Advanced Sampling Algorithms

- Mobashir Mohammad
- Hirak Sarkar
- Parvathy Sudhir
- Yamilet Serrano Llerena
- Aditya Kulkarni
- Tobias Bertelsen
- Nirandika Wanigasekara
- Malay Singh
Introduction

Mobashir Mohammad
Sampling Algorithms

- **Monte Carlo Algorithms**
  - Gives an approximate answer with a certain probability
  - May terminate with a failure signal

- **Las Vegas Algorithms**
  - Always gives an exact answer
  - Doesn’t terminate in a definite time
What are the chances of winning with a properly shuffled deck?

Hard to compute because winning or losing depends on a complex procedure of reorganizing the cards.

Why not play just a few hands, and see empirically how many do in fact we win?

More generally, we can approximate a probability density function using only a few samples from that density.
Simplest Monte Carlo Method

- Finding the value of $\pi$ — “Shooting Darts”

- $\pi/4$ is equal to the area of a circle of diameter 1

- Randomly select a large number of points inside the square

$$\int_0^1 \int_0^1 \text{if } x^2 + y^2 < 1 \text{ then } 1 \text{ else } 0 \, dx \, dy$$

Integral solved with Monte Carlo

$$\frac{1}{4} \cdot \frac{N_{\text{inside\_circle}}}{N_{\text{total}}} \quad \text{(Exact when } N_{\text{total}} \rightarrow \infty)$$
Monte Carlo Principle

- Given a very large set $X$ and a distribution $p(x)$ over it
- We draw a set of $N$ independent and identically distributed samples
- We can then approximate the distribution using these samples.

$$p_N(x) = \frac{1}{n} \sum_{i=1}^{N} 1_{\{x_i = x\}} \rightarrow_{N \to \infty} p(x)$$

- We can also use these samples to compute expectations, …
Problem:
- Exact inference from a probability distribution is hard
- Determine the number of people likely to go shopping to a Nike Store at Vivo City on a Sunday?

Solution:
- Use random samples to get an approximate inference.

How:
- Use a Markov Chain to get random samples!
Markov Chains

Hirak Sarkar
Introduction to Markov Chain

- Real world systems contain uncertainty and evolve over time
- Stochastic Processes and Markov Chains model such systems
- A discrete time stochastic process is a sequence of random variables $X_0, X_1, \ldots$ and typically denoted by $\{X_n\}$
Components of a Stochastic Process

- The *state space* of a stochastic process is the set of all values that the \( \{X_n\} \) can take.

- Time: \( n = 1, 2, \ldots \)

- Set of states is denoted by \( S \).

\[
S = \{s_1, s_2, \ldots, s_k\} \text{ here there are } k \text{ states}
\]

Clearly \( \{X_n\} \) will take one of \( k \) values
Markov Property and Markov Chain

- Consider special class of stochastic processes that satisfy a certain property called *Markov Property*.

- **Markov Property**: State for $X_n$ only depends on content of $X_{n-1}$ not any of $X_i$ for $i < n - 1$

- Formally,
  $$P(X_n = i_n | X_0 = i_0, ..., X_{n-1} = i_{n-1}) = P(X_n = i_n | X_{n-1} = i_{n-1}) = p(i_n, i_{n-1})$$

- We call such processes as *Markov Chain*

- A matrix containing $p(i_n, i_{n-1})$ in $(i_n, i_{n-1})$ cell is called Transition Matrix.
The one-step transition matrix for a Markov chain with states $S = \{0,1,2\}$ is

$$
\begin{bmatrix}
p_{00} & p_{01} & p_{02} \\
p_{10} & p_{11} & p_{12} \\
p_{20} & p_{21} & p_{22}
\end{bmatrix}
$$

Where $p_{ij} = \Pr(X_1 = j|X_0 = i)$

If $P$ is same for any point of time then it is called time homogeneous.

$$
\sum_{j=1}^{m} p_{i,j} = 1 \quad p_{i,j} \geq 0 \ \forall i, j
$$
Example: Random Walk

**Adjacency Matrix**

\[
\begin{bmatrix}
0 & 1 & 0 \\
0 & 0 & 1 \\
1 & 1 & 0 \\
\end{bmatrix}
\]

**Transition Matrix**

\[
\begin{bmatrix}
0 & 1 & 0 \\
0 & 0 & 1 \\
1/2 & 1/2 & 0 \\
\end{bmatrix}
\]
What is a random walk

t=0

A

B

1

1/2

1/2

1

C
What is a random walk
What is a random walk

$t=0$

$t=1$

$t=2$
What is a random walk

\[ \text{t} = 0 \]

\[ \text{t} = 1 \]

\[ \text{t} = 2 \]

\[ \text{t} = 3 \]
Probability Distributions

- $x_t(i) = P(X_t = i) =$ probability that the surfer is at node $i$ at time $t$

- $x_{t+1}(i) = \sum_j P(X_{t+1} = i \mid X_t = j) P(X_t = j) = \sum_j p_{i,j} x_t(j)$

- $x_{t+1} = x_t P = x_{t-1} P P = x_{t-2} P P P = \ldots = x_0 P^t$

- What happens when the surfer keeps walking for a long time?
Stationary Distribution

