BACHELOR'S THESIS

Degree in Mathematics

Title: Phylogenetic Reconstruction Based on Algebra

Author: Gustavo Cilleruelo Calderón

Advisor: Marta Casanellas Rius

Department: Matemàtica Aplicada I

Academic year: 2018-2019

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Facultat de Matemàtiques i Estadística

Phylogenetic Reconstruction Based on Algebra

Gustavo Cilleruelo

2019

Contents

Introduction

Charles Darwin first introduced and formalized the concept of evolution in *On the Origin of Species by Means of Natural Selection, or the Preservation of Favoured Races in the Struggle for Life* ([Dar59]), published in 1859. We have come a long way since those first postulates, but biologists still use the representation of the evolutionary relationships among species that was introduced by Darwin: phylogenetic trees (see Figure 1).

Phylogenetic trees are a widely used tool in biology to visualize and establish evolutionary relationships between organisms. Mathematically speaking, they are trees (in the sense of graphs) whose leaves are in bijection with the living species, where the internal nodes represent common ancestors and the branches evolutionary processes.

The reconstruction of evolutionary history of living species was inferred from morphological traits until the eighties. Nowadays, with huge amounts of sequences of molecular data at our disposal, the inference of phylogenetic trees uses DNA or protein sequences as input.

There are several algorithms to reconstruct n-leaved phylogenetic trees given sequences associated to present-day species. Some of them rely on assigning every species an evolutionary distance to the rest, such as the neighbour-joining algorithm. Others use statistical techniques such as assigning a confidence score to a subset of possible trees, such as maximum likelihood.

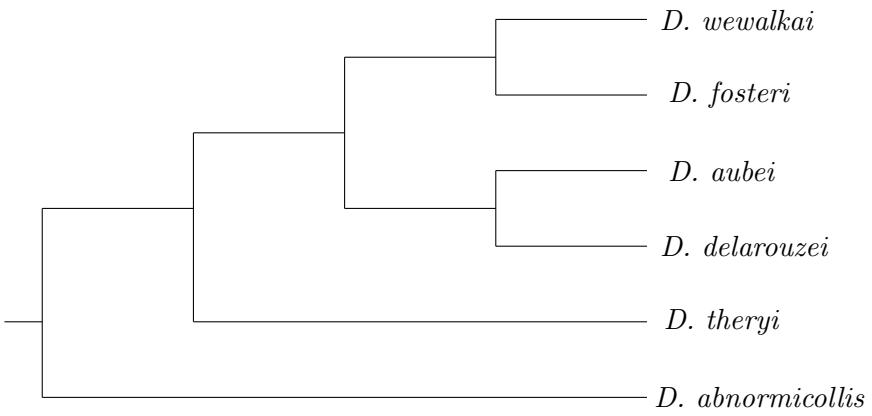


Figure 1: Example of phylogenetic tree for 6 species of the genus *Deronectes*, derived from [RGVB+16].

Although reconstructing 4-leaved phylogenetic trees may seem unnecessary since they are the simplest case for unrooted trees, a whole branch of algorithms has emerged based in quartet reconstruction, see [RG01]. In broad terms, these algorithms compute first the optimal quartets for all 4-tuples of n species (in some cases assigning a weight to each one of them). Then, a tree is reconstructed such that its topology respects the maximum possible amount of quartets, taking into account their weights. This justifies the study of new quartet reconstruction methodologies

and the improvement of accuracy of already existing ones.

Restricting to quartets allows the consideration of probabilistic models of evolution (which turns impossible for trees of more than 20 species). These are Markov processes of nucleotide substitution on the tree. Although the phylogenetic reconstruction based on these models has usually been done via maximum likelihood or bayesian approaches, a new set of techniques based on algebraic tools has emerged in the last 15 years (see [AR07] or [CFS16] for example).

More precisely, the objectives set for this project are the following:

1. Understanding and studying models of evolution from a mathematical perspective as well as the notion of phylogenetic tree.
2. Studying the $K81$ model of nucleotide substitution with algebraic tools.
3. Using multilineal algebra techniques (tensors, flattenings,...) on the joint distribution of nucleotides at the leaves of the tree.
4. Understanding and implement the quartet inference measures proposed in [CCFS].
5. Testing these inference measures on simulated and real data.

The structure of this report is the following. In Chapter 1 we introduce the biological background needed for the project: phylogenetic trees and alignments of DNA sequences. In Chapter 2 we introduce mathematical models of nucleotide substitution, in particular the $K81$ model that will be used throughout the work. In Chapter 3 we develop the algebraic tools needed for the quartet inference measures proposed in Chapter 4, where they are analyzed and tested on simulated and real data. We give conclusions and future works in Chapter 5. The Appendices contain the codes used to test and analyze the inference measures proposed.

Chapter 1

Biological background

In this Chapter we briefly introduce the biological background and context needed for this work. A good reference for the reader interested in this topic is [AR07]

1.1 Phylogenetic trees

Evolution in a biological population is understood as the divergence of a subset of it, due to environmental stresses or other reasons such as genetic drift, to eventually change so much they can no longer be considered to the same species. Phylogenetic trees are used to model and visualize the ancestral relationships among the living species in the Earth.

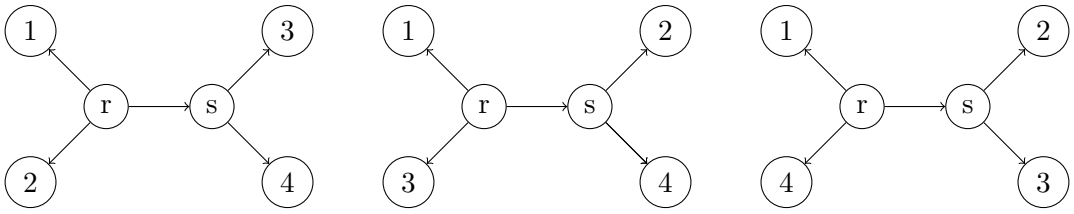
Definition 1.1. Let S be a non-empty finite set. A *phylogenetic tree* $T = (V, E)$ on S is a tree –with its set of vertices V and edges E– whose leaves are bijectively labeled in the set S.

In a phylogenetic tree, the set S represents the living species while the interior vertices are the common ancestors between species. The edges of T represent evolutionary processes and their lengths (if assigned).

The *topology of a phylogenetic tree* is the topology corresponding with the labeled graph. Two phylogenetic trees on the set S have the same topology if they represent the same evolutionary relationships among species in S (without taking branch lengths into account).

Phylogenetic trees can have a distinguished node called *root*, which is the node corresponding to the common ancestor of all the species at the leaves of the phylogenetic tree. The choice of an interior node of the tree that acts as its root implicitly directs the edges of the tree and gives a direction to the evolutionary process, in this case we call a phylogenetic tree *directed*. However, we will work with *unrooted* trees because the placement of a common ancestor to all species cannot be made solely from the data available to us. In a *directed* tree, the node from which an edge originates is called the *parent* and the node where the edge ends is called the *children*.

In this work we try to reconstruct the topology of unroooted 4-leaved phylogenetic trees, which show the evolutionary relationship between 4 species. The 3 possible topologies for such trees are notated by writing the species at the leaves opposed by the internal edge, in the fashion depicted in Figure 1.1. We will use the notation $ij|kl \equiv \{\{i, j\}, \{k, l\}\}$ as a bipartition of the $\{1, 2, 3, 4\}$ set, therefore the three distinct quartet trees are 12|34, 13|24 and 14|23.



Tree topology 12|34 Tree topology 13|24 Tree topology 14|23

Figure 1.1: The three topologies for 4-leaved phylogenetic trees.

1.2 Alignments of DNA sequences

During the XXth century the reconstruction of phylogenetic trees was done with tables of morphological traits, the more traits in common the closer two species were. Nowadays we have a more powerful tool to understand and measure how have species diverged from their common ancestor: the DNA. The DNA is a molecule present in each cell of an organism that is inherited from its progenitors. It is a nucleic acid shaped as a double helix of two strands of nucleotides. Four different nucleotides form the units of the DNA molecule: two purines, adenine (`A`) and guanine (`G`) and two pirimidines, citosine (`C`) and timine (`T`).

Definition 1.2. We refer to an *alignment of n species of length N* as a set of n ordered sequences of length N with elements in $\Sigma = \{\texttt{A}, \texttt{C}, \texttt{G}, \texttt{T}\}$, representing nucleotide sequences of length N.

Deronectes wewalkai	TTATATTTTAATCTTACCAGGATTTGGGATAATTTCCCATATTATTAGTC
Deronectes aubei	ATATATTTTAATTCTTCCAGGATTTGGTATAATTTCTCATATTATTAGTC
Deronectes theryi	TTATATTTTAATTTTACCTGGATTTGGAATAATTTCCCATATCATTAGTC
Deronectes abnormicollis	TTATATTCTAATTCTACCAGGATTTGGAATAATTTCCCATATTATTAGTC

Figure 1.2: Example of alignment for 4 species of the genus *Deronectes*.

The change of a nucleotide in a DNA chain is called a *mutation*. There are three kinds of mutations:

- *Supression*. A nucleotide is deleted.
- *Substitution*. Change of nucleotides.
- *Insertion*. An extra nucleotide is added.

In this work we will only consider substitutions, since the other two kinds of mutations can be omitted in certain circumstances.

Chapter 2

Mathematical models of nucleotide substitution

In this Chapter we introduce evolutonary models from an algebraic point of view as is done in [AR07]. We also delve into some notions on phylogenetic trees, such as branch length ([BH87]) and leaf permutations ([STHJ16]).

2.1 Algebraic models of evolution

Given an alignment of 4 species, we want to find the topology of the phylogenetic tree that best describes their evolutionary path. To do so we need a probabilistic model for the change in nucleotides from node to node, and, therefore, a set of assumptions on how this process is taking place.

Definition 2.1. In our model of substitutions in the DNA the following assumptions are made:

(i) Phylogenetic trees are *binary*. That is, internal nodes have degree 3 except the root, which may have degree 2.

(ii) The evolution of a species (in terms of mutation in the nucleotides) depends only on its father node.

(iii) Mutations are random with positive probability.

(iv) Nucleotides in a DNA sequence evolve identically and independently.

The assumptions in 2.1 and the nature of the evolutionary process, that makes only states in the present observable, give a phylogenetic tree a structure of hidden Markov model.

Definition 2.2. A *Markov model* is a stochastic model for randomly changing systems that assumes the *Markov property*, that future states depend only on the current state and not events that occurred before. When this model has unobserved states it is called a *hidden Markov model*.

In the case of a rooted phylogenetic tree, we associate discrete random variables at each node of the tree taking values in the set $\Sigma = \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$ and the Markov property states that two of these random variables are independent conditioned on the random variables of their most immediate common ancestal node.

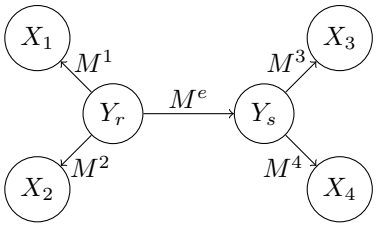


Figure 2.1: Markov process on the tree 12|34.

For a 4-leaved phylogenetic tree, labelling the leaves $i \in \{1, ..., 4\}$ we can associate a discrete random variable X_i, called *observed variables*. Similarly for the internal nodes, labelled with $j \in \{r, s\}$, Y_j will be the *hidden variables*, see Figure 2.1. The conditional probabilities of substitution of nucleotides from one node to its inmediate descendant determine the Markov process on the tree.

Definition 2.3. A Markov matrix $M \in \mathcal{M}_{n\times n}(\mathbb{R})$ is a square matrix with non-negative real entries whose rows sum 1. A *transition matrix* $M^e \in \mathcal{M}_{4\times 4}(\mathbb{R})$ for a phylogenetic tree T at a directed edge e is a Markov matrix whose entries $(M^e)_{(x,y)}$ are the conditional probability $P(y|x; e)$ for the y nucleotide to be substituded by nucleotide x along the evolutionary process along branch e.

$$M^e = \begin{array}{c} \\ \mathtt{A} \\ \mathtt{C} \\ \mathtt{G} \\ \mathtt{T} \end{array} \begin{array}{c} \begin{array}{cccc} \mathtt{A} & \mathtt{C} & \mathtt{G} & \mathtt{T} \end{array} \\ \begin{pmatrix} P(\mathtt{A}|\mathtt{A},e) & P(\mathtt{C}|\mathtt{A},e) & P(\mathtt{G}|\mathtt{A},e) & P(\mathtt{T}|\mathtt{A},e) \\ P(\mathtt{A}|\mathtt{C},e) & P(\mathtt{C}|\mathtt{C},e) & P(\mathtt{G}|\mathtt{C},e) & P(\mathtt{T}|\mathtt{C},e) \\ P(\mathtt{A}|\mathtt{G},e) & P(\mathtt{C}|\mathtt{G},e) & P(\mathtt{G}|\mathtt{G},e) & P(\mathtt{T}|\mathtt{G},e) \\ P(\mathtt{A}|\mathtt{T},e) & P(\mathtt{C}|\mathtt{T},e) & P(\mathtt{G}|\mathtt{T},e) & P(\mathtt{T}|\mathtt{T},e) \end{pmatrix} \end{array}$$

Note that the rows of the matrix sum to 1.

Remark 2.4. A 4×4 matrix can be indexed with $\Sigma = \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$ by taking $1 \to \mathtt{A}$, $2 \to \mathtt{C}$, $3 \to \mathtt{G}$, $4 \to \mathtt{T}$ and then following the natural indexation.

Given a phylogenetic tree T on 4 leaves $\{1, 2, 3, 4\}$, the probabilities of the nucleotides $\{x_1, x_2, x_3, x_4\}$ being observed at the leaves, $p^T_{x_1x_2x_3x_4} = P(X_1 = x_1, X_2 = x_2, X_3 = x_3, X_4 = x_4|T)$, can be derived by marginalizing over the interior nodes of the tree given a known distribution of nucleotides at the root,

$$\pi^r = \big(\pi_{\mathtt{A}} = P(Y_r = \mathtt{A}), \pi_{\mathtt{C}} = P(Y_r = \mathtt{C}), \pi_{\mathtt{G}} = P(Y_r = \mathtt{G}), \pi_{\mathtt{T}} = P(Y_r = \mathtt{T})\big)$$

and the conditional probabilities at the edges. We denote by M^i the transition matrix leading to leaf i and by M^e the transition matrix at the internal edge.

To find an explicit expression for those probabilities, we have to take into account the chance of a nucleotide changing along every edge from the root (or a distinguished node), with a set nucleotide distribution, to the leaves, that is given by the products of the transition matrices along those edges.

Definition 2.5. For a general n-leaved phylogenetic tree T with a set of edges $E(T)$ and a model of evolution there is a set expression for the distribution of probabilities at the leaves. If $p_{x_1...x_n} = P(x_1...x_n|T)$ probabilities of a certain configuration of nucleotides at the leaves, we have

$$p_{x_1...x_n} = \sum_{x_N \in \Sigma,\ N \in \mathrm{Int}(e)} \pi^r \prod_{e\in E(T)} M^e_{x_{pa(e)}x_{ch(e)}}$$

where $\mathrm{Int}(T)$ are the interior edges of the tree and $pa(e)$ and $ch(e)$ the parent and child nodes for an edge respectively.

Example 2.6. For the $T = 12|34$ tree the expression for $p^T_{x_1x_2x_3x_4}$ before becomes:

$$p^T_{x_1,x_2,x_3,x_4} = \sum_{x_r,x_s\in\{\mathtt{A},\mathtt{C},\mathtt{G},\mathtt{T}\}} \pi_{x_r} M^1_{(x_r,x_1)} M^2_{(x_r,x_2)} M^e_{(x_r,x_s)} M^3_{(x_r,x_3)} M^4_{(x_r,x_4)} \tag{2.1}$$

as can be derived from the Markov process on the tree.

Remark 2.7. An alignment of length N for a set of species $S = \{1, ..., n\}$ can be thought of as N independent samples of observations of the joint random variables at the leaves $(x_1, ..., x_n)$ and the relative frequencies of column patterns $x_1...x_n$ are estimators $f_{x_1...x_n}$ of the probabilities $p^T_{x_1...x_n}$ of observing $x_1...x_n$ at the leaves of the tree T which lead to the alignment.

We want to formalize now the structure generated by the assumptions that we have made in the context of the problem we are treating, in which we have a set of parameters (the transition matrices and distribution at the root) that determine a joint distribution for the variables in the nodes, and allow us to derive the joint probability at the leaves $p^T_{x_1x_2x_3x_4}$.

Definition 2.8. Let T be a n-leaved phylogenetic tree with $n+1$ transition matrices M^i along its edges $E(T)$, we define the φ_T map as:

$$\begin{aligned} \varphi_T \colon \mathbb{R}^4 \times \prod_{e\in E(T)} \mathcal{M}_{4\times4}(\mathbb{R}) &\longrightarrow \mathbb{R}^{4^n} \\ \theta = \{\pi^r, M^1, ..., M^n, M^{n+1}\} &\longmapsto \varphi_T(\theta) = p = (p_{\mathtt{AA...AA}}, p_{\mathtt{AA...AC}}, ..., p_{\mathtt{TT...TT}}). \end{aligned}$$

where the coordinates $p_{x_1...x_n}$ are given as in Definition 2.5.

Note that to be able to interpret the map φ_T as the description of the joint probability at the leaves in terms of the parameters we need to restrict the parameters to probabilities (for example π^r should be in the 3-dimensional simplex). As in this report we will be using algebraic tools we define the map in this more general setting.

Example 2.9. For T a 4-leaved phylogenetic tree we will have that φ_T is:

$$\begin{aligned} \varphi_T \colon \mathbb{R}^4 \times \prod_{e\in E(T)} \mathcal{M}_{4\times4}(\mathbb{R}) &\longrightarrow \mathbb{R}^{4^4} \\ \theta = \{\pi^r, M^1, M^2, M^3, M^4, M^e\} &\longmapsto \varphi_T(\theta) = p = (p_{\mathtt{AAAA}}, p_{\mathtt{AAAC}}, ..., p_{\mathtt{TTTT}}). \end{aligned}$$

If $T = 12|34$ then the expression of $p_{x_1x_2x_3x_4}$ is as in 2.1.

The way these transition matrices are defined and what symmetries are assumed in the process of nucleotide substitution yield a family of models of evolution.

Definition 2.10. The *general Markov model* makes no assumptions on the root distribution and the transition matrices are considered generic Markov matrices:

$$M = \begin{pmatrix} a & b & c & d \\ e & f & g & h \\ j & k & l & m \\ n & o & p & q \end{pmatrix}.$$

Definition 2.11. The *Kimura 3-parameter model* also known as *K81* includes a parameter for *transitions* (mutations from purine to purine, or pyrimidine to pyrimidine) and a parameter for each kind of *transversions* (from purine to pyrimidine or reverse), see Figure 2.2. The distribution at the root is assumed uniform:

$$\pi^r = (\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4})$$

and the transition matrix is of type:

$$M = \begin{pmatrix} a & b & c & d \\ b & a & d & c \\ c & d & a & b \\ d & c & b & a \end{pmatrix}.$$

Note that the *stationary distribution* for these matrices (v such that $vM = v$ and therefore $\lim_{n\to\infty} vM^n = v$) is the uniform distribution. Since we assume π^r to be uniform, we expect this proportion between bases to hold under iterations of this model.

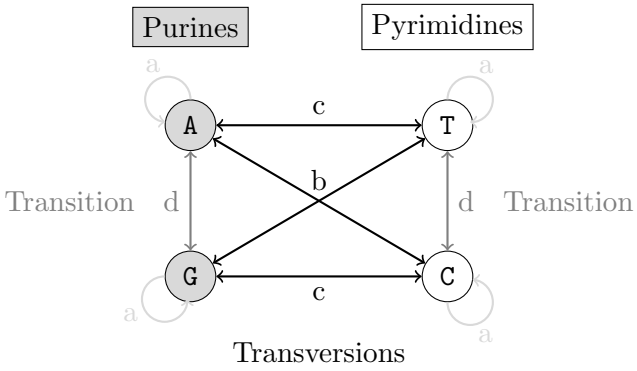


Figure 2.2: Visualization of the parameters for nucleotide substitution under the K81 model.

Remark 2.12. When the transition matrices in $\theta = \{\pi^r, M^1, M^2, M^3, M^4, M^e\}$ are $K81$ and π^r is uniform, we will refer to θ as a $K81$ *set of parameters.*

Often, one assigns lengths to the edges of the tree to represent the evolutionary distance between both nodes of the edge. This is usually measured as the expected number of substitution of nucleotides between both nodes, per nucleotide. The Markov process on a phylogenetic tree allows an estimation of this measure, see [BH87].

