Introduction to Programming and Numerical Analysis

Exercise Class 7
Exercise 3

Jonas Theodor Ø. Schmidt

UNIVERSITY OF COPENHAGEN

Today’s Program

- 15:15 – 15:30: Introduction to Problem Sets and Optimization
- 15:30 – 16:00: Work on Problem Set 1
- 16:00 – 16:15: Break
- 16:15 – 16:55: Work on Problem Set 1
- 16:55-17:00: Round off

Introduction to Problem Sets

- Problem sets are challenging
- It is a good idea to locate and replicate code from lectures, previous exercises, or past projects
- Peek at solutions if necessary, but try to replicate the solutions yourself
- The extra problems are good for preparing you for assignments as they encourage creative thinking

Introduction to Optimization

- Objective: Find x find that minimizes $f(x)$

- We are familiar with solving optimization problems by applying analytical methods, utilizing first and second order conditions

- Now, we will learn how to apply numerical methods

- The standard approach is to use minimize - should our aim be maximize, this can be achieved by minimizing the negative of the function

Grid-Search

- Method: Explore various values of x to find one that minimizes $f(x)$

- Advantages:
  - Provides a basic idea of the function's behavior
  - Avoids getting trapped in non-global minima

- Disadvantages:
  - High computational demand, particularly with increasing dimensions
  - Limited to evaluating points within the predefined grid
  - Precision of the solution is directly tied to the fineness of the grid

Solver

- Method: Iteratively searches for the minimum of $f(x)$ by exploring various values of x and refining guesses according to $f(x)$'s evaluation

- Mechanics: Algorithm dictates which x values to test, except for the initial guess which is user-defined

- Advantages:
  - More efficient and less resource-intensive compared to grid search
  - Yields a more accurate solution

- Disadvantages:
  - Outcome may be influenced by the choice of the starting point
  - Risk of not reaching any solution

Constrained optimization

- Approaches
  - Utilize grid search or specific solvers designed for constrained problems
  - Alternatively, modify the objective function to include penalties for constraint violations to use unconstrained optimization

- Advantages of Penalty Methods
  - Directs the solver back within bounds when it strays outside constraints

- Drawbacks of Penalty Methods
  - Potential to create additional local minima, further complicating the search for the global minima

Solvers

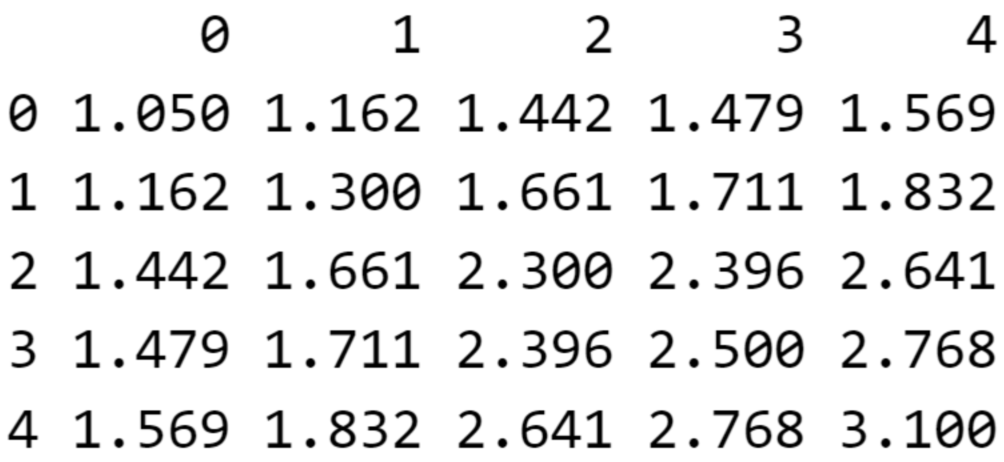
- Choice of solver depends on problem specifics: constrained or unconstrained

- Unconstrained Optimization
  - BFGS: Fast with gradients/Hessians. Ideal for efficient, derivative-informed problems
  - Nelder-Mead: Robust, suitable for derivative-less, complex problems. Slower but reliable

- Constrained Optimization
  - SLSQP: Quick, effective with gradient/Hessian info. Good for problems with direct constraints
  - Penalty Methods: Combine unconstrained solvers with penalties for constraint violations. Versatile for complex constraints

Tips for Problem Set 1, A2:

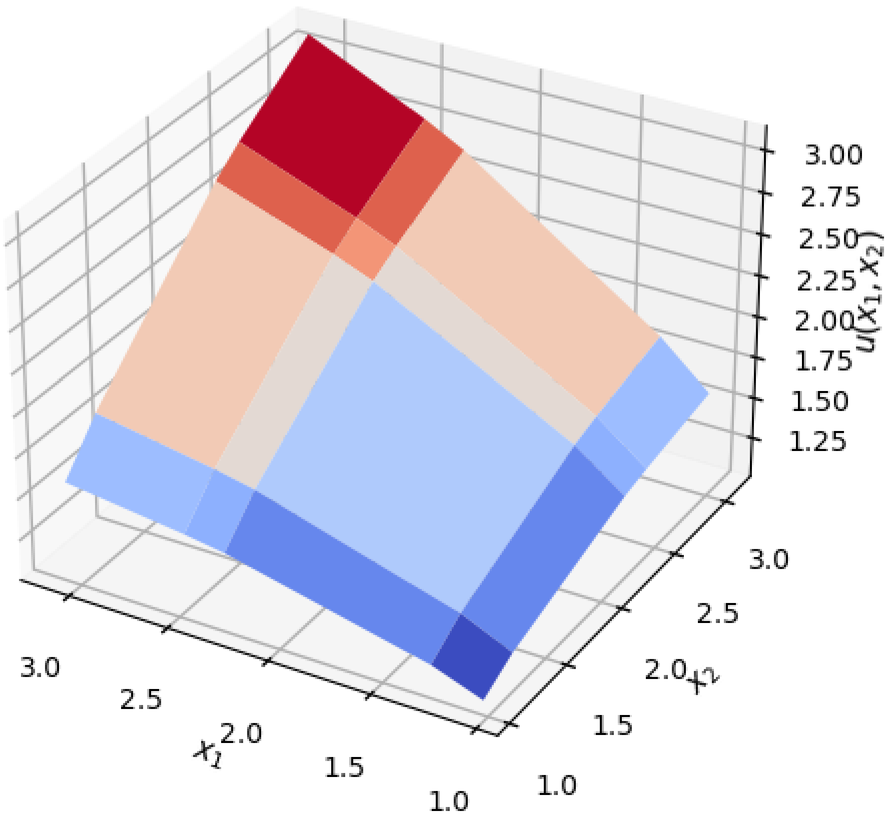
- Output should be this table



```
      0     1     2     3     4
0 1.050 1.162 1.442 1.479 1.569
1 1.162 1.300 1.661 1.711 1.832
2 1.442 1.661 2.300 2.396 2.641
3 1.479 1.711 2.396 2.500 2.768
4 1.569 1.832 2.641 2.768 3.100
```

Tips for Problem Set 1, A3:

- Output should be this figure



Beware

The seaborn-whitegrid style has depreciated

To discover alternative styles, run print(plt.style.available)

Alternatively, you may choose to comment out this line

Break

Questions & comments?