Statistical methods for estimating phylogenetic trees

part 2

Front Matter

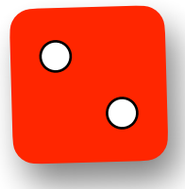
most slides are inspired by or directly lifted (these will be attributed) from lectures by:

Paul Lewis, Mark Holder, David Swofford, & John Huelsenbeck

if you wish to see these in person or access their materials go to the Workshop on Molecular Evolution at MBL website

Phylogenetics Lecture Plan

- some review and introduction
- some basic probability
- **calculating likelihood**
- **substitution models**
- **maximum likelihood methods**
- Bayesian thinking
- hierarchical models
- divergence-time estimation




Likelihood of a Single DNA Sequence

first 32 nucleotides of the ψη-globin gene of gorilla:

GAAGTCCTTGAGAAATAAACTGCACACACTGG

$$\mathcal{L} = \pi_G\pi_A\pi_A\pi_G\pi_T\pi_C\pi_C\pi_T\pi_T\pi_G\pi_A\pi_G\pi_A\pi_A\pi_A\pi_T\pi_A\pi_A\pi_A\pi_C\pi_T\pi_G\pi_C\pi_A\pi_C\pi_A\pi_C\pi_A\pi_C\pi_T\pi_G\pi_G$$
$$= \pi_A^{12}\pi_C^7\pi_G^7\pi_T^6$$

*we assume sites are independent

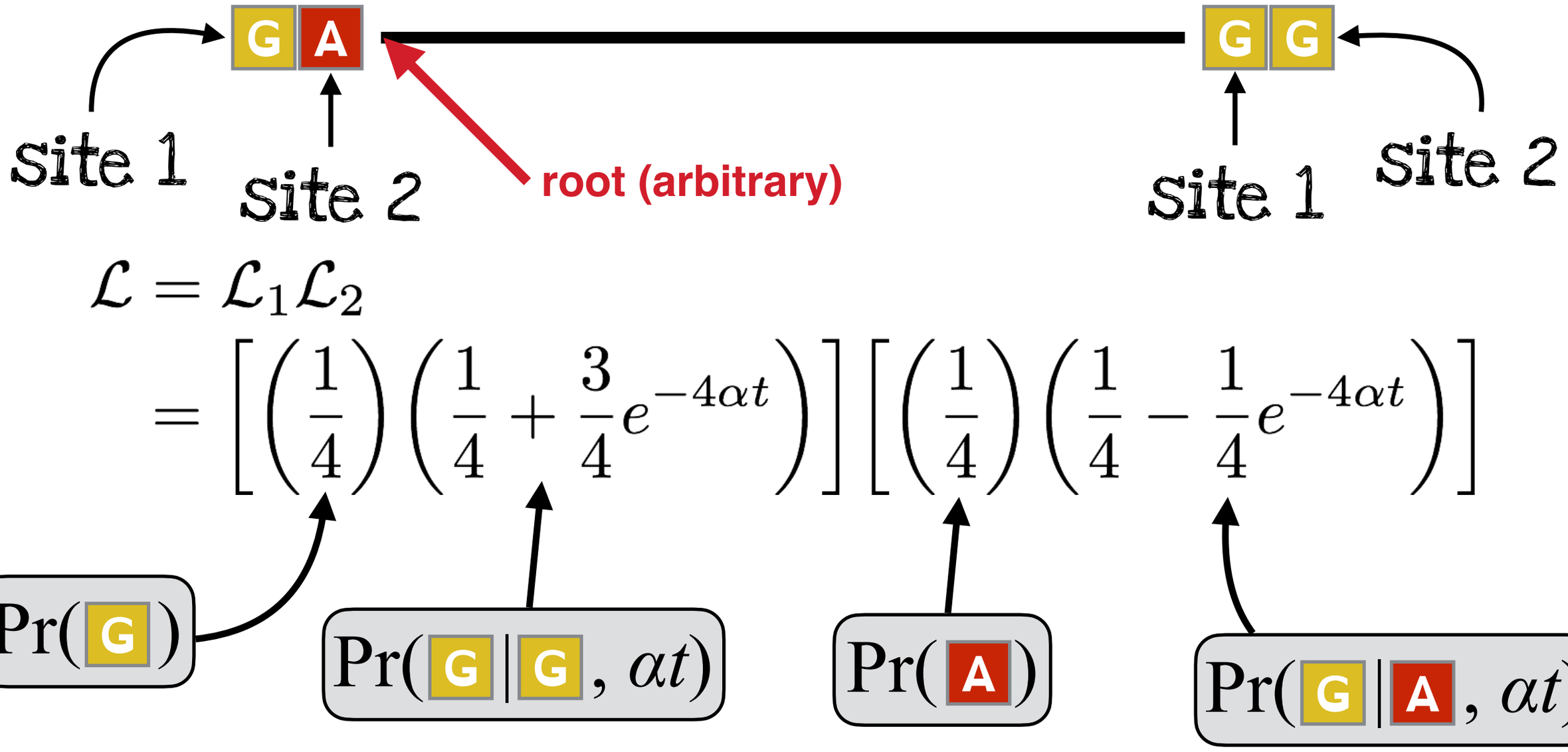
$$\log\mathcal{L} = 12\log(\pi_A) + 7\log(\pi_C) + 7\log(\pi_G) + 6\log(\pi_T)$$

$$\Pr(A) = \pi_A$$
$$\Pr(C) = \pi_C$$
$$\Pr(G) = \pi_G$$
$$\Pr(T) = \pi_T$$

we can already see by eye-balling this that a model allowing unequal base frequencies will fit better than a model that assumes equal base frequencies because there are about twice as many As as there are Cs, Gs and Ts.

Likelihood of the Simplest Tree

To keep things simple, assume that the sequences are only 2 nucleotides long:



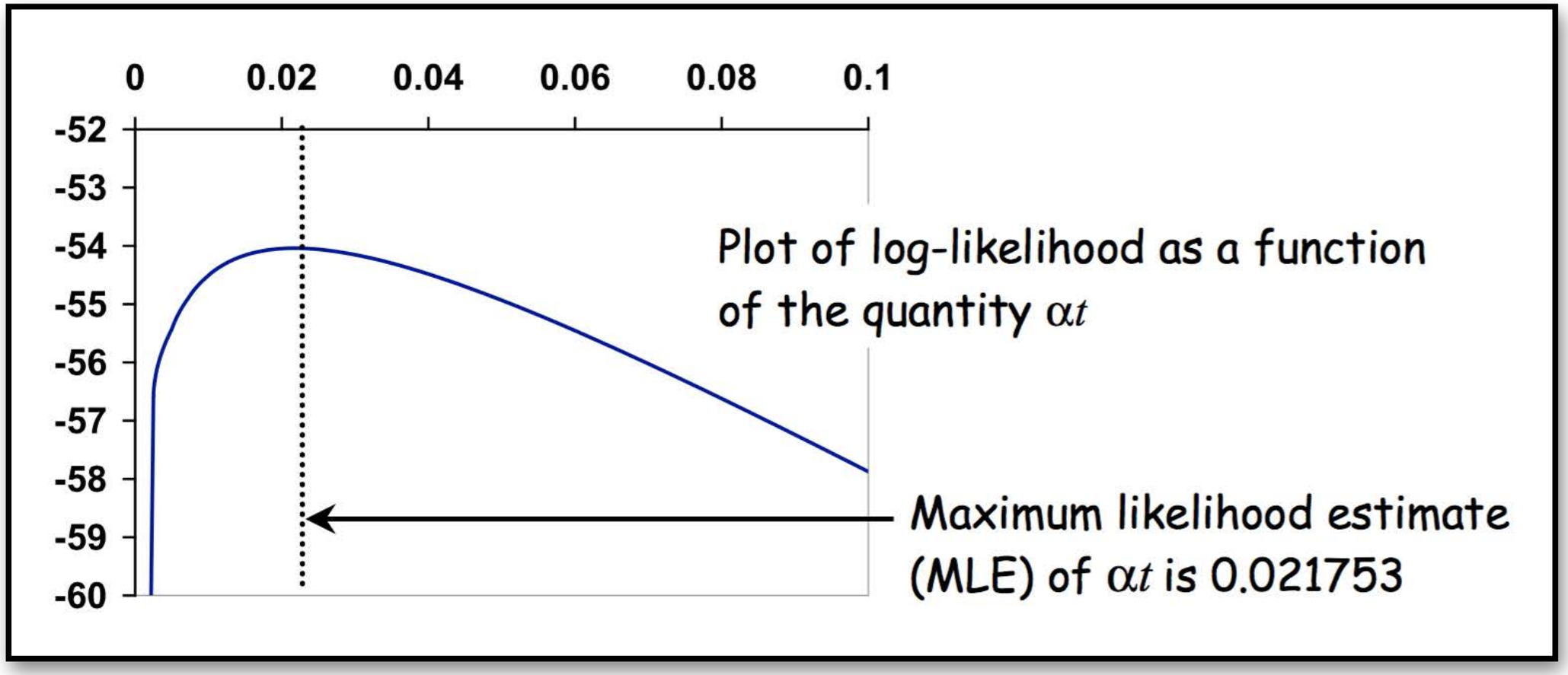
(based on slides by Paul Lewis https://molevolworkshop.github.io/faculty-lewis)

Maximum Likelihood Estimation

first 32 nucleotides of the ψη-globin gene of gorilla & orangutan:

```
gorilla   GAAGTCCTTGAGAAATAAACTGCACACACTGG
orangutan GGACTCCTTGAGAAATAAACTGCACACACTGG
```

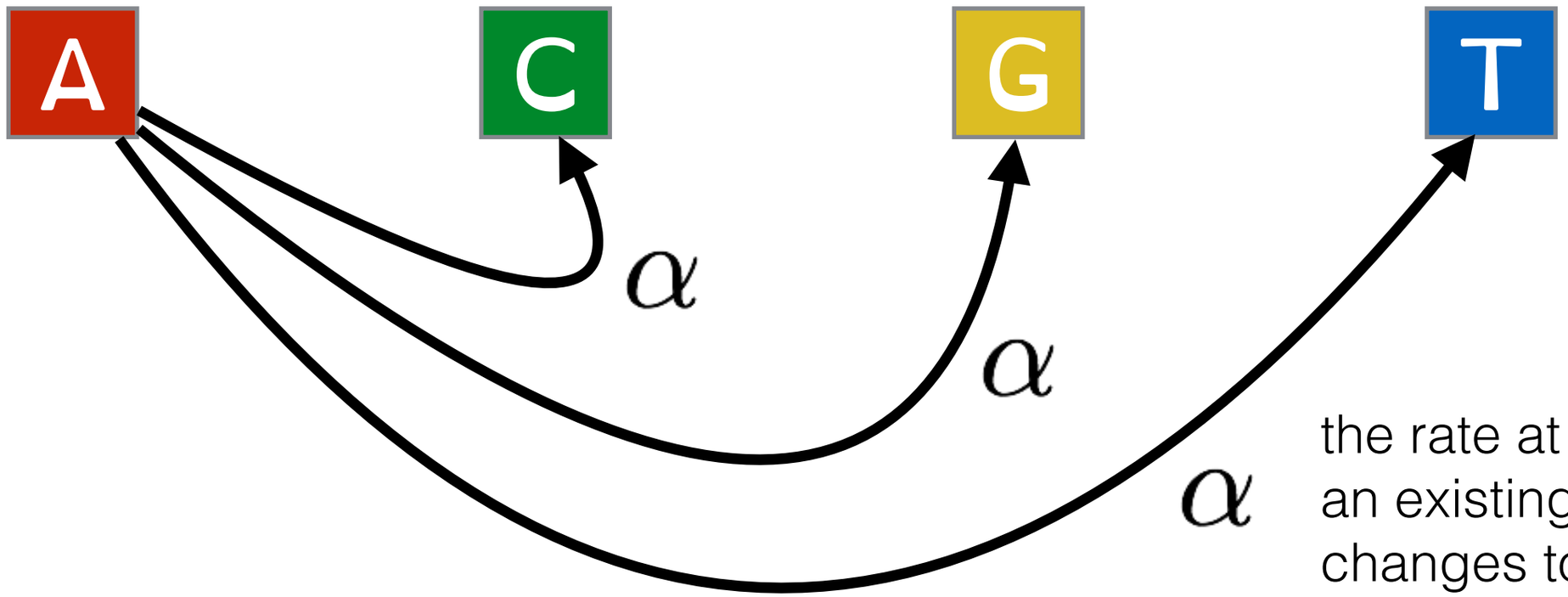
$$\mathcal{L} = \left[\left(\frac{1}{4}\right)\left(\frac{1}{4} + \frac{3}{4}e^{-4\alpha t}\right)\right]^{30}\left[\left(\frac{1}{4}\right)\left(\frac{1}{4} - \frac{1}{4}e^{-4\alpha t}\right)\right]^{2}$$



