Monte Carlo integration

Monte Carlo integration is a cubature where the nodes are chosen randomly. Typically no assumptions are made about the smoothness of the integrand, not even that it is continuous.

Plain Monte Carlo algorithm distributes points uniformly throughout the integration region using either uncorrelated pseudo-random or correlated quasi-random sequences of points.

Adaptive algorithms, such as VEGAS and MISER, distribute points non-uniformly in an attempt to reduce integration error. They use correspondingly *importance* and *stratified* sampling.

Multi-dimensional integration

One of the problems in multi-dimensional integration is that the integration region Ω is often quite complicated, with the boundary not easily described by simple functions. However, it is usually much easier to find out whether a given point lies within the integration region or not. Therefore a popular strategy is to create an auxiliary rectangular volume V which contains the integration volume Ω and an auxiliary function F which coincides with the integrand inside the volume Ω and is equal zero outside. Then the integral of the auxiliary function over the (simple rectangular) auxiliary volume is equal the original integral.

Unfortunately, the auxiliary function is generally non-continuous at the boundary and thus the ordinary quadratures which assume continuous integrand will fail badly here while the Monte-Carlo quadratures will do just as good (or as bad) as with continuous integrand.

Plain Monte Carlo sampling

Plain Monte Carlo is a quadrature with random abscissas and equal weights ,

$$\int_V f(\mathbf{x})dV \approx w \sum_{i=1}^{N} f(\mathbf{x}_i) \ , \tag{1}$$

where $\mathbf{x}$ a point in the multi-dimensional integration space. One free parameter, w, allows one condition to be satisfied: the quadrature has to integrate exactly a constant function. This gives $w = V/N$,

$$\int_V f(\mathbf{x})dV \approx \frac{V}{N} \sum_{i=1}^{N} f(\mathbf{x}_i) = V\langle f \rangle \ . \tag{2}$$

According to the *central limit theorem* the error estimate ϵ is close to

$$\epsilon = V\frac{\sigma}{\sqrt{N}} \ , \tag{3}$$

where σ is the variance of the sample,

$$\sigma^2 = \langle f^2 \rangle - \langle f \rangle^2. \tag{4}$$

The $1/\sqrt{N}$ convergence of the error, typical for a random process, is quite slow.

Table 1: Plain Monte Carlo integrator

Importance sampling

Suppose that the points are distributed not uniformly but with some density $\rho(x)$: the number of points Δn in the volume ΔV around point x is given as

$$\Delta n = \frac{N}{V}\rho\Delta V, \tag{5}$$

where ρ is normalised such that $\int_V \rho dV = V$.

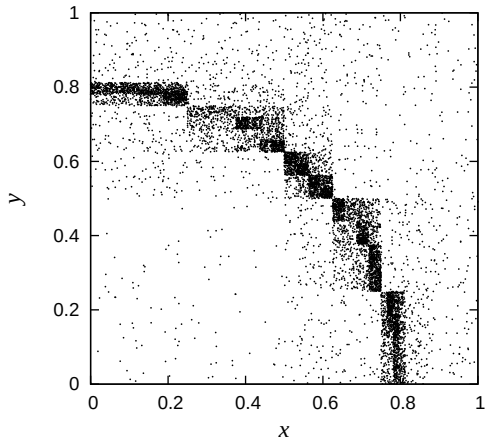

Figure 1: Stratified sample of a discontinuous function,
$f(x, y) = (x^2 + y^2 < 0.8^2)\ ?\ 1 : 0$

Table 2: Recursive stratified sampling

The estimate of the integral is then given as

$$\int_V f(\mathbf{x})dV \approx \sum_{i=1}^{N} f(\mathbf{x}_i)\Delta V_i = \sum_{i=1}^{N} f(\mathbf{x}_i)\frac{V}{N\rho(\mathbf{x}_i)} = V\left\langle \frac{f}{\rho} \right\rangle \ , \tag{6}$$

where

$$\Delta V_i = \frac{V}{N\rho(x_i)} \tag{7}$$

is the "volume per point" at the point x_i.

The corresponding variance is now given by

$$\sigma^2 = \left\langle \left(\frac{f}{\rho}\right)^2 \right\rangle - \left\langle \frac{f}{\rho} \right\rangle^2 \ . \tag{8}$$

Apparently if the ratio f/ρ is close to a constant, the variance is reduced.

