Evolutionary Genomics: Phylogenetics

Nicolas Salamin

nicolas.salamin@unil.ch – http://www.unil.ch/phylo

UNIL | Université de Lausanne

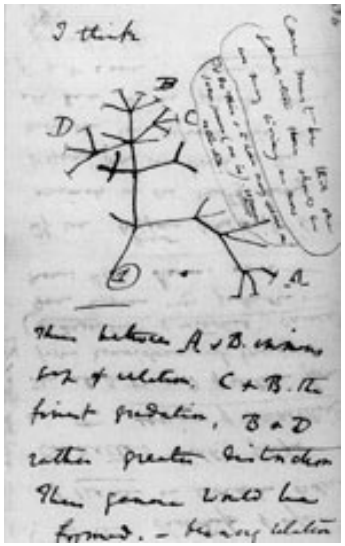
Département de biologie computationnelle

February 27th, 2020

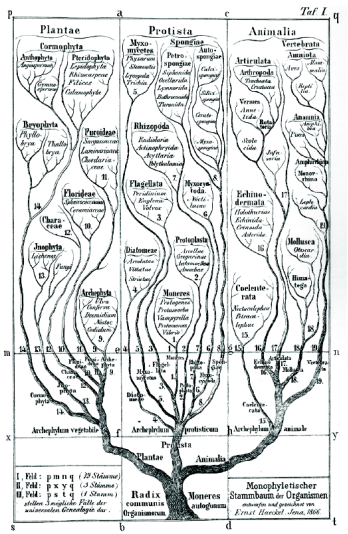
Course material

https://wp.unil.ch/phylo/teaching/phylogenomics/

Tree of Life and Evolution



Darwin, 1857



Haekel, 1866

Phylogeny is the evolutionary history of living and extinct organisms.

Some key dates:

1964 Cavalli-Sforza and Edwards introduced parsimony and likelihood

1966 Hennig and the theory of cladistics

1977 Fitch used parsimony on DNA sequences

1978 Felsenstein worked out maximum likelihood method

1996 Rannala and Yang proposed Bayesian inference

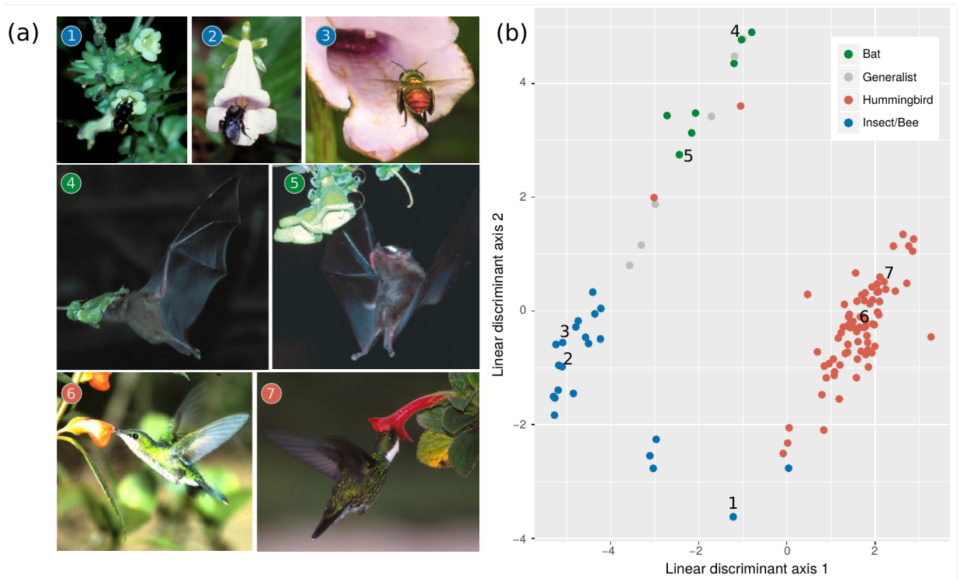
since 2000 Computational phylogenetics

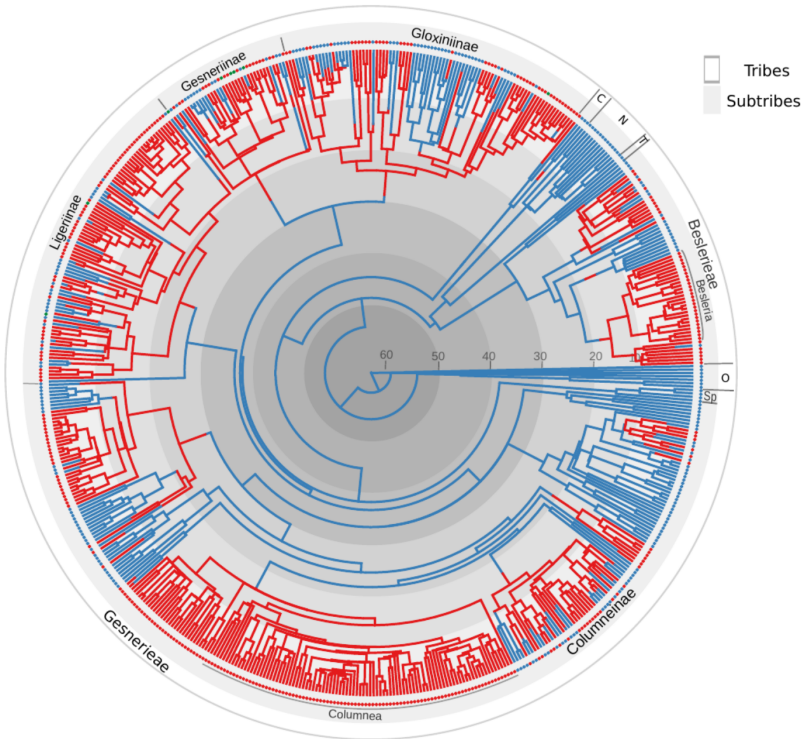
New uses for phylogenetics

Beside their use in systematics, trees are tools to

- study macroevolutionary processes
- detect evolution of genes and their function
- assess conservation issues
- drug design
- epidemiology
- evolution of languages

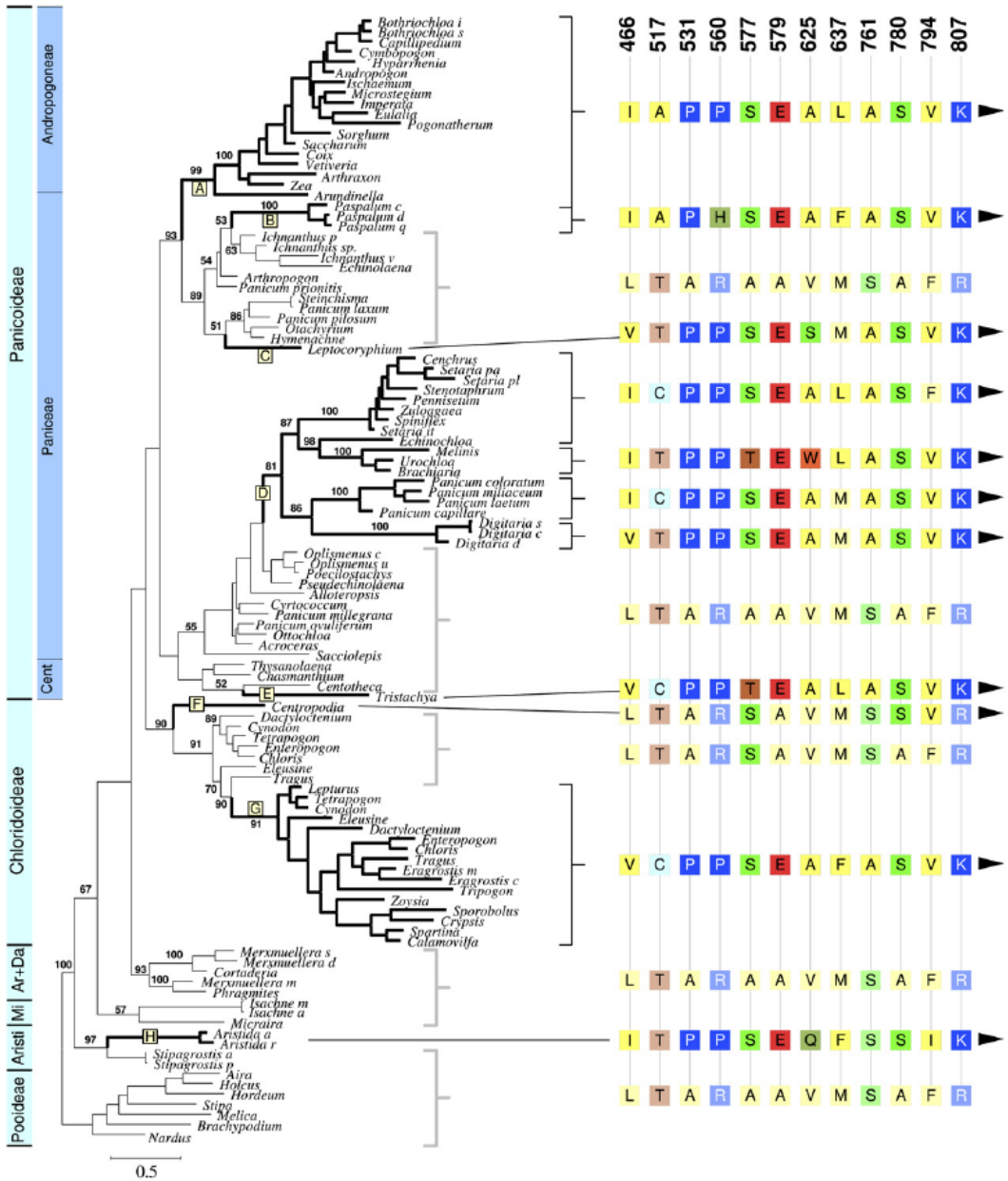
Macroevolution





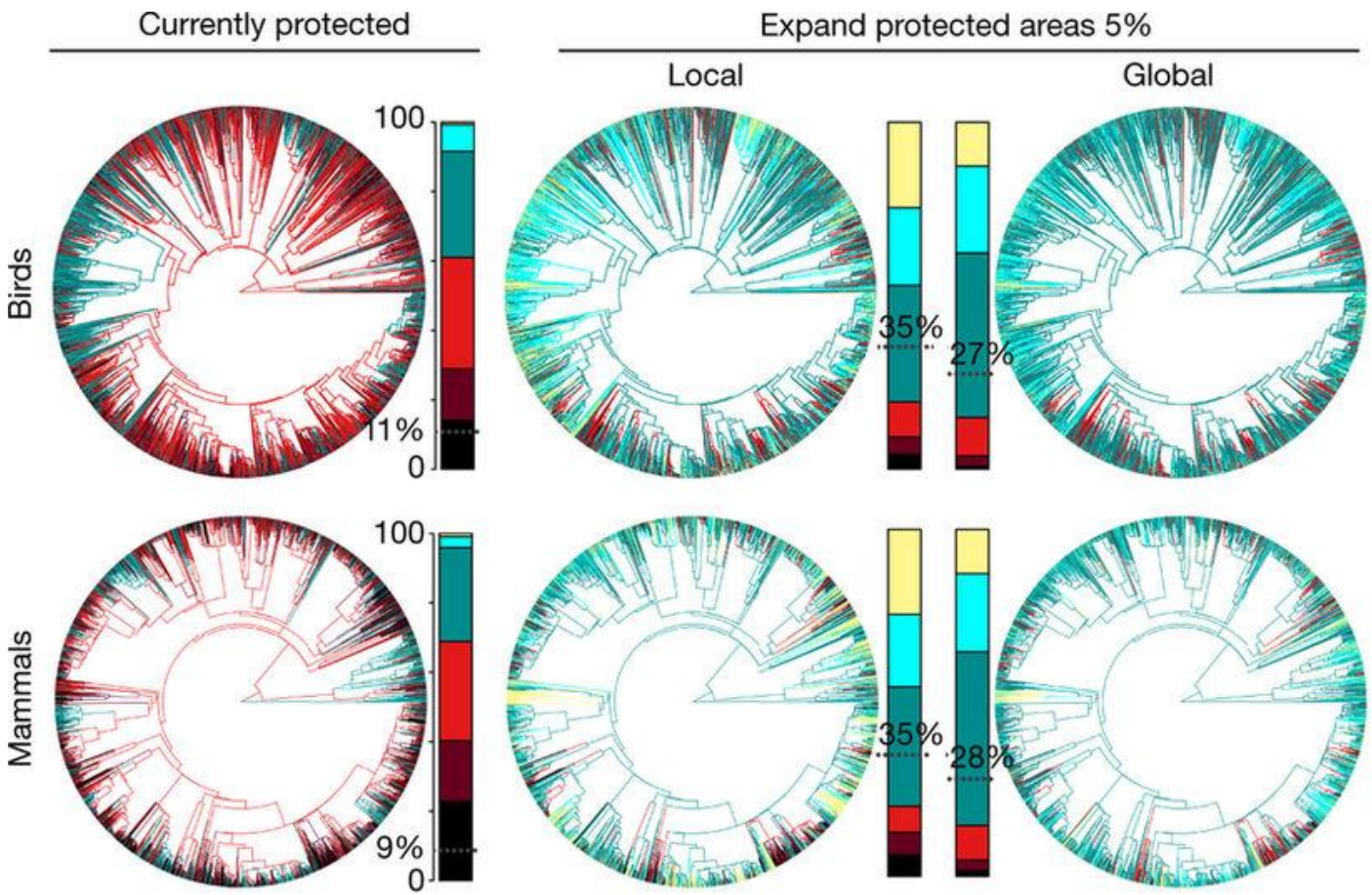
Serrano-Serrano et al. (2017) Proc. R. Soc. B

Genes evolution



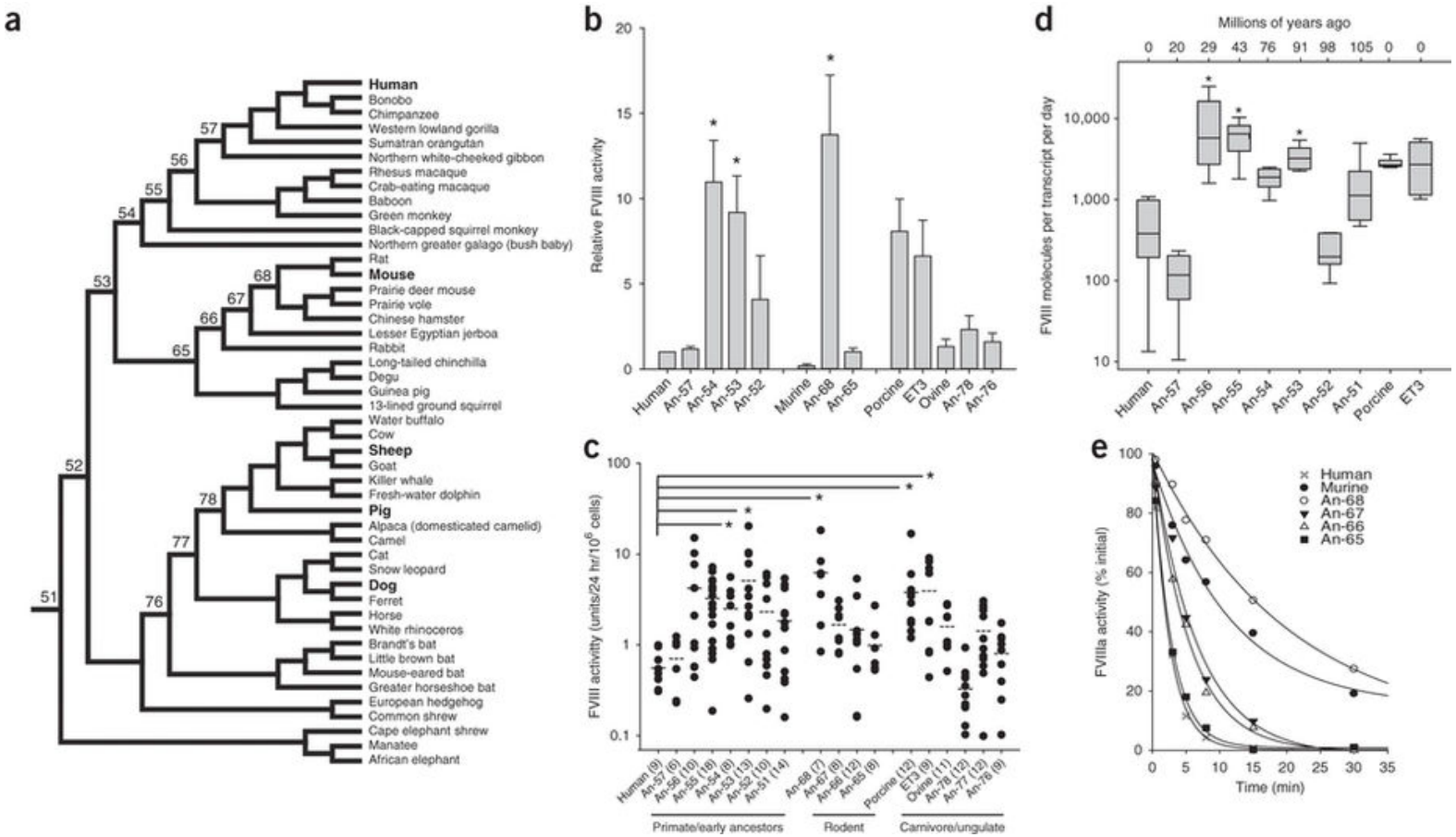
Conservation issues

Distribution across lineages of extend of protected areas



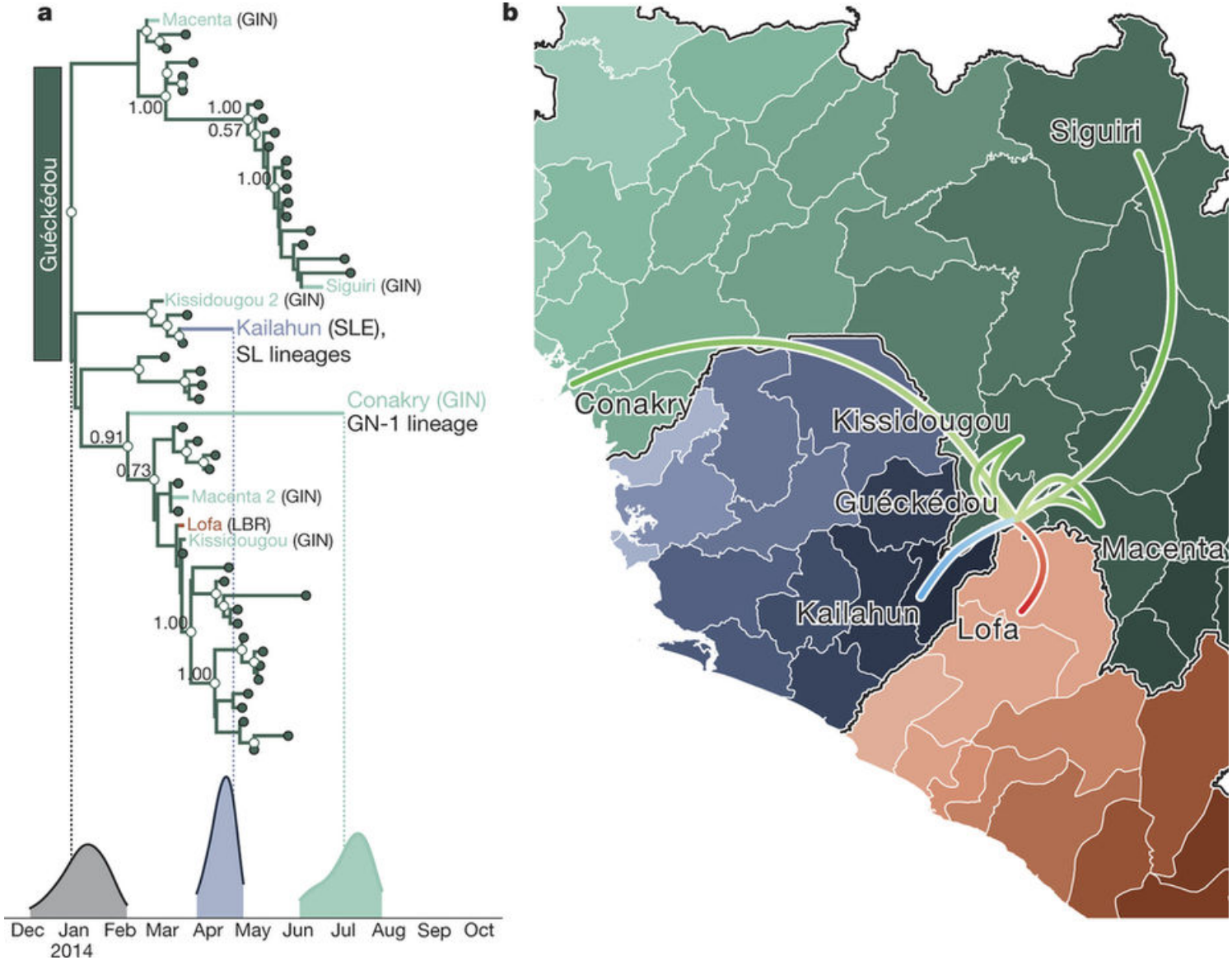
Pollock et al. 2017 Nature

Drug design



Zakas et al. 2017 Nat Biotech

Epidemiology: Ebola epidemics



Dudas et al. 2017 Nature

Phylogenetic trees: definitions.

Phylogeny and phylogenetic tree

The true and unobservable evolutionary history of a group of organisms is called a **phylogeny**.

A **phylogenetic tree** is an estimate of the true unobservable evolutionary history derived from morphology, molecular data, etc.

A **cladogram** or **topology** is the hierarchical structure of a phylogenetic tree, while a **phylogram** is a cladogram with explicit branch lengths and a **chronogram** has branch length as unit of time.

Phylogeny and phylogenetic tree

The true and unobservable evolutionary history of a group of organisms is called a **phylogeny**.

A **phylogenetic tree** is an estimate of the true unobservable evolutionary history derived from morphology, molecular data, etc.

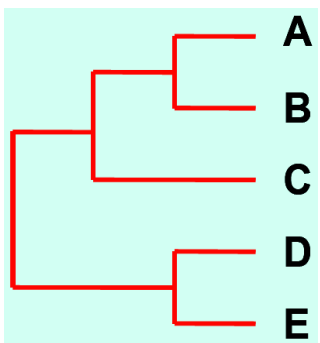
A **cladogram** or **topology** is the hierarchical structure of a phylogenetic tree, while a **phylogram** is a cladogram with explicit branch lengths and a **chronogram** has branch length as unit of time.

Phylogeny and phylogenetic tree

The true and unobservable evolutionary history of a group of organisms is called a **phylogeny**.

A **phylogenetic tree** is an estimate of the true unobservable evolutionary history derived from morphology, molecular data, etc.

A **cladogram** or **topology** is the hierarchical structure of a phylogenetic tree, while a **phylogram** is a cladogram with explicit branch lengths and a **chronogram** has branch length as unit of time.

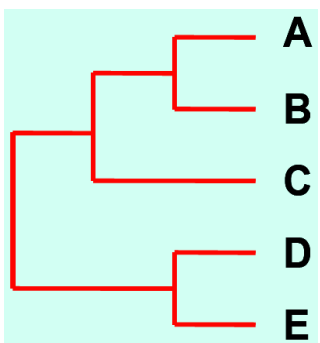


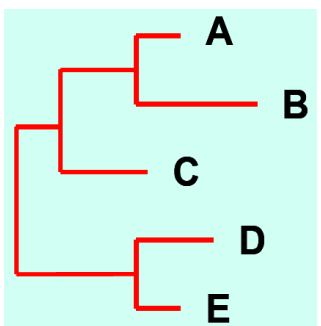
Phylogeny and phylogenetic tree

The true and unobservable evolutionary history of a group of organisms is called a **phylogeny**.

