

# MCMCTreeR

## Contents

Installation	1
Estimate parameters for node input parameters	2
Skew T . . . . .	2
estimate scale with a given shape . . . . .	2
estimate shape with a given scale . . . . .	4
Skew normal . . . . .	4
Cauchy . . . . .	6
estimate scale with a given shape . . . . .	7
Uniform distribution . . . . .	10
Gamma distribution . . . . .	11
Upper Age . . . . .	13
Fixed ages . . . . .	13
Different parameters on different nodes . . . . .	13

mcmctree is a Bayesian program contained in the software paml to estimate divergence times on fixed topologies using molecular data developed by Ziheng Yang. The program requires various inputs from the user: a phylogeny, molecular data, and selected model parameters.

A vital part of divergence time estimation in a Bayesian context are priors placed on nodes that are used to reflect **a priori** knowledge on when those lineages diverged. The guide here for mcmctreer is designed to help choose the best parameters to reflect age information for prior age distributions, visualise priors, and output files that can be read into mcmctree for analysis.

This guide does not include details about which priors are appropriate for the data, etc, so please see the mcmctree manual.

## Installation

```
## uncomment if devtools not previously installed
## install.packages('devtools')
library(devtools)
# install_github('PuttickMacroevolution/MCMCTreeR')
library(MCMCTreeR)

## Loading required package: ape
## Loading required package: sn
## Loading required package: stats4

##
## Attaching package: 'sn'

## The following object is masked from 'package:stats':
##
##      sd

## Loading required package: coda
```

For the examples here, the data are a phylogeny of apes `apeTree`. Also contained in the data are the minimum ages `minimumTimes` and maximum ages `maximumTimes` for nodes, and the tip labels descending from each node `monophyleticGroups`. These example data can be substituted for other data

```
data(apeData)
attach(apeData)
names(apeData)
```

```
## [1] "minimumTimes"      "maximumTimes"      "monophyleticGroups"
## [4] "apeTree"
```

## Estimate parameters for node input parameters

This section includes information to estimate and plot prior age distributions for node(s) used in `mcmcree` divergence time estimation. The data required to do this is a phylogeny, minimum and maximum ages for the nodes with prior distributions, and taxa that descend from each node.

The code can be used to simultaneously estimate the parameter values and write files. This can be done across the whole tree with a specific type of distribution placed on each node, or by placing different constraints on different nodes.

By default the code here estimates the parameters that are needed for a distribution the spans from the a minimum bound (lower age) to an maximum bound (upper age). The distributions are treated so there is a 'hard' minimum and 'soft maximum', and 97.5% of the distribution falls between the minimum and maximum ages. The code can estimate parameters for the cauchy, skew-t, skew-normal, and gamma distributions shown in the `mcmcree` manual on page 50.

### Skew T

#### estimate scale with a given shape

In this `estimateSkewT` function the default assumes we want to estimate scale with a given shape value (the default shape value is 50). The result parameters show the input for the `mcmcree` format - the location (lower node age), scale, shape, and degrees of freedom

```
skewT_results <- estimateSkewT(minAge = minimumTimes, maxAge = maximumTimes,
  monoGroups = monophyleticGroups, phy = apeTree, plot = FALSE)
```

```
## [1] "warning - minProb parameter value recycled"
## [1] "warning - maxProb parameter value recycled"
## [1] "warning - estimateScale argument recycled"
## [1] "warning - estimateShape argument recycled"
## [1] "warning - estimateMode argument recycled"
## [1] "warning - shape parameter value recycled"
## [1] "warning - addMode parameter value recycled"
```

```
skewT_results$mcmcree
```

```
##
## 1 7 1
##
## 1 (((human,(chimpanzee,bonobo))'ST(0.8,0.016,50,1)',gorilla)'ST(0.6,0.024,50,1)',(orangutan,sumatran))
##
## 1 //end of file
```

The function `plotMCMCTree` plots the age distribution that will be used as the user, applied distribution for each node.

```
par(mfrow = c(2, 2), family = "Palatino")
for (i in 1:4) plotMCMCTree(skewT_results$parameters[i, ], method = "skewT",
  title = paste0("node ", i), upperTime = max(maximumTimes))
```

