

REDES NEURONALES INFORMADAS POR FÍSICA - UN NUEVO CAMPO EN LA EDUCACIÓN DE LA INGENIERÍA MECÁNICA

PHYSICS-INFORMED NEURAL NETWORKS - A NEW FIELD IN MECHANICAL ENGINEERING EDUCATION

Christian Díaz-Cuadro y Santiago Correa

Instituto de Ingeniería Mecánica y Producción Industrial, Facultad de Ingeniería, Universidad de la República, Uruguay <http://www.fing.edu.uy>

Palabras clave: Enseñanza en la ingeniería; Redes Neuronales informadas por Física; Ingeniería Mecánica.

Resumen. Este trabajo presenta un breve taller para la enseñanza de métodos numéricos en la Facultad de Ingeniería de la Universidad de la República (UdelaR) que integra redes neuronales profundas y, en particular, PINNs como complemento de los enfoques tradicionales. La propuesta combina módulos semanales con clases y *notebooks* interactivos, guiando desde una red neuronal básica hasta aplicaciones de PINNs a problemas gobernados por ecuaciones diferenciales. Los casos abarcan: ecuación de un sistema físico clásico con solución analítica, transferencia de calor 1D, mecánica de sólidos lineal 2D y fluido-dinámica incompresible 2D. Se enfatizan conceptos como normalización y adimensionalización de datos, ponderación adaptativa de funciones de pérdida y diseño de condiciones de frontera. La evaluación consiste en un proyecto grupal que resuelve un *benchmark* y se contrasta con resultados obtenidos por simulación numérica. El taller articula modelado físico y aprendizaje automático, fortaleciendo la programación científica, el análisis crítico y el trabajo colaborativo, ofreciendo simultáneamente lineamientos transferibles para la enseñanza de la ingeniería.

Keywords: Engineering Education; Physics-Informed Neural Networks (PINNs); Mechanical Engineering.

Abstract. This paper presents a short workshop for teaching numerical methods at the School of Engineering of Universidad de la República (UdelaR) that integrates deep neural networks and, in particular, PINNs as a complement to traditional approaches. The workshop combines weekly modules with lectures and interactive *notebooks*, guiding learners from a basic neural network to PINN applications for differential-equation problems. Case studies include: a classical physical system with a known analytical solution, 1D heat transfer, 2D linear solid mechanics, and 2D incompressible fluid dynamics. We emphasize variable normalization and non-dimensionalization, adaptive loss weighting, and boundary-condition design. Assessment is a group project that solves a benchmark and contrasts results with numerical simulation. The workshop links physical modeling and machine learning, strengthens scientific programming, critical analysis, and collaborative work, and offers transferable guidelines for engineering education.

