



**DISEÑO Y DESARROLLO DE UN SISTEMA AUTOMÁTICO PARA
LA DETECCIÓN DE TUMORES MELANOCÍTICOS SPITZOIDES
APLICANDO TÉCNICAS MULTI-RESOLUCIÓN BASADAS EN
DEEP LEARNING SOBRE WHOLE-SLIDE IMAGES.**

Francisco Javier Curieses Sanz

Tutora: Valery Naranjo Ornedo

Cotutora: M^o Rocio del Amor del Amor

Trabajo Fin de Máster presentado en la Escuela Técnica Superior de Ingeniería de Telecomunicación de la Universitat Politècnica de València, para la obtención del Título de Máster en Ingeniería Telecomunicación.

Curso académico 2020-21

Valencia, 12 de Septiembre de 2021

Agradecimientos

“ A Rocio, por toda su ayuda y dedicación durante el desarrollo de este trabajo, ya que sin ella esto no hubiera sido posible.

A Valery, por darme la oportunidad de realizar este proyecto.

A mis abuelos, por estar siempre ahí y apoyarme día tras día con sus: ”venga, un último esfuerzo”.

A mi familia, por confiar en mi y estar orgullosos, aunque no terminen de entender que es lo que hago.

A mis amigos, que son una fuente de energía y vitalidad constante.

En definitiva, gracias a todas las personas que me apoyáis y hacéis que poco a poco logre mis metas.”

Resumen

El cáncer de piel es una de las neoplasias más comunes entre la población de piel blanca. Aunque las lesiones cutáneas más comunes son el carcinoma de células basales y el de células escamosas, el cáncer de melanoma es el más peligroso debido al alto riesgo metastásico. Las lesiones melanocíticas presentan una alta complejidad genética y biológica. En su clasificación se distingue entre lesiones benignas (nevus melanocíticas) y malignas (melanoma). Sin embargo, existen neoplasias de difícil diagnóstico. Entre estas lesiones de difícil clasificación, se encuentran los tumores spitzoides o neoplasias melanocíticas spitzoides, que presentan un aspecto morfológico ambiguo. Para la evaluación de estas lesiones, los patólogos emplean técnicas de histopatología, que permiten digitalizar la muestra del paciente. En el análisis manual de la muestra, los especialistas emplean distintas resoluciones, lo que permite obtener una mayor información sobre la neoplasia. Este proceso cuenta con una alta componente de subjetividad, ya que depende en gran medida del especialista. Es por ello que, son necesarios sistemas de análisis automático que ayuden a la clasificación de este tipo de lesiones.

En el presente trabajo fin de máster, se desarrollan algoritmos automáticos de clasificación basados en *deep learning*, aplicando técnicas de multiresolución, para la clasificación de imágenes WSI (*Whole-Slide Images*) con lesiones melanocíticas spitzoides. Para este fin, se elabora un *data set* con las imágenes a utilizar en los distintos entrenamientos. Posteriormente, se desarrollan algoritmos para entrenar redes neuronales con imágenes a una única resolución y redes híbridas con imágenes a distintas resoluciones. En la elaboración de redes multiresolución, donde se utiliza la técnica *late fusión*, se simula el proceso de análisis que emplea un patólogo en el diagnóstico de las muestras. Por último, se realiza una comparación entre los modelos a una única resolución y a multiresolución.

Palabras clave: neoplasias melanocíticas spitzoides, *Whole-Slide Images*, *deep learning*, *Convolutional Neural Networks* (CNN), *transfer learning*, multiresolución.

Resum

El càncer de pell és una de les neoplàsies més comunes entre la població de pell blanca. Encara que les lesions cutànies més comunes són el carcinoma de cèl·lules basals i el de cèl·lules escatoses, el càncer de melanoma és el més perillós a causa de l'alt risc metastàtic. Les lesions melanocítiques presenten una alta complexitat genètica i biològica. En la seua classificació es distingeix entre lesions benignes (nevus melanocítiques) i malignes (melanoma). No obstant això, existeixen neoplàsies de difícil diagnòstic. Entre aquestes lesions de difícil classificació, es troben els tumors spitzoides o neoplàsies melanocítiques spitzoides, que presenten un aspecte morfològic ambigu. Per a l'avaluació d'aquestes lesions, els patòlegs empren tècniques d'histopatologia, que permeten digitalitzar la mostra del pacient. En l'anàlisi manual de la mostra, els especialistes empren diferents resolucions, la qual cosa permet obtenir una major informació sobre la neoplàsia. Aquest procés compta amb una alta component de subjectivitat, ja que depèn en gran manera de l'especialista. És per això que, són necessaris sistemes d'anàlisi automàtica que ajuden a la classificació d'aquesta mena de lesions.

En el present treball fi de màster, es desenvolupen algorismes automàtics de classificació basats en *deep learning*, aplicant tècniques de multiresolució, per a la classificació d'imatges WSI (*Whole-Slide Images*) amb lesions melanocítiques spitzoides. Per a aquest fi, s'elabora un *data set* amb les imatges a utilitzar en els diferents entrenaments. Posteriorment, es desenvolupen algorismes per a entrenar xarxes neuronals amb imatges a una única resolució i xarxes híbrides amb imatges a diferents resolucions. En l'elaboració de xarxes multiresolució, on s'utilitza la tècnica *batega fusió*, se simula el procés d'anàlisi que empraria un patòleg en el diagnòstic de les mostres. Finalment, es realitza una comparació entre els models a una única resolució i a multiresolució.

Paraules clau: neoplàsies melanocítiques spitzoides, *Whole-Slide Images*, *deep learning*, *Convolutional Neural Networks* (CNN), *transfer learning*, multiresolució.

Abstract

Skin cancer is one of the most common neoplasms among the fair-skinned population. Although the most common skin lesions are basal cell and squamous cell carcinoma, melanoma cancer is the most dangerous due to the high metastatic risk. Melanocytic lesions present a high genetic and biological complexity. In their classification, a distinction is made between benign (melanocytic nevi) and malignant (melanoma) lesions. However, there are neoplasms that are difficult to diagnose. Among these lesions that are difficult to classify are spitzoid tumors or spitzoid melanocytic neoplasms, which have an ambiguous morphological appearance. For the evaluation of these lesions, pathologists use histopathology techniques, which allow the patient's specimen to be digitized. In the manual analysis of the specimen, specialists use different resolutions, which allows them to obtain more information about the neoplasm. This process has a high component of subjectivity, since it depends to a great extent on the specialist. For this reason, automatic analysis systems are necessary to help in the classification of this type of lesions.

In this master's thesis, automatic classification algorithms based on deep learning, applying multiresolution techniques, are developed for the classification of WSI images (Whole-Slide Images) with spitzoid melanocytic lesions. For this purpose, a data set with the images to be used in the different trainings is elaborated. Subsequently, algorithms are developed to train neural networks with images at a single resolution and hybrid networks with images at different resolutions. In the development of multiresolution networks, where the late fusion technique is used, the analysis process used by a pathologist in the diagnosis of the samples is simulated. Finally, a comparison is made between single-resolution and multiresolution models.

Keywords: spitzoid melanocytic neoplasms, Whole-Slide Images, deep learning, Convolutional Neural Networks (CNN), transfer learning, multiresolution.

Índice general

I Memoria

| | | |
|----------|--|-----------|
| 1 | Introducción | 1 |
| 1.1 | Introducción | 1 |
| 1.2 | Motivación | 2 |
| 1.3 | Estructura del documento | 2 |
| 2 | Marco del proyecto y objetivos | 5 |
| 2.1 | Marco del proyecto | 5 |
| 2.2 | Objetivos | 5 |
| 3 | Marco teórico | 7 |
| 3.1 | Neoplasias melanocíticas. Tumores spitzoides | 7 |
| 3.1.1 | Neoplasias melanocíticas | 7 |
| 3.1.2 | Tumores melanocíticos spitzoides | 7 |
| 3.2 | Proceso de análisis de imágenes histopatológicas | 8 |
| 3.2.1 | Proceso histológico | 9 |
| 3.2.2 | Propiedades morfológicas de los tumores melanocíticos spitzoides | 9 |
| 3.2.3 | Digitalización de las muestras | 12 |
| 4 | Metodología | 13 |
| 4.1 | Creación de la base de datos | 13 |
| 4.1.1 | Anotación y etiquetado de las regiones de interés | 13 |
| 4.1.2 | Parcheo de las imágenes a distintas resoluciones | 14 |
| 4.1.3 | Selección de las regiones de interés | 14 |
| 4.1.4 | Relación de parches a distinta resolución | 15 |
| 4.1.5 | Partición de los datos | 18 |
| 4.2 | Algoritmos de clasificación basados en <i>deep learning</i> | 20 |
| 4.2.1 | Redes neuronales convolucionales (CNN) | 21 |
| 4.2.1.1 | <i>Base model</i> de una red CNN | 21 |
| 4.2.1.2 | <i>Top model</i> de una red CNN | 22 |
| 4.2.2 | Técnica de <i>transfer learning</i> | 23 |
| 4.3 | Redes neuronales híbridas | 25 |
| 4.3.1 | Redes neuronales multiresolución | 25 |
| 5 | Implementación y resultados | 29 |
| 5.1 | Implementación de modelos de <i>deep learning</i> | 29 |
| 5.1.1 | Implementación de modelos a una única resolución | 29 |

| | | |
|-----------|---|-----------|
| 5.1.2 | Desarrollo de modelos a multiresolución | 31 |
| 5.2 | Métricas para la validación de los modelos | 32 |
| 5.3 | Experimentación, resultados y discusión | 34 |
| 5.3.1 | Fase de optimización: experimentos realizados | 34 |
| 5.3.2 | Fase de testeo | 43 |
| 6 | Conclusiones y líneas futuras | 47 |
| 6.1 | Conclusiones | 47 |
| 6.2 | Líneas futuras | 48 |
| | Bibliografía | 49 |
| | | |
| II | Presupuesto | |
| 1 | Presupuesto | 55 |
| 1.1 | Objetivo | 55 |
| 1.2 | Presupuesto parcial | 55 |
| 1.2.1 | Coste de personal | 55 |
| 1.2.2 | Coste <i>hardware</i> | 56 |
| 1.2.3 | Coste de <i>software</i> | 56 |
| 1.3 | Presupuesto total | 57 |

Índice de figuras

| | | |
|------|---|----|
| 3.1 | Taxonomía de las neoplasias melanocíticas. (a) Neoplasias melanocíticas derivadas de melanocitos asociados al epitelio. (b) Neoplasias melanocíticas derivadas de melanocitos no asociados al epitelio. Figura extraída de [10] | 8 |
| 3.2 | Proceso histológico. Figura extraída de [12]. | 9 |
| 3.3 | Muestra de tejido con melanoma. A la izquierda imagen a una resolución 5x, y a la derecha con una resolución 20x. Elaboración propia. | 10 |
| 3.4 | Imágenes WSI pertenecientes a la base de datos utilizada para el desarrollo del presente trabajo. Las muestras presentan una resolución 10x. Cada una de las imágenes está diagnosticada con un tipo distinto de lesión. A: Nevus Spitz (NS); B: Tumor de potencial maligno incierto (STUMP); C: Melanoma spitzoide (MS). Elaboración propia. | 10 |
| 3.5 | Composición de una imagen WSI histopatológica. | 12 |
| 4.1 | Parqueo de las imágenes (resolución 5x y 50 % de solapamiento). Elaboración propia. | 14 |
| 4.2 | Relación entre un parche a resolución 5x, 10x y 20x. Elaboración propia. | 18 |
| 4.3 | Estructura de la partición de los datos siguiente la técnica <i>k-fold</i> con $k = 4$. Elaboración propia. | 19 |
| 4.4 | Operación de convolución dentro de una capa convolucional. Figura extraída de [24]. | 21 |
| 4.5 | Gráfica de la función de activación ReLu. Figura extraída de [25]. | 22 |
| 4.6 | Proceso de <i>pooling</i> . A la izquierda, un mapa de características de tamaño 224x224x64 se sub-muestra con un filtro de tamaño 2, pasando a un volumen de 112x112x64. A la derecha, el proceso de <i>pooling</i> sobre un <i>slice</i> de 4x4x1, aplicando <i>max pooling</i> con un filtro de 2x2. Figura extraída de [26]. | 23 |
| 4.7 | Diferencias entre la capa <i>global pooling</i> y la capa <i>flatten</i> . | 24 |
| 4.8 | Representación gráfica de una capa <i>fully connected</i> en un <i>top model</i> . Figura extraída de [27]. | 24 |
| 4.9 | Red neuronal convolucional (CNN) genérica completa. Figura extraída de [23]. | 25 |
| 4.10 | Red neuronal convolucional multiresolución, basada en 3 entradas de datos a resoluciones 5x, 10x y 20x. Elaboración propia. | 26 |
| 4.11 | Red neuronal convolucional multiresolución, siguiendo el proceso de <i>late fusion</i> . Figura extraída de [32]. | 27 |
| 5.1 | Interfaz gráfica de la herramienta <i>Deep Network Designer</i> de MATLAB. En la herramienta se encuentra cargada una red multiresolución. Imagen extraída de MATLAB R2021a. | 31 |
| 5.2 | Ejemplo de las 11 primeras capas una red multiresolución formada por tres redes RESNET18. Elaboración propia. | 32 |

| | | |
|------|--|----|
| 5.3 | Relación entre los valores reales y los valores predecidos. Elaboración propia. . . | 33 |
| 5.4 | Arquitectura de la red neural VGG16. Modificación de [37]. | 35 |
| 5.5 | Arquitectura de la red neuronal <i>RESNET18</i> . Figura extraída de [38] | 36 |
| 5.6 | Arquitectura red neural Inceptionv3. Fuente: [39]. | 37 |
| 5.7 | Gráficas del entrenamiento del mejor <i>accuracy</i> logrado con RESNET18. | 38 |
| 5.8 | Arquitectura de la red RESNET18 multiresolución. | 43 |
| 5.9 | Gráficas del entrenamiento del mejor modelo a multiresolución. | 44 |
| 5.10 | Curvas ROC del los distintos entrenamientos del modelo a multiresolución. . . . | 45 |
| 5.11 | Matriz de confusión de los datos de validación, en cada uno distintos entrenamientos del modelo a multiresolución. | 45 |
| 5.12 | Matriz de confusión de los datos de test empleando los modelos multiresolución. | 46 |

Índice de tablas

| | | |
|-----|--|----|
| 3.1 | Características clínicas de los tumores nevus Spitz y tumores de Spitz atípicos. | 11 |
| 3.2 | Características clínicas de los tumores nevus Spitz y melanomas spitzoides. | 11 |
| 4.1 | Partición de los datos para los entrenamientos a resoluciones 5x, 10x y 20x. | 19 |
| 4.2 | Partición de los datos para la validación a resoluciones 5x, 10x y 20x. | 19 |
| 4.3 | Partición de los datos para el entrenamiento a multiresolución. | 20 |
| 4.4 | Partición de los datos para la validación a multiresolución. | 20 |
| 4.5 | Número total de imágenes utilizadas en el entrenamiento y validación de la red neuronal multiresolución. | 20 |
| 5.1 | Resultados del mejor modelo de RESNET18 durante la clasificación del set de validación, resolución 5x. | 39 |
| 5.2 | Resultados de clasificación del set de validación con RESNET18, resolución 10x. | 40 |
| 5.3 | Resultados de clasificación del set de validación con RESNET18, resolución 20x. | 40 |
| 5.4 | Resultados de la clasificación del set de validación, del modelo RESNET18 multiresolución. | 42 |
| 5.5 | Comparativa de la clasificación del set de validación con las redes RESNET18, empleando resolución 5x y multiresolución. | 42 |
| 5.6 | Métricas obtenidas del testeo de los distintos modelos elaborados. | 46 |
| 5.7 | Resultados de la clasificación a nivel de paciente empleando el modelo multiresolución. | 46 |
| 1.1 | Costes de personal asociados al desarrollo del presente proyecto. | 56 |
| 1.2 | Costes <i>hardware</i> asociados al desarrollo del presente proyecto. | 57 |
| 1.3 | Costes <i>software</i> asociados al desarrollo del presente proyecto. | 57 |
| 1.4 | Presupuesto de Ejecución de Material del proyecto. | 58 |
| 1.5 | Presupuesto total del proyecto. | 58 |

Listado de siglas empleadas

| | |
|-------|--|
| WSI | Whole-Slide Image |
| NS | Spitz Nevus |
| AST | Atypical Spitz Tumor |
| MS | Melanoma spitzoide |
| STUMP | Tumor spitzoide de potencial maligno incierto |
| CNN | Potencial maligno incierto |
| RRBS | Convolutional Neuronal Network (Red neuronal convolucional) |
| NAS | Network-attached storage (Almacenamiento conectado a la red) |
| GAP | Global Average Pooling |
| GMP | Global Max Pooling |
| FC | Fully Connected Layer (Capa completamente conectada) |

Parte I

Memoria

Capítulo 1

Introducción

1.1. Introducción

El cáncer de piel es el uno de los más extendidos entre la población de piel blanca. Entre estos destacan el carcinoma de células basales y escamosas, y el melanoma [1]. Aunque el melanoma no sea el cáncer de piel más común, es el más agresivo debido a su alto potencial de desarrollo metastásico. La tasa de casos de melanoma en España se sitúa en 12 casos por cada 100.000 habitantes, habiendo diagnosticados en 2020 un total de 2.489 nuevos casos. Además, cada año se diagnostican en todo el mundo 3 millones de casos de cáncer de piel no melanoma, y 132.000 casos de melanoma [2].

La gran mayoría de las lesiones cutáneas, se producen debido a la radiación ultravioleta (UV). Las neoplasias melanocíticas se clasifican en diferentes tipos según sus características biológicas y morfológicas. En la mayoría de los casos es posible clasificar las lesiones entre benignas (nevus melanocíticas) y malignas (melanoma). Sin embargo, en ocasiones existen neoplasias de difícil diagnóstico debido a sus características histológicas. Estas lesiones son conocidas como tumores spitzoides, los cuales no pueden ser clasificados como lesiones benignas, pero tampoco como malignas. Dentro de este grupo existen lesiones benignas como el nevus Spitz, lesiones malignas como el melanoma spitzoide o neoplasias con potencial maligno incierto donde su clasificación se encuentra entre la de las lesiones de Spitz benignas y malignas. Algunas lesiones de melanoma presentan proliferaciones melanocíticas fusiformes y/o epiteliales muy similares al nevus Spitz, lo que hace muy complejo su diagnóstico [3]. Una correcta clasificación de este tipo de lesiones es fundamental, ya que las lesiones de melanoma pueden llegar a desarrollar metástasis, por lo que es muy importante su detección en etapas tempranas de desarrollo [4].

En el proceso de categorización de las lesiones, los patólogos analizan las muestras del paciente obtenidas a través del proceso de histopatología, mediante el cual se digitaliza la biopsia obtenida del sujeto. En este análisis, se presta atención a rasgos histológicos de los tumores melanocíticos spitzoides, como por ejemplo las mitosis que presenta la muestra, la simetría citológica o el ratio núcleo citoplasma, entre otros. Este proceso es bastante subjetivo, ya que cada paciente muestra propiedades biológicas muy distintas, y es por ello que, se necesitan mecanismos de clasificación que ayuden al patólogo a realizar un diagnóstico lo más preciso posible.

Para clasificar correctamente las muestras histológicas, el especialista realiza un análisis en profundidad de la muestra, aumentando y disminuyendo el zoom de la imagen, obteniendo de esta

forma la mayor información posible. Esto se realiza debido a que cada resolución aporta información diferente acerca de la lesión. Con menores resoluciones de la imagen, se obtiene información contextual de la lesión, mientras que a resoluciones mayores, se obtiene información sobre los detalles más finos de las imágenes. Con la finalidad de simular este análisis, es posible desarrollar algoritmos basados en *deep learning* que simulen este proceso [5].

Con la avance de los modelos de inteligencia artificial y la computación en la nube, son cada vez más los desarrollos que se están realizando para el estudio de imágenes médicas, proporcionando una herramienta de ayuda al especialista a la hora de evaluar lesiones [6], [7].

Debido a la alta complejidad de clasificación de este tipo de lesiones, como son las neoplasias melanocíticas spitzoides, y el peligro que poseen los tumores de melanoma, se pretende hacer uso de la inteligencia artificial para desarrollar modelos de clasificación automática de imágenes que ayuden al patólogo en su diagnóstico. Durante los siguientes puntos de este documento, se detalla el trabajo realizado para conseguir desarrollar y comparar distintos modelos de redes neuronales que satisfacen lo comentado.

1.2. Motivación

Debido a la gran dificultad de diagnóstico que presentan las neoplasias melanocíticas spitzoides, no existen estudios que empleen métodos de clasificación automáticos basados en multiresolución para su diagnóstico. Es por ello que, este trabajo fin de máster pretende estudiar las posibles mejoras que ofrecen este tipo de redes neuronales respecto las redes a una única resolución.

Existen diversos estudios sobre el empleo de técnicas multiresolución para la clasificación de imágenes histopatológicas WSI. En el trabajo de Jiayun Li et al., [5], se desarrolla un modelo de clasificación automático de imágenes prostáticas histopatológicas, basado en redes neuronales que emplean técnicas multiresolución. En su desarrollo, se utilizan dos etapas de procesado, una para imágenes 5x y otra para imágenes 10x. Finalmente, se obtiene una mejora del 7.31 % respecto a la utilización de redes con una sola resolución. Otro ejemplo es el trabajo desarrollado por Mart van Rijthoven et al., [8], donde se desarrolla una red neuronal denominada HookNet, cuya finalidad es la segmentación de imágenes WSI histopatológicas. En dicha publicación, los autores emplean imágenes histopatológicas de pecho y pulmón, de cánceres benignos y malignos. La red multiresolución ofrece una mejora del 15 % en la segmentación de muestras benignas y un 3 % en la segmentación de muestras malignas, respecto a la red con una única resolución.

