



*Benemérita Universidad Autónoma de Puebla*  
*Facultad de Ciencias Físico Matemáticas*  
*Posgrado en Ciencias Matemáticas*

***Análisis de imágenes de  
resonancia magnética  
aplicado al diagnóstico del  
Alzheimer***

*Tesis que se presenta como requisito final para obtener  
el título de*

***Maestro en Ciencias (Matemáticas)***

*Presenta: Lic. Jaicer Jonás López Rivero.*

*Director de tesis: Dr. Hugo Adán Cruz Suárez.*

*Puebla, Puebla, Diciembre de 2020.*



# Dedicatoria

Dedicado a mi familia. En especial a mis padres, esposa e hijos.



# Agradecimientos

Quiero agradecer a todas las personas que colaboraron en mi formación tanto académica como personal. Primeramente agradecerle a Dios y a mi familia, a mis padres Josefina Rivero y Omar López, sin ellos no fuera sido posible este logro, a mi esposa Luz Alvarado y a mis tres hijos Jaicer Fabian, Adrian Jonás y Camila Valentina, los cuales siempre me dan ese impulso de seguir adelante y cumplir mis metas.

A quienes me ayudaron para poder emprender este viaje desde Barquisimeto a Puebla, mis amigos de la UCLA: Rafael, Uvencio, Karla, Diana, Shaday y Harry. A mi tía Dilcia y mi tía Luisa. A quienes conocí en este hermoso país como lo es México, entre ellos: Anel, mis compañeros de clase y del Laboratorio de Probabilidad y Estadística.

Al CONACYT por su apoyo económico y al personal de la FCFM de la BUAP. A los profesores de los cuales recibí una excelente formación, entre ellos mi asesor el Dr. Hugo Cruz.

A todos ellos, GRACIAS.



# Capítulo 1

## Introducción

La enfermedad de Alzheimer (AD, por sus siglas en inglés) es un proceso neuro-degenerativo, progresivo, no-reversible y hasta el momento incurable, el cual afecta progresivamente la memoria, el pensamiento y la habilidad para realizar actividades de la vida diaria, y conduce a un estado de discapacidad y dependencia. La AD es la forma más común de demencia y aparece con mayor frecuencia en personas mayores de 65 años. Se sabe que la AD comienza décadas antes del inicio de la sintomatología, algunos estudios hablan de 20 o 30 años, es por ello que la detección temprana y clasificación de la AD son tareas importantes de apoyo clínico para los médicos, ya que con el diagnóstico temprano se pueden introducir medidas terapéuticas destinadas a disminuir la progresión de la enfermedad.

El diagnóstico por imagen es una especialidad dentro de la medicina que permite obtener imágenes de alta resolución, de modo que se puedan señalar posibles patologías tras la visualización de estas. Los avances técnicos en los equipos de resonancia magnética han permitido recopilar más datos y conseguir una mayor definición de las imágenes, pero estas representaciones no tienen suficiente detalle, y es allí donde las técnicas modernas y la inteligencia computacional nos permitirán extraer información que no se puede ver a simple vista.

La AD causa la muerte de neuronas y la pérdida de tejido en todo el cerebro, con el tiempo el cerebro se encoge dramáticamente, afectando casi todas sus funciones. Según cifras de la Organización Mundial de la Salud [13], la población mundial está envejeciendo a pasos acelerados. Entre los años 2000 y 2050 la proporción de los habitantes del planeta mayores de 60 años se duplicará pasando del 11% al 22%. A medida que las personas vivan más tiempo, en todo el mundo se producirá un aumento de la cantidad de casos de demencia, como la enfermedad de Alzheimer.

La resonancia magnética puede detectar anomalías cerebrales asociadas con el deterioro cognitivo leve y se puede utilizar para predecir pacientes que podrían eventualmente desarrollar la enfermedad de Alzheimer.

Actualmente se han propuesto distintos métodos que analizan imágenes de resonancia magnética (MRI) para detectar la AD, estos se diferencian en las técnicas utilizadas y en los distintos índices de validación obtenidos. Debesh Jha *et al.* [5] proponen un modelo que incluye el filtrado de Wiener, la transformada wavelet discreta bidimensional (2D-DWT), el análisis probabilístico de componentes principales (PPCA) junto con el algoritmo de  $K$  vecinos más cercanos (KNN) como clasificador. También en [9] proponen un modelo mediante el uso de la transformada wavelet compleja de árbol dual (DTCWT), el análisis de componentes principales (PCA), el análisis discriminante lineal y la máquina de aprendizaje extremo (ELM). Lama *et al.* [14] han analizado imágenes de resonancia magnética estructural usando ELM generalizado y características de PCA para el diagnóstico de la AD. Sandeep *et al.* [15] estratificaron la AD utilizando coeficientes de wavelet discretos como un elemento para preparar y probar máquinas de vectores de soporte (SVM) y clasificadores de redes neuronales.

En este trabajo se plantea el problema de analizar MRI para detectar características del cerebro que nos puedan llevar a concluir si se puede o no diagnosticar la AD. Se propone un algoritmo secuencial que emplea técnicas estadísticas y de aprendizaje automático entre las que se encuentran la transformada wavelet discreta bidimensional, el análisis de componentes principales, el discriminante lineal de Fisher (DLF) y la máquina de aprendizaje extremo. Estas técnicas son aplicadas de manera secuencial y en cada paso cada una de ellas cumple con un objetivo particular, la 2D-DWT nos permite extraer las características relevantes de la imagen original, el PCA elimina la posible correlación entre los píxeles de la imagen, el DLF logra una mayor separabilidad entre las clases y la ELM es el algoritmo que se encarga de la clasificación de las imágenes. Esta propuesta se basa en buena medida en los artículos mencionados previamente ([5], [9], [14], [15]).

La Tesis se encuentra organizada del modo siguiente: en el Capítulo 2 se presentan los preliminares, en donde se desarrollan de manera teórica las técnicas que se van a utilizar. Luego en el Capítulo 3 se describe la metodología y se presentan los resultados obtenidos luego de la implementación sobre un conjunto de imágenes de resonancia magnética, el cual comprende 80 imágenes pertenecientes a personas con la AD y 89 imágenes pertenecientes a personas cognitivamente normal (CN), es decir, sin la presencia de síntomas de la enfermedad.

# Índice general

<b>Dedicatoria</b>	<b>3</b>
<b>Agradecimientos</b>	<b>5</b>
<b>1. Introducción</b>	<b>7</b>
<b>2. Preliminares</b>	<b>11</b>
2.1. Transformada Wavelet. . . . .	11
2.1.1. Análisis Multiresolución. . . . .	15
2.1.2. La Transformada Wavelet Discreta Bidimensional. . . . .	20
2.2. Análisis de Componentes Principales. . . . .	22
2.3. Discriminante Lineal de Fisher. . . . .	25
2.4. Máquina de Aprendizaje Extremo. . . . .	27
<b>3. Metodología Propuesta.</b>	<b>35</b>
3.1. Resultados Experimentales y Análisis. . . . .	36
3.2. Resumen y Conclusiones. . . . .	42
<b>Bibliografía</b>	<b>44</b>



## Capítulo 2

# Preliminares

En este Capítulo se describen de manera ordenada de acuerdo a su implementación las técnicas que se utilizarán en nuestra propuesta. Comenzando con la transformada wavelet, el análisis de componentes principales, el discriminante lineal de Fisher y el algoritmo de la máquina de aprendizaje extremo.

### 2.1. Transformada Wavelet.

En lo que sigue consideramos el espacio  $L^2(\mathbb{R})$  de las funciones Lebesgue medibles  $f : \mathbb{R} \rightarrow \mathbb{C}$ , que satisfacen:

$$\int_{-\infty}^{\infty} |f(t)|^2 dt < \infty.$$

El espacio  $L^2(\mathbb{R})$  es un espacio de Hilbert, cuyo producto escalar viene dado por:

$$\langle f, g \rangle = \int_{-\infty}^{\infty} f(t) \overline{g(t)} dt, \quad \forall f, g \in L^2(\mathbb{R}).$$

En este espacio consideramos la norma:

$$\|f\|_2 = \langle f, f \rangle^{1/2}.$$

La transformada wavelet es una técnica que aborda el problema de la extracción de características, especialmente en imágenes médicas [1]. Dicha técnica es una alternativa a la transformada corta de Fourier. Su característica más importante es que analiza diferentes componentes de frecuencia de una señal con diferentes resoluciones.

La extracción de características esenciales de las MRI es una tarea imprescindible para un análisis adecuado de estas. La transformada wavelet es producida por dilataciones y traslaciones de la llamada función wavelet madre.

**Definición 2.1.** Se dice que  $\psi \in L^2(\mathbb{R})$  satisface la condición de admisibilidad, si

$$C_\psi := 2 \int_0^\infty \frac{|\Psi(\omega)|^2}{\omega} d\omega < \infty, \quad (2.1)$$

donde  $\Psi(\omega)$  es la transformada de Fourier de  $\psi(t)$ , es decir,

$$\Psi(\omega) = \int_{-\infty}^\infty \psi(t) e^{-i2\pi\omega t} dt, \omega \in \mathbb{R}.$$

Si  $\psi \in L^2(\mathbb{R})$  satisface la condición de admisibilidad entonces  $\psi$  es llamada una wavelet madre.

**Ejemplo 2.1.** Existen una importante cantidad de familias de funciones wavelet, pero las más conocidas y que han probado ser útiles son las siguientes: Haar, Daubechies, Biortogonal, Coiflets, Symlets, Morlet, Sombrero mexicano y Meyer, entre otras (Ver Figura 2.1).

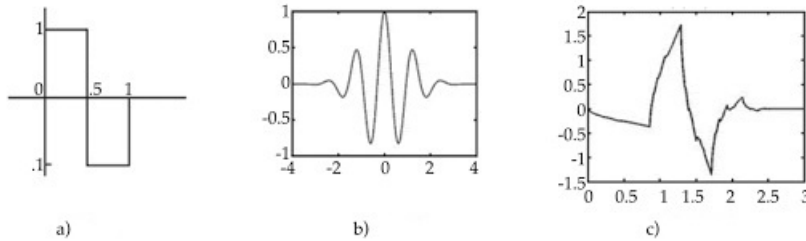


Figura 2.1: Wavelet madre: a) Haar, b) Morlet, c) Daubechies.

Las versiones desplazadas y dilatadas de la wavelet madre se denotan:

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right), t \in \mathbb{R},$$

donde  $a > 0$  representa el parámetro de escala y  $b \in \mathbb{R}$  el parámetro de traslación.

**Definición 2.2.** En relación con cada wavelet madre, la transformada wavelet continua en  $L^2(\mathbb{R})$  se define por:

$$\begin{aligned} Tf_\psi(a, b) &= \int_{-\infty}^\infty f(t) \overline{\psi_{a,b}(t)} dt, \\ &= \langle f, \psi_{a,b} \rangle, f \in L^2(\mathbb{R}), a \in \mathbb{R}^+, b \in \mathbb{R}. \end{aligned}$$

La transformada wavelet continua mide la variación de  $f$  en una vecindad de  $b$ , cuyo tamaño es proporcional a “ $a$ ”.

**Ejemplo 2.2.** En la Figura 2.2 [12] observamos una función sinusoidal y su respectiva transformada wavelet continua, para este ejemplo se utilizó como wavelet madre a el sombrero Mexicano, el cual viene dado por:

$$\psi(t) = (1 - t^2) e^{-t^2/2}, t \in \mathbb{R}.$$

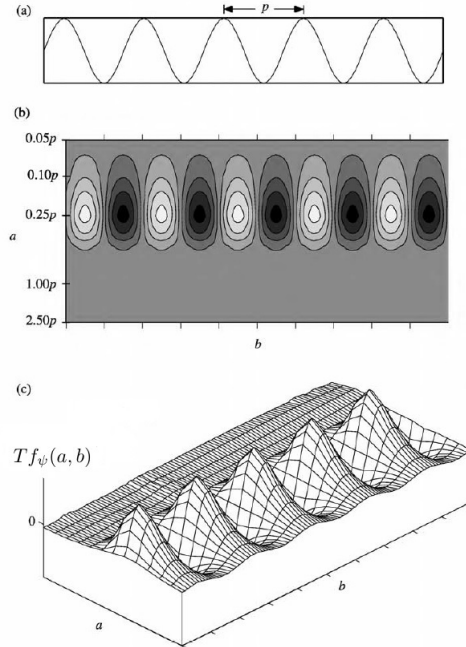


Figura 2.2: Transformada wavelet continua. (a) Función sinusoidal de periodo  $p$ . (b) Gráfico de contorno. (c) Gráfico de superficie.