1. INTRODUCCIÓN

La incorporación de modelos de aprendizaje automático en combinación con el conocimiento físico ha dado lugar, en los últimos años, al área emergente del *scientific machine learning*. En este marco, las *physics-informed neural networks* (PINNs) se han consolidado como una herramienta prometedora para resolver problemas descritos por ecuaciones en derivadas parciales (PDEs). A diferencia de los enfoques puramente basados en datos, las PINNs imponen de manera explícita las leyes físicas que gobiernan el sistema sobre la salida de la red neuronal, garantizando soluciones consistentes con los modelos matemáticos subyacentes (Raissi et al., 2019). El potencial de las PINNs se debe, en gran medida, a los avances recientes en capacidad de cómputo, algoritmos de entrenamiento y métodos de diferenciación automática. Si bien la idea de utilizar redes neuronales para aproximar soluciones de PDEs surgió en la década de 1990, solo en la actualidad se dispone de las herramientas necesarias para aplicarla de manera práctica y eficiente en problemas complejos. Entre sus ventajas destaca la posibilidad de obtener soluciones continuas en todo el dominio espacio-temporal sin recurrir a mallas rígidas, lo que reduce el costo computacional y permite trabajar con puntos de entrenamiento distribuidos de forma irregular. Asimismo, como plantean Cuomo et al. (2022), las PINNs ofrecen un marco flexible para integrar datos experimentales o de simulación en el proceso de entrenamiento. Sobre esta base conceptual surge el desafío de cómo introducir a los estudiantes a una herramienta computacional con gran proyección y potencial de desarrollo en los próximos años. En este trabajo se presenta una experiencia docente orientada a la enseñanza de PINNs como complemento de los métodos numéricos tradicionales. La propuesta se materializa en un taller de corta duración que combina clases expositivas con instancias prácticas en entornos de programación interactivos. A lo largo de distintos módulos, los estudiantes utilizan notebooks diseñados para fomentar una comprensión activa tanto de los fundamentos teóricos como de su implementación práctica, abordando progresivamente problemas de mayor complejidad: desde ecuaciones con solución analítica hasta aplicaciones en transferencia de calor, mecánica del sólido y dinámica de los fluidos. Con el avance del curso se incorporan progresivamente técnicas adicionales, tales como la normalización y adimensionalización de variables, la formulación de condiciones de frontera y el uso de penalizaciones adaptativas en la función de pérdida. La instancia final de evaluación se basa en un trabajo grupal, donde cada equipo debe aplicar lo aprendido en la resolución de un problema específico planteado por el cuerpo docente, integrando tanto los aspectos teóricos como las consideraciones prácticas de la implementación. Esta experiencia busca no solo introducir a los estudiantes al uso de PINNs, sino también fortalecer competencias en programación científica, análisis crítico, modelado físico y resolución colaborativa de problemas. De esta manera, el curso promueve el uso de metodologías modernas que amplían el repertorio de herramientas para enfrentar problemas de ingeniería y ciencias aplicadas.

2. CONTEXTO INSTITUCIONAL

El Instituto de Ingeniería Mecánica y Producción Industrial (IIMPI) de la Facultad de Ingeniería, Udelar, es responsable de la formación de grado y posgrado en Ingeniería Industrial Mecánica, y desarrolla docencia, investigación y asesoramiento en áreas clave como diseño mecánico, termodinámica aplicada y producción industrial. En este marco, el Departamento de Diseño Mecánico (DDM) dicta cursos de diseño, mecánica del sólido, robótica, instrumentación y control, y en los últimos años impulsó la incorporación de herramientas de cálculo numérico

para fortalecer competencias digitales y computacionales.

Como parte de esta innovación pedagógica y alineado con las áreas del DDM, se diseñó un taller breve para introducir redes neuronales aplicadas a problemas de ingeniería mecánica. Esta propuesta, inédita en la carrera, busca integrar herramientas de *Machine Learning* al aprendizaje de la mecánica y potenciar la resolución de problemas, lo que exigió ajustar modalidad y metodología de enseñanza para favorecer una comprensión integral y explorar el potencial de estas técnicas en el ámbito de la ingeniería.

3. METODOLOGÍA

La metodología empleada integra elementos de enfoques ampliamente discutidos en la literatura educativa. En primer lugar, se vincula con las estrategias de *active learning*, entendidas como aquellas que buscan la participación activa del estudiante en actividades significativas, rompiendo con la pasividad del modelo expositivo tradicional (Prince, 2004). En este sentido, el uso de *notebooks* interactivos con códigos incompletos y la resolución guiada de problemas constituyen instancias de involucramiento activo que promueven la construcción de conocimiento y el razonamiento crítico. Asimismo, el trabajo en pequeños grupos para discutir e implementar las secciones faltantes del código refleja principios de *collaborative* y *cooperative learning*, los cuales han demostrado beneficios en la retención, la motivación y el desarrollo de habilidades interpersonales en contextos de ingeniería (Prince, 2004). Por otra parte, el taller guarda una estrecha relación con el *Problem-Based Learning* (PBL), dado que cada sesión se organiza a partir de un problema de ingeniería que actúa como eje articulador del aprendizaje. Esta estrategia, en línea con lo señalado por Barrows (1996), fomenta tanto la integración de conocimientos de distintas áreas como la adquisición de competencias en auto-aprendizaje, resolución de problemas y trabajo en equipo, consideradas fundamentales en la formación de ingenieros. El taller se desarrolla a lo largo de cinco sesiones de tres horas cada una, en una modalidad híbrida que combina la exposición teórica con actividades prácticas en computadora. La organización de los contenidos fue diseñada con el objetivo de que estudiantes con conocimientos básicos de programación, pero con una formación avanzada en diversas áreas de la ingeniería mecánica (mecánica del sólido, transferencia de calor, dinámica de fluidos, entre otras), sean capaces de seguir adecuadamente el curso y aprovechar su desarrollo. La propuesta pedagógica se fundamenta en una construcción progresiva del conocimiento, estructurada de manera escalonada: en primer lugar, se presenta la herramienta; posteriormente se aplica en un contexto específico; luego se modifica para explorar variaciones; y finalmente, se refuerza mediante la práctica y la repetición. Con esta estrategia, se busca que los estudiantes no solo adquieran conceptos y metodologías, sino que también desarrollen la capacidad de adaptarlos a diferentes contextos a partir de un razonamiento analítico aplicado a cada sistema particular. Cada tema se introduce a través de un caso de estudio vinculado a un problema de ingeniería. Esta metodología permite incrementar de manera simultánea la complejidad y la capacidad de la herramienta presentada y la dificultad del problema abordado, favoreciendo así un aprendizaje progresivo y contextualizado.