- When the surfer keeps walking for a long time

- When the distribution does not change anymore
  - i.e. $x_{T+1} = x_T$

- We denote it as $\pi$
Basic Limit Theorem

\[ P = \begin{pmatrix} 0 & 1 & 0 \\ \frac{1}{3} & 0 & \frac{2}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{pmatrix} \quad P^5 = \begin{pmatrix} 0.246914 & 0.407407 & 0.345679 \\ 0.251029 & 0.36214 & 0.386831 \\ 0.251029 & 0.366255 & 0.382716 \end{pmatrix} \]

\[ P^{10} = \begin{pmatrix} 0.250013 & 0.37474 & 0.375248 \\ 0.249996 & 0.375095 & 0.374909 \\ 0.249996 & 0.375078 & 0.382926 \end{pmatrix} \]

\[ P^{20} = \begin{pmatrix} 0.25000000002 & 0.3749999913 & 0.3750000085 \\ 0.2499999999 & 0.3750000003 & 0.374999997 \\ 0.24999999999 & 0.3750000028 & 0.3749999973 \end{pmatrix} \]
## Example

\[
\begin{align*}
\pi^{(0)} &= (0.5, 0.5) \\
\pi^{(1)} &= (0.35, 0.65) \\
\pi^{(2)} &= (0.365, 0.635) \\
\pi^{(3)} &= (0.3635, 0.6365) \\
\pi^{(4)} &= (0.3636, 0.6363) \\
\pi^{(0)} &= (1.0) \\
\pi^{(1)} &= (0.3, 0.7) \\
\pi^{(2)} &= (0.37, 0.63) \\
\pi^{(3)} &= (0.363, 0.637) \\
\pi^{(4)} &= (0.3637, 0.6363)
\end{align*}
\]

\[
\begin{align*}
\pi^{(0)} &= (0.1, 0.9) \\
\pi^{(1)} &= (0.39, 0.61) \\
\pi^{(2)} &= (0.361, 0.639) \\
\pi^{(3)} &= (0.3639, 0.6361) \\
\pi^{(4)} &= (0.3636, 0.6364) \\
\pi^{(0)} &= (0.9, 0.1) \\
\pi^{(1)} &= (0.31, 0.69) \\
\pi^{(2)} &= (0.369, 0.631) \\
\pi^{(3)} &= (0.3631, 0.6369) \\
\pi^{(4)} &= (0.3637, 0.6363)
\end{align*}
\]
Classification of States

- State $j$ is **reachable** from state $i$ if the probability to go from $i$ to $j$ in $n > 0$ steps is greater than zero (State $j$ is reachable from state $i$ if in the state transition diagram there is a path from $i$ to $j$).

- A subset $S'$ of the state space $S$ is **closed** if $p_{ij} = 0$ for every $i \in S'$ and $j \notin S$.

- A state $i$ is said to be **absorbing** if it is a single element closed set.

- A closed set $S$ of states is **irreducible** if any state $j \in S$ is reachable from every state $i \in S$.

- A Markov chain is said to be **irreducible** if the state space $S$ is irreducible.
Example

- **Irreducible Markov Chain**

  ![Irreducible Markov Chain Diagram]

- **Reducible Markov Chain**

  ![Reducible Markov Chain Diagram]

  Absorbing State

  Closed irreducible set
Periodic and Aperiodic States

- Suppose that the structure of the Markov Chain is such that state \( i \) is visited after a number of steps that is an integer multiple of an integer \( d > 1 \). Then the state is called **periodic** with period \( d \).

- If no such integer exists (i.e., \( d = 1 \)) then the state is called **aperiodic**.

- **Example**

  Periodic State \( d = 2 \)

```
P = \[
\begin{bmatrix}
0 & 1 & 0 \\
0.5 & 0 & 0.5 \\
0 & 1 & 0 \\
\end{bmatrix}
```

1 - 0.5

0.5 - 1
Stationary Distribution

\[ \begin{align*}
\pi^0 &= \{\pi_0^0, \pi_1^0\} = (0.5, 0.5) \\
\pi^1 &= \pi^0 P = (0.5, 0.5) \begin{bmatrix} 0.3 & 0.7 \\ 0.4 & 0.6 \end{bmatrix} = (0.36, 0.65)
\end{align*} \]
Steady State Analysis

- Recall that the probability of finding the MC at state $i$ after the $k$th step is given by
  \[
  \pi_i(k) \equiv \Pr \{ X_k = i \} \quad \pi(k) = [\pi_0(k), \pi_1(k), \ldots]
  \]

- An interesting question is what happens in the “long run”, i.e.,
  \[
  \pi_i \equiv \lim_{k \to \infty} \pi_i(k)
  \]

- This is referred to as steady state or equilibrium or stationary state probability

Questions:
- Do these limits exists?
- If they exist, do they converge to a legitimate probability distribution, i.e.,
  \[
  \sum \pi_i = 1
  \]
- How do we evaluate $\pi_j$, for all $j$. 
Stealth State Analysis

- **THEOREM**: In a finite irreducible aperiodic Markov chain a unique stationary state probability vector \( \pi \) exists such that \( \pi_j > 0 \) and

\[
\lim_{k \to \infty} \pi_j(k) = \pi_j
\]

- The steady state distribution \( \pi \) is determined by solving

\[
\pi = \pi P \quad \text{and} \quad \sum_i \pi_i = 1
\]
Sampling from a Markov Chain

Parvathy
Sampling from a Markov Chain

- **Sampling** – create random samples from the desired probability distribution (P)

- Construct Markov Chain with steady state distribution $\pi$
  - $\pi = P$

- Run the Markov chain long enough

- On convergence, its states are approximately distributed according to $\pi$.

