

# Gradients and Nested Sampling

The present state of the art

Will Handley  
[<wh260@cam.ac.uk>](mailto:wh260@cam.ac.uk)

Royal Society University Research Fellow & Turing Fellow  
Astrophysics Group, Cavendish Laboratory, University of Cambridge  
Kavli Institute for Cosmology, Cambridge  
Gonville & Caius College  
[willhandley.co.uk/talks](http://willhandley.co.uk/talks)

7<sup>th</sup> July 2023



**The  
Alan Turing  
Institute**



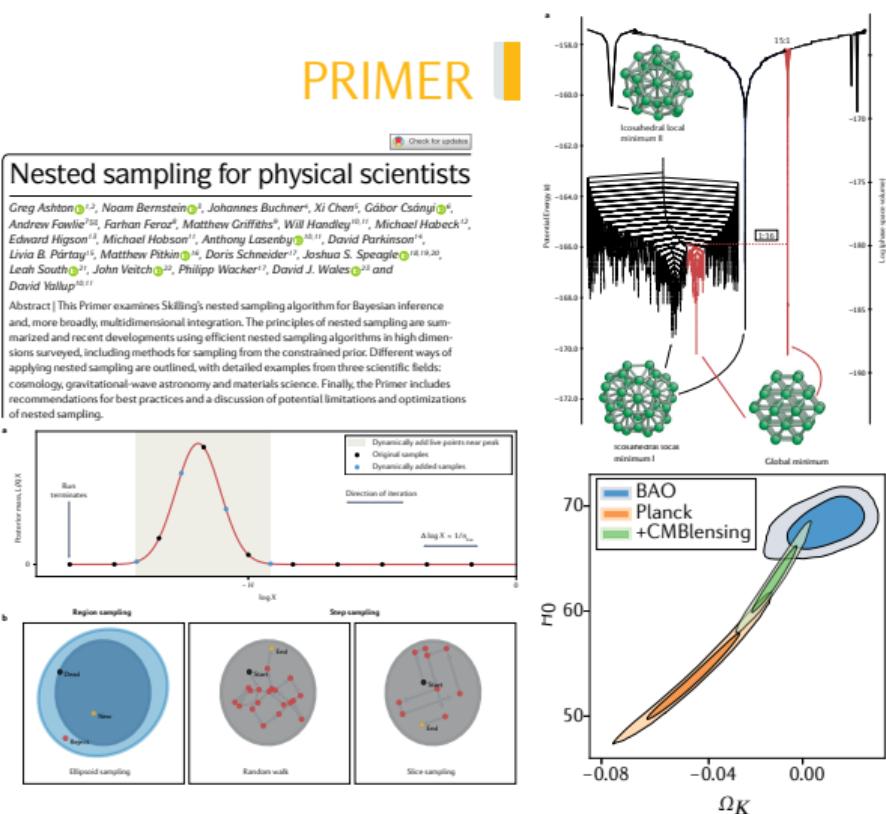
UNIVERSITY OF  
CAMBRIDGE



# Highlight: state-of-the-art Nature review primer [2205.15570]

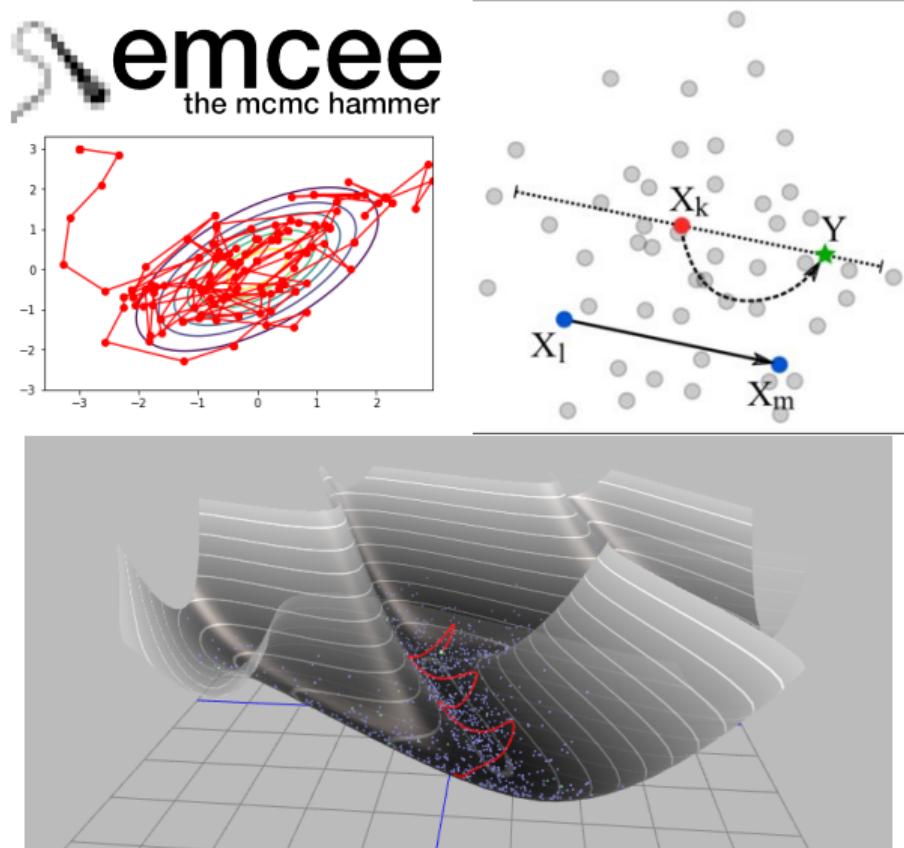
PRIMER

- ▶ Invented by John Skilling in 2004.
- ▶ Recent Nature review primer on nested sampling led by Andrew Fowlie and assembled by the community.
- ▶ Showcases the current set of tools, and applications from chemistry to cosmology.
- ▶ Buchner technical review [2101.09675]
- ▶ In this talk
  - ▶ What is nested sampling?
  - ▶ How can it use gradients?



# Where is Nested Sampling?

- ▶ For many purposes, in your Neural Net you should group Nested Sampling with (MCMC) techniques such as:
  - ▶ Metropolis-Hastings (PyMC, MontePython)
  - ▶ Hamiltonian Monte Carlo (Stan, blackjax)
  - ▶ Ensemble sampling (emcee, zeus).
  - ▶ Variational Inference (Pyro)
  - ▶ Sequential Monte Carlo
- ▶ Thermodynamic integration
- ▶ Genetic algorithms

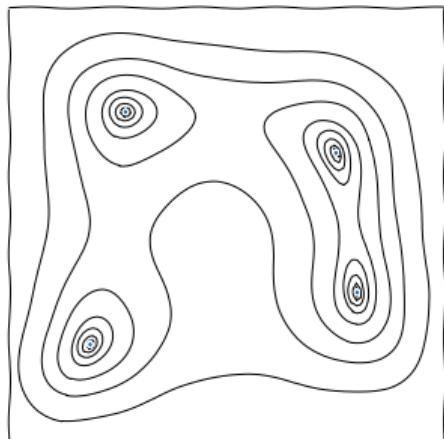


# What is Nested Sampling?

- ▶ Nested sampling is a radical, multi-purpose numerical tool.
- ▶ Given a (scalar) function  $f$  with a vector of parameters  $\theta$ , it can be used for:

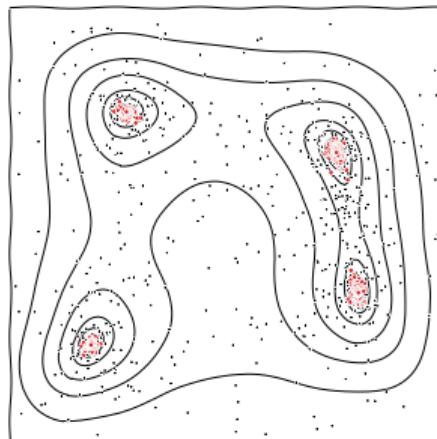
## Optimisation

$$\theta_{\max} = \max_{\theta} f(\theta)$$



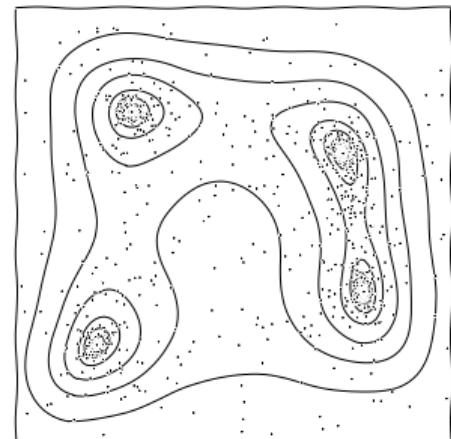
## Exploration

draw/sample  $\theta \sim f$



## Integration

$$\int f(\theta) dV$$



# Integration in Physics

- ▶ Integration is a fundamental concept in physics, statistics and data science:

## Partition functions

$$Z(\beta) = \int e^{-\beta H(q,p)} dq dp$$

## Path integrals

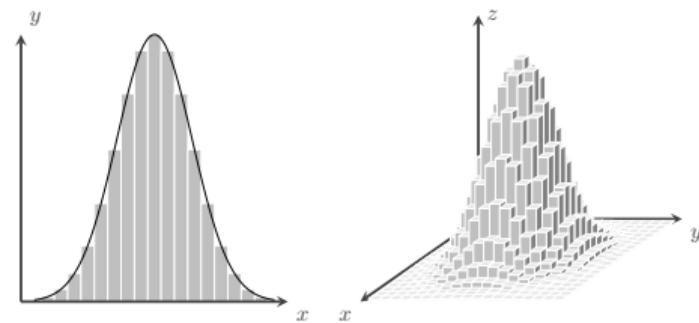
$$\Psi = \int e^{iS} \mathcal{D}x$$

## Bayesian marginals

$$\mathcal{Z}(D) = \int \mathcal{L}(D|\theta) \pi(\theta) d\theta$$

- ▶ Need numerical tools if analytic solution unavailable.
- ▶ High-dimensional numerical integration is hard.
- ▶ Riemannian strategy estimates volumes geometrically:

$$\int f(x) d^n x \approx \sum_i f(x_i) \Delta V_i \sim \mathcal{O}(e^n)$$



- ▶ Curse of dimensionality  $\Rightarrow$  exponential scaling.
- ▶ Nested sampling integrates **probabilistically**.

# Probabalistic volume estimation

- ▶ Key idea in NS: estimating volumes probabilistically

$$\frac{V_{\text{after}}}{V_{\text{before}}} \approx \frac{n_{\text{in}}}{n_{\text{out}} + n_{\text{in}}}$$

- ▶ This is the **only** way to calculate volume in high dimensions  $d > 3$ .
  - ▶ Geometry is exponentially inefficient.
- ▶ This estimation process does not depend on geometry, topology or dimensionality
- ▶ The errors however are not small.

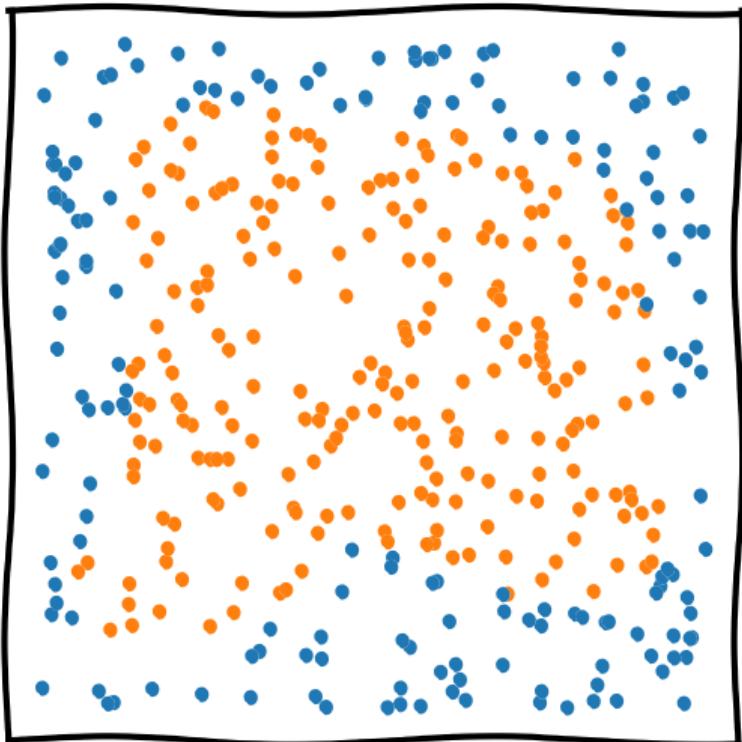


# Probabalistic volume estimation

- ▶ Key idea in NS: estimating volumes probabilistically

$$\frac{V_{\text{after}}}{V_{\text{before}}} \approx \frac{n_{\text{in}}}{n_{\text{out}} + n_{\text{in}}}$$

- ▶ This is the **only** way to calculate volume in high dimensions  $d > 3$ .
  - ▶ Geometry is exponentially inefficient.
- ▶ This estimation process does not depend on geometry, topology or dimensionality
- ▶ The errors however are not small.

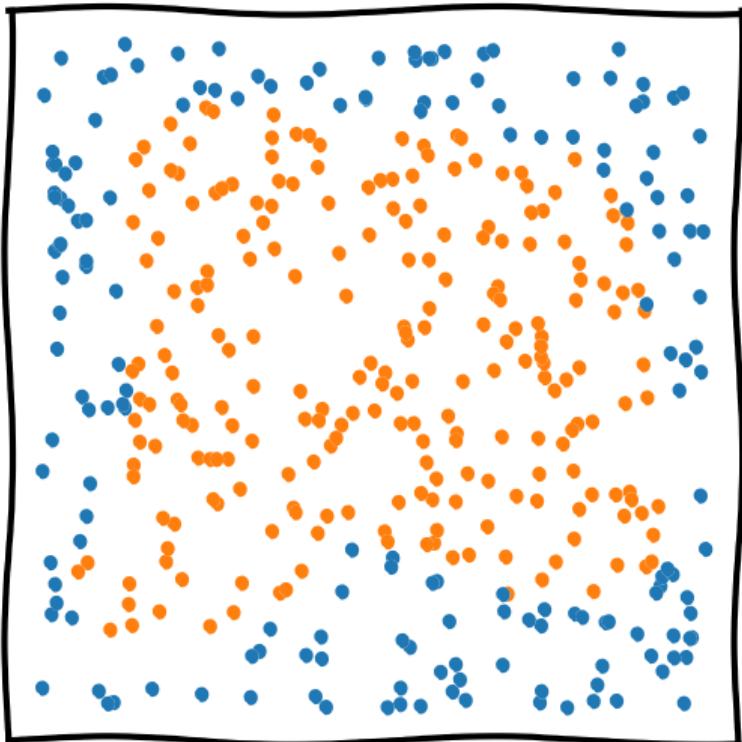


# Probabalistic volume estimation

- ▶ Key idea in NS: estimating volumes probabilistically

$$\frac{V_{\text{after}}}{V_{\text{before}}} \approx \frac{n_{\text{in}} + 1}{n_{\text{out}} + n_{\text{in}} + 2}$$

- ▶ This is the **only** way to calculate volume in high dimensions  $d > 3$ .
  - ▶ Geometry is exponentially inefficient.
- ▶ This estimation process does not depend on geometry, topology or dimensionality
- ▶ The errors however are not small.

