Nested Sampling and Likelihood Free Inference

Will Handley
<wh260@cam.ac.uk>

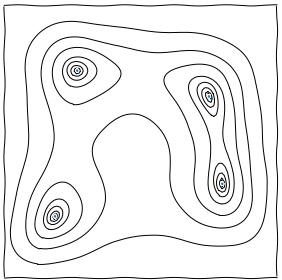
Royal Society University Research Fellow & Turing Fellow
Astrophysics Group, Cavendish Laboratory, University of Cambridge
Kavli Institute for Cosmology, Cambridge
Gonville & Caius College
github.com/williamjameshandley/talks

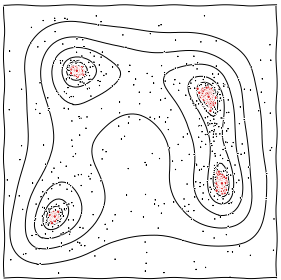
21st April 2022

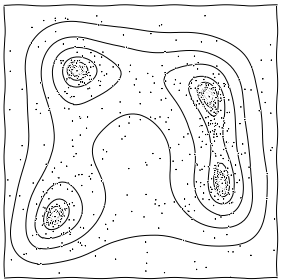

What is Nested Sampling?

- Nested sampling is a multi-purpose numerical mathematical tool.
- Given a (scalar) function f with a vector of parameters θ, it can be used for:

Optimisation	Sampling	Integration
$\theta_{\max} = \max_{\theta} f(\theta)$	draw $\theta \sim f$	$\int f(\theta) dV$







MCMC sampling

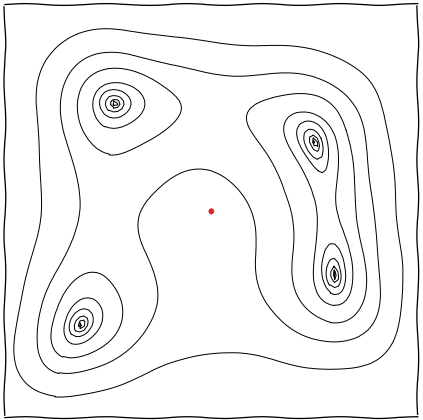
- Markov chain based methods generate samples from posterior distribution by a stepping procedure
- This can get stuck in local peaks
- Cannot compute normalisation $\mathcal{Z}$ of Bayes theorem:

$$P(\theta|D,M) = \frac{P(D|\theta,M)P(\theta|M)}{P(D|M)}$$

$$\mathcal{P} = \frac{\mathcal{L} \times \pi}{\mathcal{Z}} \qquad \text{posterior} = \frac{\text{likelihood} \times \text{prior}}{\text{evidence}}$$

- We generally want the evidence $\mathcal{Z} = P(D|M)$ for the second stage of inference: model comparison

$$P(M|D) = \frac{P(D|M)P(M)}{P(D)} \qquad \text{Science}(M) = \frac{\mathcal{Z}_M \Pi_M}{\sum_m \mathcal{Z}_m \Pi_m}$$



MCMC sampling

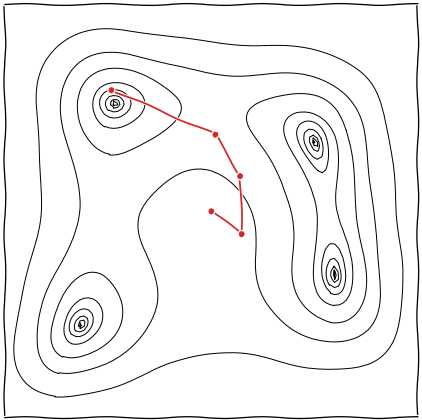
- Markov chain based methods generate samples from posterior distribution by a stepping procedure
- This can get stuck in local peaks
- Cannot compute normalisation $\mathcal{Z}$ of Bayes theorem:

$$P(\theta|D,M) = \frac{P(D|\theta,M)P(\theta|M)}{P(D|M)}$$

$$\mathcal{P} = \frac{\mathcal{L}\times\pi}{\mathcal{Z}} \qquad \text{posterior} = \frac{\text{likelihood}\times\text{prior}}{\text{evidence}}$$

- We generally want the evidence $\mathcal{Z} = P(D|M)$ for the second stage of inference: model comparison

$$P(M|D) = \frac{P(D|M)P(M)}{P(D)} \qquad \text{Science}(M) = \frac{\mathcal{Z}_M\Pi_M}{\sum_m \mathcal{Z}_m\Pi_m}$$



MCMC sampling

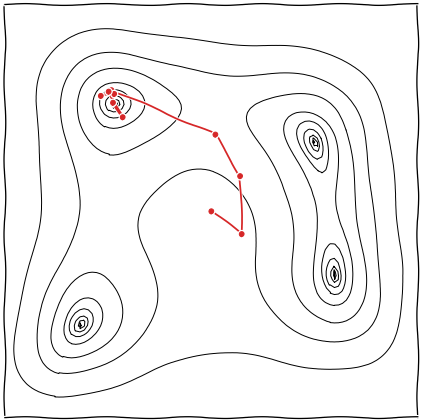
- Markov chain based methods generate samples from posterior distribution by a stepping procedure
- This can get stuck in local peaks
- Cannot compute normalisation $\mathcal{Z}$ of Bayes theorem:

$$P(\theta|D,M) = \frac{P(D|\theta,M)P(\theta|M)}{P(D|M)}$$

$$\mathcal{P} = \frac{\mathcal{L}\times\pi}{\mathcal{Z}} \qquad \text{posterior} = \frac{\text{likelihood}\times\text{prior}}{\text{evidence}}$$

