Phylogenetics:
Bayesian Phylogenetic Analysis

COMP 571
Luay Nakhleh, Rice University

Bayes Rule

$$\mathbf{P}(X = x|Y = y) = \frac{\mathbf{P}(X = x, Y = y)}{\mathbf{P}(Y = y)} = \frac{\mathbf{P}(X = x)\mathbf{P}(Y = y|X = x)}{\sum_{x'} \mathbf{P}(X = x')\mathbf{P}(Y = y|X = x')}$$

Bayes Rule

* Example (from "Machine Learning: A Probabilistic Perspective")
* Consider a woman in her 40s who decides to have a mammogram.
* Question: If the test is positive, what is the probability that she has cancer?
* The answer depends on how reliable the test is!

Bayes Rule

* Suppose the test has a sensitivity of 80%; that is, if a person has cancer, the test will be positive with probability 0.8.
* If we denote by x=1 the event that the mammogram is positive, and by y=1 the event that the person has breast cancer, then P(x=1|y=1)=0.8.

Bayes Rule

* Does the probability that the woman in our example (who tested positive) has cancer equal 0.8?

Bayes Rule

* No!

* That ignores the prior probability of having breast cancer, which, fortunately, is quite low: p(y=1)=0.004

Bayes Rule

* Further, we need to take into account the fact that the test may be a false positive.
* Mammograms have a false positive probability of $p(x=1|y=0)=0.1$.

Bayes Rule

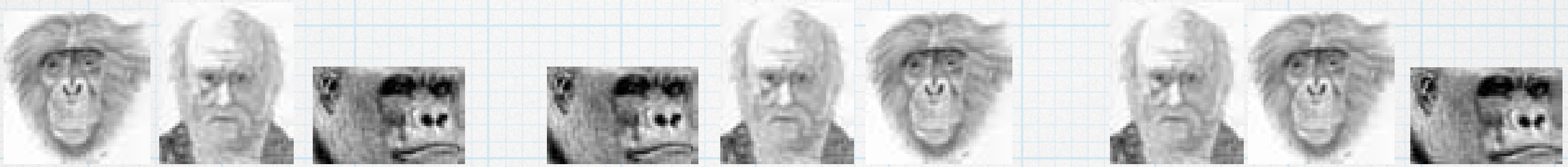
* Combining all these facts using Bayes rule, we get (using p(y=0)=1-p(y=1)):

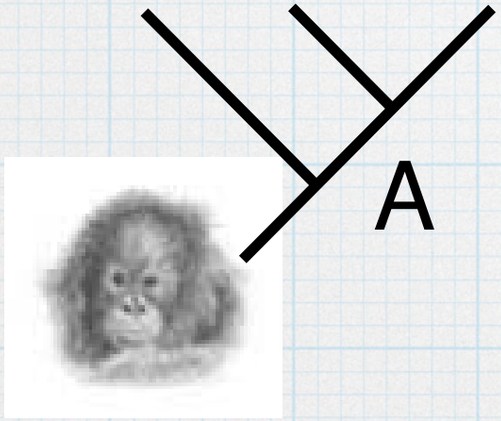
$$
\begin{aligned}
p(y=1|x=1) &= \frac{p(x=1|y=1)p(y=1)}{p(x=1|y=1)p(y=1)+p(x=1|y=0)p(y=0)} \\
&= \frac{0.8\times 0.004}{0.8\times 0.004+0.1\times 0.996} \\
&= 0.031
\end{aligned}
$$

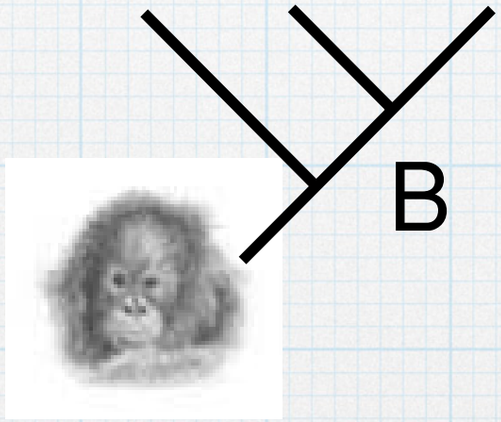
* How does Bayesian reasoning apply to phylogenetic inference?

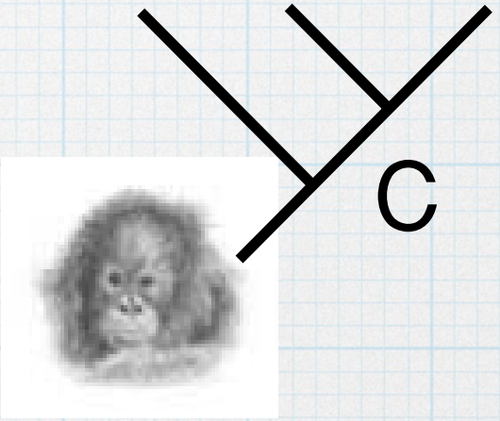
* Assume we are interested in the relationships between human, gorilla, and chimpanzee (with orangutan as an outgroup).

* There are clearly three possible relationships.



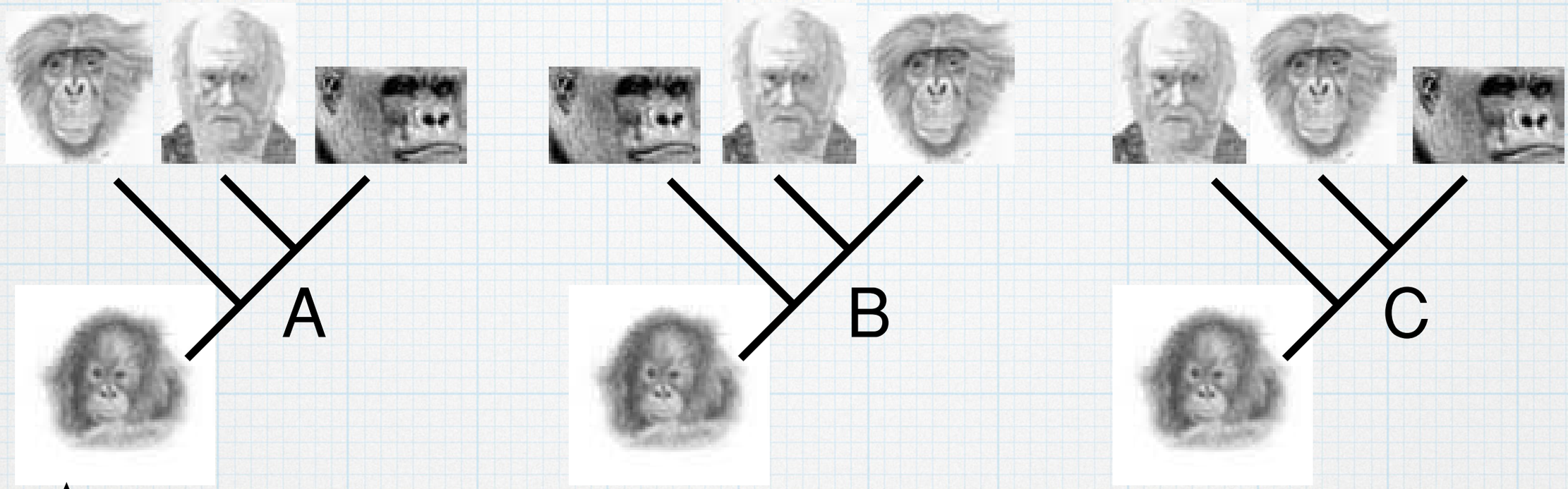

A

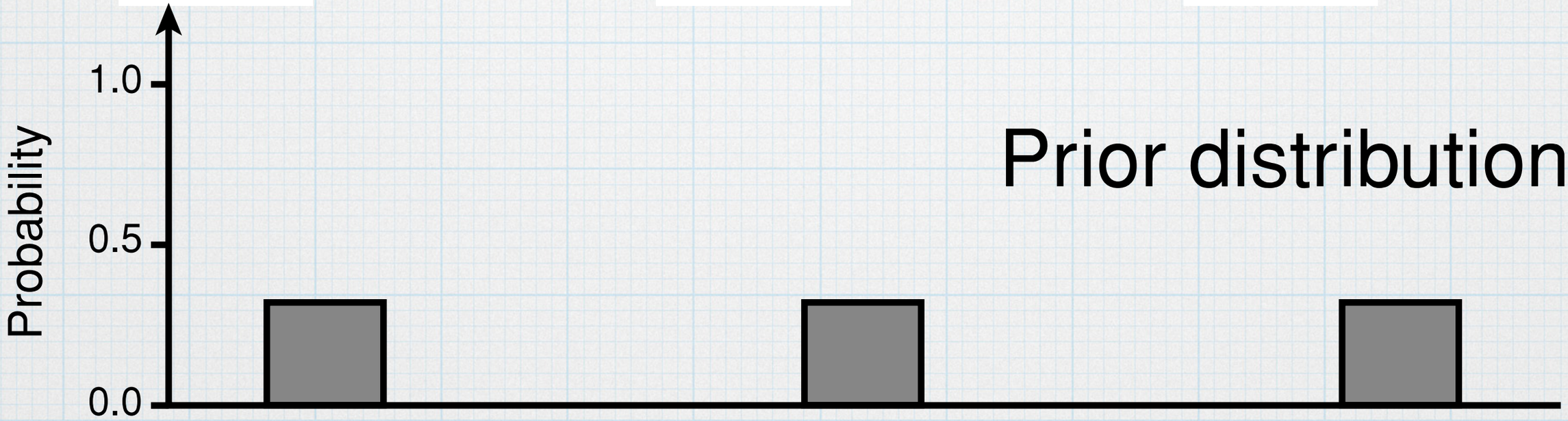

B


C

* Before the analysis, we need to specify our prior beliefs about the relationships.

* For example, in the absence of background data, a simple solution would be to assign equal probability to the possible trees.





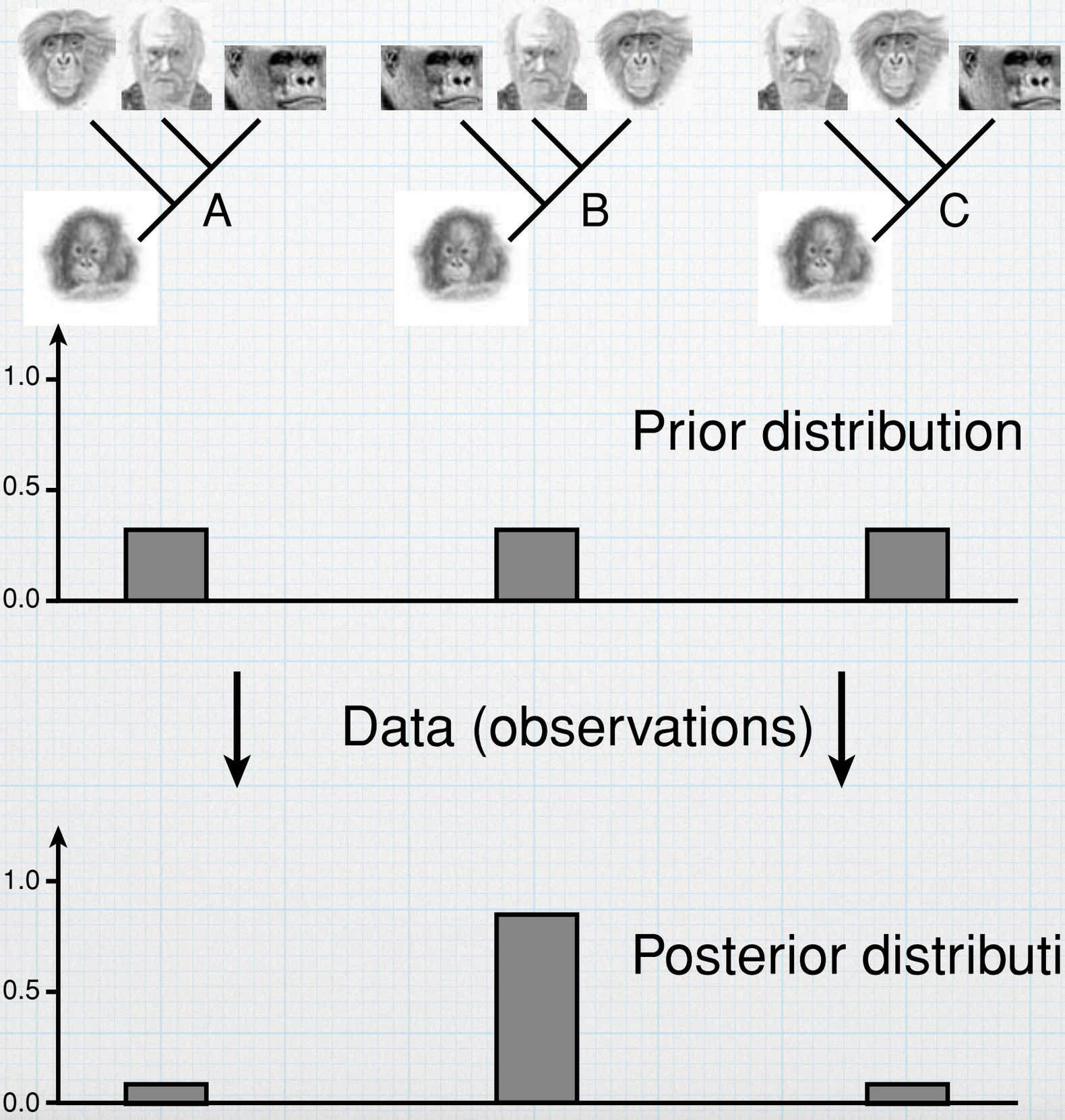
[This is an uninformative prior]

* To update the prior, we need some **data**, typically in the form of a **molecular sequence alignment**, and a **stochastic model of the process generating the data on the tree.**

* In principle, Bayes rule is then used to obtain the posterior probability distribution, which is the result of the analysis.

* The posterior specifies the probability of each tree given the model, the prior, and the data.

* When the data are informative, most of the posterior probability is typically concentrated on one tree (or, a small subset of trees in a large tree space).


