Introduction to Monte Carlo Integration

Chem 280

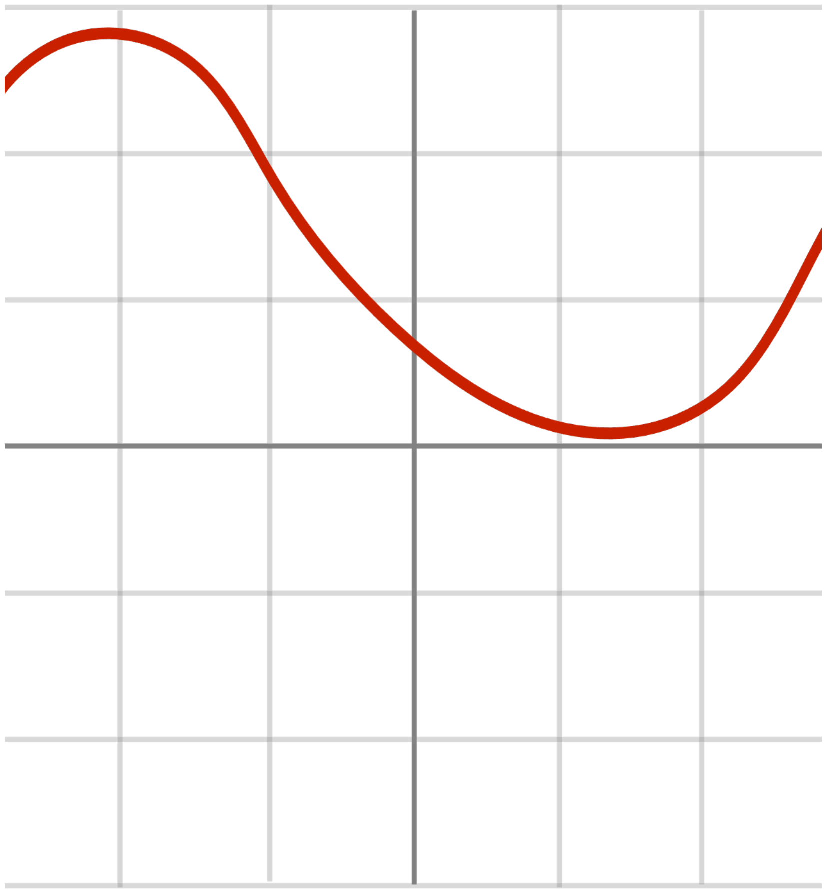
Berkeley
UNIVERSITY OF CALIFORNIA

College of Chemistry
College of Engineering

Monte Carlo Methods

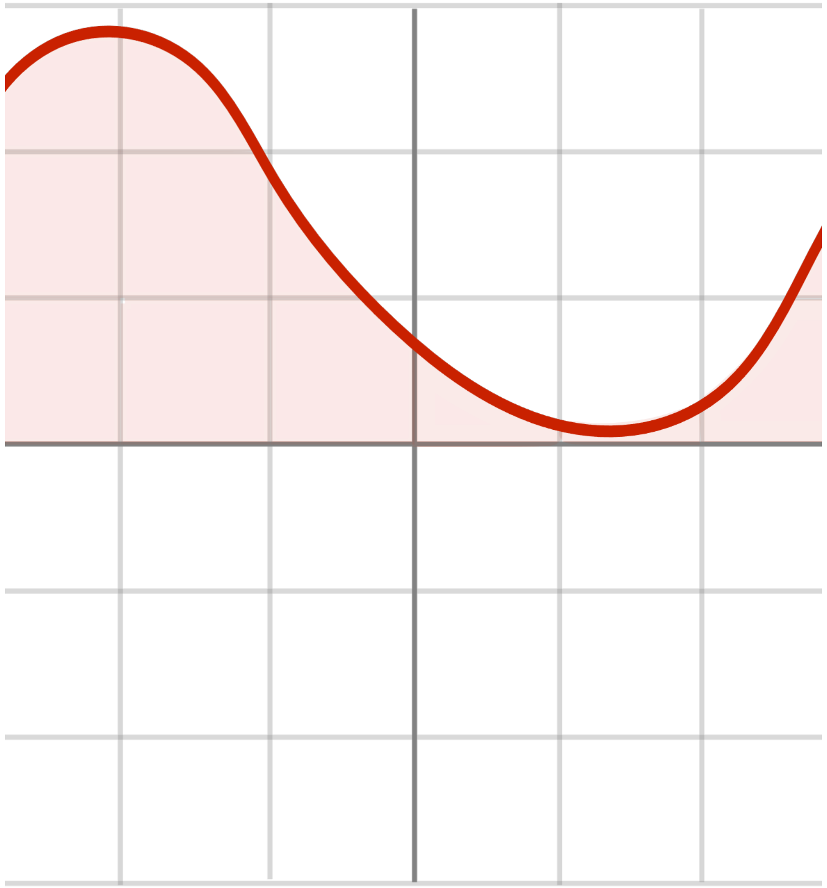
- Monte Carlo methods rely on the generation of **random numbers** to make numerical approximations.
- Can be used for problems where there is no analytical solution.
- Monte Carlo (MC) is used in many fields including molecular science, physics, and finance.
- Today, we will be using Monte Carlo methods to integrate the area under a curve