(based on slides by Paul Lewis https://molevolworkshop.github.io/faculty-lewis)

Substitution Rate

$$\text{number of substitutions} = \text{substitution rate} \times \text{time}$$

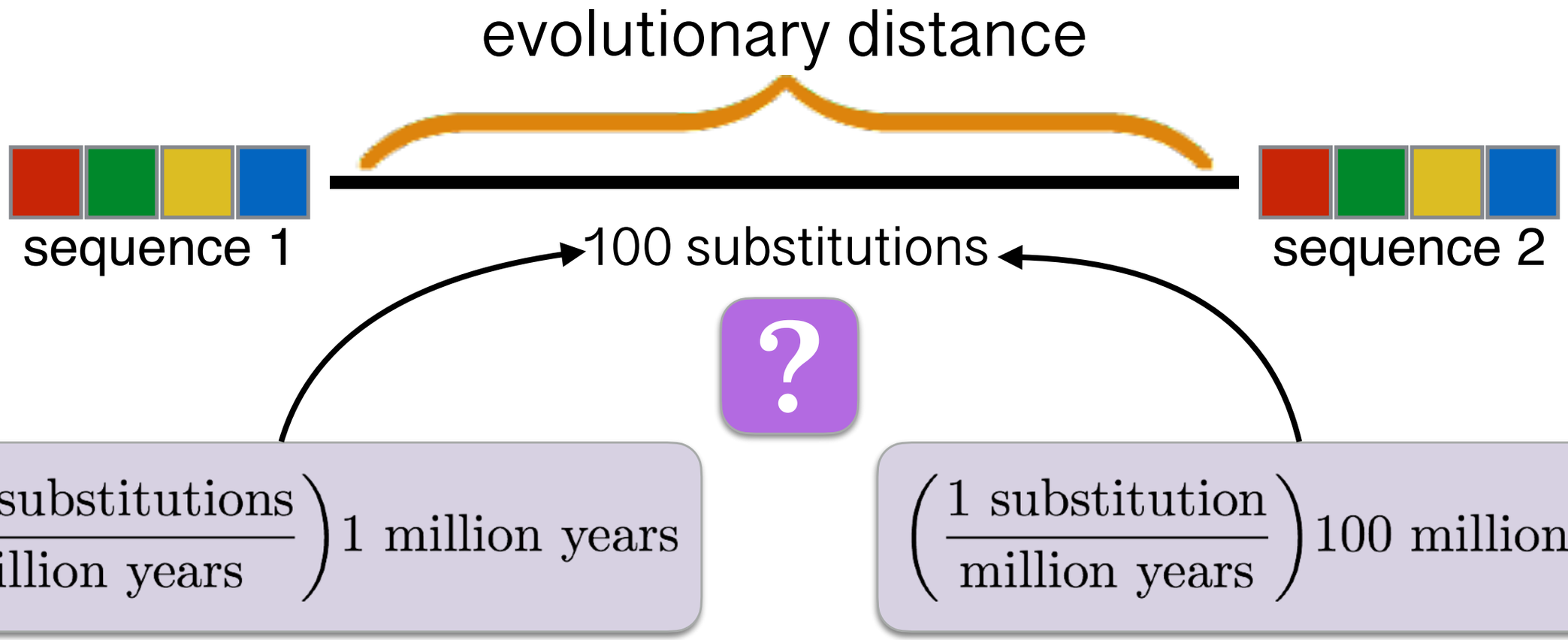


the rate at which an existing A changes to a T

the overall substitution rate is 3α, so the expected number of substitutions (ν) is:

$$\nu = 3\alpha t$$

Rate and Time are Confounded



sequence data can only provide information about the evolutionary distance as **rate × time**, we cannot identify the absolute rate or time

we will cover how to estimate substitution rates and divergence times later

Evolutionary Distances

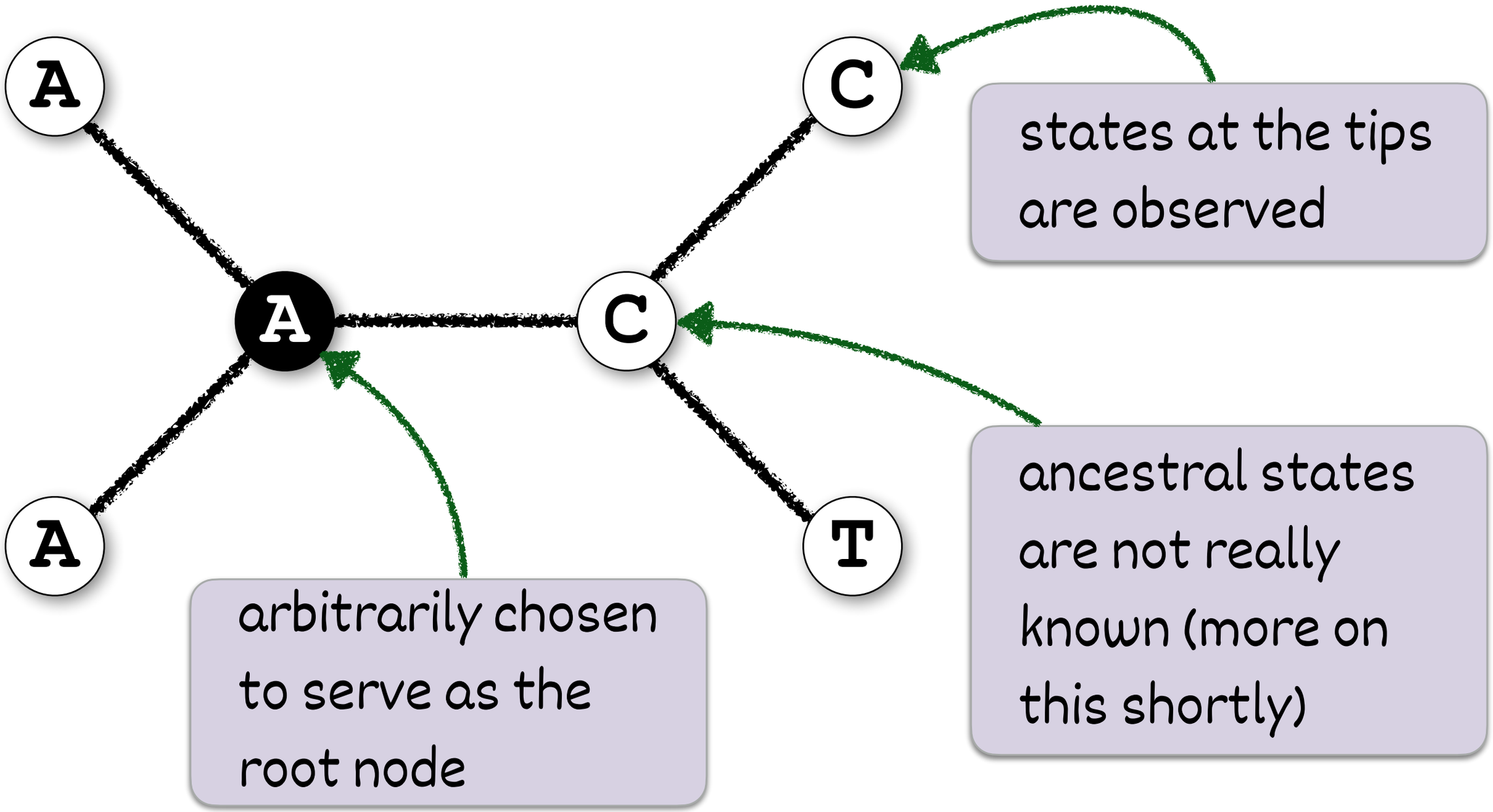
model	expected number of substitutions: $\nu = \{r\}t$
JC69	$\nu = \{3\alpha\}t$
F81	$\nu = \{2\mu(\pi_R\pi_Y + \pi_A\pi_G + \pi_C\pi_T)\}t$
K80	$\nu = \{\beta(\kappa + 2)\}t$
HKY	$\nu = \{2\mu[\pi_R\pi_Y + \kappa(\pi_A\pi_G + \pi_C\pi_T)]\}t$

in the formulas above, the overall rate r (in curly brackets) is a function of all parameters in the substitution model

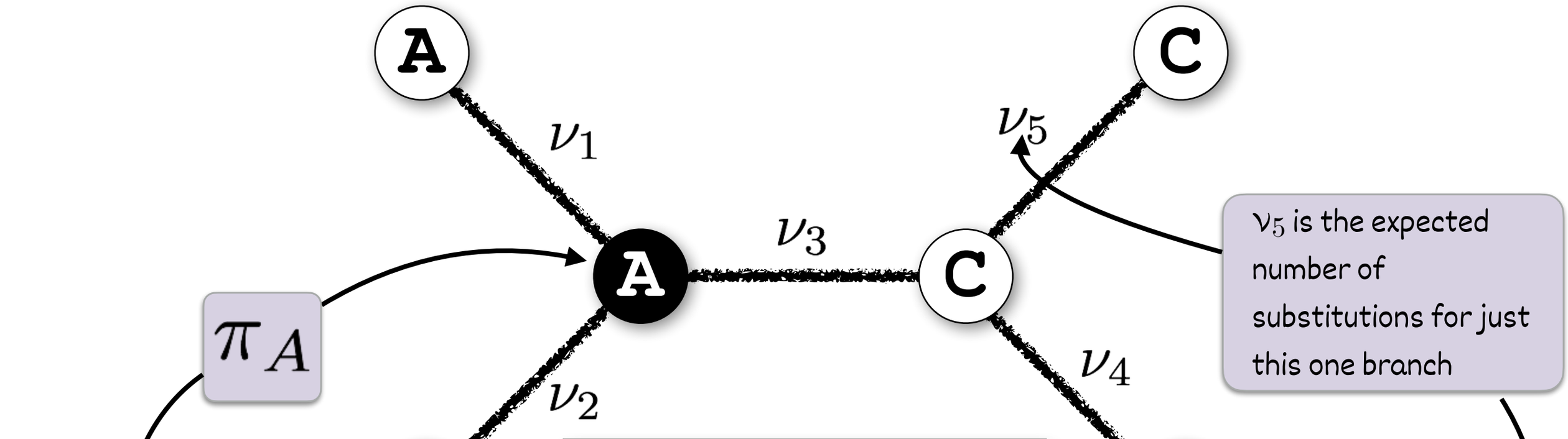
one substitution model parameter is always determined from the edge length; the others are usually global (i.e. same value applies to all edges)

Likelihood of an Unrooted Tree

(data shown only for 1 site)



Likelihood for a Single Site *k*



$$\mathcal{L}_k = \frac{1}{4}\left[\frac{1}{4}+\frac{3}{4}e^{-4\nu_1/3}\right]\left[\frac{1}{4}+\frac{3}{4}e^{-4\nu_2/3}\right]\left[\frac{1}{4}-\frac{1}{4}e^{-4\nu_3/3}\right]\left[\frac{1}{4}-\frac{1}{4}e^{-4\nu_4/3}\right]\left[\frac{1}{4}+\frac{3}{4}e^{-4\nu_5/3}\right]$$