3.1. Formato de código interactivo - Jupyter Notebook

Para la experimentación computacional se recurre a recursos gratis y de fácil acceso en la nube, en particular a la plataforma proporcionada por Google (2024), Google Colab, para la cual el único requerimiento es poseer una cuenta de Google. Este entorno permite a los usuarios

escribir y ejecutar código Python a través de su navegador web, ofreciendo acceso a recursos computacionales como GPUs, sin necesidad de instalar software en la PC. En cada sesión, se presenta el caso de estudio y se analizan las características físicas que difieren del caso anterior para planificar el abordaje que se realizará sobre el mismo. A continuación se entrega a los estudiantes un conjunto de *notebooks* con códigos incompletos en secciones consideradas fundamentales, que deben ser completados tanto de manera individual como colaborativa, discutiendo cómo implementar los vacíos. Este enfoque promueve la integración activa e inmediata de los conceptos y herramientas trabajados en clase, dado que los vacíos en los códigos se corresponden directamente con las nuevas herramientas incorporadas en cada sesión. Así se espera consolidar el aprendizaje a través de la práctica guiada y el trabajo autónomo, enfocado principalmente en los conceptos que se espera que el estudiante adquiera en esta etapa, sin que resulte particularmente intenso desde el punto de vista de la programación. En este contexto, los *Jupyter Notebooks* resultaron especialmente adecuados porque reúnen en un mismo entorno código ejecutable, texto, ecuaciones y visualizaciones. Como señala [Granger y Pérez \(2021\)](#), su flexibilidad los convierte en una herramienta idónea para la enseñanza en disciplinas STEM (Ciencia, Tecnología, Ingeniería y Matemáticas), al promover el pensamiento computacional y la comunicación de resultados mediante narrativas reproducibles. En el taller, esta integración permitió a los estudiantes comprender el problema, implementar soluciones y documentar su razonamiento de forma práctica, fortaleciendo el vínculo entre teoría y aplicación. Además, el uso de *notebooks* puso en primer plano el rol activo del estudiante, la resolución de problemas reales y la apropiación de herramientas que orientan eficazmente la construcción del conocimiento.

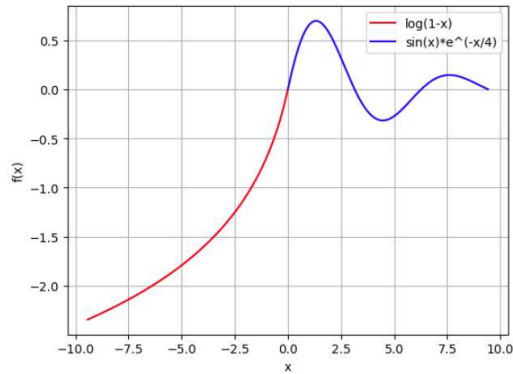
3.2. Diseño de Casos de estudio para clases