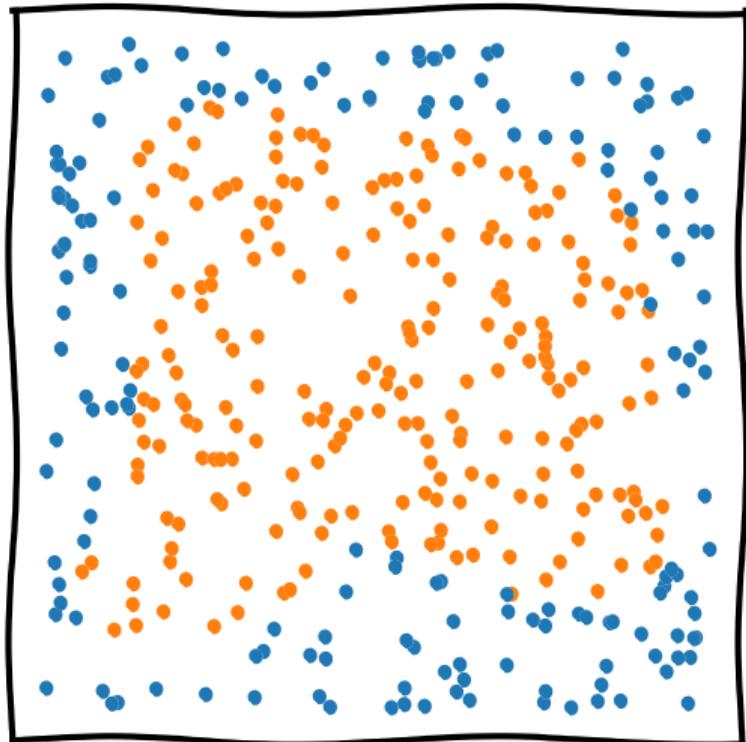


# Probabalistic volume estimation

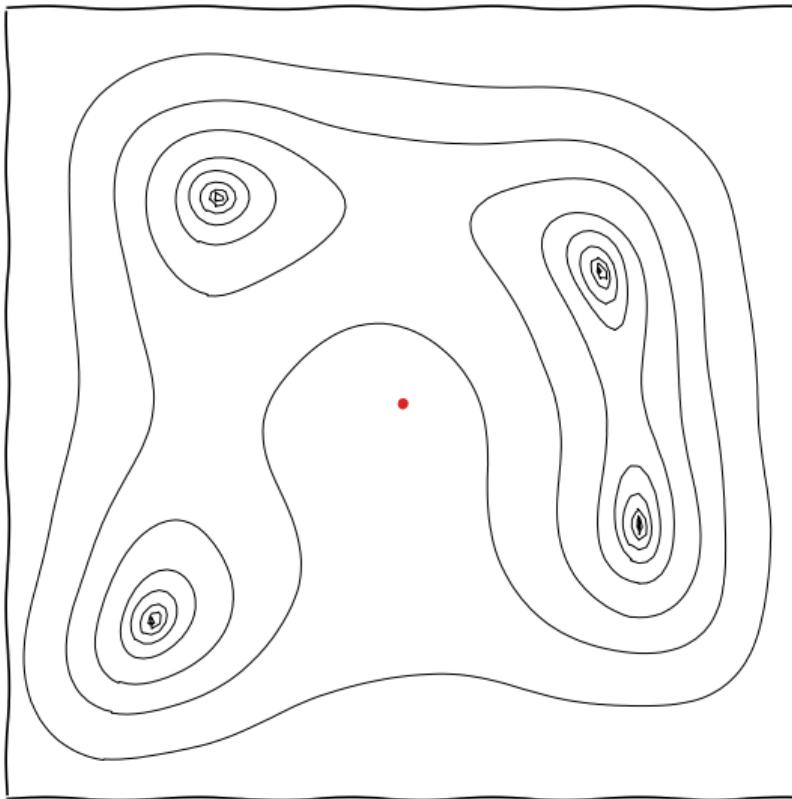
- ▶ Key idea in NS: estimating volumes probabilistically

$$\frac{V_{\text{after}}}{V_{\text{before}}} = \frac{n_{\text{in}} + 1}{n_{\text{out}} + n_{\text{in}} + 2} \pm \sqrt{\frac{(n_{\text{in}}+1)(n_{\text{out}}+1)}{(n_{\text{out}}+n_{\text{in}}+2)^2(n_{\text{out}}+n_{\text{in}}+3)}}$$

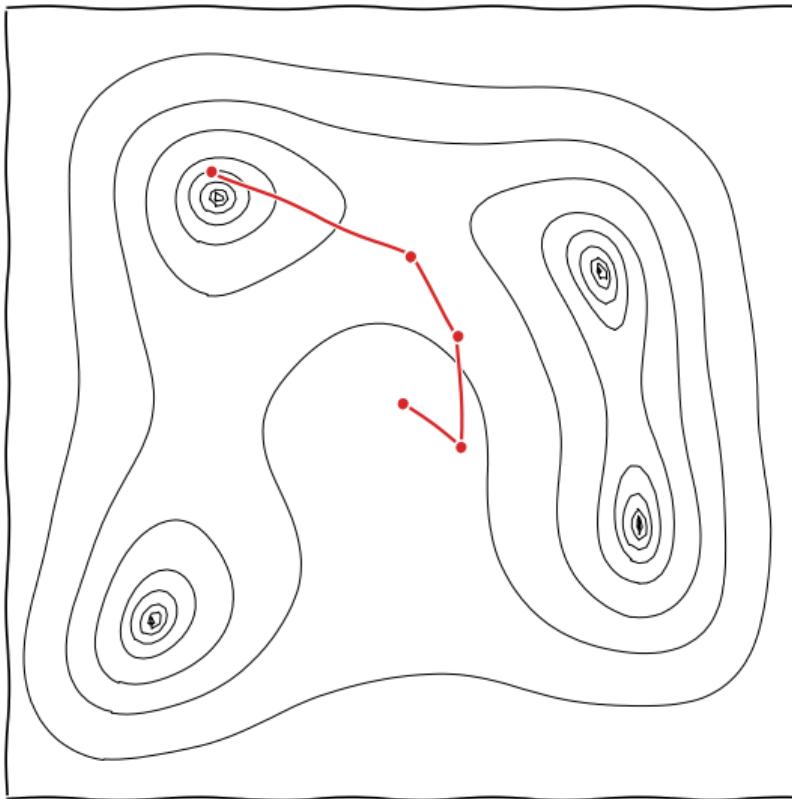
- ▶ This is the **only** way to calculate volume in high dimensions  $d > 3$ .
  - ▶ Geometry is exponentially inefficient.
- ▶ This estimation process does not depend on geometry, topology or dimensionality
- ▶ The errors however are not small.



