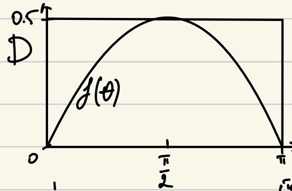
Lecture 9 - Monte Carlo

Buffon's needle (1777):



$f(\theta) = \frac{\sin\theta}{2}$

$0 \le \theta \le 180$, $0 \le D \le 1$, $D \le \frac{1}{2}\sin(\theta) \equiv \text{hit}$

$\Rightarrow P(\text{hit}) = \frac{\int_0^{\pi} f(\theta)\, d\theta}{0.5 \times \pi} = \frac{1}{\frac{\pi}{2}} = \frac{2}{\pi} \Rightarrow \pi \approx 2\, \frac{\text{drops}}{\text{hits}}$

Monte-Carlo: $\theta \sim U(0, \pi)$, $D \sim U(0, \frac{1}{2}) \Rightarrow p_{MC} = \frac{1}{N}\sum \#\text{hits} \approx p_{true} = \frac{\int_0^{\pi} f(\theta)\, d\theta}{\pi/2}$

$\mathbb{E}_p[f(x)] \approx \frac{1}{N}\sum_{i=1}^{N} f(x_i) = S, \quad x_i \sim p(x)$

$\int p(x) f(x)\, dx$

ξ - random variable $\Rightarrow$ sample mean $S = \frac{\xi_1 + \dots + \xi_n}{n}$

$S \xrightarrow{d} \mathcal{N}(\mu, \frac{\sigma^2}{N}) \Rightarrow \frac{S-\mu}{\frac{\sigma}{\sqrt{N}}} \sim N(0,1)$

$\mu = \mathbb{E}_p(f(x))$

$\sigma^2 = \mathbb{V}_p(f(x))$ Error $= \sigma/\sqrt{N}$

$X_1 \sim U(0, 0.8)$

1) $U \sim U(0,1)$

2) $X = 0 + (0.8 - 0)U$

$U(a, b)$

$U \sim U(0,1)$, $X = a + (b-a)U$

PDF: $f_X(x, a, b) = \frac{1}{a-b}$ CDF: $F_X(x, a, b) = \int_a^b x f(x)\, dx$

PDF $f(x) = \begin{cases} 2x, & 0 \le x \le 1 \\ 0, & \text{otherwise} \end{cases}$ $\Rightarrow$ CDF, $F(x) = \int_0^x 2y\, dy = x^2$, $0 \le x \le 1$

$U \sim U(0,1)$, $X = F^{-1}(U) = \sqrt{U}$

Lecture 10 - Advanced Sampling Methods

Acceptance-Rejection

$f(x) = \sqrt{\frac{2}{\pi}} e^{-\frac{x^2}{2}}, \quad x \ge 0$

1) pick $g(x)$ close to $f(x)$, $g(x) = e^{-x}$

2) choose M s.t. $\forall x$, $f(x) \le M g(x)$, $M = \sqrt{\frac{2e}{\pi}}$

3) sampling:

- $\rightarrow$ sample x from $g(x)$
- $\rightarrow$ generate sample $u \sim U(0,1)$
- $\rightarrow$ accept sample if $u \le \frac{f(x)}{Mg(x)}$

Efficiency: $P(U \le \frac{f(x)}{Mg(x)}) = \int g(x) \int \mathbb{1}\{u \le \frac{f(x)}{Mg(x)}\}\, du\, dx = \mathbb{E}_g\left[\frac{f(x)}{Mg(x)}\right] = \int \frac{f(x)}{M} dx = \frac{1}{M}$

Simplex

$K-1$-dimensional simplex $A_{K-1} = \{x = (x_1, \dots, x_k)^T \mid \sum_{i=1}^{k} x_i = 1 \wedge \forall i,\ 0 \le x_i \le 1\} \rightarrow$ Dirichlet distribution

