
TRABAJO DE FIN DE GRADO EN MATEMÁTICAS

Departamento de Matemáticas

Universidad Autónoma de Madrid. Curso académico 2024-25

Título del proyecto: Redes Neuronales en la Aproximación de Ecuaciones en Derivadas Parciales // Neural Networks in the Approximation of Partial Differential Equations

Nombre y Apellidos: Luis Hebrero Garicano

Nombre del tutor(es): Julia Novo

INFORME INTERMEDIO ¹

1.- **Labor desarrollada hasta la fecha:** *(reuniones con el tutor; búsqueda de bibliografía; planteamiento de los objetivos.)*

Hasta ahora, el trabajo se ha centrado en estudiar métodos basados en redes neuronales para resolver ecuaciones en derivadas parciales (EDPs). Se ha revisado la literatura para entender los enfoques clásicos, como el método de elementos finitos (FEM), y compararlos con técnicas más recientes como las Physics-Informed Neural Networks (PINNs) y el método Deep Ritz.

También se han hecho implementaciones experimentales en Python, utilizando librerías especializadas en inteligencia artificial y métodos numéricos.

A lo largo del proyecto, ha habido reuniones con Julia en media una vez cada dos semanas, en las que hemos discutido avances, dificultades y posibles enfoques para seguir. Estas reuniones han sido clave para recibir feedback y ajustar la dirección del trabajo según lo necesario.

2.- **Esquema de los distintos apartados del trabajo:** *(puede usarse como guía la propia tabla de contenidos.)*

a) **Introducción y preliminares**

1) Introducción a las redes neuronales

a' Comentarios sobre la función sigmoide

2) Conceptos preliminares sobre las ecuaciones en derivadas parciales

3) Métodos numéricos tradicionales: el método de los elementos finitos

b) **Aproximación de EDPs mediante redes neuronales**

1) PINN: Physics-Informed Neural Networks

a' Introducción

b' Formulación de las PINN

2) El “Deep Ritz Method”

a' El problema elíptico autoadjunto

b' Formulación del método “Deep Ritz”

¹El informe debe ser elaborado por el estudiante y presentado al tutor -o tutores- que deberá dar su conformidad antes de ser entregado al coordinador.

3.- **Descripción del proyecto:** (*motivación; principales resultados y, en su caso, aplicaciones que se esperan obtener.*) Máximo 2 páginas.

- **Motivación:** Las ecuaciones en derivadas parciales (EDPs) desempeñan un papel esencial en la modelización de fenómenos físicos, económicos y biológicos. Tradicionalmente, se han resuelto mediante métodos numéricos como el método de elementos finitos. Sin embargo, en los últimos años, las redes neuronales han surgido como una alternativa prometedora.

Gracias a su capacidad para aproximar funciones arbitrarias, una red neuronal puede entrenarse para aproximar la solución de una EDP si se define adecuadamente la función de coste. En particular, enfoques como las Physics-Informed Neural Networks (PINNs) y el método Deep Ritz han despertado un gran interés en la comunidad científica.

Este proyecto analiza la aplicación de redes neuronales en la resolución de EDPs elípticas, centrándose en los métodos PINN y Deep Ritz. Se estudian sus limitaciones y los casos en los que no producen los resultados esperados, examinando las causas subyacentes de estos problemas.

- **Principales resultados:**

a) **Sobre la redes neuronales:**

Las redes neuronales son funciones que transforman una entrada en una salida a través de múltiples capas de neuronas. Cada capa aplica transformaciones mediante pesos y sesgos, seguidas de una función de activación, como la sigmoide, tangente hiperbólica o ReLU.

Se presenta una red neuronal concreta de cuatro capas, que clasifica puntos en un plano en tres categorías. Su entrenamiento consiste en ajustar los pesos y sesgos para minimizar el error entre la salida real y la deseada, utilizando una función de coste y optimización por descenso del gradiente.

