Nested Sampling

using an ensemble sampler

Gianluca Becuzzi

October 2021

Introduction

$$Z \cdot P(\theta) = \pi(\theta) \cdot \mathcal{L}(\theta)$$

$$X(L) = \int_{\mathcal{L}(\theta)>L} \pi(\theta) d\theta$$

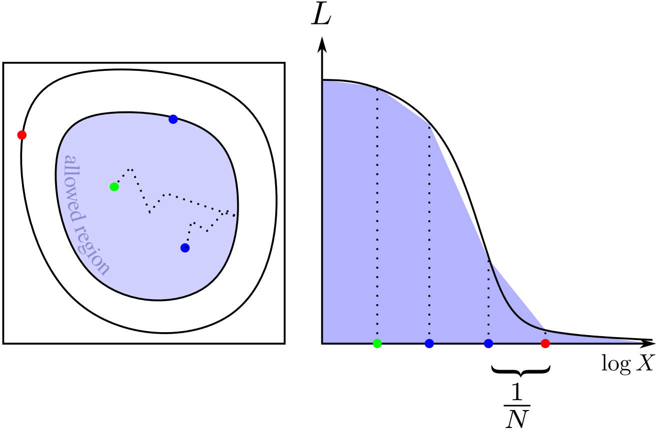
NS : what does it do?

- assesses X in a statistical way
- computes Z
- (optional) produces samples $\theta \sim P(\theta)$

Introduction

key idea

$$\theta \sim \pi(\theta)|_{\mathcal{L}(\theta)>L} \to X \sim \texttt{Uniform}(0, X(L))$$



Introduction - sampling issues

"Classical" way to generate a new live point:

- copy a current live point at random
- evolve it through MCMC/HMC
- take the last point of the chain

Domain shrinks $\rightarrow$ proposal acceptance decreases

Introduction - sampling possible solutions

Possible workarounds that reduce tuning/analytical work:

- vanishing adaptive MCMC
- proposal domain reduction strategies
- ensemble samplers (?)

I chose to test the affine invariant ensemble sampler (Goodman & Weare, 2010)

issues

Minor adjustments to the algorithm are needed

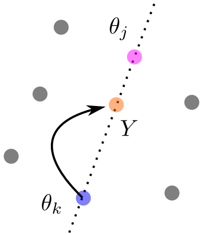
Possible interdependence of points

A.I.E.S. - main features

- uses a set of *walkers* $\{\theta_k(t)\}$
- samples $\Pi(\{\theta_k\}) = \pi(\theta_1)\dots\pi(\theta_L)$
- update for θ_k depends on $\theta_{[k]}$ by

$$Y = \theta_j + W \cdot (\theta_k - \theta_j)$$

$$P(W) = (W)^{-\frac{1}{2}}\chi[a^{-1}, a]$$



  - from the p.o.v. of the particle is more like a *jump process*
  - acceptance is *less* affected by shrinking

A.I.E.S. in NS

Given the N_{live} points, two possible (natural) routes:

1. copy all $\rightarrow$ evolve $\rightarrow$ choose one
2. copy all $\rightarrow$ evolve $\rightarrow$ choose all

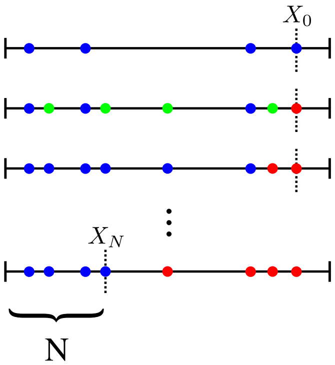
Strategy 1: slower but new point $\approx$ independent

Strategy 2: faster but dependence between new points
Furthermore, requires the management of a *variable number* of live points

AIES in NS - strategy 2

Basically a "statically-dynamic" nested sampling, for each generation:

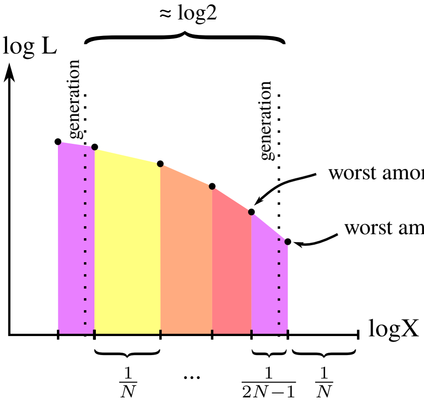
- starts with N points
- adds N points
- harvests one point at a time until $N_{live} = N$



