Nested sampling

Common implementations:
- MultiNest
- PolyChord

- Original method due to Skilling (2004)
- Actually a method for computing the Bayesian evidence, Z
- Useful by-product: We get $\bar{\theta}$ samples distributed according to $p(\bar{\theta}|D)$

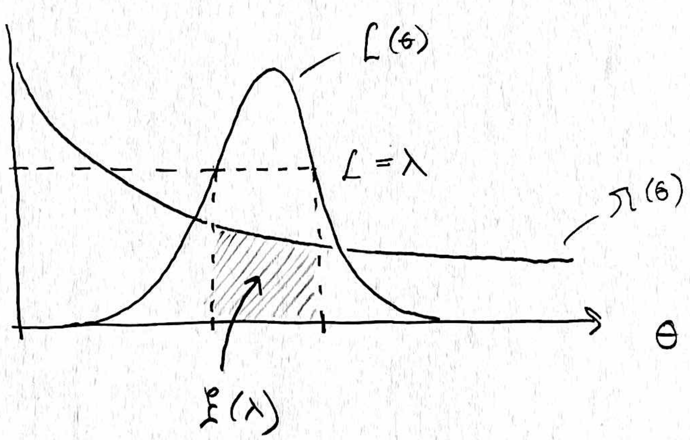
- Want to compute $Z = \int \mathcal{L}(\bar{\theta})\pi(\bar{\theta})\,d\bar{\theta}$ (*)
- High-dim integrals are hard! One-dim. are easy!
- Can we turn (*) into a one-dim. integral?
- Introduce variable: "prior mass"

Terminology:
prob. density $= \dfrac{d(\text{prob. mass})}{d(\text{volume})}$

$$\xi(\lambda) = \int_{\mathcal{L}(\theta)>\lambda} \pi(\theta)\,d\theta$$

$$d\xi = \pi(\theta)\,d\theta$$

Transf. of random variable:
$p_x(x)\,dx = p_y(y)\,dy$
Here: $\pi_\xi(\xi)\,d\xi = \pi_\theta(\theta)\,d\theta$
with $\pi_\xi(\xi) = 1$
and $\xi \in [0,1]$



$\xi(\lambda)$: The amount of prior probability contained within the regions of parameter space where the likelihood $\mathcal{L}(\theta)$ is greater than some value λ

$d\xi$: The small additional prior mass included by lowering the likelihood threshold by $d\mathcal{L}$

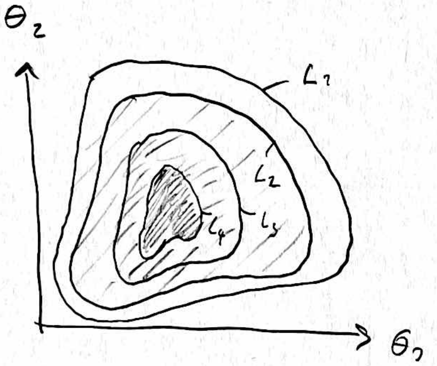
Examples: $\xi(0) = 1$, $\xi(\lambda = \mathcal{L}_{max}) = 0$

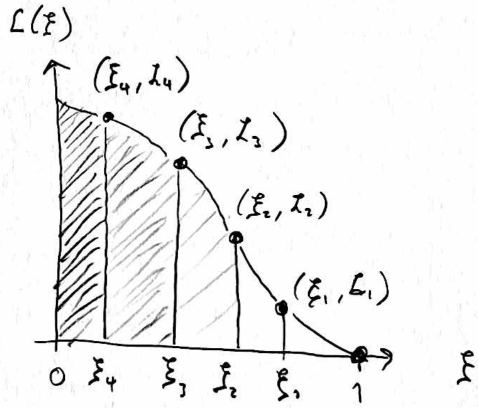
- Note that $\xi(\lambda)$ is a one-dim, decreasing function of λ
- Inverse function, denoted as $\mathcal{L}(\xi)$ is simply

$$\mathcal{L}(\xi(\lambda)) \equiv \lambda$$ = the value for the likelihood contour that contains a given prior mass ξ

- Can now express Z as one-dim. integral over ξ:

$$Z = \int_0^1 \mathcal{L}(\xi)\, d\xi$$





- If we can get a set of ordered pairs of values $(\xi_i, \mathcal{L}_i)$ we can evaluate Z integral using standard methods, (e.g trapezoidal rule)
- Nested sampling is algorithm to get these samples

Algorithm [Show slides]

1) Draw N "live points" $\bar{\theta}$ according to prior $\pi(\bar{\theta})$

2) Evaluate $\mathcal{L}(\bar{\theta})$ at each live point

3) Discard (but record) point with lowest likelihood

4) Draw a new point from prior, but with additional req. that $\mathcal{L}(\bar{\theta}_{new}) > \mathcal{L}(\bar{\theta}_{disc.})$

5) Repeat from step 3

Main challenge for algo. efficiency!

- The discarded points form ordered set of likelihood samples

$$0 < \mathcal{L}_1 < \mathcal{L}_2 < \ldots$$

- For each likelihood sample, can estimate corresponding prior mass ξ_i to obtain

(will show this later)

$$1 > \xi_1 > \xi_2 > \ldots$$

- Result:

Evidence estimate:

$$Z \approx \sum_{i=1}^{M} \mathcal{L}_i w_i = \sum_{i=1}^{M} \mathcal{L}_i \underbrace{\tfrac{1}{2}[\xi_{i-1} - \xi_{i+1}]}$$

w_i: The slice of prior mass associated with the likelihood value $\mathcal{L}_i$. (Here chosen according to trapezoidal rule)

Posterior samples:

Assign each discarded parametersample $\bar{\theta}_i$ its share of the posterior prob.

$$p_i = \frac{\mathcal{L}_i w_i}{Z}$$

Main challenge: How to efficiently draw replacement samples from the "likelihood-constrained prior"? MultiNest + friends solve this!

