Randomized adaptive cooling schedules for Monte Carlo Integration

Mark Huber[1] and Sarah Schott[2]

[1]Department of Mathematics and Computer Science, Claremont McKenna College

[2]Department of Mathematics, Duke University

27 Oct, 2010

Integration

Numerical integration

The central problem: Approximate

$$I = \int_{\vec{x} \in \Omega} f(\vec{x})\, d\mathbb{R}^n, \quad f(\vec{x}) \geq 0$$

Application: statistical physics

Normalizing constant of Gibbs distributions

$$Z = \int_{\vec{x} \in \Omega} e^{-\beta H(\vec{x})} \, d\mathbb{R}^n$$

$$Z = \sum_{\vec{x} \in \Omega} e^{-\beta H(\vec{x})}, \quad w(\vec{x}) \geq 0.$$

Application: computer science

#P complete problems

$$Z = \sum_{\vec{x} \in \Omega} w(\vec{x}), \quad w(\vec{x}) \geq 0.$$

Counting

- matchings
- linear extensions
- satisfying assignments

Application: frequentist exact p-values

For statistic S, given `data`

$$p = \int_{\vec{x} \in \Omega} \mathbf{1}(S(\vec{x}) \geq S(\texttt{data}))\, d\vec{x},$$

Application: Bayesian integrated likelihood (evidence)

Prior density $f_{\textbf{prior}}$, likelihood $f_{\textbf{like}}$, parameter space Ω:

$$Z = \int_{b \in \Omega} f_{\text{prior}}(b) f_{\text{like}}(b) \, db.$$

Posterior mean of $\theta(i)$

$$Z = \int_{b \in \Omega} b(i) f_{\text{prior}}(b) f_{\text{like}}(b) \, db.$$

Many ways to go from samples to integrals

Some methods:

- (Sequential) Importance sampling
- Bridge sampling
- Path sampling
- Nested sampling
- Harmonic mean estimator

Problem: every one of these methods has unknown variance!

- SIS and HME variance can easily be infinite
- Estimated variance itself has unknown variance

Abstract problem

What is the measure of a set A:

$$m(A) = \int_A f(\vec{x}) d\vec{x}$$

Want estimate $\hat{a}$ such that

- $\mathbb{E}[\hat{a}] = m(A)$
- $\text{SD}[\hat{a}] \leq \epsilon m(A)$

Outline

The problem

- Finding integral of a density
- Applications: physics, CS, statistics, many more

Approximation through Monte Carlo

- Want bounded variance approach
- Basic method acceptance/rejection

Randomized adaptive cooling schedules (RACS)

- Know variance of estimate
- TPA
- M-Chebyshev schedules

Approximation through Monte Carlo

Basic Monte Carlo

First write $\mu(A) = \mathbb{E}[X]$

- Draw $X_1, \ldots, X_n \overset{\text{iid}}{\sim} X$
- $\mu(B) \approx \bar{X} = \frac{1}{n} \sum_i X_i$
- $\mathbb{E}[\bar{X}] = \mu(B)$, $\mathbb{V}(\bar{X}) = (1/n)\mathbb{V}(X_i)$
- Variance could be huge, even infinite
- Estimation of variance more difficult than original problem

Good approximation

Goal for approximation

- Good up to small multiplicative factor
- Informally: $\text{SD}[\hat{a}] \leq \epsilon \mathbb{E}[a]$
- Formally:

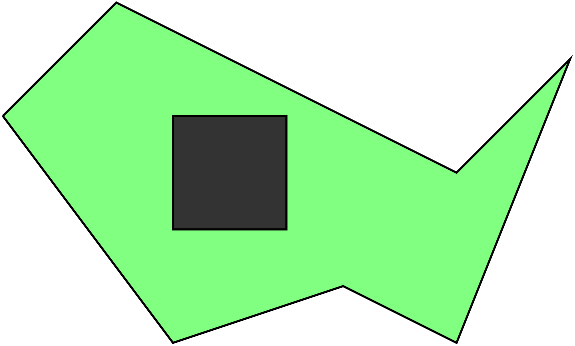
$$\mathbb{P}\left(\frac{1}{1+\epsilon} \leq \frac{\hat{a}}{a} \leq 1+\epsilon\right) \geq 1-\delta$$

Very few methods

- Acceptance/Rejection (AR)

Acceptance/Rejection

What is the measure of a set?

