



POLITÉCNICA



**UNIVERSIDAD POLITÉCNICA DE MADRID**  
**ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA**  
**AERONÁUTICA Y DEL ESPACIO**  
**GRADO EN INGENIERÍA AEROESPACIAL**

**TRABAJO FIN DE GRADO**

**Título del Trabajo Fin de Grado**

**AUTOR: Nombre APELLIDOS**

**ESPECIALIDAD: Denominación de la especialidad**

**COTUTOR: Nombre APELLIDOS**

**TUTOR DEL TRABAJO: Nombre APELLIDOS**

**Febrero de 2020**



# Agradecimientos

TITULACIÓN: Graduado en Ingeniería Aeroespacial

DEPARTAMENTO: Materiales y Producción Aeroespacial

TIPO DE TFG: Especial

INTENSIFICACIONES A LAS QUE SE OFERTA: Todas

TÍTULO DEL TRABAJO: Detección y análisis de impactos sobre estructuras de material compuesto con sensores integrados

## **CONTENIDOS Y OBJETIVOS DEL TRABAJO:**

En el marco de las estructuras inteligentes, las estructuras con sensores integrados permiten la determinación de la existencia de daños o de eventos mediante el tratamiento de las señales de los sensores, en lo que se conoce como SHM (Structural Health Monitoring). En este marco, se va a trabajar con una red de sensores PZT integrados en una placa de material compuesto con y sin rigidizadores. Se analizarán los impactos realizados sobre la misma y se estudiará su respuesta para tratar de discriminar no solo la posición del impacto, sino también discriminar la energía del mismo. Adicionalmente se tratará de discriminar entre la masa y la velocidad del impacto y, en caso de que se produzcan daños en la misma (delaminaciones, roturas, etc) cuantificarlos y caracterizarlos.

Con este objetivo, se desarrollarán redes neuronales (ANN-Artificial Neural Networks) que serán entrenadas con resultados experimentales y teóricos. Las ANN son unas excelentes clasificadoras y analizadoras de tendencias, por lo que se propone su uso para el estudio de las señales debidas al impacto.

Este TFG presenta un carácter marcadamente interdisciplinar, ya que abarca desde las estructuras de material compuesto, adquisición de medida mediante sensores y tratamiento de datos.

# Índice general

<b>1. Introducción y motivación</b>	<b>1</b>
<b>2. Estado del arte</b>	<b>3</b>
2.1. Structural Health Monitoring . . . . .	3
2.2. Monitorización de estructuras aeronáuticas - SHM 5 pags (SHM a día de hoy – futuros desarrollos) . . . . .	5
2.2.1. Estructuras aeronáuticas . . . . .	5
2.2.2. Daños en estructuras aeronáuticas . . . . .	5
2.3. Inteligencia Artificial aplicada a SHM . . . . .	7
2.3.1. Representación gráfica del funcionamiento de una Red Neuronal . . . . .	7
2.4. IA aplicado a la monitorización estructural . . . . .	15
2.5. IA aplicada a la monitorización de estructuras aeronáuticas . . . . .	16
2.6. RNN LSTM . . . . .	16
<b>3. Aplicación de Deep Learning en SHM con sensores de deformación</b>	<b>17</b>
3.1. Proyecto INESASSE . . . . .	17
3.1.1. Estructura y red de sensores . . . . .	17
3.1.2. Planteamiento de daños y descripción de ensayos . . . . .	20
3.1.3. Modelo de elementos finitos . . . . .	21
3.2. Pre-procesado de señales . . . . .	22
3.2.1. Ensayo real . . . . .	23
3.2.2. FEM . . . . .	27
3.3. Construcción del modelo de Deep Learning . . . . .	28
3.4. Resultados . . . . .	30
3.4.1. INESASSE . . . . .	30
<b>4. Caracterización de impactos para SHM</b>	<b>34</b>
4.1. Propuesta de ensayo experimental . . . . .	34
4.1.1. Impactador estructura . . . . .	34
4.1.2. Electrónica - Software y hardware . . . . .	34
4.1.3. Adquisición y procesado de datos . . . . .	34
4.2. Caracterización de la posición . . . . .	35
4.3. Caracterización de la energía . . . . .	36
4.4. Caracterización de la velocidad . . . . .	37
4.5. Caracterización completa de un impacto . . . . .	38
<b>5. Conclusiones y trabajos futuros</b>	<b>39</b>
5.1. Conclusiones . . . . .	39
5.2. Trabajos futuros . . . . .	40

# Índice de figuras

2.1.	SHM como sistema realimentado . . . . .	4
2.2.	Problemas asociados a las estructuras de una aeronave . . . . .	6
2.3.	Criterios de diseño [ <b>criterios</b> ] . . . . .	6
2.4.	Representación de las dos clases diferentes en dos colores . . . . .	7
2.5.	Representación del <i>perceptron</i> . . . . .	8
2.6.	Clasificación de datos con una sola neurona . . . . .	8
2.7.	Representación de las capas de una red . . . . .	9
2.8.	Red con dos <i>Hidden layers</i> y dos neuronas en cada capa . . . . .	10
2.9.	Efecto de la función de activación sobre el output de una neurona . . . . .	10
2.10.	Funciones de activación más populares . . . . .	11
2.11.	Combinación de funciones de activación . . . . .	11
2.12.	Combinación de funciones de activación . . . . .	12
2.13.	Resultado final de la clasificación . . . . .	12
2.14.	Influencia de un parámetro en la red . . . . .	13
2.15.	primer pase de backpropagation . . . . .	13
2.16.	Segundo pase . . . . .	14
2.17.	Tercer pase . . . . .	14
2.18.	Cuarto pase . . . . .	14
3.1.	UAV INTA Milano . . . . .	18
3.2.	Fuselaje posterior . . . . .	18
3.3.	Red de sensores FBG . . . . .	19
3.4.	Red de sensores OBR . . . . .	19
3.5.	Esquema de los daños realizados a la estructura . . . . .	20
3.6.	Realización del daño por extracción de remaches . . . . .	21
3.7.	Vista general de la configuración de los ensayos . . . . .	22
3.8.	Modelo de elementos finitos del fuselaje posterior del UAV Milano . . . . .	22
3.9.	Medida de todos los sensores durante el ensayo del Daño 1 con un remache extraído . . . . .	23
3.10.	Detección de los escalones producidos por la carga vertical . . . . .	24
3.11.	Medidas de los sensores FBG en el ensayo preprocesadas . . . . .	24
3.12.	Histograma del escalón 4 kN, 100 % de carga . . . . .	25
3.13.	Campo de deformaciones medido por los sensores FBG para todos los daños	25
3.14.	Diferencia de los campos de deformaciones referenciado con el estado sin daño (FBG) . . . . .	26
3.15.	Campo de deformaciones provocado por el daño 1 con 3 remaches extraídos para todos los casos de carga . . . . .	26
3.16.	Aplicación del filtro Savitzky-Golay a una linea de sensor OBR . . . . .	27
3.17.	Distribución de ruido gausiano con amplitud 1.5 deformaciones . . . . .	28
3.18.	Detalle de las variaciones provocadas por el ruido gausiano . . . . .	28

3.19. Campo de deformaciones medido por el sensor OBR para todos los daños . . . . .	29
3.20. Diferencia de los campos de deformaciones referenciado con el estado sin daño (OBR) . . . . .	29
3.21. Representación de los vectores de deformaciones tras reducir la dimensión de 20 a 2 . . . . .	30
3.22. Matriz de confusión . . . . .	31
3.23. Representación de los vectores de deformaciones tras reducir la dimensión de 20 a 2 . . . . .	32
3.24. Matriz de confusión . . . . .	33

# Índice de tablas

3.1. Definición de daños en el fuselaje . . . . .	20
---	----

## Capítulo 1

# Introducción y motivación

Tanto en el sector aeronáutico como espacial, las aeronaves y sistemas son diseñados siguiendo un método basado en admisibles y reglas de diseño muy conservativas. Con esto se busca que, bajo una larga lista de condiciones de diseño que varían dependiendo del tipo de aeronave, se garantice la integridad estructural y su correcto funcionamiento durante el transcurso de cada de misión.

Estos admisibles de diseño pueden ir desde defectos estructurales complejos de gran importancia en la seguridad del conjunto de la aeronave, como puede ser la pérdida de un motor, hasta defectos con una pequeña influencia estructural pero que, con el tiempo, pueden convertirse en catastróficos, sirviendo de ejemplo la rotura de un pequeño remache.

La pérdida de un motor produce defectos estructurales cuya influencia se manifiesta en toda la estructura, siendo así fácilmente detectable. Sin embargo, cuando un remache se rompe su influencia es local y de pequeña magnitud.

Para poder detectar o cuantificar el tamaño de un defecto en una estructura se comparan magnitudes físicas de la estructura bajo estudio con un estado de la misma estructura sin defectos. La forma más sencilla de hacer esta comparación es realizando una inspección visual, pero este procedimiento solo permite detectar defectos que han producido cambios grandes y evidentes sobre la estructura, quedando ocultos aquellos que no se detectan a simple vista.

Al ser necesarias inspecciones más exhaustivas para detectar defectos se realizan mantenimientos preventivos. Durante los mantenimientos se utilizan técnicas de inspección no destructivas (NDI) con las que se recogen datos que se procesan para asegurar la integridad estructural de la aeronave.

Sin embargo, durante los últimos años, y como una evolución de las técnicas NDI, ha habido un gran desarrollo de sistemas cuya función es evaluar el estado de las estructuras de una aeronave o vehículo espacial durante la operación y a tiempo real mediante el uso sensores integrados en las propias estructuras. Estos son los llamados *Structural Health Monitoring Systems* (SHMS).

La utilización de estos sistemas tiene consecuencias muy importantes. Permiten la monitorización de las estructuras sin la intervención de inspectores y sin tener que desensamblarla para analizarla. Esto hace que el tiempo entre la detección de un defecto su reparación sea muy corto, haciendo que este no aumente y evitando que termine provocando un fallo catastrófico. Entonces, si el SHMS no detecta ningún tipo de defecto, se llega a la conclusión de que la aeronave no necesita ser revisada en profundidad con tanta

frecuencia y los períodos de mantenimiento pueden ser separados en el tiempo y, mientras tanto, la aeronave sigue operando cumpliendo con los admisibles de diseño.

Por otra parte, la tendencia del sector aeroespacial desde hace varias décadas ha sido aumentar el porcentaje en peso del avión fabricado con materiales compuestos. Los materiales compuestos tienen unas propiedades específicas muy superiores a metales como el aluminio, por otro lado, presentan modos de fallo más complejos, variados y más difíciles de detectar comparados a los que sufren los materiales metálicos.

El aumento del uso de materiales compuestos, junto con el potencial de los SHMS ha sido el impulsor de este Trabajo Fin de Grado. El objetivo es desarrollar herramientas fiables basadas en algoritmos de Inteligencia Artificial (IA) para poder detectar defectos en estructuras aeronáuticas, más concretamente, en estructuras aeronáuticas con geometría compleja fabricadas con material compuesto.

Grandes empresas como Google, Facebook, Microsoft o Nvidia han invertido mucho esfuerzo en el campo de la Inteligencia Artificial (IA) durante los últimos años y han conseguido increíbles resultados, como por ejemplo el NVidia RTX Voice, un cancelador de ruido basado en algoritmos de Deep Learning (DL).

Siguiendo esta línea de investigación en DL, se va a explorar la posibilidad de utilizar este tipo de algoritmos de tratamiento de datos para desarrollar dos herramientas de SHM. La primera tendrá que ser capaz de clasificar diferentes estados de una estructura basándose en medidas de deformación. A su vez, se pretende detectar el nivel de carga al que está sometida la estructura y a la temperatura que se encuentra.

Por otra parte, se desarrollará una segunda herramienta para localizar impactos y cuantificar la energía de estos en una costilla del A380. Esta herramienta se alimentará con la onda mecánica producida por el objeto impactador.

El lado negativo del uso de IA para SHM es que se requiere un Data Set muy grande para poder entrenar los algoritmos y conseguir un nivel de confianza elevado en los resultados. Usando de ejemplo la herramienta que detectará impactos y su energía, se va a calcular el orden de magnitud de los impactos que son necesarios para tener una cantidad de datos aceptable.

Dividiendo la placa en una malla de 10 x 10, seleccionando 10 niveles de energía diferentes y 100 repeticiones de cada impacto llegamos a tener una cantidad mínima de  $10^5$  impactos, lo cual es inviable para realizar de forma manual y rigurosa, por lo que su automatización es necesaria.

Para esta automatización se va a proponer el diseño de un Impactador por gravedad de control numérico. Con esta máquina se realizarán todos los impactos que sean necesarios de forma automática, a la vez que almacenará y preprocesará los datos para alimentar la herramienta de DL.

## Capítulo 2

# Estado del arte

En este capítulo se va a hacer una introducción teórica a los pilares en los que se asienta este trabajo, SHM e IA, y sus aplicaciones en la industria aeroespacial..

### 2.1. Structural Health Monitoring

Structural Helath Monitoring (SHM) es el proceso de identificar daños en estructuras de forma no destructiva mediante el uso de sensores integrados [SHM·Aero·3]. Por lo tanto, es necesario tener el concepto de daño bien definido. Se puede definir un daño como los cambios en el material y/o en las propiedades geométricas de un sistema estructural, incluyendo cambios en las condiciones de contorno y conectividad del mismo, que pueden afectar de manera adversa al funcionamiento presente y futuro del sistema [dam].

