Tutorial 5: Physics Informed Neural Networks (PINNs)

In this tutorial, you will learn to work with a PINN which is basically a form of neural differential equation. A common problem of neural networks (both FNNs and CNNs) is that the training data is rather limited. There is a risk of overfitting to the small amount of data that one has and not being able to test the model accurately. As we have seen, a common approach to avoid overfitting is regularisation but this may fail with limited data. PINNs are a way to regularise a neural network in a more advanced way. Essentially, they help the neural network function have the right shape. To do this, one imbeds the network with information in the form of a differential equation. When there is little data available, being able to imbed additional information that isn't data into the network is extremely powerful. In this tutorial, we will consider a PINN for a simple cooling problem (e.g. of water in a container), see notebook: Tutorial_5.ipynb.

(i) Formulate the differential equation that is used in the notebook (in cell 2) for the cooling law?

(ii) Next, two (classical) networks are trained with only N training data (10 in the notebook), one with and one without regularisation. Describe the differences between the results of these networks. Does this difference decrease with the amount of data points used for training?

(iii) What is the loss function that is used in the PINN?

(iv) Why is the performance of the PINN so much better than the two classical networks? How does this performance difference change with the number of data used for training?

(v) Suppose now that the cooling rate, say r is unknown in the equation. Describe the procedure on how r can be determined from the data (as implemented in the notebook).

(vi) Study the accuracy of the cooling rate, as determined in (v), versus the number of data points on which the network is trained.