Los casos diseñados para el aprendizaje se fundamentaron en referencias bibliográficas específicas de cada tema y en datos generados a partir de simulaciones numéricas, empleando tanto software comercial como herramientas de desarrollo propio, previamente validadas. Esta estrategia garantizó la calidad y confiabilidad de la información utilizada, así como su pertinencia en el contexto de enseñanza. A partir de estos insumos, se construyó un conjunto de problemas con datos disponibles que actúan como base común para la experimentación de los estudiantes. La selección de los casos buscó cubrir diferentes niveles de complejidad, desde problemas introductorios con soluciones analíticas conocidas hasta situaciones más avanzadas que requieren el uso de métodos numéricos. De este modo, los estudiantes pudieron transitar gradualmente desde la comprensión conceptual hacia la resolución de escenarios cercanos a la realidad de la ingeniería. El diseño de los casos también consideró criterios pedagógicos, tales como la posibilidad de fragmentar los problemas en etapas y fomentar la discusión grupal. Además, se priorizó que los problemas enmarquen casos bases de cada disciplina de la ingeniería mecánica, de modo que los estudiantes pudieran desarrollar competencias para analizar, interpretar y validar resultados, en las distintas áreas de actuación. En el siguiente apartado se detalla la distribución de contenidos por sesión donde se aprecia la evolución de la temática.

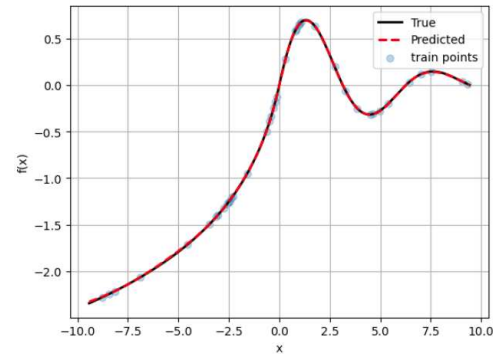
Clase 1: Introducción a las Redes Neuronales

La primera clase introdujo los fundamentos del aprendizaje automático y de las redes neuronales *feed-forward*. Se revisaron el perceptrón, los pesos y el sesgo, las funciones de activación y la arquitectura en capas (entrada, ocultas y salida). Además, se explicó la propagación hacia adelante (*forward propagation*), la función de pérdida y el cálculo de gradientes mediante re-

tropagación automática (*automatic differentiation*) dentro de un grafo computacional. También se presentaron hiperparámetros relevantes—tasa de aprendizaje, optimizadores (ADAM, BFGS) y número de épocas—y su impacto en el entrenamiento. La sesión culminó con un ejercicio práctico: aproximar una función discontinua compuesta por un tramo logarítmico para valores negativos y un tramo senoidal de amplitud decreciente para valores positivos. En la Fig. 1 se muestra la función partida (Fig. 1a), los puntos de entrenamiento y la predicción de la red (Fig. 1b).



(a) Problema matemático propuesto.

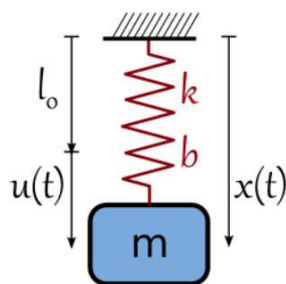


(b) Predicción de la red neuronal.

Figura 1: Problema de la clase 1

Clase 2: Introducción a PINNs mediante un Oscilador

En la segunda clase se introdujo el concepto de PINNs, enfatizando la incorporación explícita de información física en el entrenamiento de redes neuronales. Como caso de estudio, se trabajó con el problema del oscilador armónico amortiguado que se aprecia en la Fig. 2. Este problema tiene una solución analítica conocida que se presenta en la Ec. 1 para las condiciones iniciales definidas. Junto con los estudiantes se llegó a esta solución que luego fue utilizada para la generación sintética de datos para el entrenamiento. Además, se abordó la identificación de parámetros y el proceso de adimensionalización del sistema, analizando como la incorporación de esta estrategia favorece el entrenamiento de la red.

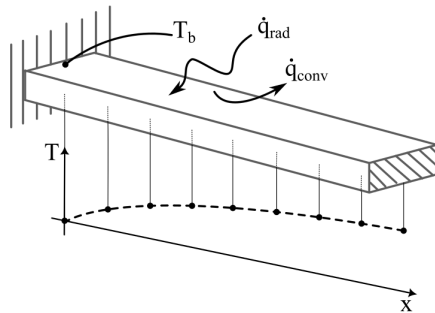


$$u(t) = \frac{g}{\omega^2} \left(1 - \frac{e^{-\xi\omega t} \cos(\omega t + \phi)}{\cos(\phi)} \right) \quad (1)$$

Figura 2: Oscilador.

