

PART C MS2A BIOINFORMATICS AND COMPUTATIONAL BIOLOGY

LECTURES 2-8, GEOFF NICHOLLS

1. SEQUENCES ON TREES

We will refer in this section to supplementary material in [2] (Chapter 7, 7.1-7.2, 7.3-7.4) [5] (Chapters 1 and 2).

Data: aligned sequences of characters homologous by column; determined by transformation from a common ancestor.

Example 1.1. : trait data

	hair	wings	lactation
bat	1	1	1
human	1	0	1
bird	0	1	0
crocodile	0	0	0

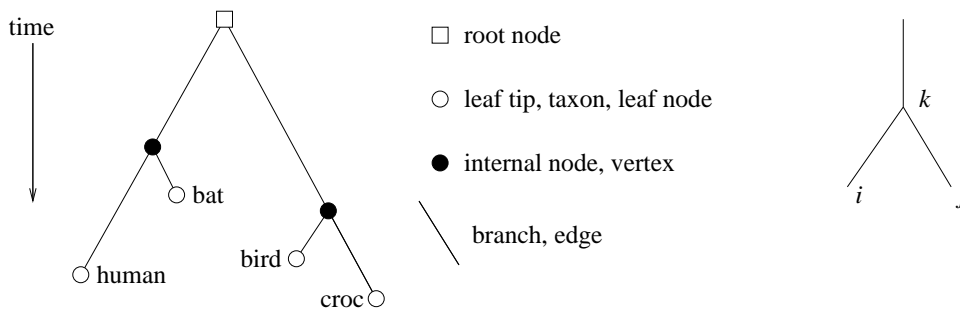
$n = 4$ individuals/taxa, $N = 3$ sites/traits. Characters $\mathcal{C} = \{0, 1\}$. $x_{i,j}$ = character at site/trait j in row/taxon i so $x_{i,j} \in \mathcal{C}$.

Example 1.2. : aligned mtDNA sequence data $n = 4$ taxa, $N = 5$ sites. Characters

Pan	TTATCC
Gorilla	TTGTTC
Pongo	CCACCC
Hylobates	CCGTCC

$\mathcal{C} = \{A, C, G, T, -\}$ with $-$ denoting a possible gap character.

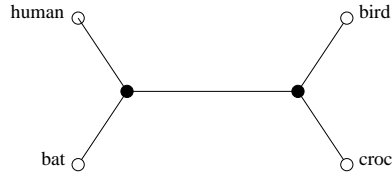
Trees: rooted



Key words and phrases. phylogenetics, parsimony, likelihood, Markov chain Monte Carlo.

We use a family metaphor to refer to relative positions on rooted trees. Node k is the parent of nodes i and j . Nodes i and j are the children of node k etc.

Trees: unrooted



In an unrooted tree the edge containing the common ancestor is not identified. The direction of time is not indicated.

Exercise 1.3. : show that

$$\# \text{ of unrooted trees with } n \text{ tips} = \prod_{i=3}^n (2i - 5),$$

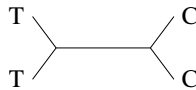
and that

$$\# \text{ rooted} = \# \text{ unrooted} \times (2n - 3).$$

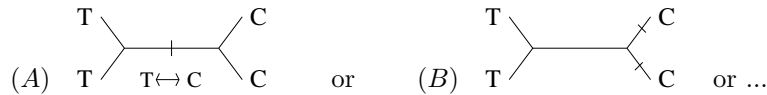
Phylogenetic analysis: assume sequences evolved on a common tree. Find the tree topology (and branch lengths, counted in substitutions if necessary, and calendar units if possible). We will use the symbol τ to represent a tree. In order to specify a tree we should give the names of nodes $i \in V$, the edges (as pairs of named nodes, $\langle i, j \rangle \in E$) and the lengths $t_{i,j}$ of the edges (indicating the amount of time, or ‘evolutionary time’, which we will define shortly, which elapses along an edge).

Transformation histories on trees:

referring to column 1 of Example 1.2 the leaf values



could arise *via*



We would like to reconstruct the history of the character transformations (which are base substitutions if $\mathcal{C} = \{A, C, G, T, -\}$ as in the example above) that give rise to the data. Notice that a transformation history specifies the character states at internal nodes of the graph. These values correspond to the trait sequences, or DNA base sequences of ancestral individuals, and may be of independent interest.

Evolutionary time is often measured in transformations - this allows us to give trees without the need to make assumptions about the rate at which transformations accumulate. If we know the number of character transformations along the edges of a tree, we draw the tree so that edge lengths are proportional to the number of transformations they carry. We sometimes assume that the amount of calendar time which elapsed along an edge is proportional to the number of character transformations which have accumulated along that edge. When this assumption can be made we can convert between time measured in transformations and time measured in calendar units and estimate the time back to a common-ancestor/speciation event.

2. PARSIMONY

Which tree best explains the data? Parsimony is a criterion for choosing the 'best' tree.

Fitch parsimony: All transitions between all characters are allowed, with equal weight. In this case history (A) above has length equal one while history (B) above has length two. In Fitch parsimony the evolution-length of an edge is the # of transformations occurring on it. For a given history the tree length is the sum of the lengths of its edges. For a given tree we favor the history which explains the data with the smallest number of transformations. We favor history (A) over history (B). Fitch parsimony allows all transformations of characters with equal weight.

Weighted parsimony: Weighted parsimony is an obvious generalization of Fitch parsimony. For $i, j \in \mathcal{C}$ count c_{ij} for a transition from i to j .

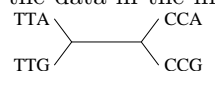
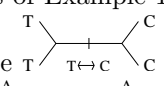
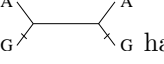
Example 2.1. The nucleotide bases A (adenine) and G (guanine) are purines. C (cytosine) and T (thymine) are pyrimidines. Substitutions $A \leftrightarrow G$ and $C \leftrightarrow T$ leaving the base type unchanged are called transitions. Substitutions ($A \leftrightarrow T$ etc) which replace a purine with a pyrimidine and *vis versa* are called transversions. For chemical reasons transitions occur more easily than transversions, so we more readily allow transitions in our transformation histories than transversions. The following weight matrix $c = [c_{ij}]_{i,j \in \mathcal{C}}$ imposes a penalty ratio of around 2, which is typical:

$$c = \begin{bmatrix} 0 & 2 & 1 & 2 \\ 2 & 0 & 2 & 1 \\ 1 & 2 & 0 & 2 \\ 2 & 1 & 2 & 0 \end{bmatrix}$$

Exercise 2.2. show that Fitch parsimony is a special case of weighted parsimony.

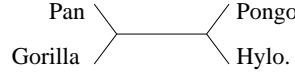
Variants of parsimony: Certain other types of parsimony rule out certain types of transformations - see the discussion of parsimony in Chapter 7 of [5].

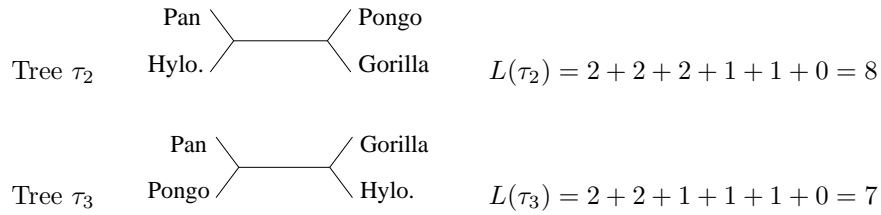
Define the **length of a tree** τ to be the smallest number of transformations explaining the data on that tree, and let $L(\tau)$ stand for that length.

Example 2.3. for the data in the first three columns of Example 1.2 the length of the tree $\tau =$  is $L(\tau) = 4$ since  has length one, but appears twice (at the first and second site) and  has length two, it follows that $L(\tau) = 1 + 1 + 2$.

The maximum parsimony tree: that tree minimizing $L(\tau)$ for given data. Note however that there are many different types of parsimony - for example

Example 2.4. Apply the Fitch parsimony criterion to decide the Fitch-optimal tree for *Pan*, *Pongo*, *Gorilla* and *Hylobates* in the mtDNA Example 1.2.

Tree τ_1  $L(\tau_1) = 1 + 1 + 2 + 1 + 1 + 0 = 6$



and the winner is ... tree τ_1 !

Rooting trees:

Sometimes we are unable to determine where on the tree the root node is located, even though we can determine the topology of the unrooted tree.

Exercise 2.5. show that, under Fitch parsimony, $L(\tau)$ is independent of the root position.

Exercise 2.6. show that, under weighted parsimony, $L(\tau)$ is independent of the root position if, for $i, j, k \in \mathcal{C}$, c_{ij} is a metric (ie, $c_{ii} = 0$, $c_{ij} = c_{ji}$ and $c_{jk} \leq c_{ji} + c_{ik}$). [show all time orientations of an edge, with maximum parsimony states j and k at its ends, $j \rightarrow k$, $k \rightarrow j$ and $j \leftarrow$ (root state) $\rightarrow k$, contribute equal total weight to $L(\tau)$]

If, under our parsimony assumptions, the data is uninformative of the position of the root, there are a couple of strategies for rooting the tree. One is to supply an outgroup - a data sequence from a taxon known to lie outside the clade formed by the rest of the taxa in the data. A second possibility is to assume that the rate of substitution was equal at all times in all species in the phylogeny - the amount of time from the root to the leaf tips is the same, so the accumulated number of substitutions should be the same too. We would expect this to be true only on-average: substitution is a somewhat random process; the number of substitutions down two branches of equal length in calendar time need not be equal.

Ancestral character states:

Notice that the parsimony criterion can be applied to determine both the phylogeny and ancestral character states from sequence data. As we will see, the minimum length character transformation history on the maximum parsimony tree assigns these ancestral character states.

3. SOME CRITICISMS OF THE PARSIMONY CRITERION

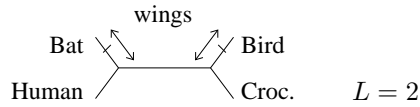
Problems with parsimony as an inference scheme.

Uncertainty:

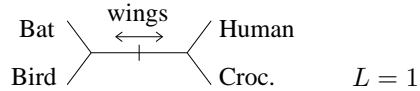
It is unsatisfactory to give a single tree as the answer, without giving some measure of the confidence we have that this tree is correct, or some sense of which other trees might plausibly explain the data. Statistical bootstrap methods (see [2] Chapter 7.5 and [5] chapter 20) are commonly used to quantify the strength of support for different tree clades. In this scheme parsimony trees are reconstructed from many randomly chosen subsamples of the data. We check that the tree-features of interest are reproduced in these new trees.

Homoplasy:

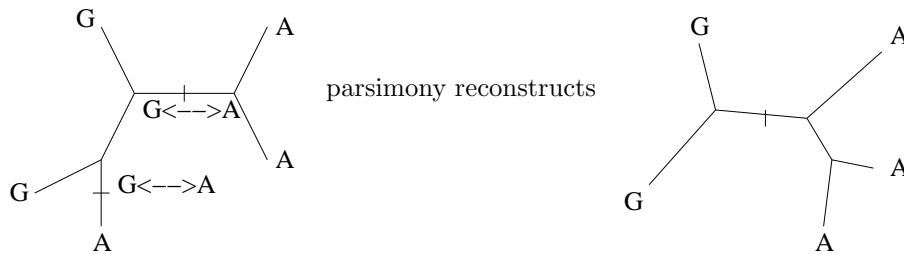
we may get the wrong tree. For example in



the trait “wings” evolved independently on two lineages. However, parsimony favors



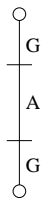
This is an instance of **parallel evolution**. We get problems of a similar kind from **back-mutation** or **reversion to ancestral type**. When the truth has length $L = 2$ due to a reversal,



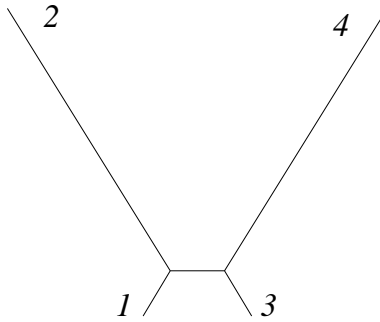
since the tree on the right has length $L = 1$.

Incorrect edge lengths:

It follows from the above discussion that we cannot rely on parsimony to give us the correct edge lengths (even if we have the correct tree topology). A homoplasy (reversion or independent innovation depends where the root is) like this

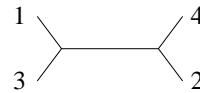


would be reconstructed with length zero when the correct length is two. This kind of error is important if we wish to convert transformations to calendar units, and estimate the length of an edge in years.

Long Branch Attraction:

Suppose the true tree is as above *ie* $((1, 2), (3, 4))$. Notice that the branches leading to nodes 2 and 4 are drawn long to indicate many more character transformations along these branches than there are along the branches leading to nodes 1 and 3. This would happen if for example the substitution rates were higher on those branches, or more time had elapsed along those branches. Suppose the data is binary so $\mathcal{C} = \{0, 1\}$ and we have four sequences of length N . At most sites $j = 1, 2, \dots, N$ in the data $x_{1,j} = x_{3,j}$ since we have set this problem up so that there are few character transformations on the edges connecting nodes 1 and 3. Suppose (without loss of generality) at site j we have $x_{1,j} = 1$ and $x_{3,j} = 1$. What are the possibilities at nodes 2 and 4?

If $x_{2,j} = 1$ and $x_{4,j} = 1$ then all trees have equal length for column j of the data. If $x_{2,j} = 1$ and $x_{4,j} = 0$ then again all trees have equal length for column j of the data (draw the trees and check this). The same holds if $x_{2,j} = 0$ and $x_{4,j} = 1$. So these data are not informative.



However, if $x_{2,j} = 0$ and $x_{4,j} = 0$ then column j does favor a tree: 3 which is the wrong one. This data groups $((2, 4), (1, 3))$. Increasing sequence length wont help. There is something wrong with parsimony as an inference scheme.

We have over-simplified - we need to be more specific about how changes took place on the true tree. The substitution process has some parameters, rate parameters, controlling the relative rates for different substitutions ($A \leftrightarrow G, A \leftrightarrow C$ *etc*). Likewise the tree branches have true lengths. Depending on the values of these parameters the problem we have described need not arise. When the combined tree and rate parameters are in the range that can cause this failure of parsimony, we say the data is in the “Felsenstein zone”, since this problem was identified in [3].

The problem is absent in Likelihood methods if the substitution model is reliable.

4. PARSIMONY ALGORITHMS

Consider the problem of finding the maximum parsimony phylogeny for n sequences of length N . The straightforward approach is to enumerate all possible trees, compute the length of each tree, and return the tree having the shortest length.

Computing weighted parsimony scores: For each tree we need to find the minimum length transformation history for each column of the data $x_{a,\alpha}$, $a =$

$1, 2, \dots, n$, $\alpha = 1, 2, \dots, N$. A dynamic programming algorithm, due to Sankoff, computes this in a time proportional to $n|\mathcal{C}|$ for each column of the data.

