

Universidad de Alcalá Escuela Politécnica Superior

Grado en Ingeniería Informática

Trabajo Fin de Grado

Detección de cáncer de próstata en imagen médica mediante
Deep Learning

ESCUELA POLITECNICA

Autor: Javier Fuentes Fernández

Tutor: Daniel Pizarro Pérez

Cotutor: David Fuentes Jiménez

2020

UNIVERSIDAD DE ALCALÁ
ESCUELA POLITÉCNICA SUPERIOR

Grado en Ingeniería Informática

Trabajo Fin de Grado

**Detección de cáncer de próstata en imagen médica mediante
Deep Learning**

Autor: Javier Fuentes Fernández

Tutor: Daniel Pizarro Pérez

Cotutor: David Fuentes Jiménez

Tribunal:

Presidente: Luciano Boquete Vázquez

Vocal 1º: María Soledad Escudero Hernanz

Vocal 2º: Daniel Pizarro Pérez

Fecha de depósito: 9 de Diciembre de 2020

Agradecimientos

Agradecer a mi familia, a mi novia y a mis amigos el apoyo, comprensión y ánimo que me han proporcionado durante la carrera y en especial durante la elaboración de este proyecto.

Por último, agradecer a mi tutor Daniel y en especial a mi co-tutor David por la disponibilidad, paciencia y confianza depositada en mí durante todo el proceso.

*Los que pueden imaginar cualquier cosa, pueden crear lo
imposible*

Alan Turing

Resumen

Este trabajo de fin de grado propone un método de apoyo al diagnóstico de cáncer de próstata a partir de imágenes de resonancia magnética (RM) multiparamétrica.

Para ello se plantea un sistema en el cual se pre-procesan dichas imágenes y hace uso de redes neuronales para detectar y localizar el posible cáncer.

Se obtiene como resultado la ubicación del cáncer en el espacio ocupado por la próstata y la significancia de la posible anomalía encontrada.

Con estos resultados se pretende facilitar la tarea de los radiólogos y permitir que el método de detección por resonancia magnética sea menos costoso y logrando así sustituir a otros métodos más invasivos y con menos especificidad como las biopsias transrectales.

Palabras clave: Detección de cáncer de próstata, redes neuronales profundas, resonancia magnética.

Abstract

This final degree project proposes a method that can support the diagnose of prostate cancer using multiparametric magnetic resonance imaging (MRI).

For this, a system is proposed in which these images are pre-processed and making use of neural networks it is able detect and locate the possible cancer tumour.

The result is the location of the cancer in the space occupied by the prostate and the significance of the possible anomaly found.

These results are intended to facilitate the task of radiologists and allow the magnetic resonance detection method to be less expensive, thus succeeding in replacing other more invasive and less specific methods such as transrectal biopsies.

Keywords: prostate cancer detection, deep neural networks, MRI..

Resumen extendido

Este trabajo tiene por objetivo la implementación de un sistema de análisis de imágenes de resonancia magnética de próstata. Este sistema está basado en redes neuronales profundas y permitirá localizar y determinar la severidad del cáncer de próstata, ayudando a su diagnóstico.

Como se observa en la figura 2, el sistema toma como entrada imágenes de resonancia magnética de la próstata. Este tipo de imágenes se obtienen mediante ondas de radio y un campo magnético de alta potencia, generando imágenes detalladas de órganos y tejidos. El sistema propuesto obtiene un mapa con la posición en las imágenes de las masas que podrían ser malignas y la probabilidad de presentar cáncer de cada imagen correspondiente al volumen de la próstata.

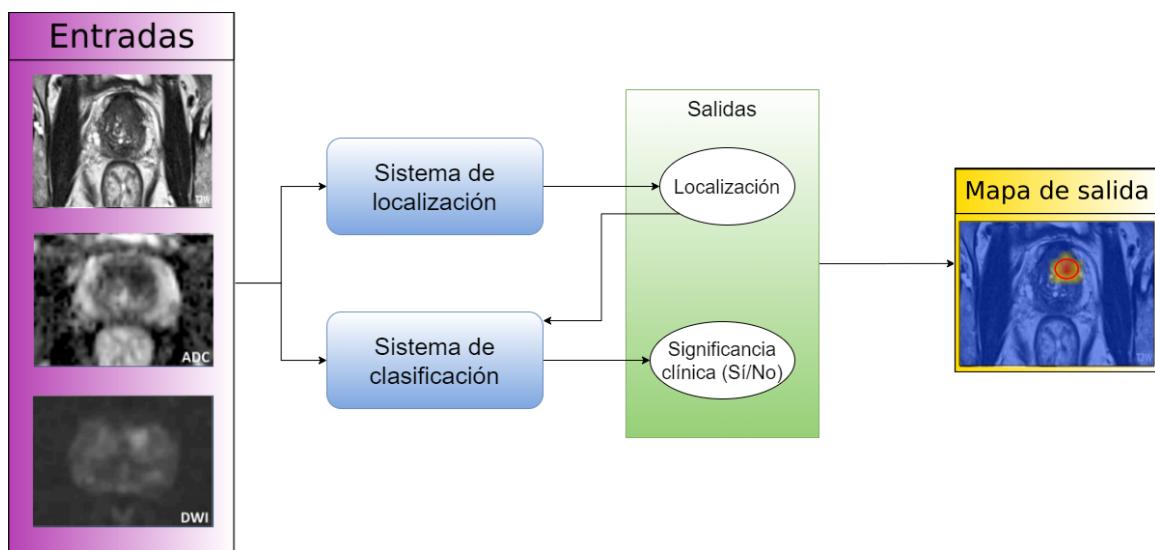


Figura 2: Esquema de sistema propuesto.

El uso de ese método supone un gran avance respecto al uso de biopsias transrectales, método preferente utilizado en la actualidad y que suele presentar complicaciones para la mayoría de los pacientes tras la operación tanto leves, como fiebres o retención urinaria, hasta algunas más graves como sangrado rectal [1]. El uso de resonancia magnética no es invasivo y se ha demostrado que su alta especificidad y sensibilidad pueden evitar un gran número de las biopsias transrectales, así como las complicaciones derivadas de éstas [2]. El principal motivo por el que este método no se usa en la actualidad es porque el coste asociado es mucho más alto, debido a que cada imagen necesita un tiempo de análisis alto y la opinión de un radiólogo experimentado. Este proyecto reduce drásticamente el tiempo de necesario para analizar cada imagen facilitando así la tarea del radiólogo y mejorando el diagnóstico de este tipo de cáncer.

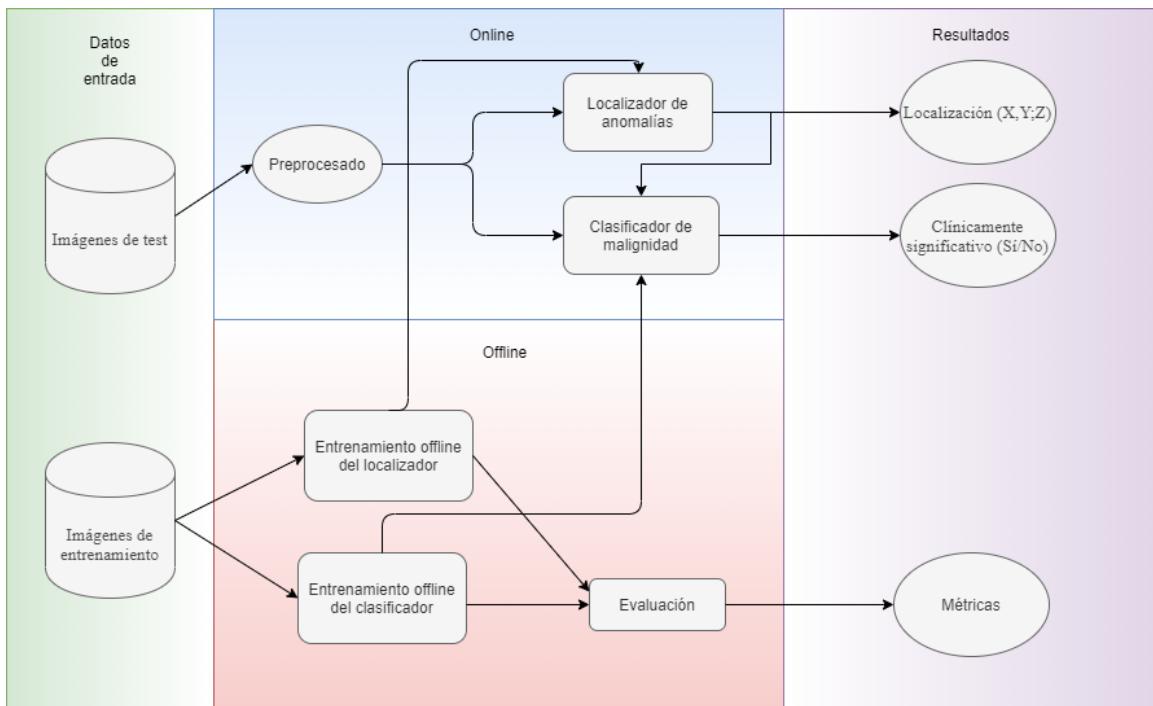


Figura 3: Diagrama de bloques completo del sistema propuesto.

El sistema implementado en este trabajo se compone de varias etapas y fases, como se explica en la figura 3, en la que se puede observar que el sistema consta de una fase *offline*, y otra *online*.

1. Offline

- **Entrenamiento:** se utilizan las imágenes de resonancia magnética proporcionadas por la base de datos Prostatax [3], sobre las que se realiza un preprocesado. Dichas imágenes serán utilizadas para entrenar el sistema de localización y el sistema de clasificación.
- **Evaluación:** se realiza una evaluación exhaustiva de los dos algoritmos desarrollados mediante el set de test y las métricas previamente definidas.

2. Online

- **Preprocesado:** las imágenes de entrada que conforman el volumen de la próstata se ajustarán y preseleccionarán dependiendo de la técnica utilizada para obtener la imagen o el plano anatómico que se observe para ser analizadas por las diferentes redes neuronales profundas.
- **Detector de anomalías:** basándose en el entrenamiento offline, este algoritmo se encarga de proporcionar, en caso de haberla, la ubicación tridimensional de la anomalía.
- **Clasificador de malignidad:** al igual que el anterior algoritmo, éste hace uso de su entrenamiento para analizar el volumen e indicar si la anomalía es significativa.

Índice general

Resumen	ix
Abstract	xI
Resumen extendido	xIII
Índice general	xv
Índice de figuras	xIX
Índice de tablas	xxI
Lista de acrónimos	xxIII
Lista de símbolos	xxIII
1. Introducción	1
1.1. Sistema propuesto	1
1.2. Organización de la memoria	2
2. Conocimientos previos	5
2.1. La resonancia magnética.	5
2.1.1. Modalidades	6
2.2. Las redes neuronales profundas.	7
2.2.1. Fundamentos y capas principales	8
2.2.2. Regularización	12
2.2.3. Principales Optimizadores	13
2.2.4. Parámetros de entrenamiento	13
2.3. La detección de cáncer	15
2.3.1. Aspectos básicos sobre el cáncer	15
2.3.2. Detección de cáncer clásica	15
2.3.3. Detección de cáncer con machine learning	15
2.3.4. El cáncer de próstata	17

3. Desarrollo	19
3.1. Introducción	19
3.2. Definición del problema	20
3.3. Arquitectura propuesta	20
3.3.1. Arquitectura de localización de anomalías	20
3.3.1.1. Arquitecturas unimodales de clasificación	21
3.3.2. Arquitecturas multimodales de clasificación	24
3.3.2.1. Arquitectura de fusión por tensor de entrada	25
3.3.2.2. Arquitectura de fusión de etapas unimodales	25
3.4. Condiciones de entrenamiento	27
3.4.1. Técnicas de aumento de datos	28
4. Resultados	31
4.1. Introducción	31
4.2. Bases de datos	31
4.2.1. Características	31
4.2.2. Lectura de datos	32
4.2.3. Muestra de imágenes	32
4.3. Función de coste y métricas de calidad	33
4.3.1. Función de coste	33
4.3.2. Ráiz del error cuadrático medio	34
4.3.3. Matriz de confusión	34
4.3.4. Accuracy o exactitud	35
4.3.5. Ratio de verdaderos positivos (TPR)	35
4.3.6. Valor predictivo positivo (PPV)	35
4.3.7. F-score	35
4.3.8. Curva de Característica Operativa del Receptor (ROC)	36
4.3.9. Área bajo la curva (AUC)	36
4.4. Resultados experimentales	36
4.4.1. Resultados de redes unimodales de clasificación	37
4.4.1.1. Matrices de confusión	37
4.4.1.2. Curvas ROC	37
4.4.1.3. Resultados de conjunto	39
4.4.2. Arquitecturas multimodales de clasificación	39
4.4.2.1. Arquitectura de fusión por tensor de entrada	39
4.4.2.2. Arquitectura de fusión de etapas unimodales	41
4.4.3. Resultados de red de localización	44

5. Conclusiones y líneas futuras	47
5.1. Conclusiones	47
5.2. Líneas futuras	48
6. Presupuesto	49
6.1. Costes de equipamiento	49
6.1.1. Equipamiento hardware utilizado:	49
6.1.2. Recursos software utilizados:	49
6.2. Costes Mano de obra	49
6.3. Costes Totales	50
Bibliografía	51
Apéndice A. Manual de usuario	57
A.1. Requisitos previos	57
A.2. Estructura del programa	58
A.3. Ejecución del programa	58
A.4. Resultados de la aplicación	60

Índice de figuras

2.	Esquema de sistema propuesto.	XIII
3.	Diagrama de bloques completo del sistema propuesto.	XIV
1.1.	Diagrama de bloques completo del sistema propuesto.	2
2.1.	Resonancia magnética cerebral secuencia T1 de alta resolución obtenida de [4]	5
2.2.	Elementos de una neurona artificial.	7
2.3.	Ejemplo de extracción de características.	8
2.4.	Ejemplo de aplicar un filtro 3x3 sobre una imagen de 7x7.	8
2.5.	Ejemplo de aplicar un filtro 3D sobre un volumen tridimensional.	9
2.6.	Ejemplo de matriz 4x4 en la cual se realiza un maxpooling de 2x2 con un stride de 2.	9
2.7.	Ejemplo de función de activación lineal.	10
2.8.	Comparación visual de RELU y Leaky RELU obtenida de [5]	10
2.9.	Gráfica de la función Sigmoidal	11
2.10.	Gráfica de la función Tangente Hiperbólica	11
2.11.	Ejemplo de aplicar dropout a una red neuronal. Obtenido del artículo [6]	12
2.12.	Ejemplo de realizar aumento de datos sobre una imagen [7]	12
2.13.	Pseudocódigo de optimizador ADAM.[8]	14
2.14.	Posibles situaciones con ratios de aprendizaje distintos.	14
2.15.	Ejemplo de del uso de un árbol de decisión basado en el estado hormonal de mujeres obtenido de [9].	16
3.1.	Diagrama de bloques completo del sistema propuesto.	19
3.2.	Ejemplos de entrada y salida del localizador. A la izquierda se muestra el volumen de entrada de la modalidad Ktrans y una de sus secciones bidimensionales, mientras que a la derecha se muestra el volumen de salida con la localización de la anomalía plasmada por una esfera blanca a lo largo del volumen, en la sección esta esfera se muestra en forma de círculo.	21
3.3.	23
3.4.	Diagrama de arquitectura genérica unimodal.	24
3.5.	Diagrama de arquitectura de fusión 1. Se observan las 5 modalidades como entrada.	26

3.6. Diagrama de arquitectura de fusión 2	27
3.7. Ejemplo de aplicar técnica de rotación.	29
3.8. Ejemplo de aplicar técnica de espejar verticalmente.	29
3.9. Ejemplo de aplicar técnica de espejado horizontal.	29
3.10. Ejemplo de aplicar técnica de desplazamiento.	30
 4.1. Visualización de la composición de un volumen 3D de zona pélvica.	32
4.2. Ejemplo del error de la recta de regresión a cada punto.	34
4.3. Posición de cada parámetro en la matriz de confusión.	35
4.4. Ejemplo del espacio en el que se representa la curva ROC.	36
4.5. Ejemplo de diferentes curvas y valores de AUC. Imagen obtenida de [10].	37
4.6. Matriz de confusión de arquitectura de fusión por tensor de entrada.	41
4.7. Curva ROC de arquitectura de fusión por tensor de entrada.	42
4.8. Matriz de confusión de arquitectura de fusión de etapas unimodales.	42
4.9. Curva ROC de arquitectura de fusión de etapas unimodales.	43
 A.1. Archivo <code>codigo.py</code> abierto en Pycharm.	59
A.2. Archivo <code>funciones.py</code> abierto en Pycharm.	59
A.3. Archivo <code>funciones.py</code> abierto en Pycharm.	60
A.4. Gráfica de curva ROC generada en ejecución.	60
A.5. Métricas finales y matriz de confusión.	61

Índice de tablas

3.1. Arquitectura de red de localización.	22
3.2. Arquitectura unimodal.	25
3.3. Arquitectura de fusión 1.	26
3.4. Arquitectura de fusión 2.	28
4.1. Propiedades de imágenes en Prostatex.	31
4.2. Imágenes de ejemplo..	33
4.3. Matrices de confusión de arquitecturas unimodales	38
4.4. Curvas ROC de arquitecturas unimodales.	40
4.5. Métricas de todas las arquitecturas unimodales.	41
4.6. Métricas obtenidas de fusión por tensor de entrada y arquitecturas unimodales anteriores.	41
4.7. Métricas obtenidas de fusión de etapas unimodales, fusión por tensor de entrada y arquitecturas unimodales anteriores.	43
4.8. Resultados cualitativos y cuantitativos de la red de localización propuesta.	44

Lista de acrónimos

DNN	Deep Neural Network.
GEINTRA	Grupo de ingeniería Electrónica aplicada a espacios inteligentes y transporte.
ML	Machine Learning.
PSA	Prostate Specific Antigen.
RMM	Resonancia magnética multiparamétrica.
RMSE	Raíz del error cuadrático medio.
TFG	trabajo de fin de grado.

Capítulo 1

Introducción

El objetivo de este trabajo es la implementación de un sistema de ayuda al diagnóstico del cáncer de próstata basado en el análisis de imágenes de Resonancia magnética multiparamétrica (RMM) mediante el uso de redes neuronales profundas. Se pretende determinar la existencia de cáncer, así como su severidad y su localización en el volumen de la próstata.

En los últimos años se ha visto un crecimiento exponencial en la aplicación de las redes neuronales profundas que tratan de clasificar multitud de imágenes distintas en las categorías correctas. Uno de los impulsores de estos sistemas es el reto Imagenet [11]. Este reto surge en 2009 con el objetivo de dotar a la base de datos de sinónimos WordNet [12] de imágenes en alta calidad que representasen cada una de las categorías. Desde entonces multitud de expertos y organizaciones compiten anualmente en lo que se conoce como "ImageNet Large Scale Visual Recognition Challenge" [13] para obtener las menores tasas de fallos al categorizar imágenes en 1000 categorías distintas.

Los avances logrados en la detección de características y clasificación de imágenes se han aplicado posteriormente a multitud de ámbitos entre los que destaca la medicina, donde el objetivo es diferenciar imágenes que presentan algún tipo de lesión pudiendo así diagnosticarse multitud de enfermedades distintas, en ocasiones basándose en características o asociaciones tan abstractas que fácilmente podrían escapar a los humanos.

Uno de los principales objetivos de su uso en la medicina es facilitar el diagnóstico de algunos tipos de cáncer, que normalmente requieren el trabajo de radiólogos expertos para poder determinar a través de estas imágenes médicas la existencia de un tumor canceroso. De esta manera se pueden encontrar en el mundo científico altas tasas de éxito en la detección del cáncer de pulmón [14, 15] o incluso mejores que las de expertos en radiología en el caso del cáncer de mama [16].

El presente trabajo se enmarca dentro de las líneas de investigación del grupo de investigación Grupo de ingeniería Electrónica aplicada a espacios inteligentes y transporte (GEINTRA), de la Universidad de Alcalá, especialmente en la subdisciplina del tratamiento y análisis de imagen médica.

1.1. Sistema propuesto

Para conseguir el objetivo planteado se propone un sistema dividido en cuatro etapas que se muestran en la siguiente figura 1.1 y que se explican brevemente a continuación.

Las fases que se observan en la figura anterior consisten en:

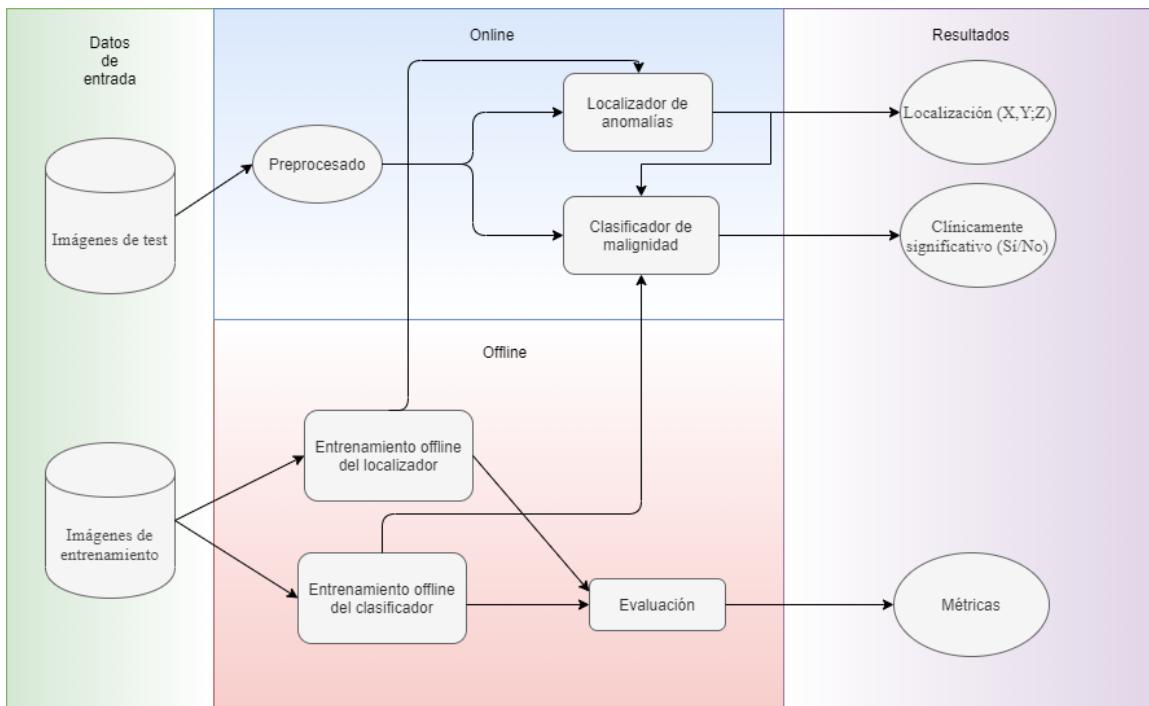


Figura 1.1: Diagrama de bloques completo del sistema propuesto.

- **Obtención de imágenes de entrada y entrenamiento:** se utilizan imágenes de análisis clínico realizados para diagnóstico médico. En este caso se realiza el entrenamiento con las imágenes de la base de datos Prostataex [3] sobre las que se hace una selección de modalidades de imagen y posteriormente un aumento de datos.
- **(Offline) Entrenamiento de localizador y clasificador:** esta fase es crucial para el correcto funcionamiento del sistema en ella se busca un conjunto de pesos que permitan realizar la tarea final correctamente. Se deben escoger y optimizar los parámetros de entrenamiento apropiados para la tarea a realizar, que permitan un refinamiento de los pesos establecidos en cada iteración, logrando así la mejor detección y localización posible. También se debe garantizar que los datos de entrada permiten una buena generalización del sistema.
- **(Online) Preprocesado de imágenes y detección y clasificación de anomalías:** en esta fase se ajustarán las imágenes de entrada para poder ser procesadas por el algoritmo clasificador y localizador entrenado de forma offline. Las imágenes son clasificadas según la significancia de las anomalías y el algoritmo de localización predice su ubicación.
- **Evaluación de resultados:** en esta última fase se comparan y optimizan los resultados tanto por el clasificador como por el detector utilizando para ello una serie de métricas establecidas previamente que se analizan en profundidad.