La gran ventaja de este proceso frente a las NDI de los que evoluciona es que se puede aplicar en tiempo real, incluyendo durante la operación del vehículo. Con esto se consigue, no solo tener un conocimiento total de la salud estructural, sino que también se puede llegar a monitorizar la magnitud de las cargas a las que las estructuras instrumentadas están sometidas.

El SHM se realiza en cuatro pasos principales:

1. **Evacuación operacional:** para aplicar la técnica de SHM, la evaluación operativa es el paso fundamental. Estudia los problemas y sus consecuencias que conducen a monitorizar las estructuras. La evaluación estructural trata de buscar solución a varias preguntas, tales como:
  - ¿qué tipo de daño se busca en la estructura monitorizada?
  - ¿a qué condiciones ambientales y operativas está sometido la estructura monitorizada mientras se lleva a cabo el proceso de SHM?
  - ¿Cuáles son las restricciones de la transferencia de datos a través del sistema SHM a causa de la operación del vehículo?
  - ¿cuáles son las ventajas del SHM en la vida segura del vehículo?
2. **Adquisición de datos:** el proceso de adquisición de datos es una parte crucial en SHM y se realiza a través de sensores integrados en la estructura. El tipo de sensores que se utilizarán está ligado a la técnica de SHM usada. En esta fase se estudia el número de sensores usados, su distribución en la estructura y el hardware que almacenará esta información.

3. **Procesado de señales y extracción de características debidas al daño:** la parte más crítica del SHM es extraer las inferencias a partir de las señales recogidas por los sensores. Este proceso se basa en descubrir qué propiedades y áreas son afectadas por el daño junto con como varían a medida que el daño evoluciona.
4. **Modelado estadístico para evaluación de las características:** el desarrollo de un modelo estadístico es el paso final del SHM. Este proceso incluye dos categorías:
  - Cuando los datos recopilados pertenecen tanto a partes dañadas como a no dañadas, el modelado estadístico hace una clasificación general también conocida como aprendizaje supervisado, como puede ser el análisis de regresión y clasificación grupal.
  - Cuando los datos proceden únicamente de estructuras dañadas, se habla de aprendizaje no supervisado.

Desde un punto de vista matemático, los sistemas SHM se pueden considerar como sistemas de control realimentado, diferenciándose de éstos en los tiempos característicos y el modo de actuación sobre la planta.

Las escalas temporales dependen de lo que el sistema busca controlar. En el caso de una grieta, los tiempos característicos son del orden de la vida operativa de la planta, en cambio, si se quiere un sistema de localización y detección de impactos, el tiempo característico se reduce a decenas de ms.

También cambian las características espaciales, dependiendo fundamentalmente de la criticidad del daño. La longitud característica será grande cuando la tolerancia al daño permita tamaños considerables, sin embargo, en las zonas donde ésta tolerancia sera pequeña, la longitud característica también lo tendrá que ser [Jaime·Tesis].

Para funcionar, un sistema SHM medirá todas las entradas y salidas que afectan a la planta monitorizada, entendiéndose como planta aquella cuya función sea soportar o transmitir cargas, siendo fija o móvil. Los parámetros de observación serían los parámetros característicos de operación propios y los relativos al entorno. La situación anómala será aquella que comprometa el funcionamiento presente y futuro de la planta, en su conjunto, dentro de las condiciones de diseño [civil].

Así pues, tenemos que el SHM es en esencia un sistema autónomo para inspeccionar y detectar daños en estructuras con una mínima intervención humana. En la Figura 2.1 se puede ver representada esta idea [Jaime·Tesis].

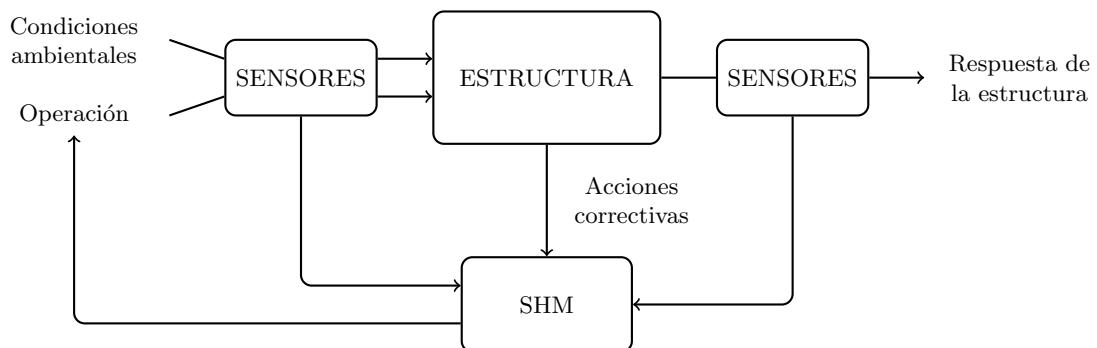


Figura 2.1: SHM como sistema realimentado

## **2.2. Monitorización de estructuras aeronáuticas - SHM 5 pags (SHM a día de hoy – futuros desarrollos)**

Dentro de la gran variedad de estructuras en las que se puede aplicar el SHM, este trabajo se va a centrar en estructuras aeronáuticas fabricadas con material compuesto. Como ya se ha introducido en el primer paso del SHM, *Evaluación operacional*, hay una serie de preguntas que se necesitan responder para elegir el sistema óptimo para monitorizar una estructura determinada dentro de todas las que componen una aeronave.

Será necesario dar respuesta a las siguientes cuestiones:

1. ¿Qué estructura se quiere monitorizar?
2. ¿Qué tipo de daño afecta a la integridad de dicha estructura?
3. ¿Cuál va a ser la característica sensitiva al daño, o Damage Sensitive Feature (DSF), la que se va a usar? El DSF tendrá que ser alguna característica o parámetro medible de la estructura que sea modificado por la presencia del daño.
4. ¿De qué modo el daño influye en la DSF?

Una vez que todas estas preguntas tengan respuesta, se tendrá toda la información necesaria para elegir una estrategia global de SHM apropiada para el objetivo que se ha fijado. A continuación, se van a responder estas cuestiones para el caso de estructuras aeronáuticas.

### **2.2.1. Estructuras aeronáuticas**

No todas las estructuras que componen una aeronave están sometidas al mismo tipo de cargas y, por lo tanto, no presentan los mismos problemas ni se usan los mismos criterios de diseño para toda la aeronave.

En aeronáutica, la inmensa mayoría de partes se diseñan a fatiga. Esto quiere decir que, para un número determinado de ciclos y con un espectro determinado de cargas, la pieza o estructura no compromete la integridad global de la aeronave.

Dentro del diseño a fatiga hay otras dos tendencias, el diseño a vida segura o a tolerancia al daño. En las estructuras diseñadas bajo el criterio de vida segura garantizan que, mientras no se supere el espectro de carga, la integridad estructural no peligra. En cambio, el diseño de tolerancia al daño tiene en cuenta que en el propio proceso de fabricación se generan grietas en el material y asume que estas grietas van a crecer, por lo que toma medidas para asegurar que no se produzcan fallos catastróficos [**criterios**].

Usando las Figuras 2.2 y 2.3 se puede tener una idea general de la estructura bajo estudio y responder a la primera pregunta. El siguiente paso será diferenciar entre los distintos daños que pueden sufrir estas estructuras.

### **2.2.2. Daños en estructuras aeronáuticas**

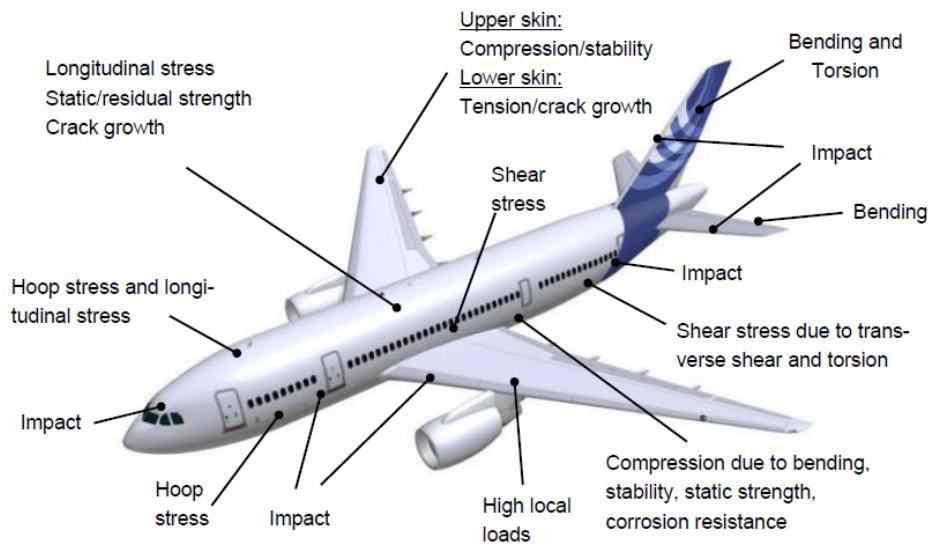


Figura 2.2: Problemas asociados a las estructuras de una aeronave

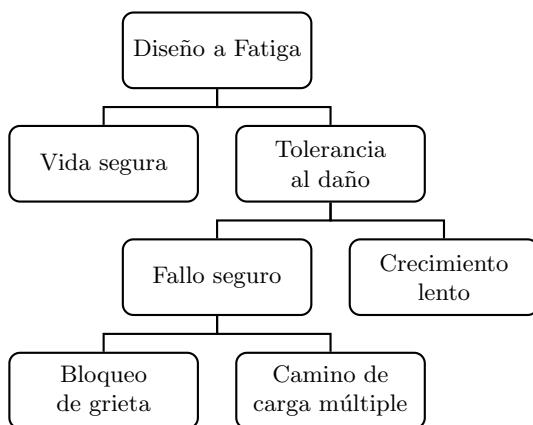


Figura 2.3: Criterios de diseño [criterios]

## 2.3. Inteligencia Artificial aplicada a SHM

2-3 pags

1. Explicar que es la inteligencia artificial y los tipos que hay
2. Diferencias que hay entre Machine learning y Deep learning y por qué decidimos que el Deep Learning es mejor para nuestro objetivo
3. Comentar las distintas redes neuronales dentro de DL y argumentos para elegir las Recurrent Neural Networks - RNN
4. Ejemplos de IA aplicada en SHM

### 2.3.1. Representación gráfica del funcionamiento de una Red Neuronal

Aunque el campo del DL lleva unos años en auge, para la mayoría de usuarios sigue siendo una simple herramienta, es desconocido cómo funciona y mucho menos se tiene una representación gráfica de qué hace y cómo llega a los resultados.

Dado que este trabajo se va a centrar en tecnología de DL, es conveniente hacer una breve explicación de cómo una NN es capaz de clasificar diferentes grupos de datos. Para ello, esta explicación se ha basado en el gran trabajo realizado por el equipo de *TensorFlow - playground* [[TensorFlow playground](#)]. Un entorno perfecto para la visualización del comportamiento de los diferentes parámetros que componen una Red Neuronal.

Lo primero de todo es saber la forma de los argumentos de entrada que, como para casi todo algoritmo, sus *inputs* son números. Dependiendo del problema que se busque resolver estos datos tendrán una forma u otra, más complejos o más simples, pero en esta introducción se va a utilizar un dataset de 2 dimensiones ( $x_1, x_2$ ) y dos clases diferentes (azul y naranja).

Los datos que se van a intentar clasificar están representados en la Figura 2.4

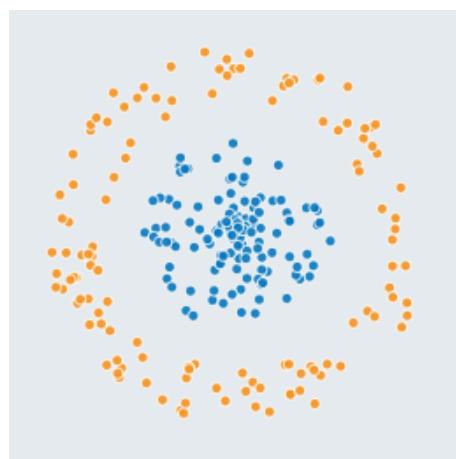


Figura 2.4: Representación de las dos clases diferentes en dos colores

Una vez se conocen los dos grupos que se busca clasificar se va a comenzar la descripción de los elementos que componen una NN y como interactúan entre ellos. Estos elementos son:

### • Neurona

Es la unidad básica de procesamiento que compone una Red Neuronal. Estas neuronas tienen conexiones a través de las que reciben los valores de entrada y realizan una suma ponderada con ellos. Cada una de las entradas es multiplicada por un valor, llamado peso, que definirá a cual de los valores de entrada se le da más importancia. A este cálculo se le añadirá un sesgo o *bias*, dicho de otro modo, se le sumará un escalar.

Si se observa detenidamente la Figura 2.5 (primera neurona llamada *perceptron*) se puede ver que, matemáticamente, una neurona es equivalente a una regresión lineal ( $w_1 \cdot x_1 + w_2 \cdot x_2 + b = 0$ ).

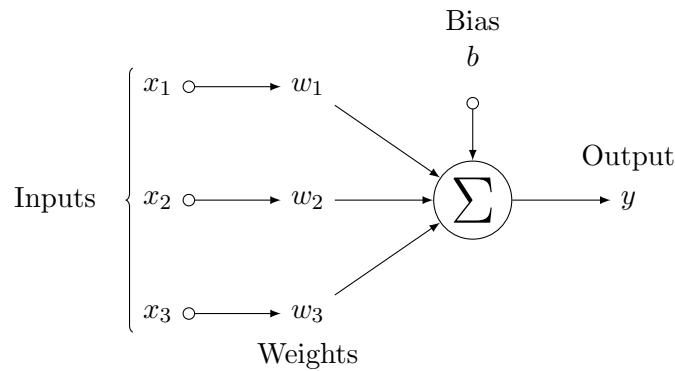


Figura 2.5: Representación del *perceptron*

En una regresión lineal bidimensional, ajustando los pesos ( $w_1, w_2$ ) se modifica la recta que separa el plano ( $x_1, x_2$ ) en dos regiones. En el ejemplo que se está resolviendo, por muy bien que se ajusten estos pesos, con una neurona solo se puede llegar a separar el plano en dos regiones, como se ve en la Figura 2.6.

