AMS 526 Homework 3, Fall 2008

Due: Thursday 10/23 in class.

1. (20 points) Suppose $\boldsymbol{A}$ is a 202×202 matrix with $\|\boldsymbol{A}\|_2 = 100$ and $\|\boldsymbol{A}\|_F = 101$. Give the sharpest possible lower bound on the condition number of $\boldsymbol{A}$ using 2-norm.

2. (20 points) Let $\boldsymbol{A}$ be a 100×100 lower bi-diagonal matrix whose diagonal entries are 0.5 and all sub-diagonal elements are -1, i.e.,

$$\boldsymbol{A} = \begin{bmatrix} 0.5 & & & & & & \\ -1 & 0.5 & & & & & \\ & -1 & 0.5 & & & & \\ & & \ddots & \ddots & & & \\ & & & -1 & 0.5 & & \\ & & & & -1 & 0.5 \end{bmatrix}.$$

   (a) Show that one component of $\boldsymbol{A}^{-1}$ is $\alpha = 2^{100}$.

   (b) Show that $\kappa(\boldsymbol{A}) > 2^{101}$ in ∞-norm.

3. (20 points) Each of the following problems describes an algorithm implemented on a computer satisfying the two axioms of floating point numbers (axioms (13.5) and (13.7) in the textbook). For each problem, answer whether the algorithm is backward stable, stable but not backward stable, or unstable. Prove your assertion or give a reasonably convincing argument.

   (a) Data: $x \in \mathbb{C}$. Solution: x^2, computed as $x \otimes x$, where $\otimes$ denotes floating-point multiplication.

   (b) Data: none. Solution: e, computed by summing $\sum_{k=0}^{\infty} 1/k!$ from left to right using floating-point multiplication $\otimes$ and floating-point addition $\oplus$, stopping when a summand is reached of magnitude $< \epsilon_{\text{machine}}$.

4. (40 points) In this exercise, you need to implement the Householder QR factorization (Algorithm 10.1), the implicit calculation of product $\hat{\boldsymbol{Q}}^* \boldsymbol{b}$ (by modifying Algorithm 10.2), and back substitution (Algorithm 17.1). Then, use these algorithms as building blocks for Algorithm 11.2 to solve the following least squares problem arising from polynomial fitting. We fit a polynomial of degree $n-1$,

$$p_{n-1}(t) = x_1 + x_2 t + x_3 t^2 + \cdots + x_n t^{n-1},$$

   to m data points (t_i, y_i), $m > n$. Let $t_i = (i-1)/(m-1)$, $i = 1, \ldots m$, so that the data points are equally spaced on the interval $[0, 1]$. We will generate the corresponding values y_i by first choosing values for the x_j, say $x_j = 1$, $j = 1, \ldots, n$, and evaluating the resulting polynomial to obtain $y_i = p_{n-1}(t_i)$, $i = 1, \ldots, m$. Our objective is to see whether we can recover the x_j that are used to generate y_i, and measure the error as the difference between the computed x_j and the exact x_j.

   In addition, use the modified Gram-Schmidt procedure in Algorithm 11.2 to solve the same least squares problem. Compare the two methods in terms of accuracy and efficiency.
   Implement your code by following the template files posted on the class website. Read the README file in the template file. Choose $n = 3, 4, \ldots, 15$ and $m = 2n$, and plot the 2-norm errors using Householder and modified Gram-Schmidt algorithms. Submit your modified C code, the plots, and your conclusions of the comparative study.