MA2501 Numeriske metoder

Assignment 2

Supervision: week 4 and 5

As answer to this assignment, you will have to write a **short** document describing briefly what you have done to solve each of the two problems below, and including tables and figures with descriptions for each of these, and with the right amount of detail.

Problem 1

We want to approximate solutions of nonlinear equations in two variables with the Newton method.

- Implement the method giving a generic function $F : \mathbb{R}^2 \to \mathbb{R}^2$ as input, use a symbolic/automatic differentiation tool to compute the Jacobian matrix of F as part of your implementation. As stopping criterion, you should use a maximum number of iterations as well as a tolerance TOL such that the iteration is stopped whenever $\mathbf{x}^k := (x_1^k, x_2^k)$, is such that $\|F(\mathbf{x}^k)\|_2 \leq TOL$ or $\|\mathbf{x}^{k+1} - \mathbf{x}^k\|_2 \leq TOL$. You should submit the code of your Newton method. Please write nice code with some comments and explanations.

- Consider exercise 4.7 in the book by Süli and Mayers. Consider the equation $F(x) = 0$ where $F = (f_1, f_2)$ and

  $$f_1(x_1, x_2) = x_1^2 + x_2^2 - 2, \qquad f_2(x_1, x_2) = x_1 - x_2.$$

  Apply your Newton method to this problem and verify with numerical experiments that your implementation of Newton method is correct and converges to the correct solution as shown in the exercise (i.e. converges to $(1, 1)$ whenever $x_1^0 + x_2^0 > 0$ and to $(-1, -1)$ whenever $x_1^0 + x_2^0 < 0$).

Verify numerically that the convergence is quadratic: find with numerical experiments the value of the constant $\lim_{k\to\infty}\frac{\|\mathbf{x}^{k+1}-\mathbf{x}^k\|_2}{\|\mathbf{x}^{k+1}-\mathbf{x}^k\|_2^2}$. Verify that your code generates approximations such that $x_1^k = x_2^k$, for all k. As evidence that the code is correct, you should submit two plots of $x_1^k = x_2^k$ as a function of k showing convergence to $(1,1)$ and $(-1,-1)$ respectively. Submit also a semilog plot of the norms $\|F(\mathbf{x}^k)\|_2$, $\|\mathbf{x}^{k+1}-\mathbf{x}^k\|_2$ versus the number of iterations.

- **Optional**. Change F to be the function you obtain from $f(z) = z^3 - 1$ defined on the complex plane (i.e. for $z = x + iy$) by separating the real part from and imaginary part. The three solutions are $z = 1$, $z = \frac{-1+i\sqrt{3}}{2}$ and $z = \frac{-1-i\sqrt{3}}{2}$. Using three different colours, plot in the complex plane the regions of initial values z_0 for which Newton method converges to the three different solutions respectively. This should generate fractal pictures. **Do this exercise only if you have already managed to do the rest of the assignment in a way that you think is satisfactory.**

Problem 2

Consider the linear system

$$A\mathbf{u} = \mathbf{f},$$

where $A = (L + \Delta x^2 k^2 I) \in \mathbb{R}^{n^2\times n^2}$, I the $n^2 \times n^2$ identity matrix, $\Delta x = \frac{1}{n}$, $\mathbf{f} \in \mathbb{R}^{n^2}$ and L is the following $n^2 \times n^2$ block matrix

$$L = \begin{pmatrix} B & I_n & & & 0 \\ I_n & \ddots & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & I_n \\ 0 & & & I_n & B \end{pmatrix}, \quad \text{where} \quad B = \begin{pmatrix} -4 & 1 & & & 0 \\ 1 & \ddots & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & 1 \\ 0 & & & 1 & -4 \end{pmatrix} \in \mathbb{R}^{n\times n}$$

and I_n is the $n \times n$ identity matrix. Just to provide you with a minimum of background on this linear system, notice that it arises from the discretization of the Helmoltz partial differential equation on the unit square $\Omega = [0,1] \times [0,1]$, $k \in \mathbb{R}$ is a free parameter (the wave number). The components of the vector $\mathbf{f}$ are obtained from the function

$$f : \Omega \to \mathbb{R}, \quad f(x,y) := \exp(-50((x-\frac{1}{2})^2 + (y-\frac{1}{2})^2)).$$

More precisely, for $\ell = 1, \ldots, n^2$, the component ℓ of $\mathbf{f}$ is $f_\ell := f(x_i, y_j)$ where $\ell = (j-1)\,n + i$ and $x_i = \Delta x \cdot i$ and $y_j = \Delta x \cdot j$.

We will now approximate the solution of this linear system using some different iteration methods of the type

$$\mathbf{u}^{(k+1)} = A_1^{-1}(A_2 \mathbf{u}^{(k)} + \mathbf{f}),$$

where $A = A_1 - A_2$ with $|A_1| \neq 0$. The way we choose the matrices A_1 and A_2 depends upon the method. We will use the following iterative methods to calculate approximate solutions to $\mathbf{u}$:

a) Jacobi ($A_1 = A_d$).

b) Forward Gauss-Seidel ($A_1 = A_d + A_l$).

c) Successive over relaxation ($A_1 = A_d + \omega A_l$, where you can choose the value of ω).

Here, A_d, A_l and A_u are the diagonal, strictly lower-triangular and strictly upper-triangular parts of the matrix A such that $A = A_d + A_l + A_u$. See the note on iterative methods for more details on these methods and how to construct them. Consider the residual vector $\mathbf{r}^k := \mathbf{f} - A\mathbf{u}^k$. Fix $n = 10$ and $k = 1/100$. For each of the above, you should:

i) Compare the convergence of each method. Submit a semi-log plot of the 2-norm of the relative residual $\frac{\|\mathbf{r}^k\|_2}{\|\mathbf{r}^0\|_2}$ versus the number of iterations. Plot all the methods in the same plot for comparison.

ii) Compare the relative time each method takes. Make a table with the results for each method.

iii) Calculate the spectral radius of $A_1^{-1}A_2$ and see how that relates to convergence of the method (Hint: Python's `numpy.linalg.eigvals(A)` function returns the eigenvalues of A). Make a semi-log plot for comparison.

iv) If you're really good: comment on how the values of k and Δx are related and how this effects the convergence of the above methods.