Marginal likelihood estimation

In ML model selection we judge models by their ML score and the number of parameters. In Bayesian context we:

- Use model averaging if we can "jump" between models (reversible jump methods, Dirichlet Process Prior, Bayesian Stochastic Search Variable Selection),
- Compare models on the basis of their marginal likelihood.

The Bayes Factor between two models:

$$B_{10} = \frac{\mathbb{P}(D|M_1)}{\mathbb{P}(D|M_0)}$$

is a form of likelihood ratio.

Bayes factor:

$$B_{10} = \frac{\mathbb{P}(D|M_1)}{\mathbb{P}(D|M_0)}$$

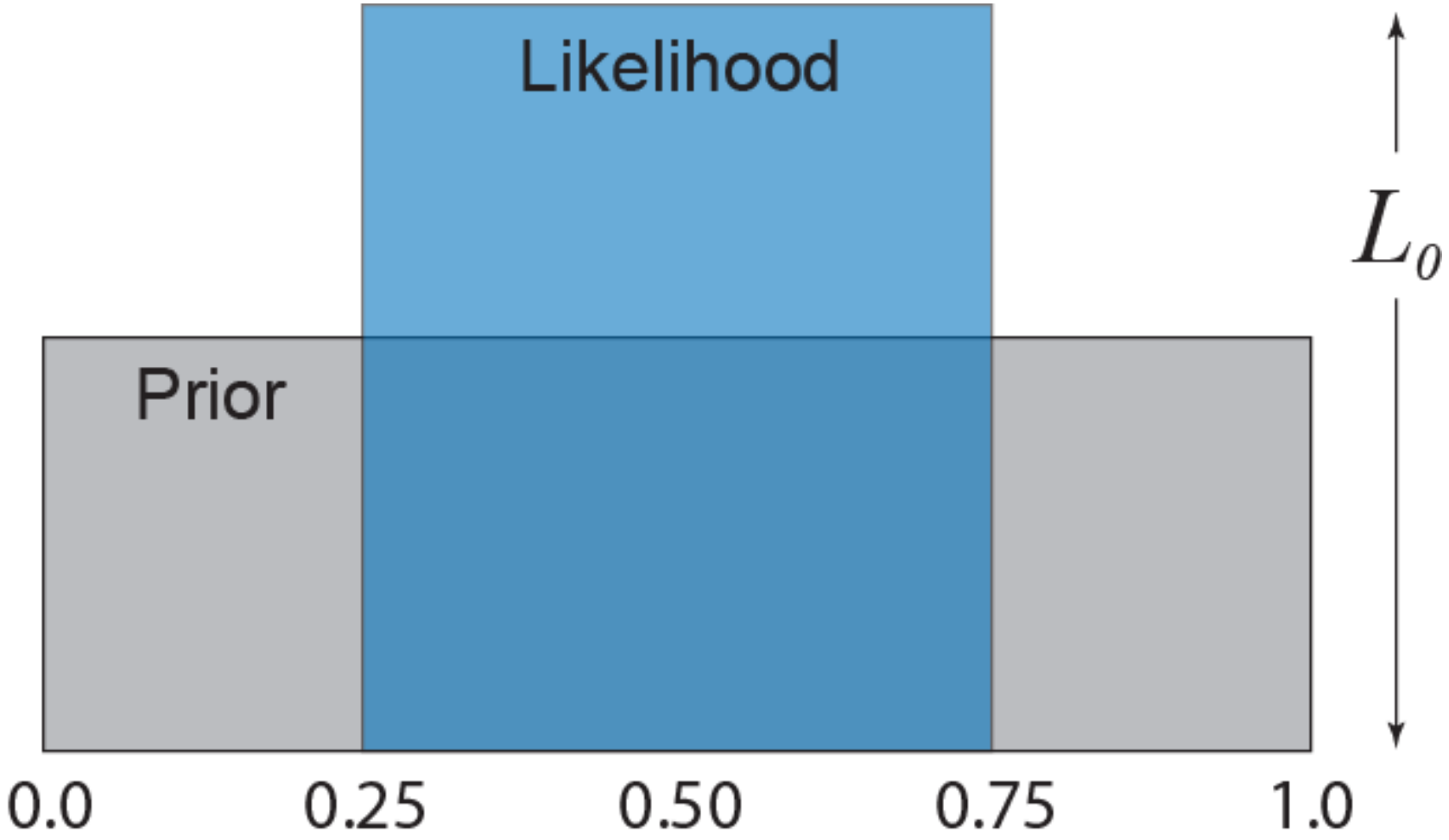
$\mathbb{P}(D|M_1)$ is the marginal probability of the data under the model, M_1:

$$\mathbb{P}(D|M_1) = \int \mathbb{P}(D|\theta, M_1)\mathbb{P}(\theta)d\theta$$

where θ is the set of parameters in the model.

