

Lista4

December 8, 2018

1 Lista 4

2018/3

Aluno: Pedro Bandeira de Mello Martins

Disciplina: CPE773 - Otimização Convexa

Professor: Wallace A. Martins PEE/COPPE - UFRJ

```
In [1]: import sys
        if '../..' not in sys.path:
            sys.path.append('../..')
        import warnings
        warnings.filterwarnings('ignore')
        from functions import exercise1116,\
            question42,\
            question43,\
            question44,\
            question45,\
            functionObj
        from models.optimizers import SteepestDescentAlgorithm,\
            FletcherReevesAlgorithm,\
            BacktrackingLineSearch,\
            FibonacciSearch,\
            ConjugateGradientAlgorithm,\
            QuasiNewtonAlgorithm, \
            BasicNewtonAlgorithm
        from models.utils import UnconstrainProblem
        from autograd import numpy as np
        from run_exercise import run_exercise
```

1.1 Questão 1

Para o problema:

minimize

$$f(x) = x_1 + 1.5x_2 + x_3 + x_4$$

subject to:

$$x_1 + 2x_2 + x_3 + 2x_4 = 3$$

$$x_1 + x_2 + 2x_3 + 4x_4 = 5$$

$$x_i \geq 0$$

```

In [2]: f_x, eqc, iqc = exercise1116()
        initial_x = np.array([0, 0], dtype=np.float64)

In [3]: solution = run_exercise(f_x, eqc, iqc,
                                [(SteepestDescentAlgorithm, BacktrackingLineSearch),
                                 ConjugateGradientAlgorithm,
                                 QuasiNewtonAlgorithm,
                                 (BasicNewtonAlgorithm, BacktrackingLineSearch)],
                                initial_x,
                                plot_charts=False)

QuasiNewtonAlgorithm didn't converge. AssertionError('X out of domain',)
minimize didn't converge. AttributeError('nhev',)

In [4]: solution[['best_f', 'best_x', 'fevals', 'grad_evals', 'nevals', 'run_time (s)']]

Out[4]:

```

	best_f \
SteepestDescentAlgorithm + BacktrackingLineSearch	1.6666708720504597
ConjugateGradientAlgorithm	1.970588699457983
QuasiNewtonAlgorithm	1.9743015462053486
BasicNewtonAlgorithm + BacktrackingLineSearch	1.6666708175852056
minimize	1.6667928434652193

	[1e-07, 0.3333333, 2e-07, 1.1e-07]
SteepestDescentAlgorithm + BacktrackingLineSearch	[0.2352941, 0.254902, 0.4509804, 0.2352941]
ConjugateGradientAlgorithm	[0.2352941, 0.254902, 0.4509804, 0.2352941]
QuasiNewtonAlgorithm	[0.2352941, 0.254902, 0.4509804, 0.2352941]
BasicNewtonAlgorithm + BacktrackingLineSearch	[5e-07, 0.3333332, 4e-07, 1.1e-07]
minimize	[0.000252, 0.3332493, 7.26e-05, 1.1e-07]

	fevals	grad_evals	nevals \
SteepestDescentAlgorithm + BacktrackingLineSearch	12511	1372	13883
ConjugateGradientAlgorithm	4000	4004	8004
QuasiNewtonAlgorithm	5	10	15
BasicNewtonAlgorithm + BacktrackingLineSearch	7075	2037	9112
minimize	390	0	390

	run_time (s)
SteepestDescentAlgorithm + BacktrackingLineSearch	8.423343
ConjugateGradientAlgorithm	22.384894
QuasiNewtonAlgorithm	0.020881
BasicNewtonAlgorithm + BacktrackingLineSearch	14.320457
minimize	0.164817

Nessa questão, o algoritmo QuasiNewton não convergiu, assim como o algoritmo Gradiente Conjugado. Na tabela, a linha indicada por "minimize" mostra o resultado encontrado pela implementação do scipy do SLQSP. Dentre as implementações feitas para o trabalho, o algoritmo de Newton apresentou os melhores resultados.