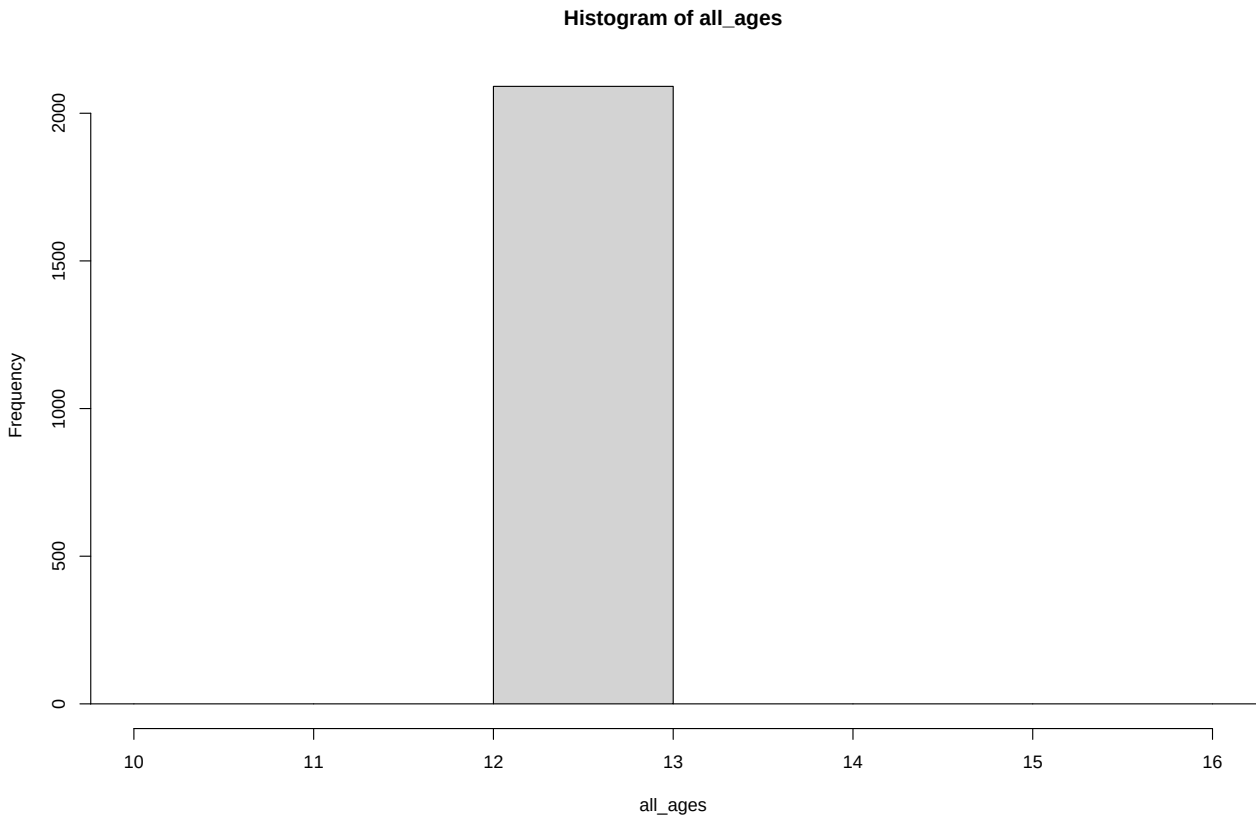
```
## Warning in YY - height * adj[2]: longer object length is not a multiple of
## shorter object length
```



This informs us we share an ancestor with Pan (Chimpansee) about 12.5MYA, and an ancestor with Gorilla ~17.3 MYA.