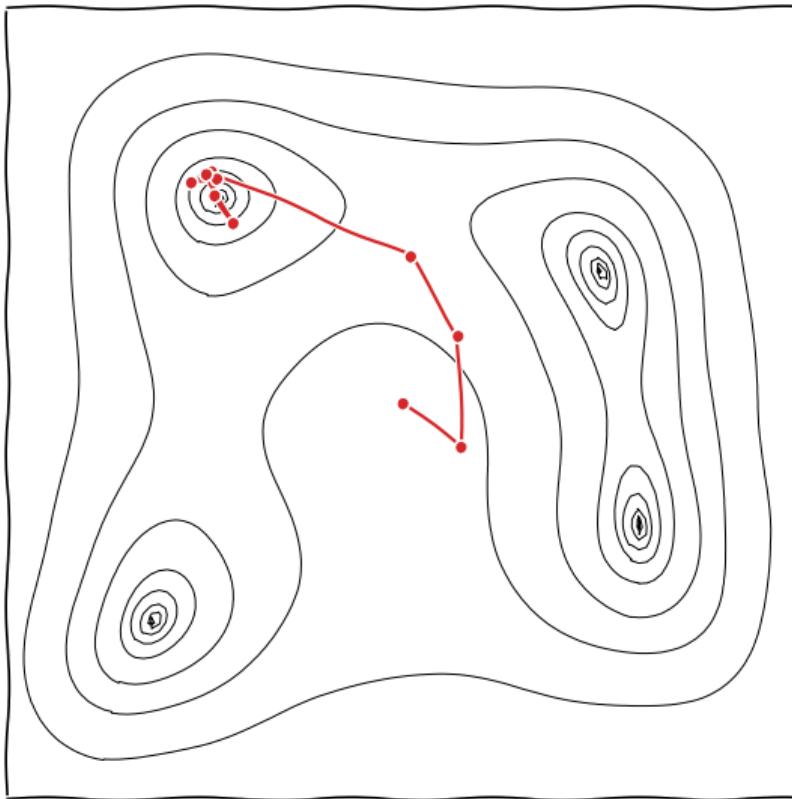
## MCMC



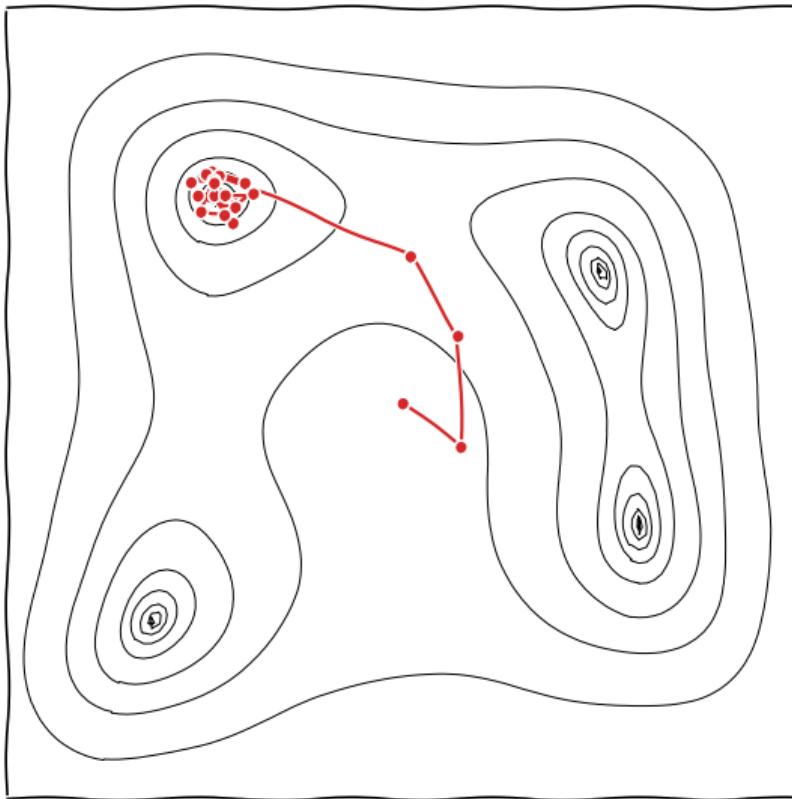
# MCMC



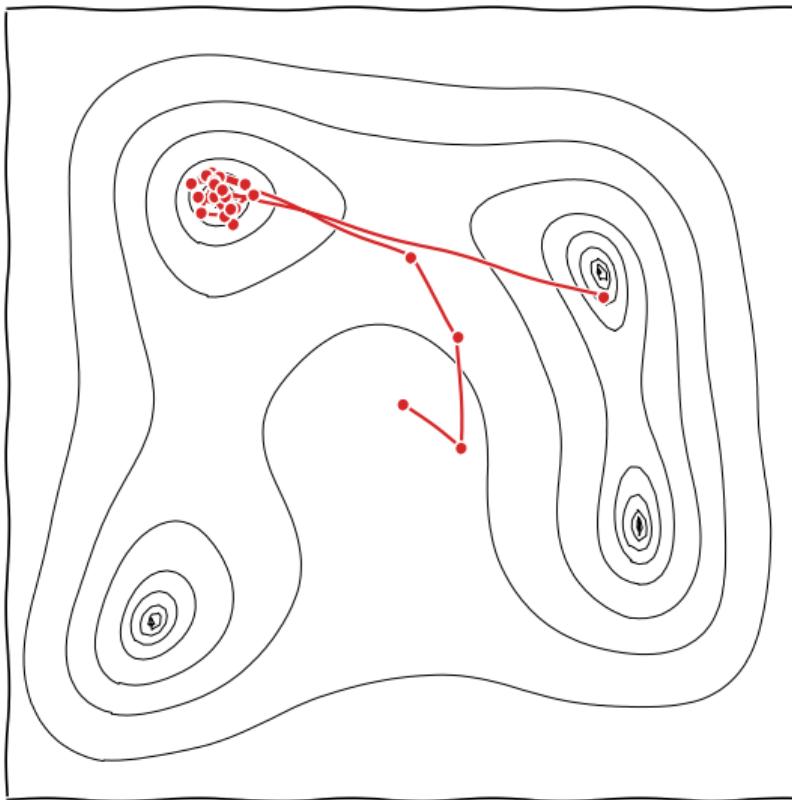
# MCMC



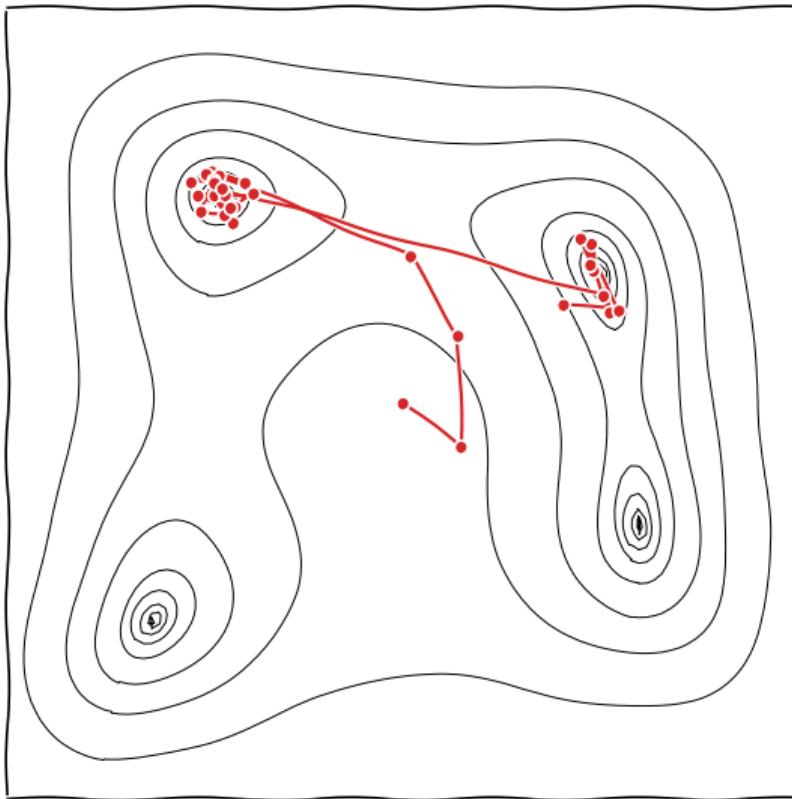
# MCMC



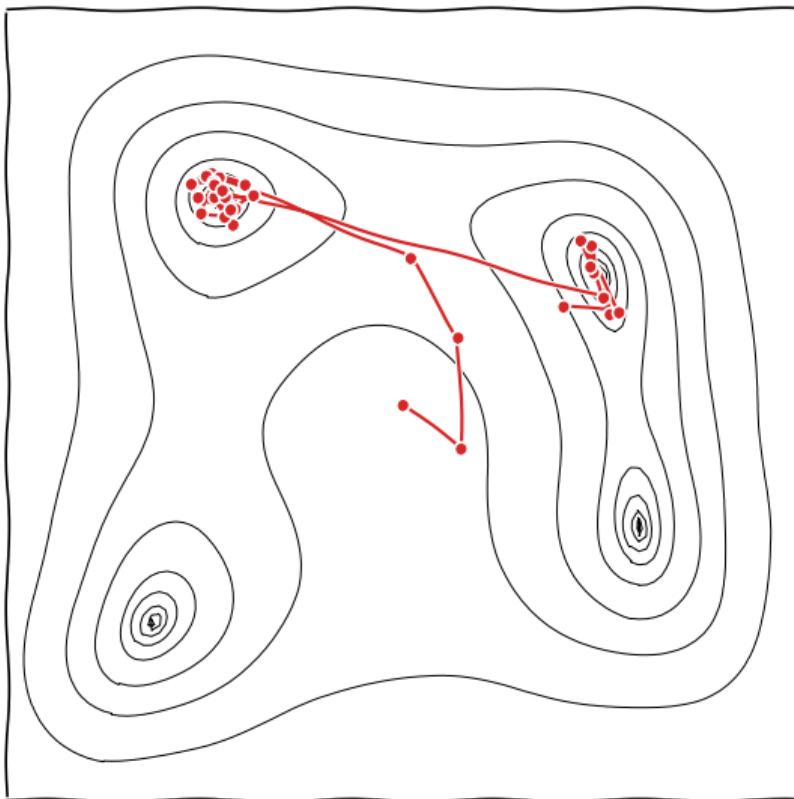
# MCMC



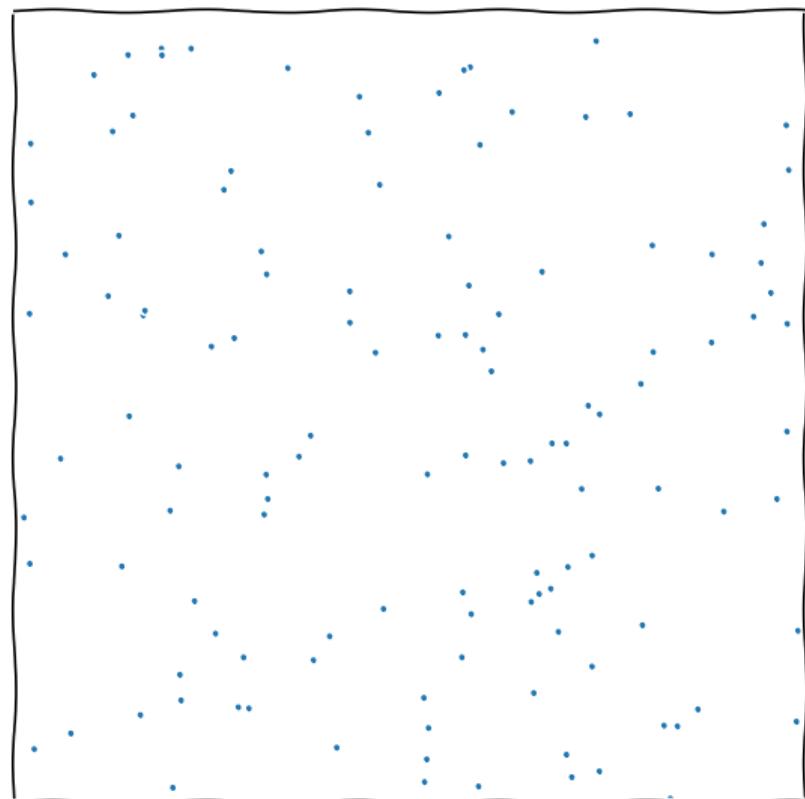
# MCMC



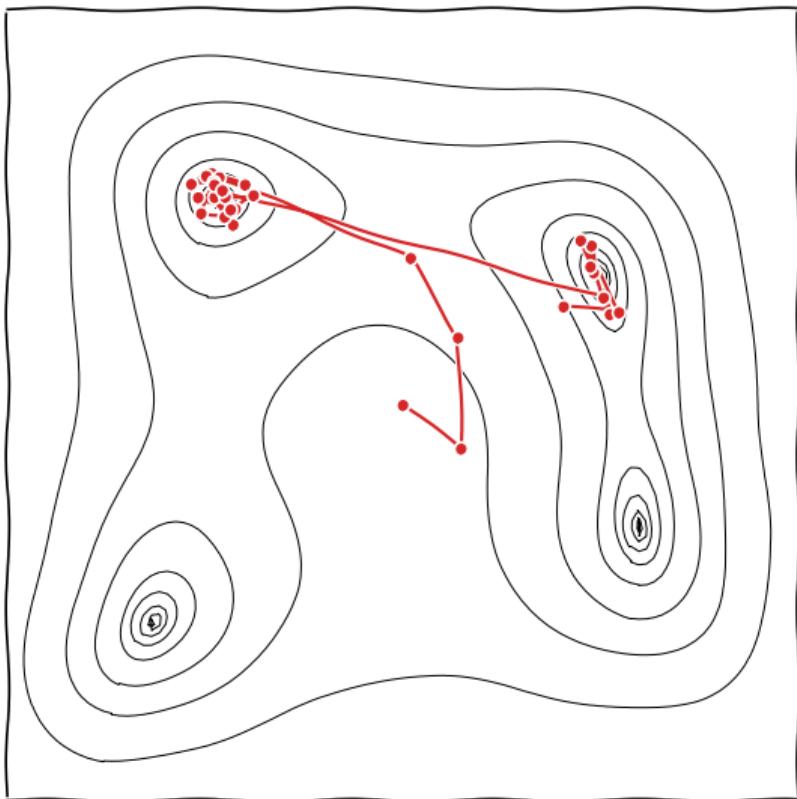
## MCMC



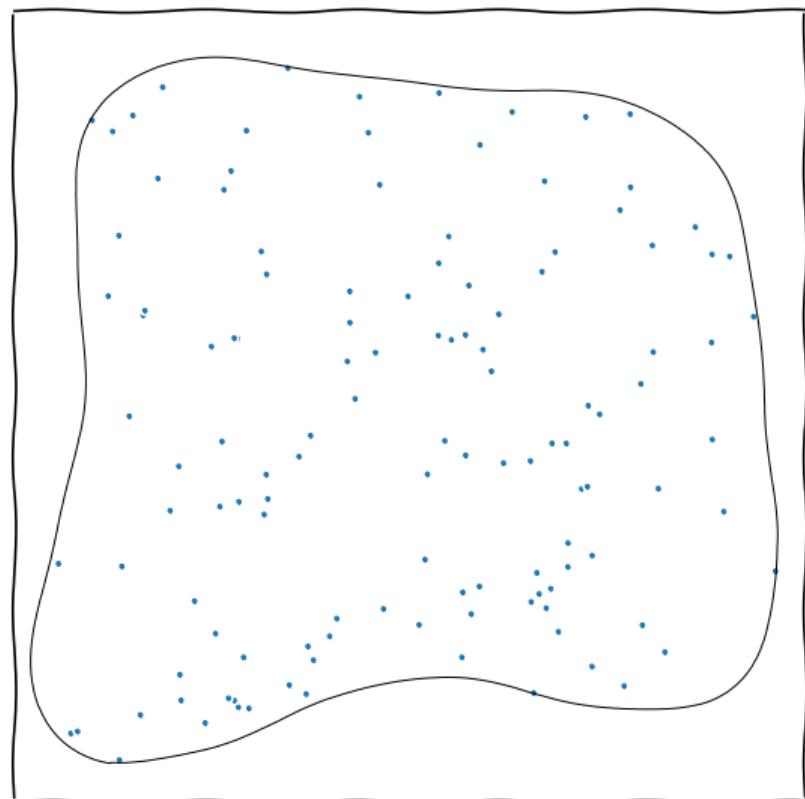
## Nested sampling



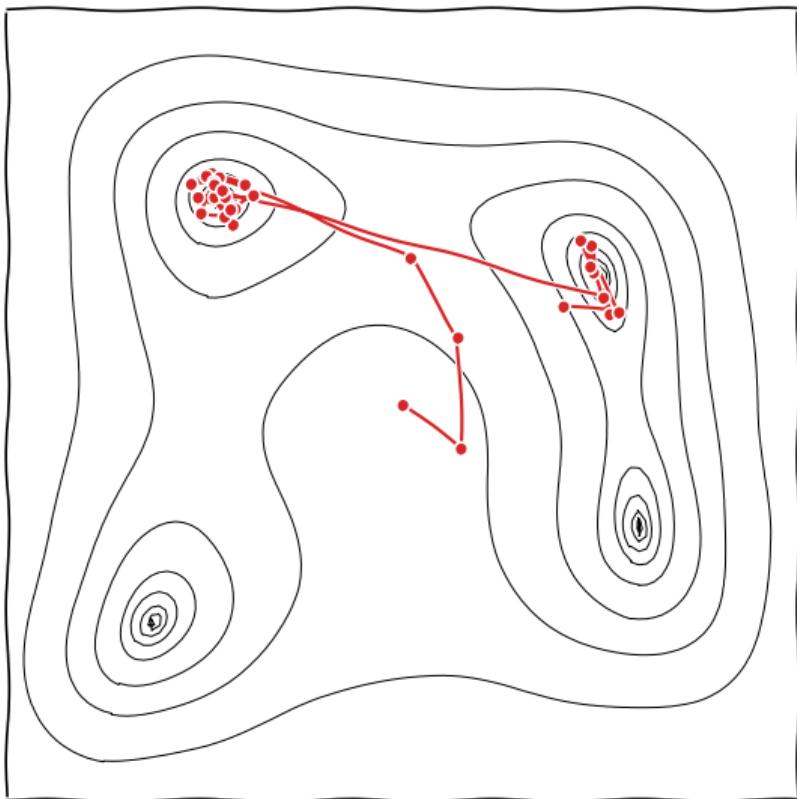
## MCMC



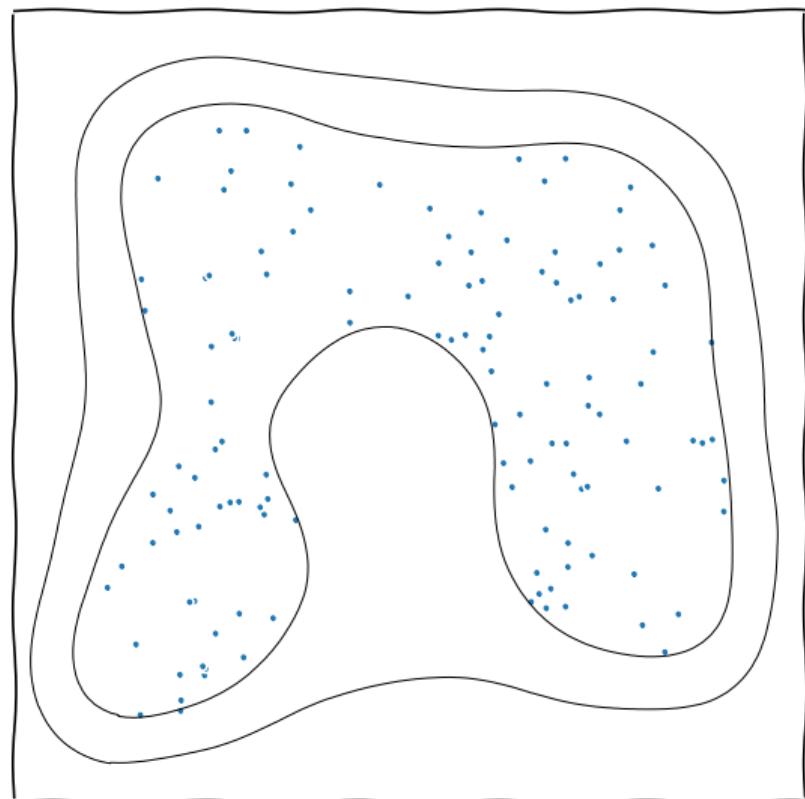
## Nested sampling



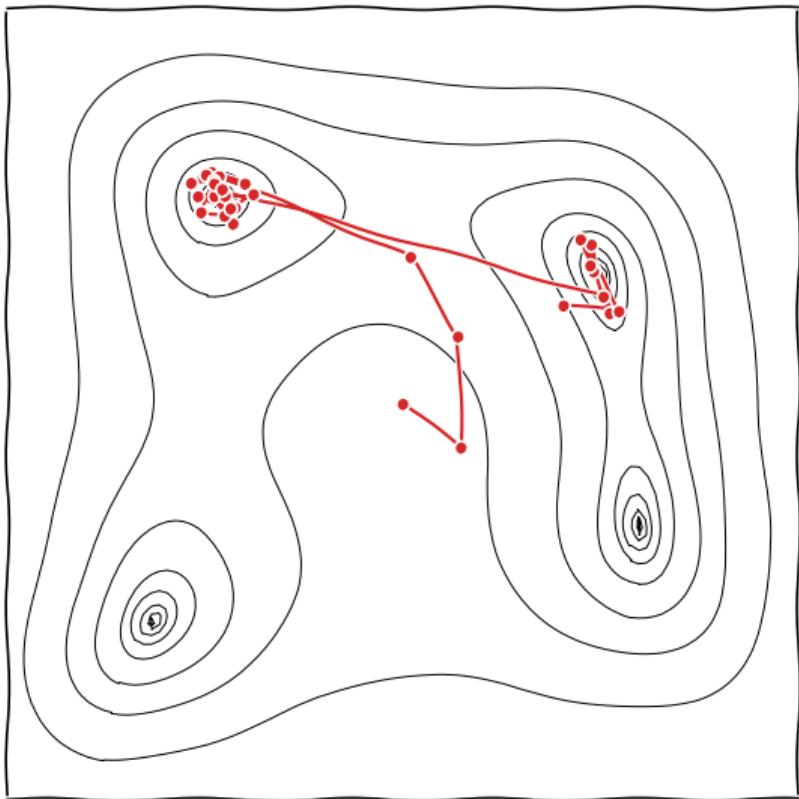
## MCMC



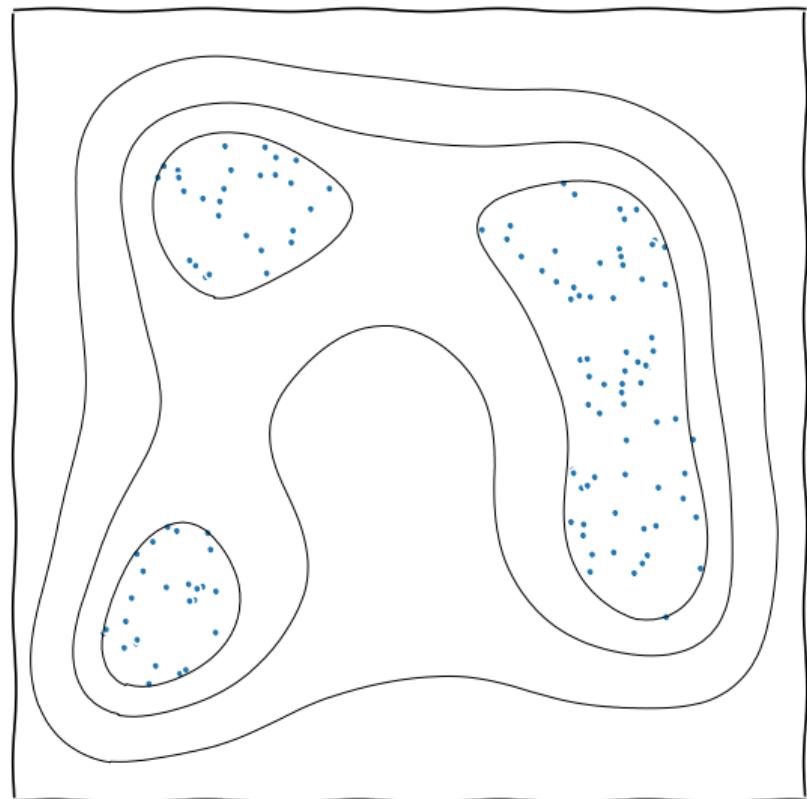
## Nested sampling



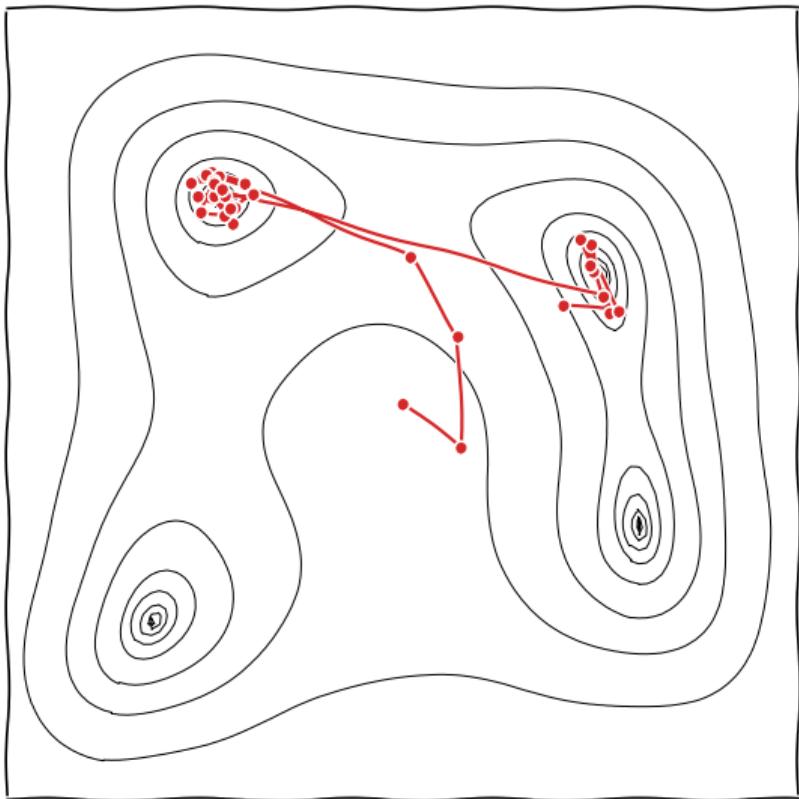
## MCMC



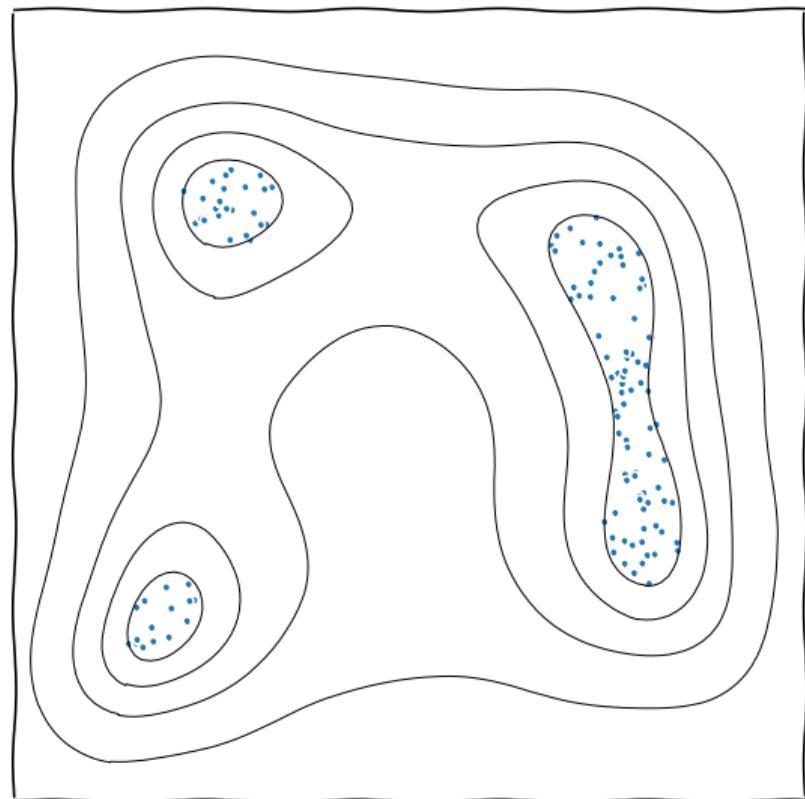
## Nested sampling



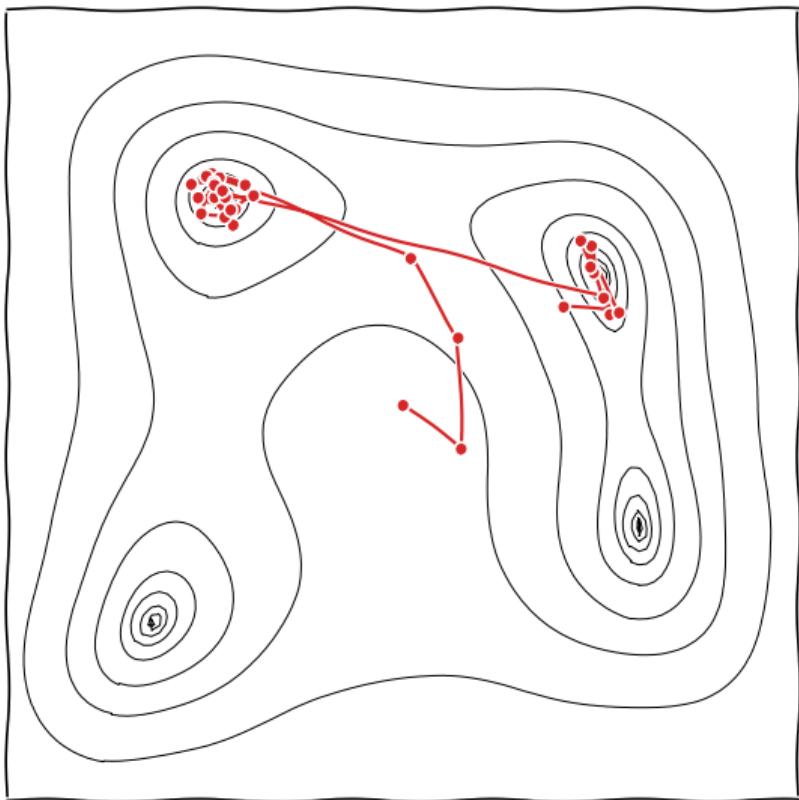
## MCMC



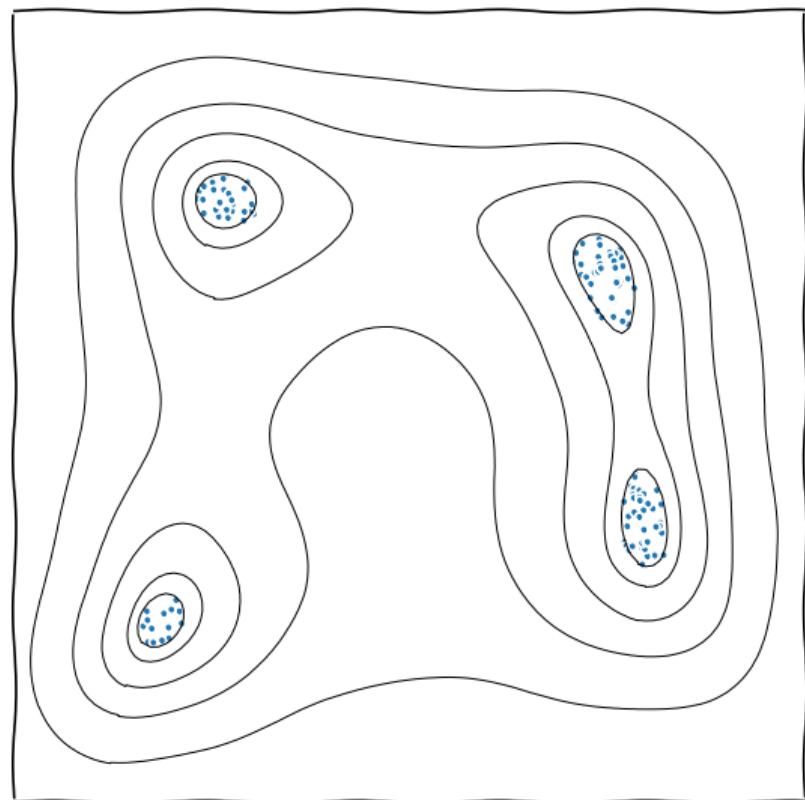
## Nested sampling



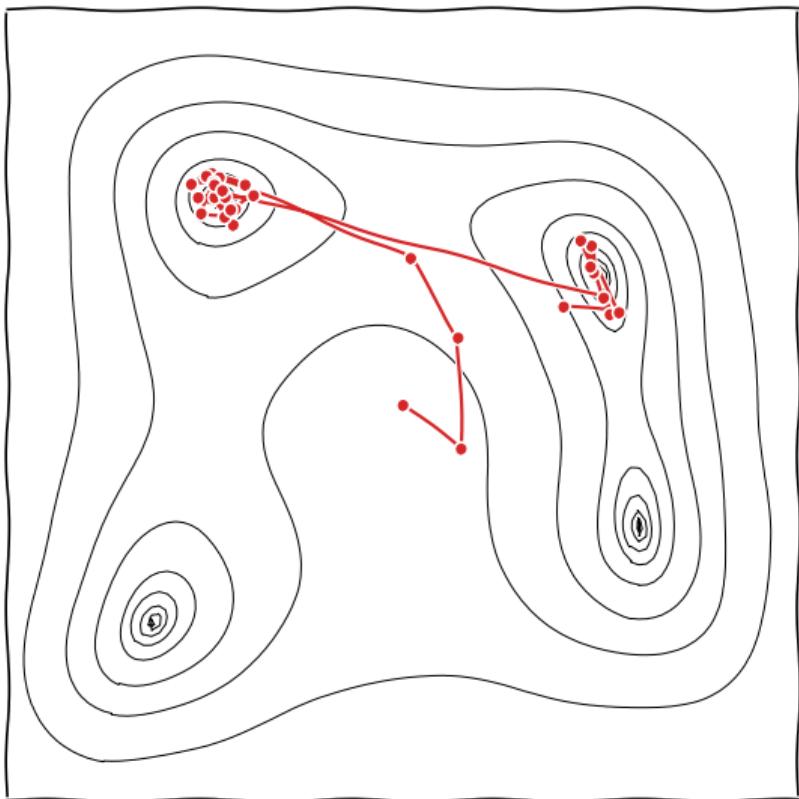
## MCMC



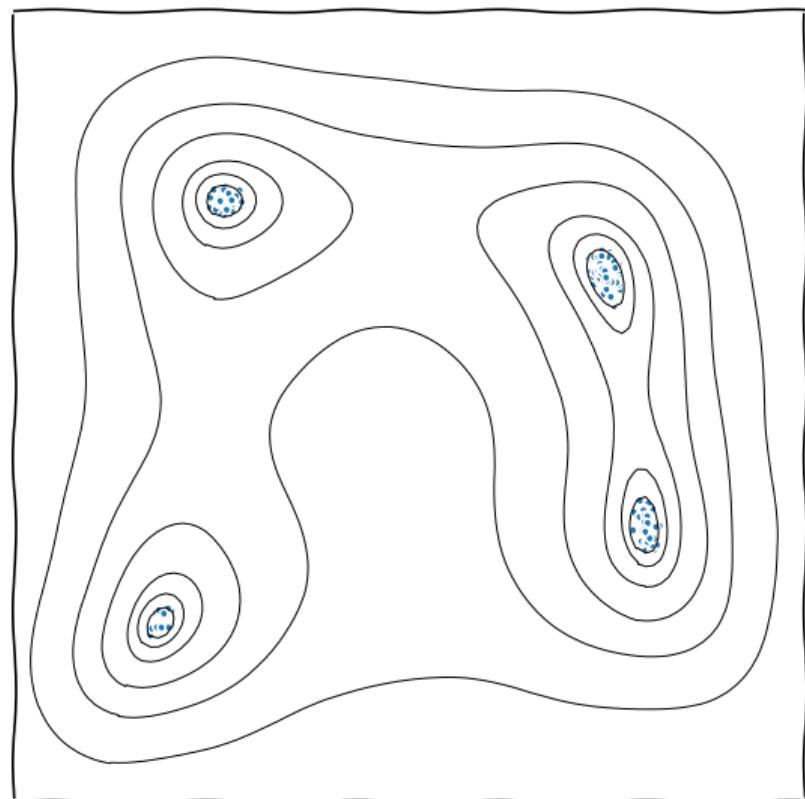
## Nested sampling



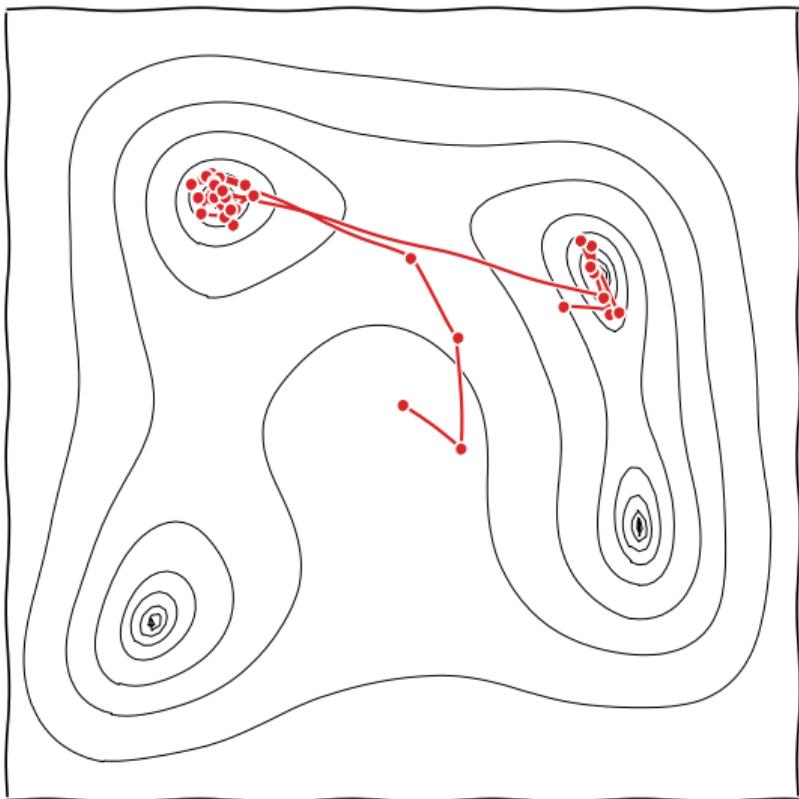
## MCMC



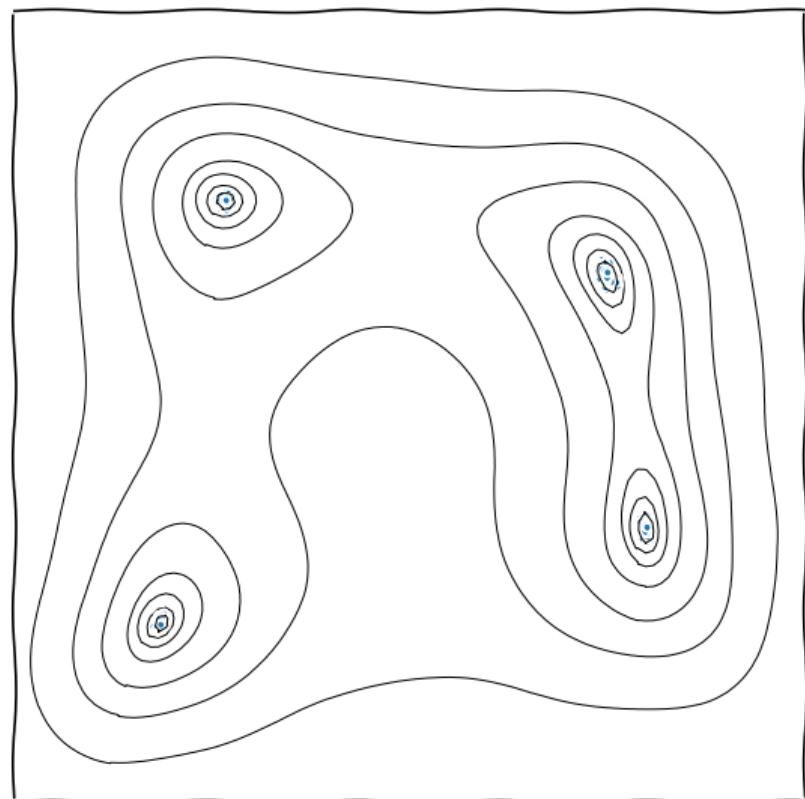
