# Monte Carlo Algorithm (10C)

Please send corrections (or suggestions) to youngwlim@hotmail.com.

This document was produced by using OpenOffice and Octave.

# Monte Carlo

- Monte Carlo Method (Monte Carlo Simulation)
- Monte Carlo Algorithm

# Monte Carlo Method

a broad class of computational algorithms
that rely on repeated random sampling
to obtain numerical results.

Their essential idea is using randomness
to solve problems that might be deterministic in principle.

They are often used in physical and mathematical problems
and are most useful when it is difficult or impossible
to use other approaches.

used in three distinct problem classes:
- optimization,
- numerical integration
- generating draws from a probability distribution.

https://en.wikipedia.org/wiki/Monte_Carlo_method

# Patterns of Monte Carlo Methods

Monte Carlo methods vary, but tend to follow a particular pattern:

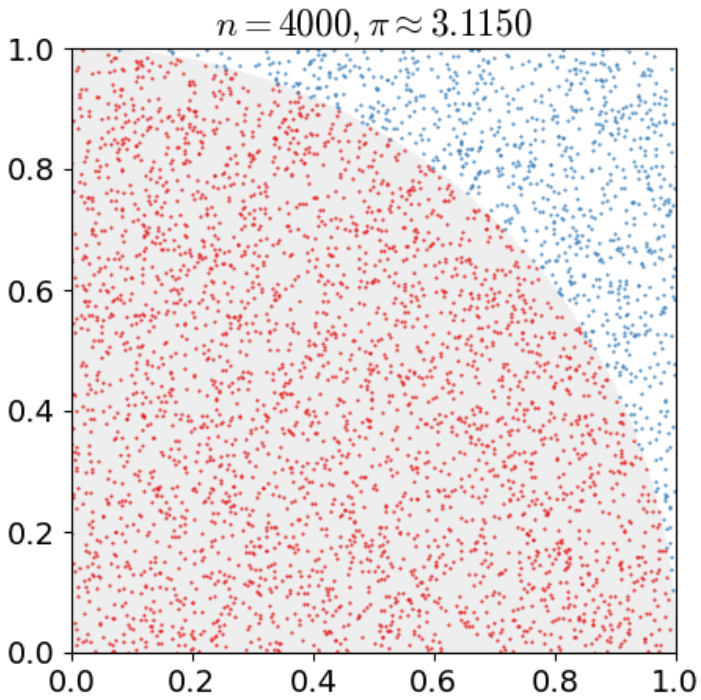
- Define a domain of possible inputs.
- Generate inputs randomly from a probability distribution over the domain.
- Perform a deterministic computation on the inputs.
- Aggregate the results.

https://en.wikipedia.org/wiki/Monte_Carlo_method

# Examples of Monte Carlo Methods

For example, consider a circle inscribed in a unit square. Given that the circle and the square have a ratio of areas that is π/4, the value of π can be approximated using a Monte Carlo method:

1) Draw a square, then inscribe a circle within it.
2) Uniformly scatter objects of uniform size over the square.
3) Count the number of objects inside the circle and the total number of objects.
4) The ratio of the inside-count and the total-sample-count is an estimate of the ratio of the two areas, which is π/4. Multiply the result by 4 to estimate π.




https://en.wikipedia.org/wiki/Monte_Carlo_method#/media/File:Pi_30K.gif

https://en.wikipedia.org/wiki/Monte_Carlo_method

# Monte Carlo Algorithms

In computing, a Monte Carlo algorithm is a randomized algorithm whose output *may be incorrect* with a certain (typically small) probability.

The name refers to the grand casino in the Principality of Monaco at Monte Carlo, which is well-known around the world as an icon of gambling.

https://en.wikipedia.org/wiki/Monte_Carlo_algorithm

# One-sided and Two-sided Errors

a deterministic algorithm returns a correct answer
a Monte Carlo algorithm does *not always* return a correct answer

**one-sided error** algorithms

false-biased algorithms : correct false answer
true-biased alogrithms : correct true answer

a false-biased Monte Carlo algorithm is always correct when it returns false

a true-biased Monte Carlo algorithm is always correct when it returns true.

https://en.wikipedia.org/wiki/Monte_Carlo_algorithm

**two-sided error** algorithms
no bias

the answer they provide (either true or false) will be incorrect, or correct, with some bounded probability.

# Amplification

For a Monte Carlo algorithm with one-sided errors,

the failure probability can be reduced

and the success probability amplified

by running the algorithm k times.

https://en.wikipedia.org/wiki/Monte_Carlo_algorithm

# The Solovay–Strassen primality test (1)

determine whether a given number is a prime number

for prime number inputs, it always answers true;

for composite inputs, it answers
**false** with probability at least ½ and
**true** with probability less than ½.

Thus, **false** answers from the algorithm are certain to be correct,
whereas the true answers remain *uncertain*;

**a ½-correct false-biased algorithm**

https://en.wikipedia.org/wiki/Monte_Carlo_algorithm

# The Solovay–Strassen primality test (2)

One may run this algorithm multiple times returning a false answer if it reaches a false response within k iterations, and otherwise returning true.

Thus, if the number is prime then the answer is always correct, and if the number is composite then the answer is correct with probability at least $1-(1-\frac{1}{2})^k = 1-2^{-k}$.

https://en.wikipedia.org/wiki/Monte_Carlo_algorithm

## References

[1] http://en.wikipedia.org/
[2] https://en.wikiversity.org/wiki/Understanding_Information_Theory