- Random sampling of Markov chain yields random samples of P

- But!
  - Difficult to determine how long to run the Markov chain.
Ising Model

- Named after physicist Ernst Ising, invented by Wilhem Lenz
- Mathematical model of ferromagnetism
- Consists of discrete variables representing magnetic dipole moments of atomic spins (+1 or −1)
1D Ising Model

- Solved by Ernst Ising
- N spins arranged in a ring
- $P(s) = \frac{e^{-\beta H(s)}}{Z}$
  - $\beta$ – inverse temperature
  - $H(s)$ – energy
  - $s = \{x_1, \ldots, x_N\}$ is one possible configuration
- Higher probability on states with lower energy
2D Ising Model

- Spins are arranged in a lattice.
  - Each spin interacts with 4 neighbors

- Simplest statistical models to show a phase transition
2D Ising Model (2)

- $G = (V, E)$
- $x_i \in \{+1, -1\}$ - value of spin at location $i$
- $S = \{s_1, s_2, ..., s_n\}$ is the state space.
- $H(s) = - \sum_{(x_i, x_j) \in E} s(x_i)s(x_j)$
  - Similar spins subtract energy
  - Opposite spins add energy
2D Ising Model (3)

\[ P(s) = \frac{e^{-\beta H(s)}}{Z} = \frac{e^{-\beta \sum_{(x_i,x_j) \in E} s(x_i)s(x_j)}}{Z} \]

- If \( \beta = 0 \) all spin configurations have same probability
- If \( \beta > 0 \) lower energy preferred
Exact Inference is Hard

- Posterior distribution over $s$
  - If $Y$ is any observed variable
  - $P(s|Y) = \frac{p(s, y)}{\sum p(s, y)}$

- Intractable computation
  - Joint probability distribution - $2^N$ possible combinations of $s$
  - Marginal probability distribution at a site
  - MAP estimation
The Big Question

- Given $\pi$ on $S = \{s_1, s_2, ..., s_n\}$ simulate a random object with distribution $\pi$
Methods

- Generate random samples to estimate a quantity
- Samples are generated “Markov-chain style”
  - Markov Chain Monte Carlo (MCMC)
  - Propp Wilson Simulation
    - Las Vegas variant
  - Sandwiching
    - Improvement on Propp Wilson
MARKOV CHAIN
MONTE CARLO METHOD
(MCMC)

YAMILLET R. SERRANO LL.
Given a probability distribution $\pi$ on $S = \{s_1, \ldots, s_k\}$, how do we simulate a random object with distribution $\pi$?
Intuition

Given a probability distribution $\pi$ on $S = \{s_1, \ldots, s_k\}$, how do we simulate a random object with distribution $\pi$?
1. Construct an irreducible and aperiodic Markov Chain \([X_0, X_1, \ldots]\), whose stationary distribution \(\pi\).

2. If we run the chain with arbitrary initial distribution then
   1. The Markov Chain Convergence Theorem guarantees that the distribution of the chain at time \(n\) converges to \(\pi\).

3. Hence, if we run the chain for a sufficiently long time \(n\), then the distribution of \(X_n\), will be very close to \(\pi\). So it can be used as a sample.
Generally, two types of MCMC algorithm
- Metropolis Hasting
- Gibbs Sampling
Metropolis Hastings Algorithm

- Original method:
  - Metropolis, Rosenbluth, Rosenbluth, Teller and Teller (1953).

- Generalized by Hasting in 1970.

- Rediscovered by Tanner and Wong (1987) and Gelfang and Smith (1990)

- Is one way to implement MCMC.
Metropolis Hastings Algorithm

Basic Idea

**GIVEN:** A probability distribution \( \pi \) on \( S = \{s_1, \ldots, s_k\} \)

**GOAL:** Approx. sample from \( \pi \)

Start with a **proposal distribution** \( Q(x,y) \)
- \( Q(x,y) \) specifies transition of Markov Chain
- \( Q(x,y) \) plays the role of the transition matrix

By accepting/rejecting the proposal, MH simulates a Markov Chain, whose stationary distribution is \( \pi \)
Metropolis Hastings Algorithm

**Algorithm**

**GIVEN:** A probability distribution $\pi$ on $S = \{s_1, \ldots, s_k\}$

**GOAL:** Approx. sample from $\pi$

Given current sample, $x^{(t)}$:

- Draw $y$ from the proposal distribution, $Q(x^{(t)}, y)$
- Draw $U \sim (0,1)$ and update

$$x^{(t+1)} = \begin{cases} y & \text{if } U \leq r(x^{(t)}, y) \\ x^{(t)} & \text{otherwise} \end{cases}$$

where the acceptance probability is

$$r(x, y) = \min\left\{1, \frac{\pi(y)Q(y, x)}{\pi(x)Q(x, y)}\right\}$$
Consider $m$ sites around a circle.

