Monte Carlo for Optimization

- Two applications for Monte-Carlo Simulation

Simulate biological processes

(1) partical swarm optimization

(2) genetic algorithm

Particle swarm

a flock of birds, a school of fish, bees, ants ...

in this case, we'll look at birds

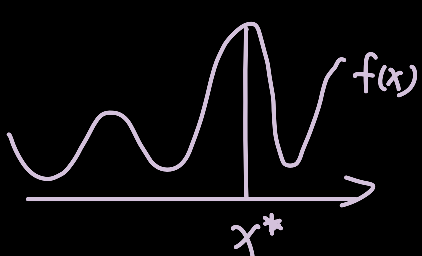
find food in a field, each bird is a particle

$$\vec{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_d \end{pmatrix}$$

d-dimensional coordinates.

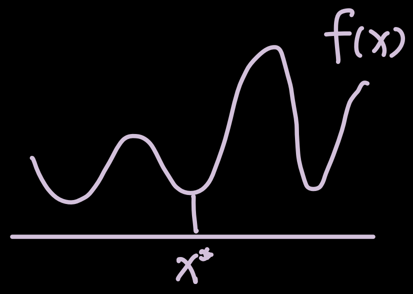
(e.g. $d=2$ in the birds' case)

Optimize Objective Function $f(\vec{x})$



max (value) 1-Dim

or

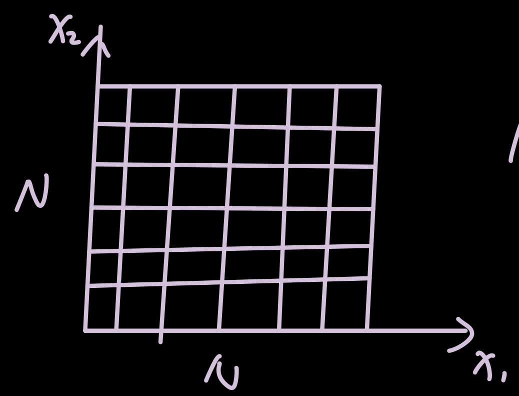


min (loss)

1- Dim

Multi- modal

In 2d, we can partition



N^2

d-dim : N^d

e.g. d = 100

Cannot search exhaustively when d is large.

Need more efficient search algorithm

↳ exploration ⟶ exploitation

We may not understand $f(\vec{x})$ very much and do not have $f'(x)$

Exploration : multiple modes, randomness for exploration.

Back to Particle Swarm :

m particles, $i = 1, 2, \dots, m$

$\vec{x}_i(t)$: position of particle i at time t   2D-coordinate

$\vec{v}_i(t)$: velocity of particle i at time t.   2D-change

① randomly initialize $(\vec{x}_i(t), \vec{v}_i(t), i = 1, \dots, m)$

discretizing time ... $\Delta t = 1$

$$\vec{x}_{(t+1)} = \vec{x}_i(t) + \vec{v}_i(t+1)$$

$$\vec{v}_i(t+1) = \underbrace{w}_{\text{inertia} < 1}\vec{v}_i(t) + \underbrace{c_1}_{\text{cognitive coefficient}}\underbrace{r_1}_{\text{random unif }[0,1]}(\underbrace{p\,best_i}_{\substack{\text{particle's best place}\\ \text{so far (up to time t)}}} - \vec{x}_i(t)) + \underbrace{c_2}_{\text{social coefficient}}\underbrace{r_2}_{\text{randomness}}(\underbrace{g\,best_i}_{\substack{\text{group / global best so far}\\ \text{(up to time t)}}} - \vec{x}_i(t))$$

* $Pbest_i$: the bird's best location
* c_1 : bird is aware of its best location
* gbest : out of all the birds, the best place so far
* C_2 : birds are aware of group's best.

Genetic Algorithm

max $f(\vec{x}) \longrightarrow$ fitness

$\vec{x}$: genetic representation

binary string (DNA sequence)

1. Initialization: population of random strings

2. Selection: use $f(\vec{x})$ to select members of population to be parents

3. Cross-over and mutation:
   to generate new generation then select again.

0 0 0 0 0 0 0 $\xrightarrow{\text{cross-over}}$
1 1 1 1 1 1 1

random position then make swap

prob. of mutation

1 1 1 0 0 0 0 $\xrightarrow{\text{mutation}}$
0 0 0 1 1 1 1

1 1 1 0 1 0 1 (the new generation)

