

Report

F. Richter

December 7th, 2015

Introduction

In this report an MLE estimation of the constant rates model is calculated. The general model, with non constant birth-death rates, is also discussed (I could not find an analytical MLE solution for that case).

In addition, simulated data and an intent of finding a numerical solution were performed.

Analytical analysis

We define,

- β : Speciation rate.
- δ : Extinction rate.
- t_i : Time when the event i (speciation or extinction) occurs.
- x_i : Number of species on time t_i .

The probability of a speciation occurs on time t is given by the exponential probability distribution

$$P_\beta(T = t; \beta) = -\beta e^{-\beta t}$$

whereas the probability of an extinction occurs on time t is

$$P_\delta(T = t; \delta) = -\delta e^{-\delta t}$$

Moreover, given the waiting time t_1 the probability of an speciation event occurs is $\beta/(\beta + \delta)$ and the probability an extinction happens on t_i is $\delta/(\beta + \delta)$

Thus, a possible likelihood function corresponding to the phylogenetic tree would be

$$L = \prod_{i=1}^n (\beta + \delta) x_i e^{-(\beta + \delta) x_i t_i} \left(\frac{\beta}{\beta + \delta} \right)^{b_i} \left(\frac{\delta}{\beta + \delta} \right)^{d_i}$$

$$P(\text{Birth}; \text{waiting time}) = \frac{\beta}{\beta + \delta}$$

$$P(\text{Death}; \text{waiting time}) = \frac{\delta}{\beta + \delta}$$

Figure 1: Probability of speciation and extinctions event.

where b_i is the number of speciations on time t_i and d_i is the number of extinctions on time t_i .

Thus, the loglikelihood function is

$$lL = \sum_{i=1}^n \ln(\beta + \delta) + \ln(x_i) - (\beta + \delta)x_i t_i + b_i(\ln(\beta) - \ln(\beta + \delta)) + d_i(\ln(\delta) - \ln(\beta + \delta))$$

Finally, to maximize the likelihood we solve the equations

$$\frac{\partial lL}{\partial \beta} = 0; \quad \frac{\partial lL}{\partial \delta} = 0 \quad (1)$$

The analytical solution corresponding to equations (1) are

$$\beta = \frac{N \sum b_i}{\sum x_i t_i (\sum b_i + \sum d_i)}$$

$$\delta = \frac{N \sum d_i}{\sum x_i t_i (\sum b_i + \sum d_i)}$$

This is an useful result as a first step, however, since we assume β and δ are constant numbers, this is not yet a realistic case. We would be interested in the case when the speciation and extinction parameters depends on covariates, for instance

$$\log(\beta) = \theta_0 + \theta_1 \text{body size} + \theta_2 \text{ave. temp.}$$

We consider the simple case

$$\beta = e^{\theta_0 + \theta_1 a}$$

$$\delta = e^{\phi_0 + \phi_1 a}$$

and we call it the model M2.

For MLE calculation, by the chain rule we have

$$\frac{\partial lL}{\partial \theta_0} = \frac{\partial lL}{\partial \beta} \frac{\partial \beta}{\partial \theta_0} = e^{\theta_0 + \theta_1 a} \frac{\partial lL}{\partial \beta};$$

$$\frac{\partial lL}{\partial \theta_1} = \frac{\partial lL}{\partial \beta} \frac{\partial \beta}{\partial \theta_1} = a e^{\theta_0 + \theta_1 a} \frac{\partial lL}{\partial \beta},$$

and similarly

$$\frac{\partial lL}{\partial \phi_0} = e^{\phi_0 + \phi_1 a} \frac{\partial lL}{\partial \delta};$$

$$\frac{\partial lL}{\partial \phi_1} = a e^{\phi_0 + \phi_1 a} \frac{\partial lL}{\partial \delta}.$$

$$\log(\beta) = A\theta$$

$$\log(\delta) = A\phi$$

Figure 2: An option is to model the parameters as a GLM

$$\frac{\partial \beta}{\partial \theta_0} = e^{\theta_0 + \theta_1 a}$$

$$\frac{\partial \beta}{\partial \theta_1} = a e^{\theta_0 + \theta_1 a}$$

$$\frac{\partial \delta}{\partial \phi_0} = e^{\phi_0 + \phi_1 a}$$

$$\frac{\partial \delta}{\partial \phi_1} = a e^{\phi_0 + \phi_1 a}$$

Thus, if $a \neq 0$, the system

$$\frac{\partial L}{\partial \theta_0} = 0; \quad \frac{\partial L}{\partial \theta_1} = 0; \quad \frac{\partial L}{\partial \phi_0} = 0; \quad \frac{\partial L}{\partial \phi_1} = 0 \quad (2)$$

is equivalent to (1) and we have two equations but four parameters.