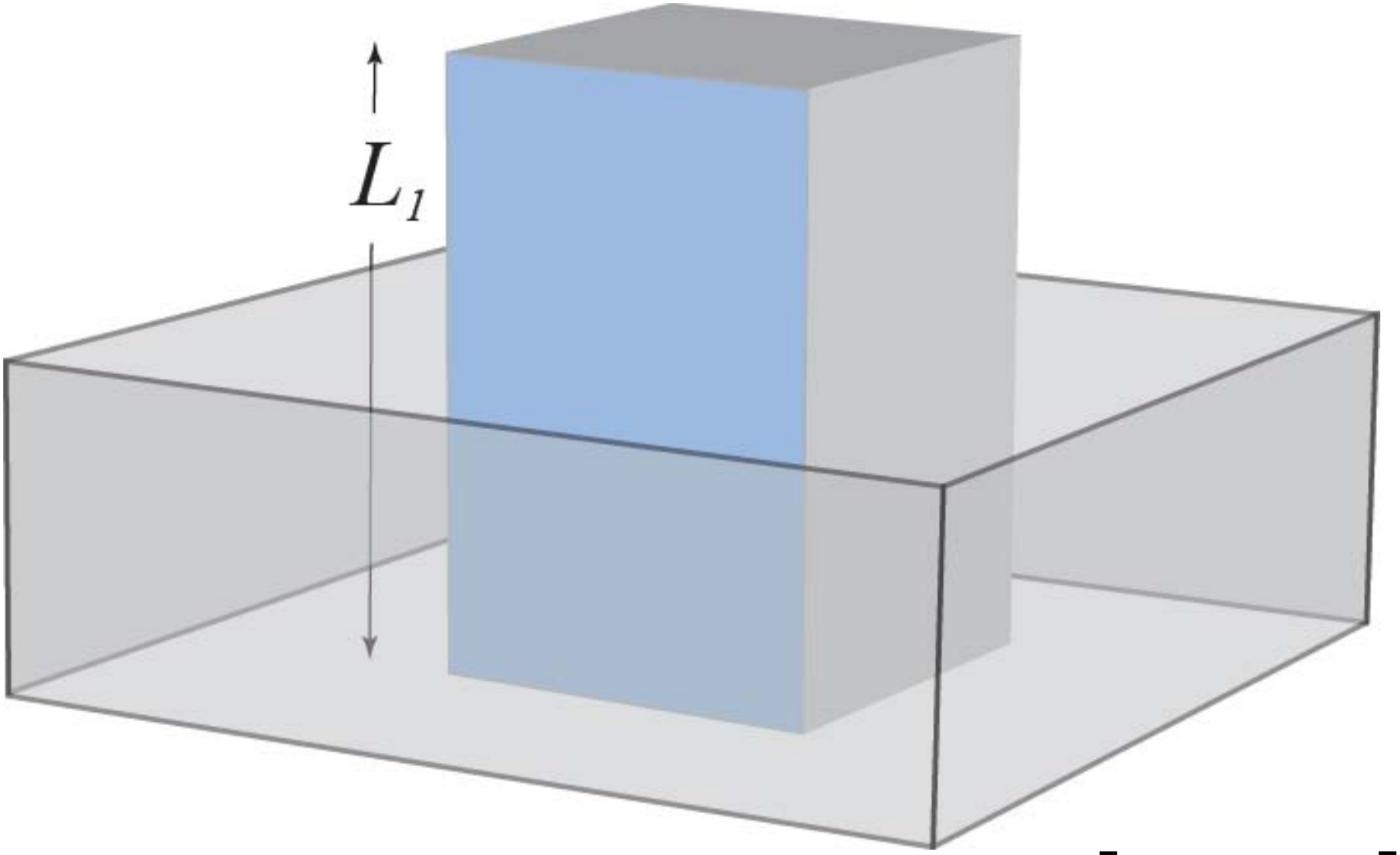
(The next slides are from Paul Lewis)

Marginal likelihood (1-param. model)



Average likelihood = $\left(\frac{1}{2}\right) L_0 + \left(\frac{1}{2}\right) (0)$

