**Numerical Optimization with Python**

**Ex.2 – programming part – final report**

All the files (including plots and console print can be found at the exercise Git repo[[1]](#footnote-1)

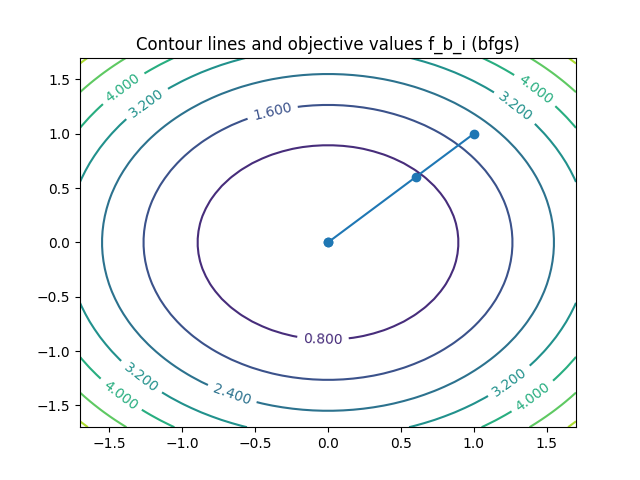
\*Note: The last console print is given in each scenario (per exercise guidelines). However, in cases of extremely small changes in values (e.g. objective value), the last two prints are given.

Functions from exercise 1:

\*For newton decrement in bfgs[[2]](#footnote-2), I used (appears in “My consts” unconstrained\_min.py)

b)

i. BFGS

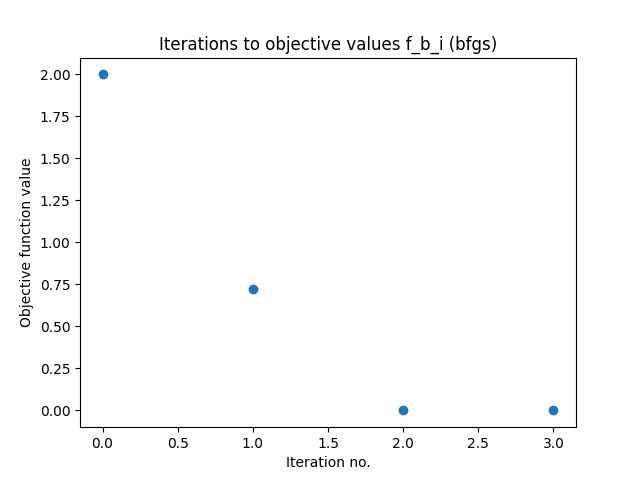


**Last print to console:**

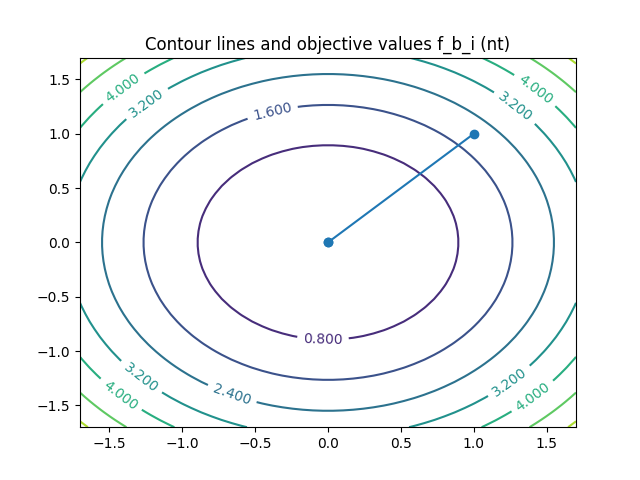
Iteration number: 3 current location: [-5.55111512e-17 5.55111512e-17] current obj val: 6.162975822039155e-33 current step length: 1.7554167342883506e-16 current change in objective function value: 6.162975822039155e-33 (bfgs)

Quad function f\_b\_i (dir method: bfgs) result: Success

Iterations to objective function values:



1. Newton



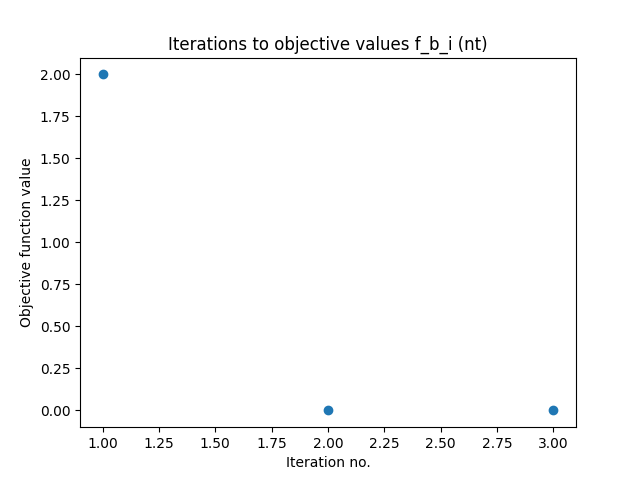
**Last print to console:**

Iteration number: 1 current location: [0. 0.] current obj val: 0.0 current step length: 1.4142135623730951 current change in objective function value: 2.0 (nt)

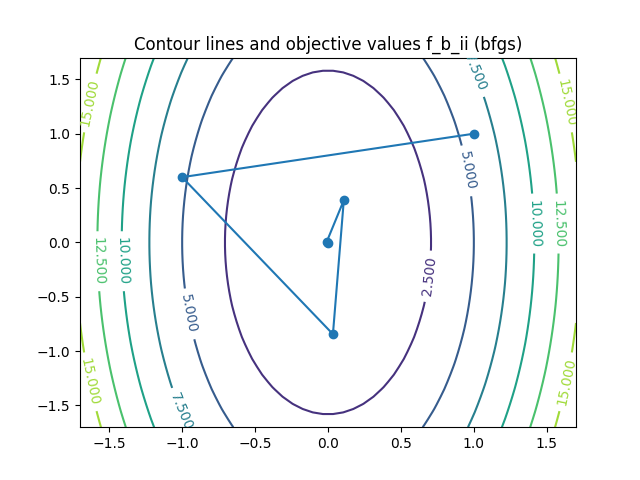
Iteration number: 2 current location: [0. 0.] current obj val: 0.0 current step length: 0.0 current change in objective function value: 0.0 (nt)

Quad function f\_b\_i (dir method: nt) result: Success

Iterations to objective function values:



1. BFGS

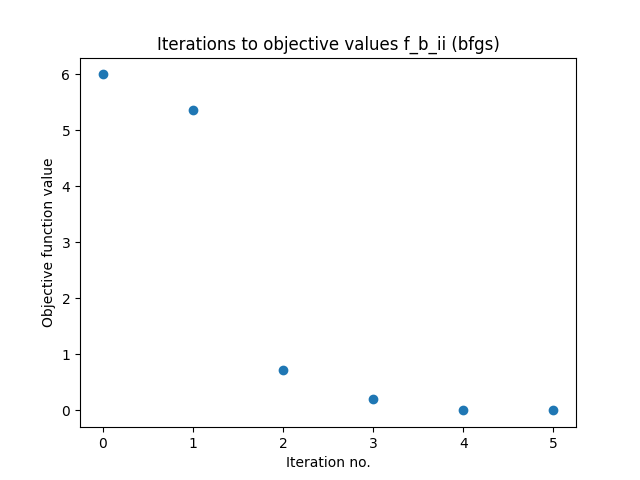


**Last print to console:**

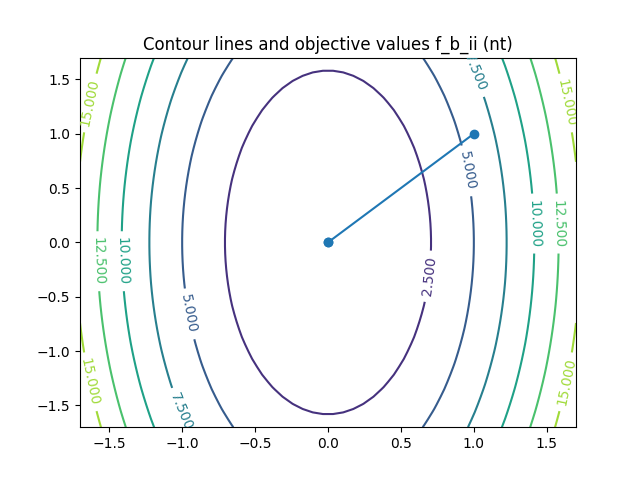
Iteration number: 5 current location: [ 0.00064771 -0.00099179] current obj val: 3.0813034946374527e-06 current step length: 0.01089517146881754 current change in objective function value: 0.00045974580951391837 (bfgs)

