Likelihood and Bayesian Inference

David L. Swofford

Research Associate
Florida Museum of Natural History
University of Florida

With much thanks to:

Paul Lewis (Univ. of Connecticut)

from whom I have stolen many of the slides for this intro
(with his permission)

2018 Workshop on Population and Speciation Genomics

Český Krumlov, Czechia

Topics covered

- Review of probability
- Principles of maximum likelihood estimation
- Introduction to Bayesian inference and MCMC

This intro will be very basic. I will assume that you have little understanding of (or have forgotten everything you knew about):

- basic probability
- statistics
- calculus

The idea is for everyone to reach a basic starting point before we proceed further.

PLEASE INTERRUPT!

Note: For technical reasons, I'm not on Slack--please use email (david.swofford@duke.edu) to contact me!

Why probability?

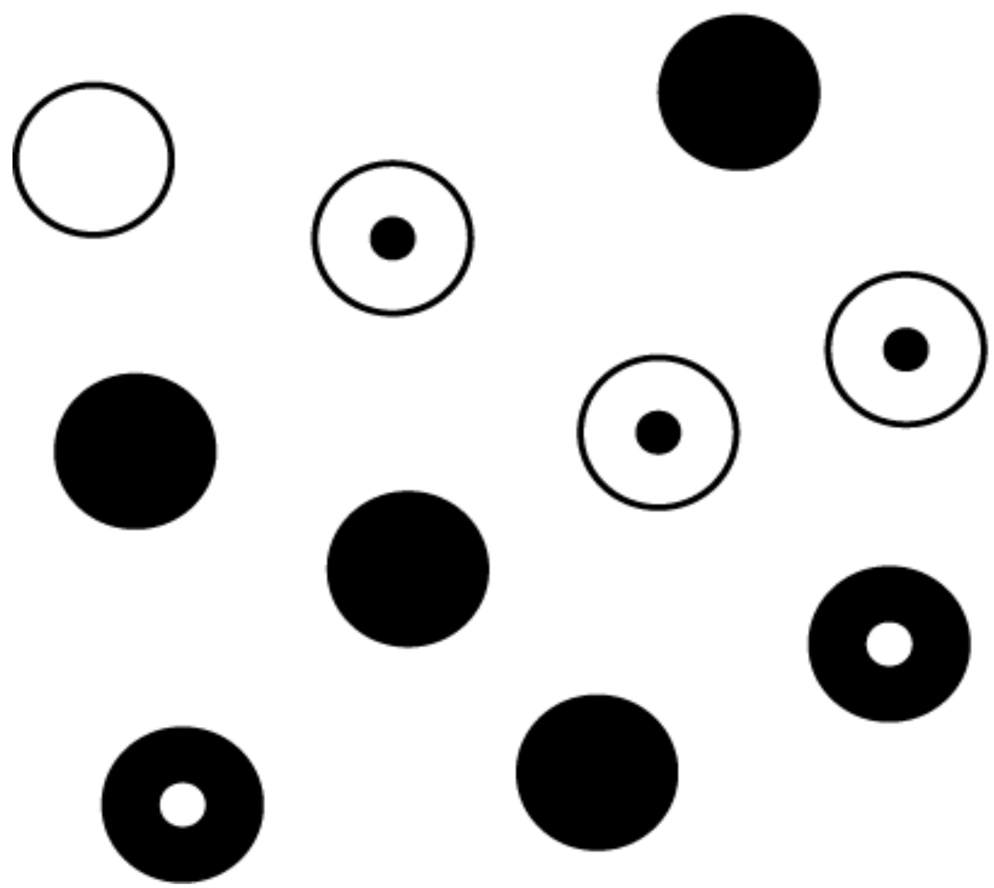
We want to estimate biology relevant quantities (parameters) from our data and probability provides the conceptual and analytical basis for this effort

- Mutation rates
- Population sizes
- Recombination rates
- Migration rates
- Selection coefficients
- Gene and species trees
- Branch lengths and divergence times
- Substitution-model parameters

... to name a few

Joint probabilities

B = Black S = Solid
W = White D = Dotted



Pr(B) = 0.6 Pr(S) = 0.5

Pr(W) = 0.4 Pr(D) = 0.5

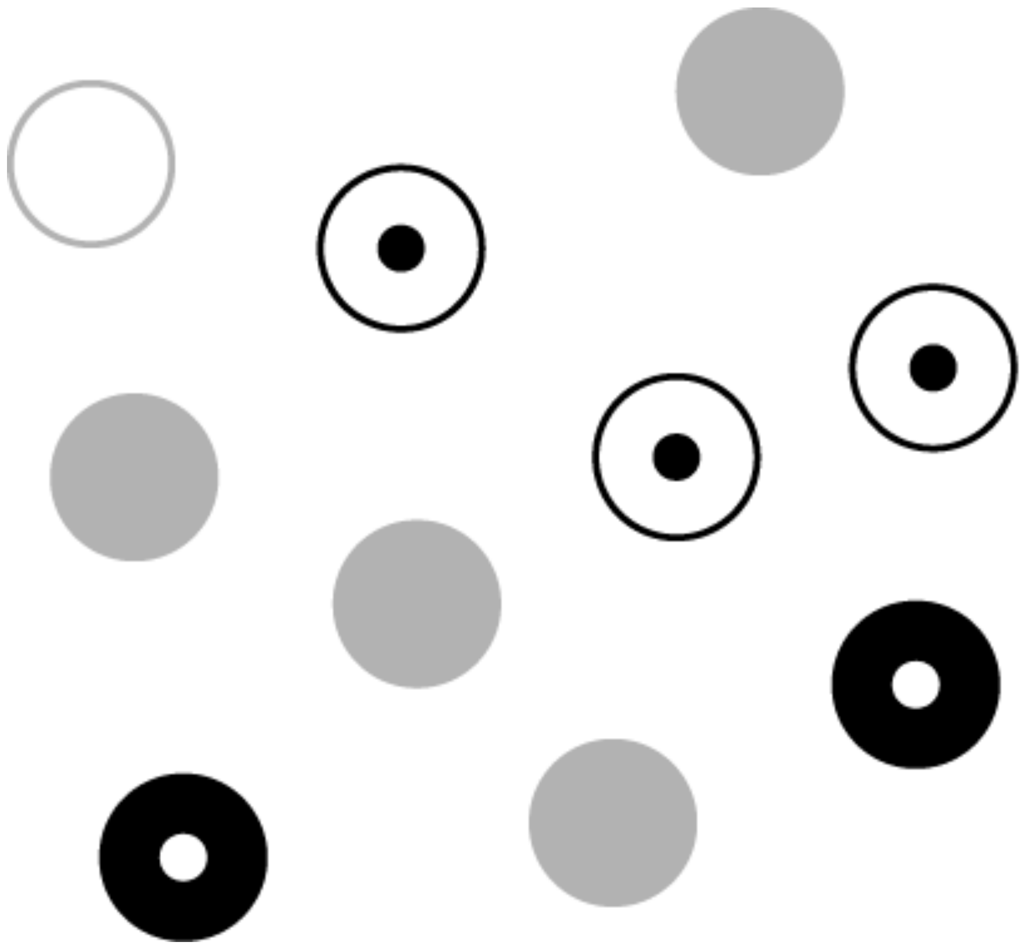
Pr(◉) = Pr(B, D) = 0.2

Pr(●) = Pr(B, S) = 0.4

Pr(⊙) = Pr(W, D) = 0.3

Pr(○) = Pr(W, S) = 0.1

Conditional probabilities



$$Pr(B|D) = \frac{2}{5} = 0.4$$

Hide all solid marbles
(leaving 5 with dot)

Of those left, 2 are black

Maximum likelihood

$$Likelihood(\theta) = \Pr\left(D \mid \theta\right)$$

To compute a likelihood, we need a model. This model allows us to compute the probability of obtaining our observed data for any given value(s) of the model parameter(s) Θ

Coin tossing as an example

Heads (H)   Tails (T)

Model: result of each toss is independent of other tosses; Pr(H) is unknown but constant across tosses

$$\theta = \Pr(H)$$

Coin tossing as an example

Suppose we toss the coin 5 times and
get the following result:

H T H H T

Simple model: Pr(H)=Pr(T)=0.5

Under independence assumption,

$$\Pr(H,T,H,H,T) = (0.5)\,(0.5)\,(0.5)\,(0.5)\,(0.5) = (0.5)^5$$

$$= 0.03125$$

(All other sequences have the same probability)

Coin tossing as an example

A more interesting question: how probable is *h* heads and *t* tails in N tosses?

HHHHH 5 H, 0 T

HHHHT
HHHTH
HHTHH
HTHHH
THHHH
4 H, 1 T

HHHTT
HHTHT
HTHHT
THHHT
HHTTH
HTHTH
THHTH
HTTHH
THTHH
TTHHH
3 H, 2 T

2 H, 3 T
TTTHH
TTHTH
THTTH
HTTTH
TTHHT
THTHT
HTTHT
THHTT
HTHTT
HHTTT

1 H, 4 T
TTTTH
TTTHT
TTHTT
THTTT
HTTTT

0 H, 5 T TTTTT

Coin tossing as an example

Binomial probability

$$\Pr\left(h \text{ heads} \mid N \text{ tosses}\right) = \begin{pmatrix} N \\ h \end{pmatrix} \theta^{h} \left(1-\theta\right)^{N-h}$$

If we let *Θ = 0.5,*

$$\Pr\left(3 \text{ heads} \mid 5 \text{ tosses}\right) = \begin{pmatrix} 5 \\ 3 \end{pmatrix} 0.5^{3} 0.5^{2}$$

$$= \frac{5!}{3!2!} 0.5^{5}$$

$$= 10(.03125)$$

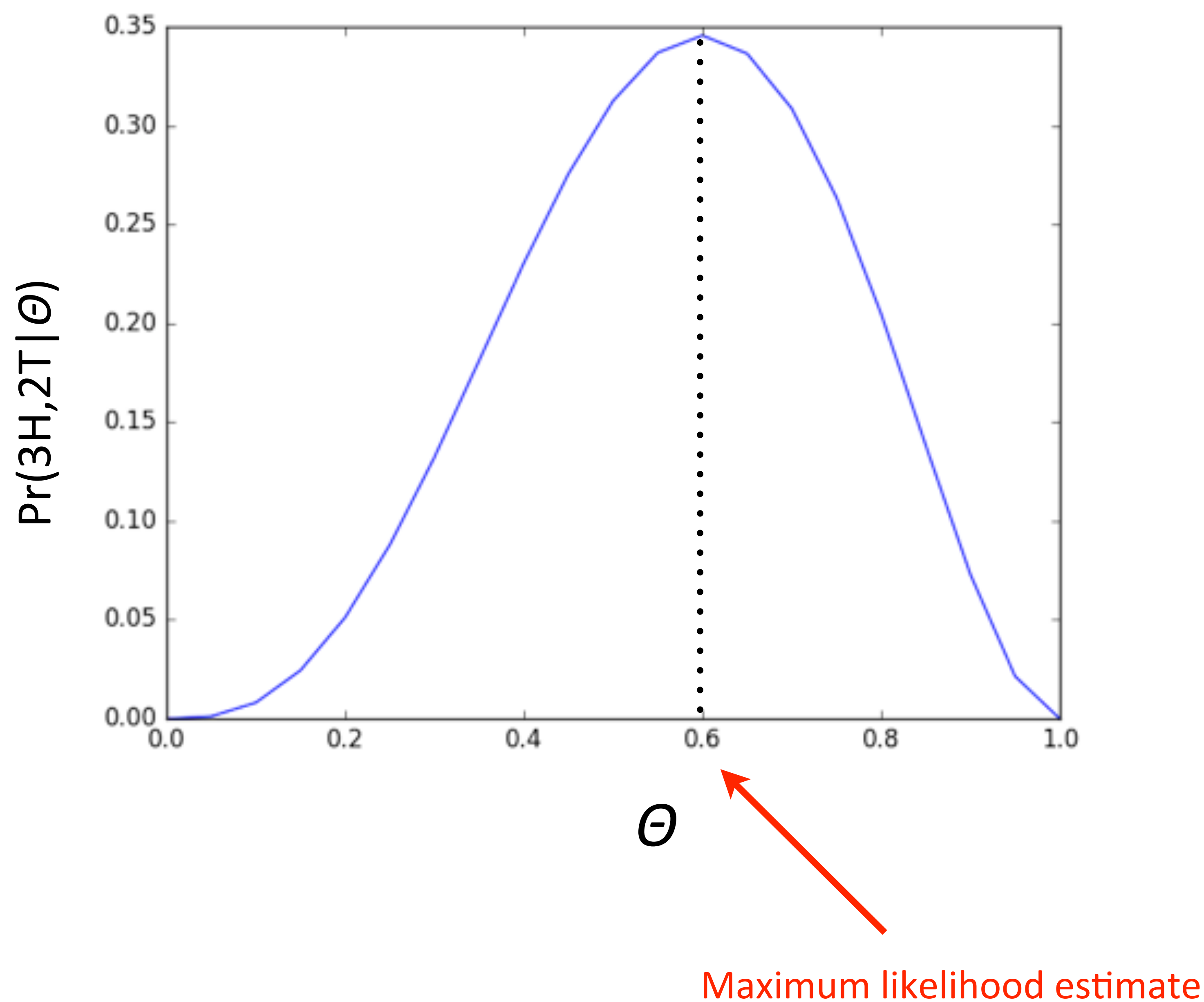
$$= 0.3125$$

Coin tossing as an example

What if we aren't willing to assume that $\Theta = 0.5$?

Estimate Θ by maximum likelihood...

