A brief introduction to Monte Carlo methods

Contents

1 Monte Carlo methods **1**
1.1 Classical Monte Carlo integration 1
1.2 Importance sampling 2

1 Monte Carlo methods

Monte Carlo methods are a class of numerical methods that originated in physics research in the 1940s, see for example Metropolis and Ulam (1949). Today, Monte Carlo methods are well-established tools in many scientific fields, they are used in cases where the utilization of mathematical models are not formally tractable.

This section briefly introduces the elements of Monte Carlo methods necessary to understand their utilization to our estimation problem. Our objective is to describe the different steps that lead to the description of the particle filter.

We start by describing the basic principles of Monte Carlo integration, then those of importance sampling and finally those of sequential importance sampling.

The particle filter derives directly from sequential importance sampling. The particle filter (Gordon et al. (1993), Doucet et al. (2001), Arulampalam et al. (2002), Gustafsson (2010)) encompasses set of methods for implementing recursive Bayesian filtering by Monte Carlo approaches. The key idea of particle filter is to represent the posterior probability density function of interest by a set of random sample (the particles) with associate weights and to use those to compute estimates.

1.1 Classical Monte Carlo integration

Let us consider the generic problem of evaluating the following integral with f any function and p a probability density function:

$$I(f) = \mathbb{E}_{p(x)}[f(x)] = \int f(x)p(x)dx \tag{1}$$

For a number of classical densities, it is possible to calculate the integral 1 formally. How is it can be done in other cases? The solution can be to carry out the calculation numerically on a computer. The question then becomes: how to do it? It is possible, in some cases, to implement deterministic numerical integration methods based on a discretization of space (rectangular rule, trapezoidal rule, Romberg integration, etc). However, the calculation of the integral, 1, is of great complexity in large dimensions. Monte Carlo methods are a set of methods also based on numerical simulations allowing to calculate the probability distribution. But, unlike deterministic methods that use a fixed grid of the parameter space, Monte Carlo methods are based on an adapted grid for each form of probability density.

It is assumed in this section that we are able to simulate N independent and identically distributed samples (i.i.d) from the density $p(x)$. Each sample is called a *particle* and is denoted as $\tilde{x}^{(i)}$. The distribution $p(x)$ can then be approximated by the following equation:

$$P_N(x) = \frac{1}{N}\sum_{i=1}^{N} \delta_{\tilde{x}^{(i)}}(x) \tag{2}$$

with $\delta_{\tilde{x}^{(i)}}(x)$ the Dirac delta function at $\tilde{x}^{(i)}$.

The integral 1 can be approximated by:

$$I_N(f) = \int f(x)P_N(x)dx = \frac{1}{N}\sum_{i=1}^{N} f\left(\tilde{x}^{(i)}\right) \tag{3}$$

$I_N(f)$ is an estimator of $I(f)$. One can prove that this estimator of the integral is unbiased, i.e. the expectation of $I_N(f)$ is equal to the true value of $I(f)$. If, moreover, the a posterior variance of $f(x)$ is finite:

$$\mathbb{E}_{p(x)}[f(x)] - I(f)^2 = \sigma_f^2 < +\infty \tag{4}$$

The variance of the estimator is equal to:

$$\mathbb{V}[I_N(f)] = \frac{\sigma_f^2}{N} \tag{5}$$

The strong law of large numbers then allows us to affirm that $I_N(f)$ converges almost surely (with probability one) to $I(f)$ when N go to infinity:

$$P\left(\lim_{N\to+\infty} I_N(f)\right) = I(f)) = 1 \tag{6}$$

By applying the central limit theorem, it is shown that $\sqrt{N}[I_N(f) - I(f)]$ converges in law to $\mathcal{N}(0, \sigma_f^2)$ when N go to infinity. And $\frac{[I_N(f)-I(f)]}{\sqrt{v_N}}$ is approximately distributed as a $\mathcal{N}(0,1)$ variable with $v_N = \frac{1}{N^2}\sum_{i=1}^{N}\left[f(\tilde{x}^{(i)}) - I_N(f)\right]^2$ an estimation of the variance of the approximation.

