



UNIVERSIDAD  
**NACIONAL**  
DE COLOMBIA

# **Estudio comparativo de algoritmos de clasificación de imágenes basados en análisis de datos funcionales. Caso de estudio en superficies de fractura de elementos mecánicos**

**Edgar Espejo Mora**

Universidad Nacional de Colombia  
Facultad de Ciencias, Departamento de Estadística  
Bogotá D.C., Colombia

2024



# **Estudio comparativo de algoritmos de clasificación de imágenes basados en análisis de datos funcionales. Caso de estudio en superficies de fractura de elementos mecánicos**

**Edgar Espejo Mora**

Trabajo final presentado como requisito parcial para optar al título de:  
**Magister en Ciencias - Estadística**

Directora:

Ph.D. Martha Patricia Bohórquez Castañeda

Línea de Investigación:

Datos funcionales espaciales

Grupo de Investigación:

Estadística Espacial

Universidad Nacional de Colombia  
Facultad de Ciencias, Departamento de Estadística  
Bogotá D.C., Colombia

2024



## *Dedicatoria*

*A mi hija Eilin, a mi señora María del Pilar, a mis padres Nubia y Alfonso, y a mi hermana Aide por su acompañamiento constante durante este proceso de formación que culmina.*

## **Agradecimientos**

A la profesora Martha Patricia Bohórquez Castañeda del departamento de estadística, por su guía y aportes durante la realización del presente trabajo, a los profesores Álvaro Castro Primiciero, Héctor Hernández Albañil y Héctor Mosquera Mosquera (QEPD) del departamento de ingeniería mecánica y Mecatrónica, quienes me encaminaron durante mi proceso formativo profesional hacia la ciencia de los materiales.

## Resumen

### **Estudio comparativo de algoritmos de clasificación de imágenes basados en análisis de datos funcionales. Caso de estudio en superficies de fractura de elementos mecánicos**

Para este trabajo se usaron fotografías tomadas a superficies de fractura de elementos mecánicos, que fallaron mediante fractura dúctil, fractura frágil y fractura por fatiga. Cada uno de estos tipos de fractura deja una textura característica, a partir de la cual un experto en análisis de fallas puede usarla para clasificarlas. De las imágenes se extrajeron datos funcionales y con ellos se evaluó la exactitud de varios modelos de clasificación. De cada imagen de 200 x 200 píxeles, se extrajeron 400 datos funcionales correspondientes a cada línea de píxeles en X e Y. Como modelos se usaron métodos estadísticos basados en distancias, modelo lineal generalizado (MLG), modelo aditivo generalizado (MAG) y modelos basados en medidas de profundidad. También se usaron modelos de aprendizaje de máquina como K-vecinos más cercanos, máquina de soporte vectorial (MSV), regresión logística, árbol de decisión, bosque aleatorio y red neuronal tipo perceptrón. En los modelos estadísticos se evaluó también el efecto sobre la exactitud de clasificación, de incluir información de autocorrelación espacial intra dato funcional o del operador de covarianza. Como conclusiones relevantes se obtuvo que la inclusión de la información de autocorrelación espacial a los clasificadores basados en métodos estadísticos, mejora la exactitud de los mismos y que los datos funcionales extraídos de las imágenes tienen la información suficiente para entrenar modelos de clasificación.

**Palabras clave:** Aprendizaje Supervisado, Fractografía, Análisis de Datos Funcionales, Autocorrelación Espacial.

## Abstract

### **Comparative study of image classification algorithms based on functional data analysis. Case study on fracture surfaces of mechanical elements**

For this work, photographs taken of fracture surfaces of mechanical elements were used, which failed through ductile fracture, brittle fracture and fatigue fracture. Each of these fracture types leaves a characteristic texture, from which a failure analysis expert can use to classify them. Functional data was extracted from the images and the accuracy of various classification models was evaluated. From each image of 200 x 200, 400 functional data were extracted corresponding to each line of pixels X and Y. Statistical methods based on distances, generalized linear model (GLM), generalized additive model (GAM) and models based on depth measurements were used. Machine learning models such as K-nearest neighbors, support vector machine (SVM), logistic regression, decision tree, random forest and perceptron-type neural network were also used. In the statistical models, the effect on classification accuracy of including spatial autocorrelation information or the covariance operator was also evaluated. As relevant conclusions, it was obtained that the inclusion of spatial autocorrelation information to classifiers based on statistical methods, improves their accuracy and that the functional data extracted from the images have sufficient information to train classification models.

**Keywords: Supervised Learning, Fractography, Functional Data Analysis, Spatial Autocorrelation.**

# Contenido

	Pág.
<b>Resumen</b> .....	<b>V</b>
<b>Lista de figuras</b> .....	<b>IX</b>
<b>Lista de tablas</b> .....	<b>X</b>
<b>Lista de Símbolos y abreviaturas</b> .....	<b>XII</b>
<b>Introducción</b> .....	<b>1</b>
<b>1. Datos</b> .....	<b>3</b>
1.1 Imágenes de partida.....	3
1.2 Datos funcionales.....	8
1.3 Estadística descriptiva.....	8
1.4 Pruebas de hipótesis sobre medias y operadores de covarianza .....	10
1.5 Funciones de autocorrelación espacial intra datos funcionales .....	14
1.6 Análisis de componentes principales.....	15
<b>2. Revisión bibliográfica</b> .....	<b>21</b>
2.1 Análisis y clasificación de datos funcionales.....	21
2.2 Clasificación de imágenes.....	24
<b>3. Objetivos</b> .....	<b>27</b>
3.1 Objetivo general .....	27
3.2 Objetivos específicos .....	27
<b>4. Marco teórico</b> .....	<b>29</b>
4.1 Datos funcionales.....	29
4.2 Clasificadores basados en distancias.....	31
4.3 Clasificadores basados en MLG y MAG .....	32
4.4 Clasificadores basados en medidas de profundidad.....	33
4.5 Clasificadores basados en aprendizaje de máquina.....	35
<b>5. Metodología</b> .....	<b>41</b>
<b>6. Resultados</b> .....	<b>53</b>
6.1 Análisis.....	58
<b>7. Conclusiones y recomendaciones</b> .....	<b>63</b>
7.1 Conclusiones.....	63
7.2 Recomendaciones.....	64

<b>Anexo A: Matrices de confusión.....</b>	<b>65</b>
<b>Anexo B: Medidas de Exactitud.....</b>	<b>77</b>
<b>Bibliografía.....</b>	<b>83</b>

## Lista de figuras

	Pág.
<b>Figura 1-1:</b> Ejemplo de datos funcionales extraídos de las imágenes. ....	8
<b>Figura 1-2:</b> Ejemplo de datos funcionales de cada categoría (eje horizontal en píxeles y vertical en intensidad de luz normalizada). Base de 150 B-Splines de orden cúbico, $\lambda$ obtenida por VCG. ....	9
<b>Figura 1-3:</b> Primeras funciones propias. Primera función en negro, en rojo la segunda y en verde la tercera. ....	17
<b>Figura 1-4:</b> "Puntajes" sobre las funciones propias 1 y 2. Dúctiles en negro, frágiles en rojo y fatiga en verde. ....	17
<b>Figura 4-1:</b> Ejemplo de un árbol de decisión. ....	36
<b>Figura 4-2:</b> Ejemplo esquemático de un bosque aleatorio. ....	36
<b>Figura 4-3:</b> Ejemplo de clasificación usando K-vecinos más cercanos. ....	37
<b>Figura 4-4:</b> Ejemplo de hiperplano óptimo de separación en una MSV (esquema derecho). ....	38
<b>Figura 4-5:</b> Ejemplo del efecto de la escogencia del núcleo en una MSV (izquierda núcleo lineal, derecha núcleo no lineal). ....	38
<b>Figura 4-6:</b> Ilustración de una red neuronal tipo perceptrón. ....	40
<b>Figura 4-7:</b> Ilustración de una red neuronal de 3 capas (entrada, capa oculta y salida), donde se esquematiza la relación entre sus variables. ....	40
<b>Figura 5-1:</b> Secciones metalográficas que intersectan superficies de fractura. Izquierda dúctil, central frágil y derecha fatiga. 200X. ....	42
<b>Figura 5-2:</b> Gráfica de las 150 funciones cúbicas de la base B-Spline usadas para representar los datos funcionales y las funciones de autocorrelación. Imagen superior con detalle hasta el píxel 50 e imagen inferior hasta el píxel 200. ....	43
<b>Figura 5-3:</b> Ejemplo de los puntos originales de un dato funcional (puntos negros) y dato funcional representado con la base B-Spline (curva roja). ....	44
<b>Figura 5-4:</b> Ejemplo de gráfica de los puntos originales de un dato funcional de una función de autocorrelación (puntos negros) y dato funcional representado con la base B-Spline (curva roja). ....	50
<b>Figura 5-5:</b> Diagrama de flujo de la metodología usada. ....	51
<b>Figura 5-6:</b> Resumen de los modelos de clasificación usados. En amarillo se resaltan aquellos que usaron la función de autocorrelación intra dato funcional o las matrices de covarianza. ....	52

## Lista de tablas

Pág.

<b>Tabla 1.1:</b>	Imágenes de las superficies de fractura dúctil (textura fibrosa). Tomadas por el autor.	5
<b>Tabla 1.2:</b>	Imágenes de las superficies de fractura frágil (textura granular). Tomadas por el autor.	6
<b>Tabla 1.3:</b>	Imágenes de las superficies de fractura por fatiga (textura tersa). Tomadas por el autor.	7
<b>Tabla 1.4:</b>	Funciones media (izquierda) y mediana (derecha) de los datos funcionales usados. Base de 150 B-Splines de orden cúbico, lambda obtenida por VCG. ....	10
<b>Tabla 1.5:</b>	Comparación entre funciones y medianas por parejas de categorías. ....	11
<b>Tabla 1.6:</b>	Funciones de covarianza por categorías.....	12
<b>Tabla 1.7:</b>	Resultados de las pruebas de igualdad de medias e igualdad de operadores de covarianza entre las categorías.....	13
<b>Tabla 1.8:</b>	Ejemplo de comparación por categorías, entre funciones de autocorrelación intra datos funcionales. ....	16
<b>Tabla 1.9:</b>	Gráficas de "Puntajes" entre la primera función propia (eje horizontal) y las demás hasta la décima. En negro dúctiles y en rojo frágiles. ....	18
<b>Tabla 1.10:</b>	Gráficas de "Puntajes" entre la primera función propia (eje horizontal) y las demás hasta la décima. En negro dúctiles y en verde fatiga. ....	19
<b>Tabla 1.11:</b>	Gráficas de "Puntajes" entre la primera función propia (eje horizontal) y las demás hasta la décima. En verde fatiga y en rojo frágiles. ....	20
<b>Tabla 4.1:</b>	Modelos de error y funciones de enlace usados en MLG y MAG. ....	33
<b>Tabla 4.2:</b>	Núcleos comunes en las MSV. ....	38
<b>Tabla 5.1:</b>	Comportamiento del error cuadrático medio vs el número de funciones de la base B-Spline.....	45
<b>Tabla 6.1:</b>	Ejemplo de matrices de confusión obtenidas para el clasificador por regresión logística. ....	53
<b>Tabla 6.2:</b>	Ejemplo de medidas de exactitud obtenidas del modelo de clasificación, basado en regresión logística de los datos funcionales no centrados. ....	54
<b>Tabla 6.3:</b>	Comparación de las medidas de exactitud obtenidas en los clasificadores por fotos. En amarillo se resaltan clasificadores que usaron información de la función de covarianza o de la función de autocorrelación intra dato funcional. En negro exactitud global, en azul para dúctil, en rojo para frágil y en verde para fatiga. ....	55



# Lista de Símbolos y abreviaturas

## Abreviaturas

<b>Abreviatura</b>	<b>Término</b>
<i>ACF</i>	Función de autocorrelación
<i>ACPF</i>	Análisis de componentes principales funcionales
<i>ADF</i>	Análisis de datos funcionales
<i>FM</i>	Profundidad de Fraiman and Muniz
<i>JPG</i>	Archivo de imagen comprimido
<i>MAG</i>	Modelo aditivo generalizado
<i>h-Modal</i>	Profundidad h-Modal
<i>MLG</i>	Modelo lineal generalizado
<i>MSV</i>	Máquina de soporte vectorial
<i>RBF</i>	Núcleo de funciones radiales base
<i>RGB</i>	Canales de color rojo, verde y azul
<i>RP</i>	Profundidad proyectada aleatoria
<i>RPD</i>	Profundidad proyectada doblemente aleatoria
<i>RT</i>	Profundidad de Tukey aleatoria
<i>VCG</i>	Validación cruzada generalizada
<i>“por curvas”</i>	Análisis por datos funcionales o curvas
<i>“por fotos”</i>	Análisis por fotografías
$\sigma_U$	Resistencia última de un material

# Introducción

La Fractografía es una disciplina de la ciencia de los materiales, que busca inferir el tipo de fractura presente en un elemento mecánico, a partir de características como la textura de la superficie de fractura, grado de deformación plástica de las piezas y microestructura del material presente en la zona de fractura.

Generalmente la identificación del tipo de fractura presente en un elemento mecánico, la realiza un experto en análisis de fallas entrenado para ello, que se apoya en inspección de la superficie de fractura a ojo desnudo, en estereoscopio, en microscopio óptico y en microscopio electrónico. El desarrollo de sistemas expertos basados en análisis de imágenes de superficies de fractura, que puedan emitir concepto sobre el tipo de fractura presente, es de utilidad tanto para apoyo de analistas de fallas, como para personal de mantenimiento mecánico general. Estos sistemas permiten tener una segunda opinión a la hora de diagnosticar fallas y a partir de ello tomar acciones correctivas para minimizar su recurrencia.

Este trabajo se propuso precisamente para determinar qué modelos estadísticos o de aprendizaje de máquina, son candidatos para desarrollar algoritmos de clasificación de superficies de fractura. Para ello se escogió por parte del autor un conjunto de imágenes de entrenamiento y de prueba de modelos, que previamente había tomado y clasificado por tres categorías de fractura. De las imágenes se extrajeron datos funcionales de intensidad de luz vs píxeles, a lo largo de los ejes X e Y. El 70% de los datos funcionales se usaron para entrenamiento de modelos de clasificación y el 30% restante para probarlos.

Se evaluaron modelos de clasificación supervisada tanto estadísticos, como de aquellos comúnmente denominados de aprendizaje de máquina. Entre los modelos estadísticos están algunos basados en distancias, en MLG (Modelos Lineales Generalizados), en MAG

(Modelos Aditivos Generalizados) y en medidas de profundidad de datos funcionales. En algunos de estos modelos se evaluó el efecto sobre la exactitud de clasificación, de incluir o no información de autocorrelación espacial (ACF), a través de datos funcionales correspondientes a las funciones de autocorrelación intra dato funcional original y de las matrices u operadores de covarianza entre datos funcionales. Los modelos de aprendizaje de máquina evaluados fueron K-vecinos más cercanos, máquina de soporte vectorial (MSV), regresión logística, árbol de decisión, bosque aleatorio y red neuronal tipo perceptrón.

Con los resultados de las clasificaciones hechas sobre los datos funcionales de prueba, se calcularon matrices de confusión y medidas de exactitud para cada clasificador. Esto se hizo tanto con los resultados de clasificación por datos funcionales, que en el cuerpo del texto se citan como “por curvas”, como también por los resultados de clasificación por imágenes o citados como “por fotos”. Con lo anterior se procedió a analizar los resultados y generar las conclusiones del trabajo.

El presente documento se organiza de la siguiente manera: en el capítulo 1 se presenta una descripción de los datos (imágenes y datos funcionales); en el capítulo 2 se resumen algunos aspectos relevantes encontrados en literatura especializada sobre clasificación de datos funcionales y de imágenes; en el capítulo 3 se exponen el objetivo general y los objetivos específicos que se persiguieron en este trabajo; en el capítulo 4 se encuentra una breve revisión teórica, sobre aspectos relevantes de los modelos de clasificación usados; en el capítulo 5 se detalla la metodología empleada en el desarrollo del trabajo; en el capítulo 6 se muestran los resultados y su análisis; y en el capítulo 7 se encuentran las conclusiones y recomendaciones que salieron del ejercicio. Al final de documento se encuentran los anexos.

# 1. Datos

En este capítulo se describirán las imágenes a partir de las cuales se extrajeron los datos funcionales para este trabajo, algunos resultados y análisis de la estadística descriptiva de los mismos, resultados de pruebas de hipótesis entre las medias y operadores de covarianza de las categorías de los datos funcionales, las funciones de autocorrelación espacial intra datos funcionales por categoría (ACF) y finalmente un análisis de componentes principales. Todo esto para verificar que los datos funcionales muestran diferencias significativas entre las diferentes categorías.

## 1.1 Imágenes de partida

La fractografía es una disciplina de la ciencia de los materiales, que se encarga de deducir a partir de inspección visual a ojo desnudo (1X), inspección visual con estereoscopio (7X a 90X), inspección con microscopía electrónica de barrido (25X a 50000X) e inspección metalográfica (50X a 2000X), el tipo de fractura que experimentó un elemento mecánico en servicio. Todas las inspecciones anteriores se hacen en la superficie de fractura o en el material adyacente a ella.

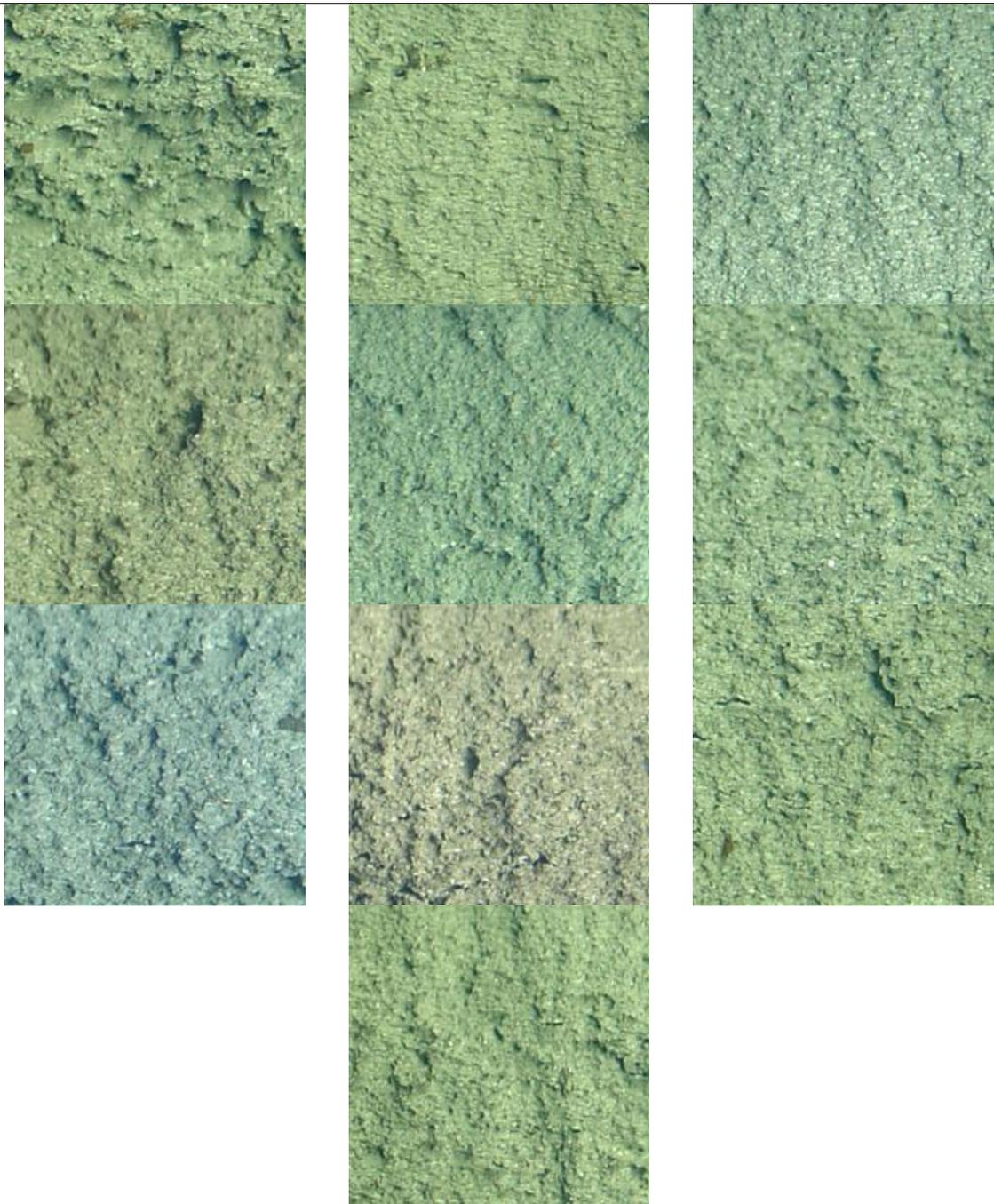
Las fracturas en los elementos mecánicos se pueden dividir en dos grandes familias: fracturas súbitas y fracturas progresivas. Las primeras se forman en fracciones de segundo, una vez la pieza mecánica en servicio, experimenta un esfuerzo que sobrepasa su resistencia última  $\sigma_U$ . En cambio, las fracturas progresivas se forman típicamente en rangos de tiempo de varios segundos hasta años (agrietamiento progresivo en el tiempo). Dentro de las fracturas súbitas se tienen dos categorías principales: fracturas dúctiles y fracturas frágiles. Las fracturas dúctiles se forman en materiales que se caracterizan por tener, bajos valores de resistencia última  $\sigma_U$  y altos de ductilidad (capacidad de deformarse plásticamente antes de fractura); mientras que las fracturas frágiles se forman en materiales con altos valores de resistencia última y bajos de ductilidad. Al observarse a ojo

desnudo o con estereoscopio, una superficie de fractura dúctil de un material metálico, la textura es característica y se le suele llamar “fibrosa”, ya que se parece a la fractura de la madera. Mientras que la fractura frágil muestra una textura que se suele llamar en la literatura especializada, como “granular brillante”, debido a que es parecida a observar un conjunto de granos de arena.

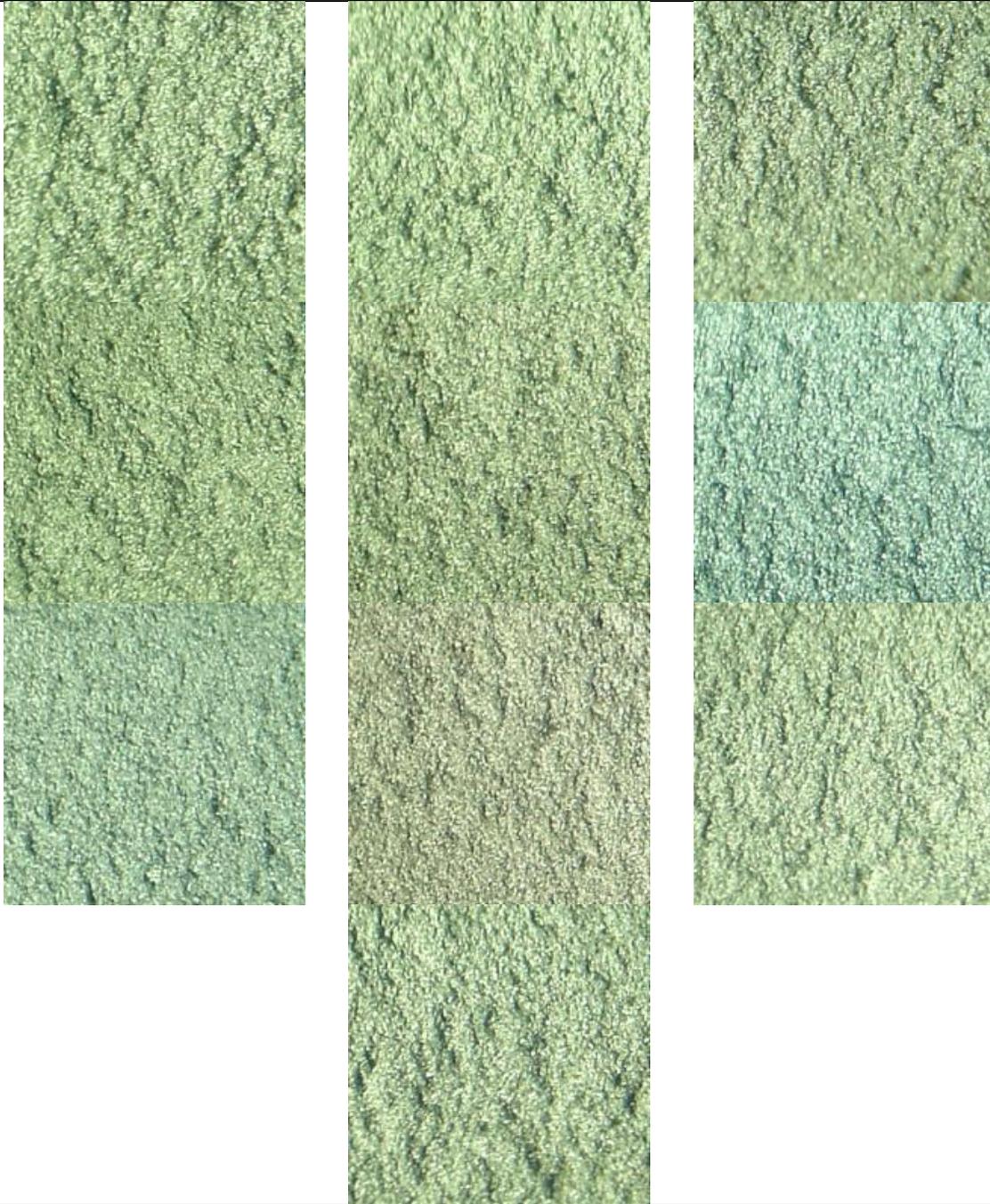
Las fracturas progresivas abarcan una variedad enorme de modos de falla, entre las cuales se tienen la fractura por fatiga, la fractura por termofluencia, la fractura por cooperación corrosión-esfuerzo, la fragilización dinámica, entre muchas otras. Todos estos tipos de fractura se caracterizan porque al pasar el tiempo, se va generando en servicio sobre la pieza en cuestión, un daño acumulado que progresivamente la agrieta hasta fracturarla. Dentro de las fracturas progresivas la más común es la fatiga, en la cual un elemento mecánico que experimenta una carga repetida en el tiempo, desarrolla y propaga grietas hasta fracturarse, todo esto debido a fenómenos a escala microscópica, de acumulación de daño por deformación plástica. La fatiga se presenta cuando el esfuerzo repetido aplicado en servicio, supera una fracción de la resistencia última del material, que para el caso de los aceros es  $0,5 \sigma_U$ . Una superficie de fractura por fatiga en metales al inspeccionarse a ojo desnudo o con estereoscopio, muestra una textura tersa, que en general es menos rugosa que la fibrosa de la fractura dúctil o la granular de la fractura frágil.