$Pr(3H,2T|\ \Theta = 0.0) = 0.0$
$Pr(3H,2T|\ \Theta = 0.1) = 0.0081$
$Pr(3H,2T|\ \Theta = 0.2) = 0.0512$
$Pr(3H,2T|\ \Theta = 0.3) = 0.1323$
$Pr(3H,2T|\ \Theta = 0.4) = 0.2304$
$Pr(3H,2T|\ \Theta = 0.5) = 0.3125$
$Pr(3H,2T|\ \Theta = 0.6) = 0.3456$
$Pr(3H,2T|\ \Theta = 0.7) = 0.3087$
$Pr(3H,2T|\ \Theta = 0.8) = 0.2048$
$Pr(3H,2T|\ \Theta = 0.9) = 0.0729$
$Pr(3H,2T|\ \Theta = 1.0) = 0.0$

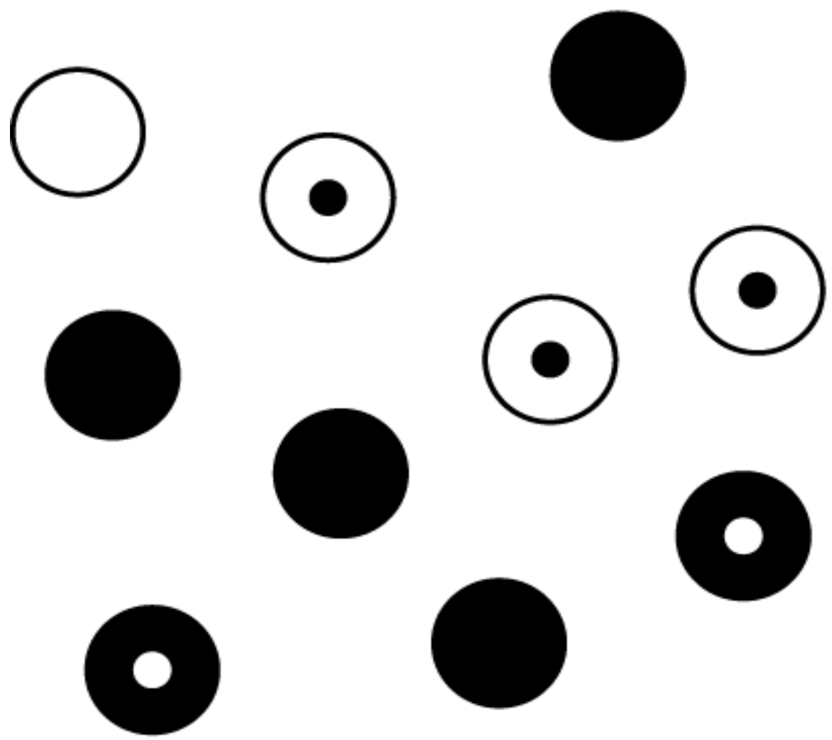


Bayes' rule

$$\Pr(B, D)$$

$$\Pr(D)\,\Pr(B|D) = \Pr(B)\,\Pr(D|B)$$

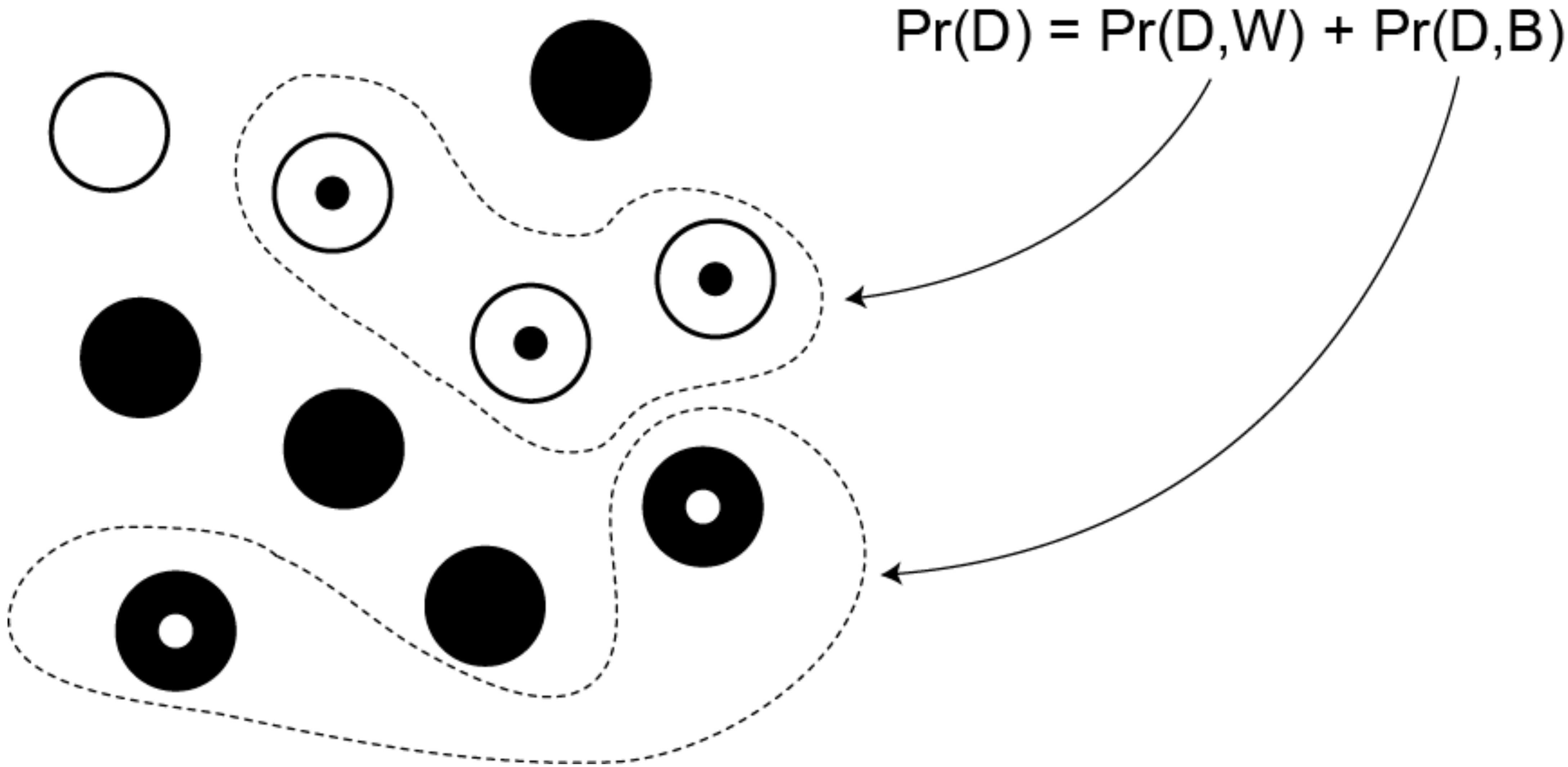
$$\frac{1}{2} \times \frac{2}{5} = \frac{3}{5} \times \frac{1}{3}$$



$$\Pr(B|D) = \frac{\Pr(B)\,\Pr(D|B)}{\Pr(D)}$$

$$= \frac{\frac{3}{5} \times \frac{1}{3}}{\frac{1}{2}} = \frac{2}{5}$$

Probability of "Dotted"



Bayes' rule (cont.)

$$\Pr(B|D) = \frac{\Pr(B)\Pr(D|B)}{\Pr(D)}$$

$$= \frac{\Pr(D,B)}{\Pr(D,B) + \Pr(D,W)}$$

Pr(*D*) is the **marginal probability** of being dotted
To compute it, we **marginalize over colors**

Bayes' rule (cont.)

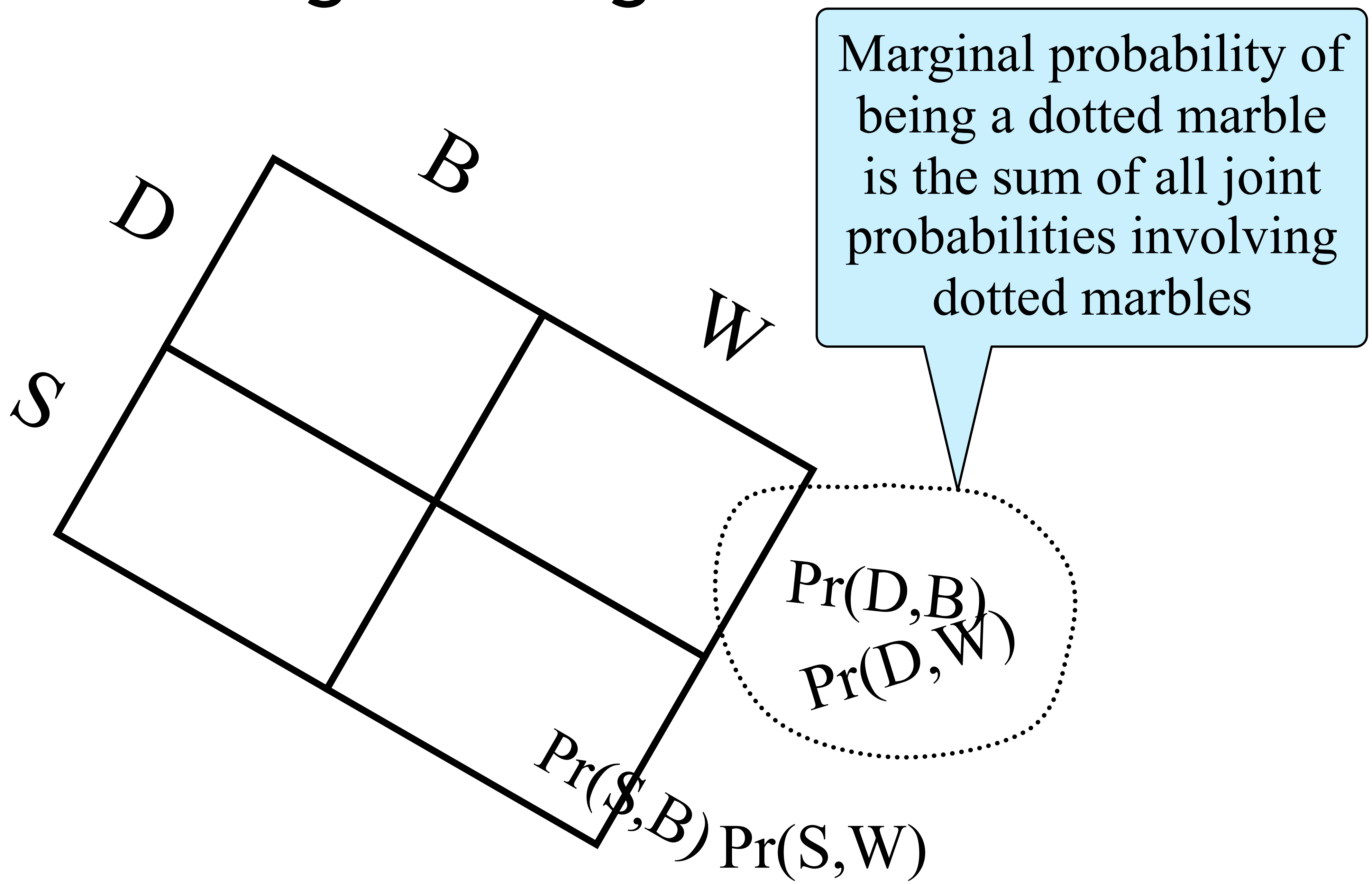
It is easy to see that Pr(*D*) serves as a *normalization constant*, ensuring that Pr(*B*|*D*) + Pr(*W*|*D*) = 1.0

$$\Pr(B|D) = \frac{\Pr(D,B)}{\Pr(D,B) + \Pr(D,W)} \quad \longleftarrow \Pr(D)$$

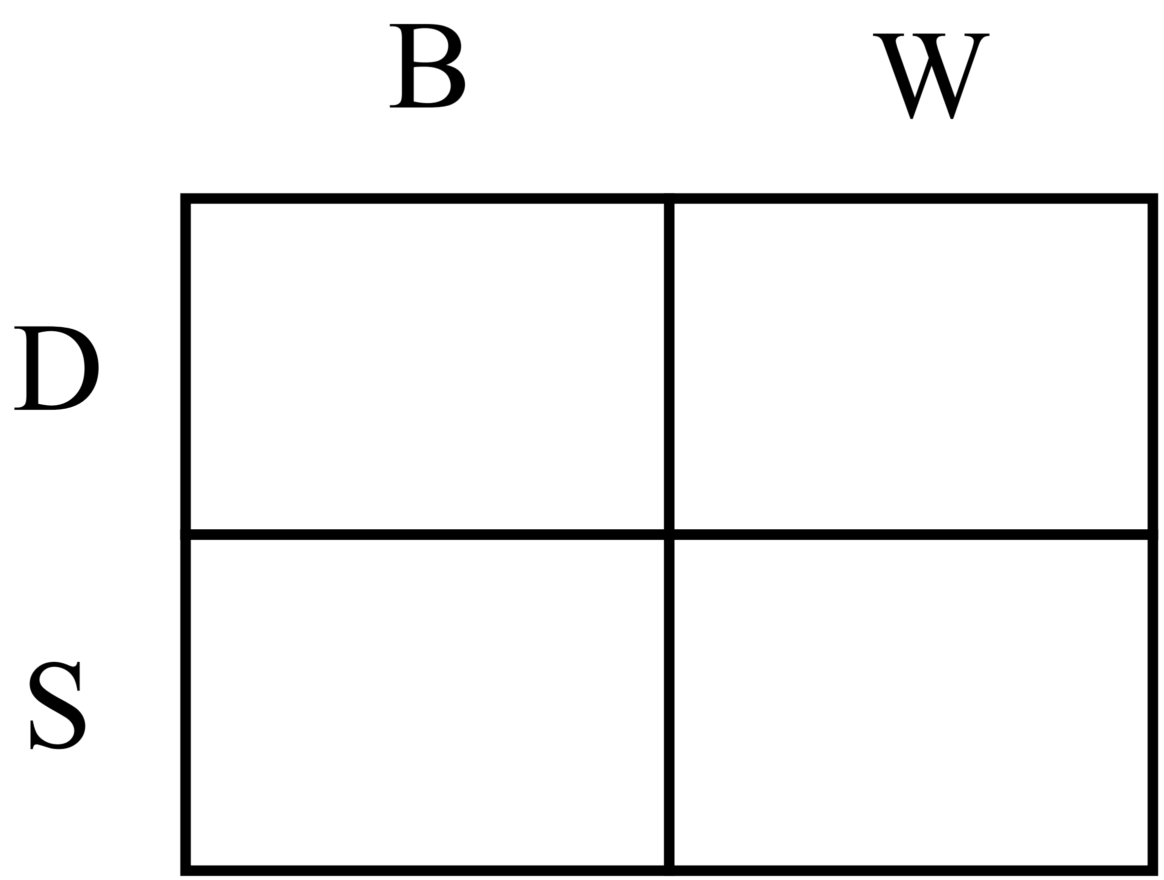
$$\Pr(W|D) = \frac{\Pr(D,W)}{\Pr(D,B) + \Pr(D,W)} \quad \longleftarrow \Pr(D)$$

$$\Pr(B|D) + \Pr(W|D) = \frac{\cancel{\Pr(D,B) + \Pr(D,W)}}{\cancel{\Pr(D,B) + \Pr(D,W)}} = 1$$

