Geometric Data analysis

Random walks, Sampling, Volume

A. Chalkis

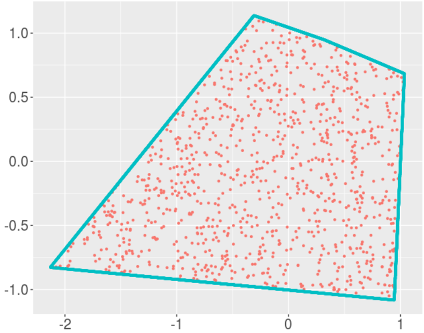
Dept Informatics & Telecoms, National Kapodistrian U. Athens
ATHENA Research & Innovation Center, Greece

Autumn 2020

Outline

Outline

- We represent a distribution Q with a Probability Density Function (PDF) $\pi(x),\ x \in K$, where K is the support of $\pi(x)$.
- The support is the subset of $\mathbb{R}^n$ which $\pi(x)$ does not map to zero.

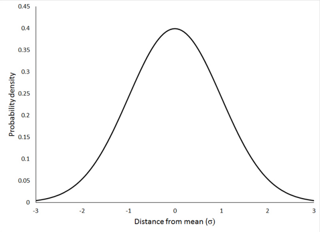


Here the support is the polytope P and the distribution Q is the uniform distribution over P, i.e. $\pi(x) = 1/\text{vol}(P)$.

- When a random variable follows X a distribution Q with PDF $\pi(x)$ then,

$$\Pr[X \in A] = Q(A) = \int_A \pi(x)dx, \text{ where } A \subseteq K.$$

- $\int_K \pi(x)dx = 1$.
- A function $f : \mathbb{R}^d \to \mathbb{R}_+$ induces a PDF $\pi(x) \propto f(x)$ when there is (possibly unknown) *normalizing constant* C such that $\pi(x) = f(x)/C$.



$$f(x) = e^{-x^2/2\sigma^2}$$

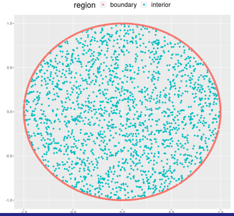
- When $\pi(x) \propto f(x)$ we say that $\pi(x)$ is proportional to $f(x)$.

Outline

Uniform Sampling from the hypersphere

- To sample uniformly from the boundary of a hypersphere with radius r:
  1. Sample d numbers $g_1, \ldots, g_d$ from $\mathcal{N}(0,1)$.
  2. The point $v = r(g_1, \ldots, g_d)/\sqrt{\sum g_i^2}$ is uniformly distributed on the surface of the d-dimensional hypersphere with radius r and centered at the origin.
- To sample uniformly from the interior of a hypersphere with radius r:
  1. Sample a point $v \sim \mathcal{U}(\partial B_d)$ and $u \sim \mathcal{U}(0,1)$.
  2. The point $p = ru^{1/d} v$ is uniformly distributed in the interior of the hypersphere with radius r and centered at the origin.

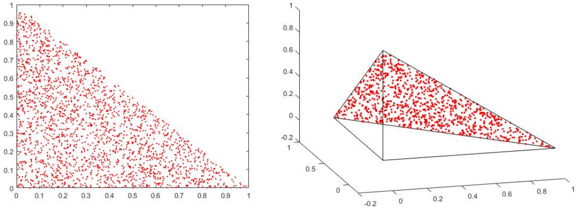
To pick a random direction through point $p \in \mathbb{R}^d$ we sample from the surface of a hypersphere centered at p.



Uniform Sampling from the simplex

1. [Smith and Tromble'04]:
    - Generate distinct: $x_0 < x_1 < \cdots < x_{d+1} \in \mathbb{N}^*$. Return y: $y_i = \dfrac{x_i - x_{i-1}}{M}$, $i = 1, \ldots, d+1$. M: largest integer.
    - To guarantee distinct choice we use a variation of Bloom filter.
    - Sampling one point takes $O(d \log d)$.

2. [Rubinstein and Melamed'98]:
    - Generate independent unit-exponential random variables, $X_1, \cdots, X_{d+1}$. Return $Y \in \mathbb{R}^{d+1}$: $Y_i = X_i / \sum_{i=1}^{d+1} X_i$.
    - Sampling one point takes $O(d)$.



Outline

Outline

Monte Carlo integration

The problem is the computation of a multidimensional definite integral,

$$I = \int_P f(\mathbf{x}) d\mathbf{x}$$

Given $\mathbf{x}_1, \ldots, \mathbf{x}_N$ uniformly distributed samples from P,

$$R_N = V \frac{1}{N} \sum_{i=1}^{N} f(\mathbf{x}_i), \; V = \text{vol}(P) = \int_P d\mathbf{x}$$

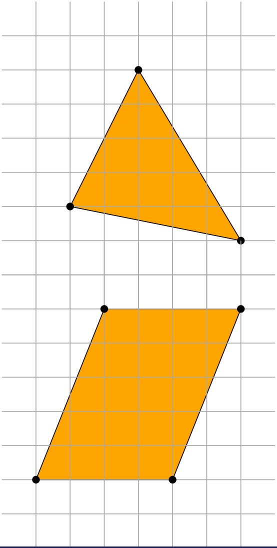
Then $\lim_{N \to \infty} R_N = I$

In general for $\mathbb{E}[f(\mathbf{x})] = \int_P f(\mathbf{x}) \pi_P(\mathbf{x}) d\mathbf{x}$ sample N i.i.d. points from π_P and $\mathbb{E}[f(\mathbf{x})] \approx \frac{1}{N} \sum_{i=1}^{N} f(\mathbf{x}_i)$.

Volume

Some easy cases

Some elementary polytopes have determinantal formulas.

