Problem Set 3: Evaluation of Functions, and Random Numbers
Computational Physics
Physics 480/680

James Sethna; Due Monday, March 3
Last correction at March 2, 2014, 9:09 pm

Reading

Numerical Recipes chapters 5, 6, and 7, skimming the technical bits

3.1 **Numerical derivatives.** (Rounding errors, Accuracy) ②

Calculate the numerical first derivative of the function $y(x) = \sin(x)$*, using the centered two-point formula* $dy/dx \sim (y(x+h) - y(x-h))/(2h)$*, and plot the error* $y'(x) - \cos(x)$ *in the range* $(-\pi, \pi)$ *at 100 points.* (Do a good job! Use the step-size h described in Numerical Recipes section 5.7 to optimize the sum of the truncation error and the rounding error. Also, make sure that the step-size h is exactly representable on the machine.) *How does your actual error compare to the fractional error estimate given in NR section 5.7? Calculate and plot the numerical second derivative using the formula*

$$\frac{\mathrm{d}^2 y}{\mathrm{d}x^2} \sim \frac{y(x+h) - 2y(x) + y(x-h)}{h^2}, \tag{1}$$

again optimizing h *and making it exactly representable. Estimate your error again, and compare to the observed error.*

3.2 **Summing series.** (Efficiency) ②

Write a routine to calculate the sum

$$s_n = \sum_{j=0}^{n} (-1)^j \frac{1}{2j+1}. \tag{2}$$

As $n \to \infty$, $s_\infty = \pi/4$*. About how many terms do you need to sum to get convergence to within* 10^{-7} *of this limit? Now try using Aitken's* Δ^2 *process to accelerate the convergence:*

$$s'_n = s_n - \frac{(s_{n+1} - s_n)^2}{s_{n+2} - 2s_{n+1} + s_n}. \tag{3}$$

About how many terms do you need with Aitken's method to get convergence to within 10^{-7}*?*

3.3 **Random histograms.** (Random numbers) ②

(a) *Investigate the random number generator for your system of choice. What is its basic algorithm? Its period?*

(b) *Plot a histogram with 100 bins, giving the normalized probability density of 100,000 random numbers sampled from (a) a uniform distribution in the range* $0 < x < 2\pi$*, (b) an exponential distribution* $\rho(x) = 6\exp(-6x)$*, and (c) a normal distribution of mean* $\bar{x} = 3\pi/2$ *and standard deviation* $\sigma = 1/\sqrt{6}$*. Before each plot, set the seed of your random number generator. Do you now get the same plot when you repeat?*

3.4 **Monte Carlo integration.** (Random numbers, Robust algorithms) ③

How hard can numerical integration be? Suppose the function f is wildly nonanalytic, or has a peculiar or high-dimensional domain of integration? In the worst case, one can always try Monte Carlo integration. The basic idea is to pepper points at random in the integration interval. The integration volume times the average of the function $V\langle f\rangle$ is the estimate of the integral.

As one might expect, the expected error in the integral after N evaluations is given by $1/\sqrt{N-1}$ times the standard deviation of the sampled points (NR equation 7.7.1).

(a) Monte Carlo in one dimensional integrals. *Use Monte Carlo integration to estimate the integral of the function introduced in the preliminary exercises*

$$y(x) = \exp(-6\sin(x)) \tag{4}$$

over $0 \le x < 2\pi$*.* (The correct value of the integral is around 422.446.) *Empirically, how many points do you need to reliably get 1% accuracy? How many points should you need, given the standard deviation?* (You may use the fact that $\langle y^2\rangle = (1/2\pi)\int_0^{2\pi} y^2(x)\mathrm{d}x \approx 18948.9$.)

Monte Carlo integration is not the most efficient method for calculating integrals of smooth functions like $y(x)$. Indeed, since $y(x)$ is periodic in the integration interval, equally spaced points weighted equally (the trapezoidal rule) gives exponentially rapid convergence; it takes only nine points to get 1% accuracy. Even for smooth functions, though, Monte Carlo integration is useful in high dimensions.

(b) Monte Carlo in many dimensions (No detailed calculations expected). *Does the number of Monte Carlo points needed depend on the dimension of the space, presuming (perhaps naively) that the variance of the function stays fixed? For a hypothetical ten-dimensional integral with the same variance, to get 1% accuracy we would need a regular grid with nine points along each axis. How much more efficient is Monte Carlo in that case?*

Our function $y(x)$ is quite close to a Gaussian. (Why? Taylor expand $\sin(x)$ about $x = 3\pi/2$.) We can use this to do *importance sampling*. The idea is to evaluate the integral of $h(x)g(x)$ by randomly sampling h with probability g, picking $h(x) = y(x)/g(x)$. The variance is then $\langle h^2\rangle - \langle h\rangle^2$. In order to properly sample the tail near $x = \pi/2$, we should mix a Gaussian and a uniform distribution:

$$g(x) = \frac{\epsilon}{2\pi} + \frac{1-\epsilon}{\sqrt{\pi/3}}\exp(-6(x-3\pi/2)^2/2). \tag{5}$$

I found minimum variance around $\epsilon = 0.005$.

(c) Importance Sampling (Optional for 480). *Generate 1000 random numbers with probability distribution* g.[1] *Use these to estimate the integral of* $y(x)$*. How accurate is your answer?*

[1] Take $(1-\epsilon)M$ Gaussian random numbers and ϵM random numbers uniformly distributed on $(0, 2\pi)$.

The Gaussian has a small tail that extends beyond the integration range $(0, 2\pi)$, so the normalization of the second term in the definition of g is not quite right. You can fix this by simply throwing away any samples that include points outside the range.