It is tempting to take $\rho = |f|$ and sample directly from the function to be integrated. However in practice it is typically expensive to evaluate the integrand. Therefore a better strategy is to build an approximate density in the product form, $\rho(x, y, \ldots, z) = \rho_x(x)\rho_y(y)\ldots\rho_z(z)$, and then sample from this approximate density. A popular routine of this sort is called VEGAS. The sampling from a given function can be done using the *Metropolis algorithm* which we shall not discuss here.

Stratified sampling

Stratified sampling is a generalisation of the recursive adaptive integration algorithm to random quadratures in multi-dimensional spaces.

The ordinary "dividing by two" strategy does not work for multi-dimensions as the number of subvolumes grows way too fast to keep track of. Instead one estimates along which dimension a subdivision should bring the most dividends and only subdivides along this dimension. Such strategy is called *recursive stratified sampling*. A simple variant of this algorithm is given in table .

In a stratified sample the points are concentrated in the regions where the variance of the function is largest, as illustrated on figure .

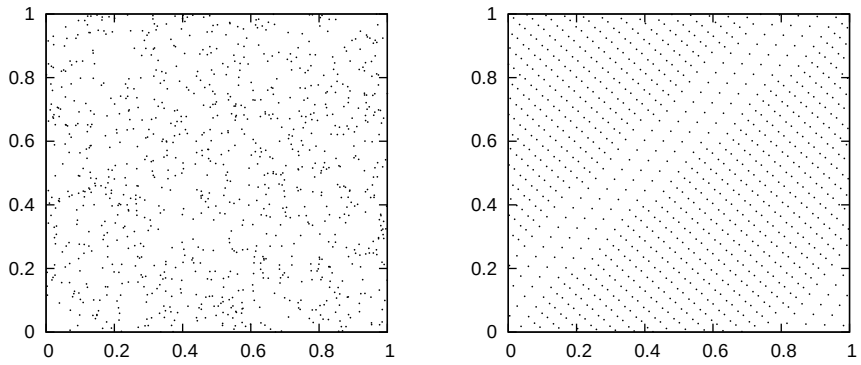

Figure 2: Typical distributions of pseudo-random (left) and quasi-random (right) points in two dimensions.

Quasi-random (low-discrepancy) sampling

Pseudo-random sampling has high discrepancy[1] – it typically creates regions with hight density of points and other regions with low density of points, as illustrated on fig. 2. With pseudo-random sampling there is a finite probability that all the N points would fall into one half of the region and none into the other half.

Quasi-random sequences avoid this phenomenon by distributing points in a highly correlated manner with a specific requirement of low discrepancy, see fig. 2 for an example. Quasi-random sampling is like a computation on a grid where the grid constant must not be known in advance as the grid is ever gradually refined and the points are always distributed uniformly over the region. The computation can be stopped at any time.

By placing points more evenly than at random, the quasi-random sequences try to improve the $1/\sqrt{N}$ convergence rate of pseudo-random sampling.

The central limit theorem does not work in this case as the points are not statistically independent. Therefore the variance can not be used as an estimate of the error. The error estimation is actually not trivial. In practice one can employ two different sequences and use their difference as the error estimate.

Quasi-random sequences can be roughly divided into *lattice rules* and *digital nets* (see e.g. arXiv:1003.4785 [math.NA] and references therein).

Lattice rules

In the simplest incarnation a lattice rule can be defined as follows.

Let α_i, $i = 1, \ldots, d$, (where d is the dimension of the integration space) be a set of cleverly chosen irrational numbers, like square roots of prime numbers. Then the kth point (in the unit volume) of the sequence is given as

$$\mathbf{x}^{(k)} = \{\mathrm{frac}(k\alpha_1), \ldots, \mathrm{frac}(k\alpha_d)\} \,, \tag{9}$$

where $\mathrm{frac}(x)$ is the fractional part of x.

A problem with this method is that a high accuracy arithmetics (e.g. `long double`) might be needed in order to generate a reasonable amount of quasi-random numbers.

[1] discrepancy is a measure of how unevenly the points are distributed over the region.