B”H

Universidad de Buenos Aires

Facultad de Ciencias Exactas y Naturales

Departamento de Matemática

Tesis de Licenciatura

Resolución de ecuaciones diferenciales elípticas con redes neuronales: formulación variacional y convergencia

Damián Yjilioff

Director: Ignacio Ojea

Fecha de presentación: 18/06/2025

Agradecimientos

Quiero dedicar esta sección para agradecer a todas las personas que me acompañaron a lo largo de esta carrera y que, sin ellas, yo no hubiera podido llegar a donde llegué. Primero quiero agradecer a mi papá, que siempre estuvo ayudándome con sus consejos, con su estabilidad y su preocupación. Eso hizo que pudiera concentrarme solamente en la carrera y no en otras cosas, y —sobre todo— de la mejor forma. A mi mamá, que siempre estuvo bancándome en el día a día mientras estudiaba, ayudándome con las cosas a las que no llegaba y haciéndome mil favores, todos los días. A mi hermana, mi abuela, mi familia, que siempre estuvo escuchando mis quejas y preguntándome cosas. Realmente hicieron que me sintiera acompañado.

También, toda esta carrera no hubiera sido posible sin la ayuda de Ceci. Realmente creo que es gracias a ella que sé lo que sé. Incontables veces me ayudó con ejercicios, explicaciones de temas, llamadas por Zoom para estudiar, en la facultad, preguntas a cualquier hora. No solo hizo posible el título, sino que lo hizo disfrutable y lleno de un sentimiento de compañía. También quería agradecerles a mis compañeros Nacho, Facu, Franco, Martín, Chanu, Jero: gente que me hacía reír cada vez que los encontraba, y no solo eso, sino que también siempre me ayudaban con preguntas y con la cantidad de molestias que yo les provocaba por las mil preguntas a las 12 de la noche.

A mis amigos de toda la vida, que me acompañaron en todo este camino y que siempre se alegraron con cada mini victoria que fui teniendo: Valen, Yaco, Teo, Juli, Juan, Tobi, Deibo, Arie, Lauty, Sanca, Ian, Nicky, Martu. A mis amigos de Anoji, en especial a Macha y a Eli, que me bancaron todo el último trayecto, y a mis amigos de Migdal.

Por último, me gustaría agradecer a la Facultad de Ciencias Exactas de la UBA, que me dejó cursar toda una carrera excelente, con los mejores docentes, de la mejor forma, y brindarme un sinfín de oportunidades en la vida. También quiero agradecer a todos los profesores que tuve a lo largo de la carrera, que en cada oportunidad en la que los molestaba con preguntas y dudas, siempre me respondieron de la mejor forma y manera. En especial al Doctor Ignacio Ojea, que gracias a él y a su continua ayuda y enseñanza, pude terminar con este último proyecto.

Índice

1 Introducción

En el marco del análisis numérico, es una práctica habitual —y hasta elemental— buscar alguna estrategia astuta para mallar el dominio de estudio de la manera más eficiente posible. Esto implica tener en cuenta tanto la geometría del dominio como las características del algoritmo a implementar. Sin embargo, surge una pregunta natural: ¿qué tipo de estrategias podemos adoptar cuando tratamos con dominios cuyo mallado es particularmente difícil o incluso inviable, ya sea por su complejidad geométrica o por su alta dimensión?

En esta tesis estudiamos un método de aproximación *sin malla*, basado en el adecuado entrenamiento de redes neuronales. Este enfoque es superado, en cuanto a velocidad computacional, por los métodos convencionales de Elementos Finitos para problemas en dimensión 1, 2 o 3. Sin embargo, permite el abordaje de problemas en dimensión alta (4 o más), para los cuales el mallado resulta técnicamente muy complicado y computacionalmente prohibitivo.

Para abordar esta problemática, estudiamos el método propuesto en [2]. La idea central es la siguiente: consideramos un problema elíptico de la forma

$$\begin{cases} -\mathrm{div}(A\nabla u) + bu &= f & \text{en } \Omega \\ u &= g_D & \text{en } \Gamma \\ \nabla u \cdot \mathbf{n} &= g_N & \text{en } \partial\Omega \setminus \Gamma \end{cases},$$

para $\Omega \subset \mathbb{R}^n$ un dominio acotado, $\Gamma \subset \partial\Omega$, η la normal unitaria a $\partial\Omega$ y f, g_D, g_N dados. En primer lugar, realizamos la formulación mixta del problema como un sistema de ecuaciones de primer orden. Luego, planteamos un problema de mínimos cuadrados cuya solución coincide con la solución de la ecuación. Para resolver este problema de optimización, modelamos la solución mediante una red neuronal y la entrenamos usando algoritmos de descenso estándar en el contexto de *Machine Learning*.

Lo más interesante del enfoque propuesto en [2] es que el problema de mínimos cuadrados se formula de manera tal que impone las condiciones de contorno del problema diferencial de manera estricta y no mediante técnicas de penalización que, si bien son utilizadas en otros abordajes similares, conducen a resultados en los que estas condiciones se satisfacen sólo de manera aproximada.

Para imponer las condiciones de contorno, en [2] se valen de funciones distancia regularizada que computan (aproximadamente) la distancia a Γ y a $\partial\Omega \setminus \Gamma$. En el caso de dominios sencillos (esferas, cubos, cilindros, etc.) estas funciones son fáciles de obtener. También resulta sencillo dar funciones distancia para dominios dados por operaciones elementales de conjuntos aplicadas a dominios simples. E.g.: una esfera a la que se le remueve un cubo. Sin embargo, para el caso de dominios más intrincados, la obtención de una fórmula para la función distancia al borde puede no ser posible. Para esta situación, en [19] se propone el entrenamiento previo de redes neuronales auxiliares que aproximen la distancia. Estas redes son luego utilizadas como insumo del algoritmo de aproximación principal.

Aquí nos limitamos al estudio de dominios sencillos para los cuáles la función distancia puede darse explícitamente. Esto nos permite prescindir de redes auxiliares, reduciendo tanto el error acumulado como el costo computacional. De este modo, nuestro enfoque se orienta principalmente a explorar el comportamiento del método en dominios de dimensión elevada, evaluando su viabilidad y rendimiento en ese contexto. Además, contar con una función distancia que se anule en el conjunto correspondiente es lo que nos permite garantizar que

las condiciones de contorno se satisfagan de manera exacta, mientras que el uso de redes neuronales aproximantes introducirá inevitablemente cierto grado de inexactitud.

En el **Capítulo 1** comenzamos introduciendo la idea general de los métodos de descenso para problemas de minimización y desarrollamos brevemente la estructura general de una red neuronal. Dado que los métodos de descenso que utilizaremos presuponen el cálculo de gradientes, explicamos también distintos enfoques posibles para el cálculo de derivadas. En capítulos posteriores, examinamos la idea de convergencia teórica asociada a estos métodos y mostramos que, bajo ciertas hipótesis sobre el dominio y las funciones involucradas, es posible garantizar dicha convergencia.

En el **Capítulo 2** presentamos nociones generales de espacios de Sóbolev. En el **Capítulo 3**, nos basamos en los resultados de [6] para demostrar que, bajo ciertas condiciones, el operador que buscamos minimizar —y cuya minimización nos proporciona la solución aproximada deseada— es **elíptico**[1] respecto de cierta norma.

Finalmente, en el **Capítulo 4** presentamos la demostración formal de la convergencia del método, y en el **Capítulo 5** analizamos los resultados obtenidos a partir de las implementaciones numéricas.

[1] Un operador $\mathcal{L}(p, \boldsymbol{\phi})$ es elíptico bajo la norma $\|\cdot\|$ si existen constantes α y β tales que

$$\alpha\|(p, \boldsymbol{\phi})\| \leq \mathcal{L}(p, \boldsymbol{\phi}) \leq \beta\|(p, \boldsymbol{\phi})\|.$$

2 Métodos de descenso

Como es sabido, un gran problema de la matemática es aproximar funciones. En esta tesis, uno de los temas a tratar es aproximar una función desconocida $u(x)$, de la cual tenemos sólo algunos datos. Estos datos pueden ser, por ejemplo, mediciones en algunos puntos, como en los problemas de interpolación (mediciones exactas) o de ajustes clásicos por cuadrados mínimos (mediciones con error), o propiedades de la función, como pueden ser ecuaciones que la función o sus derivadas deben satisfacer. Un enfoque posible para realizar la aproximación consiste en asumir que la función a aproximar responde a cierto modelo $f(x;p)$ que depende de ciertos parámetros p y que para cada valor de p representa una función de x. En este contexto, el objetivo es hallar valores p_0 de p de modo que $f(x;p_0)$ aproxime lo mejor posible a $u(x)$. Para esto es necesario medir de algún modo el error de aproximación, lo que nos induce a introducir una función de pérdida $P(p)$ que realiza la comparación y que buscaremos minimizar. Dicho esto, surge una pregunta obvia que es cómo aproximar. Para eso introduciremos los bien conocidos métodos de descenso.

2.1. Método del gradiente

Los métodos de descenso responden al siguiente modelo general:

- Tomar un punto inicial x_0.
- Seleccionar una dirección de descenso d.
- Considerar $\phi(t) = f(x_0 + td)$ y minimizar ϕ con respecto a t, obteniendo t_0.
- Actualizar el punto: $x_1 = x_0 + t_0 d$.
- Iterar el proceso hasta convergencia.

Hay dos puntos fundamentales que es necesario considerar para construir un método de descenso según este esquema general:

1. ¿Cómo elegir la dirección d?
2. ¿Cómo minimizar $\phi(t)$?

Para responder la primera pregunta, una solución sencilla es utilizar el gradiente de la función.

Para ello, queremos elegir una dirección d con $\|d\| = 1$, tal que $\frac{\partial f}{\partial d}(x_0) < 0$ y sea lo más negativa posible, asumiendo f derivable con derivada continua, tenemos:

$$\frac{\partial f}{\partial d}(x_0) = \nabla f(x_0) \cdot d = \|d\|\|\nabla f(x_0)\| \cos(\theta),$$

donde θ es el ángulo entre $\nabla f(x_0)$ y d. Pero esto es negativo cuando $\pi/2 < \theta < 3\pi/2$ y tiene máximo módulo cuando $\theta = \pi$. Es decir, el gradiente de una función apunta en la dirección de máximo ascenso, mientras que el vector opuesto al gradiente apunta en la dirección de máximo descenso. Basándonos en esta idea, podemos tomar $d = -\nabla f(x_0)$.

Sin embargo, esto plantea una nueva cuestión: ¿cómo calculamos el gradiente? Hay varias posibilidades:

a. Calcularlo analíticamente e ingresarlo manualmente en el algoritmo.

b. Utilizar algún esquema de diferencias finitas

c. Auto-diferenciación.

Si bien es un recurso útil en algunos casos, el cálculo manual de las derivadas no es práctico al trabajar con redes neuronales.
En cuanto al uso de diferencias finitas, tienen el inconveniente de que es necesario regular la longitud del paso para evitar problemas numéricos. Como ejemplo, consideremos diferencias *forward*:

$$f'(x) = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}$$

A medida que h se achica tendremos: una resta de valores muy cercanos y un cociente por un valor muy pequeño. Ambas cosas son problemáticas. Por ejemplo, si tomamos como $f(x) = \log(x)$, podemos observar cómo aumenta el error de la derivada a medida que disminuye el h.

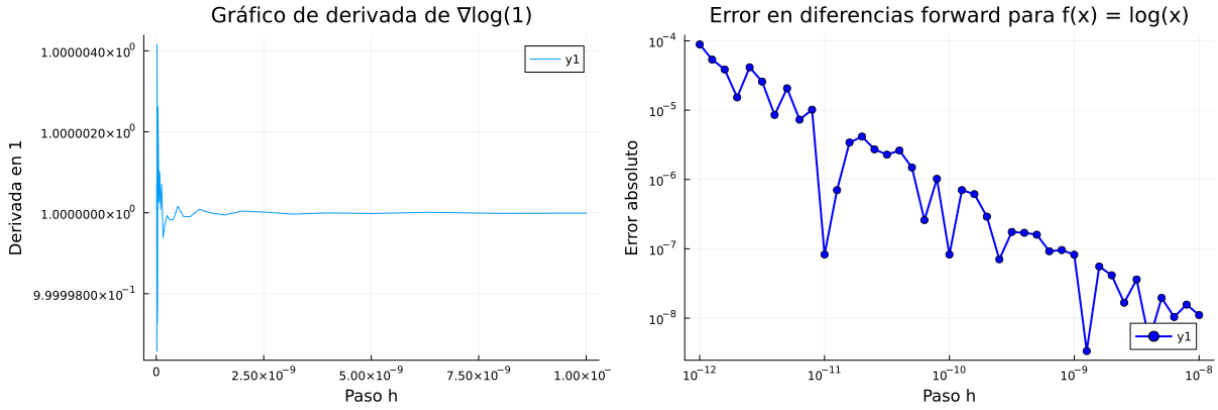


Figura 1: Gráfico del error de la derivada de la función logaritmo evaluada en 1 y también el error absoluto de dicha aproximación.

Para evitar este efecto suele tomarse $h \sim \sqrt{\varepsilon}$, donde ε es el épsilon de la máquina. Sin embargo, es importante notar que para reducir los tiempos de ejecución las librerías de Machine Learning suelen utilizar como estándar números de tipo `Float32`, en cuyo caso $\sqrt{\varepsilon} \sim 10^{-4}$ lo que puede redundar en errores no tan pequeños.

La auto-diferenciación es una técnica que permite calcular derivadas de una función de manera automática y precisa, utilizando la regla de la cadena. Esta técnica es especialmente útil ya que evita los errores numéricos asociados a los métodos de diferencias finitas y no requiere la derivación simbólica manual de las funciones. Dado que en esta tesis usaremos auto-diferenciación para el cálculo de todas las derivadas numéricas, dedicaremos parte de la siguiente sección a desarrollarlas con cierto detalle.

Descenso por el gradiente

El método de descenso por el gradiente o de máximo descenso consiste simplemente en tomar como dirección de descenso $\mathbf{d} = -\nabla f$, siendo $f : \mathbb{R}^n \to \mathbb{R}$ la función a minimizar y ∇f su gradiente (entendido como un vector columna). De este modo, el método de descenso por el gradiente se define mediante el algoritmo iterativo

$$x_{k+1} = x_k - \alpha_k \nabla f(x),$$

donde α_k es un escalar no negativo. En principio, lo ideal sería tomar α_k que minimice $f(x_k - \alpha_k \nabla f(x))$, i.e.: elegir x_{k+1} el punto que minimiza f a lo largo de la dirección del gradiente negativo $-\nabla f$. Sin embargo, esto no siempre resulta práctico o fácilmente calculable. En la literatura sobre optimización de funciones no lineales se considera una variedad de posibles criterios para la elección del paso α_k: desde algoritmos iterativos para aproximar el α_k minimizante, hasta criterios heurísticos (Armijo, Goldstein, Wolfe, etc.) cuyo único objetivo es dar un paso *razonable*, es decir: ni muy chico ni muy grande, para evitar tanto que el algoritmo se estanque como que el algoritmo se pase del mínimo y termine oscilando en torno a él. Sin embargo, en el universo de *Machine Learning* lo habitual es elegir un paso α_k fijo (denominado *learning rate*) y, en todo caso, achicar cada cierta cantidad de iteraciones.

2.2. Diferenciación automática

Diferenciación hacia adelante

Ahora, vamos a presentar una técnica llamada Diferenciación hacia adelante (forward differentiation) que viene de la idea de mejorar el método de diferencias finitas. Como bien es sabido, en [18], uno de los problemas de utilizar diferencias finitas es el hecho de que, al hacer comparaciones entre números muy pequeños, nos arriesgamos a cometer un error de máquina (o sea que el número sea tan pequeño que la máquina lo interprete como 0). Para eso, vamos a tomar la idea que se obtiene al hacer diferencias finitas con números complejos. Tomemos f una función real que podemos extender a los complejos. Si tomamos $x \in \mathbb{R}$, haciendo Taylor:

$$f(x + ih) = f(x) + f'(x)ih - \frac{1}{2}f''(x)h^2 + O(h^3).$$

reordenando los términos:

$$if'(x) = \frac{f(x + ih) - f(x)}{h} + \frac{1}{2}f''(x)h + O(h^2).$$

Como x es real y f cuando es evaluada en reales es real, if' es puramente imaginaria, luego tomando la parte imaginaria de ambos lados:

$$f'(x) = \frac{Im(f(x + ih))}{h} + O(h^2).$$

Tomando h suficientemente chico, esto es la versión de diferencias finitas en números complejos. Ahora bien, ¿cuál es la ventaja o idea que podemos tomar de esto? Recordemos que x es puramente real, por lo que $x + ih$ es un número complejo donde h **nunca interactúa**

directamente con x. O sea, podríamos considerar ambas cosas por separado. De esta forma, no vamos a tener cancelación numérica debido al tamaño de h, de modo que esta fórmula acarrea menos inconvenientes numéricos que las fórmulas clásicas de diferencias finitas, al costo de obligarnos a computar evaluaciones en los complejos. Sin embargo, podemos sofisticar un poco más el argumento, para eliminar h del cómputo. Para ello, comenzamos introduciendo los **números duales** como en [17] .

Números Duales

Un número dual es un número multidimensional, donde la sensibilidad de la función se propaga por la porción dual. Si pensamos ϵ como una cantidad pequeña, tenemos

$$f(a+\epsilon) = f(a) + f'(a)\epsilon + o(\epsilon).$$

A una función f la representaremos por su valor $f(a)$ y el de su derivada $f'(a)$ codificados como los coeficientes del polinomio de Taylor de grado 1 en ϵ

$$f \rightsquigarrow f(a) + f'(a)\epsilon.$$

Formalmente podemos definir ϵ tal que $\epsilon^2 = 0$ (de manera similar al modo en que i^2 es definido como -1), lo cual nos permite establecer la igualdad.

$$f(a+\epsilon) = f(a) = \epsilon f'(a),$$

que equivale a descartar los términos de orden mayor o igual que dos en la expansión de Taylor.

Con esto, podemos generar un álgebra sobre los números duales, es decir, si tenemos $f \rightsquigarrow f(a) + f'(a)\epsilon$ y $g \rightsquigarrow g(a) + g'(a)\epsilon$, entonces

$$\begin{aligned}(f+g) &= [f(a)+g(a)] + [f'(a)+g'(a)]\epsilon,\\ (f\cdot g) &= [f(a)\cdot g(a)] + [f(a)\cdot g'(a) + g(a)\cdot f'(a)]\epsilon\end{aligned}$$

De esta manera, la evaluación de funciones incluye de por sí la evaluación de las derivadas, y esto puede propagarse a través de cualquier tipo de función, siempre que se tengan implementadas las derivadas de algunas funciones elementales. Lo importante es que esto es fácilmente implementable. A continuación, mostramos una implementación elemental e incompleta en `Julia`:

```
struct Dual{T<:Real}
    val::T
    der::T
end

Base.:+(x::Dual,y::Dual) = Dual(x.val+y.val,x.der+y.der)
Base.:+(x::Dual,y::Real) = Dual(x.val+y,x.der)
Base.:+(x::Real,y::Dual) = y+x
```

```
Base.:-(x::Dual,y::Dual) = Dual(x.val-y.val,x.der-y.der)

Base.:*(x::Dual,y::Dual) = Dual(x.val*y.val,x.der*y.val+x.val*y.der)
Base.:*(x::Dual,y::Real) = Dual(x.val*y,x.der)
Base.:*(x::Real,y::Dual) = y*x
```

Las primeras líneas definen un nuevo tipo de dato, `Dual`, que está compuesto de dos números, `val` (valor) y `der` (la derivada), ambos de tipo `T`, que a su vez debe ser un subtipo de `Real`. De este modo:

```
z = Dual(1.0,2.5)
```

define el número dual $z = 1 + 2{,}5\epsilon$, con `T=Float64`. Las siguientes líneas definen las operaciones suma, resta y producto para números duales. De modo que el código:

```
z = Dual(3.0,1.0)
f(x) = 2x+5
f(z)
```

Computa, siguiendo las definiciones dadas para suma y producto: $2(3+\epsilon)+5 = 7+2\epsilon$ y este número dual resultante contiene el valor de f en 3, que es 7 y el valor de $f'(3)$, que es 2. Para que este comportamiento se extienda a otras funciones sólo es necesario implementar adecuadamente otras funciones elementales. Por ejemplo, si agregamos la definición:

```
Base.sin(x::Dual) = Dual(sin(x.val),cos(x.val))
```

Seremos capaces de evaluar la función $g(x) = \sin(3\sin(2x+7))$ sobre un número dual de la forma $z = a + \epsilon$ y de ese modo obtener simultáneamente $f(a)$ y $f'(a)$.

Contando con una implementación de números duales, podemos definir la función derivada de manera muy sencilla y transparente para el usuario, que no se verá obligado a manipular números duales:

```
derivada(f,x) = f(Dual(x,1.)).der
```

Obviamente, este mismo concepto se puede extender en varias dimensiones considerando a ϵ como una perturbación en diferentes direcciones, donde cumple $\epsilon_i^2 = \epsilon_i\epsilon_j = 0$. Entonces tendríamos,

$$f(a+\epsilon) = f(a) + \nabla f(a)\cdot\epsilon$$

donde $a \in \mathbb{R}^n$ y $\nabla f(a)\cdot\epsilon$ son las derivadas direccionales de f en la dirección ϵ.

Diferenciación hacia atras

Los números duales dan una forma de cálculo de derivadas *hacia adelante*. Es decir: las derivadas se computan en el mismo proceso de evaluación. Ahora vamos a hacer uso de otra técnica llamada diferenciación hacia atrás (reverse method). Para entender esta, primero volvamos a cómo funcionaba diferenciar hacia adelante con los números duales, por ejemplo:

$$d = d_0 + v_1\epsilon_1 +, \cdots, +v_m\epsilon_m$$

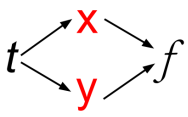
Y tenemos (haciendo el Taylor correspondiente),

$$f(d) \approx f(d_0) + f'(d_0)v_1\epsilon_1 +, \cdots, +f'(d_0)v_m\epsilon_m$$

Donde $f'(d_0)v_i$ es la derivada direccional de f en la dirección v_i. Para calcular el gradiente en la entrada deseada, simplemente tendríamos que poner $v_i = e_i$.
Ahora supongamos que tenemos una función $f(x(t), y(t))$ y queremos calcular la derivada con respecto al tiempo. Para eso, tenemos que usar regla de la cadena:

$$\frac{df}{dt} = \frac{df}{dx}\frac{\partial x}{\partial t} + \frac{df}{dy}\frac{\partial y}{\partial t}.$$

Como bien se muestra en el siguiente gráfico, para calcular la derivada total de f respecto de t, es necesario pasar por las derivadas parciales respecto de x e y.



La gracia del método es suponer que ya tenemos el valor de $f(t)$ a priori, por lo tanto, ya tenemos calculados los valores de x, y. Entonces, dada la función f, podemos calcular $\frac{df}{dx}$ y $\frac{df}{dy}$ para luego calcular $\frac{dx}{dt}$ y $\frac{dy}{dt}$.
Veamos un ejemplo simple de regresión.

Ejemplo 2.1 (Modelo logístico univariado)**.** El modelo logístico univariado se emplea en problemas de clasificación binaria, donde la variable de salida $t \in \{0, 1\}$ representa una etiqueta de clase. En este modelo, se estima la probabilidad de que una observación pertenezca a la clase $t = 1$ mediante la función sigmoide:

$$\sigma(wx + b) = \frac{1}{1 + e^{-(wx+b)}}, \tag{1}$$

donde w y b son los parámetros del modelo y $x \in \mathbb{R}$ es la variable de entrada.

Para entrenar el modelo, se minimiza la función de pérdida basada en la entropía cruzada:

$$L = -\sum_{i=1}^{N} \left[t_i \log(\sigma_i) + (1 - t_i) \log(1 - \sigma_i) \right], \tag{2}$$

donde $\sigma_i = \sigma(wx_i+b)$ representa la probabilidad estimada para la observación i. Deseamos encontrar los parámetros w y b que minimicen la función de pérdida; para eso, es necesario el cálculo de los gradientes. Estos mismos se obtienen como:

$$\frac{\partial L}{\partial w} = \sum_{i=1}^{N} (\sigma_i - t_i)x_i, \quad \frac{\partial L}{\partial b} = \sum_{i=1}^{N} (\sigma_i - t_i). \tag{3}$$

Además, si calculamos la derivada de L con respecto a σ, obtenemos:

$$\frac{\partial L}{\partial \sigma} = \sigma - t \tag{4}$$

Esto último nos está hablando de que la naturaleza del error en este ejemplo viene en medir la diferencia entre la salida de la red y el clasificador. De esta forma, una opción natural, similar a lo que hicimos antes, es considerar a la función de pérdida como:

$$L = \frac{1}{2}(\sigma(wx + b) - t)^2$$

Ahora bien, si nosotros quisiéramos calcular las derivadas a mano, se podría, pero es un cálculo largo y poco eficiente. Una alternativa es utilizar lo que aprendimos antes. Consideremos ahora el siguiente arreglo:

$$\begin{aligned} z &= wx + b \\ y &= \sigma(z) \\ L &= \frac{1}{2}(y - t)^2 \end{aligned}$$

Ahora podemos realizar un diagrama del cálculo del gradiente y la pérdida y de esta forma entender un poco mejor la naturaleza del método que queremos implementar.

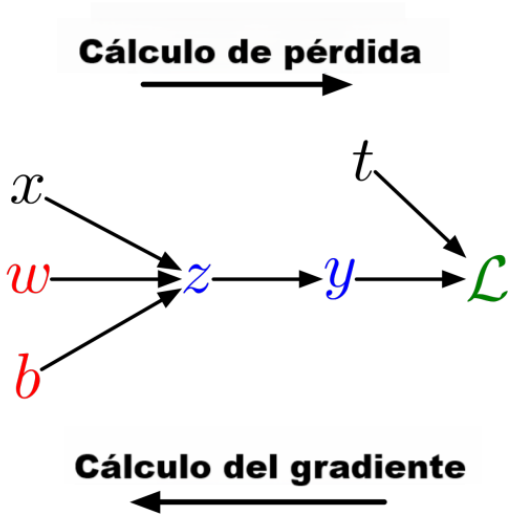


Figura 2: Los nodos representan todas las posibles entradas y cantidades calculadas y las flechas representan cuales nodos se calculan directamente como funciones de otros nodos.

Si ahora calculamos cada derivada aparte, obtenemos (usando como notación $\frac{\partial L}{\partial y} = \overline{y}$):

$$\begin{aligned}
\frac{\partial L}{\partial y} &= \overline{y} = y - t \\
\frac{\partial L}{\partial z} &= \overline{z} = \overline{y}\sigma'(z) = \frac{\partial L}{\partial y}\sigma'(z) \\
\frac{\partial L}{\partial w} &= \overline{w} = \overline{z}x = \frac{\partial L}{\partial z}x \\
\frac{\partial L}{\partial b} &= \overline{b} = \overline{z} = \frac{\partial L}{\partial z}
\end{aligned}$$

En conclusión, haciendo alusión al gráfico anterior 2, la regla de backpropagation consiste en que, para cada nodo, el gradiente total se obtiene sumando los términos correspondientes a cada flecha que sale, donde cada flecha transporta el gradiente del nodo final multiplicado por la derivada parcial local respecto de la variable de ese nodo. Este procedimiento se aplica de manera recursiva hacia atrás en el grafo, de forma que las derivadas totales se construyen como combinaciones lineales simples de derivadas locales. En nuestro caso, obtenemos, usando lo anterior que:

$$\frac{\partial L}{\partial w} = (\sigma(z) - t)x, \quad \frac{\partial L}{\partial b} = \sigma(z) - t.$$

2.3. Redes neuronales

En esta tesis nos proponemos aproximar la solución de una ecuación diferencial mediante redes neuronales. En esta sección damos algunos conceptos básicos al respecto basándonos en [12, 16].

Una red neuronal es una función $f : \mathbb{R}^n \to \mathbb{R}^m$, con una estructura particular. Esencialmente se trata de construir f como composición de transformaciones afines y funciones no lineales, usualmente llamadas *funciones de activación*.

Comencemos por el caso más simple, dado por lo que llamamos una *capa densa* de $\mathbb{R}^n \to \mathbb{R}$. Dado un vector $w \in \mathbb{R}^n$, $b \in \mathbb{R}$ y una función de activación $\phi : \mathbb{R} \to \mathbb{R}$. Definimos la capa densa como:

$$f(x) = \phi(w^t x + b).$$

De manera similar, una capa densa de $f : \mathbb{R}^n \to \mathbb{R}^m$ está determinada por una matriz $W \in \mathbb{R}^{m \times n}$, un vector $b \in \mathbb{R}^m$, y una función de activación $\phi : \mathbb{R} \to \mathbb{R}$ de manera que:

$$f(x) = \phi.(Wx + b),$$

donde la notación $\phi.()$ indica que ϕ se aplica coordenada a coordenada. Notar que la misma función de activación se aplica a todas las componentes.

Podemos construir redes neuronales *profundas* componiendo varias capas densas. De este modo, tendremos:

$$f(x) = \phi^k.(W^k(\phi^{k-1}.(W^{k-1}(\dots \phi^1.(W^1 x + b^1)\dots) + b^{k-1}) + b^k). \tag{5}$$

Donde W^ℓ, b^ℓ, $\ell = 1, \dots, k$ son matrices y vectores de tamaños adecuados y ϕ^ℓ, $\ell = 1, \dots, k$ son funciones de activación.

Una función de este tipo puede representarse gráficamente como una red de k capas con n nodos iniciales y m nodos finales:

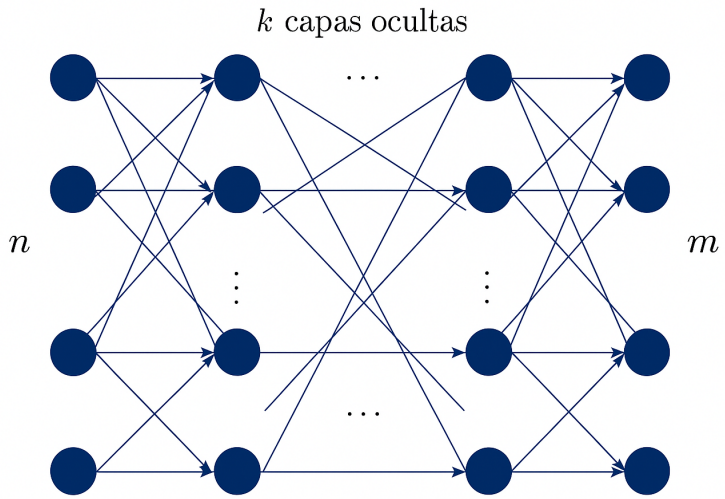


Figura 3: Representación esquemática de una red neuronal profunda como composición de funciones. Las flechas indican el flujo de datos a través de las capas. Los nodos de la izquierda corresponden a las variables de entrada (n) y los de la derecha a las variables de salida (m). En el medio hay $k-2$ capas intermedias.

Notar que tanto la cantidad de capas intermedias (*profundidad* de la red) como la cantidad de nodos de esas capas (*ancho* de la capa) son arbitrarias. Las capas intermedias se denominan *capas ocultas*, dado que son un detalle de implementación interno de la red: la enriquecen y complejizan, pero no modifican su naturaleza general como función de $\mathbb{R}^n$ en $\mathbb{R}^m$.