Clase 3: Conducción de Calor 1D en Aletas

La tercera sesión se centró en un problema de transferencia de calor en aletas, en estado estacionario, considerando la temperatura uniforme en cada sección y pérdidas de calor por convección y radiación, como se muestra en el esquema presentado en la Fig. 3.



$$\frac{\partial}{\partial x^*} \left((1 + \epsilon_C \theta) \frac{\partial \theta}{\partial x^*} \right) - N^2 \theta + N^2 G (1 + \epsilon_G \theta) = 0 \quad (2)$$

Figura 3: Problema de Conducción de Calor 1D en Aletas.

Para este caso se siguió uno de los ejemplos presentados por [Oommen y Srinivasan \(2022\)](#). Se discutió la formulación física del sistema y su resolución, para luego trasladarlo al marco de las PINNs. Se trabajó con la forma adimensionalizada de la ecuación de calor para una aleta según se define en la Ec. 2, la implementación de condiciones de borde y la comparación de distintos optimizadores (L-BFGS y ADAM). Asimismo, se introdujo la importancia de la normalización de los datos de entrada y la identificación de parámetros a partir del ajuste del modelo. (Por detalles sobre el modelado físico del sistema se refiere al lector a [Oommen y Srinivasan \(2022\)](#)).

Clase 4: Problema Mecánica del Sólido en 2D

La cuarta clase abordó un caso de mecánica de sólidos: una placa en estado plano de tensiones, sometida a una carga distribuida en el extremo, como se muestra en la Fig. 4. En trabajo conjunto con el grupo de estudiantes se revisó la formulación física que gobierna un problema de mecánica del sólido lineal, definida por la Ec. 3.

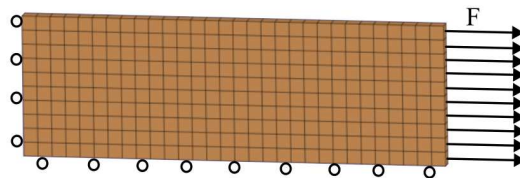


Figura 4: Problema de Mecánica del sólido 2D.

$$\nabla \cdot \sigma = 0 \quad \text{donde: } \sigma = \begin{bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \tau_{xy} \end{bmatrix} = \frac{E}{1 - \nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1 - \nu}{2} \end{bmatrix} \cdot \begin{bmatrix} \partial u / \partial x \\ \partial v / \partial y \\ \partial u / \partial y + \partial v / \partial x \end{bmatrix} \quad (3)$$

A partir de la simulación realizada con el Software FEBio Studio se obtuvieron los datos de referencia (validados), que se emplearon para el cálculo y la comparación de errores. Paralelamente se implementó la red neuronal correspondiente y fueron aplicadas las condiciones de borde pertinentes al problema estructural. Esta instancia permitió profundizar en la identificación de parámetros y en técnicas de normalización de datos.

Clase 5: Flujo en Canal 2D Transitorio

La última sesión del taller estuvo orientada al abordaje de un problema de dinámica de fluidos: el flujo en un canal bidimensional con obstáculo en régimen transitorio, problema que es presentado en la Fig. 5. Este caso se fundamenta en el trabajo presentado por [Raissi et al. \(2019\)](#), cuya solución de referencia se obtiene mediante simulación numérica. Considerando las simplificaciones de bidimensionalidad y la hipótesis de flujo incompresible, las ecuaciones de Navier-Stokes adimensionalizadas quedan definidas por las Ecs. 4. Sin embargo, para las condiciones de borde y las iniciales del sistema en este caso se utilizó una estrategia diferente que implica el uso de datos de simulación. Es decir, en lugar de forzar a que la red aprenda la física en los bordes (temporal y espacial), se la obliga a ajustar datos en estas circunstancias, mostrando una alternativa factible para la resolución de problemas.