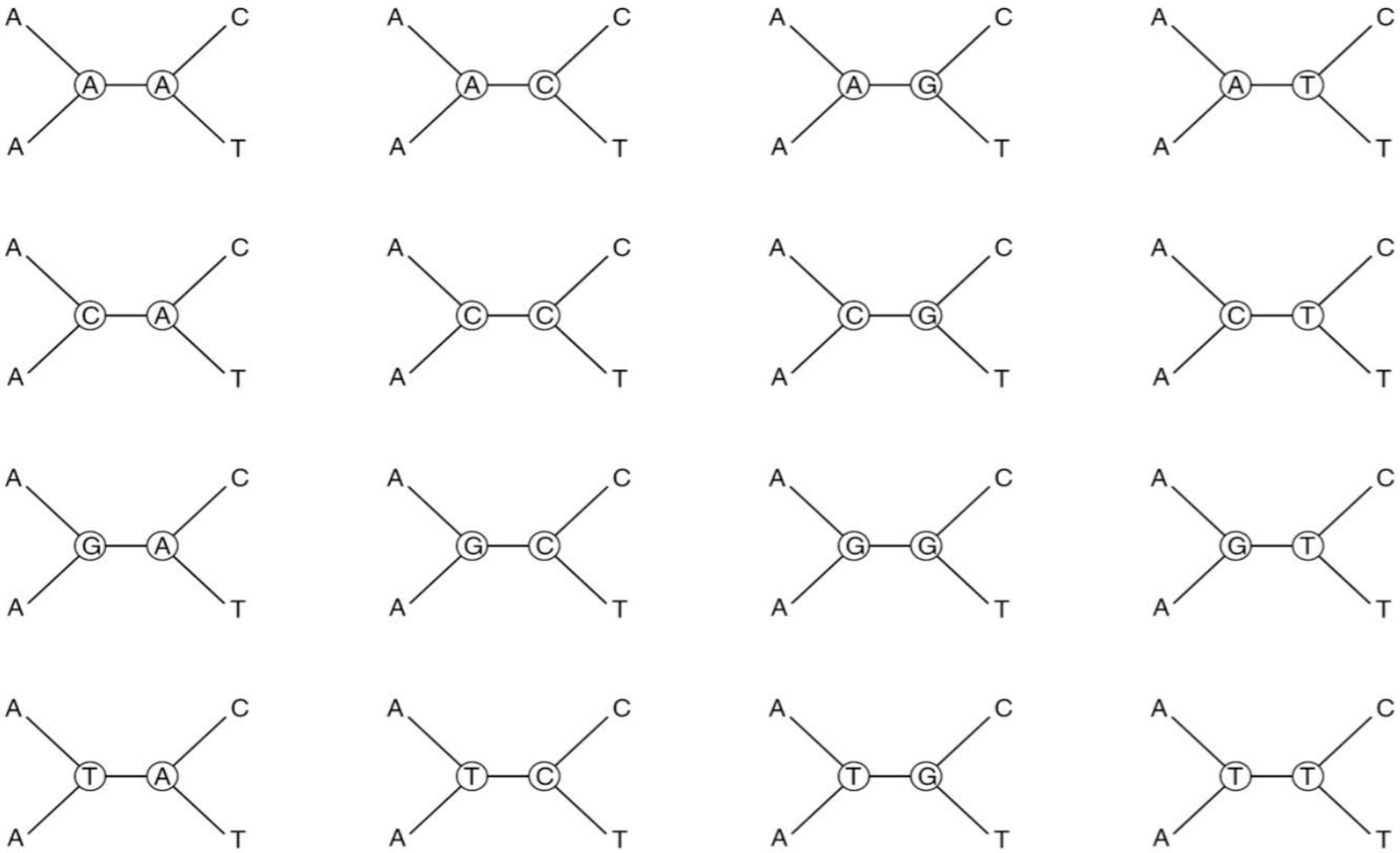
$p_{AA}(\nu_1)$ $p_{AA}(\nu_2)$ $p_{AC}(\nu_3)$ $p_{CT}(\nu_4)$ $p_{CC}(\nu_5)$

the AND rule !

(based on slides by Paul Lewis https://molevolworkshop.github.io/faculty-lewis)

Likelihood for a Single Site

Brute force approach would be to calculate $\mathcal{L}_k$ for all 16 combinations of ancestral states and sum them

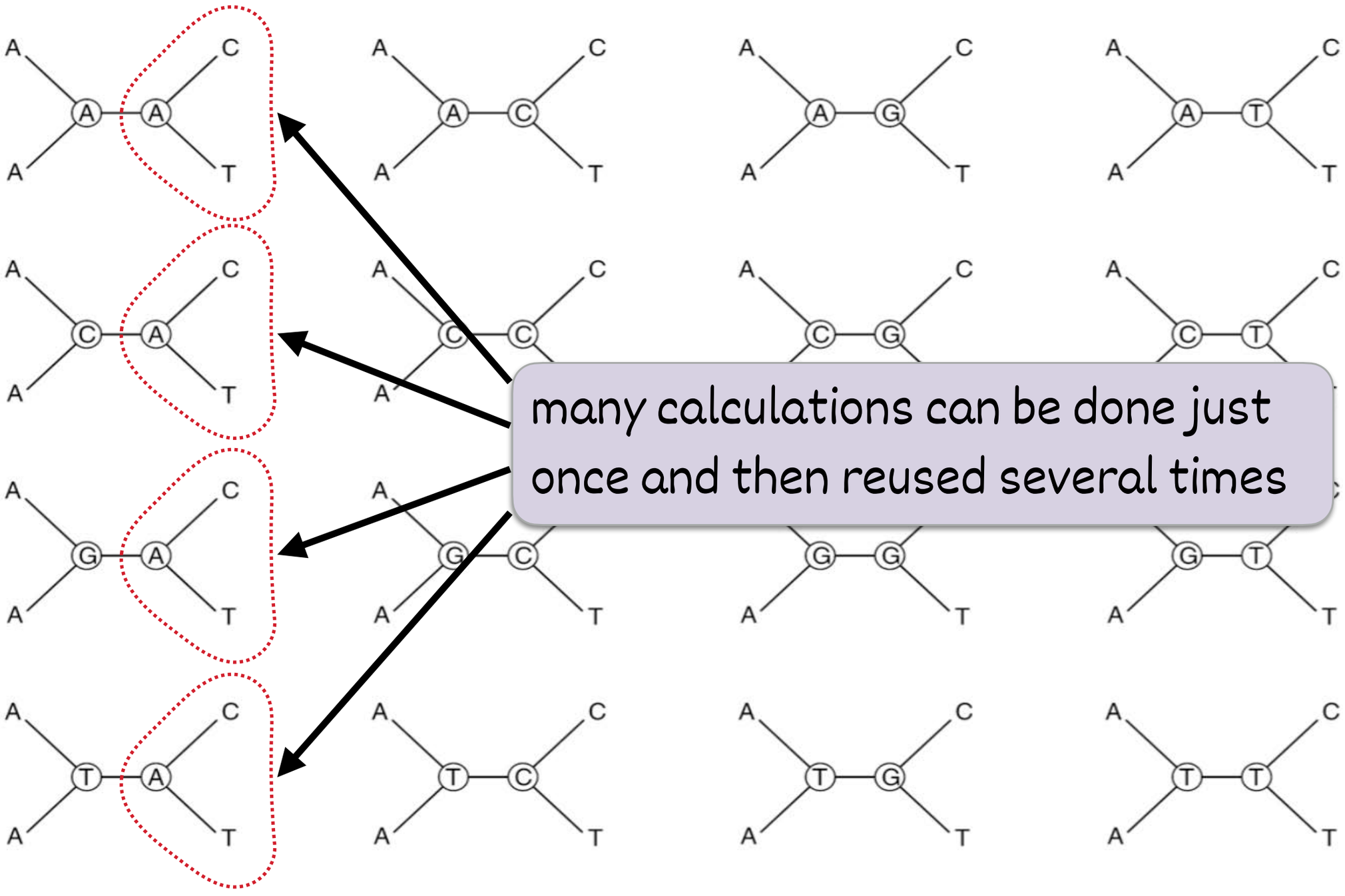


the OR rule !

(based on slides by Paul Lewis https://molevolworkshop.github.io/faculty-lewis)

Likelihood for a Single Site

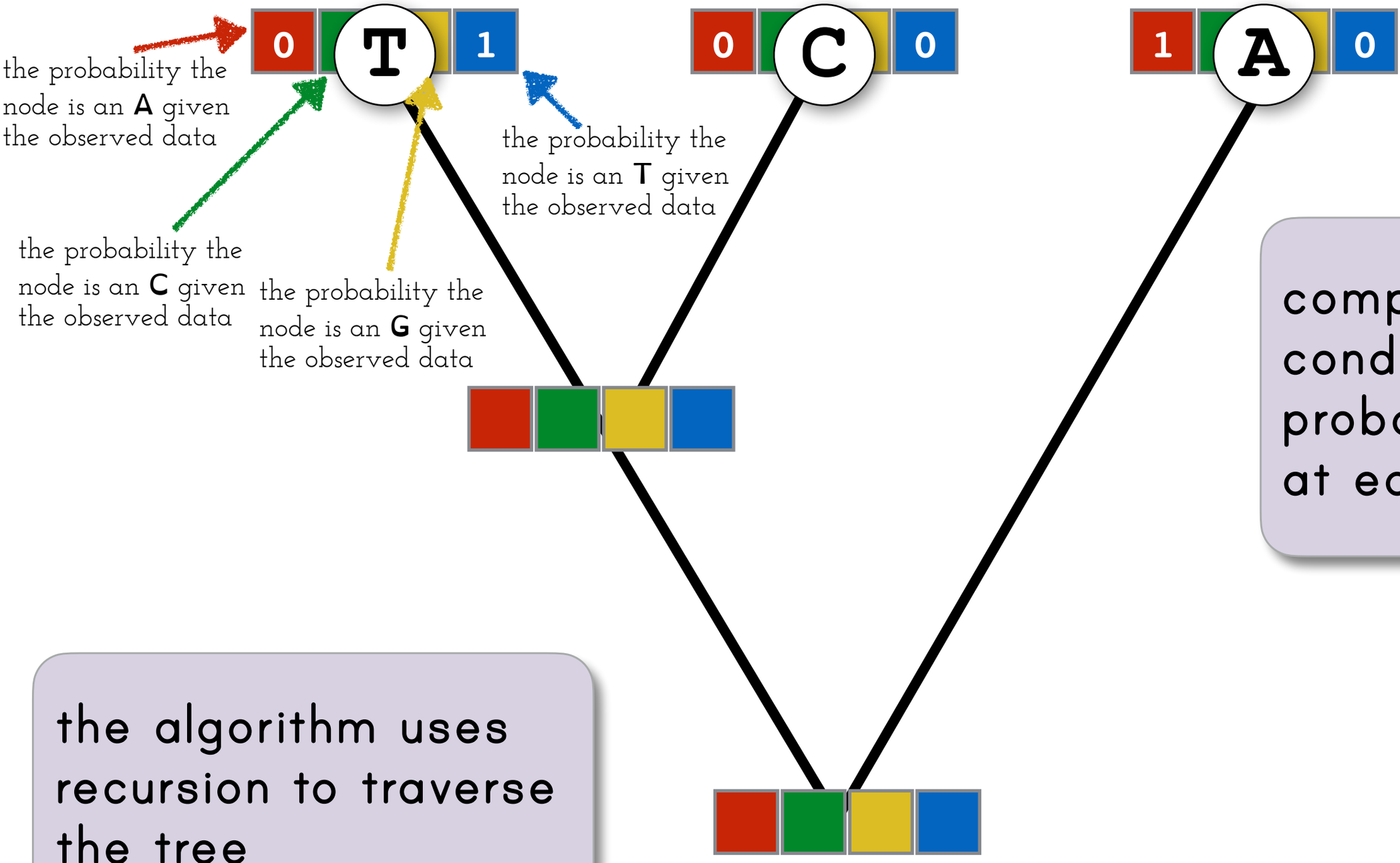
The **pruning algorithm** gives us the same result in much less time



