**TABLA DE CONTENIDOS**

2 INTRODUCCIÓN 2

3 OBJETIVOS DEL PRESENTE AVANCE 2

4 DESCRIPCIÓN DEL PROBLEMA FÍSICO 2

4.1 Estructura de archivos del proyecto 3

5 GENERACIÓN DE DATASET 4

5.1 Generación de condiciones de borde y punto caliente 4

5.2 Resolución del problema y construcción del dataset 4

5.3 Almacenamiento 4

5.4 Automatización 4

6 ENFOQUE DE MACHINE LEARNING 5

7 RESULTADOS PRELIMINARES 5

8 PRÓXIMOS PASOS 5

9 CONSULTAS 5

INTRODUCCIÓN

El presente trabajo se enfoca en el desarrollo de un modelo de Machine Learning (ML) para predecir la distribución de temperaturas en una chapa cuadrada sometida a diversas condiciones de borde (temperatura - flujo) y un punto caliente específico, utilizando simulaciones numéricas generadas por diferencias finitas como “dataset” de entrenamiento.

Este informe presenta una primera aproximación al problema, describiendo el método de generación de datos, el planteo inicial del problema físico, las variables de entrada consideradas y la configuración preliminar del modelo de ML propuesto. Asimismo, se exponen los primeros resultados obtenidos y se detallan los pasos a seguir, junto con las consultas necesarias, para consolidar el modelo de ML en este caso de estudio.

OBJETIVOS DEL PRESENTE AVANCE

Este primer avance tiene como propósito confirmar la viabilidad del enfoque propuesto y establecer las bases para el desarrollo del modelo de ML. Para ello, se plantean los siguientes objetivos específicos:

* Generar y validar un dataset de entrenamiento, asegurando que las condiciones de borde, el punto caliente y las simulaciones por diferencias finitas reflejen adecuadamente el comportamiento térmico de la chapa.
* Implementar un modelo preliminar de ML y ajustar su arquitectura y parámetros de entrenamiento, con el objetivo de obtener predicciones de temperatura con un error controlado.
* Evaluar el desempeño del modelo, analizando la función de pérdida, tiempos de entrenamiento y resultados en casos de prueba.
* Detectar posibles limitaciones del procedimiento actual y proponer mejoras, ya sea en la diversidad del dataset, en la estructura de la red o en las métricas de evaluación.
* Definir próximos pasos y aspectos a consultar, priorizando estrategias para la escalabilidad, optimización y validación del modelo en vistas a su futura aplicación.

DESCRIPCIÓN DEL PROBLEMA FÍSICO

El problema físico consiste en determinar la distribución de temperaturas en una chapa cuadrada de dimensiones conocidas, sometida a condiciones de borde fijas y la presencia de un punto caliente en una posición específica.

Se consideran las siguientes condiciones de borde:

* Temperatura fija (Dirichlet)
* Flujo de calor fijo (Neumann)
* Combinaciones de estas condiciones según cada caso de simulación.

Además, se incluye un punto caliente (que actúa como una fuente de calor puntual) con coordenadas y valor de temperatura asignado, con el objetivo de evaluar su influencia sobre el campo de temperaturas.

El comportamiento térmico de la chapa se modela utilizando la ecuación de conducción de calor en estado estacionario en dos dimensiones:

La resolución se realiza mediante el método de diferencias finitas, discretizando la chapa en una malla de nodos con pasos y determinados según las dimensiones y la resolución buscada.

La resolución del sistema resulta en un conjunto de ecuaciones lineales de la forma donde representa la temperatura en cada nodo de la malla

Los parámetros físicos considerados incluyen:

* Conductividad térmica (k) de la chapa asumida constante.
* Espaciamiento de malla ( y )
* Valores y tipos de condiciones de contorno.
* Posición y temperatura del punto caliente.

GENERACIÓN DE DATASET

Para el entrenamiento del modelo de ML se generó un conjunto de datos mediante simulaciones numéricas resueltas con diferencias finitas, utilizando la función previamente desarrollada.

El procedimiento de generación del dataset se estructuró de la siguiente manera:

## Generación de condiciones de borde y punto caliente

Se implementó una función que genera de forma aleatoria:

* **Condiciones de contorno en cada borde (A, B, C, D)**, seleccionando aleatoriamente si la condición es de **temperatura** o **flujo**, y asignando un valor dentro de un rango físico coherente:
  + **Temperatura:** desde hasta del material seleccionado.
  + **Flujo:** entre .
* **Punto caliente:** asignando una posición aleatoria dentro de la malla y una temperatura aleatoria dentro del rango del material.
* **Material y conductividad térmica:** se selecciona aleatoriamente un material desde una base de datos (materiales.csv) y se toma la conductividad térmica y la temperatura de fusión para ajustar los rangos de simulación.

