Computational assignment 9

due on November 24, 2025

Fluence map optimization with SciPy's optimizer

Nowadays, many software packages exist for constrained and unconstrained optimization. An example is SciPy's built-in optimization solver `scipy.optimize.minimize`. These optimization solvers assume that the optimization problem is formulated in a standardized form as described in the documentation of the solver. In case you encounter other optimization problems in the future, it is useful to know how to use these general optimization routines.

Use `scipy.optimize.minimize` to solve a treatment plan optimization problem of the form

$$\text{minimize} \qquad \sum_i w_i^o \left(d_i - D_i^{max}\right)_+^2 + \sum_i w_i^u \left(D_i^{min} - d_i\right)_+^2 \tag{1}$$

$$\text{subject to} \qquad d_i \leq U_i^{max} \tag{2}$$

$$F_j \geq 0 \tag{3}$$

In words, add the ability to handle constraints on the maximum dose U_i^{max} in selected voxels while minimizing a piece-wise quadratic objective function. The main application in practice would be to enforce a strict maximum dose to the spinal cord (e.g. 20 Gy) while trying to deliver a higher dose (e.g. 35 Gy) to the tumor. To check if things work properly, you can also see what happens if you limit the dose to one of the lungs to a low value (e.g. 5 Gy).

Hints

1. Use the SciPy documentation of `scipy.optimize.minimize` to learn how to use the function.
2. Use the method `SLSQP`, this allows you to define constraints and bounds.
2. Maximum dose constraints represent linear inequality constraints.
3. The non-negativity constraints represent bounds for the optimization variables.
4. In addition to the objective function, also provide the gradient to `scipy.optimize.minimize`. This can be done via the input parameter 'jac'.