Felsenstein, J. 1981. Evolutionary trees from DNA sequences: a maximum likelihood approach. *Journal of Molecular Evolution*, 17:368-376

(based on slides by Paul Lewis https://molevolworkshop.github.io/faculty-lewis)

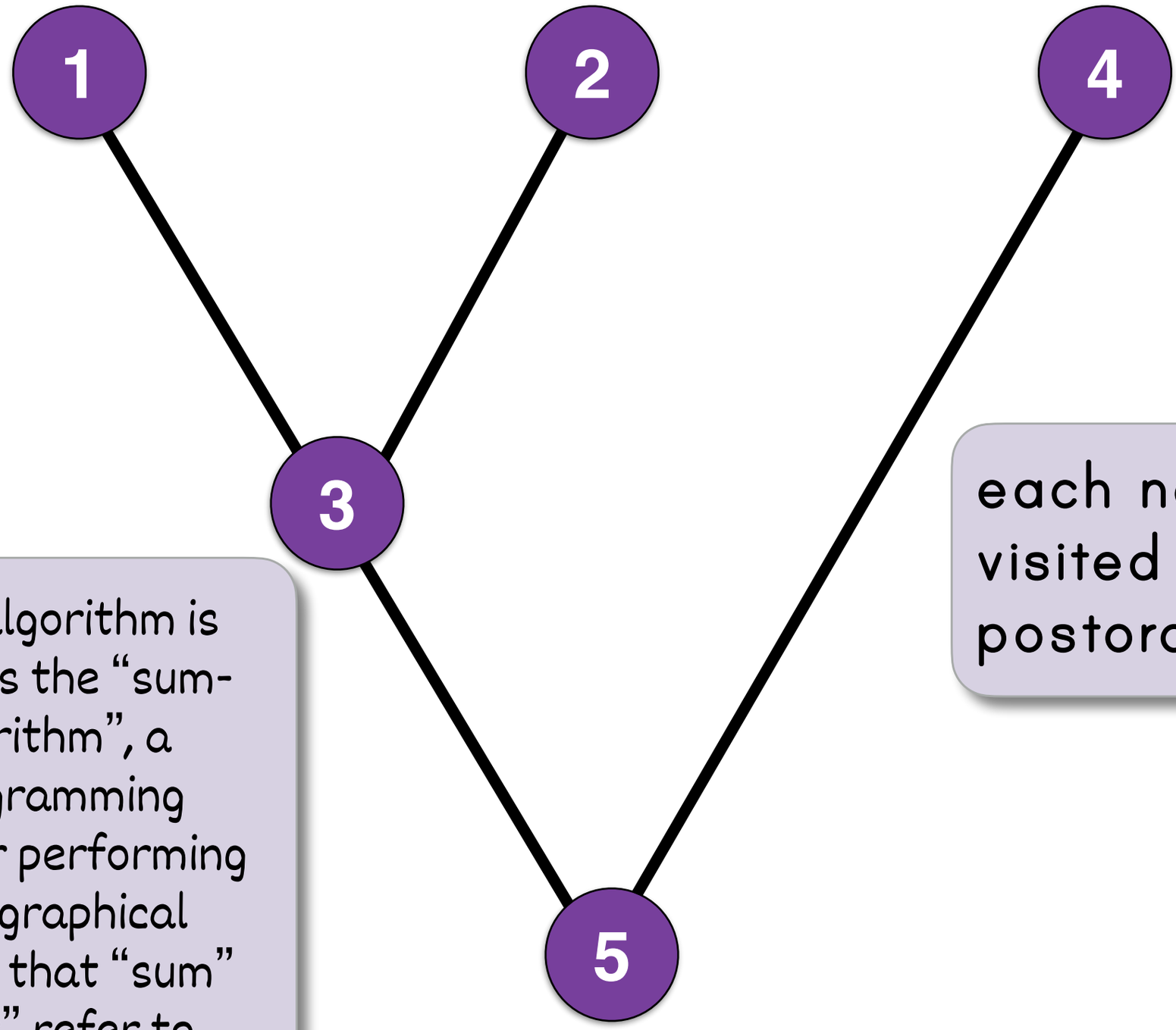
The Pruning Algorithm



computes the conditional probabilities at each node

the algorithm uses recursion to traverse the tree

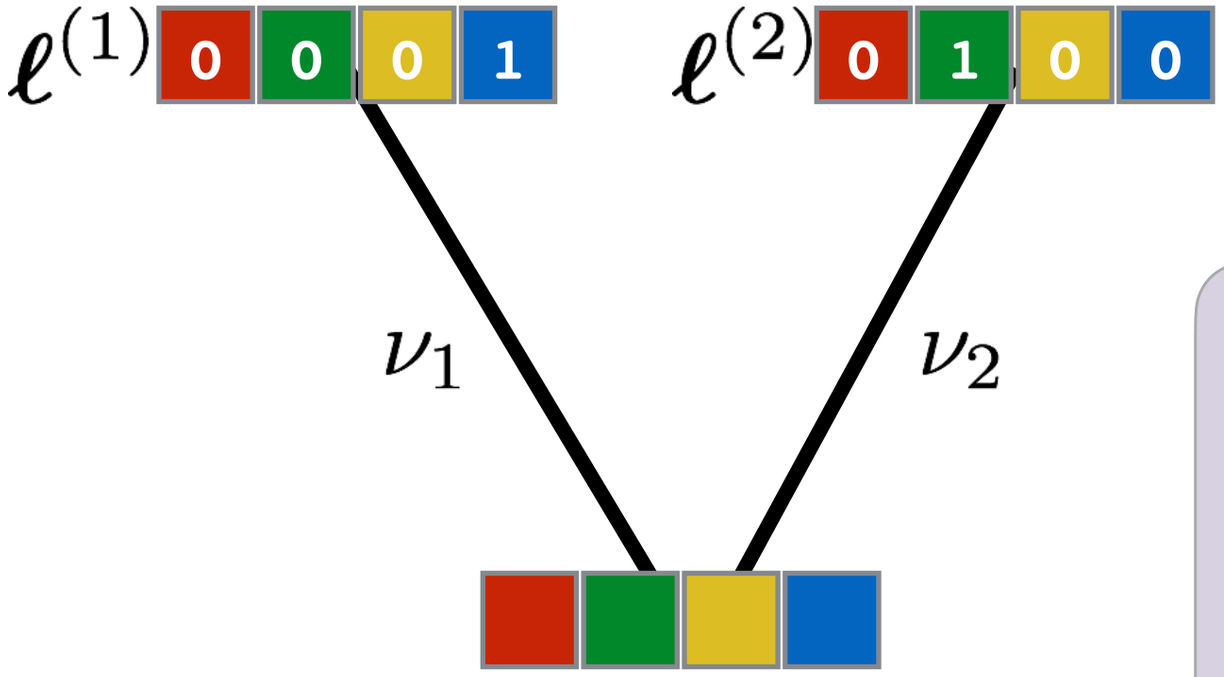
The Pruning Algorithm



each node is visited following postorder traversal

the pruning algorithm is also known as the "sum-product algorithm", a dynamic programming algorithm for performing inference on graphical models (note that "sum" and "product" refer to "OR" & "AND")

The Pruning Algorithm



$\ell^{(1)}$	$\ell^{(2)}$
$\ell_A^{(1)} = 0$	$\ell_A^{(2)} = 0$
$\ell_C^{(1)} = 0$	$\ell_C^{(2)} = 1$
$\ell_G^{(1)} = 0$	$\ell_G^{(2)} = 0$
$\ell_T^{(1)} = 1$	$\ell_T^{(2)} = 0$

we use the AND rule to combine the conditional probabilities for a given state at an ancestral node

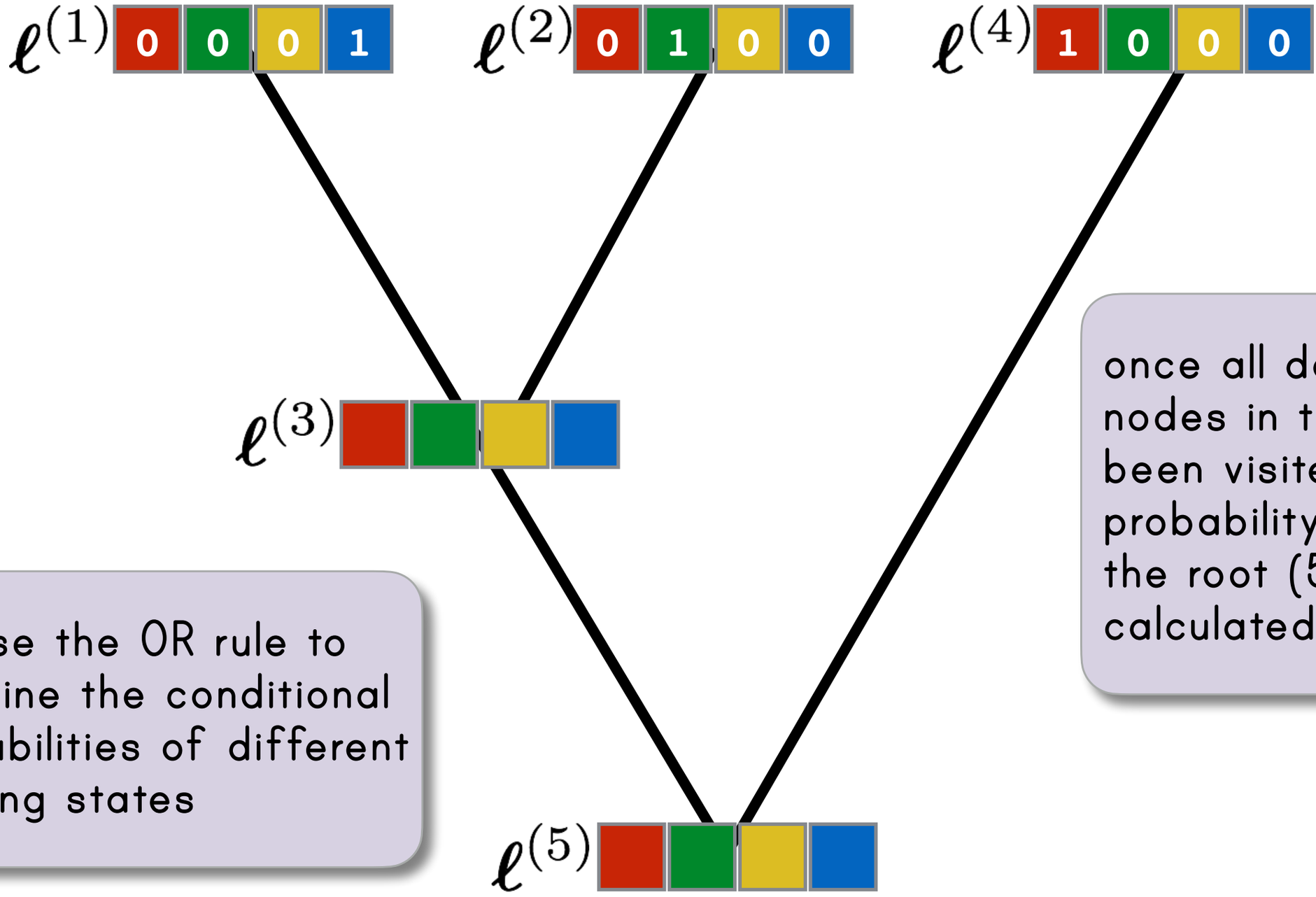
$$\ell_i^{(3)} = \left(\sum_j p_{ij}(\nu_1) \ell_j^{(1)} \right) * \left(\sum_j p_{ij}(\nu_2) \ell_j^{(2)} \right)$$

