Ingeniería Matemática
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
UNIVERSIDAD DE CHILE

The Math Behind the Magic

Neural Networks, Theory and Practice

Joaquín Fontbona, Javier Maass y Diego Olguín

Departamento de Ingeniería Matemática
Universidad de Chile

18 de diciembre

Contenidos

Physics-Informed Neural Networks

DIM Universidad de Chile

Más allá de la información de los datos

- Ya hemos visto que, si sabemos que los datos tienen estructura de imagen, entonces podemos adaptar la red y mejorar los resultados.
- Ahora, supongamos que queremos aproximar una cierta función y conocemos alguna ecuación diferencial que satisface, ¿podemos integrar ese conocimiento a la red?

Más allá de la información de los datos

- Ya hemos visto que, si sabemos que los datos tienen estructura de imagen, entonces podemos adaptar la red y mejorar los resultados.
- Ahora, supongamos que queremos aproximar una cierta función y conocemos alguna ecuación diferencial que satisface, ¿podemos integrar ese conocimiento a la red?
- Sí, es lo que se conoce como **Physics-Informed Neural Networks (PINNs)** y el *boom* de esta idea se dió después de [Raissi et al., 2019], aunque ideas similares ya existían antes.

Caso general

Sea $\Omega \subset \mathbb{R}^n$ un abierto acotado con frontera suave, supongamos que la función a aproximar $u : \overline{\Omega} \subset \mathbb{R}^n \to \mathbb{R}^m$ es suficientemente suave y satisface una ecuación diferencial de la forma

$$F(x,u) =: F(x, Du(x), D^2u(x), \ldots, D^k u(x)) = 0, \quad x \in \Omega$$

y eventualmente una condición de borde

$$G(x,u) =: G(x, Du(x), D^2u(x), \ldots, D^k u(x)) = 0, \quad x \in \partial\Omega$$

Cambiando la función de pérdida

Entonces, sean $N_{data}, N_{phys}, N_{bound} \in \mathbb{N}$ un conjunto de datos $\{(X_i^{data}, y_i^{data})\}_{i=1}^{N_{data}}$ tal que

$$u(X_i^{data}) = y_i^{data}, \quad i = 1, \dots, N_{data}.$$

y otros conjuntos $\{X_i^{phys}\}_{i=1}^{N_{phys}}$, $\{X_i^{bound}\}_{i=1}^{N_{bound}}$, que llamaremos puntos de colocación. Consideremos para números reales positivos $\lambda_{data}, \lambda_{phys}, \lambda_{bound}$, entonces la función de pérdida a minimizar es

$$\begin{aligned}\mathcal{L}(\Theta) = \lambda_{data} \sum_{i=1}^{N_{data}} \|NN_\Theta(X_i^{data}) - y_i^{data}\|^2 + \lambda_{phys} \sum_{i=1}^{N_{phys}} \|F(X_i^{phys}, NN_\Theta)\|^2 \\ + \lambda_{bound} \sum_{i=1}^{N_{bound}} \|G(X_i^{bound}, NN_\Theta)\|^2\end{aligned}$$

Intuición de la función de pérdida

Un buen óptimo de esa función de pérdida cumpliría minimizar:

- El error de ajustarse al conjunto de datos $\{(X_i^{data}, y_i^{data})\}_{i=1}^{N_{data}}$.
- El residuo de la dinámica.
- El residuo de la condición de borde.

Para ello es importante elegir bien los ponderadores $\lambda_{data}, \lambda_{phys}, \lambda_{bound}$. Veamos un resultado para EDO, ya que haremos un ejemplo con eso.

Resultado para EDO

Sea el siguiente problema de Cauchy

$$\dot{x}(t) = f(t, x(t)), \quad t \in [0, T], \quad x(0) = x_0$$

con f continua en su primera variable y Lipschitz de constante L en su segunda variable. Sea NN_Θ una red neuronal de parámetros Θ que intenta aproximar la solución al problema.
¿Se puede cuantificar el error que comete la red en base al error que comete al aproximar la condición inicial y la EDO? Notar que estas son las cantidades que se consideran en la función de pérdida.

