CS421 Assignment 5: 1997

Due Date: November 19, 1997

1. Write a MATLAB program for solving one nonlinear equation in one unknown. Implement a hybrid strategy. For a local step, use Newton's method. For globalization, use a backtracking strategy. Run your program on:

   (a) $f(x) = \sin(x) - \cos(2x),\ x_0 = 1$

   (b) $f(x) = x^3 - 7x^2 + 11x - 5,\ x_0 = 2, 7$

   (c) $f(x) = \sin(x) - \cos(x),\ x_0 = 1, 2.7$

   (d) $f(x) = x^4 - 12x^3 + 47x^2 - 60x + 24,\ x_0 = 0, 2, 4$

   You should use Matlab function **feval** to name the function of each zero finding problem. Choose a sufficiently small tolerance to observe the local convergence behavior, e.g., 10^{-12}. Print out values x_k and $f(x_k)$ at each iteration k. Print out each solution you compute. What rate of convergence do you observe in each case? It won't be always quadratic. Why?

2. (Heath 5.13) Assume that x_* is a fixed point of a function $g : \Re \to \Re$, $g(x)$ is twice continuously differentiable in $[x_* - \rho, x_* + \rho]$ for some $\rho > 0$ and $g'(x_*) = 0$. Prove that, if started close enough to x_*, the iterations $\{x_k\}$ generated by the fixed-point iteration scheme $x_{k+1} = g(x_k)$ converges and the convergence rate is at least quadratic.

3. (Heath 5.18) Newton's method can be used to compute an eigenvalue λ and corresponding eigenvector x of an n-by-n matrix A. If we define the function $f : \Re^{n+1} \to \Re^{n+1}$ by

$$f(x, \lambda) = \begin{bmatrix} Ax - \lambda x \\ x^T x - 1 \end{bmatrix}$$

then $f(x, \lambda) = 0$ precisely when λ is an eigenvalue and x is a corresponding normalized eigenvector. Since

$$J_f(x, \lambda) = \begin{bmatrix} A - \lambda I & -x \\ 2x^T & 0 \end{bmatrix},$$

Newton's method for solving this equation has the form

$$\begin{bmatrix} x_{k+1} \\ \lambda_{k+1} \end{bmatrix} = \begin{bmatrix} x_k \\ \lambda_k \end{bmatrix} + \begin{bmatrix} s_k \\ \delta_k \end{bmatrix}$$

where $[s_k \ \delta_k]^T$ is the solution to the linear system

$$\begin{bmatrix} A - \lambda_k I & -x_k \\ 2x_k^T & 0 \end{bmatrix} \begin{bmatrix} s_k \\ \delta_k \end{bmatrix} = - \begin{bmatrix} Ax_k - \lambda_k x_k \\ x_x^T x_k - 1 \end{bmatrix}$$

Write a MATLAB program to compute an eigenvalue-eigenvector pair of a given input matrix A using this Newton iteration scheme. A reasonable starting guess is to take x_0 to be an arbitrary normalized nonzero vector (i.e., $x_0^T x_0 = 1$) and take $\lambda_0 = x_0^T A x_0$ (why?). Test your program on a few randomly generated matrices and compare its accuracy and efficiency with the power method. Note, however, that Newton's method does not necessarily converge to the dominant eigenvalue.