Monte Carlo Methods

Sonia Reilly

October 20, 2023

These notes are based on Jonathan Weare's lecture notes for his Fall 2022 MCMC class.

1 Introduction

- For some probability distribution π and some function f we want to approximate

$$\pi[f] = \int f(x)\pi(x)dx.$$

- A quadrature scheme of order α has error $O(1/N^{\alpha/d})$, e.g. order $O(1/N)$ for the left-hand Riemann integral in 1D, or $O(1/N^2)$ for the trapezoidal approximation. But the dependence on d is the curse of dimensionality – to keep enough points to resolve each dimension, we need to increase N with the power of d.
- A Monte Carlo estimator has error (standard deviation) of order $O(1/\sqrt{N})$, since the standard deviation of $\frac{1}{N}\sum X_i$ is $\sigma/\sqrt{N}$ for X_i's with variance σ^2. Therefore the error decays more slowly in N than quadrature in low dimensions, but faster in high dimensions (in theory). In practice the constant can depend heavily on dimension.
- Because the error decays more slowly, *you should use a Monte Carlo method only if you cannot use a deterministic one.*

2 Exact Sampling and Rejection Sampling

- Inversion
  - Given $U \sim \text{Unif}[0,1]$ and a probability distribution π with cdf $F(y) = P(Y \leq y)$, then $Y = F^{-1}(U) \sim \pi$.
  - Easy for simple distributions like exponential.
- Change of variables
  - If we want samples from π, we can come up with a smooth, invertible $\phi(y)$ such that $\tilde{\pi} = |\det \nabla\phi(y)|\pi(\phi(y))$. Then $X = \phi^{-1}(Y) \sim \pi$.
  - Example: can find a change of variables to convert 2D uniform to 2D Gaussian.
- Rejection sampling
  - If we want samples from π, but we can get samples from $\tilde{\pi}$ with $\pi \leq K\tilde{\pi}$ for some K, then we can do rejection sampling.

- Generate samples from $\tilde{\pi}$ and accept with probability $\frac{\pi(Y)}{K\tilde{\pi}(Y)}$, until one is accepted. Repeat for next sample.
- Can be hard to find a pdf that bounds π.
- If the bound is not tight, rejection sampling can be extremely costly.
- Also considered an exact method, because the samples are $\sim \pi$.

3 Importance Sampling

- If we want to compute $\pi[f]$, the standard Monte Carlo estimator is

$$\overline{f}_N = \frac{1}{N}\sum_{i=1}^{N} f(X_i)$$

for $X_i \sim \pi$. But often we cannot find $X_i \sim \pi$. This estimator may also have high RMSE (high variance).

- The importance sampling estimator

$$\tilde{f}_N = \frac{1}{N}\sum_{i=1}^{N} f(Y_i)\frac{\pi(Y_i)}{\tilde{\pi}(Y_i)}$$

instead uses $Y \sim \tilde{\pi}$ from an easier to sample distribution.

- Requires that $\tilde{\pi}$ be nonzero wherever π is nonzero.
- This is an unbiased estimator.
- The $\tilde{\pi}$ that minimizes the variance is $\tilde{\pi} \propto |f(x)|\pi(x)$, but finding the normalizing constant would require finding $\pi[f]$, so impractical.
- However, we can approximate the normalization constant by

$$\frac{1}{\tilde{Z}_{\tilde{\pi}}} = \frac{1}{N}\sum_{i=1}^{N} \frac{\pi(Y_i)}{\tilde{\pi}(Y_i)},$$

so we can build a new estimator

$$\frac{\tilde{f}_N}{\tilde{1}_N} = \frac{1}{N}\sum_{i=1}^{N} f(Y_i)\tilde{Z}_{\tilde{\pi}}\frac{\pi(Y_i)}{\tilde{\pi}(Y_i)} = \frac{1}{N}\sum_{i=1}^{N} f(Y_i)W_i.$$

It is a biased estimator, but the bias $O(1/N)$ is less than the standard deviation $O(1/\sqrt{N})$, so it typically doesn't matter. It often has lower error than the simple importance sampling estimator, but its main advantage is that we do not need to know the relative normalization constant for $\pi/\tilde{\pi}$.

- Importance sampling in general has the same curse of dimensionality as quadrature – e.g. for Gaussian π, the variance increases exponentially with dimension since the distance between π and a similar Gaussian $\tilde{\pi}$ increases exponentially with dimension.