Para este trabajo se utilizaron imágenes de superficies de fractura de componentes mecánicos hechos en aceros, que se pueden agrupar en una de tres categorías a saber: Fractura Dúctil, Fractura Frágil y Fractura por Fatiga. Las imágenes fueron tomadas a color con cámara digital en formato JPG, por el autor de este trabajo, bajo inspección visual y con estereoscopio, en aumentos que oscilaron entre 1X y 7X. En total por cada categoría se usaron 10 imágenes a color de 200 x 200 píxeles. En la **Tabla 1.1** se muestran las imágenes correspondientes a fracturas dúctiles, en la **Tabla 1.2** a las de frágiles y en la **Tabla 1.3** a las de fatiga.

**Tabla 1.1:** Imágenes de las superficies de fractura dúctil (textura fibrosa). Tomadas por el autor.



**Tabla 1.2:** Imágenes de las superficies de fractura frágil (textura granular). Tomadas por el autor.



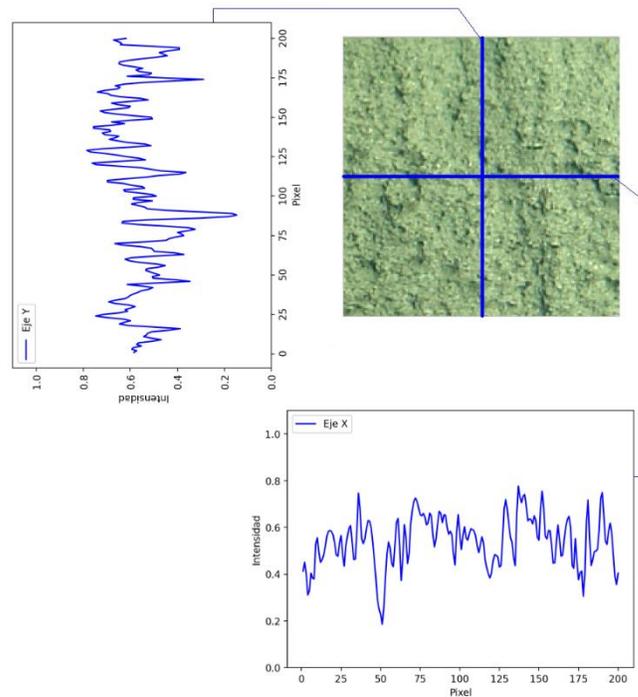
**Tabla 1.3:** Imágenes de las superficies de fractura por fatiga (textura tersa). Tomadas por el autor.



## 1.2 Datos funcionales

Crear algoritmos de identificación o clasificación de imágenes que utilicen como dato funcional toda la foto, es lo más común y en principio sería lo más sencillo, sin embargo en este trabajo se quiso extraer de cada imagen varios datos funcionales, correspondientes a perfiles de intensidad de luz medidos a lo largo de cada línea de pixeles, en el eje X e Y de la foto, **Figura 1-1**. Teniendo en cuenta que las imágenes de partida son de 200 x 200 pixeles, esto se traduce en que por cada foto se obtuvieron 400 curvas o datos funcionales (después del suavizamiento), es decir en total se tienen 4000 datos funcionales por cada una de las tres categorías de interés. En el capítulo 5 se explica por qué se decidió usar datos funcionales de la forma descrita.

**Figura 1-1:** Ejemplo de datos funcionales extraídos de las imágenes.

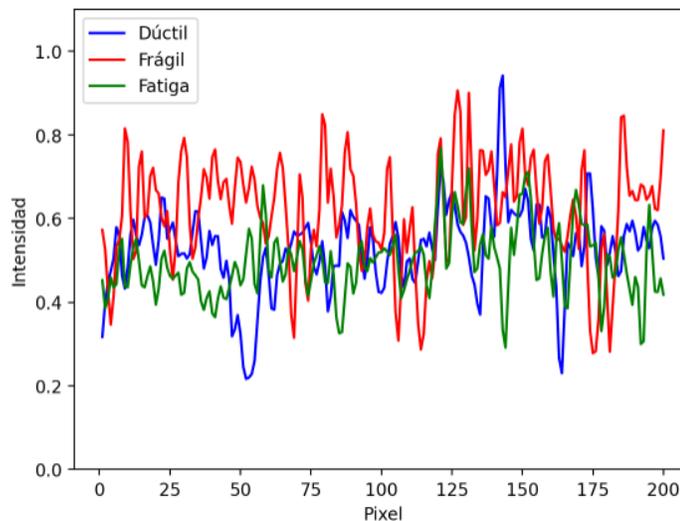


## 1.3 Estadística descriptiva

En la **Figura 1-2** se muestran tres datos funcionales de ejemplo pertenecientes a categorías diferentes, donde a simple vista no son detectables diferencias marcadas en

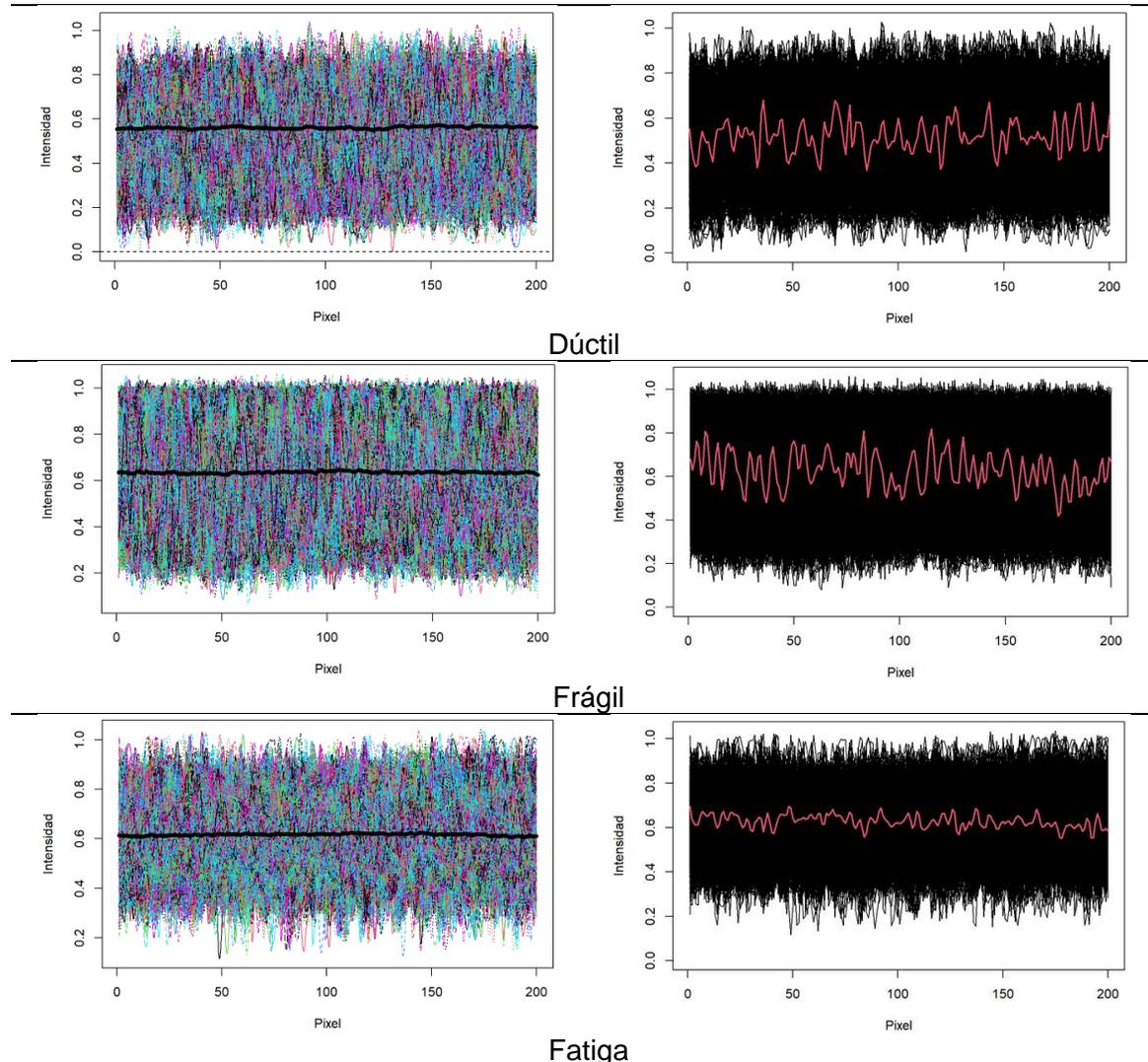
nivel o forma. Al calcular la función media por cada categoría de los datos funcionales, **Tabla 1.4** columna izquierda, se nota que tiende a ser una línea horizontal de nivel de luminosidad, donde dicho nivel en general es mayor en la categoría frágil, seguida de fatiga y por último la dúctil. La función mediana de la categoría fatiga, presenta una amplitud menor de oscilación respecto a las otras dos categorías, **Tabla 1.4** columna derecha. Lo anterior se confirma comparando todas las curvas y sus medianas por parejas de categorías, **Tabla 1.5**. Los datos funcionales se representaron en una base de 150 B-Splines de orden cúbico, con lambda obtenida por validación cruzada generalizada (VCG): 0,02592944 para imágenes dúctiles y frágiles, y 0,04175319 para imágenes de fatiga.

**Figura 1-2:** Ejemplo de datos funcionales de cada categoría (eje horizontal en pixeles y vertical en intensidad de luz normalizada). Base de 150 B-Splines de orden cúbico, lambda obtenida por VCG.



Al calcular la función de covarianza muestral, se resalta que los valores de covarianza son pequeños respecto a los de las varianzas en las tres categorías, lo que sugiere en general que se tienen bajas autocorrelaciones a altos rezagos espaciales. Además, el valor de las varianzas en la categoría frágil es en promedio mayor que en la dúctil, y a su vez ésta mayor que en la de fatiga, **Tabla 1.6**.

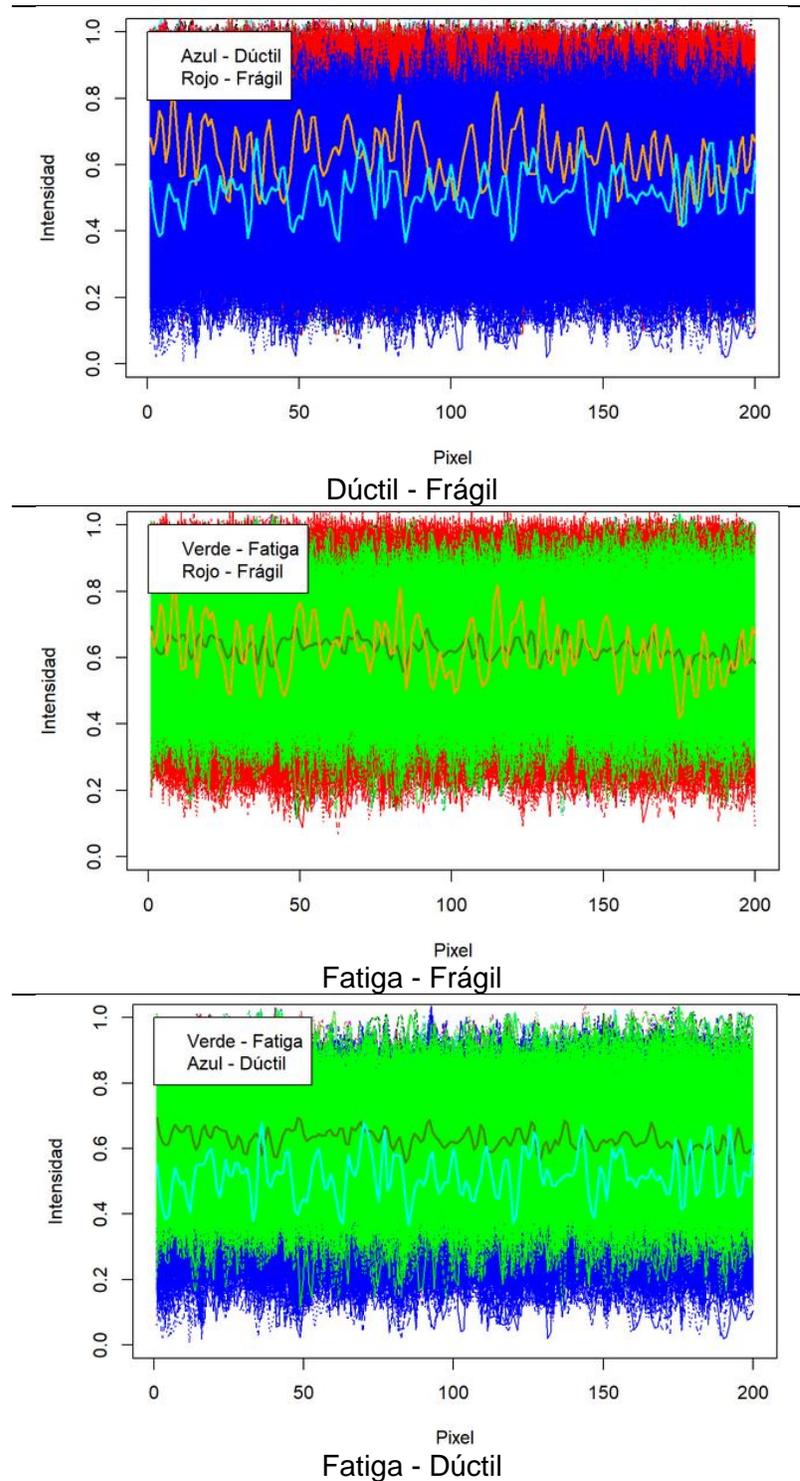
**Tabla 1.4:** Funciones media (izquierda) y mediana (derecha) de los datos funcionales usados. Base de 150 B-Splines de orden cúbico, lambda obtenida por VCG.

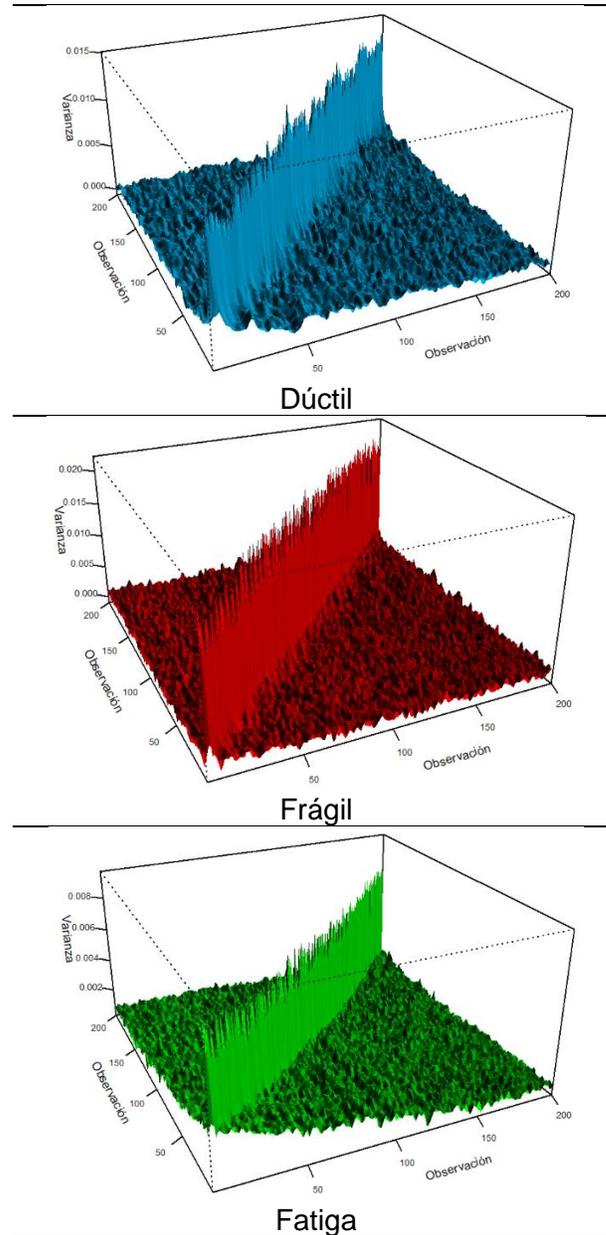


## 1.4 Pruebas de hipótesis sobre medias y operadores de covarianza

Las gráficas y estadísticas descriptivas mostradas sugieren, que pueden existir diferencias entre las funciones media y los operadores de covarianza entre categorías, por lo cual se adelantaron las pruebas estadísticas de igualdad de medias e igualdad de operadores de covarianza por parejas de categorías, obteniéndose los resultados de la **Tabla 1.7**.

**Tabla 1.5:** Comparación entre funciones y medianas por parejas de categorías.



**Tabla 1.6:** Funciones de covarianza por categorías.

En la Tabla 1.7 se puede ver que se rechazan las hipótesis múltiples nulas (alfa de 0,05/3), por lo tanto, hay diferencias estadísticas significativas en las funciones media y los operadores de covarianza, de al menos una categoría respecto a las otras dos, lo que

permite suponer en primera instancia que es posible construir algoritmos de clasificación supervisado para estos datos funcionales.

**Tabla 1.7:** Resultados de las pruebas de igualdad de medias e igualdad de operadores de covarianza entre las categorías.

Prueba de igualdad de medias		
$H_0: E(X_{Dúctil}) = E(X_{Frágil}) = E(X_{Fatiga})$		
$H_a: Al menos un valor esperado es diferente del resto$		
Dúctil - Frágil	Dúctil - Fatiga	Frágil - Fatiga
Se usaron los 126 primeros componentes principales, que capturan el 99,0344 % de la varianza, obteniéndose un Estadístico = 9911,567 y p-valor = 0	Se usaron los 121 primeros componentes principales, que capturan el 99,05286 % de la varianza, obteniéndose un Estadístico = 6143,969 y p-valor = 0	Se usaron los 128 primeros componentes principales, que capturan el 99,03965 % de la varianza, obteniéndose un Estadístico = 1067,006 y p-valor = 2,007078e-232
Prueba de igualdad de operadores de covarianza		
$H_0: C(X_{Dúctil}) = C(X_{Frágil}) = C(X_{Fatiga})$		
$H_a: Al menos un operador es diferente del resto$		
Dúctil - Frágil	Dúctil - Fatiga	Frágil - Fatiga
Se usaron los 125 primeros componentes principales que capturan el 99,06307 % de la varianza, obteniéndose un Estadístico = 21334121 y p-valor = 0	Se usaron los 122 primeros componentes principales que capturan el 99,03444 % de la varianza, obteniéndose un Estadístico = 3525551 y p-valor = 0	Se usaron los 129 primeros componentes principales que capturan el 99,06217 % de la varianza, obteniéndose un Estadístico = 21960330 y p-valor = 0

Las pruebas de igualdad de medias y de igualdad de operadores de covarianza entre datos funcionales tienen los siguientes supuestos: (i) distribución idéntica dentro de cada categoría y entre categorías, (ii) independencia dentro de cada categoría y del conjunto de curvas y (iii) proceso Gaussiano detrás de la generación de curvas para cada grupo. (i) y (ii) para la prueba de igualdad de medias y (iii) para la prueba de igualdad de operadores de covarianza.

Para la verificación del supuesto (i) se hace indirectamente a través de la verificación de igualdad de medias, usando la metodología de punto de cambio, con las primeras funciones y valores propios de cada uno de los grupos de datos en conjunto. Para el grupo dúctil las treinta primeras funciones propias explican el 65,3138 % de la varianza. Para el grupo frágil las treinta primeras funciones propias explican el 50,18721 % de la varianza.

Para el grupo fatiga las treinta primeras funciones propias explican el 72,92793 % de la varianza. Para todos los grupos las treinta primeras funciones propias explican el 61,03183 % de la varianza. Para el grupo de dúctil el estadístico de prueba arrojó un valor de 55,50913, para el grupo de frágil fue de 35,96292, para el grupo de fatiga fue de 197,7316 y para el conjunto de los datos 213,0117. El valor crítico para  $p=30$  y  $\alpha = 0,05$  es de 6,444772, es decir para los cuatro casos se rechaza la hipótesis nula, es decir no todos los valores esperados de las funciones son iguales.

Para la verificación del supuesto (ii) se usó la función “Independence\_test” de la librería “wvntests” del software R [\[R Core Team\]](#). Usando los tres primeros componentes principales y los 10 primeros rezagos se rechaza la hipótesis nula de iid para el grupo dúctil, frágil, fatiga y para el conjunto ( $p$ valores = 0).

Para la verificación del supuesto (iii) se hizo indirectamente a través de la verificación de la distribución normal de los puntajes de cada grupo, con la prueba de Shapiro-Wilk aplicada a los diez primeros puntajes (suman el 37,76304% de la varianza para dúctil, el 23,6256% de la varianza para frágil y el 34,5342% de la varianza para fatiga). En la mayoría de los casos (excepto en frágil en puntaje 7) se rechaza la hipótesis nula, es decir los puntajes no provienen de distribuciones normales, por tanto el proceso general de los espectros no es Gaussiano ( $p$ -valores menores a 0,05).

Como se verá más adelante en el documento, aun cuando los supuestos de las pruebas de igualdad de funciones media y de igualdad de operadores de covarianza no se cumplieron, los resultados que sugerían posibilidad de poder diferenciar las categorías por funciones media y/o por operadores de covarianza, fueron válidos. Esto indirectamente sugiere que estas pruebas son robustas ante el incumplimiento de supuestos, al menos en el caso de estudio de este trabajo.

## 1.5 Funciones de autocorrelación espacial intra datos funcionales

En este trabajo en algunos de los algoritmos de clasificación, se incluyó información de autocorrelación al interior de cada dato funcional, a partir del cálculo de la función de

autocorrelación (ACF) como si se tratase de una serie temporal. Al calcular estas funciones de autocorrelación espacial asociadas a cada dato funcional (intra) y compararlas por categorías, se notan tres aspectos: (a) mayores niveles de autocorrelación en los datos dúctiles, (b) en dúctil se pueden tener autocorrelaciones significativas incluso hasta el cuarto rezago o pixel, mientras en frágil y fatiga solo hasta el primero, y (c) en dúctil se nota cierto grado de estacionalidad significativo (ver **Tabla 1.8**). Esto sugiere que usar esta función de autocorrelación espacial como dato funcional, puede servir como característica o cofactor en los algoritmos de clasificación.

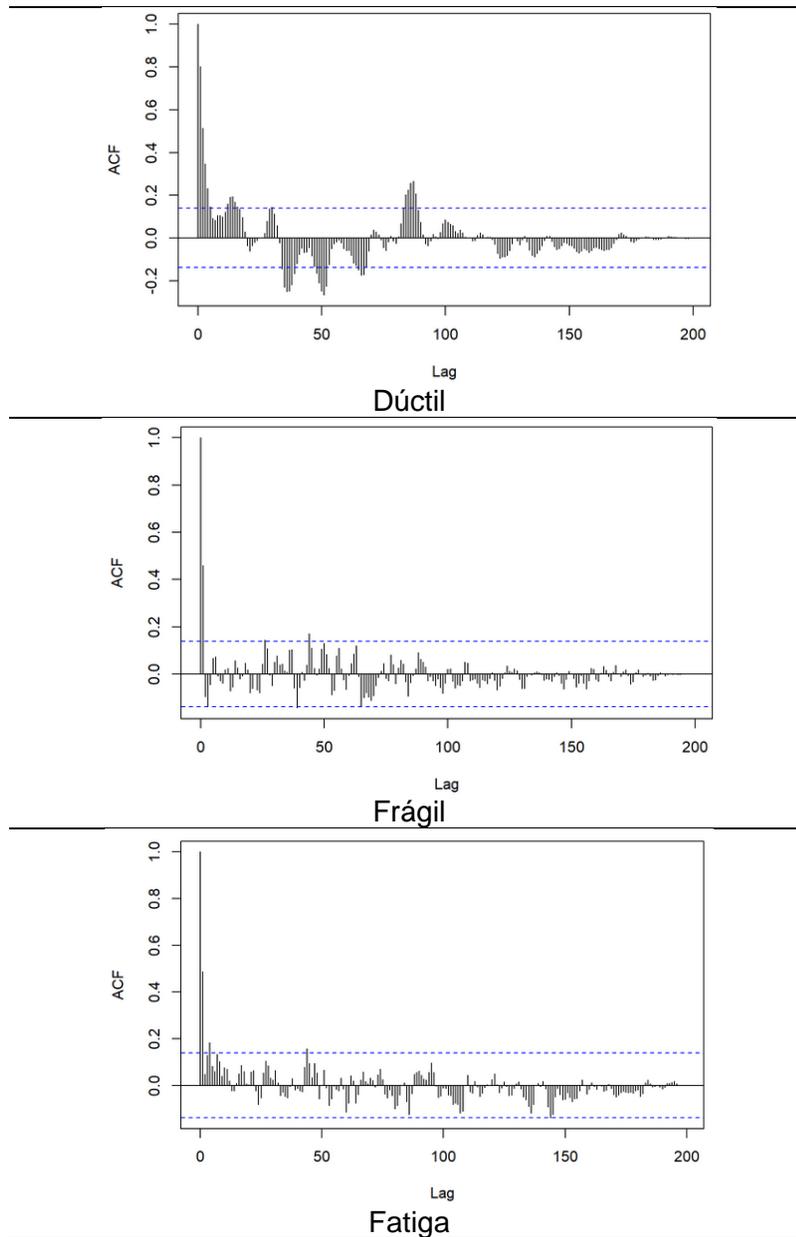
## 1.6 Análisis de componentes principales

Se hizo un ACPF sobre todo el conjunto de datos funcionales obtenidos de las imágenes, con el objetivo de identificar las funciones propias que recogen el mayor porcentaje de varianza, y verificar si de manera natural los datos funcionales forman grupos asociables a las categorías de fractura (dúctil, frágil y fatiga), **Figura 1-3**.

