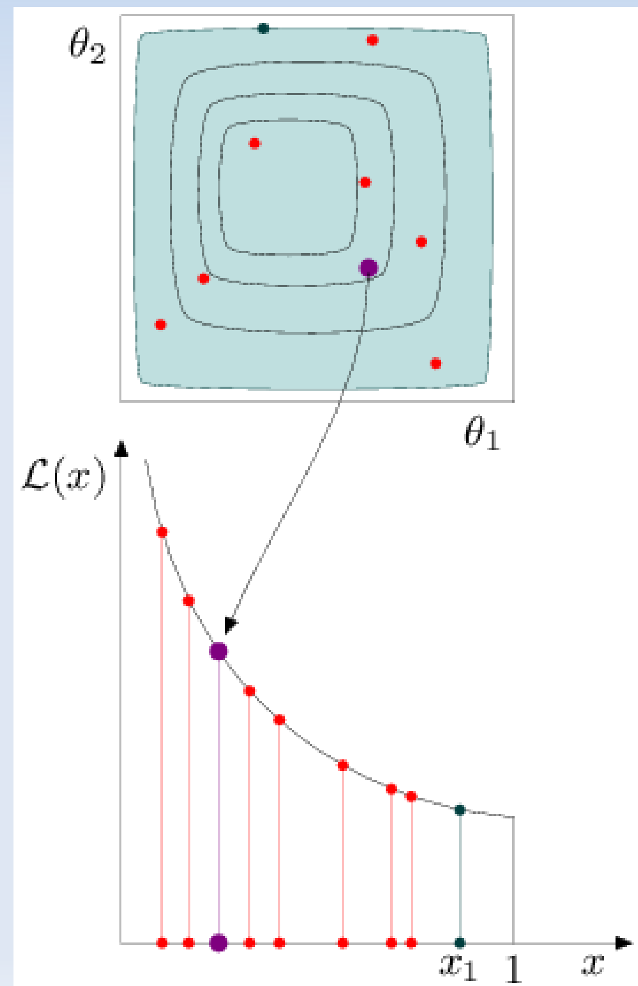
Nested Sampling

- New Monte Carlo method by John Skilling

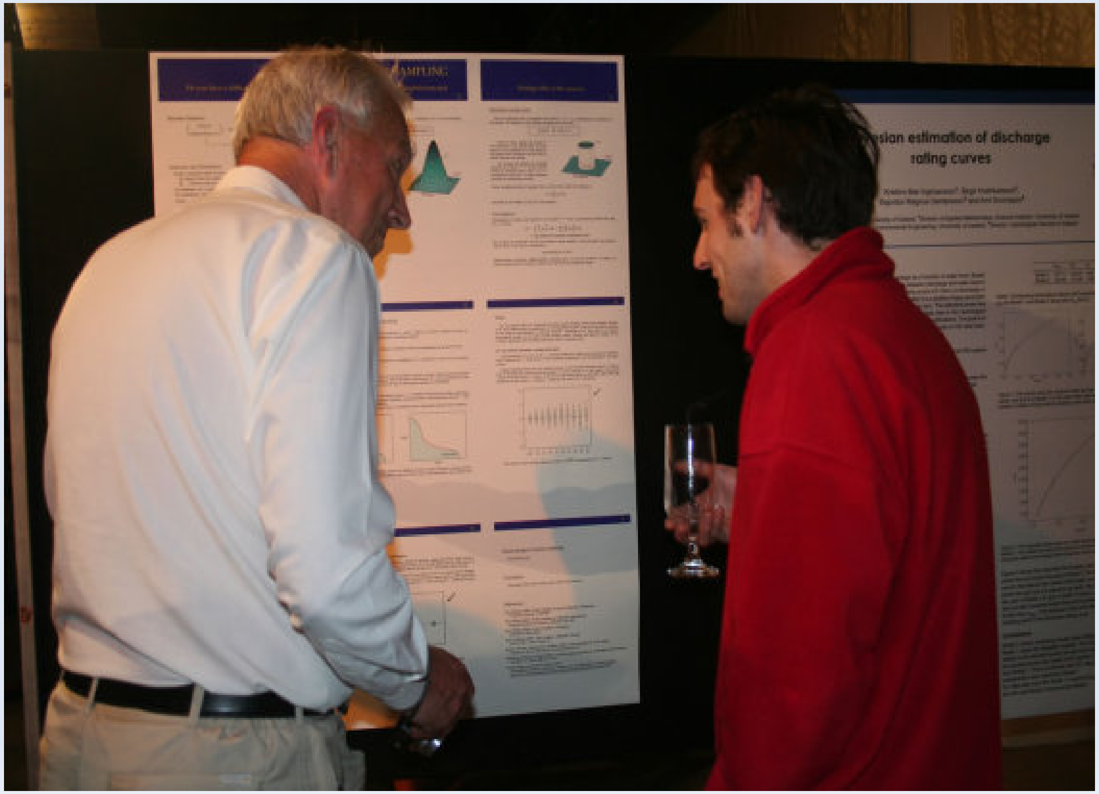


Advantages

- Very generally applicable
- Calculates the `evidence'/marginal likelihood for the model in addition to posterior samples. Most MCMC just gives posterior samples
- Can deal with difficult posterior distributions. Multimodal, strongly correlated, etc.
- Requires very little manual tuning. e.g. No need to decide on temperature levels and things like that.

Disadvantages

- Sometimes slower than some other methods.
- Implementation often requires some approximations that compromise the neatness of the main idea. More on this later.