1.2. Organización de la memoria

La memoria de este trabajo de fin de grado (TFG) se organiza en cinco capítulos principales. Una breve introducción del objetivo y el sistema desarrollado en el capítulo 1. En el capítulo 2 se explican en profundidad todos aquellos conceptos teóricos relevantes para la comprensión de este TFG así como el estado del arte de las técnicas que incumben al proyecto. Posteriormente en el capítulo 3 se profundiza en

las arquitecturas desarrolladas, métodos de entrenamiento y preparación de datos. El capítulo 4 expone comparativamente los mejores resultados obtenidos que permiten extraer una serie de conclusiones y líneas futuras de trabajo que se presenta en el capítulo 5.

Por último en el capítulo 6 se estima el presupuesto del proyecto.

Capítulo 2

Conocimientos previos

En este apartado del documento se explican en profundidad algunos conceptos teóricos esenciales para la realización del proyecto. Entre ellos se explica en que consiste la imagen por **RMM** y sus aplicaciones en la detección de cáncer. A continuación, se expone qué es y cómo se diagnostica el cáncer de próstata, exponiendo las técnicas clásicas de detección y mencionando algunos trabajos previos con métodos innovadores. Por último, se explican los fundamentos de las redes neuronales profundas, sus métodos de entrenamiento y aquellos tipos que más se relacionan con este trabajo.

2.1. La resonancia magnética.

La **RMM** se patentó para el área de la medicina en 1974 como un método que permitía distinguir cambios en los tejidos del cuerpo humano. Para obtener este tipo de imágenes se utilizan fuertes campos magnéticos y ondas de radio que modifican la orientación de los protones de los diferentes tejidos, liberando energía con cada re-orientación. Basándose en el tiempo de re-orientación y la cantidad de energía se pueden distinguir diferentes órganos o tejidos. Esto permite obtener imágenes como la de la figura 2.1.

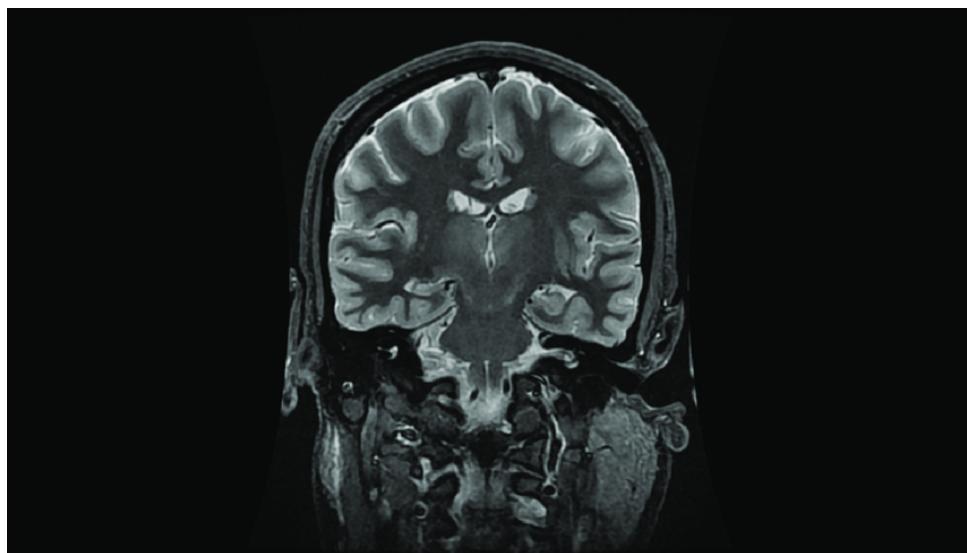


Figura 2.1: Resonancia magnética cerebral secuencia T1 de alta resolución obtenida de [4]

Esta técnica es ventajosa frente a otras ampliamente conocidas como rayos-x, ultrasonidos o ecografías, ya que no es invasiva al no utilizar ningún tipo de radiación. Además, las imágenes obtenidas tienen alta

resolución, pudiendo también obtenerlas desde diferentes ángulos o incluso en 3D. En la actualidad las imágenes por **RMM** se utilizan en cualquier parte del cuerpo en función del órgano o tejido que se quiera estudiar, entre ellas abdomen, tórax, cráneo, zona lumbar o la pelvis, en la que podemos encontrar la próstata.

La **RMM** ha cobrado importancia en la detección del cáncer debido a las características de estos tejidos, como por ejemplo su densidad, pudiéndose observar ciertos patrones en la **RMM** que permiten distinguirlos del tejido sano. Además, esta prueba puede ser útil para determinar el tamaño o la ubicación del tumor canceroso. Es por ello que la **RMM** se utiliza como herramienta para el diagnóstico de múltiples tipos de cáncer y se siguen investigando sus posibles aplicaciones. Para el diagnóstico mediante este tipo de modalidad de imagen se requiere el trabajo de un radiólogo experto que analice las imágenes y sea capaz de identificar los patrones plasmados en las pruebas. La principal investigación que se está realizando en la actualidad es sobre su utilización para un cribado previo al resto de pruebas, es decir realizar estas pruebas menos invasivas de manera masiva y con mayor frecuencia a la población para detectar aquellos casos que requieran análisis posteriores, como por ejemplo cribado en cáncer de próstata [17] o en cáncer de mama [18].

Se debe mencionar también el efecto que tiene el uso del agente de contraste en **RMM**. Esta sustancia se suministra al paciente ya sea por vía oral, rectal, o intravenosa, y permite que cuando se realice la **RMM** se observen con mejor claridad cambios en el flujo de la sangre, en el sistema linfático y la localización de crecimientos benignos [19]. Cuando esta sustancia no se administra se aprecian mejor el tamaño y forma de diferentes órganos y huesos. Además, los agentes de constante se investigan como uno de los principales factores que pueden mejorar los resultados de las resonancias y se realizan pruebas con distintos tipos de agente para encontrar lesiones específicas. [20, 21].

A continuación se exponen las principales ventajas y desventajas de la **RMM**:

- **Principales ventajas de la resonancia magnética.** La principal ventaja de la resonancia magnética frente a otros tipos de pruebas similares reside en que esta prueba no es invasiva, por lo que el paciente no sufrirá efectos secundarios ni necesitará un tiempo de recuperación, y no utiliza ningún tipo de radiación ionizante como ocurre en otras modalidades como los rayos-x. Esto permite repetir o realizar pruebas intensivas de resonancia a un mismo paciente sin riesgo alguno para su salud. Además, las imágenes que se obtienen de esta prueba alcanzan un alto nivel de detalle, pudiendo también obtener imágenes desde diferentes direcciones e incluso generar un volumen 3D de la zona estudiada.
- **Principales desventajas de la resonancia magnética.** La desventaja más clara de esta prueba es la necesidad de un radiólogo experto que analice las imágenes obtenidas y pueda determinar la existencia de alguna lesión significativa. Otras desventajas menos relevantes son el largo tiempo necesario para realizar la prueba, el alto coste asociado al que se suma la tarea del radiólogo y por último, la posibilidad remota de que el paciente sufra una reacción alérgica a la sustancia utilizada como contraste al realizar la prueba [22].

2.1.1. Modalidades

Controlando una serie de parámetros del escáner, principalmente la densidad de protones emitidos, se pueden obtener diferentes tipos de imágenes médicas, seguidamente se describen las diferentes modalidades que se utilizan en este **TFG**:

- **T2-ponderadas:** una de estas variaciones se conoce como T2 y es de normalmente utilizada para visualizar fluidos dentro de los tejidos, pudiendo detectar por ejemplo hemorragia dentro de

un tumor. De este tipo de imágenes se dispone en la base de datos de tres variaciones acordes a los distintos planos anatómicos (transversal, sagital y coronal) en algunas ocasiones estas pruebas aparecen repetidas para un mismo paciente.

- **Resonancia magnética dinámica mejorada por contraste o DCE-MR:** para la obtención imágenes se mide la acumulación de un agente de contraste en vasos sanguíneos lo que permite calcular la permeabilidad de los capilares y representarlo en forma de imagen tridimensional.
- **Imágenes ponderadas por difusión o DWI :** Esta técnica usa la difusión de las moléculas de agua para generar imágenes de contraste creando un mapa de Coeficiente de Difusión Aparente o ADC por sus siglas en inglés.

2.2. Las redes neuronales profundas.

En esta sección se van a explicar los fundamentos de las redes neuronales profundas, sus métodos de entrenamiento y aquellos tipos que se relacionen más con este trabajo. Una Deep Neural Network (DNN) es un tipo de red neuronal artificial que imita el funcionamiento de las neuronas de la corteza visual humana, permitiendo realizar tareas como clasificación [23], segmentación de imágenes [24], reconstrucción de sólidos [25].

Para ello utiliza un gran número de elementos de procesamiento que se denominan neuronas, las cuales se organizan en estructuras para trabajar hacia un objetivo común. Una neurona clásica de una DNN se compone de los siguientes subelementos:

- **Entradas:** reciben los datos del exterior ya sean los datos iniciales del sistema o el resultado de una neurona anterior.
- **Pesos:** son coeficientes que determinan la importancia de cada una de las entradas de la neurona. Estos pesos se actualizan durante el entrenamiento del modelo para alcanzar el objetivo.
- **Regla de propagación:** se opera con las entradas y sus respectivos pesos. Normalmente se realiza la suma ponderada de las entradas y pesos.
- **Función de activación:** transforma el valor obtenido anteriormente en un valor de activación acotado según la función escogida, dicho valor acotado es la salida de la neurona.

En la figura 2.2 se muestra un esquema de los elementos de una neurona artificial.

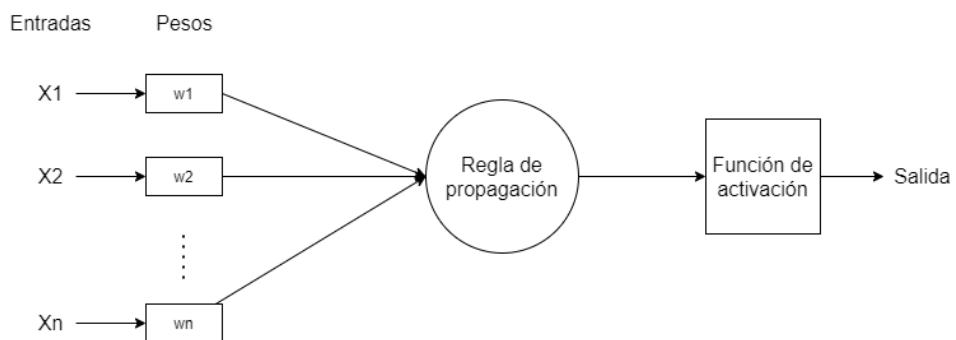


Figura 2.2: Elementos de una neurona artificial.

Las neuronas se organizan en capas que pueden desempeñar diferentes funciones, en la siguiente sección se profundiza en los distintos tipos.

2.2.1. Fundamentos y capas principales

En este apartado vamos a explicar las diferentes capas de neuronas y las partes más importantes de cada una:

- **Capas convolucionales:** estas capas operan sobre los datos de entrada sobre los que realizan sucesivos cálculos llamados convoluciones discretas, haciendo usos de filtros finitos o kernels. Estas capas realizan una de las funciones principales de las DNN, ya que extraen las características de las imágenes. Las capas exteriores obtienen los rasgos más simples, mientras que las más profundas resultan en un alto nivel de abstracción, como se observa en la figura 2.3. Los ya mencionados filtros pueden tener una, dos o tres dimensiones, dando lugar a una convolución 1D, 2D o 3D. En la figura 2.4 se observa un caso de convolución 2D.

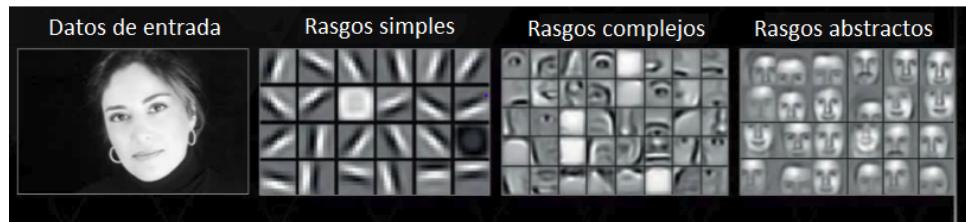


Figura 2.3: Ejemplo de extracción de características.

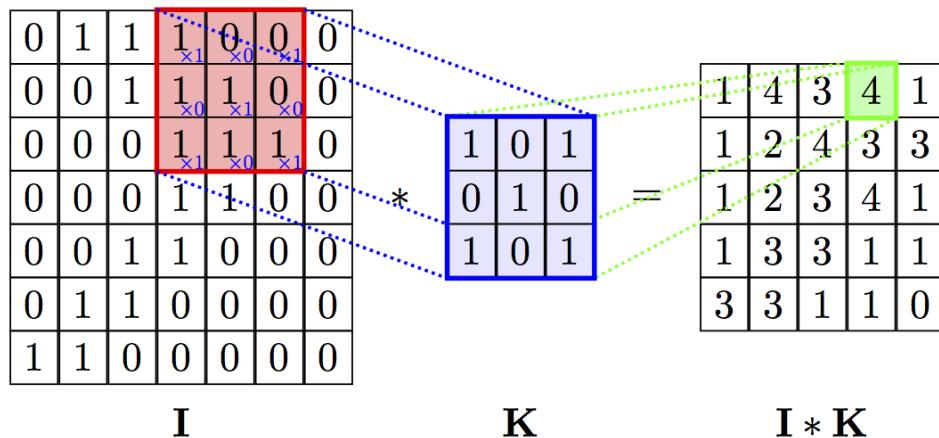


Figura 2.4: Ejemplo de aplicar un filtro 3x3 sobre una imagen de 7x7.

En el problema que ocupa a este TFG se utilizan resultados de pruebas médicas en los que se presenta el volumen completo de la próstata. Es por ello que se utilizan filtros tridimensionales con el objetivo de mantener todos los patrones que pueden existir en las tres dimensiones que se perderían en caso de realizar convoluciones 2D. En la figura 2.5 se observa un ejemplo de convoluciones 3D como las realizadas en el proyecto.

- **Densas o Fully Connected:** se utilizan neuronas clásicas interconectadas entre capas al igual que en los perceptrones multicapa [26]. Suelen realizar la tarea final de clasificación, convirtiendo mapas de características en valores concretos.
- **Normalización de lotes o Batch Normalization:** se encarga de regular las salidas de la red mediante la normalización de las salidas de la activación anterior. Para ello se resta a cada lote la media del lote y se divide por la desviación típica del lote. Se utilizan además dos parámetros entrenables por lo que se multiplicará a los datos normalizados, uno de desviación típica y otro de

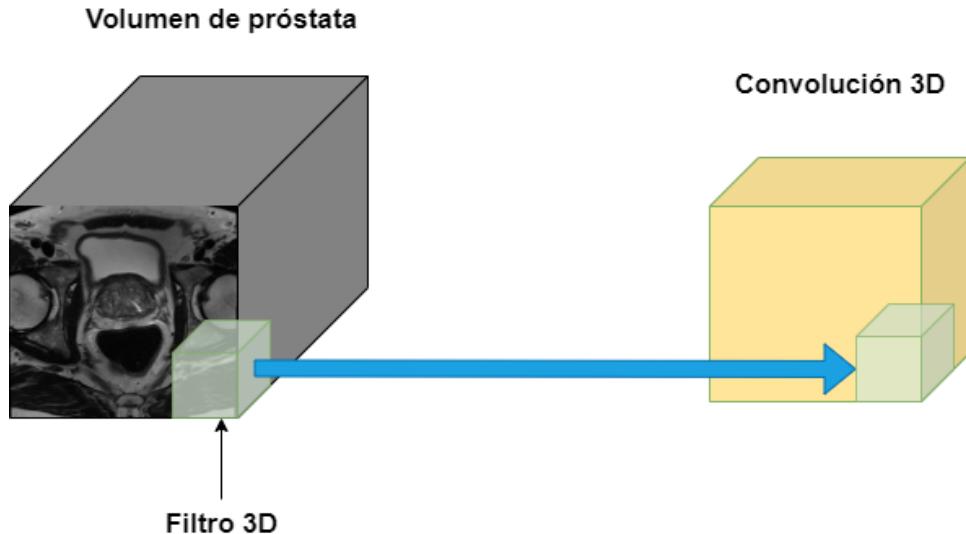


Figura 2.5: Ejemplo de aplicar un filtro 3D sobre un volumen tridimensional.

media. Este permite mantener entre capas unos datos con una media aproximada de cero y una desviación típica de 1, evitando saturaciones y desviaciones de la covarianza.

- **Capa de Pooling:** es un proceso de reducción que convierte un conjunto de valores de una subsección en un único valor. Esto ayuda a dar relevancia a aquellas características de mayor energía y reduce el coste computacional, ya que se reduce el número de parámetros. Se pueden utilizar diferentes tipos como average pooling donde se realiza la media entre los valores del kernel o max pooling si se conserva aquel valor más alto como en el ejemplo de la imagen 2.6.



Figura 2.6: Ejemplo de matriz 4x4 en la cual se realiza un maxpooling de 2x2 con un stride de 2.

- **Capas de activación:** Estas funciones permiten que las redes neuronales consigan obtener respuestas no lineales aumentando enormemente su potencial, ya sea como clasificadores o regresores. Aunque existen muchos tipos de capas de activación, a continuación, explicaremos las más conocidas y las utilizadas en este trabajo:

- **Lineal:** Este tipo de activación es utilizada generalmente en capas de salida para problemas de regresión debido a su tipo de respuesta de salida, que además es simétrica tanto en el cuadrante negativo como en el positivo. En la figura 2.7 observamos su representación gráfica.
- **Rectified Linear Unit (ReLU):** este tipo de activación permite redes más profundas y fáciles de entrenar debido a su sencillez de cálculo, puesto que es lineal en el cuadrante positivo y nula para cualquier valor negativo. Además es invariante a escala, lo que significa que un cambio

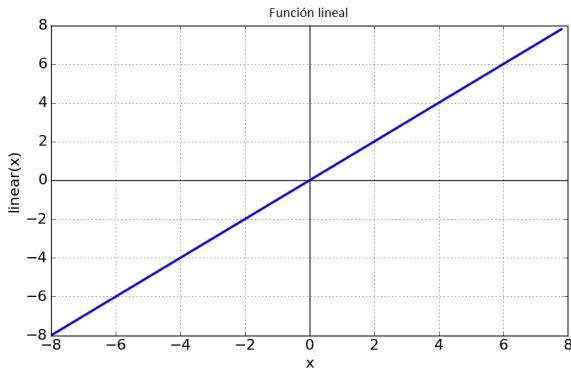


Figura 2.7: Ejemplo de función de activación lineal.

de escala en la entrada produce un cambio en la parte lineal de la función. Por último, gracias a que la función no aplica saturación en su parte lineal se evitan los problemas de desvanecimiento de gradiente. En cambio, pueden surgir problemas con aquellas entradas negativas. Esto se soluciona en su variante LeakyReLU que se explica a continuación y que se puede comparar de manera visual en la figura 2.8.

- **Leaky RELU:** Nace para resolver un inconveniente de la activación ReLU. En ReLU cualquier valor negativo es cero, por lo que impide mapear estos valores negativos durante el entrenamiento, pudiendo causar que el gradiente se desvanezca. Para ello, en la versión Leaky, a los valores negativos se les asigna una pequeña activación negativa modelada por la pendiente ay de la activación, que es un parámetro elegido por el usuario. Se pueden observar ambas funciones en 2.8.

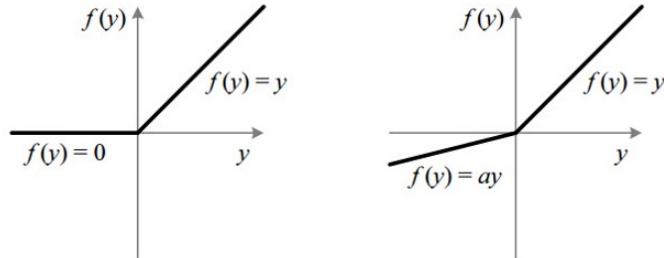


Figura 2.8: Comparación visual de RELU y Leaky RELU obtenida de [5]

- **Sigmoidal:** También conocida como logística, es la función clásica de las redes neuronales. Se utiliza principalmente en problemas de clasificación binaria, ya que su rango de valores está entre 0 y 1. Ha perdido popularidad en los últimos años ya que su alto nivel de saturación hace que en sistemas con muchas capas, como las DNN, los valores a las salidas de las neuronas no cambien significativamente, evitando así que el gradiente cambie y se actualicen el resto de parámetros. Esto se conoce como desvanecimiento del gradiente. Se observa la forma de esta función en la figura 2.9.
- **Tangente Hiperbólica:** Su forma es parecida a la función sigmoidal pero con mayor pendiente. Además, sus límites de saturación son -1 y 1, por lo que mantiene los problemas de desvanecimiento de gradiente de la función sigmoidal. Es comúnmente utilizada en problemas de regresión. Se observa la forma de esta función en la figura 2.10.

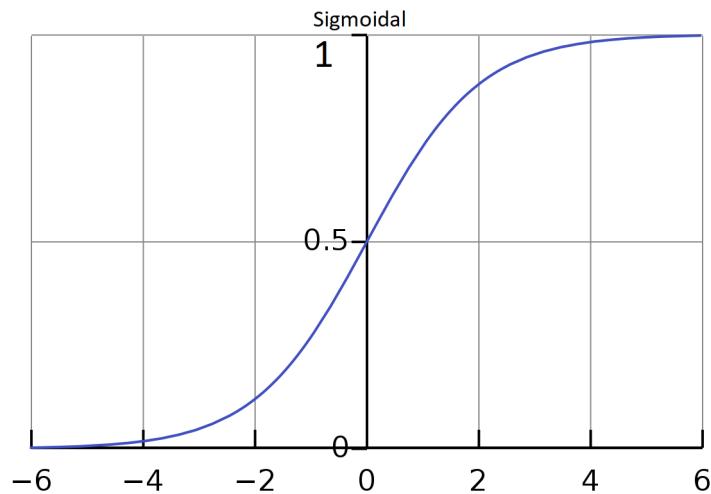


Figura 2.9: Gráfica de la función Sigmoidal .

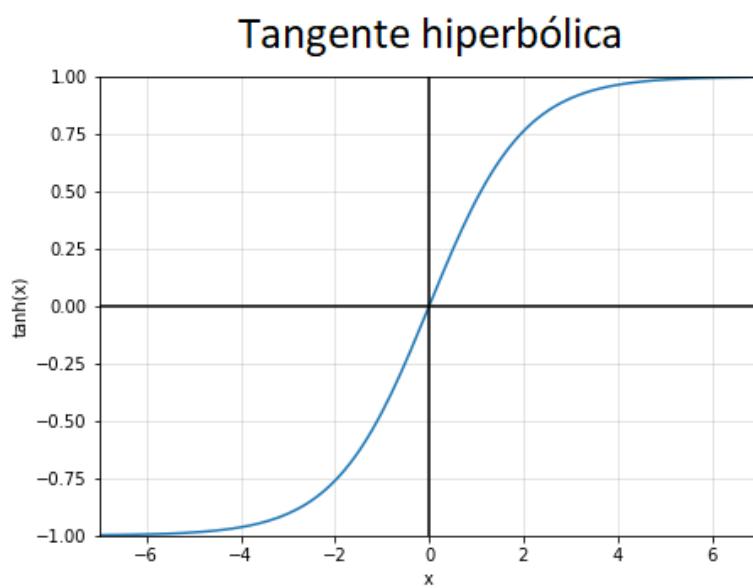


Figura 2.10: Gráfica de la función Tangente Hiperbólica .