Lemma 2.13. *Let T be a phylogenetic tree with M transition matrix associated to a directed edge e. Then, if our nucleotide substitution model assumes uniform distribution of nucleotides at the initial node of e, the length of such branch can be approximated by $-\frac{1}{4}\log(\det(M))$*

Proof. Let M be a transition matrix, it can be rewritten as: a concatenation of infinitesimal Markov processes such that

$$M = \prod_{\alpha} M_\alpha$$

where $\{M_\alpha\}_\alpha$ are transition matrices that represent the substitution process on an infinitesimal amount of time. We have that:

$$\log(\det(M)) = \sum_{\alpha} \log(\det(M_\alpha))$$

But since M_α are infinitesimal transition matrices, very little amounts of mutation are expected, therefore these matrices are close to Id_4. If $M_\alpha = (M_\alpha^{ij})$ then they can be written as:

$$M_\alpha^{ij} = \begin{cases} 1 - m_{ij} & \text{if } i = j \\ m_{ij} & \text{if } i \neq j \end{cases}$$

where $0 < m_{ij} << 1$ are small (we omit the subscript α for a moment). We can approximate the determinant of the infinitesimal transition matrices by Taylor approximation of order 1 on the small coefficients:

$$\det(M_\alpha) = (1 - m_{11})(1 - m_{22})(1 - m_{33})(1 - m_{44}) + o(m_{ij}^2) \approx$$

$$\approx 1 - m_{11} - m_{22} - m_{33} - m_{44} = 1 - (m_{11} + m_{22} + m_{33} + m_{44})$$

Taking the logarithm and expanding once more, using $\log(1 - h) = -x + o(h^2)$:

$$-\log(\det(M_\alpha)) \approx -(m_{11} + m_{22} + m_{33} + m_{44}) = (1 - M_\alpha^{11}) + (1 - M_\alpha^{22}) + (1 - M_\alpha^{33}) + (1 - M_\alpha^{44})$$

For each process α, if i was the value of the nucleotide at the start of α, then M_α^{ii} is the chance of no change happening and $(1 - M_{ii}^\alpha) = P(i \text{ changes } | i) = \frac{P(\text{change from } i)}{P(i)}$ by conditional probabilities. By assumption we have $P(i) = \frac{1}{4}$. Therefore:

$$-\log(\det(M_\alpha)) \approx \sum_{i \in \{\mathtt{A},\mathtt{C},\mathtt{G},\mathtt{T}\}} \frac{P(\text{change from } i)}{P(i)} \text{ and } -\frac{1}{4}\log(\det(M_\alpha)) \approx \sum_{i \in \{\mathtt{A},\mathtt{C},\mathtt{G},\mathtt{T}\}} P(\text{change from } i)$$

In general, this will be an estimation of the amount of changes between two species in an interval α, as long as the interval is small enough to support the hypothesis that $m_{ij} << 1$. Finally, the amount of changes along the process lead by M is the sum of the amount of changes in the process α, and can therefore be approximated by

$$\sum_\alpha -\frac{1}{4}\log\det M_\alpha = -\frac{1}{4}\log\det M.$$

□

2.2 Leaf permutations on 4-leaved phylogenetic trees

There are three different topologies for 4-leaved phylogenetic trees, shown in Figure 2.3, which correspond to the three distinct bipartitions of the set $\{1, 2, 3, 4\}$ and are denoted by:

$$T_1 = 12|34; \quad T_2 = 13|24; \quad T_3 = 14|23.$$

Each quartet has symmetries under leaf permutations which leave it invariant. For example, when considering the tree for 4 species $\{1, 2, 3, 4\}$, we have several biologically identical trees from the equalities $12|34 = 21|34 = 34|12 = 12|43 = \ldots$, that respect the 2 sides of the bipartition.

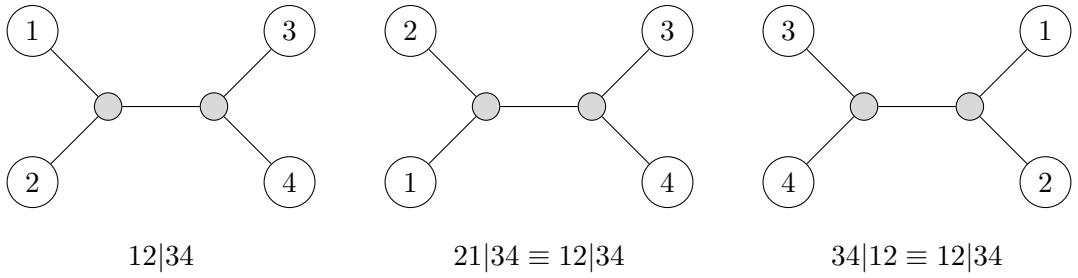


Figure 2.3: Examples of permutations that leave invariant the 12|34 topology.

Since we are shuffling a bipartition of a set with 4 elements, it makes sense to ask the question of what permutations of $\mathfrak{S}_4$ leave invariant each topology.

Definition 2.14. Let T be a 4-leaved phylogenetic tree. We call the *stabilizer of* T, $\mathrm{Stab}(T) \subset \mathfrak{S}_4$, the subset of permutations of $\mathfrak{S}_4$ that leave invariant the topology of T.

Lemma 2.15. *The stabilizers for the three topologies on 4-leaved phylogenetic trees are:*

(i) $Stab(T_1) = \{e, (12), (34), (12)(34), (13)(24), (14)(23), (1324), (1423)\}$

(ii) $Stab(T_2) = \{e, (13), (24), (12)(34), (13)(24), (14)(23), (1234), (1432)\}$

(iii) $Stab(T_3) = \{e, (14), (23), (12)(34), (13)(24), (14)(23), (1243), (1342)\}$

It is important to understand the effect of leaf permutations in a black-box algorithm for phylogenetic reconstruction. Our aim is to find an algorithm such that given an alignment for 4 species returns the correct tree topology describing their evolutionary relationships. The input for such algorithm would be 4 sequences associated to 4 species (A, B, C, D), but we have to take into account that the input is ordered and that permuting the species should give consistent results for the topology of the tree, that is, the algorithm should be *covariant under taxon permutations*. For example if for the input (A, B, C, D) the algorithm returns the 12|34 topology, we would expect for the (A, C, B, D) input to get the 13|24 topology, because they both keep A and B at the same side of the partition.

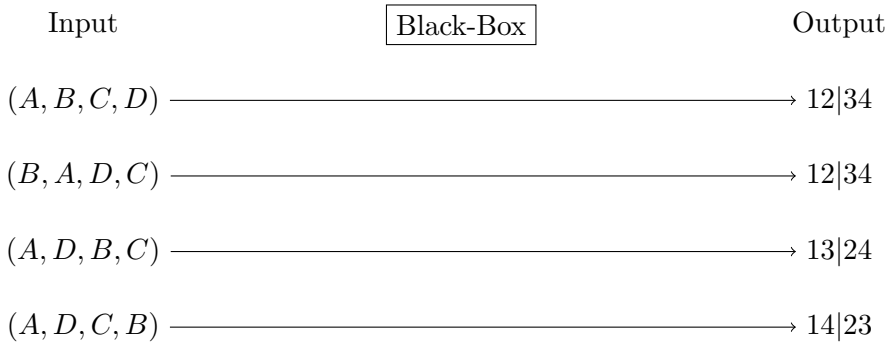


Figure 2.4: Example of covariance under taxon permutations

Chapter 3

Algebraic transformations

Given a frequency vector $f = (f_{ijkl})$ obtained from an alignment, we aim to find a transformation which provides direct ways to decide whether f is similar to a theoretical distribution in a certain phylogenetic tree. We want to transform the vector f in a way it is more manageable, to do so we will focus on:

(i) **Markov action.** An action that will allow us to modify the transition matrices in the exterior branches of the phylogenetic tree.

(ii) **Flattenings.** Converting those vectors into matrix form, which gives us new comparative tools such as ranks or determinants.

(iii) **Fourier Coordinates.** Diagonalization of the transition matrices implicit in p to have a better grasp of its structure and how to later prune unnecessary information.

The intuitive reasoning behind these transformations will become apparent when the basic properties a quartet inference measure should have are discussed in section 4.1. The results in sections 3.3 and 3.4 are based in [CCFS].

3.1 The Markov action for phylogenetic trees

Once the basic formal structure of a phylogenetic tree and the parameters that define it are set, we look for tools that allow us to alter and work with those objects. The Markov action will allow us to modify the transition matrix at exterior edges of a phylogenetic tree, which will be very useful for the proofs needed to develop methods of algebraic reconstruction.

Given a tree T, the joint probability of the nucleotides at the leaves $p^T := \left(p^T_{x_1x_2x_3x_4}\right)_{x_1x_2x_3x_4}$ can be thought of as a tensor in $\mathbb{R}^4 \otimes \mathbb{R}^4 \otimes \mathbb{R}^4 \otimes \mathbb{R}^4 = \otimes^4\mathbb{R}^4$:

$$p^T = \sum_{x_1x_2x_3x_4 \in \Sigma} p^T_{x_1x_2x_3x_4} x_1 \otimes x_2 \otimes x_3 \otimes x_4 \tag{3.1}$$

where we are considering $\Sigma = \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$ the natural basis for $\mathbb{R}^4$ and the natural basis for $\otimes^4\mathbb{R}^4$ is considered to be

$$\mathtt{A} \otimes \mathtt{A} \otimes \mathtt{A} \otimes \mathtt{A}, \mathtt{A} \otimes \mathtt{A} \otimes \mathtt{A} \otimes \mathtt{C}, ..., \mathtt{T} \otimes \mathtt{T} \otimes \mathtt{T} \otimes \mathtt{T}$$

We need the following three definitions to enable the tools that will allow us to work with this tensorial interpretation of the joint probabilities.

Definition 3.1. We denote by K the group of invertible $K81$ matrices without the stochastic condition (entries are not required to be non-negative, but sum up to 1). Note that if $M \in K$, so does M^{-1}.

Definition 3.2. The *Kroenecker product* of two matrices, where $\mathbf{A} = (a_{ij})_{ij}$ is $n \times m$ and $\mathbf{B} = (b_{kl})_{kl}$ is $p \times q$, is the $np \times mq$ matrix:

$$\mathbf{A} \otimes \mathbf{B} = \begin{pmatrix} a_{11}\mathbf{B} & \dots & a_{1n}\mathbf{B} \\ \vdots & \ddots & \vdots \\ a_{n1}\mathbf{B} & \cdots & a_{nn}\mathbf{B} \end{pmatrix}$$

We introduce the following definition from [STHJ16] adapted to our case.

Definition 3.3. The *Markov action* of the group $K \times K \times K \times K$ on the set $\otimes^4\mathbb{R}^4$ is defined as:

$$\begin{aligned} \mathcal{A}\colon \times^4 K \times \otimes^4\mathbb{R}^4 &\longrightarrow \otimes^4\mathbb{R}^4 \\ (N_1, N_2, N_3, N_4; p) &\longmapsto q = (N_1 \otimes N_2 \otimes N_3 \otimes N_4)p \end{aligned}$$

Remark 3.4. We can extend this definition for tensors in $\otimes^n\mathbb{R}^n$:

$$\begin{aligned} \mathcal{A}\colon \times^n K \times \otimes^n\mathbb{R}^n &\longrightarrow \otimes^n\mathbb{R}^n \\ (N_1, ..., N_n; p) &\longmapsto q = (N_1 \otimes ... \otimes N_n)p \end{aligned}$$

Lemma 3.5. *Let T be a phylogenetic tree, let $p = \varphi_T(\theta)$ with $\theta = \{\pi^r, M^1, M^2, M^3, M^4, M^e\}$. Consider the Markov action:*

$$\begin{aligned} \mathcal{A}\colon \times^4 K \times \otimes^4\mathbb{R}^4 &\longrightarrow \otimes^4\mathbb{R}^4 \\ (N, Id_4, Id_4, Id_4; p) &\longmapsto q = (N \otimes Id_4 \otimes Id_4 \otimes Id_4)p \end{aligned}$$

Then if we consider $p^* = \varphi_T(\theta^*)$ with $\theta^* = \{\pi^r, M^1N^t, M^2, M^3, M^4, M^e\}$ we get that

$$\mathcal{A}_{N,Id_4,Id_4,Id_4}(p) = p^*.$$

Proof. First, we compute $q = (N^t \otimes Id_4 \otimes Id_4 \otimes Id_4)p$. Since the Kroenecker product is associative: $N \otimes Id_4 \otimes Id_4 \otimes Id_4 = (N \otimes Id_4) \otimes (Id_4 \otimes Id_4)$. It easy to see that $Id_4 \otimes Id_4 = Id_{16}$. Consider $N \in K$ indexed by $\Sigma = \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$:

$$N = \begin{pmatrix} n_{\mathtt{AA}} & n_{\mathtt{AC}} & n_{\mathtt{AG}} & n_{\mathtt{AT}} \\ n_{\mathtt{CA}} & n_{\mathtt{CC}} & n_{\mathtt{CG}} & n_{\mathtt{CT}} \\ n_{\mathtt{GA}} & n_{\mathtt{GC}} & n_{\mathtt{GG}} & n_{\mathtt{GT}} \\ n_{\mathtt{TA}} & n_{\mathtt{TC}} & n_{\mathtt{TG}} & n_{\mathtt{TT}} \end{pmatrix}$$

The Kroenecker product yields:

$$N \otimes Id_4 = \begin{pmatrix} n_{\mathtt{AA}}Id_4 & \dots & n_{\mathtt{AT}}Id_4 \\ \dots & \dots & \dots \\ n_{\mathtt{TA}}Id_4 & \dots & n_{\mathtt{TT}}Id_4 \end{pmatrix} = \begin{pmatrix} n_{\mathtt{AA}} & 0 & 0 & 0 & & n_{\mathtt{AT}} & 0 & 0 & 0 \\ 0 & n_{\mathtt{AA}} & 0 & 0 & & 0 & n_{\mathtt{AT}} & 0 & 0 \\ 0 & 0 & n_{\mathtt{AA}} & 0 & \dots & 0 & 0 & n_{\mathtt{AT}} & 0 \\ 0 & 0 & 0 & n_{\mathtt{AA}} & & 0 & 0 & 0 & n_{\mathtt{AT}} \\ & & \dots & & \dots & & & \dots & \\ n_{\mathtt{TA}} & 0 & 0 & 0 & & n_{\mathtt{TT}} & 0 & 0 & 0 \\ 0 & n_{\mathtt{TA}} & 0 & 0 & & 0 & n_{\mathtt{TT}} & 0 & 0 \\ 0 & 0 & n_{\mathtt{TA}} & 0 & \dots & 0 & 0 & n_{\mathtt{TT}} & 0 \\ 0 & 0 & 0 & n_{\mathtt{TA}} & & 0 & 0 & 0 & n_{\mathtt{TT}} \end{pmatrix}$$

Now computing $(N \otimes Id_4) \otimes (Id_4 \otimes Id_4)$:

$$(N \otimes Id_4) \otimes (Id_4 \otimes Id_4) = \begin{pmatrix} n_{\mathtt{AA}} Id_{16} & \mathbf{0}_{16} & \mathbf{0}_{16} & \mathbf{0}_{16} & & n_{\mathtt{AT}} Id_{16} & \mathbf{0}_{16} & \mathbf{0}_{16} & \mathbf{0}_{16} \\ \mathbf{0}_{16} & n_{\mathtt{AA}} Id_{16} & \mathbf{0}_{16} & \mathbf{0}_{16} & \dots & \mathbf{0}_{16} & n_{\mathtt{AT}} Id_{16} & \mathbf{0}_{16} & \mathbf{0}_{16} \\ \mathbf{0}_{16} & \mathbf{0}_{16} & n_{\mathtt{AA}} Id_{16} & \mathbf{0}_{16} & & \mathbf{0}_{16} & \mathbf{0}_{16} & n_{\mathtt{AT}} Id_{16} & \mathbf{0}_{16} \\ \mathbf{0}_{16} & \mathbf{0}_{16} & \mathbf{0}_{16} & n_{\mathtt{AA}} Id_{16} & & \mathbf{0}_{16} & \mathbf{0}_{16} & \mathbf{0}_{16} & n_{\mathtt{AT}} Id_{16} \\ & \dots & & & \dots & & & \dots & \\ n_{\mathtt{TA}} Id_{16} & \mathbf{0}_{16} & \mathbf{0}_{16} & \mathbf{0}_{16} & & n_{\mathtt{TT}} Id_{16} & \mathbf{0}_{16} & \mathbf{0}_{16} & \mathbf{0}_{16} \\ \mathbf{0}_{16} & n_{\mathtt{TA}} Id_{16} & \mathbf{0}_{16} & \mathbf{0}_{16} & & \mathbf{0}_{16} & n_{\mathtt{TT}} Id_{16} & \mathbf{0}_{16} & \mathbf{0}_{16} \\ \mathbf{0}_{16} & \mathbf{0}_{16} & n_{\mathtt{TA}} Id_{16} & \mathbf{0}_{16} & \dots & \mathbf{0}_{16} & \mathbf{0}_{16} & n_{\mathtt{TT}} Id_{16} & \mathbf{0}_{16} \\ \mathbf{0}_{16} & \mathbf{0}_{16} & \mathbf{0}_{16} & n_{\mathtt{TA}} Id_{16} & & \mathbf{0}_{16} & \mathbf{0}_{16} & \mathbf{0}_{16} & n_{\mathtt{TT}} Id_{16} \end{pmatrix}$$

where $\mathbf{0}_{16}$ is the 16×16 matrix with all entries equal to 0. Now the product of this matrix by $p = (p_{\mathtt{AAAA}}, p_{\mathtt{AAAC}}, p_{\mathtt{AAAG}}, ..., p_{\mathtt{TTTT}})$:

$$q = (N \otimes Id_4) \otimes (Id_4 \otimes Id_4) \begin{pmatrix} p_{\mathtt{AAAA}} \\ p_{\mathtt{AAAC}} \\ p_{\mathtt{AAAG}} \\ \dots \\ p_{\mathtt{TTTT}} \end{pmatrix} = \begin{pmatrix} n_{\mathtt{AA}} p_{\mathtt{AAAA}} + n_{\mathtt{AC}} p_{\mathtt{CAAA}} + n_{\mathtt{AG}} p_{\mathtt{GAAA}} + n_{\mathtt{AT}} p_{\mathtt{TAAA}} \\ n_{\mathtt{AA}} p_{\mathtt{AAAC}} + n_{\mathtt{AC}} p_{\mathtt{CAAC}} + n_{\mathtt{AG}} p_{\mathtt{GAAC}} + n_{\mathtt{AT}} p_{\mathtt{TAAT}} \\ \dots \\ n_{\mathtt{GA}} p_{\mathtt{ATAG}} + n_{\mathtt{GC}} p_{\mathtt{CTAG}} + n_{\mathtt{GG}} p_{\mathtt{GTAG}} + n_{\mathtt{GT}} p_{\mathtt{TTAG}} \\ \dots \\ n_{\mathtt{TA}} p_{\mathtt{ATTT}} + n_{\mathtt{TC}} p_{\mathtt{CTTT}} + n_{\mathtt{TG}} p_{\mathtt{GTTT}} + n_{\mathtt{TT}} p_{\mathtt{TTTT}} \end{pmatrix} = \begin{pmatrix} q_{\mathtt{AAAA}} \\ q_{\mathtt{AAAC}} \\ \dots \\ q_{\mathtt{CTAG}} \\ \dots \\ q_{\mathtt{TTTT}} \end{pmatrix}$$

which can be rewritten in a more compact form as:

$$q_{x_1 x_2 x_3 x_4} = \sum_{x_i \in \{\mathtt{A},\mathtt{C},\mathtt{G},\mathtt{T}\}} n_{x_1 x_i} p_{x_i x_2 x_3 x_4}$$

It is left to see that $p^* = (p^*_{\mathtt{AAAA}}, p^*_{\mathtt{AAAC}}, p^*_{\mathtt{AAAG}}, ..., p^*_{\mathtt{TTTT}})$, $p^* = \varphi_T(\theta^*)$ with $\theta^* = \{\pi^r, M^1 N^t, M^2, M^3, M^4, M^e\}$, satisfies that $p^* = q$. Assume T has topology $t = 12|34$, but the proof extends by analogy to the other two 4-leaved tree topologies. As in 2.1, the coordinates of p^* can be computed as:

$$p^*_{x_1 x_2 x_3 x_4} = \sum_{x_r, x_s \in \{\mathtt{A},\mathtt{C},\mathtt{G},\mathtt{T}\}} \pi_{x_r} (M^1 \cdot N^t)_{(x_r, x_1)} M^2_{(x_r, x_2)} M^e_{(x_r, x_s)} M^3_{(x_s, x_3)} M^4_{(x_s, x_4)} \tag{3.2}$$

Given

$$M^1 = \begin{pmatrix} m_{\mathtt{AA}} & m_{\mathtt{AC}} & m_{\mathtt{AG}} & m_{\mathtt{AT}} \\ m_{\mathtt{CA}} & m_{\mathtt{CC}} & m_{\mathtt{CG}} & m_{\mathtt{CT}} \\ m_{\mathtt{GA}} & m_{\mathtt{GC}} & m_{\mathtt{GG}} & m_{\mathtt{GT}} \\ m_{\mathtt{TA}} & m_{\mathtt{TC}} & m_{\mathtt{TG}} & m_{\mathtt{TT}} \end{pmatrix}$$

we develop the term

$$(M^1 \cdot N^t)_{(x_r, x_1)} = \begin{pmatrix} m_{x_r \mathtt{A}} & m_{x_r \mathtt{C}} & m_{x_r \mathtt{G}} & m_{x_r \mathtt{T}}; \end{pmatrix} \cdot \begin{pmatrix} n_{x_1 \mathtt{A}} \\ n_{x_1 \mathtt{C}} \\ n_{x_1 \mathtt{G}} \\ n_{x_1 \mathtt{T}} \end{pmatrix} = n_{x_1 \mathtt{A}} m_{x_r \mathtt{A}} + n_{x_1 \mathtt{C}} m_{x_r \mathtt{C}} + n_{x_1 \mathtt{G}} m_{x_r \mathtt{G}} + n_{x_1 \mathtt{T}} m_{x_r \mathtt{T}};$$

Substituting this in 3.2 we get:

$$p^*_{x_1 x_2 x_3 x_4} = \sum_{x_i \in \{\mathtt{A},\mathtt{C},\mathtt{G},\mathtt{T}\}} n_{x_1 x_i} \sum_{x_r, x_s \in \{\mathtt{A},\mathtt{C},\mathtt{G},\mathtt{T}\}} \pi_{x_r} m_{x_r x_i} M^2_{(x_r, x_2)} M^e_{(x_r, x_s)} M^3_{(x_s, x_3)} M^4_{(x_s, x_4)}$$

and therefore:

$$p^*_{x_1x_2x_3x_4} = \sum_{x_i \in \{\mathtt{A},\mathtt{C},\mathtt{G},\mathtt{T}\}} n_{x_1x_i} p_{x_ix_2x_3x_4} = q_{x_1x_2x_3x_4},$$

which is precisely what we aimed to find. □

Proposition 3.6. *Let T be a phylogenetic tree, let $p = \varphi_T(\theta)$ with $\theta = \{\pi^r, M^1, M^2, M^3, M^4, M^e\}$. Consider the Markov action:*

$$\begin{aligned} \mathcal{A}\colon \times^4 K \times \otimes^4\mathbb{R}^4 &\longrightarrow \otimes^4\mathbb{R}^4 \\ (N_1, N_2, N_3, N_4; p) &\longmapsto q = (N_1 \otimes N_2 \otimes N_3 \otimes N_4)p \end{aligned}$$

Then if we consider $p^* = \varphi_T(\theta^*)$ with $\theta^* = \{\pi^r, M^1N_1^t, M^2N_2^t, M^3N_3^t, M^4N_4^t, M^e\}$ we get that

$$\mathcal{A}_{N_1,N_2,N_3,N_4}(p) = p^*.$$

Proof. By applying Lemma 3.5 if we consider:

$$p^1 = (N_1 \otimes Id_4 \otimes Id_4 \otimes Id_4)p \text{ then we have } p^1 = \varphi(\pi^r, M^1N_1^t, M^2, M^3, M^4, M^e)$$

and analogously:

$$p^2 = (Id_4 \otimes N_2 \otimes Id_4 \otimes Id_4)p^1 \text{ equals } p^2 = \varphi(\pi^r, M^1N_1^t, M^2N_2^t, M^3, M^4, M^e),$$

$$p^3 = (Id_4 \otimes Id_4 \otimes N_3 \otimes Id_4)p^2 \text{ equals } p^3 = \varphi(\pi^r, M^1N_1^t, M^2N_2^t, M^3N_3^t, M^4, M^e) \text{ and}$$

$$p^4 = (Id_4 \otimes Id_4 \otimes Id_4 \otimes N_4)p^3 \text{ equals } p^4 = \varphi(\pi^r, M^1N_1^t, M^2N_2^t, M^3N_3^t, M^4N_4^t, M^e).$$

We have $p^4 = \varphi_T(\pi^r, M^1N_1^t, M^2N_2^t, M^3N_3^t, M^4N_4^t, M^e) = p^*$ by definition and $p^4 = q$ because of how the group action is defined, the composition es equivalent to multiplying the matrices in the correct position in $\times^4 K$. Therefore $p^* = q$ as we wanted to see. □

Remark 3.7. In order to retain the probabilistic interpretations of the parameter of our model we have restricted this action to Markov matrices. Nevertheless, in the proof we do not use this fact and therefore this result is valid for general matrices, but the resulting parameters will not be interpretable biologically, as we will see in section 3.3.

