ywu@stat.ucla.edu

OH: Tues, Thurs. 4-5PM, MS 8971

Coursework: weekly HW + final exam on last lecture

Topics: random number generators

Monte Carlo Integration

Importance sampling

Markov Chain Monte Carlo (MCMC)

(genetic algorithm, particle swamp)

Introduction

1940s 2nd WW Manhattan Project / atomic bombs

first computers

Monte Carlo computation

Fermi: neutron diffusion, sampling randomness

Ulam Von Neumman

Metropolis: "Monte Carlo" Metropolis algorithm (MCMC)

Why Monte Carlo?

(1) Simulation

(2) Integration

Example: $\int_0^1 f(x)\,dx$

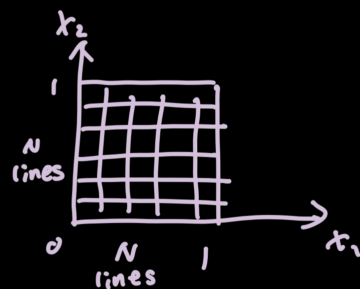
area under curve

$\approx \sum_x \underbrace{f(x)\,\Delta x}$

area of each rectangle

integral is defined as the limit of summation of areas as Δx approaches 0.

2-dimensional: $\int_0^1 \int_0^1 f(x_1, x_2)\, dx_1\, dx_2$



N^2 cells

d-dimensional: $\int_0^1 \int_0^1 \cdots \int_0^1 f(x_1, x_2 \ldots x_d)\, dx_1\, dx_2 \cdots dx_d$

N^d cells in high dimensions → crashes

$X = (x_1, x_2, x_3, \ldots x_d)$

$X^{(1)}, X^{(2)}, \ldots X^{(n)} \sim \text{Uniform}\,[0,1]^d$

randomly sample n pts w/ d coordinates and avg them

$\frac{1}{n}\sum_{i=1}^{n} f(X^{(i)})$

to approximate the integral

population → random sample
(N^d cells) (n points)
↓ ↘
1000^{100} 200

avoid curse of dimensionality

(3) Optimization

Gibbs distribution

$$\pi(x) = \frac{1}{Z} e^{\frac{-\varepsilon(x)}{T}}$$

$\varepsilon(x)$ → energy
T → temperature

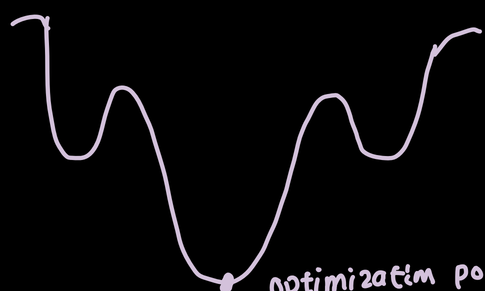
lower energy → higher probability $\pi(x)$ → more stable system

lower temp ↗

$\varepsilon(x)$ → energy of system

reduce temp → simulated annealing
→ converges to shortest path

to start from high temp. then to reduce temp. gradually



optimization point : lowest energy at certain temp.

AI / Machine Learning / Deep Learning

(1) Stochastic gradient descent

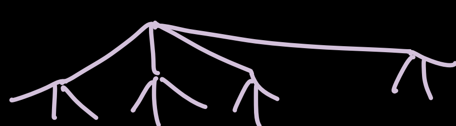
(2) generate models

(3) Alpha Go : Monte Carlo tree search

Game Tree

breadth →

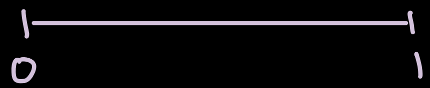
↓ depth



randomly sample some directions

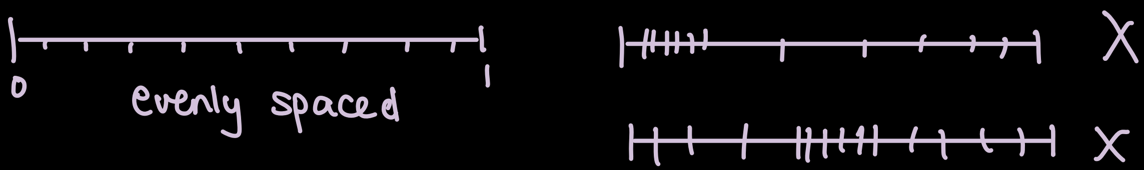
Part 1 : Random number generators

Uniform random number   Unif [0, 1]