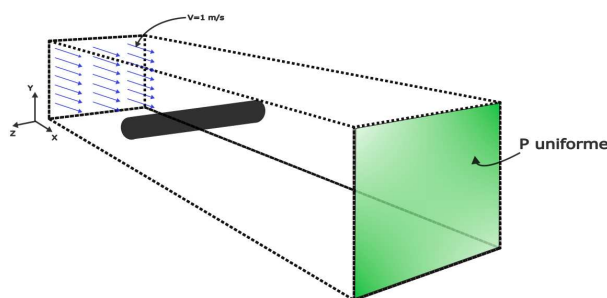


Figura 5: Canal de flujo bidimensional con obstáculo.

$$\mathbf{u}^* \cdot \nabla \mathbf{u}^* = -\nabla p^* + \frac{1}{\text{Re}} \nabla^2 \mathbf{u}^*, \quad \nabla \cdot \mathbf{u}^* = 0 \quad (4)$$

En esta instancia se integraron de manera conjunta todas las herramientas previamente presentadas a lo largo del curso: la formulación del problema físico, la implementación de condiciones de borde e iniciales, y la normalización de los datos. Asimismo, se introdujo la utilización de pesos adaptativos en la función de pérdida, subrayando su relevancia para equilibrar los distintos términos durante el entrenamiento en problemas complejos de fluido dinámica.

La disponibilidad de estos casos constituye una base sólida para futuros talleres, ya que permite reutilizar, adaptar y expandir los problemas hacia nuevos contextos. A partir de la experiencia adquirida, es posible orientar la propuesta hacia desafíos más complejos, cuyo proceso de resolución se encuentre documentado y validado, asegurando así la continuidad y escalabilidad del aprendizaje.

4. EVALUACIÓN DEL CURSO

La evaluación del curso se estructuró en formato de proyecto, donde los estudiantes debieron resolver un problema de ingeniería planteado por el equipo docente a partir de datos obtenidos

mediante simulaciones numéricas. Se propusieron tres problemas de referencia: uno asociado a la transferencia de calor bidimensional en estado transitorio en un sólido isotrópico presentado en la Fig. 6(a), otro problema de fluido dinámica en un canal con un escalón que se muestra en la Fig. 6(b), y el último relacionado a un problema de mecánica del sólido tridimensional bajo cargas dinámicas de compresión presentado en la Fig. 6(c). El número de estudiantes en esta

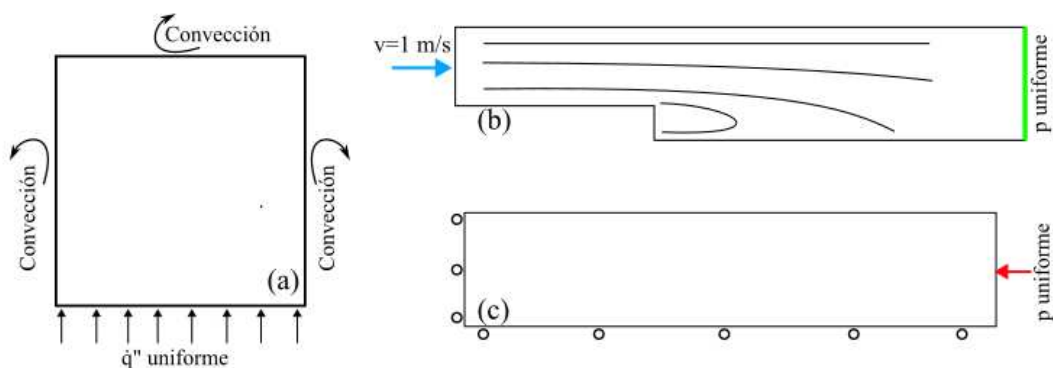


Figura 6: Ejemplos de problemas físicos abordados: (a) transferencia de calor, (b) fluidodinámica, y (c) mecánica del sólido.