- We generally want the evidence $\mathcal{Z} = P(D|M)$ for the second stage of inference: model comparison

$$P(M|D) = \frac{P(D|M)P(M)}{P(D)} \qquad \text{Science}(M) = \frac{\mathcal{Z}_M\Pi_M}{\sum_m \mathcal{Z}_m\Pi_m}$$



MCMC sampling

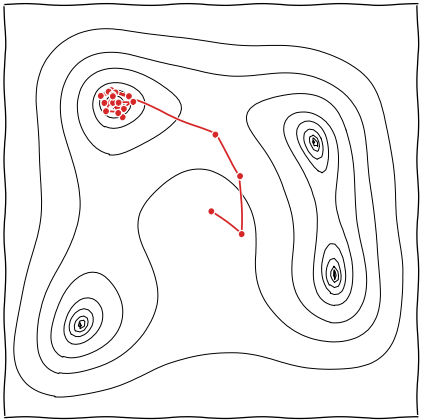
- Markov chain based methods generate samples from posterior distribution by a stepping procedure
- This can get stuck in local peaks
- Cannot compute normalisation $\mathcal{Z}$ of Bayes theorem:

$$P(\theta|D,M) = \frac{P(D|\theta,M)P(\theta|M)}{P(D|M)}$$

$$\mathcal{P} = \frac{\mathcal{L}\times\pi}{\mathcal{Z}} \qquad \text{posterior} = \frac{\text{likelihood}\times\text{prior}}{\text{evidence}}$$

- We generally want the evidence $\mathcal{Z} = P(D|M)$ for the second stage of inference: model comparison

$$P(M|D) = \frac{P(D|M)P(M)}{P(D)} \qquad \text{Science}(M) = \frac{\mathcal{Z}_M\Pi_M}{\sum_m \mathcal{Z}_m\Pi_m}$$



MCMC sampling

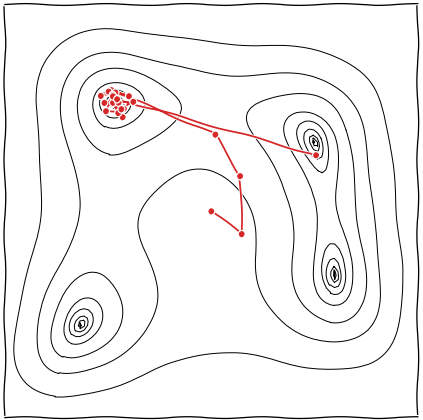
- Markov chain based methods generate samples from posterior distribution by a stepping procedure
- This can get stuck in local peaks
- Cannot compute normalisation $\mathcal{Z}$ of Bayes theorem:

$$P(\theta|D,M) = \frac{P(D|\theta,M)P(\theta|M)}{P(D|M)}$$

$$\mathcal{P} = \frac{\mathcal{L}\times\pi}{\mathcal{Z}} \qquad \text{posterior} = \frac{\text{likelihood}\times\text{prior}}{\text{evidence}}$$

- We generally want the evidence $\mathcal{Z} = P(D|M)$ for the second stage of inference: model comparison

$$P(M|D) = \frac{P(D|M)P(M)}{P(D)} \qquad \text{Science}(M) = \frac{\mathcal{Z}_M\Pi_M}{\sum_m \mathcal{Z}_m\Pi_m}$$



MCMC sampling

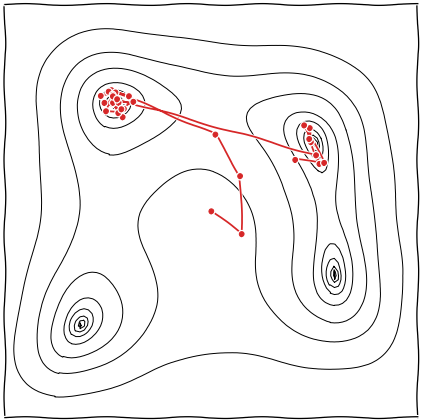
- Markov chain based methods generate samples from posterior distribution by a stepping procedure
- This can get stuck in local peaks
- Cannot compute normalisation $\mathcal{Z}$ of Bayes theorem:

$$P(\theta|D,M) = \frac{P(D|\theta,M)P(\theta|M)}{P(D|M)}$$

$$\mathcal{P} = \frac{\mathcal{L} \times \pi}{\mathcal{Z}} \qquad \text{posterior} = \frac{\text{likelihood} \times \text{prior}}{\text{evidence}}$$

- We generally want the evidence $\mathcal{Z} = P(D|M)$ for the second stage of inference: model comparison

$$P(M|D) = \frac{P(D|M)P(M)}{P(D)} \qquad \text{Science}(M) = \frac{\mathcal{Z}_M \Pi_M}{\sum_m \mathcal{Z}_m \Pi_m}$$



Nested sampling

- Nested sampling: completely different way to sample
- Ensemble sampling to compress prior to posterior.
- Sequentially update a set S of n samples:

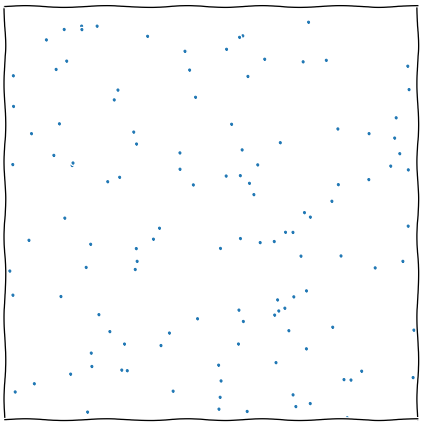
  S_0: Generate n samples uniformly over the space (from the prior π).