Each site $i$ can have one of two spins

$$X_i \in \{-1, 1\}$$

The target distribution:

$$\pi(x) \propto e^{\beta \sum_i x_i x_{i+1}}$$
Metropolis Hastings Algorithm
The Ising Model

Target Distribution:
\[ \pi(x) \propto \exp(\beta \sum_j x_j x_{j+1}) \]

Proposal Distribution:
1. Randomly pick one out of the \( m \) spins
2. Flip its sign

Acceptance probability (say \( i \)-th spin flipped):
\[ r = \min \left\{ 1, \frac{\exp \left( \beta \sum_{|j-i|>2} x_j x_{j+1} - \beta (x_i x_{i+1} + x_{i-1} x_i) \right)^{1/m}} {\exp \left( \beta \sum_{|j-i|>2} x_j x_{j+1} + \beta (x_i x_{i+1} + x_{i-1} x_i) \right)^{1/m}} \right\} \]

\[ r = \min \left\{ 1, \exp \left( -2\beta (x_i x_{i+1} + x_{i-1} x_i) \right) \right\} \]
Metropolis Hastings Algorithm
The Ising Model

Figure 3: Example states of an Ising system on a $100 \times 100$ grid. Black squares represent spin-up and white squares represent spin-down. The low-temperature ground states are not shown, since they consist of the grid entirely spin-up or entirely spin-down.

Image has been taken from Montecarlo investigation of the Ising Model(2006) – Tobin Fricke
Disadvantages of MCMC

- No matter how large \( n \) is taken to be in the MCMC algorithm, there will still be some discrepancy between the distribution of the output and the target distribution \( \pi \).

- In order to make the previous error small, we need to figure out how large \( n \) needs to be.
Bounds on Convergence of MCMC

Aditya Kulkarni
Seminar on probabilities (Strasunbourge - 1983)

- If it is difficult to obtain asymptotic bounds on the convergence time of MCMC algorithms for Ising models, use quantitative bounds

- Use a characteristics of Ising model MCMC algorithm to obtain quantitative bounds

David Aldous
What we need to ensure

- As we go along the time

We approach to the stationary distribution in a monotonically decreasing fashion.

\[ d(t) \to 0 \text{ as } t \to \infty \]

- When we stop

The sample should follow a distribution which is not further apart from a stationary distribution by some factor \( \varepsilon \).

\[ d(t) \leq \varepsilon \]
Total Variation Distance $|| \cdot ||_{TV}$

- Given two distributions $p$ and $q$ over a finite set of states $S$,
- The total variation distance $|| \cdot ||_{TV}$ is

$$||p - q||_{TV} = \frac{1}{2} ||p - q||_1 = \frac{1}{2} \sum_{x \in S} |p(x) - q(x)|$$
Convergence Time

- Let $X = [X_0, X_1, ...]$ be a Markov Chain of a state space with stationary distribution $\pi$

- We define $d(t)$ as the worst total variation distance at time $t$.

  $$d(t) = \max_{x \in V} \| P(X^t | X^0 = x) - \pi \|_{TV}$$

- The mixing time $t$ is the minimum time $t$ such that $d(t)$ is at most $\varepsilon$.

  $$\tau(\varepsilon) = \min \{ t : d(t) \leq \varepsilon \}$$

- Define

  $$\tau = \tau(\frac{1}{2\epsilon}) = \min \{ t : d(t) \leq \frac{1}{2\epsilon} \}$$
Quantitative results

\[ d(t) \]

\[ \tau \]

\[ \tau(\varepsilon) \]

\[ 1/2e \]

\[ \varepsilon \]
Consider two random walks started from state $i$ and state $j$

Define $\rho(t)$ as the worst total variation distance between their respective probability distributions at time $t$

a. $\rho(t) \leq 2d(t)$

b. $d(t)$ is decreasing
Upper bound on \( d(t) \) proof

From part a and the definition of \( \tau = \tau(1/2e) = \min \{ t : d(t) \leq 1/2e \} \), we get \( \rho(\tau) \leq e^{-1} \).

Also from part b, we deduce that upper bound of \( \rho \) at a particular time \( t_0 \) gives upper bound for later times

\[
\rho(t) \leq \left( \rho(t_0) \right)^n ; \ n t_0 \leq t \leq (n + 1) t_0
\]

\[
\rho(t) \leq \left( \rho(t_0) \right)^{\left( \frac{t}{t_0} - 1 \right)}
\]

Substitute \( \tau \) instead of \( t_0 \)

\[
\rho(t) \leq \left( \rho(\tau) \right)^{\left( \frac{t}{\tau} - 1 \right)}
\]

\[
d(t) \leq \exp(1 - \frac{t}{\tau}), \ t \geq 0
\]
Upper bound on $\tau(\varepsilon)$ proof