## Nested sampling



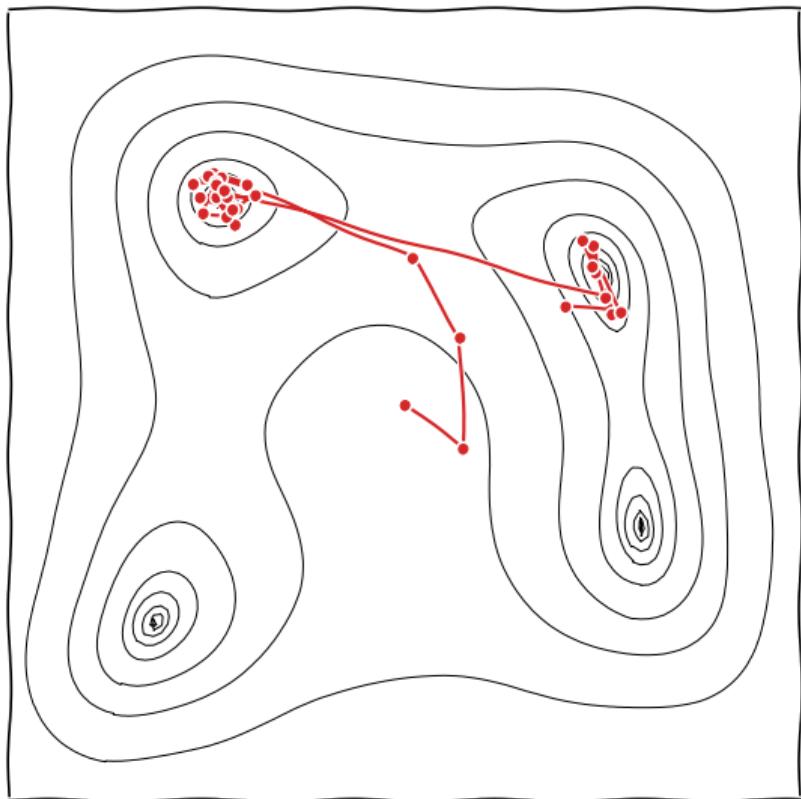
## MCMC



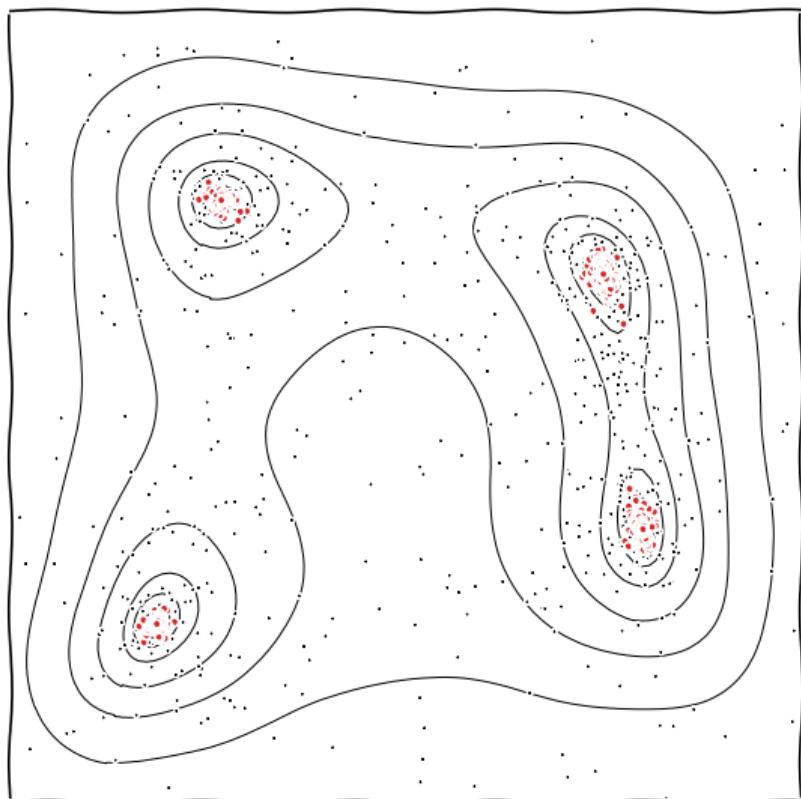
## Nested sampling



## MCMC

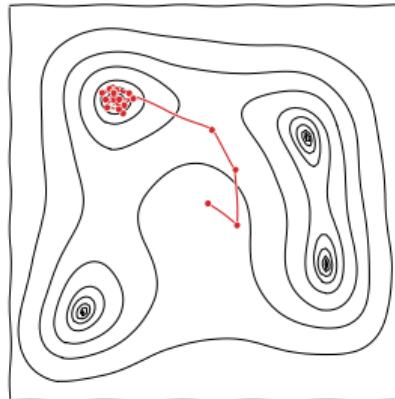


## Nested sampling



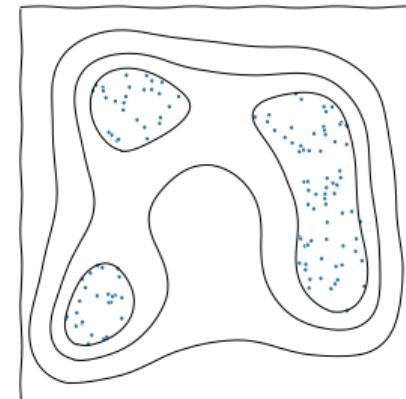
## MCMC

- ▶ Single “walker”
- ▶ Explores posterior
- ▶ Fast, if proposal matrix is tuned
- ▶ Parameter estimation, suspiciousness calculation
- ▶ Channel capacity optimised for generating posterior samples



## Nested sampling

- ▶ Ensemble of “live points”
- ▶ Scans from prior to peak of likelihood
- ▶ Slower, no tuning required
- ▶ Parameter estimation, model comparison, tension quantification
- ▶ Channel capacity optimised for computing partition function



## Nested sampling

- ▶ Sequentially update a set  $S$  of  $n$  samples:

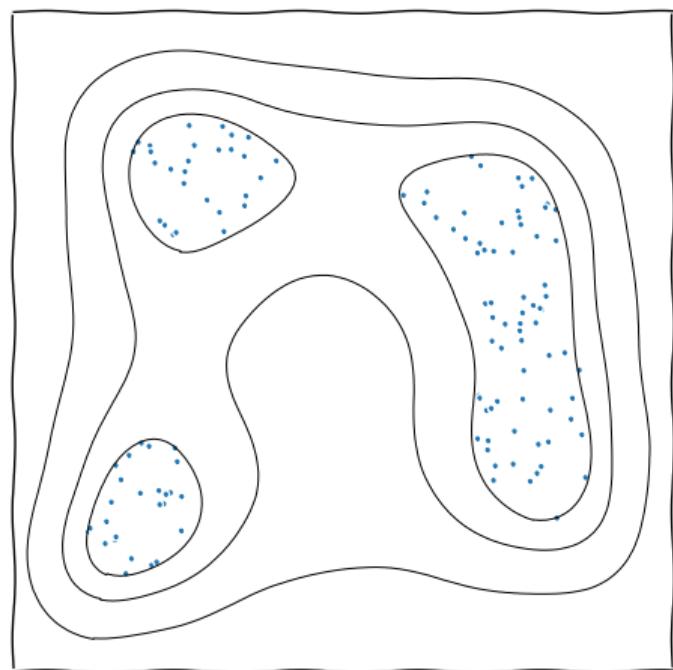
$S_0$ : Generate  $n$  samples uniformly over the space (from the prior  $\pi$ ).

$S_{i+1}$ : Delete the lowest likelihood sample in  $S_i$ , and replace it with a new uniform sample with higher likelihood.