Teorema de descomposición de error

Teorema (Descomposición de error para EDO [Hillebrecht, Unger, 2021])

Se definen el error de aproximar la condición inicial y el de aproximar el residuo de la ecuación como

$$e_{initial} = \|NN_\Theta(0) - x_0\|$$

$$e_{phys}(t) = \|\dot{NN}_\Theta(t) - f(t, NN_\Theta(t))\|, \quad t \in [0, T]$$

Sea $x : [0, T] \to \mathbb{R}^n$ la solución del problema de Cauchy, luego el error que comete la red neuronal cumple

$$\|x(t) - NN_\Theta(t)\| \leq e^{LT} e_{initial} + \int_0^t e^{L(t-s)} e_{phys}(s) ds$$

DeepONets

DIM Universidad de Chile

Más allá de una sola ecuación

- La esencia de una PINN es aproximar la solución para una ecuación en particular, ¿pero y si quisiéramos cambiar algo de la ecuación?
- En dicho caso sería ideal poder aproximar el operador completo en vez de la ecuación.
- Esto es lo que se conoce como **Deep Operator Network** y fueron presentadas por primera vez en [Lu, 2021].

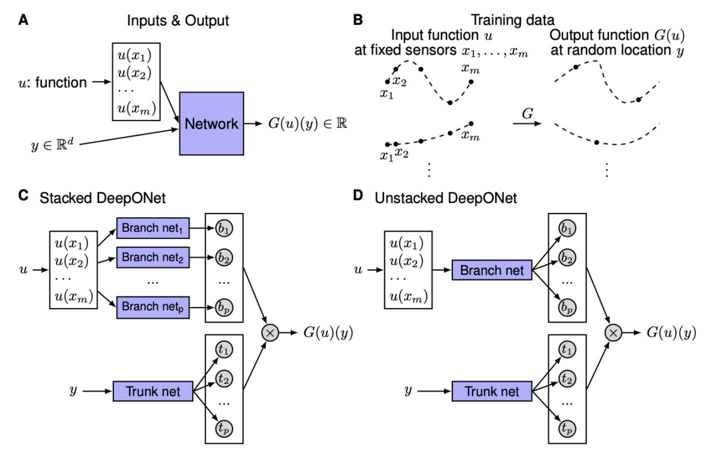
¿Cómo que operadores?

- Sea K un compacto en $\mathbb{R}^n$ y sea $G : C(K) \to C(K)$ un operador, por ejemplo G el operador tal que $G(f)$ es la solución de la ecuación

$$-\Delta u(x) = f(x), \quad x \in \text{int}(K); \quad u(x) = 0, \quad x \in \partial K$$

- Y esto puede ser muy general, no solo se restringe a operadores diferenciales, puede ser a muchos otros operadores.
- Notar que una DeepONet requiere un conjunto de entrenamiento mucho más complejo, ya que se debe entrenar con funciones.
- Veremos el esquema de entrenamiento y un teorema de aproximación universal en este contexto.

DeepONets in a nutshell



Teorema de aproximación

Teorema (Aproximación Universal para Operadores [Chen, Chen, 1995])

Supongamos que σ es una función continua no polinomial, X es un espacio de Banach, $K_1 \subset X$, $K_2 \subset \mathbb{R}^d$ compactos en X y $\mathbb{R}^d$, respectivamente, V es un conjunto compacto en $C(K_1)$, y G es un operador continuo que mapea V en $C(K_2)$. Entonces, para todo $\varepsilon > 0$, existen enteros n, p, m, constantes $c_i^k, \xi_{ij}^k, \theta_i^k, \zeta_k \in \mathbb{R}$, $w_k \in \mathbb{R}^d$, $x_j \in K_1$, donde $i = 1, \ldots, n$, $k = 1, \ldots, p$, $j = 1, \ldots, m$, tales que:

$$\left| G(u)(y) - \sum_{k=1}^{p} \sum_{i=1}^{n} c_i^k \, \sigma \left(\sum_{j=1}^{m} \xi_{ij}^k u(x_j) + \theta_i^k \right) \sigma(w_k \cdot y + \zeta_k) \right| < \varepsilon,$$

se cumple para todo $u \in V$ y $y \in K_2$.

Referencias I

Berner et al. (2019)

Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations

Journal of Computational Physics, Volume 378, Pages 686–707.

Hillebrecht, Unger (2021)

Learning nonlinear Certified machine learning: A posteriori error estimation for physics-informed neural networks.

2022 International Joint Conference on Neural Networks (IJCNN).

Lu et al. (2007)

Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators.

Nature Machine Intelligence, volume 3, pages 218–229.

Referencias II

Chen, Chen (1995)
Universal approximation to nonlinear operators by neural networks with arbitrary activation functions and its application to dynamical systems.
IEEE Transactions on Neural Networks, 6(4):911–917.

Ingeniería Matemática
FACULTAD DE CIENCIAS
FÍSICAS Y MATEMÁTICAS
UNIVERSIDAD DE CHILE

¡Gracias por su atención!

Joaquín Fontbona, Javier Maass y Diego Olguín

Departamento de Ingeniería Matemática
Universidad de Chile

18 de diciembre