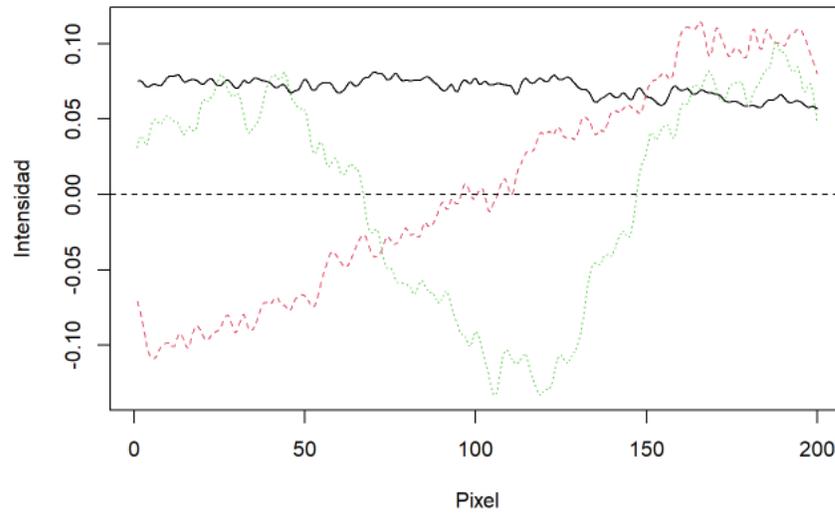
Los porcentajes de varianza que recogen las diez primeras funciones principales son: 16,206436 3,280059 2,282260 2,036683 1,953137 1,792502 1,644910 1,640767 1,630118 y 1,617428. Esto indica que, para efectos de uso en clasificación, se requerirían muchas funciones propias para alcanzar por ejemplo porcentajes de explicación de varianza superiores al 80 %. Sin embargo, es de resaltar que la primera función propia recoge la misma varianza, que la explicada por las funciones propias segunda a novena. Por la forma que muestra la primera función propia (ver **Figura 1-3**), probablemente está asociada al multiplicarse por el puntaje respectivo, al brillo promedio de cada dato funcional.

Si se grafican los “puntajes” de todos los datos funcionales de la primera función propia y de la segunda, se nota que hay un agrupamiento sobre el primer eje principal por categorías, sin embargo, con un elevado nivel de traslape entre ellas, **Figura 1-4**. Al hacer las mismas gráficas de “puntajes”, entre la primera función propia y las demás hasta la décima, además por parejas de categorías, se hace visualmente evidente que es sobre la primera función propia, que se pueden detectar las diferencias entre los grupos, aunque con alto grado de traslape, **Tabla 1.9, Tabla 1.10 y Tabla 1.11**.

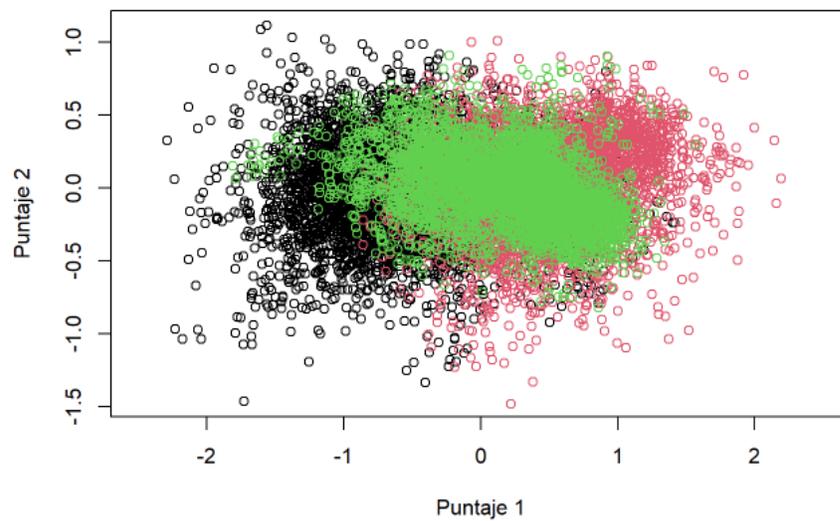
**Tabla 1.8:** Ejemplo de comparación por categorías, entre funciones de autocorrelación intra datos funcionales.



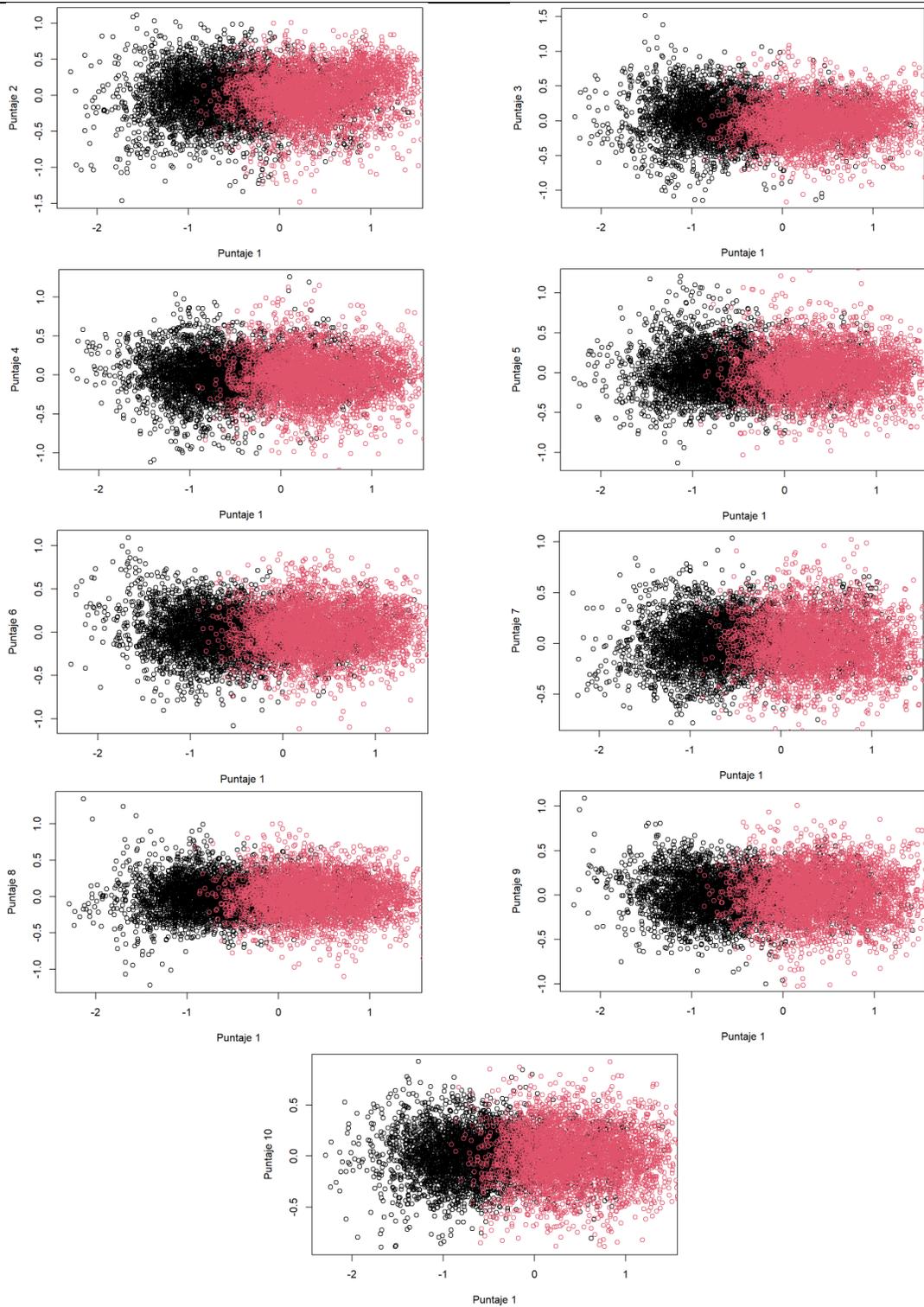
**Figura 1-3:** Primeras funciones propias. Primera función en negro, en rojo la segunda y en verde la tercera.



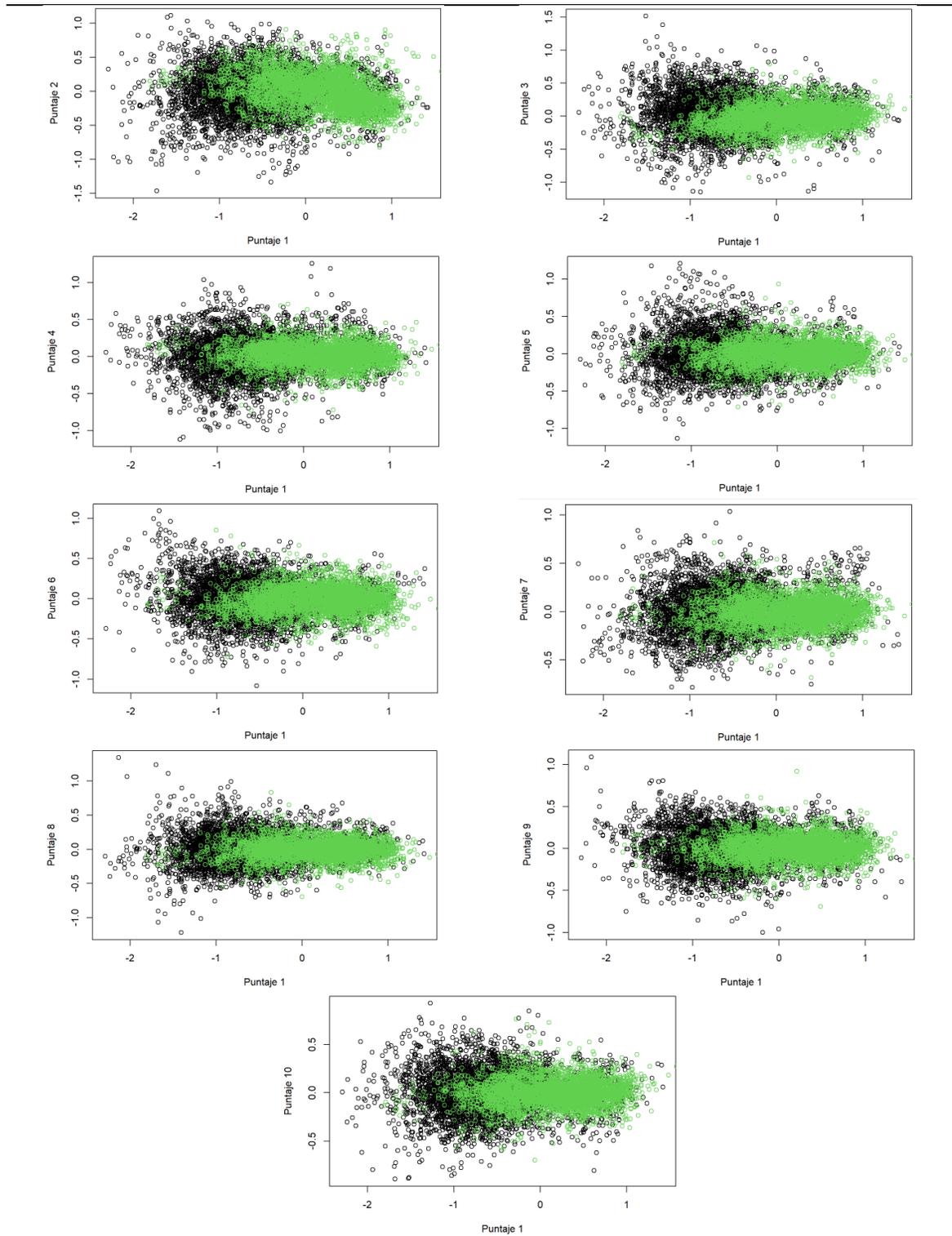
**Figura 1-4:** "Puntajes" sobre las funciones propias 1 y 2. Dúctiles en negro, frágiles en rojo y fatiga en verde.



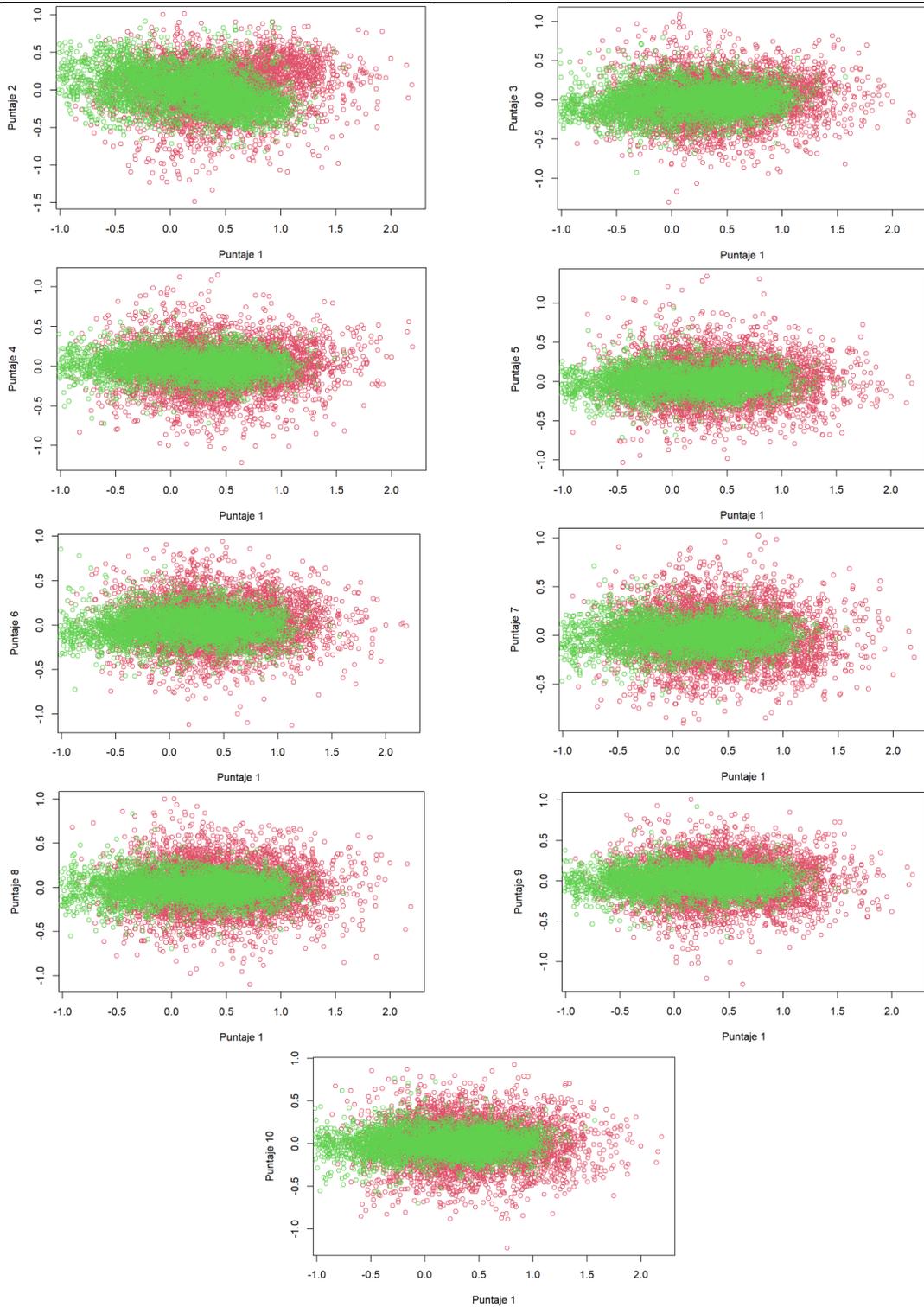
**Tabla 1.9:** Gráficas de "Puntajes" entre la primera función propia (eje horizontal) y las demás hasta la décima. En negro dúctiles y en rojo frágiles.



**Tabla 1.10:** Gráficas de "Puntajes" entre la primera función propia (eje horizontal) y las demás hasta la décima. En negro dúctiles y en verde fatiga.



**Tabla 1.11:** Gráficas de "Puntajes" entre la primera función propia (eje horizontal) y las demás hasta la décima. En verde fatiga y en rojo frágiles.



## 2. Revisión bibliográfica

Algunos trabajos previos relevantes para este proyecto se relacionan en el presente capítulo, unos relacionados con los datos funcionales y su clasificación, y otros orientados a la clasificación de imágenes en general.

### 2.1 Análisis y clasificación de datos funcionales

[\[Blanquero et al., 2019\]](#) En este trabajo se comenta que a partir de datos funcionales (series temporales), el análisis de datos se puede hacer a partir de la reducción de su dimensionalidad, a través de análisis de componentes principales funcionales (ACPF), mínimos cuadrados parciales (PLS), B-splines, entre otros. No obstante, los autores proponen como predictor, un subgrupo de datos alrededor de un tiempo, donde la selección de esos datos debería hacerse de tal manera que se obtenga el máximo de información de la serie original. Los autores proponen que la selección de dichos tiempos se base en derivadas de la serie de tiempo (datos funcionales derivados), asegurando que se maximice la correlación entre la clase y el valor predicho por una máquina de soporte vectorial (MSV).

[\[Yeonjoo et al., 2019\]](#) En este trabajo los autores proponen un método de clasificación de datos funcionales, que combina un clasificador Bayes con un clasificador basado en un modelo de efectos mixtos semiparamétrico, que hace la clasificación robusta ante curvas atípicas, y que está pensado específicamente para datos funcionales irregulares. Obteniéndose resultados comparables con los métodos de clasificación convencionales para datos funcionales, como el clasificador de Bayes, análisis de discriminante cuadrático, MLG, análisis de discriminante lineal y ACPF (errores de clasificación entre el 11 y el 15%). En los datos funcionales irregulares son muy útiles las etapas de reducción como el ACPF.

[\[Yinfeng et al., 2016\]](#) En este trabajo los autores comentan que en el análisis de datos funcionales, normalmente primero se convierten en datos multivariados y segundo mediante un algoritmo de aprendizaje de máquina se pueden realizar clasificaciones de nuevos datos. Estos métodos están basados en distancias euclidianas, pero las distancias entre datos funcionales se representan en  $L_2$ , luego ello puede genera problemas en la aplicación. Al respecto los autores concluyen que los datos funcionales se pueden representar en un espacio euclidiano si la base es ortonormal, los datos periódicos estables se pueden representar en una base de Fourier, la base de Wavelet es más adecuada para funciones periódicas irregulares o con diferencias locales y el análisis de componente principales puede ser adecuado si no se tiene ningún tipo de información previo sobre la forma de los datos funcionales. Acá usaron clasificador de MSV con núcleo Gaussiano, bosque aleatorio, regresión logística, K-vecinos más cercanos y red neuronal Convencional, con varias bases de curvas de acceso libre. Usando representación en base de Fourier, de componentes principales, ondetas y base no ortonormal. Encontraron que los clasificadores MSV, bosque aleatorio y red neuronal son los que generalmente llegan a las mayores exactitudes. En general la base de funciones principales da buenos resultados en la clasificación. Se llegaron a obtener exactitudes por encima de 96%.

[\[Alonso et al., 2012\]](#) En este trabajo los autores transforman el problema de clasificación de datos funcionales a un problema de discriminante clásico multivariado (basado en distancias), reduciendo la dimensionalidad y usando derivadas de las curvas, obteniéndose así una combinación lineal de variables y coeficientes, donde el análisis de discriminante lineal determina la combinación. La propuesta consiste en considerar como variables las distancias entre una nueva curva del dato funcional y sus derivadas. El método propuesto presentó exactitudes de clasificación entre el 70 y el 95%.

[\[Llop et al., 2008\]](#) P. Llop et al. En este trabajo se comenta que el tratamiento de datos funcionales normalmente empieza con una regularización, que consiste el bajar la dimensionalidad o suavizar la curva (lo que excluye información). Los autores proponen trabajar con los datos sin suavizar, es decir sin pretratamiento, para ello se estima una función de densidad con el método K-vecinos más cercanos. Proponen un modelo de proceso estocástico, donde alcanzaron exactitudes del 95% con datos simulados.

En [\[Nilsback et al., 2008\]](#) se usaron varias funciones núcleo para caracterizar la textura, el borde, la distribución de pétalos y el color de fotografías de flores, para con base en ellas usar una MSV para clasificarlas en una base con 103 clases. En el estudio concluyen que combinar núcleos de varias clases asociados cada uno a una característica diferente de la flor, mejora el desempeño de la clasificación del 55,1 al 72,8%.

[\[Xinyu et al., 2023\]](#) En este trabajo se propone un modelo de clasificación de datos funcionales, basado en distancia de Mahalanobis y núcleos regenerativos. Al aplicar este algoritmo a un conjunto de datos funcionales, encuentran una exactitud de clasificación de 88% vs una exactitud del 68% al usar distancia euclidiana.

[\[Shuoyang et al., 2021\]](#) Los autores proponen dos métodos de clasificación de datos funcionales, uno basado en análisis discriminante y el otro en una red neuronal profunda. El entrenamiento y la prueba de los modelos de clasificación, se hizo con curvas generadas de un proceso Gaussiano. Las exactitudes de clasificación estuvieron entre el 79 y 99%.

[\[Delaigle et al., 2012\]](#) Acá los autores muestran que en problemas de clasificación de datos funcionales, se pueden alcanzar asintóticamente exactitudes de clasificación muy altas (perfectas), gracias a que tienen intrínsecamente alta dimensionalidad. Esta alta exactitud se puede lograr con modelos lineales (por ejemplo modelos basados en distancias a centroides y en análisis de discriminante lineal), al contrario de la clasificación de análisis multivariado donde estos modelos pueden ser ineficientes. Se tiende a la alta exactitud cuando las muestras de entrenamiento son grandes, lo cual no ocurre en el análisis multivariado.

[\[Oviedo et al., 2011\]](#) Este artículo está escrito por uno de los autores del paquete fda.usc de R. Acá se explica que en el paquete está implementado un clasificador basado en MLG y otro en MAG, donde en este último se permiten predictores no lineales a través de una función de suavizamiento. A través de un ejemplo de regresión logística se muestra que tanto MLG o MAG pueden lograr altas exactitudes de clasificación (encima del 76%). Para el ejemplo concreto en que los implementaron, MAG es ligeramente más exacto que MLG cuando se usa la base de componentes principales, y al contrario cuando se usa la base B-Splines.

## 2.2 Clasificación de imágenes

En [\[Jiaohua et al., 2020\]](#) se usa una red neuronal convolucional, para clasificar imágenes del área biológica, proponiendo usar imágenes de entrenamiento de gran tamaño y bloque de residuos invertidos para aumentar la exactitud de la clasificación y hacer la red más liviana, que es uno de los problemas principales de este tipo de red (su costo computacional).

En [\[Krizhevsky et al., 2012\]](#) diseñaron un clasificador con una red neuronal convolucional profunda, para imágenes de la base ImageNet LSVRC-2010, con imágenes de 1000 clases. Obteniéndose errores del 37,5%, que mostraron mejor desempeño que clasificadores tradicionales (por encima de SIFT y vectores de Fisher) y redes neuronales convencionales que aprenden más lento. La red consistió de 60 millones de parámetros, 650000 neuronas 5 capas de convolución.

[\[Kumar et al., 2012\]](#) En este trabajo se usó una red neuronal convolucional mejorada para diagnosticar defectos en componentes de bombas centrífugas, a partir de imágenes acústicas. Las imágenes acústicas son obtenidas a partir de señales acústicas usando la transformada analítica Wavelet (AWT), posteriormente se modificó la versión de entropía basada en divergencia que se encuentra en la función de costo de la red, con el objeto de no hacer activaciones redundantes hacia capas más profundas de la red neuronal, disminuyendo así el número de parámetros de entrenamiento. La modificación mejoró la exactitud de la red llegando a exactitudes de clasificación del 100%.