A
B
C
1.0
0.5
0.0
Probability
Prior distribution
Data (observations)
1.0
0.5
0.0
Probability
Posterior distribution

* To describe the analysis mathematically, consider:
  * the matrix of aligned sequences X
  * the tree topology parameter τ
  * the branch lengths of the tree v
  * (typically, substitution model parameters are also included)

Let $\theta=(\tau,v)$

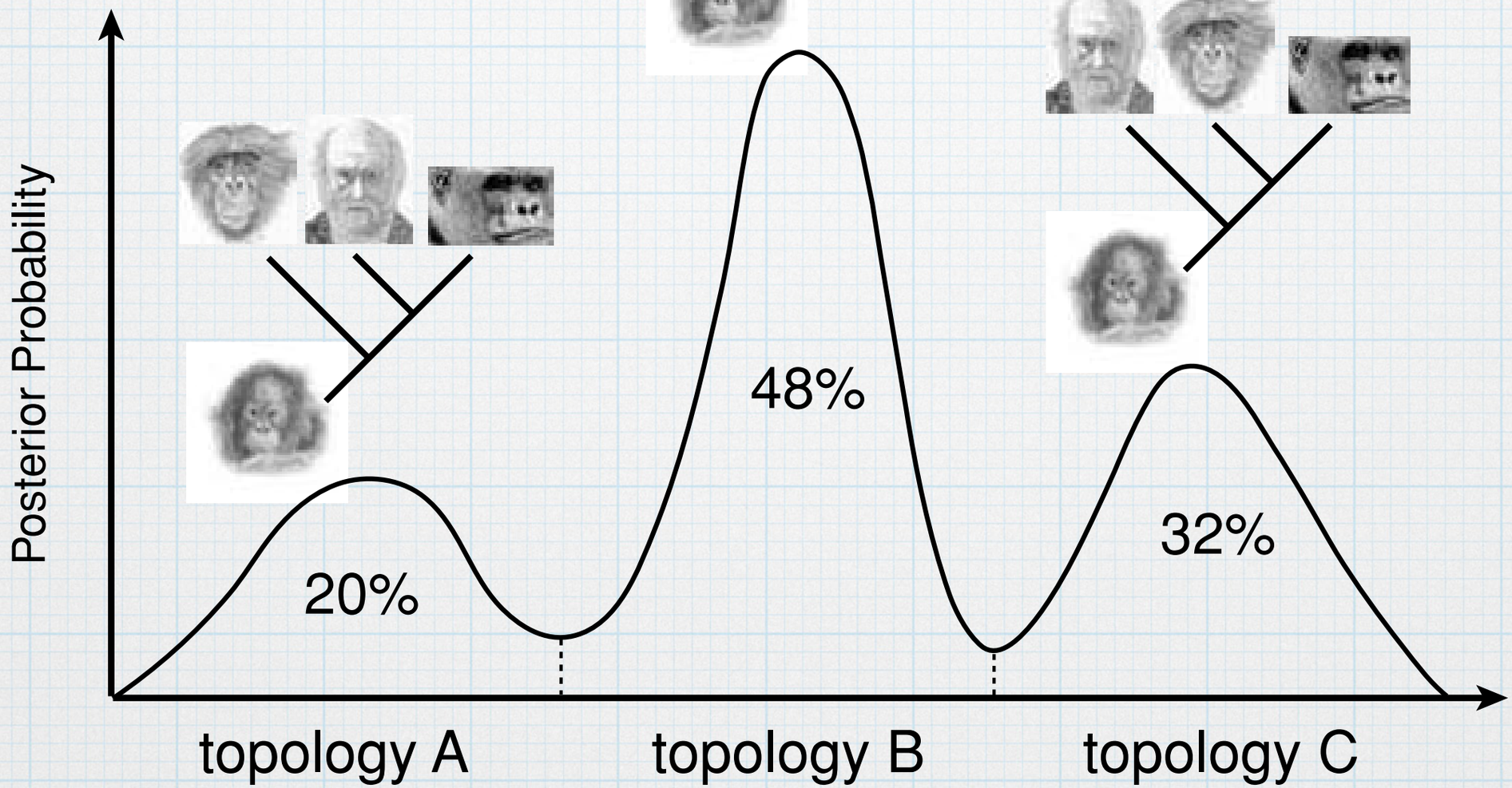
* Bayes theorem allows us to derive the posterior distribution as

$$f(\theta|X) = \frac{f(\theta)f(X|\theta)}{f(X)}$$

where

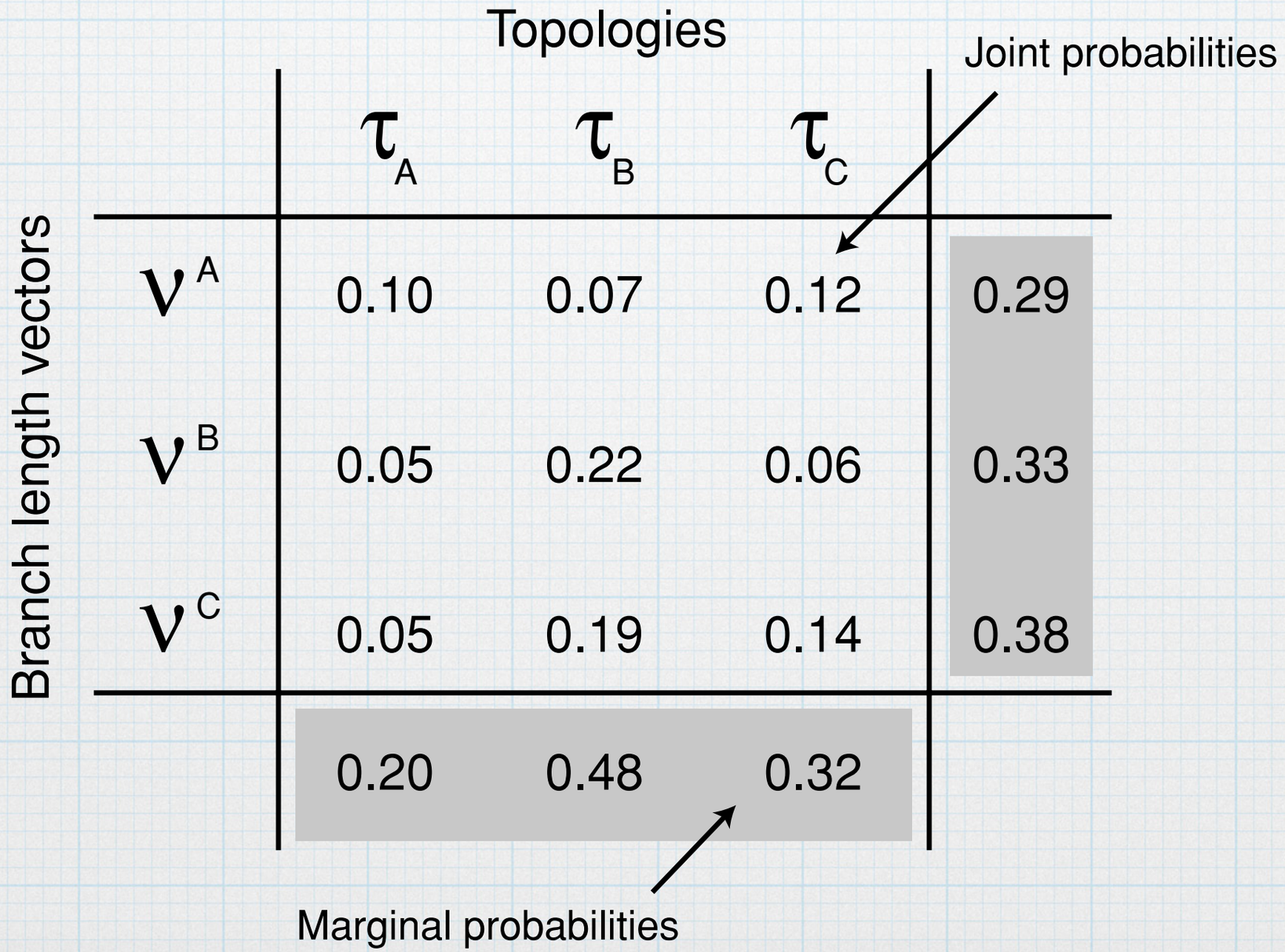
$$f(X) = \int f(\theta)\, f(X|\theta)\, \mathrm{d}\theta$$

$$= \sum_{\tau} \int_{v} f(v)\, f(X|\tau, v)\, \mathrm{d}v$$



The marginal probability distribution on topologies

* Why are they called marginal probabilities?



Markov chain Monte Carlo Sampling

* In most cases, it is impossible to derive the posterior probability distribution analytically.

* Even worse, we can't even estimate it by drawing random samples from it.

* The reason is that most of the posterior probability is likely to be concentrated in a small part of a vast parameter space.

* The solution is to estimate the posterior probability distribution using **Markov chain Monte Carlo sampling**, or **MCMC** for short.

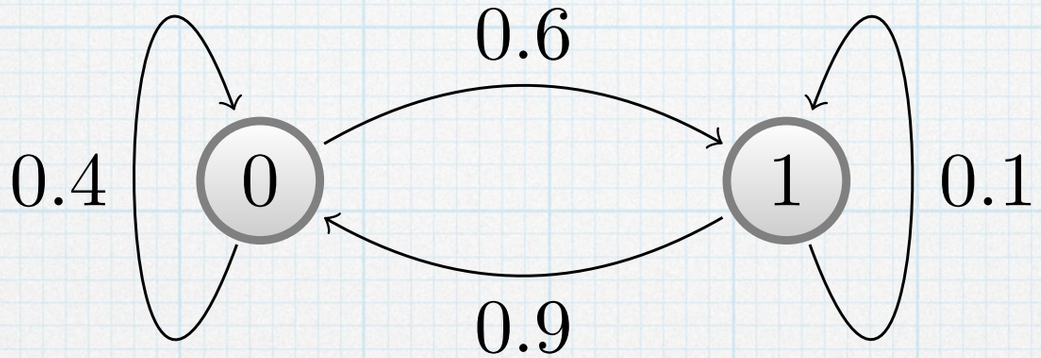
* Monte Carlo = random simulation

* Markov chain = the state of the simulator depends only on the current state

* Irreducible Markov chains (their topology is strongly connected) have the property that they converge towards an equilibrium state (stationary distribution) regardless of starting point.

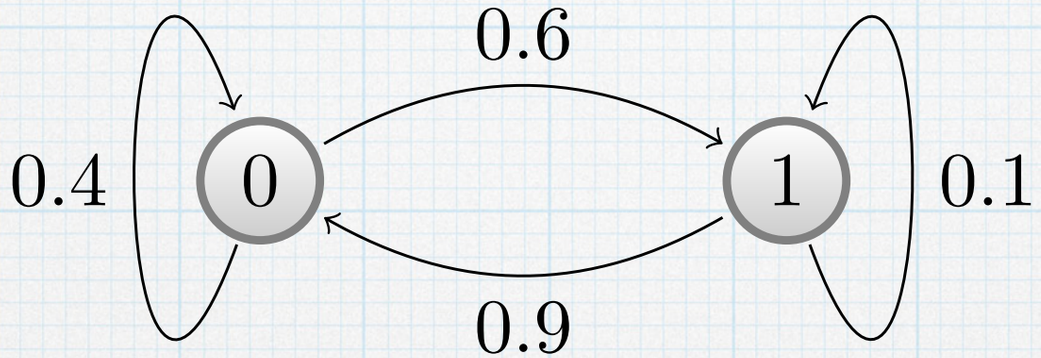
* We just need to set up a Markov chain that converges onto our posterior probability distribution!

Stationary Distribution of a Markov Chain



$$
\begin{aligned}
\mathbb{P}(x_{i+1} = 0 | x_i = 0) &= 0.4 \\
\mathbb{P}(x_{i+1} = 1 | x_i = 0) &= 0.6 \\
\mathbb{P}(x_{i+1} = 0 | x_i = 1) &= 0.9 \\
\mathbb{P}(x_{i+1} = 1 | x_i = 1) &= 0.1
\end{aligned}
$$

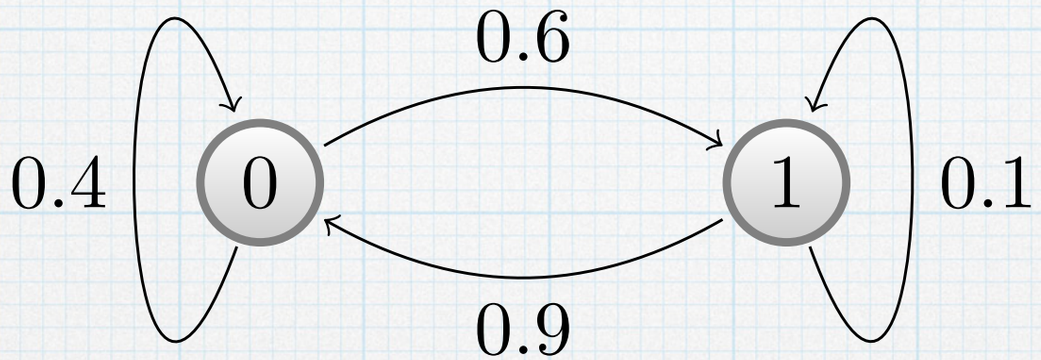
Stationary Distribution of a Markov Chain



$$\begin{aligned}
\mathbb{P}(x_{i+1} = 0 | x_i = 0) &= 0.4 \\
\mathbb{P}(x_{i+1} = 1 | x_i = 0) &= 0.6 \\
\mathbb{P}(x_{i+1} = 0 | x_i = 1) &= 0.9 \\
\mathbb{P}(x_{i+1} = 1 | x_i = 1) &= 0.1
\end{aligned}$$

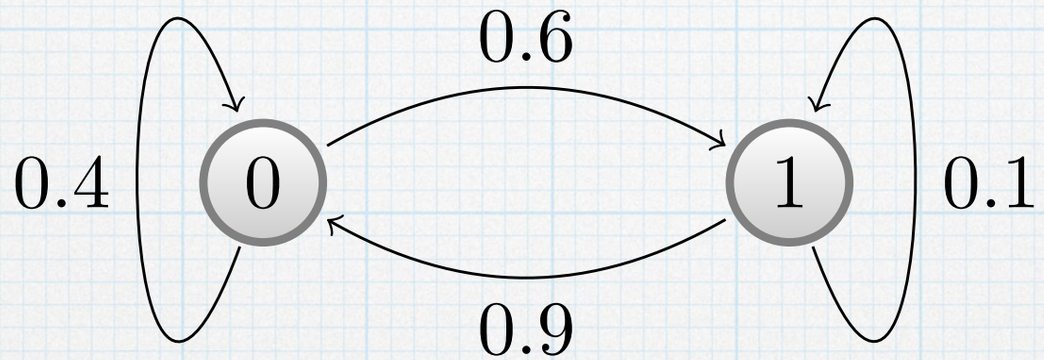
What are $\mathbb{P}(x_i = 0 | x_0 = 0)$ $\mathbb{P}(x_i = 1 | x_0 = 0)$ $\mathbb{P}(x_i = 0 | x_0 = 1)$ $\mathbb{P}(x_i = 1 | x_0 = 1)$ **?**

Stationary Distribution of a Markov Chain



$$\mathbb{P}(x_i = k | x_0 = \ell) = \mathbb{P}(x_i = k | x_{i-1} = 0)\mathbb{P}(x_{i-1} = 0 | x_0 = \ell) + \mathbb{P}(x_i = k | x_{i-1} = 1)\mathbb{P}(x_{i-1} = 1 | x_0 = \ell)$$

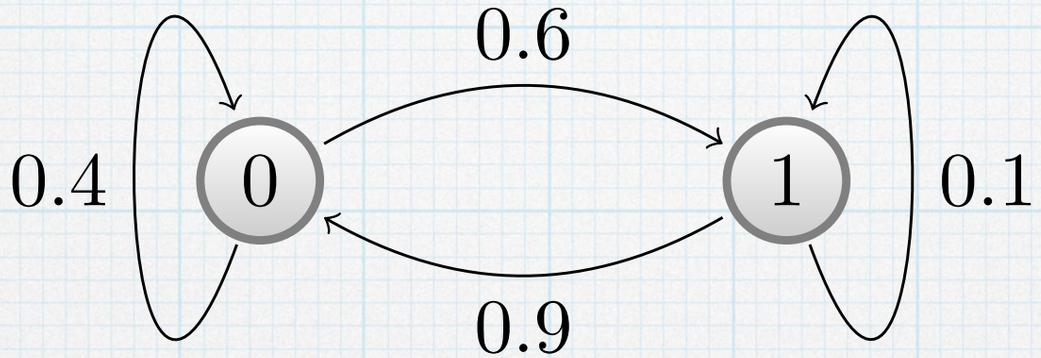
Stationary Distribution of a Markov Chain