- Algebraic calculations:

\[
d(t) \leq \exp\left(1 - \frac{t}{\tau}\right)
\]
\[
\log d(t) \leq 1 - \frac{t}{\tau}
\]
\[
\frac{t}{\tau} \leq 1 - \log d(t) \leq 1 + \log\left(\frac{1}{d(t)}\right)
\]
\[
t \leq \tau \times (1 + \log\left(\frac{1}{\varepsilon}\right))
\]
\[
\tau(\varepsilon) \leq 2et \times (1 + \log\left(\frac{1}{\varepsilon}\right))
\]
Upper bounds

\[ d(t) \leq \min(1, \exp(1 - \frac{t}{\tau})), \quad t \geq 0 \]

\[ \tau(\varepsilon) \leq \tau \times (1 + \log(\frac{1}{\varepsilon})), \quad 0 < \varepsilon < 1 \]

\[ \tau(\varepsilon) \leq 2et \times (1 + \log(\frac{1}{\varepsilon})), \quad 0 < \varepsilon < 1 \]
Entropy of initial distribution

Measure of randomness:

\[ \text{ent}(\mu) = -\sum_{x \in V} \mu(x) \log \mu(x) \]

- $x$ is initial state
- $\mu$ is initial probability distribution
Few more lemma’s

1. Let \((X_t)\) is a random walk associated with \(\mu\), and \(\mu_t\) be the distribution of \(X_t\) then

\[
\text{ent}(\mu_t) \leq t \cdot \text{ent}(\mu)
\]

2. If \(v\) is a distribution of \(S\) such that \(||v - \pi|| \leq \varepsilon\) then

\[
\text{ent}(v) \geq (1 - \varepsilon) \log |S|
\]
Lower bound on $d(t)$ proof

- From lemma 2,

$$ent(v) \geq (1 - \varepsilon) \log |S|$$

$$\frac{ent(v)}{\log |S|} \geq (1 - \varepsilon) \geq (1 - d(t))$$

$$d(t) \geq 1 - \frac{ent(v)}{\log |S|}$$

$$d(t) \geq 1 - \frac{t \cdot ent(\mu)}{\log |S|}$$
Lower bound on $\tau(\varepsilon)$ proof

- From lemma 1,

\[ \text{ent}(\mu_t) \leq t \text{ent}(\mu) \]

\[ t \geq \frac{\text{ent}(\mu_t)}{\text{ent}(\mu)} \]

- From lemma 2,

\[ \text{ent}(\mu_t) \geq (1 - \varepsilon) \log |S| \]

\[ t \geq \frac{(1 - \varepsilon) \log |S|}{\text{ent}(\mu)} \]

\[ \tau(\varepsilon) \geq \frac{(1 - \varepsilon) \log |S|}{\text{ent}(\mu)} \]
Lower bounds

\[ d(t) \geq 1 - \frac{t \cdot \text{ent}(\mu)}{\log|S|} \]

\[ \tau(\varepsilon) \geq \frac{(1 - \varepsilon) \log |S|}{\text{ent}(\mu)} \]
An exact version of MCMC

Problems with MCMC

A. Have accuracy error, which depends on starting state
B. We must know number of iterations

James Propp and David Wilson propos Coupling from the past (1996)

A.k.a. the Propp-Wilson algorithm

Idea:

Solve problems by running chain infinitely
An exact version of MCMC

**Theoretical**
- Runs all configurations infinitely
- Literarily takes $\infty$ time
- Impossible

**Coupling from the past**
- Runs all configurations for finite time
- Might take 1000’s of years
- Infeasible

**Sandwiching**
- Run few configurations for finite time
- Takes seconds
- Practicable
Theoretical exact sampling

- Recall the convergence theorem:
  - We will approach the stationary distribution as the number of steps goes to infinity

- Intuitive approach:
  - To sample perfectly we start a chain and run for infinity
  - Start at $t = 0$, sample at $t = \infty$
  - Problem: We never get a sample

- Alternative approach:
  - To sample perfectly we take a chain that have already been running for an infinite amount of time
  - Start at $t = -\infty$, sample at $t = 0$
Theoretical independence of starting state

- Sample from a Markov chain in MCMC depends solely on
  - Starting state $X_{-\infty}$
  - Sequence of random numbers $U$

- We want to be independent of the starting state.
  - For a given sequence of random numbers $[U_{-\infty}, \ldots, U_{-1}]$ we want to ensure that the starting state $X_{-\infty}$ has no effect on $X_0$
Theoretical independence of starting state

Collisions:

- For a given $[U_{-\infty}, ..., U_{-1}]$ if two Markov chains is at the same state at some $t'$ the will continue on together:
Coupling from the past

- At some finite past time $t = -N$
  - All past chains have already run infinitely and have coupled into one $\infty - N = \infty$
- We want to continue that coupled chain to $t = 0$
- But we don’t know which state they will be at $t = -N$

