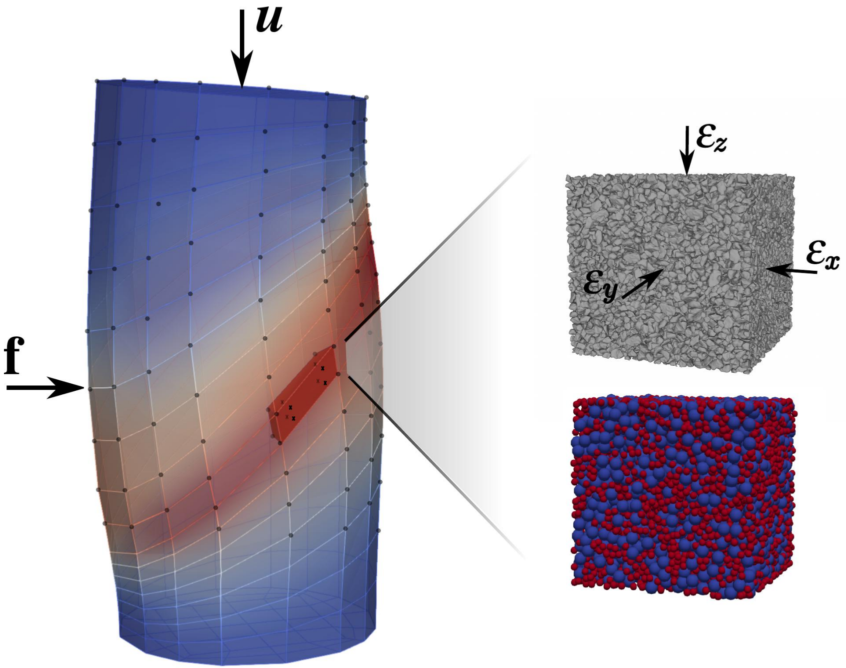


CIVIL-408 Multiscale Modeling in Mechanics

Prof. Kostas Karapiperis

Exercises - Week 10

Quiz

What is the main advantage of **a Physics-Informed Neural Operator** (PINO) over a **Physics-Informed Neural Network** (PINN) in solving mechanics problems?

a. A PINO can learn the general relationship between inputs and outputs, so it can handle new boundary conditions or loads without retraining.

b. A PINO strongly enforces physical laws in its loss function.

c. A PINO is easier to train compared to a PINN.

Quiz

What is the main advantage of **a Physics-Informed Neural Operator** (PINO) over a **Physics-Informed Neural Network** (PINN) in solving mechanics problems?

a. A PINO can learn the general relationship between inputs and outputs, so it can handle new boundary conditions or loads without retraining.

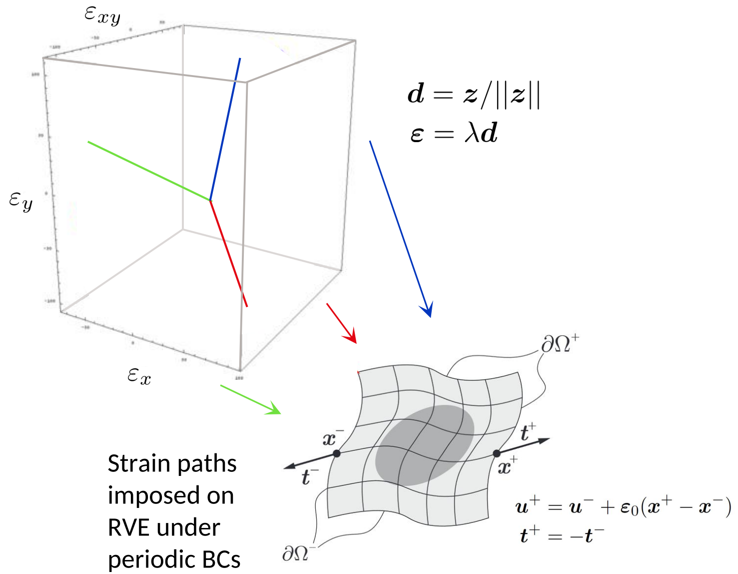
b. A PINO strongly enforces physical laws in its loss function.

c. A PINO is easier to train compared to a PINN.

