Fourier Neural Operator for Parametric PDE

Debjit Hore

Indian Institute of Technology Delhi

Prof. Sitikantha Roy

April 21, 2023

Outline

Fourier Neural Operator for Mapping between Functional Spaces

- The 1-d Burgers' equation is a non-linear PDE with various applications including modeling the **one dimensional flow** of a viscous fluid. It takes the form:

$$\frac{\partial u(x,t)}{\partial t} + u\frac{\partial u(x,t)}{\partial x} = \nu \frac{\partial^2 u(x,t)}{\partial x^2} \qquad x\epsilon(0,1), t\epsilon(0,1] \qquad (1)$$

$$u(x,0) = u_o(x) \qquad x\epsilon(0,1) \qquad (2)$$

- Where u_o is the initial condition and $\nu \ \epsilon \ \mathbb{R}_+$ is the viscosity coefficient. Our aim is to learn the **operator mapping the initial condition to the solution at time t=1**, defined by $u_o \rightarrow$ u(.,1)

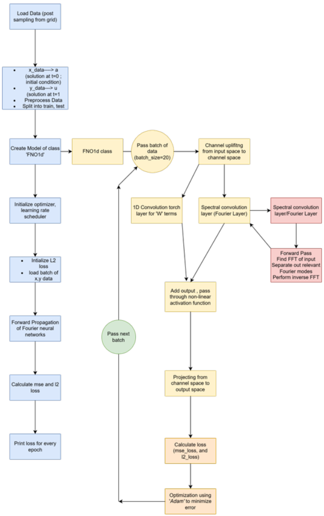
Data

- The data generation schema can be read in section 3.1 of the paper
- An initial condition has been generated $u_o(x)$ has been generated, viscosity $\nu = 0.1$ and the equation has been solved on a spatial mesh with resolution $2^{13} = 8192$ and use this dataset to subsample other resolutions

FNO Architecture

Channel Width (in_channel, out_channel)	64
Fourier Modes	16
Activation	Gelu
Optimizer	Adam
No. of Epochs	500
Loss	MSE, L2
Number of Fourier Layers	4
Number of Bias layers	4
Uplifting from Spatial to Fourier space	2→64
Downsizing from Fourier to Real Space	64 →128, 128 →1

Flowchart


Load Data (post sampling from grid)
x_data----> a (solution at t=0 ; initial condition)
y_data---> u (solution at t=1
Preprocess Data
Split into train, test
Create Model of class 'FNO1d'
FNO1d class
Pass batch of data (batch_size=20)
Channel uplifitng from input space to channel space
1D Convolution torch layer for 'W' terms
Spectral convolution layer (Fourier Layer)
Spectral convolution layer/Fourier Layer
Forward Pass
Find FFT of input
Separate out relevant Fourier modes
Perform inverse FFT
Add output , pass through non-linear activation function
Initialize optimizer, learning rate scheduler
Intialize L2 loss
load batch of x,y data
Forward Propagation of Fourier neural networks
Pass next batch
Projecting from channel space to output space
Calculate mse and l2 loss
Calculate loss (mse_loss, and l2_loss)
Print loss for every epoch
Optimization using 'Adam' to minimize error