Lecture 2.1

The Big Picture

Contents

1 Goal

The goal of this section is to furnish a conceptual map. During you travels through Monte Carlo land, you might sometimes lose your bearings and get confused. Returning to this map might be helpful.

The history of Monte Carlo is interesting. It developed somewhat independently in different disciplines (physics and statistics,[1] for example) without much cross-talk for decades. Consequently, techniques from one discipline are constantly rediscovered in another discipline leading to multiple names for basically the same method. For example, “parallel tempering”, “replica exchange MCMC”, “expanded ensembles”, “multi-canonical ensembles” are all closely related methods.

The hope is that a big-picture understanding allows us to look at broad themes, beyond these domain-specific boundaries.

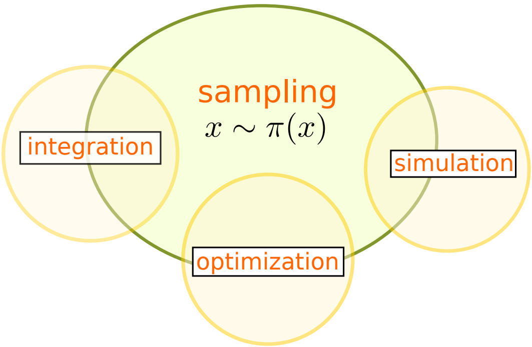
1.1 Scope

A key question to resolve is the connection/difference between:

- Monte Carlo (MC)
- *Markov Chain* Monte Carlo (MCMC)
- Other numerical methods like finite elements, steepest descent, Gauss quadrature etc.

2 Monte Carlo and Mathematical Problems

At its core MC is a **sampling** technique.



It draws or samples x from any desired distribution $\pi(x)$. Typically, x is multidimensional; and $\pi(x)$ is complicated. However, in class, we will start with simple one-dimensional distributions.

MC $\equiv$ sampling. Everything else flows from this central fact.

Three important sub-classes of mathematical problems that can be solved using sampling as its engine are integration, optimization, and simulation. They arise in numerous applications. To be clear, MC is not the only method to solve these problems. However, under the right circumstances may be the most compelling.

[1] for *inference* in statistics, and *statistical mechanics* in physics

2.1 Sampling

Sampling, by itself, is a fundamentally important problem.

$$x \sim \pi(x). \tag{1}$$

Example: Consider any physical model with **parametric uncertainty**.[2] For concreteness, suppose we are interested in modeling how a chemical spill affects a city's drinking water supply. We can write down a mathematical model for groundwater flow, using something like Darcy's law. A key input is permeability of the geology, which is not precisely known. Based on expert knowledge, we may be able to say something about the probability distribution of the permeability. We can then sample from this distribution, run our model to produce an ensemble of forecasts.

Example: **Hard combinatorial problems** are often easier to solve by generating a number of samples. Consider the problem Stan Ulam was grappling with, which might have contributed to the birth of MC.[3] In 1946, Ulam wanted to know the odds that "a particular solitaire laid out with 52 cards would come out successfully. After attempting exhaustive combinatorial calculations, he decided to go for the more practical approach of laying out several solitaires at random and then observing and counting the number of successful plays. This idea of selecting a statistical sample to approximate a hard combinatorial problem is at the heart of modern Monte Carlo simulation."

Example: In certain ways, the project of democracy can be thought of as a sampling problem.

2.2 Integration

As we talked about previously, integration problems like

$$I = \int g(x)\, dx, \tag{2}$$

can be solved using MC.

Example: In **inference**, we often carry out operations of normalization, marginalization, and expectation, which are integration problems in disguise. The expected value of a function $f(x)$, which depends on a random variable x may be written as:

$$E_\pi[f] = \int f(x)\, \pi(x)\, dx. \tag{3}$$

$\pi(x)$ highlights the relationship to the sampling problem.

Example: In **statistical mechanics**, the partition function of a system can be computed by integrating over all (N) particle positions and momenta,

$$Z = \int d\mathbf{p}^N \int d\mathbf{r}^N f(\mathbf{r}^N, \mathbf{p}^N), \tag{4}$$

where the function $f(\mathbf{r}^N, \mathbf{p}^N)$ is proportional to the probability of the system being in state $(\mathbf{r}^N, \mathbf{p}^N)$ and is related to the Hamiltonian of the system.

[2] You don't have to try to hard to find such models. They are the norm, rather than the exception.

[3] Andrieu et al., "An Introduction to MCMC for Machine Learning", *Machine Learning*, 50, 5-43, **2003**.

2.3 Optimization

$$x^* = \max_x f(x), \tag{5}$$

where x^* is the value at which $f(x)$ is maximized.

Example: Any difficult combinatorial optimization (NP-complete) problem like knapsack, traveling salesman, etc. can be solved using stochastic optimization methods like **simulated annealing**. Constrained optimization problems that can be solved using simulated annealing include floor planning in chip design, where the goal is to efficiently pack rectangles of different sizes on a motherboard, subject to constraints that arise from heat dissipation.

Example: Global optimization methods like basin hopping combines random displacements with local optimization. So called "histogram re-weighting" methods are also useful to find narrow but deep regions on the domain.

2.4 Simulation

This is a general term, which defines the evolution of a system subject to a particular stochastic mathematical model or rule. This may include standard PDE type models with parametric uncertainty (discussed earlier). Beyond these, it includes models that are stochastic at their heart.

Example: The Langevin equation in physics considers the motion of a particle subject to Brownian motion. Once can often write down a master equation to describe the underlying probability distribution. The system can be studied using kinetic or dynamic Monte Carlo.

Example: An example we shall study in this class is traffic modeling using the Nagel-Schreckenberg model. It is a simple cellular automaton model for road traffic flow that can reproduce the phenomenology of emergent traffic jams.

3 Relationship between Monte Carlo and Other Methods

A certain subclass of problems, say integration, can be attacked using MC or non-MC methods. Examples of non-MC methods the three sub-classes include:

- **Integration**

  analytical, Gauss quadrature, Newton-Cotes, Clenshaw-Curtis etc.

- **Optimization**

  steepest descent, conjugate-gradient, linear-programming, Levenberg-Marquardt etc.

- **Simulation**

  analytical, finite elements, finite differences, molecular dynamics, etc.

3.1 MC and MCMC

We prefer direct Monte Carlo to MCMC!

Why?

The key reason is that individual samples in MC[4] are independent. This simplifies error analysis. What does this mean? Recall that when we previously solved integration problems using MC, the "answer" varied from one calculation to the other.[5] We could quantify the variability by repeating the calculation, say 100 times, and reporting the standard deviation of the estimates.

What if we can only perform the calculation once? This is often the case for large problems, which take a long time to solve. In such cases, we don't have the luxury of repeating the calculation 100 times. Can we get an error estimate in such cases? It turns out that the answer is (a qualified) yes; but it needs more sophisticated techniques.

However many complex problems cannot be tacked with direct MC, and MCMC becomes the method of last resort. Samples in MCMC are correlated. This complicates error analysis. It is still possible to get error estimates, but we have to do a lot more work, and live with less fundamental grounding.[6]

[4] MC and direct MC are used synonymously

[5] it depended on the precise sequence of random numbers generated in a calculation

[6] A weak, but possibly helpful analogy may be, "analytical : numerical :: MC : MCMC". In numerical solutions, we have to worry about tolerance, stability, round-off error, convergence, choice of method etc, as the price for being able to solve a wider range of problems.