

MCTA017-17 -- Programação Matemática -- Q2023.3

Sep 18, 2023

Atualizado em 18/09

Expediente

- Professor: [Aritanan Gruber](#)
- Moodle: [PM23.3](#)
(detalhes de andamento do curso, links úteis, avaliações, notas, etc.)
- Atendimento: Ter 12–13h e Qui 18–19h na S-539.2
- Monitoria: (a definir)
- Turmas e salas:
Diurno: Ter 10–12h e Qui 08–10h na S-214.0
Noturno: Ter 21–23h e Qui 19–21h na S-214.0

Ementa

Introdução: revisões de álgebra linear e conjuntos convexos. Programação linear: modelagem; resolução gráfica; teoremas básicos; o método simplex; simplex revisado; dualidade; algoritmos primal-dual e dual-simplex; análise de sensibilidade. Programação Dinâmica.

Sendo um pouco menos sucinto...

Objetivos: Entender e usar algumas das técnicas para construção de modelos de otimização linear e realizar argumentações precisas sobre a correção deles. Adquirir intuição geométrico-algébrica aprendendo a demonstrar resultados elementares de poliedros, lemas de alternativas, dualidade e condições de otimalidade. Ser capaz de argumentar sobre correção e complexidade computacional dos métodos algorítmicos vistos, identificando semelhanças e diferenças cruciais entre os mesmos.

Conteúdo programático: Problemas de otimização linear: viabilidade e otimalidade, formulações e equivalências. Sistemas de inequações lineares e lemas de alternativas. Dualidade linear e condições de otimalidade. Geometria, estrutura e representação de poliedros. Matrizes totalmente unimodulares e poliedros inteiros. Método Simplex: fases, ciclagem e regras de pivotação, desempenho, variante dual. Elementos do método dos elipsoides: complexidade e significância teórica. Métodos de pontos interiores: redução de potencial e caminho central. Otimização paramétrica e análise de sensibilidade.

Avaliações e critérios

- listas de exercícios opcionais (não contam para nota, mas é recomendado fazê-las!)
- projeto opcional $H \in [0, 2]$ (a não entrega do projeto fará com que seja tomado $H = 0$ (desta vez))
- duas provas regulares P_1 e P_2 e uma prova substitutiva P_3 aberta;
média aritmética das 2 melhores dentre as 3 possibilidades
 - P_1 : Qui 26/10 @ 08h (D) e 19h (N)
 - P_2 : Qui 07/12 @ 08h (D) e 19h (N)
 - P_3 : Qui 14/12 @ 08h (D) e 19h (N)

Nota nominal:

$$N = \frac{1}{2} \max \left\{ \sum_{j \in S} P_j : S \in \binom{[3]}{2} \right\} + H = \frac{1}{2} \max \{P_1 + P_2, P_1 + P_3, P_2 + P_3\} + H,$$

com $P_i \in [0, 10]$.

Conceito nominal (C_N): reflete o seu desempenho frente ao material apresentado e às avaliações realizadas; obtido pelo encaixe de N em um dos intervalos:

$$-\infty < \mathbf{F} < 5.0 \leq \mathbf{D} \leq 6.0 < \mathbf{C} \leq 7.0 < \mathbf{B} \leq 8.5 < \mathbf{A} < \infty.$$

Recuperação

Caso seu conceito C_N seja \mathbf{D} ou \mathbf{F} , você tem direito a uma prova de recuperação P_R . Esta será única e contemplará toda a matéria do quadrimestre. Uma nova nota nominal $\bar{N} = (N + P_R)/2$ será utilizada para gerar um novo conceito (nominal) *final pós-recuperação* \bar{C}_N . Note que seu conceito final pós-recuperação pode ser menor que o pré-recuperação: uma vez feita, a recuperação é parte integrante da sua avaliação.

Bibliografia

Usaremos diretamente Bertsimas-Tsitsiklis [A], Gärtner-Matousek [C], e Vanderbei [1]. Os três são muito didáticos e se “complementam.”

