



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA



ESCUELA TÉCNICA  
SUPERIOR INGENIEROS  
INDUSTRIALES VALENCIA

TRABAJO FIN DE GRADO EN INGENIERÍA BIOMÉDICA

# **DISEÑO Y DESARROLLO DE UN MÉTODO DE SUPERRESOLUCIÓN EN IMÁGENES DE RM USANDO DEEP LEARNING**

AUTOR: CARLOS ANDREU VILARROIG

TUTOR: JOSÉ VICENTE MANJÓN HERRERA

**Curso Académico: 2018-19**



# Agradecimientos y dedicatoria

Uno de los pensamientos que me ha perseguido a lo largo de este último año académico es aquel tan conocido y, en ocasiones, tan olvidado refrán popular que nos recuerda que “es de bien nacido ser agradecido”. Y como resultaría de una gran desconsideración por mi parte olvidar a aquellas personas que me han hecho ser quién soy, me permito en este apartado cierta libertad literaria para agradecerles y dedicarles con mis (torpes e insuficientes) palabras este trabajo.

Quisiera agradecer en primer lugar a José Vicente Manjón, tutor, mentor y guía de este proyecto, que hace un año me aceptara como alumno y me abriera sin reservas las puertas del laboratorio ITACA. Gracias, José Vicente, por tus sabios consejos, por tu infinita paciencia al resolver mis dudas, por permitir equivocarme para poder aprender de mis errores, por guiarme en cada paso de este proyecto, por tus tantas conversaciones y moralejas, y, sobre todo, por transmitirme un poco de esa inagotable curiosidad científica que, ten por seguro, continuaré alimentando en el futuro.

Por otro lado, no sería justo olvidar a todos y cada uno de mis compañeros y amigos del laboratorio ITACA: José Enrique, Alexa, Lee, Javi, Tanja, Germán, Mari, Vicente, Pablo, Marta, Edu, Juanmi, Carlos y Elías, que con su apoyo, comprensión y ayuda incondicional, han conseguido hacerme sentir entre ellos como en mi segunda casa. Y sin ánimo de desestimar a ninguno de ellos, quisiera hacer una mención especial a José Enrique, que con su habitual paciencia, calma y peculiar sentido del humor, siempre ha estado dispuesto a ofrecer su ayuda sin reparos. A todos, gracias.

En mi recorrido académico, quisiera dar las gracias a todos aquellos profesores que me han conseguido marcar y transmitir, no solo sus conocimientos, sino también su ilusión por su ciencia y trabajo. De entre ellos, quisiera agradecer especialmente a Alejandro Vignoni y Yadira Boada por ser un verdadero ejemplo como docentes pero, ante todo, como personas. También quisiera aprovechar esta ocasión para agradecerles a Roger y Alejandro la inmensa amistad y apoyo que me han brindado en estos años de universidad, especialmente en los últimos momentos de esta experiencia. Espero haberles podido dar tanto como ellos me lo han dado a mí.

No podría concluir sin dedicarle unas palabras a uno de los pilares fundamentales de mi vida: mi familia. Gracias a mis padres, por ser mis maestros en la vida y por vuestro silencioso sacrificio. A vosotros os debo todos mis éxitos y virtudes, y sin vosotros no hubiese llegado a ser el hombre que soy. Gracias a mis hermanos, Ana, Raquel, Iván y Fátima, por soportar con paciencia mis

muchos defectos, y por ese apoyo fraternal que no se demuestra con palabras, sino con hechos. Gracias a mis abuelos, por transmitirme la experiencia de una entera vida de esfuerzo. Gracias a mi tío Jaime, padrino y filósofo, por esas innumerables y apasionantes conversaciones y por su sabio consejo. Y aun sabiendo que con estas breves palabras es tarea imposible agradecerles toda una vida, sencillamente gracias.

Por último, me he permitido incluir a una persona más en estos agradecimientos, a la que le prometí, como hombre sincero de donde crece la palma, una dedicatoria especial. A ella le debo mi fuerza en las horas bajas, con ella he compartido mis preocupaciones y alegrías, de ella tan solo he recibido, incluso en los días más grises y lluviosos, una hermosa sonrisa, y solo ella, a pesar de mis tantas debilidades y limitaciones, ha sabido ver y querer lo mejor de mí. Por todas estas razones, y por infinitas razones más, no puedo más que dedicarte a ti mi último y sincero “gracias”.

*“Este trabajo está dedicado a mi tío Miguel. Esposo, padre y trabajador ejemplar, vivió hasta su último día humildemente y siempre entregado a los demás.”*

# Resumen

La enfermedad de Alzheimer (EA) es una de las enfermedades neurodegenerativas que mayor prevalencia presenta en la población de edad avanzada. A pesar de que la etiología aún es desconocida, un diagnóstico precoz es fundamental para poder retrasar el avance de la enfermedad y mejorar los tratamientos. Actualmente, existen numerosas investigaciones que buscan nuevas pruebas diagnósticas más robustas y precoces. De entre ellas, las nuevas técnicas de imagen médica, como la superresolución, unidas a los avances en el campo del aprendizaje automático, concretamente del *Deep Learning*, han resultado ser un punto de inflexión en el diagnóstico precoz de la EA.

En el siguiente trabajo se propone un método de superresolución y clasificación de imágenes de Resonancia Magnética (IRM) basado en redes neuronales artificiales. Inicialmente se dispone de un conjunto de volúmenes IRM de la región del hipocampo pertenecientes a sujetos que han sido etiquetadas en dos clases: *sanos* o *EA*. El desarrollo del proyecto se ha dividido en dos fases claramente diferenciadas: fase de superresolución y fase de clasificación. En la primera fase se han desarrollado y entrenado dos redes neuronales convolucionales o CNN que transforman las imágenes de baja resolución potenciadas en T1 a imágenes de superresolución potenciadas en T1 y T2. En la segunda fase, a partir de las imágenes generadas por las primeras redes y sus correspondientes etiquetas, se ha diseñado y entrenado una segunda red neuronal de clasificación que extrae las características de las imágenes de modalidad T1 y T2, y clasifica al sujeto como *sano* o con *EA*.

Los resultados obtenidos tras la validación del método muestran el gran potencial de las redes neuronales artificiales en las tareas de extracción de características y clasificación. Asimismo, el trabajo demuestra que las técnicas de *Deep Learning* suponen un importante avance, no solo en el estudio de nuevas pruebas y criterios diagnósticos de la EA, sino también en el desarrollo de sistemas de ayuda al diagnóstico y a la toma de decisiones clínicas.

**Palabras Clave:** Alzheimer, diagnóstico, IRM, superresolución, síntesis, *Deep Learning*, redes neuronales artificiales, extracción de características, clasificación.



# Resum

La malaltia d'Alzheimer (EA) és una de les malalties neurodegeneratives que major prevalença presenta en la població d'edat avançada. A pesar que l'etiologia encara és desconeguda, un diagnòstic precoç és fonamental per a poder retardar l'avanç de la malaltia i millorar els tractaments. Actualment, existeixen nombroses recerques que busquen noves proves diagnòstiques més robustes i precoces. De totes elles, les noves tècniques d'imatge mèdica, com la superresolució, unides als avanços en el camp de l'aprenentatge automàtic, concretament del *Deep Learning*, han resultat ser un punt d'inflexió en el diagnòstic precoç de l'EA.

En el següent treball es proposa un mètode de superresolució i classificació d'imatges de Resonància Magnètica (IRM) basat en xarxes neuronals artificials. Inicialment es disposa d'un conjunt de volums IRM de la regió de l'hipocamp pertanyents a subjectes que han estat etiquetats en dues classes: *sans* o *EA*. El desenvolupament del projecte s'ha dividit en dues fases clarament diferenciades: fase de superresolució i fase de classificació. En la primera fase s'han desenvolupat i entrenat dues xarxes neuronals convolucionals o CNN que transformen les imatges de baixa resolució potenciades en T1 a imatges de superresolució potenciades en T1 i T2. En la segona fase, a partir de les imatges generades per les primeres xarxes i les seves corresponents etiquetes, s'ha dissenyat i entrenat una segona xarxa neuronal que extreu les característiques de les imatges de modalitat T1 i T2, i classifica al subjecte com a *sa* o amb *EA*.

Els resultats obtinguts després de la validació del mètode mostren el gran potencial de les xarxes neuronals artificials en les tasques d'extracció de característiques i classificació. Així mateix, el treball demostra que les tècniques de *Deep Learning* suposen un important avanç, no només en l'estudi de noves proves i criteris diagnòstics de l'EA, sinó també en el desenvolupament de sistemes d'ajuda al diagnòstic i a la presa de decisions clíniques.

**Paraules Clau:** Alzheimer, diagnòstic, IRM, superresolució, síntesi, *Deep Learning*, xarxes neuronals artificials, extracció de característiques, classificació.





# Abstract

Alzheimer's disease (AD) is one of the most prevalent neurodegenerative diseases in the elderly population. Although the aetiology is still unknown, early diagnosis is essential in order to delay the progression of the disease and improve treatments. Currently, there is a great deal of research looking for new, more robust and earlier diagnostic tests. Among them, new medical imaging techniques, such as superresolution, together with advances in the field of automatic learning, specifically Deep Learning, have proved to be a turning point in the early diagnosis of AD.

The following work proposes a method of superresolution and classification of Magnetic Resonance images (MRI) based on artificial neural networks. Initially, a set of volumetric MR images of the hippocampus region are available for subjects that have been tagged in two classes: *healthy* or *AD*. The development of the project has been divided into two clearly defined phases: the superresolution phase and the classification phase. In the first phase, two convolutional neural networks or CNN have been developed and trained to transform low resolution T1 enhanced images to T1 and T2 enhanced superresolution images. In the second phase, from the images generated by the first networks and its labels, a second neural network has been designed and trained to extract the characteristics of the T1 and T2 modality images from a subject, and to classify the subject as *healthy* or with *AD*.

The results obtained after the validation of the method show the great potential of artificial neural networks in feature extraction and classification tasks. In addition, the work shows that Deep Learning techniques represent an important advance, not only in the study of new tests and diagnostic criteria for AD, but also in the development of diagnosis and clinical decision support systems.

**Keywords:** Alzheimer, diagnostic, MRI, superresolution, synthesis, Deep Learning, artificial neural networks, feature extraction, classification.



# Índice de figuras

1.1. <b>Prevalencia de la EA en diferentes grupos de edad y en diferentes países europeos.</b> Se observa un incremento exponencial de la prevalencia en grupos de mayor edad en todos los casos. Fuente: Confederación Española de Alzheimer (CEAFA) 2014. . . . .	4
1.2. <b>Previsión de la prevalencia de la EA (en millones de personas) entre los años 2015-2050.</b> Se observa un aumento significativo de la prevalencia a nivel global. Fuente: Alzheimer's Disease International (ADI) 2015. . . . .	5
1.3. <b>Previsión del coste de la EA (en millones de dólares USD) entre los años 2015-2050.</b> Se observa un aumento significativo del coste a nivel global. Fuente: Alzheimer's Disease International (ADI) 2015. . . . .	5
1.4. <b>Morfología celular de la neurona.</b> Fuente: <i>Future Website of Anatomia</i> . . . . .	6
1.5. <b>Esquema del Sistema Nervioso Central y sus estructuras.</b> Fuente: <i>Células gliales</i> . . . . .	7
1.6. <b>Estructuras del HC y la EC.</b> a) Vista 3D sagital (izquierda) y coronal posterior (derecha) de la localización del HC en el encéfalo. b) Corte axial del hemisferio izquierdo del cerebro, en el que se aprecian el HC y la EC. Fuentes: <i>Lifeder.com</i> y <i>Asociación Educar</i> . . . . .	8
1.7. <b>Evolución de los parámetros en las diferentes fases la EA.</b> En la fase preclínica puede apreciarse como determinados biomarcadores propios de la EA presentan niveles anormales. Fuente: Confederación Española de Alzheimer (CEAFA) 2014. . . . .	9
1.8. <b>Orientación de los espines de los núcleos <math>H^1</math> en presencia y ausencia del campo magnético <math>\vec{B}</math>.</b> a) Orientación al azar de los núcleos en ausencia de campo magnético $\vec{B}$ . b) Orientación ordenada con movimiento de precesión de los núcleos en los dos estados ( <i>up</i> y <i>down</i> ) en presencia de campo magnético $\vec{B}$ . Fuente: Gili 1993. . . . .	13

1.9. <b>Vector magnetización <math>\vec{M}</math> de un vóxel ante la presencia de un campo magnético <math>\vec{B}</math> en estado de reposo y excitación.</b> a) El vector $\vec{M}$ en estado de reposo equivale a la suma vectorial de los momentos $\vec{\mu}$ de los núcleos $H^1$ del vóxel. b) Movimiento de nutación durante la excitación del vector $\vec{M}$ mediante un pulso RF y con una separación respecto a la posición de equilibrio $\alpha$ . Fuente: Gili 1993. . . . .	14
1.10. <b>Fase de relajación del vector de magnetización <math>\vec{M}</math> en un pulso RF de <math>90^\circ</math>.</b> La recuperación en el tiempo de las componentes del vector $\vec{M}$ en los planos longitudinal ( $\vec{M}_z$ ) y transversal ( $\vec{M}_{xy}$ ) da lugar a diferentes señales y modalidades de imagen IRM. Fuente: Gili 1993. . . . .	15
1.11. <b>Modalidades de la IRM.</b> Cortes axiales del cerebro potenciados en T1 (izquierda), T2 (centro) y densidad protónica D (derecha). Fuente: <i>Radiology Café</i> . . . . .	16
1.12. <b>Análisis de la atrofia del lóbulo temporal en IRM.</b> Cortes coronales del cerebro con diferentes niveles de atrofia en la escala visual de Scheltens en el rango de 0 (ausencia de atrofia) a 4 (atrofia severa). Fuente: <i>Radiología Club</i> . . . . .	17
1.13. <b>Comparación de la neurona biológica con la neurona artificial de una ANN.</b> El valor de salida de la neurona depende de la suma ponderada de las entradas y de un factor <i>bias</i> . Fuentes: <i>Dpto. de Ciencias de la Computación e Inteligencia Artificial. Universidad de Sevilla</i> y <i>El Blog Verde</i> . . . . .	20
1.14. <b>Operación de convolución de un píxel de una imagen 2D.</b> La operación de convolución se realiza mediante un kernel de dimensiones $3 \times 3$ (radio $r = 1$ ). Se observa que el resultado en la posición de un píxel equivale al producto entre los valores del kernel y los valores de la región $3 \times 3$ en torno al píxel. Fuente: <i>Towards Data Science (I)</i> . . . . .	22
1.15. <b>Modificación de la dimensionalidad de las neuronas de una CNN.</b> a) <i>Downsampling</i> o reducción de la dimensionalidad mediante la selección del valor máximo ( <i>Max pooling</i> ) o promedio ( <i>Average pooling</i> ) de las regiones de la imagen. b) <i>Upsampling</i> o aumento de la dimensionalidad mediante la generación de nuevos elementos en la matriz de valor 0 ( <i>Zero padding</i> ). Fuentes: <i>Quora</i> y <i>Stack Exchange (I)</i> . . . . .	24
1.16. <b>Esquema de las principales arquitecturas de las ANN.</b> a) Perceptrón multicapa o <i>DenseNet (fully-connected network)</i> . b) CNN + <i>DenseNet</i> . c) <i>Autoencoder</i> convolucional o <i>UNet</i> . d) GAN. Fuentes: Ramsundar y Zadeh 2018, <i>free Code Camp (I)</i> , <i>GitHub (I)</i> y <i>GitHub (II)</i> . . . . .	25
1.17. <b>Evolución del error o <i>loss</i> durante el entrenamiento de una ANN.</b> a) Minimización de la función de coste o <i>loss function</i> por descenso de gradiente. b) Evolución estable de las <i>losses</i> de entrenamiento y validación en las diferentes <i>epochs</i> del entrenamiento hasta alcanzar la convergencia. c) Efecto del sobreajuste u <i>overfitting</i> sobre las <i>losses</i> de entrenamiento y validación. d) Evolución inestable de las <i>losses</i> de entrenamiento y validación en las diferentes <i>epochs</i> del entrenamiento. Fuentes: <i>Science</i> , <i>Machine Learning Mastery (I)</i> , <i>Machine Learning Mastery (II)</i> y <i>free Code Camp (II)</i> . . . . .	29

1.18. <b>Técnicas de regularización en el entrenamiento de una ANN.</b> a) <i>Data augmentation</i> . b) Bloque residual. c) <i>Dropout</i> . d) <i>Fine-tuning</i> . Fuentes: <i>Research Gate (I)</i> , <i>Towards Data Science (II)</i> , <i>Stack Exchange (II)</i> y <i>Data Wow</i> . . . . .	30
3.1. <b>Visor de la aplicación ITK-SNAP de una imagen volumétrica IRM de la región del HC del cerebro.</b> La aplicación muestra los planos axial (superior izquierda), coronal (superior derecha) y sagital (inferior derecha) de la imagen, así como una representación tridimensional de los tres ejes (inferior izquierda). . . . .	39
3.2. <b>Esquema de la arquitectura de la red T1 SR.</b> . . . . .	42
3.3. <b>Esquema de la arquitectura de la red de síntesis T2 SR.</b> . . . . .	43
3.4. <b>Esquema de la arquitectura de la red de clasificación.</b> . . . . .	44
3.5. <b>Cortes sagitales de un <i>crop</i> IRM potenciado en T1.</b> a) Corte HR original. b) Corte LR. . . . .	46
3.6. <b>Esquema del preprocesado de los <i>crops</i> en la fase de SR.</b> Las etapas del preprocesado son: a) <i>Padding</i> . b) Normalización de intensidad. c) <i>Downsampling</i> (solo en datos T1). d) <i>Upsampling</i> (solo en datos T1). e) Extracción de <i>patches</i> . . . . .	47
3.7. <b>Gráfica con distribuciones de probabilidad beta con diferentes parámetros <math>\alpha</math> y <math>\beta</math>.</b> Puede apreciarse que cuando $\alpha = \beta$ , la distribución es simétrica. . . . .	53
4.1. <b>Diagramas <i>box-plot</i> de los valores PSNR de los casos de <i>test</i> de los casos en los diferentes experimentos de la fase de SR.</b> . . . . .	56
4.2. <b>Diagramas <i>box-plot</i> de los coeficientes de correlación de los casos de <i>test</i> en los diferentes experimentos de la fase de SR.</b> . . . . .	57
4.3. <b>Diagramas <i>box-plot</i> de los índices SSIM de los casos de <i>test</i> en los diferentes experimentos de la fase de SR.</b> . . . . .	57
4.4. <b>Gráficas de las métricas calculadas individualmente para cada caso de <i>test</i> en el Experimento 4.</b> a) PSNR. b) Coeficiente de correlación. c) SSIM. . . . .	58
4.5. <b>Cortes sagitales de tres <i>crops</i> IRM potenciados en T1 pertenecientes a un sujeto del conjunto de <i>test</i>.</b> a) <i>Crop</i> LR. b) <i>Crop</i> SR generado por la mejor red (Experimento 4). c) <i>Crop</i> HR. . . . .	60
4.6. <b>Diagramas <i>box-plot</i> de los PSNR de los casos de <i>test</i> en los diferentes experimentos de la fase de SR.</b> . . . . .	61
4.7. <b>Diagramas <i>box-plot</i> de los coeficientes de correlación de los casos de <i>test</i> en los diferentes experimentos de la fase de SR.</b> . . . . .	62
4.8. <b>Diagramas <i>box-plot</i> de los índices SSIM de los casos de <i>test</i> en los diferentes experimentos de la fase de SR.</b> . . . . .	62
4.9. <b>Gráficas de las métricas calculadas para cada uno de los casos de <i>test</i>.</b> a) PSNR. b) Coeficiente de correlación. c) SSIM. . . . .	63

4.10. Cortes sagitales de tres <i>crops</i> IRM potenciados en T1 y T2 pertenecientes a un sujeto del conjunto de <i>test</i> . a) <i>Crop</i> SR potenciado en T1 generado por la mejor red T1 SR. b) <i>Crop</i> SR potenciado en T2 generado por la mejor red de síntesis T2 (Experimento 5). c) <i>Crop</i> HR potenciado en T2. . . . .	65
4.11. Diagrama de barras de las métricas obtenidas en cada experimento de la fase de clasificación. . . . .	66
4.12. Matriz de confusión de los casos de <i>test</i> del Experimento 5 en la fase de clasificación. . . . .	67
4.13. Curva ROC de los casos de <i>test</i> del Experimento 5 en la fase de clasificación. La curva muestra la Sensibilidad frente a 1 - Especificidad de la red clasificadora. . . . .	68
4.14. Cortes sagitales de tres <i>crops</i> IRM potenciados en T1 pertenecientes a tres sujetos del conjunto de <i>test</i> ADNI. a) <i>Crop</i> AD clasificado como AD. b) <i>Crop</i> CN clasificado como CN. c) <i>Crop</i> AD clasificado como CN. . . . .	69
4.15. Cortes sagitales de tres <i>crops</i> IRM potenciados en T1 y T2 pertenecientes a dos sujetos CN y AD del conjunto de <i>test</i> ADNI. a) <i>Crop</i> potenciado en T1 LR original del sujeto CN. b) <i>Crop</i> potenciado en T1 SR generado por la mejor red T1 SR (Experimento 5) del sujeto CN. c) <i>Crop</i> potenciado en T2 SR generado por la mejor red de síntesis T2 SR (Experimento 5) del sujeto CN. d) <i>Crop</i> potenciado en T1 LR original del sujeto AD. e) <i>Crop</i> potenciado en T1 SR generado por la mejor red T1 SR (Experimento 5) del sujeto AD. f) <i>Crop</i> potenciado en T2 SR generado por la mejor red de síntesis T2 SR (Experimento 5) del sujeto AD. . . . .	70
4.16. Matriz de confusión de los casos de <i>test</i> SR del Experimento 5 en la fase de clasificación. . . . .	71
4.17. Curva ROC de los casos de <i>test</i> del Experimento 5 en la fase de clasificación. La curva muestra la Sensibilidad frente a 1 - Especificidad de la red clasificadora. . . . .	71

# Índice de tablas

1.1. <b>Etapas de la EA, períodos y cuadros clínicos.</b> Se observa en los cuadro clínicos, a medida que avanza la enfermedad, la demencia y la gravedad de los síntomas van en aumento. . . . .	10
1.2. <b>Principales funciones de activación de las ANN.</b> . . . . .	21
1.3. <b>Funciones de coste destacadas de las ANN.</b> . . . . .	26
3.1. <b>Especificaciones del equipo PC de experimentación.</b> . . . . .	38
3.2. <b>Conjuntos de entrenamiento, validación y <i>test</i> de la fase de clasificación.</b> . . . . .	51
4.1. <b>Media <math>\pm</math> desviación típica de las métricas resultantes de los experimentos de las redes de síntesis SR T1.</b> . . . . .	56
4.2. <b>Resultados de la prueba de normalidad Shapiro-Wilk aplicado a las muestras de las métricas de la Referencia y el Experimento 4.</b> El número de muestras de cada grupo es de $n = 20$ . . . . .	59
4.3. <b>Resultados del T-Test pareado de comparación de medias entre las dos muestras de métricas de la Referencia y el Experimento 4.</b> El número de muestras de cada grupo es de $n = 20$ . . . . .	59
4.4. <b>Comparación del método propuesto de síntesis de imágenes IRM T1 SR con algunos de los métodos de SR del estado del arte.</b> . . . . .	60
4.5. <b>Media <math>\pm</math> desviación típica de las métricas resultantes de los experimentos de las redes de síntesis SR T2.</b> . . . . .	61
4.6. <b>Resultados de la prueba de normalidad Shapiro-Wilk aplicado a las muestras de las métricas de la Referencia y el Experimento 5.</b> El número de muestras de cada grupo es de $n = 20$ . . . . .	63
4.7. <b>Resultados de la prueba de rangos y signos de Wilcoxon entre las diferentes métricas del Experimento 5 y la Referencia.</b> . . . . .	64

4.8. Comparación del método propuesto de síntesis de imágenes IRM T2 SR con algunos de los métodos del estado del arte. . . . .	65
4.9. Métricas resultantes de los experimentos de las redes de clasificación. . . . .	66
4.10. Métricas resultantes de la evaluación de la red de clasificación del Experimento 5 entrenando con datos T1 LR, T1 SR y T1 + T2 SR. . . . .	70
4.11. Comparación del método propuesto de clasificación con algunos de los métodos del estado del arte. . . . .	72
5.1. Costes de mano de obra. . . . .	77
5.2. Costes de <i>hardware</i> . . . . .	78
5.3. Costes de <i>software</i> . . . . .	78
5.4. Presupuesto de ejecución material. . . . .	79
5.5. Presupuesto de ejecución por contrata. . . . .	79



# Lista de acrónimos

- AD** *Alzheimer's disease*. 38, 51, 52, 66, 68–70
- ADNI** *Alzheimer's Disease Neuroimaging Initiative*. 37, 38, 40, 47, 51, 52, 65, 68–70, 72
- ANN** *Artificial Neural Networks*. 20, 21, 23–30, 32–35, 37, 38, 40, 41, 43, 48, 73, 74
- ANTs** *Advanced Normalization Tools*. 39, 45, 47, 78
- APP** *Amyloid Precursor Protein*. 7
- CN** *Control Normal*. 38, 51, 52, 66, 68–70
- CNN** *Convolutional Neural Networks*. 22–25, 32, 33, 41, 43, 45
- CT** *Computed Tomography*. 10, 11, 39
- DCL** *Deterioro Cognitivo Leve*. 9, 38
- DL** *Deep Learning*. 3, 19, 20, 32–35, 41, 48
- DRRN** *Deep Recursive Residual Network*. 41
- DSC-MRI** *Dinamic Suceptibility Constrast-MRI*. 33
- DSSIM** *Structural Dissimilarity*. 49, 50, 61, 62
- EA** *enfermedad de Alzheimer*. 3–5, 7–12, 17, 33–35, 38, 40, 43, 45, 69, 74
- EC** *entorhinal cortex*. 8, 17, 45
- ERM** *Espectroscopía por Resonancia Magnética*. 17
- GAN** *Generative Adversarial Networks*. 24, 25, 32
- GDL** *Gradient Difference Loss*. 26, 49, 50, 56, 57
- GPU** *Graphics Processing Unit*. 38
- HC** *hyppocampus*. 8, 17, 39, 45, 51, 53
- HR** *High Resolution*. 18, 19, 32, 45, 46, 48–50, 56, 59, 60, 64, 65

- IRM** imagen de Resonancia Magnética. 3, 11–17, 32, 33, 35, 37–49, 51–53, 55, 59, 60, 64–66, 68–73
- LR** *Low Resolution*. 17–19, 40–42, 45–51, 53, 56, 59, 60, 66, 69, 70
- MAE** *Mean Absolute Error*. 26, 49, 50, 56, 57, 61, 62
- MNI** *Montreal Neurological Institute*. 37, 45, 47
- MPR** *Multiplanar Reconstruction*. 39
- MRI** *Magnetic Resonance image*. 72
- MSE** *Mean Squared Error*. 26
- NAMICS** *National Alliance for Medical Image Computing*. 65
- NIFTI** *Neuroimaging Informatics Technology Initiative*. 38, 39, 45–49, 51
- NLM** *Non-Local Means*. 37, 45, 47
- PET** *Positron Emission Tomography*. 10, 11, 33, 38, 72, 74
- PSNR** *Peak Signal-to-Noise Ratio*. 31, 50, 55–61, 63–65
- ReLU** *Rectified Linear Unit*. 21, 41–44
- RF** radiofrecuencia. 12–15
- RM** Resonancia Magnética. 12, 17, 32, 38
- RMSE** *Root Mean Squared Error*. 31
- ROC** *Receiver Operating Characteristic*. 31, 65, 67, 68, 70, 71
- ROC AUC** *Receiver Operating Characteristic - Area Under Curve*. 31, 52, 65–68, 70, 72
- SN** Sistema Nervioso. 5–7, 20
- SNC** Sistema Nervioso Central. 6, 7
- SNP** Sistema Nervioso Periférico. 6
- SPECT** *Single-Photon Emission Computed Tomography*. 11
- SR** superresolución. 3, 17–19, 32, 34, 35, 40–45, 47–53, 55–57, 59–65, 69–71, 73, 74
- SSIM** *Structural Similarity*. 31, 49, 50, 55–65
- SVM** *Support Vectors Machines*. 33

# Índice general

Agradecimientos y dedicatoria	III
Resumen	V
Resum	VII
Abstract	IX
Lista de acrónimos	XVII
Índice general	XIX
I Memoria	1
1 Introducción	3
1.1 Enfermedad de Alzheimer	3
1.1.1 Aspectos generales	3
1.1.2 Contexto anatómico	5
1.1.3 Anatomía patológica	7
1.1.4 Cuadro clínico y fases	9
1.1.5 Diagnóstico y tratamiento	9
1.2 Imagen de Resonancia Magnética	12
1.2.1 Fundamentos físicos	12
1.2.2 Potenciones de la IRM	15
1.2.3 IRM y Alzheimer	17
1.3 Superresolución	17
1.4 <i>Deep Learning</i> y redes neuronales artificiales	19
1.4.1 Aspectos generales del <i>Deep Learning</i>	19
1.4.2 Redes neuronales artificiales. Fundamentos y arquitecturas	20
1.4.3 Entrenamiento de las ANN	25

1.4.4 Evaluación de las ANN . . . . .	30
1.5 Estado del arte . . . . .	32
1.6 Motivación y justificación . . . . .	33
<b>2 Objetivos</b>	<b>35</b>
2.1 Objetivo general . . . . .	35
2.2 Objetivos específicos . . . . .	35
<b>3 Materiales y métodos</b>	<b>37</b>
3.1 Materiales . . . . .	37
3.1.1 Datos . . . . .	37
3.1.2 <i>Hardware</i> . . . . .	38
3.1.3 <i>Software</i> . . . . .	38
3.2 Esquema metodológico general . . . . .	40
3.3 Arquitecturas propuestas de las redes neuronales artificiales . . . . .	41
3.3.1 Fase de SR . . . . .	41
3.3.2 Fase de clasificación . . . . .	43
3.4 Preprocesado y estructuración de los datos . . . . .	44
3.4.1 Fase de SR . . . . .	45
3.4.2 Fase de clasificación . . . . .	47
3.5 Entrenamiento, validación y evaluación de las redes neuronales artificiales . . . . .	48
3.5.1 Fase de SR . . . . .	48
3.5.2 Fase de clasificación . . . . .	51
<b>4 Resultados y discusión</b>	<b>55</b>
4.1 Fase de SR . . . . .	55
4.1.1 Red T1 SR . . . . .	56
4.1.2 Red de síntesis T2 SR . . . . .	60
4.2 Fase de clasificación . . . . .	65
<b>5 Conclusiones y líneas futuras</b>	<b>73</b>
<b>II Presupuesto</b>	<b>75</b>
<b>Bibliografía</b>	<b>81</b>

Parte I

**Memoria**



## Capítulo 1

# Introducción

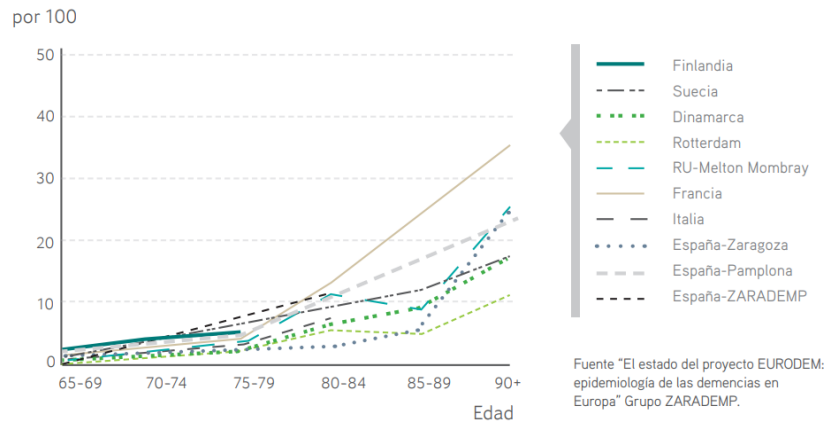
*En este capítulo introductorio se expone el marco teórico a partir del cual se contextualizará y justificará el proyecto. En primer lugar, se desarrollarán los aspectos más relevantes de la enfermedad de Alzheimer (EA) para este trabajo. A continuación, se tratará la imagen de Resonancia Magnética (IRM) como técnica de imagen médica y prueba diagnóstica de la EA, así como las técnicas de superresolución (SR) empleadas para la mejora de la calidad de la imagen. Posteriormente, se introducirá el aprendizaje automático profundo o Deep Learning (DL) y se desarrollarán en profundidad las redes neuronales artificiales. Finalmente, una vez tratados los aspectos teóricos, se resumirá brevemente el estado del arte sobre el tema propuesto y se expondrá la motivación y la justificación del proyecto.*

## 1.1 Enfermedad de Alzheimer

### 1.1.1 Aspectos generales

La enfermedad de Alzheimer (EA) es una enfermedad neurodegenerativa que se caracteriza principalmente por la pérdida progresiva de las capacidades cognitivas, especialmente la pérdida de memoria a corto plazo, y por la aparición de trastornos conductuales en el paciente. La EA fue estudiada por primera vez en el año 1906 por el neurólogo y psiquiatra alemán Alois Alzheimer, quien describió los síntomas característicos de la enfermedad y realizó un estudio del cerebro *post-mortem* en una de sus pacientes.

