FYS-4096 Computational Physics, exercise 4

Return your solution to project `exercise4` under your GitLab group for this course by Friday 5 AM.

Tag the final version with `final` keyword, and make sure to include a file `problems_solved` in the repository. The `problems_solved`-file should be a comma separated list of problems you have solved.

Problems

1. Solve ‘x‘ (2 XP)

‘$\mathbf{A}\mathbf{x} = \mathbf{b},$‘

where

‘$\mathbf{A} = \begin{bmatrix} 1 & 2 & 3 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 2 & 3 & 0 & \cdots & 0 \\ 2 & 0 & 1 & 2 & \ddots & \ddots & \vdots \\ 0 & 2 & 0 & 1 & \ddots & 3 & 0 \\ 0 & 0 & 2 & 0 & \ddots & 2 & 3 \\ \vdots & \vdots & \ddots & \ddots & \ddots & 1 & 2 \\ 0 & 0 & \cdots & 0 & 2 & 0 & 1 \end{bmatrix} \in \mathbb{R}^{301\times 301}, \text{and}$‘

‘$\mathbf{b} = [0.0, 0.1, 0.2, 0.3, 0.2, 0.1, 0.0, 0.1, \cdots, 0.2, 0.1, 0.0]^{\mathrm{T}}$‘

Write your code to `scripts/problem1.py`.

2. Implementation of Broyden’s quasi-Newton method (4 XP)

Download the package `linalg` from https://www.tut.fi/fys/fys4096 and place it under version control.

This package contains a module `linalg.nonlin_solve` which includes a function `solve_broyden`. However, the package was corrupted and it’s missing the function implementation. And since the Big Boss is a cheapskate there were no backups. So it’s up to you, a brave initiate of the Dark Arts of Computational Physics, to **finish the implementation.**

The package docstrings should contain all the information you need to complete this exercise. You can verify your implementation by passing all the unit tests. You can run the unit-tests with

```
python3 setup.py test
```

3. Solve ‘x‘ numerically (2 XP)

The equation to solve is ‘$\mathbf{f}(\mathbf{x}) = 0$‘ where

‘$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} \sqrt{|x_0 - x_2|} - 0.2 \\ \tanh\left(\sqrt{|x_1|} - x_2^2\right) \\ \cos(x_2) \end{bmatrix}$.‘

You can solve this equation using, e.g., the function you implemented in problem 3. Write the solution to a script `scripts/problem3.py`.

4. Electron energy levels (2 XP)

Intro

Let’s consider a 1D model for hydrogen atom. The potential the electron feels due to the hydrogen’s nucleus is given by

‘$V(x) = -\frac{1}{\sqrt{x^2+1}}$.‘

The *time-independent Schrödinger equation* for the electron wavefunction ‘$\psi(x)$‘ is thus

‘$\left[-\frac{1}{2}\frac{\mathrm{d}^2}{\mathrm{d}x^2} + V(x)\right]\psi(x) = E\psi(x)$.‘

This is an **eigenvalue problem** for the wavefunction ‘$\psi(x)$‘.

By choosing a coordinate grid ‘$x_0, x_0 + \Delta x, \ldots, x_0 + (N-1)\Delta x$‘ like

```
import numpy as np
x = np.linspace(-20,20,500)
```

we can represent the Schrödinger equation as a standard matrix-eigenvalue problem (we’ll go through this step in more detail in March, just take it as granted)

‘$\mathbf{H}\psi = E\psi$,‘

where the wavefunction is represented as a vector ‘$\psi \equiv [\psi(x_0), \psi(x_0 + \Delta x), \ldots, \psi(x_0 + (N-1)\Delta x)]$‘, E is the eigenvalue corresponding to ‘ψ‘, and the Hamiltonian operator becomes the matrix

$$\mathbf{H} = \begin{bmatrix} H_{00} & T_1 & 0 & \cdots & 0 \\ T_1 & H_{11} & T_1 & \ddots & \vdots \\ 0 & T_1 & H_{22} & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & T_1 \\ 0 & \cdots & 0 & T_1 & H_{(N-1)(N-1)} \end{bmatrix},$$

where $T_1 = -\frac{1}{2\Delta x^2}$, and $H_{kk} = \frac{1}{\Delta x^2} + V(x_0 + k\Delta x)$. Note that **H** here has nonzero elements only in the diagonal and the first off-diagonal bands.

What you should do?

1. **Solve the 4 smallest eigenvalues and the corresponding eigenvectors of H.** The smallest eigenvalue should be ≈ -0.6698.
2. **Normalize the eigenvectors so that** $\int_{x_0}^{x_0+(N-1)\Delta x} |\psi(x)|^2 \, \mathrm{d}x = 1$**.**
3. **Visualize the square of each eigenvector,** $|\psi(x)|^2$
4. **What is the physical interpretation of these eigenvalues/eigenvectors/eigenstates?** Write your answer to `README.rst`.

Write your solution code to `scripts/problem4.py`.