MTH 308: Numerical Analysis & Scientific Computing I
Department of Mathematics and Statistics
Indian Institute of Technology - Kanpur January 17, 2019

Lab 2

Steps you should follow :

1. Start a Matlab session by typing

   ```
   $ matlab &
   ```

   at the command prompt.

2. In the Matlab command window, type

   ```
   >> diary lab2_ID
   ```

   where ID stands for your roll number. For example, if your roll number is 12345, then the command will be `diary lab2_12345`; if your roll number is 123456, then the command will be `diary lab2_123456`. This will create a file named `lab2_ID` in the present working directory. PLEASE DO NOT EDIT THIS FILE.

3. Do your lab assignment – create a separate file to write your scripts/functions, if required; once done, at the Matlab's command prompt, type

   ```
   >> diary off
   ```

4. Attach this file (that is, `lab2_ID`) and any other matlab code file that you may have created for the labwork in an **email with subject Lab2-ID** and send it to

   mth308.iitk@gmail.com

   before the end of the lab session, that is, by 3:50 pm. Note that **late submissions will not get any credit**.

Consider the $n \times n$ nonsingular matrix

$$A_n = \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \cdots & \frac{1}{n} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \cdots & \frac{1}{n+1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{1}{n} & \frac{1}{n+1} & \frac{1}{n+2} & \cdots & \frac{1}{2n-1} \end{bmatrix}.$$

The problem of calculating the inverse of A_n is well-posed but increasing ill-conditioned as n increases. The ill-conditioning of the numerical inversion of A_n can be seen in a practical setting. Let $\widehat{A_n}$ denote the result of entering the matrix A_n using single precision floating-point format (you can achieve this in Matlab by first creating the matrix A_n and then use Matlab's `single` command, for example, `A_hat = single(A)`). Compute the inverse of A_n and $\widehat{A_n}$ for $n = 2, 3, 4, \ldots$. What is the smallest value of n for which the relative error

$$\frac{\|A_n^{-1} - \widehat{A_n}^{-1}\|}{\|A_n^{-1}\|}$$

more that 1 (that is, the error is more than 100%). What is the condition number of A_n for that n?

Some extra practice homework problems (not to be submitted as part of the lab assignment and will not be graded).

I. Show that $f_{j,k} = \sin(x_0 + (j-k)\pi/3)$ satisfies the recurrence relation

$$f_{j,k+1} = f_{j,k} - f_{j+1,k}. \tag{1}$$

We view this as a formula that computes the f values on level $k+1$ from the f values on level k. Let $f^a_{j,k}$ for $k \geq 0$ be the floating point numbers that come from implementing $f_{j,0} = \sin(x_0 + j\pi/3)$ and (1) (for $k > 0$) in double precision floating point. If $\left|f^a_{j,k} - f_{j,k}\right| \leq \epsilon$ for all j, show that $\left|f^a_{j,k+1} - f_{j,k+1}\right| \leq 2\epsilon$ for all j. Thus, if the level k values are very accurate, then the level $k+1$ values still are pretty good.

Write a program that computes $e_k = \left|f^a_{1,k} - f_{1,k}\right|$ for $1 \leq k \leq 60$ and $x_0 = 1$. Print the e_k and see whether they grow monotonically. Plot the e_k with respect to k and see that the numbers seem to go bad suddenly at around $k = 50$.

Repeat this computation in single precision and comment on the result. Note that, if you are using MATLAB, by default, it does computations in *double precision*.

II. The Fibonacci numbers, f_k, are defined by $f_0 = 1, f_1 = 1$, and

$$f_{k+1} = f_k + f_{k-1} \tag{2}$$

for any integer $k > 1$. A small perturbation of them, the *pib numbers* ("p" instead of "f" to indicate a perturbation), p_k, are defined by $p_0 = 1, p_1 = 1$, and

$$p_{k+1} = c\, p_k + p_{k-1}$$

for any integer $k > 1$, where $c = 1 + \sqrt{3}/100$.

(a) Plot the f_n and p_n together on a log scale plot. On the plot, mark $1/\epsilon_{mach}$ for single and double precision arithmetic. This can be useful in answering the questions below.

(b) Rewrite (2) to express f_{k-1} in terms of f_k and f_{k+1}. Use the computed f_n and f_{n-1} to recompute f_k for $k = n-2, n-3, \ldots, 0$. Make a plot of the difference between the original $f_0 = 1$ and the recomputed $\hat{f}_0$ as a function of n. What n values result in no accuracy for the recomputed f_0? How do the results in single and double precision differ?

(c) Repeat (b) for the *pib numbers*. Comment on the striking difference in the way precision is lost in these two cases. Which is more typical?

III. The binomial coefficients, $a_{n,k}$, are defined by

$$a_{n,k} = \binom{n}{k} = \frac{n!}{k!(n-k)!}.$$

To compute the $a_{n,k}$, for a given n, start with $a_{n,0} = 1$ and then use the recurrence relation

$$a_{n,k+1} = \frac{n-k}{k+1} a_{n,k}.$$

(a) For a range of n values, compute the $a_{n,k}$ this way, noting the largest $a_{n,k}$ and the accuracy with which $a_{n,n} = 1$ is computed. Do this in single and double precision. Why is roundoff not a problem here as it was in problem IV? Find n values for which $\hat{a}_{n,n} \approx 1$ in double precision but not in single precision. How is this possible, given that roundoff is not a problem?

(b) Use the algorithm of part (a) to compute

$$E(k) = \frac{1}{2^n} \sum_{k=0}^{n} k a_{n,k} = \frac{n}{2}. \tag{3}$$

Write a program without any safeguards against overflow or zero divide. Show (both in single and double precision) that the computed answer has high accuracy as long as the intermediate results are within the range of floating point numbers. As with (a), explain how the computer gets an accurate, small, answer when the intermediate numbers have such a wide range of values. Why is cancellation not a problem? Note the advantage of a wider range of values: we can compute E(k) for much larger n in double precision. Print $E(k)$ as computed by (3) and $M_n = \max_k a_{n,k}$. For large n, one should be `inf` and the other `NaN`. Why?

(c) For fairly large n, plot $a_{n,k}/M_n$ as a function of k for a range of k chosen to illuminate the interesting "bell shaped" behavior of the $a_{n,k}$ near $k = n/2$. Combine the curves for $n = 10, n = 20$, and $n = 50$ in a single plot. Choose the three k ranges so that the curves are close to each other. Choose different line styles for the three curves.