Scientific Computing, Fall 2020
Assignment VI: Monte Carlo Methods

Aleksandar Donev
Courant Institute, NYU, *donev@courant.nyu.edu*

Posted Nov 19th, 2020
Due by **Sunday Dec 6th**, 2020

For grading purposes the maximum is considered to be 50 points.

1 [55pts] Sampling Random Numbers

We consider here Monte Carlo calculations based on the one-dimensional probability density function

$$f(t) = te^{-t}.$$

The mean of this distribution is 2 and the variance is also 2. What we need is a method to sample i.i.d. variables from this distribution.

Recall from the lecture notes that sampling from the exponential distribution $f_E(t) = \lambda e^{-\lambda t}$ is simple to do using the inversion method. For this homework, you will need to implement a routine for sampling random numbers from the distribution $f_E(t)$, for a given λ.

1.1 [30pts] Histogram validation

[20pts] Write a MATLAB function that makes a histogram of a probability distribution $f(x)$ by generating a large number (n) of i.i.d. samples and counting how many (n_i) of them fell in a bin i of width Δx centered at x_i,

$$\hat{f}(x_i)\Delta x = \frac{n_i}{n} \approx f(x_i)\Delta x.$$

This function should take as arguments the *sampler* of $f(x)$, the number of bins used in the histogram, the number of random samples, and the interval $[x_{\min}, x_{\max}]$ over which the histogram is computed (throw away samples outside of this interval, i.e., do not count them toward n_i but do count them toward n). *Hint: This is best done by having one of the arguments of the histogram routine be a sampler of $f(x)$, which means a function handle for a function that returns a random number sampled from $f(x)$, rather than trying to pass $f(x)$ itself. Test your function by passing it one of the built-in samplers, for exampl*e, choose *$f(x)$ to be the standard Gaussian distribution, i.e., sampler = @() randn().*

In addition to just computing the empirical (numerical) distribution $\hat{f}(x) \approx f(x)$, return also estimates of the uncertainty in the answer, i.e., the uncertainty in the height of each bin in the histogram.
[*Hint: Following the lecture notes, the variance $\sigma^2(n_i)$ of the number of samples that end up in a given bin, is $\sigma^2(n_i) \approx \bar{n}_i$. Since you do not know the mean you can approximate it as $\bar{n}_i \approx n_i$.*]
[*Hint: The MATLAB function errorbar makes plots with error bars.*]

[10pts] Test your routine for sampling the exponential distribution f_E (set $\lambda = 1$, for example) by comparing the empirical histogram to the theoretical distribution function. The lecture notes showed how to sample from the exponential distribution using the inversion method.

1.2 [25pts] Simple sampler

[20pts] It turns out that one can generate a sample from $f(t)$ by simply *adding* two independent random variables, each of which is exponentially-distributed with density e^{-t}. Implement a random sampler using this trick and generate 10^4 i.i.d. samples from $f(t)$, and verify that the empirical mean and variance are in agreement with the theoretical values (10 pts). For the mean, report an error bar (give the formulas used and explain how you computed the numbers that enter in the formula) and make sure the empirical result is inside a reasonable confidence interval (e.g., two standard deviations away) around the theory (10pts) [*Hint: If you report a 95% confidence interval running the code 100 times should give ˜5 failures.*].

[5 pts] Validate your sampler by using the histogram routine from part 1.1, using 10^5 samples and 100 bins in the interval $0 \leq t \leq 10$.
[*Hint: You can eliminate any values of* $x > 10$.]