Quad function f\_b\_ii (dir method: bfgs) result: Success

Iterations to objective function values:



ii. Newton



**Last print to console:**

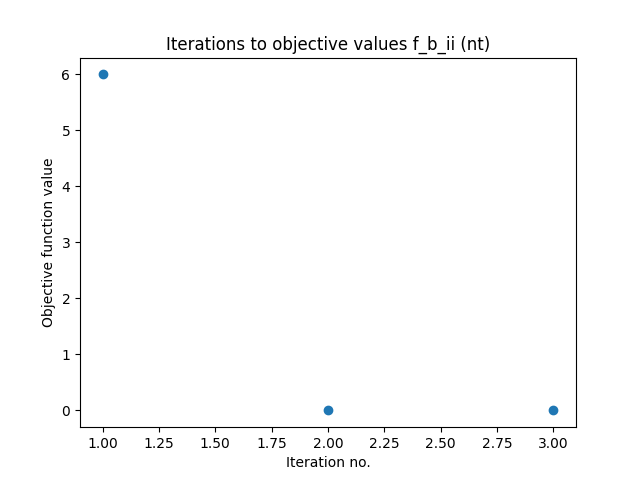
Iteration number: 0 current location: [1 1] current obj val: 6.0 current step length: nan current change in objective function value: nan (nt)

Iteration number: 1 current location: [0. 0.] current obj val: 0.0 current step length: 1.4142135623730951 current change in objective function value: 6.0 (nt)

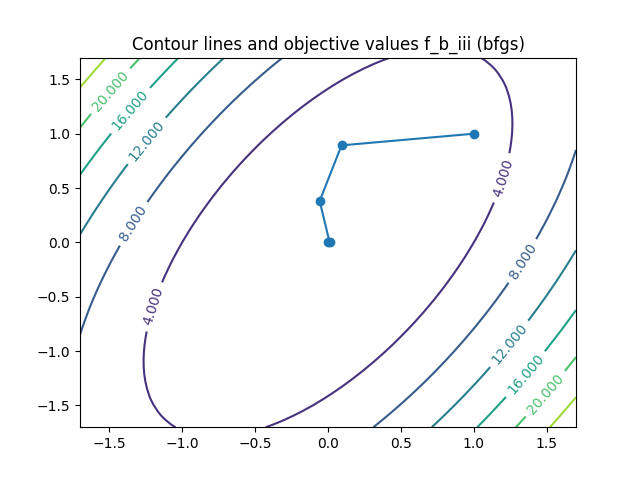
Iteration number: 2 current location: [0. 0.] current obj val: 0.0 current step length: 0.0 current change in objective function value: 0.0 (nt)

Quad function f\_b\_ii (dir method: nt) result: Success

Iterations to objective function values:



iii. BFGS

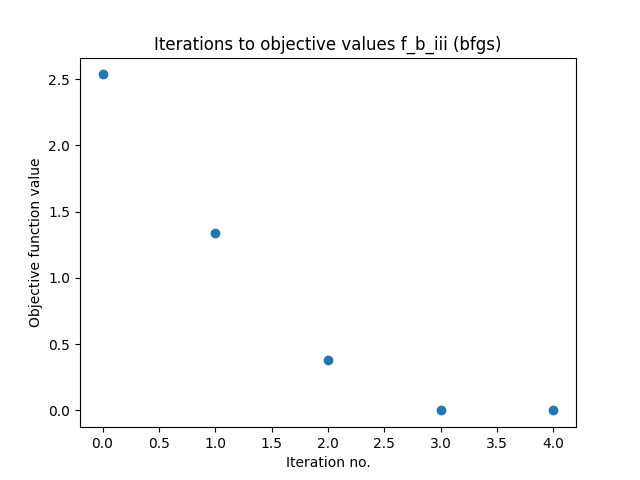


**Last print to console:**

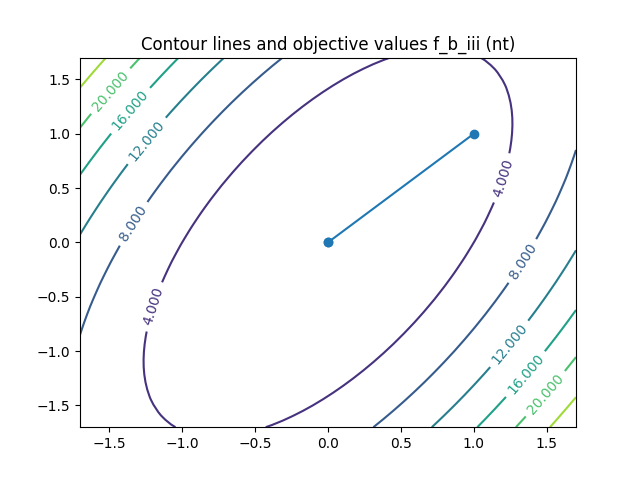
Iteration number: 4 current location: [-0.00036122 -0.00038291] current obj val: 3.360277246773555e-07 current step length: 0.011671237555906225 current change in objective function value: 0.00032720331900894836 (bfgs)

