



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA



ESCUELA TÉCNICA  
SUPERIOR INGENIERÍA  
INDUSTRIAL VALENCIA

**TRABAJO FIN DE MÁSTER EN INGENIERÍA BIOMÉDICA**

# **DISEÑO Y DESARROLLO DE SOFTWARE PARA EL ANÁLISIS DE PATRONES MICROSCÓPICOS DE CRECIMIENTO TUMORAL EN PACIENTES CON CARCINOMA VESICAL**

**AUTOR:** IVANA DEL ROCÍO AGUILAR PORRAS

**TUTOR:** GREGORIO RUBIO NAVARRO

**COTUTOR:** JOSÉ GABRIEL GARCÍA PARDO

**Curso Académico: 2021-22**



# Agradecimientos

En primer lugar quiero agradecer a Gregorio y David, por haber depositado su confianza en mí para este trabajo, por la ayuda y el interés prestado.

A Gabriel, por la ayuda y la orientación prestada en la elaboración de este trabajo y por permitirme aprender de él; uno de los mejores.

A ellos, mi admiración y aprecio.

## ***A mi familia:***

A mi esposo: Francisco, por ser el mejor compañero, amigo y padre para mis hijos, por todo su amor, su apoyo, su confianza y por siempre creer en mí.

A mis hijos: Francisco y Álvaro, por ser los mejores profesores que la vida me ha dado y por ser mi principal motor para querer superarme día a día, en todos los aspectos.

A mi padre: Galo, por lo ánimos, por sus consejos y por recordarme siempre que: “la vida es una carrera de fondo”.

A mi hermana: Karen, por siempre apoyarme, por creer en mí, por ser una buena amiga, confidente y porque me siento súper orgullosa de la mujer en la que se ha convertido.

A mi suegra: Carmen, por todo su apoyo, su ánimo y cariño

A ellos, mi amor y mi infinito agradecimiento, por todo el tiempo con la tesis y no con ellos.

## ***A mis amigas:***

A Anna Esteve Domínguez, por creer en mí y recomendarme para este proyecto, por ser la principal incitadora a que persiga mi pasión con la imagen médica, por ser una buena compañera y amiga.

A Ilse, Zenaida, Cristina, Xusi, Eva y Rocío, por lo ánimos, por vuestra paciencia y por aceptarme como soy.

A ellas, todo mi cariño, respeto y confianza.

## ***Y a título póstumo:***

Mi tío Guido, por inculcarme desde muy pequeña la pasión por la lectura, los buenos modales y los valores, que me hicieron ser, la persona que soy.



# Resumen

El cáncer de vejiga es una de las clases más frecuentes de cáncer en el mundo y afecta principalmente a la población masculina adulta, en edades comprendidas entre los 60-65 años.

Las variantes de carcinoma vesical se engloban en 2 tipos: el músculo no invasivo o también conocido por sus siglas en inglés NMIBC (*Non Muscle Invasive Bladder Cancer*) y el músculo invasivo o en inglés (*Muscle Invasive Bladder Cancer*), siendo este último el más agresivo de los dos, ya que su progreso tiende a propagarse hacia el exterior del órgano afectado (en este caso, la vejiga) comprometiendo de esta manera a tejidos y órganos vitales. Este proceso es conocido como metástasis de la enfermedad.

Es por ello, que la detección temprana de patrones asociados a esta patología es de vital importancia ya que pueden ayudar a detectar su estadio y su malignidad.

Según estudios recientes, el tumor budding (TB) o también conocido como patrón infiltrativo, es un factor predictor de progresión del cáncer de vejiga o *carcinoma urotelial*.

Con el avance de la inteligencia artificial (AI) en sistemas de aprendizaje profundo, las redes neuronales convolucionales han demostrado ser un poderoso algoritmo para poder avanzar en el campo del análisis de las imágenes histopatológicas médicas para la detección de carcinomas<sup>[33] [51]</sup> frente a los sistemas de detección tradicionales.

En el presente Trabajo Fin de Máster (TFM) se propone implementar y desarrollar algoritmos para la detección de patrón budding sobre imágenes de muestras histopatológicas de carcinoma vesical músculo infiltrantes a través de técnicas de Deep learning; como la transferencia de conocimiento (*transfer learning*) o el modelado de una red neuronal convolucional desde cero (*from scratch*).

Los resultados obtenidos en este TFM nos invitan a seguir esta línea de investigación para seguir mejorando el modelado de la red, la obtención de más imágenes y contribuir de esta manera a la creación de un software de ayuda al diagnóstico computarizado (CAD).

**Palabras clave:** software, soporte lógico, aprendizaje profundo, algoritmos, carcinoma urotelial, cáncer de vejiga, tumor budding, reconocimiento de patrones, redes neuronales, redes neuronales convolucionales, visión por computador, imágenes médicas, imágenes histopatológicas, inteligencia artificial.