Actualmente, la EA es una enfermedad incurable, puesto que aún no se conocen las causas que la provocan. Se estima que el pronóstico medio es de 7 a 10 años tras el diagnóstico, aunque depende en gran medida del paciente y de la fase en la que se encuentre la enfermedad. Hasta el momento, se han planteado diferentes hipótesis acerca del origen etiológico de la EA, aunque ninguna de ellas ha demostrado ser aún la definitiva. La teoría actualmente más aceptada propone que la causa de la enfermedad se debería a la alteración de determinadas proteínas (priones) y su acumulación en el cerebro (Kellett y Hooper 2009). Sin embargo, a pesar de desconocer la etiología de la enfermedad, existen numerosos estudios acerca de los diferentes factores de riesgo



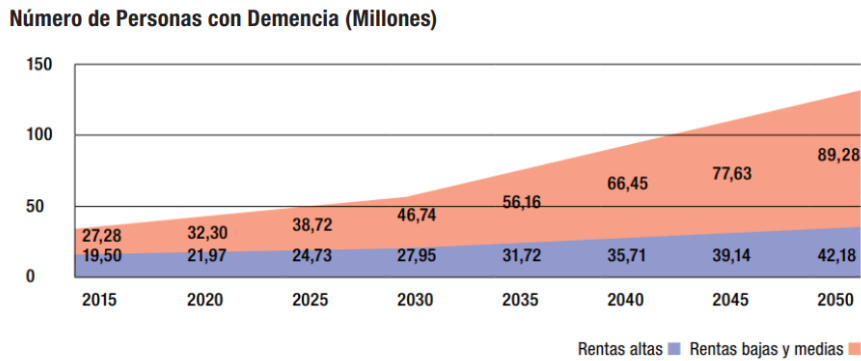
**Figura 1.1: Prevalencia de la EA en diferentes grupos de edad y en diferentes países europeos.** Se observa un incremento exponencial de la prevalencia en grupos de mayor edad en todos los casos. Fuente: Confederación Española de Alzheimer (CEAFA) 2014.

que pueden provocar la aparición de la EA. Estos estudios coinciden en que los factores de riesgo más destacados son (Acosta y col. 2012 y Barranco-Quintana y col. 2005):

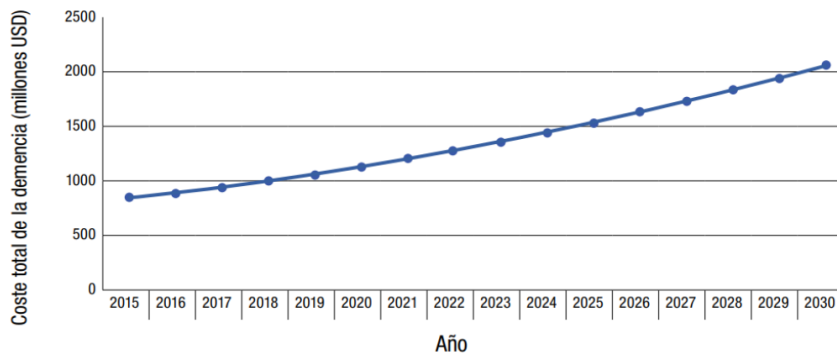
- Edad
- Sexo
- Ambiente
- Genética
- Tabaquismo y alcoholismo
- Obesidad
- Dieta

De las enfermedades consideradas como demencias, la EA es la más frecuente en la población. Numerosos estudios indican que, a partir de los 65 años de edad, la prevalencia de la enfermedad se incrementa exponencialmente (Figura 1.1). A causa del envejecimiento de la población mundial, la tendencia actual apunta a que la prevalencia global de la EA continuará aumentando en los próximos años (Niu y col. 2017). Debe considerarse además que la falta de pruebas de detección precoz de la enfermedad, unida al aumento de la población envejecida y de riesgo, implicará un mayor coste sanitario (López-Pousa y col. 2004). Según datos presentados en el Informe Mundial sobre el Alzheimer 2018 (Alzheimer's Disease International (ADI) 2018), se ha estimado que el número de personas con EA se incrementará de 50 millones a 152 millones, triplicando el número de casos, y que el coste de la enfermedad aumentará de 1 billón de dólares USD estimados en el año 2018 a 2 billones, duplicándolo. Las previsiones de la prevalencia y del coste total de la EA elaboradas en el año 2015 y presentadas en el Informe Mundial sobre el Alzheimer 2015 (Alzheimer's Disease International (ADI) 2015) se muestran en las Figuras 1.2 y 1.3, respectivamente.





**Figura 1.2: Previsión de la prevalencia de la EA (en millones de personas) entre los años 2015-2050.** Se observa un aumento significativo de la prevalencia a nivel global. Fuente: Alzheimer’s Disease International (ADI) 2015.



**Figura 1.3: Previsión del coste de la EA (en millones de dólares USD) entre los años 2015-2050.** Se observa un aumento significativo del coste a nivel global. Fuente: Alzheimer’s Disease International (ADI) 2015.

Además del impacto económico, la EA supone una considerable carga a nivel social. A medida que la enfermedad avanza y las funciones cognitivas se deterioran, el paciente sufre una pérdida de independencia que afecta de forma directa a la familia y, de forma indirecta, a la sociedad. Ante este hecho, la prevención, investigación, difusión, y educación en el cuidado y manejo de los enfermos son las mejores iniciativas a seguir para minimizar el impacto económico y social.

### 1.1.2 Contexto anatómico

El Sistema Nervioso (SN) del cuerpo humano es el conjunto de estructuras y órganos encargados del procesamiento de la información y del control de las funciones y acciones del organismo.

A nivel microscópico, la unidad básica del SN es la neurona. Las neuronas son células especializadas en la transmisión de impulsos nerviosos. Aunque la morfología de las neuronas (Figura 1.4) puede ser variable, todas ellas tienen la capacidad de conectarse con el resto de neuronas del entorno mediante conexiones o uniones sinápticas, y de este modo, se transmiten entre ellas los impulsos eléctricos generados a causa de los estímulos que recibe el organismo. La transmisión del impulso eléctrico entre dos neuronas se produce entre los botones terminales de una neurona y las dendritas de la siguiente, en un espacio denominado espacio o hendidura sináptica. Cuan-

do la neurona recibe un estímulo, se produce una excitación que se transmite por la membrana celular desde las dendritas de la neurona hasta los botones terminales presinápticos de la cola, pasando por el axón. Cabe distinguir entre dos tipos de neuronas según la estructura de su axón:

- **Neuronas mielínicas:** presentan vainas de mielina en su axón, separadas por unos espacios sin mielina denominados nódulos de Rainvier. La mielina impide el intercambio iónico en la membrana, por lo que la transmisión eléctrica del impulso tan solo se producirá en los nódulos. La transmisión del impulso en las neuronas mielínicas se produce de forma mucho más rápida, aunque atenuada. La acumulación de neuronas mielínicas es denominada *sustancia blanca*.
- **Neuronas amielínicas:** presentan un axón sin vainas de mielina. La transmisión eléctrica se realizará en toda la membrana del axón, de modo que la velocidad de transmisión en estas neuronas será menor, pero no sufrirá atenuación. La acumulación de neuronas amielínicas es denominada *sustancia gris*.

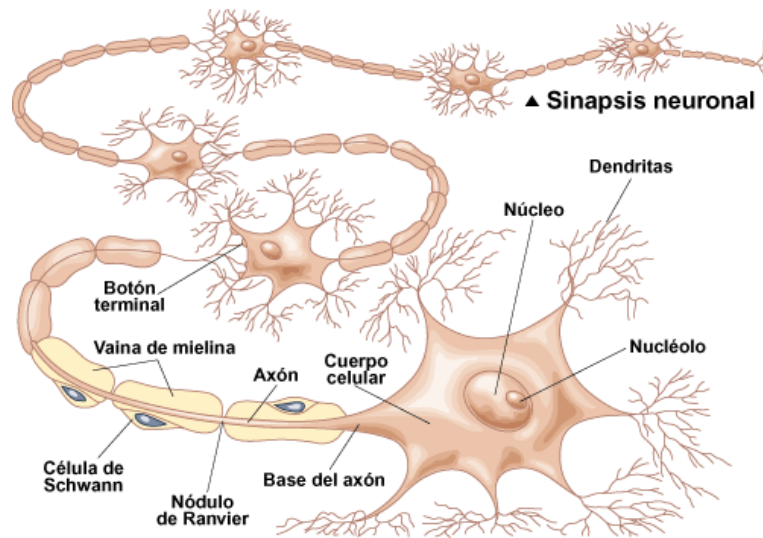


Figura 1.4: Morfología celular de la neurona. Fuente: *Future Website of Anatomia*.

La organización de las neuronas en fibras cada vez mayores da lugar a la formación de estructuras conductoras denominadas fibras nerviosas o nervios, que a su vez constituyen estructuras y órganos más complejos.

A nivel macroscópico, el SN se puede dividir en dos subsistemas:

- **Sistema Nervioso Central (SNC):** formado por el encéfalo y la médula espinal (Figura 1.5). Recibe y procesa la información de los estímulos y da la una respuesta.
- **Sistema Nervioso Periférico (SNP):** formado por los nervios periféricos sensitivos y motores. Conecta el SNC con los órganos receptores de estímulos y efectores del organismo.

A su vez, el SNC del ser humano consta de dos partes:

- **Encéfalo:** se localiza en la zona de la cabeza, protegido por los huesos del cráneo y por el líquido cefalorraquídeo. Se compone de tres estructuras:
  - **Cerebro:** consta de dos hemisferios (derecho e izquierdo), separados por la cisura interhemisférica, y de cuatro lóbulos (temporal, occipital, frontal y parietal) separados

por otras cisuras. La corteza cerebral presenta pliegues o circunvoluciones, y se compone de materia gris a nivel superficial y de materia blanca subyacente. En la región profunda del cerebro se localizan determinados núcleos de sustancia gris: diencefalo (tálamo e hipotálamo), núcleos basales, hipocampo, cuerpo calloso y cápsula interna. A nivel funcional, el cerebro es el responsable la mayor parte de la actividad del sistema nervioso y de las principales funciones del ser humano: pensamiento, emociones, habla, lenguaje, memoria...

- **Cerebelo:** se localiza en la zona basal posterior del encéfalo, bajo el cerebro. Sus principales funciones son la coordinación de movimientos del sistema musculoesquelético.
- **Tallo cerebral:** está formado por el mesencéfalo, la protuberancia anular y el bulbo raquídeo. Sus principales funciones son la regulación del ritmo cardíaco y de la ventilación pulmonar.
- **Médula espinal:** extensión del encéfalo, que se prolonga por la columna vertebral. Su exterior se compone de sustancia gris, mientras que su interior presenta sustancia blanca.

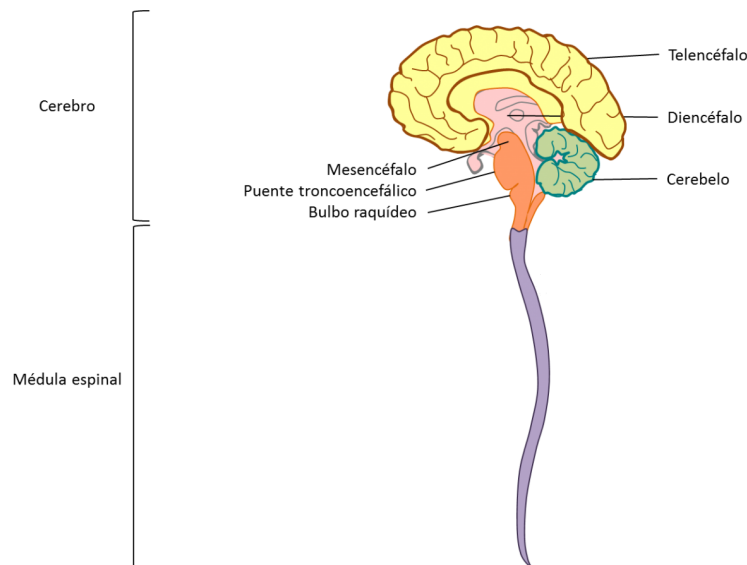


Figura 1.5: Esquema del Sistema Nervioso Central y sus estructuras. Fuente: *Células gliales*.

### 1.1.3 Anatomía patológica

En la enfermedad de Alzheimer, el cerebro es la estructura del Sistema Nervioso que resulta más afectada. Desde un punto de vista bioquímico, en la EA intervienen dos proteínas fundamentales:

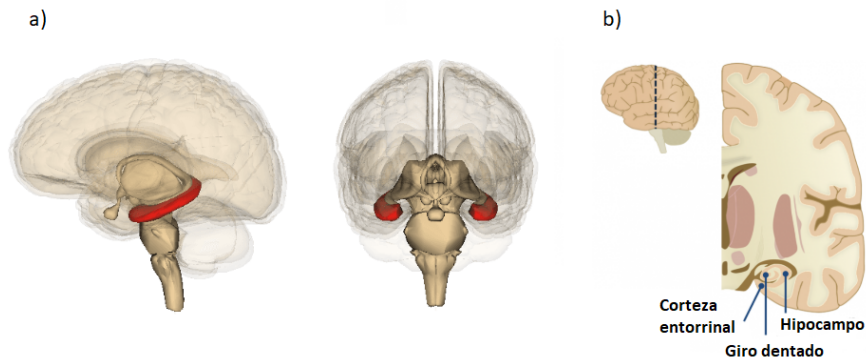
- **Péptido  $\beta$ -amiloide:** en la EA se produce una acumulación anómala de péptido  $\beta$ -amiloide a causa de una alteración en la ruta metabólica de la escisión de su molécula precursora: la proteína precursora amiloidea o *Amyloid Precursor Protein* (APP). Si la escisión de la APP se produce por las enzimas  $\beta$ -secretasa y  $\gamma$ -secretasa (y no por la  $\alpha$ -secretasa), el resultado de la reacción da lugar al péptido  $\beta$ -amiloide, el cual es más fibrogénico y, por lo tanto insoluble. Al no poder ser eliminado, el péptido se acumula en el exterior de las neuronas formando los depósitos extracelulares, que lesionan a las neuronas y provocan inflamación.

- **Proteína  $\tau$** : es la proteína encargada de la estabilización del citoesqueleto celular, concretamente de los microtúbulos. En la EA, la fosforilación de la proteína  $\tau$  provoca una alteración en la estructura celular y, en consecuencia, una pérdida de neuronas funcionales y de conexiones sinápticas.

A nivel histopatológico, se observan dos lesiones importantes en el cerebro de un paciente con EA:

- **Placas seniles, neuríticas o amiloides**: son conglomerados extracelulares de neuronas y axones amorfos, unidas por el péptido  $\beta$ -amiloide.
- **Ovillos neurofibrilares**: son conglomerados proteicos intracelulares formados por manchas proteínas  $\tau$  fosforiladas, también conocidas como Filamentos Helicoidales Pareados.

A nivel macroscópico, la EA se caracteriza por una atrofia severa, progresiva y difusa de la masa cerebral, la cual se inicia en la región temporal mesencefálica, y acaba extendiéndose hasta el córtex cerebral, afectando especialmente a las regiones frontal, temporal y parietal en las más fases avanzadas. Estudios apuntan a que las primeras estructuras que se ven afectadas por el Alzheimer son el hipocampo o *hyppocampus* (HC) y la corteza entorrinal o *entorhinal cortex* (EC) (Apostolova y col. 2006). Ambas estructuras se localizan en la región mesial del encéfalo, tal y como se muestra en la Figura 1.6, y sus funciones más destacas se encuentran asociadas a la memoria (formación de nuevos recuerdos, memoria autobiográfica, etc.) y a la orientación espacial. La atrofia de ambas estructuras en las fases iniciales es la que provoca los síntomas más característicos de la EA: la pérdida de memoria y la desorientación espacial.

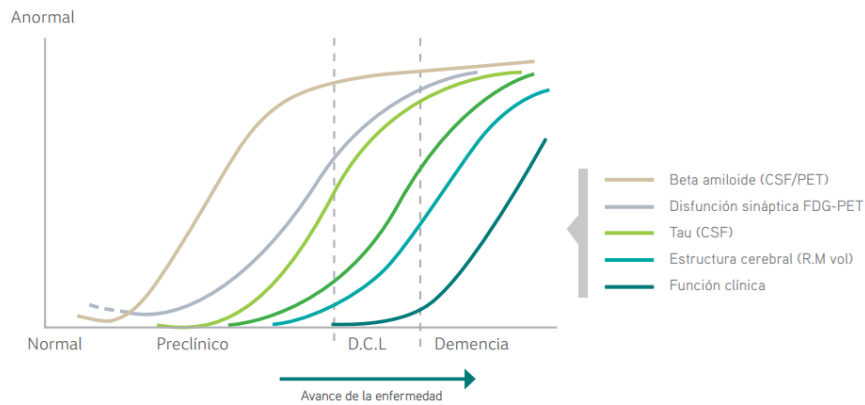


**Figura 1.6: Estructuras del HC y la EC.** a) Vista 3D sagital (izquierda) y coronal posterior (derecha) de la localización del HC en el encéfalo. b) Corte axial del hemisferio izquierdo del cerebro, en el que se aprecian el HC y la EC. Fuentes: *Lifeder.com* y *Asociación Educar*.

#### 1.1.4 Cuadro clínico y fases

Actualmente, se distinguen dos estadios en la EA (Figura 1.7):

- **Fase preclínica:** o presintomática. En esta fase, la enfermedad comienza a desarrollarse a nivel bioquímico y a generar alteraciones cerebrales, pero el paciente no presenta síntomas.
- **Fase clínica:** en esta fase, el paciente manifiesta los síntomas propios de la demencia. Viene precedida por una **fase prodrómica** o de predemencia, en la que el paciente experimenta Deterioro Cognitivo Leve (DCL).



**Figura 1.7: Evolución de los parámetros en las diferentes fases la EA.** En la fase preclínica puede apreciarse como determinados biomarcadores propios de la EA presentan niveles anormales. Fuente: Confederación Española de Alzheimer (CEAFA) 2014.

Una vez el paciente ha entrado en la fase clínica, el diagnóstico permite clasificar la enfermedad en tres etapas, según la severidad del cuadro clínico que presente el sujeto. En la Tabla 1.1 se resumen los cuadros clínicos para cada etapa de la EA en su fase clínica. Se puede observar que, a medida que los signos y síntomas propios de la demencia se agravan, la independencia del paciente queda drásticamente reducida a causa de la progresiva pérdida de las facultades cognitivas e intelectuales.

#### 1.1.5 Diagnóstico y tratamiento

El diagnóstico de la EA en la actualidad es un diagnóstico diferencial por exclusión, es decir, la enfermedad se diagnostica si los síntomas se corresponden con el cuadro clínico y si no existe un diagnóstico alternativo. Las herramientas que actualmente se emplean en el diagnóstico de la EA son:

- **Anamnesis:** entrevista clínica con el paciente, en la que se buscan síntomas de deterioro cognitivo o pérdida de memoria. También se investigan antecedentes familiares, medicación, dieta, estilo de vida...
- **Exploración neuropsicológica:** se realizan una serie de tests o pruebas psicométricas para valorar las capacidades cognitivas del paciente a nivel global o para analizar funciones cognitivas específicas.

Fase	Período	Cuadro clínico
Demencia leve	3-4 años	Disminución de la memoria a corto plazo Desorientación espacio-temporal leve Disminución de la capacidad de juicio Problemas con el pensamiento abstracto Problemas leves de lenguaje Extravío de objetos Problemas en la ejecución de tareas sencillas Cambios de personalidad Cambios anímicos o conductuales (nervios, depresión...) Falta de iniciativa
Demencia moderada	3-4 años	Alteraciones cognitivas evidentes Fallo de memoria a corto plazo (solo recuerdos pasados) Desorientación espacio-temporal grave Falta de consciencia de los errores Pensamiento abstracto y razonamiento imposible Problemas graves de lenguaje Fallos en la ejecución de tareas sencillas Transtornos conductuales (delirio, insomnio, agitación...) Pérdida de control de esfínteres ocasional
Demencia grave	1-2 años	Pérdida absoluta de las capacidades cognitivas e intelectuales Rigidez muscular Ausencia de control de esfínteres

**Tabla 1.1: Etapas de la EA, períodos y cuadros clínicos.** Se observa en los cuadro clínicos, a medida que avanza la enfermedad, la demencia y la gravedad de los síntomas van en aumento.

- **Estudio analítico completo:** análisis de sangre, hormonas, glucemia... Se lleva a cabo para descartar otras posibilidades.
- **Estudio físico y neurológico:** estudio de otros sistemas (cardiovascular, renal...) para descartar diagnósticos alternativos.
- **Valoración funcional:** permite determinar el nivel de autonomía del paciente en su vida diaria. Se realiza por medio de una serie de cuestionarios y escalas.
- **Pruebas de neuroimagen:** para el diagnóstico de la EA se emplean tres modalidades de imagen médica:
  - Tomografía computerizada o *Computed Tomography* (CT): se trata de una imagen anatómica que se obtiene por proyección de radiación gamma (fotones), concretamente rayos X, sobre el paciente. Permite descartar otras patologías no demenciales, como tumores, hematomas...
  - Tomografía por emisión de positrones o *Positron Emission Tomography* (PET): se trata de una imagen funcional, que se obtiene al detectar la radiación endógena de positrones que emite un radionucleido unido a una biomolécula inyectado en el paciente, por ejemplo, fluor unido a fluorodeoxiglucosa (18F-FDG). Con esta técnica, se puede

evaluar la actividad metabólica en el cerebro. En el caso de la EA, la imagen PET registra hipometabolismo en la región temporoparietal. También es posible combinar con otras modalidades de imagen anatómica, dando lugar a técnicas mixtas, como por ejemplo la PET/CT.

- Tomografía computarizada de emisión monofotónica o *Single-Photon Emission Computed Tomography* (SPECT): se trata de una imagen funcional que se obtiene al detectar la radiación gamma que emite un radionucleido (usualmente tecnecio (Tc), yodo (I), etc.) inyectado en el paciente. Permite evaluar la perfusión cerebral o el metabolismo de la glucosa, entre otros marcadores diagnósticos.
- Imagen de resonancia magnética (IRM): en la IRM existen múltiples modalidades que permiten evaluar tanto la anatomía como la función cerebral, y detectar deterioro cognitivo leve, atrofia... Esta técnica de imagen, en la que se basa el proyecto, se desarrolla en profundidad en la Sección 1.2.

Al tratarse de un diagnóstico por exclusión, es necesario definir una serie de criterios a partir de los cuales determinar si un paciente sufre EA. Inicialmente, se empleaba el criterio NINCDS-ADRDA, pero actualmente están siendo sustituidos por nuevos criterios, como los DSM-IV-TR o los NIA-AA (Confederación Española de Alzheimer (CEAFA) 2014). La aplicación de los criterios y protocolos es fundamental para poder elaborar un diagnóstico adecuado y determinar el estado de la enfermedad con precisión.

En todas las técnicas actuales de diagnóstico de una enfermedad se emplean una serie de indicadores denominados biomarcadores. Un biomarcador se define como una característica objetivamente evaluable y que aporta información acerca del estado de un proceso fisiológico, patológico o de respuesta a un fármaco o tratamiento. En el caso de la EA, existen biomarcadores genéticos, como determinados alelos o mutaciones del genoma relacionados con la enfermedad, o bioquímicos, como la cantidad de líquido cefalorraquídeo, radiofármacos afines a las proteínas  $\beta$ -amiloides y  $\tau$  empleados en técnicas funcionales de neuroimagen (F18-Florbetapir, FDDNP, compuesto Pittsburgh B...), etc. La investigación actual acerca de nuevas pruebas diagnósticas y biomarcadores de la EA se centra principalmente en las pruebas de imagen médica y el procesamiento de señales biomédicas para conseguir un diagnóstico precoz. Ejemplos de ello son el descubrimiento de nuevos radiofármacos empleados en técnicas funcionales de neuroimagen, como el compuesto Pittsburgh B, el procesamiento y análisis de señales electroencefalográficas, o el desarrollo de nuevas técnicas de mejora de la calidad de la imagen y clasificación basadas en aprendizaje automático.

En lo referente al tratamiento de la EA, debe tenerse en cuenta que la enfermedad es incurable hoy en día, por lo que la acción terapéutica se enfoca al tratamiento de los síntomas y al retraso del avance de la enfermedad. Distinguimos dos tipos de tratamientos:

- **Farmacológico:** el tratamiento por fármacos permite tratar ciertos síntomas de la enfermedad.
  - Inhibidores de colinesterasa (donepezilo, rivastigmina, galantamina): la enzima colinesterasa retrasa la degradación de la acetilcolina, ya que se trata de un neurotransmisor que presenta niveles bajos en pacientes con EA.

- Antagonistas no competitivos de los receptores de N-metil D-Aspartato glutaminérgicos (memantina): impiden una excesiva neuroestimulación por parte del glutamato (neurotransmisor excitatorio), la cual podría provocar reacciones tóxicas en las neuronas. Se emplean en las fases moderada y grave de la EA, y también en otro tipo de enfermedades neurodegenerativas.
- **No farmacológico:** recientemente se ha demostrado la eficacia y los beneficios de tratamientos no farmacológicos para la EA, como las sesiones de estimulación cognitiva, los cuidados, el apoyo familiar y social...

Actualmente, existen otros tratamientos en fase de investigación, como las técnicas de ultrasonidos o tratamientos neurorregenerativos con células madre.

## 1.2 Imagen de Resonancia Magnética

La imagen de Resonancia Magnética (IRM) es una técnica de imagen médica basada en el fenómeno físico de la Resonancia Magnética (RM). A diferencia de otras técnicas que emplean radiación ionizante, la IRM se basa en la aplicación de un campo magnético y señales de radiofrecuencia (RF) no ionizantes sobre el paciente, por lo que se trata de una técnica inócua. Además, la IRM permite obtener una gran cantidad de información anatómica y funcional del paciente gracias a sus múltiples variantes. Todo ello, junto con las posibilidades que ofrece actualmente el procesamiento computacional de las imágenes médicas, hacen de la IRM una herramienta de diagnóstico muy extendida en numerosas especialidades médicas.

### 1.2.1 Fundamentos físicos

La IRM se basa en el fenómeno de la Resonancia Magnética Nuclear o Resonancia Magnética (RM), descubierto en 1938 por Isidor Isaac Rabi. El principio de la RM enuncia que, cuando se coloca un átomo de un elemento de número atómico  $Z$  impar bajo un campo magnético  $\vec{B}$ , éste es capaz de absorber energía a una determinada frecuencia, denominada frecuencia de resonancia. En el caso de la IRM, la técnica se basa en las propiedades del átomo o núcleo de hidrógeno  $H^1$  ( $Z = 1$ ).

Desde el punto de vista cuántico, el núcleo de  $H^1$  presenta dos estados energéticos u orientaciones cuando se encuentra bajo un campo magnético  $\vec{B}$ : una posición de mínima energía (estado *up* o paralelo a  $\vec{B}$ ), y una posición de máxima energía (estado *down* o antiparalelo a  $\vec{B}$ ). La orientación del espín del núcleo con respecto a la dirección del campo magnético viene fijada por la mecánica cuántica y es de  $54,7^\circ$ . Como consecuencia de esta restricción que le impide al núcleo alinearse con la dirección del campo, aparece un momento magnético  $\vec{\mu}$  que genera un movimiento de precesión en torno al eje de la dirección de  $\vec{B}$  con una frecuencia de giro o frecuencia de precesión  $f_p$  (Figura 1.8). Dicha frecuencia  $f_p$ , también conocida como frecuencia de resonancia, es directamente proporcional a la magnitud  $B$  de campo magnético, y se relacionan por la ley fundamental de la Resonancia Magnética o ley de Larmor:

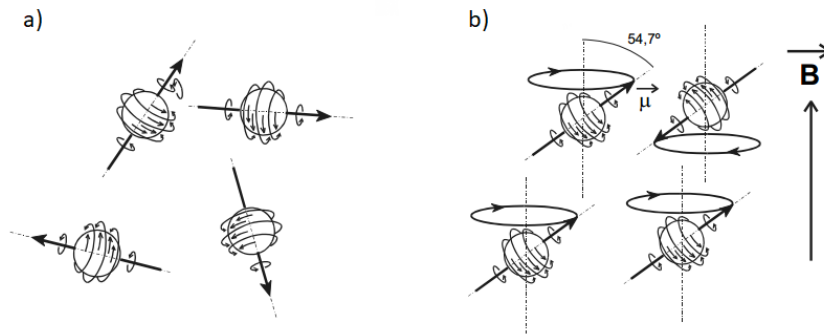
$$f_p = \frac{\gamma B}{2\pi} \text{ (Hz)} \quad (1.1)$$

---

<sup>1</sup>Las unidades de campo magnético se expresan en Gauss (G) o Teslas (T)



Donde  $\gamma$  es la constante giromagnética nuclear, la cual depende del átomo. En el caso del átomo de hidrógeno:  $\gamma_{H^1} = 42,57 \text{ MHz} \cdot \text{T}^{-1}$ .



**Figura 1.8: Orientación de los espines de los núcleos  $H^1$  en presencia y ausencia del campo magnético  $\vec{B}$ .** a) Orientación al azar de los núcleos en ausencia de campo magnético  $\vec{B}$ . b) Orientación ordenada con movimiento de precesión de los núcleos en los dos estados (*up* y *down*) en presencia de campo magnético  $\vec{B}$ . Fuente: Gili 1993.

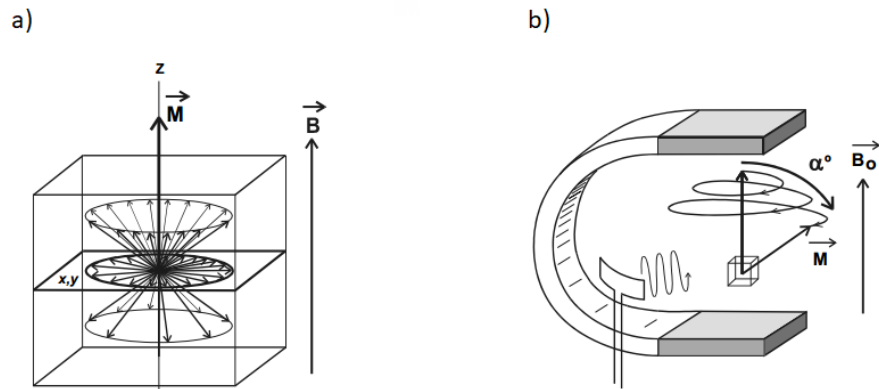
En la técnica IRM, el paciente se introduce en el interior de un potente campo magnético  $\vec{B}_0$  generado por un imán, cuyo rango es de 1.5 a 7 T, según la aplicación. El volumen del paciente se discretiza en pequeñas unidades volumétricas denominadas vóxeles. En un vóxel en estado de reposo, los núcleos  $H^1$  que contiene se encuentran afectados por un campo magnético  $\vec{B}_0$  equivalente a la suma de tres campos magnéticos:

$$\vec{B} = \vec{B}_0 + \vec{B}_G + \vec{B}_Q \quad (1.2)$$

Donde  $\vec{B}_0$  es el campo del imán,  $\vec{B}_G$  es el campo de gradiente, que permite diferenciar a los núcleos según su posición espacial, y  $\vec{B}_Q$  es el campo bioquímico propio del tejido debido a su composición bioquímica. En cuanto a su estado energético, la mayor parte de los núcleos se encuentran en un estado de reposo o mínima energía (estado *up*).

Si se calcula la suma vectorial de todos los momentos  $\vec{\mu}$  de los espines de los núcleos  $H^1$  de un vóxel, se obtiene un vector resultante denominado vector de magnetización  $\vec{M}$ . Como en estado de reposo bajo un campo magnético los espines precesan a frecuencias ligeramente distintas a causa de las pequeñas diferencias del campo magnético bioquímico  $\vec{B}_Q$  de los diferentes tejidos, éstos se encuentran ligeramente desfasados entre ellos. En consecuencia, la proyección resultante del vector  $\vec{M}$  en el plano transversal del vóxel será nula, y su sentido y dirección serán las del campo magnético  $\vec{B}$  (Figura 1.9a). En cuanto a la magnitud del vector  $\vec{M}$ , ésta depende de la densidad de núcleos  $H^1$  del vóxel (a más núcleos, mayor magnitud).

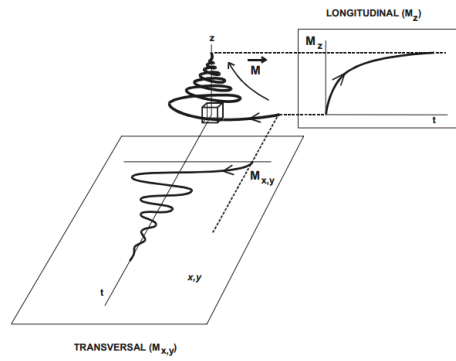
Una vez el paciente se ha situado en el interior del imán, se lanza una señal o pulso radiofrecuencia desde una antena emisora (o bobina de gradiente) en una dirección preferente sobre el cuerpo en reposo del paciente, y en un rango de frecuencias en torno a una frecuencia de precesión  $f_p$ . Como consecuencia de este pulso, los núcleos  $H^1$  que precesen en dicho rango de frecuencias, y que hasta ahora se encontraban desfasados en su movimiento, entran en resonancia y comienzan a precesar a la misma frecuencia  $f_p$ , de modo que la proyección del vector magnetización  $\vec{M}$  deja de ser nula en el plano transversal. A su vez, los núcleos comienzan a excitarse y a pasar del estado *up* de



**Figura 1.9: Vector magnetización  $\vec{M}$  de un vóxel ante la presencia de un campo magnético  $\vec{B}$  en estado de reposo y excitación.** a) El vector  $\vec{M}$  en estado de reposo equivale a la suma vectorial de los momentos  $\vec{\mu}$  de los núcleos  $H^1$  del vóxel. b) Movimiento de nutación durante la excitación del vector  $\vec{M}$  mediante un pulso RF y con una separación respecto a la posición de equilibrio  $\alpha$ . Fuente: Gili 1993.

mínima energía al estado *down* de máxima energía, de forma que la componente longitudinal del vector se reduce. Desde una perspectiva espacial, el vector  $\vec{M}$  describe un movimiento espacial en espiral o de nutación, desviándose en un ángulo  $\alpha$  de la posición de equilibrio (Figura 1.9b). Es importante destacar que el campo magnético  $\vec{B}$  cambia en el espacio a causa del campo de gradiente  $\vec{B}_G$ , por lo que solo aquellos núcleos con la frecuencia  $f_p$  que se encuentren en una posición espacial concreta entrarán en excitación. De este modo, variando la frecuencia del pulso RF, se irá excitando al paciente en las distintas posiciones espaciales.