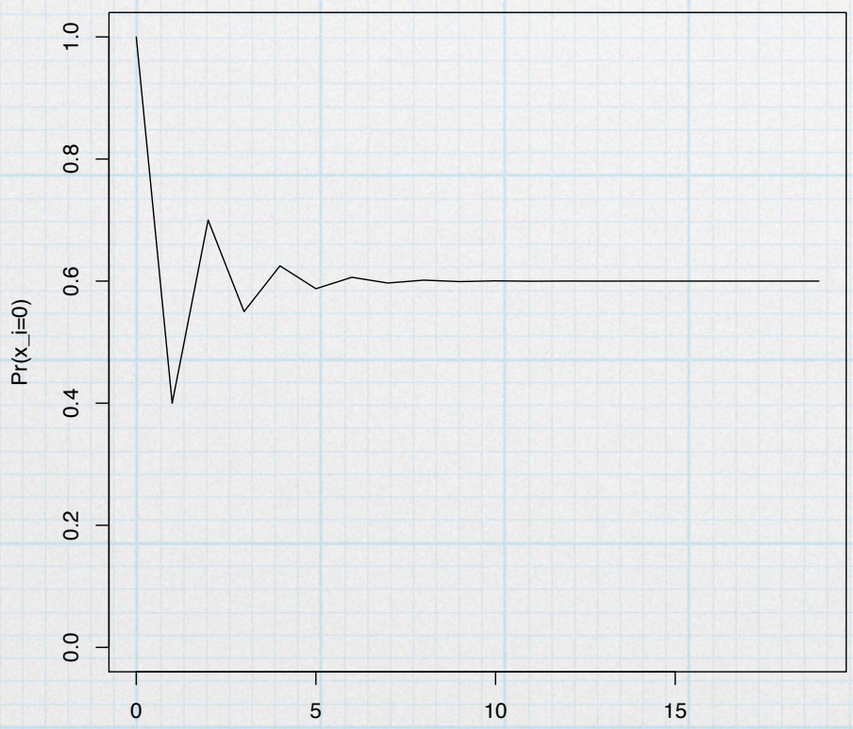


$$\mathbb{P}(x_i = k | x_0 = \ell) = \mathbb{P}(x_i = k | x_{i-1} = 0)\mathbb{P}(x_{i-1} = 0 | x_0 = \ell) + \mathbb{P}(x_i = k | x_{i-1} = 1)\mathbb{P}(x_{i-1} = 1 | x_0 = \ell)$$

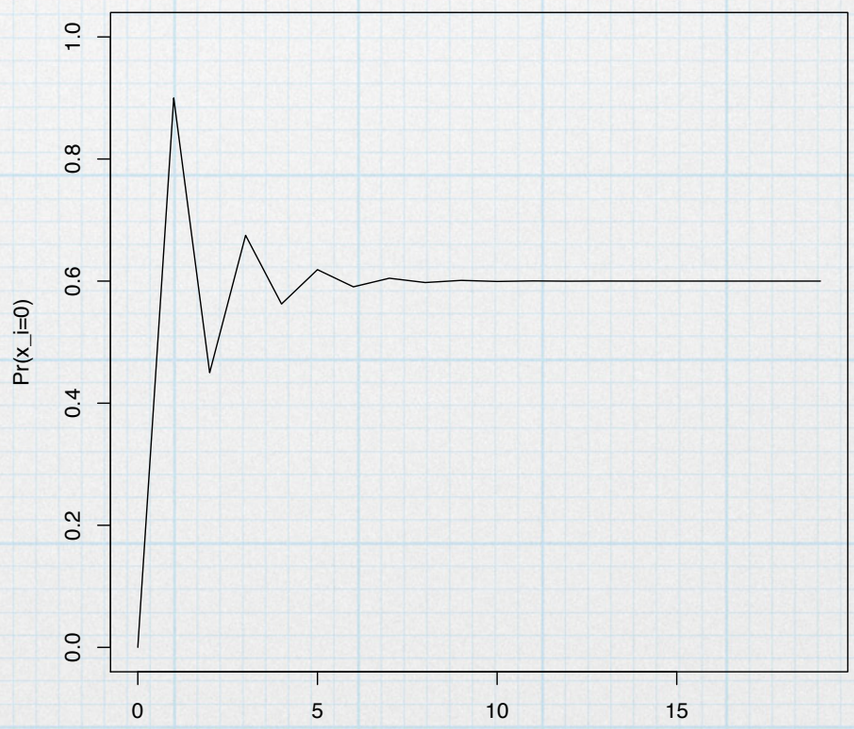
transition probabilities

Stationary Distribution of a Markov Chain



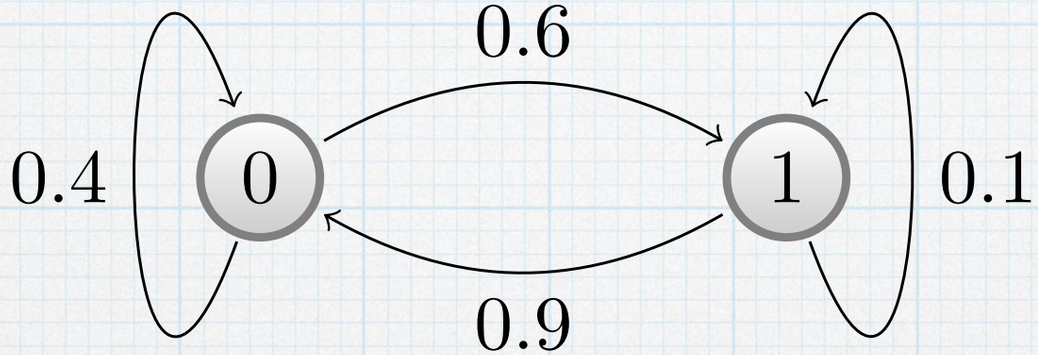


$\mathbb{P}(x_i = 0 | x_0 = 0)$

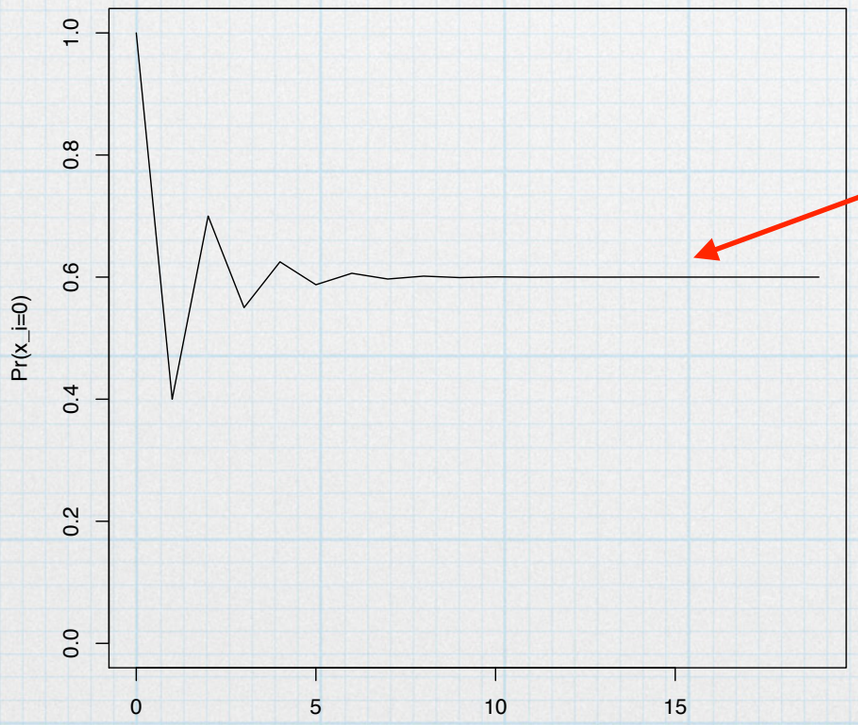


$\mathbb{P}(x_i = 0 | x_0 = 1)$

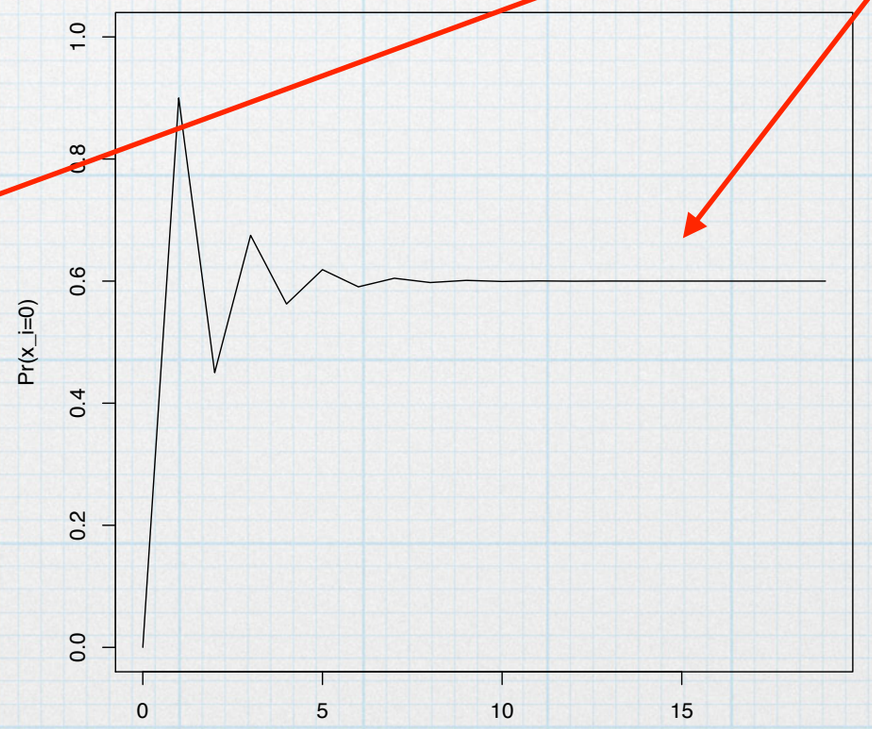
Stationary Distribution of a Markov Chain



same probability regardless of starting state!

