Master Mathématiques pour les Sciences du Vivant
Rapport de Stage

Responsable du Master : Christophe GIRAUD

Shifted stochastic processes evolving on trees : application to models of adaptive evolution on phylogenies

Paul BASTIDE

Tuteurs de Stage :

Stéphane ROBIN

UMR 518 MIA - AgroParisTech/INRA
AgroParisTech
16, rue Claude Bernard
75231 Paris Cedex 05

Mahendra MARIADASSOU

UR 1077 MIG - INRA
Bâtiment 233
Domaine de Vilvert
78352 Jouy en Josas Cedex

À Paris, le 23 Août 2014

Remerciements

Je voudrais tout d'abord remercier mes deux tuteurs, Mahendra Mariadassou et Stéphane Robin, pour leur soutien et leur investissement sans failles, leurs conseils de qualité, leur écoute attentive, et pour la confiance qu'ils m'ont accordés pour ce stage, ouvrant sur une thèse future.

Je remercie également Christophe Giraud, qui a su m'aiguiller par ses conseils éclairés, pour son aide précieuse tout au long de l'année.

Je remercie aussi toute l'équipe du laboratoire MIA de l'AgroParisTech, dont l'ambiance détendue mais laborieuse me fut propice durant ces cinq mois de stage.

J'aimerais aussi dire merci à l'équipe du laboratoire MIG à Jouy en Josas, pour leur accueil toujours chaleureux lors de mes visites hebdomadaires.

Enfin, je remercie la plateforme de bioinformatiques INRA MIGALE (http://migale.jouy.inra.fr) pour les ressources de calculs qu'elle m'a permis d'utiliser.

Summary

Le projet s'inscrit dans la dynamique de modélisation statistique qui s'opère aujourd'hui dans le champ de l'écologie comparative. Les différents traits quantitatifs d'un jeu d'espèces peuvent être vus comme le résultat d'un processus stochastique courant le long d'un arbre phylogénétique. Cette modélisation permet de prendre en compte des corrélations entre espèces issues d'une histoire évolutive commune. Le processus stochastique choisi permet de capturer les mécanismes qui gouvernent l'évolution d'un trait. Les écologues préfèrent ainsi le processus d'Orstein-Uhlenbeck (OU) au Mouvement Brownien (BM), plus simple mais moins réaliste. Le processus OU modélise la sélection naturelle s'opérant sur un trait par un mécanisme de rappel vers une valeur centrale, interprétée comme la valeur optimale du dit trait dans un environnement donné. Retracer l'historique des sauts de cette valeur centrale revient alors à repérer les changement de niche évolutive pour chaque lignée. Tous les descendants d'une espèce ancestrale ayant subi une évolution adaptative héritent de cette innovation. On peut donc définir de manière naturelle une classification des espèces. Les groupes fonctionnels ainsi formés partagent alors une même valeur optimale de trait, et sont cohérents avec la phylogénie des espèces considérées. À partir de mesures d'un trait sur différentes espèces et de l'arbre phylogénétique de ces espèces, on se propose de construire, d'étudier, et d'implémenter efficacement un modèle à données incomplètes permettant d'inférer la valeur des différents paramètres du processus, via la détection automatique des sauts. La définition du modèle présentant des problèmes d'identifiabilité, il s'agit également, d'une part, de restreindre l'espace des solutions aux allocations parcimonieuses des sauts afin d'éviter une sur-paramétrisation du modèle, et, d'autre part, de dénombrer les classes de solutions équivalentes, en terme de classification. En vue de préparer une future procédure de sélection de modèle, un calcul de la taille de l'espace des solutions à nombre de ruptures données est également proposé. Le modèle développé est ensuite appliqué à des jeux de donnés simulés sur un arbre phylogénétique réel représentant la famille des mammifères.

This project takes a step further in the process of systematic statistical modeling currently occurring in the field of comparative ecology. Quantitative traits measured on related species can be seen as the result of a stochastic process running on a phylogenetic tree. This modeling can account for correlations between species that have a shared evolutionary history. The chosen process must be able to capture the mechanisms of a trait evolution. Ecologists hence prefer the Orstein-Uhlenbeck (OU) process to the simpler but less realistic Brownian Motion (BM). This OU process has a tendency to revert to a central value, interpreted in ecology as the optimal value of a trait in a given environment. It can hence model natural selection on a functional trait. For any lineage, a shift in this central value then represents a change of evolutionary niche. As the descendants of an ancestral species inherit any adaptive evolution it went through, this model provides us with a natural way of defining a clustering of species based on unobserved evolutionary niches. Species among such groups then share the same optimal value for the trait, and are phylogenetically coherent. For this project, given measures of a trait on related species and a phylogenetic tree of those species, we aim at building, studying, and efficiently implementing an incomplete-data model that allows us to infer parameters of the stochastic process, via an automatic detection and characterization of the shifts. As some identifiability issues arise, it is necessary to, first, limit ourselves to parsimonious reconstructions of the shifts to avoid over-parametrisation, and, second, to count the number of equivalent solutions for a given clustering. As a first step toward a model selection procedure, we also computed the cardinal of the space of solutions for a given number of shifts. We then apply this model to functional traits simulated on a well established mammal phylogenetic tree.

Contents

Introduction

Comparative biology is a multidisciplinary field that uses the natural variations observed among organisms to detect patterns and understand mechanisms that produced them, at their own scales. We are interested here in the variations of a quantitative trait across related species in response to environmental changes. Specifically, we focus on functional traits that reflect the fitness of their bearer. These traits are under the influence of a selective pressure from the bearer's environment and should therefore adaptively respond to changes. The distribution of functional traits across species hence contains the footprint of adaptive events and should in principle allow us to detect unobserved past events. The observed variations have however at least two components. First, some ancestral species undergo some changes in their ecological niches, that lead to an evolution of the functional trait of interest. These species then pass their innovations on to their descendants. One component of the variations is therefore the a priori unknown structure inherited from the multiple adaptive events. Second, functional traits inherently fluctuate in time, which leads to correlations between species according to their phylogenetic relations. Indeed, related species share a common evolutionary history: the traits of two closely related species had less time to fluctuate away than those of distant species and should therefore be more similar. The second component of variation then stems from the neutral fluctuations of the trait compounded by the evolutionary history of the species. In order to accurately tease the two components apart, we need a quantitative model for the evolution of functional traits across related species. One common way to represent the evolutionary history of species is to use phylogenetic trees. We therefore consider evolution models of a trait on a tree, that we take to be given. As a side-effect, reconstructing adaptive events on a tree naturally creates groups of species that are both environmentally and phylogenetically coherent.

A natural model is to assume that the trait evolves across time according to a stochastic process. The simplest such process for quantitative traits is the Brownian Motion. Brownian Motions on trees have been extensively studied, starting from the seminal work of [Fel85]. A lot of effort has been devoted to extending of the model: replacing the Brownian Motion by a Orstein-Uhlenbeck process [BK04], allowing parameters of the Orstein-Uhlenbeck process to vary across times to reflect adaptive evolution [BK04] and/or different evolution regimes [BJBO12]. Many questions however remain open from the statistical point of view, in particular concerning our ability to estimate some parameters. Ané and Ho [Ané08] [HA13b] even proved that the estimators of some parameters, hard to infer in practice, are asymptotically inconsistent. The inconsistency arises from the underlying tree structure of the evolution model. This same structure is also responsible for identifiability issues, as first pointed out in [HA14].

The present report is organized as follows. In Chapter 1, we introduce some notations and define a precise statistical framework. We show how a stochastic process running forward on a phylogenetic tree and spanning independent copies at each node is a natural framework for our problem. In particular, it enforces higher correlations between trait values of close species compared with distant ones. Adaptive events can be introduced as shifts in the values of some parameters across the tree and lead to a phylogenetic-wise clustering of the species according to their mean trait value. We present the model as a latent variables model and detail it for two particular stochastic processes : the Brownian Motion, that represents a pure drift, and the

Orstein-Uhlenbeck process, better suited at modeling adaptive evolution and shifts of ecological niches.

In Chapter 2, we reframe our model as a linear regression problem and exhibit some combinatorial properties of the model. This reframing allows us to exhibit systematic identifiability issues of the model. Identifiability issues have a natural interpretation in terms of shifts assignment on the tree. Using an "infinite site model" assumption and parsimony assumptions, we define an equivalence relation on shifts assignments and show that the evolution model is identifiable for equivalence classes but not for shifts assignments. We then present two counting algorithms: one for the cardinal of any equivalence class and the other for the number of equivalence classes.

Finally, in Chapter 3, we propose an Expectation Maximization algorithm for parameter inference. We first present the operational implementation of the procedure and discuss some computational issues related to algorithmic efficiency and initialization. We then show the results of a simulation study and discuss the limitations of the current implementation.

Chapter 1

Modeling

1.1 Phylogenetic trees and Comparative Ecology

1.1.1 Phylogenetic Trees

When considering a set of contemporary species, one has to take into account their "genealogical" correlations, that introduce some structures in the data observed. Phylogenetic trees are used to describe the evolutionary relationships between nowadays species. They are at the root of modern classification of species in monophyletic clades. Some of these trees begin to be quite well known, thanks to reconstructions involving DNA sequencing paired with models of DNA evolution (see [Fel04], chapter 13 for an introduction). For instance, a quite complete version of the Mammals phylogeny, synthesizing several previous studies, was published in 2011 (see figure 1.1, issued from [M+11]). For this tree, all the polytomies are resolved, and the tree is binary, with 169 current species and 168 nodes, spanning on around 400 millions years (first known mammals fossils are aged of around 220 million years). The tree is also calibrated in time, so that branch lengths represent time instead of evolutionary quantities, and the tree is ultrametric, that is, in that case, that all tips are contemporary. In all applications below, we will use this fixed tree topology for simulations.

1.1.2 Stochastic Processes On Trees

In Comparative Ecology, one cannot see the species studied as independent, and must consider the correlations introduced by their shared evolutionary history in consideration. This issue was raised in particular in 1985 by Joseph Felsenstein ([Fel85]), who introduced the method of *independent contrasts* to deal with the problem. We will not explain this particular method here, but concentrate on the underlying model, which is nowadays widely used.

In order to take the phylogenetic tree into account in a quantitative way, we will use the following model, based on these two assumptions:

1. On a given branch of the tree, the functional trait considered evolves along time according to a one dimensional stochastic process, such as a Brownian Motion (BM), or an Orstein-Uhlenbeck (OU) process, as developed further.

2. The tree topology is fixed, and speciation events create on the daughters branches two identical and independent copies of the stochastic process running on the parental branch, that have the same starting point.

This principle is illustrated with a BM evolving on a simple tree figure 1.2. In this model, interactions between species are not taken into consideration, and the interaction of species with their environment can be included in the stochastic process used, as we will see further. This model definition allows us to compute easily the correlations between species, that is

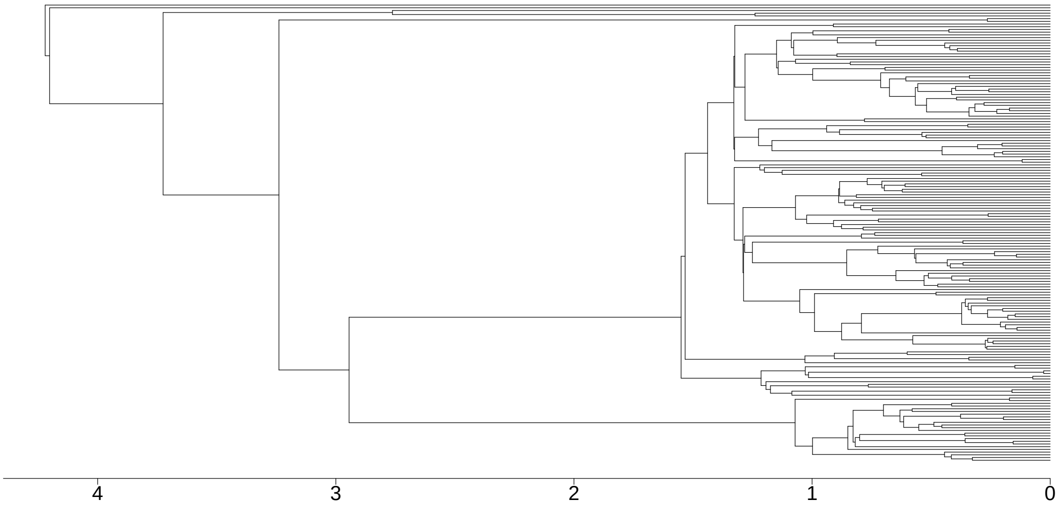
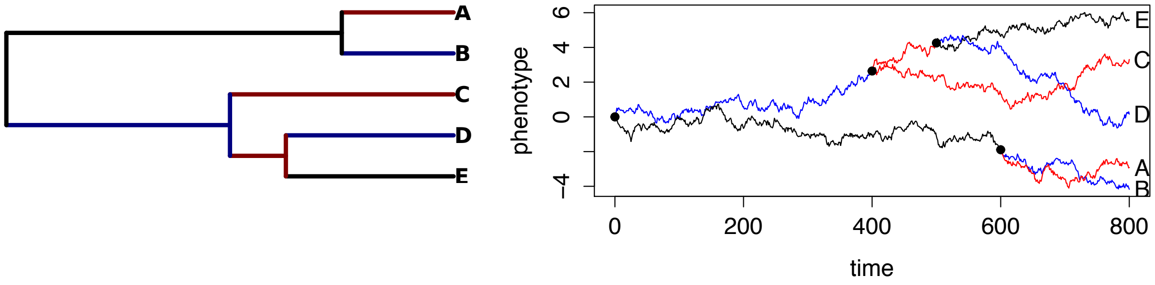


Figure 1.1: Phylogenetic tree of 169 Mammals taxa issued from [M+11] (Cetacea Constraints, Soft-bounded, Autocorrelated Rates ; Supporting Online Material, Table S5, page 140) as drawn by package ape in R. One time unit corresponds to 100 million years.

directly dependent of their time of shared evolutionary history. One can see this as a mixed effect model, where the correlation matrix of the observations depends explicitly on a tree.



(a) A phylogenetic tree with five lineages.

(b) A representation of the stochastic process modeling the evolution of a quantitative trait according to a Brownaian Motion.

Figure 1.2: Illustration of the model of evolution according to a stochastic process along a tree. For each speciation event on the tree on the left, the process on the right is split in two independent BMs.

1.1.3 Unsupervised Clustering of Species

In this document, we only consider the evolution of a single trait along time. The question we consider is then the following: given observations of a trait at the tips of a tree (i.e. for current species), how can we split them into clusters that would be coherent with both their respective trait values and their phylogenetic relationships? For well defined species, such as mammals, answering such a question could help restore some of the great events that shaped the modern repartition of one trait among the clade. For ill defined species, such as some groups of bacterias, such a question could lead to a functional definition of species, based on the phylogeny and on some character of interest for the classifier, such as traits linked with toxicity in a water bacteria community.

In order to do this clustering, we introduce some *shifts* in the stochastic processes described above. At some point in time, for a given living species, we assume that the environment can change brutally, introducing a shift in the parameters of the stochastic process. All the descendants of this species will then evolve according to new parameters, and, therefore, the distribution of the trait values among the descendant of this species will be different from all the other species, forming a differentiated group. The problem of the clustering of the tips will then be studied as a problem of allocation of shifts on the phylogenetic tree. This new parametrization allows us to use the previous model and to have an historic interpretation for the clusters defined. Unfortunately, it introduces some identifiability issues, as seen in chapter 2.

1.2 Latent Variables Model

In the following, we will adopt a description of the model as a latent variables model. We assume that we only have access to the values of a trait at the tips of a given fixed tree, and that these observations depend on the unobserved states at the internal nodes of the tree, according to a stochastic process based model yet to be specified. The problem is then to fit this model to our data, and to use it to define "phylogenetically coherent" clusters among the tips. More specifically, we will use the notations described below to specify our model (see also figure 1.3).

Notations
Tree:

- $\mathcal{T}$ is the fixed tree considered. It has m internal nodes, and n tips. The nodes are numbered from 1 (the root of the tree) to m, and the tips from $m+1$ to $m+n$. Note that if the tree is binary, then $m = n - 1$.
- $i' = m + i$ is the number in the tree $\mathcal{T}$ of the tip i, $i \in [\![1\,, n]\!]$.
- $\text{pa}(j)$ denotes the (unique) parent node of node j: $\text{pa}(j) = \{i : (i \to j) \in \mathcal{T}\}$.
- $\text{Par}(i) = \{\text{pa}^r(i) : r \geq 0\}$ is the ensemble composed of node i and all its ancestors. We denote here by pa^r the composition r times of the function pa.
- b_j is the branch from node j to its parents, of length ℓ_j.
- $\text{ca}(i, j)$ is the most recent common ancestor (mrca) of nodes i and j.
- t_j is the depth of node j in the tree, i.e. the distance from the root to node j.
- $t_{ij} = t_{\text{ca}(i,j)}$ is the time of shared ancestry between nodes i and j.
- $d_{ij} = t_i + t_j - 2t_{ij}$ is the phylogenetic distance between nodes i and j.
- If the tree $\mathcal{T}$ is *ultrametric*, we will write t_{tree} the common age of all tips: $\forall i \in [\![1\,, n]\!]$, $t_{\text{tree}} \; = \; t_{i'}$

Variables:

- $Y = (Y_1, \dots, Y_n)$ is the observed dataset of n values of a quantitative traits at the tips of the tree $\mathcal{T}$.
- $Z = (Z_1, \dots, Z_m)$ is the unobserved dataset of m values of a quantitative traits at the internal nodes of tree $\mathcal{T}$.

- $X = (Z, Y)$ is the complete dataset, with $\begin{cases} X_j = Z_j & \forall j \in [\![1\,, m]\!] \\ X_{i'} = X_{m+i} = Y_i & \forall i \in [\![1\,, n]\!] \end{cases}$.

Process Path:

- $W_i(t)$ is the value of the trait on lineage $i \in [\![1\,, n]\!]$ at time $t \in [0\,, t_{i'}]$.
- We have $\begin{cases} W_i(t_{i'}) = Y_i & \forall i \in [\![1\,, n]\!] \\ W_i(t_j) = Z_j & \forall j \in \mathrm{Par}(i'), j \neq i' \end{cases}$
- For two tips $(i, j) \in [\![1\,, n]\!]^2$, thanks to the tree structure, we have: $\forall t \leq t_{ij},\ W_i(t) = W_j(t)$.

Change points:

- K will be the number of change points allowed in the tree.
- $\tau = (\tau_1, \dots, \tau_K)$ is the vector of positions of the shifts on the tree. For any $k \in [\![1\,, K]\!]$, τ_k denotes the branch where the change point occurs: there is one $j_k \in [\![1\,, n+m]\!]$ such that $\tau_k = b_{j_k}$.
- $\nu = (\nu_1, \dots, \nu_K)$ is the vector of relative positions of the shifts on the branches. If a change point k occurs on branch b_{j_k}, its exact position on the branch is at the fraction ν_k of its length. For instance, if it is at the beginning of the branch, right after a speciation, $\nu_k = 0$, and if it is at the end of it, right before a speciation, $\nu_k = 1$.
- δ_k is the value of the shift in the mean occurring at change-point k.
- We will assume in the rest of this document that at most one change point occurs on each branch.

With these notations, a change point k occurring on branch $\tau_k = b_{j_k}$ occurs on absolute time (from the root) $t_k = t_{\mathrm{pa}(j_k)} + \nu_k \ell_{j_k}$.

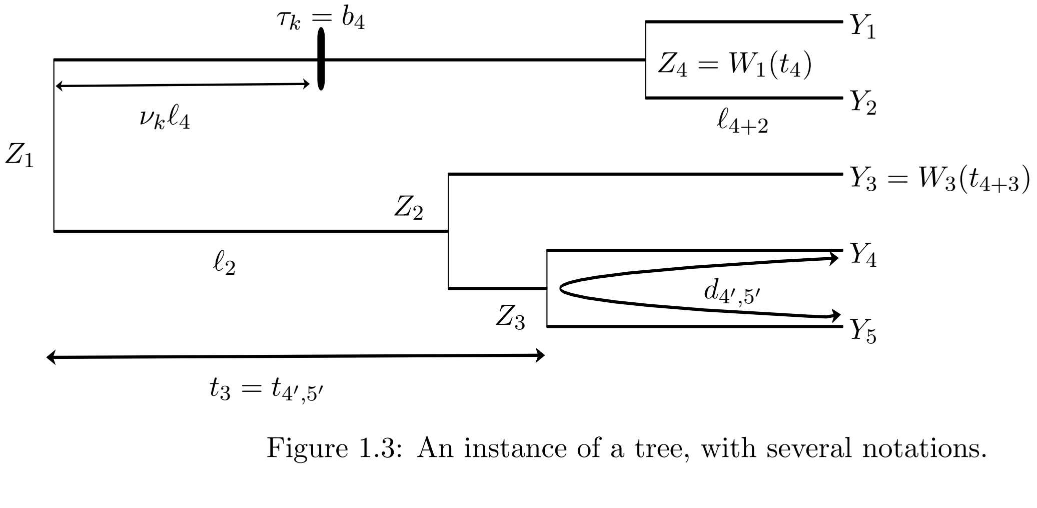


Figure 1.3: An instance of a tree, with several notations.