Zooming In

Model Selection

- Within model M_1, with parameters Θ, parameter estimation:

$$p(\theta|D,M_1) = \frac{p(\theta|M_1)p(D|\theta,M_1)}{p(D|M_1)}$$

$$= \frac{p(\theta|M_1)p(D|\theta,M_1)}{\int p(\theta|M_1)p(D|\theta,M_1)d\theta}$$

- The denominator, being independent of Θ, is irrelevant for parameter estimation, but crucial for testing M_1 against an alternative!

How to Calculate Z?

- Common notation:

$$Z = \int \pi(\theta) L(\theta) d\theta$$

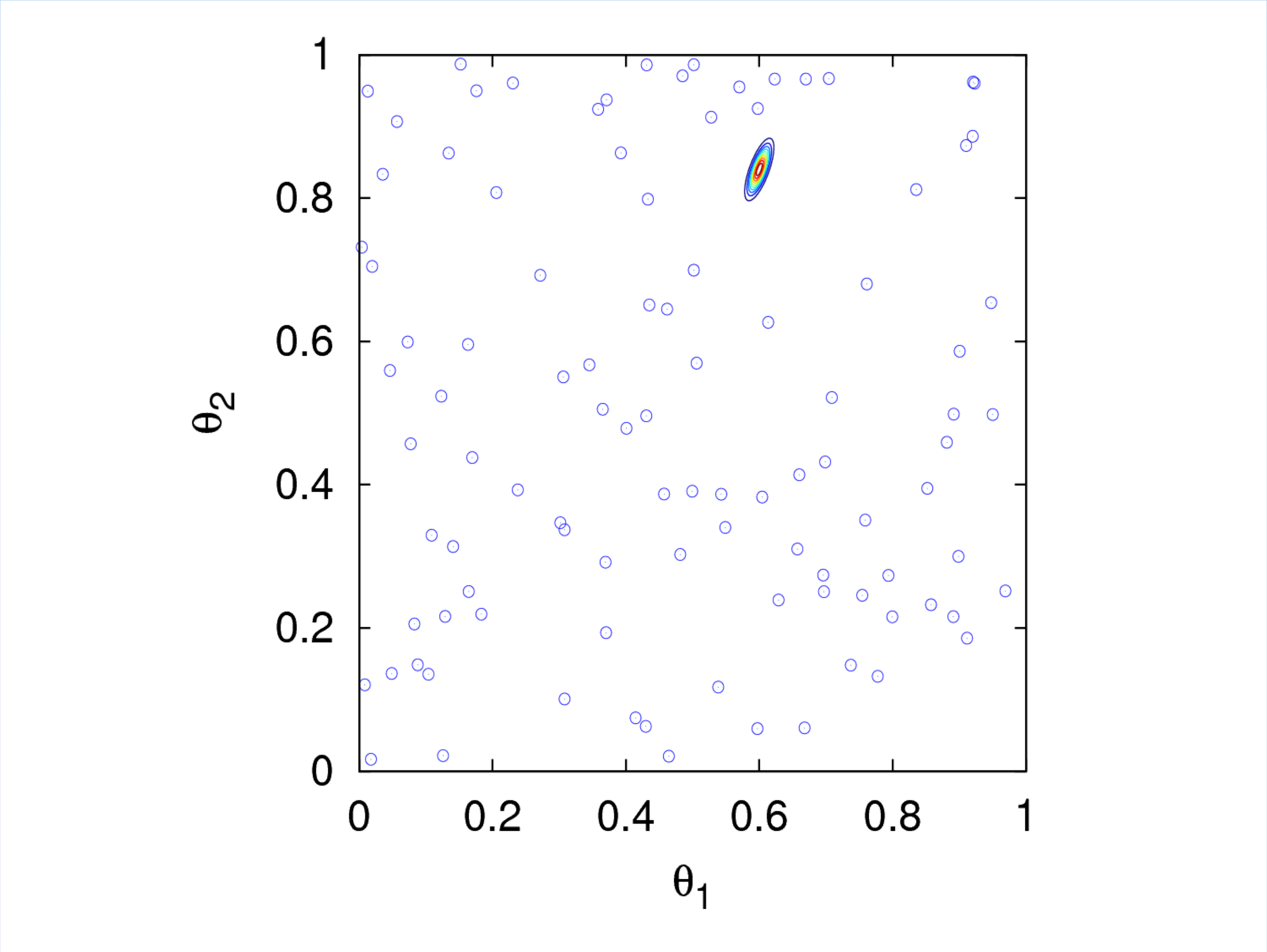
- Looks like we can sample from the prior and calculate the average likelihood