Simulations

We simulate data from the M2 model:

```
nT = 15
theta = c(1, 2)
phi = c(0, 1)
set.seed(123)
dat <- NULL
AA <- NULL
AA <- list()
A = rexp(1, 4)
dat[[1]] <- list(tm = 0, x = 1, A = A)
X = 0
X[1] = 1
tm = 0
D = 0
B = 0
Tm = tm
AA[[1]] = A
i <- 2
stm <- 0
while (i <= nT & dat[[i - 1]]$x > 0) {
  beta <- theta[1] + theta[2] * dat[[i - 1]]$A
  delta <- phi[1] + phi[2] * dat[[i - 1]]$A
  prev.x = dat[[i - 1]]$x
  tm <- rexp(1, (sum(beta) + sum(delta)))
  stm <- stm + tm
  prob <- c(delta, beta)/(sum(beta) + sum(delta))
  BD <- sample(2 * prev.x, 1, prob = prob)
  if (BD <= prev.x) {
    x = prev.x - 1
    A = dat[[i - 1]]$A[-BD]
    D[i] = 1
    B[i] = 0
  } else {
    x = prev.x + 1
    A = c(dat[[i - 1]]$A, dat[[i - 1]]$A[BD -
```

```

        prev.x] * rgamma(1, 100, 100))
    D[i] = 0
    B[i] = 1
  }
  dat[[i]] <- list(tm = stm, x = x, A = A)
  X[i] = x
  Tm[i] = tm
  AA[[i]] = A

  i <- i + 1
}

```

To see the iterations we type

```

dat

## [[1]]
## [[1]]$tm
## [1] 0
##
## [[1]]$x
## [1] 1
##
## [[1]]$A
## [1] 0.2108643
##
##
## [[2]]
## [[2]]$tm
## [1] 0.3531868
##
## [[2]]$x
## [1] 2
##
## [[2]]$A
## [1] 0.2108643 0.2355912
##
##
## [[3]]
## [[3]]$tm
## [1] 1.320555
##
## [[3]]$x
## [1] 3
##

```

```
## [[3]]$A
## [1] 0.2108643 0.2355912 0.2366910
##
##
## [[4]]
## [[4]]$tm
## [1] 1.520057
##
## [[4]]$x
## [1] 4
##
## [[4]]$A
## [1] 0.2108643 0.2355912 0.2366910 0.2136786
##
##
## [[5]]
## [[5]]$tm
## [1] 1.524415
##
## [[5]]$x
## [1] 5
##
## [[5]]$A
## [1] 0.2108643 0.2355912 0.2366910 0.2136786
## [5] 0.2758153
##
##
## [[6]]
## [[6]]$tm
## [1] 1.569685
##
## [[6]]$x
## [1] 6
##
## [[6]]$A
## [1] 0.2108643 0.2355912 0.2366910 0.2136786
## [5] 0.2758153 0.2699149
##
##
## [[7]]
## [[7]]$tm
## [1] 1.610068
##
## [[7]]$x
```

```

## [1] 7
##
## [[7]]$A
## [1] 0.2108643 0.2355912 0.2366910 0.2136786
## [5] 0.2758153 0.2699149 0.2400447
##
##
## [[8]]
## [[8]]$tm
## [1] 1.739818
##
## [[8]]$x
## [1] 6
##
## [[8]]$A
## [1] 0.2108643 0.2366910 0.2136786 0.2758153
## [5] 0.2699149 0.2400447
##
##
## [[9]]
## [[9]]$tm
## [1] 1.786115
##
## [[9]]$x
## [1] 7
##
## [[9]]$A
## [1] 0.2108643 0.2366910 0.2136786 0.2758153
## [5] 0.2699149 0.2400447 0.2729063
##
##
## [[10]]
## [[10]]$tm
## [1] 1.865547
##
## [[10]]$x
## [1] 8
##
## [[10]]$A
## [1] 0.2108643 0.2366910 0.2136786 0.2758153
## [5] 0.2699149 0.2400447 0.2729063 0.2432511
##
##
## [[11]]