4. Go back to "selection" step

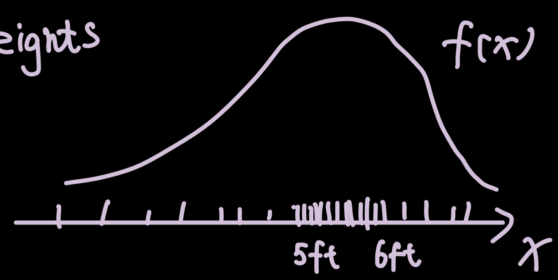
Part 2. Monte Carlo Integration

Expectation

population of ($N \approx \infty$) points $\xrightarrow{\text{random Sampling}}$

a random point $X \sim f(x)$

e.x. heights



$\{x_1, x_2, \ldots\ldots, x_N\}$

$E(X)$ = population average

$$= \frac{1}{N} \sum_{i=1}^{N} x_i$$

$$= \frac{1}{N} \sum_{\text{bins}(x,\, x+\Delta x)} X \quad N(x)$$

assumed height for all people in this bin

$N(x)$ ↓ # of people in bin $(x, x+\Delta x)$

ex: 6ft, 11 inch

$$= \sum_{\text{bins}} x \underbrace{\frac{N(x)}{N}}$$

proportion in $(x, x+\Delta x)$

$$= \sum_{\text{bins}} x f(x) \Delta x$$

$$\xrightarrow[0]{\Delta x} = \int x \cdot f(x)\, dx$$

$$E(X) = \int x f(x) d(x)$$

More generally

$$\underbrace{E(h(X))}_{\text{e.g. } h(X)=X^2} = \int h(x)\, \underbrace{f(x) \cdot d(x)}_{\text{the same}}$$

$$\mu = E(X)$$

Variance

$$Var(X) = E((X-\mu)^2) = \sigma^2$$

$$\text{population variance} = \frac{1}{N}\sum_{i=1}^{n}(x_i-\mu)^2$$

$$Var(h(X)) = E((h(X) - E(h(x)))^2)$$

same

Want to compute

$$I = E[h(X)] = \int h(x) f(x)\, dx$$

e.g.

$$= \int \sqrt{|x|} \cdot \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}\, dx$$

draw $X_1, X_2, X_3, X_4, \ldots X_n \sim f(x)$ independent

Monte Carlo Approximation

$$\hat{I} = \frac{1}{n}\sum_{i=1}^{n} h(X_i)$$

Population vs. sample

population of N ($\approx \infty$) points $\xrightarrow{\text{sample}}$ random sample of n points

↓   ↓

proportion

population average I $\xleftarrow{\text{estimate}}$ sample average $\hat{I}$

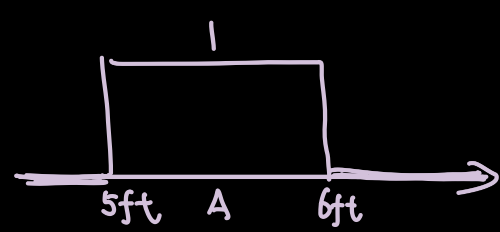
proportion

(fixed, not deterministic)   (random, fluctuates)

Special Case : avg of indicator function = proportion

$$h(x) = \begin{cases} 1 & x \in A \\ 0 & x \notin A \end{cases}$$

$$= 1_A(x) \quad \text{or} \quad 1(x \in A)$$



population average of $1(x_i \in A)$

$$= \frac{1}{N}\sum_{i=1}^{n} 1(x_i \in A)$$

$=$ population proportion $= P(A)$

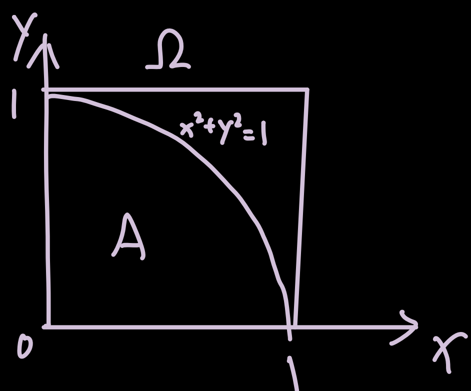
Ex. 1.

Want $I = \int_{-\infty}^{\infty} \underbrace{\sqrt{|x|}}_{h(x)} \underbrace{\frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}}_{f(x)} dx$

generate $X_1, X_2, \ldots X_n \sim N(0, 1)$ indep

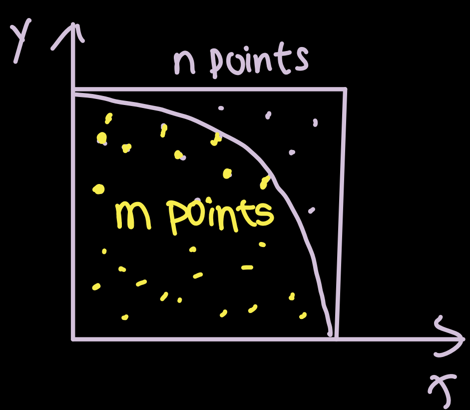
$\hat{I} = \frac{1}{n} \sum_{i=1}^{n} \sqrt{|X_i|}$ → random