- ▶ Requires one to be able to sample uniformly within a region, subject to a *hard likelihood constraint*:

$$\{\theta \sim \pi : \mathcal{L}(\theta) > \mathcal{L}_{*.\cdot}\}$$

- ▶ This procedure optimises (multimodally), and can calculate the **evidence** & **posterior** weights.
- ▶ The evolving ensemble of live points allows algorithms to perform self-tuning and mode clustering.

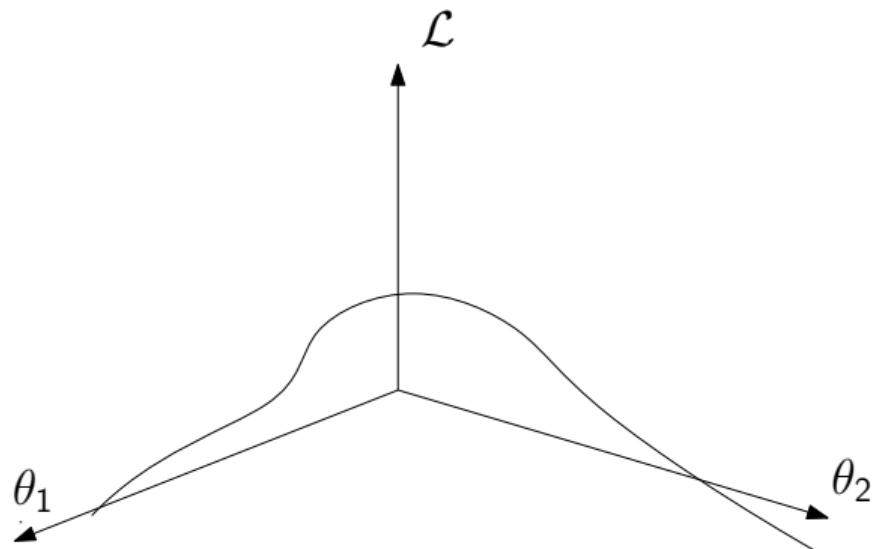


## (Lesbesgue) Integrating with nested sampling

- ▶ At each iteration, the likelihood contour shrinks in volume  $X$  by  $\approx 1/n$ .
- ▶ Nested sampling zooms in to the peak of the function  $\mathcal{L}$  exponentially.

$$\mathcal{Z} \approx \sum_i \Delta \mathcal{L}_i X_i, \quad X_{i+1} \approx \frac{n}{n+1} X_i, \quad X_0 = 1.$$

- ▶ At the end, one is left with a set of discarded points.
- ▶ These may be weighted to form weighted posterior samples using  $w_i = \mathcal{L}_i \Delta X_i$ .
- ▶ Estimates the density of states, and is therefore a partition function calculator  
 $Z(\beta) = \sum_i \mathcal{L}_i^\beta \Delta X_i$ .

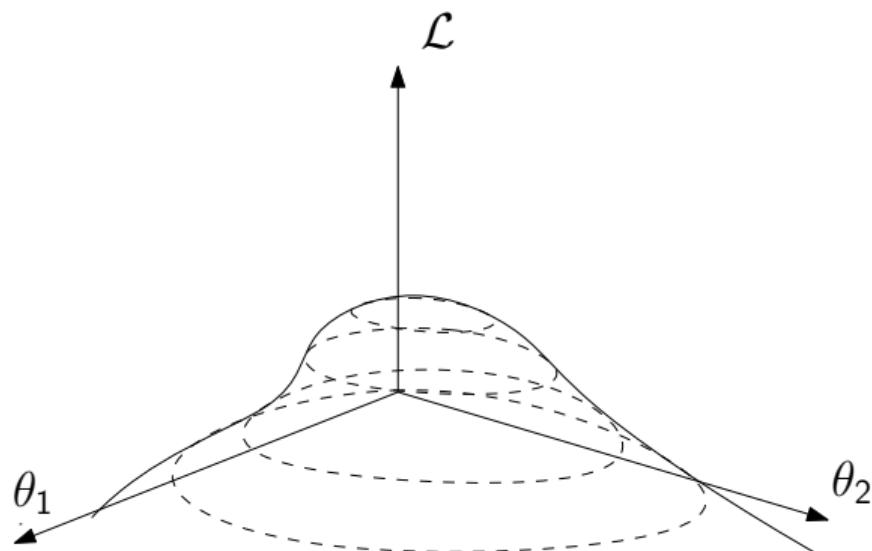


## (Lesbesgue) Integrating with nested sampling

- ▶ At each iteration, the likelihood contour shrinks in volume  $X$  by  $\approx 1/n$ .
- ▶ Nested sampling zooms in to the peak of the function  $\mathcal{L}$  exponentially.

$$\mathcal{Z} \approx \sum_i \Delta \mathcal{L}_i X_i, \quad X_{i+1} \approx \frac{n}{n+1} X_i, \quad X_0 = 1.$$

- ▶ At the end, one is left with a set of discarded points.
- ▶ These may be weighted to form weighted posterior samples using  $w_i = \mathcal{L}_i \Delta X_i$ .
- ▶ Estimates the density of states, and is therefore a partition function calculator  $Z(\beta) = \sum_i \mathcal{L}_i^\beta \Delta X_i$ .

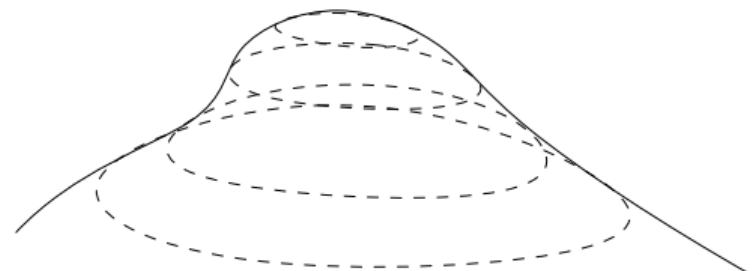


## (Lesbesgue) Integrating with nested sampling

- ▶ At each iteration, the likelihood contour shrinks in volume  $X$  by  $\approx 1/n$ .
- ▶ Nested sampling zooms in to the peak of the function  $\mathcal{L}$  exponentially.

$$\mathcal{Z} \approx \sum_i \Delta \mathcal{L}_i X_i, \quad X_{i+1} \approx \frac{n}{n+1} X_i, \quad X_0 = 1.$$

- ▶ At the end, one is left with a set of discarded points.
- ▶ These may be weighted to form weighted posterior samples using  $w_i = \mathcal{L}_i \Delta X_i$ .
- ▶ Estimates the density of states, and is therefore a partition function calculator  
 $Z(\beta) = \sum_i \mathcal{L}_i^\beta \Delta X_i$ .

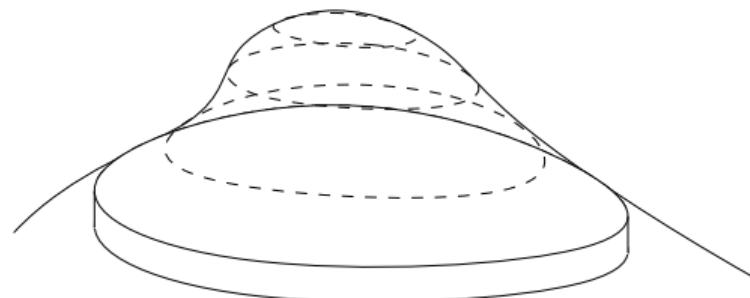


## (Lesbesgue) Integrating with nested sampling

- ▶ At each iteration, the likelihood contour shrinks in volume  $X$  by  $\approx 1/n$ .
- ▶ Nested sampling zooms in to the peak of the function  $\mathcal{L}$  exponentially.

$$\mathcal{Z} \approx \sum_i \Delta \mathcal{L}_i X_i, \quad X_{i+1} \approx \frac{n}{n+1} X_i, \quad X_0 = 1.$$

- ▶ At the end, one is left with a set of discarded points.
- ▶ These may be weighted to form weighted posterior samples using  $w_i = \mathcal{L}_i \Delta X_i$ .
- ▶ Estimates the density of states, and is therefore a partition function calculator  
 $Z(\beta) = \sum_i \mathcal{L}_i^\beta \Delta X_i$ .

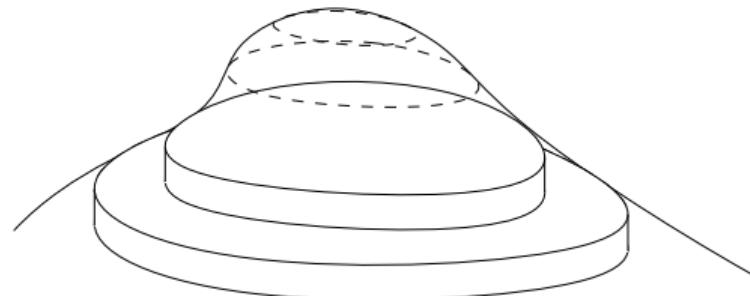


## (Lesbesgue) Integrating with nested sampling

- ▶ At each iteration, the likelihood contour shrinks in volume  $X$  by  $\approx 1/n$ .
- ▶ Nested sampling zooms in to the peak of the function  $\mathcal{L}$  exponentially.

$$\mathcal{Z} \approx \sum_i \Delta \mathcal{L}_i X_i, \quad X_{i+1} \approx \frac{n}{n+1} X_i, \quad X_0 = 1.$$

- ▶ At the end, one is left with a set of discarded points.
- ▶ These may be weighted to form weighted posterior samples using  $w_i = \mathcal{L}_i \Delta X_i$ .
- ▶ Estimates the density of states, and is therefore a partition function calculator  
 $Z(\beta) = \sum_i \mathcal{L}_i^\beta \Delta X_i$ .

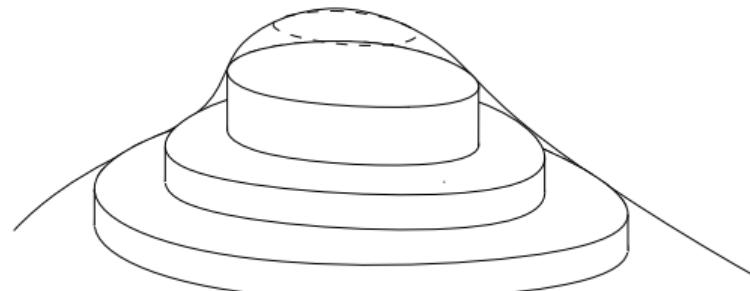


## (Lesbesgue) Integrating with nested sampling

- ▶ At each iteration, the likelihood contour shrinks in volume  $X$  by  $\approx 1/n$ .
- ▶ Nested sampling zooms in to the peak of the function  $\mathcal{L}$  exponentially.

$$\mathcal{Z} \approx \sum_i \Delta \mathcal{L}_i X_i, \quad X_{i+1} \approx \frac{n}{n+1} X_i, \quad X_0 = 1.$$

- ▶ At the end, one is left with a set of discarded points.
- ▶ These may be weighted to form weighted posterior samples using  $w_i = \mathcal{L}_i \Delta X_i$ .
- ▶ Estimates the density of states, and is therefore a partition function calculator  
 $Z(\beta) = \sum_i \mathcal{L}_i^\beta \Delta X_i$ .

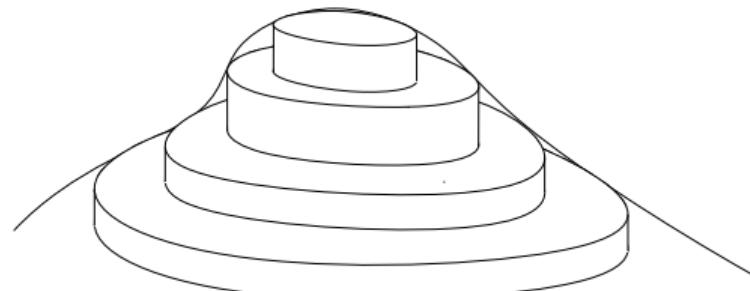


## (Lesbesgue) Integrating with nested sampling

- ▶ At each iteration, the likelihood contour shrinks in volume  $X$  by  $\approx 1/n$ .
- ▶ Nested sampling zooms in to the peak of the function  $\mathcal{L}$  exponentially.

$$\mathcal{Z} \approx \sum_i \Delta \mathcal{L}_i X_i, \quad X_{i+1} \approx \frac{n}{n+1} X_i, \quad X_0 = 1.$$

- ▶ At the end, one is left with a set of discarded points.
- ▶ These may be weighted to form weighted posterior samples using  $w_i = \mathcal{L}_i \Delta X_i$ .
- ▶ Estimates the density of states, and is therefore a partition function calculator  
 $Z(\beta) = \sum_i \mathcal{L}_i^\beta \Delta X_i$ .

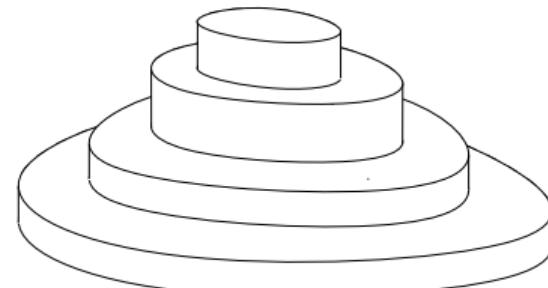


## (Lesbesgue) Integrating with nested sampling

- ▶ At each iteration, the likelihood contour shrinks in volume  $X$  by  $\approx 1/n$ .
- ▶ Nested sampling zooms in to the peak of the function  $\mathcal{L}$  exponentially.

$$\mathcal{Z} \approx \sum_i \Delta \mathcal{L}_i X_i, \quad X_{i+1} \approx \frac{n}{n+1} X_i, \quad X_0 = 1.$$

- ▶ At the end, one is left with a set of discarded points.
- ▶ These may be weighted to form weighted posterior samples using  $w_i = \mathcal{L}_i \Delta X_i$ .
- ▶ Estimates the density of states, and is therefore a partition function calculator  
 $Z(\beta) = \sum_i \mathcal{L}_i^\beta \Delta X_i$ .

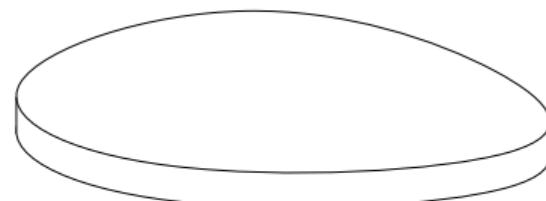


## (Lesbesgue) Integrating with nested sampling

- ▶ At each iteration, the likelihood contour shrinks in volume  $X$  by  $\approx 1/n$ .
- ▶ Nested sampling zooms in to the peak of the function  $\mathcal{L}$  exponentially.

$$\mathcal{Z} \approx \sum_i \Delta \mathcal{L}_i X_i, \quad X_{i+1} \approx \frac{n}{n+1} X_i, \quad X_0 = 1.$$

- ▶ At the end, one is left with a set of discarded points.
- ▶ These may be weighted to form weighted posterior samples using  $w_i = \mathcal{L}_i \Delta X_i$ .
- ▶ Estimates the density of states, and is therefore a partition function calculator  
 $Z(\beta) = \sum_i \mathcal{L}_i^\beta \Delta X_i$ .



## (Lesbesgue) Integrating with nested sampling

- ▶ At each iteration, the likelihood contour shrinks in volume  $X$  by  $\approx 1/n$ .
- ▶ Nested sampling zooms in to the peak of the function  $\mathcal{L}$  exponentially.

$$\mathcal{Z} \approx \sum_i \Delta \mathcal{L}_i X_i, \quad X_{i+1} \approx \frac{n}{n+1} X_i, \quad X_0 = 1.$$

- ▶ At the end, one is left with a set of discarded points.
- ▶ These may be weighted to form weighted posterior samples using  $w_i = \mathcal{L}_i \Delta X_i$ .
- ▶ Estimates the density of states, and is therefore a partition function calculator  
 $Z(\beta) = \sum_i \mathcal{L}_i^\beta \Delta X_i$ .

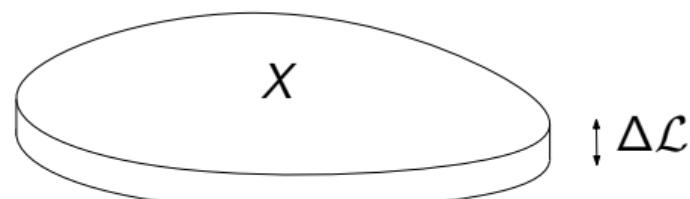


## (Lesbesgue) Integrating with nested sampling

- ▶ At each iteration, the likelihood contour shrinks in volume  $X$  by  $\approx 1/n$ .
- ▶ Nested sampling zooms in to the peak of the function  $\mathcal{L}$  exponentially.

$$\mathcal{Z} \approx \sum_i \Delta \mathcal{L}_i X_i, \quad X_{i+1} \approx \frac{n}{n+1} X_i, \quad X_0 = 1.$$

- ▶ At the end, one is left with a set of discarded points.
- ▶ These may be weighted to form weighted posterior samples using  $w_i = \mathcal{L}_i \Delta X_i$ .
- ▶ Estimates the density of states, and is therefore a partition function calculator  
 $Z(\beta) = \sum_i \mathcal{L}_i^\beta \Delta X_i$ .

