AM 147: Computational Methods and Applications: Winter 2022

Homework #6

Instructor: Abhishek Halder

Due: February 16, 2022

NOTE: Please submit your Homework as a single zip file named `YourlastnameYourfirstnameHW6.zip` via CANVAS. For example, `HalderAbhishekHW6.zip`. Please strictly follow the capital and small letters in the filename of the zip file you submit. You may not receive full credit if you do not follow the file-naming conventions. Your zip file should contain all .m files (MATLAB scripts) for the questions below.

Your zip file must be uploaded to CANVAS by 11:59 PM Pacific Time on the due date. The uploads in CANVAS are time-stamped, so please don't wait till last moment. Late homework will not be accepted.

Problem 1

Locating lost smartphone (25 points)

Suppose we want to locate a lost smartphone in two dimensions. Let the unknown position vector of the smartphone be $\boldsymbol{p} := \begin{pmatrix} x \\ y \end{pmatrix} \in \mathbb{R}^2$. Suppose that there are $m \geq 3$ nearby cell phone towers at known locations $\boldsymbol{p}_i := \begin{pmatrix} x_i \\ y_i \end{pmatrix}$ which can ping the smartphone to get the range measurements:

$$\|\boldsymbol{p} - \boldsymbol{p}_i\|_2 = r_i, \quad i = 1, 2, \ldots, m, \tag{1}$$

where r_i is the measured distance between the smartphone and the i^{th} tower.

Write a code `YourlastnameYourfirstnameHW6p1.m` that lets the user supply/declare (at the top of the file) the known data $\boldsymbol{p}_i, r_i$ where $i = 1, \ldots, m$ for any $m \geq 3$, and then numerically estimates the unknown location of the smartphone.

To do so, you need to reformulate this problem as a standard least squares problem. Start by squaring both sides of (1), and subtract the first squared equation from every other squared equation. Write the resulting equations as a tall linear system $\boldsymbol{A}\boldsymbol{p} = \boldsymbol{b}$ where $\boldsymbol{A} \in \mathbb{R}^{(m-1)\times 2}$, $\boldsymbol{b} \in \mathbb{R}^{m-1}$ depend on the known data. (You don't need to show/submit any hand calculations.)

For $m = 4$, use your code to compute the least squares estimate $\hat{\boldsymbol{p}}$ for the following data:

$$\boldsymbol{p}_1 = \begin{pmatrix} -3 \\ 1.5 \end{pmatrix}, \quad \boldsymbol{p}_2 = \begin{pmatrix} 2.1 \\ 3.7 \end{pmatrix}, \quad \boldsymbol{p}_3 = \begin{pmatrix} 0.9 \\ 1.4 \end{pmatrix}, \quad \boldsymbol{p}_4 = \begin{pmatrix} -1.5 \\ -2.7 \end{pmatrix}, \quad \begin{pmatrix} r_1 \\ r_2 \\ r_3 \\ r_4 \end{pmatrix} = \begin{pmatrix} 3.61 \\ 5.32 \\ 2.88 \\ 3.25 \end{pmatrix}.$$

Problem 2

Least squares solution and round-off errors (25 **points**)

Let k be any positive integer, and $\boldsymbol{A} = \begin{pmatrix} 1 & 1 \\ 10^{-k} & 0 \\ 0 & 10^{-k} \end{pmatrix}$, $\boldsymbol{b} = \begin{pmatrix} -10^{-k} \\ 1 + 10^{-k} \\ 1 - 10^{-k} \end{pmatrix}$.

Write a MATLAB code `YourlastnameYourfirstnameHW6p2.m` that for each $k = 5, 6, 7, 8$, computes the least squares solution for the above $\boldsymbol{A}, \boldsymbol{b}$, in three different ways: (i) analytically solve the normal equation by hand and hard code that analytical solution `xhat_analytical`, (ii) `xhat_QR` that numerically solves the normal equation using QR decomposition via MATLAB `A\b`, (iii) `xhat_normal` that numerically solves the normal equation as a square linear system. In other words, your code should output 3 different least squares solutions for each k, where the integer k increments in a for loop from 5 to 8.

For computing the analytical solution in part (i), you may use the formula for the inverse of a 2×2 matrix:

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix}^{-1} = \frac{1}{ad - bc} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}.$$

Please also submit a file `YourlastnameYourfirstnameHW6p2.pdf` showing all the steps in your hand calculations for part (i).