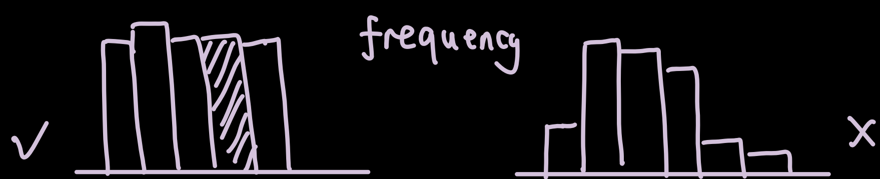


$U_1, U_2, \ldots, U_t, \ldots \sim$ Unif [0, 1] independently

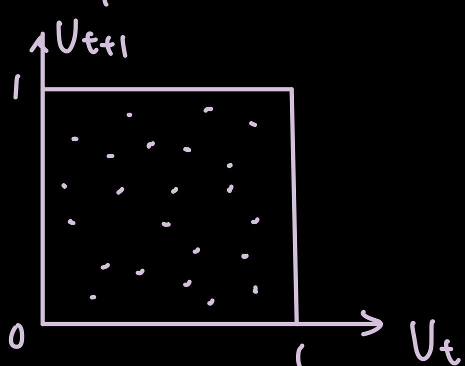
Scatter plot



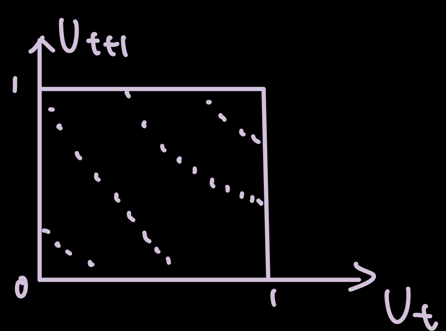
Histogram



2D Scatterplot

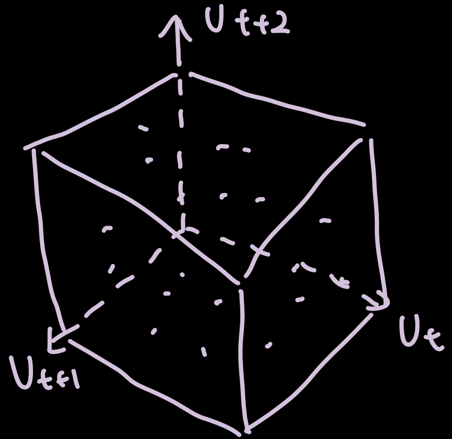


independent



dependency b/w U_t and U_{t+1}

3D



(U_1, U_2, U_3), (U_2, U_3, U_4), ...

(U_t, U_{t+1}, U_{t+2})

1. Linear Congruential Method

Start from X_0 (integer, seed)

Then iterate

$$\underbrace{X_{t+1} = aX_t + b \quad \boxed{\text{mod } M}}_{\text{carefully chosen integers}}$$

e.g. $a = 7^5$ $b = 0$, $M = 2^{31} - 1$

can run thru all numbers from 0 to $2^{31}-1$ not included

divide by M, take remainder

$7 = \boxed{2} \quad \text{mod } 5$

$X_t \in \{0, 1, 2, \ldots, M-1\}$

$U_t = \frac{X_t}{M} \sim \text{Unif}[0, 1]$

If M is large

Consider U_t as continuous real numbers

Example: $X_0 = 1 \quad X_{t+1} = 5X_t + 0 \pmod 7$

$$X_1 = (a \cdot X_0 + b) \bmod M$$
$$= (5 \cdot 1 + 0) \bmod 7 = 5$$

$$X_2 = (5 \cdot 5 + 0) \bmod 7 = 4$$

$$X_3 = (5 \cdot 4 + 0) \bmod 7 = 6$$

$$X_4 = (5 \cdot 6 + 0) \bmod 7 = 2$$

$$X_5 = (5 \cdot 2 + 0) \bmod 7 = 3$$

$$X_6 = (5 \cdot 3 + 0) \bmod 7 = 1$$

$$X_t = (1, 5, 4, 6, 2, 3, 1)$$

Pseudo - random $\Rightarrow$ deterministic not completely random

however, if you don't know the integer and seed we started with, you can treat X_t as random.

2. Inversion Method

Simulate Fair Coin

$$U \sim \text{Unif}[0, 1]$$

$$X = \begin{cases} \text{Head} & \text{if } U \geq \frac{1}{2} \\ \text{Tail} & \text{if } U < \frac{1}{2} \end{cases}$$

Simulate Fair Dice

$U \sim \mathrm{Unif}[0, 1]$