Se destaca la importancia de las funciones de activación no lineales, demostrando que sin ellas, la red se reduce a una simple función lineal, lo que limita su capacidad de aproximación. Se menciona un resultado de Pinkus que establece que una red neuronal de una sola capa puede aproximar cualquier función continua si y solo si la función de activación no es polinómica.

b) **Sobre las EDPs:**

Este trabajo se centra en las ecuaciones en derivadas parciales (EDP) elípticas, que tienen una forma general definida por la ecuación

$$-\sum_{i,j=1}^n \frac{\partial}{\partial x_j} \left(a_{ij}(x) \frac{\partial u}{\partial x_i} \right) + \sum_{i=1}^n b_i(x) \frac{\partial u}{\partial x_i} + c(x)u = f(x), \quad x \in \Omega.$$

Donde los coeficientes $a_{ij}(x)$, $b_i(x)$, $c(x)$ y f cumplen ciertas condiciones de regularidad y una condición de elipticidad uniforme.

Se estudian problemas con condiciones de contorno de Dirichlet, donde la solución toma valores fijos en el borde del dominio.

Dado que en muchos casos no es posible encontrar soluciones clásicas (que satisfacen la ecuación en todo el dominio con derivadas en sentido fuerte), se introduce la noción de solución débil. Para ello, se definen la derivada débil y los espacios de Sobolev $H^1(\Omega)$ y $H_0^1(\Omega)$, que permiten trabajar con soluciones en un espacio funcional más amplio.

La formulación débil de la ecuación elíptica se establece mediante una forma bilineal $a(u, \varphi)$ y un funcional lineal $l(\varphi)$, lo que permite reformular el problema en términos más simples. Además, se demuestra la existencia y unicidad de soluciones débiles aplicando el teorema de Lax-Milgram, que garantiza la existencia de una única solución en $H_0^1(\Omega)$ bajo ciertas condiciones adicionales de elipticidad y regularidad de los coeficientes.

c) **Sobre los métodos numéricos tradicionales:**

Se basa en la formulación débil del problema y en la aproximación de la solución dentro de un subespacio de dimensión finita.

Para un problema elíptico con condiciones de contorno de Dirichlet homogéneas, se divide el dominio Ω en pequeños subdominios llamados elementos finitos. En una dimensión, estos son intervalos, mientras que en dos dimensiones pueden ser triángulos. Se define un subespacio V_h compuesto por polinomios a trozos, contruidos sobre una base de funciones que valen cero en los bordes del dominio.

El problema diferencial se reformula en este espacio finito, reduciendo la búsqueda de una solución a encontrar los coeficientes U_i en la combinación lineal de funciones base. Esto da lugar a un sistema lineal de la forma $AU = b$, donde A es la matriz de rigidez y b el vector de términos fuente. Este sistema se resuelve mediante métodos numéricos como la factorización LU, obteniendo una aproximación u_h de la solución u .

El método se generaliza a más dimensiones utilizando elementos finitos adecuados (triángulos en 2D, tetraedros en 3D) y bases de funciones apropiadas para cada caso.

d) Sobre las PINNs:

Las Physics-Informed Neural Networks (PINN) son una técnica que combina redes neuronales con la resolución de ecuaciones en derivadas parciales (EDP). La función de coste de las PINN se define como una combinación ponderada de dos términos: uno que minimiza el residuo de la ecuación diferencial en el dominio y otro que impone las condiciones de contorno.

Para calcular la función de coste, se utilizan puntos de colocación (collocation points) que discretizan la ecuación en el dominio y en la frontera. El entrenamiento de la red neuronal consiste en encontrar los pesos y sesgos que minimizan esta función de coste mediante métodos de optimización como Adam o L-BFGS.

Se presentan ejemplos de aplicación, incluyendo la ecuación de Poisson en 1D, resolviéndola con la librería DeepXDE. Sin embargo, se demuestra que las PINN pueden fallar en ciertos problemas, especialmente cuando el problema no tiene solución fuerte. En estos casos, la red neuronal no logra aproximar correctamente la solución.