1.3 The Brownian Motion

1.3.1 Brownian Motion with Shifts

We assume here that the trait evolves on the tree $\mathcal{T}$ according to a shifted BM:

$$\forall i \in [\![1\,, n]\!],\ \forall t \in [0\,, t_{i'}], \quad dW_i(t) = \sigma dB_i(t)$$

where the structure of dependence of the tree is taken into account through the following correlations:

$$\forall (i,j) \in [\![1\,,n]\!]^2, \quad \mathbb{C}\text{ov}\left[dB_i(t); dB_j(t)\right] = \rho_{ij}(t)dt, \qquad \rho_{ij}(t) = \begin{cases} 1 & \text{if } t \leq t_{ij} \\ 0 & \text{else} \end{cases} \tag{1.1}$$

This models a neutral evolution of the trait along the tree, and the effect of the environment is only taken into account through the shifts occurring in the trait values. These shifts are instantaneous, which can be justified by the separation of evolutionary and ecological time scales. The model can be recursively defined as follow:

$$\begin{cases} X_1 \sim \mathcal{N}(\mu, \gamma^2) \\ X_j | X_{\text{pa}(j)} \sim \mathcal{N}\left(X_{\text{pa}(j)} + \sum_k \mathbb{I}\{\tau_k = b_j\}\delta_k,\ \ell_j\sigma^2\right) \quad \forall j \in [\![2\,,m+n]\!] \end{cases} \tag{1.2}$$

The joint distribution of X is then Gaussian: $X \sim \mathcal{N}(\mathbf{m}, \Sigma)$, with the mean $\mathbf{m}_i$ of a node given by:

$$\mathbf{m}_i = \mu + \sum_{j \in \text{Par}(i)} \sum_{k=1}^{K} \mathbb{I}\{\tau_k = b_j\}\delta_k \quad \forall i \in [\![1\,,m+n]\!] \tag{1.3}$$

and the covariance σ_{ij} between two nodes:

$$\sigma_{ij} = \gamma^2 + t_{\text{ca}(i,j)}\sigma^2 \quad \forall (i,j) \in [\![1\,,m+n]\!]^2 \tag{1.4}$$

Note that $\mathbf{m}_i$ is simply the root mean value shifted by all the shifts occurring along the path on the tree from the root to node i.

1.3.2 Likelihood of the Completed Dataset

From the description above, one can directly get the likelihood of the data Y, that is supposed to follow a n-dimensional Gaussian with first and second order moments depending on the parameters of the model. But, as seen in equation (1.3), some discrete parameters are involved in the expression of the mean, and therefore the likelihood obtained is hard to maximize in the parameters of the model. To get around this issue, we will need to work with the likelihood of the completed dataset X, as explain further in chapter 3.

By decomposing the likelihood according to the graphic model induced by the tree, we get the following expression for the likelihood of the completed dataset:

$$\begin{aligned} p_\theta(X) &= p_\theta(Z)p_\theta(Y|Z) \\ &= p_\theta(Z_1) \prod_{1<j\leq m} p_\theta(Z_j|Z_{\text{pa}(j)}) \prod_{1\leq i\leq n} p_\theta(Y_i|Z_{\text{pa}(i')}) \\ &= \phi\left(\frac{Z_1-\mu}{\gamma^2}\right) \prod_{1<j\leq m} \phi\left(\frac{Z_j - Z_{\text{pa}(j)}}{\ell_j\sigma^2}\right) \prod_{1\leq i\leq n} \phi\left(\frac{Y_i - Z_{\text{pa}(i')}}{\ell_{i'}\sigma^2}\right) \end{aligned}$$

where ϕ denotes the standard Gaussian density, and θ the vector of parameters of the model. We can then express the log-likelihood as follow:

$$\begin{aligned} \log p_\theta(X) = &-\frac{m+n}{2}\log(2\pi) - \frac{1}{2}\log\gamma^2 - \frac{1}{2\gamma^2}\left(Z_1-\mu\right)^2 \\ &- \sum_{j=2}^{m+n}\log(\ell_j) - \frac{m+n-1}{2}\log\sigma^2 - \frac{1}{2\sigma^2}\sum_{j=2}^{m+n}\ell_j^{-1}\left(X_j - X_{\text{pa}(j)} - \sum_k \mathbb{I}\{\tau_k = b_j\}\delta_k\right)^2 \end{aligned} \tag{1.5}$$

1.4 The Orstein-Uhlenbeck Process

1.4.1 The Orstein-Uhlenbeck Process

We want to relax the hypothesis of neutral evolution, and to take into account the interactions of the species with their environments through adaptive evolution. We make the hypothesis that, at a given time and in a given environment, one value of the trait studied is "optimal", that is, individuals with a trait value close to this optimum are more likely to survive, and have descendants, than the others. This would lead to a directive evolution, where one value of a trait is more likely than others. As in [BK04], we will model this with an OU process, that is solution of the stochastic differential equation given below:

$$dW(t) = \alpha[\beta(t) - W(t)]dt + \sigma dB(t) \tag{1.6}$$

where $B(t)$ is the BM and σ^2 the variance of the process. We call α the *selection strength*, it is the parameter that controls the strength of the restoring force that tends to make the process go to its optimal value $\beta(t)$. This optimal value can change during the evolutionary process, and we will consider that it is piecewise constant, with some shifts occurring on the phylogeny. A shift in the optimal value can account for a change of ecological niche of a species at one moment of its history, that will affect all its descendants.

In addition to its ability to model adaptive evolution, the OU process has some nice features when compared to the BM. In particular, while the Brownian motion has a variance that grows linearly with time and no stationary state, the OU has a variance bounded by $\frac{\sigma^2}{2\alpha}$ (see expressions (1.9) and (1.10) below), and a stationary state of mean $\beta(t)$ and variance $\frac{\sigma^2}{2\alpha}$.

1.4.2 Orstein-Uhlenbeck on a Tree with Shifts

On a tree $\mathcal{T}$ with n tips, we can write an OU process for each lineage:

$$\forall t \in [0\,, t_{i'}], \quad dW_i(t) = \alpha[\beta_i(t) - W_i(t)]dt + \sigma dB_i(t)$$

where the structure of dependence of the tree is taken into account as previously, through equation (1.1), and the piecewise constant evolution of the environment-dependent optimal value is given by:

$$\begin{cases} \beta_i(t=0) = \beta_0 & \forall i \in [\![1\,, n]\!] \\ \beta_i(t_j) = \beta_i(t_{\text{pa}(j)}) + \sum_k \mathbb{I}\{\tau_k = b_j\}\delta_k & \forall j \in \text{Par}(i'), j > 1,\ i \in [\![1\,, n]\!] \end{cases}$$

More precisely, for a lineage i, the optimal value evolves according to the function:

$$\forall t \leq t_{i'}, \quad \beta_i(t) = \beta_0 + \sum_{j \in \text{Par}(i')} \sum_k \mathbb{I}\{\tau_k = b_j, \nu_k \ell_j \leq t - t_{\text{pa}(j)}\}\delta_k$$

Note that for any two lineages $i, j \in [\![1\,, n]\!]$, and any time $t \leq t_{ij}$, we have $\beta_i(t) = \beta_j(t)$. Thus, the optimal value is uniquely defined at a given node. Thereafter, for a node $j \in [\![1\,, m+n]\!]$, we denote $\beta^j = \beta_i(t_j)$ for any descendant tip $i \in [\![1\,, n]\!]$ of j.

For a node $j \in [\![2\,, m+n]\!]$ in lineage $i \in [\![1\,, n]\!]$, we can express the value of a trait $X_j = W_i(t_j)$

according to its value at the parent node $X_{\mathrm{pa}(j)} = W_i(t_{\mathrm{pa}(j)})$:

$$\begin{aligned}
W_i(t_j) &= W_i(t_{\mathrm{pa}(j)})e^{-\alpha(t_j - t_{\mathrm{pa}(j)})} + \int_{t_{\mathrm{pa}(j)}}^{t_j} \alpha e^{\alpha(s-t_j)}\beta_i(s)ds + \int_{t_{\mathrm{pa}(j)}}^{t_j} \sigma e^{\alpha(s-t_j)} dB_i(s) \\
&= W_i(t_{\mathrm{pa}(j)})e^{-\alpha\ell_j} + \int_{t_{\mathrm{pa}(j)}}^{t_j} \alpha e^{\alpha(s-t_j)}\beta_i(t_{\mathrm{pa}(j)})ds + \int_{t_{\mathrm{pa}(j)}}^{t_j} \sigma e^{\alpha(s-t_j)} dB_i(s) \\
&\qquad + \sum_{k=1}^{K} \mathbb{I}\{\tau_k = b_j\} \int_{t_{\mathrm{pa}(j)}+\nu_k\ell_j}^{t_j} \alpha e^{\alpha(s-t_j)}\delta_k ds \\
&= W_i(t_{\mathrm{pa}(j)})e^{-\alpha\ell_j} + \beta^{\mathrm{pa}(j)}(1 - e^{-\alpha\ell_j}) + \sigma \int_{t_{\mathrm{pa}(j)}}^{t_j} e^{\alpha(s-t_j)} dB_i(s) \\
&\qquad + \sum_{k=1}^{K} \mathbb{I}\{\tau_k = b_j\}\delta_k \left(1 - e^{-\alpha(1-\nu_k)\ell_j}\right)
\end{aligned}$$

And, finally, we can express the model recursively as follow:

$$\begin{cases}
X_1 \sim \mathcal{N}(\mu, \gamma^2) \\
X_j | X_{\mathrm{pa}(j)} \sim \mathcal{N}\left(\begin{array}{l} X_{\mathrm{pa}(j)}e^{-\alpha\ell_j} + \beta^{\mathrm{pa}(j)}(1 - e^{-\alpha\ell_j}) \\ + \sum_{k=1}^{K} \mathbb{I}\{\tau_k = b_j\}\delta_k \left(1 - e^{-\alpha(1-\nu_k)\ell_j}\right), \; \frac{\sigma^2}{2\alpha}(1 - e^{-2\alpha\ell_j}) \end{array}\right) \quad \forall j \in [\![2\,, m+n]\!]
\end{cases} \tag{1.7}$$

We can see that the model not only depends on the branches on which the shifts occur, but also on the exact position in time of these shifts, through ν_k. For simplicity reasons, we will often make the assumption that all the shifts in the optimal value occurs right after a speciation: $\nu_k = 0$. This assumption simplifies a bit the expression of the model given above:

$$\begin{cases}
X_1 \sim \mathcal{N}(\mu, \gamma^2) \\
X_j | X_{\mathrm{pa}(j)} \sim \mathcal{N}\left(X_{\mathrm{pa}(j)}e^{-\alpha\ell_j} + \beta^{j}(1 - e^{-\alpha\ell_j}),\ \frac{\sigma^2}{2\alpha}(1 - e^{-2\alpha\ell_j})\right) \quad \forall j \in [\![2\,, m+n]\!]
\end{cases}$$

The joint distribution of X is then still Gaussian: $X \sim \mathcal{N}(\mathbf{m}, \Sigma)$, with the mean $\mathbf{m}_i$ of a node given by, for all $i \in [\![1\,, m+n]\!]$:

$$\mathbf{m}_i = \mu e^{-\alpha t_i} + \beta_0(1 - e^{-\alpha t_i}) + \sum_{j\in\mathrm{Par}(i)} \sum_{k=1}^{K} \mathbb{I}\{\tau_k = b_j\}\delta_k \left(1 - e^{-\alpha(t_i - t_{\mathrm{pa}(j)} - \nu_k\ell_j)}\right) \tag{1.8}$$

and the covariance σ_{ij} between two nodes:

$$\sigma_{ij} = \gamma^2 e^{-\alpha(t_i+t_j)} + \frac{\sigma^2}{2\alpha}\left(1 - e^{-2\alpha t_{\mathrm{ca}(i,j)}}\right)e^{-\alpha d_{ij}} \quad \forall (i,j) \in [\![1\,, m+n]\!]^2 \tag{1.9}$$

Note that if we take X_1 as the stationary distribution of the OU process: $X_1 \sim \mathcal{N}\left(\mu = \beta_0, \gamma^2 = \frac{\sigma^2}{2\alpha}\right)$, then, as $t_i + t_j = d_{ij} + 2t_{\mathrm{ca}(i,j)}$, we get:

$$\sigma_{ij} = \frac{\sigma^2}{2\alpha}e^{-\alpha d_{ij}} \quad \forall (i,j) \in [\![1\,, m+n]\!]^2 \tag{1.10}$$

Remark on the interpretation of the shift values. One can see in equation (1.8) that a shift at a position $\tau = b_j$ and of intensity δ in the optimal value $\beta(t)$ leads to a shift of "actualized" intensity $\left(1 - e^{-\alpha\Delta t}\right)\delta$ in the mean of the trait value at tip i, where $\Delta t = t_{i'} - t_{\mathrm{pa}(j)} - \nu_k\ell_j$ is the elapsed time since the shift. This means that the mean of the process *lags behind* the optimal value, and moves towards it with a speed determined by parameter α. So if α is small, a shift in the optimal value at a fixed position on a given tree will have a low impact on the mean of the process. That could be interpreted in the following way: if α is small, that means that the environment has only a small impact on the considered species, i.e. the species adapt their trait value to the optimum very slowly. Therefor, to have a significant impact on the mean of the value of the trait observed, a shift must either be very high in the tree, or be of large intensity. To briefly summarize that point, one could say that when α is low, only a "catastrophe" in the environment can impact living species.

Note that this behavior is very different from the BM modeling, where shifts are supposed to happen directly in the mean. This implies that, with our parametrization, the shifted BM can not be obtained as the limit when α goes to 0 of a shifted OU process.

1.4.3 Likelihood of the Completed Dataset

As for the BM in 1.3.2, we can easily compute the log-likelihood of the completed dataset by decomposing along the tree. Only the expressions of the mean and variance change, accordingly with the model defined above, and we get:

$$
\begin{aligned}
\log p_\theta(X) = & -\frac{m+n}{2}\log(2\pi) - \frac{1}{2}\log\gamma^2 - \frac{1}{2\gamma^2}\left(Z_1 - \mu\right)^2 \\
& - \sum_{j=2}^{m+n}\log\left(1 - e^{-2\alpha\ell_j}\right) - \frac{m+n-1}{2}\log\sigma^2 + \frac{m+n-1}{2}\log 2\alpha \\
& - \frac{\alpha}{\sigma^2}\sum_{j=2}^{m+n}\left(1 - e^{-2\alpha\ell_j}\right)^{-1}\Bigg(X_j - X_{\mathrm{pa}(j)}e^{-\alpha\ell_j} - \beta^{\mathrm{pa}(j)}\left(1 - e^{-\alpha\ell_j}\right) \\
& \qquad\qquad - \sum_{k=1}^{K}\mathbb{I}\{\tau_k = b_j\}\delta_k\left(1 - e^{-\alpha(1-\nu_k)\ell_j}\right)\Bigg)^2
\end{aligned}
\tag{1.11}
$$

Chapter 2

Identifiability Issues and Model Selection

2.1 Identifiability Issues

2.1.1 A simple Example

In the definition of the model above, the introduction of shifts in the stochastic process in order to account for clusters in the trait values of the observed species makes the model unidentifiable. One easy way to see the problem in the BM case is to assume that a shift occurs on all the branches directly stemming from the root. Then, the model would be unchanged shifting the root mean μ by a certain amount η, and all the shifts of the daughter branches by the opposite amount $-\eta$. The position, and even the number of shifts on the tree are not identifiable either. Looking only at a single binary node, figure 2.1 illustrate this problem. If the mean of the process is μ before the speciation, and $\mu + \delta_1$, $\mu + \delta_2$ on the two daughter branches, then we can see that there are at least five ways of placing one or two shifts to get this same distribution.

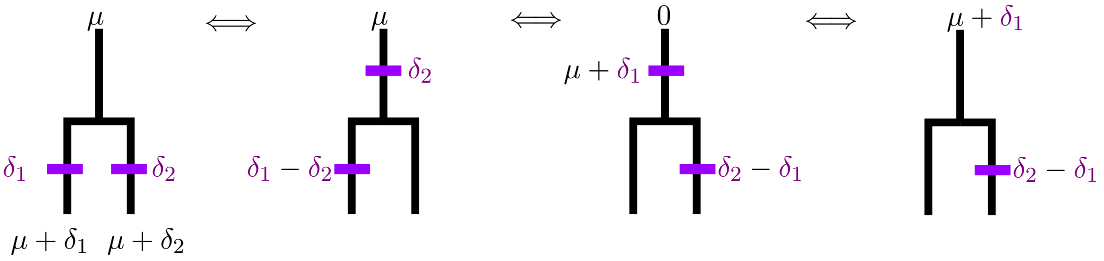


Figure 2.1: Some basic equivalencies for a binary node. These four allocations of shifts all produce the same output distribution.

For symmetry reasons, we can change a bit the parametrization, setting the root mean to 0, and imposing a shift δ_0 of a given value on the root branch b_1 (see equations below). With that in mind, let's consider the not-identifiable example presented figure 2.2. We can see that the fork composed of branches b_3, b_8 and b_9, at the bottom right of the tree (see dashed rectangle), has two shifts, so is not identifiable, according to the basic equivalencies seen above. But the shifts define four different clusters (defined as $\{Y_1\}$, $\{Y_2, Y_3\}$, $\{Y_4\}$, $\{Y_5\}$), that could not be defined with less than three shifts, so is minimal in a certain way. We will call such an allocation *parsimonious*, a class we will study in more details section 2.2.

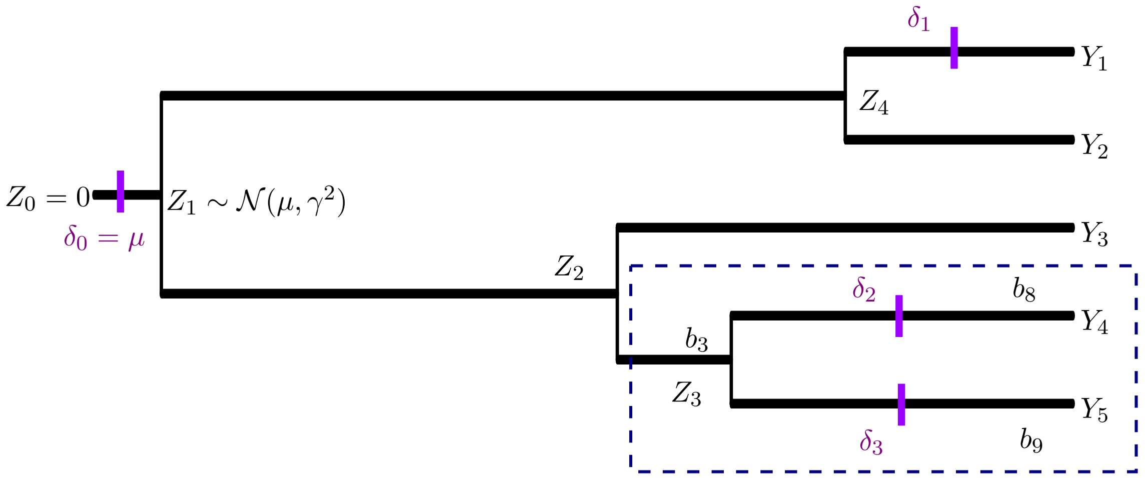


Figure 2.2: An instance of a shift allocation, that is not identifiable, but parsimonious (there are $K-1$ shifts for K clusters).

2.1.2 The Problem Seen as a Linear Model

In order to express the problem more specifically, we reformulate equations (1.2) and (1.7)) as a linear regression model:

$$Y = T\Delta + E \tag{2.1}$$

where:

- $E \sim \mathcal{N}(0, \Sigma_{YY})$ is the error vector.
- T (size $n \times m+n$) is the matrix representing the lineage structure of the tree: for a lineage $i \in [\![1\,, n]\!]$ and a branch $b \in [\![1\,, m+n]\!]$,

$$T_{ib} = \begin{cases} 1 & \text{if } b \text{ is in the } i^{th} \text{ lineage} \\ 0 & \text{else} \end{cases}$$

  This can be written: $T = [\mathbb{I}\{j \in \text{Par}(i')\}]_{\substack{1\leq i\leq n\\ 1\leq j\leq n+m}}$. Note that column j of T is exactly the indicator vector of the leaves stemming from branch j.

- Δ (size $(m+n)\times 1$) is the vector of the shifts on the $m+n-1$ branches of the tree, plus an intercept. It has $K+1$ non-zero entries at the K shifts chosen by the algorithm, plus one "imposed" at the root branch, $\Delta_1 = \delta_0$, corresponding to the intercept. It takes the following forms for the two stochastic models studied:

  BM: τ_0 is fixed on b_1, with $\delta_0 = \mu$, and:

$$\forall j \in [\![1\,, n+m]\!],\ \Delta_j = \begin{cases} \delta_k & \text{if } \tau_k = b_j \\ 0 & \text{otherwise.} \end{cases}$$

  OU, with Ultrametric Tree: τ_0 is fixed on b_1, with $\delta_0 = \frac{\mu}{e^{\alpha t_{\text{tree}}-1}} + \beta_0$, and $t_{\text{pa}(1)}$ and ν_0 formally set to 0 for the coherence of the notation, and:

$$\forall j \in [\![1\,, n+m]\!],\ \Delta(\alpha)_j = \begin{cases} \delta_k \left(1 - e^{-\alpha(t_{\text{tree}} - t_{\text{pa}(j)} - \nu_k \ell_j)}\right) & \text{if } \tau_k = b_j \\ 0 & \text{otherwise.} \end{cases}$$

  It is the vector of shifts on the optimal values "actualized" from the tips.