# Resum

El càncer de bufeta és una de les classes més freqüents de càncer en el món i afecta principalment a la població masculina adulta, en edats compreses entre els 60-65 anys.

Les variants de carcinoma vesical s'engloben en 2 tipus: el múscul no invasiu o també conegut per les seues sigles en anglés NMIBC (*Non Muscle Invasive Bladder Cancer*) i el múscul invasiu o en anglés (*Muscle Invasive Bladder Cancer*), sent aquest últim el més agressiu dels dos, ja que el seu progrés tendeix a propagar-se cap a l'exterior de l'òrgan afectat (en aquest cas, la bufeta) compromentent d'aquesta manera a teixits i òrgans vitals. Aquest procés és conegut com a metàstasi de la malaltia.

És per això, que la detecció precoç de patrons associats a aquesta patologia és de vital importància ja que poden ajudar a detectar el seu estadi i la seua malignitat.

Segons estudis recents, el *tumor budding* (TB) o també conegut com a patró infiltratiu, és un factor predictor de progressió del càncer de bufeta o carcinoma urotelial.

Amb l'avanç de la intel·ligència artificial (AI) en sistemes d'aprenentatge profund, les xarxes neuronals convolucionals han demostrat ser un poderós algorisme per a poder avançar en el camp de l'anàlisi de les imatges histopatològiques metgesses per a la detecció de carcinomes<sup>[33] [51]</sup> enfront dels sistemes de detecció tradicionals.

En el present Treball Fi de Màster (TFM) es proposa implementar i desenvolupar algorismes per a la detecció de patró budding sobre imatges de mostres histopatològiques de carcinoma vesical múscul infiltrantiu a través de tècniques de Deep learning; com la transferència de coneixement (*transfer learning*) o el modelatge d'una xarxa neuronal convolucional des de zero (*from scratch*).

Els resultats obtinguts en aquest TFM ens conviden a seguir aquesta línia d'investigació per a continuar millorant el modelatge de la xarxa, l'obtenció de més imatges i contribuir d'aquesta manera a la creació d'un programari d'ajuda al diagnòstic informatitzat (CAD).

Paraules clau: programari, suport lògic, aprenentatge profund, algorismes, carcinoma \*urotelial, càncer de bufeta, tumor budding, reconeixement de patrons, xarxes neuronals, xarxes neuronals convolucionals, visió per computador, imatges mèdiques, imatges histopatològiques, intel·ligència artificial.





## Abstract

Bladder cancer is one of the most frequent types of cancer in the world and mainly affects the adult male population, aged between 60-65 years.

The variants of bladder carcinoma are included in 2 types: the non-invasive muscle or also known by its acronym in English NMIBC (*Non Muscle Invasive Bladder Cancer*) and the invasive muscle or in English (*Muscle Invasive Bladder Cancer*), the latter being the most aggressive of the two, since its progress tends to spread to the outside of the affected organ (in this case, the bladder), thus compromising vital tissues and organs. This process is known as metastatic disease.

For this reason, the early detection of patterns associated with this pathology is of vital importance since they can help detect its stage and malignancy.

According to recent studies, the budding tumor (TB) or also known as infiltrative pattern, is a predictor of progression of bladder cancer or urothelial carcinoma.

With the advancement of artificial intelligence (AI) in deep learning systems, convolutional neural networks have proven to be a powerful algorithm to advance the field of medical histopathological image analysis for carcinoma detection<sup>[33] [51]</sup> compared to traditional detection systems.

In this Master's Thesis (TFM) it is proposed to implement and develop algorithms for the detection of budding patterns on images of histopathological samples of muscle-infiltrating bladder carcinoma through Deep learning techniques; such as transfer learning or modeling a convolutional neural network from scratch.

The results obtained in this TFM invite us to continue this line of research to continue improving the modeling of the network, obtaining more images and thus contributing to the creation of computerized diagnostic aid software (CAD).

**Keywords:** software, software, deep learning, algorithms, urothelial carcinoma, bladder cancer, tumor budding, pattern recognition, neural networks, convolutional neural networks, computer vision, medical imaging, histopathological imaging, artificial intelligence.