conditional probability

$\ell_i^{(x)}$ the probability of the observed states if the ancestral node *x* had state *i*

$$p_{ij}(\nu) = \frac{1}{4}\left(1 - e^{-4\nu}\right)$$

The Pruning Algorithm

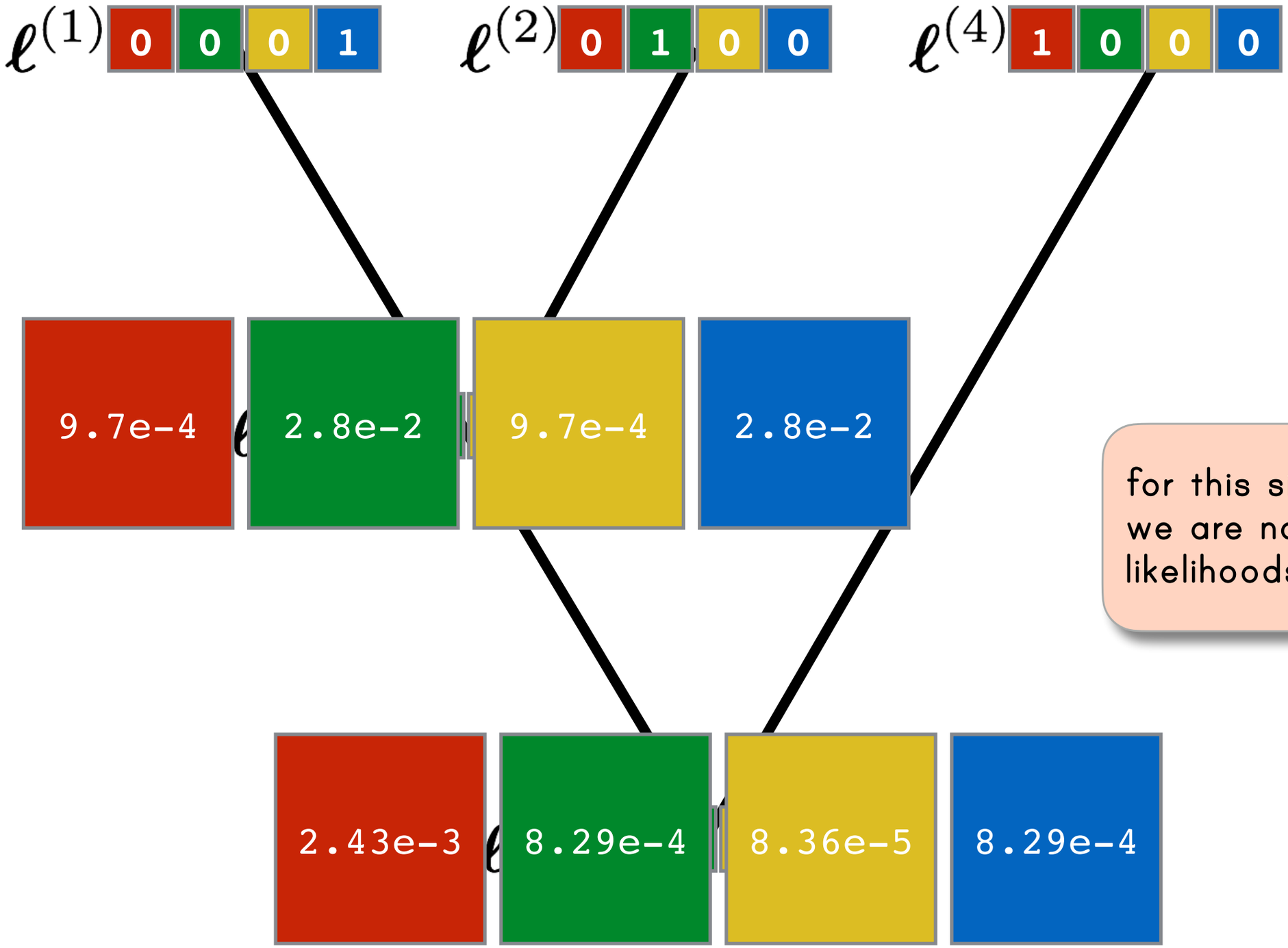


we use the OR rule to combine the conditional probabilities of different starting states

once all descendant nodes in the tree have been visited, the probability vector at the root (5) can be calculated

$$\mathcal{L}_{site} = \pi_A \times \ell_A^{(5)} + \pi_C \times \ell_C^{(5)} + \pi_G \times \ell_G^{(5)} + \pi_T \times \ell_T^{(5)}$$

The Pruning Algorithm



for this simple example we are not using log likelihoods

$$\mathcal{L}_{site} = \pi_A \times \ell_A^{(5)} + \pi_C \times \ell_C^{(5)} + \pi_G \times \ell_G^{(5)} + \pi_T \times \ell_T^{(5)}$$

Where is the model?

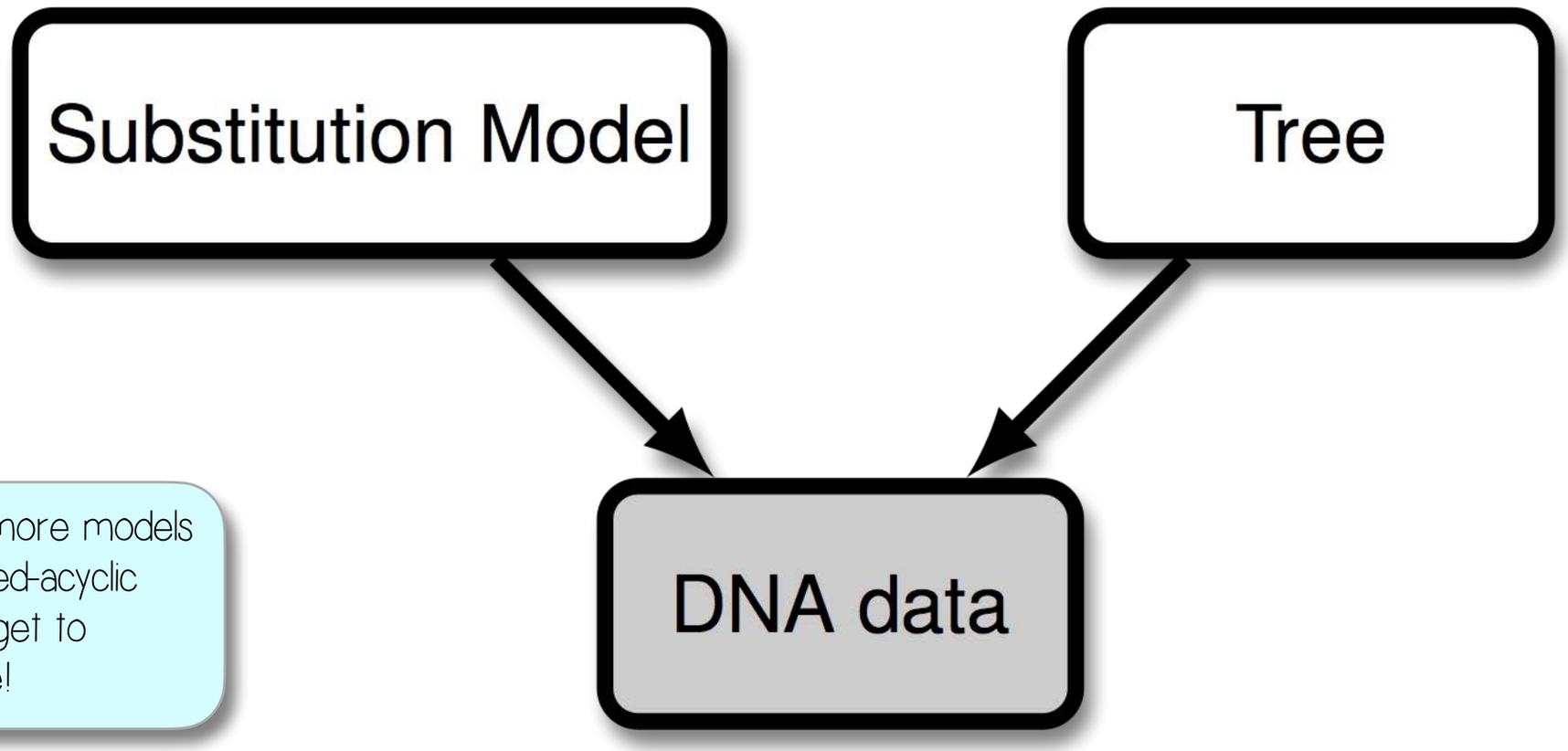
here is what we have been assuming so far:

1. substitutions occur according to a Markov process
2. the equilibrium base frequencies are all equal ($\pi_A = \pi_C = \pi_G = \pi_T = 0.25$)
3. the rate of substitution is equal for all substitution types ($r_{AC} = r_{AG} = r_{AT} = r_{CG} = r_{CT} = r_{GT}$)
4. rates of substitution are time reversible (e.g., $r_{AC} = r_{CA}$)
5. the expected number of substitutions is $\nu = 3\alpha t$

the Jukes-Cantor (1969) substitution model!

How does it all fit together?

this graphical model shows how we assume our DNA data are conditionally dependent on the tree (with branch lengths) and substitution model



you will see a lot more models depicted as directed-acyclic graphs when we get to Bayesian inference!

the node containing the data is shaded gray to indicate that these values are observed and cannot change

Substitution Models

models describing discrete character change are substitution models

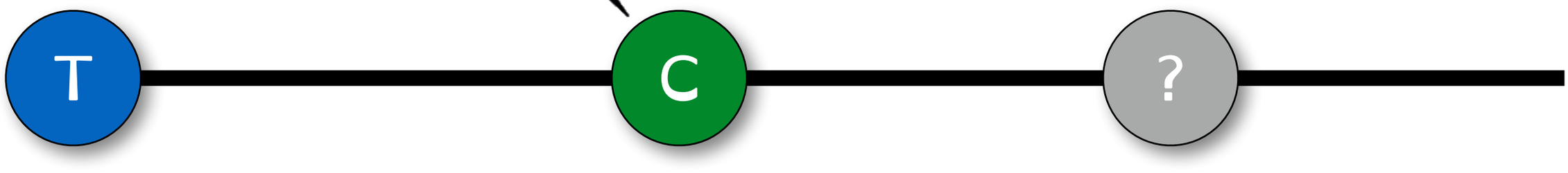
these take are also called **continuous-time Markov chain (CTMC)** models

these models all possess the **Markov property**: the probability that a character will be in a given state at time t, only depends on its state at time 0

What is a Markov process?

lineage starts with base T at some site

a substitution occurs changing T → C



MARKOV PROPERTY

it is irrelevant that there was a T present at this site before time 0 (this makes it a Markov process!)