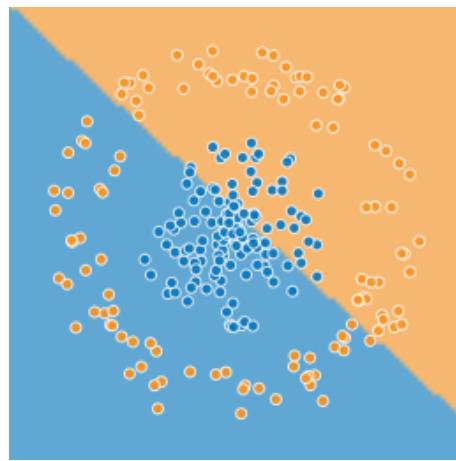


Figura 2.6: Clasificación de datos con una sola neurona

Para definir que región del plano es asignada a la clase azul o naranja se utiliza la

siguiente expresión:

$$w_1 \cdot x_1 + w_2 \cdot x_2 + b = y \quad \begin{cases} y \geq 0, & \text{naranja} \\ y < 0, & \text{azul} \end{cases} \quad (2.1)$$

Por lo tanto, la neurona se puede sintetizar una función. Como una neurona no consigue separar las dos regiones de puntos de forma efectiva, hace que sea necesario dar un paso más.

### • Capas

Para modelizar conocimiento complejo no es suficiente con utilizar una única neurona, es necesario concatenar varias de ellas. Una forma de colocar las neuronas sería una debajo de otra en forma de columna, lo que se define como, en la misma capa.

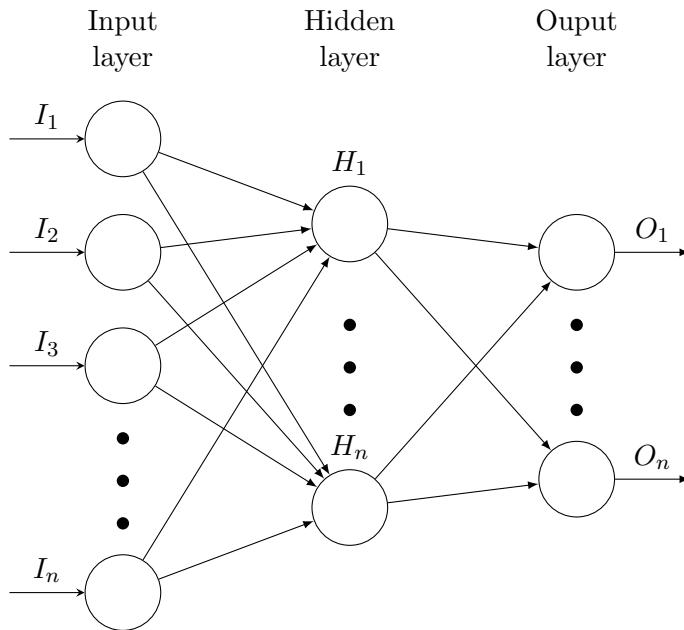


Figura 2.7: Representación de las capas de una red

Como se representa en la Figura 2.7, las capas se pueden concatenar de manera secuencial, una detrás de otra y dependiendo de dónde estén colocadas tendrán un nombre u otro. La capa que recibe primero los argumentos de entrada se llama *Input layer*, la que calcula el argumento de salida se llama *Output layer* y todas las capas que estén entre estas dos, se llamarán *Hidden layers*.

La concatenación secuencial de las capas hace que la red sea capaz de aprender conocimiento jerarquizado y es lo que da nombre al Deep Learning (Aprendizaje Profundo). Dicho de otro modo, si a la red se le introduce una imagen de un coche, es capaz de aprender que un coche no es un único objeto, una capa sintetiza lo que es una rueda, la siguiente lo que es una puerta y las últimas capas son capaces de juntar toda esa información para construir la idea de coche como un conjunto de objetos.

Las neuronas dentro de la misma capa lo que permiten es diversificar el conocimiento. Cada neurona se especializa en una parte de la información ajustando los pesos para dar más importancia a un canal de entrada u otro.

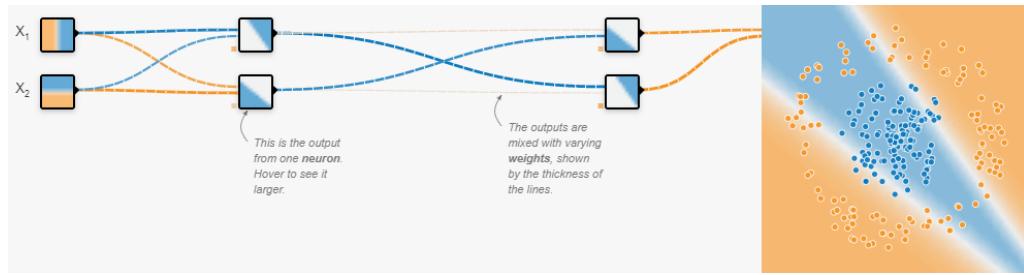


Figura 2.8: Red con dos *Hidden layers* y dos neuronas en cada capa

La aplicación de lo explicado anteriormente se representa en la Figura 2.8. Se puede observar en la caja que representa cada neurona que, en esencia, todas hacen una regresión lineal y que no pueden crear una frontera circular para separar el espacio. Esto se debe a que la operación matemática de concatenación de regresiones lineales termina colapsando en una regresión lineal, no puede dibujar curvas. Para conseguirlo, es necesario añadir un elemento más.

#### • Función de activación

La necesidad de aplicar una distorsión no lineal a la salida de las neuronas convierte en indispensables a las funciones de activación. Con ellas se podrá encadenar de forma efectiva la computación de varias neuronas.

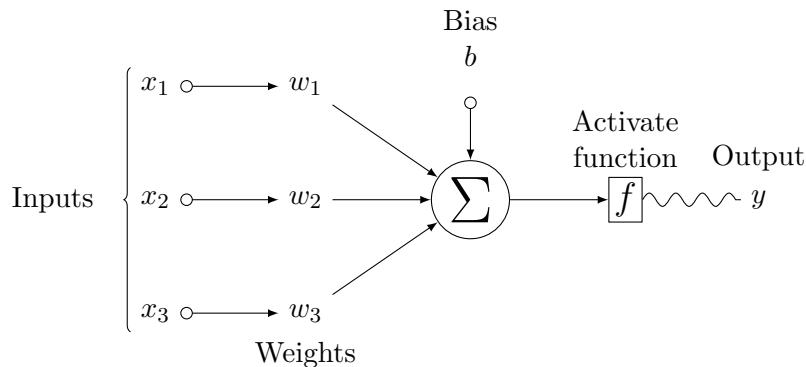


Figura 2.9: Efecto de la función de activación sobre el output de una neurona

En cierto modo ya se estaba usando una función de activación antes de nombrarlas. En la Ecuación 2.1 se había usado la función escalón para separar las dos regiones del plano, pero esta función no provoca no linealidades y no es muy efectiva cuando se requieren fronteras complejas.

Las funciones de activación más conocidas están representadas en la Figura 2.10. Su función principal es hacer que los valores muy grandes o muy pequeños se saturen a determinado valor, dependiendo de la función elegida.

Para ver el efecto de distorsión geométrica que tienen estas funciones sobre el *output* de las neuronas se usa de ejemplo la Figura 2.11. La nueva función de salida se convierte en una superficie, continua y derivable, que para cada *input*  $(x_1, x_2)$  le corresponde una altura  $y$  entre 0 y 1.

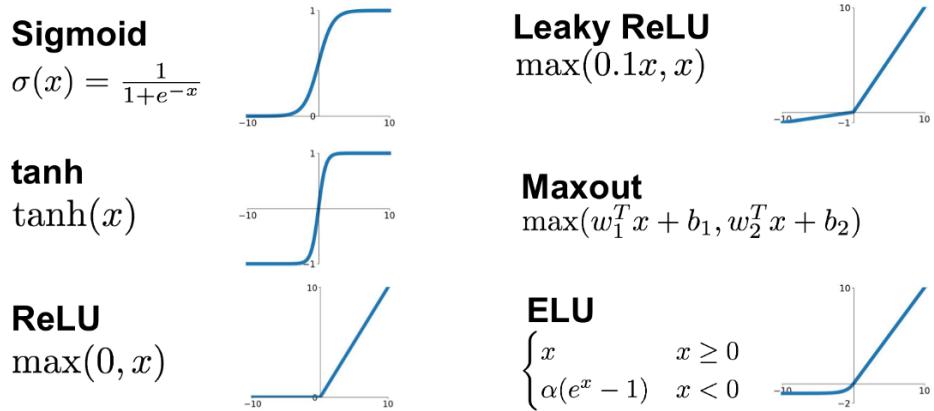


Figura 2.10: Funciones de activación más populares

Tal y como se ha hecho anteriormente, se selecciona un valor umbral de  $y$  a partir del cual se clasificarán los puntos como azules o naranjas. En la Figura 2.11 este valor umbral se representa como un plano a una altura de 0,5, por lo tanto, los puntos  $(x_1, x_2)$  que tras ser evaluados en la neurona saquen un valor de  $y$  inferior a 0,5 se clasificarán como *naranjas* y los que den un valor superior o igual a 0,5 se clasificarán como *azules*. Esto recuerda a la Figura 2.6.

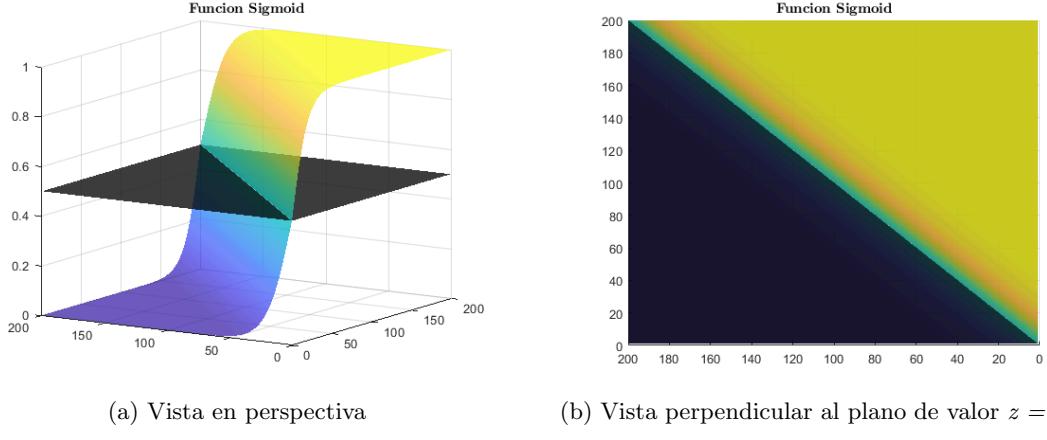


Figura 2.11: Combinación de funciones de activación

Ahora ya se puede utilizar una *hidden layer* con varias neuronas de forma efectiva. La combinación de varias neuronas hace que se sumen superficies (funciones) como la representada en la Figura 2.11, pero con diferentes orientaciones, generando superficies más complejas, Figura 2.12. Igual que antes, los puntos  $(x_1, x_2)$  que tras ser evaluados tienen un  $y$  mayor o igual a 0,5 pertenecerán a la clase azul y los menores a la naranja.

Finalmente se puede ver en la Figura 2.13 la aplicación de todo lo explicado. Gracias a las funciones de activación se consiguen las no linealidades necesarias para generar la superficie que separa exitosamente las dos clases de datos diferentes.

Las neuronas de la Figura 2.13 están unidas mediante unas líneas discontinuas de

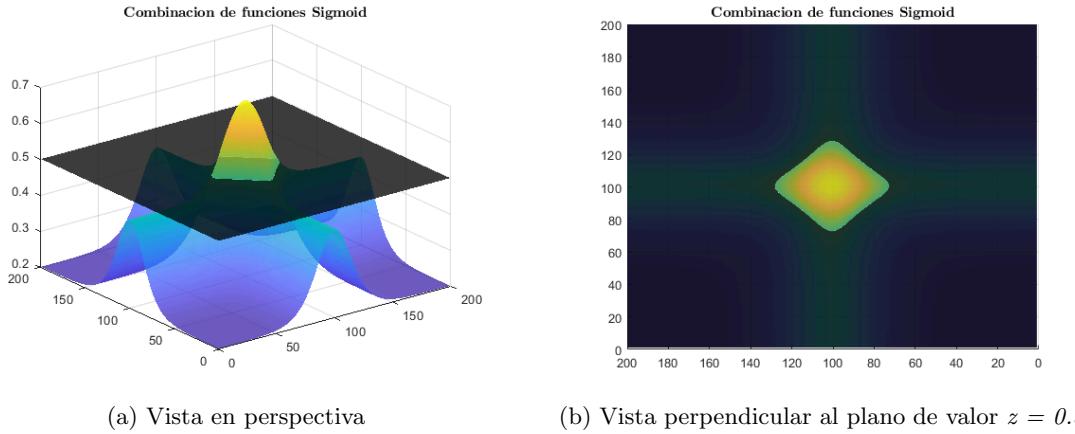


Figura 2.12: Combinación de funciones de activación

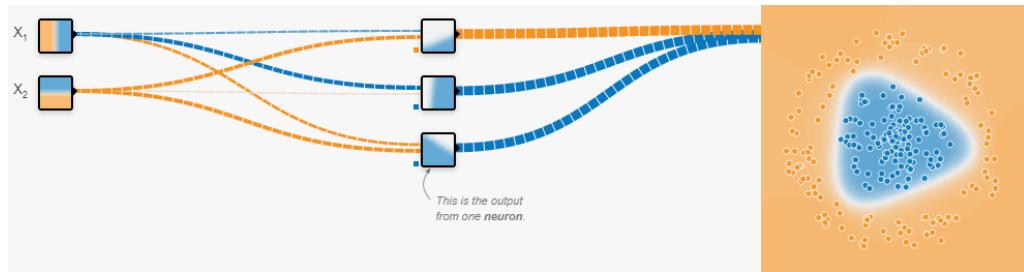


Figura 2.13: Resultado final de la clasificación

diferente color y grosor. Esto representa qué pesos son mayores y que zona perturban más. Los pesos son los elementos que ajustan las funciones de activación para conseguir la superficie final que mejor realice la clasificación. Pero, ¿cómo se realiza el ajuste de los pesos?

