MA3227 Numerical Analysis II

Assignment 1

Simon Etter, 2019/2020
Deadline: 19 February 2020, 12.00 (noon)
Total marks: 10

1 Fill path theorem [4 marks]

Consider the matrix

$$A = \left(\begin{array}{c|c|c|c|c|c|c}
1 & \bullet & \bullet & & & & \\ \hline
\bullet & 2 & & \bullet & \bullet & & \\ \hline
\bullet & & 3 & & & \bullet & \bullet \\ \hline
 & \bullet & & 4 & & & \\ \hline
 & \bullet & & & 5 & & \\ \hline
 & & \bullet & & & 6 & \\ \hline
 & & \bullet & & & & 7
\end{array}\right)$$

where each number and $\bullet$ represents a nonzero entry.

1. Draw the graph $G(A)$ associated with A. Make sure to clearly label the vertices in the graph.

   Hints.

   - Note that we have $i \to j \in E(A) \iff j \to i \in E(A)$. You may simplify your drawing by drawing only a single undirected edge for each pair of directed edges.
   - The graph of A belongs to a particular class of graphs. The following questions may be easier to answer if you recognise this class and (re-)draw the graph accordingly.

2. Determine the structural nonzero pattern of the LU factorisation of A. State your result as a single matrix with the numbers 1 to 7 on the diagonal, a $\bullet$ for a nonzero entry in A and a x for a fill-in entry in L or U.

   Briefly explain why the following entries are structurally zero or nonzero:

   $$L[3,2], \qquad L[6,2], \qquad L[4,3].$$

3. Determine a permutation $\pi : \{1, \dots 7\} \to \{1, \dots, 7\}$ such that the LU factorisation of the permuted matrix $B[i,j] := A[\pi(i), \pi(j)]$ does not produce any fill-in. State your result as a table of the form

i	1	2	3	4	5	6	7
$\pi(i)$							

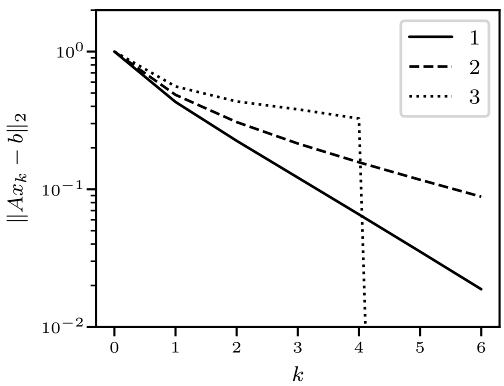
   .

   You do not have to motivate your result, but you are welcome to do so if you want me to check your reasoning.

2 Convergence of GMRES [2 marks]

Match the following matrices with their GMRES convergence history. Motivate your answer.

```
A = Diagonal(LinRange(1,10,100))
B = Diagonal(LinRange(1,40,100))
C = Diagonal(repeat(LinRange(1,40,5),20))
```



3 Conjugate gradients [4 marks]

Let $A \in \mathbb{R}^{N\times N}$ be a symmetric positive definite matrix and $b \in \mathbb{R}^N$. The conjugate gradients algorithm approximates the solution to $Ax = b$ by

$$x_k = p_{k-1}(A)\, b \qquad \text{where} \qquad p_{k-1} = \underset{p_{k-1}\in\mathcal{P}_{k-1}}{\arg\min}\ \|A p_{k-1}(A)\, b - b\|_{A^{-1}} \qquad .$$

This minimisation problem is solved by the following algorithm.

Algorithm 1 Conjugate gradients

$x_0 = 0, r_0 = b, p_0 = r_0$
for $\ell = 1, \dots, k$ **do**
  $\alpha_\ell = (r_{\ell-1}^T r_{\ell-1})/(p_{\ell-1}^T A p_{\ell-1})$
  $x_\ell = x_{\ell-1} + \alpha_\ell\, p_{\ell-1}$
  $r_\ell = r_{\ell-1} - \alpha_\ell\, A p_{\ell-1}$
  $\beta_\ell = (r_\ell^T r_\ell)/(r_{\ell-1}^T r_{\ell-1})$
  $p_\ell = r_\ell + \beta_\ell\, p_{\ell-1}$
end for

Write a function `conjugate_gradients(A,b,k)` which implements this algorithm. Write tests to verify that your code is correct.

Your tests are part of the assessment, and you will be given zero marks for this entire task if your tests are missing / trivial or your code does not pass the tests. If you have difficulties getting your Julia code to run, contact me before submitting.

Hint. You can get a "random" symmetric positive definite matrix using the following function:

```
function spd_rand(n)
    A = rand(n,n)
    return A'*A
end
```