Una vez se ha alcanzado el estado de máxima excitación, se deja de emitir el pulso RF, y el vector  $\vec{M}$  entra en fase de relajación nuclear. En esta fase, el vector  $\vec{M}$  recupera su posición inicial a causa del desfase que se produce entre los núcleos por las diferencias del campo bioquímico  $\vec{B}_Q$ , y los núcleos vuelven progresivamente al estado de mínima energía, puesto que dejan de estar excitados (Figura 1.10). Como consecuencia de la liberación de energía, los vóxeles emiten una señal energética RF que es captada por una antena receptora y transformada a señal eléctrica. Dependiendo del campo magnético que afecte al vóxel, las frecuencias de las señales RF emitidas por el paciente varían, por lo que, mediante una secuencia de pulsos de excitación, las señales recibidas pueden codificarse en frecuencia en un espacio matemático denominado *espacio-k*. Posteriormente, gracias a la reconstrucción por la transformada inversa de Fourier, se obtienen los planos que conforman la imagen IRM del paciente.



**Figura 1.10:** Fase de relajación del vector de magnetización  $\vec{M}$  en un pulso RF de  $90^\circ$ . La recuperación en el tiempo de las componentes del vector  $\vec{M}$  en los planos longitudinal ( $\vec{M}_z$ ) y transversal ( $\vec{M}_{xy}$ ) da lugar a diferentes señales y modalidades de imagen IRM. Fuente: Gili 1993.

### 1.2.2 Potenciaciones de la IRM

En la técnica de IRM existen diferentes potenciaciones o modalidades de imagen en función de los parámetros de la fase de relajación que se deseen potenciar. Las modalidades existentes en IRM se emplean en numerosas aplicaciones: análisis las estructuras anatómicas, contraste de tejidos, detección de lesiones, evaluación de funciones (angiografía, tractografía, función cardíaca...), etc. De las modalidades que ofrece la IRM, las más extendidas son las imágenes potenciadas en densidad protónica D, T1 y T2.

En las imágenes potenciadas en densidad protónica D, se estudia la densidad de núcleos  $H^1$  que presenta el tejido. Sabiendo que el módulo del vector magnetización  $\vec{M}$  es mayor cuanto mayor sea la densidad de núcleos  $H^1$  del vóxel, la intensidad  $I_D$  de la imagen potenciada en D se obtiene tomando el valor inicial de la señal de relajación en cada vóxel tras dejar de emitir el pulso de excitación RF. Este valor de intensidad corresponde con el valor máximo del módulo del vector  $\vec{M}$ , tal y como se refleja en la Ecuación (1.3). En consecuencia, cuanta mayor densidad de núcleos, mayor intensidad presentará un vóxel. Sin embargo, debe tenerse en cuenta que el número de núcleos que son resonantes depende de la frecuencia de precesión  $f_p$ , la cual puede variar dependiendo del entorno bioquímico del núcleo (por ejemplo, la frecuencia  $f_p$  es diferente en los radicales  $-CH_3$  y en los radicales  $-OH$ ).

$$I_D = M_{xy}(0) \quad (1.3)$$

En las imágenes potenciadas en T1, se analiza la relajación de la componente longitudinal  $\vec{M}_z$  del vector de magnetización. La recuperación de esta componente viene dada por la función exponencial creciente expresada en la Ecuación (1.4), y su crecimiento está regulado por la constante de tiempo T1 (Figura 1.10). Una constante T1 baja indica que la relajación de la componente longitudinal es rápida, y viceversa. La relajación de la componente longitudinal se encuentra directamente relacionada con la liberación energética de los núcleos  $H^1$  al pasar del estado *down* a *up* y, por consiguiente, con los grados de libertad o la capacidad de movilidad de las moléculas. Cuanta mayor movilidad presente una molécula, mayor facilidad para liberar energía y mayor velocidad en la relajación. En la modalidad T1, la intensidad de la imagen se obtiene con el módulo la componente longitudinal  $M_z(T1)$  en el instante de tiempo  $t = T1$ , que

coincide con la recuperación del 63 % de la componente  $\vec{M}_z$  inicial, tal y como se refleja en la Ecuación (1.5). A partir de lo anterior se deduce que aquellos tejidos cuyos núcleos  $H^1$  presenten una liberación energética más rápida serán hiperintensos (por ejemplo, los lípidos), mientras que los tejidos con menor capacidad de liberación energética serán hipointensos (por ejemplo, el agua libre).

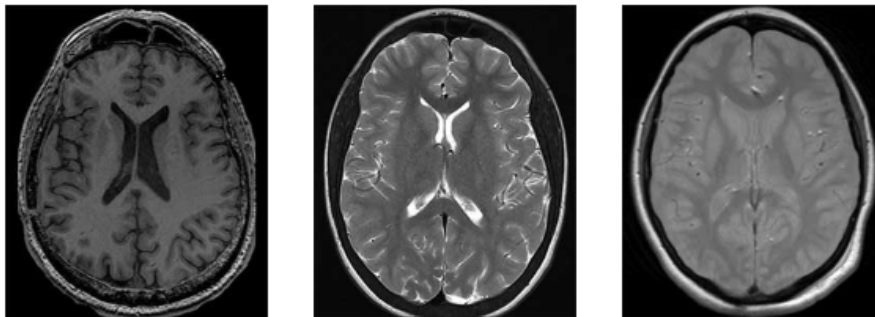
$$M_z(t) = M_z(0) \left(1 - e^{-\frac{t}{T1}}\right) \quad (1.4)$$

$$I_{T1} = M_z(T1) = M_z(0) (1 - e^{-1}) = 0,63M_z(0) \quad (1.5)$$

En cuanto a las imágenes potenciadas en T2, se analiza la recuperación de la componente transversal  $\vec{M}_{xy}$  del vector de magnetización. La señal generada durante la relajación de la componente  $\vec{M}_{xy}$  es una senoidal atenuada, cuya envolvente es la función exponencial decreciente expresada en la Ecuación (1.6), y su velocidad de decrecimiento depende de la constante de tiempo T2 (Figura 1.10). Una constante T2 baja indica que la relajación de la componente transversal es más rápida, y viceversa. La relajación de la componente transversal se encuentra directamente relacionada con el desfase o asincronía en la precesión de los núcleos  $H^1$ . La asincronía de los espines, como se ha comentado anteriormente, se debe a las heterogeneidades de la composición a nivel local en los tejidos: cuanta mayor heterogeneidad presente el tejido, mayor asincronía en los espines, y mayor velocidad en la relajación. En la modalidad T2, la intensidad de la imagen se obtiene con el módulo la componente transversal  $M_{xy}(T2)$  en el instante de tiempo  $t = T2$ , que coincide con la reducción hasta el 37 % de la componente  $\vec{M}_z$  transversal, tal y como se refleja en la Ecuación (1.7). En consecuencia, y por todo lo anterior, se deduce que aquellos tejidos cuyos núcleos  $H^1$  presenten una mayor heterogeneidad de composición serán hiperintensos (por ejemplo, el agua libre), mientras que los tejidos con menor heterogeneidad serán hipointensos (por ejemplo, la sustancia blanca, o el músculo).

$$M_{xy}(t) = M_{xy}(0)e^{-\frac{t}{T2}} \quad (1.6)$$

$$I_{T1} = M_{xy}(T2) = M_{xy}(0)e^{-1} = 0,37M_{xy}(0) \quad (1.7)$$



**Figura 1.11: Modalidades de la IRM.** Cortes axiales del cerebro potenciados en T1 (izquierda), T2 (centro) y densidad protónica D (derecha). Fuente: *Radiology Café*.

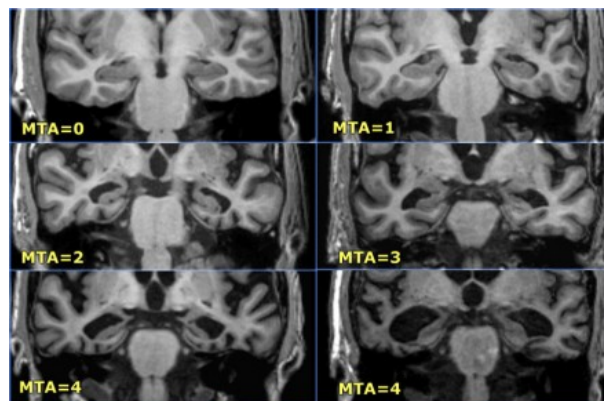
Además de estas modalidades, mostradas en la Figura 1.11, existen muchas otras modalidades que potencian otro tipo de parámetros: T1 con contraste de gadolinio (T1-Gd), T2\*, FLAIR,

imágenes de difusión... Además de las posibilidades mencionadas, la IRM puede combinarse con otras técnicas de imagen, de modo que las posibilidades que ofrece son cada vez mayores.

### 1.2.3 IRM y Alzheimer

Actualmente, la técnica de la IRM es una de las más extendidas en el diagnóstico de la EA. Existen diferentes técnicas tanto anatómicas como funcionales, que emplean la RM para el diagnóstico, como la IRM estructural, IRM funcional, imágenes de difusión y perfusión o la Espectroscopía por Resonancia Magnética (ERM), entre otras.

La técnica más habitual en los estudios diagnósticos es la IRM estructural. Gracias a la información anatómica que se registra en las imágenes IRM volumétricas, es posible evaluar uno de los parámetros fundamentales en la EA: la atrofia de la masa cerebral. La evaluación del grado de atrofia de estructuras clave, como el lóbulo temporal (concretamente el HC y la EC), permiten establecer un diagnóstico de la EA y su estadio con una determinada probabilidad. Actualmente se emplea la escala visual de Scheltens para la valoración de la atrofia del lóbulo temporal basándose en tres estructuras: la cisura coroidea, el asta temporal y el HC. Sin embargo, ya han comenzado a surgir nuevas técnicas de análisis automáticas con numerosas aplicaciones diagnósticas: segmentación de estructuras, evaluación de biomarcadores sobre la imagen, sistemas de clasificación, etc.



**Figura 1.12: Análisis de la atrofia del lóbulo temporal en IRM.** Cortes coronales del cerebro con diferentes niveles de atrofia en la escala visual de Scheltens en el rango de 0 (ausencia de atrofia) a 4 (atrofia severa). Fuente: *Radiología Club*.

## 1.3 Superresolución

Las técnicas de superresolución (SR) son un conjunto de algoritmos que permiten mejorar la resolución espacial de una imagen por encima de la resolución de adquisición. Actualmente, existen dos grupos de técnicas SR con las que se puede lograr la mejora de la resolución:

- **Técnicas multi-imagen:** la imagen SR se obtiene a partir de una secuencia de imágenes de baja resolución o *Low Resolution* (LR) mediante la extracción y fusión de la información común de las imágenes. Sin embargo, la obtención de una serie de imágenes supone un mayor tiempo de adquisición.

- **Técnicas monoimagen:** la imagen SR se obtiene a partir de una única imagen LR. A diferencia de las técnicas tradicionales de interpolación, las técnicas SR asumen que el muestreo de las imágenes no es ideal, es decir, que los valores de baja resolución obtenidos no son ideales o verdaderos. En el proyecto, emplearemos este tipo de técnicas.

Para llevar a cabo la reconstrucción de una imagen de alta resolución o *High Resolution* (HR) a partir de una imagen inicial LR, se parte de la base de que existe una información subyacente HR que, siguiendo un modelo matemático con una serie de restricciones, da lugar a la información LR de la que se dispone en la imagen inicial. A raíz de este planteamiento, en las técnicas de mejora de resolución de la imagen, cada unidad de la imagen original LR puede dividirse en  $N$  subunidades HR, cuyo número viene determinado por un factor de escala  $f$ . Este factor es el que determinará la resolución espacial de la nueva imagen.

$$\text{Imágenes 2D: } N = f^2 \quad (1.8)$$

$$\text{Imágenes 3D: } N = f^3 \quad (1.9)$$

Las nuevas subunidades de la imagen HR, como se ha comentado anteriormente, deben cumplir con un modelo matemático. A la hora de formular dicho modelo, si suponemos como en las técnicas clásicas de interpolación que la adquisición de las imágenes LR se realiza con un muestreo ideal (*ideal sampling*), el valor  $x_{HR}$  de una nueva unidad de la imagen de SR puede estimarse con una función interpoladora (lineal, cuadrática, cúbica...) a partir de los valores  $x_{LR}$  de baja resolución de la imagen inicial. Sin embargo, en la realidad, la adquisición de la información de las imágenes se realiza en presencia de ruido y de otras influencias, por lo que no puede suponerse un muestreo ideal (*non ideal sampling* o *real sampling*). Ante este hecho, los modelos de los algoritmos de SR suponen que el valor  $x_{LR}$  se obtiene al promediar los valores  $x_{HR}$  subyacentes de la imagen, tal y como se muestra en el modelo de la Ecuación (1.10). Además, incluyen en su expresión un factor de influencia o *bias*  $n$  que modela el ruido. De este modo, en las técnicas de SR, la nueva imagen HR estimada sigue un modelo más realista.

$$x_{LR} = \frac{1}{N} \sum_{i=1}^N x_{HR} + n \quad (1.10)$$

Los algoritmos de SR se pueden agrupar en dos grandes familias:

- I) **Algoritmos basados en redundancia de patrones:** estos algoritmos reconstruyen la imagen de SR a partir de la información de regiones o *patches* similares de las imágenes. El proceso de reconstrucción se realiza de forma no local e iterativa, y minimizando una función matemática o función de coste que mide la diferencia entre la imagen reconstruida y la imagen original de SR. Normalmente, este tipo de algoritmos incluyen una serie de restricciones en su modelo matemático (Manjón y col. 2010a). Dos de ellas son:

- **Regularidad:** la condición de regularidad implica que el valor de cualquier unidad HR inferida debe ser coherente con los valores de las unidades HR de su contorno. Matemáticamente:

$$\hat{x}_i^{t+1} = \frac{1}{C_i} \sum_{\forall j \in \Omega} \omega(\hat{x}_i^t, \hat{x}_j^t) \hat{x}_j^t \quad (1.11)$$

Donde  $\hat{x}_i^{t+1}$  es el valor de la unidad HR inferida de la siguiente iteración,  $\hat{x}_i^t$  y  $\hat{x}_j^t$  son los valores de las unidades HR inferida y del contorno de la iteración actual respectivamente,  $C_i$  es el factor de normalización,  $\Omega$  es la región de contorno y  $\omega(\hat{x}_i^t, \hat{x}_j^t)$  es el peso asociado entre las unidades  $\hat{x}_i^t$  y  $\hat{x}_j^t$ , y depende de la distancia entre unidades y la similitud entre valores (a mayor cercanía y similitud, mayor peso asociado tendrá el valor  $\hat{x}_j^t$ ).

- **Consistencia:** la condición de consistencia implica que el valor de las unidades  $x_{LR}$  LR originales equivale al promedio de las subunidades  $\hat{x}_{HR}$  subyacentes estimadas. Matemáticamente:

$$x_{LR} - \frac{1}{N} \sum_{i=1}^N \hat{x}_{HR} = 0 \quad (1.12)$$

Al imponer la condición de consistencia en un algoritmo, se está suponiendo que el ruido  $n$  es despreciable, por lo que antes de iniciar el algoritmo, es fundamental filtrar los datos para reducir al máximo el ruido.

- II) **Algoritmos basados en *sparsity*:** estos algoritmos de SR se basan en la reconstrucción de las imágenes HR a partir de la combinación de los elementos de un diccionario. Este tipo de técnicas calculan a partir de una imagen LR los coeficientes que multiplican a los elementos del diccionario para generar con ellos la imagen HR. Éstos métodos también se basan un algoritmo iterativo que emplea una función de coste con dos términos a minimizar: una distancia (por ejemplo, euclídea) que mide la diferencia entre las imágenes SR estimada y HR de referencia, y un término *sparsity*, que indica la complejidad de la reconstrucción, es decir, el número de elementos empleados por el diccionario.

En los últimos años, debido a los importantes avances en el campo del aprendizaje automático profundo, cada vez son más habituales las técnicas de SR basadas en las redes neuronales artificiales, el cual se desarrolla en mayor profundidad en la Sección 1.4.

## 1.4 *Deep Learning* y redes neuronales artificiales

### 1.4.1 Aspectos generales del *Deep Learning*

El aprendizaje automático profundo o *Deep Learning* (DL) es un conjunto de técnicas y algoritmos cuyo objetivo es la obtención de modelos abstractos que sean capaces de reconocer y extraer características de los datos mediante transformaciones matemáticas no lineales. Las técnicas de DL se enmarcan dentro del campo del aprendizaje automático o *machine learning*. El aprendizaje automático aplicado al campo de la computación consiste en la modificación de un sistema para que éste puedan llevar a cabo una tarea concreta de forma eficiente. Según la forma de adquisición de conocimiento y perfeccionamiento de la tarea del sistema, las técnicas de aprendizaje automático se pueden agrupar en dos grandes familias:

- **Aprendizaje no supervisado:** los sistemas aprenden a partir de los datos, sin ofrecerles la respuesta. Durante el proceso de entrenamiento, los algoritmos de aprendizaje no

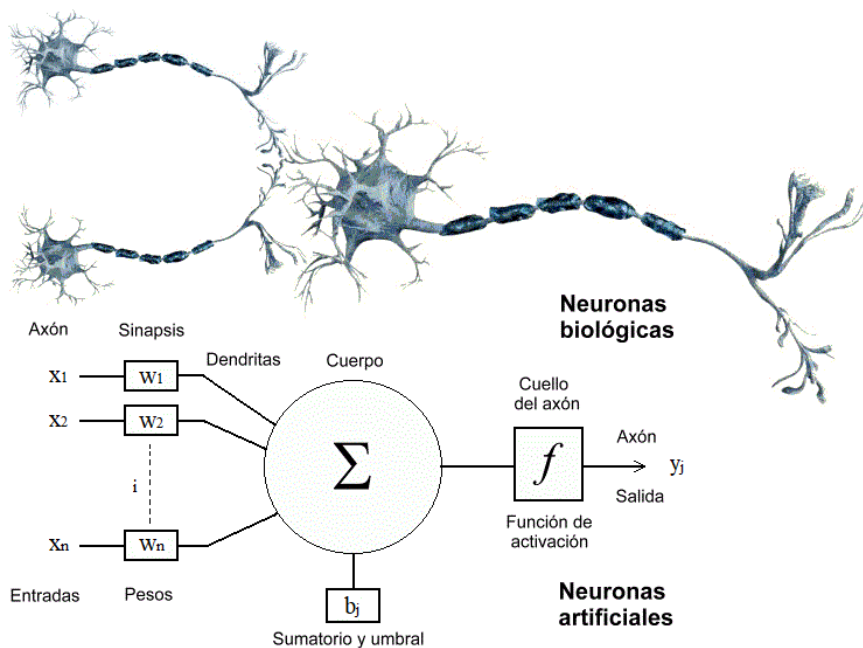
supervisado se adaptan las características propias de los datos, y calculan la solución al problema en base a un determinado criterio.

- **Aprendizaje supervisado:** los sistemas aprenden a resolver un problema a partir de pares dato-solución. Al dar la respuesta para cada caso, se fuerza a que el sistema aprenda una tarea concreta. El DL pertenece a esta familia de técnicas.

Los algoritmos de DL requieren un gran coste computacional a la hora de ejecutarlos. En sus inicios, se desarrollaron de forma teórica, ya que no se disponía de los medios necesarios para su ejecución en un tiempo razonable. Sin embargo, gracias a los avances en el campo de la computación en los últimos años, las técnicas de aprendizaje automático se han expandido en múltiples áreas científicas, adaptando los algoritmos a los diferentes problemas que en ellas se presentan.

### 1.4.2 Redes neuronales artificiales. Fundamentos y arquitecturas

Dentro de los algoritmos de DL, se encuentran las denominadas Redes Neuronales Artificiales o ANN. Como su propio nombre indica, la arquitectura de las ANN se inspira en las redes que forman las neuronas en el Sistema Nervioso (SN) de los animales (Figura 1.13).



**Figura 1.13:** Comparación de la neurona biológica con la neurona artificial de una ANN. El valor de salida de la neurona depende de la suma ponderada de las entradas y de un factor *bias*. Fuentes: *Dpto. de Ciencias de la Computación e Inteligencia Artificial. Universidad de Sevilla* y *El Blog Verde*.

La unidad fundamental de las ANN, al igual que ocurre en el SN, es la neurona o perceptrón simple. En una neurona del SN, la señal o impulso eléctrico de salida depende de los impulsos de entrada de las neuronas conectadas sinápticamente, que pueden activar o inhibir el impulso. De forma análoga, el valor de salida de las neuronas de una ANN depende del valor de entrada de las neuronas conectadas multiplicadas por un peso, que puede incrementar o reducir el valor, y de un umbral o *bias*. Además, en la salida de la neurona puede incluirse una función de activación, que aplica una transformación no lineal sobre el valor de salida. El modelo matemático general que



siguen las neuronas artificiales o perceptrones simples de una ANN se muestra en la Ecuación (1.13).

$$y_j = f \left( \sum_{i=1}^N \omega_{ij} x_i + b_j \right) \quad (1.13)$$

Donde  $y_j$  es el valor de salida de la neurona  $j$ ,  $x_i$  el el valor de las neuronas  $i$  conectada a la neurona  $j$ ,  $\omega_{ij}$  es el peso asociado a la conexión entre las neuronas  $i$  y  $j$ ,  $b_j$  es el *bias* de la neurona  $j$ , y  $f(\cdot)$  es la función de activación. Las funciones de activación más conocidas y utilizadas se resumen en la Tabla 1.2.

Nombre	Función $f$	Rango
Lineal	$f(x) = x$	$[-\infty, +\infty]$
Sigmoide o logística	$f(x) = \frac{1}{1 + e^{-x}}$	$[0, +\infty]$
Softplus	$f(x) = \log(1 + e^{-x})$	$[0, +\infty]$
Softmax	$f(z)_i = \frac{e^{-z_i}}{\sum_{k=1}^K e^{-z_k}} ; z \in \mathbb{R}^K ; i = 1, \dots, K$	$[0, 1]^K$
<i>Rectified Linear Unit</i> (ReLU)	$f(x) = \max(0, x)$	$[0, +\infty]$
Leaky ReLU	$f(x) = \begin{cases} x & \text{si } x \geq 0 \\ \alpha x & \text{si } x < 0 \end{cases}$	$[-\infty, +\infty]$

**Tabla 1.2: Principales funciones de activación de las ANN.**

Las neuronas de una ANN se organizan en capas o *layers*, de tal modo que el valor de las neuronas de una capa depende del valor de las neuronas de las capas anteriores. Las capas de las redes se diferencian según el tipo de transformación matemática que aplica sobre los datos de entrada. La arquitectura de una ANN se define en función del tipo de capas que se emplean y de su distribución en la red.

Actualmente, existen numerosos tipos de arquitecturas. Destacan:

- **Perceptrón multicapa:** o *DenseNet* (Figura 1.16a). Se trata de una red compuesta por capas de perceptrones simples. Las capas se clasifican en:
  - Capa de entrada o *input layer*: es la capa por la que entran los datos en la red.

- Capas ocultas o *hidden layers*: son las capas intermedias, en las que se producen la transformación de los datos en los perceptrones simples, según el modelo de la Ecuación (1.13). Cada capa de la red se encuentra conectada a su capa inmediatamente anterior y posterior, y sus neuronas pueden estar total o localmente conectada. La más común es la red totalmente conectada, o *fully-connected network*.
- Capa de salida o *output layer*: es la capa por la que se devuelve el resultado de la red.

Los perceptrones multicapa son redes generalmente empleadas para la resolución de problemas de clasificación.

- **Redes Neuronales Convolucionales o *Convolutional Neural Networks (CNN)***: las redes CNN (Figura 1.16b) se caracterizan por incluir capas u operaciones de convolución. La operación de convolución entre dos matrices se define matemáticamente como:

**Convolución 2D:**

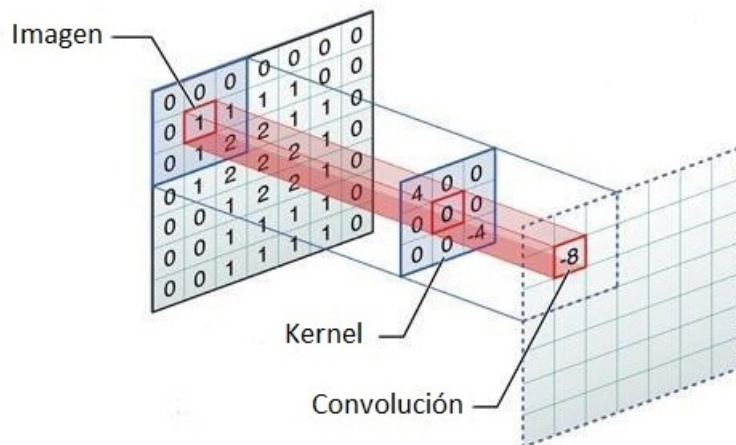
$$Y[i, j] = \sum_{k=-r}^r \sum_{k=-r}^r K[u, v]X[i - u, j - v] \quad (1.14)$$

**Convolución 3D:**

$$Y[i, j, k] = \sum_{k=-r}^r \sum_{k=-r}^r \sum_{k=-r}^r K[u, v, w]X[i - u, j - v, k - w] \quad (1.15)$$

$$\text{Notación: } Y = K \otimes X \quad (1.16)$$

Donde  $Y$  es el resultado de la convolución entre la imagen o matriz  $X$  y un filtro, kernel o matriz de convolución  $K$  de radio  $r$ . Un ejemplo de la operación de convolución sobre una imagen se muestra en la Figura 1.14.



**Figura 1.14: Operación de convolución de un píxel de una imagen 2D.** La operación de convolución se realiza mediante un kernel de dimensiones  $3 \times 3$  (radio  $r = 1$ ). Se observa que el resultado en la posición de un píxel equivale al producto entre los valores del kernel y los valores de la región  $3 \times 3$  en torno al píxel. Fuente: *Towards Data Science (I)*.

En el caso de las CNN, cada capa realiza una operación de convolución entre los datos de entrada contenidos en  $N$  matrices o imágenes (las cuales actúan como neuronas artificiales)

y un número determinado de filtros o kernels. Cada kernel convoluciona las  $N$  matrices de entrada, dando lugar a  $N$  matrices convolucionadas. Éstas posteriormente se suman, y la matriz resultante se suma con un factor *bias*. Finalmente se aplica una función de activación sobre el resultado, de forma análoga al perceptrón simple. Por cada kernel se obtiene una matriz o neurona de salida, de modo que el número de kernels determina el número de neuronas de salida de cada capa, también conocidas como canales. En este tipo de arquitectura, los pesos de la red equivalen a los valores de los kernels, de modo que el modelo general de una ANN planteado en (1.13) se puede particularizar en una red CNN por la expresión (1.17).

$$Y_j = f \left( b_j + \sum_i^N K_{ij} \otimes X_i \right) \quad (1.17)$$

Donde  $Y_j$  es la neurona o matriz de salida  $j$ ,  $X_i$  es cada una de las  $N$  neuronas o matrices de entrada,  $K_{ij}$  es el kernel o filtro asociado a la conexión entre la neurona de entrada  $i$  y la neurona de salida  $j$ ,  $b_j$  es el *bias* asociado a la neurona de salida  $j$  y  $f(\cdot)$  es la función de activación.

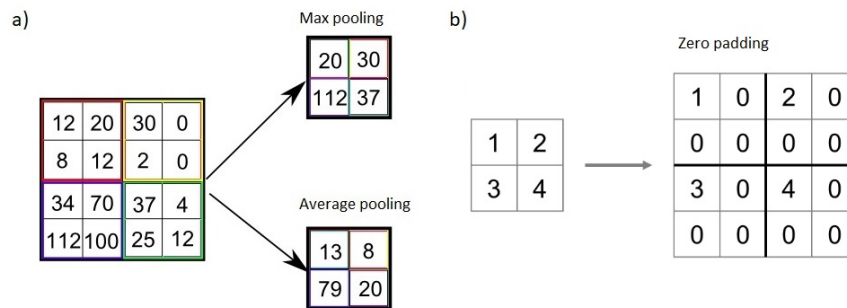
Las CNN se emplean habitualmente para la extracción de características de imágenes (2D o 3D), y se suelen concatenar con perceptrones multicapa *fully-connected* para resolver problemas de clasificación de imágenes.

- **Autoencoders convolucionales:** también conocidas como *UNets* (Figura 1.16c). Los *autoencoders* convolucionales, tal y como su nombre indica, son un tipo particular de las CNN que aprenden a codificar y decodificar los datos (autocodificación) tomando como entrada y salida de la red los mismos datos. La arquitectura de estas redes se caracteriza por combinar las capas convolucionales con capas de modificación de la dimensionalidad de las neuronas o matrices. Este tipo de redes se estratifican por niveles de tal forma que, en un mismo nivel, las dimensiones de las matrices en las capa convolucionales son iguales. Las redes *autoencoder* constan de dos partes o tramos:

- *Encoder:* en esta sección de la red se codifica la información desde el espacio inicial o nativo al espacio reducido o latente mediante la reducción de la dimensionalidad o *downsampling*. En cada nivel se realizan dos tipos de operaciones en el siguiente orden:
  1. Capas de convolución: procesan la información en cada nivel.
  2. Capa de *downsampling*: las dimensiones de las matrices se reducen mediante la selección de un valor por región, como el valor máximo (*Max pooling*) o el valor promedio (*Average pooling*), tal y como se muestra en la Figura 1.15a.
- *Decoder:* en esta sección de la red se decodifica la información desde el espacio latente al espacio inicial mediante el aumento de la dimensionalidad o *upsampling*. En cada nivel se realizan dos tipos de operaciones en el siguiente orden:
  1. Capa de *upsampling*: las dimensiones de las matrices aumentan mediante la generación de nuevos valores por diferentes técnicas, como el *Zero padding* (Figura 1.15b) o interpolación lineal o cúbica.
  2. Capas de convolución: procesan la información en cada nivel.

Además, para evitar la pérdida de información durante el proceso de codificación y decodificación, los *autoencoders* convolucionales pueden incluir conexiones directas entre los tramos *encoder* y *decoder*. Antes de iniciar las operaciones convolucionales sobre las matrices redimensionadas en cada nivel del *decoder*, las matrices convolucionadas del correspondiente nivel del tramo *encoder* se concatenan con el conjunto remuestreado tras el *upsampling*. De este modo, la información que se ha extraído en cada nivel durante la fase de codificación antes de la reducción de dimensiones se transfiere de forma íntegra y sin pérdidas al tramo de decodificación de la red. En las conexiones directas también pueden incluirse capas convolucionales adicionales.

Los *autoencoders* convolucionales se emplean en diversas aplicaciones, como la codificación y decodificación espacial de la información, la síntesis de imágenes, o la extracción de características o *feature extraction* empleando únicamente el tramo *encoder* de la red.



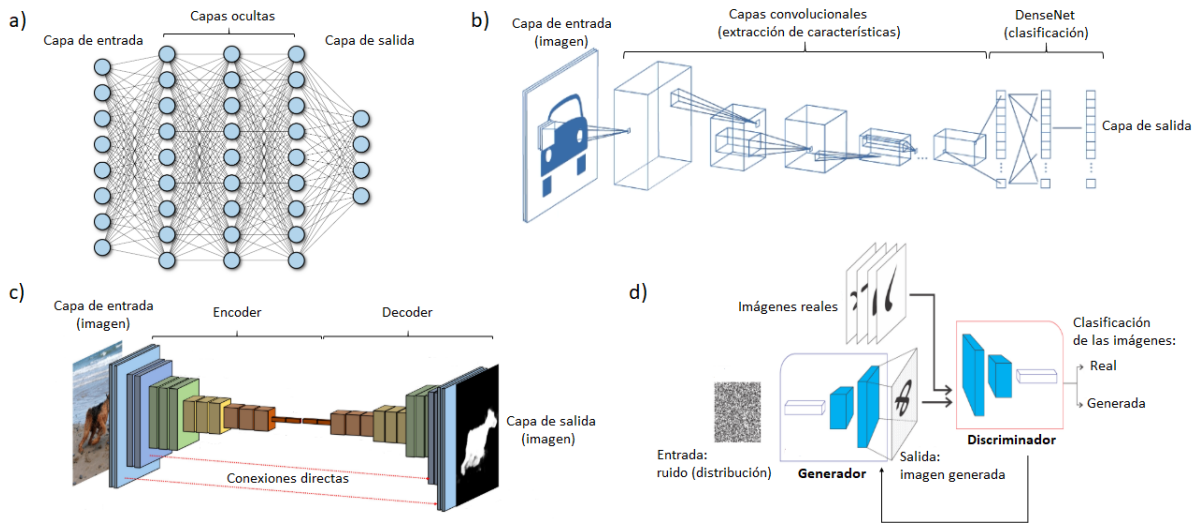
**Figura 1.15: Modificación de la dimensionalidad de las neuronas de una CNN.** a) *Downsampling* o reducción de la dimensionalidad mediante la selección del valor máximo (*Max pooling*) o promedio (*Average pooling*) de las regiones de la imagen. b) *Upsampling* o aumento de la dimensionalidad mediante la generación de nuevos elementos en la matriz de valor 0 (*Zero padding*). Fuentes: *Quora* y *Stack Exchange (1)*.

- **Redes Generativas Antagónicas o *Generative Adversarial Networks (GAN)*:** las redes GAN (Figura 1.16d) son un tipo de redes recientemente propuestas (Goodfellow y col. 2014) que se basan en la implementación de dos redes adversarias, el generador y el discriminador, las cuales compiten entre ellas para mejorar su tarea y superarse mutuamente.

La función de la red generadora es la de registrar y modificar la distribución de los datos dando lugar a muestras realistas, mientras que la red discriminadora trata de identificar si las muestras son originales o generadas por la primera red. El objetivo global de la red GAN es conseguir que el generador minimice el error entre las muestras generadas y originales, de forma que el discriminador no sea capaz de distinguirlos. Este método es conocido en teoría de juegos como minimax.