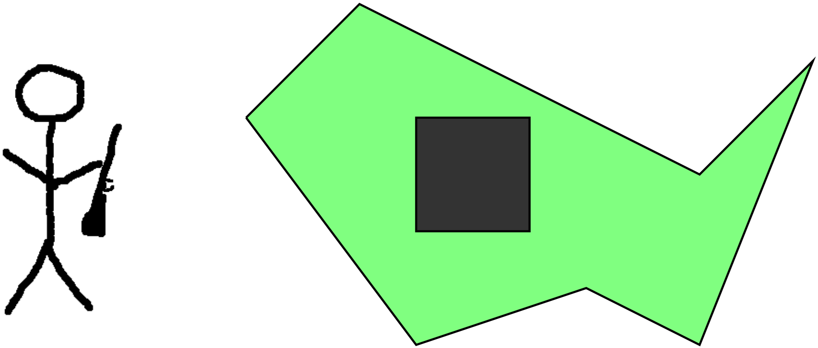


Green area = B

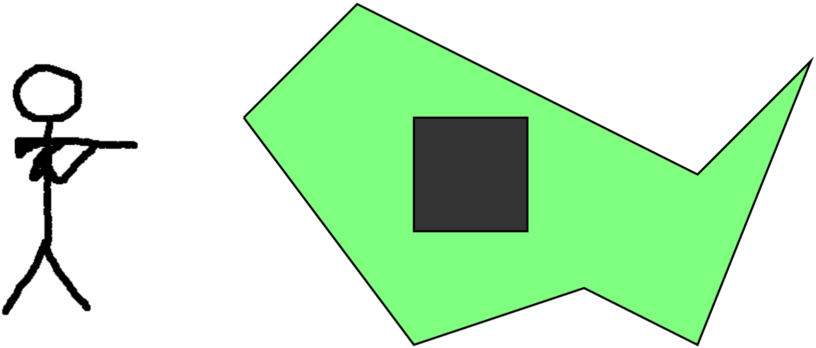
Black area = A

$$\mu(B) = \mu(A)\frac{\mu(B)}{\mu(A)}$$

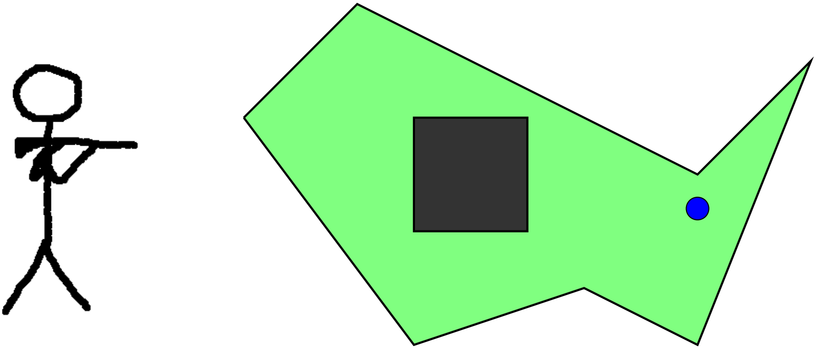
Acceptance/Rejection a.k.a "Shoot at it randomly"



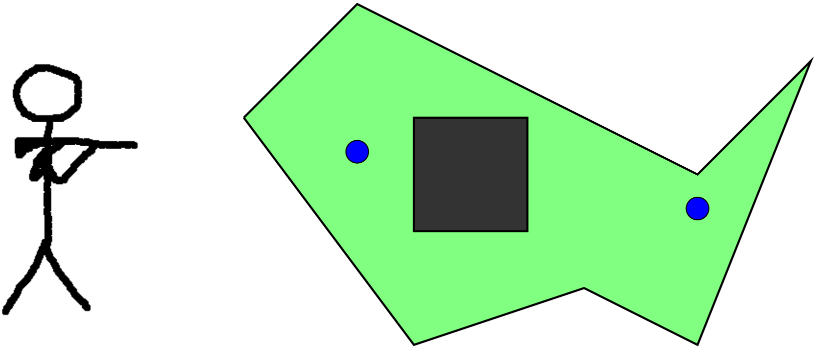
Acceptance/Rejection a.k.a "Shoot at it randomly"



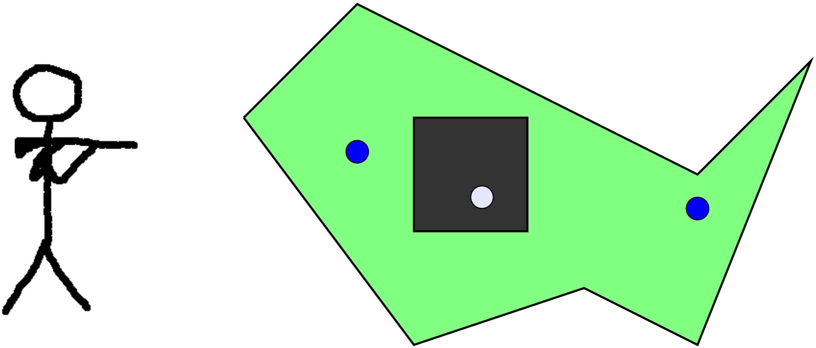
Acceptance/Rejection a.k.a “Shoot at it randomly”



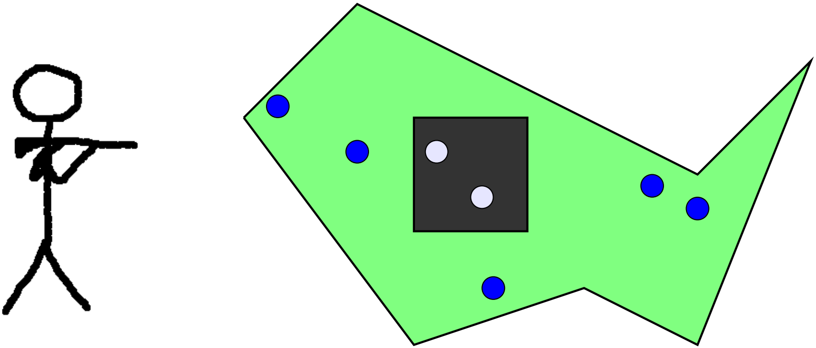
Acceptance/Rejection a.k.a “Shoot at it randomly”



