Computational Physics; January–May 2020
Assignment 4
Due: Monday, 25 May 2020

Instructions

- When you are asked to write a code, submit your code by posting it to Github and sending in the Github link.
- When you are asked to solve something manually, or when you are asked for a number or a plot as an answer, submit your response by email.

---

1. Write a linear congruential random number generator in Python. Use it to obtain 10,000 uniformly distributed random numbers between 0 and 1. Make a density histogram of your numbers and compare it to the uniform PDF.

2. Write a Python code to obtain 10,000 uniformly distributed random numbers between 0 and 1 using the library function `np.random.rand()`. Make a density histogram of your numbers and compare it to the uniform PDF.

3. How long did each of your codes above take to produce 10,000 uniform deviates?

4. Use the Transformation method in a C code to produce 10,000 random numbers distributed according to an exponential distribution with mean 0.5. Make a density histogram of your numbers and compare it to the exponential PDF.

5. Use the Box-Muller method in a Python code to produce 10,000 random numbers distributed according to a Gaussian distribution with mean 0 and variance 1. Make a density histogram of your numbers and compare it to the Gaussian PDF.

6. Use the Rejection Method in a Python code to produce random numbers distributed according to the distribution

$$f(x) = \sqrt{\frac{2}{\pi}} e^{-x^2/2} \qquad (x \geq 0).$$

Make a density histogram of your numbers and compare it to the distribution.

7. Consider the simulation of two dice that we discussed in class. Each dice yields an integer 1, 2, 3, 4, 5, 6 with equal probability. Suppose we got these counts in two runs of our simulation:

Score	2	3	4	5	6	7	8	9	10	11	12
Observed counts 1	4	10	10	13	20	18	18	11	13	14	13
Observed counts 2	3	7	11	15	19	24	21	17	13	9	5

Apply the χ^2 test and label the random numbers as “not sufficiently random”, “suspect”, “almost suspect”, or “sufficiently random”.

8. The area of a circle with unit radius is given by

$$I = \int \int_{-1}^{1} f(x, y) dx dy,$$

where

$$f(x) \begin{cases} 1 & \text{if } x^2 + y^2 \leq 1 \\ 0, & \text{otherwise,} \end{cases}$$

Calculate this area using Monte Carlo integration. Then calculate the volume of a ten-dimensional unit sphere.

9. Use the Metropolis algorithm to get a sample from a density that is uniform for $3 < x < 7$ and zero elsewhere. Write your own code for the Metropolis method; don't use a library function. Make a plot showing your Markov Chain. Make a density histogram of your numbers and compare it to the distribution.

10. Consider the data that we discussed in Lecture 25. These data are listed in the file `http://theory.tifr.res.in/~kulkarni/data.txt`. Use the `emcee` library to fit the model

$$y = ax^2 + bx + c$$

to these data by performing Bayesian probabilistic inference using MCMC. What are the best-fit values for the parameters a, b, and c, where "best-fit" is defined as the median of the posterior PDF? What are the one-sigma uncertainties on these values? Use a Gaussian likelihood and uniform priors in your analysis. Use 50 Markov chains and 4,000 steps. Make a plot showing all your chains. Use the `corner` library to make a plot showing the joint and marginalised posterior PDFs for the three model parameters. Make a plot showing the data with the best-fit model and 200 models randomly chosen from the posterior. (Consult the documentation websites for `emcee` and `corner`, or the recording of our Lecture 25, to figure out how to use these libraries.)