#### • Entrenamiento

Se llama entrenamiento al proceso de aprendizaje automático que utilizan las redes neuronales para ajustar sus parámetros internos. El algoritmo que hizo posible que la optimización de parámetros en DL fuera eficiente se llama *backpropagation* y es el responsable del final del *Invierno de la Inteligencia Artificial*

Para llevar a cabo el proceso de optimización es necesario calcular las derivadas parciales de cada uno de los parámetros de la red con respecto a la función de error. Una vez se consigan las derivadas, se utilizará el algoritmo de descenso del gradiente para variar los parámetros y minimizar la función de error.

La función de error ( $E$ ) puede ser, por ejemplo, la suma de las distancias de cada punto a la frontera de su clase. Volviendo a la Figura 2.6, los puntos azules que se encuentran en la región naranja aumentarán mucho el valor de la función de error, mientras que los que están en la región azul no aumentarán casi su valor. La variación de la función de error respecto a un pequeño cambio los parámetros se representa matemáticamente como el gradiente gradiente:  $\frac{\partial E}{\partial w}$ .

Como se ve en la Figura 2.14, el peso asociado a una de las primeras conexiones afecta prácticamente todas las conexiones posteriores y por lo tanto, también influirán

en su gradiente. Esto hace que el aprendizaje de este primer peso sea especialmente complicado y convierte en esencial la utilización del *backpropagation* para el proceso de optimización.

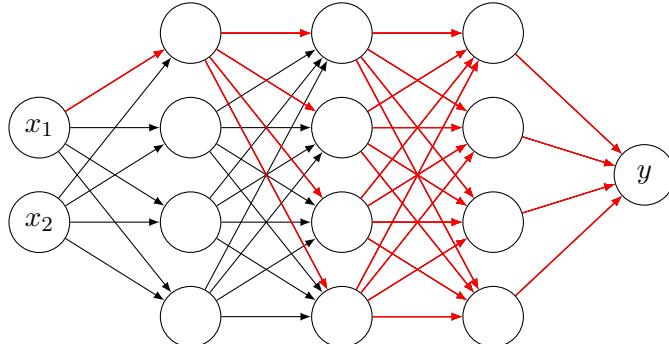


Figura 2.14: Influencia de un parámetro en la red

Cuando la red clasifica erróneamente un *input* genera un valor alto de la función de error. En una NN, el error de las últimas capas depende directamente de las capas anteriores, es decir, si se detecta que una neurona no tiene influencia sobre el error, las capas posteriores a ella tampoco lo tendrán. En esto consiste el *backpropagation*, en la retropropagación de errores, lo que hace muy eficiente el proceso. En la Figura 2.15 se puede ver el inicio del algoritmo partiendo de la última capa.

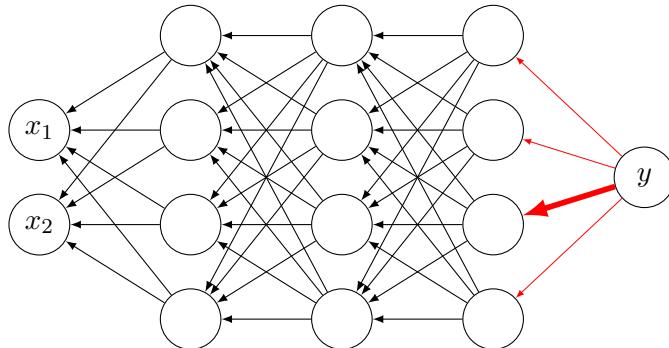


Figura 2.15: primer pase de backpropagation

Esta distribución de errores se utiliza para actualizar los pesos y bias de cada una de las neuronas.

Ahora que se han imputado los errores a las neuronas de la última capa se puede proceder a repetir el mismo proceso de antes como si este fuera el error final de la red, como si esta fuera ahora la última capa. Se va avanzando capa tras capa moviendo el error hacia atrás.

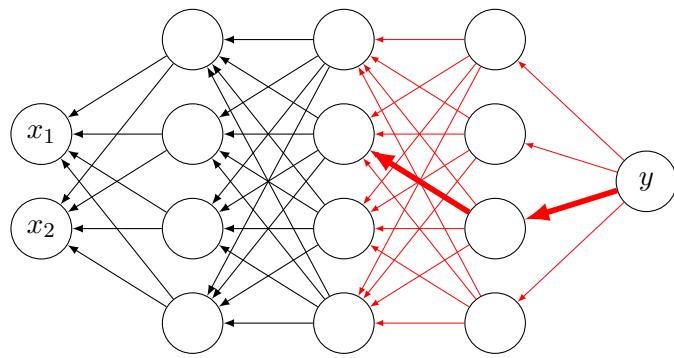


Figura 2.16: Segundo pase

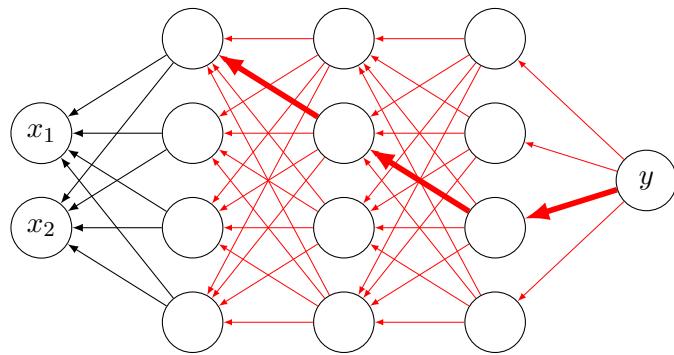


Figura 2.17: Tercer pase

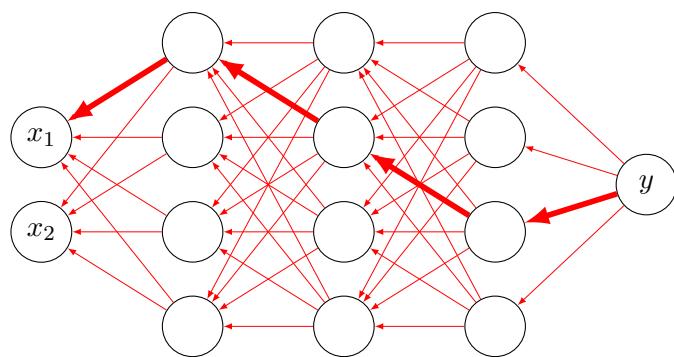


Figura 2.18: Cuarto pase

## **2.4. IA aplicado a la monitorización estructural**

TODOS los sectores menos aeronáutica - entre 5 y 10 pags

Poner un estado actuar de como se esta aplicando la IA a la monitorización estructuras en general

## **2.5. IA aplicada a la monitorización de estructuras aeronáuticas**

INESASSE (5 pags VER TFM Jaime García SOLO INTRODUCIR)

Dar una vuelta al apartado anterior centrándolo sobre todo en estructuras aeronáuticas. Si se puede, poner ejemplos de estructuras de material compuesto e inferir en la complejidad que dan frente a las metálicas por su direccionalidad de propiedades.

## **2.6. RNN LSTM**

Esplicación completa y detallada

## Capítulo 3

# Aplicación de Deep Learning en SHM con sensores de deformación

### 3.1. Proyecto INESASSE

El Proyecto INESASSE (Integración y Explotación de Sistemas de Autodiagnóstico y Supervisión de Salud Estructural en aviones no tripulados) tiene como objetivo principal investigar la capacidad de implantación, en condiciones operativas, de sistemas de supervisión estructural (SSE), empleando redes de sensores de fibra óptica.

Este proyecto de investigación fue realizado por el INTA (Instituto Nacional de Técnica Aeronáutica), en conjunto con la ETSIAE (Escuela Técnica Superior de Ingeniería Aeronáutica y del Espacio), y consta de 3 fases:

1. Elaboración de un modelo de elementos finitos (FEM) del sistema estructural, a través del cual se establece la morfología de la red de sensores. Sobre este FEM se simularán distintos daños producidos sobre la estructura, a fin de analizarlos mediante distintas técnicas SHM y determinar la posición óptima de la red sensorial.
2. Fabricación de dos estructuras idénticas de CFRP que se instrumentaran con la red de sensores definida en la fase 1, integrables en el UAV en cuestión. Tras esto, a una de estas estructuras se le realizará una campaña de ensayos en tierra a fin de validar los FEM, además de comparar los resultados obtenidos mediante herramientas SHM sobre el modelo teórico y el real. Al igual que en el FEM, se realizarán los mismos daños estructurales controlados.
3. En la última fase del proyecto, la estructura fabricada anteriormente que no fue ensayada, se integrará en un UAV en operaciones de vuelo reales, procesando datos de los sensores integrados en la fase 2, e investigando las capacidades del sistema SHM seleccionado para el vuelo.

El vehículo aéreo no tripulado que protagoniza este proyecto es el UAV Milano, con una masa en servicio de 900 kg y una autonomía de hasta 20 horas. En la Figura 3.1 se puede apreciar una imagen del mismo.

#### 3.1.1. Estructura y red de sensores

La estructura de la aeronave sobre la que se ha realizado el FEM y se ha fabricado después es el fuselaje posterior del UAV Milano. Esta estructura está compuesta por:



Figura 3.1: UAV INTA Milano

- Revestimiento inferior con un larguero longitudinal en T dividido en dos y refuerzos sobre las pestañas. Fabricado con cinta UD fuera de autoclave.
- Revestimiento superior con dos largueros en L y refuerzos sobre las pestañas. Fabricado con cinta UD fuera de autoclave.
- Cuatro cuadernas con refuerzos sobre las mismas. Fabricadas con cinta UD fuera de autoclave.
- Dimensiones aproximadas de 2600x850x890 mm.

En la Figura 3.2 se puede ver el diseño explosionado de los componentes descritos.

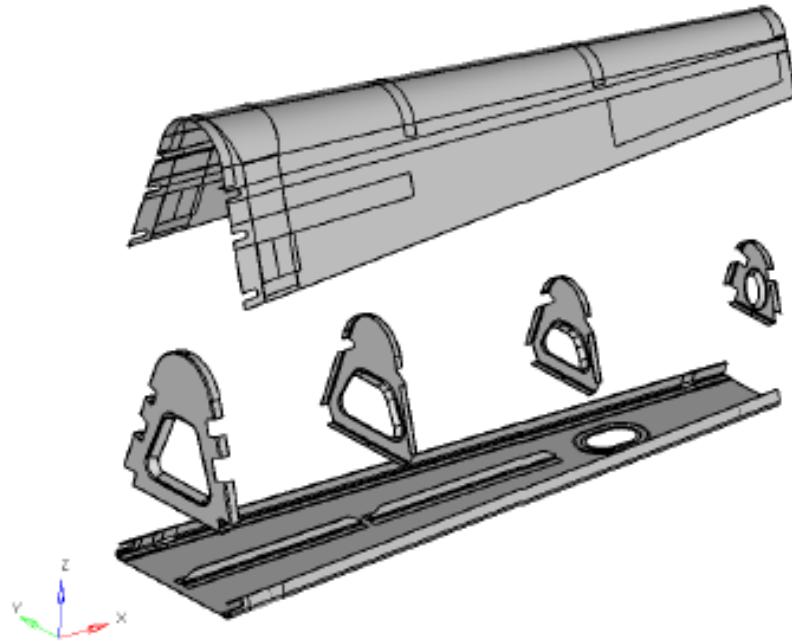


Figura 3.2: Fuselaje posterior

Sobre esta estructura se van a distribuir dos redes de sensores de fibra óptica, uno formado por sensores FBG y otro por OBR. Con ellos se recogerá el campo de deformaciones

de la estructura que se procesará para obtener la clasificación de los daños. En las Figuras 3.3 y 3.4 se tiene una representación gráfica de la posición de estas redes en la estructura.