$$= \langle L(\theta) \rangle_{\pi}$$

- Works in principle but can be extremely inefficient to the point of practical impossibility

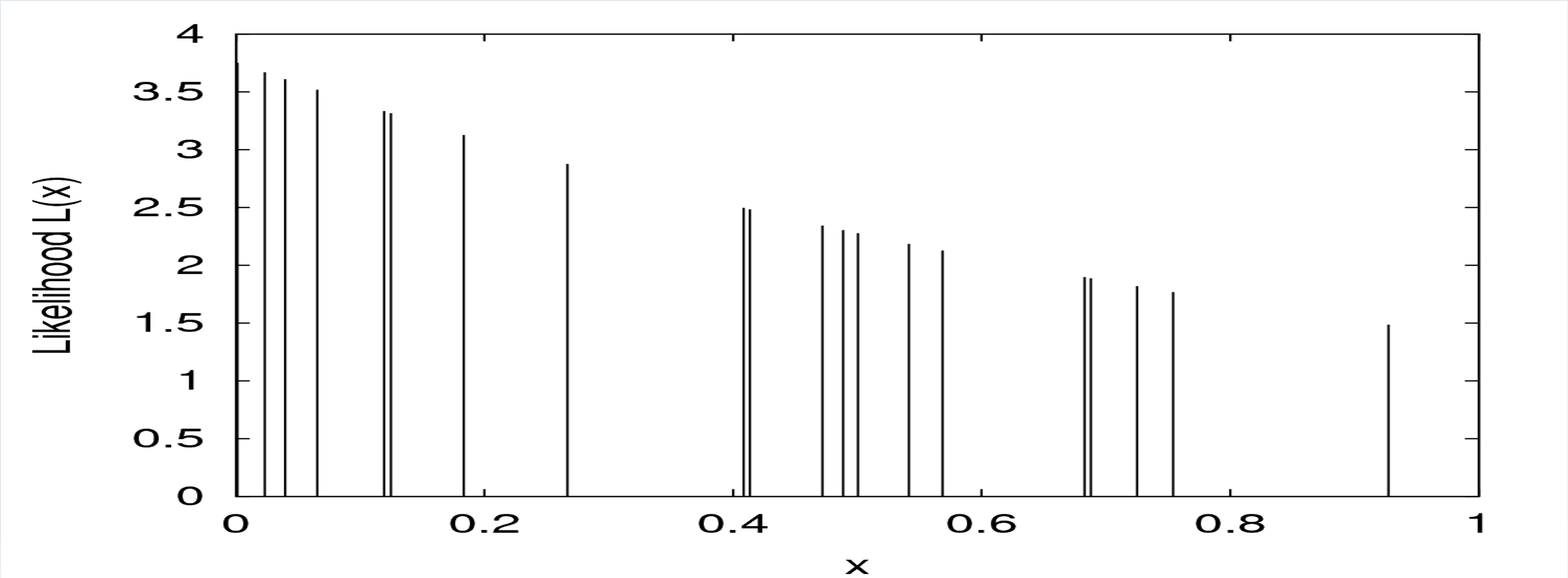
Especially in High Dimensions

- The high likelihood region, which dominates the Z integral, can be extremely small, occupying, for example, 10^{-30} of the prior probability.