Expression of T and Δ in a simple example. To illustrate the above quantities, let's express them entirely in the case of a BM on the example tree presented above figure 2.2. In that case, we have:

$$T = \begin{array}{cc} & \begin{array}{cccc} 1 & 2 & 6 & 4 \end{array} \quad \begin{array}{ccccc} 5 & 6 & 7 & 8 & 9 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} & \left(\underbrace{\begin{array}{cccc} 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \end{array}}_{T_1} \quad \underbrace{\begin{array}{ccccc} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{array}}_{I_5} \right) \end{array} , \quad \Delta = \begin{pmatrix} \delta_0 \\ 0 \\ 0 \\ 0 \\ \delta_1 \\ 0 \\ 0 \\ \delta_2 \\ \delta_3 \end{pmatrix} \quad \text{and} \quad m_Y = T\Delta = \begin{pmatrix} \delta_0 + \delta_1 \\ \delta_0 \\ \delta_0 \\ \delta_0 + \delta_2 \\ \delta_0 + \delta_3 \end{pmatrix}$$

2.1.3 Linear Space of Unidentifiable Solutions

Now that we have an expression of the problem as a linear model, we can tackle the identifiability issues by looking for the kernel of the matrix T. We can easily find vectors in the kernel of T: for $b \in \llbracket 1\,, m \rrbracket$ a branch ending at an interior node, take the vector with 1 at the b^{th} entry, and -1 at all the direct children of b. That corresponds to a situation where the effect of a shift is immediately canceled by the opposite effect of shifts on its children branches. In fact, as there are m such independent vectors, and as the matrix T as a rank superior or equal to n (it contains the identity matrix) so that the dimension of the kernel is inferior or equal to m, this set of vectors $(k_1, \dots, k_m)$ is a basis of the kernel. In the example described above (figure 2.2), we get:

$$\mathrm{Ker}(T) = \mathrm{Span} \begin{pmatrix} 1 \\ -1 \\ 0 \\ -1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ -1 \\ 0 \\ 0 \\ 0 \\ -1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ -1 \\ -1 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ -1 \\ -1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Denote by $e = (e_1, \dots, e_{m+n})$ the canonical basis, and by $b = (k_1, \dots, k_m, e_{m+1}, \dots, e_{m+n})$ the basis adapted the decomposition $\ker(T) \oplus S$, where S is the supplementary space of $\ker(T)$, that has the set of vectors corresponding to the branches above the tips $(e_{m+1}, \dots, e_{m+n})$ as a natural basis. We then know the transfer matrix P from base e to base b, and we can express its inverse $P^{-1} = U$ in a very simple way as the matrix of occurrence of all nodes:

$$U_{ij} = \begin{cases} 1 & \text{if } j \in \mathrm{Par}(i) \\ 0 & \text{else} \end{cases} \quad \text{i.e.} \quad U = \begin{pmatrix} U_1 & 0 \\ T_1 & I_n \end{pmatrix}$$

where we made the decomposition $T = (T_1 \ I_n)$, with T_1 the bloc matrix of T (size $m \times n$) corresponding to the internal nodes of the tree, and I_n the identity matrix; and U_1 a $m \times m$ matrix.

If $\Delta = (\Delta_1, \dots, \Delta_{m+n})$ in the canonical basis e, denote by $\Delta' = (\Delta'_1, \dots, \Delta'_{m+n})$ the same vector in the decomposition base b. The identifiable part of the vector is then $\mathrm{Proj}_S(\Delta') = (0, \dots, 0, \Delta'_{m+1}, \dots, \Delta'_{m+n})$. The question of identifiability could then be expressed as follow: can we impose a set of constraints on the space of admissible parameters that would allow us to reconstruct unambiguously Δ from the n last components of Δ'? Finding such a set of constraints would allow us to automatically choose an unique representative of an equivalence class of solutions, in the sense defined below. This problem remains open, and a simple criteria is yet to be found. In the following section, we explore a bit more the set of parsimonious solutions, as parsimony is a natural constraint to consider.

2.2 Parsimony

2.2.1 Definition and Examples

We already introduced the concept of parsimony above (figure 2.2), and we can give a formal definition of the concept:

Definition. *A clustering of the tips being given, a parsimonious allocation of the shifts is such that it has a minimum number of shifts, while producing the given clustering.*

Assuming that we have an "infinite site model", i.e. that any new shift creates a new group, we get the following criterion:

Proposition 1. *A parsimonious allocation of K shifts on a tree creates exactly $K+1$ different clusters of the tips. Such a clustering is called* tree-compatible.

Proof. First, note that an allocation of K shifts cannot create more than $K+1$ clusters, but can possibly create less. In the case of an infinite site model where all the shifts produce a new value of the mean, the only way to create less than $K+1$ clusters is to "forget" one of the shifts, i.e. to put shifts on every descendant of the branch where it happens. Such an allocation is not parsimonious, as we could just add the value of the forgotten shift to all its descendent to get the same clustering of the tips with one shift less. So a parsimonious allocation cannot create less than $K+1$ clusters, and hence creates exactly $K+1$ clusters. □

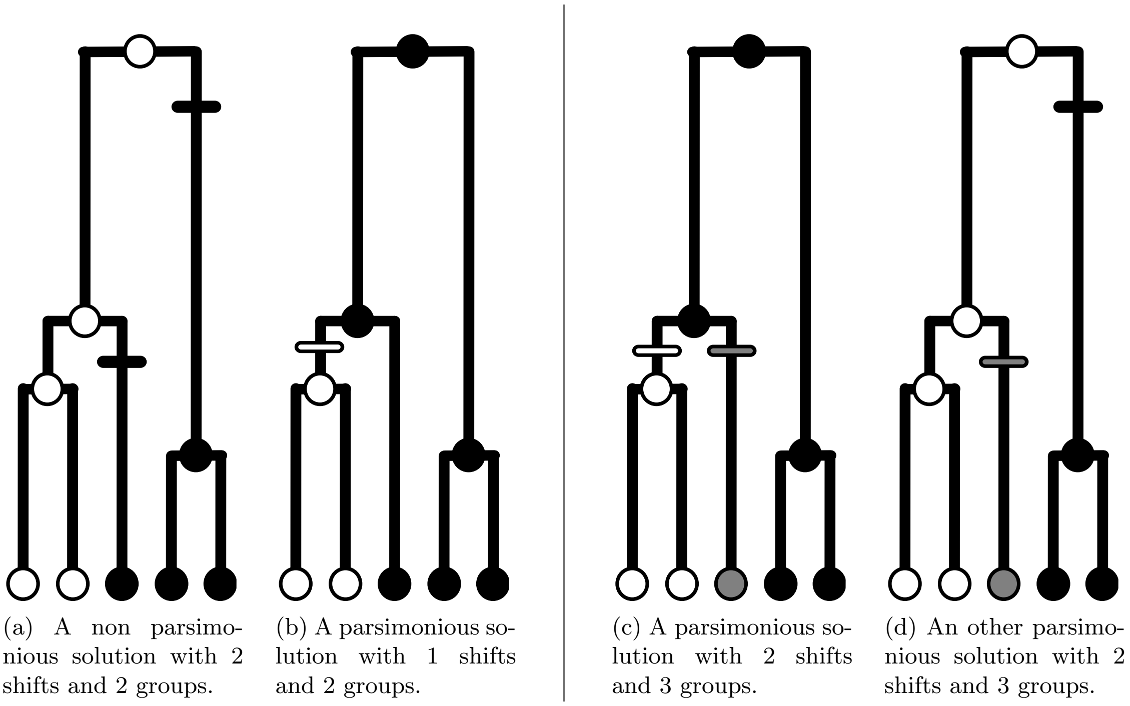
The assumption of an infinite site model means that two species can only be classified in the same group if they have exactly the same history of shifts, and that nature "do not repeat itself". In the exemple above (figure 2.2), that would mean that species 1 and 5 could not be grouped together, unless they both have the ancestral state, that is, unless no shift occurred in their respective histories.

We can illustrate this concept of parsimony by representing each group with a different color, as in figures 2.3. Figure 2.3a shows 2 groups and 2 shifts. It is not parsimonious, and do not respect the "infinite site model" rule, as opposed to figure 2.3b, that is parsimonious. Figures 2.3c and 2.3d are two different possible parsimonious allocations of shifts for a clustering in 3 groups, illustrating the non-uniqueness of a parsimonious solution.

2.2.2 Fitch and Sankoff Algorithms

Given a clustering of the tips, several algorithms already exist to find *one* parsimonious reconstruction of the ancestral states. Fitch and Sankoff algorithms, that are tailored to this problem, are described in [Fel04], chapter 2, and we will not present them in details here. We will just recall the principle of the Sankof algorithm, that is based on a *Dynamic Programing* (DP) approach.

To apply these algorithms to our problem, we must assume that the groups observed at the tips are the only possible states, and that there is a certain known cost to go from one group to another. Assume we have K groups, and denote by c_{ab} the cost of transition $a \to b$. Doing so, we have to relax our infinite site model assumption, as we can not forbid a priori reverse transitions. The algorithm then computes, from the tips to the root, the cost in term of transitions $S_j(a)$ for each node j to be in state a:



(a) A non parsimonious solution with 2 shifts and 2 groups.

(b) A parsimonious solution with 1 shifts and 2 groups.

(c) A parsimonious solution with 2 shifts and 3 groups.

(d) An other parsimonious solution with 2 shifts and 3 groups.

Figure 2.3: A tree with 5 tips and 2 ((a) and (b)) or 3 ((c) and (d)) groups of tips. The way one paints the ancestral states determine the number of shifts, and the parsimonious character of the model.

Initialization For every tip $i \in \llbracket 1\,, n \rrbracket$, every state $a \in \llbracket 1\,, K \rrbracket$,

$$S_{i'}(a) = \begin{cases} 0 & \text{if } j \text{ is in group } a \\ +\infty & \text{otherwise} \end{cases}$$

Upward For a node j with L daughter nodes $j_1, \dots, j_L$, we have:

$$\forall a \in \llbracket 1\,, K \rrbracket, \quad S_j(a) = \sum_{l=1}^{L} \min_{b_l} [c_{ab_l} + S_{j_l}(b_l)]$$

Root At the root 1, the vector S_1 gives the cost in term of transition to choose one particular group as the ancestral group, and one only need to choose one group that has the minimum value.

Downward To reconstruct a parsimonious solution, one needs to go down the tree, selecting at each node a state that realizes the minimum cost, and choosing the state that is identical to its parent's state when possible.

We can see that the algorithm provides us with one solution in time linear with the number of tips (the tree is only browsed twice). To get all the solutions, we would need to change the downward step in order to visit all possible ways, introducing a combinatorial complexity. In the next section, we will describe an algorithm that allows us to compute the *number* of parsimonious solutions, without enumerating them, in time linear with the number of tips.

2.2.3 Equivalence Classes

Definition. *Two allocations of shifts on the tree are said to be* equivalent *if they are both parsimonious, and define the same distribution at the tips.*

Note that it follows immediately from this definition that two equivalent allocations have the same number of shifts.

Our goal in this section is to find a recursive algorithm that would compute, a clustering of the tips being given, the number of equivalent allocations of shifts that can produce this clustering. We will use for this task a tailored Dynamic Programing algorithm, described below.

Suppose we have K distinct clusters among the n tips. These can be seen as discrete states, numbered from 1 to K. We will browse the tree upward, from the tips to the root, computing at each node the vector of size K describing the number of parsimonious allocations of the shifts in the subtree below the node, knowing the state of that node. For instance, in figure 2.4, p_1 is the number of parsimonious allocations for subtree below node p if node p is in state 1.

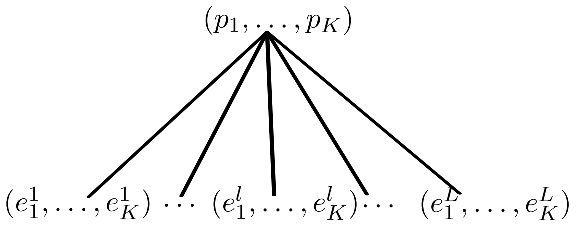

$(p_1, \dots, p_K)$
$(e_1^1, \dots, e_K^1)$ $\cdots$ $(e_1^l, \dots, e_K^l)$ $\cdots$ $(e_1^L, \dots, e_K^L)$

Figure 2.4: A parent node with L daughters

Initialization For every tip of the tree, define a vector of dimension K, with all entries to 0, except the one corresponding to the state of the tip, set to 1.

Propagation Take an internal node with L sub-trees below it, with known vectors $(e_1^l, \dots, e_K^l)$ ($l \in [\![1\,, L]\!]$). Denote by $p = (p_1, \dots, p_K)$ the vector associated to this node (see figure 2.4). Define:

$$\mathcal{K} = \underset{1 \leq k \leq K}{\operatorname{argmax}} \sum_{l=1}^{L} \mathbb{I}\{e_k^l \neq 0\}$$

the ensemble of admissible states for the node, i.e. states that minimize the "parsimonious cost" of a state k. Then the vector p can be computed thanks to the following formula:

$$p_k = \begin{cases} \prod_{l=1}^{L} \left(e_k^l \mathbb{I}\{e_k^l \neq 0\} + \left(\sum_{p=1}^{K} e_p^l \right) \mathbb{I}\{e_k^l = 0\} \right) & \text{if } k \in \mathcal{K} \\ 0 & \text{if } k \notin \mathcal{K} \end{cases}$$

In particular, when we consider a binary tree, a node only has two children of vectors of states l and r, and:

$$\mathcal{K} = \{k \in [\![1\,, K]\!] \mid l_k + r_k > 0\}$$

We can then can distinguish two cases:

$$\begin{cases} \text{If } \langle l, r \rangle \neq 0 & p_k = l_k r_k \\ \text{If } \langle l, r \rangle = 0 & p_k = \begin{cases} l_k \left(\sum_{p=1}^{K} r_p \right) & \text{if } l_k > 0 \\ r_k \left(\sum_{p=1}^{K} l_p \right) & \text{if } r_k > 0 \\ 0 & \text{if } l_k + r_k = 0 \end{cases} \end{cases}$$

where $\langle l, r \rangle$ is the scalaire product between vectors l and r.

Termination At the root 1, the total number of parsimonious solution is the sum of the elements of p the computed vector.

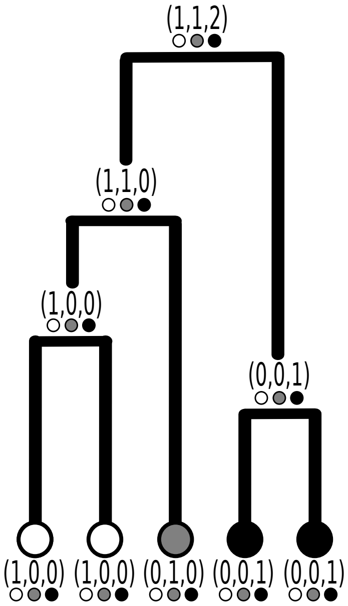


Figure 2.5: An example of the output of the algorithm for a tree with 5 tips grouped in 3 clusters. Here, there are $1 + 1 + 2 = 4$ parsimonious allocations possible. Two of them are presented in figures 2.3c and 2.3d.

Proof of the algorithm The initialization guarantees that this algorithm works for trees with only one node. Recursively, let's show that it works for a tree with one root and L subtrees.

First, notice that, if the root is in a state k, we have to put shifts in the L branches so that every children node is in a state that allows for at least one parsimonious repartition of the shift in its subtree. So, if we put the parent node to state k, then we will need a change on every branches that end to a child with $e_k^l = 0$. We hence get $\sum_{l=1}^L \mathbb{I}\{e_k^l = 0\}$ shifts, a quantity we call the "parsimonious cost" of state k. We have to minimize this quantity, to impose the fewest shifts possible. As $\sum_{l=1}^L \mathbb{I}\{e_k^l = 0\} = n - \sum_{l=1}^L \mathbb{I}\{e_k^l \neq 0\}$, $\mathcal{K} = \operatorname{argmax}_{1 \leq k \leq K} \sum_{l=1}^L \mathbb{I}\{e_k^l \neq 0\}$ is the ensemble of states that minimize that cost, and hence the ensemble of admissible states.

- If $k \notin \mathcal{K}$, then we would have to put more shifts than the minimum amount required to be in an admissible situation, so no parsimonious repartition can start with state k at the root, and $p_k = 0$.
- If $k \in \mathcal{K}$, we have to count all the situations leading to a parsimonious reconstruction. Let $R = \min_{k \in [\![1,K]\!]} \sum_{l=1}^L \mathbb{I}\{e_k^l = 0\}$. We have to put shifts on the R branches that lead to a node with $e_k^l = 0$, and let all the other branches in the parental state. We have two possibilities:
  - If $e_k^l \neq 0$, then we have e_k^l possible parsimonious allocations for the subtree l starting with state k.
  - If $e_k^l = 0$, then we can jump to any state that has parsimonious allocations, and we have $\sum_{p=1}^K e_p^l$ possible parsimonious allocations for the subtree l starting with any state other than k.

  Putting all these sub-trees together, we get $p_k = \prod_{l=1}^L \left(e_k^l \mathbb{I}\{e_k^l \neq 0\} + \left(\sum_{p=1}^K e_p^l \right) \mathbb{I}\{e_k^l = 0\} \right)$.

Implementation. This algorithm was implemented in R [R C14], using package ape [PCS04] for tree manipulations.

Remark. Note that this algorithm can count solutions that do not respect the infinite site model assumption. However, if the clustering of the tips considered is *tree compatible*, then all the solutions counted by the algorithm will respect this assumption.

2.3 Model Selection

2.3.1 Problem and Definitions

Let $\mathcal{T}$ be a fixed rooted tree with m internal nodes and n tips. Denote by $\mathcal{S}_K^P$ the ensemble of parsimonious allocations of K shifts on the $m+n-1$ branches of the tree, and by $\mathcal{C}_{K+1}$ the ensemble of tree-compatible clusterings of the tips in $K+1$ groups. Under the assumption of an *infinite site model*, by proposition 1, the application ϕ (that can be defined by matrix T)

$$\phi : \mathcal{S}_K^P \to \mathcal{C}_{K+1}$$

is surjective, and can be used to define the equivalence relation of the equivalence classes defined in section 2.2.3:

$$\forall s_1, s_2 \in \mathcal{S}_K^P, s_1 \sim s_2 \iff \phi(s_1) = \phi(s_2)$$

And the algorithm described in the previous section counts the cardinal of $\phi^{-}(c)$, for any clustering $c \in \mathcal{C}_{K+1}$. To have a bijective application, we can take the quotient set:

$$\mathcal{S}_K^{PI} = \mathcal{S}_K^P / \sim$$

where $\mathcal{S}_K^{PI}$ can be defined as the ensemble of parsimonious allocations of K shifts on the $m+n-1$ branches of the tree that are identifiable (taking one single representative by equivalence classes). Finally, in the case of an *infinite site model*, we have a bijection between identifiable parsimonious allocation of K shifts and tree-compatible clustering of the tips in $K+1$ groups.

In order to address the issue of model selection, we would like to use the systematic approach developed for Gaussian Model Selection (see [Mas07]). To that end, one of the firsts steps for finding an adapted penalty is to compute the *dimension* of each model. We assume that we have a collection of models $\{S_\eta,\ \eta \in \mathcal{M}\}$, where $\mathcal{M} = \bigcup_{K\geq 0} \mathcal{S}_K^{PI}$ is the set of all the subsets of $[\![1\,, m+n-1]\!]$ that define an identifiable parsimonious allocation of the shifts on the branches of the tree, and a model S_η is the linear subspace of $\mathbb{R}^{m+n-1}$ spanned by coordinates defined by $\eta \in \mathcal{M}$, of dimension $|\eta|$.

The difference here with a classical coordinate sparsity pattern is that not all combinations of coordinates are allowed. For a number of shifts K fixed, the problem is then to count the cardinal $\left|\mathcal{S}_K^{PI}\right| = |\mathcal{C}_{K+1}|$. Indeed, in the process of building a good penalty term, one has to find coefficients L_η such that the sum $\sum_{\eta\in\mathcal{M}} e^{-L_\eta D_\eta}$ is bounded, where D_η is the dimension of the model S_η. In our case:

$$\sum_{\eta\in\mathcal{M}} e^{-L_\eta D_\eta} = \sum_{|\eta|\geq 0} \left|\mathcal{S}_{|\eta|}^{PI}\right| e^{-L_{|\eta|}|\eta|} = \sum_{K\geq 0} \exp\left[-L_K K + \log\left|\mathcal{S}_K^{PI}\right|\right]$$

And to control the sum, we can take, denoting A a positive constant

$$L_K = A + \frac{1}{K}\log\left|\mathcal{S}_K^{PI}\right|$$

In the rest of this chapter, we will compute the cardinal of $\mathcal{C}_{K+1}$.