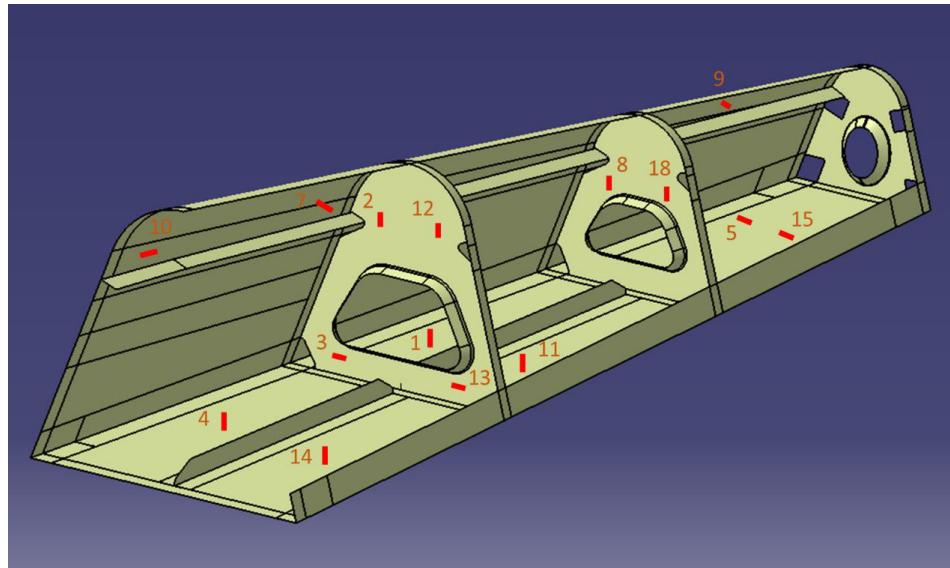


Figura 3.3: Red de sensores FBG

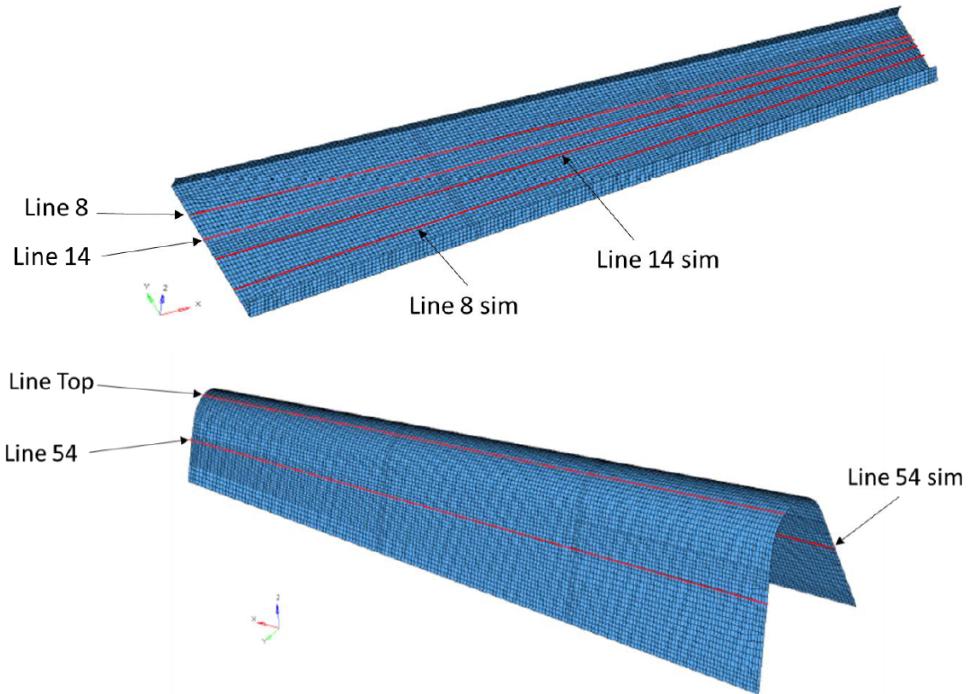


Figura 3.4: Red de sensores OBR

Es importante indicar que, puesto que los sensores de medida distribuida son unidireccionales, se han obtenido las deformaciones en la dirección longitudinal de la fibra (tracción/compresión).

### 3.1.2. Planteamiento de daños y descripción de ensayos

El objetivo de este proyecto es tener un sistema SHM para detectar daños en un fuselaje de material compuesto durante su operación, por lo tanto, se necesita definir la localización de estos daños y sus tamaños.

Estos daños corresponden con desencolados parciales o totales en las zonas que se definirán a continuación. En la Figura 3.5 se puede ver la localización de cada daño en la estructura y en la Tabla 3.1 se pueden ver sus características.

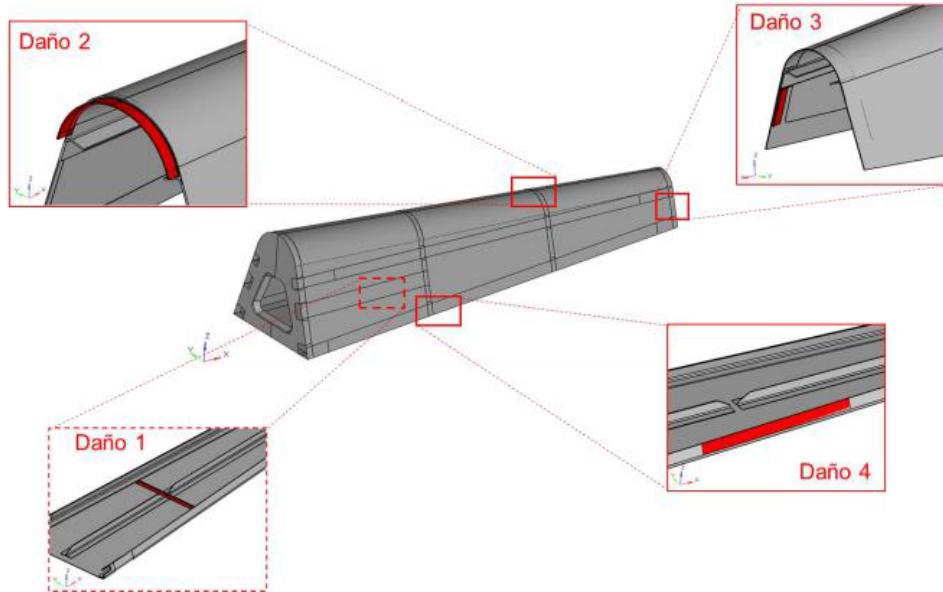


Figura 3.5: Esquema de los daños realizados a la estructura

DAÑO	TIPO DE DAÑO	ELEMENTOS AFECTADOS	LONGITUD MÁXIMA	INCREMENTO
D1	Desencolado parcial del pie de la cuaderna	Cuaderna 2, revestimiento inferior	450 mm	30 mm
D2	Desencolado parcial de la pestaña superior de la cuaderna	Cuaderna 3, revestimiento superior	350 mm	25 mm
D3	Desencolado parcial de la pestaña lateral de la cuaderna	Cuaderna 4, revestimiento superior	180 mm	30 mm
D4	Desencolado parcial entre el revestimiento inferior y el revestimiento superior	Revestimiento inferior, revestimiento superior	900 mm	30 mm

Tabla 3.1: Definición de daños en el fuselaje

Es importante resaltar que, aunque el ensayo estructural y las simulaciones se realizaron para los cuatro daños, se llegó a la conclusión de que el daño 3 era irrelevante tras ver los resultados obtenidos. Por lo tanto, dicho daño fue descartado.

Para facilitar el trabajo a los operarios que realizaron los ensayos del fuselaje, se sustituyó la unión adhesiva por uniones remachadas. Esto permitió controlar mejor la progresión de los daños y también que fueran reparados tal y como se ve en la Figura 3.6.

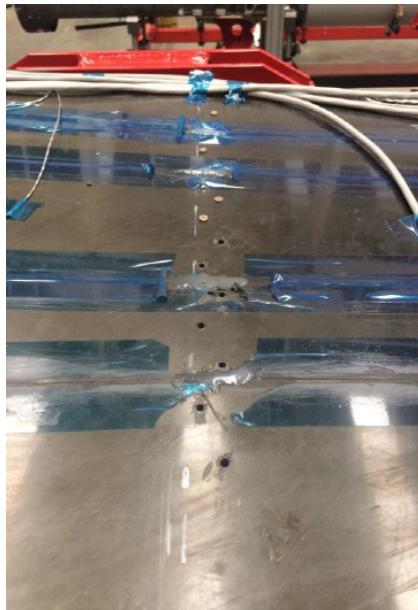


Figura 3.6: Realización del daño por extracción de remaches

Los ensayos sobre la estructura real se simularon también siguiendo la misma configuración en el FEM. Las condiciones de contorno a las que está sometido el fuselaje son las siguientes:

- Empotramiento en la sección de unión con el fuselaje anterior
- Carga vertical en la sección de unión con la cola

El ensayo comienza con la estructura descargada y se aplica la carga de forma escalonada. La carga va aumentando 400 N en 10 escalones hasta llegar a un máximo de 4 kN.

La configuración del banco de ensayos se puede ver en la Figura 3.7

### 3.1.3. Modelo de elementos finitos

Se va a hacer un breve resumen de las características del FEM utilizado. Este modelo consta de 62802 grados de libertad, el tamaño medio de sus elementos es de 15 mm, y está formado por:

- 26181 nodos
- 34080 elementos quad4 (Ratio máximo de 4.768)
- 25 elementos tria3 (Ratio máximo de 6.219)

Los elementos tria3 del modelo se encuentran principalmente en las cuatro cuadernas debido a que su geometría es más complicada de mallar que la del resto del fuselaje.

En la Figura 3.8 se pueden ver los distintos elementos del modelo y en cuales de ellos se introducen las condiciones de contorno.

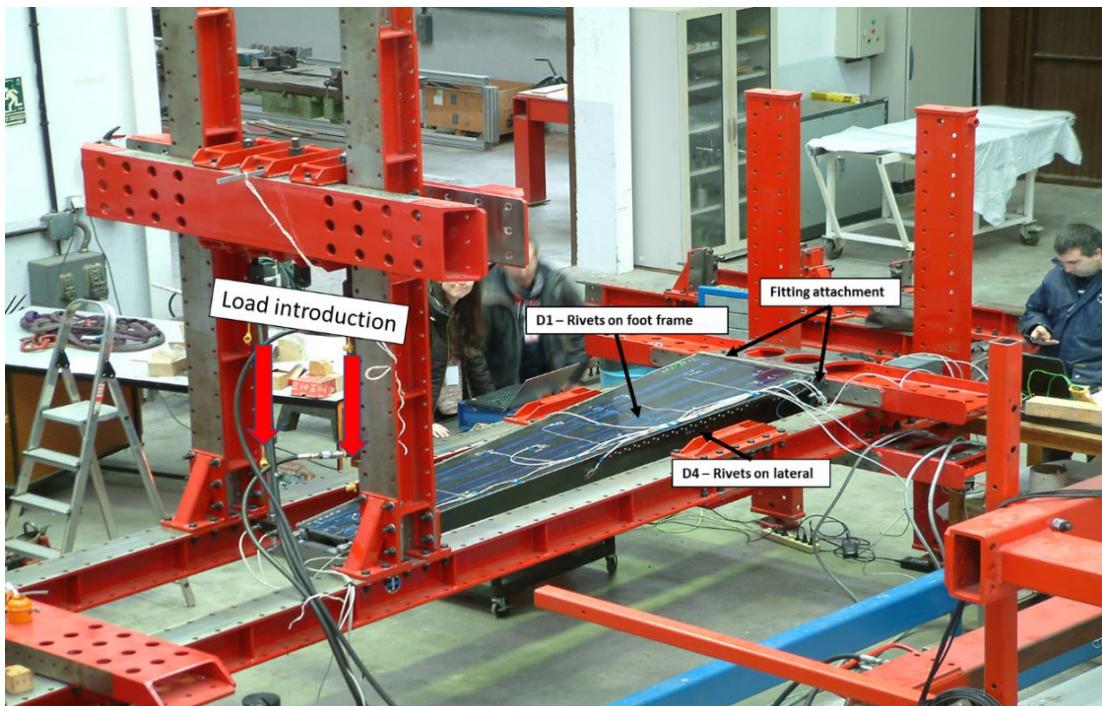


Figura 3.7: Vista general de la configuración de los ensayos

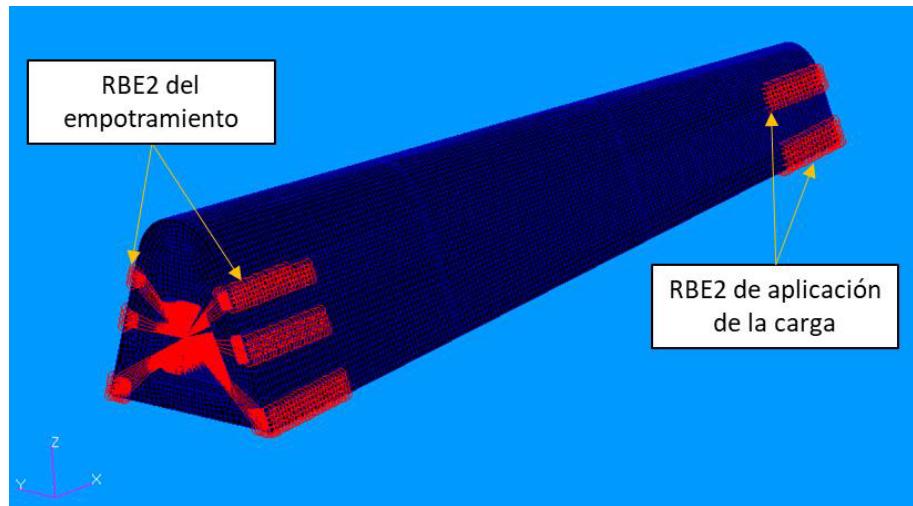


Figura 3.8: Modelo de elementos finitos del fuselaje posterior del UAV Milano

### 3.2. Pre-procesado de señales

Una vez que se han realizado los ensayos y simulaciones, es necesario pre-procesar los datos recogidos por las redes de sensores para ajustarlos en contenido y forma para que las redes neuronales puedan realizar una clasificación correcta.