**Teorema 2.1.** Sea  $\psi$  una wavelet madre que define una transformada wavelet continua  $Tf_\psi$ . Entonces para cualquier  $f \in L^2(\mathbb{R})$  y  $t \in \mathbb{R}$  en la que  $f$  es continua,

$$f(t) = \frac{2}{C_\psi} \int_0^\infty \left[ \int_{-\infty}^\infty Tf_\psi(a, b) \psi_{a,b}(t) db \right] \frac{da}{a^2}.$$

Una demostración del Teorema 2.1 puede ser consultada en [4] (Teorema 3.11, p. 64).

**Observación 2.1.** Observemos que la condición de admisibilidad (2.1) de la wavelet es necesaria para garantizar la invertibilidad de la transformada.

La transformada wavelet continua tiene dos inconvenientes: redundancia y falta de practicidad. El primero es obvio por la naturaleza de la transformada wavelet y el segundo por el hecho de que ambos parámetros de la transformada son continuos. Podemos tratar de resolver ambos problemas muestreando los parámetros  $a$  y  $b$  para obtener un conjunto de funciones wavelet en parámetros discretos.

La cuadrícula de muestreo se define de la siguiente manera:

$$a = a_0^j \text{ y } b = kb_0a_0^j, \text{ donde } j, k \in \mathbb{Z} \text{ y } a_0 > 1, b_0 > 0 \text{ fijos.}$$

Las opciones comunes para los parámetros  $a_0$  y  $b_0$  son 2 y 1, respectivamente, la cual se conoce como la cuadrícula diádica. De esta manera vemos que la wavelet de cuadrícula diádica se puede escribir como:

$$\begin{aligned} \psi_{j,k}(t) &= \frac{1}{\sqrt{2^j}} \psi\left(\frac{t - k2^j}{2^j}\right) \\ &= 2^{-\frac{j}{2}} \psi(2^{-j}t - k). \end{aligned}$$

El símbolo de Kronecker

$$\delta_{j,k} := \begin{cases} 1 & \text{si } j = k, \\ 0 & \text{si } j \neq k, \end{cases}$$

definido en  $\mathbb{Z} \times \mathbb{Z}$ , se usará a menudo.

**Definición 2.3.** Una función  $\psi \in L^2(\mathbb{R})$  se llama wavelet ortogonal, si la familia  $\{\psi_{j,k}\}_{(j,k) \in \mathbb{Z}^2}$  es una base ortonormal de  $L^2(\mathbb{R})$ , es decir,

$$\langle \psi_{j,k}, \psi_{l,m} \rangle = \delta_{j,l} \delta_{k,m},$$

$j, k, l, m \in \mathbb{Z}$  y cada  $f \in L^2(\mathbb{R})$  se puede escribir como:

$$f(t) = \sum_{j,k \in \mathbb{Z}} c_{j,k} \psi_{j,k}, \quad (2.2)$$

donde la convergencia de la serie en (2.2) es en  $L^2(\mathbb{R})$ .

**Observación 2.2.** La representación en serie de  $f$  en (2.2) se denomina serie wavelet. De forma análoga a la noción de coeficientes de Fourier, los coeficientes wavelet  $c_{j,k}$  están dados por:

$$c_{j,k} = \langle f, \psi_{j,k} \rangle.$$

**Ejemplo 2.3.** El ejemplo más simple de una wavelet ortogonal es la función de Haar  $\psi_H$  definida por:

$$\psi_H(t) := \begin{cases} 1 & 0 \leq t < \frac{1}{2}, \\ -1 & \frac{1}{2} \leq t < 1, \\ 0 & \text{en otro caso.} \end{cases}$$

Se pueden construir wavelets  $\psi$  tal que la familia  $\{\psi_{j,k}\}_{(j,k)\in\mathbb{Z}^2}$  sea una base ortonormal de  $L^2(\mathbb{R})$ . La construcción de la wavelet  $\psi_H$  se presenta en el Ejemplo 2.4. La búsqueda de wavelets ortogonales comienza con aproximaciones multiresolución.

### 2.1.1. Análisis Multiresolución.

Las aproximaciones multiresolución calculan la aproximación de una función a varias resoluciones con proyecciones ortogonales en diferentes espacios  $\{V_j\}_{j\in\mathbb{Z}}$ . La aproximación de una función en una resolución  $2^{-j}$  se define como una proyección ortogonal en un espacio  $V_j \subset L^2(\mathbb{R})$ . El espacio  $V_j$  reagrupa todas las aproximaciones posibles en la resolución  $2^{-j}$ . La proyección ortogonal de  $f$  es la función  $P_{V_j}f \in V_j$  que minimiza  $\|f - P_{V_j}f\|_2$ . Para evitar confusiones, recordemos que el parámetro de escala  $2^j$  es el inverso de la resolución  $2^{-j}$ .

**Definición 2.4.** La familia  $\{f_k\}_{k\in\mathbb{Z}}$  es una base de Riesz para  $L^2(\mathbb{R})$  si se cumplen las dos propiedades siguientes:

1. El espacio generado

$$\langle f_k : k \in \mathbb{Z} \rangle$$

es denso en  $L^2(\mathbb{R})$ , y

2. Existen constantes positivas  $A$  y  $B$ , con  $0 < A \leq B < \infty$ , tales que:

$$A\|\{c_k\}\|_{l^2}^2 \leq \left\| \sum_{k\in\mathbb{Z}} c_k f_k \right\|_2^2 \leq B\|\{c_k\}\|_{l^2}^2,$$

$\forall \{c_k\} \in l^2(\mathbb{Z})$ . A las constantes  $A$  y  $B$  se les llama cota inferior y cota superior de la base de Riesz, respectivamente.

**Observación 2.3.**  $l^2(\mathbb{Z})$  denota el espacio de las sucesiones  $\{c_k\}_{k\in\mathbb{Z}}$  doblemente infinitas tales que:

$$\sum_{k=-\infty}^{\infty} |c_k|^2 < \infty,$$

y cuya norma viene dada por:

$$\|\{c_k\}\|_{l^2}^2 = \left( \sum_{k=-\infty}^{\infty} |c_k|^2 \right)^{1/2}.$$

**Definición 2.5.** Una sucesión  $\{V_j\}_{j\in\mathbb{Z}}$  de subespacios cerrados de  $L^2(\mathbb{R})$  es un análisis multiresolución (MRA) de  $L^2(\mathbb{R})$  si se cumplen las siguientes propiedades:

1.  $\forall (j, k) \in \mathbb{Z}^2, f(t) \in V_j \Leftrightarrow f(t - 2^j k) \in V_j,$
2.  $\forall j \in \mathbb{Z}, V_{j+1} \subset V_j,$
3.  $\forall j \in \mathbb{Z}, f(t) \in V_j \Leftrightarrow f(\frac{t}{2}) \in V_{j+1},$
4.  $\lim_{j \rightarrow +\infty} V_j = \bigcap_{j=-\infty}^{+\infty} V_j = \{0\},$
5.  $\lim_{j \rightarrow -\infty} V_j = \overline{\bigcup_{j=-\infty}^{+\infty} V_j} = L^2(\mathbb{R}), y$
6. Existe  $\phi \in L^2(\mathbb{R})$  tal que  $\{\phi(t - k)\}_{k \in \mathbb{Z}}$  es una base de Riesz de  $V_0$ .

A la función  $\phi$  se llama función de escala, la cual genera un MRA de  $L^2(\mathbb{R})$ .

**Observación 2.4.** La propiedad 1 significa que  $V_j$  es invariante por cualquier traslación proporcional a la escala  $2^j$ . La inclusión de la propiedad 2 nos dice que una aproximación en una resolución  $2^{-j}$  contiene toda la información necesaria para calcular una aproximación en una resolución más gruesa  $2^{-j-1}$ . Cuando la resolución  $2^{-j}$  tiende a 0 la propiedad 4 implica que:

$$\lim_{j \rightarrow +\infty} \|P_{V_j} f\| = 0.$$

Por otro lado, cuando la resolución  $2^{-j}$  tiende a  $+\infty$ , la propiedad 5 impone que la aproximación de la función converja a la función original:

$$\lim_{j \rightarrow -\infty} \|f - P_{V_j} f\| = 0.$$

La aproximación de  $f$  en la resolución  $2^{-j}$  se define como la proyección ortogonal  $P_{V_j} f$  en  $V_j$ . Para calcular esta proyección, debemos encontrar una base ortonormal de  $V_j$ .

El siguiente teorema ortogonaliza la base de Riesz  $\{\phi(t - k)\}_{k \in \mathbb{Z}}$  y construye una base ortogonal de cada espacio  $V_j$  utilizando la función de escala  $\phi$ .

**Teorema 2.2.** Sea  $\{V_j\}$  un MRA de  $L^2(\mathbb{R})$  generado por la función de escala  $\phi$ . Denotemos

$$\phi_{j,k}(t) = 2^{-\frac{j}{2}} \phi(2^{-j} t - k),$$

entonces la familia  $\{\phi_{j,k}\}_{k \in \mathbb{Z}}$  es una base ortonormal de  $V_j$  para todo  $j \in \mathbb{Z}$ .

Una demostración del Teorema 2.2 puede ser consultada en [17] (Teorema 7.1, p. 225).

La función de escala  $\phi$  es adecuada para codificar una función  $f \in L^2(\mathbb{R})$  por completo, pero una descomposición basada en la función de escala y una función wavelet asociada es más eficiente, ya que por medio de la función wavelet se recuperan los detalles que se pierden cuando solo se utiliza la función de escala.

Dado que  $V_{j+1} \subset V_j$ ,  $\forall j \in \mathbb{Z}$ , en particular se tiene que  $2^{-\frac{1}{2}}\phi(\frac{t}{2}) \in V_1 \subset V_0$ . Además  $\{\phi(t-k)\}_{k \in \mathbb{Z}}$  es una base ortonormal de  $V_0$ , por tanto existe  $\{h_k\}_{k \in \mathbb{Z}} \in l^2(\mathbb{Z})$  tal que:

$$\phi(t) = \sqrt{2} \sum_{k \in \mathbb{Z}} h_k \phi(2t - k), t \in \mathbb{R}. \quad (2.3)$$

La ecuación (2.3) es conocida como ecuación de escala.

Sea  $W_j$  el complemento ortogonal de  $V_j$  en  $V_{j-1}$ :

$$V_{j-1} = V_j \oplus W_j.$$

La proyección ortogonal de  $f$  en  $V_{j-1}$  puede descomponerse como la suma de las proyecciones ortogonales en  $V_j$  y  $W_j$ :

$$P_{V_{j-1}}f = P_{V_j}f + P_{W_j}f.$$

El complemento  $P_{W_j}f$  proporciona los “detalles” que aparecen en la escala  $2^{j-1}$  pero que desaparecen en la escala más gruesa  $2^j$ .

El siguiente teorema prueba que a partir de una función de escala  $\phi$  se puede construir una wavelet  $\psi$  que genere una base ortonormal para  $W_j$  y para  $L^2(\mathbb{R})$ .

**Teorema 2.3.** *Sea  $\phi$  una función de escala y la correspondiente  $l^2$ -sucesión  $\{h_k\}_{k \in \mathbb{Z}}$  asociada a la ecuación de escala. Definimos la sucesión  $\{g_k\}_{k \in \mathbb{Z}}$  dada por:*

$$g_k = (-1)^k \bar{h}_{(1-k)}, k \in \mathbb{Z}, \quad (2.4)$$

y la wavelet  $\psi(t)$  dada por:

$$\psi(t) = \sqrt{2} \sum_{k \in \mathbb{Z}} g_k \phi(2t - k). \quad (2.5)$$

Entonces, para cualquier escala  $2^j$ ,  $\{\psi_{j,k}\}_{k \in \mathbb{Z}}$  es una base ortonormal de  $W_j$  y para todas las escalas,  $\{\psi_{j,k}\}_{(j,k) \in \mathbb{Z}^2}$  es una base ortonormal de  $L^2(\mathbb{R})$ .