2.3.2 Number of Partitions of the Tips of a Tree

Definitions. Let $\mathcal{T}$ be a rooted tree with n tips. We want to count the number $N_K^{(\mathcal{T})} = |\mathcal{C}_{K+1}|$ of possible partitions of the tips in K clusters that are compatible with the tree $\mathcal{T}$ and with the shift process. In the following algorithm, we will also need to count the number $A_K^{(\mathcal{T})}$ of possible *marked* partitions that are compatible with the tree, that is, partitions for which the position of one marked group matters.

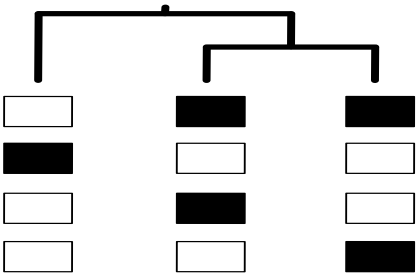


Figure 2.6: Partitions in two groups for a binary tree with 3 tips

We can see here the difference between $N_2^{(\mathcal{T}_3)}$ and $A_2^{(\mathcal{T}_3)}$:

- If we consider only the unmarked partitions, then partitions 1 and 2 are equivalent, and $N_2^{(\mathcal{T}_3)} = 3$.
- If one partition is marked (here for instance, white tips are supposed to be in the "ancestral" state), then partitions 1 and 2 are not equivalent, and $A_2^{(\mathcal{T}_3)} = 4$

Recursion formula, binary case.

Proposition 2. *If $\mathcal{T}$ is a binary tree, consider T_ℓ and $\mathcal{T}_r$ the left and right sub-trees of $\mathcal{T}$. We have the following recursion formula:*

$$\begin{cases} N_K^{(\mathcal{T})} = \displaystyle\sum_{k_1+k_2=K} N_{k_1}^{(\mathcal{T}_\ell)} N_{k_2}^{(\mathcal{T}_r)} + \sum_{k_1+k_2=K+1} A_{k_1}^{(\mathcal{T}_\ell)} A_{k_2}^{(\mathcal{T}_r)} \\ A_K^{(\mathcal{T})} = \displaystyle\sum_{k_1+k_2=K} A_{k_1}^{(\mathcal{T}_\ell)} N_{k_2}^{(\mathcal{T}_r)} + N_{k_1}^{(\mathcal{T}_\ell)} A_{k_2}^{(\mathcal{T}_r)} + \sum_{k_1+k_2=K+1} A_{k_1}^{(\mathcal{T}_\ell)} A_{k_2}^{(\mathcal{T}_r)} \end{cases} \tag{2.2}$$

Proof. If $\mathcal{T}$ is a tree with $\mathcal{T}_\ell$ and $\mathcal{T}_r$ as left and right sub-trees, in odder to get K partitions of the tips of $\mathcal{T}$, one can face two situations (see figure 2.7):

- Left and right sub-trees do not have any group in common. Then, the number of groups in $\mathcal{T}$ is equal to the number of groups in its two sub-trees, and there are $\sum_{k_1+k_2=K} N_{k_1}^{(\mathcal{T}_\ell)} N_{k_2}^{(\mathcal{T}_r)}$ such partitions. This is the first term of the equation on $N_K^{(\mathcal{T})}$ of the proposition.
- Left and right sub-trees have at least one group in common. Then, from the shift process, as we made the hypothesis of an infinite site model, they have exactly one group in common, that corresponds to the ancestral state of the root. Suppose that this ancestral state is marked. Then it must be present in the two sub-trees, and there are $\sum_{k_1+k_2=K+1} A_{k_1}^{(\mathcal{T}_\ell)} A_{k_2}^{(\mathcal{T}_r)}$ such partitions. This ends the proof of the formula on $N_K^{(\mathcal{T})}$.

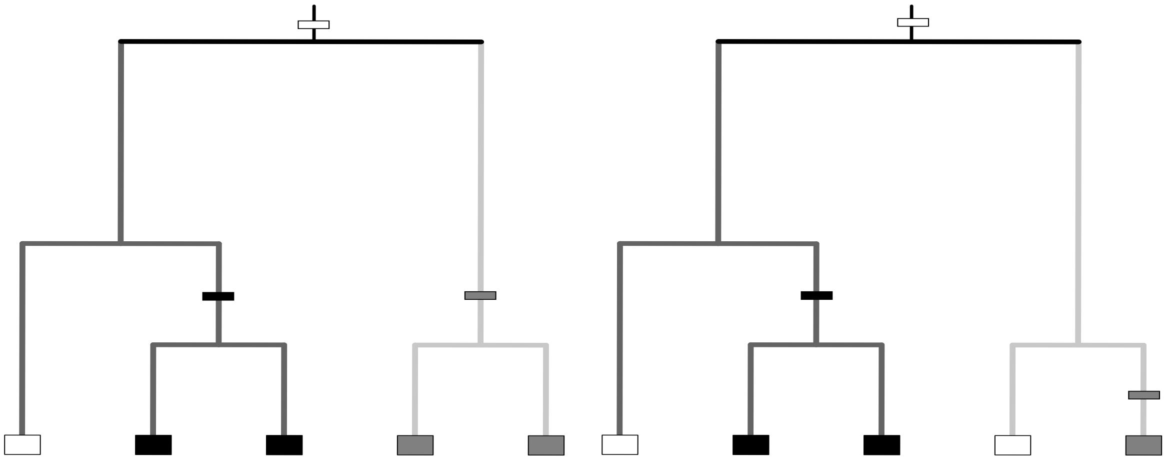
To get the formula on $A_K^{(\mathcal{T})}$, we use the same kind of arguments. The second part of the formula is the same than the on for $N_K^{(\mathcal{T})}$, and the first part corresponds to trees for which the marked partition is present in only one of the two sub-trees. □

Recursion formula, general case.

Proposition 3. *If we are at a node defining a tree $\mathcal{T}$ that has p daughters, with sub-trees $\mathcal{T}_1, \ldots, \mathcal{T}_p$, then we get the following recursion formulas:*

$$\begin{cases} N_K^{(\mathcal{T})} = \displaystyle\sum_{\substack{k_1+\cdots+k_p=K \\ k_1,\ldots,k_p\geq 1}} \prod_{i=1}^{p} N_{k_i}^{(\mathcal{T}_i)} + \sum_{\substack{I\subset[\![1,p]\!] \\ |I|\geq 2}} \sum_{\substack{k_1+\cdots+k_p=K+|I|-1 \\ k_1,\ldots,k_p\geq 1}} \prod_{i\in I} A_{k_i}^{(\mathcal{T}_i)} \prod_{i\notin I} N_{k_i}^{(\mathcal{T}_i)} \\ A_K^{(\mathcal{T})} = \displaystyle\sum_{\substack{I\subset[\![1,p]\!] \\ |I|\geq 1}} \sum_{\substack{k_1+\cdots+k_p=K+|I|-1 \\ k_1,\ldots,k_p\geq 1}} \prod_{i\in I} A_{k_i}^{(\mathcal{T}_i)} \prod_{i\notin I} N_{k_i}^{(\mathcal{T}_i)} \end{cases} \tag{2.3}$$

The proof of this recursion formula uses the same kinds of arguments than in the binary case. I is here the set of subtrees that retain the ancestral state.



(a) Case 1: left and right sub-trees do not have any color in common

(b) Case 2: left and right sub-trees have the ancestral color (white) in common

Figure 2.7: Illustration in the binary case, left and right sub-trees are painted in two different shades of grey.

General expression in the binary case. In the binary case, we can show that $N_K^{(\mathcal{T})}$ does not depends on the topology of the tree, and can hence be denoted $N_K^{(n)}$, with n the number of tips. We have a general expression for this quantity:

Proposition 4. *In the case of a rooted binary tree with n tips, we have:*

$$N_K^{(\mathcal{T})} = N_K^{(n)} = \binom{2n-1-K}{K-1} \quad \text{and} \quad A_K^{(\mathcal{T})} = A_K^{(n)} = \binom{2n-K}{K-1}$$

To prove that, we first need to show that these expressions are coherent with the recursion of proposition 2. This can be done using the following lemma, with $P = K-1$, $n = 2n_\ell - 1$, $n' = 2n_r - 1$, with n_ℓ and n_r the number of tips of the left and right sub-trees of $\mathcal{T}$:

Lemma 1. *Let $(n, n') \in \mathbb{N}$ and $P \in \mathbb{N}$. With the standard convention that $\binom{n}{p} = 0$ if $n < p$,*

$$\binom{n+n'-P}{P} = \sum_{p=0}^{P} \binom{n-p}{p}\binom{n'-P+p}{P-p} + \sum_{p=0}^{P-1} \binom{(n-1)-p}{p}\binom{(n'-1)-(P-1)+p}{(P-1)-p}$$

and

$$\binom{n+n'+1-P}{P} = \sum_{p=0}^{P} \binom{n-p}{p}\binom{n'-P+p}{P-p} \\ + \sum_{p=0}^{P-1} \binom{(n-1)-p}{p}\binom{n'-(P-1)+p}{(P-1)-p} + \binom{n-p}{p}\binom{(n'-1)-(P-1)+p}{(P-1)-p}$$

This lemma is proven in annex A. Then, the property "$N_K^{(\mathcal{T})}$ and $A_K^{(\mathcal{T})}$ do not depend on the topology of $\mathcal{T}$ and have the expressions given in proposition 4" can be proven with a strong induction, as it is obviously true for trees with one or two tips.

Implementation. We implemented this algorithm in the general case in R [R C14], using package combinat [Cha12] for the combinatorial sums.

Chapter 3

Inference of the Parameters

3.1 The Expectation Maximization Algorithm in the Brownian Motion Case

In the rest of this document, we consider the problem of the estimation of the parameters of the model of evolution of a single trait, with the number of change-points K pre-specified.
In the following, expectations and variances are all taken for a given set of parameters θ ($\mathbb{E}_\theta$, $\mathbb{V}\mathrm{ar}_\theta$), that will be omitted for the sake of clarity.

3.1.1 The EM algorithm

The goal of the EM algorithm is to maximize the conditional expectation of the log likelihood of the completed dataset $\mathbb{E}\left[\log p_\theta(X) \mid Y\right]$. In the case of the BM, using equation (1.5), we get the following expression to maximize:

$$
\begin{aligned}
\mathbb{E}^Y\left[\log p_\theta(X)\right] = & \mathrm{cst} - \frac{1}{2}\log\gamma^2 - \frac{m+n-1}{2}\log\sigma^2 - \frac{1}{2\gamma^2}\left(\mathbb{V}\mathrm{ar}^Y[Z_1] + \left(\mathbb{E}^Y[Z_1]-\mu\right)^2\right) \\
& - \frac{1}{2\sigma^2}\left(\sum_{1<j\leq m}\ell_j^{-1}\mathbb{V}\mathrm{ar}^Y\left[Z_j - Z_{\mathrm{pa}(j)}\right] + \sum_{1\leq i\leq n}\ell_{i'}^{-1}\mathbb{V}\mathrm{ar}^Y Z_{\mathrm{pa}(i')}\right) \\
& - \frac{1}{2\sigma^2}\left(\sum_{1<j\leq m}\ell_j^{-1}\left(\mathbb{E}^Y Z_j - \mathbb{E}^Y Z_{\mathrm{pa}(j)} - \sum_k \mathbb{I}\{\tau_k = b_j\}\delta_k\right)^2\right. \\
& \qquad \left. + \sum_{1\leq i\leq n}\ell_{i'}^{-1}\left(Y_i - \mathbb{E}^Y Z_{\mathrm{pa}(i')} - \sum_k \mathbb{I}\{\tau_k = b_{i'}\}\delta_k\right)^2\right)
\end{aligned}
\tag{3.1}
$$

where $\mathbb{E}^Y$ (resp. $\mathbb{V}\mathrm{ar}^Y$) is the conditional mean (resp. variance) given Y.

The basic scheme of the EM algorithm is then the following:

E step Compute the following moments:

$$
\begin{cases}
\mathbb{E}^{(h)}[Z_1 \mid Y] \text{ and } \mathbb{V}\mathrm{ar}^{(h)}[Z_1 \mid Y] & \\
\mathbb{E}^{(h)}[Z_j \mid Y] - \mathbb{E}^{(h)}[Z_{\mathrm{pa}(j)} \mid Y] \text{ and } \mathbb{V}\mathrm{ar}^{(h)}[Z_j - Z_{\mathrm{pa}(j)} \mid Y] & \forall j \in [\![2\,, m]\!] \\
Y_i - \mathbb{E}^{(h)}[Z_{\mathrm{pa}(i')} \mid Y] \text{ and } \mathbb{V}\mathrm{ar}^{(h)}[Z_{\mathrm{pa}(i')} \mid Y] & \forall i \in [\![1\,, n]\!]
\end{cases}
$$

assuming parameters $(\mu^{(h)}, \gamma^{2(h)}, \sigma^{2(h)}, \tau^{(h)}, \delta^{(h)})$ are known.

M step Given the moments computed at the previous E step, maximize function (3.1) in $(\mu, \gamma^2, \sigma^2, \tau, \delta)$ to find $(\mu^{(h+1)}, \gamma^{2(h+1)}, \sigma^{2(h+1)}, \tau^{(h+1)}, \delta^{(h+1)})$.

In the rest of the section, we will discuss how these two steps can be performed efficiently in the case of the BM.

3.1.2 M step: Segmentation

In the maximization step, we have to deal with discrete variables: the position of the change-points τ. Notice that only the last term of the expression to be maximized (1.5), which is a sum of squared quantities, depends on τ. The problem is then to minimize $C(\tau, \delta) = \sum_{1<j\leq m+n} \ell_j^{-1} \left(\mathbb{E}^Y X_j - \mathbb{E}^Y X_{\mathrm{pa}(j)} - \sum_k \mathbb{I}\{\tau_k = b_j\}\delta_k\right)^2$, that can be seen as a sum of costs associated with each branch (recall that $\mathbb{E}^Y Z_j = \mathbb{E}^Y X_j$ and $Y_i = \mathbb{E}^Y Y_i = \mathbb{E}^Y X_{m+i}$). To minimize this cost, we can use the following algorithm.

Algorithm of Segmentation. Define

$$C(\tau, \delta) = \sum_{1<j\leq m+n} C_j(\tau, \delta)$$

where the cost C_j are defined as

$$C_j(\tau, \delta) = \ell_j^{-1} \left(\mathbb{E}^Y X_j - \mathbb{E}^Y X_{\mathrm{pa}(j)} - \sum_k \mathbb{I}\{\tau_k = b_j\}\delta_k\right)^2$$

Each cost can only take 2 values:

- if $\sum_k \mathbb{I}\{\tau_k = b_j\} = 0$ (no shift occurs on branch b_j) ,

$$C_j(\tau, \delta) = \ell_j^{-1} \left(\mathbb{E}^Y X_j - \mathbb{E}^Y X_{\mathrm{pa}(j)}\right)^2 =: C_j^0(\tau)$$

- if $\sum_k \mathbb{I}\{\tau_k = b_j\} = 1$ (one shift occurs on branch b_j),

$$C_j(\tau, \delta) = \ell_j^{-1} \left(\mathbb{E}^Y X_j - \mathbb{E}^Y X_{\mathrm{pa}(j)} - \sum_k \mathbb{I}\{\tau_k = b_j\}\delta_k\right)^2 =: C_j^1(\tau, \delta)$$

We can see that if a change-point is allocated to branch b_j, then, taking $\delta_k = \mathbb{E}^Y X_j - \mathbb{E}^Y X_{\mathrm{pa}(j)}$ for the k such that $\tau_k = b_j$, we can cancel the cost $C_j^1(\tau, \delta)$:

$$C_j^1(\tau) := \arg\min_{\delta} C_j^1(\tau, \delta) = 0$$

In the sum, the only remaining costs are then the ones associated with branches with no shifts, and we get to minimize:

$$C(\tau) = \arg\min_{\delta} C(\tau, \delta) = \sum_{j=2}^{m+n} \mathbb{I}\left\{\sum_k \mathbb{I}\{\tau_k = b_j\} = 0\right\} C_j^0(\tau)$$

This can be done easily: it suffices to allocate change points on the branches associated with the K largest costs $C_j^0(\tau)$.

We end up with the following segmentation algorithm, for a known K:

1. Find the K branches $j_1, \dots, j_K$ with largest $C_j^0(\tau)$;
2. Allocate one change point in the first K branches;
3. For each of these, set $\delta_{j_k}^{(h+1)} = \mathbb{E}^{(h)}\left[X_{j_k} \mid Y\right] - \mathbb{E}^{(h)}\left[X_{\mathrm{pa}(j_k)} \,\middle|\, Y\right]$.

3.1.3 M step: Continuous Variables

The maximization in the other continuous variables is then straightforward, and we get the following expression for the actualization formulas:

$$\begin{cases} \mu^{(h+1)} &= \mathbb{E}^{(h)}[Z_1 \mid Y] \\ \gamma^{2(h+1)} &= \mathbb{V}\mathrm{ar}^{(h)}[Z_1 \mid Y] \\ \sigma^{2(h+1)} &= \dfrac{1}{m+n-1}\left[\sum\limits_{1<j\leq m+n} \ell_j^{-1}\mathbb{V}\mathrm{ar}^{(h)}\left[X_j - X_{\mathrm{pa}(j)} \mid Y\right] + C\left(\tau^{(h+1)}, \delta^{(h+1)}\right)\right] \end{cases}$$

3.1.4 E step: Upward-Downward Algorithm

For the E step, we have several possibilities:

- We can directly use the the fact that the process is Gaussian, and use conditional Gaussian properties to compute the mean vector and variance-covariance matrix of $Z \mid Y$. As:

  $$X = (Z, Y) \sim \mathcal{N}\left(m = \begin{pmatrix} m_Z \\ m_Y \end{pmatrix}, \Sigma = \begin{pmatrix} \Sigma_{ZZ} & \Sigma_{YZ} \\ \Sigma_{ZY} & \Sigma_{YY} \end{pmatrix}\right)$$

  we easily get:

  $$Z \mid Y \sim \mathcal{N}\left(m_{Z|Y} = m_Z + \Sigma_{ZY}\Sigma_{YY}^{-1}(Y - m_Y)\ ,\ \Sigma_{Z|Y} = \Sigma_{ZZ} - \Sigma_{ZY}\Sigma_{YY}^{-1}\Sigma_{YZ}\right)$$

  We can see that this approach forces us to invert Σ_{YY}, a $n \times n$ matrix, which is a costly operation, in $O(n^3)$.

- We could also use the algorithm described in [HA13a], that is designed specifically to compute quantities of the form $X^T V^{-1} Y$ for a class of matrices V that includes our variance-covariance matrix Σ_{YY} of the tips of a tree. This algorithm has a number of steps that is linear in n the number of tips, but, as we need to compute $2m$ quantities, we end up with a complexity in $O(mn)$.

- In order to take advantage of the specific structure of our problem, that consists in computing moments of Gaussian variables that are mother and daughter on a tree, we can use an “Upward-Downward” algorithm, that is inspired from the well known forward-backward algorithm used in segmentation problems. This algorithm has a complexity in $O(n)$. It is described in details in appendix B.

3.2 The Expectation Maximization Algorithm in the Orstein-Uhlenbeck Case

3.2.1 Adaptation from the Brownian Motion Case

In the OU case, we need to maximize the following function, issued from (1.11):

$$
\begin{aligned}
\mathbb{E}^{Y}\left(\log p_{\theta}(X)\right) = \text{cst} - \frac{1}{2}\log\gamma^2 - \frac{m+n-1}{2}\log\sigma^2 + \frac{m+n-1}{2}\log 2\alpha - \frac{1}{2}\sum_{j=2}^{m+n}\log\left(1-e^{-2\alpha\ell_j}\right) \\
- \frac{1}{2\gamma^2}\left(\mathbb{V}\text{ar}^{Y}[Z_1] + \left(\mathbb{E}^{Y}[Z_1]-\mu\right)^2\right) \\
- \frac{\alpha}{\sigma^2}\sum_{1<j\leq m}\left(1-e^{-2\alpha\ell_j}\right)^{-1}\mathbb{V}\text{ar}^{Y}\left[Z_j - Z_{\text{pa}(j)}e^{-\alpha\ell_j}\right] \\
- \frac{\alpha}{\sigma^2}\sum_{1\leq i\leq n}\left(1-e^{-2\alpha\ell_{i'}}\right)^{-1}e^{-2\alpha\ell_{i'}}\mathbb{V}\text{ar}^{Y}\left[Z_{\text{pa}(i')}\right] \\
-\frac{\alpha}{\sigma^2}\sum_{1<j\leq m}\left(1-e^{-2\alpha\ell_j}\right)^{-1}\Bigg(\mathbb{E}^{Y}Z_j - e^{-\alpha\ell_j}\mathbb{E}^{Y}Z_{\text{pa}(j)} - \beta^{\text{pa}(j)}\left(1-e^{-\alpha\ell_j}\right) \\
- \sum_{k=1}^{K}\mathbb{I}\{\tau_k = b_j\}\delta_k\left(1-e^{-\alpha(1-\nu_k)\ell_j}\right)\Bigg)^2 \\
-\frac{\alpha}{\sigma^2}\sum_{1\leq i\leq n}\left(1-e^{-2\alpha\ell_{i'}}\right)^{-1}\Bigg(Y_i - e^{-\alpha\ell_{i'}}\mathbb{E}^{Y}Z_{\text{pa}(i')} - \beta^{\text{pa}(i')}\left(1-e^{-\alpha\ell_{i'}}\right) \\
- \sum_{k=1}^{K}\mathbb{I}\{\tau_k = b_{i'}\}\delta_k\left(1-e^{-\alpha(1-\nu_k)\ell_{i'}}\right)\Bigg)^2
\end{aligned}
\tag{3.2}
$$

We can see that, compared with the BM, there are $K+2$ extra parameters to estimate: β_0, ν_k and α. In the rest of this document, we will get rid of the first two families of parameters by making the following assumptions:

Hypotheses:

- The root node is in the primitive stationary state: $\mu = \beta_0$ and $\gamma^2 = \frac{\sigma^2}{2\alpha}$. This allows us to get rid of parameters β_0 and σ^2 is the estimation.
- The change-points occur right after speciation events: $\nu_k = 0$. This assumption fixes K parameters, and simplifies the expressions to be maximized.