Example 1.1. Let us consider the evaluation of the integral of a function f over an interval. $[a, b]$ whose boundaries are finite using the Monte Carlo method. The $\tilde{x}^{(i)}$ are randomly drawn particles on the interval $[a, b]$ with a uniform probability density on $[a, b]$. The approximation of the integral will be given by:

$$\int_a^b f(x)dx \approx \frac{b-a}{N}\sum_{i=0}^{N-1} f\left(\tilde{x}^{(i)}\right) \tag{7}$$

By using Monte Carlo approximation let us evaluate the integral: $\int_0^{\frac{\pi}{2}} \cos(x)dx$.

Remark 1.1. It is possible to integrate this function analytically, the result is $\int_0^{\frac{\pi}{2}} \cos(x)dx = 1$.

Listing 1: Monte Carlo evaluation of the integral $\int_0^{\frac{\pi}{2}} \cos(x)dx$

```
def integral_MC(nbParticle):
    x = uniform.rvs(loc = 0, scale = np.pi / 2, size = nbParticle)
    f = np.cos(x)
    return (np.sum(f) * (np.pi / 2) / nbParticle)
```

Listing 2: Monte Carlo evaluation of the integral $\int_0^{\frac{\pi}{2}} \cos(x)dx$, results

```
Number of particle =  100 The integral =  0.967679289527075
Number of particle =  100 The error =  0.03232071047292495
Number of particle =  1500 The integral =  1.001866941473194
Number of particle =  1500 The error =  0.0018669414731939948
```

1.2 Importance sampling

When it is possible, one wants to generate samples (particles) directly from $p(x)$ then estimate $I(f)$ by using the estimator 3. The method studied now is called importance sampling (the reader may refer to the following references for a more detailed description of the method (Geweke (1989); Glynn and Iglehart (1989); Robert and Casella (2004, 2010)). Suppose one can only generate samples from a density $\pi(x)$, which is similar to $p(x)$. The definition of a correct weighting of the samples set must always allow the utilization of the Monte Carlo estimation. The probability density function $\pi(x)$ is referred to as the importance density function or the proposal density function. The similarity between $p(x)$ and $\pi(x)$ can be expressed by the following condition:

$$p(x) > 0 \Rightarrow \pi(x) > 0 \qquad \forall x. \tag{8}$$

The previous condition 8 is necessary for the importance sampling theory can be used. If valid, any integral of the form 1 can be rewritten in the following form:

$$I(f) = \mathbb{E}_{p(x)}[f(x)] = \int_{\mathcal{X}} f(x)p(x)dx = \int_{\mathcal{X}} f(x)\frac{p(x)}{\pi(x)}\pi(x)dx \tag{9}$$

provided that $\frac{p(x)}{\pi(x)}$ is upper bounded. A Monte Carlo estimation of the integral $I(f)$ can be computed by generating N i.i.d. $\{\tilde{x}^{(i)}; i = 1, \cdots, N\}$ particles distributed according to $\tilde{x}^{(i)} \sim \pi(x)$ and evaluate the following weighted sum:

$$I(f) \approx I_N(f) = \frac{\sum_{i=1}^{N} f(\tilde{x}^{(i)})w(\tilde{x}^{(i)})}{N} \tag{10}$$

With

$$w(\tilde{x}^{(i)}) = \frac{p(\tilde{x}^{(i)})}{\pi(\tilde{x}^{(i)})} \tag{11}$$

are the unnormalized importance weights. In the case of the normalization factor of the density $p(x)$ is unknown, the importance weights need to be normalized. In this situation the integral 1 will be estimate by using:

$$I_N(f) = \frac{\frac{1}{N}\sum_{i=1}^{N} f(\tilde{x}^{(i)})w(\tilde{x}^{(i)})}{\frac{1}{N}\sum_{j=1}^{N} w(\tilde{x}^{(j)})} = \sum_{i=1}^{N} f(\tilde{x}^{(i)})\tilde{w}(\tilde{x}^{(i)}) \tag{12}$$

With the normalized importance weigths expressed as:

$$\tilde{w}(\tilde{x}^{(i)}) = \frac{w(\tilde{x}^{(i)})}{\sum_{j=1}^{N} w(\tilde{x}^{(j)})} \tag{13}$$