Marginal likelihood (2-param. model)

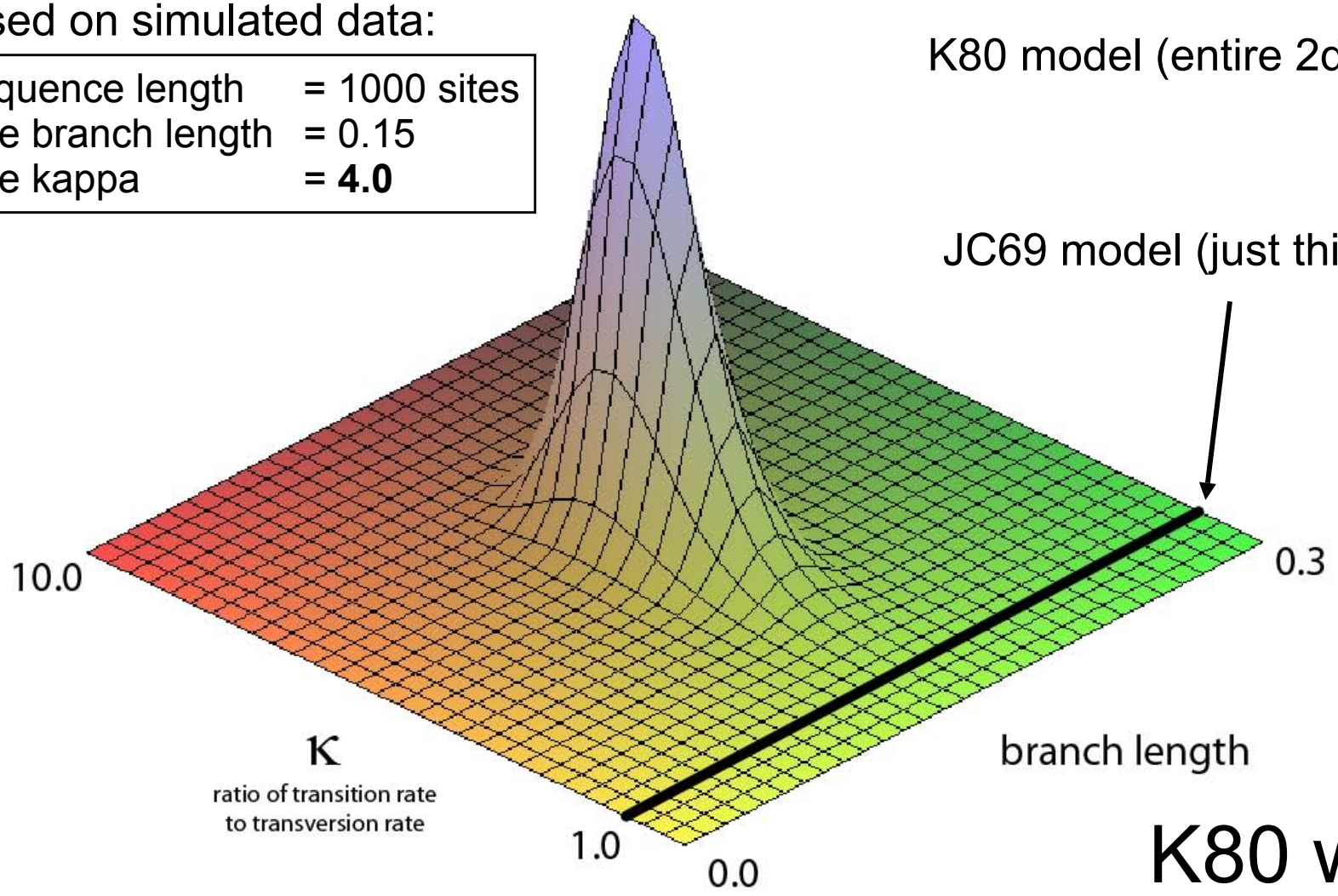


Average likelihood = $\left(\frac{1}{2}\right)^2 L_1 + \left[1 - \left(\frac{1}{2}\right)^2\right](0)$