En el ámbito del Machine Learning, las redes neuronales se utilizan como modelos de funciones para ajustar datos. La idea es que una red neuronal puede ser una función no lineal muy complicada, con versatilidad para ajustar datos variados incluso en dimensión alta, pero tiene una estructura sencilla que facilita la aplicación de métodos de optimización para realizar el ajuste.

A modo de ejemplo, analicemos un problema general. Siguiendo este marco, sería el siguiente. Dados datos $x_1, \dots, x_r \in \mathbb{R}^n$, $y_1, \dots, y_r \in \mathbb{R}^m$, buscamos una función f de manera tal que $f(x_i) \sim y_i$. Para fijar ideas, podemos suponer que medimos el error mediante mínimos cuadrados (aunque, en general, uno puede considerar funciones de pérdida más generales y luego veremos un ejemplo en el que hacemos uso de esta generalidad). Es decir, buscamos f que minimice:

$$\sum_{i=1}^{r} \|f(x_i) - y_i\|^2. \tag{6}$$

Para lograr esto, tomamos f una red neuronal según el modelo (5) y optimizamos sobre los parámetros W^ℓ, b^ℓ, $\ell = 1, \dots, k$ para minimizar (6).

El ejemplo más básico corresponde a un ajuste lineal clásico de datos $x_i \in \mathbb{R}^n$, $y_i \in \mathbb{R}$, tomando una red neuronal de una única capa y con función de activación dada por la identidad: $id(x) = x$. Esto equivale a buscar valores de los parámetros $w \in \mathbb{R}^n$, $b \in \mathbb{R}$ que minimicen

$$\sum_{i=1}^{n} (w^t x_i + b - y_i)^2.$$

En este caso, el problema de optimización resulta lineal y puede resolverse mediante técnicas usuales de álgebra lineal.

Para el modelado de situaciones más complejas necesitamos usar funciones de activación no lineales. Si bien en principio podría usarse cualquier función de activación, hay algunas funciones habituales que se eligen por sus propiedades o por su efectividad para distinto tipo de problemas, verificadas en la práctica. Algunas de las más comunes son:

$$\begin{aligned} \text{Sigmoide: } & \sigma(x) = \tfrac{1}{1+e^{-x}}, \\ \text{ReLU: } & \phi(x) = \text{máx}\{0, x\}, \\ \text{Softplus: } & \phi(x) = \ln(1 + e^x). \end{aligned}$$

En esta tesis, aunque en general se pueden demostrar los resultados con más generalidad, vamos a usar principalmente funciones sigmoides.

Para el proceso de minimización podemos utilizar el método del gradiente. Más adelante veremos que en la práctica se utilizan distintas variantes de este método. Volvamos a la ecuación (5) y notemos:

$$\begin{aligned} a^0 &= x \in \mathbb{R}^n, \\ z^\ell &= W^\ell a^{\ell-1} + b^\ell \in \mathbb{R}^{n_\ell}, \\ a^\ell &= \phi^\ell(z^\ell) \in \mathbb{R}^{n_l}. \end{aligned}$$

Entonces

$$f(w, a') = f(w, x) = \underbrace{\phi^k(\cdots \underbrace{\phi^2(W^2(\underbrace{\phi^1(W^1 a^0 + b^1)}_{a^1}) + b^2)}_{a^2} + \cdots + b^k)}_{a^k}.$$

Ahora bien, con esta definición, si derivamos la función en base a sus parámetros hasta orden ℓ, obtenemos:

$$\begin{cases} \nabla_{W^\ell} f &= \nabla_{z^k} a^k \cdot \nabla_{a^{k-1}} z^k \cdots \nabla_{z^\ell} a^\ell \cdot \nabla_{W^\ell} z^\ell. \\ \nabla_{b^\ell} f &= \nabla_{z^k} a^k \cdot \nabla_{a^{k-1}} z^k \cdots \nabla_{z^\ell} a^\ell \cdot \nabla_{b^\ell} z^\ell. \end{cases}$$

Notar que una vez alcanzado el orden ℓ, como z^ℓ depende de $\ell - 1$, no hay que seguir derivando.

Ahora que sabemos cómo funciona una red, podemos realizar la siguiente observación: en el caso de la diferenciación hacia adelante también es posible calcular las derivadas de la

función, pero esto puede implicar un mayor costo computacional dependiendo de la estructura del problema. ¿Cómo se traduce esto en el contexto de redes neuronales? Consideramos la expresión (2.2), pero ahora queremos derivar no con respecto a los datos, sino con respecto a los parámetros p. Para ello, basta con tomar:

$$x = x_0 + 0\epsilon_1 + \cdots + 0\epsilon_k,$$
$$P = p_0 + p_1\epsilon_1 + \cdots + p_k\epsilon_k,$$

donde k es la cantidad de parámetros. Siguiendo el mismo procedimiento que antes, se obtiene:

$$f(x, P) = f(x, p) + \frac{df}{dp_1}\epsilon_1 + \cdots + \frac{df}{dp_k}\epsilon_k.$$

Luego realizamos todos los cálculos como hemos visto en esa sección. Por otro lado, también se puede utilizar diferenciación hacia atrás en este contexto (conocida comúnmente como *backpropagation*). Utilizando las mismas ideas del ejemplo (2.1), una vez computada toda la red, es posible calcular todas las derivadas de atrás hacia adelante de manera eficiente.

Habiendo aclarado que ambos métodos pueden utilizarse en redes neuronales, podemos entender mejor cuándo conviene aplicar diferenciación hacia adelante y cuándo diferenciación hacia atrás. La elección depende de la estructura de la función: la diferenciación hacia adelante calcula las derivadas con respecto a cada variable de entrada de forma individual, mientras que la diferenciación hacia atrás permite obtener todas las derivadas con respecto a los parámetros cuando la función tiene una única salida.

Visualmente (por ejemplo, en la figura (3)), esto puede entenderse como que en diferenciación hacia adelante (o *forward differentation*) calculamos las derivadas a lo largo de cada columna (una por cada parámetro), mientras que en diferenciación hacia atrás las derivadas se propagan fila por fila, desde la salida hacia los parámetros.

En nuestro caso, la función f depende de muchos parámetros y produce una única salida (el valor de la función de pérdida). Por lo tanto, la diferenciación hacia atrás es mucho más eficiente, ya que permite calcular todas las derivadas en una sola pasada, a diferencia del enfoque hacia adelante, que requeriría una pasada por cada parámetro.

Para aplicar este método de forma eficiente, es fundamental que las funciones de activación tengan derivadas simples, de modo que puedan ser almacenadas previamente y el cálculo del gradiente se reduzca a operaciones básicas, como ya se mostró en los ejemplos anteriores.

Ahora, con estos métodos explicados, veamos un ejemplo más de cómo funciona una red y así poder adentrarnos más profundamente en ella.

Ejemplo 2.2. Consideremos una red neuronal con una sola neurona y una función de activación σ. Tomamos $w \in \mathbb{R}^n$, $b \in \mathbb{R}$, y $\hat{y} = \sigma(w^T x + b)$. Así, podemos definir la función de pérdida como: $G(y, \sigma(w^T x + b))$. Entonces, los gradientes con respecto a w y b son:

$$\nabla_w G = \underbrace{G'(y, \sigma(w^T x + b))}_{\in \mathbb{R}} \cdot \underbrace{\sigma'(w^T x + b)}_{\in \mathbb{R}} \cdot \underbrace{x}_{\in \mathbb{R}^n}, \quad \nabla_b G = \underbrace{G'(y, \sigma(w^T x + b))}_{\in \mathbb{R}} \cdot \underbrace{\sigma'(w^T x + b)}_{\in \mathbb{R}}.$$

Además:

$$G' = \frac{\partial G}{\partial \hat{y}} = -(y - \hat{y}) = \hat{y} - y, \; \sigma'(x) = ((1 + e^{-x})^{-1})' = \cdots = \sigma(x)(1 - \sigma(x)).$$

Luego, el gradiente completo se expresa como: $\nabla G = \begin{pmatrix} \nabla_w G \\ \nabla_b G \end{pmatrix} = (\sigma(w^T x+b)-y)\cdot\sigma(w^T x+b)(1-\sigma(w^T x+b))\cdot\begin{pmatrix} x \\ 1 \end{pmatrix}$. Finalmente, podemos simplificar: $\nabla G = (\hat{y}-y)\hat{y}(1-\hat{y})\begin{pmatrix} x \\ 1 \end{pmatrix}$. Este gradiente se puede utilizar en el método del gradiente para actualizar los parámetros w y b.

El uso del método del gradiente en problemas de este tipo presenta dos dificultades principales. En primer lugar, el costo computacional del cálculo del funcional y de su gradiente depende de la cantidad de datos disponibles, que en la práctica puede ser muy grande. En segundo lugar, se trata de un método de descenso, por lo que si la función de pérdida es no lineal y presenta múltiples mínimos locales, el algoritmo puede converger a un mínimo local que depende fuertemente del punto inicial. Para atender ambos problemas, lo habitual no es utilizar el método del gradiente puro, sino alguna de sus variantes.

2.4. Variantes al método del gradiente

Ahora, con lo visto previamente, tenemos que trabajar con este problema. Tanto la función de pérdida como su gradiente van a depender de la cantidad de puntos L, que pueden ser muy grandes. Por ese motivo, un primer acercamiento es considerar el método de **gradiente estocástico**. En este se toma la estrategia de que en cada iteración tomo un subconjunto $S \subset \{1, \cdots, L\}$, con $\#S = s \ll L$ y considero ahora cómo en (6) de la siguiente manera.

$$J(w) = \sum_{l \in S} ||f(x_l) - y_l||^2.$$

Con w, x_k, y_k los parámetros y los datos con su esperada respuesta, respectivamente.
Algo a considerar es el hecho de que, al realizar lo anterior, no obtenemos el gradiente real, por lo que este método no es un método de descenso y, por lo tanto, los teoremas análogos no son válidos. Por otro lado, hay una gran cantidad de resultados que respaldan este tipo de criterios. Como en [3], en el cual podemos ver demostraciones en casos generales sobre convergencia (casi segura) sobre las funciones costo o también en [7] donde se presentan resultados de garantías de regret y mejoras empíricas que justifican la convergencia de métodos adaptativos. Ahora bien, con las ideas previamente expuestas, podemos identificar un problema común al aproximar funciones usando datos: si los datos contienen errores, no quisiéramos sobreajustar nuestra aproximación a esos datos ni aprender características demasiado específicas que no generalicen bien. Para abordar este problema, existen varias soluciones, como vemos en [14], entre las cuales se incluyen algoritmos como Nesterov, Adagrad, RMSprop, Adadelta y ADAM, entre otros.

Nosotros profundizaremos en el algoritmo de ADAM, ya que será el que utilizaremos a lo largo de nuestros ejemplos. A modo de introducción, comenzaremos explicando algunos métodos que nos llevarán de manera intuitiva hacia el algoritmo que utilizaremos en cuestión.

Método de Momentos

En este enfoque, agregamos un concepto de inercia, es decir, conservamos parte del impulso del paso anterior. De esta manera, el método de gradiente se modifica para incluir un "momento"que suaviza las actualizaciones.

Primero, recordemos cómo se ve el método de gradiente simple:

$$\Delta w_t = \eta \nabla_w J(w),$$
$$w_{t+1} = w_t - \Delta w_t.$$

Donde η es la tasa de aprendizaje (learning rate) y $\nabla_w J(w)$ es el gradiente del funcional de costo J con respecto a w.

Ahora, con el método de momentos, introducimos una dependencia en la actualización anterior:

$$\Delta w_t = \rho \Delta w_{t-1} + \eta \nabla_w J(w),$$
$$w_{t+1} = w_t - \Delta w_t.$$

Comúnmente ρ se toma un valor cercano a 0.9. Este término de momento permite que la actualización de los pesos tenga en cuenta no solo la dirección actual del gradiente, sino también la dirección del gradiente en pasos anteriores, ayudando a estabilizar el proceso de optimización.

A pesar de que el método de momentos mejora la estabilidad de la optimización al suavizar las actualizaciones de los parámetros, presenta algunas limitaciones. No se adapta bien a gradientes que varían en magnitud entre diferentes direcciones, lo que puede provocar actualizaciones ineficientes. Además, depende fuertemente de una tasa ρ que debe ajustarse con precisión, y no responde de manera óptima a gradientes variables. Estas desventajas motivan el uso de métodos más avanzados, como Nesterov, que ajustan adaptativamente la tasa de aprendizaje.

Método de Nesterov

El método de Nesterov es una extensión del método de momentos, con la diferencia clave de que utiliza una **aproximación** de la posición futura para calcular el gradiente. En lugar de calcular el gradiente en la posición actual w_t, lo hacemos en $w_t - \rho \Delta w_{t-1}$, que es una estimación de la próxima posición w_{t+1}. Esto nos permite corregir el curso de la actualización anticipadamente y mejorar la eficiencia del método. El algoritmo se define de la siguiente manera:

$$\Delta w_t = \rho \Delta w_{t-1} + \eta \nabla_w J(w_t - \rho \Delta w_{t-1}),$$
$$w_{t+1} = w_t - \Delta w_t.$$

Este enfoque permite ajustar de manera más precisa la dirección de las actualizaciones al tomar en cuenta el cambio anticipado en w, reduciendo el riesgo de overshooting en la optimización.

Ahora, nos vamos a meter un poco con los llamados **métodos adaptativos** que básicamente su novedad es adaptar la tasa de aprendizaje en cada coordenada, una ventaja que no tuvimos anteriormente:

ADAGRAD

Primero recordemos el método de gradiente clásico:

$$g_{t,i} = \frac{\partial J}{\partial w_i}(w),$$
$$w_i = w_i - \eta g_{t,i}.$$

Aquí, $g_{t,i}$ es la derivada en la coordenada i del funcional J en el tiempo t.

En el método ADAGRAD [7], la actualización se ajusta acumulando el cuadrado de los gradientes anteriores para cada coordenada. El algoritmo queda de la siguiente forma:

$$G_{t,i} = \sum_{s \leq t} g_{s,i}^2,$$
$$w_i = w_i - \frac{\eta}{\sqrt{G_{t,i} + \epsilon}} g_{t,i}.$$

Donde ϵ es un valor pequeño que se utiliza en la práctica para evitar problemas de singularidad de la matriz.

Este método ajusta automáticamente la tasa de aprendizaje usando la información de los gradientes. Dos ventajas principales son que asigna tasas de aprendizaje más grandes a los parámetros relacionados con características de baja frecuencia y que las actualizaciones en direcciones de alta curvatura tienden a ser más pequeñas que en las de baja curvatura. Por lo tanto, el método es especialmente útil en problemas donde los datos de entrada son dispersos. De todas formas, a medida que avanza el entrenamiento, los acumuladores $G_{t,i}$ crecen, y las tasas de aprendizaje tienden a cero. Esto puede llevar a una desaceleración prematura del aprendizaje.

Una variante de este método es RMSProp, que busca evitar que los valores acumulados de $G_{t,i}$ crezcan demasiado rápido, lo que podría hacer que las actualizaciones sean muy pequeñas.

RMSProp

En este método [20], en lugar de acumular todos los gradientes anteriores de manera indefinida, utilizamos una media exponencialmente ponderada de los cuadrados de los gradientes. Definimos esta media como:

$$E(g^2)_t = \beta E(g^2)_{t-1} + (1 - \beta) g_{t,i}^2.$$

Donde $\beta \in [0, 1)$ es un factor que controla la velocidad de decaimiento de la media, y $g_{t,i}$ es el gradiente en la coordenada i en el tiempo t.

Luego, la actualización de los pesos se realiza de la siguiente manera:

$$w_i = w_i - \frac{\eta}{\sqrt{E(g^2)_t + \epsilon}} g_{t,i}.$$

Aquí, ϵ es un valor pequeño que se añade para evitar divisiones por cero y mantener la estabilidad numérica. De esta forma, el método logra controlar las tasas de aprendizaje en cada coordenada, adaptándolas de manera que no se vean afectadas drásticamente por gradientes grandes o pequeños de manera puntual.

Con todos estos métodos llegamos al que estaremos usando en la tesis.

ADAM (Adaptive Moment Estimation)

El algoritmo Adam es un método de optimización de primer orden que combina las ventajas de dos técnicas populares: el momentum y el escalado adaptativo de los gradientes. Su principal objetivo es mejorar la eficiencia y estabilidad del descenso de gradiente, especialmente en problemas de alta dimensión o con gradientes ruidosos y dispersos.

Adam calcula tasas de aprendizaje adaptativas para cada parámetro del modelo, utilizando promedios móviles del primer momento (la media) y del segundo momento (la varianza no centrada) del gradiente. Esto permite realizar actualizaciones más informadas, controladas y estables.

Ventajas del método Adam:

- Las actualizaciones son invariantes ante reescalamientos del gradiente.
- Los tamaños de paso están acotados, lo que mejora la estabilidad numérica.
- No requiere que la función objetivo sea estacionaria.
- Es robusto frente a gradientes dispersos o ruidosos.
- Realiza un ajuste automático y gradual del tamaño de paso.

Descripción del algoritmo: Sean $J(w)$ el funcional de pérdida que se desea minimizar y w_0 la condición inicial sobre los parámetros del modelo. Además, fijamos los hiperparámetros:

- $\eta > 0$: tasa de aprendizaje.
- $\beta_1 \in [0, 1)$: coeficiente de decaimiento del primer momento.
- $\beta_2 \in [0, 1)$: coeficiente de decaimiento del segundo momento.
- $\epsilon > 0$: pequeño número para evitar divisiones por cero (tip. 10^{-8}).

Inicializamos:

$$m_0 = 0, \quad v_0 = 0,$$

donde m_t y v_t son las estimaciones del primer y segundo momento en la iteración t, respectivamente.

El algoritmo itera de la siguiente forma:

$$
\begin{aligned}
g_t &= \nabla_w J_t(w_{t-1}), \\
m_t &= \beta_1 m_{t-1} + (1-\beta_1) g_t, \quad \text{(media del gradiente)} \\
v_t &= \beta_2 v_{t-1} + (1-\beta_2) g_t^2, \quad \text{(media del cuadrado del gradiente)} \\
\hat{m}_t &= \frac{m_t}{1-\beta_1^t}, \quad \text{(corrección del sesgo de } m_t\text{)} \\
\hat{v}_t &= \frac{v_t}{1-\beta_2^t}, \quad \text{(corrección del sesgo de } v_t\text{)} \\
w_t &= w_{t-1} - \frac{\eta\, \hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon}.
\end{aligned}
$$

Cabe aclarar que en el caso de w_t, la división entre $\hat{m}_t$ y $\sqrt{\hat{v}_t}+\epsilon$ es componente a componente.

Interpretación:

- El vector m_t actúa como un término de *momentum*, acumulando gradientes pasados para suavizar la dirección del descenso.
- El vector v_t estima la varianza del gradiente, lo que permite ajustar individualmente la magnitud del paso para cada componente del parámetro w. Componentes con gradientes grandes reciben pasos más pequeños, y viceversa.
- La corrección del sesgo en $\hat{m}_t$ y $\hat{v}_t$ es esencial, especialmente en las primeras iteraciones, ya que m_0 y v_0 están inicializados en cero, lo que introduce un sesgo hacia valores bajos.

Elección típica de hiperparámetros: En la práctica, se suele usar:

$$
\beta_1 = 0{,}9, \quad \beta_2 = 0{,}999, \quad \epsilon = 10^{-8}, \quad \eta \in [10^{-4}, 10^{-3}].
$$

Comentario final: En esta tesis se utilizará el método Adam como algoritmo de optimización principal. Su elección se justifica tanto por sus propiedades teóricas como por su rendimiento práctico en problemas con muchos parámetros, como los que aparecen en el entrenamiento de redes neuronales. Más adelante, en la sección de experimentación, se mostrará que Adam presenta un desempeño considerablemente superior al de otros métodos como el descenso por gradiente estándar o el método de Momentum en los casos que se estudian, tanto en velocidad de convergencia como en estabilidad numérica.

3 Espacios de Sobolev y redes neuronales

El objetivo de esta sección es demostrar que, efectivamente, es posible encontrar funciones en espacios generados por redes neuronales que aproximen a funciones en espacios de Sobolev. Para ello, comenzaremos por entender con precisión qué es un espacio de Sobolev y cuáles son sus propiedades fundamentales. Además, será necesario formalizar las definiciones correspondientes a los espacios funcionales inducidos por redes neuronales, así como las nociones de convergencia que consideraremos en dicho contexto.

3.1. Formulación débil

Como introducción, veremos de dónde surge la idea de estos espacios (ver más en [5]), a partir de un problema basado en el problema de Poisson en una dimensión.

Problema modelo

Consideremos el siguiente problema modelo:

$$\begin{cases} -u''(x) = f(x), & \text{para } x \in (0,1), \\ u(0) = 0, \\ u'(1) = 0, \end{cases} \tag{7}$$

donde $u(x)$ es la función desconocida y $f(x)$ es una función dada.

Problema variacional

A partir de este problema clásico, denominado también **problema en forma fuerte** o (PC), podemos derivar lo que se conoce como el **problema variacional** o **problema débil** (PV). La transición hacia este nuevo enfoque se logra multiplicando la ecuación diferencial por funciones denominadas *funciones test* (que serán introducidas más adelante) e integrando sobre todo el dominio del problema.

De esta forma, comenzamos a trabajar en un espacio funcional adecuado, lo que nos permite obtener una solución en un sentido más general, relajando las condiciones de diferenciabilidad exigidas a la misma.

Multiplicando por una función cualquiera v tenemos:

$$-u''(x)v(x) = f(x)v(x), \quad \text{con } x \in (0,1).$$

Ahora integramos sobre el dominio de u:

$$\int_0^1 -u''(x)v(x)\,dx = \int_0^1 f(x)v(x)\,dx \quad \overset{\text{partes}}{\longrightarrow} \quad \int_0^1 u'(x)v'(x)\,dx - \underbrace{u'(1)v(1)}_{=0} + u'(0)v(0) = \int_0^1 f(x)v(x)\,dx.$$

Para simplificar la expresión, deseamos que el término $u'(0)v(0)$ se anule. Por lo tanto, pedimos que v satisfaga la condición $v(0) = 0$, es decir, tomamos v en el espacio:

$$\mathbb{V} = \left\{ v \in C[0,1] \ : \ v(0) = 0, \ v \text{ derivable a trozos, y } \int_0^1 (v')^2 \, dx < \infty \right\}.$$

Así, llegamos a la siguiente formulación:

$$\int_0^1 u'(x)v'(x) \, dx = \int_0^1 f(x)v(x) \, dx \quad \forall v \in \mathbb{V}. \tag{8}$$

Encontrar una función u que satisfaga la ecuación (8) es lo que denominamos el **problema variacional** o **problema débil**.

3.2. Espacios de Sobolev

Para formalizar correctamente el espacio en el que vive la solución u, debemos introducir herramientas de integración más generales que la integración de Riemann. En particular, la integral de Lebesgue nos permitirá trabajar con funciones que no necesariamente son continuas, pero sí integrables. Para esto, observamos en [1, 8, 11, 15], definiciones formales a estas ideas, pero de todas formas, intentaremos dar ideas generales con respecto a esto.

A partir de la teoría de Lebesgue, definimos los espacios de funciones integrables en potencia, conocidos como espacios L^p, los cuales serán fundamentales para introducir posteriormente los espacios de Sobolev.

Recordemos que, dado un intervalo $\Omega \subset \mathbb{R}$, el espacio $L^p(\Omega)$ se define como:

$$L^p(\Omega) = \left\{ f : \Omega \to \mathbb{R} \text{ medible} \quad : \ \int_\Omega |f(x)|^p \, dx < \infty \right\}.$$

En particular, el caso $p = 2$ da lugar a un espacio con estructura de espacio de Hilbert, que será el escenario más conveniente para nuestra formulación débil.

Observación 1. *Sea $f : E \to \mathbb{R}$ una función medible. Para cada $\alpha \in \mathbb{R}$, definimos el conjunto:*

$$E_\alpha = \{x \in E : |f(x)| > \alpha\}.$$

Si se cumple que $|E_\alpha| > 0$ para todo $\alpha \in \mathbb{R}$, decimos que el supremo esencial de f es $+\infty$, y lo expresamos de cualquiera de las siguientes maneras:

$$\operatorname{ess\,sup}_E f = +\infty, \quad \|f\|_{L^\infty(E)} = +\infty.$$

En caso contrario, definimos:

$$\|f\|_{L^\infty(E)} = \operatorname{ess\,sup}_E |f| = \text{ínf} \left\{ \alpha \in \mathbb{R} : |E_\alpha| = 0 \right\}.$$

En el caso en que $|E| = 0$, según la definición obtendríamos $\|f\|_{L^\infty(E)} = -\infty$ para cualquier función f definida en E. No obstante, adoptamos la convención de que, cuando $|E| = 0$, entonces $\|f\|_{L^\infty(E)} = 0$.

Además, en estos espacios se cumplen importantes desigualdades funcionales, como la desigualdad de Hölder y la desigualdad de Young, que nos permitirán estimar los términos del problema variacional.

Desigualdad de Minkowski. Para $1 \leq p \leq \infty$ y $f, g \in L^p(\Omega)$, se cumple que:

$$\|f+g\|_{L^p(\Omega)} \leq \|f\|_{L^p(\Omega)} + \|g\|_{L^p(\Omega)}. \tag{9}$$

Desigualdad de Hölder. Para $1 \leq p, q \leq \infty$ tales que $\frac{1}{p} + \frac{1}{q} = 1$, si $f \in L^p(\Omega)$ y $g \in L^q(\Omega)$, entonces $fg \in L^1(\Omega)$ y se cumple que:

$$\|fg\|_{L^1(\Omega)} \leq \|f\|_{L^p(\Omega)} \|g\|_{L^q(\Omega)}. \tag{10}$$

Desigualdad de Cauchy-Schwarz. Como caso particular de la desigualdad de Hölder, cuando $p = q = 2$, si $f, g \in L^2(\Omega)$, entonces $fg \in L^1(\Omega)$ y se tiene que:

$$\int_\Omega |f(x)g(x)|\, dx \leq \|f\|_{L^2(\Omega)} \|g\|_{L^2(\Omega)}. \tag{11}$$

Es importante mencionar que este concepto de integral puede generalizarse al caso en el cual trabajemos sobre un espacio abstracto. En particular, consideremos espacios medibles (X, Σ), donde X es un conjunto y Σ es una σ-álgebra de subconjuntos de X.

Dada una función medible $f : X \to \mathbb{R}^k$ en un espacio de medida (X, Σ, μ), denotaremos su integral en este sentido abstracto como $\int_X f\, d\mu$. Además, su norma en L^p se define como:

$$\|f\|_{p,\mu} = \left(\int_X |f(x)|^p \, d\mu(x) \right)^{\frac{1}{p}}.$$

De esta forma, podemos definir la distancia entre dos funciones f y g en este contexto como $\rho_{p,\mu}(f, g) = \|f - g\|_{p,\mu}$.

A continuación, vamos a precisar algunos resultados clásicos del análisis real, junto con una definición que será útil para futuras demostraciones.

Proposición 3.1. *Sean $1 \leq p_1 \leq p_2 \leq \infty$ y $\mu(\Omega) < \infty$. Entonces, para cada $f \in L^{p_2}$ se tiene que*

$$\|f\|_{L^{p_1}} \leq \mu(\Omega)^{\frac{p_2 - p_1}{p_1 p_2}} \|f\|_{L^{p_2}}.$$

Teorema 3.2 (Teorema de Egorov)**.** *Sea E un conjunto de medida finita. Sean $\{f_k\}_{k=1}^\infty$ funciones medibles tales que convergen en casi todo punto a una función medible y finita f. Entonces, para todo $\delta > 0$, existe un conjunto cerrado $F \subseteq E$ tal que $|E \setminus F| < \delta$ y $\lim_{k\to\infty} f_k = f$ uniformemente en F.*

Definición 3.3 (Propiedad $\mathcal{C}$)**.** Sea Ω un conjunto medible y sea $f : \Omega \to \mathbb{R}$ una función. Decimos que f tiene la *propiedad $\mathcal{C}$* sobre Ω si, para cada $\varepsilon > 0$, existe un conjunto $F \subset \Omega$ cerrado tal que $|\Omega \setminus F| < \varepsilon$ y $f|_F$ es continua. En este caso, también decimos que f es *continua relativa a F*.

Teorema 3.4 (Teorema de Lusin)**.** *Sea Ω un conjunto medible y $f : \Omega \to \mathbb{R}$. Entonces, f es medible si y sólo si tiene la propiedad $\mathcal{C}$.*

Teorema 3.5 (Teorema de Convergencia Dominada)**.** *Sea $\{f_k\}_{k=1}^{\infty}$ una sucesión de funciones medibles no negativas definidas en un conjunto Ω que convergen casi en todo punto a una función medible f. Supongamos que existe una función $\Phi : \Omega \to \mathbb{R}$ definida en Ω tal que $\int_\Omega \Phi\, dx < \infty$ y $0 \leq f_k(x) \leq \Phi(x)$ casi en todo $x \in \Omega$, para todo $k \geq 1$. Entonces,*

$$\lim_{k\to\infty} \int_\Omega f_k\, dx = \int_\Omega f\, dx.$$

La función Φ suele llamarse función mayorante, lo que justifica que este resultado también sea conocido como el Teorema de Convergencia Mayorada. La idea intuitiva es que, si todo el "movimiento"de la masa de las f_k ocurre "dentro"de la región dada por la función Φ, no hay posibilidad de que algo escape, como ocurre en ciertos ejemplos contrarios a la convergencia.

Teorema 3.6. *Dada una función f con dominio $A \subset \mathbb{R}^n$, existe una sucesión $\{s_n\}$ de funciones simples que convergen puntualmente a f en A. Si f es acotada, $\{s_n\}$, pueden ser seleccionadas de forma tal que la convergencia sea uniforme. Si f es medible, cada s_n puede ser elegido medible. Si f es no negativa, la sucesión puede ser elegida monótona creciente en cada punto.*

Con estos resultados podemos demostrar nuestro primer teorema de densidad.

Teorema 3.7. *El conjunto de las funciones continuas $C(\Omega)$ es denso en $L^p(\Omega)$ si $1 \leq p < \infty$.*

Demostración. Tomemos $u \in L^p(\Omega)$ y sea $\epsilon > 0$. Queremos ver que existe una función $\phi \in C(\Omega)$ tal que $\|u - \phi\| < \epsilon$. Notar que podemos tomar $u = u_1 - u_2$ con u_1, u_2 funciones reales no negativas, en cuyo caso bastaría con encontrar funciones aproximantes para u_1 y u_2. De esta forma podemos asumir que u es real no negativa y aplicar el teorema anterior según el cual existe una sucesión monótona creciente $\{s_n\}$ de funciones simples no negativas que converge crecientemente a u en Ω. Como $0 \leq s_n(x) \leq u(x)$ tenemos $s_n \in L^p(\Omega)$. Como $(u(x) - s_n(x))^p \leq (u(x))^p$, tenemos que $s_n \to u$ en $L^p(\Omega)$, por el Teorema 3.5 (Convergencia dominada). Podemos tomar un $s \in \{s_n\}$ tal que $\|u - s\|_p < \epsilon/2$. Como s es simple y $p < \infty$, su soporte deberá tener medida finita. También podemos asumir que $s(x) = 0$ para todo $x \in \Omega^c$. Aplicando el teorema de Lusin 3.4, obtenemos una función $\phi \in C(\Omega)$ tal que $|\phi(x)| \leq \|s\|_\infty$ para todo $x \in \Omega$. Además,

$$\mu\{x \in \Omega : s(x) \neq \phi(x)\} < (\epsilon/4\|s\|_\infty)^p.$$

Por lo tanto, por la propiedad 3.1, tenemos:

$$\begin{aligned}\|s - \phi\|_p &\leq \|s - \phi\|_\infty (\mu\{x \in \Omega : s(x) \neq \phi(x)\})^{1/p} \\ &< 2\|s\|_\infty (\epsilon/4\|s\|_\infty) = \epsilon/2.\end{aligned}$$

De esto se sigue $\|u - \phi\|_p < \epsilon$. □

3.2.1. Más propiedades

Antes de profundizar más sobre estos espacios, necesitaremos enunciar algunas definiciones y propiedades pertinentes para lo que seguirá después.