Consider node $a \in V$ of a rooted tree τ with root node label $r \in V$. For $i \in \mathcal{C}$, Denote by $S_a(i)$ the length of the minimum length transformation history on the tree descended from node a , if there is character i at node a . Once we know $S_r(i)$ for $i \in \mathcal{C}$, we are done, since $L(\tau) = \min_{i \in \mathcal{C}} S_r(i)$. We will compute $S_a(i)$ up from the leaves. First if a is a leaf with character $x_{a,\alpha}$ then

$$S_a(i) = \begin{cases} 0 & \text{if } i = x_{a,\alpha} \\ \infty & \text{if } i \neq x_{a,\alpha} \end{cases}$$

Now, let $b \in V$ and $c \in V$ be two child nodes of node a . To compute $S_a(i)$ we have

$$S_a(i) = \min_{j \in \mathcal{C}} (c_{ij} + S_b(j)) + \min_{k \in \mathcal{C}} (c_{ik} + S_c(k)).$$

and we must compute this for each character $i \in \mathcal{C}$. The cost of a transformation history on the tree descended from a is given in terms of the cost of the transformations on the edges from a to b and c plus the cost of any transformations in the trees descended from b and c . If $S_a(i)$ was not minimal then one of the terms on the right cannot be minimal and this leads to a contradiction, so $S_a(i)$ is minimal. We iterate this rule up the tree to obtain $S_r(i)$.

The algorithm can be modified to find a set of ancestral character states $i_a, a \in V$, which achieve the minimal length transformation history. If we record for each edge (a, b) the j -value at the child b that achieves the minimum length for each i -value at the parent a ,

$$J_{a,b}(i) = \arg \min_{j \in \mathcal{C}} (c_{ij} + S_b(j))$$

then we can set $i_r = \arg \min_{i \in \mathcal{C}} S_r(i)$ and descend, setting $i_b = J_{a,b}(i_a)$ along each edge.

Exercise 4.1. take tree τ_1 in Section 2, together with column 3 of its sequence data $x_{a,3}$. Root the tree at the central edge, and compute $S_a(i)$ for each node a and each character state i , using the algorithm above, under weighted parsimony with the weight matrix c of Example 2.1. Compute the ancestral node labels.

Exercise 4.2. Sankoff's weighted parsimony algorithm and the Viterbi algorithm, which you may encounter in lectures 9-15, are dynamic programming algorithms. How are these two algorithms related? (try googling *viterbi algorithm* if you are unfamiliar with this algorithm).

Searching over trees: Having computed $L(\tau)$ we must search over trees for that tree having the shortest length. It has been shown that this problem is NP-hard: we expect that the worst case asymptotic run-time complexity of any algorithm solving this problem is not polynomial but exponential.

Branch and bound is a natural search scheme for this problem. See chapter 5 of [5] for a full explanation of branch and bound computation for phylogenetics. Our search space is the set of all tree topologies. Some tree τ_0 is chosen to start the search, and its length L_0 computed. This is our current best tree. We now start with two taxa, and build a new tree adding taxa one at a time, computing the minimum transformation length L' of the sub-tree as it is built up. If at any stage $L' > L_0$, the length of the sub-tree exceeds the length of our current best tree, we remove from the search space all trees that contain that subtree. If we build a full

tree τ with length L smaller than L_0 then we set $\tau_0 = \tau$ and $L_0 = L$. We continue till the search space is exhausted.

5. LIKELIHOOD METHODS I: NEUTRAL INDEPENDENT FINITE SITES MODEL

We refer in this section to supplementary material in [2] (Chapter 8) and [5] (Chapter 13).

Likelihood methods for deciding the phylogeny of sequence data are explicitly **model based**.

- We model the mutation process *via* a stochastic model.
- We favor trees which make the data a likely outcome of the model substitution process.

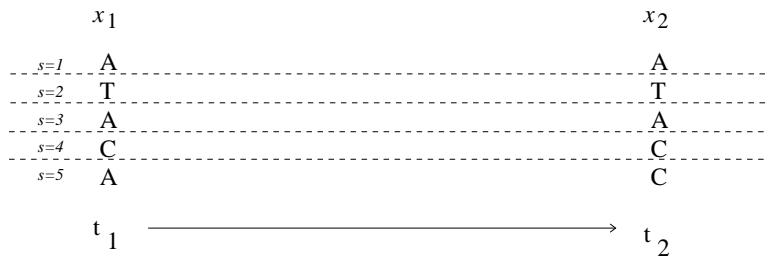
The **neutral independent finite-sites model** was proposed by Felsenstein in [4]. It seems to be a suitable model for the base substitution process in nucleotide-base sequence data. It can be used to model mutational processes acting on protein sequence data also.

- **neutral** meaning we ignore the effects of selection
- **independent** we suppose substitution events act independently at each site - so we are ignoring the effects of molecular structure. The model can be made more sophisticated - for example the reading-frame effect associated with codon/protein structure is handled by most modern likelihood based phylogeny software packages.
- **finite sites** allow multiple substitutions at a site.

The order of the sites in the sequence is unimportant. We are ignoring recombination, and any intra-site variation or correlation of the substitution process between sites.

5.1. Substitution transition probabilities.

Let x_1 and x_2 be two aligned sequences with N sites each, $s = 1, 2, \dots, N$ eg $x_1 = \text{ATACA}$ and $x_2 = \text{ATACC}$ so $x_{1,4} = \text{C}$ etc. Now switch to coding $\text{A} = 1, \text{C} = 2, \text{G} = 3, \text{T} = 4$ so $x_1 = 14121$ and $x_2 = 14122$ and we will write $x_{1,4} = 2$ or $x_{1,4} = \text{C}$ as convenient.



Suppose it is known that sequence x_1 evolved to become sequence x_2 over some time $t = t_2 - t_1$. For $i, j \in \mathcal{C}$, let

$$P_{i,j}(t) = \Pr(x_{2,s} = j | x_{1,s} = i),$$

that is, $P_{i,j}(t)$ is the probability to find a base $j \in \{A, C, G, T\}$ at site s in x_2 given there was a base $i \in \{A, C, G, T\}$ at that site in x_1 .

We assume the substitution process we observe is Markov. If we are given the sequence y of a creature ancestral to x_2 and descended from x_1 , which existed at some time after t_1 and before t_2 , then

$$\Pr(x_{2,s} = j | x_{1,s} = i, y_s = k) = \Pr(x_{2,s} = j | y_s = k).$$

The probability to see any particular child sequence is determined once we know the parent sequence. If we know in addition the sequence of the grandparent, this tells us nothing new.

Assume $P_{i,j}(t)$ does not depend on the site s . The outcome $x_{2,s} = j$ is equally likely at each site $s = 1, 2 \dots N$ where $x_{1,s} = i$.

Exercise 5.1. Explain why selection acting on proteins might lead to rate heterogeneity between sites in a substitution process, like the one above which acts on DNA bases, (think about the translation of mRNA to protein, and the codon reading-frame).

$P = [P_{i,j}]$, $i, j = 1, 2, 3, 4$ is a 4×4 **transition probability matrix**. Now, $\Pr(x_{2,s} = 1 \text{ or } 2 \text{ or } 3 \text{ or } 4 | x_{1,s} = i) = 1$ since the outcome is certain to occur, so

$$\sum_{j=1}^4 P_{i,j} = 1$$

Example 5.2. [1] analyze $n = 60$ aligned HIV-I *env* sequences with $N = 660$ sites. Imagine tracking the base at some fixed *env*-site as it evolves for around 16667 days (about 46 years) down an HIV-I lineage (why did I choose 16667 days? we will come back to that). For that gene in that organism, in the model used by [1], the transition probability for base changes at a site over $t = 16667$ days is

$$P(t) = \begin{pmatrix} & \text{A} & \text{C} & \text{G} & \text{T} \\ \text{A} & 0.5583 & 0.1154 & 0.2231 & 0.1031 \\ \text{C} & 0.2867 & 0.3663 & 0.1102 & 0.2368 \\ \text{G} & 0.4230 & 0.0841 & 0.3813 & 0.1116 \\ \text{T} & 0.1810 & 0.1672 & 0.1032 & 0.5485 \end{pmatrix}$$

Notice that $\sum_{j=1}^4 P_{i,j} = 1$, rows sum to one. Also, transitions, $A \leftrightarrow G$ and $C \leftrightarrow T$, are somewhat more probable than transversions, except that transversions to A seem somewhat favored. Where did the above matrix come from? We will discuss this below.

Example 5.3. The Jukes-Cantor model (1969) of base substitutions is the simplest model one can construct. The transition matrix $P(t)$ is

$$\begin{pmatrix} P_{1,1} & P_{1,2} & P_{1,3} & P_{1,4} \\ P_{2,1} & P_{2,2} & P_{2,3} & P_{2,4} \\ P_{3,1} & P_{3,2} & P_{3,3} & P_{3,4} \\ P_{4,1} & P_{4,2} & P_{4,3} & P_{4,4} \end{pmatrix} = \begin{pmatrix} \frac{1+3e^{-4t/3}}{4} & \frac{1-e^{-4t/3}}{4} & \frac{1-e^{-4t/3}}{4} & \frac{1-e^{-4t/3}}{4} \\ \frac{1-e^{-4t/3}}{4} & \frac{1+3e^{-4t/3}}{4} & \frac{1-e^{-4t/3}}{4} & \frac{1-e^{-4t/3}}{4} \\ \frac{1-e^{-4t/3}}{4} & \frac{1-e^{-4t/3}}{4} & \frac{1+3e^{-4t/3}}{4} & \frac{1-e^{-4t/3}}{4} \\ \frac{1-e^{-4t/3}}{4} & \frac{1-e^{-4t/3}}{4} & \frac{1-e^{-4t/3}}{4} & \frac{1+3e^{-4t/3}}{4} \end{pmatrix}$$

ie,

$$P_{i,j} = \begin{cases} \frac{1}{4} + \frac{3}{4}e^{-4t/3} & \text{if bases } i \text{ and } j \text{ are the same (no change), and} \\ \frac{1}{4} - \frac{1}{4}e^{-4t/3} & \text{if bases } i \text{ and } j \text{ differ,} \end{cases}$$

(so one or more substitutions has occurred)

Notice that $\sum_{j=1}^4 P_{i,j} = 1$.

5.2. Equilibrium base frequencies.

The **equilibrium base frequencies** $\pi = (\pi_A, \pi_C, \pi_G, \pi_T)$ (or equivalently $\pi = (\pi_1, \pi_2, \pi_3, \pi_4)$) give the probability to find any particular base at a site. π_i gives the probability $\Pr(x_{2,s} = i)$ to find base $i \in \{\mathbf{A}, \mathbf{C}, \mathbf{G}, \mathbf{T}\}$ at site s in sequence x_2 if no information about the history of sequence x_2 is available. We assume that these proportions of bases don't change over time (in a long sequence). They are the same at all times, *ie*,

$$\pi_i = \Pr(x_{2,s} = i) = \Pr(x_{1,s} = i) \quad \text{for each } i = 1, 2, 3, 4, \text{ also.}$$

The equilibrium base frequencies $\pi = (\pi_A, \pi_C, \pi_G, \pi_T)$ can also be thought of as the proportion of A's, C's, G's and T's in a long sequence. Consider the sequence x_2 . It has length N sites. If $i \in \{\mathbf{A}, \mathbf{C}, \mathbf{G}, \mathbf{T}\}$ and $N_i^{(2)}$ is the number of i -bases in sequence x_2 then $\pi_i = \mathbf{E}(N_i^{(2)}/N)$. Because we are assuming these proportions don't change, $\pi_i = \mathbf{E}(N_i^{(1)}/N)$, also.

We can use the **empirical base frequencies** $\hat{\pi}_i = N_i/N$ in the data to estimate π_i , the equilibrium base frequencies, if N is large so that N_i/N is a reliable estimate for π_i .

Exercise 5.4. show that the random vector $(N_i, i \in \mathcal{C})$ of base frequencies in a sequence of length N has a multinomial distribution [and if you have done enough statistics, show $\hat{\pi}_i = N_i/N$ is the maximum-likelihood estimator for π_i and $\sqrt{\hat{\pi}_i(1-\hat{\pi}_i)/N}$ is an estimate for the standard error of $\hat{\pi}_i$].

Example 5.5. For the HIV-I *env* sequence substitution model of [1], we might estimate the equilibrium base frequencies empirically by simply counting the proportions of each base in the 60×660 sequence alignment. We estimate

$$(\hat{\pi}_A, \hat{\pi}_C, \hat{\pi}_G, \hat{\pi}_T) = (0.4, 0.161, 0.211, 0.228)$$

How are the proportions $N^{(1)}/N$ of bases in x_1 related to those in x_2 ?

$$\frac{N_j^{(2)}}{N} = \sum_{i=1}^4 \frac{N_i^{(1)}}{N} \times \text{the proportion of } i\text{'s that become } j\text{'s.}$$

The proportion of j 's in x_2 is the proportion of A's in x_1 times the proportion of A's that become j 's plus the proportion of C's in x_1 times the proportion of C's that become j 's and so on. But the proportion of i 's that become j 's over the time interval t is just $P_{i,j}(t)$ so the equation above says

$$\pi_j = \sum_{i=1}^4 \pi_i P_{i,j}$$

or $\pi = \pi P$ in matrix notation. In order for the base frequencies to be constant in time we need them to be related to the base transition probabilities by $\pi = \pi P$.

Exercise 5.6. check that $\pi = \pi P$ for the transition matrix P in Example 5.2 and the equilibrium base frequencies of Example 5.5.

The difference between $P_{i,j}(t) = \Pr(x_{2,s} = j | x_{1,s} = i)$ and $\pi_j = \Pr(x_{2,s} = j)$ is that $P_{i,j}(t)$ contains the extra information that $x_{1,s} = i$ at time t_1 . But if time t_1

is a time in the very far distant past (so t is large), knowing $x_{1,s} = i$ tells us little about the likely value of $x_{2,s}$ - the base at site s might be substituted many times. So we expect $P_{i,j}(t) \rightarrow \pi_j$ as $t \rightarrow \infty$. The rows of $P(\infty)$ are equal to π .

Example 5.7. For the HIV-I *env* sequence substitution model of [1], here is the transition probability at a site taken over a very long time, $t \simeq \infty$,

$$P(\infty) = \begin{pmatrix} 0.4 & 0.161 & 0.211 & 0.228 \\ 0.4 & 0.161 & 0.211 & 0.228 \\ 0.4 & 0.161 & 0.211 & 0.228 \\ 0.4 & 0.161 & 0.211 & 0.228 \end{pmatrix}$$

The probability to find any particular base at a site doesn't depend on what the base was at that site in the far distant past. The probability, after a long time, to find a **C** at a site is 0.161 irrespective of whether the base at that site was **A**, **C**, **G** or **T** at some very large time t ago. That is, $P_{i,2}(\infty)$ doesn't depend on i .

Exercise 5.8. check that $\pi = \pi P$ and $P = PP$ for the transition matrix $P(\infty)$ in Example 5.7 and the equilibrium base frequencies of Example 5.5, and interpret the result $P = PP$.

5.3. Conditions on substitution probabilities from the biology. Before we make a model that determines $P(t)$ let us summarize some common-sense properties $P(t)$ must have.