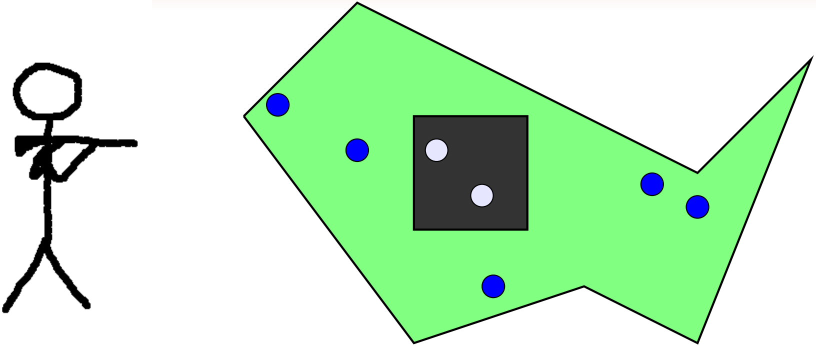
Acceptance/Rejection a.k.a "Shoot at it randomly"



Acceptance/Rejection a.k.a "Shoot at it randomly"



Acceptance/Rejection a.k.a "Shoot at it randomly"



Best estimate:

$$\hat{\mu}(B) = \frac{7}{2}\mu(A), \quad A = \text{black rectangle inside region}$$

Analysis of Accept/Reject

Algorithm

- Fire n times at box
- Say you hit H times
- Estimate: $\hat{a} = \mu(A)n/H$

How well does it work?

True answer: 3.0933...

- After 10 iterations 5.04
- After 10^3 iterations 2.8656
- After 10^5 iterations 3.0870
- After 10^7 iterations 3.0935

About a factor of 100 per extra digit

How well does it work?

True answer: 3.0933...

- After 10 iterations 5.04
- After 10^3 iterations 2.8656
- After 10^5 iterations 3.0870
- After 10^7 iterations 3.0935

About a factor of 100 per extra digit

How well does it work?

True answer: 3.0933...

- After 10 iterations 5.04
- After 10^3 iterations 2.8656
- After 10^5 iterations 3.0870
- After 10^7 iterations 3.0935

About a factor of 100 per extra digit

How well does it work?

True answer: 3.0933...

- After 10 iterations 5.04
- After 10^3 iterations 2.8656
- After 10^5 iterations 3.0870
- After 10^7 iterations 3.0935

About a factor of 100 per extra digit

How well does it work?

True answer: 3.0933...

- After 10 iterations 5.04
- After 10^3 iterations 2.8656
- After 10^5 iterations 3.0870
- After 10^7 iterations 3.0935

About a factor of 100 per extra digit

Bounding error

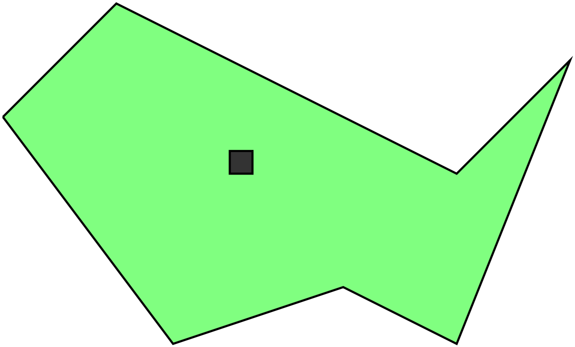
Get relative error below ϵ with probability at least $1 - \delta$:

- Need to analyze tails of binomial distribution to show:

$$n \approx \frac{2(1+\epsilon)}{p} \cdot \frac{1}{\epsilon^2} \cdot \ln \frac{1}{\delta}, \quad p = \frac{\mu(A)}{\mu(B)}$$

- The $(1/\epsilon^2)\ln(1/\delta)$ often called "Monte Carlo" error
- Best you can do in general
- So concentrate on improving $1/p$ part

When p small, runtime large



Usually p exponentially small in dimension of problem

Running times

Acceptance/Rejection:

$$2 \cdot \frac{1}{p}.$$

Product Estimator [1]:

$$192 \cdot \left[\log \frac{1}{p}\right]^2.$$

Using RACS

- TPA: $2[\log 1/p]^2$
- M-Chebyshev: $100[\log 1/p]$

Nested Sampling [Skilling 2007]

Nested sampling another approach to these integrals

- Mix of product estimator-like algorithm and classical 1-D numerical integration
- Not quite approximation algorithm
- Roughly speaking also $[\log(1/p)]^2$
- Does introduce a nice idea
- Combination nice idea + product estimator = TPA

Randomized adaptive cooling schedule: TPA

TPA

Idea

- Product estimator...
- ...plus idea from Nested sampling

Result

- Product estimator with random cooling schedule
- Output can be analyzed exactly (like A/R)

The Tootsie Pop Algorithm

What is a Tootsie Pop?

- Hard candy lollipops with a tootsie roll (chewy chocolate) at the center

In 1970, Mr. Owl was asked the question:

- How many licks does it take to get to the center of a Tootsie Pop?

List of ingredients of TPA

(a) A measure space $(\Omega, \mathcal{F}, \mu)$

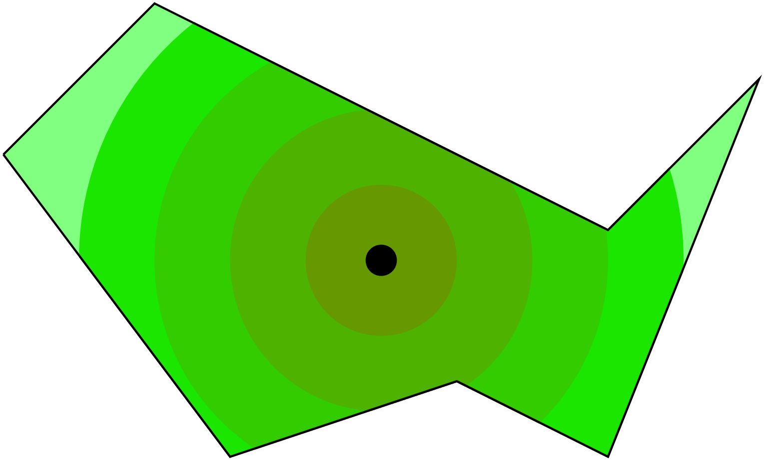
(b) Two measurable sets: the *center* B' and the *shell* B with $B' \subset B$