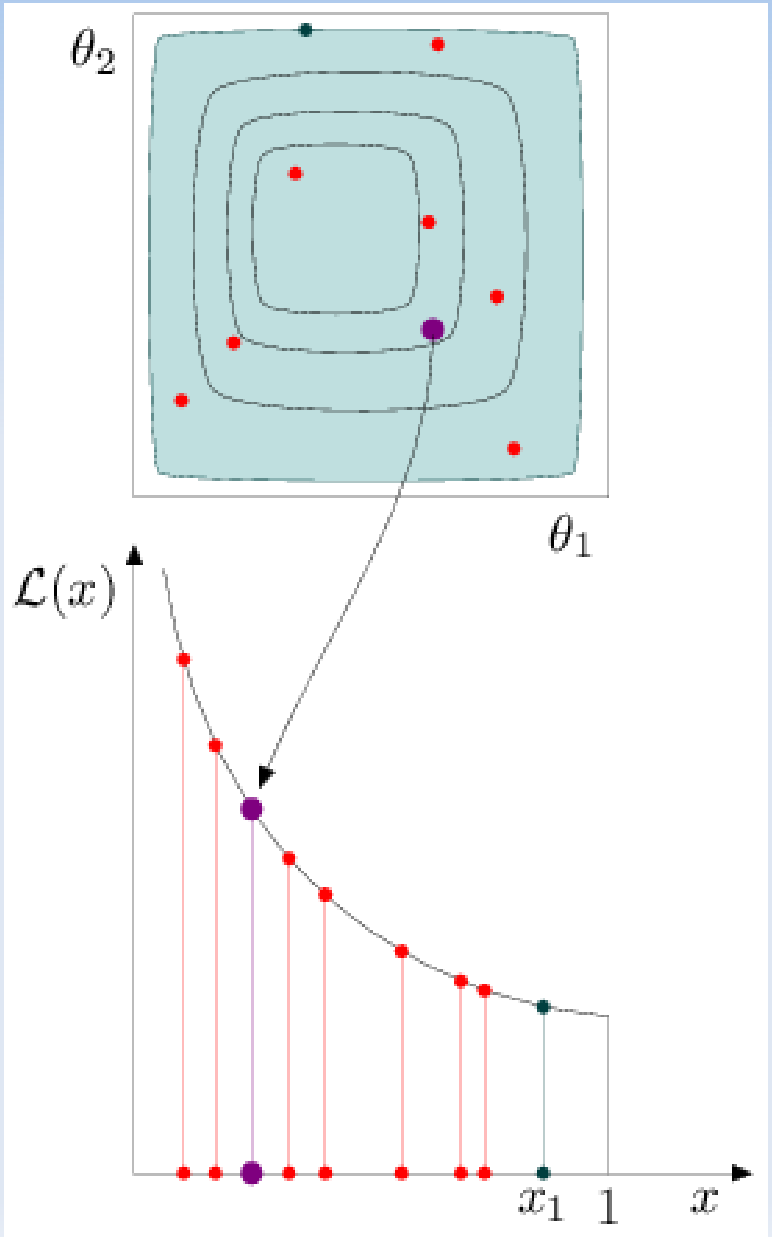
We could do it in 1 dimension...