primera oportunidad fue reducido por lo que se optó por armar dos grupos de estudiantes (uno de tres, grupo G1, y otro de dos, grupo G2) para trabajar cada uno en un problema a elección. El grupo G1 seleccionó el problema de termodinámica mientras que el grupo G2 decidió trabajar en el problema de mecánica del sólido. Cada proyecto incluyó un conjunto de datos base (temperaturas o desplazamientos, según el caso) y una serie de tareas progresivas. Las pautas se pueden encontrar en el siguiente [link](#). Estas abarcaron desde la representación gráfica de resultados hasta la implementación de PINNs con distintas configuraciones, incorporando técnicas como normalización, adimensionalización y asignación de pesos adaptativos en la función de pérdida, tópicos vistos e implementados en clase. Asimismo, se solicitó la identificación de parámetros físicos relevantes (por ejemplo, difusividad térmica, módulo de Young y coeficiente de Poisson) y el análisis de robustez frente a la incorporación de ruido en los datos. El trabajo final consistió en la entrega de los códigos desarrollados y un informe en formato PDF donde los estudiantes documentaron las técnicas utilizadas, la arquitectura y los hiperparámetros seleccionados, la evolución de las funciones de pérdida y métricas de desempeño (error relativo y absoluto). Además, se requirió un análisis crítico de los resultados obtenidos y una conclusión que integrara los aprendizajes alcanzados. Este formato de evaluación, basado en la resolución autónoma y colaborativa de problemas abiertos, buscó fomentar la aplicación práctica de las herramientas introducidas durante el taller, al tiempo que promovió el desarrollo de competencias analíticas, de investigación y comunicación técnica en los estudiantes.

5. RESULTADOS OBTENIDOS Y VALORACIÓN DE LA EXPERIENCIA

A modo ilustrativo se presentan algunos resultados de ambos equipos, los cuales cumplieron satisfactoriamente con la pauta del problema. En ambos casos, el flujo de trabajo incluyó: (i) preprocesamiento de los datos para entender la estructura y evolución de las variables; (ii) desarrollo de una PINN base para predecir el campo de temperatura o los desplazamientos (según el caso); y (iii) aplicación de las estrategias de mejora vistas en el taller, que fueron elevando gradualmente la calidad de las predicciones. En particular, el grupo G1 (transferencia de calor),

tras adimensionalizar la ecuación gobernante, normalizar los datos y sintonizar los pesos de las funciones de pérdida, alcanzó errores máximos del 8 % en el campo de temperaturas (Fig. 7). Además, en un estudio adicional con distintos niveles de ruido (no mostrado aquí), la precisión se mantuvo razonable, sin superar el 15 % de error en el peor de los casos (5 % de ruido gaussiano).

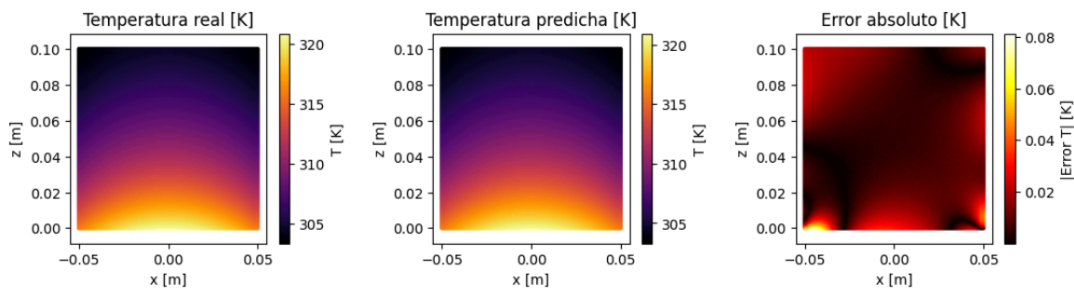


Figura 7: Predicción obtenida por los estudiantes del grupo G1

Por otro lado, el grupo G2 que trabajó con el problema de dinámica de un sólido tridimensional a compresión, también presenta resultados progresivamente mejores, hasta alcanzar un error máximo en la predicción de los desplazamientos de 10 % como se muestra en la Fig. 8. Estos resultados se obtuvieron una vez normalizados los datos de entrada e implementar un algoritmo de pesos adaptativos para las funciones de pérdida.