A **phylogenetic tree** is an estimate of the true unobservable evolutionary history derived from morphology, molecular data, etc.

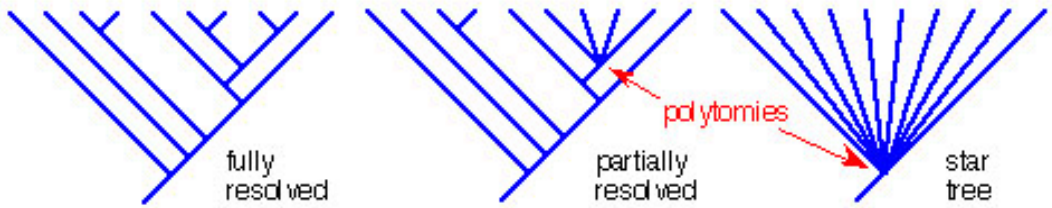
A **cladogram** or **topology** is the hierarchical structure of a phylogenetic tree, while a **phylogram** is a cladogram with explicit branch lengths and a **chronogram** has branch length as unit of time.





Hard and soft polytomies

A clade on a phylogenetic tree is **unresolved** if there is a node with three or more direct descendants.



from biology.fullerton.edu

This creates **polytomies**, which are called

soft if it's due to lack of data or a conflict between two plausible trees

hard if they represent real two or more consecutive events of speciation

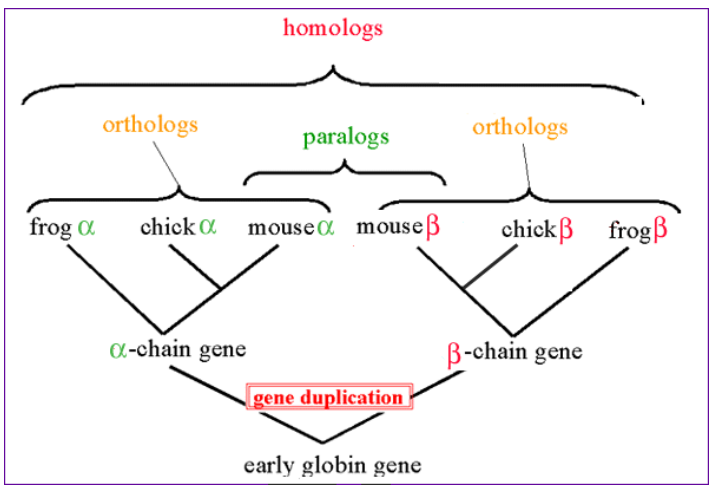
Orthologs, paralogs, xenologs

Two homologous genes are

ortholog if derived from a gene present in their common ancestor

paralog if resulted from the duplication of the same ancestral gene

xenolog if arose by lateral gene transfer



from www.discoveryandinnovation.com

When inferring species phylogenetic trees, it is **essential** to have orthologous DNA sequences!

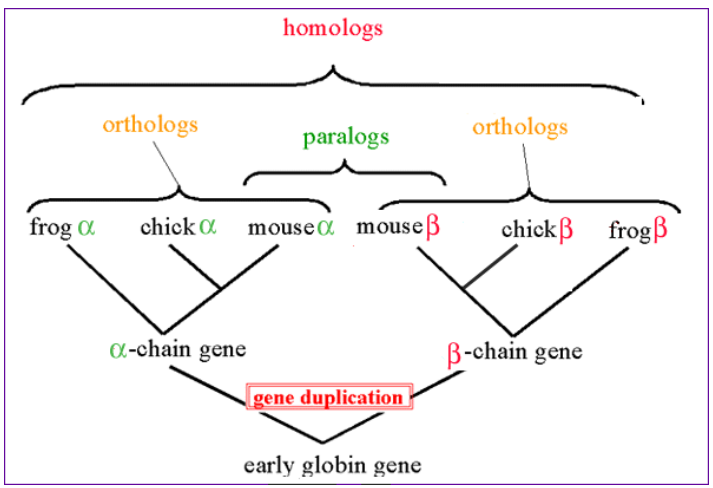
Orthologs, paralogs, xenologs

Two homologous genes are

ortholog if derived from a gene present in their common ancestor

paralog if resulted from the duplication of the same ancestral gene

xenolog if arose by lateral gene transfer



from www.discoveryandinnovation.com

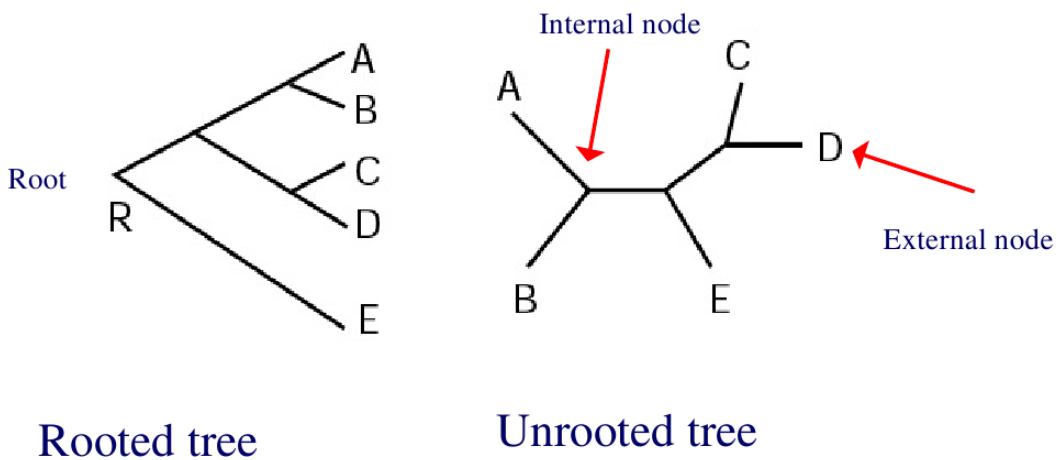
When inferring species phylogenetic trees, it is **essential** to have orthologous DNA sequences!

The root of a tree

In evolutionary studies, phylogenetic trees are drawn as branching trees deriving from a single ancestral species.

This species is know as the **root** of the tree.

- A rooted tree is a tree to which a special internal node r is added with degrees $= 2$.



Rooted tree Unrooted tree

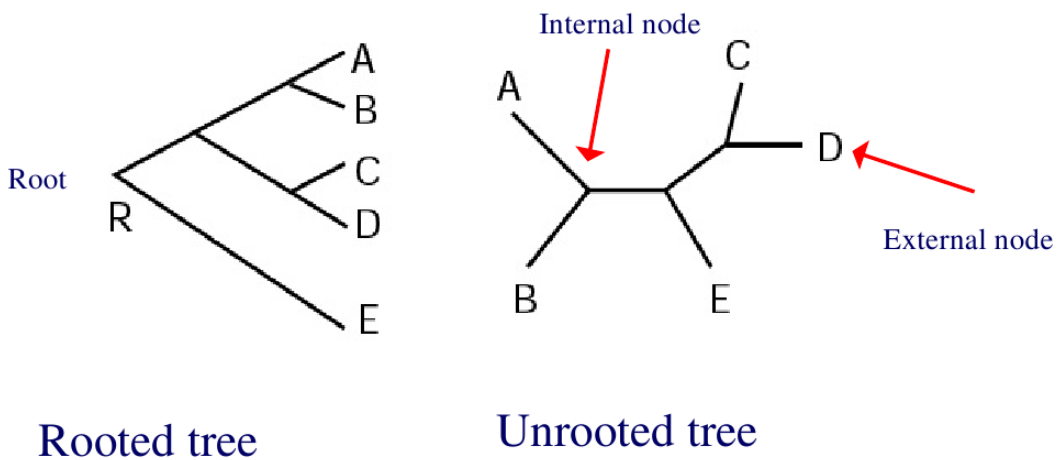
Labeled histories: rooted trees with interior nodes ordered according to their age.

The root of a tree

In evolutionary studies, phylogenetic trees are drawn as branching trees deriving from a single ancestral species.

This species is know as the **root** of the tree.

- A rooted tree is a tree to which a special internal node r is added with degrees $= 2$.



Rooted tree Unrooted tree

Labeled histories: rooted trees with interior nodes ordered according to their age.

Consensus tree

A tree that summarized the common features of a set of trees is called a consensus tree.

- strict consensus: shows only the groups that are shared among **all** trees.
- semi-strict consensus: shows only the **resolved** groups that are shared among **all** trees.
- n% majority-rule consensus: shows the groups that are shared by n% of the trees.

How do we measure evolution?

Type of models

Depending on the question, several type of models of molecular evolution:

1. infinite allele model

each mutation creates a new allele and allele types are all equally different from each other

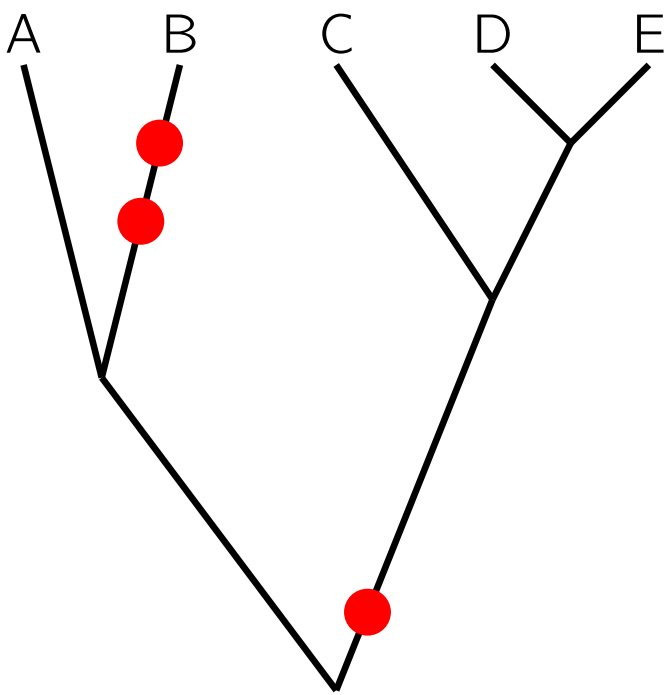
2. infinite site model

each mutation happens at a site that has not previously experienced mutation since common ancestor of sample

3. finite site model

allow the possibility that each sequence site has experienced multiple mutations

Infinite allele model



Ancestral allele of type 0

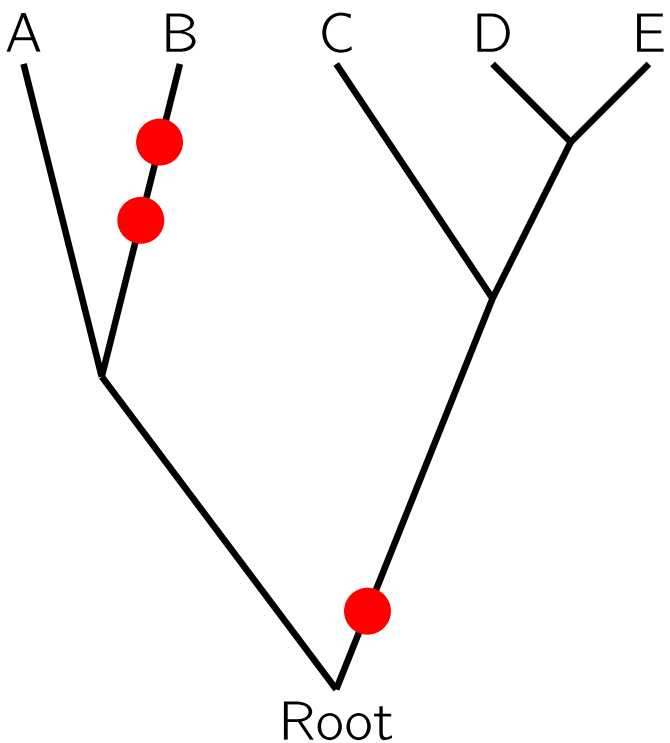
- A has the ancestral type 0
- (C,D,E) have allele type 1
- B has allele type 3
- allele type 2 not observed

Diversity typically measured as $\theta = 4N\mu$ where

- N = population size (diploid)
- μ = rate of mutation of new alleles

Typically used with microsats, SSRs, AFLP.

Infinite site model



Assuming an ancestral state at the root

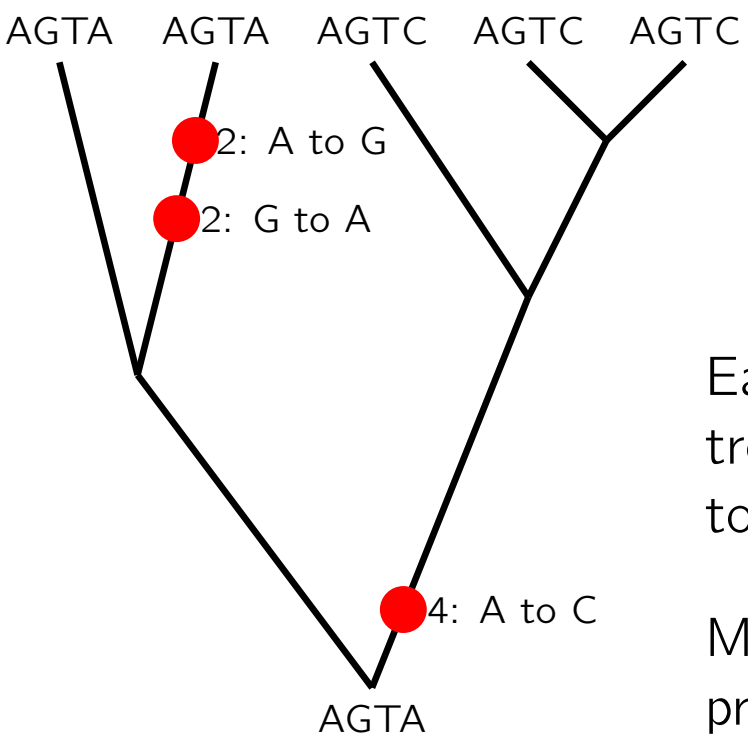
- A has the same state as the root
- (C,D,E) are identical and differ from A and Root by 1 site
- B differs from A and Root by two sites
- B differs from C, D and E by 3 sites

Expected number of sites between 2 random sequences is $\theta = 4N\mu$

Average number of sites at which a sequence pair differs denoted as $\hat{\pi}$. Tajima's D is known as an estimate of θ.

Typically used with SNPs and sequence data representing haplotypes.

Finite site model



Given an ancestral sequence at the root

- A has the same sequence as the root
- (C,D,E) are identical and differ from A and Root by 1 mutation
- B is identical from A and Root despite two mutations

Each sequence site evolves on the same gene tree as each other but mutations are assumed to occur independently.

Most often used for nucleotide substitution process (mutation plus fixation) rather than mutation process alone.

Basic model to analyse sequence data.

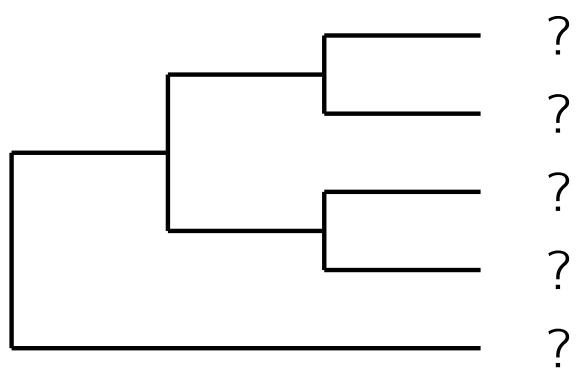
Finite site model and evolutionary trees.

Sequence data: a small example

Let's assume that you have 5 sequences:

one	ACC
two	ACG
three	TCT
four	TCA
five	CTC

How would you classify these sequences based on share ancestry?

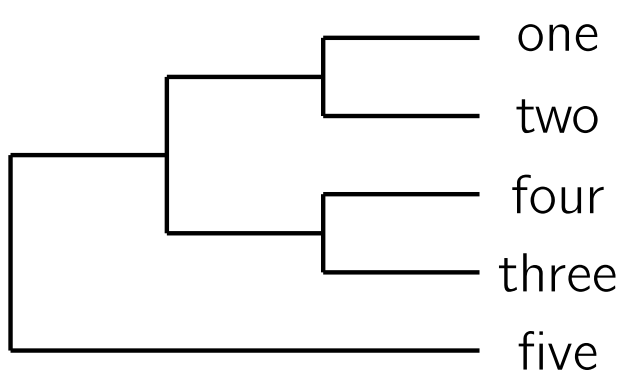


Sequence data: a small example

Let's assume that you have 5 sequences:

one	`ACC`
two	`ACG`
three	`TCT`
four	`TCA`
five	`CTC`

How would you classify these sequences based on share ancestry?

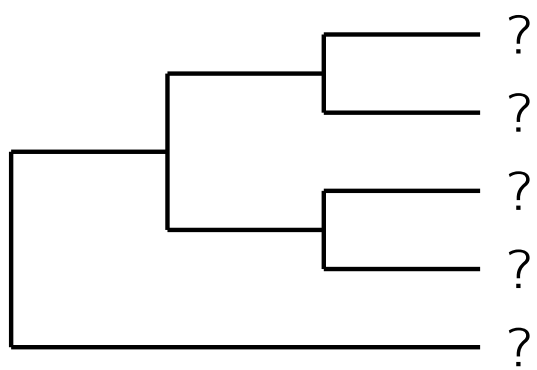


A small example, extended

Let's assume that you have 5 sequences:

one	ACCACCATGGTACCAGTGCG
two	ACGAAATACGTACCAACGAG
three	TCTAACTACGGTACTACGCA
four	TCAACCATCGGATACACGAA
five	CTCACTACCTTACTTACGCT

How would you classify these sequences based on share ancestry?



Evolutionary trees

Finding the evolutionary trees is essential to answer most of the biological questions of interest. There are four major methods for this.

Distance Methods Evolutionary distances are computed for all OTUs and these are used to construct trees.

Maximum Parsimony Trees are chosen to minimize the number of changes required to explain the data.

Maximum Likelihood Tree which gives the highest likelihood under the given model of sequence evolution.

Bayesian Methods Tree with the highest probability to give rise to the sequence data observed.

List of most tree reconstruction packages available at

`http://evolution.genetics.washington.edu/phylip/software.html`

Differences between methods

The end result of the four methods is an evolutionary tree, but they differ in the steps to reach it.

Distance methods are clustering methods and find quickly a tree. You use bootstrapping to assess the reliability of the tree found in a second set of analyses

Maximum parsimony and likelihood are optimality criteria. You need algorithm to sample the trees and select the best one based on these criteria. You use bootstrapping to assess the reliability of the tree found in a second set of analyses.

Bayesian methods use likelihood criteria to sample trees and assess their reliability all in once.

Modeling evolution through time and lineages?

Likelihood and models

Maximum likelihood relies on **explicit** probabilistic models of evolution.

But, the process of evolution is so complex and multifaceted that basic models involve assumption built upon assumption.

This reliance is often seen as a weakness of the likelihood framework, but

- the need to make explicit assumptions is a strength
- enables both inferences about evolutionary history and assessments of the accuracy of the assumptions made
- this leads to a better understanding of evolution

"The purpose of models is not to fit the data, but to sharpen the questions" (S. Karlin)

Likelihood and models

Maximum likelihood relies on **explicit** probabilistic models of evolution.

But, the process of evolution is so complex and multifaceted that basic models involve assumption built upon assumption.

This reliance is often seen as a weakness of the likelihood framework, but

- the need to make explicit assumptions is a strength
- enables both inferences about evolutionary history and assessments of the accuracy of the assumptions made
- this leads to a better understanding of evolution

> *"The purpose of models is not to fit the data, but to sharpen the questions" (S. Karlin)*

Likelihood and models

Maximum likelihood relies on **explicit** probabilistic models of evolution.

But, the process of evolution is so complex and multifaceted that basic models involve assumption built upon assumption.

This reliance is often seen as a weakness of the likelihood framework, but

- the need to make explicit assumptions is a strength
- enables both inferences about evolutionary history and assessments of the accuracy of the assumptions made
- this leads to a better understanding of evolution

"The purpose of models is not to fit the data, but to sharpen the questions" (S. Karlin)

Description of Maximum Likelihood

Given an hypothesis H and some data D, the likelihood of H is

$$L(H) = Prob(D|H) = Prob(D_1|H)Prob(D_2|H)\cdots Prob(D_n|H)$$

if the D can be split in n independent parts.

Note that $L(H)$ is **not** the probability of the hypothesis, but the probability of the data, given the hypothesis.

In our case, H will be the tree topology T, branch lengths l and all the parameters of the model of evolution ϕ.

Maximum likelihood properties (Fisher, 1922)

- consistency – converge to correct value of the parameter
- efficiency – has the smallest possible variance around true parameter value

Description of Maximum Likelihood

Given an hypothesis H and some data D, the likelihood of H is

$$L(H) = Prob(D|H) = Prob(D_1|H)Prob(D_2|H)\cdots Prob(D_n|H)$$

if the D can be split in n independent parts.

Note that $L(H)$ is **not** the probability of the hypothesis, but the probability of the data, given the hypothesis.

In our case, H will be the tree topology T, branch lengths l and all the parameters of the model of evolution ϕ.

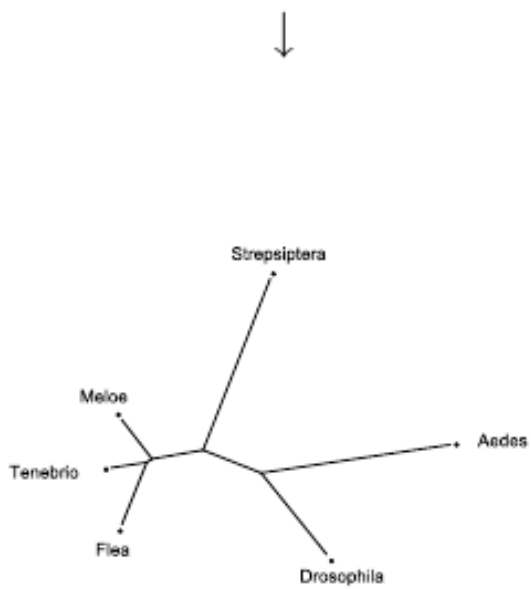
Maximum likelihood properties (Fisher, 1922)

- consistency – converge to correct value of the parameter
- efficiency – has the smallest possible variance around true parameter value

Measuring evolution

The problem that we are dealing with here is how to calculate the probability of evolution in sequences (whether DNA, AA at the inter- or intra-specific level).

```
Strepsiptera    AAGCTCATTAAATCGCTTTGGTTCCTTAGATAGTTGGAT...
Aedes           AGGCTCAGTATAACACTATAATTTACAAGATCATTGGAT...
Drosophila      AGGCTCATTATATCATTATGGTTCCTTAGATCGTTGGAT...
Flea            TGGCTCATTATATCATTATGGTTCATTAGATCGTTGGAT...
Meloe           AGGCTCATTAAATCATTATGGTTCCTTAGATCGTTGGAT...
Tenebrio        AGGCTCATTAAATCATTATGGTTCCTTAGATCGTTGGAT...
```



What do we try to model

From the alignment, it is easy to come up with a distance between species $\hat{p} = n_d/n$, which is the proportion of different nucleotides.