A lo largo de la tesis utilizaremos la notación habitual de derivada, aunque en muchos casos estaremos refiriéndonos a la derivada débil de una función.

Definición 3.8. Sea $\Omega \subset \mathbb{R}^n$. Dado un entero no negativo m, definimos a $C^m(\Omega)$ como al espacio que consiste en todas las funciones ϕ, las cuales, junto a todas sus derivadas parciales $D^\alpha\phi$ de orden $|\alpha| \leq m$[2], son continuas en Ω. Abreviamos a $C^0(\Omega) = C(\Omega)$ las funciones continuas y $C^\infty(\Omega) = \bigcap_{m=0}^{\infty} C^m(\Omega)$.

Definición 3.9. Sea Ω un dominio en $\mathbb{R}^n$. Denotamos por $\mathcal{C}_0^\infty(\Omega)$ al conjunto de funciones $C^\infty(\Omega)$ con soporte compacto en Ω.

Definición 3.10. Dado un dominio Ω, el conjunto de funciones localmente integrables se denota por

$$L^1_{\text{loc}}(\Omega) := \left\{ f : f \in L^1(K) \quad \forall\, K \text{ compacto } \subset \Omega^\circ \right\}.$$

El siguiente paso es encontrar una definición formal para funciones que sean derivables pero no con la definición formal, sino como una representación por funciones de un espacio L^p.

Sea $g \in C_0^1(\Omega)$ y sea $v \in C_0^\infty(\Omega)$. Aplicando integración por partes, obtenemos:

$$\int_\Omega \frac{\partial g}{\partial x_i}(x)\, v(x)\, \mathrm{d}x = -\int_\Omega g(x)\, \frac{\partial v}{\partial x_i}(x)\, \mathrm{d}x,$$

donde el término de borde se anula debido al soporte compacto de v.

Esto sugiere que si g no es necesariamente derivable en el sentido clásico, pero existe una función u_i tal que

$$\int_\Omega u_i(x)\, v(x)\, \mathrm{d}x = -\int_\Omega g(x)\, \frac{\partial v}{\partial x_i}(x)\, \mathrm{d}x \quad \text{para todo } v \in C_0^\infty(\Omega),$$

entonces podríamos considerar a u_i como la "derivada débil" de g respecto de x_i.

Sea $f \in L^1_{\text{loc}}(\Omega)$. Decimos que una función $u_i \in L^1_{\text{loc}}(\Omega)$ es la *derivada débil* de f respecto de x_i si se cumple que

$$\int_\Omega f(x)\, \frac{\partial \varphi}{\partial x_i}(x)\, \mathrm{d}x = -\int_\Omega u_i(x)\, \varphi(x)\, \mathrm{d}x \quad \text{para todo } \varphi \in C_0^\infty(\Omega).$$

En ese caso escribimos $\frac{\partial f}{\partial x_i} = u_i$ en el sentido débil. Esta idea puede extenderse naturalmente a derivadas de orden superior utilizando la notación multiíndice. En este caso, la derivada débil de orden $|\alpha|$, denotada $D^\alpha f$, se define del siguiente modo:

[2]Un multiíndice α es una n-tupla de enteros no negativos α_i, donde su tamaño esta definido por $|\alpha| = \sum_{i=1}^n \alpha_i$.
Dada una función $\phi \in C^\infty$ denotamos $D^\alpha\phi = \frac{\partial^{|\alpha|}\phi}{\partial x_1^{\alpha_1} \cdots \partial x_n^{\alpha_n}}$.

Definición 3.11. Decimos que una función $f \in L^1_{\text{loc}}(\Omega)$ tiene una derivada débil $D^\alpha f$, de orden $|\alpha|$, si existe una función $g \in L^1_{\text{loc}}(\Omega)$ tal que

$$\int_\Omega g(x)\,\phi(x)\,\mathrm{d}x = (-1)^{|\alpha|}\int_\Omega f(x)\,D^\alpha\phi(x)\,\mathrm{d}x \quad \forall\,\phi \in C_0^\infty(\Omega).$$

Si tal función g existe, definimos $D^\alpha f = g$.

Formalmente definimos:

Definición 3.12 (Espacios de Sobolev)**.** Sea $\Omega \subseteq \mathbb{R}^n$ un abierto y $1 \leq p \leq \infty$. Para un entero $k \geq 0$, definimos el espacio de Sobolev $W^{k,p}(\Omega)$ como el conjunto de funciones $f \in L^p(\Omega)$ tales que todas las derivadas débiles $D^\alpha f$ de orden $|\alpha| \leq k$ existen y pertenecen a $L^p(\Omega)$. Es decir,

$$W^{k,p}(\Omega) := \{f \in L^p(\Omega) \mid D^\alpha f \in L^p(\Omega) \text{ para todo } \alpha \in \mathbb{N}_0^n \text{ con } |\alpha| \leq k\}.$$

También definimos:

$$H^k(\Omega) = W^{k,2}(\Omega).$$

Definición 3.13. Sea k un entero no negativo, y sea $f \in L^1_{\text{loc}}(\Omega)$. Supongamos que las derivadas débiles $D^\alpha f$ existen para todo $|\alpha| \leq k$. Definimos la norma de Sobolev:

$$\|f\|_{W^{k,p}(\Omega)} := \left(\sum_{|\alpha|\leq k} \|D^\alpha f\|^p_{L^p(\Omega)}\right)^{1/p}, \quad \text{para } 1 \leq p < \infty,$$

y, en el caso $p = \infty$,

$$\|f\|_{W^{k,\infty(\Omega)}} := \max_{|\alpha|\leq k} \|D^\alpha f\|_{L^\infty(\Omega)}.$$

También, vamos a utilizar ciertos casos particulares del espacio anterior, por lo que es pertinente definirlos aparte:

Definición 3.14. Dado $1 \leq p < \infty$ denotamos a $W_0^{k,p}(\Omega) = \overline{C_0^\infty}(\Omega)$. Donde la clausura se toma respecto a la norma $\|\cdot\|_{k,p}$. Además, definimos $H_0^k(\Omega) = W_0^{k,2}(\Omega)$.

Teorema 3.15. *El espacio de Sobolev $W^{k,p}(\Omega)$ es un espacio de Banach.*

También introduciremos dos desigualdades que serán clave a la hora de demostrar nuestros resultados:

Teorema 3.16 (Desigualdad de Poincaré)**.** *La desigualdad de Poincaré clásica establece que, dado un exponente p tal que $1 \leq p < \infty$ y un dominio Ω abierto, si suponemos que existen constantes $a, b \in \mathbb{R}$, $a < b$ tales que $\Omega \subset \{x = (x_1, x') \in \mathbb{R}^n : a < x_1 < b\}$, entonces existe una constante C que depende únicamente de $(b - a)$ y de p, tal que para cualquier función f que pertenezca al espacio de Sobolev $W_0^{1,p}(\Omega)$ se cumple la siguiente desigualdad:*

$$\|u\|_{L^p(\Omega)} \leq C\|\nabla u\|_{L^p(\Omega)}, \tag{12}$$

Demostración. Dado que $W_0^{1,p}(\Omega)$ es la clausura de las funciones suaves con soporte compacto, basta establecer el teorema para funciones $f \in C_0^\infty(\mathbb{R}^n)$ tales que

$$\operatorname{supp}(f) \subset \{a < x_1 < b\}.$$

Sea entonces una función f con esa propiedad. Podemos escribir:

$$f(x) = f(x_1, x') = \int_a^{x_1} \frac{d}{dx_1} f(t, x')\, \mathrm{d}t,$$

donde hemos usado que $f(a, x') = 0$ debido al soporte compacto.

Aplicando la desigualdad de Hölder, se deduce:

$$|f(x)|^p = \left| \int_a^{x_1} \frac{\partial f}{\partial x_1}(t, x')\, \mathrm{d}t \right|^p \leq (b-a)^{p-1} \int_a^b \left| \frac{\partial f}{\partial x_1}(t, x') \right|^p \mathrm{d}t \leq (b-a)^{p-1} \int_a^b |\nabla f(t, x')|^p \, \mathrm{d}t.$$

Ahora integramos en $\mathbb{R}^n$, separando las variables x_1 y $x' \in \mathbb{R}^{n-1}$:

$$\int_{\mathbb{R}^n} |f(x)|^p \, \mathrm{d}x \leq (b-a)^{p-1} \int_{\mathbb{R}^{n-1}} \int_a^b \int_a^b |\nabla f(t, x')|^p \, \mathrm{d}t \, \mathrm{d}x_1 \, \mathrm{d}x' = (b-a)^p \int_{\mathbb{R}^n} |\nabla f(t)|^p \, \mathrm{d}x.$$

Finalmente, observando que $\operatorname{supp}(f) \subset \Omega$, se concluye el teorema.

□

3.3. Densidad de funciones suaves

Hasta ahora, hemos explorado algunos conceptos de redes neuronales y teoría general de espacios de Sobolev, pero surge una pregunta natural: ¿es posible conectar ambas teorías?

La respuesta es afirmativa. En [13], se muestra que es posible aproximar funciones en un espacio de Sóbolev mediante una sucesión de funciones generada por una red neuronal arbitraria, asumiendo ciertas condiciones sobre el dominio. Los resultados de [13] se basan en un argumento de densidad, por lo cual comenzamos mostrando que las funciones suaves son densas en los Sóbolev.

Definición 3.17. Sea J una función no negativa real perteneciente a $C_0^\infty(\mathbb{R}^n)$ con las siguientes propiedades:

(i) $J(x) = 0$ si $|x| \geq 1$.

(ii) $\int_{\mathbb{R}^n} J ds = 1$.

Por ejemplo, podemos tomar $J(x) = \begin{cases} k \exp[-1/(1-|x|^2)] & \text{si } |x| < 1 \\ 0 & \text{si } |x| \geq 1, \end{cases}$ Donde $k > 0$ es elegido de forma tal que la condición (ii) se cumpla. Si tomamos $\epsilon > 0$, la función $J_\epsilon(x) = \epsilon^{-n} J(x/\epsilon)$ es no negativa, pertenece a C_0^∞ y cumple,

(i) $J_\epsilon(x) = 0$ si $|x| \geq \epsilon$.

(ii) $\int_{\mathbb{R}^n} J_\epsilon ds = 1$.

J_ϵ es llamado un regularizante o una aproximación de la identidad y la convolución

$$J_\epsilon * u(x) = \int_{\mathbb{R}^n} J_\epsilon(x-y)u(y)dy, \tag{13}$$

definida para la función u tiene sentido y se llama regularización de u.
Generalmente, este proceso de regularización se realiza de la siguiente forma: Primero, se considera un recubrimiento del dominio Ω (el cual uno trabaja) mediante bolas abiertas $\{B_k\}_{k\geq 1}$, cuyas clausuras están contenidas en Ω. Luego, se construye una sucesión $\{\phi_k\}_{k\geq 1}$ de funciones en $C_0^\infty(\Omega)$ que forma una **partición de la unidad subordinada** a dicho recubrimiento [3].

Dada una función $u \in L^p(\Omega)$, multiplicamos por cada ϕ_k para obtener funciones $u\phi_k$ que tienen soporte compacto en las bolas B_k. Ahora, al producto de $u\phi_k$ podemos aplicar un mollifier (regularizarlas) y obtener funciones suaves w_k.

Finalmente, se define la función

$$w := \sum_k w_k,$$

la cual es suave, ya que todas las w_k lo son y la suma es localmente finita. Esta función w aproxima a u en la seminorma. Este proceso lo podemos ilustrar en el siguiente gráfico:

[3]Esto significa que las funciones ϕ_k cumplen:

1. $0 \leq \phi_k(x) \leq 1$ para todo $x \in \Omega$.
2. $\text{supp}(\phi_k) \subset B_k$.
3. $\sum_k \phi_k(x) = 1$ para todo $x \in \Omega$.

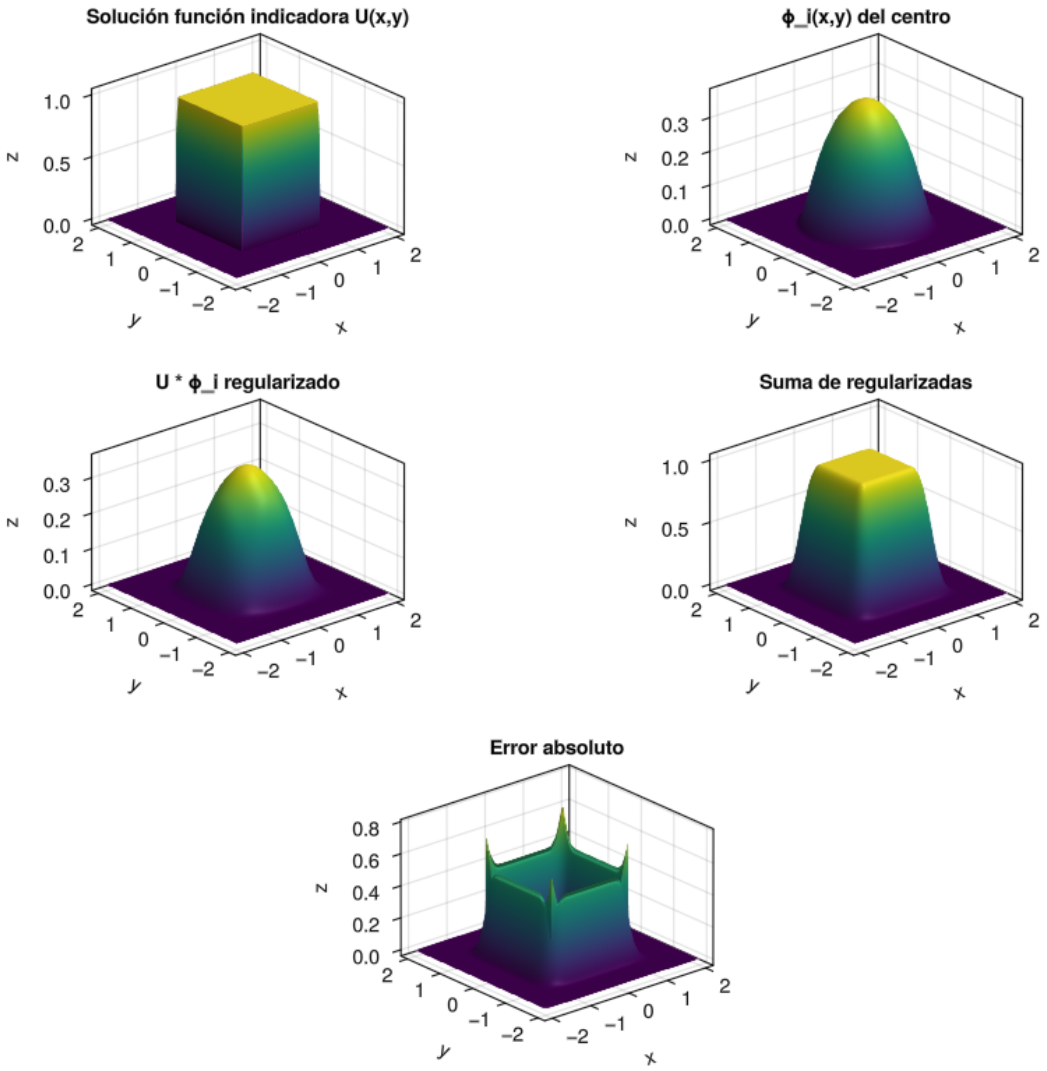


Figura 4: **Proceso de regularización de una función indicadora de un cuadrado.** Realizado con $i = 8$ funciones $\phi_i(x, y) = e^{\frac{1}{1-\left(\frac{\|(x,y)\|}{r}\right)^2}}$, definidas como nulas fuera de una bola de radio r. El parámetro r determina el tamaño del soporte de cada función ϕ_i, es decir, el área en la cual tienen efecto. Fuera de ese radio, ϕ_i vale cero. Esta construcción permite mostrar cómo se suaviza cada producto $u \cdot \phi_i$ mediante una mollificación, y luego cómo se suman estos términos suavizados para aproximar la función original de manera regular.

Ahora vamos a enunciar las propiedades fundamentales sobre esta función.

Lema 3.18. *Sea u una función de soporte compacto en Ω, entonces:*

(a) Si $u \in L^1_{loc}(\mathbb{R}^n)$, entonces $J_\epsilon * u \in C^\infty(\mathbb{R}^n)$.

(b) Si $u \in L^1_{loc}(\Omega)$ y además $sop(u) \subset\subset \Omega$ entonces, $J_\epsilon * u \in C^\infty(\Omega)$ con $\epsilon < dist(sop(u), \partial\Omega)$.

(c) Si $u \in L^p(\Omega)$ con $1 \leq p < \infty$, entonces $J_\epsilon * u \in L^p(\Omega)$. Más aun,

$$\|J_\epsilon * u\|_p \leq \|u\|_p \quad y \quad \lim_{\epsilon \to 0^+} \|J_\epsilon * u - u\|_p = 0.$$

(d) Si $u \in C(\Omega)$ y $G \subset\subset \Omega$, entonces $\lim_{\epsilon \to 0^+} J_\epsilon * u = u(x)$ uniformemente en G

(e) Si $u \in C(\overline{\Omega})$ entonces $\lim_{\epsilon \to 0^+} J_\epsilon * u = u(x)$ uniformemente en Ω.

Demostración. Para demostrar (a), el argumento clave es aplicar el **teorema de derivación bajo el signo de la integral**:

Lema 3.19. *Sean $U \subset \mathbb{R}^n$ abierto, $V \subset \mathbb{R}^m$ medible, $f : U \times V \to \mathbb{R}$ una función medible, $x_0 \in U$ y $j \in \{1, \ldots, n\}$. Supongamos que existe $\epsilon > 0$ tal que:*

- *Para todo $x \in B_\epsilon(x_0)$, se tiene $f(x, \cdot) \in L^1(V)$,*
- *Para casi todo $y \in V$, la función $f(\cdot, y)$ es derivable con respecto a x_j en $B_\epsilon(x_0)$,*
- *Existe $g \in L^1(V)$ tal que $|\partial_{x_j} f(x, y)| \leq g(y)$ para todo $x \in B_\epsilon(x_0)$ y casi todo $y \in V$,*

entonces la función

$$F(x) := \int_V f(x, y)\, \mathrm{d}y$$

es derivable con respecto a x_j en $B_\epsilon(x_0)$, y la derivada está dada por:

$$\partial_j F(x) = \int_V \partial_{x_j} f(x, y)\, \mathrm{d}y.$$

Aplicamos este lema para probar que, si $u \in L^1_{\mathrm{loc}}(\mathbb{R}^n)$, entonces $J_\epsilon * u \in C^\infty(\mathbb{R}^n)$.
Sea

$$f(x, y) := J_\epsilon(x - y)u(y), \qquad x, y \in \mathbb{R}^n,$$

y definimos:

$$F(x) := \int_{\mathbb{R}^n} f(x, y)\, \mathrm{d}y = \int_{\mathbb{R}^n} J_\epsilon(x - y)u(y)\, \mathrm{d}y = (J_\epsilon * u)(x).$$

Verificamos las hipótesis del lema:

- Como $u \in L^1_{\mathrm{loc}}(\mathbb{R}^n)$, entonces para cada $x \in \mathbb{R}^n$, la función $y \mapsto J_\epsilon(x - y)u(y)$ es integrable, ya que $J_\epsilon(x - y)$ tiene soporte compacto en y y es acotada.
- Para cada y, la función $x \mapsto J_\epsilon(x - y)$ es C^∞ , y la derivada en x es simplemente $D^\alpha_x J_\epsilon(x - y)$. Por lo tanto, $f(\cdot, y)$ es derivable en x (incluso suave) para todo y.
- Para cada derivada D^α_x, existe una cota uniforme:

  $$|D^\alpha_x J_\epsilon(x - y)u(y)| \leq C_\alpha\, |u(y)|\, \chi_{B_\epsilon(x)}(y),$$

  donde C_α depende de la derivada del J_ϵ. Como $u \in L^1_{\mathrm{loc}}$, el lado derecho es integrable en y.

Así, todas las hipótesis del lema se cumplen, y podemos concluir que $F(x) = (J_\epsilon * u)(x)$ es diferenciable, y

$$D^\alpha (J_\epsilon * u)(x) = \int_{\mathbb{R}^n} D_x^\alpha J_\epsilon (x-y) u(y)\, \mathrm{d}y.$$

Dado que este razonamiento se aplica a cualquier derivada de cualquier orden, se concluye que $J_\epsilon * u \in C^\infty(\mathbb{R}^n)$.

Para la parte (b) basta observar que $\operatorname{sop}(u * J_\epsilon) = \operatorname{sop}(u) + \operatorname{sop}(J_\epsilon)$.

Ahora supongamos que $u \in L^p(\Omega)$. Si $1 < p < \infty$, tomamos $p' = p/(p-1)$ y por la desigualdad de Hölder (tomando como $f(y) = J_\epsilon(x-y)^{1/p'}$ y $g(y) = u(y) J_\epsilon(x-y)^{1/p}$,

$$\begin{aligned} |J_\epsilon * u(x)| &= \left| \int_{\mathbb{R}^n} J_\epsilon(x-y) u(y) \mathrm{d}y \right| \\ &\leq \left\{ \int_{\mathbb{R}^n} J_\epsilon(x-y) \mathrm{d}y \right\}^{1/p'} \left\{ \int_{\mathbb{R}^n} J_\epsilon(x-y) |u(y)|^p \mathrm{d}y \right\}^{1/p} \\ &= \left\{ \int_{\mathbb{R}^n} J_\epsilon(x-y) |u(y)|^p \mathrm{d}y \right\}^{1/p} \end{aligned}$$

Luego, por el teorema de Fubini,

$$\int_\Omega |J_\epsilon * u(x)|^p \mathrm{d}x \leq \int_{\mathbb{R}^n} \int_{\mathbb{R}^n} J_\epsilon(x-y) |u(y)|^p \mathrm{d}y \mathrm{d}x \tag{14}$$

$$= \int_{\mathbb{R}^n} |u(y)|^p \mathrm{d}y \int_{\mathbb{R}^n} J_\epsilon(x-y) \mathrm{d}x = \|u\|_p^p. \tag{15}$$

Sea $\eta > 0$. Por el teorema 3.7, existe una función $\phi \in C(\Omega)$ tal que $\|u - \phi\|_p < \eta/3$. Utilizando la desigualdad (15) obtenemos $\|J_\epsilon * u - J_\epsilon * \phi\|_p < \eta/3$. Ahora bien, como ϕ es uniformemente continua sobre Ω:

$$\|J_\epsilon * \phi(x) - \phi(x)\| = \left\{ \int_{\mathbb{R}^n} J_\epsilon(x-y)(\phi(y) - \phi(x)) \mathrm{d}y \right\}. \tag{16}$$

$$\leq \sup_{|y-x|<\epsilon} |\phi(y) - \phi(x)|. \tag{17}$$

En el lado derecho de (17) tiende a 0 si $\epsilon \to 0^+$. También, dado que $sop(\phi)$ es compacto, podemos obtener $\|J_\epsilon * \phi - \phi\|_p < \eta/3$, tomando ϵ suficientemente pequeño. Para ese ϵ tenemos $\|J_\epsilon * u - u\|_p < \eta$ como queríamos. Si $p = 1$ la desigualdad de (15) se sigue directamente de (13) sin usar Hölder y el resto de la prueba es igual. Después, para demostrar (d) y (e) tenemos que reemplazar a ϕ por u en (17) y listo.

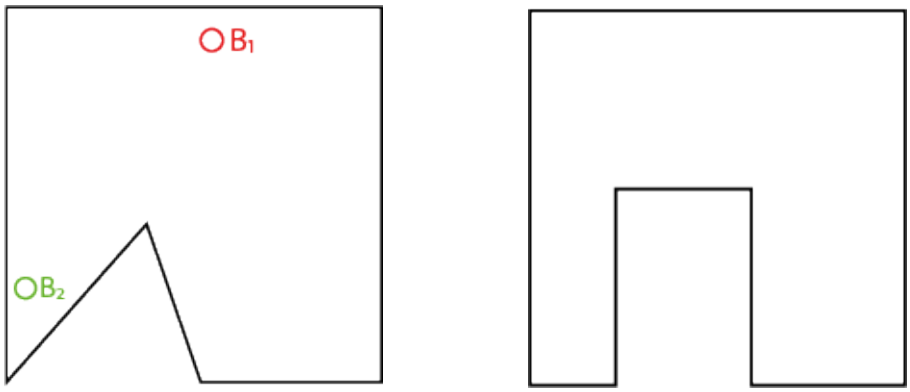
□

Finalmente estamos en condiciones de probar la densidad de $C_0^\infty(\mathbb{R}^n)$ en los espacios de Sóbolev $W^{k,p}(\Omega)$. Resultados de este tipo dependen fuertemente de las características del dominio Ω, pero son válidos para clases muy generales de dominios. En [1, Teorema 3.22] se prueba que si Ω satisface la propiedad del segmento, entonces las restricciones a Ω de funciones en $C_0^\infty(\mathbb{R}^n)$ son densas en $W^{k,p}(\Omega)$. Aquí damos una demostración algo más sencilla, para dominios estrellados, que es suficiente para los fines de esta tesis.

Definición 3.20. Un conjunto abierto $\Omega \subset \mathbb{R}^n$ se dice que es **estrellado con respecto a un punto** $x_0 \in \Omega$ si para todo punto $x \in \Omega$, el segmento de línea recta que une x_0 con x está contenido completamente en Ω; es decir,

$$\forall x \in \Omega, \quad \forall t \in [0,1], \quad x_0 + t(x - x_0) \in \Omega.$$

Equivalente y geométricamente, esto significa que cualquier **rayo** de la forma $\{x_0+tv : t \geq 0\}$, con $v \in \mathbb{R}^n$ tal que $x_0 + tv \in \Omega$ para algún $t > 0$, intersecta el borde $\partial\Omega$ a lo sumo en un único punto. En otras palabras, todos los puntos de Ω son "visibles"desde x_0, como podemos observar en el siguiente ejemplo.



(a) (b)

Figura 5: Por un lado el conjunto (a) es estrellado con respecto a B_1 pero **no** a B_2. Por otro lado el conjunto (b) no es estrellado con respecto a ningun punto.

Teorema 3.21. *Si Ω es acotado, estrellado respecto de un punto, entonces $C^\infty(\overline{\Omega})$ es denso en $W^{k,p}(\Omega)$.*

Demostración. Sea $u \in W^{k,p}(\Omega)$. Sin pérdida de generalidad, podemos asumir que Ω es estrellado respecto del origen. Notamos $u_\tau(x) = u(\tau x)$ con $\tau \in (0,1)$. Observemos que $\|u - u_\tau\|_{L^p(\Omega)} \to 0$ si $\tau \to 1$. Utilizando la definición de derivada en sentido débil, obtenemos que $D^\alpha(u_\tau) = \tau^k (D^\alpha u)_\tau$, con $|\alpha| = k$. Esto implica que $u_\tau \in W^{k,p}(\tau^{-1}\Omega)$ y

$$\|D^\alpha(u - u_\tau)\|_{L^p(\Omega)} \leq (1 - \tau^k)\|D^\alpha u\|_{L^p(\Omega)} + \|D^\alpha u - (D^\alpha u)_\tau\|_{L^p(\Omega)}.$$

El miembro derecho de esta desigualdad tiende a 0 si $\tau \to 1$, por lo que $u_\tau \to u$ en $W^{k,p}(\Omega)$.

Dado $\epsilon_0 > 0$, tomamos τ_0 tal que para cualquier $x \in \partial\Omega$, $B(x, \epsilon_0) \subset \tau^{-1}\Omega$ para todo $\tau < \tau_0$, tal como muestra la Figura 6. En particular, tenemos que $\overline{\Omega} \subset \tau_0^{-1}\Omega$. Dado J_ϵ definido en (3.17) para $\epsilon < \epsilon_0$ consideramos $J_\epsilon * u_{\tau_0}$. Por lo visto en el Lema 3.18, esta convolución es C^∞ y aproxima a u_{τ_0}, en el sentido de que $J_\epsilon * u_{\tau_0}$ converge a u_{τ_0} en $Wk, p(\Omega)$. De esta forma podemos tomar una sucesión τ_ℓ creciente tal que $J_{\epsilon_\ell} * u_{\tau_\ell}$ aproxime a u_{τ_ℓ} en $W^{k,p}(\Omega)$. Usando un argumento diagonal podemos elegir una sub-sucesión de funciones C^∞ que aproxime a u en $W^{k,p}(\Omega)$.

Esto demuestra la densidad deseada. □

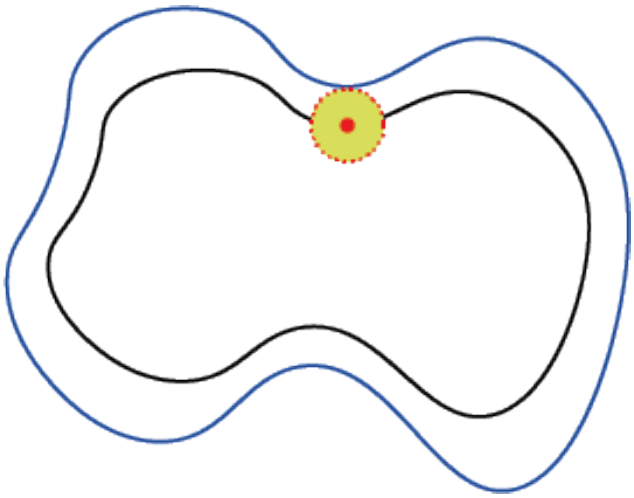


Figura 6: Conjunto estrellado agrandado un ϵ

Observación 2. *Es importante destacar que la demostración anterior depende crucialmente de que el dominio sea estrellado respecto a un punto. Esta propiedad garantiza que, al aplicar una dilatación centrada en dicho punto, el dominio original queda contenido en su imagen agrandada. En dominios que no son estrellados, como, por ejemplo, un círculo con una abertura en el medio, esta inclusión puede fallar: la dilatación podría sacar puntos del dominio original, impidiendo definir la convolución de manera adecuada y, por lo tanto, invalidando el argumento utilizado.*

Como corolario inmediato del teorema 3.7 y 3.18[(b,e)] obtenemos el siguiente resultado.

Corolario 3.22. *C_0^∞ es denso en $L^p(\Omega)$ si $1 \leq p < \infty$.*

3.4. Aproximación por redes neuronales

Ahora sí, podemos comenzar a demostrar que si tenemos una función en un espacio de Sobolev arbitrario, podemos estimarla con una red.