Marginalizing over colors



Marginal probabilities



Pr(D) = marginal probability of being dotted

$$\Pr(D,B) + \Pr(D,W)$$

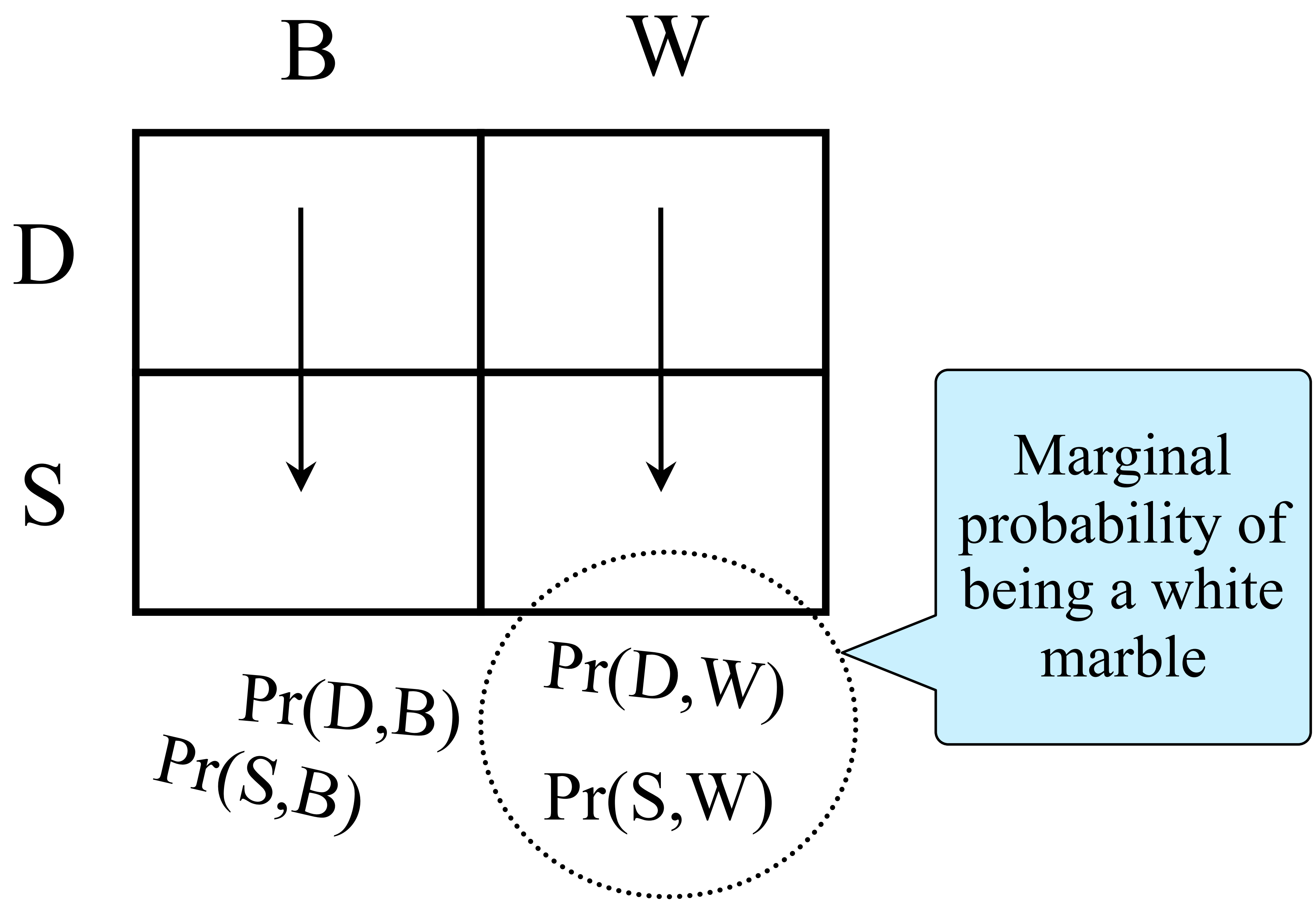
$$\Pr(S,B) + \Pr(S,W)$$

Pr(S) = marginal probability of being solid

Joint probabilities

	B	W
D	Pr(D,B)	Pr(D,W)
S	Pr(S,B)	Pr(S,W)

Marginalizing over "dottedness"



Bayes' rule (cont.)

$$\Pr(B|D) = \frac{\Pr(B)\Pr(D|B)}{\Pr(D,B)+\Pr(D,W)}$$

$$= \frac{\Pr(B)\Pr(D|B)}{\Pr(B)\Pr(D|B)+\Pr(W)\Pr(D|W)}$$

$$= \frac{\Pr(B)\Pr(D|B)}{\sum_{\theta\in\{B,W\}}\Pr(\theta)\Pr(D|\theta)}$$

Bayes' rule in statistics

Likelihood of hypothesis θ

Prior probability of hypothesis θ

$$\Pr(\theta|D) = \frac{\Pr(D|\theta)\,\Pr(\theta)}{\sum_\theta \Pr(D|\theta)\,\Pr(\theta)}$$

Posterior probability
of hypothesis θ

Marginal probability
of the data (marginalizing
over hypotheses)

Practical application of Bayes' rule

(modified from Durbin et al. 1998 *Biological Sequence Analysis*)

A rare genetic disease is discovered. Although only one in a million people carry it, you consider getting screened. You are told that the genetic test is extremely good; it is 100% sensitive (it is always correct if you have the disease), and it has a false positive rate of only 1%. If you have the disease, a new drug can save your life if taken before the onset of symptoms; it costs $10,000/year.

$$\Pr(\text{disease}|+) = \frac{\Pr(+|\text{disease}) \times \Pr(\text{disease})}{\Pr(+|\text{disease}) \times \Pr(\text{disease}) + \Pr(+|\text{healthy}) \times \Pr(\text{healthy})}$$

$$= \frac{1 \times 0.000001}{1 \times 0.000001 + 0.01 \times 0.999999}$$

$$= 0.00009999$$

$$\Pr(\text{healthy}|+) = \frac{\Pr(+|\text{healthy}) \times \Pr(\text{healthy})}{\Pr(+|\text{disease}) \times \Pr(\text{disease}) + \Pr(+|\text{healthy}) \times \Pr(\text{healthy})}$$

$$= \frac{0.01 \times 0.999999}{1 \times 0.000001 + 0.01 \times 0.999999}$$

$$= 0.99990001$$

If test positive, approximately 10,000 times more likely to NOT have the disease than to have it!

(Is it worth $10,000?)

Simple (albeit silly) paternity example

θ_1 and θ_2 are assumed to be the only possible fathers, child has genotype Aa, mother has genotype aa, so child must have received allele **A** from the true father. Note: the data in this case is the child's genotype (Aa)

Possibilities	θ_1	θ_2	Row sum
Genotypes	AA	Aa	---
Prior	1/2	1/2	1
Likelihood	1	1/2	---
Prior X Likelihood	1/2	1/4	3/4
Posterior	2/3	1/3	1

Bayes' rule: continuous case

Likelihood

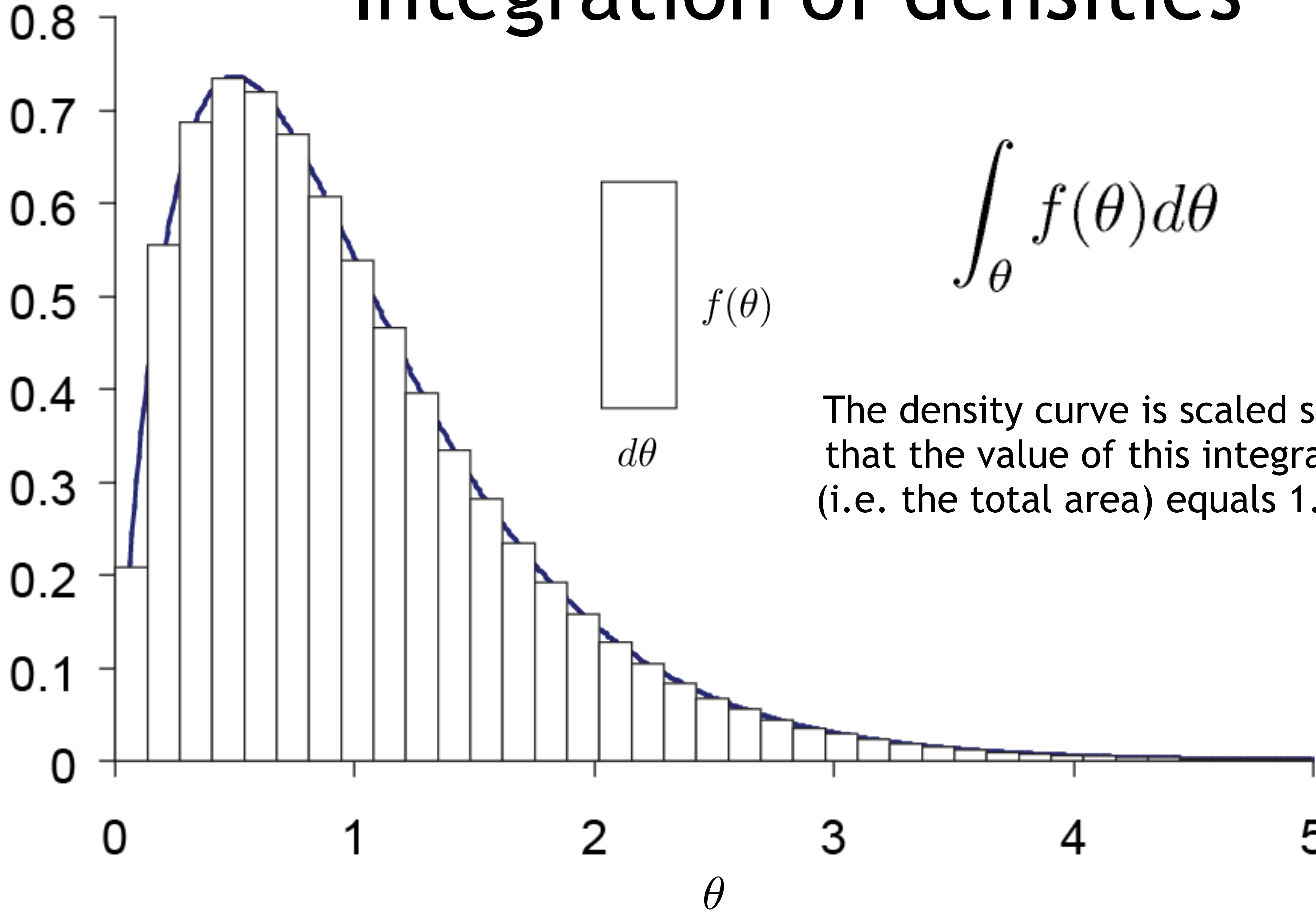
Prior probability ***density***

$$f(\theta|D) = \frac{f(D|\theta)\,f(\theta)}{\int f(D|\theta)\,f(\theta)\,d\theta}$$