- (1) The probability for there to be a substitution goes to zero as the time in which the substitutions can occur goes to zero. When $t = 0$ (no time between x_1 and x_2 , $P_{i,j}(0) = 0$ when $i \neq j$ and $P_{i,j}(0) = 1$ when $i = j$.)
- (2) When $t \rightarrow \infty$, knowing $x_{1,s} = i$ tells us nothing about the value of $x_{2,s}$ so

$$\Pr(x_{2,s} = j | x_{1,s} = i) \rightarrow \Pr(x_{2,s} = j) \quad \text{as } t \rightarrow \infty,$$

or in other words

$$P_{i,j}(t) \rightarrow \pi_j \quad \text{as } t \rightarrow \infty.$$

Example 5.9. The Jukes-Cantor model of Example 5.3, has at least these basic properties. When $t \rightarrow 0$ we have $P \rightarrow \mathbb{I}_4$, that is

$$\begin{pmatrix} P_{1,1} & P_{1,2} & P_{1,3} & P_{1,4} \\ P_{2,1} & P_{2,2} & P_{2,3} & P_{2,4} \\ P_{3,1} & P_{3,2} & P_{3,3} & P_{3,4} \\ P_{4,1} & P_{4,2} & P_{4,3} & P_{4,4} \end{pmatrix} \longrightarrow \mathbb{I}_4 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \text{as } t \rightarrow \infty$$

The probability for there to be a substitution goes to zero as the time in which substitutions can occur goes to zero. On the other hand, as $t \rightarrow \infty$, P becomes just a 4×4 matrix of $1/4$'s:

$$\begin{pmatrix} P_{1,1} & P_{1,2} & P_{1,3} & P_{1,4} \\ P_{2,1} & P_{2,2} & P_{2,3} & P_{2,4} \\ P_{3,1} & P_{3,2} & P_{3,3} & P_{3,4} \\ P_{4,1} & P_{4,2} & P_{4,3} & P_{4,4} \end{pmatrix} \longrightarrow \begin{pmatrix} 1/4 & 1/4 & 1/4 & 1/4 \\ 1/4 & 1/4 & 1/4 & 1/4 \\ 1/4 & 1/4 & 1/4 & 1/4 \\ 1/4 & 1/4 & 1/4 & 1/4 \end{pmatrix} \quad \text{as } t \rightarrow \infty$$

In this model a sequence that has a base i at site s is equally likely to have an **A**, **C**, **G** or **T** at site s after a very long time. For this model the equilibrium base frequencies (discussed below) are $\pi = (1/4, 1/4, 1/4, 1/4)$ and indeed the rows of $P(\infty)$ correspond to the equilibrium base frequencies, that is $P_{i,j} \rightarrow \pi_j$ as $t \rightarrow \infty$.

Exercise 5.10. Check $\pi = \pi P$ for the Jukes Cantor model. ♠

5.4. Markov mutation model.

Rate matrices: it is convenient to parameterise the model *via* a 4×4 rate matrix $Q = [Q_{i,j}]$, $i, j = 1, 2, 3, 4$.

$Q_{i,j}$ = the instantaneous rate for the substitution $i \rightarrow j$ at each site.

$Q_{i,j}$ has units “substitutions per unit time” [subs/time]. We treat a process acting at a single site, so we could add “per site”. It is standard practice to normalize Q so that there is one substitution per unit time *ie* for time measured in substitutions.

Example 5.11. For the HIV-I *env* sequence substitution model of [1], the rate matrix (normalized to give one substitution per unit time) is

$$Q = \begin{bmatrix} -0.91065 & 0.23919 & 0.5637 & 0.10777 \\ 0.59423 & -1.2136 & 0.095945 & 0.52343 \\ 1.0687 & 0.073194 & -1.3125 & 0.17064 \\ 0.18912 & 0.36964 & 0.15791 & -0.71667 \end{bmatrix}$$

What are the diagonal elements and why are they negative? How do we know this is “normalized to give one substitution per site per unit time”? We will come back to these questions.

Example 5.12. For the Jukes-Cantor model of base substitutions, all changes have equal rate and all bases equal frequency.

$$Q = \begin{pmatrix} -1 & 1/3 & 1/3 & 1/3 \\ 1/3 & -1 & 1/3 & 1/3 \\ 1/3 & 1/3 & -1 & 1/3 \\ 1/3 & 1/3 & 1/3 & -1 \end{pmatrix} \quad \pi = (1/4, 1/4, 1/4, 1/4).$$

This is the simplest (and generally unrealistic) model of base substitutions.

Units of time When we want to work in some time units other than substitutions (for example years, or millions of years) we use a conversion factor μ . For example, if time t is measured in years, the units of μ are [subs/year] so that μt has units [subs] = [subs/year] \times [years]. Then $Q_{i,j} \times \mu t$ always has units [subs]. The quantity μ is called the **substitution rate**.

Example 5.13. For the HIV-I *env* sequence substitution model of [1], the substitution rate is very high, about $6e-5$ substitutions per site per day (the HIV generation time is of the order of one day) or about 0.02 substitutions per site per year. Notice that $1/(6 \times 10^{-5})$ gives 16667[days/sub] so 46 years is about the time it takes for one substitution to accumulate at each site in this HIV-I *env* base substitution model.

Stochastic model. Consider events at a single site:

suppose the character $x_{1,s}$ at site s at a given time t is i ; we assume that the probability $P_{i,j}(\delta)$ to find character j at site s a small time δ later on is proportional to the rate

$$P_{i,j}(\delta) = Q_{i,j}\delta + f(\delta),$$

with the addition of terms $f(\delta)$ satisfying $\lim_{\delta \rightarrow 0} f(\delta)/\delta = 0$, due to improbable multiple substitutions in the interval $(t, t + \delta]$. The discussion which follows is simply a sketch; we will be interested in the behavior of the substitution process as

$\delta \rightarrow 0$, and in this limit we may omit terms of order $f(\delta)$. See [6] Sections 6.8-6.9 for more detail.

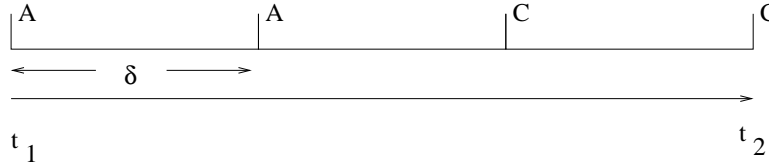
What about $Q_{i,i}$? Well $\sum_{j=1}^4 P_{i,j} = 1$ so $P_{i,i} = 1 - \sum_{i \neq j} P_{i,j}$. It follows that $P_{i,i} = 1 - \delta \sum_{i \neq j} Q_{i,j}$. We set

$$Q_{i,i} = - \sum_{j \neq i} Q_{i,j}$$

so that $P_{i,i} = 1 + \delta Q_{i,i}$. Notice that the diagonal entry in row i of Q is $Q_{i,i}$, and $Q_{i,i}$ is minus the sum of the other entries in row i . So the row sums of Q are zero (and the row sums of P are one).

How is $P(t)$ related to Q when the time intervals get larger, so we cant assume that there is at most one substitution in the interval t ?

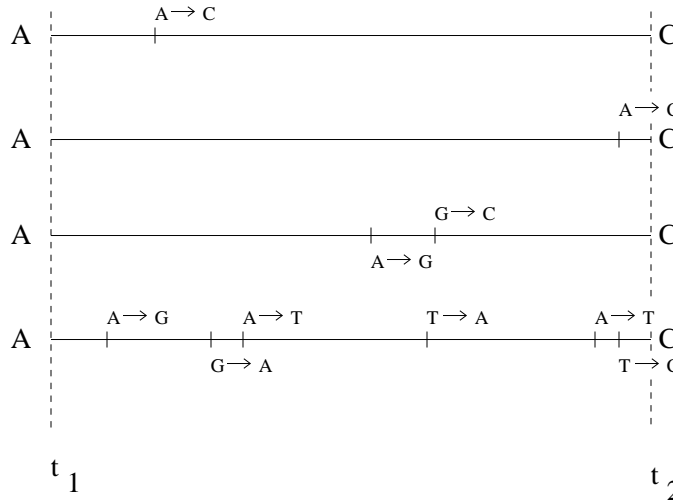
Transformation histories on a branch are sequences of characters and transition times. Here is an example to illustrate this idea. Consider three small time steps of length δ , $t_1 \rightarrow t_1 + \delta \rightarrow t_1 + 2\delta \rightarrow t_1 + 3\delta$. Suppose the base at a site starts out A , flips to C in the second step, and then stays C , as shown in the figure:



Because each of the time steps δ is small, we may assume that at most one transition occurred in each time step. Our assumption $P_{i,j}(\delta) \simeq Q_{i,j}\delta$ applies here so the probability to get from A to C by this path is

$$\begin{aligned} \Pr(A \rightarrow A \rightarrow C \rightarrow C) &= P_{A,A}(\delta)P_{A,C}(\delta)P_{C,C}(\delta) \\ &\simeq (1 + \delta Q_{1,1})Q_{1,3}\delta(1 + \delta Q_{3,3}) \end{aligned}$$

Over longer intervals of time $t \gg \delta$ the transition $i \rightarrow j$ may occur in many ways.



The nett probability $P_{i,j}(t)$ to get a j at site s in sequence x_2 at time t_2 given there was an i at site s in sequence x_1 at time t_1 is the sum, over all paths from an i

to a j , of the probability for that path to occur. We sketch how this sum may be computed.

If we break an interval $[0, t]$ into two intervals of length $t/2$, then the probability to find a site in character j at the end of the full interval given it was in character i at the start of that interval is

$$[P(t)]_{ij} = \sum_{k \in \mathcal{C}} [P(t/2)]_{i,k} [P(t/2)]_{k,j}$$

or in matrix notation, $P(t) = P(t/2)^2$. If we break an interval into m pieces, so that each piece covers a small interval of time, we have $P(t) = P(t/m)^m$. In our model, for small intervals $P(\delta) \simeq \mathbb{I} + Q\delta$, which suggests

$$\begin{aligned} P(t) &= P(t/m)^m \\ &= [\mathbb{I} + Q(t/m)]^m \\ &\rightarrow \exp(Qt) \quad \text{as } m \rightarrow \infty. \end{aligned}$$

The rate matrices we consider below are diagonalizable. Recall that if Q is diagonalizable (so we can write $Q = VDV^{-1}$ with D a diagonal matrix) then we compute $f(Q) = Vf(D)V^{-1}$, with $f(D)$ a diagonal matrix with $[f(D)]_{i,i} = f(D_{i,i})$. The sketch above is intended to make it clear that the matrix exponential $\exp(Qt)$ gives a transition probability which takes into account all possible transformation histories between the two endpoints. See [6] for the full picture.

For elapsed times $t \geq 0$, we have then

$$P_{i,j}(t) = [e^{Q\mu t}]_{i,j}$$

where $\exp(Q\mu t)$ is called a **matrix exponential**. If $M = Q\mu t$ then $\exp(M)$ is

$$e^M = \mathbb{I}_4 + M + \frac{1}{2!}M^2 + \frac{1}{3!}M^3 + \dots$$

the Taylor series for an exponential, using matrix multiplication $M^2 = MM$ etc.

Fortunately, most numerical software packages (MatLab, R and so on) will compute these matrix exponentials for you. Also, for some of the best known models, the matrix exponential has a simple form.

Example 5.14. In Example 5.12 we gave the rate matrix for the Jukes-Cantor model. In Example 5.3 we gave the transition matrix. These two things are related since if

$$Q = \begin{pmatrix} -1 & 1/3 & 1/3 & 1/3 \\ 1/3 & -1 & 1/3 & 1/3 \\ 1/3 & 1/3 & -1 & 1/3 \\ 1/3 & 1/3 & 1/3 & -1 \end{pmatrix}$$

then $P(t) = \exp(Q\mu t)$. If t is measured in substitutions (so $\mu = 1$ we are already in the right time units no conversion factor) then

$$P(t) = [e^{Qt}] = \begin{pmatrix} \frac{1+3e^{-4t/3}}{4} & \frac{1-e^{-4t/3}}{4} & \frac{1-e^{-4t/3}}{4} & \frac{1-e^{-4t/3}}{4} \\ \frac{1-e^{-4t/3}}{4} & \frac{1+3e^{-4t/3}}{4} & \frac{1-e^{-4t/3}}{4} & \frac{1-e^{-4t/3}}{4} \\ \frac{1-e^{-4t/3}}{4} & \frac{1-e^{-4t/3}}{4} & \frac{1+3e^{-4t/3}}{4} & \frac{1-e^{-4t/3}}{4} \\ \frac{1-e^{-4t/3}}{4} & \frac{1-e^{-4t/3}}{4} & \frac{1-e^{-4t/3}}{4} & \frac{1+3e^{-4t/3}}{4} \end{pmatrix}$$

Interpretation: (of $P = \exp(Q\mu t)$)

- (1) When t is small, $t \sim \delta$ say

$$e^{Q\delta} \simeq \mathbb{I} + Q\delta + O(\delta^2)$$

so

$$P_{i,j} = \delta Q_{i,j} + O(\delta^2)$$

when $i \neq j$ (since $\mathbb{I}_{i,j} = 0$ when $i \neq j$). Also,

$$P_{i,i} = 1 + Q_{i,i}\delta + O(\delta^2).$$

- (2) Item 1 decides the diagonal entries in Q . As we saw before, $\sum_{j=1}^4 P_{i,j} = 1$ implies

$$\begin{aligned} P_{i,i} &= 1 - \sum_{j \neq i} P_{i,j} \\ &= 1 - \delta \sum_{j \neq i} Q_{i,j}. \end{aligned}$$

In order to get agreement with Item 1 we need

$$Q_{i,i} = - \sum_{j \neq i} Q_{i,j}.$$

For this reason we often leave out the diagonal entries of Q when we write it down.

- (3) The total rate for events $i \rightarrow j$ (over a long time, at a single site) is $\mu\pi_i Q_{i,j}$. This says that a single site spends a fraction π_i of its time in the i -base state, and leaves that state for a j -base at rate $\mu Q_{i,j}$, so if you track events at a site you see $i \rightarrow j$ events at total rate $\mu\pi_i Q_{i,j}$. If $\pi_i Q_{i,j}$ equals $\pi_j Q_{i,j}$, then i 's are turning into j 's at the same rate j 's are turning into i 's, and it follows the process looks the same in either time direction. We cannot tell the direction of time by observations of the substitution process and we say the process is time-reversible.
- (4) Since $\pi P = \pi$ and $P \simeq \mathbb{I} + \delta Q$ (at small δ) it follows that $\pi Q = 0$. The point here is that if we want our substitution model to give some particular base frequencies $\pi = (\pi_A, \pi_C, \pi_G, \pi_T)$ we must choose Q so that $\pi Q = 0$. Then we will get $\pi P = \pi$ automatically.
- (5) The general time reversible (GTR) substitution model parameterization. It is common to parameterize Q in such a way that we get Items 3 and 4 automatically. We write

$$\begin{aligned} Q &= R\Pi \\ &= \begin{pmatrix} - & a & b & c \\ a & - & d & e \\ b & d & - & f \\ c & e & f & - \end{pmatrix} \begin{pmatrix} \pi_A & 0 & 0 & 0 \\ 0 & \pi_C & 0 & 0 \\ 0 & 0 & \pi_G & 0 \\ 0 & 0 & 0 & \pi_T \end{pmatrix} \\ &= \begin{pmatrix} - & \pi_C a & \pi_G b & \pi_T c \\ \pi_A a & - & \pi_G d & \pi_T e \\ \pi_A b & \pi_C d & - & \pi_T f \\ \pi_A c & \pi_C e & \pi_G f & - \end{pmatrix} \end{aligned}$$

and $Q_{i,i} = -\sum_{j \neq i} Q_{i,j}$ so that for example $Q_{1,1} = -\pi_C a - \pi_G b - \pi_T c$. The parameters a, c, d, e and f are called relative rates. When we define Q we often set $f = 1$ and give the other rates relative to $G \leftrightarrow T$. We use $\tilde{\mu}$ to scale back to one substitution per unit time.