With these assumption, the only remaining extra parameter is α. First, let's assume that this parameter is known. If so, we can use the almost exact same algorithm as in the BM case:

E step: We can use the same strategy as before, and the linear Upward-Downward algorithm described in appendix B works.

M step, Segmentation: If α is known, we can use the same algorithm, with modified costs:

1. Find the K branches $j_1, \dots, j_K$ with largest $C_j^0(\tau)$,

$$
C_j^0(\tau) = \left(1-e^{-2\alpha\ell_j}\right)^{-1}\left(\mathbb{E}^{(h)}[X_j \mid Y] - e^{-\alpha\ell_j}\mathbb{E}^{(h)}[X_{\text{pa}(j)} \mid Y] - \beta^{(h)}_{\text{lin}(j)}(t_{\text{pa}(j)})\left(1-e^{-\alpha\ell_j}\right)\right)^2
$$

2. Allocate one change point in the first K branches;
3. For each of these branches, set:

$$\delta_{j_k}^{(h+1)} = \left(1 - e^{-\alpha \ell_{j_k}}\right)^{-1} \left(\mathbb{E}^{(h)}\left[X_{j_k} \mid Y\right] - e^{-\alpha \ell_{j_k}} \mathbb{E}^{(h)}\left[X_{\text{pa}(j_k)} \,\middle|\, Y\right]\right) - \beta_{\text{lin}(j_k)}^{(h)}(t_{\text{pa}(j_k)})$$

M step, Continuous Parameters: We have close formulas for the actualization of parameters μ and γ^2:

$$\begin{cases} \mu^{(h+1)} = \mathbb{E}^{(h)}[Z_1 \mid Y] \\ \gamma^{2(h+1)} = \dfrac{1}{m+n} \left(\begin{array}{r} \mathbb{V}\text{ar}^{(h)}[Z_1 \mid Y] + \displaystyle\sum_{1<j\leq m+n} \left(1 - e^{-2\alpha\ell_j}\right)^{-1} \mathbb{V}\text{ar}^{(h)}\left[X_j - X_{\text{pa}(j)} e^{-\alpha\ell_j} \mid Y\right] \\ + \displaystyle\sum_{1<j\leq m+n} C_j\left(\tau^{(h+1)}, \delta^{(h+1)}\right) \end{array} \right) \end{cases}$$

3.2.2 Estimation of the Selection Strength

In order to complete our EM algorithm, we need to find a way of estimating α. To do that, we can maximize in α expression (3.2), all the other parameters being known. With the assumption made above, the function to be maximized can be re-written in a function of the following form:

$$R(\alpha) = \frac{K_1\alpha}{\sigma^2} + \sum_{j=2}^{m+n} -\frac{1}{2}\log\alpha + \frac{1}{2}\log\left(1 - e^{-2\alpha\ell_j}\right) + \frac{1}{\sigma^2}\frac{\alpha}{1 - e^{-2\alpha\ell_j}}\left(a_j e^{-2\alpha\ell_j} - 2b_j e^{-\alpha\ell_j} + c_j\right)$$

where K_1 is a positive constant, and a_j, b_j and c_j are such that: $a_j, c_j \geq 0$, and $b_j^2 - a_j c_j \leq 0$ (the polynomial in $e^{-\alpha\ell_j}$ is always positive).

We tried to study the properties of this function, that seems to be convex for several sets of parameters, but with no success so far. As a function of one parameter, it can be numerically minimized, but with no theoretical guaranty of finding a global optimum. In [BK04], the authors, trying to maximize the likelihood of the data conditionally to the position of the shifts in the tree, had already encountered problems with the estimation of α.

At the M step of an EM algorithm, we actualize τ, δ, μ, γ^2 knowing the previous estimation of α, and then compute a new estimation of α using the actualized versions of the above parameters.

3.3 Initializations

The initialization is always crucial in an EM algorithm. We discuss here ways of initializing two determining parameters of the model: the position of the shifts, and the selection strength α.

3.3.1 A Lasso Initialization of the Shifts

As seen section 2.1.2, when the tree is ultrametric, we can express the model as a linear regression problem:

$$Y = T\Delta(\alpha) + E$$

with T a $n \times (m+n)$ structural matrix depending on the topology of the tree $\mathcal{T}$ studied, E a vector of error of size n: $E \sim \mathcal{N}(0, \Sigma_{YY})$, and $\Delta(\alpha)$ a vector of (actualized) shifts.

Assume Σ_{YY} (that depends on all the parameters of the model) is known. Then we would have a simple linear regression, with a known estimator for Δ. Here, as the non-zero entries of Δ represents the positions of the shifts, we would like to impose a coherent structure on admissible solution vectors (see section 2.3). Finding the good penalty for our exact problem is still an open question, but we have a good way for selecting solutions that are coordinate sparse: the lasso penalty. The problem of estimation of Δ would then be:

$$\hat{\Delta} = \underset{\Delta}{\operatorname{argmin}} \left\{ \|Y - T\Delta\|^2_{\Sigma_{YY}} + \lambda \left|\Delta\right|_1 \right\}$$

An initialization procedure could then be the following:

1. Compute a variance-covariance matrix Σ_{YY} according to a default Stochastic Process model. This first approximation is likely to be very rough, but we assume, as it seems to be the case in our simulation, that the final estimation of the position of the shifts is not very dependent on this matrix.

2. Make a Gauss-Lasso estimation of Δ. This give us a first estimation of the position of the change-points (that might not be parsimonious) and of the values of the root node mean and of the shifts.

3.3.2 Robust Estimation of the Selection Strength

Suppose that we can find to tips i, j such as there is no change point in the path on the tree between them. Then, the observations Y_i and Y_j would have the same mean m, same variance γ^2 (we assumed that the root is in the stationary state), and a covariance $\sigma_{ij} = \gamma^2 e^{-\alpha d_{ij}}$. Hence:

$$\mathbb{E}\left[(Y_i - Y_j)^2\right] = 2\gamma^2 - 2\sigma_{ij}^2 = 2\gamma^2(1 - e^{-\alpha d_{ij}})$$

and $(Y_i - Y_j)^2$ is an unbiased estimator for $2\gamma^2(1 - e^{-\alpha d_{ij}})$.

Assume now that we can find many such estimators. Then, in an approach inspired by geostatistics, we could plot $(Y_i - Y_j)^2$ as a function of d_{ij}, and fit the function $d \mapsto 2\gamma^2(1 - e^{-\alpha d})$ to the data:

$$(\hat{\alpha}, \hat{\gamma^2}) = \underset{(\alpha,\gamma^2)}{\operatorname{argmin}} \sum_{ij \text{ couples}} L\left((Y_i - Y_j)^2 - 2\gamma^2(1 - e^{-\alpha d_{ij}})\right)$$

To compensate for the many poor estimator we might have, we want to fit this function in a robust way, using for the loss function L, instead of the simple quadratic loss, a Huber loss (for instance):

$$L^H_\epsilon : x \mapsto \begin{cases} \frac{x^2}{2} & \text{if } |x| < \epsilon \\ \epsilon(|x| - \frac{\epsilon}{2}) & \text{otherwise.} \end{cases}$$

that is linear after a threshold ϵ, hence less penalizing for very bad points, compared with the quadratic loss.

This method only works if we are able to select couples of tips with no shifts between then, i.e. belonging to the same cluster. To do this, we can limit ourselves to couples of tips the most recent common ancestor of which is “not too high” in the tree, hoping that the occurrence of a shift in that (short) period of time is unlikely. Another way to do that is to get a first initialization of the shifts before estimating α. Hence we could use the following procedure:

1. Get a first estimation of the position of the change points with a lasso procedure (described above).

2. Compute the estimator $(Y_i - Y_j)^2$ for all couples of tips that are in the same cluster according to the initialization of the shifts we just did.

3. Make the regression on (α, γ) as described above, and take $\hat{\alpha}$ for the initialization of the EM algorithm.

Note that this can also provide us with an estimation of γ^2: we just have to take the empirical variance of the tips, respecting their repartition in clusters found by the first lasso initialization.

3.4 Some Results

We implemented all the algorithms described in the document in R ([R C14]), using packages ape [PCS04] for the tree manipulation; glmnet [FHT10] for lasso regression; robustbase [RCT+14] for robust estimation of α; ggplot2 [Wic09], reshape2 [Wic07] and grid [R C14] for plotting the results; and mclust [FRMS12] for computations of the Adjusted Rank Index when evaluating quality of the tip clustering.

3.4.1 Difficulty of a simulation

The difficulty of inference of one configuration depends on several parameters, such as variance parameters (σ^2, α) and expectation parameters (β_0, δ). But it also depends on the number and position of shifts in the tree. For instance, a shift, even of great intensity, that affects only one branch is difficult to infer.

Suppose we simulate a process with some known parameters. To quantify its inference difficulty, we use the Gaussian description of the vector of traits at the tips $Y \sim \mathcal{N}(m_Y, \Sigma_{YY})$. If all tips were in the same group, the mean vector of Y would be of the form $m_Y = m_0\mathbf{1}$, and the best value for m_0 would be given by:

$$m_0 = (\mathbf{1}^T\Sigma_{YY}^{-1}\mathbf{1})^{-1}\mathbf{1}^T\Sigma_{YY}^{-1}m_Y$$

Then groups will be easier to find if the “difference” between m_Y and $m_0\mathbf{1}$ is large, in Mahalanobis distance. We thus compute the difficulty D of a setting as:

$$D = \|m_Y - m_0\mathbf{1}\|^2_{\Sigma_{YY}^{-1}}$$

3.4.2 Orstein-Uhlenbeck: Simple Simulations

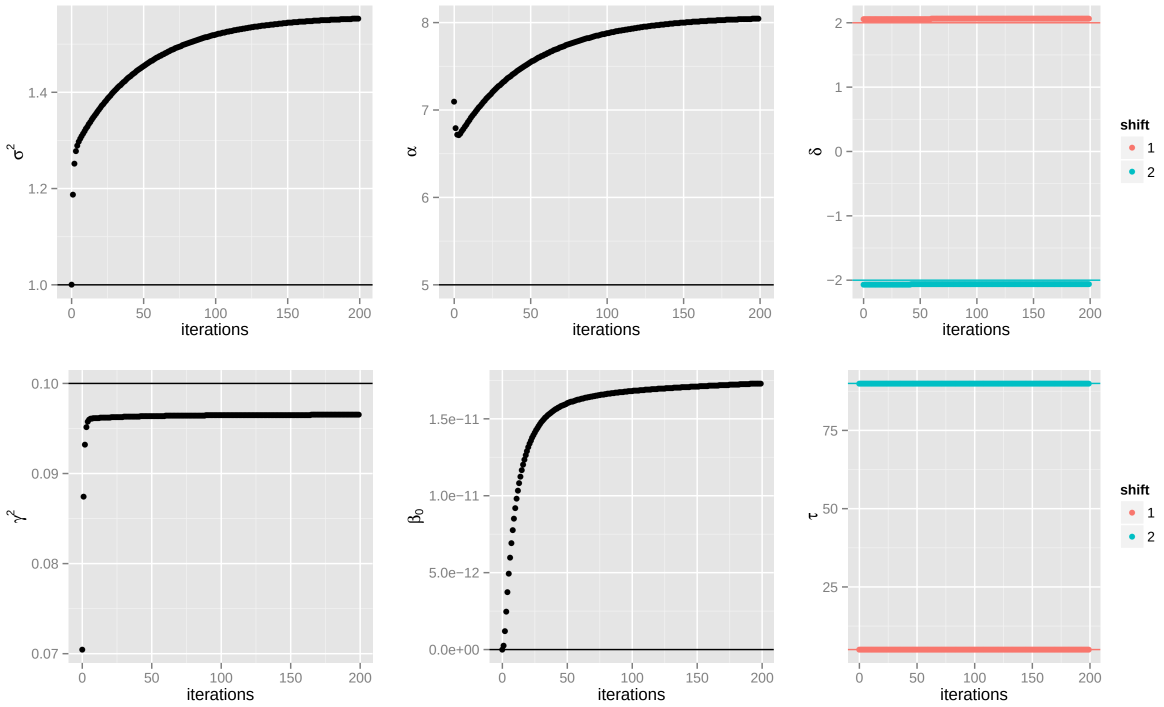
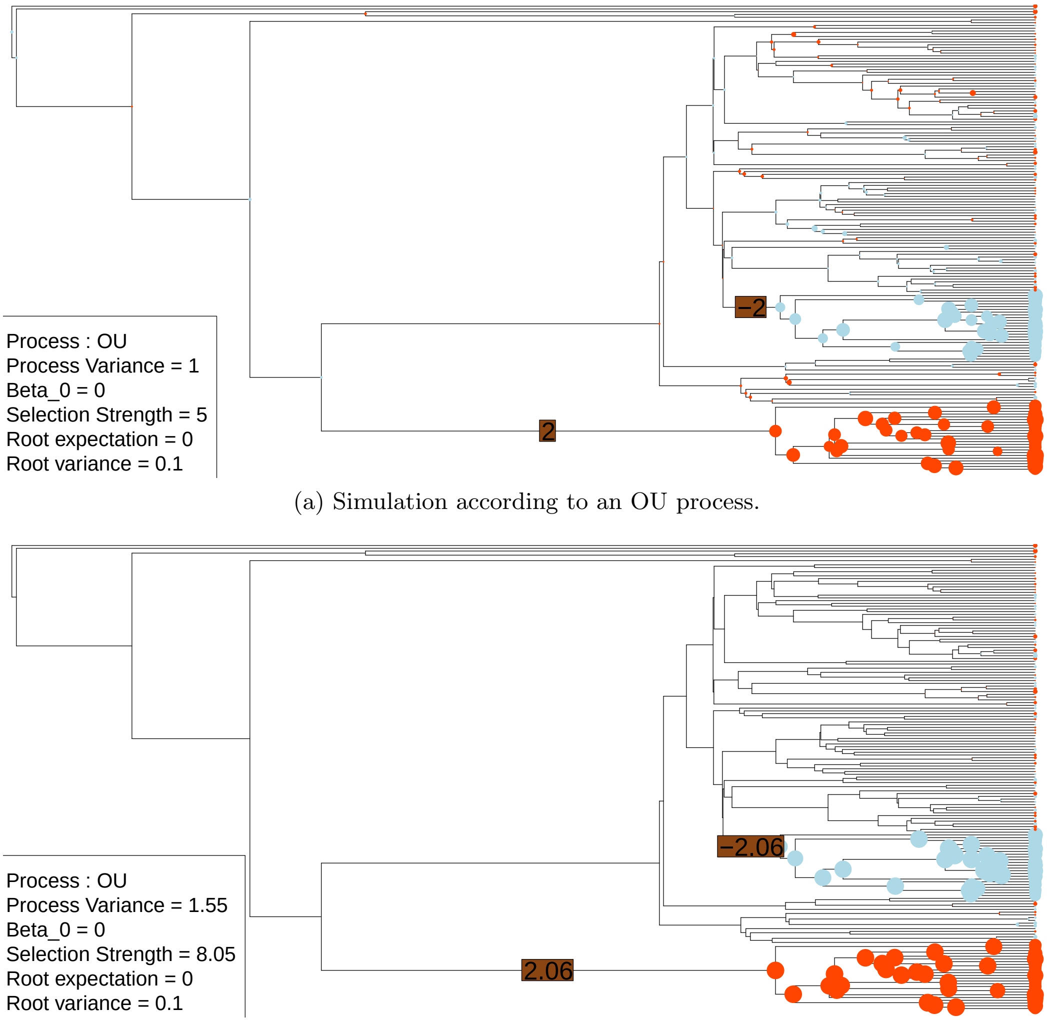


Figure 3.1: Historic of the estimation of the parameters during the iterations of the EM algorithm. Iteration 0 is the initialisation. Vertical lines are the true values of the parameters.

We present here some results for a simple simulation, according to an OU process evolving on the Mammals phylogeny (presented figure 1.1) with two shifts of intensity 2 and -2 (see figure 3.2a), a large selection strength $\alpha = 5$, a variance $\sigma = 1$, and an initial optimal state $\beta_0 = 0$. The root is taken in the stationary state, and shifts occur right after speciation events (at the beginning of a branch). The difficulty criterion is here equal to $D = 1574.102$ (which is quite high).

Figure 3.1 shows the evolution of the estimation of the parameters through the 200 iterations of the EM algorithm. Iteration 0 is the initialization. We can see that the Lasso initialization does a very good job: it is able to find directly the two shifts, with the correct value. This estimation is then kept by the EM, that computes estimations for the other parameters. The initial optimal value, along with the shift values, is well estimated. But we can see that even in this simple case, the selection strength α parameter is not very good, over estimated, by the robust estimation (around 7), and then again by the EM algorithm (around 8). The variance σ^2 is also over estimated (around 1.6), but the root variance $\gamma^2 = \frac{\sigma^2}{2\alpha}$ is quite well estimated (around 0.095 instead of 0.1). This is not surprising, since that, in the case were the root in in the stationary state, all the tips share the same variance γ^2, and there are plenty of observations to estimate it.



(a) Simulation according to an OU process.

(b) Estimation with an EM algorithm.

Figure 3.2: A representation of the value of the trait on the tree issued from [M+11]. At each node, the diameter of the disc is proportional to the absolute value of the trait, and the disc is red for positive traits, blue for negative ones.

3.4.3 Brownian Motion: Systematic Tests

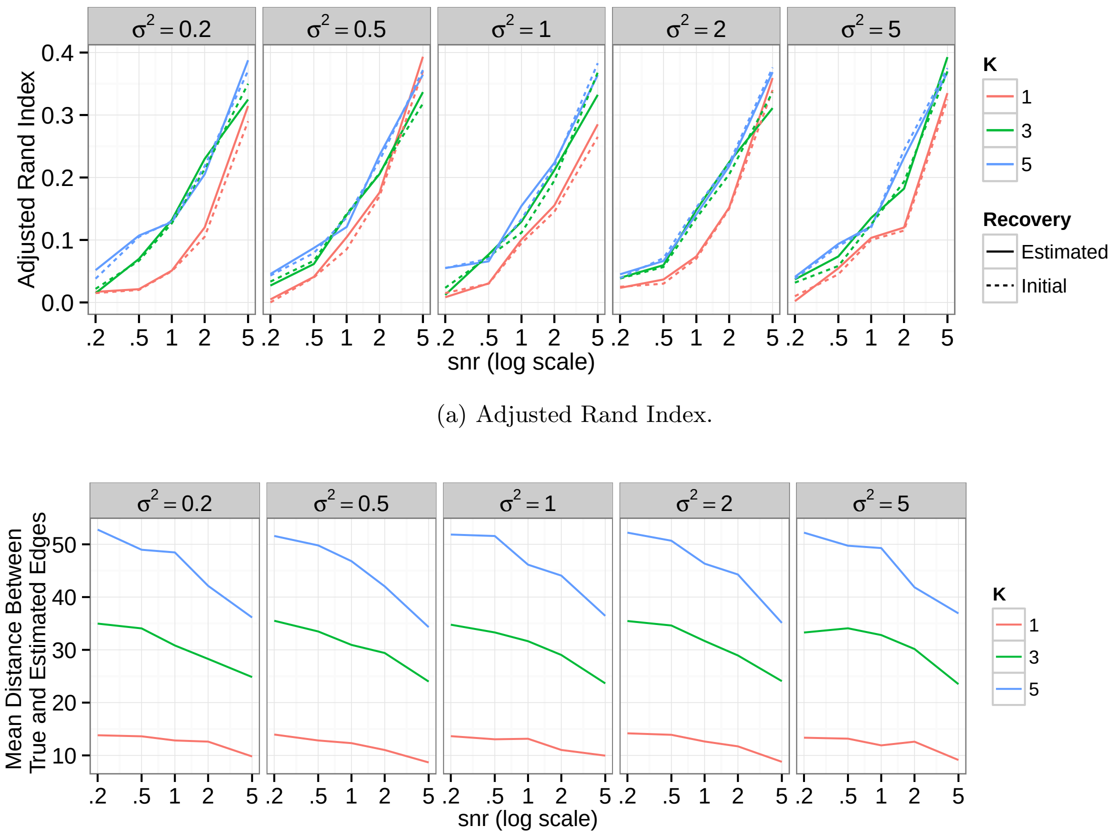
We first analyze in details the performance of our algorithm in the case of the inference of the parameters of a BM. We simulate many sets of data according to a shifted BM, with a deterministic root, and with the following parameters:

- The topology of the tree $\mathcal{T}$ is fixed (see figure 1.1).
- The initial value of the trait at the root is fixed: $\mu = 0$
- The variance σ^2 varies: $\sigma^2 \in \{0.2, 0.5, 1, 2, 5\}$.
- The number of shifts varies: $K \in \{1, 3, 5\}$. The shifts are randomly allocated to the branches of the tree, according to a uniform random sampling without replacement.
- The "signal to noise ratio", defined as $\text{snr} = \frac{\sigma_\delta^2}{\sigma^2}$, varies: $\text{snr} \in \{0.2, 0.5, 1, 2, 5\}$. Here, σ_δ^2 is the variance of the randomly selected values of the shifts for the several simulations. The intensity of each shift is chosen according to a Gaussian of mean 0 and variance σ_δ^2.
- Each configuration is simulated and estimated $n = 200$ times.