Likelihood Surface when K80 true

Based on simulated data:

sequence length	= 1000 sites
true branch length	= 0.15
true kappa	= **4.0**

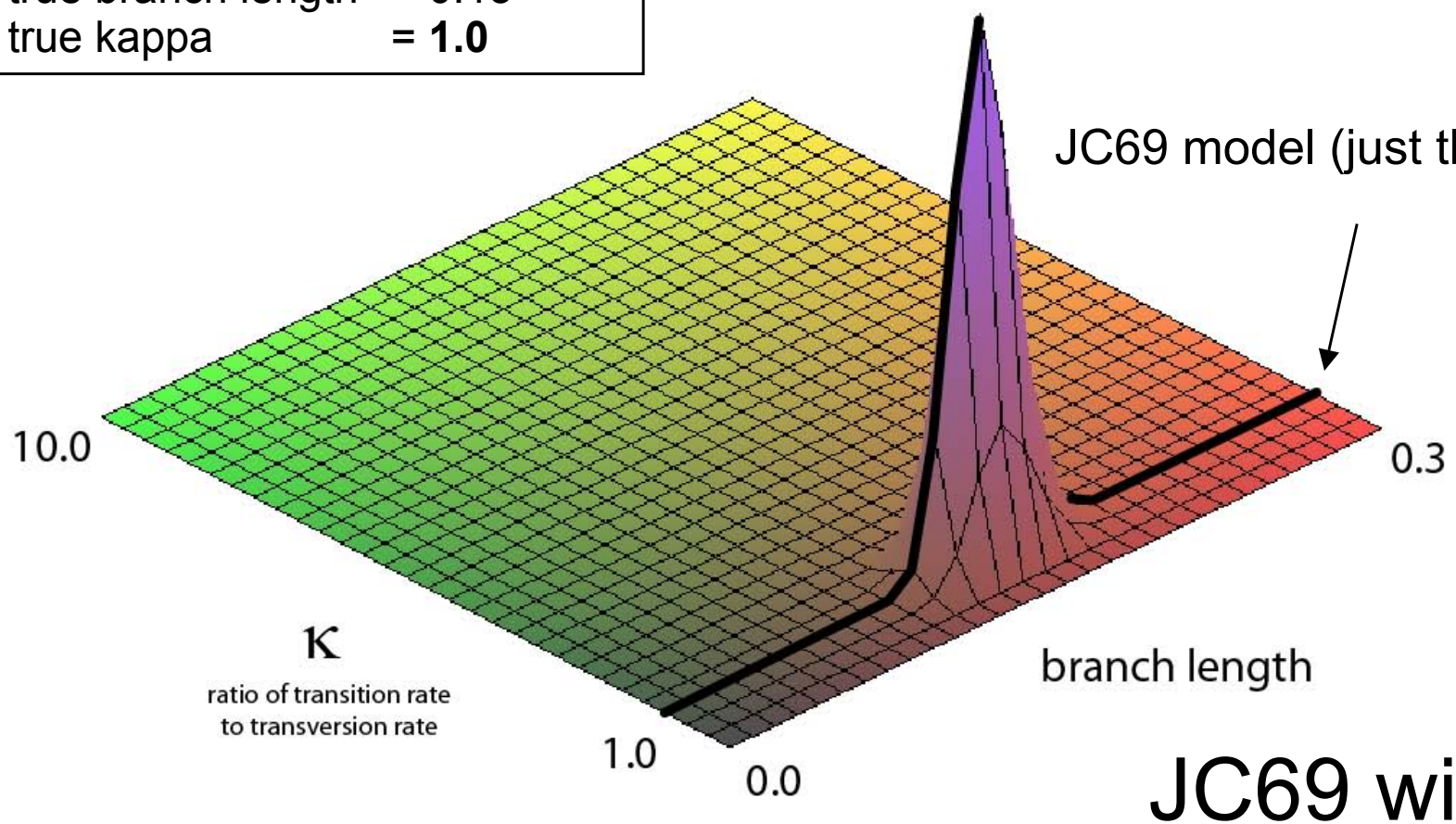


K80 wins

Likelihood Surface when JC true

Based on simulated data:

sequence length	= 1000 sites
true branch length	= 0.15
true kappa	= **1.0**



JC69 wins

Important point: Bayes Factor comparison remove the effect of the prior on the model itself, but the priors on nuisance parameters still matter!