  S_{n+1}: Delete the lowest likelihood sample in S_n, 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*:

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

- This procedure optimises (multimodally), and can calculate the evidence & posterior weights.



Nested sampling

- Nested sampling: completely different way to sample
- Ensemble sampling to compress prior to posterior.
- Sequentially update a set S of n samples:

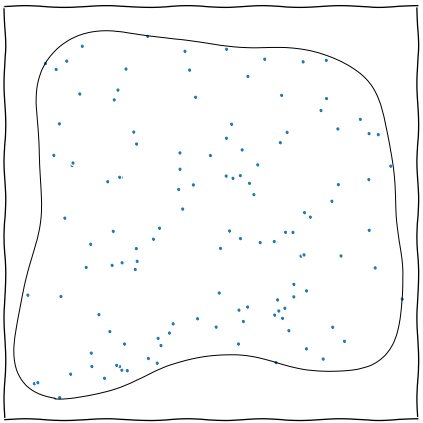
  S_0: Generate n samples uniformly over the space (from the prior π).

  S_{n+1}: Delete the lowest likelihood sample in S_n, 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*:

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

- This procedure optimises (multimodally), and can calculate the evidence & posterior weights.



Nested sampling

- Nested sampling: completely different way to sample
- Ensemble sampling to compress prior to posterior.
- Sequentially update a set S of n samples:

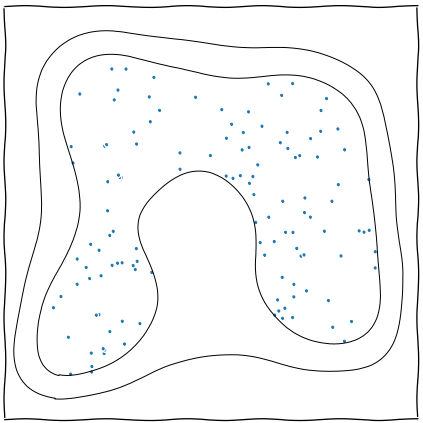
  S_0: Generate n samples uniformly over the space (from the prior π).

  S_{n+1}: Delete the lowest likelihood sample in S_n, 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*:

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

- This procedure optimises (multimodally), and can calculate the evidence & posterior weights.



Nested sampling

- Nested sampling: completely different way to sample
- Ensemble sampling to compress prior to posterior.
- Sequentially update a set S of n samples:

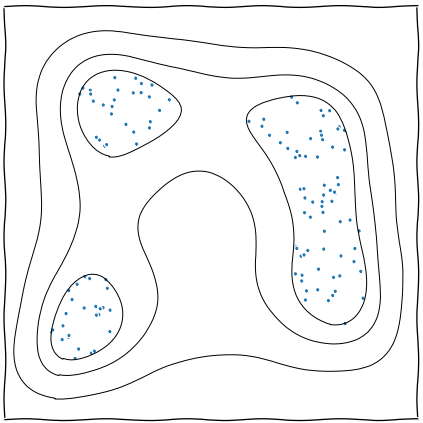
  S_0: Generate n samples uniformly over the space (from the prior π).

  S_{n+1}: Delete the lowest likelihood sample in S_n, 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*:

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

- This procedure optimises (multimodally), and can calculate the evidence & posterior weights.



Nested sampling

- Nested sampling: completely different way to sample
- Ensemble sampling to compress prior to posterior.
- Sequentially update a set S of n samples:

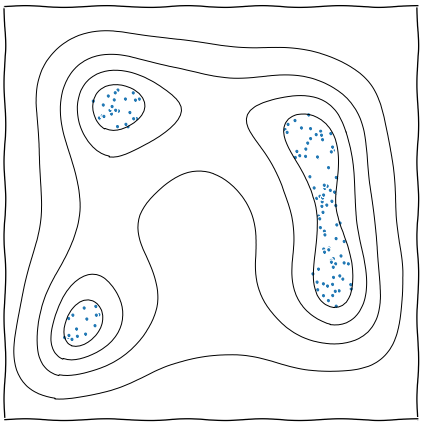
  S_0: Generate n samples uniformly over the space (from the prior π).

  S_{n+1}: Delete the lowest likelihood sample in S_n, 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*:

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

- This procedure optimises (multimodally), and can calculate the evidence & posterior weights.



Nested sampling

- Nested sampling: completely different way to sample
- Ensemble sampling to compress prior to posterior.
- Sequentially update a set S of n samples:

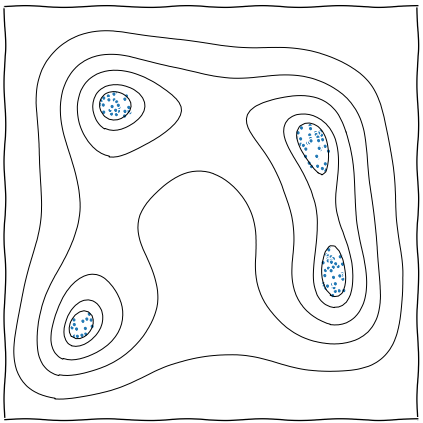
  S_0: Generate n samples uniformly over the space (from the prior π).

  S_{n+1}: Delete the lowest likelihood sample in S_n, 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*:

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

- This procedure optimises (multimodally), and can calculate the evidence & posterior weights.



Nested sampling

- Nested sampling: completely different way to sample
- Ensemble sampling to compress prior to posterior.
- Sequentially update a set S of n samples:

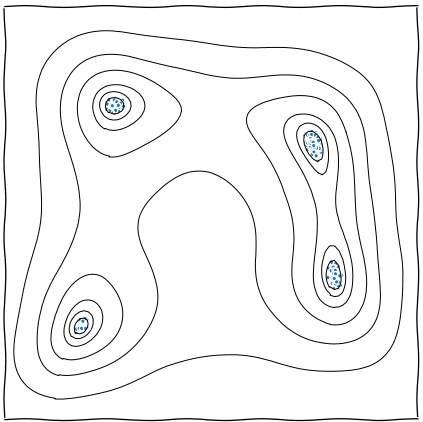
  S_0: Generate n samples uniformly over the space (from the prior π).

