Seattle SciML Summer School: Case Study Two
An Introduction to Fourier Neural Operators (FNO)

Chris Budd OBE (University of Bath) and
Chaoyu Liu (University of Cambridge)

June 3, 2025

1 Introduction

This case study introduces the **Fourier Neural Operator** (FNO) [LKA+21], a neural operator-based method for learning mappings between function spaces. FNOs have shown excellent performance on learning solutions to PDEs, especially those with complex geometries or parametric coefficients. In this exercise, we will train an FNO model from the repository

`https://github.com/neuraloperator/neuraloperator`

to learn the solution operator of a two-dimensional linear elliptic PDE: **Darcy flow**. If time permits you can look at some other challenging examples of applications of the FNO methodology.

2 The 2D Darcy Flow Problem

We consider the linear elliptic PDE given by

$$-\nabla \cdot (a(x)\nabla u(x)) = f(x), \quad x \in [0,1]^2, \quad u|_{\partial\Omega} = 0. \tag{1}$$

which describes the flow of ground water, or oil, through fractured rock Here,

- $a(x)$ is a spatially varying permeability coefficient.
- $f(x)$ is a known source term (typically constant).
- $u(x)$ is the unknown pressure field to be learned.

We assume $a(x)$ is sampled from a Gaussian random field. The goal of this problem is to learn the mapping from $a(x)$ to $u(x)$, by training the FNO over a dataset of pre-computed solutions.

3 Training FNO on Darcy Flow

1. Open a **Colab notebook**.
2. Enable GPU support via `Runtime` → `Change runtime type` → `T4 GPU` → `Save`.
3. Clone the following GitHub repository:
   ! git clone `https://github.com/NeuralOperator/neuraloperator`
4. Navigate to the folder `fourier_neural_operator/`:

   ```
   %cd neuraloperator
   ```

5. Install the necessary dependencies using:

   ```
   !pip install -r requirements.txt
   ```

6. After installing the dependencies, install the package in editable mode using:

```
!pip install -e .
```

7. Open the train_darcy.py in the Python-Examples Git-Hub folder and copy its content and replace scripts/train_darcy.py in the FNO repository.

8. Train the FNO model using the training script:

```
!PYTHONPATH=$(pwd) python scripts/train_darcy.py
```

9. Visualise the predicted pressure field and compare it with the ground truth:

```
from IPython.display import Image
Image("/content/neuraloperator/input_prediction_truth.png")
```

4 Further Exploration

- Locate the configuration file for FNO and adjust the number of training epochs.
- Experiment with different numbers of Fourier modes and network layers to observe their impact on training performance and generalisation.

5 Advanced Exercise

If you're interested, try training the FNO on the 2-d Navier-Stokes equation for a viscous, incompressible fluid in vorticity form.

References

[LKA+21] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. *International Conference on Learning Representations*, 2021.