- Run all states from $-N$ instead of $-\infty$
Coupling from the past

1. Let \( U = [U_{-1}, U_{-2}, U_{-3}, ...] \) be the sequence of independent uniformly random numbers

2. For \( N_j \in [1,2,4,8, ...] \)
   1. Extend \( U \) to length \( n_j \), keeping \( U_{-1}, ..., U_{n_{j-1}} \) the same
   2. Start one chain from each state at \( t = -N_j \)
   3. For \( t \) from \( -N_j \) to zero: Simulate the chains using \( U_t \)
   4. If all chains has converged at \( S_i \) at \( t = 0 \), return \( S_i \)
   5. Else repeat loop
Coupling from the past

\[ n = \begin{array}{cccccc}
  -4 & -3 & -2 & -1 & 0 \\
 S_1 & S_1 & S_1 & S_1 & S_1 \\
 S_2 & S_2 & S_2 & S_2 & S_2 \\
 S_3 & S_3 & S_3 & S_3 & S_3 \\
 S_4 & S_4 & S_4 & S_4 & S_4 \\
 U_{-4} & U_{-3} & U_{-2} & U_{-1} & \\
\end{array} \]
Questions

Why do we double the lengths?

- Worst case $< 4N_{opt}$ steps, where $N_{opt}$ is the minimal $N$ at which we can achieve convergence
- Compare to $N \in [1,2,3,4,5, ...], O(N_{opt}^2)$ steps

Why do we have to use the same random numbers?

- Different samples might take longer or shorter to converge.
- We must evaluate the same sample in each iteration.
- The sample should only be dependent on $U$ not the different $N$
Using the same $U$

- We have $k$ states with the following update function

\[
(S_i, U) = \begin{cases} 
S_1 & \text{if } U < \frac{1}{2} \\
S_{i+1} & \text{otherwise}
\end{cases}
\]

\[
\phi(S_k, U) = \begin{cases} 
S_1 & \text{if } U < \frac{1}{2} \\
S_k & \text{otherwise}
\end{cases}
\]

- $\pi(S_i) = \frac{1}{2^i}, \pi(k) = \frac{2}{2^k}$
Using the same $U$

- The probability of $S_1$ only depends on the last random number:

\[
P(S_1) = P\left(U_{-1} < \frac{1}{2}\right) = \frac{1}{2}
\]

- Let's assume we generate a new $U$ for each run: $U^1, U^2, U^3, ...$

<table>
<thead>
<tr>
<th>$n$</th>
<th>$U$</th>
<th>$P(S_1)$</th>
<th>$P(S_1)$ acc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$[U^1_{-1}]$</td>
<td>$P\left(U^1_{-1} &lt; \frac{1}{2}\right)$</td>
<td>50 %</td>
</tr>
<tr>
<td>2</td>
<td>$[U^2_{-2}, U^2_{-1}]$</td>
<td>$P\left(U^2_{-1} &lt; \frac{1}{2} \lor U^1_{-1} &lt; \frac{1}{2}\right)$</td>
<td>75 %</td>
</tr>
<tr>
<td>4</td>
<td>$[U^3_{-4}, ..., U^3_{-1}]$</td>
<td>$P\left(U^3_{-1} &lt; \frac{1}{2} \lor U^2_{-1} &lt; \frac{1}{2} \lor U^1_{-1} &lt; \frac{1}{2}\right)$</td>
<td>81.25 %</td>
</tr>
<tr>
<td>8</td>
<td>$[U^4_{-8}, ..., U^4_{-1}]$</td>
<td>$P\left(U^4_{-1} &lt; \frac{1}{2} \lor U^3_{-1} &lt; \frac{1}{2} \lor U^2_{-1} &lt; \frac{1}{2} \lor U^1_{-1} &lt; \frac{1}{2}\right)$</td>
<td>81.64 %</td>
</tr>
</tbody>
</table>
The probability of $S_1$ only depends on the last random number:

$$P(S_1) = P\left(U_{-1} < \frac{1}{2}\right) = \frac{1}{2}$$

Let's instead use the same $U$ for each run: $U^1$

<table>
<thead>
<tr>
<th>$n$</th>
<th>$U$</th>
<th>$P(S_1)$</th>
<th>$P(S_1)$ acc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$[U_{-1}^1]$</td>
<td>$P\left(U_{-1}^1 &lt; \frac{1}{2}\right)$</td>
<td>50 %</td>
</tr>
<tr>
<td>2</td>
<td>$[U_{-2}^1, U_{-1}^1]$</td>
<td>$P\left(U_{-1}^1 &lt; \frac{1}{2}\right)$</td>
<td>50 %</td>
</tr>
<tr>
<td>4</td>
<td>$[U_{-4}^1, ..., U_{-1}^1]$</td>
<td>$P\left(U_{-1}^1 &lt; \frac{1}{2}\right)$</td>
<td>50 %</td>
</tr>
<tr>
<td>8</td>
<td>$[U_{-8}^1, ..., U_{-1}^1]$</td>
<td>$P\left(U_{-1}^1 &lt; \frac{1}{2}\right)$</td>
<td>50 %</td>
</tr>
</tbody>
</table>
Problem

- In each step we update up to $k$ chains
- Total execution time is $O(N_{opt}k)$
- BUT $k = 2^{(L^2)}$ in the Ising model
- Worse than the naïve approach $O(k)$
Sandwiching

Nirandika Wanigasekara
Sandwiching

Propp Wilson algorithm

- Many vertices $\rightarrow$ running $k$ chains will take time
- Impractical for large $k$