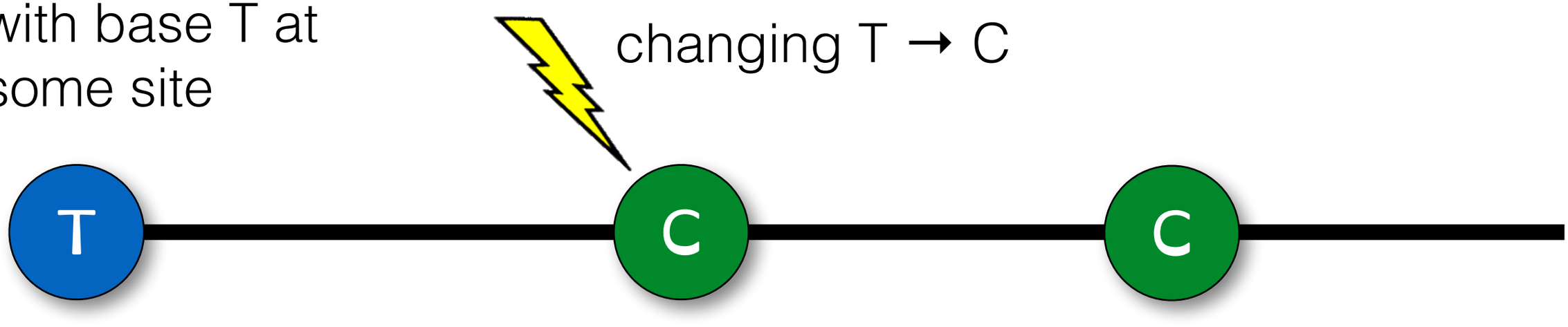
to predict which base will be present after some time t we only need to know which base was present at time 0 (C in this case)

(based on slides by Paul Lewis https://molevolworkshop.github.io/faculty-lewis)

Transition Probabilities

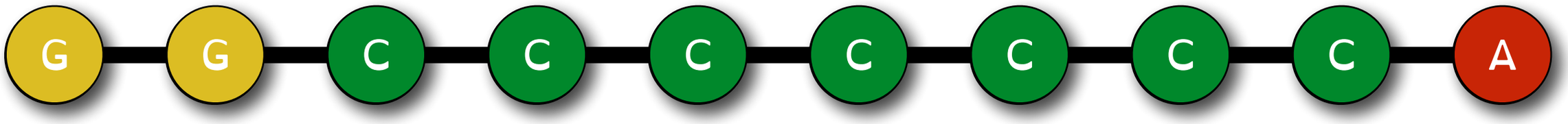
lineage starts with base T at some site

a substitution occurs changing T → C



the transition probability is the conditional probability that there is a C present at a site after time t given that there was a C present at time 0

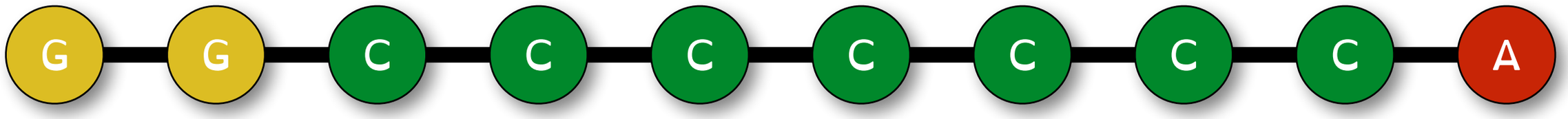
Toy Example: Observing a Lineage Evolving



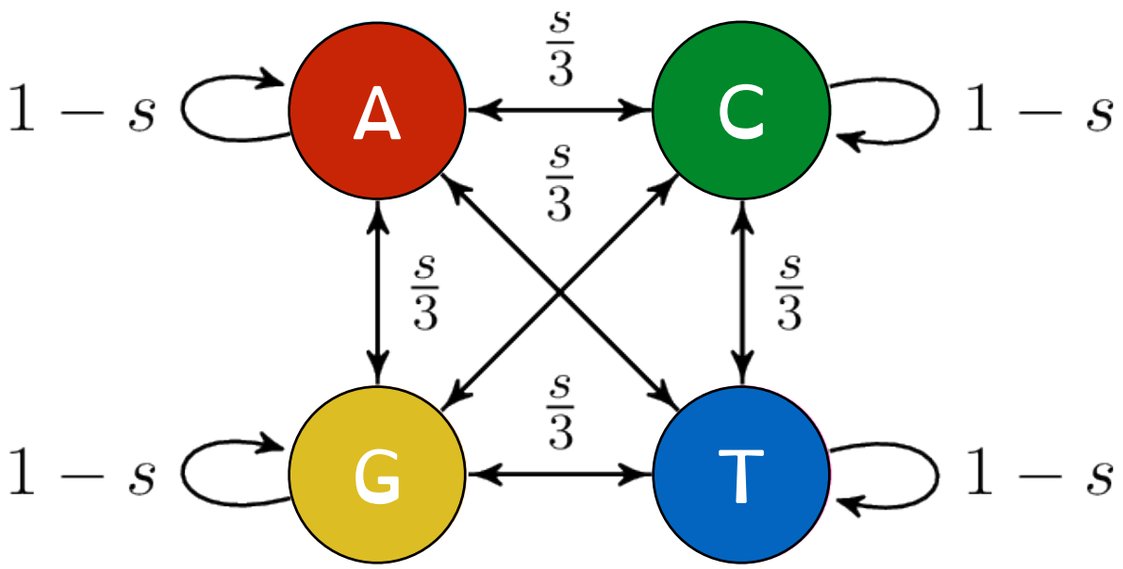
- We have a time machine that can only go back in time at 10,000 year intervals.
- We want to estimate s, the probability that the site will be different the next time we sample.
- For 10 time-travel events, our data might look like: **GGCCCCCCCA**

(based on demo by Mark Holder phylo.bio.ku.edu/mephytis/tex/discrete-time-and-state-Markov.pdf)

Toy Example: Observing a Lineage Evolving



The simplest model



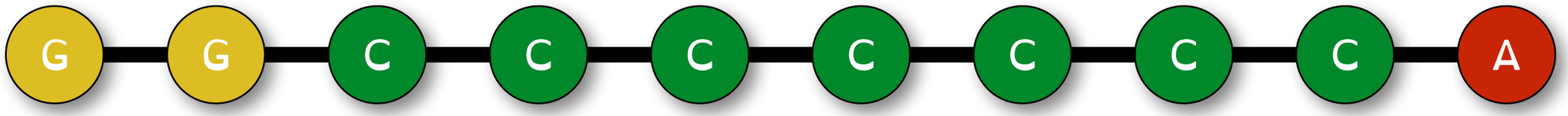
s is the probability of a state switch between samples and is the same over all intervals

each state is equally likely when a switch occurs

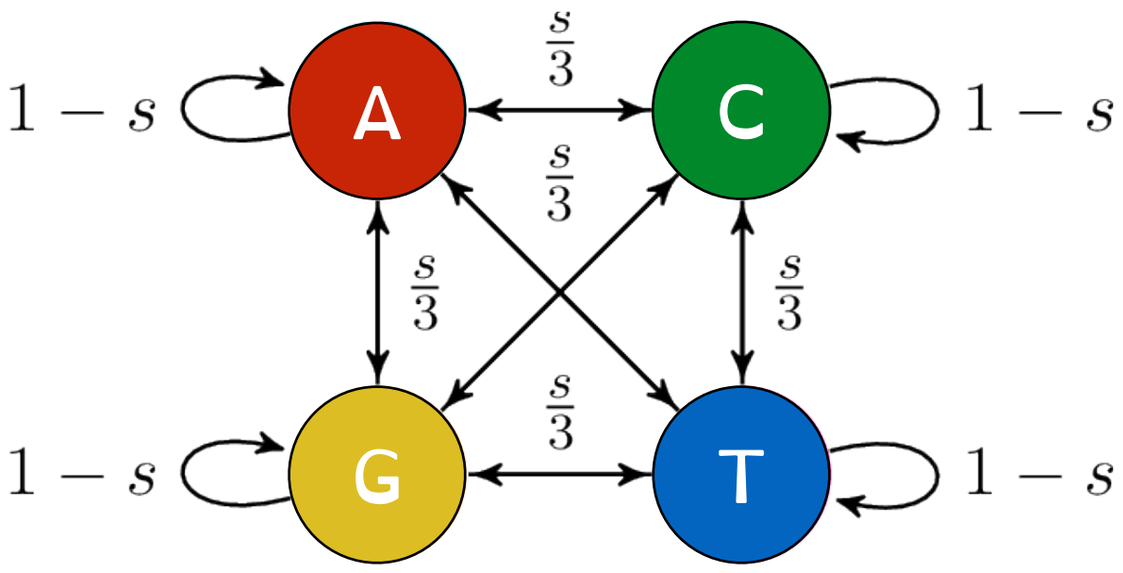
(based on demo by Mark Holder phylo.bio.ku.edu/mephytis/tex/discrete-time-and-state-Markov.pdf)

Toy Example: Observing a Lineage Evolving

x_i



i	1	2	3	4	5	6	7	8	9	10
$\Pr(x_i \mid s)$	$\left(\frac{1}{4}\right)$	$(1-s)$	$\left(\frac{s}{3}\right)$	$(1-s)$	$(1-s)$	$(1-s)$	$(1-s)$	$(1-s)$	$(1-s)$	$\left(\frac{s}{3}\right)$

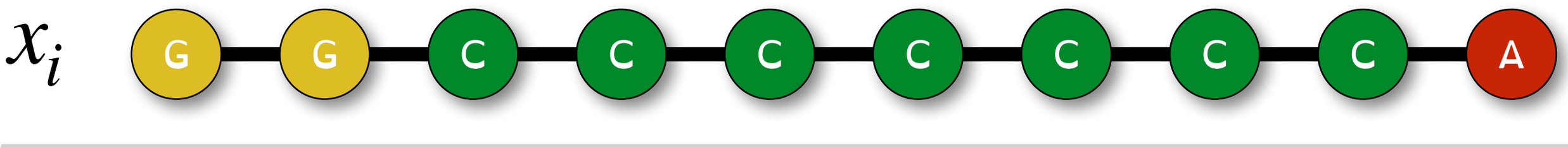


$$\mathbf{X} = \text{GGCCCCCCCCA}$$

$$\Pr(\mathbf{X} \mid s) = ?$$

(based on demo by Mark Holder phylo.bio.ku.edu/mephytis/tex/discrete-time-and-state-Markov.pdf)

Toy Example: Observing a Lineage Evolving



i	1	2	3	4	5	6	7	8	9	10
$\Pr(x_i \mid s)$	$\left(\frac{1}{4}\right)$	$(1-s)$	$\left(\frac{s}{3}\right)$	$(1-s)$	$(1-s)$	$(1-s)$	$(1-s)$	$(1-s)$	$(1-s)$	$\left(\frac{s}{3}\right)$

$$L(s) = \Pr(\mathbf{X} \mid s) = \prod_{i=1}^{n} \Pr(x_i \mid s)$$

$$= \frac{1}{4}(1-s)^{n_s}\left(\frac{s}{3}\right)^{n_d}$$

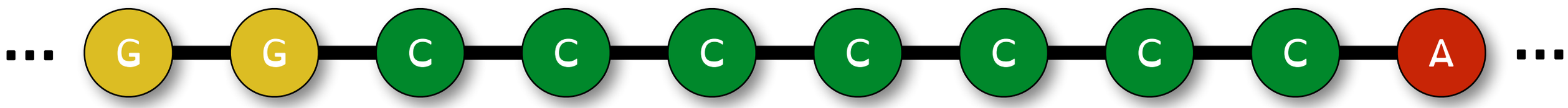
the AND rule !