(c) A family of sets $\{A(\beta)\}$ where

  1. $\beta' < \beta$ implies $A(\beta') \subseteq A(\beta)$,
  2. $\mu(A(\beta))$ is continuous in β

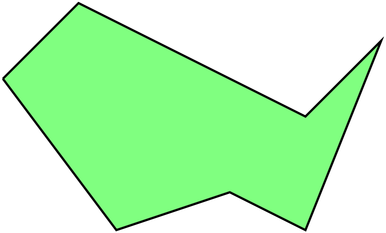
(d) Two special values β_B and $\beta_{B'}$ with $A(\beta_B) = B$ and $A(\beta_{B'}) = B'$.

Example of nested sets



$A(\beta)$ = all points within distance β of center

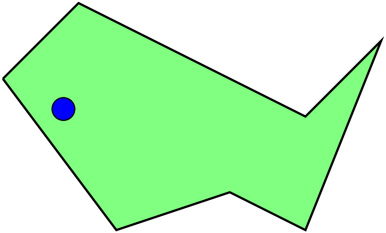
Idea behind TPA



Step	β
0	∞
1	
2	
3	

1. $\beta \leftarrow \beta_B$
2. Repeat
3. Draw $X \leftarrow \mu(A(\beta))$
4. $\beta \leftarrow \inf\{\beta' : X \in A(\beta')\}$
5. Until $\beta \leq \beta_{B'}$

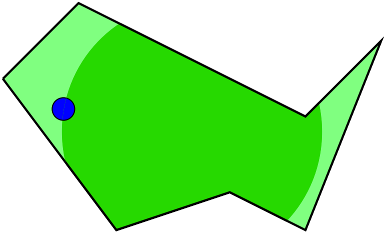
Idea behind TPA



Step	β
0	∞
1	
2	
3	

1. $\beta \leftarrow \beta_B$
2. Repeat
3. Draw $X \leftarrow \mu(A(\beta))$
4. $\beta \leftarrow \inf\{\beta' : X \in A(\beta')\}$
5. Until $\beta \leq \beta_{B'}$

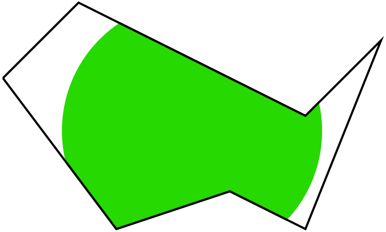
Idea behind TPA



Step	β
0	∞
1	1.72
2	
3	

1. $\beta \leftarrow \beta_B$
2. Repeat
3. Draw $X \leftarrow \mu(A(\beta))$
4. $\beta \leftarrow \inf\{\beta' : X \in A(\beta')\}$
5. Until $\beta \leq \beta_{B'}$

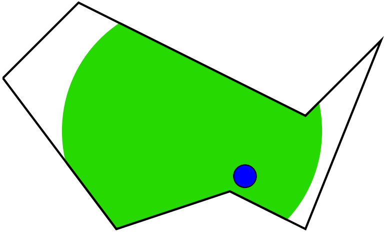
Idea behind TPA



Step	β
0	∞
1	1.72
2	
3	

1. $\beta \leftarrow \beta_B$
2. Repeat
3. Draw $X \leftarrow \mu(A(\beta))$
4. $\beta \leftarrow \inf\{\beta' : X \in A(\beta')\}$
5. Until $\beta \leq \beta_{B'}$

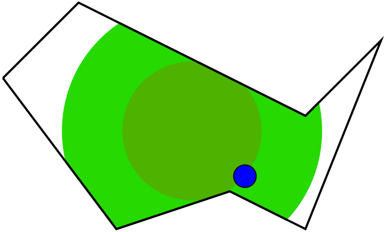
Idea behind TPA



Step	β
0	∞
1	1.72
2	
3	

1. $\beta \leftarrow \beta_B$
2. Repeat
3. Draw $X \leftarrow \mu(A(\beta))$
4. $\beta \leftarrow \inf\{\beta' : X \in A(\beta')\}$
5. Until $\beta \leq \beta_{B'}$

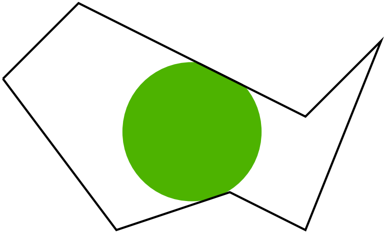
Idea behind TPA



Step	β
0	∞
1	1.72
2	0.92
3	

1. $\beta \leftarrow \beta_B$
2. Repeat
3. Draw $X \leftarrow \mu(A(\beta))$
4. $\beta \leftarrow \inf\{\beta' : X \in A(\beta')\}$
5. Until $\beta \leq \beta_{B'}$