- If the prior was x ~ Uniform(0,1) and we obtained some points (x, L) we could just use the trapezoidal rule or something
- Posterior: weight=likelihood x 'width'



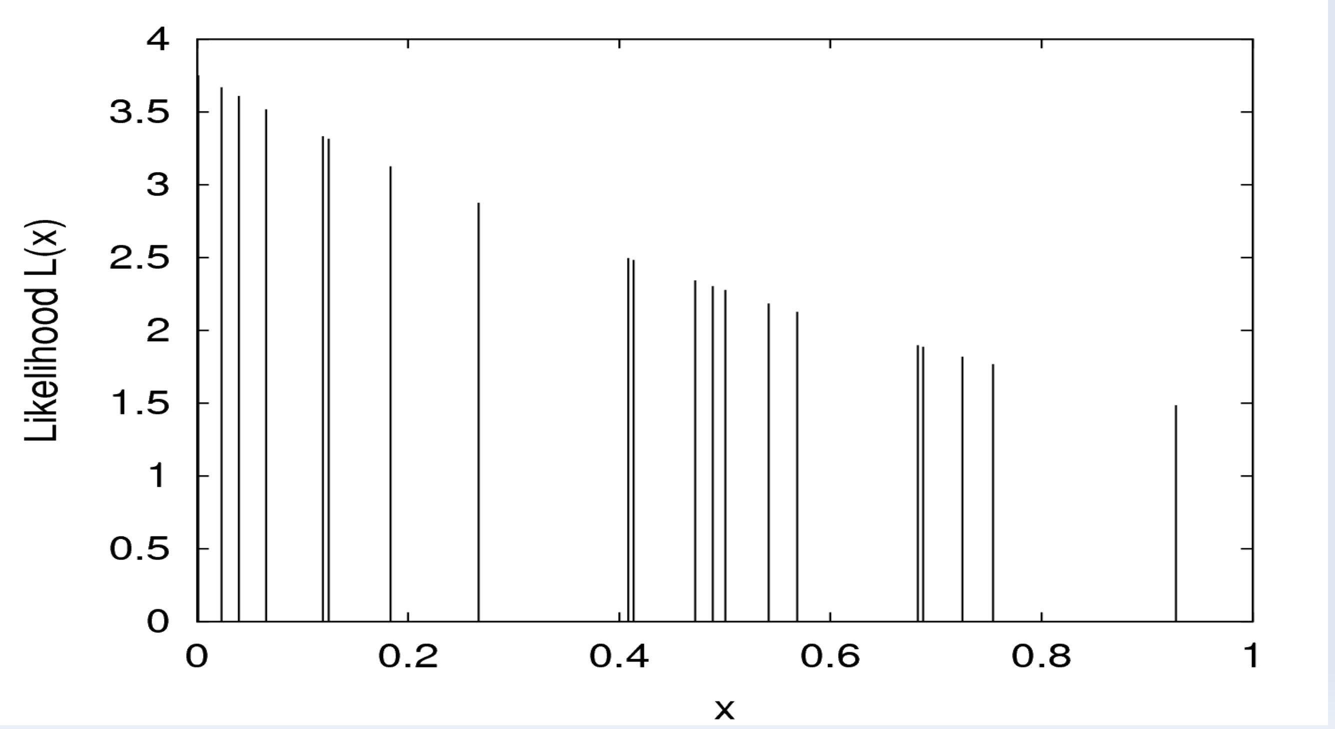
Nested Sampling Idea

- Imagine sorting all points in parameter space by likelihood
- Then map the parameter space onto [0,1] in order of decreasing likelihood
- If we could get some points then we can do the integral numerically, easily!
- Need more points with small x than large x



Let's Get Some Points!

- The hard thing is getting the x value. i.e. Here's a point, what fraction of the prior probability lies at a higher likelihood?
- But we can *estimate* the x's.