This Markov action, in affecting the transition matrices of the tree has the following effect in the length of the edges of T, in relation to Lemma 2.13.

Lemma 3.8. *The Markov action of the matrix N onto a branch i changes the approximation of the length of such branch by adding $-\frac{1}{4}\log(|\det(N)|)$ to it.*

Proof. From Lemma 2.13:

$$length(i^*) = -\frac{1}{4}\log(|\det(M^{i*})|) = -\frac{1}{4}\log(|\det(M^iN^t)|) =$$

$$= -\frac{1}{4}\log(|\det(M^i)det(N^t)|) = -\frac{1}{4}\log(|\det(M^i)|) - \frac{1}{4}\log(|\det(N)|) = length(i) - \frac{1}{4}\log(|\det(N)|)$$

. □

3.2 Flattenings

A flattening is a way to flatten the $\mathbb{R}^4 \otimes \mathbb{R}^4 \otimes \mathbb{R}^4 \otimes \mathbb{R}^4$ tensors we are working with into matrices by indexing the elements according to a bipartition of $\Sigma = \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$ as basis for $\mathbb{R}^4$. Given a $f = (f_{ijkl})$ we want to provide 3 different matrices, one corresponding to each possible tree topology, which can be compared to the theoretical matrices obtained with $p = (p_{ijkl})$ theoretical distribution.

Definition 3.9. Let $p = (p_{\mathtt{AAAA}}, p_{\mathtt{AAAC}}, p_{\mathtt{AAAG}}, ..., p_{\mathtt{TTTT}})$ be a tensor in $\mathbb{R}^4 \otimes \mathbb{R}^4 \otimes \mathbb{R}^4 \otimes \mathbb{R}^4$. Then, the flattenings for the three different bipartitions are obtained by indexing the elements of p by the states of the leaves in the opposed sites of the bipartition:

$$flatt_{12|34}(p) = \begin{array}{c} \\ \text{States} \\ \text{at leaves} \\ \text{1 and 2} \end{array} \begin{array}{c} \text{States at leaves 3 and 4} \\ \begin{pmatrix} p_{\mathtt{AAAA}} & p_{\mathtt{AAAC}} & \cdots & p_{\mathtt{AATT}} \\ p_{\mathtt{ACAA}} & p_{\mathtt{ACAC}} & \cdots & p_{\mathtt{ACTT}} \\ p_{\mathtt{AGAA}} & p_{\mathtt{AGAC}} & \cdots & p_{\mathtt{AGTT}} \\ \cdots & \cdots & \cdots & \cdots \\ p_{\mathtt{TTAA}} & p_{\mathtt{TTAC}} & \cdots & p_{\mathtt{TTTT}} \end{pmatrix} \end{array}$$

$$flatt_{13|24}(p) = \begin{array}{c} \\ \text{States} \\ \text{at leaves} \\ \text{1 and 3} \end{array} \begin{array}{c} \text{States at leaves 2 and 4} \\ \begin{pmatrix} p_{\mathtt{AAAA}} & p_{\mathtt{AAAC}} & \cdots & p_{\mathtt{ATAT}} \\ p_{\mathtt{AACA}} & p_{\mathtt{AACC}} & \cdots & p_{\mathtt{ATCT}} \\ p_{\mathtt{AAGA}} & p_{\mathtt{AAGC}} & \cdots & p_{\mathtt{ATGT}} \\ \cdots & \cdots & \cdots & \cdots \\ p_{\mathtt{TATA}} & p_{\mathtt{TATC}} & \cdots & p_{\mathtt{TTTT}} \end{pmatrix} \end{array}$$

$$flatt_{14|23}(p) = \begin{array}{c} \\ \text{States} \\ \text{at leaves} \\ \text{1 and 4} \end{array} \begin{array}{c} \text{States at leaves 2 and 3} \\ \begin{pmatrix} p_{\mathtt{AAAA}} & p_{\mathtt{AACA}} & \cdots & p_{\mathtt{ATTA}} \\ p_{\mathtt{AAAC}} & p_{\mathtt{AACC}} & \cdots & p_{\mathtt{ATTC}} \\ p_{\mathtt{AAAG}} & p_{\mathtt{AACG}} & \cdots & p_{\mathtt{ATTG}} \\ \cdots & \cdots & \cdots & \cdots \\ p_{\mathtt{TAAT}} & p_{\mathtt{TACT}} & \cdots & p_{\mathtt{TTTT}} \end{pmatrix} \end{array}$$

Theorem 3.10. ***(Allman-Rhodes)*** *Under the general Markov model 2.10, let p be the joint distribution at the leaves of the tree 12|34 under some transition matrices, $p = \varphi_{12|34}(\theta)$. Then the following hold:*

(i) $rank(flatt_{12|34}(p)) \leq 4$

(ii) $rank(flatt_{13|24}(p)) = 16$ *for general parameters*

(iii) $rank(flatt_{14|23}(p)) = 16$ *for general parameters*

Proof. See [AR07]. □

This theorem already gives us a ground for comparison, because upon computing the flattening for the vector of frequencies $f = (f_{ijkl})$ obtained from an alignment we would imagine it would have arisen from the 12|34 topology if its 12|34 flattening had rank less or equal than 4 approximately.

3.3 Fourier coordinates

As seen in Chapter 2, the probability vector $p = (p_{ijkl})$ can be written in terms of the transition matrices along the 5 edges of the tree, for example under the 12|34 topology:

$$p_{x_1x_2x_3x_4} = \sum_{x_r,x_s \in \{\mathtt{A},\mathtt{C},\mathtt{G},\mathtt{T}\}} \pi_r M^1_{(x_r,x_1)} M^2_{(x_r,x_2)} M^e_{(x_r,x_s)} M^3_{(x_s,x_3)} M^4_{(x_s,x_4)}$$

In order to see more clearly what is the structure behind the transformations we are making to such vector and asses its dependency on the different transition matrices we need to simplify the terms in the expression, and an intuitive way to do it is choosing a basis in which the transition matrices are diagonal.

Lemma 3.11. *Let M be a transition matrix under the $K81$ model for a tree T and $\pi^r = (\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4})$ its root distribution. The Hadamard matrix*

$$H = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix}$$

has the following properties:

(i) $H^{-1} = \frac{1}{4}H$

(ii) $H^t = H$

(iii) $H_{(i,k)}H_{(j,k)} = H_{(i+j,k)}$

(iv) *H diagonalizes M: $H^{-1}MH = diag(m_{\mathtt{A}}, m_{\mathtt{C}}, m_{\mathtt{G}}, m_{\mathtt{T}})$ where $m_{\mathtt{A}} = a+b+c+d, m_{\mathtt{C}} = a+b-c-d, m_{\mathtt{G}} = a-b+c-d, m_{\mathtt{T}} = a-b-c-d$ are the eigenvalues and a basis of corresponding eigenvectors is $\bar{\Sigma} = \{\bar{\mathtt{A}}, \bar{\mathtt{C}}, \bar{\mathtt{G}}, \bar{\mathtt{T}}\}$ where*

$$\bar{\mathtt{A}} = (1,1,1,1)^t \qquad \bar{\mathtt{G}} = (1,-1,1,-1)^t$$
$$\bar{\mathtt{C}} = (1,1,-1,-1)^t \qquad \bar{\mathtt{T}} = (1,-1,-1,1)^t$$

(v) $m_{\mathtt{A}} = 1$

(vi) *In this new basis of eigenvectors $\bar{\Sigma}$ the root distribution becomes $\bar{\pi}^r = (\frac{1}{4}, 0, 0, 0)$*

Proof. (i) It follows by observing that $H^2 = \frac{1}{4}Id_4$

(ii) It holds because H is symmetric.

(iii) Easily verifiable.

(iv) We compute:

$$H^{-1}MH = H^{-1}\begin{pmatrix} a & b & c & d \\ b & a & d & c \\ c & d & a & b \\ d & c & b & a \end{pmatrix} H = \begin{pmatrix} a+b+c+d & 0 & 0 & 0 \\ 0 & a+b-c-d & 0 & 0 \\ 0 & 0 & a-b+c-d & 0 \\ 0 & 0 & 0 & a-b-c+d \end{pmatrix}$$

Since this is a diagonal matrix, a basis of eigenvectors is given by the columns of H.

(v) By definition of Markov matrix, $a+b+c+d = m_{\mathtt{A}} = 1$.

(vi) We compute:

$$\bar{\pi}^r = H^{-1}\pi^r = \frac{1}{4}\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix}\begin{pmatrix} \frac{1}{4} \\ \frac{1}{4} \\ \frac{1}{4} \\ \frac{1}{4} \end{pmatrix} = \begin{pmatrix} \frac{1}{4} \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

□

We now define the Fourier coordinates for tensors and give a basic property for this new basis of eigenvectors.

Definition 3.12. Let p be a tensor in $\mathbb{R}^4 \otimes \mathbb{R}^4 \otimes \mathbb{R}^4 \otimes \mathbb{R}^4$ with coordinates

$$p_\Sigma = (p_{\mathtt{AAAA}}, p_{\mathtt{AAAC}}, ..., p_{\mathtt{TTTT}})$$

in the $\mathtt{A}\otimes\mathtt{A}\otimes\mathtt{A}\otimes\mathtt{A}, ..., \mathtt{T}\otimes\mathtt{T}\otimes\mathtt{T}\otimes\mathtt{T}$ basis. We denote by

$$p_{\bar{\Sigma}} = (\bar{p}_{\mathtt{AAAA}}, \bar{p}_{\mathtt{AAAC}}, ..., \bar{p}_{\mathtt{TTTT}})$$

its coordinates in the $\bar{\mathtt{A}}\otimes\bar{\mathtt{A}}\otimes\bar{\mathtt{A}}\otimes\bar{\mathtt{A}}, ..., \bar{\mathtt{T}}\otimes\bar{\mathtt{T}}\otimes\bar{\mathtt{T}}\otimes\bar{\mathtt{T}}$ basis. This is,

$$p = \sum_{x_1x_2x_3x_4\in\Sigma} p_{x_1x_2x_3x_4} x_1\otimes x_2\otimes x_3\otimes x_4 = \sum_{x_1x_2x_3x_4\in\Sigma} \bar{p}_{x_1x_2x_3x_4} \bar{x_1}\otimes\bar{x_2}\otimes\bar{x_3}\otimes\bar{x_4}$$

and we call $p_{\bar{\Sigma}}$ the *Fourier coordinates*.

Lemma 3.13. *The following rule applies to change coordinates with tensors in $\otimes^4\mathbb{R}^4$, where H is the change of basis matrix between $\bar{\Sigma}$ and Σ:*

$$p_{\bar{\Sigma}} = (H^{-1}\otimes H^{-1}\otimes H^{-1}\otimes H^{-1})p_\Sigma = \frac{1}{4^4}(H\otimes H\otimes H\otimes H)p_\Sigma$$

Proof. H^{-1} is the change of basis matrix from Σ to $\bar{\Sigma}$ and hence $H^{-1}\otimes H^{-1}\otimes H^{-1}\otimes H^{-1}$ is the change of basis from $\{x_1\otimes x_2\otimes x_3\otimes x_4\}_{x_i\in\Sigma}$ to $\{\bar{x_1}\otimes\bar{x_2}\otimes\bar{x_3}\otimes\bar{x_4}\}_{\bar{x_i}\in\bar{\Sigma}}$ and the last equality follows from Lemma 3.11 (i). □

We introduce now an interpretation of Σ as a group by the following bijection with $\mathbb{Z}_2\times\mathbb{Z}_2$:

$$\begin{array}{ccc} \Sigma = \{\mathtt{A},\mathtt{C},\mathtt{G},\mathtt{T}\} & \longleftrightarrow & (\mathbb{Z}_2\times\mathbb{Z}_2, +) \\ \mathtt{A} & \longleftrightarrow & (0,0) \\ \mathtt{C} & \longleftrightarrow & (1,0) \\ \mathtt{G} & \longleftrightarrow & (0,1) \\ \mathtt{T} & \longleftrightarrow & (1,1) \end{array}$$

which gives an additive group structure to Σ, with neutral element $\mathtt{A}$ and the following addition table:

+	A	C	G	T
A	A	C	G	T
C	C	A	T	G
G	G	T	A	C
T	T	G	C	A

This will be a very useful tool to index matrices and vectors. The same change of basis $\Sigma = \{\mathtt{A},\mathtt{C},\mathtt{G},\mathtt{T}\}$ to $\bar{\Sigma} = \{\bar{\mathtt{A}},\bar{\mathtt{C}},\bar{\mathtt{G}},\bar{\mathtt{T}}\}$, can be formalized by considering Σ as the $\mathbb{Z}_2 \times \mathbb{Z}_2$ group and the entries of the transition matrices as characters on it. Then the discrete Fourier transform plays the role of the change of basis described above, see [CGS05]. This is the reason for the adjective Fourier for these coordinates. Now we can examine how the expression of p in Fourier coordinates simplifies greatly.

Proposition 3.14. *If $p = \varphi_{12|34}(\theta)$ with $\theta = \{\pi^r, M^1, M^2, M^3, M^4, M^e\}$ a K81 set of parameters, then p can be rewritten in Fourier coordinates in terms of the eigenvalues of the transition matrices as:*

$$\bar{p}_{x_1x_2x_3x_4} = \begin{cases} \frac{1}{4^4} \cdot m^1_{x_1} m^2_{x_2} m^e_{x_1+x_2} m^3_{x_3} m^4_{x_4}, & \text{if } x_1 + x_2 = x_3 + x_4; \\ 0, & \text{otherwise.} \end{cases}$$

where m^i_x refers to the $x \in \Sigma$ eigenvalue (in the notation of Lemma 3.11 (iv)) of the transition matrix M^i and Σ has the additive group structure defined above.

Proof. Firstly, we take the equality in Lemma 3.13 and combining it with Proposition 3.6 we have that considering $p_{\bar{\Sigma}}$ Fourier coordinates for p is the same as taking $\bar{p} = \varphi_T(\tilde{\theta})$ arising from T with

$$\tilde{\theta}_M = \{\pi^r, M^1(H^{-1})^t, M^2(H^{-1})^t, M^3(H^{-1})^t, M^4(H^{-1})^t, M^e\} = \{\pi^r, \tilde{M}^1, \tilde{M}^2, \tilde{M}^3, \tilde{M}^4, M^e\}.$$

Knowing that $(H^{-1})^t = \frac{1}{4}H^t = \frac{1}{4}H$ we have the following result for each $\tilde{M}^i$:

$$\tilde{M}^i = M^i \cdot \frac{1}{4}H = \begin{pmatrix} a^i & b^i & c^i & d^i \\ b^i & a^i & d^i & c^i \\ c^i & d^i & a^i & b^i \\ d^i & c^i & b^i & a^i \end{pmatrix} \cdot \frac{1}{4}\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix} = \frac{1}{4}\begin{pmatrix} m^i_{\mathtt{A}} & m^i_{\mathtt{C}} & m^i_{\mathtt{G}} & m^i_{\mathtt{T}} \\ m^i_{\mathtt{A}} & m^i_{\mathtt{C}} & -m^i_{\mathtt{G}} & -m^i_{\mathtt{T}} \\ m^i_{\mathtt{A}} & -m^i_{\mathtt{C}} & m^i_{\mathtt{G}} & -m^i_{\mathtt{T}} \\ m^i_{\mathtt{A}} & -m^i_{\mathtt{C}} & -m^i_{\mathtt{G}} & m^i_{\mathtt{T}} \end{pmatrix}$$

Using the expression (2.1) in terms of the transition matrices we have:

$$\bar{p}_{x_1x_2x_3x_4} = \sum_{x_r,x_s \in \{\mathtt{A},\mathtt{C},\mathtt{G},\mathtt{T}\}} \pi_{x_r} \tilde{M}^1_{(x_r,x_1)} \tilde{M}^2_{(x_r,x_2)} M^e_{(x_r,x_s)} \tilde{M}^3_{(x_r,x_3)} \tilde{M}^4_{(x_r,x_4)}$$

By observing that

$$\tilde{M}^i_{(x_j,x_i)} = \frac{1}{4} H_{(x_j,x_i)} m^i_{x_i}$$

and with $\pi_{x_r} = (\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4})^t$ the expression can be rewritten as:

$$\bar{p}_{x_1x_2x_3x_4} = \frac{1}{4^5} m^1_{x_1} m^2_{x_2} m^3_{x_3} m^4_{x_4} \sum_{x_r,x_s \in \{\mathtt{A},\mathtt{C},\mathtt{G},\mathtt{T}\}} H_{(x_r,x_1)} H_{(x_r,x_2)} H_{(x_s,x_3)} H_{(x_s,x_4)} M^e_{(x_r,x_s)}$$

By property (iii) in Lemma 3.11 we have:

$$H_{(x_r,x_1)} H_{(x_r,x_2)} H_{(x_s,x_3)} H_{(x_s,x_4)} = H_{(x_r,x_1+x_2)} H_{(x_s,x_3+x_4)}$$

Going back to the sum:

$$\sum_{x_r,x_s \in \{\mathtt{A},\mathtt{C},\mathtt{G},\mathtt{T}\}} H_{(x_r,x_1+x_2)} H_{(x_s,x_3+x_4)} M^e_{(x_r,x_s)} = \sum_{x_r \in \{\mathtt{A},\mathtt{C},\mathtt{G},\mathtt{T}\}} H_{(x_r,x_1+x_2)} \sum_{x_s \in \{\mathtt{A},\mathtt{C},\mathtt{G},\mathtt{T}\}} H_{(x_r,x_3+x_4)} M^e_{(x_r,x_s)}$$

and using that H is symmetric we have:

$$\sum_{x_r \in \{\mathtt{A},\mathtt{C},\mathtt{G},\mathtt{T}\}} H_{(x_1+x_2,x_r)}(M^e H)_{(x_r,x_3+x_4)} = (HM^e H)_{(x_1+x_2,x_3+x_4)}$$

which by taking $H = \frac{1}{4}H^{-1}$ and noting it is precisely the change to the diagonal form of M^e yields:

$$(HM^e H)_{(x_1+x_2,x_3+x_4)} = 4(H^{-1}M^e H)_{(x_1+x_2,x_3+x_4)} = \begin{cases} 4m^e_{x_1+x_2}, & \text{if } x_1+x_2 = x_3+x_4; \\ 0, & \text{otherwise.} \end{cases}$$

Therefore, by plugging this result into the previous expression we get:

$$\bar{p}_{x_1x_2x_3x_4} = \begin{cases} \frac{1}{4^4} \cdot m^1_{x_1} m^2_{x_2} m^e_{x_1+x_2} m^3_{x_3} m^4_{x_4}, & \text{if } x_1+x_2 = x_3+x_4; \\ 0, & \text{otherwise,} \end{cases}$$

which is precisely what we aimed to prove. □

3.4 Marginalization Adjustment

If we combine the two transformations presented so far, we can consider the flattening in Fourier coordinates of p, which will be the backbone of the reconstruction methods presented.

Definition 3.15. The *Fourier flattening of p relative to* $12|34$, with $p = \varphi_T(\theta)$, is:

$$\overline{flatt}_{12|34}(p) := flatt_{12|34}(p_{\bar{\Sigma}})$$

which by Lemma 3.13 equals to

$$\overline{flatt}_{12|34}(p) = flatt_{12|34}((H^{-1} \otimes H^{-1} \otimes H^{-1} \otimes H^{-1})p).$$

However, in the case of the Fourier flattening matrix is considered to be indexed in the rows/columns in the following order:

$$\{\mathtt{AA}, \mathtt{CC}, \mathtt{GG}, \mathtt{TT}, \mathtt{AC}, \mathtt{CA}, \mathtt{GT}, \mathtt{TG}, \mathtt{AG}, \mathtt{CT}, \mathtt{GA}, \mathtt{TC}, \mathtt{AT}, \mathtt{CG}, \mathtt{GC}, \mathtt{TA}\}.$$

Note that with the structure of Σ as additive group we have:

$$\begin{gathered} \mathtt{A}+\mathtt{A} = \mathtt{C}+\mathtt{C} = \mathtt{G}+\mathtt{G} = \mathtt{T}+\mathtt{T} = \mathtt{A} \\ \mathtt{A}+\mathtt{C} = \mathtt{C}+\mathtt{A} = \mathtt{G}+\mathtt{T} = \mathtt{T}+\mathtt{G} = \mathtt{C} \\ \mathtt{A}+\mathtt{G} = \mathtt{C}+\mathtt{T} = \mathtt{G}+\mathtt{A} = \mathtt{T}+\mathtt{C} = \mathtt{G} \\ \mathtt{A}+\mathtt{T} = \mathtt{C}+\mathtt{G} = \mathtt{G}+\mathtt{C} = \mathtt{T}+\mathtt{A} = \mathtt{T}. \end{gathered}$$

For example, with this indexation we will have $\overline{flatt}_{12|34}(p)_{(\mathtt{AC},\mathtt{TG})} = \bar{p}_{\mathtt{ACTG}}$.