Los principales problemas de las PINN incluyen:

- Falta de unicidad en la solución, ya que el entrenamiento es un problema de optimización no convexo.
- Falta de criterios teóricos para elegir los hiperparámetros óptimos.
- Aproximación del residuo en forma fuerte, lo que impide capturar soluciones débiles cuando estas existen pero no son diferenciables en el sentido clásico.

e) Sobre el método Deep Ritz:

El método “Deep Ritz” utiliza redes neuronales para resolver ecuaciones en derivadas parciales (EDP) mediante su formulación débil, evitando las restricciones de las PINN. Se basa en minimizar un funcional que caracteriza la solución del problema elíptico autoadjunto. La red neuronal aproxima la solución, y su entrenamiento consiste en minimizar este funcional mediante algoritmos de optimización por gradiente.

4.- Grado de consecución de los objetivos y plan de trabajo a desarrollar en la segunda mitad del periodo:

Los objetivos iniciales del trabajo se han cumplido en su mayoría. Durante el resto del semestre, me centraré en formalizar algunos de los conceptos y en ampliar los ejemplos implementando algunas ecuaciones en Python para evaluar el desempeño de los métodos estudiados.

5.- Bibliografía usada hasta la fecha o que se piensa utilizar:

Se van a utilizar todas las siguientes referencias como mínimo. [1],[5][11][7][9][12][6][2][4][8][3][10]

Referencias

- [1] Yuri Aikawa, Naonori Ueda y Toshiyuki Tanaka. “Improving the efficiency of training physics-informed neural networks using active learning”. En: *New Generation Computing* (2024), págs. 1-22.
- [2] C. *Possible Failure Modes in Physics-Informed Neural Networks*. <https://github.com/a1k12/characterizing-pinns-failure-modes>. 2021.
- [3] Weinan E y Bing Yu. *The Deep Ritz method: A deep learning-based numerical algorithm for solving variational problems*. 2017. arXiv: 1710.00211 [cs.LG]. URL: <https://arxiv.org/abs/1710.00211>.
- [4] Tamara G. Grossmann et al. *Can Physics-Informed Neural Networks beat the Finite Element Method?* 2023. arXiv: 2302.04107 [math.NA]. URL: <https://arxiv.org/abs/2302.04107>.
- [5] Jie Hou, Ying Li y Shihui Ying. “Enhancing PINNs for solving PDEs via adaptive collocation point movement and adaptive loss weighting”. En: *Nonlinear Dynamics* 111.16 (2023), págs. 15233-15261.
- [6] Aditi S. Krishnapriyan et al. *Characterizing possible failure modes in physics-informed neural networks*. 2021. arXiv: 2109.01050 [cs.LG]. URL: <https://arxiv.org/abs/2109.01050>.
- [7] Lu Lu et al. “DeepXDE: A deep learning library for solving differential equations”. En: *SIAM Review* 63.1 (2021), págs. 208-228. DOI: 10.1137/19M1274067.
- [8] Tao Luo y Qixuan Zhou. *On Residual Minimization for PDEs: Failure of PINN, Modified Equation, and Implicit Bias*. 2023. arXiv: 2310.18201 [math.AP]. URL: <https://arxiv.org/abs/2310.18201>.
- [9] Takashi Matsubara y Takaharu Yaguchi. *Good Lattice Training: Physics-Informed Neural Networks Accelerated by Number Theory*. 2023. arXiv: 2307.13869 [cs.LG]. URL: <https://arxiv.org/abs/2307.13869>.
- [10] Marcus Münzer y Chris Bard. *A Curriculum-Training-Based Strategy for Distributing Collocation Points during Physics-Informed Neural Network Training*. 2022. arXiv: 2211.11396 [cs.LG]. URL: <https://arxiv.org/abs/2211.11396>.
- [11] Allan Pinkus. “Approximation theory of the MLP model in neural networks”. En: *Acta numerica* 8 (1999), págs. 143-195.
- [12] Shashank Subramanian et al. *Adaptive Self-supervision Algorithms for Physics-informed Neural Networks*. 2022. arXiv: 2207.04084 [cs.LG]. URL: <https://arxiv.org/abs/2207.04084>.