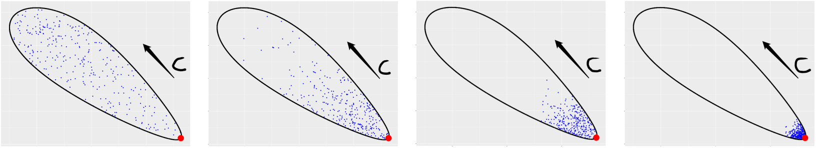


$$\begin{vmatrix} 1 & 2 & 1 \\ 3 & 6 & 1 \\ 6 & 1 & 1 \end{vmatrix} /2! = 11$$

$$\begin{vmatrix} 2 & 5 \\ 4 & 0 \end{vmatrix} = 20$$

Problem: Minimize a linear function $f(\mathbf{x}) = \mathbf{c} \cdot \mathbf{x}$ in body K.

Answer: Sample from $\pi_T(\mathbf{x}) \propto e^{-\mathbf{c}\cdot\mathbf{x}/T}$, for $T = T_0 > \cdots > T_l$.



T_0 T_1 T_2 T_3

A sample from π_{T_l} is ϵ-close to the optimal solution with high probability.

Outline

Acceptance-rejection sampling

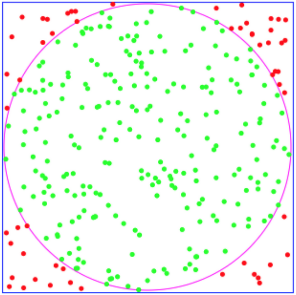
- Let $\pi(x) = f(x)/C, \; x \in \mathbb{R}^d$, where $f(x)$ is an *unnormalized* density and $C \in \mathbb{R}$ a normalizing constant.
- Let $h(x)$ a PDF that can be simulated by some known method and $f(x) \leq kh(x)$, where $k \in \mathbb{R}$ is a constant.

To obtain a random variate from $\pi(x)$,

1. Generate a candidate Z from $h(x)$ and a value u from $\mathcal{U}(0,1)$, the uniform distribution on $(0,1)$.
2. **If** $u \leq f(Z)/kh(Z)$ **return** Z.
3. **Otherwise** goto 1.

Acceptance-rejection sampling
Drawbacks

- Sampling/rejections techniques (sample from bounding box) fail in high dimensions

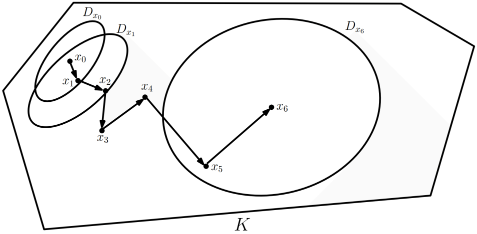


$$\frac{\mathrm{vol}(\textit{unitball})}{\mathrm{vol}(\textit{unitcube})} = O((1/d)^d)$$

Markov Chain Monte Carlo sampling

A MCMC sampling algorithm is applied on a continuous state space $K \subseteq \mathbb{R}^d$. The algorithm,

- Starts at a point $x_0 \in K$.
- When being at the point x_i moves to the next point x_{i+1} according to a transition kernel $p_x(A)$.
- The transition kernel of a MCMC algorithm gives the probability to jump from x to a set $A \subseteq K$.
- For example $p_x(K) = 1$.



Markov Chain Monte Carlo sampling

To sample from a density $\pi(x)$ define a random walk on a continuous state space with a transition kernel $p_x(A)$ such that,

1. [Convergence]

$$\int_P p_x(A)\pi(x)dx = \int_A \pi(y)dy$$

   Then $\pi(x)$ is called target density.

2. [Uniqueness] $\lim_{n\to\infty} p_x^n(A) = \int_A \pi(y)dy$, where

$$p_x^n(A) = \int_P p_x^{n-1}(y)p_y(A)dy,$$

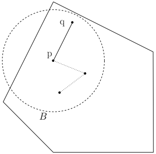
   the transition kernel of the n-th iteration.

[Understanding the Metropolis-Hastings Algorithm, '95].

Ball walk

Ball Walk(P, p, δ, f): Polytope $P \subset \mathbb{R}^d$, point $p \in P$, radius δ, $f : \mathbb{R}^d \rightarrow \mathbb{R}_+$

1. Pick a uniform random point x from the ball of radius δ centered at p
2. **return** x with probability $\min\left\{1, \frac{f(x)}{f(p)}\right\}$; **return** p with the remaining probability.

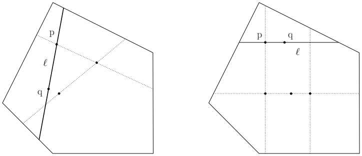


- When the density is not restricted to a body then the algorithm is known as the **Metropolis-Hastings** algorithm.
- **Task**: write the pseudocode for the special case of uniform sampling.

Hit-and-Run

Hit and Run(P, p, f): Polytope $P \subset \mathbb{R}^d$, point $p \in P$, $f : \mathbb{R}^d \to \mathbb{R}_+$

1. Pick a line ℓ through p.
2. **return** a random point on the chord $\ell \cap P$ chosen from the distribution $\pi_{\ell,f}$ restricted in P.

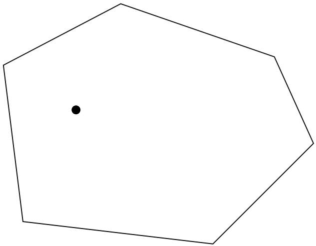


- When the density is not restricted to a body then the algorithm samples from $\pi_{\ell,f}$.
- **Task**: write the pseudocode for the special case of uniform sampling.
- **Q**: How can we compute the $\ell \cap P$?

