Matlab Exercises_Recitation 7 — 2.086 Spring 2012

Recitation 7: Wednesday, 21 March / Friday, 23 March
MATLAB Exercises_Recitation 7 due: *Monday, 2 April 2012 at 5 PM by upload to Stellar*

Format for upload: Students should upload to the course Stellar website a folder

`YOURNAME_MatlabExercises_Rec7`

which contains the completed scripts and functions for the assigned MATLAB Exercises_Recitation 7: all the scripts should be in a single file, with each script preceded by a comment line which indicates the exercise number; each function `.m` file should contain a comment line which indicates the exercise number.

---

1. (*i*) Write a function with signature

   ```
   function [Ainv] = my2by2inv(A)
   ```

   which takes as input `A` a 2×2 matrix and returns an output `Ainv`, the inverse of this matrix. (You may assume that the input `A` is indeed a 2×2 matrix and that furthermore `A` is a non-singular matrix and hence the inverse exists.) Note: You should not use the MATLAB built-in `inv` but rather write your own code based on the explicit formula for the inverse of a 2×2 matrix.

   (*ii*) Write a script which calls your function `my2by2inv` with input matrix `A = [3, -1; 1, 1]` and displays the output (i.e., the inverse of `A` as computed by your function).

2. For X and Y (with $m = 20$) as defined in MATLAB Exercises Recitation 6 write a script which finds the least squares solution $\hat{\beta}$ — which we recall minimizes $\|Y - X\beta\|^2$ — in the four fashions below. In each case, have the script display the result.

   (*i*) `betahat_a = my2by2inv(X'*X)*(X'*Y)`

   (*ii*) `betahat_b = inv(X'*X)*(X'*Y)`

   (*iii*) `betahat_c = (X'*X) \ (X'*Y)`

   (*iv*) `betahat_d = X \ Y`

   Note option (*iv*) is in general the best way — the MATLAB backslash (and underlying numerical approaches) is the fastest and most stable option for solution of linear systems and of least squares problems.

   In general, you should avoid the direct computation of the matrix inverse and the MATLAB built-in `inv`, however for small problems `inv` works just fine. (You will need `inv(X'*X)` to calculate your confidence intervals in Problem Set 3.)

2.086 Numerical Computation for Mechanical Engineers
Fall 2012