Midterm Exam 2 (15 points + 1 bonus point)

Name: ID:

CS3330 Scientific Computing, Instructor: Cheng-Hsin Hsu

Department of Computing Science, National Tsing Hua University, Taiwan

8:00 a.m. – 9:50 a.m., December 13th, 2017

- **You need to show your work (the detailed procedures) to get the points.**
- **There will be absolutely no time extension, since the questions are fairly basic.**
- **You are allowed to bring an A4 cheat sheet with handwritten formulas/notes. Printed (by computers or xerox machines) cheat sheets, or cheat sheets with information about specific questions/answers are not allowed. The TA will check your cheat sheets during the exam, and any cheat sheets violating the above rules will be removed from the classroom.**

1) Answer the following questions (2 pts + 1 Bonus pt) 0.5 points each

   a) If the following numbers represent the magnitudes of errors at successive iterations, what is the convergence rate of individual iterative algorithms? Hint: use the terms linear, superlinear, and quadratic; for linear convergence rate, please also give the constant C.

      i) $10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}$ linear 10^{-1}

      ii) $10^{-2}, 10^{-4}, 10^{-6}, 10^{-8}$ linear 10^{-2}

      iii) $10^{-2}, 10^{-3}, 10^{-5}, 10^{-8}$ superlinear

      iv) $10^{-2}, 10^{-4}, 10^{-8}, 10^{-16}$ quadratic

b) Compared to the Newton's method, what are the drawbacks of the Muller's method?

no real roots
not easy to comput
next iteration chosen is hard

c) Under typical setups of nonlinear equation systems, compare Newton's method and Secant updating method in terms of: (i) quality improvement per step, (ii) number of steps before convergence, and (iii) overall time consumption.

(i) newton is ~~better~~ better
(ii) newton is fewer
(iii) newton is slower

d) Givens QR factorization is more complex than Householder QR factorization. When do you think we should consider Givens?

Sparse and special pattern

e) For any given *orthogonal* square matrix **Q**, what can we say about **Qv** and **v** for an arbitrary vector **v**?

norm of v wouldn't change

f) Does Cubic Spline interpolation give unique solution? Why?

No, because the # of equations is fewer than the # of parameter

2) (2 pts) Express the Newton iteration for solving each of the following systems of nonlinear equations.

a) $\begin{cases} x_1^2 + x_2^2 = 1, \\ x_1^2 - x_2 = 0. \end{cases}$

b) $\begin{cases} x_1^3 - x_2^2 = 0, \\ x_1 + x_1^2 x_2 = 2. \end{cases}$

The newton's iteration : $X_{k+1} = X_k + S_k$, where S_k is determined by solving the linear system $J_f(X_k) S_k = -f(X_k)$

(a)

$$J_f(X_k) S_k = \begin{bmatrix} 2X_1 & 2X_2 \\ 2X_1 & -1 \end{bmatrix} \begin{bmatrix} S_1 \\ S_2 \end{bmatrix}$$

$$= -\begin{bmatrix} X_1^2 + X_2^2 - 1 \\ X_1^2 - X_2 \end{bmatrix} = -f(X_k)$$

(b)

$$J_f(X_k) S_k = \begin{bmatrix} 3X_1^2 & -2X_2 \\ 1+2X_1X_2 & X_1^2 \end{bmatrix} \begin{bmatrix} S_1 \\ S_2 \end{bmatrix}$$

$$= -\begin{bmatrix} X_1^3 - X_2^2 \\ X_1 + X_1^2 X_2 - 2 \end{bmatrix} = -f(X_k)$$

3) (3 pts) On a computer with no floating-point division capability, we have to use multiplication by the reciprocal of the divisor to emulate divisions. Apply Newton's method to produce an iterative scheme for approximating the reciprocal of a number $y > 0$, i.e., to solve the equation $f(x) = (1/x) - y = 0$, given y. Remember that, your formula cannot contain any divisions because your computer doesn't support it.

The derivative of $f(x) = x^{-1} - y$ is $f'(x) = -x^{-2}$.