The $\overline{flatt}_{13|24}(p)$ and $\overline{flatt}_{14|23}(p)$ matrices are defined analogously by permuting the leaves $2 \leftrightarrow 3$ and $1 \leftrightarrow 4$ respectively. Then we have, for example:

$$\overline{flatt}_{13|24}(p)_{(\mathtt{AC},\mathtt{TG})} = \bar{p}_{\mathtt{ATCG}}; \quad \overline{flatt}_{14|23}(p)_{(\mathtt{AC},\mathtt{TG})} = \bar{p}_{\mathtt{ATGC}}$$

Remark 3.16. Due to Proposition 3.14, the 12|34 flattening in Fourier coordinates of a probability vector $p = \varphi_T(\theta)$ with T 4-leaved phylogenetic tree with θ a $K81$ set of parameters is a block diagonal matrix:

$$\overline{flatt}_{12|34}(p) = \begin{pmatrix} B^{12}_{\mathtt{A}} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & B^{12}_{\mathtt{C}} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & B^{12}_{\mathtt{G}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & B^{12}_{\mathtt{T}} \end{pmatrix}$$

where $\mathbf{0}$ is a 4×4 matrix with all entries equal to 0 and

$$B^{12}_{\mathtt{A}} = \begin{pmatrix} \bar{p}_{\mathtt{AAAA}} & \bar{p}_{\mathtt{AACC}} & \bar{p}_{\mathtt{AAGG}} & \bar{p}_{\mathtt{AATT}} \\ \bar{p}_{\mathtt{CCAA}} & \bar{p}_{\mathtt{CCCC}} & \bar{p}_{\mathtt{CCGG}} & \bar{p}_{\mathtt{CCTT}} \\ \bar{p}_{\mathtt{GGAA}} & \bar{p}_{\mathtt{GGCC}} & \bar{p}_{\mathtt{GGGG}} & \bar{p}_{\mathtt{GGTT}} \\ \bar{p}_{\mathtt{TTAA}} & \bar{p}_{\mathtt{TTCC}} & \bar{p}_{\mathtt{TTGG}} & \bar{p}_{\mathtt{TTTT}} \end{pmatrix} \qquad B^{12}_{\mathtt{G}} = \begin{pmatrix} \bar{p}_{\mathtt{AGAG}} & \bar{p}_{\mathtt{AGCT}} & \bar{p}_{\mathtt{AGGA}} & \bar{p}_{\mathtt{AGTC}} \\ \bar{p}_{\mathtt{CTAG}} & \bar{p}_{\mathtt{CTCT}} & \bar{p}_{\mathtt{CTGA}} & \bar{p}_{\mathtt{CTTC}} \\ \bar{p}_{\mathtt{GAAG}} & \bar{p}_{\mathtt{GACT}} & \bar{p}_{\mathtt{GAGA}} & \bar{p}_{\mathtt{GATC}} \\ \bar{p}_{\mathtt{TCAG}} & \bar{p}_{\mathtt{TCCT}} & \bar{p}_{\mathtt{TCGA}} & \bar{p}_{\mathtt{TCTC}} \end{pmatrix}$$

$$B^{12}_{\mathtt{C}} = \begin{pmatrix} \bar{p}_{\mathtt{ACAC}} & \bar{p}_{\mathtt{ACCA}} & \bar{p}_{\mathtt{ACGT}} & \bar{p}_{\mathtt{ACTG}} \\ \bar{p}_{\mathtt{CAAC}} & \bar{p}_{\mathtt{CACA}} & \bar{p}_{\mathtt{CAGT}} & \bar{p}_{\mathtt{CATG}} \\ \bar{p}_{\mathtt{GTAC}} & \bar{p}_{\mathtt{GTCA}} & \bar{p}_{\mathtt{GTGT}} & \bar{p}_{\mathtt{GTTG}} \\ \bar{p}_{\mathtt{TGAC}} & \bar{p}_{\mathtt{TGCA}} & \bar{p}_{\mathtt{TGGT}} & \bar{p}_{\mathtt{TGTG}} \end{pmatrix} \qquad B^{12}_{\mathtt{T}} = \begin{pmatrix} \bar{p}_{\mathtt{ATAT}} & \bar{p}_{\mathtt{ATCG}} & \bar{p}_{\mathtt{ATGC}} & \bar{p}_{\mathtt{ATTA}} \\ \bar{p}_{\mathtt{CGAT}} & \bar{p}_{\mathtt{CGCG}} & \bar{p}_{\mathtt{CGGC}} & \bar{p}_{\mathtt{CGTA}} \\ \bar{p}_{\mathtt{GCAT}} & \bar{p}_{\mathtt{GCCG}} & \bar{p}_{\mathtt{GCGC}} & \bar{p}_{\mathtt{GCTA}} \\ \bar{p}_{\mathtt{TAAT}} & \bar{p}_{\mathtt{TACG}} & \bar{p}_{\mathtt{TAGC}} & \bar{p}_{\mathtt{TATA}} \end{pmatrix}.$$

Note that even if $p = \varphi_T(p)$ with $T = 13|24$ or $T = 14|23$, the $\overline{flatt}_{12|34}(p)$ has still block-diagonal form. Indeed, in this case one can apply the analogous version of Proposition 3.14 and by observing that

$$x_1 + x_2 = x_3 + x_4 \Leftrightarrow x_1 + x_3 = x_2 + x_4 \Leftrightarrow x_1 + x_4 = x_2 + x_3$$

we are done.

Moreover, if p is a distribution on the 12|34 topology, by Proposition 3.14, each block in the Fourier flattening $\overline{flatt}_{12|34}(p)$ will have rank 1, while it has generally rank 4 otherwise.

The qualitative idea of how different the topologies of two metric phylogenetic trees are should rely on the length of the interior edge (implicit in M^e under the $K81$ model, as seen in Lemma 2.13). Therefore, the dependence on the length of the pendant edges should not be important when determining the tree topology that originated the alignment. In section 4.1 this idea will be reinforced in Properties 2 and 3 for quartet inference measures, where we will ask the expectation of our measures to not be 'too' dependant on how we modify the matrices (or lengths) at the pendant edges. We now develop a tool to modify the Fourier flattening matrix in a way that it only depends on the entries of the M^e matrix.

Definition 3.17. Given a 4-tensor with coordinates $(p_{x_1x_2x_3x_4})_{x_1x_2x_3x_4}$ in the Σ basis, the *marginal of the* $1-2$ *leaves* is the 2-tensor $(p_{12})_{++}$ given by:

$$(p_{12})_{x_1x_2} = \sum_{x_3,x_4 \in \Sigma} p_{x_1x_2x_3x_4}$$

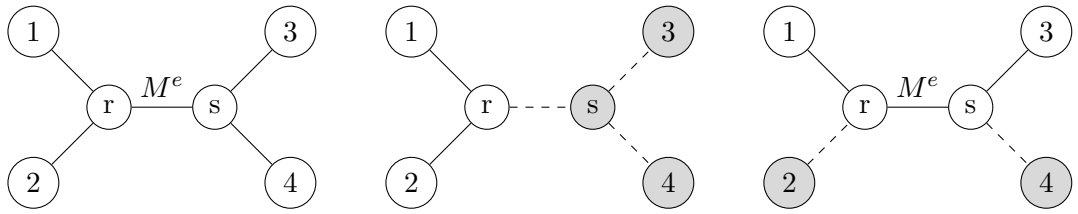
Analogous definitions are given for the $(p_{ij})_{++}$ marginals, with $i, j \in \{1, 2, 3, 4\}$.

For example, the marginals of the $1-3$ leaves is written as:

$$(p_{13})_{x_1x_3} = p_{x_1 \cdot x_3 \cdot} = \sum_{x_2,x_4 \in \Sigma} p_{x_1x_2x_3x_4}$$

This transformation describes a smaller tree with observations only at the leaves we are marginalizing. This is because we are grouping the probabilities by the leaves we take the marginal of, and

this new probabilities do not account for the observation at the leaves we are summing in respect of, because of the law of total probability. For example, in the 12|34 topology, the marginals for $1-2$ and $1-3$ would correspond to the trees shown in Figure 3.1.



Tree topology 12|34 Tree for the $1-2$ marginal Tree for the $1-3$ marginal

Figure 3.1: Marginals for the 12|34 topology

We can verify this fact by developing the expression of $(p_{12})_{x_1x_2}$:

$$(p_{12})_{x_1x_2} = p_{x_1x_2\cdot\cdot} = \sum_{x_3,x_4\in\Sigma} p_{x_1x_2x_3x_4} = \sum_{x_3,x_4\in\Sigma}\sum_{x_r,x_s\in\Sigma} \pi^r M^1_{(x_r,x_1)} M^2_{(x_r,x_2)} M^e_{(x_r,x_s)} M^3_{(x_s,x_3)} M^4_{(x_s,x_4)}$$

$$= \sum_{x_r\in\Sigma} \pi^r M^1_{(x_r,x_1)} M^2_{(x_r,x_2)} \sum_{x_s\in\Sigma} M^e_{(x_r,x_s)} \sum_{x_3\in\Sigma} M^3_{(x_s,x_3)} \sum_{x_4\in\Sigma} M^4_{(x_s,x_4)} = \frac{1}{4}\sum_{x_r\in\Sigma} M^1_{(x_r,x_1)} M^2_{(x_r,x_2)}$$

which as expected is congruent with $(p_{12})_{x_1x_2} = \varphi(\theta_{12})$ and $\theta_{12} = \{\pi^r, M^1, M^2\}$. This can be generalized for the marginals of leaves at the same side of the bipartition.

As seen in Figure 3.1, for marginals of leaves in opposed sides of the bipartition, the s node is in the middle of an edge, with degree 2 instead of 3. For the $1-3$ marginalization we proceed as before and develop:

$$(p_{13})_{x_1x_3} = p_{x_1x_3\cdot\cdot} = \sum_{x_2,x_4\in\Sigma} p_{x_1x_2x_3x_4} = \sum_{x_2,x_4\in\Sigma}\sum_{x_r,x_s\in\Sigma} \pi^r M^1_{(x_r,x_1)} M^2_{(x_r,x_2)} M^e_{(x_r,x_s)} M^3_{(x_s,x_3)} M^4_{(x_s,x_4)}$$

$$= \sum_{x_r\in\Sigma} \pi^r M^1_{(x_r,x_1)} \sum_{x_s\in\Sigma} M^3_{(x_s,x_3)} M^e_{(x_r,x_s)} \sum_{x_2\in\Sigma} M^2_{(x_r,x_2)} \sum_{x_4\in\Sigma} M^4_{(x_s,x_4)} = \frac{1}{4}\sum_{x_r\in\Sigma} M^1_{(x_r,x_1)} (M^3M^e)_{(x_r,x_3)}$$

which indeed corresponds to the tree with $\theta_{13} = \{\pi^r, M^1, M^3M^e\}$. This means we are considering the $r \to s$ edge and the $s \to 3$ to be one, so that the transition matrix for the $r \to 3$ edge will be M^3M^e. This is analogous for marginals of leaves in opposing sides of the bipartition.

In the following proposition we give an explicit expression for the marginals in Fourier coordinates by developing the intuitions seen here.

Proposition 3.18. *Let T phylogenetic tree with the 12|34 topology and $p = \varphi_T(\theta)$ under the K81 model, with $\theta = \{\pi^r, M^1, M^2, M^3, M^4, M^e\}$. In Fourier coordinates the marginalization of the $1-2$ leaves is given by:*

$$(\overline{p_{12}})_{x_1x_2} = \begin{cases} \frac{1}{4^2} m^1_{x_1} m^2_{x_2} & \text{if } x_1 = x_2 \\ 0 & \text{if } x_1 \neq x_2 \end{cases}$$

and for the $1-3$ leaves:

$$(\overline{p_{13}})_{x_1x_3} = \begin{cases} \frac{1}{4^2} m^1_{x_1} m^3_{x_3} m^e_{x_1} & \text{if } x_1 = x_3 \\ 0 & \text{if } x_1 \neq x_3 \end{cases}$$

and similar expressions are deduced for marginals of leaves at the same and at opposite sides of the bipartition respectively.

Proof. We have proposed and will prove this result for the $1-2$ and $1-3$ marginals, the other cases are proven by analogy depending on the kind of bipartition they are.

As seen above, $(p_{12})_{x_1x_2} = \varphi(\theta_{12})$ and $\theta_{12} = \{\pi^r, M^1, M^2\}$. In the same fashion as with 4-tensors, we can understand the change to Fourier coordinates as

$$(\overline{p_{12}})_{x_1x_2} = \frac{1}{4^2}(H \otimes H)(p_{12})_{x_1x_2}$$

which translates to the tree with $(\overline{p_{12}})_{x_1x_2} = \varphi(\bar{\theta}_{12})$, $\bar{\theta}_{12} = \{\pi^r, \frac{1}{4}M^1H, \frac{1}{4}M^2H\}$. Plugging this result into the previous expression we get:

$$(\overline{p_{12}})_{x_1x_2} = \frac{1}{4^3}\sum_{x_r\in\Sigma}(M^1H)_{(x_r,x_1)}(M^2H)_{(x_r,x_2)}$$

The matrices have the following structure:

$$M^i \cdot H = \begin{pmatrix} m^i_{\mathtt{A}} & m^i_{\mathtt{C}} & m^i_{\mathtt{G}} & m^i_{\mathtt{T}} \\ m^i_{\mathtt{A}} & m^i_{\mathtt{C}} & -m^i_{\mathtt{G}} & -m^i_{\mathtt{T}} \\ m^i_{\mathtt{A}} & -m^i_{\mathtt{C}} & m^i_{\mathtt{G}} & -m^i_{\mathtt{T}} \\ m^i_{\mathtt{A}} & -m^i_{\mathtt{C}} & -m^i_{\mathtt{G}} & m^i_{\mathtt{T}} \end{pmatrix}$$

and it is easy to check that if $x_1 \neq x_2$ the sum of the product between the elements of the x_1 and x_2 columns will always cancel out. Therefore the $\sum_{x_r\in\Sigma}(M^1H)_{(x_r,x_1)}(M^2H)_{(x_r,x_2)}$ sum will only be different from 0 when $x_1 = x_2$, when the result will be $4m^1_{x_1}m^2_{x_2}$. Therefore, we get the desired expression:

$$(\overline{p_{12}})_{x_1x_2} = \begin{cases} \frac{1}{4^2}m^1_{x_1}m^2_{x_2} & \text{if } x_1 = x_2 \\ 0 & \text{if } x_1 \neq x_2 \end{cases}$$

For the $1-3$ marginalization in Fourier coordinates we proceed as before, with $(p_13)_{x_1x_3} = \varphi_T(\theta_{13})$ and $\theta_{13} = \{\pi^r, M^1, M^3M^e\}$. Now the tree in Fourier coordinates will have $\theta_{13} = \{\pi^r, M^1H, M^3M^eH\}$, computing as before:

$$(p_{13})_{x_1x_3} = \frac{1}{4^3}\sum_{x_r\in\Sigma}(M^1_{(x_r,x_1)}H)(M^3M^eH)_{(x_r,x_3)}$$

which by the same argument as before cancels out when $x_1 \neq x_3$. Taking into account that:

$$M^3 \cdot M^e \cdot H = \begin{pmatrix} m^3_{\mathtt{A}}m^e_{\mathtt{A}} & m^3_{\mathtt{C}}m^e_{\mathtt{C}} & m^3_{\mathtt{G}}m^e_{\mathtt{G}} & m^3_{\mathtt{T}}m^e_{\mathtt{T}} \\ m^3_{\mathtt{A}}m^e_{\mathtt{A}} & m^3_{\mathtt{C}}m^e_{\mathtt{C}} & -m^3_{\mathtt{G}}m^e_{\mathtt{G}} & -m^3_{\mathtt{T}}m^e_{\mathtt{T}} \\ m^3_{\mathtt{A}}m^e_{\mathtt{A}} & -m^3_{\mathtt{C}}m^e_{\mathtt{C}} & m^3_{\mathtt{G}}m^e_{\mathtt{G}} & -m^3_{\mathtt{T}}m^e_{\mathtt{T}} \\ m^3_{\mathtt{A}}m^e_{\mathtt{A}} & -m^3_{\mathtt{C}}m^e_{\mathtt{C}} & -m^3_{\mathtt{G}}m^e_{\mathtt{G}} & m^3_{\mathtt{T}}m^e_{\mathtt{T}} \end{pmatrix}$$

we get the desired result:

$$(\overline{p_{13}})_{x_1x_3} = \begin{cases} \frac{1}{4^2}m^1_{x_1}m^3_{x_3}m^e_{x_1} & \text{if } x_1 = x_3 \\ 0 & \text{if } x_1 \neq x_3 \end{cases}$$

□

Remark 3.19. The Fourier transform of the marginal does not coincide with the marginal of the Fourier transform:

$$\sum_{x_3x_4} \bar{p}_{x_1x_2x_3x_4} = \ldots = \frac{1}{4^4} m^1_{x_1} m^2_{x_2} m^e_{x_1+x_2} \sum_{x_3} m^3_{x_3} m^4_{x_1+x_2-x_3}$$

which depends on transition matrices M^3, M^4 and M^e and cannot be the same in general as the Fourier transform of the marginal.

We will now consider the $\mathtt{A}$ block of the Fourier flattening block diagonal matrix and 'normalize' its entries so that they depend only on the interior branch matrix M^e. To do so, we will divide by the marginals. Although analogous results can be derived for the other blocks of the Fourier flattening block diagonal matrix, we only work with $B^{ij}_{\mathtt{A}}$ to develop further methods. Previous work on the subject seemed to indicate that the results were similar when considering all 4 blocks instead of the first.

Definition 3.20. Let p be a probability tensor in $\mathbb{R}^4 \otimes \mathbb{R}^4 \otimes \mathbb{R}^4 \otimes \mathbb{R}^4$. For each bipartition $ij|kl$ we define the matrix $G_{ij|kl}(p)$ as:

$$G_{ij|kl}(p) = 4^2(\overline{p_{ik}})_{\mathtt{AA}} \begin{pmatrix} \frac{1}{(\overline{p_{ij}})_{\mathtt{AA}}} & 0 & 0 & 0 \\ 0 & \frac{1}{(\overline{p_{ij}})_{\mathtt{CC}}} & 0 & 0 \\ 0 & 0 & \frac{1}{(\overline{p_{ij}})_{\mathtt{GG}}} & 0 \\ 0 & 0 & 0 & \frac{1}{(\overline{p_{ij}})_{\mathtt{TT}}} \end{pmatrix} B^{ij}_{\mathtt{A}} \begin{pmatrix} \frac{1}{(\overline{p_{kl}})_{\mathtt{AA}}} & 0 & 0 & 0 \\ 0 & \frac{1}{(\overline{p_{kl}})_{\mathtt{CC}}} & 0 & 0 \\ 0 & 0 & \frac{1}{(\overline{p_{kl}})_{\mathtt{GG}}} & 0 \\ 0 & 0 & 0 & \frac{1}{(\overline{p_{kl}})_{\mathtt{TT}}} \end{pmatrix}$$

Example 3.21. For the $12|34$ Fourier flattening we get the matrix:

$$G_{12|34}(p) = 4^2(\overline{p_{13}})_{\mathtt{AA}} \begin{pmatrix} \frac{\bar{p}_{\mathtt{AAAA}}}{(\overline{p_{12}})_{\mathtt{AA}}(\overline{p_{34}})_{\mathtt{AA}}} & \frac{\bar{p}_{\mathtt{AACC}}}{(\overline{p_{12}})_{\mathtt{AA}}(\overline{p_{34}})_{\mathtt{CC}}} & \frac{\bar{p}_{\mathtt{AAGG}}}{(\overline{p_{12}})_{\mathtt{AA}}(\overline{p_{34}})_{\mathtt{GG}}} & \frac{\bar{p}_{\mathtt{AATT}}}{(\overline{p_{12}})_{\mathtt{AA}}(\overline{p_{34}})_{\mathtt{TT}}} \\ \frac{\bar{p}_{\mathtt{CCAA}}}{(\overline{p_{12}})_{\mathtt{CC}}(\overline{p_{34}})_{\mathtt{AA}}} & \frac{\bar{p}_{\mathtt{CCCC}}}{(\overline{p_{12}})_{\mathtt{CC}}(\overline{p_{34}})_{\mathtt{CC}}} & \frac{\bar{p}_{\mathtt{CCGG}}}{(\overline{p_{12}})_{\mathtt{CC}}(\overline{p_{34}})_{\mathtt{GG}}} & \frac{\bar{p}_{\mathtt{CCTT}}}{(\overline{p_{12}})_{\mathtt{CC}}(\overline{p_{34}})_{\mathtt{TT}}} \\ \frac{\bar{p}_{\mathtt{GGAA}}}{(\overline{p_{12}})_{\mathtt{GG}}(\overline{p_{34}})_{\mathtt{AA}}} & \frac{\bar{p}_{\mathtt{GGCC}}}{(\overline{p_{12}})_{\mathtt{GG}}(\overline{p_{34}})_{\mathtt{CC}}} & \frac{\bar{p}_{\mathtt{GGGG}}}{(\overline{p_{12}})_{\mathtt{GG}}(\overline{p_{34}})_{\mathtt{GG}}} & \frac{\bar{p}_{\mathtt{GGTT}}}{(\overline{p_{12}})_{\mathtt{GG}}(\overline{p_{34}})_{\mathtt{TT}}} \\ \frac{\bar{p}_{\mathtt{TTAA}}}{(\overline{p_{12}})_{\mathtt{TT}}(\overline{p_{34}})_{\mathtt{AA}}} & \frac{\bar{p}_{\mathtt{TTCC}}}{(\overline{p_{12}})_{\mathtt{TT}}(\overline{p_{34}})_{\mathtt{CC}}} & \frac{\bar{p}_{\mathtt{TTGG}}}{(\overline{p_{12}})_{\mathtt{TT}}(\overline{p_{34}})_{\mathtt{GG}}} & \frac{\bar{p}_{\mathtt{TTTT}}}{(\overline{p_{12}})_{\mathtt{TT}}(\overline{p_{34}})_{\mathtt{TT}}} \end{pmatrix}$$

Remark 3.22. Although from Proposition 3.18 we have that $4^2(\overline{p_{ik}})_{\mathtt{AA}} = 1$, this term will be necessary for section 4.2. In the following proposition it will be omitted to avoid convoluting the notation.