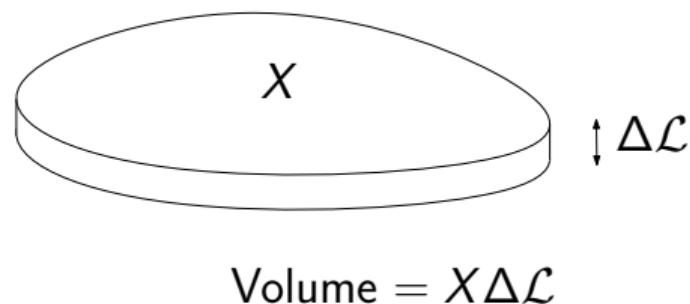


## (Lesbesgue) Integrating with nested sampling

- ▶ At each iteration, the likelihood contour shrinks in volume  $X$  by  $\approx 1/n$ .
- ▶ Nested sampling zooms in to the peak of the function  $\mathcal{L}$  exponentially.

$$\mathcal{Z} \approx \sum_i \Delta \mathcal{L}_i X_i, \quad X_{i+1} \approx \frac{n}{n+1} X_i, \quad X_0 = 1.$$

- ▶ At the end, one is left with a set of discarded points.
- ▶ These may be weighted to form weighted posterior samples using  $w_i = \mathcal{L}_i \Delta X_i$ .
- ▶ Estimates the density of states, and is therefore a partition function calculator  
 $Z(\beta) = \sum_i \mathcal{L}_i^\beta \Delta X_i$ .

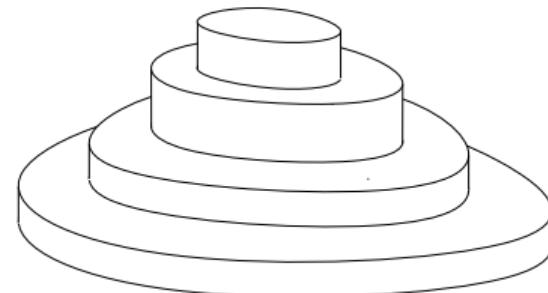


## (Lesbesgue) Integrating with nested sampling

- ▶ At each iteration, the likelihood contour shrinks in volume  $X$  by  $\approx 1/n$ .
- ▶ Nested sampling zooms in to the peak of the function  $\mathcal{L}$  exponentially.

$$\mathcal{Z} \approx \sum_i \Delta \mathcal{L}_i X_i, \quad X_{i+1} \approx \frac{n}{n+1} X_i, \quad X_0 = 1.$$

- ▶ At the end, one is left with a set of discarded points.
- ▶ These may be weighted to form weighted posterior samples using  $w_i = \mathcal{L}_i \Delta X_i$ .
- ▶ Estimates the density of states, and is therefore a partition function calculator  
 $Z(\beta) = \sum_i \mathcal{L}_i^\beta \Delta X_i$ .

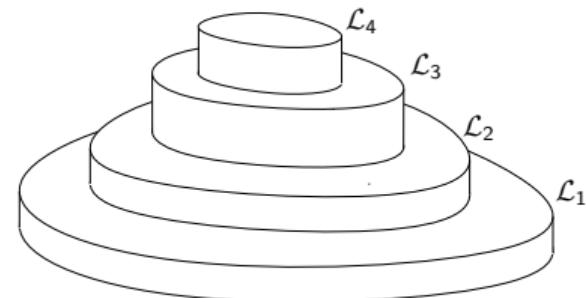


## (Lesbesgue) Integrating with nested sampling

- ▶ At each iteration, the likelihood contour shrinks in volume  $X$  by  $\approx 1/n$ .
- ▶ Nested sampling zooms in to the peak of the function  $\mathcal{L}$  exponentially.

$$\mathcal{Z} \approx \sum_i \Delta \mathcal{L}_i X_i, \quad X_{i+1} \approx \frac{n}{n+1} X_i, \quad X_0 = 1.$$

- ▶ At the end, one is left with a set of discarded points.
- ▶ These may be weighted to form weighted posterior samples using  $w_i = \mathcal{L}_i \Delta X_i$ .
- ▶ Estimates the density of states, and is therefore a partition function calculator  
 $Z(\beta) = \sum_i \mathcal{L}_i^\beta \Delta X_i$ .

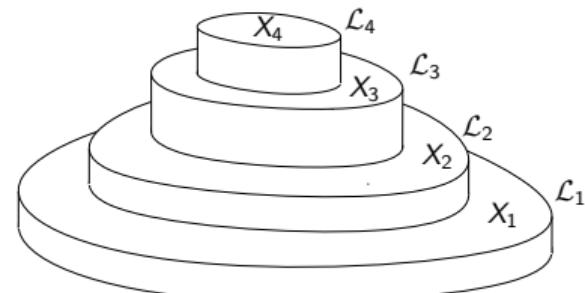


## (Lesbesgue) Integrating with nested sampling

- ▶ At each iteration, the likelihood contour shrinks in volume  $X$  by  $\approx 1/n$ .
- ▶ Nested sampling zooms in to the peak of the function  $\mathcal{L}$  exponentially.

$$\mathcal{Z} \approx \sum_i \Delta \mathcal{L}_i X_i, \quad X_{i+1} \approx \frac{n}{n+1} X_i, \quad X_0 = 1.$$

- ▶ At the end, one is left with a set of discarded points.
- ▶ These may be weighted to form weighted posterior samples using  $w_i = \mathcal{L}_i \Delta X_i$ .
- ▶ Estimates the density of states, and is therefore a partition function calculator  
 $Z(\beta) = \sum_i \mathcal{L}_i^\beta \Delta X_i$ .



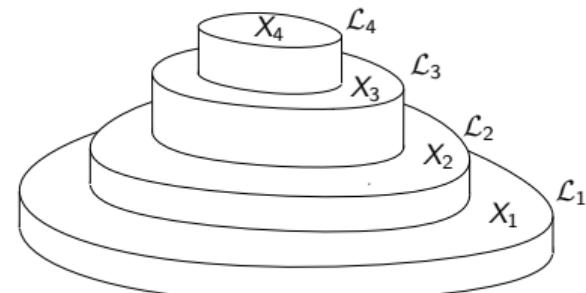
## (Lesbesgue) Integrating with nested sampling

- ▶ At each iteration, the likelihood contour shrinks in volume  $X$  by  $\approx 1/n$ .
- ▶ Nested sampling zooms in to the peak of the function  $\mathcal{L}$  exponentially.

$$\mathcal{Z} \approx \sum_i \Delta \mathcal{L}_i X_i, \quad X_{i+1} \approx \frac{n}{n+1} X_i, \quad X_0 = 1.$$

- ▶ At the end, one is left with a set of discarded points.
- ▶ These may be weighted to form weighted posterior samples using  $w_i = \mathcal{L}_i \Delta X_i$ .
- ▶ Estimates the density of states, and is therefore a partition function calculator  
 $Z(\beta) = \sum_i \mathcal{L}_i^\beta \Delta X_i$ .

$$\mathcal{Z} \approx \sum_i X_i \Delta \mathcal{L}_i$$

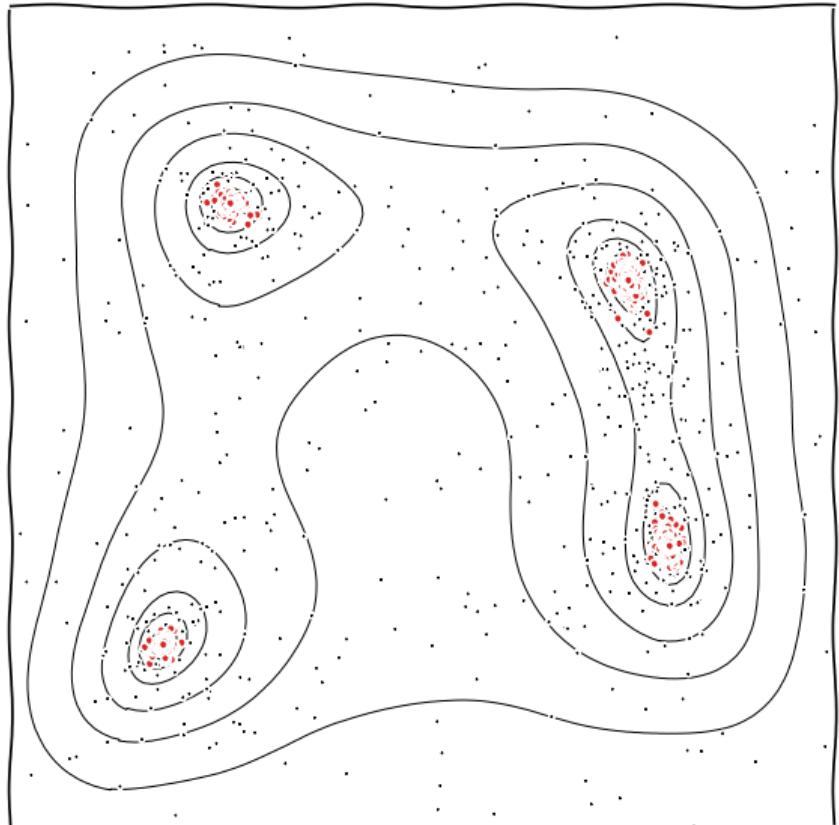


## (Lesbesgue) Integrating with nested sampling

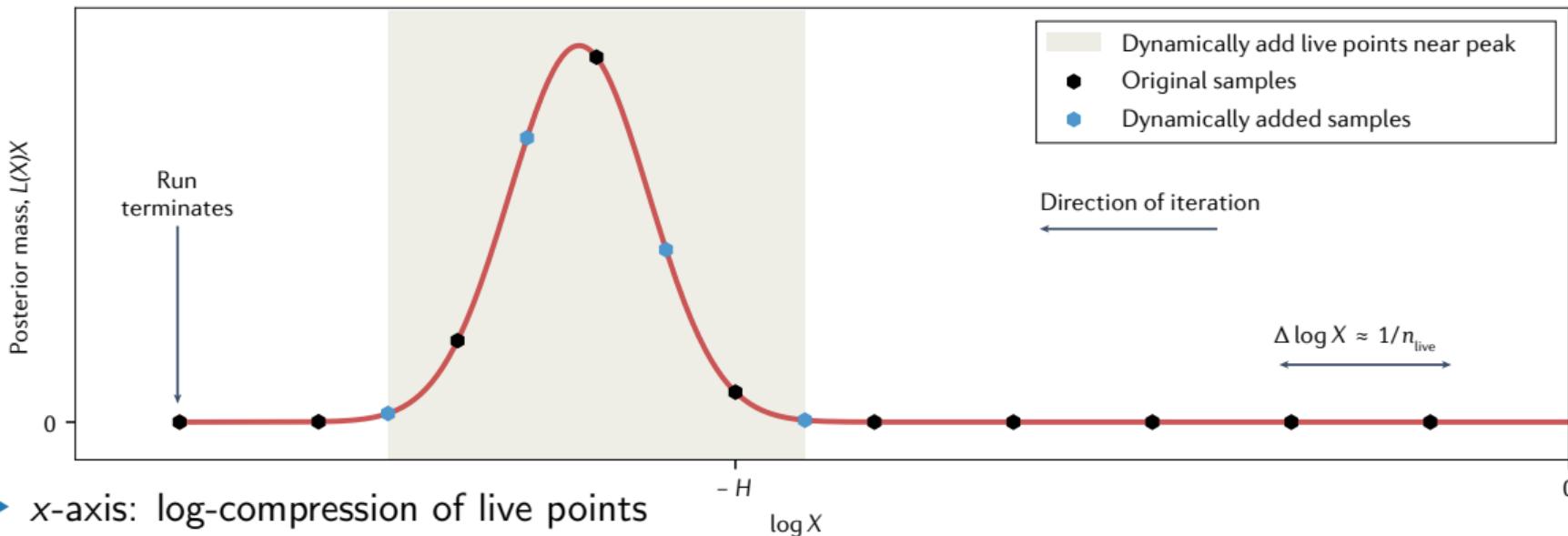
- ▶ At each iteration, the likelihood contour shrinks in volume  $X$  by  $\approx 1/n$ .
- ▶ Nested sampling zooms in to the peak of the function  $\mathcal{L}$  exponentially.

$$\mathcal{Z} \approx \sum_i \Delta \mathcal{L}_i X_i, \quad X_{i+1} \approx \frac{n}{n+1} X_i, \quad X_0 = 1.$$

- ▶ At the end, one is left with a set of discarded points.
- ▶ These may be weighted to form weighted posterior samples using  $w_i = \mathcal{L}_i \Delta X_i$ .
- ▶ Estimates the density of states, and is therefore a partition function calculator  $Z(\beta) = \sum_i \mathcal{L}_i^\beta \Delta X_i$ .



# Time complexity of nested sampling



► Time complexity

$$T = n_{\text{live}} \times T_{\mathcal{L}} \times T_{\text{sampler}} \times D_{\text{KL}}(\mathcal{P} \parallel \pi)$$

► Error complexity  $\sigma \propto \sqrt{D_{\text{KL}}(\mathcal{P} \parallel \pi) / n_{\text{live}}}$

## Sampling from a hard likelihood constraint

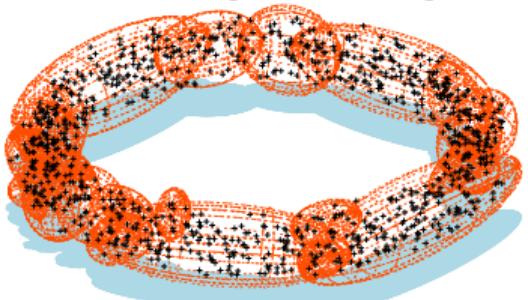
*"It is not the purpose of this introductory paper to develop the technology of navigation within such a volume. We merely note that exploring a hard-edged likelihood-constrained domain should prove to be neither more nor less demanding than exploring a likelihood-weighted space."*

— John Skilling

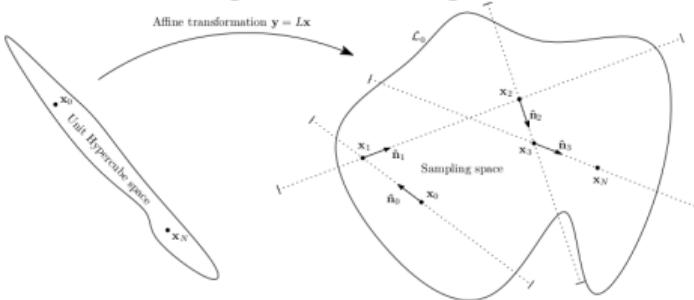
- ▶ A large fraction of the work in NS to date has been in attempting to implement a hard-edged sampler in the NS meta-algorithm  $\{\theta \sim \pi : \mathcal{L}(\theta) > \mathcal{L}_*\}$ .
- ▶ <https://projecteuclid.org/euclid.ba/1340370944>.
- ▶ There has also been much work beyond this (see "Frontiers of nested sampling" talk from last year: [willhandley.co.uk/talks](http://willhandley.co.uk/talks))

# Implementations of Nested Sampling [2205.15570](NatReview)

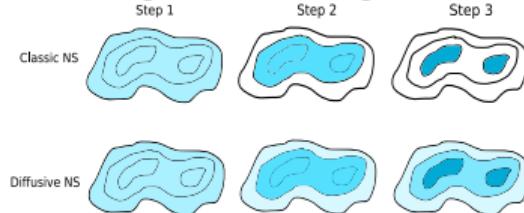
MultiNest [0809.3437]



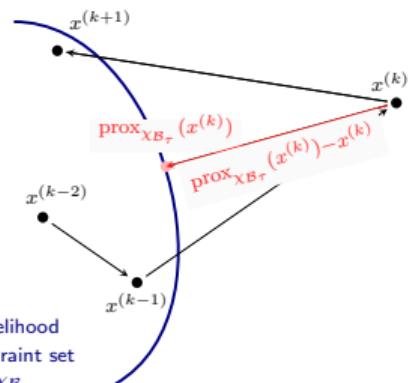
PolyChord [1506.00171]



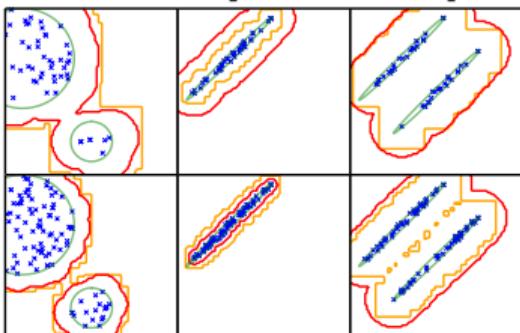
DNest [1606.03757]



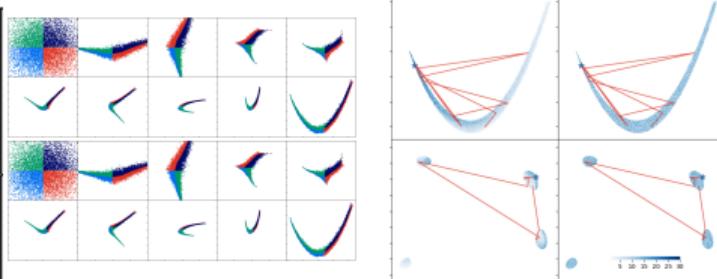
ProxNest [2106.03646]



UltraNest [2101.09604]



NeuralNest [1903.10860]



nessai [2102.11056]

nora [2305.19267]

dynesty [1904.02180]

# Types of nested sampler

- ▶ Broadly, most nested samplers can be split into how they create new live points.
- ▶ i.e. how they sample from the hard likelihood constraint  $\{\theta \sim \pi : \mathcal{L}(\theta) > \mathcal{L}_*\}$ .

## Rejection samplers

- ▶ e.g. MultiNest, UltraNest.
- ▶ Constructs bounding region and draws many invalid points until  $\mathcal{L}(\theta) > \mathcal{L}_*$ .
- ▶ Efficient in low dimensions, exponentially inefficient  $\sim \mathcal{O}(e^{d/d_0})$  in high  $d > d_0 \sim 10$ .