Por simplicidad, el resultado está formulado para redes de una capa oculta y un solo output. El conjunto de todas las redes sobre $\mathbb{R}^n$ es:

Definición 3.23. $\mathscr{R}_k^{(n)}(\psi) := \left\{h : \mathbb{R}^k \to \mathbb{R} : h(x) = \sum_{j=1}^n \beta_j \psi(a_j' x - \theta_j)\right\}$
Donde ψ es la función de activación y $'$ denota la transpuesta, por lo que si a tiene componentes $\alpha_1 \cdots, \alpha_k$ y x tiene $\xi_1 \cdots \xi_k$, entonces el producto $a' x = \alpha_1 \xi_1 + \cdots + \alpha_k \xi_k$.

El conjunto de todas las redes de este tipo con activación ψ es:

$$\mathscr{R}_k(\psi) = \bigcup_{n=1}^{\infty} \mathscr{R}_k^{(n)}(\psi).$$

En general, nuestras funciones de activación son sigmoideas, que serían suficientes para lo que probaremos ahora. Sin embargo, uno puede demostrar el resultado para funciones de activación más generales, llamadas **discriminadoras**.

Definición 3.24. Una función acotada ψ es llamada discriminadora, si dada una medida signada finita μ en $\mathbb{R}^k$ tal que,

$$\int_{\mathbb{R}^k} \psi(a'x - \theta)\mathrm{d}\mu(x) = 0 \quad \text{para todo } a \in \mathbb{R}^k,\ \theta \in \mathbb{R},$$

entonces la medida μ es la medida nula.

Con esta definición podemos enunciar un lema. Su demostración se encuentra en [13, Theorem 5].

Teorema 3.25. *Si ψ es acotada y no constante, entonces es discriminatoria.*

De este teorema, podemos observar que cualquier función de activación que cumpla propiedades básicas como que sea acotada y no constante nos va a ser de utilidad. En estas se encuentran, por ejemplo, las sigmoideas o las ReLU vistas anteriormente.

Dada J_ϵ la función dada en la Definición 3.17, notamos:

$$\hat{J}_\epsilon\mu(x) = \int_{\mathbb{R}^k} J_\epsilon(x - y)\mathrm{d}\mu(y).$$

Lema 3.26. *Supongamos que la función f y la medida σ cumplen alguna de las dos siguientes condiciones:*

(a) f es continua y μ es una medida finita signada con soporte compacto;

(b) f es acotada y continua y μ es una medida finita signada.

Si notamos T_y la traslación $T_y f(x) = f(x + y)$, entonces:

$$\int_{\mathbb{R}^k} f\hat{J}_\epsilon\mu\mathrm{d}x = \int_{\mathbb{R}^k} \left[\int_{\mathbb{R}^k} T_y f d\mu\right] J_\epsilon(y)\mathrm{d}y$$

Demostración. Primero, notamos que utilizando las condiciones (a) o (b) nos garantizan el poder utilizar el teorema de Fubini. De esa forma, la prueba se realiza con un cambio de variable:

$$\begin{aligned}\int_{\mathbb{R}^k} f\hat{J}_\epsilon\mu\mathrm{d}x &= \int_{\mathbb{R}^k}\int_{\mathbb{R}^k} f(x)J_\epsilon(x - y)dxd\mu(y)\\ &= \int_{\mathbb{R}^k}\int_{\mathbb{R}^{\mathbb{k}}} f(z + y)J_\epsilon(z)dzd\mu(y)\\ &= \int_{\mathbb{R}^k}\int_{\mathbb{R}^k} T_y f(z)J_\epsilon(z)dzd\mu(y)\end{aligned}$$

Donde en la primera igualdad utilizamos Fubini y en la segunda el cambio de variable $z = x - y$. □

Antes de enunciar el teorema de aproximación a las funciones C^∞ por las redes, veamos una definición y un par de teoremas del análisis funcional.

Teorema 3.27 (Hahn–Banach)**.** *Sea E un $\mathbb{R}$ espacio vectorial y $p : E \to \mathbb{R}$ una función que satisface:*

1. $p(\lambda x) = \lambda p(x) \quad \forall x \in E \ y \ \forall \lambda > 0,$

2. $p(x + y) \leq p(x) + p(y) \quad \forall x, y \in E.$

Sea $G \subset E$ un subespacio lineal y sea $g : G \to \mathbb{R}$ un funcional lineal tal que

$$g(x) \leq p(x) \quad \forall x \in G.$$

Bajo estas condiciones, existe un funcional lineal f definido en todo E que extiende a g, es decir, $f(x) = g(x)$ para todo $x \in G$ y que también satisface $f(x) \leq p(x)$ para todos $x \in E$.

Teorema 3.28 (Teorema de Representación de Riesz)**.** *Sea $\varphi \in (L^1)^*$. Entonces, existe una única función $u \in L^\infty$ tal que*

$$\langle \varphi, f \rangle = \int_\Omega uf \, dx \quad \forall f \in L^1.$$

Además,

$$\|u\|_\infty = \|\varphi\|_{(L^1)^*}.$$

Definición 3.29. Sea $C^m(\mathbb{R}^k)$ el espacio de todas las funciones f que tienen todas sus derivadas parciales de orden $|\alpha| \leq m$ continuas en $\mathbb{R}^k$. Un subconjunto S de $C^m(\mathbb{R}^k)$ se le dice **uniformemente m-denso en compactos** en $C^m(\mathbb{R}^k)$, si para todo $f \in C^m(\mathbb{R}^k)$, para todo conjunto compacto X de $\mathbb{R}^k$ y para todo $\epsilon > 0$ existe una función $g = g(f, X, \epsilon) \in S$ tal que $\|f - g\|_{m,\alpha,X} < \epsilon$. Con

$$\|f\|_{m,\infty,X} := \max_{|\alpha| \leq m} \sup_{x \in X} |D^\alpha f(x)|.$$

Para una función $f \in C^m(\mathbb{R}^k)$, μ una medida finita en $\mathbb{R}^k$ y $1 \leq p < \infty$, definimos:

$$\|f\|_{m,p,\mu} := \left[\sum_{|\alpha| \leq m} \int_{\mathbb{R}^k} |D^\alpha f| d\mu \right]^{1/p}$$

Ahora veamos el resultado:

Teorema 3.30. *Si $\psi \in C^m(\mathbb{R}^k)$ no es constante y es acotada, entonces $\mathscr{R}_k(\psi)$ es uniformemente m-denso en compactos en $C^m(\mathbb{R}^k)$.*

Demostración. Supongamos que no es cierto. Es decir, asumamos que existe un conjunto compacto X tal que $\overline{\mathscr{R}_k(\psi)} \neq C^m(X)$. Observemos primero que $\mathscr{R}_k(\psi)$ es un subespacio propio de $C^m(X)$. Esto es inmediato, dado que contiene al 0, y las combinaciones lineales de funciones en $\mathscr{R}_k(\psi)$ son funciones del mismo espacio. Aplicando el Teorema de Hahn-Banach 3.27 obtenemos la existencia de un funcional $\Lambda : C^k(X) \to \mathbb{R}$ tal que $\Lambda|_{\mathscr{R}_k(\psi)} = 0$ pero $\Lambda \neq 0$. Luego, el Teorema de Riesz 3.28 nos dice que existe una colección μ_α, $|\alpha| \leq m$ (una por cada multi-índice α) de medidas finitas signadas con soporte en algún compacto X de $\mathbb{R}^k$ tal que el funcional

$$\Lambda(f) = \sum_{|\alpha| \leq m} \int_{\mathbb{R}^k} D^\alpha f d\mu_\alpha,$$

se anula en $\mathscr{R}_k(\psi)$ pero no es idénticamente cero en $C^m(\mathbb{R}^k)$. Expliquemos brevemente este paso:
Cuando aplicamos el Teorema de Riesz, obtenemos una representación que depende de cada derivada, por ese motivo, sumamos en todas las derivadas y ahí es cuando se obtiene una colección de medidas. En resumen, las derivadas son nuestras f y cada medida existe por el teorema de Riesz.
Ahora, vamos a convolucionar al igual que veníamos haciendo, con un núcleo regularizante. Entonces definimos

$$\Lambda_\epsilon(f) = \sum_{|\alpha|\leq m} \int_{\mathbb{R}^k} D^\alpha f \hat{J}_\epsilon \mu_\alpha dx,$$

Expandiendo la integral y utilizando el Lema 3.26

$$\begin{aligned}\Lambda_\epsilon(f) &= \int_{\mathbb{R}^k} \left[\sum_{|\alpha|\leq m} \int_{\mathbb{R}^k} D^\alpha T_y f \, d\mu_\alpha \right] J_\epsilon(y) \mathrm{d}y, \\ &= \int_{\mathbb{R}^k} \Lambda(T_y f) J_\epsilon(y) \mathrm{d}y.\end{aligned}$$

Finalmente, podemos tomar límite usando el Lema 3.18, lo que da

$$\lim_{\epsilon\to 0} \Lambda_\epsilon(f) = \Lambda(f)$$

para toda $f \in C^m(\mathbb{R}^k)$. Ahora, como sabemos que los regularizadores son C^∞, nos gustaría pasarle las derivadas de la f. Para eso integramos por partes:

$$\Lambda_\epsilon(f) = \int_{\mathbb{R}^k} \underbrace{\left[\sum_{|\alpha|\leq m} (-1)^\alpha D^\alpha \hat{J}_\epsilon \mu_\alpha \right]}_{:= h_\epsilon} f \, dx.$$

Primero escribamos $\psi_{a,\theta}(x) = \psi(a'x - \theta)$. Ahora recordemos que habíamos supuesto que Λ se anula en $\mathscr{R}_k(\psi)$, entonces, como $\psi_{a,\theta} \in \mathscr{R}_k(\psi)$ para todo $a \in \mathbb{R}^k$ y todo $\theta \in \mathbb{R}$, $\Lambda(\psi_{a,\theta}) = 0$. Como aplicar una traslación a ψ hace que siga dentro de $\mathscr{R}_k(\psi)$, obtenemos $\Lambda(T_y\psi_{a,\theta}) = 0$ para todo $a, y \in \mathbb{R}^k$ y $\theta \in \mathbb{R}$. Aplicando la expresión anterior para $\Lambda_\epsilon(\psi_{a,\theta})$ tenemos:

$$\int_{\mathbb{R}^k} \psi_{a,\theta} h_\epsilon dx = \Lambda_\epsilon(\psi_{a,\theta}) = \int_{\mathbb{R}^k} \Lambda(T_y\psi_{a,\theta}) J_\epsilon(y) \mathrm{d}y = 0,$$

para toda $a \in \mathbb{R}^k$ y $\theta \in \mathbb{R}$.
Luego, por hipótesis, sabemos que ψ es acotada y no constante, por lo que podemos aplicar el Teorema 3.25, que nos dice entonces que la medida en realidad es la medida nula, por lo que $h_\epsilon \equiv 0$ (considerando la medida $h_\epsilon \mathrm{d}x$). Pero entonces $\Lambda_\epsilon(f) = \int_{\mathbb{R}^k} f h_\epsilon dx$ es nulo para toda función $f \in C^m(\mathbb{R}^k)$, por lo tanto, $\Lambda(f) = \lim_{\epsilon\to 0} \Lambda_\epsilon(f) = 0$ para toda $f \in C^m(\mathbb{R}^k)$. Lo que nos lleva a un absurdo, puesto que es precisamente lo que supusimos que no debía pasar. □

4 Mínimos cuadrados para sistemas de primer orden

Como anticipamos, nuestro objetivo será aproximar la solución de ecuaciones elípticas de segundo orden a través de redes neuronales. Para ello, daremos la formulación mixta de las ecuaciones. Es decir: desarrollaremos la ecuación de segundo orden como un sistema de ecuaciones de primer orden. Luego plantearemos un problema de cuadrados mínimos cuya solución coincida con la de las ecuaciones. Finalmente, resolveremos el problema de cuadrados mínimos mediante redes neuronales entrenadas con algoritmos de descenso. En esta sección realizaremos la formulación mixta y el planteo del problema de cuadrados mínimos y probaremos un resultado técnico respecto del funcional que nos interesa minimizar. Este resultado será utilizado en la siguiente sección para probar la Γ-convergencia del método propuesto.

4.1. Formulación Mixta

Sea $\Omega \subset \mathbb{R}^n$, con $n \geq 2$, un dominio acotado con frontera de tipo Lipschitz[4]. Consideremos el siguiente problema, que es una versión ligeramente más general del que nos interesará resolver.

$$\begin{cases} -\operatorname{div}(A\nabla u) + Xu = f & \text{en } \Omega, \\ u = g_{\mathcal{D}} & \text{en } \Gamma_{\mathcal{D}}, \\ A\nabla u \cdot \mathbf{n} = g_{\mathcal{N}} \end{cases}$$

donde $f \in L^2(\Omega)$, $A(x)$ es una matriz simétrica de $n \times n$ cuyas entradas son funciones en $L^\infty(\Omega)$ y X es un operador diferencial lineal de orden a lo sumo 1. Además, $\partial\Omega = \Gamma_D \cup \Gamma_N$, siendo Γ_D la parte de la frontera donde se impone la condición de Dirichlet y Γ_N la parte donde se impone la condición de Neumann, $g_D \in L^2(\Gamma_D)$ y $g_N \in L^2(\Gamma_N)$ y $\mathbf{n}$ es el vector normal unitario saliente a la frontera. Para demostrar el resultado principal de esta sección tomaremos $g_{\mathcal{D}}$ y $g_{\mathcal{N}}$ **iguales a 0**.

Además, asumimos que $A(x)$ es uniformemente definida positiva, es decir, existen constantes positivas.

$$0 < \lambda \leq 1 \leq \Lambda \tag{18}$$

tales que

$$\lambda \xi^T \xi \leq \xi^T A(x) \xi \leq \Lambda \xi^T \xi, \tag{19}$$

para todo $\xi \in \mathbb{R}^n$ y casi todo $x \in \overline{\Omega}$.

Para el operador diferencial X, se pueden considerar varias opciones comunes:

[4]Un dominio $\Omega \subset \mathbb{R}^n$ se dice que tiene **frontera de tipo Lipschitz** si para todo punto $x_0 \in \partial\Omega$, existe un sistema de coordenadas (una isometría de $\mathbb{R}^n$), un radio $r > 0$ y una función Lipschitz $\varphi : \mathbb{R}^{n-1} \to \mathbb{R}$ tal que en esas coordenadas, la intersección del dominio con un cilindro $Q = B' \times (-r, r)$ alrededor de x_0 se puede describir como

$$\Omega \cap Q = \{(x', x_n) \in \mathbb{R}^{n-1} \times \mathbb{R} : x_n > \varphi(x'),\ x' \in B'\}.$$

1 $Xu = 0$. En este caso, el término Xu desaparece, lo que simplifica el problema a una forma más básica.

2 $Xu = \text{div}(bu)$, donde $b = (b_1(x), b_2(x), \ldots, b_n(x)) \in (L^2(\Omega))^n$. Aquí, b es un campo vectorial cuyas componentes son funciones en $L^2(\Omega)$, lo que introduce una dependencia lineal del campo vectorial b en la ecuación.

3 $Xu = a \cdot \nabla u + cu$, donde $a \in (L^2(\Omega))^n$ y $c(x) \in L^2(\Omega)$. En este caso, el operador incluye un término de transporte $a \cdot \nabla u$ y un término de reacción cu, lo que aporta mayor complejidad a la ecuación.

Es muy importante destacar el hecho de que tomamos todas estas medidas para poder luego garantizar solución (y además obtener las propiedades que deseamos). Esta labor no es compleja, los detalles profundos se encuentran en [11] pero, de todas formas, comentaremos algo un poco más adelante.

Continuando, a fin de realizar la formulación mixta, podemos realizar un análisis similar a la Sección 3, donde vimos cómo se realiza la formulación débil, con la salvedad de un detalle. Tenemos que utilizar la igualdad del Teorema de divergencia que dice:

$$\int_\Omega div(u)\phi = -\int_\Omega u\nabla\phi + \int_{\partial\Omega} (\nabla u \cdot \mathbf{n})\phi,$$

para toda $\phi \in H^1(\Omega)$. Ahora sí, repitiendo los argumentos de la Sección 3, podemos multiplicar por una función test $v \in H_0^1(\Omega, \Gamma_\mathcal{D}) = \{w \in H^1(\Omega) : w = 0 \text{ en } \Gamma_\mathcal{D}\}$. e integrar usando el Teorema de la divergencia, obteniendo la formulación débil:

$$u \in H_0^1(\Omega, \Gamma_\mathcal{D}) : \qquad \int_\Omega A\nabla u \cdot \nabla v + \int_\Omega Xuv = \int_\Omega fv, \qquad \forall v \in H_0^1(\Omega, \Gamma_\mathcal{D}).$$

En esta tesis trabajamos con lo que se denomina *formulación mixta* del problema. La formulación mixta consiste simplemente en plantear un problema de orden 2 como un sistema de orden 1. Concretamente, la formulación mixta de nuestro problema original sería:

$$\begin{cases} \boldsymbol{\phi} - A\nabla u = 0 \quad \text{en } \Omega, \\ -\text{div}(\phi) + Xu = f \quad \text{en } \Omega, \\ u = g_\mathcal{D} \quad \text{en } \Gamma_D, \\ \boldsymbol{\phi} \cdot \mathbf{n} = g_\mathcal{N} \quad \text{en } \Gamma_N, \end{cases} \tag{20}$$

La primera ecuación es la identidad $\phi = \alpha\nabla u$, mientras que la segunda escribe una ecuación para ϕ. Nuevamente tomamos $g_\mathcal{D} = g_\mathcal{N} = 0$.

Para la primera ecuación, multiplicamos por una función v y, para la segunda ecuación, por una función q (también, por comodidad, multiplicaremos a toda la primera ecuación por A^{-1}, cosa que no es problema puesto que consideramos A de manera tal que esto sea posible):

$$\int_\Omega A^{-1}\boldsymbol{\phi} \cdot \mathbf{v} - \int_\Omega \nabla u \cdot \mathbf{v} = 0.$$

$$\int_\Omega - \text{div}\, \boldsymbol{\phi}\, q + \int_\Omega Xuq = \int_\Omega f\, q,$$

Utilizando nuevamente el teorema de la divergencia en la primera ecuación, obtenemos:

$$0 = \int_\Omega A^{-1}\boldsymbol{\phi}\cdot\mathbf{v} - \int_\Omega \nabla u\cdot\mathbf{v} = \int_\Omega A^{-1}\boldsymbol{\phi}\cdot\mathbf{v} + \int_\Omega u\,\mathrm{div}\ \mathbf{v} - \int_{\partial\Omega} u\,(\mathbf{v}\cdot n).$$

Dado que $u = 0$ en $\Gamma_{\mathcal{D}}$ esto se simplifica a:

$$0 = \int_\Omega A^{-1}\boldsymbol{\phi}\cdot\mathbf{v} + \int_\Omega u\,\mathrm{div}\ v - \int_{\Gamma_{\mathcal{N}}} u(\mathbf{v}\cdot\mathbf{n}).$$

Al observar las integrales, notamos que la formulación estará bien planteada si tomamos:

$$\mathbf{v}\in H_0(\mathrm{div},\Omega,\Gamma_{\mathcal{N}}) = \{\mathbf{v}\in L^2(\Omega) : \mathrm{div}\ \mathbf{v}\in L^2(\Omega) \text{ y } \mathbf{v}\cdot\mathbf{n} = 0 \text{ en } \Gamma_{\mathcal{N}}\},$$
$$q\in L^2(\Omega).$$

Y entonces el problema a resolver es buscar $(\mathbf{v},q)\in H_0(\mathrm{div},\Omega,\Gamma_{\mathcal{N}})\times L^2(\Omega)$ tal que,

$$\begin{cases} \int_\Omega A^{-1}\boldsymbol{\phi}\cdot\mathbf{v} + \int_\Omega u\,\mathrm{div}\ \mathbf{v} = 0 \quad \forall\mathbf{v}\in H_0(\mathrm{div},\Omega,\Gamma_{\mathcal{N}}), \\ \int_\Omega -\mathrm{div}\ \boldsymbol{\phi}\,q + \int_\Omega Xuq = \int_\Omega f\,q \quad \forall q\in L^2(\Omega). \end{cases}$$

Con esta base, podemos ahora entender una descripción más formal de los espacios de Sobolev y los operadores adjuntos que se utilizan en la formulación débil de los problemas variacionales. En particular, más adelante, exploraremos cómo los operadores lineales y sus adjuntos juegan un papel crucial en el análisis y la solución de estas ecuaciones. Además, es interesante observar que, realizando esta formulación, a diferencia de la formulación débil estándar que conseguimos anteriormente, ganamos regularidad. Es decir, con la formulación débil necesitábamos que valga para funciones en H_0^1, pero ahora, pese a que ahora trabajamos con dos espacios, solo pedimos funciones en $H(\mathrm{div})$ y en L^2, por lo que necesitamos menos regularidad.

4.2. Formulación del problema

Para poder continuar, es necesario agregar varias definiciones. Utilizaremos los espacios de Sobolev clásicos $H^k(\Omega)$, con la norma $\|\cdot\|_{k,\Omega}$ y seminormas $|\cdot|_{k,\Omega}$, donde $0\le k\le\infty$. Como es habitual, el espacio $L^2(\Omega)$ es denotado por $H^0(\Omega)$. Las normas correspondientes en los espacios producto $(H^k(\Omega))^n$ se denotarán por $\|\cdot\|_{k,\Omega,n}$ y $|\cdot|_{k,\Omega,n}$. También utilizamos el espacio de Sobolev

$$H(\mathrm{div},\Omega) = \{\mathbf{v}\in(L^2(\Omega))^n : \mathrm{div}\,\mathbf{v}\in L^2(\Omega)\}$$

Nos interesarán los siguientes espacios:

$$W = \{\mathbf{v}\in H(\mathrm{div},\Omega) : \mathbf{n}\cdot\mathbf{v} = 0\,\mathrm{en}\,\Gamma_N\},$$
$$V = \{q\in H^1(\Omega) : q = 0\,\mathrm{en}\,\Gamma_D\}.$$

con las normas respectivas

$$\|\mathbf{v}\|^2_{H(\mathrm{div},\Omega)} = \|\mathbf{v}\|^2_{0,\Omega} + \|\mathrm{div}\,\mathbf{v}\|^2_{0,\Omega},$$
$$\|q\|^2_{H^1(\Omega)} = \|q\|^2_{0,\Omega} + \|\nabla q\|^2_{0,\Omega}.$$

Como es habitual, el producto interno en $L^2(\Omega)$ se denota por $(\cdot,\cdot)$; es decir, para cualquier $u, q \in L^2(\Omega)$,

$$(u, q) = \int_\Omega u(x)q(x)\,dx.$$

De manera similar, el producto interno en $(L^2(\Omega))^n$ se denota por

$$(\boldsymbol{\phi}, \mathbf{v})_n = \sum_{i=1}^{n} \int_\Omega \phi_i(x)v_i(x)\,dx \quad \forall\, \boldsymbol{\phi}, \mathbf{v} \in (L^2(\Omega))^n.$$

Dado un operador lineal $X : H \to L^2(\Omega)$, denotamos por $X^* : H \to L^2(\Omega)$ a su adjunto formal en $L^2(\Omega)$, el cual está definido por

$$(Xu, q) = (u, X^*q) \quad \forall\, u, q \in C^\infty(\Omega).$$

Es decir, definimos el adjunto sin tener en cuenta las condiciones de borde. De manera similar, dado un operador lineal $X : H \to (L^2(\Omega))^n$, su adjunto formal $X^* : (H^1(\Omega))^n \to L^2(\Omega)$ se define por

$$(Xq, \mathbf{v})_n = (q, X^*\mathbf{v}) \quad \forall\, q \in C^\infty(\Omega), v \in (C^\infty(\Omega))^n.$$

Definimos el operador $\nabla : H^1(\Omega) \to (L^2(\Omega))^n$ como

$$\nabla q = \left(\frac{\partial q}{\partial x_1}, \frac{\partial q}{\partial x_2}, \ldots, \frac{\partial q}{\partial x_n}\right).$$

Su adjunto formal $\nabla^* : (H^1(\Omega))^n \to L^2(\Omega)$ se define como

$$\nabla^*\mathbf{v} = -\operatorname{div}\mathbf{v} = -\left(\frac{\partial v_1}{\partial x_1} + \cdots + \frac{\partial v_n}{\partial x_n}\right).$$

Observar que en efecto este operador cumple la propiedad de ser adjunto, puesto que,

$$(\nabla q, \mathbf{v})_n = \int_\Omega \nabla q \cdot \mathbf{v} = \int_\Omega q(-\operatorname{div}\mathbf{v}) + \int_{\partial\Omega} q\,(\mathbf{n}\cdot\mathbf{v}) = \int_\Omega q(-\operatorname{div}\mathbf{v}) = (q, \nabla^*\mathbf{v})$$

Donde en la segunda igualdad integramos por partes y en el último paso, restringimos a ∇ al espacio V y ∇^* al espacio W y, por lo tanto, las condiciones de borde se anulan de manera natural. Así obtenemos que este operador es el adjunto en $L^2(\Omega)$.

Ahora, la idea será observar una equivalencia entre el sistema dado por (20) y el siguiente funcional:

$$\mathcal{L}(u, \boldsymbol{\phi}, f) = \|A\nabla u - \boldsymbol{\phi}\|^2_{L^2(\Omega)} + \|\nabla^*\boldsymbol{\phi} + Xu - f\|^2_{L^2(\Omega)} \tag{21}$$

Con su correspondiente forma bilineal:

$$\mathcal{F}(\boldsymbol{\phi}, u; \mathbf{v}, q) = \left(\begin{pmatrix} -I & A\nabla \\ \nabla^* & X \end{pmatrix}\begin{pmatrix} \boldsymbol{\phi} \\ u \end{pmatrix}, \begin{pmatrix} -I & A\nabla \\ \nabla^* & X \end{pmatrix}\begin{pmatrix} \mathbf{v} \\ q \end{pmatrix}\right) \tag{22}$$

Donde para obtener esta forma bilineal, escribimos la suma de normas al cuadrado de forma matricial. Al realizar las multiplicaciones correspondientes, obtenemos un producto interno entre dos vectores, lo que nos lleva nuevamente a la definición de (21) (con $f = 0$). Aquí, estamos utilizando el producto interno estándar en L^2. O sea, nuestro objetivo se transformó en encontrar las funciones u y $\boldsymbol{\phi}$ que minimicen el funcional anterior. Esto debe hacerse en un espacio adecuado, definido como:

$$\mathcal{A} := \{q = (u, \boldsymbol{\phi}) \in H^1(\Omega) \times H(\mathrm{div}, \Omega) : u = g_{\mathcal{D}} \text{ en } \Gamma_{\mathcal{D}}, \boldsymbol{\phi} \cdot \mathbf{n} = g_{\mathcal{N}} \text{ en } \Gamma_{\mathcal{N}}\}$$

Ahora, el objetivo que resta de esta sección es demostrar un resultado sobre el operador definido previamente. Específicamente, buscamos probar el siguiente teorema:

Teorema 4.1. *Asumiendo que V es un dominio donde vale Poincaré 12, X satisface la cota (36) y que nuestro sistema (20) tiene solución. Entonces existen constantes positivas α y β tales que*

$$\mathcal{F}(\boldsymbol{\phi}, u; \boldsymbol{v}, q) \leq \beta(||\boldsymbol{\phi}||^2_{H(div)} + ||u||^2_{1,\Omega})^{1/2}(||\boldsymbol{v}||^2_{H(div)} + ||q||^2_{1,\Omega})^{1/2} \tag{23}$$

para cada $\boldsymbol{\phi}, \boldsymbol{v} \in W$ y cada $u, q \in V$ y además,

$$\mathcal{F}(\boldsymbol{\phi}, u; \boldsymbol{\phi}, u) \geq \alpha(||\boldsymbol{\phi}||^2_{H(div)} + ||u||^2_{1,\Omega}) \tag{24}$$

para cada $\boldsymbol{\phi} \in W$ y $u \in V$.

Para demostrar esto, construiremos algunos funcionales auxiliares, con sus correspondientes formas bilineales. Estos funcionales auxiliares resultarán más sencillos de analizar. Finalmente, probaremos su equivalencia con el funcional L, lo que nos dará el resultado deseado.

Como A es simétrica y definida positiva, es diagonalizable mediante una base ortonormal y todos sus autovalores son positivos. Por lo tanto, tiene sentido calcular $A^{1/2}$, y de esta forma podemos plantear el siguiente funcional:

$$\hat{\mathcal{L}}(u, \boldsymbol{\phi}, f) = \|A^{1/2}\nabla u - A^{-1/2}\boldsymbol{\phi}\|^2_{L^2(\Omega)} + \|\nabla^* \boldsymbol{\phi} + Xp - f\|^2_{L^2(\Omega)} \tag{25}$$

$\hat{\mathcal{L}}$ resulta equivalente a $\mathcal{L}$, es decir, valen las siguientes estimaciones:

$$\lambda\hat{\mathcal{L}}(u, \boldsymbol{\phi}, f) \leq \mathcal{L}(u, \boldsymbol{\phi}, f) \leq \Lambda\hat{\mathcal{L}}(u, \boldsymbol{\phi}, f) \tag{26}$$

para todo $\boldsymbol{\phi} \in W$, $u \in V$ y $f \in L^2(\Omega)$. Esta equivalencia se obtiene del hecho de que A es simétrica, junto con las cotas en (19).

En base a esto, podemos, al igual que antes, obtener su forma bilineal:

$$\hat{\mathcal{F}}(\boldsymbol{\phi}, u; \mathbf{v}, q) = \left(\begin{pmatrix} -A^{-1/2} & A^{1/2}\nabla \\ \nabla^* & X \end{pmatrix}\begin{pmatrix} \boldsymbol{\phi} \\ u \end{pmatrix}, \begin{pmatrix} -A^{-1/2} & A^{1/2} \\ \nabla^* & X \end{pmatrix}\begin{pmatrix} \mathbf{v} \\ q \end{pmatrix}\right) \tag{27}$$

Podemos observar que reescribiendo los productos internos y suponiendo suficiente suavidad en nuestras funciones, podemos reescribir la anterior forma bilineal de la siguiente manera:

$$\hat{\mathcal{F}}(\boldsymbol{\phi}, u; \mathbf{v}, q) = \left(\begin{pmatrix} A^{-1} + \nabla\nabla^* & \nabla(X - I) \\ (X^* - I)\nabla^* & \nabla^* A\nabla + X^* X \end{pmatrix}\begin{pmatrix} \boldsymbol{\phi} \\ u \end{pmatrix}, \begin{pmatrix} \mathbf{v} \\ q \end{pmatrix}\right) \tag{28}$$

Ahora, dado el par $(\boldsymbol{\phi}, u) \in H(\mathrm{div}) \times \mathrm{H}^1$ consideramos la norma en el espacio producto:

Definición 4.2. Dado $\Omega \subset \mathbb{R}^n$ acotado, definimos:

$$\begin{aligned}\|(\boldsymbol{\phi}, u; \boldsymbol{\phi}, u)\|_{H(\mathrm{div})\times \mathrm{H}^1} &= \|(\boldsymbol{\phi}, u)\|_{H(\mathrm{div},\Omega)\times \mathrm{H}^1(\Omega)} \\ &= \left(\|\boldsymbol{\phi}\|^2_{L^2(\Omega)^n} + \|\nabla^* \boldsymbol{\phi}\|^2_{L^2(\Omega)} + \|u\|^2_{L^2(\Omega)} + \|\nabla u\|^2_{L^2(\Omega)^n}\right)^{1/2}\end{aligned}$$

Que está inducida por el producto interno:

$$(\boldsymbol{\phi}, u; \mathbf{v}, q)_{H(\mathrm{div})\times H^1} = (\boldsymbol{\phi}, \mathbf{v})_n + (\nabla^* \boldsymbol{\phi}, \nabla^* \mathbf{v}) + (u, q) + (\nabla u, \nabla q)_n. \tag{29}$$

Consideramos ahora

$$\mathcal{S}(\boldsymbol{\phi}, u; \mathbf{v}, q) = (A^{-1/2}\boldsymbol{\phi}, A^{-1/2}\mathbf{v})_n + (\nabla^* \boldsymbol{\phi}, \nabla^* \mathbf{v}) + (u, q) + (A^{1/2}\nabla u, A^{1/2}\nabla q)_n, \tag{30}$$

que satisface las cotas:

$$\frac{1}{C}\mathcal{S}(\boldsymbol{\phi}, u; \boldsymbol{\phi}, u) \leq (\boldsymbol{\phi}, u; \boldsymbol{\phi}, u)_{H(\mathrm{div})\times H^1} \leq C\mathcal{S}(\boldsymbol{\phi}, u; \boldsymbol{\phi}, u), \tag{31}$$

donde $C = \max\left\{\frac{1}{\lambda}, \Lambda\right\}$. Nuevamente, tomando ciertas libertades respecto a la suavidad y las condiciones de frontera, podemos escribir

$$\hat{\mathcal{S}}(\boldsymbol{\phi}, u; \mathbf{v}, q) = \left(\begin{pmatrix} A^{-1} + \nabla\nabla^* & 0 \\ 0 & I + \nabla^* A \nabla \end{pmatrix} \begin{pmatrix} \boldsymbol{\phi} \\ u \end{pmatrix}, \begin{pmatrix} \mathbf{v} \\ q \end{pmatrix}\right) \tag{32}$$

Observamos la similitud entre los términos diagonales de las matrices que definen las formas bilineales en las ecuaciones (28) y (32). Con eso en mente, uno podría esperar que el resultado que queremos demostrar se pueda interpretar como la demostración de la equivalencia entre estas dos formas bilineales. Por supuesto, todavía no hemos hablado sobre los aspectos relacionados con la suavidad y las condiciones de frontera para garantizar que la equivalencia se mantenga en todo el espacio $W \times V$, por ese motivo haremos varias observaciones antes de demostrar el resultado (que será enunciado formalmente) para poder resolver esta cuestión.