Proposition 3.23. *If $p = \varphi_T(\theta)$ with $T = 12|34$ and $\theta = \{\pi^r, M^1, M^2, M^3, M^4, M^e\}$ a K81 set of parameters, then:*

(i) all entries of $G_{12|34}(p)$ are equal to 1

(ii) $(G_{13|24}(p))_{(\mathtt{X},\mathtt{Y})} = (G_{14|23}(p))_{(\mathtt{X},\mathtt{Y})} = \frac{m^e_{\mathtt{X}+\mathtt{Y}}}{m^e_{\mathtt{X}} m^e_{\mathtt{Y}}}$

(iii) $detG_{13|24}(p) = detG_{14|23}(p) = \frac{4^4 M^e_{(\mathtt{A},\mathtt{A})} M^e_{(\mathtt{A},\mathtt{C})} M^e_{(\mathtt{A},\mathtt{G})} M^e_{(\mathtt{A},\mathtt{T})}}{(detM^e)^2}$

(iv) All $G_{ij|kl}(p)$ matrices are invariant under permutations of leaves that preserve both sides of the bipartition.

(v) If the sides of the bipartition are permuted, transposed matrices are obtained.

Proof. (i) In this case we get that the entries of the matrix are:

$$(G_{12|34}(p))_{(\mathtt{X},\mathtt{Y})} = \frac{\bar{p}_{\mathtt{XXYY}}}{(\overline{p_{12}})_{\mathtt{XX}}(\overline{p_{34}})_{\mathtt{YY}}} = m^e_{\mathtt{X}+\mathtt{Y}} = m^e_{\mathtt{A}} = 1$$

because by Propositions 3.14 and 3.18:

$$\bar{p}_{\mathtt{XXYY}} = \frac{1}{4^4} m^1_{\mathtt{X}} m^2_{\mathtt{X}} m^e_{\mathtt{X}+\mathtt{Y}} m^3_{\mathtt{Y}} m^4_{\mathtt{Y}}, \quad (\overline{p_{12}})_{\mathtt{XX}} = \frac{1}{4^2} m^1_{\mathtt{X}} m^2_{\mathtt{X}}, \quad (\overline{p_{34}})_{\mathtt{YY}} = \frac{1}{4^2} m^3_{\mathtt{Y}} m^4_{\mathtt{Y}}.$$

(ii) The entries of these matrices are:

$$(G_{13|24}(p))_{(\mathtt{X},\mathtt{Y})} = \frac{\bar{p}_{\mathtt{XYXY}}}{(\overline{p_{13}})_{\mathtt{XX}}(\overline{p_{24}})_{\mathtt{YY}}} = \frac{m^e_{\mathtt{X}+\mathtt{Y}}}{m^e_{\mathtt{X}} m^e_{\mathtt{Y}}}; \quad (G_{14|23}(p))_{(\mathtt{X},\mathtt{Y})} = \frac{\bar{p}_{\mathtt{XYYX}}}{(\overline{p_{14}})_{\mathtt{XX}}(\overline{p_{23}})_{\mathtt{YY}}} = \frac{m^e_{\mathtt{X}+\mathtt{Y}}}{m^e_{\mathtt{X}} m^e_{\mathtt{Y}}}$$

by Propositions 3.14 and 3.18.

(iii) Since both matrices are equal by (ii), we will use $(G_{13|24}(p))_{(\mathtt{X},\mathtt{Y})}$ in the proof. In the last item we have seen that $(G_{13|24}(p))_{(\mathtt{X},\mathtt{Y})} = \frac{m^e_{\mathtt{X}+\mathtt{Y}}}{m^e_{\mathtt{X}} m^e_{\mathtt{Y}}}$:

$$G_{13|24}(p) = \begin{pmatrix} \frac{m^e_{\mathtt{A}}}{m^e_{\mathtt{A}} m^e_{\mathtt{A}}} & \frac{m^e_{\mathtt{C}}}{m^e_{\mathtt{A}} m^e_{\mathtt{C}}} & \frac{m^e_{\mathtt{G}}}{m^e_{\mathtt{A}} m^e_{\mathtt{G}}} & \frac{m^e_{\mathtt{T}}}{m^e_{\mathtt{A}} m^e_{\mathtt{T}}} \\ \frac{m^e_{\mathtt{C}}}{m^e_{\mathtt{C}} m^e_{\mathtt{A}}} & \frac{m^e_{\mathtt{A}}}{m^e_{\mathtt{C}} m^e_{\mathtt{C}}} & \frac{m^e_{\mathtt{T}}}{m^e_{\mathtt{C}} m^e_{\mathtt{G}}} & \frac{m^e_{\mathtt{G}}}{m^e_{\mathtt{C}} m^e_{\mathtt{T}}} \\ \frac{m^e_{\mathtt{G}}}{m^e_{\mathtt{G}} m^e_{\mathtt{A}}} & \frac{m^e_{\mathtt{T}}}{m^e_{\mathtt{G}} m^e_{\mathtt{C}}} & \frac{m^e_{\mathtt{A}}}{m^e_{\mathtt{G}} m^e_{\mathtt{G}}} & \frac{m^e_{\mathtt{C}}}{m^e_{\mathtt{G}} m^e_{\mathtt{T}}} \\ \frac{m^e_{\mathtt{T}}}{m^e_{\mathtt{T}} m^e_{\mathtt{A}}} & \frac{m^e_{\mathtt{G}}}{m^e_{\mathtt{T}} m^e_{\mathtt{C}}} & \frac{m^e_{\mathtt{C}}}{m^e_{\mathtt{T}} m^e_{\mathtt{G}}} & \frac{m^e_{\mathtt{A}}}{m^e_{\mathtt{T}} m^e_{\mathtt{T}}} \end{pmatrix}$$

Each column and each row has a $m^e_{x_i}$ dividing to all its elements. By properties of the determinant we get:

$$\det G_{13|24}(p) = \frac{1}{(m^e_{\mathtt{A}} m^e_{\mathtt{C}} m^e_{\mathtt{G}} m^e_{\mathtt{T}})^2} \begin{vmatrix} m^e_{\mathtt{A}} & m^e_{\mathtt{C}} & m^e_{\mathtt{G}} & m^e_{\mathtt{T}} \\ m^e_{\mathtt{C}} & m^e_{\mathtt{A}} & m^e_{\mathtt{T}} & m^e_{\mathtt{G}} \\ m^e_{\mathtt{G}} & m^e_{\mathtt{T}} & m^e_{\mathtt{A}} & m^e_{\mathtt{C}} \\ m^e_{\mathtt{T}} & m^e_{\mathtt{G}} & m^e_{\mathtt{C}} & m^e_{\mathtt{A}} \end{vmatrix}$$

This is a determinant of a matrix with the same symmetries as a $K81$ matrix, therefore its eigenvalues are, as seen in Lemma 3.11:

$$\begin{aligned} m^e_{\mathtt{A}} + m^e_{\mathtt{C}} + m^e_{\mathtt{G}} + m^e_{\mathtt{T}} &= 4M^e_{\mathtt{A},\mathtt{A}}, \\ m^e_{\mathtt{A}} + m^e_{\mathtt{C}} - m^e_{\mathtt{G}} - m^e_{\mathtt{T}} &= 4M^e_{\mathtt{A},\mathtt{C}}, \\ m^e_{\mathtt{A}} - m^e_{\mathtt{C}} + m^e_{\mathtt{G}} - m^e_{\mathtt{T}} &= 4M^e_{\mathtt{A},\mathtt{G}}, \\ m^e_{\mathtt{A}} - m^e_{\mathtt{C}} - m^e_{\mathtt{G}} + m^e_{\mathtt{T}} &= 4M^e_{\mathtt{A},\mathtt{T}}. \end{aligned}$$

and we get the following determinant:

$$detG_{13|24}(p) = detG_{14|23}(p) = \frac{4^4 M^e_{(\mathtt{A},\mathtt{A})} M^e_{(\mathtt{A},\mathtt{C})} M^e_{(\mathtt{A},\mathtt{G})} M^e_{(\mathtt{A},\mathtt{T})}}{(detM^e)^2}$$

(iv) Follows from the structure of the matrix, since the leaves at the same side of the partition play the same role.

(v) We will show that

$$G_{34|12}(p) = G_{12|34}(p)^t$$

and the other cases will follow analogously. The element $(\mathtt{X}, \mathtt{Y})$ in the $G_{34|12}(p)$ matrix is:

$$G_{34|12}(p)_{(\mathtt{X},\mathtt{Y})} = \frac{\bar{p}_{\mathtt{YYXX}}}{(\overline{p_{12}})_{\mathtt{YY}}(\overline{p_{34}})_{\mathtt{XX}}}$$

which is the same as $G_{12|34}(p)_{(\mathtt{Y},\mathtt{X})}$.

□

Chapter 4

Algebraic techniques for phylogenetic reconstruction

In this Chapter we propose and test several quartet inference measures based in [CCFS] and the properties for inference measures in [STHJ16]. We test the inference measures on the Huelsenbeck ([Hue95]) tree space and perform some preliminary analysis of their bias.

4.1 Quartet inference measures

In this section we introduce some 'natural' properties an inference method for quartets should have, based on [STHJ16].

Let us consider an alignment A of DNA sequences with length N for 4 species. Let p be a *probability distribution* in $\mathbb{R}^4 \otimes \mathbb{R}^4 \otimes \mathbb{R}^4 \otimes \mathbb{R}^4$ and $P = Np$ the corresponding *absolute probability distribution* on the set of 4-tuple nucleotides configurations. Let f and F be the array of *relative and absolute frequencies* (respectively) observed in the alignment A. By assuming that the columns of A are independent samples from the distribution P, we have that the vector F follows a multinomial distribution,

$$F \sim \text{MultiNom}(p, N).$$

We understand the observed absolute frequency F as the count of the successes in 256 categories (combinations of $\{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$) for the N trials that are configurations at the columns of A. With this statistical interpretation of the problem, we can expect to define a *quartet inference measure* and to develop properties it should have, since now we can work with expectancies and confidence.

Definition 4.1. A *quartet inference measure* for a pattern frequency array $F = (F_{ijkl})$ is a triple

$$\Delta(F) = (R_{12}, R_{13}, R_{14})$$

where R_{ij} is a (statistically interpretable) confidence in $F \sim \text{MultiNom}(p, N)$, with p arising as a distribution on the tree $T = ij|\{1, 2, 3, 4\} \setminus \{i, j\}$.

The following properties are desirable for the purposes of phylogenetic reconstruction. They aim to guarantee consistency of the algorithm: good behaviour under covariancy of taxon permutations and certain control on the transition matrices of the external branches.

Property 1. *A quartet inference method $\Delta(F)$ should be covariant under taxon permutations.*

Property 2. ***(Strong)*** *In expectation, the quartet inference measure $\Delta(F)$ should be covariant under the Markov action (3.3). That is, if $F \sim MultiNom(p, N)$ where $p = \varphi_T(\theta)$ is the distribution on a quartet tree T with $\theta = \{\pi^r, M^1, M^2, M^3, M^4, M^e\}$ and $F' \sim MultiNom(p', N)$ such that $p' = \varphi_T(\theta')$ with $\theta' = \{\pi^r, M^1N_1, M^2N_2, M^3N_3, M^4N_4; M^e\}$, there should exist a scalar $\lambda(N_1, N_2, N_3, N_4)$ such that:*

$$\mathbb{E}[\Delta(F')] = \lambda(N_1, N_2, N_3, N_4)\mathbb{E}[\Delta(F)]$$

Property 3. ***(Weak)*** *In the same conditions as Property 2, the equality of expectations is only necessary to hold in the limit of infinite sequence length:*

$$\lim_{N\to\infty} \mathbb{E}[\Delta(F')] = \lambda(N_1, N_2, N_3, N_4) \lim_{N\to\infty} \mathbb{E}[\Delta(F)]$$

These properties induce the definition of specific type of function that could be used as quartet inference measure.

Definition 4.2. A *Markov invariant* for the $K81$ model $q : \mathbb{R}^4 \otimes \mathbb{R}^4 \otimes \mathbb{R}^4 \otimes \mathbb{R}^4 \longrightarrow \mathbb{R}$ is a function such that if p is a tensor in $\mathbb{R}^4 \otimes \mathbb{R}^4 \otimes \mathbb{R}^4 \otimes \mathbb{R}^4$ and p' is obtained from p by the Markov action $p' = (N_1 \otimes N_2 \otimes N_3 \otimes N_4)p$ for some matrices $N_i \in K$, then

$$q(P') = \lambda(N_1, N_2, N_3, N_4)q(P)$$

with λ a group homomorphism ($\lambda(N_1N_1', N_2N_2', N_3N_3', N_4N_4') = \lambda(N_1, N_2, N_3, N_4)\lambda(N_1', N_2', N_3', N_4')$).

Intuitively, this idea of a Markov invariant follows from properties 2 and 3, as it would induce a quartet inference measure that satisfies the weak property. Even though the Markov invariant is defined for an arbitrary function λ, we have some interest in 'controlling' it, since λ is a measure of the dependence of our method to elongations of the external edges, which is something we want to minimize. Therefore Markov invariants with $\lambda \equiv 1$ seem to be a good approach to finding a robust phylogenetic reconstruction method.

4.2 Proposed quartet inference measures

In this section we propose some inference quartet measures derived from the algebraic transformations presented before. We also discuss their behaviour in relation with the properties and intuitions in section 4.1. In general we assume without loss of generality that a Δ inference quartet measure is designed so that smaller values of R_{ij} correspond to greater confidence in the tree $T = ij|kl$.

Going back to the idea of Markov invariants and as a result of the transformations we have made in order to avoid dependance on external branches, we find the following property for the $G_{ij|kl}(p)$ matrices.

Lemma 4.3. *The entries of $G_{12|34}(p)$, $G_{13|24}(p)$ and $G_{14|23}(p)$ are Markov invariants with $\lambda = 1$.*

Proof. Let p be a tensor in $\mathbb{R}^4 \otimes \mathbb{R}^4 \otimes \mathbb{R}^4 \otimes \mathbb{R}^4$ and let p' be obtained from p by the Markov action:

$$p' = (N_1 \otimes N_2 \otimes N_3 \otimes N_4)p$$

for some matrices $N_i \in K$. Then, when we transform p and p' into Fourier coordinates we have:

$$\overline{p'} = (H^{-1} \otimes H^{-1} \otimes H^{-1} \otimes H^{-1})(N_1 \otimes N_2 \otimes N_3 \otimes N_4)p$$

and since $p = (H \otimes H \otimes H \otimes H)\overline{p}$ we obtain:

$$\overline{p'} = (H^{-1}N_1H \otimes H^{-1}N_2H \otimes H^{-1}N_3H \otimes H^{-1}N_4H)$$

As N_i share the symmetries of $K81$ matrices, H diagonalizes them, $H^{-1}N_iH = \mathrm{diag}(n^i_{\mathtt{A}}, n^i_{\mathtt{C}}, n^i_{\mathtt{G}}, n^i_{\mathtt{T}})$. The Kroenecker product of diagonal matrices is a diagonal matrix and therefore we obtain

$$\overline{p'}_{x_1x_2x_3x_4} = n^1_{x_1}n^2_{x_2}n^3_{x_3}n^4_{x_4}\overline{p}_{x_1x_2x_3x_4}. \tag{4.1}$$

An analogous formula holds for the marginalizations.

Now the entries of $G_{ij|kl}$ are Markov invariant with $\lambda = 1$ if $G_{ij|kl}(p') = G_{ij|kl}(p)$ for any tensor p. Note that in the definition of $G_{ij|kl}(p')$ the terms n^i_x cancel out in each fraction (only remaining $n^i_{\mathtt{A}}n^k_{\mathtt{A}}$ which is 1, from the $(\overline{p_{ik}})_{\mathtt{AA}}$ term). Thus, $G_{ij|kl}(p') = G_{ij|kl}(p)$ and hence each entry of the matrix is a Markov invariant with $\lambda = 1$. □

Therefore we propose the following quartet inference measure based on the $G_{ij|kl}$ matrices in Definition 3.20. Since for the correct topology the result of $G_{ij|kl}(p)$ is a matrix with all entries equal to 1, we should expect the smaller determinant to be the one for the correct tree topology.

Definition 4.4. Determinant of G matrices (*detG*).

$$\Delta_{detG}(F) = \big(\det G_{12|34}(F), \det G_{13|24}(F), \det G_{14|23}(F)\big)$$

We now describe the properties for the *detG* inference measure.

Proposition 4.5. *$\Delta_{detG}(F)$ is a quartet inference measure that satisfies Properties 1 and 3 with $\lambda = 1$. Moreover the limit expectation of $\Delta_{detG}(F)$ when $F \sim MultiNom(p, N)$ and $p = \varphi_T(\theta)$ with a set θ of $K81$ parameters, is:*

(i) $\lim_{N\to\infty}\mathbb{E}[\Delta_{detG}(F)] = \Delta_{detG}(p) = \big(0, \frac{4^4 M^e_{\mathtt{A},\mathtt{A}}M^e_{\mathtt{A},\mathtt{C}}M^e_{\mathtt{A},\mathtt{G}}M^e_{\mathtt{A},\mathtt{T}}}{(\det M^e)^2}, \frac{4^4 M^e_{\mathtt{A},\mathtt{A}}M^e_{\mathtt{A},\mathtt{C}}M^e_{\mathtt{A},\mathtt{G}}M^e_{\mathtt{A},\mathtt{T}}}{(\det M^e)^2}\big)$, *if p arises from* $12|34$.

(ii) $\lim_{N\to\infty}\mathbb{E}[\Delta_{detG}(F)] = \Delta_{detG}(p) = \big(\frac{4^4 M^e_{\mathtt{A},\mathtt{A}}M^e_{\mathtt{A},\mathtt{C}}M^e_{\mathtt{A},\mathtt{G}}M^e_{\mathtt{A},\mathtt{T}}}{(\det M^e)^2}, 0, \frac{4^4 M^e_{\mathtt{A},\mathtt{A}}M^e_{\mathtt{A},\mathtt{C}}M^e_{\mathtt{A},\mathtt{G}}M^e_{\mathtt{A},\mathtt{T}}}{(\det M^e)^2}\big)$, *if p arises from* $13|24$.

(iii) $\lim_{N\to\infty}\mathbb{E}[\Delta_{detG}(F)] = \Delta_{detG}(p) = \big(\frac{4^4 M^e_{\mathtt{A},\mathtt{A}}M^e_{\mathtt{A},\mathtt{C}}M^e_{\mathtt{A},\mathtt{G}}M^e_{\mathtt{A},\mathtt{T}}}{(\det M^e)^2}, \frac{4^4 M^e_{\mathtt{A},\mathtt{A}}M^e_{\mathtt{A},\mathtt{C}}M^e_{\mathtt{A},\mathtt{G}}M^e_{\mathtt{A},\mathtt{T}}}{(\det M^e)^2}, 0\big)$, *if p arises from* $14|23$.

Proof. Firstly, the inference measure verifies Property 1 because the determinant is invariant for transposition, therefore applying points *(iv)* and *(v)* in Proposition 3.23 the property is verified.

To prove Property 3 is verified, consider F frequency with $F \sim \mathrm{MultiNom}(p, N)$, with $p = \varphi_T(\theta)$ a distribution on a 4-leaved tree T with $\theta = \{\pi^r, M^1, M^2, M^3, M^4, M^e\}$. Using the multivariate Delta method (or Taylor's expansion) we have:

$$\lim_{N\to\infty}\mathbb{E}[\det G_{ij|kl}(F)] = \det G_{ij|kl}(Np).$$

Analogously, if we now consider $F' \sim \mathrm{MultiNom}(p', N)$ with $p' = \varphi_T(\theta')$ arising from T with $\theta' = \{\pi^r, M^1N_1, M^2N_2, M^3N_3, M^4N_4, M^e\}$, we have:

$$\lim_{N\to\infty}\mathbb{E}[\det G_{ij|kl}(F')] = \det G_{ij|kl}(Np').$$

Note that the entries of $G_{ij|kl}(p)$ are quotients of homogeneous polynomials of degree 2 on the coordinates of p. In particular, we have

$$G_{ij|kl}(Np) = G_{ij|kl}(p) \text{ and } G_{ij|kl}(Np') = G_{ij|kl}(p').$$

Because of Lemma 4.3, the determinant of G is a Markov invariant with $\lambda = 1$ because it is a combination of entries of $G_{ij|kl}$, therefore for two distributions p and p' holds that:

$$\det G_{ij|kl}(p') = \det G_{ij|kl}(p).$$

Therefore the determinant verifies Property 3,

$$\lim_{N\to\infty} \mathbb{E}[\det G_{ij|kl}(F)] = \lim_{N\to\infty} \mathbb{E}[\det G_{ij|kl}(F')].$$

The limit expectations are deduced from Proposition 3.23. □

With this idea of the G matrix for the correct topology having all its entries equal to one, it is intuitive to use the Frobenius distance to $\mathbf{1} \in \mathcal{M}_{4\times4}(\mathbb{R})$ as inference measure.

Definition 4.6. Frobenius distance from G matrices to 1 ($dG1$). Let $\mathbf{1} \in \mathcal{M}_{4\times4}(\mathbb{R})$ be the matrix with all entries equal to 1.

$$\Delta_{dG1}(F) = \big(\text{dist}_{Frob}(G_{12|34}(F), \mathbf{1}), \text{dist}_{Frob}(G_{13|24}(F), \mathbf{1}), \text{dist}_{Frob}(G_{14|23}(F), \mathbf{1})\big)$$

Proposition 4.7. *$\Delta_{dG1}(F)$ is a quartet inference measure that satisfies Properties 1 and 3 with $\lambda = 1$. Moreover the limit expectation of $\Delta_{dG1}(F)$ when $F \sim P$ is:*

(i) $\lim_{N\to\infty} \mathbb{E}[\Delta_{dG1}(F)] = \Delta_{dG1}(P) = \big(0, \sum_{x,y\in\mathtt{C},\mathtt{G},\mathtt{T}}(\frac{m^e_{x+y}}{m^e_x m^e_y} - 1)^2, \sum_{x,y\in\mathtt{C},\mathtt{G},\mathtt{T}}(\frac{m^e_{x+y}}{m^e_x m^e_y} - 1)^2\big)$, *if P arises from* 12|34.