- Sequential importance sampling
  - Breaks up a high-dimensional sampling problem into pieces when the structure is convenient
  - We can split up π into
$$\pi(x_{1:d}) = \pi(x_1) \prod_{n=2}^{d} \pi(x_n|x_{1:n-1}).$$
So we can sample $X \sim \pi$ by sampling $X_1 \sim \pi(x_1)$, then $X_2 \sim \pi(x_2|x_1)$ and so forth.
  - Idea is to do this by importance sampling. To illustrate by example, suppose we want to sample uniformly from self-avoiding random walks (SAWs) of length d.
  - First, choose a sequence of $\pi_n(x_{1:n})$ such that $\pi = \pi_d$. Need not be marginal probabilities. For example, uniform over SAWs of length n (not the same as uniform over subsets of length n of SAWs of length d).
  - Then choose easy to sample $\tilde{\pi}_n(x_{1:n})$ with conditional probabilities $q(x_n|x_{1:n-1})$. E.g., for SAWs, q could be choosing uniformly from the possible next steps (0 for chains with no possible next steps).
  - Finally, we construct a sequence of importance sampling estimators for $\pi_n(x_{1:n})$ culminating in a $\tilde{f}_N/\tilde{1}_N$ estimator for $\pi[f]$. The weights of the n-th estimator are
$$W_n^{(i)} \propto \frac{\pi_n(Y_{1:n}^{(i)})}{\tilde{\pi}_n(Y_{1:n}^{(i)})} = \frac{\pi_{n-1}(Y_{1:n-1}^{(i)})}{\tilde{\pi}_{n-1}(Y_{1:n-1}^{(i)})} w_n(Y_{1:n}^{(i)}),$$
where
$$w_n(x_{1:n}) = \frac{\pi_n(x_{1:n})}{\pi_{n-1}(x_{1:n-1}) q_n(x_n|x_{1:n-1})}$$
and the normalization constant of the weight is calculated at each step as previously for $\tilde{f}_N/\tilde{1}_N$.

- Sequential importance sampling with resampling
  - Takes advantage of recursive sampling to replace low probability samples at each step with copies of high probability samples. Improves the performance of sequential importance sampling in higher dimensions (since we don't spend most of our time generating very low probability samples by choosing a random point in all dimensions at once).
  - Given a set of samples and weights $\{W_n^{(i)}, X_{1:n}^{(i)}\}_{i=1}^{N_n}$, we resample to get $\{1/N, Y_{1:n}^{(i)}\}_{i=1}^{N_n}$ such that this collection estimates the same integral. Then choose each new $X_{n+1}^{(i)}$ from q, find the new weights using $W_{n+1}^{(i)} \propto w_n(X_{1:n+1}^{(i)})$, and repeat. (Old weights do not appear recursively because they are accounted for in the resampling step.)
  - For example, resampling can be done by choosing the number of copies of each sample from a multinomial distribution with total of N samples and probabilities $W_{1:n}^{(i)}$ for each.

4 Markov Chain Monte Carlo

- We want to find a Markov chain X_i such that
$$\overline{f}_N = \frac{1}{N} \sum_{i=1}^{N} f(X_i)$$

satisfies $\overline{f}_N \to \pi[f]$.

- Ergodicity: Often ergodic = irreducible, but in the context of MCMC we call ergodicity the property that any distribution will limit to the stationary distribution (requires X_i irreducible, aperiodic, visits every point infinitely many times).
- Ergodicity does not strictly imply $\overline{f}_N \to \pi[f]$, but it is almost always required.
- Detailed balance is also not always necessary, but it a good way to ensure π is the stationary distribution.
- Convergence
  - if π is a unique stationary distribution of X_i, can prove that the Monte Carlo estimator using X_i converges to $\pi[f]$ almost surely.
  - Under the additional Lyapunov conditions (essentially that the chain visits any point infinitely many times), we have the Central Limit Theorem,

    $$\lim_{N\to\infty} \sqrt{N}\left(\overline{f}_N - \pi[f]\right) = Z$$

    for $Z \sim N(0, \tau\sigma^2)$, where $\sigma^2 = \mathrm{var}_\pi(X_1)$ and

    $$\tau = 1 + 2\sum_{k=1}^{N} \mathrm{cor}_\pi(f(X_0), f(X_k)).$$

    The τ is called the Integrated Autocorrelation Time (IAT), and it represents how many MCMC samples are needed to achieve the statistical "value" of one independent sample.
- Principle of Partial Resampling: If we update only one component of X_i at each step but preserve the conditional probability of that component given all the other ones, we preserve π. (Can be generalized to updating one component in any coordinate system.)
- Gibbs sampling
  - Simplest form of partial resampling, where at each step a coordinate is chosen from the conditional distribution over that coordinate given the others.
  - Only feasible when conditional distribution is simple, e.g. Gaussian. For example, for Ising model,

    $$\pi(\sigma) \propto e^{\beta \sum_{\vec{i}\leftrightarrow\hat{j}} \sigma_{\hat{i}}\sigma_{\hat{j}}}$$

    with $\sigma_{\hat{i}} \in \{-1, 1\}$ magnetic spins at lattice locations, conditional probability is Bernoulli.
- Metropolis-Hastings
  - Basis of most MCMC methods
  - Generate Y_{t+1} from proposal distribution $q(y|X_t)$
  - With probability

    $$p_{acc}(X_t, Y_{t+1}) = \min\left\{1, \frac{\pi(Y_{t+1})q(X_t|Y_{t+1})}{\pi(X_t)q(Y_{t+1}|X_t)}\right\}$$

    set $X_{t+1} = Y_{t+1}$. Otherwise set $X_{t+1} = X_t$.
  - Want large steps but low rejection rate (rule of thumb: 25% acceptance).

