Technion — Department of Electrical Engineering

Monte Carlo Methods for Computation and Optimization (048715)

Lecture Notes

Prof. Nahum Shimkin

Spring 2015

PREFACE

These lecture notes are intended for a first, graduate-level, course on Monte-Carlo simulation methods. The required prerequisites are merely an elementary course on probability and statistics, and a course on stochastic processes (emphasizing Markov chains).

There exists many excellent textbooks on this subject, with differing emphasis. Listed below are the main ones that were consulted in preparing these notes.

(1) R. Rubinstein and D. Kroese, *Simulation and the Monte Carlo Method,* Wiley, 2008.

(2) C. Robert and G. Casella, *Monte Carlo Statistical Methods,* 2nd ed., Springer, 2005.

(3) S. Asmussen and P. Glynn, *Stochastic Simulation,* Springer, 2007.

(4) J. Liu, *Monte Carlo Strategies in Scientific Computing,* Springer, 2008.

Additional specific references will be given as needed.

1 Introduction

In their most basic form, Monte Carlo schemes compute the *expected value* of random variables of interest, by averaging random draws from an appropriate probability distribution. This simple idea gives rise to a wide variety of methods and algorithms, which are needed to facilitate this computation and render it efficient.

A notable feature of Monte Carlo methods is their wide applicability, making them the method of choice for many complex problems where other (analytical or numeric) methods fail. Essentially – whenever we can simulate a random variable or process, we can calculate probabilities and expectations by averaging over these simulations.

1.1 Historical Background

A first systematic use of Monte Carlo methods can be traced to the development of atomic weapons in Los Alamos during the years 1944-5, using the early digital computers that were developed at that time. Notable names related to these initial efforts include Stanislaw Ulam, John von Neumann, Nickolas Metropolis, and Enrico Fermi. In the early 1950s, statistical physicists introduced Markov-chain based methods for simulation of simple fluids. This method was gradually extended to simulated more complex physical systems, such as spin glass models and many others. In the 1980s, statisticians and computer scientists developed Monte-carlo methods for a wide variety of problems such as combinatorial optimization, statistical inference, likelihood computation, and Bayesian modeling.

Today, Monte Carlo methods find extensive use in a wide range of scientific disciplines, that include physics, chemistry, computer science, economics and finance, operations research, and statistics. We will not attempt to cover in detail the application areas, but rater focus on the basic methods, complemented by some illustrative examples of applications in selected areas.

1.2 Example: Approximating π

Buffon's needle: An early experiment known as "Buffon's needle" (dating back to 1777) serves to illustrate the basic idea and potential of elementary Monte Carlo methods.

In this experiment, one draws a grid of parallel lines with spacing D on a flat surface, and repeatedly throws a needle of length l into that grid. Under ideal conditions, the chances that the needed will intersect at least one of the drawn lines is easily computed to be $2l/\pi D \stackrel{\triangle}{=} a$. Thus, if we let $\hat{p}_N$ denote the proportion of intersects out of N draws, by the law of large numbers we get that

$$\lim_{N\to\infty} \hat{p}_N = \frac{2l}{\pi D}$$

from which we can approximate $\pi \approx 2l/D\hat{p}_N$.

To get an idea of the required number of samples for a given accuracy, note that $E(\hat{p}_N) = a$, and

$$\mathrm{Var}(\hat{p}_N) = \frac{1}{N}\mathrm{Var}(\hat{p}_1) = \frac{a(1-a)}{N}$$

(since $\hat{p}_1$ is a Bernoulli RV with parameter a). Therefore, by Markov's inequality,

$$P\{|\hat{p}_N - a| \geq \epsilon\} \leq \frac{\mathrm{Var}(\hat{p}_N)}{\epsilon^2} = \frac{a(1-a)}{N\epsilon^2}\,.$$

Thus, to get an accuracy of $\epsilon = 10^{-3}$ with 99% confidence, we need $\frac{a(1-a)}{N10^{-6}} \leq 0.01$, or $N \geq 10^8/a(1-a)$. Obviously, an improvement in efficiency will be helpful here.

Circle in a Square: A related experiment is the following. Draw a circle of diameter 1 inside a square of side 1. Now throw a small ball inside the square, and let p_N denote the proportion of throws that land inside the circle. Under ideal conditions, the chances for that are just $\pi/4$, and we can approximate $\pi \approx 4\hat{p}_N$ for N large.

The last experiment is easily simulated: Draw at each stage two independent random variables X and Y that are uniformly distributed on $[0,1]$, and count the fraction of times in which $X^2 + Y^2 \leq 1$. Is the same as p_N in our circle-in-square experiment.

Question: Repeat the last experiment with a square of side 10. How does this compare to the above? Which option is better?

1.3 The Basic Framework

The basic problem in stochastic simulation is to estimate the expected value of a function h of some random variable X:

$$\ell \stackrel{\triangle}{=} E(h(X))$$

The most basic approach (called *crude Monte-Carlo*) is to use a sequence of i.i.d. samples of X, namely $X_i \sim f_X$, and compute the average:

$$\hat{\ell} = \frac{1}{n}\sum_{i=1}^{n} h(X_i)$$

Note that *probabilities* of events can be computed as the expected value of their indicator functions:

$$P(\mathbf{x} \in A) = E(1_{\{\mathbf{x}\in A\}})$$

Basic issues that arise in this context include:

1. Efficient sampling: How to sample efficiently from a required probability distribution f_X (which may be complicated, multivariate, and implicitly specified).