(ii) $\lim_{N\to\infty} \mathbb{E}[\Delta_{dG1}(F)] = \Delta_{dG1}(P) = \big(\sum_{x,y\in\mathtt{C},\mathtt{G},\mathtt{T}}(\frac{m^e_{x+y}}{m^e_x m^e_y} - 1)^2, 0, \sum_{x,y\in\mathtt{C},\mathtt{G},\mathtt{T}}(\frac{m^e_{x+y}}{m^e_x m^e_y} - 1)^2\big)$, *if P arises from* 13|24.

(iii) $\lim_{N\to\infty} \mathbb{E}[\Delta_{dG1}(F)] = \Delta_{dG1}(P) = \big(\sum_{x,y\in\mathtt{C},\mathtt{G},\mathtt{T}}(\frac{m^e_{x+y}}{m^e_x m^e_y} - 1)^2, \sum_{x,y\in\mathtt{C},\mathtt{G},\mathtt{T}}(\frac{m^e_{x+y}}{m^e_x m^e_y} - 1)^2, 0\big)$, *if P arises from* 14|23.

Proof. The argument in the proof for Proposition 4.5 also proves that Δ_{dG1} satisfies Properties 1 and 3. The limit expectation follows from Proposition 3.23. □

Remark 4.8. For these two inference measures $detG$ and $d1G$, the method we follow is to compute the measure and then choose the topology with the smaller score as the one that better fits the data.

Finally, after Definition 3.15 one has that $B^{\mathtt{A}}_{ij}(p)$ has rank 1 if p arises from the $ij|kl$ topology. We need the following result in order to determine the distance of a matrix to the space of matrices with a rank k.

Theorem 4.9. ***(Eckart-Young)*** *Let M be an $m \times n$ matrix with singular value decomposition $M = U\Sigma V^t$. Let $A \in \mathcal{M}_{m\times n}$ have rank k. Then:*

$$||M - A||_{Frob} \geq ||M - U_k\Sigma_k V_k^t||_{Frob} = \sqrt{\sum_{i=k+1}^{m} \sigma_i^2}$$

where U_k, Σ_k, V_k are the truncations to the first k columns.

Therefore, since our $B_{\mathtt{A}}^{ij}$ are 4×4 and we want to know which one is the closer to rank 1 in order to infer the correct tree topology.

$$\text{dist}_{Frob}(B_{\mathtt{A}}^{ij}, \{X|\text{rank}(X) \leq 1\}) = \min_{X|\text{rank}(X)\leq 1} ||B_{\mathtt{A}}^{ij} - X||_{Frob} = \sqrt{\sigma_2^2 + \sigma_3^2 + \sigma_4^2},$$

where σ_i are the singular values of $B_{\mathtt{A}}^{ij}$. Since the three flattening matrices have the same entries but in different order we have that their Frobenius norms

$$||B_{\mathtt{A}}^{12}||_{Frob} = ||B_{\mathtt{A}}^{13}||_{Frob} = ||B_{\mathtt{A}}^{14}||_{Frob} = \sqrt{\sigma_1^2 + \sigma_3^2 + \sigma_4^2}.$$

Therefore, if we want to find the minimum distance and the Frobenius norm is constant in the matrices, the following two problems are equivalent: minimizing $\sigma_2^2 + \sigma_3^2 + \sigma_4^2$ and maximizing σ_1^2.

Since it is known that $||B_{\mathtt{A}}^{ij}||_2 = \sigma_1$, we can propose the following inference measure:

Definition 4.10. Euclidean norm of the $B_{\mathtt{A}}^{ij}$ matrices ($n2B$).

$$\Delta_{n2B}(F) = \frac{1}{N}\left(||B_{\mathtt{A}}^{12}(F)||_2, ||B_{\mathtt{A}}^{13}(F)||_2, ||B_{\mathtt{A}}^{14}(F)||_2\right)$$

Remark 4.11. As it is made apparent in the justification for this inference measure, in this case we will take as correct the topology that presents a higher score, as opposed to the other two measures.

4.3 On the simulation of alignments

Once a method of quartet reconstruction is presented, it makes sense to find a systematic way to to test its efectiveness in a different range of trees and to understand what tree topologies can systematically produce weaknesses in reconstruction accuracy. The *tree space*, or the set of all phylogenetic trees, is huge, even when restricting it to quartets. To simulate alignments corresponding to species evolving by a certain phylogenetic tree, we need to define a subspace of the tree space small enough so that we can run simulations on it, but varied enough so that we are not leaving out some patological trees, which could lead to a misleading evaluation of the accuracy of our reconstruction method.

Definition 4.12. The *Huelsenbeck tree space* (see [Hue95]) is the subspace of the tree space obtained by considering the phylogenetic trees $T = 12|34$ with two branch lengths a and b where a is the length of the interior edge and the edge of external branches leading to 2 and 4 while b is the length of the external branches leading to 1 and 3, see Figure 4.1.

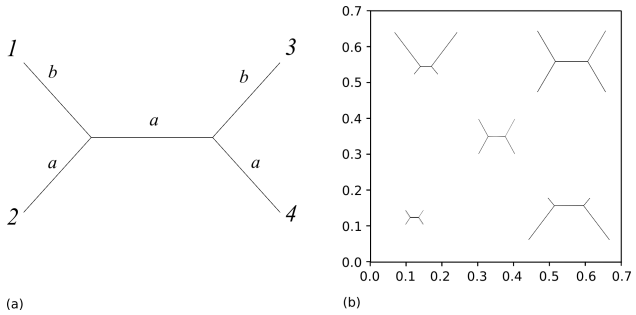


Figure 4.1: (a) Generic tree in the Huelsenbeck space. (b) Overview of a section of the Huelsenbeck space.

The Huelsenbeck subspace not only provides a solid sample of the tree space, but also allows us to represent results on the accuracy of a method in two dimensions, since it is indexed by two parameters.

As seen in Lemma 2.13, under certain hypothesis which are supported by the $K81$ model, we can find a relationship between branch length and transition matrices. The algorithm for generating $K81$ transition matrices given a branch length can be found in [CK13]. To go from the transition matrices of a Huelsenbeck phylogenetic tree to an alignment of length N coherent with such tree, the `GenNon-h` algorithm, developed in [KC12], is used. The general flow of the simulation is shown in Figure 4.2.

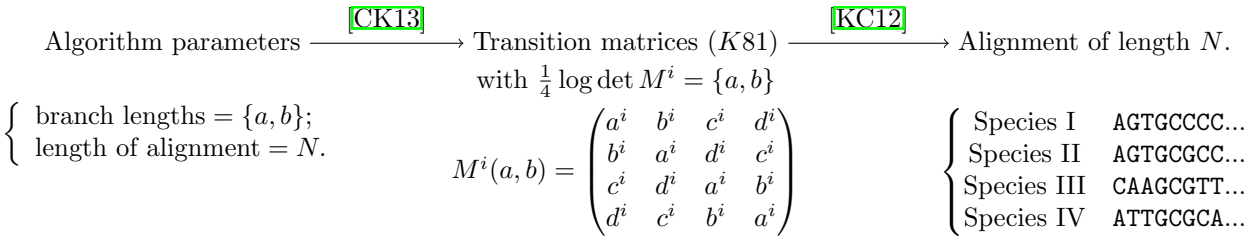


Figure 4.2: Flow of a simulation of alignments.

To test our methods, we vary the parameters a and b in the interval $[0, 1.5]$ and the generate 100 alignments of a given length N. Then we generate the frequency array $F = (F_{ijkl})$ for each nucleotide configuration in the columns of the alignment, perform the algebraic transformations discussed before and finally test the proposed quartet inference measures. The accuracy of a measure is quantified by the amount of times it reconstructs the correct tree topology.

4.4 Overview of the code

To compute the algebraic transformations for the proposed inference measures we have written a function in the `R` programming language that, given a relative frequency array arising from an alignment of 4 species returns the topology a certain inference measure is more confident in. Since the simulated data we received were alignments and not frequencies, we use the `read.fasta` function in the `seqinr` package to read them and then simply count the frequencies of each nucleotide configuration (more details in the Appendix).

The main function `QuartetTopogy` has the following structure:

Input: Relative frequency array `f` (1×256) from an alignment of 4 species and the choice of inference measure.

Output: Topology a quartet inference measure is more confident in. (`1` for 12|34, `2` for 13|24, `3` for 14|32)

1. Compute the flattenings for the 3 topologies.
2. Transform the flattenings to Fourier coordinates.
3. Compute the $B_{ij}^{\mathtt{A}}$ block for each topology.
4. Compute the 6 marginals.
5. Transform the marginals to Fourier coordinates.

6. Compute $G_{ij|kl}$ matrix for each topology.
7. Compute the quartet inference measure.
8. Return topology with the most confidence.

We will refer to the Appendices to find the whole commented code, but we give some intuitions about each of the previous steps of the code now, including some exherts of it. The explanation assumes knowledge of the commands used in the R language.

1. **Compute the flattenings for the 3 topologies.** This is done through a character vector of the form $\texttt{pt} = (\texttt{AAAA}, \texttt{AAAC}, \texttt{AAAG}, ..., \texttt{TTTT})$ which is used to index the frequency array `f` according to each flattening. For example for the 12|34 flattening we would use the loop:

```
rn <- c("AA","AC","AG","AT","CA","CC","CG","CT","GA","GC","GG","GT","TA","TC","TG","TT")
 for(i in 1:16){
   pf1234[rn[i],] <- f[str_which(pt,str_c(rn[i],"[:upper:][:upper:]"))]
 }
```

   where the flattenings are initialized as data frames indexed by `rn`.

2. **Transform the flattenings to Fourier coordinates.** We use the Hadamard matrix with each flattening to transform it to Fourier coordinates, for example:

```
pf1234fou <- kronecker(solve(H),solve(H))%*%as.matrix(pf1234)...
...%*%t(kronecker(solve(H),solve(H)))
```

   The property used in this computation is not explicitly introduced in this work but can be seen in [Cas18].

3. **Compute the $B^{\mathtt{A}}_{ij}$ block for each topology.** This is done by selecting the elements of the Fourier flattenings whose Σ indexation sums $\mathtt{A}$ (understanding Σ as a group) and indexing those correctly into the $B^{\mathtt{A}}_{ij}$ matrices, as described in 3.15.

4. **Compute the 6 marginals.** This is following using the initial definition 3.17 and summing the elements (using the 12|34 flattening). For example, for the $1-2$ marginal we sum like:

```
marg12 <- transmute(pf1234,marginal12=AA+AC+AG+AT+CA+CC+CG+CT+GA+GC+GG+GT+TA+TC+TG+TT) %>%
    t() %>%
    matrix(4,4)
```

   where the `transmute` function sums the rows of the flattening, which is then transposed and transformed into a matrix. The `%>%` is the pipe operator, that passes the element to its left as argument to the function below.

5. **Transform the marginals to Fourier coordinates.** Through the Hadamard matrix, as before. For example:

```
foumarg12 <- solve(H) %*% marg12 %*% solve(H) %>%
    diag()
```

   The resulting matrices are diagonal, therefore they are saved in a 4 position vector.

6. **Compute the $G_{ij|kl}$ matrix for each topology.** As introduced in Definition 3.20 they are computed with two auxiliary matrices, in the 12|34 case we use:

```
G12rightA <- blockdiag(1/foumarg34[1],1/foumarg34[2],1/foumarg34[3],1/foumarg34[4])
G12leftA <- blockdiag(1/foumarg12[1],1/foumarg12[2],1/foumarg12[3],1/foumarg12[4])
G12 <- G12leftA %*% bA12 %*% G12rightA
```

7. **Compute the quartet inference measure.** This step changes depending on the inference measure we want to implement. For example for the detG measure we would use the following:

```
Ddet <- map(list(G12,G13,G14),det) %>%
    unlist() %>%
    which.min()
```

8. **Return topology with the most confidence.** Following the previous example with the determinant, we would just return what topology corresponds with the minimum (or maximum for $n2B$) value of the measure:

```
return(Ddet)
```

4.5 Results on simulated data

To test the quartet inference measures introduced in 4.2 we consider a subset of the Huelsenbeck space, in the fashion of Figure 4.1(b). The simulated data covers $a \in [0, 1.5]$ and $b \in [0, 1.5]$ with steps of 0.1. For each pair (a, b), 100 sets of 4 alignments with $N = 1000$ are generated and their frequencies calculated. The accuracy of the inference measure for the 100 trials is represented in a scale of greys for every square, which corresponds with the (a, b) pair. We use the methods described in Remarks 4.8 and 4.11. The results, plotted with the `ggplot` function in `R`, shown in Figure 4.3.

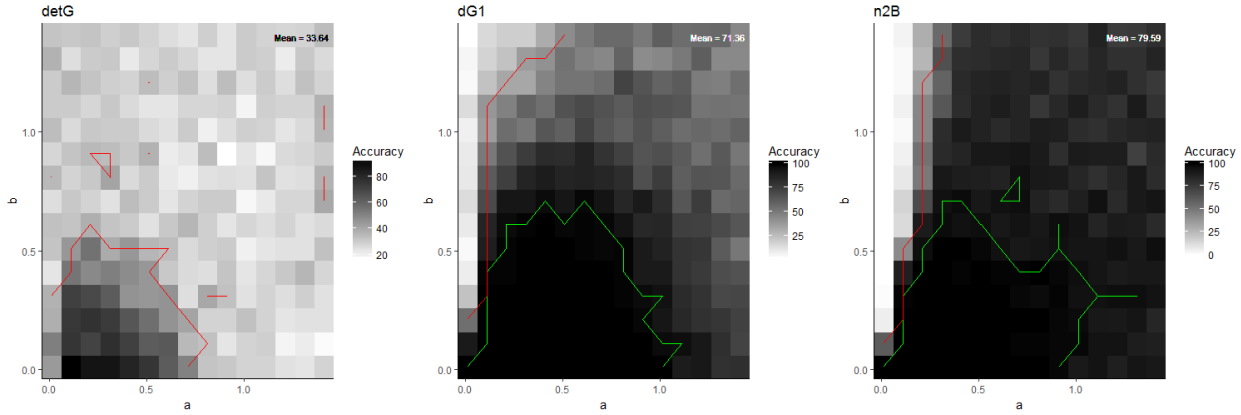


Figure 4.3: Performance plots for inference measures 4.4 ($detG$), 4.6 ($dG1$) and 4.10 ($n2B$) respectively, with $N = 1000$. In red the contour for 33% accuracy (equivalent to choice at random) and in green for 95%.

The results show that measure 4.4 ($detG$) seems to be a very bad inference measure, since it performs worse than a choice at random for most of the space where we have tested it. We will delve on the reason behind this poor behaviour in section 4.6 and modify it accordingly in section 4.7.

For measures 4.6 ($dG1$) and 4.10 ($n2B$) we observe a plateau of over 95% accuracy and a solid performance average, but there is a symptomatic area of poor accuracy in prediction for both of the methods. This is due to the phenomena of long branch attraction, that occurs when a tree has two long and two short edges in relation with the central one, with one of each at each side of the partition. This can make most methods group as closer the species with similar branch lengths, failing to correctly predict the topology, see Figure 4.4.

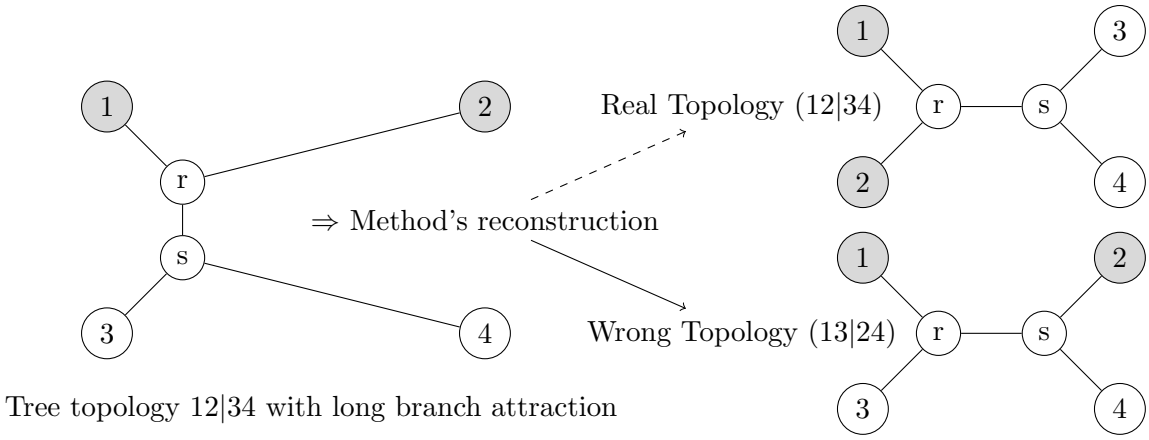


Figure 4.4: Example of long branch attraction

This highlights the importance of the choice of parameters a and b, since we do not want to omit patological cases in the tree space where the methods have trouble reconstructing the correct topology, giving us an incomplete picture of the performance.

4.6 Empirical bias

When we consider an alignment of length N, the absolute frequency F is a vector of samplings on the distribution of the configurations of the nucleotides at the leaves p:

$$F \sim \text{MultiNom}(p, N)$$

We can understand F as an *estimator* of $P = Np$, parameter in our statistical model, because given an alignment, we can calculate an estimate of P by counting how many times a configuration of nucleotide appears.

Definition 4.13. Let $\hat{\eta}$ be a estimator of the statistic η. Then the *bias* of $\hat{\eta}$ relative to η is defined as:

$$\text{bias}_P[\hat{\eta}] = \mathbb{E}[\hat{\eta}|\eta] - \eta$$

where $\mathbb{E}[\hat{\eta}|\eta]$ denotes the expected value of $\hat{\eta}$ subject to the value of η.

In general a quartet inference measure is defined by

$$\Delta(F) = (R_{12} = \delta_{12}(F), R_{13} = \delta_{13}(F), R_{14} = \delta_{14}(F))$$

where $\delta_{ij} : F \longrightarrow R_{ij}$ are functions that given F and through transformations as the ones shown in this work, generate a statistically interpretable confidence in the reconstruction. What can we say about the bias of $\mathbb{E}[\Delta(F)]$ for the inference measures developed when comparing them to their expected value on the limit $N \longrightarrow \infty$? We will take a practical approach to this question by analyzing how the scores evolve when N increases in measure 4.4 ($detG$):

$$\Delta_{detG}(F) = \big(\det G_{12|34}(F), \det G_{13|24}(F), \det G_{14|23}(F)\big).$$

For this case study we assume that T has the 12|34 topology. Then, as proven in Proposition 4.5 we have:

$$\lim_{N\to\infty} \mathbb{E}[\Delta_{detG}(F)] = \Delta_{detG}(P).$$

Therefore,

$$\lim_{N\to\infty} \text{bias}[\Delta_{detG}(F)] = 0.$$

Nevertheless, for small N we could have huge bias and the convergence to 0 could be really slow. It could be useful to generate this score for alignments with different N and asses its qualitative behaviour, and how fast this bias is reduced when N is increased.

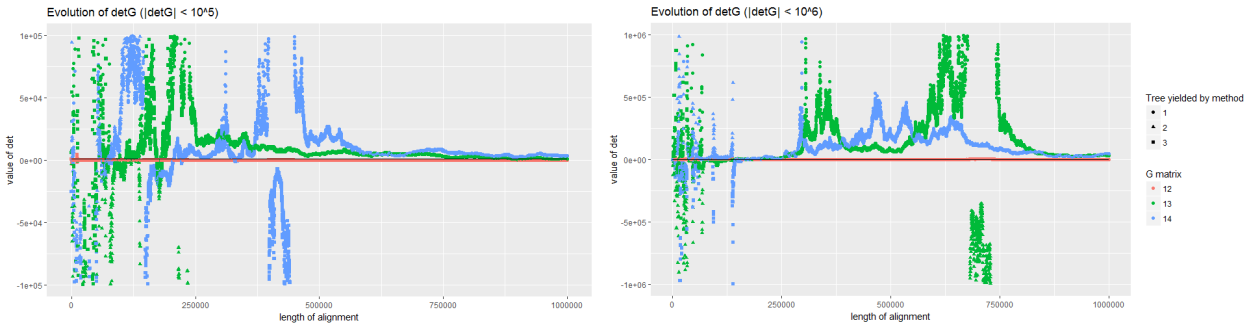


Figure 4.5: Evolution of the scores for the *detG* inference measure for subalignments of two different alignments of length $N = 1000000$. In black the expected value for the scores of the 13|24 and 14|23 topologies, which are equal to 1281.234 and 587.1062 respectively.

To get a qualitative idea of the behaviour, we simulated an alignment A of length $N = 1000000$ for 4 species, using the techniques in section 4.3. To see how the scores evolve, we use a loop (starting with $N_0 = 100$) where at each iteration we consider a subalignment of A of length $N_i = N_{i-1} + 100$, where N_{i-1} was the length at the previous iteration (the same alignment but 100 nucleotides longer) and compute the scores for the *detG* inference measure. The results are shown in Figure 4.5. The plots correspond to two different alignments of length $N = 1000000$, but it is pruned to omit big scores to provide better readability. There are several things to be observed:

- The score for the correct topology (in this data 12|34) converges rapidly towards 0 and stays there as N increases.
- The convergence towards the expected value (in black) is slow and erratic, and only seems to happen for N larger than what is the alignment length attainable in practice.
- For small values of N the behaviour of the scores for the false topologies is very erratic, and takes negatives values.
- As we increase N, the scores still have oscillations but in positive values. In both cases islands of negatives values appear.

The huge values of the scores come from dividing by the marginals in Fourier coordinates in the $G_{ij|kl}(p)$ matrices, because they can be very close or in some cases 0 for the incorrect topologies. To compensate for the negative values, which are what is causing the method to fail since we look for the minimum value in $\Delta_{detG}(F)$, we will consider the absolute value of the determinants instead, in section 4.7.

