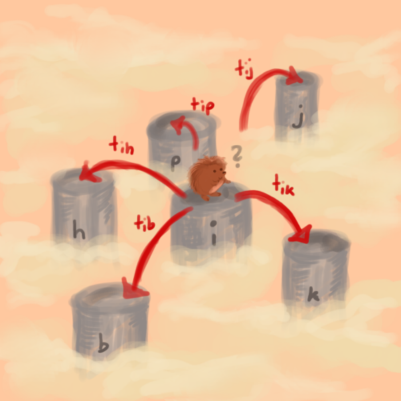
Markov chains and MCMC methods



Outline

1. Basics on Markov chains
2. Live demo: an IRC chat bot
3. Sampling from distributions
4. Evaluating integrals over high-dimensional domains

What is a Markov chain?

- A Markov chain is a system which undergoes transitions from one state to another according to probabilities $P(X' = j \mid X = i) =: T_{ij}$.
- More abstractly, a Markov chain on a state space S is a map $S \to D(S)$, where $D(S)$ is the set of probability measures on S.
- Categorically, a Markov chain is a coalgebra for the functor $D : \mathrm{Set} \to \mathrm{Set}$.

The following systems can be modeled by Markov chains:

- the peg in the game of snakes and ladders
- a random walk
- the weather, if we oversimplify a lot
- randomly surfing on the web

The following cannot:

- the state of a game of blackjack

Basic theory on Markov chains

- The transition matrix is a stochastic matrix:

$$T_{ij} \geq 1, \qquad \sum_j T_{ij} = 1.$$

- If $p \in \mathbb{R}^S$ is a distribution of the initial state, then $p^\mathsf{T} \cdot T^N \in \mathbb{R}^S$ is the distribution of the N'th state.
- If the Markov chain is irreducible and aperiodic, $p^\mathsf{T} \cdot T^N$ approaches a unique limiting distribution p^∞ independent of p as $N \to \infty$.
- A sufficient condition for $p^\infty = q$ is the detailled balance condition

$$q_i T_{ij} = q_j T_{ji}.$$

Live demo: an IRC chat bot

How can we sample from distributions?

Given a density f, want independent samples $x_1, x_2, \ldots$

- If the inverse of the cumulative distribution function F is available:

  1. Sample $u \sim U(0,1)$.
  2. Output $x := F^{-1}(u)$.

How can we sample from distributions?

Given a density f, want independent samples $x_1, x_2, \ldots$

- If the inverse of the cumulative distribution function F is available:
    1. Sample $u \sim U(0, 1)$.
    2. Output $x := F^{-1}(u)$.
- Unfortunately, calculating F^{-1} is expensive in general.

How can we sample from distributions?

Given a density f, want independent samples $x_1, x_2, \ldots$

- If some other sampleable density g with $f \leq Mg$ is available, where $M \geq 1$ is a constant, we can use rejection sampling:

  1. Sample $x \sim g$.
  2. Sample $u \sim U(0, 1)$.
  3. If $u < \frac{1}{M} f(x)/g(x)$, output x; else, retry.

How can we sample from distributions?

Given a density f, want independent samples $x_1, x_2, \ldots$

- If some other sampleable density g with $f \leq Mg$ is available, where $M \geq 1$ is a constant, we can use rejection sampling:

  1. Sample $x \sim g$.
  2. Sample $u \sim U(0, 1)$.
  3. If $u < \frac{1}{M} f(x)/g(x)$, output x; else, retry.

- Works even if f is only known up to a constant factor.
- Acceptance probability is $1/M$, this might be small.