A la vista de los resultados que ofrecen las redes neuronales multiresolución, en la clasificación de imágenes histopatológicas, en el presente trabajo se motiva a aplicar estas técnicas multiresolución a imágenes de tumores melanocíticos spitzoides, con la finalidad de obtener una mejora en su clasificación. El presente trabajo pretende servir de entrada al desarrollo de nuevos modelos de *deep learning* basados en multiresolución para la detección de tumores melanocíticos spitzoides. Con estos avances se podrán desarrollar modelos sofisticados que ayuden al patólogo en su trabajo, pudiendo realizar diagnósticos mucho más precisos.

1.3. Estructura del documento

El resto del documento se organiza de la forma que se describe a continuación.

En el capítulo 2 se detalla el marco en el cual está englobado el proyecto y los objetivos del mismo.

En el capítulo 3 se describen las neoplasias melanocíticas y las características morfológicas y biológicas que presentan, centrándose en los tumores melanocíticos spitzoides. Además, se detalla el proceso histopatológico, el cual permite al especialista analizar las muestras obtenidas de los pacientes bajo estudio.

El capítulo 4 recoge la metodología empleada para el desarrollo del trabajo y sus distintas etapas, desde la creación de la base de datos, hasta los distintos tipos de redes neuronales y algoritmos de *deep learning* empleados. Se detalla que son las redes neuronales a multiresolución y que beneficio conlleva su uso en la clasificación de WSI.

En el capítulo 5 se explican todos los entrenamientos realizados con las redes neuronales a resolución 5x, 10x, 20x y multiresolución. Además, se detallan los resultados de validación de las redes desarrolladas, así como los de testeo en los que se utilizan nuevas imágenes hasta ahora desconocidas por la red.

Por último, en el capítulo 6 se resumen las conclusiones extraídas del trabajo realizado. Además, se proponen nuevas líneas futuras de desarrollo del proyecto.

Capítulo 2

Marco del proyecto y objetivos

2.1. Marco del proyecto

El presente trabajo se enmarca dentro del proyecto de investigación CLARIFY¹ (*Cloud Artificial Intelligence For Pathology*). CLARIFY es un programa de investigación e innovación, que vincula la ingeniería y la medicina, combinando tecnologías como la inteligencia artificial (IA) y la computación en la nube (*cloud computing*) para su aplicación en la patología digital.

El objetivo principal de CLARIFY es desarrollar un entorno *cloud* de diagnóstico digital automatizado, basado en inteligencia artificial. Esta plataforma estará orientada a facilitar la interpretación y el diagnóstico de imágenes WSI, con el objetivo de maximizar los beneficios de la patología digital y ayudar a los patólogos en su trabajo diario.

CLARIFY es un proyecto subvencionado por la Unión Europea a través del programa de investigación e innovación Horizonte 2020. Este proyecto se encuentra coordinado por el grupo de investigación CVBLab (*Computer Vision and Behaviour Analysis Lab*) perteneciente a la Universidad Politécnica de València (UPV).

El grupo CVLab tiene la labor, dentro de este proyecto, de desarrollar algoritmos automáticos basados en inteligencia artificial. Estos algoritmos serán aplicados sobre las imágenes WSI obtenidas y datos de estudios moleculares. Dentro de este grupo, el presente trabajo tiene como fin el desarrollo de algoritmos de clasificación automáticos basados en multiresolución, que permitan mejorar el diagnóstico de una muestra histopatológica, al emplear distintos niveles de detalle.

2.2. Objetivos

El objetivo principal de este trabajo fin de máster, es el diseño y desarrollo de un sistema automático para la detección de tumores melanocíticos spitzoides. Este sistema se desarrollará aplicando técnicas multiresolución basadas en *deep learning* sobre imágenes WSI (*Whole-Slide Images*). Para lograr este objetivo principal, este trabajo se ha dividido en las siguientes tareas:

1. Estudio del estado del arte en el ámbito de técnicas de clasificación automáticas sobre imáge-

¹Más información acerca del proyecto CLARIFY en <http://www.clarify-project.eu/>.

nes WSI. En concreto, se ha de investigar el empleo de redes neuronales multiresolución para la clasificación de estas imágenes. También se ha de estudiar las principales características de las neoplasias melanocíticas spitzoides.

2. Creación y adaptación de la base de datos. Para ello, se han de adaptar las imágenes WSI, generando los parches a distintas resoluciones y separando las regiones de interés anotadas por el especialista.
3. Relacionar los parches que pertenecen a la misma zona de la imagen original, es decir, los parches de mayor resolución que están contenidos en parches de menor resolución.
4. Aplicar técnicas de *transfer learning*, a redes neuronales pre-entrenadas, con los datos de las imágenes histopatológicas.
5. Creación de redes neuronales híbridas con tres entradas diferentes, es decir, una entrada para los datos a una resolución 5x, otra a 10x y otra a 20x. Una vez creada la red neuronal se ha de entrenar con la base de datos creada en el punto 3.
6. Comparación entre los resultados obtenidos con los modelos a una única resolución y multiresolución.
7. Elaboración de las conclusiones y posibles líneas de desarrollo futuro.

Capítulo 3

Marco teórico

3.1. Neoplasias melanocíticas. Tumores spitzoides

3.1.1. Neoplasias melanocíticas

Los melanocitos, células dendríticas cuya función es la producción de melanina [9] pueden dar lugar a diversas neoplasias con distinta distribución anatómica. Estas neoplasias surgen con mayor frecuencia en los melanocitos de la piel, pero también pueden aparecer en diversos órganos internos [10].

Las neoplasias melanocíticas pueden estar derivadas de melanocitos asociados al epitelio (capa delgada de tejido que cubre los órganos, las glándulas y otras estructuras dentro del cuerpo [11]) o de melanocitos que no están asociados al epitelio. Las neoplasias melanocíticas asociadas al epitelio, presentan distintas clases de lesiones en función de la radiación ultravioleta (UV), y una distribución en función de la edad. En este grupo, se encuentran las neoplasias melanocíticas spitzoides, las cuales se hablará más en detalle a continuación. Por otro lado, se encuentran las neoplasias melanocíticas no asociadas al epitelio, y que sus distintas clases dependen de la zona corporal donde se encuentre la lesión (ojos, órganos internos o piel) [10]. En la Figura 3.1 se puede observar una gráfica con la clasificación de los distintos tipos de neoplasias melanocíticas, así como sus clases.

La mayoría de los tumores melanocíticos presentan unas características microscópicas y genéticas que permiten su clasificación en benignos y malignos. Sin embargo, en ocasiones, existen neoplasias melanocíticas que presentan un alto grado de dificultad en el diagnóstico, lo que ocasiona un insuficiente diagnóstico como nevus, a un sobrediagnóstico como melanoma o a un diagnóstico de "tumor de potencial maligno incierto", o "límite tumor melanocítico". Dentro de este grupo, altamente difícil de clasificar, se encuentran los tumores melanocíticos spitzoides [12].

3.1.2. Tumores melanocíticos spitzoides

Las neoplasias spitzoides son tumores melanocíticos con una alta variabilidad biológica. Existen lesiones benignas como el nevus Spitz (NS), neoplasias con potencial maligno incierto como los tumores de Spitz atípicos (AST), y lesiones malignas como el melanoma spitzoide (MS). En 1948, Shopie Spitz describió por primera vez una serie de casos, a los que llamó "melanoma juvenil", cu-

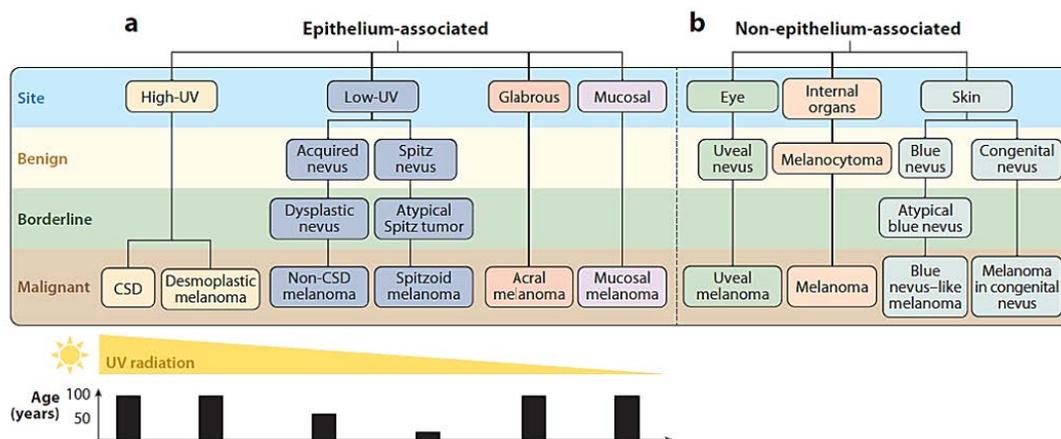


Figura 3.1: Taxonomía de las neoplasias melanocíticas. (a) Neoplasias melanocíticas derivadas de melanocitos asociados al epitelio. (b) Neoplasias melanocíticas derivadas de melanocitos no asociados al epitelio. Figura extraída de [10]

ya apariencia histopatológica era similar al melanoma maligno, pero con un pronóstico favorable. No fue hasta 1989, cuando Smith *et al.* [13], denominó un grupo de nevus Spitz que histopatológicamente no se podían clasificar como melanoma, pero que presentaban un comportamiento agresivo, llegando a originar metástasis. Más tarde este tipo de neoplasias paso a denominarse tumor melanocítico spitzoide [14].

Los tumores melanocíticos spitzoides (MS), representan la parte maligna de los NS y se caracterizan morfológicamente por grandes melanocitos fusiformes y/o melanocitos epitelioidales. Las lesiones que presentan características genéticas y/o fisiopatológicas intermedias se denominan tumores de Spitz atípicos (AST). Estos últimos se han clasificado como "tumor spitzoide de potencial maligno incierto"(STUMP), y son más comunes en grupos de mayor edad. [15].

En la Figura 3.4 se puede visualizar, de izquierda a derecha, muestras de tejido catalogado como nevus Spitz (NS), tumor spitzoide de potencial maligno incierto (STUMP) y melanoma spitzoide (MS), respectivamente.

La correcta clasificación de los tumores de Spitz como NS, AST o MS es todo un reto para los especialistas, es por ello que se han de diseñar sistemas de clasificación automática que ayuden al patólogo en su diagnóstico.

3.2. Proceso de análisis de imágenes histopatológicas

Para poder realizar un diagnóstico de la posible lesión cutánea que presenta el paciente, es necesario extraer muestras de tejido extraída mediante un proceso conocido como biopsia. Tras realizar la biopsia, ese bloque de tejido ha de ser tratado para facilitar su inspección, lo que se conoce como técnica histológica. Finalmente, esa muestra puede ser digitalizada, permitiendo un mejor estudio por parte del especialista. A continuación se detalla el proceso histológico y qué partes de la muestra resultan de interés.

3.2.1. Proceso histológico

Para poder realizar un diagnóstico fiable, el patólogo necesita la mayor resolución posible de la muestra tumoral biopsiada del paciente. Es por ello, que se recurre a microscopios ópticos, electrónicos o sistemas digitales para analizar la muestra obtenida durante este proceso, a continuación descrito [12]:

1. **Fijación de la muestra:** Se realiza tras la extracción del tejido con el fin de inmovilizar y conservar la muestra en el mejor estado posible, deteniendo la vida de las células de tejido. Puede realizarse mediante una solución fijadora o por congelación.
2. **Inclusión:** Consiste en endurecer la muestra de tejido en un medio, normalmente de parafina. Primero se deshidrata la muestra, reemplazando las moléculas de agua por etanol. La muestra se coloca en un molde lleno de parafina fundida, y transcurrido un tiempo, la muestra se solidifica, formando un bloque con mayor dureza y consistencia que el tejido original.
3. **Microtomía:** Una vez se tiene la muestra, se pasa a realizar un proceso de seccionamiento mediante microtomía. Las muestras se cortan en sección transversal, utilizando un micrómetro, en láminas de 5 micrómetros de grosor.
4. **Tinción:** El último paso, es la tinción de la muestra. Esto permite aumentar los contrastes para reconocer y diferenciar las distintas componentes biológicas del tejido. Para realizar la tinción, se eliminan los restos de parafina y posteriormente, se hidrata con pigmentos como hematoxilina y eosina (HE). [16]

El proceso anteriormente descrito se puede observar de manera gráfica en la Figura 3.2

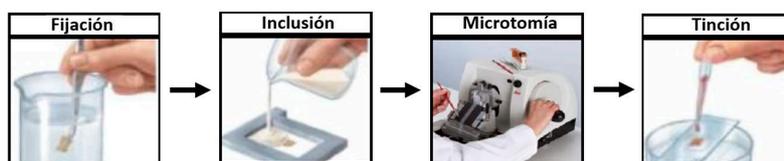


Figura 3.2: Proceso histológico. Figura extraída de [12].

3.2.2. Propiedades morfológicas de los tumores melanocíticos spitzoides

Una vez realizado el proceso histológico, el especialista analiza la composición y forma de la estructura biológica del tejido, donde destacan los núcleos, los citoplasmas, el lumen y los estromas (ver Figura 3.3).

A la hora de diagnosticar el tipo de neoplasia que presenta el paciente, el patólogo inspecciona las características morfológicas que presentan las muestras extraídas. A continuación, se detallan las diferencias entre estos tres tipos de neoplasias, pudiendo ver sus diferencias en la Figura 3.4.

Los tumores de Spitz típicos presentan un tamaño inferior a 10 mm y frecuentemente entre 5 mm y 6 mm. Se habla de tumores atípicos, cuando el tamaño es superior a los 10 mm y normalmente presenta una asimetría con bordes irregulares. Mientras que los nevus Spitz clásicos presentan un

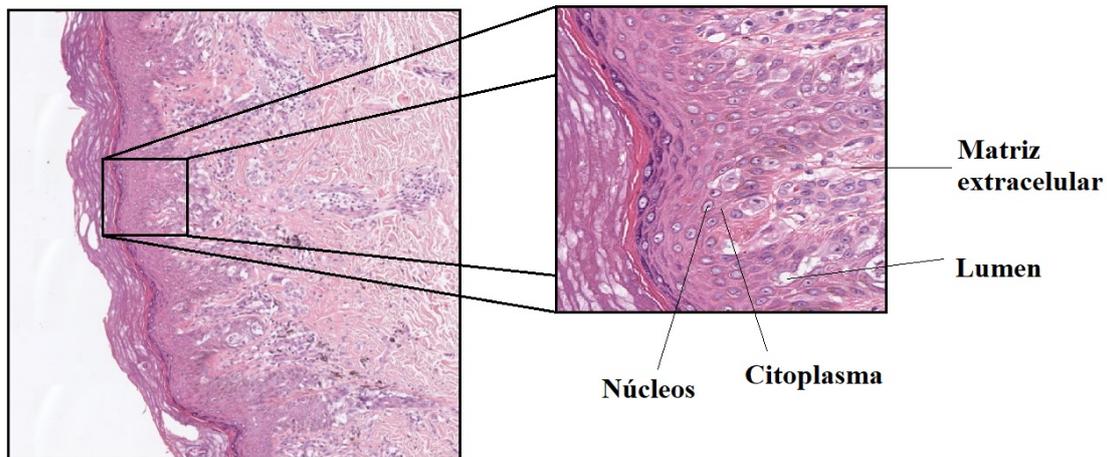


Figura 3.3: Muestra de tejido con melanoma. A la izquierda imagen a una resolución 5x, y a la derecha con una resolución 20x. Elaboración propia.

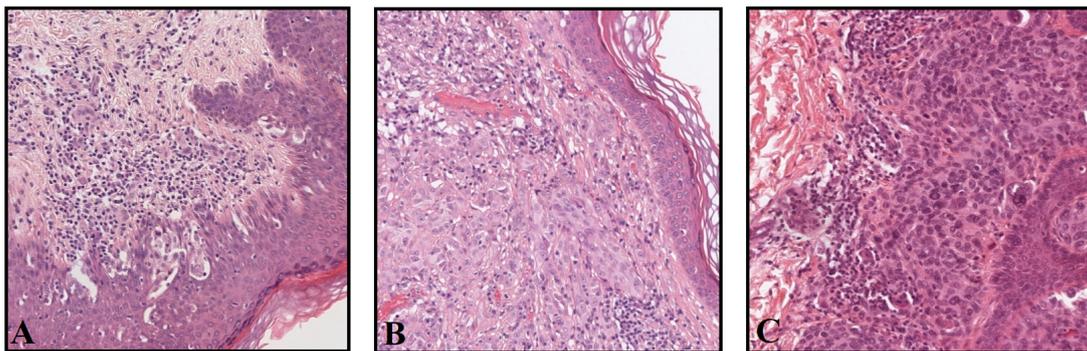


Figura 3.4: Imágenes WSI pertenecientes a la base de datos utilizada para el desarrollo del presente trabajo. Las muestras presentan una resolución 10x. Cada una de las imágenes está diagnosticada con un tipo distinto de lesión. A: Nevus Spitz (NS); B: Tumor de potencial maligno incierto (STUMP); C: Melanoma spitzoide (MS). Elaboración propia.

color rosa-rojizo, los tumores de Spitz atípicos presentan tonalidades mucho más irregulares. [17] [18].

Existen dos parámetros fuertemente correlacionados que se encuentran entre los criterios más importantes para evaluar las lesiones melanocíticas, estos son la confluencia prominente de melanocitos y la alta densidad celular. Otros atributos importantes son la alta homogeneidad y la maduración de los melanocitos. Un alto grado de homogeneidad está asociado a tumores de Spitz, mientras que un alto grado de maduración (disminución progresiva del tamaño de los melanocitos y una dispersión gradual por el tejido) y falta de homogeneidad, están asociadas a tumores mucho más agresivos. [18].

Uno de los atributos más importantes para la evaluación de las lesiones de spitzoides, son las mitosis por mm^2 . Un aumento de este ratio, está asociado a lesiones malignas. La mitosis se pue-

de observar en el fondo de las componentes dérmicas o subcutáneas. No existe ningún límite a partir del cual se considerará una lesión como maligna, pero algunos expertos como [19], lo sitúan en torno a 6 mm^2 . Otro parámetro, pero en este caso más subjetivo, es la evaluación de los atributos citológicos. Un patrón asimétrico o desordenado, una elevada relación núcleo-citoplasma, un citoplasma granular, una alta proporción de melanocitos con núcleos hiper cromáticos (es decir, células con un alto grado de cromatina) y presencia de grandes nucléolos eosinófilos, representa una lesión de carácter maligno [18].

En las Tablas 3.1 y 3.2 se puede observar un breve resumen de las diferencias biológicas entre los tumores de nevus Spitz y tumores de Spitz atípicos, y tumores de nevus Spitz y melanomas spitzoides, respectivamente.

Como se ha descrito en los párrafos anteriores, a la hora de evaluar una lesión, existe un alto grado de subjetividad, llevando en ocasiones a errores de diagnóstico. Es por ello, que los especialistas cuenten con imágenes con una alta resolución, lo que permite examinar la muestra con distintos aumentos (zoom), teniendo una mayor visión e información de la lesión. Los zoom más utilizados en este tipo de diagnóstico suelen ser: 5x, 10x y 20x, siendo estos los utilizados para desarrollar el clasificador automático de imágenes de este trabajo.

| Características clínicas | Spitz nevus clásico | Tumor de Spitz atípico |
|--------------------------|------------------------------|-----------------------------------|
| Edad | <10 años | >10 años |
| Ubicación | Cara, cuello y extremidades | Tronco |
| Tamaño | <10 mm (generalmente 5-6 mm) | >10 mm |
| Bordes | Bien definidos y regulares | Malamente definidos e irregulares |
| Superficie | Lisa | Irregular, ulcerada |
| Color | Rosa, rojizo | Irregular |

Tabla 3.1: Características clínicas de los tumores nevus Spitz y tumores de Spitz atípicos.

| Spitz nevus clásico | Melanoma spitzoide |
|---|---|
| Simétrico | Asimétrico |
| Bien definido | Malamente definido |
| Epidermis intacta | Epidermis puede ser ulcerada |
| Kamino <i>bodies</i> | Ausencia de Kamino <i>bodies</i> |
| Difusión de pagetoides limitada | Alta difusión de pagetoides |
| Menos de 2 mitosis/ mm^2 | Alta mitosis en la dermis |
| Bajo ratio núcleo/citoplasma | Alto ratio núcleo/citoplasma |
| Núcleos aumentan de tamaño, pero ausencia de pleomorfismo | Pleomorfismo celular, alto grado de atipia citológica |

Tabla 3.2: Características clínicas de los tumores nevus Spitz y melanomas spitzoides.

Con el fin de realizar modelos de clasificación automática que permitan distinguir los distintos

tipos de lesiones, es necesario una digitalización las muestras histopatológicas.

3.2.3. Digitalización de las muestras

Para la digitalización de las muestras se emplean equipos de alta prestaciones, cuyo precio oscila entre los 30.000 € - 250.000 € en función de sus características, como el tiempo de escaneado, resolución de la imagen, robustez del *hardware*, modos de adquisición, métodos/formatos/tamaño de compresión de los archivos o la capacidad del entorno aplicativo, entre otros [20].

