Statistics & Numerical Method, Problem Set 1 (due Oct. 15, 2024)

Attention: you are required to attach your codes in original format for problems 1, 3, 4 whenever applicable, and your codes should be properly documented/commented (e.g., explain the meaning of variables, role of each loop, etc.), which are important references for grading.

1. Machine precision (4 pts)

Write a computer program in your favourite language that *experimentally* determines the following.

(1). The machine precision ϵ_m, which can be considered as the smallest number ϵ_m such that $1+\epsilon_m$ still evaluates to something different from 1. Do so for both single and double precisions. (2pts)

(2). The smallest positive number $f_{\min}$, for each data type. Explain your findings. (2pts)

2. Numerical derivative on non-uniform grid (4 pts)

A function $f(x)$ is known at three consecutive points $x_1 < x_2 < x_3$ to be y_1, y_2 and y_3, where these points are not uniformly spaced. Estimate $f'(x)$ and $f''(x)$ at the middle point $x = x_2$, which will reduce to central differencing formula when x_1, x_2, x_3 are evenly spaced.

3. Numerical integration (4 pts)

The comoving distance to an object at redshift z in a flat universe is given by

$$D_C(z) = \frac{c}{H_0}\int_0^z \frac{H_0}{H(z')}dz' \ , \tag{1}$$

where

$$H(z)/H_0 = [\Omega_m(1+z)^3 + \Omega_\Lambda]^{1/2} \ . \tag{2}$$

Here, $H_0 = 67$ km s^{-1} Mpc^{-1} is the present-day Hubble constant, $\Omega_m = 0.3$ represents the total matter (dark and baryonic) density, and $\Omega_\Lambda = 1 - \Omega_m$ represents the dark energy density. (For non-astronomy students, Mpc is mega parsec, i.e., 10^6pc=3.26×10^6 light years.)

Using composite Simpson's rule, estimate the comoving distance for redshift $z = 1.0, 3.0$ and 8.2 within a relative precision of $\epsilon = 10^{-4}$. Try to adaptively determine the precision during your integration. How many steps are needed to achieve this accuracy in each case?

4. Hilbert Matrix (18 pts)

Hilbert matrix is a well-known example of ill-conditioned matrices, which looks like this:

$$A = \begin{bmatrix} 1 & 1/2 & 1/3 & \dots & 1/n \\ 1/2 & 1/3 & 1/4 & \dots & 1/(n+1) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1/n & 1/(n+1) & 1/(n+2) & \dots & 1/(2n-1) \end{bmatrix} \ . \tag{3}$$

In this problem, we ask you to write your own piece of code, in your favorite programming language, to solve the problem of $A\boldsymbol{x} = \boldsymbol{b}$. You should make your code sufficiently flexible so that you can switch between single and double precisions, and compare the results in problems (2)-(3) below.

(1). The Hilbert matrix is symmetric and positive definite. Write a direct matrix solver using Cholesky decomposition. (5 pts)

(2). From your decomposition, solve $A\boldsymbol{x} = \boldsymbol{b}$ where $b_i = \sum_{j=1}^{n}(i+j-1)^{-1}$ so that the exact solution should be $\boldsymbol{x} = (1, 1, \ldots, 1)^T$. Take $n = 5$ for this question, and show results for both single and double precisions. What is the level of error compared to the expected solution? (4pts)

(3). At what n will your method become unstable for single and double precisions (in this problem, let us say you find a more than $\sim 50\%$ relative error in terms of a vector norm)? (2pts)

(4). Calculate the condition number of A based on $\infty-$norm for $n = 3, 6, 9, 12$. How are the results related to your observations in (3)? (5pts)

(5). Can you suggest and/or demonstrate ways to improve the situation? (2 pts)