So

$$N_{live}(t) = \{N, 2N-1, 2N-2, \ldots, N, 2N-1 \ldots\}$$

AIES in NS - X assessment



$$X(t) = X(t-1)\alpha(N_{live}(t))$$

$$P(\alpha|N) = N\alpha^{N-1}$$

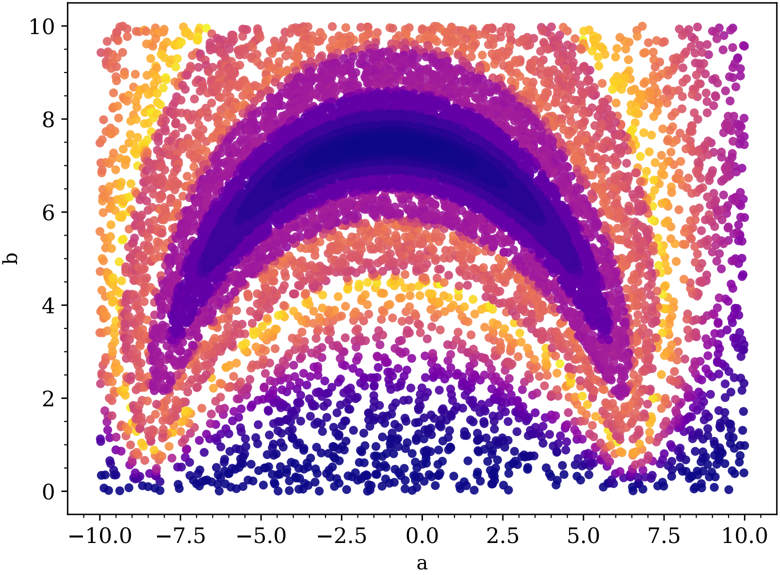
AIES in NS - X assessment

At the end of each harvest

$$\Delta \log X \approx -\sum_{k=N}^{2N-1} \frac{1}{k} \approx -\log\left(2 - \frac{1}{N}\right) \approx -\log 2$$

$$\delta Z = \sum \left(\frac{L_{i-1} + L_i}{2}\right) \delta X_i$$

Weighting samples



Stop criterion and error estimation

Stop when

$$\max(L_{live}) \cdot X_{current} < f \cdot Z_{current}$$

Last N points

harvest with $N_{live}(t) : N \to 1$

Error estimate

Simulate α outcomes $\to$ integrate $\to$ mean over $\log Z$ sample

Using inverse transform sampling:

$$\alpha(N(t)) = \sqrt[N(t)]{u}\,, \quad u \sim \texttt{Uniform(0,1)}$$

Parallelization

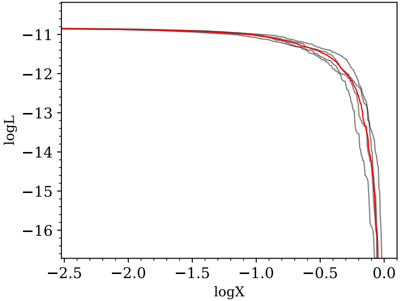
K runs of this type can be *merged* considering the number of live points at the time a certain likelihood threshold L^* is overcome:

$$N^{(i)}(t), L^{(i)}(t) \to N^{(i)}(L^*)$$

then considering

$$N_{merged}(L^*) = \sum N^{(i)}(L^*)$$

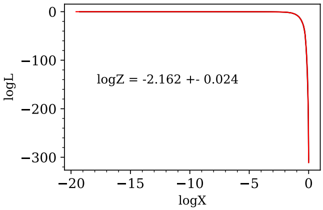
$$\log X(t) = -\sum_0^t \frac{1}{N_{merged}(t')}$$

