

# PDE 神经算子与 PINN 神经网络的映射差异

## 1 PDE 神经算子 (Neural Operator)

PDE 神经算子直接学习从 PDE 参数到 PDE 解的映射，形式上是从函数空间到函数空间的映射。

- 给定一个偏微分方程的参数（例如源项或系数），记作  $\mathbf{a}(x)$ ，神经算子学习的映射是：

$$\mathcal{G} : \mathbf{a}(x) \mapsto u(x)$$

其中：

- $\mathcal{G}$  是神经算子，它将 PDE 的参数  $\mathbf{a}(x)$  映射到解  $u(x)$ 。
- $u(x)$  是 PDE 的解，它可以是标量场或向量场，例如  $u(x) = [u_x(x), u_y(x)]$  表示速度场的两个分量。
- 神经算子的训练过程基于输入输出对  $(\mathbf{a}(x), u(x))$ ，其中  $\mathbf{a}(x)$  是输入的 PDE 参数，而  $u(x)$  是相应的 PDE 解。训练目标是最小化下列损失函数：

$$\min_{\theta} \sum_i \|\mathcal{G}_{\theta}(\mathbf{a}_i(x)) - u_i(x)\|^2$$

其中：

- $\theta$  是神经算子的参数（权重和偏置）。
- $\mathbf{a}_i(x)$  是训练集中的参数输入， $u_i(x)$  是对应的 PDE 解。

## 2 PINN 神经网络 (Physics-Informed Neural Network)

PINN 通过同时优化 PDE 的物理残差和数据误差，学习 PDE 解的近似函数。PINN 的核心是构造一个深度神经网络  $u(x; \mathbf{W})$ ，其中  $\mathbf{W}$  是神经网络的可训练参数，网络输出  $u(x)$  是 PDE 的解。

- 记 PDE 的一般形式为：

$$\mathcal{L}(u, \mathbf{a}) = 0$$

其中  $\mathcal{L}$  是 PDE 算子， $u(x)$  是待求解的函数， $\mathbf{a}$  是 PDE 的参数。

- PINN 的损失函数包括两部分：

1. 残差损失 (PDE 的物理约束)：

$$L_{\text{res}}(\mathbf{W}, \mathbf{a}) = \|\mathcal{L}(u(x; \mathbf{W}), \mathbf{a})\|^2$$

这是 PDE 约束的残差损失，用于确保网络解满足 PDE。

2. 数据损失 (拟合观测数据)：

$$L_{\text{data}}(\mathbf{W}) = \sum_i \|u(x_i; \mathbf{W}) - u_i^{\text{data}}\|^2$$

其中  $u_i^{\text{data}}$  是观测到的数据，网络输出  $u(x_i; \mathbf{W})$  应与之匹配。

- 最终的 PINN 优化目标是最小化总损失：

$$\min_{\mathbf{W}, \mathbf{a}} (L_{\text{res}}(\mathbf{W}, \mathbf{a}) + \lambda L_{\text{data}}(\mathbf{W}))$$

其中  $\lambda$  是权衡数据损失和物理残差损失的超参数。

## 3 映射形式对比

- PDE 神经算子学习的是从参数到解的映射：

$$\mathcal{G} : \mathbf{a}(x) \mapsto u(x)$$

神经算子不依赖于具体的 PDE 形式，而是通过大量的 PDE 解数据来训练从参数到解的泛化映射。

- **PINN 神经网络**学习的是从输入空间  $x$  到解空间  $u(x)$  的映射，且在优化过程中包含了对 PDE 的物理约束：

$$u(x; \mathbf{W}) : x \mapsto u(x)$$

同时需要最小化 PDE 残差  $L_{\text{res}}$  和数据损失  $L_{\text{data}}$  的总和。