[\[Radhesyam et al., 2020\]](#) Los autores realizan clasificación a partir de imágenes hiperespectrales de tres bases de datos (Indian Pines, Pavia University and Salinas). Para reducir el costo computacional las imágenes primero se someten a un proceso de normalización reduciendo sus valores escalares y aplicando un filtro de Gabor (para extraer características espaciales), posteriormente se realiza un análisis de componentes principales probabilísticos, finalmente los datos espaciales y espectrales se integran para formar características. A partir de lo anterior se emplea una red neuronal CNN para hacer la clasificación. El método mostró un aumento de exactitud del 98 al 99% en promedio.

[\[Zhao et al., 2020\]](#) Los autores comentan que para la clasificación de defectos en textiles, se han usado métodos basados en clasificadores bayesianos, K-vecinos más cercanos, MSV, redes neuronales y características locales binarias. Los defectos en textiles pueden ser difíciles de identificar porque tienden a esconderse con el fondo de la imagen. Los autores proponen una red neuronal de tres sistemas en paralelo que imitan la forma en que opera la visión humana. Uno llamado VP que corresponde al comparar la imagen con características para poderla clasificar, y otras llamadas VLTM y VSTM que son memorias de largo y corto plazo, en donde la clasificación se basa en “recuerdos”, figura 26. El método de clasificación propuesto mostró mejor desempeño, aumentando la exactitud del 80 al 97% aproximadamente.

[\[Waqar et al., 2019\]](#) Los autores proponen una red neuronal mejorada y optimizada para correr en una computadora personal y poder hacer monitoreo en línea de defectos en celdas fotovoltaicas (una red ligera), a partir de imágenes de electroluminiscencia. Probaron varias arquitecturas de red, números de capas, números de filtros, etc., que les permitió llegar a la de mejor desempeño con bajo costo computacional, Otra estrategia que utilizaron fue la de aumentar los datos disponibles en las imágenes, como por ejemplo rotarlas o reflejarlas, así de una misma imagen la multiplican varias veces para entrenar modelos. Con los datos evaluados pasaron de exactitudes del 88 al 93%.

[\[Watanabe et al., 2019\]](#) Los autores desarrollaron un sistema de clasificación de imágenes en línea con base en una red neuronal convolucional, para clasificar imágenes de microscopia de barrido acústico de diodos de potencia, en normales y anormales producto del uso. La red desarrollada se basó en la VGG16. Por lo reducido del número de imágenes disponibles para entrenar, se hizo validación cruzada comparando la red sin entrenamiento, con entrenamiento y entrenamiento más traslape, siendo mejor el desempeño en el último caso. Se obtuvieron exactitudes de clasificación entre el 92 y el 97%.

[\[Chen et al., 2021\]](#) Los autores hacen una revisión sobre los algoritmos basados en redes neuronales convolucionales, usados para clasificación de imágenes y comentan que este tipo de clasificadores, se han posicionado como los más comunes desde el año 2012 para este propósito, llegando a alcanzar altas exactitudes de clasificación. Sin embargo,

actualmente se busca bajar la complejidad de las redes y por tanto su costo computacional, para hacerlas más accesibles, pero ello implica perder exactitud.

[[Ashraf et al., 2012](#)] En este artículo se resalta que en los métodos “clásicos” de clasificación de imágenes, a partir de la imagen se extraen características de bajo nivel mediante algún procedimiento y posteriormente sobre ellas el algoritmo de clasificación se entrena. Mientras que en métodos de aprendizaje profundo como las redes neuronales convolucionales, el algoritmo a través de la red encuentra características de alto nivel y se entrena.

[[Krishna et al., 2018](#)] En este trabajo se prueba la arquitectura AlexNet con una red neuronal convolucional, para verificar si es lo suficientemente potente de distinguir imágenes del mismo objeto, que han sido rotadas, recortadas o distorsionadas. Encontrando que la red es robusta para poder identificar las imágenes.

[[Sheykhmousa et al., 2020](#)] Se compara el uso y desempeño de las máquinas de soporte vectorial y los bosques aleatorios, en la clasificación de imágenes de sensado remoto. Hacen notar que estas dos técnicas de clasificación al igual que las redes neuronales, son de difícil interpretabilidad, pero en general de alta exactitud, solo superadas por las redes de aprendizaje profundo. En el caso de imágenes de baja resolución espacial el bosque aleatorio se comporta mejor que la MSV, pero la MSV supera al bosque cuando haya más características.

[[Sabat et al., 2020](#)] Se aplicaron algoritmos de clasificación de imágenes aéreas, de la presencia de plantas invasoras en hábitats naturales. Como algoritmos de clasificación se usaron bosque aleatorio y una MSV con núcleo RBF. Dependiendo la planta invasora, algunas veces fue más exacta la clasificación con MSV y otras con bosque aleatorio, pero siempre mayor que el 79%.

## **3. Objetivos**

En el presente trabajo se buscó alcanzar los siguientes objetivos.

### **3.1 Objetivo general**

Comparar el desempeño de algoritmos de clasificación de imágenes, que permitan identificar fotografías de superficies de fractura en elementos mecánicos metálicos, formadas por fatiga mecánica, fractura súbita dúctil y fractura súbita frágil, usándose técnicas de clasificación basadas en análisis de datos funcionales.

### **3.2 Objetivos específicos**

- Extraer a partir de imágenes de entrenamiento datos funcionales y estadísticos que permitan diferenciar el tipo de fractura presente (fatiga mecánica, fractura súbita dúctil y fractura súbita frágil).
- Implementar algoritmos basados en técnicas de clasificación supervisada, que usen los datos funcionales y sus estadísticos, con y sin la inclusión de la autocorrelación espacial.
- Aplicación de los algoritmos a imágenes de entrenamiento y de prueba para determinar la clasificación generada por cada uno.
- Encontrar la matriz de confusión para cada algoritmo, de tal manera que se pueda evaluar cuál tiene el mejor desempeño y si la inclusión de la autocorrelación espacial mejora la precisión en la clasificación de las imágenes con datos funcionales.



## 4. Marco teórico

En este capítulo se incluyen algunos elementos básicos de la teoría asociada a los datos funcionales, y a los algoritmos de clasificación que se emplearon.

### 4.1 Datos funcionales

La información base para este numeral se tomó de [\[Ramsay et al., 2005\]](#). El análisis de datos funcionales ADF, contiene los métodos estadísticos para tratar datos que no son números reales o vectores sino curvas o funciones. En este caso una muestra consiste de  $n$  funciones provenientes de un espacio de Hilbert ( $x_1 = x_1(t), \dots, x_n = x_n(t)$ ). Antes de aplicar técnicas de análisis de los datos funcionales, es conveniente realizar un tratamiento preliminar que puede buscar: bajar la dimensionalidad o quitar ruido de las curvas (se conoce como suavizamiento). La versión suavizada de las curvas se expresa por la Ecuación 4.1.

$$\tilde{x}(t) = \sum_{j=1}^J c_j e_j(t) \quad (4.1)$$

donde  $x \in L^2$  que es el espacio de las funciones cuadrado integrables y  $e_j(t)$  es la base de funciones.  $j$  es el número de funciones base, que normalmente es más pequeño que  $n$  (reduce la dimensionalidad).  $c_j$  son los coeficiente de Fourier y se obtiene de minimizar la Ecuación 4.2.

$$\sum_{k=1}^N (x(t_k) - \sum_{j=1}^J c_j e_j(t_k))^2 \quad (4.2)$$

La transformación o suavizamiento se puede resumir en:  $(x(t_1), \dots, x(t_N)) \rightarrow (c_1, \dots, c_J) \rightarrow (\tilde{x}(t_1), \dots, \tilde{x}(t_N))$ . En este trabajo se usaron bases B-Spline, que son polinomios

suavizados, cuyos parámetros son grado del polinomio, número de funciones de la base y número de nodos. Esta base en particular es no ortogonal.

A partir de una muestra de datos funcionales la función media se estima con la ecuación 4.3, la función de varianza con la ecuación 4.4, la función de covarianza con la ecuación 4.5 y la función de correlación con la ecuación 4.6, para cualquier pareja  $t_1, t_2$ .

$$\hat{\mu}(t) = N^{-1} \sum_{i=1}^N x_i(t) \quad (4.3)$$

$$var_x(t) = (N - 1)^{-1} \sum_{i=1}^N (x_i(t) - \hat{\mu}(t))^2 \quad (4.4)$$

$$cov_x(t_1, t_2) = (N - 1)^{-1} \sum_{i=1}^N (x_i(t_1) - \hat{\mu}(t_1)) (x_i(t_2) - \hat{\mu}(t_2)) \quad (4.5)$$

$$corr_x(t_1, t_2) = \frac{cov_x(t_1, t_2)}{\sqrt{var_x(t_1) var_x(t_2)}} \quad (4.6)$$

A través de un ACPF se pueden encontrar las funciones propias, valores propios y puntajes de cada dato funcional. Con lo anterior se pueden representar los datos funcionales en la base de funciones propias, la cual es ortonormal a través de la expansión de Karhunen-Loève (Ecuación 4.7). Donde las  $v_j$  son las funciones propias, los  $\xi_j$  son los puntajes, los cuales se obtiene del producto punto entre la función y la función propia ( $\xi_j = \int x_1(t)v_j(t)$ ,  $E(\xi_j) = 0$ ,  $E(\xi_j^2) = \lambda_j$ ) y los  $\lambda_j$  son los valores propios. Nótese que el ACPF se hace con los datos funcionales centrados, es decir con la media sustraída.

$$x_i(t) = \hat{\mu}(t) + \sum_{j=1}^{\infty} \xi_j v_j(t) \quad (4.7)$$

En los métodos de clasificación funcional de datos supervisados la variable aleatoria  $X$  puede ser observada en dos poblaciones  $P_0$  y  $P_1$ ,  $X|Y = j$ ,  $j = 0$  o  $1$ ,  $\{(X_i, Y_i), 1 \leq i \leq n\}$  e indica la pertenencia a la población 0 o 1. Para cada nueva imagen la regla de clasificación debe minimizar al menos asintóticamente el error de clasificación o riesgo  $P(g(X) \neq Y)$ . La regla de clasificación óptima es la regla de Bayes  $g^*(x) = I_{(\eta(x) > 1/2)}$ , donde  $\eta(x) = E(Y|X = x)$ . El error mínimo de clasificación  $L^* = P(g^*(X) \neq Y)$  se deduce a partir de los datos de entrenamiento.

## 4.2 Clasificadores basados en distancias

En este grupo de modelos la clasificación de cada dato funcional, se hace por la cercanía que tenga (distancia mínima), a un dato funcional representativo de una categoría dada. Algunos de los clasificadores evaluados usaron (a) distancias euclidianas después de ACPF entre el dato funcional y la función media de cada categoría, (b) distancias  $L2$  o de Mahalanobis entre el dato funcional y la función media de cada categoría, o (3) distancias Procrustes entre la matriz de covarianza de cada foto y la matriz de covarianza de cada categoría. A continuación se mencionan algunas de las medidas usadas.

Distancia entre dos funciones en  $L2$  (espacio de las funciones cuadrado integrables): Corresponde a la norma de su diferencia (ecuación 4.8, donde  $\|h\| = \sqrt{\langle h, h \rangle} = (\int h^2(t) dt)^{1/2}$ ). Esta distancia es equivalente a la distancia euclidiana si la base empleada en la representación es ortogonal.

$$d(f, g) = \|f - g\| \quad (4.8)$$

Distancia de Mahalanobis: Tiene en cuenta la estructura de correlación subyacente en los datos. En el contexto multivariado se haya con la ecuación 4.9, mientras en el contexto funcional se puede usar la distancia de Mahalanobis generalizada, ecuación 4.10 [\[Ghigliet et al., 2017\]](#).

$$M(x, y) = ((x - y)' \Sigma^{-1} (x - y))^{1/2} \quad (4.9)$$

$$\hat{d}_p^2(x_i(t), x_j(t)) = \sum_{k=1}^{\min\{n-1, P\}} \hat{d}_{M,k}^2(x_i(t), x_j(t)) \hat{h}_k(p) + \sum_{k=\min\{n-1, P\}+1}^P p((x_i(t) - x_j(t), \hat{v}_k))^2 \quad (4.10)$$

En la ecuación 4.10 las  $x_i(t)$ ,  $x_j(t)$  son las funciones entre las cuales se toma distancia,  $P$  la longitud de la grilla de variables independientes,  $\hat{h}_k$  es una función de regularización,  $\hat{d}_{M,k}$  es la contribución a la distancia a lo largo del  $k$ -ésimo componente,  $n$  es el número de

muestras (curvas), las  $\hat{v}_k$  son las funciones propias y  $k$  es el número de funciones propias usadas

Distancia Procrustes: Es este caso se usó una versión para datos multivariados [\[Dryden et al., 2009\]](#), dada en la ecuación 4.11. Donde  $S_1$  y  $S_2$  son las matrices de covarianza de tamaño  $k$ , las  $L_i$  son las descomposiciones de las matrices de covarianza por empleo del tipo Cholesky  $L_i = chol(S_i)$ ,  $i = 1, 2$ .  $S = (LR)(LR)'$  para cualquier  $R \in O(k)$ .

$$d_S(S_1, S_2) = \inf_{R \in O(k)} \|L_1 - L_2 R\| \quad (4.11)$$

### 4.3 Clasificadores basados en MLG y MAG

Para este numeral y el siguiente, se usó la información de apoyo de la Liberia *fda.usc* de  $R$  [\[Febrero et al., 2012\]](#). El clasificador de MLG usa un predictor lineal como el mostrado en la ecuación 4.12, donde  $g^{-1}$  es la inversa de la función de enlace, las  $z^p$  son covariables no funcionales, las  $x_i^q(t)$  son covariables funcionales y los  $\epsilon_i$  son los errores aleatorios. En el MAG el predictor es el mostrado en la ecuación 4.13, donde ya no es lineal en los parámetros sino que se reemplazan los betas por funciones de suavizamiento  $f$ . Estos modelos pueden utilizar varias distribuciones candidato para modelar el error, lo cual depende de la naturaleza de la variable respuesta y a su vez varias funciones de enlace. En la **Tabla 4.1** se mencionan los modelos de distribución de error usados y sus funciones de enlace.

$$y_i = g^{-1} \left( \alpha + \beta_1 z_i^1 + \dots + \beta_p z_i^p + \int_{T_1} x_i^1(t) \beta_1(t) dt + \dots + \int_{T_f} x_i^q(t) \beta_q(t) dt \right) + \epsilon_i \quad (4.12)$$

$$y_i = g^{-1} \left( \alpha + f_1(z_i^1) + \dots + f_p(z_i^p) + \int_{T_1} f^1(x_i^1(t)) dt + \dots + \int_{T_f} f^q(x_i^q(t)) dt \right) + \epsilon_i$$

$$(4.13)$$

## 4.4 Clasificadores basados en medidas de profundidad

También se puede usar como método de clasificación la profundidad  $D(x)$ , que indica que tan profundo está un dato funcional  $x$  dentro de una distribución. En el contexto de clasificación indica que tan profundo está un nuevo dato funcional en las poblaciones  $P_0$  y  $P_1$  o más si hay más categorías, asignándose donde la profundidad es mayor. Este método no es tan eficiente cuando los datos son heterocedásticos. La medida de profundidad permite generar estadísticas de orden, rangos y la medida de centralidad (la mediana). La mediana es la función con máxima profundidad dentro de una distribución de datos funcionales. La mediana se define como se indica en la ecuación 4.14.

**Tabla 4.1:** Modelos de error y funciones de enlace usados en MLG y MAG.

Modelo del error	Función de enlace usada	Tipo de variable para la que se aconseja
Binomial	logit	Tipo éxito o fracaso
Gaussiano	Identidad	Continua
Poisson	log	Conteos
Quasi	Identidad	Con sub o sobredispersión
Quasi-Binomial	logit	Tipo éxito o fracaso con sub o sobre dispersiones
Quasi-Poisson	log	Conteos con sub o sobre dispersiones

$$y_{0,5} = \underset{y \in \{x_1, \dots, x_n\}}{\operatorname{arg\,max}} D(x) \quad (4.14)$$

Para realizar la clasificación se usaron las siguientes medidas de profundidad: (1) Fraiman y Muniz - FM, (2) proyectada aleatoria - RP, (3) proyectada doblemente aleatoria - RPD, es decir de las curvas y sus derivadas, (4) Tukey aleatoria - RT, (5) h-Modal y (6) Extrema.

En la ecuación 4.15 está la versión muestral de la medida de profundidad FM, que está relacionada con que tan larga es una curva respecto a los demás datos. En esta ecuación  $i$  es la  $i$  – esima curva,  $n$  es el número de curvas con  $\{x_i(t_j); i = 1, 2, \dots, n, j = 1, 2, \dots, m\}$ ,

$\Delta_j = t_j - t_{j-1}$  y  $F_{n,t_j}(x_i(t_j))$  es la función acumulada de distribución empírica de los puntos  $x_1(t_j), x_2(t_j), \dots, x_n(t_j)$ .

$$FM D_n(x_i) = \sum_{j=2}^m \left[ 1 - \left| \frac{1}{2} - F_{n,t_j}(x_i(t_j)) \right| \right] \Delta_j \quad (4.15)$$

La ecuación 4.16 corresponde a la versión muestral de la medida de profundidad h-Modal. En este caso alta profundidad implica que la curva está más densamente rodeada por otras curvas, en esta ecuación  $i$  es la  $i$  -ésima curva respecto a las  $n$  curvas,  $h$  es el ancho de banda,  $K_h$  es una función núcleo y la doble barra indica norma.

$$M D_n(x_i, h) = \frac{1}{n} \sum_{k=1}^n K_h(\|x_i - x_k\|) \quad (4.16)$$

La profundidad proyectada aleatoria RP se calcula con la ecuación 4.17. Es una medida aleatoria, ya que depende de la selección de la dirección de proyección. En la ecuación  $v$  es la dirección aleatoria independiente de las  $x$ ,  $P_{v,i}$  son las proyecciones de las  $x$  en  $v$  ( $P_{v,i} = \langle v, x_i \rangle = \int v(t)x_i(t)dt$ ) y  $F_n(x) = \frac{1}{n} \sum_{i=1}^n I\{P_{v,i} \leq x\}$  es la distribución empírica de las Proyecciones. En la profundidad proyectada doblemente aleatoria RPD, se incluyen además de las curvas originales sus derivadas.

$$RP D_n(x_i, v) = \min\{F_n(P_{v,i}), 1 - F_n(P_{v,i})\} \quad (4.17)$$

La profundidad de Tukey aleatoria RT se calcula según las ecuaciones 4.18 y 4.19, donde  $v_i$  son vectores aleatorios,  $P$  es la probabilidad en  $R^P$  y  $\mu$  pertenece a  $R^P$ . Las  $\Pi_v$  son las proyecciones en un espacio unidimensional generado por  $v$  y  $P_\mu$ .

$$RT D_{k,v}(x, P) = \min\{TD_1(\Pi_{v_i}(x), P_{v_i}), i = 1, 2, \dots, k\} \quad (4.18)$$

$$T D(x, P) = \inf\{TD_1(\Pi_v(x), P_\mu): v \in R^P\} \quad (4.19)$$

La profundidad extrema de una función  $g$  dentro de una muestra de funciones  $S = \{f_1, \dots, f_n\}$ , se define por la ecuación 4.20 donde  $g \succcurlyeq f_i$  si se cumple  $g > f_i$  o  $g \sim f_i$ .

$$E D(g, S) = \frac{\#\{i: g \neq f_i\}}{n} \quad (4.20)$$

## 4.5 Clasificadores basados en aprendizaje de máquina

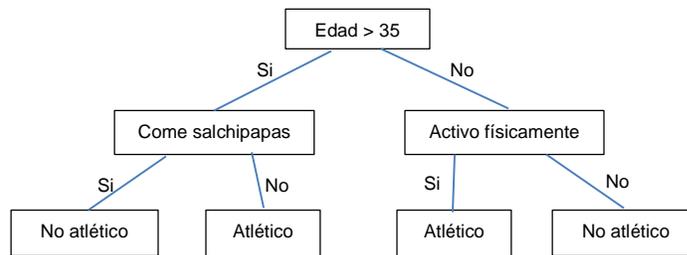
Los clasificadores de este grupo son aquellos que generan sus propias características o variables a partir de los datos para clasificar, por ello también a algunos de ellos se les conoce como de aprendizaje profundo. Como tal estos clasificadores no trabajan directamente con datos funcionales, para ello se suavizaron las curvas (con la base de 150 B-Splines) y posteriormente se les entregó la matriz de datos evaluada sobre una grilla. Estos algoritmos poseen unos hiperparámetros de ajuste del modelo, los cuales se deben ajustar por estrategias como validación cruzada con un determinado número de pliegues, para maximizar la exactitud de clasificación. A continuación se da la teoría básica de los algoritmos empleados, donde la principal fuente de información fue la documentación del paquete *Scikit-learn* de *Python* [[Pedregosa et al., 2011](#)] y [[Van Rossum et al., 1995](#)]

**Árbol de decisión:** es un algoritmo para tomar decisiones, basado en las variables de entrada, en este caso las decisiones son de clasificación. En cada nodo del árbol se toman decisiones y las ramas son consecuencias de esas decisiones. El árbol progresivamente va dividiendo los datos en grupos más pequeños, utilizando atributos como criterios de división. Los atributos que generan las divisiones se escogen de acuerdo a una medida de ganancia de información. Para evaluar qué atributo tiene mayor impacto en la ganancia de información (división), se tiene la opción de varios criterios, dentro de los cuales el criterio de Gini, es uno de los más comunes. El tamaño del árbol se define por su profundidad (número de niveles en que se hacen divisiones). Para este modelo los hiperparámetros son el criterio de división y la profundidad. En la **Figura 4-1** se presenta un ejemplo de un árbol de decisión, para determinar si se tiene un cuerpo atlético o no, en función de la edad, la actividad física y tipo de comida.

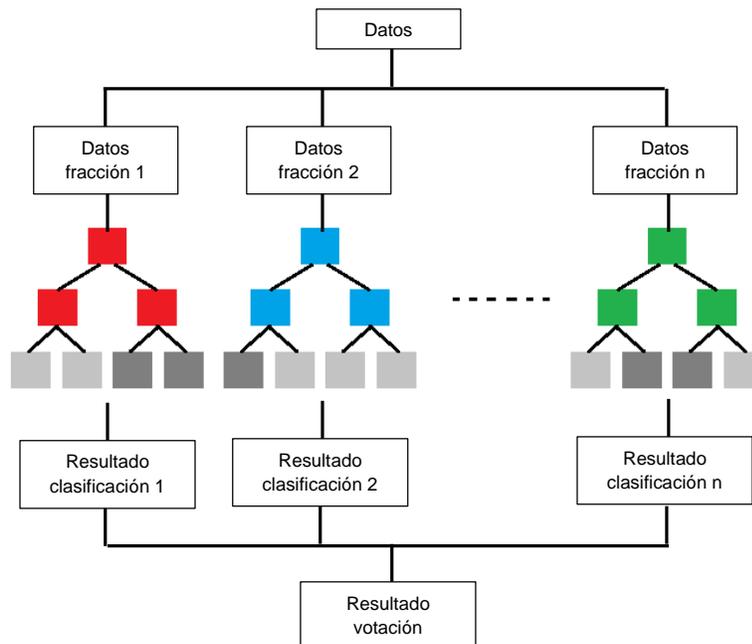
**Bosque aleatorio:** También se trata de un algoritmo de toma de decisiones o de clasificación, en el cual la decisión de clasificación se toma por votación, a partir de los resultados de clasificación para unos datos de entrada, que arrojen un conjunto de árboles de decisión. Cada árbol toma aleatoriamente una fracción de las variables de entrada (ve una parte de los datos). En este algoritmo los hiperparámetros de interés son el criterio de

división, la profundidad de los árboles, la fracción de variables de entrada usadas en cada árbol y el número total de árboles. En la **Figura 4-2** se tiene un ejemplo esquemático de un bosque aleatorio.

**Figura 4-1:** Ejemplo de un árbol de decisión.



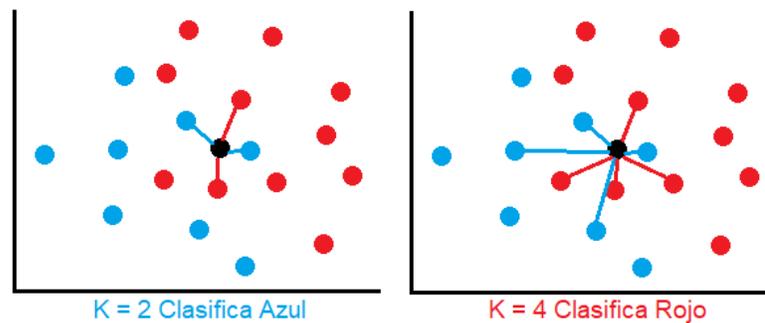
**Figura 4-2:** Ejemplo esquemático de un bosque aleatorio.