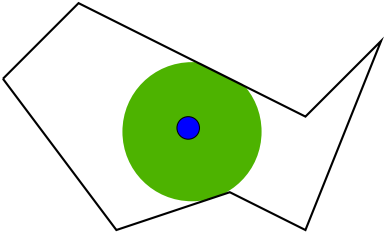
Idea behind TPA



Step	β
0	∞
1	1.72
2	0.92
3	

1. $\beta \leftarrow \beta_B$
2. Repeat
3.   Draw $X \leftarrow \mu(A(\beta))$
4. $\beta \leftarrow \inf\{\beta' : X \in A(\beta')\}$
5. Until $\beta \leq \beta_{B'}$

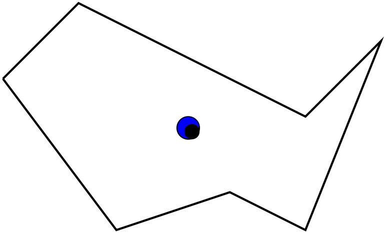
Idea behind TPA



Step	β
0	∞
1	1.72
2	0.92
3	

1. $\beta \leftarrow \beta_B$
2. Repeat
3. Draw $X \leftarrow \mu(A(\beta))$
4. $\beta \leftarrow \inf\{\beta' : X \in A(\beta')\}$
5. Until $\beta \leq \beta_{B'}$

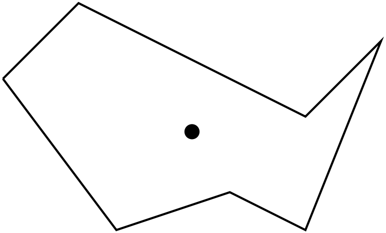
Idea behind TPA



Step	β
0	∞
1	1.72
2	0.92
3	0.09

1. $\beta \leftarrow \beta_B$
2. Repeat
3. Draw $X \leftarrow \mu(A(\beta))$
4. $\beta \leftarrow \inf\{\beta' : X \in A(\beta')\}$
5. Until $\beta \leq \beta_{B'}$

Idea behind TPA



Step	β
0	∞
1	1.72
2	0.92
3	0.09

1. $\beta \leftarrow \beta_B$
2. Repeat
3. Draw $X \leftarrow \mu(A(\beta))$
4. $\beta \leftarrow \inf\{\beta' : X \in A(\beta')\}$
5. Until $\beta \leq \beta_{B'}$

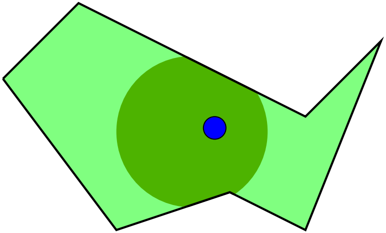
How much is shaved off at each step?

Notation: $Z(\beta) := \mu(A(\beta))$

Lemma

Say $X \sim \mu(A(\beta))$ *and* $\beta' = \min\{\beta' : X \in A(\beta')\}$. *Then*

$$\frac{Z(\beta')}{Z(\beta)} \sim \text{Unif}([0,1])$$



Proof by picture:

Let b satisfy $Z(b)/Z(\beta) = 1/3$

Then

$$\mathbb{P}\left(\frac{Z(\beta')}{Z(\beta)} \leq 1/3\right) = \mathbb{P}(X \in A(b))$$

Each step removes on average 1/2 the measure

Continue until reach center

- Original measure $\mu(B) = Z(\beta_B)$
- After k steps measure $Z(\beta_k) = Z(\beta_B) r_1 r_2 \cdots r_k$, where $r_i \overset{\text{iid}}{\sim} \text{Unif}([0, 1])$
- Recall $\beta_{B'}$ index of center
- Let ℓ be number of steps until hit center

$$\ell := \min\{k : Z(\beta_B) r_1 \cdots r_k < Z(\beta_{B'})\} - 1$$

Question: what is the distribution of ℓ?

Logarithms

Recall if $U \sim \text{Unif}([0,1])$**,**

$$-\ln U \sim \text{Exp}(1)$$

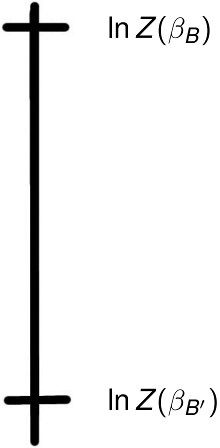
Since

$$\frac{Z(\beta_k)}{Z(\beta_B)} \sim r_1 r_2 \cdots r_k, \text{ where } r_i \overset{\text{iid}}{\sim} \text{Unif}([0,1]),$$

Consider the points

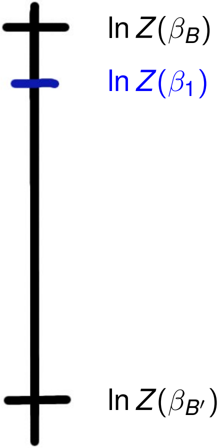
$$P_i = -\ln\left(\frac{Z(\beta_k)}{Z(\beta_B)}\right) \sim e_1 + e_2 + \cdots + e_k, \text{ where } e_i \overset{\text{iid}}{\sim} \text{Exp}(1)$$

The Poisson point process



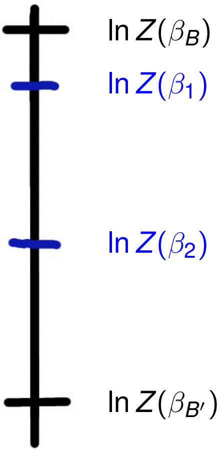
- Better to work in $\ln Z(\beta_i)$ space
- Recall: $U \sim \text{Unif}([0, 1]) \Rightarrow -\ln U \sim \text{Exp}(1)$
- In log space, each step moves down Exp(1)
- Result: a Poisson point process from $\ln Z(\beta_B)$ to $\ln Z(\beta_{B'})$