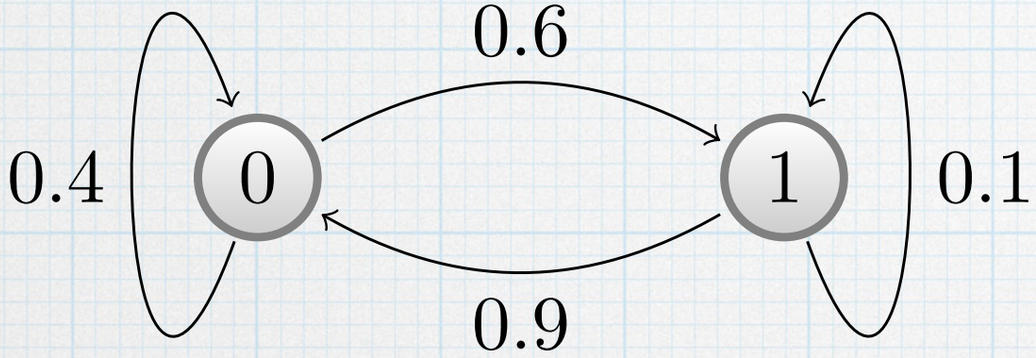


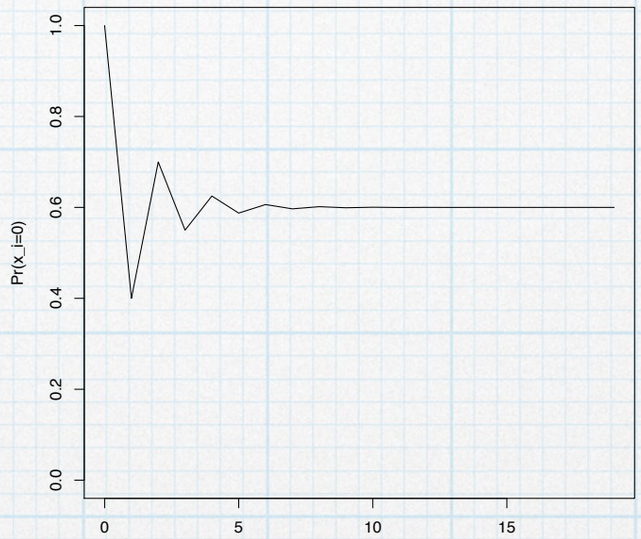
$\mathbb{P}(x_i = 0 | x_0 = 0)$



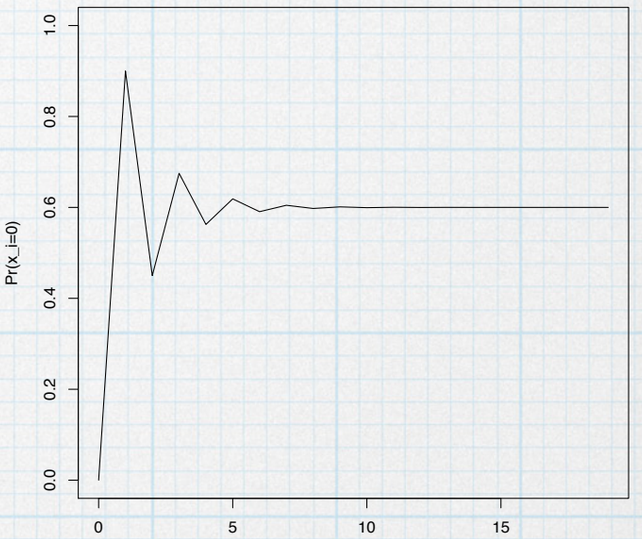
$\mathbb{P}(x_i = 0 | x_0 = 1)$

Stationary Distribution of a Markov Chain



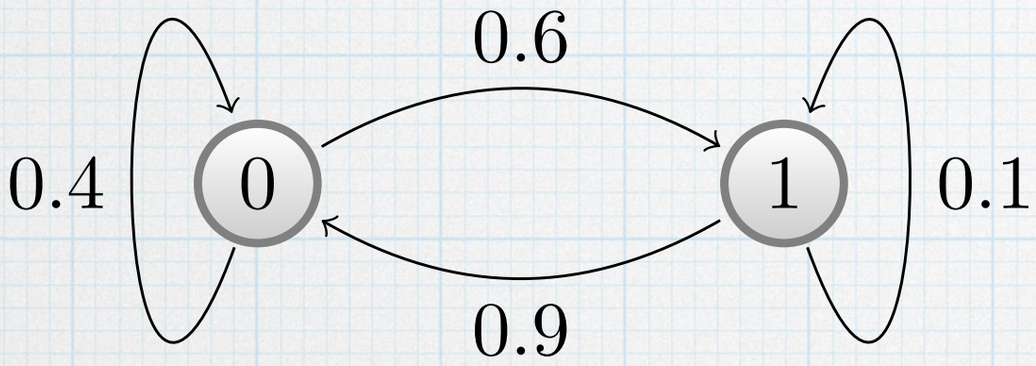


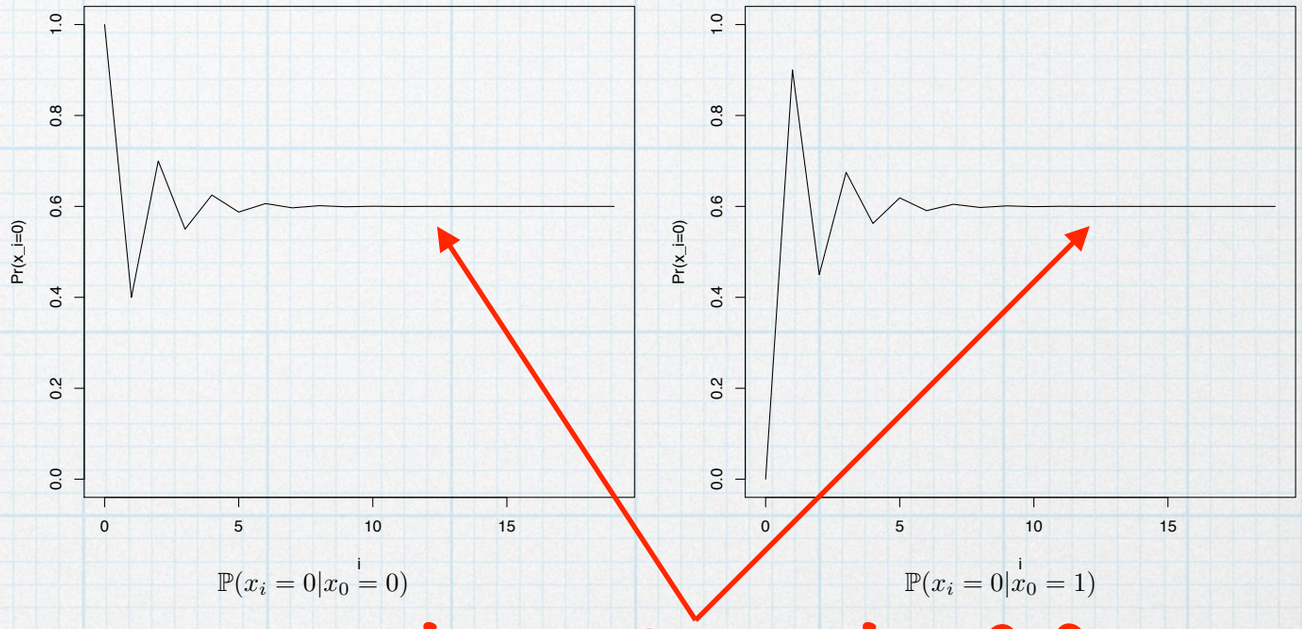
$\mathbb{P}(x_i = 0 | x_0 = 0)$



$\mathbb{P}(x_i = 0 | x_0 = 1)$

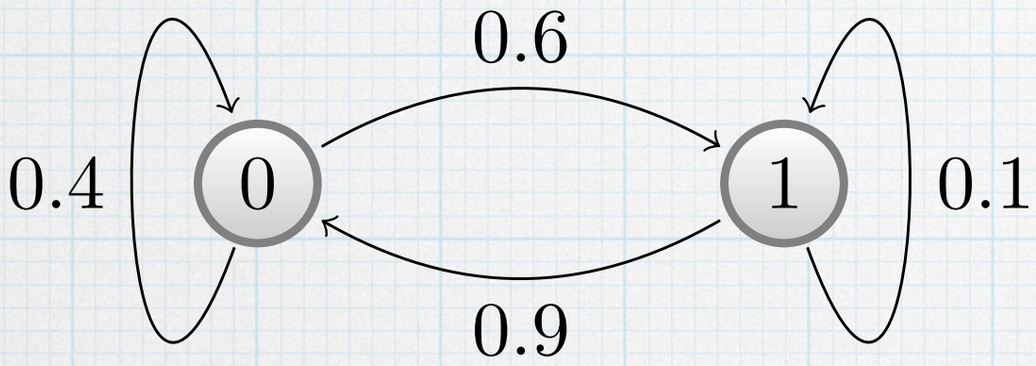
Stationary Distribution of a Markov Chain

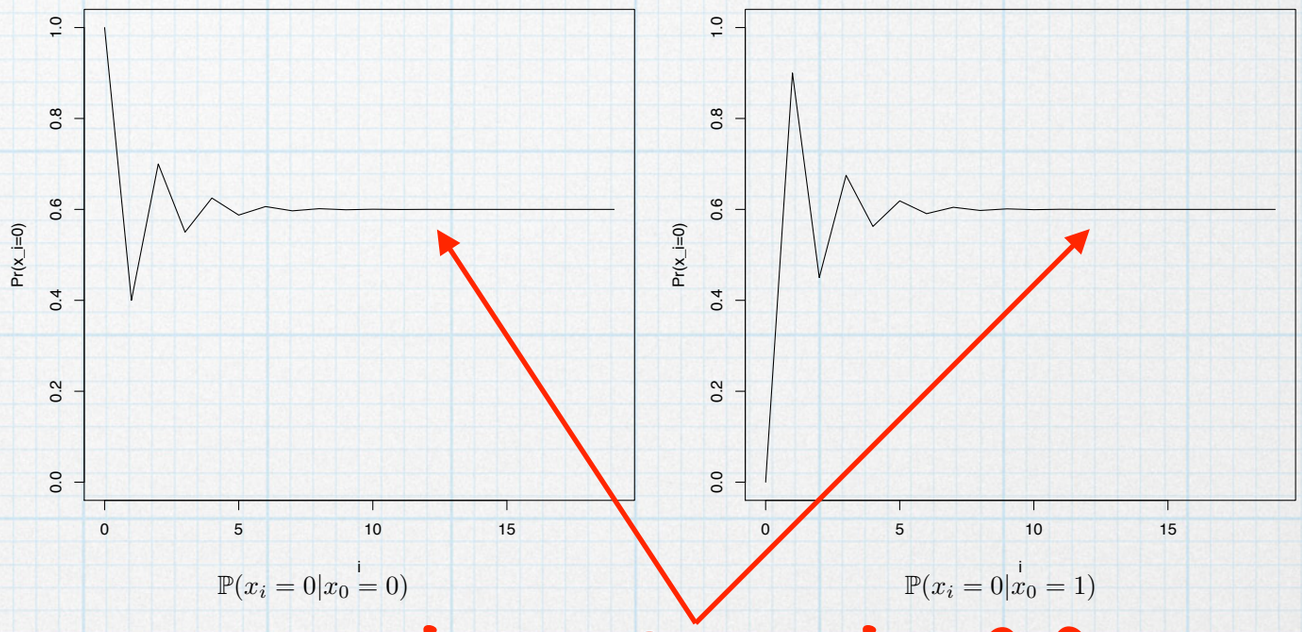




where does the 0.6 come from?

Stationary Distribution of a Markov Chain

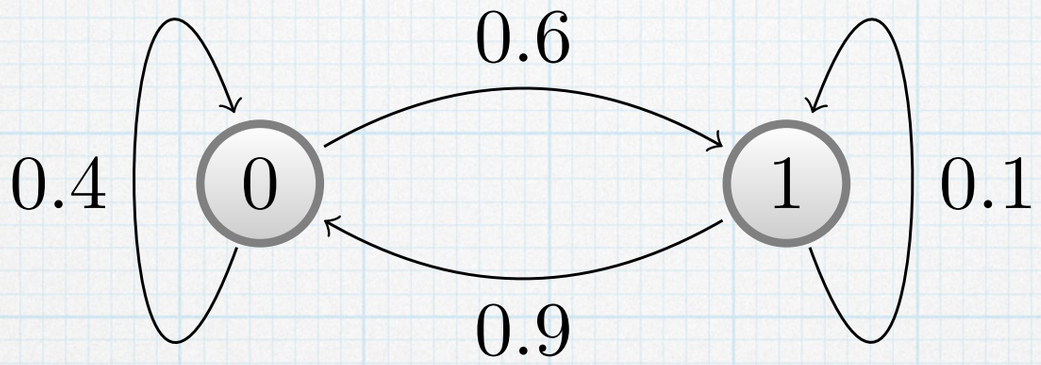




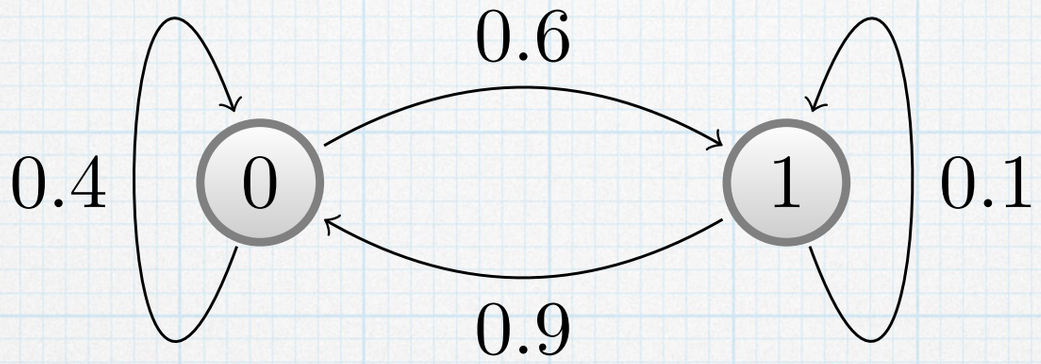
where does the 0.6 come from?

stationary distribution: π_0=0.6 π_1=0.4

Stationary Distribution of a Markov Chain

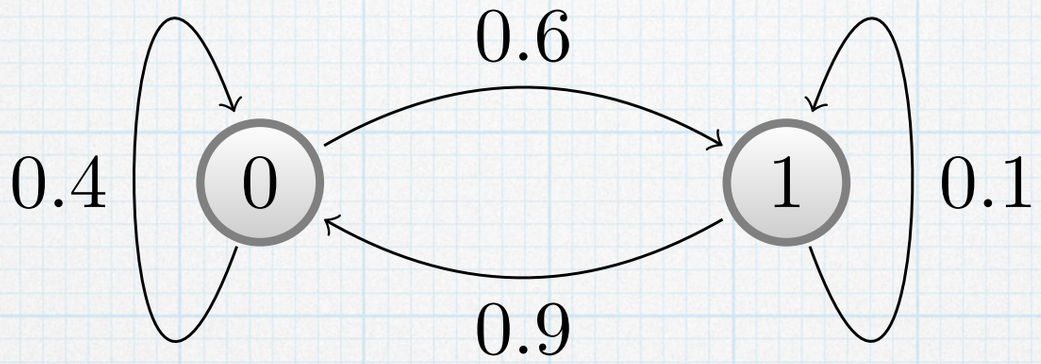


Stationary Distribution of a Markov Chain



Imagine infinitely many chains. At equilibrium (steady-state), the "flux out" of each state must be equal to the "flux into" that state.

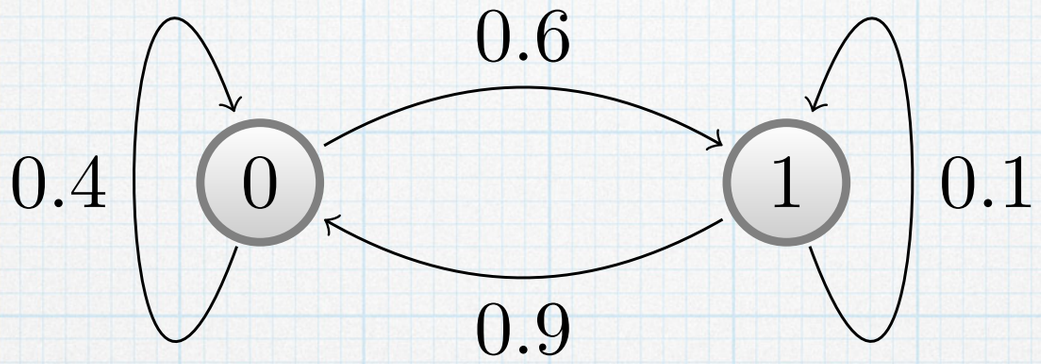
Stationary Distribution of a Markov Chain



Imagine infinitely many chains. At equilibrium (steady-state), the "flux out" of each state must be equal to the "flux into" that state.

$$\begin{aligned} \boldsymbol{\pi}_0 \mathbb{P}(x_{i+1} = 1 | x_i = 0) &= \boldsymbol{\pi}_1 \mathbb{P}(x_{i+1} = 0 | x_i = 1) \\ \frac{\boldsymbol{\pi}_0}{\boldsymbol{\pi}_1} &= \frac{\mathbb{P}(x_{i+1} = 0 | x_i = 1)}{\mathbb{P}(x_{i+1} = 1 | x_i = 0)} \end{aligned}$$

Stationary Distribution of a Markov Chain

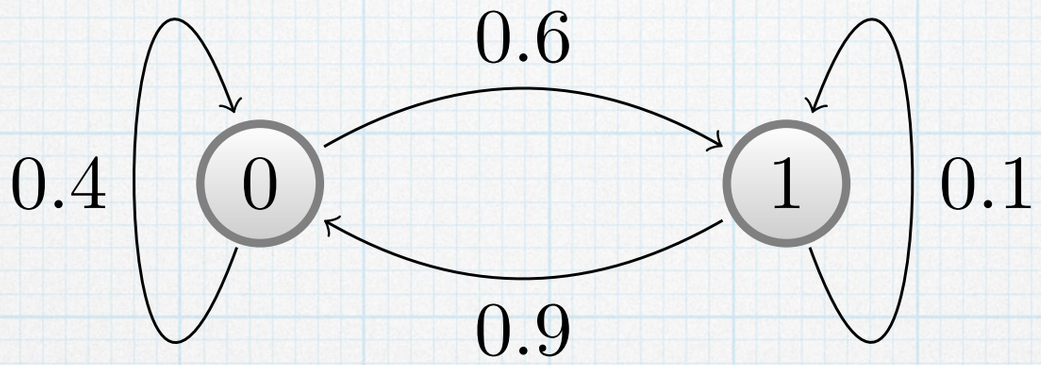


Imagine infinitely many chains. At equilibrium (steady-state), the "flux out" of each state must be equal to the "flux into" that state.

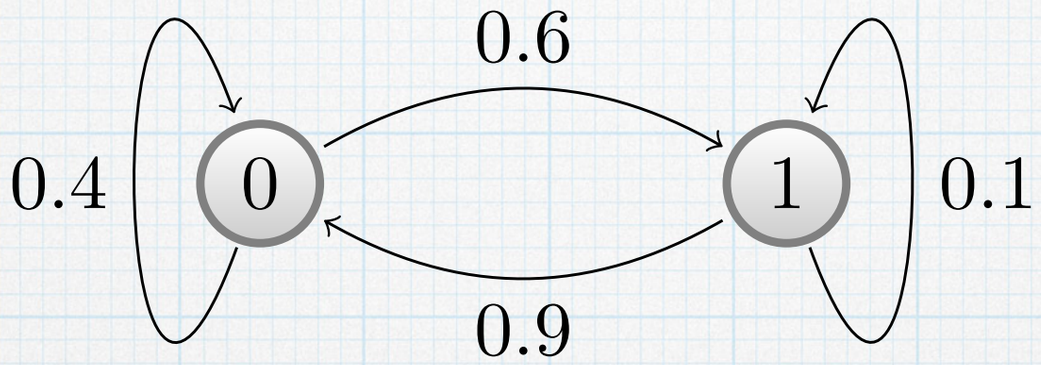
$$\begin{aligned} \boldsymbol{\pi}_0 \mathbb{P}(x_{i+1} = 1 | x_i = 0) &= \boldsymbol{\pi}_1 \mathbb{P}(x_{i+1} = 0 | x_i = 1) \\ \frac{\boldsymbol{\pi}_0}{\boldsymbol{\pi}_1} &= \frac{\mathbb{P}(x_{i+1} = 0 | x_i = 1)}{\mathbb{P}(x_{i+1} = 1 | x_i = 0)} \end{aligned}$$

$$\pi_0 = \mathbb{P}(x_i = 0) \qquad \pi_1 = \mathbb{P}(x_i = 1)$$

Stationary Distribution of a Markov Chain

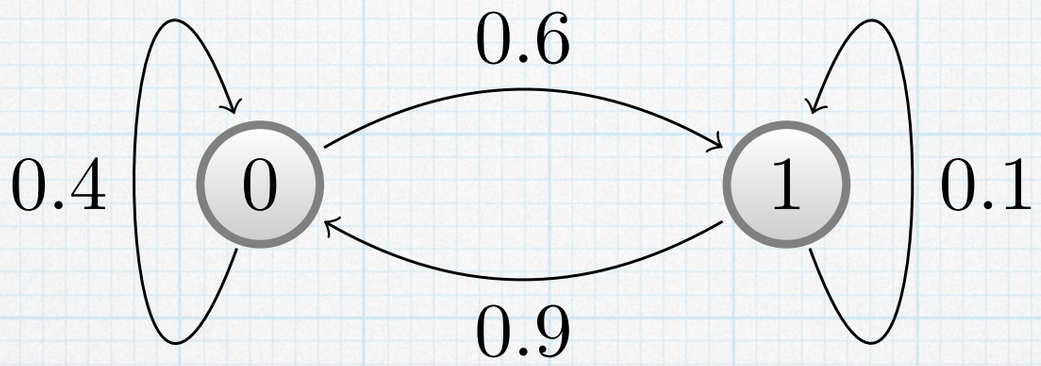


Stationary Distribution of a Markov Chain



$$\frac{\boldsymbol{\pi}_0}{\boldsymbol{\pi}_1} = \frac{\mathbb{P}(x_{i+1}=0|x_i=1)}{\mathbb{P}(x_{i+1}=1|x_i=0)}$$