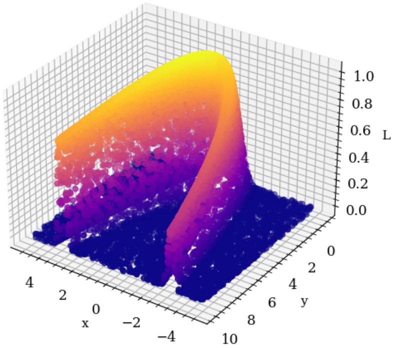


If the runs' uncertainties are similar $\Delta \log Z_{merged} \approx \frac{\Delta \log Z}{\sqrt{K}}$

Some test models - Rosenbrock

$$\log \mathcal{L}(x, y) = -\frac{1}{2}(y - x^2)^2 - \frac{(1-x)^2}{20}$$

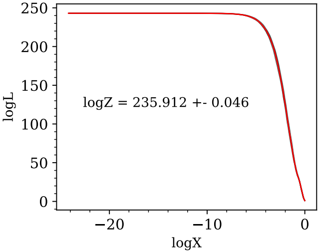


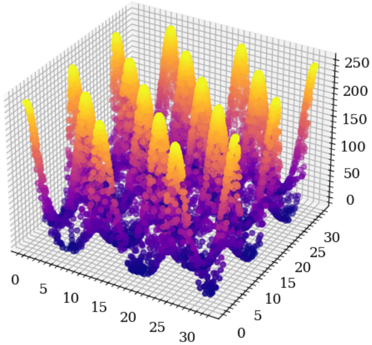


$$\log Z_{w\alpha} = -2.17 \quad \log Z_{NS} = -2.16 \pm 0.02 \;\; (N = 500, T = 25s)$$

Some test models - eggbox

$$\log \mathcal{L}(x,y) = \left[2 + \cos(\frac{x}{2})\cos(\frac{y}{2})\right]^5$$

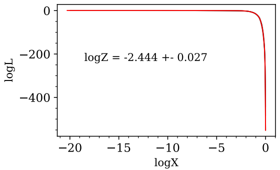


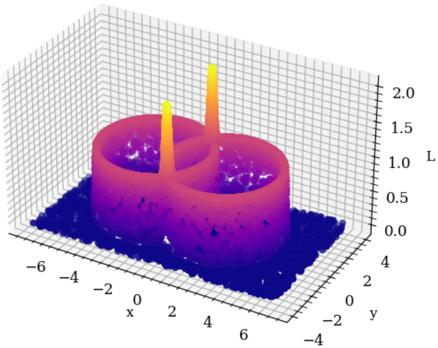


$$\log Z_{w\alpha} = 235.88 \quad \log Z_{NS} = 235.91 \pm 0.05 \;\; (N = 500, T = 37s)$$

Some test models - intersecting gaussian shells

$$\mathcal{L}(x,y) = e^{-\frac{(x-2)^2+y^2-3^2}{2(0.1)^2}} + e^{-\frac{(x+2)^2+y^2-3^2}{2(0.1)^2}}$$

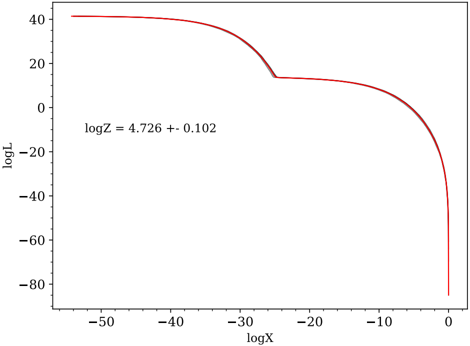




$$\log Z_{w\alpha} = -2.47 \quad \log Z_{NS} = -2.44 \pm 0.03 \;\; (N = 500, T = 62s)$$

Some test models - Skilling's "Statistics problem"

$$\mathcal{L}(x_1, \ldots, x_{10}) = \prod_i \frac{\exp\left(-\frac{1}{2}\left(\frac{x_i}{v}\right)^2\right)}{v\sqrt{2\pi}} + 100 \prod_i \frac{\exp\left(-\frac{1}{2}\left(\frac{x_i - 0.031}{u}\right)^2\right)}{u\sqrt{2\pi}}$$