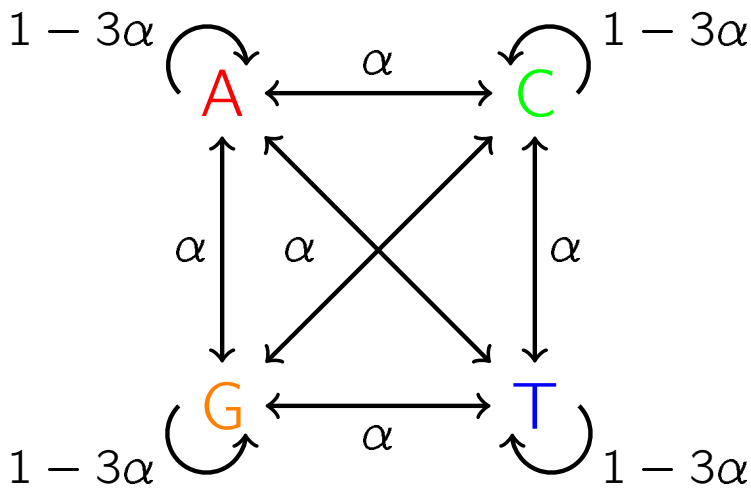
However,

- if p is small, $\hat{p}$ is a good approximation of the number of substitutions per site.
- if p is large, $\hat{p}$ will underestimate p because of multiple substitutions.
- distance methods could be used to do that, but they are approximations.

We thus need a better way to model how substitutions/mutations are appearing during evolution. Markov processes can help us a lot in this.

Derivation of the process

The simplest model assumes the following:



Each nucleotide has an equal chance of changing and when it changes, it changes to one of the three other nucleotide with equal probability (α).

The parameter α is the probability of a substitution (i.e. mutation + fixation).

This will create a Markov chain explaining the evolution of the sequence.

Probabilities of the Markov chain

Imagine that φ is the stationary distribution of the Markov chain. The probability of observing A at site i after one time step is

$$\begin{aligned} \varphi_A^{t+1} &= (1-3\alpha)\varphi_A^t + \alpha\varphi_C^t + \alpha\varphi_G^t + \alpha\varphi_T^t \\ &= (1-3\alpha)\varphi_A^t + \alpha\left[1-\varphi_A^t\right] \end{aligned}$$

The model is symmetric and it applies equally well to C, G or T, thus

$$\varphi_y^{t+1} = (1-4\alpha)\varphi_y^t + \alpha$$

Differential equation

We can substract φ_y^t from both sides

$$\varphi_y^{t+1} - \varphi_y^t = \alpha \left(1 - 4\varphi_y^t\right)$$

and apply a time continuous approximation to get a differential equation

$$\frac{d\varphi_y^t}{dt} = \alpha \left(1 - 4\varphi_y^t\right)$$

with solution

$$\varphi_y^t = \frac{1}{4} + \left(\varphi_y^0 - \frac{1}{4}\right) e^{-4\alpha t}$$

plot

Probabilities of [no] change

We can estimate the probability of observing a nucleotide A at site i after an arbitrarily long time t.

We have however two cases

1. the ancestral nucleotide was also A ($\varphi^0_{x=y} = 1$):

$$p_{A\to A}(t) = \frac{1}{4} + \frac{3}{4}e^{-4\alpha t}$$

2. the ancestral nucleotide was, say, a C ($\varphi^0_{x\neq y} = 0$):

$$p_{C\to A}(t) = \frac{1}{4} - \frac{1}{4}e^{-4\alpha t}$$

but 3 ways descendant with A could differ from the ancestor

$$p_{[CGT]\to A}(t) = \frac{3}{4} - \frac{3}{4}e^{-4\alpha t}$$

The process in R

```
pxx <-  function(r=1/3,t=0.1) {
  return(0.25+0.75*exp(-4*r*t));
}

pyx <- function(r=1/3,t=0.1) {
  return(0.75-0.75*exp(-4*r*t));
}
time <- seq(0,1.0,0.01)
plot(time, pyx(r=1/3,t=time), type="l", xlab="Time",
    ylab="Differences per site", ylim=c(0,0.75))
abline(a=0,b=1,lty=2)
lines(time, pyx(r=1/4,t=time), type="l", col="red")
lines(time, pyx(r=1/5,t=time), type="l", col="blue")
```

How do you explain the plot? Change the t from $0 \ldots 1.0$ to $0 \ldots 10.0$. What is the difference? What is the effect of r and t?

Generalizing the model of evolution

The initial condition $P(0)$ of the probability matrix P gives the probability of change for a time of 0. It is obviously

$$\lim_{t \to 0} P(t) = I$$

For continuous Markov chains, it exists an infinitesimal or instantaneous rate matrix that fully describes the process of the chain

$$\lim_{t \to 0} \frac{P(t) - P(0)}{t} = Q$$

For a tiny time h, the transition probabilities are approximated by (as shown before)

$$P(h) \approx I + Qh$$

Instantaneous matrix

For DNA, Q is

$$Q = \begin{bmatrix} - & r_{AC} & r_{AG} & r_{AT} \\ r_{CA} & - & r_{CG} & r_{CT} \\ r_{GA} & r_{GC} & - & r_{GT} \\ r_{TA} & r_{TC} & r_{TG} & - \end{bmatrix}$$

with $r_{xy} = \mu_{x \to y} * \pi_y$ and

- from theory of finite Markov process,

$$q_{ii} = - \sum_{j=0, j \neq i}^{4} q_{ij},$$

- thus

$$\sum_{j=0}^{4} q_{ij} = 0.$$

like initial model

Differential equation

We want the probability $P(t)$ of substitutions over a time t. Let's estimate first $P(t+h)$, with h being an tiny time interval.

The definition of Markov chains tells us that $P(t+h) = P(t)P(h)$. Replacing $P(h)$ by its approximation gives

$$P(t+h) = P(t)(I+Qh) = P(t) + P(t)Qh$$

Rearranging, we get

$$\frac{P(t+h) - P(t)}{h} = P(t)Q$$

Taking the limit as $h \to 0$ gives $P' = PQ$, with intial condition $P(0) = I$, which leads to

$$P(t) = e^{Qt}$$

Rate of substitutions

As Q and t occur only in the form of a product in $P(t) = e^{Qt}$, it is conventional to scale Q so that the average rate is 1.

We do this by normalizing Q by the total amount of change involved in the Markov chain:

$$\sum_{i=0}^{4} \sum_{j=0, j \neq i}^{4} \pi_i q_{ij}$$

The branch length is therefore measured in **expected substitutions per site**.

A long branch can therefore either by due to

- long evolutionary time
- a rapid rate of substitution
- a combination of both

Jukes-Cantor, 1969

$$Q = \begin{bmatrix} - & 1 & 1 & 1 \\ 1 & - & 1 & 1 \\ 1 & 1 & - & 1 \\ 1 & 1 & 1 & - \end{bmatrix} \times \begin{bmatrix} \pi_A \\ \pi_C \\ \pi_G \\ \pi_T \end{bmatrix}$$

with $\pi_A = \pi_C = \pi_G = \pi_T = 0.25$.

Transition probabilities are $P(t) = e^{Qt}$

$$p_{ij}(t) = \begin{cases} 1/4 + 3 * exp(-4\mu t)/4, & \text{if } i = j, \\ 3/4 - 3 * exp(-4\mu t)/4, & \text{if } i \neq j, \end{cases}$$

Matrix representation in R

```
JC <- function() {
    freq <- matrix(0, nrow=4, ncol=4)
    diag(freq) <- c(.25,.25,.25,.25)
    R <- matrix(c(0,1,1,1,1,0,1,1,1,1,0,1,1,1,1,0), nrow=4)
    Q <- (R %*% freq) - diag(apply(R %*% freq, 1, sum))
    scaleQ <- sum(apply(freq %*% R %*% freq, 1, sum))
    return(Q * scaleQ);
}

library(Matrix)

calcP <- function(R, t=0.1) { return(sum(expm(R * t)[1, -1]))}
time <- seq (0, 1, 0.01)
plot(time, pyx(r=0.3, t=time), type="l")
lines(time, sapply(time/0.625, calcP, JC()), col="red", lty=2)
```

Is it the same as before? Where is the 0.625 coming from? Try with more complex models (see below).

Kimura 2-parameters, 1981

$$Q = \begin{bmatrix} - & 1 & \kappa & 1 \\ 1 & - & 1 & \kappa \\ \kappa & 1 & - & 1 \\ 1 & \kappa & 1 & - \end{bmatrix} \times \begin{bmatrix} \pi_A \\ \pi_C \\ \pi_G \\ \pi_T \end{bmatrix}$$

Here, $\kappa = \alpha/\beta$, where

- α is the transition rate
- β is the transversion rate.

and $\pi_A = \pi_C = \pi_G = \pi_T = 0.25$.

It simplifies to

- Jukes-Cantor, 1969 if $\alpha = \beta$.

Matrix representation in R

```
K2P <- function(k) {
    freq <- matrix(0, nrow=4, ncol=4);
    diag(freq) <- c(.25,.25,.25,.25);
    R <- matrix(c (0,1,k,1,1,0,1,k,k,1,0,1,1,k,1,0), nrow=4);
    Q <- (R %*% freq) - diag(apply(R %*% freq, 1, sum))
    scaleQ <- sum(apply(freq %*% R %*% freq, 1, sum))
    return(Q * scaleQ);
}

library(Matrix)

calcP <- function(R, t=0.1) { return(sum(expm(R * t)[1, -1]));}
time <- seq (0, 5, 0.01)
plot(time, sapply(time, calcP, JC()), type="l")
lines(time, sapply(time, calcP, K2P(1)), col="green", lty=2)
lines(time, sapply(time, calcP, K2P(2)), col="blue", lty=2)
lines(time, sapply(time, calcP, K2P(5)), col="red", lty=2)
```

What is the difference between JC69 and K2P?

Felsenstein, 1981

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

where π_i is the equilibrium frequency of nucleotide i.

It simplifies to

- Jukes-Cantor, 1969 if $\pi_A = \pi_C = \pi_G = \pi_T$.

Matrix representation in R

```
F81 <- function(f) {
    freq <- matrix(0, nrow=4, ncol=4);
    diag(freq) <- f;
    R <- matrix(c(0,1,1,1,1,0,1,1,1,1,0,1,1,1,1,0), nrow=4);
    Q <- (R %*% freq) - diag(apply(R %*% freq, 1, sum))
    scaleQ <- sum(apply(freq %*% R %*% freq, 1, sum))
    return(Q * scaleQ);
}

library(Matrix)

calcP <- function(R, t=0.1) { return(sum(expm(R * t)[1, -1]));}
time <- seq (0, 1, 0.01)
plot(time, sapply(time, calcP, JC()), type="l")
lines(time, sapply(time, calcP, K2P(2), col="blue", lty=2)
lines(time, sapply(time, calcP, F81(c(0.23,0.27,0.28,0.22)),
    col="red", lty=2)
```

What is the difference between JC69 and F81?

Hasegawa-Kishino-Yano, 1985

$$Q = \begin{bmatrix} - & \pi_C & \kappa\pi_G & \pi_T \\ \pi_A & - & \pi_G & \kappa\pi_T \\ \kappa\pi_A & \pi_C & - & \pi_T \\ \pi_A & \kappa\pi_C & \pi_G & - \end{bmatrix}$$

Here, π_i is the equilibrium frequency of nucleotide i and $\kappa = \alpha/\beta$, where

- α is the transition rate
- β is the transversion rate.

It simplifies to

- Kimura, 1980 if $\pi_A = \pi_C = \pi_G = \pi_T$.
- Felsenstein, 1981 if $\kappa = 1$.

Tamura-Nei, 1993

$$Q = \begin{bmatrix} - & \pi_C & \kappa_2\pi_G & \pi_T \\ \pi_A & - & \pi_G & \kappa_1\pi_T \\ \kappa_2\pi_A & \pi_C & - & \pi_T \\ \pi_A & \kappa_1\pi_C & \pi_G & - \end{bmatrix}$$

Here, π_i is the equilibrium frequency of nucleotide i, $\kappa_1 = \alpha_Y/\beta$ and $\kappa_2 = \alpha_R/\beta$ where

- α_R is purine transition rate (A+G).
- α_Y is pyrimidine transition rate (C+T).
- β is the transversion rate.

It simplifies to

- Hasegawa, Kishino, Yano, 1985 if $\alpha_R/\alpha_Y = \pi_R/\pi_Y$.
- Felsenstein, 1984 if $\alpha_R = \alpha_Y$.

General-time reversible

$$Q = \begin{bmatrix} - & \alpha\pi_C & \beta\pi_G & \gamma\pi_T \\ \alpha\pi_A & - & \delta\pi_G & \epsilon\pi_T \\ \beta\pi_A & \delta\pi_C & - & \eta\pi_T \\ \gamma\pi_A & \epsilon\pi_C & \eta\pi_G & - \end{bmatrix}$$

where

- $\alpha \cdots \eta$ are rates of changes from one nucleotide to another
- π_i are frequencies of nucleotides

It simplifies to

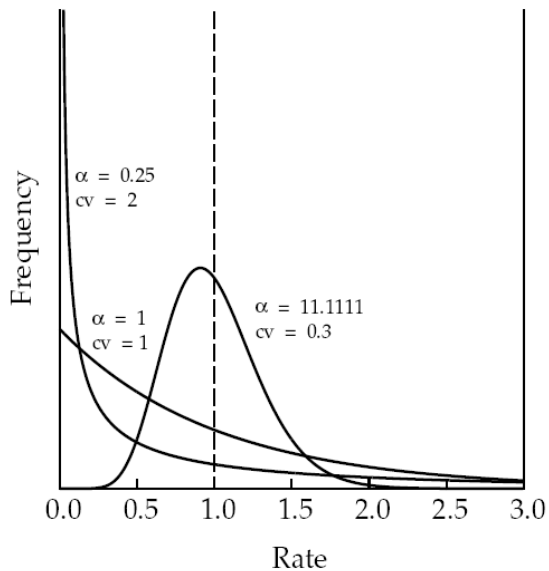
- all other models if parameters are correctly constrained

Variation of substitution rates

How to model rate variation among sites?

The idea is to use a probability distribution to model changes in rates of substitution among sites, e.g. Gamma distribution

- mean of distribution is $\alpha\beta$, and variance $\alpha\beta^2$
- set the mean rate of substitution to 1, so assume $\beta = 1/\alpha$
- α parameter allows to change characteristics of distribution



Felsenstein, 2004

Number of site classes

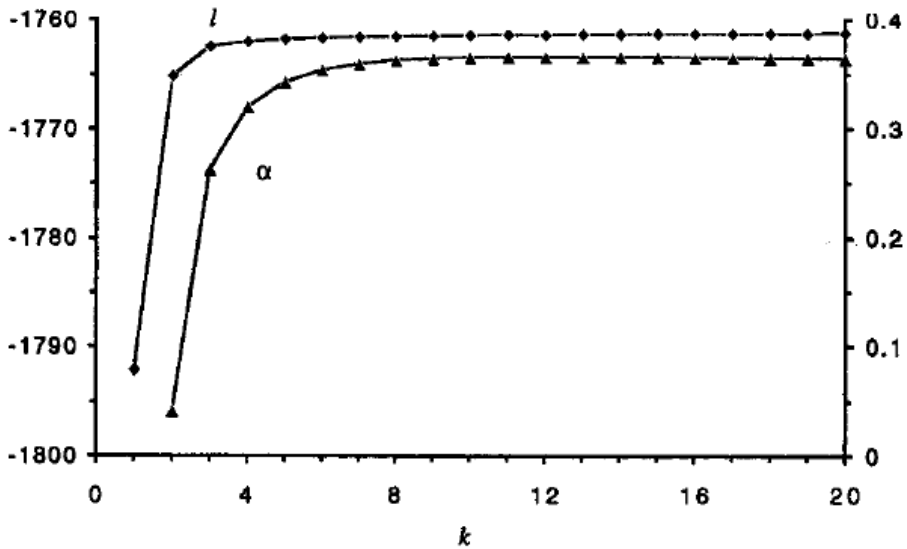


Fig. 4. Likelihood values and estimates of the α parameter as functions of k, the number of categories in the discrete gamma model. The α and β globin genes for the five mammalian orders (570 bp) are analyzed, assuming the best tree (Fig. 3) and the F84 + dG model. The average nucleotide frequencies are $\pi_T = 0.2200$, $\pi_C = 0.2449$, $\pi_A = 0.2761$, and $\pi_G = 0.2590$, with $\ell_{max} = -1{,}579.76$. When $k = \infty$, that is, with the F84 + Γ model, $\ell = -1{,}761.17$ and $\hat{\alpha} = 0.360$.

Yang, 1994

How to calculate the likelihood of a tree?

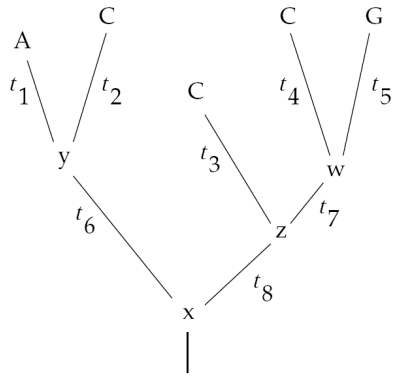
Prerequisite

Suppose we have a data set D of DNA sequences with m sites.

We are given a topology T with branch lengths and a model of evolution, Q that allow us to compute $P_{ij}(t)$.

Assumptions made to compute likelihood $L(T, Q) = Prob(D|T, Q)$

- evolution in different sites is independent
- evolution in different lineages is independent
- the rate of evolution is the same between site and along lineages



Felsenstein, 2004

Independence of sites

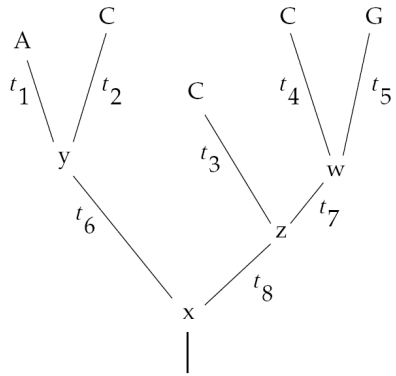
The assumption of independence of sites along a sequence allow us to decompose the likelihood in a product of likelihood for each site

$$L(T, Q) = Prob(D|T, Q) = \prod_{i=1}^{m} Prob(D_i|T, Q)$$

The likelihood for this tree for site i is

$$Prob(D_i|T, Q) = \sum_{x}^{ACGT} \sum_{y}^{ACGT} \sum_{z}^{ACGT} \sum_{w}^{ACGT}$$

$$Prob(A, C, C, C, G, x, y, z, w|T, Q)$$



Felsenstein, 2004

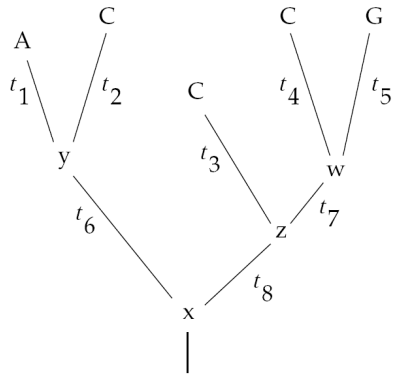
Independence of sites

The assumption of independence of sites along a sequence allow us to decompose the likelihood in a product of likelihood for each site

$$L(T, Q) = Prob(D|T, Q) = \prod_{i=1}^{m} Prob(D_i|T, Q)$$

The likelihood for this tree for site i is

$$Prob(D_i|T, Q) = \sum_{x}^{ACGT} \sum_{y}^{ACGT} \sum_{z}^{ACGT} \sum_{w}^{ACGT} Prob(A, C, C, C, G, x, y, z, w|T, Q)$$



Felsenstein, 2004

Independence of lineages

With the independence of lineages assumptions, we can decompose the right hand side of the equation a bit further.

$$Prob(A, C, C, C, G, x, y, z, w | T, Q) = \sum_{x}^{ACGT} \sum_{y}^{ACGT} \sum_{z}^{ACGT} \sum_{w}^{ACGT}$$

$$Prob(x) P_{xy}(t_6) P_{yA}(t_1) P_{yC}(t_2)$$
$$P_{xz}(t_8) P_{zC}(t_3)$$
$$P_{zw}(t_7) P_{wC}(t_4) P_{wG}(t_3)$$

Pruning algorithm

- **Goal**: render the likelihood computation practicable using "dynamic programming"
- **Idea**: move summation signs as far right as possible and enclose them in parentheses where possible

$$Prob(A, C, C, C, G, x, y, z, w|T) =$$
$$\sum_{x}^{ACGT} Prob(x)\Big(\sum_{y}^{ACGT} P_{xy}(t_6)P_{yA}(t_1)P_{yC}(t_2)\Big)$$
$$\times \Big(\sum_{z}^{ACGT} P_{xz}(t_8)P_{zC}(t_3)$$
$$\times \Big(\sum_{w}^{ACGT} P_{zw}(t_7)P_{wC}(t_4)P_{wG}(t_5)\Big)\Big)$$

The parentheses pattern and terms for tips has an exact correspondence to the structure of the tree.

Pruning algorithm

- **Goal**: render the likelihood computation practicable using "dynamic programming"
- **Idea**: move summation signs as far right as possible and enclose them in parentheses where possible

$$Prob(A, C, C, C, G, x, y, z, w|T) =$$
$$\sum_{x}^{ACGT} Prob(x)\Big(\sum_{y}^{ACGT} P_{xy}(t_6)P_{yA}(t_1)P_{yC}(t_2)\Big)$$
$$\times \Big(\sum_{z}^{ACGT} P_{xz}(t_8)P_{zC}(t_3)$$
$$\times \Big(\sum_{w}^{ACGT} P_{zw}(t_7)P_{wC}(t_4)P_{wG}(t_5)\Big)\Big)$$

The parentheses pattern and terms for tips has an exact correspondence to the structure of the tree.

Numerical integration

Up to now we assumed that both branch lengths and model parameters were fixed at a given value.

Of course, we don't know these values so

- to get these values, start from a random guess
- make small changes and compare the improvement in likelihood
- do so until the likelihood do not change any more

- every time the topology is changed, reestimate these parameters

Drawback: very very very computationally intensive, but there are solutions to that.

Consistency and overparameterisation

Maximum likelihood can be proved to be consistent (see Felsenstein, 2004 pp. 271)

- true if we use the correct model of evolution
- but what happen if model in not correct? No guarantee can be given
- less problematic if what we want to infer is just the topology

Beware of overparameterisation problems

- adding more and more parameters to the model will result in a better fit to the data
- but this will lead to inconsistency
- a few parameters are more important than others, in particular the gamma distribution

How to choose the best tree?

Number of rooted trees?

