Monte Carlo Methods

1 Monte Carlo Integration

In Monte Carlo Integration, we estimate the value of an integral by choosing a collection of random points on our integration domain that we evaluate the integral for. For a given integral

$$I = \int_V f(\mathbf{x}) \mathrm{d}^d\mathbf{x} \tag{1}$$

we then draw N uniformly distributed random vectors $\mathbf{x}_i \in V$ and then calculate

$$I_N = \frac{V}{N} \sum_i^N f(\mathbf{x}_i). \tag{2}$$

It can be shown that I_N converges to I for $N \to \infty$. To quantify how the error of this approximation changes with N, we define $y_i = Vf(\mathbf{x}_i)$ and then rewrite our approximation as

$$I_N \equiv \frac{y_1 + y_2 + \ldots + y_N}{N}. \tag{3}$$

Since this is a sum of independent and identically distributed variables, we can apply the *central limit theorem*. This theorem says that if our variables y_i are distributed with $\mathrm{E}[y_i] = \mu$ and $\mathrm{Var}[y_i] = \sigma^2 < \infty$, then as n approaches infinity, the random variable $\varepsilon = \sqrt{n}\left(\left(\frac{1}{n}\sum_{i=1}^n y_i\right) - \mu\right)$ converges in distribution to a normal $\mathrm{N}(0, \sigma^2)$.

We can therefore express our I_N as $I + \frac{\varepsilon}{\sqrt{n}}$ with ε drawn from $\mathrm{N}(0, \sigma^2)$ in the limit of large N. Therefore, the error of our approximation scales with $1/\sqrt{N}$.

1.1 Comparison to other integration methods

In usual numerical methods for integrating, we subdivide our integration domain into n equally spaced intervals in every dimension. This gives us $N = n^d$ points that we evaluate our function on in order to then approximate the integral with e.g. the midpoint rule or Simpson's rule.

In case of Simpson's rule, our approximation error scales with $\frac{1}{n^4} = \frac{1}{N^{4/d}}$. Clearly, this is superior to our Monte Carlo approach if d is small, but starting with $d = 9$, our error becomes smaller much faster with Monte Carlo for a growing number of function evaluations.

1.2 Importance sampling

Often, our integrand is small on a very large part of the integration domain and large on a very small part of the integration domain. In order to reduce our error further, we'd like to predominantly sample our points in the areas that contribute to the integral significantly. Therefore, we do not want to sample our points $\mathbf{x}_i$ uniformly, but instead from a probability distribution $p(x)$ similar to f but simple to sample from.

Ideally, we'd use $p(x) \propto f(x)$, but often we can not simply sample random numbers from this distribution. Therefore, we employ methods like the rejection method or sampling from a stochastic process.

1.2.1 Stochastic sampling

To sample from a stochastic process, we create a Markov chain. A Markov chain is a series of states

$$x_1 \to x_2 \to x_3 \to \ldots \to x_n \tag{4}$$

with a transition operation t that gives us the next state for every current state. This transition operation may only depend on the current state, not on previous states, i.e. $W_t(x_i \to x_{i+1}) = W(x_{i+1}|x_i)$. We can also apply this transition operation to a whole ensemble of states distributed with a probability function $p(x)$ by applying it as

$$p(x) \to p'(x') = \int p(x) W_t(x \to x'). \tag{5}$$

We now demand (a) that there exists an equilibrium distribution satisfying $p_{eq}(x') = \int p_{eq}(x) W_t(x \to x') \mathrm{d}x$ and (b) that t is ergodic. By ergodic we mean that starting at any x, repeated applications of t must be able to get us arbitrarily close to any other state x'.

It can be shown that this way, an ensemble will approach the equilibrium distribution after a large number of applications of t and that the collection of states on the chain will approach $p(x)$ in distribution.

1.2.2 Metropolis-Hastings algorithm

The Metropolis-Hastings algorithm allows us to build a Markov chain satisfying the conditions we

mentioned for most probability distributions. It is composed of three steps:

1. Propose a state x' using a distribution $q(x \to x')$ that is easy to sample from, e.g. a Gaussian around x.
2. Calculate
$$r = \min\left[1, \frac{p(x')q(x' \to x)}{p(x)q(x \to x')}\right].$$
3. With probability r, set $x_{i+1} = x'$, otherwise $x_{i+1} = x_i$.