Los datos que se obtienen de los ensayos reales se van a tratar de forma diferente a los obtenidos a través del FEM, por ello, este apartado está separado en dos grupos, *Ensayo real* y *FEM*.

### 3.2.1. Ensayo real

La red de sensores FBG registra las deformaciones de forma ininterrumpida durante el ensayo completo, dando así una curva de deformación frente al tiempo para cada uno de los sensores distribuidos en la estructura, como se puede ver a continuación en la Figura 3.9.

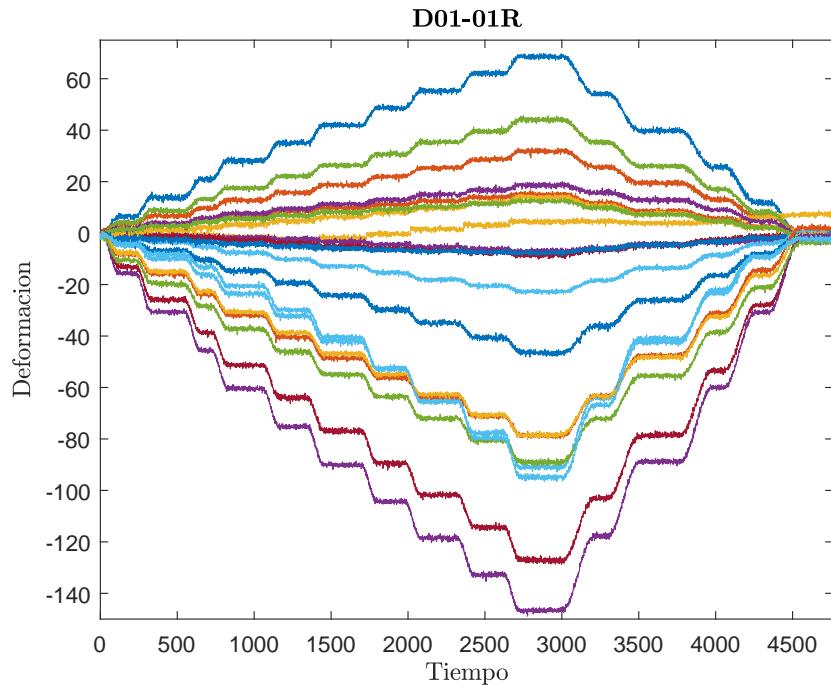


Figura 3.9: Medida de todos los sensores durante el ensayo del Daño 1 con un remache extraído

Cuando los actuadores llegan a un nivel de carga paran durante un tiempo antes de aumentar o disminuir la carga aplicada con el fin de estabilizar la carga. Por lo tanto, para separar las medidas en diferentes niveles de carga hay que detectar cuando el actuador comienza a moverse. La detección de el comienzo de aplicación de carga está representada en las Figuras 3.10 y 3.11.

Finalmente se eliminan las medidas obtenidas durante la transición entre escalones usando como referencia las líneas naranjas de la Figura 3.10 y se fija la carga mínima que se va a usar durante el estudio en un 30 % de la carga máxima.

Una vez que se han aislado los escalones se dibuja un histograma para ver la variabilidad de los datos obtenidos. Como se ve en la Figura 3.24, los valores de las medidas se asemejan a una distribución normal con mida -127.75 deformaciones. Esto quiere decir que el ruido introducido por los sensores FBG hace que cada medida que se realiza en un tiempo diferente pueda ser considerada como un ensayo independiente. De un único ciclo de carga se obtienen una gran cantidad de datos para entrenar a la red.

Tras pre-procesar las señales se puede visualizar el campo de deformaciones medido por los sensores FBG para cada estado de daño y el estado sin daño o de referencia en la Figura 3.13. Al estar cada daño localizado en una zona diferente de la estructura, los cambios en el campo de deformaciones son locales y en la Figura 3.14 se aprecia que sensores están

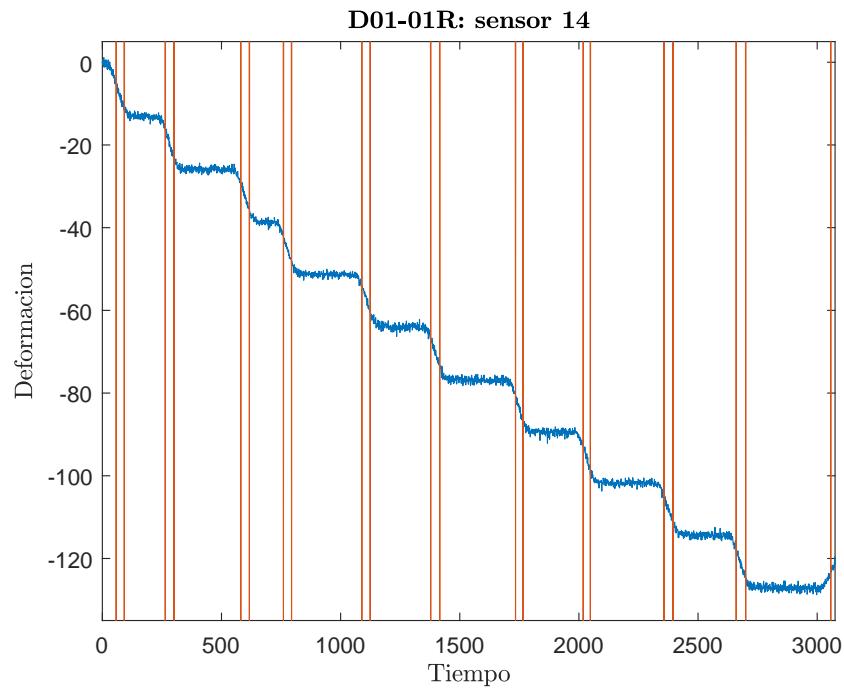


Figura 3.10: Detección de los escalones producidos por las carga vertical

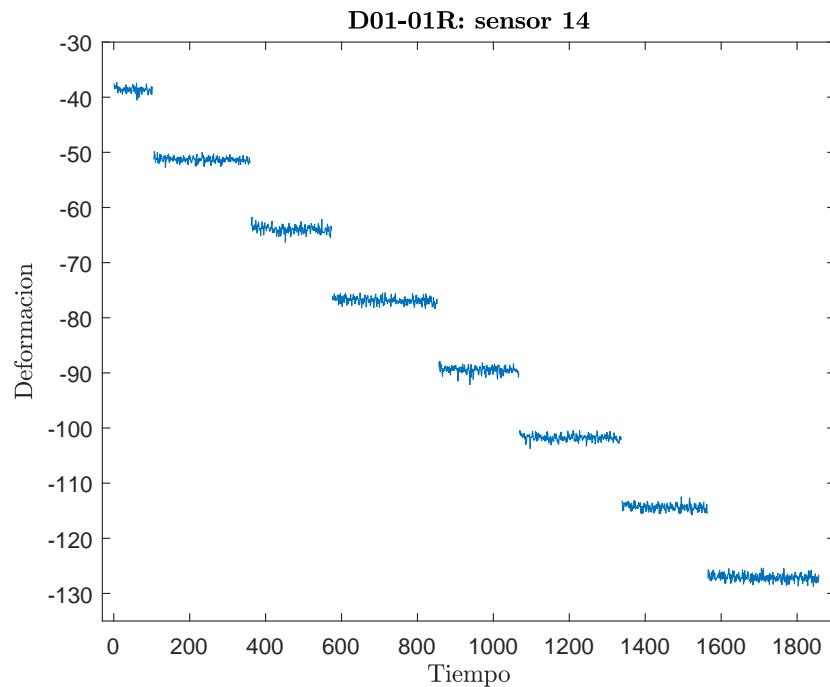


Figura 3.11: Medidas de los sensores FBG en el ensayo preprocesadas

más cerca de la zona dañada ya que sufren una mayor variación respecto a las medidas obtenidas de la estructura sin daño.

Por otro lado, los sensores OBR realizan una captura instantánea del campo de defor-

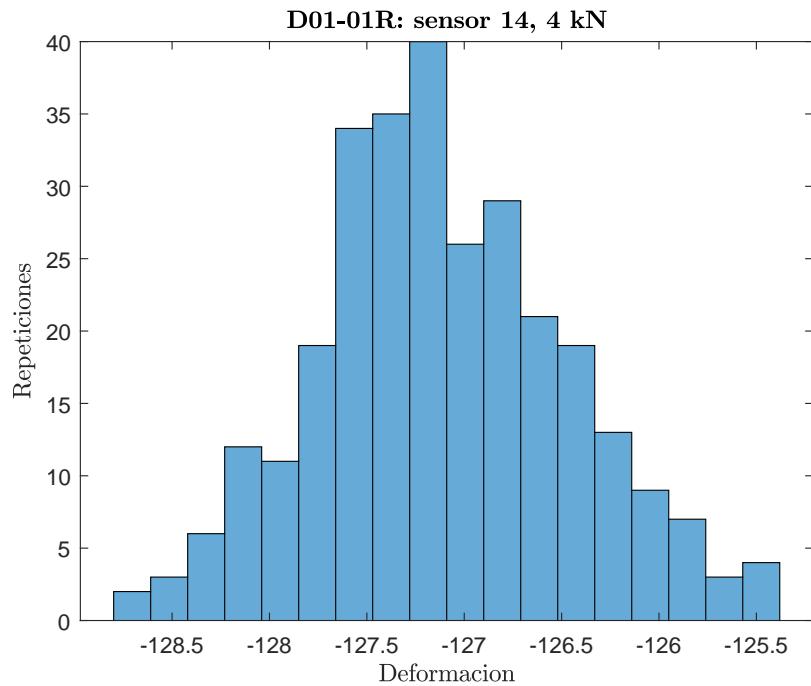


Figura 3.12: Histograma del escalón 4 kN, 100 % de carga

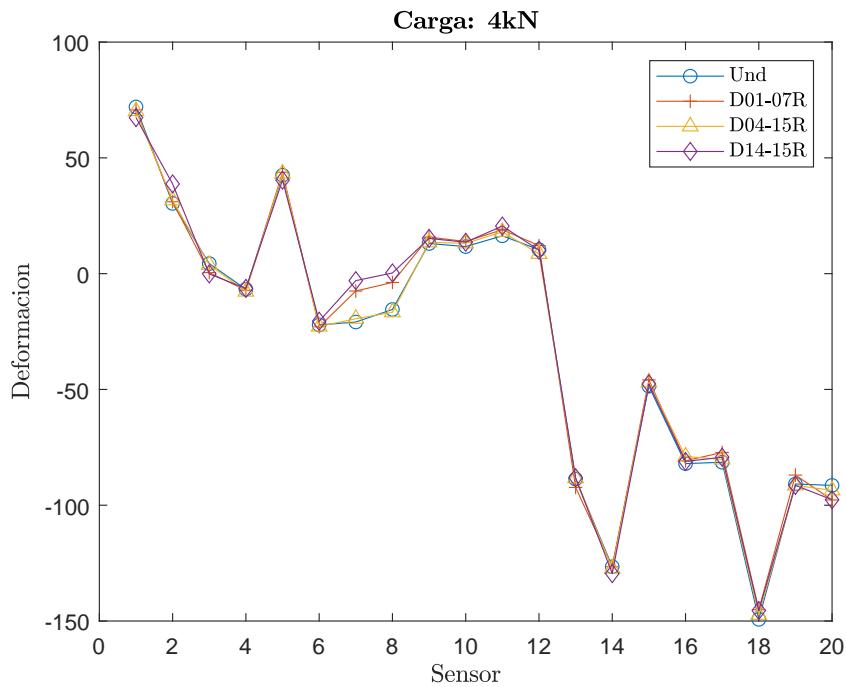


Figura 3.13: Campo de deformaciones medido por los sensores FBG para todos los daños

maciones. Esto quiere decir que se tiene una única muestra para cada tipo de daño y nivel de carga (ejemplo en la Figura 3.15), lo cual es insuficiente para conseguir unos buenos resultados con la red.