Tras el proceso de digitalización de un tejido, se obtiene una WSI, la cual está formada por un conjunto de imágenes que presentan distintas resoluciones. La resolución hace referencia al número de píxeles por pulgada que tiene una imagen, cuantos más píxeles por pulgada existan, mayor información presentará la imagen. Esta combinación de resoluciones, permite obtener una imagen completa de la biopsia con un alto grado de detalle. Esto hace que en multitud de ocasiones, las WSI solo puedan ser visualizadas y tratadas desde un navegador web y almacenadas en servidores, debido al gran tamaño que ocupan (entre los 200 MB y los 5 GB [20]). Una WSI tiene una estructura piramidal, donde se van insertando imágenes con distintas resoluciones. La base de la estructura, contiene las imágenes con una mayor resolución, y según se va ascendiendo en la pirámide, la resolución va disminuyendo [21]. Las imágenes con una resolución superior, contienen una mayor información sobre la distribución de los núcleos celulares, citoplasma, lumen, etc., pudiéndose observar atributos como las mitosis por $/mm^2$ o el ratio núcleo/citoplasma. Por el contrario, cuanto menor es la resolución de la imagen, se puede observar el contexto de la lesión, y analizar si esta está o no extendida. En la Figura 3.5 se puede visualizar un ejemplo claro de una imagen WSI.

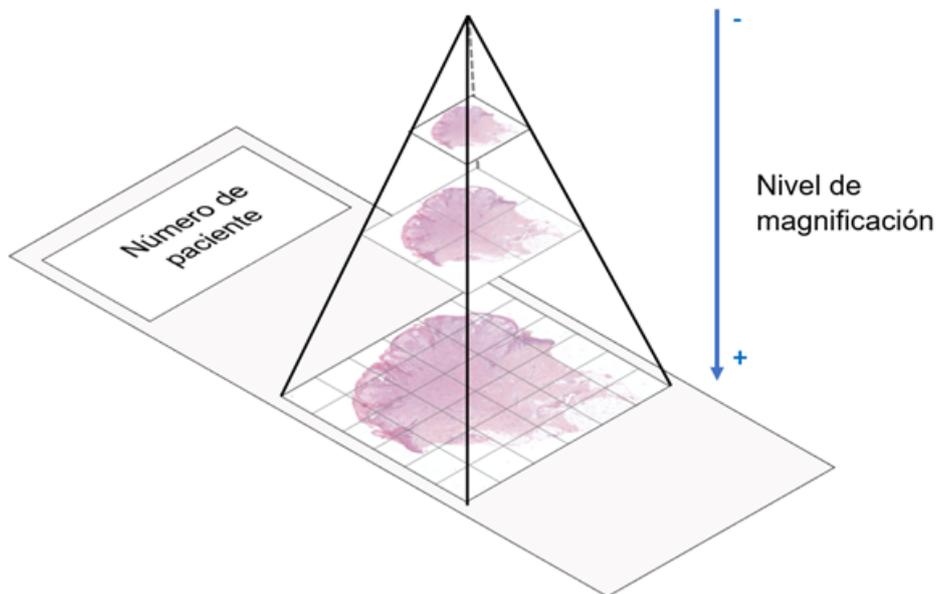


Figura 3.5: Composición de una imagen WSI histopatológica.

Capítulo 4

Metodología

4.1. Creación de la base de datos

Para poder trabajar con algoritmos de clasificación automáticos basados en técnicas de deep learning, es necesario crear una base de datos sólida con numerosas muestras. En este caso, las muestras serán imágenes hispatológicas obtenidas por el Servicio de Anatomía Patológica del Hospital Clínico Universitario de València. La base de datos está formada por un total de 49 muestras histopatológicas de diversos pacientes, diagnosticadas con una lesión melanocítica maligna (melanoma), una lesión melanocítica benigna (nevus) o una lesión de potencial maligno incierto (PMI). Para el desarrollo de este trabajo, solo se utilizan WSI pertenecientes a lesiones melanocíticas malignas y benignas. Para cumplir con la ley de protección de datos, esta base de datos está anonimizada, de forma que no se puede relacionar la muestra con ningún paciente.

Todas las muestras obtenidas son imágenes Whole Slide, también conocidas como "microscopía virtual", que están creadas a partir de una la digitalización, en alta resolución, del tejido [22]. Este tipo de imágenes permite obtener un archivo digital de una muestra visualizada a microscopio con alto detalle, facilitando el computo, visualización y almacenamiento de esta. Todas las muestras han de estar marcadas y etiquetadas por un especialista (patólogo), donde refleje las zonas de interés, qué anomalías han de ser analizadas y su respectivo diagnóstico.

4.1.1. Anotación y etiquetado de las regiones de interés

Para poder entrenar los modelos de clasificación automáticos, es necesario disponer de las regiones de interés de cada WSI. Para ello, se toma la muestra biopsiada digitalizada, normalmente en formato `.tif`, y se carga en un *software* específico como puede ser *MicroDraw*¹. Dentro de la aplicación, el patólogo se encarga de seleccionar y marcar las zonas de tejido que presentan anomalías. Una vez delimitadas las zonas, se procede con su etiquetado, donde se define el *ground truth*, es decir, la clase a la que pertenece esa muestra. En este caso, las muestras son clasificadas en benignas y malignas. Estos valores son anotados en un fichero `.xlsx` donde se relaciona el nombre de la imagen WSI con su etiqueta *ground truth* correspondiente.

Las regiones anotadas de cada WSI serán utilizadas durante los entrenamientos de cada modelo.

¹MicroDraw es una aplicación web para la visualización y anotación de datos histológicos de alta resolución.

Esto permite entrenar a la red neuronal con regiones que presentan neoplasias, evitando introducir ruido con otras regiones de la muestra donde no se presenta ningún tipo de lesión. El correcto proceso de etiquetado es fundamental en el proceso, ya que de ello depende que la red aprenda de forma precisa las distintas características que forman las neoplasias melanocíticas spitzoides benignas, y malignas.

4.1.2. Parcheo de las imágenes a distintas resoluciones

Debido al gran tamaño de las WSI, es necesario dividir las imágenes en imágenes de menor tamaño, las cuales puedan ser procesadas por la red neuronal. Por tanto, el primer paso para la creación de la base de datos es "parchear" todas las imágenes, es decir, obtener parches, llamados *patch*, de cada una de las imágenes. En este caso se han parcheado las imágenes con tres resoluciones diferentes: 5x, 10x y 20x, con un solapamiento del 50 % entre parches y un tamaño de imagen de 512x512 píxeles. El solapamiento de los parches, hace que aumente el número de total de imágenes disponibles para el entrenamiento, logrando una base de datos de mayor calidad. En la Figura 4.1, se puede observar en que consiste este proceso. Para realizar esta operación se ha utilizado un código escrito en Python, desarrollado por el grupo CVBLab. Cuanto mayor sea la resolución de la imagen, mayor será el número de parches finales obtenidos, y viceversa.

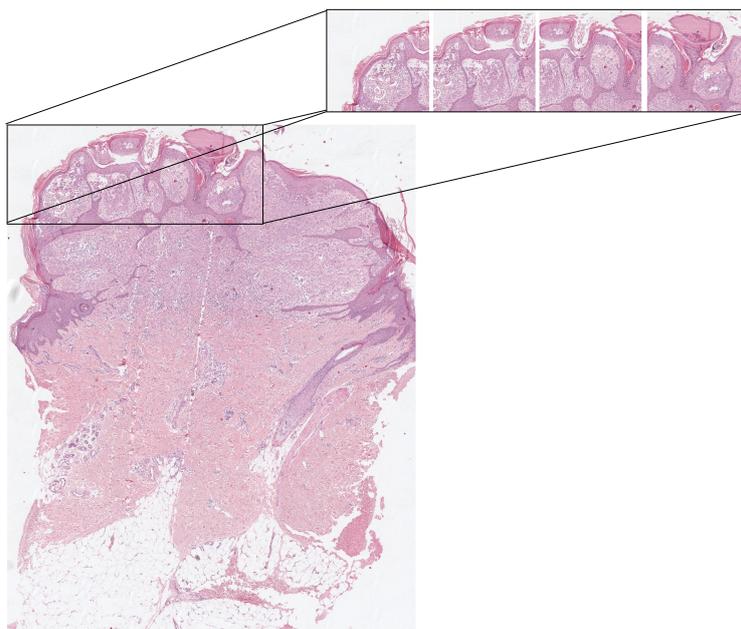


Figura 4.1: Parcheo de las imágenes (resolución 5x y 50 % de solapamiento). Elaboración propia.

4.1.3. Selección de las regiones de interés

Una vez parcheadas las imágenes es necesario separar los parches que contienen regiones anotadas por el patólogo. Para ello, se dividen los parches entre "parches con anotación" (*annotation*) y

“parches sin anotación” (*noannotation*). Para realizar estas acciones se ha utilizado un *script* en MATLAB, que recorre todas las imágenes guardando los parches que tienen regiones anotadas.

Con la finalidad de obtener los mejores resultados posibles a la hora de entrenar la red neural, es necesario seleccionar los parches que tienen un mínimo de información, es decir, aquellos que presentan un porcentaje mínimo de tejido y de región anotada. En este caso se han seleccionado los parches que contienen más de un 20% de tejido y al menos un 75% de región anotada. Para ello, se ha utilizado un *script* desarrollado en MATLAB, que recorre los parches de cada imagen, seleccionando aquellos que cumplen con estas condiciones. Para conocer el porcentaje de tejido en cada parche, se ha analizado el canal de color magenta, ya que es el color predominante en cada imagen. Este proceso se ha llevado a cabo para las bases de datos a 5x, 10x y 20x.

4.1.4. Relación de parches a distinta resolución

Para entrenar la red neural con datos a distintas resoluciones, es necesario establecer una relación entre los distintos parches a resolución 5x, 10x y 20x. Al existir un solapamiento del 50% entre parches, dentro de cada parche a resolución 5x se tienen un total de 9 parches a resolución 10x que están completamente contenidos en el parche a 5x. De manera análoga, por cada parche a resolución 10x, existen 9 parches a resolución 20x contenidos en el parche a 10x. Cada parche se encuentra identificado por la región donde se ubica dentro de la imagen WSI, y la columna y la fila que ocupa en dicha región.

La relación entre los parches a distintas resoluciones se ha establecido a través de la posición de las filas y columnas que ocupa cada uno. Dentro de un parche a 5x estarán los parches a 10x que cumplen una de las siguientes condiciones:

- La fila del parche a 10x es el doble de la fila a 5x.
- La fila del parche a 10x es el doble más uno de la fila a 5x.
- La fila del parche a 10x es el doble más dos de la fila a 5x.

Y además que cumplan una de las siguientes condiciones:

- La columna del parche a 10x es el doble de la columna a 5x.
- La columna del parche a 10x es el doble más uno de la columna a 5x.
- La columna del parche a 10x es el doble más dos de la columna a 5x.

Por tanto, para determinar que un parche está contenido en otro se tienen que dar dos condiciones: que la fila del parche cumpla con una de las condiciones anteriormente comentadas y que la columna del parche cumpla con una de las condiciones anteriormente comentadas. La relación que existe entre los parches a 10x y 20x es la misma que la comentada anteriormente.

Para crear estas relaciones, se ha elaborado un *script* de MATLAB que implementa las condiciones comentadas anteriormente y crea un archivo `.xlsx` con estos datos. Para este desarrollo, se ha empezado recorriendo los parches con una resolución mayor, es decir 20x, y analizando si se encuentran contenidos dentro de un parche a resolución menor. A continuación se detalla el código en MATLAB desarrollado, para considerar que un parche está contenido en otro de menor resolución y escribirlo en un archivo `.xlsx`.

Código 4.1: Código elaborado en MATLAB para relacionar los parches a distintas resoluciones.

```

% Cargamos los parches de cada imagen
patch_5x = dir([routeimages5x '\ ' dirimages5x(image).name '\annotation
\*.jpg']);
patch_10x = dir([routeimages10x '\ ' dirimages20x(image).name '\
annotation_selected_lesion\*.jpg']);
patch_20x = dir([routeimages20x '\ ' dirimages20x(image).name '\
annotation_selected_lesion\*.jpg']);

% Calculamos el número máximo de columnas de cada imagen
img_dir = dir([routeimages5x '\ ' dirimages5x(image).name]);

% Transformamos el array a un tipo cell para calcular las columnas y
las filas de interés
C_5x = cellfun(@(x) split(x, '_'), string({patch_5x.name}), '
UniformOutput', false);
C_10x = cellfun(@(x) split(x, '_'), string({patch_10x.name}), '
UniformOutput', false);
C_20x = cellfun(@(x) split(x, '_'), string({patch_20x.name}), '
UniformOutput', false);

% Buscamos la posición que indica el número de fila y columna (
posiciones 7 y 9 del array)
for 20x=1:length(C_20x)

    coincidencia = 0; % Ponemos a 0 el contador de coincidencia

    row_20x = str2num(cell2mat(C_20x{20x}(7)));
    col_20x = str2num(cell2mat(C_20x{20x}(8)));

    % Recorremos los patch 10x
    for 10x=1:length(C_10x)

        % Comprobamos si hemos encontrado una coincidencia
        if coincidencia == 1
            break;
        else
            % La relación que existe es que por cada rol/col de 10x
            existen 2*row, 2*row+1 y 2*row + 2 de 20x
            % Leemos las filas y columnas de cada parche
            row_10x = str2num(cell2mat(C_10x{10x}(7)));
            col_10x = str2num(cell2mat(C_10x{10x}(8)));

            % Imágenes con nombres distintos
            C_col10x = cellfun(@(x) split(x, '.'), string({C_10x{10x}
            ,1}(8)}), 'UniformOutput', false);
            col_10x = str2num(cell2mat(C_col10x{1,1}(1)));

            if (row_20x == 2*row_10x || row_20x == 2*row_10x + 1 ||
            row_20x == 2*row_10x + 2) ...
                && (col_20x == 2*col_10x || col_20x == 2*col_10x +
            1 || col_20x == 2*col_10x + 2)

```

```

%Buscamos la coincidencia de los parches 20x
for 5x = 1:length(C_5x)
    %La relación que existe es que por cada rol/col de
    % 10x existen 2*row, 2*row+1 y 2*row + 2
    %Leemos las filas y columnas de cada parche
    row_5x = str2num(cell2mat(C_5x{5x}(7)));
    %
    %                               col_5x = str2num(
    %                               cell2mat(C_5x{5x}(10)));

    %Imágenes con nombres distintos
    C_col5x = cellfun(@(x) split(x, '.'), string({C_5x{5
        x,1}(8)}), 'UniformOutput', false)';
    col_5x = str2num(cell2mat(C_col5x{1,1}(1)));

    if ((row_10x == 2*row_5x || row_10x == 2*row_5x + 1
        || row_10x == 2*row_5x + 2) ...
        && (col_10x == 2*col_5x || col_10x == 2*
            col_5x + 1 || col_10x == 2*col_5x + 2))
        %Guardamos los parches que coinciden para
        % escribirlo en el excel
        all_patch = {patch_5x(5x).name, patch_10x(10x).
            name, patch_20x(20x).name};

        %Escribimos en el excel la relación de parches
        writecell(all_patch, namexlsx, 'WriteMode', '
            append');

        %Una vez encontrado una, continuamos buscando
        % más coincidencias pero con otros parches a
        % 20x
        coincidencia = 1;
        break;
    end
end
end
end
end
end
end
end
end

```

En la Figura 4.3 se puede visualizar las relaciones entre el parche a una resolución 5x y los parches a resolución 10x y 20x.

En el archivo `.xlsx` final generado, que se utilizará en los entrenamientos de los modelos, se han relacionado las siguientes carpetas de parches: para la resolución 5x se han utilizado los parches de las regiones anotadas; para las resoluciones 10x y 20x se han utilizado los parches de las regiones con lesiones anotadas y que presentan al menos más de un 20% de tejido. Esto hace que los datos de entrada a la red pertenezcan siempre a regiones que han sido anotadas por el especialista como zona de lesión.

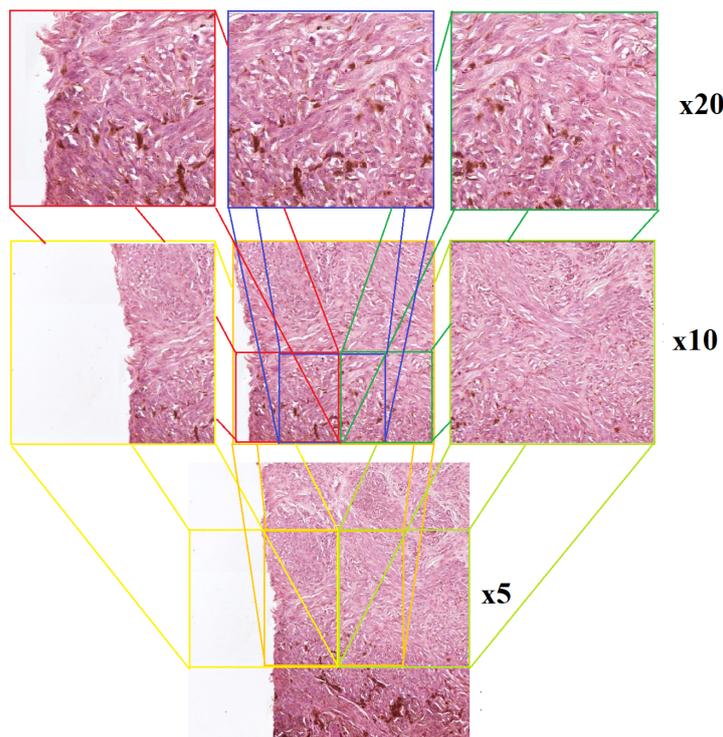


Figura 4.2: Relación entre un parche a resolución 5x, 10x y 20x. Elaboración propia.

4.1.5. Partición de los datos

La partición de los datos se realiza con el objetivo de realizar un mejor entrenamiento de la red y evitar *overfitting* o *underfitting*. En este proceso, se han de establecer que datos se van a utilizar para entrenar y cuales para validar, siendo estos totalmente diferentes. Si se utilizan los mismos datos para entrenar y validar, el modelo no tendrá validez alguna, ya que habrá aprendido una serie de datos y cuando se valide, esa información ya estará aprendida. También hay que tener en cuenta la variabilidad de las muestras, es decir, es necesario que las clases a clasificar estén balanceadas, teniendo aproximadamente el mismo número de muestras de cada clase.

En el presente trabajo, únicamente se consideran dos clases, cuando la muestra ha sido diagnosticada con nevus Spitz (neoplasia benigna) y con melanoma (maligno). Para realizar los entrenamientos de los distintos modelos, se cuenta con una base de datos con 46 muestras de pacientes diferentes. Se han utilizado 35 muestras (cada una representa un paciente), para los datos de entrenamiento y validación. Para realizar esta partición se ha llevado a cabo la técnica *k-fold cross validation*. En el testeo de del modelo final se han empleado las 11 imágenes restantes, las cuales la red no había visto antes.

La técnica *k-fold cross validation* consiste en dividir los datos en varios subconjuntos, y realizar entrenamientos y validación de forma iterativa con cada uno de estos. Se ha implementado la metodología *k-fold* con $k = 4$. Se han realizado 4 iteraciones de entrenamiento con datos distintos, como se puede apreciar en la Figura 4.3. Estos datos se han distribuido de forma balanceada para intentar que existan el mismo numero de parches pertenecientes a la clase 0 (benigno) como a la

clase 1 (maligno). En las Tablas 4.1 y 4.2 se aprecia la relación de datos para los entrenamientos y validación de las distintas redes neuronales a una única resolución.

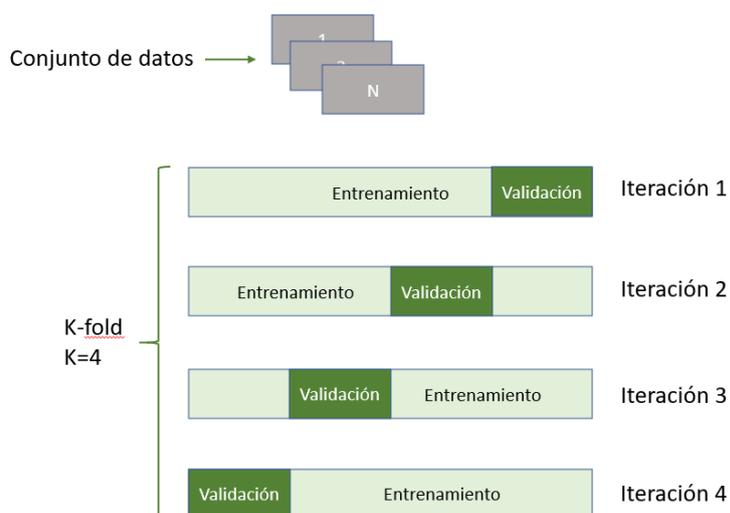


Figura 4.3: Estructura de la partición de los datos siguiendo la técnica k -fold con $k = 4$. Elaboración propia.

| k -fold | Imágenes totales | Clase 0 | Clase 1 |
|----------------|------------------|---------|---------|
| <i>Fold 1</i> | 1880 | 1037 | 843 |
| <i>Fold 2</i> | 1949 | 962 | 987 |
| <i>Fold 3</i> | 1704 | 916 | 788 |
| <i>Fold 4</i> | 1501 | 666 | 835 |
| Total imágenes | 7034 | 3581 | 3453 |

Tabla 4.1: Partición de los datos para los entrenamientos a resoluciones 5x, 10x y 20x.

| k -fold | Imágenes totales | Clase 0 | Clase 1 |
|----------------|------------------|---------|---------|
| <i>Fold 1</i> | 435 | 127 | 308 |
| <i>Fold 2</i> | 403 | 239 | 164 |
| <i>Fold 3</i> | 678 | 315 | 316 |
| <i>Fold 4</i> | 833 | 517 | 835 |
| Total imágenes | 2349 | 1168 | 1623 |

Tabla 4.2: Partición de los datos para la validación a resoluciones 5x, 10x y 20x.