Para material mais avançado em teoria e algoritmos, consulte Nemhauser-Wolsey [D] e, principalmente, Schrijver [E]. Para modelagem, recomendamos Chen-Batson-Dand [B], Fourer-Gay-Kernighan [I] e Williams [J]. Conceitos de álgebra linear são cobertos em nível introdutório em Axler [K] e Strang [N] e, em forma mais avançada, em Hoffman-Kunze [O]. A referência clássica para análise convexa é Rockafellar [M]; Bertsekas-Nedic-Ozdaglar [L] é uma boa

alternativa – ambos contêm muito mais material do que vamos precisar. Para otimização além de linear, consulte Bertsekas [F], Boyd-Vendenberghé [G] e Ruszczyński [H].

Otimização Linear e Linear Inteira

- [A] D. Bertsimas e J. Tsitsiklis, [Introduction to Linear Optimization](#), Athena Scientific (1997)
- [B] D. Chen, R. Batson e Y. Dand, [Applied Integer Programming: Modeling and Solution](#), Wiley (2010)
- [C] B. Gärtner e J. Matousek, [Understanding and Using Linear Programming](#), Springer (2007)
- [D] G. Nemhauser e L. Wolsey, [Integer and Combinatorial Optimization](#), Wiley (1988)
- [E] A. Schrijver, [Theory of Linear and Integer Programming](#), Wiley (1986)
- [1] R. Vanderbei, [Linear Programming - Foundations and Extensions](#), 5th ed., Springer (2020)

Otimização Convexa e Não-Linear

- [F] D. Bertsekas, [Nonlinear Programming](#), 3rd ed., Athena Scientific (2016)
- [G] S. Boyd e L. Vendenberghé, [Convex Optimization](#), Cambridge University Press (2004)
- [H] A. Ruszczyński, [Nonlinear Optimization](#), Princeton University Press (2006)
- [P] N.K. Vishnoi, [Algorithms for Convex Optimization](#) (2020)

Modelagem e Ferramentas

- [I] R. Fourer, D. Gay e B. Kerningham, [AMPL: A Modeling Language for Mathematical Programming](#), 2nd ed., Thompson (2003)
- [J] H. Williams, [Model Building in Mathematical Programming](#), 5th ed., Wiley (2013)

Álgebra Linear e Análise Convexa

- [K] S. Axler, [Linear Algebra Done Right](#), 3rd ed., Springer (2015)
- [L] D. Bertsekas, A. Nedic e E. Ozdaglar, [Convex Analysis and Optimization](#), Athena Scientific (2003)
- [M] T. Rockafellar, [Convex Analysis](#), Princeton University Press (1970)
- [N] G. Strang, [Introduction to Linear Algebra](#), 5th ed., Wellesley Publishers (2016)
- [O] K. Hoffman e R. Kunze, [Linear Algebra](#), 2nd ed., Prentice Hall (1971)

Lista de tópicos por semana (tentativa)

- S01 – Introdução à Otimização e Dualidade Lineares
(BT[A] 1.1 – 1.4, 1.6, 1.8, 4.1, 4.2; GM[C] 1.1 – 1.4, 2.1 – 2.5, 3.1, 4.1, 6.1, 6.2)
- S02 – Elementos de Álgebra Linear e Convexidade
(BT[A] 1.5, 2.1; GM[C] Apêndice, 4.3; A[N] 1.B, 1.C, 2.A – 2.C, 3.A – 3.C, 6.A – 6.C)
- S03 – Geometria e Estrutura de Poliedros I
(BT[A] 2.2 – 2.8, 4.2 – 4.4, 4.6, 4.7; GM[C] 4.1 – 4.4, 6.1, 6.2, 6.4, 6.7)
- S04 – Geometria e Estrutura de Poliedros II
(BT[A] 2.2 – 2.8, 4.2 – 4.4, 4.6, 4.7; GM[C] 4.1 – 4.4, 6.1, 6.2, 6.4, 6.7)
- S05 – Geometria e Estrutura de Poliedros III
(BT[A] 2.2 – 2.8, 4.2 – 4.4, 4.6, 4.7; GM[C] 4.1 – 4.4, 6.1, 6.2, 6.4, 6.7)
- S06 – Método(s) Simplex I
(BT[A] 3.1 – 3.5, 3.7, 4.5; GM[C] 5.1 – 5.10)
- S07 – Método(s) Simplex II
(BT[A] 3.1 – 3.5, 3.7, 4.5; GM[C] 5.1 – 5.10)
- S08 – Análises Paramétrica e de Sensibilidade
(BT[A] 5.1 – 5.4, 6.1, 6.2, 12.1)
- S09 – Método(s) de Pontos Interiores
(BT[A] 9.1 – 9.6)
- S10 – Método de Elipsóides
(BT[A] 8.1 – 8.5)
- S11 – Matrizes Totalmente Unimodulares e Poliedros Inteiros
(BT[A] 10.1 – 10.3, 11.1, 11.2; GM[C] 3.1 – 3.4)
- S12 – Exame final e substitutiva