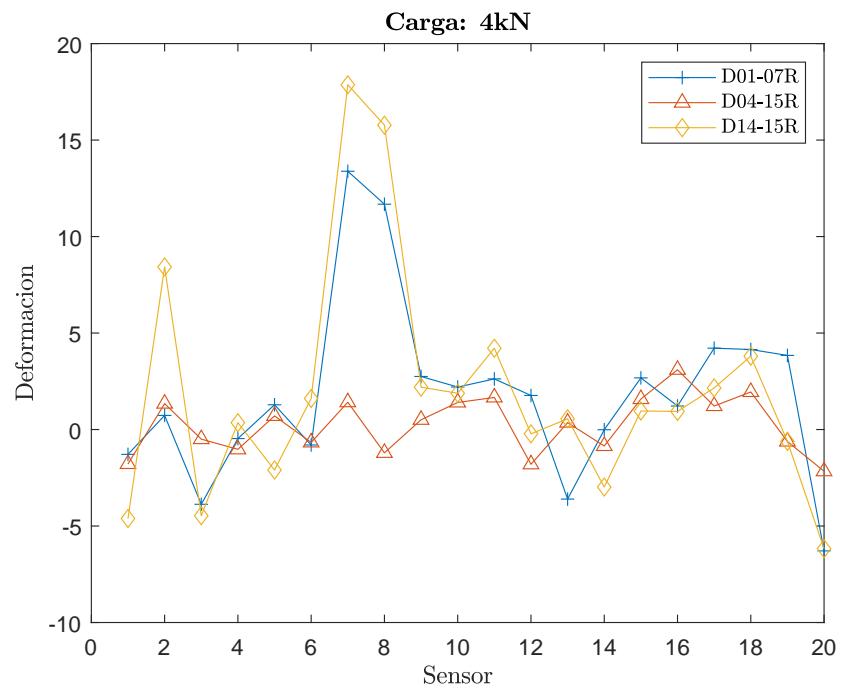


Figura 3.14: Diferencia de los campos de deformaciones referenciado con el estado sin daño (FBG)

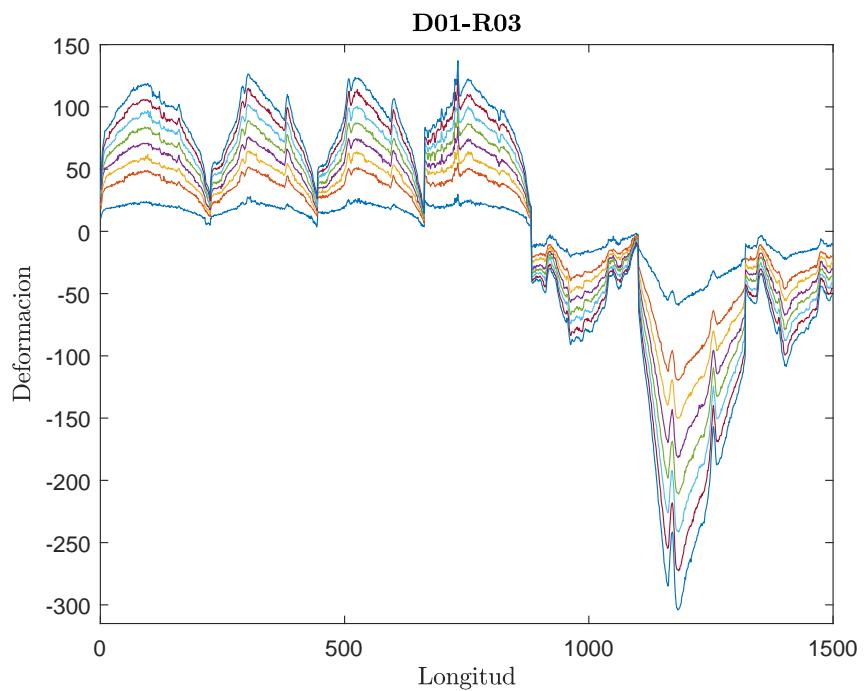


Figura 3.15: Campo de deformaciones provocado por el daño 1 con 3 remaches extraídos para todos los casos de carga

Siguiendo el mismo razonamiento usado con los sensores FBGs, si se añade una distribución de ruido gausiano a un valor del campo de deformaciones, se pueden generar ensayos independientes con los que entrenar a la red. Sin embargo, el campo de deformaciones de la Figura 3.15 ya contiene el ruido asociado a la medida.

Para conseguir un campo de deformaciones que usar como referencia se va a utilizar el filtro *Savitzky-Golay*. Este filtro digital es comúnmente usado para suavizar series de datos equiespaciados [Savitzky-Golay].

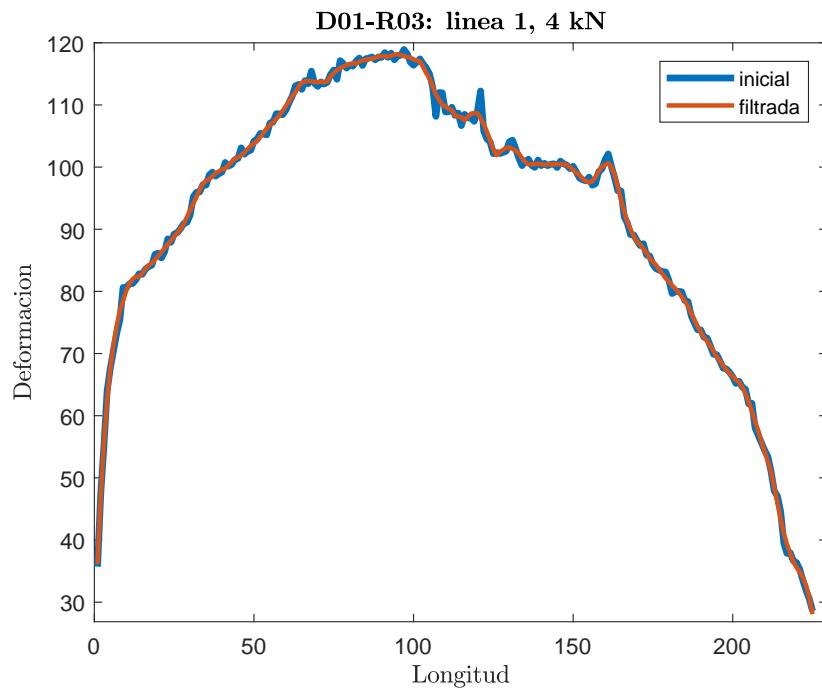


Figura 3.16: Aplicación del filtro Savitzky-Golay a una linea de sensor OBR

La diferencia entre la medida sin filtrar y filtrada se aprecia claramente en la Figura 3.16.

Ahora que se tiene una medida de referencia, ya se puede añadir la distribución de ruido de amplitud 1.5 deformaciones (Figura 3.17) y generar más muestras, Figura 3.18.

En la Figura ?? se puede ver un detalle del caso D01-R03 en el que se ha ampliado una línea de sensor. Aquí se aprecia más claramente que cada una de estas muestras se pueda considerar como un ensayo independiente.

Una vez se han procesado los datos proporcionados por las redes de sensores, se puede representar el campo de deformaciones que produce cada daño. Para tener una referencia visual clara, a continuación se van a comparar estos daños entre sí, cogiendo como representación de cada uno el número medio de remaches extraídos y la carga máxima.

Lo mismo se aprecia en las Figuras 3.19 y 3.20 para el sensor OBR. Se puede extraer la misma conclusión que con los sensores FBG pero al tener muchas más medidas, las diferencias entre daños no se ven tan claras.

### 3.2.2. FEM

<https://towardsdatascience.com/hands-on-graph-neural-networks-with-pytorch-pytorch->

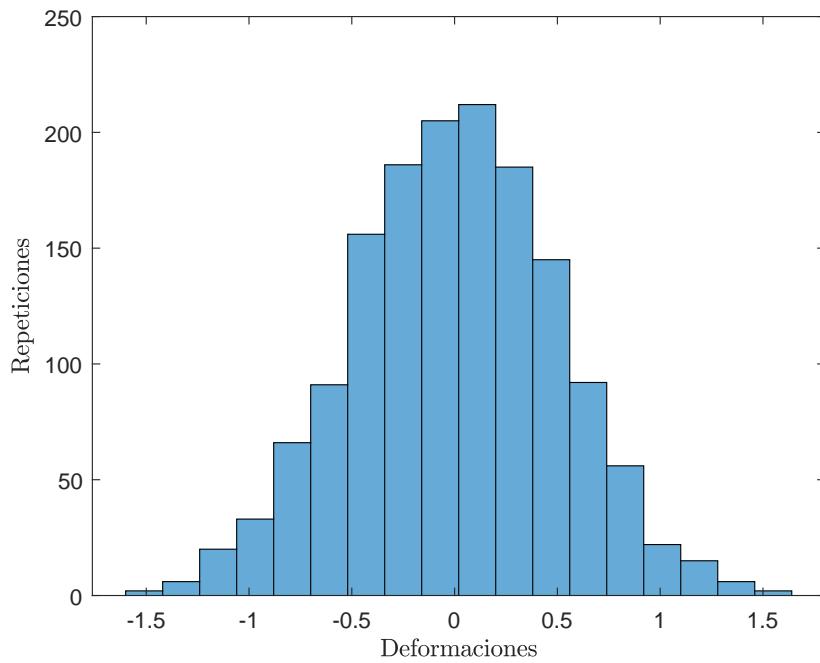


Figura 3.17: Distribución de ruido gausiano con amplitud 1.5 deformaciones

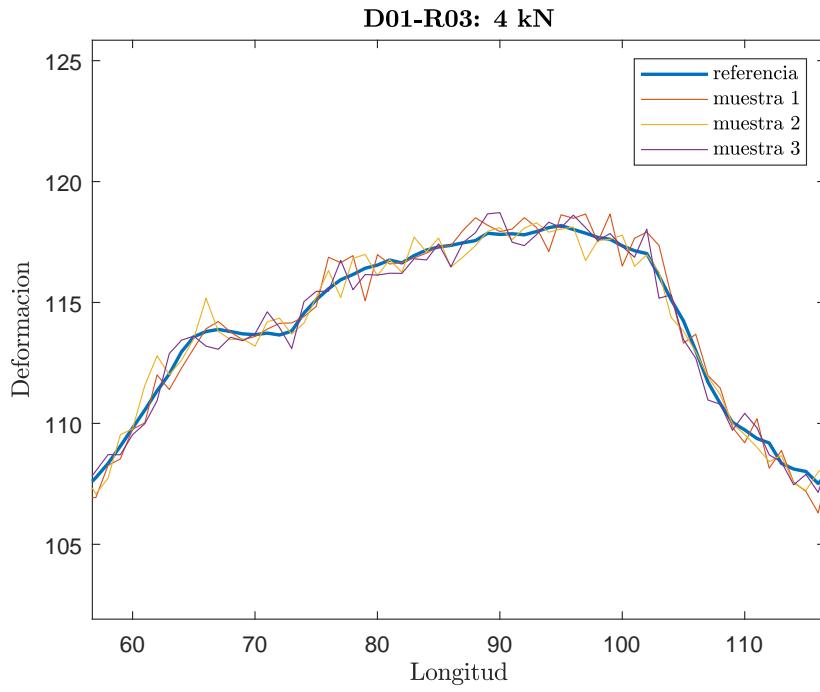


Figura 3.18: Detalle de las variaciones provocadas por el ruido gausiano

### 3.3. Construcción del modelo de Deep Learning

Explicación del funcionamiento de una red neuronal

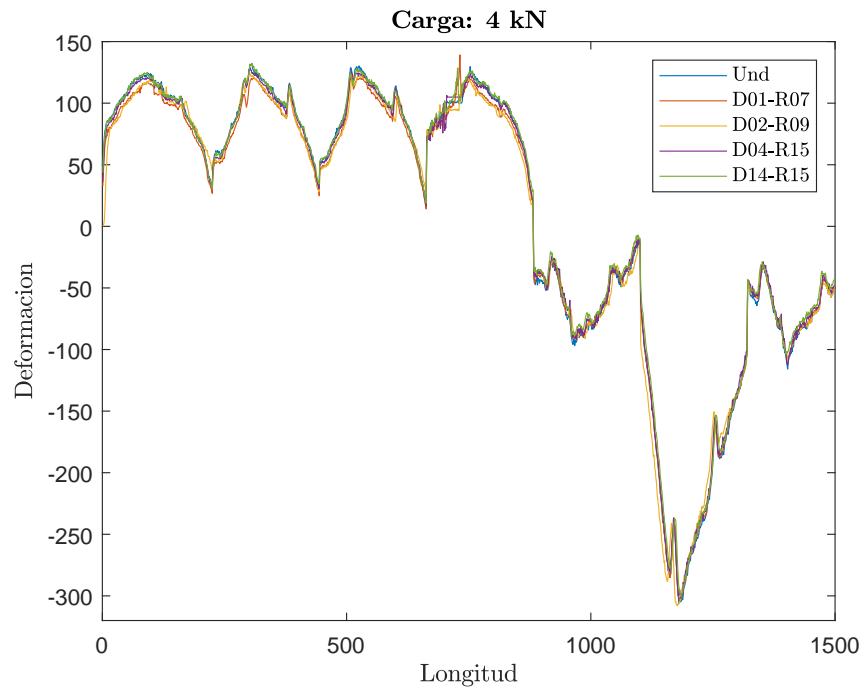


Figura 3.19: Campo de deformaciones medido por el sensor OBR para todos los daños

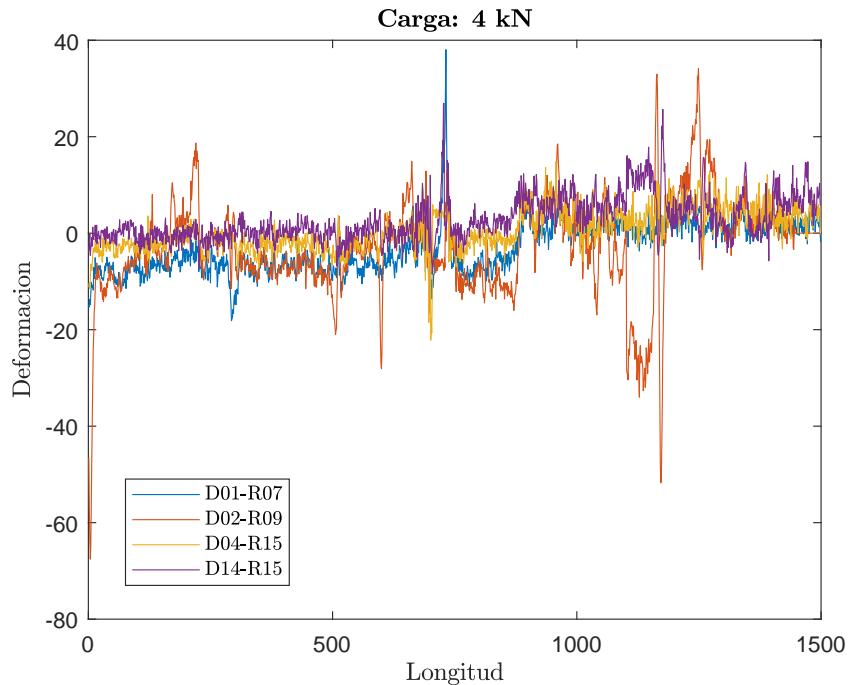


Figura 3.20: Diferencia de los campos de deformaciones referenciado con el estado sin daño (OBR)

Una vez que se han preparado los datos, se puede pasar a construir el modelo de DL que se va a utilizar para clasificar los daños descritos.