Para la partición de los datos a multiresolución se ha utilizado la misma estructura que la empleada en modelos a una única resolución, con la diferencia que ahora los datos de entrada son parches a resoluciones 5x, 10x y 20x. Como se comentaba en el apartado 4.1.4, se ha establecido una relación entre los parches a distinta resolución, en esa relación se reflejan todos los parches a 20x contenidos en 10x y a su vez en 5x. Por tanto, el número total de parches introducidos de forma

simultánea a la red por paciente, será el número total de parches a resolución 20x. En las Tablas 4.3 y 4.4 se refleja la partición de datos para el entrenamiento y la validación respectivamente. Como se puede apreciar, el número de imágenes aumenta considerablemente, y esto se debe a que, como se comentó en el apartado 3.2.3, las imágenes con mayor resolución, están formadas a su vez por un mayor número de imágenes.

| <i>k-fold</i> | Imágenes totales | Clase 0 | Clase 1 |
|----------------|------------------|---------|---------|
| <i>Fold 1</i> | 14681 | 9160 | 5521 |
| <i>Fold 2</i> | 17257 | 8913 | 8344 |
| <i>Fold 3</i> | 13446 | 8558 | 4888 |
| <i>Fold 4</i> | 11085 | 4313 | 6772 |
| Total imágenes | 56469 | 30944 | 25525 |

Tabla 4.3: Partición de los datos para el entrenamiento a multiresolución.

| <i>k-fold</i> | Imágenes totales | Clase 0 | Clase 1 |
|----------------|------------------|---------|---------|
| <i>Fold 1</i> | 4142 | 1154 | 2988 |
| <i>Fold 2</i> | 1566 | 1401 | 165 |
| <i>Fold 3</i> | 5377 | 5377 | 3621 |
| <i>Fold 4</i> | 7738 | 6001 | 1737 |
| Total imágenes | 18823 | 13933 | 8511 |

Tabla 4.4: Partición de los datos para la validación a multiresolución.

El número total de imágenes en el entrenamiento y validación de esta red se muestra en la Tabla 4.5.

| <i>k-fold</i> | Imágenes totales | Clase 0 | Clase 1 |
|---------------|------------------|---------|---------|
| Entrenamiento | 18823 | 10314 | 8509 |
| Test | 7747 | 2446 | 5301 |

Tabla 4.5: Número total de imágenes utilizadas en el entrenamiento y validación de la red neuronal multiresolución.

4.2. Algoritmos de clasificación basados en *deep learning*

Para el desarrollo de un clasificador automático para la detección de tumores melanocíticos spitzoides, se hará uso de técnicas de inteligencia artificial basadas en *deep learning*, y en específico en la utilización de redes neuronales convolucionales (CNN). Las redes neuronales convolucionales permiten extraer de forma automática, las características más relevantes de los datos, en este caso imágenes, que son introducidos como *input* a la red.

4.2.1. Redes neuronales convolucionales (CNN)

Una red neuronal convolucional o CCN por sus siglas en inglés (*Convolutional Neural Network*), es un algoritmo de *deep learning* diseñado para extraer características automáticamente de una serie de datos. Las CNN están diseñadas para el reconocimiento y clasificación de imágenes. Entre sus principales características destaca que son capaces de: aprender características directamente sin intervención humana, generar resultados de reconocimiento con un alto nivel de precisión y volver a entrenarse con distintos datos para nuevas tareas de reconocimiento [23].

Las redes neuronales convolucionales están formadas por una serie de capas (*layers*) encargadas de realizar operaciones matemáticas a los datos de entrada, con la finalidad de extraer ciertas características que identifiquen los datos suministrados a la red. Estas redes se dividen en dos partes bien diferenciadas: *base model* y *top model*.

4.2.1.1. Base model de una red CNN

El *base model* es el encargado de realizar el grueso de las operaciones matemáticas y extraer las características de los datos de entrada, donde las principales capas que existen son: capas convolucionales, funciones de activación, capas de *dropout* y capas de *pooling* o sub-muestreo.

Las capas convolucionales, aplican un filtro (kernel) al mapa de características de entrada (conocido en inglés como *feature maps*), dando como resultado un mapa de características de salida. Este kernel va realizando convoluciones de forma secuencial por la matriz de entrada. Esta operación está definida por dos parámetros: el tamaño de la matriz del filtro y la profundidad del mapa de atributos de salida, que corresponde a la cantidad de filtros que se aplican. En la Figura 4.4, se puede observar el funcionamiento de las capas convolucionales con un filtro de 3x3 [24].

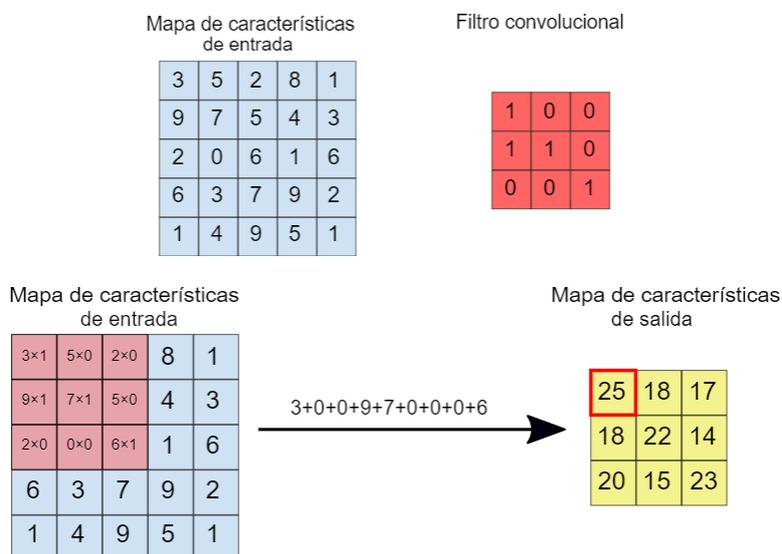


Figura 4.4: Operación de convolución dentro de una capa convolucional. Figura extraída de [24].

Las capas de activación se aplican después de cada operación de convolución, y en el caso de las redes CCN, la más utilizada es la función de activación ReLu (*Rectified Linear Unit*). La capa

ReLU es una función lineal que devuelve 0 si la entrada es negativa, y el mismo valor de entrada para cualquier valor positivo, ver Figura 4.5. La función ReLU ayuda al modelo a tener en cuenta los efectos de interacción (cuanto afecta una variable de los datos de entrada a otra) y a tener en cuenta los efectos no lineales. La expresión matemática viene definida por la ecuación 4.1 [25].

$$f(x) = \max(0, x) \quad (4.1)$$

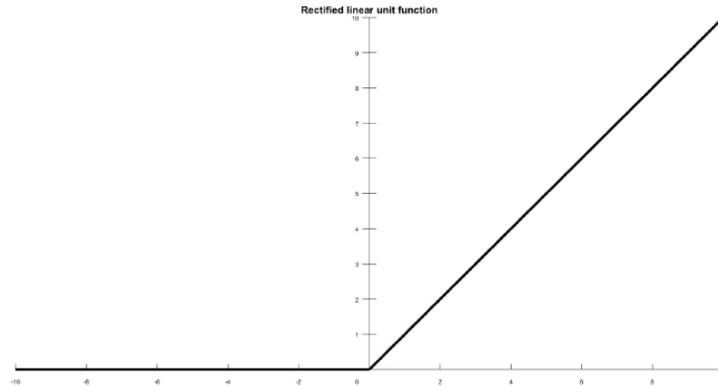


Figura 4.5: Gráfica de la función de activación ReLU. Figura extraída de [25].

Las capas de *dropout* establecen de manera aleatoria los datos de entrada a 0 con una probabilidad dada, reduciendo así el número de conexiones entre neuronas. Por ejemplo, si se utiliza una capa de *dropout* de 0.2, la red neural en cada iteración de entrenamiento, establecerá a 0 un 20 % de los datos de entrada. Esta capa evita que se produzca *overfitting*, es decir, que la red se sobreentrene y solo aprenda los datos de entrenamiento pero no aprenda a clasificar datos distintos.

Finalmente, las capas de *pooling* o de sub-muestreo se encargan de reducir las dimensiones del mapa de características, disminuyendo así el número de parámetros y el coste computacional de la red. Existen tres tipos de *pooling*: *max pooling*, que selecciona el máximo de los elementos de la región del mapa de características cubierta por el filtro aplicado; *average pooling*, que realiza la media de los elementos seleccionados por el filtro; y *global pooling*, que reduce cada canal del mapa de características en un único valor, ya sea aplicando la técnica del máximo valor (*max pooling*) o la media de los valores (*average pooling*). En la Figura 4.6 se puede observar un ejemplo del proceso de sub-muestreo de una matriz de características 224x224x64 a una matriz de 112x112x64.

4.2.1.2. Top model de una red CNN

El *top model* de una red neural se refiere a las últimas capas de esta, y son las encargadas de clasificar los datos de entrada en función de las características extraídas por el *base model*. Las principales capas que forman un *top model* son: capas de *global pooling* (comentadas anteriormente), capas *fully connected* (FC) - o completamente conectadas -, capa *softmax* y capa de clasificación. También es común, añadir capas de *dropout* o *flatten*, la cual convierte los datos de entrada (normalmente un volumen de datos) en una matriz unidimensional. Mientras que la capa *global pooling*, aplica operaciones matemáticas de máximos, o media aritmética a los valores numéricos de los volúmenes de entrada, la capa *flatten*, no modifica el valor de estos datos. Con las capas de *global*

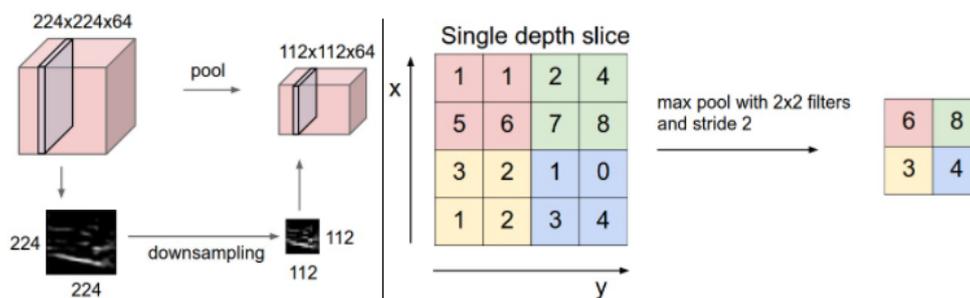


Figura 4.6: Proceso de *pooling*. A la izquierda, un mapa de características de tamaño $224 \times 224 \times 64$ se sub-muestra con un filtro de tamaño 2, pasando a un volumen de $112 \times 112 \times 64$. A la derecha, el proceso de *pooling* sobre un *slice* de $4 \times 4 \times 1$, aplicando *max pooling* con un filtro de 2×2 . Figura extraída de [26].

pooling se consigue reducir el número de parámetros de la matriz de características, mientras que con la capa *flatten*, se obtiene un mayor número de parámetros. En la Figura 4.7 se puede ver un ejemplo de la diferencia entre una capa *global pooling* y una capa *flatten*.

La capa *fully connected*, se encarga de conectar las neuronas de la capa anterior con la siguiente. Estas neuronas que forman la capa *fully connected*, también se conocen como neuronas ocultas, y su función dentro del *top model* es pasar de un número determinado de neuronas menor. En la Figura 4.8 se visualiza un ejemplo donde se pasa de m neuronas de entrada a 2 neuronas de clasificación (x e y).

La capa *softmax* está ubicada justo en la capa de clasificación de una red neural CNN, y se encarga de asignar probabilidades a cada clase. Si se toma como referencia la Figura 4.8, donde existen 2 clases, la capa *softmax* asignará a cada una de esas clases una probabilidad, entre 0 y 1, en función de los valores de cada neurona. Estas probabilidades deben de sumar 1. La función *softmax* se encuentra definida en la ecuación 4.2.

$$softmax(x)_i = \frac{exp(x_i)}{\sum_j exp(x_j)} \quad (4.2)$$

Finalmente, en la capa de clasificación se decide cual es la clase predecida en función de la probabilidad mayor asignada a las clases por la capa *softmax*.

En la Figura 4.9 se muestra la arquitectura de una red neuronal convolucional genérica al completo.

4.2.2. Técnica de *transfer learning*

La técnica de *transfer learning* es una de las más usadas a la hora de crear clasificadores automáticos de imágenes. Como su propio nombre indica, *transfer learning* no es más que la transferencia de conocimiento de una red a otra. Para ello se toman redes ya entrenadas, en algunos casos con más de 1.000.000 de imágenes de 1.000 categorías diferentes. Para realizar esta transferencia de conocimiento existen varias formas [28]:

- Utilizar una CNN como un extractor fijo de características, donde se congelen todas las capas

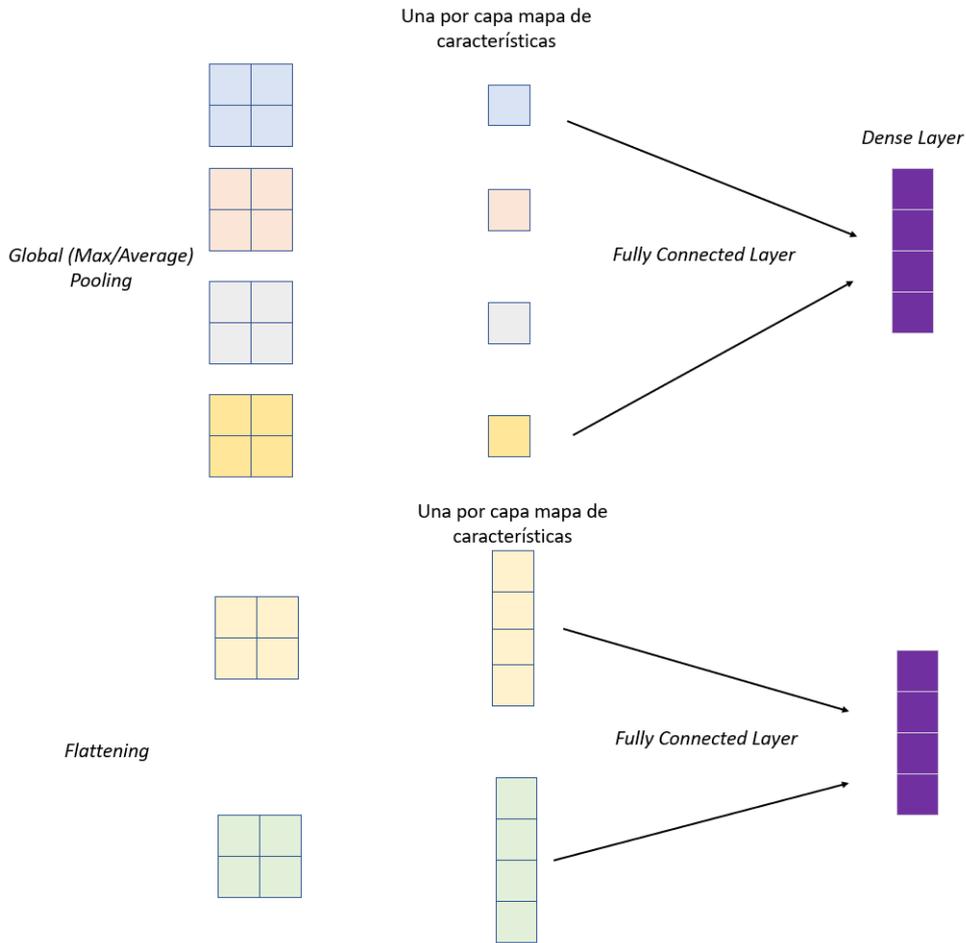


Figura 4.7: Diferencias entre la capa *global pooling* y la capa *flatten*.

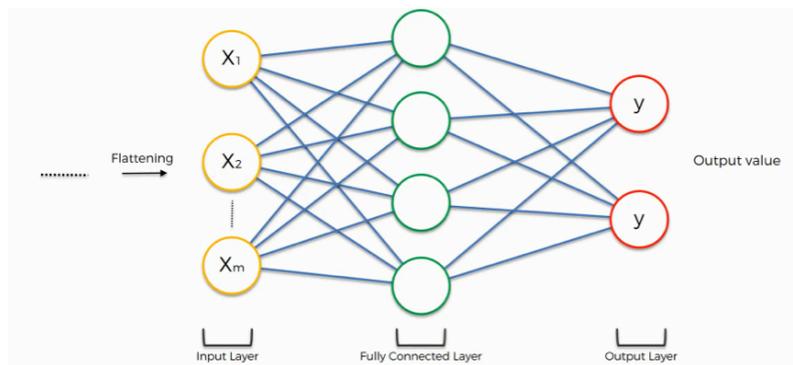


Figura 4.8: Representación gráfica de una capa *fully connected* en un *top model*. Figura extraída de [27].

del *base model* y únicamente se entrena un nuevo *top model* con un clasificador basado en *softmax* o en un *Support Vector Machine (SVM)*.

- Por el contrario, es posible entrenar las capas del *base model*, pudiendo congelar algunas

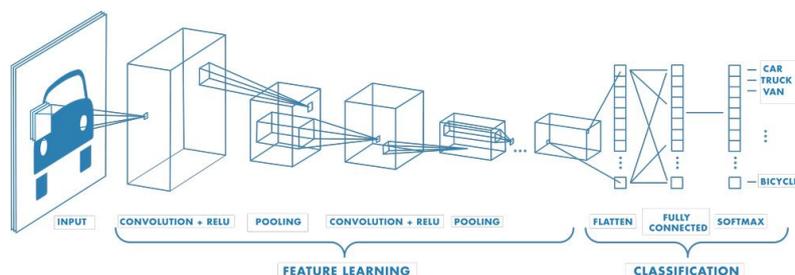


Figura 4.9: Red neuronal convolucional (CNN) genérica completa. Figura extraída de [23].

de ellas, optimizando así sus pesos. Esta técnica se le conoce con el nombre de *fine-tuning*. Como se verá en el siguiente apartado, esta ha sido la técnica elegida para crear los modelos de clasificación de este trabajo.

Entre los *dataset* más famosos a la hora de entrenar redes neuronales, se encuentra *ImageNet*, el cual contiene 1.000 clases diferentes y más de 14 millones de imágenes a fecha de la realización de este trabajo. Entre las redes más populares entrenadas con este *dataset* destacan: VGG16 [29], Inceptionv3 [30] y RESNET [31].

4.3. Redes neuronales híbridas

En una red neuronal, es posible utilizar diferentes datos de entrada, a estos modelos se les conoce como redes neuronales híbridas (Hybrid Neural Networks, HNN). Estas redes reciben una mayor información acerca de los datos de entrada, pudiendo obtener mejores resultados que si se entrenan las redes con los datos de manera independiente. Este tipo de redes neuronales dan la posibilidad de fusionar los datos en una etapa temprana (*early fusion*) o tardía (*late fusion*). En la técnica *early fusion*, los modelos de clasificación se desarrollan con los datos fusionados en las primeras capas de la red. Para ello, se elaboran modelos independientes de un número de capas reducido, para cada entrada de datos. Una vez se obtienen los vectores de características de estas capas, estos se concatenan formando un único volumen de datos, el cual se introduce como entrada en la red clasificadora. Por el contrario, en la técnica *late fusion*, se desarrollan distintos modelos de clasificación para cada entrada de datos. Una vez los datos han sido procesados por la red neuronal, y se han obtenidos los distintos vectores de características, estos se concatenan y se pasa a la etapa de clasificación de la red.

En el presente trabajo fin de máster, se han empleado redes híbridas para clasificar imágenes histopatológicas a distinta resolución. Este tipo de redes neuronales multiresolución, se explican en más detalle a continuación.

4.3.1. Redes neuronales multiresolución

Las redes neuronales convolucionales multiresolución, son una modificación de las redes neuronales convolucionales tradicionales. Consisten en fijar como entrada a la red, el mismo fragmento de imagen, pero a distintas resoluciones, de manera simultánea. Es decir, si se parte de la resolución mayor, esa imagen ha de estar contenida dentro de las imágenes de menor resolución. La principal

razón para utilizar multiresolución en el análisis de imágenes médicas, es aprender la estructura general y sus características a distintos niveles de detalle, como por ejemplo, la organización de las células. Esto permite eliminar la pérdida de resolución que sufre la imagen de entrada con el paso a través de las capas de las CNN, ya que debido a las múltiples operaciones convolucionales y capas de *pooling*, mucha información visual se pierde [32]. En la Figura 4.11 se muestra un ejemplo genérico de una red CNN basada en multiresolución con resoluciones a 5x, 10x y 20x y con dos clases de clasificación.

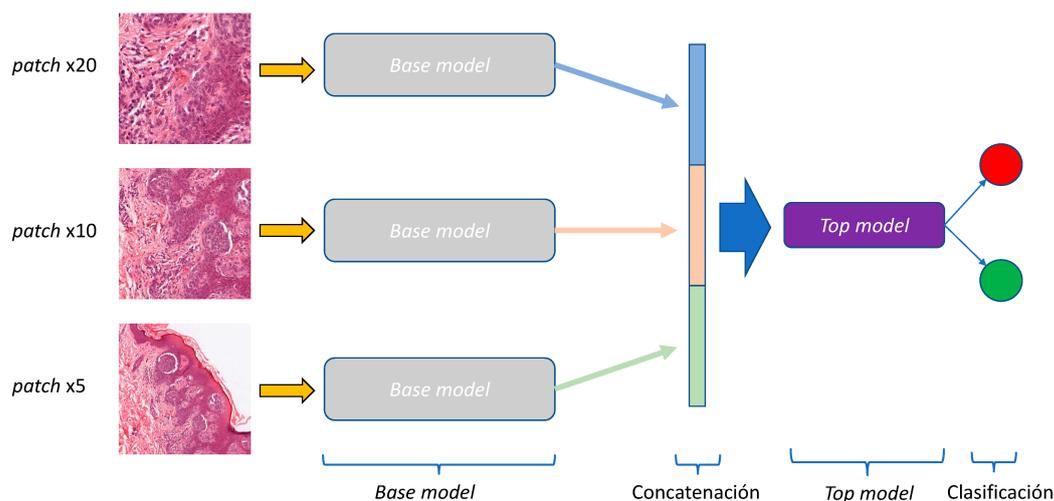


Figura 4.10: Red neuronal convolucional multiresolución, basada en 3 entradas de datos a resoluciones 5x, 10x y 20x. Elaboración propia.

