EE1103: Numerical Methods

Week 2 - Random Number Generation

Due: 26 February 2022

1 Problem 1

To estimate π by playing darts game. [1]

In order to emulate throwing darts and estimating π,

- Generate random samples x, y where x and y are uniformly distributed in $[0, 1)$ and (x, y) denote a dart's location on the square grid. Use rand() for generating random numbers and invoke srand(3141592653) [2] as the seed for the random number generator to compare results among your teammates by running your code on onlinegdb.
- Imagine a circular arc constructed using the equation

$$x^2 + y^2 \leq 1 \tag{1.1}$$

as shown in Fig 1.

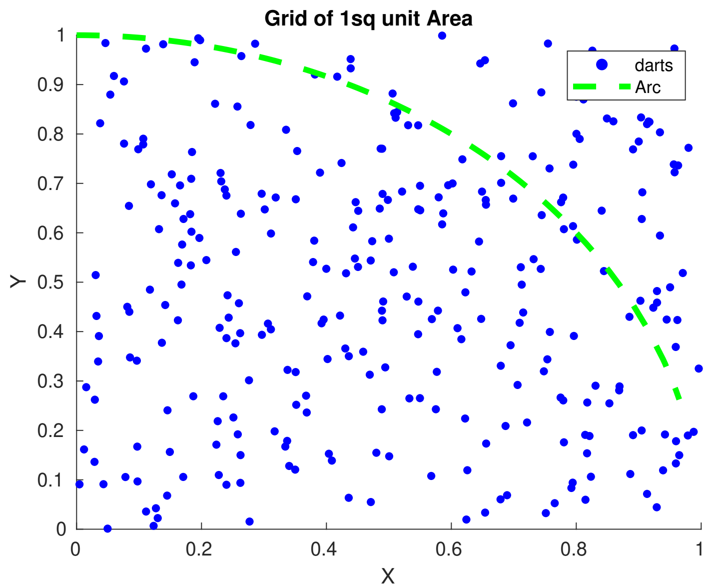

Figure 1: Points on the grid denoting landed darts ($n = 300$)

- Iterate the sample generation to simulate the n throws of the dart uniformly in the unit square.
- Let n_{arc} denote the number of darts that land inside the arc. Since the darts land uniformly, the ratio of number of samples inside the arc to those in the square will be proportional to their respective areas. π can be estimated by the following equation

$$\frac{n_{\text{arc}}}{n} \approx \frac{\text{Area of Quarter}}{\text{Area of Square}} = \frac{\pi}{4}. \tag{1.2}$$

Let $\hat{\pi}_n$ denote the estimate of π using n throws of the dart. We have

$$\hat{\pi}_n = 4 * \frac{n_{\text{arc}}}{n}$$

1 (a)

(i) Estimate π using $\hat{\pi}_n$ for $n = 10^2, 10^3, ..., 10^5$. Test out the maximum value of n, you are able to reach without any errors and tabulate your results.

(ii) Plot $\hat{\pi}_n$ as a function of n ($\log_{10}$ scale)

1 (b)

(i) Compute the absolute error $\% = \frac{|\hat{\pi}_n - \pi|}{\pi} * 100$ and tabulate along with (a).

(ii) Plot the absolute error % as a function of n ($\log_{10}$ scale)

2 Problem 2

Generate samples from a *standard normal distribution* from uniform random samples, using the *Box-Muller transform.*

- Generate two sets of uniform distribution (say U_1 and U_2) in $[0, 1)$.
- Obtain a standard exponential random variable from U_1

$$E = -\log(U_1). \tag{2.1}$$

- Generate the two sets of standard normal variables using the below transform :

$$X = \sqrt{E}\cos(2\pi U_2) \tag{2.2}$$
$$Y = \sqrt{E}\sin(2\pi U_2) \tag{2.3}$$

- X and Y will be independent zero-mean unit variance Gaussian random variables (or standard Normal random variables).

2 (a)

Generate $n = 100, 1000, 10000$ samples of X defined above, use srand(0) as the seed for comparison of outputs. (onlinegdb)

(i) Plot histograms for the samples generated.

2 (b)

(i) Plot the Empirical Distribution Function(EDF) of X (to approximate the Cumulative Distribution Function) for values of $n = 10, 100, 1000$. You can choose the x-axis range to be $[-4, 4]$, since most of the probability mass of the standard normal lies in this interval.

(ii) Compute the empirical estimate for Erf(1) and Erf(2) for $n = 10000$ samples, as the fraction of the generated samples that are less than or equal to 1 (and 2 respectively). Equivalently, these are just the EDF values at 1 and 2 respectively.

You can use the following procedure for obtaining the EDF plot.

- Export the normal distribution values (from **a**) to Google sheets/ Microsoft excel
- Sort the data in ascending order
- Create a column for EDF and fill the column with increment of $\frac{1}{n}$ for every row against the sorted normal random variables.
- Plot EDF along the vertical axis and sorted values of X along the horizontal axis
- If you are using Google Sheets for plotting, you can use line plot for (b), since there is no step plot option available.

References

[1] Dianna Cowern, Derek Muller. Calculating Pi with Darts, 03 2015. URL https://www.youtube.com/watch?v=M34TO71SKGk.

[2] GeeksforGeeks. rand() and srand() in C/C++, 11 2020. URL https://www.geeksforgeeks.org/rand-and-srand-in-ccpp/.