- ▶ Nested samplers usually come with:

- ▶ *resolution* parameter  $n_{\text{live}}$  (which improve results as  $\sim \mathcal{O}(n_{\text{live}}^{-1/2})$ ).
- ▶ set of *reliability* parameters [2101.04525], which don't improve results if set arbitrarily high, but introduce systematic errors if set too low.
- ▶ e.g. Multinest efficiency  $\text{eff}$  or PolyChord chain length  $n_{\text{repeats}}$ .

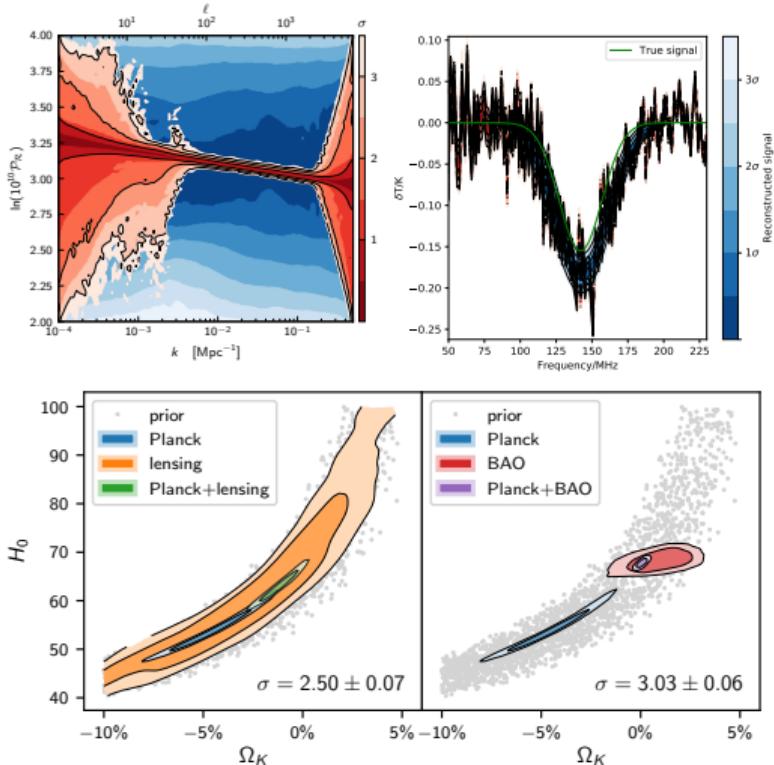
## Chain-based samplers

- ▶ e.g. PolyChord, ProxNest.
- ▶ Run Markov chain starting at a live point, generating many valid (correlated) points.
- ▶ Linear  $\sim \mathcal{O}(d)$  penalty in decorrelating new live point from the original seed point.

# Applications of nested sampling

## Cosmology

- ▶ Battle-tested in Bayesian cosmology on
  - ▶ Parameter estimation: multimodal alternative to MCMC samplers.
  - ▶ Model comparison: using integration to compute the Bayesian evidence
  - ▶ Tension quantification: using deep tail sampling and suspiciousness computations.
- ▶ Plays a critical role in major cosmology pipelines: Planck, DES, KiDS, BAO, SNe.
- ▶ The default  $\Lambda$ CDM cosmology is well-tuned to have Gaussian-like posteriors for CMB data.
- ▶ Less true for alternative cosmologies/models and orthogonal datasets, so nested sampling crucial.

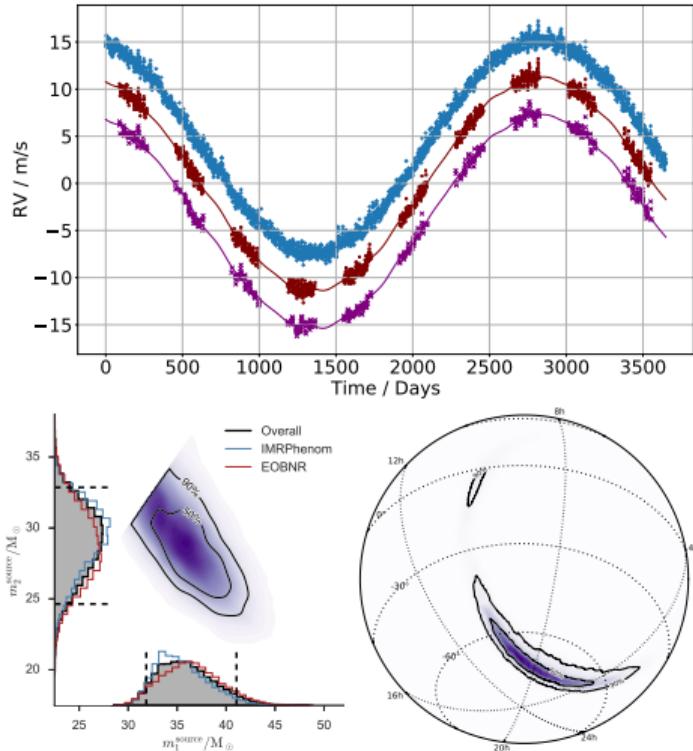


▪

# Applications of nested sampling

## Astrophysics

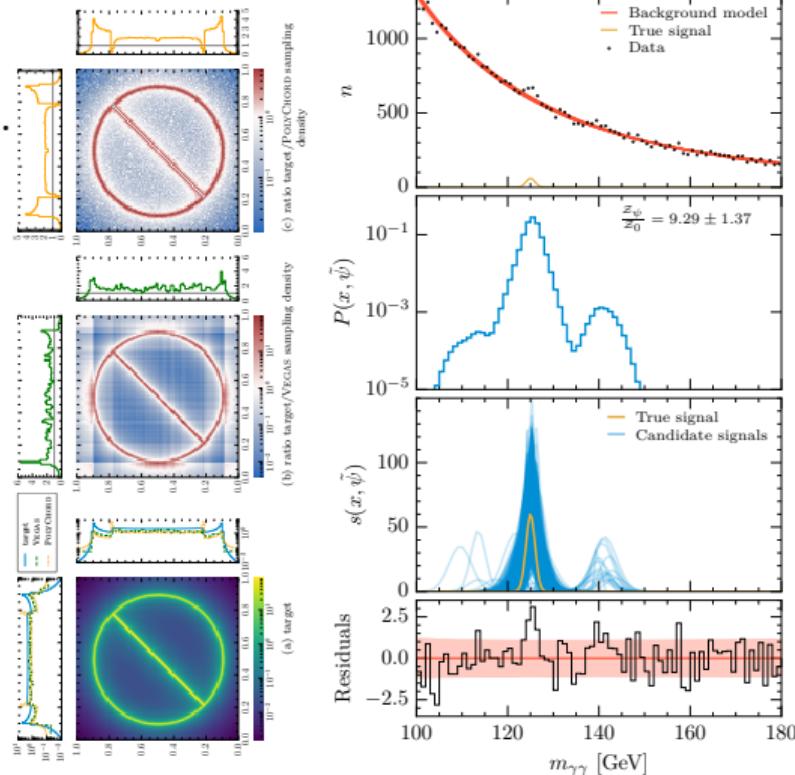
- ▶ In exoplanets [1806.00518]
  - ▶ Parameter estimation: determining properties of planets.
  - ▶ Model comparison: how many planets? Stellar modelling [2007.07278].
  - ▶ exoplanet problems regularly have posterior phase transitions [2102.03387]
- ▶ In gravitational waves
  - ▶ Parameter estimation: Binary merger properties
  - ▶ Model comparison: Modified theories of gravity, selecting phenomenological parameterisations [1803.10210]
  - ▶ Likelihood reweighting: fast slow properties



# Applications of nested sampling

## Particle physics

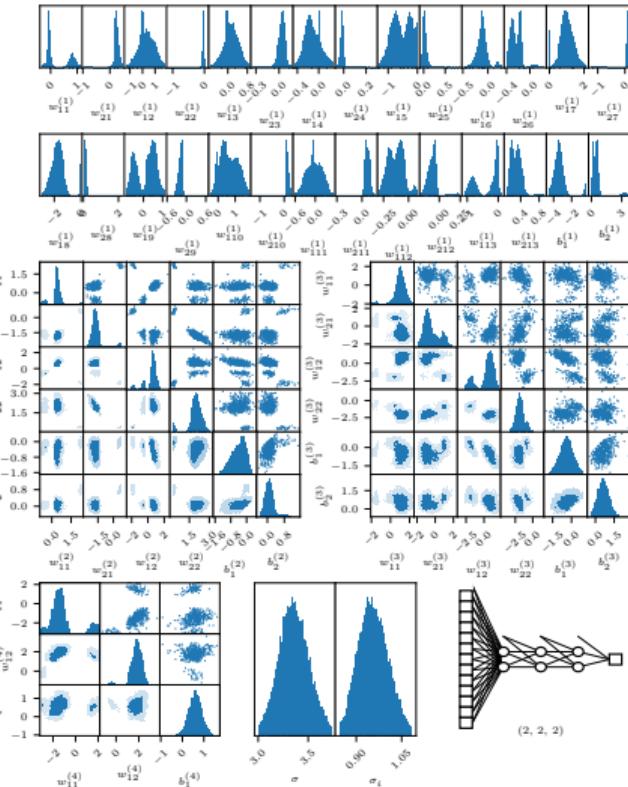
- ▶ Nested sampling for cross section computation/event generation  $\sigma = \int_{\Omega} d\Phi |\mathcal{M}|^2$ .
- ▶ Nested sampling can explore the phase space  $\Omega$  and compute integral blind with comparable efficiency to HAAG/RAMBO [2205.02030].
- ▶ Bayesian sparse reconstruction [1809.04598] applied to bump hunting allows evidence-based detection of signals in phenomenological backgrounds [2211.10391].
- ▶ Now applying to lattice field theory, and lattice gravity Lagrangians.
- ▶ Fine tuning quantification



# Applications of nested sampling

## Machine learning

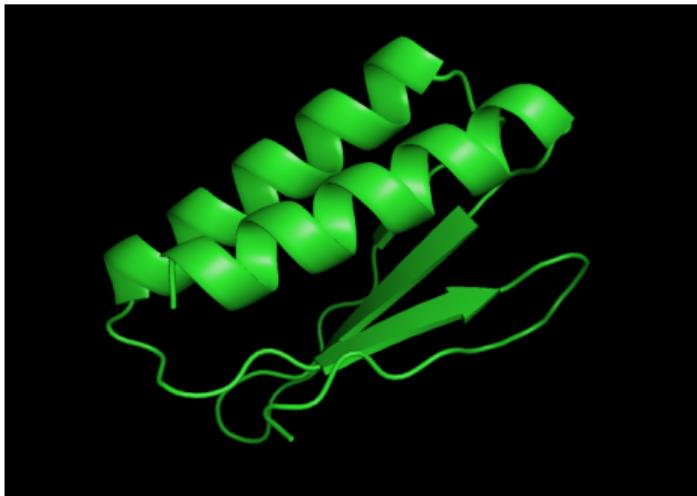
- ▶ Machine learning requires:
  - ▶ Training to find weights
  - ▶ Choice of architecture/topology/hyperparameters
- ▶ Bayesian NNs treat training as a model fitting problem
- ▶ Compute posterior of weights (parameter estimation), rather than optimisation (gradient descent)
- ▶ Use evidence to determine best architecture (model comparison), correlates with out-of-sample performance!
- ▶ Solving the full “shallow learning” problem without compromise [2004.12211][2211.10391].
- ▶ Promising work ongoing to extend this to transfer learning and deep nets.



# Applications of nested sampling

and beyond...

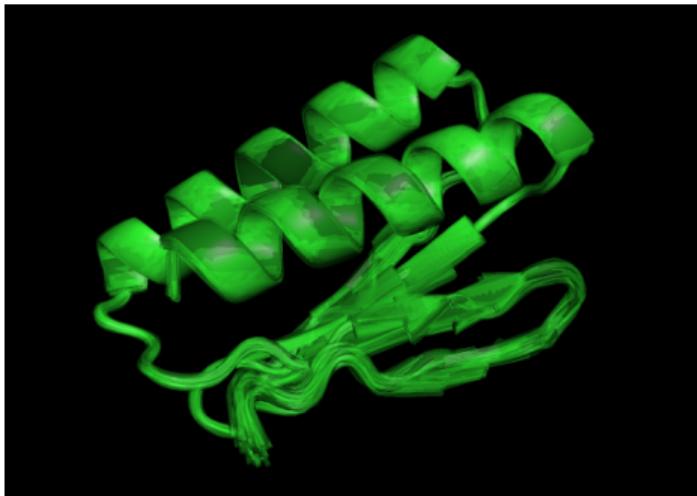
- ▶ Techniques have been spun-out (PolyChord Ltd) to:
- ▶ Protein folding
  - ▶ Navigating free energy surface.
  - ▶ Computing misfolds.
  - ▶ Thermal motion.
- ▶ Nuclear fusion reactor optimisation
  - ▶ multi-objective.
  - ▶ uncertainty propagation.
- ▶ Telecoms & DSTL research (MIDAS)
  - ▶ Optimising placement of transmitters/sensors.
  - ▶ Maximum information data acquisition strategies.



# Applications of nested sampling

and beyond...

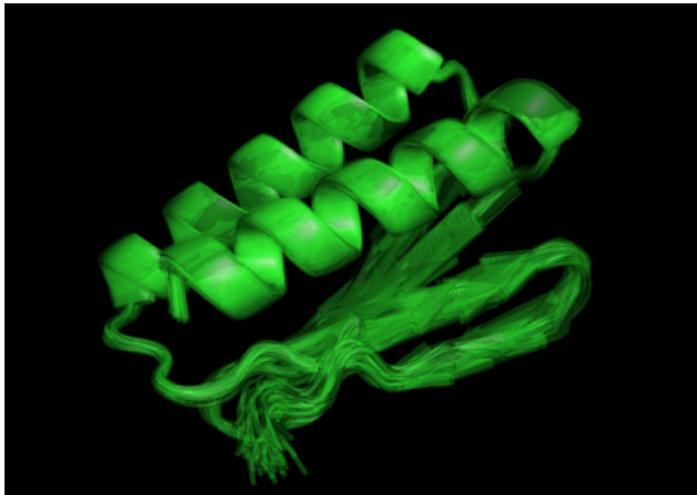
- ▶ Techniques have been spun-out (PolyChord Ltd) to:
- ▶ Protein folding
  - ▶ Navigating free energy surface.
  - ▶ Computing misfolds.
  - ▶ Thermal motion.
- ▶ Nuclear fusion reactor optimisation
  - ▶ multi-objective.
  - ▶ uncertainty propagation.
- ▶ Telecoms & DSTL research (MIDAS)
  - ▶ Optimising placement of transmitters/sensors.
  - ▶ Maximum information data acquisition strategies.



# Applications of nested sampling

and beyond...

- ▶ Techniques have been spun-out (PolyChord Ltd) to:
- ▶ Protein folding
  - ▶ Navigating free energy surface.
  - ▶ Computing misfolds.
  - ▶ Thermal motion.
- ▶ Nuclear fusion reactor optimisation
  - ▶ multi-objective.
  - ▶ uncertainty propagation.
- ▶ Telecoms & DSTL research (MIDAS)
  - ▶ Optimising placement of transmitters/sensors.
  - ▶ Maximum information data acquisition strategies.



# How does Nested Sampling compare to other approaches?

- ▶ In all cases:
    - + NS can handle multimodal functions
    - + NS computes evidences, partition functions and integrals
    - + NS is self-tuning/black-box
- Modern Nested Sampling algorithms can do this in  $\sim \mathcal{O}(100s)$  dimensions

## Optimisation

- ▶ Gradient descent
  - NS cannot use gradients
  - + NS does not require gradients
- ▶ Genetic algorithms
  - + NS discarded points have statistical meaning

## Sampling

- ▶ Metropolis-Hastings?
  - Nothing beats well-tuned customised MH
  - + NS is self tuning
- ▶ Hamiltonian Monte Carlo?
  - In millions of dimensions, HMC is king
  - + NS does not require gradients

## Integration

- ▶ Thermodynamic integration
  - protective against phase transitions
  - + No annealing schedule tuning
- ▶ Sequential Monte Carlo
  - SMC experts classify NS as a kind of SMC
  - + NS is athermal

# Advantages and disadvantages of nested sampling

## Advantages

- ▶ Doesn't need gradients
- ▶ Ensemble sampler
- ▶ Multimodal exploration
- ▶ Very Parallelisable

## Disadvantages

- ▶ Doesn't use gradients
- ▶ Slow (but steady)
- ▶ Struggles with stochastic likelihoods (nondeterminism)
- ▶ Limited to  $\sim \mathcal{O}(10^3)$  dimensions

## Unique elements

- ▶ Ensemble sampler
- ▶ Estimates volumes
- ▶ Athermal evolution
- ▶ Order statistics

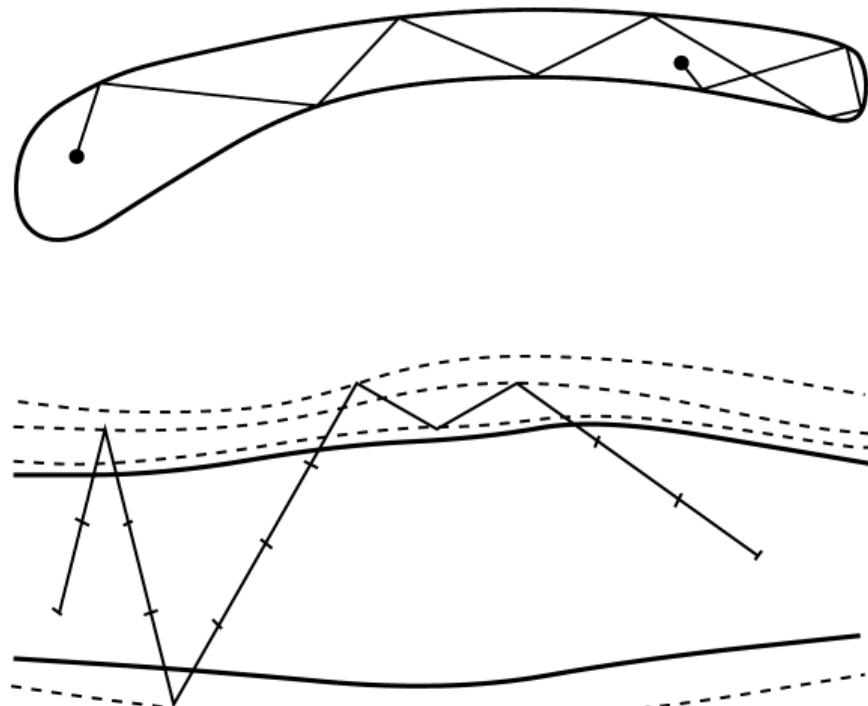
## Why doesn't HMC work?