The Poisson point process



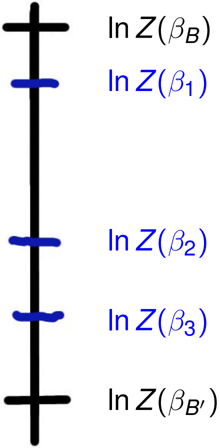
- Better to work in $\ln Z(\beta_i)$ space
- Recall: $U \sim \text{Unif}([0,1]) \Rightarrow -\ln U \sim \text{Exp}(1)$
- In log space, each step moves down Exp(1)
- Result: a Poisson point process from $\ln Z(\beta_B)$ to $\ln Z(\beta_{B'})$

The Poisson point process



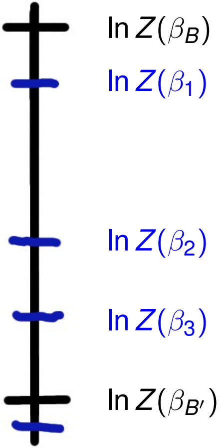
- Better to work in $\ln Z(\beta_i)$ space
- Recall: $U \sim \text{Unif}([0,1]) \Rightarrow -\ln U \sim \text{Exp}(1)$
- In log space, each step moves down Exp(1)
- Result: a Poisson point process from $\ln Z(\beta_B)$ to $\ln Z(\beta_{B'})$

The Poisson point process



- Better to work in $\ln Z(\beta_i)$ space
- Recall: $U \sim \text{Unif}([0,1]) \Rightarrow -\ln U \sim \text{Exp}(1)$
- In log space, each step moves down Exp(1)
- Result: a Poisson point process from $\ln Z(\beta_B)$ to $\ln Z(\beta_{B'})$

The Poisson point process



- Better to work in $\ln Z(\beta_i)$ space
- Recall: $U \sim \text{Unif}([0,1]) \Rightarrow -\ln U \sim \text{Exp}(1)$
- In log space, each step moves down Exp(1)
- Result: a Poisson point process from $\ln Z(\beta_B)$ to $\ln Z(\beta_{B'})$

The result

Output of TPA:

- $\ell \sim \text{Pois}(\ln(Z(\beta_B)/Z(\beta_{B'})))$

Output of A/R:

- $H \sim \text{Bin}(n, Z(\beta_{B'})/Z(\beta_B))$

Repeating the Poisson point process

Suppose run the Poisson point process twice

- Result also Poisson point process rate 2 instead of rate 1

Now run k times

- Result also Poisson point process rate k instead of rate 1
- Final answer $\text{Pois}(k \ln(Z(\beta_{B'})/Z(\beta_B)))$
- Divide by k, result close to $\ln[Z(\beta_{B'})/Z(\beta_B)]$
- Exponentiate, result close to $Z(\beta_{B'})/Z(\beta_B)$
- Can use Chernoff's Bound to choose k large enough

Bounding the tails

Theorem

Let $p = Z(\beta_{B'})/Z(\beta_B)$. For $p < \exp(-1)$ and $\epsilon < .3$, after

$$k = 2\left\lceil \ln \frac{1}{p} \right\rceil \left(\frac{3}{\epsilon} + \frac{1}{\epsilon^2} \right) \ln \frac{1}{2\delta}$$

runs, each of which uses on average $\ln(1/p)$ samples, the output $\hat{p}$ satisfies:

$$\mathbb{P}((1+\epsilon)^{-1} \leq \hat{p}/p \leq 1+\epsilon) > 1 - \delta.$$

Bonus: Approximate for all parameters simultaneously

Can cut Poisson point process at any point:

Right half still Poisson point process
Yields omnithermal approximation

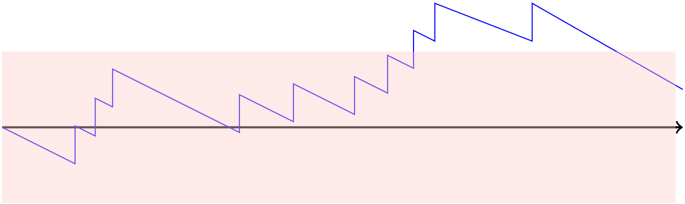
- Approximate $Z(\beta)/Z(\beta_{B'})$ for all $\beta \in [\beta_{B'}, \beta_B]$ at same time
- Number of runs still same:

$$k = 2\left[\ln \frac{1}{p}\right]\left(\frac{3}{\epsilon} + \frac{1}{\epsilon^2}\right)\ln \frac{1}{2\delta}$$

Proof idea:

Poisson process

- Let $N(t)$ be rate r Poisson process
- $N(t) - rt$ is a right continuous martingale
- Omnithermal approximation valid means did not drift too far away from 0



Examples

Example 1: Mixture Gaussian Spikes

Multimodal toy example

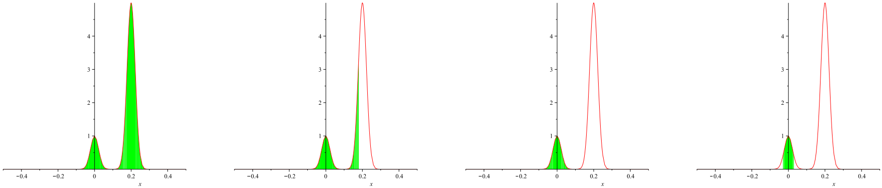
- Prior uniform over cube
- Likelihood mixture of two normals
- Small spike centered at $(0, 0, \ldots, 0)$
- Large spike centered at $(0.2, 0.2, \ldots, 0.2)$

$$p_\theta \sim \text{Unif}([-1/2, 1/2]^d)$$

$$L(\theta) = 100 \prod_{i=1}^{d} \frac{1}{\sqrt{2\pi}u} \exp\left(-\frac{(\theta_i - 0.2)^2}{2u^2}\right) + \prod_{i=1}^{d} \frac{1}{\sqrt{2\pi}v} \exp\left(-\frac{\theta_i^2}{2v^2}\right)$$