**K-vecinos más cercanos:** Es una técnica no paramétrica que asigna a un dato (en este caso funcional) su pertenencia a una categoría, basado en a la que tenga la menor distancia a  $K$  puntos ya clasificados (distancia euclidiana, manhattan, Minkowsky o

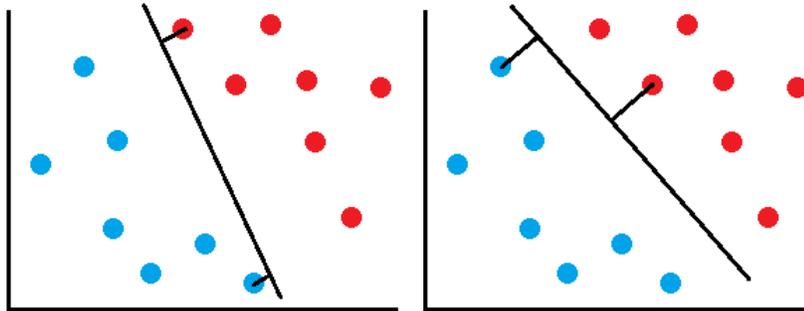
hamming). No tiene como tal un modelo construido subyacente y la clasificación se hace por votación. Por defecto en el algoritmo usado en este trabajo, se usó la distancia Minkowsky, que es una distancia euclidiana. Los hiperparámetros en este modelo son los pesos que se le dan a los  $K$ -vecinos (uniforme o dependiente de la distancia por ejemplo) y el número de vecinos más cercano. En la **Figura 4-3** se tiene un ejemplo de clasificación con  $K$  vecinos más cercanos.

**Figura 4-3:** Ejemplo de clasificación usando  $K$ -vecinos más cercanos.



Máquinas de soporte vectorial (MSV): Es un método de clasificación supervisada, en el cual se busca un hiperplano que separe de forma óptima, las nubes de puntos de las clases de entrenamiento, cosa tal que se optimice la clasificación cuando un nuevo punto deba ser adjudicado a un grupo. El hiperplano maximiza la distancia con los puntos que estén más cerca de él de los grupos que separa (ver **Figura 4-4**). En MSV se usan funciones núcleo que permiten transformar los datos a otros espacios donde se facilite la separación (ver **Tabla 4.2**). El núcleo lineal se aconseja cuando en el espacio de las variables se puede construir con facilidad un hiperplano de separación entre categorías, los demás núcleos de la tabla transforman los datos a otros espacios, cuando existe una alta no linealidad en los límites de separación, haciendo poco eficiente al núcleo lineal. En la **Figura 4-5** se esquematiza el efecto que tiene la escogencia del núcleo, en la forma de la superficie de separación entre categorías en el espacio de las variables.

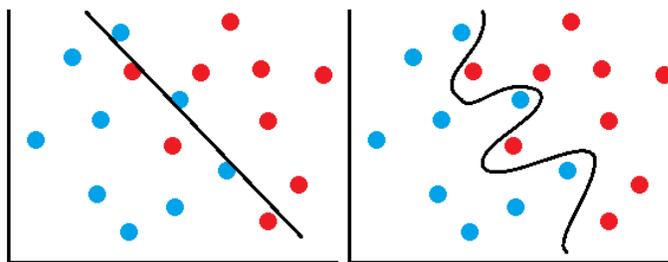
**Figura 4-4:** Ejemplo de hiperplano óptimo de separación en una MSV (esquema derecho).



**Tabla 4.2:** Núcleos comunes en las MSV.

	$K(x_1, x_2) = \langle x_1, x_2' \rangle$	$x_1, x_2$ : Observaciones
Lineal		
Polinomial	$K(x_1, x_2) = (\gamma \langle x_1, x_2' \rangle + r)^d$	$d$ : Grado del polinomio, $\gamma$ : Parámetro de influencia de cada individuo, $r$ : bias.
RBF	$K(x_1, x_2) = \exp(-\gamma \ x_1 - x_2\ ^2)$	$\gamma$ : Parámetro de influencia de cada individuo
Sigmoide	$K(x_1, x_2) = \tanh(\gamma \langle x_1, x_2' \rangle + r)$	$\gamma$ : Parámetro de influencia de cada individuo, $r$ : bias.

**Figura 4-5:** Ejemplo del efecto de la escogencia del núcleo en una MSV (izquierda núcleo lineal, derecha núcleo no lineal).



Regresión logística: Hace parte de los modelos lineales estadísticos, correspondiendo a un modelo de error binomial con función de enlace logit. Sin embargo, se suele clasificar también dentro de los algoritmos de aprendizaje de máquina, cuando por defecto la regresión está regularizada. La regularización se utiliza para evitar sobreajuste, agregando

un término de penalización a la función de pérdida, que el modelo intenta minimizar al buscar sus parámetros. La regularización usada en este trabajo fue la L2 o Ridge, que usa un término proporcional al cuadrado de los coeficientes. En la ecuación 4.21 se muestra el término de penalización usado, donde  $k$  corresponde a la categoría de clasificación. En este caso el hiperparámetro es la selección del tipo de regularización

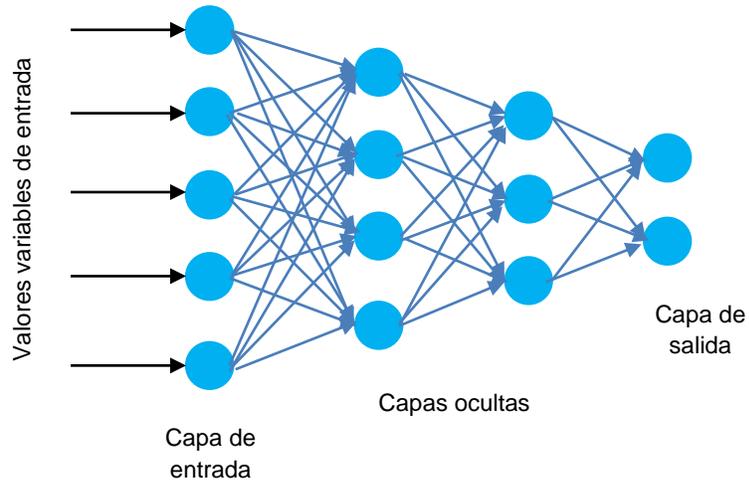
$$\frac{1}{2} \|W\|_F^2 = \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^K W_{i,j}^2 \quad (4.21)$$

Red neuronal tipo perceptrón: Estas redes fueron de las primeras a usarse en clasificación de imágenes. Constan de la capa de variables de entrada, la o las capas ocultas y la capa de salida (variables de respuesta o clasificación). En estas redes todas las neuronas de una capa están conectadas con todas las de la capa previa y subsiguiente (ver **Figura 4-6**). Estas redes tratan de emular el funcionamiento de las redes neuronales naturales, donde cada neurona o nodo tiene conexiones con muchas más de las que recibe señales y con otras a las que les entrega señales también. El nivel de respuesta de una neurona depende de si las señales de entrada sobrepasan un umbral. La relación entre las señales de entrada y las de salida ( $E, a$  y  $S$ ), están dadas a través de las funciones de transferencia ( $f$ ), donde los ( $b$ ) son los sesgos y las  $W$  son los pesos. (Ver **Figura 4-7:**

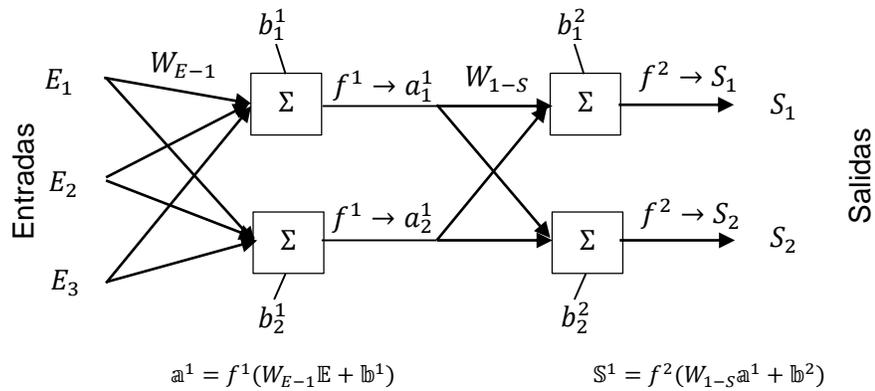
Ilustración). Los valores de las señales de salida, los sesgos y los pesos se calculan a través del entrenamiento de la red.

Los hiperparámetros más relevantes de una red neuronal son el tipo de función de activación o transferencia a usar (la función relu es muy común), el número de capas ocultas, el número de neuronas por cada capa, el número de iteraciones en el entrenamiento de la red y el método de resolución para encontrar los pesos (es este último caso un método común es el Adam).

**Figura 4-6:** Ilustración de una red neuronal tipo perceptrón.



**Figura 4-7:** Ilustración de una red neuronal de 3 capas (entrada, capa oculta y salida), donde se esquematiza la relación entre sus variables.



## 5. Metodología

Las imágenes de partida utilizadas fueron tomadas por el autor en color, en diferentes tiempos previos a la ejecución de este trabajo, con diferentes equipos de fotografía y con diferentes resoluciones, usando formato comprimido JPG.

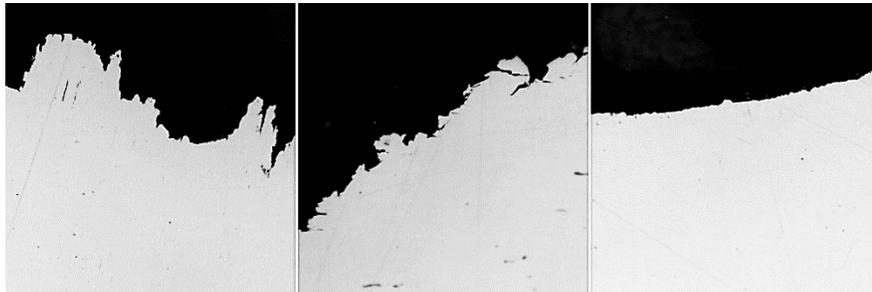
A continuación se describen los pasos metodológicos usados para preprocesar las imágenes y la aplicación de los algoritmos de clasificación:

- Selección entre el banco de imágenes que poseía el autor, de diez por cada una de las tres categorías, atendiendo a que mostraran claramente la textura característica de cada modo de fractura. Lo anterior debido a que hay fracturas mixtas, es decir, que presentan características combinadas entre varios tipos, y por ello no fueron tenidas en cuenta en la selección. Si bien se poseía un banco de imágenes cercano a 300, el lograr encontrar aquellas con la calidad suficiente y que tuvieran las texturas puras de cada uno de los tres modos de fractura, fue un proceso largo, del cual en el mejor de los casos se encontraron doce imágenes en el modo de falla más común (fatiga). Finalmente se lograron seleccionar diez imágenes por cada categoría.
- Las imágenes seleccionadas fueron recortadas para que quedaran todas con tamaños de 200 x 200 píxeles, asegurando así el poder comparar los datos funcionales (igual recorrido en píxeles).
- Como las imágenes fueron tomadas en color, al leerlas en el software R usando la librería “readbitmap”, a cada píxel le quedan asignados tres canales de información (RGB), de donde se escogió solo la información de intensidad del primer canal, ya que para los tres canales el valor de intensidad es igual. De esta manera cada imagen se convirtió en una matriz de 200 x 200, donde a cada píxel se le asignó un número entre 0 (negro o ausencia de luz en el píxel) y 1 (blanco o saturación de luz en el píxel), lo

que es últimas es la normalización de las cantidades originales medidas en bits de 0 a 256.

- El uso de treinta imágenes para entrenar y probar modelos de clasificación es en general muy bajo. Por lo anterior se decidió aplicar una metodología alternativa consistente en extraer de cada fotografía 400 datos funcionales, correspondientes a seguir las líneas de píxeles en X e Y de cada foto. Detrás de lo anterior está el supuesto de que cada dato funcional extraído de esta manera, tiene la información suficiente para poder clasificarse en la categoría de fractura correspondiente. Esta hipótesis se fundamenta en que cuando se hacen cortes a lo largo de superficies de fractura, y se observan los mismos en microscopio, se puede ver en el perfil una morfología característica de cada tipo de fractura, **Figura 5-1**. Con lo anterior las 30 imágenes se convierten en 12000 datos funcionales (4000 para cada categoría de fractura).

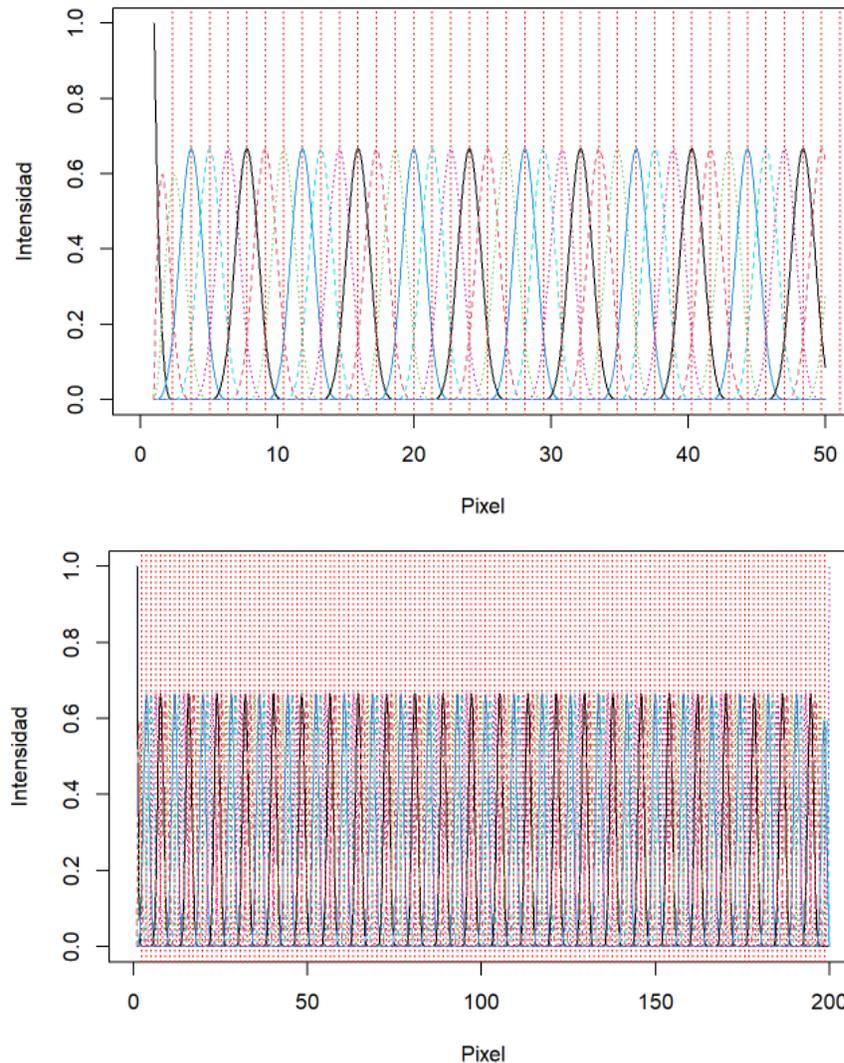
**Figura 5-1:** Secciones metalográficas que intersectan superficies de fractura. Izquierda dúctil, central frágil y derecha fatiga. 200X.



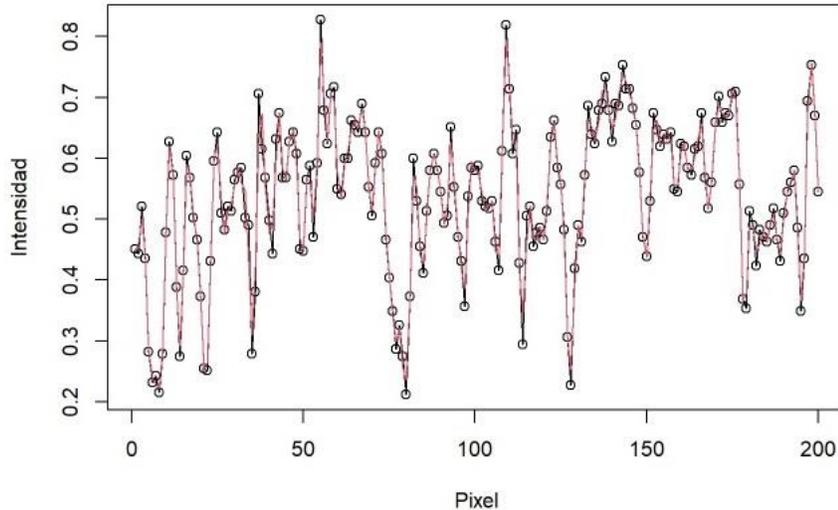
- Como se vio en el numeral 1 cada dato funcional muestra una función bastante irregular (no periódica), con muchos picos y valles agudos sucesivos. Para representar dichos datos se escogió una base B-Spline de 150 funciones de orden cúbica (en la librería fda de R, **Figura 5-2**), usando un parámetro de suavizamiento lambda, obtenido por VCG (variando lambda entre  $10^{-2}$  y  $10^4$ ). Se buscó un ajuste detallado de los datos (alta rugosidad o bajo suavizamiento, **Figura 5-3**), ya que en principio se consideró que el obtener los picos y valles en los datos suavizados, podía mejorar el desempeño en algoritmos de clasificación. El valor de Lambda cuando se suavizaron todas las curvas (entrenamiento y ensayo) por categorías fue de 0,02592944 para imágenes dúctiles y

frágiles, y 0,04175319 para imágenes de fatiga. El valor de Lambda cuando se suavizaron las curvas de entrenamiento como conjunto y las de ensayo como conjunto fue de 0,02592944, El valor de Lambda cuando se suavizaron las curvas de ACF de todas las curvas de entrenamiento como conjunto y de todas las de ensayo como conjunto fue de 0,1082637. El número de 150 funciones de la base de B-Spline, se obtuvo a partir del análisis del comportamiento del error cuadrático medio, el cual tiende a estabilizarse después de este número (ver **Tabla 5.1**).

**Figura 5-2:** Gráfica de las 150 funciones cúbicas de la base B-Spline usadas para representar los datos funcionales y las funciones de autocorrelación. Imagen superior con detalle hasta el pixel 50 e imagen inferior hasta el pixel 200.



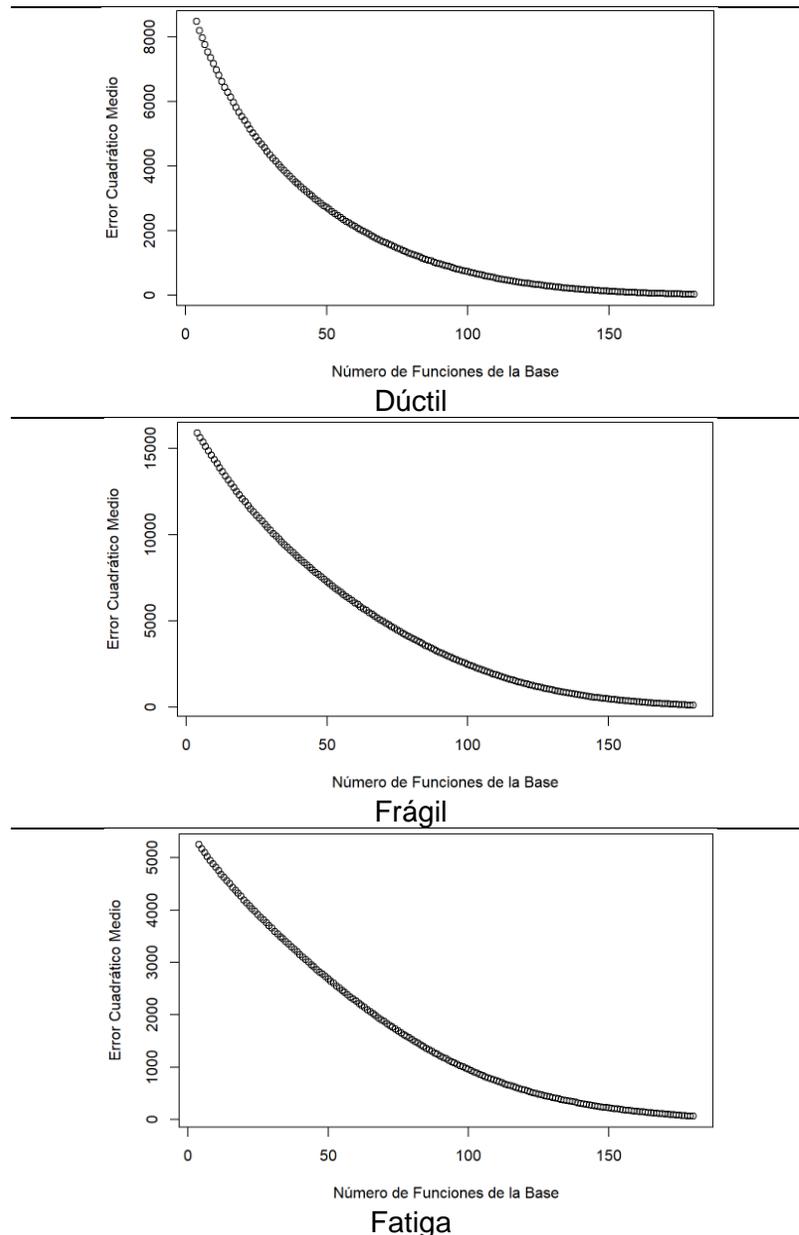
**Figura 5-3:** Ejemplo de los puntos originales de un dato funcional (puntos negros) y dato funcional representado con la base B-Spline (curva roja).



- Para entrenar los modelos de clasificación, se escogieron de manera aleatoria siete imágenes de cada grupo, dejándose tres para prueba; siguiendo la práctica común en técnicas de aprendizaje de máquina, de dejar el 70 % de los datos para entrenar modelos y el 30% para ensayo. En cada uno de los modelos de clasificación empleados, se usaron las mismas imágenes para entrenamiento y las mismas para ensayo.
- Los modelos usados para realizar la clasificación de los datos funcionales fueron los siguientes:
  - MLG (modelo lineal generalizado usando distribuciones de error/funciones de enlace: Binomial/logit, Gaussiana/identidad, Poisson/log, Quasi/identidad, Quasi-binomial/logit y Quasi-poisson/log), de las curvas originales sin centrado. Se usó la función “`classif.glm`” de la librería “`fda.usc`” del software R.
  - MLG (con Binomial/logit), de las curvas con primera y segunda derivada.
  - MLG (con Binomial/logit, Gaussiana/identidad, Poisson/log, Quasi/identidad, Quasi-binomial/logit y Quasi-poisson/log), de las curvas centradas respecto a las funciones medias de entrenamiento y ensayo.
  - MLG (con Binomial/logit), de las curvas pasadas a su representación en componentes principales funcionales, usando uno, dos y 150 armónicos.

- MAG (modelo aditivo generalizado usando Binomial/logit, Gaussiana/identidad, Poisson/log, Quasi/identidad, Quasi-binomial/logit y Quasi-poisson/log), de las curvas originales sin centrado. Se usó la función “`classif.gsam`” de la librería “`fda.usc`” del software R.
- MAG (con Binomial/logit), de las curvas con primera y segunda derivada.

**Tabla 5.1:** Comportamiento del error cuadrático medio vs el número de funciones de la base B-Spline.



- MAG (con Binomial/logit, Gaussiana/identidad, Poisson/log, Quasi/identidad, Quasi-binomial/logit y Quasi-poisson/log), de las curvas centradas respecto a las funciones medias de entrenamiento y ensayo.
- MAG (con Binomial/logit), de las curvas pasadas a su representación en componentes principales funcionales, usando uno, dos y 150 armónicos.
- Distancia de Mahalanobis entre los “Puntajes” del primer eje principal, a la media de cada grupo de entrenamiento. Se hizo ACPF (análisis de componentes principales funcionales) a todo el conjunto de datos. Se usó la función “funDist” de la librería “gmfd” del software R.
- Distancia de Mahalanobis entre los “Puntajes” de los dos primeros ejes principales funcionales, a la media de cada grupo de entrenamiento. Se hizo ACPF a todo el conjunto de datos.
- Distancia de Mahalanobis entre los “Puntajes” de los tres primeros ejes principales, a la media de cada grupo de entrenamiento. Se hizo ACPF a todo el conjunto de datos.
- Distancia de Mahalanobis entre los “Puntajes” de los 150 primeros armónicos, a la media de cada grupo de entrenamiento. Se hizo ACPF a todo el conjunto de datos.
- Distancia L2 entre los “Puntajes” del primer eje principal funcional, a la media de cada grupo de entrenamiento. Se hizo ACPF a todo el conjunto de datos. Se usó la función “funDist” de la librería “gmfd” del software R.
- Distancia L2 entre los “Puntajes” de los dos primeros ejes principales funcionales, a la media de cada grupo de entrenamiento. Se hizo ACPF a todo el conjunto de datos.
- Distancia L2 entre los “Puntajes” de los tres primeros ejes principales funcionales, a la media de cada grupo de entrenamiento. Se hizo ACPF a todo el conjunto de datos.
- Distancia L2 entre los “Puntajes” de los 150 primeros armónicos funcionales, a la media de cada grupo de entrenamiento. Se hizo ACPF a todo el conjunto de datos.

