Sampling Aware Ancestral State Inference

Yexuan Song

2025-05-29

Contents

Introduction

This vignette serves as a guide of how to use the `saasi` package.

Note: In the future version it will also include a guide of how to visualize the inference results (phylogenetic tree and alluvial plot visualization).

saasi

Phylogenetic tree

The tree that you are interested in to infer ancestral states using 'saasi' should be a timed - rooted binary tree, including tip states (either geographic locations or some certain traits). The traits should be discrete traits.

In this vignette, we will go through saasi's workflow using a bird tree example in the `ape` package.

```
library(ape)
library(saasi)
data(bird.orders)

# set some random discrete characters to the bird tree.
x <- factor(c(rep(1, 5), rep(2, 18)))

bird.orders$tip.state <- x
```

Getting input parameters

In addition to a phylogenetic tree, saasi also requires the user to know the phylodynamic parameters (speciation, extinction, transition & sampling).

Note that the user should have some knowledge about the sampling difference between states of interest (or for testing different sampling scenarios).

The speciation and extinction rates (the time unit should be consistent with the time scale of the tree) can be estimated using "mle_lm" function. The method is described in Stadler et al. (2012).

```
estimates <- mle_lm(bird.orders,lambda = .2, mu = .05, psi = .1,lower = c(0.001,0.001), upper = c(0.5,0

estimates
#> speciation extinction
#> 0.08408307 0.00100000
```

Transition rates can be estimated using 'ace'.

```
ace_qij <- extract_ace_q(ace(x, bird.orders, type = "d"))
```

Running saasi

Now we can run `saasi` function.

```
# set up the parameters
pars <- data.frame(state=c(1,2),prior=c(0.5,0.5),lambda=c(.08,.08),mu=c(0.001,0.001),
                   psi=c(.01,.09))

saasi_result <- saasi(bird.orders,pars,ace_qij)
```

transition rate adjustment

Sometimes the transition rate adjustment using `ace` is biased due to uneven sampling. We implement a quick way of adjusting the transition rates based on the sampling assumptions.

Suppose in this example State 1 samples 1/9.

```
adj_qij <- q_adjust(ace_qij,"1",1/9)
saasi_result_with_adjusted_qij <- saasi(bird.orders,pars,adj_qij)
```

Visualization

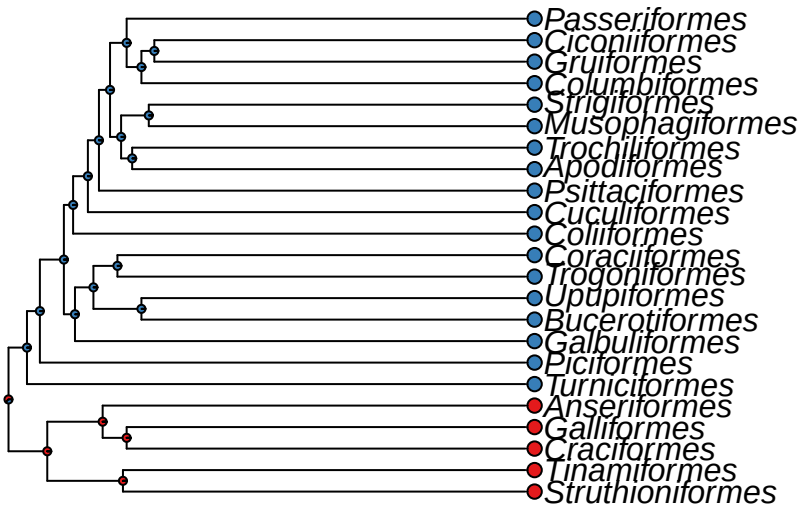
A quick way of checking the result is by using the default functions (`tiplabels` & `nodelabels`) in `ape`.

```
plot(bird.orders, label.offset = 1)

color <- c("#E41A1C", "#377EB8")

# Map tip states to colors
tip_colors <- color[bird.orders$tip.state]
tiplabels(bg = tip_colors, cex = 1, adj = 1, pch = 21)

nodelabels(pie = saasi_result,
           piecol = color[1:ncol(saasi_result)],
           cex = 0.2)# plotting
```




```
plot(bird.orders, label.offset = 1)

color <- c("#E41A1C", "#377EB8")

# Map tip states to colors
tip_colors <- color[bird.orders$tip.state]
tiplabels(bg = tip_colors, cex = 1, adj = 1, pch = 21)

nodelabels(pie = saasi_result_with_adjusted_qij,
           piecol = color[1:ncol(saasi_result_with_adjusted_qij)],
           cex = 0.2)# plotting
```