Monte Carlo Simulation

Jaqueline V Moreira

2025-04-24

Monte Carlo Simulation

the MC simulation is a concept that follows the Central Limit Theorem and the Law of Large Numbers.

A large number of computational samplings (pseudo-random values generated from a probability distribution function - PDF) are performed until the obtained result converges to the expected behavior of the population.

This method is used to determine the probability of occurrence of a series of outcomes and to solve complex integrals that cannot be solved analytically.

Monte Carlo methods are mainly used in three distinct problem classes: **optimization, numerical integration, and generating draws from a probability distribution**. They can also be used to model phenomena with significant uncertainty in inputs, such as calculating the risk of a nuclear power plant failure. Monte Carlo methods are often implemented using computer simulations, and they can provide approximate solutions to problems that are otherwise intractable or too complex to analyze mathematically.

In principle, Monte Carlo methods can be used to solve any problem having a probabilistic interpretation. By the law of large numbers, integrals described by the expected value of some random variable can be approximated by taking the empirical mean (a.k.a. the 'sample mean') of independent samples of the variable.

Many problems that involves integrals can be represented as expectation and therefore can benefit from the law of large numbers

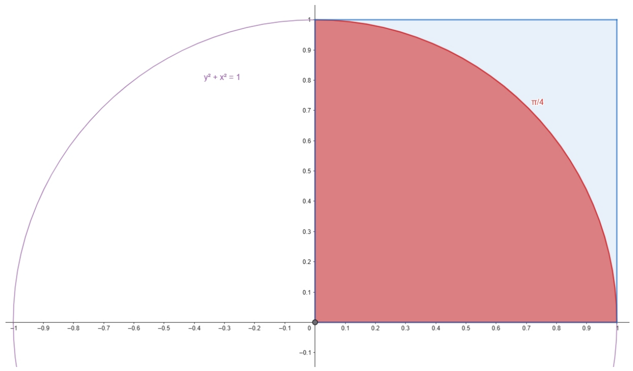
Monte Carlo Simulation Applications

Estimating π Using Naive Monte Carlo (hit or miss)

you can approximate it by considering the area under f(x) as a fraction of a known bounding box.

Step 1: Define the Problem

- Consider a unit circle (radius = 1) centered at the origin
- The equation of the circle is: $x^2 + y^2 <= 1$
- The area of the circle is $= \pi * 1^2$, so the quarter-circle area (in the first quadrant) is $= \frac{\pi}{4}$



Step 2: Monte Carlo Sampling

- Generate random points (x,y) where $x, y \sim U(0, 1)$ (uniformly sampled from the unit square).
- Check if each point lies inside the quarter-circle using $x^2 + y^2 \leq 1$.
- Compute the ratio of points inside the circle to the total points: $\frac{\text{Points inside quarter-circle}}{\text{Total points}} \approx \frac{\pi}{4}$
- By the end, multiplying the ratio by 4 the outcome will converge to π:

$$\pi \approx 4 \times \frac{\text{Points inside}}{\text{Total points}}$$

Step 3: Run the Simulation Samples will be drawn from a uniform distribution, ensuring an equal probability of selection, within the range $0 < x < 1$ and $0 < y < 1$. For each sample, the following condition will be evaluated:

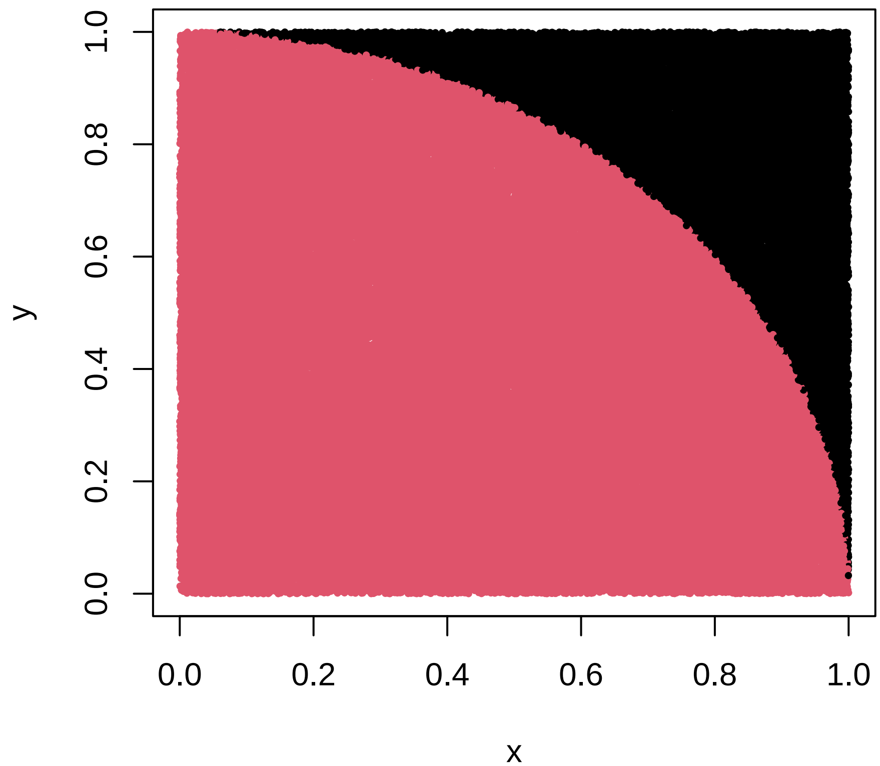
$$x^2 + y^2 <= 1$$

If the condition is satisfied (meaning the sampled point lies inside the defined region), it will be counted as a match (inside=TRUE) At the end of the simulation, the proportion of matches relative to the total number of trials will be multiplied by 4 to estimate the desired area.

```
## [1] "The estimated value of Pi results in 3.14132 with 1e+05 trials and 78534 matches"
```

```
## [1] "The analitycal value of pi is = 3.14159"
```

```
## [1] "a difference of  0.00027"
```



Solving an Integral computationally by Mean Value Estimation method

Introduction If you have a definite integral of the form $I = \int_a^b f(x)dx$ you can approximate it using Monte Carlo by treating it as an **expectation**. In probability theory, the expectation (or mean) of a continuous random variable X with probability density function p(x) is given by the following **general equation**:

$$E[g(X)] = \int_{-\infty}^{\infty} g(x)p(x)dx$$

- g(x) is any function of X We use g(x) here to emphasize that expectation applies to any function of a random variable, not just the specific case of numerical integration.
- p(x) is the probability density function (PDF) of X

This means that the expectation of a function g(X) with respect to X is just an integral of the function g(x) weighted by the pdf(X).

In Monte Carlo integration, our goal is to estimate an integral of a function f(x) over some interval [a,b]: $I = \int_a^b f(x)dx$

To express this in expectation form, we introduce a probability density function p(x) that defines how we sample x, in other words, which pdf will originate the samples. A simple choice is the uniform distribution over [a,b] that guarantees that the samples will be equally likely.

$$\text{The pdf of an uniform distribution} = \frac{1}{b-a}, \text{ for x in [a,b]}$$

OBS: Since this pdf integrates to 1 over [a,b], it is a p(x).

Substituting this into the expectation formula we set g(x)=f(x), so that:

$$E[f(X)] = \int_a^b f(x)p(x)dx = \int_a^b f(x)\frac{1}{b-a}dx$$
$$E[f(X)] = \int_a^b f(x)\frac{1}{(b-a)}dx$$

The constant can be put out of the integral

$$E[f(X)] = \int_a^b \frac{1}{(b-a)}f(x)dx$$

Multiplying both sides by (b—a), we recover our integral

$$(b-a) * E[f(X)] = \frac{(b-a)}{(b-a)}\int_a^b f(x)dx$$
$$I = (b-a) * E[f(X)]$$

The expectation $E[f(X)]$ can be estimated using Monte Carlo sampling:

$$E[f(X)] \approx \frac{1}{N}\sum_{i=1}^{N} f(x_i)$$

where x_i are N random samples drawn from U~(a,b). Substituting this into our integral formula:

$$I \approx (b-a) \times \frac{1}{N}\sum_{i=1}^{N} f(x_i)$$

That's the Monte Carlo estimator for the integral.

Example 1 One can obtain the approximate value of the integral $\int_0^{10} x^2dx$ through the approximation

$$(10-0)\frac{1}{N}\sum_{n=1}^{N} x^2$$

where x_n are random samples drawn uniformly from the interval [0,10].

Simulation

$$\int_0^{10} x^2dx \approx (10-0)\frac{1}{N}\sum_{n=1}^{N} x^2$$

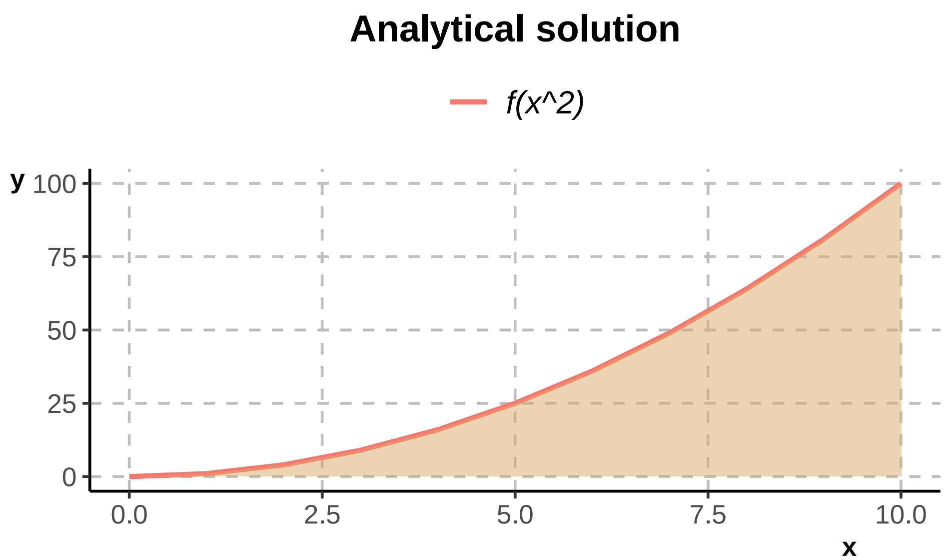
```
## [1] "integral computacional obtida com N=10^4     333.92948"

## [1] "integral computacional obtida com N=10^5     332.42351"

## [1] "integral computacional obtida com N=10^6     333.082"
```