Una demostración del Teorema 2.3 puede ser consultada en [19] (Teorema 7.35, p. 185).

**Ejemplo 2.4.** *La función wavelet de Haar se construye a partir de la función de escala  $\phi(t) = \chi_{[0,1)}(t)$ . Dicha función se puede escribir como:*

$$\begin{aligned} \phi(t) &= \phi(2t) + \phi(2t - 1) \\ &= \frac{1}{\sqrt{2}}\phi_{1,0}(t) + \frac{1}{\sqrt{2}}\phi_{1,1}(t). \end{aligned}$$

Por lo tanto,

$$h_k = \begin{cases} \frac{1}{\sqrt{2}} & \text{si } k = 0, 1, \\ 0 & \text{si } k \neq 0, 1, \end{cases}$$

y de acuerdo a la ecuación (2.4) del Teorema 2.3, se tiene que:

$$g_k = \begin{cases} \frac{1}{\sqrt{2}} & \text{si } k = 0, \\ -\frac{1}{\sqrt{2}} & \text{si } k = 1, \\ 0 & \text{si } k \neq 0, 1. \end{cases}$$

Así, de acuerdo con la ecuación (2.5), se tiene que:

$$\begin{aligned} \psi(t) &= \frac{1}{\sqrt{2}}\phi_{1,0}(t) - \frac{1}{\sqrt{2}}\phi_{1,1}(t) \\ &= \phi(2t) - \phi(2t-1) \\ &= \chi_{[0,1/2)}(t) - \chi_{[1/2,1)}(t) \\ &= \psi_H(t). \end{aligned}$$

Ambas funciones se observan en la Figura 2.3 [12].

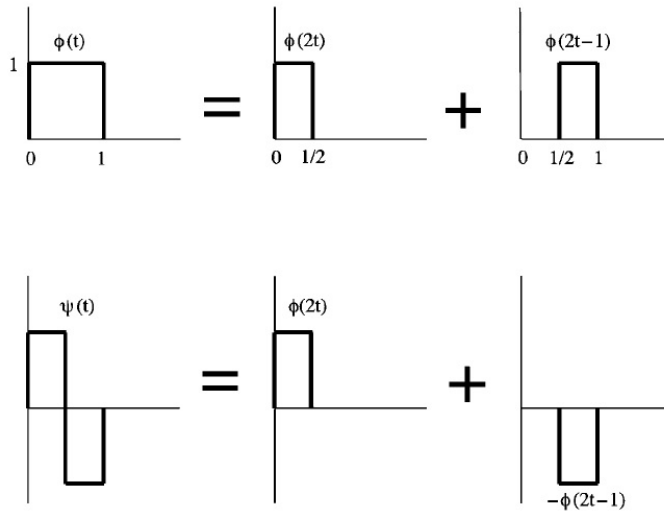


Figura 2.3: Haar (Función de escala y wavelet).

Una función puede aproximarse al grado deseado sumando la función de escala y tantas funciones wavelet de detalle como sea necesario.

Sean  $J \in \mathbb{N}$  fijo y  $f \in V_0$ , entonces

$$f(t) = f_V^1(t) + f_W^1(t),$$

con  $f_V^1 \in V_1$  y  $f_W^1 \in W_1$ . De igual manera

$$f_V^1(t) = f_V^2(t) + f_W^2(t),$$

por lo tanto

$$f(t) = f_V^2(t) + f_W^2(t) + f_W^1(t).$$

Continuando con este procedimiento, se tiene que para la escala  $2^J$ :

$$f(t) = \underbrace{f_V^J(t)}_{\text{Aproximación}} + \underbrace{f_W^J(t) + f_W^{J-1}(t) + \dots + f_W^1(t)}_{\text{Detalles}}. \quad (2.6)$$

La expansión de la ecuación (2.2) la cual solo utiliza funciones wavelets, requiere un número infinito de resoluciones para la representación completa de la función. Por otro lado, la ecuación (2.6) muestra que  $f(t)$  se puede representar como una aproximación en la escala  $2^J$  más la suma de las componentes de detalle a diferentes resoluciones (Ver Figura 2.4 [11]). Esta última forma es claramente la representación más práctica y señala el papel complementario de la base de escala en tales representaciones.

**Ejemplo 2.5.** En la Figura 2.4 se observa la descomposición en cuatro niveles dada por la ecuación (2.6). En rojo la función original, en azul las aproximaciones y en verde los detalles.

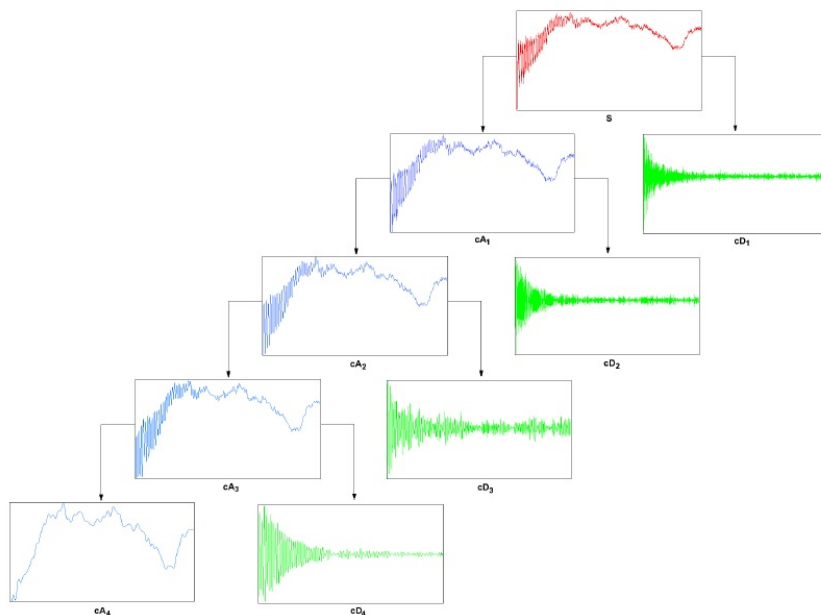


Figura 2.4: Descomposición Wavelet Multiresolución.

En el caso de imágenes, se consideran funciones en el espacio  $L^2(\mathbb{R}^2)$  y se obtiene una descomposición similar a la dada por la ecuación (2.6).

### 2.1.2. La Transformada Wavelet Discreta Bidimensional.

La aproximación de una imagen  $f(t_1, t_2)$  en la resolución  $2^{-j}$  se define como la proyección ortogonal de  $f$  en un espacio  $V_j^2$  que está incluido en  $L^2(\mathbb{R}^2)$ . La definición formal de un MRA  $\{V_j^2\}_{j \in \mathbb{Z}}$  de  $L^2(\mathbb{R}^2)$  es una extensión directa de la Definición 2.5, por lo que se deben cumplir las mismas propiedades.

Consideramos el caso particular de los MRA separables. Sea  $\{V_j\}_{j \in \mathbb{Z}}$  un MRA de  $L^2(\mathbb{R})$ , un MRA bidimensional separable está compuesto por los espacios del producto tensorial:

$$V_j^2 = V_j \otimes V_j, j \in \mathbb{Z}. \quad (2.7)$$

Utilizando las propiedades del producto tensorial se puede probar que si  $\{V_j\}_{j \in \mathbb{Z}}$  es un MRA de  $L^2(\mathbb{R})$ , entonces  $\{V_j^2\}_{j \in \mathbb{Z}}$  es un MRA de  $L^2(\mathbb{R}^2)$ .

**Teorema 2.4** (Teorema A.3, p. 598 [17]). *Sea  $\mathcal{H} = \mathcal{H}_1 \otimes \mathcal{H}_2$  un espacio de Hilbert. Si  $\{e_n^1\}_{n \in \mathbb{N}}$  y  $\{e_n^2\}_{n \in \mathbb{N}}$  son bases Riesz de  $\mathcal{H}_1$  y  $\mathcal{H}_2$  respectivamente, entonces  $\{e_n^1 \otimes e_n^2\}_{(n,m) \in \mathbb{N}^2}$  es una base de Riesz de  $\mathcal{H}$ . Si las dos bases son ortonormales, entonces la base del producto tensorial también es ortonormal.*

Como  $V_j^2 = V_j \otimes V_j$ , el Teorema 2.4 prueba que para  $t = (t_1, t_2)$  y  $k = (k_1, k_2)$

$$\left\{ \phi_{j,k}^2(t) = \frac{1}{2^j} \phi\left(\frac{t_1 - 2^j k_1}{2^j}\right) \phi\left(\frac{t_2 - 2^j k_2}{2^j}\right) \right\}_{k \in \mathbb{N}^2}$$

es una base ortogonal de  $V_j^2$ , con  $\phi^2(t) = \phi(t_1)\phi(t_2)$ .

Luego, se construye una base ortonormal wavelet separable de  $L^2(\mathbb{R}^2)$  con productos separables de una función de escala  $\phi$  y su wavelet  $\psi$  asociada. Sea  $\{V_j^2\}_{j \in \mathbb{Z}}$  un MRA separable definido por (2.7), sea  $W_j^2$  el espacio de detalle igual al complemento ortogonal del espacio de aproximación de resolución inferior  $V_j^2$  en  $V_{j-1}^2$ :

$$V_{j-1}^2 = V_j^2 \oplus W_j^2.$$

Para construir una base wavelet ortonormal de  $L^2(\mathbb{R}^2)$ , el siguiente teorema construye una base wavelet de cada espacio de detalle  $W_j^2$ .

**Teorema 2.5.** *Sea  $\phi$  una función de escala y  $\psi$  la wavelet correspondiente que genera una base ortonormal wavelet de  $L^2(\mathbb{R})$ . Definimos tres wavelets:*

$$\psi^1(t) = \phi(t_1)\psi(t_2), \psi^2(t) = \psi(t_1)\phi(t_2), \psi^3(t) = \psi(t_1)\psi(t_2),$$

y denotamos para  $1 \leq i \leq 3$

$$\psi_{j,k}^i(t) = \frac{1}{2^j} \psi^i\left(\frac{t_1 - 2^j k_1}{2^j}, \frac{t_2 - 2^j k_2}{2^j}\right).$$

La familia wavelet

$$\{\psi_{j,k}^1, \psi_{j,k}^2, \psi_{j,k}^3\}_{k \in \mathbb{Z}^2}$$

es una base ortonormal de  $W_j^2$  y

$$\{\psi_{j,k}^1, \psi_{j,k}^2, \psi_{j,k}^3\}_{(j,k) \in \mathbb{Z}^3}$$

es una base ortonormal de  $L^2(\mathbb{R}^2)$ .