Para comenzar a preparar la demostración, tendremos que analizar ciertas cuestiones:

Observación 3. *Para especificar más sobre la condición que tiene que cumplir el espacio V en nuestro teorema a probar, asumiremos que o $\Gamma_{\mathcal{D}} \neq 0$ o una condición adicional en V de modo que valga la desigualdad de Poincaré (3.16), eso es $\int_\Omega u\, dx = 0$.*
La forma que utilizaremos de esta desigualdad está modificada de la siguiente forma: al utilizar (19), podemos tomar como $\xi = \nabla u$ y usar que A es invertible para poder escribir a (12) como

$$||u||^2_{0,\Omega} \leq C\|\nabla u\|^2_{0,\Omega} \leq \widetilde{C}||A^{1/2}\nabla u||^2_{0,\Omega} \tag{33}$$

Con C la constante de Poincaré y $\widetilde{C} = \frac{C}{\lambda}$, con λ la constante de (19) y $u \in V$.

Observación 4. *Vamos a asumir que para cualquier $f \in H^{-1}(\Omega)$ existe un único u que cumple la ecuación (20). Es decir, que es solución de nuestra ecuación. Para eso, observemos su forma débil dada en (4.1), esto último es equivalente a pedir que exista un único $u \in V$ tal que,*

$$(A\nabla u, \nabla v)_n + (Xu, v) = (f, v) \tag{34}$$

para cada $v \in V$.

Esto es cierto para el caso $X = 0$, $X = I$, o X dado por (2) o (3). La demostración de esto se encuentra en [11] en donde, en pocas palabras, la idea es utilizar el teorema de Lax-Milgram. Como segunda observación nos gustaría saber qué tan regular es nuestra solución u obtenida. Para eso, vemos que siguiendo las ideas de [8, Capitulo 8.3 y 8.4], vemos que en principio podemos asegurar que si $f \in L^2(\Omega)$, entonces para cualquier subdominio $\Omega' \subset\subset \Omega$ obtenemos que $u \in H^2(\Omega')$. Esto no es suficiente dado que no habla sobre la regularidad de u en el borde. Para eso, vamos a suponer como hipótesis cierta regularidad de Ω en el borde, como por ejemplo que el borde sea C^2 o que el dominio sea convexo y con eso obtener que si $f \in L^2(\Omega)$ entonces $u \in H^2(\Omega)$. Con esto en mente, realizamos la siguiente definición:

$$D := \{u \in V \mid A\nabla u \in W\}, \tag{35}$$

Observación 5. *Aunque la suposición de que (20) es invertible lo hace casi implícito, haremos uso explícito de la siguiente cota:*

$$||Xu||_{0,\Omega} \leq \eta ||A^{1/2}\nabla u||_{0,\Omega}, \tag{36}$$

para algún $\eta > 0$ y para todo $u \in D$. En el caso dado por (2), esta cota es inmediata, ya que podemos acotar la divergencia mediante el mismo gradiente, y luego usar el hecho de que la matriz A es invertible para obtener una cota con una constante. El caso de (3) es aún más sencillo, ya que basta aplicar la desigualdad de Poincaré en el segundo término.

Observación 6. *Finalmente, utilizaremos la siguiente desigualdad:*

$$||\nabla^* A\nabla u + Xu||^2_{0,\Omega} \geq \delta ||\nabla^* A\nabla u||^2_{0,\Omega}, \tag{37}$$

para algún $\delta > 0$ y para todo $u \in D$. La validez de esta desigualdad se sigue de los resultados obtenidos en [10], donde se demuestra que si dos operadores uniformemente elípticos son invertibles en $H^1(\Omega)$ y comparten la misma parte principal y las mismas condiciones de frontera, entonces son equivalentes en norma $L^2(\Omega)$, incluso en ausencia de regularidad en $H^2(\Omega)$. Además, dado que $(\nabla^* A\nabla)^{-1}$ es acotado en L^2, existe una constante K tal que,

$$||u||_{0,\Omega} \leq K ||\nabla^* A\nabla u||_{0,\Omega} \tag{38}$$

para cada $u \in D$. Donde esto sale de la definición del sistema.

Ahora, si hacemos integración por partes, utilizamos las cotas sobre la matriz A y el hecho que esta es definida positiva y simétrica y si utilizamos la definición de D para las condiciones de borde, obtenemos:

$$\begin{aligned}
||A^{1/2}\nabla u||^2_{0,\Omega} &= (A^{1/2}\nabla u, A^{1/2}\nabla u) \\
&= (A\nabla u, \nabla u) \\
&= (\nabla^* A\nabla u, u) \ \ (Prop\ Adjunto) \\
&\leq K(\nabla^* A\nabla u, \nabla^* A\nabla u) \ \ (partes\ y\ (38)) \\
&= K||\nabla^* A\nabla u||^2_{0,\Omega}.
\end{aligned}$$

Para todo $u \in D$. Esta última desigualdad junto a (37) obtenemos:

$$||\nabla^* A\nabla u + Xu||^2_{0,\Omega} \geq \gamma(||\nabla^* A\nabla u||^2_{0,\Omega} + ||A^{1/2}\nabla u||^2_{0,\Omega}) \tag{39}$$

para todo $u \in D$ y $\gamma = \lambda/(K+1)$.

Finalmente estamos en condiciones de probar el teorema. Recordemos el enunciado:

Teorema 4.3. *Asumiendo que V cumple con la observación 33, X satisface la cota (36) y que nuestro sistema (4.1) tiene solución. Entonces existen constantes positivas α y β tales que*

$$\mathcal{F}(\boldsymbol{\phi}, u; \boldsymbol{v}, q) \leq \beta(||\boldsymbol{\phi}||^2_{H(div)} + ||u||^2_{1,\Omega})^{1/2}(||\boldsymbol{v}||^2_{H(div)} + ||q||^2_{1,\Omega})^{1/2} \tag{40}$$

para cada $\boldsymbol{\phi}, \boldsymbol{v} \in W$ y cada $u, q \in V$ y además,

$$\mathcal{F}(\boldsymbol{\phi}, u; \boldsymbol{\phi}, u) \geq \alpha(||\boldsymbol{\phi}||^2_{H(div)} + ||u||^2_{1,\Omega}) \tag{41}$$

para cada $\boldsymbol{\phi} \in W$ y $u \in V$.

Demostración. La continuidad de $\mathcal{F}$ en (40) se sigue directamente de la suposición (19), de las definiciones de las normas y la definición de $\mathcal{F}$, y de (36).

$$\begin{aligned}\mathcal{F}(\boldsymbol{\phi}, u; \mathbf{v}, q) &= \int (-\boldsymbol{\phi} + A\nabla u)(-\mathbf{v} + A\nabla q) + (\nabla^*\boldsymbol{\phi} + Xu)(\nabla^*\mathbf{v} + Xq)dx \\ &= \int \boldsymbol{\phi}\mathbf{v} - \boldsymbol{\phi}A\nabla q - \mathbf{v}A\nabla u + (A\nabla u)\cdot(A\nabla q) \\ &+ (\nabla^*\boldsymbol{\phi})\cdot(\nabla^*\mathbf{v}) + Xu\nabla^*\mathbf{v} + \nabla^*\boldsymbol{\phi}Xu + (Xu)(Xq).\end{aligned}$$

Ahora sí, en cada término lo acotamos por su multiplicación entre normas y cuando aparezca la norma de la matriz A la acotamos utilizando las correspondientes cotas; obtenemos lo que queremos.

La demostración de la cota inferior en (41) se establece a través de una serie de pasos.

1 Primero, formulamos un problema equivalente: Por (26) y (31), es suficiente encontrar una constante positiva α_0 tal que:

$$\alpha_0 \mathcal{S}(\boldsymbol{\phi}, u; \boldsymbol{\phi}, u) \leq \hat{\mathcal{F}}(\boldsymbol{\phi}, u; \boldsymbol{\phi}, u) \tag{42}$$

para todo $\boldsymbol{\phi} \in W$ y $u \in V$. Entonces, (41) seguiría con $\alpha = \alpha_0\lambda \min\{\lambda, 1/\Lambda\}$.

2 A continuación, utilizando ideas análogas al Capítulo 3 de [9], vamos a descomponer el vector $\boldsymbol{\phi}$ en dos funciones, una con divergencia nula (esto es lo que comúnmente se llama *"divergence free"*) : Para cualquier $\boldsymbol{\phi} \in W$:

$$\boldsymbol{\phi} = A\nabla p + \boldsymbol{\psi}, \tag{43}$$

donde $p \in D$,

$$\nabla^* \boldsymbol{\psi} = 0, \tag{44}$$

$$\mathbf{n} \cdot \boldsymbol{\psi} = 0 \text{ en } \Gamma_{\mathcal{N}} \tag{45}$$

Esto se logra eligiendo a p como la solución débil de

$$\begin{cases} \nabla^* A \nabla p = \nabla^* \boldsymbol{\phi}, \\ p = 0 \ \text{ en } \Gamma_{\mathcal{D}}, \\ \mathbf{n} \cdot A \nabla p = 0 \ \text{ en } \Gamma_{\mathcal{N}}. \end{cases}$$

De la ecuación de una observación anterior (34) se sigue que $p \in D$, por lo tanto, $A\nabla p \in W$. Al establecer $\boldsymbol{\psi} = \boldsymbol{\phi} - A\nabla p$, entonces $\boldsymbol{\psi} \in W$ y las ecuaciones (43) - (45) se cumplen.

Las ecuaciones (22), (29) y (30) nos dan:

$$\mathcal{S}(\boldsymbol{\psi}, 0; \boldsymbol{\psi}, 0) = \hat{\mathcal{F}}(\boldsymbol{\psi}, 0; \boldsymbol{\psi}, 0) = (A^{-1/2}\boldsymbol{\psi}, A^{-1/2}\boldsymbol{\psi}). \tag{46}$$

Consideremos los términos de productos cruzados. Dado que $\boldsymbol{\psi} \in W$ y $u, p \in V$, podemos integrar por partes para obtener:

$$\mathcal{S}(A\nabla p, u; \boldsymbol{\psi}, 0) = (\nabla p, \boldsymbol{\psi})_n = (u, \nabla^* \boldsymbol{\psi}) \underbrace{=}_{(44)} 0, \tag{47}$$

$$\hat{\mathcal{F}}(A\nabla p, u; \boldsymbol{\psi}, 0) = (\nabla u - \nabla p, \boldsymbol{\psi})_n = (u - p, \nabla^* \boldsymbol{\psi}) \underbrace{=}_{(44)} 0. \tag{48}$$

Esto implica:

$$\mathcal{S}(A\nabla p + \boldsymbol{\psi}, u; A\nabla p + \boldsymbol{\psi}, u) = \mathcal{S}(A\nabla p, u; A\nabla p, u) + \mathcal{S}(\boldsymbol{\psi}, 0; \boldsymbol{\psi}, 0).$$
$$\hat{\mathcal{F}}(A\nabla p + \boldsymbol{\psi}, u; A\nabla p + \boldsymbol{\psi}, u) = \hat{\mathcal{F}}(A\nabla p, u; A\nabla p, u) + \hat{\mathcal{F}}(\boldsymbol{\psi}, 0; \boldsymbol{\psi}, 0).$$

Así, solo queda mostrar que existe una constante positiva $\alpha_0 \leq 1$ tal que:

$$\alpha_0 \mathcal{S}(A\nabla p, u; A\nabla p, u) \leq \hat{\mathcal{F}}(A\nabla p, u; A\nabla p, u). \tag{49}$$

Para todo $p \in D$ y para todo $u \in V$.

3 Ahora definiremos unos operadores $\mathcal{S}_0$ y $\mathcal{F}_0$ y veamos que $c\mathcal{S}_0 \leq \mathcal{F}_0$.

Sea

$$\mathcal{F}_0(p, u; \varphi, q) = \hat{\mathcal{F}}(A\nabla p, u; A\nabla \varphi, q) \tag{50}$$

$$= \left(\begin{pmatrix} -A^{1/2}\nabla & A^{1/2}\nabla \\ \nabla^* A \nabla & X \end{pmatrix} \begin{pmatrix} p \\ u \end{pmatrix}, \begin{pmatrix} -A^{1/2}\nabla & A^{1/2}\nabla \\ \nabla^* A \nabla & X \end{pmatrix} \begin{pmatrix} \varphi \\ q \end{pmatrix} \right)_{n+1}. \tag{51}$$

$$\begin{aligned}\mathcal{S}_0(p,u,\varphi,q) &= (\nabla^* A\nabla p, \nabla^* A\nabla\varphi)_n + (A^{1/2}\nabla u, A^{1/2}\nabla q)_n \\ &= \left(\begin{pmatrix} 0 & A^{1/2}\nabla \\ \nabla^* A\nabla & 0 \end{pmatrix}\begin{pmatrix} p \\ u \end{pmatrix}, \begin{pmatrix} 0 & A^{1/2}\nabla \\ \nabla^* A\nabla & 0 \end{pmatrix}\begin{pmatrix} \varphi \\ q \end{pmatrix}\right)_{n+1}.\end{aligned}$$

Ahora veamos que existe esa constante para todo $p \in D$ y $u \in V$.

Primero escribamos

$$\begin{pmatrix} p \\ u \end{pmatrix} = \begin{pmatrix} z \\ z \end{pmatrix} + \begin{pmatrix} w_1 \\ w_2 \end{pmatrix} \tag{52}$$

Donde $z, w_1 \in D$ y $w_2 \in V$ tal que

$$\mathcal{S}_0(z,z;w_1,w_2) = 0. \tag{53}$$

y

$$\nabla^* A\nabla w_1 = -w_2 \tag{54}$$

Esto se logra eligiendo $z, w_1 \in D$ tales que cumplan:

$$\nabla^* A\nabla z + z = \nabla^* A\nabla p + u \tag{55}$$

$$\nabla^* A\nabla w_1 + w_1 = p - u \tag{56}$$

Para ver esto, notar que al sumar las dos ecuaciones anteriores, obtenemos

$$\begin{aligned}\nabla^* A\nabla(z+w_1) + (z+w_1) &= \nabla^* A\nabla p + p \\ \nabla^* A\nabla(z+w_1-p) &= -((z+w_1)-p)\end{aligned}$$

La solución a esta ecuación tiene única solución y la solución $(z+w_1-p) = 0$ cumple la ecuación (si suponemos que tenemos condiciones de borde homogéneas), por lo que

$$p = z + w_1 \tag{57}$$

Como $D \subset V$, si definimos a w_2 de la siguiente forma:

$$w_2 = u - z \tag{58}$$

cumple que $w_2 \in V$, pues $u, z \in V$ y se cumple (52). Ahora sustituyendo (57) en (56) y usando (58) prueba (54), que también cumple (53).

Ahora, como $z \in D$, podemos usar la cota (39) y obtener

$$\begin{aligned}\mathcal{F}_0(z,z;z,z) &= (\nabla^* A\nabla z + Xz, \nabla^* A\nabla z + Xz) = ||\nabla^* A\nabla z + Xz||^2_{0,\Omega} \\ &\geq \gamma(||\nabla^* A\nabla z||^2_{0,\Omega} + ||A^{1/2}\nabla z||^2_{0,\Omega}) = \gamma\mathcal{S}_0(z,z;z,z).\end{aligned}$$

Notar además que,

$$\mathcal{F}_0(w_1, w_2; w_1, w_2) = ||A^{1/2}\nabla w_2 - A^{1/2}\nabla w_1||^2_{0,\Omega,n} + ||\nabla^* A\nabla w_1 + Xw_2||^2_{0,\Omega} \tag{59}$$

Esto sale de usar la fórmula (51) en los puntos y distribuir. Usando (58) en el primer término de esta última ecuación, obtenemos que:

$$\begin{aligned}
||A^{1/2}\nabla w_2 - A^{1/2}\nabla w_1||^2_{0,\Omega,n} &= (A\nabla w_1, \nabla w_1)_n - 2(A\nabla w_1, \nabla w_2)_n + (A\nabla w_2, \nabla w_2)_n \\
&= (A\nabla w_1, \nabla w_1)_n - 2(\nabla^* A\nabla w_1, w_2) + (A\nabla w_2, \nabla w_2)_n \\
&= (A\nabla w_1, \nabla w_1)_n + 2(\nabla^* A\nabla w_1, \nabla^* A\nabla w_1) + (A\nabla w_2, \nabla w_2)_n \\
&\geq (\nabla^* A\nabla w_1, \nabla^* A\nabla w_1) + (A\nabla w_2, \nabla w_2)_n \\
&= \mathcal{S}_0(w_1, w_2; w_1, w_2).
\end{aligned}$$

Donde utilizamos (58) en la tercera igualdad y la acotación fue posible dado que el primer término es positivo. También fue posible utilizar la integración por partes en el segundo igual puesto que $w_1 \in D$ y $w_2 \in V$.
Usando la cota (36) y asumiendo que $\eta \geq 1$, el segundo término de (59) cumple,

$$\begin{aligned}
||\nabla^* A\nabla w_1 + Xw_2||^2_{0,\Omega} &\leq 2(||\nabla^* A\nabla w_1||^2_{0,\Omega} + ||Xw_2||^2_{0,\Omega}) \\
&\leq 2(||\nabla^* A\nabla w_1||^2_{0,\Omega} + \eta^2||A^{1/2}\nabla w_2||^2_{0,\Omega,n}) \\
&\leq 2\eta^2\mathcal{S}_0(w_1, w_2; w_1, w_2).
\end{aligned}$$

Ahora notemos que si en (51) evaluamos en $(p, u; p, u)$ y usamos la elección de estos en (52), podemos escribir a $\mathcal{F}_0$ como:

$$\begin{aligned}
\mathcal{F}_0(p, u; p, u) &= ||A^{1/2}\nabla w_2 - A^{1/2}\nabla w_1||^2_{0,\Omega,n} + ||\nabla^* A\nabla w_1 + Xw_2||^2_{0,\Omega} \\
&+ 2(\nabla^* A\nabla w_1 + Xw_2, \nabla^* A\nabla z + Xz) + ||\nabla^* A\nabla z + Xz||^2_{0,\Omega} \\
&\geq ||A^{1/2}\nabla w_2 - A^{1/2}\nabla w_1||^2_{0,\Omega,n} + ||\nabla^* A\nabla w_1 + Xw_2||^2_{0,\Omega} \\
&- 2||\nabla^* A\nabla w_1 + Xw_2||_{0,\Omega}||\nabla^* A\nabla z + Xz||_{0,\Omega} + ||\nabla^* A\nabla z + Xz||^2_{0,\Omega} \\
&\geq ||A^{1/2}\nabla w_2 - A^{1/2}\nabla w_1||^2_{0,\Omega,n} + (1 - \frac{1}{\epsilon})||\nabla^* A\nabla w_1 + Xw_2||^2_{0,\Omega} \\
&+ (1 - \epsilon)||\nabla^* A\nabla z + Xz||^2_{0,\Omega}
\end{aligned}$$

Donde en la primera desigualdad utilizamos la desigualdad de Cauchy-Schwarz. En la tercer igualdad utilizamos la siguiente desigualdad:

$$\begin{aligned}
x^2 - 2xy + y^2 = (x - y)^2 > 0 \\
-2xy \geq -x^2 - y^2
\end{aligned}$$

Si escribimos $x = \sqrt{\epsilon}x$ e $y = \frac{1}{\sqrt{\epsilon}}y$ en la desigualdad anterior tenemos la cota.
Esto vale para todo $\epsilon > 0$. Considerando únicamente el caso donde $\epsilon < 1$ para así poder acotar el término de $(1 - \frac{1}{\epsilon} < 0$ y usando las cotas que obtuvimos antes para el primer y segundo término de $\mathcal{F}_0$ obtenemos,

$$\mathcal{F}_0(p, u; p, u) \geq (1 + (1 - \frac{1}{\epsilon})2\eta^2)\mathcal{S}_0(w_1, w_2; w_1, w_2) + (1 - \epsilon)\gamma\mathcal{S}_0(z, z; z, z).$$

Ahora la idea sería juntar las constantes para llegar a lo que queremos, para eso, tomemos

$$\epsilon = \frac{\sqrt{(1+2\eta^2-\gamma)^2+8\eta^2\gamma}-(1+2\eta^2-\gamma)}{2\gamma}$$

De esta forma $(1+(1-\frac{1}{\epsilon})2\eta^2) = (1-\epsilon)\gamma$.
Ahora como sabemos que $\mathcal{S}_0(z,z;w_1,w_2) = 0$ (Por construcción), cuando escribamos $\mathcal{S}_0(p,u;p,u)$ los términos cruzados se cancelaran y así, obtenemos:

$$\mathcal{F}_0(p,u;p,u) \geq c(\mathcal{S}_0(w_1,w_2;w_1,w_2)+\mathcal{S}_0(z,z;z,z)) = c\mathcal{S}_0(p,u;p,u)$$

Con

$$c = \frac{(1+2\eta^2+\gamma)-\sqrt{(1+2\eta^2+\gamma)^2-4\gamma}}{2} > 0,$$

Que es exactamente lo que queríamos ver.

4 Mostraremos finalmente que $\alpha\mathcal{S} \leq \hat{\mathcal{F}}$. Hasta ahora tenemos que,

$$\mathcal{S}_0(p,u;p,u) = ||\nabla^* A\nabla p||^2_{0,\Omega} + ||A^{1/2}\nabla u||^2_{0,\Omega,n} \leq \frac{1}{c}\mathcal{F}_0(p,u;p,u). \tag{60}$$

Utilizando esta última desigualdad y la definición de $\mathcal{F}_0$ obtenemos:

$$\begin{aligned} ||A^{1/2}\nabla p||^2_{0,\Omega,n} &\leq 2(||A^{1/2}\nabla u||^2_{0,\Omega,n} + ||A^{1/2}\nabla u - A^{1/2}\nabla p||^2_{0,\Omega,n}) \\ &\leq 2(\frac{1}{c}+1)\mathcal{F}_0(p,u;p,u). \end{aligned}$$

Observar que la primera desigualdad se obtiene de a $||A^{1/2}\nabla p||^2_{0,\Omega,n}$ sumar y restarle $A^{1/2}\nabla u$ luego usar la desigualdad triangular y acotar por 2 veces la suma de las normas al cuadrado.
Usando la desigualdad (33), la definición de $\mathcal{F}_0$ y las dos desigualdades anteriores nos dejan:

$$\begin{aligned} \mathcal{S}(A\nabla p,u;A\nabla p,u) &= ||A^{1/2}\nabla p||^2_{0,\Omega,n} + ||\nabla^* A\nabla p||^2_{0,\Omega} + ||u||^2_{0,\Omega} + ||A^{1/2}\nabla u||^2_{0,\Omega,n} \\ &\leq (\frac{3+2c+\widetilde{C}}{c})\hat{\mathcal{F}}(A\nabla p,u;A\nabla p,u) \end{aligned}$$

Luego obtenemos nuestro resultado tomando $\alpha_0 = \frac{c}{3+2c+\widetilde{C}}$.

□

5 Aproximación de soluciones mediante redes neuronales

En esta sección presentamos un método para la aproximación de soluciones a ecuaciones elípticas basado en Deep Learning, desarrollado en [2] y probamos un resultado de convergencia. El método se basa en la aproximación de la solución al problema de cuadrados mínimos presentado en la sección anterior mediante redes neuronales. Las ideas básicas son muy sencillas. El aporte fundamental de [2] es proponer una configuración adecuada para que con nuestro método generado, obtengamos convergencia al mínimo del problema continuo. En trabajos anteriores, las condiciones de contorno se incorporaban de manera laxa, a través de términos de penalidad en el funcional a minimizar. De este modo, las soluciones halladas sólo cumplían estas condiciones de manera aproximada. En el método que estudiamos aquí, construimos la solución combinando la función distancia al borde con la red neuronal a entrenar. De este modo, podemos garantizar que la solución satisface de manera exacta las condiciones de contorno.

Es importante remarcar que en dimensiones bajas (1, 2 y 3) estos métodos basados en Machine Learning aún resultan perceptiblemente más ineficientes e inexactos que los métodos clásicos de elementos finitos, basados en un adecuado mallado del dominio y de la frontera. Estos métodos aprovechan mejor el rico conocimiento que tenemos sobre los dominios, sobre los espacios funcionales y sobre las características de la solución y llevan décadas de perfeccionamiento y optimización. Es en dimensión más alta, donde el uso de métodos basados en el mallado de dominio resulta impracticable, que la aproximación por redes neuronales cobra más sentido práctico.

5.1. Formulación del problema

Recordemos que tenemos el funcional:

$$\mathcal{L}(u, \boldsymbol{\phi}, f) = \|A\nabla u - \boldsymbol{\phi}\|^2_{L^2(\Omega)} + \| - \nabla^* \boldsymbol{\phi} - Bu + f\|^2_{L^2(\Omega)}$$

Donde tomamos un caso particular del operador X, llamado $B : H^1(\Omega) \to L^2(\Omega)$ que cumple:

$$\|Bv\|_{L^2(\Omega)} \leq \|v\|_{L^2(\Omega)} \quad \forall v \in H^1(\Omega) \text{ tal que } v = 0 \text{ en } \Gamma_{\mathcal{D}}.$$

Además, el operador $\mathcal{L}$ proviene de (20) en el espacio adecuado:

$$\mathcal{A} := \left\{ q = (u, \boldsymbol{\phi}) \in H^1(\Omega) \times H(\mathrm{div}, \Omega) : u = g_{\mathcal{D}} \text{ en } \Gamma_{\mathcal{D}}, \boldsymbol{\phi} \cdot \mathbf{n} = g_{\mathcal{N}} \text{ en } \Gamma_{\mathcal{N}} \right\}$$

Es claro que, si la ecuación (20) tiene una solución única $u \in H^1(\Omega)$, entonces el minimizador único de $\mathcal{L}$ en $\mathcal{A}$ es $\mathbf{q} := (u, A\nabla u)$. Nuestro objetivo es calcular aproximaciones de dicho minimizador dentro de un espacio adecuado $A_m \subset A$. En particular, en el método propuesto consideramos un espacio A_m compuesto por redes neuronales con una arquitectura y parámetros Θ en $\mathbb{R}^m$.

El objetivo del método propuesto es aproximar al único minimizador $(u, \boldsymbol{\phi})$ del funcional $\mathcal{L}$. Un primer acercamiento sería buscar un conjunto de parámetros $\Theta_0 \in \mathbb{R}^m$ tal que

$\mathcal{L}(u_{\Theta_0}, \boldsymbol{\phi}_{\boldsymbol{\Theta}_0}) = \min_{\Theta \in \mathbb{R}^m} \mathcal{L}(u_\Theta, \boldsymbol{\phi}_{\boldsymbol{\Theta}})$. Con las funciones $(u_{\Theta_0}, \boldsymbol{\phi}_{\boldsymbol{\Theta}_0})$ pertenecientes a un espacio adecuado.
El uso de redes neuronales en este contexto tiene la ventaja de que se pueden implementar fácilmente métodos sin malla mediante el muestreo aleatorio de puntos, lo que permite abordar problemas en espacios de alta dimensión, donde la mayoría de los métodos numéricos clásicos para EDP se vuelven inviables.

La imposición de condiciones de contorno es un aspecto no trivial a tener en cuenta en este enfoque. Una manera típica de abordar este problema es incorporar las condiciones de contorno añadiendo un término de penalización; en el marco de nuestro método, estas se consideran condiciones de tipo **débil**. Sin embargo, en la práctica, se ha observado que hacer que las funciones discretas cumplan estrictamente las condiciones de contorno acelera el proceso de entrenamiento. En primer lugar, crearemos funciones auxiliares adecuadas con el objetivo de imponer las condiciones de contorno de forma **fuerte**.
En líneas generales, el procedimiento sería el siguiente, primero nos aseguramos $(u_\Theta, \boldsymbol{\phi}_{\boldsymbol{\Theta}}) \in \mathcal{A}_m \subset A$ para todo $\Theta \in \mathbb{R}^m$, después sampleamos N puntos $\{x_k\}_{k=1}^N \subset \Omega$ uniformemente y aproximamos $\mathcal{L}(u_\Theta, \boldsymbol{\phi}_{\boldsymbol{\Theta}}) \approx \mathcal{L}_N(u_\Theta, \boldsymbol{\phi}_{\boldsymbol{\Theta}})$ en cada paso del algoritmo de descenso por gradiente, con $\mathcal{L}_N$ definido como:

$$\mathcal{L}_N(u, \boldsymbol{\phi}) := \frac{|\Omega|}{N} \sum_{k=1}^{N} (\boldsymbol{\phi}(x_k) - A\nabla u(x_k))^2 + (\mathrm{div}\boldsymbol{\phi}(x_k) - Bu(x_k) + f(x_k))^2 \tag{61}$$

5.2. Condiciones de contorno

Como dijimos brevemente antes, hay varias formas de imponer condiciones de borde a la solución de la ecuación. Estas pueden ser impuestas de forma débil o fuerte.

Por un lado, las condiciones de borde **débiles** se imponen indirectamente a través de la formulación débil del problema. En lugar de exigir que la solución cumpla las condiciones de contorno de manera exacta, estas se incorporan en la integral que define el problema en su forma débil. Esta forma es bastante versátil y, en particular, permite incorporar las condiciones de borde tipo Neumann de manera bastante sencilla. Su principal desventaja es que el optimizador difícilmente logre anular el término de penalidad, lo que equivale a obtener soluciones que sólo cumplirán la condición de contorno de forma aproximada.