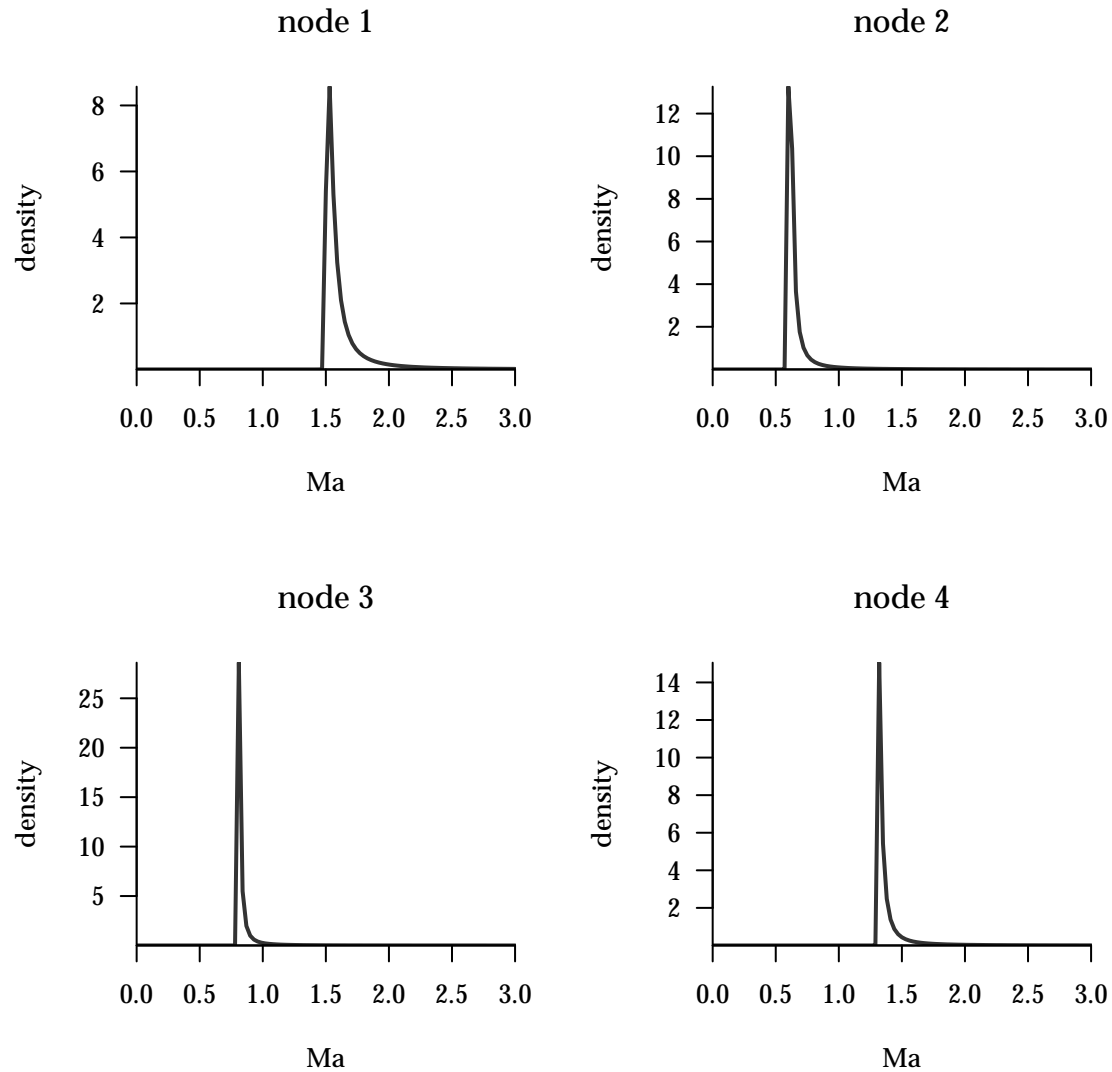


Figure 1: Skew T distributions for all nodes

```
skewT_results$mcmctree
```

```
##
## 1 7 1
##
## 1 (((human,(chimpanzee,bonobo))'ST(0.8,0.016,50,1)',gorilla)'ST(0.6,0.024,50,1)',(orangutan,sumatran))
##
## 1 //end of file
```

If the distributions are acceptable, the output is written into a tree file ready to be input into mcmcTree using the `writeMCMCtree` argument, and set the name in the `mcmcTreeName` argument. Additionally, we could have requested a pdf output of the estimate distributions by using `plot=TRUE` and specifying the file of this name using the `pdfOutput` argument

```
# result in tree MCMCtree format
skewT_results$mcmcTree

##
## 1 7 1
##
## 1 (((human,(chimpanzee,bonobo))'ST(0.8,0.016,50,1)',gorilla)'ST(0.6,0.024,50,1)',(orangutan,sumatran))'ST(0.5,0.01,50,1)'
##
## 1 //end of file

## not run skewT_results <- estimateSkewT(minAge=minimumTimes,
## maxAge=maximumTimes, monoGroups=monophyleticGroups,
## phy=apeTree, plot=FALSE, pdfOutput='skewTPlot.pdf',
## writeMCMCtree=TRUE, mcmcTreeName='skewTInput.tre')
```

It is not necessary to specify the same shape value for each parameter: a different value of shape can be set for each distribution

```
## not run (remove ## to run) skewT_results <-
## estimateSkewT(minAge=minimumTimes, maxAge=maximumTimes,
## monoGroups=monophyleticGroups, shape=c(9, 10, 8, 10),
## phy=apeTree, plot=TRUE, pdfOutput='skewTPlot.pdf',
## writeMCMCtree=TRUE, mcmcTreeName='skewTInput.tre')
## skewT_results$parameters
```

### estimate shape with a given scale

The function will take input minimum input times, and estimate the value of the shape that will produce the desired distribution with a scale equal to 0.05

```
skewT_results <- estimateSkewT(minAge = minimumTimes[2], maxAge = maximumTimes[2],
  monoGroups = monophyleticGroups, scale = 0.05, estimateShape = TRUE,
  estimateScale = FALSE, phy = apeTree, plot = FALSE, writeMCMCtree = FALSE)
skewT_results$parameters
```

```
##      location scale shape df
## node_1      0.6  0.05      1  1
```

### Skew normal

The function will take input minimum input times, and estimate the value of the scale that will produce a skew normal distribution with the 97.5% cumulative probability of the distribution at the maximum age.

```
skewNormal_results <- estimateSkewNormal(minAge = minimumTimes,
  maxAge = maximumTimes, monoGroups = monophyleticGroups, addMode = 0.05,
  phy = apeTree, plot = FALSE)
```

```
## [1] "warning - minProb parameter value recycled"
## [1] "warning - maxProb parameter value recycled"
## [1] "warning - estimateScale argument recycled"
## [1] "warning - estimateShape argument recycled"
```

```
## [1] "warning - estimateMode argument recycled"
## [1] "warning - shape parameter value recycled"
## [1] "warning - addMode parameter value recycled"
```

```
skewNormal_results$parameters
```

```
##           location scale shape
## node_1      1.55  0.65    50
## node_2      0.65  0.25    50
## node_3      0.85  0.16    50
## node_4      1.35  0.29    50
```

As only a single value is provided for each parameter, the function outputs warnings to indicate these values are recycled for each node.

