Overview
Novel Deep Learning Approach for the Laplace equation
\[ \Delta u(x) = f(x), x \in \Omega \subset R^d \]
\[ u(x) = g_D(x), x \in \partial \Omega \]

Motivation
- Traditional methods in engineering are good at \( d = 1, 2, 3 \);
- For high dimensions, they suffer from the curse of the dimensionality;
- Deep Learning is a very promising approach.

Datasets
- Generate Boundary Data \((x_i, y_i) \in \partial \Omega, \ (g_D)_i = g_D(x_i, y_i)\)
- Generate Domain Data \(\{x_i, y_i \in \Omega, f_i = f(x_i, y_i)\} \)

Data Generation: A 2D Example.

Models
Boundary Network \(A(x, y; w)\)
Approximation on the boundary
PDE Network \(N(x, y; w)\)
Coupled with \(A(x, y; w)\)
Approximation within the domain
Loss Function
\[ \sum_{i} (f(x_i, y_i) - u(x_i, y_i))^2 + \sum_{i} (f(x_i, y_i) - \Delta u(x_i, y_i))^2 \]

Training Algorithm (GAN style):
for number of training iterations
for \( k \) steps do
sample minibatch on the boundary
train the boundary network
end for
sample minibatch within the domain
train the PDE network
end for

Results
- PDE: \( \Delta u(x, y) = f(x, y) \)
- Boundary condition: \( u(x, y) = g_D(x, y) = 0, x, y \in \partial [0, 1]^2 \)
- Exact Solution: \( u(x, y) = \sin(\pi x) \sin(\pi y), f(x, y) = -4\pi^2 u(x, y) \)

Insights
- For small dimensions, increasing #layers does not increase accuracy, but accelerate convergence.
- For large dimensions, more iterations in training are needed to see convergence, while increasing #layers may also accelerate convergence.

Future Work
- Generalize results to other types of PDEs.
- Investigate algorithms for ill-behaved solutions, such as peaks, exploding gradients, oscillations, etc.

References