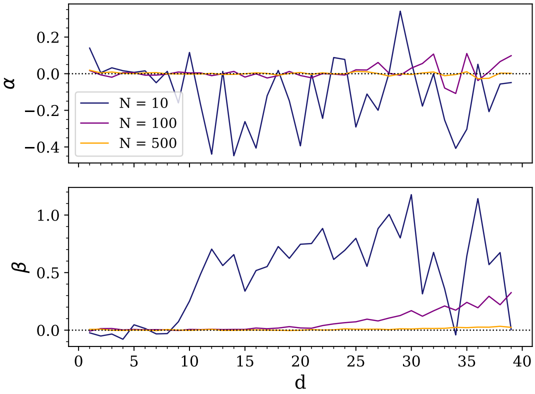


$$\log Z_{an} = \log(101) \approx 4.61 \quad \log Z_{NS} = 4.7 \pm 0.1 \;\; (N = 500, T = 85s)$$

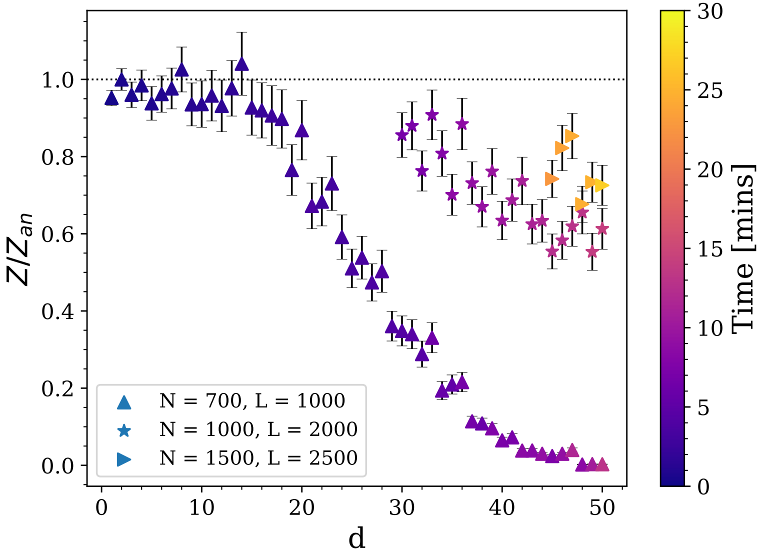
What about parameter estimation?

On a d-multivariate normal distribution

$$\alpha(d) = \sum_{i=0}^{d} \frac{\mu_i}{d} \qquad \beta(d) = \sum_{i=0}^{d} \frac{\sigma_i}{d} - 1$$



Space dimension dependence - MVN



Diffusive Nested Sampling

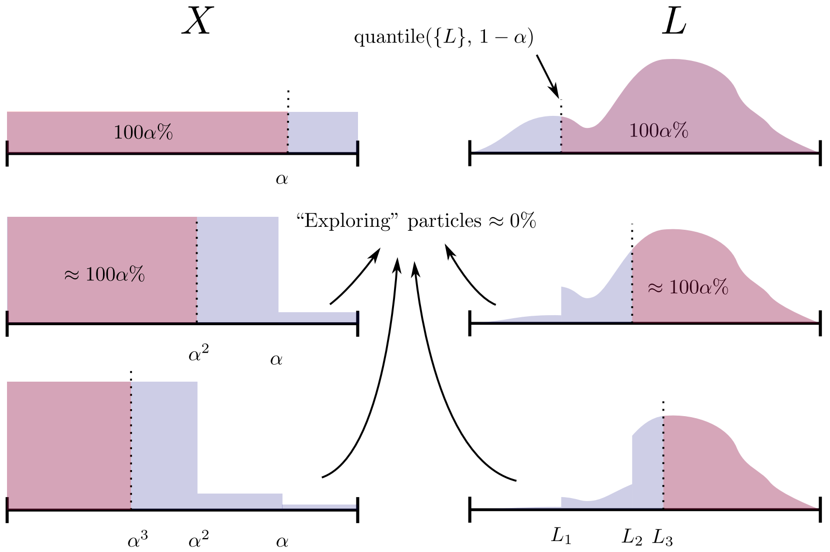
- divides the likelihood range in levels
- samples a weighted mixture of the LCP(s)