This scheme of experiments leads to 15000 simulated sets of data. For each of these sets, we try to estimate back the parameters of the process, using the right model: a BM with the right number of shifts. These simulation-estimation cycles were computed in approximately 10 hours on the cluster *migale*, hosted by the laboratory MIG, Jouy-en-Josas, with 8 cores mobilized. Parallel computations were achieved in R using packages foreach [AW14b] and doParallel [AW14a].

The EM algorithm converged every time. To asses the quality of the reconstruction of the position of the shifts, we used several scores (see figure 3.3). The *Adjusted Rand Index* (ARI, figure 3.3a) is a measure of the quality of the clustering of the tips obtained, compared with the true clustering. It is between -1 and 1, and positive if our partition "does better" than a simple random partition. We can see that our algorithm gets mediocre results, with an ARI varying between 0 and 0.4. The ARI rises with the snr, which is coherent. On this figure, the dashed lines represent the ARI of the clustering of the tips given by the allocation of the shifts we got from the lasso initialization. We can see that it is almost as good as the ARI of the final clustering. The lasso initialization hence seems to do "all the work" in term of allocation of the shifts, as the EM is not able to improve this first solution. Figure 3.3b shows the distance on the tree between estimated and true positions of the shifts. The distance between two sets of edges is the minimum sum of pairwise distances for a bijection between the two sets. Exact matches correspond to a distance of 0, and the maximal distance between two edges is bounded by the total number of edges in the tree, here $m + n - 1 = 336$. As expected, this score decreases with the snr. It also increases when the number of shifts rises, indicating that our estimation of the position of the shifts is to become bad (at least for some of the shifts) when allowing for more shifts.

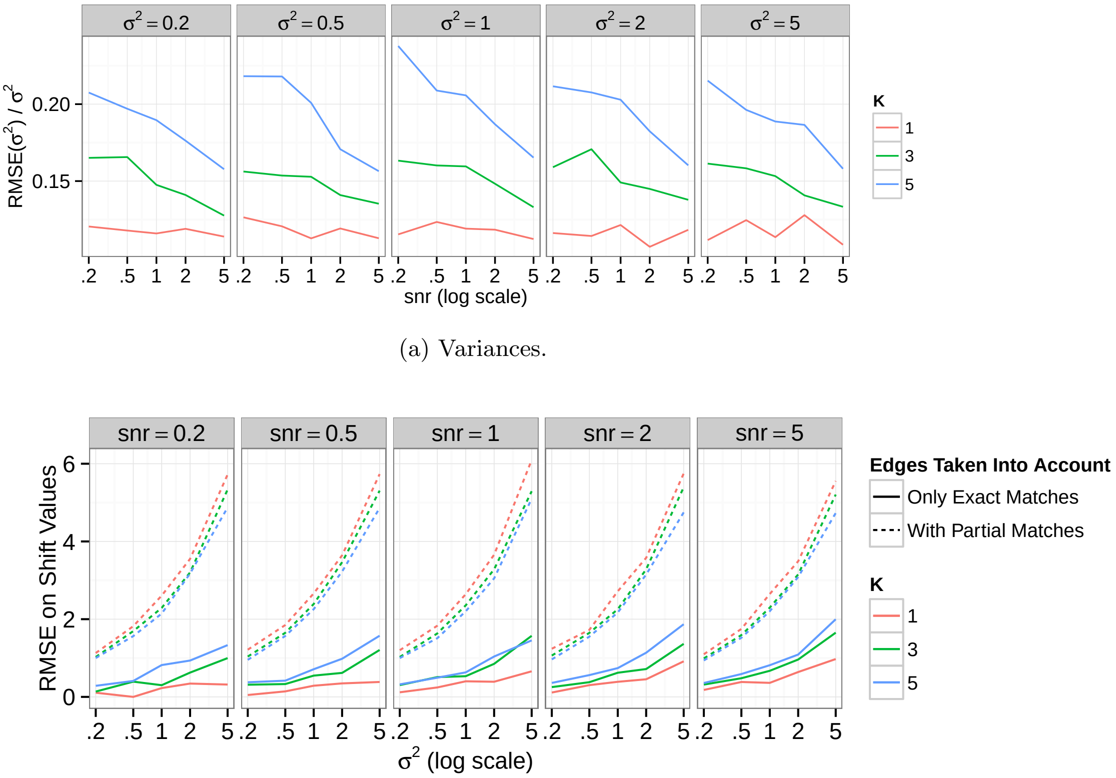
The quality of the estimation of continuous parameters is assessed thanks to the Root Mean Square Expectation (RMSE), shown figure 3.4. On figure 3.4a we can see that the relative RMSE on the variance is never greater than 0.25, and decreases, as expected, when the snr rises, or when the number of shifts decreases. Figure 3.4b shows the RMSE on the shifts values when matching estimated edges with true edges to get the minimum distance (dashed lines) or when considering only exact matches (solid line). The dependence in σ^2 is here more pronounced than the dependence in the snr.



(a) Adjusted Rand Index.

(b) Distances on the tree.

Figure 3.3: Edge Recovery Information, mean on the n realizations.



(a) Variances.

(b) Shifts Values.

Figure 3.4: RMSE of several estimated parameters.

3.4.4 Orstein-Uhlenbeck: Systematic Tests

We now use a similar simulation-estimation procedure to test our algorithm in the case of an OU process with a stationary root. We simulate 75000 sets of data using the following parameters:

- The topology of the tree $\mathcal{T}$ is fixed (see figure 1.1).
- The initial optimal value is fixed: $\beta_0 = 0$
- The selection strength varies: $\alpha \in \{0.2, 0.5, 1, 2, 5\}$.
- The root variance $\gamma^2 = \frac{\sigma^2}{2\alpha}$ varies: $\gamma^2 \in \{0.2, 0.5, 1, 2, 5\}$.
- The "signal to noise ratio" (snr $= \frac{\sigma_\delta^2}{\sigma^2}$) varies: snr $\in \{0.2, 0.5, 1, 2, 5\}$.
- The number of shifts varies: $K \in \{1, 3, 5\}$. The shifts are randomly allocated to the branches of the tree, according to a uniform random sampling without replacement.
- Each configuration is simulated and estimated $n = 200$ times.

As previously, we try to estimate the parameters of the right model, an OU process with the right number of shifts, for each set of data. We here make two attempts to estimate the parameters: a first one assuming that the selection strength α is known, and a second one relaxing this hypothesis and estimating α from the data. As expected, the first attempt was much more successful, confirming the fact that α is a critical and hard to estimate parameter. The computations for the first and second attempts took respectively around 24 and 44 hours on the cluster *migale*, hosted by the laboratory MIG, Jouy-en-Josas, with 8 cores mobilized.

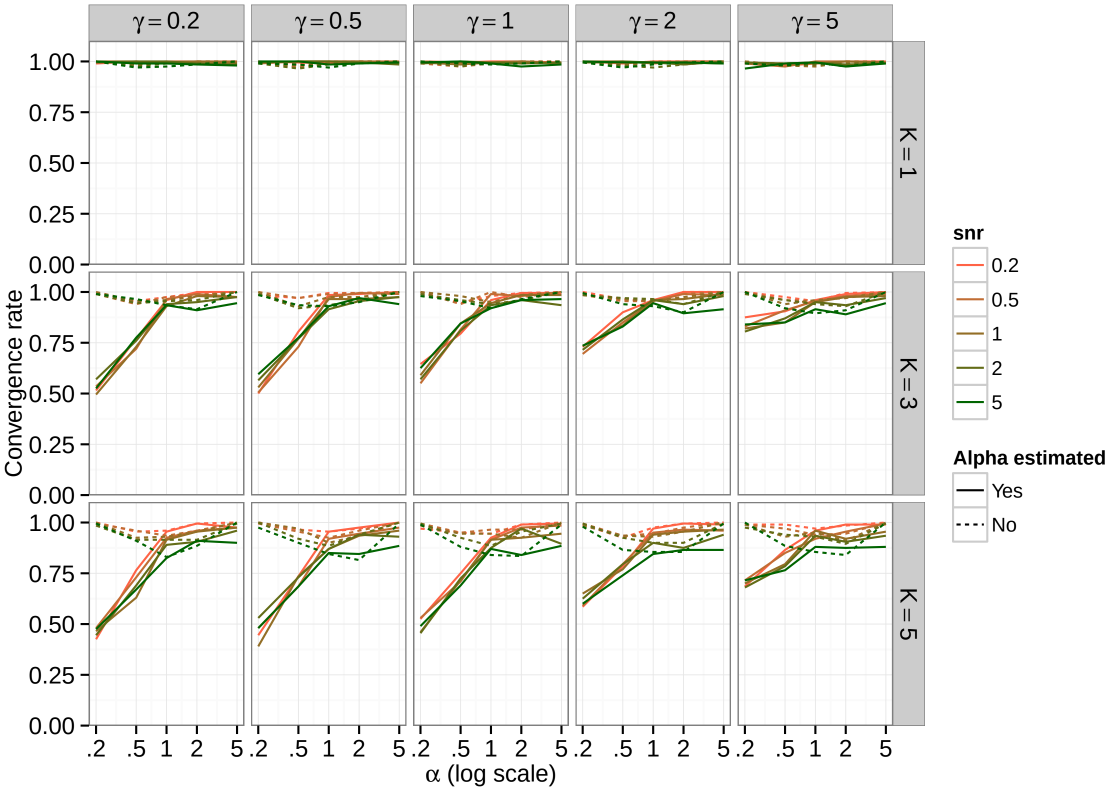


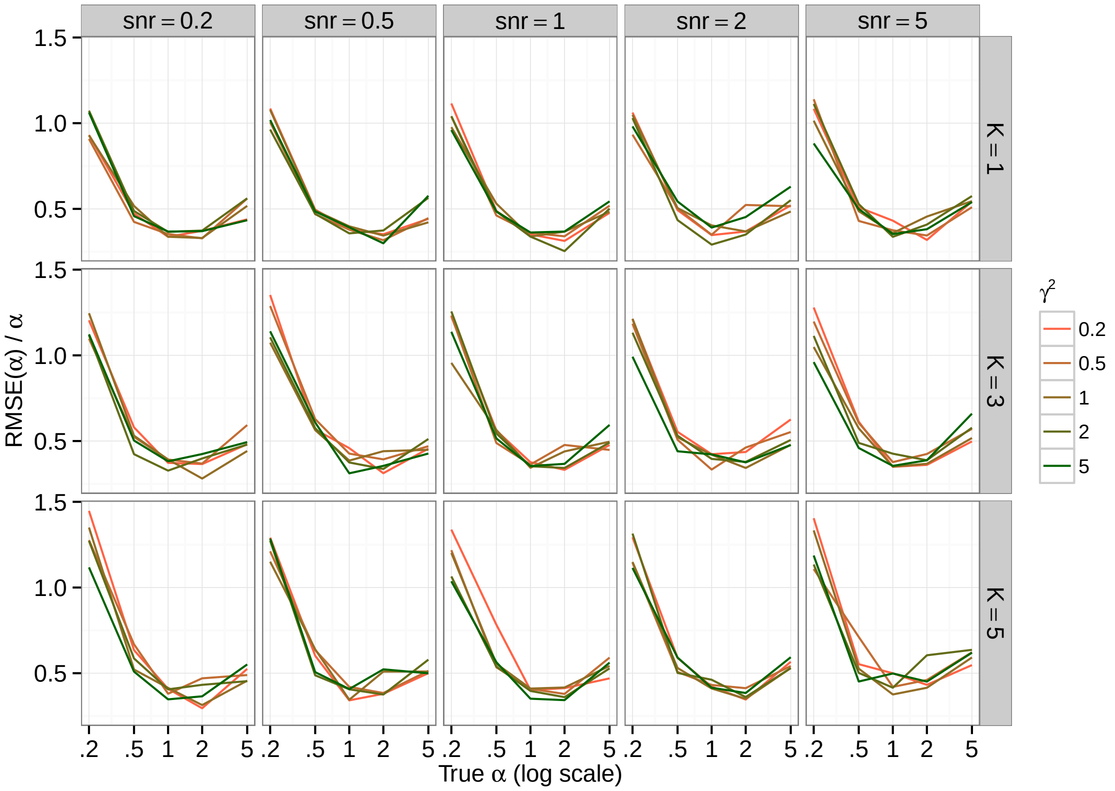
Figure 3.5: Convergence rate for all sets of parameters, mean on the n realizations.

The first observation is that the EM does not converge in every situation. If the maximum number of iteration fixed (1000) is rarely attained, there are numerous situation where one or several parameters diverge and go to infinity, especially in the case where α is supposed unknown and estimated form the data. A bound on the parameters was fixed in order to stop the algorithm in case of parameters obviously wrong, and to save some unnecessary computations. The algorithm is said to be *divergent* if one of these bound is attained. It is said to be *convergent* if the parameters vary less than 10^{-3} between the two last iterations (shutoff condition). In our setting, the algorithm converges all the times it does not diverge. We show figure 3.5 the variations of the convergence rate of the algorithm with the parameters. We can see that small α and snr, and large K, seems to increase the divergence rate, that can be as high as 50% when α is supposed unknown. This divergent behavior remains unexplained, and might be due to numerical or implementation errors, but can also result from the lack of identifiability of our model. Further investigation on this aspect of the question is needed, but we can already see that a good estimation of α is central for the convergence of the algorithm, as high divergence rates correspond to high RMSE on α, as shown figure 3.6a.

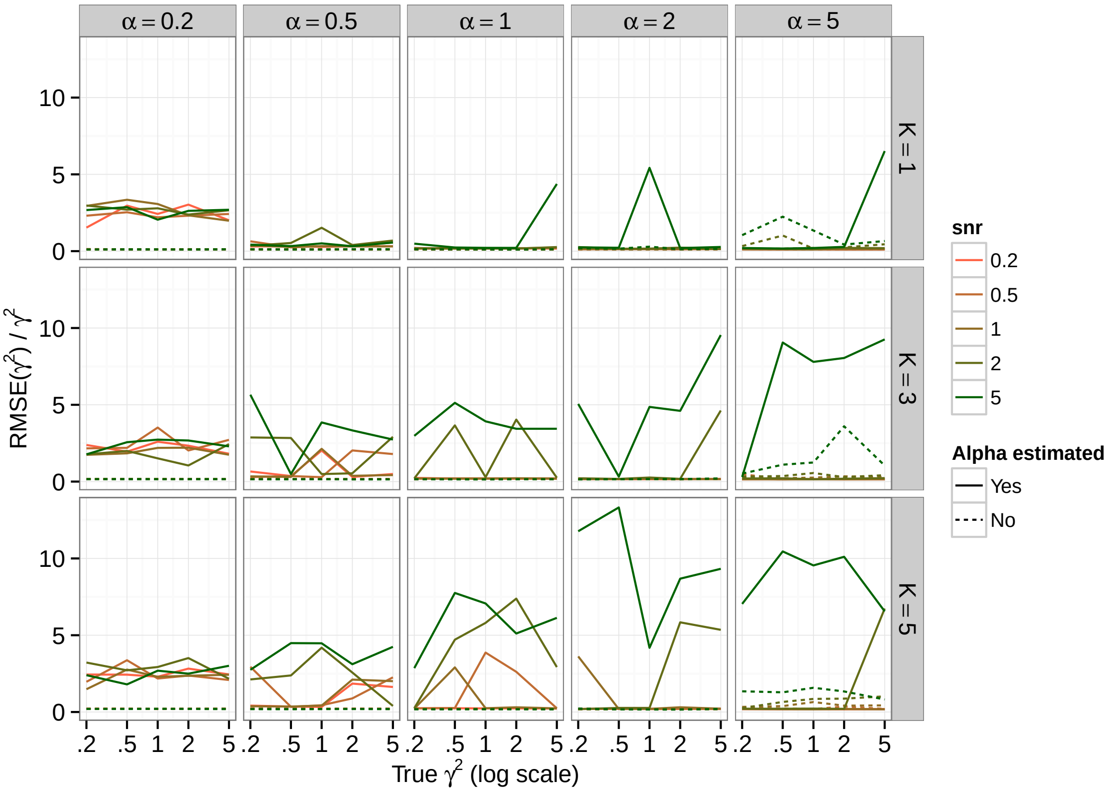
In the case where the EM converge, we can look at the quality of the estimation of the parameters. Figure 3.6 shows the relative Root Mean Squared Errors (RMSE) on the estimations of α and γ^2. We observe that contrary to our intuition, when α is supposed unknown, RMSE are smaller on α than on γ^2. The best estimated values of α seems to be for α in the middle of our range of variation, around 1. The variations of the RMSE on γ^2 seem more hieratic. When α is unknown, relative RMSE on γ^2 is between 1 and 3 most of the time, but it has several outliers points that go up to almost 10 for large values of snr and α. When α is known, the estimation of γ^2 is much better, with a relative RMSE comprised between 0.1 and 0.2 most of the time.

The quality of the estimation of the allocation of shifts is represented figure 3.7. Figure 3.7a, we show the Adjusted Rand Index for initial and final estimations with α unknown. Our first observation is that the lasso initialization seems to do better than the whole EM algorithm, that has a tendency of "messing up" rather good initial approximations. This behavior is problematic, and calls for re-evaluation of our cost-based procedure of evaluation of shifts positions at the M step of the algorithm. Apart from that, we can see that the global edge recovery rate is very low (less than 40%), and unsurprisingly rises with the snr and with α. Figure 3.7b shows distances on the tree between true and estimated positions of the shifts, as defined previously, for estimations with α known or unknown. We can see here that knowing α does not improve substantially our reconstruction of the shifts. This reconstruction is, as expected, better for small values of K and large values of the snr.

Finally, on figure 3.8, we show the evolution of the Adjusted Rand Index with the Mahalanobis distance. The loess regression confirm the role of the difficulty: the inference of the clusters is better for large Mahalanobis distances. We can also see that our experimental setting produces many configurations with a rather small Mahalanobis distance, and therefor hard to infer. This might partly be due to the way we selected the branches where we put shifts, using a uniform random sampling. In a binary tree, half of the branches are terminal and lead to leaves, so we selected many unfavorable configuration where a shift affects only a single species. A weighted sampling (according to the number of descendants of a branch, for instance), could here be more appropriate.

