

CS 510 Homework - due 3/10/08

I. Consider the nonlinear system:

$$f_1(x, y) = x^2 - y^2 - 1 = 0$$

$$f_2(x, y) = xy = 0$$

with solution $(\bar{x}, \bar{y}) = (1, 0)$.

(i) For what α , if any, will the fixed point iteration

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \end{bmatrix} = \begin{bmatrix} x_k \\ y_k \end{bmatrix} + \alpha \begin{bmatrix} x_k^2 - y_k^2 - 1 \\ x_k y_k \end{bmatrix}$$

be locally convergent to $(1, 0)$?

(ii) With a sufficiently good initial iterate, will Newton's method converge quadratically to $(1, 0)$?

(iii) Apply one iteration of Newton's method to the given system with initial iterate $(x_0, y_0) = (1.1, .1)$. Feel free to use Matlab, for example, to solve the resulting 2×2 linear system.

II. Newton's method for an $n \times n$ nonlinear system $\mathbf{f}(\mathbf{x}) = 0$ requires evaluation of the Jacobian

$$\mathbf{f}'(\mathbf{x}) = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & & \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{pmatrix}.$$

This is commonly done by numerical differentiation:

$$\frac{\partial f_i}{\partial x_j} \approx \frac{f_i(\mathbf{x} + h_j \mathbf{e}^{(j)}) - f_i(\mathbf{x})}{h_j},$$

where $\mathbf{e}^{(j)}$ is the j^{th} unit vector.

Suppose you have a subroutine which evaluates all n components of $\mathbf{f}(\mathbf{x})$ for an arbitrary vector \mathbf{x} . How many times must this subroutine be called to approximate a Jacobian that:

a) is dense (i.e., full)?

b) is tridiagonal?

c) has bandwidth w ?

Explain briefly. In general, efficient approximation of a sparse Jacobian via numerical differentiation can be viewed as a graph coloring problem.

III. Using Gaussian elimination with partial pivoting, obtain a $PA = LU$ decomposition of

$$A = \begin{bmatrix} 1 & -1 & -2 \\ 2 & -2 & -1 \\ 1 & 0 & 2 \end{bmatrix}$$

and use it to solve

$$Ax = \begin{bmatrix} 1 \\ 5 \\ 4 \end{bmatrix}.$$

IV. Obtain a Cholesky ($\tilde{L}\tilde{L}^T$) factorization of

$$A = \begin{bmatrix} 4 & -2 & & & \\ -2 & 5 & -2 & & \\ & -2 & 5 & 4 & \\ & & 4 & 5 & 2 \\ & & & 2 & 5 \end{bmatrix}.$$

V. Assume A is a nonsingular $n \times n$ matrix...

(i) Suppose A^{-1} is computed by applying Gaussian elimination to A , then using the resulting LU decomposition to solve $Ax^{(k)} = e^{(k)}$, $k = 1, \dots, n$. How many operations will this take: (a) if the zero structure of the $e^{(k)}$'s is not taken into account, (b) if the zero structure of the $e^{(k)}$'s is taken into account.

(ii) Suppose $C = A^{-1}B$ is to be computed, where B is $n \times n$. How many operations will this take (a) if this is done by first computing A^{-1} , then postmultiplying by B , (b) if the columns $c_j = A^{-1}b_j$ of C ($b_j =$ column j of B) are computed by solving $Ac_j = b_j$, $j = 1, \dots, n$.

VI. Suppose we've computed the solution x of $Ax = b$, saving the LU factors of A , and that we need to solve a new system $A'x' = b$ where $A' = A - uv^T$. Here u, v are vectors; thus uv^T is a 'rank one' matrix - it has only one independent row (or column).

(i) Show that x' can be written in the form

$$x' = x + \alpha A^{-1}u$$

where α , a scalar, is given by

$$\alpha = \frac{v^T x}{1 - v^T A^{-1}u}.$$

(ii) Suppose A' differs from A only in its k^{th} column. How would you choose u, v ? How many operations are needed to compute x' ?