## 3.4. Resultados

### 3.4.1. INESASSE

- CLASIFICACIÓN DE CARGA

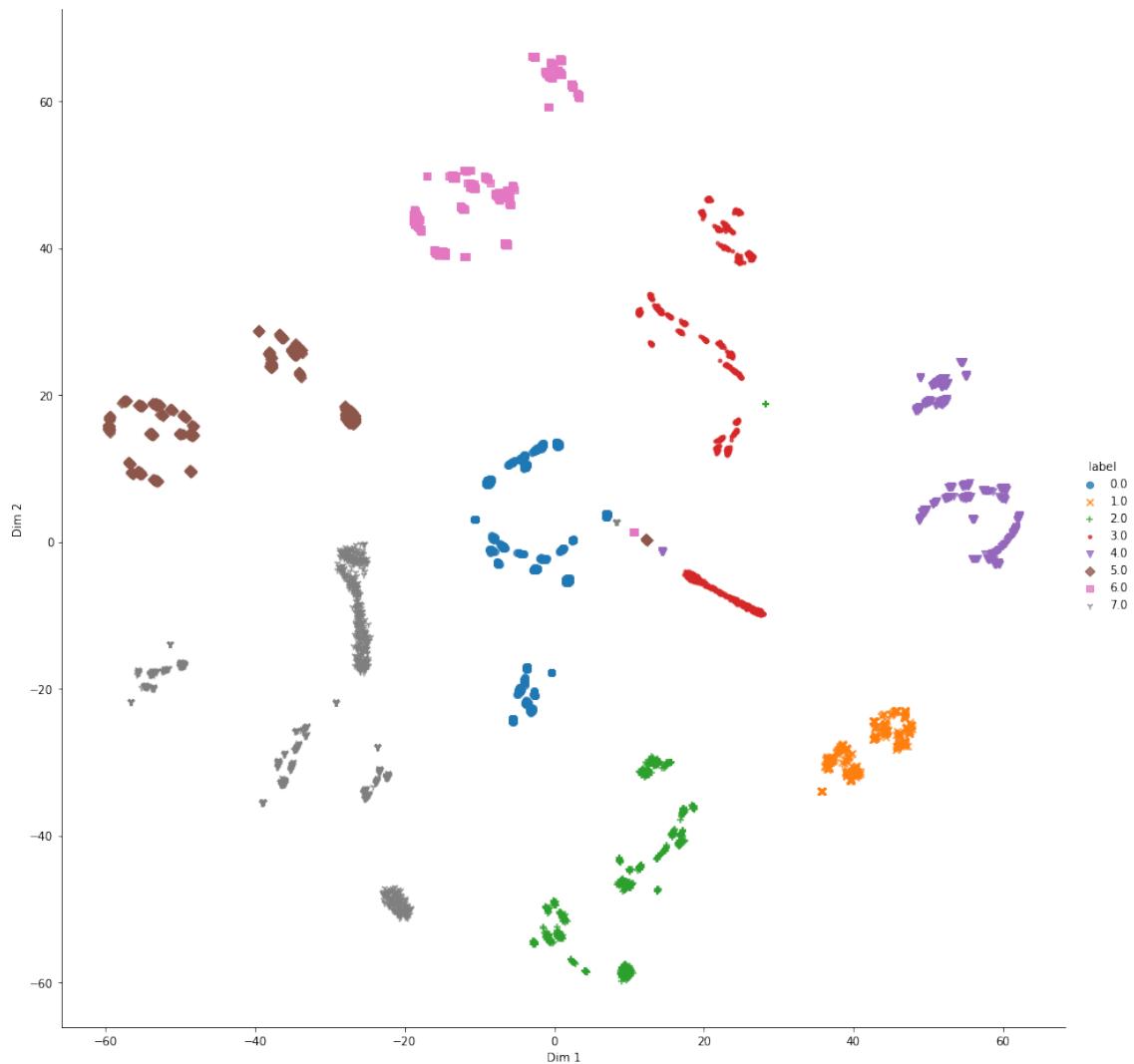


Figura 3.21: Representación de los vectores de deformaciones tras reducir la dimensión de 20 a 2

- CLASIFICACIÓN DE CASO DE DAÑO



Figura 3.22: Matriz de confusión

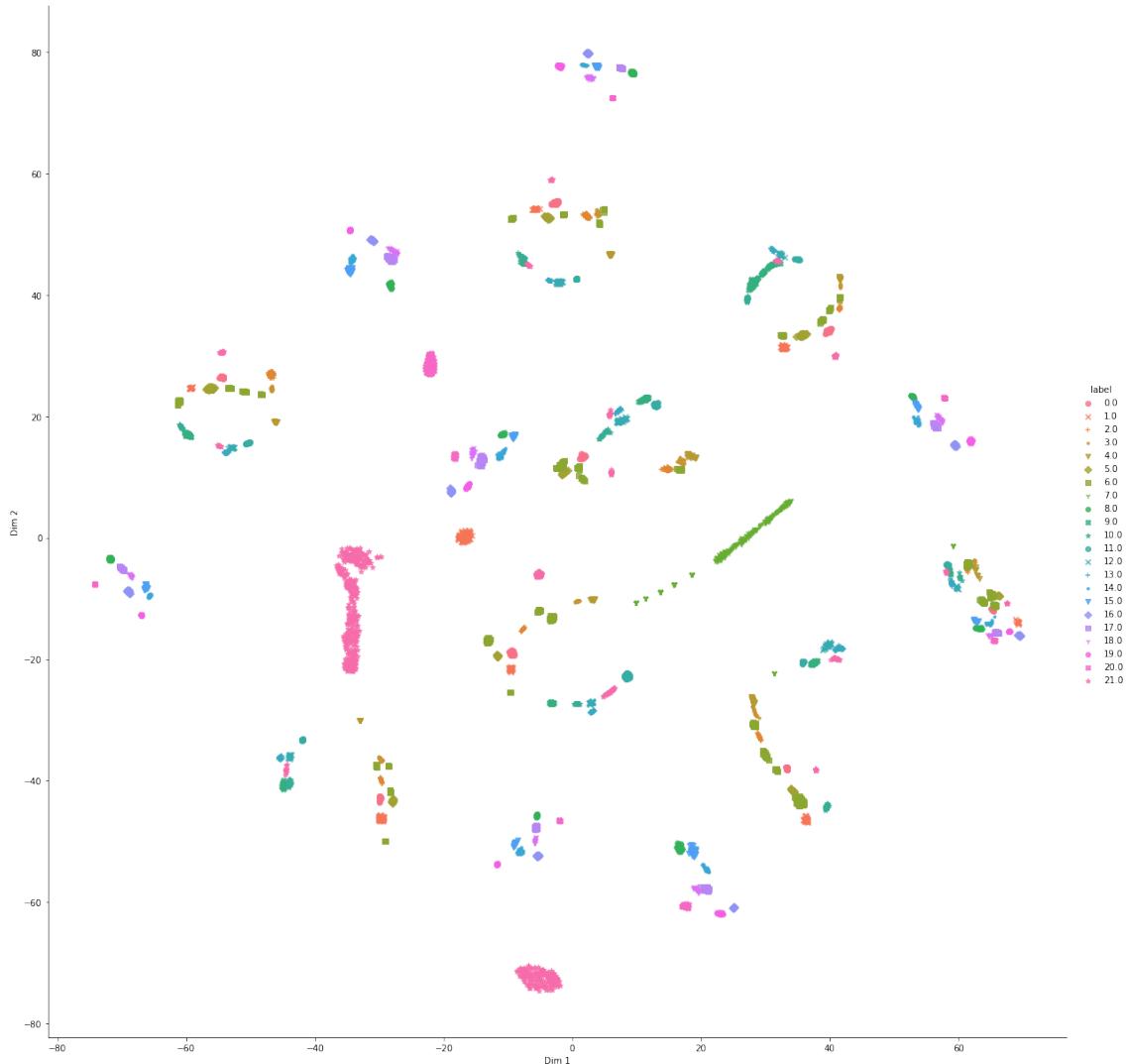


Figura 3.23: Representación de los vectores de deformaciones tras reducir la dimensión de 20 a 2

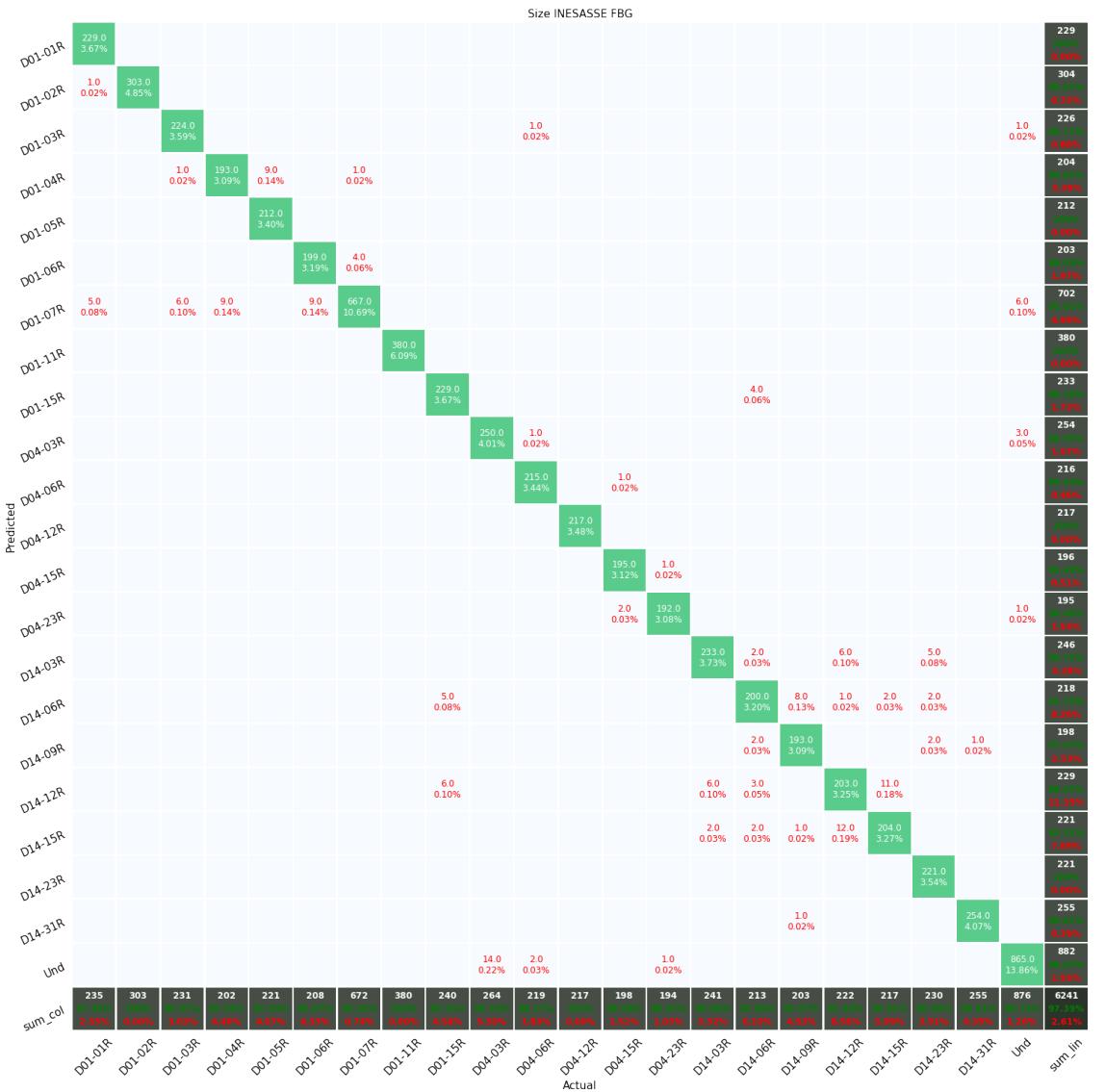


Figura 3.24: Matriz de confusión

## **Capítulo 4**

# **Caracterización de impactos para SHM**

### **4.1. Propuesta de ensayo experimental**

#### **4.1.1. Impactador estructura**

#### **4.1.2. Electrónica - Software y hardware**

#### **4.1.3. Adquisición y procesado de datos**

## **4.2. Caracterización de la posición**

CON DATOS DE CHRISTIAN Y AIRBUS AL MENOS  
Red que clasifica la posición de un impacto

### **4.3. Caracterización de la energía**

CON DATOS DE CHRISTIAN Y AIRBUS AL MENOS  
Red que clasifica la energía de impacto.

#### **4.4. Caracterización de la velocidad**

Si se termina por realizar el impactador, también se buscará una clasificación de velocidad da impacto para diferentes masas y misma altura de suelta.

#### **4.5. Caracterización completa de un impacto**

Combinación de los anteriores clasificadores para comprobar la precisión completa de un impacto.

## **Capítulo 5**

# **Conclusiones y trabajos futuros**

### **5.1. Conclusiones**

al menos 3-4 págs

## **5.2. Trabajos futuros**

1 pag qewrg weqwewe [**einstein**]