Example 1.2. Let us consider $X \sim \mathcal{N}(0, 1)$, what is the probability $P(X > 5) = \int_5^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} dx$? The value of $P(X > 5)$ is very small, one generates $\tilde{x}^{(i)} \sim \mathcal{N}(0, 1)$ for $i = 1 \cdots N$, the probability $P(X > 5)$ can be evaluated by Robert and Casella (2004); Glasserman and Juneja (2008); Robert and Casella (2010):

$$P(X > 5) \approx \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}_{\tilde{x}^{(i)}>5}$$

Listing 3: Monte Carlo evaluation of the value of $P(X > 5)$

```
nbSimulation = 1000000
Sum = 0
for i in range(1, nbSimulation + 1, 1):
    x = norm.rvs(loc = 0, scale = 1, size = 1)
    Sum = Sum + int(x > 5)
P = (1 / nbSimulation) * Sum
```

The values obtained for $nbSimulation = 10000, 100000, 1000000$ and 2000000 of the probability $P(X > 5)$ are zero. For the values $nbSimulation = 3000000$ the probability $P(X > 5)$ are equal to $\frac{1}{nbSimulation}$. This means that for this value of $nbSimulation$, the $\tilde{x}^{(i)} \sim \mathcal{N}(0, 1)$ simulations produce only one non-zero result. The problem is that one try to calculate the probability of a very rare event. the simulation implemented here needs a lot of iterations to get a reasonable value of $P(X > 5)$.

One try to improve these results with importance sampling method. Let us choose the exponential density with support limited to $[5, \infty[$ as the importance density. The importance density can be written as:

$$\pi(y) = e^{-y}\frac{1}{\int_5^\infty e^{-x}dx} = e^{-(y-5)}$$

By using equation 10 the estimator of the probability $P(X > 5)$ can be written:

$$P(X > 5) \approx \frac{1}{N}\sum_{i=1}^{N}\frac{p(\tilde{y}^{(i)})}{\pi(\tilde{y}^{(i)})} = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{\sqrt{2\pi}}e^{-\left(\frac{\left(\tilde{y}^{(i)}\right)^2}{2}-\tilde{y}^{(i)}+5\right)}$$

Listing 4: Evaluation of the value of $P(X > 5)$ with importance sampling method

```
nbSimulation = 1000
borne = 5
y = expon.rvs(loc = 0, scale = 1, size = nbSimulation) + borne
weight = norm.pdf(y, loc = 0, scale = 1) /
         expon.pdf(y - borne, loc =0, scale = 1)
n = np.arange(start = 1,stop = nbSimulation + 1)
P = np.cumsum(weight) / n
```

Listing 5: Evaluation of the value of $P(X > 5)$ with importance sampling method, results

```
Final value =  2.826852772417995e-07
True value =  2.87e-07
```

Example 1.3. We want to use the sequential importance sampling algorithm to obtain an approximation of the probability density $p(x) = 0.7\mathcal{N}(3,1) + 0.3\mathcal{N}(-2,1)$. To do that we will use the density of importance $\pi(x) = \mathcal{N}(0,9)$[1] and 30 particles.

Listing 6: Implementation example of importance sampling - Definitions needed

```
from scipy.stats import norm
import numpy as np
import matplotlib.pyplot as plt
# Probability density function we want to approximate
def p(x):
    norm1 = norm(loc = 3, scale = 1)
    norm2 = norm(loc = -2, scale = 1)
    return 0.7 * norm1.pdf(x = x) + 0.3 * norm2.pdf(x = x)
# Importance probability density function
importanceDensity = norm(loc = 0, scale = 3)
def piPdf(importanceDistribution, x):
    return importanceDistribution.pdf(x = x)
# Importance probability density function samples
def piSample(importanceDistribution, nbSample):
    return importanceDistribution.rvs(size = nbSample)
# Variables needed for the implementation
nbParticle = 30
xTilde = np.zeros(shape = nbParticle)
wTilde = np.zeros(shape = nbParticle)
wNormalizedTilde  = np.zeros(shape = nbParticle)
```

[1] With $\mathcal{N}(\mu, \sigma^2)$, the normal density with mean equal to μ and variance equal to σ^2