- (6) If $\tilde{\mu}\pi_i Q_{i,j}$ is the total rate for substitutions $i \rightarrow j$ at a site (as in Item 3), the total rate T for all substitutions at a site is the sum over the rates for all possibilities,

$$T = \tilde{\mu} \sum_{i=1}^4 \pi_i \sum_{j \neq i} Q_{i,j}$$

or $T = \tilde{\mu} \sum_{i=1}^4 \pi_i Q_{i,i}$. We usually choose $\tilde{\mu}$ so that $T = 1$, and this formula shows us how to do it.

Exercise 5.15. Show that $\pi Q = 0$ for the GTR Q in Item 5.

Exercise 5.16. Show that $\pi_i Q_{i,j} = \pi_j Q_{j,i}$ for the GTR Q in Item 5.

Famous Models

The GTR form for Q in Item 5 is the most general in widespread use (ignoring extensions which accommodate site correlations in various forms, codons *etc*). Since π sums to one, the GTR Q has nine parameters (counting μ , and ignoring trivial rescalings which are fixed by setting $f = 1$). The GTR model allows distinct base frequencies π and distinct rates for all pairs $A \leftrightarrow C \dots G \leftrightarrow T$. A minor negative for the GTR model: it is not possible to calculate $P = \exp(Q\mu t)$ in any simple way. We have to rely on numerical software. However, such software is now widely available, efficient and easy to drive.

A number of other Q matrices in widespread use are special cases of GTR obtained by fixing some of its parameters to special values. The aim here is to get a simpler form with fewer parameters, either to make the analysis simpler, or to avoid over-parameterizing (when sequences are short *etc*).

The HKY85 model allows unequal base frequencies and distinguishes transitions ($A \leftrightarrow G$ and $C \leftrightarrow T$) from transversions (all other substitutions). The parameterization is

$$Q = \begin{pmatrix} - & \pi_C & \pi_G k & \pi_T \\ \pi_A & - & \pi_G & \pi_T k \\ \pi_A k & \pi_C & - & \pi_T \\ \pi_A & \pi_C k & \pi_G & - \end{pmatrix}$$

The idea here is that we choose $k > 1$ to favor transitions. This model has 5 parameters. It is handy as $P = \exp(Q\mu t)$ is available in closed form. In [7] the transition probability for this model and several others are given explicitly in terms of the model rates. Chapter 13 of [5] gives a good overview of the subject.

Exercise 5.17. calculate $\tilde{\mu}$ using the formula in Item 6 so that the HKY85 model has unit rate.

The Jukes-Cantor model is even simpler. All transitions have equal relative rates so and the equilibrium base frequencies are equal. The Q matrix is given in Item 5.12 above. Again, $P = \exp(Q\mu t)$ is available in closed form and is given in Item 5.3 above.

Exercise 5.18. check that Q in Example 5.12 has unit rate, so it can be used with $\mu = 1$ if t is in [subs].

6. LIKELIHOOD METHODS II: MAXIMUM LIKELIHOOD

The material in the section is discussed in Chapter 8 (Sections 8.1-8.3) of [2] and at the start of Chapter 16 in [5].

We have defined a stochastic process modelling the way DNA sequences evolve over time. Imagine fixing on some particular tree, τ with n leaves. By a tree, we mean for the moment a rooted tree topology, with branch lengths (measured in substitutions) assigned to each branch. Consider DNA sequence data, *ie*, n sequences of length N , one sequences associated with each leaf. Focus on a single site s . At leaf a we observe base $x_{a,s}$ at this site. What is the probability to get any particular site pattern $D_s = (x_{1,s}, x_{2,s}, \dots, x_{n,s})$ at site s on this tree?

Denote by $P(D_s|\tau, Q)$ the probability to generate site pattern D_s at the leaves of the tree, given a tree τ , and a rate matrix Q (notice that giving Q would specify π , the corresponding equilibrium base frequencies). Let $P(D|\tau, Q)$ be the probability to generate all N sites of the n sequences observed at the leaves. Since we assume sites evolve independently,

$$P(D|\tau, Q) = \prod_{s=1}^N P(D_s|\tau, Q).$$

The function $P(D|\tau, Q)$ is called the likelihood of Q and τ .

Defer for the moment the problem of actually calculating the likelihood $P(D|\tau, Q)$. How does all this help us determine the phylogeny τ from sequence data D ? Suppose τ_1 and τ_2 were two trees and Q , the rate parameters, were known. The data should help us decide between the two trees. We could ask, “on which tree is the data a more likely outcome”, that is we compute

$$r_{1,2} = \frac{P(D|\tau_1, Q)}{P(D|\tau_2, Q)}$$

and favor tree τ_1 if $r_{1,2} > 1$.

We define the maximum likelihood tree, τ^* say, to be that tree maximizing $P(D_s|\tau, Q)$ as a function of τ , that is, if T is our space of all possible trees (and branch lengths) with n leaves,

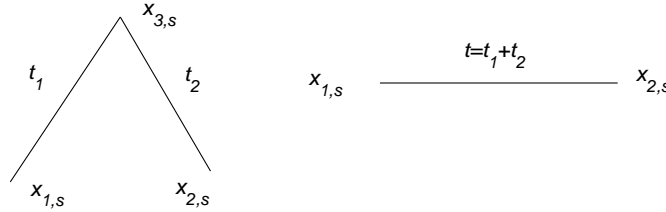
$$\tau^* = \arg \max_{\tau \in T} P(D|\tau, Q).$$

We choose the tree which makes the data as likely as possible an outcome. This is all much the same as parsimony: there we had an objective function measuring the ‘cost’ of a tree in substitutions, and we chose the tree by minimising the tree length and thereby maximising parsimony; here we have an objective function measuring how ‘good’ a tree is (at explaining the data), and we choose the tree which maximises likelihood.

Notice that the same principle could be applied to estimate Q : choose (τ^*, Q^*) to maximize $P(D|\tau, Q)$ simultaneously over all possible Q and τ .

6.1. Likelihood calculations for finite-sites substitution. Now, how do we compute $P(D_s|\tau, Q, \pi)$? The way to answer this question is to imagine how nature generates the site pattern.

Example 6.1. Consider the small rooted tree at left below:



How do we compute $P(D|\tau, Q)$ for this tree? The branches above nodes 1 and 2 are respectively t_1 and t_2 substitutions in length, and this information specifies τ . The characters $x_{1,s}$ and $x_{2,s}$ at leaf nodes 1 and 2 are observed. Suppose $D_s = (x_{1,s}, x_{2,s}) = (j, k)$. Now, $x_{3,s}$, the ancestral character at the root, node 3 here, is not observed. If we knew $x_{3,s} = i$ say, it would be easy to compute the probability

$$P(x_{1,s} = j, x_{2,s} = k | x_{3,s} = i, \tau, Q) = P_{i,j}(t_1)P_{i,k}(t_2)$$

to arrive at nodes 1 and 2 with characters j and k respectively, given that we started at node 3 with character i , since the evolution is independent down the two branches $\langle 3, 1 \rangle$ and $\langle 3, 2 \rangle$. Since we do not know the ancestral character $x_{3,s}$, we must average over all possible values it might take. The probability $\pi_i = \Pr(x_{3,s} = i)$ to have any particular base i at node 3 is just the equilibrium base frequency, and hence

$$P(x_{1,s} = j, x_{2,s} = k | \tau, Q) = \sum_{i \in \mathcal{C}} P_{i,j}(t_1)P_{i,k}(t_2)\pi_i.$$

This is essentially just $P(a) = \sum_b P(a|b)P(b)$. The quantity $P(x_{1,s} = j, x_{2,s} = k | \tau, Q)$ is $P(D_s|\tau, Q)$, the probability for the data at site s given the “parameters” Q and τ .

Example 6.2. Consider the setup in Example 6.1, and suppose we choose the Jukes-Cantor model of base substitution. We have some DNA sequence data for the organisms at the two leaves: the observed base sequences on the unrooted two leaf tree at right in Example 6.1 are $x_1 = AG$ and $x_2 = AA$. Compute a maximum-likelihood value for $t = t_1 + t_2$. The JC rate matrix is given in 5.12 and the corresponding expression for $P = \exp(Qt)$, for t in units of substitutions, is given in 5.3. The JC equilibrium base frequencies are $\pi_i = 1/4$, $i = 1, 2, 3, 4$.

Abbreviate $P(D|\tau, Q)$ as $P(D|t)$, since we are assuming Q is known, and τ is specified by its single branch length t . Now Jukes-Cantor is time reversible, so we can root the tree at an arbitrary point, and get the same expression for the probability of the data. Rooting the tree as at left in the figure above, and using

the results we wrote down for that case,

$$\begin{aligned}
 P(D_1|t) &= \sum_{i \in \mathcal{C}} P_{i,A}(t_1)P_{i,A}(t_2)\pi_i \\
 &= \frac{3}{4} \left[\frac{1}{4} - \frac{1}{4}e^{-4t_1/3} \right] \left[\frac{1}{4} - \frac{1}{4}e^{-4t_2/3} \right] + \frac{1}{4} \left[\frac{1}{4} + \frac{3}{4}e^{-4t_1/3} \right] \left[\frac{1}{4} + \frac{3}{4}e^{-4t_2/3} \right] \\
 &= \frac{3}{16} \exp(-4t_1/3 - 4t_2/3) + \frac{1}{16}
 \end{aligned}$$

and

$$\begin{aligned}
 P(D_2|t) &= \sum_{i \in \mathcal{C}} P_{i,G}(t_1)P_{i,A}(t_2)\pi_i \\
 &= \frac{2}{4} \left[\frac{1}{4} - \frac{1}{4}e^{-4t_1/3} \right] \left[\frac{1}{4} - \frac{1}{4}e^{-4t_2/3} \right] + \frac{1}{4} \left[\frac{1}{4} + \frac{3}{4}e^{-4t_1/3} \right] \left[\frac{1}{4} - \frac{1}{4}e^{-4t_2/3} \right] \\
 &\quad + \frac{1}{4} \left[\frac{1}{4} - \frac{1}{4}e^{-4t_1/3} \right] \left[\frac{1}{4} + \frac{3}{4}e^{-4t_2/3} \right] \\
 &= \frac{1}{16} - \frac{1}{16} \exp(-4t_1/3 - 4t_2/3)
 \end{aligned}$$

so that

$$\begin{aligned}
 P(D|t) &= \left(\frac{3}{16} \exp(-4t_1/3 - 4t_2/3) + \frac{1}{16} \right) \left(\frac{1}{16} - \frac{1}{16} \exp(-4t_1/3 - 4t_2/3) \right) \\
 &= \frac{1}{256} (2 \exp(-4t/3) - 3 \exp(-8t/3) + 1).
 \end{aligned}$$

The maximum likelihood estimate for t is the solution $t = 3 \log(3)/4$ of

$$\frac{d}{dt} P(D|t) = 0.$$

The branch is about 0.82 substitutions in length. The Maple worksheet used to carry out this straightforward but tedious calculation is bundled with these notes.

Exercise 6.3. Show that, if we had rooted the branch $\langle 1, 2 \rangle$ at node 1 (instead of adding node 3 and making that the root) the above calculations are greatly simplified. The point is that there are now no unknown internal states in this configuration, so there is no summation. Here are the steps. The Jukes Cantor substitution process is reversible. Use this property to show that

$$\sum_{i \in \mathcal{C}} P_{i,x_1,s}(t_1)P_{i,x_2,s}(t_2)\pi_i = \pi_{x_1,s} P_{x_1,s,x_2,s}(t)$$

and hence

$$P(D|t) = \pi_A \left[\frac{1}{4} + \frac{3}{4}e^{-4t/3} \right] \times \pi_G \left[\frac{1}{4} - \frac{1}{4}e^{-4t/3} \right]$$

Check that this expression is equal to the one computed above.

6.2. The gap character. We need to allow for gaps in DNA sequences (represented by the ‘-’ character. These are interpreted as sites where the value of the base was not observed, and so any base character is allowed at the gap site. They

are easy to accommodate in our calculation: the transition probability from any character $i \in \mathcal{C}$ at site s at node a to the gap character at site s at node b is

$$\begin{aligned} P_{i,-}(t_b - t_a) &= \Pr(x_{2,s} = \text{A or C or G or T} | x_{1,s} = i) \\ &= 1 \end{aligned}$$

since the event $x_{2,s} = \text{A or C or G or T}$ occurs with certainty.

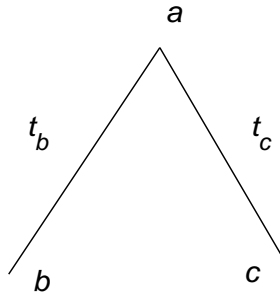
We have only to worry about transitions *to* the gap character, since gaps are present in the data only, *ie* at the leaves, and not at internal nodes of the tree.

6.3. The pruning algorithm. When the number of trees is large, we must sum over a large number of internal states in order to compute the likelihood $P(D_s | \tau, Q)$ (n leaves, so $n - 1$ internal nodes on a rooted tree, that's $4^{(n-1)}$ distinct internal states for the characters $\{A, C, G, T\}$). A dynamic programming algorithm, referred to by [4] as the pruning algorithm, computes the summation efficiently.

Consider a rooted tree τ with root node r and data $D = (D_1, D_2, \dots, D_N)$. We want to compute $P(D_s | \tau, Q)$ (and then take the product over sites $s = 1, 2, \dots, N$ to get $P(D_s | \tau, Q)$, the full likelihood). Suppose we could work out $P(D_s | \tau, Q, x_{r,s} = i)$, the probability for our substitution process to generate the characters in our data at site s at the leaves, given that we know that the substitution process started with character i at the root. We don't know what the root character was, so we sum it out, weighted by its probability:

$$P(D_s | \tau, Q) = \sum_{i \in \mathcal{C}} P(D_s | \tau, Q, x_{r,s} = i) \pi_i$$

(again, this is essentially $P(b) = \sum_a P(b|a)P(a)$). This is a sum with just $|\mathcal{C}|$ addends, so we would have $P(D_s | \tau, Q)$ if we had $P(D_s | \tau, Q, x_{r,s} = i)$ for each $i \in \mathcal{C}$. How to compute $P(D_s | \tau, Q, x_{r,s} = i)$? Consider the configuration shown in the figure:



Node a is a node somewhere in a rooted tree τ , with descendants at nodes b and c . Denote by $D_s^{(a)}$ the site- s data on all leaves descended from node a . What is the probability for our substitution process to generate that pattern of characters $D_s^{(a)}$ at the leaves, given some character i at node a ? Well,

$$P(D_s^{(a)} | \tau, Q, x_{a,s} = i) = P(D_s^{(b)} | \tau, Q, x_{a,s} = i) P(D_s^{(c)} | \tau, Q, x_{a,s} = i),$$

the probability to generate the data below node a , given $x_{a,s} = i$ is equal to the probability to generate the sequences descended from node b times the corresponding probability for node c . But,

$$P(D_s^{(b)}|\tau, Q, x_{a,s} = i) = \sum_{j \in \mathcal{C}} P_{i,j}(t_b)P(D_s^{(b)}|\tau, Q, x_{b,s} = j),$$

that is, to get the data under b we could have any character j at b , and then we have the probability to get the data under b given a j at b . We now have the kind of iteration we need. Let

$$L_s^{(a)}(i) = P(D_s^{(a)}|\tau, Q, x_{a,s} = i),$$

be the subtree-likelihood at a . Recall we want to compute $L_s^{(r)}(i) = P(D_s|\tau, Q, x_{r,s} = i)$ for $i \in \mathcal{C}$. We compute up from the leaves. If a is a leaf, set $L_s^{(a)}(i) = 1$ if $x_{a,s} = i$ and $L_s^{(a)}(i) = 0$ if $x_{a,s} \neq i$. Now, for each ancestral node a with child nodes b and c ,

$$(6.1) \quad L_s^{(a)}(i) = \sum_{j \in \mathcal{C}} P_{i,j}(t_b)L_s^{(b)}(j) \times \sum_{k \in \mathcal{C}} P_{i,k}(t_c)L_s^{(c)}(k).$$

In this way $L_s^{(a)}(i)$ is determined from the leaves up to the root.