Billiard walk - Uniform case

BW(P, p_i, τ, R) [Polyak'14]

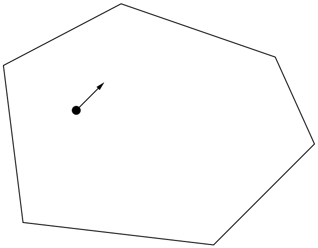
1. Generate the length of the trajectory $L = -\tau \ln \eta$, $\eta \sim U(0,1)$.
2. Pick a uniform direction v to define the trajectory.
3. When the trajectory meets a boundary with internal normal s, $||s|| = 1$, the direction is changed as $v \leftarrow v - 2 < v, s > s$.
4. **return** the end of the trajectory as p_{i+1}. If the number of reflections exceeds R **return** $p_{i+1} = p_i$.



Billiard walk - Uniform case

BW(P, p_i, τ, R) [Polyak'14]

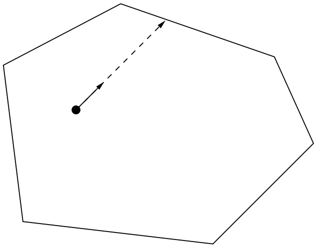
1. Generate the length of the trajectory $L = -\tau \ln \eta$, $\eta \sim U(0,1)$.
2. Pick a uniform direction v to define the trajectory.
3. When the trajectory meets a boundary with internal normal s, $||s|| = 1$, the direction is changed as $v \leftarrow v - 2 < v, s > s$.
4. **return** the end of the trajectory as p_{i+1}. If the number of reflections exceeds R **return** $p_{i+1} = p_i$.



Billiard walk - Uniform case

BW(P, p_i, τ, R) [Polyak'14]

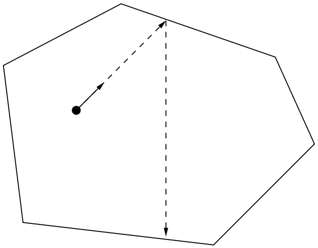
1. Generate the length of the trajectory $L = -\tau \ln \eta$, $\eta \sim U(0, 1)$.
2. Pick a uniform direction v to define the trajectory.
3. When the trajectory meets a boundary with internal normal s, $||s|| = 1$, the direction is changed as $v \leftarrow v - 2 < v, s > s$.
4. **return** the end of the trajectory as p_{i+1}. If the number of reflections exceeds R **return** $p_{i+1} = p_i$.



Billiard walk - Uniform case

BW(P, p_i, τ, R) [Polyak'14]

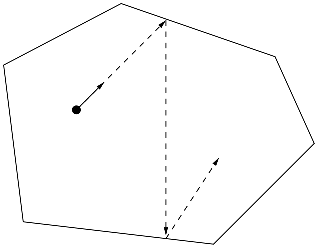
1. Generate the length of the trajectory $L = -\tau \ln \eta$, $\eta \sim U(0,1)$.
2. Pick a uniform direction v to define the trajectory.
3. When the trajectory meets a boundary with internal normal s, $||s|| = 1$, the direction is changed as $v \leftarrow v - 2 < v, s > s$.
4. **return** the end of the trajectory as p_{i+1}. If the number of reflections exceeds R **return** $p_{i+1} = p_i$.



Billiard walk - Uniform case

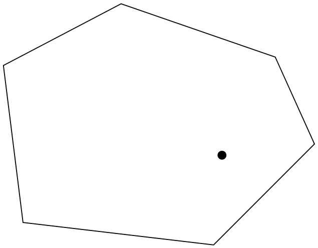
BW(P, p_i, τ, R) [Polyak'14]

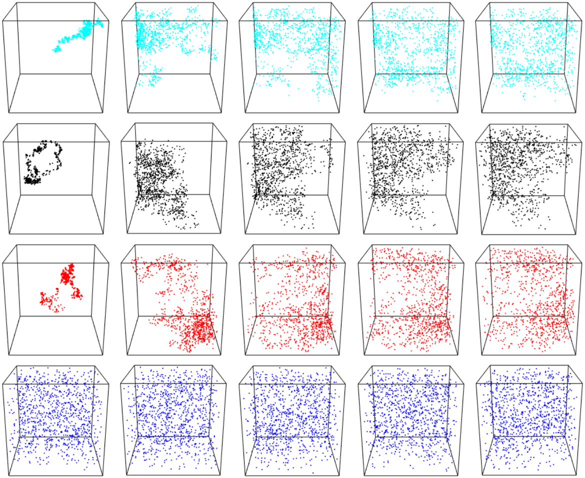
1. Generate the length of the trajectory $L = -\tau \ln \eta$, $\eta \sim U(0, 1)$.
2. Pick a uniform direction v to define the trajectory.
3. When the trajectory meets a boundary with internal normal s, $||s|| = 1$, the direction is changed as $v \leftarrow v - 2 < v, s > s$.
4. **return** the end of the trajectory as p_{i+1}. If the number of reflections exceeds R **return** $p_{i+1} = p_i$.



BW(P, p_i, τ, R) [Polyak'14]

1. Generate the length of the trajectory $L = -\tau \ln \eta$, $\eta \sim U(0,1)$.
2. Pick a uniform direction v to define the trajectory.
3. When the trajectory meets a boundary with internal normal s, $||s|| = 1$, the direction is changed as $v \leftarrow v - 2 < v, s > s$.
4. **return** the end of the trajectory as p_{i+1}. If the number of reflections exceeds R **return** $p_{i+1} = p_i$.





- Uniform sampling from the hypercube $[-1, 1]^{200}$ and projection to $\mathbb{R}^3$.
- Rows: Ball Walk, Coordinate Directions Hit and Run, Random Directions Hit and Run, Billiard Walk.
- Columns: walk length, $\{1, 50, 100, 150, 200\}$

Limitations of BW and HnR

- Their mixing time is $O^*(d^3)$ for log-concave distributions.
- Their performance is crucially affected by the starting point.