$n = 2$

a b

$n = 3$

a b c a c b b c a

$n = 4$

a b c d a c b d b c a d a b d c a d b c

b d a c a c d b a d c b d c a b b c d a

b d c a c d b a a b c d a c b d a d b c

More trees than electrons in Universe

Species	Number of trees
1	1
2	1
3	3
4	15
5	105
6	945
7	10,395
8	135,135
9	2,027,025
10	34,459,425
11	654,729,075
12	13,749,310,575
13	316,234,143,225
14	7,905,853,580,625
15	213,458,046,676,875
16	6,190,283,353,629,375
17	191,898,783,962,510,625
18	6,332,659,870,762,850,625
19	221,643,095,476,699,771,875
20	8,200,794,532,637,891,559,375
30	4.9518×10^{38}
40	1.00985×10^{57}
50	2.75292×10^{76}

Branching process

- n-th taxa can be added to $2n - 3$ places
- $3 \times 5 \times 7 \times \cdots \times (2n - 3)$
- $\frac{(2n-3)!}{2^{n-1}(n-1)!}$
- with 50 taxa, approaching Eddington's number of electons in the visible universe

More trees than electrons in Universe

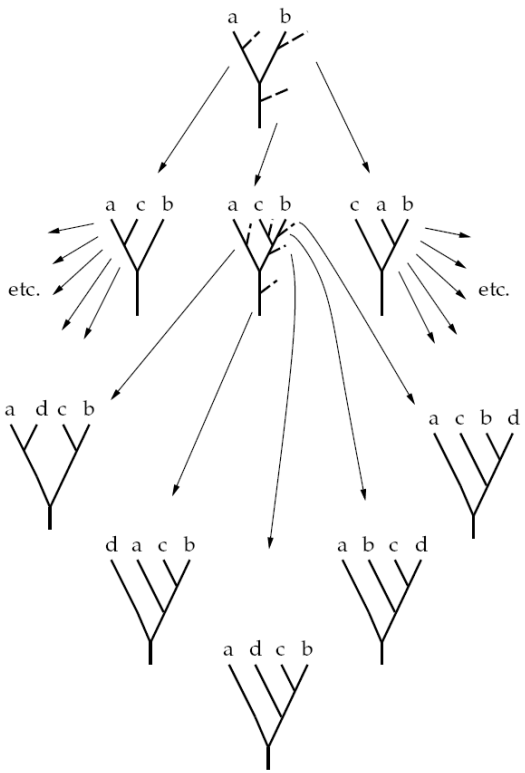
Species	Number of trees
1	1
2	1
3	3
4	15
5	105
6	945
7	10,395
8	135,135
9	2,027,025
10	34,459,425
11	654,729,075
12	13,749,310,575
13	316,234,143,225
14	7,905,853,580,625
15	213,458,046,676,875
16	6,190,283,353,629,375
17	191,898,783,962,510,625
18	6,332,659,870,762,850,625
19	221,643,095,476,699,771,875
20	8,200,794,532,637,891,559,375
30	4.9518×10^{38}
40	1.00985×10^{57}
50	2.75292×10^{76}

Branching process

- n-th taxa can be added to $2n - 3$ places
- $3 \times 5 \times 7 \times \cdots \times (2n - 3)$
- $\frac{(2n-3)!}{2^{n-1}(n-1)!}$
- with 50 taxa, approaching Eddington's number of electons in the visible universe

Exhaustive search

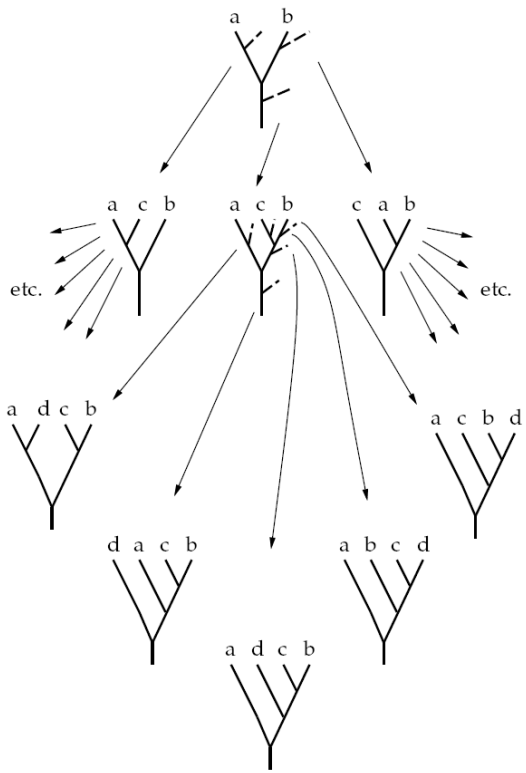
To find the best tree, enumerate all the possible trees from the tree space:



But you can quickly see that it is impossible for trees larger than 10 taxa. . . We need to take shortcuts

Exhaustive search

To find the best tree, enumerate all the possible trees from the tree space:



But you can quickly see that it is impossible for trees larger than 10 taxa. . . We need to take shortcuts

Swapping algorithms: shortcuts

Algorithms to speed up even more the search through the tree space

- take an initial tree T_i
- make small rearrangements (i.e. swap branches) to reach neighboring trees
- if you find a better, keep it and start again

- strategy will find local optimum in the tree space

They are called **greedy** algorithm because they sieze the first improvement they see

Swapping algorithms: shortcuts

Algorithms to speed up even more the search through the tree space

- take an initial tree T_i
- make small rearrangements (i.e. swap branches) to reach neighboring trees
- if you find a better, keep it and start again

- strategy will find local optimum in the tree space

They are called **greedy** algorithm because they sieze the first improvement they see

Swapping algorithms: shortcuts

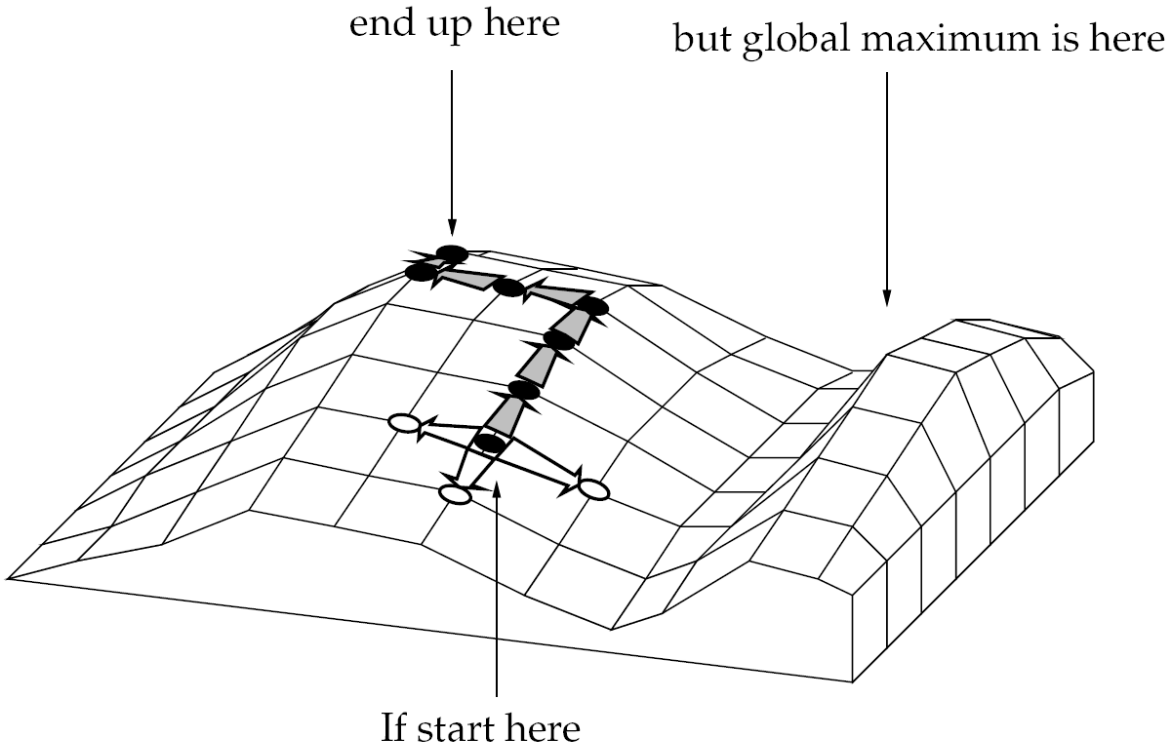
Algorithms to speed up even more the search through the tree space

- take an initial tree T_i
- make small rearrangements (i.e. swap branches) to reach neighboring trees
- if you find a better, keep it and start again

- strategy will find local optimum in the tree space

They are called **greedy** algorithm because they sieze the first improvement they see

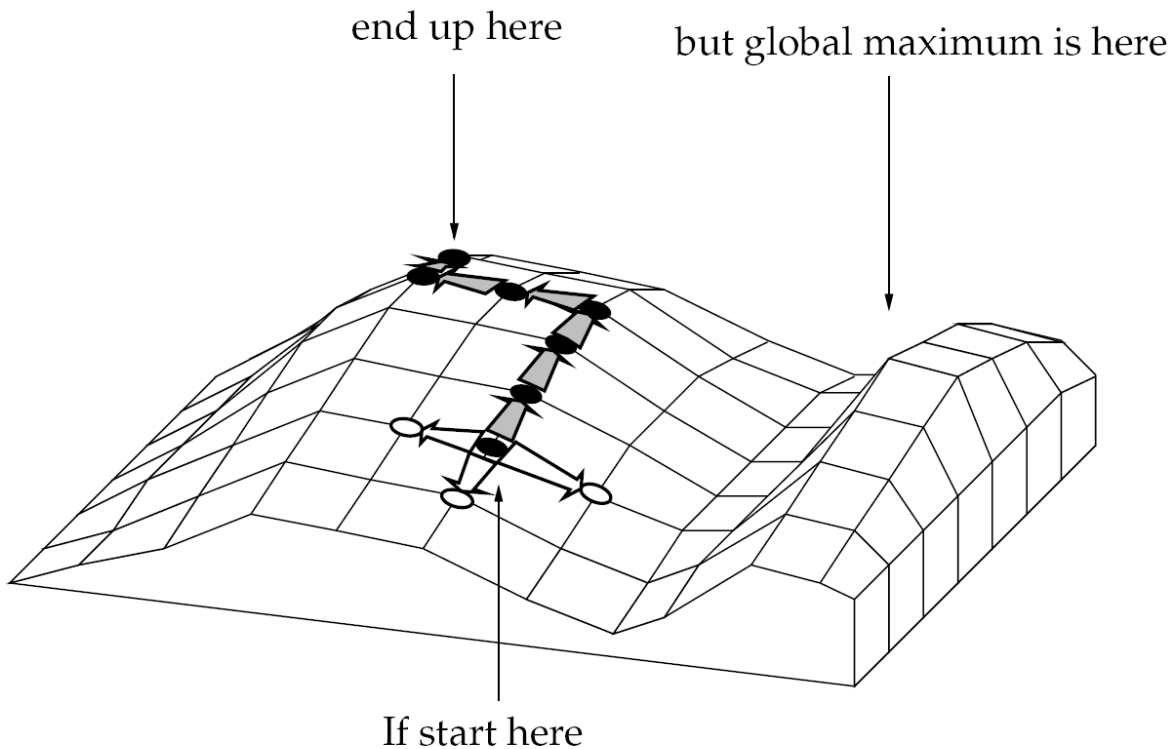
Greedy algorithms



The **problem** with greedy algorithms:

- you can miss the global optimum!
- but we don't have any other choice...

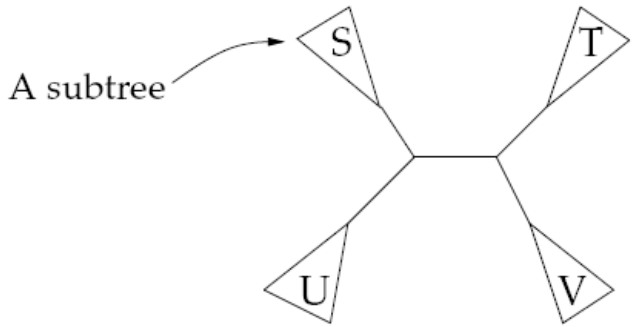
Greedy algorithms



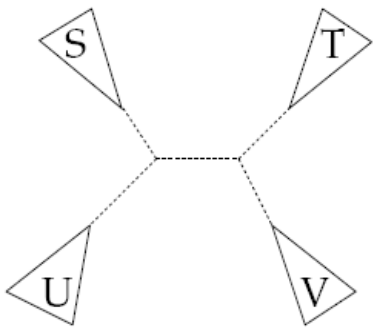
The **problem** with greedy algorithms:

- you can miss the global optimum!
- but we don't have any other choice...

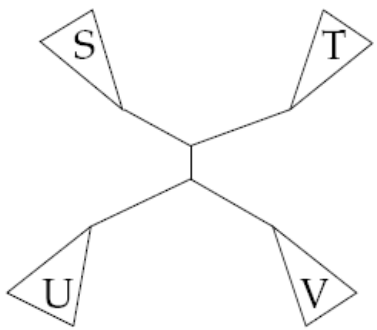
Nearest-Neighbour Interchange

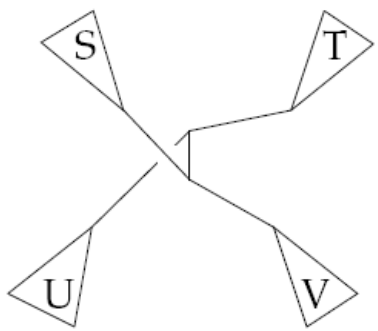


is rearranged by dissolving the connections to an interior branch

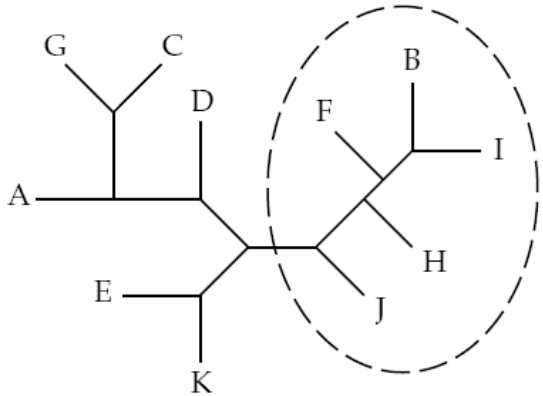


and reforming them in one of the two possible alternative ways:

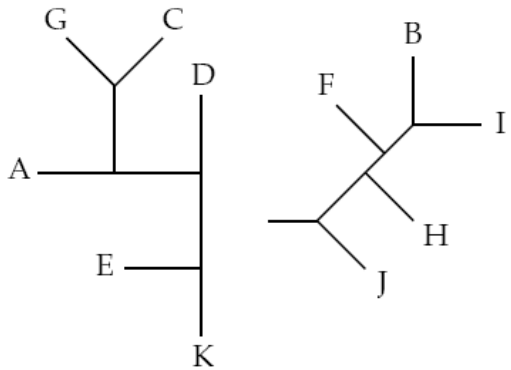




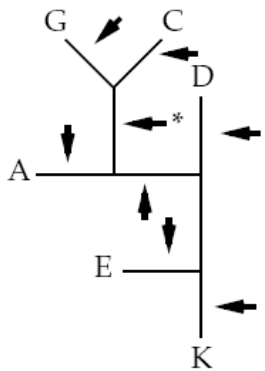
Subtree Pruning and Regrafting



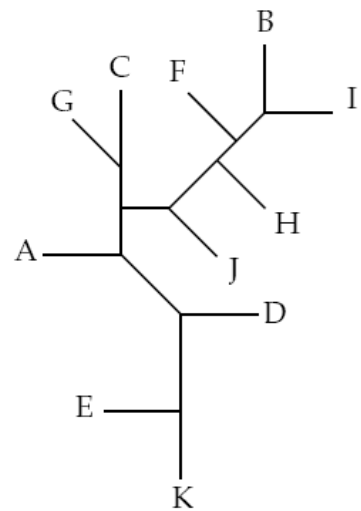
Break a branch, remove a subtree



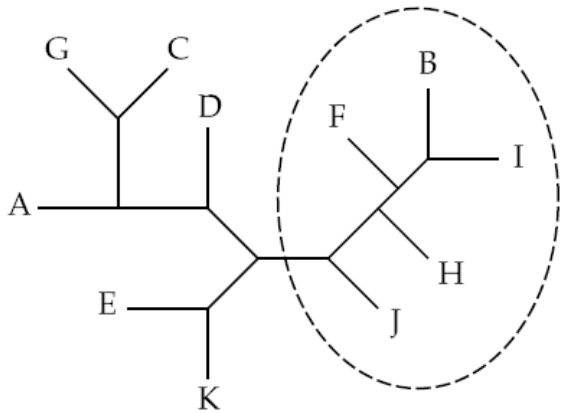
Add it in, attaching it to one (*) of the other branches



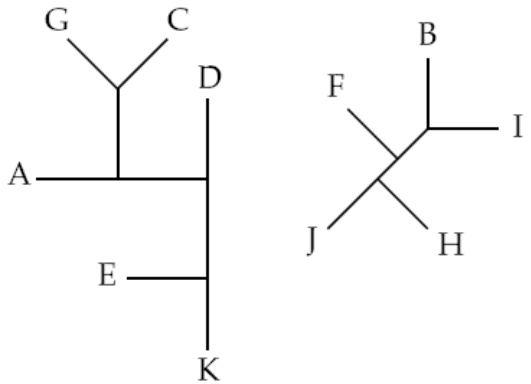
Here is the result:



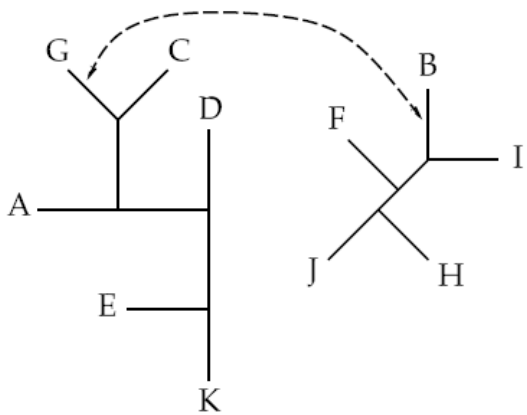
Tree Bisection and Reconnection



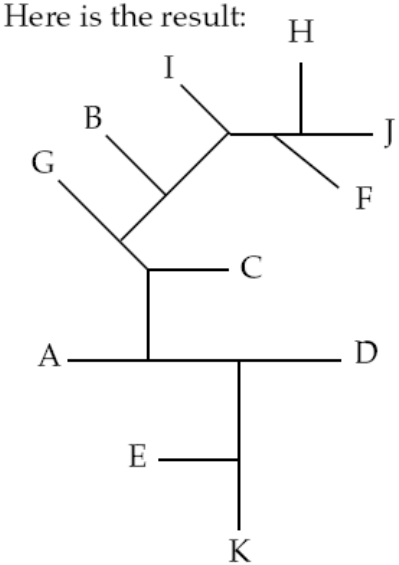
Break a branch, separate the subtrees



Connect a branch of one to a branch of the other



Here is the result:



Swapping is not all

Number of "neighbor" searched

NNI $2(n-3)$

SPR $4(n-3)(n-2)$, although some may be the same neighour

TBR $(2n_1-3)(2n_2-3)$ possible ways to reconnect the two trees

Previous methods just make rearrangements to a particular tree by swapping branches, which is not enough to fully cover the tree space. Additional techniques have to be used

- stepwise taxon addition: start several heuristic searches by starting from a different initial tree each time
- allow suboptimal trees to be swapped on in order to cross "valleys" in the tree space

Other heuristic searches

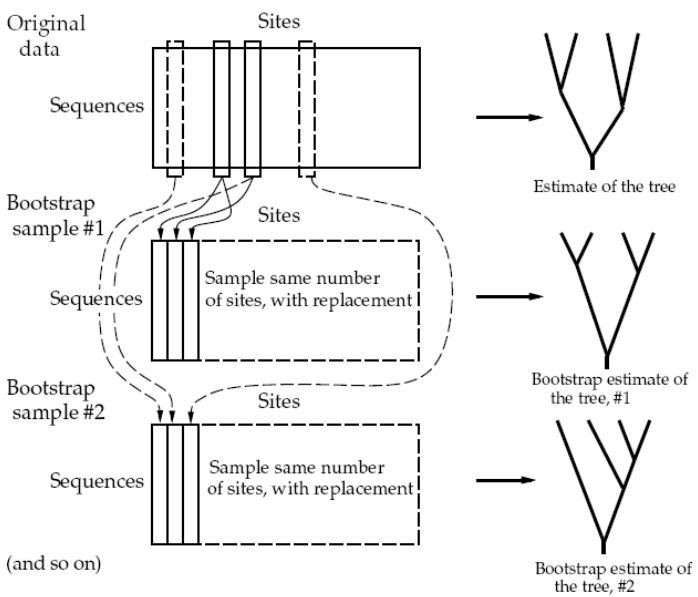
You can "invent" all kinds of heuristic algorithm, and many have been proposed

- tree-fusing: exchange subtrees from two equally optimal trees
- genetic algorithms: let populations of trees evolve by setting fitness functions and mutation, migration, etc...
- tree windowing: do extensive rearrangement but on a local region of the tree
- search by reweighting: change the landscape of the tree space by reweighting characters
- simulated annealing: iteratively change the tree scoring function to lower the difference between optimal and suboptimal trees

How to know if the tree is correct?

Bootstrap on phylogenetic tree

Allow us to infer the variability of parameters in models that are too complex for easy calculation of their variance. This is the case of topologies!



Procedure

- sample whole columns of data with replacement
- recreate n pseudomatrices with the same number of species and sites than original one
- build n phylogenetic trees from these n pseudoreplicates
- weigh each tree in replicate i by the number of trees obtained

Summarizing results

The end result of a bootstrap is a cloud of trees.

What is the best way to summarize this given that trees have discrete topologies and continuous branch lengths?

We could make a histogram of the length of a particular branches

- it will give a lower limit on the branch length
- then check if 0 is in the 95% interval, we would assert the existence of the branch
- branch-length tests available (see practicals)

A simpler solution is to count how many times a particular branch appears in the list of trees estimated by bootstrap.

A majority-rule consensus tree containing clades appearing in more than 50% of them can then be built

Summarizing results

The end result of a bootstrap is a cloud of trees.

What is the best way to summarize this given that trees have discrete topologies and continuous branch lengths?

We could make a histogram of the length of a particular branches

- it will give a lower limit on the branch length
- then check if 0 is in the 95% interval, we would assert the existence of the branch
- branch-length tests available (see practicals)

A simpler solution is to count how many times a particular branch appears in the list of trees estimated by bootstrap.

A majority-rule consensus tree containing clades appearing in more than 50% of them can then be built

Summarizing results

The end result of a bootstrap is a cloud of trees.

What is the best way to summarize this given that trees have discrete topologies and continuous branch lengths?

We could make a histogram of the length of a particular branches

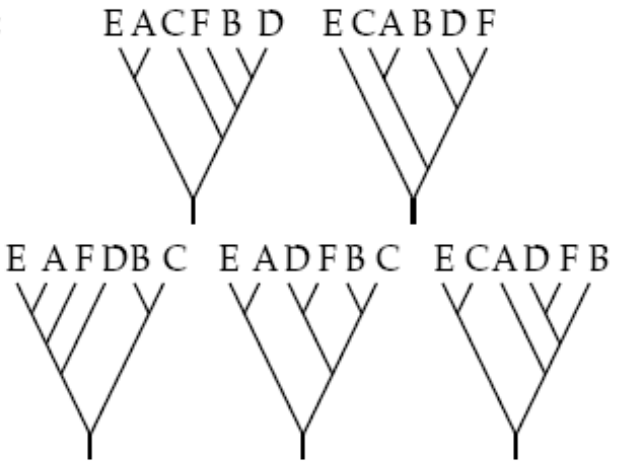
- it will give a lower limit on the branch length
- then check if 0 is in the 95% interval, we would assert the existence of the branch
- branch-length tests available (see practicals)

A simpler solution is to count how many times a particular branch appears in the list of trees estimated by bootstrap.

A majority-rule consensus tree containing clades appearing in more than 50% of them can then be built

Simple example

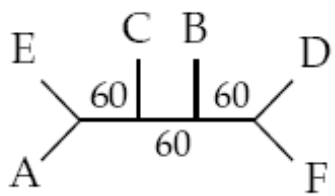
Trees:



Number of times each partition of species is found:

AE | BCDF	3
ACE | BDF	3
ACEF | BD	1
AC | BDEF	1
AEF | BCD	1
ADEF | BC	2
ABDF | EC	1
ABCE | DF	3

Majority–rule consensus tree of the unrooted trees:



Multiple tests

We don't (or rarely) know in advance which group interest us.