*Demostración.* En primer lugar se tiene que:

$$\begin{aligned} V_{j-1} \otimes V_{j-1} &= V_{j-1}^2 \\ &= V_j^2 \oplus W_j^2 \\ &= (V_j \otimes V_j) \oplus W_j^2. \end{aligned} \quad (2.8)$$

El espacio unidimensional  $V_{j-1}$  también se puede descomponer en:

$$V_{j-1} = V_j \oplus W_j. \quad (2.9)$$

Sustituyendo la ecuación (2.9) en (2.8), la distributividad de  $\oplus$  con respecto a  $\otimes$  demuestra que:

$$W_j^2 = (V_j \otimes W_j) \oplus (W_j \otimes V_j) \oplus (W_j \otimes W_j).$$

Dado que  $\{\phi_{j,k}\}_{k \in \mathbb{Z}}$  y  $\{\psi_{j,k}\}_{k \in \mathbb{Z}}$  son bases ortonormales de  $V_j$  y  $W_j$  respectivamente (Teoremas 2.2 y 2.3), obtenemos que:

$$\{\phi_{j,k_1}(t_1) \psi_{j,k_2}(t_2), \psi_{j,k_1}(t_1) \phi_{j,k_2}(t_2), \psi_{j,k_1}(t_1) \psi_{j,k_2}(t_2)\}_{(k_1,k_2) \in \mathbb{Z}^2}$$

es una base ortonormal de  $W_j^2$ . Como en el caso unidimensional, el espacio total  $L^2(\mathbb{R}^2)$  se puede descomponer como una suma ortogonal de los espacios de detalle en todas las resoluciones:

$$L^2(\mathbb{R}^2) = \bigoplus_{j=-\infty}^{+\infty} W_j^2.$$

Por lo tanto,

$$\{\phi_{j,k_1}(t_1) \psi_{j,k_2}(t_2), \psi_{j,k_1}(t_1) \phi_{j,k_2}(t_2), \psi_{j,k_1}(t_1) \psi_{j,k_2}(t_2)\}_{(j,k_1,k_2) \in \mathbb{Z}^3}$$

es una base ortonormal de  $L^2(\mathbb{R}^2)$ .  $\square$

Al igual que en la ecuación (2.6), una imagen se puede descomponer en términos de la 2-D DWT. En cada nivel de descomposición, la imagen descompone en cuatro imágenes y el tamaño de cada una de ellas es una cuarta parte del tamaño de la original, como se muestra en la Figura 2.5 [19].

Estas cuatro imágenes muestran los detalles y una aproximación de la imagen original. De acuerdo a las propiedades de los subespacios  $\{V_j^2\}_{j \in \mathbb{Z}}$  se tiene que las escalas de menor resolución pierden puntos de interés y las escalas de alta resolución a menudo son ruidosas.

Luego de aplicar la 2D-DWT, por lo general se obtienen imágenes con una gran cantidad de píxeles, es por ello que es importante aplicar una técnica como el PCA para reducir la dimensionalidad de las imágenes.

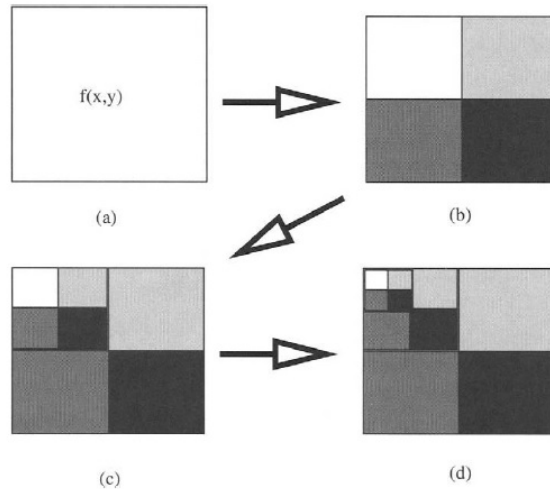


Figura 2.5: Transformada wavelet discreta 2D: (a) imagen original, (b) primer, (c) segundo y (d) tercer nivel de descomposición.

## 2.2. Análisis de Componentes Principales.

Trabajar directamente con datos de alta dimensión, como imágenes, presenta algunas dificultades: es difícil de analizar, la interpretación es difícil, la visualización es casi imposible. Sin embargo, los datos de alta dimensión a menudo tienen propiedades que podemos explotar. Dado un conjunto de datos, en el PCA estamos interesados en encontrar proyecciones que sean lo más similares posible a los datos originales, pero que tengan una dimensionalidad intrínseca significativamente menor [10], es decir, encontrar una representación comprimida de baja dimensión de nuestros datos.

El PCA es una técnica utilizada para describir un conjunto de datos en términos de nuevas variables (llamadas componentes) no correlacionadas. Esta es una propiedad altamente deseable, ya que se elimina la redundancia en la información de datos.

Consideremos un conjunto de datos i.i.d.  $\mathcal{X} = \{x_1, \dots, x_N\}$ , con media 0 que posee la matriz de covarianza:

$$S = \frac{1}{N} \sum_{n=1}^N x_n x_n^\top,$$

donde cada  $x_n$  con  $1 \leq n \leq N$ , es un vector aleatorio de dimensión  $D$ , el cual representa al  $n$ -ésimo individuo de  $\mathcal{X}$  sobre el cual se observan  $D$  variables.

Supongamos que existe una representación comprimida de baja dimensión

$$z_n = B^\top x_n \in \mathbb{R}^M,$$

de  $x_n$ , donde  $B = [b_1, \dots, b_M] \in \mathbb{R}^{D \times M}$  es la matriz de proyección ortogonal, es decir, que  $b_i^\top b_j = 0$  si y solo si  $i \neq j$  y  $b_i^\top b_i = 1$ . El objetivo es encontrar la matriz  $B$  que retenga la mayor cantidad posible de información al proyectar los datos en el subespacio generado por las columnas de  $B$ .

Consideramos la relación lineal entre los datos originales  $x$  y su código de baja dimensión  $z$  de modo que  $z = B^\top x$ . Para maximizar la varianza del código de baja dimensión se utiliza un enfoque secuencial, se comienza buscando el vector  $b_1$  de manera que maximice la varianza de los datos proyectados, por lo tanto nuestro objetivo es maximizar la varianza de la primera coordenada  $z_1$  de  $z$ :

$$\begin{aligned} F_1 &= F(z_1) \\ &= \frac{1}{N} \sum_{n=1}^N z_{1n}^2. \end{aligned}$$

La primera coordenada de  $z_n$  es  $z_{1n} = b_1^\top x_n$ , es decir, la coordenada de la proyección ortogonal de  $x_n$  en el subespacio unidimensional generado por  $b_1$ , por lo tanto:

$$\begin{aligned} F_1 &= \frac{1}{N} \sum_{n=1}^N (b_1^\top x_n)^2 \\ &= \frac{1}{N} \sum_{n=1}^N b_1^\top x_n x_n^\top b_1 \\ &= b_1^\top \left( \frac{1}{N} \sum_{n=1}^N x_n x_n^\top \right) b_1 \\ &= b_1^\top S b_1. \end{aligned}$$

Restringimos todas las soluciones a  $\|b_1\|^2 = 1$ , donde  $\|\cdot\|$  denota la norma euclidiana. Así, se tiene el problema de optimización restringido:

$$\max_{b_1} b_1^\top S b_1$$

$$\text{sujeto a: } \|b_1\|^2 = 1,$$

el cual encuentra el vector  $b_1$  que apunta en la dirección de varianza máxima. Usando los multiplicadores de Lagrange para resolver este problema de optimización, tenemos que:

$$L(b_1, \lambda_1) = b_1^\top S b_1 - \lambda_1 (b_1^\top b_1 - 1), \lambda_1 \in \mathbb{R}.$$

Derivamos con respecto a  $b_1$  y a  $\lambda_1$  respectivamente.

- $\frac{\partial}{\partial b_1}(L(b_1, \lambda_1)) = 2Sb_1 - 2\lambda_1 b_1.$
- $\frac{\partial}{\partial \lambda_1}(L(b_1, \lambda_1)) = 1 - b_1^\top b_1.$

Igualando ambas ecuaciones a cero se tiene que:

- $Sb_1 = \lambda_1 b_1.$
- $b_1^\top b_1 = 1.$

Vemos que  $b_1$  es un autovector de la matriz de covarianza de los datos  $S$  y el multiplicador de Lagrange  $\lambda_1$  desempeña el papel del autovalor correspondiente. Así, la varianza de  $z_1$  nos quedaría:

$$F_1 = b_1^\top S b_1 = b_1^\top \lambda_1 b_1 = \lambda_1,$$

es decir, la varianza de los datos proyectados en un subespacio unidimensional es igual al autovalor que esta asociado con el vector base  $b_1$  que genera dicho subespacio.  $b_1$  es llamada la primera componente principal.

En general, para encontrar un subespacio  $M$ -dimensional de  $\mathbb{R}^D$  que retenga la mayor cantidad de información, el PCA nos dice que escojamos las columnas de la matriz  $B$  como los autovectores de la matriz de covarianza de datos  $S$  que están asociados con los  $M$  autovalores más grandes.

La cantidad máxima de varianza que el PCA puede capturar con las primeras  $M$  componentes principales es:

$$F_M = \sum_{i=1}^M \lambda_i.$$

La varianza perdida por la compresión de los datos a través del PCA es:

$$\begin{aligned} P_M &= \sum_{j=M+1}^D \lambda_j \\ &= F_D - F_M. \end{aligned}$$

El PCA es una técnica de aprendizaje no supervisado y como tal, no incluye información de la etiqueta de los datos. Para utilizar la información de la pertenencia de los datos a una clase particular se utilizan técnicas como el Discriminante Lineal de Fisher.

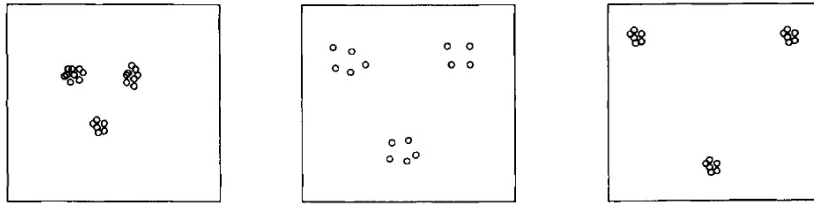


Figura 2.6: Clases con: (a) pequeña variación dentro de las clase y pequeñas distancias entre las clases, (b) gran variación dentro de las clase y pequeñas distancias entre las clases, y (c) pequeña variación dentro de las clases y grandes distancias entre las clases.

## 2.3. Discriminante Lineal de Fisher.

Este método se basa en información relacionada con la forma en que los vectores de muestra están dispersos en el espacio de características. El DLF se basa en lograr una máxima separabilidad entre las clases (Ver Figura 2.6 [1]).

El enfoque de Fisher determina a partir de  $P$  características dadas, las  $p$  mejores características basado en una  $p \times P$  matriz de transformación lineal. Es decir, supongamos que se nos da un vector de características  $P$ -dimensional y obtenemos un nuevo vector de características  $p$ -dimensional  $x$  dado por:

$$x = Ey.$$

A partir de  $q$  grupos donde se asignan a una serie de objetos y de  $P$  variables medidas sobre ellos  $(x_1, \dots, x_P)$  se puede descomponer la variabilidad total de la muestra en variabilidad dentro de los grupos y entre los grupos. Partimos de

$$Cov(x_j, x_{j'}) = \frac{1}{n} \sum_{i=1}^n (x_{ij} - \bar{x}_j)(x_{ij'} - \bar{x}_{j'}), \quad (2.10)$$

en donde  $j, j' \in \{1, 2, \dots, P\}$ ,  $n$  representa el cardinal del conjunto de datos,  $x_{ij}$  es el valor de la variable  $x_j$  en la  $i$ -ésima observación y  $\bar{x}_j$  es la media de la variable  $x_j$ . Se puede considerar la media de la variable  $x_j$  en cada uno de los grupos  $I_1, \dots, I_q$ , es decir,

$$\bar{x}_{kj} = \frac{1}{n_k} \sum_{i \in I_k} x_{ij},$$

donde  $k \in \{1, 2, \dots, q\}$  y  $n_k$  es el cardinal del  $k$ -ésimo grupo. De este modo, la media total de la variable  $x_j$  se puede expresar como función de las medias

dentro de cada grupo. Así,

$$\sum_{i \in I_k} x_{ij} = n_k \bar{x}_{kj},$$

entonces,

$$\begin{aligned} \bar{x}_j &= \frac{1}{n} \sum_{i=1}^n x_{ij} \\ &= \frac{1}{n} \sum_{k=1}^q \sum_{i \in I_k} x_{ij} \\ &= \frac{1}{n} \sum_{k=1}^q n_k \bar{x}_{kj} \\ &= \sum_{k=1}^q \frac{n_k}{n} \bar{x}_{kj}. \end{aligned}$$

La ecuación (2.10) se puede escribir de la siguiente manera:

$$Cov(x_j, x_{j'}) = \sum_{k=1}^q \sum_{i \in I_k} (x_{ij} - \bar{x}_j)(x_{ij'} - \bar{x}_{j'}).$$

Si cada uno de los términos se sustituye por:

$$(x_{ij} - \bar{x}_j) = (x_{ij} - \bar{x}_{kj}) + (\bar{x}_{kj} - \bar{x}_j),$$

$$(x_{ij'} - \bar{x}_{j'}) = (x_{ij'} - \bar{x}_{kj'}) + (\bar{x}_{kj'} - \bar{x}_{j'}),$$

al simplificar se obtiene:

$$Cov(x_j, x_{j'}) = \underbrace{\frac{1}{n} \sum_{k=1}^q \sum_{i \in I_k} (x_{ij} - \bar{x}_{kj})(x_{ij'} - \bar{x}_{kj'})}_{:=S_I} + \underbrace{\sum_{k=1}^q \frac{n_k}{n} (\bar{x}_{kj} - \bar{x}_j)(\bar{x}_{kj'} - \bar{x}_{j'})}_{:=S_O}.$$

$S_O$  es la “matriz de dispersión entre clases” y  $S_I$  es la “matriz de dispersión dentro de las clases”. Es decir, la covarianza total es igual a la covarianza dentro de los grupos más la covarianza entre los grupos.