- ▶ Nested sampling requires you to sample from the truncated *prior*, not the likelihood
- ▶ Other than at the boundaries, it is not obvious how to use a likelihood gradient to navigate the prior.
- ▶ In addition, since nested sampling begins in the tails, proceeding through the typical set and onto the peak, the normalisation of the gradient is normally not very useful.

# Constrained Hamiltonian Monte Carlo [1005.0157]

aka: Gailean nested sampling [1312.5638]; Reflective slice sampling [physics/0009028]

- ▶ The primary way to sample from “Tabletop” distributions is with reflection:
  - ▶ Define start  $x_0$ , velocity  $v$ ,
  - ▶ Update  $x_{i+1} = x_i + v\Delta t$
  - ▶ When you reach the edge, reflect using  $\hat{n}$ :  
 $v \rightarrow v - 2(v \cdot \hat{n})\hat{n}$
  - ▶  $n$  can be taken to be  $\nabla \log P$
- ▶ In practice since don’t know the exact boundary, care needs to be taken to generate unbiased samples
  - ▶ e.g. by reflecting whenever one is outside, not just once to get us back in.
- ▶ Radford Neal [physics/0009028] section 7 is best reference for this.



## Historical attempts

[Betancourt](#) Hamiltonian constrained nested sampling [1005.0157].

[Feroz](#) Galilean Nested Sampling [1312.5638].

[Speagle](#) Incorporated into dynesty [1904.02180].

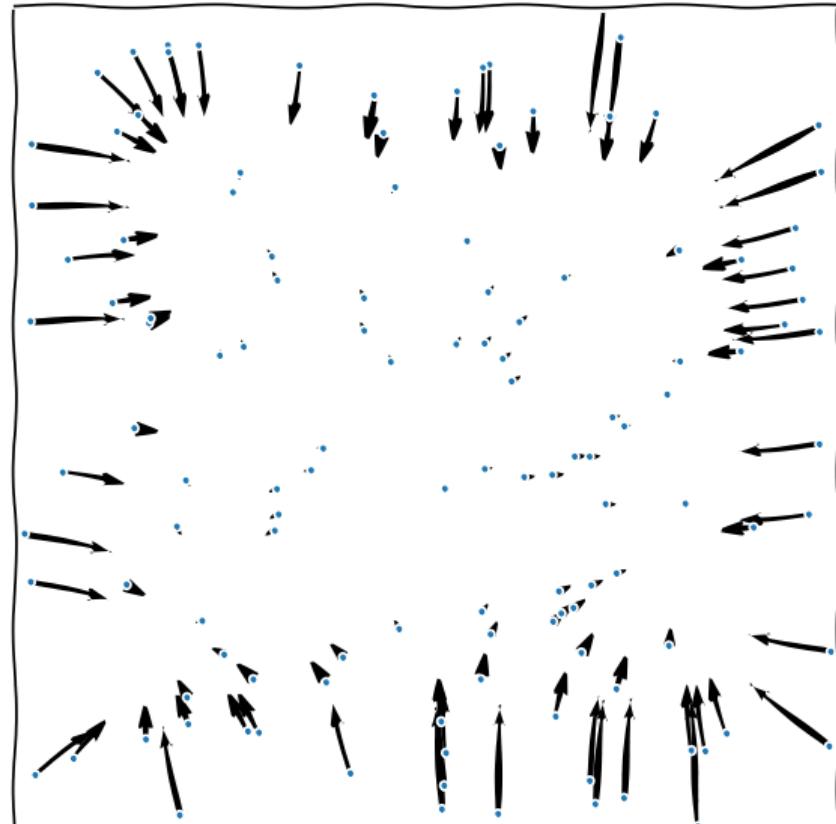
[Habeck](#) Habeck Demonic nested sampling – uses thermodynamic analogy to soften the hard boundary with a Maxwell daemon [doi:10.1063/1.4905971].

[Baldock](#) Total Enthalpy HMC, incorporating momenta in a more HMC like way, but specialised to materials science [1710.11085].

[Cai](#) ProxNest for high-dimensional convex imaging problems [2106.03646].

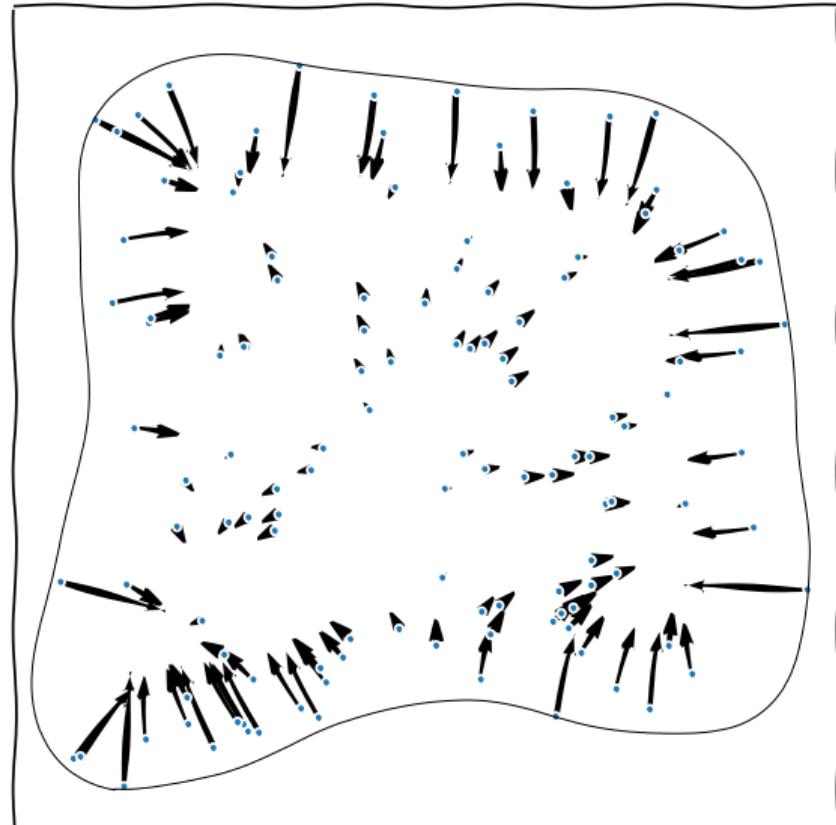
## Nested sampling with gradients

- ▶ Existing techniques don't make use of:
- ▶ We have "cloud" of gradients at every point
- ▶ We have an estimate of the prior volume  $X$
- ▶  $\nabla \log X \propto \nabla \log L$



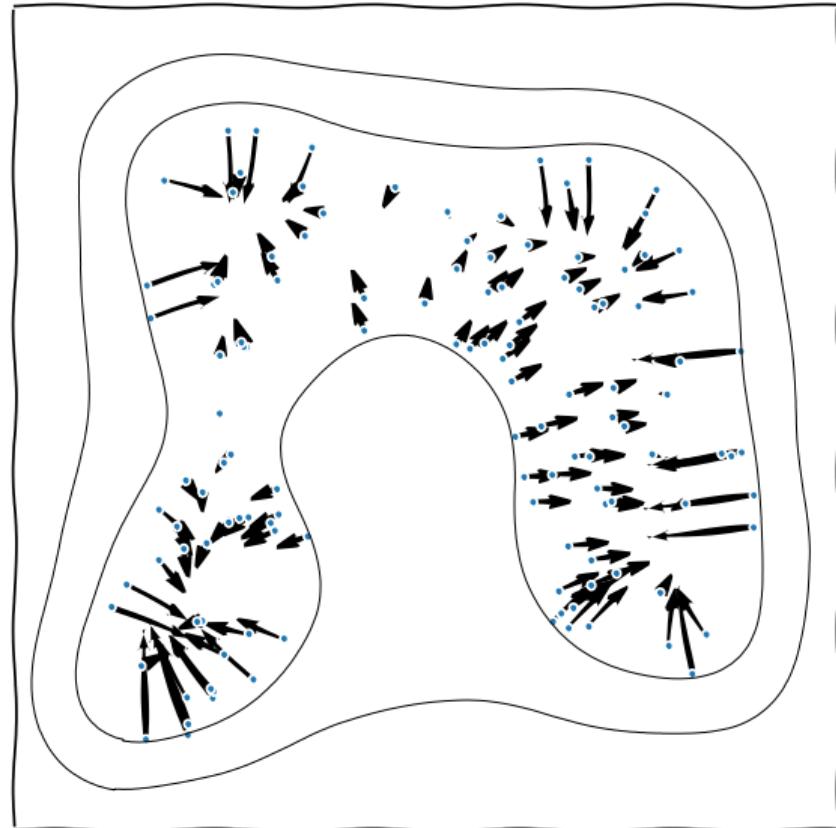
## Nested sampling with gradients

- ▶ Existing techniques don't make use of:
- ▶ We have "cloud" of gradients at every point
- ▶ We have an estimate of the prior volume  $X$
- ▶  $\nabla \log X \propto \nabla \log L$



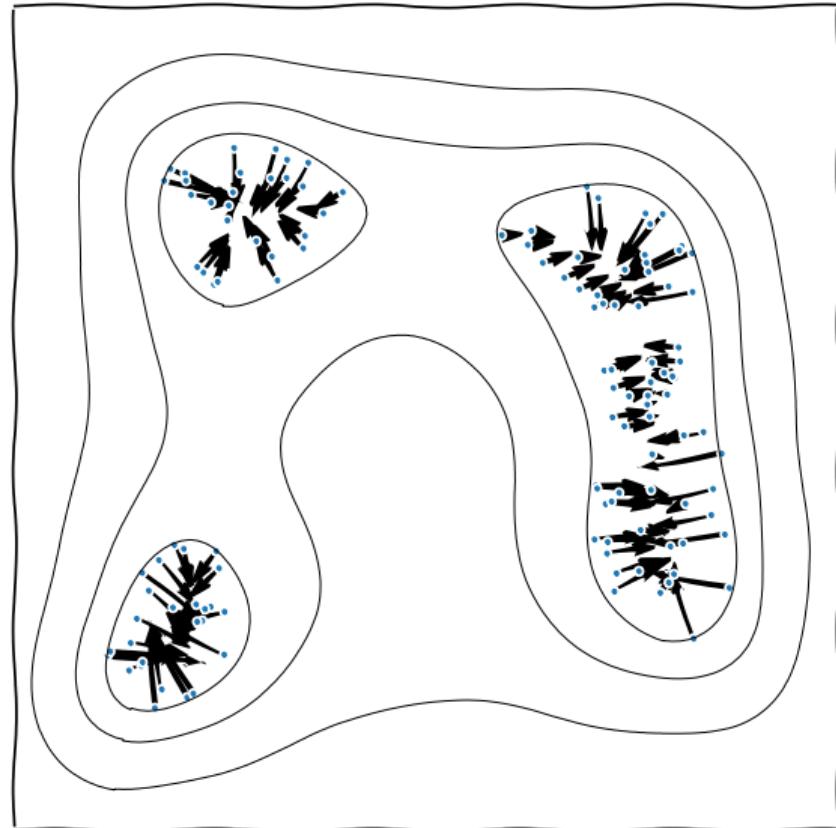
## Nested sampling with gradients

- ▶ Existing techniques don't make use of:
- ▶ We have "cloud" of gradients at every point
- ▶ We have an estimate of the prior volume  $X$
- ▶  $\nabla \log X \propto \nabla \log L$



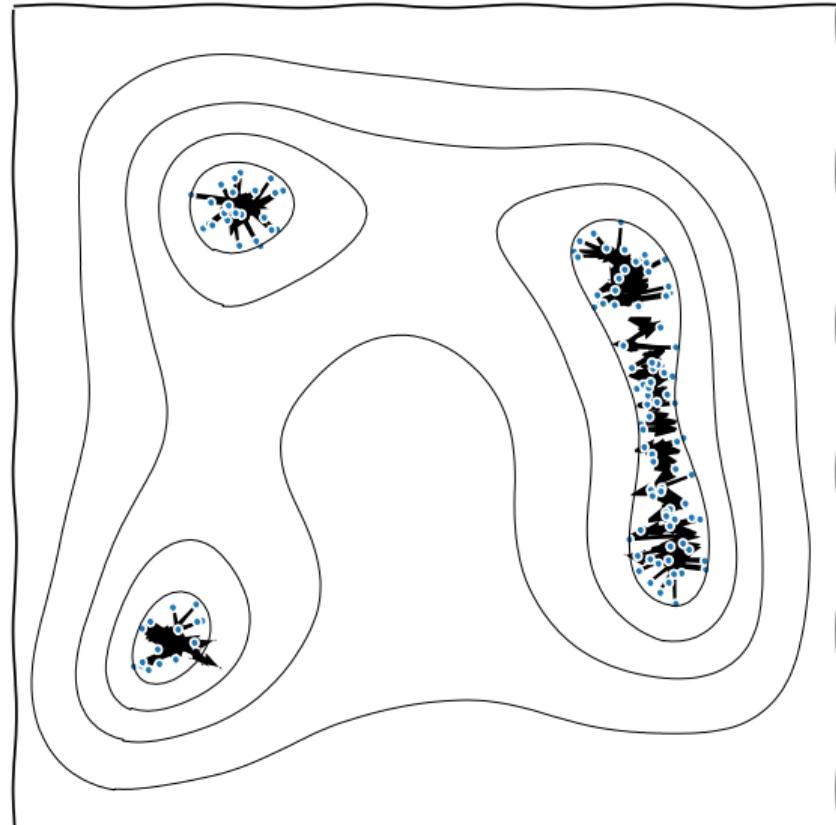
## Nested sampling with gradients

- ▶ Existing techniques don't make use of:
- ▶ We have "cloud" of gradients at every point
- ▶ We have an estimate of the prior volume  $X$
- ▶  $\nabla \log X \propto \nabla \log L$



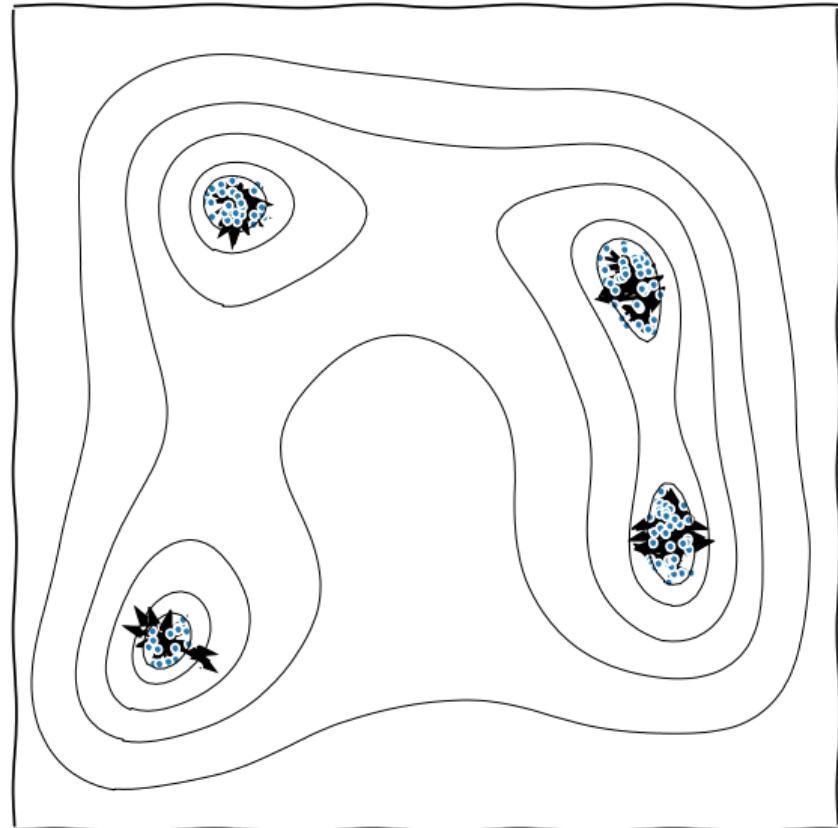
## Nested sampling with gradients

- ▶ Existing techniques don't make use of:
- ▶ We have "cloud" of gradients at every point
- ▶ We have an estimate of the prior volume  $X$
- ▶  $\nabla \log X \propto \nabla \log L$



## Nested sampling with gradients

- ▶ Existing techniques don't make use of:
- ▶ We have "cloud" of gradients at every point
- ▶ We have an estimate of the prior volume  $X$
- ▶  $\nabla \log X \propto \nabla \log L$



# Conclusions

- ▶ Nested sampling is a robust, multi-purpose numerical tool for:
  - ▶ Numerical integration  $\int f(x)dV$ ,
  - ▶ Exploring/scanning/optimising *a priori* unknown functions,
  - ▶ Performing Bayesian inference and model comparison.
- ▶ It is applied widely across a variety of fields
- ▶ It can't currently use gradients very effectively.
- ▶ If it could, probabilistic programming means there are a lot of analyses which would benefit.