MAE384, Spring 2020 Homework #2

The deadline for submission of hard-copy reports will be announced by instructor. Please follow the rules on collaboration as given in Homework #1. A statement on collaboration is required for all reports. <u>Computer codes used to complete the tasks should be included in the report.</u>

Prob 1 (4 points)
A system of linear equations is given:

$$\begin{pmatrix} 8 & 3 & 4 & 0.5 \\ 1 & 3 & 1 & 0.5 \\ 0.5 & 2 & 5 & 1.5 \\ 2 & 0.5 & 1 & 4 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 6 \\ 4.5 \\ 7.5 \\ 9 \end{pmatrix} . \qquad Eq.(1)$$

Symbolically, we write *Eq*. (1) as

$$\boldsymbol{A}\,\vec{\boldsymbol{x}} = \vec{\boldsymbol{b}} \ , \qquad Eq.(2)$$

which also gives the definitions of the matrix $\boldsymbol{A}$ and vectors $\vec{\boldsymbol{x}}$ and $\vec{\boldsymbol{b}}$. For this system, the exact solution is known to be

$$\vec{\boldsymbol{x}}_S = \begin{pmatrix} 0 \\ 1 \\ 0.5 \\ 2 \end{pmatrix} .$$

Let us also recall a few relevant definitions. Given $\vec{\boldsymbol{x}}_N$ as a numerical solution, the **relative true error** (*RTE*) is defined as

$$RTE = \frac{||\vec{\boldsymbol{e}}||}{||\vec{\boldsymbol{x}}_S||} \ , \qquad Eq.(3)$$

where $\vec{\boldsymbol{e}} = \vec{\boldsymbol{x}}_N - \vec{\boldsymbol{x}}_S$ is the vector of absolute error. The **relative residual** (*RRS*) is defined as

$$RRS = \frac{||\vec{\boldsymbol{r}}||}{||\vec{\boldsymbol{b}}||} \ , \qquad Eq.(4)$$

where $\vec{\boldsymbol{r}} = \boldsymbol{A}\vec{\boldsymbol{x}}_N - \vec{\boldsymbol{b}}$ is the residual vector. The key conclusion of the discussion (in Lecture 11) on **condition number**, *C*, is summarized by the inequality,

$$\frac{1}{C} * (RRS) \le (RTE) \le C * (RRS) \ . \qquad Eq.(5)$$

Conceptually, *Eq*. (5) provides the upper and lower bounds of *RTE* by computing *RRS* and *C*. Lastly, the condition number itself is defined by

$$C = ||\boldsymbol{A}||\ ||\boldsymbol{A}^{-1}|| \ . \qquad Eq.(6)$$

In *Eqs*. (3)-(6), Euclidean norm is used for a vector and Euclidean 2-norm (or "Frobenius norm") is used for a matrix. Let x_j be the j-th element of a vector $\vec{\boldsymbol{x}}$ with N elements, then

$$\|\vec{\boldsymbol{x}}\| = \sqrt{\sum_{j=1}^{N} (x_j)^2} \ .$$

Let A_{ij} be the element of an N x N matrix $\boldsymbol{A}$ at the i-th row and j-th column, then

$$\|\boldsymbol{A}\| = \sqrt{\sum_{i=1}^{N} \sum_{j=1}^{N} (A_{ij})^2} \ .$$

(a) Solve the system in *Eq*. (1) with two methods: (i) *Jacobi iterative method*, using (x_1, x_2, x_3, x_4) = (0,0,0,0) as the initial guess to perform 20 iterations. (ii) *Gauss-Seidel iterative method*, using (x_2, x_3, x_4) = (0,0,0) as the initial guess to perform 20 iterations. (Note that the *G-S* method does not require an initial guess for x_1.) Here, one iteration means completing the update of all x_1, x_2, x_3, and x_4 once. From the results, make a plot of the relative true error (*RTE*) vs. the number of iterations for both methods. Please <u>collect the two curves of *RTE* in the same plot</u> for a clear comparison. Since *RTE* decreases dramatically with the number of iterations, for clarity we require that the actual plot be made by taking the 10-based logarithm of *RTE*. In other words, the deliverable is the line plot of log_{10}(*RTE*) vs. the number of iterations. (In Matlab, the built-in function for 10-based log is `log10`.)

(b) Compute the condition number, *C*, for the system in *Eq*. (1). Using the result from the case in Part (a) with *Gauss-Seidel method*, make a plot with 3 curves (must be collected in the same plot): The first is the curve of log_{10}(*RTE*) vs. the number of iterations, as recycled from Part (a). (Despite the redundancy, for clarity we ask that a new plot be made for Part (b). Do not merge the deliverables in Part (a) and (b) into a single plot.) The other two are the curves for $log_{10}(RRS*C)$ and $log_{10}(RRS*C^{-1})$, vs. the number of iterations. This plot serves the purpose of demonstrating the relations in *Eq*. (5). Namely, $RRS*C$ and $RRS*C^{-1}$ provide the upper bound and lower bound of *RTE*. (If *C* is close to 1, these bounds are "tight" and *RRS* is a useful proxy of *RTE*. If *C* is very large, *RTE* is detached from *RRS*.)

Prob (2 points)
A set of 8 data points is given:

x	y
4	406
5	440
7	496
8	565
9	684
10	812
11	971
13	1000

(a) Perform linear least-squares regression (Sec 6.2.2) to obtain a line, $y = ax + b$, to represent the data. In addition, calculate the error of the least-squares fit, E, as defined by Eq. (6.6) in textbook. The deliverables are the linear formula (please provide the values of a and b) and the value of E.

(b) Perform quadratic least-squares regression (pp. 207-208; Eq. (6.22)-(6.28)) to obtain a quadratic formula, $y = p\,x^2 + q\,x + r$, to represent the data. In addition, calculate the error of the least-squares fit, E, as defined by Eq. (6.22) in textbook. The deliverables are the quadratic formula (please provide the values of p, q, and r) and the value of E. Compare this E value with that obtained in Part (a). Does the quadratic fit produce a smaller error compared to linear fit?

(c) Draw the two curves obtained in (a) and (b), i.e., the linear and quadratic curves, along with the original data points in a single plot. (Do not connect the original data points. Present them as isolated points. See Matlab Example 41-44 for the proper Matlab commands to use.)

Prob 3 (2 points)
A set of 4 data points is given:

x	y
1	1.2
1.8	3
3	2
5	2.5

(a) Following the procedure in Sec. 6.6.2, determine the quadratic splines that fit the data. Plot the quadratic splines and the original data points in a single figure, in the fashion of the figure in Example 6-7 in textbook. Show your procedure.

(b) Directly fit the data by a single 3rd-order polynomial that runs through all of the data points. In this case, you are required to use the Lagrange interpolation method (Sec 6.5.1) and show the procedure. Plot the 3rd-order polynomial and the original data points in a single figure.

If you choose to do so, it is fine to merge the plots for Part (a) and Part (b) into a single plot. Note that the “original data points” are the same for the two tasks.