2.2.2. Regularización

Se denomina así a una serie de técnicas ampliamente utilizadas que tienen como objetivo favorecer la generalización de los algoritmos y evitar efectos indeseados como el sobre-entrenamiento, mejorando los resultados de la red ante datos que no se hayan procesado antes. En este trabajo se han empleado dos de estas técnicas:

- **Dropout:** esta capa se utiliza para prevenir el sobre-entrenamiento de nuestra red. Basándose en un parámetro establecido previamente, desactiva de forma aleatoria una serie de neuronas y sus conexiones eliminando así la aportación de las mismas únicamente durante el entrenamiento. Los resultados de utilizar esta capa se estudian en el artículo [6] y sus efectos se observan en la figura 2.11.

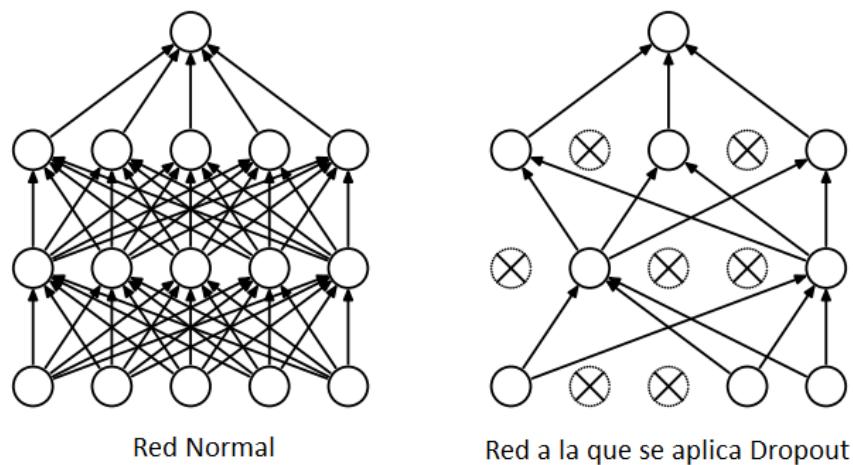


Figura 2.11: Ejemplo de aplicar dropout a una red neuronal. Obtenido del artículo [6]

- **Aumento de datos:** esta técnica es especialmente útil en aquellos problemas en los que el set de datos es reducido. Consiste en generar nuevos datos aplicando diferentes transformaciones a los ya existentes, obteniendo así un mayor conjunto sobre el que entrenar el modelo. En el caso de utilizar imágenes como datos de entrada este tipo de transformaciones pueden consistir en desplazamientos, escalados, giros, cambios de iluminación...etc. De esta manera de una misma imagen se pueden obtener múltiples imágenes diferentes, pero con características similares como se observa en la figura 2.12, aportando así al sistema mayor generalización. La efectividad de esta técnica se demuestra en el siguiente artículo [27].

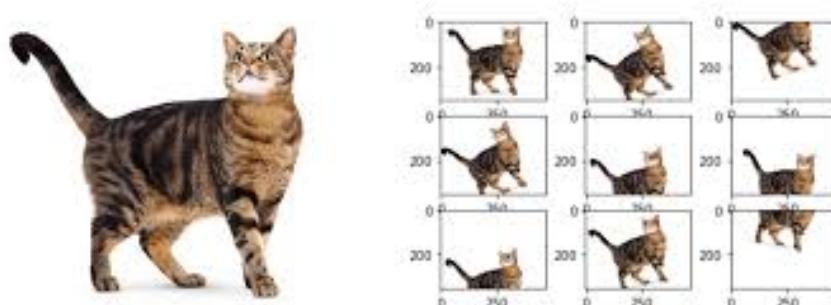


Figura 2.12: Ejemplo de realizar aumento de datos sobre una imagen [7]

2.2.3. Principales Optimizadores

En el marco de las **DNN** se encuentran una gran variedad de optimizadores basados en el algoritmo de descenso por gradiente estocástico. Este algoritmo funciona seleccionando pequeños conjuntos de entrenamiento llamados lotes o batches de datos. Con estos lotes se realizan iteraciones de descenso por gradiente y así minimiza la función de pérdidas del entrenamiento. A partir de SGD [28] se han desarrollado otros optimizadores que modifican de manera dinámica la magnitud de descenso, también llamado "learning rate", durante el entrenamiento. En este apartado se profundiza un poco más en algunos de los optimizadores adaptativos que derivan de SGD:

- **Adagrad:** este algoritmo adaptativo modifica el ratio de aprendizaje basándose en parámetros de la red. Para ello escala los parámetros de manera inversamente proporcional a la raíz cuadrada del sumatorio de los valores de gradiente anteriores [29]. Realiza actualizaciones mayores en aquellos parámetros que menos cambian, mientras los que sufren modificaciones frecuentes reciben ajustes menores. Este algoritmo tiene una excelente base teórica pero llevado a la práctica el ratio de aprendizaje se vuelve cada vez más pequeño por lo que no resulta útil para todos los modelos de aprendizaje profundo. El algoritmo se desarrolla en profundidad en [8] [30].
- **RMSprop:** este optimizador modifica el anterior con el objetivo de eliminar el problema del ratio de aprendizaje que disminuye infinitamente, esto se consigue disminuyendo también con cada actualización la media de los gradientes al cuadrado como se explica de manera más extensa en [31].
- **ADAM:** nace de combinar las ventajas que presentan RMSprop y Adagrad. El nombre surge de la estimación de momentos adaptativa, y es que este algoritmo trata de adaptar los momentos de primer y segundo orden, por lo que solo requiere gradientes de primer orden para determinar el ratio de entrenamiento mediante el primer y segundo momento de estos gradientes. Entre sus principales ventajas están el bajo coste computacional, su fácil implementación y su eficiencia en problemas con multitud de datos. En la figura 2.13 se adjunta el pseudocódigo de este optimizador.

2.2.4. Parámetros de entrenamiento

En esta sección se explican aquellos valores que afectan a la fase del entrenamiento, explicando su función y como pueden variar los resultados.

- **Tamaño de lote o batch:** un batch es un subconjunto de datos de entrenamiento que la red procesa de manera simultánea. Cada vez que un batch es procesado durante el entrenamiento se calcula el error entre las predicciones y los valores esperados usando la función de pérdidas escogida y se minimiza dicha función haciendo uso del optimizador. Determinar un tamaño de batch óptimo para un modelo es algo que se debe hacer empíricamente ya que aunque un valor alto suele resultar en un modelo con poca generalización y uno muy bajo en entrenamientos con mucha divergencia e inestables, no se garantiza que estos rangos no sean óptimos para un modelo determinado.
- **Número de épocas:** se llama época a la iteración completa del modelo sobre todo el conjunto de datos, es decir cuando todas las muestras han pasado por la red y han ajustado sus pesos. Normalmente se asigna un número elevado de épocas ya que en un caso ideal cada época nos acercará más al mínimo de la función de perdidas aumentando la precisión. Cuando el aprendizaje del set de entrenamiento y el de validación comienzan a divergir y se considera que se ha alcanzado el mínimo en la función de pérdidas, a partir de este punto puede producirse sobre-entrenamiento.

Algoritmo ADAM:

```

Ratio de aprendizaje global    $\epsilon$ 
Decay    $\rho$ 
Coeficiente del momento    $\alpha$ 
Parámetro inicial    $\theta$ 
Velocidad inicial    $v$ 

Inicializar acumulación variable    $r = 0$ 
While criterio de parada no alcanzado do:
    Tomar muestra de  $m$  ejemplos del set de entrenamiento    $\{x^{(1)}, \dots, x^{(m)}\}$ 
    con correspondientes objetivos    $\tilde{\theta} y^{(i)} + \alpha v$ .
    Calcular actualización interna:
        Calcular gradiente:    $g \leftarrow \frac{1}{m} \nabla_{\tilde{\theta}} \sum_i L(f(x^{(i)}; \tilde{\theta}), y^{(i)})$ .
        Acumular gradiente:    $r \leftarrow \rho r + (1 - \rho)g \odot g$ .
        Calcular actualización de velocidad:    $v \leftarrow \alpha v - \frac{\epsilon}{\sqrt{r}} \odot g$ .
        Aplicar actualización:    $\theta \leftarrow \theta + v$ .
end while

```

Figura 2.13: Pseudocódigo de optimizador ADAM.[8]

- **Ratio de entrenamiento:** este valor indica el tamaño de descenso en cada iteración hacia el mínimo de la función de perdidas. Un ratio demasiado bajo puede tomar como mínimo absoluto un mínimo local, mientras que uno demasiado alto, puede oscilar entre valores y nunca alcanzar el mínimo absoluto. Se representan estas situaciones en la figura 2.15.

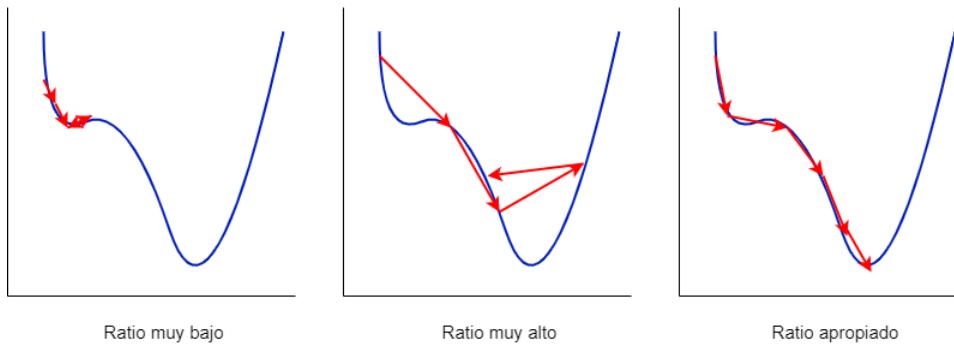


Figura 2.14: Posibles situaciones con ratios de aprendizaje distintos.

- **Decay:** es una función predefinida o definida por el usuario que modela como se reducirá el ratio de aprendizaje tras cada época. Esto permite iniciar con un ratio elevado lo que agiliza el entrenamiento inicial y evita mínimos locales, que gradualmente se reduce evitando así oscilaciones sobre el mínimo objetivo.

2.3. La detección de cáncer

2.3.1. Aspectos básicos sobre el cáncer

El cáncer es una enfermedad que consiste en un crecimiento y reproducción de manera descontrolada de las células de una o varias zonas de cuerpo, creando lo que se denomina tumor canceroso [32]. Esto se puede producir en cualquier zona del cuerpo humano, generando diferentes tipos de cáncer como por ejemplo, de pulmón [33], de colon [34], de seno [35], o el que nos incumbe en este proyecto, el cáncer de próstata [36].

2.3.2. Detección de cáncer clásica

Uno de los principales factores para conseguir la recuperación de un paciente es detectar el cáncer precozmente. Sin embargo, no existe para ello una prueba específica ya que como hemos comentado hay diferentes tipos de cáncer. Por lo tanto, se suelen realizar las pruebas en base a antecedentes familiares o síntomas relacionados. En base a los diferentes tipos de cáncer, se exponen a continuación los tres tipos de pruebas más comunes para su detección.

- **Endoscopia.** Consiste en un delgado tubo con una cámara en el extremo que se introduce en el cuerpo del paciente en la zona que se quiera examinar, esto permite al oncólogo buscar algún cuerpo sospechoso y en caso de hallarlo proceder con otro tipo de pruebas [37]. La prueba lleva el nombre de la zona que se examine, por ejemplo, laringoscopia [38] para examinar la laringe, broncoscopia [39] para los pulmones, colonoscopia [40] para el intestino grueso, etc.
- **Análisis de sangre.** Si algún síntoma indica que el paciente puede sufrir cáncer, se realiza un análisis de sangre para examinar una serie de marcadores [41]. Altos niveles de Alfafetoproteína pueden indicar la existencia de distintos tipos de cáncer como pulmón [42] o de hígado [43], y por ejemplo un alto nivel de Antígeno prostático específico puede denotar que un cáncer de origen desconocido haya comenzado en la próstata [44].
- **Estudio por imágenes.** Para este tipo de estudios se obtiene una imagen de una o varias zonas concretas del cuerpo de un paciente que puedan ser zona de riesgo. Estas imágenes se pueden obtener con multitud de técnicas como radiografías [45], ecografía [46], tomografía [47], RMM [48], etc. En todos estos casos es necesaria la labor de un experto que pueda diferenciar un tejido canceroso de uno normal.
- **Biopsias.** El resto de pruebas comentadas pueden sugerir la presencia de cuerpos o bultos sospechosos que puedan ser un tumor canceroso, pero en muchas ocasiones es necesario extraer una muestra del cuerpo y analizarla para comprobar si realmente se trata de un cáncer. También se puede utilizar este tipo de pruebas sin haber encontrado antes un posible tumor canceroso, para ello se toman varias muestras de distintas zonas con el objetivo de encontrar en alguna de ellas células cancerosas [49, 50].

2.3.3. Detección de cáncer con machine learning

En este apartado vamos a explicar la evolución del uso del Machine Learning (ML) en el diagnóstico del cáncer, comenzando por los métodos más clásicos hasta llegar a aquellos más modernos. Los métodos previos a la aparición de las redes neuronales profundas ofrecían una guía para el personal experto, como

puede ser el caso de los árboles de decisión para recetar tratamientos [9]. También ganaron relevancia sistemas de cribado usando análisis de regresión utilizando diferentes marcadores y condiciones sociales.

Con el aumento de capacidad de cómputo y la creación de nuevas pruebas médicas como los estudios por imágenes, surgen nuevos modelos basados en aprendizaje profundo que se apoyan en gran cantidad de información de diferentes formatos y que llegan a alcanzar la misma precisión que los expertos en la detección de algunos tipos de cáncer. A continuación se explica en profundidad cada uno de los métodos expuestos anteriormente.

- **Árboles de decisión:** los árboles de decisión son un sistema de clasificación en el que se parte de un nodo inicial y se realizan divisiones binarias hasta llegar a las hojas del árbol. Estas divisiones se realizan en función de marcadores y normalmente se establecen utilizando redes neuronales. El uso de estos árboles permite crear una ruta de decisión de manera sencilla e intuitiva que puede llegar a facilitar el diagnóstico de algunos tipos de cáncer basándose en datos de entrada obtenidos de pruebas previas como en [51] [9].

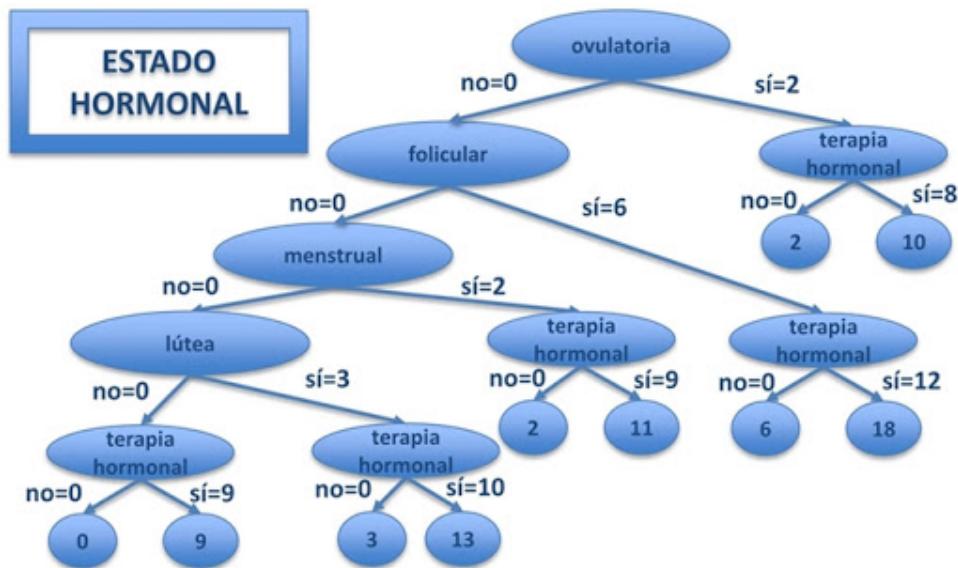


Figura 2.15: Ejemplo de uso de un árbol de decisión basado en el estado hormonal de mujeres obtenido de [9].

- **Análisis de regresión** estos modelos buscan determinar la relación de una variable respecto a otras. En este caso se utilizan datos del entorno de los pacientes, de su estilo de vida, historial médico, etc. para intentar determinar la existencia de cáncer [52] o hacer un cribado inicial [53].
- **Deep learning o aprendizaje profundo:** esta técnica es una de las que más se está investigando y que está obteniendo los mejores resultados entre las técnicas modernas [54]. Se utilizan DNN que se entrena utilizando bases de datos con imágenes de pacientes, resultados de análisis o ambos datos en conjunto. Estas redes logran un alto porcentaje de éxito en el diagnóstico del cáncer de mama, llegando a alcanzar la misma precisión que un radiólogo experto como se concluye en [16]. También se investiga su uso para determinar el estado de este tipo de cáncer [55,56]. El diagnóstico de cáncer de pulmón con estas técnicas también está siendo desarrollado con éxito, como se ve en [14,15]. Por último, cabe destacar, entre otros, los avances en el diagnóstico del cáncer de próstata donde ya se presentan algunos modelos de ayuda al diagnóstico temprano utilizando imágenes de RMM y marcadores de antígeno [57], y que ya se empiezan a reconocer por asociaciones médicas como un avance en el diagnóstico [58].

2.3.4. El cáncer de próstata

El cáncer de próstata es el tercer tumor más frecuente en varones españoles y la tercera causa de muerte por cáncer en hombres [59]. Los valores de mortalidad han disminuido en los últimos años debido a una mejora del diagnóstico. Aun así, se siguen buscando nuevos métodos que faciliten la identificación de este cáncer en estados más prematuros. Los principales métodos son:

- **Biopsia transrectal guiada por ultrasonidos:** se trata de una prueba invasiva y con alto coste la cual se realiza una vez que se tienen indicios de la existencia de una anomalía para comprobar si se trata de un cáncer. En esta prueba se toman una o varias muestras del tejido de la próstata usando una guía por ultrasonidos en tiempo real para incrementar la precisión. Se trata de una prueba con múltiples efectos secundarios [60] para la que se buscan alternativas a día de hoy.
- **Ecografía transrectal:** se utiliza la imagen por ecografía para tratar de observar alguna lesión en la zona de la próstata. Esto requiere de un experto capaz de distinguir fácilmente este tipo de lesiones en las imágenes de ecografía. Es por ello que esta prueba ha perdido valor de diagnóstico en los últimos años y ha sido relegada a la determinación del volumen prostático y como guía para realizar la biopsia [61].
- **Tacto rectal:** Al igual que la prueba anterior, últimamente ha sido puesta en entredicho debido a su amplio rango de valor predictivo, que se sitúa entre un 5 y un 83 %. Aun así sigue siendo una prueba recomendada debido a la falta de otros métodos más precisos, pero siempre junto a otras pruebas como la medición de Prostate Specific Antigen (PSA) [62].
- **PSA o antígeno prostático específico:** En esta prueba se busca un valor anormal en la sangre de esta sustancia producida por la próstata, comprendiéndose un valor normal menos de 4ng/ml. En los últimos años se ha observado a paciente con niveles superiores a los 4ng/ml que no presentan ninguna lesión en la próstata. también se han identificado factores externos que pueden variar la producción de esta sustancia como son la prostatitis, infección en las vías urinarias o incluso las biopsias o cirugías de la próstata. Es por ello que aunque esta prueba sea la más eficiente, con una sensibilidad de en torno un 80 % tampoco se postula como una prueba completamente fiable[62] [63].
- **Basados en Inteligencia artificial junto a mapas de resonancia magnética:** como se ha comentado anteriormente 2.3, están surgiendo nuevos métodos que tratan de ayudar al diagnóstico de diferentes tipos de cáncer utilizando diferentes sistemas basados en inteligencia artificial. Ya en 2005 se hacía referencia a estos sistemas como el futuro de la detección de cáncer, pero se encontraban en una fase muy temprana de desarrollo [62]. Con el paso del tiempo y el avance en la capacidad de computación, junto a la mejora de algoritmos y técnicas, comienzan a aparecer investigaciones que han obtenido resultados que llegan a obtener una tasa de éxito en el diagnóstico que radiólogos experimentados [64]. Además del uso de DNN sobre los mapas de RMM, también se estudia el uso de DNN que ayuden a determinar el resultado de las biopsias transrectales [65].

Capítulo 3

Desarrollo

3.1. Introducción

El objetivo de este **TFG** es la implementación de un sistema de ayuda al diagnóstico del cáncer de próstata que tomará como entrada diferentes imágenes de **RMM** del volumen de la próstata. Se pretende que el sistema sea capaz de identificar los posibles tumores, ubicarlos e indicar su naturaleza ya sea benigna o cancerígena. Para ilustrar el sistema desarrollado se presenta de nuevo la figura 3.1 expuesta en la introducción del trabajo. Las principales etapas que se han seguido para el desarrollo del proyecto

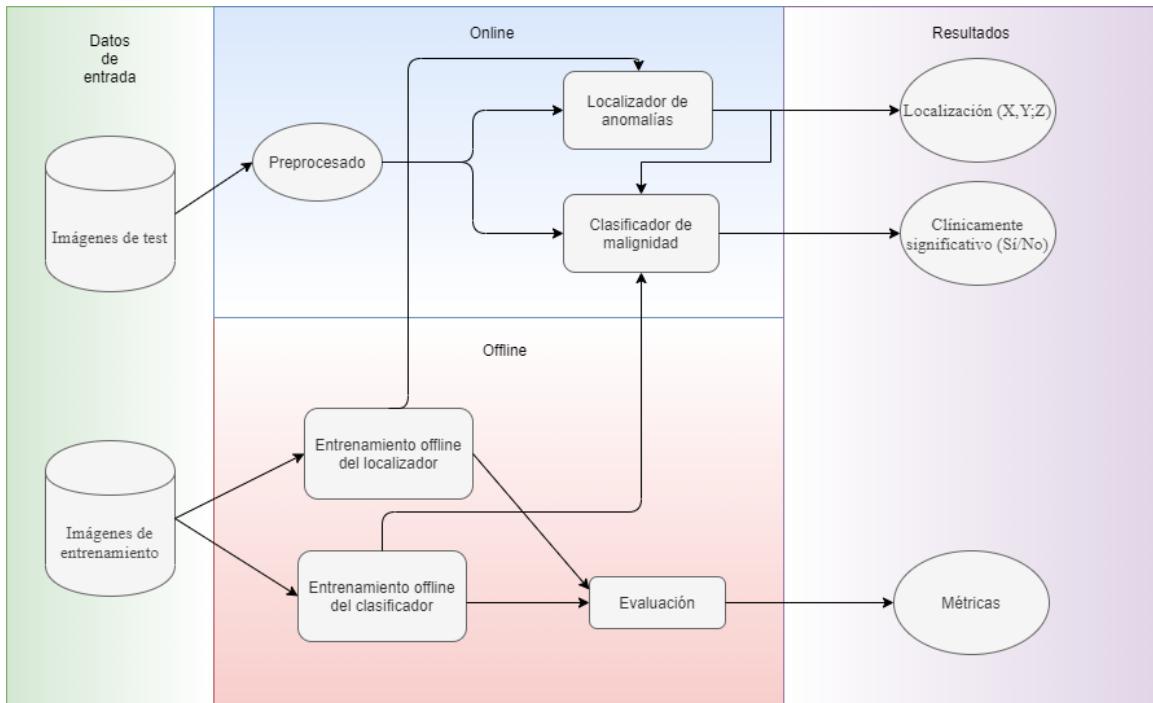


Figura 3.1: Diagrama de bloques completo del sistema propuesto.

son:

1. **Preprocesado y selección de imágenes:** antes de comenzar las arquitecturas se seleccionaron las modalidades de imágenes con las que se va a trabajar y que servirán de entrada a cada tipo de red específica. También es necesario definir un preprocesado para normalizar las imágenes, realizando la resta de la media de sus valores y dividiendo por su desviación típica.