  S_{n+1}: Delete the lowest likelihood sample in S_n, 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*:

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

- This procedure optimises (multimodally), and can calculate the evidence & posterior weights.



Nested sampling

- Nested sampling: completely different way to sample
- Ensemble sampling to compress prior to posterior.
- Sequentially update a set S of n samples:

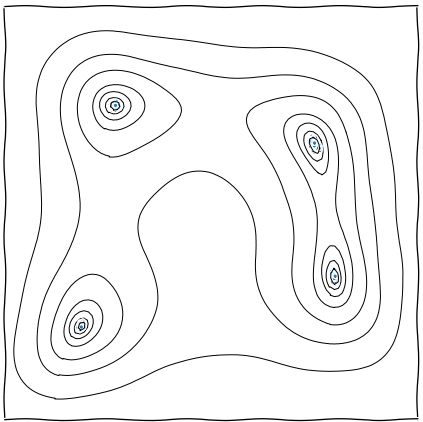
  S_0: Generate n samples uniformly over the space (from the prior π).

  S_{n+1}: Delete the lowest likelihood sample in S_n, 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*:

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

- This procedure optimises (multimodally), and can calculate the evidence & posterior weights.



Mathematics of Nested Sampling

A probabilistic Lebesgque integrator

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

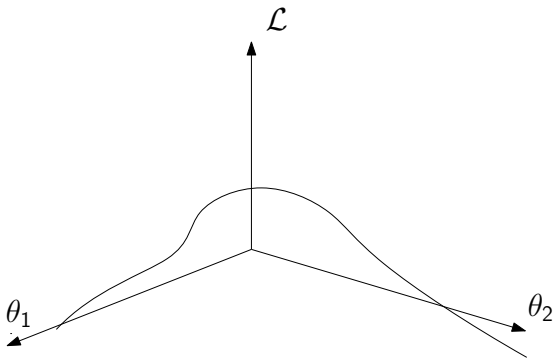
$$\mathcal{Z} \approx \sum_i \Delta\mathcal{L}_i X_i, \quad X_{i+1} \approx \frac{n}{n+1} X_i, \quad X_0 = 1$$

- Although this is only approximate, we can quantify the error

$$P(X_i|X_{i-1}) = \frac{X_i^{n-1}}{nX_{i-1}^n} \times [0 < X_i < X_{i-1}]$$

- Integral can be expressed in one of two ways

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



Mathematics of Nested Sampling

A probabilistic Lebesgque integrator

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

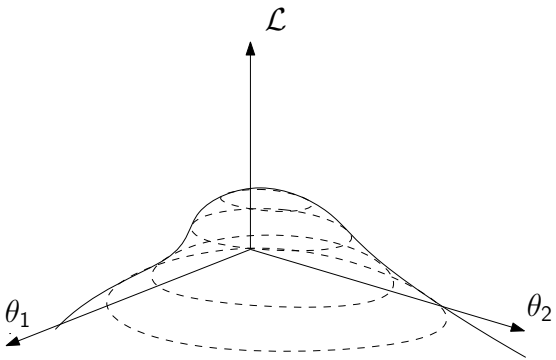
$$\mathcal{Z} \approx \sum_i \Delta\mathcal{L}_i X_i, \quad X_{i+1} \approx \frac{n}{n+1} X_i, \quad X_0 = 1$$

- Although this is only approximate, we can quantify the error

$$P(X_i|X_{i-1}) = \frac{X_i^{n-1}}{nX_{i-1}^n} \times [0 < X_i < X_{i-1}]$$

- Integral can be expressed in one of two ways

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



Mathematics of Nested Sampling

A probabilistic Lebesgque integrator

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

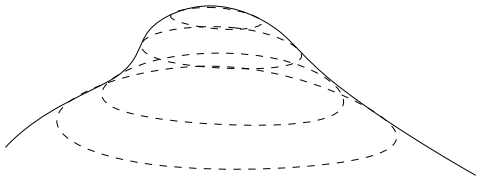
$$\mathcal{Z} \approx \sum_i \Delta\mathcal{L}_i X_i, \quad X_{i+1} \approx \frac{n}{n+1} X_i, \quad X_0 = 1$$

- Although this is only approximate, we can quantify the error

$$P(X_i|X_{i-1}) = \frac{X_i^{n-1}}{nX_{i-1}^n} \times [0 < X_i < X_{i-1}]$$

- Integral can be expressed in one of two ways

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



Mathematics of Nested Sampling

A probabilistic Lebesgque integrator

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

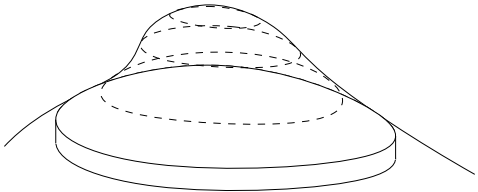
$$\mathcal{Z} \approx \sum_i \Delta\mathcal{L}_i X_i, \quad X_{i+1} \approx \frac{n}{n+1} X_i, \quad X_0 = 1$$