1) Sampling from $K-1$ dimensional cube

2) Accept points that lie inside the simplex

Efficiency: $\frac{1}{M} = \frac{1}{(k-1)!} \xrightarrow{k \to \infty} 0$

Importance Sampling

$\mathbb{E}_p[f(x)] = \int_R f(x)p(x)\,dx$, choose $q(x)$ - spans superset of R, $\sim p(x)$

$q(x) \neq 0$: $\mathbb{E}_p[f(x)] = \int_R q(x)\frac{f(x)p(x)}{q(x)}\,dx = \mathbb{E}_q[h(x)] \approx \frac{1}{N}\sum_{i=1}^{N} h(x_i^q)$

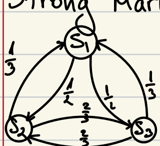
$X_i = a_i U_i$, $U_i \sim_{iid} U(0,1)$, $p(x) = \prod_{i=1}^{N}\frac{1}{a_i}$, $q(u) = \prod_{i=1}^{N}\frac{1}{v_i} \Rightarrow \frac{p(x)}{q(x)} = \prod_{i=1}^{N}\frac{v_i}{a_i}$

$P(S) = 0.01\% = 10^{-4}$, $P(D) = 1 - P(S) = 0.9999$, $P(T^+|S) = 0.99$, $P(T^+|D) = 0.01$

$$P(S|T^+) = \frac{P(T^+|S)P(S)}{P(T^+)} = \frac{P(T^+|S)P(S)}{P(T^+|S)P(S)+P(T^+|D)P(D)} = \frac{0.000099}{0.010098} \approx 0.009 / 0.9\%$$

Lecture 11 - Markov Chain Monte Carlo

Strong Markov Property: $P(X_{T+m} = j \mid X_k = x_k,\ 0 \le k \le T;\ X_T = i) = P(X_{T+m} = j \mid X_T = i)$



$P(X_4 = s_3 \mid X_1 = s_2, X_2 = s_1, X_3 = s_2) = P(X_4 = s_3 \mid X_3 = s_2) = \frac{2}{3}$

$P(X_{i+1} = s_3 \mid X_i = s_1) + P(X_{i+1} = s_2 \mid X_i = s_1) = 1$

$n = 10$: $P(s_1) = \frac{4}{10}$, $P(s_2) = \frac{3}{10}$, $P(s_3) = \frac{3}{10}$

$n = 100$: $P(s_1) = 0.25$, $P(s_2) = 0.375$, $P(s_3) = 0.375$

$$\begin{array}{c|ccc} & s_1 & s_2 & s_3 \\ \hline s_1 & 0 & \frac{1}{2} & \frac{1}{2} \\ s_2 & \frac{1}{3} & 0 & \frac{2}{3} \\ s_3 & \frac{1}{3} & \frac{2}{3} & 0 \end{array} = A$$

$\pi_t = [P(X_t = s_1) \quad P(X_t = s_2) \quad P(X_t = s_3)]$

$X_0 = s_1 \Rightarrow \pi_0 = [1\ 0\ 0]$ $\quad \pi_0 A = [0\ \frac{1}{2}\ \frac{1}{2}] = \pi_1$

$\pi_1 A = [\frac{1}{3}\ \frac{1}{3}\ \frac{1}{3}] = \pi_2 = \pi_0 A^2 \Rightarrow \pi_k = \pi_0 A^k$

$\pi A = \pi$, $Av = \lambda v$, $\lambda = 1$, $\sum_{i=1}^{n}\pi_i = 1 \wedge \forall i = \overline{1,n}\ (\pi_i \ge 0)$

Knapsack Problem Model: $\vec{a} = (a_0, \dots, a_{m-1}) \in \mathbb{N}^m$, $\vec{x} \in \{0,1\}^m$, $\vec{a}\cdot\vec{x} = \sum_{i=0}^{m-1} a_i x_i \le b$

Monte Carlo

$\vec{x} \sim U(\{0,1\}^m)$

$X_i = \begin{cases} 2^m & \vec{a}\cdot\vec{x} \le b \\ 0 & \text{otherwise} \end{cases}$

Markov Chain Monte Carlo

feasible solution $\vec{x} \in \Omega$

$p = \frac{1}{2}$ flip 1 vector component:

$\vec{x}' = \begin{cases} \vec{x} \in \Omega & \text{if } u \sim U(0,1) < \frac{1}{2} \\ (x_0, \dots, 1-x_i, \dots, x_{m-1}),\ i \sim U\{0, \dots, m-1\} & \text{otherwise} \end{cases}$

$y = x'$ if $\vec{a}\cdot\vec{x}' \le b$ otherwise $y = \vec{x}$

M on Ω

Irreducible and aperiodic on Ω

Uniform stationary distribution over Ω

Convergence after polynomial in m steps

Metropolis-Hastings Algorithm

sample from $f(\theta) = p(\text{data} \mid \theta)\, p(\theta)$

proposal distribution $q(\theta' \mid \theta)$

current state θ_0, $\forall t = 0, 1, \dots, T$ do:

1) $\theta' \sim q(\theta' \mid \theta_t)$
2) $\alpha(\theta_t, \theta') = \min\left\{ \dfrac{f(\theta')\, q(\theta_t \mid \theta')}{f(\theta_t)\, q(\theta' \mid \theta_t)}, 1 \right\}$
3) $U \sim \mathcal{U}(0, 1)$
4) $\theta_{t+1} = \begin{cases} \theta', & U \leq \alpha(\theta_t, \theta') \\ \theta_t, & \text{otherwise} \end{cases}$

Independence Sampler, $q(\theta' \mid \theta_t) = g(\theta') \Rightarrow \alpha(\theta_t, \theta') = \min\left\{ \dfrac{f(\theta')\, g(\theta_t)}{f(\theta_t)\, g(\theta')}, 1 \right\}$

↳ produces dependent samples

Random Walk Sampler, $q(\theta' \mid \theta_t) = q(\theta_t \mid \theta') \Rightarrow \alpha(\theta_t, \theta') = \min\left\{ \dfrac{f(\theta')}{f(\theta_t)}, 1 \right\}$

↳ usually $N(\theta, \sigma^2)$

Lecture 12 - Optimization & Planning

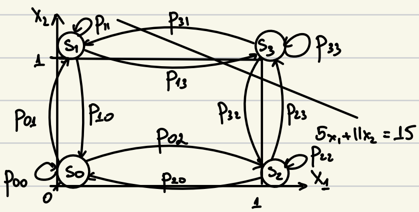
Tech Stock Returns → Metropolis-Hastings Random Walk Sampler

$P(X_A > 1\%, X_B > 1\%) = \int_1^\infty \int_1^\infty f(X_A, X_B)\, dX_A\, dX_B$

$\vec{X}_0 = (X_A^0, X_B^0)$, $\forall t = 0, 1, \dots, T$ do:

1. Generate sample $\vec{X}' = \vec{X}_t + \vec{Z}$, $\vec{Z} \sim \mathcal{N}(0, \sigma^2)$, $\sigma > 0$
2. Compute acceptance criterion $\alpha(\vec{X}_t, \vec{X}') = \min\{ f(\vec{X}') / f(\vec{X}_t), 1 \}$
3. Generate $u \sim \mathcal{U}(0, 1)$
4. Set $\vec{X}_{t+1} = \begin{cases} \vec{X}' & \text{if } u \leq \alpha(\vec{X}_t, \vec{X}') \\ \vec{X}_t & \text{otherwise} \end{cases}$

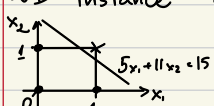
$\left\{ \begin{array}{l} 1.\ \text{Draw } Z_A, Z_B \sim \mathcal{N}(0, 1) \\ 2.\ \text{Let } \vec{Z} = (Z_A, Z_B) \\ 3.\ \text{Set } \vec{X}' = \vec{X}_t + \sigma \vec{Z} \end{array} \right.$



Knapsack Problem

build M on $\Omega = \{ \vec{x} \in \{0, 1\}^n : \vec{a} \cdot \vec{x} \leq b \}$