2. **Definición de las arquitecturas:** ha sido necesario definir las diferentes arquitecturas que se han construido para procesar las diferentes modalidades de imágenes de manera unimodal o multimodal. También se ha definido la red de localización que ubica las posibles lesiones.
3. **Entrenamiento de modelos:** una vez definidas las diferentes arquitecturas se han realizado los respectivos entrenamientos utilizando la base de datos Prostatex [3] y ajustando los parámetros para cada una de las arquitecturas.

3.2. Definición del problema

Aquí se explica brevemente el problema que se pretende resolver con este [TFG](#) así como su interés científico y cómo resolverlo.

1. **Prueba Invasivas:** Actualmente el método con mayor efectividad en el diagnóstico es la biopsia transrectal, una prueba invasiva, dolorosa y con efectos secundarios. Además, esta prueba requiere de unos análisis posteriores en lo que se pueden obtener falsos negativos.
2. **Necesidad de experto:** Para valorar los resultados previos obtenidos mediante estudios por imágenes como endoscopias, radiografías, ecografías o resonancias siempre es necesaria la labor de un radiólogo experto que sea capaz de identificar un posible tumor o malformación en la próstata. Este estudio es lento y tiene un alto coste asociado.

Para resolver los problemas anteriores se deben lograr dos principales objetivos planteados en este [TFG](#):

1. **Localizador de anomalías:** se desarrolla un sistema capaz de posicionar en el volumen ocupado por la próstata aquellos cuerpos extraños clasificados anteriormente como sospechosos.
2. **Clasificación de anomalías:** se debe desarrollar un sistema capaz de clasificar con la mayor exactitud las imágenes de [RMM](#) en dos grupos, según presenten o no cáncer de próstata.

El sistema planteado busca facilitar la tarea de los radiólogos en los estudios por imágenes mejorando la tasas de detección de este tipo de pruebas y en el mejor de los casos convirtiéndola en la principal prueba para diagnosticar este tipo de cáncer, dejando atrás aquellas más invasivas.

3.3. Arquitectura propuesta

Para resolver el problema comentado en el apartado anterior se han propuesto un total de siete arquitecturas diferentes de clasificación y una de localización. Entre las creadas para la clasificación se diseñan cinco redes llamadas unimodales puesto que reciben como entrada una única modalidad de imagen, y dos redes multimodales que reciben las cinco modalidades a la vez. Con estos diseños se pretende desacoplar el problema de manera que si un paciente no presenta todos los tipos de imágenes también podrá ser evaluado.

3.3.1. Arquitectura de localización de anomalías

Para la tarea de la localización de anomalías se ha propuesto la creación de una [DNN](#) convolucional 3D especializada en información volumétrica médica, ya que empleamos un espacio tridimensional en

el cual se han obtenido diferentes planos de corte del volumen prostático. La arquitectura empleada se ha basado en la V-net [66], que es una conocida red de información volumétrica empleada en imagen médica. Esta arquitectura ha tenido que ser adaptada a nuestro problema y ha sido empleada sobre los mapas de entrada en modalidad Ktrans, dado que estos muestran las características más representativas a la hora de detectar los tumores [67]. Esto también es causa de la imposibilidad de registrar todos los mapas en una representación común y del mismo tamaño, debido al uso de diferente maquinaria para obtener cada una de las modalidades de mapas médicos. Las principales modificaciones llevadas a cabo tienen que ver con el tensor de entrada y salida empleados y las activaciones de la misma. El tensor de entrada y el tensor de salida tendrán un tamaño de (128,128,16,1) y las activaciones empleadas serán Leaky ReLU en casos generales, dado que las activaciones originales PreLU pueden aumentar el riesgo de sobre-entrenamiento en bases de datos pequeñas, como es nuestro caso. Mientras tanto la capa de salida empleará una activación sigmoidal, ya que este es un caso de localización binaria del tumor, cuya localización se plasmará como una segmentación del mismo en los mapas. En la figura 3.3 observamos una representación gráfica de la arquitectura para facilitar su reproducibilidad.

Las entradas y salidas del localizador estarán constituidas, en el primer caso por el volumen tridimensional de la modalidad Ktrans y en el segundo con un volumen tridimensional que contiene una esfera de valor 1.0 con radio 10 px alrededor de la anomalía etiquetada en la base de datos empleada. Un ejemplo de estos volúmenes se muestra en la figura 3.2. Dado que los volúmenes de la base de datos empleada, solo tienen etiquetado su centroide aproximado (x,y,z) a lo largo del volumen, se ha procesado la base de datos para que a raíz de este centroide se generen esferas tridimensionales, que permitan modelar un campo de acción y localización aproximada de estas anomalías. El radio de las esferas etiquetadas se ha fijado experimentalmente en 10 px. El hecho de emplear estas esferas tridimensionales, permite modelar a través de la totalidad del volumen, la lesión etiquetada, y además da facilidades al localizador a la hora de buscar la lesión, mientras que en caso contrario debería buscar un único punto exacto.

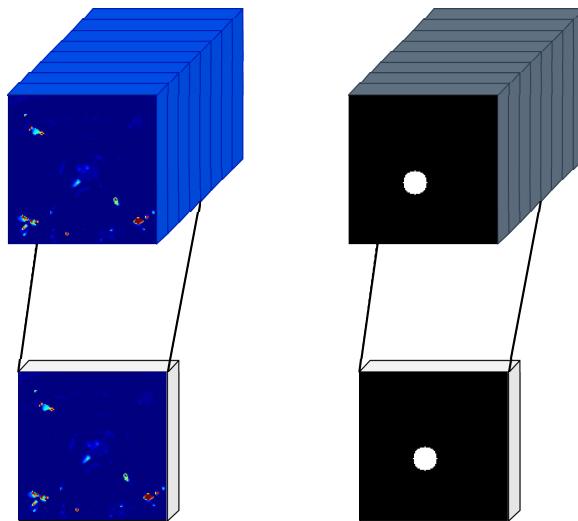


Figura 3.2: Ejemplos de entrada y salida del localizador. A la izquierda se muestra el volumen de entrada de la modalidad Ktrans y una de sus secciones bidimensionales, mientras que a la derecha se muestra el volumen de salida con la localización de la anomalía plasmada por una esfera blanca a lo largo del volumen, en la sección esta esfera se muestra en forma de círculo.

3.3.1.1. Arquitecturas unimodales de clasificación

Para poder abordar la clasificación unimodal de anomalías, se ha propuesto una arquitectura ligera, que permita procesar volúmenes tridimensionales a múltiples escalas dentro del propio volumen y

Número de capa	Tipo	Tamaño Output	Kernels/Activación
1	Entrada	(128,128,16,1)	—
2	Convolución 3D	(128,128,16,16)	(5,5,5)
3	Activación	(128,128,16,16)	Leaky Relu
4	Concatenación 3D	(128,128,16,16)	(2,2,2)
5	Convolución 3D	(128,128,16,16)	(5,5,5)
6	Activación	(128,128,16,16)	Leaky Relu
7	Convolución 3D	(64,64,8,32)	(5,5,5)
8	Activación	(64,64,8,32)	Leaky Relu
9	Convolución 3D	(32,32,4,64)	(5,5,5)
10	Activación	(32,32,4,64)	Leaky Relu
11	Convolución 3D	(16,16,2,128)	(5,5,5)
12	Activación	(16,16,2,128)	Leaky Relu
13	Convolución 3D	(8,8,1,256)	(5,5,5)
14	Activación	(8,8,1,256)	Leaky Relu
15	Convolución 3D	(8,8,1,256)	(5,5,5)
16	Activación	(8,8,1,256)	Leaky Relu
17	Convolución 3D	(8,8,1,256)	(5,5,5)
18	Activación	(8,8,1,256)	Leaky Relu
19	Convolución 3D	(16,16,2,128)	(5,5,5)
20	Activación	(16,16,2,128)	Leaky Relu
21	Convolución 3D	(32,32,4,64)	(5,5,5)
22	Activación	(32,32,4,64)	Leaky Relu
23	Convolución 3D	(64,64,8,32)	(5,5,5)
24	Activación	(64,64,8,32)	Leaky Relu
23	Convolución 3D	(128,128,16,16)	(5,5,5)
24	Activación	(128,128,16,16)	Leaky Relu
25	Concatenación 3D	(128,128,16,32)	(2,2,2)
26	Convolución 3D	(128,128,16,32)	(5,5,5)
27	Activación	(128,128,16,32)	Leaky Relu
26	Convolución 3D	(128,128,16,1)	(1,1,1)
27	Activación	(128,128,16,1)	Leaky Relu
28	Convolución 3D	(128,128,16,1)	(1,1,1)
29	Activación	(128,128,16,1)	Sigmoidal
Número de parámetros		280,483	

Tabla 3.1: Arquitectura de red de localización.

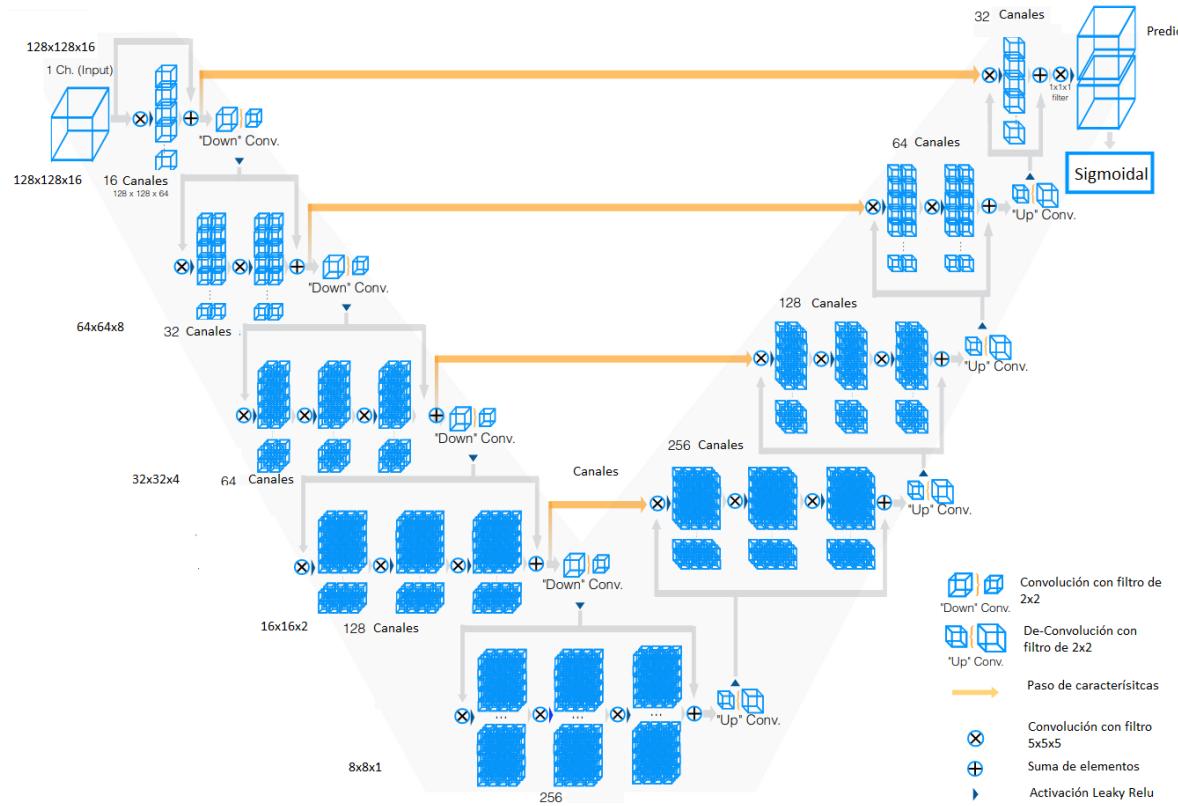


Figura 3.3: .

obteniendo buenos rendimientos al mismo tiempo. Pese a ser un modelo específico, está basado en las redes Inception-V3 [68][69] y VGG-16 [70] como se comenta más adelante. Usualmente en este tipo de arquitecturas, se opta por el uso funciones de activación de tipo ReLU [71], debido a sus características. Sin embargo, para evitar el problema del desvanecimiento del gradiente que estas traen consigo frente a activaciones negativas, se opta por usar la variante LeakyReLU, como se ha comentado anteriormente en el apartado 2.2.1.

Esta arquitectura ha sido específicamente creada para la resolución del problema planteado en este trabajo, pero a su vez, se busca que la misma sea lo suficientemente genérica como para poder ser replicada y usada con cada uno de los tipos de volúmenes empleados en este estudio, es decir, poder usar Ktrans, ADC, y volúmenes de modalidad T2(sagital, coronal y transversal).

Como entradas a esta arquitectura se utilizan volúmenes de cada una de las modalidades. Los volúmenes empleados no cubren la totalidad de las dimensiones del volumen original, es decir son parches, seleccionados alrededor de la localización de la anomalía, la cual viene provista por la arquitectura de localización. De forma empírica se ha determinado que el tamaño de parche tridimensional óptimo es de 40x40x3 alrededor de la anomalía detectada. De tal manera que si por ejemplo en el caso de la modalidad Ktrans, el tensor de entrada tiene unas dimensiones de 128x128x16, el parche que empleará la arquitectura unimodal tiene un tamaño de 40x40x3 alrededor del centroide de la anomalía detectada por el localizador propuesto en este trabajo. Mientras tanto, las salidas de esta arquitectura están compuestas por un único valor que oscila entre 0 y 1, siendo un problema de clasificación binaria en el que la clase 0 es una detección no significativa y la clase 1 una detección significativa.

A continuación se enumeran y explican cada una de las fases de estas arquitecturas en el orden en el que se aplican.

- **Etapa multiescala:** Se aplica a los datos de entrada una serie de convoluciones 3D multiescala dado que los tumores pueden tener diferentes tamaños o estar ubicados en distintas posiciones. Esto nos va a permitir detectar dichas anomalías sin importar lugar o ubicación. Esta técnica se toma de las redes tipo Inception-V3 ya que se demuestra en varios artículos como estos bancos de filtros multiescala proporcionan buenos resultados en problemas de clasificación. [68, 69]
- **Etapa de procesado:** esta fase se inicia con una capa de MaxPooling, lo que permite obtener las características de mayor energía. Posteriormente se usan bancos de filtros convolucionales que permitirán obtener esas características más complejas que se utilizan para clasificar las anomalías. este diseño basado en filtros se inspira en redes como la VGG-16 [70]. Tras esto se unifican todas las dimensiones utilizando una capa de flatten o de vectorización para convertir las múltiples dimensiones a una sola.
- **Etapa de clasificación:** se utilizan capas densas para realizar la clasificación. Empíricamente se ha llegado a que el uso de dos capas densas de diez neuronas aporta el mejor resultado para estas pruebas. La última capa del clasificador cuenta con una única neurona con activación sigmoidal puesto que se trata de un problema de clasificación binaria.

A continuación se muestra un diagrama 3.4 de este tipo de arquitectura, así como una tabla 3.2, en ellos se ve claramente la división de la etapa multiescala.

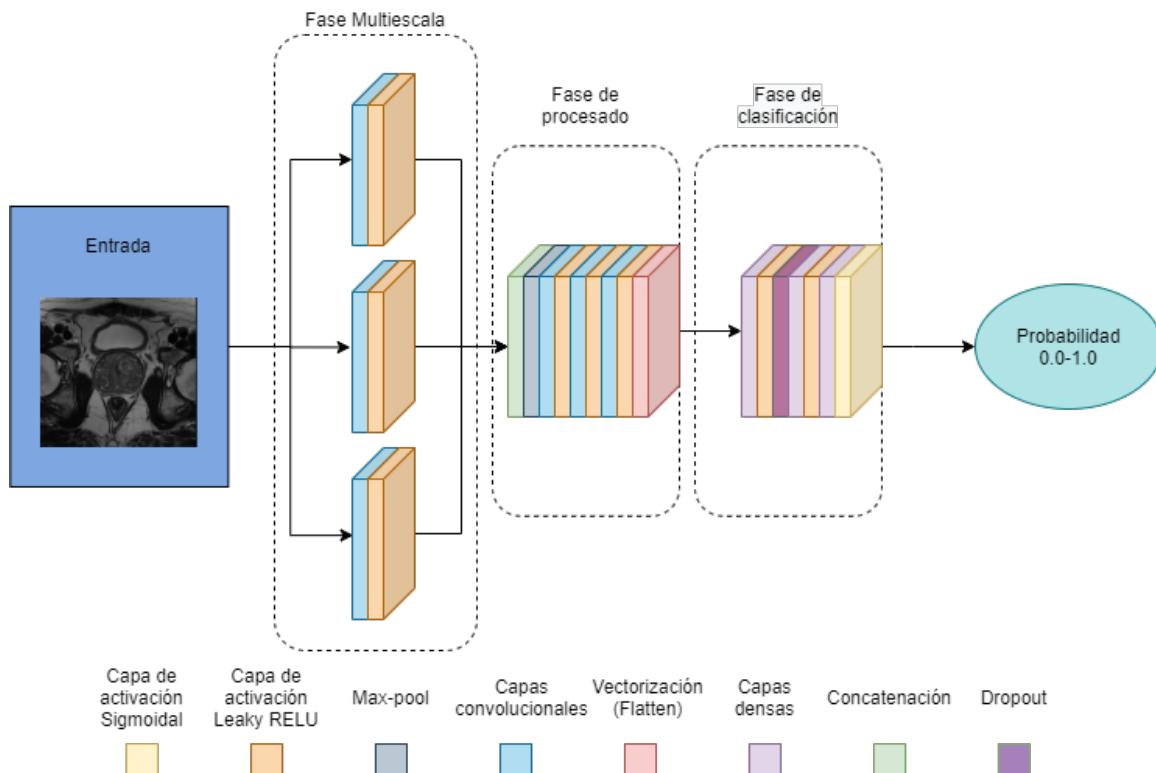


Figura 3.4: Diagrama de arquitectura genérica unimodal.

3.3.2. Arquitecturas multimodales de clasificación

En este apartado se plantean las arquitecturas de fusión que tratarán de usar todos los volúmenes médicos de un paciente en lugar de una sola modalidad de los mismos, tratando de fusionar información de diferente naturaleza. A continuación se explican las dos principales arquitecturas propuestas.

Número de capa	Tipo	Tamaño Output	Kernels/Activación
1	Entrada	(3,40,40,1)	-
2a	Convolución 3D	(3,40,40,16)	(1,7,7)
2b	Convolución 3D	(3,40,40,16)	(1,5,5)
2c	Convolución 3D	(3,40,40,32)	(1,3,3)
3a	Activación	(3,40,40,16)	Leaky Relu
3b	Activación	(3,40,40,16)	Leaky Relu
3c	Activación	(3,40,40,32)	Leaky Relu
4	Concatenación 3D	(1,20,20,65)	(2,2,2)
5	Max Pooling 3D	(1,20,20,66)	(2,2,2)
6	Convolución 3D	(1,20,20,64)	(2,3,3)
7	Activación	(1,20,20,64)	Leaky Relu
8	Convolución 3D	(1,20,20,64)	(2,3,3)
9	Activación	(1,20,20,64)	Leaky Relu
10	Convolución 3D	(1,20,20,32)	(2,3,3)
11	Activación	(1,20,20,32)	Leaky Relu
12	Vectorización	(12800)	-
13	Densa	(10)	Leaky Relu
14	Dropout	(10)	-
15	Densa	(10)	Leaky Relu
16	Densa	(1)	Sigmoidal
Numero de parámetros		280,483	

Tabla 3.2: Arquitectura unimodal.

3.3.2.1. Arquitectura de fusión por tensor de entrada

Esta arquitectura parte de la premisa de que el modelo obtendrá mejores resultados pudiendo entrenar con las diferentes modalidades codificadas como canales del volumen tridimensional, al igual que se haría con los mapas bidimensionales de una imagen de color (RGB) y por lo tanto, diferentes perspectivas de una misma lesión. Es por ello que en este modelo se utilizan como entrada las cinco imágenes a la vez de un mismo paciente, para ello se necesita seleccionar previamente aquellos pacientes con las cinco imágenes disponibles lo que reduce el set de entrenamiento. Una vez seleccionadas, la arquitectura de la red neuronal es igual que en la estructura unimodal de clasificación anterior, pero la entrada consta de 5 imágenes cada vez, en lugar de una única imagen esta diferencia se observa en la entrada del diagrama 3.5. En la tabla 3.3 de arquitectura se observa la variación con la arquitectura unimodal, en los canales de entrada que se deben aumentar de 1 a 5.

3.3.2.2. Arquitectura de fusión de etapas unimodales

Este modelo es el más complejo del proyecto y se pretende utilizar todo el conocimiento obtenido anteriormente para generar un modelo más preciso. Esta arquitectura está basada en los ensambladores de clasificadores [72], puesto que se utilizan clasificadores de menor complejidad de manera conjunta para obtener un clasificador robusto. Para ello se utilizan los 5 tipos de imágenes, pero a diferencia de en la arquitectura de fusión por tensor de entrada, ahora cada tipo de imagen se evalúa en su arquitectura unimodal específica, que utiliza los pesos ya obtenidos anteriormente para garantizar el mejor resultado. La salida de estas 5 redes se evalúa en dos capas convolucionales, la primera de cinco neuronas con activación Leaky-RELU y la última de una neurona con activación Sigmoidal para lograr la clasificación binaria.