2. Variance reduction: How to modify the basic scheme to reduce the variance of the estimate (so that a smaller number of samples will suffice for a given accuracy).

The central methods that address the first point include *rejection sampling*, and, most importantly, *Markov Chain Monte Carlo* (MCMC).

Methods that address the second issue include *Importance Sampling*, along with many other and more specific schemes.

1.4 Some Application Examples

1.4.1 Statistical Physics

The state probabilities of physical systems are often described by the *Boltzmann distribution* (or Gibbs distribution) of the form

$$\pi(\mathbf{x}) = \frac{1}{Z} e^{-U(\mathbf{x})/kT} ,$$

where $\mathbf{x} \in X$ is the system configuration or state, $U(\mathbf{x})$ its potential energy, T is the temperature, and k the Boltzmann constant. The normalization constant $Z = Z(T)$ is referred to as the *partition function.*

A canonical example is the **Ising model**, which describes the interaction of magnetic dipoles in a magnetic material. Here $\mathbf{x} = (x_i)_{i\in L}$, where L is a two or three dimensional discrete domain, and $x_i \in \{-1, 1\}$ is the random state (orientation) of the dipole in position i. The potential energy is

$$U(\mathbf{x}) = -\frac{J}{2} \sum_{i \in X, j \in N_i} x_i x_j + \sum_i h_i x_i \,.$$

Here N_i are the neighbors of i, J the interaction strength, and h_i the external magnetic field.

Macroscopic quantities of interest include expected values of various function of the state. For example, $\langle U \rangle = EU(\mathbf{x})$ is the (mean) potential energy, $\langle m \rangle = E(|\sum_i x_i|)/|L|$ is the mean magnetization per dipole, etc. These averages can be computed using Monte-Carlo sampling from $\pi(\mathbf{x})$.

1.4.2 Statistical Inference

Suppose we wish to estimate the value of a parameter vector θ based on measurements x. In a Bayesian setting, the posterior distribution of θ is given by

$$\pi(\theta|x) = \frac{f(x|\theta)\pi(\theta)}{\int f(x|\theta)\pi(\theta)d\theta}$$

Therefore

$$\hat{\theta}(x) \overset{\triangle}{=} E(\theta|x) = \int \theta\pi(\theta|x)d\theta$$

$$\text{cov}(\theta|x) = \text{cov}(\theta - \hat{\theta}(x)|x) = \int (\theta - \hat{\theta}(x))(\theta - \hat{\theta}(x))^T \pi(\theta|x)d\theta$$

These computations, which involve the expected value of certain functions, can be carried out using Monte-Carlo sampling methods. Obviously, we need to be able to sample from the conditional distribution $\pi(\theta|x)$.

1.4.3 Queueing Networks

Networks of queues are a common model for various communication, computer, and service systems. Here, customers (jobs, packets) arrive to the system, queue up, and are served in one or more service stations.

Queuing networks generally belong to the class of discrete-event stochastic systems, and can be simulated using simulation methods and tools available for such systems. Based on a set of *primitive* random variables, such as the arrival and service times of the customers, the system can be simulated forward in time, and the quantities of interest (such as queue sizes, delays) can be computed.

Monte Carlo simulations are an important tool for performance analysis and parameter optimization in such systems. Quantities of interest may include:

- The expected delay of a job from arrival to service completion.
- The fraction of time that a certain buffer is full.
- The expected number of rejected calls.
- The expected time till some buffer becomes full, starting from an empty system.

Note that the first three quantities are steady-state ones, which the last is a single-shot quantity. Accordingly, the first may be computed from a single, long simulation run of the system, while the last requires repeated trials from an empty system.

We note that events such as "buffer full" may be rare events, which require excessive running time to encounter and estimate directly. In these cases, specific methods are required to handle rare event simulations.

1.4.4 Sampling Web Content

Various statistics about the content of usage of the Web are of interest. Due to its extent, it is obviously not possible to go over its entire content, and sampling methods are required.

Consider for example the specific problem of sampling random pages from the corpus of documents indexed by a search engine, using only the search engines public interface. Simple-minded sampling methods suffer from a bias that exists in favor of long documents. More elaborate use of Monte Carlo sampling can overcome this problem.

1.4.5 Combinatorial Optimization

Suppose we wish to minimize a function $c(x)$, $x \in X$, where X is a large discrete set. This problem can be posed as a stochastic simulation problem by defining the probability distribution

$$\pi(x) = \frac{1}{Z} \exp(-c(x)/T) \,.$$

Evidently, for small T, this distribution is concentrated on points x with small cost. Thus, drawing sample from π will give low-cost x's with high probability.

Sampling from π is often carried out using MCMC methods. This is closely related to the Simulated Annealing algorithm for global optimization, which is in fact a special case.