These skew normal distributions can be plotted to the screen

```
par(mfrow = c(2, 2), family = "Palatino")
for (i in 1:4) plotMCMCTree(skewNormal_results$parameters[i,
], method = "skewNormal", title = paste0("node ", i), upperTime = max(maximumTimes))
```

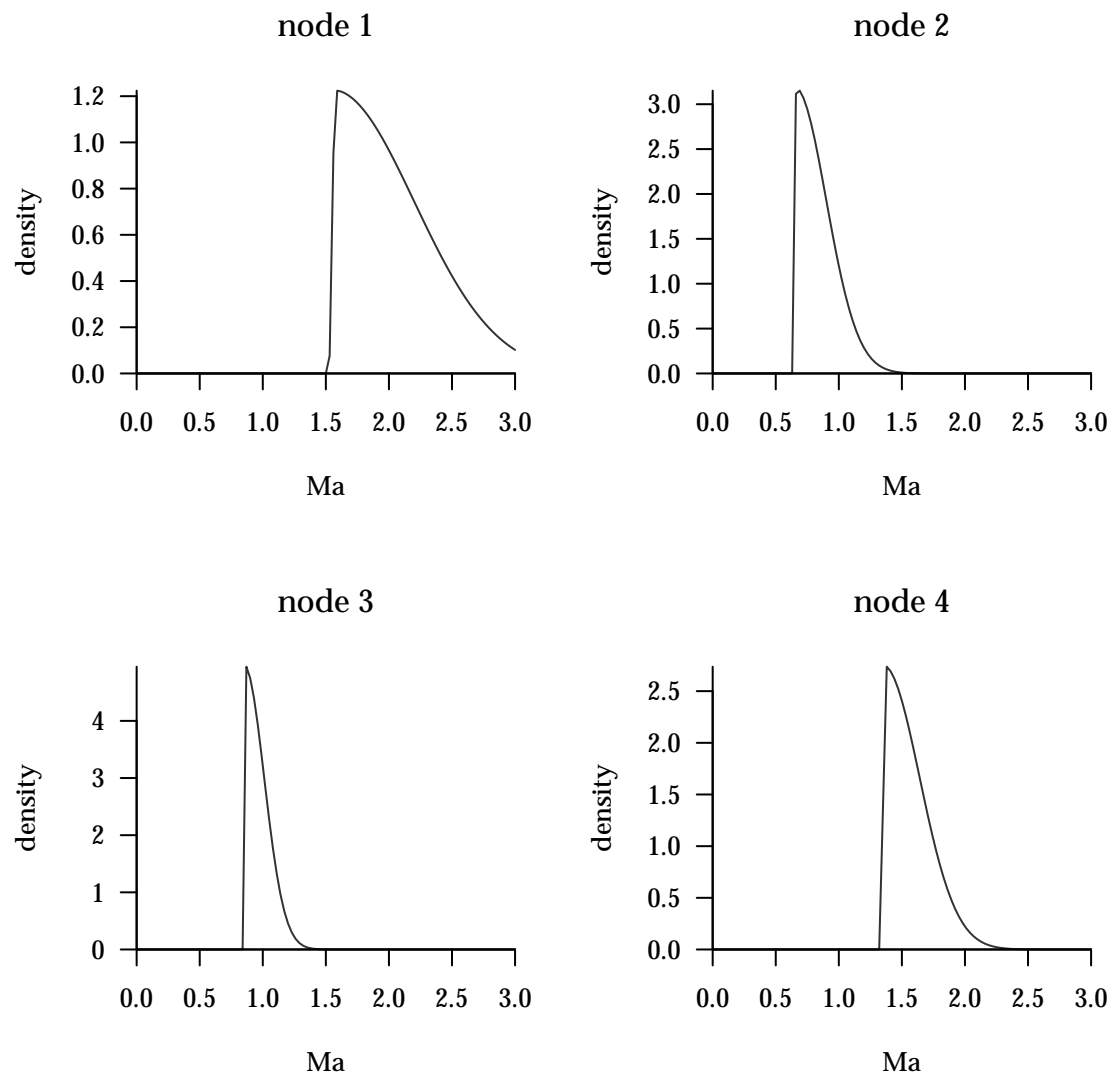


Figure 2: Skew normal distributions for all nodes

## Cauchy

plot example on page 50 of PAML manual

```
example_page_50 <- estimateCauchy(minAge = 1, maxAge = 4.32,
  monoGroups = monophyleticGroups[[1]], phy = apeTree, offset = 0.5,
  minProb = 0.025, plot = FALSE)[[1]]
plotMCMCTree(example_page_50, method = "cauchy", title = paste0("node ",
  i), upperTime = max(maximumTimes))
```