El siguiente diagrama 3.6 ayuda a comprender la arquitectura ya que se observa como las imágenes de la entrada se distribuyen a cada una de las arquitecturas unimodales replicadas y entrenadas para

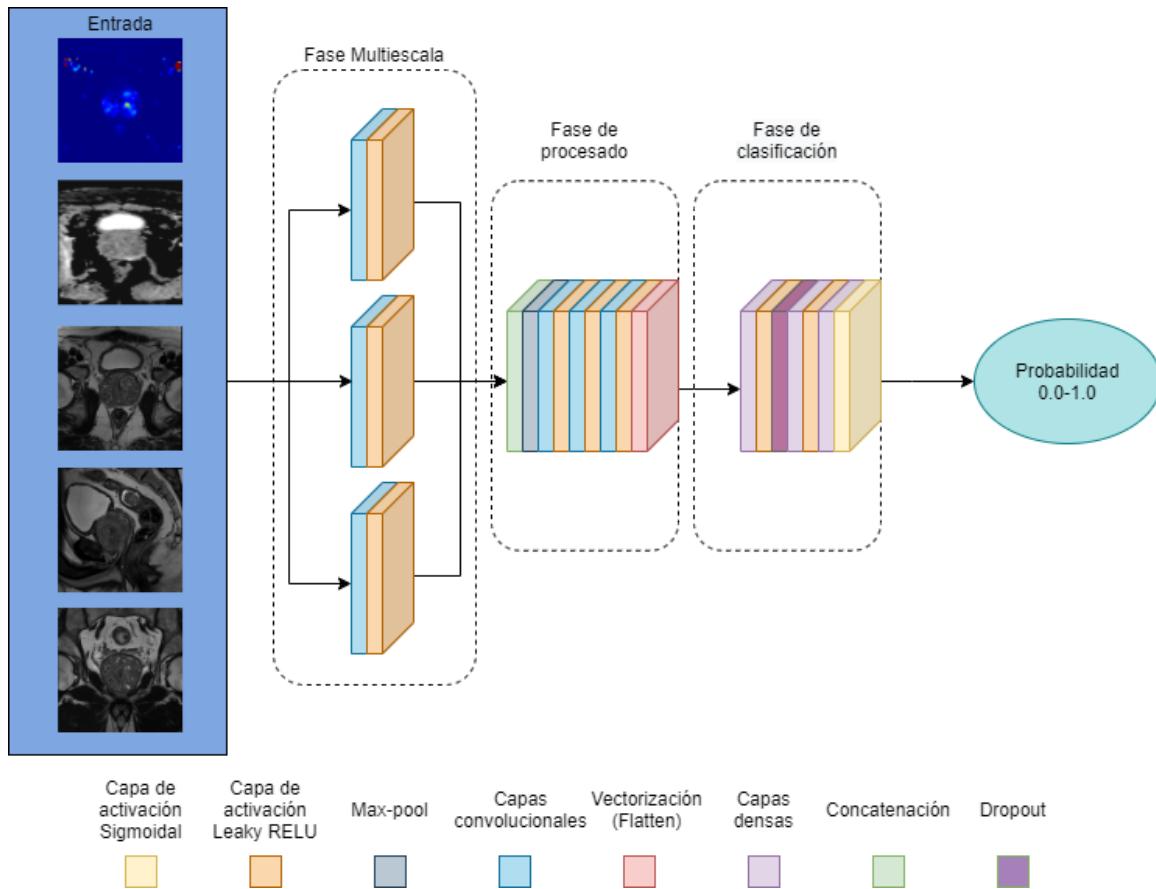


Figura 3.5: Diagrama de arquitectura de fusión 1. Se observan las 5 modalidades como entrada.

Número de capa	Tipo	Tamaño Output	Kernels/Activación
1	Entrada	(3,40,40,5)	-
2a	Convolución 3D	(3,40,40,16)	(1,7,7)
2b	Convolución 3D	(3,40,40,16)	(1,5,5)
2c	Convolución 3D	(3,40,40,32)	(1,3,3)
3a	Activación	(3,40,40,16)	Leaky Relu
3b	Activación	(3,40,40,16)	Leaky Relu
3c	Activación	(3,40,40,32)	Leaky Relu
4	Concatenación 3D	(1,20,20,65)	(2,2,2)
5	Max Pooling 3D	(1,20,20,66)	(2,2,2)
6	Convolución 3D	(1,20,20,64)	(2,3,3)
7	Activación	(1,20,20,64)	Leaky Relu
8	Convolución 3D	(1,20,20,64)	(2,3,3)
9	Activación	(1,20,20,64)	Leaky Relu
10	Convolución 3D	(1,20,20,32)	(2,3,3)
11	Activación	(1,20,20,32)	Leaky Relu
12	Vectorización	(12800)	-
13	Densa	(10)	Leaky Relu
14	Dropout	(10)	-
15	Densa	(10)	Leaky Relu
16	Densa	(1)	Sigmoidal
Number of parameters		290,979	

Tabla 3.3: Arquitectura de fusión 1.

esa modalidad específica. En la tabla 3.4 de esta arquitectura se contraen las filas de las arquitecturas unimodal (filas 2 a 16) y se añade un campo más a la última columna para indicar la modalidad de cada red.

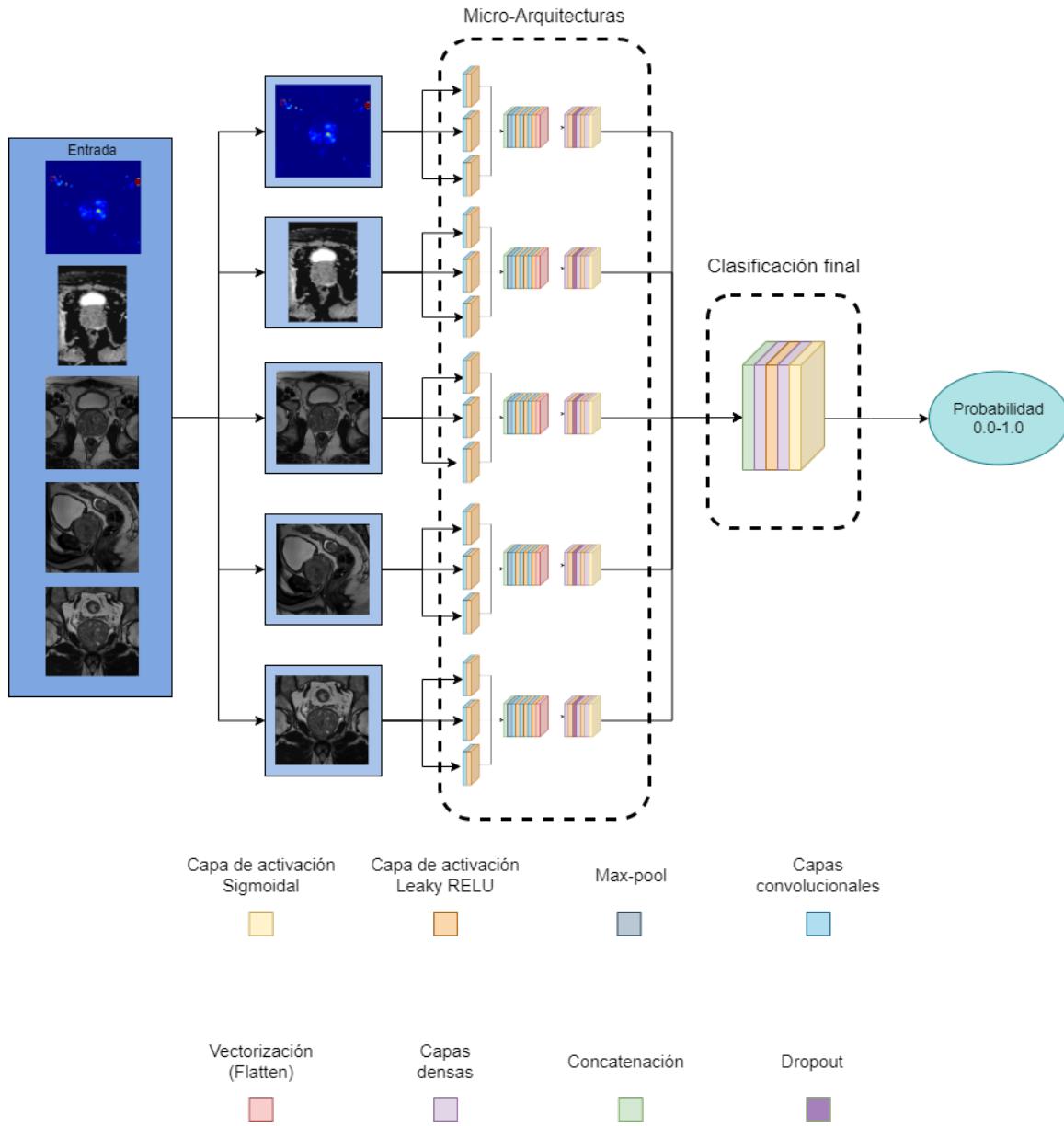


Figura 3.6: Diagrama de arquitectura de fusión 2.

3.4. Condiciones de entrenamiento

En este apartado se exponen los parámetros escogidos para el correcto entrenamiento de las redes explicadas anteriormente.

1. **Optimizador:** para este proyecto se ha optado por el optimizador adaptativo ADAM. Las causas de esta elección subyacen en que ADAM introduce el concepto de adaptación del learning rate intra-época, lo que permite que cada época parte de un learning rate inicial y a través del cálculo de los primeros y segundos momentos de los gradientes modifique este a voluntad. Esto sumado al decay,

Número de capa	Tipo	Tamaño Output	Kernels/Activación/Tipo imagen
1	Entrada	(3,40,40,5)	—
(2-16)a	Unimodal	-	ADC
(2-16)b	Unimodal	-	T2-cor
(2-16)c	Unimodal	-	T2-sag
(2-16)d	Unimodal	-	T2-tra
(2-16)e	Unimodal	-	Ktrans
17	Concatenación 3D	(1,20,20,65)	(2,2,2)
18	Densa	(5)	Leaky Relu
19	Densa	(1)	Sigmoidal
Number of parameters		280,483	

Tabla 3.4: Arquitectura de fusión 2.

permite hacer de ADAM un excelente candidato a la hora de realizar la optimización. Siempre hay que tener cuidado con los tamaños de batch y learning rate utilizados, ya que los optimizadores adaptativos especialmente pueden diverger fuertemente ante malos tuneados de parámetros.

2. **Función de perdidas:** puesto que el problema a resolver se trata de un problema de clasificación binaria se ha optado por utilizar la función de entropía cruzada binaria que resulta idónea para este tipo de problemas como se explica en el apartado 4.3.1. En el caso de la red de localización, al tratarse como un problema de segmentación binaria se utiliza esta misma función.

Es importante destacar que para que esta función sea efectiva se debe contar con un número de casos positivo y negativos equilibrados. Esto no se produce en la base de datos utilizada que presenta una mayoría de casos negativos. Para compensar este desequilibrio se fuerza el equilibrio en la función generadora del entrenamiento, entrenando en cada iteración con un caso positivo y uno negativo.

$$\text{Loss} = -\frac{1}{n} \sum_{\text{size}}^n y_i \cdot \log \hat{y}_i + (1 - y_i) \cdot \log (1 - \hat{y}_i) \quad (3.1)$$

3. **Tamaño de batch:** dado el tamaño reducido de la base de datos y menor aún en las arquitecturas multimodales se utiliza un tamaño de batch (o lote) de 2 imágenes. Este es un número muy bajo, lo que puede derivar en una dirección de descenso de gradiente muy ruidosa. Por esa razón también se usa un ratio de aprendizaje bajo, que permita mejorar la convergencia.
4. **Número de épocas:** el número de épocas utilizado es 30 aunque durante el entrenamiento se guardan aquellos pesos que obtengan un mejor resultado en validación evitando así obtener un modelo sobre-entrenado.
5. **Ratio de aprendizaje:** se establece un ratio de $5e^{-6}$, un valor pequeño por la razón que se ha comentado el escoger el tamaño de batch.

3.4.1. Técnicas de aumento de datos

En este apartado se definen las técnicas utilizadas durante el entrenamiento para aumentar la cantidad de datos. Con esto se consigue disponer de set de entrenamiento mucho mayor aumentando así la generalización del sistema. Dado que usamos volúmenes 3D, la transformación empleada se usa en todas las capas de los volúmenes tridimensionales para darle coherencia espacial. Se definen a continuación cada una de las técnicas aplicadas a las diferentes capas de los volúmenes de la base de

- **Rotación:** se aplican pequeños giros a las imágenes lo que permite simular diferentes posiciones del paciente durante la realización de la prueba. Para este tipo de aumento en este TFG se utiliza un rango entre 0 y 5 grados. En la figura 3.7 se muestra un ejemplo de esta técnica.

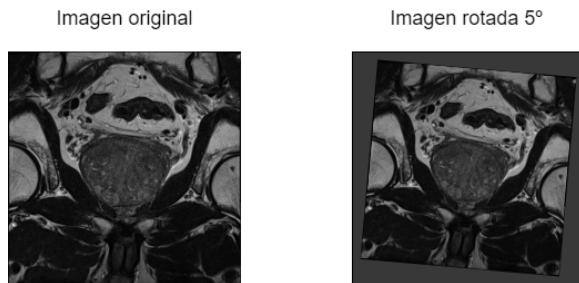


Figura 3.7: Ejemplo de aplicar técnica de rotación.

- **Espejado vertical:** aumenta las muestras espejando la imagen verticalmente generando una nueva perspectiva. En la figura 3.8 se muestra un ejemplo de esta técnica.

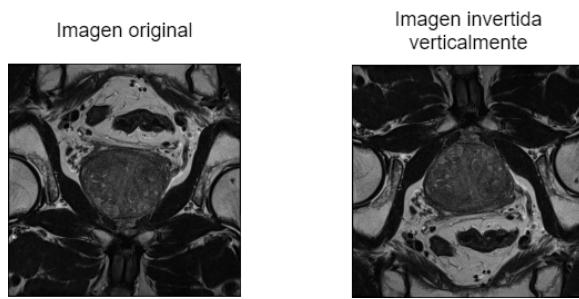


Figura 3.8: Ejemplo de aplicar técnica de espejar verticalmente.

- **Espejado horizontal:** aumenta las muestras espejando la imagen horizontalmente generando una nueva perspectiva. En la figura 3.9 se muestra un ejemplo de esta técnica.

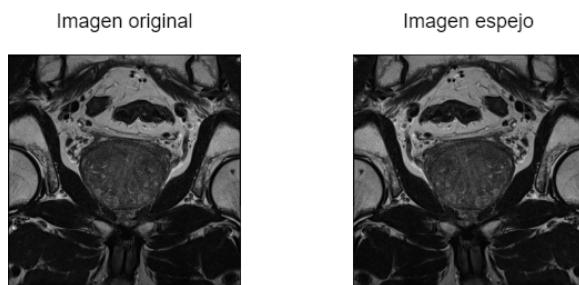


Figura 3.9: Ejemplo de aplicar técnica de espejado horizontal.

- **Desplazamiento:** se desplaza la imagen en una o dos direcciones para lograr así muestras en la que la posible anomalía parezca en diferentes coordenadas. En la figura 3.10 se muestra un ejemplo de esta técnica.

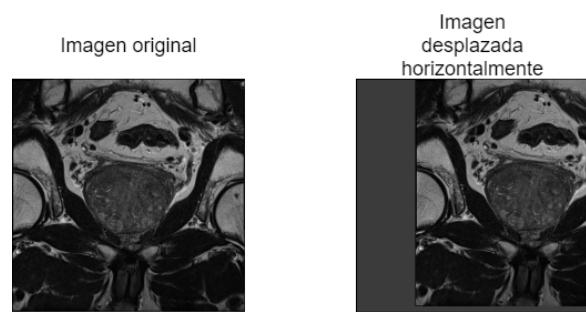


Figura 3.10: Ejemplo de aplicar técnica de desplazamiento.

Capítulo 4

Resultados

4.1. Introducción

En este apartado se muestran los resultados obtenidos con las arquitecturas expuestas anteriormente. Primero se describe en profundidad la base de datos utilizada, posteriormente se explican las métricas que se utilizan para evaluar los sistemas y por último se muestran y analizan los resultados experimentales obtenidos

4.2. Bases de datos

4.2.1. Características

La base de datos Prostatax [3] utilizada en este proyecto es parte de un desafío internacional que busca facilitar el diagnóstico del cáncer de próstata mediante el análisis de este tipo de imágenes. Para ello se dispone de imágenes de RMM de 346 pacientes distintos, lo que resulta en mas de 300.000 imágenes, obtenidas en Radboud University Medical Centre en Países Bajos utilizando dos variantes del escáner Siemens 3T MR, concretamente MAGNETOM Trio y Skyra cuyas características se exponen a continuación.

- **MAGNETOM Trio:** este escáner cuenta con uno de los imanes más potentes, con una longitud de 198 centímetros logra imágenes en alta calidad y es considerado uno de los más versátiles.
- **Skyra:** cuenta con un imán menor de 163 centímetros pero se trata de uno de los escáneres más modernos y cuenta con software específico que mejora la calidad de las imágenes obtenidas.

Con estos escáneres se utilizan tres técnicas para obtener diferentes tipos de imágenes con resoluciones específicas las cuales se exponen en la siguiente tabla 4.1.

Técnica	Resolución	Grosor de capa	Formato
T2-Ponderadas	0.5mm	3.6mm	DICOM
DW (ADC)	2mm	3.6mm	DICOM
DCE (Ktrans)	1.5mm	4mm	MHD

Tabla 4.1: Propiedades de imágenes en Prostatax.

Antes de explicar el estudio realizado a cada paciente se esclarece a que hace referencia el grosor de capa. Esto se debe a que las imágenes tridimensionales obtenidas por los escáneres están divididas en múltiples capas como se observa en el diagrama 4.1, una resolución menor daría como resultado un mayor número de capas por imagen.

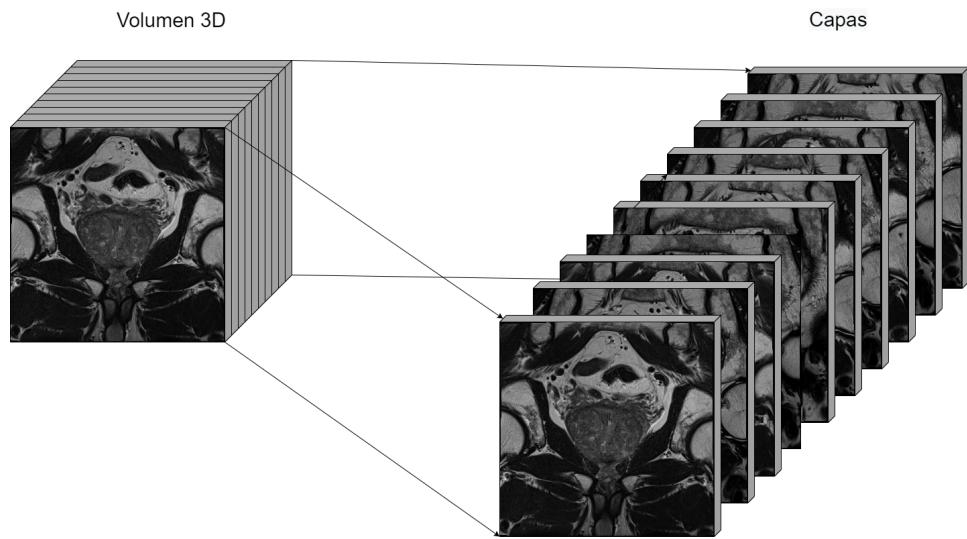


Figura 4.1: Visualización de la composición de un volumen 3D de zona pélvica.

A cada paciente se le realiza un estudio con una imagen de modalidad Ktrans, una o varias en modalidad DW y una o varias de modalidad T2-Ponderada por cada uno de los planos anatómicos. Es por ello que para la realización de este [TFG](#) se han utilizado un total de cinco modalidades por paciente para obtener el conjunto de datos más amplio posible. Estas modalidades se explican en profundidad en el apartado [2.1.1](#).

4.2.2. Lectura de datos

Al descargar el fichero de la base de datos se obtienen aparte de las imágenes anteriormente mencionadas, una serie de ficheros en formato csv con los hallazgos encontrados que mediante el identificador del paciente nos permite conocer información relevante como tamaño de imagen, posición del hallazgo en el escáner y en las coordenadas ijk, el tipo de prueba del que proviene, espaciamiento entre capas y número de capas. Para facilitar el procesado de estos datos en conjunto se crean dos funciones de lectura una para imágenes con formato MHD en las que solo se encuentre la modalidad Ktrans, y otra para aquellas con formato DICOM que incluye las imágenes ADC y las tres variedades de T2-ponderadas. Estas funciones combinan las imágenes y su información de los fichero csv en una sola estructura llamada dataframe que permite un acceso organizado y simple a cada uno de los campos. Este proceso de lectura tarda aproximadamente 20 minutos en completarse, es por ello qe para facilitar el trabajo y agilizar la ejecución del programa, una vez los dataframes son creados por primera vez, su estructura se almacena en un archivo binario que se recuperara en ejecuciones posteriores evitando así rehacer la lectura completa.

4.2.3. Muestra de imágenes

A continuación se observa una muestra de las variedades seleccionadas para el proyecto, compuesta por imágenes de tipo Ktrans a las que se aplica el filtro de color "jet", ADC y los tres planos de las imágenes T2-Ponderadas.

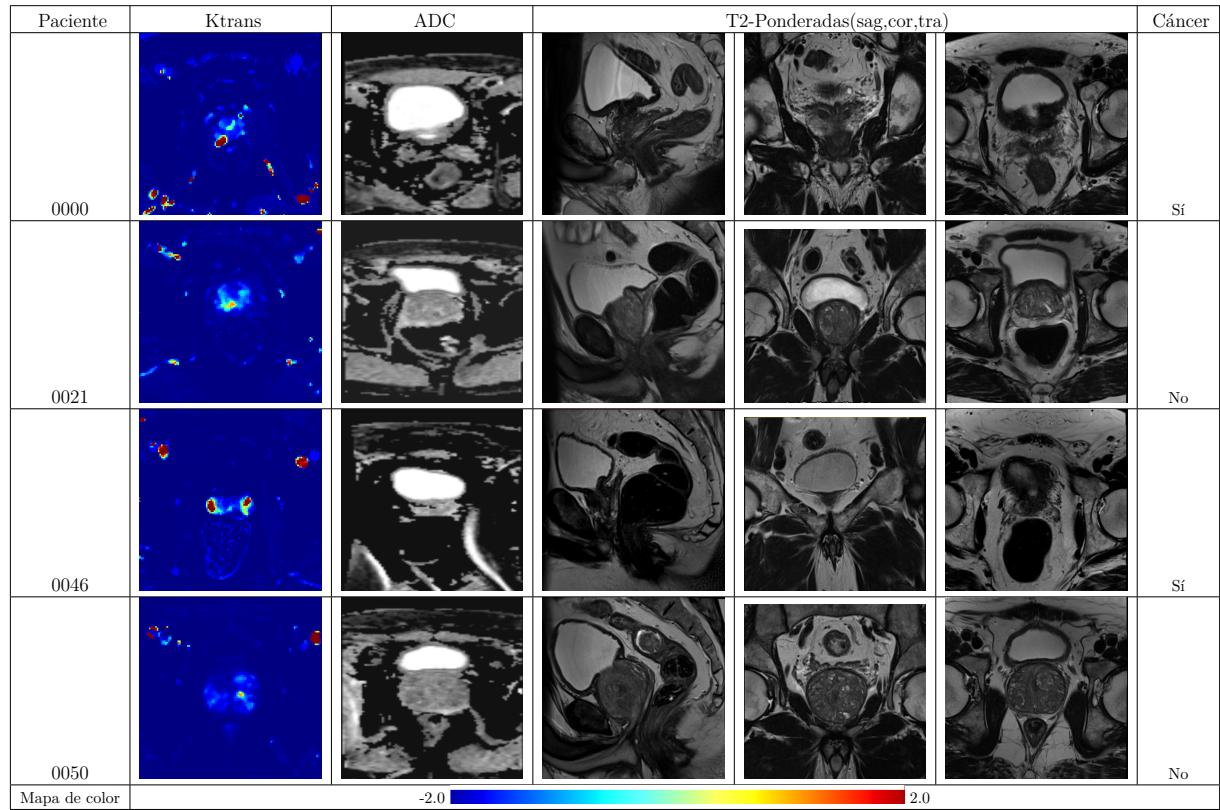


Tabla 4.2: Imágenes de ejemplo..

4.3. Función de coste y métricas de calidad

En esta sección se procede a explicar las diferentes métricas utilizadas para evaluar el rendimiento de las redes neuronales desarrolladas, así como la función de coste escogida.

4.3.1. Función de coste

La función de perdidas es posiblemente el factor más importante en cualquier red neuronal, ya que permite la obtención del resto de métricas. Este valor representa la penalización que se realiza por una predicción errónea. El objetivo de un entrenamiento es lograr minimizar los valores de pérdida para cada uno de los elementos.

Existen múltiples funciones de pérdidas, pero la más utilizada en los problemas tanto de clasificación como de segmentación binaria, a los cuales nos enfrentamos en el clasificador y localizador, es la conocida como pérdida de entropía cruzada binaria. A continuación se desarrolla su funcionamiento y se justifica la decisión de escogerla para este proyecto.