## Resolución del problema y construcción del dataset

Para cada muestra:

1. Se generan condiciones de borde, el punto caliente y el material (variables móviles).
2. Se resuelve el problema estacionario de conducción de calor en la chapa con usando estas condiciones.
3. Se construye el vector de entrada (X) con las variables móviles
4. Se construye el vector de salida (Y) con la distribución de temperaturas
5. Se registra la información de cada simulación en un archivo de registro (dataset\_variables.csv).

## Almacenamiento

Los datos generados se almacenan en:

* : entradas del modelo
* : salidas del modelo
* : tabla de registro de todas las combinaciones de materiales, condiciones de contorno y punto caliente utilizadas.

## Automatización

El proceso de generación se automatizó mediante la función:

que permite definir:

* El número de muestras a generar.
* El tamaño de malla de la chapa ().
* Los pasos espaciales ().
* La carpeta de guardado.

|  |  |  |  |
| --- | --- | --- | --- |
| (a) | | (b) | |
|  |  |  |  |

***Figura* 1*:*** ***(a)*** Histograma de temperaturas para 1000 muestras sin aplicar filtro, considerando condiciones de flujo y temperatura en el rango de [, ] y []. ***(b)*** Histograma de temperaturas para las mismas 1000 muestras tras aplicar un filtro que descarta aquellas muestras que presentan temperaturas .

ENFOQUE DE MACHINE LEARNING

## Arquitectura y funcionamiento de la red neuronal para predicción de temperatura

Según lo descrito anteriormente, el sistema dispone de 12 variables móviles que pueden variar dentro de diferentes rangos. La salida está compuesta por 2500 nodos, cada uno representando la temperatura de un punto específico de la placa.

Para modelar esta relación, se implementó una arquitectura que recibe las 12 variables de entrada y devuelve las 2500 temperaturas correspondientes. En este caso, se utiliza una red neuronal multicapa (MLP) con tres capas ocultas, pensada para capturar la relación no lineal entre las variables de entrada y la distribución de temperaturas en la placa.

La arquitectura está conformada por:

* Una capa de entrada de 12 nodos, correspondientes a las variables móviles del sistema.
* Una primera capa oculta con 128 neuronas y función de activación ReLU.
* Una segunda capa oculta con 256 neuronas y función de activación ReLU.
* Una tercera capa oculta con 512 neuronas y función de activación ReLU.
* Una capa de salida con 2500 neuronas, correspondiente a cada punto de temperatura de la placa.

Esta estructura se implementa de forma secuencial, empleando funciones de activación ReLU en las capas ocultas buscando modelar de manera eficiente las no linealidades del sistema, mientras que la capa de salida se mantiene lineal para predecir valores de temperatura en una escala continua.

Cada neurona de la red neuronal realiza, de manera individual, el siguiente procedimiento: recibe señales de entrada correspondientes a las variables móviles o a las salidas de la capa anterior, multiplica cada entrada por un peso sináptico que se ajusta durante el entrenamiento y suma un término independiente denominado bias (), que permite ajustar el nivel de activación de la neurona. Sobre este valor resultante se aplica la función de activación ReLU en las capas ocultas, lo que permite a la red capturar relaciones complejas entre las variables de entrada y la salida. Finalmente, la salida de cada neurona se transmite a la siguiente capa repitiendo nuevamente el proceso.

|  |  |
| --- | --- |
| Diagrama, Dibujo de ingeniería  El contenido generado por IA puede ser incorrecto.  (a) | Diagrama  El contenido generado por IA puede ser incorrecto.  (b) |

***Figura* 2**: ***(a)*** Esquema de la arquitectura de la red neuronal multicapa (MLP) utilizada para predecir las temperaturas en cada uno de los 2500 puntos de la placa a partir de las 12 variables móviles del sistema. ***(b)*** Esquema del funcionamiento de una neurona dentro de la red, mostrando la combinación ponderada de las entradas, la adición del término bias y la aplicación de la función de activación

## Configuración de entrenamiento

Para entrenar el modelo descrito en la sección anterior, se utilizaron los “datasets” generados mediante simulaciones por diferencias finitas, divididos en 70% para entrenamiento y 30% para validación, permitiendo evaluar la capacidad de generalización del modelo mientras ajusta sus parámetros.

Con el objetivo de facilitar el aprendizaje y asegurar estabilidad numérica, tanto las variables de entrada como las de salida fueron normalizadas respecto de su media ) y desvío estándar (), de acuerdo con las siguientes expresiones:

De este modo, las entradas y las salidas quedaron centradas en cero y escaladas, favoreciendo la eficiencia en el proceso de entrenamiento.