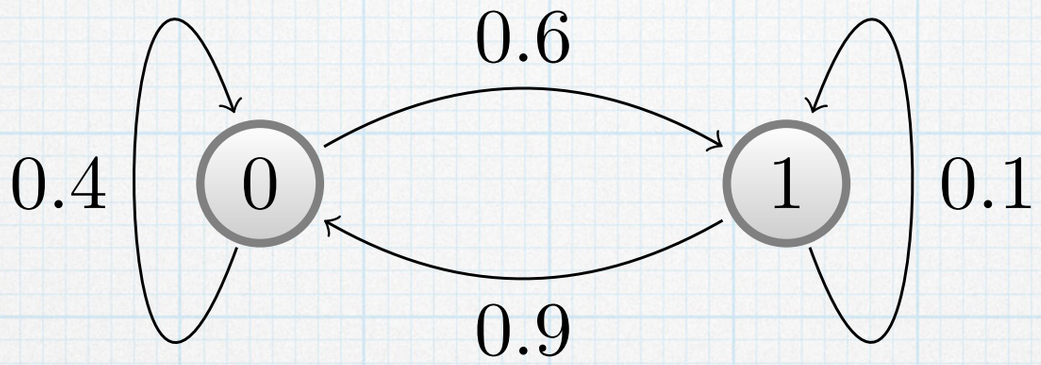
Stationary Distribution of a Markov Chain



$$\frac{\boldsymbol{\pi}_0}{\boldsymbol{\pi}_1} = \frac{\mathbb{P}(x_{i+1}=0|x_i=1)}{\mathbb{P}(x_{i+1}=1|x_i=0)}$$

$$\pi_0 + \pi_1 = 1$$

Stationary Distribution of a Markov Chain



$$\frac{\boldsymbol{\pi}_0}{\boldsymbol{\pi}_1} = \frac{\mathbb{P}(x_{i+1}=0|x_i=1)}{\mathbb{P}(x_{i+1}=1|x_i=0)}$$

$$\pi_0 + \pi_1 = 1$$

$$\begin{aligned}
\frac{\boldsymbol{\pi}_0}{\boldsymbol{\pi}_1} &= \frac{0.9}{0.6} = 1.5 \\
\boldsymbol{\pi}_0 &= 1.5\boldsymbol{\pi}_1 \\
1.5\boldsymbol{\pi}_1 + \boldsymbol{\pi}_1 &= 1.0 \\
\boldsymbol{\pi}_1 &= 0.4 \\
\boldsymbol{\pi}_0 &= 0.6
\end{aligned}$$

Stationary Distribution of a Markov Chain

If we can choose the transition probabilities of the Markov chain, then we can construct a sampler that will converge to any distribution that we desire!

Stationary Distribution of a Markov Chain

* For the general case of more than 2 states:

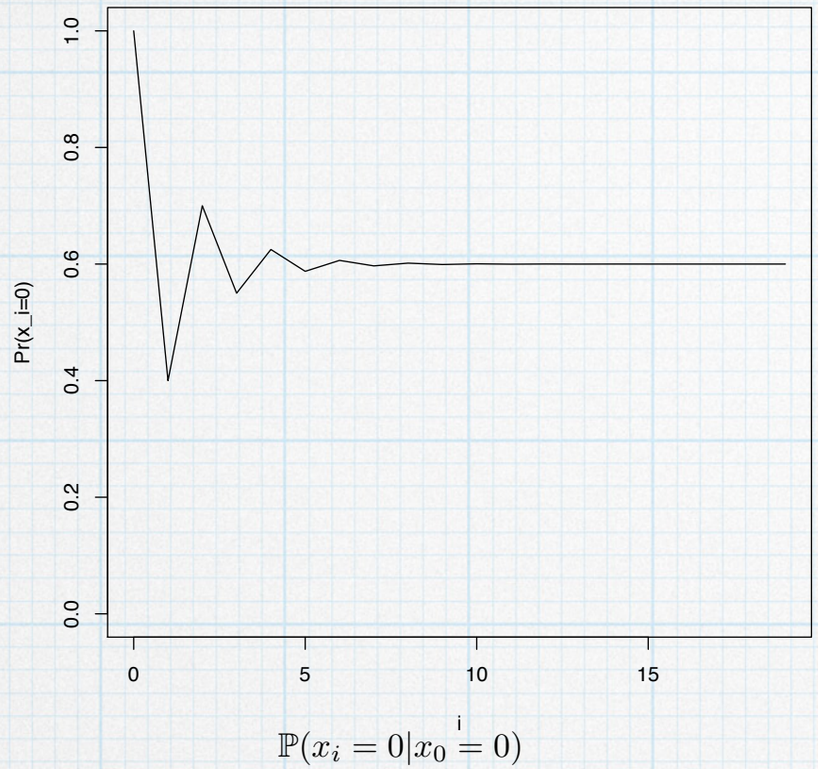
$$\begin{aligned}
\text{flux out of } j &= \boldsymbol{\pi}_j \mathbb{P}(x_{i+1} \in \mathcal{S}_{\neq j} | x_i = j) \\
&= \boldsymbol{\pi}_j \left[1 - \mathbb{P}(x_{i+1} \in j | x_i = j)\right]
\end{aligned}$$

$$\text{flux into } j = \sum_{k \in \mathcal{S}_{\neq j}} \pi_k \mathbb{P}(x_{i+1} = j | x_i = k)$$

$$\begin{aligned}
\boldsymbol{\pi}_j \left[1 - \mathbb{P}(x_{i+1} = j | x_i = j)\right] &= \sum_{k \in \mathcal{S}_{\neq j}} \boldsymbol{\pi}_k \mathbb{P}(x_{i+1} = j | x_i = k) \\
\boldsymbol{\pi}_j &= \boldsymbol{\pi}_j \mathbb{P}(x_{i+1} = j | x_i = j) + \sum_{k \in \mathcal{S}_{\neq j}} \boldsymbol{\pi}_k \mathbb{P}(x_{i+1} = j | x_i = k) \\
&= \sum_{k \in \mathcal{S}} \boldsymbol{\pi}_k \mathbb{P}(x_{i+1} = j | x_i = k)
\end{aligned}$$

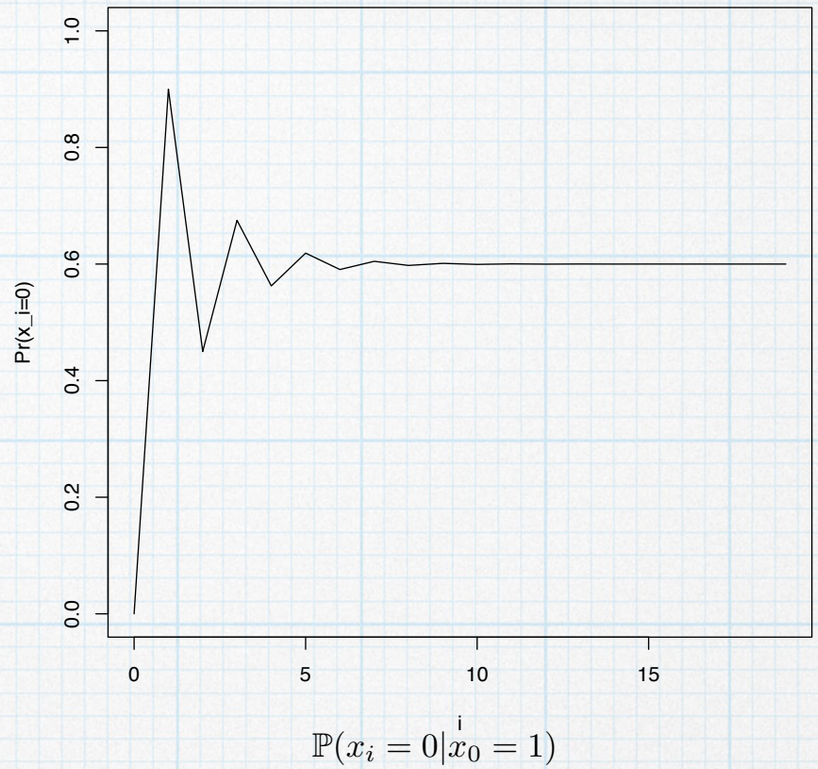
Mixing

* While setting the transition probabilities to specific values affects the stationary distribution, the transition probabilities cannot be determined uniquely from the stationary distribution.