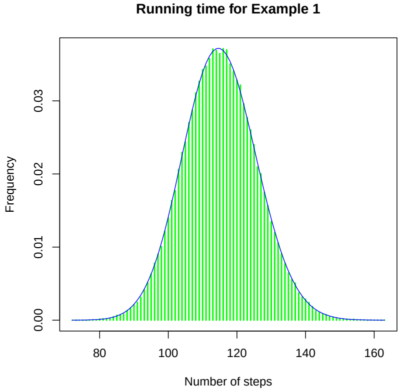
Parameter truncation

Create family by limiting distance to center of small spike



Running time results

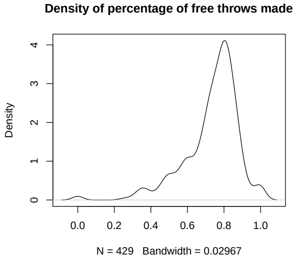
Problem: $d = 20$, $u = .01$, $v = .02$
True value: $\ln(1/p) = 115.0993$
Algorithm (10^5 runs): $\ln(1/p) \approx 115.10321$



Example 2: Beta-binomial model

Hierarchical model

- Data set: free throw numbers for 429 NBA players '08-'09
- Example data point: Kobe Bryant made 483 out of 564
- Model: number made by player i is $\text{Bin}(n_i, p_i)$
- n_i are known, $p_i \sim \text{Beta}(a, b)$
- Hyperparameters a and b, $a \sim 1 + \text{Exp}(1)$, $b \sim 1 + \text{Exp}(1)$



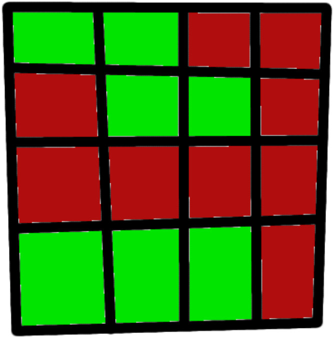
Again use parameter trunctation

Goal: find integrated likelihood

- Use β to limit distance from mode
- 2-D Unimodal problem so sampling easy
- True value (via numerical integration) -1577.250
- After 10^5 runs -1577.256

Example 3: Ising model

Besag[1974] modeled soil plots as good (green) or bad (red)



$h(x) = 13$ (# adj like colored plots)

$$\pi(x) = \frac{\exp(2\beta h(x))}{Z(\beta)}$$

parameter β is inv temp

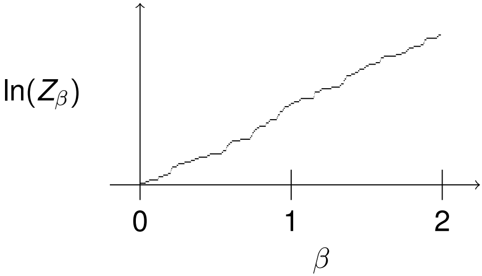
Integrated likelihood for Ising

Parameter space one dimensional

$$Z = \int_0^\infty p_\beta(b) \frac{\exp(2bh(x))}{Z(b)}\, db,$$

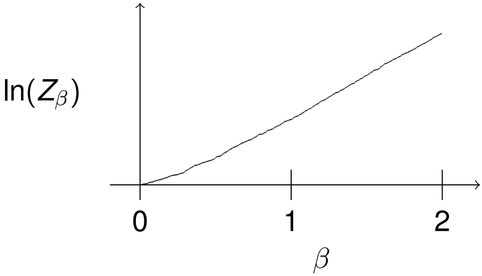
easy to do numerically if you know $Z(\beta)$ over $(0, \infty)$.

Use omnithermal approximation



One run of TPA

Use omnithermal approximation



Sixteen runs of TPA

Connection to MCMC

Several sampling methods use temperatures

- Simulated annealing
- Simulated tempering

TPA easy for these problems

- Can speed up chain by giving well balanced cooling schedule

Randomized adaptive cooling schedule: M-Chebyshev

Product Estimator [1]

Cooling schedule fixed

$$\beta_{B'} = \beta_0 < \beta_1 < \beta_2 < \cdots < \beta_k = \beta_B$$

- Find k independent unbiased estimates:

$$\mathbb{E}[\hat{a}_i] = \frac{\mu(A(\beta_i))}{\mu(A(\beta_{i+1}))}$$

- Make $\hat{a}$ product of $\hat{a}_i$:

$$\mathbb{E}[\hat{a}] = \mathbb{E}\left[\prod_{i=0}^{k-1} \hat{a}_i\right] = \frac{\mu(A(\beta_0))}{\mu(A(\beta_k))}$$

How accurate is product estimator?