Para lograr la máxima separabilidad entre las clases, Fisher considera maximizar el siguiente objetivo:

$$Q(e) = \frac{e^\top S_O e}{e^\top S_I e}.$$

Dado que  $Q$  es una función homogénea, es decir:

$$Q(\alpha e) = Q(e), \forall \alpha \in \mathbb{R},$$

podemos elegir  $e$  de tal manera que:  $e^\top S_I e = 1$ . Así, el problema se puede transformar en el problema de optimización restringido:

$$\begin{aligned} \max_e e^\top S_O e \\ \text{sujeto a: } e^\top S_I e = 1, \end{aligned}$$

produciendo el Lagrangiano

$$L(e, \lambda) = e^\top S_O e - \lambda(e^\top S_I e - 1), \lambda \in \mathbb{R}.$$

Para resolver este problema, derivamos con respecto a  $w$  y obtenemos:

$$\blacksquare \frac{\partial}{\partial e}(L(e, \lambda)) = 2S_O e - 2\lambda S_I e.$$

Igualando esta ecuación a cero se tiene que:

$$\blacksquare S_O e = \lambda S_I e.$$

Lo que implica que:

$$S_I^{-1} S_O e = \lambda e,$$

el cual es llamado un problema de autovector generalizado. Así, se tiene que:

$$\begin{aligned} \max_e e^\top S_O e &= e^\top (\lambda S_I e) \\ &= \lambda (e^\top S_I e) \\ &= \lambda. \end{aligned}$$

Por lo tanto, la matriz  $E$  la conforman los autovectores de la matriz  $S_I^{-1} S_O$  asociados a los autovalores significativos más grandes. Al transformar los datos a través de la matriz  $E$ , se logra una mayor separabilidad entre las clases y se reduce la dimensionalidad de los datos.

En este caso, no estamos utilizando el DLF para tareas de clasificación, solo queremos encontrar una transformación lineal que nos permita lograr una máxima separabilidad entre las clases. Para tareas de clasificación se pueden utilizar las redes neuronales artificiales entre otras.

## 2.4. Máquina de Aprendizaje Extremo.

Las redes neuronales artificiales son modelos matemáticos que intentan reproducir el comportamiento del cerebro humano. El principal objetivo de este modelo es la construcción de sistemas capaces de presentar un cierto comportamiento inteligente. Esto implica la capacidad de aprender a realizar una determinada tarea. Una neurona artificial es una unidad de procesamiento de

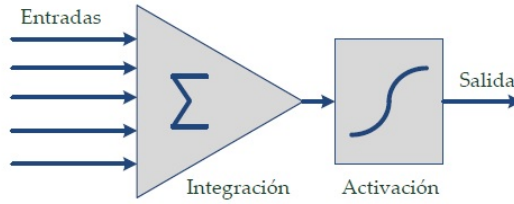


Figura 2.7: Modelo estándar de una neurona artificial.

información que es fundamental para el funcionamiento de una red neuronal (Ver Figura 2.7 [3]).

Habitualmente, las redes neuronales artificiales están formadas por conjuntos de neuronas que se agrupan en capas, de forma que todas las neuronas de una misma capa compartan ciertas características. Estas capas se dividen en capas de entrada, ocultas y de salida. Normalmente, todas las neuronas de una capa reciben señales de entrada desde otra capa anterior y envían señales de salida a una capa posterior. A estas conexiones se las denomina conexiones hacia adelante o feedforward.

Una máquina de Aprendizaje Extremo (ELM) es un algoritmo que inicialmente entrena una red neuronal feedforward con una única capa oculta (SLFN) [6] [7]. Esta configuración se observa en la Figura (2.8) [9].

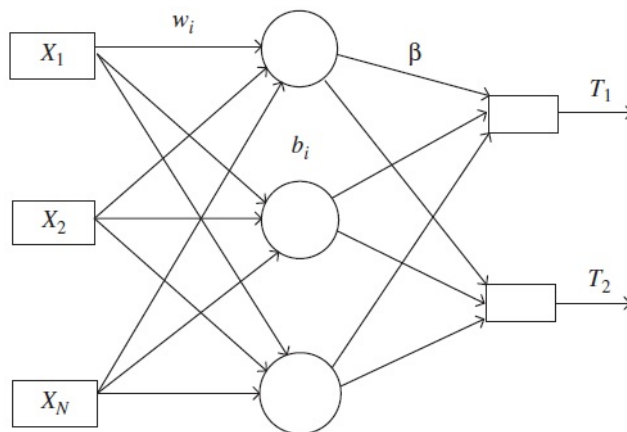


Figura 2.8: Modelo de una SLFN.

Los pesos y sesgos de entrada ( $w_i$  y  $b_i$ ) se generan aleatoriamente, mientras que los pesos de salida ( $\beta$ ) se pueden determinar automáticamente. En comparación con los métodos tradicionales de entrenamiento de una red como la back-propagation, la velocidad del ELM es más rápida.

Para  $N$  muestras distintas arbitrarias  $\{(x_1, t_1), \dots, (x_i, t_i), \dots, (x_N, t_N)\}$ , donde  $x_i = [x_{i1}, x_{i2}, \dots, x_{in}]^\top \in \mathbb{R}^n$  y  $t_i = [t_{i1}, t_{i2}, \dots, t_{im}]^\top \in \mathbb{R}^m$ , una SLFN estándar con  $L$  nodos (neuronas) ocultos y función de activación  $g(x)$  se modela matemáticamente como:

$$\sum_{i=1}^L \beta_i g_i(x_j) = \sum_{i=1}^L \beta_i g(w_i \cdot x_j + b_i),$$

donde  $j = 1, 2, \dots, N$ ,  $w_i = [w_{i1}, w_{i2}, \dots, w_{in}]^\top$  es el vector de pesos que conecta el  $i$ -ésimo nodo oculto y los nodos de entrada,  $\beta_i = [\beta_{i1}, \beta_{i2}, \dots, \beta_{im}]^\top$  es el vector de pesos que conecta el  $i$ -ésimo nodo oculto y los nodos de salida, y  $b_i$  es el sesgo del  $i$ -ésimo nodo oculto.

Considerando la norma euclidiana, los parámetros adecuados se obtienen minimizando la función de error

$\|H\beta - T\|$ , donde:

$$H = \begin{bmatrix} g(w_1 \cdot x_1 + b_1) & \dots & g(w_L \cdot x_1 + b_L) \\ \vdots & \ddots & \vdots \\ g(w_1 \cdot x_N + b_1) & \dots & g(w_L \cdot x_N + b_L) \end{bmatrix}_{N \times L},$$

$$\beta = \begin{bmatrix} \beta_1^\top \\ \vdots \\ \beta_L^\top \end{bmatrix}_{L \times m} \quad y \quad T = \begin{bmatrix} t_1^\top \\ \vdots \\ t_N^\top \end{bmatrix}_{N \times m}.$$

A la matriz  $H$  se le denomina matriz de salida de la capa oculta de la red neuronal. Si la función de activación  $g$  es infinitamente diferenciable, se prueba en el Teorema 2.6 que el número requerido de nodos ocultos es  $L \leq N$ .

**Teorema 2.6.** *Dada una SLFN estándar con  $N$  nodos ocultos cuya función de activación  $g : \mathbb{R} \rightarrow \mathbb{R}$  es infinitamente diferenciable en cualquier intervalo, para  $N$  muestras distintas arbitrarias  $\{(x_i, t_i)\}_{i=1}^N$ , con  $x_i \in \mathbb{R}^n$  y  $t_i \in \mathbb{R}^m$ ,  $\{(w_i, b_i)\}_{i=1}^N$  generados aleatoriamente con cualquier distribución de probabilidad continua, respectivamente, se tiene que con probabilidad uno, la matriz de salida de capa oculta  $H$  de la SLFN es invertible y  $\|H\beta - T\| = 0$ .*

*Demostración.* Consideremos el vector:

$$\begin{aligned} v_{b_i} &= [g_i(x_1), \dots, g_i(x_N)]^\top \\ &= [g(w_i \cdot x_1 + b_i), \dots, g(w_i \cdot x_N + b_i)]^\top, \end{aligned}$$

la  $i$ -ésima columna de  $H$ , en el espacio euclidiano  $\mathbb{R}^N$ , donde  $b_i \in (a, b)$  y  $(a, b)$  es cualquier intervalo de  $\mathbb{R}$ . Lo que sigue por demostrar es que  $v_{b_i}$  no pertenece a ningún subespacio cuya dimensión sea menor que  $N$ .

Dado que  $w_i$  se genera aleatoriamente basado en una distribución de probabilidad continua, podemos suponer que:

$$w_i \cdot x_k \neq w_i \cdot x_{k'}, \forall k \neq k'.$$

Supongamos que  $v_{b_i}$  pertenece a un subespacio de dimensión  $N - 1$ . Entonces existe un vector  $u$  que es ortogonal a este subespacio, es decir:

$$\begin{aligned} \langle u, v_{b_i} - v_a \rangle &= u_1 \cdot g(b_i + d_1) + u_2 \cdot g(b_i + d_2) \\ &\quad + \dots + u_N \cdot g(b_i + d_N) - z = 0, \end{aligned} \quad (2.11)$$

donde  $d_k = w_i \cdot x_k$ ,  $k = 1, \dots, N$  y  $z = u \cdot v_a$ ,  $\forall b_i \in (a, b)$ . Supongamos que  $u_N \neq 0$ , la ecuación (2.11) se puede escribir como:

$$g(b_i + d_N) = - \sum_{p=1}^{N-1} \gamma_p g(b_i + d_p) + z/u_N,$$

donde  $\gamma_p = u_p/u_N$ ,  $p = 1, \dots, N - 1$ . Dado que  $g$  es infinitamente diferenciable en cualquier intervalo, tenemos que:

$$g^{(l)}(b_i + d_N) = - \sum_{p=1}^{N-1} \gamma_p g^{(l)}(b_i + d_p), \quad (2.12)$$

$l = 1, 2, \dots, N, N + 1, \dots$ , donde  $g^{(l)}$  es la  $l$ -ésima derivada de la función  $g$ . Sin embargo, sólo hay  $(N - 1)$  coeficientes libres:  $\gamma_1, \dots, \gamma_N$  para las más de  $(N - 1)$  ecuaciones lineales en (2.12), esto es contradictorio. Así, el vector  $v_{b_i}$  no pertenece a ningún subespacio cuya dimensión sea menor que  $N$ .