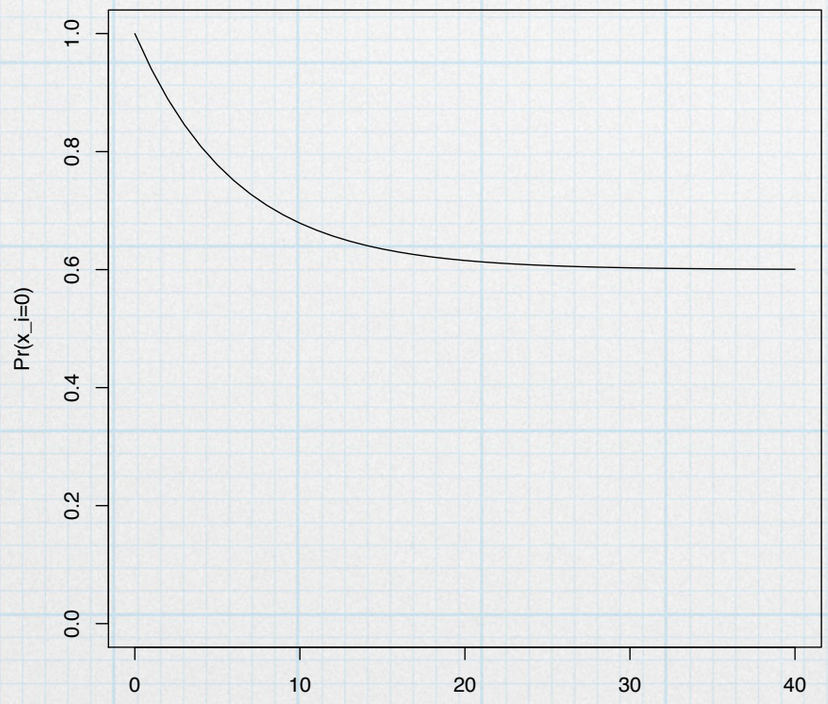
$\mathbb{P}(x_i = 0|x_0 = 0)$

$$\mathbb{P}(x_{i+1} = 1|x_i = 0) = 0.6$$



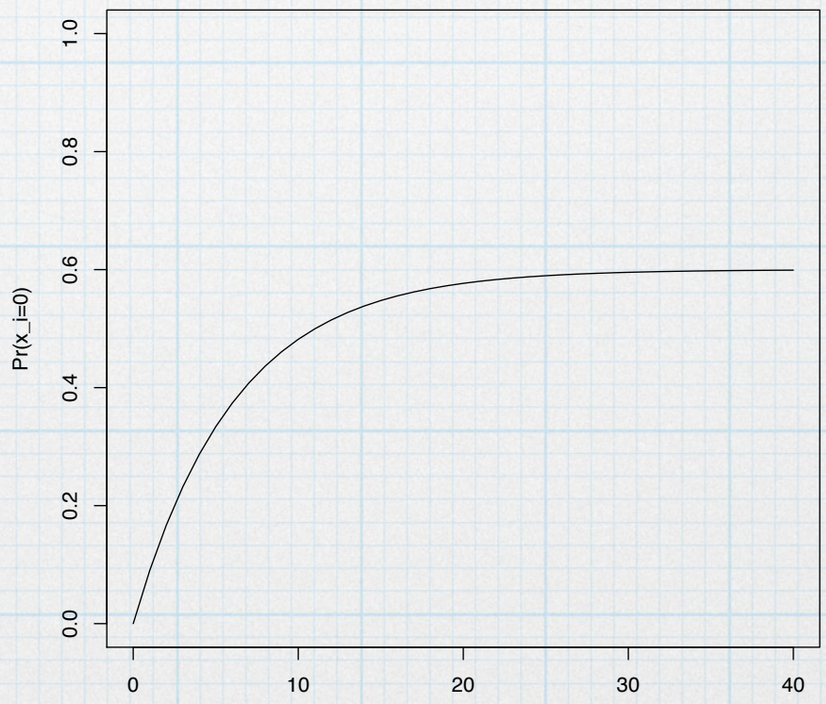
$\mathbb{P}(x_i = 0|x_0 = 1)$

$$\mathbb{P}(x_{i+1} = 0|x_i = 1) = 0.9$$



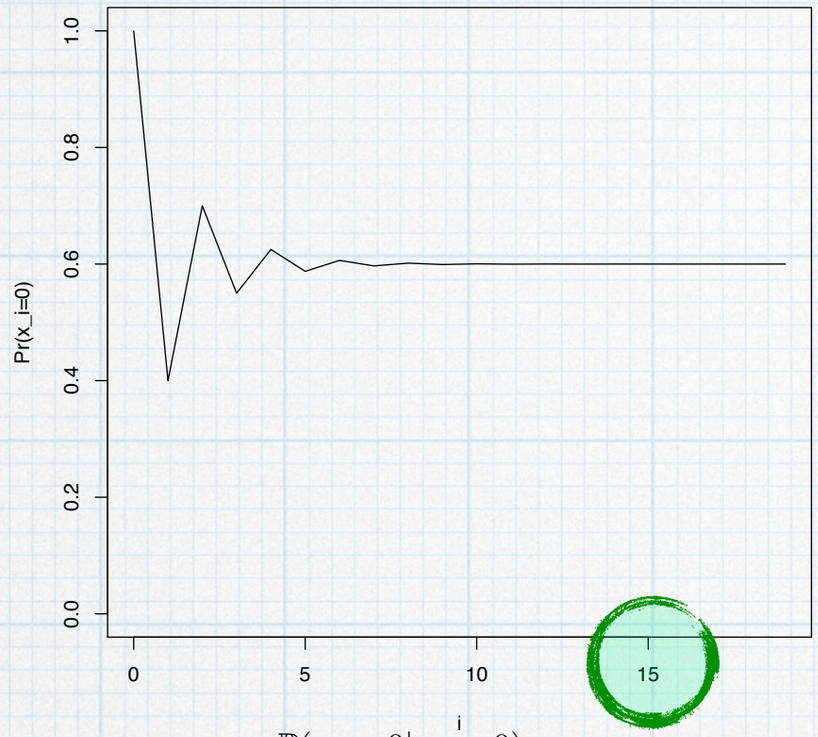
$\mathbb{P}(x_i = 0|x_0 = 0)$

$$\mathbb{P}(x_{i+1} = 1|x_i = 0) = 0.06$$

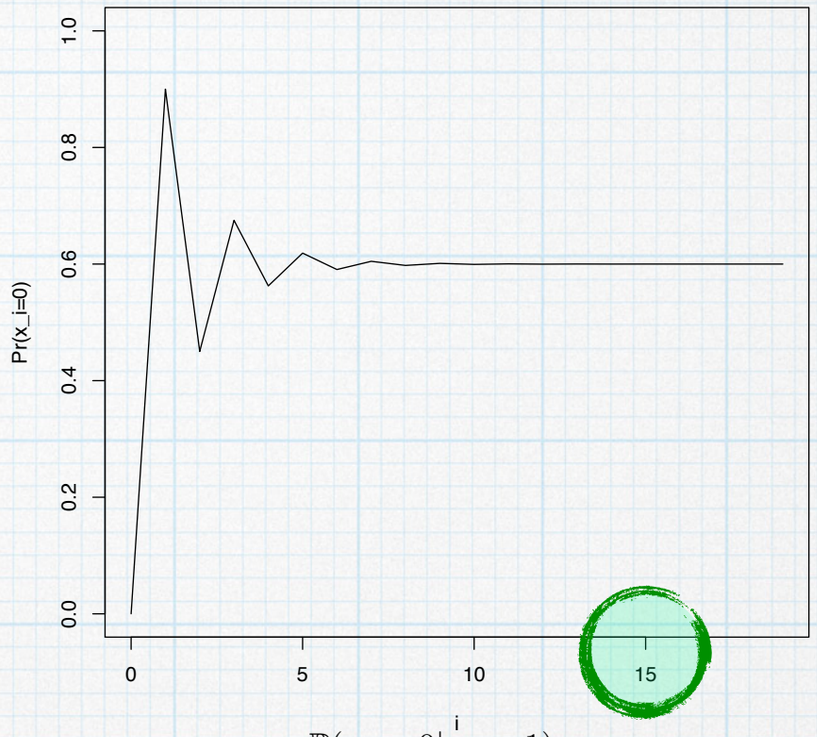


$\mathbb{P}(x_i = 0|x_0 = 1)$

$$\mathbb{P}(x_{i+1} = 0|x_i = 1) = 0.09$$

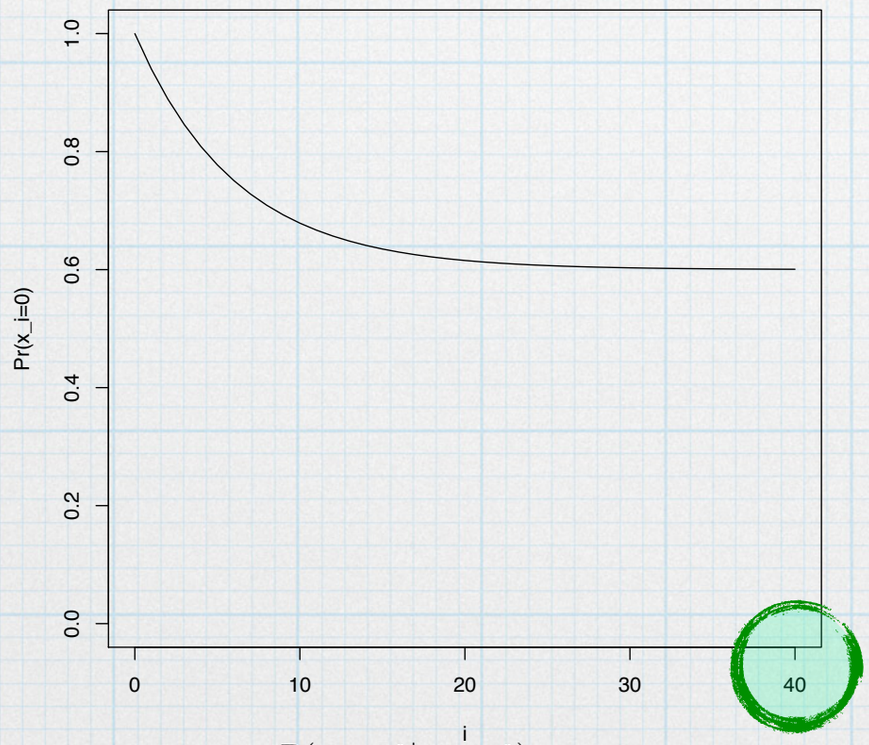


$\mathbb{P}(x_i = 0|x_0 = 0)$

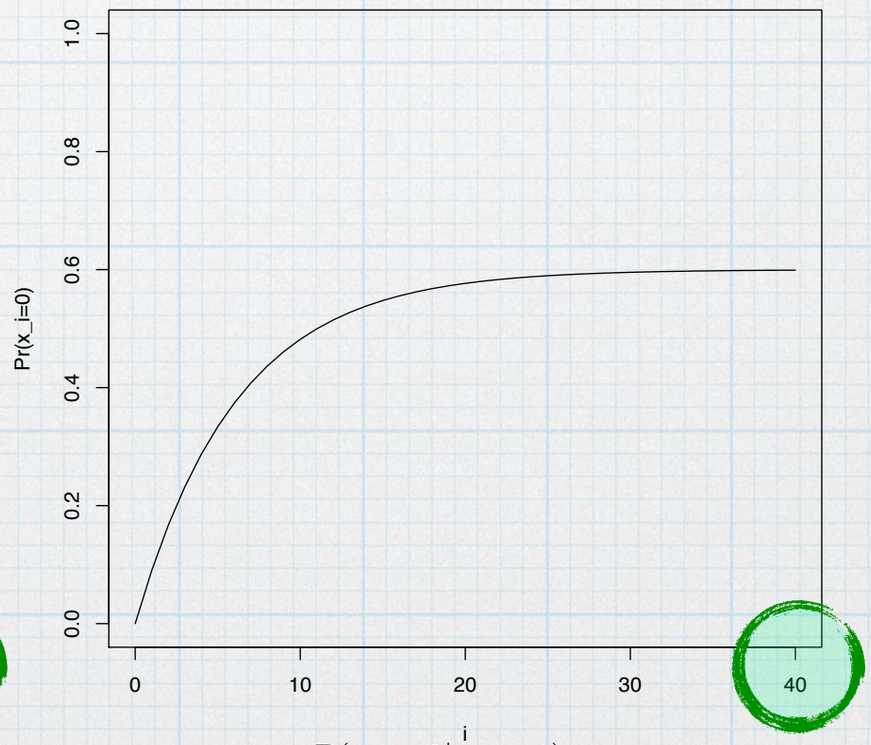


$\mathbb{P}(x_i = 0|x_0 = 1)$

$$\mathbb{P}(x_{i+1} = 1|x_i = 0) = 0.6 \quad \mathbb{P}(x_{i+1} = 0|x_i = 1) = 0.9$$



$\mathbb{P}(x_i = 0|x_0 = 0)$



$\mathbb{P}(x_i = 0|x_0 = 1)$

$$\mathbb{P}(x_{i+1} = 1|x_i = 0) = 0.06 \quad \mathbb{P}(x_{i+1} = 0|x_i = 1) = 0.09$$

Mixing

* Setting the transition probabilities to lower values resulted in a chain that "mixed" more slowly: Adjacent steps would be more likely to be in the same state and, thus, would require a larger number of iterations before the chain "forgets" its starting state.

Mixing

* The rate of convergence of a chain to its stationary distribution is an aspect of a Markov chain that is separate from what the stationary distribution is.

Mixing

* In MCMC, we will design a Markov chain whose stationary distribution is identical to the posterior probability distribution over the space of parameters.
* We try to design chains that have high transition probabilities to achieve faster convergence.

Detailed Balance

* In practice, the number of states is very large.
* Setting the transition probabilities so that we have equal flux into and out of any state is tricky.
* What we use instead is detailed balance.

Detailed Balance

* We restrict ourselves to Markov chains that satisfy detailed balance for all pairs of states j and k:

$$\pi_j \mathbb{P}(x_{i+1} = k | x_i = j) = \pi_k \mathbb{P}(x_{i+1} = j | x_i = k)$$

(equivalently: $\frac{\pi_j}{\pi_k} = \frac{\mathbb{P}(x_{i+1} = j | x_i = k)}{\mathbb{P}(x_{i+1} = k | x_i = j)}$)

* This can be achieved using several different methods, the most flexible of which is known as the **Metropolis algorithm** and its extension, the **Metropolis-Hastings method**.

* In the Metropolis-Hastings algorithm, we choose rules for constructing a random walk through the parameter space.

* We adopt transition probabilities such that the stationary distribution of our Markov chain is equal to the posterior probability distribution:

$$\pi_{\theta_j} = \mathbb{P}(\theta_j | \text{Data})$$

$$\frac{t_{\ell,k}}{t_{k,\ell}} = \frac{\pi_k}{\pi_\ell} = \left(\frac{\mathbb{P}(D|\theta_j = k)\mathbb{P}(\theta_j = k)}{\mathbb{P}(D)}\right) \Big/ \left(\frac{\mathbb{P}(D|\theta_j = \ell)\mathbb{P}(\theta_j = \ell)}{\mathbb{P}(D)}\right)$$

desired property

$$\frac{t_{\ell,k}}{t_{k,\ell}} = \frac{\pi_k}{\pi_\ell} = \left(\frac{\mathbb{P}(D|\theta_j = k)\mathbb{P}(\theta_j = k)}{\mathbb{P}(D)}\right) \Big/ \left(\frac{\mathbb{P}(D|\theta_j = \ell)\mathbb{P}(\theta_j = \ell)}{\mathbb{P}(D)}\right)$$

desired property

$$\frac{t_{\ell,k}}{t_{k,\ell}} = \frac{\pi_k}{\pi_\ell} = \left(\frac{\mathbb{P}(D|\theta_j = k)\mathbb{P}(\theta_j = k)}{\mathbb{P}(D)}\right) \Big/ \left(\frac{\mathbb{P}(D|\theta_j = \ell)\mathbb{P}(\theta_j = \ell)}{\mathbb{P}(D)}\right)$$

detailed balance

desired property

$$\frac{t_{\ell,k}}{t_{k,\ell}} = \frac{\pi_k}{\pi_\ell} = \left(\frac{\mathbb{P}(D|\theta_j = k)\mathbb{P}(\theta_j = k)}{\mathbb{P}(D)}\right) \Big/ \left(\frac{\mathbb{P}(D|\theta_j = \ell)\mathbb{P}(\theta_j = \ell)}{\mathbb{P}(D)}\right)$$

detailed balance

P(D) cancels out, so doesn't need to be computed!

* Therefore, we need to set the transition probabilities so that

$$\frac{t_{\ell,k}}{t_{k,\ell}} = \frac{\mathbb{P}(D|\theta_j = k)\mathbb{P}(\theta_j = k)}{\mathbb{P}(D|\theta_j = \ell)\mathbb{P}(\theta_j = \ell)}$$

* However, an important problem arises when doing this, which can be illustrated as follows:

  * when dealing with states k,l, it could be that we need to set $t_{k,l}=1$ and $t_{l,k}=0.5$

  * when dealing with states k,m, it could be that we need to set $t_{k,m}=0.3$ and $t_{m,k}=0.1$

  * Then, we have $t_{k,m}+t_{k,l}=1.3$, which violates the fundamental rules of probability!

* Solution:
    * view the transition probability as a joint event: (1) the move is proposed with probability q, and (2) the move is accepted with probability α.
    * If we denote by x'_{i+1} the state proposed at step i+1, then

$$
\begin{aligned}
q(j,k) &= \mathbb{P}(x'_{i+1} = k | x_i = j) \\
\alpha(j,k) &= \mathbb{P}(x_{i+1} = k | x'_i = j, x'_{i+1} = k)
\end{aligned}
$$

* We can choose proposal probabilities that sum to one for all the state-changing transitions.

* Then, we can multiply them by the appropriate acceptance probabilities (keeping them as high as possible, but ≤ 1).

* We get

$$\frac{t_{\ell,k}}{t_{k,\ell}} = \frac{q(\ell,k)\alpha(\ell,k)}{q(k,\ell)\alpha(k,\ell)} = \frac{\mathbb{P}(D|\theta_j = k)\mathbb{P}(\theta_j = k)}{\mathbb{P}(D|\theta_j = \ell)\mathbb{P}(\theta_j = \ell)}$$

$$\frac{t_{\ell,k}}{t_{k,\ell}} = \frac{q(\ell,k)\alpha(\ell,k)}{q(k,\ell)\alpha(k,\ell)} = \frac{\mathbb{P}(D|\theta_j = k)\mathbb{P}(\theta_j = k)}{\mathbb{P}(D|\theta_j = \ell)\mathbb{P}(\theta_j = \ell)}$$

* We have flexibility in selecting how we perform proposals on new states in MCMC.

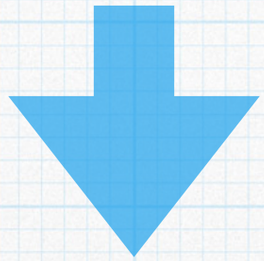
* We have to ensure that q(l,k)>0 whenever q(k,l)>0 (it is fine if both are 0, but we can't have one being 0 and the other greater than 0).

$$\frac{t_{\ell,k}}{t_{k,\ell}} = \frac{q(\ell,k)\alpha(\ell,k)}{q(k,\ell)\alpha(k,\ell)} = \frac{\mathbb{P}(D|\theta_j = k)\mathbb{P}(\theta_j = k)}{\mathbb{P}(D|\theta_j = \ell)\mathbb{P}(\theta_j = \ell)}$$

* However, once we have chosen a proposal scheme, we do not have much flexibility in choosing whether or not to accept a proposal.

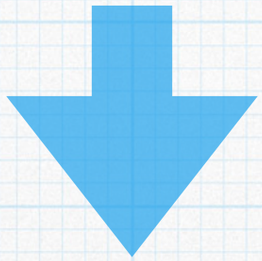
$$\frac{t_{\ell,k}}{t_{k,\ell}} = \frac{q(\ell,k)\alpha(\ell,k)}{q(k,\ell)\alpha(k,\ell)} = \frac{\mathbb{P}(D|\theta_j = k)\mathbb{P}(\theta_j = k)}{\mathbb{P}(D|\theta_j = \ell)\mathbb{P}(\theta_j = \ell)}$$

$$\frac{t_{\ell,k}}{t_{k,\ell}} = \frac{q(\ell,k)\alpha(\ell,k)}{q(k,\ell)\alpha(k,\ell)} = \frac{\mathbb{P}(D|\theta_j = k)\mathbb{P}(\theta_j = k)}{\mathbb{P}(D|\theta_j = \ell)\mathbb{P}(\theta_j = \ell)}$$

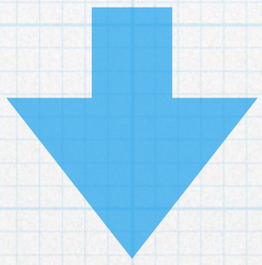


$$\frac{\alpha(\ell,k)}{\alpha(k,\ell)} = \frac{\mathbb{P}(D|\theta_j = k)\mathbb{P}(\theta_j = k)q(k,\ell)}{\mathbb{P}(D|\theta_j = \ell)\mathbb{P}(\theta_j = \ell)q(\ell,k)}$$

$$\frac{t_{\ell,k}}{t_{k,\ell}} = \frac{q(\ell,k)\alpha(\ell,k)}{q(k,\ell)\alpha(k,\ell)} = \frac{\mathbb{P}(D|\theta_j = k)\mathbb{P}(\theta_j = k)}{\mathbb{P}(D|\theta_j = \ell)\mathbb{P}(\theta_j = \ell)}$$