Exercise 6.4. How do we handle the case where node a is a leaf and $x_{a,s} = '-'$, the gap character?

Exercise 6.5. Show that pruning recursion Eqn. 6.1 gives the same expression for the likelihood of the rooted two leaf tree of Example 6.1 as that we computed in Section 6.1.

7. LIKELIHOOD METHODS III: MODELS OF AMINO ACID AND CODON SUBSTITUTION

The material in the section is discussed in Chapter 14 of [5].

In biology there are exceptions to every rule (except the next one). The generalizations in this section are no exception to the rule (except the previous one...).

We have discussed neutral, finite-sites models of nucleotide-base substitution. When a cell divides, each of its offspring receives a copy of the DNA sequences in the original cell. We have modelled errors in this copy process. However, DNA acts in the cell through the proteins it encodes; the protein a gene encodes evolves as the gene evolves. DNA-base substitutions which encode faulty proteins are removed by negative selection. DNA-base substitutions which enhance protein function are fixed by positive selection. The sequences we observe in the present are at the leaf tips of lineages which came through this selection process. When we model the substitution process acting down an ancestral lineage, we are in effect modelling a substitution process thinned by these and other selection effects. How might this affect things?

Recall the central dogma of genetics: a DNA base sequence is transcribed sequentially, without loss of information, into tRNA; the tRNA base sequence is translated, in groups of 3 bases (codons), into a sequence of amino acids (*ie*, protein). There are 20 amino acids. The character set is

$$\mathcal{C}_A = \{A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, Y\}.$$

Here are a few examples of the codon mapping (laid out in detail in the ppt slides from lecture 1). The DNA codons ACA , ACC , ACG and ACT code for the amino acid T (threonine). The codons ATT , ATC and ATA code for I (isoleucine) and ATG encodes M (methionine).

In order to "read" a DNA-base sequence as amino acids, we need to know where the codons start. This is the codon reading frame. Suppose we have a gene sequence with codon ACA and the substitution $ACA \rightarrow ACC$ occurs. The coded amino acid remains T . Substitutions like this, which do not change the amino acid, are called synonymous (silent) substitutions. The substitution $ACA \rightarrow ATA$ does change the coded amino acid (from T to I), and is called non-synonymous.

Whether selection acts positively (so non-synonymous changes are present in disproportionate numbers on ancestral lineages) or negatively (non-synonymous changes are thinned), one effect of this kind of selection will be to correlate substitutions between bases in the same codon. Our neutral model, which treated each site as evolving independently, will be wrong. Here are a few ways we might alter the model to respond.

7.1. Rate variation at the wobble site. Notice that the third base in the codons ACA , ACC , ACG and ACT carries no information about the amino acid (they're all T). This rule does not hold for all the codon to amino-acid mappings, but it is nevertheless a useful generalization. The third position is called the wobble position. Variation at the wobble position is often synonymous. We could model this by allowing the three codon positions to evolve at distinct total rates μ_1 , μ_2 and μ_3 . Normalize the rate matrix to one substitution per unit time at position 1, so $\mu_1 \equiv 1$, and define a tree $\tau' = \mu\tau$ identical to tree τ , except having branch lengths scaled by μ . Suppose $p(s) \in \{1, 2, 3\}$ maps sites $s = 1, 2, \dots, N$ to their codon position. We replace $P(D_s|\tau, Q)$ with $P(D_s|\mu_{p(s)}\tau, Q)$ in the expression for the likelihood. A similar more general strategy assigns a distinct rate matrix $Q^{(1)}$, $Q^{(2)}$, $Q^{(3)}$ at each codon position. In this case $P(D_s|\tau, Q)$ becomes $P(D_s|\mu_{p(s)}\tau, Q^{(p(s))})$ with each matrix at unit rate.

7.2. Models of protein substitution. One strategy is to give up on models of DNA-base substitution, and model the evolution of the protein sequences themselves. [5] reviews the literature on models of this kind. The set up is essentially the same as for DNA-bases. We have now a 20×20 rate matrix $Q_{i,j}$, $i, j \in \mathcal{C}_A$. The problem of estimating the parameters of this model (Exercise: show that the GTR model would have 208 parameters, not counting μ) is substantial. The strategy has been to take long protein sequences from organisms at the leaf tips of known phylogenies, and maximize the likelihood $Q^* = \arg \max_Q P(D|\tau, Q)$, varying Q rather than τ to obtain the maximum likelihood parameter values.

One feature of this protein-level model is that it allows instantaneous amino acid substitutions which would correspond to multiple, simultaneous, base substitutions. For example the amino acid W (tryptophan) has just one codon, TGG . The transition $W \rightarrow M$ corresponds to a codon substitution $TGG \rightarrow ATG$ involving two base substitutions. The number of parameters in the protein-level model may be reduced by imposing $Q_{i,j} = 0$ for all pairs $i, j \in \mathcal{C}_A$ of this kind.

Exercise 7.1. (not examinable) what is the mathematical relation between a base substitution model and an amino acid substitution model? A base substitution model determines a (marginal) stochastic amino acid substitution process, but what are the properties of that process? Is it Markov? Can the marginal amino acid substitution process be represented in terms of a transition matrix expressed as a matrix exponential? Suppose we had a satisfactory base substitution model. How might we estimate a Q -matrix for an amino acid substitution model which approximated the marginal amino acid process of the base model?

7.3. Models of codon substitution. Models of codon substitution have been developed in the last seven years, and are now available in general purpose phylogenetic software packages. The usual assumptions are applied, but this time at the level of codons. Codons are assumed to evolve independently. There are 61 distinct codons $\mathcal{C}_C = \{AAA, AAC, \dots, TTT\}$ (ignoring the 3 stop codons TAA, TGA and TAG) and a corresponding 61×61 matrix \tilde{Q} of rate parameters and a 1×61 vector $\tilde{\pi}$ of equilibrium codon frequencies (satisfying $\tilde{\pi}\tilde{Q} = 0$). The codon substitution rates are determined from a model of base substitutions, with an extra weighting to penalize (or promote) non-synonymous substitutions. For example, suppose we start with base model $Q = R\Pi$ with symmetric relative rates $R_{a,b}$, $a, b \in \mathcal{C}$ and equilibrium base frequencies π_a , $a \in \mathcal{C}$. If $i, j \in \mathcal{C}_C$ are two codon states at some codon, then

$$\tilde{Q}_{i,j} = \begin{cases} 0 & \text{if } i \text{ and } j \text{ differ at more than one position} \\ R_{a,b}\tilde{\pi}_j & \text{if } i, j \text{ differ by the base substitution } a \rightarrow b \\ & \text{and } i \rightarrow j \text{ is synonymous} \\ \omega R_{a,b}\tilde{\pi}_j & \text{if } i, j \text{ differ by the base substitution } a \rightarrow b \\ & \text{and } i \rightarrow j \text{ is non-synonymous} \end{cases}$$

Exercise 7.2. check that this specifies a reversible model with equilibrium codon frequencies $\tilde{\pi}$.

Exercise 7.3. check that this model is identical to the corresponding neutral finite-sites model of base substitution when $\omega = 1$.

The number of parameters in this model is manageable if some assumption can be made about the equilibrium codon frequencies $\tilde{\pi}_i, i \in \mathcal{C}_C$. For example, if i is the codon abc , it is common to assume $\tilde{\pi}_i = \pi_a\pi_b\pi_c$.

The likelihood $P(D|\tau, \tilde{Q})$ for the phylogeny τ and the codon-rate parameters \tilde{Q} is now a product over codon-sites $s = 1, 2, \dots, N_C$ (where presumably, $N_C \simeq N/3$) of the probability $P(D_s|\tau, \tilde{Q})$ to generate the codon-site pattern D_s , for the given tree and rates. The pruning algorithm is used to compute this likelihood.

The parameter ω (often " d_N/d_S ") is of great interest in the analysis of genetic data. It is the ratio of the rate for a non-synonymous substitution to that of a synonymous substitution. A value of $\omega > 1$ indicates positive (diversifying) selection; $\omega < 1$ indicates negative (conserving) selection; $\omega = 1$ is the neutral case. For example, the HIV envelope gene shows extraordinarily high levels of diversifying selection, as it evades the human immune response.

Until recently, models assumed a constant ω over all (codon) sites. However, different types of selection might act at different sites. [8] replace ω by ω_s , that is, they allow the d_N/d_S -ratio ω to vary with codon position.

7.4. Protein structure. We have so far looked at just one model-mispecification issue: that due to selection on codons. Protein function is sensitive to the way the chain of amino acids making up the protein folds in 3D. Substitutions which affect folding are unlikely to find their way to the leaves of a phylogeny (though they must occur in deep phylogenies). *Ab initio* folding computation is just feasible, for short sequences. However, a quantitative understanding of the impact of structure on substitution, and *vis versa*, is work-in-progress.

7.5. Constant rates. We have, in this section, considered variation of rates across sites. We have so far avoided the problem of reconstructing phylogenies with branch lengths in calendar time units. There is considerable interest in the time depth of phylogenies, and time depth is estimated whenever feasible. We need to have some independent means of estimating μ , the number of substitutions per unit calendar time at a site, in order to convert branch lengths measured in substitutions into branch lengths measured in calendar units. Also, we need to be confident that the same rate applies on each branch of the phylogeny (so, in all the genes which contribute to our data sequences, substitutions occurred at equal rate for all ancestral species represented in the phylogeny). This is called the "molecular clock" assumption and the phylogeny "clock-like". The branch lengths of a clock-like tree are proportional to elapsed calendar time.

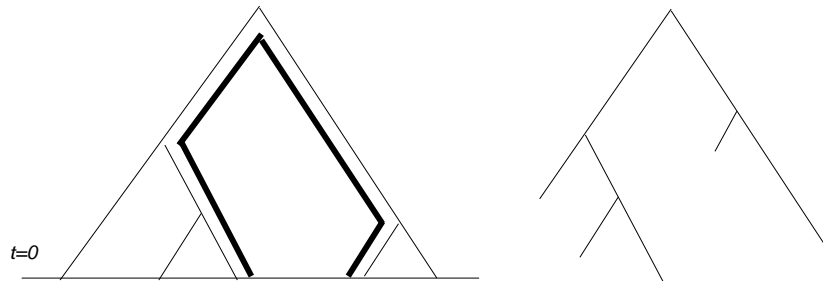
Denote by T'_n the set of rooted clock-like phylogenies with n leaves. This space T'_n of trees is certainly smaller than T_n , the space of rooted phylogenies with n leaves. Leaf node times of a clock-like tree are determined by the time at which the corresponding taxa were sampled. There are $n - 1$ free variables, the ages of the $n - 1$ ancestral nodes in the phylogeny. These are constrained by the phylogeny, since parents nodes antedate their child nodes. In the non-clock tree there are $2n - 2$ free variables (one positive length for each edge of a rooted n -leaf tree).

The maximum likelihood clock-like phylogeny τ^* is obtained as before,

$$\tau^* = \arg \max_{\tau \in T'_n} P(D|\tau, Q),$$

but maximizing the likelihood over the restricted space of clock-like trees.

The constraint is illustrated in the following figure. In this example, the taxa are isochronous. In a clock like tree they must then be equidistant from the root.



Under the molecular clock hypothesis, we are able to estimate a root position, even though we are fitting a GTR substitution model, since varying root position varies branch length when the leaf times are fixed in this way. If we estimate the branch lengths using maximum likelihood, without the constraint of a molecular clock, and root the tree using for example, an outgroup (as discussed in connection with

parsimony), we will reconstruct leaf times separated by variable amounts of time (ie variable numbers of substitutions) from the root.

8. BAYESIAN INFERENCE OF PHYLOGENY

The discussion in [5] Chapter 18 may be useful. The author is famously skeptical of Bayesian inference.

So far we have looked at parsimony and maximum likelihood methods for tree-estimation from sequence data. We will now consider a third inference scheme, Bayesian inference. Why bother? Very often we have information about the phylogeny and rate parameters coming from sources other than the sequence data. Bayesian inference gives us a framework to bring that information into the analysis. Also we can quantify certain forms of uncertainty in a straightforward way.

8.1. A simple example. Suppose we have two dice, one (F) fair, the other (U) throws only 4,5 or 6 (but is otherwise fair). A fair coin with sides labelled F and U is used to choose a die. It is tossed, a die is chosen, and the die thrown twice and the upside recorded. In one such experiment the outcomes were 4 and 6. Was the die fair or unfair?

Let Θ denote the unknown true state of the coin. We are interested in $\Pr(\Theta = U|D = \{4, 6\})$. Now, by Bayes rule,

$$\Pr(\Theta = U|D = \{4, 6\}) = \Pr(D = \{4, 6\}|\Theta = U) \Pr(\Theta = U) / \Pr(D = \{4, 6\}).$$

We have $\Pr(D = \{4, 6\}|\Theta = U) = 2/9$ (order doesnt matter) and $\Pr(\Theta = U) = 1/2$. The constant $\Pr(D = \{4, 6\})$ is a normalising constant. We require $\Pr(\Theta = F|D = \{4, 6\}) + \Pr(\Theta = U|D = \{4, 6\}) = 1$, and since $\Pr(D = \{4, 6\}|\Theta = F) = 2 \times 1/36$ we have $\Pr(D = \{4, 6\}) = 2/9 \times 1/2 + 2/36 \times 1/2$, which is $\Pr(D = \{4, 6\}) = 5/36$. We have $\Pr(\Theta = U|D = \{4, 6\}) = (2/9) \times (1/2) / (5/36)$ which is

$$\Pr(\Theta = U|D = \{4, 6\}) = 4/5 \quad \text{and} \quad \Pr(\Theta = F|D = \{4, 6\}) = 1/5.$$

There is an 80% chance the die is unfair.