```

```

## [[11]]$tm
## [1] 1.949677
##
## [[11]]$x
## [1] 9
##
## [[11]]$A
## [1] 0.2108643 0.2366910 0.2136786 0.2758153
## [5] 0.2699149 0.2400447 0.2729063 0.2432511
## [9] 0.2427908
##
##
## [[12]]
## [[12]]$tm
## [1] 1.99815
##
## [[12]]$x
## [1] 10
##
## [[12]]$A
## [1] 0.2108643 0.2366910 0.2136786 0.2758153
## [5] 0.2699149 0.2400447 0.2729063 0.2432511
## [9] 0.2427908 0.1758324
##
##
## [[13]]
## [[13]]$tm
## [1] 2.124588
##
## [[13]]$x
## [1] 11
##
## [[13]]$A
## [1] 0.2108643 0.2366910 0.2136786 0.2758153
## [5] 0.2699149 0.2400447 0.2729063 0.2432511
## [9] 0.2427908 0.1758324 0.2370110
##
##
## [[14]]
## [[14]]$tm
## [1] 2.142101
##
## [[14]]$x
## [1] 12

```

```
##
## [[14]]$A
## [1] 0.2108643 0.2366910 0.2136786 0.2758153
## [5] 0.2699149 0.2400447 0.2729063 0.2432511
## [9] 0.2427908 0.1758324 0.2370110 0.2635687
##
##
## [[15]]
## [[15]]$tm
## [1] 2.202867
##
## [[15]]$x
## [1] 13
##
## [[15]]$A
## [1] 0.2108643 0.2366910 0.2136786 0.2758153
## [5] 0.2699149 0.2400447 0.2729063 0.2432511
## [9] 0.2427908 0.1758324 0.2370110 0.2635687
## [13] 0.2066377
```


That is

```
it = length(dat)
tree <- data.frame(time = it, x = 2)
for (i in 1:it) {
  tree[i, ] = c(t = dat[[i]]$tm, x = dat[[i]]$x)
}
library(xtable)
options(xtable.comment = FALSE)
options(xtable.booktabs = TRUE)
xtable(tree, caption = "Table with simulated waiting times and number of species")
```

	time	x
1	0.00	1.00
2	0.35	2.00
3	1.32	3.00
4	1.52	4.00
5	1.52	5.00
6	1.57	6.00
7	1.61	7.00
8	1.74	6.00
9	1.79	7.00
10	1.87	8.00
11	1.95	9.00
12	2.00	10.00
13	2.12	11.00
14	2.14	12.00
15	2.20	13.00

Table 1: Table with simulated waiting times and number of species

We try an MLE calculation maximizing the Likelihood function, but for different initial values the estimated parameters are very different and most of them does not make any sense.

```
fn <- function(theta) {
  beta = 0
  delta = 0
  for (i in 1:nT) {
    beta[i] = theta[1] + sum(theta[2] * AA[[i]])
    delta[i] = theta[3] + sum(theta[4] * AA[[i]])
  }
  sum(log(beta + delta) + log(x) - (beta + delta) *
    x * Tm + D * (log(beta) - log(beta + delta)) +
    B * (log(delta) - log(beta + delta)))
}
```

```
nlm(fn, theta <- c(0.5, 0.1, 0.7, 0.8), hessian = TRUE)
```

```
## $minimum
## [1] -362635.6
##
## $estimate
## [1] 2861.808 3155.901 2855.841 3147.169
##
## $gradient
## [1] -28.63700 -31.58295 -28.63543 -31.58056
##
## $hessian
##           [,1]      [,2]
## [1,] -3.908965e-08 -3.093552e-08
## [2,] -3.093552e-08 -3.915688e-08
## [3,] -1.994177e-08 -3.875021e-09
## [4,] -3.877670e-09 -5.860527e-10
##           [,3]      [,4]
## [1,] -1.994177e-08 -3.877670e-09
## [2,] -3.875021e-09 -5.860527e-10
## [3,] -2.697764e-07 -3.179857e-07
## [4,] -3.179857e-07 -5.359631e-07
##
## $code
## [1] 4
##
## $iterations
## [1] 100
```