- Distancia euclidiana entre los “Puntajes” del primer eje principal funcional, a la media de cada grupo de entrenamiento. Se hizo ACPF a todo el conjunto de datos. El algoritmo de clasificación fue programado por el autor.
- Distancia euclidiana entre los “Puntajes” del primer eje principal funcional, a la media de cada grupo de entrenamiento. Se hizo ACPF a todo el conjunto de datos. El algoritmo de clasificación fue programado por el autor.
- Distancia euclidiana entre los “Puntajes” de los 150 primeros armónicos funcionales, a la media de cada grupo de entrenamiento. Se hizo ACPF a todo el conjunto de datos. El algoritmo de clasificación fue programado por el autor.
- Distancia Procrustes entre la matriz de covarianza de cada foto de ensayo, respecto a la matriz de covarianza de cada categoría de entrenamiento. Se usó la función “distcov” de la librería “shapes” del software R.
- Clasificador basado en medida de profundidad, usando profundidades FM (Profundidad de Fraiman and Muniz), h-Modal (Profundidad h-Modal), RT (Profundidad de Tukey Aleatoria), RP (Profundidad Proyectada Aleatoria) y RPD (Profundidad Proyectada Doblemente Aleatoria). Se usó la función “classif.depth” de la librería “fda.usc” del software R.
- Clasificador basado en la medida de profundidad extrema. Para calcular esta profundidad se usó la función “extremal\_depth” de la librería “fdaoutlier” del software R y el algoritmo de clasificación fue programado por el autor.
- Clasificador basado en un árbol de decisión, usando como hiperparámetros: criterio de división Gini y profundidad de 25. La búsqueda de hiperparámetros se hizo con validación cruzada de tres pliegues. Se usó la función “DecisionTreeClassifier” de la librería “sklearn” del software Phyton.
- Clasificador basado en un bosque aleatorio, usando como hiperparámetros: criterio de división Gini, fracción de características usadas para cada árbol 0,04 y número de árboles 200. La búsqueda de hiperparámetros se hizo con validación cruzada de tres pliegues. Se usó la función “RandomForestClassifier” de la librería “sklearn” del software Phyton.
- Clasificador basado en una máquina de soporte vectorial MSV con función núcleo de funciones radiales base RBF, usando como hiperparámetros: parámetro de regularización 0,0625 y coeficiente de la función núcleo 0,5. La búsqueda de hiperparámetros se hizo con validación cruzada de tres pliegues. Se usó la función “SVC” de la librería “sklearn” del software Phyton.

- Clasificador basado en una máquina de soporte vectorial MSV con función núcleo polinomial de orden tres, usando como hiperparámetros: parámetro de regularización 0,0625 y coeficiente de la función núcleo 0,5. La búsqueda de hiperparámetros se hizo con validación cruzada de tres pliegues. Se usó la función “SVC” de la librería “sklearn” del software Phyton.
- Clasificador basado en una máquina de soporte vectorial MSV con función núcleo sigmoide, usando como hiperparámetros: parámetro de regularización  $9,5367431640625e-07$  y coeficiente de la función núcleo 0,5. La búsqueda de hiperparámetros se hizo con validación cruzada de tres pliegues. Se usó la función “SVC” de la librería “sklearn” del software Phyton.
- Clasificador basado en una máquina de soporte vectorial MSV con función núcleo lineal, usando como hiperparámetro el parámetro de regularización 0,015625. La búsqueda del hiperparámetro se hizo con validación cruzada de tres pliegues. Se usó la función “SVC” de la librería “sklearn” del software Phyton.
- Regresión logística. Se usó la función “LogisticRegression” de la librería “sklearn” del software Phyton, usando regularización Ridge.
- K-vecinos más cercanos, usando como hiperparámetro  $K = 2$  y pesos uniformes. La búsqueda del hiperparámetro se hizo con validación cruzada de tres pliegues. Se usó la función “KNeighborsClassifier” de la librería “sklearn” del software Phyton.
- Red neuronal del tipo “perceptrón” de una capa oculta, usando como hiperparámetros: función de activación tipo “relu”, 300 neuronas en la capa oculta, 250 iteraciones y como método de resolución de las funciones de peso “Adam”. La búsqueda de hiperparámetros se hizo con validación cruzada de tres pliegues. Se usó la función “MLPClassifier” de la librería “sklearn” del software Phyton.
- Red neuronal del tipo “perceptrón” de doble capa oculta, usando como hiperparámetros: función de activación tipo “relu”, 300 neuronas en la primera capa oculta, 12 neuronas en la segunda capa oculta, 200 iteraciones y como método de resolución de las funciones de peso “Adam”. La búsqueda de hiperparámetros se hizo con validación cruzada de tres pliegues. Se usó la función “MLPClassifier” de la librería “sklearn” del software Phyton.

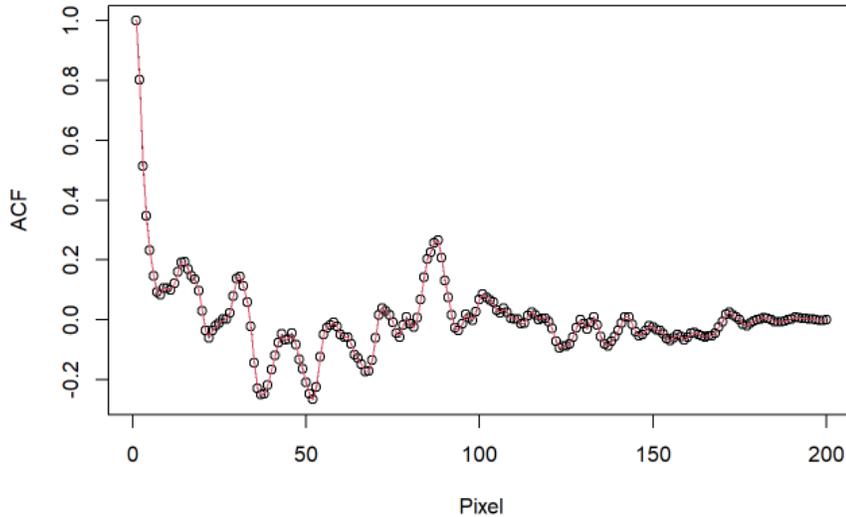
- La función de autocorrelación espacial intra de cada dato funcional, se incorporó en los siguientes clasificadores adicionales:
  - MLG (modelo lineal generalizado usando funciones de enlace Binomial, Gaussiana, Poisson, Quasi, Quasi-binomial y Quasi-poisson), de las curvas originales sin centrado. Los datos funcionales de las funciones de autocorrelación (ACF) se incluyeron como un cofactor en el modelo. Para ello también se representaron en la base B-Spline de 150 funciones base de orden cúbica, usando un parámetro de suavizamiento lambda, obtenido por validación cruzada generalizada (variando lambda entre  $10^{-2}$  y  $10^4$ ). **Figura 5-2** y **Figura 5-4**. En la ecuación 5.1 se especifica el modelo usado, donde las  $x^1$  son los datos funcionales, las  $x^2$  son los datos funcionales de las ACF y los  $\beta$  son las funciones parámetro del modelo.
  - MAG (modelo aditivo generalizado usando funciones de enlace Binomial, Gaussiana, Poisson, Quasi, Quasi-binomial y Quasi-poisson), de las curvas originales sin centrado. Los datos funcionales de las funciones de autocorrelación (AC) se incluyeron como un cofactor en el modelo. En la ecuación 5.2 se especifica el modelo usado, donde las  $x^1$  son los datos funcionales, las  $x^2$  son los datos funcionales de las ACF y las  $f$  son las funciones de suavizamiento del modelo.

$$y_i = g^{-1} \left( \int_{T_1} x_i^1(t) \beta_1(t) dt + \int_{T_2} x_i^2(t) \beta_2(t) dt \right) + \epsilon_i \quad (5.1)$$

$$y_i = g^{-1} \left( \int_{T_1} f^1(x_i^1(t)) dt + \int_{T_2} f^2(x_i^2(t)) dt \right) + \epsilon_i \quad (5.2)$$

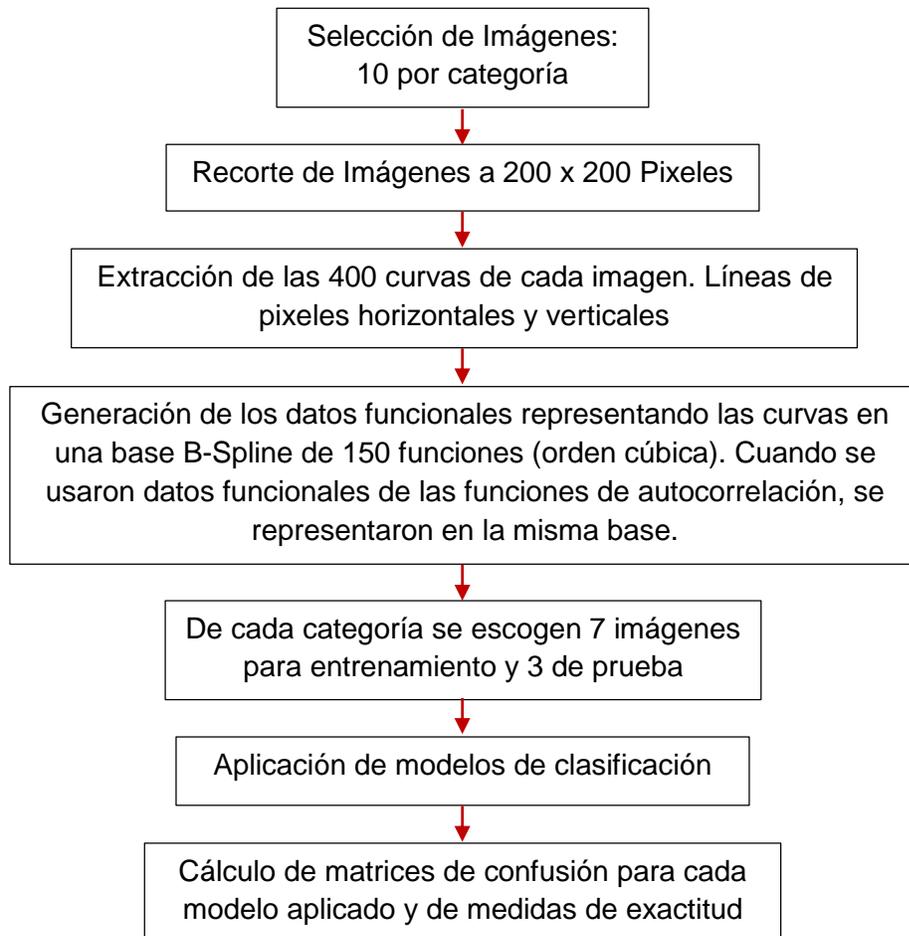
- Clasificador basado en la medida de profundidad extrema, aplicado a los datos funcionales de las funciones de autocorrelación (ACF).
- Clasificador basado en la medida de profundidad extrema, promediando el orden obtenido en los datos funcionales y el obtenido en los datos funcionales de las funciones de autocorrelación (ACF).

**Figura 5-4:** Ejemplo de gráfica de los puntos originales de un dato funcional de una función de autocorrelación (puntos negros) y dato funcional representado con la base B-Spline (curva roja).

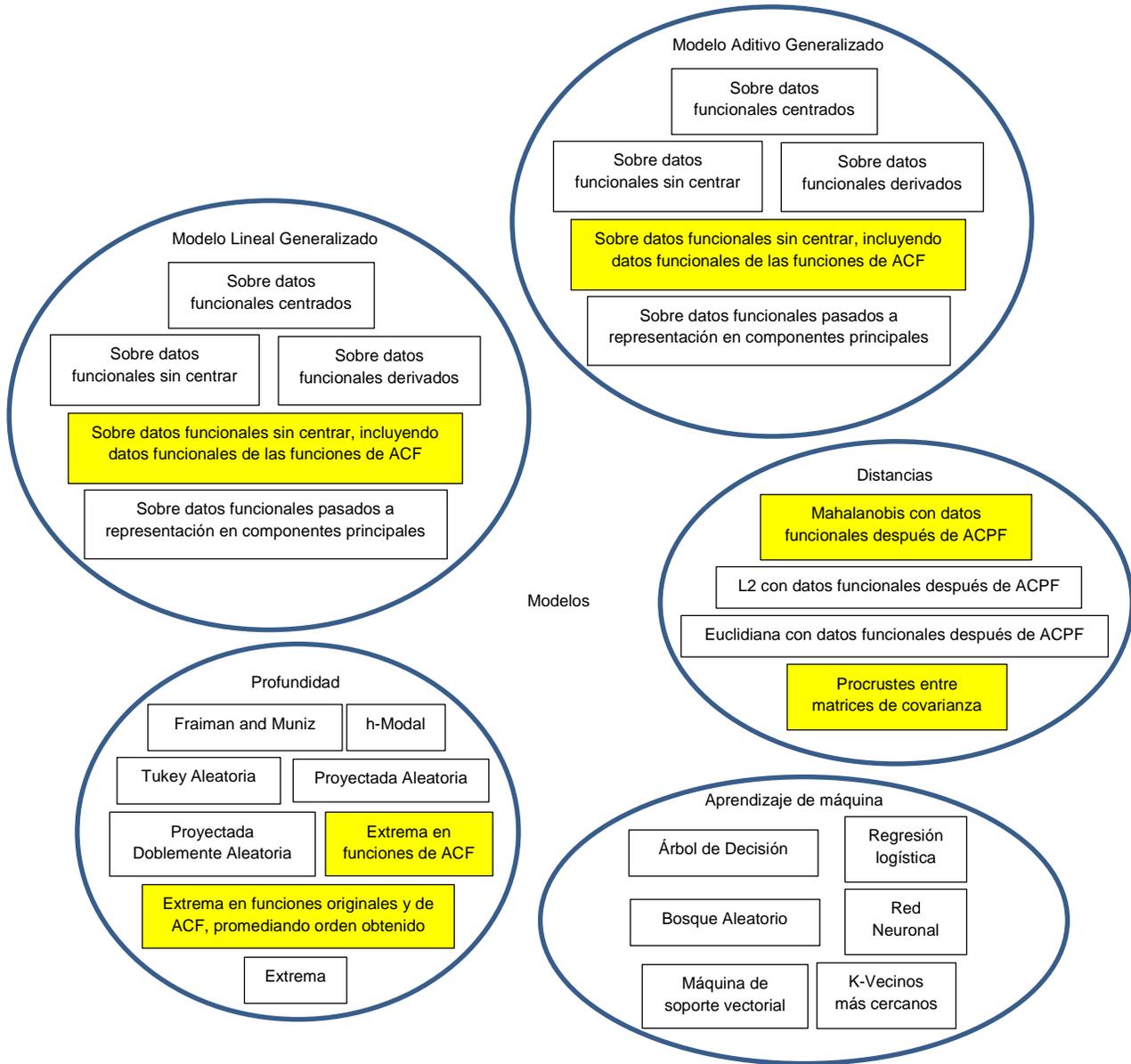


- Con la clasificación realizada por cada modelo, se procedió a calcular las matrices de confusión respectivas y a partir de ellas las métricas de exactitud, esto para poder comparar los modelos y detectar los de mejor desempeño. Para cada clasificador se obtuvieron dos matrices de confusión: “por curvas” y “por fotos”, la primera se refiere a la clasificación realizada contando cada uno de los datos funcionales y la segunda a la clasificación realizada por fotos. En el segundo caso una foto se clasifica en una categoría dada, si en la clasificación por cada una de sus curvas (datos funcionales), dicha categoría obtuvo la mayoría, es decir por votación.

En la **Figura 5-5** se resumen los pasos generales de la metodología empleada y en la **Figura 5-6** están los modelos de clasificación usados.

**Figura 5-5:** Diagrama de flujo de la metodología usada.

**Figura 5-6:** Resumen de los modelos de clasificación usados. En amarillo se resaltan aquellos que usaron la función de autocorrelación intra dato funcional o las matrices de covarianza.



## 6. Resultados

De acuerdo a lo descrito en el capítulo de metodología, se aplicaron a las imágenes de prueba los algoritmos de clasificación, entrenados con los datos extraídos de las imágenes de entrenamiento, usando códigos en lenguajes R y Python.

Una vez realizadas las clasificaciones se calcularon para cada modelo, dos matrices de confusión, donde la primera corresponde a la clasificación sobre los datos funcionales y la segunda al resultado de la votación para la clasificación por fotografía. En total se tuvieron 3600 datos funcionales de prueba (1200 por cada categoría), que corresponden a 9 imágenes de prueba (3 por categoría). En la **Tabla 6.1** se muestran las matrices de confusión obtenidas en el caso del clasificador por regresión logística, donde la matriz “por curvas” hace referencia a la de los datos funcionales y la de “por fotos” a la de votación por fotografías. El total de las matrices de confusión se puede consultar, en el [Anexo A](#).

**Tabla 6.1:** Ejemplo de matrices de confusión obtenidas para el clasificador por regresión logística.

Por curvas					Por fotos				
Regresión Logística									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	842	73	68	Predicción	Dúctil	3	0	0
	Frágil	68	697	659		Frágil	0	2	2
	Fatiga	290	430	473		Fatiga	0	1	1

A partir de las matrices de confusión se calculó la fracción de exactitud redondeada a tres cifras decimales, en la clasificación de cada modelo, tanto “por curvas” como “por fotos” y discriminando en (a) exactitud global de la clasificación, (b) por casos de fractura dúctil, (c) de fractura frágil y (d) de fractura por fatiga. En la **Tabla 6.2** se encuentra como ejemplo

las medidas de exactitud obtenidas, para el modelo de clasificación por regresión logística, donde estos datos se calculan con los valores de la **Tabla 6.1**. En el [Anexo B](#) se encuentran todas las medidas de exactitud obtenidas de los clasificadores. En la **Tabla 6.3** se comparan los resultados de exactitud obtenidos “por fotos”, con el fin de poder notar mejor el desempeño relativo obtenido entre los clasificadores.

**Tabla 6.2:** Ejemplo de medidas de exactitud obtenidas del modelo de clasificación, basado en regresión logística de los datos funcionales no centrados.

Por curvas				Por fotos			
Global	Dúctil	Frágil	Fatiga	Global	Dúctil	Frágil	Fatiga
Regresión Logística							
0,559	0,702	0,581	0,394	0,667	1,000	0,667	0,333

En los Anexos [A](#) y [B](#) y en la **Tabla 6.3** se encuentra el resumen de los resultados obtenidos en este proyecto, donde se resaltan los siguientes hechos generales:

- La exactitud de cada clasificador en general aumenta evaluando “por fotos” respecto a cuando se evalúa “por curvas” o datos funcionales. Indicando que la clasificación por votación mejora el desempeño (ver [Anexo B](#)).
- En los grupos de clasificadores donde se incluyó información de las curvas de autocorrelación (ACF) intra dato funcional o de la función de covarianza (Distancias Procrustes o Mahalanobis) entre datos funcionales, algunos de dichos algoritmos lograron la mejor exactitud del grupo. Ver los cuadros amarillos de la **Tabla 6.3** para los clasificadores basados en distancias, modelo lineal generalizado (MLG), modelo aditivo generalizado (MAG) y en profundidad.
- Dentro de los modelos de clasificación estadísticos evaluados, el basado en MLG logra el máximo de exactitud “por fotos”, cuando se incluye la función de autocorrelación intra (ACF) de los datos funcionales como cofactor, con familia binomial, gaussiana, quasi y quasi-binomial (**Tabla 6.3**). En este caso la exactitud por votación llegó al 100%, equiparándose con los mejores modelos de aprendizaje de máquina, como la MSV con núcleos RBF y polinomial, el árbol de clasificación, el bosque aleatorio y la red neuronal de doble capa oculta tipo perceptrón.



- Los modelos de clasificación que lograron las mayores exactitudes “por fotos”, son aquellos en los cuales la exactitud en la clasificación “por curvas” o datos funcionales en todas las categorías fue simultáneamente alta. Hubo clasificadores como se detalla más adelante, que se comportaron bien en una o dos categorías de fractura y en la otras no, y son precisamente estos los de más bajo desempeño. Es decir, los clasificadores de más bajo desempeño en general lo son por no clasificar eficientemente una o dos de las tres categorías de fractura.

Respecto a los clasificadores basados en medidas de distancia se encontró lo siguiente:

- Ningún clasificador logró en la votación exactitud del 100%.
- Los clasificadores de mejor desempeño fueron Procrustes, Mahalanobis usando los primeros uno o dos armónicos y distancia euclidiana en las medidas de “puntajes” después del ACPF.
- El uso en la distancia de Mahalanobis con más de dos armónicos o componentes principales, baja notablemente la exactitud de la clasificación.
- La clasificación usando la distancia de Mahalanobis lleva a mayor exactitud, respecto al uso de la distancia L2. Esto similar a lo encontrado en [\[Xinyu et al., 2023\]](#).
- En general en los métodos de clasificación usando distancias, el orden descendente de exactitud por categorías es frágil, dúctil y fatiga. La categoría fatiga tiene un desempeño en estos clasificadores deficiente.
- En resumen, métodos de clasificación por distancias que tengan en cuenta la información de covarianza entre los datos funcionales (Mahalanobis o Procrustes), se desempeñan mejor que los que se basan en distancia L2 usando base B-Splines. Si se usa base de componentes principales funcionales, la clasificación basada en distancia euclidiana entre “Puntajes”, es comparable en exactitud con las de Mahalanobis o Procrustes con base B-Splines.

Referente a los clasificadores basados en MLG o MAG los resultados mostraron lo siguiente:

- Clasificar usando los datos funcionales no centrados da mayor exactitud, cuando se usan las familias binomial y quasi-binomial, comparado con las demás familias.
- En general en los métodos de clasificación usando los datos funcionales no centrados, el orden descendente de exactitud por categorías es dúctil, frágil y fatiga. La categoría fatiga tiene un desempeño en estos clasificadores deficiente.
- No hay un cambio notable en los resultados de clasificación “por fotos”, entre usar los datos funcionales originales o centrados respecto a la media.
- En estos clasificadores el uso de datos funcionales en base B-Splines o de componentes principales funcionales (con uno, dos o 150 armónicos), no tiene incidencia notable en los valores de exactitud obtenidos en la clasificación “por fotos”.
- Usar clasificadores de los datos funcionales con primera o segunda derivada, desmejora la exactitud obtenida en la clasificación respecto al uso de los datos funcionales originales.
- Usar clasificadores donde se use como cofactor los datos funcionales de las funciones de autocorrelación intra, mejoran la exactitud de la clasificación en algunos casos, fundamentalmente cuando mejoran el desempeño en la categoría fatiga. El uso como cofactores de los datos funcionales de las funciones de autocorrelación, aumentan más la exactitud en los MLG que en los MAG. Como ya se mencionó en algunos casos se logra exactitud del 100% para MLG en la clasificación “por fotos”.

Respecto a los clasificadores basados en medidas de profundidad se halló lo siguiente:

- En general el uso de medidas de profundidad para la clasificación no logra una exactitud satisfactoria.
- Dentro de las medidas de profundidad usadas para la clasificación, la de mejor desempeño fue la de profundidad extrema.
- La clasificación basada en la profundidad extrema de las curvas de autocorrelación intra (ACF), tiene una exactitud “por fotos” similar a la alcanzada con los datos funcionales originales.
- Clasificar a partir de un promedio en el orden obtenido con medida de profundidad extrema de datos funcionales originales y de las funciones autocorrelación intra (ACF), logra el mejor desempeño dentro del grupo de clasificadores basados en medidas de profundidad.

- En general en los métodos de clasificación usando medidas de profundidad, el orden descendente de exactitud por categorías es frágil, dúctil y fatiga. La categoría fatiga tiene un desempeño en estos clasificadores deficiente.

Referente a los clasificadores basados en aprendizaje de máquina los resultados mostraron lo siguiente:

- La exactitud de la clasificación con K-vecinos más cercanos es baja, al igual que con la MSV con núcleo sigmoide (en este último clasificador se obtuvo la exactitud más baja dentro del grupo de aprendizaje de máquina).
- En la clasificación con K-vecinos más cercanos y con MSV con núcleo sigmoide, el orden descendente de exactitud por categorías es fatiga, dúctil y frágil. Las categorías frágil y dúctil tienen un desempeño deficiente en estos clasificadores.
- La exactitud de 100% en la clasificación “por fotos”, se logra con el clasificador basado en MSV con núcleos RBF y polinomial, con el árbol de decisión, con el bosque aleatorio y con la red neuronal tipo perceptrón de doble capa oculta. Lo anterior concuerda con [\[Yinfeng et al., 2016\]](#).
- La red neuronal tipo perceptrón de una sola capa oculta es de menor desempeño que la de doble capa, sin embargo su exactitud de clasificación sigue siendo alta (77,8%). En este caso la categoría con baja exactitud de clasificación es la de fractura dúctil.
- La MSV con núcleo lineal y la regresión logística, obtuvieron exactitudes de clasificación “por fotos” intermedias (66,7%). El orden descendente de exactitud por categorías es dúctil, frágil y fatiga. La categoría fatiga tiene un desempeño deficiente en estos clasificadores.

## 6.1 Análisis

En general se encontró en la mayoría de clasificadores, que la categoría fractura por fatiga es la que mostró menor exactitud de clasificación durante las pruebas. Los datos funcionales de la categoría fatiga tienen una función media que es intermedia entre las otras dos categorías y además existe un alto grado de traslape entre los datos funcionales de las tres categorías. Lo anterior hace que en clasificadores donde solo se use la

información de los datos funcionales, se presente lo siguiente: (a) al existir una mayor separación entre las funciones media dúctil y frágil es menos probable que los clasificadores se equivoquen entre estas dos categorías, pero (b) la separación entre las funciones media dúctil-fatiga y frágil-fatiga es baja, lo que implica mayor probabilidad de clasificación errónea entre estas categorías. Esto tiende a acentuarse en los clasificadores basados en distancias, MLG, MAG y profundidad.

Debido al alto grado de traslape entre los datos funcionales de las tres categorías, para una misma foto de prueba se presentan importantes porcentajes de inexactitud en la clasificación “por curvas”, sin embargo en la clasificación “por fotos” la votación realizada aumenta la exactitud. Esto se debe a que se tienen 400 datos funcionales por foto, cuya clasificación debe marcar una tendencia hacia una categoría, como lo sugirió la prueba de igualdad de funciones media, cuya hipótesis nula fue rechazada. Es decir los cuatrocientos datos están generados por el mismo proceso aleatorio, que tiene una media o categoría marcada.