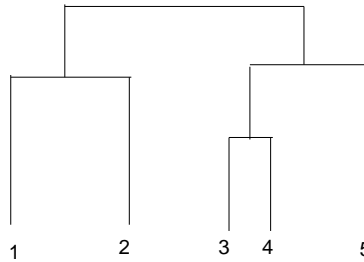
If we stopped before throwing the die, and asked for the probability the die is unfair, we would be told $\Pr(\Theta = U) = 1/2$. This represents our state of knowledge before the arrival of the data (the results of the throws). We regard $\Theta = U$ and $\Theta = F$ as equally likely *a priori*, since $\Pr(\Theta = U) / \Pr(\Theta = F) = 1$. Our state of knowledge changes with the arrival of the data. Now $\Theta = U$ is $\frac{\Pr(\Theta=U|D=\{4,6\})}{\Pr(\Theta=F|D=\{4,6\})} = 4$ times more probable than $\Theta = F$.

The distribution $\Pr(\Theta = \theta)$ is called the prior probability distribution. The distribution $\Pr(\Theta = \theta|D = \{4, 6\})$ is called the posterior probability distribution, since it describes our state of knowledge after the arrival of the data.

Notice that if we base inference on the posterior probability distribution, we are able to answer a question like "does the parameter posses such-and-such a property?" (ie is $\Theta = U$) with a probability. This is one of the attractions of this form of inference: we express our confidence in our conclusion in this straightforward way.

8.2. **Bayesian inference.** Biologists often ask questions like "does this new virus belong to this group or that group". The details of the phylogeny are of secondary interest. How do we answer questions of this kind?

Consider the following problem. We are given some sequence data, and asked to decide whether or not the associated phylogeny contains a particular clade. This kind of question is fundamental in phylogenetics. m taxa form a clade of an n -leaf tree if they lie in a subtree which can be separated from the $n - m$ remaining taxa by cutting a single edge. For rooted trees we require that the subtree does not contain the root.



In the figure, taxa (3,4,5) form a clade, taxa (1,2,5) do not, though they may be separated from the others by cutting a single edge.

A bad way to answer a question of this sort would be to compute the parsimony or ML tree, see if it contained the clade of interest, and reply "yes" or "no" accordingly. The weakness of such an approach is that there may be other trees, with different topologies and branch lengths, which have likelihood scores just slightly lower than the likelihood of the ML tree. If we considered one of these trees we might reach a different conclusion. The unknown true phylogeny might be one of these also-ran trees - the data is an outcome of a random substitution process, and the data is still a relatively probable outcome on these trees. Of course, this is the problem of giving a confidence interval for a parameter estimate.

The problem might be approached as follows: what is the probability that the unknown true tree \mathbb{T} contained the clade of interest, given the data? A clade is just one feature of a tree. We might more generally ask, what is the probability the unknown true tree possessed some property, property- S say? Denote by T the space of all trees, and let S be the set of trees in T possessing the property of interest. Suppose the parameters of our base-substitution model, Q , are known. We need to give some meaning to the probability distribution $P(\mathbb{T} \in S|D, Q)$. We will drop Q from the following for the moment.

We can deal with this by defining a density on tree-space, $h(\tau|D)$, the so called "posterior probability density". For any set $S \subseteq T$ of interest we have

$$P(\mathbb{T} \in S|D) = \int_S h(\tau|D) d\tau.$$

In order to compute this integral we sum over distinct tree topologies in S and integrate over branch-lengths. We can define h in terms of the likelihood, $P(D|\tau)$ and a second density function $p(\tau)$ called the prior probability density for tree τ :

$$h(\tau|D) = \frac{P(D|\tau)p(\tau)}{P(D)}.$$

The above relation is just Bayes rule $P(A|B) = P(B|A)P(A)/P(B)$ for a density. In this relation, $P(D)$ is a normalizing constant. Since $P(\mathbb{T} \in T|D) = 1$, we have

$$P(D) = \int_T P(D|\tau)p(\tau)d\tau.$$

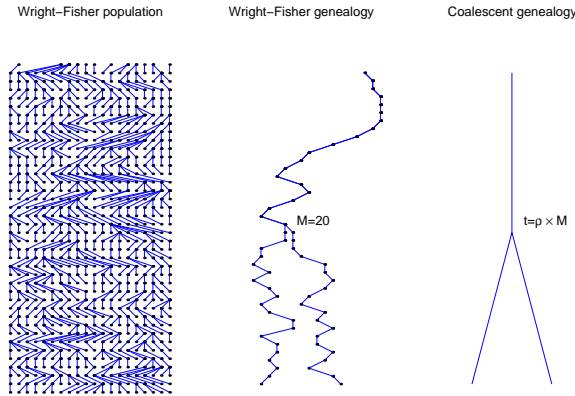
How do we interpret p and h ? Suppose we had no sequence data. We might still have some information about the phylogeny. We assume this information can be expressed as a prior probability density on tree space. We are using probability distributions to summarize our state of knowledge. The arrival of the data changes our state of knowledge about the phylogeny. The probability that the unknown true tree possessed property- S , before the data arrived was $P(\mathbb{T} \in S) = \int_S p(\tau)d\tau$. The probability that the unknown true tree possessed property- S , after the data arrived is $P(\mathbb{T} \in S|D) = \int_S p(\tau|D)d\tau$.

8.3. Priors on trees. What kinds of prior information arise in phylogenetics?

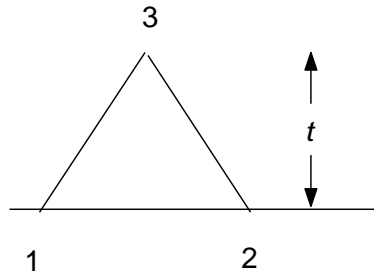
- (frequentist) When we have a model of the stochastic process which generated the unknown true phylogeny, the prior $p(\tau)$ is the probability density for this process to generate tree τ .
- (posterior becomes prior) We may have access to some previous study of the same kind as our own, which yielded a posterior distribution $h(\tau|D')$ on the same tree space. In this case, $p(\tau) = h(\tau|D')$, the posterior distribution of the previous study becomes the prior in our own.
- (subjective constraints) We may have information from quite different studies, for example studies of body morphology, or information from the fossil record, which make certain phylogenies highly improbable or impossible. We constrain the space of trees accordingly.
- (subjective) We may impose a generic model (such as the Yule process) of phylogeny, weighting in favor of phylogenies with shapes and depths in some broad class thought likely *a priori*.
- (ignorance) We may assert that any phylogeny is as probable as any other. This is typically represented by imposing $p(\tau) = p(\tau')$ for $\tau, \tau' \in T$ any two trees. We will see that care must be taken in imposing priors of this kind.

Example 8.1. The following is an example of a frequentist tree-prior from population genetics. Population genetics is concerned with ancestral inference within single species, whilst phylogenetic inference is concerned inference of phylogenies - species trees. Many of the issues are the same. Consider two individuals sampled from the present in a haploid population of known fixed size N_e . Their ancestral tree has one parameter, the number of generations back to the common ancestor. In the Wright-Fisher model, each individual in each child generation chooses its parent

uniformly at random from the individuals in the previous generation.



Tracing back in time from the present, the probability that the ancestral lineages coalesce at generation M is $N_e^{-1}(1 - N_e^{-1})^{(M-1)}$. It will turn out that the expected time back to coalescence is N_e generations, so choose units of time so that N_e generations is $t = 1$ time units and one generation is $\rho = N_e^{-1}$ time units. On this time scale $t = M\rho$ and $(1 - \rho)^{(t/\rho)-1}$ is approximately $\exp(-t)$. It follows that, for largish population sizes, the time back to coalescence is distributed exponentially, $t \sim \text{Exp}(1)$ with mean at $t = 1$.



This is an example of a tree prior. Since a two leaf tree with isochronous leaves is

$$\tau = \{V = \{1, 2, 3\}, E = \{\langle 3, 1 \rangle, \langle 3, 2 \rangle\}, (t_1, t_2, t_3) = (0, 0, t)\},$$

tree space is just $T = \{\tau; t \geq 0\}$, and

$$p(\tau) = \exp(-t)$$

is our prior. This model can be extended to the case of $n > 2$ leaves. The resulting distribution on the space of n -leaf trees is called the Kingman coalescent.

Example 8.2. Subjective priors are widely used to weight in favor of trees thought to be *a priori* plausible. For example, we may penalize branch lengths exponentially, with mean branch length $1/\lambda$. If $|\tau|$ is the total tree length (ie, $|\tau| = \sum_{\langle a,b \rangle \in E} t_{\langle a,b \rangle}$), then

$$p(\tau) = n\lambda^{n-1} \exp(-\lambda|\tau|).$$

This is the probability density for a species tree τ generated by a process in which each species branches independently at rate λ . The process starts from the branching event at the root, and is stopped at the first time the tree has $n + 1$ leaves (so

the resulting tree has n leaves). [On a two leaf tree this distribution is essentially the same as the coalescent example above - they differ for 3 or more leaves]. Unlike Example 8.1, this is not intended to represent reproducible biological reality.

Example 8.3. Non-informative priors. This appears to be the simplest case: assume $p(\tau) \propto c$ for all trees $\tau \in T$ and c a constant. However, there is a problem. For the posterior probability $P(\mathbb{T} \in S|D)$ to make sense, $h(\tau|D)$ must exist as a distribution. In particular $\int_T h d\tau$ must equal one, and of course, the integral must be defined. This will be OK if $P(D)$ is finite. But under the constant prior, $P(D) = c \int_T P(D|\tau) d\tau$, and the integral is not finite. If we fix a tree topology, and let the branch lengths become large, $P(D|\tau)$ tends to a positive constant, so $\int_T P(D|\tau) d\tau$ cannot be finite.

Exercise 8.4. Show that $\lim_{\rho \rightarrow \infty} P(D|\rho\tau) = \prod_{a=1}^N \prod_{s=1}^N \pi_{x_{a,s}}$.

In order to overcome this we may set a bound on branch lengths. We may impose some prior information asserting that branch lengths are shorter than some maximum value t^* (which may be very conservative). Our new tree space is $T^* = \{\tau \in T : \text{for each } \langle a, b \rangle \in E, t_{\langle a, b \rangle} \leq t^*\}$. Since T^* is closed and bounded, and $P(D|\tau)$ is bounded on T^* , (in fact $P(D|\tau) \leq 1$), the integral $P(D)$ must be finite. Priors of this type are in widespread use.

Unfortunately, problems remain. We now have a well defined prior, $p(\tau) = c$ defined on a closed and bounded space. However, the marginal distribution of the root time t_r turns out to be proportion to t_r^{n-2} on an n -leaf tree.

Exercise 8.5. prove this for a rooted 3 leaf tree. Take leaf nodes 1,2 and 3 at time zero, and root node $r=5$ at time t_5 , fix a topology with node 4 a child of node 5 and consider $\int_0^{t_5} c dt_4$.

Our apparently uniform prior represents a state of knowledge which favors a tree with depth $2t_r$ over a tree with depth t_r by a factor $2^{(n-2)}$. This may not be what we intended! The moral of the story is this: if we wish to use non-informative priors, we should write down a density which we believe represents a state of ignorance, and check, for example by sampling the distribution, whether typical draws are representative of the range of trees we had in mind.

8.4. Inference. Having written down a prior density on tree space, representing our state of knowledge before the data arrives, we return to the data, and the phylogenetic features of scientific interest. The purpose of the inference is to determine whether the unknown true phylogeny possess some particular property (ie, is $\tau \in S$, as we set it out in Section 8.2).

Example 8.6. Referring to Example 8.1, suppose $t \sim Exp(1)$. Suppose two sequences $x_1 = (01)$ and $x_2 = (00)$ (ie, the two site patterns are $D_1 = (00)$ and $D_2 = (10)$) are generated at the leaves by a substitution process with the same overall rate μ substitutions (per site) per N_e generations on each branch, and rate matrix

$$Q = \begin{pmatrix} -1 & 1 \\ 1 & -1 \end{pmatrix}.$$

Write down an expression for the probability that the root age in the unknown true tree exceeds 1 (in units of N_e generations) in two cases: (i) in the absence of data and (ii) conditional on the data.

(i) before the data arrives, all we know is that the tree is a draw from the coalescent distribution on two leaf trees, which we saw was $Exp(1)$. The probability that the tree height exceeds 1 is $P(t > 1) = \int_S p(\tau) d\tau$ with $S = \{\tau \in T; t \geq 1\}$, so $P(t > 1) = \int_1^\infty e^{-t} dt$, that is, $P(t > 1) = e^{-1}$.

(ii) this time we have data. The probability that the tree height exceeds 1 is $P(t > 1) = \int_S h(\tau|D) d\tau$ with $h = P(D|\tau)p(\tau)/P(D)$ and $p(\tau) = \exp(-t)$. It remains to compute

$$P(D|\tau) = P(00|t)P(10|t).$$

The transition probability is $P(t) = \exp(Q\mu t)$, with t in units of N_e generations, Q normalized to one substitution per unit time, and μ converting the units of t to those of Q . Evaluating the matrix exponential we find

$$P(t) = \frac{1}{2} \begin{pmatrix} 1 + \exp(-2\mu t) & 1 - \exp(-2\mu t) \\ 1 - \exp(-2\mu t) & 1 + \exp(-2\mu t) \end{pmatrix}$$

Now

$$\begin{aligned} P(00|t) &= \pi_0[P_{00}(t)]^2 + \pi_1[P_{10}(t)]^2 \\ P(10|t) &= \pi_0[P_{01}(t)][P_{00}(t)] + \pi_1[P_{11}(t)][P_{10}(t)] \end{aligned}$$

so expanding and simplifying,

$$P(D|\tau) = \frac{1}{16}(1 - e^{-8\mu t}).$$

The normalizing constant, $P(D) = (1/16) \int_0^\infty (1 - e^{-8\mu t}) e^{-t} dt$ is $P(D) = \mu/(16\mu + 2)$, and

$$\begin{aligned} P(t > 1) &= \frac{(8\mu + 1)}{8\mu} \int_1^\infty (1 - \exp(-8\mu t)) e^{-t} dt \\ &= \frac{8\mu + 1 - \exp(-8\mu)}{8\mu} \end{aligned}$$

Now, our prior expectation is for a t -value around one. If $\mu \gg 1$ then $P_{i,j}(t) \simeq \pi_j$, as the base substitution process reaches equilibrium very rapidly. The data on the two leaves are then close to being independent draws from the equilibrium base frequencies, and the probability to get the data is insensitive to the root time (unless the root time $t \ll 1$). The data brings little information with it, and our state of knowledge is unchanged (or little changed) by the arrival of the data. Looking above $h(\tau|D)$ coincides with $p(\tau)$ when $\mu \rightarrow \infty$ for any $t > 0$.

9. SAMPLE-BASED INFERENCE OF PHYLOGENY

See Chapter 6 and in particular Section 6.14 of [6] for (non-examinable) technical background for this section. Both [2] (Section 8.4) and [5] (Chapters 18 and 27) cover this topic, but are somewhat off-topic for our purpose.

In the last section we described a way of estimating phylogenies from sequence data, and quantifying the confidence we place in any particular hypothesis about the properties of the unknown true phylogeny.

The problem we face in practice is quite simply an intractable integration. The probability $\Pr(\mathbb{T} \in S|D)$, that the unknown true phylogeny possesses property- S ,

is given in terms of an integral, over all the trees in tree space,

$$\begin{aligned}
 P(\mathbb{T} \in S|D) &= \int_S h(\tau|D)d\tau \\
 (9.1) \qquad &= \frac{1}{P(D)} \int_T P(D|\tau)p(\tau)\mathbb{I}_{\tau \in S}d\tau \\
 &= \mathbb{E}(\mathbb{I}_{\tau \in S}),
 \end{aligned}$$

where $\mathbb{I}_{\tau \in S}$ is a function, called the indicator function, which equals one if $\tau \in S$ and is otherwise zero. In computing the maximum parsimony and maximum likelihood trees we had to search over the space T of all trees for the best tree. We have now to average the function $\mathbb{I}_{\tau \in S}$ over the same space. There are, as we have discussed, many many distinct topologies, and for each topology we must integrate over all allowed branch lengths. We were able to do this integration by hand for the two leaf tree above, because there was just one distinct topology.