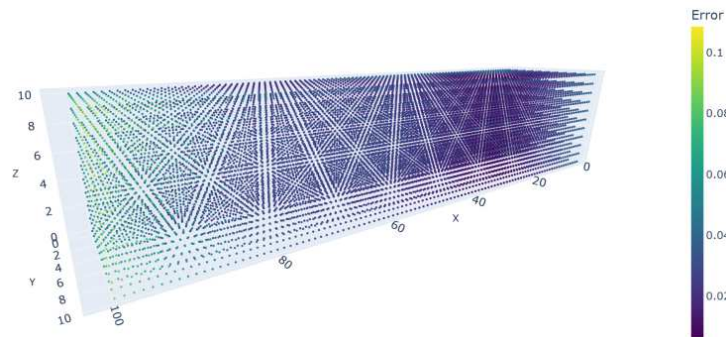


Figura 8: Predicción obtenida por los estudiantes del grupo G2

6. CONCLUSIONES

Previo a la evaluación de los informes entregados por los estudiantes, la valoración docente relativa del taller es muy positiva. Aunque, como primera experiencia, hay aspectos a mejorar, se destacan:

- Un acercamiento temprano de los estudiantes a una temática que parecía lejana y hoy acorta brechas, ampliando recursos para resolver problemas diversos.
- Un taller *hands on*, pensado para transferir de inmediato lo aprendido a casos reales de ingeniería mecánica.

- Alto potencial de escalado por su transversalidad y adaptación: basta cambiar los casos; la teoría de *machine learning* subyacente permanece.
- Notorio interés en clase, que aunque es una variable inmensurable, queda evidenciado por la disposición al trabajo y la curiosidad mostrada por los participantes en clase.

Por otro lado, en lo que respecta a los trabajos presentados por los grupos, ambos informes muestran una gran dedicación. No solo por la cantidad y calidad del producto en sí, sino también al considerar que se los está evaluando en un área completamente nueva para los estudiantes de la carrera de Ingeniería Industrial Mecánica de la Facultad de Ingeniería de la UdelAR, por lo que resultan aún más valiosos cada uno de los resultados. Ambos grupos presentan informes completos y detallados que cumplen la pauta y guían al lector por un desarrollo incremental de la solución, mejorado con las técnicas vistas en clase: primero resuelven el campo de la variable de interés incorporando la física en todo el rango temporal; luego aplican normalización/adimensionalización; y, finalmente, ponderan las funciones de pérdida para balancear la atención del algoritmo. Estas estrategias abordan las principales áreas de mejora de una Red Neuronal Informada por Física, evidenciando que los estudiantes no solo incorporaron una nueva herramienta para enfrentar problemas de ingeniería, sino que también desarrollaron criterios para mejorarlas en sus aspectos clave.

AGRADECIMIENTOS

Los autores agradecen a Felipe Caltieri, Nicole Castroman, Joaquín Laventure, Agustín Lopez y Lucas Mendoza por participar de este taller con tal excelencia.

REFERENCIAS

- Barrows H.S. Problem-based learning in medicine and beyond: A brief overview. *New directions for teaching and learning*, 1996(68):3–12, 1996. <http://doi.org/10.1002/tl.37219966804>.
- Cuomo S., Cola V.S.D., Giampaolo F., Rozza G., Raissi M., y Piccialli F. Scientific machine learning through physics-informed neural networks: Where we are and what's next. *CoRR*, abs/2201.05624, 2022. <http://doi.org/10.1007/s10915-022-01939-z>.
- Google. Google colabory. <https://colab.research.google.com/>, 2024. Recuperado el 18 de abril de 2024.
- Granger B. y Pérez F. Jupyter: Thinking and storytelling with code and data. En *Proceedings of the 26th ACM SIGPLAN conference on programming language design and implementation*, páginas 1–7. 2021. <http://doi.org/10.1109/MCSE.2021.3059263>.
- Oommen V. y Srinivasan B. Solving inverse heat transfer problems without surrogate models: A fast, data-sparse, physics informed neural network approach. *Journal of Computing and Information Science in Engineering*, 22(4):041012, 2022. ISSN 1530-9827. <http://doi.org/10.1115/1.4053800>.
- Prince M. Does active learning work? a review of the research. *Journal of engineering education*, 93(3):223–231, 2004. <http://doi.org/10.1002/j.2168-9830.2004.tb00809.x>.
- Raissi M., Perdikaris P., y Karniadakis G.E. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019. <http://doi.org/10.1016/j.jcp.2018.10.045>.