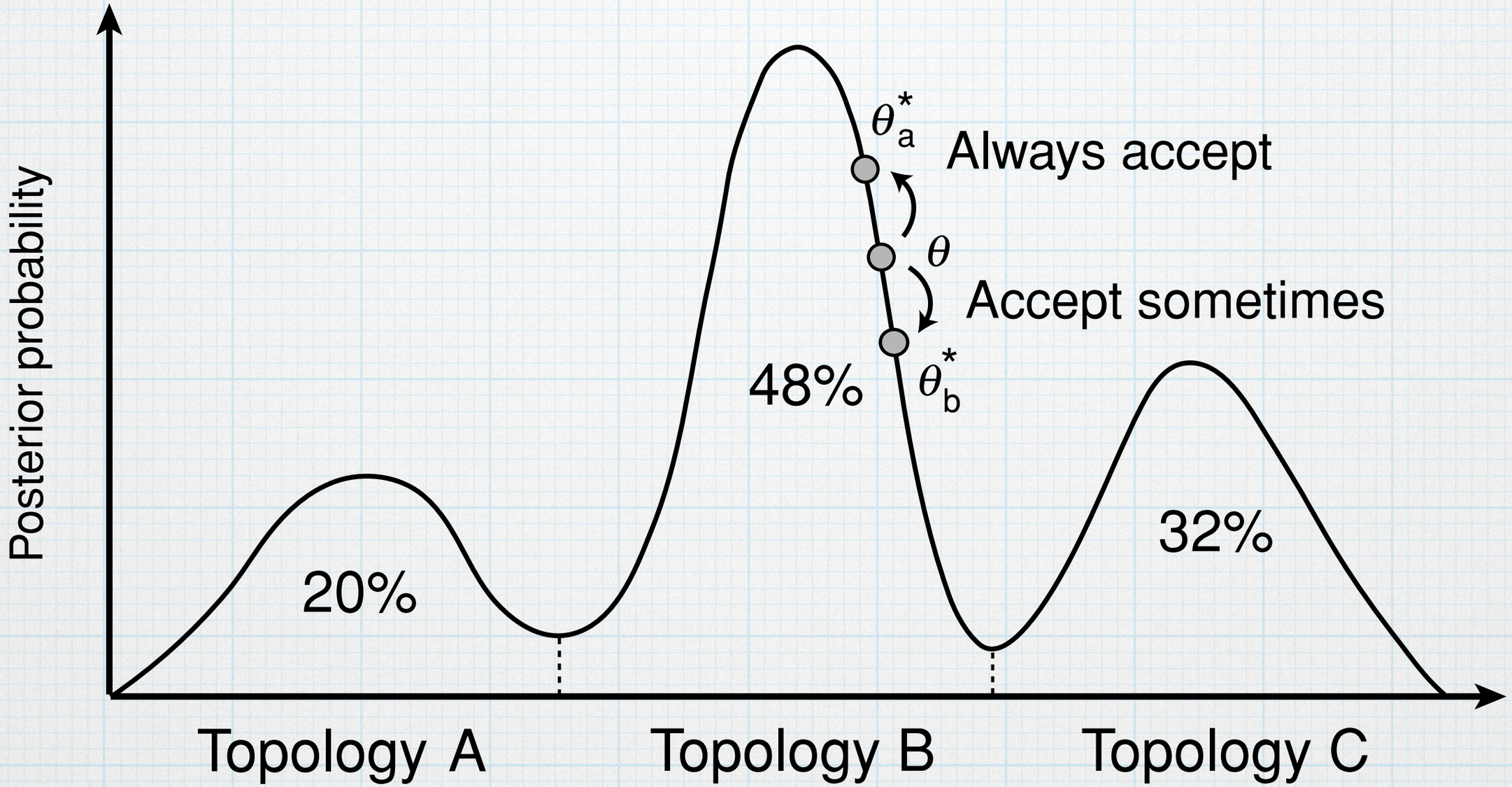


$$\frac{\alpha(\ell,k)}{\alpha(k,\ell)} = \frac{\mathbb{P}(D|\theta_j = k)\mathbb{P}(\theta_j = k)q(k,\ell)}{\mathbb{P}(D|\theta_j = \ell)\mathbb{P}(\theta_j = \ell)q(\ell,k)}$$



acceptance ratio = (likelihood ratio)(prior ratio)(Hastings ratio)

* The central idea is to make small random changes to some current parameter values, and then accept or reject the changes according to the appropriate probabilities


Posterior probability
θ^*_a
Always accept
θ
Accept sometimes
48%
θ^*_b
20%
32%
Topology A
Topology B
Topology C

Markov chain Monte Carlo steps

1. Start at an arbitrary point (θ)
2. Make a small random move (to θ^*)
3. Calculate height ratio (r) of new state (to θ^*) to old state (θ)
   - (a) $r > 1$: new state accepted
   - (b) $r < 1$: new state accepted with probability r
     if new state rejected, stay in old state
4. Go to step 2

$$r = \min\left(1, \frac{f(\theta^*|X)}{f(\theta|X)} \times \frac{f(\theta|\theta^*)}{f(\theta^*|\theta)}\right)$$

$$= \min\left(1, \frac{f(\theta^*)\, f(X|\theta^*)/f(X)}{f(\theta)\, f(X|\theta)/f(X)} \times \frac{f(\theta|\theta^*)}{f(\theta^*|\theta)}\right)$$

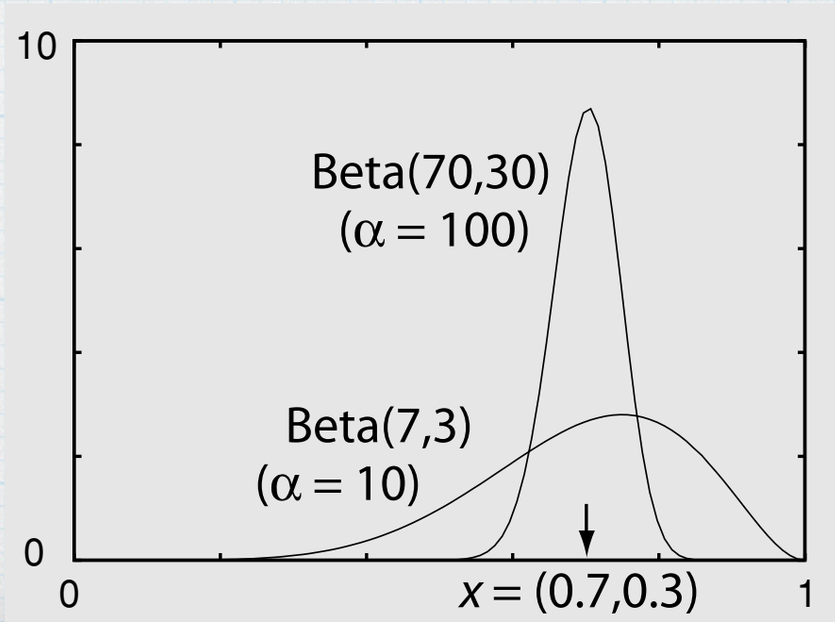
$$= \min\left(1, \frac{f(\theta^*)}{f(\theta)} \times \frac{f(X|\theta^*)}{f(X|\theta)} \times \frac{f(\theta|\theta^*)}{f(\theta^*|\theta)}\right)$$

prior ratio

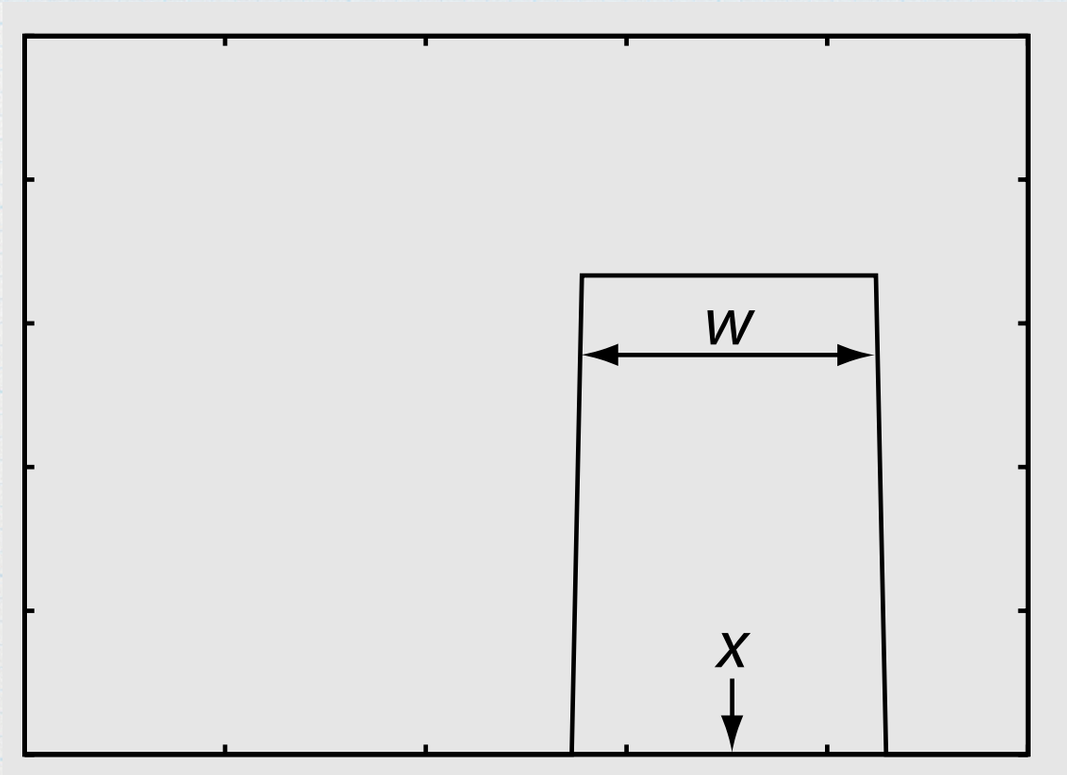
likelihood ratio

proposal ratio

* An example of a proposal mechanism is the **beta proposal**:
    * Assume the current values are (x1,x2);
    * Multiply them with a value α;
    * Pick new values from Beta(αx1,αx2)



* A simpler proposal mechanism is to define a continuous uniform distribution of width w, centered on the current value x, and the new value x* is drawn from this distribution.



* More complex moves are needed to change tree topology.

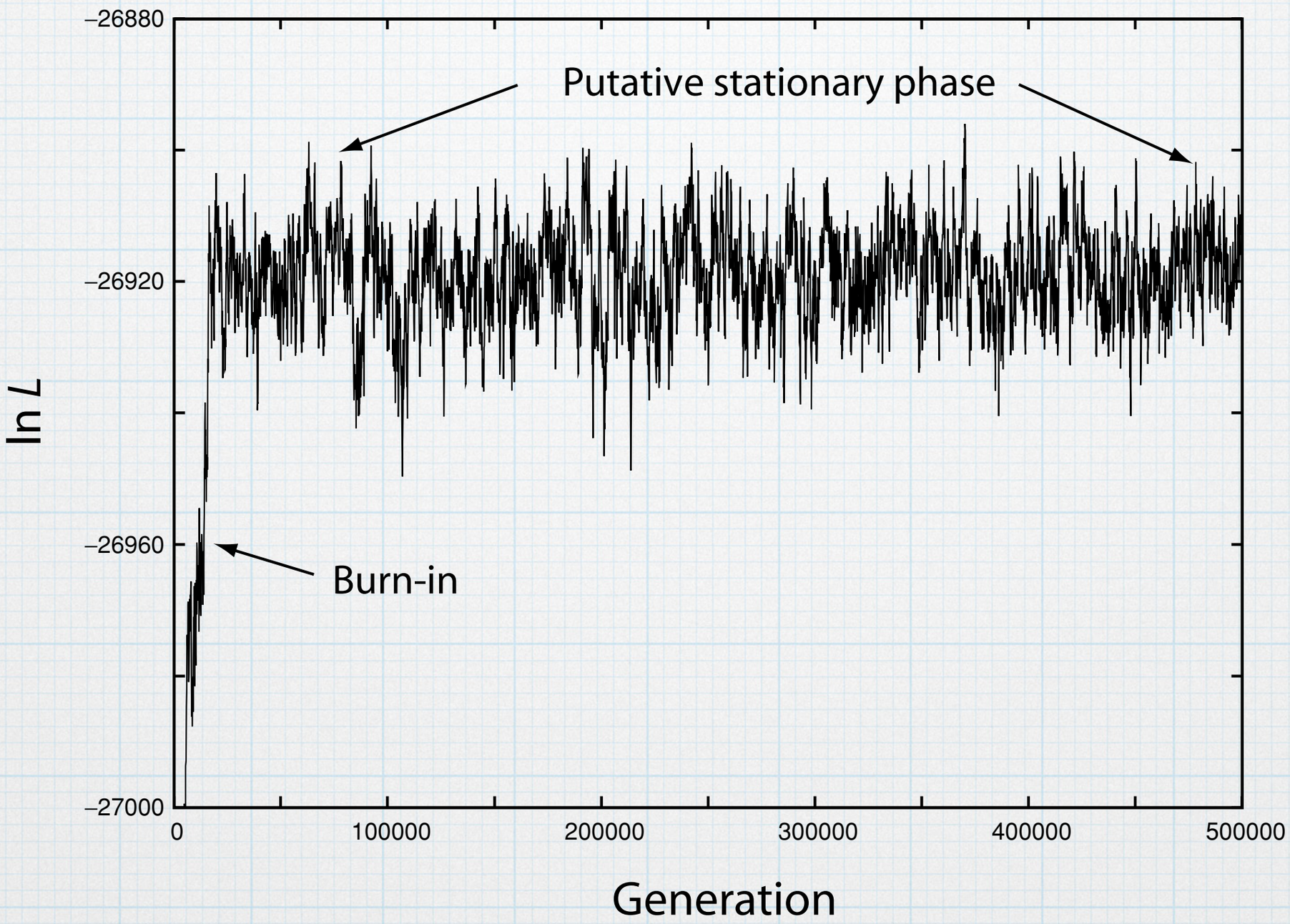
* A common type uses operations such as SPR, TBR, and NNI.

Burn-in, mixing, and convergence

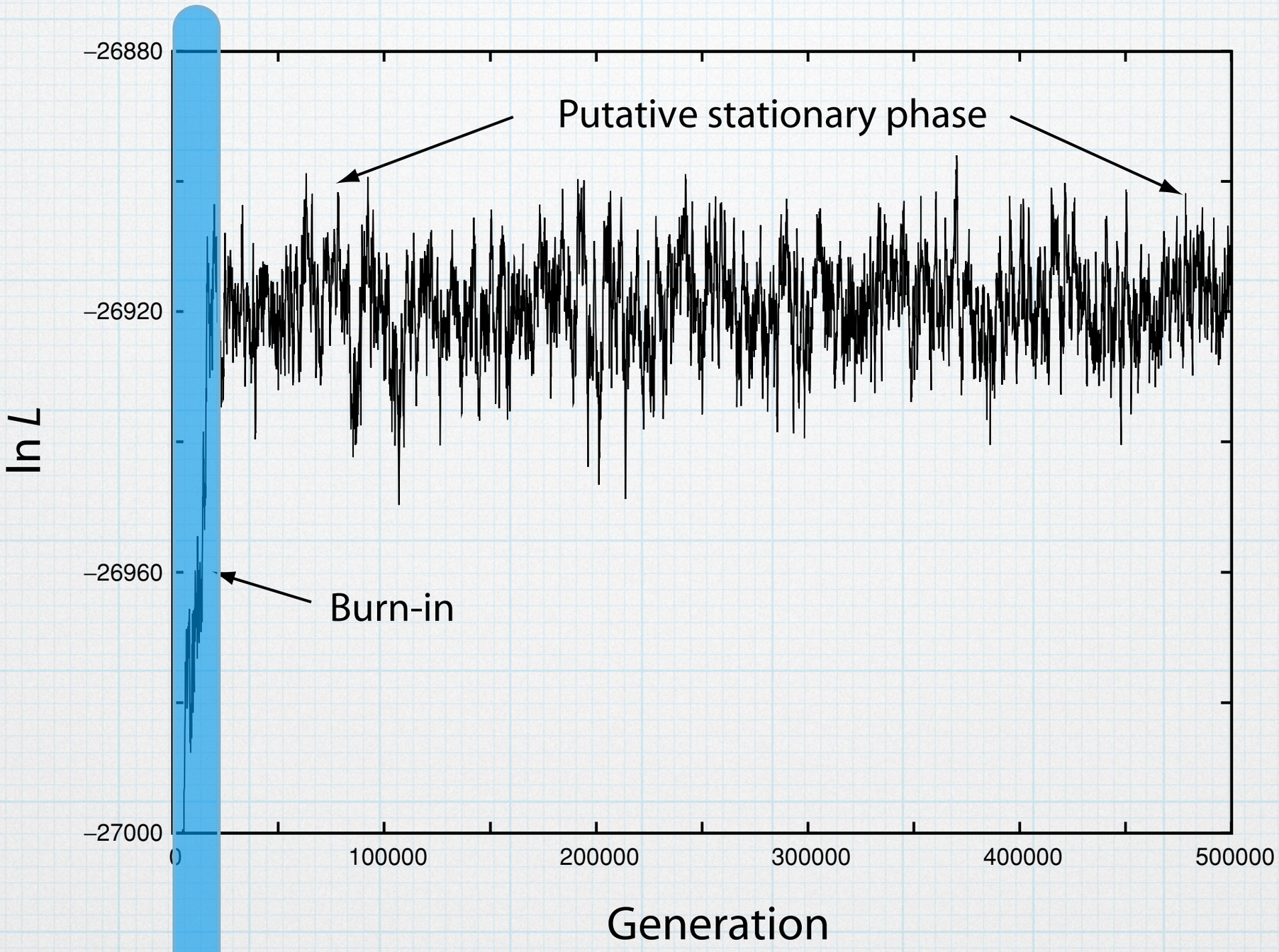
* If the chain is started from a random tree and arbitrarily chosen branch lengths, chances are that the initial likelihood is low.