Debemos establecer para explicar el funcionamiento de la función los valores de dos parámetros: el valor esperado se representa como y_i , mientras que el valor de la predicción se indica como \hat{y}_i

Con esta función se consigue que cuando el valor esperado es 1, la segunda mitad de la ecuación desaparece, mientras que si es igual a 0 se ignora la primera mitad, de esta manera una predicción errónea tiene un alto valor de pérdida ya que los valores para log entre 1 y 0 aumentan exponencialmente. Una vez obtenidos todos los resultados se realiza una media para obtener el valor final de la función de pérdidas.

$$\text{Loss} = -\frac{1}{n} \sum_{\text{size}}^n y_i \cdot \log \hat{y}_i + (1 - y_i) \cdot \log (1 - \hat{y}_i) \quad (4.1)$$

4.3.2. Ráiz del error cuadrático medio

También conocido como Ráiz del error cuadrático medio (RMSE) por sus siglas en inglés (root-mean-square error) y es la métrica más utilizada en las tareas de regresión y representa la raíz cuadrada de la distancia promedio cuadrada entre dos conjuntos de datos, normalmente entre valores reales y valores de predicción. Esto indica el ajuste del modelo, es decir cómo de cerca están los valores que nuestro modelo predice de los valores reales, por lo tanto, en un caso idóneo con un modelo perfecto el resultado de RMSE sería 0 ya que podría predecir con total precisión todos y cada uno de los puntos. En la figura 4.2 se observa una recta de regresión en rojo, los puntos reales en azul, y en negro el error entre ambos utilizado para calcular RMSE.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (4.2)$$

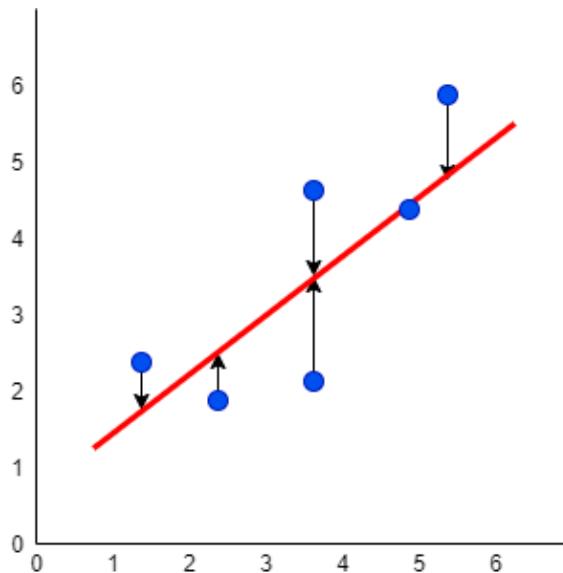


Figura 4.2: Ejemplo del error de la recta de regresión a cada punto.

4.3.3. Matriz de confusión

Antes de explicar las siguientes métricas, definimos cuatro métricas propias de las clasificaciones binarias en las que se basan la mayoría de ellas y que se representan de forma conjunta en forma de matriz. Estos valores se extraen una vez entrenado el modelo y realizadas las predicciones comparando los valores de la predicción con los reales.

- **Verdaderos positivos (TP)** Número de casos en los que se identifica cáncer y presenta cáncer en el etiquetado.
- **Falsos positivos (FP)** Número de casos en los que se identifica cáncer pero no están registrados como pacientes con cáncer.
- **Verdaderos negativos (TN)** Número de casos que no se ha detectado cáncer y en el etiquetado tampoco lo presentan.

- **Falsos negativos (FN)** Número de casos en los que no se ha identificado cáncer pero sí están registrados como pacientes con cáncer.

El caso idóneo en obtener el 100 % de los valores en las celdas de la diagonal, lo que supondría que el valor de la predicción ha sido el correcto en todos los casos.

		Predicción	
		Negativa	Positiva
Etiqueta real	Negativa	TN	FP
	Positiva	FN	TP

Figura 4.3: Posición de cada parámetro en la matriz de confusión.

El escenario idóneo es tener el máximo de verdaderos positivos y negativos ya que esto significaría que nuestra red es capaz de realizar una clasificación perfecta.

4.3.4. Accuracy o exactitud

Esta métrica nos indica la tasa de acierto de nuestro modelo, para ello se divide el número de TP y TN entre el número total de casos. Para que esta métrica sea representativa se debe contar con un número equilibrado de casos positivos y negativos.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (4.3)$$

4.3.5. Ratio de verdaderos positivos (TPR)

Hace referencia a la capacidad del sistema de detectar correctamente aquellos pacientes enfermos, para ello se divide el número de casos detectados entre el total de casos con cáncer.

$$\text{TPR} = \frac{\text{TP}}{\text{P}} = \frac{\text{TP}}{\text{TP} + \text{FN}} = 1 - \text{FNR} \quad (4.4)$$

4.3.6. Valor predictivo positivo (PPV)

Este valor indica la probabilidad de que un paciente que haya sido marcado como positivo sea realmente positivo.

$$\text{PPV} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (4.5)$$

4.3.7. F-score

Esta métrica es la media armónica de los ya comentados TPR y PPV. El mejor valor que este parámetro puede alcanzar es 1 lo que supondría que TPR y PPV son perfectos, en su defecto el menor

valor es 0 si TPR o PPV son 0.

$$F = 2 \cdot \frac{\text{TPR} \cdot \text{PPV}}{\text{TPR} + \text{PPV}} \quad (4.6)$$

4.3.8. Curva de Característica Operativa del Receptor (ROC)

Es una representación gráfica de los valores de TPR frente a FPR a medida que se varía el umbral de decisión. En este gráfico se suele observar una línea diagonal que representara unos resultados de clasificación aleatorios. Los valores por encima de esta diagonal suponen una predicción mejor que el azar y peor si los valores están por debajo. Así un valor de clasificación perfecto sería un punto en la coordenada (0,1). Esto se observa con mayor claridad en la figura 4.4.

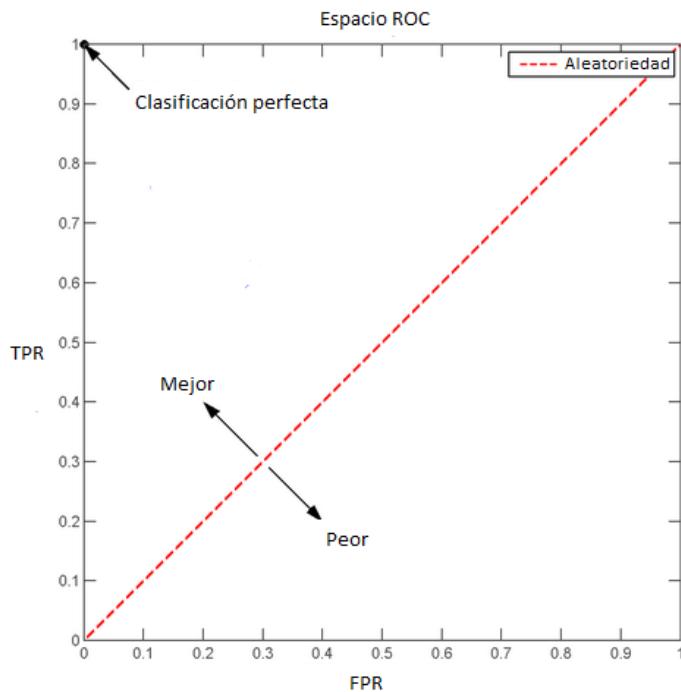


Figura 4.4: Ejemplo del espacio en el que se representa la curva ROC.

4.3.9. Área bajo la curva (AUC)

El indicador más utilizado de la curva ROC consiste en medir el área que queda bajo la curva dibujada, obteniendo así la capacidad de clasificación de nuestro clasificador en función de un umbral. Actúa como una métrica de estabilidad del clasificador entrenado puesto que una AUC de 1 supondría que nuestro clasificador es perfecto con cualquier umbral.

4.4. Resultados experimentales

En esta sección se exponen los mejores resultados de los entrenamientos realizados a los sistemas propuestos anteriormente. Primero se expondrán los resultados de las redes unimodales de clasificación para cada tipo de imagen y posteriormente los resultados de los sistemas multimodales de clasificación. De cada modelo se mostrará la curva ROC, la matriz de confusión y una tabla con los resultados de:

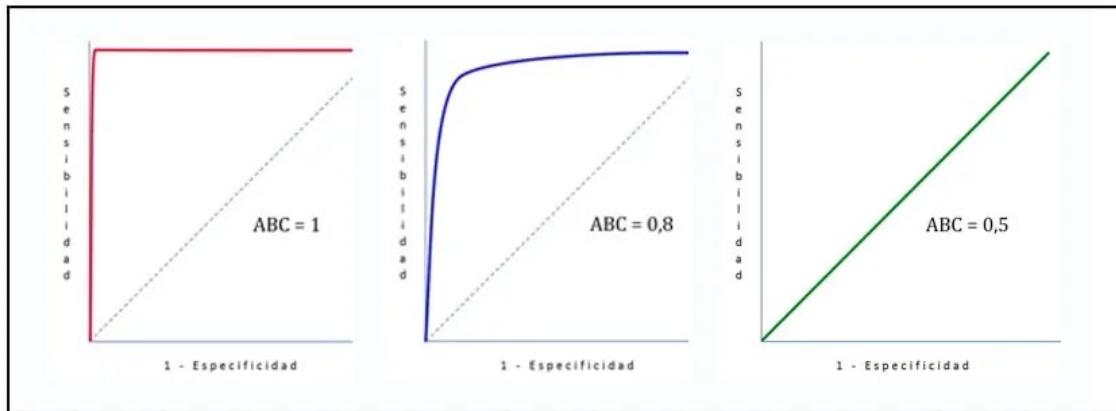


Figura 4.5: Ejemplo de diferentes curvas y valores de AUC. Imagen obtenida de [10].

exactitud, F-score y AUC. Por último, se muestran los resultados obtenidos en la red de localización de anomalías.

4.4.1. Resultados de redes unimodales de clasificación

En este apartado se analizan los resultados obtenidos de manera conjunta por ello solo se destacarán aquellos resultados más reseñables y se expondrán todos los datos a modo de comparativa.

4.4.1.1. Matrices de confusión

En la tabla 4.3 podemos observar las matrices obtenidas por las cinco arquitecturas unimodales de clasificación, a simple vista se observa que los mejores resultados los aportan las modalidades de Ktrans y ADC con valores de TPR y TNR bastante equilibrados y en torno a un 70 % de media (73 % y 68,5 % respectivamente) esto es un primer indicativo de que estas modalidades pueden ser potentes indicadores para la detección de anomalías, siendo Ktrans la mejor de entre las dos.

En cuanto a las imágenes T2-Ponderadas observamos una gran disparidad entre cada uno de los planos anatómicos. Se obtiene el mejor resultado en las imágenes coronales, siendo estas las más equilibradas con un 62 % de media, por lo que aunque no sean resultado tan buenos como Ktrans y ADC, sigue siendo un valor que supera por bastante a la aleatoriedad. Con las imágenes Sagitales y Trasversales aunque se pueda obtener una media superior a la aleatoriedad, se aprecia un gran desequilibrio en las diagonales por lo que en un primer análisis no parecen un buen indicador para la detección de anomalías, ya que están muy desbalanceadas entre la detección de positivos y negativos correctamente.

Cabe mencionar en cuanto a estas dos últimas modalidades, que en este caso que para el objetivo que se plantea en este TFG el resultado obtenido en la modalidad Trasversal es preferible al de Sagital, esto se debe a que en última instancia se trata de identificar un posible cáncer en los pacientes por lo que es preferible obtener el menor número de falsos negativos que uno de falsos positivos ya que esto supondría pasar por alto una posible lesión grave.

4.4.1.2. Curvas ROC

En la tabla 4.4 podemos observar las curvas ROC en la misma disposición que las matrices de confusión. Recordamos para este análisis que el mejor resultado es aquel cuya curva se acerque al punto (0,1) y que la aleatoriedad aparece marcada por la recta diagonal roja. Teniendo esto en cuenta no es

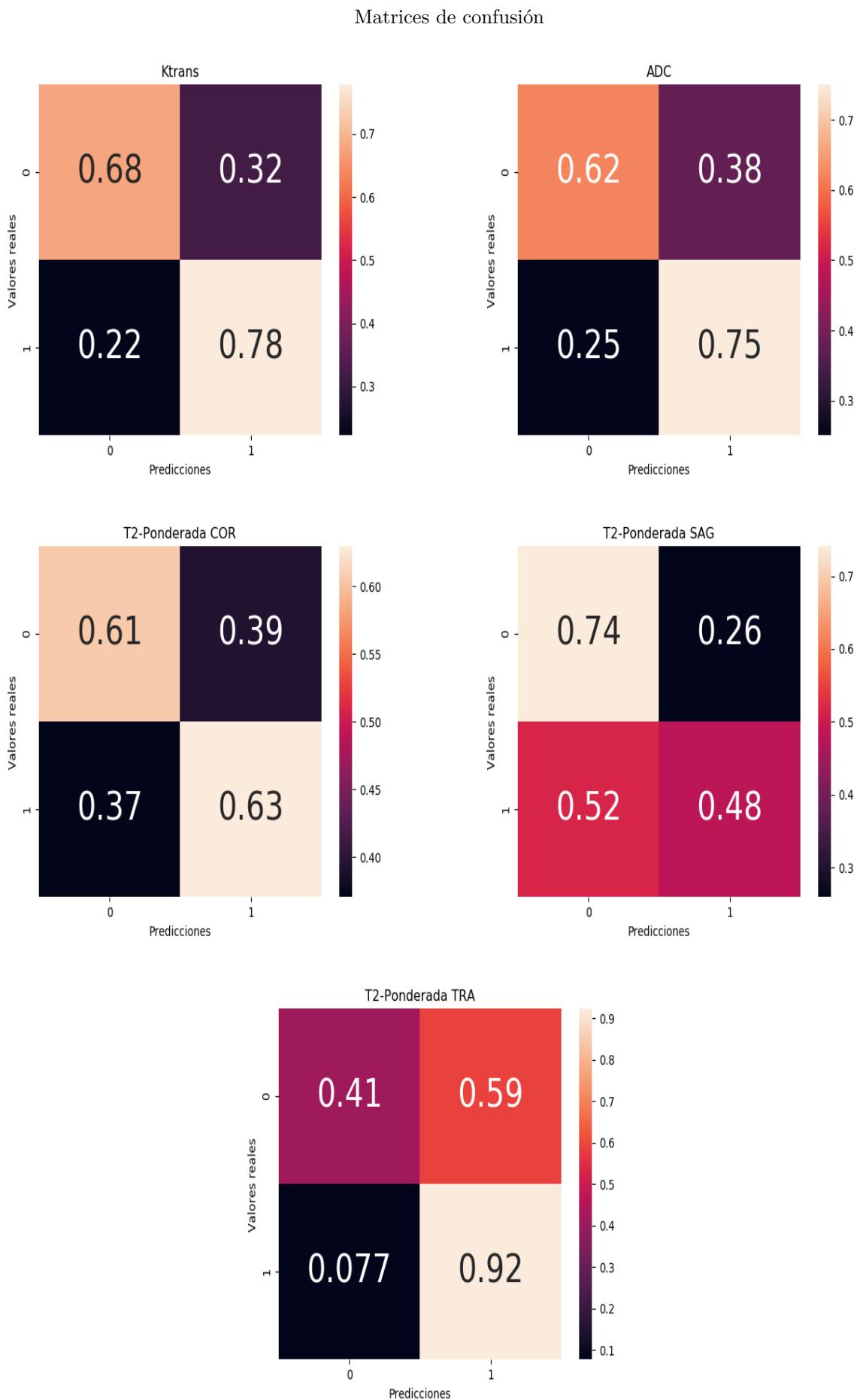


Tabla 4.3: Matrices de confusión de arquitecturas unimodales

difícil apreciar como las curvas obtenidas por las modalidades Ktrans y ADC son las que más se acercan a este punto. En cambio se observa como Ktrans tiene una forma suavizada y una respuesta esperable de un buen clasificador mientras que ADC tiene una forma más abrupta y para algunos umbrales roza la aleatoriedad. Las modalidades T2-Ponderadas, COR y SAG obtienen resultados muy cercanos a esta diagonal, mientras que TRA tiene inestabilidades en los valores centrales del umbral de clasificación, por lo que sus resultados no parecen indicar que estas modalidades sean de gran utilidad a la hora de detectar anomalías. Se realiza un mejor análisis de estos resultados en la siguiente sección donde se presenta el área bajo la curva (AUC) de cada modalidad.

4.4.1.3. Resultados de conjunto

En la tabla 4.5 se recoge un resumen de los gráficos mostrados anteriormente junto a las métricas F-score, AUC y Exactitud.

Comenzaremos evaluando los resultados de F-score y recordando que un valor óptimo para esta métrica es 1. Observamos que Ktrans es la modalidad que más supera 0.5, ADC y las imágenes Coronales giran entorno a este valor, mientras que Sagitales y Trasversales no superan el umbral de 0.5. En base a esta métrica la modalidad Ktrans sería el mejor marcador.

A continuación, se evalúa el AUC para completar en análisis de las curvas ROC efectuado anteriormente, una vez más Ktrans y ADC destacan por encima del resto logrando un 75 % y un 67% de superficie respectivamente en cambio la imágenes de modalidad T2-Ponderada permanecen cerca de un 60% un valor no excesivamente alto pero sí por encima de la aleatoriedad como se podía observar en sus curvas ROC.

Finalmente, la métrica de exactitud, obtenida con un umbral de 0.5, nos ayuda a confirmar que Ktrans obtiene los mejores resultados logrando un 70 % de exactitud y convirtiéndola en la mejor modalidad para la detección de anomalías. En segundo lugar, está la modalidad ADC obteniendo 65 % de exactitud y resultados sólidos en el resto de métricas ya comentadas. Las imágenes de modalidad T2-Ponderada obtienen resultados dispares al igual que en las otras métricas por lo que no se posicionan como modalidad de referencia en este proyecto, aun así sus valores, que se sitúan siempre ligeramente por encima de la aleatoriedad pueden ser útiles en las arquitecturas multimodales lo cual se analizará más adelante.

4.4.2. Arquitecturas multimodales de clasificación

4.4.2.1. Arquitectura de fusión por tensor de entrada

En esta arquitectura se utilizan las 5 modalidades anteriores de un mismo paciente en una red igual a las unimodales. Se analizan a continuación las métricas obtenidas con esta arquitectura con el fin de comprobar si la unión de diferentes modalidades obtiene mejores resultados. Mediante el análisis de la curva ROC 4.7 podemos observar que el clasificador sitúa su capacidad bastante por encima de la aleatoriedad y a simple vista se asemeja al resultado obtenido por la modalidad Ktrans. Por otro lado, a través de la matriz de confusión 4.6 observamos un TPR y un TNR que tienen valores de 80 % y 70 % y un valor medio de 75 %, por encima del 73%, siendo el mejor resultado obtenido en las arquitecturas unimodales. Podemos concluir que el análisis en conjunto obtiene mejores resultados, y los resultados previos de las redes unimodales nos llevan a pensar que la modalidad Ktrans aporta la mayor parte de la eficacia, mientras que las otras modalidades aportan la mejoría de un 2-3 por ciento con respecto a Ktrans de forma unimodal.

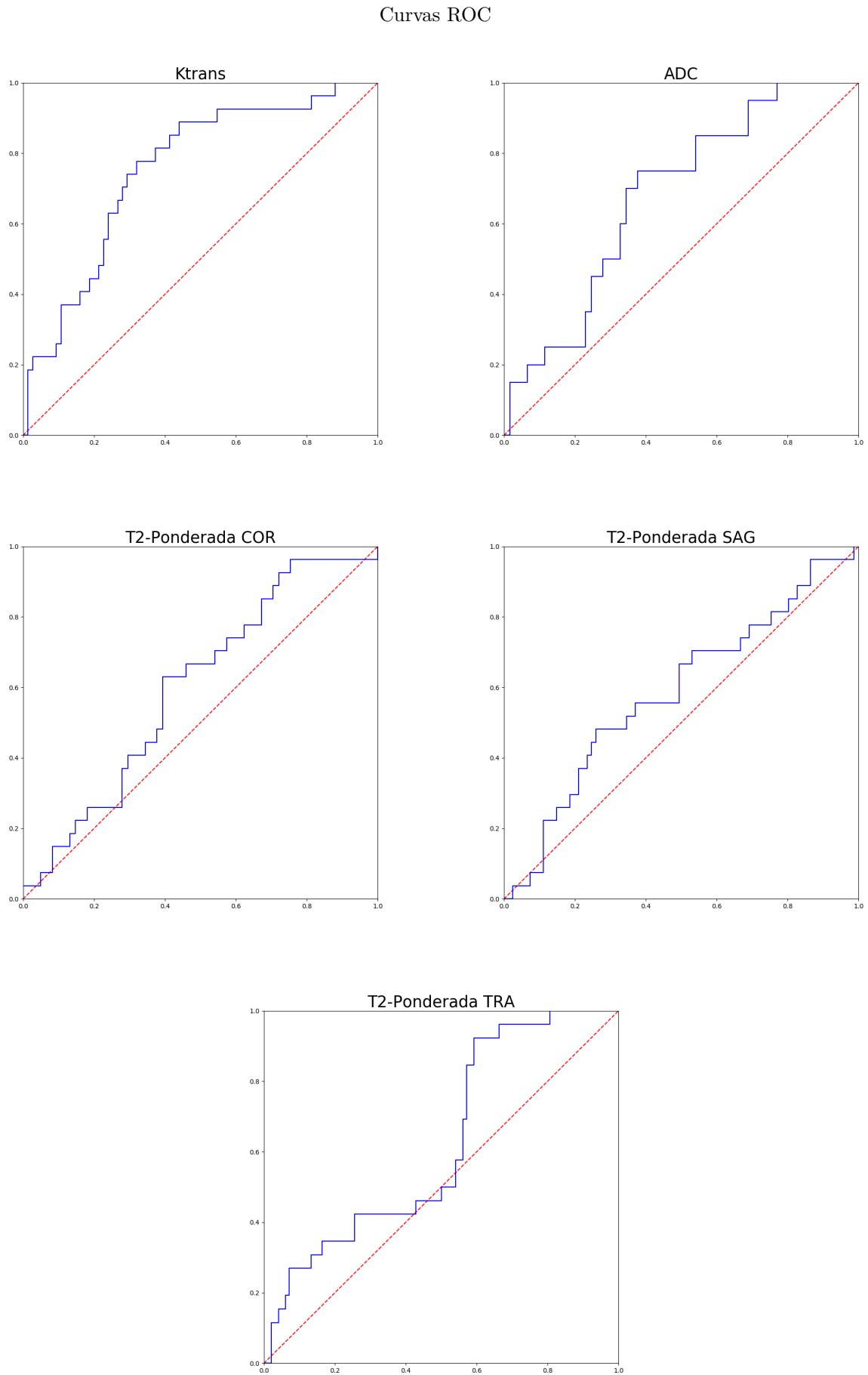


Tabla 4.4: Curvas ROC de arquitecturas unimodales.

Modalidad	TPR	TNR	F-score	AUC	Exactitud
Ktrans	0.78	0.68	0.5833	0.7545	0.705
ADC	0.75	0.62	0.5172	0.6795	0.6543
T2-Ponderada COR	0.63	0.61	0.5	0.5992	0.6136
T2-Ponderada SAG	0.48	0.74	0.4262	0.58	0.6759
T2-Ponderada TRA	0.92	0.41	0.4444	0.6251	0.5161

Tabla 4.5: Métricas de todas las arquitecturas unimodales.

