

MEC-502 – Tópicos Especiais em Mecânica dos Sólidos

Prof. Dr. Wallace G. Ferreira / Prof. Dr. William Maluf Filho / Prof. Eng. Paulo S. Silva

Disciplina: MEC-502 Tópicos Especiais em Mecânica dos Sólidos

Programa de Pós-Graduação em Engenharia Mecânica – PosMEC – UFABC

AE1 – Introdução à biblioteca PyTorch – Machine Learning Aplicado a Mecânica Computacional:

- 1) Busque na internet em artigos, blogs (ex. Medium), canais do Youtube, etc, tutoriais para se familiarizar com a linguagem Python e aplicações em Cálculo Vetorial e Tensorial, Machine Learning (regressão, classificação, etc), em especial aplicações e exemplos de cálculos tensoriais e implementações de redes neurais com as bibliotecas PyTorch e Keras / TensorFlow.

- 2) Ler os capítulos 1, 2 e 3 do livro:

Kollmannsberger, S. et al. *Deep Learning in Computational Mechanics – An Introductory Course*. Springer, 1st edition. 2021.

➔ Reproduzir exemplos e exercícios em PyTorch dos capítulos 1, 2 e 3 da 2ª Edição (2025) em:

<https://github.com/cmpmech/deep-learning-in-computational-mechanics>

- 3) Implemente exemplos de Cálculo Vetorial e Tensorial usando PyTorch (e outras bibliotecas Python pertinentes), considerando:

- Derivadas parciais de funções escalares com até três variáveis, explorando polinômios e trigonométricas.
- Operações de cálculo vetorial e matricial: produto escalar, produto vetorial, produto matriz-vetor, produto matriz-matriz, transposições, etc
- Para campos escalares e vetoriais, onde aplicável, calcular: gradientes, divergentes, hessianas, jacobianos e rotacionais.
- Use biblioteca gráfica (matplotlib, etc) para plotar campos escalares e vetoriais (1d, 2d, e 3d).
- Buscar em livros de Mecânica dos Sólidos, Mecânica do Contínuo e/ou Teoria da Elasticidade exemplos considerando: campos de deslocamento (1d, 2d e 3d) e derivar tensor de deformação, tensor de material, tensor de tensões. Ver exemplos básicos nos capítulos 1, 2 e 3 em: <https://github.com/cmpmech/deep-learning-in-computational-mechanics>

- 4) Implementar exemplos de Regressão Linear e Não Linear (1d) usando PyTorch.

- 5) Implementar exemplos de Redes Neurais usando PyTorch, variando hiperparâmetros de treinamento.

- 6) Forma de apresentação do relatório – [Prazo de Entrega: 23-jun-2025](#)

6.1) Criar um repositório no GitHub para organizar os arquivos das atividades em pastas

6.2) Enviar por e-mail para wallace.ferreira@ufabc.edu.br um relatório sucinto em formato PDF com a descrição da resolução da atividade. **Indicar no PDF o link do repositório GitHub para os arquivos de programas desenvolvidos.**

➔ Padrão para nome do arquivo: [PrimeiroNome_UltimoSobrenome_AE1.pdf](#)

6.3) Utilizar arquivos Jupiter Notebook (.ipynb) para desenvolver as atividades.

6.4) Não apresentar apenas os códigos, **descrever no Notebook passo a passo** do que está executando e as discussões dos resultados dos códigos usando campos de texto em **formatação Markdown**.