Applied Math 5610 **Homework 4** **Due: February 13, 2019**
Bengt Fornberg

1. Explain why Jacobi's method for a symmetric matrix A can be expected to give quadratic convergence. You can for simplicity assume that all the eigenvalues to A are distinct.

   <u>Hint:</u> Suppose you have already iterated sufficiently far so that all off diagonal elements are of size $O(\varepsilon)$ (where ε is a small number). One 'sweep' consists of looping once over the successive off-diagonal entries, bringing one at a time to zero. This will 'destroy' already introduced zeros, but show they during the rest of the sweep can't come back any larger size than $O(\varepsilon^2)$.

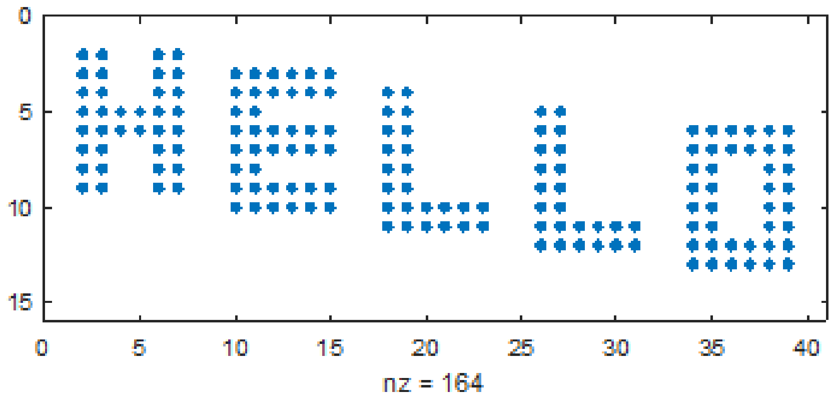
2. Solve numerically the same problem as in the previous homework, Problem 2 (b), use the same initial guess as then for $\underline{x}_0$, but use the Rayleigh quotient iteration rather than the Newton iteration. Compare the two convergence rates.

   <u>Note:</u> While you will see an extremely rapid convergence, one should remember that the Newton iteration is not as limited (to just regular eigenvalue problems for symmetric matrices).

3. The following Matlab program

```
A = zeros(15,40);
a=['                                        ';...
   ' xx  xx                                 ';...
   ' xx  xx  xxxxxx                         ';...
   ' xx  xx  xxxxxx  xx                     ';...
   ' xxxxxx  xx      xx      xx             ';...
   ' xxxxxx  xxxxxx  xx      xx      xxxxxx ';...
   ' xx  xx  xxxxxx  xx      xx      xxxxxx ';...
   ' xx  xx  xx      xx      xx      xx  xx ';...
   ' xx  xx  xxxxxx  xx      xx      xx  xx ';...
   '         xxxxxx  xxxxxx  xx      xx  xx ';...
   '                 xxxxxx  xxxxxx  xx  xx ';...
   '                         xxxxxx  xxxxxx ';...
   '                                 xxxxxx ';...
   '                                        ';...
   '                                        '];
A = 1.0*(a~=' ');
spy(A)
```

sets up a matrix A of size 15×40 containing only ones and zeros, and then displays its non-zero pattern as



a. Call Matlab's svd to compute the singular values of A and print the result (use 'format long'). What is the rank of A?

b. For each $k = 1, 2, \ldots,$ rank(A), graphically display the matrix which is the best rank(k) approximation to A in the 2-norm. Use the subplot command, so you get all these illustrations collected together in a single figure.

4. a. Find, by hand calculation, the least squares solution to the overdetermined linear system

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}.$$

b. Find (by any suitable means) the vector $\underline{x}$ that minimizes the quantity $E^2 = b_1^2 + 4b_2^2 + 25b_3^2 + 9b_4^2$, when it holds that

$$\begin{bmatrix} 1 & 3 \\ 6 & -1 \\ 4 & 0 \\ 2 & 7 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} - \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix}.$$

<u>Hint:</u> When solving a square linear system of equations $A\underline{x} = \underline{b}$, multiplying the different equations with different constants does not change the solution. This is no longer the case when finding least squares solutions to an overdetermined system. Exploit this observation.