n_d = number of adjacent sites that are different

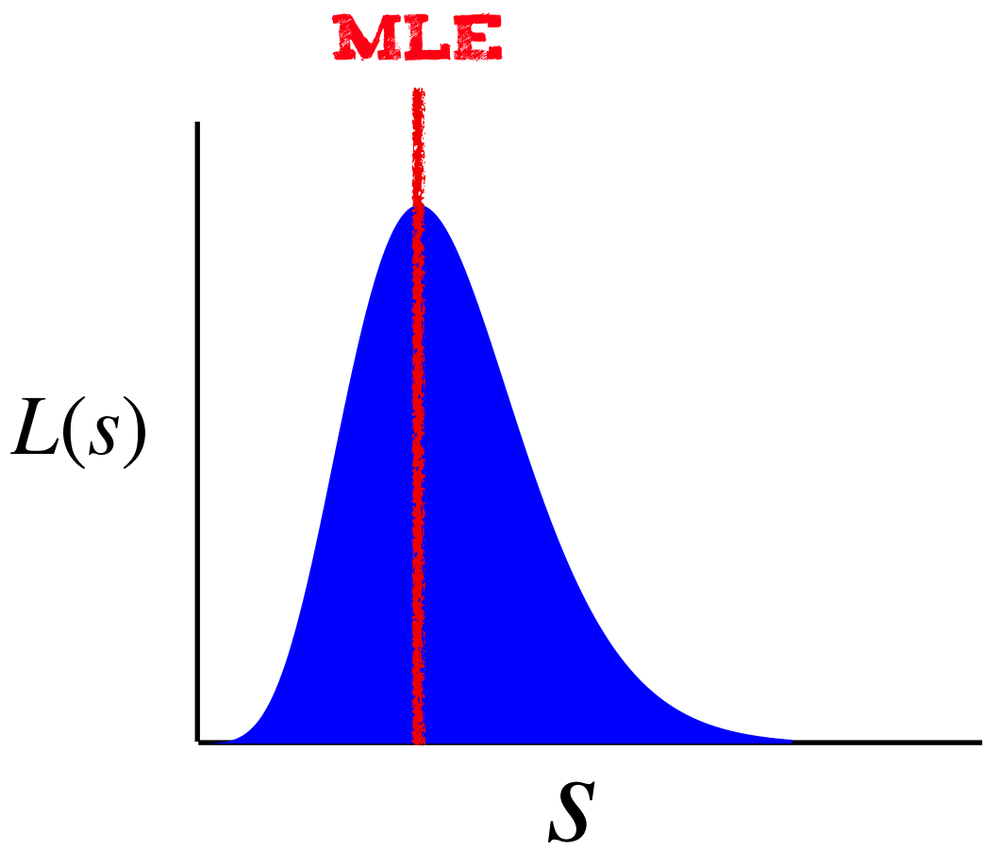
n_s = number of adjacent sites that are the same

(based on demo by Mark Holder phylo.bio.ku.edu/mephytis/tex/discrete-time-and-state-Markov.pdf)

Toy Example: Observing a Lineage Evolving



$$L(s) = \Pr(\mathbf{X} \mid s) = \prod_{i=1}^{n} \Pr(x_i \mid s)$$



With a sequence of samples, we can compute $L(s)$ over a range of plausible values of s to get the maximum likelihood estimate

(based on demo by Mark Holder phylo.bio.ku.edu/mephytis/tex/discrete-time-and-state-Markov.pdf)

Demo:
http://phylo.bio.ku.edu/mephytis/disc-state-disc-time-Markov/index.html

See discrete-time-and-state-Markov.pdf for notes and background information. See the missing data version here

Simulation:

True color switch probability = 0.44

0 0.1 0.2 0.3 0.44 0.5 0.6 0.7 0.8 0.9 1

samples per button click:

1 20 30 40 50 60 70 80 90 100

Simulate: Clear Draw next 1 sample(s)

Simulated data (n = 0):

Inferential stats:

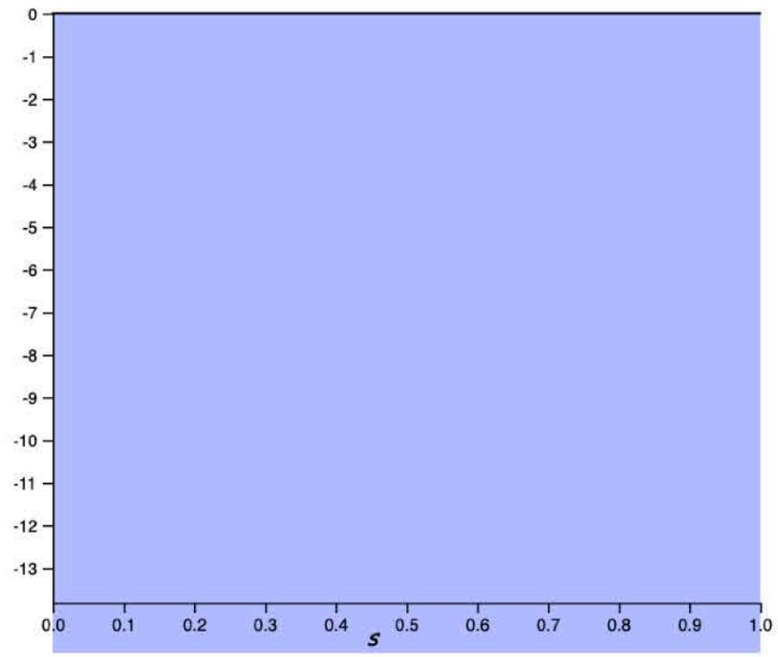
$n - 1 = 0$ # observed transitions

$n_d = 0$ number of color changes

$n_s = 0$ # adjacent samples with same color

$\hat{s} = ?$ estimate of the switch probability

Ln Likelihood Move to probability scale



Back to the demo table of contents...

Source code at https://github.com/mtholder/mephytis

Jukes-Cantor (JC69) Transition Probabilities

the probability that a site starting in state T will end up in state G after time t when the individual substitution rates are all α:

$$p_{TG}(t) = \frac{1}{4}\left(1 - e^{-4\alpha t}\right) = \Pr(G \mid T, \alpha t)$$

JC69 has only 1 unknown parameter: αt

the symbol e represents the base of the natural logarithms: its value is 2.718281828459045...

REMEMBER:

$$\alpha t = \frac{\nu}{3}$$

Jukes-Cantor (JC69) Transition Probabilities

under this model the probability of substituting one state for another is the same for all types

$$p_{AC}(t) = p_{AG}(t) = p_{AT}(t) = p_{CG}(t) = p_{CT}(t) = p_{GT}(t)$$

and the probability of staying at the same state is the same for all states

$$p_{AA}(t) = p_{CC}(t) = p_{GG}(t) = p_{TT}(t) = \frac{1}{4} + \frac{3}{4}e^{-4\alpha t}$$

remember: the process is time-reversible, so $p_{AC}(t) = p_{CA}(t)$

(based on slides by Paul Lewis https://molevolworkshop.github.io/faculty-lewis)

Parameters of Substitution Models

the CTMC substitution models have two important sets of parameters:

equilibrium frequencies

$$\pi_A,\ \pi_C,\ \pi_G,\ \pi_T$$

the long-term stationary distribution of nucleotide states

exchangeability rates

$$r_{AC},\ r_{AG},\ r_{AT},\ r_{CG},\ r_{CT},\ r_{GT}$$

the relative rates of each substitution type

these parameters are combined in the instantaneous rate matrix, which allows us to compute the transition probabilities

Instantaneous Rate Matrix

rows and columns are ordered A, C, G, T

defines the instantaneous rate of change from one base to another for a given substitution model

$$\boldsymbol{Q}_{JC69} = \begin{bmatrix} -3(\alpha) & \alpha & \alpha & \alpha \\ \alpha & -3(\alpha) & \alpha & \alpha \\ \alpha & \alpha & -3(\alpha) & \alpha \\ \alpha & \alpha & \alpha & -3(\alpha) \end{bmatrix}$$

also called the $\boldsymbol{Q}$ matrix and allows us to compute the transition probabilities for any given time t needed to calculate the likelihood of the model

JC69 $\boldsymbol{Q}$ Matrix

free parameters: α

only has 1 free parameter α because the stationary frequencies and exchangeability rates are equal

$$\boldsymbol{Q}_{JC69} = \begin{bmatrix} -3(\alpha) & \alpha & \alpha & \alpha \\ \alpha & -3(\alpha) & \alpha & \alpha \\ \alpha & \alpha & -3(\alpha) & \alpha \\ \alpha & \alpha & \alpha & -3(\alpha) \end{bmatrix}$$

How do you go from $\boldsymbol{Q}$ to the transition probability matrix?

Jukes, T. H., and C. R. Cantor. 1969. Evolution of protein molecules. Pages 21-132 in H. N. Munro (ed.), *Mammalian Protein Metabolism*. Academic Press, New York.

Transition Probability Matrix

for any $\boldsymbol{Q}$ matrix, we can calculate the transition probabilities for all the ways the process can start in one state and end in another, or the same state, after time t

$$\boldsymbol{P}(t) = e^{\boldsymbol{Q}t}$$

we simply have to multiply the matrix by t and then take the exponential

Transition Probability Matrix

we represent the matrix of transition probabilities such that the values are functions that correspond to the probability of going from state i to state j after time t

$$\boldsymbol{P}(t) = e^{\boldsymbol{Q}t} = \begin{bmatrix} p_{AA}(t) & p_{AC}(t) & p_{AG}(t) & p_{AT}(t) \\ p_{CA}(t) & p_{CC}(t) & p_{CG}(t) & p_{CT}(t) \\ p_{GA}(t) & p_{GC}(t) & p_{GG}(t) & p_{GT}(t) \\ p_{TA}(t) & p_{TC}(t) & p_{TG}(t) & p_{TT}(t) \end{bmatrix}$$

every row of $\boldsymbol{P}(t)$ must sum to 1

when $t \rightarrow \infty$, $p_{ij}(t) = 1/4$ because so many substitutions have occurred that the end state is effectively random

JC69 ***P(t)*** Matrix

for JC69, we only have to compute two probabilities:

$p_0(t)$: when the end state is the same as the starting state

$p_1(t)$: when the end state is different from the starting state

$$\boldsymbol{P}_{JC69}(t) = \begin{bmatrix} p_0(t) & p_1(t) & p_1(t) & p_1(t) \\ p_1(t) & p_0(t) & p_1(t) & p_1(t) \\ p_1(t) & p_1(t) & p_0(t) & p_1(t) \\ p_1(t) & p_1(t) & p_1(t) & p_0(t) \end{bmatrix}, \text{where } \begin{cases} p_0(t) = \frac{1}{4} + \frac{3}{4}e^{-4\alpha t} \\ p_1(t) = \frac{1}{4} - \frac{1}{4}e^{-4\alpha t} \end{cases}$$

for more detail on this and other substitution models see:

Yang (2014) *Molecular Evolution: A Statistical Approach*, Chapter 1

REMEMBER:

$$\alpha t = \frac{\nu}{3}$$

K80 (aka K2P) $\boldsymbol{Q}$ Matrix

free parameters: α, β

accounts for biases in types of substitutions based on biochemical properties

has 2 free parameters: the transition rate (α) and transversion rate (β)

$$\boldsymbol{Q}_{K80} = \begin{bmatrix} -\alpha - 2\beta & \beta & \alpha & \beta \\ \beta & -\alpha - 2\beta & \beta & \alpha \\ \alpha & \beta & -\alpha - 2\beta & \beta \\ \beta & \alpha & \beta & -\alpha - 2\beta \end{bmatrix}$$

Kimura, M. 1980. A simple method for estimating evolutionary rate of base substitutions through comparative studies of nucleotide sequences. *Journal of Molecular Evolution* 16:111-120.

transition: A⟷G or C⟷T

transversion: A⟷C, C⟷G, A⟷T,

K80 (aka K2P) - Re-parameterized

you will commonly see this model parameterized so that our parameter of interest is the transition-transversion rate ratio (κ)

free parameters: κ and β

$$\boldsymbol{Q}_{K80} = \begin{bmatrix} -\beta(\kappa+2) & \beta & \kappa\beta & \beta \\ \beta & -\beta(\kappa+2) & \beta & \kappa\beta \\ \kappa\beta & \beta & -\beta(\kappa+2) & \beta \\ \beta & \kappa\beta & \beta & -\beta(\kappa+2) \end{bmatrix}$$

K80 collapses to JC69 if κ = 1, i.e., α=β

$$\kappa = \frac{\alpha}{\beta}$$

Kimura, M. 1980. A simple method for estimating evolutionary rate of base substitutions through comparative studies of nucleotide sequences. *Journal of Molecular Evolution* 16:111-120.