Posterior probability ***density***

Marginal probability of the data

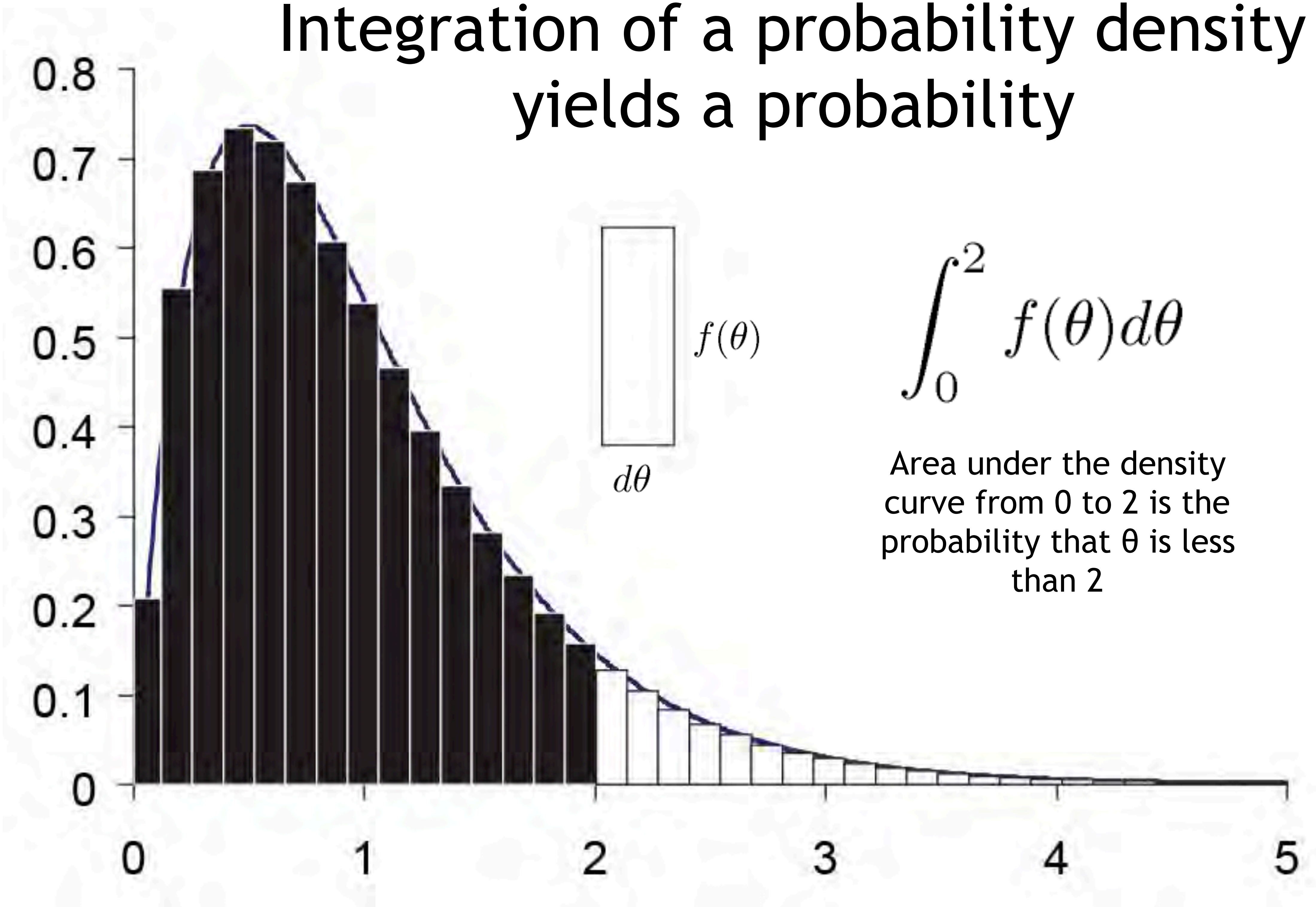
Integration of densities



$$\int_{\theta} f(\theta) d\theta$$

The density curve is scaled so that the value of this integral (i.e. the total area) equals 1.0

Integration of a probability density yields a probability



$$\int_0^2 f(\theta)d\theta$$

Area under the density curve from 0 to 2 is the probability that θ is less than 2

Usually there are many parameters...

A 2-parameter example

Likelihood

Prior probability density

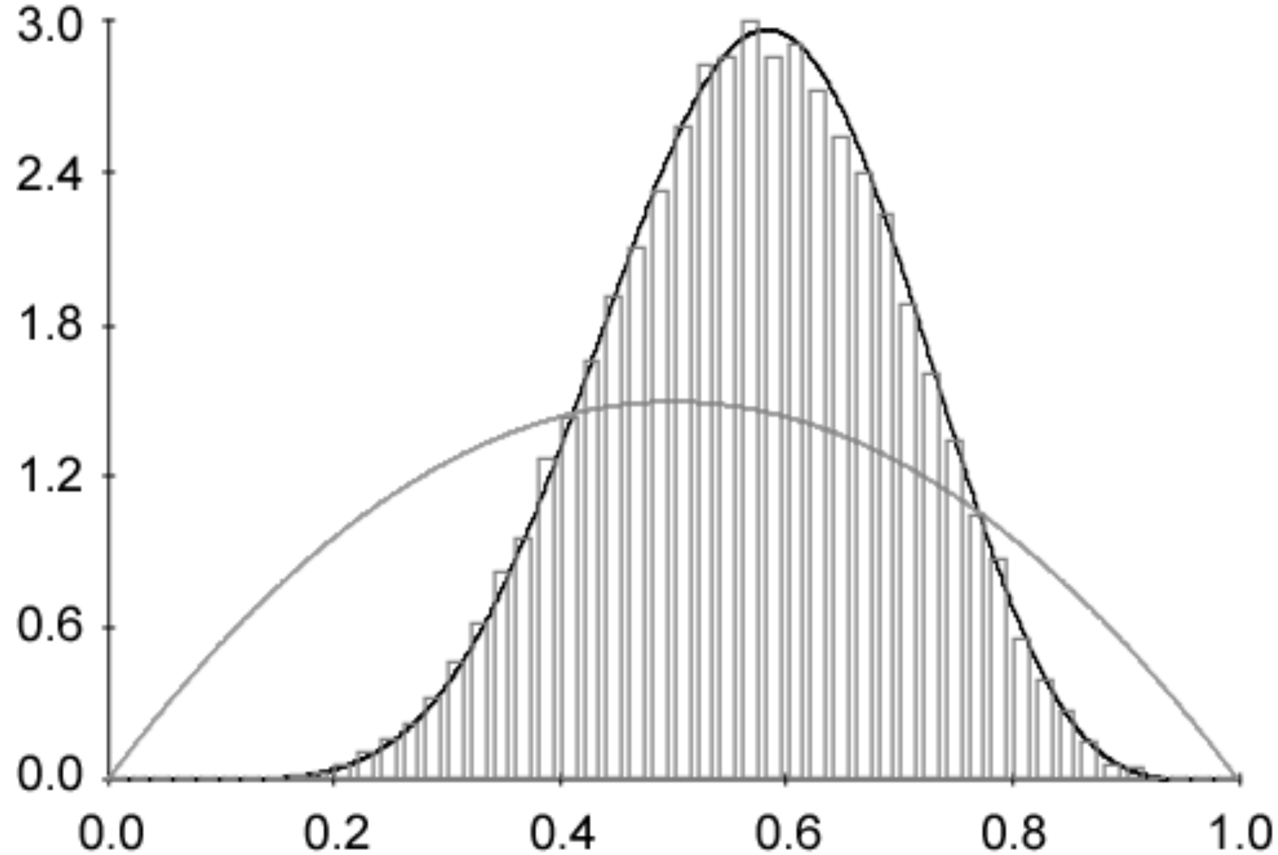
$$f(\theta, \phi|D) = \frac{f(D|\theta, \phi)\ f(\theta) f(\phi)}{\int_\theta \int_\phi f(D|\theta, \phi)\ f(\theta) f(\phi)\ d\theta d\phi}$$

Posterior probability density

Marginal probability of data

An analysis of **100 sequences** under the simplest model (JC69) requires 197 branch length parameters. The denominator is a **197-fold integral** in this case! Now consider summing over **all possible tree topologies**! It would thus be nice to avoid having to calculate the marginal probability of the data...

Markov chain Monte Carlo (MCMC)

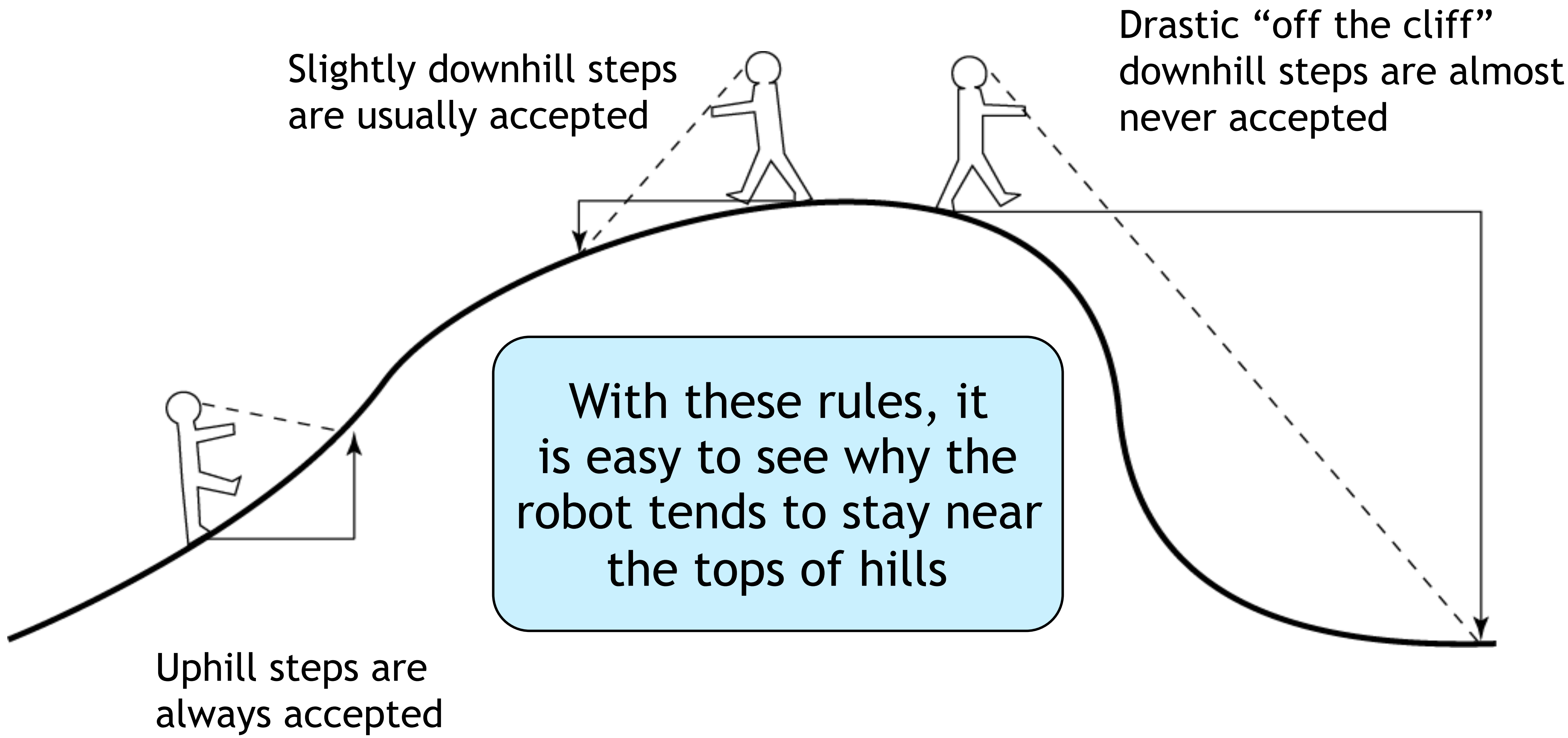


For more complex problems, we might settle for a

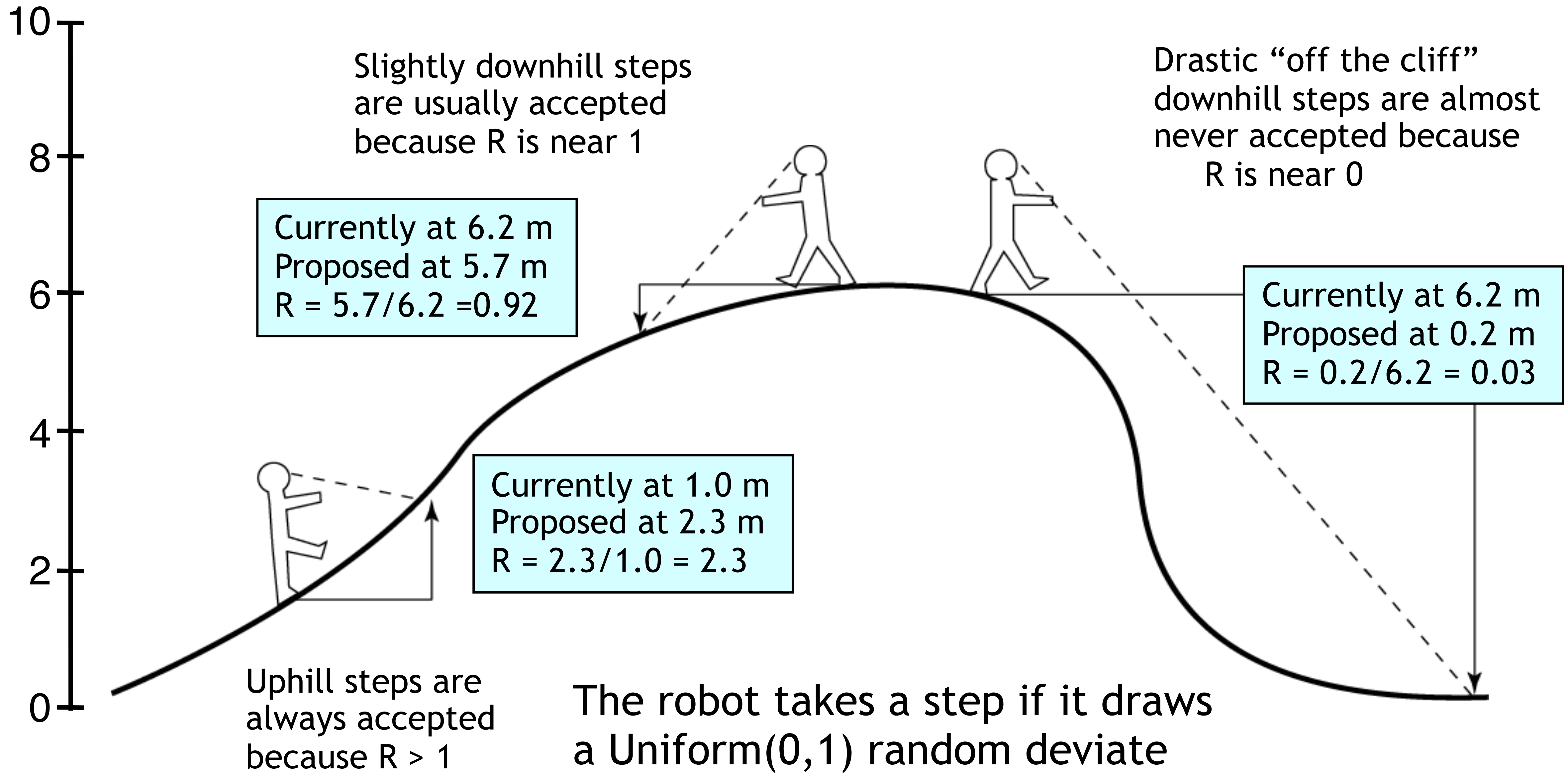
good approximation

to the posterior distribution

MCMC robot's rules



(Actual) MCMC robot rules



The robot takes a step if it draws a Uniform(0,1) random deviate that is less than or equal to R