The newton iteration for solving $f(x) = 0$ is thus

$$x_{k+1} = x_k - \frac{x_k^{-1} - y}{-x_k^{-2}} = x_k + \boxed{x_k^2 (x_k^{-1} - y)}$$

$$= \underline{2x_k - x_k^2 y}$$, which doesn't need any divisions.

4) (3 pts) Show that if $\mathbf{A} = \mathbf{QR}$ is the QR factorization for matrix $\mathbf{A}$, then $\mathbf{R}$ is *not* singular, even in floating-point arithmetic. Notice that we made an observation that the cross-product matrix $\mathbf{A}^T\mathbf{A}$ is exactly singular in floating-point arithmetic if

$$\mathbf{A} = \begin{bmatrix} 1 & 1 \\ \epsilon & 0 \\ 0 & \epsilon \end{bmatrix},$$

where ϵ is a positive number smaller than $\sqrt{\epsilon_{\text{mach}}}$ in a given floating-point system.

We compute R in floating points using Given rotations.

To zero the (2,1) entry, we let $t = \varepsilon$, $c = 1$, and $s = \varepsilon$,

this gives

$$G_1 A = \begin{bmatrix} 1 & \varepsilon & 0 \\ -\varepsilon & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ \varepsilon & 0 \\ 0 & \varepsilon \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & -\varepsilon \\ 0 & \varepsilon \end{bmatrix}$$

To get rid of (3,2) entry, we let $t = 1$, $c = 1/\sqrt{2}$, and $s = -1/\sqrt{2}$, which gives

$$G_2 G_1 A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1/\sqrt{2} & -1/\sqrt{2} \\ 0 & 1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 0 & -\varepsilon \\ 0 & \varepsilon \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & -\varepsilon\sqrt{2} \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} R \\ 0 \end{bmatrix}$$

The upper 2x2 R is nonsingular since $\varepsilon \neq 0$

5) (3 pts) Show that if the vector $\mathbf{v} \neq \mathbf{o}$, then the matrix

$$\mathbf{H} = \mathbf{I} - 2\frac{\mathbf{v}\mathbf{v}^T}{\mathbf{v}^T\mathbf{v}}$$

is orthogonal and symmetric.

Symmetric: $H^T = I^T - 2\frac{(vv^T)^T}{(v^Tv)^T} = I^T - 2\frac{vv^T}{v^Tv} = \underline{H}$

Orthogonal: $H^TH = H^2 = \left(I - 2\frac{vv^T}{v^Tv}\right)^2 = I - 4\frac{vv^T}{v^Tv} + 4\frac{v(v^Tv)v^T}{(v^Tv)^2}$

$= \underline{I}.$

6) (2 pts) Consider the following sample points:

t	1	2	3	4
y	11	29	65	125

a) Determine the polynomial interpolant to the data using the monomial basis.

b) Compute the Newton polynomial interpolant using any one of the three approaches discussed in the lecture (triangular matrix, incremental interpolation, and divided differences). Do you get the same polynomial compared to the one in part (a)?

a) The linear system:

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 2 & 4 & 8 \\ 1 & 3 & 9 & 27 \\ 1 & 4 & 16 & 64 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 11 \\ 29 \\ 65 \\ 125 \end{bmatrix}$$

Solving this using Gaussian elimination gives $\begin{bmatrix} 5 \\ 2 \\ 3 \\ 1 \end{bmatrix}$

$$\Rightarrow \quad p(t) = 5 + 2t + 3t^2 + t^3$$

(b) - Triangular

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 2 & 2 & 0 \\ 1 & 3 & 6 & 6 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 11 \\ 29 \\ 65 \\ 125 \end{bmatrix}$$

- Incremental interpolation

$$p_0(t) = y_1 = 11$$

$$p_1(t) = p_0(t) + x_2 \pi_1(t) = 11 + 18(t-1)$$

$\vdots$

- Divided difference

$$f[t_1, t_2] = \frac{f[t_2] - f[t_1]}{t_2 - t_1} = 18 \; \ldots$$

$$p(3) = 11 + 18(t-1) + 9(t-1)(t-2) + (t-1)(t-2)(t-3)$$

$\Rightarrow$ Same polynomial.