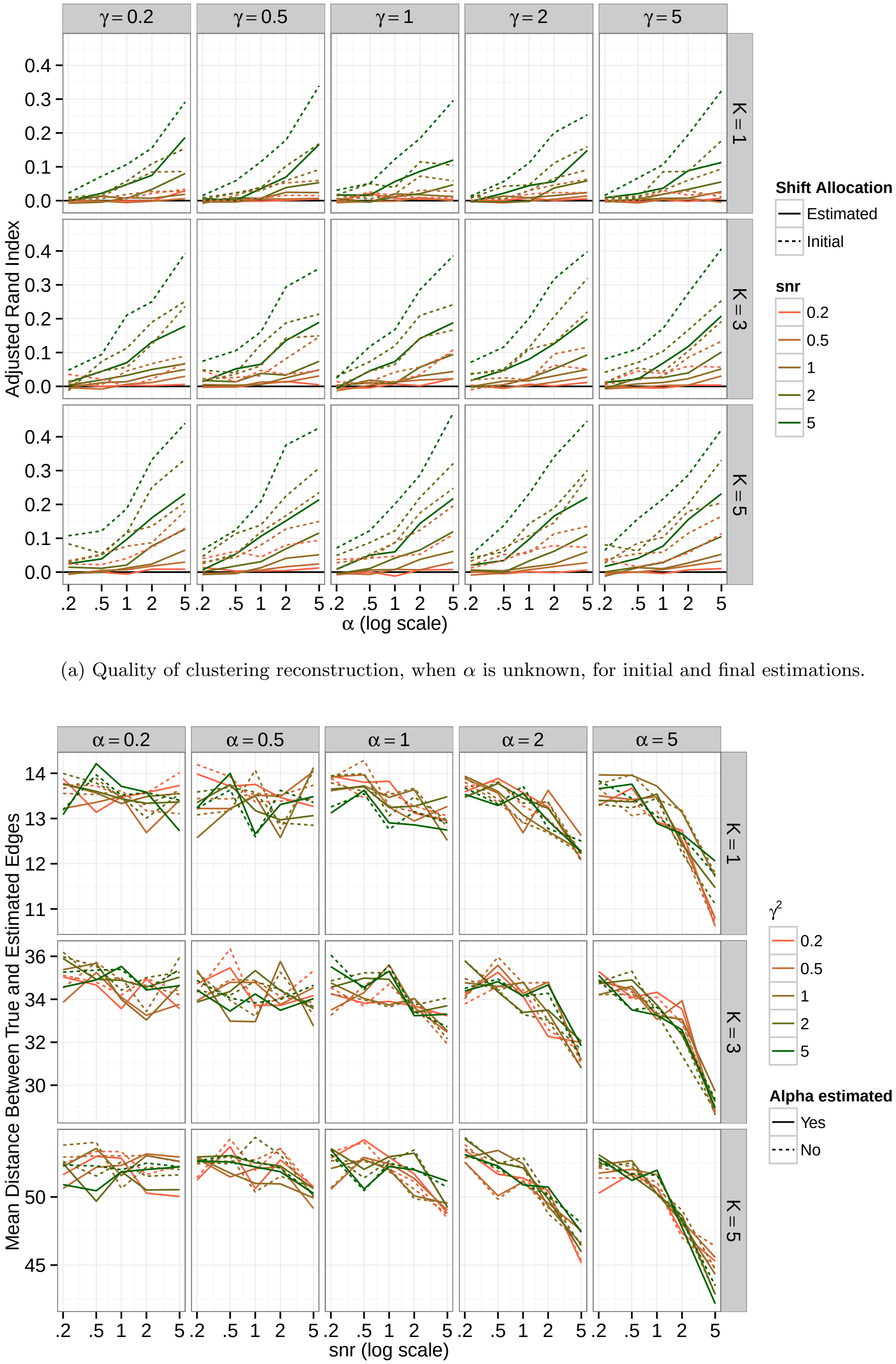


(a) Relative RMSE on α.



(b) Relative RMSE on γ.

Figure 3.6: Relative RMSE of the estimations of parameters, for EM that converged.



(a) Quality of clustering reconstruction, when α is unknown, for initial and final estimations.

(b) Distances on the tree. Note that the scale changes for the different values of K.

Figure 3.7: Quality of estimated shifts allocations.

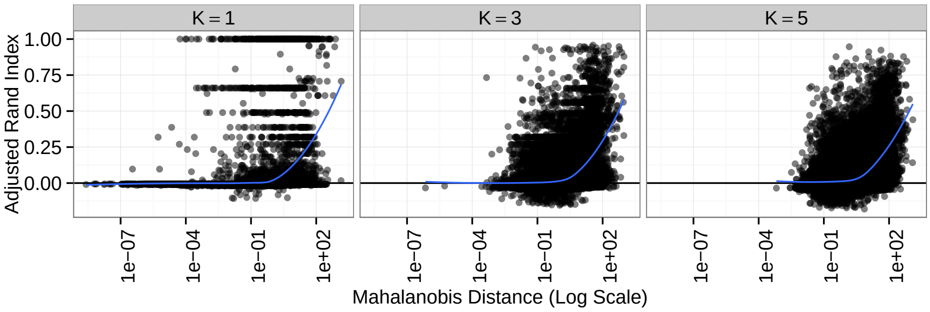


Figure 3.8: Evolution of the ARI with the Mahalanobis Distance for the 75000 estimation. Scatter plot (black dots) and loess regression (blue curve).

3.4.5 Test on a Small Dataset Issued from Butler and King

In [BK04], the authors used a small dataset of 23 antillean anolis lizards. The trait considered is the body size, and the phylogeny used is issued from literature. This dataset is freely available with R package ouch [KB09], developed by the authors of the article. Using a known clustering of the species into three groups ("small", "medium" and "large"), the authors tested several configurations for the allocations of selective regimes on the internal branches of the tree, to find that the model based on a linear parsimony allocation was the most statistically relevant.

We applied our algorithm to this same dataset to compare our results with theirs. First, note that our problem is different from theirs, as we do not impose an a priori allocation of the selective regimes, which in that case is equivalent to impose an a priori clustering of the tips. That means that, rather than trying to find a plausible explanation for a biologically constructed clustering, we try to produce a clustering coherent with the tree structure using uniquely the trait values of the tips. We tried to fit an OU model with a stationary root and 0 to 4 shifts. All the configurations diverged, except when imposing 2 shifts. For this configuration, the initial robust initialization of α failed, so it was initialized to a default value of 1. We found $\beta_0 = 3.12$, $\gamma^2 = 0.94$, $\sigma^2 = 0.019$ (BK: 0.22), $\alpha = 0.01$ (BK: 2.49), $\delta_1 = -47.44$ and $\delta_2 = 139.10$. These surprisingly large values for the shifts can be explained by the very low value of α: as explained in a remark at the end of section 1.4.2, when α is low, the effect of the environment on the mean value of the process is low when the total time is fixed, so that the change must be high to be taken into account. The position of the shifts (see figure 3.9) are coherent with the data. The values of parameters σ^2 and α are quite different form the ones found in [BK04]. Overall, these results, and the fact that the algorithm diverged in many case, show that our method is not really adapted for this small dataset, and that some work can still be done on it.

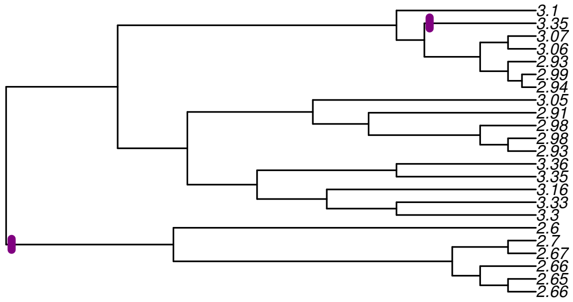


Figure 3.9: Dataset of the antillean anolis lizards. Values on the tips are log of body sizes. The two shifts found are showed in purple.

Conclusion

We present in this work a probabilistic framework to detect adaptive events on a tree. The framework is shared with previous works and use an Orstein-Uhlenbeck process to model the evolution of a functional traits across time in related species. Adaptive events are modeled as shifts of the parameters of the Orstein-Uhlenbeck process. This mathematical formulation of the problem enables us to study the model from a theoretical point of view, which gives us some insights on the phenomena studied, while raising some new questions. It also suggests an iterative, EM-like, estimation procedure.

The model is not identifiable in general: the non-identifiability arises from the locations of shifts on the tree, and some constraints are required for practical inference. Assuming that shifts occur in a parsimonious way is a natural way to avoid over-parametrization but is not sufficient to ensure identifiability. Many allocations of shifts are equivalent from a statistical point of view: they lead to the same probability distribution of the observations at the tips of the tree. We devised a linear recursive algorithm to count the cardinal of each equivalence class, but we still lack a set of constraints to uniquely choose a single representative for each class. This is necessary to ensure that iterative procedures like our EM algorithm does not waste time cycling through equivalent locations. An approach leveraging the kernel of the tree structure matrix T could lead to results in that direction.

For the sake of simplicity, we made several assumptions in our model. We took the root to be at the stationary state and all shifts to occur immediately after a speciation event. The first assumption could be very wrong in some real biological situations, and needs to be studied in more details. An alternative would be to consider X_1 as an additional parameter and work conditionally on X_1. The second assumption is not limiting for ultrametric tree, as a shift can then be moved anywhere on its branch if changing its intensity accordingly. This is however not the case for general trees. Relaxing the shift position along a branch would introduce new parameters that could lead to new identifiability issues, not directly related to the discrete component of the model.

We considered K known throughout this work to avoid the model selection question. Choosing K appropriately is however important for our model and call for further investigations. Under the assumption of infinite site model, we derived a linear algorithm to compute the complexity of a collection of models with a fixed number K of shifts. The algorithm can be bypassed for binary trees as the complexity depends only on n and K. This complexity is the first step towards a model selection criterion. We note however that the infinite site model is an extreme case of parsimonious allocations of the shifts where K shifts always lead to $K + 1$ group. Additional work is necessary to compute or bound the complexity of the collection of parsimonious model with K shifts, when derived states can occur more than once.

We implemented the Expectation-Maximization algorithm to infer the parameters of the model and detect adaptive events. We also assessed the performance of the procedure using simulation studies. For Orstein-Uhlenbeck process, the simulations show that the selection strength α is critical: it is difficult to estimate and essential for the convergence of the algorithm. They also revealed regions of the parameter space where the algorithm does not converge. This suggests that our parametrization of the model may be inadequate and lead the EM algorithm to wander forever on an isocline of the likelihood. Use of biological knowledge and/or another parametrization of the model could mitigate the problem. We explored different initializations of the algorithm, but further exploration is required, in particular for α. Finally and to our surprise, the lasso-based initialization of the shifts allocation seems to do a better job in term of clustering than the whole EM algorithm. This calls for further investigations and suggests that reframing the problem as a linear regression might be more efficient.

In this work, we only considered a one dimensional trait evolving on a tree and faced some theoretical and computational difficulties. In parallel to solving those difficulties, we plan to extend the model to multivariate traits. This extension comes with its own challenges (like the choice of the correlation structure of the traits and the question of the simultaneity of the shifts) but is more realistic. It also would allow us to take more information into account for the clustering of the species. In addition to validating our method through simulations, we must validate it on biological datasets: detected shifts could be compared with documented ecological ruptures that are known to have affected some living species. It could then be applied to others datasets to gain some insights into the evolutionary history of sets of species. In the particular case of bacteria, a working algorithm would help define operational units, coherent both in terms of environment and phylogeny, and therefore easier to use as bioindicators.

Appendix A

Two Generalized Vandermonde Identities

A.1 Statement of the Identities

Proposition 5. *Let* $(n, n') \in N$ *and* $K \in \mathbb{N}$*. With the standard convention that* $\binom{n}{i} = 0$ *if* $n < i$*,*

$$\binom{n+n'-K}{K} = \sum_{i=0}^{K} \binom{n-i}{i}\binom{n'-K+i}{K-i} + \sum_{i=0}^{K-1} \binom{(n-1)-i}{i}\binom{(n'-1)-(K-1)+i}{(K-1)-i} \tag{A.1}$$

which can be rewritten in a more symmetric way as:

$$\binom{n+n'-K}{K} = \sum_{k,k'\geq 0:k+k'=K} \binom{n-k}{k}\binom{n'-k'}{k'} + \sum_{k,k'\geq 0:k+k'=K-1} \binom{(n-1)-k}{k}\binom{(n'-1)-k'}{k'}$$

Similarly,

$$\binom{n+n'+1-K}{K} = \sum_{i=0}^{K} \binom{n-i}{i}\binom{n'-K+i}{K-i} + \sum_{i=0}^{K-1} \binom{(n-1)-i}{i}\binom{n'-(K-1)+i}{(K-1)-i} + \binom{n-i}{i}\binom{(n'-1)-(K-1)+i}{(K-1)-i} \tag{A.2}$$

which can be rewritten in a more symmetric way as:

$$\binom{n+n'+1-K}{K} = \sum_{k,k'\geq 0:k+k'=K} \binom{n-k}{k}\binom{n'-k'}{k'} + \sum_{k,k'\geq 0:k+k'=K-1} \binom{(n-1)-k}{k}\binom{n'-k'}{k'} + \binom{n-k}{k}\binom{(n'-1)-k'}{k'}$$

Note that equation (A.1) generalizes in some way the Vandermonde identity which states

$$\binom{n+n'}{K} = \sum_{k=0}^{K} \binom{n}{k}\binom{n'}{K-k} \tag{A.3}$$

A.2 Proof of the First Identity

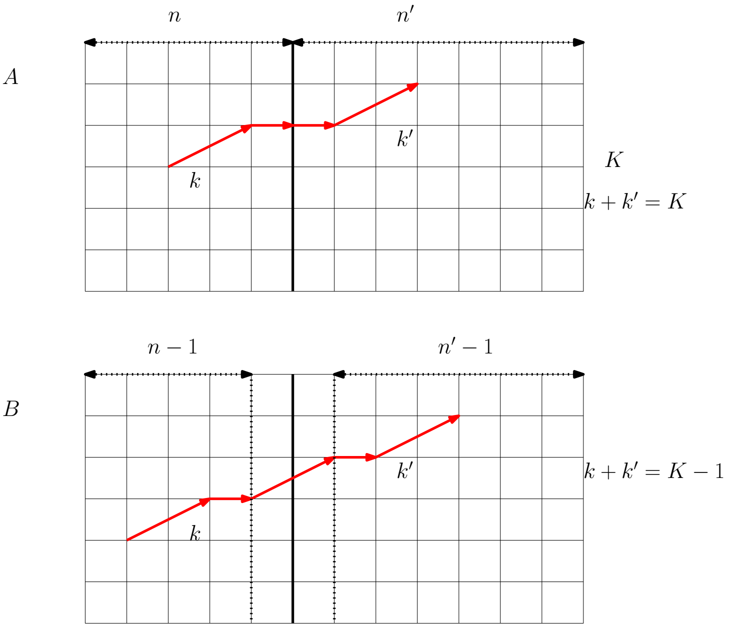


Figure A.1: Partition of paths according to whether they reach (A) or cross (B) the line $x = n$

Although several proofs of the Vandermonde identity are known (geometric, algebraic and combinatorial), we only provide a geometric proof of this Vandermonde-like identity.

Consider a grid of size $(n + n') \times K$ (see figure A.1). We are interested in grid-valued paths that can move either by $(1, 0)$ or by $(2, 1)$. In other words, if the k^{th} position of a path is (x_k, y_k), then its next position (x_{k+1}, y_{k+1}) is either $(x_k + 1, y_k)$ or $(x_k + 2, y_k + 1)$. We are interested in paths starting at $(0, 0)$ and ending at $(n + n', K)$.

Such a path consists of $n + n' - K$ moves : K moves of type $(2, 1)$ and $n + n' - 2K$ moves of type $(1, 0)$. It is uniquely determined by the positions of the moves of the first type. There are $\binom{n+n'-K}{K}$ distinct set of positions and therefore as many such paths.

We now sort the paths according to the value i they take when either reaching the line $x = n$ or reaching the line $x = n + 1$ without reaching the line $x = n$ first. We refer to the latter as crossing the line $x = n$. Note that this sorting induces a partition of all paths.

A path reaching $x = n$ at position i uniquely gives rise to two paths: one from $(0, 0)$ to (n, i) and one from (n, i) to $(n + n', K)$ or equivalently from 0 to $(n', K - i)$. There are $\binom{n-i}{i}$ different paths of the first kind and $\binom{n'-K-i}{K-i}$ of the second. There are therefore $\binom{n-i}{i}\binom{n'-K+i}{K-i}$ paths that pass through (n, i).

A path crossing the line $x = n$ and reaching the line $x = n + 1$ at i must do so with a last move of type $(2, 1)$. It therefore uniquely defines a path from $(0, 0)$ to $(n - 1, i - 1)$ and a path from $(n + 1, i)$ to $(n + n', K)$, or equivalently from $(0, 0)$ to $(n' - 1, K - i)$. There are therefore $\binom{n-i}{i-1}\binom{n'-1-K+i}{K-i}$ paths that cross the line $x = n$ and pass through $(n + 1, i)$.

Putting everything together, we get:

$$\begin{aligned}\binom{n+n'-K}{K} &= \sum_{i=0}^{K}\binom{n-i}{i}\binom{n'-K+i}{K-i} + \sum_{i=0}^{K}\binom{n-i}{i-1}\binom{n'-1-K+i}{K-i} \\ &= \sum_{i=0}^{K}\binom{n-i}{i}\binom{n'-K+i}{K-i} + \sum_{i=0}^{K-1}\binom{(n-1)-i}{i}\binom{(n'-1)-(K-1)+i}{(K-1)-i}\end{aligned}$$

which is exactly equation (A.1).

A.3 Proof of the Second Identity

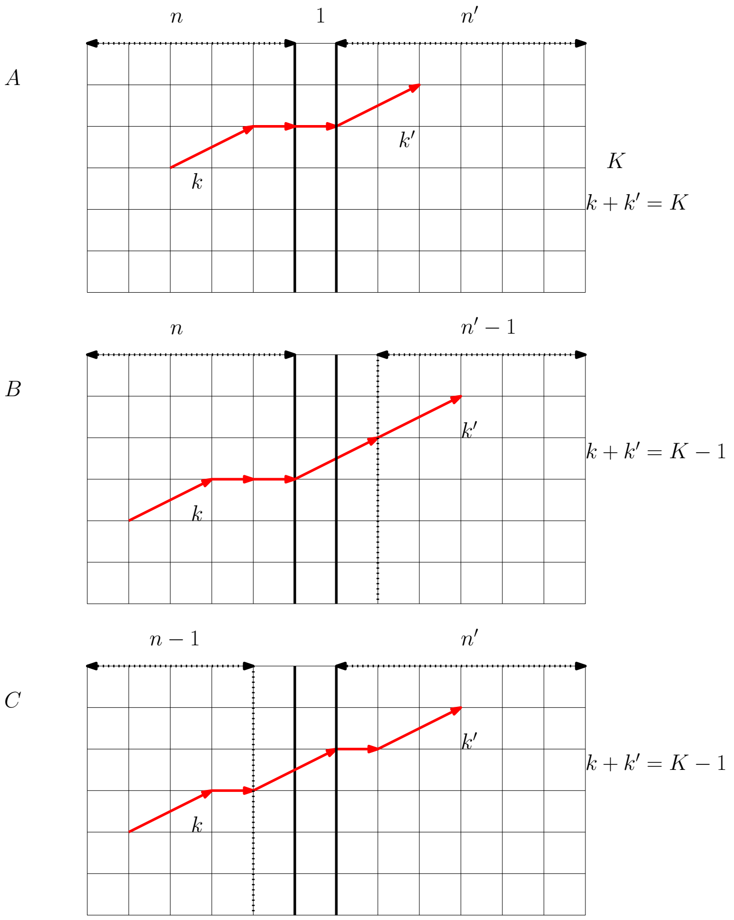


Figure A.2: Partition of paths according to whether to the move used between $x = n$ and $x = n + 1$. Cases A, B and C correspond to the items listed in the main text.

To prove equation (A.2), we start from a grid of size $(n+n'+1) \times K$ and are again interested in the paths starting from the bottom left corner and ending in the upper right corner using only $(2, 1)$ and $(1, 0)$ moves. These paths have exactly K moves of type $(2, 1)$ and there are $\binom{n+n'+1-K}{K}$ of them. This time, we partition paths upon the move observed between $x = n$ and $x = (n + 1)$.

The move can be (see figure A.2):

- $(1, 0)$: then k moves of type $(2, 1)$ are used in the interval $[1, n]$, and k' in the interval $[n + 1, n + n' + 1]$, with $k + k' = K$. There are $\sum_{k+k'=K} \binom{n-k}{k}\binom{n'-k'}{k'}$ such paths.
- $(2, 1)$ starting from $x = n$ and therefore ending at $x = n + 2$: then k moves of type $(2, 1)$ are used in the interval $[1, n]$, and k' in the interval $[n+2, n+n'+1]$, with $k+k' = K-1$ (one move $(2, 1)$ has already been used). There are $\sum_{k+k'=K-1} \binom{n-k}{k}\binom{(n'-1)-k'}{k'}$ such paths.
- $(2, 1)$ ending at $x = n + 1$ and therefore starting from $x = n - 1$: then k moves of type $(2, 1)$ are used in the interval $[1, n - 1]$ and k' in the interval $[n + 1, n + n'1]$, with $k + k' = K - 1$. There are $\sum_{k+k'=K-1} \binom{(n-1)-k}{k}\binom{n'-k'}{k'}$ such paths.

Putting everything together, we get equation (A.2) as wanted.

Appendix B

Upward-Downward Algorithm

Here we describe a “forward-backward like” algorithm, that is adapted to the tree structure of our problem, and exploits the Gaussian properties of the process. This algorithm will work for any such process, and in particular for the Brownian Motion and Orstein-Uhlenbeck process.