Los clasificadores que usaron distancias Procrustes y de Mahalanobis (con uno y dos armónicos), obtuvieron las mayores exactitudes dentro del grupo de clasificadores por distancias. Esto indica que los operadores de covarianza o matrices de covarianza entre las categorías, tienen información suficiente que permiten discernir la pertenencia de una foto a una categoría dada; como lo sugirió la prueba de igualdad de operadores de covarianza, cuya hipótesis nula fue rechazada. El hecho de no necesitarse sino a lo sumo dos armónicos para lograr una buena clasificación con distancia de Mahalanobis, se debe a que como se vio en la estadística descriptiva, es sobre el primer eje principal funcional, en que se logra mayor separación entre los “puntajes” de las diferentes categorías.

La clasificación basada en medidas de profundidad en este caso tuvo en general bajo desempeño, lo que se debió a que las funciones medianas de las diferentes categorías son próximas en nivel y forma, lo que dificulta el logro de elevados niveles de exactitud en las clasificaciones. Es decir, un dato funcional logra simultáneamente altas profundidades en dos o tres categorías, bajando la capacidad de discriminación.

Los clasificadores MLG, MAG y basados en profundidad mejoraron en algunos casos su desempeño, al agregar datos funcionales de las funciones de autocorrelación intra al

modelo, a consecuencia de que dichas funciones tienen información que permite distinguir las categorías. Esto en lo referente a (a) nivel de autocorrelación (mayor en dúctil, seguido de frágil y fatiga), (b) rezagos hasta los cuales son significativas las autocorrelaciones (cuatro en dúctil y uno en frágil y fatiga) y (c) la presencia de estacionalidad espacial en dúctil. En [\[Radhesyam et al., 2020\]](#) también se encontró, que la adición de información de correlación espacial mejora la exactitud de clasificación

En los clasificadores MSV con núcleos RBF y polinomial, árbol de decisión, bosque aleatorio y red neuronal de doble capa, no fue necesario incluir información de autocorrelación intra o de funciones de covarianza, ya que con sus hiperparámetros optimizados, lograron clasificación perfecta “por fotos”. Es decir, con la información contenida en los datos funcionales fue suficiente para que dichos algoritmos, extrajeran las características necesarias y lograran una correcta clasificación. Esto concuerda con lo expresado en [\[Ashraf et al., 2012\]](#), [\[Sheykhmousa et al., 2020\]](#) o [\[Sabat et al., 2020\]](#).

Los clasificadores de MSV que mejor se comportaron fueron los de núcleo RBF y polinomial (exactitud del 100%), lo que indica que el lograr un hiperplano eficiente de separación entre las categorías de fractura, implica modelar relaciones complejas no lineales, que es donde estos núcleos funcionan mejor. El núcleo sigmoide tuvo el menor desempeño, ya que este en general se recomienda cuando la MSV modele redes neuronales, donde este no fue el caso. La MSV con núcleo lineal y la regresión logística, lograron exactitudes intermedias (66%), donde la causa de las inexactitudes está en que las fronteras entre las categorías son no lineales.

El clasificador de K-vecinos más cercanos puede modelar fronteras entre categorías no lineales, sin embargo en este caso se tiene como ya se mencionó, alto grado de traslape entre los datos funcionales de las diferentes categorías, lo que lleva a errores de clasificación que en este caso hicieron del clasificador uno de los de más bajo desempeño en el grupo de aprendizaje de máquina.

El clasificador de MLG con familia binomial y quasi-binomial, obtuvo mayor exactitud respecto a la familia Gaussiana, quasi, Poisson y quasi-Poisson, lo cual se debe a que la familia binomial es más adecuada para problemas de clasificación del tipo 0-1, que para

este caso significa pertenencia a una categoría de fractura o no. En cambio la familia Poisson se adecua mejor a modelos donde la variable de respuesta sean conteos y la gaussiana a variables continuas. Por otro lado la familia quasi binomial permite modelar posibles sub o sobredispersiones de varianza del modelo; sin embargo el hecho de encontrarse los mismos niveles de exactitud “por fotos” respecto a la familia binomial, sugiere que en este caso fueron pequeñas y no afectaron la clasificación.

La inclusión de los datos funcionales de las funciones de autocorrelación intra en el MLG, hizo que en el caso de la familia binomial y quasi-binomial, se lograra exactitud del 100% “por fotos”, e incluso familias como la Gaussiana, quasi, quasi-Poisson y Poisson aumentaron las exactitudes de clasificación. Es decir, la inclusión de la información de autocorrelación intra, compenso el uso de familias no adecuadas para los datos como la Gaussiana o Poisson.

Los modelos MLG y MAG tuvieron desempeños similares, como en [\[Oviedo et al., 2011\]](#), lo cual se puede deber a que en el modelo MAG al aplicar las funciones de suavizamiento, el resultado sea una dependencia cercana a la lineal. Sin embargo, agregando las funciones de autocorrelación intra a los modelos MAG, no se logró aumentar hasta el 100% de exactitud, como sí ocurrió en algunos modelos MLG.

La extracción de los datos funcionales a lo largo de las líneas de píxeles X e Y de las imágenes, tiene la ventaja de permitir aumentar los datos de entrenamiento. En [\[Waqar et al., 2019\]](#) también aumentaron los datos, pero con estrategias como distorsionar, rotar o reflejar las imágenes. El aumentar los datos de entrenamiento puede hacer que aumente la exactitud de clasificación, como se sugiere en [\[Delaigle et al., 2012\]](#). La estrategia de tomar datos funcionales a lo largo de las líneas de píxeles X e Y, en últimas ayuda a bajar la complejidad de los modelos de clasificación y su costo computacional, concordando con trabajos como [\[Chen et al., 2021\]](#) y [\[Radhesyam et al., 2020\]](#).



## 7. Conclusiones y recomendaciones

### 7.1 Conclusiones

La clasificación de imágenes de texturas, a partir de la extracción de datos funcionales a lo largo de las líneas de píxeles X e Y, permite entrenar modelos cuando se tenga limitaciones en número de imágenes, sin que ello lleve a la obtención de exactitudes de clasificación bajas.

Los métodos de clasificación basados en profundidades, en general muestran menor desempeño comparados con MLG, MAG y distancias.

Adicionarles a los clasificadores basados en distancias, MLG, MAG o en profundidades, información de las funciones de autocorrelación intra o de los operadores o matrices de covarianza entre los datos funcionales, en general aumenta la exactitud de la clasificación.

Un método de clasificación eminentemente estadístico como el basado en MLG, puede lograr desempeños en clasificación tan altos como los de los algoritmos de aprendizaje de máquina (redes neuronales, árboles de decisión, bosque aleatorio y MSV), si a los datos funcionales se les agrega en los modelos cofactores que tengan información de la autocorrelación de los datos. En este caso se usó la autocorrelación intra dato funcional (serie espacial). Con la ventaja de la interpretabilidad de parámetros.

La exactitud de la clasificación “por fotos” en general es mayor que en la clasificación “por curvas” o datos funcionales.

Modelos de alta exactitud global de clasificación, son aquellos que simultáneamente tienen alta exactitud de clasificación por cada categoría.

La categoría de fractura por fatiga, fue la que en general tuvo más baja exactitud de clasificación, ya que presenta alto grado de traslape de sus datos funcionales con los de las otras dos categorías.

Los modelos de clasificación que mejor desempeño tuvieron fueron (a) MLG (Binomial/logit, Gaussiana/identidad, Quasi/identidad y Quasi-binomial/logit, usando como datos funcionales las curvas de luminosidad por pixeles y las curvas de autocorrelación intra), (b) MSV (núcleos RBF y polinomial), (c) árbol de decisión, (d) bosque aleatorio y (e) red neuronal de doble capa oculta.

Los algoritmos de aprendizaje de máquina después de optimizar sus hiperparámetros, logran exactitudes de clasificación elevados, sin requerir la inclusión de la información de la autocorrelación espacial explícita.

## 7.2 Recomendaciones

Se recomienda explorar otras formas de inclusión de la información de autocorrelación espacial a los modelos, por ejemplo a través de autocorrelación entre curvas, que puedan mejorar su desempeño.

Para construir sistemas expertos que clasifiquen imágenes de superficies de fractura, se recomienda usar modelos de clasificación basados en MLG que involucre información de las funciones de autocorrelación intra y modelos de aprendizaje de máquina como la MSV, el árbol de decisión, el bosque aleatorio y redes neuronales tipo perceptrón de doble capa, cuyos hiperparámetros se optimicen por validación cruzada.

## **Anexo A: Matrices de confusión**

A continuación se encuentran las matrices de confusión obtenidas para cada uno de los clasificadores, tanto “por curvas” como “por fotos”.

Tabla A 1. Matrices de confusión para los clasificadores basados en distancias.

Por curvas					Por fotos				
Distancia Procruster									
							Actual		
							Dúctil	Frágil	Fatiga
		N/A			Predicción	Dúctil	1	0	0
						Frágil	0	3	0
						Fatiga	2	0	3
Distancia de Mahalanobis usando un armónico de curvas de ensayo a cada media de grupo de entrenamiento									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	926	71	83	Predicción	Dúctil	3	0	0
	Frágil	19	786	678		Frágil	0	3	2
	Fatiga	255	343	439		Fatiga	0	0	1
Distancia de Mahalanobis usando dos armónicos de curvas de ensayo a cada media de grupo de entrenamiento									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	984	123	133	Predicción	Dúctil	3	0	0
	Frágil	110	862	828		Frágil	0	3	2
	Fatiga	106	215	239		Fatiga	0	0	1
Distancia de Mahalanobis usando tres armónicos de curvas de ensayo a cada media de grupo de entrenamiento									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	943	147	143	Predicción	Dúctil	3	0	0
	Frágil	162	879	892		Frágil	0	3	3
	Fatiga	95	174	165		Fatiga	0	0	0
Distancia de Mahalanobis usando 150 armónicos de curvas de ensayo a cada media de grupo de entrenamiento									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	105	10	77	Predicción	Dúctil	0	0	0
	Frágil	1095	1190	1123		Frágil	3	3	3
	Fatiga	0	0	0		Fatiga	0	0	0
Distancia L2 usando un armónico de curvas de ensayo a cada media de grupo de entrenamiento									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	913	56	68	Predicción	Dúctil	3	0	0
	Frágil	26	785	689		Frágil	0	2	2
	Fatiga	261	359	443		Fatiga	0	1	1
Distancia L2 usando dos armónicos de curvas de ensayo a cada media de grupo de entrenamiento									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	913	56	68	Predicción	Dúctil	3	0	0
	Frágil	26	785	689		Frágil	0	2	2
	Fatiga	261	359	443		Fatiga	0	1	1
Distancia L2 usando tres armónicos de curvas de ensayo a cada media de grupo de entrenamiento									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	913	56	68	Predicción	Dúctil	3	0	0
	Frágil	26	785	689		Frágil	0	2	2
	Fatiga	261	359	443		Fatiga	0	1	1
Distancia L2 usando 150 armónicos de curvas de ensayo a cada media de grupo de entrenamiento									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	913	56	68	Predicción	Dúctil	3	0	0
	Frágil	26	785	689		Frágil	0	2	2
	Fatiga	261	359	443		Fatiga	0	1	1
Distancia Euclidiana en el primer puntaje de cada curva a la media del primer puntaje de cada grupo de entrenamiento									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	924	56	73	Predicción	Dúctil	3	0	0
	Frágil	18	791	690		Frágil	0	3	2
	Fatiga	258	353	437		Fatiga	0	0	1
Distancia Euclidiana en los 150 primeros puntajes de cada curva a la media del 150 primeros puntajes de cada grupo de entrenamiento									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	914	56	68	Predicción	Dúctil	3	0	0
	Frágil	26	784	689		Frágil	0	3	2
	Fatiga	260	360	443		Fatiga	0	0	1

Tabla A 2. Matrices de confusión para los clasificadores de MLG usando como variable explicativa, los datos funcionales no centrados.

Por curvas					Por fotos				
Familia Binomial									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	984	131	181	Predicción	Dúctil	3	0	0
	Frágil	43	840	726		Frágil	0	3	2
	Fatiga	173	229	293		Fatiga	0	0	1
Familia Gaussiana									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	998	139	197	Predicción	Dúctil	3	0	1
	Frágil	42	831	705		Frágil	0	3	2
	Fatiga	160	230	298		Fatiga	0	0	0
Familia Poisson									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	853	76	88	Predicción	Dúctil	3	0	0
	Frágil	6	656	519		Frágil	0	1	1
	Fatiga	341	468	593		Fatiga	0	2	2
Familia Quasi									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	998	139	197	Predicción	Dúctil	3	0	1
	Frágil	42	831	705		Frágil	0	3	2
	Fatiga	160	230	298		Fatiga	0	0	0
Familia Quasi-binomial									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	984	131	181	Predicción	Dúctil	3	0	0
	Frágil	43	840	726		Frágil	0	3	2
	Fatiga	173	229	293		Fatiga	0	0	1
Familia Quasi-poisson									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	853	76	88	Predicción	Dúctil	3	0	0
	Frágil	6	656	519		Frágil	0	1	1
	Fatiga	341	468	593		Fatiga	0	2	2

Tabla A 3. Matrices de confusión para los clasificadores de MLG usando como variables explicativas, los datos funcionales no centrados y las funciones de autocorrelación de cada dato funcional.

Por curvas					Por fotos				
Familia Binomial									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	857	124	208	Predicción	Dúctil	3	0	0
	Frágil	65	854	445		Frágil	0	3	0
	Fatiga	278	222	547		Fatiga	0	0	3
Familia Gaussiana									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	845	107	183	Predicción	Dúctil	3	0	0
	Frágil	65	847	455		Frágil	0	3	0
	Fatiga	290	246	562		Fatiga	0	0	3
Familia Poisson									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	802	82	148	Predicción	Dúctil	3	0	0
	Frágil	71	828	422		Frágil	0	2	0
	Fatiga	327	290	630		Fatiga	0	1	3
Familia Quasi									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	845	107	183	Predicción	Dúctil	3	0	0
	Frágil	65	847	455		Frágil	0	3	0
	Fatiga	290	246	562		Fatiga	0	0	3
Familia Quasi-binomial									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	857	124	208	Predicción	Dúctil	3	0	0
	Frágil	65	854	445		Frágil	0	3	0
	Fatiga	278	222	547		Fatiga	0	0	3
Familia Quasi-poisson									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	802	82	148	Predicción	Dúctil	3	0	0
	Frágil	71	828	422		Frágil	0	2	0
	Fatiga	327	290	630		Fatiga	0	1	3

Tabla A 4. Matrices de confusión para los clasificadores de MLG usando como variable explicativa, los datos funcionales centrados.

Por curvas					Por fotos				
Familia Binomial									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	984	131	181	Predicción	Dúctil	3	0	0
	Frágil	43	840	726		Frágil	0	3	2
	Fatiga	173	229	293		Fatiga	0	0	1
Familia Gaussiana									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	998	139	197	Predicción	Dúctil	3	0	1
	Frágil	42	831	705		Frágil	0	3	2
	Fatiga	160	230	298		Fatiga	0	0	0
Familia Poisson									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	853	76	88	Predicción	Dúctil	3	0	0
	Frágil	6	656	519		Frágil	0	1	1
	Fatiga	341	468	593		Fatiga	0	2	2
Familia Quasi									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	998	139	197	Predicción	Dúctil	3	0	1
	Frágil	42	831	705		Frágil	0	3	2
	Fatiga	160	230	298		Fatiga	0	0	0
Familia Quasi-binomial									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	984	131	181	Predicción	Dúctil	3	0	0
	Frágil	43	840	726		Frágil	0	3	2
	Fatiga	173	229	293		Fatiga	0	0	1
Familia Quasi-poisson									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	853	76	88	Predicción	Dúctil	3	0	0
	Frágil	6	656	519		Frágil	0	1	1
	Fatiga	341	468	593		Fatiga	0	2	2

Tabla A 5. Matrices de confusión para los clasificadores de MLG usando como variable explicativa, los datos funcionales centrados representados en la base de componentes principales.

Por curvas					Por fotos				
Usando un Armónico - Familia Binomial									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	1049	131	177	Predicción	Dúctil	3	0	0
	Frágil	32	846	734		Frágil	0	3	2
	Fatiga	119	223	289		Fatiga	0	0	1
Usando dos Armónicos - Familia Binomial									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	1033	124	169	Predicción	Dúctil	3	0	0
	Frágil	32	851	738		Frágil	0	3	2
	Fatiga	135	225	293		Fatiga	0	0	1
Usando 150 Armónicos - Familia Binomial									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	984	131	181	Predicción	Dúctil	3	0	0
	Frágil	43	840	726		Frágil	0	3	2
	Fatiga	173	229	293		Fatiga	0	0	1

Tabla A 6. Matrices de confusión para los clasificadores de MLG usando como variable explicativa, los datos funcionales derivados.

Por curvas					Por fotos				
Usando Primera Derivada - Familia Binomial									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	439	486	427	Predicción	Dúctil	1	3	1
	Frágil	318	347	262		Frágil	0	0	0
	Fatiga	443	367	511		Fatiga	2	0	2
Usando Segunda Derivada - Familia Binomial									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	423	384	419	Predicción	Dúctil	0	1	2
	Frágil	478	461	364		Frágil	3	2	0
	Fatiga	299	355	417		Fatiga	0	0	1

Tabla A 7. Matrices de confusión para los clasificadores de MAG usando como variable explicativa, los datos funcionales no centrados.

Por curvas					Por fotos				
Familia Binomial									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	982	132	183	Predicción	Dúctil	3	0	0
	Frágil	44	837	729		Frágil	0	3	2
	Fatiga	174	231	288		Fatiga	0	0	1
Familia Gaussiana									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	997	140	198	Predicción	Dúctil	3	0	1
	Frágil	45	827	709		Frágil	0	3	2
	Fatiga	158	233	293		Fatiga	0	0	0
Familia Poisson									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	852	76	88	Predicción	Dúctil	3	0	0
	Frágil	6	662	516		Frágil	0	1	1
	Fatiga	342	462	596		Fatiga	0	2	2
Familia Quasi									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	997	140	198	Predicción	Dúctil	3	0	1
	Frágil	45	827	709		Frágil	0	3	2
	Fatiga	158	233	293		Fatiga	0	0	0
Familia Quasi-binomial									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	982	132	183	Predicción	Dúctil	3	0	0
	Frágil	44	837	729		Frágil	0	3	2
	Fatiga	174	231	288		Fatiga	0	0	1
Familia Quasi-poisson									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	852	76	88	Predicción	Dúctil	3	0	0
	Frágil	6	662	516		Frágil	0	1	1
	Fatiga	342	462	596		Fatiga	0	2	2

Tabla A 8. Matrices de confusión para los clasificadores de MAG usando como variables explicativas, los datos funcionales no centrados y las funciones de autocorrelación de cada dato funcional.

Por curvas					Por fotos				
Familia Binomial									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	963	177	300	Predicción	Dúctil	3	0	1
	Frágil	74	885	497		Frágil	0	3	2
	Fatiga	163	138	403		Fatiga	0	0	0
Familia Gaussiana									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	961	151	263	Predicción	Dúctil	3	0	0
	Frágil	79	919	543		Frágil	0	3	3
	Fatiga	160	130	394		Fatiga	0	0	0
Familia Poisson									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	899	117	211	Predicción	Dúctil	3	0	0
	Frágil	68	874	439		Frágil	0	3	1
	Fatiga	233	209	550		Fatiga	0	0	2
Familia Quasi									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	961	151	263	Predicción	Dúctil	3	0	0
	Frágil	79	919	543		Frágil	0	3	3
	Fatiga	160	130	394		Fatiga	0	0	0
Familia Quasi-binomial									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	963	177	300	Predicción	Dúctil	3	0	1
	Frágil	74	885	497		Frágil	0	3	2
	Fatiga	163	138	403		Fatiga	0	0	0
Familia Quasi-poisson									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	899	117	211	Predicción	Dúctil	3	0	0
	Frágil	68	874	439		Frágil	0	3	1
	Fatiga	233	209	550		Fatiga	0	0	2

Tabla A 9. Matrices de confusión para los clasificadores de MAG usando como variable explicativa, los datos funcionales centrados.

Por curvas					Por fotos				
Familia Binomial									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	982	132	183	Predicción	Dúctil	3	0	0
	Frágil	44	836	730		Frágil	0	3	2
	Fatiga	174	232	287		Fatiga	0	0	1
Familia Gaussiana									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	997	140	198	Predicción	Dúctil	3	0	1
	Frágil	45	827	709		Frágil	0	3	2
	Fatiga	158	233	293		Fatiga	0	0	0
Familia Poisson									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	852	76	88	Predicción	Dúctil	3	0	0
	Frágil	6	662	516		Frágil	0	1	1
	Fatiga	342	462	596		Fatiga	0	2	2
Familia Quasi									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	997	140	198	Predicción	Dúctil	3	0	1
	Frágil	45	827	709		Frágil	0	3	2
	Fatiga	158	233	293		Fatiga	0	0	0
Familia Quasi-binomial									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	982	132	183	Predicción	Dúctil	3	0	0
	Frágil	44	836	730		Frágil	0	3	2
	Fatiga	174	232	287		Fatiga	0	0	1
Familia Quasi-poisson									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	852	76	88	Predicción	Dúctil	3	0	0
	Frágil	6	662	516		Frágil	0	1	1
	Fatiga	342	462	596		Fatiga	0	2	2

Tabla A 10. Matrices de confusión para los clasificadores de MAG usando como variable explicativa, los datos funcionales centrados representados en la base de componentes principales.

Por curvas					Por fotos				
Usando un Armónico - Familia Binomial									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	1049	131	177	Predicción	Dúctil	3	0	0
	Frágil	32	846	734		Frágil	0	3	2
	Fatiga	119	223	289		Fatiga	0	0	1
Usando dos Armónicos - Familia Binomial									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	1033	124	169	Predicción	Dúctil	3	0	0
	Frágil	32	851	738		Frágil	0	3	2
	Fatiga	135	225	293		Fatiga	0	0	1
Usando 150 Armónicos - Familia Binomial									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	982	132	183	Predicción	Dúctil	3	0	0
	Frágil	44	837	731		Frágil	0	3	2
	Fatiga	174	231	286		Fatiga	0	0	1

Tabla A 11. Matrices de confusión para los clasificadores de MAG usando como variable explicativa, los datos funcionales derivados.

Por curvas					Por fotos				
Usando Primera derivada - Familia Binomial									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	436	486	418	Predicción	Dúctil	1	3	1
	Frágil	316	356	271		Frágil	0	0	0
	Fatiga	448	358	511		Fatiga	2	0	2
Usando Segunda derivada - Familia Binomial									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	414	380	397	Predicción	Dúctil	1	1	1
	Frágil	471	458	381		Frágil	2	2	1
	Fatiga	315	362	422		Fatiga	0	0	1

Tabla A 12. Matrices de confusión para los clasificadores basados en medidas de profundidad, usando los datos funcionales no centrados.

Por curvas					Por fotos				
Frainan y Muniz									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	396	0	30	Predicción	Dúctil	1	0	0
	Frágil	804	1200	1170		Frágil	2	3	3
	Fatiga	0	0	0		Fatiga	0	0	0
Modal									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	57	0	13	Predicción	Dúctil	0	0	0
	Frágil	1143	1200	1187		Frágil	3	3	3
	Fatiga	0	0	0		Fatiga	0	0	0
Tukey Aleatoria									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	462	397	50	Predicción	Dúctil	1	0	0
	Frágil	733	803	1150		Frágil	2	3	3
	Fatiga	5	0	0		Fatiga	0	0	0
Proyectada Aleatoria									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	358	0	21	Predicción	Dúctil	0	0	0
	Frágil	842	1200	1179		Frágil	3	3	3
	Fatiga	0	0	0		Fatiga	0	0	0
Proyectada Doblemente Aleatoria									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	1	0	14	Predicción	Dúctil	0	0	0
	Frágil	1199	1200	1186		Frágil	3	3	3
	Fatiga	0	0	0		Fatiga	0	0	0
Extrema									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	702	6	3	Predicción	Dúctil	2	0	0
	Frágil	495	1191	1197		Frágil	1	3	3
	Fatiga	3	3	0		Fatiga	0	0	0

Tabla A 13. Matrices de confusión para los clasificadores basados en medidas de profundidad extrema, usando los datos funcionales no centrados y las funciones de autocorrelación.

Por curvas					Por fotos				
Extrema en Funciones de Autocorrelación									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	964	761	653	Predicción	Dúctil	3	3	2
	Frágil	105	270	141		Frágil	0	0	0
	Fatiga	131	169	406		Fatiga	0	0	1
Extrema Promediando los Rangos Obtenidos en Curvas Originales con los de las Funciones de Autocorrelación									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	876	160	194	Predicción	Dúctil	3	0	0
	Frágil	314	1004	830		Frágil	0	3	3
	Fatiga	10	36	176		Fatiga	0	0	0

Tabla A 14. Matriz de confusión para el clasificador basado en K-vecinos más cercanos, de los datos funcionales no centrados.

Por curvas					Por fotos				
K = 2									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	462	92	4	Predicción	Dúctil	1	0	0
	Frágil	0	3	0		Frágil	0	0	0
	Fatiga	738	1105	1196		Fatiga	2	3	3

Tabla A 15. Matrices de confusión para los clasificadores basados en MSV, usando los datos funcionales no centrados.