* The early phase of the run in which the likelihood increases very rapidly towards regions in the posterior with high probability mass is known as the **burn-in**.

Trace plot


Putative stationary phase
Burn-in
ln L
−26880
−26920
−26960
−27000
0
100000
200000
300000
400000
500000
Generation

Trace plot



samples in this region are discarded!

* As the chain approaches its stationary distribution, the likelihood values tend to reach a plateau.

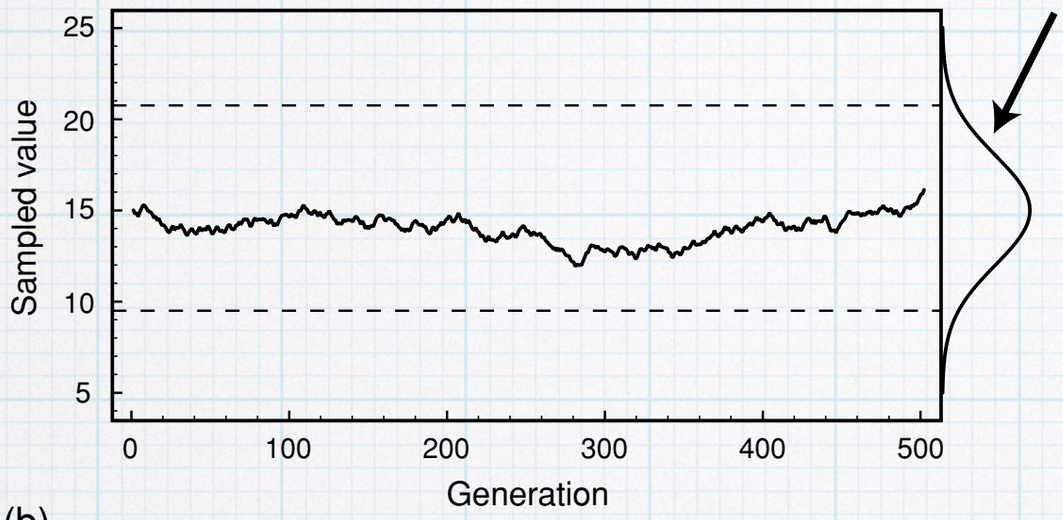
* This is the first sign that the chain may have converged onto the target distribution.

* However, it is not sufficient for the chain to reach the region of high probability in the posterior; it must also cover this region adequately.

* The speed with which the chain covers the interesting regions of the posterior is known as its **mixing behavior**.

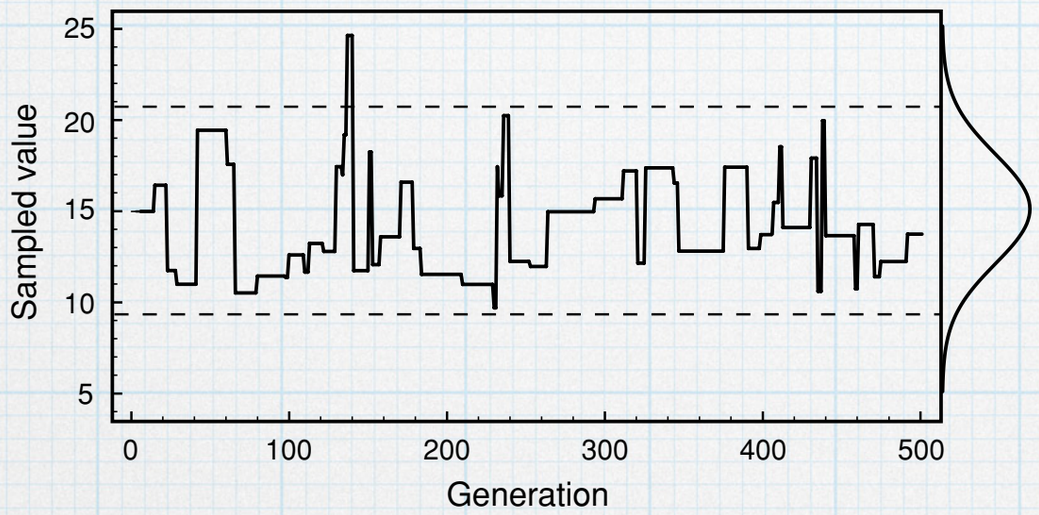
* The better the mixing, the faster the chain will generate an adequate sample of the posterior.

(a)



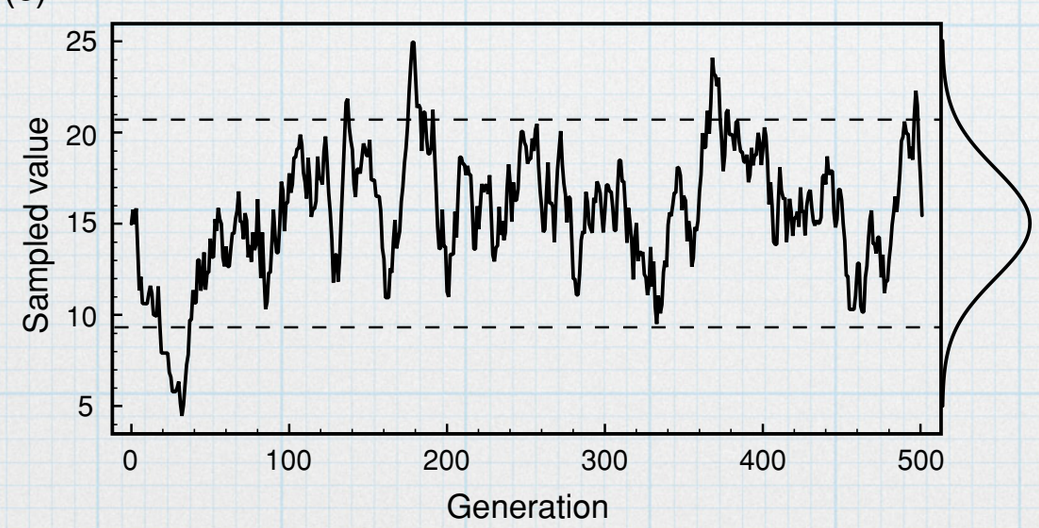
Too modest proposals
Acceptance rate too high
Poor mixing

(b)



Too bold proposals
Acceptance rate too low
Poor mixing

(c)



Moderately bold proposals
Acceptance rate intermediate
Good mixing

* In Bayesian MCMC sampling of phylogenetic problems, the tree topology is typically the most difficult parameter to sample from.

* Therefore, it makes sense to focus on this parameter when monitoring convergence.

Summarizing the results

* The stationary phase of the chain is typically sampled with some thinning, for instance every 50th or 100th generation.

* Once an adequate sample is obtained, it is usually trivial to compute an estimate of the marginal posterior distribution for the parameter(s) of interest.

* For example, this can take the form of a frequency histogram of the sampled values.
* When it is difficult to visualize this distribution or when space does not permit it, various summary statistics are used instead.

* The most common approach to summarizing topology posteriors is to give the frequencies of the most common splits, since there are much fewer splits than topologies.

Summary

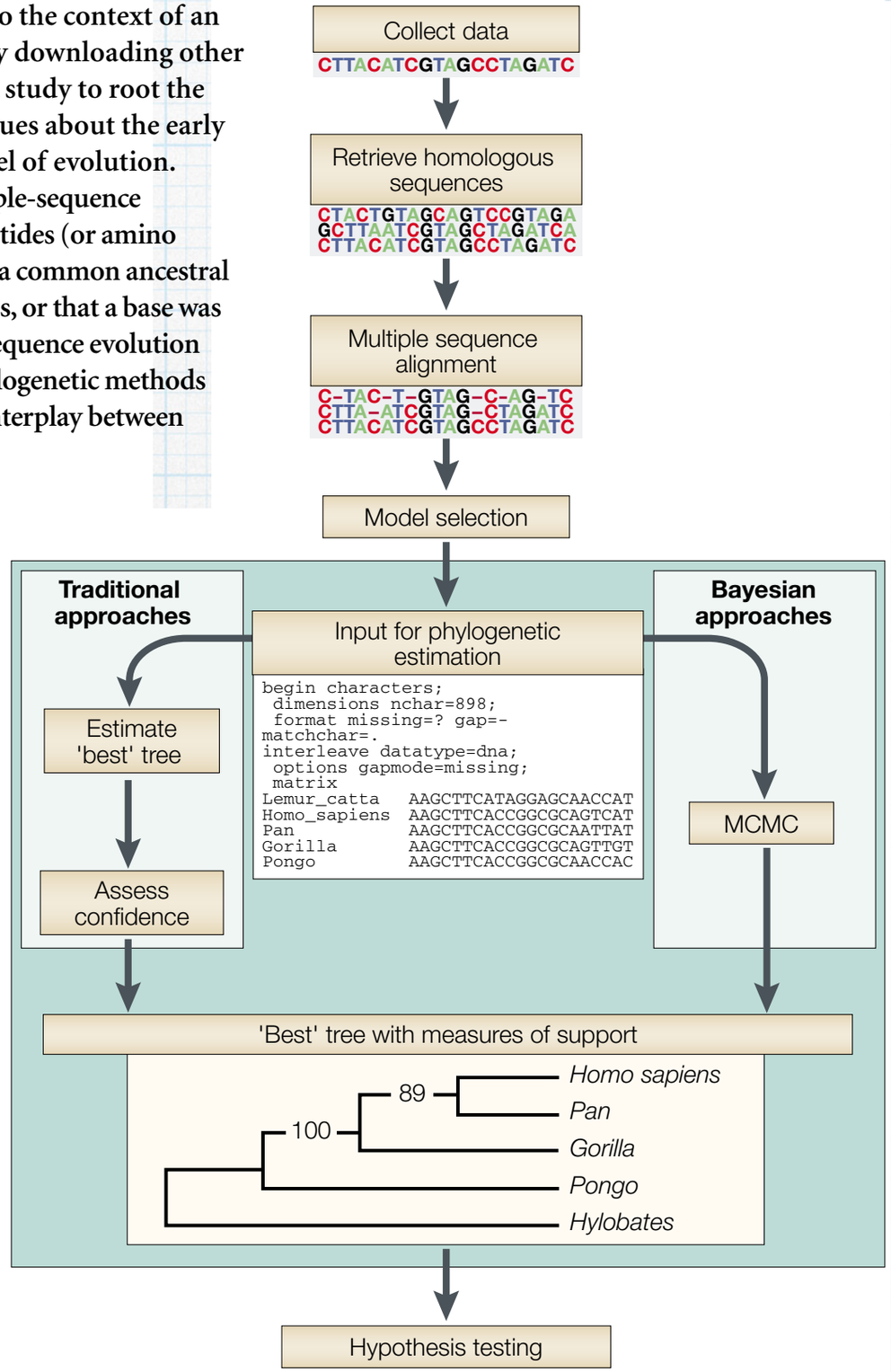
Box 2 | **The phylogenetic inference process**

The flowchart puts phylogenetic estimation (shown in the green box) into the context of an entire study. After new sequence data are collected, the first step is usually downloading other relevant sequences. Typically, a few outgroup sequences are included in a study to root the tree (that is, to indicate which nodes in the tree are the oldest), provide clues about the early ancestral sequences and improve the estimates of parameters in the model of evolution.

Insertions and deletions obscure which of the sites are homologous. Multiple-sequence alignment is the process of adding gaps to a matrix of data so that the nucleotides (or amino acids) in one column of the matrix are related to each other by descent from a common ancestral residue (a gap in a sequence indicates that the site has been lost in that species, or that a base was inserted at that position in some of the other species). Although models of sequence evolution that incorporate insertions and deletions have been proposed[55–58], most phylogenetic methods proceed using an aligned matrix as the input (see REF. 59 for a review of the interplay between alignment and tree inference).

In addition to the data, the scientist must choose a model of sequence evolution (even if this means just choosing a family of models and letting software infer the parameters of these models). Increasing model complexity improves the fit to the data but also increases variance in estimated parameters. Model selection[60–63] strategies attempt to find the appropriate level of complexity on the basis of the available data. Model complexity can often lead to computational intractability, so pragmatic concerns sometimes outweigh statistical ones (for example, NJ and parsimony are mainly justifiable by their speed).

As discussed in BOX 3, data and a model can be used to create a sample of trees through either Markov chain Monte Carlo (MCMC) or multiple tree searches on bootstrapped data (the 'traditional' approach). This collection of trees is often summarized using consensus-tree techniques, which show the parts of the tree that are found in most, or all, of the trees in a set. Although useful, CONSENSUS METHODS are just one way of summarizing the information in a group of trees. AGREEMENT SUBTREES are more resistant to 'rogue sequences' (one or a few sequences that are difficult to place on the tree); the presence of such sequences can make a consensus tree relatively unresolved, even when there is considerable agreement on the relationships between the other sequences. Sometimes, the bootstrap or MCMC sample might show substantial support for multiple trees that are not topologically similar. In such cases, presenting more than one tree (or more than one consensus of trees) might be the only way to appropriately summarize the data.



Source: Nat Rev Genet, 4:275, 2003

Summary

Table 1 | **Comparison of methods**

Method	Advantages	Disadvantages	Software
Neighbour joining	Fast	Information is lost in compressing sequences into distances; reliable estimates of pairwise distances can be hard to obtain for divergent sequences	PAUP* MEGA PHYLIP
Parsimony	Fast enough for the analysis of hundreds of sequences; robust if branches are short (closely related sequences or dense sampling)	Can perform poorly if there is substantial variation in branch lengths	PAUP* NONA MEGA PHYLIP
Minimum evolution	Uses models to correct for unseen changes	Distance corrections can break down when distances are large	PAUP* MEGA PHYLIP
Maximum likelihood	The likelihood fully captures what the data tell us about the phylogeny under a given model	Can be prohibitively slow (depending on the thoroughness of the search and access to computational resources)	PAUP* PAML PHYLIP
Bayesian	Has a strong connection to the maximum likelihood method; might be a faster way to assess support for treesthan maximum likelihood bootstrapping	The prior distributions for parameters must be specified; it can be difficult to determine whether the Markov chain Monte Carlo (MCMC) approximation has run for long enough	MrBayes BAMBE

Source: Nat Rev Genet, 4:275, 2003

Acknowledgment

* Material in these slides are based on Chapter 7 in "The Phylogenetic Handbook", Lemey, Salemi, Vandamme (Eds.)
* Some of the material is based on MCMC notes by Prof. Mark Holder

Questions?