Las redes GAN han resultado ser de gran utilidad a la hora de resolver determinados problemas: generación de imágenes realistas a partir de ruido, mejora de la resolución de imágenes y vídeos, control de sistemas dinámicos no lineales...

A la hora de resolver un problema mediante la aplicación de las ANN, el primer paso consiste en la elección de una arquitectura adecuada, que dependerá fundamentalmente del tipo de



**Figura 1.16: Esquema de las principales arquitecturas de las ANN.** a) Perceptrón multicapa o *DenseNet* (*fully-connected network*). b) CNN + *DenseNet*. c) *Autoencoder* convolucional o *UNet*. d) GAN. Fuentes: Ramsundar y Zadeh 2018, *free Code Camp (I)*, *GitHub (I)* y *GitHub (II)*.

problema que se desee resolver. Una vez que se haya determinado la arquitectura, se procede al entrenamiento de la ANN.

### 1.4.3 Entrenamiento de las ANN

La fase de entrenamiento de una ANN es el proceso en el cual la red aprende a ejecutar una tarea y a perfeccionarla mediante la modificación de sus parámetros. Para poder llevar a cabo el entrenamiento, es necesario disponer de datos o ejemplos de entrada de la red, y de las soluciones correspondientes a estos datos (se trata de un aprendizaje supervisado). Además de los datos, debe definirse una función de coste o *loss function*, que se tratará de minimizar durante el entrenamiento de la red.

Previo al inicio del entrenamiento, los datos disponibles se dividen en tres grupos: datos de entrenamiento o *training*, datos de validación o *validation* y datos de evaluación o *test*. En la fase de entrenamiento, la red solo empleará los conjuntos de datos de entrenamiento y validación. Aunque no existen unas proporciones fijas, entre el 60 y 80% de los datos suelen emplearse para entrenamiento y validación, mientras que el resto se reserva para la fase de *test*. También es importante llevar a cabo un adecuado preprocesado de los datos para garantizar un buen entrenamiento de la red. En el caso particular de las imágenes biomédicas, algunas técnicas habitualmente aplicadas en el preprocesado son el filtrado, la corrección de inhomogeneidad, el registro, la normalización de la intensidad, etc.

Una vez se han preparado los datos para el entrenamiento y se ha diseñado la arquitectura, la red inicia el algoritmo de entrenamiento, que consta de las siguientes fases:

1. **Inicialización:** la red inicializa sus parámetros de entrenamiento: los pesos  $\omega_{ij}$  y los *bias*  $b_j$ , o, en el caso de las CNN, los valores de los kernels  $K_{ij}$ . La inicialización de los parámetros puede ser aleatoria siguiendo una distribución uniforme, normal, etc., o puede inicializarse tomando los valores de los pesos de una red preentrenada.

2. **Elección de la función de coste o *loss function*  $F$ :** se trata de una expresión matemática que calcula el error o *loss*  $F(y, \hat{y})$  entre la solución verdadera  $y$  y salida estimada por la red  $\hat{y}$  para unos datos de entrada. En función del error cometido por la red, el algoritmo calculará las modificaciones de los parámetros necesarias para minimizar el error en las sucesivas iteraciones (Figura 1.17a). Algunas de las funciones de coste más importantes se resumen en la Tabla 1.3. Su elección depende del tipo de datos que se desean comparar.

Nombre	Función $F$	Aplicación
<i>Mean Squared Error</i> (MSE) o norma L2	$F(y, \hat{y}) = \frac{1}{n} \sum_{i=1}^n (\hat{y}(i) - y(i))^2$	Problemas de regresión (diferencias de intensidad)
<i>Mean Absolute Error</i> (MAE) o norma L1	$F(y, \hat{y}) = \frac{1}{n} \sum_{i=1}^n  \hat{y}(i) - y(i) $	Problemas de regresión (diferencias de intensidad)
<i>Gradient Difference Loss</i> (GDL)	$F(y, \hat{y}) = \sum_i \left   \nabla_i y  -  \nabla_i \hat{y}  \right ^\alpha$ $i = \text{direcciones espaciales (x, y, z)}$	Problemas de regresión (diferencias de gradiente)
<i>Cross-entropy</i> o <i>log loss</i>	$F(y, \hat{y}) = - \sum_{i=1}^n y(i) \log(\hat{y}(i))$	Problemas de clasificación
Divergencia de Kullback-Leibler	$F(y, \hat{y}) = - \sum_{i=1}^n y(i) \log \left( \frac{y(i)}{\hat{y}(i)} \right)$	Problemas de comparación entre distribuciones de probabilidad

Tabla 1.3: Funciones de coste destacadas de las ANN.

3. **Introducción de los datos de entrada:** los datos se introducen en las primeras capas de la red. Generalmente, se introducen en pequeños conjuntos o *batches* de entrenamiento.
4. **Transformación de los datos:** la red realiza las operaciones con los datos de entrada, transformándolos en cada una de las capas en función de los valores de los parámetros (los pesos y *biases*). Las operaciones y, por lo tanto, la transformación de los datos, dependen de la arquitectura de la red seleccionada.
5. **Generación de los datos de salida y comparación:** para cada uno de los  $N$  datos de entrada del *batch*, la red devuelve unos datos de salida y calcula el error o *loss* existente entre éstos y sus correspondientes soluciones aplicando la función de coste  $F(y, \hat{y})$ . El error empleado para modificar la red será el error promedio de todos los elementos del *batch*.
6. **Retropropagación del error:** en función del error calculado, se modifican los valores de los parámetros de las capas de la red mediante algoritmos matemáticas de retropropagación del error o *backpropagation* basadas en la derivada o gradiente. La más conocida y extendida

es la técnica del Descenso de Gradiente o *Gradient Descent*, aunque también existen otras técnicas similares, como el Gradiente Conjugado, el método de Newton o el algoritmo Levenberg-Marquardt, entre otras. Los algoritmos basados en el Descenso de Gradiente se basan en el cálculo de la derivada del error total  $E_T$  cometido con respecto a cada peso  $\omega_i^t$  de la red en cada iteración  $t$ . Esta derivada, multiplicada por un factor  $\eta$ , es la modificación del peso  $\Delta\omega_i^t$  que se aplica. El factor  $\eta \in [0, 1]$ , denominado *learning rate* determina la “agresividad” con la que se modifican los parámetros de la red. A mayor valor de *learning rate*, la modificación de los parámetros será mayor.

$$\text{Error total cometido por la red: } E_T = F(y, \hat{y}) \quad (1.18)$$

$$\text{Modificación de los pesos de la red: } \omega_i^{t+1} = \omega_i^t + \Delta\omega_i^t \quad (1.19)$$

$$\Delta\omega_i^t = -\eta \frac{dE_T}{d\omega_i} \quad (1.20)$$

El algoritmo calcula en primer lugar las modificaciones de los pesos de las capas de salida, y, puesto que los pesos de una capa dependen de los pesos de la capa anterior, aplica la “regla de la cadena” de derivación para calcular las modificaciones de los pesos de las primeras capas. Es por ello que este tipo de algoritmos son conocidos como algoritmos *backpropagation*.

Generalmente, el entrenamiento se realiza en un número de iteraciones o *epochs*. En cada epoch, se van extrayendo y entrenando de forma iterativa (desde el Paso 3) pequeños *batches* de datos, hasta recorrer por completo todo el conjunto de datos de entrenamiento. Una vez ha finalizado la *epoch*, se introduce en la red el conjunto de datos de validación, los cuales no han sido entrenados por la red, y se predicen las soluciones de estos datos. Si el error o *loss* de validación cometido en una *epoch* es menor que en las anteriores, se almacenan los parámetros de la red al final de dicha *epoch*. De este modo, en cada *epoch*, quedan almacenados los parámetros que mejores soluciones generan ante datos que no han sido “vistos” por la red. Las *epochs* se suceden hasta que la *loss* calculada entre la salida de la red y las soluciones converge, es decir, se estabiliza (Figura 1.17b). Llegados a este punto, la fase de entrenamiento ha concluido.

Se dice que el entrenamiento de las ANN sigue un modelo de “caja negra”, ya que el proceso interno de la modificación de los parámetros no es un proceso determinista, sino que lo decide la red en cada entrenamiento. Durante el proceso de entrenamiento, pueden surgir diferentes problemas que impidan el correcto aprendizaje de la tarea:

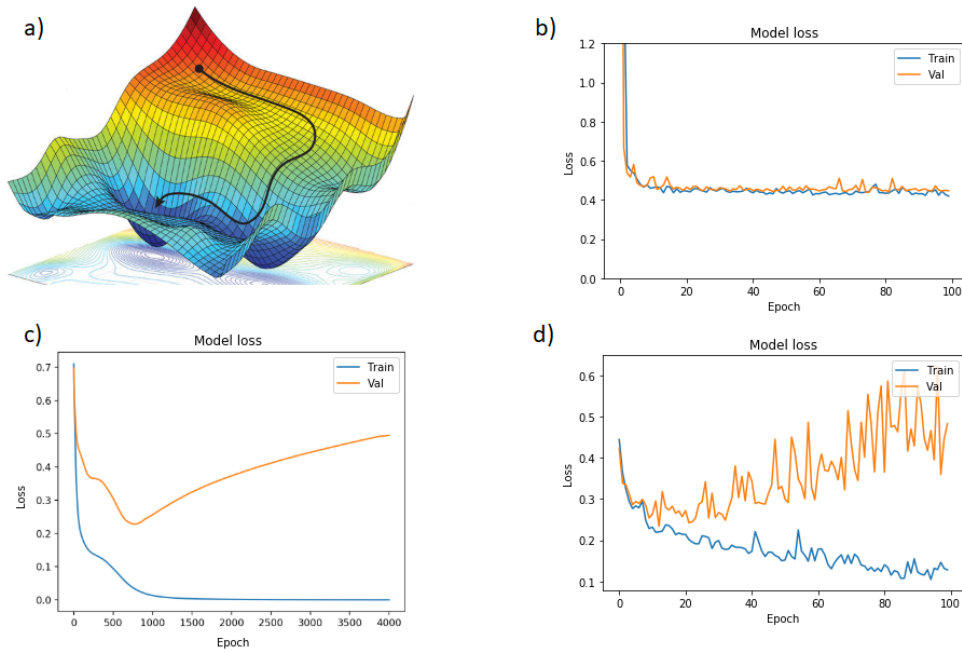
- **Falta de datos:** la falta de suficientes ejemplos dato-solución impide que la red pueda aprender adecuadamente la tarea, y que sea capaz de predecir la solución ante nuevos casos.
- **Sobreajuste u *overfitting*:** surge ante un entrenamiento demasiado largo, en el que la red se entrena demasiadas veces sobre el mismo conjunto de datos. El exceso de entrenamiento provoca que la red minimice el error de los datos de entrenamiento a costa de aumentarlo ante casos nuevos o no entrenados, como los datos de validación. La red aprende demasiado de sus datos y pierde capacidad de predecir adecuadamente los datos no entrenados. En un caso de *overfitting*, se observa que en las últimas etapas del entrenamiento la *loss* de entrenamiento disminuye, mientras que la de validación aumenta (Figura 1.17c).

- **Desvanecimiento del gradiente o *gradient vanishing*:** ocurre ante redes muy profundas, es decir, con muchas capas. La derivación del error total  $E_T$  con respecto a los pesos  $\omega_i^t$  de las primeras capas de la red implica la multiplicación de muchas derivadas sucesivas al aplicar la “regla de la cadena”. En consecuencia, las capas iniciales de la red percibirán una variación  $\Delta\omega_i^t$  muy atenuada con la que apenas modificarán sus pesos en cada iteración, y, por lo tanto, no aprenderán durante el entrenamiento.
- **Problemas de arquitectura:** la elección y/o el diseño de una mala arquitectura provoca una mala resolución de la tarea por parte de la red. No existe una única red para cada problema, aunque sí que existen ciertas arquitecturas que son mejores que otras para resolver determinadas tareas. Tampoco existe un número correcto de capas, sino que éste dependerá de la complejidad de la tarea. Una red con falta de capas no resolverá bien los problemas complejos, ya que no será capaz de extraer toda la información, mientras que una red con exceso de capas puede aumentar significativamente el tiempo de entrenamiento y provocar *overfitting*.
- **Problemas de evolución de la *loss*:** destacan dos problemas relacionados con la minimización de la *loss*:
  - Falta de estabilidad: se produce una modificación abrupta de los parámetros en cada iteración por varias razones: un mal preprocesado de los datos (por ejemplo, datos no normalizados), un *learning rate* demasiado elevado... Todo ello se traduce en un comportamiento errático de la *loss* durante el entrenamiento (Figura 1.17d).
  - Estancamiento en mínimos locales: la *loss* se estanca en un valor que no es el mínimo global de la función que mejor resuelve el problema, sino que se trata de un mínimo local. Generalmente, este problema se asocia a una mala elección del *learning rate*.

Ante los problemas que surgen durante el entrenamiento de las ANN, existen una serie de técnicas y operaciones de regularización que solucionan los problemas anteriormente enumerados, y que suelen dar mejores resultados en el entrenamiento de las ANN:

- ***Data augmentation*:** las técnicas de *data augmentation* (Figura 1.18a) permiten generar nuevos casos de entrenamiento mediante la aplicación de transformaciones geométricas sobre los casos existentes: rotación, traslación, reescalado, deformación, combinación lineal (*mixup*), etc. De esta forma, solucionan el problema de la falta de ejemplos en la fase de entrenamiento.
- ***Batch normalization*:** la capa de *batch normalization* normaliza los valores de las neuronas de entrada mediante el siguiente procedimiento.
  1. Se calcula la media  $\mu$  y la varianza  $\sigma^2$  del conjunto o *batch* de  $N$  datos  $x_i$  que entran en la capa. Media:  $\mu = \frac{1}{N} \sum_{i=1}^N x_i$ . Varianza:  $\sigma^2 = \frac{1}{N} \sum_{i=1}^N (x_i - \mu)^2$ .
  2. Se normalizan los datos restándoles la media y dividiendo por la desviación. A ésta se le suma un pequeño valor  $\epsilon$  para evitar dividir los datos entre 0, en caso de que la varianza fuese nula. Normalización:  $\hat{x}_i = \frac{x_i - \mu}{\sqrt{\sigma^2 + \epsilon}}$ .



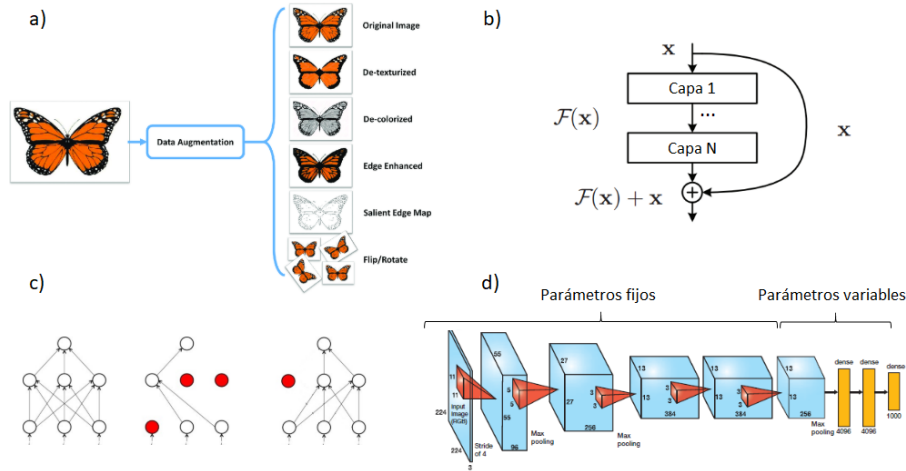


**Figura 1.17: Evolución del error o *loss* durante el entrenamiento de una ANN.** a) Minimización de la función de coste o *loss function* por descenso de gradiente. b) Evolución estable de las *losses* de entrenamiento y validación en las diferentes *epochs* del entrenamiento hasta alcanzar la convergencia. c) Efecto del sobreajuste u *overfitting* sobre las *losses* de entrenamiento y validación. d) Evolución inestable de las *losses* de entrenamiento y validación en las diferentes *epochs* del entrenamiento. Fuentes: *Science, Machine Learning Mastery (I), Machine Learning Mastery (II)* y *free Code Camp (II)*.

3. Se reescalan los datos multiplicando por un factor  $\gamma$ , y se les suma un *bias*  $\beta$ . Ambos factores se van optimizando durante el proceso de entrenamiento de la red. Reescalado y suma del umbral:  $y_i = \gamma \hat{x}_i + \beta$ .

- **Bloques residuales:** las redes formadas por bloques residuales, o *ResNets* se basan en la inclusión de conexiones directas entre diferentes puntos de la red, denominadas *shortcut connections*, *skipp-connections* o *residual connections* (Figura 1.18b). Un bloque residual consiste en un conjunto de capas cuya salida  $F(x)$  se suma con la entrada  $x$  del bloque. De este modo, las capas del bloque residual aprenden el residuo que hay que sumarle a los datos de entrada para transformarlos en la salida deseada. Los bloques residuales resuelven los problemas del exceso de capas y del *gradient vanishing*, ya que si las capas finales de la red son innecesarias, el residuo aprendido de estos bloques residuales será nulo ( $F(x) = 0$ ), y los datos transformados alcanzarán la salida de la red gracias a los *shortcuts*.
- **Dropout:** el *dropout* aplicado a una capa consiste en fijar o bloquear un porcentaje de los parámetros de las neuronas de dicha capa en cada iteración de forma aleatoria (Figura 1.18c). Gracias a esta técnica de regularización, el aprendizaje de los pesos de la red produce de forma homogénea, evitando así que determinadas capas aprendan y ponderen más que otras.
- **Fine-tuning:** la técnica del *fine-tuning* consiste en bloquear parte de los parámetros de una red preentrenada (normalmente las primeras capas), de forma que tan solo unas determinadas capas de la red puedan modificarse (Figura 1.18d). Esta técnica permite adaptar

una red preentrenada a una tarea particular, sin necesidad de entrenar una nueva red desde el principio.



**Figura 1.18: Técnicas de regularización en el entrenamiento de una ANN.** a) *Data augmentation*. b) Bloque residual. c) *Dropout*. d) *Fine-tuning*. Fuentes: *Research Gate (I)*, *Towards Data Science (II)*, *Stack Exchange (II)* y *Data Wow*.

#### 1.4.4 Evaluación de las ANN

Una vez ha finalizado el entrenamiento de la ANN, se inicia la fase de evaluación o *test* de la red, con el objetivo de comprobar si la red es capaz de resolver correctamente la tarea, o en otras palabras, si es capaz de predecir soluciones correctas para nuevos datos.

En la fase de *test* se emplea un conjunto de pares dato-solución de *test* que no han sido entrenados o “vistos” por la red. Para cuantificar la exactitud de la red a la hora de predecir, se emplean una serie de métricas o funciones matemáticas que analizan el error o acierto entre las soluciones  $y$  verdaderas y las soluciones  $\hat{y}$  estimadas por la red. La elección de las métricas es fundamental a la hora de evaluar una red, ya que no existe una métrica única para cada problema. Existen numerosas métricas en función del problema que la ANN resuelve. Algunas de las métricas más extendidas son:

- **Precisión o *accuracy*:** es una métrica de uso muy extendido en problemas de clasificación. Calcula el porcentaje de aciertos entre dos elementos con respecto al total de casos. Su rango de valores es  $[0, 100]$ . Matemáticamente:

$$Accuracy(y, \hat{y}) = \frac{n_c}{N} \times 100 (\%) \quad (1.21)$$

Donde  $n_c$  representa el número de coincidencias ( $y(i) = \hat{y}(i)$ ), y  $N$  representa el número total de elementos o comparaciones realizadas

- **Sensibilidad y especificidad:** son dos métricas de uso muy extendido en problemas de clasificación. Calculan el porcentaje de clasificaciones verdaderamente positivas y negativas que ofrece una prueba diagnóstica, respectivamente. Matemáticamente:

$$S = \frac{VP}{VP + FN} \times 100 (\%) \quad (1.22)$$

$$E = \frac{VN}{VN + FP} \times 100 (\%) \quad (1.23)$$

Donde  $S$  y  $E$  son la sensibilidad y especificidad,  $VP$  y  $VN$  son los verdaderos positivos y verdaderos negativos, y  $FP$  y  $FN$  son los falsos positivos y falsos negativos, respectivamente.

- **Análisis Receiver Operating Characteristic (ROC):** se trata de un análisis que evalúa la sensibilidad frente a la especificidad de una clasificación variando el umbral de discriminación. Se representa generalmente con la curva ROC. Un indicador asociado de este análisis es el *Receiver Operating Characteristic - Area Under Curve* (ROC AUC), el cual mide el área bajo la curva ROC. Este indicador se puede interpretar como la probabilidad de que la clasificación del método de un elemento positivo sea más correcta que la de un elemento negativo.
- **Root Mean Squared Error (RMSE):** evalúa la distancia euclídea entre dos elementos. Su rango de valores es  $[0, +\infty]$ .

$$RMSE(y, \hat{y}) = \sqrt{MSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (\hat{y}(i) - y(i))^2} \quad (1.24)$$

- **Proporción Máxima de Señal a Ruido o Peak Signal-to-Noise Ratio (PSNR):** evalúa la calidad de la reconstrucción de una imagen estimando la relación señal-ruido. Cuanta mayor relación, mayor calidad presenta la reconstrucción. La ganancia se expresa en decibelios (dB), ya que el rango dinámico de valores  $y$ , por lo tanto, la ganancia que puede tomar la imagen es muy amplio. Su rango de valores es  $[0, +\infty]$ .

$$PSNR(y, \hat{y}) = 20 \log_{10} \left( \frac{M_I}{RMSE(y, \hat{y})} \right) \text{ (dB)} \quad (1.25)$$

Donde  $M_I$  es el máximo valor de intensidad de la imagen.

- **Índice de similitud estructural o Structural Similarity (SSIM):** mide la similitud estructural entre dos imágenes  $y$  e  $\hat{y}$ . En esta métrica se comparan tres aspectos de las imágenes: la estructura, luminancia y el contraste. Es una métrica adimensional cuyo rango de valores es  $[-1, 1]$ , siendo los extremos (-1 y 1) los valores de mínima y máxima similitud estructural, respectivamente. Matemáticamente:

$$SSIM(y, \hat{y}) = \frac{(2\mu_{\hat{y}}\mu_y + c_1)(2\sigma_{\hat{y}y} + c_2)}{(\mu_{\hat{y}}^2 + \mu_y^2 + c_1)(\sigma_{\hat{y}}^2 + \sigma_y^2 + c_2)} \quad (1.26)$$

Donde  $\mu_{\hat{y}}$  y  $\mu_y$  son las medias de las imágenes,  $\sigma_{\hat{y}}^2$  y  $\sigma_y^2$  son las varianzas de las imágenes, y  $\sigma_{\hat{y}y}$  es la covarianza entre ambas imágenes,  $c_1$  y  $c_2$  son dos constantes que estabilizan la métrica.

- **Coefficiente de correlación de Pearson:** mide el grado de correlación lineal entre dos señales o imágenes  $y$  e  $\hat{y}$ . Es una métrica adimensional cuyo rango de valores es  $[-1, 1]$ , siendo los extremos  $(-1$  y  $1)$  los valores de máxima correlación lineal.

$$\rho(y, \hat{y}) = \frac{\frac{1}{N} \sum_{i=1}^N (y(i) - \mu_y)(\hat{y}(i) - \mu_{\hat{y}})}{\sqrt{\frac{1}{N} \sum_{i=1}^N (y(i) - \mu_y)^2 \frac{1}{N} \sum_{i=1}^N (\hat{y}(i) - \mu_{\hat{y}})^2}} = \frac{\sigma_{\hat{y}y}}{\sigma_{\hat{y}}\sigma_y} \quad (1.27)$$

Donde  $\sigma_{\hat{y}}$  y  $\sigma_y$  son las desviaciones típicas, y  $\sigma_{\hat{y}y}$  es la covarianza de las imágenes.

## 1.5 Estado del arte

El creciente uso de la imagen RM aplicada al diagnóstico debido a sus múltiples ventajas (imagen de gran calidad, inócua, multimodal...) ha abierto diferentes líneas de investigación. De entre todas ellas, la aplicación de las nuevas técnicas de SR en imágenes IRM en todo tipo de modalidades ha ido en aumento en los últimos años. Destacan en este campo la técnica de reconstrucción adaptativa no local basada en *patches* de Manjón y col. 2010b, o la técnica de reconstrucción de una imagen SR a partir de información de alta resolución HR coplanar de un mismo sujeto (o *self-similarity*) propuesta por Manjón y col. 2010a. Ambos son métodos no locales que obtienen la información HR de la imagen mediante un algoritmo iterativo que sigue un modelo matemático (véase Ecuación 1.10) con una serie de restricciones, y minimiza una función de coste basada en la distancia o diferencia entre imágenes. En ambos casos, los resultados obtenidos superan ampliamente a las técnicas de interpolación clásicas. En otros estudios, como en Wang y col. 2014 o Zhang y col. 2019, se proponen técnicas basadas en la reconstrucción de imágenes IRM por *sparsity*, es decir, por combinación lineal de elementos de un diccionario. En cuanto a la resolución del problema de síntesis de imágenes entre modalidades, también existen ciertos estudios, aunque en menor proporción con respecto a la SR. El método simultáneo de SR y síntesis WAG, propuesto por Huang, Shao y Frangi 2017, se basa en el aprendizaje a partir de un diccionario siguiendo un criterio de emparejamiento o *matching* intermodal. En esta misma publicación, el método se compara con otros métodos menos potentes, pero también pertenecientes al estado del arte en síntesis intermodal: Yang y col. 2010, basado en representación codificada por *sparsity* (método SC), Roy, Carass y Prince 2013, basado en ejemplo de contraste (método MIMECS), y Vemulapalli, Van Nguyen y Kevin Zhou 2015, basado en aprendizaje no supervisado.

Por otro lado, el auge del aprendizaje automático basado en DL y de las ANN a causa de los últimos avances tecnológicos en la computación ha permitido resolver problemas muy variados (LeCun, Bengio e Hinton 2015). De entre los avances de mayor impacto en el campo del *machine learning*, destacan la red CNN ImageNet de clasificación de imágenes propuesta en Krizhevsky, Sutskever e Hinton 2012, la aplicación de *autoencoders* como filtros de ruido en Vincent y col. 2010, las redes GAN iniciadas por Goodfellow y col. 2014 y su aplicación en Ledig y col. 2017 para la obtención de imágenes realistas, o las redes CNN de SR de imagen en Dong y col. 2015. Su aplicación en la imagen IRM como método de SR, también ha sido notable y ha arrojado muy buenos resultados. Como ejemplos, cabe resaltar la red 3D SRCNN de Pham y col. 2017, la red 3D CNN entrenada para varios factores de escalado de Pham, Fablet y Rousseau 2017 o la red adversaria 3D SR GAN propuesta por Chen y col. 2018.

Finalmente, en referencia a la aplicación de la imagen IRM como herramienta de diagnóstico de la EA, la investigación ha ido dirigida al descubrimiento de nuevos biomarcadores y al diseño de métodos de clasificación automáticos. En Eskildsen y col. 2017 se desarrolla un estudio estadístico en la modalidad de imágenes de perfusión *Dinamic Suceptibility Contrast-MRI* (DSC-MRI) en el que se propone un modelo paramétrico de detección de alteraciones en el flujo vascular características de la EA, y se plantea como futuro biomarcador. En otros estudios, como Ben-Ahmed y col. 2017 o Daza y Rueda 2016, se propone la “relevancia visual” o *visual saliency* como criterio para los modelos computacionales de clasificación de pacientes. En Eskildsen y col. 2013 se emplea el grosor cortical de la medido en imágenes T1 como indicador del grado de atrofia, mientras que en Coupé y col. 2012a y Coupé y col. 2012b se desarrolla el método SNIPE, un sistema de segmentación y clasificación basado en *patches*, y se evalúa su eficiencia mediante biomarcadores cerebrales volumétricos. En relación a sistemas de ayuda al diagnóstico y detección de la EA basados en imagen médica, algunos estudios como Zhang y col. 2015 o Suk y Shen 2013 proponen la combinación de algoritmos de aprendizaje supervisado *Support Vectors Machines* (SVM) y DL en imágenes IRM e IRM y PET, respectivamente, mientras que otros se basan en el entrenamiento de ANN. En esta línea, Li y col. 2014 plantea el entrenamiento de una red CNN aplicada a la extracción de características y la generación de imágenes de nuevas modalidades (como PET) para completar las bases de datos y mejorar el problema de clasificación; Liu y col. 2014 propone una red de clasificación basada en *autoencoders* para la detección de la EA en la fase prodrómica en imágenes IRM; y Ortiz y col. 2016 plantea el entrenamiento de varias redes profundas a partir de *patches* IRM y PET e imágenes IRM y PET unido a un método de clasificación basado en votación.

## 1.6 Motivación y justificación

Actualmente, la EA es una de las enfermedades neurodegenerativas que mayor prevalencia presentan en la población de edad avanzada, y que mayor impacto económico y social provoca. Al no presentar una etiología conocida, resulta difícil asegurar su diagnóstico y, en consecuencia, se retrasa el inicio de su tratamiento. Sin embargo, diferentes estudios han demostrado que existe una fase preclínica previa a la aparición de los síntomas que se caracteriza por la presencia de determinadas alteraciones neuropatológicas. Por ello, es evidente que la investigación en nuevos biomarcadores y pruebas diagnósticas resulta fundamental a la hora de realizar un diagnóstico precoz de la EA.

Una de las vías de investigación diagnóstica más prometedoras en este aspecto ha resultado ser la imagen médica. De entre las numerosas técnicas que existen, destaca por sus múltiples ventajas la técnica de imagen IRM. Su inocuidad para el paciente, junto con las múltiples variantes que ofrece, hacen de la IRM una técnica con un gran potencial en el ámbito del diagnóstico. Sin embargo, la imagen médica presenta limitaciones de resolución, tiempo de adquisición y otros factores como ruido y artefactos, que empeoran la imagen obtenida. A estas limitaciones se suman la restricción de tiempo que sufren los profesionales médicos a la hora de la adquirir la imagen y diagnosticar al paciente y la falta de automatización de determinados protocolos. Ante esta realidad, queda patente la necesidad de nuevos métodos de diagnóstico basados en el análisis de la imagen médica automáticos, precisos y robustos.

Frente este reto, los grandes avances tecnológicos en el campo de la computación han permitido el desarrollo de potentes algoritmos de aprendizaje automático o *machine learning*. Concretamente, algoritmos propios del campo del DL, como las redes neuronales artificiales o ANN, han demostrado su gran versatilidad a la hora de adaptarse a una gran variedad de problemas, entre los que destaca la extracción de características y clasificación de imágenes. Por otro lado, la investigación en nuevas técnicas de mejora de la calidad de la imagen, como la SR, han presentado resultados realmente destacables.

Este proyecto, en definitiva, pretende fusionar ambos aspectos tecnológicos - superresolución y *Deep Learning* - y mostrar su aplicación en el campo del diagnóstico, en este caso de la EA. Con ello, pretendemos evidenciar el gran potencial de una tecnología emergente como lo son las redes neuronales artificiales aplicadas al ámbito biomédico como sistema de ayuda al diagnóstico.

## Capítulo 2

# Objetivos

*En el siguiente capítulo se definirán el objetivo general del proyecto y los objetivos específicos que se han pretendido alcanzar durante su desarrollo.*

### 2.1 Objetivo general

El objetivo general y propósito final del proyecto es diseñar y desarrollar un método de super-resolución (SR) en imágenes IRM y de detección automática de la enfermedad de Alzheimer (EA) basado en algoritmos de redes neuronales artificiales (ANN), pertenecientes al campo del aprendizaje automático profundo o *Deep Learning* (DL).

### 2.2 Objetivos específicos

Durante el desarrollo del proyecto, se pretenden alcanzar los siguientes objetivos específicos:

- Adquirir nuevos conocimientos sobre las nuevas técnicas de DL, concretamente sobre los algoritmos ANN (diseño de la arquitectura, entrenamiento, validación...)
- Aprender un nuevo lenguaje de programación, Python, y el manejo de librerías orientadas a DL.
- Adquirir la capacidad de implementar ANN en función del problema a resolver.
- Familiarizarse con las imágenes biomédicas de modalidad IRM como prueba diagnóstica de la EA.
- Desarrollar un adecuado preprocesado y estructuración de los datos IRM como fase previa a la ejecución de las ANN.
- Demostrar que una mejora en la resolución de las imágenes aplicando técnicas de SR mejora el diagnóstico automático de la EA llevado a cabo por las ANN.
- Estudiar el efecto de la inclusión de varias modalidades IRM en el aprendizaje de las ANN.





# Materiales y métodos

*En este capítulo se expondrán los materiales y métodos llevados a cabo para la resolución del problema planteado. En primer lugar, se enumerarán y describirán los materiales empleados. A continuación, se desarrollarán el esquema metodológico general que se ha seguido, las arquitecturas de las ANN propuestas, las técnicas de preprocesado que se han aplicado sobre los datos y su estructuración, y el entrenamiento y validación de las redes. Finalmente, se comentarán los experimentos diseñados y ejecutados, de los cuales se extraerán los resultados.*

### 3.1 Materiales

Los materiales empleados para el desarrollo del proyecto se agrupan en tres clases: datos, *hardware* y *software*.

#### 3.1.1 Datos

Los datos de los que se parte en el proyecto son imágenes volumétricas de modalidad IRM potenciadas en T1 y T2 del cerebro. Las imágenes proceden de dos conjuntos diferentes.

Las imágenes del primer conjunto se han obtenido del estudio publicado por Kulaga-Yoskovitz y col. 2015. En el estudio se dispone de una muestra de 25 sujetos sanos (12 hombres y 13 mujeres, edad: 21-53, media  $\pm$  desviación:  $31 \pm 7$  años), de los cuales se han adquirido dos imágenes IRM potenciadas en T1 (de resolución  $0,6 \text{ mm}^3$ ) y T2 (de resolución  $0,4 \times 0,4 \times 2 \text{ mm}^3$ ). La adquisición de los cortes se ha realizado con un equipo de 3 Tesla Siemens TimTrio. Tras la adquisición de las imágenes, se ha aplicado una serie de técnicas de preprocesado para mejorar la calidad de la imagen: corrección de inhomogeneidad, normalización de la intensidad, registro lineal al espacio estándar del *Montreal Neurological Institute* (MNI): el espacio MNI152, remuestreo mediante interpolación a una resolución de  $0,4 \text{ mm}^3$ , y un método de superresolución NLM (Manjón y col. 2010b) para reducir los artefactos de interpolación.