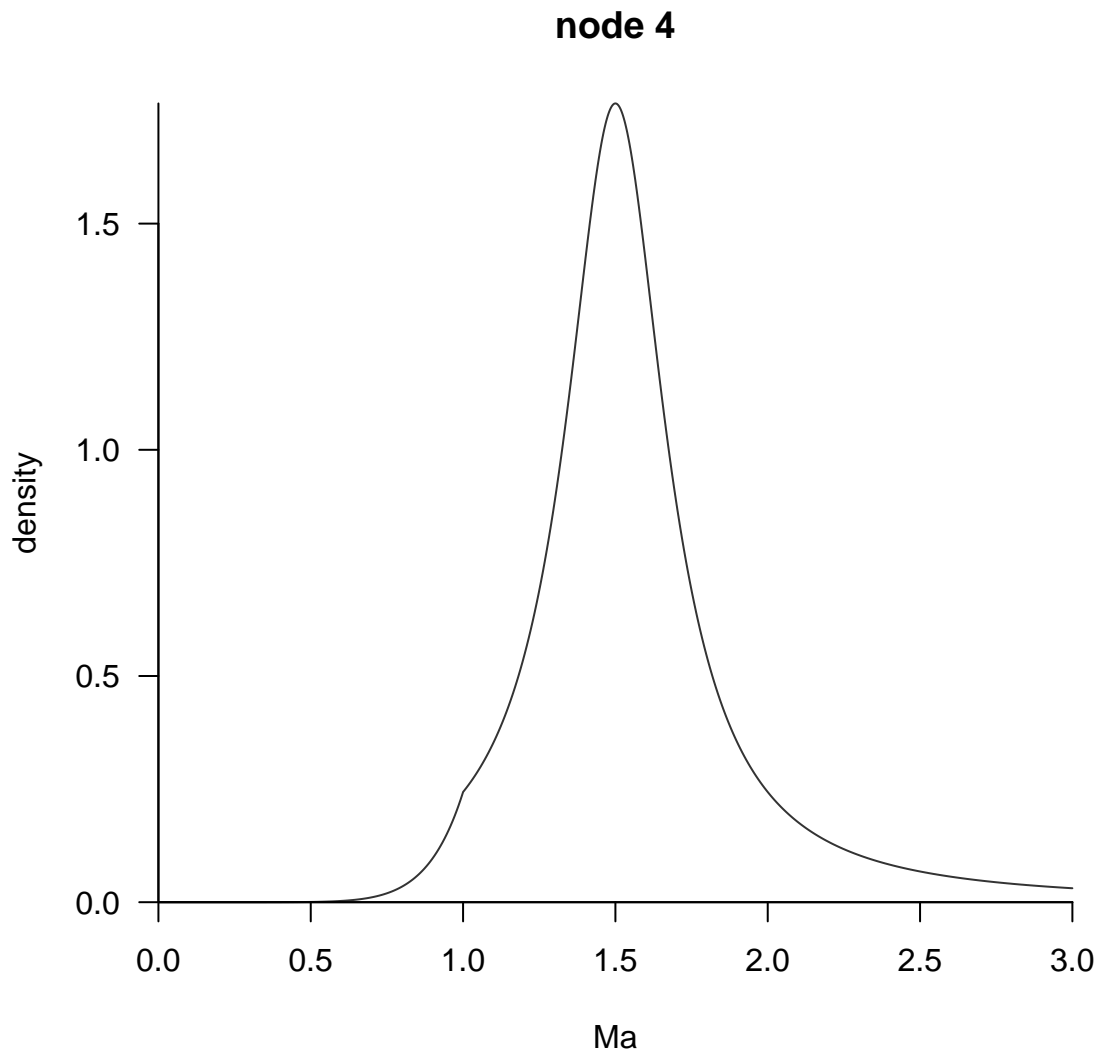


Figure 3: Cauchy distributions for all nodes (with a given scale)

#### estimate scale with a given shape

The function will take input minimum input times, and estimate the value of the scale that will produce a cauchy distribution with the 97.5% cumulative probability of the distribution at the maximum age.

```
cauchy_results <- estimateCauchy(minAge = minimumTimes, maxAge = maximumTimes,
  monoGroups = monophyleticGroups, offset = 0.5, phy = apeTree,
  plot = FALSE)
```

```
## [1] "warning - minProb parameter value recycled"
## [1] "warning - maxProb parameter value recycled"
## [1] "warning - offset parameter value recycled"
## [1] "warning - scale parameter value recycled"
## [1] "warning - estimateScale argument recycled"
```

```
cauchy_results$parameters
```

```
##          tL    p      c    pL
## node_1 1.5 0.5 0.075 1e-300
## node_2 0.6 0.5 0.008 1e-300
## node_3 0.8 0.5 0.001 1e-300
## node_4 1.3 0.5 0.016 1e-300
```

```
par(mfrow = c(2, 2), family = "Times")
for (i in 1:4) plotMCMCTree(cauchy_results$parameters[i, ], method = "cauchy",
  title = paste0("node ", i), upperTime = max(maximumTimes))
```