To generate new levels the algorithm plans to use the quantiles of the MC evolved $\{L\}$ over the current weighted mixture , using

$$L_{N+1} = G_L^{-1}(1 - G_X(X_{N+1})) \approx \mathcal{Q}(\{L\}, 1 - G_X(X_{N+1}))$$

At each generation, deleting the chain with $L < L_N$ ensures that $X \approx \texttt{Uniform}(0, X_N)$, so $G_X(X_{N+1}) \approx X_{N+1}/X_N = \alpha$

Diffusive Nested Sampling



DNS - mixture AIES

$$P(\theta, j) = P(\theta|j)P(j)$$

$$P(\theta|j) = \pi(\theta)\Theta(L(\theta) - L_j) \qquad P(j) = \frac{w_j}{X_j}$$

single walker

- MC for $j \rightarrow$ select the level
- MC for $\theta \rightarrow$ sample $P(\theta|j)$

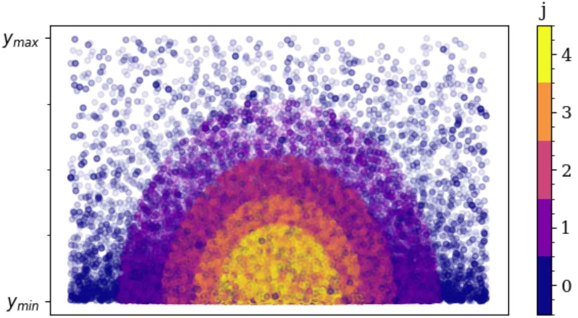
multiparticle

- MC for $\mathcal{J} = \{j_1, \ldots, j_L\}$
- AIES for $\Pi(\{\theta_k\}|\mathcal{J}) = P(\theta_1|j_1) \ldots P(\theta_L|j_L)$

DNS - mixture AIES

Partial sampling principle doesn't require each walker to be i.i.d. $\rightarrow$ proposals are A/R with the same probability

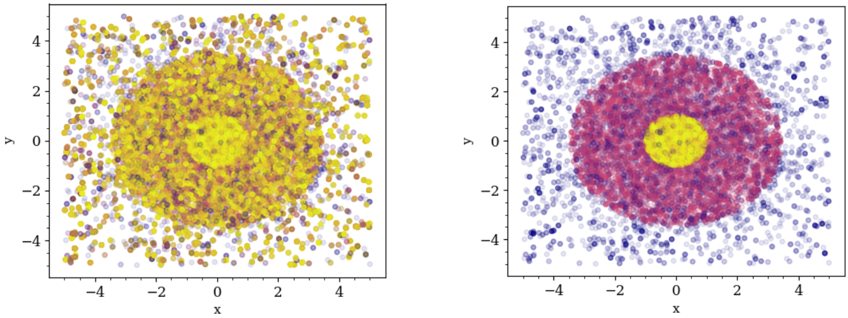
$$\alpha = 1 \wedge W^{n-1} \frac{P(Y|j_k)}{P(\theta_k|j_k)}$$