El problema fundamental de las redes multiresolución es que requieren de una mayor potencia de computación, ya que se han de procesar un mayor número de operaciones, siendo estas proporcionales al número de resoluciones utilizadas. Esto también hace que el número de parámetros en la *top model* sea mayor, teniendo que aplicar capas de *pooling* para su disminución.

Existen diversas formas y métodos de crear modelos de clasificación automática con imágenes a distintas resoluciones. Por ejemplo, en [5], utilizan dos etapas de procesamiento consecutivas. Primero se extraen las características de la red a una resolución x5, una vez se han mapeado las zonas de interés en las imágenes, se realiza una extracción de características y clasificación de los parches seleccionados a una resolución de 10x. Otro ejemplo es [32], donde realizan una etapa de pre-procesado para obtener parches de las imágenes a distintas resoluciones, concatenan las imágenes con la misma resolución y las dividen en los datos de entrenamiento y validación. Por último, elaboran un modelo cuya entrada son las imágenes con distinta resolución, la información de cada imagen se concatena al final del *base model*, técnica que se conoce como *late fusion*. En la Figura 4.11 se puede apreciar el proceso seguido. En este caso, esta ha sido la técnica elegida para el desarrollo del presente trabajo.

Estos modelos de redes neuronales que combinan imágenes a distintas resoluciones, se elaboran con el fin de simular el proceso que realizan los patólogos al aumentar y disminuir el zoom sobre la biopsia, mientras examinan el tejido en imágenes WSI [8].

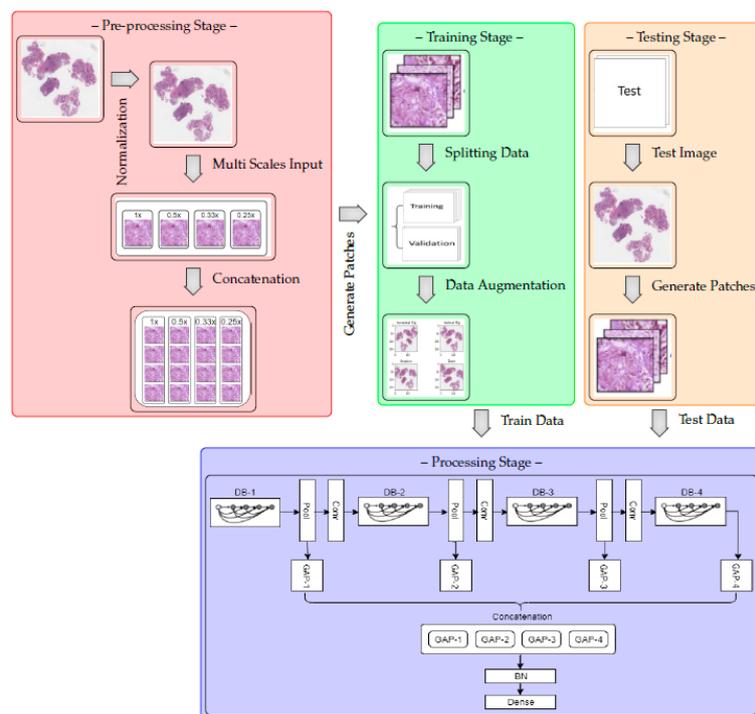


Figura 4.11: Red neuronal convolucional multiresolución, siguiendo el proceso de *late fusion*.
 Figura extraída de [32].

Capítulo 5

Implementación y resultados

5.1. Implementación de modelos de *deep learning*

En el desarrollo de los modelos de clasificación automática de las imágenes WSI para la detección de tumores melanocíticos spitzoides, se ha utilizado el software de desarrollo MATLAB. MATLAB es una plataforma de programación y cálculo numérico, utilizada por ingenieros y científicos para analizar datos, desarrollar algoritmos y crear modelos [33].

El desarrollo del código necesario para la elaboración del presente trabajo se ha llevado a cabo en un equipo portátil con Windows 10, procesador Intel(R) Core(TM) i7-10510U CPU @ 1.80GHz - 2.30 GHz y 8 GB de memoria RAM. En este equipo se ha contado con la versión de MATLAB R2021a. Como la computación de los algoritmos de *deep learning* es altamente costosa, se ha hecho uso de un servidor del grupo de investigación CVBLab. Este servidor, cuenta con un sistema operativo Linux, procesador Intel(R) Core(TM) i7-7700K CPU @ 4.20 GHz, 32 GB DIMM DDR4 de memoria RAM y una tarjeta gráfica NVIDIA GP102 TITAN Xp con 12GB de memoria RAM. En este servidor, se encuentra instalada la versión de MATLAB R2020b.

El proceso de implementación desarrollado en este trabajo fin de máster, se ha dividido en dos partes: desarrollo de los algoritmos para el entrenamiento de modelos a una única resolución y elaboración de los algoritmos para el entrenamiento de modelos a multiresolución (5x, 10x y 20x). A continuación se detallan ambas fases del proceso.

5.1.1. Implementación de modelos a una única resolución

Para desarrollar los modelos de clasificación, se ha empleado la técnica de *fine-tuning*, comentada en el apartado 4.2.2.

En primer lugar, se han establecido los distintos sets de datos, los cuales serán el *input* de la red neuronal. Estos sets de datos se dividen en: entrenamiento, validación y testeo de la red. Para ello, se han generado *datastore* independientes, donde se almacenan las imágenes con su etiqueta. Estos objetos son fácilmente manipulables, permitiendo aplicar escalados o modificaciones a todas las imágenes de manera simultánea.

Una vez se tienen todos los sets de datos (en este caso cuatro, ya que se está utilizando *k-fold* con $k = 4$), se pasa a cargar la red neuronal a utilizar. Antes de realizar un re-entrenamiento de la

red, la cual es pre-cargada con los pesos originales, se ha de reemplazar las capas del *top model* por otras donde sus pesos no posean ningún valor previo, con el fin de ajustarlos durante las distintas épocas del entrenamiento. Posteriormente, se ha de decidir cuantas capas de la red neural se van a dejar con los mismos pesos que la original, es decir, que capas no se van a ajustar durante el entrenamiento. Para ello, se congela un número de capas de la red determinado.

Cuando se dispone de la red neuronal lista para entrenar, se ha de valorar la opción de realizar *data augmentation* a las imágenes. El proceso de *data augmentation* consiste en realizar modificaciones de traslación, rotación, escalado y/o reflexión a las imágenes. Esto permite obtener un mayor número de imágenes variadas para el entrenamiento, evitando el *overfitting* de la red. Durante este paso, todas las imágenes han de ser escalas al tamaño de las capas de entrada de la red neuronal.

Por último, se han de fijar las opciones de entrenamiento, estos valores se conocen como hiperparámetros. A continuación, se detallan los distintos parámetros que se han ajustado para el entrenamiento de los modelos de *deep learning* [34]:

- ***solverName***: optimizador que se utilizará durante el entrenamiento de la red. Permite elegir entre: ‘*sgdm*’ (descenso estocástico del gradiente con ‘*momentum*’), ‘*rmsprop*’ (RMSProp) y ‘*adam*’ (Adam).
- ‘***MaxEpochs***’: número máximo de épocas que se entrenará el modelo.
- ‘***MiniBatchSize***’: tamaño del *mini-bath* en cada iteración de entrenamiento. Un *mini-batch* es un conjunto de datos de entrenamiento que se utilizan para evaluar la función de pérdidas y actualizar los pesos.
- ‘***Shuffle***’: desordena de forma aleatoria los datos de los *data sets* antes de entrenar o antes de cada época.
- ‘***ValidationData***’: datos que se van a utilizar para la validación del modelo.
- ‘***ValidationFrecuency***’: número de iteraciones tras los que se realiza la validación del modelo.
- ‘***ValidationPatience***’: número de veces que las pérdidas de validación son superiores o iguales que una pérdida menor antes de que se tenga el entrenamiento. Es decir, permite establecer el umbral donde el desarrollador cree que la red ha convergido, y la red finaliza red el entrenamiento.
- ‘***InitialearnRate***’: tasa de aprendizaje inicial, también conocida como *learning rate*. Un *learning rate* bajo, llevará un entrenamiento más largo, pero por el contrario, con un *learning rate* alto, el entrenamiento puede alcanzar resultados sub-óptimos o diverger.
- ‘***LarnRateSchedule***’: permite especificar la disminución de la tasa de aprendizaje de la red con el paso de las épocas.
- ‘***LearnDropPeriod***’: número de épocas para disminuir el *learning rate*.
- ‘***LearnDropFactor***’: factor de disminución del *learning rate*. Cada X épocas indicadas en ‘*LearnDropPeriod*’, se multiplica el *learning rate* actual por el factor indicado en este parámetro.

La optimización de los hiperparámetros listados anteriormente es fundamental para la obtención de buenos resultados durante los entrenamientos.

5.1.2. Desarrollo de modelos a multiresolución

Para la elaboración de los modelos de redes neuronales multiresolución, se siguen de manera muy similar, los pasos comentados en el apartado anterior. La diferencia radica en la carga de los datos, teniendo que elaborar *scripts* más complejos, y en la construcción de redes neuronales con múltiples entradas.

Para simular el proceso de inspección que realiza el especialista a la hora de analizar las muestras, se introduce en la red neuronal parches de imágenes a tres resoluciones diferentes, 5x, 10x y 20x. Como se explicó en el punto 4.1.4, los parches deben de pertenecer a la misma región de la imagen WSI original. En este caso, se deben de crear cuatro *datastore*, tres para cada resolución de imagen y uno para las etiquetas correspondientes de cada imagen.

Una vez se han creado los *imageDatastore* para cada una de las resoluciones, y para cada uno de los sets de datos de entrenamiento y validación marcados en los *k-fold*, se pasa al diseño de la red neuronal. Para realizar este diseño se ha contado con la herramienta *Deep Network Designer*, cuya interfaz de puede apreciar el la Figura 5.1, y permite importar, editar redes neuronales, construir nuevas redes, añadir nuevas capas, crear nuevas conexiones, combinar redes, editar las propiedades de las capas o generar el código de la red creada, entre otras funcionalidades [35].

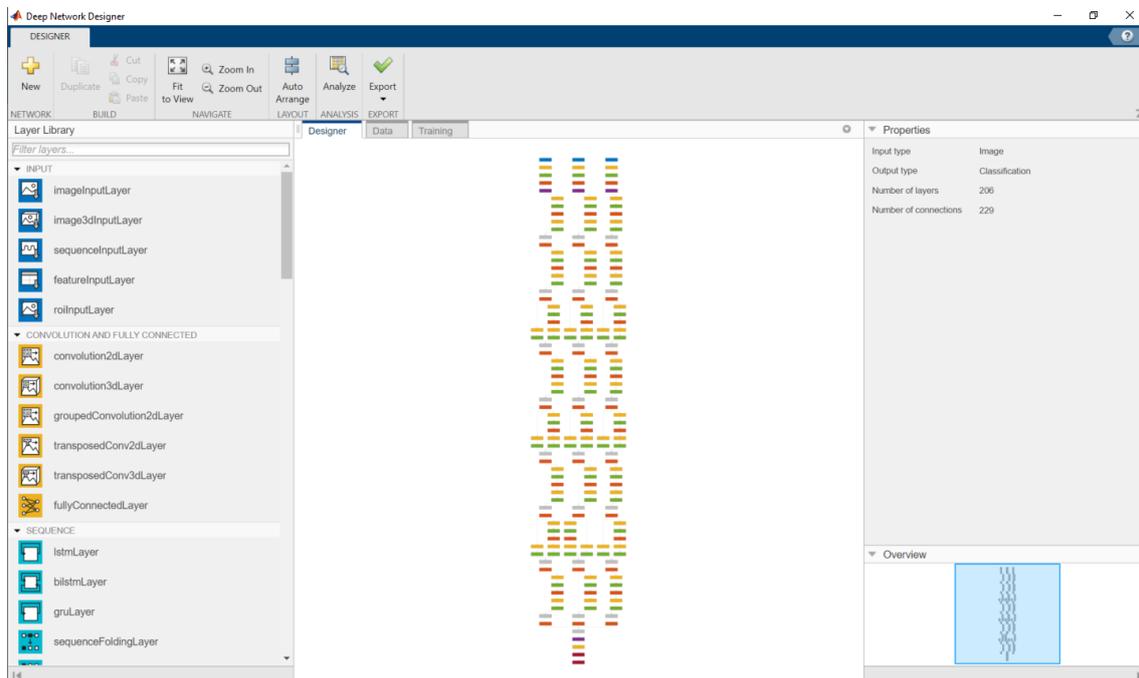


Figura 5.1: Interfaz gráfica de la herramienta *Deep Network Designer* de MATLAB. En la herramienta se encuentra cargada una red multiresolución. Imagen extraída de MATLAB R2021a.

Para crear la red multiresolución, se ha contado con tres *base model* independientes, donde cada uno se ha empleado en la extracción de características de una única resolución (lo que se co-

noce como técnica de *late fusion*). Posteriormente, la salida de cada *base model* se ha concatenado, fusionando así las características de cada una de las entradas. De esta forma, se obtiene una red neuronal que extrae las características de los parches a distintas resoluciones de forma totalmente independiente, con el fin de obtener una única predicción. En la Figura 5.2 se pueden ver las 11 primeras capas de las tres redes neuronales que forman la red a multiresolución, donde se aprecian las tres entradas a la red.



Figura 5.2: Ejemplo de las 11 primeras capas una red multiresolución formada por tres redes RESNET18. Elaboración propia.

5.2. Métricas para la validación de los modelos

Con el fin de evaluar la validez de los modelos desarrollados, se utilizan una serie de métricas. Para calcular estas métricas se emplean los valores: verdadero positivo (VP), falso positivo (FP), verdadero negativo (VN) y falso positivo (FP) (ver Figura 5.3).

| | | Valor predicho | |
|------------|-------------|-------------------------|-------------------------|
| | | Spitz nevus | Melanoma |
| Valor real | Spitz nevus | Verdadero Positivo (VP) | Falso Positivo (FP) |
| | Melanoma | Falso Negativo (FN) | Verdadero Negativo (VN) |

Figura 5.3: Relación entre los valores reales y los valores predichos. Elaboración propia.

Las métricas que se han utilizado para la validación de los modelos son:

- **Accuracy (ACC):** indica la precisión que tiene el modelo a la hora de clasificar las imágenes.

$$Accuracy = \frac{VP + VN}{VP + VN + FP + FN} \quad (5.1)$$

- **Sensibilidad:** probabilidad de clasificar de forma correcta una muestra positiva, en este caso muestras con melanoma.

$$Sensibilidad = \frac{VP}{VP + FN} \quad (5.2)$$

- **Especificidad:** probabilidad de clasificar de forma correcta una muestra negativa, en este caso muestras con Spitz nevus.

$$Especificidad = \frac{VN}{VN + FP} \quad (5.3)$$

- **Valor predictivo positivo (VPP):** refleja la eficacia real del modelo cuando el resultado de clasificación ha sido positivo, es decir, la probabilidad de padecer melanoma cuando el resultado ha sido positivo.

$$VPP = \frac{VP}{VP + FP} \quad (5.4)$$

- **Valor predictivo negativo (VPN):** refleja la eficacia real del modelo cuando el resultado de clasificación ha sido negativo, es decir, la probabilidad de padecer Spitz nevus cuando el resultado ha sido negativo.

$$VPN = \frac{VN}{VN + FN} \quad (5.5)$$

- **F1-score:** relaciona la precisión que tiene un modelo respecto a su sensibilidad.

$$F1 - score = \frac{2 \cdot VPP \cdot Sensibilidad}{VPP + Sensibilidad} \quad (5.6)$$

- **Area Under Curve (AUC):** en castellano área bajo la Curva (ROC), mide la capacidad que presenta el modelo de distinguir muestras sanas frente a enfermas. En este caso, la AUC mide la capacidad que tiene el modelo para discernir entre muestras con Spitz nevus y muestras con lesiones de melanomas. La curva ROC representa la tasa de verdaderos positivos frente a la tasa de falsos positivos. Cuanto mayor es este valor, mayor capacidad discriminadora presenta el modelo.

Para validar los modelos de forma correcta, utilizando estas métricas, se han empleado tres resultados diferentes:

- **Resultado de referencia** (*reference result*): es el valor de las métricas por cada set de datos de manera individual, por lo que se tendrá 4 valores distintos, uno por entrenamiento.
- **Resultado micro**: se tiene en cuenta el total de las imágenes, es decir, no se diferencia por set de datos, la media se computa con todas las imágenes clasificadas en el archivo `.xlsx`.
- **Resultado macro**: se realiza la media de los resultados de cada set de datos, es decir, la media de los cuatro valores de *reference result*.

5.3. Experimentación, resultados y discusión

En esta última sección del capítulo 5 se detallan los experimentos que se han realizado con distintas redes pre-entrenadas, así como los resultados obtenidos.

5.3.1. Fase de optimización: experimentos realizados

Las pruebas realizadas se dividen en dos: las modelos donde únicamente se emplea una única resolución de imágenes y modelos que utilizan tres resoluciones diferentes. Para ello, se han empleado datos de entrenamiento y de validación, tal y como se recoge en las Tablas 4.1 y 4.2.

- Experimentos con una única resolución

Los primeros experimentos se han realizado con modelos empleando imágenes a una resolución 5x. Para ello, se han utilizado variantes de las redes VGG, RESNET e Inception, siendo estas las más empleadas en el desarrollo de modelos de clasificación automáticos de imágenes, debido a su buen rendimiento. Concretamente, se han empleado las redes: VGG16, RESNET18 e InceptionV3. Además de estas redes, se ha utilizado una red pre-entrenada con imágenes histopatológicas, denominada DeLogu y desarrollada por los autores de [36]. Este modelo está basado en Inception-Resnetv3 y entrenado con distintas muestras de melanomas. Todas estas redes neuronales cuentan con un tamaño de entrada de 224x224 píxeles, por lo que es necesario redimensionar las imágenes antes de introducirlas en la red.

Todos los experimentos se han realizado variando el número de capas congeladas, el optimizador a utilizar (*sgdm*, *rmsprop* o *adam*), el tamaño de *mini-batch*, el número máximo de épocas, el *learning rate* inicial, así como su disminución paulatina y la frecuencia de validación. A continuación se describen las distintas variaciones de estos hiperparámetros en cada una de las redes neuronales.

Experimentos con VGG16 (cuya arquitectura se puede observar en la Figura 5.4):

- Congelar 1, 2, 5, 10, 20 y 37 (*base model* completo) capas de la red.
- Optimizadores *sgdm* y *adam*.
- Tamaño del *mini-batch* 10, 20, 32 y 64.

- Número máximo de épocas comprendido entre 8 y 20.
- Variación del hiperparámetro *learning rate* inicial, con valores comprendidos entre $5 \cdot 10^{-2}$ y $5 \cdot 10^{-4}$.

El mejor resultado de *accuracy* para el set de validación, se obtuvo con el optimizador *sgdm*, un tamaño de *mini-batch* de 10, un total de 20 épocas y un *learning rate* inicial de $1 \cdot 10^{-4}$. Se consiguió un *accuracy* del 48.33 % de media entre los cuatro *sets* de *k-fold*.

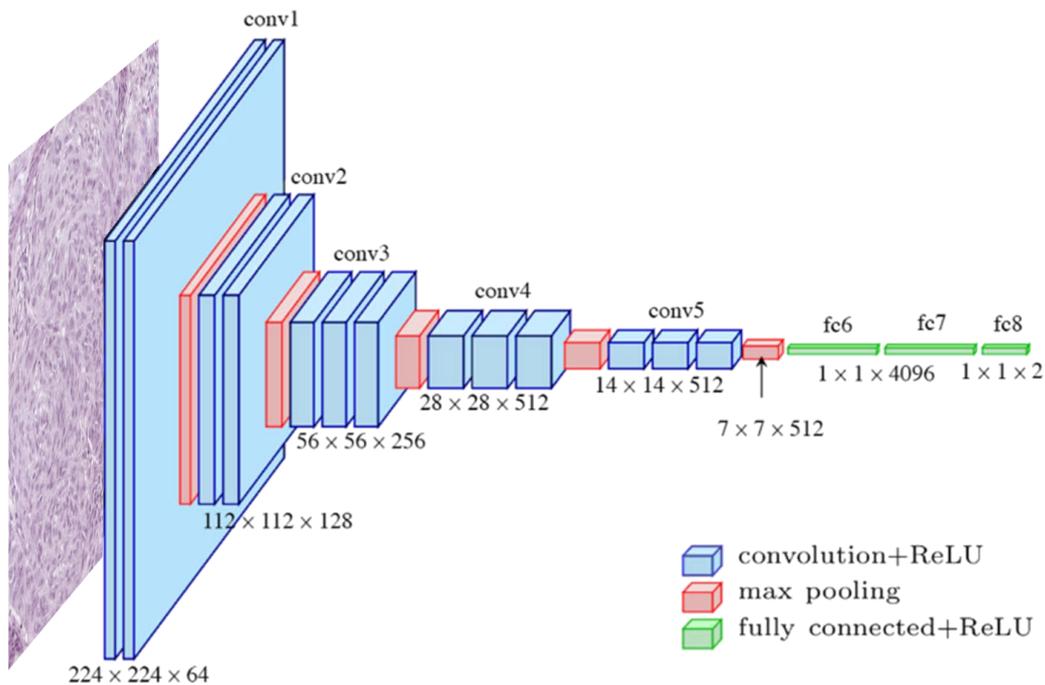


Figura 5.4: Arquitectura de la red neuronal VGG16. Modificación de [37].

