Scientific Computing, Spring 2012
Assignment V: Monte Carlo

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

Posted April 11th, 2012
Due: **Sunday April 22nd**, 2012

A total of 75 points is possible.

1 [75pts] Monte Carlo in One Dimension

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.

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 [10pts] Histogram validation

[7.5pts] 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,

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

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.
Hint: This is best done by having one of the arguments of the historgram 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 example, 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 historgram.
[*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 $\bar{n}_i \approx n_i$.*]

[2.5pts] 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. [*Hint: The MATLAB function errorbar makes plots with error bars.*]

1.2 [10pts] Simple sampler

[7.5pts] 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. For the mean, report an error bar and make sure the empirical result is inside a reasonable confidence interval (e.g., two standard deviations away) around the theory.

[*Hint: To verify that your code gives the right answer, it is a good idea to test it on some known distribution, for example, the uniform or normal distributions, for which MATLAB has built-in samplers.*]

[2.5 pts] Validate your sampler by using the histogram routine from part 1.1, using 10^5 samples and 100 bins in the interval $0 \leq t \leq 10$.

1.3 [25pts] Rejection Sampler

[10pts] Implement a rejection sampler for $f(t)$ based on accepting/rejecting samples from the exponential distribution $g(t) \equiv f_E(t)$ with $\lambda = 1/2$. Explain what envelope function $\tilde{g}(t)$ you used and how you determined a suitable normalization factor Z. Do a quick test of your sampler by verifying that the empirical mean and variance are correct, as in part 1.2 above.

[2.5 pts] Validate your sampler by using the histogram routine from part 1.1, using 10^5 samples and 100 bins in the interval $0 \leq t \leq 10$.

[5pts] Estimate empirically what fraction of the trials are accepted for your sampler, for $\lambda = 1/2$.

[7.5pts] Estimate the optimal λ, for which the acceptance ratio is largest, theoretically or empirically, or both.

1.4 [30pts] Monte Carlo Integration

Implement a Monte Carlo procedure for computing the value of the integral

$$J = \int_{t=0}^{\infty} t^2 e^{-t} dt = 2.$$

For this, you will need to use random samples from some importance function $g(t)$. Try the following importance functions:

1. The simple exponential distribution $g(t) = e^{-t}$.
2. The distribution function $g(t) = te^{-t}$, which you can sample using the method developed in part 1.2 or 1.3.
3. The positive part of the normal distribution function,

$$g(t) = \sqrt{\frac{2}{\pi}} e^{-\frac{t^2}{2}},$$

which you can sample in MATLAB using $t = abs(randn())$. [*Hint: This is a trick question.*]

[15pts, 5pts for each importance function] For each importance function, report the 95% (two standard deviations) confidence intervals for the value of the integral using $N = 10^2, 10^3, 10^4$ and 10^5 samples, based on empirically measuring the variance. [*Hint: The theoretical answer $J = 2$ should be inside this interval most of the time.*]

[5pts] Compare the empirical estimates of the variance to the theoretical prediction for the variance of the Monte Carlo estimator given in class. *Hint:*

$$\int_{t=0}^{\infty} (t^2 - 2)^2 e^{-t} dt = 20.$$

[10pts] If you use the exponential distribution $g(t) = \lambda e^{-\lambda t}$ as an importance function, it may be possible to further reduce the variance by choosing $\lambda \neq 1$. Can you find the λ that minimizes the variance? [*Hint: You can do this empirically or analytically, but note that the analytical calculation is not trivial.*]