Choose a relatively small state space

- Can we still get the same results?
- Try sandwiching
Sandwiching

- Idea
  - Find two chains bounding all other chains
  - If we have such two boundary chains
  - Check if those two chains converge
  - Then all other chains have also converged
Sandwiching

- To come up with the boundary chains
  - Need a way to order the states
    - $S_1 \leq S_2 \leq S_3 \ldots$
  - Chain in higher state does not cross a chain in lower state

- Results in a Markov Chain obeying certain monotonicity properties
Sandwiching

- Let's consider
  - A fixed set of states \( \rightarrow k \)
    - State space \( S = \{1, \ldots, k\} \)
  - A transition matrix
    - \( P_{11} = P_{12} = \frac{1}{2} \)
    - \( P_{kk} = P_{k,k-1} = \frac{1}{2} \)
    - For \( i=2, \ldots, k-1 \), \( P_{i,i-1} = P_{i,i+1} = \frac{1}{2} \)
    - All the other entries are 0

\[
\begin{pmatrix}
0.5 & 0.5 & 0 & 0 & 0 \\
0.5 & 0 & 0.5 & 0 & 0 \\
0 & 0.5 & 0 & 0.5 & 0 \\
0 & 0 & 0.5 & 0 & 0.5 \\
0 & 0 & 0 & 0.5 & 0.5 \\
\end{pmatrix}
\]
Sandwiching

- What is this Markov Chain doing?
  - Take one step up and one step down the ladder, at each integer time, with probability $\frac{1}{2}$
  - If at the top or the bottom (state $k$)
    - It will stay where it is
  - **Ladder Walk on $k$ vertices**

- The stationary distribution $\pi$ of this Markov Chain
  - $\pi_i = \frac{1}{k}$ for $i = 1, \ldots, k$
Sandwiching

- Propp-Wilson for this Markov Chain with
  - Valid update function $\phi$
    - if $u < \frac{1}{2}$ then step down
    - if $u \geq \frac{1}{2}$ then step up
  - negative starting times
    - $(N_1, N_2, \ldots) = (1, 2, 4, 8, \ldots)$
  - States $k = 5$

$$
\begin{align*}
\phi(1, u) &= \begin{cases} 
\frac{1}{2} & \text{for } u \in [0, \frac{1}{2}] \\
2 & \text{for } u \in [\frac{1}{2}, 1]
\end{cases} \\
\phi(k, u) &= \begin{cases} 
k^{-1} & \text{for } u \in [0, \frac{1}{2}] \\
k & \text{for } u \in [\frac{1}{2}, 1]
\end{cases}
\end{align*}
$$

and for $i = 2, \ldots, k - 1$

$$
\phi(i, u) = \begin{cases} 
i^{-1} & \text{for } u \in [0, \frac{1}{2}] \\
i+1 & \text{for } u \in [\frac{1}{2}, 1]
\end{cases}
$$
Sandwiching
Sandwiching

- Update function preserves ordering between states
  - for all $U \in [0, 1]$ and all $i, j \in \{1, \ldots, k\}$ such that $i \leq j$ we have $\phi(i, U) \leq \phi(j, U)$

- It is sufficient to run only 2 chains rather than k
Sandwiching

- Are these conditions always met
  - No, not always
  - But, there are frequent instances where these conditions are met
  - Especially useful when $k$ is large
  - Ising model is a good example for this
Ising Model

Malay Singh
2D Ising Model

- A grid with sites numbered from 1 to $L^2$ where $L$ is grid size.
- Each site $i$ can have spin $x_i \in \{-1, +1\}$
- $V$ is set of sites $S = \{-1,1\}^V$ defines all possible configurations (states)
- Magnetisation $m$ for a state $s$ is
  $$ m(s) = \frac{\sum_i s(x_i)}{L^2} $$
- The energy of a state $s$
  $$ H(s) = -\sum_{ij} s(x_i)s(x_j) $$
Ordering of states

- For two states \( s, s' \) we say \( s \leq s' \) if \( s(x) < s'(x) \) \( \forall x \in V \)

Minima \( m = -1 \)

Maxima \( m = 1 \)

\[
\begin{align*}
  s^{\min}(x) &= -1 \ \forall x \in V \\
  s^{\max}(x) &= +1 \ \forall x \in V
\end{align*}
\]

Hence \( s^{\min} \leq s \leq s^{\max} \) for all \( s \)
The update function

We use the sequence of random numbers \([U_n, U_{n-1}, \ldots, U_0]\)

We are updating the state at \(X_n\). We choose a site \(x\) in \([0, L^2]\) uniformly.

\[
X_{n+1}(x) = \begin{cases} 
+1, & \text{if } U_{n+1} < \frac{\exp \left(2\beta (K_+(x, s) - K_-(x, s)) \right)}{\exp \left(2\beta (K_+(x, s) - K_-(x, s)) \right) + 1} \\
-1, & \text{otherwise}
\end{cases}
\]
Maintaining ordering

- **Ordering after update** (From $X_n$ to $X_{n+1}$)
  - We choose the same site $x$ to update in both chains
  - The spin of $x$ at $X_{n+1}$ depends on update function