- Typically a warm start is required. A distribution S is said to be M-warm with respect to the distribution Q if,

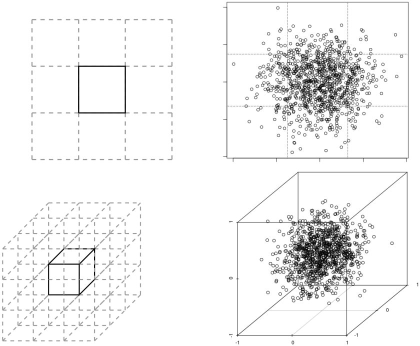
$$M = \sup_{A \in P} \frac{S(A)}{Q(A)}$$

- They perform better when the distribution is (approximately) isotropic. A distribution Q is said to be isotropic if

$$\mathbb{E}_Q[X] = 0, \text{ and } \mathbb{E}_Q[XX^T] = I_d$$

Limitations of BW and HnR

- They spend many steps around the mode of the distribution.
- Consider the spherical Gaussian centered at the origin with $\sigma^2 = 0.1$.
- The mode of the PDF in two dimensions it is $1/9$ and in three dimensions it is only $1/27$ of the volume of the cube.



- **Q**: In 20 dimensions?

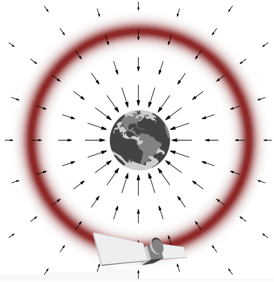
Typical set

- The problem that motivates us is to compute the expectation of a function f, say $\mathbb{E}_\pi[f]$, which reduces to the integral,

$$\mathbb{E}_\pi[f] = \int_P f(x)\pi(x)dx \tag{1}$$

- In high dimensions a probability density, $\pi(x)$, will concentrate around its mode (local maximum of $\pi(x)$).
- Contributions to the expectation are determined by the product of density and volume, $\pi(x)dx$.
- The points with dominant contribution to (1) concentrate in a neighborhood called the typical set.

Typical set



- We can interpret the mode of the target density as a massive planet and the gradient of the target density as that planet's gravitational field.
- The typical set becomes the space around the planet through which we want a satellite to orbit .

Hamiltonian Monte Carlo

- HMC defines trajectories that guide the walk inside the typical set.
- The choice of the derivative of $\pi(x)$ at the current point $\mathbf{p}$ of the walk would be wrong as it points directly towards its mode.
- The Hamiltonian dynamics behind HMC operate on a position vector $\mathbf{p}$ and a velocity $\mathbf{v}$.

Hamiltonian Monte Carlo

- The system is described by a function of $\mathbf{p}$ and $\mathbf{v}$ known as the Hamiltonian,

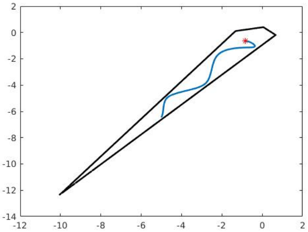
$$H(\mathbf{p},\mathbf{v}) = U(\mathbf{p}) + K(\mathbf{v}) = -\log(\pi(\mathbf{p})) + \frac{1}{2}|\mathbf{v}|^2.$$

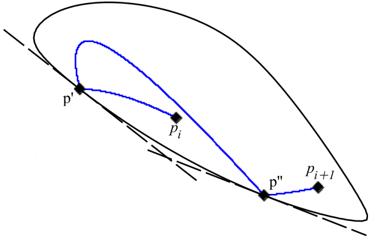
To sample from π, one has to solve the following system of Ordinary Differential Equations (ODE):

$$\begin{aligned} \frac{d\mathbf{p}}{dt} &= \frac{\partial H(\mathbf{p},\mathbf{v})}{\partial \mathbf{v}} \\ \frac{d\mathbf{v}}{dt} &= -\frac{\partial H(\mathbf{p},\mathbf{v})}{\partial \mathbf{p}} \end{aligned} \quad \Rightarrow \quad \begin{cases} \frac{d\mathbf{p}(t)}{dt} = \mathbf{v}(t) \\ \frac{d\mathbf{v}(t)}{dt} = \nabla \log(\pi(\mathbf{p})) \end{cases} \tag{2}$$

Hamiltonian Monte Carlo

- When the density is restricted in a Polytope P then HMC walks on trajectories inside P.





Outline

MCMC Convergence Diagnostics

How can we evaluate the quality of a sample obtained by a random walk?

- [Convergence diagnostics for Markov chain Monte Carlo, Vivekananda Roy, '19].
- [Revisiting the Gelman-Rubin Diagnostic, Dootika Vats, Christina Knudson, '20].

A MCMC convergence diagnostic can also be used as a termination criterion for sampling.

Outline

Package volesti

- open source, written in C++
  https://github.com/GeomScale/volume_approximation
- R package in CRAN
  https://CRAN.R-project.org/package=volesti.
- Python interface.
- since 2018.

Problem	currently	soon	description
volume computation	✓		8 algo. / thousands of dimensions / fastest practical estimation
sampling distributions			
uniform / gaussian/ Exp	✓		4 algo. / thousands of dimensions
log-concave densities	✓		HMC / Langevin Diffusion
convex optimization			
Semidefinite Programming	✓		beating `SDPA` / working to improve
Linear Programming		✓	goal: best open source
multivariate integration			
simple MC integration	✓		hundreds of dimensions
importance sampling		✓	goal: best open source approximation
Preprocessing	✓		3 rounding algo. / 4 MCMC diagnostics

GeomScale/volesti on Google Summer of Code 2020

Mentoring organization

- https://geomscale.github.io/
- https://summerofcode.withgoogle.com/organizations/5673184117915648/

Three student projects this year:

1. Sampling log-concave densities.
2. Convex optimization.
3. Uniform sampling / metabolic networks in biology.

Stan & cobra

- stan is a platform for statistical modeling.
- Provides HMC implementations.
- https://mc-stan.org/.

