Machine Learning per il Calcolo Scientifico

Third laboratory exercises

General guide

For each laboratory, an incomplete Python notebook will be provided with exercises (steps) that must be completed in the given order (some of the exercises will be needed in future laboratories). In step zero, all the Python packages that are needed in order to complete the notebook are listed. This PDF includes the text for the exercises and the expected outcomes for each step. While following the notebook is recommended, you are also welcome to attempt the exercises without using it.

Step Zero

Here are the required Python (`https://www.python.org/`) packages for this laboratory:

- PyTorch (`https://pytorch.org/`)
- Numpy (`https://numpy.org/`)
- Matplotlib (`https://matplotlib.org/`)
- ipymlp (`https://matplotlib.org/ipympl/`)

Exercise one: Solve a simple ODE

$$\begin{cases} \dfrac{dy(t)}{dt} = -2y(t), \quad t \in (0,1) \\ \\ y(0) = 1 \end{cases}$$

a. Using DeepNet (DNN.py) define a FNN with 3 layer and 5 neuron, with tanh as activation function
b. Using torch.rand define the training points (50), while torch.linspace for test (100)
c. Complete the functions eval_loss_ode and eval_loss_IC.
d. Following the documentation define the optimizer.
e. Complete the training loop (2000 epochs).
f. Visualize the results obtained by completing the plots written in the notebook

Exercise two: Non-linear ODE

$$\begin{cases} -\dfrac{d}{dx}\left[(1+u^2)\dfrac{du}{dx}\right] = -2\pi\cos(\pi x)^2\sin(\pi x) + \pi^2\sin(\pi x)\big(\sin(\pi x)^2+1\big), \quad x \in (0,2). \\ \\ u(0) = u(2) = 0. \end{cases}$$

a. Using DeepNet (DNN.py) define a FNN with 5 layer and 20 neuron, with tanh as activation function
b. Using torch.rand define the training points (200), while torch.linspace for test (500)
c. Complete the functions eval_loss_ode and eval_loss_IC.
d. Following the documentation define the optimizer.
e. Complete the training loop (5000 epochs).
f. Visualize the results obtained by completing the plots written in the notebook

Exercise three: Wave equation

$$\begin{cases} \dfrac{\partial^2 u}{\partial t^2} - 2\dfrac{\partial^2 u}{\partial x^2} = 0, & x \in [0,4], \quad t \in [0,3], \\ u(x,0) = 0, & x \in [0,4], \\ \dfrac{\partial u}{\partial t}(x,0) = \dfrac{\pi\sqrt{2}}{2} \sin\left(\dfrac{\pi x}{2}\right), & x \in [0,4], \\ u(0,t) = u(4,t) = 0, & t \in [0,3]. \end{cases}$$

a. Using DeepNet (DNN.py) define a FNN with 3 layer and 50 neuron, with tanh as activation function
b. Using torch.rand.uniform define the training points (In order 1500,500,100,100).
c. Complete the functions eval_loss_PDE and eval_loss_IC.
d. Following the documentation define the optimizer.
e. Complete the training loop (2000 epochs).
f. Visualize the results obtained by completing the plots written in the notebook