4.7 The adG inference measure

As seen in the previous section, the $detG$ method has some undesired behaviours, both because it takes negative values and because it fails to converge to the expected values for meaningful and practical lengths of alignments. Since this latter behaviour is for the scores of the wrong topologies, will not affect the performance of the method as the first, once we proceed naturally to consider the absolute value of the determinant as a new inference measure adG.

Definition 4.14. Absolute value of the determinant of the G matrices

$$\Delta_{adG}(F) = \left(|\det G_{12|34}(F)|, |\det G_{13|24}(F)|, |\det G_{14|23}(F)|\right)$$

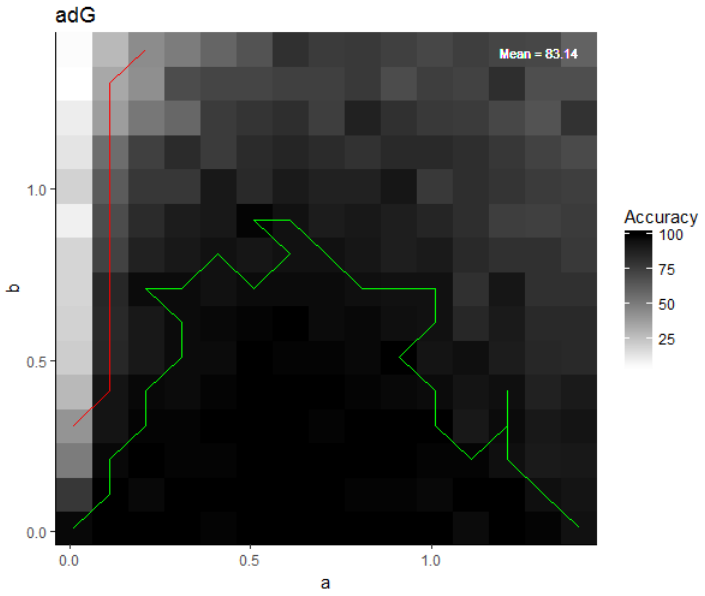


Figure 4.6: Performance plot for inference measure 4.14 (adG), with $N = 1000$. In red the contour for 33% accuracy (equivalent to choice at random) and in green for 95%.

The absolute value of the determinant of the G matrices is the best performing method of the ones tried in this work, both presenting the bigger plateau of 95% accuracy and having the best mean overall. Not only that but, when we compare its performance test with the one for the $n2B$ or $dG1$ measures, the region in which the phenomenon of long branch attraction forces the method under 33% accuracy is slimmer, and reduced only to the squares where this behaviour is more extreme.

This result of the adG inference measure on the Huelsenbeck space for sequences of length $N = 1000$ is comparable with the methods most widely used by biologists: neighbour-joining and maximum likelihood, which have a performance of around 80% on this tree space (see [CFS16], Figure 3).

4.8 Example of quartet reconstruction with real data alignments

In this section we will apply the algebraic method *adG* for quartet reconstruction to real alignments of a lineages of water beetles of the genus *Deronectes*. The alignments are the ones used to construct the phylogenetic trees in [RGV16] and were gently made accessible by Ignacio Ribera (IBE-CSIC) in order to be used in this work. Aquatic beetles are one of the best examples of how phylogenetic trees can unearth information not only about the evolution of the species themselves but also of the geographical enviornment that surrounded such process. This is due to aquatic beetles having precise needs for survival and having certain characteristics regarding range expansion. In [RGVB+16] it is exposed how major geological and climatic events influenced the history of the genus, and sometimes these kind of relationships can give valuable information or hints about when and how events like such happened.

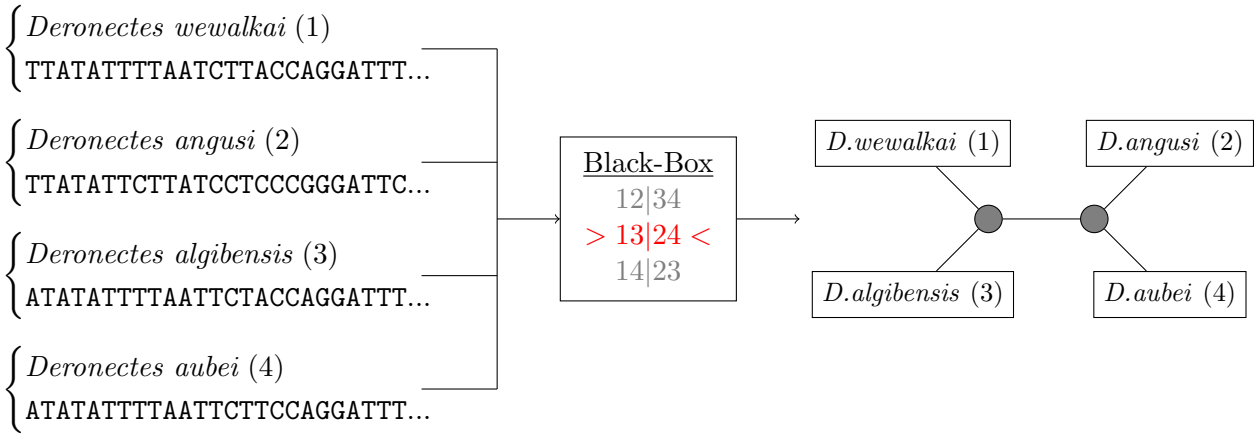


Figure 4.7: Example of implementation of the algorithm to reconstruct the phylogenetic relationship between 4 species of the genus *Deronectes*.

Given 4 alignments of length $N = 801$ for 4 species of the *Deronectes* genus, we use the *adG* inference measure to reconstruct their phylogenetic relationship, as shown in Figure 4.7. The method successfully reconstructs the quartet, as can be checked in [RGVB+16]. By permuting the order in which we label and introduce the species into our `R` function (the Black Box) we get covariant outputs, for example, for the labeling:

$$\{D.wewalkai\ (1),\ D.algibensis\ (2),\ D.angusi\ (3),\ D.aubei\ (4)\}$$

the returned topology is 12|34, as expected by the reasoning introduced in section 2.2.

Even though in this particular case the phylogenetic tree was successfully reconstructed, we have evidence that the $K81$ model may not be adequate for this data, since if we count the frequencies of $\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}$ in the alignment, we get that their distribution is:

$$f_{\mathtt{A}} = 0.298377,\ \ f_{\mathtt{C}} = 0.156367,\ \ f_{\mathtt{G}} = 0.1413858,\ \ f_{\mathtt{T}} = 0.4038702.$$

The $K81$ model has the uniform distribution as its stationary and root distribution, and the data we are using is not close to this distribution.

Chapter 5

Conclusion and future work

In this work we have aimed to discuss the theoretical foundations of quartet reconstruction with algebraic techniques and then implement some new quartet inference measures and test their performance on a subset of the Huelsenbeck tree space. At the end of the project, we have successfully fulfilled the following:

1. *Understood the usefulness and limitations of both phylogenetic trees and the $K81$ model of nucleotide substitution.* Throughout the bibliographical research we have come across papers using phylogenetic trees to model the evolution of species in various ways (see [RGV16] for example) but also realized the difficulty in reconstruction, especially when the sequences have small length. In section 4.8 we have also seen that the $K81$ model may not be suitable for some sets of data.

2. *Compiled the results on the bibliography for quartet reconstruction.* The results were spread between bibliography focusing in the algebraic aspect of the transformations ([AR07]) and others focusing on a more statistical point of view ([STHJ16]). We also have tried to give intuitive reasoning behind the steps followed in this work from the alignments to the inference measures.

3. *Given alternative proofs for some of the results in [CCFS].* In order to make some proofs more intuitive, we have used the Markov action to follow a set structure throughout the work: first modifying the transition matrices in the trees and then computing the probabilities to reach our conclusions.

4. *Implemented in the `R` language the algebraic transformations needed to use the quartet inference measures proposed in [CCFS].* We have developed a function that performs the transformations described in this work and can be modified to test inference measures based in the $B^{ij}_{\mathtt{A}}$ and $G_{ij|kl}$ matrices. To do so we have used several features of the library `tidyverse`, for example all the plots are done with the `ggplot` function.

5. *Tested the accuracy of such measures with simulated data.* The data was provided by Marta Casanellas using the algorithms described in section 4.3. The inference measures performed reasonably well except for $detG$, because negative values would appear in the score.

6. *Propose modifications to such measures.* The adG inference measure is an attempt to improve $detG$ which ended up being the best performing measure. There were more modifications tried to the proposed measures but none of them yielded relevant results.

7. *Made a preliminary study on the behaviour and bias of the detG and adG inference measures.* We tried to answer the question of how good were the inference measures in relation to the properties in [STHJ16]. We analyzed the bias for some cases of the *detG* measure in order to asses its general behaviour. The results show erratic behaviour for small alignments and a tendency to the expected values as N increases.

We have seen that quartet inference measures based on algebraic tools might have a performance of 83% on the Huelsenbeck tree space for sequences of length 1000. This is compatible with the methods most widely used by biologists: neighbour-joining and maximum-likelihood, which have a performance of around 80% on this tree space (see [CFS16], Figure 3). Therefore it is interesting to pursue this study in a future work.

The specific things that were not tackled in this project and are left as future works:

1. Give a more extensive description on the huge bias for the *adG* inference measure. Specifically we should deal with the problem that the denominators of the matrices $G_{ij|kl}$ might be close to 0.

2. Perform an analysis on the expectation for the inference measures.

3. Give an in-depth comparison between the methods proposed and neighbour-joining or maximum likelihood.

4. Implement the inference measures in `C++` in order to complement and improve existing code, such as reconstruction algorithms based in quartets.

5. Partake in the authorship of a coming version of [CCFS].

Chapter 6

Acknowledgments

I would like to thank my parents Marisol and José Ángel for their support throughout this work, my advisor Marta Casanellas for her dedication and time, my very good friends Miquel and Dome for sitting next to me throughout the writing process, Caterina for her impressive illustration and Ignacio Ribera for introducing to me the subject of phylogenetics.

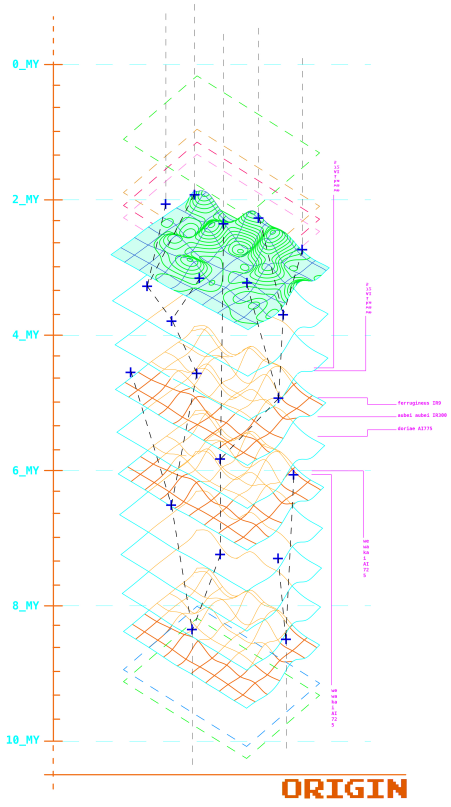


Figure 6.1: Interpretation of a Phylogenetic Tree by artist Caterina Miralles.

Bibliography

[AR07] E. S. Allman and J. A. Rhodes. Phylogenetic invariants. In O. Gascuel and M. A. Steel, editors, *Reconstructing Evolution*. Oxford University Press, 2007.

[AR10] E. S. Allman and John A. Rhodes. *Mathematical models in biology: an introduction*. Cambridge University Press, 2010.

[BH87] Daniel Barry and J. A. Hartigan. Statistical analysis of hominoid molecular evolution. *Statist. Sci.*, 2(2):191–207, 05 1987.

[Cas18] M. Casanellas. El modelo evolutivo de kimura, un enlace entre el álgebra, la biología i la estadística. *La Gaceta de la Real Sociedad Matemática Española*, 21, 2018.

[CCFS] M. Casanellas, L. Cifuentes, and J. Fernández-Sánchez. Application of algebraic techniques to phylogenetic reconstruction. *[in preparation]*.

[CFS16] M. Casanellas and J. Fernández-Sánchez. Invariant versus classical quartet inference when evolution is heterogeneous across sites and lineages. *Systematic Biology*, 65(2):280–291, 2016.

[CGS05] M. Casanellas, L. D. García, and S. Sullivant. Catalog of small trees. In L. Patcher and B. Sturmfels, editors, *Algebraic Statistics for computational biology*. Cambridge University Press, 2005.

[Cif15] L. Cifuentes. Application of algebraic techniques to phylogenetic reconstruction, 2015. Advisors: M. Casanellas and J. Fernández-Sánchez.

[CK13] M. Casanellas and A. M. Kedzierska. Generating markov evolutionary matrices for a given branch length. *Linear Algebra and its Applications*, 438(5):2484 – 2499, 2013.

[Dar59] C. Darwin. *On the Origin of Species by Means of Natural Selection, or the Preservation of Favoured Races in the Struggle for Life*. 1859.

[Hue95] J. P. Huelsenbeck. Performance of phylogenetic methods in simulation. *Systematic Biology*, 44(1):17–48, 1995.

[IC13] E. Ibáñez and M. Casanellas. Em for phylogenetic topology reconstruction on nonhomogeneous data. *BMC evolutionary biology*, 14, 2013.

[KC12] A. Kedzierska and M. Casanellas. Gennon-h: Generating multiple sequence alignments on nonhomogeneous phylogenetic trees. *BMC bioinformatics*, 13, 2012.

[RG01] Vincent R. and Olivier G. Quartet-based phylogenetic inference: Improvements and limits. *Molecular Biology and Evolution*, 18(6):1103–1116, 2001.

[RGV16] I. Ribera and D. García-Vázquez. The origin of widespread species in a poor dispersing lineage (diving beetle genus deronectes). *PeerJ*, 4, 09 2016.

[RGVB+16] I. Ribera, D. García-Vázquez, D. T. Bilton, R. Alonso, L. F. Valladares, C. J. Benetti, and J. Garrido. Reconstructing ancient mediterranean crossroads in deronectes diving beetles. *Journal of Biogeography*, 43, 08 2016.

[STHJ16] J. G. Sumner, A. Taylor, B. Holland, and D. P. Jarvis. Developing a statistically powerful measure for phylogenetic tree inference using phylogenetic identities and markov invariants. *Journal of Mathematical Biology*, 75, 08 2016.