5 Continuous Time MCMC Schemes

- Overdamped Langevin schemes
  - We want methods that work better than Metropolis-Hastings in high dimensions
  - Take MH with $N(x, 2hS(x))$ proposal step, for some covariance $S(x)$. The generator

$$\mathcal{L}_h = \frac{\mathbb{E}[f(X_h^{(1)})|X_h^{(0)} = x] - f(x)}{h}$$

    casts this as a discretization of a continuous time MC with $\Delta t = h$. Taylor expanding $X_h^{(1)}$ around x we find that

$$\mathcal{L}_h = \mathcal{L}_O + O(\sqrt{h}),$$

    where

$$\mathcal{L}_O = \frac{1}{\pi(x)}\text{div}(\pi(x)\nabla f(x)S(x)).$$

    Here div represents the rowwise divergence of a matrix-valued function. We can show that π is stationary for $\mathcal{L}_O$ and that it satisfies detailed balance.
  - The ctMC with generator $\mathcal{L}_O$ (no accept/reject step) is the continuous limit of the Metropolis-Hastings method with accept/reject. In SDE form, it has drift $b(x) = \text{div}(\pi S)/\pi$ and diffusion $\sigma\sigma^T = 2S$. The simplest 1D form is

$$dX_t = \nabla \log \pi(X_t)dt + \sqrt{2}dW_t.$$

  - The Overdamped Langevin scheme discretizes this ctMC (still without accept/reject). It preserves a slightly different invariant measure π_h satisfying $\pi_h[f] - \pi[f] = O(h)$.
  - Letting $X = X_h^{(k)}$ for simplicity, the Euler-Maruyama discretization is

$$X_h^{(k+1)} = X + hS(X)\nabla^T \log(\pi(X)) + h\text{div}S(X) + \sqrt{2hS(X)}\xi^{(k+1)},$$

    where $\xi^{(k+1)} \sim N(0, I)$. The first term is the starting point, the middle two terms are the drift coming from the shift in the direction of higher probability that corresponds to the accept/reject, and the final term is the diffusion corresponding to the proposal step in Metropolis-Hastings.
  - Overdamped Langevin performs better in higher dimensions than standard Metropolis-Hastings because it is equivalent to taking a large number of tiny MH steps, but without the computational effort required to do that. Alternatively, it is an MH diffusion but with an added drift term tending toward higher probability, which keeps it in higher probability areas of high dimensional space.

- Hamiltonian MCMC (HMC)
  - Solutions to the Hamiltonian system of ODEs

$$\frac{d}{dt}y^{(t)} = -J(y^{(t)})\nabla^T H(y^{(t)}) + \text{div}J(y^{(t)})$$

    for scalar H and skew-symmetric J preserve the Boltzmann density $\pi_H(y^{(t)}) \propto \exp(-H(y^{(t)}))$ (these equations are typically used for energy-preserving systems).

- So idea is to discretize a solution to Hamiltonian ODE and choose H such that π is the Boltzmann density. In practice add additional dimensions to achieve irreducibility (can be interpreted as momentum coordinates). Discretization can be done using the Velocity Verlet scheme if J is constant.
- Each MCMC step includes n ODE discretization steps. For $n = 1$, HMC reduces to overdamped Langevin, with $JJ^T = S$. For n too large each step becomes slow. Intermediate n often outperforms overdamped Langevin.

- Underdamped Langevin
  - Combines Hamiltonian MC with overdamped Langevin, $\mathcal{L}_U = \mathcal{L}_H + \mathcal{L}_O$, so

$$\mathcal{L}_U = \frac{1}{\pi_H(x)}\text{div}(\pi_H(x)\nabla f(x)(S+J)(x)).$$

    Preserves the augmented higher-dimensional π_H.
  - One possible discretization (just in the real coordinates) looks like discretized OL but with $S + J$ replacing S in the drift terms.

- Any of these methods can be metropolized (treated as a proposal in a Metropolis-Hastings algorithm) to get rid of the usually-insignificant error in the distributions they converge to.
- The Metropolized overdamped Langevin method is MALA (Metropolis-adjusted Langevin algorithm).

6 Affine Invariant Methods

- A method is affine invariant if applying an affine transformation to a distribution and then sampling it is the same (in distribution) as sampling it and then transforming the samples.
- Overdamped Langevin can be made affine invariant by choosing the diffusion covariance S to be the inverse Hessian, $S(x) = -(D^2 \log \pi(x))^{-1}$. The result is called the stochastic Newton iteration because the first two terms of the iteration are just Newton's method.
- Downsides are that the Hessian may be unavailable or not be spd, in which case we can use an approximation which will be approximately affine invariant.
- Affine Invariant Ensemble Sampler creates approximation to local Hessian using ensemble covariance