1st problem set for Computer Simulations, 2024

Hand-in: week of April 8 (schedule A), week of April 15 (schedule B)
Please upload/submit your program files in the TeachCenter *before* the conversation with the tutor.

Drawing random numbers from a given distribution

1. Histograms and their fluctuations: In this first simple problem we will examine histograms themselves. The description is detailed in order to make the task straightforward. We use uniformly distributed random numbers for simplicity. Generate them with a library routine.

   (a) Generate N uniformly distributed random numbers (*rand(N,1)* in Matlab). Plot a histogram, normalized as a pdf. This means that for each bar, the count of each bin has to be normalized by $N \cdot b$, where b is the bin width. (In Matlab, you can use *histogram* with appropriate parameters.) Vary N and look at the fluctuations of the bar heights. Do the fluctuations have about the expected size? You will have to zoom into the plot.

   (b) We examine the fluctuations of the height of an individual bar. Generate $L = 1000$ samples of N random numbers each. For each sample, calculate (but do not plot) a histogram, normalized for a probability density function. Use just the height of the bar corresponding to random numbers $0.5 < x < 0.6$ (i.e. the sixth of 10 bars). What is the expectation value of this height ? Collect the L heights and plot a histogram of these values. How should the width of this histogram depend on N ? Verify (approximately) the expected behavior with a few examples, e.g. with $N = 10^m$, m=3 and 5 (m between 2 and 6 should work well). In Matlab, the commands *histcounts* and the very similar *histogram* make this easy.

   (c) Now we add error bars to the histograms of part (a). Again generate a histogram of N uniform random numbers, normalized as a pdf. Estimate the size of the error of a given count N_i from the frequentist expression (see lecture notes). Draw the histogram with error bars. (In Matlab, error bars need to be added with the routine *errorbar*, which needs the x-coordinates of the bin-centers, the heights of the bars, and the (normalized) size of the errors as input. These numbers can be obtained like *h=histogram(...); w=h.BinWidth* etc.) Vary N and verify, using a few examples, that the size of the error-bars correspond to the fluctuations in the histograms.

2. Inverse transformation, Cauchy distribution:

   Write a program which generates random numbers according to the Cauchy distribution

   $$h(x) = \frac{1}{\pi} \frac{1}{1 + x^2}$$

   by using the inverse transformation method. You first need to find the corresponding cumulative distribution function. You can use a library function for generating uniform random numbers.

3. Rejection method:

   Write a program that generates random numbers from the normalized probability distribution

   $$g(x) = \frac{1}{z}\frac{1}{\pi}\frac{\sin^2(x)}{1+x^2}$$

   with z a normalizing constant (here $z = (1-e^{-2})/2$).

   (a) Use the Cauchy distribution as an enveloping function. You then need a constant c such that

   $$g(x) \leq c\, h(x), \qquad \forall x \in (-\infty, \infty).$$

   Any solution to this equation is correct, but a reasonably small value of c will produce a more efficient generator.

   In the present case, we already know an enveloping function, since $g(x) \leq \frac{1}{z}\, h(x)$. Thus $c = \frac{1}{z}$ is an obvious solution. In the acceptance condition $r\, c\, h(x_T) \leq g(x_T)$, the variable z then cancels, thus the normalizing constant z of $g(x)$ does not need to be known explicitely for the rejection method here.

   (b) Alternate envelope: Choose an alternate enveloping function for the rejection method, e.g. constant for $|x| < x_0$ and decreasing like $1/x^2$ for $|x| \geq x_0$, with a suitably chosen value for x_0. Depending on your choice of enveloping function, you may need to choose further constants. Note that your parameters do *not* need to be optimal. The envelope can be noticeably larger than the desired distribution and still be quite good.

4. Analysis of histograms and statistical deviations for the generators

   Generate random numbers with all three generators. When you generate random numbers with the two rejection methods, are the acceptance rates reasonable?

   For all generators, perform a frequentist analysis to calculate histograms with error bars. For the first generator (Cauchy distribution), also perform a Bayesian analysis (see lecture notes) to calculate the histogram and error estimates.

   In all cases, plot the histograms together with the error bars and the desired pdf. Are the sizes of the "error bars" reasonable? What size should you (roughly) expect? Do the sizes scale correctly with the number N of entries in the histogram? Are the histograms compatible with the desired distributions, given the uncertainties?

   When do the differences between the frequentist and the Bayesian analysis become visible ?

   Note that you need to normalize the histograms and the desired pdf. In the present case you also need to restrict them to a finite interval of the x-axis. You can either cut $|x|$ off at some x_{max} and normalize both the histogram and the desired distribution inside the remaining interval, or you might combine values $|x|$ larger than x_{max} into one or two histogram bars.