- cobra is the state-of-the-art package for the analysis of metabolic networks.
- Provides three random walks for uniform / Gaussian sampling from convex polytopes.
- https://github.com/opencobra.

Outline

Computing the exact volume of P,

- is #P-hard for all the representations [DyerFrieze'88]
- is open if both H- and V- representations available
- is APX-hard (oracle model) [Elekes'86]

Theorem

[Dyer, Frieze, Kannan'91] For any convex body P and any $0 \leq \epsilon,\ \delta \leq 1$, there is a randomized algorithm which computes an estimate V s.t. with probability $1-\delta$ we have $(1-\epsilon)vol(P) \leq V \leq (1+\epsilon)vol(P)$, and the number of oracle calls is $poly(d, 1/\epsilon, log(1/\delta))$.

- Using randomness, we can go from an exponential approximation to an arbitrarily small one.

Outline

Multiphase Monte Carlo

Let a sequence of functions $\{f_0, \ldots, f_m\}$, $f_i : \mathbb{R}^d \to \mathbb{R}$. Then,

$$\mathrm{vol}(P) = \int_P dx = \int_P f_m(x)dx \, \frac{\int_P f_{m-1}(x)dx}{\int_P f_m(x)dx} \cdots \frac{\int_P f_0(x)dx}{\int_P f_1(x)dx} \, \frac{\int_P dx}{\int_P f_0(x)dx}$$

Then select f_i s.t.,

- The number of phases, m, is as small as possible.
- Each integral ratio can be efficiently estimated by sampling from $\pi \propto f_i$ restricted to P (using geometric random walks).
- There is a closed formula for $\int_P f_m(x)dx$.

Multiphase Monte Carlo

Let a sequence of functions $\{f_0, \ldots, f_m\}$, $f_i : \mathbb{R}^d \to \mathbb{R}$. Then,

$$\mathrm{vol}(P) = \int_P dx = \int_P f_m(x)dx \; \frac{\int_P f_{m-1}(x)dx}{\int_P f_m(x)dx} \cdots \frac{\int_P f_0(x)dx}{\int_P f_1(x)dx} \; \frac{\int_P dx}{\int_P f_0(x)dx}$$

Then select f_i s.t.,

- The number of phases, m, is as small as possible.
- Each integral ratio can be efficiently estimated by sampling from $\pi \propto f_i$ restricted to P (using geometric random walks).
- There is a closed formula for $\int_P f_m(x)dx$.

complexity = #phases × #points per phase × cost per point

State-of-the-art

Authors-Year	Complexity (oracle calls)	f_i	random walk
[Dyer, Frieze, Kannan'91]	$O^*(d^{23})$	Indicator function of a ball	grid walk
[Kannan, Lovasz, Simonovits'97]	$O^*(d^5)$	Indicator function of a ball	ball walk
[Lovasz, Vempala'03]	$O^*(d^4)$	Exponential	hit-and-run
[Cousins, Vempala'15]	$O^*(d^3)$	Spherical Gaussians	ball walk

State-of-the-art

Authors-Year	Complexity (oracle calls)	f_i	random walk
[Dyer, Frieze, Kannan'91]	$O^*(d^{23})$	Indicator function of a ball	grid walk
[Kannan, Lovasz, Simonovits'97]	$O^*(d^5)$	Indicator function of a ball	ball walk
[Lovasz, Vempala'03]	$O^*(d^4)$	Exponential	hit-and-run
[Cousins, Vempala'15]	$O^*(d^3)$	Spherical Gaussians	ball walk

- Can not be implemented as they are due to large constants in the complexity and pessimistic theoretical bounds.

State-of-the-art

Authors-Year	Complexity (oracle calls)	f_i	random walk
[Dyer, Frieze, Kannan'91]	$O^*(d^{23})$	Indicator function of a ball	grid walk
[Kannan, Lovasz, Simonovits'97]	$O^*(d^5)$	Indicator function of a ball	ball walk
[Lovasz, Vempala'03]	$O^*(d^4)$	Exponential	hit-and-run
[Cousins, Vempala'15]	$O^*(d^3)$	Spherical Gaussians	ball walk

- Can not be implemented as they are due to large constants in the complexity and pessimistic theoretical bounds.

Practical algorithms:

- Follow theory but make practical adjustments (experimental).

State-of-the-art

Authors-Year	Complexity (oracle calls)	f_i	random walk
[Dyer, Frieze, Kannan'91]	$O^*(d^{23})$	Indicator function of a ball	grid walk
[Kannan, Lovasz, Simonovits'97]	$O^*(d^5)$	Indicator function of a ball	ball walk
[Lovasz, Vempala'03]	$O^*(d^4)$	Exponential	hit-and-run
[Cousins, Vempala'15]	$O^*(d^3)$	Spherical Gaussians	ball walk

- Can not be implemented as they are due to large constants in the complexity and pessimistic theoretical bounds.

Practical algorithms:

- Follow theory but make practical adjustments (experimental).
- [Emiris, Fisikopoulos'14] Sequence of balls + coordinate hit-and-run.

State-of-the-art

Authors-Year	Complexity (oracle calls)	f_i	random walk
[Dyer, Frieze, Kannan'91]	$O^*(d^{23})$	Indicator function of a ball	grid walk
[Kannan, Lovasz, Simonovits'97]	$O^*(d^5)$	Indicator function of a ball	ball walk
[Lovasz, Vempala'03]	$O^*(d^4)$	Exponential	hit-and-run
[Cousins, Vempala'15]	$O^*(d^3)$	Spherical Gaussians	ball walk

- Can not be implemented as they are due to large constants in the complexity and pessimistic theoretical bounds.

