Bayes Rule

\[ P(X = x | Y = y) = \frac{P(X = x, Y = y)}{P(Y = y)} = \frac{\sum_{x} P(X = x | Y = y) = \phi(x) = y}{\sum_{x} P(X = x | Y = y)} \]

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: $p(y=1)=0.004$.
Further, we need to take into account the fact that the test may be a false positive. Mammograms have a false positive probability of \( p(x=1|y=0) = 0.1 \).

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

\[
\begin{align*}
p(y = 1|x = 1) &= \frac{p(x=1|y=1)p(y=1)}{p(x=1|y=0)p(y=0) + p(x=1|y=1)p(y=1)} \\
&= \frac{0.031 \times 0.004}{0.031 \times 0.004 + 0.1 \times 0.996} \\
&= 0.031
\end{align*}
\]

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.

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.
7.2 Bayesian phylogenetic inference

How does Bayesian reasoning apply to phylogenetic inference? Assume we are interested in the relationships between man, gorilla, and chimpanzee. In the standard case, we need an additional species to root the tree, and the orangutan would be appropriate here. There are three possible ways of arranging these species in a phylogenetic tree: the chimpanzee is our closest relative, the gorilla is our closest relative, or the chimpanzee and the gorilla are each other's closest relatives (Fig. 7.1).

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).

To describe the analysis mathematically, consider:

- the matrix of aligned sequences $X$
- the tree topology parameter $\tau$
- the branch lengths of the tree $\nu$
- (typically, substitution model parameters are also included)

Let $\theta(\tau, \nu)$
Bayes theorem allows us to derive the posterior distribution as

\[ f(\theta | X) = \frac{f(\theta) f(X | \theta)}{f(X)} \]

where

\[ f(X) = \int f(\theta) f(X | \theta) \, d\theta \]

\[ = \sum \int f(\nu) f(X | \tau, \nu) \, d\nu \]

Why are they called marginal probabilities?

<table>
<thead>
<tr>
<th>Topologies</th>
<th>( \tau_A )</th>
<th>( \tau_B )</th>
<th>( \tau_C )</th>
<th>Joint probabilities</th>
</tr>
</thead>
<tbody>
<tr>
<td>( V^A )</td>
<td>0.10</td>
<td>0.07</td>
<td>0.12</td>
<td>0.29</td>
</tr>
<tr>
<td>( V^B )</td>
<td>0.05</td>
<td>0.22</td>
<td>0.06</td>
<td>0.33</td>
</tr>
<tr>
<td>( V^C )</td>
<td>0.05</td>
<td>0.19</td>
<td>0.14</td>
<td>0.38</td>
</tr>
<tr>
<td>Marginal probabilities</td>
<td>0.20</td>
<td>0.48</td>
<td>0.32</td>
<td></td>
</tr>
</tbody>
</table>

The marginal probability distribution on topologies.
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

![Diagram of a two-state Markov chain]

\[
P(x_{i+1} = 0 | x_i = 0) = 0.4 \\
P(x_{i+1} = 1 | x_i = 0) = 0.6 \\
P(x_{i+1} = 0 | x_i = 1) = 0.9 \\
P(x_{i+1} = 1 | x_i = 1) = 0.1
\]

### Stationary Distribution of a Markov Chain

![Diagram of a two-state Markov chain]

\[
P(x_{i+1} = 0 | x_i = 0) = 0.4 \\
P(x_{i+1} = 1 | x_i = 0) = 0.6 \\
P(x_{i+1} = 0 | x_i = 1) = 0.9 \\
P(x_{i+1} = 1 | x_i = 1) = 0.1
\]

What are \( P(x_0 = 0 | x_0 = 0) \), \( P(x_0 = 0 | x_0 = 1) \), \( P(x_0 = 1 | x_0 = 0) \), \( P(x_0 = 1 | x_0 = 1) \), \( P(x_0 = 0 | x_0 = 0) \), \( P(x_0 = 0 | x_0 = 1) \), \( P(x_0 = 1 | x_0 = 0) \), \( P(x_0 = 1 | x_0 = 1) \)?
Here we are exploiting the fact that the same "rules" (transition probabilities) apply when we consider the state at the previous step is enough. This is the Markov property. More formally we could state it as:

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

Consider the simplest possible Markov chain: one with two states

$$0 \rightarrow 1 \rightarrow 0$$

The full probability statements that correspond to the graph are:

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

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

Transition probabilities

If we consider an even larger number of iterations (e.g. the state at step 100), then the probabilities stabilize to a steady state distribution. Knowing whether the chain started in state 0 or 1 tells you very little about the state of the chain in step 15. Technically, we say the chain has reached the stationary distribution. Note that the probability stabilizes to a steady state distribution. Knowing whether the chain started in state 0 or 1 tells you very little about the state of the chain in step 15. Technically, we say the chain has reached the stationary distribution. Note that the probability stabilizes to a steady state distribution.
The running of the Markov chain, then we are dealing with a time-homogeneous Markov chain. Here we are exploiting the fact that the same "rules" (transition probabilities) apply when we know the state at the previous step is enough. Chains we don't need to concern ourselves with the full history of the chain, merely knowing the

Consider the simplest possible Markov chain: one with two states $0$ and $1$. Transition probabilities are analogous to transition rates if we are working with continuous-time Markov processes.

