MAU11601:Introduction To Programming - Tutorial 1

Liam Burke
School Of Mathematics, Trinity College Dublin

December 5, 2022

Introduction

In this tutorial we you will learn some basic MATLAB skills by writing a simple program to estimate the value of π using Monte Carlo methods.

Some Background

Monte Carlo methods are a broad class of computational algorithms that obtain estimates of quantities of interest using random numbers.

A Monte Carlo Estimate of π

Consider a square of length 1 with the upper right quadrant of a circle or radius 1 embedded as shown in Figure 1. Suppose we generate N uniform pairs of random numbers $(x_i, y_i), i = 1, 2, .., N$ in the interval $(0, 1)$. For a large enough number of such points the ratio of the number of points contained within the circle N_{hit} and the total number of points used N, serves as good approximation to the area of this upper half quadrant of the circle. We have

$$\frac{N_{hit}}{N} \approx \frac{1}{4}\pi,$$

and so a good approximation for π is given by

$$\pi \approx 4\frac{N_{hit}}{N}$$

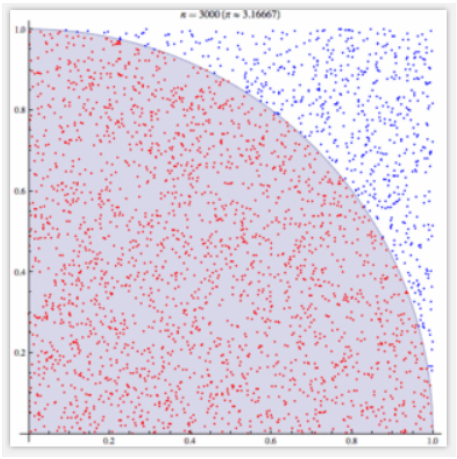

Figure 1: A square with length 1 embedded with the upper right quadrant of a circle of radius 1.

Exercise 1

Write a MATLAB program in a single script .m file which estimates a value of π using a chosen number of sample points N.

Some questions to consider when writing your program include:

- How can we ensure a generated point is contained within the circle? *hint: The equation of a circle of radius r is $x^2 + y^2 = r^2$*
- How do we count these points?

Exercise 2

Modify your code from Exercise 1 to estimate a value of π for the following numbers of sample points

$$N = 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 15, 20, 30, 50, 100, 120.$$

You must pre define a vector containing these numbers and loop over the vector appropriately.

Make a plot of the number of sample points used vs. the estimate obtained for π. Include a horizontal line on the y-axis to represent the exact value of π. This will serve as a means for observing the accuracy of your approximation.

Attempt to replicate a plot of your results using the same title, legend and axis labels as the plot shown in the following diagram.

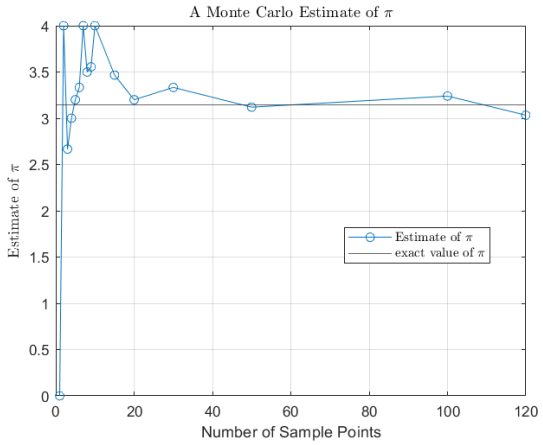

Figure 2: Sample results for exercise 2.

What do you notice about your plot as the number of sample points is increased?

Exercise 3

Modify your code from Exercise 2 by adding the main body of your code into a function

```
function [result_vec] = pi_estimate(N)
```

where N is the vector of sample points, and result_vec is a vector storing the estimate for each inputted value of N. Run your code using a script .m file which creates the vector N and calls the function.