Ex. 2. Approximate π



$P(A) = \frac{|A|}{|\Omega|} = \frac{\pi}{4}$

Monte-Carlo Method



$\frac{m}{n} \approx \frac{\pi}{4}$

frequency $\approx$ probability

$\hat{\pi} = \frac{4m}{n}$

Can think of this as

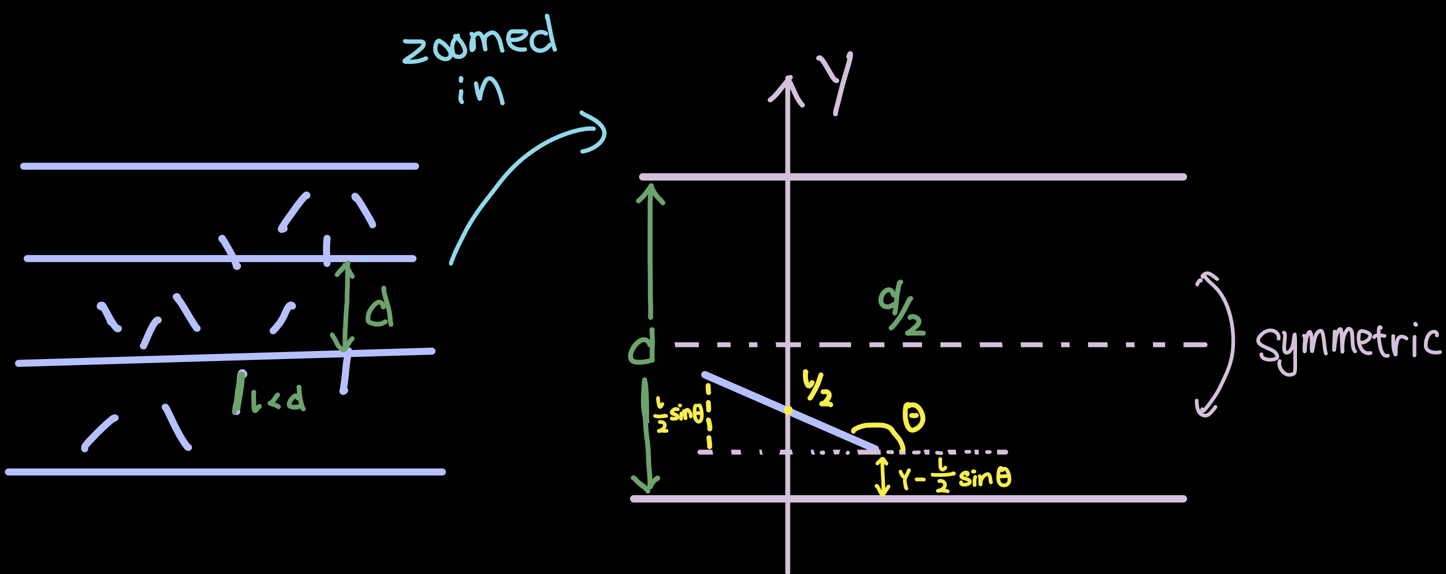
$$P(A) = E(1(X^2+Y^2 \le 1))$$

$$= \int_0^1 \int_0^1 1(x^2+y^2 \le 1)\, dx\, dy$$

$X, Y \sim \text{Unif}[0,1]$ indep

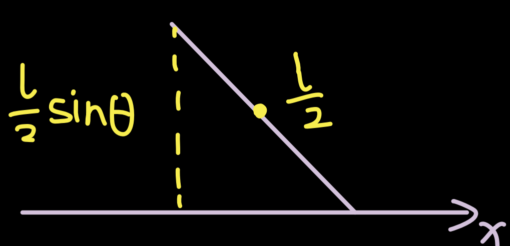
Buffon's Needle

Approximate π by dropping needles on a grid of parallel lines and calculating the probability that it will cross the line.

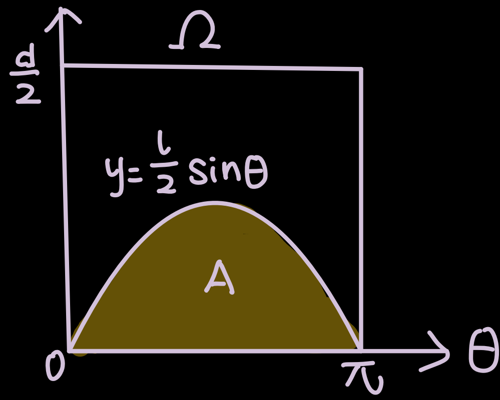


Y = center of needle $\sim \text{Unif}\left[0, \frac{d}{2}\right]$

Θ = angle $\sim \text{Unif}[0, \pi]$:



$$P\left(Y \le \frac{l}{2}\sin\theta\right) =$$



$$= \frac{|A|}{|\Omega|}$$

$$= \frac{\int_0^{\pi} \frac{l}{2}\sin\theta \, d\theta}{\frac{d}{2}\pi}$$

$$= \frac{\frac{l}{2}\left(-\cos\theta \Big|_0^{\pi}\right)}{\frac{d}{2}\pi}$$

$$= \frac{l}{\frac{d}{2}\pi}$$

$$= \frac{2l}{d\pi}$$ probability