

Lecture #11Quasi Monte Carlo :

Unlike ordinary Monte Carlo , the Quasi Monte Carlo methods make no attempt to mimic randomness. In fact, they seek to increase accuracy by generating points that are too evenly distributed to be random.

General Principles: In order to set forth the theory of Quasi Monte Carlo (QMC), it is customary to consider the problem of numerical integration over the unit hypercube. Each replication in a Monte-Carlo simulation can be interpreted as the result of applying a series of transformations (implicit in the simulation algorithm) to an input sequence

of independent uniformly distributed random variables,
 U_1, U_2, \dots

Suppose there is an upper bound "d" on the number
of uniforms required to produce a simulation output and
let $f(U_1, U_2, \dots, U_d)$ denote this output.

For example, f may be the result of transformations that
convert the U_i to normal random variables, the normal random
variables to paths of underlying assets, and the paths to

the discounted payoff of a derivative security.

We suppose that the objective is to calculate :

$$E[f(u_1, u_2, \dots, u_d)] = \int_{[0,1]^d} f(x) dx \quad \textcircled{1}$$

QMC approximates this integral using,

$$\int_{[0,1]^d} f(x) dx \approx \frac{1}{n} \sum_{i=1}^n f(x_i) \quad \textcircled{2}$$

for carefully (and deterministically) chosen points x_1, x_2, \dots, x_n
in the unit hypercube $[0, 1]^d$.

The dependence of QMC methods on the problem dimension
is one of the features that most distinguishes them from
Monte Carlo. If two different Monte Carlo algorithms
corresponding to functions $f: [0, 1]^{d_1} \rightarrow \mathbb{R}$ and
 $g: [0, 1]^{d_2} \rightarrow \mathbb{R}$ resulted in $f(U_1, U_2, \dots, U_{d_1})$ and

$g(U_1, U_2, \dots, U_{d_2})$ having the same distribution, then these two algorithms would have the same bias and variance properties.

Accordingly, the preferred algorithm would be the one that requires less time to evaluate. The dimensions d_1 and d_2 are irrelevant except to the extent that they affect the computing times.

Unlike ordinary Monte Carlo, in QMC the dimension must be identified explicitly before points can be generated. Lower-dimensional representations generally result in smaller errors. Without an upper bound d , QMC methods are inapplicable.

Note: We focus on problems with a finite dimension d and consider approximations of the form (2).

Goal: Low-discrepancy methods to be used to construct points x_i that makes the error in ② to be small for a large class of integrands f .

Intuitively: This is equivalent to choosing the points x_i to fill the hypercube uniformly.

Discrepancy: We begin with the precise notion of uniformity - or deviation from uniformity.

Given a collection \mathcal{A} of (Lebesgue measurable) subsets of $[0, 1]^d$, the discrepancy of the point set $\{x_1, x_2, \dots, x_n\}$ relative to \mathcal{A} is,

$$D(x_1, x_2, \dots, x_n; \mathcal{A}) = \sup_{A \in \mathcal{A}} \left| \frac{\#\{x_i \in A\}}{n} - \text{vol}(A) \right| \quad \textcircled{3}$$

Here, $\#\{x_i \in A\}$ denotes the number of x_i contained in A and $\text{vol}(A)$ denotes the volume (measure) of A .

Thus, the discrepancy is the supremum over errors in integrating the indicator function of A using the points x_1, x_2, \dots, x_n .

Taking A to be the collection of all rectangles on $[0, 1]^d$ of the form :

$$\prod_{j=1}^d [u_j, v_j], \quad 0 \leq u_j < v_j \leq 1,$$

yields the ordinary (or extreme) discrepancy $D(x_1, x_2, \dots, x_n)$.

Restricting A to rectangles of the form:

$$\prod_{j=1}^d [0, u_j) \quad \text{—— } 4$$

defines the star discrepancy $D^*(x_1, x_2, \dots, x_n)$.

The star discrepancy is obviously no larger than the ordinary discrepancy. It has been shown that,

$$D^*(x_1, x_2, \dots, x_n) \leq D(x_1, x_2, \dots, x_n) \leq 2^d D^*(x_1, x_2, \dots, x_n).$$

For $d=1$, it has been shown that,

$$D^*(x_1, x_2, \dots, x_n) \geq \frac{1}{2n}, \quad D(x_1, x_2, \dots, x_n) \geq \frac{1}{n}, \quad \text{--- (5)}$$

and that in both cases, the minimum is attained by

$$x_i = \frac{2i-1}{2n}, \quad i = 1, 2, \dots, n. \quad \text{--- (6)}$$

For this set of points x_i (2) reduces to the midpoint rule for integration over the unit interval.

Suppose, in contrast, that we fix an infinite sequence x_1, x_2, \dots

of points in $[0,1)$ and measure the discrepancy of the first n points. From the perspective of numerical integration, this is a more relevant case, if we hope to be able to increase the number of points in an approximation of the form (2). It has been shown that,

$$D(x_1, x_2, \dots, x_n) \geq D^*(x_1, x_2, \dots, x_n) \geq \frac{c \log n}{n}$$

for infinitely many n , with c a constant.

This is typical of low discrepancy methods, and also in higher dimensions. One can achieve a lower discrepancy by fixing the number of points " n " in advance.

Using the first " n " points of a sequence rather than a different set of points for each " n " typically increases discrepancy by a factor of $\log n$. Much less is known about the best possible discrepancy in dimension more than 1.