Analytical Solution of the integral

$$\int_0^{10} f(x^2)dx = \frac{x^3}{3}\Big|_0^{10} = \frac{10^3}{3} - \frac{0^3}{3} = \frac{1000}{3} = 333,333$$




```
## [1] "integral analitica  =  333.33333"

## [1] "integral computacional obtida com N=10^8     333.39329"

## [1] "uma diferença de  0.02344"
```

Example 2 If you have a definite integral of the form: $I = \int_1^2 e^{-x}dx$ you can approximate it using Monte Carlo by treating it as an expectation:

$$I = (2-1)\frac{1}{N}\sum_{n=1}^{N} e^{-x}$$

Simulation

$$\int_1^2 e^{-x}dx \approx (2-1)\frac{1}{N}\sum_{n=1}^{N} e^{-x}$$

```
# Utilizando a função de integração computacional construida anteriormente e a função nativa exp = e^x
set.seed(36)
integral_comp_2 = function(func,lim_inf=1, lim_sup=2, N=1000)
  {x = runif(N, lim_inf, lim_sup)
  fx = func(-x)
```

```r
  return(sum(fx)/N*(lim_sup-lim_inf))}


paste("integral computacional obtida com N=10^4    ",round(integral_comp_2(exp, N=10^4),5))
```

```
## [1] "integral computacional obtida com N=10^4     0.23209"
```

```r
paste("integral computacional obtida com N=10^5    ",round(integral_comp_2(exp, N=10^5),5))
```

```
## [1] "integral computacional obtida com N=10^5     0.23302"
```

```r
paste("integral computacional obtida com N=10^6    ",round(integral_comp_2(exp, N=10^6),5))
```

```
## [1] "integral computacional obtida com N=10^6     0.23253"
```

Resolução analitica: aplicando o método da substituição

$$u = -x$$

$$du = -dx \rightarrow -du = dx$$

$$\int e^u - du$$

$$\int -e^u du = -e^u \Big|_1^2$$

substituindo u por x

$$-e^{-x}\Big|_1^2 = -e^{-2} + e^{-1}$$

```
## [1] "integral analitica  =  0.2325442"

## [1] "integral computacional obtida com N=10^8  0.2325398"

## [1] "uma diferença de  4.36e-06"
```

Estimating a Hard to Solve Integral Miraculin—a protein naturally produced in a rare tropical fruit—can convert a sour taste into a sweet taste. Consequently, miraculin has the potential to be an alternative low-calorie sweetener. In Plant Science (May, 2010), a group of Japanese environmental engineers investigated the ability of a hybrid tomato plant to produce miraculin. For a particular generation of the tomato plant, the amount of miraculin produced (measured in micro-grams per gram of fresh weight) had a mean of 105.3 and a standard deviation of 8.0. Assume that is normally distributed.

Find the probability that the amount of miraculin produced for a batch of tomatos ranges from 100 micro-grams to 110 micro-grams.

$\mu = 105,3$ $\sigma = 8$

Para estimar através do método de Monte Carlo, a probabilidade de que a quantidade de miraculin produzida esteja contida entre 100 e 110 micro-gramas utiliza-se a aproximação: $\int_a^b f(x)dx \approx (b-a)\frac{1}{N}\sum_{n=1}^{N} f(x)$ na qual

- f(x) = Normal distribution formula = $f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2*\sigma^2}}$
- $\mu = 105,3$
- $\sigma = 8$
- a = 100
- b = 110

replacing the parameters and defining the limits

$$\int_{100}^{110} \frac{1}{8\sqrt{2\pi}} e^{-\frac{(x-105,3)^2}{2*8^2}} \approx (110-100)\frac{1}{N}\sum_{n=1}^{N} \frac{1}{8\sqrt{2\pi}} e^{-\frac{(x-105,3)^2}{2*8^2}}$$

Simulation

```
## [1] "The computational integration, for N = 10000, results in 0.46817126"
```

```
## [1] "The analytically integration = 0.46774061"
```

```
## [1] "A difference of 0.00043065"
```