Integrating the area under a curve



$y = f(x)$

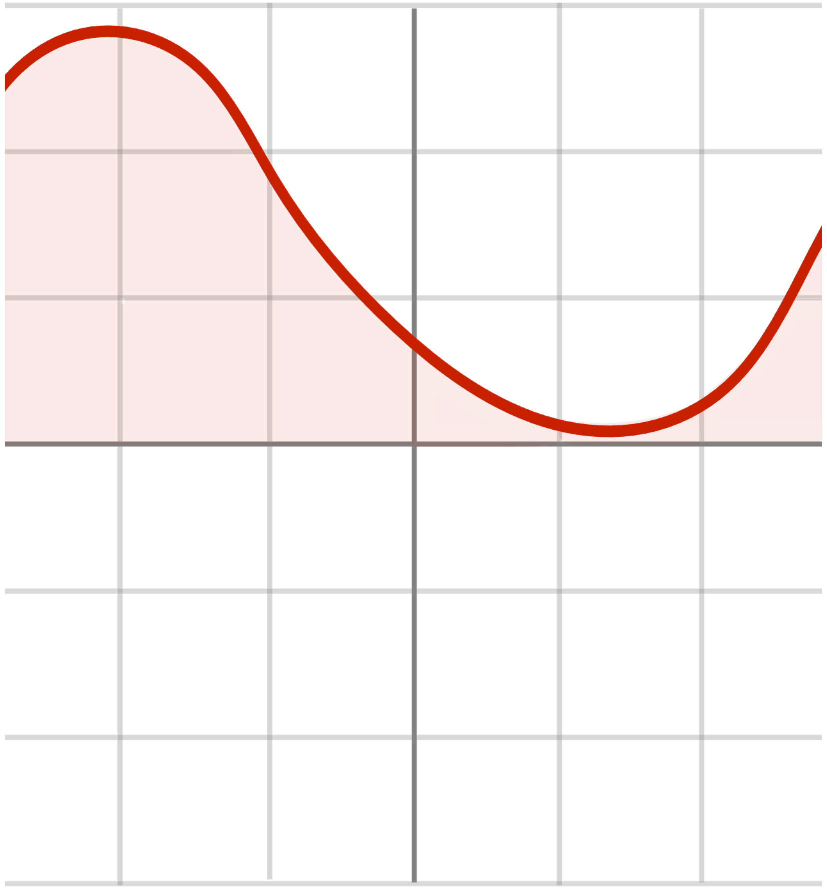
Integrating the area under a curve



$$y = f(x)$$

$$Area = \int f(x)dx$$

Integrating the area under a curve

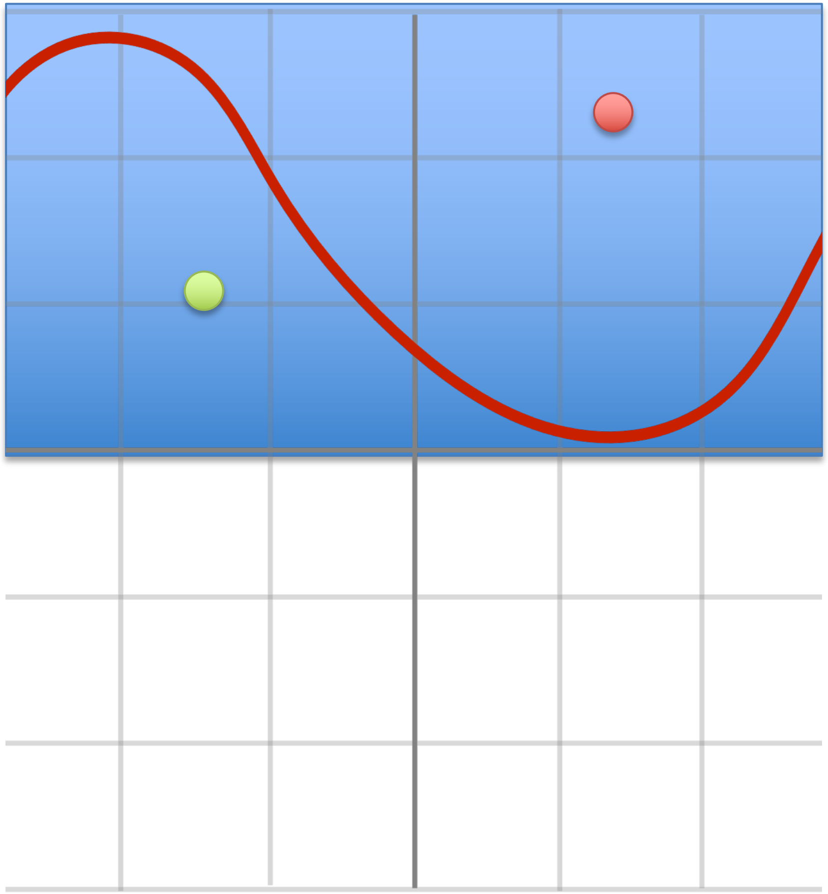


$$Area = \int f(x)dx$$

We could solve this example analytically. But what if