- **We want to check**
  \[
  \frac{\exp \left( 2\beta (K_+ (x, s) - K_- (x, s)) \right)}{\exp \left( 2\beta (K_+ (x, s) - K_- (x, s)) \right) + 1} \leq \frac{\exp \left( 2\beta (K_+ (x, s') - K_- (x, s')) \right)}{\exp \left( 2\beta (K_+ (x, s') - K_- (x, s')) \right) + 1}
  \]

- **That is equivalent to checking**
  \[
  K_+ (x, s) - K_- (x, s) \leq K_+ (x, s') - K_- (x, s')
  \]
Maintaining ordering

- As $s \leq s'$ we have
  - $K_+(x, s) \leq K_+(x, s')$
  - $K_-(x, s) \geq K_-(x, s')$

- First equation minus second equation
  - $K_+(x, s) - K_-(x, s) \leq K_+(x, s') - K_-(x, s')$
Ising Model $L = 4$

$T = 3.5$ and $N = 512$

$T = 4.8$ and $N = 128$
Ising Model $L = 8$

$T = 5.9$ and $N = 512$
Ising Model $L = 16$

$T = 5.3$ and $N = 16384$
Summary

- Exact sampling
- Markov chains monte carlo but when we converge
- Propp Wilson with Sandwiching to rescue
Questions?
Example 6.2

Example 6.2 Let \( (X_0, X_1, \ldots) \) be a Markov chain with state space \( S = \{s_1, \ldots, s_k\} \) and transition matrix \( P \), and suppose that the transition matrix has the properties that

(i) \( P_{i,j} > 0 \) whenever \( |i - j| = 1 \), and
(ii) \( P_{i,j} = 0 \) whenever \( |i - j| \geq 2 \).

Such a Markov chain is often called a birth-and-death process, and its transition graph has the form outlined in Figure 5 (with some or all of the \( P_{i,i} \)-“loops” possibly being absent). We claim that any Markov chain of this kind is reversible. To construct a reversible distribution \( \pi \) for the chain, we begin by setting \( \pi_i^+ \) equal to some arbitrary strictly positive number \( a \). The condition (38) with \( i = 1 \) and \( j = 2 \) forces us to take

\[
\pi_2^+ = \frac{aP_{1,2}}{P_{2,1}}.
\]

Applying (38) again, now with \( i = 2 \) and \( j = 3 \), we get

\[
\pi_3^+ = \frac{\pi_2^+P_{2,3}}{P_{3,2}} = \frac{aP_{1,2}P_{2,3}}{P_{2,1}P_{3,2}}.
\]

We can continue in this way, and get

\[
\pi_i^+ = \frac{a \prod_{j=1}^{i-1} P_{j,j+1}}{\prod_{j=1}^{i} P_{j+1,j}}
\]

for each \( i \). Then \( \pi^+ = (\pi_1^+, \ldots, \pi_k^+) \) satisfies the requirements of a reversible distribution, except possibly that the entries do not sum to 1, as is required for any probability distribution. But this is easily taken care of by dividing all entries by their sum. It is readily checked that

\[
\pi = (\pi_1, \pi_2, \ldots, \pi_k) = \left( \frac{\pi_1^+}{\sum_{i=1}^{k} \pi_i^+}, \frac{\pi_2^+}{\sum_{i=1}^{k} \pi_i^+}, \ldots, \frac{\pi_k^+}{\sum_{i=1}^{k} \pi_i^+} \right)
\]

is a reversible distribution.

Having come this far, one might perhaps get the impression that most Markov chains are reversible. This is not really true, however, and to make up for this false impression, let us also consider an example of a Markov chain which is not reversible.
Update function

It remains to construct such a valid update function, but this is no harder than the construction of a valid initiation function: Set, for each $s_i \in S$,

$$
\phi(s_i, x) = \begin{cases} 
  s_1 & \text{for } x \in [0, P_i, 1) \\
  s_2 & \text{for } x \in [P_i, 1, P_i + P_i, 2) \\
  \vdots & \vdots \\
  s_j & \text{for } x \in \left[ \sum_{i=1}^{j-1} P_i, l, \sum_{i=1}^{j} P_i, l \right) \\
  \vdots & \vdots \\
  s_k & \text{for } x \in \left[ \sum_{i=1}^{k-1} P_i, l, 1 \right].
\end{cases} \quad (21)
$$

To see that this is a valid update function, note that for any $s_i, s_j \in S$, we have

$$
\int_0^1 I_{\phi(s_i, x) = s_j} \, dx = \sum_{i=1}^j P_i, l - \sum_{i=1}^{j-1} P_i, l = P_i, l.
$$

Thus, we have a complete recipe for simulating a Markov chain: First construct valid initiation and update functions $\psi$ and $\phi$ (for instance as in (18) and (21)), and then set

$$
X_0 = \psi(U_0) \\
X_1 = \phi(X_0, U_1) \\
X_2 = \phi(X_1, U_2) \\
X_3 = \phi(X_2, U_3)
$$

and so on.

Let us now see how the above works for a simple example.
Additional

- Proof for order of between states?? Need to find