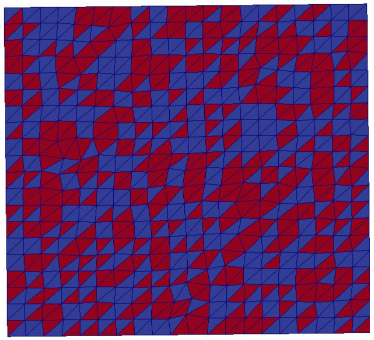
Machine learning-based multiscale modeling

Application to homogenization of elastic composite: Data

You are given a **material dataset** obtained by probing the effective constitutive response of a 2D isotropic composite material along different directions by repeatedly solving a boundary value problem an RVE:



Strain paths imposed on RVE under periodic BCs

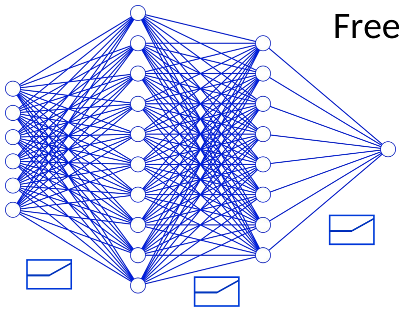


Isotropic composite

Machine learning-based multiscale modeling

Neural network-based surrogate constitutive law

We will use a thermodynamics-informed neural network. Recall from our lecture:



$$\boldsymbol{\varepsilon} \quad \hat{\psi}(\boldsymbol{\varepsilon}) \rightarrow \hat{\boldsymbol{\sigma}} = \frac{\partial \hat{\psi}}{\partial \boldsymbol{\varepsilon}}$$

We will use a simple Multilayer perceptron (Feedforward NN) to approximate the energy.

Note: Whoever is interested may also implement Input Convex Neural Networks, which guarantee convexity of the free energy!

Training:

$$\min_{\theta} \gamma_1 ||\hat{\psi} - \psi|| + \gamma_2 ||\hat{\boldsymbol{\sigma}} - \boldsymbol{\sigma}|| + \gamma_3 ||\hat{\boldsymbol{C}} - \boldsymbol{C}||$$

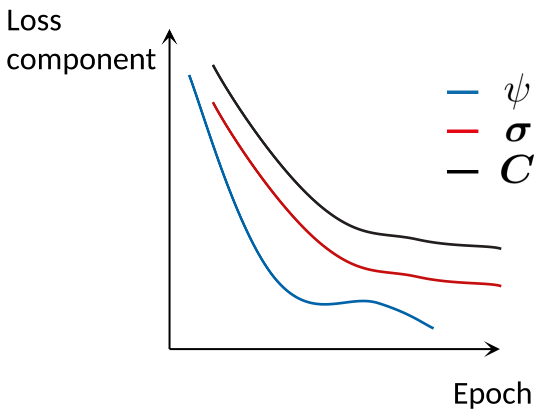
K.Karapiperis

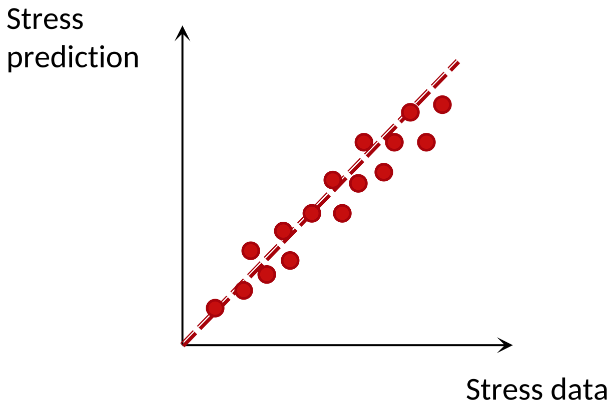
Exercise

Part 1: Training the neural network

Our first task is to train our neural network in order to obtain a surrogate model which we can later use within an FEM framework.