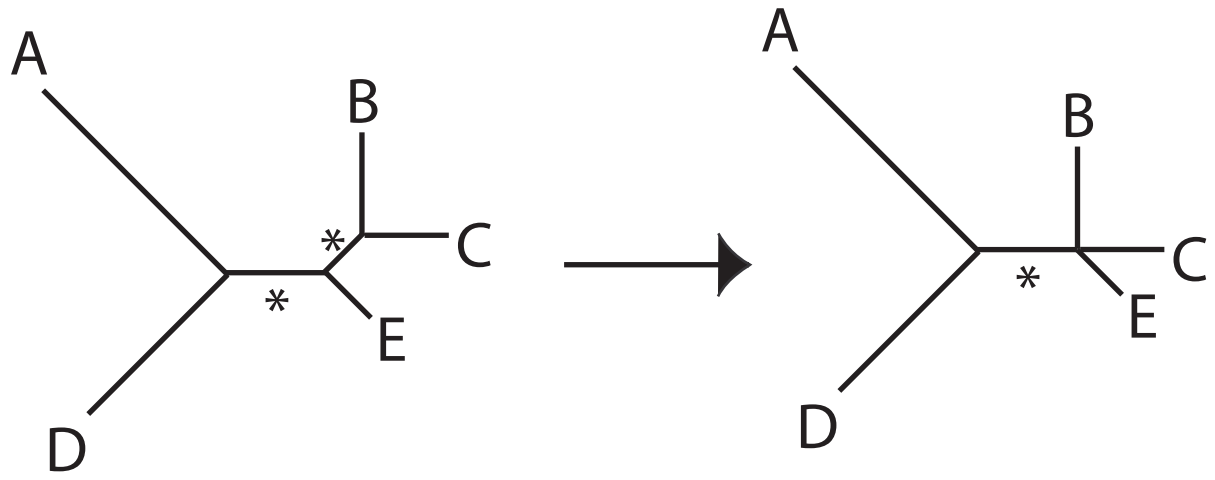
Think about your priors - using a very parameter-rich model may not be overparameterized if you have prior knowledge about the parameter values.

It is tricky to estimate $\mathbb{P}(\text{Data})$, there are "black-box" techniques (such as using the harmonic mean of the likelihoods sampled during MCMC), but they are quite unreliable.

Ideally, you can construct an MCMC sampler that "walks" over different models then you can use MCMC to estimate a posterior probability of models. Or you can conduct parameter inference that averages over models. Some common techniques for this are:

- reversible jump methods,
- use of a Dirichlet Process Prior to partition groups of data into subsets which share a homogeneous process,
- Bayesian Stochastic Search Variable Selection

Delete-Edge Move



There would have to be a "reverse" Add-Edge move

Homework

Questions?

Richer models

What if we think:

- There is a threshold level of N, P, and soil moisture required to set seed,
- N and P from fertilizer can run off if there is a lot of rain,
- decomposition of leaf litter returns N and P to the soil in 3 - 4 years.

How can we model this?

Likelihood-based inference when we cannot calculate a likelihood

The likelihood of parameter point θ is $\mathbb{P}(X|\theta)$, where $X =$ data

We can:

- calculate $\mathbb{P}(X|\theta)$ using rules of probability,
- approximate $\mathbb{P}(X|\theta)$ by simulating lots of data sets, Y_i. Then count the fraction of simulations for which $Y_i = X$

$$\mathbb{P}(X|\theta) \approx \frac{\sum_{i=1}^{n} I(Y_i = X)}{n}$$

where $I(Y_i = X)$ is an indicator function that is 1 if $Y_i = X$ and 0 otherwise.

Approximate Bayesian Computation

Set S to be an empty list.
While the number of samples in S is small (below some threshold):

- Draw a set of parameter values, θ_j, from the prior, $\mathbb{P}(\theta)$,
- Simulate 1 dataset, Y_1, according to the parameter values, θ_j.
- If $Y_1 = X$, then add θ_j to S

S is then a sample of parameter values that approximate posterior.

There is no autocorrelation in this procedure!

Approximate Bayesian Computation - a variant

Downsides: Slower than analytical calculations, and if $\mathbb{P}(X|\theta)$ is small (and it usually is) then you'll need lots of replicates.

Set S to be an empty list.
While the number of samples in S is small (below some threshold):

- Draw a set of parameter values, θ_j, from the prior, $\mathbb{P}(\theta)$,
- Simulate n datasets, Y_i for $i \in \{1, 2, \ldots n\}$ according to the parameter values, θ_j.
- Add θ_j to S and associate it with a weight, $w_j \approx \frac{\sum_{i=1}^{n} I(Y_i = X)}{n}$

Do posterior calculations on weighted averages of the samples in S.

Approximate Bayesian Computation - another variant

Set S to be an empty list.
While the number of samples in S is small (below some threshold):

- Draw a set of parameter values, θ_j, from the prior, $\mathbb{P}(\theta)$,
- Simulate 1 dataset, Y_1, according to the parameter values, θ_j.
- Add θ_j to S if $||Y_1 - X|| < \epsilon$, where ϵ is a threshold distance.