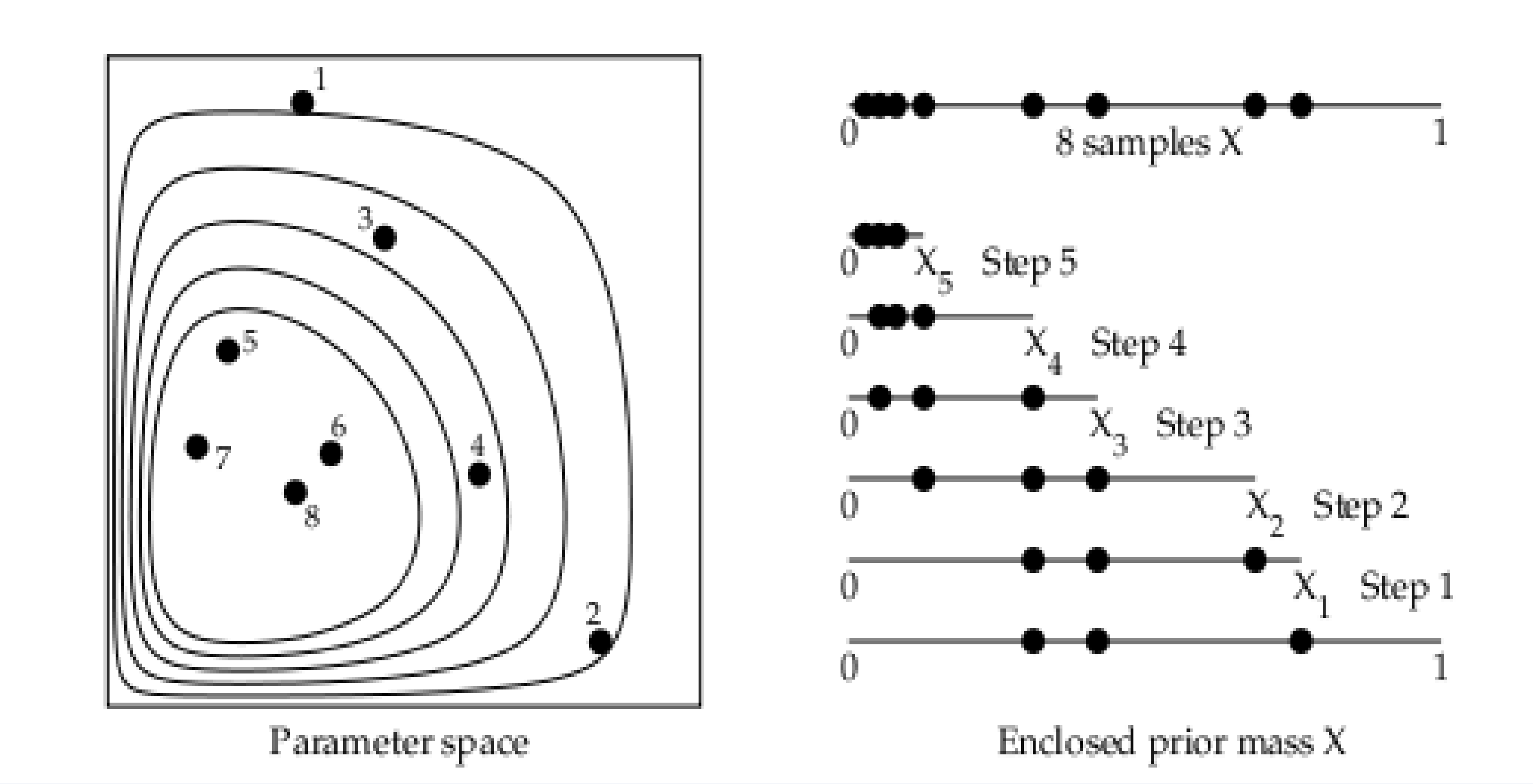


The Algorithm

- Generate n points from the prior
- Loop for i=1,2,3,...
- {
- Find the one with the lowest likelihood, L_{worst}. Remove it from the population, but store it for results. Estimate its x value as $(n/(n+1))^i$.
- Replace that point with a new one generated from the prior, but subject to the hard constraint $L>L_{worst}$.
- }

Easy

- Slight modification necessary in case of a tie for the worst point. I won't go into detail about this here.



- Thus we get a set of points (the discarded ones), their corresponding estimated x value and their likelihood.
- x decreases *exponentially* as the algorithm proceeds
- Just do the integral with the trapezoidal rule to get Z
- You get posterior samples by weighting each point according to its likelihood*width, where width is determined from its distance from a neighbour.

The x's aren't exact though, they're estimates

- That's okay, we know how the x's are distributed
- Jiggle around within that probability distribution. Can even get uncertainty on Z (optimistic though, doesn't take into account numerical integration error)