El entrenamiento de la red neuronal multicapa (MLP) se realizó durante un número definido de épocas, donde en cada época el conjunto de entrenamiento se recorre en lotes de muestras de tamaño fijo (batch size**)**. Este enfoque permite actualizar los parámetros de la red de manera progresiva y estable, evitando los problemas que pueden surgir al utilizar todo el dataset en una única actualización.

En cada iteración sobre un lote de datos:

* La red recibe un conjunto de muestras normalizadas como entrada y calcula las temperaturas de salida estimadas
* Se calcula el error de predicción utilizando como métrica el Error Cuadrático Medio, definido como:

Donde N representa la cantidad de puntos de salida (2500 en este caso) por muestra.

* A partir del calculo de error, se computan las derivadas necesarias para conocer como deben ajustarse los pesos y bias de la red, y se actualizan los parámetros de forma tal de reducir el error en las siguientes iteraciones.

Una vez finalizada cada época, se evalúa el desempeño del modelo sobre el conjunto de validación, registrado nuevamente el MSE para monitorear el progreso del entrenamiento y prevenir el sobreajuste.

# RESULTADOS

## MODELO CON DATASET AMPLIO – SIN PONDERACION

|  |  |
| --- | --- |
|  |  |

***Figura* 3*:*** Evolución del RMSE durante el entrenamiento y la validación del modelo para dos valores de tasa de aprendizaje, mostrando el impacto de este hiperparámetro en la estabilidad y velocidad de convergencia.

|  |  |
| --- | --- |
| (a) | (b) |
| (c) | |

***Figura* 4*:*** ***(a)*** Evolución del RMSE en el conjunto de entrenamiento para distintos tamaños de dataset, mostrando la reducción progresiva del error a lo largo de las épocas. ***(b)*** Evolución del RMSE en el conjunto de validación para los mismos tamaños de dataset, permitiendo evaluar la capacidad de generalización del modelo durante el entrenamiento. ***(c)*** Comparación directa entre los casos de menor y mayor tamaño de dataset, mostrando el efecto de la cantidad de muestras sobre la estabilidad y el valor final del error alcanzado.

## MODELO CON DATASET AMPLIO – CON PONDERACION

|  |  |
| --- | --- |
| (a) | (b) |
| (c) | |

|  |
| --- |
|  |
|  |
|  |
|  |
|  |

Texto

El contenido generado por IA puede ser incorrecto.

|  |  |
| --- | --- |
| (a) | (b) |
| (c) | |

## MODELO CON DATASET AMPLIO – CON PONDERACION

SSSAssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssss

aaaaaaaaaaaaaaaaaaaaaaaaaaaaaa

## MODELO CON DATASET RESTRINIDO

### Dataset

|  |  |
| --- | --- |
|  |  |

Gráfico, Histograma

El contenido generado por IA puede ser incorrecto.

# ANEXO

## Ecuaciones – Método de las Diferencias Finitas

Ecuación diferencial de la Difusión del Calor en estado estacionario

Se parte de la discretización del dominio de la chapa en un conjunto de puntos/nodos que esta distribuidos de forma regular.

Diagrama, Esquemático

El contenido generado por IA puede ser incorrecto.

Figura 5: Discretización de la chapa

Se discretizan las derivadas segundas de (1) y se unifican los índices.

Tomando

Tomando se obtiene:

La ecuación (3) es valida para todos los nodos internos de la chapa.

Aplicando las diferentes condiciones de contorno (temperatura y flujo constante) en los bordes (A, B, C y D) se llegan a las siguientes expresiones:

Borde A:

Si la temperatura es constante:

Si el flujo es constante:

Borde B:

Si la temperatura es constante:

Si el flujo es constante:

Borde C:

Si la temperatura es constante:

Si el flujo es constante:

Borde D:

Si la temperatura es constante:

Si el flujo es constante:

## Registro de prueba, hiperparámetros y archivos asociados

Se presenta un resumen de las diferentes configuraciones estudiadas en este trabajo.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Tipo de entrenamiento | Prueba N° | Muestras | Hiperparámetros utilizados en el entrenamiento | | | | | | | Nombre de archivos asociados | | Observaciones generales |
| lr | Epoch | Batch Size |  |  | RMSE Final [°C] | Observaciones | Dataset | Modelo |
| Modelo con dataset amplio – Sin ponderación | 1 | 1000 | 1e-3 | 1000 | 32 | - | - | 337,8 |  | dataset\_1000\_test | MLP\_1000\_test\_run3 |  |
| 2 | 5000 | 1e-3 | 1000 | 32 | - | - | 146,3 |  | dataset\_5000\_test | MLP\_5000\_test\_run3 |  |
| 3 | 50 000 | 1e-3 | 1000 | 32 | - | - | 52,5 |  | dataset\_50000\_test | MLP\_50000\_test\_run3 |  |
| 4 | 100 000 | 1e-3 | 1000 | 32 | - | - | 29,6 |  | dataset\_100000\_test | MLP\_100000\_test\_run3 |  |
| 5 | 1000 | 1e-4 | 1000 | 32 | - | - | 342,7 |  | dataset\_1000\_test | MLP\_1000\_test\_run4 |  |
| 6 | 5000 | 1e-4 | 1000 | 32 | - | - | 203,0 |  | dataset\_5000\_test | MLP\_5000\_test\_run4 |  |
| 7 | 50 000 | 1e-4 | 1000 | 32 | - | - | 57,5 |  | dataset\_50000\_test | MLP\_50000\_test\_run4 |  |
| 8 | 100 000 | 1e-4 | 1000 | 32 | - | - | 28,0 |  | dataset\_100000\_test | MLP\_100000\_test\_run4 |  |
| 9 | 200 000 | 1e-4 | 1000 | 32 | - | - | 24,7 |  | dataset\_200000\_test | MLP\_200000\_test\_run4 |  |
| 10 | 100 000 | 1e-4 | 2000 | 32 | - | - | 24,5 |  | dataset\_100000\_test | MLP\_100000\_test\_run42 | No se observa una disminución pronunciada de la función de perdida -. No vale la pena aumentar las épocas bajo estas condiciones |
| Modelo con dataset amplio – Con ponderación | 11 | 1000 | 1e-4 | 1000 | 64 | 0.5 | 2 | 333,9 |  | dataset\_1000\_test | MLP\_1000\_test\_run00011 |  |
| 12 | 5000 | 1e-4 | 1000 | 64 | 0.5 | 2 | 200,5 |  | dataset\_5000\_test | MLP\_5000\_test\_run00011 |  |
| 13 | 50 000 | 1e-4 | 1000 | 64 | 0.5 | 2 | 89,3 |  | dataset\_50000\_test | MLP\_50000\_test\_run00011 |  |
| 14 | 100 000 | 1e-4 | 1000 | 64 | 0.5 | 2 | 72,6 |  | dataset\_100000\_test | MLP\_100000\_test\_run00011 |  |
| 15 | 200 000 | 1e-4 | 1000 | 64 | 0.5 | 2 | 69,96 |  | dataset\_200000\_test | MLP\_200000\_test\_run00011 |  |
| 16 | 200 000 | 1e-4 | 1000 | 64 | 0.3 | 2 |  |  | dataset\_200000\_test | MLP\_200000\_test\_run00013 |  |
| 17 | 100 000 | 1e-4 | 1000 | 64 | 0.3 | 1 |  |  | dataset\_100000\_test | MLP\_100000\_test\_run00014 |  |
| 18 | 100 000 | 1e-3 | 1000 | 64 | 0.3 | 1 | 69.9 |  | dataset\_100000\_test | MLP\_100000\_test\_run00015 |  |
| 19 | 100 000 | 1e-3 | 1000 | 64 | 0.2 | 1 | 62.4 |  | dataset\_100000\_test | MLP\_100000\_test\_run00016 |  |
| 20 | 100 000 | 1e-3 | 1000 | 64 | 0.1 | 1 | 47.3 |  | dataset\_100000\_test | MLP\_100000\_test\_run00017 |  |
| 21 | 100 000 | 1e-3 | 1000 | 64 | 0.05 | 1 | 38.8 |  | dataset\_100000\_test | MLP\_100000\_test\_run00018 |  |
| 22 | 100 000 | 1e-3 | 1000 | 64 | 0.025 | 1 | 34.1 |  | dataset\_100000\_test | MLP\_100000\_test\_run00019 |  |
| 23 | 200 000 | 1e-3 | 1000 | 64 | 0.025 | 1 |  |  | dataset\_200000\_test | MLP\_200000\_test\_run00019 |  |
| 24 | 100 000 | 1e-3 | 1000 | 64 | 0.025 | 1 |  | Modificación en la ventana de selección – se consideró un circulo | dataset\_100000\_test | MLP\_100000\_test\_run00019R |  |
| Modelo con dataset restringido | 25 | 100 000 | 1e-3 | 1000 | 64 | - | - |  | La curva de validación presenta oscilaciones marcadas | dataset\_100000\_test\_old | MLP\_100000\_test\_old\_run03 | Convergencia en 250 epoch |
| 26 | 100 000 | 1e-4 | 1000 | 64 | - | - |  | La curva de validación presenta oscilaciones | dataset\_100000\_test\_old | MLP\_100000\_test\_old\_run04 | Convergencia en 700 epoch |