Practical algorithms:

- Follow theory but make practical adjustments (experimental).
- [Emiris, Fisikopoulos'14] Sequence of balls + coordinate hit-and-run.
- [Cousins, Vempala'16] Spherical Gaussians + hit-and-run

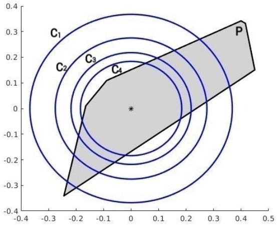
Outline

Contributions

- **M**ultiphase **M**onte **C**arlo algorithm with statistical tests.
- Sequence of scaled copies of any convex body (generalization of sequence of balls).
- Faster practical algorithm for zonotopes and V-polytopes besides H-polytopes. Performs computations in:
  - thousands of dimensions for H-polytopes in few hours.
  - 100 dimensions for zonotopes and V-polytopes in $\leq$ 1hour.

Multiphase Monte Carlo

- Let $C_m \subseteq \cdots \subseteq C_1$ a sequence of concentric balls intersecting P, s.t. $C_m \subseteq P \subseteq C_1$.



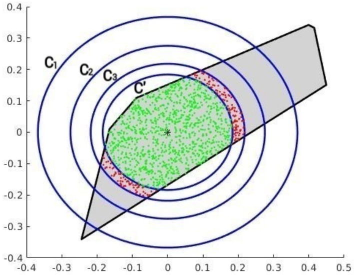
- Construct a sequence of balls intersecting P, then:

$$\mathrm{vol}(P) = \mathrm{vol}(P \cap C_m)\frac{\mathrm{vol}(P \cap C_{m-1})}{\mathrm{vol}(P \cap C_m)} \cdots \frac{\mathrm{vol}(P \cap C_1)}{\mathrm{vol}(P \cap C_2)}\frac{\mathrm{vol}(P)}{\mathrm{vol}(P \cap C_1)}$$

Ratio estimation

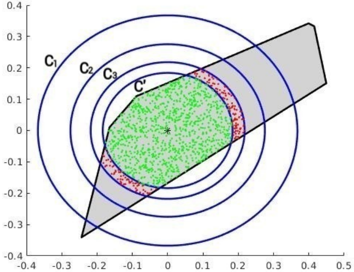
- Estimate $r_i = \frac{\text{vol}(P \cap C_{i+1})}{\text{vol}(P \cap C_i)}$ within some target relative error ϵ_i.

- Estimate $r_i = \frac{\text{vol}(P \cap C_{i+1})}{\text{vol}(P \cap C_i)}$ within some target relative error ϵ_i.
- Sample N uniform points from $P_i = C_i \cap P$ and count points in $P_{i+1} = C_{i+1} \cap P \subseteq P_i$.



Ratio estimation

- Estimate $r_i = \frac{\text{vol}(P \cap C_{i+1})}{\text{vol}(P \cap C_i)}$ within some target relative error ϵ_i.
- Sample N uniform points from $P_i = C_i \cap P$ and count points in $P_{i+1} = C_{i+1} \cap P \subseteq P_i$.



- Keep each ratio bounded, then $N = O(1/\epsilon_i^2)$ points suffices.

Keep r_i bounded

Statistical test

Given convex bodies $P_1 \supseteq P_2$, we define two statistical tests:

[**U-test**(P_1, P_2)] H_0: $\text{vol}(P_2)/\text{vol}(P_1) \geq r + \delta$
[**L-test** (P_1, P_2)] H_0: $\text{vol}(P_2)/\text{vol}(P_1) \leq r$

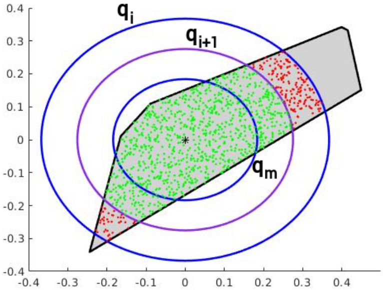
- The **U-test** and **L-test** are successful iff both H_0 are rejected.
- If both **U-test** and **L-test** are successful then $r_i = \text{vol}(P_{i+1})/\text{vol}(P_i) \in [r, r + \delta]$, with high probability.

How to fix the sequence of balls

Let C_d the unit ball. Given $C_i = q_i C_d,\ C_m = q_m C_m$, with $q_i > q_m$,

Problem: Compute a new ball C_{i+1} with radius $q_m < q_{i+1} < q_i$, s.t. $\frac{\text{vol}(P \cap C_{i+1})}{\text{vol}(P \cap C_i)} \in [r, r + \delta]$.

Answer: binary search to compute $q_{i+1} \in [q_m, q_i]$ until both **U-test**$(P \cap C_i, P \cap C_{i+1})$ and **L-test**$(P \cap C_i, P \cap C_{i+1})$ are successful.



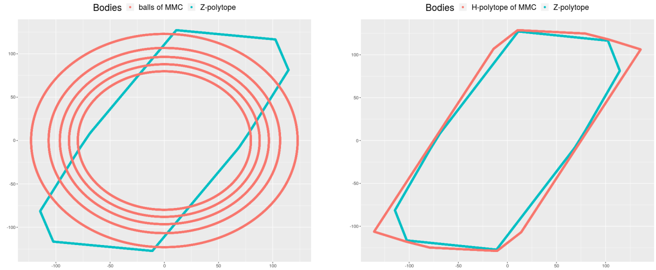
Bound #phases

- Our algorithm terminates with constant probability.
  - Bound the probability that the construction of the sequence of bodies in MMC fails.
- #phases $m = O\Big(log(\text{vol}(P)/\text{vol}(P \cap C_m)) \Big)$.
- If the body we use in MMC is a good fit to P the $\text{vol}(P \cap C_m)$ increases and the number of phases m decreases.