- Although this is only approximate, we can quantify the error

$$P(X_i|X_{i-1}) = \frac{X_i^{n-1}}{nX_{i-1}^n} \times [0 < X_i < X_{i-1}]$$

- Integral can be expressed in one of two ways

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



Mathematics of Nested Sampling

A probabilistic Lebesgque integrator

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

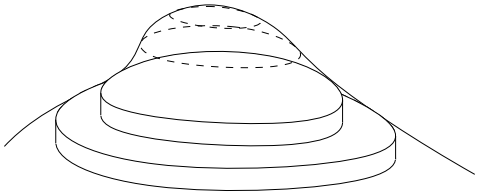
$$\mathcal{Z} \approx \sum_i \Delta\mathcal{L}_i X_i, \quad X_{i+1} \approx \frac{n}{n+1} X_i, \quad X_0 = 1$$

- Although this is only approximate, we can quantify the error

$$P(X_i|X_{i-1}) = \frac{X_i^{n-1}}{nX_{i-1}^n} \times [0 < X_i < X_{i-1}]$$

- Integral can be expressed in one of two ways

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



Mathematics of Nested Sampling

A probabilistic Lebesgque integrator

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

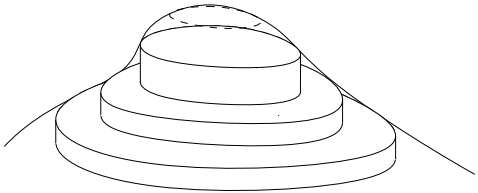
$$\mathcal{Z} \approx \sum_i \Delta\mathcal{L}_i X_i, \quad X_{i+1} \approx \frac{n}{n+1} X_i, \quad X_0 = 1$$

- Although this is only approximate, we can quantify the error

$$P(X_i|X_{i-1}) = \frac{X_i^{n-1}}{nX_{i-1}^n} \times [0 < X_i < X_{i-1}]$$

- Integral can be expressed in one of two ways

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



Mathematics of Nested Sampling

A probabilistic Lebesgque integrator

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

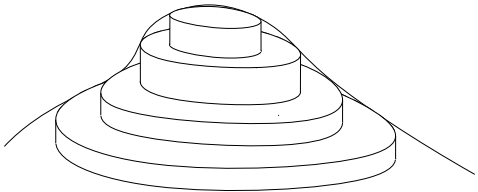
$$\mathcal{Z} \approx \sum_i \Delta\mathcal{L}_i X_i, \quad X_{i+1} \approx \frac{n}{n+1} X_i, \quad X_0 = 1$$

- Although this is only approximate, we can quantify the error

$$P(X_i|X_{i-1}) = \frac{X_i^{n-1}}{nX_{i-1}^n} \times [0 < X_i < X_{i-1}]$$

- Integral can be expressed in one of two ways

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



Mathematics of Nested Sampling

A probabilistic Lebesgque integrator

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

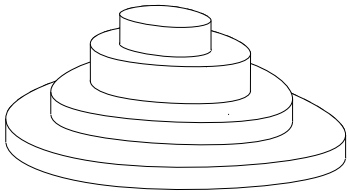
$$\mathcal{Z} \approx \sum_i \Delta\mathcal{L}_i X_i, \quad X_{i+1} \approx \frac{n}{n+1} X_i, \quad X_0 = 1$$

- Although this is only approximate, we can quantify the error

$$P(X_i|X_{i-1}) = \frac{X_i^{n-1}}{nX_{i-1}^n} \times [0 < X_i < X_{i-1}]$$

- Integral can be expressed in one of two ways

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

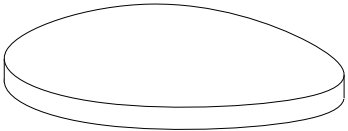


Mathematics of Nested Sampling

A probabilistic Lebesgque integrator

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

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



- Although this is only approximate, we can quantify the error

$$P(X_i|X_{i-1}) = \frac{X_i^{n-1}}{nX_{i-1}^n} \times [0 < X_i < X_{i-1}]$$

- Integral can be expressed in one of two ways

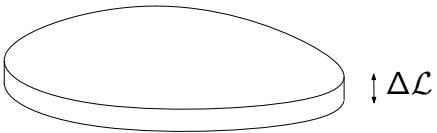
$$\mathcal{Z} \approx \sum_i \Delta\mathcal{L}_i X_i = \sum_i \mathcal{L}_i \Delta X_i$$

Mathematics of Nested Sampling

A probabilistic Lebesgque integrator

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

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



- Although this is only approximate, we can quantify the error

$$P(X_i|X_{i-1}) = \frac{X_i^{n-1}}{nX_{i-1}^n} \times [0 < X_i < X_{i-1}]$$

- Integral can be expressed in one of two ways

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

Mathematics of Nested Sampling

A probabilistic Lebesgque integrator

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

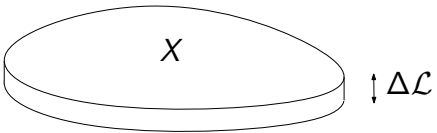
$$\mathcal{Z} \approx \sum_i \Delta\mathcal{L}_i X_i, \quad X_{i+1} \approx \frac{n}{n+1} X_i, \quad X_0 = 1$$

- Although this is only approximate, we can quantify the error

$$P(X_i|X_{i-1}) = \frac{X_i^{n-1}}{nX_{i-1}^n} \times [0 < X_i < X_{i-1}]$$

- Integral can be expressed in one of two ways

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



Mathematics of Nested Sampling

A probabilistic Lebesgque integrator

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

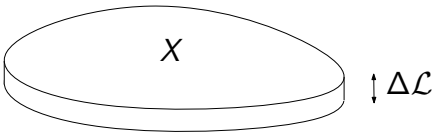
$$\mathcal{Z} \approx \sum_i \Delta\mathcal{L}_i X_i, \quad X_{i+1} \approx \frac{n}{n+1} X_i, \quad X_0 = 1$$