Do posterior calculations on the samples in S.

Approximate Bayesian Computation - a fourth variant

Let $A(X)$ be a set of summary statistics calculated on X.

Set S to be an empty list.
While the number of samples in S is small (below some threshold):

- Draw a set of parameter values, θ_j, from the prior, $\mathbb{P}(\theta)$,
- Simulate 1 dataset, Y_1, according to the parameter values, θ_j.
- Add θ_j to S if $||A(Y_1) - A(X)|| < \epsilon$, where ϵ is a threshold distance.

Do posterior calculations on the samples in S.

Approximate Bayesian Computation - yet another variant

Let $A(X)$ be a set of summary statistics calculated on X.

Set S to be an empty list.
While the number of samples in S is small (below some threshold):

- Draw a set of parameter values, θ_j, from the prior, $\mathbb{P}(\theta)$,
- Simulate 1 dataset, Y_1, according to the parameter values, θ_j.
- Add θ_j to S with a weight, w_j, proportional to $w_j \approx ||A(Y_1) - A(X)||$.

Do posterior calculations on weighted averages of the samples in S.

If A is not a set of sufficient summary statistics, then you are throwing away information.

In general ABC let's you tackle more difficult problems: it is easier to simulate under a complicated problem than it is to do inference.

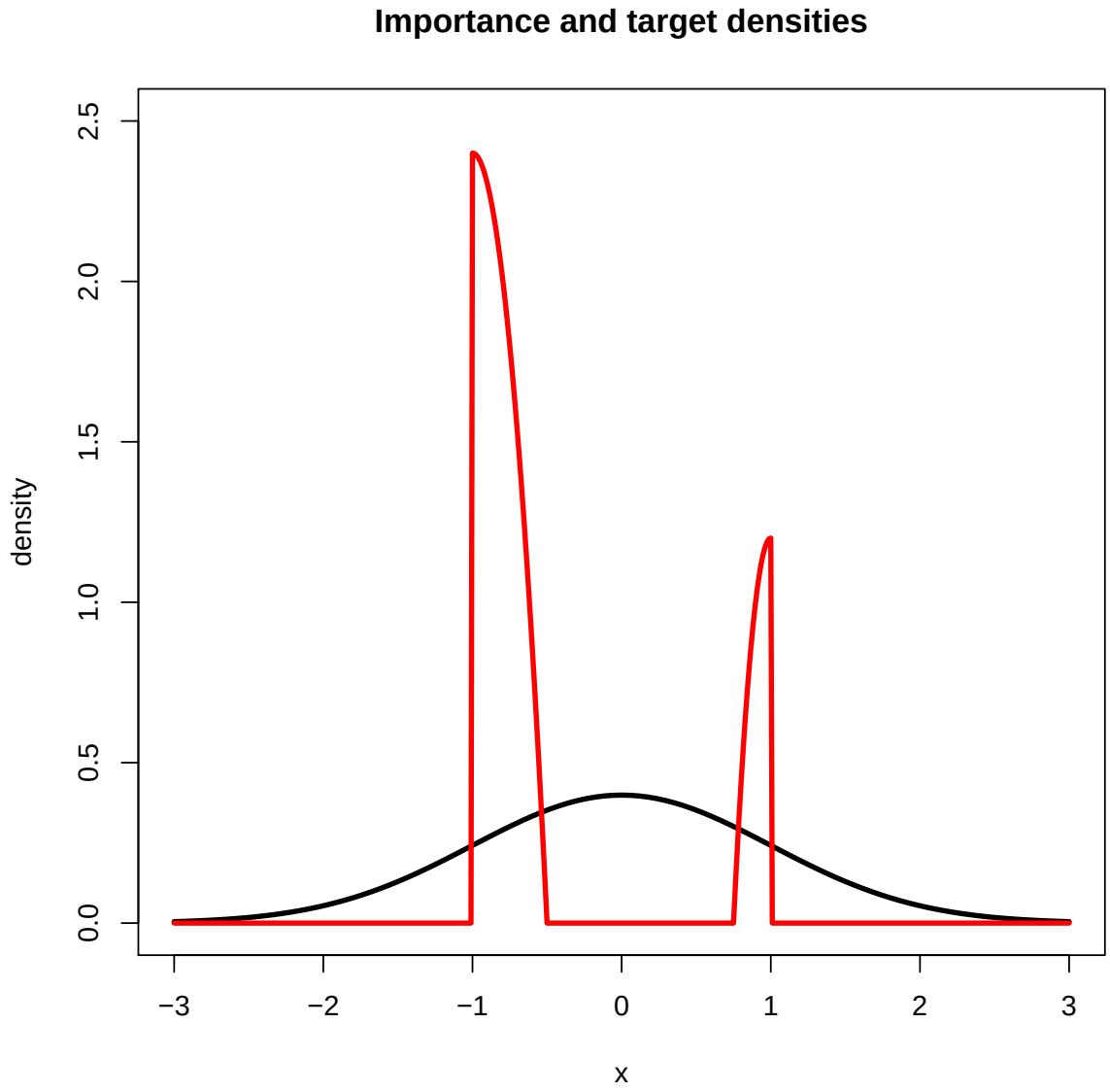
Usually a problem that can be tackled with ABC can be tackled by adding lots of latent variables. But it may not be practical.

