Project #1

assigned: Tuesday, Feb 11, 2025
due: Tuesday, Mar 4, 2025, 17:00
Please submit your project online through **Moodle**!

---

collaboration policy
You are encouraged to discuss the homework problems and solution strategies with your classmates, but you must find the solutions and write down your answers to all problems by yourself.

Problem 1: Learning constitutive model from data (45 points).

You are provided with 3 data set of stress-strain curves measured for 3 types of materials A, B and C under *random* loading paths. Let us use neural networks to find the constitutive model of the three materials. Your task consist of writing Pytorch codes to train neural networks and interpret the results:

(a) Define the input, and output of this problem, and write down the **mapping** to be approximated by the neural network.

(b) **Explain** your choice of:

1. training and test data
2. loss function
3. neural network architecture

(c) **Build, train and test** the neural networks using the PyTorch skeleton code provided. Plot the train and test error versus the training epochs, as well as one sample of truth stress and approxiamte stress from Nueral Networks for mateirals A, B and C each. Comment on how the neural network performances with different hyper parameters and explain your hyper parameter tuning strategy.

(d) What physics of this three materials can you learn from the data? Interpret your trained neural networks by testing it with different **loading paths**.

Code and data: You are provided with 3 material data in Matlab format (see "Materials_A-C.mat" in the data folder), and a skeleton code ("NN_skeleton.py").

For Materials A and B, the datasets consist of stress and strain measurements organized as 3D tensors. These tensors have dimensions defined by $[n\_sample, n\_dir, n\_step]$, where:

- n_sample denotes the number of samples,
- n_dir indicates the directiona of the strain or stress measurements, and
- n_step represents the discretization level or the number of measurement points.

For stress, n_dir is order as $[\sigma_{11}, \sigma_{22}, \sigma_{33}, \sigma_{12}, \sigma_{23}, \sigma_{13}]$, and the strain tensor follows the same order. You will find some entry of Material A is empty, and this represents a plane strain condition in the context of solid mechanics (no strain in the perpendicular direction.).

Material C represents a case where the dataset only includes 1D uni-axial deformation data, implying $n\_dir = 1$. (i.e., only σ_{11} and ϵ_{11} are provided)

Problem 2: Learning based design certification (55 points).

Let us study a "real-life" structure. Consider the certification of Eiffel tower, where we aim to assess whether the structure can withstand a certain distribution of pressure loading, as depicted in. 1(a). The structure consists of struts with a failure strength of 500 MPa. We are provided with a previously obtained data-set (see "Eiffel_data.mat" in data folder of problem 2) which records whether the structure will fail (output = 0) or survive(output = 1) from the pressure loading. To simplify the problem, we consider this loading is distributed over the top half of the tower but with random profile.

(a) Define the input, and output of this problem, and write down the **mapping** to be approximated by the neural network.

(b) Use the matlab code "GenData_Eiffel.m" (you should be able to find it from "Problem2_student" folder) to plot two examples of the original and deformed structure with title indicating whether the sample will survive the loading.

(c) Explain your choice of

1. training and test data
2. loss function

(d) **Build, train and test** the neural networks using the PyTorch skeleton code provided. You should build your codes based on problem 1, and develop three different network architectures: (i) fully connected neural network, (ii) Res-net based architecture and (iii) U-net based architecture for this task. Explain your design of the three architectures. Plot train error versus epochs, as well as the model accuracy for all three architectures. (**Hint**: You are encouraged to innovate within your network designs, particularly by integrating the concept of skip connections as seen in ResNet and U-Net. This is an opportunity to experiment with various layers and nonlinear functions. Enjoy the process of exploration and creativity in your architectural design!)

(e) Which architecture works the best, with fine tuned parameters? And what might be the reason behind it?

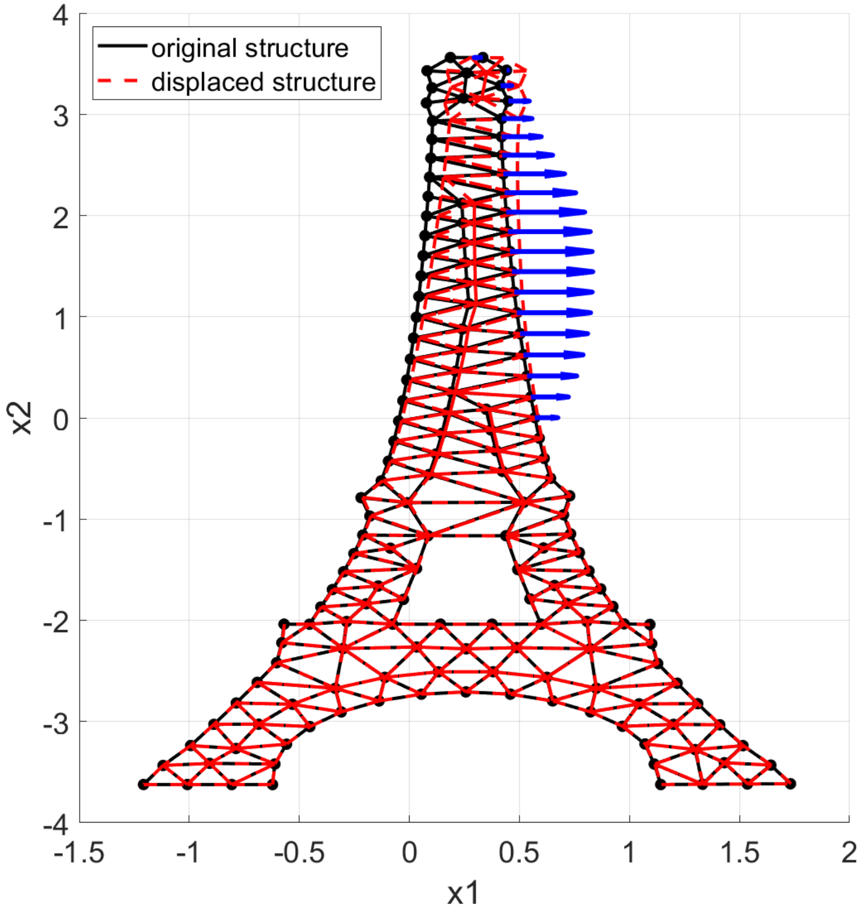

Figure 1: Effiel tower. (a) Original structure with distributed loading. (b) deformed structure and stress distribution

Your submission should include

- **Python scripts** answering problems 1 and 2: Problem1_Material_A.py, Problem1_Material_B.py, Problem1_Material_C.py, Problem2_FCNN.py, Problem2_Res_net.py, and Problem2_U_net.py
- a **PDF** file required to answer parts of problems 1 and 2.

Please note that commenting your code to make it clearly understandable will help the grading process.

Package all of the above in a **single** `.zip` **file** and submit it using the *Assignments* tool in Moodle.