El segundo conjunto de imágenes se ha extraído de la base de datos *Alzheimer's Disease Neuroimaging Initiative* (ADNI) <sup>1</sup>. Para este proyecto, se ha dispuesto una muestra de 226 sujetos

---

<sup>1</sup>Los datos utilizados en la preparación de este artículo se obtuvieron a partir de *Alzheimer's Disease Neuroimaging Initiative* (ADNI) ([adni.loni.usc.edu](http://adni.loni.usc.edu)). ADNI fue lanzada en 2003 como una asociación público-privada, liderada por el

sanos pertenecientes al grupo Control Normal (CN) y 186 sujetos que presentan EA y que pertenecen al grupo AD. Cada sujeto tiene asociada una imagen IRM potenciada en T1 de procedencia variable, por lo que factores como el preprocesado, la resolución, etc. previas a la incorporación a la base de datos son diferentes entre sujetos. Asimismo, se dispone de un archivo CSV en el que se han registrado los identificadores de los sujetos, su edad y la etiqueta del grupo al que pertenecen (AD o CN).

### 3.1.2 Hardware

A nivel *hardware*, se ha empleado un equipo PC de sobremesa para realizar los experimentos. Las características del equipo se resumen en la Tabla 3.1.

Especificaciones	Características
Procesador	AMD FX(tm)-8150 Eight-Core, 3600 Mhz
Memoria RAM	16 GB
Sistema operativo	Microsoft Windows 10 Enterprise
Almacenamiento en disco mecánico	1 TB
Almacenamiento en disco sólido	256 GB

Tabla 3.1: Especificaciones del equipo PC de experimentación.

Los entrenamientos y validaciones de las ANN se han ejecutado en una tarjeta gráfica o *Graphics Processing Unit* (GPU) MSI GeForce GTX 1080Ti AERO, con un espacio de memoria de 11 GB.

### 3.1.3 Software

El desarrollo del código a lo largo del proyecto se ha realizado en lenguaje de programación Python. Python es un lenguaje interpretado y dinámico, desarrollado a finales de los años 80 por Guido Van Rossum. Actualmente se encuentra muy extendido debido a las grandes ventajas que presenta: código abierto, sintaxis sencilla, multiplataforma, multiparadigma (soporta la programación orientada a objetos, la programación funcional, etc.)... Al tratarse de un lenguaje de código abierto, Python ha permitido el desarrollo de entornos específicos con numerosas funcionalidades, denominados bibliotecas o librerías. Para este proyecto, se han empleado una serie de librerías con unas funciones concretas:

- **Tensorflow** y **Keras**: desarrollo de las ANN (arquitectura, entrenamiento, validación, etc.).
- **Numpy**, **Math**, **Matplotlib**: operaciones matemáticas, operaciones con vectores o *arrays*, gráficas...
- **Nibabel**: tratamiento de imágenes en formato NIfTI.
- **Os**: funciones propias del sistema operativo, concretamente manipulación de rutas y directorios.

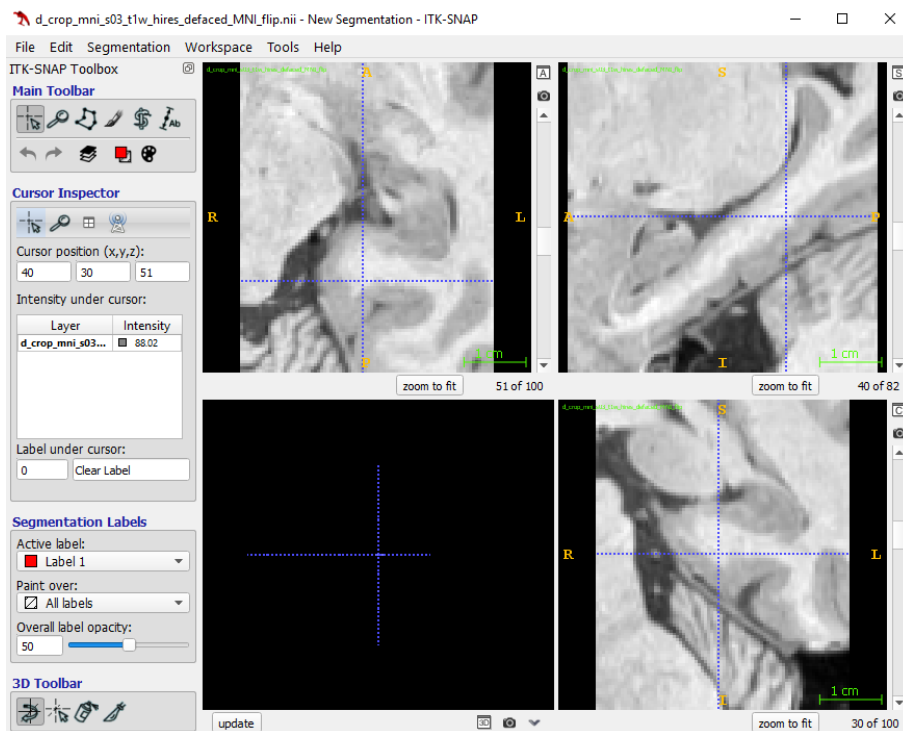
---

Investigador Principal Michael W. Weiner, MD. El objetivo principal de ADNI ha sido probar si la resonancia magnética en serie (RM), la tomografía por emisión de positrones (PET), otros marcadores biológicos y la evaluación clínica y neuropsicológica pueden combinarse para medir la progresión del deterioro cognitivo leve (DCL) y la enfermedad de Alzheimer temprana (EA).

- **Glob:** generación y tratamiento de listas.
- **Skimage** y **Scipy:** procesamiento de imágenes y cálculo de métricas.

Los *scripts* y las funciones del código se han desarrollado con el editor de código Sublime Text 3.0, y se han ejecutado en la plataforma Anaconda (versión 3.7). Esta plataforma permite, entre otras aplicaciones, la creación de entornos virtuales específicos con las librerías de Python necesarias para ejecutar los programas.

En lo referente a la visualización de las imágenes volumétricas IRM en formato NIFTI, ésta se ha llevado a cabo con la aplicación ITK-SNAP (versión 3.6). ITK-SNAP es una aplicación gratuita, de libre descarga y código abierto, multiplataforma e interactiva, que ha sido desarrollada en colaboración por investigadores de las universidades de Pennsylvania y Utah (EE.UU). Sus principales aplicaciones son la visualización y la segmentación semiautomática de imágenes 3D de modalidades IRM y CT. Para la visualización de imágenes 3D, el programa realiza una reconstrucción multiplanar o *Multiplanar Reconstruction* (MPR) y muestra los principales cortes ortogonales de la imagen - coronal, sagital y axial - en el visor, tal y como se muestra en la Figura 3.1. Además, incluye múltiples funcionalidades, como el desplazamiento en los diferentes planos con el cursor, el ajuste manual o automático del contraste y el brillo, la medición de distancias y áreas de las imágenes, etc.



**Figura 3.1:** Visor de la aplicación ITK-SNAP de una imagen volumétrica IRM de la región del HC del cerebro. La aplicación muestra los planos axial (superior izquierda), coronal (superior derecha) y sagital (inferior derecha) de la imagen, así como una representación tridimensional de los tres ejes (inferior izquierda).

Finalmente, parte del preprocesado de las imágenes IRM se ha llevado a cabo en lenguaje M y con el entorno de programación MATLAB R2018a de MathWorks. Para el registro de las imágenes, se ha empleado la herramienta de registro *Advanced Normalization Tools* (ANTs).

## 3.2 Esquema metodológico general

El esquema metodológico general del proyecto se ha desarrollado en base al objetivo final que se pretende alcanzar. En consecuencia, se han definido dos fases, en las que se plantean y resuelven los dos problemas fundamentales del proyecto por el siguiente orden:

1. **Fase de superresolución (SR):** resuelve el problema de la mejora de la calidad de la imagen mediante el diseño, entrenamiento y validación de dos ANN que transformen las imágenes IRM potenciadas en T1 de baja resolución o LR en imágenes IRM potenciadas en T1 y T2 SR. En esta fase se ha empleado el conjunto de datos propuesto en Kulaga-Yoskovitz y col. 2015.
2. **Fase de clasificación:** resuelve el problema de la detección de imágenes IRM de sujetos sanos y con EA a través del diseño, entrenamiento y validación de una ANN que extraiga las características de las imágenes IRM de entrada y las clasifique en dos grupos: sujetos sanos y con EA. En esta fase se ha empleado el conjunto de datos de ADNI.

En cada uno de los experimentos desarrollados en ambas fases, se ha seguido un procedimiento general común y ordenado que se compone de los siguientes pasos:

1. **Investigación:** se investiga en primer lugar acerca del problema a resolver y de las soluciones propuestas hasta el momento en el área de las ANN. Un mismo problema puede presentar diferentes soluciones válidas.
2. **Diseño y/o modificación de la arquitectura de la ANN:** a continuación se eligen las arquitecturas de las ANN, en función de los datos disponibles y de la tarea que la red debe aprender. Pueden existir diferentes arquitecturas para un mismo problema.
3. **Estructuración y preprocesado de los datos:** en este paso se escogen los datos, se estructuran y preprocesan, y se organizan los diferentes conjuntos de entrenamiento y validación.
4. **Diseño y ejecución del entrenamiento y validación de la ANN:** en este último paso se establecen los parámetros de entrenamiento, las funciones de coste, las métricas, etc. A continuación se entrena y evalúa la red.
5. **Evaluación de los resultados y correcciones:** los resultados obtenidos se comparan con las referencias bibliográficas y con otros experimentos realizados, y se proponen y aplican las modificaciones necesarias en los sucesivos experimentos.

Es importante destacar que la experimentación en el área de las ANN no es determinista, por lo que un mismo problema puede resolverse mediante diferentes técnicas. Por ello, la metodología propuesta se ha llevado a cabo de forma iterativa, con la finalidad de analizar las diferentes soluciones al mismo problema y escoger la mejor. Además, en cada una de las iteraciones, tan solo se ha realizado una única modificación con respecto a la anterior para comprobar su efecto en la resolución del problema.

### 3.3 Arquitecturas propuestas de las redes neuronales artificiales

El diseño de la arquitectura de las ANN es un aspecto fundamental en la resolución los dos problemas que se pretenden resolver en este proyecto: SR y clasificación. Al tratarse de problemas distintos, las arquitecturas propuestas en cada fase también lo serán. A continuación se presentan los esquemas de las arquitecturas de red que mejores resultados han ofrecido, aunque es importante mencionar que algunas de ellas se han visto sujetas a diversas modificaciones durante el proceso de experimentación. La implementación de la arquitectura se ha realizado con el lenguaje de programación Python, y con las librerías **tensorflow** y **keras** de DL, las cuales incorporan todas las capas empleadas en el diseño.

#### 3.3.1 Fase de SR

El objetivo en la fase de SR es generar nuevas imágenes volumétricas IRM potenciadas en T1 y T2 SR a partir de imágenes volumétricas IRM potenciadas en T1 LR. Para resolver el problema, se proponen dos ANN que resuelven dos problemas distintos:

1. **Red T1 SR:** la red estima los residuos  $R_{T1}$  que deben sumarse a las imágenes IRM potenciadas en T1 LR ( $I_{T1}^{LR}$ ) para transformarlas en imágenes IRM potenciadas en T1 SR ( $I_{T1}^{SR}$ ). La transformación se expresa matemáticamente como:

$$R_{T1} = \text{SRT1Net}(I_{T1}^{LR}) \quad (3.1)$$

$$I_{T1}^{SR} = I_{T1}^{LR} + R_{T1} \quad (3.2)$$

La red escogida es una red *Deep Recursive Residual Network* (DRRN) modificada (un tipo de red CNN) que recibe como entrada un tensor de 5 dimensiones con las imágenes  $I_{T1}^{LR}$ , y devuelve como salida otro tensor de 5 dimensiones que contiene los residuos  $R_{T1}$ . Las dimensiones de los tensores son  $N \times x \times y \times z \times c$ , donde  $N$  es el número de elementos o *batch size* del vector,  $x \times y \times z$  son las dimensiones de los volúmenes, y  $c$  es el número de canales (las imágenes de entrada son monocanal, por lo que  $c = 1$ ). La arquitectura de la red (Figura 3.2) consta de 1 113 921 parámetros - de los cuales 1 112 001 son entrenables - y se compone de:

- I Una capa convolucional 3D o *Conv3D* (función de activación lineal, 64 filtros) seguida de una capa *batch normalization*.
- II Un bloque residual constituido por dos conjuntos *Conv3D* (función de activación ReLU, 64 filtros) + *batch normalization*. A la salida de un bloque residual, el tensor resultante de la convolución se suma con la entrada.
- III Tres bloques sucesivos formados por un bloque *batch normalization* + *Conv3D* (función de activación ReLU, 64 filtros) + *batch normalization* seguido de un bloque residual *Conv3D* (función de activación ReLU, 64 filtros) + *batch normalization*.
- IV Dos bloques convolucionales sucesivos *batch normalization* + *Conv3D* (función de activación ReLU, 64 filtros).

V Una salida *batch normalization* + *Conv3D* (función de activación lineal, 1 filtro).

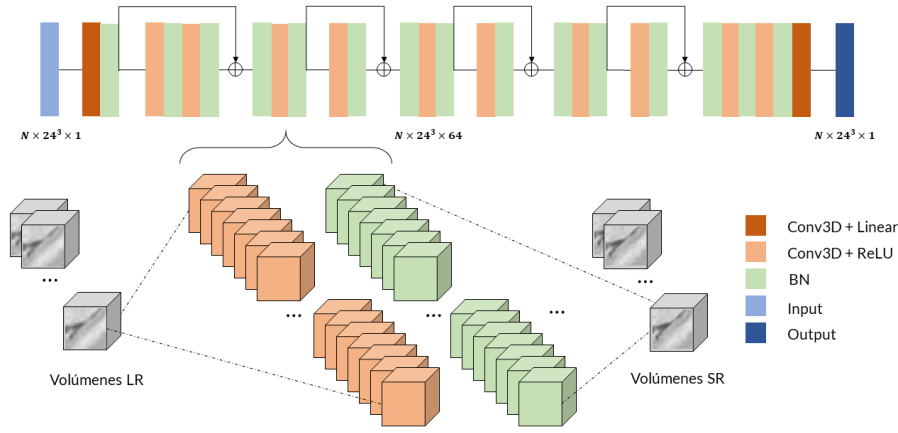


Figura 3.2: Esquema de la arquitectura de la red T1 SR.

2. **Red de síntesis T2 SR:** calcula los residuos  $R_{T2}$  que hay que sumarle a las imágenes IRM potenciadas en T1 SR obtenidas de la primera red ( $I_{T1}^{SR}$ ) para transformarlas en imágenes IRM potenciadas en T2 SR ( $I_{T2}^{SR}$ ). La transformación se expresa matemáticamente como:

$$R_{T2} = \text{SRT2Net}(I_{T1}^{SR}, I_{T1}^{LR}) \quad (3.3)$$

$$I_{T2}^{SR} = I_{T1}^{SR} + R_{T2} \quad (3.4)$$

La red escogida es una red *UNet* modificada que, al igual que en la red de síntesis SR T1, recibe como entrada un tensor de 5 dimensiones ( $N \times x \times y \times z \times c$ ), y devuelve como salida otro tensor de idénticas dimensiones. En este caso, al existir dos volúmenes IRM de entrada (LR y SR, el número de canales es de  $c = 2$ ). Sin embargo, la arquitectura de la red (Figura 3.3) es completamente diferente. Consta de un total de 7 933 953 parámetros - de los cuales 7 929 345 son entrenables - y su estructura se compone de:

- I Una capa convolucional de entrada *Conv3D* (función de activación ReLU, 32 filtros).
- II Tramo descendente *encoder* con cuatro niveles: la salida de un nivel es la entrada del siguiente. Cada nivel se compone de dos bloques sucesivos *batch normalization* + *Conv3D* (función de activación ReLU,  $n$  filtros), a excepción del primer nivel, que tan solo presenta un bloque. El número de filtros  $n$  de convolución en cada nivel se duplica con respecto al anterior: 32, 64, 128 y 256 (todos los valores son potencias de 2). Al final de cada nivel, una capa de *MaxPooling* reduce las dimensiones a la mitad (factor de *downsampling*  $f = 2$ ).
- III Tres conexiones o *shortcuts* convolucionales: la salida de cada nivel en el tramo *encoder* es convolucionada a través de dos bloques sucesivos formados por un conjunto *batch normalization* + *Conv3D* (función de activación ReLU,  $n$  filtros) + *batch normalization* seguido de un bloque residual *Conv3D* (función de activación ReLU,  $n$  filtros) + *batch normalization*. El número de filtros  $n$  equivale al de cada nivel: 32, 64 y 128, respectivamente.

- IV Tramo ascendente *decoder* con tres niveles: inicialmente, cada nivel dispone de una capa de *UpSampling* que aumenta las dimensiones de la entrada (factor  $f = 2$ ), y una capa *Conv3D* consecutiva (ReLU,  $n$  filtros). El resultado obtenido tras el *upsampling* se concatena con la salida de la conexión *shortcut* correspondiente a ese nivel, y el tensor resultante se convoluciona finalmente en dos bloques sucesivos *batch normalization* + *Conv3D* (función de activación ReLU,  $n$  filtros). En el último nivel del tramo ascendente, tras la concatenación, el tensor atraviesa un conjunto *batch normalization* + *Conv3D* (ReLU, 32 filtros) + *batch normalization*.
- V Un bloque residual *Conv3D* (función de activación ReLU, 32 filtros) + *batch normalization*
- VI Una capa convolucional de salida *Conv3D* (función de activación ReLU, 2 filtros).

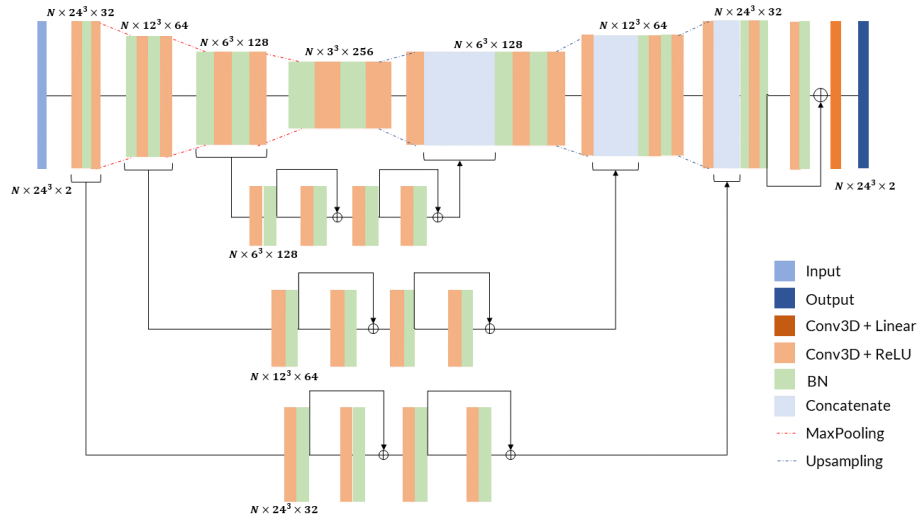


Figura 3.3: Esquema de la arquitectura de la red de síntesis T2 SR.

### 3.3.2 Fase de clasificación

El objetivo en la fase de clasificación es, a partir de las imágenes IRM potenciadas en T1 y T2 obtenidas de las primeras redes ( $I_{T1}^{LR}$ ,  $I_{T1}^{SR}$  y  $I_{T2}^{SR}$ ), determinar la probabilidad de que el sujeto presente o no la EA ( $p_{AD}$  y  $p_{CN}$ , respectivamente). En este caso, se propone una red ANN de clasificación, cuya transformación se expresa matemáticamente como:

$$[p_{CN}, p_{AD}] = \text{ClassNet} (I_{T1}^{LR}, [I_{T1}^{SR}, I_{T2}^{SR}]) \quad (3.5)$$

Se trata de una red CNN que recibe como entrada un tensor de 5 dimensiones  $N \times x \times y \times z \times c$  con la imagen o imágenes IRM de una o varias modalidades, y devuelve como salida un vector de 2 neuronas con las probabilidades de pertenencia a cada clase. La arquitectura de la red (Figura 3.4) consta de 27 130 898 parámetros - de los cuales 27 129 426 son entrenables - y se compone de:

- I Una capa convolucional de entrada *Conv3D* (función de activación ReLU, 16 filtros).

- II Tramo descendente *encoder* con cinco niveles: en este tramo la red extrae las características de los tensores de entrada mediante una serie de filtros convolucionales 3D, a la vez que reduce las dimensiones de los datos. Se trata de un tramo descendente similar al de la red *UNet* de la fase de SR, aunque con algunas variaciones. Cada nivel se compone de dos bloques sucesivos *batch normalization* + *Conv3D* (función de activación ReLU,  $n$  filtros) sucesivos, salvo en el primer nivel, que tan solo se compone de un bloque. El número de filtros  $n$  de convolución en cada nivel es de 16, 32, 64, 128 y 256. Al final de cada nivel, una capa de *MaxPooling* reduce las dimensiones a la mitad (factor de *downsampling*  $f = 2$ ). Como particularidad, en el segundo nivel, tras la capa de *MaxPooling*, se ha introducido una capa de *Dropout* con una probabilidad de bloqueo del 50 %, con la finalidad de regularizar la red.
- III Tramo *DenseNet*: en este tramo la red asigna una clase a los datos de entrada en función de las características extraídas en el tramo *encoder*. Inicialmente, se vectorizan los datos del último nivel del *encoder* mediante una capa *Flatten*, dando lugar a la capa de entrada de la *DenseNet*. La capa de entrada se conecta totalmente a una capa oculta de 512 neuronas, y ésta a una capa de salida de 2 neuronas. Las funciones de activación de las neuronas de las capas de entrada y oculta son ReLU, mientras que la de salida es Softmax.

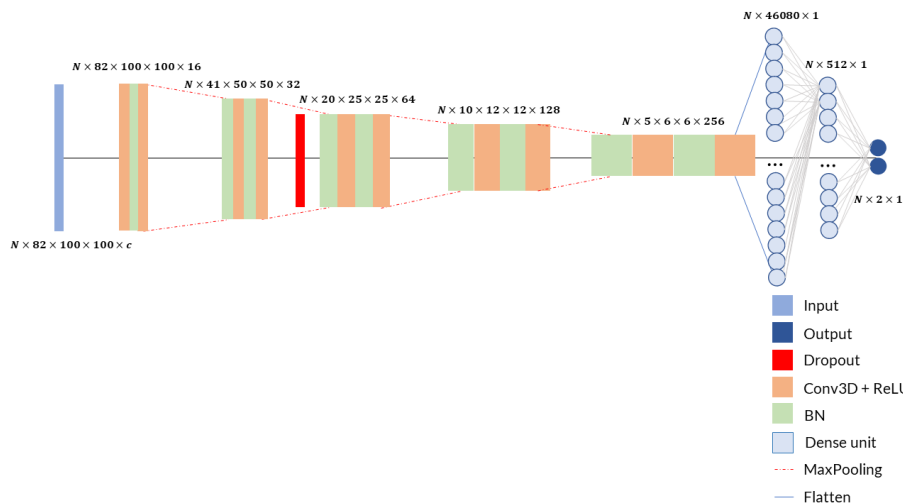


Figura 3.4: Esquema de la arquitectura de la red de clasificación.

### 3.4 Preprocesado y estructuración de los datos

El preprocesado de las imágenes se define como el conjunto de técnicas de transformación de las imágenes previas a su análisis con el objetivo de mejorar su calidad o preparar el conjunto de datos para una determinada aplicación. El preprocesado en el área de la imagen IRM engloba diferentes procesos: filtrado (reducción del ruido de la imagen), normalización de intensidad, corrección de inhomogeneidad, registro (transformación de la imagen a un sistema de coordenadas de referencia), superresolución, etc. Aunque parte de estas técnicas de preprocesado ya han sido aplicadas en las imágenes originales, es necesario llevar a cabo una serie de procesos adicionales para normalizar y adaptar los datos al problema a resolver.



### 3.4.1 Fase de SR

A partir de las 25 imágenes IRM potenciadas en T1 y T2 del primer conjunto de datos, se han aplicado inicialmente tres procesos siguiendo el protocolo propuesto por Romero, Coupé y Manjón 2017: filtrado adaptativo NLM (Manjón y col. 2010b), corrección de inhomogeneidad basada en el algoritmo N4 (Tustison y col. 2010) y registro lineal al espacio estándar MNI152. Durante el proceso de registro, las imágenes se han remuestreado mediante interpolación a una resolución de  $0,5 \text{ mm}^3$ . A continuación, se han extraído dos volúmenes o *crops* de dimensiones  $82 \times 100 \times 100$  de la zona del lóbulo temporal, en la que se encuentran las estructuras más afectadas por la EA (el hipocampo o HC y la corteza entorrinal o EC). Los dos volúmenes se corresponden con las regiones del HC derecho e izquierdo, ambas dispuestas en la misma orientación. Finalmente se ha aplicado un último filtrado NLM para suavizar los errores cometidos en el registro. Los *crops* de las imágenes se han almacenado en formato NIFTI. De este modo, se han obtenido 50 *crops* IRM potenciadas T1 y 50 *crops* IRM potenciadas en T2 (cada sujeto presenta dos *crops* de cada modalidad). Estos procesos se han programado y ejecutado con el *software* MATLAB 2018b y con la herramienta de registro ANTs. Los *crops* de las imágenes han sido almacenados en formato NIFTI.

En base a las arquitecturas de las redes CNN propuestas en la fase de SR, se han definido las etapas del preprocesado que se deben aplicar sobre los *crops* para obtener los datos de entrada y de salida de las dos redes. Se ha considerado que, como las operaciones en las capas de convolución 3D son computacionalmente muy costosas, las dimensiones de los volúmenes deben ser menores a las de los *crops*. Por este motivo, se ha decidido que los datos de entrada y de referencia de la red sean pequeños volúmenes o *patches* extraídos de los *crops*. A la hora de extraer los *patches*, es importante definir dos parámetros:

- **Dimensiones:** indican el tamaño del *patch*. En este caso, los *patches* son de  $24 \times 24 \times 24$  vóxeles.
- **Solapamiento:** determina el grado de solape entre dos *patches* contiguos. En este caso, el solapamiento es  $12 \times 12 \times 12$  vóxeles de separación entre *patches* (un 50%) por defecto <sup>2</sup>.

En primer lugar, para que los *crops* se hayan podido recorrer al completo durante la extracción de *patches*, se han añadido vóxeles adicionales en los bordes de la imagen en las tres dimensiones del espacio mediante la técnica de *padding*. El número de vóxeles añadidos en cada dimensión al *crop* ha sido igual al solapamiento (es decir, 12 vóxeles por dimensión). El valor de intensidad que han tomado los nuevos vóxeles es equivalente al valor de sus correspondientes vóxeles simétricos, tomando como ejes de simetría los bordes originales del volumen. Las dimensiones finales de los *crops* tras el proceso de *padding* son de  $94 \times 112 \times 112$  vóxeles.

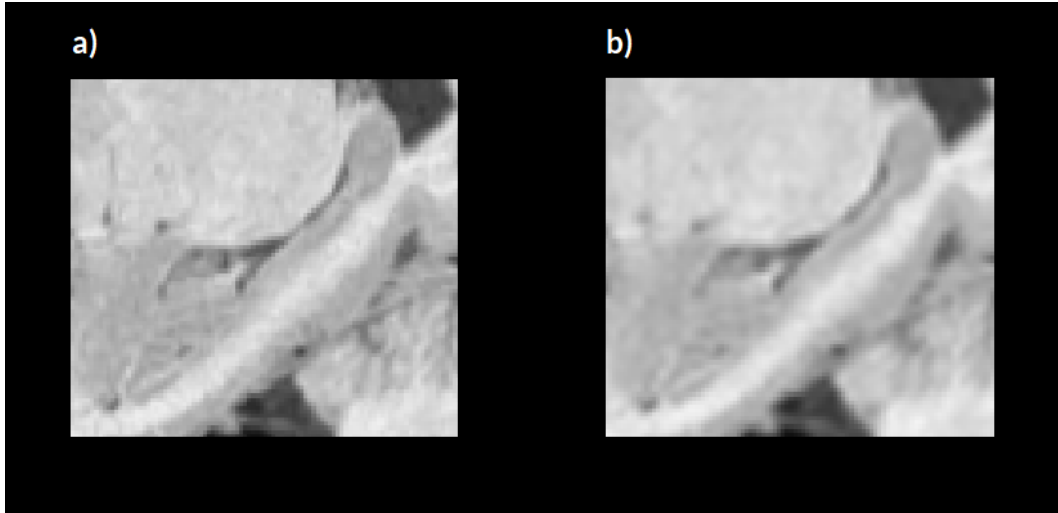
A continuación, se han obtenido en dos pasos los *crops* LR a partir de los *crops* HR <sup>3</sup>. Primero, se han reducido las dimensiones del volumen a la mitad promediando regiones de tamaño  $2 \times 2 \times 2$ , realizando de este modo un proceso de *downsampling*. En segundo lugar, se han recuperado las dimensiones originales aplicando un proceso de *upsampling* con factor  $f = 2$  (cada vóxel

---

<sup>2</sup>En la fase de experimentación se ha modificado el solapamiento en determinados experimentos para comprobar su efecto.

<sup>3</sup>La bajada de la resolución se ha aplicado únicamente para los *crops* potenciados en T1, ya que son necesarios para la red T1 SR como datos de entrada. En el caso de la red de síntesis T2 SR, la entrada son volúmenes T1, por lo que no es necesario bajar la resolución a los volúmenes T2.

se ha dividido en  $2 \times 2 \times 2$  vóxeles) en el que los valores de los vóxeles se han estimado por interpolación bicúbica (orden 3). Como resultado de los dos procesos sucesivos, el *crop* HR original pierde resolución y, visualmente, la imagen queda emborronada y pierde texturas y definición en los contornos, tal y como se observa en la Figura 3.5.



**Figura 3.5:** Cortes sagitales de un *crop* IRM potenciado en T1. a) Corte HR original. b) Corte LR.

Posteriormente, se ha aplicado una normalización de la intensidad en todos los *crops*, con el objetivo de que el aprendizaje de la red no dependa ni del valor medio ni de la desviación típica de la intensidad de los vóxeles del *crop*. Matemáticamente, la operación de normalización de una imagen se expresa como:

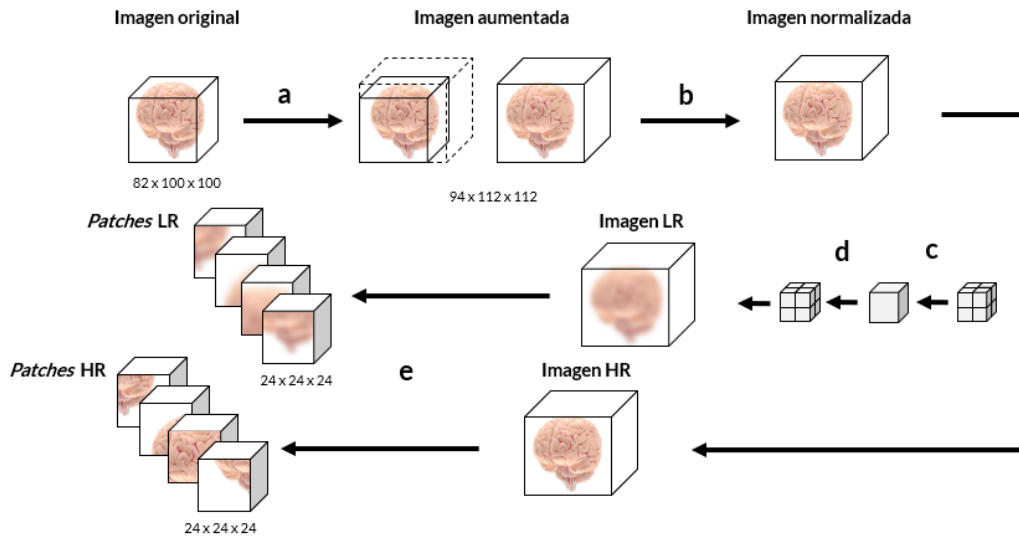
$$I_n = \frac{I - \mu_I}{\sigma_I} \quad (3.6)$$

Donde  $I_n$  es la imagen normalizada,  $I$  es la imagen original,  $\mu_I$  es la media de la intensidad de la imagen, y  $\sigma_I$  es la desviación típica de la intensidad de la imagen.

Como última fase del preprocesado de los *crops*, se han extraído los *patches* de los volúmenes LR y HR recorriendo cada uno de los *crops*. Por cada volumen recorrido se han generado 384 *patches* de dimensiones  $24 \times 24 \times 24$  vóxeles.

En cuanto a la estructuración y el formato de los datos, los *crops* se han cargado de uno en uno previamente a la fase de entrenamiento de la red y se les ha aplicado el preprocesado completo de forma individual (Figura 3.6). Los conjuntos de *patches* LR y HR que se han ido extrayendo de cada volumen se han ido concatenando en dos grandes vectores, respectivamente. Cada vector consta de 5 dimensiones:  $N \times 24 \times 24 \times 24 \times 1$ , donde  $N$  indica el número de elementos o *patches*, acumulados en el vector, las dimensiones intermedias ( $24^3$ ) se corresponden con las dimensiones de los *patches*, y la última dimensión indica que las imágenes T1 son monocanal. Todas las transformaciones del preprocesado que se han aplicado desde el *crop* original hasta la obtención de los *patches* se han programado y ejecutado en lenguaje Python. Para la carga de los volúmenes en formato NIfTI se han empleado las funciones de la librería **nibabel**, mientras que los procesos de modificación de los vectores (*padding*, normalización de intensidad, *downsampling*

y *upsampling*, extracción de *patches*...) se han ejecutado con las librerías **numpy**, **skimage** y **scipy**. Los datos de los volúmenes se han representado en formato **float32**.