Cancellation of marginal likelihood

When calculating the ratio *R* of posterior densities, the marginal probability of the data cancels.

$$\frac{f(\theta^*|D)}{f(\theta|D)} = \frac{\dfrac{f(D|\theta^*)f(\theta^*)}{\cancel{f(D)}}}{\dfrac{f(D|\theta)f(\theta)}{\cancel{f(D)}}} = \frac{f(D|\theta^*)f(\theta^*)}{f(D|\theta)f(\theta)}$$

Posterior odds

Likelihood ratio

Prior odds

Cancellation of marginal likelihood

When calculating the ratio *R* of posterior densities, the marginal probability of the data cancels.

$$\frac{f(\theta^*|D)}{f(\theta|D)} = \frac{\dfrac{f(D|\theta^*)f(\theta^*)}{\cancel{f(D)}}}{\dfrac{f(D|\theta)f(\theta)}{\cancel{f(D)}}} = \frac{f(D|\theta^*)f(\theta^*)}{f(D|\theta)f(\theta)}$$

Posterior odds

Likelihood ratio

Prior odds

MCRobot (or "MCMC Robot")

https://phylogeny.uconn.edu/mcmc-robot/

Bayesian coin-tossing with MCMC

```
# A tiny little Python program to demonstrate MCMC
# Dave Swofford, 22 January 2018

# NOTE: This code is written for clarity/readability, not efficiency!  Do NOT use it as the
#       basis for a real MCMC program.

from math import exp, sqrt
from scipy.stats import binom, beta
import numpy as np
from numpy import random

do_monte_carlo_sim = False
do_mcmc = True
sample_from_prior = False             # run "without data" if true

def reflect_back(x, xmin, xmax):
        while x < xmin or x > xmax:
                if x < xmin:
                        x = 2*xmin - x
                else:
                        x = 2*xmax - x
        return x

##############################
# Simulation of coin tossing #
##############################
if do_monte_carlo_sim:
        num_iters = 1
        num_tosses = 5
        p = 0.5
        print "\n%10s%10s%12s\n%s" % ("H", "T", "p(H)", '-'*32)
        for iter in range(num_iters):
                num_heads = random.binomial(num_tosses, p, 1)
                print "%10d%10d%12.5f" % (num_heads, num_tosses - num_heads, float(num_heads)/num_tosses)

# Generate a data set:
num_tosses = 5
true_theta = 0.5
num_heads = random.binomial(num_tosses, true_theta, 1)
num_tails = num_tosses - num_heads
print "\nSimulation of coin tossing performed: %d heads, %d tails" % (num_heads, num_tails)
```

Bayesian coin-tossing with MCMC

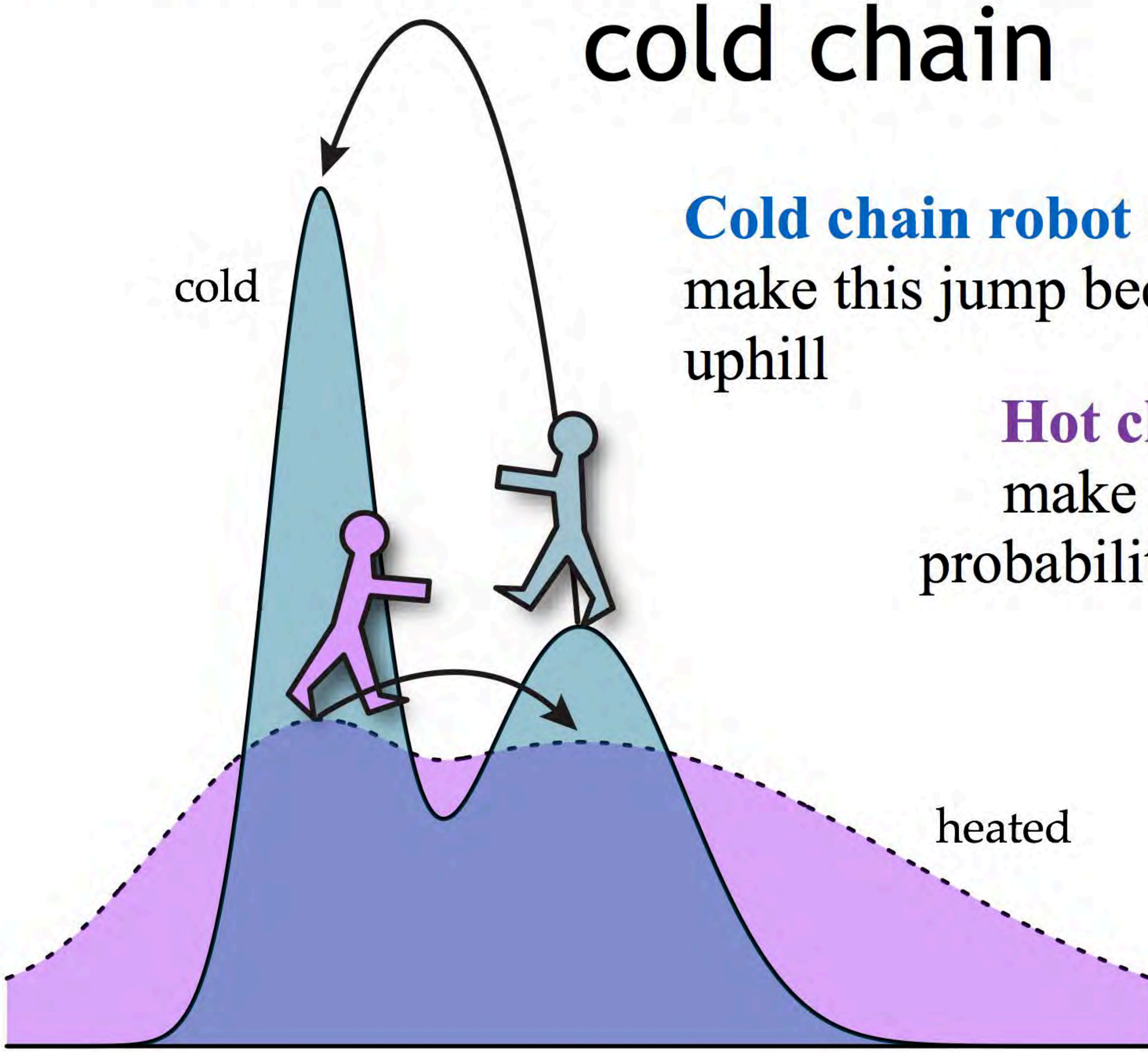
```
#################################
# Estimate theta=Pr(H) via MCMC #
#################################
if do_mcmc:
	a = 0.2						# alpha parameter of Beta distribution
	b = 0.2						# beta parameter of Beta distribution
	w = 0.5						# width for sliding window proposal
	mcmc_iters = 10000		# set number of MCMC iterations (generations)
	hastings_ratio = 1.0	# we're using a symmetric proposal distribution

	# Open a file to receive the posterior samples:
	fp = open("samples.txt", "w")

	# We'll use a random draw from the prior as the starting point
	theta = random.beta(a, b)
	fp.write("%s\t%s\t%s\t%s\t%s\t%s\t%s\t%s\t%s\t%s\n" %
		("iter", "theta", "thetaStar", "prior_theta", "prior_thetaStar", "like_theta",
		"like_thetaStar", "post_theta", "post_thetaStar", "R"))
	fp.write("%d\t%.10f\t%.10f\t%.10f\t%.10f\t%.10f\t%.10f\t%.10f\t%.10f\t%.10f\n" %
		 (0, theta, 0, 0, 0, 0, 0, 0, 0, 0))

	# Begin MCMC iterations using this starting point
	num_accepted = 0
	for iter in range(mcmc_iters):

		# Propose a new theta using sliding window proposal with window width w
		thetaStar = random.uniform(theta - w/2.0, theta + w/2.0)
		if thetaStar < 0 or thetaStar > 1:
			thetaStar = reflect_back(thetaStar, 0.0, 1.0)

		# Calculate acceptance probability and decide whether or not to accept
		prior_theta = beta.pdf(theta, a, b)
		prior_thetaStar = beta.pdf(thetaStar, a, b)
		if sample_from_prior:
			like_theta = 1.0
			like_thetaStar = 1.0
		else:
			like_theta = binom.pmf(num_heads, num_tosses, theta)
			like_thetaStar = binom.pmf(num_heads, num_tosses, thetaStar)
		post_theta = prior_theta * like_theta
		post_thetaStar = prior_thetaStar * like_thetaStar
		posterior_odds = post_thetaStar / post_theta
		r = posterior_odds * hastings_ratio
		if r >= 1.0:
			theta = thetaStar
			num_accepted += 1
		else:
			u = random.random()			# random draw from Uniform(0, 1)
			if r > u:
				theta = thetaStar
				num_accepted += 1
		fp.write("%d\t%.10f\t%.10f\t%.10f\t%.10f\t%.10f\t%.10f\t%.10f\t%.10f\t%.10f\n" %
			(iter + 1, theta, thetaStar, prior_theta, prior_thetaStar, like_theta,
			like_thetaStar, post_theta, post_thetaStar, r))

	fp.close()
	acceptanceRate = float(num_accepted)/mcmc_iters
	print "\nMCMC completed; acceptance ratio for theta proposals =", acceptanceRate
```

Metropolis-coupled Markov chain Monte Carlo (MCMCMC)

- MCMCMC involves running **several chains simultaneously**
- The **cold chain** is the one that counts, the rest are **heated chains**
- Chain is heated by raising densities to a power less than 1.0 (values closer to 0.0 are warmer)

Geyer, C. J. 1991. Markov chain Monte Carlo maximum likelihood for dependent data. Pages 156-163 *in* Computing Science and Statistics (E. Keramidas, ed.).

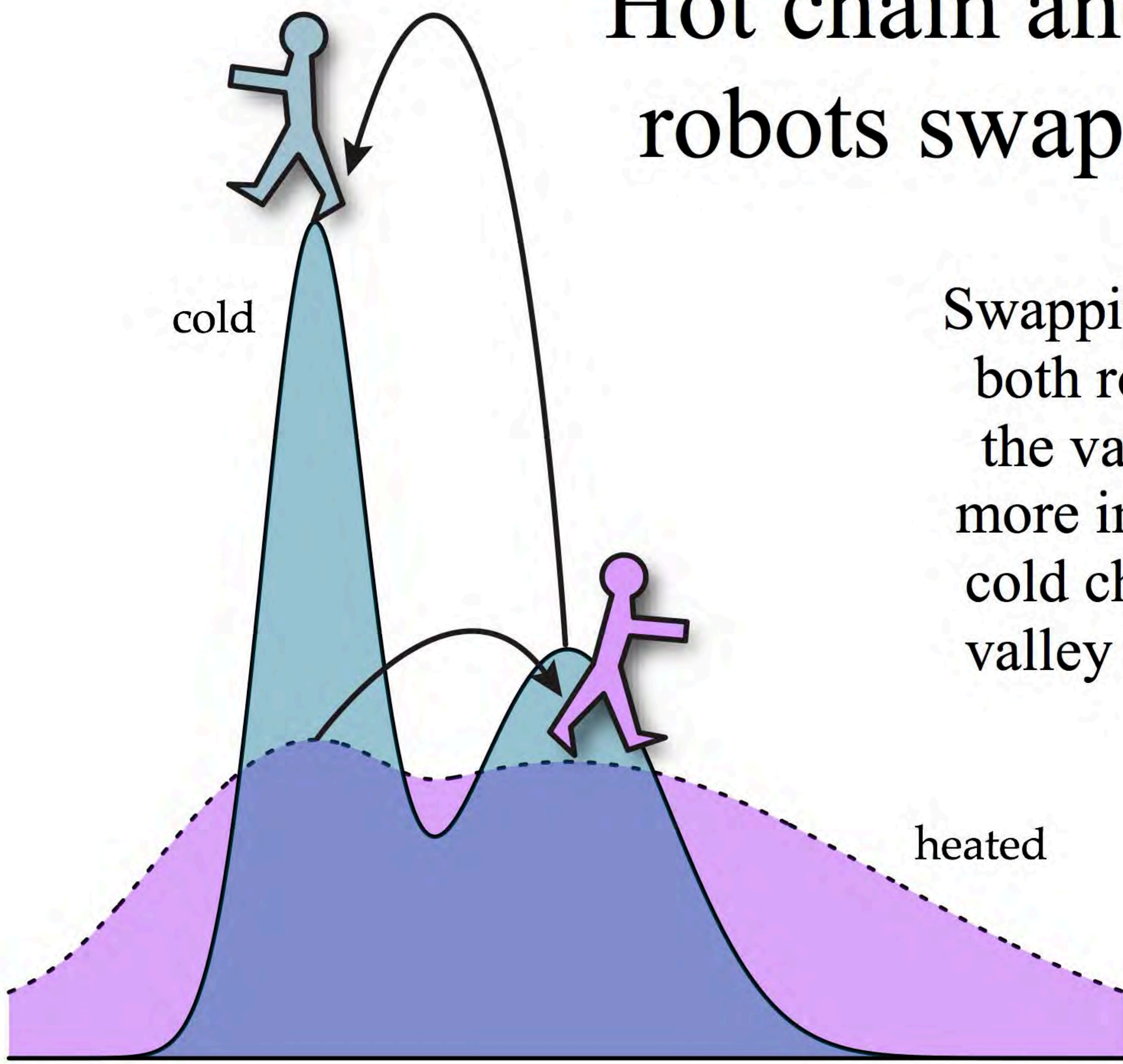
Heated chains act as scouts for the cold chain