We can employ this algorithm to generate a Markov chain using any probability distribution $p(x)$ for which we are able to compute r. It is quite useful in practice that the normalization of $p(x)$ does not need to be known for this purpose.

It is important to note that neighbored states on the chain are highly correlated, only the complete chain resembles the probability distribution well.

1.3 Monte Carlo simulation of lattice models

If we assume a lattice with lattice sites $\mathbf{x}$ and an external field $\phi_{\mathbf{x}}$ at every lattice site, we get the partitioning function as

$$Z = \int \exp\left(-\frac{H(\phi)}{kT}\right) \mathrm{d}\phi_1 \mathrm{d}\phi_2 \ldots \mathrm{d}\phi_N \tag{6}$$

assuming a canonical ensemble. Often, we want to use it to compute thermal averages like

$$\langle A \rangle = \frac{1}{Z} \int A(\phi) \exp\left(-\frac{H(\phi)}{kT}\right) \mathrm{d}\phi_1 \mathrm{d}\phi_2 \ldots \mathrm{d}\phi_N. \tag{7}$$

We can now apply importance sampling with $p(x) \propto \exp\left(-\frac{H(\phi)}{kT}\right)$ by using the Metropolis-Hastings algorithm. In the second step, we will get

$$r = \min\left(1, \exp\left(-\frac{H(\phi') - H(\phi)}{kT}\right)\right) \tag{8}$$

which is easy to evaluate numerically.

As a specific example, we can have a look at the Ising model with its partitioning function

$$Z = \sum_{\{s_{\mathbf{x}}\}} \exp\left(-\beta\left(\frac{1}{2}\sum_{\langle \mathbf{x},\mathbf{y}\rangle}(1 - s_{\mathbf{x}}s_{\mathbf{y}}) + B\sum_{\mathbf{x}} s_{\mathbf{x}}\right)\right) \tag{9}$$

with $s_{\mathbf{x}} = \pm 1$, $\beta = 1/(kT)$, $s_{\mathbf{x}}$ being the spin at site $\mathbf{x}$ and the second sum only summing over neighbored sites.

In this model, we can express the Metropolis-Hastings step as selecting a random lattice site $\mathbf{x}$ and propose to flip the spin of that site. Then, we can compute the change of energy $\delta E_{\mathbf{x}}$, which is easy to evaluate since it only involves neighbored terms. Then, we can accept the proposal with a probability of

$$r = \min(1, \exp(-\beta E_{\mathbf{x}})). \tag{10}$$

Once we have a very long Markov chain, we can compute thermodynamic quantities by just averaging $A(\phi)$ over the chain.

2 Monte Carlo Markov Chains in parameter estimation

In experimental physics, we often measure some data $\mathbf{z} = (z_1, z_2, \ldots, z_n)$ to infer some parameters $\theta = (\theta_1, \theta_2, \ldots, \theta_n)$. Our theoretical model usually gives us $p(\mathbf{z}|\theta)$, but in order to do a fit, we are looking for $p(\theta|\mathbf{z})$. Using Bayes' theorem

$$p(\theta|\mathbf{z}) = \frac{p(\theta)p(\mathbf{z}|\theta)}{p(z)} \tag{11}$$

we can use Metropolis-Hastings with the acceptance probability

$$r = \min\left(1, \frac{p(\theta')p(\mathbf{x}|\theta')q(\theta' \to \theta)}{p(\theta)p(\mathbf{x}|\theta)q(\theta \to \theta')}\right) \tag{12}$$

to sample the posterior $p(\theta|\mathbf{z})$ without even knowing the evidence $p(z)$.

References

- D. P. Landau, K. Binder, A Guide to Monte Carlo Simulations in Statistical Physics 2013 (3rd Edition)
- V. Springel, F. Gräter, Lecture Notes for "Fundamentals of Simulation Methods", 2017
- U. Schwarz, Lecture Notes for "Statistical Physics", 2016
- L. Amendola, Lecture notes for "Statistical Methods", 2017