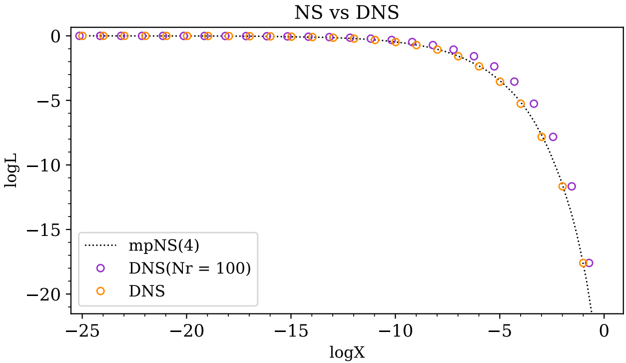
Cleaning the samples

Proposal rejected → walker doesn't move

sample for lvl. j is outside lvl. j → bad revising

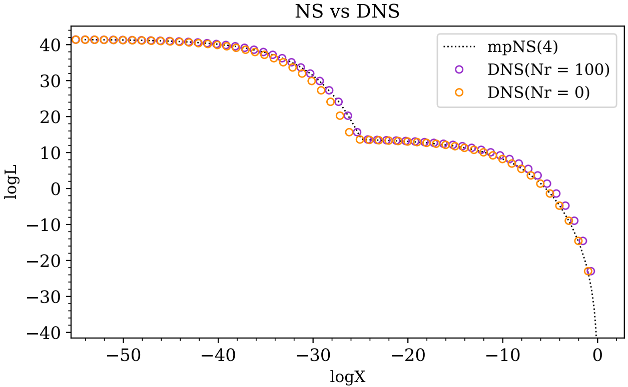


$$X_{j+1}/X_j = (n(j+1|j) + C\alpha)/(n(j) + C)$$



-	Z/Z_{an}	$\Delta Z/Z_{an}$	$T[mins]$
DNS($Nr = 100$)	2.26	–	2.45
DNS($Nr = 0$)	1.14	–	1.07
ENS	1.02	0.04	0.44

Comparison - “Statistics problem”



-	Z/Z_{an}	$\Delta Z/Z_{an}$	$T[mins]$
DNS($Nr = 100$)	1.31	–	2.94
DNS($Nr = 0$)	0.55	–	1.21
ENS	1.12	0.1	1.52

Conclusions

- A.I.E.S. gives N.S. improvements (multimododality, almost no tuning necessary) while retaining other features (Z error estimate, parallelization)
- A.I.E.S. does not seem to improve D.N.S.

Quantile reconstruction

x independent, $y = y(x)$ unknown monotonic function

$$G_y(y) = \int_{y_{min}}^{y} p_y(y')dy' = \int_{x(y_{min})}^{x(y)} p_x(x(y'))|dx(y')| = \begin{cases} G_x(x(y)) \\ 1 - G_x(x(y)) \end{cases}$$

so

$$y = G_y^{-1}(G_x(x(y)))$$

Since the sample quantile function $\mathcal{Q}(\{samples\}, p)$ approximates $G^{-1}(p)$

If $G_x(x)$ is known and $\{y\}$ is sampled

$$y(x) \approx \mathcal{Q}(\{y\}, G_y(x))$$

Affine invariant ensemble sampler

Takes L *walkers* in the sample space $\Omega \subset \mathbb{R}^n$, for each generates a MC *dependent on the others* that samples:

$$\Pi(\theta_1, \ldots, \theta_L) = \pi(\theta_1) \ldots \pi(\theta_L)$$

Sampling is done by generating a *stretch coefficient*

$$W \sim 1/\sqrt{W}\ , \ \ W \in [a^{-1}, a]$$

Then, for each θ_k, choosing a *pivot* θ_j for the stretch, with $\theta_j \in \theta_{[\,k]} = \{\theta_i\}_{i \neq k}$ and proposing

$$\theta_k(t+1) \leftarrow Y = \theta_j + W(\theta_k - \theta_j)$$

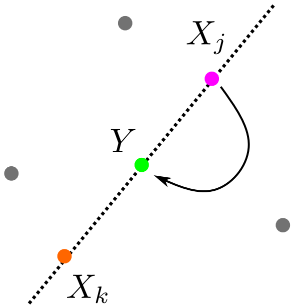
as the update value for θ_k.

Robust in $a \rightarrow$ "almost-non-parametric" sampler

Affine invariant ensemble sampler

Partial sampling:

$P(\theta_k|\theta_{[\,k]}\,)$ invariant under
$\theta_k \to Y$
$\downarrow$
$\Pi(\{\theta\})$ invariant under
$\{\theta\}(t) \to \{\theta\}(t+1)$



This is done by accepting Y with probability

$$\alpha(Y, \theta_k) = 1 \wedge Z^{n-1}\frac{\pi(Y)}{\pi(\theta_k)}$$

AIES in NS - X assessment

When harvesting a live point, its X has to be estimated

Since the old and the generated points are uniform in $[0, X_0]$, at each kill X can be estimated as

$$X(t) = X(t-1)\alpha(N(t))$$

where α is distributed the max of N points uniform in $[0, 1]$:

$$P(\alpha|N) = N\alpha^{N-1}$$

Like in classic sampling strategies

$$\delta \log X \simeq \langle \log \alpha \rangle_N \simeq -\frac{1}{N}$$