Cold chain robot can easily make this jump because it is uphill

Hot chain robot can also make this jump with high probability because it is only slightly downhill

Hot chain and cold chain robots swapping places

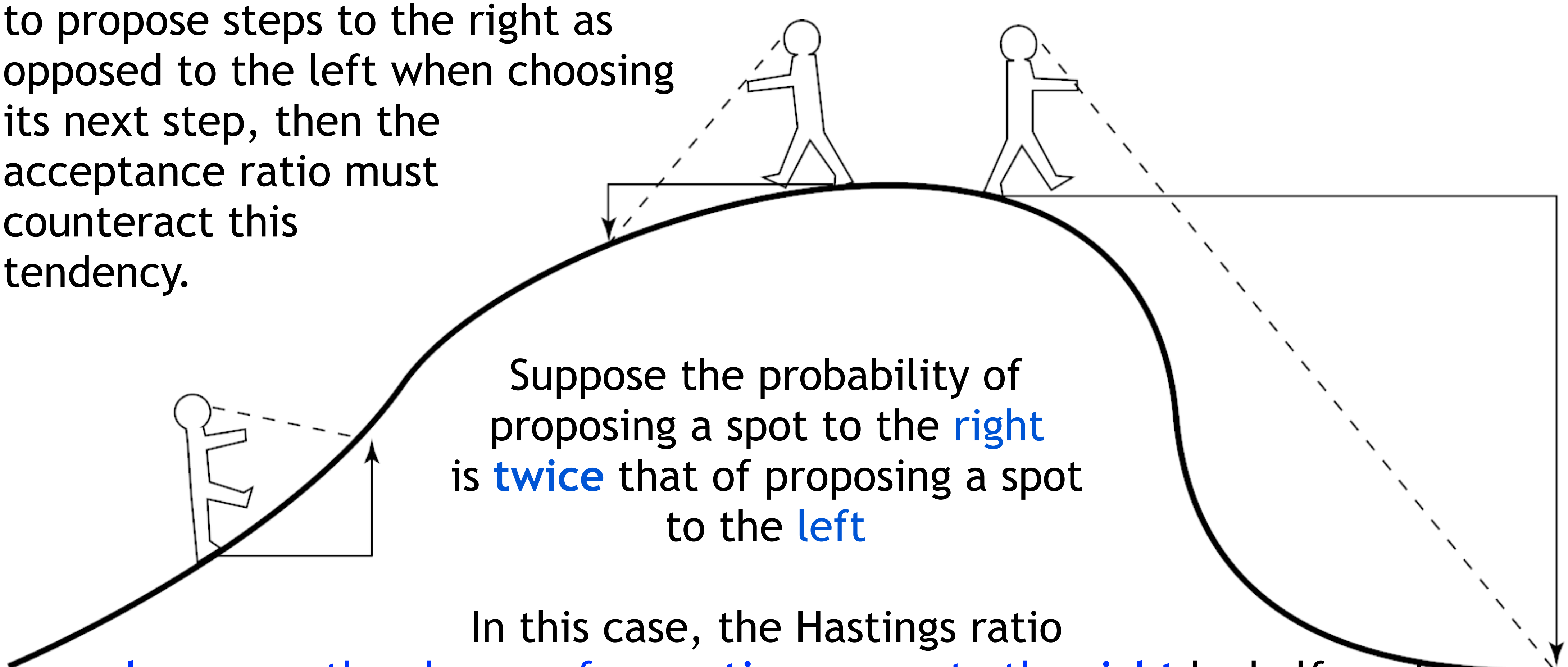


Swapping places means both robots can cross the valley, but this is more important for the cold chain because its valley is much deeper

Back to MCRobot...

The Hastings ratio

If robot has a greater tendency to propose steps to the right as opposed to the left when choosing its next step, then the acceptance ratio must counteract this tendency.



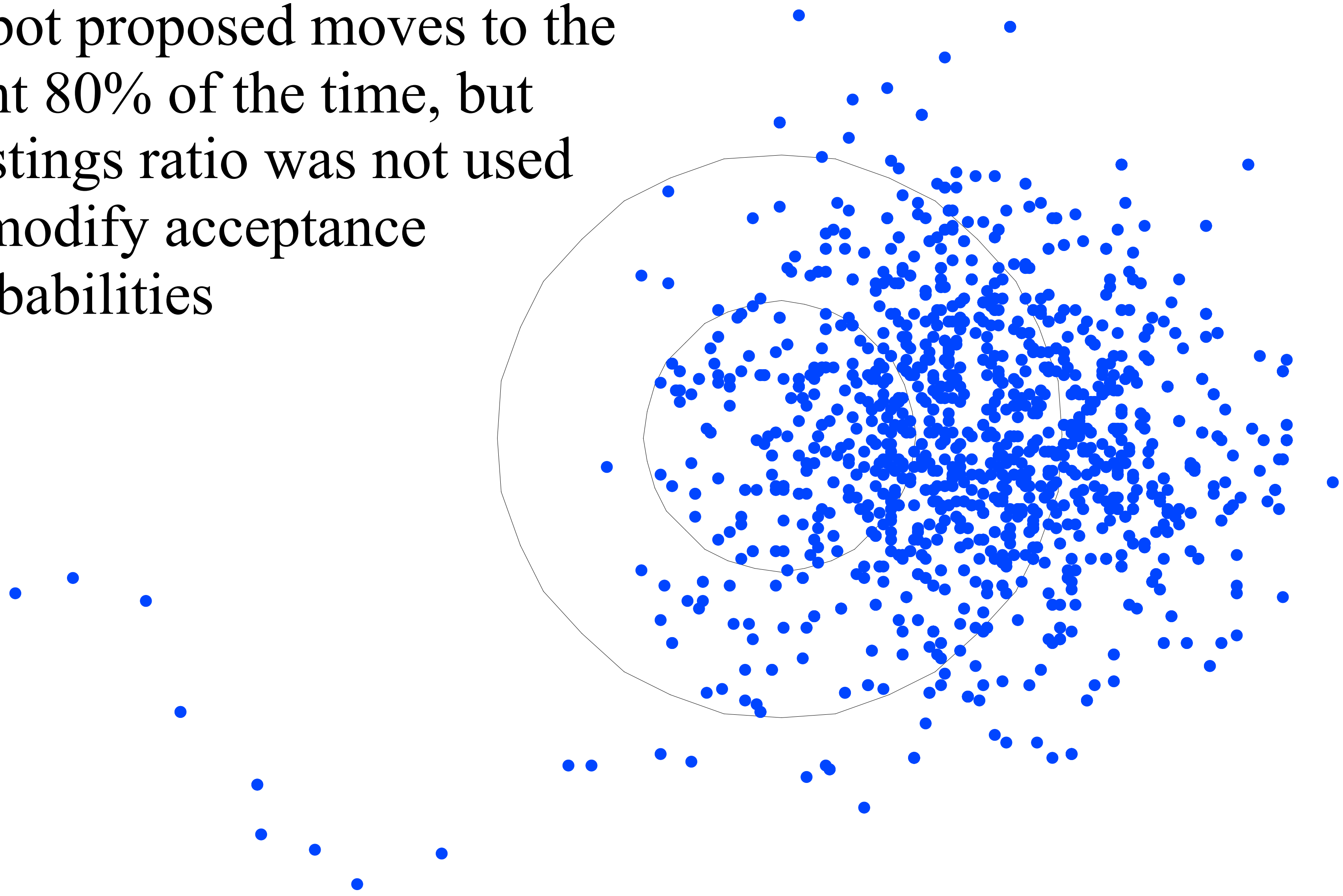
Suppose the probability of proposing a spot to the right is **twice** that of proposing a spot to the left

In this case, the Hastings ratio **decreases** the chance of **accepting** moves to the **right** by half, and **increases** the chance of **accepting** moves to the **left** (by a factor of 2), thus **exactly compensating** for the asymmetry in the proposal distribution.

Hastings, W. K. 1970. Monte Carlo sampling methods using Markov chains and their applications. Biometrika 57:97-109.

The Hastings ratio

Example where MCMC Robot proposed moves to the right 80% of the time, but Hastings ratio was not used to modify acceptance probabilities



Hastings Ratio

$$R = \left[\frac{f(D|\theta^*)\ f(\theta^*)}{f(D|\theta)\ f(\theta)}\right] \left[\frac{q(\theta|\theta^*)}{q(\theta^*|\theta)}\right]$$