Con RESNET18 (cuya arquitectura se puede ver en la Figura 5.5) se han realizado experimentos similares a los anteriores:

- Congelar 1, 5, 10, 20 y 63 (*base model* completo salvo la fase convolucional) capas de la red.
- Optimizadores *sgdm*, *rmsprop* y *adam*.
- Tamaño del *mini-bacth* 10, 20, 32, 64 y 256.
- Número máximo de épocas comprendido entre 7 y 200.
- Variación del hiperparámetro *learning rate* inicial, con valores comprendidos entre $5 \cdot 7^{-2}$ y $7 \cdot 10^{-4}$.
- Disminución gradual del *learning rate* en un factor comprendido entre 0.3 y 0.5.

El mejor resultado de *accuracy* para el set de validación, se obtuvo con el optimizador *rms-prop*, un tamaño de *mini-batch* de 64, un total de 50 épocas (convergencia en las 13 primeras), un *learning rate* inicial de $9 \cdot 10^{-2}$ y con una disminución del *learning rate* a la mitad cada 5 épocas. Se consiguió un *accuracy* del 56.06 % de media entre los cuatro *sets* de *k-fold* (69.19 %, 33.49 %, 67.55 % y 54.02 %, respectivamente). En la Figura 5.7 se puede ver las gráficas de los entrenamientos de cada iteración.

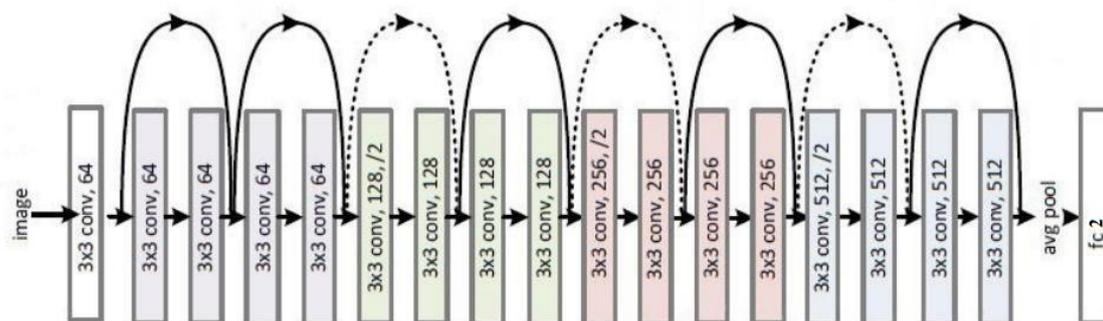


Figura 5.5: Arquitectura de la red neuronal RESNET18. Figura extraída de [38]

Al igual que con las anteriores, las pruebas realizadas con Inceptionv3 (cuya arquitectura se puede observar en la Figura 5.6) han sido:

- Congelar 1, 10 y 100 capas de la red.
- Optimizadores *sgdm* y *rmsprop*.
- Tamaño del *mini-batch* 10, 64, 100 y 256.
- Número máximo de épocas: 20 y 50.
- Variación del hiperparámetro *learning rate* inicial, con valores comprendidos entre $5 \cdot 5^{-2}$ y $5 \cdot 10^{-3}$.
- Disminución gradual del *learning rate* en un factor comprendido entre 0.3 y 0.9.

El mejor resultado de *accuracy* para el set de validación, se obtuvo con el optimizador *sgdm*, un tamaño de *mini-batch* de 100, un total de 50 épocas (convergencia en las 6 primeras), un *learning rate* inicial de $9 \cdot 10^{-3}$. Se consiguió un *accuracy* del 55.84 % de media entre los cuatro *sets* de *k-fold*.

Por último, se han realizado experimentos con la red DeloGu, pre-entrenada con imágenes histopatológicas de melanomas cutáneos. Los experimentos que se han sido:

- Congelar 1, 10 y 100 capas de la red.
- Optimizador *sgdm*.
- Tamaño del *mini-batch* 10, 64, 100 y 256.
- Número máximo de épocas: 16, 20 y 50.

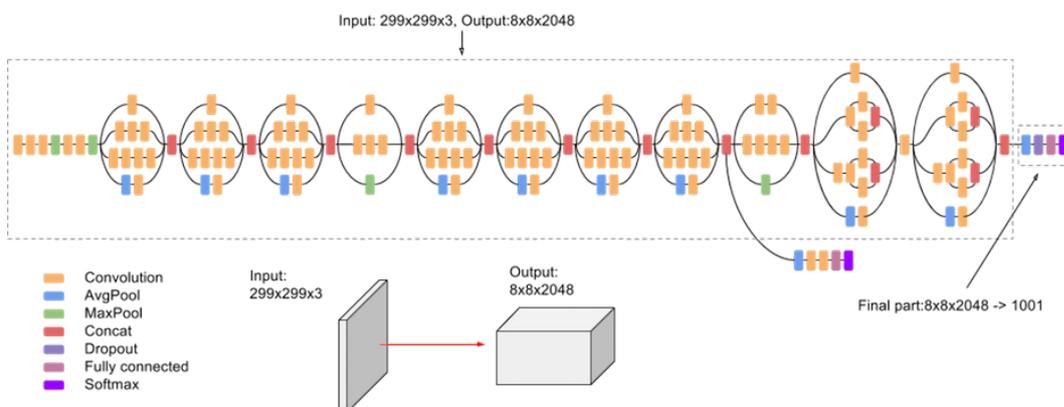


Figura 5.6: Arquitectura red neural Inceptionv3. Fuente: [39].

- Variación del hiperparámetro *learning rate* inicial, con valores comprendidos entre $3 \cdot 10^{-3}$ y $9 \cdot 10^{-3}$.
- Disminución gradual del *learning rate* en un factor comprendido entre 0.7 y 1.
- Factor de *momentum* de 0.9.

El mejor resultado de *accuracy* para el set de validación se obtuvo con el optimizador *sgdm*, un tamaño de *mini-batch* de 15, un total de 20 épocas (convergencia en las 6 primeras), un *learning rate* inicial de $3 \cdot 10^{-3}$ y un factor de *momentum* de 0.9. Se consiguió un *accuracy* del 55.50 % de media entre los cuatro *sets* de *k-fold*.

Comparando los resultados ofrecidos por las cuatro redes empleadas, RESNET18 es la que mejor *accuracy* presenta a la hora de clasificar las imágenes. En la Tabla 5.1, se reflejan los hiperparámetros, el *top model* empleado y los resultados obtenidos con el mejor modelo. Además, en la Figura 5.7 se muestran las gráficas de la evolución del *accuracy* y las pérdidas durante el entrenamiento de la red.

Con estos hiperparámetros y *top model* se han realizado los entrenamientos a resolución 10x y 20x, cuyos resultados se muestran en las Tablas 5.2 y 5.3 respectivamente.

- Experimentos con multiresolución

Como se ha comentado en el apartado 5.1.2, las redes neuronales multiresolución elaboradas, están formadas por la combinación de tres *base model*, una capa de concatenación y un *top model*.

Para la elaboración de las redes multiresolución, se ha utilizado RESNET18, ya que es la red que mejores resultados ha ofrecido en una única resolución.

En los distintos experimentos con RESNET18, se han ido ajustando los hiperparámetros, el *top model* y los pesos iniciales del *base model*. A continuación se muestran los experimentos más destacados realizados.

- **Test 1:** Se ha obtenido un 76.93 % de *accuracy* medio entre las cuatro iteraciones del *k-fold* (74.09 %, 79.88 %, 91.65 % y 62.08 %). Para ello se ha empleado:

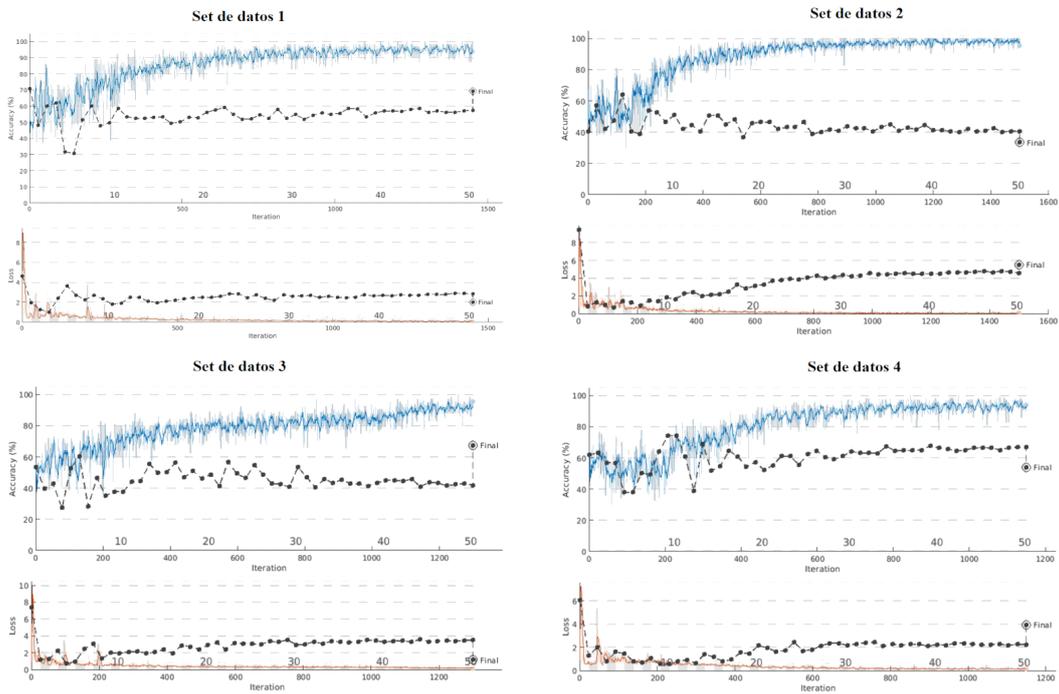


Figura 5.7: Gráficas del entrenamiento del mejor *accuracy* logrado con RESNET18.

- Red neuronal construida con el mismo *base model* que el resultante del entrenamiento que proporcionó el mejor resultado a resolución 5x (*accuracy* del 67.55 %), pero añadiendo un *global average pooling* al final de cada rama. El *top model* está formado por: *depth concatenation layer*, *fully connected layer*, *softmax* y la capa de clasificación.
 - Optimizador *rmsprop*.
 - Tamaño de *mini-batch* igual a 32.
 - Número máximo de épocas igual a 20.
 - *Learning rate* inicial igual a $5 \cdot 10^{-3}$, con un factor de disminución progresiva de 0.5.
 - Validación del modelo cada 400 iteraciones.
- **Test 2:** Se ha obtenido un 63.00 % de *accuracy* medio entre las cuatro iteraciones del *k-fold* (67.86 %, 49.17 %, 86.14 % y 48.83 %). Para ello se ha empelado:
- Red neuronal construida con los mismos *base models* que los resultantes del entrenamiento que proporcionó mejor resultado a resolución 5x, es decir, para cada iteración se ha utilizado el *base model* entrenado con el mismo *set* de datos que anteriormente. El *top model* está formado por: *depth concatenation layer*, *global average pooling*, *fully connected layer*, *softmax* y la capa de clasificación.
 - Optimizador *sgdm*.
 - Tamaño de *mini-batch* igual a 32.
 - Número máximo de épocas igual a 20.
 - *Learning rate* inicial igual a $2 \cdot 10^{-3}$, con un factor de disminución progresiva de 0.5.

| Resolución 5x | | | | | | |
|-------------------------------|-------------------|--------|--------|------------|--------|--------|
| Hiperparámetros | | | | | | |
| Optimizador | <i>rmsprop</i> | | | | | |
| Tamaño <i>mini-batch</i> | 64 | | | | | |
| Número épocas | 50 | | | | | |
| <i>Learning rate</i> inicial | $9 \cdot 10^{-2}$ | | | | | |
| <i>Top model</i> empleado | | | | | | |
| Nombre capa | | | | Activación | | |
| <i>Global Average Pooling</i> | | | | 1x1x512 | | |
| <i>Fully Connected Layer</i> | | | | 1x1x2 | | |
| <i>Softmax</i> | | | | 1x1x2 | | |
| <i>Classification Layer</i> | | | | 1x1x2 | | |
| Clasificación | | | | | | |
| | Set 1 | Set 2 | Set 3 | Set 4 | Micro | Macro |
| ACC | 0.6919 | 0.3349 | 0.6755 | 0.5402 | 0.5721 | 0.5606 |
| Sensibilidad | 0.8051 | 0.2317 | 0.7327 | 0.6487 | 0.6576 | 0.6046 |
| Especificidad | 0.4173 | 0.4058 | 0.6095 | 0.4738 | 0.4899 | 0.4766 |
| VPP | 0.7701 | 0.2111 | 0.6838 | 0.4297 | 0.5533 | 0.5237 |
| VPN | 0.4690 | 0.4349 | 0.6643 | 0.6882 | 0.5983 | 0.5641 |
| F1-score | 0.7873 | 0.2209 | 0.7074 | 0.5170 | 0.5197 | 0.5581 |
| AUC | 0.67 | 0.23 | 0.7 | 0.59 | - | 0.56 |

Tabla 5.1: Resultados del mejor modelo de RESNET18 durante la clasificación del set de validación, resolución 5x.

- Validación del modelo cada 400 iteraciones.
- **Test 3:** Se ha obtenido un 64.65 % de *accuracy* medio entre las cuatro iteraciones del *k-fold* (52.87 %, 56.77 %, 87.96 % y 61.16 %). Para ello se ha empleado:
 - Red neuronal construida con los mismos *base models* que los resultantes de la mejor red a resolución x5, es decir, para cada iteración se ha utilizado el *base model* entrenado con el mismo *set* de datos que anteriormente. El *top model* está formado por: *depth concatenation layer*, *global average pooling*, *dropout layer* con una probabilidad de 0.5, *fully connected layer*, *softmax* y la capa de clasificación.
 - Optimizador *sgdm*.
 - Tamaño de *mini-batch* igual a 32.
 - Número máximo de épocas igual a 10.
 - *Learning rate* inicial igual a $1 \cdot 10^{-4}$, con un factor de disminución progresiva de 0.5 cada 2 épocas.
 - Validación del modelo cada 400 iteraciones.
- **Test 4:** Se ha obtenido un 65.48 % de *accuracy* medio entre las cuatro iteraciones del *k-fold* (65.78 %, 68.07 %, 85.62 % y 42.58 %). Para ello se ha empleado:

| Resolución 10x | | | | | | |
|----------------|--------|--------|--------|--------|--------|--------|
| | Set 1 | Set 2 | Set 3 | Set 4 | Micro | Macro |
| ACC | 0.6643 | 0.5657 | 0.4675 | 0.4898 | 0.5287 | 0.5468 |
| Sensibilidad | 0.8084 | 0.0365 | 0.0137 | 0.1360 | 0.2632 | 0.2487 |
| Especificidad | 0.3149 | 0.9288 | 0.9904 | 0.7060 | 0.7838 | 0.7350 |
| VPP | 0.7410 | 0.2608 | 0.6250 | 0.2205 | 0.5391 | 0.4618 |
| VPN | 0.4040 | 0.5842 | 0.4656 | 0.5721 | 0.5254 | 0.5065 |
| F1-score | 0.7732 | 0.0641 | 0.0269 | 0.1683 | 0.6388 | 0.2581 |
| AUC | 0.58 | 0.36 | 0.097 | 0.43 | - | 0.37 |

Tabla 5.2: Resultados de clasificación del set de validación con RESNET18, resolución 10x.

| Resolución 20x | | | | | | |
|----------------|--------|--------|--------|--------|--------|--------|
| | Set 1 | Set 2 | Set 3 | Set 4 | Micro | Macro |
| ACC | 0.2897 | 0.4988 | 0.3333 | 0.7131 | 0.4883 | 0.4587 |
| Sensibilidad | 0.2078 | 0.1280 | 0.3416 | 0.4462 | 0.3041 | 0.2809 |
| Especificidad | 0.4882 | 0.7531 | 0.3238 | 0.8762 | 0.6653 | 0.6103 |
| VPP | 0.4961 | 0.2625 | 0.3680 | 0.6878 | 0.4660 | 0.4536 |
| VPN | 0.2026 | 0.5573 | 0.2991 | 0.7213 | 0.4987 | 0.4451 |
| F1-score | 0.2929 | 0.1721 | 0.3543 | 0.5413 | 0.5481 | 0.3401 |
| AUC | 0.30 | 0.43 | 0.30 | 0.59 | - | 0.40 |

Tabla 5.3: Resultados de clasificación del set de validación con RESNET18, resolución 20x.

- Red neuronal construida con el *base model* y los pesos originales de la red RESNET18, cuyas 10 primeras capas se encuentran congeladas. El *top model* está formado por: *depth concatenation layer*, *global average pooling*, *dropout layer* con una probabilidad de 0.5, *fully connected layer*, *softmax* y la capa de clasificación.
 - Optimizador *sgdm*.
 - Tamaño de *mini-batch* igual a 32.
 - Número máximo de épocas igual a 80.
 - *Learning rate* inicial igual a $1 \cdot 10^{-3}$.
 - Validación del modelo cada 400 iteraciones.
 - Parámetro *ValidationPatience* fijado a 10 épocas.
- **Test 5:** Se ha obtenido un 67.51 % de *accuracy* medio entre las cuatro iteraciones del *k-fold* (69.94 %, 70.30 %, 85.66 % y 44.12 %). Para ello se ha empleado:
- Red neuronal construida con el *base model* y los pesos originales de la red RESNET18, donde al final de cada uno se ha añadido un *global max pooling*. El *top model* está formado por: *depth concatenation layer*, *dropout layer* con una probabilidad de 0.5, *fully connected layer*, *softmax* y la capa de clasificación.
 - Optimizador *sgdm*.
 - Tamaño de *mini-batch* igual a 32.

- Número máximo de épocas igual a 80.
 - *Learning rate* inicial igual a $1 \cdot 10^{-4}$.
 - Validación del modelo cada 400 iteraciones.
 - Parámetro *ValidationPatience* fijado a 10 épocas.
- **Test 6:** Se ha obtenido un 75.28 % de *accuracy* medio entre las cuatro iteraciones del *k-fold* (65.95 %, 60.66 %, 88.13 % y 86.37 %). Para ello se ha empleado:
- Red neuronal construida con el *base model* y los pesos originales de la red RESNET18. El *top model* está formado por: *depth concatenation layer*, *globalmax pooling*, *dropout layer* con una probabilidad de 0.5, *fully connected layer*, *softmax* y la capa de clasificación.
 - Optimizador *sgdm*.
 - Tamaño de *mini-batch* igual a 32.
 - Número máximo de épocas igual a 20.
 - *Learning rate* inicial igual a $2 \cdot 10^{-3}$.
 - Validación del modelo cada 400 iteraciones.

En vista de los experimentos anteriores, el clasificador que ha presentado menores pérdidas y mayor *accuracy*, es el modelo identificado en el Test 6. En la Tabla 5.4 se muestra un resumen de los distintos hiperparámetros empleados en el entrenamiento, así como el resultado obtenidos de las distintas métricas. Este modelo desarrollado, cuya arquitectura se muestra en la Figura 5.8, presenta un total de 207 capas, de las cuales 6 se corresponden con el *top model*. Las gráficas del entrenamiento se muestran en la Figura 5.9, donde se puede ver que el modelo converge a partir de la cuarta época.

Como se puede observar, el uso de técnicas multiresolución en los modelos de *deep learning*, para la clasificación automática de imágenes, mejora de forma considerable respecto a la utilización de redes neuronales con una única resolución. El *accuracy* medio de todas las imágenes a resolución 5x clasificadas (correspondiente al resultado micro), es de 57.21 %, mientras que utilizando multiresolución, ese *accuracy* aumenta hasta un 80.25 %, lo que representa una mejora del 23.04 %. También aumentan los parámetros de sensibilidad (un 13.59 %), especificidad (un 32.42 %), VPP (un 22.44 %), VPN (un 22.49 %), F1-score (un 27.58 %) y el AUC macro (un 0.269). En la Tabla 5.5 se muestra una comparativa, de los mejores resultados obtenidos, entre la red a resolución 5x y multiresolución.

Para conocer más información sobre las predicción del modelo a multiresolución desarrollado, se han representado la curva ROC, así como la matriz de confusión de cada set de validación. Como se observa en la Figura 5.10, el set de datos 3 es el que mejor curva ROC presenta, siendo un valor cercano a la unidad (0.93). A la hora de clasificar las imágenes, distribuidas como se indica en la Tabla 4.4, se obtienen muy buenos resultados en los sets 3 y 4, siendo algo inferiores en los sets 1 y 2. La matriz de confusión se representa en la Figura 5.11.