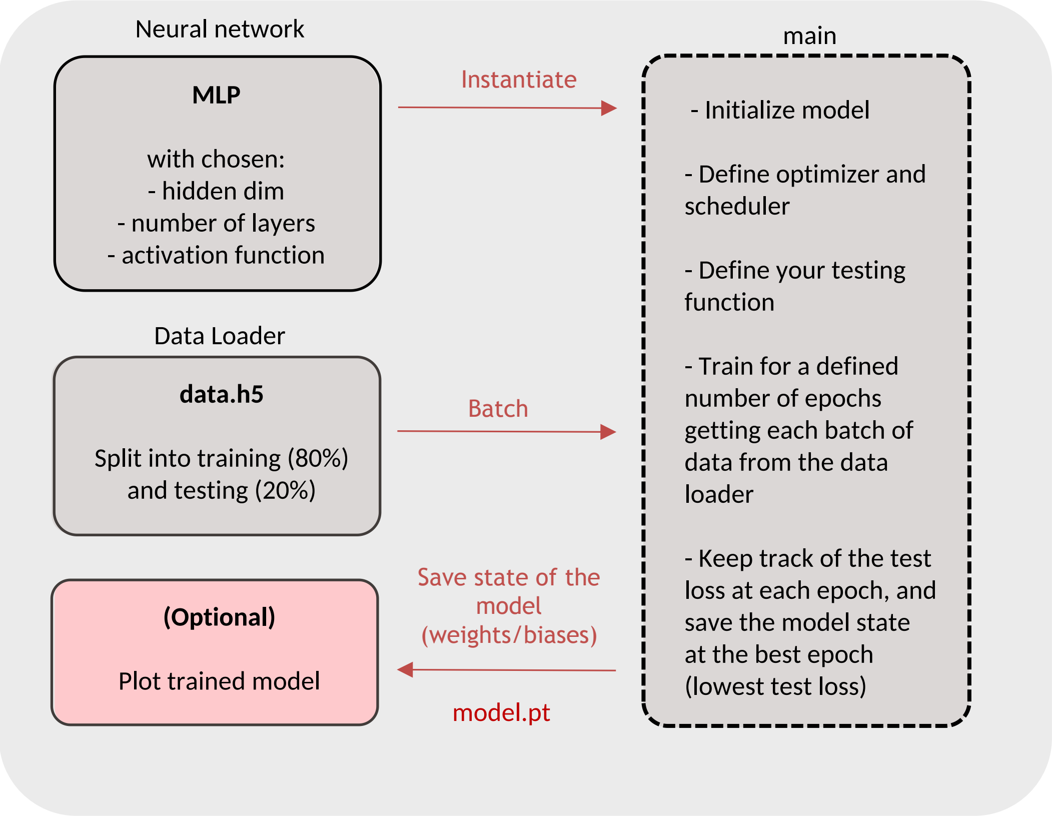
We monitor the components of the loss function during training and investigate the accuracy of a trained model at an unseen test dataset.