If we look for the most supported group on the tree and report its p-value, we have a "multiple-tests" problem

- if no significant evidence for existence of any groups on a tree
- 5% of branches are expected to be above 0.95
- so one out of every 20 branches of a tree would be significant
- furthermore, branches are not independent...

The p-value cannot be interpreted as statistical test.

Multiple tests

We don't (or rarely) know in advance which group interest us.

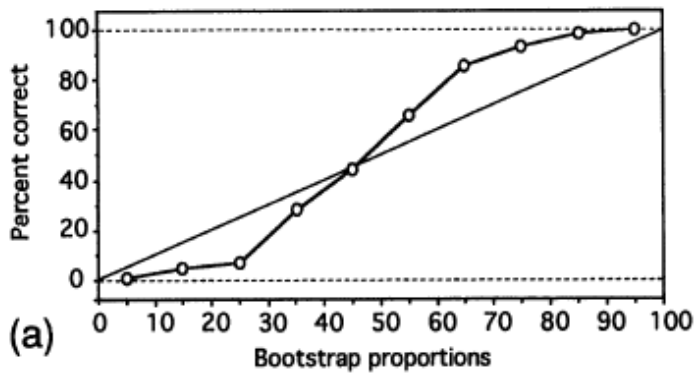
If we look for the most supported group on the tree and report its p-value, we have a "multiple-tests" problem

- if no significant evidence for existence of any groups on a tree
- 5% of branches are expected to be above 0.95
- so one out of every 20 branches of a tree would be significant
- furthermore, branches are not independent...

The p-value cannot be interpreted as statistical test.

Biases in bootstrap

Estimated p-value is conservative:



Hillis and Bull (1993)

One source of conservatism

- with bootstrap, we make statement about branch lengths μ
- then we reduce that to statements about tree topology, i.e. $\mu > 0$ or $\mu < 0$

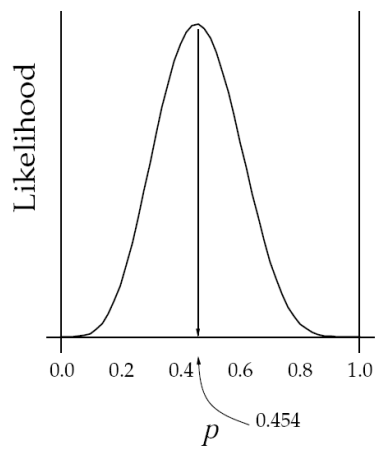
- generalisation of Hillis and Bull (1993) results is not clear

How to test models in phylogenetics?

Uncertainty assessment

Likelihood does not only allow to make point estimate of the topology and branch length, it also gives information about the uncertainty of our estimate.

It is possible to use the likelihood curve to test hypothesis and to make interval estimates.



Asymptotically (i.e. when the number of data point tend towards ∞), the ML estimate $\hat{\theta}$ is normally distributed around its true value θ_0.

Likelihood ratio test

At the maximum $\hat{\theta}$

$$S(\hat{\theta}) = \frac{\partial lnL(\hat{\theta})}{\partial \theta} = 0$$

$$I(\hat{\theta}) = \frac{\partial^2 lnL(\hat{\theta})}{\partial \theta^2} < 0$$

Then the log likelihood of the true parameter value can be linked with the maximum value estimated by a Taylor series:

$$lnL(\theta_0) = lnL(\hat{\theta}) + S(\hat{\theta})(\theta_0 - \hat{\theta}) - \frac{1}{2} I(\hat{\theta})(\theta_0 - \hat{\theta})^2$$

Subtracting $lnL(\hat{\theta})$ from both sides, setting $I(\hat{\theta}) = 1/\sigma^2$ and rearranging leads to

$$-2\big[lnL(\theta_0) - lnL(\hat{\theta})\big] \approx \frac{(\theta_0 - \hat{\theta})^2}{\sigma^2} = \chi_1^2$$

For p parameters, twice the difference in likelihood is χ_p^2 distributed.

Likelihood ratio test

At the maximum $\hat{\theta}$

$$S(\hat{\theta}) = \frac{\partial lnL(\hat{\theta})}{\partial \theta} = 0$$

$$I(\hat{\theta}) = \frac{\partial^2 lnL(\hat{\theta})}{\partial \theta^2} < 0$$

Then the log likelihood of the true parameter value can be linked with the maximum value estimated by a Taylor series:

$$lnL(\theta_0) = lnL(\hat{\theta}) + S(\hat{\theta})(\theta_0 - \hat{\theta}) - \frac{1}{2} I(\hat{\theta})(\theta_0 - \hat{\theta})^2$$

Subtracting $lnL(\hat{\theta})$ from both sides, setting $I(\hat{\theta}) = 1/\sigma^2$ and rearranging leads to

$$-2\big[lnL(\theta_0) - lnL(\hat{\theta})\big] \approx \frac{(\theta_0 - \hat{\theta})^2}{\sigma^2} = \chi_1^2$$

For p parameters, twice the difference in likelihood is χ_p^2 distributed.

Likelihood ratio test

At the maximum $\hat{\theta}$

$$S(\hat{\theta}) = \frac{\partial lnL(\hat{\theta})}{\partial \theta} = 0$$

$$I(\hat{\theta}) = \frac{\partial^2 lnL(\hat{\theta})}{\partial \theta^2} < 0$$

Then the log likelihood of the true parameter value can be linked with the maximum value estimated by a Taylor series:

$$lnL(\theta_0) = lnL(\hat{\theta}) + S(\hat{\theta})(\theta_0 - \hat{\theta}) - \frac{1}{2} I(\hat{\theta})(\theta_0 - \hat{\theta})^2$$

Subtracting $lnL(\hat{\theta})$ from both sides, setting $I(\hat{\theta}) = 1/\sigma^2$ and rearranging leads to

$$-2\big[lnL(\theta_0) - lnL(\hat{\theta})\big] \approx \frac{(\theta_0 - \hat{\theta})^2}{\sigma^2} = \chi_1^2$$

For p parameters, twice the difference in likelihood is χ_p^2 distributed.

Nested models

Assumptions of the likelihood ratio test

- null hypothesis should be in the interior space that contains the alternative hypotheses
- if q parameters have been constrained, they must be able to vary in both sense
  - if L_0 restricts one parameter to the end of its range, distribution of twice log likelihood ratio has half its mass at 0 and the other half in the usual χ^2 distribution
  - halve the tail probability obtained with usual χ^2

Valid only asymptotically

- should be close enough to the true values
- true with very large amounts of data

Nested models

Assumptions of the likelihood ratio test

- null hypothesis should be in the interior space that contains the alternative hypotheses
- if q parameters have been constrained, they must be able to vary in both sense
  - if L_0 restricts one parameter to the end of its range, distribution of twice log likelihood ratio has half its mass at 0 and the other half in the usual χ^2 distribution
  - halve the tail probability obtained with usual χ^2

Valid only asymptotically

- should be close enough to the true values
- true with very large amounts of data

Akaike Information Criterion

More general model will always have higher likelihood than restricted models.
So, choosing model with highest likelihood will lead to one that is unnecessarily complex.

We should therefore compromise goodness of fit with complexity of model.

AIC for hypothesis i with p_i parameters:

$$AIC_i = -2lnL_i + 2p_i$$

Hypothesis with the lowest AIC is preferred.

Akaike Information Criterion

More general model will always have higher likelihood than restricted models.
So, choosing model with highest likelihood will lead to one that is unnecessarily complex.

We should therefore compromise goodness of fit with complexity of model.

AIC for hypothesis i with p_i parameters:

$$AIC_i = -2lnL_i + 2p_i$$

Hypothesis with the lowest AIC is preferred.

Adding more complexity

All the different models seen so far are **special** cases of the GTR+Γ+I model

- setting Γ and I to 0 leads to GTR
- setting transversion rates to β and transition rates to α leads to F84
- setting $\beta = \alpha$ leads to K2P
- setting all nucleotide frequencies to $\frac{1}{4}$ lead to JC69

Examples

Possible comparisons

- GTR+Γ vs GTR
  - $2 \times [lnL_{\text{GTR}+\Gamma} - lnL_{\text{GTR}}]$
  - difference in df $= 1$
- GTR+Γ vs F84+Γ
  - $2 \times [lnL_{\text{GTR}+\Gamma} - lnL_{\text{F84}+\Gamma}]$
  - difference in df $= 4$
- F84+Γ vs JC69
  - $2 \times [lnL_{\text{F84}+\Gamma} - lnL_{\text{J69}}]$
  - difference in df $= 5$

But not JC69+Γ vs GTR because they are not nested!

Are two topologies different?

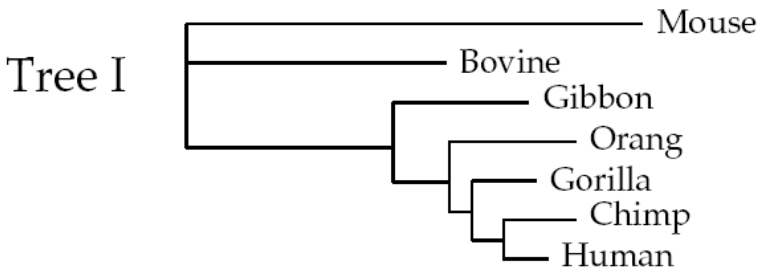
LRT or AIC not possible on topologies because they are discrete entities, and not quantitative parameters.

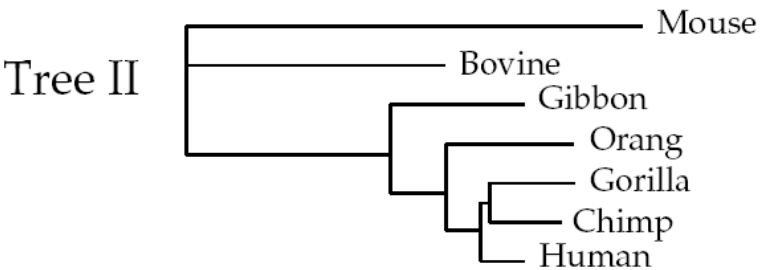
We therefore have to rely on **paired-sites** tests

- expected *lnL* is average *lnL* per site as number of sites grows without limit
- if sites are independent and two trees have equal expected *lnL*, then differences in *lnL* at each site is drawn independently with expectation zero
- statistical test of mean of these differences is zero
- valid for likelihood and parsimony

Example

232-sites mtDNA for 7 mammals





Tree \ Site	1	2	3	4	5	6	…	231	232	ln L
I	−2.971	−4.483	−5.673	−5.883	−2.691	−8.003	…	−2.971	−2.691	−1405.61
II	−2.983	−4.494	−5.685	−5.898	−2.700	−7.572	…	−2.987	−2.705	−1408.80
Diff	+0.012	+0.111	+0.013	+0.015	+0.010	−0.431	…	+0.012	+0.010	+3.19

Felsenstein, 2004

Possible tests

Different form of tests possible

- *z* assumes the differences at each site are normally distributed and estimate the variance of differences of the scores
- **KH** use bootstrap sampling to infer distribution of sum of differences of scores and see whether 0 lay in the tails of distribution

Results

- *z* sum of $lnL = 3.18$, $\sigma^2 = 0.04$, so variance of sum of differences is 11.31; z statistic is $3.18/3.36 = 0.94$ with p-value of 0.34
- **KH** 10,000 bootstrap samples of sites; 8,326 favored tree II, which yields a one-tailed probability of 0.16, and a 0.33 two-tailed probability

Multiple tests, again

If we want to test more than two trees

- compare each tree to best tree
- accept all trees that cannot be rejected by KH test

- multiple tests setting, but no reduction of nominal rejection level possible
- need to correct for all different ways the data can vary, ways that support different trees

When two trees are compared, but one of them is the actual best tree, we should do a one-tailed test.

Multiple tests, again

If we want to test more than two trees

- compare each tree to best tree
- accept all trees that cannot be rejected by KH test

- multiple tests setting, but no reduction of nominal rejection level possible
- need to correct for all different ways the data can vary, ways that support different trees

When two trees are compared, but one of them is the actual best tree, we should do a one-tailed test.

SH test

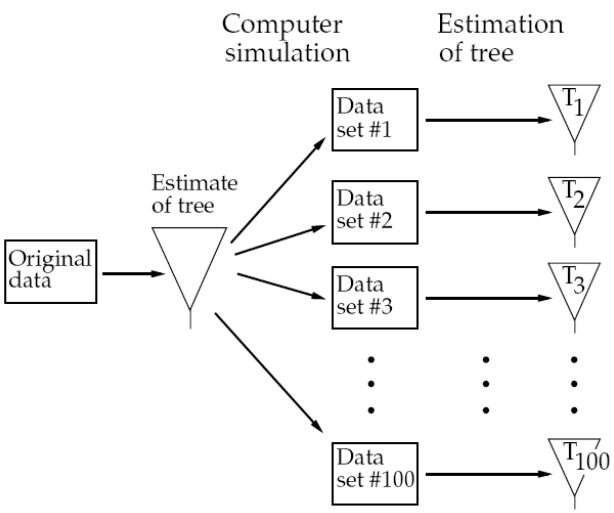
Resampling technique that approximately corrects for testing multiple trees

1. make R bootstrap of the N sites
2. for each tree, normalize resampled *lnL* so they have same expectation
3. for jth bootstrap, calculate $\tilde{S}_{ij}$ for ith tree how far normalized value is below maximum across all trees for that replicate
4. for each tree i, the tail probability is proportion of bootstrap replicates in which $\tilde{S}_{ij}$ is less than the actual difference between ML and *lnL* of that tree

Resampling build a "least-favorable" case in which the trees show some patterns of covariation of site as in actual data but do not differ in overall *lnL*.

One limitation: assume that all proposed trees are possibly equal in likelihood

Parametric bootstrap



Use computer simulations to create pseudorandom data sets

Advantages and disadvantages

- can sample from the desired distribution, even with small data sets
- flexible hypothesis testing framework
- close reliance on the correctness of the model of evolution

Measuring, searching and assessing trees all in one!

Bayesian methods

The likelihood calculation is used as well by Bayesian methods. However, another component is added to the method: the **prior distributions**.

Before observing any data, each parameter will be assigned a prior distribution

- topologies
- branch lengths
- each parameter of the model of evolution

The prior distributions are then combined with the likelihood of the data to give the **posterior distribution**.

This is a highly attractive quantity because it computes what we most need: the probabilities of different hypotheses in the light of the data.

Bayesian methods

The likelihood calculation is used as well by Bayesian methods. However, another component is added to the method: the **prior distributions**.

Before observing any data, each parameter will be assigned a prior distribution

- topologies
- branch lengths
- each parameter of the model of evolution

The prior distributions are then combined with the likelihood of the data to give the **posterior distribution**.

This is a highly attractive quantity because it computes what we most need: the probabilities of different hypotheses in the light of the data.

Bayesian methods

The likelihood calculation is used as well by Bayesian methods. However, another component is added to the method: the **prior distributions**.

Before observing any data, each parameter will be assigned a prior distribution

- topologies
- branch lengths
- each parameter of the model of evolution

The prior distributions are then combined with the likelihood of the data to give the **posterior distribution**.

This is a highly attractive quantity because it computes what we most need: the probabilities of different hypotheses in the light of the data.

Bayes theorem

To combine all this together, we use the Bayes theorem

$$Prob(T|D) = \frac{Prob(T \cup D)}{Prob(D)}$$

where $Prob(T \cup D) = Prob(T)Prob(D|T)$

so that

$$Prob(T|D) = \frac{Prob(T)Prob(D|T)}{Prob(D)}$$

Bayes theorem

To combine all this together, we use the Bayes theorem

$$Prob(T|D) = \frac{Prob(T \cup D)}{Prob(D)}$$

where $Prob(T \cup D) = Prob(T)Prob(D|T)$

so that

$$Prob(T|D) = \frac{Prob(T)Prob(D|T)}{Prob(D)}$$

Bayes theorem

To combine all this together, we use the Bayes theorem

$$Prob(T|D) = \frac{Prob(T \cup D)}{Prob(D)}$$

where $Prob(T \cup D) = Prob(T)Prob(D|T)$

so that

$$Prob(T|D) = \frac{Prob(T)Prob(D|T)}{Prob(D)}$$

Normalizing constant

The denominator $Prob(D)$ is the sum of the numerator $Prob(T)Prob(D|T)$ over all possible hypotheses T.

This quantity is needed to normalize the probabilities of all T so that they add up to 1.

This leads to

$$Prob(T|D) = \frac{Prob(T)Prob(D|T)}{\sum_T Prob(T)Prob(D|T)}$$

In words:

$$\text{posterior probability} = \frac{\text{prior probability} \times \text{likelihood}}{\text{normalizing constant}}$$

Normalizing constant

The denominator $Prob(D)$ is the sum of the numerator $Prob(T)Prob(D|T)$ over all possible hypotheses T.

This quantity is needed to normalize the probabilities of all T so that they add up to 1.

This leads to

$$Prob(T|D) = \frac{Prob(T)Prob(D|T)}{\sum_T Prob(T)Prob(D|T)}$$

In words:

$$\text{posterior probability} = \frac{\text{prior probability} \times \text{likelihood}}{\text{normalizing constant}}$$

Estimate normalizing constant

Posterior distribution expression has a denominator, i.e. $\sum_T Prob(T)Prob(D|T)$, that is often impossible to compute.

Fortunately, samples from the posterior distribution can be drawn using a Markov chain that does not need to know the denominator

- draw a random sample from posterior distribution of trees
- becomes possible to make probability statements about true tree
- e.g. if 96% of the samples from posterior distribution have (human,chimp) as monophyletic group, probability of this group is 96%

Estimate normalizing constant

Posterior distribution expression has a denominator, i.e. $\sum_T Prob(T)Prob(D|T)$, that is often impossible to compute.

Fortunately, samples from the posterior distribution can be drawn using a Markov chain that does not need to know the denominator

- draw a random sample from posterior distribution of trees
- becomes possible to make probability statements about true tree
- e.g. if 96% of the samples from posterior distribution have (human,chimp) as monophyletic group, probability of this group is 96%

Makov chain Monte Carlo

Idea: to wander randomly through tree space by sampling trees until we settle down into an equilibrium distribution of trees that has the desired distribution, i.e. posterior distribution.

- Markov chain: the new proposed tree will depend only on the previous one
- to reach equilibrium distribution, the Markov chain must be
  - aperiodic – no cycles should be present in the Markov chain
  - irreducible – every trees must be accessible from any other tree
  - probability of proposing T_j when we are at T_i is the same as probability of proposing T_i when we are at T_j
- the Markov chain has no end

MCMC in practice

Metropolis algorithm

- start with a random tree T_i
- select a new tree T_j by modifying T_i in some way
- compute

$$R = \frac{Prob(T_j|D)}{Prob(T_i|D)}$$

The beauty of MCMC: the normalizing constant being the same, it simplifies

$$R = \frac{Prob(T_j)Prob(D|T_j)}{Prob(T_i)Prob(D|T_i)}$$

- if $R \geq 1$, accept T_j
- if $R < 1$, draw a random number n between $[0, 1]$ and accept T_j if $R > n$, otherwise keep T_i

MCMC in practice

Metropolis algorithm

- start with a random tree T_i
- select a new tree T_j by modifying T_i in some way
- compute

$$R = \frac{Prob(T_j|D)}{Prob(T_i|D)}$$

The beauty of MCMC: the normalizing constant being the same, it simplifies

$$R = \frac{Prob(T_j)Prob(D|T_j)}{Prob(T_i)Prob(D|T_i)}$$

- if $R \geq 1$, accept T_j
- if $R < 1$, draw a random number n between $[0, 1]$ and accept T_j if $R > n$, otherwise keep T_i

MCMC in practice

Metropolis algorithm

- start with a random tree T_i
- select a new tree T_j by modifying T_i in some way
- compute

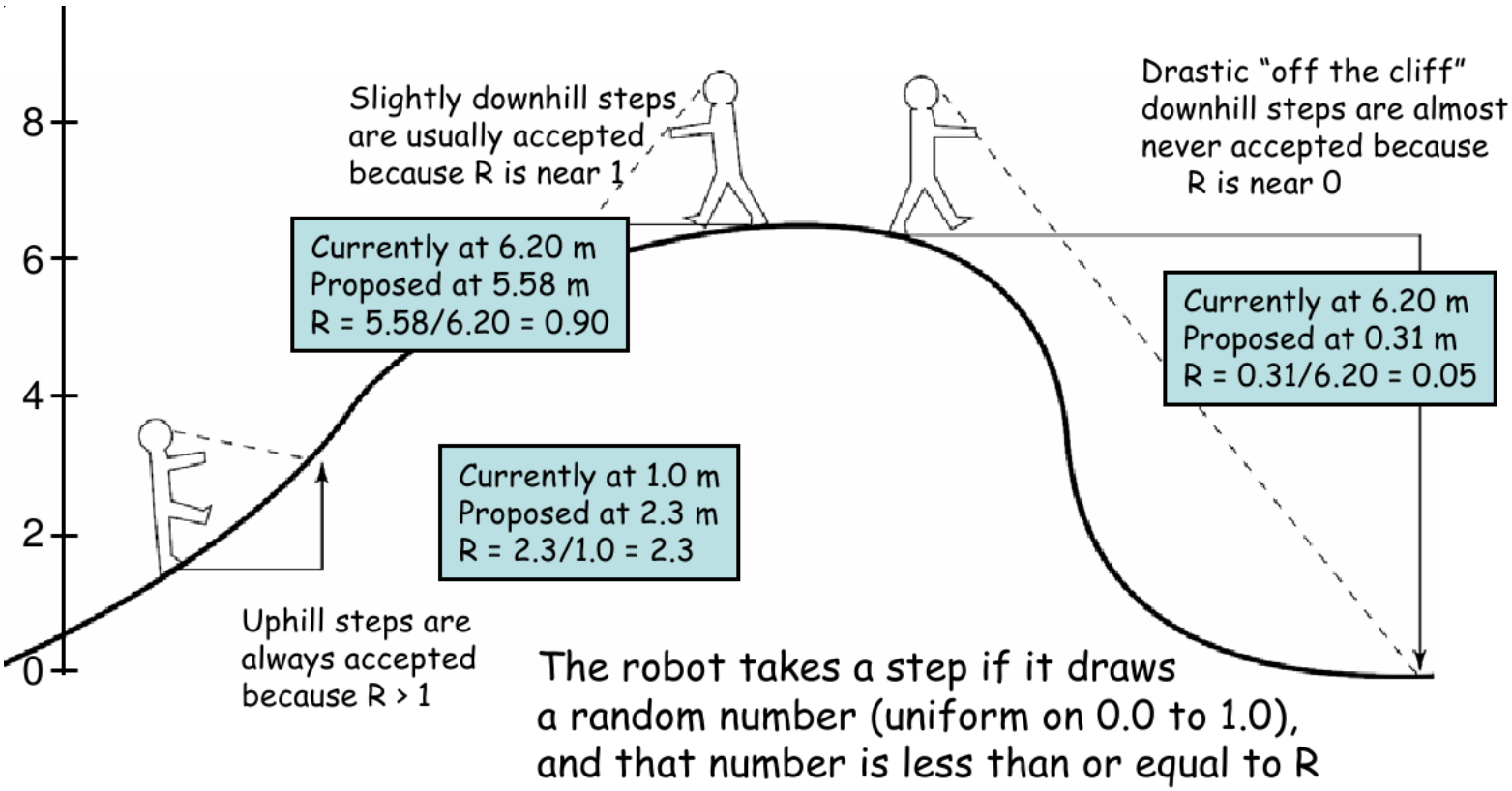
$$R = \frac{Prob(T_j|D)}{Prob(T_i|D)}$$

The beauty of MCMC: the normalizing constant being the same, it simplifies

$$R = \frac{Prob(T_j)Prob(D|T_j)}{Prob(T_i)Prob(D|T_i)}$$

- if $R \geq 1$, accept T_j
- if $R < 1$, draw a random number n between $[0, 1]$ and accept T_j if $R > n$, otherwise keep T_i

Schematic view



P. Lewis, 2006

Putting it all together

- Start with random tree and arbitrary initial values for branch lengths and model parameters
- Each generations, chose at random to propose either:
  - a new tree
  - new branch length
  - new model parameter values
- Either accept or reject the move
- Every k generations, save tree topology, branch lengths and model parameters (i.e. sample the chain)
- After n generations, summarize the sample using histograms, means, credibility intervals, etc.