Chapter 7

Appendices

7.1 Code: Inference Measures

```
library(tidyverse)
library(seqinr)
library(purrr)
library(stringr)
library(linpk)
library(ggplot2)

QuartetTopology <- function(f){
  #Auxiliary vector with the character indexation of the probability vector.
  pt <- c("AAAA", "AAAC", "AAAG", "AAAT", "AACA", "AACC", "AACG", "AACT", "AAGA", "AAGC", "AAGG", "AAGT", "AATA", "AATC", "AATG",
          "AATT", "ACAA", "ACAC", "ACAG", "ACAT", "ACCA", "ACCC", "ACCG", "ACCT", "ACGA", "ACGC", "ACGG", "ACGT", "ACTA", "ACTC",
          "ACTG", "ACTT", "AGAA", "AGAC", "AGAG", "AGAT", "AGCA", "AGCC", "AGCG", "AGCT", "AGGA", "AGGC", "AGGG", "AGGT", "AGTA",
          "AGTC", "AGTG", "AGTT", "ATAA", "ATAC", "ATAG", "ATAT", "ATCA", "ATCC", "ATCG", "ATCT", "ATGA", "ATGC", "ATGG", "ATGT",
          "ATTA", "ATTC", "ATTG", "ATTT", "CAAA", "CAAC", "CAAG", "CAAT", "CACA", "CACC", "CACG", "CACT", "CAGA", "CAGC", "CAGG",
          "CAGT", "CATA", "CATC", "CATG", "CATT", "CCAA", "CCAC", "CCAG", "CCAT", "CCCA", "CCCC", "CCCG", "CCCT", "CCGA", "CCGC",
          "CCGG", "CCGT", "CCTA", "CCTC", "CCTG", "CCTT", "CGAA", "CGAC", "CGAG", "CGAT", "CGCA", "CGCC", "CGCG", "CGCT", "CGGA",
          "CGGC", "CGGG", "CGGT", "CGTA", "CGTC", "CGTG", "CGTT", "CTAA", "CTAC", "CTAG", "CTAT", "CTCA", "CTCC", "CTCG", "CTCT",
          "CTGA", "CTGC", "CTGG", "CTGT", "CTTA", "CTTC", "CTTG", "CTTT", "GAAA", "GAAC", "GAAG", "GAAT", "GACA", "GACC", "GACG",
          "GACT", "GAGA", "GAGC", "GAGG", "GAGT", "GATA", "GATC", "GATG", "GATT", "GCAA", "GCAC", "GCAG", "GCAT", "GCCA", "GCCC",
          "GCCG", "GCCT", "GCGA", "GCGC", "GCGG", "GCGT", "GCTA", "GCTC", "GCTG", "GCTT", "GGAA", "GGAC", "GGAG", "GGAT", "GGCA",
          "GGCC", "GGCG", "GGCT", "GGGA", "GGGC", "GGGG", "GGGT", "GGTA", "GGTC", "GGTG", "GGTT", "GTAA", "GTAC", "GTAG", "GTAT",
          "GTCA", "GTCC", "GTCG", "GTCT", "GTGA", "GTGC", "GTGG", "GTGT", "GTTA", "GTTC", "GTTG", "GTTT", "TAAA", "TAAC", "TAAG",
          "TAAT", "TACA", "TACC", "TACG", "TACT", "TAGA", "TAGC", "TAGG", "TAGT", "TATA", "TATC", "TATG", "TATT", "TCAA", "TCAC",
          "TCAG", "TCAT", "TCCA", "TCCC", "TCCG", "TCCT", "TCGA", "TCGC", "TCGG", "TCGT", "TCTA", "TCTC", "TCTG", "TCTT", "TGAA",
          "TGAC", "TGAG", "TGAT", "TGCA", "TGCC", "TGCG", "TGCT", "TGGA", "TGGC", "TGGG", "TGGT", "TGTA", "TGTC", "TGTG", "TGTT",
          "TTAA", "TTAC", "TTAG", "TTAT", "TTCA", "TTCC", "TTCG", "TTCT", "TTGA", "TTGC", "TTGG", "TTGT", "TTTA", "TTTC", "TTTG",
          "TTTT")

  # Flattenings ----

  #Initializing the three Flattenings
  a <- rep(0,16)
  rn <- c("AA","AC","AG","AT","CA","CC","CG","CT","GA","GC","GG","GT","TA","TC","TG","TT")
  pf <- data.frame(rnames = rn,AA=a,AC=a,AG=a,AT=a,CA=a,CC=a,CG=a,CT=a,GA=a,GC=a,GG=a,GT=a,TA=a,TC=a,TG=a,TT=a)
  pf <- column_to_rownames(pf,"rnames")
  pf1234 <- pf
  pf1324 <- pf
  pf1423 <- pf

  #Uses character vector to index correctly the elements of p into the flattening
  for(i in 1:16){
    pf1234[rn[i],] <- f[str_which(pt,str_c(rn[i],"[:upper:][:upper:]"))]
    pf1324[rn[i],] <- f[str_which(pt,str_c(substr(rn[i],1,1),"[:upper:]",substr(rn[i],2,2),"[:upper:]"))]
    pf1423[rn[i],] <- f[str_which(pt,str_c(substr(rn[i],1,1),"[:upper:]","[:upper:]",substr(rn[i],2,2)))]
  }
  rm(a,pf,i,rn)
  #OBS: Substituting p with pt yields the theoretical flattenings (with characters)

  # Flattenings in Fourier coordinates ----

  H <- matrix(c(1,1,1,1,1,1,-1,-1,1,-1,1,-1,1,-1,-1,1),4,4)
  pf1234fou <- kronecker(solve(H),solve(H))%*%as.matrix(pf1234)%*%t(kronecker(solve(H),solve(H)))
  pf1324fou <- kronecker(solve(H),solve(H))%*%as.matrix(pf1324)%*%t(kronecker(solve(H),solve(H)))
  pf1423fou <- kronecker(solve(H),solve(H))%*%as.matrix(pf1423)%*%t(kronecker(solve(H),solve(H)))

  # Block A for each topology in the Fourier flattening ----
  groupsum <- function(x){ #returns the sum (as a group) of the pair of elements in a row
    if(x == 1 | x == 6 | x == 11 | x == 16) return(1) #A
    if(x == 2 | x == 5 | x == 12 | x == 15) return(2) #C
    if(x == 3 | x == 8 | x == 9 | x == 14) return(3) #G
    if(x == 4 | x == 7 | x == 10 | x == 13) return(4) #T
  }
```

```
#Construct the blocks, first making the vector then reorganizing with t(matrix(.))
bA12 <- c()
bA13 <- c()
bA14 <- c()
for(i in 1:16){
  for(j in 1:16){
    si <- groupsum(i); sj <- groupsum(j);
    if(si == sj){
      if(si == 1){
        bA12 <- append(bA12,pf1234fou[i,j])
        bA13 <- append(bA13,pf1324fou[i,j])
        bA14 <- append(bA14,pf1423fou[i,j])
      }
    }
  }
}
rm(si,sj,i,j)

bA12 <- t(matrix(bA12,4,4))
bA13 <- t(matrix(bA13,4,4))
bA14 <- t(matrix(bA14,4,4))

## Computing marginals ----

marg12 <- transmute(pf1234,marginal12=AA+AC+AG+AT+CA+CC+CG+CT+GA+GC+GG+GT+TA+TC+TG+TT) %>% #as vector
  t() %>%
  matrix(4,4) #to access with margpos: marg12[[mp("AA")]]
marg34 <- transmute(as.data.frame(t(pf1234)),marginal34=AA+AC+AG+AT+CA+CC+CG+CT+GA+GC+GG+GT+TA+TC+TG+TT) %>%
  t() %>%
  matrix(4,4)
a1 <- c(1,2,3,4); c1 <- c(5,6,7,8); g1 <- c(9,10,11,12); t1 <- c(13,14,15,16)
a2 <- c(1,5,9,13); c2 <- c(2,6,10,14);g2 <- c(3,7,11,15); t2 <- c(4,8,12,16)
marg13 <- c(sum(pf1234[a1,a1]), sum(pf1234[a1,c1]), sum(pf1234[a1,g1]),sum(pf1234[a1,t1]), #A+
            sum(pf1234[c1,a1]), sum(pf1234[c1,c1]), sum(pf1234[c1,g1]),sum(pf1234[c1,t1]), #c+
            sum(pf1234[g1,a1]), sum(pf1234[g1,c1]), sum(pf1234[g1,g1]),sum(pf1234[g1,t1]), #g1+
            sum(pf1234[t1,a1]), sum(pf1234[t1,c1]), sum(pf1234[t1,g1]),sum(pf1234[t1,t1])) %>%  #T+
  t() %>%
  matrix(4,4)
marg24 <- c(sum(pf1234[a2,a2]), sum(pf1234[a2,c2]), sum(pf1234[a2,g2]),sum(pf1234[a2,t2]), #a2+
            sum(pf1234[c2,a2]), sum(pf1234[c2,c2]), sum(pf1234[c2,g2]),sum(pf1234[c2,t2]), #c2+
            sum(pf1234[g2,a2]), sum(pf1234[g2,c2]), sum(pf1234[g2,g2]),sum(pf1234[g2,t2]), #g2+
            sum(pf1234[t2,a2]), sum(pf1234[t2,c2]), sum(pf1234[t2,g2]),sum(pf1234[t2,t2])) %>%  #t2+
  t() %>%
  matrix(4,4)
marg14 <- c(sum(pf1234[a1,a2]), sum(pf1234[a1,c2]), sum(pf1234[a1,g2]),sum(pf1234[a1,t2]), #A+
            sum(pf1234[c1,a2]), sum(pf1234[c1,c2]), sum(pf1234[c1,g2]),sum(pf1234[c1,t2]), #C+
            sum(pf1234[g1,a2]), sum(pf1234[g1,c2]), sum(pf1234[g1,g2]),sum(pf1234[g1,t2]), #G+
            sum(pf1234[t1,a2]), sum(pf1234[t1,c2]), sum(pf1234[t1,g2]),sum(pf1234[t1,t2])) %>%  #T+
  t() %>%
  matrix(4,4)
marg23 <- c(sum(pf1234[a2,a1]), sum(pf1234[a2,c1]), sum(pf1234[a2,g1]),sum(pf1234[a2,t1]), #a2+
            sum(pf1234[c2,a1]), sum(pf1234[c2,c1]), sum(pf1234[c2,g1]),sum(pf1234[c2,t1]), #c2+
            sum(pf1234[g2,a1]), sum(pf1234[g2,c1]), sum(pf1234[g2,g1]),sum(pf1234[g2,t1]), #g2+
            sum(pf1234[t2,a1]), sum(pf1234[t2,c1]), sum(pf1234[t2,g1]),sum(pf1234[t2,t1])) %>%  #t2+
  t() %>%
  matrix(4,4)
rm(a1,c1,g1,t1,a2,c2,g2,t2)

## Marginals in Fourier coordinates ----

#OBS: They are diagonal matrices
foumarg12 <- solve(H) %*% marg12 %*% solve(H) %>%
  diag()
foumarg34 <- solve(H) %*% marg34 %*% solve(H) %>%
  diag()
foumarg13 <- solve(H) %*% marg13 %*% solve(H) %>%
  diag()
foumarg24 <- solve(H) %*% marg24 %*% solve(H) %>%
  diag()
foumarg14 <- solve(H) %*% marg14 %*% solve(H) %>%
  diag()
foumarg23 <- solve(H) %*% marg23 %*% solve(H) %>%
  diag()

## G matrix for each tree ----
#OBS: 1 ~ A, 2 ~ C, 3 ~ G, 4 ~ T
#G12|34
G12rightA <- blockdiag(1/foumarg34[1],1/foumarg34[2],1/foumarg34[3],1/foumarg34[4])
G12leftA <- blockdiag(1/foumarg12[1],1/foumarg12[2],1/foumarg12[3],1/foumarg12[4])
G12 <- G12leftA %*% bA12 %*% G12rightA
#G13|24
G13rightA <- blockdiag(1/foumarg24[1],1/foumarg24[2],1/foumarg24[3],1/foumarg24[4])
G13leftA <- blockdiag(1/foumarg13[1],1/foumarg13[2],1/foumarg13[3],1/foumarg13[4])
G13 <- G13leftA %*% bA13 %*% G13rightA
#G14|23
G14rightA <- blockdiag(1/foumarg23[1],1/foumarg23[2],1/foumarg23[3],1/foumarg23[4])
G14leftA <- blockdiag(1/foumarg14[1],1/foumarg14[2],1/foumarg14[3],1/foumarg14[4])
G14 <- G14leftA %*% bA14 %*% G14rightA

## Quartet Inference Measures ----
Ddet <- map(list(G12,G13,G14),det) %>%
  unlist() %>%
```

```
    #abs() %>%
    which.min()
  m1 <- matrix(c(1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1),4,4)
  Ddist <- list(norm(G12-m1,"F"),norm(G13-m1,"F"),norm(G14-m1,"F")) %>%
    unlist() %>%
    which.min()
  Dnorm2B <- map(list(bA12,bA13,bA14),norm,"2") %>%
    unlist() %>%
    which.max()
  return(c(Ddet,Ddist,Dnorm2B))


}

p <- 1
Nmethods <- 3
v <- c(1,11,21,31,41,51,61,71,81,91,101,111,121,131,141)
res.method <- matrix(0L,ncol=Nmethods+2,nrow=1)
for(a in v){
  for(b in v){
    ab.res <- matrix(ncol=Nmethods,nrow=100)
    for(i in 1:100) ab.res[i,] <- QuartetTopology(all.probs[[p]][i,])
    res.method <- rbind(res.method,c(a,b,sum(ab.res[,1] == 1),sum(ab.res[,2] == 1),sum(ab.res[,3] == 1)))#falta optimizar
    p = p+1;
    print("-")
    print(a)
    print(b)
  }
}

## Plots ----

f95 <- function(x){
  for(i in 1:length(x)){
    if(x[i] < 95) x[i] = 0;
    if(x[i] >= 95) x[i] = 95;
  }
  return(x)
}

f33 <- function(x){
  for(i in 1:length(x)){
    if(x[i] < 33) x[i] = 0;
    if(x[i] >= 33) x[i] = 33;
  }
  return(x)
}

res.method <- as.data.frame(res.method[2:length(res.method[,1]),])
colnames(res.method) <- c("a","b","detG","distG1","norm2B")

ggdistG1 <-  ggplot(res.method,aes(a/100,b/100))+
  geom_raster(aes(fill=distG1))+
  scale_fill_gradient(low = "white", high = "black")+
  geom_contour(aes(z=f95(distG1)),color="green",bins=1)+
  geom_contour(aes(z=f33(distG1)),color="red",bins=1)+
  geom_text(aes(label = str_c("Mean = ", round(mean(res.method$distG1),2)), x = 1.30, y = 1.40),size=3,color="white")+
  labs(title="sfmB",
    fill="Accuracy",
    x="a",y="b")+
  scale_x_continuous(expand = c(0, 0))+
  scale_y_continuous(expand = c(0, 0))+
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
        panel.background = element_blank(), axis.line = element_line(colour = "black"))

ggdetG <-  ggplot(res.method,aes(a/100,b/100))+
  geom_raster(aes(fill=detG))+
  scale_fill_gradient(low = "white", high = "black")+
  geom_contour(aes(z=f95(detG)),color="green",bins=1)+
  geom_contour(aes(z=f33(detG)),color="red",bins=1)+
  geom_text(aes(label = str_c("Mean = ", round(mean(res.method$detG),2)), x = 1.30, y = 1.40),size=3,color="white")+
  labs(title="adG",
    fill="Accuracy",
    x="a",y="b")+
  scale_x_continuous(expand = c(0, 0))+
  scale_y_continuous(expand = c(0, 0))+
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
        panel.background = element_blank(), axis.line = element_line(colour = "black"))

ggnorm2B <- ggplot(res.method,aes(a/100,b/100))+
  geom_raster(aes(fill=norm2B))+
  scale_fill_gradient(low = "white", high = "black")+
  geom_contour(aes(z=f95(norm2B)),color="green",bins=1)+
  geom_contour(aes(z=f33(norm2B)),color="red",bins=1)+
  geom_text(aes(label = str_c("Mean = ", round(mean(res.method$norm2B),2)), x = 1.30, y = 1.40),size=3,color="white")+
  labs(title="n2B",
    fill="Accuracy",
    x="a",y="b")+
  scale_x_continuous(expand = c(0, 0))+
  scale_y_continuous(expand = c(0, 0))+
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
        panel.background = element_blank(), axis.line = element_line(colour = "black"))
```

7.2 Code: Bias of *detG*

```
library(tidyverse)
library(seqinr)
library(purrr)
library(stringr)
library(linpk)


## These functions are a less pruned version of QuartetTopology with a slightly different return
probtheo <- function(){
  k <- 1
  pt <- c()
  for(i1 in 1:4){
    for(i2 in 1:4){
      for(i3 in 1:4){
        for(i4 in 1:4){
          pt[k] <- str_c(i1,i2,i3,i4)
          k <- k + 1
        }
      }
    }
  }

  translateACGT <- function(s){
    s <- str_replace_all(s,"1","A") %>%
      str_replace_all("2","C") %>%
      str_replace_all("3","G") %>%
      str_replace_all("4","T")
    return(s)
  }
  rm(i1,i2,i3,i4)
  return(translateACGT(pt))
} #generates probability vector with CHARACTERS
pt <- probtheo()
wherequart <- function(q){
  q %>%
    str_to_upper() %>%
    #str_which(probtheo()) %>%
    str_which(pt) %>%
    return()
} #where is a ACGT (in chars) quartet in the vector

det_alchemy <- function(p){
  #theoretical probability vector (with characters) for auxiliary porpuses
  probtheo <- function(){
    k <- 1
    pt <- c()
    for(i1 in 1:4){
      for(i2 in 1:4){
        for(i3 in 1:4){
          for(i4 in 1:4){
            pt[k] <- str_c(i1,i2,i3,i4)
            k <- k + 1
          }
        }
      }
    }

    translateACGT <- function(s){
      s <- str_replace_all(s,"1","A") %>%
        str_replace_all("2","C") %>%
        str_replace_all("3","G") %>%
        str_replace_all("4","T")
      return(s)
    }
    rm(i1,i2,i3,i4)
    return(translateACGT(pt))
  }
  pt <- probtheo()
  #making the 3 flattenings
  a <- c(0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0)
  rn <- c("AA","AC","AG","AT","CA","CC","CG","CT","GA","GC","GG","GT","TA","TC","TG","TT")
  pf <- data.frame(rnames = rn,AA=a,AC=a,AG=a,AT=a,CA=a,CC=a,CG=a,CT=a,GA=a,GC=a,GG=a,GT=a,TA=a,TC=a,TG=a,TT=a)
  pf <- column_to_rownames(pf,"rnames")
  pf1234 <- pf;pf1324 <- pf;pf1423 <- pf;
  for(i in 1:16){ #here use pt instead of p (on the first one) to see the character flattenings
    pf1234[rn[i],] <- p[str_which(pt,str_c(rn[i],"[:upper:][:upper:]"))] #busca las coincidencias en el vector teÃ³rico
    pf1324[rn[i],] <- p[str_which(pt,str_c(substr(rn[i],1,1),"[:upper:]",substr(rn[i],2,2),"[:upper:]"))]
    pf1423[rn[i],] <- p[str_which(pt,str_c(substr(rn[i],1,1),"[:upper:]","[:upper:]",substr(rn[i],2,2)))]
  }
  rm(a,pf,i)
  #return(list(pf1234,pf1324,pf1423))
  #Flattenings in Fourier coordinates:
  H <- matrix(c(1,1,1,1,1,1,1,-1,-1,1,-1,1,-1,1,-1,1,-1,-1,1,-1,1,-1,-1,1,1,-1,-1,1),4,4)
  pf1234fou <- kronecker(solve(H),solve(H))%*%as.matrix(pf1234)%*%t(kronecker(solve(H),solve(H)))
  pf1324fou <- kronecker(solve(H),solve(H))%*%as.matrix(pf1324)%*%t(kronecker(solve(H),solve(H)))
  pf1423fou <- kronecker(solve(H),solve(H))%*%as.matrix(pf1423)%*%t(kronecker(solve(H),solve(H)))
  #Block A of the fourier flattening of p
  groupsum <- function(x){ #returns the sum (as a group) of the pair of elements in a row
    if(x == 1 | x == 6 | x == 11 | x == 16) return(1) #A
    if(x == 2 | x == 5 | x == 12 | x == 15) return(2) #C
```

```
  if(x == 3 | x == 8 | x == 9 | x == 14) return(3) #G
  if(x == 4 | x == 7 | x == 10 | x == 13) return(4) #T
}
bA12 <- c()
bA13 <- c()
bA14 <- c()
#construct the blocks, first making the vector then reorganizing with t(matrix(.))
for(i in 1:16){
  for(j in 1:16){
    si <- groupsum(i); sj <- groupsum(j);
    if(si == sj){
      if(si == 1){
        bA12 <- append(bA12,pf1234fou[i,j])
        bA13 <- append(bA13,pf1324fou[i,j])
        bA14 <- append(bA14,pf1423fou[i,j])
      }
    }
  }
}
rm(si,sj,i,j)

bA12 <- t(matrix(bA12,4,4))
bA13 <- t(matrix(bA13,4,4))
bA14 <- t(matrix(bA14,4,4))

#Constructing the G matrix for each tree
#The marginals
marg12 <- transmute(pf1234,marginal12=AA+AC+AG+AT+CA+CC+CG+CT+GA+GC+GG+GT+TA+TC+TG+TT) %>% #as vector
  t() %>% #to accurately convert
  matrix(4,4) #to access with margpos: marg12[[mp("AA")]]
#the following equality holds: marg12 == 0.25*tM[[1]]%*%tM[[2]]
marg34 <- transmute(as.data.frame(t(pf1234)),marginal34=AA+AC+AG+AT+CA+CC+CG+CT+GA+GC+GG+GT+TA+TC+TG+TT) %>% #t(), transposing to sum columns
  t() %>%
  matrix(4,4)
a1 <- c(1,2,3,4); c1 <- c(5,6,7,8); g1 <- c(9,10,11,12); t1 <- c(13,14,15,16)
a2 <- c(1,5,9,13); c2 <- c(2,6,10,14);g2 <- c(3,7,11,15); t2 <- c(4,8,12,16)
marg13 <- c(sum(pf1234[a1,a1]), sum(pf1234[a1,c1]), sum(pf1234[a1,g1]),sum(pf1234[a1,t1]), #A+
            sum(pf1234[c1,a1]), sum(pf1234[c1,c1]), sum(pf1234[c1,g1]),sum(pf1234[c1,t1]), #c+
            sum(pf1234[g1,a1]), sum(pf1234[g1,c1]), sum(pf1234[g1,g1]),sum(pf1234[g1,t1]), #g1+
            sum(pf1234[t1,a1]), sum(pf1234[t1,c1]), sum(pf1234[t1,g1]),sum(pf1234[t1,t1])) %>%  #T+
  t() %>%
  matrix(4,4)
marg24 <- c(sum(pf1234[a2,a2]), sum(pf1234[a2,c2]), sum(pf1234[a2,g2]),sum(pf1234[a2,t2]), #a2+
            sum(pf1234[c2,a2]), sum(pf1234[c2,c2]), sum(pf1234[c2,g2]),sum(pf1234[c2,t2]), #c2+
            sum(pf1234[g2,a2]), sum(pf1234[g2,c2]), sum(pf1234[g2,g2]),sum(pf1234[g2,t2]), #g2+
            sum(pf1234[t2,a2]), sum(pf1234[t2,c2]), sum(pf1234[t2,g2]),sum(pf1234[t2,t2])) %>%  #t2+
  t() %>%
  matrix(4,4)
#marg14
marg14 <- c(sum(pf1234[a1,a2]), sum(pf1234[a1,c2]), sum(pf1234[a1,g2]),sum(pf1234[a1,t2]), #A+
            sum(pf1234[c1,a2]), sum(pf1234[c1,c2]), sum(pf1234[c1,g2]),sum(pf1234[c1,t2]), #C+
            sum(pf1234[g1,a2]), sum(pf1234[g1,c2]), sum(pf1234[g1,g2]),sum(pf1234[g1,t2]), #G+
            sum(pf1234[t1,a2]), sum(pf1234[t1,c2]), sum(pf1234[t1,g2]),sum(pf1234[t1,t2])) %>%  #T+
  t() %>%
  matrix(4,4)
marg23 <- c(sum(pf1234[a2,a1]), sum(pf1234[a2,c1]), sum(pf1234[a2,g1]),sum(pf1234[a2,t1]), #a2+
            sum(pf1234[c2,a1]), sum(pf1234[c2,c1]), sum(pf1234[c2,g1]),sum(pf1234[c2,t1]), #c2+
            sum(pf1234[g2,a1]), sum(pf1234[g2,c1]), sum(pf1234[g2,g1]),sum(pf1234[g2,t1]), #g2+
            sum(pf1234[t2,a1]), sum(pf1234[t2,c1]), sum(pf1234[t2,g1]),sum(pf1234[t2,t1])) %>%  #t2+
  t() %>%
  matrix(4,4)
rm(a1,c1,g1,t1,a2,c2,g2,t2)

#Fourier marginals
foumarg12 <- solve(H) %*% marg12 %*% solve(H) #should be a diagonal matrix
foumarg34 <- solve(H) %*% marg34 %*% solve(H)
foumarg13 <- solve(H) %*% marg13 %*% solve(H)
foumarg24 <- solve(H) %*% marg24 %*% solve(H)
foumarg14 <- solve(H) %*% marg14 %*% solve(H)
foumarg23 <- solve(H) %*% marg23 %*% solve(H)

mp <- function(XY){  #margpos
  c("AA","AC","AG","AT","CA","CC","CG","CT","GA","GC","GG","GT","TA","TC","TG","TT") %>%
    str_which(XY) %>%
    return()
}
pm <- function(x){
  c("AA","AC","AG","AT","CA","CC","CG","CT","GA","GC","GG","GT","TA","TC","TG","TT")[x] %>%
    return()
}

#G12|34
G12rightA <- 16*blockdiag(1/foumarg34[[mp("AA")]],1/foumarg34[[mp("CC")]],1/foumarg34[[mp("GG")]],1/foumarg34[[mp("TT")]])
G12leftA <- foumarg13[[mp("AA")]]*blockdiag(1/foumarg12[[mp("AA")]],1/foumarg12[[mp("CC")]],1/foumarg12[[mp("GG")]],1/foumarg12[[mp("TT")]])
G12 <- G12leftA %*% bA12 %*% G12rightA
#G13|24
G13rightA <- 16*blockdiag(1/foumarg24[[mp("AA")]],1/foumarg24[[mp("CC")]],1/foumarg24[[mp("GG")]],1/foumarg24[[mp("TT")]])
G13leftA <- foumarg12[[mp("AA")]]*blockdiag(1/foumarg13[[mp("AA")]],1/foumarg13[[mp("CC")]],1/foumarg13[[mp("GG")]],1/foumarg13[[mp("TT")]])
G13 <- G13leftA %*% bA13 %*% G13rightA
#G14|23
G14rightA <- 16*blockdiag(1/foumarg23[[mp("AA")]],1/foumarg23[[mp("CC")]],1/foumarg23[[mp("GG")]],1/foumarg23[[mp("TT")]])
G14leftA <- foumarg12[[mp("AA")]]*blockdiag(1/foumarg14[[mp("AA")]],1/foumarg14[[mp("CC")]],1/foumarg14[[mp("GG")]],1/foumarg14[[mp("TT")]])
```

```
  G14 <- G14leftA %*% bA14 %*% G14rightA

  G <- list(G12,G13,G14)

  dets <- map(list(G[[1]],G[[2]],G[[3]]),det) %>%
    unlist() #%>%
    #abs()
  return(dets) #returns the 3 determinants

}

#Matrices to compute expected value for the 2 cases

Me <- matrix(c(0.482794071309854, 0.140081335066641, 0.292151277919313, 0.0849733157041928,
               0.140081335066641, 0.482794071309854, 0.0849733157041928, 0.292151277919313,
               0.292151277919313, 0.0849733157041928, 0.482794071309854, 0.140081335066641,
               0.0849733157041928, 0.292151277919313, 0.140081335066641, 0.482794071309854),4,4)


# Me <- matrix(c(0.513802714081176, 0.0689575677123846, 0.0609441668572961, 0.356295551349143,
# 0.0689575677123846, 0.513802714081176, 0.356295551349143, 0.0609441668572961,
# 0.0609441668572961, 0.356295551349143, 0.513802714081176, 0.0689575677123846,
# 0.356295551349143, 0.0609441668572961, 0.0689575677123846, 0.513802714081176),4,4)

detG13 <- (4^4*Me[1]*Me[2]*Me[3]*Me[4])/(det(Me)^2)

##Process

setwd("") #Location of fasta file
align <- read.fasta(file="") #Name of fasta file

kdets <- matrix(0L,ncol=4,nrow=1)
freqk<-rep(0L,256)
for(k in 1:10000){
  alignk <- list(align[[1]][1:k*100],align[[2]][1:k*100],align[[3]][1:k*100],align[[4]][1:k*100])
  for(j in (100*(k-1)+1):(100*k)){ #make probability vector for each alignment (length 1000)
            idx <- str_c(align[[1]][j],align[[2]][j],align[[3]][j],align[[4]][j]) %>%
              wherequart()
            freqk[idx] <- freqk[idx] + 1 #add probability
          }
          freqk2 <- freqk/(100*k)
          detsk <- det_alchemy(freqk2)
          kdets <- rbind(kdets,cbind(c(which.min(detsk),which.min(detsk),which.min(detsk)),detsk,c(k,k,k),c(12,13,14)))
          print(k)
}

kdets <- as.data.frame(kdets[2:length(kdets[,1]),])
colnames(kdets) <- c("sol","det","k","flatt")

#Changing parameters different plots are obtained
ggdet <- ggplot(filter(kdets,abs(det)<10^4))+
  geom_point(aes(x=k*100,y=det,color=factor(flatt),shape=factor(sol)))+
  geom_line(aes(x=k*100,y=detG13),color="black",size=0.5)+
  labs(title= "Evolution of detG (|detG| < 10^4)",
       x="length of alignment",
       y="value of det",
       color="G matrix",
       shape="Tree yielded by method")
```