Por otro lado, tenemos las condiciones de borde **fuertes** que su imposición es directa de las condiciones de contorno en la formulación del problema. Es decir, la solución aproximada debe satisfacer exactamente estas condiciones en los puntos correspondientes del dominio. La ventaja de esto viene de su misma definición; al definirlo de esta forma, obtenemos que las condiciones de contorno se respeten estrictamente. Pero puede ser complicado, especialmente en geometrías complejas o en problemas donde las condiciones de borde no son fácilmente manejables. Nosotros intentaremos imponerlas de forma **fuerte**, para eso precisamos unas definiciones.

Definición 5.1 (*función distancia suave*)**.** Sea $\Gamma_* \subset \overline{\Omega}$ un conjunto cerrado. Decimos que la función lipschitz continua $d_* : \Omega \to \mathbb{R}$ es una función distancia suave si $d_* \geq 0$ y $d_*(x) = 0$ si y solo si $x \in \Gamma_*$.

En general, el estándar es pedir que la distancia al borde se comporte como la distancia usual cerca del borde. Sin embargo, es necesario tomar esta definición, dado que más adelante, veremos que la distancia me influye en lo que respecta a la regularidad de mis funciones a aproximar continuas. De todas formas, esto se verá de mejor forma en (67).

Ahora, a partir de esta definición podemos imponer condición fuerte sobre el borde Dirichlet y Neumann de la siguiente forma:

Cuando calculamos la solución u de la ecuación diferencial (20), vamos a restringir a las funciones del tipo,

$$u(x) := G_{\mathcal{D}}(x) + d_{\mathcal{D}}(x)v(x). \tag{62}$$

donde la función desconocida es $v : \Omega \to \mathbb{R}$ y $d_{\mathcal{D}}$ es la función distancia suave de $\Gamma_{\mathcal{D}}$. $G_{\mathcal{D}}$ es una extensión del dato de borde sobre $\Gamma_{\mathcal{D}}$ a todo el dominio. Es decir que $G_{\mathcal{D}} : \Omega \to \mathbb{R}$ es tal que $G_{\mathcal{D}}|_{\Gamma_{\mathcal{D}}} = g_{\mathcal{D}}$.

De forma similar, intentaremos hacer lo mismo sobre la variable $\boldsymbol{\phi}$. Primero construimos un vector $\boldsymbol{\nu} : \Omega \to \mathbb{R}^d$ tal que $\boldsymbol{\nu}|_{\Gamma_{\mathcal{N}}} = \mathbf{n}$ y $|\boldsymbol{\nu}(x)| = 1$ en casi todo punto. Luego para $x \in \Omega$ consideramos:

$$\boldsymbol{\phi}(x) = \boldsymbol{\psi}(x) + \left(G_{\mathcal{N}} - \frac{\boldsymbol{\psi}(x) \cdot \boldsymbol{\nu}(x)}{1 + d_{\mathcal{N}}(x)}\right) \boldsymbol{\nu}(x). \tag{63}$$

Al igual que antes $G_{\mathcal{N}}$ es una extensión a Ω del dato $g_{\mathcal{N}}$, $d_{\mathcal{N}}$ es la función distancia suave al dominio Neumann y la función desconocida es $\psi : \Omega \to \mathbb{R}^d$.
Notar que esta definición de $\boldsymbol{\phi}$ cumple con el dato de borde, dado que si restringimos $x \in \Gamma_{\mathcal{N}}$, entonces $\boldsymbol{\nu}(x) = \mathbf{n}(x)$, entonces:

$$\begin{aligned}
\boldsymbol{\phi}(x) \cdot \mathbf{n}(x) &= \boldsymbol{\psi}(\boldsymbol{x}) + \left(g_{\mathcal{N}}(x) - \frac{\boldsymbol{\psi}(x) \cdot \mathbf{n}(x)}{1 + d_{\mathcal{N}}(x)} \mathbf{n}\right) \\
&= \boldsymbol{\psi}(\boldsymbol{x}) + \left(g_{\mathcal{N}}(x) - \boldsymbol{\psi}(\boldsymbol{x}) |\mathbf{n}(x)|^2\right) \\
&= g_{\mathcal{N}}(x).
\end{aligned}$$

Luego cumple la condición de borde como queríamos.

Entonces para construir la aproximación de las soluciones, tendríamos que primero calcular el vector $\boldsymbol{\nu}$ y luego las funciones escalares $d_{\mathcal{D}}, d_{\mathcal{N}}, G_{\mathcal{D}}, G_{\mathcal{N}}$. Y una vez obtenido eso, buscaríamos $y = (v, \boldsymbol{\psi})$ tal que el par correspondiente $(u, \boldsymbol{\phi})$ minimice la función de pérdida $\mathcal{L}$.

Hay diversas formas de obtener estas funciones de distancias, abordaremos este tema más adelante, por el momento supongamos que realizamos nuestro algoritmo y las tenemos calculadas.

5.3. Análisis del método

En esta sección demostraremos la convergencia del método. Para ello, introduciremos el concepto de discretización en el contexto de la Γ-convergencia como vemos en [4]. Este enfoque nos permitirá analizar la convergencia de una secuencia de funcionales discretos hacia el funcional continuo original. Específicamente, consideraremos la secuencia de funcionales (61), que se asocia con métodos sin malla para aproximar una versión regularizada del funcional de pérdida discreto $\mathbb{R}^m \ni \Theta \mapsto \mathcal{L}(u_\Theta, \boldsymbol{\phi}_\Theta)$.

En términos generales, la teoría de Γ-convergencia garantiza que, bajo ciertas condiciones, los minimizadores de los funcionales discretos convergen a un minimizador del funcional continuo a medida que el tamaño de discretización se refina. De esta forma, podemos asegurar que el método propuesto no solo es consistente, sino que también proporciona una solución aproximada que se acerca a la solución exacta del problema continuo conforme se incrementa la precisión de la discretización. Abordaremos brevemente este concepto más adelante.

Por otro lado, vamos a aprovechar la coercitividad del funcional (que demostramos anteriormente) y las propiedades de aproximación de las redes neuronales para concluir que la secuencia de minimizadores de la función de pérdida discreta regularizada tiende a la solución (4.1) si el número de parámetros de la red m tiende a infinito.

Para simplificar el problema, consideramos $g_\mathcal{D} = g_\mathcal{N} = 0$. Puesto que si no, podríamos considerar $G_\mathcal{D}$ y $G_\mathcal{N}$ tal que $G_\mathcal{D} = g_\mathcal{D}$ en $\Gamma_\mathcal{D}$ y $G_\mathcal{N} = g_\mathcal{N}$ en $\Gamma_\mathcal{N}$, un campo normal suave $\boldsymbol{\nu}$ tal que $\boldsymbol{\nu} = \mathbf{n}$ en $\Gamma_\mathcal{N}$ y las funciones auxiliares $u_0 = u - G_\mathcal{D}$ y $\boldsymbol{\phi}_0 = \boldsymbol{\phi} - G_\mathcal{N}\boldsymbol{\nu}$ y el sistema a resolver quedaría:

$$\begin{cases} \boldsymbol{\phi}_0 - A\nabla u_0 & = A\nabla G_\mathcal{D} - G_\mathcal{N}\boldsymbol{\nu} & \text{en } \Omega \\ -\mathrm{div}(\boldsymbol{\phi}_0) + Bu_0 & = f + \mathrm{div}(G_\mathcal{N}\boldsymbol{\nu}) - BG_\mathcal{D} & \text{en } \Omega \\ u_0 & = 0 & \text{en } \Gamma_\mathcal{D} \\ \boldsymbol{\phi}_0 \cdot \boldsymbol{\nu} & = 0 & \text{en } \Gamma_\mathcal{N} \end{cases}$$

Al igual que antes, la solución a este sistema está determinada por el mínimo del funcional de mínimos cuadrados:

$$(u, \boldsymbol{\phi}) \to ||\boldsymbol{\phi} - A\nabla u + \widetilde{g}||^2_{L^2(\Omega)} + ||\mathrm{div}(\boldsymbol{\phi}) - Bu + \widetilde{f}||^2_{L^2(\Omega)},$$

Con $\widetilde{g} = -A\nabla G_\mathcal{D} + G_\mathcal{N}\boldsymbol{\nu}$ y $\widetilde{f} = f + \mathrm{div}(G_\mathcal{N}\boldsymbol{\nu}) - BG_\mathcal{D}$.
Este funcional lo podemos trabajar con las mismas herramientas que trabajamos a (21) con la única diferencia que en la primera norma tenemos a $\widetilde{g}$ un término de corrección de orden cero.

A continuación de la demostración nos limitaremos, sin pérdida de generalidad, a una red con solo una capa oculta con n neuronas. Definimos el conjunto de funciones discretas como,

$$\mathcal{C}_m := \{(v_\Theta, \boldsymbol{\psi}_\Theta) : v_\Theta = B_v\sigma(A_v x + c_v), \boldsymbol{\psi}_\Theta = B_{\boldsymbol{\psi}}\sigma(A_{\boldsymbol{\psi}} x + c_{\boldsymbol{\psi}})\} \tag{64}$$

Con $A_v, A_{\boldsymbol{\psi}} \in \mathbb{R}^{n\times d}$, $c_v, c_{\boldsymbol{\psi}} \in \mathbb{R}^{n\times 1}$, $B_v \in \mathbb{R}^{1\times n}$, $B_{\boldsymbol{\psi}} \in \mathbb{R}^{d\times n}$ y $\sigma : \mathbb{R}^n \to \mathbb{R}^n$, donde σ es una función de activación suave, acotada y no constante, que se aplica elemento a elemento.

Vamos a recolectar la información de todos los parámetros en $\Theta \in \mathbb{R}^m$ con $m = 3n(d+1)$. Notar que cada vez que aparezca $m \to \infty$ implica que la cantidad de neuronas n tiende a infinito.

Asumiendo que podemos efectivamente encontrar las funciones distancia y normal, mencionadas anteriormente, podemos definir el espacio discreto correspondiente:

$$\mathcal{A}_m := \left\{ \mathbf{q}_\Theta = (u_\Theta, \boldsymbol{\phi}_\Theta) : u_\Theta = d_{\mathcal{D}} v_\Theta \text{ y } \boldsymbol{\phi}_\Theta = \boldsymbol{\psi}_\Theta - \left(\frac{\boldsymbol{\psi}_\Theta \cdot \boldsymbol{\nu}}{1 + d_{\mathcal{N}}} \right) \boldsymbol{\nu}, (v_\Theta, \boldsymbol{\psi}_\Theta) \in \mathcal{C}_m \right\}. \tag{65}$$

Observar que en efecto se cumple la condición de borde en el conjunto $\mathcal{A}_m$. Dado que $u_\Theta = 0$ si $d_{\mathcal{D}} = 0$ y de la misma forma $\boldsymbol{\phi}_\Theta \cdot \boldsymbol{\nu} = 0$ si $d_{\mathcal{N}} = 0$.

Observación 7. *Un detalle importante es que al utilizar integración de Monte Carlo, no se puede evaluar puntualmente una función arbitraria $f \in L^2(\Omega)$. Sin embargo, por densidad, para todo $\epsilon > 0$ podemos encontrar una función continua f_ϵ tal que $\|f - f_\epsilon\|_{L^2(\Omega)} < \epsilon$. Gracias a la elipticidad del funcional $\mathcal{L}$, como vimos antes, en la norma $H^1(\Omega) \times H(div; \Omega)$, si consideramos el funcional $\mathcal{L}_\epsilon$ en lugar de $\mathcal{L}$ usando f_ϵ en lugar de f, y $q_{0,\epsilon}$ como su minimizador, entonces obtenemos que $\|q_0 - q_{0,\epsilon}\|_{H^1(\Omega)\times H(div;\Omega)} < \epsilon$. Por lo tanto, podemos implementar el método utilizando f_ϵ en lugar de f y haciendo que $\epsilon \to 0$ conforme $m \to \infty$.*

5.4. Propiedades de aproximación de una red

La idea de esta parte es juntar las ideas de densidad mencionadas en la Sección 3.4 con nuestro problema en una forma más concreta. Para eso, el primer acercamiento es definir $\mathbf{q}_0 = (u_0, \boldsymbol{\phi}_0) \in \mathcal{A}$ el único minimizador de (21) y vamos a asumir que $\mathbf{q}_0$ puede ser aproximado por un espacio de redes neuronales. Es decir,

$$d(\mathbf{q}_0, \mathcal{A}_m) := \inf_{\mathbf{q}_\Theta \in \mathcal{A}_m} ||\mathbf{q}_0 - \mathbf{q}_\Theta||_{H^1(\Omega)\times H(\text{div};\Omega)} \to 0 \quad \text{cuando } m \to \infty. \tag{66}$$

Una primera pregunta que uno podría hacerse es por qué tiene sentido esta hipótesis. En primer lugar, existen varios resultados clásicos en la literatura que estudian las propiedades de aproximación de redes neuronales, aunque generalmente sin considerar condiciones de borde. En particular, se ha demostrado que redes profundas con función de activación ReLU (con a lo sumo $\log_2(d+1)$ capas ocultas) pueden aproximar funciones a trozos lineales definidas sobre triangulaciones del dominio, similares a las que se utilizan en el método de los elementos finitos. Estas funciones poseen buenas propiedades de aproximación en la norma H^1.

Por lo tanto, si se utiliza una función de activación no constante, se puede garantizar que $d(\mathbf{q}, \mathcal{C}_m) \to 0$ cuando $m \to \infty$, para cualquier $\mathbf{q} \in H^1(\Omega) \times H(\text{div}; \Omega)$.

Además, la hipótesis (66) considera específicamente que podemos aproximar a $\mathbf{q}_0$ mediante los espacios $\mathcal{A}_m$, que sí incorporan las condiciones de contorno. Esta suposición es razonable, especialmente si asumimos cierta regularidad en la solución de (20). Por ejemplo $u_0 \in C^1(\overline{\Omega})$,

$$\left| \lim_{t \to 0^+} \frac{\overbrace{u_0(z)}^{=0} - u_0(z - t\mathbf{s})}{t} \right| = \left| \frac{\partial u_0}{\partial \mathbf{s}}(z) \right| < \infty, \quad z \in \Gamma_{\mathcal{D}}$$

$u_0(z) = 0$ puesto que $u_0 = g_{\mathcal{D}} = 0$ en $\Gamma_{\mathcal{D}}$. Además, si tomamos $x = z - t\mathbf{s} \in \Omega$, si $\partial\Omega$ es razonable, $||\mathbf{s}||_2 = 1$. Luego $t \approx d(x, \Gamma_{\mathcal{D}}) \approx d_{\mathcal{D}}(x)$ y entonces, reemplazando en lo anterior, obtenemos que ese límite es aproximadamente $\frac{u_0(x)}{d(x, d_{\mathcal{D}}(x))}$ y, por lo tanto, está acotado.

Además, agregando a lo anterior, si podemos construir $d_{\mathcal{D}}, d_{\mathcal{N}}, \boldsymbol{\nu}$ de forma tal que

$$\frac{u_0}{d_{\mathcal{D}}} \in H^1(\Omega), \text{ y } \frac{(\boldsymbol{\phi}_0 \cdot \boldsymbol{\nu})\boldsymbol{\nu}}{d_{\mathcal{N}}} \in H(\text{div}; \Omega). \tag{67}$$

Entonces, podríamos primero utilizar el Teorema 3.21, dado que el conjunto Ω lo vamos a considerar estrellado (también acotado), para generar una sucesión $\{(v_m, \boldsymbol{\psi}_m)\}_{m\in\mathbb{N}} \subset C^m$ que aproxime a $(\frac{u_0}{d_{\mathcal{D}}}, \frac{(\boldsymbol{\phi}_0\cdot\boldsymbol{\nu})\boldsymbol{\nu}}{d_{\mathcal{N}}})$ (dado que el resultado asegura en C^∞ y en particular lo tenemos para la derivada m-ésima). Ahora, usando el Teorema 3.30, existe una sucesión que está en el espacio de la red que aproxima a esta sucesión de C^m (teniendo en cuenta que el resultado está sobre compactos pero al considerar Ω acotado nos basta). Entonces puedo, en particular, hacer un abuso de notación para considerar a $\{(v_m, \boldsymbol{\psi}_m)\}_{m\in\mathbb{N}}$ con $(v_m, \boldsymbol{\psi}_m) \in \mathcal{C}_m$ para todo m, tal que

$$\left\| v_m - \frac{u_0}{d_{\mathcal{D}}} \right\|_{H^1(\Omega)} \to 0 \quad \text{y} \quad \left\| \boldsymbol{\psi}_m - \sum_{i=1}^{d-1} (\boldsymbol{\phi}_0 \cdot \mathbf{t})\mathbf{t} - \frac{1 + d_{\mathcal{N}}}{d_{\mathcal{N}}} (\boldsymbol{\phi}_0 \cdot \boldsymbol{\nu})\boldsymbol{\nu} \right\|_{H(\text{div};\Omega)} \to 0.$$

Si $m \to \infty$. Notar que el resultado de densidad está probado para espacios $W^{k,p}$, pero si el resultado es válido para funciones con una derivada débil, cuanto más para funciones en el espacio $H(\text{div})$. Además, notar que en los Teoremas 3.21 y 3.30, las aproximaciones fueron hechas tanto de la función como de sus derivadas, por lo que podemos aproximar las respectivas normas H^1 y $H(\text{div})$. Ahora, si definimos a la secuencia $\{(u_m, \boldsymbol{\phi}_m)\}_{m\in\mathbb{N}}$ tal que $u_m = d_{\mathcal{D}} v_m$ y $\boldsymbol{\phi}_m = \boldsymbol{\psi}_m - \left(\frac{\boldsymbol{\psi}_m \cdot \boldsymbol{\nu}}{1+d_{\mathcal{N}}}\right)\boldsymbol{\nu}$, vamos a tener que $(u_m, \boldsymbol{\phi}_m) \in \mathcal{A}_m$ para todo m, y $(u_m, \boldsymbol{\phi}_m) \to (u_0, \boldsymbol{\phi}_0)$ en $||\cdot||_{H^1(\Omega)\times H(\text{div};\Omega)}$ y, por lo tanto, se cumple la hipótesis (66). Observar que (67) es una hipótesis sobre la regularidad de la solución y eso justamente desemboca en que la solución termine siendo aproximable por una red neuronal.

Ahora bien, como veremos más adelante en (71), vamos a utilizar una versión regularizada del funcional $\mathcal{L}$ definido anteriormente. Esto nos lleva a considerar una colección de funcionales, evaluados en una sucesión de parámetros de red, denotados por Θ_N. Surge entonces la necesidad de establecer un marco teórico que nos permita hablar de convergencia en este tipo de escenarios. Justamente para eso tenemos el concepto de Γ-convergencia.

Para entender mejor esta noción, consideremos una familia de problemas del tipo

$$m_\epsilon = \text{mín}\{F_\epsilon(x) : x \in X_\epsilon\},$$

y nos preguntamos cómo se comporta esta colección cuando $\epsilon \to 0$. En lugar de estudiar directamente el comportamiento puntual de los minimizadores x_ϵ, la idea es encontrar un funcional límite F_0 tal que el problema

$$m_0 = \text{mín}\{F_0(x) : x \in X_0\}$$

sea una buena aproximación del comportamiento asintótico de los anteriores. Es decir, que se cumpla $m_\epsilon \to m_0$ y que, al menos para una subsucesión, $x_\epsilon \to x_0$, donde x_0 es solución de m_0.

Para que este tipo de resultados tenga validez, es necesario exigir ciertas propiedades. Una de ellas es la *equi-coercitividad* de la familia $\{F_\epsilon\}$, que nos garantiza la existencia de una *sucesión mínima pre-compacta*. Esto significa, por un lado, que cada funcional F_ϵ admite un minimizador x_ϵ tal que

$$F_\epsilon(x_\epsilon) \leq \text{ínf}\, F_\epsilon + o(1),$$

y, por otro, que la sucesión $\{x_\epsilon\}$ es precompacta, es decir, admite una subsucesión convergente. Esta condición es clave: aunque la convergencia de x_ϵ no esté garantizada a priori, si logramos extraer una subsucesión convergente, la teoría de la Γ-convergencia asegura que su límite x_0 es minimizador del funcional límite F_0.

La existencia de este funcional F_0, denominado *límite* Γ de la familia $\{F_\epsilon\}$, se establece a partir de dos condiciones fundamentales, que se presentan en la siguiente definición.

Definición 5.2. Sea X un espacio métrico y $F_n, F_0 : X \to \overline{\mathbb{R}}$, donde $\overline{\mathbb{R}} = [-\infty, \infty]$. Decimos que F_n Γ -converge a F_0 (lo notamos por $F_n \xrightarrow{\Gamma} F_0$) si, para cada x_0 tenemos:

- (lim-inf) Para cada sucesión $\{x_n\}$ convergente a x_0 tal que,

$$F_0(x_0) \leq \liminf_{n\to\infty} F_n(x_n);$$

- (lim-sup) Existe una sucesión $\{\overline{x}_n\}$ convergente a x_0 tal que,

$$F_0(x_0) \geq \limsup_{n\to\infty} F_n(\overline{x}_n).$$

En otras palabras, F_0 actúa como una cota inferior de la familia $\{F_n\}$ en el siguiente sentido: si $x_n \to x_0$, entonces

$$F_0(x_0) \leq \liminf_{n\to\infty} F_n(x_n).$$

Si además la familia $\{F_n\}$ es equi-coercitiva, entonces podemos asegurar la existencia de una subsucesión convergente de minimizadores. Recordemos la definición:

Definición 5.3 (Equi-coercitividad)**.** Sea $\{F_n\}_{n\in\mathbb{N}}$ una sucesión de funcionales $F_n : X \to \overline{\mathbb{R}}$. Decimos que $\{F_n\}$ es *equi-coercitiva* si, para todo $t \in \mathbb{R}$, existe un conjunto compacto $K_t \subset X$ tal que

$$\{x \in X : F_n(x) \leq t\} \subset K_t \quad \text{para todo } n.$$

Esta condición implica que toda sucesión $\{x_n\}$ tal que $F_n(x_n) \leq t$ para todo n, admite una subsucesión convergente $x_{n_k} \to x_0$. En particular, permite extraer subsucesiones convergentes de sucesiones acotadas en energía, lo cual es clave para el análisis del comportamiento asintótico de los cuasi-minimizadores.

Supongamos ahora que $F_n \xrightarrow{\Gamma} F$ en X y por equi-coercitividad, tomo $\{x_n\}$ una sucesión de *cuasi-minimizadores*, es decir:

$$F_n(x_n) \leq \text{ínf}\, F_n + \varepsilon_n \quad \text{con } \varepsilon_n \to 0.$$

Además, por equi-coercitividad, existe una subsucesión $x_{n_k} \to x^* \in X$, y por las propiedades de la Γ-convergencia, se cumple:

$$F(x^*) = \lim_{k\to\infty} F_{n_k}(x_{n_k}) = \liminf_{n\to\infty} F_n(x_n).$$

Esto implica que x^* es un minimizador de F.

En particular, si además $x_n \to x_0$, entonces, usando la desigualdad de tipo lím inf, se deduce:

$$\text{ínf } F \leq F(x_0) \leq \liminf_{n\to\infty} F_n(x_n) \leq \liminf_{n\to\infty} \text{ínf } F_n.$$

Por otro lado, si existe una sucesión $\widetilde{x}_n \to x_0$ tal que

$$F(x_0) \geq \limsup_{n\to\infty} F_n(\widetilde{x}_n),$$

entonces, usando que ínf $F_n \leq F_n(\widetilde{x}_n)$, se obtiene:

$$\text{ínf } F \geq \limsup_{n\to\infty} \text{ínf } F_n.$$

Combinando ambas desigualdades, se concluye:

$$\lim_{n\to\infty} \text{ínf } F_n = \text{ínf } F.$$

Este resultado se conoce como el *teorema fundamental de la* Γ*-convergencia*, y justifica el estudio de esta noción para el análisis de problemas variacionales dependientes de parámetros.

Teorema 5.4 (Teorema fundamental de la Γ-convergencia)**.** *Sea (X, d) un espacio métrico, $\{F_n\}_{n\in\mathbb{N}}$ una sucesión equi-coercitiva en X, y F una función tal que $F_n \xrightarrow{\Gamma} F$. Entonces,*

$$\min_X F = \lim_{n\to\infty} \inf_X F_n.$$

Además, si $\{x_n\}_{n\in\mathbb{N}} \subset X$ es una sucesión precompacta tal que

$$\lim_{n\to\infty} F_n(x_n) = \lim_{n\to\infty} \inf_X F_n,$$

entonces todo punto de acumulación de $\{x_n\}$ es un mínimo de F.

Observación 5.5. Es importante destacar que el resultado anterior asegura que la combinación de la equi-coercitividad de una familia de funcionales con su Γ-convergencia garantiza la convergencia de los minimizadores hacia los minimizadores del funcional límite. Es decir, no basta con tener solamente Γ-convergencia, ya que aún necesitaríamos asegurar la existencia de una sucesión que satisfaga las condiciones requeridas para aplicar el teorema. Al agregar la hipótesis de equi-coercitividad, garantizamos que, bajo ciertas condiciones sobre los funcionales F_n, existe al menos una sucesión de cuasi-minimizadores que converge a un mínimo de F.

5.5. Convergencia

Algo razonable sería intentar demostrar que el método propuesto converge; para eso, primero tendremos que demostrar una serie de lemas.
Primero, mostremos la continuidad de las funciones de la red neuronal con respecto a los parámetros.

Lema 5.6. *El mapa*

$$\Theta \to \boldsymbol{q}_\Theta = (u_\Theta, \boldsymbol{\phi}_\Theta) \in (\mathcal{A}_m, ||\cdot||_{H^1(\Omega)\times H(div;\Omega)})$$

es continuo. Más aún, definiendo $G_1, G_2 : \mathbb{R}^m \times \Omega \to \mathbb{R}$,

$$G_1(\Theta, x) := |\boldsymbol{\phi}_\Theta(x) - A\nabla u_\Theta(x)|^2, G_2(\Theta, x) := |div\ \boldsymbol{\phi}_\Theta(x) - Bu_\Theta(x) + f(x)|^2, \tag{68}$$

para todo $R > 0$ *tenemos que* $G_1 \in L^\infty(B(0,R)\times\Omega)$ *y si asumimos que* $f \in L^2(\Omega)$*, existe una función* $s \in L^1((B(0,R)\times\Omega)$*, que depende de* R*, tal que* $|G_2(\Theta, x)| \le s(\Theta, x)$*, para todo* $(\Theta, x) \in B(0,R)\times\Omega$.

Demostración. Primero, consideremos una red neuronal genérica $v_\Theta : \mathbb{R}^d \to \mathbb{R}$ con solo una capa oculta, dada por:

$$v_\Theta(x) = B\sigma(Ax + c).$$

Aquí, asumimos que σ es una función de activación Lipschitz continua, y los parámetros $B \in \mathbb{R}^{1\times n}$, $A \in \mathbb{R}^{n\times d}$ y $c \in \mathbb{R}^{n\times 1}$ están recopilados en $\Theta \in \mathbb{R}^m$, donde $m = n(d+2)$. Utilizando el hecho de que v_Θ y sus derivadas dependen de forma continua de los parámetros, dado que $\sigma \in C^\infty$ y multiplicar por matrices también, la función que va de $\mathbb{R}^m \to W^{1,\infty}(\Omega)$ dada por $\Theta \mapsto v_\Theta$ es continua. Además, la función $G : \mathbb{R}^m \times \Omega \to \mathbb{R}$, definida como $G(\Theta, x) := v_\Theta(x)$, es Lipschitz continua, pues v_Θ lo es, por lo que está acotada en $B(0,R)\times\Omega$, pues los parámetros están acotados y σ es una función de activación , y sus derivadas débiles también son esencialmente acotadas en ese mismo conjunto. Además, si $f \in L^2(\Omega)$, entonces:

$$|G(\Theta, x) + f(x)|^2 \le 2|G(\Theta, x)|^2 + 2|f(x)|^2 \le 2M + 2|f(x)|^2 =: s(\Theta, x),$$

donde $s \in L^1(B(0,R)\times\Omega)$.

Ahora, para funciones arbitrarias de una red $(u_\Theta, \boldsymbol{\phi}_\Theta)$ en el espacio $\mathcal{A}_m$, definido por (65), usamos exactamente la idea anterior, con la observación de que las funciones auxiliares $d_\mathcal{D}$, $d_\mathcal{N}$ y $\boldsymbol{\nu}$ son suaves, para concluir el resultado deseado.

□

Lema 5.7 (Aproximaciones de $\mathcal{A}_m$)**.** *Para cada* $m \in \mathbb{N}$*, definimos el conjunto de cuasi-minimizadores de redes neuronales como:*

$$\mathcal{I}_m := \{\boldsymbol{q} \in \mathcal{A}_m : \mathcal{L}(\boldsymbol{q}) \le \mathcal{L}(\boldsymbol{q}^*) + \frac{1}{m}\ \forall \boldsymbol{q}^* \in \mathcal{A}_m\}.$$

Entonces, si $\boldsymbol{q}_0$ *es el único minimizador de* $\mathcal{L}$ *en* $\mathcal{A}$*, tenemos que:*

$$\sup_{\boldsymbol{q}_m \in \mathcal{I}_m} \|\boldsymbol{q}_m - \boldsymbol{q}_0\|_{H^1(\Omega)\times H(div;\Omega)} \to 0 \quad cuando \quad m \to \infty.$$

Demostración. De 4.3 sabemos que $\mathcal{L}$ es elíptico con respecto a la norma $H^1(\Omega)\times H(\mathrm{div};\Omega)$. Es decir, existen constantes positivas α y β tales que

$$\alpha\|(u,\boldsymbol{\phi})\|_{H^1(\Omega)\times H(\mathrm{div};\Omega)} \leq \|\boldsymbol{\phi}-A\nabla u\|^2_{L^2(\Omega)} + \|\mathrm{div}(\boldsymbol{\phi})-Bu\|^2_{L^2(\Omega)} \leq \beta\|(u,\boldsymbol{\phi})\|_{H^1(\Omega)\times H(\mathrm{div};\Omega)}, \tag{69}$$

para todo $(u,\boldsymbol{\phi})\in H^1(\Omega)\times H(\mathrm{div};\Omega)$.