How to propose a new tree

We could invent any type of proposal distribution to wander through the tree space

- e.g. NNI by selecting a node at random
- should be able to reach all trees from any starting tree
- at least after "sufficient" running, but impossible to know how much running is enough

Should be careful because

- if trees proposed are too different $\Rightarrow$ these trees will be rejected too often
- if trees proposed are too similar $\Rightarrow$ tree space won't be sampled well enough

How to propose a new tree

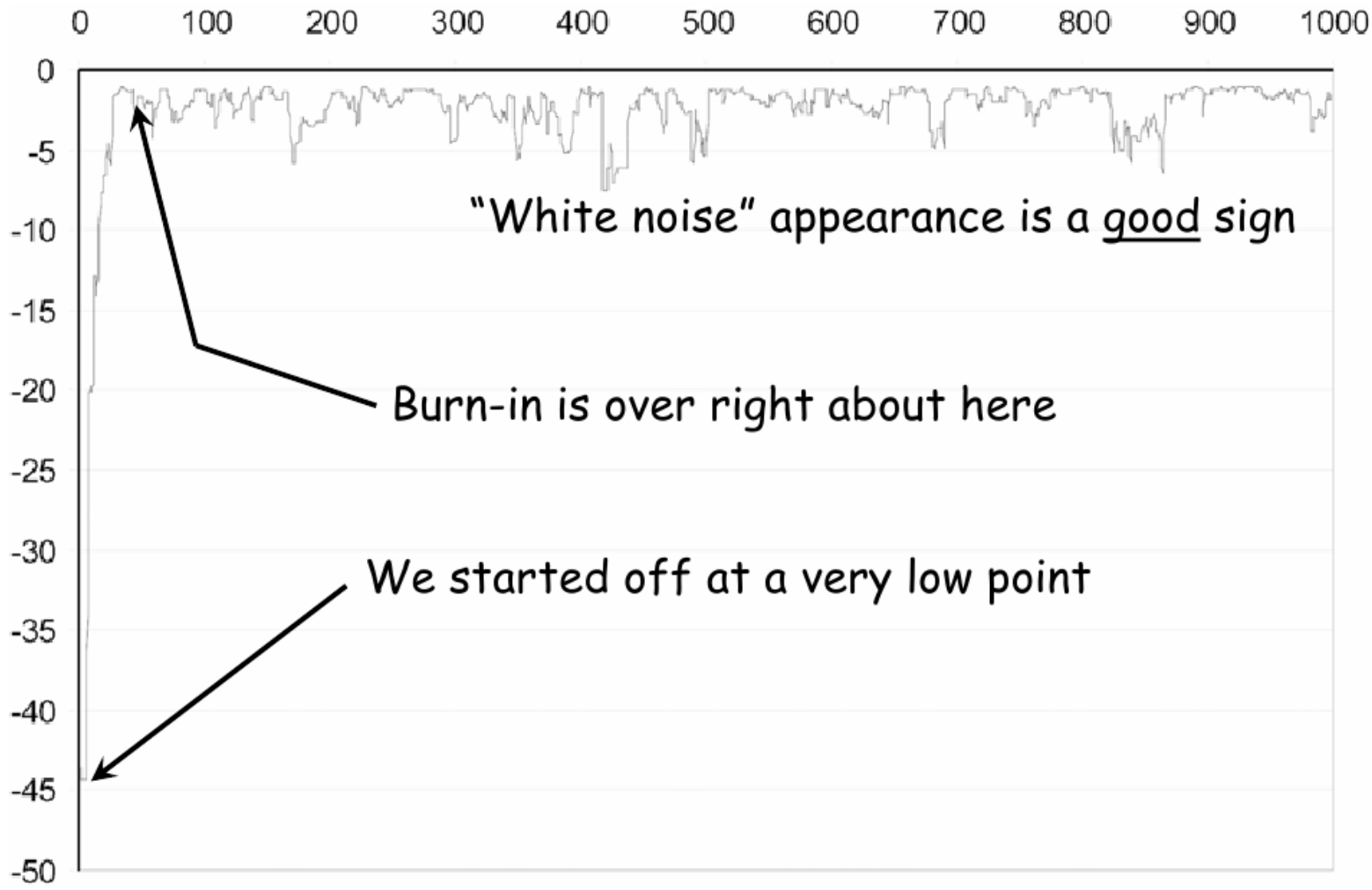
We could invent any type of proposal distribution to wander through the tree space

- e.g. NNI by selecting a node at random
- should be able to reach all trees from any starting tree
- at least after "sufficient" running, but impossible to know how much running is enough

Should be careful because

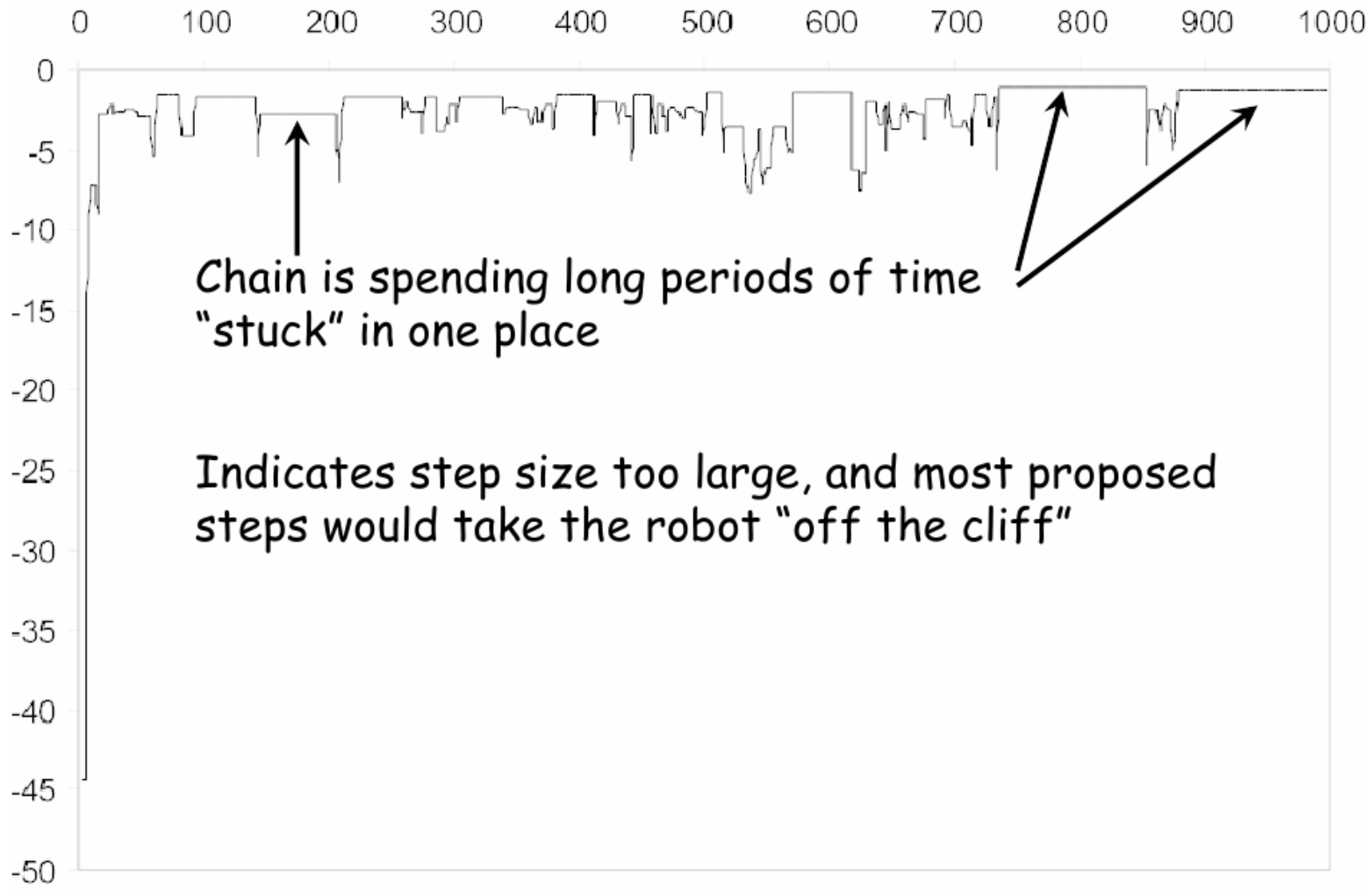
- if trees proposed are too different $\Rightarrow$ these trees will be rejected too often
- if trees proposed are too similar $\Rightarrow$ tree space won't be sampled well enough

MCMC trace plot



P. Lewis, 2006

Slow mixing

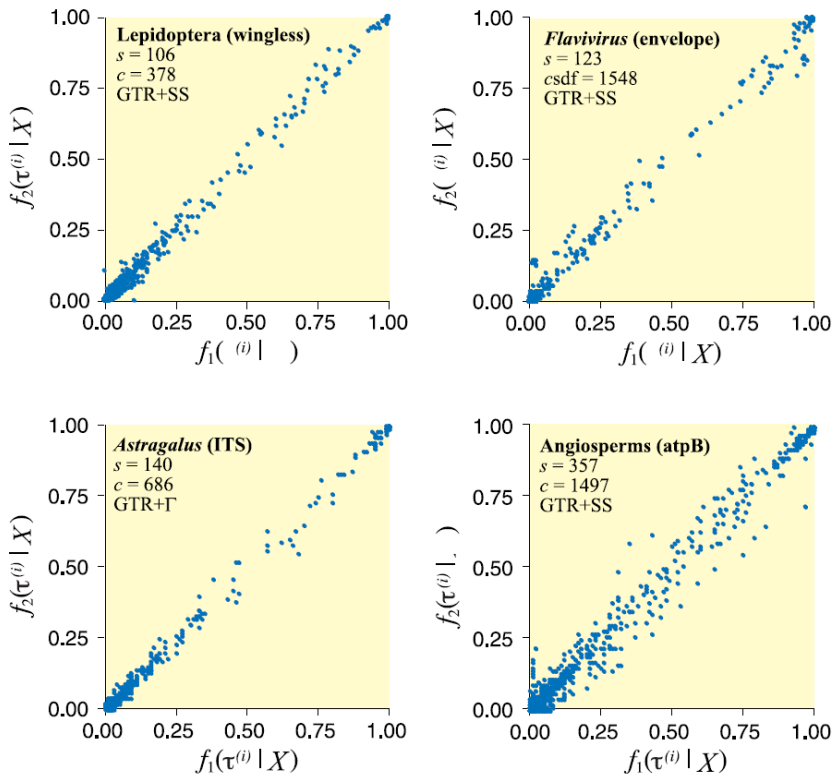


P. Lewis, 2006

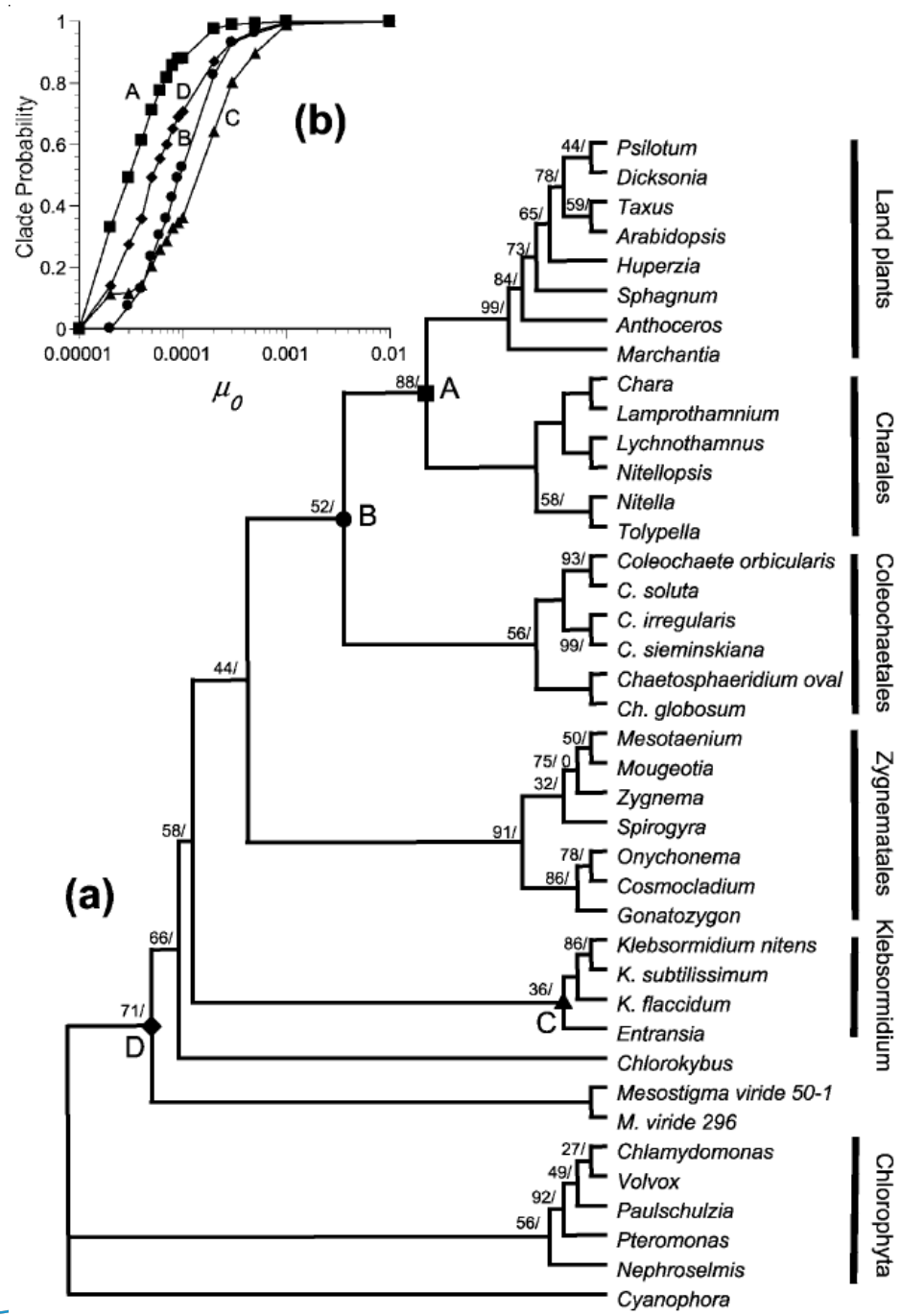
Convergence in MCMC

Greatest practical problem: how long to run a chain to obtain a good approximation of the posterior probabilities

The most important is to check that independent runs lead to the same posterior distribution.

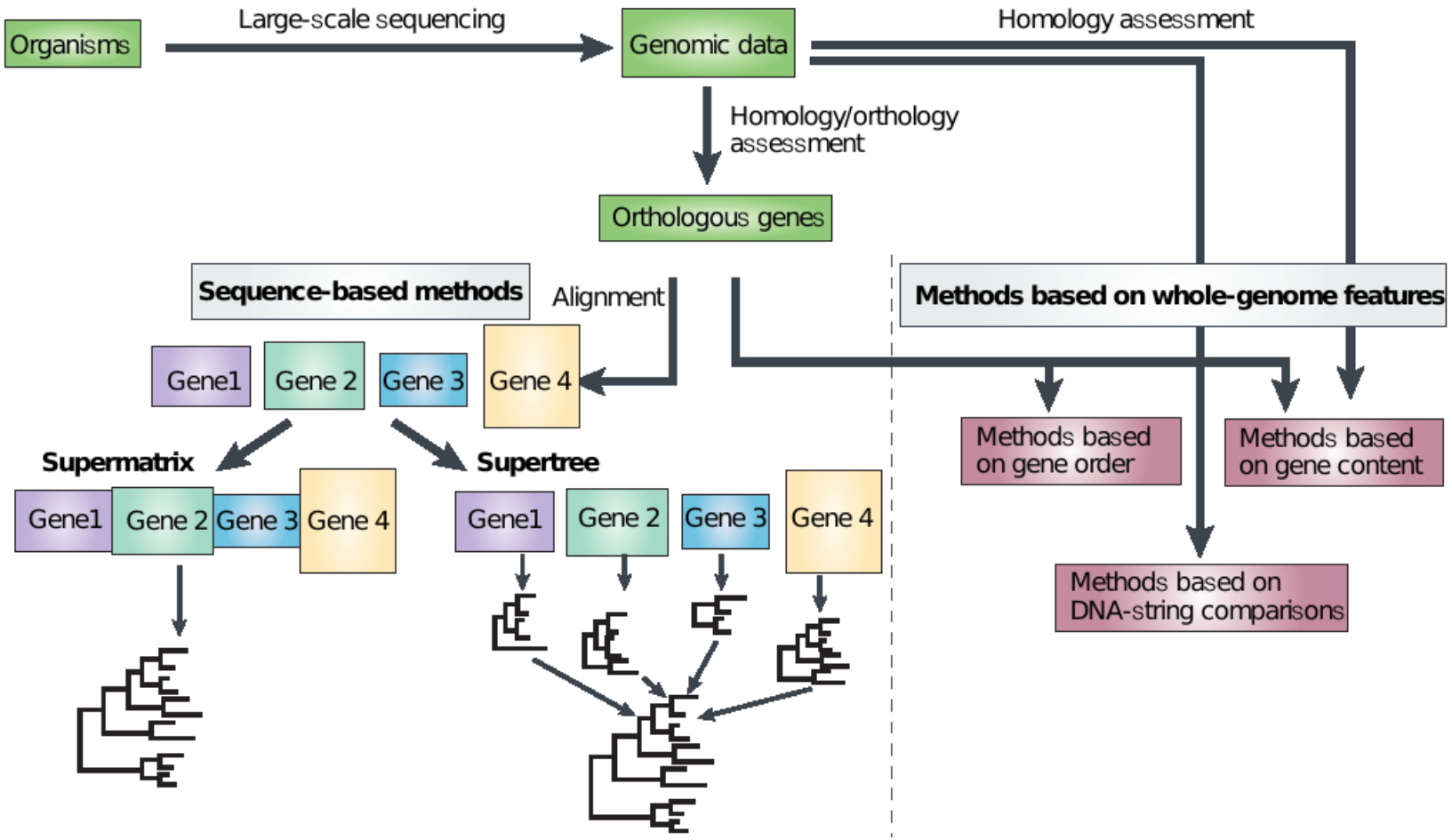


Effect of priors



Yang and Rannala, 2006

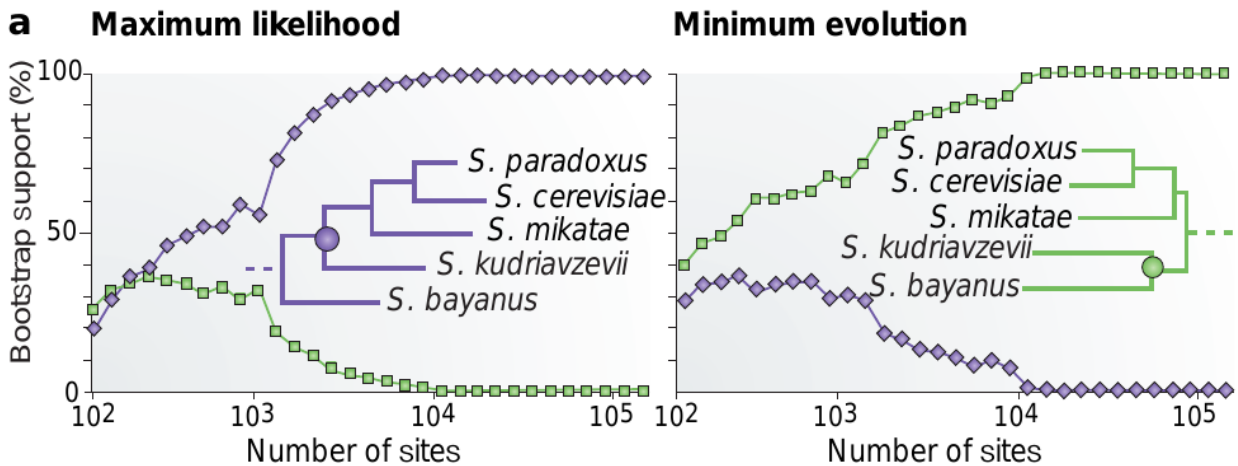
Phylogenomics



Delsuc et al. 2005

Issues: compositional bias

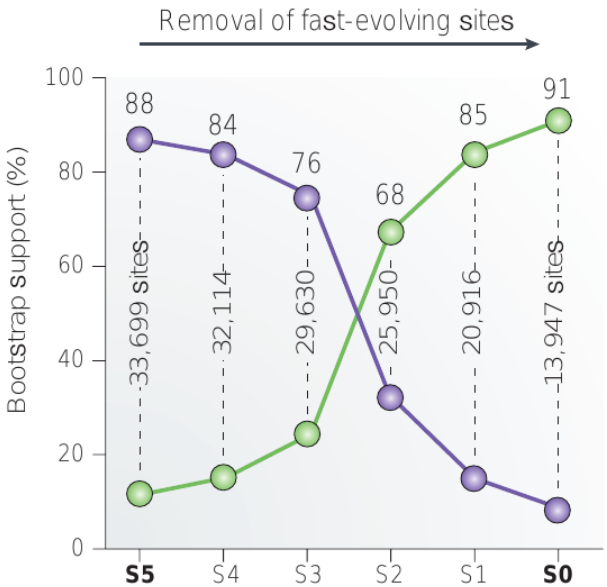
Species and/or genes might have different amounts of GC content.