our derivative were very complicated? - We would have to use a different method

Let’s consider how we could use **Monte Carlo** to evaluate this integral

Monte Carlo Integration



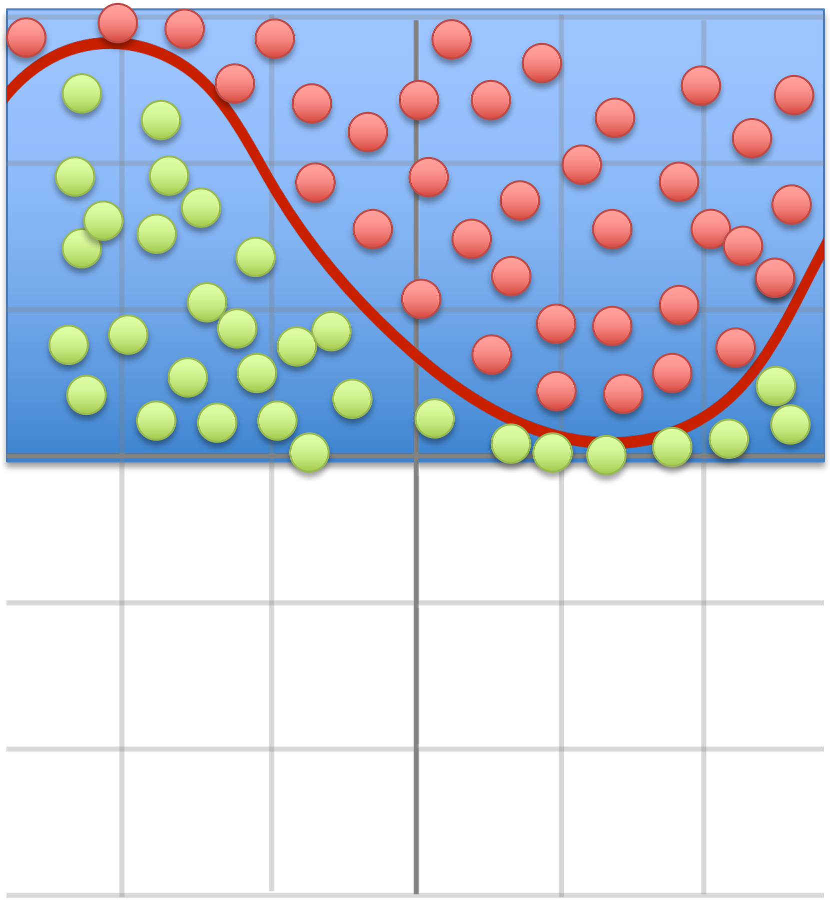
Imagine we are evaluating this integral on the range from x = -3 to x = 3, as highlighted in blue.

