

Trabalho 1

Simplex

O objetivo deste trabalho é basicamente um só. Implementar o simplex.

- (i) A linguagem pode ser python. É permitido o uso de estruturas de dados de bibliotecas de álgebra linear (numpy). Também recomendo usar scipy para fração. Se python for um problema, consideramos C. Mas aí você precisará tomar cuidado com o condicionamento da matriz. Sugiro implementar um 0 para números pequenos ou implementar fração.
- (ii) Pivoteamento deverá ser implementado como uma função separada. Ou seja, deverá haver uma função que recebe como entrada uma matriz e um par de índices (linha e coluna), e devolve uma matriz com aquela entrada pivoteada.
- (iii) Para escolher o elemento a ser pivoteado, você deve implementar duas funções. Ambas recebem como entrada uma matriz e cospem um par de índices (linha e coluna). A primeira função faz essa escolha usando a estratégia do simplex primal. A segunda o simplex dual. Desde que faça sentido, claro.
- (iv) A entrada, em arquivo .txt, do algoritmo será uma matriz com $(m+1)$ linhas e $(n+1)$ colunas da forma

$$\begin{pmatrix} \mathbf{c}^T & 0 \\ A & \mathbf{b} \end{pmatrix}$$

escrita no formato m **enter** n **enter** $[[\text{linha } 0], [\text{linha } 1], \dots, [\text{linha } n]]$, correspondendo à PL

$$\begin{aligned} \max \quad & \mathbf{c}^T \mathbf{x} \\ \text{sujeita a} \quad & A\mathbf{x} \leq \mathbf{b} \\ & \mathbf{x} \geq 0. \end{aligned}$$

Por exemplo, a PL

$$\begin{aligned} \max \quad & (1 \ 2 \ 3) \mathbf{x} \\ \text{sujeita a} \quad & \begin{pmatrix} 1 & 1 & 1 \\ 0 & -1 & 3 \end{pmatrix} \mathbf{x} \leq \begin{pmatrix} 2 \\ 9 \end{pmatrix} \\ & \mathbf{x} \geq 0 \end{aligned}$$

será dada por

$$\begin{aligned} & 2 \\ & 3 \\ & [[1, 2, 3, 0], [1, 1, 1, 2], [0, -1, 3, 9]] \end{aligned}$$

- (v) Ao ler a entrada, seu programa deve, primeiro, colocá-la em FPI. Então, deverá testar para ver se é possível rodar o simplex dual diretamente. Se não for possível, ele deve testar para tentar rodar o simplex primal diretamente. Se não for possível, ele deve montar uma PL auxiliar, resolvê-la, e caso seja viável, voltar para rodar o simplex primal.
- (vi) A saída deverá ser dois arquivos .txt. Ao final de cada chamada da função de pivoteamento (seja na PL original ou na auxiliar), você deve imprimir a nova matriz no primeiro .txt. Ao término do algoritmo, uma das três opções abaixo deverá ser impressa num arquivo .txt de nome “conclusao.txt”.

- (a) Se a PL for inviável, seu programa deve dizer

0
[certificado]

(o vetor certificado deverá estar impresso entre chaves com vírgulas separando as entradas).

- (b) Se a PL for viável e ilimitada, seu programa deve dizer

1
[certificado]

- (c) Se a PL for viável e limitada, seu programa deve dizer

2
[solução]
valor objetivo
[certificado]

- (vii) Os arquivos .py devem ser enviados para o email dcc035ufmg@gmail.com com a tag [TP1] no subject até o dia 07 de maio, 23h59. Não haverá aula nos dias 01 e 03 de maio. Seu arquivo deverá estar bem comentado. Esteja informado que iremos comparar todos os arquivos (incluindo de semestres anteriores) para plágio. Este TP deve ser escrito individualmente, mas você pode discutir com seus colegas ou pesquisar na internet para entender a idéia geral da implementação do simplex.
- (viii) Este TP será usado como base para o TP seguinte. Implemente de forma modularizada, como descrito acima.