Python implementation

train_constitutive_model.py



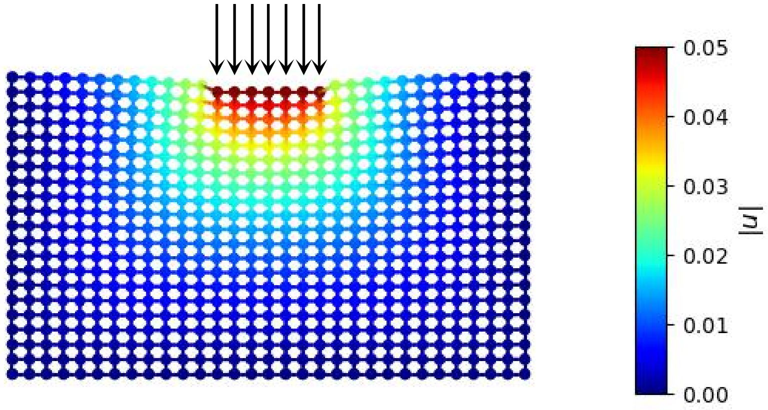
Legend

To do

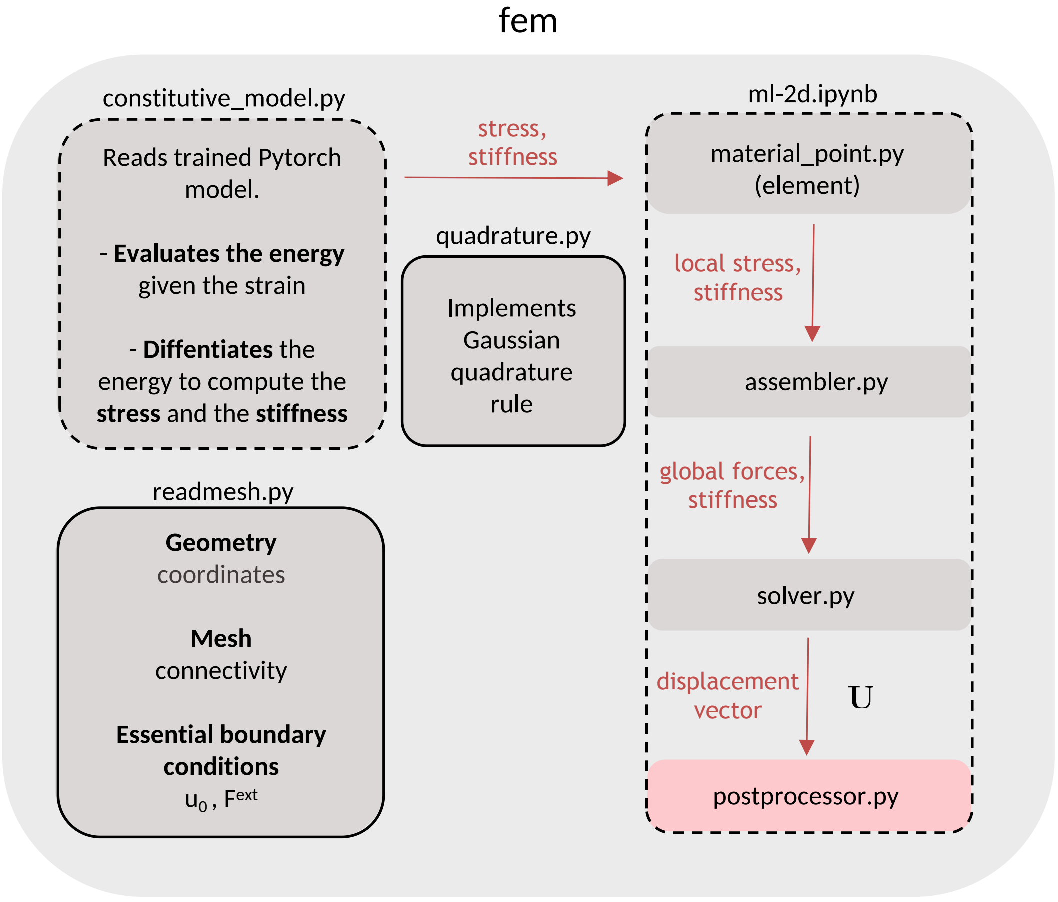
Exercise

Part 2: Utilizing the neural network as a surrogate constitutive law

Our second task is to setup the Finite Element model and interface it with our trained thermodynamically-informed neural network-based constitutive law.



Python implementation



Legend

To do

K.Karapiperis

Python notebook

Let’s move to the Python notebook