Por último, en la tabla 4.6 con las diferentes métricas se ha optado por incluir los valores de las redes unimodales para poder realizar una comparación directa. Tras analizar estos datos se observa que esta arquitectura de fusión mejora cualquier dato de las arquitecturas unimodales, se deben valorar estas mejoras, aunque puedan parecer escasas, puesto que se está mejorando sobre una tasa de acierto ya elevada (en el caso de ktrans). Esta mejora en los resultados confirma la hipótesis de que aunque no todas las modalidades obtienen resultados óptimos de manera individual en el conjunto esas pequeñas aportaciones llegan a marcar la diferencia a la hora de detectar anomalías clínicas relacionadas con el cáncer de próstata.

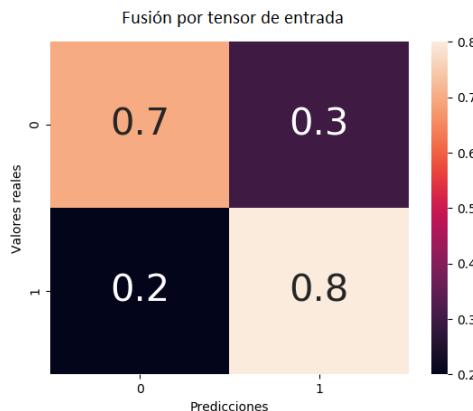


Figura 4.6: Matriz de confusión de arquitectura de fusión por tensor de entrada.

Modalidad	TPR	TNR	F-score	AUC	Exactitud
Fusión por tensor de entrada	0.80	0.70	0.6666	0.7577	0.7333
Ktrans	0.78	0.68	0.5833	0.7545	0.705
ADC	0.75	0.62	0.5172	0.6795	0.6543
T2-Ponderada COR	0.63	0.61	0.5	0.5992	0.6136
T2-Ponderada SAG	0.48	0.74	0.4262	0.58	0.6759
T2-Ponderada TRA	0.92	0.41	0.4444	0.6251	0.5161

Tabla 4.6: Métricas obtenidas de fusión por tensor de entrada y arquitecturas unimodales anteriores.

4.4.2.2. Arquitectura de fusión de etapas unimodales

Para este modelo se utilizan las 5 arquitecturas unimodales de clasificación de manera combinada y se procesa en cada una de ella la imagen de la modalidad correspondiente de un mismo paciente. Se analizan a continuación los resultados obtenidos y se comparan con todas las arquitecturas anteriores. Analizando el resultado de la curva ROC 4.9 podemos obtener una primera idea de que este clasificador tendrá unos resultados similares al de las arquitecturas previas, puesto que se sitúa su capacidad bastante

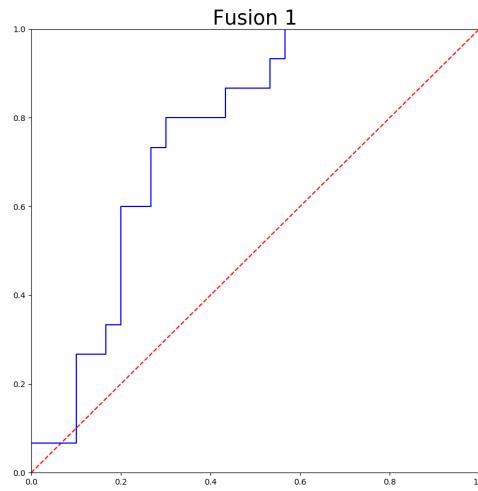


Figura 4.7: Curva ROC de arquitectura de fusión por tensor de entrada.

por encima de la aleatoriedad, pero no se acerca en exceso a la clasificación perfecta. Mediante el análisis de la matriz de confusión 4.8 observamos un valor de TPR y TNR de 87% y 70%, superando el mejor valor hasta ahora que era 80% y 70% obtenidos mediante la fusión por tensor de entrada. Esto nos indica que la combinación de estas arquitecturas unimodales de clasificación, no solo nos permiten crear sistemas más robustos y desacoplados, sino que también se mejoran los resultados obtenidos, tanto por las arquitecturas unimodales por separado como por la arquitectura de fusión por tensor de entrada. Por último para validar esta afirmación se presentan los datos del resto de métricas obtenidas, en la tabla 4.7 se incluyen todos los resultados obtenidos hasta ahora para apreciar más fácilmente las mejoras. Se puede observar una ligera mejoría en todas las métricas respecto a los mejores resultados obtenidos hasta ahora con la arquitectura de fusión por tensor de entrada logrando por ejemplo una mejora de un 2% en la exactitud de la predicción y un 5% si comparamos con la arquitectura unimodal Ktrans. Una vez más se destaca la relevancia de esta ligera mejoría puesto que cuanto mayor sea la tasa de éxito anterior más reseñable es obtener mejoras sobre la misma y más en un campo tan complejo como el que cubre este **TFG**.

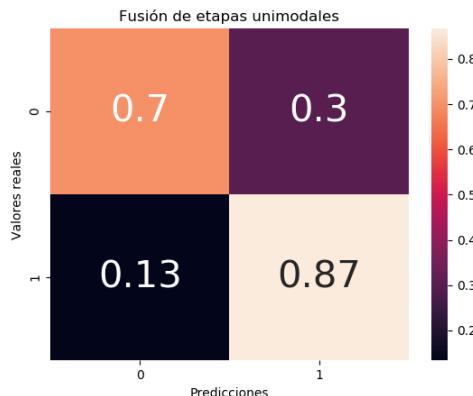


Figura 4.8: Matriz de confusión de arquitectura de fusión de etapas unimodales.

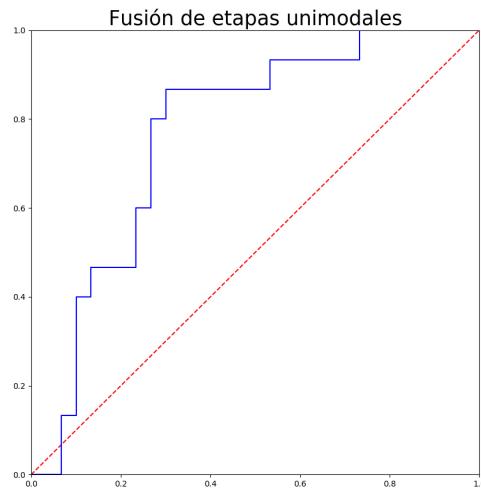


Figura 4.9: Curva ROC de arquitectura de fusión de etapas unimodales.

Modalidad	TPR	TNR	F-score	AUC	Exactitud
Fusión de etapas unimodales	0.87	0.70	0.7027	0.7667	0.7555
Fusión por tensor de entrada	0.80	0.70	0.6666	0.7577	0.7333
Ktrans	0.78	0.68	0.5833	0.7545	0.705
ADC	0.75	0.62	0.5172	0.6795	0.6543
T2-Ponderada COR	0.63	0.61	0.5	0.5992	0.6136
T2-Ponderada SAG	0.48	0.74	0.4262	0.58	0.6759
T2-Ponderada TRA	0.92	0.41	0.4444	0.6251	0.5161

Tabla 4.7: Métricas obtenidas de fusión de etapas unimodales, fusión por tensor de entrada y arquitecturas unimodales anteriores.

4.4.3. Resultados de red de localización

A continuación se exponen los resultados del localizador de anomalías propuesto. Estos resultados se expondrán de dos maneras. La primera de ellas será cuantitativa, mostrando el RMSE de varias imágenes ejemplificativas como el RMSE medio en milímetros de la base de datos, teniendo en cuenta la resolución del instrumento utilizado y definida previamente en 4.2. El RMSE será calculado entre el centroide de la circunferencia del etiquetado y el centroide de la superficie obtenida en la predicción tras ser umbralizada. La segunda forma será de forma gráfica mostrando la sección bidimensional del volumen 3D en el que se encuentra el cáncer tanto en el etiquetado, como en la predicción de salida del sistema. Todos estos resultados se visualizan en la tabla 4.8.

Paciente	Mapa de interés bidimensional	Etiquetado	Predicción	RMSE
0056				3.21
0026				2.53
0023				2.89
0000				4.41
RMSE medio			3.46	
Mapa de color	-2.0		2.0	

Tabla 4.8: Resultados cualitativos y cuantitativos de la red de localización propuesta.

En la tabla anterior se pueden observar los resultados para 4 pacientes distintos de la base de datos. En cada una de las columnas, se muestran los mapas los mapas bidimensionales de las rodajas (obtenidas

del volumen 3D) en las cuales se encuentra la anomalía detectada, tanto en el volumen Ktrans de entrada, como en el etiquetado y la predicción del sistema propuesto. En la primera columna, el mapa bidimensional Ktrans se representa a través de la caracterización de colores "viridis" para una mejor visualización. En la segunda y tercera columna, se muestran los mapas sin la caracterización de colores, solo en escala de grises. En estos se plasman las superficies de detección tanto del etiquetado como la propuesta por el localizador.

Es posible observar que las predicciones se acercan bastante al etiquetado, siempre teniendo en cuenta lo pequeña que es la próstata que mide de media 40x30x20 mm. Gráficamente podemos observar que en la mayoría de los casos la predicción proyectada sobre el mapa XY se aproxima bastante al etiquetado, mientras que numéricamente observamos resultados en los 4 pacientes no mayores a 5 mm de error, siendo la media de todas las localizaciones de la base de datos de 3.46 mm de error. Estos datos indican que las predicciones tienen un error del orden del 10 % con respecto a la dimensión máxima de la próstata. Todo esto nos lleva a pensar que esta red de localización funciona correctamente y podría mejorar sus resultados a través del uso de bases de datos mayores u otros tipos de aumento de datos.

Capítulo 5

Conclusiones y líneas futuras

A continuación se presentan las principales conclusiones del trabajo realizado, así como algunas líneas acerca de los posibles trabajos a realizar en el futuro.

5.1. Conclusiones

En este trabajo se ha propuesto un sistema de detección de cáncer de próstata basado en diversas modalidades de imágenes de [RMM](#) mediante [DNN](#), con la finalidad de facilitar el diagnóstico de este tipo de enfermedad.

Durante el desarrollo se han presentado diferentes arquitecturas, las cuales han sido evaluadas de forma exhaustiva de manera experimental, con el objetivo obtener aquella con los mejores resultados. Las métricas obtenidas por las diferentes arquitecturas nos permiten afirmar que las imágenes de modalidad Ktrans obtienen los mejores resultados de manera individual, llegando a un 70 % de exactitud. Sin embargo, al combinar las cinco modalidades de imágenes en la arquitectura de fusión por tensor de entrada, se consiguen superar estos resultados. Esto indica que los patrones reconocidos en todas las modalidades resultan en un sistema más robusto al combinarse, alcanzando una exactitud del 73 %. Por último, el mejor de los resultados se logra en la arquitectura de fusión de etapas unimodales que, pese a suponer una pequeña mejora del 2 % alcanzando así un 75 %, sigue siendo una mejora destacable en este tipo de sistemas de clasificación. Este último resultado no solo confirma una vez más la hipótesis de que el conjunto de modalidades aporta información útil a la hora de hacer la clasificación, sino que además nos permite combinar las imágenes de manera desacoplada, ya que se puede entrenar cada arquitectura unimodal de forma independiente y solo sería necesario reajustar los pesos finales de la red de fusión.

En lo que respecta a la red de localización se obtiene un [RMSE](#) medio de 3.46mm, que corresponde aproximadamente a un 10 % de error en comparación con la dimensión máxima de la próstata. Este es un error mejorable pero que en cualquier caso puede servir de indicador para un radiólogo experimentado que podrá ubicarlo posteriormente con mayor precisión.

Pese a los buenos resultados obtenidos no se puede considerar que este sistema tenga una precisión lo suficientemente alta como para utilizarlo como prueba de diagnóstico única, pero si pudiera ser de gran utilidad como sistema de apoyo o segunda opinión a los radiólogos, pudiendo llegar a detectar casos positivos pasados por alto o simplemente facilitando y agilizando la tarea de detección de aquellos tumores más fáciles de detectar.

5.2. Líneas futuras

En este apartado se proponen una serie de líneas de investigación del sistema desarrollado que pueden mejorar o evaluar de forma diferente los resultados.

- **Elaboración de nuevas arquitecturas:** en este proyecto se han definido un total de siete arquitecturas pero las posibilidades son infinitas. Se propone elaborar nuevas arquitecturas unimodales que alcancen mayor tasa de éxito en los clasificadores, principalmente en las modalidades T2-Ponderadas pudiendo así de forma indirecta mejorar los resultados de la arquitectura de fusión de etapas unimodales.
- **Búsqueda de nuevos marcadores:** añadir al entrenamiento otro tipo de marcadores médicos como edad del paciente, peso, PSA que como se ha comentado en la sección [2.3.4](#) se utilizan en la actualidad como marcadores para el diagnóstico. Se podría estudiar la posibilidad de añadir estos datos en la etapa final de la arquitectura de fusión de etapas unimodales.
- **Colaboración para prueba en entorno real:** actualmente la Universidad de Alcalá colabora con el Hospital Universitario Príncipe de Asturias para diferentes estudios se podría plantear una colaboración para:
 - **Creación de una base de datos propia:** trabajar en conjunto con el Hospital Universitario Príncipe de Asturias para crear una base de datos propia en la que se colaboraría con expertos para el correcto etiquetado de las pruebas.
 - **Prototipado y prueba:** abordar el desarrollo de un prototipo que permita probar el sistema en un entorno médico real.

Capítulo 6

Presupuesto

6.1. Costes de equipamiento

6.1.1. Equipamiento hardware utilizado:

Concepto	Cantidad	Coste unitario	Subtotal(euros)
Ordenador AMD Ryzen 5	1	500	500
tarjeta Grafica NVIDIA 1660	1	250	250
Coste Total			750

6.1.2. Recursos software utilizados:

Concepto	Cantidad	Coste unitario	Subtotal(euros)
Windows 10	1	0	0
Librerias tensorflow	1	0	0
Librerias Keras	1	0	0
Librerias Scipy	1	0	0
Librerias Opencv	1	2000	2000
Texstudio	1	0	0
Total			0

6.2. Costes Mano de obra

Concepto	Cantidad	Coste unitario	Subtotal(euros)
Programación y Desarrollo de Software	280	60euros/hora	17220
Mecanografiado de documentación, manuales y tutoriales	65	15 euros/hora	975
Total			18345

6.3. Costes Totales

Concepto	Subtotal
Hardware	676,71
Software	5250
Mano de obra	18345
Total	18345

El importe total del presupuesto asciende a la cantidad de: DIECIOCHO MIL TRESCIENTOS CUA- RENTA Y CINCO EUROS

En Alcalá de Henares, a 9 de Diciembre de 2020.

Javier Fuentes Fernández Graduado en Ingeniería Informática

Bibliografía

- [1] M. L. galan, A. P. borda, I. F. gonzalez, I. R. cajigal, J. G. de vicente, and A. B. sanchez, “Efectos adversos de la biopsia prostatica transrectal. un analisis de 303 procedimientos,” *Actas Urologicas Espanolas*, vol. 25, no. 1, pp. 46–49, 2001.
- [2] M. A. G. Perez, “Valor de la resonancia magnetica multiparametrica en el diagnostico del cancer de prostata en pacientes sin biopsia previa,” 2019.
- [3] S. G. Armato, H. Huisman, K. Drukker, L. Hadjiiski, J. S. Kirby, N. Petrick, G. Redmond, M. L. Giger, K. Cha, A. Mamonov *et al.*, “Prostatex challenges for computerized classification of prostate lesions from multiparametric magnetic resonance images,” *Journal of Medical Imaging*, vol. 5, no. 4, p. 044501, 2018.
- [4] J. G. Burneo, D. A. Steven, M. Arango, W. Zapata, C. M. Vasquez, and A. Becerra, “La cirugía de epilepsia y el establecimiento de programas quirúrgicos en el perú: El proyecto de colaboración entre perú y canadá.” *Revista de Neuro-Psiquiatría*, vol. 80, no. 3, pp. 181–188, 2017.
- [5] S. Sharma, “Activation functions in neural networks,” Feb 2019. [Online]. Available: <https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6>
- [6] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: A simple way to prevent neural networks from overfitting,” vol. 15, no. 1, 2014.
- [7] R. Dwivedi, “How data augmentation impacts performance of image classification, with codes,” Nov 2020. [Online]. Available: <https://analyticsindiamag.com/image-data-augmentation-impacts-performance-of-image-classification-with-codes/>
- [8] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, url<http://www.deeplearningbook.org>.
- [9] M. Bastida, M. Cabello, A. Rodríguez, and J. García, “Sistema de apoyo para la toma de decisiones en termografía de glándulas mamarias,” *Revista mexicana de ingeniería biomédica*, vol. 38, no. 1, pp. 166–187, 2017.
- [10] “Sensibilidad, especificidad,” Oct 2020. [Online]. Available: <https://www.cienciasinseso.com/sensibilidad-especificidad-valores-predictivos/>
- [11] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “Imagenet: A large-scale hierarchical image database,” in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.
- [12] G. A. Miller, “Wordnet: a lexical database for english,” *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.

- [13] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.
- [14] W. Sun, B. Zheng, and W. Qian, “Computer aided lung cancer diagnosis with deep learning algorithms,” in *Medical imaging 2016: computer-aided diagnosis*, vol. 9785. International Society for Optics and Photonics, 2016, p. 97850Z.
- [15] N. Coudray, P. S. Ocampo, T. Sakellaropoulos, N. Narula, M. Snuderl, D. Fenyö, A. L. Moreira, N. Razavian, and A. Tsirigos, “Classification and mutation prediction from non-small cell lung cancer histopathology images using deep learning,” *Nature medicine*, vol. 24, no. 10, pp. 1559–1567, 2018.
- [16] A. S. Becker, M. Marcon, S. Ghafoor, M. C. Wurnig, T. Frauenfelder, and A. Boss, “Deep learning in mammography: diagnostic accuracy of a multipurpose image analysis software in the detection of breast cancer,” *Investigative radiology*, vol. 52, no. 7, pp. 434–440, 2017.
- [17] J. Vilanova and V. Catala, “La resonancia magnetica en el nuevo paradigma del diagnostico del cáncer de prostata,” *Radiología*, vol. 59, no. 2, pp. 94 – 99, 2017. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0033833816301795>
- [18] S. Alonso Roca, A. Delgado Laguna, J. Arantzeta Lexarreta, B. Cajal Campo, and S. Santamaría Jareno, “Cribado en pacientes con riesgo incrementado de cancer de mama (parte 1). pros y contras del cribado con resonancia magnetica,” *Radiología*, vol. 62, no. 4, pp. 252 – 265, 2020. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0033833820300308>
- [19] R. S. of North America (RSNA) and A. C. of Radiology (ACR), “Materiales de contraste,” 2018. [Online]. Available: <https://www.radiologyinfo.org/sp/info.cfm?pg=safety-contrast>
- [20] Y. Cao, L. Xu, Y. Kuang, D. Xiong, and R. Pei, “Gadolinium-based nanoscale mri contrast agents for tumor imaging,” *J. Mater. Chem. B*, vol. 5, pp. 3431–3461, 2017. [Online]. Available: <http://dx.doi.org/10.1039/C7TB00382J>
- [21] Z. Han, Y. Li, S. Roelle, Z. Zhou, Y. Liu, R. Sabatelle, A. DeSanto, X. Yu, H. Zhu, C. Magi-Galluzzi, and Z.-R. Lu, “Targeted contrast agent specific to an oncoprotein in tumor microenvironment with the potential for detection and risk stratification of prostate cancer with mri,” *Bioconjugate Chemistry*, vol. 28, no. 4, pp. 1031–1040, 2017, pMID: 28201871. [Online]. Available: <https://doi.org/10.1021/acs.bioconjchem.6b00719>
- [22] E. University, “La resonancia magnetica (rm),” 2010. [Online]. Available: <https://www.cancerquest.org/es/para-los-pacientes/deteccion-y-diagnosis/resonancia-magnetica-rm>
- [23] D. Ciregan, U. Meier, and J. Schmidhuber, “Multi-column deep neural networks for image classification,” pp. 3642–3649, 2012.
- [24] T. M. Quan, D. G. Hildebrand, and W.-K. Jeong, “Fusionnet: A deep fully residual convolutional neural network for image segmentation in connectomics,” *arXiv preprint arXiv:1612.05360*, 2016.
- [25] D. Fuentes Jiménez *et al.*, “Reconstrucción 3d de sólidos deformables mediante el uso de redes convolucionales,” 2018.
- [26] I. Stephen, “Perceptron-based learning algorithms,” *IEEE Transactions on neural networks*, vol. 50, no. 2, p. 179, 1990.

- [27] L. Perez and J. Wang, “The effectiveness of data augmentation in image classification using deep learning,” *arXiv preprint arXiv:1712.04621*, 2017.
- [28] L. Bottou, “Large scale machine learning with stochastic gradient descent,” in *Proceedings of COPSTAT’2010*. Springer, 2010, pp. 177–186.
- [29] J. Duchi, E. Hazan, and Y. Singer, “Adaptive subgradient methods for online learning and stochastic optimization.” *Journal of machine learning research*, vol. 12, no. 7, 2011.
- [30] L. Agnes and F. Sagayaraj, “Adagrad-an optimizer for stochastic gradient descent,” *Int. J. Inf. Comput. Sci.*, vol. 6, no. 5, 2019.
- [31] G. Hinton, N. Srivastava, and K. Swersky, “Lecture 6a overview of mini-batch gradient descent,” 2012. [Online]. Available: http://www.cs.toronto.edu/~tijmen/csc321/slides/lecture{__}slides{__}lec6.pdf
- [32] A. C. Society, “Que es el cancer?” 2016. [Online]. Available: <https://www.cancer.org/es/cancer/aspectos-basicos-sobre-el-cancer/que-es-el-cancer.html>
- [33] A. J. Alberg and J. M. Samet, “Epidemiology of lung cancer,” *Chest*, vol. 123, no. 1, pp. 21S–49S, 2003.
- [34] P. F. Engstrom, J. P. Arnoletti, A. B. Benson, Y.-J. Chen, M. A. Choti, H. S. Cooper, A. Covey, R. A. Dilawari, D. S. Early, P. C. Enzinger *et al.*, “Colon cancer,” *Journal of the National Comprehensive Cancer Network*, vol. 7, no. 8, pp. 778–831, 2009.
- [35] T. J. Key, P. K. Verkasalo, and E. Banks, “Epidemiology of breast cancer,” *The lancet oncology*, vol. 2, no. 3, pp. 133–140, 2001.
- [36] H. Grönberg, “Prostate cancer epidemiology,” *The Lancet*, vol. 361, no. 9360, pp. 859–864, 2003.
- [37] K. Yao, G. Anagnostopoulos, and K. Ragunath, “Magnifying endoscopy for diagnosing and delineating early gastric cancer,” *Endoscopy*, vol. 41, no. 05, pp. 462–467, 2009.
- [38] D. Ojeda *et al.*, “Predictores de laringoscopía difícil,” *Rev Chil Anest*, vol. 41, pp. 179–87, 2012.
- [39] E. García Silveira, H. Pérez Cruz, N. Pérez Cruz, and D. M. Yera, “Historia de la broncoscopía,” *Revista Habanera de Ciencias Médicas*, vol. 7, no. 4, pp. 0–0, 2008.
- [40] R. B. Sierra, “Colonoscopia óptica y colonoscopia virtual. el papel de cada una hoy,” *Radiología*, vol. 57, no. 2, pp. 95–100, 2015.
- [41] J. D. Cohen, L. Li, Y. Wang, C. Thoburn, B. Afsari, L. Danilova, C. Douville, A. A. Javed, F. Wong, A. Mattox *et al.*, “Detection and localization of surgically resectable cancers with a multi-analyte blood test,” *Science*, vol. 359, no. 6378, pp. 926–930, 2018.
- [42] M. Villacampa, R. Moro, J. Naval, C. Failly-Crepin, F. Lampreave, and J. Uriel, “Alpha-fetoprotein receptors in a human breast cancer cell line,” *Biochemical and biophysical research communications*, vol. 122, no. 3, pp. 1322–1327, 1984.
- [43] H. Yokoo, T. Kondo, K. Fujii, T. Yamada, S. Todo, and S. Hirohashi, “Proteomic signature corresponding to alpha fetoprotein expression in liver cancer cells,” *Hepatology*, vol. 40, no. 3, pp. 609–617, 2004.
- [44] D. Chan, D. Bruzek, J. Oesterling, R. Rock, and P. Walsh, “Prostate-specific antigen as a marker for prostatic cancer: a monoclonal and a polyclonal immunoassay compared.” *Clinical chemistry*, vol. 33, no. 10, pp. 1916–1920, 1987.

