Numerical Integration

Monte Carlo Integration

Sachin Shanbhag

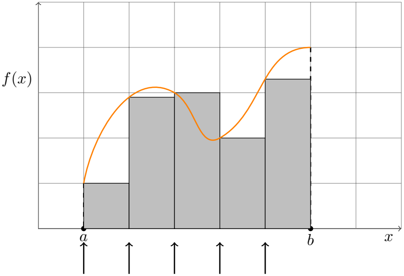
Department of Scientific Computing
Florida State University,
Tallahassee, FL 32306.

References

- S. Chapra and R. Canale, Numerical Methods for Engineers
- Carnahan and Wilkes, Applied Numerical Methods, University of Michigan, Class Notes, 1996.
- Pal, Numerical Analysis for Scientists and Engineers, 2007.
- Holistic Numerical Methods Institute webpage
- Samir Al-Amer, Class Notes
- wikipedia.org

Idea

In Newton-Cotes and Gaussian Quadrature



n grid points either uniformly separated or well-chosen to approximate the integral as

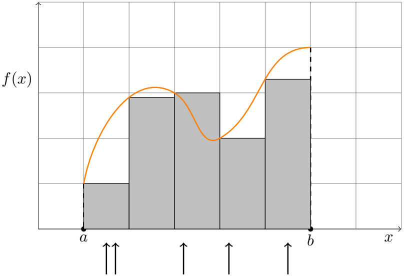
$$\int_a^b f(x)dx \approx \sum_{i=1}^{n} w_i f(x_i)$$

Idea

The simplest of these rules (rectangle rule)

$$\int_a^b f(x)dx \approx \sum_{i=1}^{n} f(x_i)\Delta x$$

In Monte Carlo, same formula but points selected randomly (from uniform RNG)



Motivation

Monte Carlo is a good integration technique when one of the following three features becomes prominent

- High dimensional integrals: Convergence rates and the size of the problem become overwhelming
- Complex Boundaries: Often with high-dimensional integrals.
- Noisy integrands: When the integrand is smooth, methods like Gauss quadrature work very well. Conversely, when the integrand is noisy, then the high-order accuracy of these methods (which is only valid for smooth functions) becomes less relevant.

Multidimensional Integrals

- So far, we have only considered 1-D integrals of the form

$$I = \int_a^b f(x)\ dx.$$

- Multidimensional integrals often arise naturally in many physical problems

$$I = \int_{a_1}^{b_1} \int_{a_2}^{b_2} \cdots \int_{a_m}^{b_m} f(x_1, x_2, ...x_m)\ dx_m ... dx_2 dx_1$$

  where the limits of the inner integrals may depend on the outer variables. For example,

$$I = \int_0^1 \int_{x_1}^{x_1^2} x_1 x_2\ dx_2 dx_1.$$

High dimensional integrals

- Consider a 10-D integral (far less uncommon then you would think) with simple boundaries

$$I = \int_0^1 \int_0^1 \cdots \int_0^1 f(x_1, x_2, ...x_{10})\ dx_{10}...dx_2 dx_1.$$

- Even if you considered 10 points along each dimension x_i, you would have to 10^{10} points to evaluate the function at.
- If your integrand required a millisecond to compute, then evaluating simply the function at all the grid-points would require more than 100 days.
- That is, the size of your problem can increase very quickly with dimensionality.

High dimensional integrals: Convergence

- For 1D, the error for multistep Simpson's rule scales with the number of grid points n as:

$$E \sim \frac{1}{n^4}$$

- For a similar problem, error vanishes more slowly for Monte Carlo

$$E \sim \frac{1}{\sqrt{n}}$$

- For 1D MC offers no advantage over quadrature.
- Things change as the dimensionality of the integral increases

High dimensional integrals: Convergence

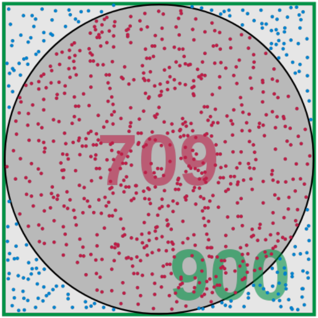
- For d dimensional integral

Simpson's	Monte Carlo
$E \sim \frac{1}{n^{4/d}}$	$E \sim \frac{1}{\sqrt{n}}$

- MC independent of dimension
- It usually starts becoming competitive for $d > 4$
- So both convergence and problem size make MC the default choice in areas where such high-dimensional integrals occur

Complex Boundaries

- As the dimensionality of a problem increases, the boundaries of the integral can become quite complicated
- Consider finding the area of a circle by throwing darts*



- Generate random number $(x, y) \in [-1, 1]^2$ and throw a dart at (x, y).

*wikipedia

Complex Boundaries

- If the dart lies inside the circle then color it “red”. This can be checked by testing whether $x^2 + y^2 < 1$.
- Otherwise color it “blue”
- The ratio of the red darts, N_r to the total number of darts thrown $(N_r + N_b)$ is the ratio of the areas of the circle and the square

$$\frac{A_c}{A_s} = \frac{N_r}{N_r + N_b}$$

- If we know $A_s = 2^2 = 4$, then, in this case,

$$A_c = 709/900 \times A_s = 3.1511.$$

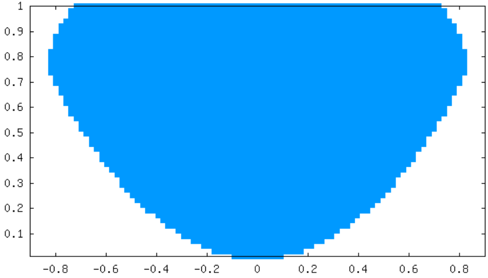
which is close to π as expected.

Complex Boundaries

Consider something slightly more complicated than a circle.

$$I = \iint\limits_{x^2<\sin(y),\, y^2<\cos(x)} e^{\sqrt{x^2+y^2}}\, dx\, dy$$

where the domain looks like:



Complex Boundaries

- In MC, we simply throw n darts $(x, y) \in [-1, 1]^2$
- If $x^2 < \sin(y)$, and $y^2 < \cos(x)$, we compute $e^{\sqrt{x^2+y^2}}$ and add it to a running total $S = S + e^{\sqrt{x^2+y^2}}$
- Otherwise, do nothing. This is equivalent to the operation $S = S + 0$.
- Since $\Delta A = \Delta x \times \Delta y = 4/n$, we evaluate the integral as

$$I \approx \sum_{i=1}^{n} f(x_i, y_i) \Delta A = \Delta A \sum_{i=1}^{n} f(x_i, y_i) = \frac{4}{n} S$$

- Convince yourself that we are using essentially the same algorithm as we did to determine the area of a circle.

Complex Boundaries

- Over there
$$I = \iint_{x^2+y^2<1} 1 \; dxdy$$
- Thus, $S = N_r$, and $n = N_r + N_b$.