- Although this is only approximate, we can quantify the error

$$P(X_i|X_{i-1}) = \frac{X_i^{n-1}}{nX_{i-1}^n} \times [0 < X_i < X_{i-1}]$$

- Integral can be expressed in one of two ways

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



Volume $= X\Delta\mathcal{L}$

Mathematics of Nested Sampling

A probabilistic Lebesgque integrator

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

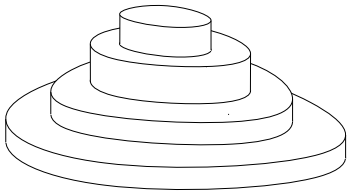
$$\mathcal{Z} \approx \sum_i \Delta\mathcal{L}_i X_i, \quad X_{i+1} \approx \frac{n}{n+1} X_i, \quad X_0 = 1$$

- Although this is only approximate, we can quantify the error

$$P(X_i|X_{i-1}) = \frac{X_i^{n-1}}{nX_{i-1}^n} \times [0 < X_i < X_{i-1}]$$

- Integral can be expressed in one of two ways

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



Mathematics of Nested Sampling

A probabilistic Lebesgque integrator

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

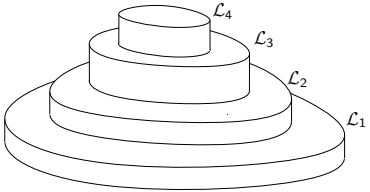
$$\mathcal{Z} \approx \sum_i \Delta\mathcal{L}_i X_i, \quad X_{i+1} \approx \frac{n}{n+1} X_i, \quad X_0 = 1$$

- Although this is only approximate, we can quantify the error

$$P(X_i|X_{i-1}) = \frac{X_i^{n-1}}{nX_{i-1}^n} \times [0 < X_i < X_{i-1}]$$

- Integral can be expressed in one of two ways

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



Mathematics of Nested Sampling

A probabilistic Lebesgque integrator

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

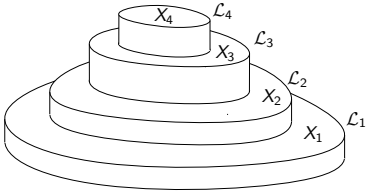
$$\mathcal{Z} \approx \sum_i \Delta\mathcal{L}_i X_i, \quad X_{i+1} \approx \frac{n}{n+1} X_i, \quad X_0 = 1$$

- Although this is only approximate, we can quantify the error

$$P(X_i|X_{i-1}) = \frac{X_i^{n-1}}{nX_{i-1}^n} \times [0 < X_i < X_{i-1}]$$

- Integral can be expressed in one of two ways

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



Mathematics of Nested Sampling

A probabilistic Lebesgque integrator

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

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

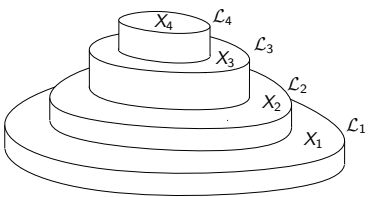
- Although this is only approximate, we can quantify the error

$$P(X_i|X_{i-1}) = \frac{X_i^{n-1}}{nX_{i-1}^n} \times [0 < X_i < X_{i-1}]$$

- Integral can be expressed in one of two ways

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

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

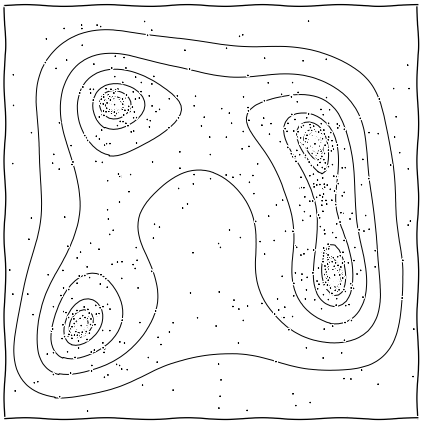


Dead points: posteriors & evidences

- At the end, one is left with a set of discarded points
- These may be weighted to form weighted posterior samples using $w_i = \mathcal{L}_i \Delta X_i$
- They can also be used to calculate the normalisation $\mathcal{Z} = \sum \mathcal{L}_i \Delta X_i$, or more generally $\sum_i f(\mathcal{L}_i) \Delta X_i$.
  - Nested sampling probabilistically estimates the volume of the parameter space

$$X_i \approx \left(\frac{n}{n+1}\right) X_{i-1} \quad \Rightarrow \quad X_i \approx \left(\frac{n}{n+1}\right)^i \approx e^{-i/n}$$

  - only statistical estimates, but we know the error bar
  - Nested sampling thus estimates the density of states
  - it is therefore a partition function calculator
- The evolving ensemble of live points allows algorithms to perform self-tuning and mode clustering.

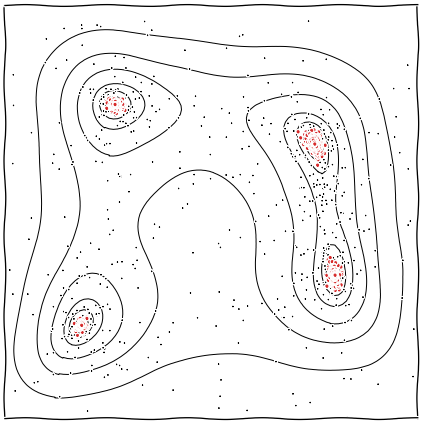


Dead points: posteriors & evidences

- At the end, one is left with a set of discarded points
- These may be weighted to form weighted posterior samples using $w_i = \mathcal{L}_i \Delta X_i$
- They can also be used to calculate the normalisation $\mathcal{Z} = \sum \mathcal{L}_i \Delta X_i$, or more generally $\sum_i f(\mathcal{L}_i) \Delta X_i$.
  - Nested sampling probabilistically estimates the volume of the parameter space