## Listado de abreviaturas

TB	TUMOR BUDDING
DP	DEEP LEARNING
OMS	ORGANIZACIÓN MUNDIAL DE LA SALUD
FR	FACTOR DE RIESGO
GCO	GLOBAL CÁNCER OBSERVATORY
CVBLab	COMPUTER VISION AND BEHAVIOUR LABORATORY
FM	FIBRAS MUSCULARES
NMIBC	NON-MUSCLE-INVASIVE CÁNCER
MIBC	MUSCLE-INVASIVE CÁNCER
TURBT	TRANS-URETRAL RESECTION BLADDER TUMOR
CIS	CARCINOMA IN SITU
FISH	HIBRIDACIÓN FLUORESCENTE IN SITU
ELIZA	ANÁLISIS DE INMUNO-ABSORCIÓN LIGADO A ENZIMAS
TMN	SISTEMA TUMOR-METÁSTASIS-NODO
AP	APRENDIZAJE PROFUNDO
WSI	WHOLE SLIDE IMAGE
CC	CANCER COLORRECTAL
CNN	CONVOLUTIONAL NEURAL NETWORK
NN	NEURAL NETWORK



# Índice general

## Índice de contenido

<b>MEMORIA .....</b>	<b>1</b>
<b>INTRODUCCIÓN .....</b>	<b>3</b>
1.1 Motivación y descripción del problema .....	4
1.2 Objetivos del proyecto .....	5
1.3 Estructura del documento.....	6
<b>MARCO TEÓRICO .....</b>	<b>7</b>
2.1 Cáncer: Definición, factor de riesgo (FR), prevención y tratamiento.....	8
2.2 Cáncer de Vejiga: Inicio, crecimiento y propagación, incidencia y mortalidad.....	12
2.3 Proceso diagnóstico, estadio de la enfermedad, imagen histológica y patrones nodulares más comunes.....	18
<b>MATERIALES .....</b>	<b>26</b>
3.1 Base de datos .....	27
3.2 Software utilizado .....	27
3.3 Hardware utilizado .....	30
<b>METODOLOGÍA .....</b>	<b>31</b>
4.1 Preparación de los datos: Etiquetado, generación de patches y partición de datos.....	33
4.2 Algoritmos implementados.....	39
4.3 Algoritmos de clasificación supervisada para las imágenes histopatológicas de los grados del carcinoma urotelial .....	44
<b>EXPERIMENTACIÓN Y RESULTADOS .....</b>	<b>51</b>
5.1 Experimentos realizados .....	52
5.2 Métricas de evaluación .....	54
5.2.1 Análisis cuantitativo: .....	54
5.3 Resultados obtenidos.....	56
5.3.1 Top model: ‘GMP’ transfer Learning.....	56
5.3.2 Modelo From scratch IAP - Top model: IAP: .....	60
<b>CONCLUSIONES Y LÍNEAS FUTURAS .....</b>	<b>63</b>
6.1 Conclusiones .....	64
6.2 Líneas futuras .....	64
<b>BIBLIOGRAFÍA .....</b>	<b>65</b>
<b>PRESUPUESTOS .....</b>	<b>69</b>
<b>1.1 OBJETIVO.....</b>	<b>1</b>
<b>1.2 PRESUPUESTOS PARCIALES.....</b>	<b>1</b>
i) Costes de personal .....	1
ii) Coste de materiales.....	1
iii) Coste de software .....	2
iv) Coste de hardware .....	2
<b>1.3 PRESUPUESTO TOTAL.....</b>	<b>3</b>

# Índice de figuras

## Índice de contenido

Figura 2.1: Número de nuevos casos de cáncer en 2020; Hombres (todas las edades). Fuente [2].....	9
Figura 2.2: Número de nuevos casos de cáncer en 2020; Mujeres (todas las edades). Fuente [2].....	10
Figura 2.3: Capas de la pared vesical. Imagen obtenida del blog “Understand to learn”, fuente [5].....	12
Figura 2.4: Crecimiento tumoral vesical (músculo no invasivo y músculo invasivo). Imagen obtenida del artículo de Oner S et al. (2017), “Bladder cancer”. Fuente [8]. .....	13
Figura 2.5: Edad estándar (mundial) para el cálculo de la ratio de la incidencia del cáncer de vejiga. Fuente [2]. .....	16
Figura 2.6: Fijación, lavado y deshidratación de la muestra. Figura modificada de referencia [16].....	22
Figura 2.7: Inclusión de una muestra histológica. Imagen modificada de referencia [17]. .....	23
Figura 2.8: Corte de las muestras utilizando el microtomo. Imagen modificada de referencia [17].....	23
Figura 2.9: Procedimiento de tinción de una muestra histopatológica. Figura modificada de referencia [17]. .....	24
Figura 2.10: Patrones morfológicos; nodular (I), trabecular(II) e infiltrativo(III). Imagen obtenida de referencia [18]. .....	24
Figura 2.11: Patrón nodular en imagen histológica con MIBC. Elaboración propia.....	24
Figura 2.12: Patrón trabecular en imagen histológica con MIBC. Elaboración propia.....	25
Figura 2.13: Patrón trabecular en imagen histológica con MIBC. Elaboración propia.....	25
Figura 3.1: (a) soporte para los portaobjetos analógicos, (b) Interfaz del software Aperio ImageScope y (c) Escáner Aperio CS2. Imagen modificada de la referencia [46]. .....	27
Figura 3.2: Captura de pantalla de la herramienta de etiquetado Microdraw que pertenece al grupo de investigación CVB Lab de la UPV.....	28
Figura 3.3: Clasificación de los grados de TB según su densidad encontrada en la WSI. Las imágenes se corresponden a parches de tamaño 512x512x3 píxeles. Elaboración propia. ....	28
Figura 3.4: Logotipos de las herramientas software utilizadas; Docker, Matlab, Python y MobaXterm.....	29
Figura 4.1: Diagrama de flujo correspondiente a la metodología empleada en este trabajo..	32
Figura 4.2: Sub-división de la WSI. Parches obtenidos: a) Fondo, b) Tejido sano, c) Tejido tumoral no etiquetado, d) Tejido tumoral en grado bajo, e) Tejido tumoral en grado medio y f) Tejido tumoral en grado alto. Elaboración propia. ....	33
Figura 4.3: Selección de parches obtenidos de la WSI. Elaboración Propia.....	34
Figura 4.4: Distribución de clases del Ground truth. Elaboración propia.....	34