- Proof that the easy sampling algorithm is correct:

$$P(F^{-1}(U) \le x) = P(U \le F(x)) = F(x).$$

- Acceptance probability in rejection sampling:

$$\begin{aligned} P(U < \tfrac{1}{M} f(G)/g(G)) &= E(\tfrac{1}{M} f(G)/g(G)) \\ &= \tfrac{1}{M} \cdot \int f(x)/g(x) \cdot g(x)\, dx = \tfrac{1}{M}. \end{aligned}$$

- Proof of correctness of rejection sampling:

$$\begin{aligned} P(G \le x \,\wedge\, U < \tfrac{1}{M} f(G)/g(G)) &= \int P(G \le x \,\wedge\, U < \tfrac{1}{M} f(G)/g(G) \mid G = t) g(t)\, dt \\ &= \int \mathbf{1}_{t \le x} \cdot P(U < \tfrac{1}{M} f(t)/g(t)) \cdot g(t)\, dt \\ &= \int \mathbf{1}_{t \le x} \cdot \tfrac{1}{M} f(t)/g(t) \cdot g(t)\, dt \\ &= \tfrac{1}{M} F(x), \end{aligned}$$

so $\quad P(G \le x \mid U < \tfrac{1}{M} f(G)/g(G)) = F(x).$

Markov chain Monte Carlo methods

Given a density f, want independent samples $x_1, x_2, \ldots$

1. Construct a Markov chain with limiting density f.
2. Draw samples from the chain.
3. Discard first samples (burn-in period).
4. From the remaining samples, retain only every N'th.

Works very well in practice.

Metropolis–Hastings algorithm

Given a density f, want independent samples $x_1, x_2, \ldots$

Let $g(y, x)$ be such that for any x, $g(\cdot, x)$ is sampleable. Set $B(x, y) := \frac{f(y)g(x,y)}{f(x)g(y,x)}$.

1. Initialize x.
2. Sample $u \sim U(0, 1)$.
3. Sample $y \sim g(\cdot, x)$.
4. If $u < B(x, y)$, set $x := y$; else, keep x unchanged.
5. Output x and go back to step 2.

Works even if f and g are only known up to constant factors.

- The Metropolis algorithm was first published in an 1953 paper *Equation of State Calculations by Fast Computing Machines* by Metropolis, Rosenbluth, Augusta Teller, and Edward Teller. Hastings' addition was in 1970.

- Special case: $g(x, y) = g(y, x)$, then $B(x, y) = f(y)/f(x)$; this is the original Metropolis algorithm.

- Example: $g(\cdot, x) = N(x, \sigma^2)$.

- Set $A(x, y) := \min\{1,\ B(x, y)\}$.

- Transition matrix (really, kernel):

$$T(x, y) = \hat{g}(y, x)A(x, y) + \delta(x, y) \int (1 - A(x, z))\hat{g}(z, x)\, dz.$$

- Balance condition (for $x \neq y$):

$$f(x)T(x, y) = \min\{f(x)\hat{g}(y, x),\ f(y)\hat{g}(x, y)\} = f(y)T(y, x).$$

Evaluating integrals

How can we evaluate integrals

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

where f is a density on a high-dimensional domain?

- $\int_a^b$: standard numerical quadrature
- $\int_{-\infty}^{\infty}$: numerical quadrature after coordinate transform
- $\int_{\mathbb{R}^n}$: iterated numerical quadrature

Evaluating integrals

How can we evaluate integrals

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

where f is a density on a high-dimensional domain?

- $\int_a^b$: standard numerical quadrature
- $\int_{-\infty}^{\infty}$: numerical quadrature after coordinate transform
- $\int_{\mathbb{R}^n}$: iterated numerical quadrature

These techniques sample the domain uniformly and require many evaluations of the integrand.

- Evaluation of such integrals is, of course, important in Bayesian learning and elsewhere.
- Note that adaptive numerical quadrature rules do exist.

The Monte Carlo approach

Draw indep. samples $x_1, \ldots, x_N$ from f and approximate

$$f \approx \frac{1}{N} \sum_{i=1}^{N} \delta_{x_i}, \qquad I := \int a(x)\, f(x)\, dx \approx \frac{1}{N} \sum_{i=1}^{N} f(x_i) =: I_N.$$

- $E(I_N) = I$.
- $\mathrm{Var}(I_N) = \mathrm{Var}_f(a)/N$.
- $I_N \longrightarrow I$ almost surely (strong law of large numbers).

To sample f, use Markov chain techniques; obtain MCMC methods. These made Bayesian ideas useful in practice.