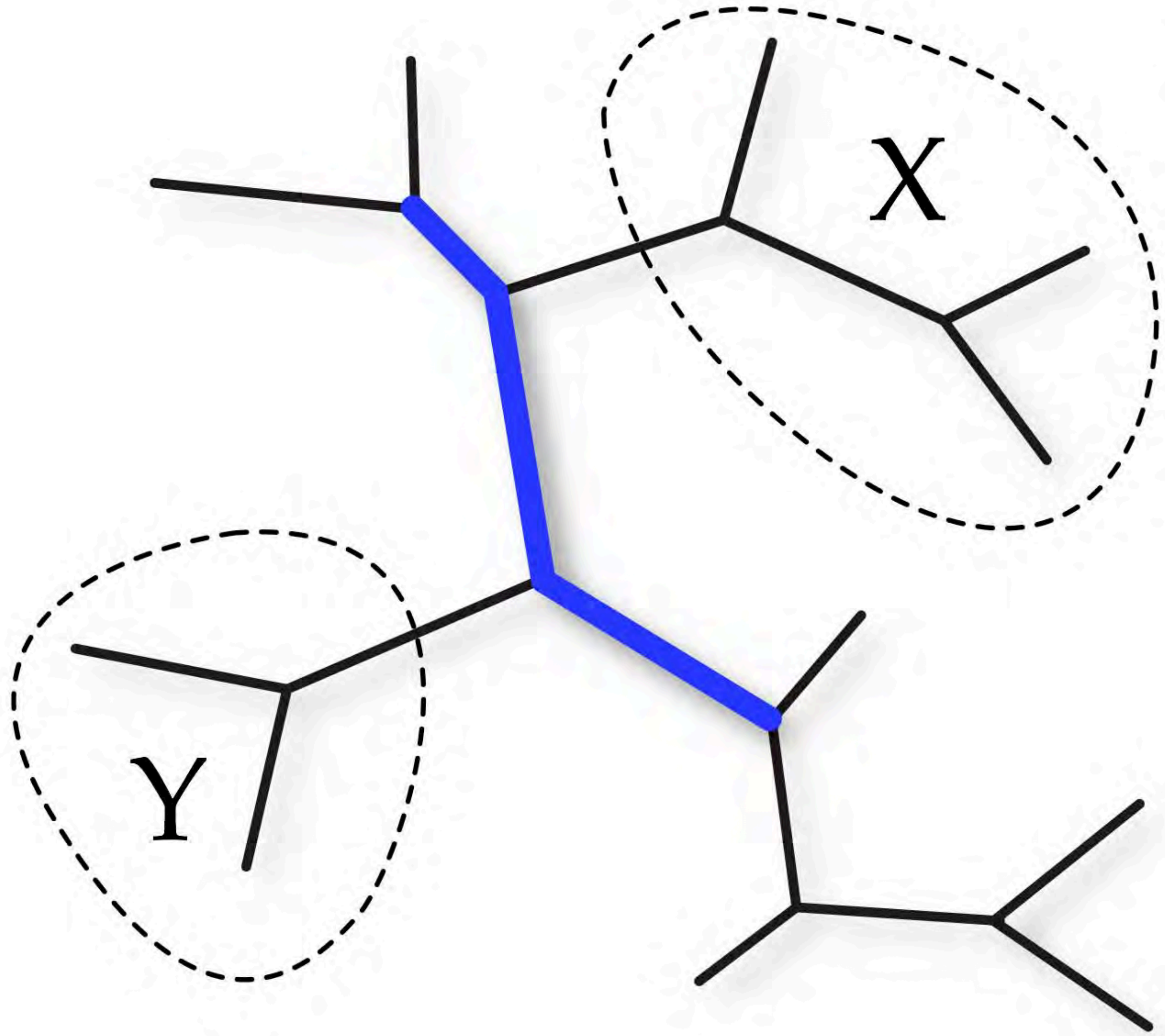
Acceptance ratio

Posterior ratio

Hastings ratio

Note that if $q(\theta|\theta^*) = q(\theta^*|\theta)$, the Hastings ratio is 1

Moving through treespace

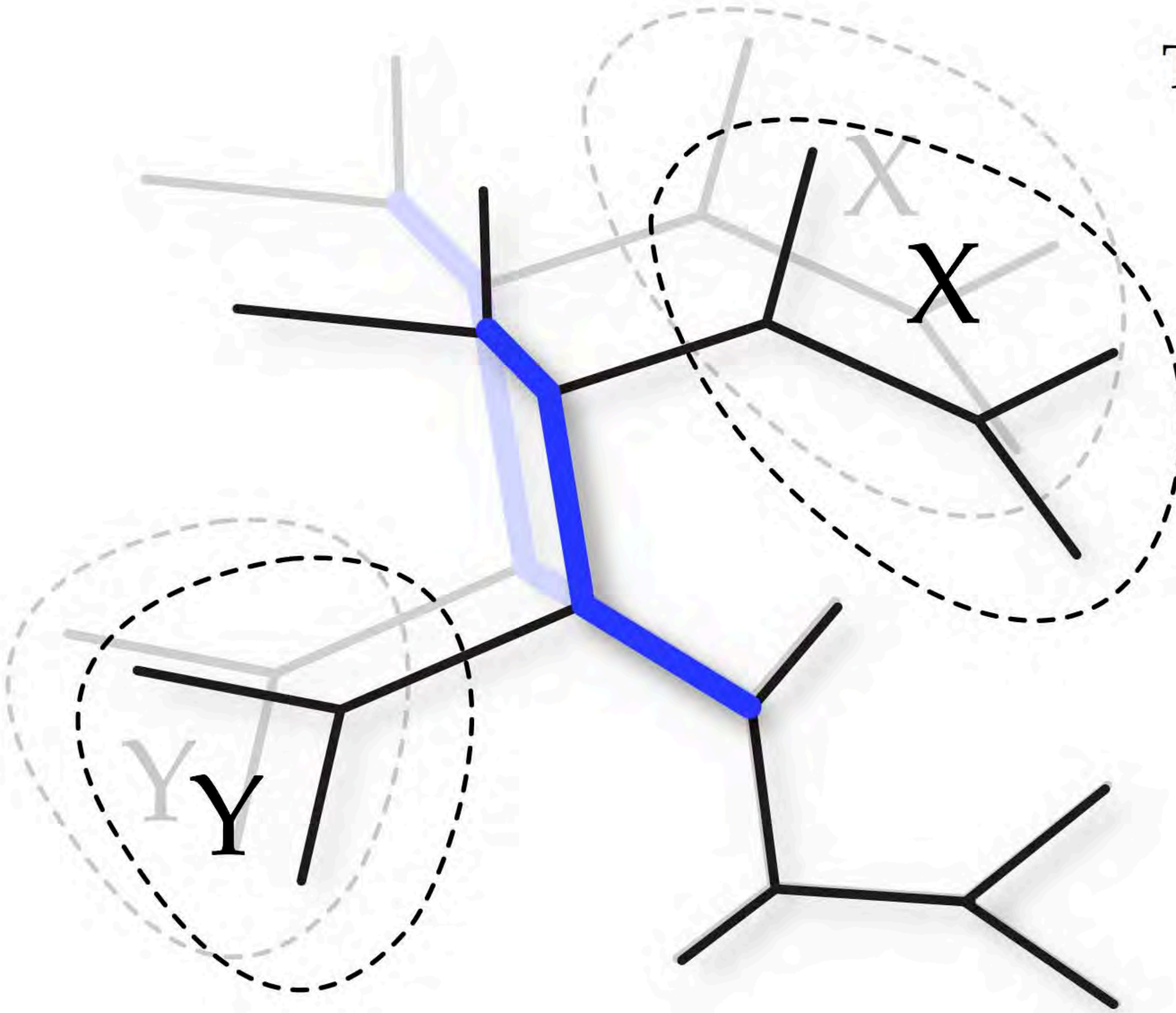


The Larget-Simon move

Step 1:
Pick 3 contiguous edges randomly, defining two subtrees, X and Y

*Larget, B., and D. L. Simon. 1999. Markov chain monte carlo algorithms for the Bayesian analysis of phylogenetic trees. Molecular Biology and Evolution 16: 750-759. See also: Holder et al. 2005. Syst. Biol. 54: 961-965.

Moving through treespace

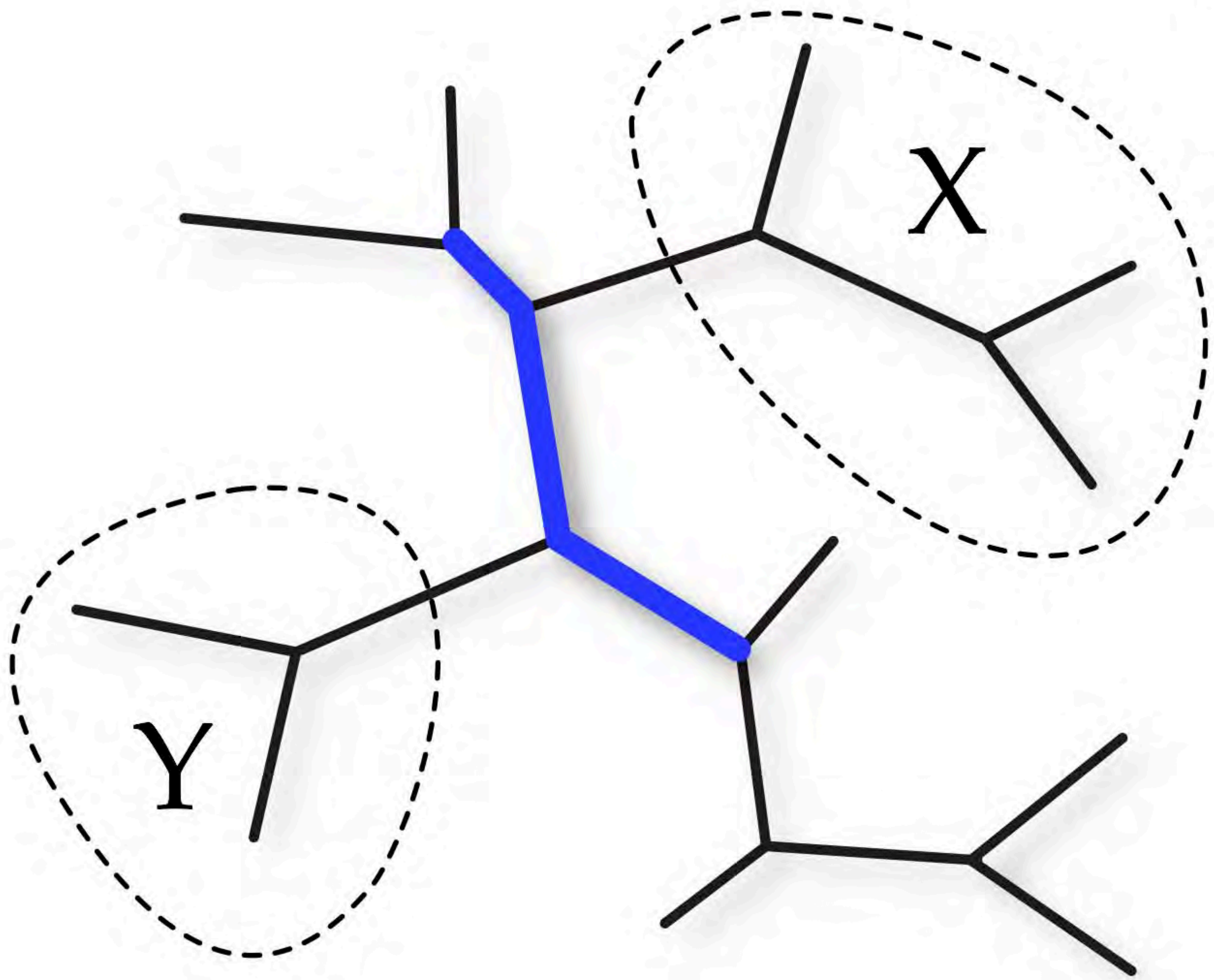


The Larget-Simon move

Step 1:
Pick 3 contiguous edges randomly, defining two subtrees, X and Y

Step 2:
Shrink or grow selected 3-edge segment by a random amount

Moving through treespace

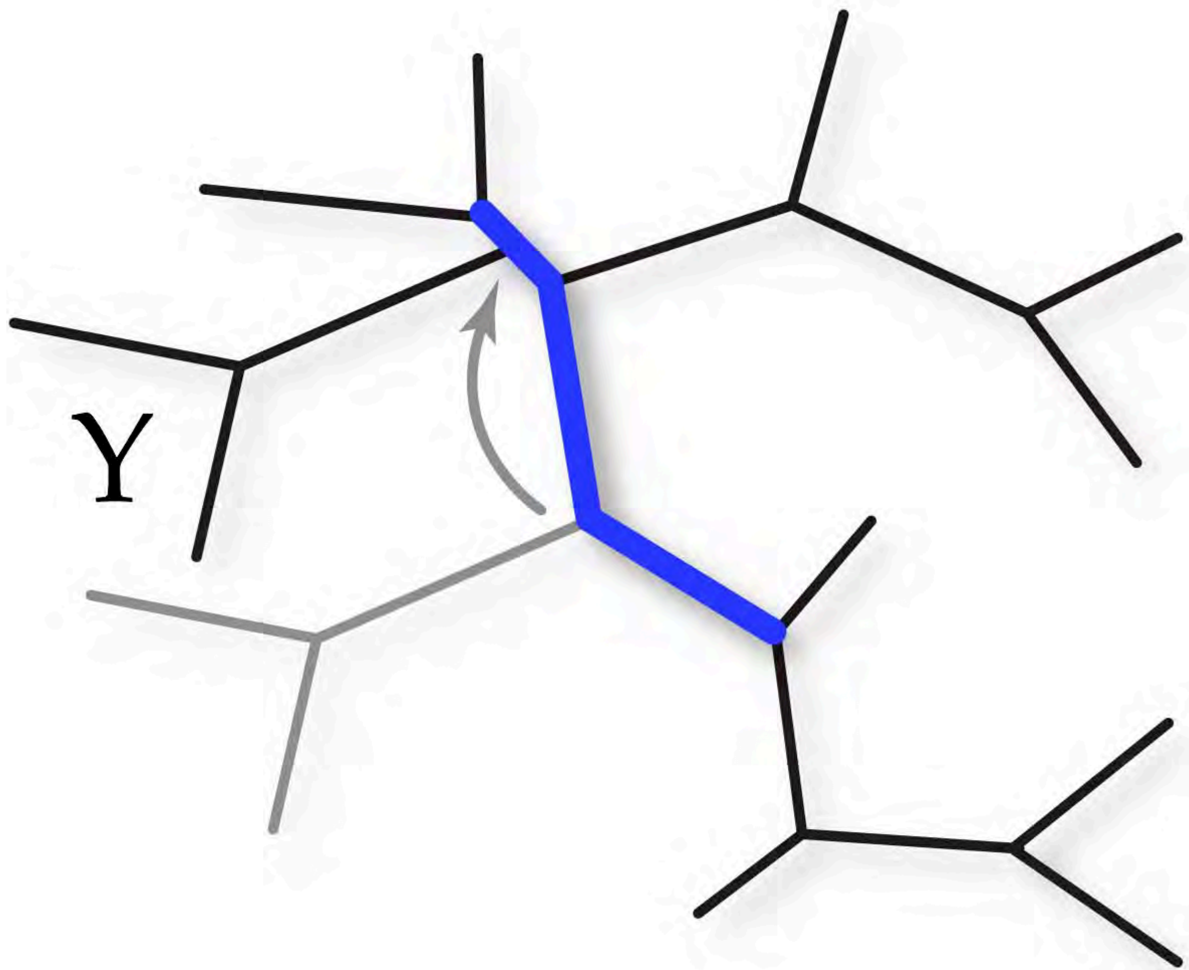


The Larget-Simon move

Step 1:
Pick 3 contiguous edges randomly, defining two subtrees, X and Y

Step 2:
Shrink or grow selected 3-edge segment by a random amount

Moving through treespace



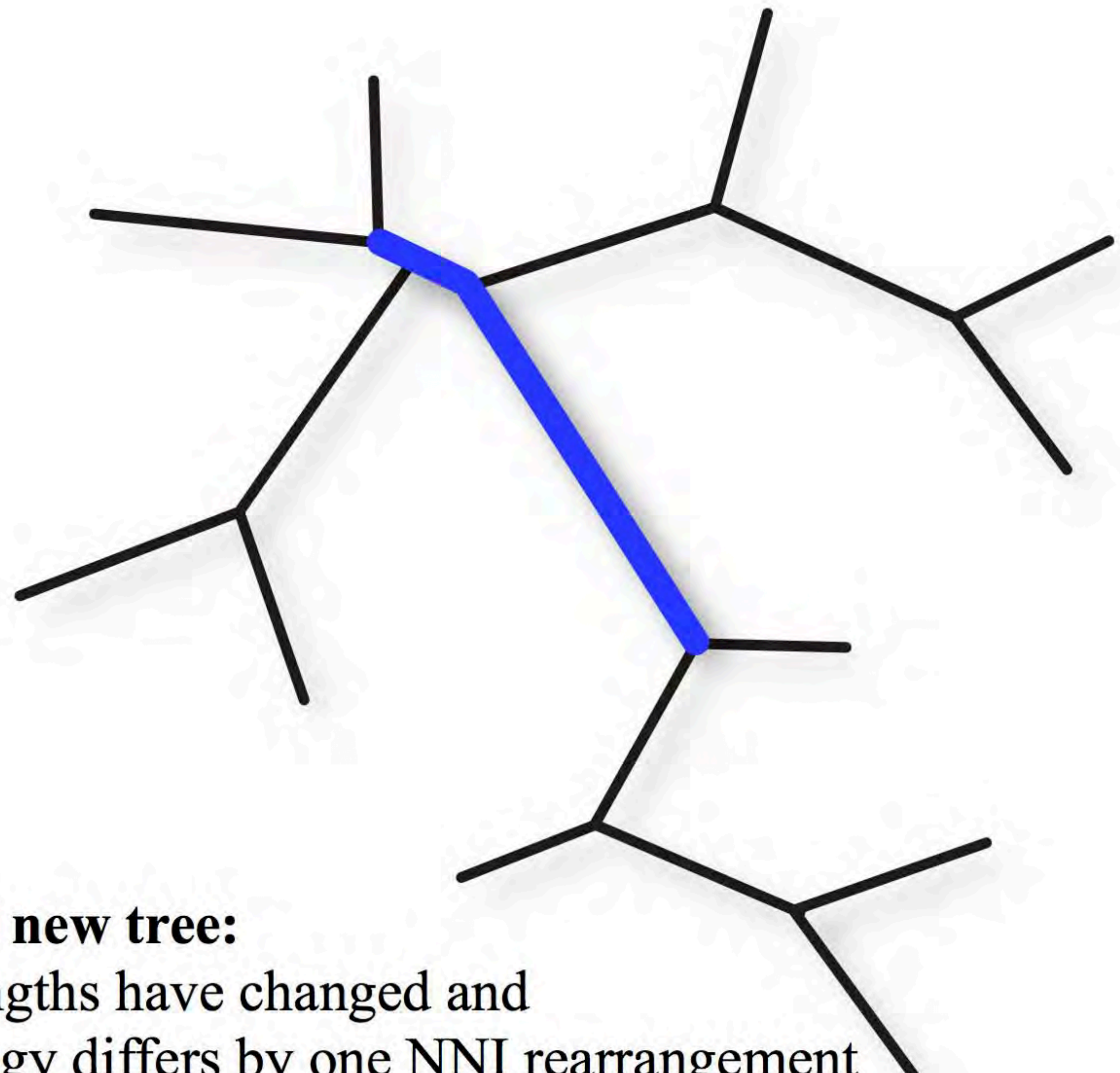
The Larget-Simon move

Step 1:
Pick 3 contiguous edges randomly, defining two subtrees, X and Y

Step 2:
Shrink or grow selected 3-edge segment by a random amount

Step 3:
Choose X or Y randomly, then reposition randomly

Moving through treespace



Proposed new tree:
3 edge lengths have changed and
the topology differs by one NNI rearrangement