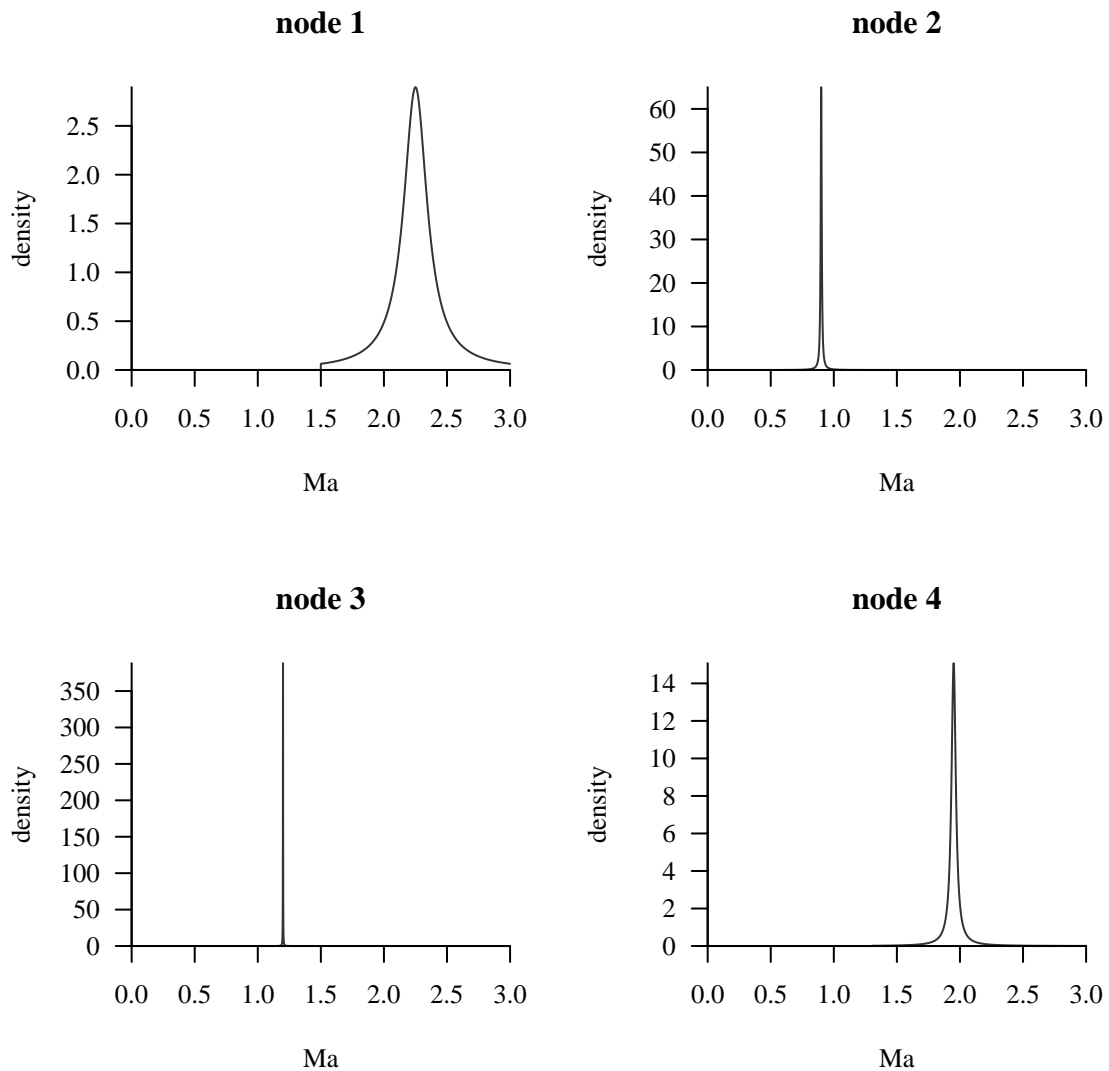


Figure 4: Cauchy distributions for all nodes (with a given shape)

We may have constrained our distribution too much for the 2nd, 3rd, and 4th distribution so we can modify that to allow for smaller offset.



```
cauchy_results <- estimateCauchy(minAge = minimumTimes, maxAge = maximumTimes,
  monoGroups = monophyleticGroups, offset = c(0.5, 0.1, 0.1,
    0.05), phy = apeTree, plot = FALSE)
```

```
## [1] "warning - minProb parameter value recycled"
## [1] "warning - maxProb parameter value recycled"
## [1] "warning - scale parameter value recycled"
## [1] "warning - estimateScale argument recycled"
```

```
cauchy_results$parameters
```

```
##      tL    p    c    pL
## node_1 1.5 0.50 0.075 1e-300
## node_2 0.6 0.10 0.035 1e-300
## node_3 0.8 0.10 0.022 1e-300
## node_4 1.3 0.05 0.040 1e-300
```

```
par(mfrow = c(2, 2), family = "Times")
for (i in 1:4) plotMCMCTree(cauchy_results$parameters[i, ], method = "cauchy",
  title = paste0("node ", i), upperTime = maximumTimes[i])
```