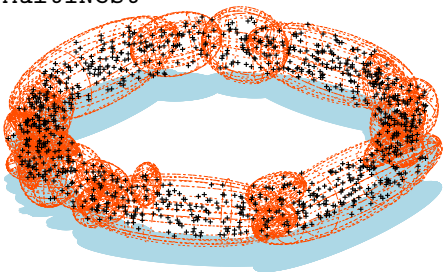
$$X_i \approx \left(\frac{n}{n+1}\right) X_{i-1} \quad \Rightarrow \quad X_i \approx \left(\frac{n}{n+1}\right)^i \approx e^{-i/n}$$

  - only statistical estimates, but we know the error bar
  - Nested sampling thus estimates the density of states
  - it is therefore a partition function calculator
- The evolving ensemble of live points allows algorithms to perform self-tuning and mode clustering.

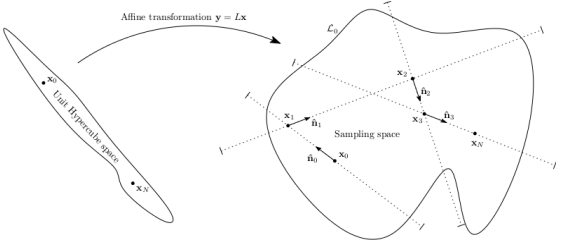


Implementations of Nested Sampling

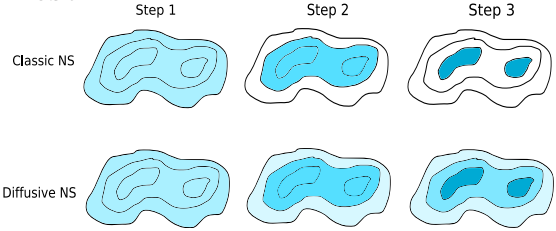
MultiNest



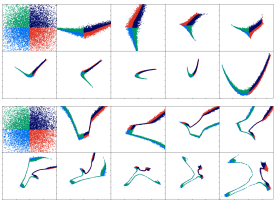
PolyChord

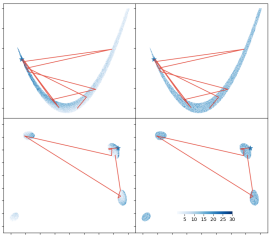


DNest



NeuralNest





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 does not require gradients
- Genetic algorithms
  - \+ NS discarded points have statistical meaning

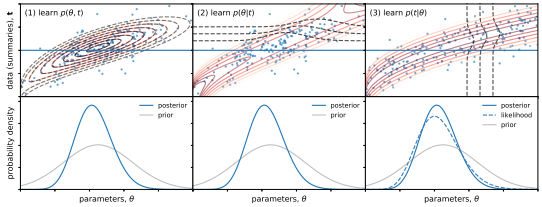
Sampling

- Metropolis-Hastings?
  - – Very little beats a 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 trasitions
  - \+ No annealing schedule tuning
- Sequential Monte Carlo
  - – Some people (SMC experts) classify NS as a kind of SMC
  - \+ NS is athermal

Nested Sampling with Likelihood Free Inference



Alsing *et al.* [1903.00007]

- In density estimation likelihood free inference, the output is to learn one/all of:

  Likelihood $P(D|\theta)$
  Posterior $P(\theta|D)$
  Joint $P(D, \theta)$

- In the first instance, nested sampling can be used to scan these learnt functions

- Data are compressed, so joint space (D, θ) is navigable by off-the-shelf codes.
  - Sanity checking the solution
  - Computing evidences/Kullback Liebler divergences from likelihoods
- Its self-tuning capacity and ability to handle multi-modal distributions can be very useful for diagnosing incompletely learnt functions
- Emulated likelihoods (e.g. normalising flows) are generally fast, so can deploy more likelihood hungry techniques like NS.
- As Pablo Lemos & David Yallup will discuss, in principle can use it to train emulators by marginalisation rather than maximisation.

Nested Sampling for Approximate Bayesian Computation/SBI

- Assume one has a generative model capable of turning parameters into mock data $D(\theta)$
- Given infinite computing power, ABC works by selecting $\{\theta : D(\theta) = D_{\text{observed}}\}$
- These are samples from the posterior, without using a likelihood.
- In practice $D = D_{\text{obs}}$ becomes $D \approx D_{\text{obs}}$
- i.e. $|D - D_{\text{obs}}| < \varepsilon$, or more generally $\boxed{\rho(D, D_{\text{obs}}) < \varepsilon}$, where ρ is some suitably chosen objective function
- Main challenges are
  1. Choice of ρ/summary stats
  2. Choice of ε schedule
  3. Rejection sampling

- Nested sampling fits this well: In principle, can just change the usual hard likelihood constraints $\{\theta \sim \pi : \mathcal{L}(\theta) > \mathcal{L}_*\}$ to

$$\{\theta \sim \pi : \rho(D(\theta), D_{\text{obs}}) < \varepsilon\}$$

(Brewer & Foreman-Mackey [1606.03757])