If we want the uncertainty in this estimate, we have to go over all trees in the posterior. And we can do so with lapply, which applies a function to a list. Because our all_trees object is a list in R, we can do the following:

```
get_homo_age <- function(tree) {
  human_index <- which(tree$tip.label == "Homo_sapiens")
  tip_index <- which(tree$edge[, 2] == human_index)
  return(cons_tree$edge.length[tip_index])
}

all_ages <- lapply(all_trees, get_homo_age)
all_ages <- unlist(all_ages)
hist(all_ages, xlim = c(10, 16))
```




```
mean(all_ages)
```

```
## [1] 12.44923
```

```
median(all_ages)
```

```
## [1] 12.43287
```

```
quantile(all_ages, c(0.025, 0.975))
```

```
##     2.5%    97.5%
## 12.43287 12.46185
```

This shows that under our current analysis, our uncertainty of the age estimate is very low, e.g. the 95% CI interval is very narrow around our mean estimate. We also see that the histogram looks a bit weird. This is due to a number of trees (out of 9000!) where the branch lengths are very different:

```
length(which(all_ages < 10))
```

```
## [1] 3
```

```
length(which(all_ages > 14))
```

```
## [1] 8
```

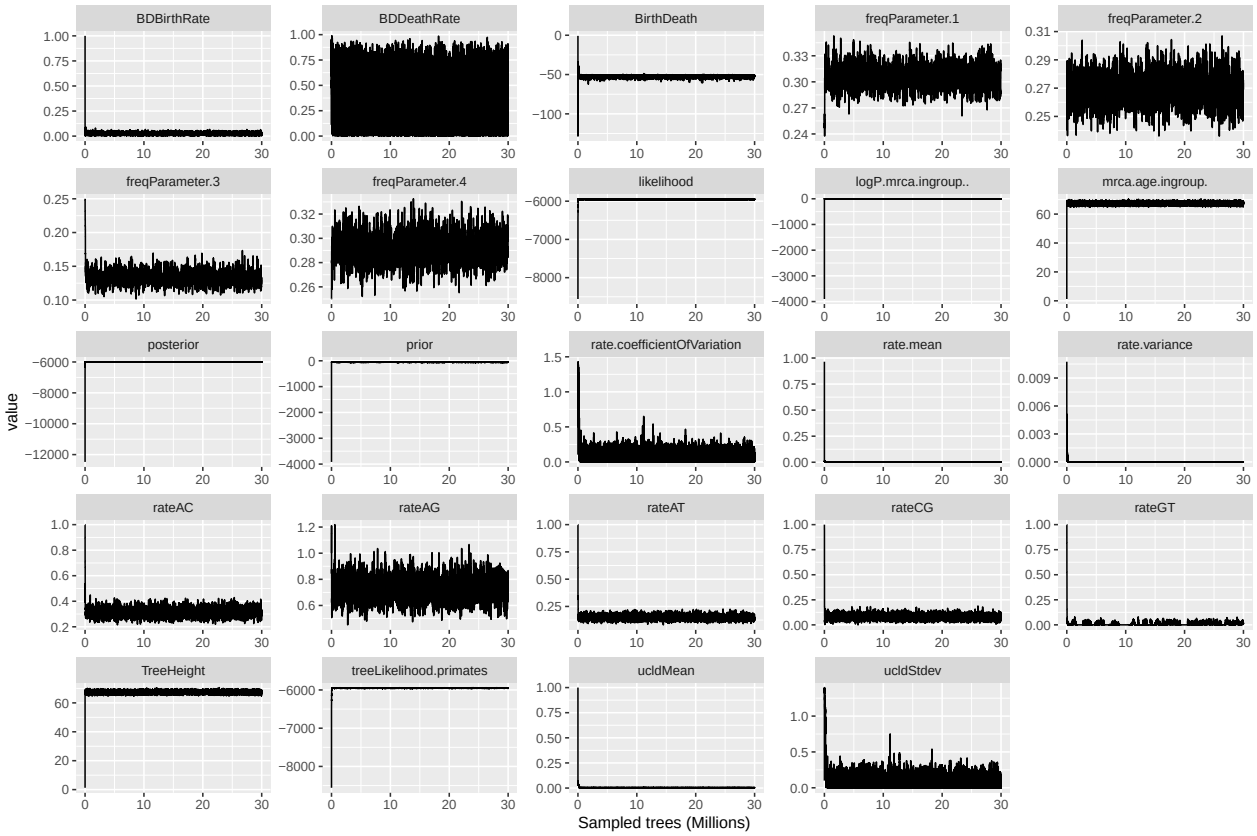
Impact of substitution model

The JC69 model is not the most realistic substitution model. Instead, let's see if using a GTR model changes our findings. Because we have extra parameters to estimate, we use a longer chain.

```
sub_model <- beautier::create_site_model_gtr()

mcmc_settings <- beautier::create_mcmc(chain_length = 3e7, store_every = 5000)

inf_model <- beautier::create_inference_model(tree_prior = bd_prior,
                                              clock_model = rln_clock,
                                              site_model = sub_model,
                                              mcmc = mcmc_settings,
                                              mrca_prior = mrca_prior,
                                              beauti_options = beauti_options)

beast_result_gtr <- babette::bbt_run_from_model(fasta_filename = "primates.fas",
                                                inference_model = inf_model,
                                                beast2_options =
                                  beastier::create_beast2_options(rng_seed = 4321))
```