Estudando para esta disciplina

A natureza do tópico, o posicionamento do curso na grade, e a lista de pré-requisitos indicam que esta é uma disciplina de nível intermediário; e será tratada como tal. Você deve assistir às aulas, estudar a bibliografia indicada, e dedicar-se às listas de exercícios e ao projeto (caso atribuído).

Caso seus pré-requisitos não estejam tão sólidos quanto desejável (falta de familiaridade com formalismo matemático e raciocínio algorítmico, atitude passiva com relação ao aprendizado, tempo dedicado insuficiente, etc.), será possível fazer confusões e sentir-se perdido no início.

Alguns procedimentos que costumam funcionar em cursos introdutórios para mitigar os motivos relacionados também costumam funcionar por aqui:

- Refaça os exemplos e re-prove os resultados fornecidos em sala de aula.
- Preste atenção aos processos de solução (aprenda-os!) e não foque somente nos resultados finais.
- Assista ativamente às aulas; resolva os exercícios nelas propostos e os contidos nas listas.
- Estude a bibliografia indicada, monte grupos de estudo, e faça um bom uso dos horários de atendimento.
- Tenha sempre em mente que aprendizado é uma tarefa *ativa*; não fique somente assistindo. “Ouvir” às aulas e “ler” os livros tem pouco ou nenhum efeito neste curso – e em qualquer disciplina matemática/algorítmica que o valha.

- Se ainda assim, sentir-se perdido, repita os passos acima. Mais cedo ou mais tarde, eles convergirão à compreensão.

Note que você não será convidado a regurgitar respostas fornecidas em aula ou presente nos livros. As questões em listas e provas testarão sua capacidade de entender os problemas e apresentar uma solução para eles; às vezes, serão uma adaptação simples ou uma extensão direta do que foi visto, outras, será necessário relacionar dois ou mais métodos ou conceitos apresentados, e outras ainda, irão requerer análise e raciocínio mais profundo (o que leva tempo, então não deixe nada para a última hora!).

Integridade acadêmica e transgressões

O Artigo 25 do [Código de Ética da UFABC](#) estabelece, à página 23: "Quanto aos trabalhos acadêmicos, é eticamente inaceitável que os discentes:

- I - fraudem avaliações;
- II - fabriquem ou falsifiquem dados;
- III - plajeiem ou não creditem devidamente autoria;
- IV - aceitem autoria de material acadêmico sem participação na produção;
- V - vendam ou cedam autoria de material acadêmico próprio a pessoas que não participaram da produção."

Trabalhos (listas, provas, programas) suspeitos de cópia ou de outra representação fraudulenta acarretarão aos envolvidos conceitos **F** (falha) no curso. A atividade será reportada à [Comissão Disciplinar Discente](#) da universidade para que sejam tomadas todas as providências disciplinares cabíveis.

ChatGPT e similares

Antes de mais nada, representam um grande avanço no processamento automático de corpos linguísticos e são, sem sombra de dúvidas, resultados estupendos! [Mais considerações sobre isso ao longo do curso.]

Dito isso, soluções entregues que tenham sido produzidas por eles enquadram-se no Artigo 25 acima.

Para pensar ao longo do curso: *Do que adianta as máquinas aprenderem e os alunos não?*



Aritanan Gruber

Assistant Professor

"See, if y'all haven't the same feeling for *this*, I really don't give a damn. If you ain't feeling it, then dammit *this* ain't for you!"
(desconheço a autoria; agradeço a indicação)

[✉](#) [G+](#) [R^g](#) [ID](#) [in](#) [T](#) [CV](#)