**Figura 3.6: Esquema del preprocesado de los *crops* en la fase de SR.** Las etapas del preprocesado son: a) *Padding*. b) Normalización de intensidad. c) *Downsampling* (solo en datos T1). d) *Upsampling* (solo en datos T1). e) Extracción de *patches*.

### 3.4.2 Fase de clasificación

Como ya se ha comentado en la Subsección 3.1.1, la procedencia de las imágenes de la base de datos ADNI y las técnicas que se han aplicado sobre ellas son muy variables. Por ello, la fase del preprocesado para su normalización resulta fundamental en este caso.

Inicialmente, todo el conjunto de imágenes IRM potenciadas en T1 de la fase de clasificación ha sido filtrado NLM (Manjón y col. 2010b) y registrado linealmente al espacio MNI152. Seguidamente, se ha aplicado una corrección de inhomogeneidad basada en el algoritmo N4 (Tustison y col. 2010) y una normalización de intensidad basada en las medias de las intensidades de cada tejido (Romero y col. 2015). Finalmente, se han extraído dos volúmenes o *crops* de dimensiones  $82 \times 100 \times 100$  vóxeles de la zona del lóbulo temporal, y se han remuestreado a una resolución de  $0,5 \text{ mm}^3$  aplicando interpolación bicúbica. Los procesos se han ejecutado con el *software* MATLAB 2018b y la herramienta de registro ANTs. Los *crops* de las imágenes han sido almacenados en formato NIfTI.

A diferencia de las redes de SR, se ha supuesto que red de clasificación requiere un análisis completo de toda la información de la imagen para poder clasificar al paciente, por lo que, en este caso, los dos *crops* LR de cada sujeto serán los datos de entrada de la red (no es necesario extraer *patches* de los *crops*). La única técnica de preprocesado que se ha aplicado sobre los *crops* previo al entrenamiento y validación de la red, ha sido la normalización de intensidad expresada en la Ecuación (3.6).

Al igual que en la fase de SR, los dos *crops* de cada sujeto se han cargado progresivamente durante la fase de entrenamiento, se han preprocesado individualmente, y se han almacenado en un vector 5-dimensional  $N \times 82 \times 100 \times 100 \times c$ , donde  $N$  es el número de elementos del

vector, las dimensiones intermedias ( $82 \times 100 \times 100$ ) son las dimensiones de los *crops*, y la última dimension indica los canales de la imagen. Las transformaciones del preprocesado que se han aplicado sobre los *crops* se han programado y ejecutado en lenguaje Python. La carga de los volúmenes en formato NIfTI se han empleado las funciones de la librería **nibabel**, mientras que el proceso de normalización de intensidad se ha llevado a cabo con las librería **numpy**. Los datos de los volúmenes se han representado en formato **float32**.

### 3.5 Entrenamiento, validación y evaluación de las redes neuronales artificiales

Una vez las ANN han sido diseñadas y los datos han sido preprocesados, se han procedido a la selección de los conjuntos de datos y a la ejecución (fase de entrenamiento o *train*) y evaluación (fase de evaluación o *test*) de los algoritmos. En este punto se han establecido los parámetros de las ANN, se han diseñado y programado los experimentos, y se han evaluado las diferentes métricas de cada problema. La implementación de las etapas de entrenamiento y evaluación se han llevado a cabo con el lenguaje de programación Python, y con las librerías de DL **tensorflow** y **keras**.

#### 3.5.1 Fase de SR

En la fase de SR, se han seleccionado un 60 % de los datos para la fase de entrenamiento, y un 40 % para la fase de evaluación. Del conjunto de entrenamiento, un 10 % se ha empleado en la validación durante el entrenamiento de las redes. Tras el preprocesado de los 50 *crops* IRM de cada modalidad (T1 y T2), se han obtenido los siguientes datos:

- 19 200 *patches* potenciados en T1 LR normalizados: son los datos de entrada de la red T1 SR.
- 19 200 *patches* potenciados en T1 HR normalizados: son los datos de referencia de la red T1 SR.
- 19 200 *patches* potenciados en T2 HR normalizados: son los datos de referencia de la red de síntesis T2 SR.

De los 19 200 *patches* de cada grupo, siguiendo las proporciones establecidas, se han dedicado 10 368 *patches* (27 *crops*) al entrenamiento, 1 152 a la validación (3 *crops*), y 7 680 al *test* (20 *crops*).

Con respecto al orden de experimentación, se han ejecutado en primer lugar los entrenamientos de la red T1 SR. Como datos de entrada, se han tomado los *patches* IRM potenciados en T1 LR y, como datos de referencia en la salida, los residuos entre los *patches* LR y HR. Una vez entrenada y evaluada la primera red, se han generado los volúmenes IRM potenciados en T1 SR con la misma, y se ha entrenado la segunda red de síntesis T2 SR. Como datos de entrada, la red ha recibido los *patches* IRM potenciados en T1 LR y SR en dos canales y, como datos de referencia en la salida, los residuos entre los *patches* potenciados en T1 SR y potenciados en T2 HR.

### *Entrenamiento y validación*

El entrenamiento de las redes se ha ejecutado con el optimizador Adam de la librería **keras**. Dicho optimizador se basa en el algoritmo *Gradient Descent* estocástico propuesto por Kingma y Ba 2014. Se ha establecido un *learning rate* inicial de  $10^{-3}$  en todos los entrenamientos. En cuanto a la función de coste, se han propuesto varias funciones en los experimentos realizados (véase Tabla 1.3):

- MAE o norma L1.
- GDL: se ha utilizado la versión propuesta en Mathieu, Couprie y LeCun 2015 y se ha adaptado a imágenes 3D.
- *Structural Dissimilarity* (DSSIM): función inversa a la métrica SSIM.  $DSSIM = \frac{1}{2}(1 - SSIM)$ .
- Combinación ponderada de las funciones anteriores.

Las dos redes de la fase de SR se han entrenado con la función *fit()* de la librería **keras** durante 120 *epochs*. En cada *epoch* se ha recorrido todo el conjunto de entrenamiento seleccionando de forma aleatoria pequeños conjuntos o *batches* de 32 elementos (*batch size* = 32). Al final de cada *epoch*, se ha realizado la validación y se ha almacenado el modelo y los pesos que menor *loss* de validación ha presentado durante todo el entrenamiento.

Finalmente, una vez cargados los datos y definidos los parámetros, se ha compilado el modelo de red y se ha ejecutado los entrenamientos.

### *Evaluación*

Para la fase de *test* se han seguido los siguientes pasos en las dos redes de SR desarrolladas: preprocesado, predicción, reconstrucción y estimación de métricas.

Para cada red, primeramente se han cargado uno a uno los *crops* IRM LR y HR. Cada *crop* de *test* ha sido preprocesado y se han obtenido sus correspondientes *patches*. Con ellos, se ha ejecutado la predicción de la red mediante la función *predict()* de **keras**. Posteriormente, se han llevado a cabo los procesos inversos del preprocesado para recuperar la imagen IRM en el espacio inicial:

1. Reconstrucción del *crop* a partir de los *patches*. El solapamiento durante la extracción de *patches* y su reconstrucción en la fase de *test* no tiene por qué ser idéntico al de entrenamiento. Por defecto, el solapamiento es de 12 vóxeles (50%).
2. Proceso inverso a la normalización de intensidad:  $I = I_n \sigma_I + \mu_I$ .
3. Eliminación de los vóxeles añadidos en el *padding*. Recuperación de las dimensiones originales de la imagen.
4. Almacenamiento de los *crops* de SR en formato NIfTI

Finalmente se han calculado diferentes métricas con el fin de evaluar la calidad de los resultados de SR:

- I. Red T1 SR: las métricas se han obtenido entre los *crops* T1 LR - HR (como referencia) y SR - HR, siendo los *crops* HR la referencia. Se han calculado tres métricas: PSNR, coeficiente de correlación y SSIM.
- II. Red de síntesis T2 SR: las métricas se han obtenido entre los *crops* T1 HR - T2 HR (como referencia) y T2 SR - T2 HR, siendo los *crops* T2 HR la referencia. Se han calculado tres métricas: PSNR, coeficiente de correlación y SSIM.

Para cada experimento, se han registrado las distintas métricas, se han comparado con experimentos anteriores, y se han añadido las modificaciones sobre los modelos que mejores resultados han presentado.

### *Experimentos diseñados*

A continuación se detallan los experimentos llevados a cabo sobre los modelos de las dos redes de síntesis propuestos en la fase de SR:

#### I. Red T1 SR:

- 1) Entrenamiento de la red empleando la función de coste  $F = MAE$ .
- 2) Entrenamiento de la red empleando la función de coste  $F = MAE + \alpha GDL$ , donde  $\alpha$  indica el peso de la función de coste GDL.
- 3) Aplicación de la técnica *mean correction*. Dicha técnica estima la media de regiones de  $2^3$  vóxeles en los volúmenes LR, calcula la diferencia u *offset* con los volúmenes SR y, finalmente, se suma la diferencia a la imagen SR para corregir el error de media cometido por la red.
- 4) Modificación del solapamiento en la reconstrucción de *patches* a 24 (0%) y 6 (75%) vóxeles.

#### II. Red de síntesis T2 SR:

- 1) Entrenamiento de la red de síntesis T2 SR con una capa de regularización *Dropout* (al 50%) en el último nivel del tramo *encoder* y función de coste  $F = MAE$ .
- 2) Entrenamiento de la red de síntesis T2 SR con una capa de regularización *Dropout* (al 50%) en el último nivel del tramo *encoder* y función de coste  $F = DSSIM$ .
- 3) Entrenamiento de la red de síntesis T2 SR con una capa de regularización *Dropout* (al 50%) en el último nivel del tramo *encoder* y función de coste  $F = MAE + DSSIM$ .
- 4) Entrenamiento de la red *UNet* sin capas *Dropout*<sup>4</sup> y función de coste  $F = MAE + DSSIM$ .
- 5) Modificación del solapamiento en la reconstrucción de *patches* a 24 y 6 vóxeles de separación entre *patches* (0% y 75%, respectivamente).

---

<sup>4</sup>Esta arquitectura de red se muestra en la Figura 3.3.

### 3.5.2 Fase de clasificación

En la fase de clasificación, se han seleccionado un 72 % de los datos para la fase de entrenamiento, un 10 % para la validación y un 18 % para la fase de *test*. Los tres conjuntos han sido seleccionados de forma balanceada, es decir, que el número de sujetos sanos (CN) y enfermos (AD) en cada conjunto ha sido el mismo o similar.

El balanceo de los datos se ha realizado a partir de la información contenida en el archivo CSV. En primer lugar, se ha leído el archivo y se ha almacenado la información relevante (en nuestro caso, identificadores y etiquetas) en una tabla o *dataframe*. Seguidamente, se han seleccionado de forma aleatoria los casos de entrenamiento, validación y *test*. De los 412 sujetos disponibles - 226 CN y 186 AD -, se han escogido 130 sujetos AD y 170 sujetos CN para la fase de entrenamiento, 19 sujetos AD y 19 sujetos CN para validación, y 38 sujetos AD y 38 sujetos CN para la fase de *test*. La agrupación de los datos de los conjuntos de datos se muestra en la Tabla 3.2.

	AD	CN	Total
<b>Entrenamiento</b>	130	170	<b>300</b>
<b>Validación</b>	19	19	<b>38</b>
<b><i>Test</i></b>	37	37	<b>74</b>
<b>Total</b>	<b>186</b>	<b>226</b>	<b>412</b>

Tabla 3.2: Conjuntos de entrenamiento, validación y *test* de la fase de clasificación.

Para cada sujeto se dispone de tres pares de volúmenes IRM: T1 LR, T1 SR y T2 SR. Cada par está formado por los *crops* de las regiones izquierda y derecha del HC. Los volúmenes NIfTI de los *crops* de ADNI preprocesados se han organizado en tres directorios - *train*, *val* y *test* - y, dentro de cada uno de ellos, en dos directorios más: AD y CN.

#### *Entrenamiento*

El entrenamiento de las redes se ha ejecutado con el optimizador Adam de la librería **keras**. Se ha establecido un *learning rate* inicial de  $10^{-4}$  en todos los entrenamientos. En cuanto a la función de coste, se han empleado en todos los casos la función *categorical\_crossentropy*.

Las dos redes de la fase de SR se han entrenado con la función *fit\_generator()* de la librería **keras**, la cual requiere de una función generadora. Las funciones generadoras desarrolladas escogen en cada paso o *step* un número de sujetos o *batch size* de entrenamiento de forma aleatoria a través de la lista de identificadores de la que se dispone. Seguidamente, cargan los *crops* (HC derecho e izquierdo) asociados al sujeto, se preprocesan (aplicando la normalización de intensidad) y se almacenan en un vector **numpy**. Finalmente, se asignan las clases o etiquetas de salida:  $[1, 0]$  para sujetos de la clase CN y  $[0, 1]$  para sujetos de la clase AD. La función generadora repite continuamente este proceso. El *batch size* que se ha empleado en todos los experimentos ha sido de *batch size* = 6, ya que los vectores de entrada son de grandes dimensiones y multicanal y, por lo tanto, computacionalmente costosos de operar.

El número de *steps* de la función generadora por *epoch* debe ser tal que se recorra todo el conjunto de entrenamiento. Por lo tanto:  $steps\ per\ epoch = \frac{Casos\ train}{Batch\ size} = \frac{300}{6} = 50$ . Se han establecido un total de 200 *epochs* y, al final de cada *epoch*, se ha realizado la validación y se ha almacenado el modelo y los pesos que menor *loss* de validación ha presentado durante todo el entrenamiento.

Finalmente, definidos los parámetros y las funciones generadoras, se ha compilado el modelo de red y se ha ejecutado los entrenamientos.

### *Evaluación*

Para la fase de *test* de la red de clasificación se han seguido los siguientes pasos: preprocesado, predicción y estimación de métricas.

En primer lugar, se han cargado los dos *crops* IRM de ADNI asociados a cada sujeto del grupo de *test* y se han normalizado sus intensidades. A continuación, se ha llevado a cabo la predicción con la red entrenada mediante la función *predict()* de **keras**, la cual devuelve para cada sujeto una probabilidad de pertenecer a una clase: AD o CN.

Por último, se han calculado diferentes métricas con el fin de evaluar la calidad de los resultados de SR. Las métricas se han calculado con las probabilidades predichas por la red y las etiquetas verdaderas de cada sujeto ( $[1, 0]$  para CN o  $[0, 1]$  para AD). Se han calculado las siguientes métricas: precisión o *accuracy*, sensibilidad, especificidad y ROC AUC.

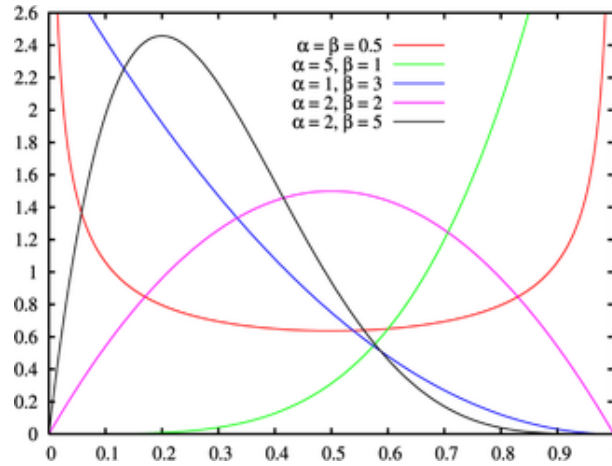
Al igual que en las redes de SR, para cada experimento se han registrado las distintas métricas, se han comparado con experimentos anteriores, y se han añadido las modificaciones sobre los modelos que mejores resultados han presentado.

### *Experimentos diseñados*

A continuación se detallan los experimentos llevados a cabo sobre los modelos de la red de clasificación propuestos en esta fase:

- 1) Entrenamiento de la red de clasificación con dos capas *Dropout* al 50% en los dos últimos niveles del tramo *encoder*.
- 2) Entrenamiento de la red de clasificación con dos capas *Dropout* al 50% en los dos últimos niveles del tramo *encoder* aplicando la técnica de *label smoothing*. Esta técnica consiste en “suavizar” las etiquetas de salida dándoles valores cercanos a 0 y 1 (por ejemplo, en rangos de  $[0, 0,05]$  y  $[0,95, 1]$ , respectivamente), de forma que la diferencia de gradiente entre las dos clases no sea tan brusco durante el entrenamiento.
- 3) Entrenamiento de la red de clasificación con dos capas *Dropout* al 50% en los dos últimos niveles del tramo *encoder* aplicando la técnica de *mixup*. Esta técnica genera nuevos casos ponderando dos *crops* de entrenamiento aleatorios con dos pesos  $\omega_1, \omega_2 \in [0, 1]/\omega_1 + \omega_2 = 1$ . Para calcular los pesos, se selecciona un número aleatorio  $p \in [0, 1]$  de una distribución de probabilidad beta (Figura 3.7) con parámetros  $\alpha = \beta$  y se establece que  $\omega_1 = p$  y  $\omega_2 = 1 - p$ . Valores bajos de  $\alpha$  dan lugar a que los pesos sean más extremos y que los nuevos casos no sean sufran demasiada mezcla, y viceversa.
- 4) Entrenamiento de la red de clasificación con dos capas *Dropout* al 50% en los dos últimos niveles del tramo *encoder* y sustituyendo el *MaxPooling* por *strides=2<sup>3</sup>* y aplicando la técnica de *mixup*. El número de filtros se ha reducido a 12 para evitar que la red presente demasiados parámetros.





**Figura 3.7:** Gráfica con distribuciones de probabilidad beta con diferentes parámetros  $\alpha$  y  $\beta$ . Puede apreciarse que cuando  $\alpha = \beta$ , la distribución es simétrica.

- 5) Entrenamiento de la red de clasificación con una capa *Dropout* en el segundo nivel del tramo *encoder* y aplicando la técnica *mixup*.

Todos los experimentos se han ejecutado tomando como datos de entrada los pares de *crops* T1 LR (HC derecho e izquierdo) en dos canales de entrada ( $c = 2$ ). Una vez seleccionado el mejor experimento, la red se ha evaluado con los *crops* potenciados en T1 SR generados por la primera red T1 SR. Finalmente, se ha entrenado de nuevo el mismo experimento tomando esta vez como datos de entrada nuevos pares de *crops* *crops* T1 SR y T2 SR en cuatro canales ( $c = 4$ ). Con ello, se comprueba el efecto de introducir información de otra modalidad IRM en la red de clasificación.



# Resultados y discusión

*En este capítulo se presentarán e interpretarán los resultados obtenidos en cada uno de los experimentos en las dos fases del proyecto: SR y clasificación. Dichos experimentos serán validados mediante una serie métricas y de pruebas estadísticas, y posteriormente discutidos y comparados con otros métodos del estado del arte. Asimismo, se visualizarán las imágenes generadas por las redes de síntesis y su posterior clasificación.*

Una vez finalizada la fase de entrenamiento y validación, se han evaluado los resultados de las redes entrenadas en los experimentos de las dos fases - SR y clasificación - y se han extraído las métricas correspondientes para poder comparar las modificaciones realizadas en la experimentación y contrastar con otros métodos. A continuación se detallan los resultados obtenidos de los experimentos de cada fase.

### 4.1 Fase de SR

Los resultados de las métricas de la fase de SR se han presentado en tres formatos:

- Tablas de las medias  $\pm$  desviaciones de las métricas calculadas sobre el conjunto de *test*: PSNR, coeficiente de correlación y SSIM. En este tipo de problemas, la métrica más extendida para comparar métodos es el PSNR.
- Diagramas de caja o *box-plot* de los experimentos.
- Gráficas del valor de las métricas por cada caso evaluado individualmente.

Para determinar las variaciones significativas entre los conjuntos de métricas de los casos generados por la red y su referencia, se han realizado una serie de pruebas estadísticas. También se han extraído y analizado visualmente diferentes cortes de las imágenes IRM SR potenciadas en T1 y T2 generadas por las redes. Finalmente, se han comparado las redes de SR con los métodos del estado del arte que afrontan el mismo problema.

#### 4.1.1 Red T1 SR

En la Tabla 4.1 se resumen las medias y desviaciones de las métricas de cada experimento evaluadas sobre el conjunto de *test*: 20 *crops* potenciados en T1 pertenecientes a 10 sujetos. Se han tomado como referencias las métricas estimadas entre los *crops* T1 LR y T1 HR con diferentes solapamientos en reconstrucción <sup>1</sup>.

Experimento	Características	PSNR (dB)	Coefficiente de correlación	SSIM
Referencias	Solapamiento: 0%	31,228 ± 0,403	0,986 ± 0,002	0,884 ± 0,006
	Solapamiento: 50%	31,186 ± 0,384	0,986 ± 0,007	0,883 ± 0,007
	Solapamiento: 75%	31,100 ± 0,372	0,985 ± 0,002	0,881 ± 0,007
Experimento 1	MAE	35,881 ± 0,415	0,998 ± 0,004	0,934 ± 0,004
Experimento 2	MAE + 0.3 GDL	35,888 ± 0,403	0,995 ± 0,001	0,935 ± 0,004
	MAE + 0.5 GDL	35,913 ± 0,407	0,995 ± 0,001	0,935 ± 0,003
	MAE + GDL	35,813 ± 0,414	0,994 ± 0,001	0,935 ± 0,004
Experimento 3	Mean correction	36,012 ± 0,412	0,995 ± 0,001	0,936 ± 0,004
Experimento 4	Solapamiento: 0%	35,853 ± 0,415	0,995 ± 0,001	0,935 ± 0,004
	Solapamiento: 75%	<b>36,023 ± 0,405</b>	<b>0,995 ± 0,001</b>	<b>0,936 ± 0,004</b>

Tabla 4.1: Media ± desviación típica de las métricas resultantes de los experimentos de las redes de síntesis SR T1.

Por otro lado, los diagramas *box-plot* de las Figuras 4.1, 4.2 y 4.3 muestran gráficamente la evolución de las métricas - PSNR, coeficiente de correlación y SSIM, respectivamente - en cada experimento realizado.

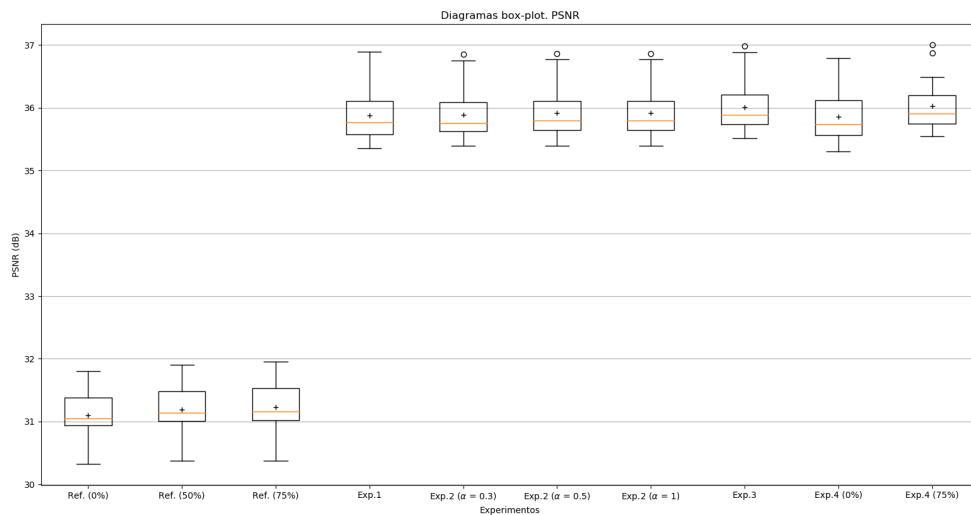


Figura 4.1: Diagramas *box-plot* de los valores PSNR de los casos de *test* de los casos en los diferentes experimentos de la fase de SR.

<sup>1</sup>En todos los experimentos, salvo que se indique lo contrario, la extracción de *patches* (de dimensiones  $24^3$  vóxeles) y la reconstrucción de la imagen se han realizado por defecto con un solapamiento de 12 vóxeles, al 50%.

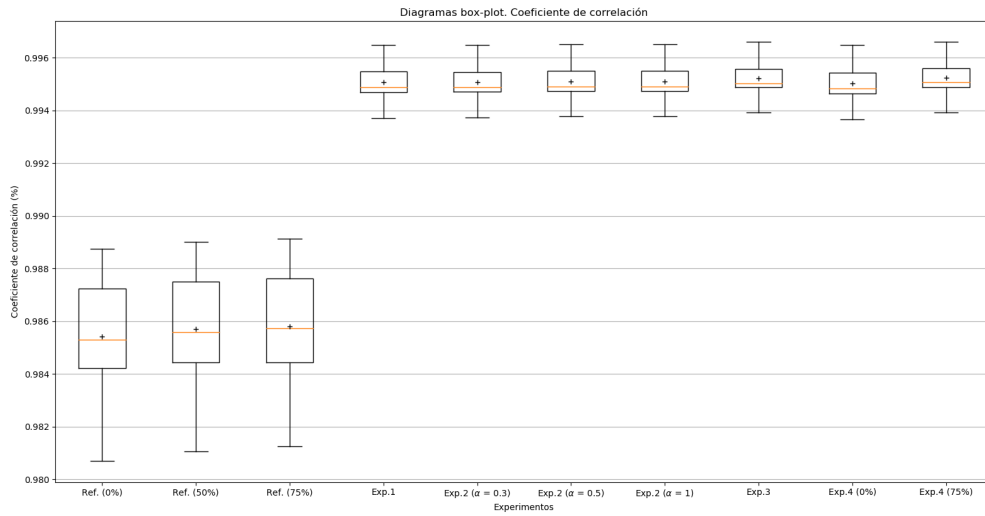


Figura 4.2: Diagramas *box-plot* de los coeficientes de correlación de los casos de *test* en los diferentes experimentos de la fase de SR.

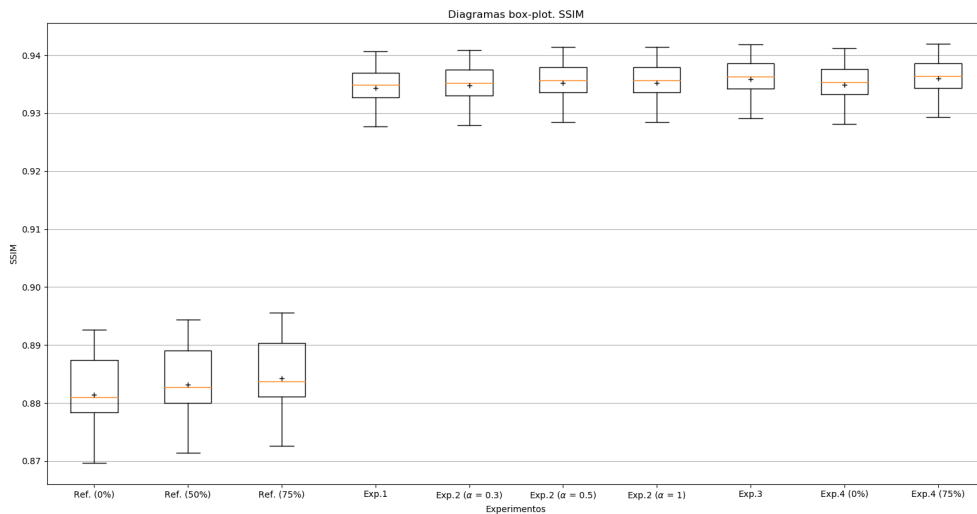


Figura 4.3: Diagramas *box-plot* de los índices SSIM de los casos de *test* en los diferentes experimentos de la fase de SR.

En referencia a las modificaciones llevadas a cabo, cabe resaltar los siguientes aspectos:

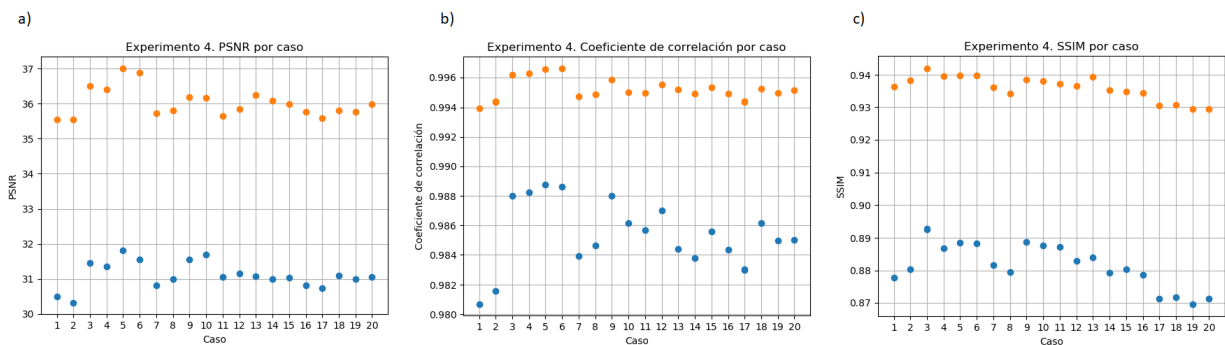
- El uso de una función de coste que combina las funciones MAE y GDL (Experimento 2) permite incluir información de gradiente (bordes, texturas...) a la distancia absoluta entre los valores de intensidad de los vóxeles. Esta modificación en la función de coste se traduce en una ganancia 0,025 dB de PSNR con respecto a la red entrenada con la función de coste MAE (Experimento 1).

- La técnica de corrección de la media o *mean correction* (Experimento 3) supone una ganancia en el PSNR de 0,099 dB con respecto al resto de redes que no la aplican (Experimentos 1 y 2).
- Empleando la misma red entrenada con *patches* extraídos con un solapamiento al 50 %, un mayor porcentaje de solapamiento en la reconstrucción en la fase de *test* - en este caso, un 75 % - conlleva una mejora de 0,011 dB con respecto a una reconstrucción con solapamiento del 50 % (Experimento 3), ya que existen más *patches* extraídos que reconstruir y, por lo tanto, más información. Sin embargo, un mayor número de *patches* incrementa el tiempo de la predicción.

A partir de los resultados presentados, se determina que la mejor red es la obtenida en el Experimento 4, cuya mejora global con respecto a su Referencia (al 75 % de solapamiento) es de:

- 4,923 dB de ganancia en PSNR.
- 0,010 (10 %) de ganancia en coeficiente de correlación.
- 0,055 (5,5 %) de ganancia en SSIM.

La evolución de las métricas en cada caso del conjunto de *test* en el Experimento 4 se muestra en las gráficas de la Figura 4.4.



**Figura 4.4:** Gráficas de las métricas calculadas individualmente para cada caso de *test* en el Experimento 4. a) PSNR. b) Coeficiente de correlación. c) SSIM.

Para comprobar si las métricas siguen una distribución normal, se ha aplicado la prueba de normalidad de Shapiro-Wilk sobre las dos muestras de 20 métricas de la Referencia y del Experimento 4. En esta prueba estadística se asume como hipótesis nula  $H_0$  que los datos siguen una distribución normal, y como hipótesis alternativa  $H_1$  que los datos no siguen una distribución normal. Los resultados de la prueba se muestran en la Tabla 4.2.

A partir de los p-valores obtenidos en la prueba, se rechaza con una probabilidad de error inferior al 1 % (nivel de significación  $\alpha = 0,01$ ) la hipótesis alternativa  $H_1$  en las dos muestras. En consecuencia, puede afirmarse que las métricas en ambas muestras siguen una distribución normal.

		Prueba de normalidad Shapiro-Wilk		
		PSNR	Coefficiente de correlación	SSIM
Referencia	Estadístico $W$	0,965	0,961	0,941
	p-valor	0,645	0,555	0,261
Experimento 4	Estadístico $W$	0,898	0,940	0,933
	p-valor	0,037	0,243	0,175

Tabla 4.2: Resultados de la prueba de normalidad Shapiro-Wilk aplicado a las muestras de las métricas de la Referencia y el Experimento 4. El número de muestras de cada grupo es de  $n = 20$ .

Una vez garantizada la normalidad de los datos, se comprueba si existen diferencias significativas entre las medias de dos muestras independientes de las 20 métricas obtenidas en el Experimento 4 y en la Referencia mediante un T-Test pareado de comparación de medias. Esta prueba estadística paramétrica asume como hipótesis nula  $H_0$  que no existe una diferencia significativa entre las medias de ambos grupos y, como hipótesis alternativa  $H_1$ , que dicha diferencia sí que existe y es significativa. Los resultados de la prueba se muestran en la Tabla 4.3.

T-Test pareado de comparación de medias			
	PSNR	Coefficiente de correlación	SSIM
Estadístico T	38,983	18,439	31,805
p-valor	$2,97 \times 10^{-32}$	$1,50 \times 10^{-32}$	$5,41 \times 10^{-29}$

Tabla 4.3: Resultados del T-Test pareado de comparación de medias entre las dos muestras de métricas de la Referencia y el Experimento 4. El número de muestras de cada grupo es de  $n = 20$ .

Los p-valores obtenidos permiten aceptar con una probabilidad de error inferior al 1% (nivel de significación  $\alpha = 0.01$ ) la hipótesis alternativa  $H_1$  para todos los casos. En otras palabras, existe una diferencia significativa entre las métricas obtenidas en el Experimento 4 a partir de los casos generados por la red y las métricas de los casos de referencia.

Las imágenes IRM SR potenciadas en T1 obtenidas en el Experimento 4 se muestran en la Figura 4.5. Visualmente, se observa que las imágenes IRM SR generadas por la red definen los contornos y las texturas de las estructuras que se encuentran borrosos en las imágenes IRM LR. Por otro lado, los casos SR no presentan el ruido propio de las imágenes originales IRM HR (puesto que la red no aprende ruido), de modo que la imagen adquiere un aspecto más suavizado y artificial que la original.

