Assignment #1

Due Wednesday 1 September, 2021 (*updated*) at the start of class

A major purpose of this assignment is to familiarize you with MATLAB (or OCTAVE, PYTHON, JULIA, etc.; see the "Programming languages" handout). You will need to find, download, or purchase a copy. Make sure you can open and close the command window, use it as a calculator, and plot simple things. Make sure you can create a new program (text file), save it, edit it, and run it at the command line.

Please read Lectures 1 and 2 in the textbook *Numerical Linear Algebra* by Trefethen and Bau. Do these exercises:

P1. (*This exercise is, in part, a reminder of some things you knew how to do during your undergraduate linear algebra course.*)

Consider the 3×3 real matrix

$$A = \begin{bmatrix} 2 & 1 & 1 \\ 4 & 0 & -2 \\ 2 & 2 & 3 \end{bmatrix}$$

(a) By hand, on paper:

- compute the eigenvalues of A,
- compute the rank of A,
- compute the determinant of A,
- compute the inverse of A (if possible),
- compute the inverse of $B = A(2:3, 1:2)$ (if possible), and
- solve the linear system $Ax = b$ where $b = \begin{bmatrix} -1 & 8 & -6 \end{bmatrix}^*$ (if possible).

(b) Now check your work at the MATLAB command line. (*You'll use these* MATLAB *commands among others:* `eig`, `rank`, `det`, `inv`, `\` .)

P2. (*This problem exercises additional* MATLAB *commands such as* `rand`, `ones`, `norm`, `plot`, `loglog`, `semilogy`. *It is also an exercise in communicating numerical results. In fact, because you will generate 80 matrices, please do* not *turn in a giant table of the matrices themselves! Don't even turn in a table of their ranks, norms, or determinants. Instead, use plots to communicate data and patterns, and sentences to state conclusions. It may make sense to compute averages over the 10 tries, but sometimes raw data can appear nicely in a plot.*)

Write a MATLAB script which generates 10 random matrices of size $m \times m$ for each of these powers of two: $m = 2, 4, 8, \ldots, 256$. Every matrix will have entries which are random real numbers uniformly distributed on $[-10, 10]$. For each of these matrices compute the rank, the 2-norm, and the absolute value of the determinant. Communicate these data using plots in reasonable ways; a significant part of your script will be devoted to generating the plots.

P3. (*This problem is about the most basic algorithms of linear algebra, and about their elementary implementations. Nothing fancy. Be careful with indices!*)

(a) Consider the algorithm which computes the product of a rectangular matrix $A \in \mathbb{C}^{m\times n}$ and a column vector $v \in \mathbb{C}^{n\times 1}$. Count the number of floating point operations exactly, i.e. as an expression in m and n, for this algorithm.

(b) Now implement this algorithm in a program `matvec.m` which is a function; the first line will say

```
function z = matvec(A,v)
```

The code will start by extracting the sizes of the input objects using the MATLAB command `size`. Use `error` if the user-supplied inputs `A` and `v` have incompatible sizes. Use `for` loops to implement the algorithm. (That is, use `for` loops instead of colon notation; thereby show the underlying implementation.) Check your code against some example $A \in \mathbb{R}^{4\times 3}$ and $v \in \mathbb{R}^3$, perhaps using `A*v` in MATLAB.

(c) Write a function `matmat.m` for the product $C = AB$ of matrices $A \in \mathbb{R}^{m\times n}$ and $B \in \mathbb{R}^{n\times k}$. Count operations. Check your code against the MATLAB result `A*B` on some example with $m = 3$, $n = 4$, and $k = 3$.

P4. Let B be any 4×4 matrix to which we apply the following operations in turn:

1. interchange rows 1 and 3
2. interchange columns 2 and 4
3. double column 3
4. add row 3 to row 1
5. subtract row 2 from each of the other rows
6. replace column 3 by column 4
7. delete row 1 (so the resulting matrix is 3×4)

(a) Write the result as a product of eight matrices.

(b) Write it again as a product ABC of three matrices; B is the same.

Exercise 1.3 in Lecture 1.