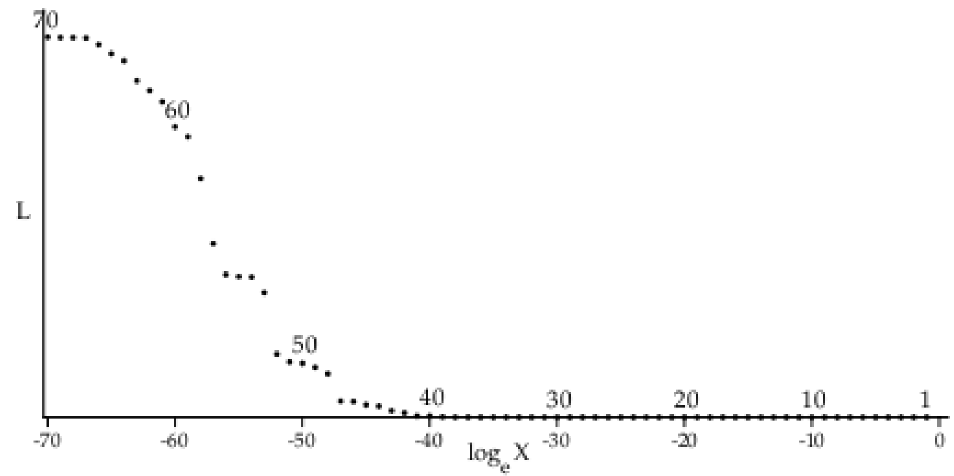


Figure 7: The same sequence of nested sampling points as in Fig. 6, shown at the crude central estimate of their X values (*i.e.* uniformly in $\log X$).

deviation.

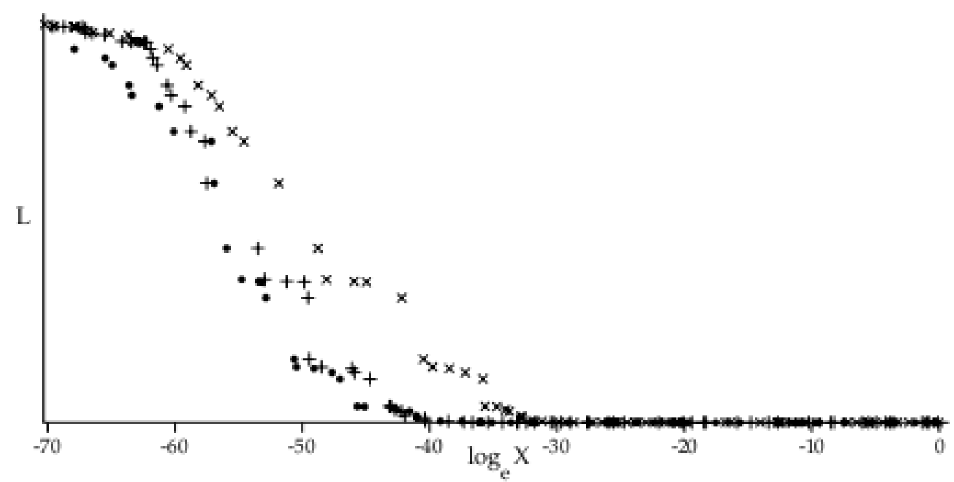


Figure 8: The same sequence of nested sampling points as in Fig. 6, showing three random assignments ($\bullet, +, \times$) for X values taken from $\Pr(X)$.