Sea $\epsilon>0$. De acuerdo con (66), consideramos un $m_0>0$ tal que $d(\mathbf{q}_0,\mathcal{A}_m)<\epsilon$ y $1/m<\epsilon$ para $m>m_0$. Para cada $m>0$, existe un $\mathbf{q}^*_m=(u^*_m,\boldsymbol{\phi}^*_m)\in\mathcal{A}_m$ con $d(\mathbf{q}_0,\mathcal{A}_m)\geq \|\mathbf{q}^*_m-\mathbf{q}_0\|_{H^1(\Omega)\times H(\mathrm{div};\Omega)}-\epsilon$. Entonces, para todo $m>m_0$ y para cada cuasi-minimizador de red neuronal $\mathbf{q}_m=(u_m,\boldsymbol{\phi}_m)\in\mathcal{I}_m$, utilizando que la solución $\mathbf{q}_0=(u_0,\boldsymbol{\phi}_0)$ de (20) satisface $\boldsymbol{\phi}_0=A\nabla u_0$ y $-\mathrm{div}(\boldsymbol{\phi}_0)+Bu_0=f$ casi en todas partes en Ω , y aprovechando la cota superior en (4.3), obtenemos

$$\begin{aligned}
0\leq \mathcal{L}(\mathbf{q}_m)\leq \mathcal{L}(\mathbf{q}^*_m)+\epsilon &= \|\boldsymbol{\phi}^*_m-A\nabla u^*_m\|^2_{L^2(\Omega)}+\|\mathrm{div}(\boldsymbol{\phi}^*_m)-Bu^*_m+f\|^2_{L^2(\Omega)}+\epsilon.\\
&= \|\boldsymbol{\phi}^*-\boldsymbol{\phi}_0-A\nabla(u^*_m-u_0)\|^2_{L^2(\Omega)}+\|\,\mathrm{div}(\boldsymbol{\phi}^*_m-\boldsymbol{\phi}_0)-B(u^*_m-u_0)\|^2_{L^2(\Omega)}+\epsilon\\
&\leq \beta\|\mathbf{q}^*_m-\mathbf{q}_0\|_{H^1(\Omega)\times H(\mathrm{div};\Omega)}+\epsilon\leq \beta(d(\mathbf{q}_0,\mathcal{A}_m)+\epsilon)+\epsilon=(2\beta+1)\epsilon
\end{aligned}$$

Donde en la segunda igualdad utilizamos que $\mathbf{q}_0$ es solución de la ecuación y en la tercera desigualdad utilizamos (4.3).

Exactamente lo mismo hacemos ahora pero con la cota inferior de (4.3),usando nuevamente el hecho de que $\mathbf{q}_0$ cumple la ecuación casi en todas partes en Ω , concluimos que

$$\begin{aligned}
\|\mathbf{q}_m-\mathbf{q}_0\|_{H^1(\Omega)\times H(\mathrm{div};\Omega)} &\leq \frac{1}{\alpha}\left(\|\boldsymbol{\phi}_m-\boldsymbol{\phi}_0-A\nabla(u_m-u_0)\|^2_{L^2(\Omega)}+\|\mathrm{div}(\boldsymbol{\phi}_m-\boldsymbol{\phi}_0)+B(u_m-u_0)\|^2_{L^2(\Omega)}\right)\\
&\leq \frac{1}{\alpha}\left(\|\boldsymbol{\phi}_m-A\nabla u_m\|^2_{L^2(\Omega)}+\|\mathrm{div}(\boldsymbol{\phi}_m)+Bu_m+f\|^2_{L^2(\Omega)}\right)=\frac{\mathcal{L}(\mathbf{q}_m)}{\alpha}\leq\frac{2\beta+1}{\alpha}\epsilon,
\end{aligned}$$

para cada $\mathbf{q}_m\in\mathcal{I}_m$ y $m>m_0$. Dado que ϵ es arbitrariamente pequeño, esto concluye la demostración.

□

Observación 8. *Hay que observar el hecho de que el último resultado asume que, dados los parámetros $\Theta\in\mathbb{R}^m$, uno puede calcular de forma exacta a $\mathcal{L}(u_\Theta,\boldsymbol{\phi}_\Theta)$. Esto no es necesariamente cierto, observar que en el operador están involucradas las normas, que en nuestro caso, computaremos mediante el método de Monte Carlo. Para tener esto en cuenta, vamos a considerar una forma regularizada del operador de pérdida $\mathcal{L}$, eso es $\mathcal{L}_N:\mathcal{A}_m\to\overline{\mathbb{R}}$, usando a $\mathbb{R}^n$ como dominio. Entonces dado $R>0$ definimos la formulación regularizada del funcional como*

$$L(\Theta):=\begin{cases}\mathcal{L}(u_\Theta,\boldsymbol{\phi}_\Theta) & \text{si } |\Theta|\leq R,\\ +\infty & \text{cualquier otro caso.}\end{cases} \tag{70}$$

Ahora si tomamos $\{X_i\}_{i\in\mathbb{N}}$ sucesión de variables aleatorias idénticamente distribuidas, definidas en el espacio de probabilidades (Λ,Σ,P) con $X_i:\Lambda\to\Omega$ para todo $i\in\mathbb{N}$, con densidad uniforme sobre Ω. Dado $\lambda\in\Lambda$, $R>0$, y $N\in\mathbb{N}$, escribimos $V_N(\lambda):=\bigcup_{i\leq N}\{X_i(\lambda)\}$ y la forma regularizada discreta del funcional $L_{\lambda,N}:\mathbb{R}^m\to\overline{\mathbb{R}}$ como

$$L_{\lambda,N}(\Theta):=\begin{cases}\frac{|\Omega|}{N}\sum\limits_{x\in V_n(\lambda)}G_1(\Theta,x)+G_2(\Theta,x) & \text{si } |\Theta|\leq R,\\ +\infty & \text{cualquier otro caso,}\end{cases} \tag{71}$$

Con G_1 y G_2 como en (68).

Con estas definiciones podemos probar una convergencia P-casi segura puntual de la sucesión $\{L_{\lambda,N}\}$ hacia L.

Lema 5.8. *Consideremos $R > 0$, el funcional L como en (70), $L_{\lambda,N}$ y una familia $\{X_i\}_{i\in\mathbb{N}}$ de variables aleatorias independientes e idénticamente distribuidas (i.i.d.) definidas en el espacio de probabilidad (Λ, Σ, P) como en (71). Entonces, $L_{\lambda,N}(\Theta) \to L(\Theta)$ cuando $N \to \infty$, casi seguramente en P, para todo $\Theta \in \mathbb{R}^m$.*

Demostración. Dado que estamos utilizando el mismo parámetro R en las definiciones de L y L_N, si $\Theta > R$, entonces $L(\Theta) = L_N(\Theta) = +\infty$, y no hay nada que demostrar. Por lo tanto, supongamos que $\Theta \leq R$. Recordando que $V_N(\lambda) := \bigcup_{i\leq N}\{X_i(\lambda)\}$, junto con la definición de G_1 y G_2 en (68) y $\lambda \in \Lambda$, y si consideramos que los puntos $x \in V_N(\lambda)$ son muestreados de manera i.i.d., entonces una aplicación de la ley fuerte de los grandes números implica que, para toda función integrable g, se verifica:

$$\frac{|\Omega|}{N}\sum_{x\in V_N(\lambda)} g(x) \xrightarrow{\text{c.t.p}} \int_\Omega g(x)\,dx \quad \text{cuando } N \to \infty.$$

Aplicando esto a $g(x) = |\boldsymbol{\phi}(x) - A\nabla u(x)|^2$ y $g(x) = |\operatorname{div}(\boldsymbol{\phi}(x)) - Bu(x) + f(x)|^2$, obtenemos:

$$\frac{|\Omega|}{N}\sum_{x\in V_N(\lambda)} |\boldsymbol{\phi}(\boldsymbol{x}) - A\nabla u(x)|^2 \underset{N\to\infty}{\overset{\text{c.t.p}}{\to}} \int_\Omega |\boldsymbol{\phi} - A\nabla u|^2$$

y

$$\frac{|\Omega|}{N}\sum_{x\in V_N(\lambda)} |\operatorname{div}(\boldsymbol{\phi}(x)) - Bu(x) + f(x)|^2 \underset{N\to\infty}{\overset{\text{c.t.p}}{\to}} \int_\Omega |\operatorname{div}(\boldsymbol{\phi}) - Bu + f|^2$$

para todo $(u, \boldsymbol{\phi}) \in \mathcal{A}_m$ a medida que $N \to \infty$. Esto implica inmediatamente que $L_{\lambda,N}(\Theta) \to L(\Theta)$ casi seguramente en probabilidad, cuando $N \to \infty$.

□

Ahora queremos ver más todavía, nos gustaría probar que tenemos en realidad una Γ convergencia de $L_{\lambda,N}$ hacia L, para así luego poder utilizar resultados anteriores y garantizar la convergencia real de los minimizadores hacia el límite L.

Teorema 5.9 (Γ convergencia c.t.p.)**.** *Sea $R > 0$, y sea L como en (70). Considérese además a $L_{\lambda,N}$ y a una familia de variables aleatorias i.i.d. X_i definidas en el espacio de probabilidad (Λ, Σ, P) como se describe en (71). Suponiendo que $f \in L^2(\Omega)$, se cumple que $L_{\lambda,N} \xrightarrow{\Gamma} L$ cuando $N \to \infty$, P-casi seguramente.*

Demostración. Primero observemos que el límite superior es un corolario trivial del lema anterior, dado que podemos considerar $\{\Theta_N\}_{N\in\mathbb{N}} \subset \mathbb{R}^m$, con $\Theta_N \equiv \Theta$ y por el lema anterior que $L_{\lambda,N}(\Theta_N) \to L(\Theta)$ con $N \to \infty$ c.t.p en probabilidad.

Ahora faltaría probar el límite inferior, es decir, dado $\Theta \in \mathbb{R}^m$, sea $\{\Theta_N\}_{N\in\mathbb{N}} \subset \mathbb{R}^m$ una sucesión de parámetros tal que $\Theta_N \to \Theta$, entonces queremos probar que

$$L(\Theta) \leq \liminf_{N\to\infty} L_{\lambda,N}(\Theta_N). \tag{72}$$

Notemos que si $|\Theta| > R$, entonces existe $N_0 = N_0(\lambda)$ tal que $L(\Theta) = L_{\lambda,N}(\Theta_N) = +\infty$ para todo $N > N_0$, por lo que (72) se cumple trivialmente. Por este motivo podemos asumir sin pérdida de generalidad que, $\{\Theta_N\}_{N\in\mathbb{N}} \subset \overline{B(0,R)}$. Entonces podemos ahora tomar una subsucesión que converja al límite inferior, $L_{\lambda,N}(\Theta_N) \to \liminf_{N\to\infty} L_{\lambda,N}(\Theta_N)$ y por simplicidad nos ahorramos renombrar la sucesión. Ahora, por el lema 5.6, sabemos que el mapa $\Theta \to (u_\Theta, \boldsymbol{\phi}_\Theta) \in (\mathcal{A}_m, \|\cdot\|_{H^1(\Omega)\times H(\mathrm{div};\Omega)})$ es continuo y por lo tanto $(u_{\Theta_N}, \boldsymbol{\phi}_{\Theta_N}) \to (u_\Theta, \boldsymbol{\phi}_\Theta)$ en la norma $H^1(\Omega) \times H(\mathrm{div};\Omega)$, puesto que $\Theta_N \to \Theta$. Ahora, utilizando el hecho de que Ω es acotado, podemos usar la proposición 3.1, para poder obtener

$$\|u_{\Theta_N} - u_\Theta\|_{L^1(\Omega)} \to 0, \qquad \|\nabla u_{\Theta_N} - \nabla u_\Theta\|_{L^1(\Omega)} \to 0,$$
$$\|\boldsymbol{\phi}_{\Theta_N} - \boldsymbol{\phi}_\Theta\|_{L^1(\Omega)} \to 0, \quad \text{y} \quad \|\mathrm{div}\ \boldsymbol{\phi}_{\Theta_N} - \mathrm{div}\ \boldsymbol{\phi}_\Theta\|_{L^1(\Omega)} \to 0.$$

Si definimos a G_1 y G_2 como en (68), podemos como antes tomar una subsucesión tal que $G_1(\Theta_N, x) + G_2(\Theta_N, x) \to G_1(\Theta, x) + G_2(\Theta, x)$ c.t.p en Ω y como antes omitimos renombrar la sucesión.
Sea $\epsilon > 0$, usando desigualdad triangular

$$|L_{\lambda,N}(\Theta_N) - L(\Theta)| \le |L_{\lambda,N}(\Theta_N) - L_{\lambda,N}(\Theta)| + |L_{\lambda,N}(\Theta) - L(\Theta)|. \tag{73}$$

Entonces por el lema anterior, $|L_{\lambda,N}(\Theta) - L(\Theta)| \to 0$ c.t.p en probabilidad, que es lo mismo que exista un $N_0 = N_0(\lambda)$ tal que $|L_{\lambda,N}(\Theta) - L(\Theta)| \le \epsilon/4$ para todo $N > N_0$.
Queremos acotar el primer término, para eso volvamos un poco al lema 5.6, que nos dice que G_1 es uniformemente acotada y G_2 está acotada por alguna función integrable $s \in L^1(\Omega)$, que depende de R. Entonces tenemos,

$$|G_1(\Theta_N, x) + G_2(\Theta_N, x) - G_1(\Theta, x) - G_2(\Theta, x)| \le s(x), \tag{74}$$

para todo $(\Theta, x) \in B(0,R) \times \Omega$. Ahora podemos aplicar el teorema de Egorov 3.2 para construir un conjunto $F \subset \Omega$ tal que $\int_F s(x)dx < \epsilon/8$, pues está en $L^1(\Omega)$ y en particular de F y $G_1(\Theta_N, \cdot) + G_2(\Theta_N, \cdot) \to G_1(\Theta, \cdot) + G_2(\Theta, \cdot)$ uniformemente en $\Omega \setminus F$. Acotamos (utilizando la desigualdad triangular del módulo),

$$|L_{\lambda,N}(\Theta_N) - L_{\lambda,N}(\Theta)| \le A_1 + A_2$$

Con

$$A_1 = \frac{|\Omega|}{N} \sum_{x\in V_N(\lambda)\cap(\Omega\setminus F)} |G_1(\Theta_N, x) + G_2(\Theta_N, x) - G_1(\Theta, x) - G_2(\Theta, x)|,$$
$$A_2 = \frac{|\Omega|}{N} \sum_{x\in V_N(\lambda)\cap F} |G_1(\Theta_N, x) + G_2(\Theta_N, x) - G_1(\Theta, x) - G_2(\Theta, x)|.$$

Usando la convergencia uniforme sobre $\Omega \setminus F$, existe un $N_1 = N_1(\lambda)$ c.t.p en probabilidad, tal que si $N > N_1$, entonces $|G_1(\Theta_N, x) + G_2(\Theta_N, x) - G_1(\Theta, x) - G_2(\Theta, x)| < \frac{\epsilon}{4|\Omega|}$ para todo $x \in \Omega \setminus F$. De esto se sigue que $A_1 < \epsilon/4$ si $N > N_1$.
Por otro lado, usamos (74) para acotar,

$$A_2 \le \frac{|\Omega|}{N} \sum_{x\in V_N(\lambda)} \chi_F(x) s(x).$$

Y por el mismo argumento de la ley de los grandes números que veníamos utilizando, obtenemos

$$\frac{|\Omega|}{N}\sum_{x\in V_N(\lambda)}\chi_F(x)s(x) \overset{\text{c.t.p}}{\underset{N\to\infty}{\to}} \int_F s(x) < \frac{\epsilon}{8}$$

Por lo tanto, existe $N_2 = N_2(\lambda)$ c.t.p en probabilidad tal que si $N > N_2$, entonces

$$\left|\frac{|\Omega|}{N}\sum_{x\in V_N(\lambda)}\chi_F(x)s(x) - \int_F s(x)\right| < \frac{\epsilon}{8}.$$

esto implica que $\frac{|\Omega|}{N}\sum_{x\in V_N(\lambda)}\chi_F(x)s(x) < \frac{\epsilon}{4}$, es decir, $A_2 < \frac{\epsilon}{4}$.
Ahora, si tomamos todos los N_i tal que $N' = N'(\lambda) = max\{N_0, N_1, N_2\}$, tenemos que c.t.p en probabilidad,

$$|L_{\lambda,N}(\Theta_N) - L(\Theta)| \leq |L_{\lambda,N}(\Theta_N) - L_{\lambda,N}(\Theta)| + |L_{\lambda,N}(\Theta) - L(\Theta)| \leq \epsilon,$$

para todo $N > N'$ □

Con esto último podemos enunciar y demostrar el teorema de convergencia.

Teorema 5.10 (Convergencia)**.** *Supongamos que, para cualquier $m \in \mathbb{N}$ fijo y $R > 0$, se puede construir una sucesión $\{\Theta_N\}_{N\in\mathbb{N}} \subset B(0,R) \subset \mathbb{R}^m$, tal que*

$$\lim_{N\to\infty} L_{\lambda,N}(\Theta_N) = \lim_{N\to\infty} \inf_{\Theta\in\mathbb{R}^m} L_{\lambda,N}(\Theta).$$

Sea $(u_0, \boldsymbol{\phi}_0) = \boldsymbol{q}_0 = \arg\min_{q\in\mathcal{A}} \mathcal{L}(q)$. Dado $\epsilon > 0$, existe (casi seguramente en probabilidad) $m_0 = m_0(\epsilon) \in \mathbb{N}$, $R = R(m_0) > 0$ y $N_0 = N_0(m_0) \in \mathbb{N}$ tales que, si se construye una sucesión $\{\Theta_N\}_{N\in\mathbb{N}} \subset B(0,R)$ como se indicó anteriormente, entonces

$$\|(u_0, \boldsymbol{\phi}_0) - (u_{\Theta_N}, \boldsymbol{\phi}_{\Theta_N})\|_{H^1(\Omega)\times H(div;\Omega)} \leq \epsilon,$$

para todo $N > N_0$, donde $(u_{\Theta_N}, \boldsymbol{\phi}_{\Theta_N})$ es la función de red neuronal definida por los parámetros Θ_N.

Demostración. Sea $\epsilon > 0$ y consideremos el conjunto de cuasi-minimizadores de redes neuronales introducido en el Lema 5.7:

$$\mathcal{I}_m := \left\{\mathbf{q} \in \mathcal{A}_m : \mathcal{L}(\mathbf{q}) \leq \mathcal{L}(\mathbf{q}^*) + \frac{1}{m}\ \forall \mathbf{q}^* \in \mathcal{A}_m\right\}.$$

Por dicho lema, existe $m_0 \in \mathbb{N}$ tal que

$$\|\mathbf{q}_0 - \mathbf{q}_{m_0}\|_{H^1(\Omega)\times H(\text{div};\Omega)} < \frac{\epsilon}{2}, \tag{75}$$

para todo $\mathbf{q}_{m_0} \in \mathcal{I}_{m_0}$.

Fijamos ahora $R_0 > 0$ suficientemente grande para que exista $\Theta \in B(0,R_0)$ tal que la red neuronal correspondiente, $\mathbf{q}^*_\Theta = (p^*_\Theta, \boldsymbol{\phi}^*_\Theta)$, pertenezca a $\mathcal{I}_{m_0}$. Esto implica que $\Theta \in \arg\min_{\Theta\in B(0,R_0)} L(\Theta)$ para el funcional L definido en (70).

Para esta elección de m_0 y R_0, del teorema anterior se sigue que $L_{\lambda,N} \xrightarrow{\Gamma} L$ casi seguramente en probabilidad. Además, por la definición de $L_{\lambda,N}$ en (71), se tiene que la familia $\{L_{\lambda,N}\}_{N\in\mathbb{N}}$ es equi-coerciva (ver Definición 5.3), ya que los conjuntos $\{L_{\lambda,N} \leq t\}$ están contenidos en bolas debido a la forma funcional de $L_{\lambda,N}$.

Entonces, por el teorema fundamental de la Γ-convergencia 5.4, toda subsucesión de minimizadores de $L_{\lambda,N}$ tiene una subsucesión convergente a un minimizador del funcional límite L. En particular, como por hipótesis $\lim_{N\to\infty} L_{\lambda,N}(\Theta_N) = \lim_{N\to\infty} \inf L_{\lambda,N}$, se deduce que existe una subsucesión de Θ_N convergente a algún mínimo $\widetilde{\Theta}$ de L.

Como el mapa $\Theta \mapsto (u_\Theta, \boldsymbol{\phi}_\Theta)$ es continuo (ver 5.6), se sigue que

$$\|(u_{\Theta_N}, \boldsymbol{\phi}_{\Theta_N}) - \mathbf{q}_{m_0}\|_{H^1(\Omega)\times H(\mathrm{div};\Omega)} < \frac{\epsilon}{2}, \tag{76}$$

para todo $N > N_0$ suficientemente grande, con algún $\mathbf{q}_{m_0} \in \mathcal{I}_{m_0}$.

Finalmente, combinando las desigualdades (75) y (76), se obtiene

$$\|\mathbf{q}_0 - (u_{\Theta_N}, \boldsymbol{\phi}_{\Theta_N})\|_{H^1(\Omega)\times H(\mathrm{div};\Omega)} < \epsilon,$$

lo cual concluye la demostración. □

6 Resultados Numéricos

En esta sección presentamos los resultados obtenidos al aplicar el método propuesto. Siguiendo la estrategia planteada en [2], utilizamos funciones auxiliares de distancia para imponer las condiciones de borde de manera fuerte. A diferencia de trabajos previos, en nuestro caso calculamos dichas funciones —$\mathbf{n}, d_{\mathcal{D}}, d_{\mathcal{N}}, G_{\mathcal{D}}, G_{\mathcal{N}}$— de forma exacta, aprovechando que los dominios considerados permiten expresiones analíticas. Esto elimina posibles fuentes de error derivadas de su aproximación y permite un análisis más preciso del desempeño del método.

Las redes neuronales utilizadas para aproximar la solución fueron diseñadas con arquitecturas que varían entre cinco y diez capas ocultas, y entre diez y quince neuronas por capa, empleando funciones de activación tipo sigmoide o ReLU. El entrenamiento se realizó principalmente con el algoritmo ADAM, aunque se llevaron a cabo experimentos comparativos utilizando también Momentum y descenso por gradiente clásico.

Los primeros experimentos se realizaron sobre dominios regulares —en particular, un círculo— con el objetivo de validar el método en un entorno controlado y evaluar su precisión. Posteriormente, se planea extender el análisis a dominios con singularidades geométricas, como una región en forma de L. Además, se exploraron variantes modificando la cantidad de puntos de entrenamiento, la arquitectura de la red y el algoritmo de optimización.

Ejemplo 6.1. Primer ejemplo: dominio circular

Como primer caso de prueba, consideramos un dominio circular:

$$\Omega = \left\{ x \in \mathbb{R}^2 \;\middle|\; \|x\|_2^2 \leq 1 \right\},$$

y utilizamos como solución exacta la función distancia cuadrada al borde:

$$d(x) = 1 - \|x\|_2^2.$$

Recordando la formulación del problema (20), proponemos el siguiente sistema con condiciones de borde homogéneas de Dirichlet y sin contribuciones de tipo Neumann:

$$\begin{cases} \boldsymbol{\phi} - I\nabla u = 0 \quad \text{en } \Omega, \\ -div(\boldsymbol{\phi}) - 4 = 0 \quad \text{en } \Omega, \\ u = 0 \quad \text{en } \Gamma_D = \partial\Omega. \end{cases}$$

La función $d(x)$ cumple las condiciones de borde impuestas y resuelve exactamente el sistema, por lo que se puede utilizar para comparar la aproximación obtenida con el método propuesto.

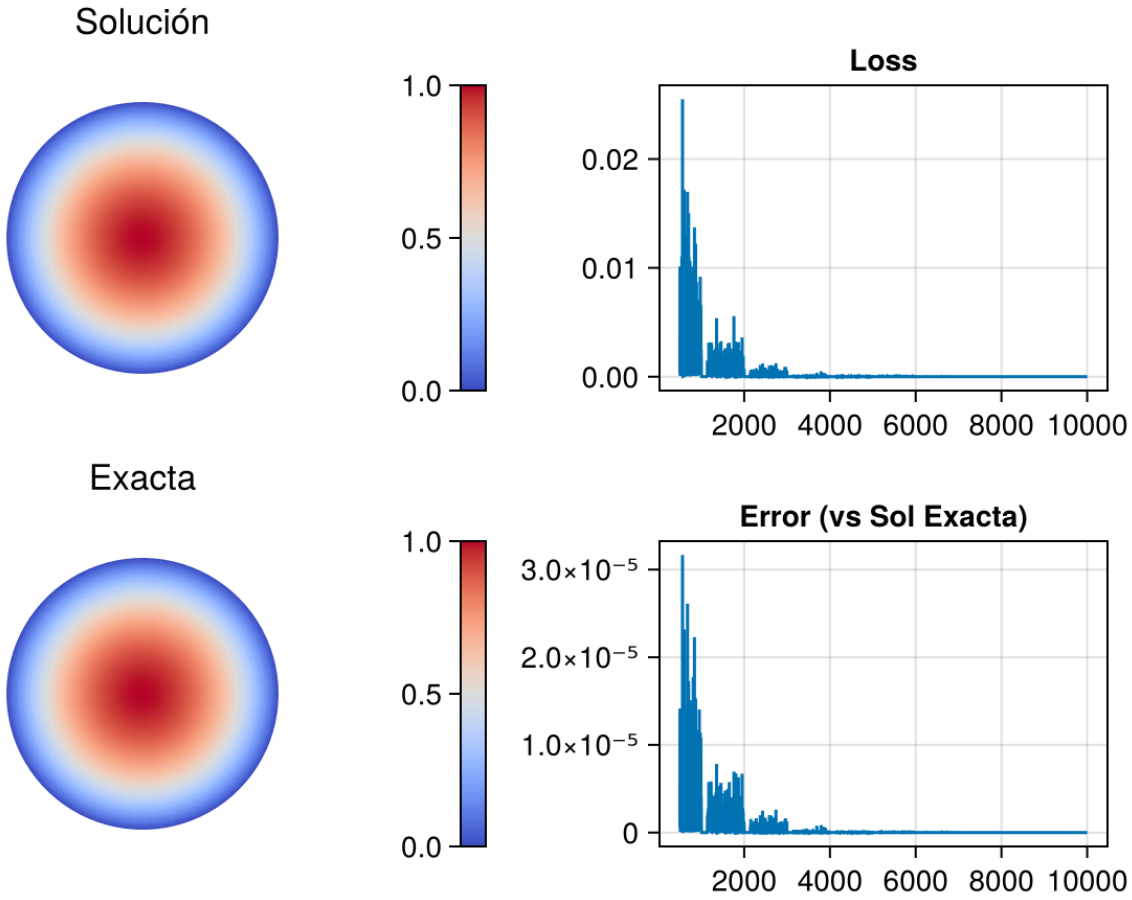


Figura 7: Solución aproximada del problema en un dominio circular utilizando una red neuronal con 15 capas y 5 neuronas por capa oculta. Se emplearon 10000 puntos. El valor final del loss fue $7{,}23 \times 10^{-8}$, con un error absoluto de $5{,}82 \times 10^{-11}$.

En la Figura 7 se muestra la solución aproximada para este caso. Se observa una excelente coincidencia con la solución analítica, con un error absoluto del orden de 10^{-11}. Al aumentar la cantidad de puntos utilizados, se observó sistemáticamente una mejora en la aproximación, aunque también un incremento significativo en el costo computacional. En este ejemplo, el

tiempo de cómputo fue de 558 segundos. En contraste, reduciendo la cantidad de puntos a 2000, el error aumentó a $1{,}08 \times 10^{-8}$, pero el tiempo de cómputo se redujo a 258 segundos.

Segundo ejemplo: dominio circular con condiciones mixtas de Dirichlet y Neumann

Consideramos ahora el siguiente problema:

$$\begin{cases} \boldsymbol{\phi} - A\nabla u = 0 \quad \text{en } \Omega, \\ -div(\boldsymbol{\phi}) + bu + 6 = 0 \quad \text{en } \Omega, \\ u = 1 \quad \text{en } \Gamma_{\mathcal{D}}, \\ \boldsymbol{\phi} \cdot \mathbf{n} = 2x_1^2 + 4x_2^2 \quad \text{en } \Gamma_{\mathcal{N}}. \end{cases}$$

Aquí, $A = \begin{pmatrix} 1 & -1 \\ 1 & 2 \end{pmatrix}$ y $b = 0$ y la solución del problema es la norma de x al cuadrado. Este ejemplo introduce condiciones mixtas de borde, lo cual permite verificar el comportamiento del método bajo una configuración más general.

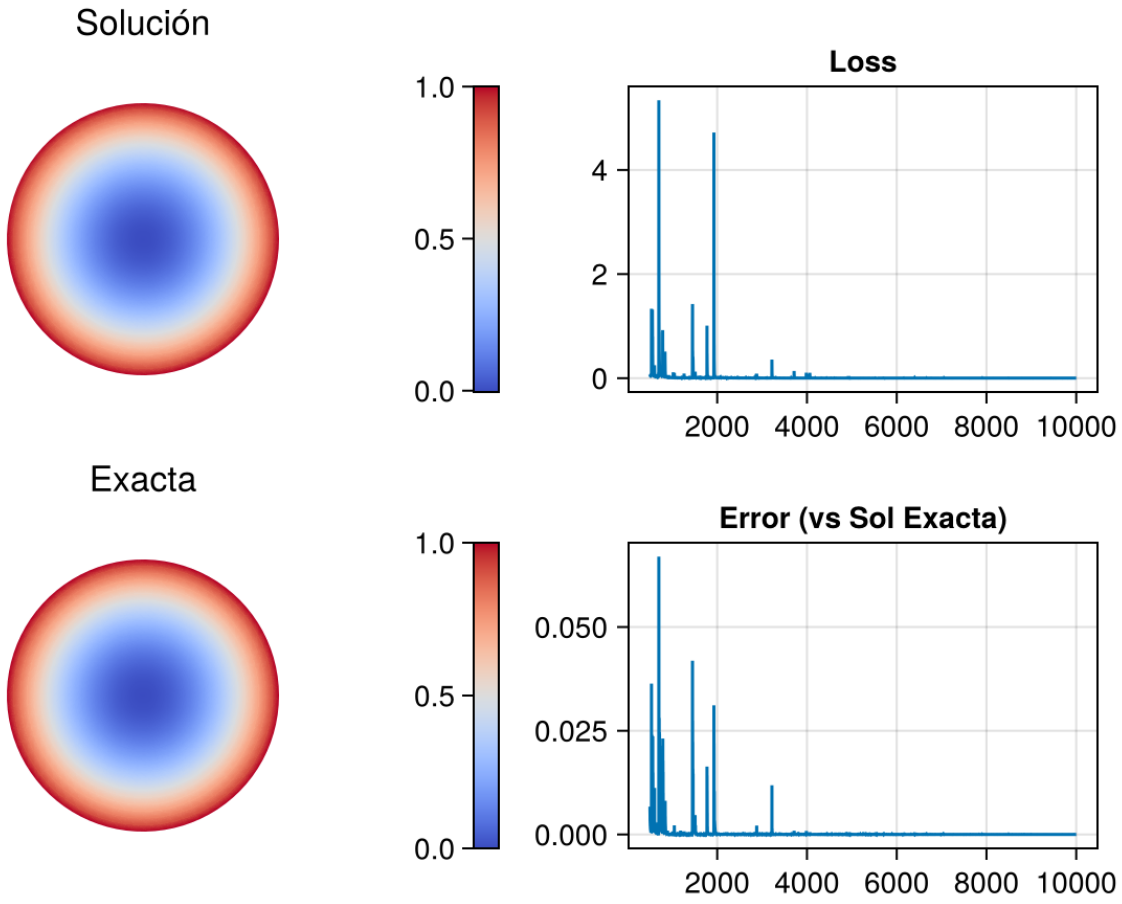


Figura 8: Solución aproximada del problema en un dominio circular utilizando una red neuronal con 15 capas y 5 neuronas por capa oculta. Se emplearon 12000 puntos. El valor final del loss fue $1{,}02 \times 10^{-3}$, con un error absoluto de $1{,}25 \times 10^{-5}$.