Transition probabilities are $P(x_{i+1} | x_i)$ for every possible $i$. These probabilistic statements are $P(x_{i+1} | x_i) = 0$ if $i = 0$ or $i = 1$. So when working with Markov processes, $P(x_{i+1} | x_i)$ is positive integer. What this probability statement is saying is that, conditional on $x_i$, the probability of $x_{i+1}$ only depends on $x_i$ and the transition probabilities, then we can make a probabilistic statement $P(x_{i+1} | x_i) = P(x_{i+1} | x_{i-1})$ for every possible $i$.

There are second-order Markov processes that depend on the two previous states, and third-order Markov processes. These second-order and third-order Markov processes do not have the Markov property.

Figure 1: A graphical depiction of a two-state Markov chain. Figure 2: The probability of being in state 0 as a function of the step number, with $P(x_i=0)$ for $x_{i+2}=0$.
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

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
- 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.

\[
\begin{align*}
\pi_0 P(x_{i+1} = 1 | x_i = 0) &= \pi_1 P(x_{i+1} = 0 | x_i = 1) \\
\pi_0 P(x_{i+1} = 0 | x_i = 1) &= \pi_1 P(x_{i+1} = 1 | x_i = 0)
\end{align*}
\]

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.

\[
\begin{align*}
\pi_0 P(x_{i+1} = 1 | x_i = 0) &= \pi_1 P(x_{i+1} = 0 | x_i = 1) \\
\pi_0 P(x_{i+1} = 0 | x_i = 1) &= \pi_1 P(x_{i+1} = 1 | x_i = 0)
\end{align*}
\]

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.

\[
\begin{align*}
\pi_0 P(x_{i+1} = 1 | x_i = 0) &= \pi_1 P(x_{i+1} = 0 | x_i = 1) \\
\pi_0 P(x_{i+1} = 0 | x_i = 1) &= \pi_1 P(x_{i+1} = 1 | x_i = 0)
\end{align*}
\]

Stationary Distribution of a Markov Chain
Stationary Distribution of a Markov Chain

Let's consider a simple Markov chain with two states, 0 and 1. The transition probabilities are given by the diagram:

- From state 0 to state 0: 0.4
- From state 0 to state 1: 0.6
- From state 1 to state 0: 0.9
- From state 1 to state 1: 0.1

The stationary distribution can be found by solving the equations:

\[ \pi_0 + \pi_1 = 1 \]

\[ \frac{\pi_0}{\pi_1} = \frac{p_{00}}{p_{10}} \]

\[ \pi_0 = 0.9 \pi_1 \]

\[ 1.5 \pi_0 + \pi_1 = 1 \]

Solving these equations, we get:

\[ \pi_0 = 0.4 \]

\[ \pi_1 = 0.6 \]

These values satisfy the stationary distribution conditions.

Notice in the discussion above that the probability of ending up in state 0 as the number of iterations increases approached 0.6. What is special about this value?

At the equilibrium process, this is the joint probability of being in state 0 at step \( i \), so:

\[ P(X_i = 0) \]

The transition probabilities are shown in the diagram.

There are second-order Markov processes that depend on the two previous states, and third-order Markov process is a positive integer. What this probability statement is saying is that, conditional on

\[ \pi_0 + \pi_1 = 1 \]

\[ \pi_0 = 0.9 \pi_1 \]

\[ 1.5 \pi_0 + \pi_1 = 1 \]

\[ \pi_0 = 0.4 \]

\[ \pi_1 = 0.6 \]
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:

flux out of \( j \) = \( \pi_j P(x_{i+1} \in S_j | x_i = j) \)

flux into \( j \) = \( \sum_{k \in S_j} \pi_k P(x_{i+1} = j | x_i = k) \)

\( \pi_j [1 - P(x_{i+1} \in S_j | x_i = j)] \) = \( \sum_{k \in S_j} \pi_k P(x_{i+1} \in S_j = \emptyset | x_i = k) \)

Flux into \( j \) = \( \pi_j [1 - P(x_{i+1} \in S_j | x_i = j)] \) = \( \sum_{k \in S_j} \pi_k P(x_{i+1} \in S_j | x_i = k) \)

Mixing

* While setting the transition probabilities to specific values affects the stationary distribution, the transition probabilities cannot be determined uniquely from the stationary distribution.
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 \( k \):

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

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

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:

\[
\pi_{\theta_j} = P(\theta_j | \text{Data})
\]
\[
\frac{t_{t,k}}{t_{k,t}} = \frac{\pi_t}{\pi_t} = \left( \frac{P(D | \theta_j = k) \pi_j \pi_t}{P(D)} \right) \div \left( \frac{P(D | \theta_j = \ell) \pi_j \pi_t}{P(D)} \right)
\]
Therefore, we need to set the transition probabilities so that

\[
\frac{t_{k,l}}{t_{l,k}} = \frac{P(D|\theta_j = k)P(\theta_j = k)}{P(D|\theta_j = l)P(\theta_j = l)}
\]

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 \( \alpha \).

