PINN Working

For Example, let's take a Logistic Differential Equation.

Why?

Logistic Differential Equation is simple and intuitive: It is a well-known first-order ordinary differential equation that is straightforward to understand.

What is First Order Ordinary Differential Equation?

It is a mathematical equation that involves an unknown function and its first derivative with respect to a single independent variable. They can describe relationships such as rates of change, growth, decay, and many other dynamic processes.

$$\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{t}} = \boldsymbol{Rf}(\boldsymbol{t}).(\mathbf{1} - \boldsymbol{f}(\boldsymbol{t}))$$

Boundary Condition,

If, t(0) = 0.5

PINNs are based on two fundamental properties of NNs:

1- NNs are **universal function approximators.**
2- NN is **automatic differentiation.**
3-

How can we make the NN learn the solution?

loss contribution is taken as the residual of the differential equation as follows

$$\frac{df_{\mathrm{NN}}}{dt} - R\, f_{\mathrm{NN}}(t)\,(1 - f_{\mathrm{NN}}(t)) = 0$$

where f_NN(t) is the output of an NN with one input and its derivative is computed using AD. It is immediate to see that if the NN output respects the equation above, one is solving the differential equation. To compute the actual loss contribution coming from the DE residual, one needs to specify a set of points in the equation domain (usually referred to as colocation points) and evaluate the mean square error (MSE) or another loss function as an average over all the chosen colocation points.

What are Colocation points? How are they measured? How do we know what points to use, keep, and discard?

- Collocation points are spatial or temporal locations within the domain of the PDE where the residual of the PDE is computed.
- Collocation points are typically chosen based on various criteria
    - Uniform Sampling. Points are chosen uniformly across functions. For example, between 0 and 1, this can be written as a torch.linespace(0, 1, steps=30).
    - Adaptive sampling: Concentrate more points in regions where the solution varies rapidly or where boundary conditions are located. Use adaptive mesh refinement (AMR) or hierarchical refinement to concentrate sampling points in specific regions of interest while reducing the number of points elsewhere.
    - Random Sampling: Randomly selecting collocation points to ensure a diverse representation of the domain.
- Points that are too close together or too sparse may lead to poor convergence or inaccurate solutions.

$$\mathcal{L}_{\mathrm{DE}} = \frac{1}{M} \sum_{j=1}^{M} \left(\left. \frac{d f_{\mathrm{NN}}}{dt} \right|_{t_j} - R\, f_{\mathrm{NN}}(t_j)(1 - f_{\mathrm{NN}}(t_j)) \right)^2$$

Loss contribution given by the differential equation residual averaged over a set of collocation points.

However, a loss based only on the above residual does not ensure to have a unique solution to the equation. Therefore, let's include the boundary condition by adding it to the loss computation in the same way as above:

$$\mathcal{L}_{\mathrm{BC}} = (f_{\mathrm{NN}}(t_0) - 0.5)^2 \quad \text{with } t_0 = 0$$

Boundary conditions loss contribution added to the MSE loss.

Hence, the final loss simply reads:

$$\mathcal{L} = \mathcal{L}_{DE} + \mathcal{L}_{BC}$$

During the optimization procedure, this is minimized and the NN output is trained to respect both the differential equation and the given boundary condition, thus approximating the final DE solution.