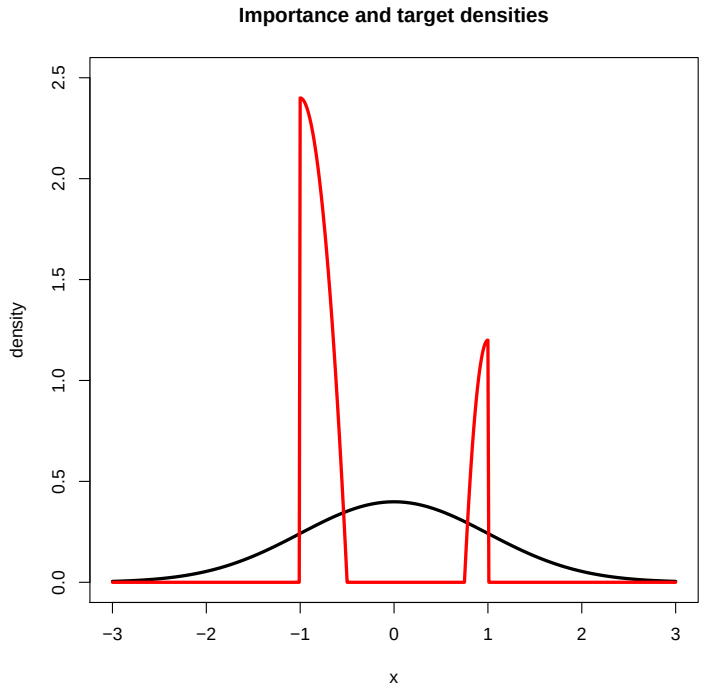
Alternative MCMC samplers

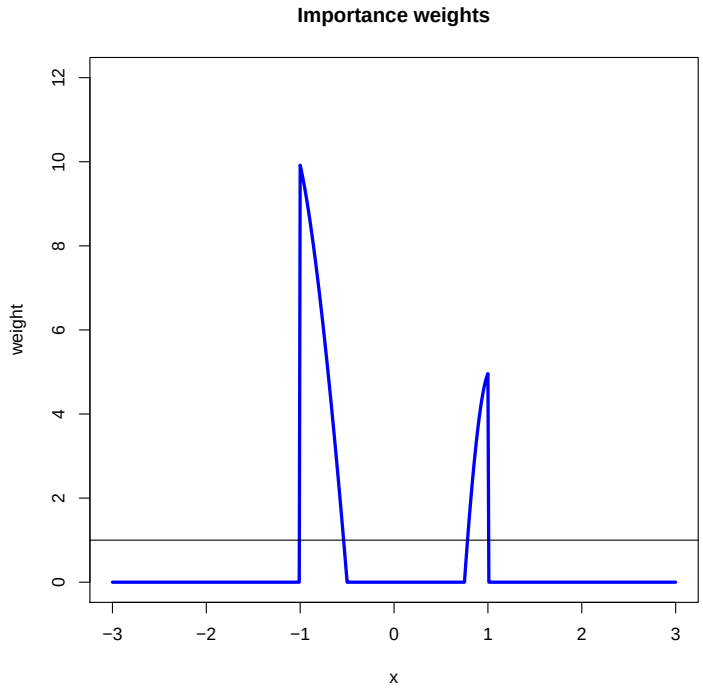
There are lots:

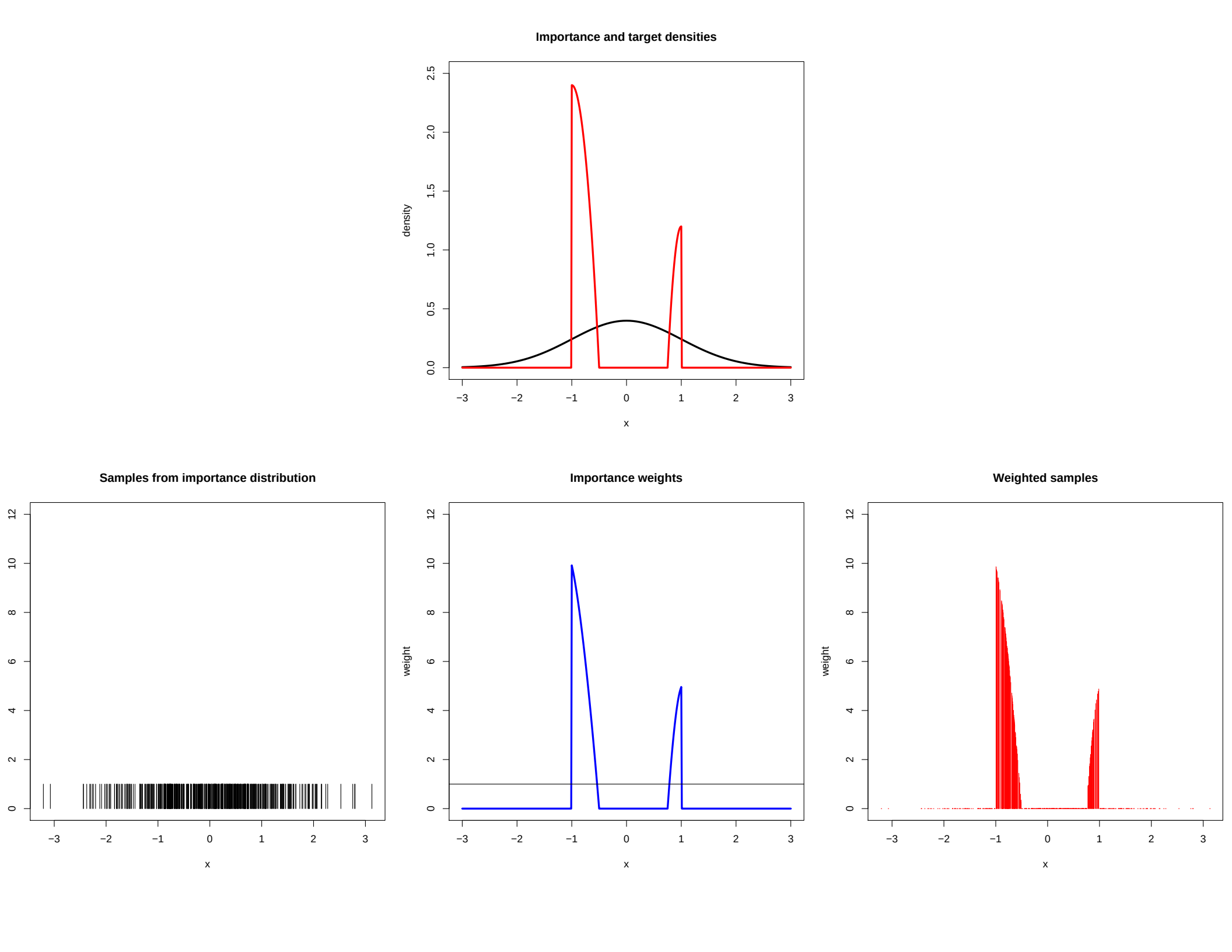
- Metropolis-Hastings with the proposed state being drawn from:
  - an arbitrary proposal distribution,
  - the prior,
  - the conditional posterior (Gibbs Sampling).
- adaptive rejection,
- slice sampling,
- Metropolis-coupled MCMC,
- delayed-rejection Metropolis-Hastings,
- SAMC,
- importance sampling,
- ...

Importance sampling: we simulate points from one distribution, and then reweight the points to transform them into samples from a target distribution that we are interested in:




Importance and target densities
density
0.0
0.5
1.0
1.5
2.0
2.5
−3
−2
−1
0
1
2
3
x


Importance weights
weight
0
2
4
6
8
10
12
−3
−2
−1
0
1
2
3
x


Importance and target densities
density
x
Samples from importance distribution
Importance weights
weight
Weighted samples

Importance sampling

The method works well if the importance distribution is:

- fairly similar to the target distribution, and
- not "too tight" to allow sampling the full range of the target distribution