Los resultados obtenidos no se pueden comparar de forma directa con los del estado arte, ya que existen muchos factores diferentes. Por ejemplo, el más importante de ellos, es la base de datos utilizada. Hasta el momento, no existen trabajos que empleen técnicas de multiresolución en imágenes histopatológicas de neoplasias melanocíticas spitzoides. En el presente trabajo, la finalidad es demostrar que existe una mejora de la clasificación de las imágenes WSI empleando redes

| Multiresolución | | | | | | |
|------------------------------|-------------------|--------|------------|--------|--------|--------|
| Hiperparámetros | | | | | | |
| Optimizador | <i>sgdm</i> | | | | | |
| Tamaño <i>mini-batch</i> | 32 | | | | | |
| Número épocas | 20 | | | | | |
| <i>Learning rate</i> inicial | $2 \cdot 10^{-3}$ | | | | | |
| <i>Top model</i> empleado | | | | | | |
| Nombre capa | | | Activación | | | |
| Concatenador de 3 entradas | | | 7x7x1536 | | | |
| <i>Global Max Pooling</i> | | | 1x1x1536 | | | |
| <i>Dropout 0.5</i> | | | 1x1x1536 | | | |
| <i>Fully Connected Layer</i> | | | 1x1x2 | | | |
| <i>Softmax</i> | | | 1x1x2 | | | |
| <i>Classification Layer</i> | | | 1x1x2 | | | |
| Clasificación | | | | | | |
| | Set 1 | Set 2 | Set 3 | Set 4 | Micro | Macro |
| ACC | 0.6596 | 0.6066 | 0.8813 | 0.8638 | 0.8025 | 0.7528 |
| Sensibilidad | 0.7165 | 0.8424 | 0.8793 | 0.7168 | 0.7883 | 0.7888 |
| Especificidad | 0.5121 | 0.5789 | 0.8855 | 0.9063 | 0.8142 | 0.7207 |
| VPP | 0.7918 | 0.1907 | 0.9406 | 0.6890 | 0.7779 | 0.6530 |
| VPN | 0.4110 | 0.9689 | 0.7806 | 0.9170 | 0.8233 | 0.7694 |
| F1-score | 0.7523 | 0.3110 | 0.9089 | 0.7026 | 0.7956 | 0.6687 |
| AUC | 0.68 | 0.86 | 0.93 | 0.89 | - | 0.84 |

Tabla 5.4: Resultados de la clasificación del set de validación, del modelo RESNET18 multi-resolución.

| Métrica | Resolución 5x | | Multiresolución | |
|---------------|---------------|--------|-----------------|---------------|
| | Micro | Macro | Micro | Macro |
| ACC | 0.5721 | 0.5606 | 0.8025 | 0.7528 |
| Sensibilidad | 0.6576 | 0.6046 | 0.7883 | 0.7888 |
| Especificidad | 0.4899 | 0.4766 | 0.8142 | 0.7207 |
| VPP | 0.5533 | 0.5237 | 0.7779 | 0.6530 |
| VPN | 0.5983 | 0.5641 | 0.8233 | 0.7694 |
| F1-score | 0.5197 | 0.5581 | 0.7956 | 0.6687 |
| AUC | - | 0.56 | - | 0.84 |

Tabla 5.5: Comparativa de la clasificación del set de validación con las redes RESNET18, empleando resolución 5x y multiresolución.

neuronales multiresolución. En los estudios de [5], [36], [8] y [32], consiguen buenos resultados aplicando técnicas multiresolución, pero todos en ellos se emplean distintas técnicas y diferentes datos.

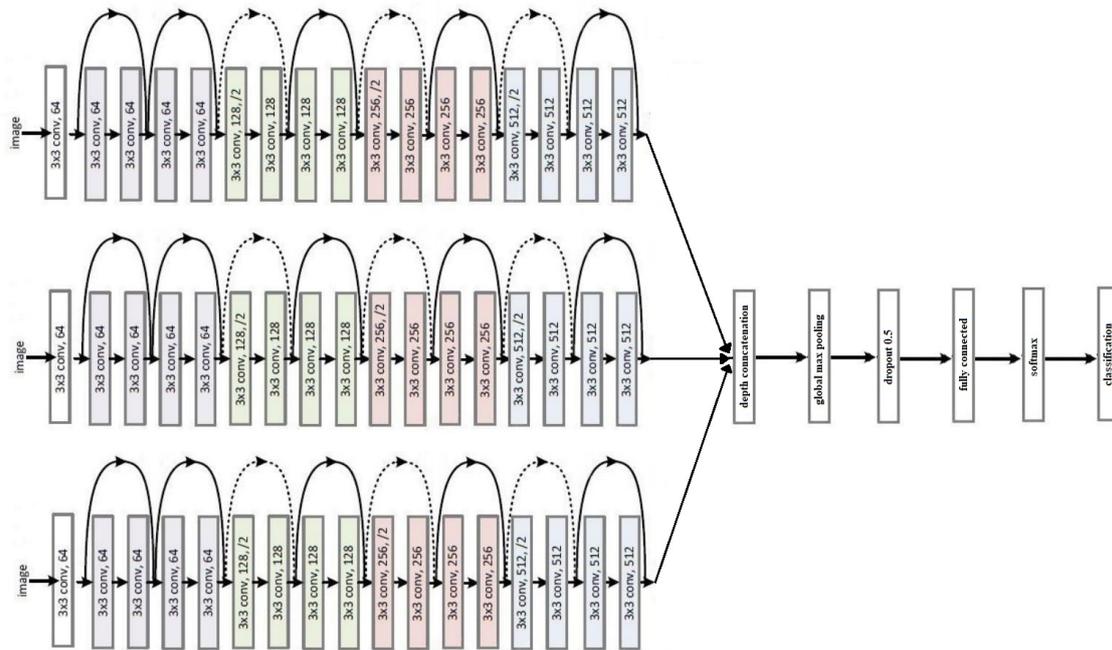


Figura 5.8: Arquitectura de la red RESNET18 multiresolución.

5.3.2. Fase de testeo

Para evaluar de forma correcta el funcionamiento de las redes neuronales, se ha realizado un test de clasificación con muestras que la red nunca ha visto. Para ello, se ha contado con un total de 11 WSI, donde cada una representa un paciente diferente. En este test, se ha evaluado el funcionamiento de las redes a una única resolución (5x, 10x y 20x) con los hiperparámetros y *top model* mostrados en la Tabla 5.1, y las redes neuronales multiresolución entrenadas con los hiperparámetros y el *top model* mostrados en la Tabla 5.4.

Cada uno de los distintos modelos: resolución 5x, 10x, 20x y multiresolución, ha sido entrenado y validado con los datos indicados en el apartado 4.1.5. A la hora de realizar la clasificación de los datos de test, el cual está dividido entre muestras de melanoma y Spitz nevus, se ha obtenido un *accuracy* macro del 50.88 % para la resolución 5x, 53.54 % para la resolución 10x, 48.65 % para la resolución 20x, y un 58.16 % para multiresolución. Como se aprecia, a la hora de evaluar los distintos modelos, se logra un 4.62 % más de *accuracy* aplicando la técnica multiresolución, respecto al empleo de una única resolución, que en este caso es la red 10x la que consigue un mejor resultado. En la Tabla 5.6, se muestra un resumen de los resultados de las métricas empleadas en la validación de estos modelos.

Todos los modelos presentan un valor de VPP superior a VPN, es decir, clasifican de manera más precisas aquellos parches negativos (Spitz nevus), que los positivos (melanoma). Esto se debe al alto grado de complejidad que presentan las muestras de melanoma, haciendo que sus características sean más difíciles de diferenciar respecto a las Spitz nevus. Únicamente en los valores de especificidad micro y macro, y VPP micro, el modelo multiresolución es superado por los modelos 10x y 5x respectivamente.

Finalmente, se ha comprobado la precisión de clasificación a nivel de paciente que ofrece la

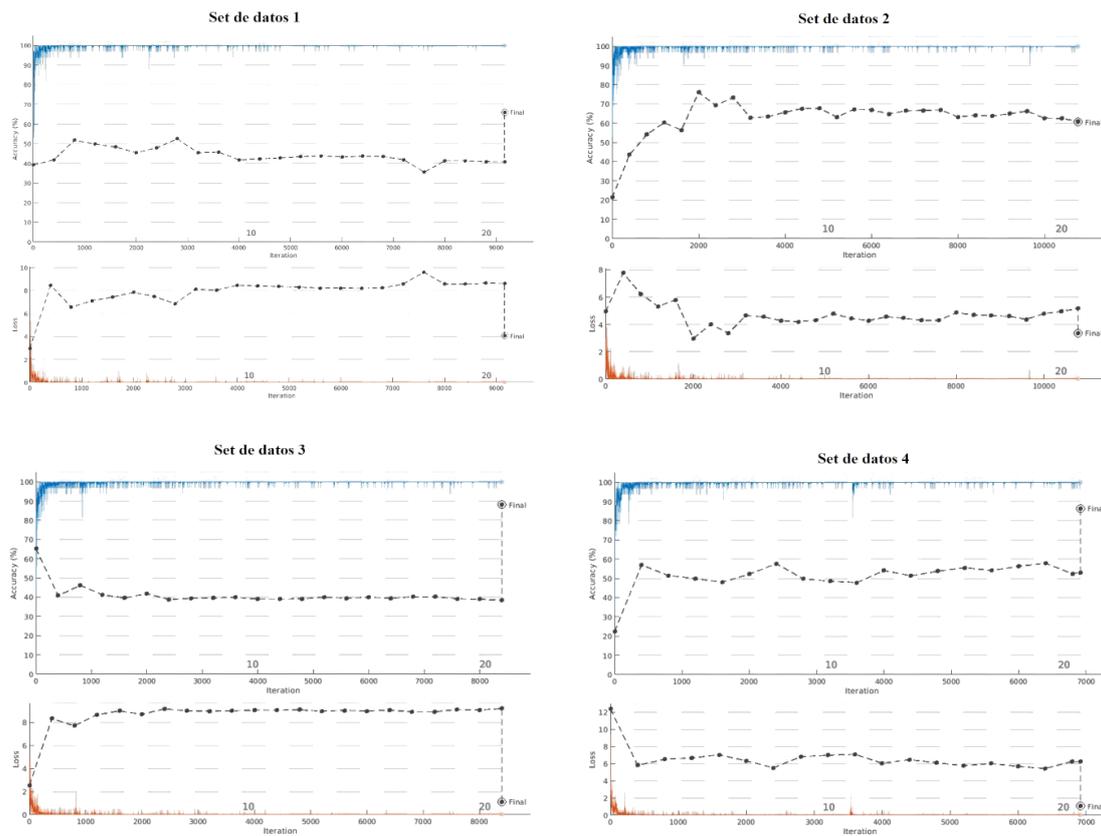


Figura 5.9: Gráficas del entrenamiento del mejor modelo a multiresolución.

red que mejores resultados ha dado, que en este caso es la red a multiresolución. Para ello, se ha calculado la probabilidad media de que un parche sea clasificado como melanoma, entre las cuatro redes neuronales empleadas. A partir de este valor, se ha determinado la clase a la que pertenece ese parche, es decir, si la probabilidad es superior al 50 %, ese parche se tomará como positivo. Dentro de cada muestra, se ha contabilizado el número total de parches que pertenecen a cada una de las clases, y finalmente, la muestra se ha clasificado en función del mayor número de parches pertenecientes a una clase. Aplicando este criterio, se ha obtenido que 8 de las 11 muestras, han sido clasificadas correctamente, lo que supone un 72.72 % de *accuracy*. Analizando los resultados obtenidos, se observa que la red ha clasificado con mayor precisión los parches que no presentan melanoma, como se puede apreciar en la matriz de confusión de la Figura 5.12. En la Tabla 5.7 se muestran los resultados de la clasificación a nivel de paciente empleando el modelo multiresolución.

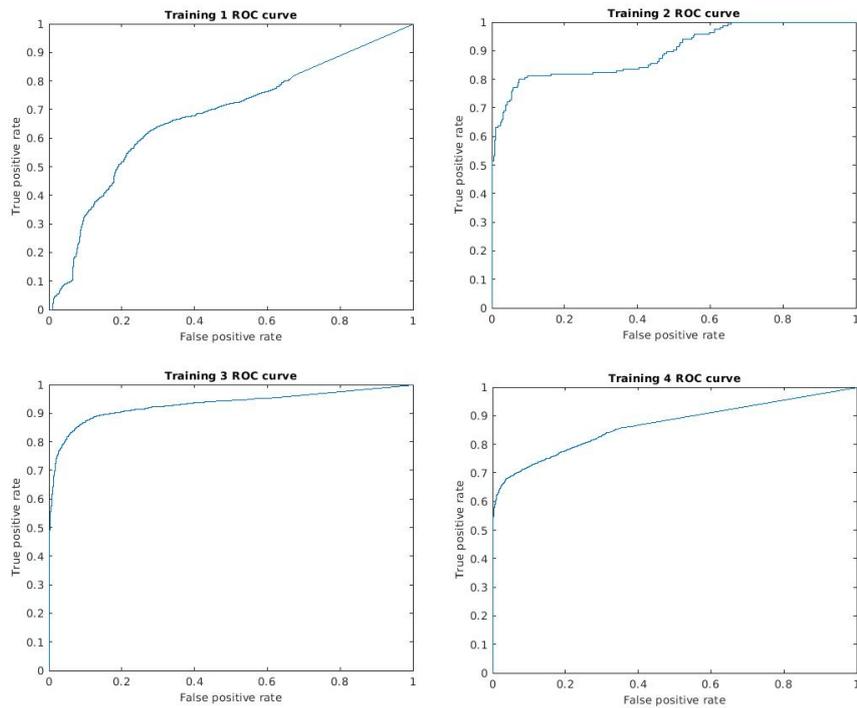


Figura 5.10: Curvas ROC del los distintos entrenamientos del modelo a multiresolución.

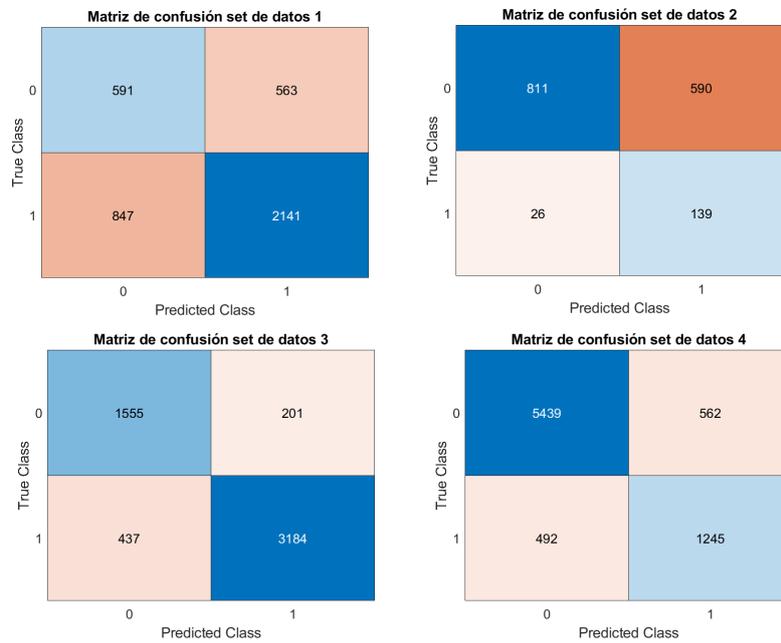


Figura 5.11: Matriz de confusión de los datos de validación, en cada uno distintos entrenamientos del modelo a multiresolución.

| Test de los distintos modelos | | | | | | | | |
|-------------------------------|---------------|--------|---------------|---------------|------------|--------|------------------------|---------------|
| | Modelo 5x | | Modelo 10x | | Modelo 20x | | Modelo multiresolución | |
| | Micro | Macro | Micro | Macro | Micro | Macro | Micro | Macro |
| ACC | 0.5089 | 0.5089 | 0.5354 | 0.5354 | 0.4866 | 0.4866 | 0.5816 | 0.5816 |
| Sensibilidad | 0.5406 | 0.5406 | 0.4431 | 0.4431 | 0.6228 | 0.6228 | 0.7052 | 0.7052 |
| Especificidad | 0.4955 | 0.4955 | 0.5703 | 0.5703 | 0.4392 | 0.4392 | 0.5464 | 0.5464 |
| VPP | 0.3116 | 0.3104 | 0.2802 | 0.2791 | 0.2788 | 0.2779 | 0.3070 | 0.3105 |
| VPN | 0.7186 | 0.7213 | 0.7307 | 0.7425 | 0.7698 | 0.7732 | 0.8668 | 0.8724 |
| F1-score | 0.3826 | 0.3903 | 0.3757 | 0.3258 | 0.3411 | 0.3836 | 0.3931 | 0.4241 |
| AUC | - | 0.5071 | - | 0.4964 | - | 0.5345 | - | 0.6625 |

Tabla 5.6: Métricas obtenidas del testeo de los distintos modelos elaborados.

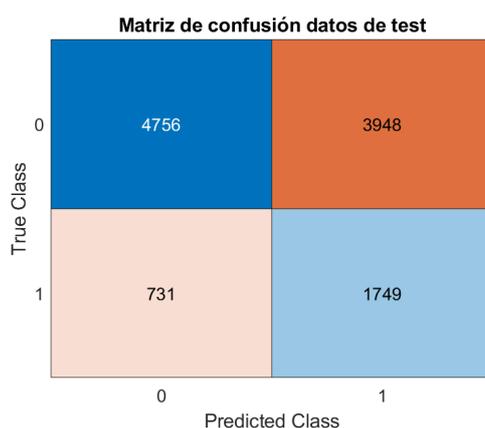


Figura 5.12: Matriz de confusión de los datos de test empleando los modelos multiresolución.

| Clasificación a nivel de paciente | |
|-----------------------------------|--------|
| ACC | 0.7272 |
| Sensibilidad | 0.3333 |
| Especificidad | 0.8750 |
| VPP | 0.500 |
| VPN | 0.7777 |
| F1-score | 0.4000 |

Tabla 5.7: Resultados de la clasificación a nivel de paciente empleando el modelo multiresolución.

Capítulo 6

Conclusiones y líneas futuras

6.1. Conclusiones

Durante el desarrollo del presente trabajo fin de máster, se han logrado los objetivos señalados en la sección 2.2. En primer lugar, se ha realizado un estudio sobre las neoplasias melanocíticas spitzoides, que suponen un gran desafío en su diagnóstico. Dentro de este tipo de lesiones, existen neoplasias benignas como el nevus Spitz, neoplasias malignas como el melanoma spitzoide, o neoplasias que presentan características de tumores benignos y malignos, conocidas como tumores atípicos de Spitz. La correcta clasificación de estas lesiones es de vital importancia, ya que los melanomas spitzoides pueden presentar un comportamiento con alto potencial metastásico. Por ello, este trabajo pretende evaluar y desarrollar sistemas automáticos para la clasificación de estas lesiones, aplicando técnicas multiresolución basadas en *deep learning* sobre *Whole-Slide Images*.

Una vez estudiadas las características morfológicas de las neoplasias melanocíticas spitzoides, se ha procedido con la elaboración de la base de datos. Para ello, se ha contado con diversas muestras de imágenes histopatológicas en formato WSI, que se han empleado para la construcción de tres *datasets*, cada uno de ellos con imágenes a distinta resolución (5x, 10x y 20x). Las WSI, se han dividido en parches con un tamaño de 512x512 píxeles y un solapamiento del 50 % entre ellos. Este solapamiento se aplica para lograr un mayor número de imágenes dentro del *dataset*, ya que actualmente no se dispone de un gran número de muestras. Posteriormente, se han elegido aquellos parches que poseen un mínimo de tejido previamente marcado por el especialista como región de interés. Uno de los pasos más importante en este trabajo, es la asociación de los parches entre las diferentes resoluciones. Para ello, se ha desarrollado un algoritmo que relaciona los parches de una mayor resolución, con los parches de menor resolución que lo contienen.

Con la base de datos elaborada, se ha procedido al desarrollo de los algoritmos para el entrenamiento de redes neuronales a partir de imágenes a una única resolución. En este paso, se han empleado distintas redes ya pre-entrenadas, utilizando la técnica conocida como *transfer learning*. Para evitar *overfitting* en los entrenamientos, se ha empleado *k-fold cross validation*.

Para el desarrollo de los algoritmos de *deep learning*, aplicando técnicas multiresolución, se han empleado redes híbridas, las cuales permiten la entrada de varios datos independientes a la red neuronal. En este tipo de redes, es posible fusionar los datos en las primeras capas de la red, o al final, una vez ya se han extraído las características de cada imagen de entrada. En este caso, se ha empleado la técnica *late fusion*, donde existen diversos *base models* independientes entre si,

encargados de extraer las características de cada imagen a distinta resolución de manera independiente. Se ha elegido esta técnica, debido a que en diversos trabajos del estado del arte analizados, ha ofrecido buenos resultados. Posteriormente, los tres volúmenes de características son concatenados e introducidos en el *top model* de la red, donde se realiza su clasificación. Los resultados obtenidos con las redes neuronales multiresolución, mejoran los resultados logrados con las redes a una única resolución.

A la hora de realizar el testeo de la red con nuevos datos, se ha verificado que el empleo de técnicas multiresolución, mejoran los resultados obtenidos con una única resolución, lo cual se satisface. En este caso, el *accuracy* de test disminuye respecto al de validación, y eso es debido a la alta complejidad de los datos utilizados. Es por ello, que para mejorar estos modelos, será necesario disponer de un mayor número de muestras, lo que contribuye a la creación de modelos de clasificación más robustos.

El empleo de redes neuronales multiresolución, aporta una mayor información sobre los datos de entrada, ya que debido a las distintas capas convolucionales y de *pooling*, mucha información visual se pierde. Además, con este tipo de redes, se pretende simular el proceso que realiza un patólogo en el diagnóstico de un paciente, analizando la muestra a distintos niveles de detalle.

Por tanto, se ha demostrado que la utilización de redes multiresolución, obtiene una mejor clasificación de imágenes histopatológicas de neoplasias melanocíticas spitzoides. En la actualidad, no existe ningún estudio en la literatura que haya empleado este tipo de técnicas a la clasificación de estas lesiones, por lo que no se puede comparar con ningún estudio del estado del arte.