If we denote by \( x'_{i+1} \) the state proposed at step \( i+1 \), then

\[
q(j, k) = P(x'_{i+1} = k|x_i = j)
\]

\[
\alpha(j, k) = P(x_{i+1} = k|x'_i = j, x'_{i+1} = k)
\]

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 \( \leq 1 \)).

We get

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

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

We have to ensure that \( q(1|k) > 0 \) whenever \( q(1|\ell) > 0 \) (it is fine if both are 0, but we can’t have one being 0 and the other greater than 0).
However, once we have chosen a proposal scheme, we do not have much flexibility in choosing whether or not to accept a proposal.

\[
\frac{t_{f,k}}{t_{k,f}} = \frac{q(f,k)\alpha(f,k)}{q(k,f)\alpha(k,f)} = \frac{P(D|\theta_j = k)P(\theta_j = k)}{P(D|\theta_j = f)P(\theta_j = f)}
\]
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.

The acceptance ratio is given by:

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

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

\[
\text{acceptance ratio} = \left( \frac{\text{likelihood ratio}}{\text{prior ratio}} \right) \left( \frac{\text{likelihood ratio}}{\text{prior ratio}} \right) ^{\text{Hastings ratio}}
\]

The three ratios in the last equation are referred to as the prior ratio, the likelihood ratio, and the Hastings ratio, respectively. The first two ratios correspond to the ratio of the numerators in Bayes' theorem; note that the complex

\[
\text{min} \left( 1, \frac{f(\theta^*|X)}{f(\theta|X)} \times \frac{f(\theta|\theta^*)}{f(\theta^*|\theta)} \right)
\]

\[
\text{min} \left( 1, \frac{f(\theta^*|X)}{f(X|\theta^*)} \times \frac{f(X|\theta)}{f(X|\theta^*)} \times \frac{f(\theta|\theta^*)}{f(\theta^*|\theta)} \right)
\]

\[
\text{min} \left( 1, \frac{f(\theta^*|X)}{f(X|\theta^*)} \times \frac{f(X|\theta)}{f(X|\theta^*)} \times \frac{f(\theta|\theta^*)}{f(\theta^*|\theta)} \right)
\]

The posterior probability distributions for different topologies are shown in the figure, with different acceptance rates for each topology.
Markov chain Monte Carlo steps
1. Start at an arbitrary point ($\theta$)
2. Make a small random move (to $\theta'$)
3. Calculate height ratio ($r$) of new state (to $\theta'$) to old state ($\theta$)
   (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

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

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

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

prior ratio  likelihood ratio  proposal ratio

An example of a proposal mechanism is the beta proposal:

- Assume the current values are $(x_1, x_2)$
- Multiply them with a value $\alpha$
- Pick new values from Beta($\alpha x_1, \alpha x_2$)
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.

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. Therefore, the plot of the likelihood values against the generation of the chain, known as the trace plot, is important in monitoring the performance of an MCMC run. However, it is extremely important to confirm convergence using other diagnostic tools because 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.
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.

The mixing behavior of a Metropolis sampler can be adjusted using its tuning parameter(s). Assume, for instance, that we are sampling from a normal distribution using a sliding window proposal (Fig. 7.6). The sliding window proposal has one tuning parameter, the width of the window. If the width is too small, then the proposed value will be very similar to the current one (Fig. 7.6a). The posterior probabilities will also be very similar, so the proposal will tend to be accepted. But each proposal will only move the chain a tiny distance in parameter space, so it will take the chain a long time to cover the entire region of interest; mixing is poor.
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
## Summary

<table>
<thead>
<tr>
<th>Method</th>
<th>Advantage</th>
<th>Disadvantage</th>
<th>Software</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neighbor Join</td>
<td>Fast</td>
<td>Information is lost during compression of sequences; multiple biologically plausible estimates of pairwise distances can be hard to obtain</td>
<td>RAXML, MRBAYES, PAUP*</td>
</tr>
<tr>
<td>Parsimony</td>
<td>Fast enough for the analysis of hundreds of biologically related sequences on a desktop computer</td>
<td>Distance estimation not biologically plausible because sequences are not biologically related sequences or there are dependencies in the data</td>
<td>RAXML, MRBAYES, PAUP*</td>
</tr>
<tr>
<td>Maximum likelihood</td>
<td>Uses models to correct for uneven changes</td>
<td>Distance corrections can be broken down when data are average</td>
<td>RAXML, MRBAYES, PAUP*</td>
</tr>
<tr>
<td>Maximum likelihood</td>
<td>The method fully captures what the biologically related sequences are</td>
<td>Can be prohibitively slow depending on the model (Bayesian models are computationally expensive)</td>
<td>RAXML, MRBAYES, PAUP*</td>
</tr>
</tbody>
</table>

## Acknowledgment

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

## Questions?