Por curvas					Por fotos				
Núcleo RBF									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	800	5	32	Predicción	Dúctil	3	0	0
	Frágil	143	1164	5		Frágil	0	3	0
	Fatiga	257	31	1163		Fatiga	0	0	3
Núcleo Polinomial									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	817	135	88	Predicción	Dúctil	3	0	0
	Frágil	22	982	9		Frágil	0	3	0
	Fatiga	361	83	1103		Fatiga	0	0	3
Núcleo Sigmoide									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	0	0	0	Predicción	Dúctil	0	0	0
	Frágil	0	0	0		Frágil	0	0	0
	Fatiga	1200	1200	1200		Fatiga	3	3	3
Núcleo Lineal									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	863	53	77	Predicción	Dúctil	3	0	0
	Frágil	41	762	690		Frágil	0	2	2
	Fatiga	296	385	433		Fatiga	0	1	1

Tabla A 16. Matriz de confusión para el clasificador basado en un árbol de decisión, de los datos funcionales no centrados.

Por curvas					Por fotos				
Gini y 25 de Profundidad									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	699	204	166	Predicción	Dúctil	3	0	0
	Frágil	153	744	220		Frágil	0	3	0
	Fatiga	348	252	814		Fatiga	0	0	3

Tabla A 17. Matriz de confusión para el clasificador basado en un bosque aleatorio, de los datos funcionales no centrados.

Por curvas					Por fotos				
0,04 Fracción de Características y 200 Estimadores									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	953	12	68	Predicción	Dúctil	3	0	0
	Frágil	41	1165	61		Frágil	0	3	0
	Fatiga	206	23	1071		Fatiga	0	0	3

Tabla A 18. Matriz de confusión para el clasificador basado en regresión logística, de los datos funcionales no centrados.

Por curvas					Por fotos				
Regresión Logística									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	842	73	68	Predicción	Dúctil	3	0	0
	Frágil	68	697	659		Frágil	0	2	2
	Fatiga	290	430	473		Fatiga	0	1	1

Tabla A 19. Matrices de confusión para los clasificadores basados en redes neuronales tipo perceptrón, usando los datos funcionales no centrados.

Por curvas					Por fotos				
Perceptron de Una Capa Oculta - Activación Relu - 300 Neuronas - 250 Iteraciones - Solucionador ADAM									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	729	65	52	Predicción	Dúctil	1	0	0
	Frágil	47	1027	11		Frágil	0	3	0
	Fatiga	424	108	1137		Fatiga	2	0	3

Perceptron de Doble Capa Oculta - Activación Relu - 300 y 12 Neuronas - 200 Iteraciones - Solucionador ADAM									
		Actual					Actual		
		Dúctil	Frágil	Fatiga			Dúctil	Frágil	Fatiga
Predicción	Dúctil	836	168	93	Predicción	Dúctil	3	0	0
	Frágil	23	919	0		Frágil	0	3	0
	Fatiga	341	113	1107		Fatiga	0	0	3



## Anexo B: Medidas de Exactitud

A continuación se encuentran las medidas de exactitud obtenidas para cada uno de los clasificadores, tanto “por curvas” como “por fotos”.

Tabla B 1. Medidas de exactitud obtenidas de los modelos de clasificación, basadas en distancias.

Por Curvas				Por Fotos			
Global	Dúctil	Frágil	Fatiga	Global	Dúctil	Frágil	Fatiga
Distancia Procruster							
				0,778	0,333	1,000	1,000
Distancia de Mahalanobis usando un armónico de curvas de ensayo a cada media de grupo de entrenamiento							
0,598	0,772	0,655	0,366	0,778	1,000	1,000	0,333
Distancia de Mahalanobis usando dos armónicos de curvas de ensayo a cada media de grupo de entrenamiento							
0,579	0,820	0,718	0,199	0,778	1,000	1,000	0,333
Distancia de Mahalanobis usando tres armónicos de curvas de ensayo a cada media de grupo de entrenamiento							
0,552	0,786	0,733	0,138	0,667	1,000	1,000	0,000
Distancia de Mahalanobis usando 150 armónicos de curvas de ensayo a cada media de grupo de entrenamiento							
0,360	0,088	0,992	0,000	0,333	0,000	1,000	0,000
Distancia L2 usando un armónico de curvas de ensayo a cada media de grupo de entrenamiento							
0,595	0,761	0,654	0,369	0,667	1,000	0,667	0,333
Distancia L2 usando dos armónicos de curvas de ensayo a cada media de grupo de entrenamiento							
0,595	0,761	0,654	0,369	0,667	1,000	0,667	0,333
Distancia L2 usando tres armónicos de curvas de ensayo a cada media de grupo de entrenamiento							
0,595	0,761	0,654	0,369	0,667	1,000	0,667	0,333
Distancia L2 usando 150 armónicos de curvas de ensayo a cada media de grupo de entrenamiento							
0,595	0,761	0,654	0,369	0,667	1,000	0,667	0,333
Distancia Euclidiana en el primer puntaje de cada curva a la media del primer puntaje de cada grupo de entrenamiento							
0,598	0,770	0,659	0,364	0,778	1,000	1,000	0,333
Distancia Euclidiana en los 150 primeros puntajes de cada curva a la media del 150 primeros puntajes de cada grupo de entrenamiento							
0,595	0,762	0,653	0,369	0,778	1,000	1,000	0,333

Tabla B 2. Medidas de exactitud obtenidas de los modelos de clasificación, basados en MLG usando como variable explicativa los datos funcionales no centrados.

Por curvas				Por fotos			
Global	Dúctil	Frágil	Fatiga	Global	Dúctil	Frágil	Fatiga
Familia Binomial							
0,588	0,820	0,700	0,244	0,778	1,000	1,000	0,333
Familia Gaussiana							
0,591	0,832	0,693	0,248	0,667	1,000	1,000	0,000
Familia Poisson							
0,584	0,711	0,547	0,494	0,667	1,000	0,333	0,667
Familia Quasi							
0,591	0,832	0,693	0,248	0,667	1,000	1,000	0,000
Familia Quasi-binomial							
0,588	0,820	0,700	0,244	0,778	1,000	1,000	0,333
Familia Quasi-poisson							
0,584	0,711	0,547	0,494	0,667	1,000	0,333	0,667

Tabla B 3. Medidas de exactitud obtenidas de los modelos de clasificación, basados en MLG usando como variables explicativas los datos funcionales no centrados y las funciones de autocorrelación de cada dato funcional.

Por curvas				Por fotos			
Global	Dúctil	Frágil	Fatiga	Global	Dúctil	Frágil	Fatiga
Familia Binomial							
0,627	0,714	0,712	0,456	1,000	1,000	1,000	1,000
Familia Gaussiana							
0,626	0,704	0,706	0,468	1,000	1,000	1,000	1,000
Familia Poisson							
0,628	0,668	0,690	0,525	0,889	1,000	0,667	1,000
Familia Quasi							
0,626	0,704	0,706	0,468	1,000	1,000	1,000	1,000
Familia Quasi-binomial							
0,627	0,714	0,712	0,456	1,000	1,000	1,000	1,000
Familia Quasi-poisson							
0,628	0,668	0,690	0,525	0,889	1,000	0,667	1,000

Tabla B 4. Medidas de exactitud obtenidas de los modelos de clasificación, basados en MLG usando como variable explicativa los datos funcionales centrados.

Por curvas				Por fotos			
Global	Dúctil	Frágil	Fatiga	Global	Dúctil	Frágil	Fatiga
Familia Binomial							
0,588	0,820	0,700	0,244	0,778	1,000	1,000	0,333
Familia Gaussiana							
0,591	0,832	0,693	0,248	0,667	1,000	1,000	0,000
Familia Poisson							
0,584	0,711	0,547	0,494	0,667	1,000	0,333	0,667
Familia Quasi							
0,591	0,832	0,693	0,248	0,667	1,000	1,000	0,000
Familia Quasi-binomial							
0,588	0,820	0,700	0,244	0,778	1,000	1,000	0,333
Familia Quasi-poisson							
0,584	0,711	0,547	0,494	0,667	1,000	0,333	0,667

Tabla B 5. Medidas de exactitud obtenidas de los modelos de clasificación, basados en MLG usando como variable explicativa, los datos funcionales centrados representados en la base de componentes principales.

Por curvas				Por fotos			
Global	Dúctil	Frágil	Fatiga	Global	Dúctil	Frágil	Fatiga
Usando un Armónico - Familia Binomial							
0,607	0,874	0,705	0,241	0,778	1,000	1,000	0,333
Usando dos Armónicos - Familia Binomial							
0,605	0,861	0,709	0,244	0,778	1,000	1,000	0,333
Usando 150 Armónicos - Familia Binomial							
0,588	0,820	0,700	0,244	0,778	1,000	1,000	0,333

Tabla B 6. Medidas de exactitud obtenidas de los modelos de clasificación, basados en MLG usando como variable explicativa los datos funcionales derivados.

Por curvas				Por fotos			
Global	Dúctil	Frágil	Fatiga	Global	Dúctil	Frágil	Fatiga
Usando Primera Derivada - Familia Binomial							
0,360	0,366	0,289	0,426	0,333	0,333	0,000	0,667
Usando Segunda Derivada - Familia Binomial							
0,361	0,353	0,384	0,348	0,333	0,000	0,667	0,333

Tabla B 7. Medidas de exactitud obtenidas de los modelos de clasificación, basados en MAG usando como variable explicativa los datos funcionales no centrados.

Por curvas				Por fotos			
Global	Dúctil	Frágil	Fatiga	Global	Dúctil	Frágil	Fatiga
Familia Binomial							
0,585	0,818	0,698	0,240	0,778	1,000	1,000	0,333
Familia Gaussiana							
0,588	0,831	0,689	0,244	0,667	1,000	1,000	0,000
Familia Poisson							
0,586	0,710	0,552	0,497	0,667	1,000	0,333	0,667
Familia Quasi							
0,588	0,831	0,689	0,244	0,667	1,000	1,000	0,000
Familia Quasi-binomial							
0,585	0,818	0,698	0,240	0,778	1,000	1,000	0,333
Familia Quasi-poisson							
0,586	0,710	0,552	0,497	0,667	1,000	0,333	0,667

Tabla B 8. Medidas de exactitud obtenidas de los modelos de clasificación, basados en MAG usando como variables explicativas los datos funcionales no centrados y las funciones de autocorrelación de cada dato funcional.

Por curvas				Por fotos			
Global	Dúctil	Frágil	Fatiga	Global	Dúctil	Frágil	Fatiga
Familia Binomial							
0,625	0,803	0,738	0,336	0,667	1,000	1,000	0,000
Familia Gaussiana							
0,632	0,801	0,766	0,328	0,667	1,000	1,000	0,000
Familia Poisson							
0,645	0,749	0,728	0,458	0,889	1,000	1,000	0,667
Familia Quasi							
0,632	0,801	0,766	0,328	0,667	1,000	1,000	0,000
Familia Quasi-binomial							
0,625	0,803	0,738	0,336	0,667	1,000	1,000	0,000
Familia Quasi-poisson							
0,645	0,749	0,728	0,458	0,889	1,000	1,000	0,667

Tabla B 9. Medidas de exactitud obtenidas de los modelos de clasificación, basados en MAG usando como variable explicativa los datos funcionales centrados.

Por curvas				Por fotos			
Global	Dúctil	Frágil	Fatiga	Global	Dúctil	Frágil	Fatiga
Familia Binomial							
0,585	0,818	0,697	0,239	0,778	1,000	1,000	0,333
Familia Gaussiana							
0,588	0,831	0,689	0,244	0,667	1,000	1,000	0,000
Familia Poisson							
0,586	0,710	0,552	0,497	0,667	1,000	0,333	0,667
Familia Quasi							
0,588	0,831	0,689	0,244	0,667	1,000	1,000	0,000
Familia Quasi-binomial							
0,585	0,818	0,697	0,239	0,778	1,000	1,000	0,333
Familia Quasi-poisson							
0,586	0,710	0,552	0,497	0,667	1,000	0,333	0,667

Tabla B 10. Medidas de exactitud obtenidas de los modelos de clasificación, basados en MAG usando como variable explicativa, los datos funcionales centrados representados en la base de componentes principales.

Por curvas				Por fotos			
Global	Dúctil	Frágil	Fatiga	Global	Dúctil	Frágil	Fatiga
Usando un Armónico - Familia Binomial							
0,607	0,874	0,705	0,241	0,778	1,000	1,000	0,333
Usando dos Armónicos - Familia Binomial							
0,605	0,861	0,709	0,244	0,778	1,000	1,000	0,333
Usando 150 Armónicos - Familia Binomial							
0,585	0,818	0,698	0,238	0,778	1,000	1,000	0,333

Tabla B 11. Medidas de exactitud obtenidas de los modelos de clasificación, basados en MAG usando como variable explicativa los datos funcionales derivados.

Por curvas				Por fotos			
Global	Dúctil	Frágil	Fatiga	Global	Dúctil	Frágil	Fatiga
Usando Primera Derivada - Familia Binomial							
0,362	0,363	0,297	0,426	0,333	0,333	0,000	0,667
Usando Segunda Derivada - Familia Binomial							
0,359	0,345	0,382	0,352	0,444	0,333	0,667	0,333

Tabla B 12. Medidas de exactitud obtenidas de los modelos de clasificación, basados en medidas de profundidad usando los datos funcionales no centrados.

Por curvas				Por fotos			
Global	Dúctil	Frágil	Fatiga	Global	Dúctil	Frágil	Fatiga
Fraiman y Muniz							
0,443	0,330	1,000	0,000	0,444	0,333	1,000	0,000
Modal							
0,349	0,048	1,000	0,000	0,333	0,000	1,000	0,000
Tukey Aleatoria							
0,351	0,385	0,669	0,000	0,444	0,333	1,000	0,000
Proyectada Aleatoria							
0,433	0,298	1,000	0,000	0,333	0,000	1,000	0,000
Proyectada Doblemente Aleatoria							
0,334	0,001	1,000	0,000	0,333	0,000	1,000	0,000
Extrema							
0,526	0,585	0,993	0,000	0,556	0,667	1,000	0,000

Tabla B 13. Medidas de exactitud obtenidas de los modelos de clasificación, basados en medidas de profundidad extrema usando los datos funcionales no centrados y las funciones de autocorrelación.

Por curvas				Por fotos			
Global	Dúctil	Frágil	Fatiga	Global	Dúctil	Frágil	Fatiga
Extrema en Funciones de Autocorrelación							
0,456	0,803	0,225	0,338	0,444	1,000	0,000	0,333
Extrema Promediando los Rangos Obtenidos en Curvas Originales con los de las Funciones de Autocorrelación							
0,571	0,730	0,837	0,147	0,667	1,000	1,000	0,000

Tabla B 14. Medidas de exactitud obtenidas de los modelos de clasificación, basados en K-vecinos más cercanos de los datos funcionales no centrados.

Por curvas				Por fotos			
Global	Dúctil	Frágil	Fatiga	Global	Dúctil	Frágil	Fatiga
K = 2							
0,461	0,385	0,003	0,997	0,444	0,333	0,000	1,000

Tabla B 15. Medidas de exactitud obtenidas de los modelos de clasificación, basados en MSV usando los datos funcionales no centrados.

Por curvas				Por fotos			
Global	Dúctil	Frágil	Fatiga	Global	Dúctil	Frágil	Fatiga
Núcleo RBF							
0,869	0,667	0,970	0,969	1,000	1,000	1,000	1,000
Núcleo Polinomial							
0,806	0,681	0,818	0,919	1,000	1,000	1,000	1,000
Núcleo Sigmoide							
0,333	0,000	0,000	1,000	0,333	0,000	0,000	1,000
Núcleo Lineal							
0,572	0,719	0,635	0,361	0,667	1,000	0,667	0,333

Tabla B 16. Medidas de exactitud obtenidas del modelo de clasificación, basado en un árbol de decisión de los datos funcionales no centrados.

Por curvas				Por fotos			
Global	Dúctil	Frágil	Fatiga	Global	Dúctil	Frágil	Fatiga
Gini y 25 de Profundidad							
0,627	0,583	0,620	0,678	1,000	1,000	1,000	1,000

Tabla B 17. Medidas de exactitud obtenidas del modelo de clasificación, basado en un bosque aleatorio de los datos funcionales no centrados.

Por curvas				Por fotos			
Global	Dúctil	Frágil	Fatiga	Global	Dúctil	Frágil	Fatiga
0,04 Fracción de Características y 200 Estimadores							
0,886	0,794	0,971	0,893	1,000	1,000	1,000	1,000

Tabla B 18. Medidas de exactitud obtenidas del modelo de clasificación, basado en regresión logística de los datos funcionales no centrados.

Por curvas				Por fotos			
Global	Dúctil	Frágil	Fatiga	Global	Dúctil	Frágil	Fatiga
Regresión Logística							
0,559	0,702	0,581	0,394	0,667	1,000	0,667	0,333

Tabla B 19. Medidas de exactitud obtenidas de los modelos de clasificación, basados en redes neuronales tipo perceptrón usando los datos funcionales no centrados.

Por curvas				Por fotos			
Global	Dúctil	Frágil	Fatiga	Global	Dúctil	Frágil	Fatiga
Perceptron de Una Capa Oculta - Activación Relu - 300 Neuronas - 250 Iteraciones - Solucionador ADAM							
0,804	0,608	0,856	0,948	0,778	0,333	1,000	1,000
Perceptron de Doble Capa Oculta - Activación Relu - 300 y 12 Neuronas - 200 Iteraciones - Solucionador ADAM							
0,795	0,697	0,766	0,923	1,000	1,000	1,000	1,000

## Bibliografía

[Alonso et al., 2012] Andrés M. Alonso, David Casado, Juan Romo, Supervised classification for functional data: A weighted distance approach, *Computational Statistics & Data Analysis*, Volume 56, Issue 7, 2012, Pages 2334-2346, ISSN 0167-9473, <https://doi.org/10.1016/j.csda.2012.01.013>.

[Ashraf et al., 2012] Hemeida, Ashraf & Hassan, Moatamad. (2022). Image classification based deep learning: A Review. *Aswan University Journal of Sciences and Technology*. 2. 10.21608/aujst.2022.259887.

[Blanquero et al., 2019] Rafael Blanquero, Emilio Carrizosa, Asunción Jiménez-Cordero, Belén Martín-Barragán, Variable selection in classification for multivariate functional data, *Information Sciences*, Volume 481, 2019, Pages 445-462, ISSN 0020-0255, <https://doi.org/10.1016/j.ins.2018.12.060>.

[Chen et al., 2021] Chen, L.; Li, S.; Bai, Q.; Yang, J.; Jiang, S.; Miao, Y. Review of Image Classification Algorithms Based on Convolutional Neural Networks. *Remote Sens*. 2021, 13, 4712. <https://doi.org/10.3390/rs13224712>.

[Delaigle et al., 2012] Delaigle, Aurore & Hall, Peter. (2012). Achieving near Perfect Classification for Functional Data. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*. 74. 267 - 286. 10.1111/j.1467-9868.2011.01003.x.

[Dryden et al., 2009] Ian L. Dryden, Alexey Koloydenko, Diwei Zhou "Non-Euclidean statistics for covariance matrices, with applications to diffusion tensor imaging," *The Annals of Applied Statistics*, *Ann. Appl. Stat.* 3(3), 1102-1123, (September 2009)

[Febrero et al., 2012] Febrero-Bande M, Oviedo de la Fuente M (2012). "Statistical Computing in Functional Data Analysis: The R Package fda.usc." *Journal of Statistical Software*, 51(4), 1–28. <https://www.jstatsoft.org/v51/i04/>.

[Ghiglietti et al., 2017] Ghiglietti, Andrea & Ieva, Francesca & Paganoni, Anna. (2017). Statistical inference for stochastic processes: Two-sample hypothesis tests. *Journal of Statistical Planning and Inference*. 180. 49-68. 10.1016/j.jspi.2016.08.004.

[Jiaohua et al., 2020] Jiaohua Qin, Wenyan Pan, Xuyu Xiang, Yun Tan, Guimin Hou, A biological image classification method based on improved CNN, *Ecological Informatics*, Volume 58, 2020, 101093, ISSN 1574-9541, <https://doi.org/10.1016/j.ecoinf.2020.101093>.

[Krishna et al., 2018] krishna, M & Neelima, M & Mane, Harshali & Matcha, Venu. (2018). Image classification using Deep learning. *International Journal of Engineering & Technology*. 7. 614. 10.14419/ijet.v7i2.7.10892.

[Krizhevsky et al., 2012] Krizhevsky, Alex & Sutskever, Ilya & Hinton, Geoffrey. (2012). ImageNet Classification with Deep Convolutional Neural Networks. *Neural Information Processing Systems*. 25. 10.1145/3065386.

[Kumar et al., 2012] Anil Kumar, C.P. Gandhi, Yuqing Zhou, Rajesh Kumar, Jiawei Xiang, Improved deep convolution neural network (CNN) for the identification of defects in the centrifugal pump using acoustic images, *Applied Acoustics*, Volume 167, 2020, 107399, ISSN 0003-682X, <https://doi.org/10.1016/j.apacoust.2020.107399>.

[Llop et al., 2008] P. Llop, L. Forzani, R. Fraiman, On local times, density estimation and supervised classification from functional data, *Journal of Multivariate Analysis*, Volume 102, Issue 1, 2011, Pages 73-86, ISSN 0047-259X, <https://doi.org/10.1016/j.jmva.2010.08.002>.

[Nilsback et al., 2008] M. -E. Nilsback and A. Zisserman, "Automated Flower Classification over a Large Number of Classes," 2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing, Bhubaneswar, India, 2008, pp. 722-729, doi: 10.1109/ICVGIP.2008.47.

[Oviedo et al., 2011] Manuel Oviedo de la Fuente, Manuel Febrero Bande, Localización: X Congreso SGAPEIO, Congreso Galego de Estatística e Investigación de Operacións: Pazo de Congresos e Exposicións, Pontevedra, 3, 4 y 5 de novembro de 2011, 2011, ISBN 978-84-938642-2-4.

[Pedregosa et al., 2011] Pedregosa, F., Varoquaux, Ga"el, Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... others. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12(Oct), 2825–2830.

[R Core Team] R Core Team (2021). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL <https://www.R-project.org/>.

[Radhesyam et al., 2020] Radhesyam Vaddi, Prabukumar Manoharan, Hyperspectral image classification using CNN with spectral and spatial features integration, *Infrared Physics & Technology*, Volume 107, 2020, 103296, ISSN 1350-4495, <https://doi.org/10.1016/j.infrared.2020.103296>.

[Ramsay et al., 2005] J. O. Ramsay, B. W. Silverman, *Functional Data Analysis*, Springer, 2005, New York, ISBN 978-0-387-40080-8, <https://doi.org/10.1007/b98888>.

[Sabat-Tomala et al., 2020] Sabat-Tomala, A.; Raczko, E.; Zagajewski, B. Comparison of Support Vector Machine and Random Forest Algorithms for Invasive and Expansive Species Classification Using Airborne Hyperspectral Data. *Remote Sens.* 2020, 12, 516. <https://doi.org/10.3390/rs12030516>

[Sheykhmousa et al., 2020] M. Sheykhmousa, M. Mahdianpari, H. Ghanbari, F. Mohammadimanesh, P. Ghamisi and S. Homayouni, "Support Vector Machine Versus Random Forest for Remote Sensing Image Classification: A Meta-Analysis and Systematic Review," in *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 13, pp. 6308-6325, 2020, doi: 10.1109/JSTARS.2020.3026724.

[Shuoyang et al., 2021] Shuoyang Wang, Zuofeng Shang, Guanqun Cao and Jun S. Liu, Optimal classification for functional data, 2021, arXiv:2103.00569 [stat.ME]

[Van Rossum et al., 1995] Van Rossum G and Drake Jr FL. Python reference manual. Centrum voor Wiskunde en Informatica Amsterdam; 1995.

[Waqar et al., 2019] M. Waqar Akram, Guiqiang Li, Yi Jin, Xiao Chen, Changan Zhu, Xudong Zhao, Abdul Khaliq, M. Faheem, Ashfaq Ahmad, CNN based automatic detection of photovoltaic cell defects in electroluminescence images, Energy, Volume 189, 2019, 116319, ISSN 0360-5442, <https://doi.org/10.1016/j.energy.2019.116319>.

[Watanabe et al., 2019] Akihiko Watanabe, Naoto Hirose, Hyoungseop Kim, Ichiro Omura, Convolutional neural network (CNNs) based image diagnosis for failure analysis of power devices, Microelectronics Reliability, Volumes 100–101, 2019, 113399, ISSN 0026-2714, <https://doi.org/10.1016/j.microrel.2019.113399>.

[Xinyu et al., 2023] Xinyu Huang, Ziyang Pan, A Functional Data Classification Model Utilizing Functional Mahalanobis Distance and Regenerative Kernel Methods. Journal of Electronics and Information Science (2023) Vol. 8: 104-110. DOI: <http://dx.doi.org/10.23977/10.23977/jeis.2023.080613>.

[Yeonjoo et al., 2019] Yeonjoo Park, Douglas G. Simpson, Robust probabilistic classification applicable to irregularly sampled functional data, Computational Statistics & Data Analysis, Volume 131, 2019, Pages 37-49, ISSN 0167-9473, <https://doi.org/10.1016/j.csda.2018.08.001>.

[Yinfeng et al., 2016] Yinfeng Meng, Jiye Liang, Yuhua Qian, Comparison study of orthonormal representations of functional data in classification, Knowledge-Based Systems, Volume 97, 2016, Pages 224-236, ISSN 0950-7051, <https://doi.org/10.1016/j.knosys.2015.12.016>.

[Zhao et al., 2020] Yudi Zhao, Kuangrong Hao, Haibo He, Xuesong Tang, Bing Wei, A visual long-short-term memory based integrated CNN model for fabric defect image

classification, Neurocomputing, Volume 380, 2020, Pages 259-270, ISSN 0925-2312, <https://doi.org/10.1016/j.neucom.2019.10.067>.