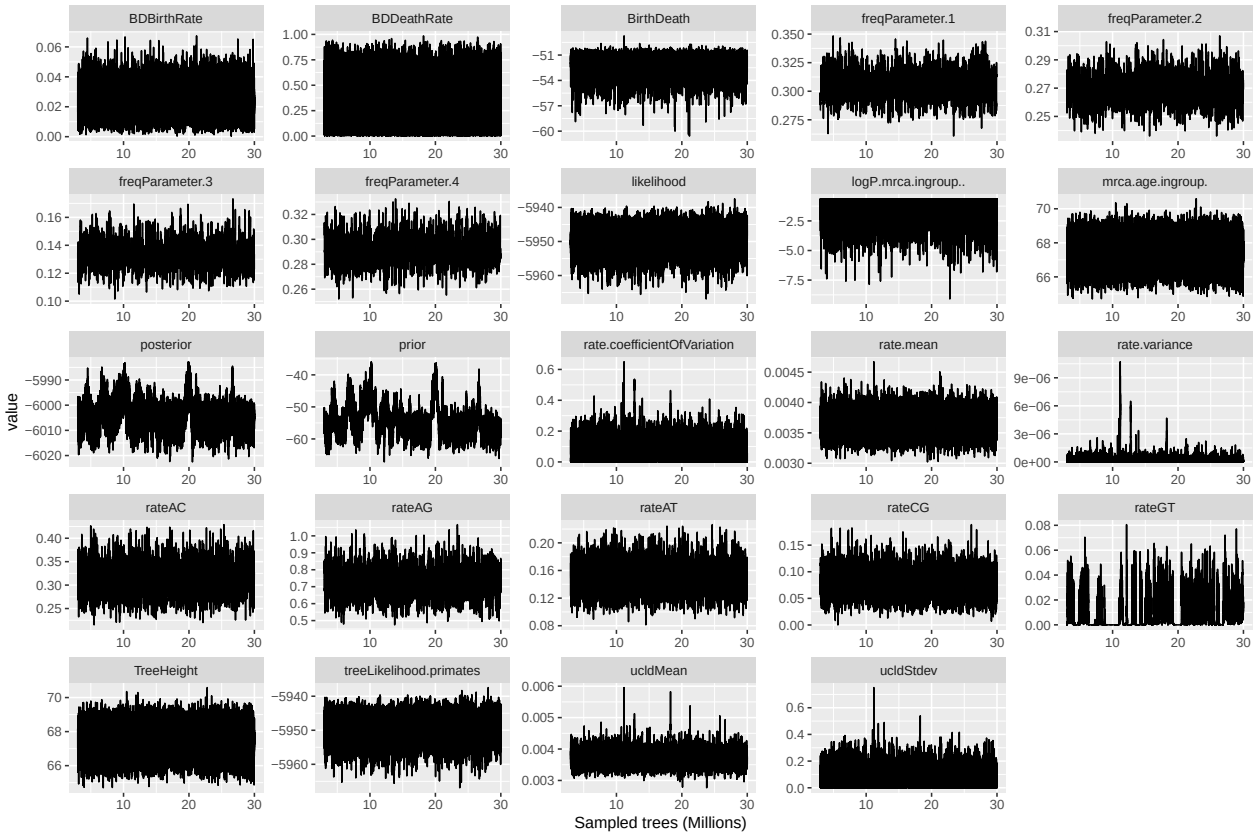
We find the following results:

```
beast_result_gtr$estimates %>%
  mutate("Sample" = Sample / 1000000) %>%
  gather(key = "statistic", val = "value", -c(Sample)) %>%
  ggplot(aes(x = Sample, y = value)) +
  geom_line() +
  facet_wrap(~statistic, scales = "free") +
  xlab("Sampled trees (Millions)")
```