Lets take the following general notations:

$$\forall j \in [\![2\,, m+n]\!], \begin{cases} \mathbb{E}\left[X_j \;\middle|\; X_{\text{pa}(j)}\right] = m_j(X_{\text{pa}(j)}) = q_j X_{\text{pa}(j)} + r_j \\ \mathbb{V}\left[X_j \;\middle|\; X_{\text{pa}(j)}\right] = \sigma_j^2 \end{cases}$$

Note that for the Brownian Motion:

$$\begin{cases} q_j = 1 \\ r_j = \sum_k \mathbb{I}\{\tau_k = b_j\}\delta_k \\ \sigma_j^2 = \ell_j \sigma^2 \end{cases}$$

And for the Orstein-Uhlenbeck:

$$\begin{cases} q_j = e^{-\alpha\ell_j} \\ r_j = \beta^{\text{pa}(j)}(1 - e^{-\alpha\ell_j}) + \sum_k \mathbb{I}\{\tau_k = b_j\}\delta_k \left(1 - e^{-\alpha(1-\nu_k)\ell_j}\right) \\ \sigma_j^2 = \dfrac{\sigma^2}{2\alpha}(1 - e^{-2\alpha\ell_j}) \end{cases}$$

B.1 Basic Identities on Gaussian Densities

For $(m, s) \in \mathbb{R}^2$, denote $\Phi_{m,s^2} : x \mapsto \frac{1}{\sqrt{2\pi s^2}} \exp\left(-\frac{(x-m)^2}{2s^2}\right)$ the Gaussian density.
Remark that $\forall (m_1, m_2, s^2) \in \mathbb{R}^3, \Phi_{m_1,s^2}(m_2) = \Phi_{m_2,s^2}(m_1)$

Proposition 6 (Product of two Gaussian densities)**.** *Let $(m_1, m_2, s_1, s_2, x) \in \mathbb{R}^5$. Then*

$$\Phi_{m_1,s_1}(x)\Phi_{m_2,s_2}(x) = \frac{1}{\sqrt{2\pi(s_1^2 + s_2^2)}} \exp\left(-\frac{(m_1 - m_2)^2}{2(s_1^2 + s_2^2)}\right) \Phi_{\bar{m}_{1,2},\bar{s}_{1,2}^2}(x) \tag{B.1}$$

Where

$$\bar{s}_{1,2}^2 = \left(\frac{1}{s_1^2} + \frac{1}{s_2^2}\right)^{-1} \quad \textit{and} \quad \bar{m}_{1,2} = \bar{s}_{1,2}^2 \left(\frac{m_1}{s_1^2} + \frac{m_2}{s_2^2}\right)$$

In particular:

$$\int_{\mathbb{R}} \Phi_{m_1,s_1}(x)\Phi_{m_2,s_2}(x) = \Phi_{m_1,s_1^2+s_2^2}(m_2) = \Phi_{m_2,s_1^2+s_2^2}(m_1) \tag{B.2}$$

Proposition 7 (Product of L Gaussian densities)**.** *Let* $(m_1, \dots, m_L) \in \mathbb{R}^L$, $(s_1, \dots, s_L) \in \mathbb{R}^L$ *and* $x \in \mathbb{R}$. *Then*

$$\prod_{l=1}^{L} \Phi_{m_l, s_l}(x) = (2\pi)^{-(L-1)/2} \sqrt{\frac{\bar{s}^2_{1:L}}{\prod_{l=1}^{L} s_l^2}} \Phi_{\bar{m}_{1:L}, \bar{s}^2_{1:L}}(x) \tag{B.3}$$

Where

$$\bar{s}^2_{1:L} = \left(\sum_{l=1}^{L} \frac{1}{s_l^2} \right)^{-1} \quad \text{and} \quad \bar{m}_{1:L} = \bar{s}^2_{1:L} \sum_{l=1}^{L} \frac{m_l}{s_l^2}$$

This other lemma will be useful:

Lemma 2 (Two dimensional Gaussian Density)**.** *Let* $\begin{pmatrix} X \\ Y \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} m_X \\ m_Y \end{pmatrix}, \begin{pmatrix} \sigma^2_{XX} & \sigma^2_{XY} \\ \sigma^2_{XY} & \sigma^2_{YY} \end{pmatrix} \right)$ *Then, by Gaussian properties:* $Y|X \sim \mathcal{N}(m_{Y|X}, \sigma_{Y|X})$, *where*

$$m_{Y|X} = m_Y + \sigma^2_{XY}(\sigma^2_{XX})^{-1}(X - m_X), \qquad \sigma_{Y|X} = \sigma^2_{YY} - \sigma^4_{XY}(\sigma^2_{XX})^{-1}$$

B.2 Upward

For each node X_j of the tree, denote by $\mathbf{Y}^j$ the vector of all the tips that are below it. We are going to compute recursively $f_{\mathbf{Y}^j|X_j}(\mathbf{Y}^j; a)$ the Gaussian density function of $\mathbf{Y}^j \mid X_j$. To do that, we write $f_{\mathbf{Y}^j|X_j}(\mathbf{Y}^j; a)$ as proportional to a gaussian density in a:

$$f_{\mathbf{Y}^j|X_j}(\mathbf{Y}^j; a) = A_j(\mathbf{Y}^j) \Phi_{M_j(\mathbf{Y}^j), S_j^2(\mathbf{Y}^j)}(a)$$

Initialization

$$\forall i \in [\![1\,, n]\!], f_{Y_i|Y_i}(Y_i; a) = \Phi_{Y_i, 0}(a) = \mathbb{I}\{Y_i = a\}$$

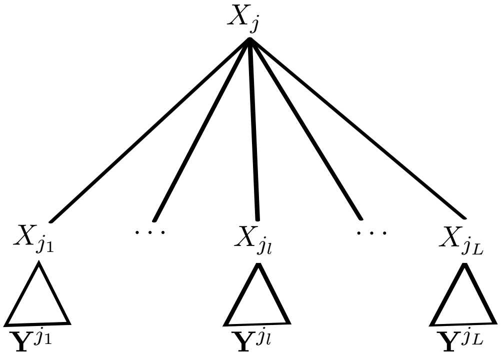


Figure B.1: A parent node with L daughters, each above several tips $\mathbf{Y}^{j_l}$

Propagation Suppose that we know $f_{\mathbf{Y}^{j_l}|X_{j_l}}(\mathbf{Y}^{j_l}; a)$ for all the L daughters of a node X_j (see figure B.1). Then by conditional independence of the daughters knowing the mother, we get:

$$f_{\mathbf{Y}^j|X_j}(\mathbf{Y}^j; a) = \prod_{l=1}^{L} f_{\mathbf{Y}^{j_l}|X_j}(\mathbf{Y}^{j_l}; a)$$

And, for $1 \leq l \leq L$:

$$\begin{aligned} f_{\mathbf{Y}^{j_l}|X_j}(\mathbf{Y}^{j_l}; a) &= \int_{\mathbb{R}} f_{\mathbf{Y}^{j_l}|X_{j_l}}(\mathbf{Y}^{j_l}; b) f_{X_{j_l}|X_j}(b; a) db \\ &= A_{j_l}(\mathbf{Y}^{j_l}) \int_{\mathbb{R}} \Phi_{M_{j_l}(\mathbf{Y}^{j_l}), S^2_{j_l}(\mathbf{Y}^{j_l})}(b) \Phi_{m_{j_l}(a), \sigma^2_{j_l}}(b) db \end{aligned}$$

as $X_{j_l} \mid X_j \sim \mathcal{N}\left(m_{j_l}(X_j), \sigma^2_{j_l}\right)$. From proposition 6 (equation (B.2)), we get:

$$\begin{aligned}
f_{\mathbf{Y}^{j_l}|X_j}(\mathbf{Y}^{j_l}; a) &= \frac{A_{j_l}(\mathbf{Y}^{j_l})}{\sqrt{2\pi(S^2_{j_l}(\mathbf{Y}^{j_l}) + \sigma^2_{j_l})}} \exp\left(-\frac{(m_{j_l}(a) - M_{j_l}(\mathbf{Y}^{j_l}))^2}{2(S^2_{j_l}(\mathbf{Y}^{j_l}) + \sigma^2_{j_l})}\right) \\
&= \frac{A_{j_l}(\mathbf{Y}^{j_l})}{\sqrt{2\pi(S^2_{j_l}(\mathbf{Y}^{j_l}) + \sigma^2_{j_l})}} \exp\left(-\frac{\left(a - \frac{M_{j_l}(\mathbf{Y}^{j_l}) - r_{j_l}}{q_{j_l}}\right)^2}{2\frac{S^2_{j_l}(\mathbf{Y}^{j_l}) + \sigma^2_{j_l}}{q^2_{j_l}}}\right) \\
&= \frac{A_{j_l}(\mathbf{Y}^{j_l})}{q_{j_l}} \Phi_{\frac{M_{j_l}(\mathbf{Y}^{j_l}) - r_{j_l}}{q_{j_l}}, \frac{S^2_{j_l}(\mathbf{Y}^{j_l}) + \sigma^2_{j_l}}{q^2_{j_l}}}(a)
\end{aligned}$$

And then, from proposition 7:

$$f_{\mathbf{Y}^j|X_j}(\mathbf{Y}^j; a) = A_j(\mathbf{Y}^j)\Phi_{M_j(\mathbf{Y}^j), S^2_j(\mathbf{Y}^j)}(a)$$

with

$$\begin{cases}
S^2_j(\mathbf{Y}^j) = \left(\displaystyle\sum_{l=1}^{L} \frac{q^2_{j_l}}{S^2_{j_l}(\mathbf{Y}^{j_l}) + \sigma^2_{j_l}}\right)^{-1} \\
M_j(\mathbf{Y}^j) = S^2_j(\mathbf{Y}^j) \displaystyle\sum_{l=1}^{L} q_{j_l} \frac{M_{j_l}(\mathbf{Y}^{j_l}) - r_{j_l}}{S^2_{j_l}(\mathbf{Y}^{j_l}) + \sigma^2_{j_l}} \\
A_j(\mathbf{Y}^j) = (2\pi)^{-(L-1)/2} \sqrt{S^2_j(\mathbf{Y}^j)} \displaystyle\prod_{l=1}^{L} \frac{A_{j_l}(\mathbf{Y}^{j_l})}{\sqrt{S^2_{j_l}(\mathbf{Y}^{j_l}) + \sigma^2_{j_l}}}
\end{cases} \tag{B.4}$$

Root Node and Likelihood Once at the root, we have $f_{\mathbf{Y}|X_1}(\mathbf{Y}; a)$, which is the likelihood function of the observation given the root state, and we write:

$$f_{X_1|\mathbf{Y}}(a; \mathbf{Y}) \propto f_{\mathbf{Y}|X_1}(\mathbf{Y}; a) f_{X_1}(a)$$

And, by proposition 6:

$$\begin{cases}
\mathbb{V}\left[X_1 \mid \mathbf{Y}\right] = \left(\frac{1}{\gamma^2} + \frac{1}{S^2_1(\mathbf{Y})}\right)^{-1} \\
\mathbb{E}\left[X_1 \mid \mathbf{Y}\right] = \mathbb{V}\left[X_1 \mid \mathbf{Y}\right]\left(\frac{\mu}{\gamma^2} + \frac{M_1(\mathbf{Y})}{S^2_1(\mathbf{Y})}\right)
\end{cases} \tag{B.5}$$

These are the first two quantities needed.

B.3 Downward

Going down the tree, we need to compute, for each node X_j, $2 \leq j \leq m$:

$$\begin{cases}
E_j = \mathbb{E}\left[X_j \mid \mathbf{Y}\right] \\
V^2_j = \mathbb{V}\left[X_j \mid \mathbf{Y}\right] \\
C^2_{j,\text{pa}(j)} = \mathbb{C}\text{ov}\left[X_j ; X_{\text{pa}(j)} \;\middle|\; \mathbf{Y}\right]
\end{cases}$$

Initialization The initialization of the downward is given by the last step of the upward.

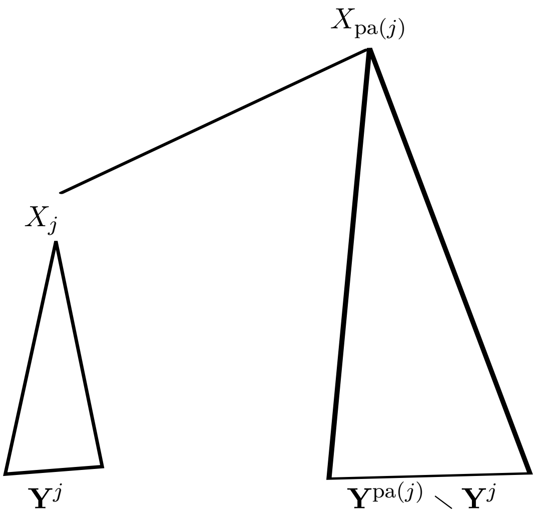


Figure B.2: A node with its parent and daughter tips.

Propagation Let $2 \leq j \leq m$ (see figure B.2)

We have:

$$f_{X_{\mathrm{pa}(j)},X_j|\mathbf{Y}}(a,b;\mathbf{Y}) = f_{X_{\mathrm{pa}(j)}|\mathbf{Y}}(a;\mathbf{Y}) f_{X_j|X_{\mathrm{pa}(j)},\mathbf{Y}}(b;a,\mathbf{Y})$$

We know the first term from the recurrence, and we can compute the second term thanks to the upward step:

$$f_{X_j|X_{\mathrm{pa}(j)},\mathbf{Y}}(b;a,\mathbf{Y}) = f_{X_j|X_{\mathrm{pa}(j)},\mathbf{Y}^j}(b;a,\mathbf{Y}^j) \propto f_{X_j|X_{\mathrm{pa}(j)}}(b;a) f_{\mathbf{Y}^j|X_j}(\mathbf{Y}^j;b)$$

As $\mathbf{Y}^j \mid X_j \sim \mathcal{N}\left(M_j(\mathbf{Y}^j), S_j^2(\mathbf{Y}^j)\right)$ and $X_j \,\Big|\, X_{\mathrm{pa}(j)} \sim \mathcal{N}\left(m_j(X_{\mathrm{pa}(j)}), \sigma_j^2\right)$, we get, from proposition 6:

$$X_j \,\Big|\, X_{\mathrm{pa}(j)}, \mathbf{Y} \sim \mathcal{N}\left(\bar{m}_j(X_{\mathrm{pa}(j)}), \bar{\sigma}_j^2\right)$$

with

$$\begin{cases} \bar{\sigma}_j^2 = \left(\dfrac{1}{S_j^2(\mathbf{Y}^j)} + \dfrac{1}{\sigma_j^2}\right)^{-1} = \dfrac{S_j^2(\mathbf{Y}^j)\sigma_j^2}{S_j^2(\mathbf{Y}^j) + \sigma_j^2} \\ \bar{m}_j(X_{\mathrm{pa}(j)}) = \bar{\sigma}_j^2 \left(\dfrac{M_j(\mathbf{Y}^j)}{S_j^2(\mathbf{Y}^j)} + \dfrac{m_j(X_{\mathrm{pa}(j)})}{\sigma_j^2}\right) = \underbrace{\dfrac{q_j S_j^2(\mathbf{Y}^j)}{S_j^2(\mathbf{Y}^j) + \sigma_j^2}}_{\bar{q}_j} X_{\mathrm{pa}(j)} + \underbrace{\dfrac{S_j^2(\mathbf{Y}^j) r_j + \sigma_j^2 M_j(\mathbf{Y}^j)}{S_j^2(\mathbf{Y}^j) + \sigma_j^2}}_{\bar{r}_j} \end{cases}$$

Hence:

$$f_{X_j|X_{\mathrm{pa}(j)},\mathbf{Y}}(b;a,\mathbf{Y}) \propto \exp\left(-\frac{(b - \bar{m}_j(a))^2}{2\bar{\sigma}_j^2}\right)$$

And, from lemma 2:

$$\begin{cases} \bar{m}_j(a) = E_j - \dfrac{C_{j,\mathrm{pa}(j)}^2}{V_{\mathrm{pa}(j)}^2}(a - E_{\mathrm{pa}(j)}) \\ \bar{\sigma}_j^2 = V_j^2 - \dfrac{C_{j,\mathrm{pa}(j)}^4}{V_{\mathrm{pa}(j)}^2} \end{cases}$$

From this we get:

$$\begin{cases} C_{j,\mathrm{pa}(j)}^2 = \bar{q}_j V_{\mathrm{pa}(j)}^2 \\ E_j = \bar{r}_j + \bar{q}_j E_{\mathrm{pa}(j)} \\ V_j^2 = \bar{\sigma}_j^2 + \bar{q}_j^2 V_{\mathrm{pa}(j)}^2 \end{cases}$$

And, finally:

$$\begin{cases} C_{j,\mathrm{pa}(j)}^2 = q_j \dfrac{S_j^2(\mathbf{Y}^j)}{S_j^2(\mathbf{Y}^j) + \sigma_j^2} V_{\mathrm{pa}(j)}^2 \\ E_j = \dfrac{S_j^2(\mathbf{Y}^j)(q_j E_{\mathrm{pa}(j)} + r_j) + \sigma_j^2 M_j(\mathbf{Y}^j)}{S_j^2(\mathbf{Y}^j) + \sigma_j^2} \\ V_j^2 = \dfrac{S_j^2(\mathbf{Y}^j)}{S_j^2(\mathbf{Y}^j) + \sigma_j^2} \left(\sigma_j^2 + p_j^2 \dfrac{S_j^2(\mathbf{Y}^j)}{S_j^2(\mathbf{Y}^j) + \sigma_j^2} V_{\mathrm{pa}(j)}^2 \right) \end{cases}$$

B.4 Computational and Memorial Cost

If L is the maximum degree of the tree, then, for the upward, we need to do $O(L)$ basic algebraic operations for the actualization of $S_j^2(\mathbf{Y}^j)$, $M_j(\mathbf{Y}^j)$ and $A_j(\mathbf{Y}^j)$, hence $O(mL)$ operations to get $\mathbb{V}[X_1 \mid \mathbf{Y}]$, $\mathbb{E}[X_1 \mid \mathbf{Y}]$ and the likelihood of the data; and we have to keep $3m$ quantities in memory. For the downward, we need to compute and keep $2m + m + n - 1$ quantities, with $O(m)$ basic algebraic operations.

Finally, we need $O(m)$ operations, and $O(n)$ memory cases.

Bibliography

[Ané08] Cécile Ané. Analysis of Comparative Data with Hierarchical Autocorrelation. *The Annals of Applied Statistics*, 2(3):1078–1102, 2008.

[AW14a] Revolution Analytics and Steve Weston. *doParallel: Foreach parallel adaptor for the parallel package*, 2014. R package version 1.0.8.

[AW14b] Revolution Analytics and Steve Weston. *foreach: Foreach looping construct for R*, 2014. R package version 1.4.2.

[BJBO12] Jeremy M. Beaulieu, Dwueng-Chwuan Jhwueng, Carl Boettiger, and Brian C. O'Meara. Modeling Stabilizing Selection: Expanding The Ornstein–Uhlenbeck Model Of Adaptive Evolution. *Evolution*, 66(8):2369–2383, 2012.

[BK04] Marguerite A. Butler and Aaron A. King. Phylogenetic Comparative Analysis: A Modeling Approach for Adaptive Evolution. *The American Naturalist*, 164(6):pp. 683–695, 2004.

[Cha12] Scott Chasalow. *combinat: combinatorics utilities*, 2012. R package version 0.0-8.

[Fel85] Joseph Felsenstein. Phylogenies and the Comparative Method. *The American Naturalist*, 125(1):pp. 1–15, January 1985.

[Fel04] Joseph Felsenstein. *Inferring Phylogenies.* Sinauer Associates, Suderland, USA, 2004.

[FHT10] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Regularization Paths for Generalized Linear Models via Coordinate Descent. *Journal of Statistical Software*, 334(1):1–22, 2010.

[FRMS12] Chris Fraley, Adrian E. Raftery, T. Brendan Murphy, and Luca Scrucca. mclust Version 4 for R: Normal Mixture Modeling for Model-Based Clustering, Classification, and Density Estimation. Technical report, 2012.

[HA13a] Lam si Tung Ho and Cécille Ané. A Linear-Time Algorithm for Gaussian and Non-Gaussian Trait Evolution Models. *Systematic Biology*, 63(3):397–408, 2013.

[HA13b] Lam si Tung Ho and Cécille Ané. Asymptotic Theory with Hierarchical Autocorrelation : Ornstein-Uhlenbeck Tree Models. *The Annals of Statistics*, 41(2):957–981, 2013.

[HA14] Lam si Tung Ho and Cécile Ané. Intrinsic Inference Difficulties for Trait Evolution with Ornstein-Uhlenbeck Models. 2014.

[KB09] Aaron A. King and Marguerite A. Butler. *ouch: Ornstein-Uhlenbeck models for phylogenetic comparative hypotheses*, 2009. R package version 2.8-4.

[M+11] Robert W. Meredith et al. Impacts of the Cretaceous Terrestrial Revolution and KPg Extinction on Mammal Diversification. *Science*, 334:521–524, October 2011.

[Mas07] Pascal Massart. Concentration Inequalities and Model Selection: Ecole d'Eté de Probabilités de Saint-Flour XXXIII - 2003. 2007.

[PCS04] E. Paradis, J. Claude, and K. Strimmer. APE: analyses of phylogenetics and evolution in R language. *Bioinformatics*, 20:289–290, 2004.

[R C14] R Core Team. *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria, 2014.

[RCT+14] Peter Rousseeuw, Christophe Croux, Valentin Todorov, Andreas Ruckstuhl, Matias Salibian-Barrera, Tobias Verbeke, Manuel Koller, and Martin Maechler. *robustbase: Basic Robust Statistics.*, 2014.

[Wic07] Hadley Wickham. Reshaping Data with the reshape Package. *Journal of Statistical Software*, 21(12):1–20, 2007.

[Wic09] Hadley Wickham. *ggplot2: elegant graphics for data analysis.* Springer New York, 2009.