

## MAE384, Spring 2018 Homework #2

Hard copy of report is due at 12:15 PM on the due date. Computer codes used to complete the tasks should be included in the report. Uses of `polyfit` and `interp1` functions in Matlab are not allowed for this assignment.

**Task 0** (no point, but mandatory to complete for the report to be accepted)

Provide a statement to address whether collaboration occurred in completing this assignment. **This statement must be placed in the beginning of the first page of report.** Please see related clarifications in Homework #1.

**Task 1** (5 points)

A system of linear equations is given as

$$\begin{aligned} 2x_1 + x_2 + 0.5x_3 &= 7 \\ x_1 + 4x_2 + 0.5x_3 &= 8 \\ x_1 + 0.5x_2 + 2x_3 &= 10.5 \end{aligned} \quad \text{Eq. (1)}$$

(a) Solve the system by *Gauss elimination*. Show your procedure. For this task (Task 1a) only, no rounding is allowed in the intermediate steps. We want the exact solution. Save the outcome of this part as the "exact" solution,  $\mathbf{x}_S$ , for the use in Part (d).

(b) Solve the system by the *Jacobi iterative method*, using  $(x_1, x_2, x_3) = (0,0,0)$  as the initial guess and perform 20 iterations. Here, one iteration means updating all  $x_1, x_2$ , and  $x_3$  once. The deliverable is a list of the outcome of  $(x_1, x_2, x_3)$  after each iteration.

(c) Solve the system by the *Gauss-Seidel iterative method*, using  $(x_1, x_2, x_3) = (0,0,0)$  as the initial guess. (Note that with the Gauss-Seidel method the initial guess of  $x_1$  is never used.) Perform 20 iterations. Again, one iteration means updating all  $x_1, x_2$ , and  $x_3$  once (see Example 4-8 in textbook). The deliverable is a list of the outcome of  $(x_1, x_2, x_3)$  after each iteration.

(d) From the results of (a), (b), and (c), calculate and plot the numerical error as a function of the number of iteration for both Jacobi and Gauss-Seidel methods. The error is defined using the *Euclidean 2-norm* (see Eq. 4.72) as

$$E \equiv \|\mathbf{x}_N - \mathbf{x}_S\| \equiv \sqrt{(x_{N,1} - x_{S,1})^2 + (x_{N,2} - x_{S,2})^2 + (x_{N,3} - x_{S,3})^2} \quad ,$$

where  $\mathbf{x}_N \equiv (x_{N,1}, x_{N,2}, x_{N,3})$  is the numerical solution from the iterative method and

$\mathbf{x}_S \equiv (x_{S,1}, x_{S,2}, x_{S,3})$  is the "exact" solution obtained by Gauss elimination from Part (a). For this task, the deliverable is a plot of  $\log_{10}(E)$  as a function of the number of iteration. (Here,  $\log_{10}$  is 10-based logarithm. In Matlab, the built-in function to use is `log10`.) The plot should include two curves (put together in the same plot), one for Jacobi and one for Gauss-Seidel method. Each curve should represent 21 data points associated to the errors for the initial guess and the subsequent 20 iterations. For the convenience of making the plot, you may designate the initial guess as the outcome of the "zeroth iteration".

(e) Using the *Euclidean 2-norm* for matrix (Eq. 4.76 in textbook), evaluate the *condition number* for the system in Eq. (1). (For this task, it is fine to use Matlab function `inv` to calculate the inverse of a matrix. See Matlab Example 39.)

**Task 2** (2 points)

The set of the following 8 data points is given:

$x$	$y$
8	56
9	68
10	81
12	97
13	100
14	101
15	104
17	112

(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 and the value of  $E$ .

(b) Perform quadratic least-squares regression (pp. 207-208; Eq. (6.22)-(6.28)) to obtain a quadratic formula,  $y = px^2 + qx + r$ , 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 and the value of  $E$ . Compare this  $E$  value with that obtained in (a). Does the quadratic fit produce a smaller error compared to linear fit?

(c) Draw the two curves obtained in (a) and (b) 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.) The recommended range for this plot is  $5 \leq x \leq 20$ ,  $40 \leq y \leq 140$ .

**Task 3** (2 points)

The set of the following 4 data points is given:

$x$	$y$
0.5	3
1.1	1
2.5	2
4.5	1.5

(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. You may use either the method of direct matrix solution (see the introduction of Sec 6.5, pp. 211-212) or the Lagrange interpolation method (Sec 6.5.1). Plot the polynomial and the original data points in a single figure. Show your procedure.