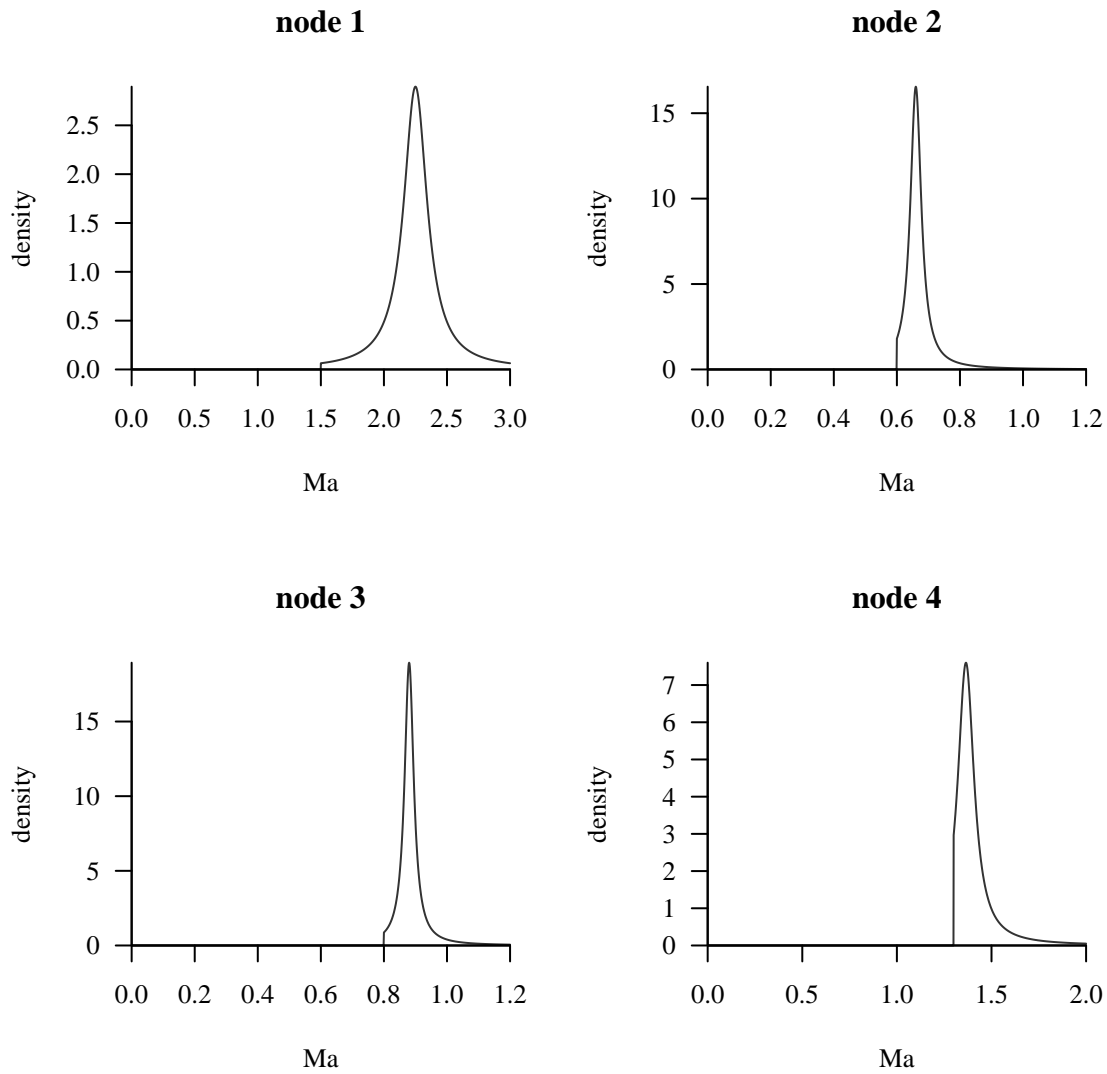


Figure 5: Cauchy distributions for all nodes (with a given shape) and smaller offset

## Uniform distribution

```
uniform_results <- estimateBound(minAge = minimumTimes, maxAge = maximumTimes,
  monoGroups = monophyleticGroups, phy = apeTree, plot = FALSE)
```

```
## [1] "warning - minProb parameter value recycled"
## [1] "warning - maxProb parameter value recycled"
```

```
uniform_results$parameters
```

```
##      tL  tU   pL   pU
## node_1 1.5 3.0 0.025 0.025
## node_2 0.6 1.2 0.025 0.025
## node_3 0.8 1.2 0.025 0.025
```

```
## node_4 1.3 2.0 0.025 0.025
```

```
par(mfrow = c(2, 2), family = "Palatino")
for (i in 1:4) plotMCMCTree(uniform_results$parameters[i, ],
  method = "bound", title = paste0("node ", i), upperTime = maximumTimes[i] +
  1)
```

