DLSC

Matthias Sanicanin

15.06.2023

Task 1: PINNs for solving PDEs

In the firs task we wanted to solve a system of equations with PINNs. The template from tutorial 4 contained most of the basic structure needed to complete this task, so I will only go over the points where my solution differs from the template.

The loss function has two key components, the function loss and the PDE loss.
The function loss is comprised of the loss of the fluid and solid approximations at the temporal boundary, the spatial boundary.
The **compute_temporal_boundary_residual** function, in my solution, simply enforces the boundary conditions

$$T_f(x, t = 0) = T_s(x, t = 0) = T_0$$

by returning the difference between the true solution and the approximate solution tensors of T_f, T_s, respectively, at the corresponding training points.
The **compute_spatial_boundary_residual** function, simultaneously penalises non-zero values of the approximate derivatives of T_f at $x = 1$ and of T_s at $x = 0$, $x = 1$ while also punishing deviations of the approximate solution of T_f at $x = 0$ from the true solution at the boundary. In short, we want to enforce

$$\frac{\partial T_f}{\partial x}|_{x=1} = \frac{\partial T_s}{\partial x}|_{x=0} = \frac{\partial T_s}{\partial x}|_{x=1} = 0 \quad \text{and} \quad T_f(x = 0, t) = \frac{T_{hot} - T_0}{1 + exp(-200(t - 0.25))} + T_0.$$

Finally, the **compute_temporal_interior_residual** function calculates large deviations of the approximate solutions of T_f and T_s from the two PDEs governing the fluid and solid temperatures,

$$\begin{aligned} \frac{\partial T_f}{\partial x} + U_f \frac{\partial T_f}{\partial x} &= \alpha_f \frac{\partial^2 T_f}{\partial x^2} - h_f(T_f - T_s) \\ \frac{\partial T_s}{\partial t} &= \alpha_s \frac{\partial^2 T_s}{\partial x^2} + h_s(T_f - T_s). \end{aligned}$$

Hyperparameter tuning turned out to be quite simple in this case, I did some tests to gauge the required network capacity, i.e. how many hidden-layers and neurons are necessary to get a decent result and then I played around with different weights of the function loss vs. PDE loss. Decided to make sure the initial and spatial boundary conditions were satisfied by weighting the function loss heavily.