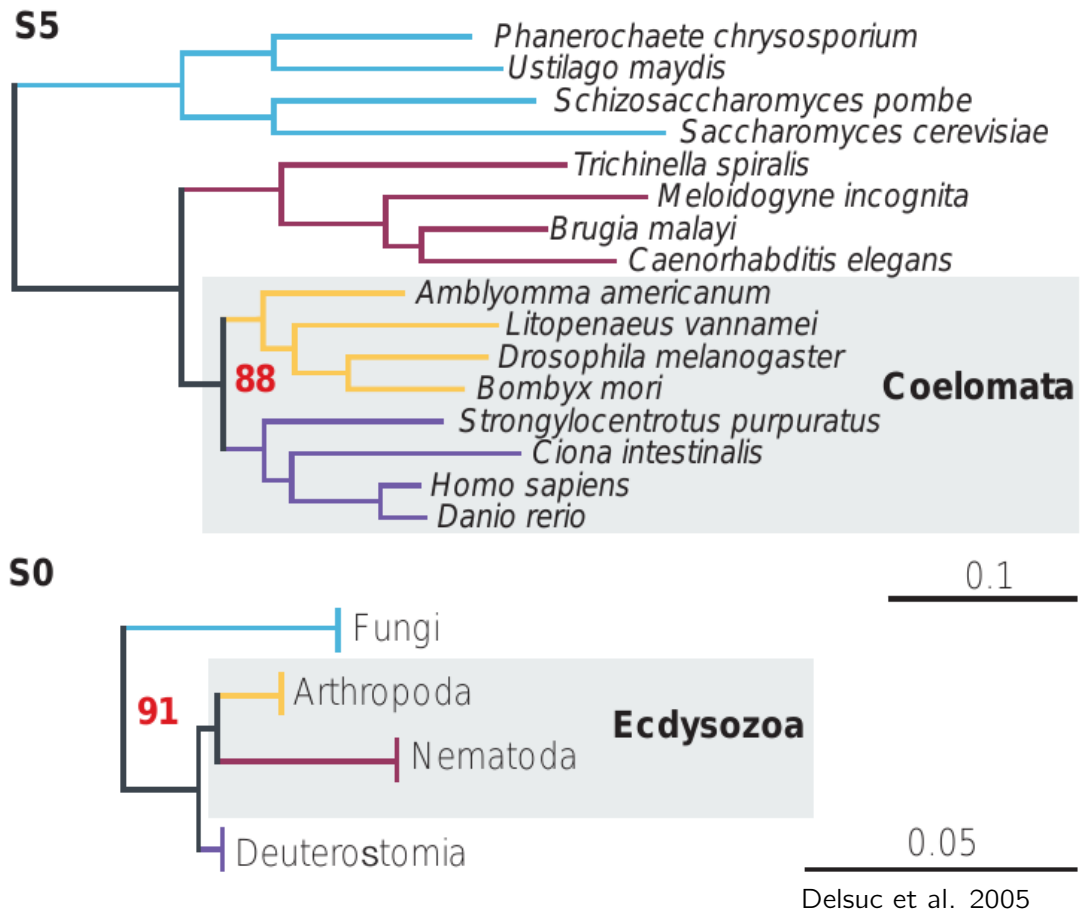


Delsuc et al. 2005

Issues: Rate of evolution and saturation

The rate of evolution of sites and/or genes can affect drastically the resulting tree (even with Γ model of rate variation)





Delsuc et al. 2005

Heterogeneity of change through time

Substitution rate for a site can change over time (heterotachy). This does not violate any assumptions of the Markov process, but can lead to wrong models.

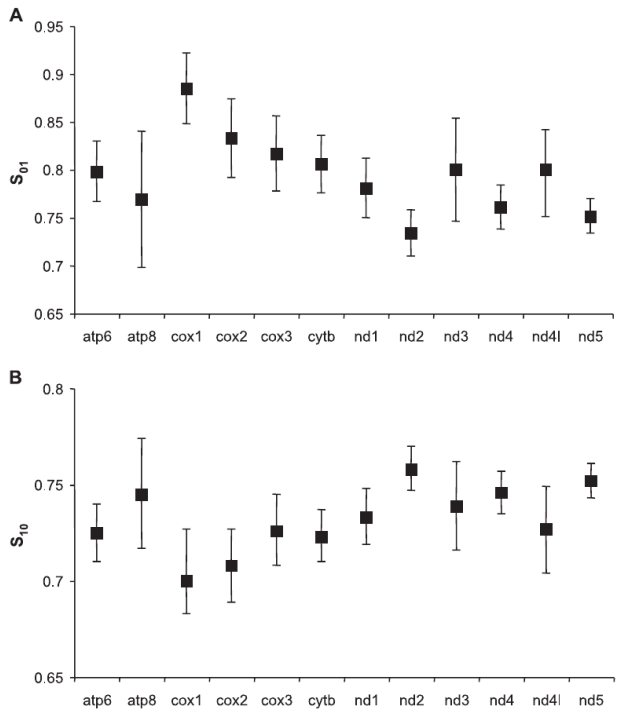
Covarion model, where I is the $4x4$ identity matrix, Q is the instantaneous rate matrix, S is the switch rate from ON (1) to OFF (0):

$$Q = \begin{bmatrix} -S_{01}I & S_{01}I \\ S_{10}I & Q - S_{10}I \end{bmatrix}$$

The equilibrium frequencies are $\pi_{ON} = S_{01}/(S_{01} + S_{10})$ and $\pi_{OFF} = S_{10}/(S_{01} + S_{10})$.

The number of S_{01} and S_{10} can be extended using Dirichlet process to allow variation among sites.

Heterotachy in mitochondrial genes



Zhou et al 2010, MBE

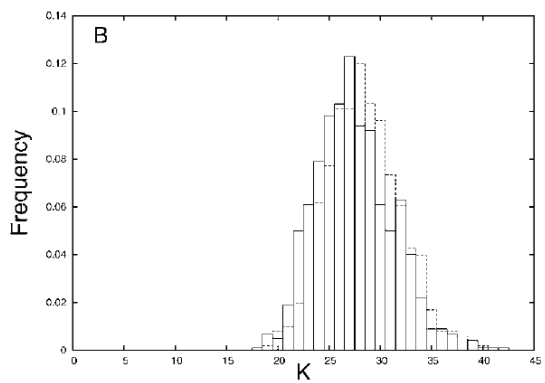
Exact effect on the tree reconstruction are not known clearly outside simulations.

Mixture models

Sites might also evolved under different processes and belong to classes characterized by its own substitution matrix Q and equilibrium frequencies.

CAT model introduces a mixture of K matrices with different equilibrium frequencies π^k.

- it is easy to calculate the total log-likelihood once sites are assigned to a category (site independence)
- but a priori assignment is not known
- complex Bayesian framework to define the number of K between $[1 \ldots N]$ and assign categories to sites using a Dirichlet process



Lartillot and Philippe, 2004

Dependency between sites

Difficult assumption to remove and only specific solutions have been found

- using codon models to account for dependency between codon position
- use sequence transition probabilities (Nasrallah et al 2011)

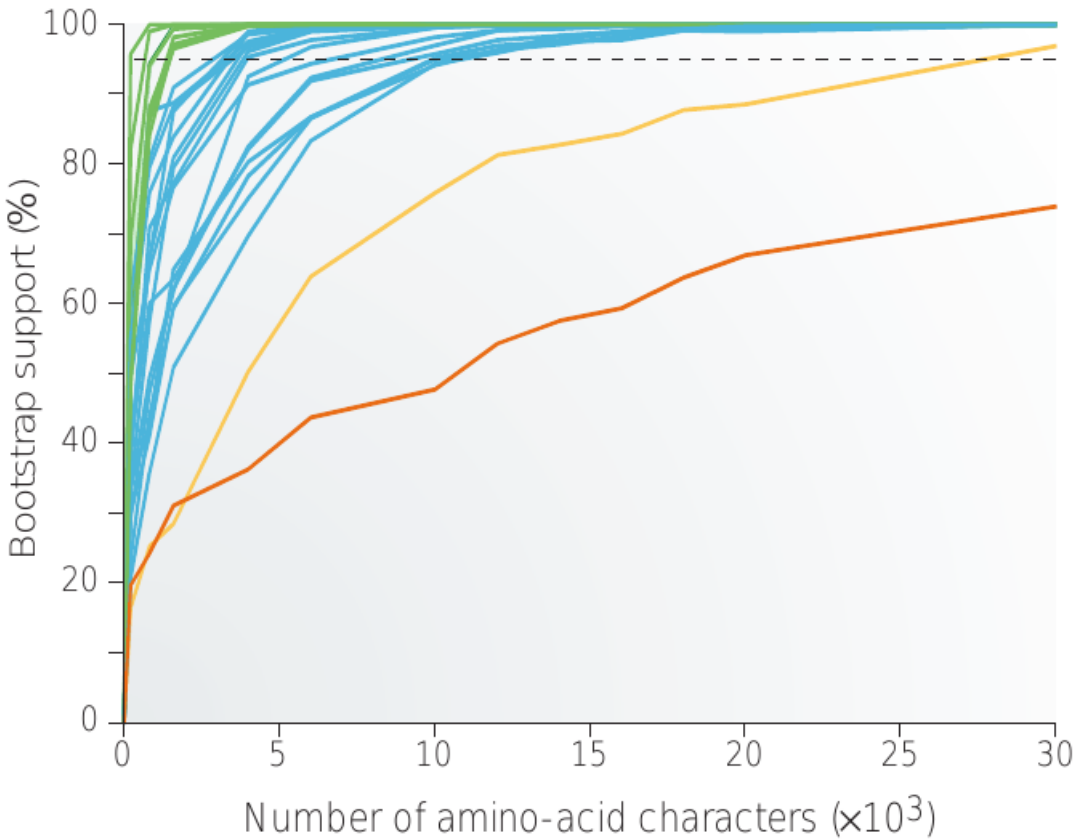
$$Q_{ij} = \begin{cases} 0 & \text{if } x \text{ and } y \text{ differ by two nucleotide positions,} \\ uq_{ij}E(x,y) & \text{if } x \text{ and } y \text{ differ by one nucleotide position,} \\ -\sum_{x \neq y} r_{xy} & \text{if} x = y, \end{cases}$$

- sites in a coevolution profile (Dib et al. 2014)

$$Q_{ij} = \begin{cases} 0 & \text{if i and j differ by two nucleotide positions,} \\ w_1 & \text{if } \{\text{i, j}\} \notin \textit{profile} \text{ and if i differs from j at position 1,} \\ w_2 & \text{if } \{\text{i, j}\} \notin \textit{profile} \text{ and if i differs from j at position 2,} \\ s & \text{if i} \in \textit{profile} \text{ and j} \notin \textit{profile,} \\ d & \text{if i} \notin \textit{profile} \text{ and j} \in \textit{profile} \end{cases}$$

Issues: data size

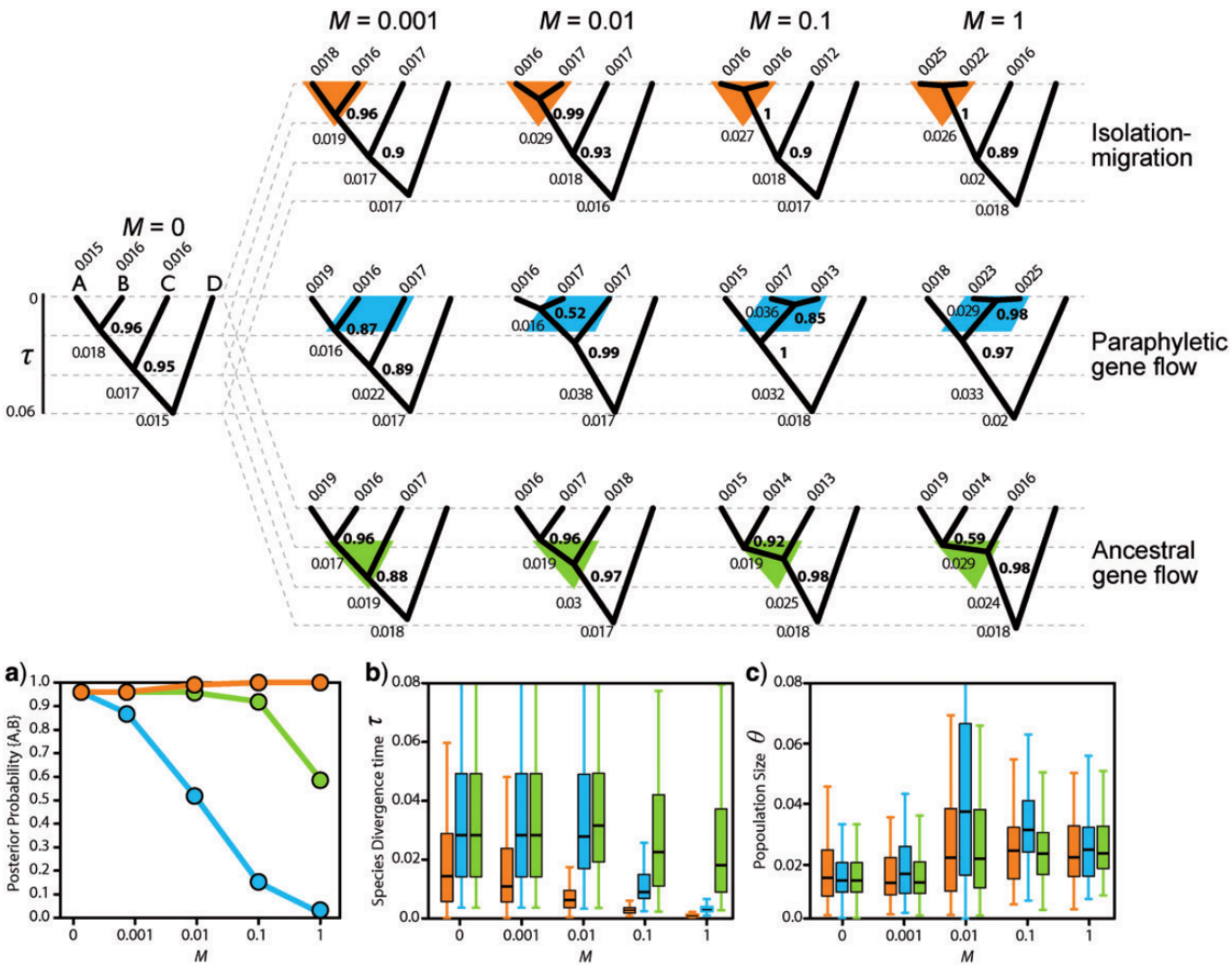
Phylogenetic trees are complex entities and each node requires different amount and types of data



Desluc et al. 2005

Issues: gene tree vs species tree

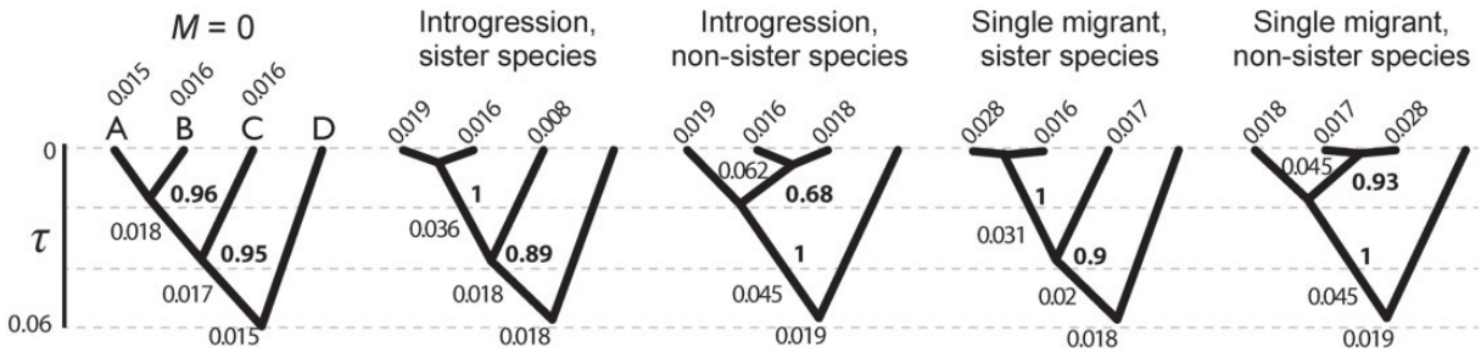
Effects of migration events on species tree reconstruction



Leachee et al. 2014

Issues: gene tree vs species tree

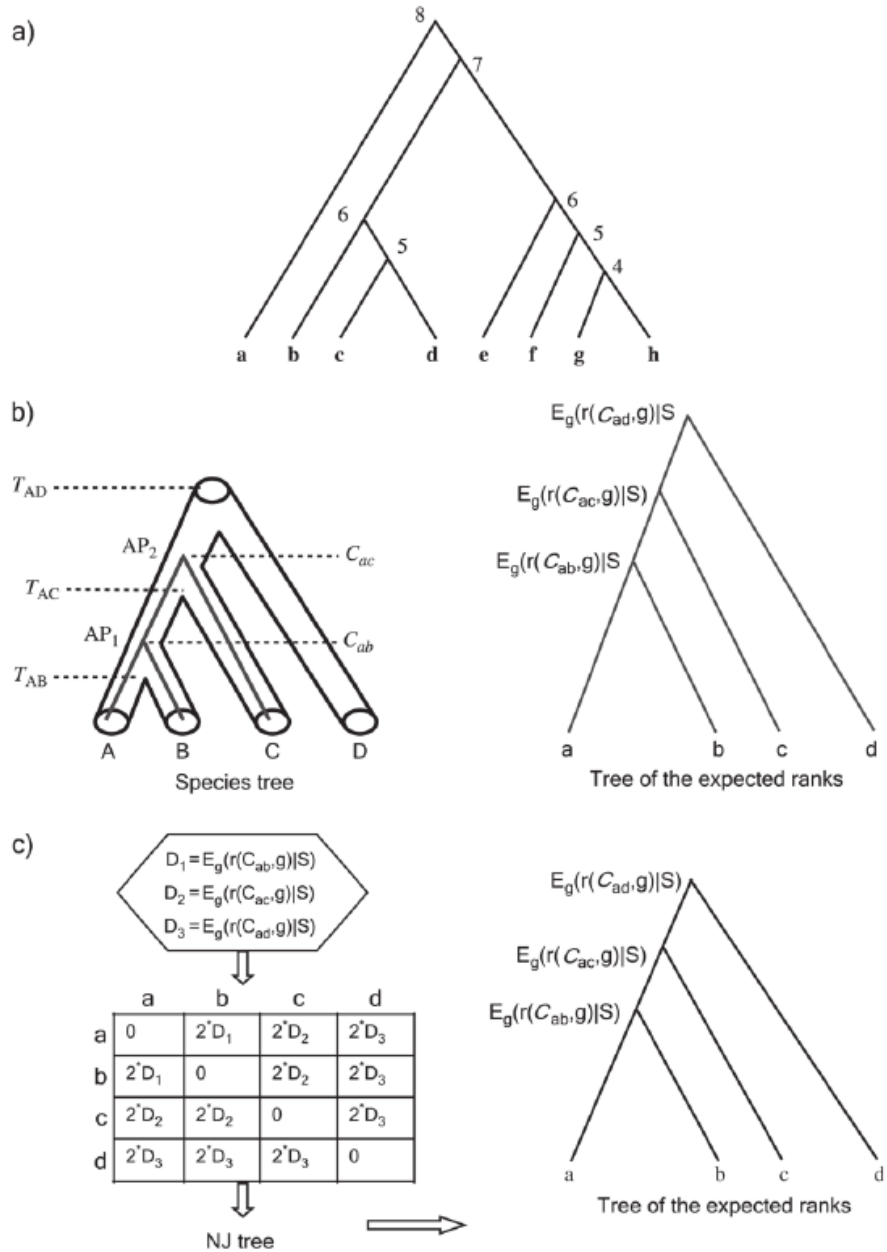
Single locus introgression or migration



Leachee et al. 2014

Average ranks of coalescent

STAR approach (Liu et al. 2009): use average ranks of coalescent from gene trees



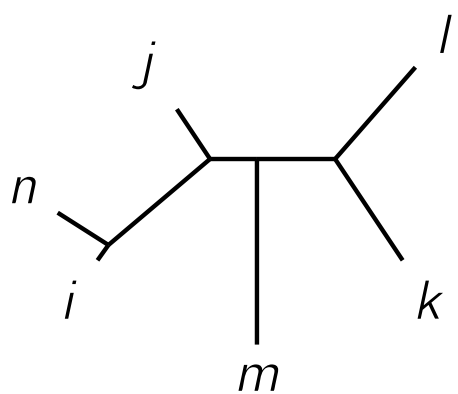
Species trees from quartets

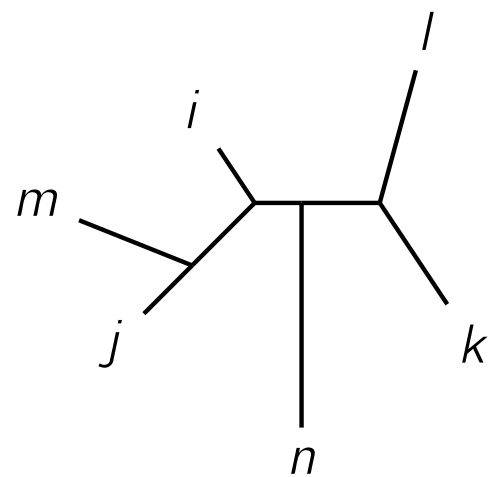
ASTRAL (Mirarab et al. 2014) maximizes

$$\sum_{q \in Q(T)} w(q, \mathcal{T})$$

where

- $\mathcal{T}$ is a set of unrooted gene trees
- $Q(T)$ is the set of quartet trees from T
- $w(q, \mathcal{T})$ is the number of trees in $\mathcal{T}$ having quartet q





Bayesian gene tree / species tree estimation

BEST approach (Liu et al. 2008): two-steps MCMC algorithm

$$P(S|D) = \int_G P(G|D)P(S|G)dG$$

- posterior distribution of gene trees is estimated in the first MCMC
- it is then used to estimate the posterior distribution of the species tree in the second MCMC

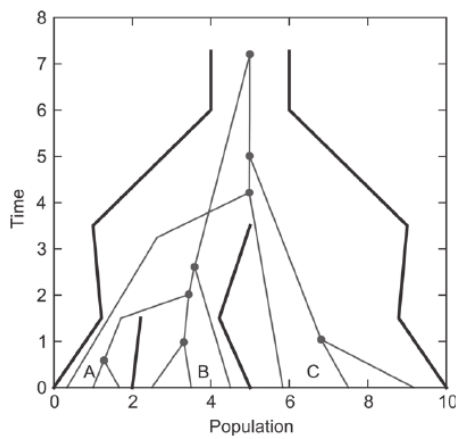
However, $P(G)$, the prior on gene trees is unknown in the first MCMC. They approximate $\int_S P(G|S)P(S)dS$ using harmonic means.

Bayesian gene tree / species tree estimation

*BEAST approach (Heled and Drummond, 2010): you can use a prior on species trees to compute the multi-species coalescent, by making some assumptions on the coalescent parameters

$$P(S|D) = \int_G \left(\prod_{i=1}^{n} P(d_i|g_i) P(g_i|S) \right) P(S) dG$$

- $P(d_i|g_i)$ is the pruning algorithm that we already saw
- $P(g_i|S)$ is the multi-species coalescent of g embedded in S: $\prod_{b \in S} P(L_b(g)|N_b(t))$



Intense area of research

So far, there are many limitations in terms of the type of coalescent model that can be incorporated

- no recombination
- limited population dynamics

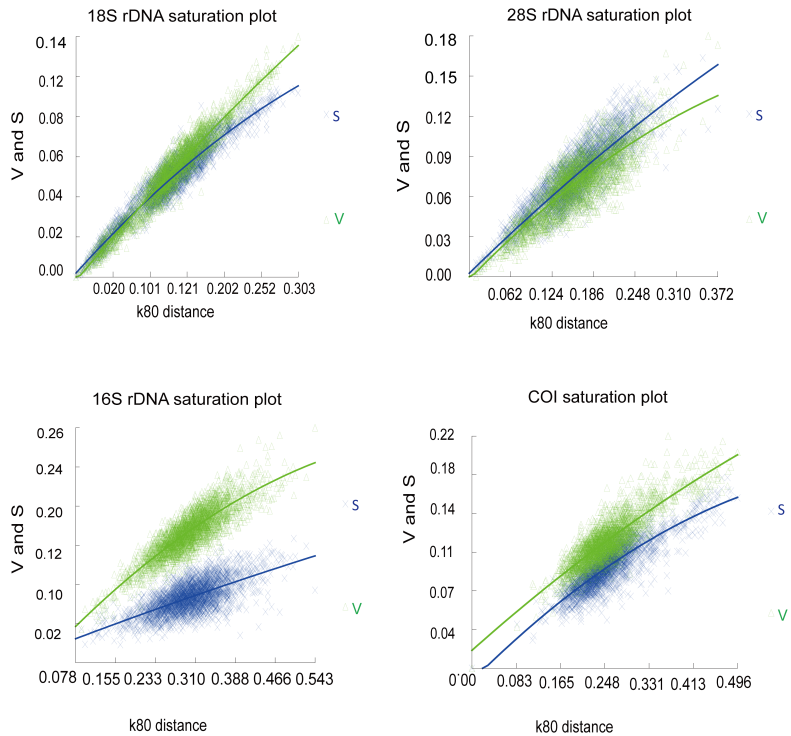
Many different approaches have been published recently

- STAR, BEST, *BEAST
- BuCKy
- PhyloDOG
- etc...