Tercer ejemplo: dominio esférico en dimensión $n = 3, 4, 5, 6$
Estudiamos el siguiente sistema con condiciones de borde homogéneas:

$$\begin{cases} \boldsymbol{\phi} - Id\nabla u = 0 \quad \text{en } \Omega, \\ -div(\boldsymbol{\phi}) + 4\pi^2 - f = 0 \quad \text{en } \Omega, \\ u = 0 \quad \text{en } \Gamma_D = \partial\Omega. \end{cases}$$

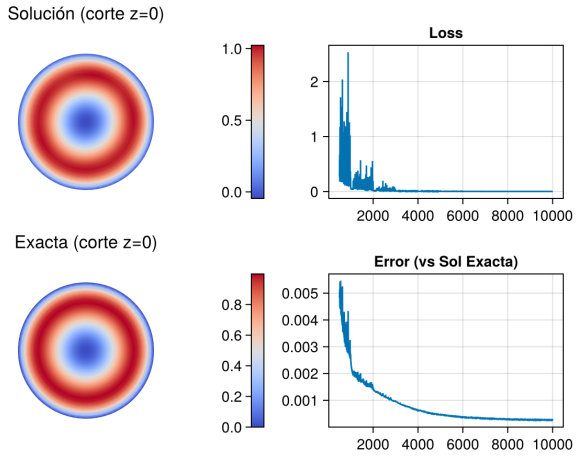
donde

$$f(x) = 4\pi^2 \sin(\pi r^2)(1 + x_1x_2 + x_2x_3 + x_1x_3),$$

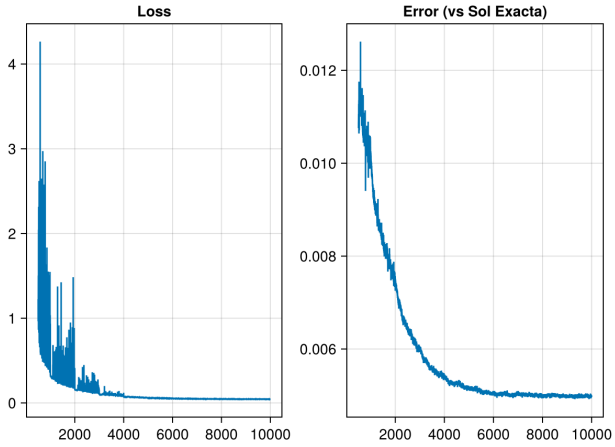
y la solución exacta es:

$$u(x) = \sin(\pi x_1^2 + \pi x_2^2 + \pi x_3^2),$$

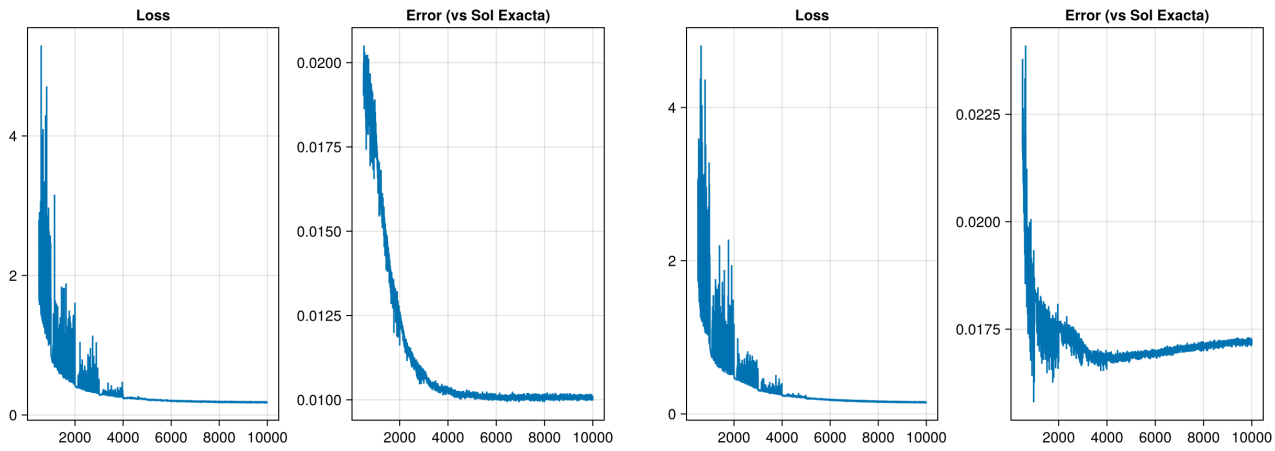
la cual satisface las condiciones de borde y también resuelve la ecuación.



(a) Caso $n = 3$



(b) Caso $n = 4$



(c) Caso $n = 5$ (d) Caso $n = 6$

Figura 9: Soluciones aproximadas en dimensiones crecientes con condiciones de borde homogéneas. Se utilizaron 10 capas ocultas con 5 neuronas cada una. En el caso $n = 3$ se emplearon 60000 puntos, y en los casos $n = 4, 5, 6$ se utilizaron 120, 140 y 340 mil puntos de colocación del dominio respectivamente.

Es interesante observar que en el caso de $n = 6$, se necesitaron muchos más puntos que en los demás solo para obtener una pequeña convergencia. Esta conclusión es debida a que con menos de 340 mil puntos, el error oscilaba. También es interesante observar los tiempos de ejecución. Con $n = 4$ se utilizaron 120 mil puntos y tardó aproximadamente 25 minutos (y cuando se aumentó la cantidad de puntos a 140 igualmente el tiempo no aumentó demasiado) y en $n = 5$ también aproximadamente 30 minutos de duración. Pero, por otro lado, en el caso $n = 6$ tardó aproximadamente una hora y media.

Ejemplo 6.2. En este ejemplo nos enfocamos en un dominio cúbico:

$$\Omega = \{x \in \mathbb{R}^n \mid -1 < x_i < 1, \quad i \in \{1, \dots, n\}\}.$$

Primer caso: dominio cuadrado.

$$\begin{cases} \boldsymbol{\phi} - A\nabla u = 0 \quad \text{en } \Omega, \\ -\text{div}(\boldsymbol{\phi}) - 2\pi^2 u - f = 0 \quad \text{en } \Omega, \\ u = 0 \quad \text{en } \Gamma_D = \partial\Omega. \end{cases} \tag{77}$$

Donde:

$$f(x) = -2\pi^2 \cos(\pi x_1)\cos(\pi x_2), \quad A = \begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix}, \quad u(x) = \sin(\pi x_1)\sin(\pi x_2).$$

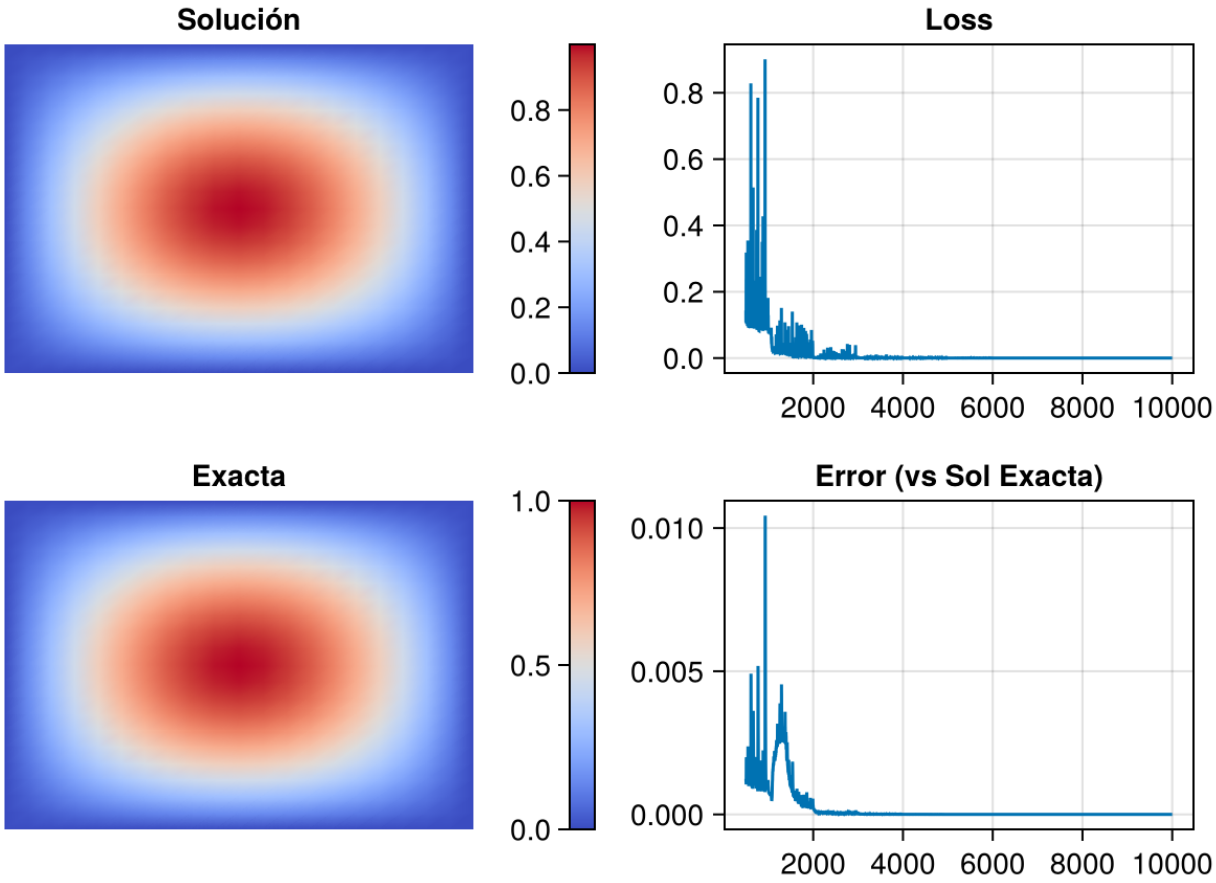


Figura 10: Solución aproximada del problema en un dominio cuadrado utilizando una red con 15 capas ocultas con 5 neuronas cada una. Se emplearon 5000 puntos y 10000 iteraciones. El valor final del loss fue de $7{,}42 \times 10^{-5}$ y el error absoluto de $1{,}29 \times 10^{-6}$.

Segundo caso: dominio cuadrado con condiciones de Dirichlet no homogéneas.

$$\begin{cases} \boldsymbol{\phi} - A\nabla u = 0 \quad \text{en } \Omega, \\ -\text{div}(\boldsymbol{\phi}) - \pi^2 - f = 0 \quad \text{en } \Omega, \\ u = g_{\mathcal{D}} \quad \text{en } \Gamma_D = \partial\Omega. \end{cases} \tag{78}$$

Donde:

$$f(x) = e^{x_1}\left(\pi\cos(\pi x_2) - \sin(\pi x_2)\right),$$
$$A = \begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix}, \quad g_{\mathcal{D}}(x) = (e-1)x_1\sin(\pi x_2),$$
$$u(x) = (e^{x_1} - 1)\sin(\pi x_2).$$

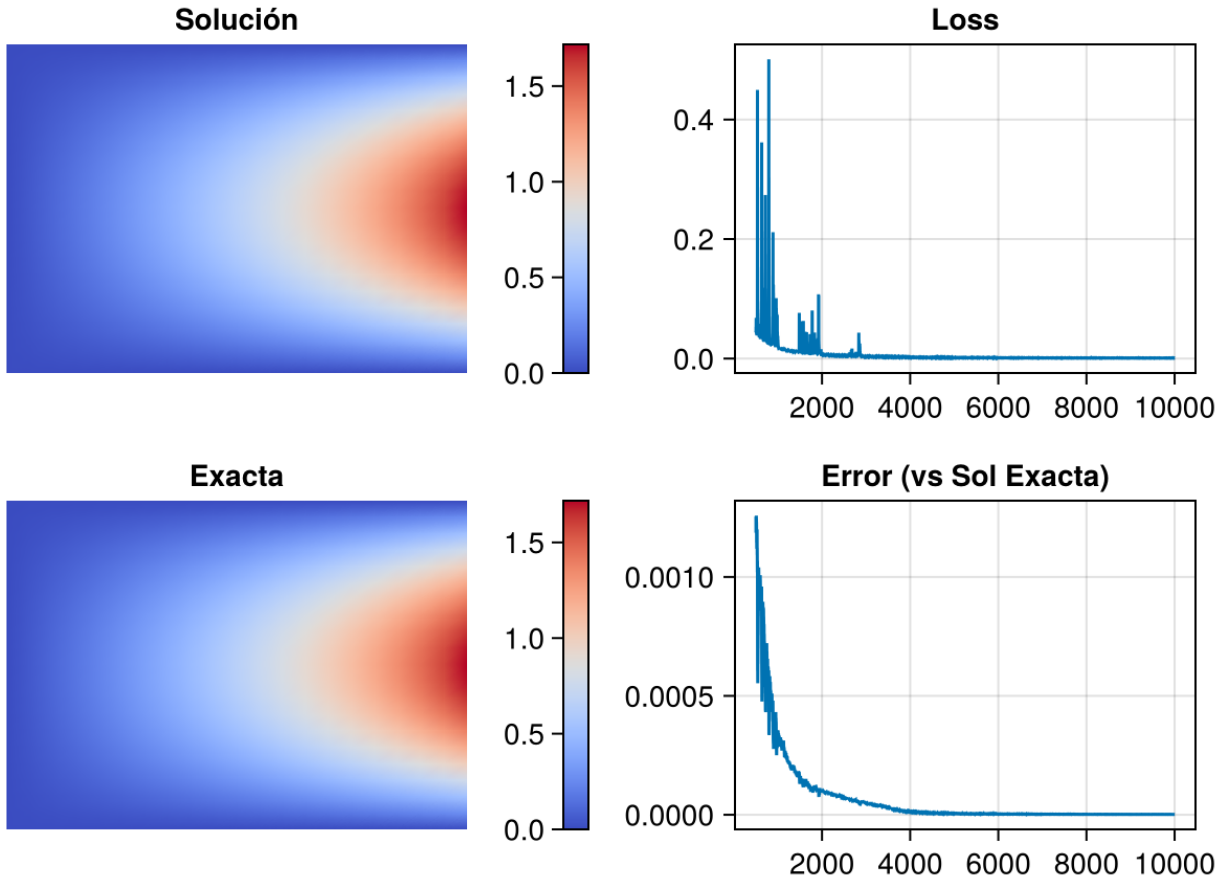


Figura 11: Solución aproximada en un dominio cuadrado con condiciones de borde no homogéneas. Se utilizó una red con 12 capas ocultas con 4 neuronas cada una, 10000 puntos y 10000 iteraciones. El valor del loss fue $8{,}54 \times 10^{-4}$ y el error absoluto $1{,}80 \times 10^{-6}$.

Tercer caso: condiciones mixtas en un dominio cuadrado

Consideramos el dominio cuadrado $\Omega = [-1, 1]^2$ con condiciones de frontera mixtas. El sistema a resolver es:

$$\begin{cases} \boldsymbol{\phi} - A\nabla u = 0 & \text{en } \Omega, \\ -\operatorname{div}(\boldsymbol{\phi}) = f & \text{en } \Omega, \\ u = g_{\mathcal{D}} & \text{en } \Gamma_{\mathcal{D}}, \\ \boldsymbol{\phi} \cdot \mathbf{n} = g_{\mathcal{N}} & \text{en } \Gamma_{\mathcal{N}}, \end{cases} \tag{79}$$

donde la frontera se encuentra particionada como

$$\Gamma_{\mathcal{N}} = \{1\} \times [-1, 1], \quad \Gamma_{\mathcal{D}} = \partial\Omega \setminus \Gamma_{\mathcal{N}}.$$

Se toman las siguientes funciones y coeficientes:

$$
\begin{aligned}
u(x) &= x_1 \sin(\pi x_2), \\
A &= \begin{bmatrix} 1 & -0{,}5 \\ -0{,}5 & 1 \end{bmatrix}, \quad \mathbf{n} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \\
g_{\mathcal{D}} &= 0, \quad g_{\mathcal{N}}(x) = \sin(\pi x_2) - 0{,}5\pi \cos(\pi x_2), \\
f(x) &= \pi \cos(\pi x_2) + \pi^2 x_1 \sin(\pi x_2).
\end{aligned}
$$

La función u satisface las condiciones impuestas y se utiliza como solución exacta para evaluar la precisión del método.

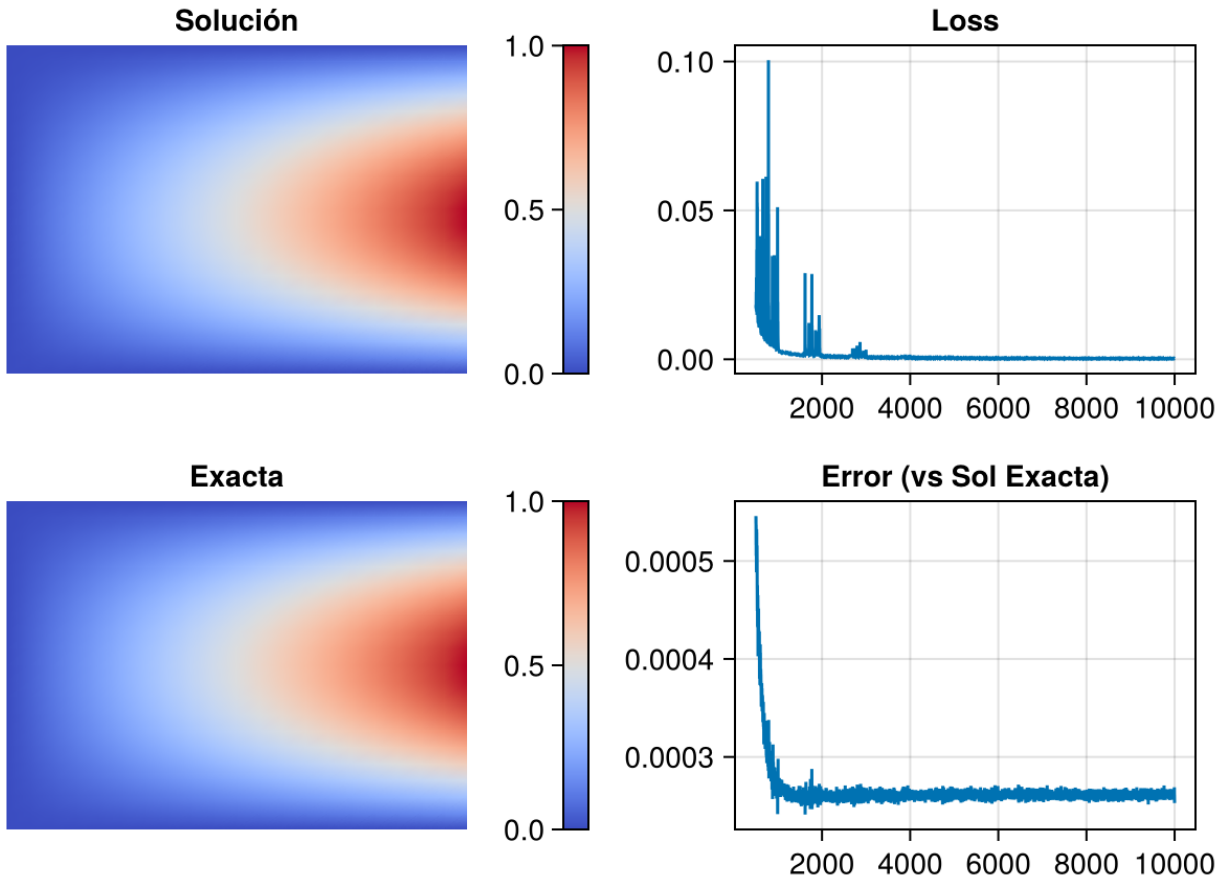


Figura 12: Solución aproximada para un problema con condiciones de frontera mixtas. Se utilizó una red neuronal con 18 capas ocultas de 6 neuronas cada una, entrenada con 20.000 puntos durante 10.000 iteraciones. El valor final del funcional fue $2{,}58 \times 10^{-4}$ y el error absoluto respecto de la solución exacta fue $2{,}62 \times 10^{-4}$.

Algo para destacar de este último caso es que tardó aproximadamente 5 minutos, mientras que por regla general, en todos los otros ejemplos, se demoró más del doble.

Ejemplo 6.3. Cuarto caso: dominio en forma de L

Como último ejemplo, consideramos un dominio no convexo en forma de L, definido como:

$$\Omega = \left\{ (x, y) \in [0, 1]^2 \;\middle|\; (x, y) \notin (0{,}5, 1] \times (0{,}5, 1] \right\}.$$

Planteamos el siguiente problema elíptico con condiciones de Dirichlet homogéneas:

$$\begin{cases} \boldsymbol{\phi} - A\nabla u = 0 & \text{en } \Omega, \\ -\operatorname{div}(\boldsymbol{\phi}) = f & \text{en } \Omega, \\ u = 0 & \text{en } \partial\Omega, \end{cases} \tag{80}$$

donde la matriz difusiva es

$$A = \begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix},$$

y se prescribe como solución exacta la función:

$$u(x) = \sin(2\pi x_1) \sin(2\pi x_2),$$

definida en Ω, con el correspondiente término fuente:

$$f(x) = -8\pi^2 \cos(2\pi x_1) \cos(2\pi x_2).$$

Cabe destacar que esta función se anula en múltiples líneas del dominio, produciendo visualmente una segmentación de la solución en componentes aparentemente desconectados. No obstante, esto es una consecuencia natural de su estructura nodal, y no una propiedad geométrica del dominio.

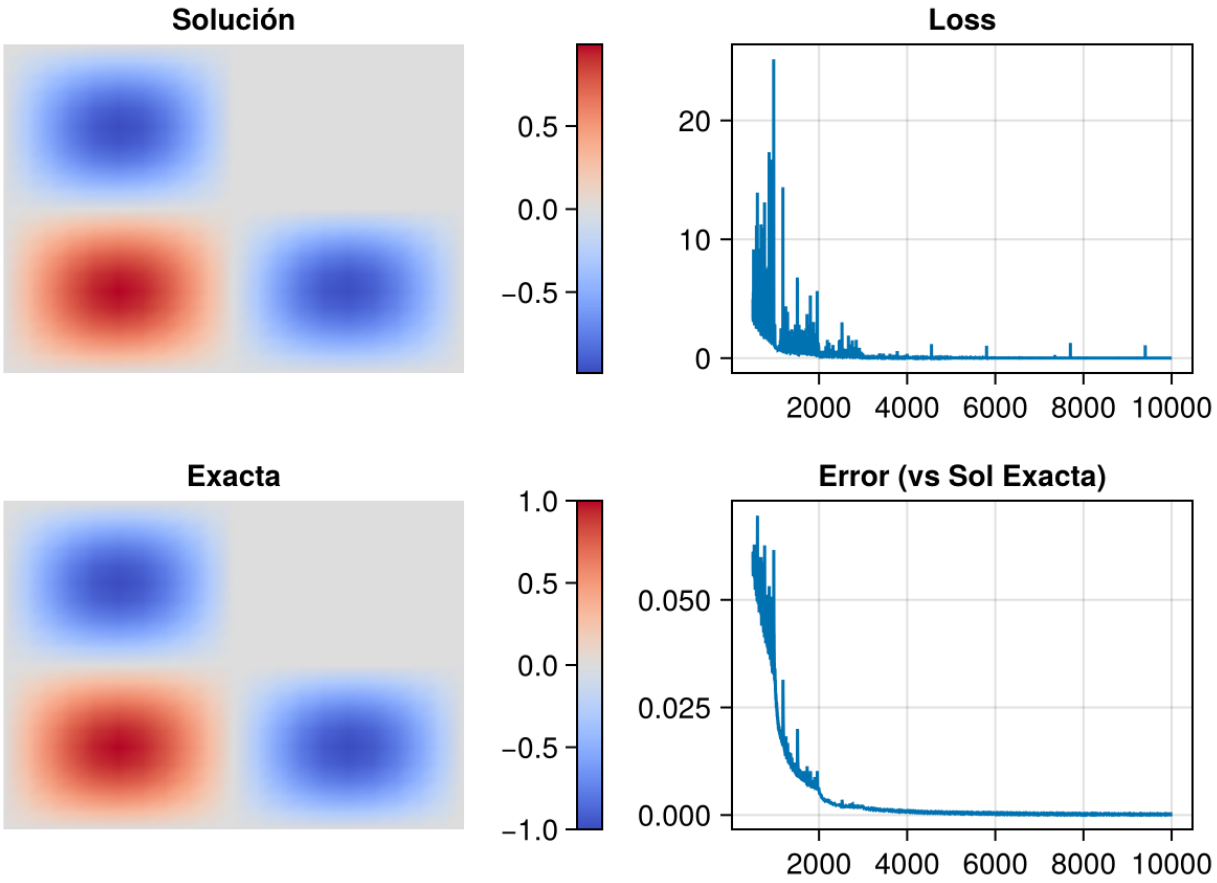


Figura 13: Solución aproximada en un dominio en forma de L con condiciones de borde homogéneas. Se utilizó una red con 15 capas ocultas de 5 neuronas, entrenada con 5000 puntos durante 10.000 iteraciones. El valor del funcional fue $1{,}00 \times 10^{-2}$ y el error absoluto $1{,}61 \times 10^{-4}$.

Finalmente, es importante notar que el dominio en forma de L no cumple las hipótesis clásicas de regularidad necesarias para garantizar la validez del marco teórico propuesto (por ejemplo, la frontera no es suave ni convexa). Aun así, este experimento permite explorar la aplicabilidad del método en geometrías más complejas.

Además de las pruebas de convergencia de los ejemplos, se realizaron pasadas modificando los logaritmos de optimización. Se utilizaron los algoritmos de 'Descend' y 'Momentum' como forma de ejemplo.

7 Conclusiones

En esta tesis abordamos la resolución numérica de ecuaciones diferenciales parciales elípticas de segundo orden mediante métodos basados en redes neuronales profundas. A diferencia de enfoques previos que emplean funciones auxiliares de distancia para imponer condiciones

de borde, nuestro enfoque se centró directamente en la aproximación de la solución, aprovechando que los dominios considerados permitían conocer de forma exacta la distancia al borde. Esta elección nos permitió evitar posibles fuentes de error asociadas a construcciones aproximadas y concentrarnos en el análisis del comportamiento de la red.

Durante los experimentos se observó una marcada sensibilidad del método tanto a la geometría del dominio como a la elección de la arquitectura, funciones de activación y parámetros de entrenamiento. En dominios con bordes suaves, como círculos o esferas, el método mostró un desempeño notablemente superior respecto a dominios con esquinas o singularidades geométricas, como cuadrados o regiones tipo L. Esto sugiere que la regularidad del borde tiene un impacto significativo en la capacidad de la red para capturar el comportamiento de la solución cerca del borde.

Se realizaron pruebas con distintas funciones de activación. Las funciones sigmoideas demostraron ser más estables y eficaces, en particular para dominios no suaves. En contraste, al utilizar ReLU bajo las mismas condiciones, el método no logró converger en dominios tipo L, salvo al aumentar considerablemente la cantidad de puntos de entrenamiento y complejizar la red.

También se compararon varios algoritmos de optimización. Si bien ADAM fue el más eficiente y confiable en los casos considerados, alternativas como Descenso por gradiente o Momentum arrojaron resultados sustancialmente inferiores, tanto en velocidad de convergencia como en precisión final.

Desde el punto de vista computacional, se implementaron varias mejoras para optimizar los tiempos de entrenamiento. En particular, se adaptó el código para utilizar procesamiento en GPU, lo que permitió ejecutar ejemplos más demandantes, incluyendo problemas en hasta seis dimensiones (como se mostró con la esfera en dimensión alta). Además, el entrenamiento se realizó por lotes (*batches*), dividiendo el conjunto total de puntos en subconjuntos más pequeños. Esta técnica, común en aprendizaje profundo, reduce la carga de memoria y mejora la eficiencia sin comprometer la precisión. En este trabajo se utilizó una política de tamaño fijo de los lotes, pero resulta interesante estudiar el efecto del tamaño del *batch* en la velocidad de convergencia y el error final.

Finalmente, se aplicó una estrategia adaptativa para el *learning rate*, usando la regla:

```
step(i) = 0.05f0 / 2**(i/1000),
```

lo que permitió reducir progresivamente la tasa de aprendizaje a medida que avanzaba el entrenamiento. Esta política mostró buen desempeño, aunque su impacto específico en la convergencia merece un análisis más detallado.

En resumen, los resultados obtenidos validan la viabilidad del enfoque propuesto, resaltando la importancia de la regularidad del dominio, la arquitectura de la red y el diseño del entrenamiento. Como líneas futuras de trabajo se propone:

- Estudiar de forma sistemática el impacto de la regularidad geométrica del dominio sobre el rendimiento del método.
- Investigar el efecto del tamaño del *batch* en la eficiencia y precisión del entrenamiento.
- Explorar otras funciones de activación, incluyendo variantes suaves o adaptativas.

- Probar más algoritmos de optimización y combinaciones con esquemas adaptativos de tasa de aprendizaje.
- Experimentar con políticas alternativas para el ajuste dinámico del *learning rate* y su relación con la estabilidad del entrenamiento.
- Extender el análisis a dominios más complejos en dimensiones altas, aprovechando la capacidad computacional de GPUs.

Referencias

[1] R.A. Adams and J.J.F. Fournier. *Sobolev Spaces*. Pure and Applied Mathematics. Academic Press, 2003.

[2] Francisco M. Bersetche and Juan Pablo Borthagaray. A deep first-order system least squares method for solving elliptic pdes. *Computers & Mathematics with Applications*, 129:136–150, January 2023.

[3] Léon Bottou. *On-line Learning and Stochastic Approximations*, page 9–42. Cambridge University Press, January 1999.

[4] Andrea Braides. *Gamma-Convergence for Beginners*. Oxford University Press, July 2002.

[5] Susanne C. Brenner and L. Ridgway Scott. *The Mathematical Theory of Finite Element Methods*. Springer New York, 2008.

[6] Z. Cai, R. Lazarov, T. A. Manteuffel, and S. F. McCormick. First-order system least squares for second-order partial differential equations: Part i. *SIAM Journal on Numerical Analysis*, 31(6):1785–1799, December 1994.

[7] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7), 2011.

[8] David Gilbarg and Neil S. Trudinger. *Elliptic Partial Differential Equations of Second Order*. Springer Berlin Heidelberg, 2001.

[9] Vivette Girault and Pierre-Arnaud Raviart. *Finite Element Methods for Navier-Stokes Equations*. Springer Berlin Heidelberg, 1986.

[10] C. I. Goldstein, Thomas A. Manteuffel, and Seymour V. Parter. Preconditioning and boundary conditions without h_2 estimates: l_2 condition numbers and the distribution of the singular values. *SIAM Journal on Numerical Analysis*, 30(2):343–376, April 1993.

[11] Pierre Grisvard. *Elliptic Problems in Nonsmooth Domains*. Society for Industrial and Applied Mathematics, January 2011.

[12] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer New York, 2009.

[13] Kurt Hornik. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2):251–257, 1991.

[14] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014.

[15] Vladimir G. Maz'ja. *Sobolev Spaces*. Springer Berlin Heidelberg, 1985.

[16] N.J. Nilsson. *Introduction to Machine Learning—An Early Draft of a Proposed Textbook*. 1998.

[17] F. Peñuñuri, K. B. Cantún-Avila, and R. Peón-Escalante. Dual numbers for arbitrary order automatic differentiation, 2025.

[18] Chris Rackauckas. Parallel computing and scientific machine learning (sciml): Methods and applications, 2025.

[19] N. Sukumar and Ankit Srivastava. Exact imposition of boundary conditions with distance functions in physics-informed deep neural networks. *Computer Methods in Applied Mechanics and Engineering*, 389:114333, February 2022.

[20] T. Tieleman. Lecture 6.5-RMSprop: Divide the gradient by a running average of its recent magnitude, 2012.