Procedure:

- Generate a set of uniformly distributed random points in this highlighted area.
  - Uniformly distributed means they are equally likely to occur anywhere in this box.
- Count the number of points that fall under the curve.
  - With a large number of points this will give you the ratio of area under the curve to total area,
- Multiply the area of consideration by the calculated ratio

Berkeley
UNIVERSITY OF CALIFORNIA

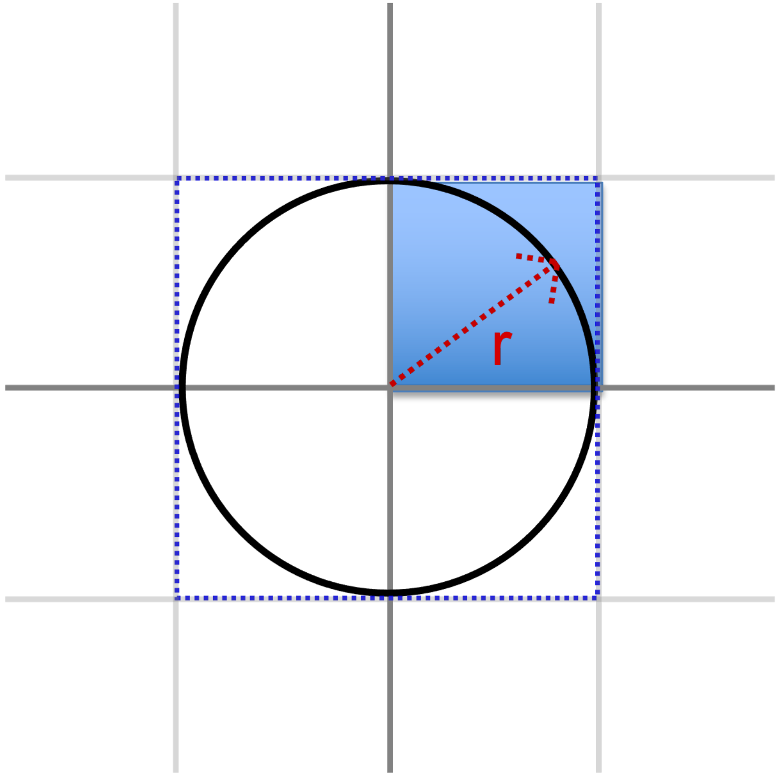
Monte Carlo Integration



$$A_{curve} = \frac{N_{inside}}{N_{inside} + N_{outside}} * A_{total}$$

$$A_{curve} = \frac{N_{inside}}{N_{total}} * A_{total}$$

Monte Carlo Estimation of π



Consider the area of a circle

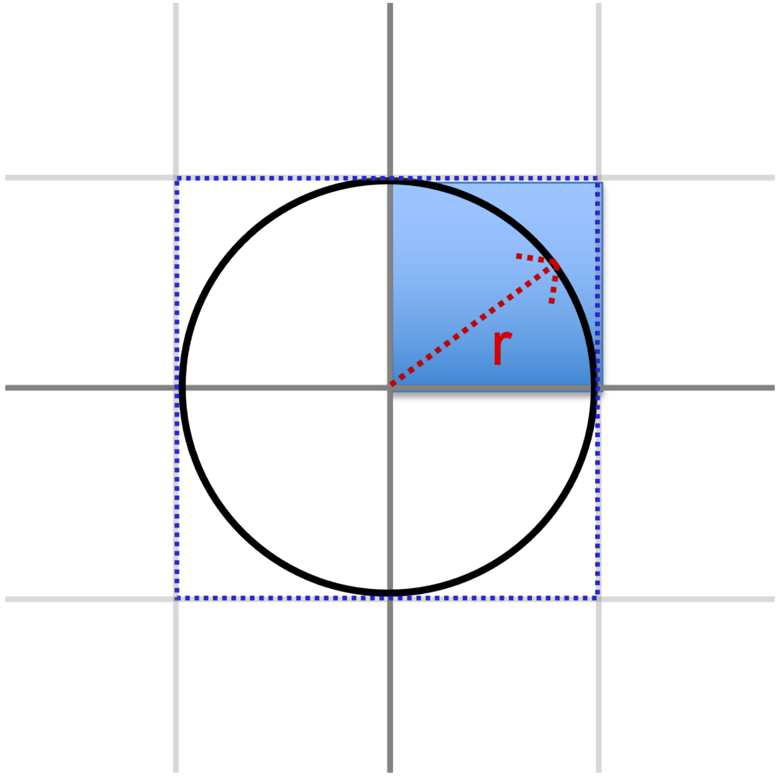
$$A = \pi r^2$$

For the unit circle, r = 1

$$A = \pi$$

Berkeley
UNIVERSITY OF CALIFORNIA

Monte Carlo Estimation of π in Python



We will use the **Python Standard Library** for our implementation.

The **Python Standard Library** is the set of modules that is distributed with Python. If you have Python, you will have these modules available to you.

Procedure:

1. Start with count inside circle = 0
2. Generate a a random point.
3. Determine if random point lies within the unit circle.
4. If point is inside circle, increase counter.
5. Repeat 2 – 4 as many times as desired.
6. Calculate ratio of points inside the circle to total number of points.