Por lo tanto, de cualquier intervalo  $(a, b)$  es posible elegir aleatoriamente  $N$  valores de sesgo  $b_1, \dots, b_N$  para los  $N$  nodos ocultos de tal manera que los vectores correspondientes  $v_{b_1}, v_{b_2}, \dots, v_{b_N}$  generan a  $\mathbb{R}^N$ . Esto significa que para cualesquiera vectores de peso  $w_i$  y valores de sesgo  $b_i$  elegidos de cualquier subconjunto de  $\mathbb{R}^n$  y  $\mathbb{R}$ , respectivamente, de acuerdo con cualquier distribución de probabilidad continua, entonces con probabilidad uno, los vectores columna de  $H$  pueden hacerse de rango completo.  $\square$

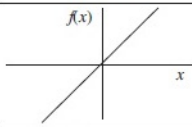
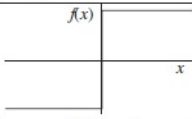
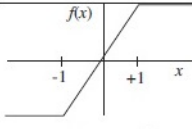
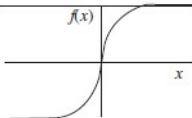
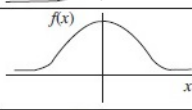
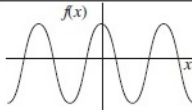
	Función	Rango	Gráfica
Identidad	$y = x$	$[-\infty, +\infty]$	
Escalón	$y = \text{sign}(x)$ $y = H(x)$	$\{-1, +1\}$ $\{0, +1\}$	
Lineal a tramos	$y = \begin{cases} -1, & \text{si } x < -l \\ x, & \text{si } -l \leq x \leq +l \\ +1, & \text{si } x > +l \end{cases}$	$[-1, +1]$	
Sigmoidea	$y = \frac{1}{1 + e^x}$ $y = \text{tgh}(x)$	$[0, +1]$ $[-1, +1]$	
Gaussiana	$y = Ae^{-Bx^2}$	$[0, +1]$	
Sinusoidal	$y = A \text{sen}(\omega x + \varphi)$	$[-1, +1]$	

Figura 2.9: Principales funciones de activación.

Dichas funciones de activación incluyen las funciones sigmoideas, así como las funciones de base radial, seno, coseno, exponencial y muchas otras funciones (Ver Figura 2.9 [8]).

**Teorema 2.7.** *Sea  $g : \mathbb{R} \rightarrow \mathbb{R}$  una función de activación infinitamente diferenciable en cualquier intervalo, para cualquier  $\epsilon > 0$ , existe  $L \leq N$  tal que para  $N$  muestras distintas arbitrarias  $\{(x_i, t_i)\}_{i=1}^N$ , con  $x_i \in \mathbb{R}^n$  y  $t_i \in \mathbb{R}^m$ ,  $\{(w_i, b_i)\}_{i=1}^L$  generados aleatoriamente con cualquier distribución de probabilidad continua, respectivamente, se tiene que con probabilidad uno:*

$$\|H_{N \times L} \beta_{L \times m} - T_{N \times m}\| < \epsilon.$$

*Demostración.* La validez del Teorema es clara, de lo contrario, simplemente se podría elegir  $L = N$ , lo cual hace que:  $\|H_{N \times L} \beta_{L \times m} - T_{N \times m}\| < \epsilon$ , de acuerdo con el Teorema 2.6. □

Con base en los Teoremas 2.6 y 2.7, el algoritmo ELM es un método extremadamente simple y eficiente para entrenar SLFNs. Tradicionalmente, para entrenar una SLFN, se busca encontrar específicos  $\hat{w}_i, \hat{b}_i, \hat{\beta}$  ( $i = 1, \dots, L$ ) tales que:

$$\|H(\hat{w}_1, \dots, \hat{w}_L, \hat{b}_1, \dots, \hat{b}_L)\hat{\beta} - T\| = \min_{w_i, b_i, \beta} \|H(w_1, \dots, w_L, b_1, \dots, b_L)\beta - T\|. \quad (2.13)$$

A diferencia de la comprensión más común de que todos los parámetros de una SLFN necesitan ser ajustados, los pesos de entrada  $w_i$  y los sesgos de capa oculta  $b_i$  no están necesariamente ajustados y la matriz de salida de la capa oculta  $H$  puede permanecer sin cambios una vez que se hayan asignado valores aleatorios a estos parámetros al comienzo del aprendizaje.

Para los pesos de entrada fijos  $w_i$  y los sesgos de capa ocultos  $b_i$ , vistos desde la ecuación (2.13), entrenar una SLFN es simplemente equivalente a encontrar una solución de mínimos cuadrados  $\hat{\beta}$  del sistema lineal  $H\beta = T$ :

$$\|H(w_1, \dots, w_L, b_1, \dots, b_L)\hat{\beta} - T\| = \min_{\beta} \|H(w_1, \dots, w_L, b_1, \dots, b_L)\beta - T\|.$$

Si el número de nodos ocultos  $L$  es igual al número  $N$  de muestras de entrenamiento distintas, la matriz  $H$  es cuadrada e invertible cuando los vectores de peso de entrada  $w_i$  y los sesgos ocultos  $b_i$  se eligen al azar, y las SLFNs pueden aproximar estas muestras de entrenamiento con error cero (Teorema 2.6).

Sin embargo, en la mayoría de los casos, el número de nodos ocultos es mucho menor que el número de muestras de entrenamiento distintas,  $H$  es una matriz rectangular y puede que no existan  $w_i, b_i, \beta_i$  ( $i = 1, \dots, L$ ) tales que  $H\beta = T$ .

La solución de un sistema lineal general  $Ax = y$ , donde  $A$  puede ser singular e incluso puede no ser cuadrada, puede encontrarse mediante el uso de la inversa generalizada de Moore-Penrose.

**Definición 2.6** (Definición de Penrose). *Sea  $A \in M_{m \times n}(\mathbb{R})$ . Se llama inversa generalizada de Moore-Penrose de  $A$  a la única matriz  $Y \in M_{m \times n}(\mathbb{R})$  que satisface:*

- $AYA = A$ ,
- $YAY = Y$ ,
- $(AY)^\top = AY$ ,
- $(YA)^\top = YA$ ,

La matriz  $Y$  se denota por  $A^\dagger$ .

**Teorema 2.8.** *Sean  $A \in M_{m \times n}(\mathbb{R})$  y  $b \in M_{m \times 1}(\mathbb{R})$ , entonces la única solución de mínimos cuadrados de norma mínima del sistema  $Ax = b$  es  $A^\dagger b$ .*

Una demostración del Teorema 2.8 puede ser consultada en [18] (Teorema 3.1, p. 33).

**Observación 2.5.** *Se pueden utilizar diferentes métodos para calcular la inversa generalizada de Moore-Penrose de una matriz: método de proyección ortogonal, método de ortogonalización, método iterativo y descomposición de valor singular.*

De acuerdo a el Teorema 2.8 la solución de mínimos cuadrados de norma mínima del sistema lineal  $H\beta = T$  es:

$$\hat{\beta} = H^\dagger T, \quad (2.14)$$

donde  $H^\dagger$  es la inversa generalizada de Moore-Penrose de la matriz  $H$ . La solución presentada en la ecuación (2.14) tiene las siguientes propiedades importantes:

- Error mínimo de entrenamiento. La solución especial  $\hat{\beta} = H^\dagger T$  alcanza el error de entrenamiento más pequeño:

$$\|H\hat{\beta} - T\| = \|HH^\dagger T - T\| = \min_{\beta} \|H\beta - T\|.$$

- La norma más pequeña de los pesos. Además, la solución especial  $\hat{\beta} = H^\dagger T$  tiene la norma más pequeña entre todas las soluciones de mínimos cuadrados de  $H\beta = T$ :

$$\|\hat{\beta}\| = \|H^\dagger T\| \leq \|\beta\|, \forall \beta \in \{\beta : \|H\beta - T\| \leq \|Hz - T\|, \forall z \in \mathbb{R}^{L \times N}\}.$$

- La solución de mínimos cuadrados de norma mínima del sistema  $H\beta = T$  es única, la cual es  $\hat{\beta} = H^\dagger T$ .

Por lo tanto, un método de aprendizaje simple para SLFNs llamado máquina de aprendizaje extremo se puede resumir de la siguiente manera:

Algoritmo ELM: Dado un conjunto de entrenamiento  $\mathcal{X} = \{(x_i, t_i) | x_i \in \mathbb{R}^n, t_i \in \mathbb{R}^m, i = 1, 2, \dots, N\}$ , una función de activación  $g(x)$  y el número de nodos ocultos  $L$ ,

1. Asignar aleatoriamente los pesos de entrada  $w_i$  y los sesgos  $b_i$ ,  $i = 1, 2, \dots, N$ .
2. Calcular la matriz de salida de la capa oculta  $H$ .
3. Calcular el peso de salida  $\beta$ ,  $\beta = H^\dagger T$ , donde  $T = [t_1, t_2, \dots, t_N]^\top$ .



## Capítulo 3

# Metodología Propuesta.

En este Capítulo se describe la metodología propuesta, así como el objetivo de cada una de las técnicas aplicadas. Luego se presentan los resultados obtenidos al implementar esta metodología sobre la base de datos con la cual se cuenta y por último las conclusiones a las cuales se ha llegado por medio de este trabajo de investigación.

Dado un conjunto de entrenamiento, formado por las imágenes de resonancia magnética, de las cuales se conoce previamente la clase a la cual pertenecen, la idea es aplicar todas las técnicas ya mencionadas de manera secuencial, con lo cual se espera diagnosticar de manera temprana la AD.

La descripción detallada de esta metodología se muestra en la Figura 3.1 .

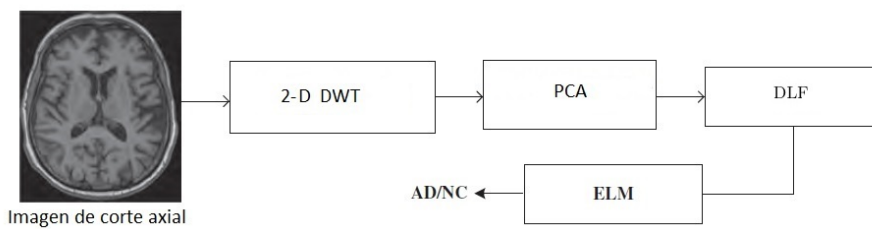


Figura 3.1: Metodología para clasificar MRI's.

Cada una de las técnicas aplicadas tiene un objetivo particular. En primer lugar la 2-D DWT (Sección 2.1.2) permite descomponer la imagen original en cuatro imágenes que proporcionan algunos detalles que no son perceptibles por el ojo humano. La matriz de aproximación de la imagen es la que nos permitirá extraer las características relevantes que se encuentran en la imagen original.

Luego el PCA (Sección 2.2) nos ayudará en primer lugar a reducir la dimensionalidad de nuestras imágenes y a la vez a eliminar la posible correlación que pudiese existir entre los píxeles, todo esto preservando la mayor cantidad de información posible y eliminando la información redundante.

Con el DLF (Sección 2.3) nuevamente se obtiene una reducción de la dimensionalidad y al mismo tiempo se logra una mayor separabilidad de las clases una vez que se proyectan los datos. Esto será de mucha ayuda para la clasificación que se llevará a cabo con una red neuronal artificial SLFN entrenada con la ELM (Sección 2.4). Podemos ver que cada una de las técnicas juega un papel importante, ya que por medio de ellas se seleccionan las variables determinantes que permiten una excelente discriminación entre las clases.

### 3.1. Resultados Experimentales y Análisis.

En esta investigación, el método propuesto es implementado sobre el conjunto de imágenes de resonancia magnética descargadas del archivo de imágenes y datos (IDA) (<https://ida.loni.usc.edu/login.jsp>), el cual proporciona herramientas y recursos para identificar, integrar, buscar, visualizar y compartir una amplia gama de datos de neurociencia, lo que ayuda a facilitar la colaboración entre científicos de todo el mundo.

Las imágenes se obtuvieron de la base de datos de la Iniciativa de Neuroimagen de la enfermedad de Alzheimer (ADNI), la cual es un estudio longitudinal multicéntrico diseñado para desarrollar biomarcadores clínicos, de imagen, genéticos y bioquímicos para la detección temprana y el seguimiento de la enfermedad de Alzheimer. El conjunto de datos de ADNI consta de más de 6000 sujetos de entre 18 y 96 años.

A través del programa ONIS Viewer, un visor de imágenes médicas obtuvimos los cortes 2D, lo cual nos proporciona imágenes de 256x256 píxeles con ponderación T2 axial, en formato PNG (Figura 3.2). Se utilizaron un total de 169 sujetos del conjunto de datos ADNI, 80 AD y 89 CN, cuyas características se observan en la Tabla 3.1.