However, there is a well known strategy for estimating the values of high-dimensional integrals, the method of Monte Carlo integration. Suppose $f(x)$ is a probability density on some space X , and $g : X \rightarrow R$ is a real-valued function on X . In order to compute the integral $\mathbb{E}(g(x)) = \int_X f(x)g(x)dx$ we may draw M samples, x_1, x_2, \dots, x_M distributed at random according to f , compute $\bar{g} = \frac{1}{M} \sum_{m=1}^M g(x_m)$, and estimate $\mathbb{E}(g(x)) \simeq \bar{g}$. The difference $\mathbb{E}(g(x)) - \bar{g}$ is a random variable, since \bar{g} is a function of the random draws x_m , $m = 1..M$. Under mild conditions [6] this difference is asymptotically normally distributed, with mean zero, and a standard deviation σ_g/\sqrt{M} which goes to zero as the number of samples, M , gets large,

$$(9.2) \qquad \sqrt{M}(\mathbb{E}(g(x)) - \bar{g}) \rightsquigarrow N(0, \sigma_g^2).$$

Our strategy to compute an estimate, \hat{p} say, for the integral $P(\mathbb{T} \in S|D)$ in Equation 9.1 will be to draw sample phylogenies $\tau_1, \tau_2, \dots, \tau_M$ distributed at random according to $h(\tau|D)$, and compute

$$\begin{aligned}
 \hat{p} &= \frac{1}{M} \sum_{m=1}^M \mathbb{I}_{\tau_m \in S} \\
 &= \text{the proportion of the } M \text{ sampled} \\
 &\quad \text{phylogenies possessing property-}S \text{ .}
 \end{aligned}$$

We expect \hat{p} to converge to $P(\mathbb{T} \in S|D)$ when the sample size M is large.

The problem then is to draw samples τ_m , $m = 1, 2, \dots, M$ distributed at random according to $h(\tau|D)$. Software, described in the next section, is available which carries out this task, for some combinations $P(D|\tau)p(\tau)$ of the base substitution models $P(D|\tau)$ and priors $p(\tau)$ which commonly arise in phylogenetics. The software dates for the most part from the last 5 years. Various methods are used to gather the sample phylogenies, and all have their limitations. One popular method is called Markov chain Monte Carlo (MCMC): it is Monte Carlo, as above, but the samples $\tau_1, \tau_2, \dots, \tau_M$ are created using a Markov chain. Under certain conditions, this method fails. We need to know enough about how it works to diagnose failure from its output.

A starting state τ_0 is chosen. This phylogeny is chosen arbitrarily, and need not be at all representative of the the data or prior. The MCMC then generates a sequence $\tau_1, \tau_2, \dots, \tau_M$ by simulating a guided ‘‘random walk’’ in tree-space. At step

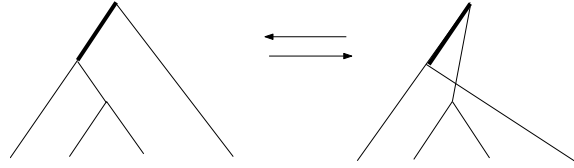
m of the random walk, the current tree is τ_m . The next tree τ_{m+1} in the sequence is generated using a two step algorithm called the Metropolis algorithm.

- (1) Generate a candidate state τ' by making a random modification to the current state, τ_m . The random operation which chooses τ' by modifying τ has a symmetry - the probability to generate τ' from τ_m is the same as the probability to generate τ_m from τ' .
- (2) (a) If $h(\tau'|D) \geq h(\tau_m|D)$, set $\tau_{m+1} = \tau'$ (ie, accept the candidate).
 (b) If $h(\tau'|D) < h(\tau_m|D)$, toss a coin that has probability

$$h(\tau'|D)/h(\tau_m|D)$$

of coming up heads. If it comes up heads, set $\tau_{m+1} = \tau'$. If it comes up tails, set $\tau_{m+1} = \tau_m$ (ie, retain the current state).

Example 9.1. for the candidate generation process at step 1, we have some freedom. One possibility is the so-called branch interchange: choose two branches uniformly at random; each branch has a child node and a parent node; exchange the parent nodes. If τ and τ' are two trees connected by this operation, then the probability to propose τ' from τ is equal to the probability to propose τ from τ' .



Example 9.2. another candidate generation operation is needed to change branch length: choose a branch $\langle a, b \rangle$ uniformly at random from the branches of the tree. Alter the length $t_{\langle a, b \rangle}$ of this branch by adding a random number ϵ drawn uniformly at random from the interval $[-1, 1]$. The candidate tree τ' has a branch of length $t_{\langle a, b \rangle} + \epsilon$ between nodes a and b . Notice that this may lead to an illegal tree (for example, a negative branch length). We simply assign $h(\tau'|D) = 0$ for trees $\tau \notin T$.

Exercise 9.3. (not examinable) suppose $q(\tau'|\tau_m)$ is the probability density to generate candidate τ' from τ_m at step 1. Show that the transition probability density $P(\tau'|\tau_m)$ for the transition from tree τ_m to tree τ' is $q(\tau'|\tau_m)$ if $h(\tau'|D) \geq h(\tau_m|D)$ and $q(\tau'|\tau_m)h(\tau'|D)/h(\tau_m|D)$. Verify that the reversibility condition,

$$h(\tau_m|D)P(\tau'|\tau_m) = h(\tau'|D)P(\tau_m|\tau')$$

and the stationarity condition,

$$\int_T h(\tau_m|D)P(\tau'|\tau_m)d\tau_m = h(\tau'|D)$$

are both satisfied. Interpret.

Exercise 9.4. show that, if we use the Metropolis algorithm, we do not need to know the value of $P(D)$ in order to simulate τ_{m+1} given τ_m , that is, it is sufficient to have $h(\tau|D)$ defined up to a multiplicative constant.

The Metropolis algorithm moves towards better trees when it finds them, and towards worse trees occasionally, but with a probability that gets small as the posterior probability of the candidate tree gets small relative to the current tree. Recall our discussion of the Markov chain on base characters - as time goes on the

chain “forgets” its start state and the probability to find the chain in any particular state is given by the equilibrium base frequency. It is the same story here. For large M the Markov chain $\tau_m, m = 0, 1, 2, \dots, M$ defined by the Metropolis algorithm tends to an equilibrium distribution: that equilibrium distribution is exactly $h(\tau|D)$ (you prove $h(\tau|D)$ is an equilibrium distribution if you complete Exercise 9.3, to show it is the only equilibrium distribution, you have to establish an additional property, irreducibility).

Theory tells us that if we run the MCMC long enough (large M) then, under mild conditions [6] the distribution of our samples τ_m converges to $h(\tau_m|D)$, and the sequence $\tau_1, \tau_2, \dots, \tau_m$ can be used to make estimates, as in Equation 9.2 (which is the real point of the exercise). But how long is long enough? There is no widely useful sufficient condition we can test in order to establish convergence to equilibrium. MCMC is a tantalizingly useful method, but we need to be aware of this fundamental weakness.

10. SOFTWARE FOR SAMPLE-BASED INFERENCE OF PHYLOGENY

See <http://evolution.genetics.washington.edu/phylip/software.html> for a useful overview of available software, including packages mentioned in [5].

In this and the next section we will illustrate the use of one package for sample-based inference of phylogeny: MrBayes. This package, written by John Huelsenbeck, Bret Larget, Paul van der Mark and Fredrik Ronquist is one of several widely used packages for sample based inference of Phylogeny. Of the many other packages implementing inference of phylogeny, Paup 4.0* (<http://paup.csit.fsu.edu/>) is one of the more widely used. It implements maximum likelihood and parsimony methods for tree-estimation. A number of related visualization tools are linked from the same site. Ziheng Yang’s PAML package is another ML tool, particularly strong on codon models of the kind discussed in Section 7.3 and in [8]. There is a separate group of packages designed specifically for population genetic inference (they estimate parameters of the coalescent model we mentioned briefly in Section 8.4). BEAST, <http://evolve.zoo.ox.ac.uk/beast/>, a local product, written by Andrew Rambaut and Alexei Drummond, is one such.

Returning to MrBayes, I recommend you download and install this package, and have a trial run. It is freely available from

<http://mrbayes.csit.fsu.edu/index.php>.

Installation is straightforward. There links to documentation at the same site.

Some background: MrBayes implements Bayesian inference and MCMC in order to estimate phylogenies, and answer questions of the kind we posed: what is the probability that the unknown true tree on which the sequence data was generated possessed such-and-such a property? The sequence data is presented to MrBayes in a file format called a nexus-format. The data in the file `synthdata.nex`, bundled with these notes, has this format:

```
#NEXUS

begin data;
  dimensions ntax=22 nchar=400;
```

```

format datatype=dna missing=-;
matrix
1   CAATACTTGGACATTCT... +370 sites+   ...TCATATATTAGCA
2   TAATACTCAAGCATTCT...               ...TCGTCTATCGATA
.
.   +18 sequences+
.
21  TAATACTCAAGCATTCT...               ...TCGTCTATCGATA
22  CGATACTTGAACATTCT...               ...TTATATATTGACA
;
end;

```

There are three stages to a MrBayes analysis.

(1) First we add some commands to the nexus-file telling MrBayes what we want it to do. We have to tell MrBayes which parameters need to be estimated (the tree and rate parameters, or just the tree?), we must give a prior for the tree and priors for any other parameters of interest, as well as telling MrBayes what substitution model to use, and the number of MCMC samples for MrBayes to simulate. If R are the relative rates and Π is a diagonal matrix of equilibrium base frequencies, so that $Q = R\Pi$, then MrBayes can generate samples from the joint posterior distribution

$$h(\tau, R, \pi|D) = P(D|\tau, R, \pi) \frac{p(\tau, R, \pi)}{P(D)}.$$

If we know the rates and base frequencies, there is no need to get MrBayes to estimate them, and we may instruct MrBayes to generate samples from the posterior distribution for the tree alone,

$$h(\tau|D, Q) = P(D|\tau, Q) \frac{p(\tau)}{P(D)}.$$

(2) Secondly, we execute the nexus file from the MrBayes command-line

```
Mr Bayes > execute synthfile.nex
```

MrBayes runs two independent MCMC simulations, writes sample phylogenies $\tau_0^{(1)}, \tau_1^{(1)}, \dots, \tau_M^{(1)}$ from one MCMC run to a file called `synthdata.nex.run1.t` and sample phylogenies $\tau_0^{(2)}, \tau_1^{(2)}, \dots, \tau_M^{(2)}$ from the other MCMC run to a file called `synthdata.nex.run2.t` as well as writing corresponding sample parameter values (estimates of the parameters of a GTR substitution model) to two files called `synthdata.nex.run1.p` and `synthdata.nex.run2.p`. The purpose of all this duplication is to check that average results computed from the MCMC simulation are independent of the start state - *ie* equal between runs.

(3) Thirdly, we analyze the output samples: we run `sump` and `sumt`:

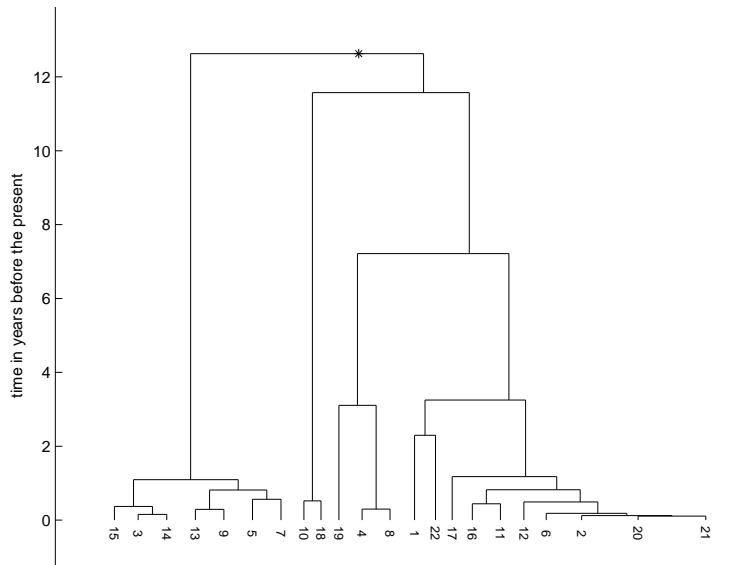
```
Mr Bayes > sump
```

```
Mr Bayes > sumt
```

MrBayes computes summaries of the output and writes them to the MrBayes window, as well as writing files `synthdata.nex.parts` (estimated posterior probabilities for clades), `synthdata.nex.trprobs` (estimated posterior probabilities for trees), and `synthdata.nex.con` (a consensus tree - something like a “best” single tree).

11. MRBAYES: A SHORT EXAMPLE

The alignment in `synthdata.nex` has 22 taxa and 400 sites. It was generated by taking this fixed tree



with total tree length 6.018×10^6 years, drawing 400 independent characters for the root (using equilibrium base frequency $\pi = (\pi_A = 0.31, \pi_C = 0.19, \pi_G = 0.2, \pi_T = 0.3)$), and then simulating the GTR substitution process (with relative rates ($r_{AC} = 1.1, r_{AG} = 45, r_{AT} = 0.02, r_{CG} = 0.55, r_{CT} = 27, r_{GT} = 1$) independently at each site along the branches to the leaves. The rate matrix Q was normalized to unit rate, and then multiplied by an overall rate $\mu = 1 \times 10^{-6}$ [subs/site/year] (these numbers happen to be appropriate for certain species of penguin). It follows that the data is 'perfectly' clock-like and GTR - there is no model misspecification error. We use synthetic data here in order to check that we understand what the program is doing. We will analyze the data in `synthdata.nex`, treating this problem exactly as we would if the true tree (and rate parameters) were unknown. However, we will have the advantage of being able to compare our results with known true phylogeny. We have time to make just one analysis of this data. In general we would make many analyses, varying priors to see how sensitive our estimates were to different prior states of knowledge. The idea then is as before, to take this sequence data and estimate a phylogeny or set of phylogenies.

We will impose a GTR base substitution model, and assume the parameters

$$(r_{AC}, r_{AG}, r_{AT}, r_{CG}, r_{CT}, r_{GT})$$

of the relative rate matrix R and equilibrium base frequencies π are known. We will assume the unknown true tree is clock-like, with leaves fixed at $t = 0$. MrBayes will estimate a rooted tree with branch lengths in substitutions. We will have to use the overall rate μ to convert back to years. We will impose something close to a uniform prior on trees: $p(\tau) \propto \exp(-t_r(\tau))$ where $t_r(\tau)$ is the root time of tree τ measured in substitutions (the exponential penalty is added to ensure the

posterior is finitely integrable - we discussed imposing a cut off in Section 8.3). One substitution is rather deep (pairs of sequences separated by the root will be at substitution-saturation) so this is approximately equal to the constant prior $p(\tau) = c$ for all trees with $t_r \ll 1$.