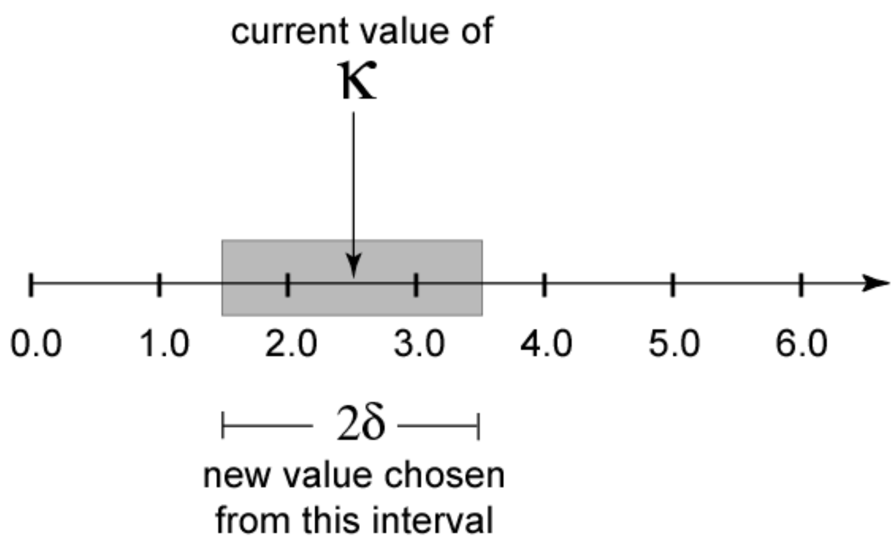
The Large-Simon move

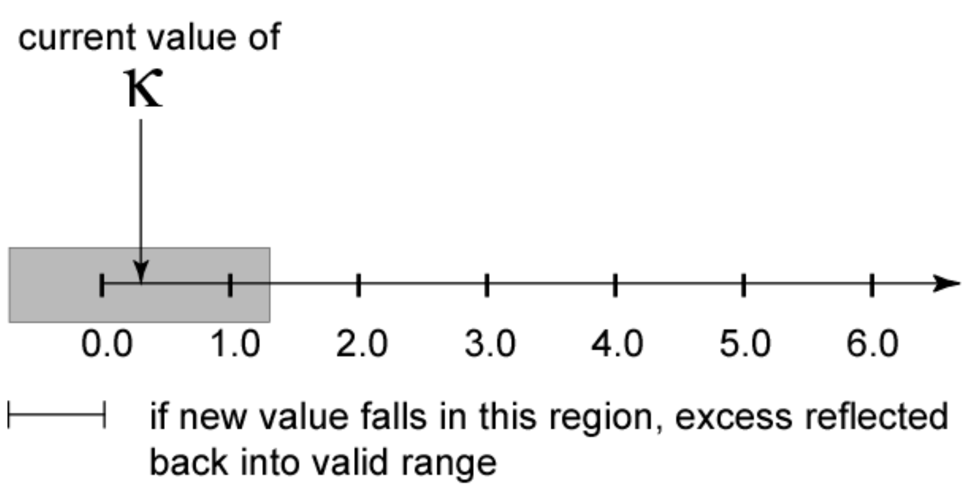
Step 1:
Pick 3 contiguous edges randomly, defining two subtrees, X and Y

Step 2:
Shrink or grow selected 3-edge segment by a random amount

Step 3:
Choose X or Y randomly, then reposition randomly

Moving through parameter space





Using κ (ratio of the transition rate to the transversion rate) as an example of a model parameter.

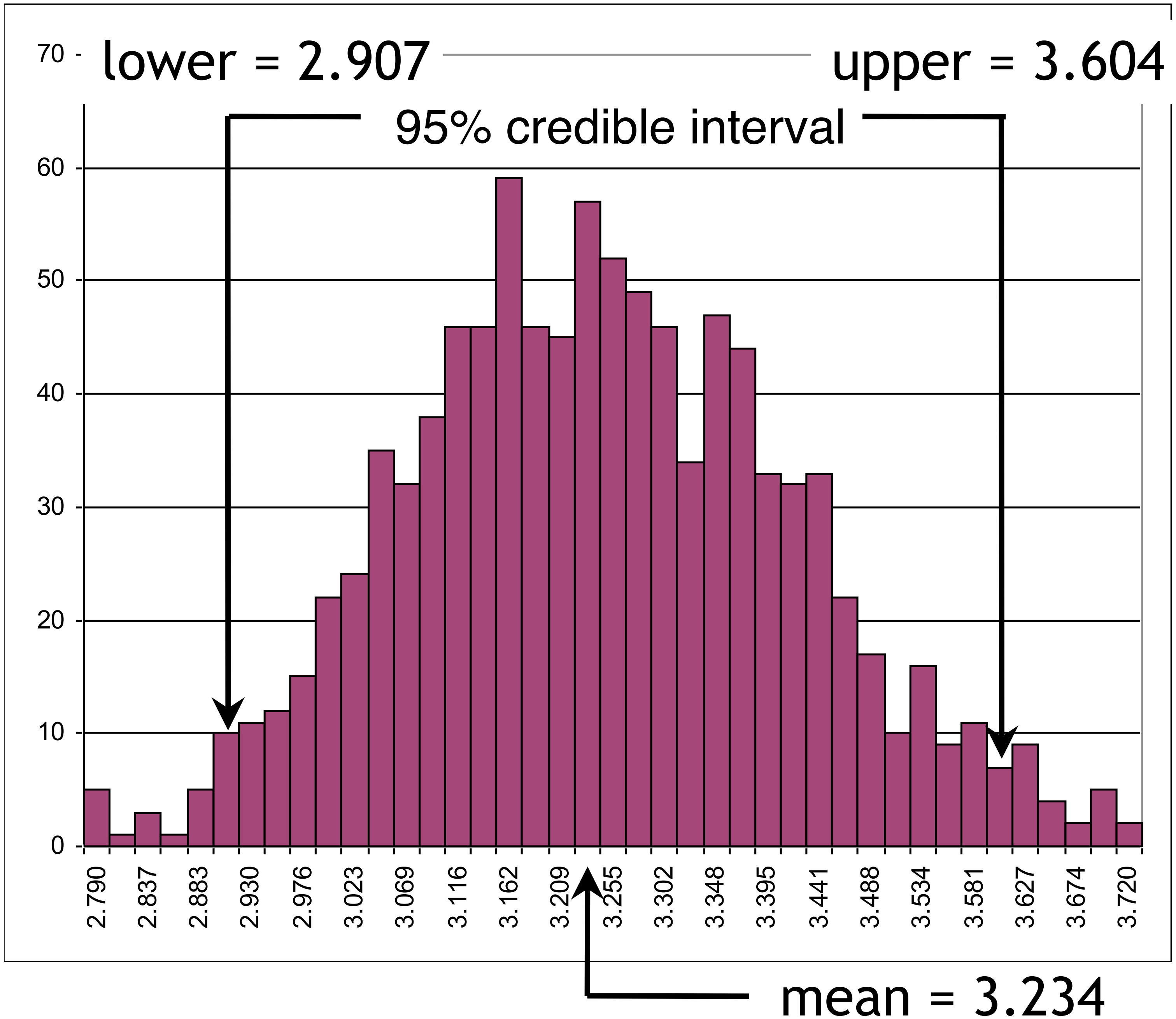
Proposal distribution is the uniform distribution on the interval (κ-δ, κ+δ)

The "step size" of the MCMC robot is defined by δ: a larger δ means that the robot will attempt to make larger jumps on average.

Putting it all together

- **Start with** random tree and arbitrary initial values for branch lengths and model parameters
- **Each generation** consists of one of these (chosen at random):
  - Propose a **new tree** (e.g. Larget-Simon move) and either accept or reject the move
  - Propose (and either accept or reject) a **new model parameter value**
- Every *k* generations, save tree topology, branch lengths and all model parameters (i.e. **sample the chain**)
- After *n* generations, **summarize sample** using histograms, means, credible intervals, etc.

Marginal Posterior Distribution of κ



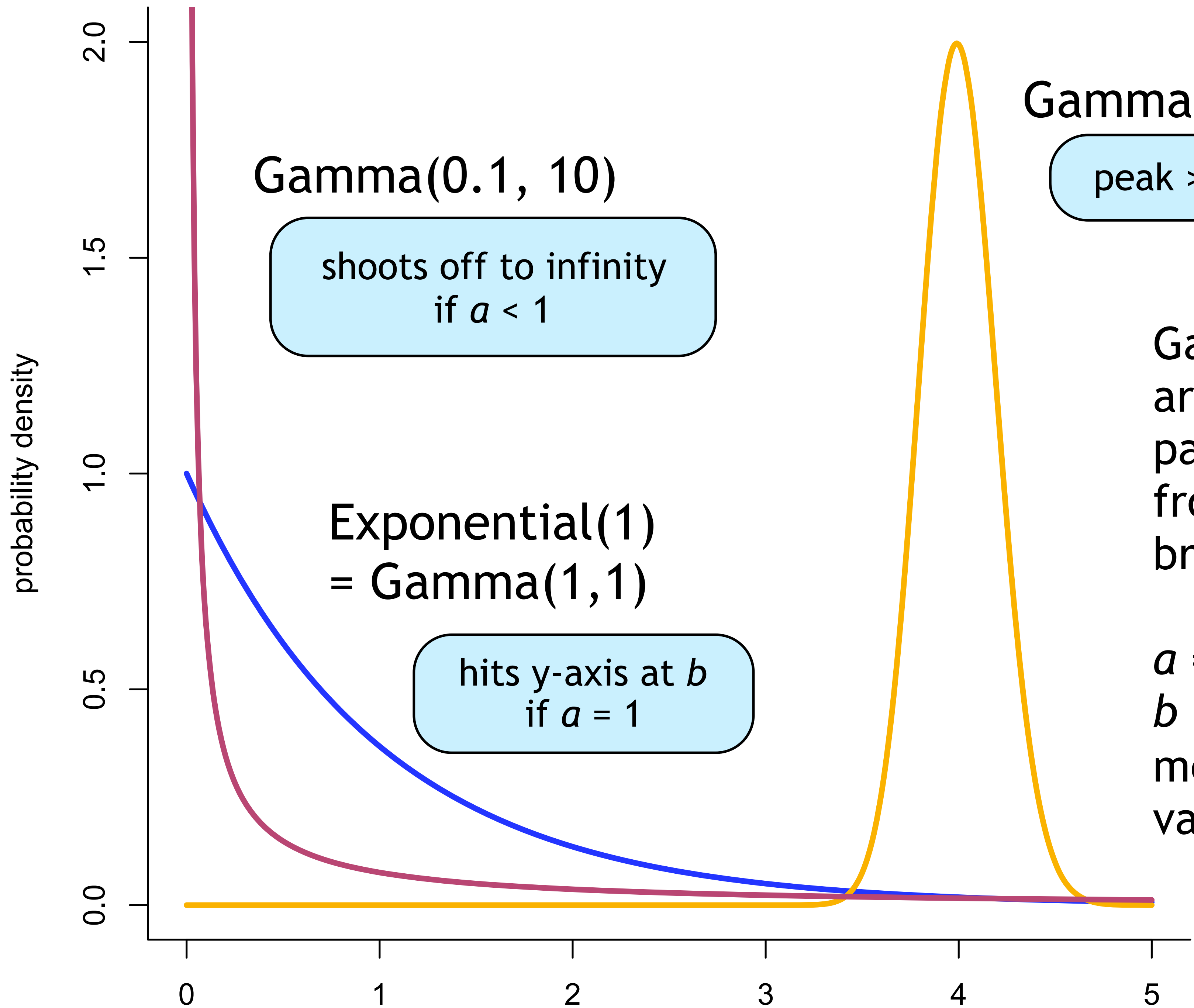
Histogram created from a sample of 1000 kappa values.

Data from Lewis, L., and Flechtner, V. 2002. Taxon 51: 443-451.

Common Priors

- **Discrete uniform** for topologies
  - exceptions becoming more common
- **Beta** for proportions (http://eurekastatistics.com/beta-distribution-pdf-grapher/)
- **Gamma** or **Log-normal** for branch lengths and other parameters with support $[0,\infty)$
  - Exponential is common special case of the gamma distribution
- **Dirichlet** for state frequencies and GTR relative rates

Gamma(a,b) distributions



Gamma distributions are ideal for parameters that range from 0 to infinity (e.g. branch lengths)

a = shape
b = scale
mean* = ab
variance* = ab^2

*Note: be aware that in many papers the Gamma distribution is defined such that the second (scale) parameter is the *inverse* of the value b used in this slide! In this case, the mean and variance would be a/b and a/b^2, respectively.

Beta(α,β) distribution

http://eurekastatistics.com/beta-distribution-pdf-grapher/

Dirichlet(a,b,c,d) distribution

https://phylogeny.uconn.edu/dirichlet-prior/


KONEC
přepravního
placeného
prostoru
THE END
of compulsory
ticket area