- [45] R. Leborgne, “Diagnóstico de los tumores de la mama por la radiografía simple,” *Revista Cirugía del Uruguay*, vol. 20, no. 4, pp. 407–422, 1949.
- [46] F. Herranz Amo, F. Verdú Tartajo, and J. I. Martínez Salamanca, “Cáncer de próstata y ecografía transrectal,” *Archivos Españoles de Urología (Ed. impresa)*, vol. 59, no. 4, pp. 361–375, 2006.
- [47] P. Csendes and A. Sanhueza, “Evaluación del cáncer gástrico por tomografía computada,” *Revista chilena de radiología*, vol. 12, no. 2, pp. 57–63, 2006.
- [48] D. Saslow, C. Boetes, W. Burke, S. Harms, M. O. Leach, C. D. Lehman, E. Morris, E. Pisano, M. Schnall, S. Sener *et al.*, “American cancer society guidelines for breast screening with mri as an adjunct to mammography,” *CA: a cancer journal for clinicians*, vol. 57, no. 2, pp. 75–89, 2007.
- [49] J. E. FOWLER, S. A. BIGLER, D. MILES, and D. A. YALKUT, “Predictors of first repeat biopsy cancer detection with suspected local stage prostate cancer,” *The Journal of urology*, vol. 163, no. 3, pp. 813–818, 2000.
- [50] C. Criscitiello, F. André, A. M. Thompson, M. De Laurentiis, A. Esposito, L. Gelao, L. Fumagalli, M. Locatelli, I. Minchella, F. Orsi *et al.*, “Biopsy confirmation of metastatic sites in breast cancer patients: clinical impact and future perspectives,” *Breast Cancer Research*, vol. 16, no. 2, pp. 1–8, 2014.
- [51] A. M. Elsayad and H. Elsalamony, “Diagnosis of breast cancer using decision tree models and svm,” *International Journal of Computer Applications*, vol. 83, no. 5, 2013.
- [52] F. H. Schröder, I. van der CRUIJSEN-KOETER, H. J. de KONING, A. N. VIS, R. F. HOEDEMAEKER, and R. KRANSE, “Prostate cancer detection at low prostate specific antigen,” *The Journal of urology*, vol. 163, no. 3, pp. 806–812, 2000.
- [53] J. Chhatwal, O. Alagoz, M. J. Lindstrom, C. E. Kahn Jr, K. A. Shaffer, and E. S. Burnside, “A logistic regression model based on the national mammography database format to aid breast cancer diagnosis,” *American Journal of Roentgenology*, vol. 192, no. 4, pp. 1117–1127, 2009.
- [54] Z. Hu, J. Tang, Z. Wang, K. Zhang, L. Zhang, and Q. Sun, “Deep learning for image-based cancer detection and diagnosis- a survey,” *Pattern Recognition*, vol. 83, pp. 134–149, 2018.
- [55] A. Cruz-Roa, H. Gilmore, A. Basavanhally, M. Feldman, S. Ganesan, N. N. Shih, J. Tomaszewski, F. A. González, and A. Madabhushi, “Accurate and reproducible invasive breast cancer detection in whole-slide images: A deep learning approach for quantifying tumor extent,” *Scientific reports*, vol. 7, p. 46450, 2017.
- [56] M. E. Vandenberghe, M. L. Scott, P. W. Scorer, M. Söderberg, D. Balcerzak, and C. Barker, “Relevance of deep learning to facilitate the diagnosis of her2 status in breast cancer,” *Scientific reports*, vol. 7, no. 1, pp. 1–11, 2017.
- [57] I. Reda, A. Khalil, M. Elmogy, A. Abou El-Fetouh, A. Shalaby, M. Abou El-Ghar, A. Elmaghraby, M. Ghazal, and A. El-Baz, “Deep learning role in early diagnosis of prostate cancer,” *Technology in cancer research & treatment*, vol. 17, p. 1533034618775530, 2018.
- [58] A. R. Padhani and B. Turkbey, “Detecting prostate cancer with deep learning for mri: a small step forward,” 2019.
- [59] S. G. de la Orden, C. S. Requejo, and A. Q. Viqueira, “Situación epidemiológica del cáncer de próstata en España,” *Actas Urológicas Españolas*, vol. 30, no. 6, pp. 574–582, 2006.

- [60] M. L. Galán, A. P. Bordá, I. F. González, I. R. Cajigal, J. G. de Vicente, and A. B. Sánchez, “Efectos adversos de la biopsia prostática transrectal. un análisis de 303 procedimientos,” *Actas Urol Esp*, vol. 25, no. 1, pp. 46–49, 2001.
- [61] A. Heidenreich, M. Bolla, S. Joniau, M. Mason, V. Matveev, N. Mottet, H. Schmid, T. Van Der Kwast, T. Wiegel, and F. Zattoni, “Guía clínica sobre el cáncer de próstata,” *Asociación Europea de Urología*, 2010.
- [62] C. Allepuz Losa, Á. Borque Fernando, and L. Rioja Sanz, “¿ cuál es el algoritmo diagnóstico más apropiado para el cribado del cáncer de próstata?” *Actas Urológicas Españolas*, vol. 29, no. 5, pp. 465–472, 2005.
- [63] W. J. Catalona, D. S. Smith, and D. K. Ornstein, “Prostate cancer detection in men with serum psa concentrations of 2.6 to 4.0 ng/ml and benign prostate examination: enhancement of specificity with free psa measurements,” *Jama*, vol. 277, no. 18, pp. 1452–1455, 1997.
- [64] E. Arvaniti, K. S. Fricker, M. Moret, N. Rupp, T. Hermanns, C. Fankhauser, N. Wey, P. J. Wild, J. H. Rueschoff, and M. Claassen, “Automated gleason grading of prostate cancer tissue microarrays via deep learning,” *Scientific reports*, vol. 8, no. 1, pp. 1–11, 2018.
- [65] A. Rodríguez Alonso, S. Pertega Díaz, A. González Blanco, S. Pita Fernández, G. Suárez Pascual, and C. Pérez, “Utilidad de las redes neuronales artificiales en la predicción de cáncer de próstata en la biopsia transrectal,” *Actas Urológicas Españolas*, vol. 30, no. 1, pp. 18–24, 2006.
- [66] F. Milletari, N. Navab, and S.-A. Ahmadi, “V-net: Fully convolutional neural networks for volumetric medical image segmentation,” 2016.
- [67] Y. Gutiérrez, J. Arevalo, and F. Martínez, “A Ktrans deep characterization to measure clinical significance regions on prostate cancer,” in *15th International Symposium on Medical Information Processing and Analysis*, E. Romero, N. Lepore, and J. Brieva, Eds., vol. 11330, International Society for Optics and Photonics. SPIE, 2020, pp. 80 – 88. [Online]. Available: <https://doi.org/10.1117/12.2542606>
- [68] X. Xia, C. Xu, and B. Nan, “Inception-v3 for flower classification,” in *2017 2nd International Conference on Image, Vision and Computing (ICIVC)*. IEEE, 2017, pp. 783–787.
- [69] A. E. Tio, “Face shape classification using inception v3,” *arXiv preprint arXiv:1911.07916*, 2019.
- [70] X. Zhang, J. Zou, K. He, and J. Sun, “Accelerating very deep convolutional networks for classification and detection,” *CoRR*, vol. abs/1505.06798, 2015. [Online]. Available: <http://arxiv.org/abs/1505.06798>
- [71] A. F. Agarap, “Deep learning using rectified linear units (relu),” *CoRR*, vol. abs/1803.08375, 2018. [Online]. Available: <http://arxiv.org/abs/1803.08375>
- [72] Y. Xiao, J. Wu, Z. Lin, and X. Zhao, “A deep learning-based multi-model ensemble method for cancer prediction,” *Computer methods and programs in biomedicine*, vol. 153, pp. 1–9, 2018.

Apéndice A

Manual de usuario

Este manual de usuario especifica cada uno de los pasos que permiten instalar, compilar y manejar la aplicación desarrollada en este trabajo de fin de grado.

A.1. Requisitos previos

El sistema implementado en este trabajo se ha desarrollado en el lenguaje de programación Python en entorno Windows, en concreto se ha empleado Windows 10, por lo que se recomienda el uso de esa versión o una superior.

Para el funcionamiento del sistema desarrollado es necesario contar con un sistema operativo basado en Windows o Linux que incluya un compilador del lenguaje Python, así como con las librerías especificadas a continuación, cuya función se detalla.

- Python 3.7 se debe disponer de una versión de este lenguaje en el que se ha desarrollado el proyecto que sea compatible con el resto de librerías.
- Tensorflow 2.1.0 es una biblioteca de código abierto desarrollado por Google que permite desarrollar y entrenar modelos de aprendizaje automático.
- Keras 2.3.1 es una biblioteca de redes neuronales de código abierto desarrollada en Python con el fin de agilizar la experimentación con [DNN](#).
- Librerías Opencv 3.4.2: permiten realizar el procesamiento de las imágenes y datos, así como su visualización. Ésta es la mas importante de las librerías utilizadas ya que sus funciones se emplean en la programación del núcleo del algoritmo.
- Cuda Toolkit 10.1.243: permite acelerar el procesamiento mediante el uso del procesador gráfico.
- cudnn 7.6.5: se trata de una librería específica de CUDA para acelerar operaciones comunes de la [DNN](#).
- Pycharm 2019.2.1 Community Edition: se trata de un IDE especializado para Python que consta de versión de código abierto y que permite un mejor control del proyecto y el entorno.
- TexStudio 3.0: un editor de LaTeX de código abierto utilizado para generar esta memoria.

A.2. Estructura del programa

El programa desarrollado se encuentra en la carpeta *TFG-ProstateX*. En ella se encuentran todos los ficheros necesarios para la compilación y ejecución del mismo. Los más importantes son los siguientes.

- **codigo.py:** El archivo que se debe ejecutar para iniciar el sistema, alberga el flujo de ejecución principal, así como la definición de cada una de las arquitecturas. En el mismo se puede elegir la arquitectura a utilizar.
- **funciones.py:** Alberga todas aquellas funciones auxiliares que conciernen a la lectura de datos, su clasificación de los mismos, funciones generadoras para el entrenamiento y funciones de preprocesado de imágenes.
- **Archivos de datos:** Aquí diferenciamos entre los archivos de la base de datos original y los archivos binarios que se generan tras la primera lectura.
 - **Base de datos:** estos archivos mantienen la forma original de descarga. Encontramos las imágenes de [RMM](#) divididas en el archivo *ProstateXKtrans* que contiene las imágenes de modalidad Ktrans, y por otra parte una carpeta llamada *train* y otra *test* que contienen el resto de modalidades (ADC y T2-Ponderadas).
 - **Archivos binarios:** tras la primera ejecución se generarán dos archivos con extensión .pkl que almacenan los datos de entrada en una estructura dataframe uniforme, esto agiliza enormemente las posteriores ejecuciones reduciendo los accesos a disco.
- **Archivos .h5:** Estos archivos contienen los pesos guardados para un modelo específico tras el entrenamiento, por ello existe un archivo con el nombre de cada modalidad. Si al ejecutar el entrenamiento se establece un fichero con el mismo nombre de destino, este se sobreescribe, por ello para guardar diferentes entrenamientos se debe modificar el nombre del archivo cada vez.

A.3. Ejecución del programa

Para ejecutar el código utilizaremos nuestro IDE de Pycharm que nos facilita seleccionar el entorno con todas las librerías correctas. Para ello simplemente abriremos el archivo *codigo.py* con Pycharm y escogeremos el entorno correcto. En la figura A.1 se observa el código en el IDE Pycharm. Para ejecutar el código solo se deben configurar los parámetros de entrenamiento; epochs, learning_rate, batch_size y window. En la imagen se muestran los valores que se han valorados como óptimos durante el desarrollo. También se debe marcar el flag de que arquitectura queremos ejecutar, cambiando el correspondiente 0 por 1.

También será necesario ajustar la ruta de nuestra base de datos en el archivo *funciones.py*, para ello solo se debe modificar la variable global *path_entrada* que se observa en la figura A.2.

Una vez configurado esto podremos ejecutar el archivo *codigo.py*, si es la primera vez que ejecutamos este archivo o no contamos con los archivos binarios de datos, se realizará la lectura secuencial de la base de datos lo cual puede tardar aproximadamente 30-40 minutos, posteriormente se generaran con esa información los archivos binarios para evitar esta lectura secuencial en próximas ejecuciones. Tras ello comenzará la ejecución de la arquitectura que hayamos indicado.

```
File Edit View Navigate Code Refactor Run Tools VCS Window Help codige_jarvis_15112020.py [C:\Users\Javier\Desktop\LocalTemp\codige_jarvis_15112020] - C:\Users\Javier\Desktop\PyCharm\codige_jarvis\codige.py PyCharm
D:\Users\Javier\Desktop\codige_jarvis\codige.py
codige.py x funciones.py x
1# Jupyter
2
3from scipy import *
4import numpy as np
5import pandas as pd
6import os
7import matplotlib.pyplot as plt
8import SimpleITK as sitk
9from scikeras.classifiers import GaussianNB
10from sklearn import tree
11from sklearn import tree
12from sklearn.ensemble import RandomForestClassifier
13from sklearn import neighbors
14from IPython.display import clear_output
15from funciones_javi import*
16from tensorflow.keras.layers import LeakyReLU
17from sklearn.model_selection import train_test_split
18from sklearn import metrics
19import seaborn as sns
20
21from tensorflow.keras.layers import LeakyReLU
22from sklearn.model_selection import train_test_split
23from sklearn import metrics
24pd.options.display.max_columns = None
25pd.options.display.max_rows = None
26results = []
27
28epochs = 20
29learning_rate = 5e-6
30batch_size = 2
31window = 40
32KTRMS=1
33
34ARC4=0
35SOG0=0
36COR0=0
37TRB0=0
38FUSION1=0
39FUSION2=0
40
41
42training_MRI_imgs_dataframe, training_ktrans_imgs_dataframe = generate_training_dataframes()
43testing_MRI_imgs_dataframe, testing_ktrans_imgs_dataframe = generate_testing_dataframes()
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
```

Figura A.1: Archivo `codigo.py` abierto en Pycharm.

The screenshot shows a PyCharm interface with the following details:

- File Path:** codigos_para_javi_15112020.py [C:\Users\Javier\AppData\Local\Temp\codigos_para_javi_15112020.py] - C:\Users\Javier\Desktop\ifly-codigos\funciones.py - PyCharm
- Code Content:** The code is a Python script named "codigos_para_javi_15112020.py". It imports "funciones.py" and defines a function "def fill_MRI_images_dataframe_at_training(dataFrame, window)". The function iterates through rows of a DataFrame, extracts patient ID, series ID, and slice index, and constructs a path to an MRI image file. It then prints the condition for selecting sequences based on the patient ID and the requested sequence name.
- Toolbars and Menus:** Standard PyCharm menus like File, Edit, View, Navigate, Code, Refactor, Run, Tools, VCS, Window, Help are visible at the top.
- Status Bar:** Shows the current file as "codigos_para_javi_15112020.py" and the line number as "1".

Figura A.2: Archivo funciones.py abierto en Pycharm.

A.4. Resultados de la aplicación

Una vez el algoritmo haya sido iniciado, podremos observar los resultados de las diferentes épocas de entrenamiento en la salida por consola como se observa en la figura , tras cada época se muestra un pequeño resumen de algunas estadísticas para valorar el progreso de dicho entrenamiento.

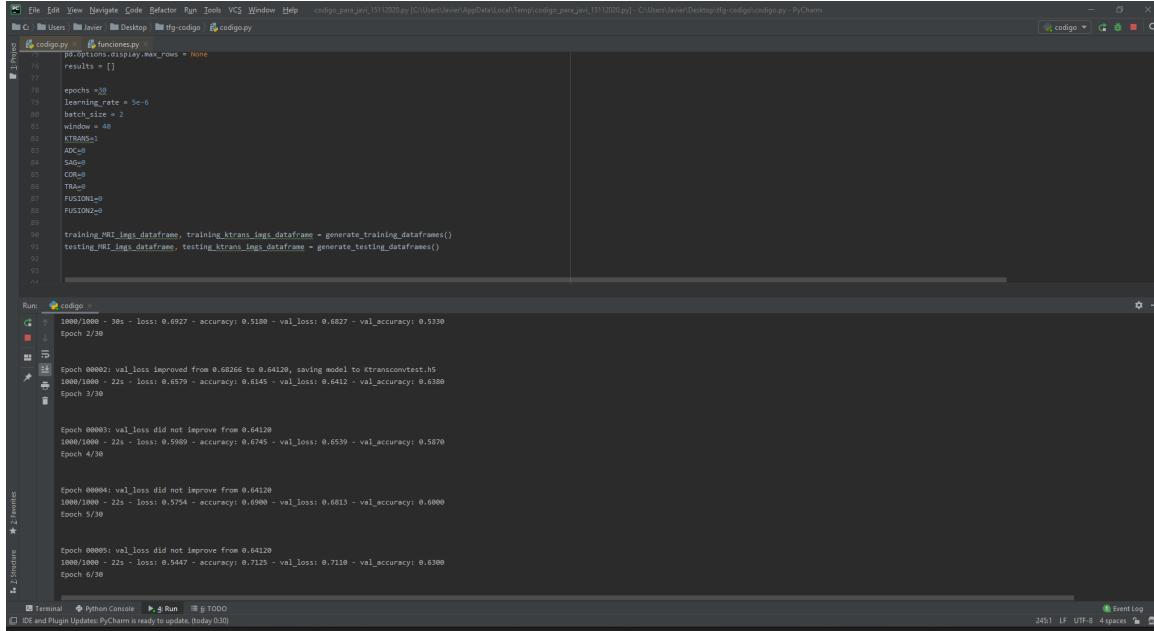


Figura A.3: Archivo funciones.py abierto en Pycharm.

Cuando finaliza obtenemos el primer resultado en forma de gráfica, la curva ROC, la cual aparecerá como una ventana emergente como se aprecia en la figura A.4. Tras cerrar esta ventana emergente obtenemos por consola el resto de métricas y en otra venta emergente un gráfico con la matriz de confusión como en la figura A.5.

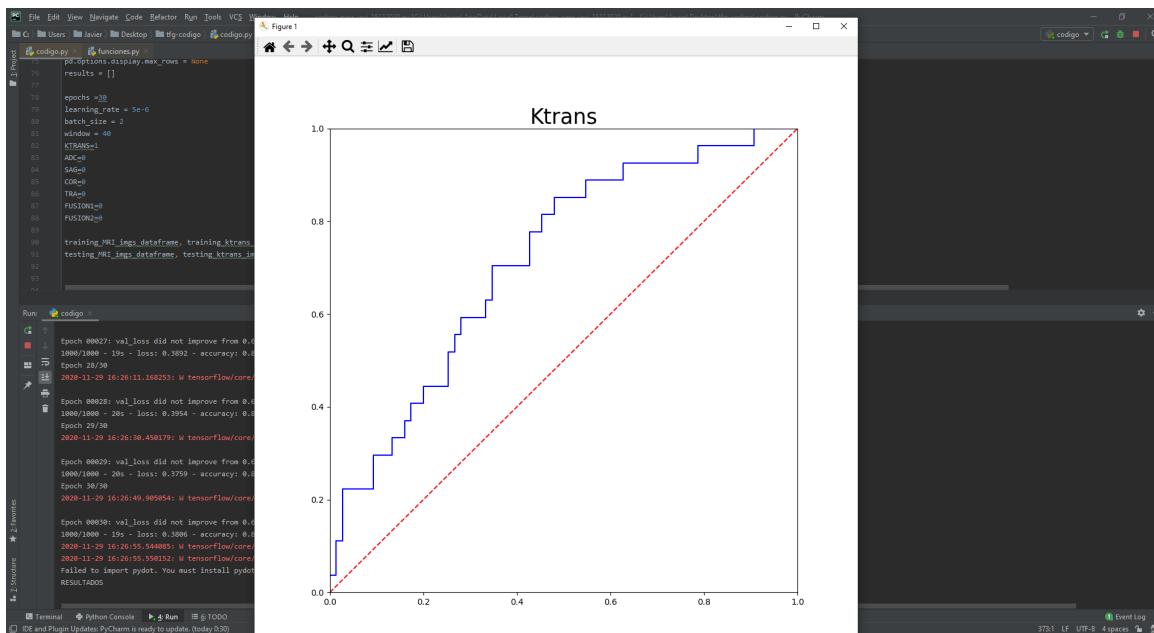


Figura A.4: Gráfica de curva ROC generada en ejecución.

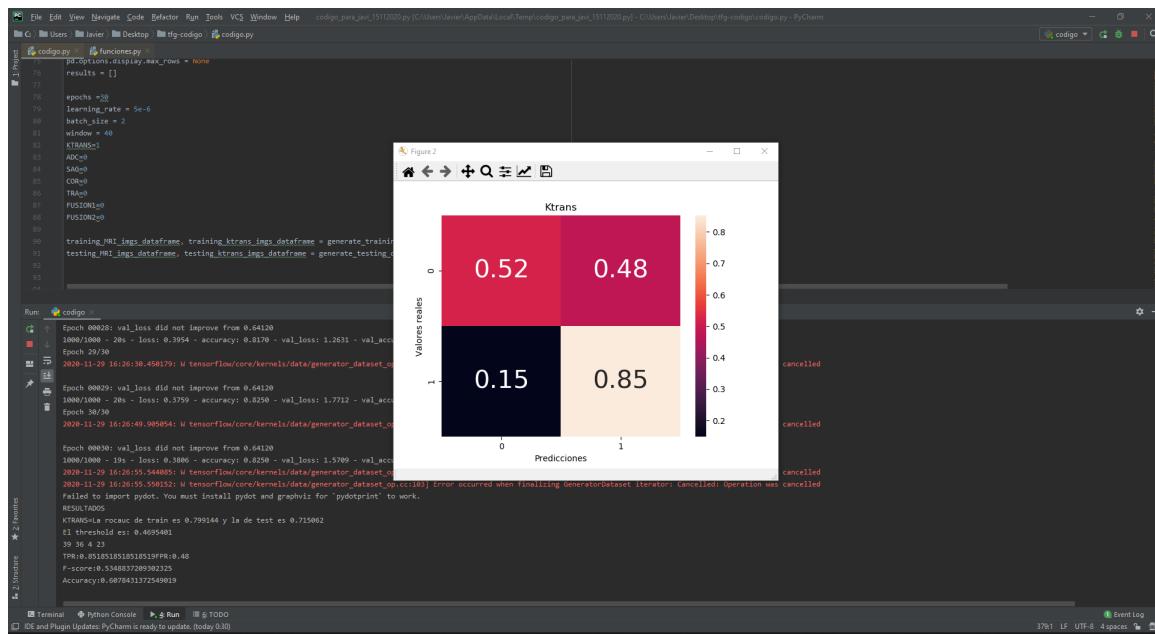


Figura A.5: Métricas finales y matriz de confusión.

Tras cerrar esta última ventana emergente si hemos marcado con 1 el flag de alguna otra arquitecturas, ésta comenzará a ejecutarse, de lo contrario finaliza la ejecución del programa.

Universidad de Alcalá
Escuela Politécnica Superior



ESCUELA POLITECNICA
SUPERIOR