Listing 7: Implementation example of importance sampling - The algorithm

```
    for i in range(0, nbParticle, 1):
        xTilde[i] = piSample(
                    importanceDistribution = importanceDensity,
                    nbSample = 1)
        pTilde = p(x = xTilde[i])
        piTilde = piPdf(
                  importanceDistribution = importanceDensity,
                  x = xTilde[i])
        wTilde[i] = pTilde / piTilde
    wSum = np.sum(wTilde)
    for i in range(0, nbParticle, 1):
         wNormalizedTilde[i] = wTilde[i] / wSum
```

The result of the approximation of $p(x)$ is illustrated by the figure 1.

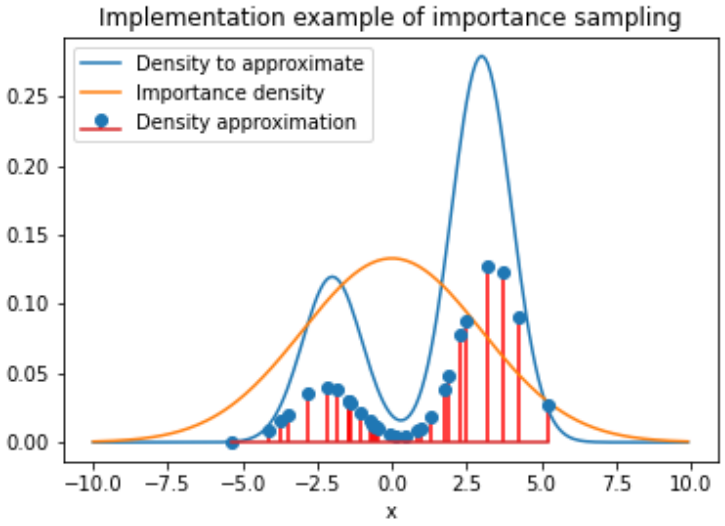

Figure 1: Approximation of the probability density $p(x) = 0.7\mathcal{N}(3,1) + 0.3\mathcal{N}(-2,1)$ with the density of importance $\pi(x) = \mathcal{N}(0,9)$ and 30 particles.

It is possible to show that there is a density of optimal importance, in the sense that there is a density that minimizes the variance of the estimator Robert and Casella (2004).

Theorem 1.1. The importance density that minimizes the variance of 10 is:

$$\pi_{opt}(x) = \frac{|f(x)|\, p(x)}{\int_{\mathcal{X}} |f(u)|\, p(u) du} \tag{14}$$

References

M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Transactions on Signal Processing*, 50 (2):174–188, Feb 2002. ISSN 1941-0476. doi: 10.1109/78.978374.

A. Doucet, N. de Freitas, and N. Gordon. *Sequential Monte Carlo Methods in Practice*. Statistics for Engineering and Information Science. Springer, 2001.

J. Geweke. Bayesian inference in econometric models using monte carlo integration. *Econometrica*, 57(6):1317–1339, 1989.

P. Glasserman and S. Juneja. Uniformly efficient importance sampling for the tail distribution of sums of random variables. *Mathematics of Operations Research*, 33(1):36–50, 2008.

P. W. Glynn and D. L. Iglehart. Importance sampling for stochastic simulations. *Management Science*, 35(11):1367–1392, 1989.

N. J. Gordon, D. J. Salmond, and A. F. M. Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. *IEE proceedings. Part F, Radar and signal processing*, 140(2):107–113, 1993. ISSN 0956-375X.

F. Gustafsson. Particle filter theory and practice with positioning applications. *IEEE aerospace and electronic systems magazine*, 25(7):53–82, 2010. ISSN 0885-8985.

N. Metropolis and S. Ulam. The monte carlo method. *Journal of the American Statistical Association*, 44(247):335–341, 1949. ISSN 0162-1459.

C. P. Robert and G. Casella. *Monte Carlo Statistical Methods*. Springer Texts in Statistics. Springer-Verlag, Berlin, Heidelberg, 2004.

C. P. Robert and G. Casella. *Introducing Monte Carlo Methods with R*. Springer-Verlag, Berlin, Heidelberg, 2010.