Finalmente, se ha comparado el método de síntesis de imágenes IRM T1 SR desarrollado con las referencias del estado del arte calculando la ganancia o diferencia de las métricas obtenidas por el método y las de referencia (en todos los casos, el método de referencia ha sido el *BSplines*). La comparación se muestra en la Tabla 4.4.

Tras la comparación de ganancias queda constancia de que, a excepción del estudio propuesto por Chen y col. 2018 y Manjón y col. 2010a, la red de SR T1 propuesta supera al resto de referencias consultadas.

<sup>2</sup>Debe tenerse en cuenta que en Manjón y col. 2010a se emplea una imagen de referencia IRM HR en la reconstrucción, por lo que la ganancia de PSNR es muy elevada. El resto de métodos generan la imagen IRM SR a partir de la imagen LR sin ninguna fuente de información HR.

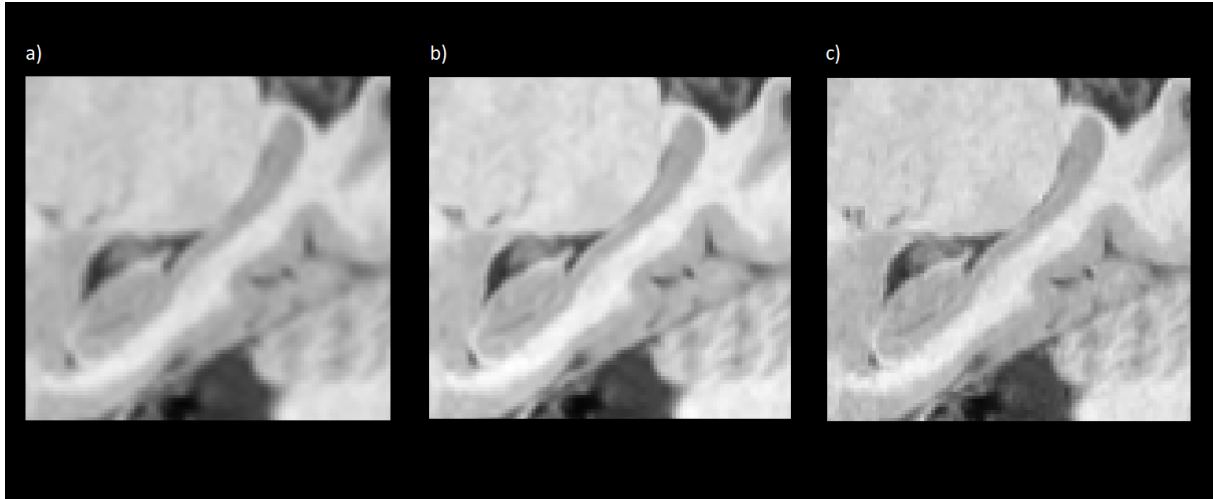


Figura 4.5: Cortes sagitales de tres *crops* IRM potenciados en T1 pertenecientes a un sujeto del conjunto de *test*. a) *Crop* LR. b) *Crop* SR generado por la mejor red (Experimento 4). c) *Crop* HR.

Método	Ganancia Método - Referencia ( <i>BSpline</i> )	
	PSNR (dB)	SSIM
Manjón y col. 2010b	3,28	0,0088
Pham y col. 2017	3,36	0,0330
Pham, Fablet y Rousseau 2017	4,50	0,0393
Chen y col. 2018	6,81	0,105
Manjón y col. 2010a	12,56 <sup>2</sup>	-
<b>Propuesto</b>	4,923	0,055

Tabla 4.4: Comparación del método propuesto de síntesis de imágenes IRM T1 SR con algunos de los métodos de SR del estado del arte.

#### 4.1.2 Red de síntesis T2 SR

En la Tabla 4.5 se resumen las medias y desviaciones de las métricas de cada experimento evaluadas sobre el conjunto de *test*: 20 *crops* potenciados en T2 pertenecientes a 10 sujetos. Se han tomado como referencias las métricas estimadas entre los *crops* T1 HR y T2 HR con diferentes solapamientos en reconstrucción <sup>3</sup>.

Por otro lado, los diagramas *box-plot* de las Figuras 4.6, 4.7 y 4.8 muestran gráficamente la evolución de las métricas - PSNR, coeficiente de correlación y SSIM, respectivamente - en cada experimento realizado.

<sup>3</sup>En todos los experimentos, salvo que se indique lo contrario, la extracción de *patches* (de dimensiones  $24^3$  vóxeles) y la reconstrucción de la imagen se han realizado por defecto con un solapamiento de 12 vóxeles, al 50%.



Experimento	Características	PSNR	Coefficiente de correlación	SSIM
Referencia	-	$5,631 \pm 0,398$	$-0,658 \pm 0,080$	$-0,302 \pm 0,034$
Experimento 1	MAE	$22,730 \pm 1,415$	$0,885 \pm 0,041$	$0,533 \pm 0,035$
Experimento 2	DSSIM	$20,668 \pm 1,036$	$0,813 \pm 0,038$	$0,414 \pm 0,028$
Experimento 3	MAE + DSSIM	$22,812 \pm 1,436$	$0,887 \pm 0,040$	$0,547 \pm 0,034$
Experimento 4	Sin <i>Dropout</i>	$22,847 \pm 1,429$	$0,888 \pm 0,041$	$0,549 \pm 0,035$
Experimento 5	Solapamiento: 0% Solapamiento: 75%	$22,206 \pm 1,297$ <b><math>23,002 \pm 1,458</math></b>	$0,871 \pm 0,041$ <b><math>0,892 \pm 0,041</math></b>	$0,527 \pm 0,032$ <b><math>0,552 \pm 0,036</math></b>

Tabla 4.5: Media  $\pm$  desviación típica de las métricas resultantes de los experimentos de las redes de síntesis SR T2.

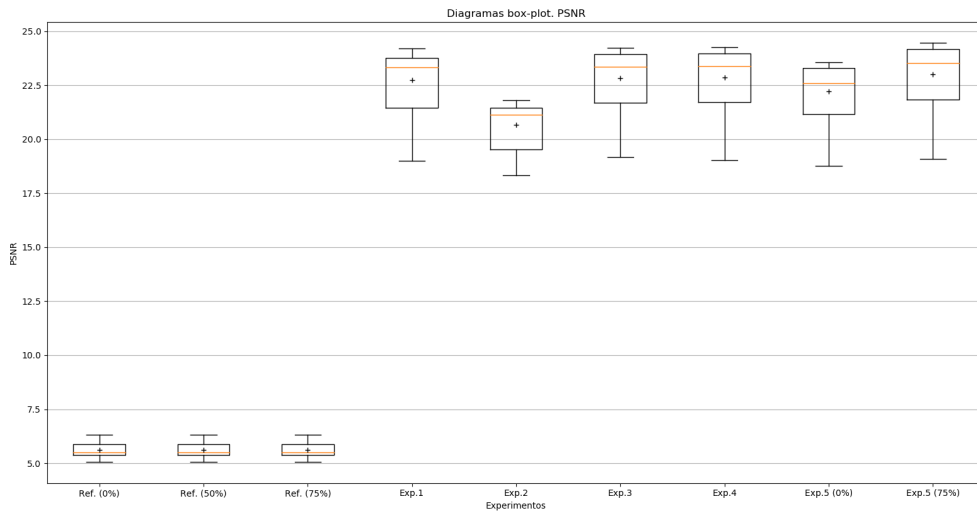


Figura 4.6: Diagramas *box-plot* de los PSNR de los casos de *test* en los diferentes experimentos de la fase de SR.

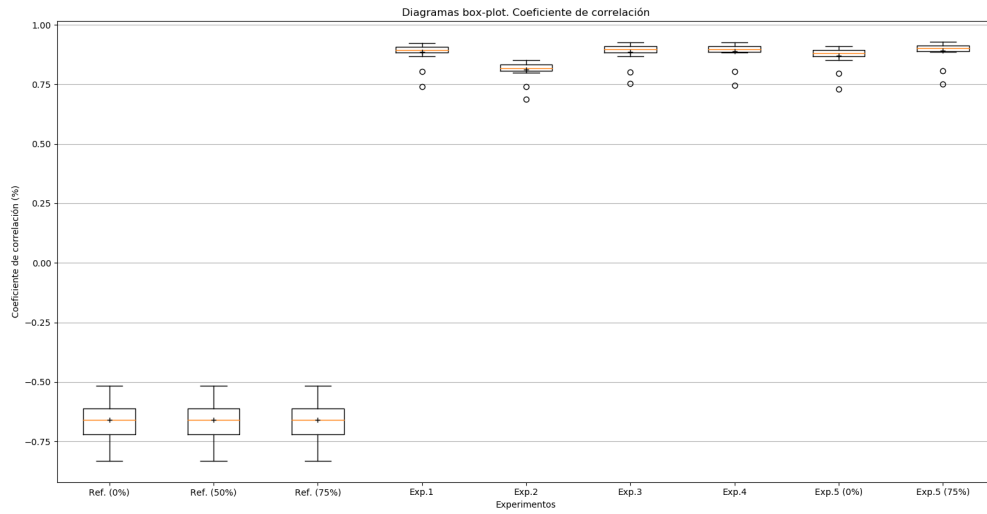


Figura 4.7: Diagramas *box-plot* de los coeficientes de correlación de los casos de *test* en los diferentes experimentos de la fase de SR.

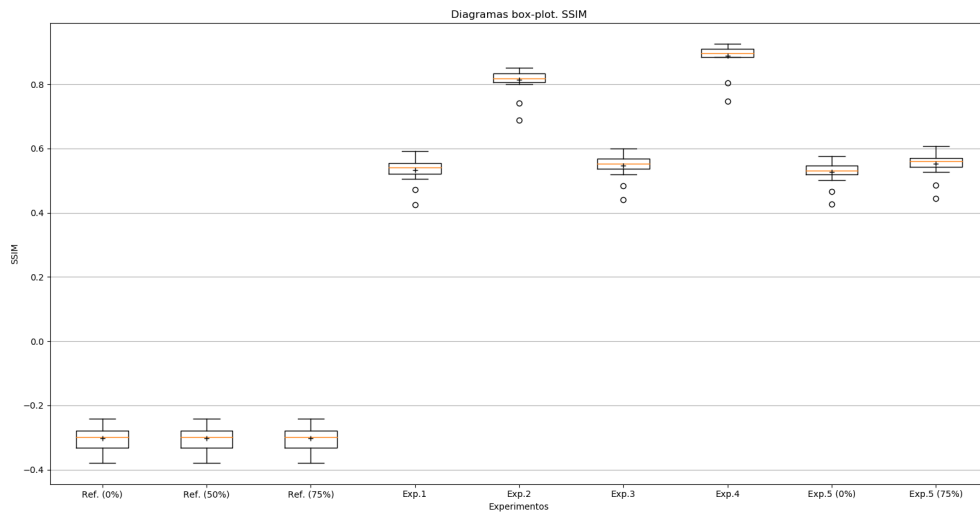


Figura 4.8: Diagramas *box-plot* de los índices SSIM de los casos de *test* en los diferentes experimentos de la fase de SR.

En referencia a las modificaciones llevadas a cabo, cabe resaltar los siguientes aspectos:

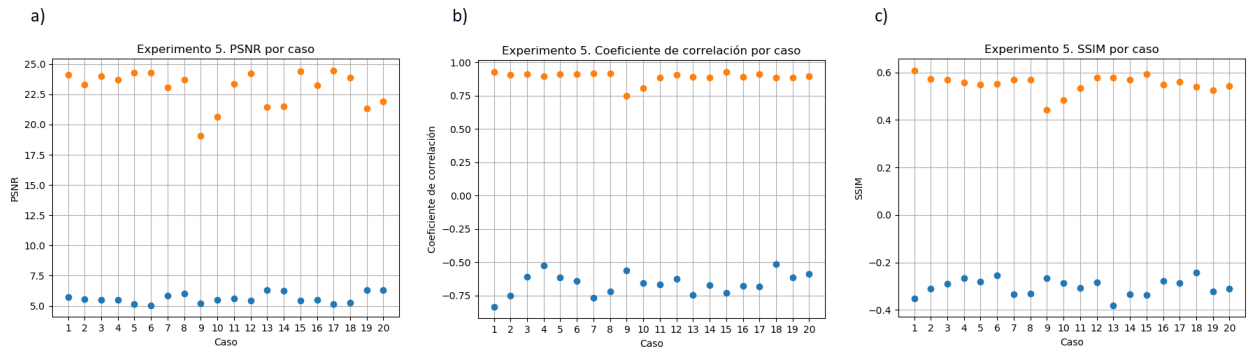
- El uso de una función de coste formada por las funciones funciones MAE y DSSIM (Experimento 3) combina información estructural de la imagen con la distancia absoluta entre los valores de intensidad de los vóxeles. Con ella, todas las métricas han mejorado ligeramente con respecto a las redes entrenadas con las funciones de coste individuales (Experimentos 1 y 2).

- La eliminación de la capa de regularización *Dropout* ha permitido un mejor aprendizaje de la red en este problema, puesto las métricas han mejorado.
- Al igual que en la red T1 SR, empleando la misma red entrenada con *patches* extraídos con un solapamiento al 50%, ante un mayor porcentaje de solapamiento en la reconstrucción en la fase de *test* - al 75% - se produce una mejora en todas las métricas con respecto a la red anterior (Experimento 3). Sin embargo, también en este caso el tiempo de ejecución se ve incrementado.

A partir de los resultados presentados, se concluye que la mejor red es la obtenida en el Experimento 5, cuya mejora global con respecto a su Referencia (al 75% de solapamiento) es de:

- 17,37 dB de ganancia en PSNR.
- 1,55 (155%) de ganancia en coeficiente de correlación.
- 0,854 (85.4%) de ganancia en SSIM.

La evolución de las métricas en cada caso del conjunto de *test* en el Experimento 5 se muestra en las gráficas de la Figura 4.9.



**Figura 4.9:** Gráficas de las métricas calculadas para cada uno de los casos de *test*. a) PSNR. b) Coeficiente de correlación. c) SSIM.

Para comprobar si las métricas siguen una distribución normal, se ha aplicado la prueba de normalidad de Shapiro-Wilk sobre las 20 muestras de métricas de la Referencia y el Experimento 5. Los resultados de la prueba se muestran en la Tabla 4.6.

		Prueba de normalidad Shapiro-Wilk		
		PSNR	Coeficiente de correlación	SSIM
Referencia	Estadístico <i>W</i>	0,900	0,984	0,974
	p-valor	0,042	0,974	0,841
Experimento 5	Estadístico <i>W</i>	0,850	0,664	0,868
	p-valor	0,005	$1,45 \times 10^{-5}$	0,011

**Tabla 4.6:** Resultados de la prueba de normalidad Shapiro-Wilk aplicado a las muestras de las métricas de la Referencia y el Experimento 5. El número de muestras de cada grupo es de  $n = 20$ .

A partir de los p-valores obtenido en el test, se rechaza con una probabilidad de error inferior al 1 % (nivel de significación  $\alpha = 0,01$ ) la hipótesis alternativa  $H_1$  para la muestra de Referencia. Sin embargo, para la muestra del Experimento 5, se debe aceptar siguiendo el mismo criterio. En consecuencia podemos afirmar que las métricas de esta segunda muestra no siguen una distribución normal.

Al no quedar garantizada la normalidad, comprobamos si existen diferencias significativas ambas muestras aplicando la prueba de los rangos con signo de Wilcoxon. Esta prueba estadística no paramétrica asume como hipótesis nula  $H_0$  que no existe una diferencia significativa entre las medianas de ambas muestras y, como hipótesis alternativa  $H_1$ , que dicha diferencia sí que existe y es significativa. El número de muestras de cada grupo es de  $n = 20$  (por norma general, es deseable para esta prueba que  $n \geq 20$ ). Los resultados del test se muestran en la Tabla 4.7.

<b>Prueba de rangos y signos de Wilcoxon</b>			
	<b>PSNR</b>	<b>Coefficiente de correlación</b>	<b>SSIM</b>
<b>Estadístico <math>W^+</math></b>	0,0	0,0	0,0
<b>p-valor</b>	$8,857 \times 10^{-5}$	$8,857 \times 10^{-5}$	$8,857 \times 10^{-5}$

**Tabla 4.7: Resultados de la prueba de rangos y signos de Wilcoxon entre las diferentes métricas del Experimento 5 y la Referencia.**

Los p-valores obtenidos permiten aceptar con una probabilidad de error inferior al 1 % (nivel de significación  $\alpha = 0.01$ ) la hipótesis alternativa  $H_1$  para todos los casos. En otras palabras, existe una diferencia significativa entre las métricas obtenidas en el Experimento 5 a partir de los casos generados por la red y las métricas de los casos de referencia.

Finalmente, las imágenes IRM SR potenciadas en T2 obtenidas en el Experimento 5 se muestran en la Figura 4.10. Visualmente, se observa que las imágenes IRM SR generadas por la red se asemejan notablemente a las imágenes IRM HR en estructura. Sin embargo, a diferencia de la red T1 SR, las imágenes se encuentran excesivamente suavizadas y existen determinados contornos y texturas que la red no ha sido capaz de generar. Esto es debido a la mayor complejidad del problema de síntesis de imágenes IRM T2, a causa de la diferencia que existe entre las modalidades T1 y T2 en los valores de intensidad de la imagen y en la potenciación de determinadas estructuras y tejidos.

Finalmente, se ha comparado el método de síntesis de imágenes IRM T2 SR desarrollado con las referencias del estado del arte calculando la ganancia o diferencia entre el método propuesto y la referencia. La comparación se muestra en la Tabla 4.8.

Según los resultados presentados, el método de síntesis propuesto se aproximaría al estado del arte en PSNR pero no en el caso de la métrica SSIM. Debe tenerse en cuenta que la métrica SSIM es sensible a las variaciones de la media, y que, en nuestro caso, el proceso inverso de normalización para evaluar las métricas se ha realizado tomando valores de media  $\mu_I$  y desviación  $\sigma_I$  de los volúmenes IRM T2 HR originales <sup>4</sup>. Por otro lado, la eliminación del ruido de los volúmenes IRM T2 SR generadas por la red provoca una pérdida considerable de similitud estructural entre las imágenes originales y sintéticas. Finalmente, es importante resaltar que el

---

<sup>4</sup>Recuérdese que la red de síntesis T2 SR devuelve los residuos que, sumados a los datos T1 SR, generan las imágenes T2 SR normalizadas (media  $\mu_I = 0$  y  $\sigma_I = 1$ ). Por ello, no se dispone de la información de la media y desviación de los volúmenes T2 SR.

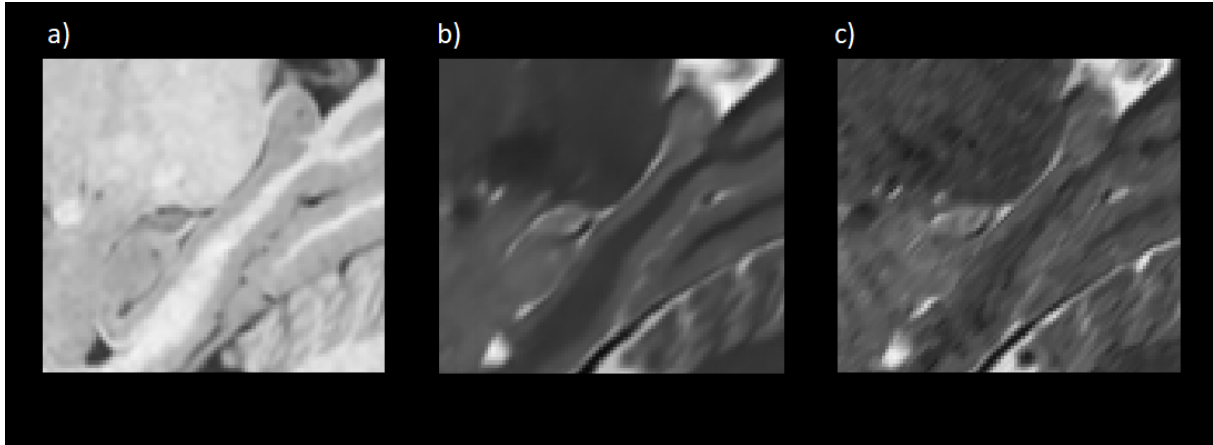


Figura 4.10: Cortes sagitales de tres *crops* IRM potenciados en T1 y T2 pertenecientes a un sujeto del conjunto de *test*. a) *Crop* SR potenciado en T1 generado por la mejor red T1 SR. b) *Crop* SR potenciado en T2 generado por la mejor red de síntesis T2 (Experimento 5). c) *Crop* HR potenciado en T2.

Método	Métricas	
	PSNR	SSIM
Roy, Carass y Prince 2013 (MIMECS)	23,88	0,878
Yang y col. 2010 (SC)	24,88	0,878
Vemulapalli, Van Nguyen y Kevin Zhou 2015 (Ve-US)	26,70	0,883
Vemulapalli, Van Nguyen y Kevin Zhou 2015 (Ve-S)	27,76	0,887
Huang, Shao y Frangi 2017	27,96	0,899
<b>Propuesto</b>	<b>23,002</b>	<b>0,552</b>

Tabla 4.8: Comparación del método propuesto de síntesis de imágenes IRM T2 SR con algunos de los métodos del estado del arte.

conjunto de datos de *test* ha sido distinto al del resto de referencias, de modo que, para realizar una adecuada comparativa, debería evaluarse el método propuesto sobre el mismo conjunto de datos empleado por el resto de métodos: el conjunto de imágenes de *National Alliance for Medical Image Computing* (NAMICS) <sup>5</sup>. Todos estos factores explicarían la baja SSIM del método propuesto en comparación con el resto de métodos evaluados.

## 4.2 Fase de clasificación

Los resultados de las métricas de la fase de clasificación se han presentado en tres formatos:

- Tabla de las métricas calculadas sobre el conjunto de *test*: precisión, sensibilidad, especificidad y ROC AUC.
- Matrices de confusión.
- Curvas ROC.

También se muestran diferentes cortes de imágenes IRM de la base de datos ADNI clasificadas por la red con sus respectivas probabilidades.

<sup>5</sup><http://hdl.handle.net/1926/1687>

En la Tabla 4.9 se resumen los valores de las métricas de cada experimento evaluadas sobre el conjunto de *test*: 38 sujetos de la clase AD y 38 sujetos de la clase CN, cada uno de ellos con dos *crops* IRM LR potenciados en T1. A partir de las predicciones resultantes de los casos, se han generado las etiquetas redondeando las probabilidades de pertenecer a la clase AD  $p_{AD}$  predichas por la red aplicando un umbral de 0,5, de modo que:

- Si  $p_{AD} > 0,5$ , entonces el sujeto se considera AD. Etiqueta: 1.
- Si  $p_{AD} < 0,5$ , entonces el sujeto se considera CN. Etiqueta: 0.

Experimento	Características	Precisión	Sensibilidad	Especificidad	ROC AUC
Experimento 1	<i>Dropout</i> niveles 4 y 5	0,757	0,892	0,622	0,870
Experimento 2	<i>Label smoothing</i>	0,757	0,838	0,676	0,870
Experimento 3	<i>Mixup</i> $\alpha = 0,1$	0,824	0,784	0,864	0,895
	<i>Mixup</i> $\alpha = 0,2$	0,838	0,892	0,784	0,893
	<i>Mixup</i> $\alpha = 1,0$	0,730	0,811	0,649	0,843
Experimento 4	<i>Strides</i> ( $2^3$ )	0,811	0,757	<b>0,865</b>	0,823
Experimento 5	<i>Dropout</i> nivel 2	<b>0,878</b>	<b>0,946</b>	0,811	<b>0,958</b>

Tabla 4.9: Métricas resultantes de los experimentos de las redes de clasificación.

Los resultados de las métricas en cada experimento quedan representados gráficamente en el diagrama de barras de la Figura 4.11.

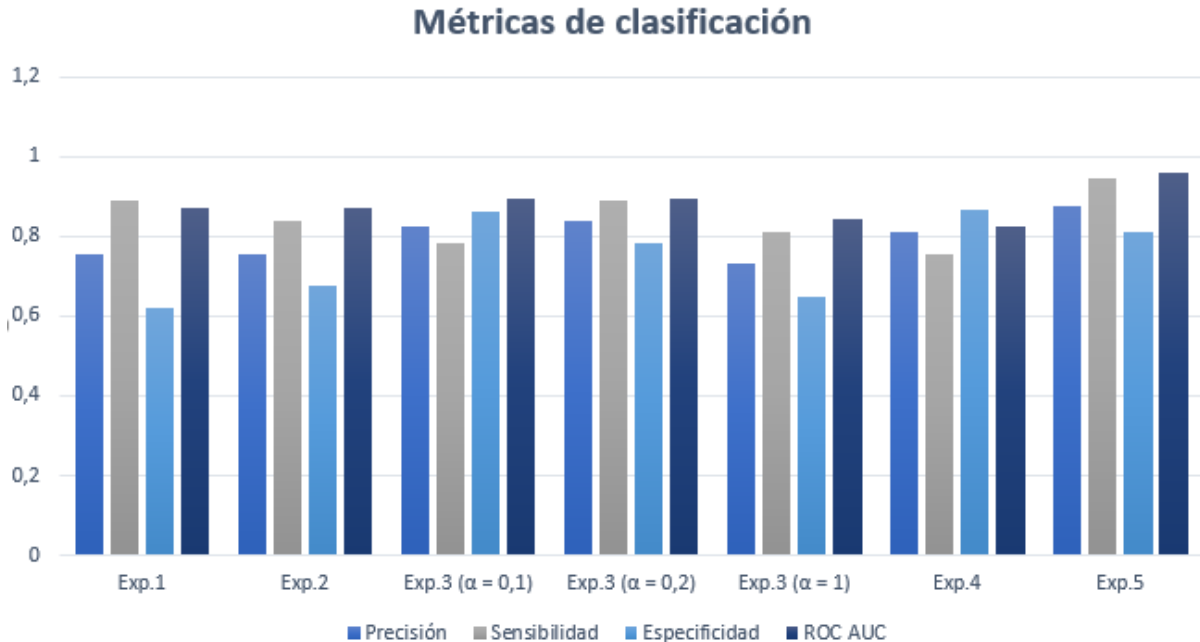


Figura 4.11: Diagrama de barras de las métricas obtenidas en cada experimento de la fase de clasificación.

Sobre los experimentos realizados y sus modificaciones, cabe destacar los siguientes aspectos:

- Las capas de regularización de *Dropout* han permitido estabilizar los entrenamientos.

- El uso de técnica *mixup* de *data augmentation* (Experimento 3) supone una clara mejora en la predicción de la red con respecto al entrenamiento con los datos originales (Experimento 1) y a otras técnicas como *label smoothing* (Experimento 2). El parámetro  $\alpha = 0,2$  de la distribución beta ha resultado ser el óptimo para la selección de los pesos del *mixup*.
- La reducción de las dimensiones mediante *strides* en lugar de *MaxPooling* no altera los resultados notablemente.
- La colocación de las capas *Dropout* en los experimentos realizados para que el entrenamiento sea óptimo es en el segundo nivel.

En base a los resultados obtenidos, se concluye que la mejor red para la resolución del problema de clasificación es la red entrenada en el Experimento 5, con las siguientes métricas:

- *Accuracy*: 87,8 %
- Sensibilidad: 94,6 %
- Especificidad: 81,1 %
- ROC AUC: 95,8 %

Los resultados de su clasificación sobre los 74 casos de *test* se muestran en la matriz de confusión de la Figura 4.12 y en la curva ROC de la Figura 4.13.

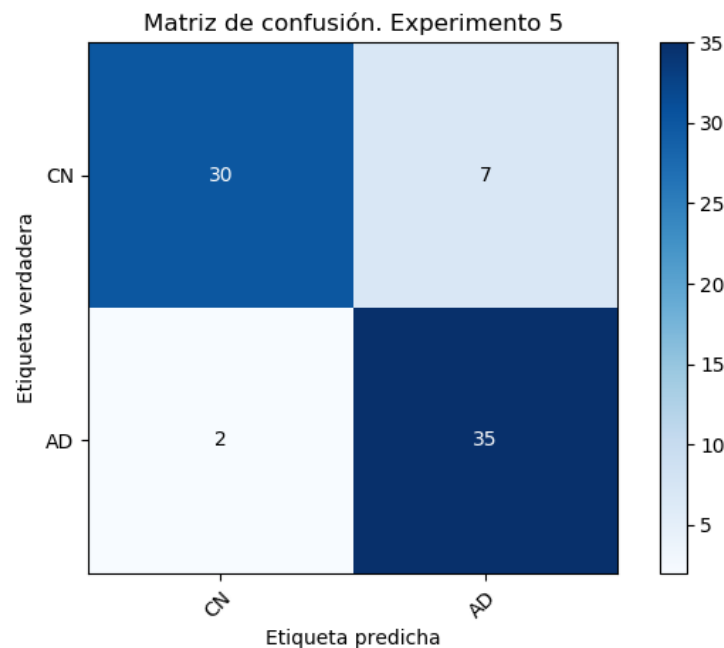
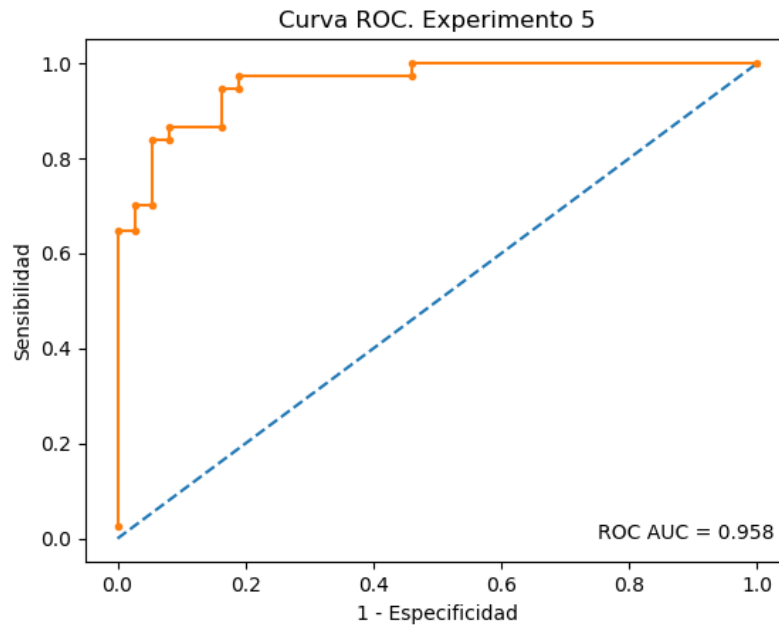


Figura 4.12: Matriz de confusión de los casos de *test* del Experimento 5 en la fase de clasificación.

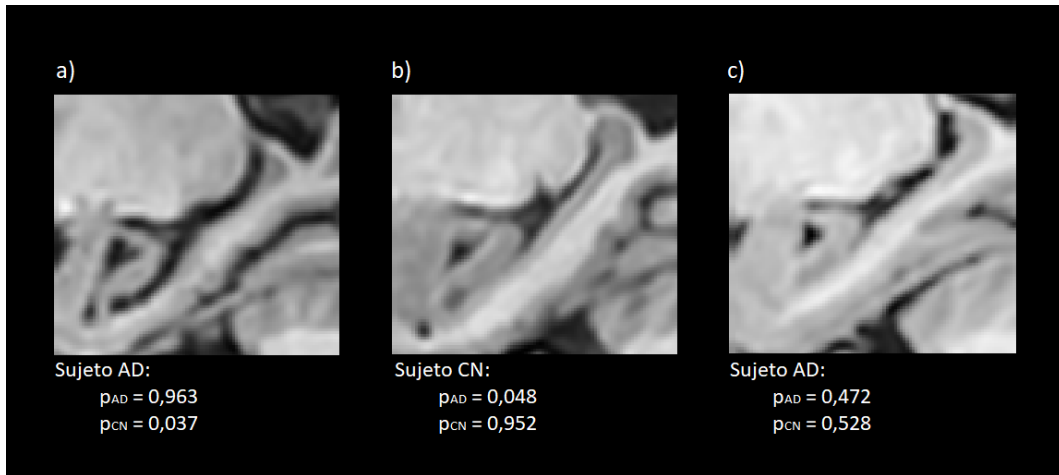


**Figura 4.13: Curva ROC de los casos de *test* del Experimento 5 en la fase de clasificación.** La curva muestra la Sensibilidad frente a 1 - Especificidad de la red clasificadora.

En la matriz se observa como la sensibilidad de la red ante los casos AD es superior a la especificidad ante los casos CN, es decir, que la detección de casos AD es mejor que la detección de casos CN. Cabe destacar que, aunque en otros experimentos, como el Experimento 3 ( $\alpha = 0,1$ ) o el 4, la especificidad ha sido mayor, es preferible que la clasificación tenga una mayor sensibilidad como sistema de ayuda diagnóstico, es decir, una mayor detección de verdaderos positivos. Esta decisión se complementa con el área bajo la curva ROC o ROC AUC, el cual indica en este caso que existe un 95,8% de probabilidad de que la clasificación de la red a un paciente del grupo AD (enfermo) sea más correcta la de un sujeto del grupo CN (sano).

Finalmente, la Figura 4.14 se pueden observar tres ejemplos de imágenes IRM ADNI clasificadas por la red con sus respectivas probabilidades de pertenencia a su clase.