After removal of 10% burn-in we have:

```
estimates_gtr <- tracerer::remove_burn_ins(beast_result_gtr$estimates, 0.1)
estimates_gtr %>%
  mutate("Sample" = Sample / 1000000) %>%
  gather(key = "statistic", val = "value", -c(Sample)) %>%
  ggplot(aes(x = Sample, y = value)) +
  geom_line() +
  facet_wrap(~statistic, scales = "free") +
  xlab("Sampled trees (Millions)")
```

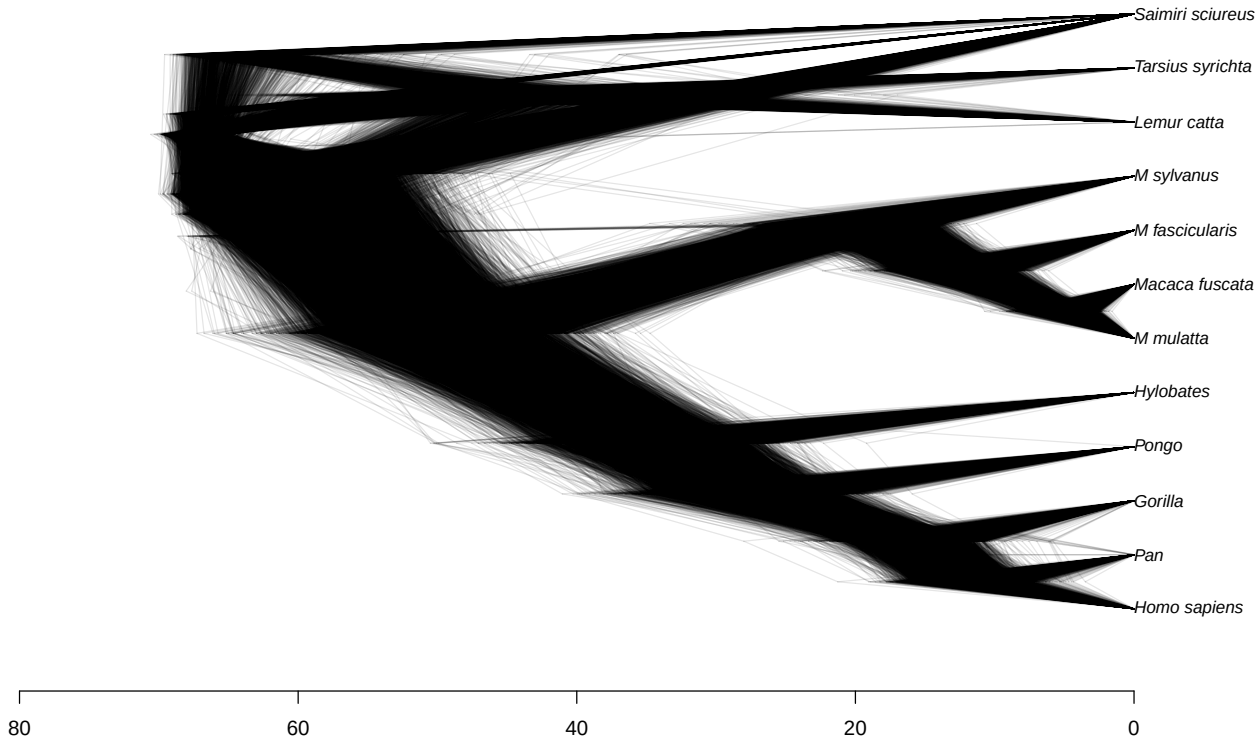



```
tracerer::calc_esses(estimates_gtr, sample_interval = 5000)
```