2D instance: $\vec{a} = \begin{pmatrix} 5 \\ 11 \end{pmatrix}$, $b = 15$



$\vec{a} \cdot \vec{x} = \begin{pmatrix} 5 \\ 11 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = 5x_1 + 11x_2 \leq 15$

invalid solution: $\vec{x}^{\,x} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$

$$T = \begin{pmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

item values: $\vec{v} = (v_0, v_1, \dots, v_{m-1}) \in \mathbb{N}^m$

$\vec{a} = (a_0, a_1, \dots, a_{m-1}) \in \mathbb{N}^n$

$\left\{ \begin{array}{l} \Omega = \{ \vec{x} \mid \vec{a} \cdot \vec{x} \leq b \} \\ \text{optimization objective: } \max_{x \in \Omega} \sum_{i=0}^{m-1} v_i x_i \end{array} \right.$

Metropolis-Hastings: $\alpha(\vec{x}, \vec{x}') = \min\{ f(\vec{x}') / f(\vec{x}), 1 \}$

$f(\vec{x}) = \frac{1}{Z} \exp(\beta \sum_{i=0}^{m-1} v_i x_i)$ ← Boltzmann Distribution, β – inverse temperature measure

$\Rightarrow \alpha(\vec{x}, \vec{x}') = \min\{1, \exp(\beta \sum_{i=0}^{m-1} v_i (x_i' - x_i))\}$ $\beta\downarrow$ = explore, $\beta\uparrow$ - exploit

Simulated Annealing, $\beta(t) = \log(t)$

$G(\vec{x})$, $\forall \vec{x} \in \Omega$ – speed up convergence when close to global optimum

$\alpha(\vec{x}, \vec{x}') = \min\{1, \exp(\log(t)(G(\vec{x}') - G(\vec{x})))\}$

Boltzmann distribution: $P(E) \propto e^{-\frac{E}{k_B T}}$ $\Rightarrow$ $T\uparrow$ = explore, $T\downarrow$ = exploit

Energy, $E(\vec{x})$ = cost function to minimize

Temperature, $T(t)$: control parameter, decreasing

$\left.\right\}$ $\alpha(\vec{x}', \vec{x}_t) = \min\{1, \exp(-\Delta E / T(t))\}$ where $\Delta E = E(\vec{x}') - E(\vec{x}_t)$

Algorithm:

- → initialization: high T_0, random solution $\vec{x}'$
- → propose: $\vec{x}'$
- → accept/reject: $\alpha(\vec{x}', \vec{x}_t)$
- → cooling: $T(t)\downarrow$
  - → exponential: $T(t) = T_0 \cdot a^t$
  - → logarithmic: $T(t) = \frac{T_0}{\ln t}$
  - → linear: $T(t) = T_0 - \beta_t$
  - → adaptive: dynamically tuned

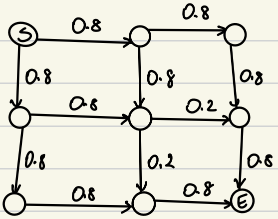
Travelling Salesperson

- → initial: some allowed ordering $A \to B \to C \to D$
- → swap order of neighboring cities (new ordering - allowed)
- → Metropolis exploration: $e^{-\frac{\Delta L}{T(t)}} > U$, $U \sim \mathcal{U}(0, 1)$
- → cooling: $T(t+1) = aT(t)$

Probabilistic Reachability (Reliability) Network

most reliable path:

- → compute $R(x) = \prod_{e \in edges} P(e)_{included} \cdot (1 - P(e))_{not\ included}$
- → flip random edge (include/exclude): x'
- → acceptance: $\alpha(x', x) = \exp(-\frac{\Delta R}{T})$
- → cooling: $T(t+1) = aT(t)$



maximum reachability path:

-> generate initial probabilities realization: x

-> slight perturb: add ε to each probability

-> convert reliability to shortest-path cost: $G(x) = -\log(R(x))$

-> flip a random edge: x'

-> acceptance: $\alpha(x', x) = \exp\left(-\frac{\Delta G}{T}\right)$

-> cooling: $T(t+1) = a \cdot T(t)$