Lab Session 5

MA-423 : Matrix Computations Lab 2017 R. Alam

The purpose of this lab tutorial is to solve the Least-Squares Problem (in short, LSP) $Ax = b$. Here $A \in \mathbb{C}^{m\times n}$ and $b \in \mathbb{C}^m$, and usually m is much bigger than n.

Origin: Suppose that we have a data set (t_i, b_i), for $i = 1 : m$, that have been obtained from some experiment. These data are governed by some unknown laws. So, the task is to come up with a model that best fits these data. A model is generated by a few functions, called model functions, $\phi_1, \ldots, \phi_n$. Therefore once a model is chosen, the task is to find a function p from the span of the model functions that best fits the data.

Suppose that the model functions $\phi_1, \ldots, \phi_n$ are given. For $p \in \mathrm{span}(\phi_1, \ldots, \phi_n)$, we have $p = x_1\phi_1 + \cdots + x_n\phi_n$ for some $x_j \in \mathbb{C}$. Now, forcing p to pass through the data (t_i, b_i) for $i = 1 : m$, we have $p(t_i) = b_i + r_i$, where r_i is the error. We want to choose that p for which the sum of the squares of the errors r_i is the smallest, that is, $\sum_{i=1}^{m} |r_i|^2$ is minimized.

Now $p(t_i) = b_i + r_i$ gives $x_1\phi_1(t_i) + \cdots + x_n\phi_n(t_i) = b_i + r_i$. Thus in matrix notation,

$$\begin{bmatrix} \phi_1(t_1) & \cdots & \phi_n(t_1) \\ \phi_1(t_2) & \cdots & \phi_n(t_2) \\ \vdots & \cdots & \vdots \\ \phi_1(t_m) & \cdots & \phi_n(t_m) \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix} + \begin{bmatrix} r_1 \\ r_2 \\ \vdots \\ r_m \end{bmatrix}.$$

This is of the form $Ax = b + r$ and we have to choose $x \in \mathbb{C}^n$ for which $\|r\|_2 = \|Ax - b\|_2$ is minimized. We write this as LSP $Ax = b$.

1. Consider the following data

t	1.0	1.5	2.0	2.5	3.0
b	1.1	1.2	1.3	1.3	1.4

   (a) Setup the LSP $Ax = b$ for a straight line passing through the data points. Use the standard basis polynomials $\phi_1(t) = 1$ and $\phi_2(t) = t$.
   Useful MATLAB commands: `t = 1:.5:3; t = t'; s = ones(5,1); A = [s t];`

   (b) Compute solution of the LSP $Ax = b$ by using the MATLAB command $x = A\backslash b$ (the "backslash" command).

   (c) Use the MATLAB `plot` command to plot the five data points and your least squares straight line. Type `help plot` for information about using the `plot` command.

   (d) Use MATLAB to compute $\|r\|_2$, the norm of the residual r.

2. Determine the polynomial of degree 19 that best fits the function $f(t) = \sin(\frac{\pi}{5}t) + \frac{t}{5}$ for $t_1 = -5, t_2 = -4.5, \ldots, t_{23} = 6$. Setup the LSP $Ax = b$ and determine the polynomial p in three different ways:

   (a) By using the matlab command
   `>> A \ b`
   This uses QR factorization to solve the LSP $Ax = b$. Call this polynomial p_1.

(b) By solving the normal equation $A^*Ax = A^*b$. Use `x = (A'*A)\(A'*b)`. Call this polynomial p_2.

(c) By solving the system $\begin{bmatrix} I_m & A \\ A^* & 0 \end{bmatrix} \begin{bmatrix} -r \\ x \end{bmatrix} = \begin{bmatrix} b \\ 0 \end{bmatrix}$. Call this polynomial p_3.

Compute the condition number (use the matlab command `cond(A)`) of the coefficient matrix associated with each of the systems that you are solving. Print the result to 16 digits (use `format long e`). Which one is the most ill conditioned?

The norm of the residual $\|r\|_2 = \sqrt{\Sigma_{i=1}^{23}|p_j(t_i) - f(t_i)|^2}$ $j = 1, 2, 3$ for each of these methods gives an idea of the goodness of the fit in each case. Compute these norms (again in `format long e`). You may use the `polyval` command to evaluate the polynomials p_1, p_2 and p_3 at the points $t_i, i = 1 : 23$. However you must flip the vectors p_1, p_2 and p_3 upside down by using the `flipud` command before this. Type `help polyval` and `help flipud` for details.

Finally, plot the polynomials p_1, p_2, p_3 and the function f on $[-5, 6]$. Use different colours to distinguish these plots. Do you observe any difference? If yes, which polynomial is a better approximation of f?

*** End ***