Tabla 3.1: Descripción de la base de datos.

	AD	CN
Numero de Sujetos	80	89
Hombres	41	46
Mujeres	39	43
Rango de edad	56-89	70-90
Edad Promedio	75.13	76.62

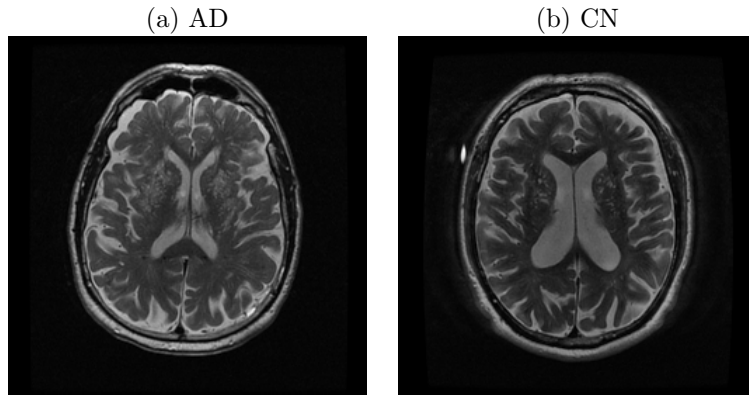


Figura 3.2: Corte axial 2D de una imagen de Resonancia Magnética.

Todos los programas se ejecutan en MATLAB 2017b, el cual se instaló en el sistema de CPU AMD RYZEN 5. A continuación se enumeran los pasos llevados a cabo en el análisis.

1. **Elección de una wavelet madre:** la primera técnica que se aplicará es la 2-D DWT, para ello debemos elegir una wavelet madre. En el caso de las imágenes de resonancia magnética del cerebro, los valores de intensidad de los píxeles varían suavemente, lo que no puede representarse de manera muy eficiente por una wavelet de Haar [15]. La wavelet Daubechies-4 (DAUB4) tiene la ventaja de una mejor resolución para señales que cambian suavemente en comparación con la wavelet de Haar [15]. Por lo tanto, hemos elegido la wavelet Daubechies-4, la cual ofrece una excelente precisión de clasificación.
2. **Nivel de descomposición:** con respecto al nivel de descomposición, lo que se busca es reducir el tamaño de los vectores a los cuales se les aplicará el PCA. El número de coeficientes de aproximación de acuerdo al nivel de descomposición se observa en la Tabla 3.2.

Observemos que a partir del tercer nivel de descomposición se reduce en gran medida el tamaño del vector de coeficientes. Por lo tanto, elegiremos  $N \in \{3, 4, 5\}$  de acuerdo a los resultados de clasificación obtenidos. Sin embargo, cuando trabajamos con  $N = 3$  y  $N = 4$ , los resultados obtenidos luego del PCA no son correctos, debido a que la matriz de covarianza de los datos no es definida positiva. Este problema surge ya que cuando  $N = 3$  y  $N = 4$ , el número de coeficientes es mucho mayor que la cantidad de datos que se tiene, lo que nos lleva a una pobre estimación de la matriz de covarianzas. Esto no ocurre cuando  $N = 5$ , por lo tanto, se elige el quinto nivel de descomposición (Ver Figura 3.3).

Tabla 3.2: Número de coeficientes de acuerdo al nivel de descomposición de la 2-D DWT.

Nivel de descomposición	Tamaño de la imagen	Número de coeficientes
$N = 1$	131x131	17161
$N = 2$	69x69	4761
$N = 3$	38x38	1444
$N = 4$	22x22	484
$N = 5$	14x14	196

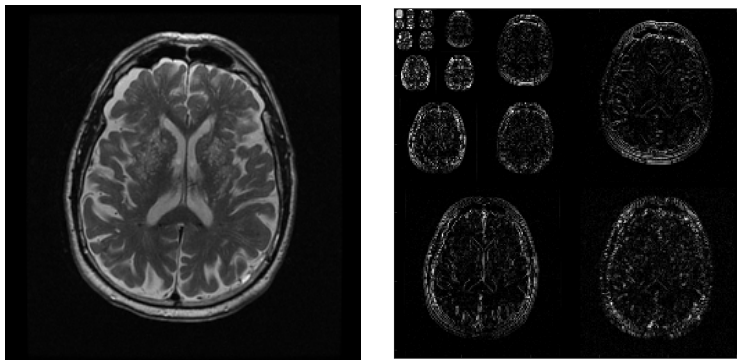


Figura 3.3: Imagen original y sus coeficientes wavelet en la descomposición de cinco niveles.

Una vez elegidos tanto la wavelet madre como el nivel de descomposición a utilizar, se aplica a cada una de las imágenes la 2D-DWT. Para ello, en primer lugar se lee la imagen a través de la función **imread** de Matlab, lo que da como resultado un arreglo de tamaño  $256 \times 256 \times 3$  de tipo `uint8`, el cual corresponde a una imagen en formato RGB. Posteriormente, este arreglo se utiliza como la entrada del Algoritmo 3.1, el cual utiliza la función **rgbgray** de Matlab para convertir la imagen RGB en una imagen de intensidad en escala de grises, lo que da como resultado un arreglo de tamaño  $256 \times 256$  de tipo `uint8`. Por medio de la función **double** este arreglo se convierte en uno de tipo `double`, el cual es el tipo de datos numéricos predeterminado de Matlab.

Luego, por medio de las funciones **wavedec2** y **appcoef2** se obtienen los coeficientes de aproximación de la imagen en el quinto nivel de descomposición aplicando la 2D-DWT y utilizando la Daubechies-4 como wavelet madre, lo que da como resultado un arreglo de tamaño  $14 \times 14$  de tipo `double`. Este arreglo es concatenado por filas para de esta manera

obtener un vector perteneciente a  $\mathbb{R}^{196}$ . Este procedimiento es aplicado a cada una de las 169 imágenes, formando así una matriz de datos de tamaño 169x196, donde las filas representan a los sujetos y las columnas a las variables.

---

**Algoritmo 3.1:** Código para aplicar la 2D-DWT.

---

```

1 function[v]=img(I) ;
2 x=rgb2gray(I) ;
3 y=double(x) ;
4 [C, S]=wavedec2(y, 5, 'db4') ;
5 A=appcoef2(C,S,'db4', 5) ;
6 [m , -]=size(A) ;
7 B=A(1, : ) ;
8 for j=2:m ;
9 B=[B A(j, : )];
10 end ;
11 v=B;
12 end;

```

---

3. **Eliminación de variables constantes:** luego de aplicar la 2D-DWT se obtienen 196 variables, de las cuales se observa que 46 de ellas son constantes, es decir, el valor de estos 46 píxeles es el mismo para cada una de las 169 imágenes, estas variables representan las zonas que son completamente negras en cada imagen y se decide eliminarlas ya que no representan información importante. Por lo tanto, solo nos quedarían un total de 150 variables.
4. **PCA:** el PCA se aplica a la matriz de tamaño 169x150, esto se realiza mediante la función **pca** de Matlab, la cual nos da como resultado la proyección de los sujetos sobre las componentes principales. Luego de aplicar el PCA se eligen las primeras 18 PC, las cuales capturan el 95 % de la varianza. Posterior a este paso, nuestra matriz de datos es de tamaño 169x18.
5. **DLF:** recordemos que en el PCA no se utiliza la información de la clase a la que pertenecen los sujetos, es por ello que para confirmar que las PC sean más separables, es necesario transformar los datos utilizando el DLF, lo que inflará la brecha entre las diferentes clases. Dado nuestro conjunto de datos  $I = \{x_1, x_2, \dots, x_{169}\}$ , donde  $x_i \in \mathbb{R}^{18}$  utilizaremos la siguiente etiqueta:

$$t_i = \begin{cases} +1 & \text{si } x_i \in \text{AD}, \\ -1 & \text{si } x_i \in \text{CN}. \end{cases}$$

El DLF no está implementado internamente en Matlab, por lo que se utilizó el algoritmo desarrollado por S. Mostapha Kalami Heris, el cual se puede encontrar en el File Exchange de Matlab [20]. El DLF nos da como resultado solo un autovalor significativo, el cual captura casi el 100 % de variabilidad, por lo cual se debería elegir solo un eje de proyección.

Sin embargo, los resultados de clasificación mejoran significativamente a medida que agregamos más ejes, por lo que para la proyección de los datos se eligen los tres primeros ejes (Ver Figura 3.4). Así, posterior a este paso, nuestra matriz de datos es de tamaño 169x3.

6. **ELM**: para la clasificación de los sujetos entrenamos una red neuronal SLFN con el algoritmo de aprendizaje ELM, donde utilizamos un 80 % de los datos para entrenamiento y el 20 % para validación. De igual manera, el algoritmo ELM no está implementado internamente en Matlab, por lo que se utilizó el algoritmo desarrollado por Tarek Berghout [2], el cual también se puede encontrar en el File Exchange de Matlab.

Este algoritmo necesita como datos de entrada la cantidad de nodos que tendrá la capa oculta, así como la proporción de los datos que se utiliza para el entrenamiento. También se debe especificar si se utilizará para clasificación o regresión y las etiquetas son codificadas de la siguiente manera:

$$t_i^* = \begin{cases} [1, 0] & \text{si } t_i = +1, \\ [0, 1] & \text{si } t_i = -1. \end{cases}$$

Los pesos de entrada son generados de manera uniforme en el intervalo  $(0, 1)$  a través de la función **rand** de Matlab. El algoritmo utiliza una función de activación Gaussiana a través de la función **radbas** de Matlab y la inversa generalizada de Moore-Penrose se calcula con la función **pinv** de Matlab cuyo cálculo se basa en la descomposición de valores singulares.

La salida del algoritmo nos proporciona entre otras cosas la precisión en los conjuntos de entrenamiento y validación, así como las correspondientes etiquetas predichas.

**Observación 3.1.** *Al utilizar distintos números de nodos ocultos se observó que el rendimiento de clasificación en el conjunto de validación no cambia significativamente.*

7. **Resultados de clasificación**: el rendimiento de la clasificación binaria se puede visualizar mediante una matriz de confusión como se muestra en la Tabla 3.3.

Los elementos diagonales de la matriz indican el número de predicciones correctas por parte del clasificador. Los elementos se pueden dividir

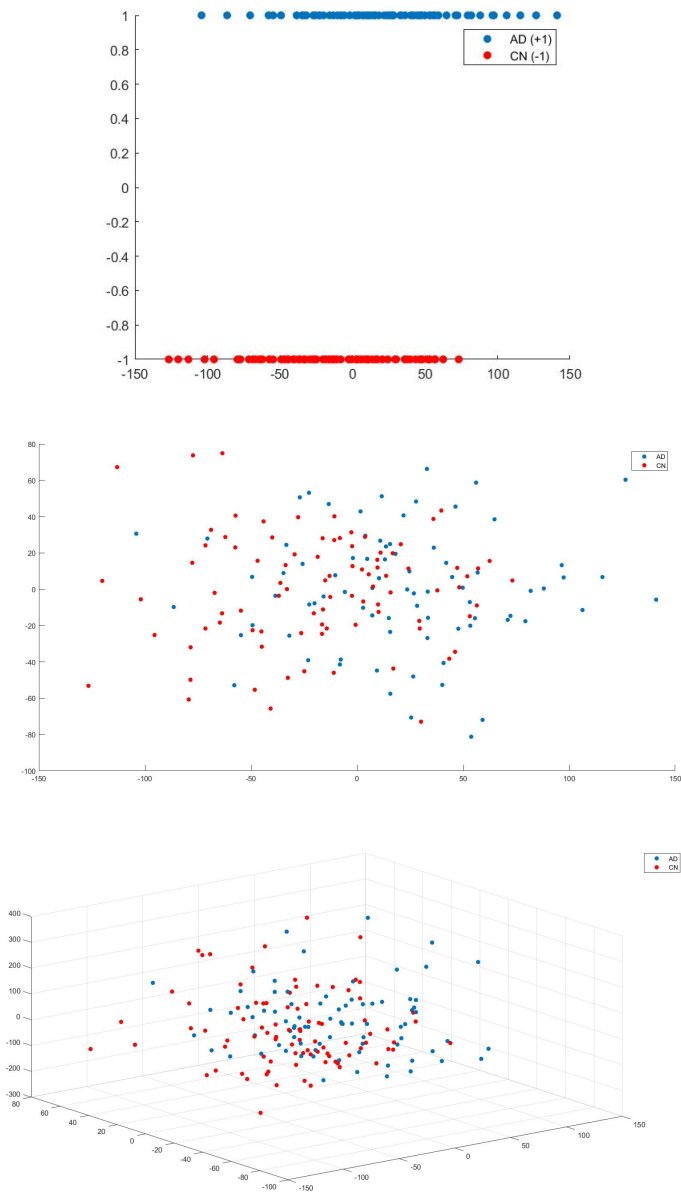


Figura 3.4: Gráfico de dispersión de las proyecciones DLF sobre los tres primeros ejes.