F81 *Q* Matrix

free parameters: α, π_A, π_C, π_G

equal exchangeability rates and unequal equilibrium base frequencies

$$Q_{F81} = \begin{bmatrix} -\alpha(1-\pi_A) & \pi_C\alpha & \pi_G\alpha & \pi_T\alpha \\ \pi_A\alpha & -\alpha(1-\pi_C) & \pi_G\alpha & \pi_T\alpha \\ \pi_A\alpha & \pi_C\alpha & -\alpha(1-\pi_G) & \pi_T\alpha \\ \pi_A\alpha & \pi_C\alpha & \pi_G\alpha & -\alpha(1-\pi_T) \end{bmatrix}$$

F81 collapses to JC69 if $\pi_A = \pi_C = \pi_G = \pi_T$

Felsenstein, J. 1981. Evolutionary trees from DNA sequences: a maximum likelihood approach. *Journal of Molecular Evolution*

HKY85 *Q* Matrix

free parameters: κ, β, π_A, π_C, π_G

unequal rates for transitions and transversions and unequal equilibrium base frequencies

$$Q_{HKY85} = \begin{bmatrix} - & \pi_C\beta & \pi_G\beta\kappa & \pi_T\beta \\ \pi_A\beta & - & \pi_G\beta & \pi_T\beta\kappa \\ \pi_A\beta\kappa & \pi_C\beta & - & \pi_T\beta \\ \pi_A\beta & \pi_C\beta\kappa & \pi_G\beta & - \end{bmatrix}$$

the dash just means that the value that goes here makes the row sum to 0

HKY collapses to F81 if $\kappa = 1$

HKY collapses to K80 if $\pi_A = \pi_C = \pi_G = \pi_T$

Hasegawa, M., H. Kishino, and T. Yano. 1985. Dating the human-ape splitting by a molecular clock of mitochondrial DNA. *Journal of Molecular Evolution* 21:160-174.

(based on slides by Paul Lewis https://molevolworkshop.github.io/faculty-lewis)

GTR *Q* Matrix

free parameters: μ, π_A, π_C, π_G, *a, b, c, d, e*

the most general model: unequal rates for all substitution types and unequal equilibrium base frequencies

$$Q_{GTR} = \begin{bmatrix} - & \pi_C a\mu & \pi_G b\mu & \pi_T c\mu \\ \pi_A a\mu & - & \pi_G d\mu & \pi_T e\mu \\ \pi_A b\mu & \pi_C d\mu & - & \pi_T f\mu \\ \pi_A c\mu & \pi_C e\mu & \pi_G f\mu & - \end{bmatrix}$$

GTR collapses to HKY if $a = c = d = f = \beta$ *and* $b = e = \kappa\beta$

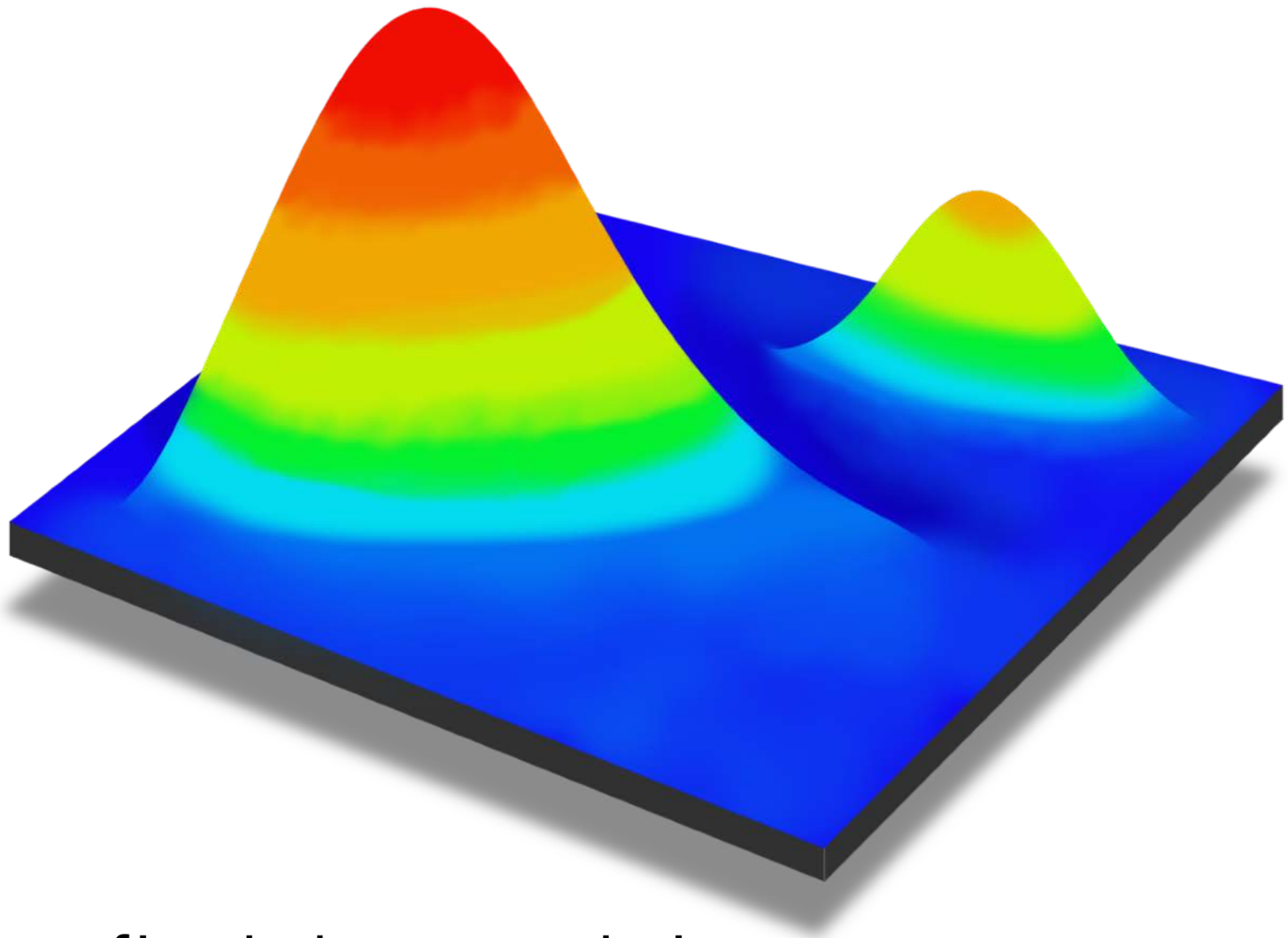
GTR collapses to F81 if: $a = b = c = d = e = f = 1$

Lanave, C., G. Preparata, C. Saccone, and G. Serio. 1984. A new method for calculating evolutionary substitution rates. *Journal of Molecular Evolution*

(based on slides by Paul Lewis https://molevolworkshop.github.io/faculty-lewis)

How do I find the best tree?

we can use the maximum likelihood as an optimality criterion to search for the best tree topology, branch lengths, and model parameters that fit our data



How can we find the peak in our parameter space?

Maximum Likelihood Methods

finding the tree with the highest likelihood is difficult

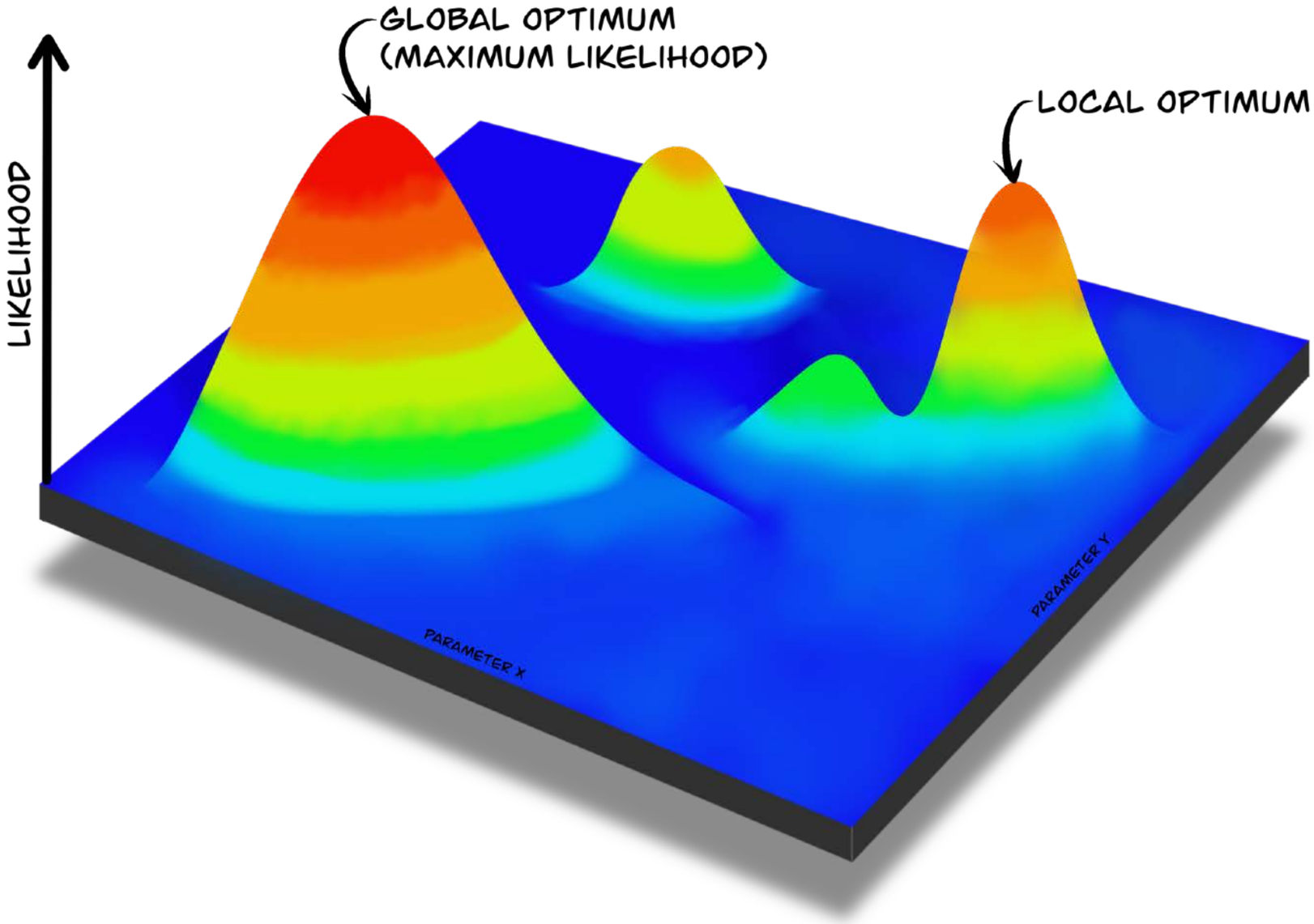
there is a vast number of possible tree topologies for even small datasets

the likelihood “landscape” may have more than one peak (local optima)

we not only have to find the tree topology that maximizes the likelihood, we also have to find branch lengths and values for the other free parameters of our substitution model

Heuristic Search Methods

to traverse state space, we use heuristic searching methods to attempt to find the global optimum set of parameter values



for a 20-taxon dataset under GTR we need to optimize: the tree topology, μ, π_A, π_C, π_G, *a, b, c, d, e,* and 37 branch lengths

(based on slides by Emily Jane McTavish https://molevolworkshop.github.io/faculty-mctavish)

Heuristic Search Methods

the general concept of a heuristic tree search

(based on slides by Emily Jane McTavish https://molevolworkshop.github.io/faculty-mctavish)

Heuristic Search Runtimes

heuristic searches can lead to long runtimes depending on:

- the number of tree topologies to evaluate
- the time to compute the likelihood

these are a function of the number of sequences and the number of characters in the data matrix (molecular sites or morphological characters)

Many More Heuristic Search Algorithms

- swapping need not include all neighbors (RAxML, reconlimit in PAUP*)
- “lazy” scoring of swaps (RAxML)
- ignoring (at some stage) interactions between different branch swaps (PHYML)
- stochastic searches
  - genetic algorithms (GAML, MetaPIGA, GARLI)
  - simulated annealing
- divide and conquer methods (the sectortial searching of Goloboff, 1999; Rec-I-DCM3 Roshan 2004)
- data perturbation methods (e.g. Kevin Nixon’s “ratchet”)