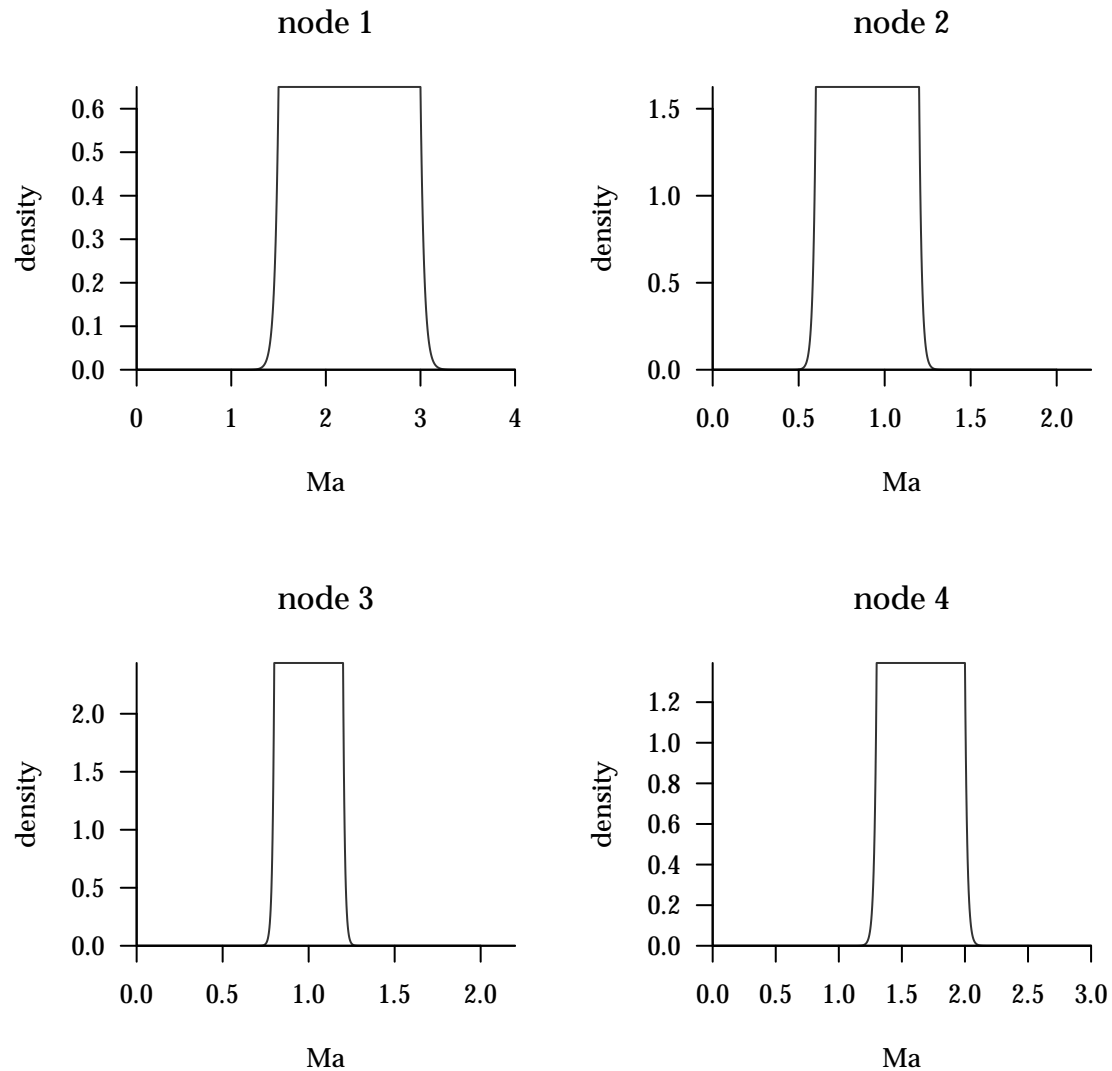


Figure 6: Uniform distributions for all nodes

## Gamma distribution

```
gamma_results <- estimateGamma(minAge = minimumTimes, maxAge = maximumTimes,
  monoGroups = monophyleticGroups, alpha = 188, beta = 2690,
  offset = 0.1, phy = apeTree, plot = FALSE)
```

```
## [1] "warning - alpha parameter value recycled"
```

```
## [1] "warning - beta parameter value recycled"
## [1] "warning - offset parameter value recycled"
## [1] "warning - estimateAlpha argument recycled"
## [1] "warning - estimateBeta argument recycled"
```

```
gamma_results$parameters
```

```
##      alpha beta
## node_1 4304 2690
## node_2 1883 2690
## node_3 2421 2690
## node_4 3766 2690
```

```
par(mfrow = c(2, 2), family = "Palatino")
for (i in 1:4) plotMCMCTree(gamma_results$parameters[i, ], method = "gamma",
  title = paste0("node ", i), upperTime = maximumTimes[i])
```

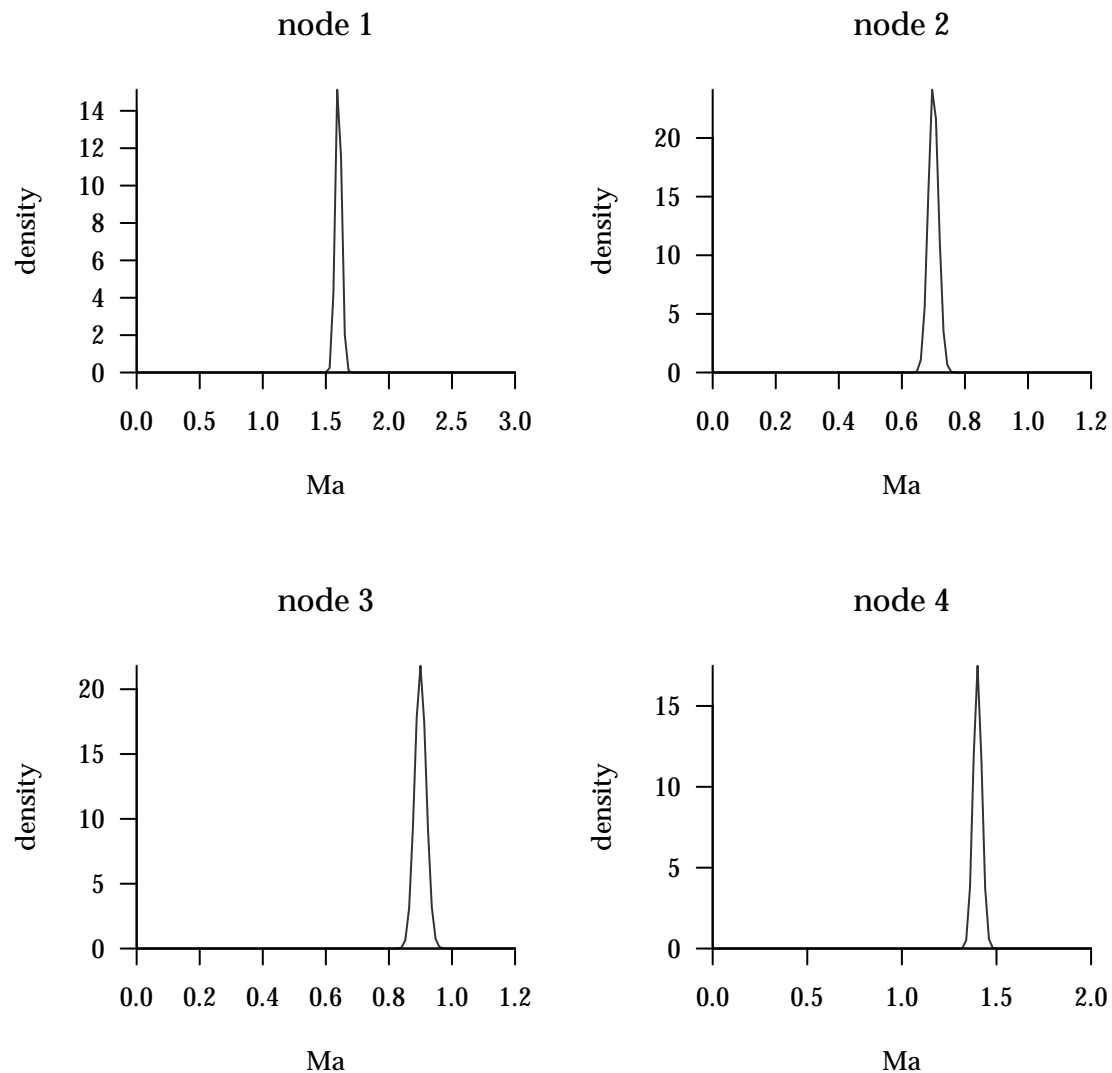


Figure 7: Gamma distributions for all nodes

## Upper Age

```
upper_results <- estimateUpper(maxAge = maximumTimes, monoGroups = monophyleticGroups,  
  rightTail = 0.025, phy = apeTree)
```

```
## [1] "warning - maxProb parameter value recycled"
```

```
upper_results$parameters
```

```
##          tU      pR  
## node_1 3.0 0.025  
## node_2 1.2 0.025  
## node_3 1.2 0.025  
## node_4 2.0 0.025
```