Use any body in MMC

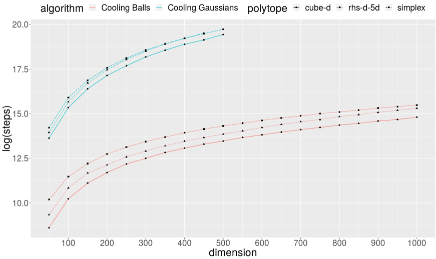
MMC for zonotopes

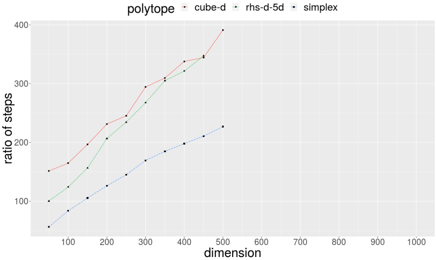
- Use the generators of a zonotope P to define a centrally symmetric H-polytope that is a good fit to P.



$r = 0.8, \ r + \delta = 0.85$. #phases: Left $m = 5$. Right $m = 1$.

Comparison with other software



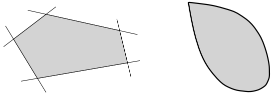


Outline

Problem

Given P a convex body in $\mathbb{R}^n$:

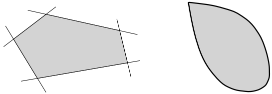
- minimize a convex function f in P (**convex optimization**).



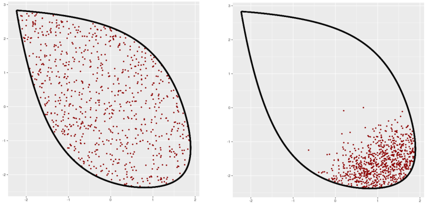
Problem

Given P a convex body in $\mathbb{R}^n$:

- minimize a convex function f in P (**convex optimization**).



Goal: Randomized approximation algorithms based on sampling from P with geometric random walks.



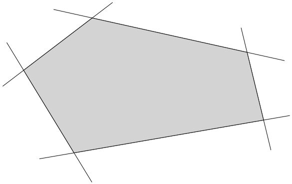
Convex optimization - Special cases

Linear program

- The objective function is linear $f(\mathbf{x}) = \mathbf{c} \cdot \mathbf{x}$.
- The body is given as an intersection of m half-spaces.

- The objective function is linear $f(\mathbf{x}) = \mathbf{c} \cdot \mathbf{x}$.
- The body is given as an intersection of m half-spaces.

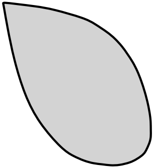
H-polytope : $P = \{x \mid Ax \leq b,\ A \in \mathbb{R}^{m \times n},\ b \in \mathbb{R}^m\}$



- The objective function is linear $f(\mathbf{x}) = \mathbf{c} \cdot \mathbf{x}$.
- The body is given as a Linear Matrix Inequality (LMI).

- The objective function is linear $f(\mathbf{x}) = \mathbf{c} \cdot \mathbf{x}$.
- The body is given as a Linear Matrix Inequality (LMI).

Spectrahedron : $P = \{x \mid A_0 + x_1 A_1 + \cdots + x_d A_d \succeq 0\}$,
where A_i: symmetric matrices, $B \succeq 0$: B is positive semidefinite

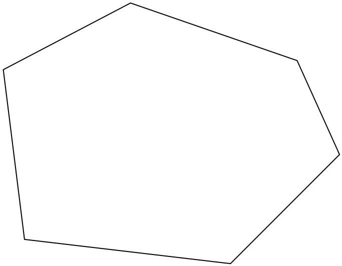


Outline

Cutting planes

Dabbene, Shcherbakov, Polyak, 10'

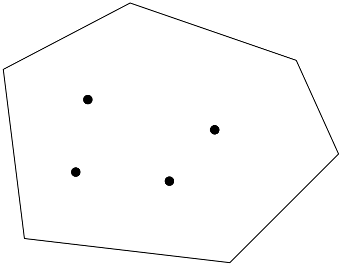
- Input: convex body K, objective function c.



Cutting planes

Dabbene, Shcherbakov, Polyak, 10'

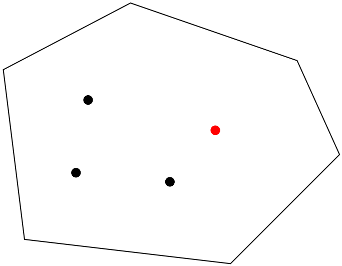
- Input: convex body K, objective function c.
- Sample N points under the uniform distribution.



Cutting planes

Dabbene, Shcherbakov, Polyak, 10'

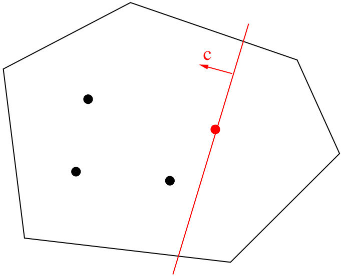
- Input: convex body K, objective function c.
- Sample N points under the uniform distribution.
- Find the point x minimizing the objective function.



Cutting planes

Dabbene, Shcherbakov, Polyak, 10'

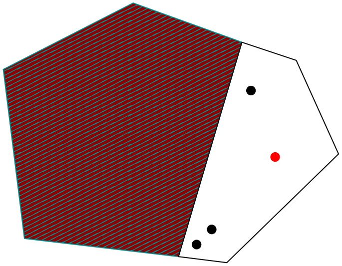
- Input: convex body K, objective function c.
- Sample N points under the uniform distribution.
- Find the point x minimizing the objective function.
- Cut the convex body at x.



Cutting planes

Dabbene, Shcherbakov, Polyak, 10'

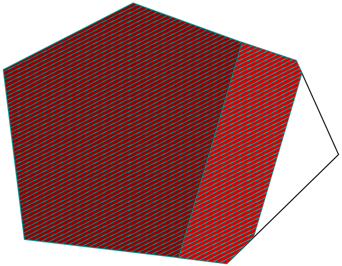
- Input: convex body K, objective function c.
- Sample N points under the uniform distribution.
- Find the point x minimizing the objective function.
- Cut the convex body at x.
- Repeat I times.



