Numerical Methods in Scientific Computing, spring 2023

Problem sheet 9: random numbers

Return code and solutions by Monday 27.3.2023 23:59 to Moodle.

Bonus exercises are voluntary: they don't count towards the maximum for the exercises. They do, however, give you extra points in the grading of the course – it's possible to score over 100% of the points on the course with bonus exercises. No solutions will be provided for the bonus exercises.

Hint: these exercises are easy to do in Octave/Matlab (or with NumPy).

Note: notation in problem 2 and the bonus problem has been clarified.

1. (computer, 3 points) The simplest test for a random number x is verifying that it is uniformly distributed in $x \in [0, 1]$, which can be done e.g. by computing the histogram and plotting it.

   The mathematical version of checking the uniformity of the distribution is to calculate the chi square metric:

   $$\chi^2 = \sum_{i=1}^{M} \frac{(y_i - E_i)^2}{E_i} \tag{1}$$

   where y_i is the number of samples in the i:th bin, and E_i is the expected number found in bin i, which is simply

   $$E_i = \frac{N}{M} \tag{2}$$

   where M is the number of bins and N is the total number of random numbers. Statisticians tell us that if the x_i are really uniformly distributed random numbers, then the χ^2 value of equation (1) should have a mean of $M - 1$. (Note that this gives the expected $\chi^2 = 0$ for the trivial case of a single bin $M = 1$)

   Verify this numerically: write a function `chi2(N,M)` that generates N uniformly distributed random numbers (you can use a library random number generator), creates a histogram $\{y_i\}_{i=1}^{M}$ with M bins, and returns the chi squared value computed from the histogram using equation (1).

   Next, study $N = 100$ random numbers binned into $M = 10$ bins, and collect statistics over 50 000 runs; then plot a histogram of the χ^2 values and report the mean of the observed χ^2 values.

2. (computer, 3 points) Verify that the inverse cumulative distribution functions $F^{-1}(y)$ for the exponential and Lorentz distributions given in the lecture notes work by generating 10^6 uniformly distributed random numbers y, and plotting the histograms for $F^{-1}(y)$ against the respective, expected probability density functions $f(x)$ ($F(x)$ is the corresponding cumulative distribution function). Start with a uniform random number generator.

   You don't need to return code, but you should describe your method and results in detail.

3. (computer) Demonstrate that the quotient $x = y_1/y_2$ of two Gaussian random numbers y_1 and y_2 produces a Lorentz distribution:

   (a) (3 points) First, write a function `gaussrng(N)` that uses the Box–Muller algorithm from the lecture notes to generate N Gaussian random numbers. You can use a standard library function to obtain the needed uniformly distributed random numbers.

   (b) (3 points) Now, using the function you just wrote, generate $N = 10^6$ points for the ratio x, and bin the data in a histogram in $x \in [-10, 10]$. Plot the histogram and the Lorentz distribution in the same figure.

   Hint: Normalize the Lorentz distribution and the histogram data to the same scale, e.g. to have the value 1 at the peak at $x = 0$.

4. (computer, 6 points) Another simple smoke test for random number generators is the moment test:

$$\langle x^k \rangle = \int_0^1 x^k p(x) dx = \frac{1}{k+1},$$

since a uniformly distributed random number in $x \in [0,1]$ should have $p(x) = 1$. Study the accuracy of estimates for $\langle x^k \rangle$ as

$$\langle x^k \rangle \approx \frac{1}{N} \sum_{i=1}^{N} x_i^k.$$

Run 100 independent calculations of $\langle x \rangle$ using $N = 1000$ points, and plot the results as a function of number of the trial. Repeat the calculation with $N = 10^5$ points, and plot this result in the same figure. You can use a standard library function to generate the uniformly distributed random numbers.

Next, repeat the analysis for $\langle x^2 \rangle$.

You don't need to return any code, just give a detailed description of your results.

5. (computer) The hit-and-miss method is one way to estimate the value of π. Here, points (x, y) are generated at random in the square $x, y \in [-1, 1]$. If the point falls within a unit circle $x^2 + y^2 \leq 1$, the trial is accepted; otherwise, the trial is rejected. *Note that an explicit expression for the boundary is not needed—we only need to be able to determine if a given point is inside the boundary or not!*

   It is easy to see that the fraction f of accepted trials is given by the ratio of the area of the circle to the area of the square

   $$f = \frac{A_{\text{circle}}}{A_{\text{square}}} = \frac{\pi r^2}{(2r)^2} = \frac{\pi}{4}.$$

   This means that one can estimate the value of π from the fraction of accepted trials as

   $$\pi = 4f \approx \frac{4N_{\text{accepted}}}{N_{\text{accepted}} + N_{\text{rejected}}}. \tag{3}$$

   (a) (2 points) Write a function `mypi(N)` which returns an approximate value for π using equation (3). Again, you can use a standard library function to generate the uniformly distributed random numbers.

   (b) (2 points) Next, study the approximate value of π you get with $N = 10^n$ trials with $n = 2, 3, 4$. For each N, form 1000 different estimates for the value of π to form a histogram, and plot the histograms corresponding to the different values of N in the same figure. Mark also the accurate value of pi

   $$\pi \approx 3.14159265358979323846264338327950288419716939937510582097494459230781640628620899862803483$$

   in the figure. What do you see?

   (c) (2 points) Extend the shooting method for π from two to three dimensions, *i.e.* from a circle to a sphere. Repeat the calculations with the same number of trials and again plot histograms. Compare the accuracy of the two methods. Which variant is more accurate? Can you explain why?

   Note that the volume of a n-dimensional hypersphere with radius R is given by

   $$V_n(R) = \frac{\pi^{n/2}}{\Gamma\left(\frac{n}{2} + 1\right)} R^n,$$

   where $\Gamma(x)$ is the Euler gamma function that satisfies

   $$\Gamma(x+1) = x\Gamma(x),$$
   $$\Gamma(1) = 1,$$
   $$\Gamma\left(\frac{1}{2}\right) = \sqrt{\pi}.$$

   Based on this expression, what can you say about the efficiency in even higher dimensions than $n = 3$?

6. (BONUS, computer, 6 points) Study the determination of Gaussian random numbers using importance sampling with an exponential function, $f(x) \propto \exp(-\gamma x)$, $\gamma > 0$. As discussed in the lecture notes, it is enough to study the case $f(x) \propto \exp(-x^2/2)$, since transforming these numbers to $\sigma \neq 1$ and $\mu \neq 0$ is trivial. Start out by determining an optimal value for γ and the K parameter. Study the performance of random number generation using

   (a) Gaussian inverse transform sampling, i.e. getting $y = F^{-1}(x)$ from the Gaussian $f(x)$

   (b) Importance sampling: generating random numbers that follow an exponential distribution and then using the rejection method to get the numbers to follow a normal distribution

   (c) the polar Box–Muller algorithm

   (d) the Marsaglia polar method

   What is the relative performance of the methods?