## Fixed ages

```
fixed_results <- estimateFixed(minAge = minimumTimes[1], monoGroups = monophyleticGroups[[1]],  
  phy = apeTree)  
fixed_results
```

```
## $parameters  
## fixed age.nodeOne  
##           1.5  
##  
## $apePhy  
##  
## Phylogenetic tree with 7 tips and 6 internal nodes.  
##  
## Tip labels:  
## human, chimpanzee, bonobo, gorilla, orangutan, sumatran, ...  
## Node labels:  
## [1] "'=1.5'"  
##  
## Rooted; no branch lengths.  
##  
## $mcmctree  
##  
## 1 7 1  
##  
## 1 (((human,(chimpanzee,bonobo)),gorilla),(orangutan,sumatran)),gibbon)'=1.5';  
##  
## 1 //end of file  
##  
## $nodeLabels  
## [1] "'=1.5'"
```

## Different parameters on different nodes

If we want different parameters on different nodes we can specify this by using the function ‘mcmcTreePhy’. Here there are various distributions: a fixed root (node 1), skew normal (node 2), gamma (node 3), and upper

distribution (node 4) to our tree. For each input we give the associated parameter values in a vector in the order of nodes. i.e - for the 'min prob' on four nodes we want nodes 1, 2, 4 to be 1e-8 and node 3 to be 0.025

```
each.node.method <- c("skewT", "cauchy", "gamma", "upper")
output.full <- mcmcTreePhy(phy = apeTree, minAge = minimumTimes,
  maxAge = maximumTimes, monoGroups = monophyleticGroups, method = each.node.method,
  writeMCMCTree = FALSE)
```

```
## [1] "length of some parameters and nodes do not match - first parameter will be used for each node"
```

This can be fine-tuned. For example, to estimate alpha not beta for the 3rd node

```
estimate.alpha <- c(FALSE, FALSE, TRUE, FALSE)
estimate.beta <- c(TRUE, TRUE, FALSE, TRUE)
outputFull <- mcmcTreePhy(phy = apeTree, minAges = minimumTimes,
  maxAges = maximumTimes, monoGroups = monophyleticGroups,
  method = each.node.method, estimateAlpha = estimate.alpha,
  estimateBeta = estimate.beta, alphaInput = 188, betaInput = 100,
  writeMCMCTree = FALSE)
```

```
## [1] "length of some parameters and nodes do not match - first parameter will be used for each node"
```

Outputs from individual methods can be added to the input to subsequent node estimation. This allows for easier fine-tuning. Perhaps easier to explain with an example. A skew normal is applied to the first node

```
skewNormal_results_nodeOne <- estimateSkewNormal(minAge = minimumTimes[1],
  maxAge = maximumTimes[1], monoGroups = monophyleticGroups[[1]],
  addMode = 0.05, phy = apeTree, plot = FALSE, writeMCMCTree = FALSE)
skewNormal_results_nodeOne$apePhy
```

```
##
## Phylogenetic tree with 7 tips and 6 internal nodes.
##
## Tip labels:
## human, chimpanzee, bonobo, gorilla, orangutan, sumatran, ...
## Node labels:
## [1] "'SN[1.55~0.65~50]'"
##
## Rooted; no branch lengths.
```

This output is then used when a Cauchy distribution is applied to the second node

```
cauchy_results_nodeTwo <- estimateCauchy(minAge = minimumTimes[2],
  maxAge = maximumTimes[2], monoGroups = monophyleticGroups[[2]],
  offset = 0.5, phy = skewNormal_results_nodeOne$apePhy, plot = FALSE,
  writeMCMCTree = FALSE)
cauchy_results_nodeTwo$apePhy
```

```
##
## Phylogenetic tree with 7 tips and 6 internal nodes.
##
## Tip labels:
## human, chimpanzee, bonobo, gorilla, orangutan, sumatran, ...
## Node labels:
## [1] "'SN[1.55~0.65~50]'"          NA
## [3] "'L[0.6~0.5~0.008~1e-300]'"
##
## Rooted; no branch lengths.
```

The third node is uniform

```
uniform_results_nodeThree <- estimateBound(minAge = minimumTimes[3],
  maxAge = maximumTimes[3], monoGroups = monophyleticGroups[[3]],
  phy = cauchy_results_nodeTwo$apePhy, plot = FALSE, writeMCMCTree = FALSE)
uniform_results_nodeThree$apePhy
```

```
##
## Phylogenetic tree with 7 tips and 6 internal nodes.
##
## Tip labels:
## human, chimpanzee, bonobo, gorilla, orangutan, sumatran, ...
## Node labels:
## [1] "'SN[1.55~0.65~50]'"          NA
## [3] "'L[0.6~0.5~0.008~1e-300]'"    "'B[0.8~1.2~0.025~0.025]'"
##
## Rooted; no branch lengths.
```

The fourth is a skewT distribution, and the tree can be written to file for input into MCMCtree

```
## not run skewT_results_nodeFour <-
## estimateSkewT(minAge=minimumTimes[4],
## maxAge=maximumTimes[4], monoGroups=monophyleticGroups[[4]],
## scale=0.5, phy=cauchy_results_nodeTwo$apePhy, plot=FALSE,
## writeMCMCTree = TRUE) skewT_results_nodeFour$apePhy
```