Homework 5, Math 6610-1, Due Oct. 19

In this project, we will investigate the performances of various algorithms for a least square problem to fit function values on a grid by a polynomial of degree $n - 1$. Use MATLAB's `linspace` to define t to be the m-vector corresponding to equally spaced grid points from 0 to 1, and MATLAB's `vander` and `fliplr` to form the $m \times n$ matrix A associated with the least square fitting on this grid. Take b to be the function $\cos(4t)$ evaluated on the grid. The least square solution vector x will contain all the coefficients of the polynomial. Compute and print (to 16-digit precision) the solution vector x by all the algorithms below. You can take $m = 50$, $n = 12$ for this project.

1. Normal equation, using MATLAB's linear system solver, $x = (A' * A) \backslash (A' * b)$;

2. QR factorization computed by `mgs` (modified Gram-Schmidt);

3. QR factorization computed by `house` (Householder triangularization);

4. QR factorization computed by `qr` (also Householder triangularization);

5. $x = A \backslash b$ in MATLAB (also based on QR factorization);

6. SVD, using MATLAB's `svd`.

The calculations above will produce six lists of 12 coefficients. For each list, mark those digits that appear to be wrong. Comment on the differences you observe. Does the normal equation exhibit particularly serious instability?