Quad function f\_b\_iii (dir method: bfgs) result: Success

Iterations to objective function values:



iii. Newton



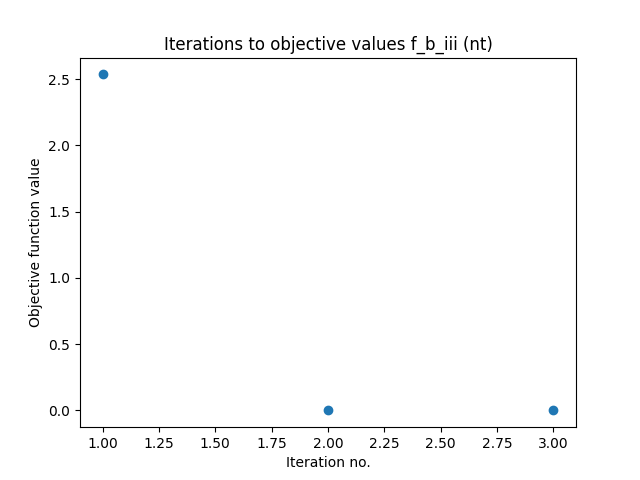
**Last print to console:**

Iteration number: 1 current location: [0. 0.] current obj val: 0.0 current step length: 1.4142135623730951 current change in objective function value: 2.535898384862245 (nt)

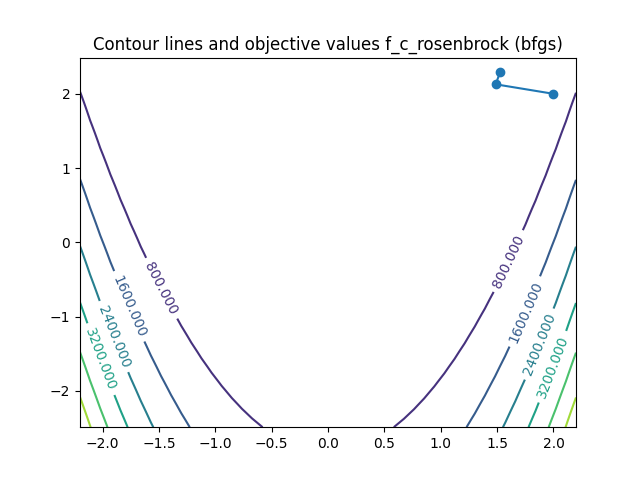
Iteration number: 2 current location: [0. 0.] current obj val: 0.0 current step length: 0.0 current change in objective function value: 0.0 (nt)

Quad function f\_b\_iii (dir method: nt) result: Success

Iterations to objective function values:



c. Rosenbrock function - BFGS:

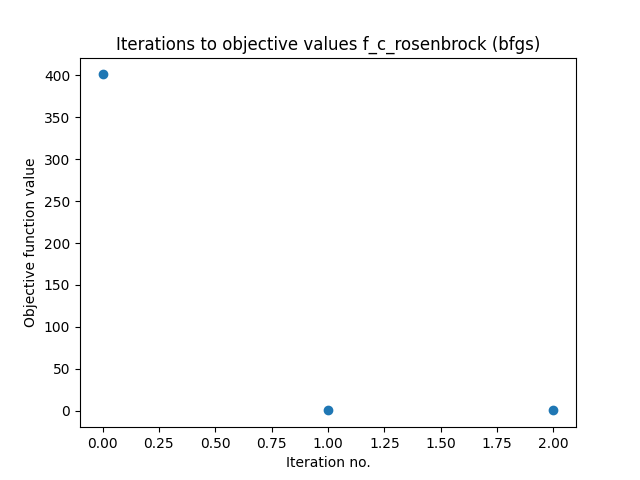


**Last print to console:**

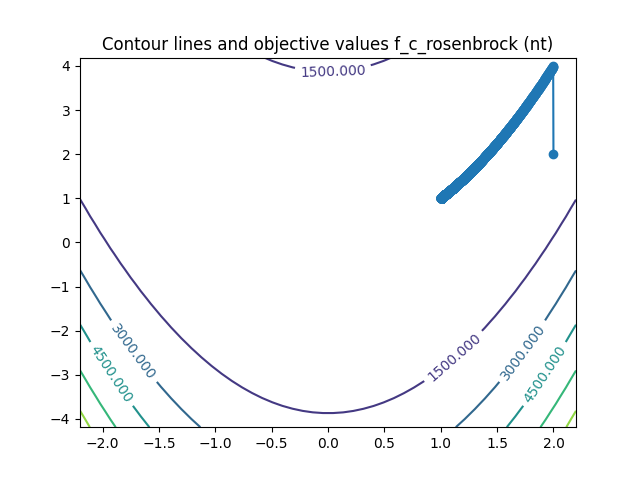
Iteration number: 2 current location: [1.52602603 2.28747033] current obj val: 0.44714955865017353 current step length: 0.16409097583095522 current change in objective function value: 0.5000040891759254 (bfgs)

Rosenbrock function (dir method: bfgs) result: Success

Iterations to objective function values:



Rosenbrock function - Newton:

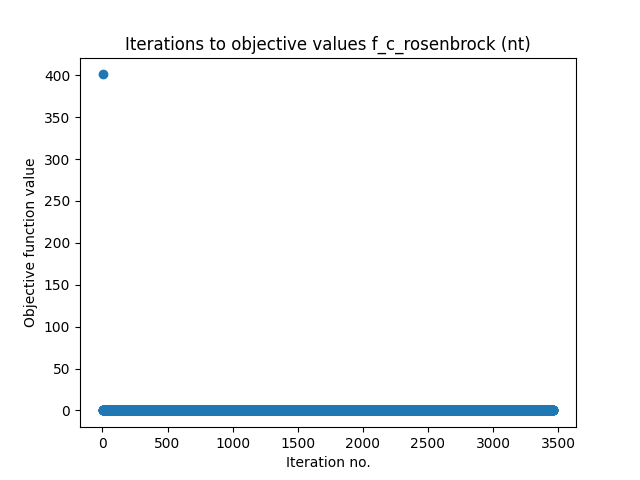


**Last print to console:**

Iteration number: 3463 current location: [1.00629979 1.01262354] current obj val: 3.971217146234164e-05 current step length: 1.7780427101892694e-05 current change in objective function value: 9.990883531765224e-08 (nt)

Rosenbrock function (dir method: nt) result: Success

Iterations to objective function values:



Functions from exercise 2:

\*For newton decrement in bfgs[[3]](#footnote-3) and termination condition[[4]](#footnote-4), I used .

- Also used the following consts:

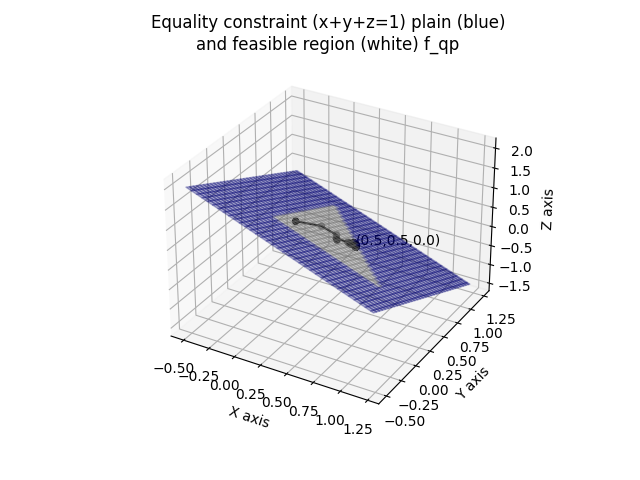
- Max iterations number (external & internal): 2000

- For convergence check: step tolerance = , objective value change tolerance =

- They all appear in “My consts” constrained\_min.py

QP:

The plain is the plain induced by the equality constraint and the polygon is induced by the inequality constraints. The points are plotted and connected. The final point coordinates are shown next to it (0.5,0.5,0).