Cutting planes

Dabbene, Shcherbakov, Polyak, 10'

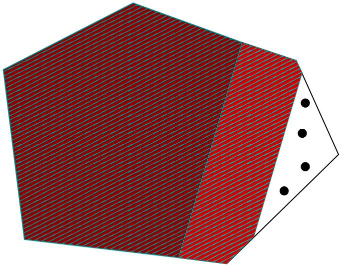
- Input: convex body K, objective function c.
- Sample N points under the uniform distribution.
- Find the point x minimizing the objective function.
- Cut the convex body at x.
- Repeat I times.



Cutting planes

Dabbene, Shcherbakov, Polyak, 10'

- Input: convex body K, objective function c.
- Sample N points under the uniform distribution.
- Find the point x minimizing the objective function.
- Cut the convex body at x.
- Repeat I times.



Cutting planes

- Let $rB_d \subseteq K \subseteq RB_d$.
- The expected number of phases s.t. $|f_l - f^*| < \epsilon$ is,

$$l = \left\lceil \frac{1}{\ln(N+1)} d \ln(R/\epsilon) \right\rceil = O^*(d)$$

- Total number of uniform points minimized for $N = 1$.
- Total cost,

$$\left\lceil d \ln(R/\epsilon) \right\rceil \times \text{ cost per point}$$

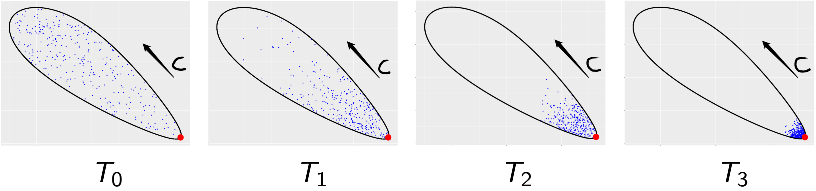
Outline

Exponential sampling and Simulated Annealing

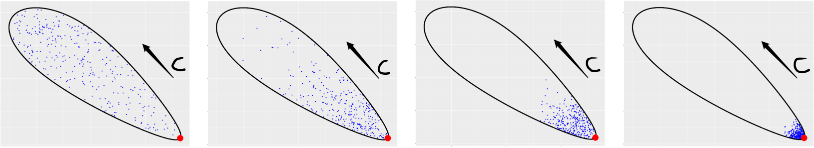
Kalai, Vempala, 06'

Problem: Minimize a linear function $f(\mathbf{x}) = \mathbf{c} \cdot \mathbf{x}$ in body K.

Answer: Sample from $\pi_T(\mathbf{x}) \propto e^{-\mathbf{c}\cdot\mathbf{x}/T}$, for $T = T_0 > \cdots > T_l$.



A sample from π_{T_l} is ϵ-close to the optimal solution with high probability.



T_0 T_1 T_2 T_3

- The sequence $T_0 > \cdots > T_I$ is fixed s.t. the L_2 norm of π_{T_i} w.r.t. $\pi_{T_{i+1}}$ is bounded by a constant,

$$\|\pi_{T_i}/\pi_{T_{i+1}}\| = \mathbb{E}_{\pi_{T_i}}\left[\frac{d\pi_{T_i}}{d\pi_{T_{i+1}}}\right] = \int_K \frac{\pi_{T_i}(x)}{\pi_{T_i+1}(x)}\pi_{T_i}(x)dx = O(1)$$

- Then π_{T_i} is a warm start for $\pi_{T_{i+1}}$ (Hit-and-Run).

- Starting with $T_0 = R$ (uniform distribution is a warm start).

- Starting with $T_0 = R$ (uniform distribution is a warm start).
- Set $T_i = T_0(1 - \frac{1}{\sqrt{d}})^i,\ i = 1, \dots, I$ (T_i is a warm start for T_{i+1}).

- Starting with $T_0 = R$ (uniform distribution is a warm start).
- Set $T_i = T_0(1 - \frac{1}{\sqrt{d}})^i, \ i = 1, \ldots, I$ (T_i is a warm start for T_{i+1}).
- Knowing that for a temp. T,

$$\mathbb{E}_{\pi_T}[\mathbf{c} \cdot \mathbf{x}] \leq dT + \min_{x \in K} \mathbf{c} \cdot \mathbf{x}$$

- Starting with $T_0 = R$ (uniform distribution is a warm start).
- Set $T_i = T_0(1 - \frac{1}{\sqrt{d}})^i,\ i = 1, \dots, I$ (T_i is a warm start for T_{i+1}).
- Knowing that for a temp. T,

$$\mathbb{E}_{\pi_T}[\mathbf{c} \cdot \mathbf{x}] \leq dT + \min_{x \in K} \mathbf{c} \cdot \mathbf{x}$$

- $I = O^*(\sqrt{d})$ phases suffices to obtain a solution $|f_I - f^*| \leq \epsilon$.

- Starting with $T_0 = R$ (uniform distribution is a warm start).
- Set $T_i = T_0(1 - \frac{1}{\sqrt{d}})^i,\ i = 1, \dots, I$ (T_i is a warm start for T_{i+1}).
- Knowing that for a temp. T,

$$\mathbb{E}_{\pi_T}[\mathbf{c} \cdot \mathbf{x}] \leq dT + \min_{x \in K} \mathbf{c} \cdot \mathbf{x}$$

- $I = O^*(\sqrt{d})$ phases suffices to obtain a solution $|f_I - f^*| \leq \epsilon$.
- No sequence of distributions $\propto f_i(\mathbf{c} \cdot \mathbf{x})$ can, in general, solve the problem in less than $\Omega(\sqrt{d})$ phases.