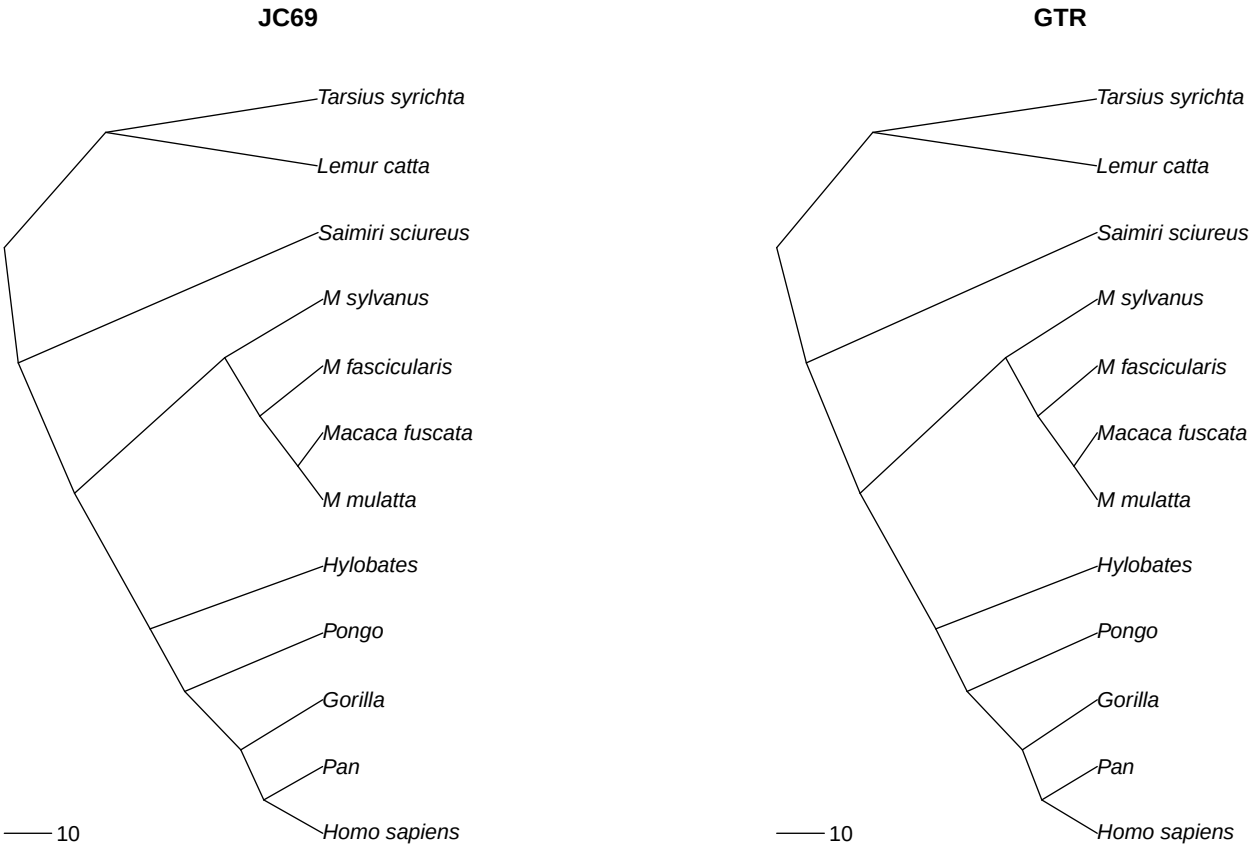
```
##      posterior likelihood prior treeLikelihood.primates TreeHeight rateAC
## 3001        62       2464    40                    2464      25723   1550
##      rateAG rateAT rateCG rateGT freqParameter.1 freqParameter.2
## 3001   1096   1963    746     97             937             724
##      freqParameter.3 freqParameter.4 ucldMean ucldStdev rate.mean rate.variance
## 3001             923             990     4513      4714      7525          2609
##      rate.coefficientOfVariation BirthDeath BDBirthRate BDDeathRate
## 3001                        5041      17918       17983       14823
##      logP.mrca.ingroup.. mrca.age.ingroup.
## 3001               27001             25723
```

This seems sufficient to continue:

```
all_trees_gtr <- beast_result_gtr$primates_trees
start <- floor(0.1 * length(all_trees_gtr))
end <- length(all_trees_gtr)
all_trees_gtr <- all_trees_gtr[start:end]

babette::plot_densitree(all_trees_gtr, alpha = 0.1)
```




```
cons_tree_gtr <- phytools::consensus.edges(trees = all_trees_gtr)
par(mfrow = c(1, 2))
plot(cons_tree, main = "JC69", type = "cladogram")
add.scale.bar()
plot(cons_tree_gtr, main = "GTR", type = "cladogram")
add.scale.bar()
```



Again, we see that the information contained in the sequences is very high, overpowering choices we have made about the substitution model.