Applied Math 5610 **Homework 5** **Due: February 20, 2019**
Bengt Fornberg

1. If the vector $\underline{x}$ is equal to an eigenvector $\underline{v}$ of a symmetric (real-valued) matrix A, then the Rayleigh quotient $r(\underline{x}) = \dfrac{\underline{x}^T A \underline{x}}{\underline{x}^T \underline{x}}$ evaluates to its matching eigenvalue λ. If $\underline{x}$ is not exactly equal to $\underline{v}$, but somewhat close to it, $r(\underline{x})$ becomes a good approximation to λ.

   a. Let $\nabla r(\underline{x})$ denote the *gradient* of the Rayleigh quotient with respect to the components of $\underline{x}$. Show that is equals to $\nabla r(\underline{x}) = \dfrac{2}{\underline{x}^T \underline{x}}(A\underline{x} - r(x)\underline{x})$.

   b. From the result in Part (a), deduce that if $\underline{x}$ is near to $\underline{v}$, then $r(\underline{x})$ is within $O(\| \underline{x} - \underline{v} \|^2)$ of λ.

      Hint: Consider the Taylor expansion of $r(\underline{x})$ around the true eigenvector $\underline{v}$.

2. In Problem 2 b of Homework 3, you chose a random initial guess for λ_0 and $\underline{x}_0$ and then iterated until convergence, using a Newton-based iteration. Solve this same problem (with the same Hilbert matrix), using instead the Rayleigh quotient iteration for eigenvalue/eigenvector. How does the convergence rate you now observe compare with the quadratic convergence you observed before?

3. Matlab's symbolic tool box contains a routine 'pade' which returns Padé approximations for specified values of the numerator and denominator degrees *N* and *M*, respectively, of a given function. For example, using this routine, create an extended version of the table in "Application 1" in the Lecture Notes: "Padé approximations" (on the class web page). Evaluate the Padé approximant for all values $0 \le M, N \le 6$ (i.e. not just for $M = 0$ and $M = N$ as in that table), and display log|error| as a surface plot over a *M*, *N*-plane. In what areas of the *M*, *N*-plane are the errors the lowest?

   Note: When you request a Padé approximation of order *M*, *N*, of the function $\dfrac{\log(1+x)}{x}$, the routine 'pade' will utilize no other information than the function's $M + N + 1$ leading Taylor coefficients.

4. Use both the Mathematica and the Matlab algorithms in the Lecture Notes: "FD weights" (on the class web page) to produce tables of weights matching the ones shown in its Tables 1.1 and 1.2. Calculate the weights for accuracy orders 2, ... ,10 (omitting the line from the tables denoted "Limit").

   In the case of Mathematica algorithm, give weights as rational numbers. Display them also as floating point numbers, and check that the two approaches match (to some moderate level of accuracy).