VII. Show that the 1-norm of a matrix $A_{n \times n}$ is given by

$$\|A\|_1 = \max_{1 \leq j \leq n} \sum_{i=1}^n |a_{i,j}|.$$

For what vector $x \neq 0$ is $\|Ax\|_1 = \|A\|_1 \|x\|_1$?

VIII. Consider the linear system $Ax = b$ in question III, for which

$$A^{-1} = \frac{1}{3} \begin{pmatrix} 4 & -2 & 3 \\ 5 & -4 & 3 \\ -2 & 1 & 0 \end{pmatrix}.$$

(i) What are $\|A\|_\infty$, $\|A^{-1}\|_\infty$, and $\kappa_\infty(A)$?

(ii) Suppose \hat{x} is an approximate solution to $Ax = b$ such that $\|A\hat{x} - b\|_\infty \leq .001$. Bound the absolute and relative error in \hat{x} ($\|\hat{x} - x\|_\infty$ and $\|\hat{x} - x\|_\infty / \|x\|_\infty$).

IX. Suppose $A_{n \times n}$ is a real symmetric matrix with (real) eigenvalues $|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_n| > 0$ and corresponding orthonormal eigenvectors $v^{(1)}, v^{(2)}, \dots, v^{(n)}$:

$$Av^{(i)} = \lambda_i v^{(i)}, \quad (v^{(i)}, v^{(j)}) = \delta_{i,j} \equiv \begin{cases} 1, & i = j \\ 0, & i \neq j. \end{cases}$$

(i) What are $\|A\|_2$, $\|A^{-1}\|_2$, and $\kappa_2(A)$?

(ii) Find right-hand side vectors b and $b + \delta b$ such that equality is achieved in the condition number bound as applied to $Ax = b$ and $A(x + \delta x) = b + \delta b$, i.e.,

$$\frac{\|\delta x\|_2}{\|x\|_2} = \kappa_2(A) \frac{\|\delta b\|_2}{\|b\|_2}.$$

X. Consider the problem of approximating a function $f(x)$ over $x \in [a, b]$ in the least squares sense by a linear combination of the form

$$y(x) = \sum_{j=1}^n c_j \phi_j(x).$$

Here $\{\phi_j(x)\}_{j=1}^n$ are user-defined “basis functions,” chosen according to the type of approximation sought. For example, to generate a polynomial of degree p , we might use “monomial” basis functions

$$\phi_j(x) = x^{j-1}, \quad j = 1, \dots, n, \quad n = p + 1,$$

(an obvious - but very bad - way to represent a polynomial). We seek the set of coefficients c_j which minimizes

$$E = \int_a^b (y(x) - f(x))^2 dx$$

over all functions $y(x)$ of the given form. Substituting for $y(x)$ in the above, and setting the partial derivatives $\frac{\partial E}{\partial c_i} = 0$, $i = 1, \dots, n$, we obtain the “normal equations” for our least squares approximation problem:

$$\sum_{j=1}^n \left(\int_a^b \phi_i(x) \phi_j(x) dx \right) c_j = \int_a^b f(x) \phi_i(x) dx, \quad i = 1, \dots, n,$$

which we write in matrix form as $Ac = b$.

For the case where the approximation interval is $[0, 1]$ and the monomial basis is used, we get as our coefficient matrix A the “Hilbert” matrix:

$$a_{i,j} = \frac{1}{i+j-1}, \quad 1 \leq i \leq n, \quad 1 \leq j \leq n,$$

a classic ill-conditioned matrix. Compute least squares approximation of $f(x) = 1 + x$ by polynomials of degree 3, 6, 9, 12 using the monomial basis. For each approximation, print the corresponding computed coefficients \hat{c} , the relative error $\frac{\|\hat{c} - c\|_\infty}{\|c\|_\infty}$, and the ∞ -norm condition number of A . Use Matlab, and the very convenient commands *hilb*, *norm*, *cond*, *diary*; also `'c = A\b'` to solve the system, `'format long'` to print your results to full machine precision. Type `'help hilb'`, `'help norm'`, etc. to find out how to use these commands.

Interpret your results. To what extent is $\kappa(A)$ a good indicator of the number of digits lost in your computed solutions?