How to Generate That New Point

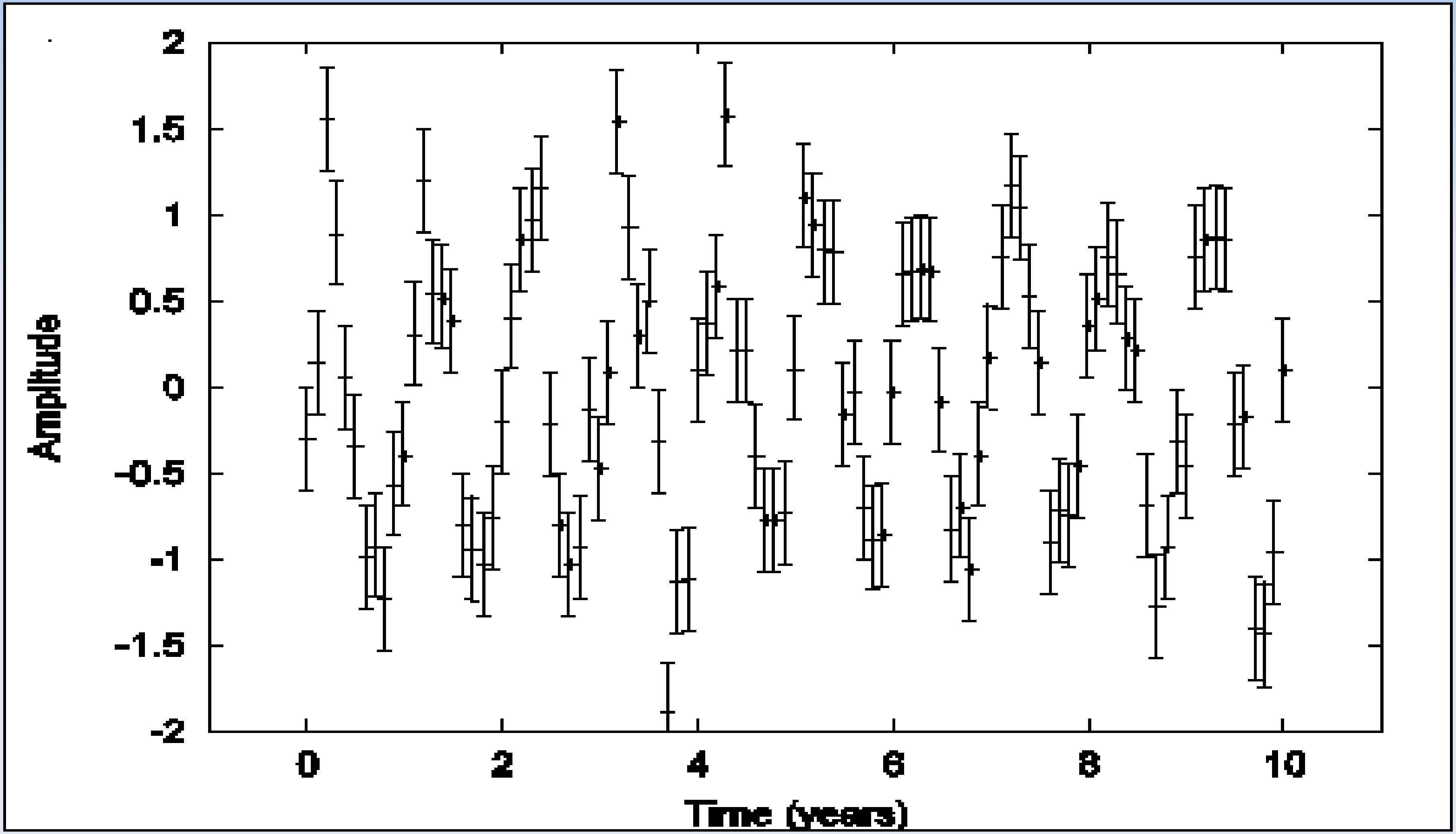
- This part of the algorithm was swept under the rug in the previous slide:
- Replace that point with a new one generated from the prior, but subject to the hard constraint $L > L_{worst}$.
- Generally quite hard to do this
- An easy but approximate way: Copy a surviving point (guaranteed to have $L > L_{worst}$) and evolve it with MCMC according to the prior, but rejecting if $L \leq L_{worst}$.

- In my experience this approximation works better than it sounds
- Gets more accurate if you increase population size or number of MCMC steps taken
- Think about the distribution your generated point will be sampled from. Is it a good approximation to required one (prior but with a hard lower limit on likelihood)?

Nested Sampling Demo

- Question 3 from Assignment 1
- Had really annoying integrals (sorry)
- But Nested Sampling makes it easy!
- Same Matlab/Octave code used for Q4 of Assignment 2. So pay attention now ;-)

Is there a sine wave?



The Two Hypotheses

- H0, the data is just gaussian noise with sigma=0.3
- H1, the data is a sine wave plus noise. Frequency = 1 year and unknown amplitude A, with $p(A|H_1) \sim \text{gaussian}(0,\delta^2)$
- ``Evidence'' for $H_0 = p(D|H_0) = 3.81 \times 10^{-136}$.
- Evidence for H_1 - Let's do it with Nested Sampling.

Results

- Run the code and see!

References

- Sivia and Skilling, J. 2006. Data Analysis: A Bayesian Tutorial. **2nd Edition**.
- Skilling, 2006. Nested Sampling for General Bayesian Computation. Bayesian Analysis. 1, Number 4, 833-860.
- Murray, I. 2007. Advances in Markov chain Monte Carlo methods. PhD thesis