Suppose

$$\frac{\text{SD}(\hat{a}_i)}{\mathbb{E}[\hat{a}_i]} \leq \epsilon'.$$

Then

$$\frac{\text{SD}(\hat{a})}{\mathbb{E}[\hat{a}]} \leq \sqrt{e^{k\epsilon'^2} - 1} \approx \epsilon'\sqrt{k}$$

So if $\epsilon' = \epsilon/\sqrt{k}$,

$$\frac{\text{SD}(\hat{a})}{\mathbb{E}[\hat{a}]} \approx \epsilon$$

M-Chebyshev schedules

Comes from Štefankovič, Vigoda, Vempala 2009 [4]

Definition

A set of estimators $\hat{b}_i$ is *M-Chebyshev* if for all i, $\text{SD}(\hat{b}_i)/\mathbb{E}[b_i] \leq \sqrt{M}$.

Procedure:

1. For each i, draw $Mk\epsilon^2$ copies of $\hat{b}_i$
2. Average to get $\hat{a}_i$
3. Multiply $\hat{a}_i$ to get $\hat{a}$

Result: need $Mk^2\epsilon^2$ samples to get good approximation

Gibbs distributions

Definition

A *Gibbs distribution* is of the form:

$$\pi(\{x\}) = \frac{\exp(-\beta H(x))}{Z(\beta)},$$

where $H(x)$ is a function of the configuration.

SVV for Gibbs

In [4],

Theorem

For a Gibbs distribution with $H(x) \in \{0, \ldots, n\}$, there exists a set of M-Chebyshev estimators for $M = 3 \cdot 10^6$ of length

$$O^* \left(\sqrt{\ln(Z(0)/Z(\infty))} \right).$$

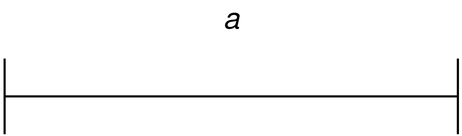
Corollary

An (ϵ, δ) randomized algorithm exists that requires $3 \cdot 10^6 [\ln(Z(0)/Z(\infty))] \epsilon^{-2} \ln(1/\delta)$ (times some log factors) samples.

Building a better set of estimators

Ideas:

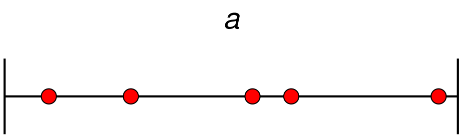
1. Use TPA to break problem in half
2. Recursively find short M-Chebyshev schedule for each half
3. Use TPA with M-Chebyshev schedule to get omnithermal approximation
4. Use omnithermal approximation to get new short M-Chebyshev schedule for entire problem



Building a better set of estimators

Ideas:

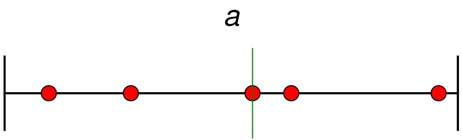
1. Use TPA to break problem in half
2. Recursively find short M-Chebyshev schedule for each half
3. Use TPA with M-Chebyshev schedule to get omnithermal approximation
4. Use omnithermal approximation to get new short M-Chebyshev schedule for entire problem



Building a better set of estimators

Ideas:

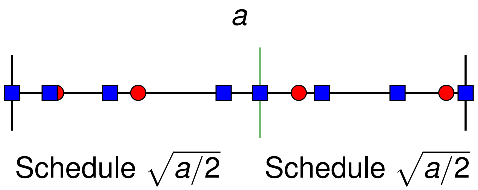
1. Use TPA to break problem in half
2. Recursively find short M-Chebyshev schedule for each half
3. Use TPA with M-Chebyshev schedule to get omnithermal approximation
4. Use omnithermal approximation to get new short M-Chebyshev schedule for entire problem



Building a better set of estimators

Ideas:

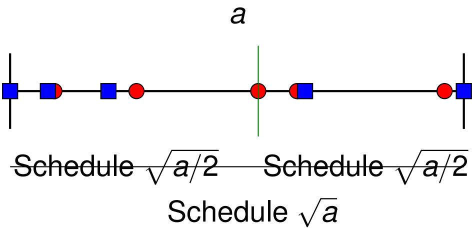
1. Use TPA to break problem in half
2. Recursively find short M-Chebyshev schedule for each half
3. Use TPA with M-Chebyshev schedule to get omnithermal approximation
4. Use omnithermal approximation to get new short M-Chebyshev schedule for entire problem



Building a better set of estimators

Ideas:

1. Use TPA to break problem in half
2. Recursively find short M-Chebyshev schedule for each half
3. Use TPA with M-Chebyshev schedule to get omnithermal approximation
4. Use omnithermal approximation to get new short M-Chebyshev schedule for entire problem



Conclusions

Randomized adaptive cooling schedules

- Guaranteed performance bound for MC integration
- (No variance estimate or unknown derivatives appear)
- Speed: $2[\ln(1/p)]^2$ much better than previous methods
- Speed*: $100[\ln(1/p)]$ even better for Gibbs distributions

Future directions

- Extend $\ln(1/p)$ method to non-Gibbs distributions
- Improve constant of 100

References

M. Jerrum, L. Valiant, and V. Vazirani,
Random generation of combinatorial structures from a uniform distribuiton
Theoret. Comput. Sci., **43**, 169–188, 1986

M. Huber and S. Schott,
Using TPA for Bayesian inference
Bayesian Statistics 9

J. Skilling,
Nested sampling for general Bayesian computation,
Bayesian Anal., **1**(4), 833–860, 2006

D. Štefankovič, S. Vempala and E. Vigoda,
Adaptive Simulated Annealing: A Near-Optimal Connection between Sampling and Counting,
J. of the ACM, **56**(3), 1–36, 2009