In order to achieve these results we add the block

```
begin mrbayes;
  lset nst=6 nucmodel=4by4 rates=equal;
  prset brlenspr=clock:uniform treeheightpr = Exponential(1.0)
        statefreqpr=fixed(0.31,0.19,0.2,0.3)
        revmatpr=fixed(1.1,45,0.02,0.55,27,1);
  mcmc ngen=100000 nchains=1 printfreq=1000 samplefreq=100;
  mcmc savebrlens=yes;
  mcmc;
end;
```

to the end of the nexus file `synthdata.nex`. The `lset` line fixes the substitution model. `nst=6` is GTR, `nucmodel=4by4` is base substitution (rather than codon or amino-acid), `rates=equal` imposes the assumption that the rate parameters of Q are equal at all sites (rather than varying across codon position, as in Section 7.1). The `prset` line fixes priors for parameters and tree. The tree is clocklike, with prior $p(\tau) = \exp(-t_r)$ determined by `brlenspr=clock:uniform treeheightpr = Exponential(1.0)`. The relative rates and equilibrium base frequencies are fixed, by `statefreqpr=fixed(...)` `revmatpr=fixed(...)`. The Markov chain will run 100000 steps, and gather $M = 1000$ samples (since MrBayes will gather every 100th sample - it is convenient to represent the distribution h with as few samples as possible, so we sub-sample, since consecutive samples are excessively correlated). These numbers were chosen by making some pilot runs - more on this below.

Executing `synthdata.nex` and running `sump` and `sumt` we get raw output

```
Chain results:
  1 -- -11375.338 * -11328.781
 1000 -- -5708.314 * -4989.389 -- 0:01:39

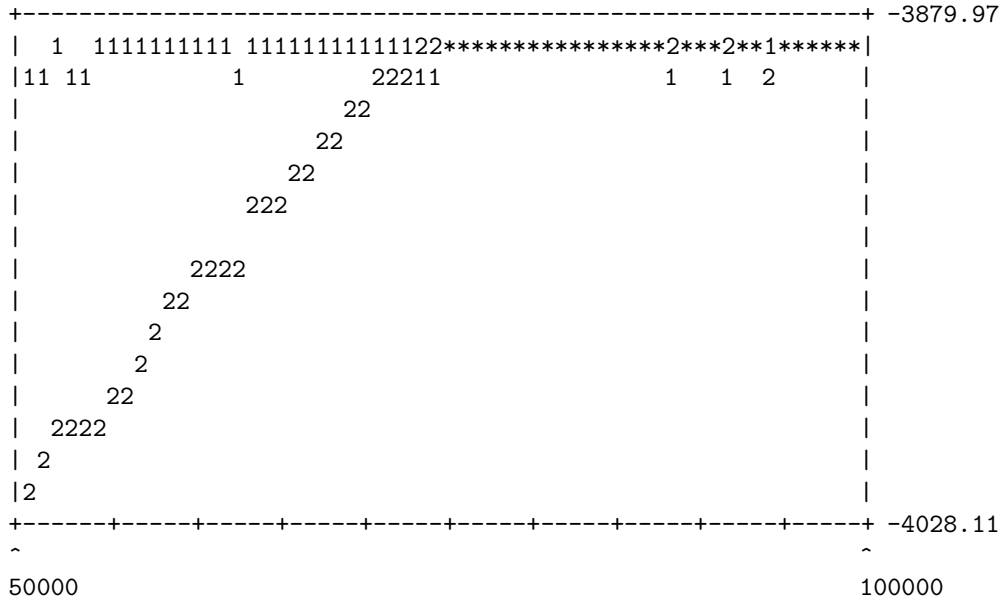
Average standard deviation of split frequencies: 0.257701
.
.
.
99000 -- -3882.319 * -3883.005 -- 0:00:00

Average standard deviation of split frequencies: 0.033411
100000 -- -3892.052 * -3887.690 -- 0:00:00

Average standard deviation of split frequencies: 0.032628

Continue with analysis? (yes/no): no
MrBayes >
```

MrBayes is, by default, running two MCMC chains, from different start states. In order to check for convergence to equilibrium (*ie*, in order to answer the question,

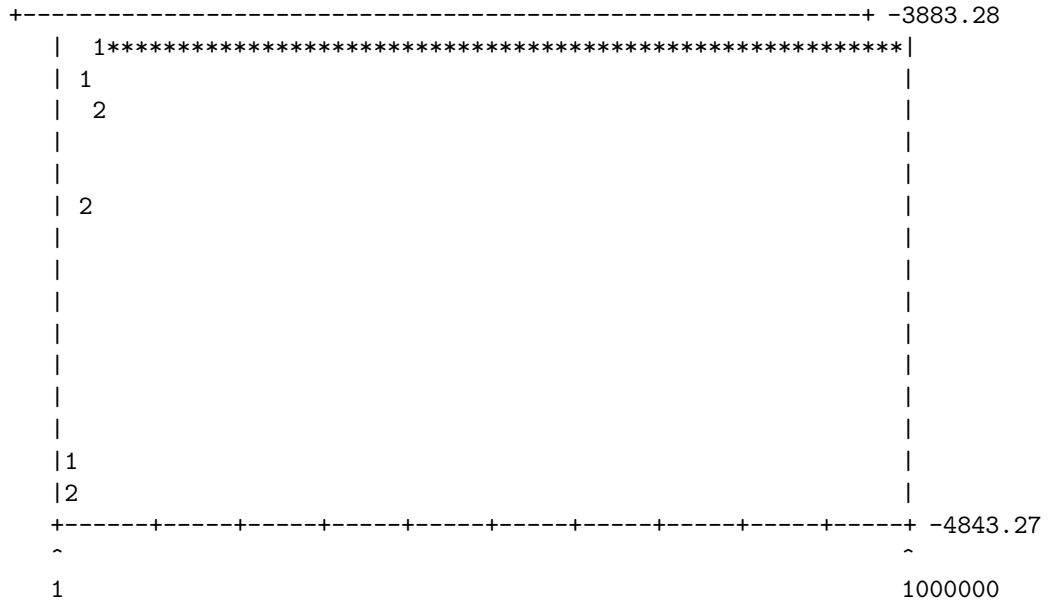


Samples from the 2nd chain show a clear trend towards better trees till late in the run. Most of our sampled trees are not representative of target distribution $h(\tau|D)$, but are instead biased by the initial state.

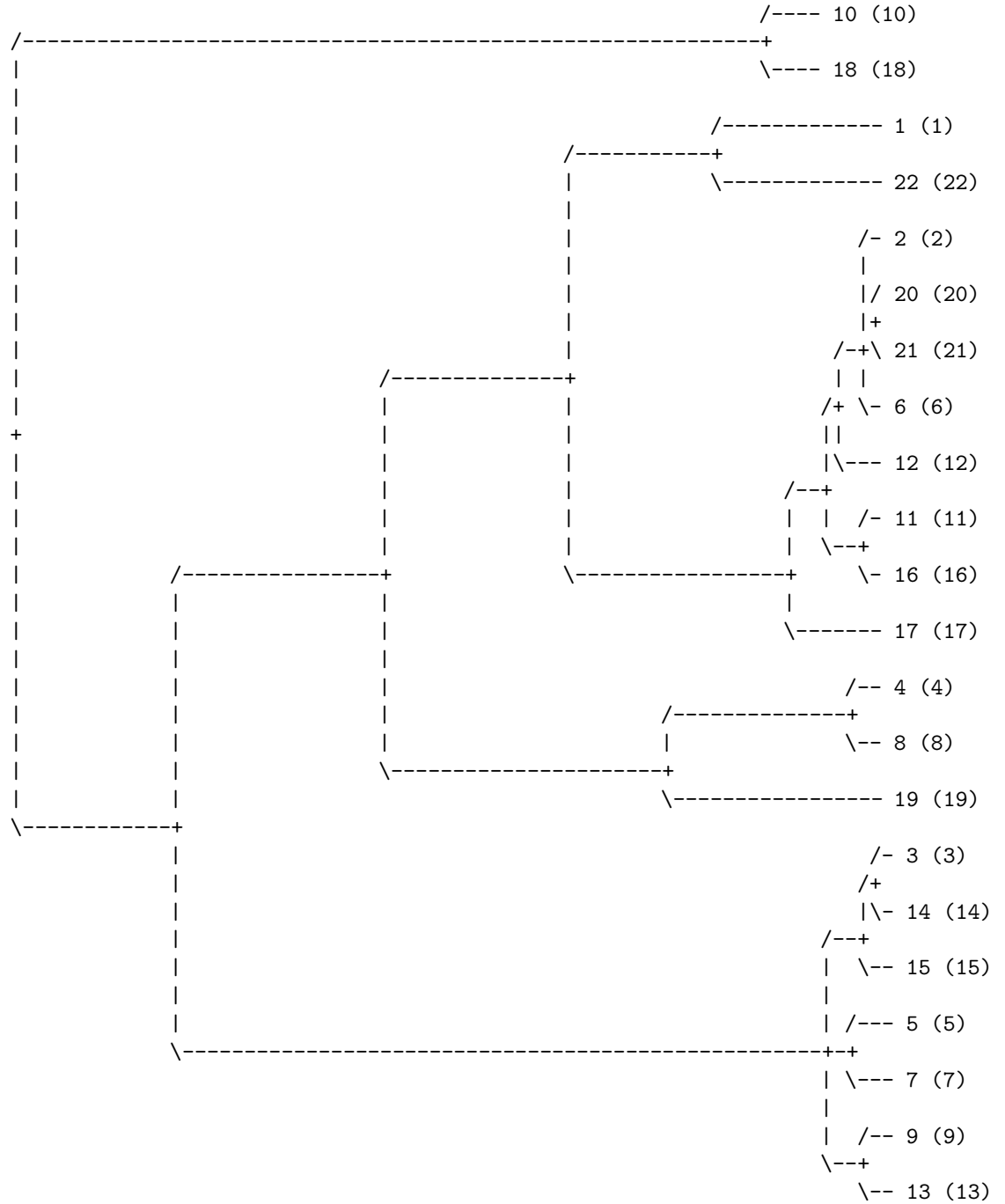
We therefore repeat the whole exercise, but make a run of 10 times the length.

```
mcmcp ngen=100000 nchains=1 printfreq=1000 samplefreq=100;
```

This is in effect giving the chain time to search through tree space, find the plausible set, and then explore that set. The trending section is now a much smaller portion of the entire run



Phylogram:



-----| 0.200 expected changes per site

-----|-----|-----|-----|-----|-----|-----|

Comparing with the true-tree drawn at the start of this section, there is agreement here with essentially the entire tree. We have seen that there is real uncertainty in the location of certain clades, so we shouldn't put too much store in a single tree.

A multifurcation occurs in the consensus tree, at the root of the clade (3,14,15,5,7,9,13).

ID	Partition	#obs	Prob.	Stdev(s)
...				
34	-- ..*.*.*.*...***.....	1998	0.9980	0.00000
42	--*.*.*.....	976	0.4875	0.01978
44	-- ..*.*.*.....**.....	606	0.3027	0.00141
46	-- ..*.....*...***.....	413	0.2063	0.01766

The gross clade 34:..*.*.*.*...***..... is strongly supported by the data, but the branching at the top of this clade is not resolved, as it occurs over a time-scale which is short compared to the substitution rate. Referring to the true tree, the branch at the top of this clade is about 2×10^4 years long. At $\mu = 10^{-6}$ subs/site/year and 400 sites we might expect the sequences (5, 7, 9, 13) to share around 8 substitutions lacking in (3, 14, 15). However the actual number achieved is random and may be less. We see the correct clade structure is supported, at 48% posterior probability, just below the threshold used to draw the consensus tree.

Note also the time scale bar below the phylogram. The total tree depth is estimated at around 1.1 substitutions. This agrees well with the truth (12×10^5 years at 10^{-6} subs/yr). However, sequences connecting across the root will be at substitution-saturation. They will be very nearly as distant from one another as two sequences drawn from the equilibrium base frequency. It is not then surprising that location of the (10,18) clade is uncertain: the substitutions it shares with (1, 2, 4, 6, 8, 11, 12, 16, 17, 19 – 22) could easily be erased on the long branch leading to the (10,18) taxa.

Finally, the file `synthdata.nex.trprobs` contains a list of distinct tree-topologies, with an estimate of their posterior probability. Here are the top three trees, reformatted to fit:

```
label post.pr tree
tree_1 p = 0.17 (10,18),(((19,(4,8)),((1,22),(17,((12,(6,(2,(20,21))))),(11,16))))),
              (((9,13),(5,7)),(15,(3,14))))
tree_2 p = 0.11 (10,18),(((19,(4,8)),((1,22),(17,((12,((2,6),(20,21))))),(11,16))))),
              (((9,13),(5,7)),(15,(3,14))))
tree_3 p = 0.10 (10,18),(((19,(4,8)),((1,22),(17,((12,(6,(2,(20,21))))),(11,16))))),
              (((15,(3,14)),(5,7)),(9,13)))
...
```

Because there are so many tree-topologies, these posterior probabilities are typically very small. It is salutary to see where the true tree comes in this list: it isn't there. There are simply too many trees; the posterior probability for the true tree is a very small number, and it would take many, many samples to estimate such a small number accurately. It would be unwise to report the most probable tree (`tree_1`) - it has less than one chance in 5 of being correct.

To conclude then, this example has shown that we can recover many features of the phylogeny from sequence data. Rather than reporting a single tree, we report the

posterior probability for those features to be present. This is just one analysis. One needs to check that results are reproducible. We should repeat the entire analysis, with a different random number seed, and check that any important features of our results are unchanged. The discussion of MCMC convergence to equilibrium given above is cursory. The latest incarnation of the MrBayes manual contains much good sense on this topic. We should make further studies, with different priors, to check that results are insensitive to any features of the prior about which we are not confident. For example, how does the estimated root time depend on our prior choice $t_r \sim \text{Exp}(1)$? We could repeat the analysis with $t_r \sim \text{Exp}(2)$ and find out.

REFERENCES

1. A. J. Drummond, G. K. Nicholls, A. G. Rodrigo, and W. Solomon, *Estimating mutation parameters, population history and genealogy simultaneously from temporally spaced sequence data*, *Genetics* **161** (2002), 1307–1320.
2. R. Durbin, S. Eddy, A. Krogh, and G. Mitchison, *Biological sequence analysis: probabilistic models of proteins and nucleic acids*, Cambridge, 1998.
3. J. Felsenstein, *Syst. Zool.* **27** (1978), 401–410.
4. J. Felsenstein, *Evolutionary trees from DNA sequences: a maximum likelihood approach*, *J. Mol. Evol.* **17** (1981), 368–376.
5. ———, *Inferring phylogenies*, Sinauer, 2004.
6. G. Grimmet and D. Stirzaker, *Probability and random processes*, OUP, 2001.
7. D.M. Hillis, C. Moritz, and B.K. Mable, *Molecular systematics*, 2nd ed., Sinauer Associates, Sunderland, Mass, 1996.
8. Z. Yang, R. Nielsen, N. Goldman, and A. M. Krabbe-Pedersen, *Codon-substitution models for heterogeneous selection pressure at amino acid sites*, *Genetics* **155** (2000), 431449.

STATISTICS DEPARTMENT

E-mail address: nicholls@stats.ox.ac.uk