Tabla 3.3: Matriz de confusión.

Clase verdadera	Clase predicha	
	AD	CN
AD	TP	FN
CN	FP	TN

en verdadero positivo (TP) y verdadero negativo (TN), que representan sujetos correctamente etiquetados. Del mismo modo, el número de sujetos clasificados erróneamente puede estar representado por falso positivo (FP) y falso negativo (FN). La precisión mide la proporción de sujetos etiquetados correctamente por el clasificador.

$$\text{Precisión} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}.$$

Sin embargo, para un conjunto de datos con una distribución de clase muy desequilibrada, la precisión puede ser una métrica de rendimiento engañosa. Por lo tanto, también se utilizan dos métricas de rendimiento conocidas como sensibilidad y especificidad:

$$\text{Sensibilidad} = \frac{\text{TP}}{\text{TP} + \text{FN}},$$

$$\text{Especificidad} = \frac{\text{TN}}{\text{TN} + \text{FP}}.$$

La sensibilidad mide la tasa de verdaderos positivos, mientras que la especificidad mide la tasa de verdaderos negativos, en nuestro caso estas métricas representan la tasa de los sujetos AD clasificados correctamente y la tasa de los sujetos CN clasificados correctamente, respectivamente.

El algoritmo ELM es ejecutado diez veces y el rendimiento de clasificación para cada ejecución es presentado en la Tabla 3.4 y en la Figura 3.5.

Con base en los resultados obtenidos y al análisis realizado podemos llegar a las conclusiones presentadas en la siguiente sección.

## 3.2. Resumen y Conclusiones.

Se presenta una metodología que combina diferentes técnicas estadísticas y de aprendizaje automático, la cual se espera que pueda ayudar a diagnosticar la enfermedad de Alzheimer a través de una imagen de resonancia magnética.

Tabla 3.4: Rendimiento de clasificación sobre el conjunto de datos ADNI.

Ejecución	Precisión	Sensibilidad	Especificidad
1	83.43	83.33	83.52
2	84.62	82.93	86.21
3	80.47	81.33	79.79
4	86.98	88.16	86.02
5	85.21	84.81	85.56
6	87.57	86.42	88.64
7	88.76	90.67	87.23
8	86.98	85.37	88.51
9	86.39	85.19	87.50
10	90.53	89.02	91.95
	$86.09 \pm 2.83$	$85.72 \pm 2.90$	$86.49 \pm 3.25$

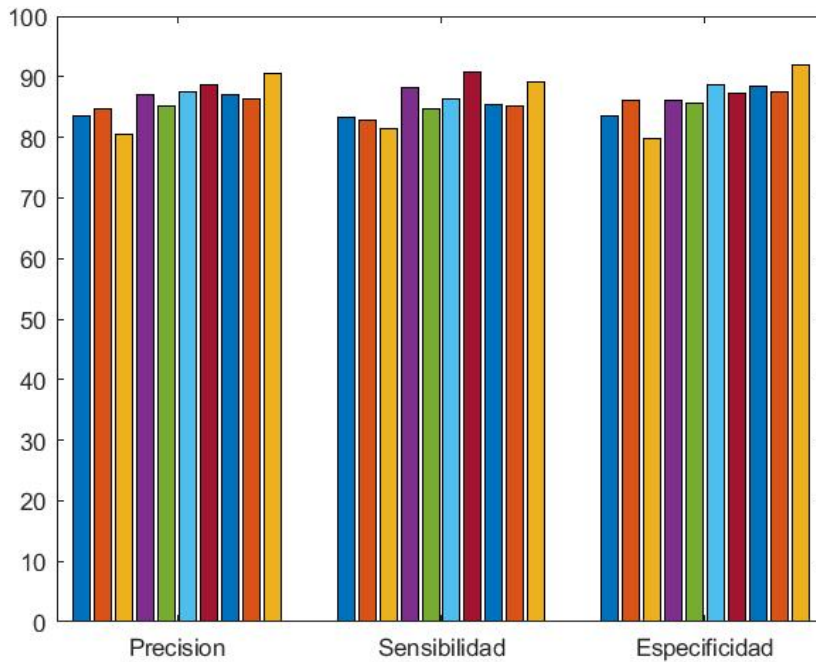


Figura 3.5: Diagrama de barras del rendimiento de clasificación.

Cada una de las técnicas utilizadas juega un rol importante en cuanto a la extracción eficiente de características, reducción de dimensionalidad y clasificación. Este trabajo puede llevar al diseño de un diagnóstico automatizado asistido por computadora, el cual tendrá el potencial de cambiar los métodos de diagnóstico actuales, los cuales son realizados por radiólogos y médicos de forma manual.

Además, mediante esta metodología y una base de datos lo suficientemente grande y de buena calidad se puede lograr una detección temprana de la AD cuando los síntomas son muy leves o incluso antes de que aparezcan. Esto tendría una enorme repercusión en lo que respecta a la calidad de vida de los pacientes. El diagnóstico debe complementarse mediante otras pruebas y evaluaciones médicas. Con un diagnóstico temprano los médicos pueden ofrecer intervenciones con medicamentos y sin medicamentos que pueden aliviar la carga de la enfermedad, también permite que los pacientes puedan tomar decisiones sobre su futuro y brinda a sus familiares más oportunidades de aprender sobre la enfermedad.

Los resultados de la implementación de esta metodología en el conjunto de datos de ADNI, los cuales se observan en la Tabla 3.4 y la Figura 3.5, validan la utilidad de nuestra propuesta. El rendimiento de clasificación obtenido es prometedor ya que se encuentra alrededor del 85 % y se espera que este porcentaje se pueda mejorar notablemente al tener una base de datos mas amplia y que los cortes adecuados de las imágenes sean identificados mediante la ayuda de los médicos especialistas en la AD.

La investigación acerca del desarrollo de la enfermedad de Alzheimer se ha convertido en la actualidad en una de las áreas de mayor interés para los investigadores en esta patología. El análisis de neuro imágenes ha sido una herramienta básica en diagnósticos médicos, sin embargo, el avance en ciencias computacionales y equipos de gran capacidad ha permitido el almacenaje de imágenes y de datos de grandes dimensiones, los cuales son actualizados periódicamente. Por tal motivo el aumento en el tamaño de las bases de datos ha originado problemas de interés en estas áreas médicas, y la intervención de especialistas en análisis de datos es requerida.

En los últimos años se ha observado que las aplicaciones de las matemáticas en la medicina cada vez son mayores [16], es por ello que se debe contar cada vez mas con datos médicos a los cuales se puedan acceder de manera libre. Este tipo de investigaciones sugiere que debe existir un trabajo colaborativo entre matemáticos y médicos, lo que dará como resultado diagnósticos automatizados basados en los datos y que esto a su vez pueda disminuir la cantidad de pacientes con diagnósticos erróneos y nos de la oportunidad de diagnosticar la enfermedad de Alzheimer en las fases tempranas para que los pacientes y sus familiares puedan sobrellevar de mejor manera esta enfermedad.

# Bibliografía

- [1] A. Meyer-Base, V. Schmid (2014). *Pattern Recognition and signal analysis in medical imaging*, Second Edition, Academic Press.
- [2] Berghout Tarek (2020). Extreme Learning Machine for classification and regression (<https://www.mathworks.com/matlabcentral/fileexchange/69812-extreme-learning-machine-for-classification-and-regression>), MATLAB Central File Exchange.
- [3] Berzal Fernando (2018). *Redes Neuronales & Deep Learning*, 1st edition, EUG, Editorial Universidad de Granada.
- [4] C. Chui (1992). *An Introduction to Wavelets*, 1st edition, Academic Press.
- [5] D. Jha, Ji-In Kim, and Goo-Rak Kwon (2017). *Pathological brain detection using weiner filtering, 2D-discrete wavelet transform, probabilistic PCA, and random subspace ensemble classifier*. Computat. Intelli. Neurosci. 2017, pp. 4205141-4205141.
- [6] E. Cambria and G. B. Huang (2013). “*Extreme learning machines*”, IEEE Intelligent Systems, vol. 28, no. 6, pp. 2–31.
- [7] G. B. Huang, Q. Y. Zhu, and C. K. Siew (2006). *Extreme learning machine: Theory and applications*. Neurocomputing 70, pp. 489-501.
- [8] Hilera José, Martínez Victor (1994). *Redes Neuronales Artificiales. Fundamentos, Modelos y Aplicaciones*. Serie Paradigma, RaMa.
- [9] J. Debes, A. Saruar, P. Jae-Young , L. Kun Ho, K. Goo-Rak (2018). *Alzheimer’s Disease Detection Using Extreme Learning Machine, Complex Dual Tree Wavelet Principal Coefficients and Linear Discriminant Analysis*, Journal of Medical Imaging and Health Informatics Vol. 8, pp. 1–10.
- [10] M. Deisenroth, A. Aldo Faisal, C. Soon Ong (2020). *Mathematics for Machine Learning*, 1st edition, Cambridge University Press.

- [11] M. G. González, G. D. Santiago, V. Slezakz, A. Peuriotz (2018). *Aplicación de wavelets en la detección fotoacústica de gases traza con señales ruidosas*, Revista elektron, Vol. 2, No. 1, pp. 26-29.
- [12] Napler Addison (2002). *The Illustrated Wavelet Transform Handbook*, 1st edition, Taylor & Francis.
- [13] Organización Mundial de la Salud (2017). *Datos interesantes acerca del envejecimiento*. (<https://www.who.int/ageing/about/facts/es/>).
- [14] R. K. Lama, J. Gwak, and S. W. Lee (2017). *Diagnosis of Alzheimer's disease based on structural MRI images using a regularized extreme learning machine and PCA features*. Journal of Healthcare Engineering in Hindawi 2017, 5485080.
- [15] S. Chaplot, L. M. Patnaik, N. R. Jagannathan (2006). *Classification of magnetic resonance brain images using wavelets as input to support vector machine and neural network*. Biomedical Signal Processing and Control 1, pp. 86-92.
- [16] S. K. J. and G. S. (2019). *Prediction of Heart Disease Using Machine Learning Algorithms*. 2019 1st International Conference on Innovations in Information and Communication Technology (ICIICT), CHENNAI, India, 2019, pp. 1-5.
- [17] S. Mallat (1999). *A Wavelet Tour of Signal Processing*, Second edition, Academic Press.
- [18] Thome Néstor (2019). *La inversa generalizada de Moore-Penrose y aplicaciones*, Serie: Textos. Vol. 21, Publicaciones Electrónicas Sociedad Matemática Mexicana ([https://www.pesmm.org.mx/Serie%20Textos\\_archivos/T21.pdf](https://www.pesmm.org.mx/Serie%20Textos_archivos/T21.pdf)).
- [19] Walnut David (2002). *An Introduction to Wavelet Analysis*, Birkhäuser.
- [20] Yarpiz (2020). Linear Discriminant Analysis (LDA) aka. Fisher Discriminant Analysis (FDA) (<https://www.mathworks.com/matlabcentral/fileexchange/53151-linear-discriminant-analysis-lda-aka-fisher-discriminant-analysis-fda>), MATLAB Central File Exchange.