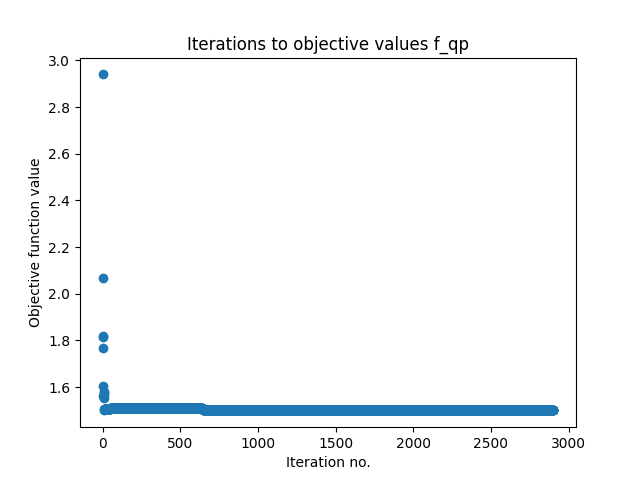
**Last print to console:**

Iteration number: 2902 current location: [4.99852617e-01 5.00012065e-01 1.35318454e-04] current obj val: 1.5001353586324684 current step length: 0.00012885441294046939 current change in objective function value: 6.443446243231676e-08

Function f\_qp result: Success.

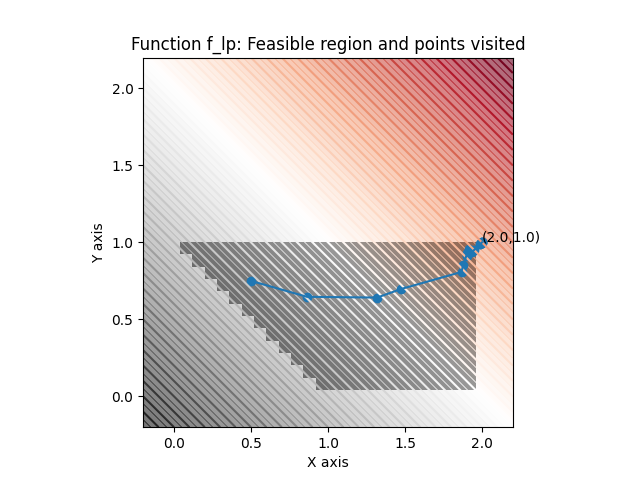
Final candidate: [4.99852617e-01 5.00012065e-01 1.35318454e-04] objective function value: 1.5001353586324684 inequality constraints values: qp\_ineq1: -0.49985261700003797 qp\_ineq2: -0.5000120645458818 qp\_ineq3: -0.00013531845408127767

Iterations to objective function values:



LP:

The feasible region induced by the inequality constraints is shown, as well as the points.  
The final point coordinates are shown next to it (2,1).



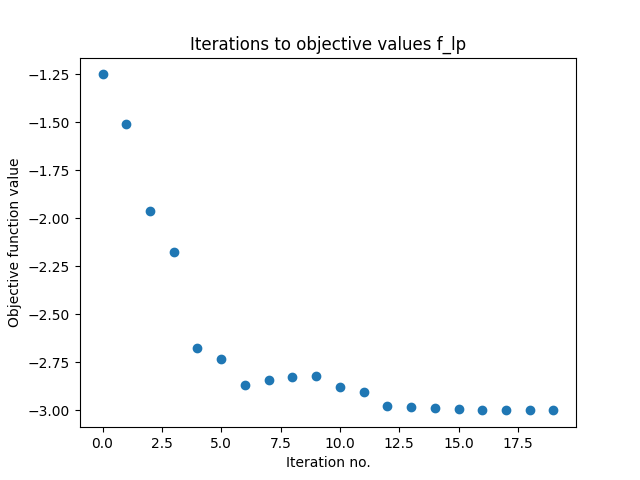
**Last print to console:**

Iteration number: 19 current location: [1.99990051 0.9999033 ] current obj val: -2.9998038114581878 current step length: 0.0005716479350954364 current change in objective function value: 0.0008084086192412521

Function f\_lp result: Success.

Final candidate: [1.99990051 0.9999033 ] objective function value: -2.9998038114581878 inequality constraints values: lp\_ineq1: -1.9998038114581878 lp\_ineq2: -9.669746198892604e-05 lp\_ineq3: -9.949107982354377e-05 lp\_ineq4: -0.9999033025380111

Iterations to objective function values:



1. <https://github.com/Amannor/python_numerical_optimizations/tree/main/ex2/programming_part> [↑](#footnote-ref-1)
2. See lecture 7+8, slide no. 36 [↑](#footnote-ref-2)
3. See lecture 7+8, slide no. 36 [↑](#footnote-ref-3)
4. See lecture 7+8, slide no. 64 [↑](#footnote-ref-4)