When nucleotides are not good...

Nucleotide models are widely used, but

- can become saturated if sequences are too divergent



- might not encapsulate the true processes behind molecular evolution (e.g. Seo and Kishino, 2009).

For protein-coding genes, codon or amino-acid models can be useful.

"Empirical" models

Dayhoff et al. (1979) model

- amino acid changes inferred by parsimony in 71 sets of closely related proteins.
- substitution probability matrix compiled for different amounts of evolutionary divergence.
- PAM matrices
  - PAM001: probability of changing amino acid along a branch short enough that 1% of the aa are expected to have changed.
  - PAM250 = PAM001^{250}

Other models proposed:

- e.g. JTT use either more or different data or different ways of obtaining PAM matrices.
- WAG (Whelan and Goldman 2001) or LG (Le and Gascuel 2008) are GTR-like amino-acid models

Probabilistic codon models

Markov model for codons have been proposed

- based on standard nucleotide models (HKY85)
- probability of rejection of change increasing as the chemical properties of the two amino acids becomes different [e.g. Yang and Goldman (1994); Muse and Gaut (1994)].
- but number of states limits its use due to computational complexity.

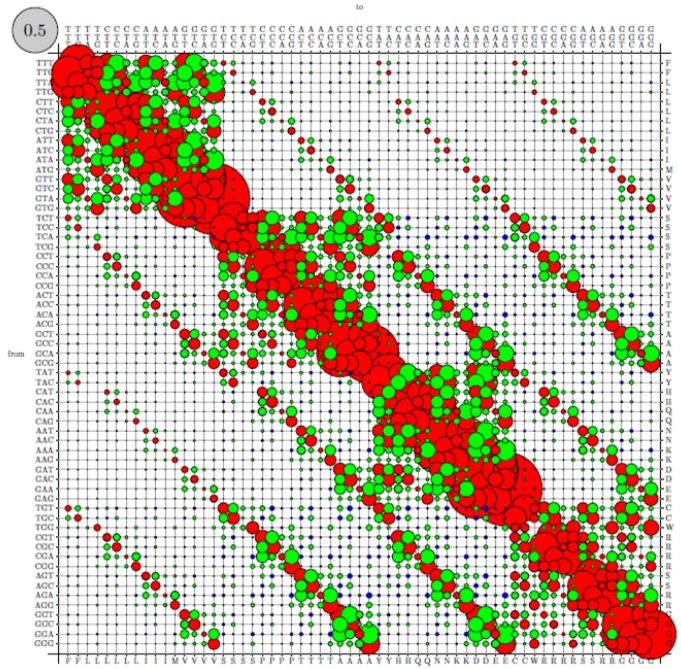
Instantaneous rate matrix for codons:

$$q_{ij} = \begin{cases} 0, & \text{if two codons differ at more than one position,} \\ \pi_j, & \text{for synonymous transversion,} \\ \kappa\pi_j, & \text{for synonymous transition,} \\ \omega\pi_j, & \text{for non-synonymous transversion,} \\ \omega\kappa\pi_j, & \text{for non-synonymous transition,} \end{cases}$$

where π_j is the codon frequency, κ is the transition/transversion rate, and ω is the d_N/d_S.

Generalisation of codon models

Current codon models are too simplistic



Extending the Q matrix

- create empirical models of codon evolution
- find ways to reduce the number of parameters (1,830 parameters for GTR-style model), while keeping generality

Extending codon rate matrix

KCM model of codon evolution: use the structure of codons to generalize the model using Kronecker product

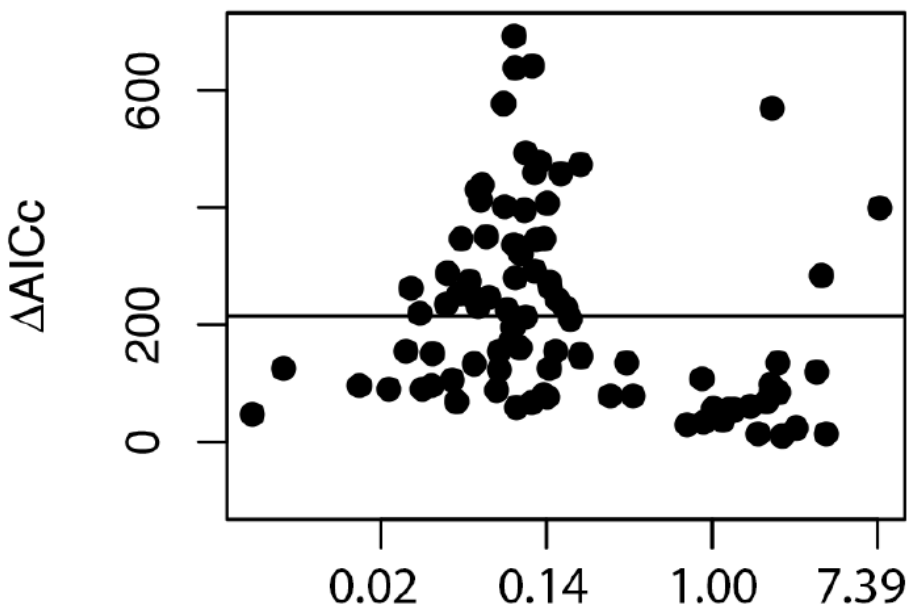
$$q_i = \begin{pmatrix} 1 & q_i^{AC} & q_i^{AG} & q_i^{AT} \\ q_i^{CA} & 1 & q_i^{CG} & q_i^{CT} \\ q_i^{GA} & q_i^{GC} & 1 & q_i^{GT} \\ q_i^{TA} & q_i^{TC} & q_i^{TG} & 1 \end{pmatrix}$$

$$\Psi = \begin{pmatrix} \cdot & q_1^{AC} q_2^{AC} q_3^{AC} & \cdots & q_1^{AT} q_2^{AT} q_3^{AT} \\ & & & \vdots \\ q_1^{CA} q_2^{CA} q_3^{CA} & \cdot & \cdots & \vdots \\ \vdots & & & \\ & \cdots & \cdot & q_1^{GT} q_2^{GT} q_3^{GT} \\ q_1^{TA} q_2^{TA} q_3^{TA} & \cdots & \cdots & \cdot \end{pmatrix} \times \begin{pmatrix} \pi_{AAA} & 0 & \cdots & 0 \\ 0 & \pi_{AAC} & \cdots & 0 \\ & & & \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & \cdots & \pi_{TTT} \end{pmatrix}$$

Zaheri et al. 2014

Codon models on empirical datasets

Testing KCM vs simple or empirical models on 100 different data sets from selectome database



Zaheri et al. 2014

Adaptive molecular evolution

Why do we want to test for adaptive molecular evolution?

- at least partly responsible for evolutionary innovations and species divergence
- allow to gain understanding of forces and mechanisms of molecular evolution
- important for understanding gene function
  - if variability can be shown to be driven by positive selection, functional importance is established

Neutral mutations

Neutral theory claims

- most observed molecular variation due to random fixation of selectively neutral mutations
- affect both polymorphism within species and divergence between species
- falsifiable null hypothesis

It has nowadays been replaced by nearly-neutral theory, which although it has more biological support, is not as easily tested.

Population genetics tests

Several tests of neutral theory have been proposed

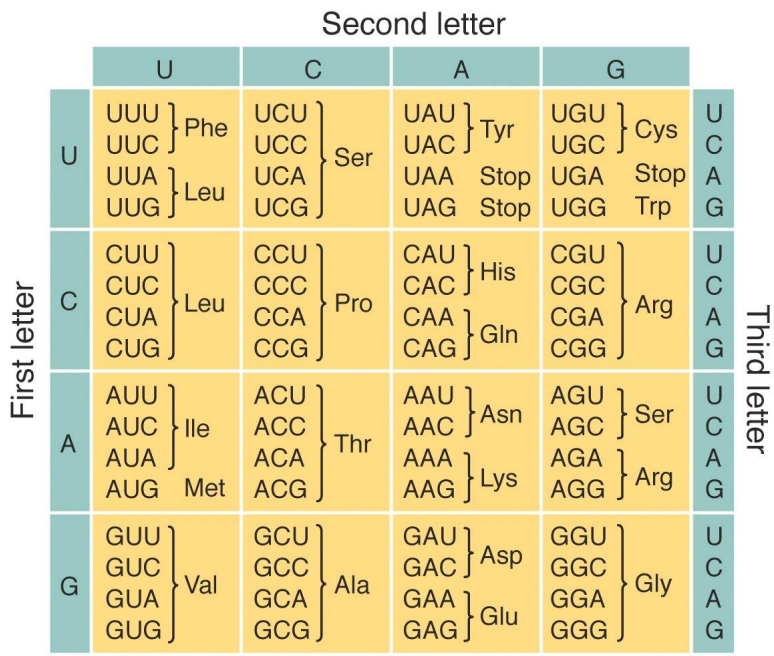
- Hudson-Kreitman-Aguadé test
  - use χ^2 statistic to test if variation within species correlated with variation between species
  - should be the same unless locus is under selection
- McDonald-Kreitman test
  - use Fisher's exact test to test if ratio of non-synonymous to synonymous substitutions are the same within and between species
  - build a 2×2 contingency table within/between species and non-synonymous vs synonymous substitution
- Tajima's D and others

Interpretations of such tests are seldom unequivocal and depend on assumptions about population demography (no migration, random mating, no recombination, etc.) and details of the selection model.

d_N vs d_S

More robust criterion for detecting adaptive evolution in coding-gene

- compare non-synonymous d_N with synonymous d_S rate of substitution along a gene
- ratio $\omega = d_N/d_S$
- if $\omega > 1$ suggests accelerated non-synonymous substitutions



Meaning of ω

Silent mutations do not change amino acid whereas replacement mutations do, the difference in their fixation rates provides a measure of selective pressure on the protein

- silent and replacement sites are interspersed in the same segment of DNA, population effects are shared

- if amino acid change is neutral $\Rightarrow$ will be fixed at same rate as synonymous mutation, so $\omega = 1$
- if amino acid change is deleterious $\Rightarrow$ purifying selection reduce its fixation rate so $\omega < 1$
- if amino acid change is selectively advantageous $\Rightarrow$ will be fixed at higher rate than synonymous mutation, so $\omega > 1$

Likelihood approach

Instead of doing pairwise comparisons of coding sequences, we can modify the HKY85 model of DNA evolution to explain codon evolution and do a full maximum likelihood approach

$$q_{ij} = \begin{cases} 0, & \text{if } i \text{ and } j \text{ differ at more than one position} \\ \pi_j, & \text{for synonymous transversion} \\ \kappa\pi_j, & \text{for synonymous transition} \\ \omega\pi_j, & \text{for non-synonymous transversion} \\ \omega\kappa\pi_j, & \text{for non-synonymous transition} \end{cases}$$

where

- κ is the transition/transversion ratio
- π_j is the frequency of codon j
- ω measures selective pressure on amino acid

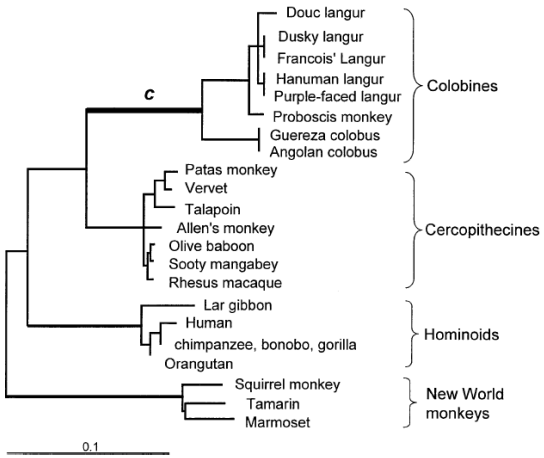
Branch specific selection

Likelihood framework gives a versatile model to test selection on specific branches of a tree

- H_0: all branches have the same ω
- H_1: different ω_i on branches
  - "free-ratio" model with independent ω for each branch (M0)
  - ω different on some branches of the tree (B)

- uses likelihood ratio test to found significantly different model

Example



Yang and Nielsen, 2002

Lysosyme *c* gene (Yang and Nielsen, 2002)

- fight invading bacteria usually in tears and saliva, but found expressed in foreguts for Colobines

Is new function in Colobines associated with positive selection?

- ω_c for branch leading to Colobines, ω_0 for all other branches
- results
  - one-ratio model $lnL = -1,043.83$
  - two-ratios model $lnL = -1,041.70$
  - LRT $= -2 \times -2.13 = 4.26$, $P = 0.039$ with 1 d.f.
  - $\hat{\omega_0} = 0.49$, $\hat{\omega_c} = 3.38$

Selection pressure at each ite

Different approaches have been proposed

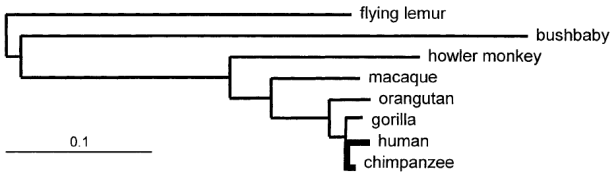
- calculate d_N and d_S rates for a sliding window
- reconstruct ancestral sequences (e.g. with parsimony or likelihood) and count how many times non-synonymous and synonymous changes happened throught time at a single site

- use a statistical distribution to model ω in likelihood framework
- same idea as Gamma distribution for substitution rate heterogeneity

Testing selection among sites

Maximum likelihood framework

- testing for positive selection means that we have some sites with $\omega > 1$
- we can therefore compare two models
  - H_0: distribution that does not allow for sites with $\omega > 1$
    - p_0 sites with $\omega_0 < 1$, p_1 with $\omega_1 = 1$ (M1a)
    - beta distribution which forces $0 < \omega < 1$ (M7)
  - H_1: distribution that allow $\omega > 1$
    - p_0 sites with $\omega_0 < 1$, p_1 with $\omega_1 = 1$, p_2 with $\omega > 1$ (M2a)
    - add another site category on top of beta distribution that take care of $\omega > 1$ (M8)
- the M1a vs M2a test has 2. d.f.
- the M7 vs M8 test has 1 d.f.



Yang and Nielsen, 2002

Example

BRCA1 gene (Yang and Nielsen, 2002)

- tumor suppressor gene that plays a role in genome integrity maintenance

Are some codon under positive selection?

- test simple beta distribution against a model adding an extra rate class (M7 vs M8)
- results
  - beta distribution $lnL = -9,543.52$
  - beta distribution + ω $lnL = -9,535.90$
  - LRT = $-2 \times -7.62 = 15.24$, $P = 0.00049$ with 1 d.f.

  - position 285, 479, 672, 892, 905 and 1144 under positive selection
  - $\hat{\omega} = 2.25$

Pinpointing selected sites

Likelihood ratio test answer the simple question whether sequence has sites under positive selection. It is another problem to find at **which** exact position they are along the sequence.

Posterior probabilities calculated to assign each site to a rate class

- naive Empirical Bayes approach
  - uses maximum likelihood estimates of parameters without accounting for their sampling errors
  - problem in small data sets
  - estimation is not reliable
- Bayes Empirical Bayes approach
  - assign prior to these parameters
  - accomodate uncertainties in parameters using numerical integration of these priors
- full Bayesian estimation $\Rightarrow$ use MrBayes

Locating important sites

Identification of sites under positive selection in sperm lysin gene in abalone species

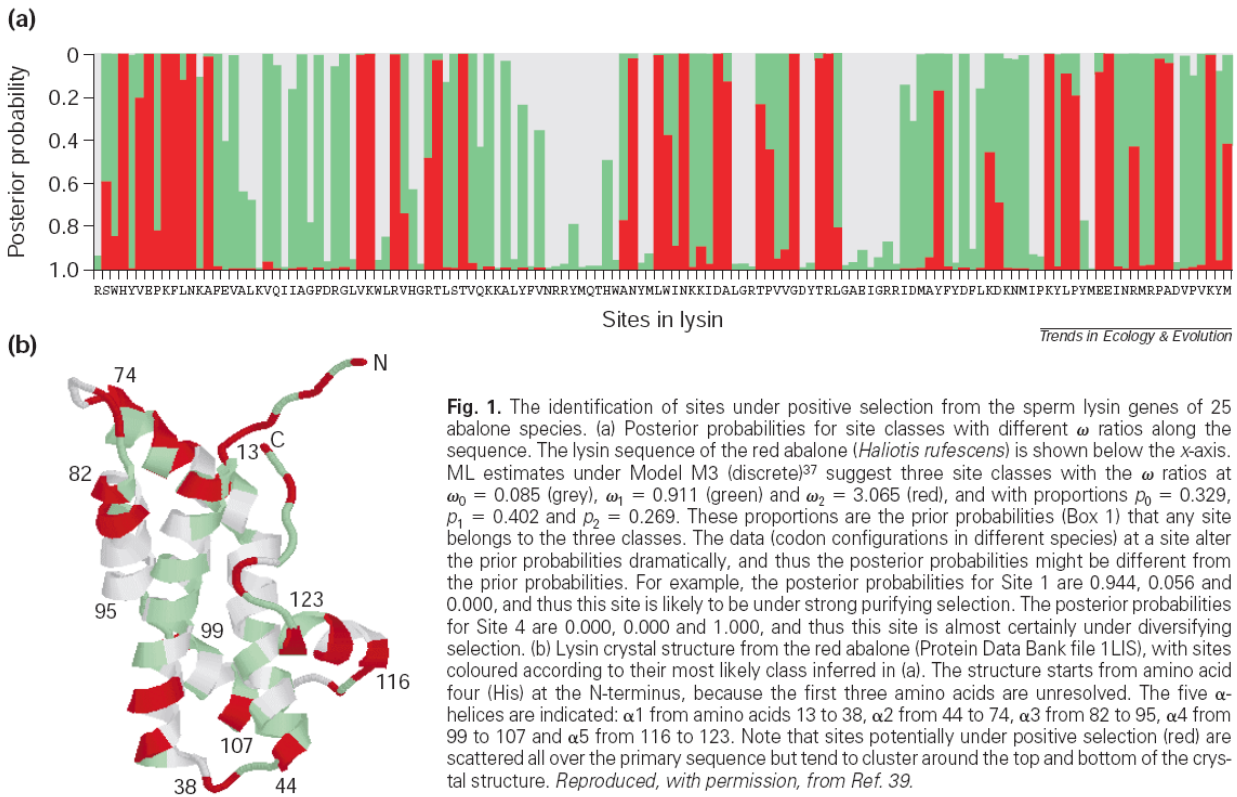


Fig. 1. The identification of sites under positive selection from the sperm lysin genes of 25 abalone species. (a) Posterior probabilities for site classes with different ω ratios along the sequence. The lysin sequence of the red abalone (*Haliotis rufescens*) is shown below the *x*-axis. ML estimates under Model M3 (discrete)[37] suggest three site classes with the ω ratios at $\omega_0 = 0.085$ (grey), $\omega_1 = 0.911$ (green) and $\omega_2 = 3.065$ (red), and with proportions $p_0 = 0.329$, $p_1 = 0.402$ and $p_2 = 0.269$. These proportions are the prior probabilities (Box 1) that any site belongs to the three classes. The data (codon configurations in different species) at a site alter the prior probabilities dramatically, and thus the posterior probabilities might be different from the prior probabilities. For example, the posterior probabilities for Site 1 are 0.944, 0.056 and 0.000, and thus this site is likely to be under strong purifying selection. The posterior probabilities for Site 4 are 0.000, 0.000 and 1.000, and thus this site is almost certainly under diversifying selection. (b) Lysin crystal structure from the red abalone (Protein Data Bank file 1LIS), with sites coloured according to their most likely class inferred in (a). The structure starts from amino acid four (His) at the N-terminus, because the first three amino acids are unresolved. The five α-helices are indicated: α1 from amino acids 13 to 38, α2 from 44 to 74, α3 from 82 to 95, α4 from 99 to 107 and α5 from 116 to 123. Note that sites potentially under positive selection (red) are scattered all over the primary sequence but tend to cluster around the top and bottom of the crystal structure. *Reproduced, with permission, from Ref. 39.*

Yang and Bielaweski, 2000

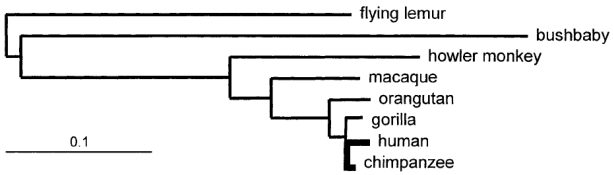
More complex model

The furthest we can go is to allow branch-site specific model. To do so, we introduce four site classes on two different types of lineages

- foreground where positive selection occurs
- background where neutral or purifying selection occurs

Class	Proportion	Background ω	Foreground ω
0	p_0	$0 < \omega_0 < 1$	$0 < \omega_0 < 1$
1	p_1	$\omega_1 = 1$	$\omega_1 = 1$
2a	$\frac{(1-p_0-p_1)p_0}{(p_0+p_1)}$	$0 < \omega_0 < 1$	$\omega_2 > 1$
2b	$\frac{(1-p_0-p_1)p_1}{(p_0+p_1)}$	$\omega_1 = 1$	$\omega_2 > 1$

The test to make is to compare this model with the M1a model described before with 2 d.f.



Yang and Nielsen, 2002

Example

BRCA1 gene (Yang and Nielsen, 2002)

- tumor suppressor gene that plays a role in genome integrity maintenance

Are some codon under selection along the branch leading to the (human,chimp) clade?

- test neutral model against lineage specific discrete (M1a vs A)
- results
  - neutral $lnL = -9,545.19$
  - discrete lineage specific $lnL = -9,540.89$
  - LRT = $-2 \times -4.30 = 8.60$, $P = 0.01$ with 2 d.f.

  - position 902 under positive selection;
  - $\hat{\omega}_0 = 0.38$, $\hat{\omega}_1 = 2.086$, $\hat{\omega}_2 = 6.42$

Branch-site models

When estimating ω, we also know that

- ω will vary between sites
- ω will vary between branches if we have episodic events of positive selection

If we fix a priori the branch(es) that have $\omega > 1$, we can integrate the likelihood over different site categories using a mixture of Q matrices with different ω values

$$P(Data|\theta, T) = \sum_{\omega_s} P(\omega_s) P(Data|\theta, T, \omega_s)$$

We can estimate from the data ω_s and the proportion of sites in each category (Zhang et al 2005).

Likelihood calculations are time-consuming and we need fast ways of calculated $P = e^{Qt}$.

Mixed codon models

The approach of applying different categories of sites can be extending to branches as well (Kosakovsky-Pond et al. 2011).

If the branch categories are independent, then, for a given site s, $P(\Omega_s) = \prod_{b=1}^{B} P(\omega_{bs})$.

The full likelihood becomes:

$$\begin{aligned} P(Data|\theta, T) &= \sum_{\Omega_s} P(\Omega_s) P(Data|\theta, T, \Omega_s) \\ &= \sum_{\Omega_s} \prod_{b=1}^{B} P(\omega_{bs}) \sum_{A} P(Data, A|\theta, T, \Omega_s) \end{aligned}$$

where A is all possible ancestral vectors at each node.

Considerations

We should be careful to the following aspects

- do not use pairwise tests to find positive selection!
- use either full Bayesian or BEB approach to find which sites are under positive selection
- detecting selection along lineages works only if ω average over all sites is > 1
- detecting selection at sites works only if ω averaged over all branches is > 1
- averaging over sites is a more serious problem than averaging over lineages