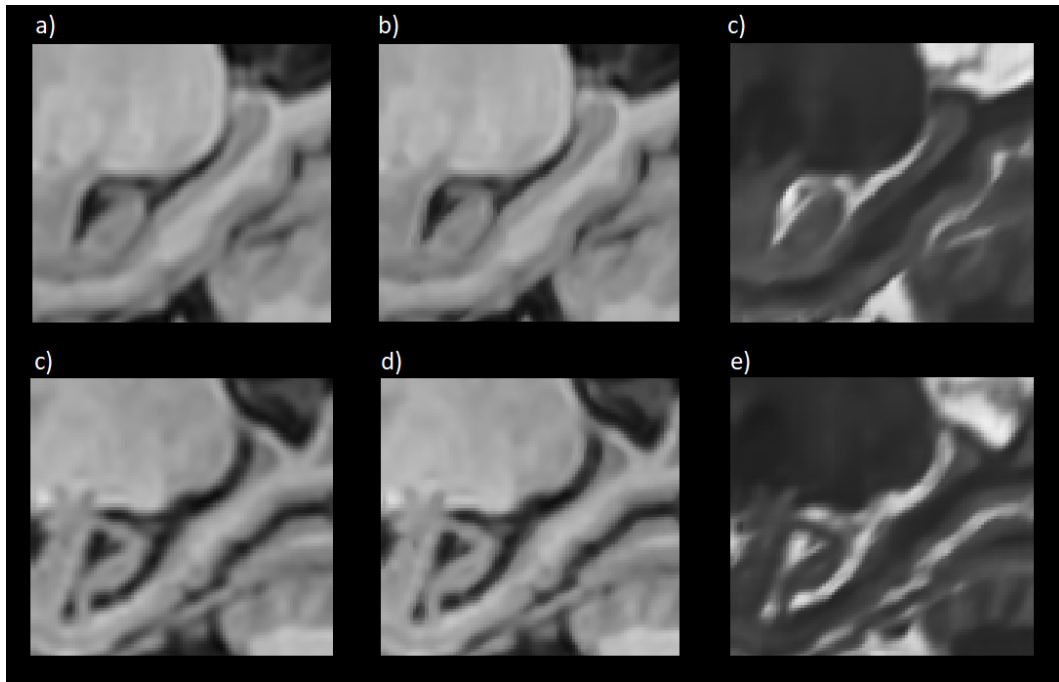




**Figura 4.14:** Cortes sagitales de tres *crops* IRM potenciados en T1 pertenecientes a tres sujetos del conjunto de *test* ADNI. a) *Crop* AD clasificado como AD. b) *Crop* CN clasificado como CN. c) *Crop* AD clasificado como CN.

Analizando los cortes y las probabilidades asociadas, se observa que los dos primeros *crops* (Figura 4.14a y 4.14b) han sido clasificados correctamente con una la probabilidad de pertenencia a su clase es muy elevada. Sin embargo, el tercer *crop* es clasificado por la red como AD con una probabilidad mucho más baja que las anteriores. Al analizar las imágenes, se observa una clara atrofia de la estructura hipocampal en el corte con EA (Figura 4.14b) con respecto al corte normal (Figura 4.14a), mientras que el tercero (Figura 4.14c) se encontraría en un estado intermedio. Deducimos a raíz de los resultados que la red ha sido capaz de aprender y diferenciar entre estructuras anatómicas normales y estructuras con atrofia severa en la región del lóbulo temporal, pero no ha sido capaz de reconocer grados de atrofia leves.

Una vez seleccionado el mejor experimento, el Experimento 5, se han generado las imágenes ADNI SR potenciadas en T1 y T2 a partir de las imágenes originales LR. En la Figura 4.15 se muestran una serie de cortes para dos sujetos AD y CN. Comparando los cortes potenciados en T1 LR y SR, se aprecia que la mejora de la resolución, aunque no es tan evidente como en los resultados de las redes de síntesis en la fase de SR. Esto puede atribuirse a que la procedencia los casos de la base de datos ADNI, su preprocesado, los factores de *upsampling*... son muy variados, mientras que las redes de SR solo han sido entrenadas para casos de un mismo estudio, bajo las mismas condiciones.



**Figura 4.15:** Cortes sagitales de tres *crops* IRM potenciados en T1 y T2 pertenecientes a dos sujetos CN y AD del conjunto de *test* ADNI. a) *Crop* potenciado en T1 LR original del sujeto CN. b) *Crop* potenciado en T1 SR generado por la mejor red T1 SR (Experimento 5) del sujeto CN. c) *Crop* potenciado en T2 SR generado por la mejor red de síntesis T2 SR (Experimento 5) del sujeto CN. d) *Crop* potenciado en T1 LR original del sujeto AD. e) *Crop* potenciado en T1 SR generado por la mejor red T1 SR (Experimento 5) del sujeto AD. f) *Crop* potenciado en T2 SR generado por la mejor red de síntesis T2 SR (Experimento 5) del sujeto AD.

El Experimento 5 se ha evaluado de nuevo con el mismo conjunto de datos de *test* (es decir, con los mismos sujetos), pero empleando los nuevos *crops* SR generados. Como último experimento, se ha entrenado una red con los casos IRM potenciados en T1 y T2, introduciéndolos en 4 canales de entrada, aplicando las mismas condiciones que se han establecido en el Experimento 5. En la Tabla 4.10 se comparan los resultados.

Experimento	Precisión	Sensibilidad	Especificidad	ROC AUC
<i>Test</i> : T1 LR	0,878	0,946	0,811	0,958
<i>Test</i> : T1 SR	<b>0,892</b>	<b>0,946</b>	<b>0,837</b>	<b>0,959</b>
<i>Train/Test</i> : T1 + T2 SR	0,797	0,784	0,811	0,891

**Tabla 4.10:** Métricas resultantes de la evaluación de la red de clasificación del Experimento 5 entrenando con datos T1 LR, T1 SR y T1 + T2 SR.

Al comparar los resultados, observamos que la mejora de la resolución de las imágenes mejora ligeramente la clasificación de algunos casos de la red de los sujetos CN, lo cual se traduce en un aumento de la especificidad. La matriz de confusión y la curva ROC de la clasificación de los casos T1 SR se muestran en las Figuras 4.16 y 4.17, respectivamente.

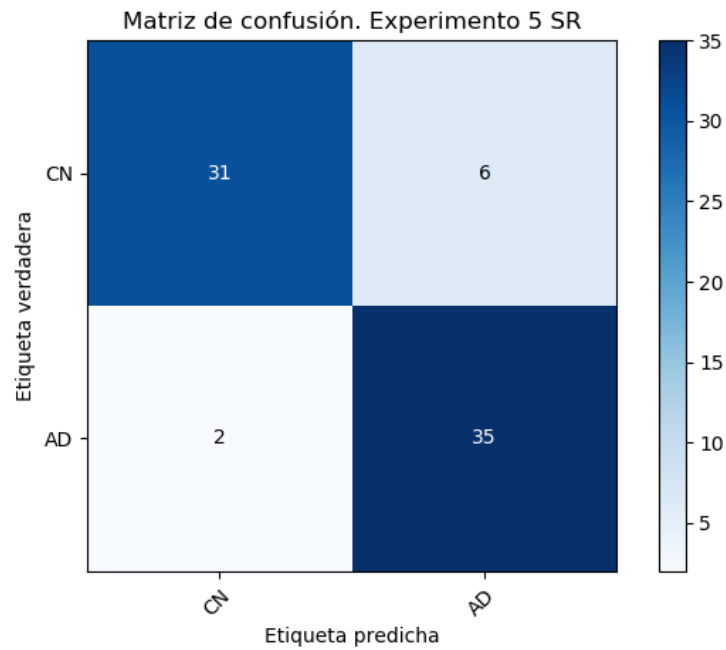


Figura 4.16: Matriz de confusión de los casos de *test* SR del Experimento 5 en la fase de clasificación.

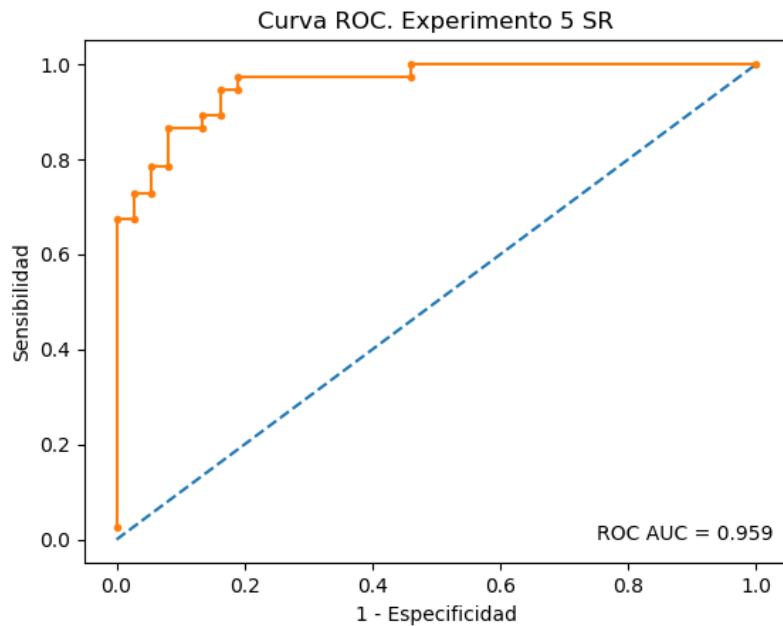


Figura 4.17: Curva ROC de los casos de *test* del Experimento 5 en la fase de clasificación. La curva muestra la Sensibilidad frente a 1 - Especificidad de la red clasificadora.

Sin embargo, la incorporación de los casos IRM potenciados en T2 de la forma propuesta no mejora la clasificación. Esto puede deberse a que las imágenes IRM T2 sintetizadas no presentan la suficiente resolución, o a la necesidad de una arquitectura de red diferente que pueda extraer y procesar la información adecuadamente.

Finalmente, se ha comparado el método de clasificación desarrollado con algunas de las referencias del estado del arte que resuelven el mismo problema. La comparación se muestra en la Tabla 4.11.

Método	Métricas			
	Precisión	Sensibilidad	Método	ROC AUC
Liu y col. 2014 (IRM + PET)	0,878	0,886	0,872	-
Li y col. 2014 (IRM + PET)	0,929	-	-	-
Coupé y col. 2012a (IRM)	0,890	0,840	0,930	-
Ortiz y col. 2016 (IRM + PET)	0,90	0,860	0,940	0,950
<b>Propuesto (IRM)</b>	0,892	0,946	0,837	0,959

**Tabla 4.11: Comparación del método propuesto de clasificación con algunos de los métodos del estado del arte.**

Todos los métodos han sido evaluados sobre imágenes de modalidad MRI de la base de datos ADNI, aunque debe tenerse en cuenta que los conjuntos de datos no son iguales, y que algunos de ellos incluyen la modalidad PET en el problema de la clasificación. A raíz de la comparación se puede concluir que el método de clasificación propuesto presenta resultados cercanos al estado del arte.

# Conclusiones y líneas futuras

*En este capítulo final se expondrán las conclusiones del proyecto, dejando constancia de los logros alcanzados y de las limitaciones encontradas durante su desarrollo. Asimismo, se plantearán las posibles líneas futuras que quedan abiertas tras la conclusión del trabajo.*

En el presente trabajo se ha conseguido diseñar y desarrollar un método automático y robusto de SR y clasificación de basado en redes neuronales artificiales, tal y como se ha propuesto en el objetivo general del proyecto. Durante el desarrollo del proyecto, se han cumplido todos los objetivos específicos, destacando entre ellos la adquisición de nuevos conocimientos en el campo del aprendizaje automático profundo y a la programación de ANN.

Los resultados obtenidos y su análisis muestran en las tres fases del proyecto que las soluciones propuestas se aproximan a los resultados de otros métodos del estado del arte, y constatan que la combinación de las técnicas de SR con los métodos de clasificación basados en extracción de características mejoran el diagnóstico, tal y como se esperaba demostrar. Por otro lado, el uso de las ANN ha mostrado la capacidad de las mismas para automatizar el procesado y la clasificación de imágenes a gran velocidad. Sin embargo, a la hora de desarrollar un proyecto basado en ANN, debe tenerse en cuenta que gran parte del trabajo se basa en la experimentación y en las modificaciones del algoritmo de forma empírica, ya que el proceso de aprendizaje de las redes es estocástico. Cabe mencionar además que una de las limitaciones más importantes ha sido el tiempo de entrenamiento de las ANN en los experimentos (de horas o días), por lo que la metodología y la organización han sido de vital importancia en el desarrollo del proyecto. También el tratamiento de los datos ha resultado ser fundamental a la hora de preparar la red para su entrenamiento y ha ocupado gran parte del tiempo de trabajo.

A raíz del trabajo desarrollado y de los resultados obtenidos, se proponen a continuación una serie de líneas futuras:

- Desarrollo de sistemas *software* de ayuda al diagnóstico intuitivos y sencillos para los profesionales de la salud.
- Diseño de nuevas redes de mejora de la SR y síntesis de nuevas modalidades de la imagen IRM.

- Entrenamiento de las redes de clasificación con otras modalidades de imagen (T2, PET...) con el fin de mejorar el problema de clasificación de sujetos.
- Evaluación de las redes con más casos de *test*, aplicando técnicas de *cross-validation* (o *K-fold*), con el fin de ofrecer resultados más robustos y representativos.
- Aplicación de las ANN de SR y clasificación a otras técnicas de imagen médica y a la detección de otras neuropatologías (glioblastoma, enfermedad de Parkinson, etc.).

Con este proyecto, esperamos haber contribuido a la investigación de nuevas metodologías automáticas, robustas, objetivas, reproducibles y escalables en el campo de la imagen médica y del diagnóstico clínico. Y, en su aplicación práctica, esperamos aportar un estudio más en el diagnóstico de la EA que, en un futuro, pueda facilitar a los profesionales de la salud la labor llevada a cabo en su día a día.

Parte II

# Presupuesto





# Presupuesto

## Introducción y objetivo

En este documento se presenta una estimación económica del coste del proyecto. Los costes del proyecto se han desglosado en tres cuadros de costes: costes de mano de obra, costes de *hardware* y costes de *software*. Al final del documento se presenta el coste global del proyecto en forma de presupuesto total de ejecución material y de ejecución por contrata.

## Presupuesto desglosado

### *Costes de mano de obra*

En la Tabla 5.1 se resumen los costes de mano de obra que ha acarreado el proyecto. La realización del trabajo ha requerido de un alumno del Grado en Ingeniería Biomédica (GIB), que ha desarrollado el proyecto, y de un tutor doctor en Ingeniería Informática especialista en el campo de la imagen médica, que ha supervisado la labor del alumno. El sueldo base de un ingeniero informático *senior* es de aproximadamente 40 000 €/año, mientras que el de un ingeniero biomédico *junior* es de aproximadamente 20 000 €/año, según las cifras del informe The Adecco Group 2018. Si se considera una jornada laboral de 8 h y se descuentan fines de semanas, 15 días festivos, y 30 días de periodo vacacional, se obtienen un total de 224 jornadas o 1 792 h laborales al año. Se deduce que el coste por hora es de 11,16 €/h para el ingeniero biomédico *junior* y de 22,32 €/h para el ingeniero informático *senior*.

Denominación	Precio unitario (€/h)	Cantidad (h)	Coste total (€)
Ingeniero biomédico	11,16	480	5 356,80
Ingeniero informático	16,74	120	2 678,40
<b>Total</b>			<b>8 035,20</b>

Tabla 5.1: Costes de mano de obra.

El coste total de la mano de obra en este proyecto ha sido de **ocho mil treinta y cinco euros con veinte céntimos**.

### *Costes de hardware*

El equipo *hardware* empleado en el proyecto ha sido un ordenador con las características descritas en la 3.1 del Capítulo 2: Materiales y Métodos del documento de la Memoria. A este equipo hay que añadirle el coste de la tarjeta gráfica. A la hora de calcular los costes totales, se ha considerado el tiempo de vida de trabajo del equipo y el factor de amortización. Teniendo en cuenta que los equipos se han empleado durante un periodo de 2 meses de desarrollo del proyecto, y que el periodo de vida útil de los componentes es de unos 5 años (60 meses), se ha obtenido el factor de amortización y los costes de cada componente en su periodo de uso. Los costes *hardware* se resumen en la Tabla 5.2.

Denominación	Precio (€/ud.)	Cantidad (uds.)	Factor de amortización	Coste total (€)
Ordenador	1000,00	1	2/60	33,33
Tarjeta gráfica	780,00	1	2/60	26,00
<b>Total</b>				<b>59,33</b>

Tabla 5.2: Costes de *hardware*.

El coste total del *hardware* en este proyecto ha sido de **cincuenta y nueve euros con treinta y tres céntimos**.

### *Costes de software*

Los costes de los programas *software* empleados en el proyecto se han calculado en función de sus correspondientes licencias y su periodo de uso, aplicando el factor de amortización de igual modo que en los equipos *hardware*. Para calcular dicho factor, en el caso del sistema operativo Windows 10 Enterprise se ha considerado como periodo de amortización un máximo de 4 años (48 meses) hasta la obsolescencia del producto. En referencia al *software* MATLAB 2018a, se trata de una licencia anual (12 meses). En el resto de programas empleados, la licencia es gratuita, por lo que no han supuesto ningún coste económico. Los costes *software* se resumen en la Tabla 5.3.

Denominación	Precio (€)	Cantidad (uds.)	Factor de amortización	Coste total (€)
Microsoft Windows 10 Enterprise	259,00	1	2/48	10,79
MATLAB 2018a	250,00	1	2/12	41,67
ITK-SNAP 3.6	0,00	1	-	0,00
Anaconda 3.7	0,00	1	-	0,00
Sublime Text 3.0	0,00	1	-	0,00
ANTs	0,00	1	-	0,00
OverLeaf <sup>1</sup>	0,00	1	-	0,00
<b>Total</b>				<b>52,46</b>

Tabla 5.3: Costes de *software*.

<sup>1</sup>Este editor online de LaTeX se ha empleado para la redacción del proyecto.

El coste total del *software* en este proyecto ha sido de **cincuenta y dos euros con cuarenta y seis céntimos**.

## Presupuesto total

### *Presupuesto de ejecución material*

El presupuesto de ejecución material se ha calculado sumando los costes parciales de cada unidad de obra anteriormente expuestos. Los resultados se muestran en la Tabla 5.4.

Unidad de obra	Coste total (€)
Mano de obra	8 035,20
<i>Hardware</i>	59,33
<i>Software</i>	52,46
<b>Total</b>	<b>8 146,99</b>

Tabla 5.4: Presupuesto de ejecución material.

El presupuesto material del proyecto antes de impuestos asciende a un coste total de **ocho mil ciento cuarenta y seis euros con noventa y nueve céntimos**.

### *Presupuesto de ejecución por contrata*

La obtención del presupuesto de ejecución por contrata se lleva a cabo incluyendo los gastos generales, el beneficio industrial y los impuestos en el presupuesto de ejecución material. Los gastos generales o indirectos equivalen a un 13 % del presupuesto de ejecución material<sup>2</sup>, y el beneficio industrial se estima que equivale al 6 % del presupuesto de ejecución material en este tipo de proyectos. Finalmente, se aplica el impuesto IVA que, para este proyecto, equivale a un 21 % del presupuesto total antes de impuestos. Los resultados se muestran en la Tabla 5.5.

Denominación	Coste total (€)
<b>Presupuesto de ejecución material</b>	8 146,99
Gastos generales (13 %)	1 059,11
Beneficio industrial (6 %)	488,82
<b>Total antes de impuestos</b>	<b>9 694,92</b>
Impuesto IVA (21 %)	2 035,93
<b>Presupuesto de ejecución por contrata</b>	<b>11 730,85</b>

Tabla 5.5: Presupuesto de ejecución por contrata.

El presupuesto material del proyecto antes de impuestos asciende a un coste total de **once mil setecientos treinta euros con ochenta y cinco céntimos**.

---

<sup>2</sup>Este porcentaje se aplica conforme al Reglamento General de la Ley de Contratos de las Administraciones Públicas (Real Decreto 1098/2001).



# Bibliografía

- Acosta, D. y col. (2012). *La enfermedad de Alzheimer, diagnóstico y tratamiento: una perspectiva latinoamericana*. 1ª Edición. Editorial Médica Panamericana (vid. pág. 4).
- Alzheimer's Disease International (ADI) (2015). *Informe mundial sobre el Alzheimer 2015. Las consecuencias de la demencia: análisis de prevalencia, incidencia, coste y tendencias*. <https://www.alz.co.uk/research/worldalzheimerrreport2015-summary-spanish.pdf>. [Consulta: 6 de Mayo de 2019] (vid. págs. 4, 5).
- (2018). *Informe mundial sobre el Alzheimer 2018. La investigación de vanguardia sobre la demencia: nuevas fronteras*. <https://www.alz.co.uk/research/worldalzheimerrreport2018-spanish.pdf>. [Consulta: 6 de Mayo de 2019] (vid. pág. 4).
- Apostolova, Liana G y col. (2006). «3D comparison of hippocampal atrophy in amnesic mild cognitive impairment and Alzheimer's disease». En: *Brain* 129.11, págs. 2867-2873 (vid. pág. 8).
- Asociación Educar*. <https://asociacioneducar.com/ilustracion-giro-dentado>. Consulta: 7 de Mayo de 2019 (vid. pág. 8).
- Barranco-Quintana, J.L. y col. (2005). «Factores de riesgo en la enfermedad de Alzheimer». En: *Revista de Neurología* 40, págs. 613-618 (vid. pág. 4).
- Ben-Ahmed, Olfa y col. (2017). «Multi-view visual saliency-based MRI classification for alzheimer's disease diagnosis». En: *2017 Seventh International Conference on Image Processing Theory, Tools and Applications (IPTA)*. IEEE, págs. 1-6 (vid. pág. 33).
- Chen, Yuhua y col. (2018). «Efficient and accurate MRI super-resolution using a generative adversarial network and 3D multi-Level densely connected network». En: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, págs. 91-99 (vid. págs. 32, 59, 60).

- Confederación Española de Alzheimer (CEAFA) (2014). *Estado del arte de la enfermedad de Alzheimer en España*. [https://www.ceafa.es/files/2014/06/estado\\_del\\_arte-2.pdf](https://www.ceafa.es/files/2014/06/estado_del_arte-2.pdf). [Consulta: 6 de Mayo de 2019] (vid. págs. 4, 9, 11).
- Coupé, Pierrick y col. (2012a). «Scoring by nonlocal image patch estimator for early detection of Alzheimer’s disease». En: *NeuroImage: clinical* 1.1, págs. 141-152 (vid. págs. 33, 72).
- Coupé, Pierrick y col. (2012b). «Simultaneous segmentation and grading of anatomical structures for patient’s classification: application to Alzheimer’s disease». En: *NeuroImage* 59.4, págs. 3736-3747 (vid. pág. 33).
- Células gliales*. <https://celulasgliales.com/sistema-nervioso-central-snc/>. Consulta: 7 de Mayo de 2019 (vid. pág. 7).
- Data Wow*. <https://blog.datawow.io/cnn-models-ef356bc11032>. Consulta: 18 de Mayo de 2019 (vid. pág. 30).
- Daza, Julian Camilo y Andrea Rueda (2016). «Classification of alzheimer’s disease in mri using visual saliency information». En: *2016 IEEE 11th Colombian Computing Conference (CCC)*. IEEE, págs. 1-7 (vid. pág. 33).
- Dong, Chao y col. (2015). «Image super-resolution using deep convolutional networks». En: *IEEE transactions on pattern analysis and machine intelligence* 38.2, págs. 295-307 (vid. pág. 32).
- Dpto. de Ciencias de la Computación e Inteligencia Artificial. Universidad de Sevilla*. <http://www.cs.us.es/~fsancho/?e=72>. Consulta: 14 de Mayo de 2019 (vid. pág. 20).
- El Blog Verde*. <https://elblogverde.com/impulso-nervioso/neurons-4/>. Consulta: 14 de Mayo de 2019 (vid. pág. 20).
- Eskildsen, Simon F y col. (2013). «Prediction of Alzheimer’s disease in subjects with mild cognitive impairment from the ADNI cohort using patterns of cortical thinning». En: *Neuroimage* 65, págs. 511-521 (vid. pág. 33).
- Eskildsen, Simon F y col. (2017). «Increased cortical capillary transit time heterogeneity in Alzheimer’s disease: a DSC-MRI perfusion study». En: *Neurobiology of aging* 50, págs. 107-118 (vid. pág. 33).
- free Code Camp (I)*. <https://medium.freecodecamp.org/an-intuitive-guide-to-convolutional-neural-networks-260c2de0a050?gi=7123bc6d5df8>. Consulta: 17 de Mayo de 2019 (vid. pág. 25).
- free Code Camp (II)*. <https://medium.freecodecamp.org/how-to-build-your-first-neural-network-to-predict-house-prices-with-keras-f8db83049159>. Consulta: 18 de Mayo de 2019 (vid. pág. 29).

- Future Website of Anatomia*. [http://www.anatomia.tripod.com/neuroanatomia\\_imagenes.htm](http://www.anatomia.tripod.com/neuroanatomia_imagenes.htm). Consulta: 7 de Mayo de 2019 (vid. pág. 6).
- Gili, Jaume (1993). «Introducción biofísica a la resonancia magnética». En: *Centre Diagnòstic Pedralbes* (vid. págs. 13-15).
- GitHub (I)*. [https://github.com/arthurmeyer/Saliency\\_Detection\\_Convolutional\\_Autoencoder](https://github.com/arthurmeyer/Saliency_Detection_Convolutional_Autoencoder). Consulta: 17 de Mayo de 2019 (vid. pág. 25).
- GitHub (II)*. [https://github.com/arthurmeyer/Saliency\\_Detection\\_Convolutional\\_Autoencoder](https://github.com/arthurmeyer/Saliency_Detection_Convolutional_Autoencoder). Consulta: 17 de Mayo de 2019 (vid. pág. 25).
- Goodfellow, Ian y col. (2014). «Generative adversarial nets». En: *Advances in neural information processing systems*, págs. 2672-2680 (vid. págs. 24, 32).
- Huang, Yawen, Ling Shao y Alejandro F Frangi (2017). «Cross-modality image synthesis via weakly coupled and geometry co-regularized joint dictionary learning». En: *IEEE transactions on medical imaging* 37.3, págs. 815-827 (vid. págs. 32, 65).
- Kellett, Katherine AB y Nigel M Hooper (2009). «Prion protein and Alzheimer disease». En: *Prion* 3.4, págs. 190-194 (vid. pág. 3).
- Kingma, Diederik P y Jimmy Ba (2014). «Adam: A method for stochastic optimization». En: *arXiv preprint arXiv:1412.6980* (vid. pág. 49).
- Krizhevsky, Alex, Ilya Sutskever y Geoffrey E Hinton (2012). «Imagenet classification with deep convolutional neural networks». En: *Advances in neural information processing systems*, págs. 1097-1105 (vid. pág. 32).
- Kulaga-Yoskovitz, Jessie y col. (2015). «Multi-contrast submillimetric 3 Tesla hippocampal sub-field segmentation protocol and dataset». En: *Scientific data* 2, pág. 150059 (vid. págs. 37, 40).
- LeCun, Yann, Yoshua Bengio y Geoffrey Hinton (2015). «Deep learning». En: *nature* 521.7553, pág. 436 (vid. pág. 32).
- Ledig, Christian y col. (2017). «Photo-realistic single image super-resolution using a generative adversarial network». En: *Proceedings of the IEEE conference on computer vision and pattern recognition*, págs. 4681-4690 (vid. pág. 32).
- Li, Rongjian y col. (2014). «Deep learning based imaging data completion for improved brain disease diagnosis». En: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, págs. 305-312 (vid. págs. 33, 72).
- Lifeder.com*. <https://www.lifeder.com/hipocampo/>. Consulta: 7 de Mayo de 2019 (vid. pág. 8).

- Liu, Siqi y col. (2014). «Early diagnosis of Alzheimer's disease with deep learning». En: *2014 IEEE 11th international symposium on biomedical imaging (ISBI)*. IEEE, págs. 1015-1018 (vid. págs. 33, 72).
- López-Pousa, Secundino y col. (2004). «Análisis de los costes de la enfermedad de Alzheimer en función del deterioro cognitivo y funcional». En: *Medicina clínica* 122.20, págs. 767-772 (vid. pág. 4).
- Machine Learning Mastery (I)*. <https://machinelearningmastery.com/how-to-choose-loss-functions-when-training-deep-learning-neural-networks/>. Consulta: 18 de Mayo de 2019 (vid. pág. 29).
- Machine Learning Mastery (II)*. <https://machinelearningmastery.com/how-to-stop-training-deep-neural-networks-at-the-right-time-using-early-stopping/>. Consulta: 18 de Mayo de 2019 (vid. pág. 29).
- Manjón, José V y col. (2010a). «MRI superresolution using self-similarity and image priors». En: *Journal of Biomedical Imaging* 2010, pág. 17 (vid. págs. 18, 32, 59, 60).
- Manjón, José V y col. (2010b). «Non-local MRI upsampling». En: *Medical image analysis* 14.6, págs. 784-792 (vid. págs. 32, 37, 45, 47, 60).
- Mathieu, Michael, Camille Couprie y Yann LeCun (2015). «Deep multi-scale video prediction beyond mean square error». En: *arXiv preprint arXiv:1511.05440* (vid. pág. 49).
- Niu, Hao y col. (2017). «Prevalence and incidence of Alzheimer's disease in Europe: A meta-analysis». En: *Neurología (English Edition)* 32.8, págs. 523-532 (vid. pág. 4).
- Ortiz, Andres y col. (2016). «Ensembles of deep learning architectures for the early diagnosis of the Alzheimer's disease». En: *International journal of neural systems* 26.07, pág. 1650025 (vid. págs. 33, 72).
- Pham, Chi-Hieu, Ronan Fablet y François Rousseau (2017). «Multi-scale brain MRI super-resolution using deep 3D convolutional networks». En: (vid. págs. 32, 60).
- Pham, Chi-Hieu y col. (2017). «Brain MRI super-resolution using deep 3D convolutional networks». En: *2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)*. IEEE, págs. 197-200 (vid. págs. 32, 60).
- Quora*. <https://www.quora.com/What-is-max-pooling-in-convolutional-neural-networks>. Consulta: 16 de Mayo de 2019 (vid. pág. 24).
- Radiology Café*. <https://www.radiologycafe.com/radiology-trainees/frcr-physics-notes/t1-t2-and-pd-weighted-imaging>. Consulta: 30 de Junio de 2019 (vid. pág. 16).



- Radiología Club*. <https://radiologiaclub.com/2016/05/10/utilidad-de-la-resonancia-magnetica-en-el-diagnostico-de-alzheimer/>. Consulta: 13 de Mayo de 2019 (vid. pág. 17).
- Ramsundar, Bharath y Reza Bosagh Zadeh (2018). *TensorFlow for deep learning: from linear regression to reinforcement learning*. "O'Reilly Media, Inc." (vid. pág. 25).
- Research Gate (I)*. [https://www.researchgate.net/figure/Data-augmentation-using-semantic-preserving-transformation-for-SBIR\\_fig2\\_319413978](https://www.researchgate.net/figure/Data-augmentation-using-semantic-preserving-transformation-for-SBIR_fig2_319413978). Consulta: 18 de Mayo de 2019 (vid. pág. 30).
- Romero, José E, Pierrick Coupé y José V Manjón (2017). «HIPS: A new hippocampus subfield segmentation method». En: *NeuroImage* 163, págs. 286-295 (vid. pág. 45).
- Romero, José E y col. (2015). «NABS: non-local automatic brain hemisphere segmentation». En: *Magnetic resonance imaging* 33.4, págs. 474-484 (vid. pág. 47).
- Roy, Snehashis, Aaron Carass y Jerry L Prince (2013). «Magnetic resonance image example-based contrast synthesis». En: *IEEE transactions on medical imaging* 32.12, págs. 2348-2363 (vid. págs. 32, 65).
- Science*. <https://www.sciencemag.org/news/2018/05/ai-researchers-allege-machine-learning-alchemy>. Consulta: 18 de Mayo de 2019 (vid. pág. 29).
- Stack Exchange (I)*. <https://datascience.stackexchange.com/questions/38118/what-is-the-difference-between-upsampling-and-bi-linear-upsampling-in-a-cnn>. Consulta: 16 de Mayo de 2019 (vid. pág. 24).
- Stack Exchange (II)*. <https://ai.stackexchange.com/questions/40/what-is-the-dropout-technique>. Consulta: 18 de Mayo de 2019 (vid. pág. 30).
- Suk, Heung-Il y Dinggang Shen (2013). «Deep learning-based feature representation for AD/MCI classification». En: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, págs. 583-590 (vid. pág. 33).
- The Adecco Group (2018). *Informe Infoempleo Adecco. Oferta y demanda de empleo en 2018*. <https://iestatic.net/infoempleo/documentacion/Informe-infoempleo-adecco-2018.pdf>. [Consulta: 20 de Junio de 2019] (vid. pág. 77).
- Towards Data Science (I)*. <https://towardsdatascience.com/convolutional-neural-networks-from-the-ground-up-c67bb41454e1>. Consulta: 16 de Mayo de 2019 (vid. pág. 22).
- Towards Data Science (II)*. <https://towardsdatascience.com/residual-blocks-building-blocks-of-resnet-fd90ca15d6ec>. Consulta: 18 de Mayo de 2019 (vid. pág. 30).

- Tustison, Nicholas J y col. (2010). «N4ITK: improved N3 bias correction». En: *IEEE transactions on medical imaging* 29.6, pág. 1310 (vid. págs. 45, 47).
- Vemulapalli, Raviteja, Hien Van Nguyen y Shaohua Kevin Zhou (2015). «Unsupervised cross-modal synthesis of subject-specific scans». En: *Proceedings of the IEEE International Conference on Computer Vision*, págs. 630-638 (vid. págs. 32, 65).
- Vincent, Pascal y col. (2010). «Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion». En: *Journal of machine learning research* 11.Dec, págs. 3371-3408 (vid. pág. 32).
- Wang, Yun-Heng y col. (2014). «Sparse representation-based MRI super-resolution reconstruction». En: *Measurement* 47, págs. 946-953 (vid. pág. 32).
- Yang, Jianchao y col. (2010). «Image super-resolution via sparse representation». En: *IEEE transactions on image processing* 19.11, págs. 2861-2873 (vid. págs. 32, 65).
- Zhang, Yongqin y col. (2019). «Super-resolution reconstruction of neonatal brain magnetic resonance images via residual structured sparse representation». En: *Medical image analysis* 55, págs. 76-87 (vid. pág. 32).
- Zhang, Yudong y col. (2015). «Detection of subjects and brain regions related to Alzheimer's disease using 3D MRI scans based on eigenbrain and machine learning». En: *Frontiers in computational neuroscience* 9, pág. 66 (vid. pág. 33).