- How can we estimate the prior mass ξ_i corresponding to a likelihood value L_i ?

$$0 < L_1 < L_2 < \dots$$

$$1 > \xi_1 > \xi_2 > \dots$$

- From $d\xi = \pi(\bar{\theta})\, d\bar{\theta}$, we know that sampling $\bar{\theta}$ according to $\pi(\bar{\theta})$ corresponds to sampling ξ from uniform distribution $U(0,1)$

[Recall relation :

$$\bar{\theta} \longrightarrow L(\bar{\theta}) \longrightarrow \lambda \longrightarrow \text{Integration contur for } \int_{L(\bar{\theta})>\lambda} \pi(\bar{\theta})\, d\bar{\theta} \longrightarrow \xi$$

]

- Sampling constraint $L(\bar{\theta}_{new}) > L(\bar{\theta}_{disc.})$ ensures that the prior mass associated with $\bar{\theta}_{new}$ is <u>smaller</u> than that for $\bar{\theta}_{disc.}$

$$\xi_{new} < \xi_{disc.}$$

- We use N live points. At start of iteration i we have N $\bar{\theta}$-samples that should correspond to N ξ-samples from uniform distr. on $(0, \xi_{i-1})$

- The prior mass ξ_i of next point to be discarded is an unknow/random variable

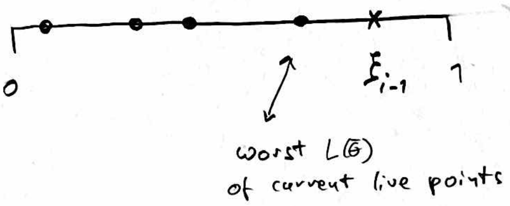
$$\xi_i = t_i\, \xi_{i-1}$$

where the shrinkage factor $t_i = \dfrac{\xi_i}{\xi_{i-1}}$ has a pdf

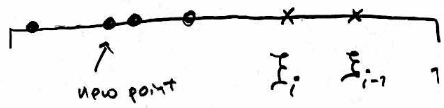
$$p(t) = N t^{N-1}$$

[Pdf for the largest value t of N samples drawn from $U(0,1)$]

Iteration i $\quad (N=4)$

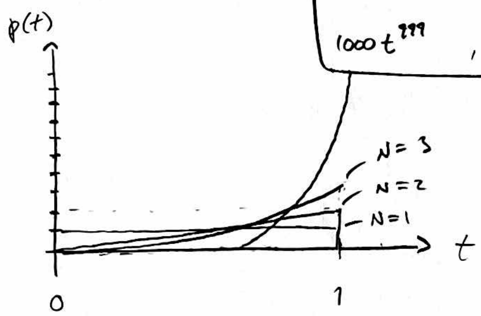


$\Downarrow$ Discard and sample a new point under constraint



[We don't try to compute the exact ξ_i for the iteration i, we just estimate it]

$$p(t) = N t^{N-1} = \begin{cases} 1 & , N=1 \\ 2t & , N=2 \\ 3t^2 & , N=3 \\ 1000\, t^{999} & , N=1000 \end{cases}$$



- Since we start from $\xi_0 = 1$, we can express ξ_i as the random variable

$$\xi_i = t_i t_{i-1} \dots t_1 \qquad \left(\text{since } \xi_i = t_i \xi_{i-1} = t_i t_{i-1} \xi_{i-2} = \dots\right)$$

or $\quad \ln \xi_i = \ln t_i + \ln t_{i-1} + \dots$

- All the t_i have pdf $p(t) = N t^{N-1}$ which give

$$E[\ln t] = -\frac{1}{N} \quad , \quad Var[\ln t] = \frac{1}{N^2}$$

- This means that the sum $\ln \xi_i = \ln t_i + \ln t_{i-1} + \dots =$ has expectation and variance

$$E[\ln \xi_i] = E[\ln t_i] + E[\ln t_{i-1}] + \dots$$

$$= -\frac{1}{N} - \frac{1}{N} - \dots$$

$$= -\frac{i}{N}$$

$$\mathrm{Var}[\ln \xi_i] = \mathrm{Var}[\ln t_i] + \mathrm{Var}[\ln t_{i-1}] + \dots \quad \left(\text{Since the } t_i \text{ are uncorrelated}\right)$$

$$= \frac{i}{N^2}$$

- In short: $\ln \xi_i \approx -\frac{i}{N} \pm \frac{\sqrt{i}}{N}$

- So we approximate the prior mass ξ_i associated with the likelihood value L_i of the discarded point $\bar{\theta}$ at iteration i as

$$\boxed{\xi_i \approx e^{-\frac{i}{N}}}$$

- Then we have what we need to compute a new approx. for Z after each new iteration i

- The sampling stops when the largest possible contribution ΔZ from the current live points is much smaller than the current estimate for Z

  (But this will fail if the sampling has missed some region of high likelihood)

- Uncertainty on final evidence estimate is dominated by uncertainty in ξ_i estimates

  (assuming the sampling has found all relevant parameter regions.)

- Efficiency challenge:
  - Naively sampling $\bar{\theta}$ points from entire $\Omega(\bar{\theta})$ at every iteration will lead to ever decreasing efficiency, due to constraint $L(\bar{\theta}_{new}) > L(\bar{\theta}_{disc.})$
  - One appr. used to alleviate problem:
    - Draw samples from ellipsoids containing current live points
    - Use clustering algos. to assign sep. ellipsoids to sep. clusters of live points

- Much used packages: MultiNest, PolyChord
  (pymultinest)