6.2. Líneas futuras

Este trabajo se encuentra enmarcado dentro del proyecto europeo CLARIFY, como se comenta en la sección 2.1, por lo que las líneas futuras vienen definidas dentro del *road map* del proyecto. Dentro de este proyecto, en el cual se pretende desarrollar una plataforma de diagnóstico de imágenes WSI automatizada, basada en inteligencia artificial, el presente trabajo aporta un estudio sobre la mejora que ofrece la utilización de técnicas multiresolución basadas en *deep learning* sobre *Whole-Slide Images*.

Actualmente, el número de muestras empleado es bajo, ya que el proyecto CLARIFY se encuentra en una etapa temprana de desarrollo. Por lo que, para mejorar de la robustez de este tipo de algoritmos, será necesario disponer de un mayor número de muestras, que se irán consiguiendo a lo largo del proyecto.

Unas posibles líneas de trabajo futuro puede ser la posibilidad de explorar este tipo de técnicas multiresolución, ya no solo en muestras de neoplasias melanocíticas spitzoides, si no es muestras histológicas en general. También, la comparación de estos modelos con otros entrenados con la técnica *early fusion* y/o el desarrollo de estos modelos en otros lenguajes de programación, como puede ser Python.

Este trabajo, sirve como ayuda en la investigación y desarrollo de la plataforma de inteligencia artificial que se pretende desarrollar en el proyecto CLARIFY.

Bibliografía

- [1] Zoe Apalla y col. “Epidemiological trends in skin cancer”. En: *Dermatology Practical & Conceptual* 7.2 (30 de abr. de 2017), págs. 1-6. ISSN: 2160-9381. DOI: 10.5826/dpc.0702a01. URL: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5424654/> (visitado 31-07-2021).
- [2] *Cáncer de piel*. Asociación Española Contra el Cancer (AECC). URL: <https://www.aecc.es/es/todo-sobre-cancer/tipos-cancer/cancer-piel> (visitado 31-07-2021).
- [3] Lyndon D. Su y col. “Sentinel lymph node biopsy for patients with problematic spitzoid melanocytic lesions”. En: *Cancer* 97.2 (2003), págs. 499-507. ISSN: 1097-0142. DOI: 10.1002/cncr.11074. URL: <https://acsjournals.onlinelibrary.wiley.com/doi/abs/10.1002/cncr.11074> (visitado 01-08-2021).
- [4] Hongming Xu y col. “Automatic measurement of melanoma depth of invasion in skin histopathological images”. En: *Micron* 97 (1 de jun. de 2017), págs. 56-67. ISSN: 0968-4328. DOI: 10.1016/j.micron.2017.03.004. URL: <https://www.sciencedirect.com/science/article/pii/S096843281630381X> (visitado 31-07-2021).
- [5] Jiayun Li y col. “An attention-based multi-resolution model for prostate whole slide image classification and localization”. En: *arXiv:1905.13208 [cs, eess]* (30 de mayo de 2019). arXiv: 1905.13208. URL: <http://arxiv.org/abs/1905.13208> (visitado 08-03-2021).
- [6] Harshita Sharma y col. “Deep convolutional neural networks for automatic classification of gastric carcinoma using whole slide images in digital histopathology”. En: *Computerized Medical Imaging and Graphics* 61 (nov. de 2017), págs. 2-13. ISSN: 08956111. DOI: 10.1016/j.compmedimag.2017.06.001. URL: <https://linkinghub.elsevier.com/retrieve/pii/S0895611117300502> (visitado 04-03-2021).
- [7] Fabio Alexandre Spanhol y col. “Breast cancer histopathological image classification using Convolutional Neural Networks”. En: *2016 International Joint Conference on Neural Networks (IJCNN)*. 2016 International Joint Conference on Neural Networks (IJCNN). Vancouver, BC, Canada: IEEE, jul. de 2016, págs. 2560-2567. ISBN: 978-1-5090-0620-5. DOI: 10.1109/IJCNN.2016.7727519. URL: <http://ieeexplore.ieee.org/document/7727519/> (visitado 04-03-2021).
- [8] Mart van Rijthoven y col. “HookNet: Multi-resolution convolutional neural networks for semantic segmentation in histopathology whole-slide images”. En: *Medical Image Analysis* 68 (feb. de 2021), pág. 101890. ISSN: 13618415. DOI: 10.1016/j.media.2020.101890. URL: <https://linkinghub.elsevier.com/retrieve/pii/S1361841520302541> (visitado 04-03-2021).

- [9] *Clase de anatomía de la piel. Dr. Victor Alegre*. URL: <https://www.uv.es/derma/CLindex/CLdermatopat/CLdermatopatologia.html> (visitado 17-07-2021).
- [10] Boris C. Bastian. "THE MOLECULAR PATHOLOGY OF MELANOMA: AN INTEGRATED TAXONOMY OF MELANOCYTIC NEOPLASIA". En: *Annual review of pathology* 9 (2014), págs. 239-271. ISSN: 1553-4006. DOI: 10.1146/annurev-pathol-012513-104658. URL: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4831647/> (visitado 20-07-2021).
- [11] *Definición de epitelio - Diccionario de cáncer del NCI - Instituto Nacional del Cáncer*. Archive Location: nciglobal.ncicenterprise. 2 de feb. de 2011. URL: <https://www.cancer.gov/espanol/publicaciones/diccionarios/diccionario-cancer/def/epitelio> (visitado 20-07-2021).
- [12] M^o Rocio del Amor del Amor. "Diseño y desarrollo de sistemas automáticos de clasificación de tumores melanocíticos spitzoides". Trabajo Fin de Máster. València: Universitat Politècnica de València, 2019.
- [13] K. J. Smith y col. "Spindle cell and epithelioid cell nevi with atypia and metastasis (malignant Spitz nevus)". En: *The American Journal of Surgical Pathology* 13.11 (nov. de 1989), págs. 931-939. ISSN: 0147-5185. DOI: 10.1097/00000478-198911000-00003.
- [14] Pablo Vargas-Mora y col. "Neoplasias melanocíticas spitzoides: Estudio clínico-histopatológico de 62 lesiones en un hospital universitario chileno." En: *Revista Chilena de Dermatología* 35.4 (4 de mayo de 2020). ISSN: 0719-9406, 0717-2273. DOI: 10.31879/rcderm.v35i4.271. URL: <https://rcderm.org/index.php/rcderm/article/view/271> (visitado 04-03-2021).
- [15] David E. Elder y col. "The 2018 World Health Organization Classification of Cutaneous, Mucosal, and Uveal Melanoma: Detailed Analysis of 9 Distinct Subtypes Defined by Their Evolutionary Pathway". En: *Archives of Pathology & Laboratory Medicine* 144.4 (1 de abr. de 2020), págs. 500-522. ISSN: 0003-9985, 1543-2165. DOI: 10.5858/arpa.2019-0561-RA. URL: <http://meridian.allenpress.com/aplm/article/144/4/500/427497/The-2018-World-Health-Organization-Classification> (visitado 20-07-2021).
- [16] *Sample and slide preparation - Histology*. Bioalternatives. 16 de jul. de 2019. URL: <https://www.bioalternatives.com/en/sample-and-slide-preparation/> (visitado 22-07-2021).
- [17] *Spitz Nevus and Other Spitzoid Tumors in Children—Part 1: Clinical, Histopathologic, and Immunohistochemical Features | Elsevier Enhanced Reader*. DOI: 10.1016/j.adengl.2019.12.006. URL: <https://u.i3m.upv.es/d-05d> (visitado 13-06-2021).
- [18] Raymond L Barnhill. "The Spitzoid lesion: rethinking Spitz tumors, atypical variants, 'Spitzoid melanoma' and risk assessment". En: *Modern Pathology* 19 (S2 feb. de 2006), S21-S33. ISSN: 0893-3952, 1530-0285. DOI: 10.1038/modpathol.3800519. URL: <http://www.nature.com/articles/3800519> (visitado 04-03-2021).
- [19] A. Spatz y col. "Spitz tumors in children: a grading system for risk stratification". En: *Archives of Dermatology* 135.3 (mar. de 1999), págs. 282-285. ISSN: 0003-987X. DOI: 10.1001/archderm.135.3.282.

- [20] K. Isse y col. “Digital Transplantation Pathology: Combining Whole Slide Imaging, Multiplex Staining and Automated Image Analysis”. En: *American Journal of Transplantation* 12.1 (2012). _eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1600-6143.2011.03797.x>, págs. 27-37. ISSN: 1600-6143. DOI: 10.1111/j.1600-6143.2011.03797.x. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1600-6143.2011.03797.x> (visitado 03-08-2021).
- [21] Christel Daniel y col. “Standardizing the use of whole slide images in digital pathology”. En: *Computerized Medical Imaging and Graphics*. Whole Slide Image Process 35.7 (1 de oct. de 2011), págs. 496-505. ISSN: 0895-6111. DOI: 10.1016/j.compmedimag.2010.12.004. URL: <https://www.sciencedirect.com/science/article/pii/S0895611110001357> (visitado 03-08-2021).
- [22] *Whole Slide Imaging | MBF Bioscience*. URL: <https://www.mbfbioscience.com/whole-slide-imaging> (visitado 16-05-2021).
- [23] *Redes neuronales convolucionales*. Redes neuronales convolucionales - MATLAB. URL: <https://es.mathworks.com/discovery/convolutional-neural-network-matlab.html> (visitado 13-06-2021).
- [24] *Práctica de AA: Clasificación de imágenes*. Google Developers. URL: <https://developers.google.com/machine-learning/practica/image-classification/convolutional-neural-networks?hl=es> (visitado 17-07-2021).
- [25] *Rectified Linear Units (ReLU) in Deep Learning*. URL: <https://kaggle.com/dansbecker/rectified-linear-units-relu-in-deep-learning> (visitado 17-07-2021).
- [26] *CS231n Convolutional Neural Networks for Visual Recognition*. URL: <https://cs231n.github.io/convolutional-networks/> (visitado 17-07-2021).
- [27] *Convolutional Neural Networks (CNN): Step 4 - Full Connection - Blogs SuperDataScience - Machine Learning | AI | Data Science Career | Analytics | Success*. URL: <https://www.superdatascience.com/blogs/convolutional-neural-networks-cnn-step-4-full-connection> (visitado 17-07-2021).
- [28] Nicolas Diaz Salazar, Jesus Alfonso Lopez Sotelo y Gustavo Andres Salazar Gomez. “Application of Transfer Learning for Object Recognition Using Convolutional Neural Networks”. En: *2018 IEEE 1st Colombian Conference on Applications in Computational Intelligence (ColCACI)*. 2018 IEEE 1st Colombian Conference on Applications in Computational Intelligence (ColCACI). Mayo de 2018, págs. 1-6. DOI: 10.1109/ColCACI.2018.8484856.
- [29] Karen Simonyan y Andrew Zisserman. “Very Deep Convolutional Networks for Large-Scale Image Recognition”. En: *arXiv:1409.1556 [cs]* (10 de abr. de 2015). arXiv: 1409.1556. URL: <http://arxiv.org/abs/1409.1556> (visitado 04-08-2021).
- [30] Christian Szegedy y col. “Rethinking the Inception Architecture for Computer Vision”. En: *arXiv:1512.00567 [cs]* (11 de dic. de 2015). arXiv: 1512.00567. URL: <http://arxiv.org/abs/1512.00567> (visitado 04-08-2021).
- [31] Kaiming He y col. “Deep Residual Learning for Image Recognition”. En: *arXiv:1512.03385 [cs]* (10 de dic. de 2015). arXiv: 1512.03385. URL: <http://arxiv.org/abs/1512.03385> (visitado 04-08-2021).

- [32] Taimoor Shakeel Sheikh, Yonghee Lee y Migyung Cho. “Histopathological Classification of Breast Cancer Images Using a Multi-Scale Input and Multi-Feature Network”. En: *Cancers* 12.8 (24 de jul. de 2020), pág. 2031. ISSN: 2072-6694. DOI: 10.3390/cancers12082031. URL: <https://www.mdpi.com/2072-6694/12/8/2031> (visitado 08-03-2021).
- [33] *MATLAB - El lenguaje del cálculo técnico*. URL: <https://es.mathworks.com/products/matlab.html> (visitado 24-07-2021).
- [34] *Options for training deep learning neural network - MATLAB trainingOptions - MathWorks España*. URL: <https://es.mathworks.com/help/deeplearning/ref/trainingoptions.html> (visitado 25-07-2021).
- [35] *Build Networks with Deep Network Designer - MATLAB & Simulink - MathWorks España*. URL: <https://es.mathworks.com/help/deeplearning/ug/build-networks-with-deep-network-designer.html> (visitado 25-07-2021).
- [36] Francesco De Logu y col. “Recognition of Cutaneous Melanoma on Digitized Histopathological Slides via Artificial Intelligence Algorithm”. En: *Frontiers in Oncology* 0 (2020). Publisher: Frontiers. ISSN: 2234-943X. DOI: 10.3389/fonc.2020.01559. URL: <https://www.frontiersin.org/articles/10.3389/fonc.2020.01559/full> (visitado 25-07-2021).
- [37] Max Ferguson y col. *Automatic localization of casting defects with convolutional neural networks*. Pages: 1735. 1 de dic. de 2017. 1726 págs. DOI: 10.1109/BigData.2017.8258115.
- [38] Raúl García Jiménez, Marcos Escudero Viñolo y Jesús Bescós Cano. “Mejora del rendimiento de redes convoluciones entrenadas para el reconocimiento de escena mediante el uso de información sobre los objetos comunes a éstas”. En: (), pág. 90.
- [39] *Guía avanzada de Inception v3 para Cloud TPU | Cloud TPU*. Google Cloud. URL: <https://cloud.google.com/tpu/docs/inception-v3-advanced?hl=es-419> (visitado 25-07-2021).

Parte II

Presupuesto

Capítulo 1

Presupuesto

1.1. Objetivo

El objetivo de este capítulo es calcular el coste económico del trabajo fin de máster (TFM) "Diseño y desarrollo de un sistema automático para la detección de tumores melanocíticos spitzoides aplicando técnicas multi-resolución basadas en deep learning sobre Whole-Slide Images". En este cálculo se tendrán en cuenta tanto las horas trabajadas por el personal involucrado, como la amortización de los equipos y *software* empleado durante todo el periodo de desarrollo.

1.2. Presupuesto parcial

El presupuesto parcial involucra los costes de personal, el coste de *hardware* de los equipos y el *software* empleado. Este desglose de costes permite tener una visión amplia y específica de los distintos costes del proyecto, pudiendo optimizarlos para obtener un mayor rendimiento si fuese necesario.

1.2.1. Coste de personal

El coste de personal refleja el coste total de las horas trabajadas en este proyecto. Para ello, se han calculado las horas del siguiente personal involucrado:

- *D^a. Valery Naranjo Ornedo*: directora del Instituto de Investigación e Innovación en Bioingeniería (I3B) y tutora del presente proyecto.
- *D^o. M^o Rocio del Amor del Amor*: técnica superior de investigación en el Instituto de Investigación e Innovación en Bioingeniería (I3B) y cotutora del presente proyecto.
- *D. Francisco Javier Curieses Sanz*: alumno del Máster e Ingeniería de Telecomunicación y autor del presente proyecto.

En la Tabla 1.1 se muestra un desglose de la cantidad de horas dedicadas por cada una de las personas anteriormente mencionados, así como el coste total de esas horas.

| Personal | Unidades | Cantidad | Precio unitario (€/h) | Coste imputable (€) |
|--------------|----------|----------|-----------------------|---------------------|
| Tutora | h. | 15 | 42,00 | 630,00 |
| Cotura | h. | 40 | 17,20 | 688,00 |
| Autor | h. | 780 | 14,00 | 10.920,00 |
| Total | | | | 12.238,00 € |

Tabla 1.1: Costes de personal asociados al desarrollo del presente proyecto.

1.2.2. Coste *hardware*

Para el desarrollo del presente trabajo, se han utilizado principalmente tres equipos de computación y una unidad de almacenamiento: dos ordenadores de uso personal, un servidor de computación y un servidor NAS (estos dos últimos, propiedad de el grupo CVBLab).

Uno de los ordenadores de uso personal, identificado como el activo *hw_1*, se ha empleado para el desarrollo de todos los algoritmos de *deep learning*, búsqueda de información del estado del arte y elaboración de la memoria final de este trabajo. Mientras que el segundo ordenador personal, identificado como el activo *hw_2*, se ha utilizado para realizar las conexiones SSH (*Secure Shell*) con el servidor del CVBLab. En estas conexiones, se ejecutaba el programa MATLAB para realizar las simulaciones de los algoritmos desarrollados en *hw_1*.

El servidor de computación está formado por dos componentes principales, el procesador (identificado como el activo *hw_3*, y la tarjeta gráfica (identificada como el activo *hw_4*, ambos ejecutan y procesan los distintos entrenamientos de las redes neuronales desarrolladas. Por último, el servidor NAS, identificado como el activo *hw_5*, se ha utilizado como servidor de datos, almacenando todas las imágenes WSI utilizadas. Como se comentó anteriormente, estos equipos han sido proporcionados por el grupo CVBLab.

En la Tabla 1.2 se muestra el presupuesto total del coste de los equipos *hardware* empleados. Las descripciones y características técnicas de los equipos son:

- *hw_1*: Lenovo IdeaPad S540 Intel(R) Core(TM) i7-10510UCPU @ 1.80GHz 2.30GHz y 8 GB de memoria RAM.
- *hw_2*: HP pavilion 15-n014ss Intel(R) Core(TM) i7-4500UCPU @ 1.80GHz, 8 GB de memoria RAM y NVIDIA GeForce GT 740M (2GB RAM DDR3).
- *hw_4*: Procesador Intel(R) Core(TM) i7-7700K CPU @ 4.20 GHz.
- *hw_4*: Tarjeta gráfica NVIDIA GP102 TITAN Xp con 12GB de memoria RAM.
- *hw_5*: NAS Sygnology DS918.

1.2.3. Coste de *software*

Por último, se ha de imputar el coste del *software* utilizado para el desarrollo de este trabajo. Para desarrollar los algoritmos de acondicionamiento de datos y de *deep learning*, se ha hecho uso de la

| Activo | Cant. | Coste unitario (€) sin IVA | Periodo de amortización (meses) | Intervalo amortizado (meses) | Coste imputable (€) sin IVA |
|--------------|-------|----------------------------|---------------------------------|------------------------------|-----------------------------|
| hw_1 | 1 | 789,21 | 72 | 8 | 87,69 |
| hw_2 | 1 | 347,89 | 72 | 4 | 19,32 |
| hw_3 | 1 | 290,26 | 48 | 4 | 24,18 |
| hw_4 | 1 | 957,79 | 48 | 4 | 79,81 |
| hw_5 | 1 | 77,00 | 48 | 4 | 6,42 |
| Total | | | | | 247,42 € |

Tabla 1.2: Costes *hardware* asociados al desarrollo del presente proyecto.

herramienta MATLAB R2020b. Para trabajar en el desarrollo de este documento, se ha utilizado el editor de texto de LaTeX Overleaf y Zotero para organizar la bibliográfica empleada, ambos *software* de uso gratuito. Por último, se ha utilizado el paquete Microsoft 365 para anotar los resultados que se han ido obteniendo durante los entrenamientos y almacenar los *scripts* desarrollados en la nube. En la Tabla 1.3 se reflejan los costes totales de *software*.

| Descripción | Cantidad | Coste unitario (€) sin IVA | Periodo de amortización (meses) | Intervalo amortizado (meses) | Coste imputable (€) sin IVA |
|---------------|----------|----------------------------|---------------------------------|------------------------------|-----------------------------|
| MATLAB R2020b | 1 | 800,00 | 12 | 6 | 400,00 |
| Overleaf | 1 | 0 | 12 | 5 | 0 |
| Zotero | 1 | 0 | 48 | 5 | 0 |
| Microsoft 365 | 1 | 50,40 | 12 | 6 | 25,20 |
| Total | | | | | 425,20 € |

Tabla 1.3: Costes *software* asociados al desarrollo del presente proyecto.

1.3. Presupuesto total

El Presupuesto de Ejecución de Material (PEM), se calcula con la suma de los costes de personal, *hardware* y *software*. En la Tabla 1.4 se refleja el presupuesto total.

Para calcular el coste total real que supone el desarrollo de este trabajo fin de máster, se han de contemplar los costes generales asociados al proyecto, el beneficio industrial y el IVA (Impuesto sobre el Valor Añadido) correspondiente. Como gastos generales se contempla un 13 % del Presupuesto de Ejecución de Materiales. En concepto de beneficio industrial se considera un 6 % del PEM. Y por último, el IVA a fecha de la elaboración del presente trabajo, se sitúa en un 21 %. El coste total de desarrollar el proyecto se puede observar en la Tabla 1.5

| Descripción | Coste total (€) sin IVA |
|-----------------|-------------------------|
| Personal | 12.238,00 |
| <i>Hardware</i> | 247,42 |
| Software | 425,20 |
| Total | 12.910,62 € |

Tabla 1.4: Presupuesto de Ejecución de Material del proyecto.

| Descripción | Coste total (€) |
|--------------------------------------|--------------------|
| Presupuesto de Ejecución de Material | 12.910,62 |
| Gastos generales | 1.678,38 |
| Beneficio industrial | 774,64 |
| Subtotal | 15.363,64 |
| IVA (21 %) | 3.226,36 |
| Presupuesto total | 18.590,01 € |

Tabla 1.5: Presupuesto total del proyecto.

Por tanto, el coste total de la realización del presente proyecto es de **DIECIOCHO MIL QUINIENTOS NOVENTA EUROS CON UN CÉNTIMO**.