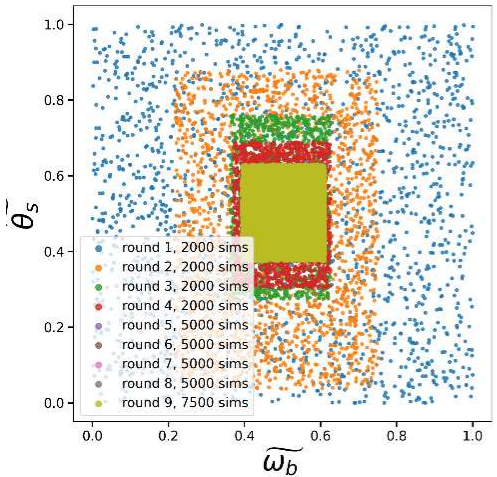
- Ongoing work with Andrew Fowlie & Sebastian Hoof
  - How to deal with nondeterminism
  - How to interpret ρ as a "likelihood"
  - How to interpret the evidence $\mathcal{Z}$

Nested sampling for truncated methods

- Will hear more on this tomorrow from Christoph
- Many Likelihood implicit approaches at the moment have some element of sampling direct from the prior
- Inefficient if number of parameters $> \mathcal{O}$(a few)
- Can get round this by truncating to region:

$$\Gamma\{\theta \in \mathrm{supp}p(\theta) \mid p(\theta|x_0) > \tilde{\varepsilon}\}$$

- At the moment regions defined by nested boxes
- This seems ripe for replacement by NS
  - Has anybody tried this?
  - If not, why not?
  - Why not why not? (let's talk)



Cole *et al.* [2111.08030]

Nested Sampling: a user's guide

1. Nested sampling is a likelihood scanner, rather than posterior explorer.
   - This means typically most of its time is spent on burn-in rather than posterior sampling
   - Changing the stopping criterion from 10^{-3} to 0.5 does little to speed up the run, but can make results very unreliable
2. The number of live points n_{live} is a resolution parameter.
   - Run time is linear in n_{live}, posterior and evidence accuracy goes as $\frac{1}{\sqrt{n_{\mathrm{live}}}}$.
   - Set low for exploratory runs $\sim \mathcal{O}(10)$ and increased to $\sim \mathcal{O}(1000)$ for production standard.
3. Most algorithms come with additional reliability parameter(s).
   - e.g. `MultiNest`: eff, `PolyChord`: n_{repeats}
   - These are parameters which have no gain if set too conservatively, but increase the reliability
   - Check that results do not degrade if you reduce them from defaults, otherwise increase.

Occam's Razor [2102.11511]

- Bayesian inference quantifies Occam's Razor:
  - *"Entities are not to be multiplied without necessity"* — William of Occam
  - *"Everything should be kept as simple as possible, but not simpler"* — "Albert Einstein"
- Properties of the evidence: rearrange Bayes' theorem for parameter estimation

$$\mathcal{P}(\theta) = \frac{\mathcal{L}(\theta)\pi(\theta)}{\mathcal{Z}} \qquad \Rightarrow \qquad \log \mathcal{Z} = \log \mathcal{L}(\theta) - \log \frac{\mathcal{P}(\theta)}{\pi(\theta)}$$

- Evidence is composed of a "goodness of fit" term and "Occam Penalty"

- RHS true for all θ. Take max likelihood value θ_*:

$$\log \mathcal{Z} = -\chi^2_{\min} - \text{Mackay penalty}$$

- Be more Bayesian and take posterior average to get the "Occam's razor equation"

$$\boxed{\log \mathcal{Z} = \langle \log \mathcal{L} \rangle_{\mathcal{P}} - \mathcal{D}_{\mathrm{KL}}}$$

- Natural regularisation which penalises models with too many parameters.

Kullback Liebler divergence

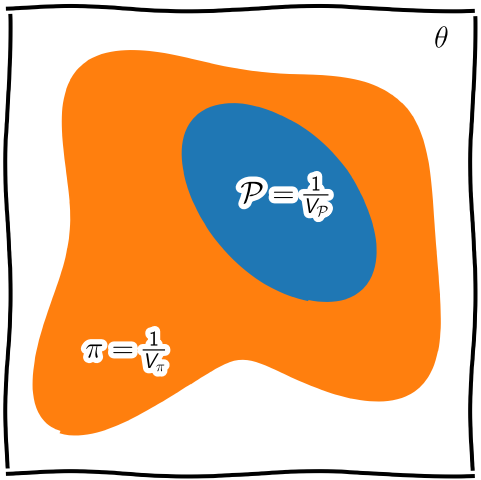
- The KL divergence between prior π and posterior $\mathcal{P}$ is is defined as:

$$\mathcal{D}_{\mathrm{KL}} = \left\langle \log \frac{\mathcal{P}}{\pi} \right\rangle_{\mathcal{P}} = \int \mathcal{P}(\theta) \log \frac{\mathcal{P}(\theta)}{\pi(\theta)} d\theta.$$

- Whilst not a distance, $\mathcal{D} = 0$ when $\mathcal{P} = \pi$.
- Occurs in the context of machine learning as an objective function for training functions.
- In Bayesian inference it can be understood as a log-ratio of "volumes":

$$\mathcal{D}_{\mathrm{KL}} \approx \log \frac{V_\pi}{V_\mathrm{P}}.$$

(this is exact for top-hat distributions).



Key tools for Nested Sampling

`anesthetic`	Nested sampling post processing [1905.04768]
`insertion`	cross-checks using order statistics [2006.03371] `github.com/williamjameshandley/anesthetic`
`nestcheck`	cross-checks using unthreaded runs [1804.06406] `github.com/ejhigson/nestcheck`
`MultiNest`	Ellipsoidal rejection sampling [0809.3437] `github.com/farhanferoz/MultiNest`
`PolyChord`	Python/C++/Fortran state of the art [1506.00171] `github.com/PolyChord/PolyChordLite`
`dynesty`	Python re-implementation of several codes [1904.02180] `github.com/joshspeagle/dynesty`
& `UltraNest`	`github.com/JohannesBuchner/UltraNest` [2101.09604]