Problem Set 2
Math 146 Spring 2023
Due Thursday, February 16, 11:59 PM

Note: Numbers 1-4 are "pen and paper" problems for which you should show work, where appropriate. Numbers 5-7 are Matlab problems that require code submissions.

1. Let $A = \begin{bmatrix} -2 & 2 & 0 \\ 0 & 0 & 1 \\ -1 & -1 & 0 \end{bmatrix}$

   (a) Find $||A||_1, ||A||_2,$ and $||A||_\infty$. (You may check these with Matlab, but you should show the work to do them by hand.)

   Find $A\mathbf{x}$ for $\mathbf{x} = \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}$ and show that $||A\mathbf{x}|| \leq ||A|| ||\mathbf{x}||$ for all three norms.

   (b) Find $\kappa_1(A)$, and $\kappa_\infty(A)$, the relative condition numbers for the matrix A, with respect to the 1 and max norms, respectively. You may use Matlab to find A^{-1}.

   (c) Based on your answers in part (c), how many digits of accuracy do you expect to have when performing matrix-vector multiplication with this matrix A (in double precision)? Explain.

2. Consider the matrix $V \in \mathbb{R}^{n\times n}$ whose components are expressed as

$$V_{ij} = \begin{cases} \sum_{k=1}^{i+j} \frac{1}{k} & \text{if } i \leq j, \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

   (a) Write down the matrix V for $n = 3$.

   (b) Write a pseudocode to create the matrix V for given n.

   (c) Perform a FLOP count for your pseudocode. Note that $1/k$ is a floating-point operation even though 1 and k are integers. Write your FLOP count as a summation. You do NOT need to simplify to a polynomial expression. However, determine the *order* of the FLOP count (i.e, in addition to providing the exact FLOP count in summation notation, also provide the FLOP count in "Big-Oh" notation (ex, $O(n), O(n^2)$, etc). Justify the order you find.

3. An upper bidiagonal matrix is a matrix with a main diagonal and one upper diagonal:

$$A = \begin{bmatrix} a_{11} & a_{12} & & & 0 \\ & a_{22} & a_{23} & & \\ & & \ddots & \ddots & \\ & & & a_{n-1,n-1} & a_{n-1,n} \\ 0 & & & & a_{nn} \end{bmatrix} \tag{2}$$

(a) For a linear system $A\mathbf{x} = \mathbf{b}$, derive expressions for x_n and x_k $(k = n-1, \cdots, 1)$.

(b) Write a pseudocode to solve the linear system that uses only the nonzero element of A.

(c) Perform a flop count for your pseudocode.

4. Solve the following system (by hand) by first finding the LU decomposition of the matrix (without pivoting) and then using forward/backward substitutions.

$$\begin{bmatrix} 1 & 0 & 0 & 1 \\ -1 & 2 & 0 & 1 \\ -1 & -1 & 1 & 1 \\ -1 & 3 & -1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} -3 \\ 3 \\ 0 \\ 2 \end{bmatrix} \tag{3}$$

5. (a) Write a Matlab function to perform forward substitution to solve a linear system $L\mathbf{x} = \mathbf{b}$. Your function should take as input L, an $n \times n$ lower triangular matrix with no zeros on the diagonal, and $\mathbf{b}$, a vector of length n. It should return the solution $\mathbf{x}$ as output.

   (b) Validate your code by forming a 10×10 lower triangular matrix for L made up of random numbers between 0 and 20, as well as a similarly formed random vector, $\mathbf{b}$. Solve for $\mathbf{x}$ using both your function and Matlab function `backslash`. Then find the relative difference of your solution from the Matlab solution, with respect to the max norm. (i.e. find the relative "error" of your vector, using the Matlab vector as the "exact" solution).

   (c) Comment on the size of your relative difference found in part (b). Why is *error* written in quotations? (Why is this not a relative error?) Lastly, how many digits would you say that you can "trust" for each component of your solution $\mathbf{x}$ (specifically for the matrix generated for part (b))? (And why?)

6. (a) Write a Matlab function to perform backward substitution to solve a linear system $U\mathbf{x} = \mathbf{b}$. Your function should take as input U, an $n \times n$ upper triangular matrix with no zeros on the diagonal, and $\mathbf{b}$, a vector of length n. It should return the solution $\mathbf{x}$ as output. (Note: As we did not write pseudocode in class, it is advised that you begin by writing pseudocode before writing Matlab code.)

   (b) Validate your code by forming a 10×10 upper triangular matrix for U made up of random numbers between 0 and 20, as well as a similarly formed random vector, $\mathbf{b}$. Solve for $\mathbf{x}$ using both your function and Matlab function `backslash`. Then find the relative difference of your solution from the Matlab solution, with respect to the max norm.

7. (a) Write a Matlab function to use Gaussian Elimination (without pivoting) to perform LU decomposition for a matrix A. Your function should take as input A, an $n \times n$ matrix. Its output should be the lower triangular and upper triangular matrices, L and U.

   (b) Use your code to find the LU decomposition to the matrix in problem (4).

(c) Also validate your code by forming a 1000×1000 matrix for A_1 made up of random numbers between 0 and 20. Use your function to find L and U, find the difference between A and LU, and then report the component of the difference matrix that has the largest magnitude. (Hint: Make sure to use semi-colons to suppress the output - you do not want to print out A,L,or U.)

(d) In class, we discussed that the FLOP count for this operation is $O(n^3)$. Knowing that, what do you expect to happen to the run time if you double the "size" of the problem, n? Test this by `tic toc` and and running your code on A_1, a 1000×1000 matrix and then A_2, a 2000×2000 matrix. Give the run times, and comment on your results (Is it what you expected? If not, what do you expect is the reason? Also, comment on the scalability of this algorithm.)