Figura 4.5: Ejemplo de submuestreo o under-sampling y sobremuestreo u over-sampling. Imagen modificada de referencia [35].	35
Figura 4.6: Captura de pantalla de los pesos proporcionados por la función <code>Class_weight_compute_class_weight</code> de Skicit-learn, en la terminal MovaXterm. Ref [37]. Elaboración propia.	36
Figura 4.7: Diferencias entre subajuste (underfitting) y sobreajuste (overfitting). Referencia [47].	36
Figura 4.8: Transformación de un patch obtenido de una WSI (perteneciente a la clase grado medio) con la técnica Data augmentation de tensorflow[36]. Elaboración propia.	37
Figura 4.9: Transformación de otros patches de la base de datos aplicando la técnica Data augmentation de tensorflow[36]. Elaboración propia.	37
Figura 4.10: Funcionamiento de la técnica K-fold cross validation empleada en este estudio. Elaboración propia.	38
Figura 4.11: Partición de datos realizada con la técnica K-fold cross validation.	38
Figura 4.12: Estructura de una Red neuronal. Imagen obtenida de referencia [44].	39
Figura 4.13: Comportamiento del learning rate o tasa de aprendizaje en función del número de épocas.	40
Figura 4.14: Comportamiento del hiperparámetro momentum. Imagen obtenida de referencia [39].	41
Figura 4.15: Gráfica y ecuación de la función de activación Relu. Imagen obtenida de la referencia [39].	41
Figura 4.16: Comparativa de los optimizadores Adagrad y el descenso del gradiente: El primero puede corregir su dirección y llegar rápidamente al punto óptimo. Imagen obtenida de referencia [24].	43
Figura 4.17: Estructura de una CNN. Elaboración propia.	44
Figura 4.18: Capas convolucionales con múltiples mapas de características en una imagen con 3 canales de colores (RGB). Imagen modificada de la referencia [24].	45
Figura 4.19: Representación de una capa de Max pooling con 2x2 de pooling kernel, zancada, no padding. Figura modificada de referencia[24].	46
Figura 4.20: Estructura de una capa fully connected o capa densamente conectada. Imagen obtenida de la referencia [39].	46
Figura 4.21: Funcionamiento de la capa MaxGlobalpooling2D. Imagen obtenida de la referencia [50].	47
Figura 4.22: Comportamiento de capas densas sin y con dropout. Imagen obtenida de la referencia [39].	47
Figura 4.23: Arquitectura VGG16 diseñada para llevar a cabo tareas de clasificación. Figura modificada de referencia [39].	48
Figura 4.24: Configuración de una CNN modelo VGG-16. Referencia [42].	49
Figura 4.25: Formas de Fine tuning, según la capa que se modifique. Elaboración propia.	50
Figura 5.1: Representación gráfica e interpretación de la matriz de confusión. Elaboración propia.	54

Figura 5.2: Gráfica de la curva ROC. Elaboración propia.....	55
Figura 5.3: Representación de las curvas de aprendizaje de la exactitud durante el entrenamiento para el modelo VGG16 utilizando transfer learning con la 5 capa convolucional reentrenada y top model 'GMP'. Valor de la exactitud (Acc) en cada época: (a) Kfold1, (b) Kfold2, (c) Kfold3, (d) Kfold4 y (e) Kfold5. ....	56
Figura 5.4: Representación de las curvas de aprendizaje de las pérdidas durante el entrenamiento para el modelo VGG16 utilizando transfer learning con la 5 capa convolucional reentrenada y top model 'GMP'. Valor de la exactitud (Acc) en cada época: (a) Kfold1, (b) Kfold2, (c) Kfold3, (d) Kfold4 y (e) Kfold5. ....	56
Figura 5.5: Matrices de confusión obtenidas de la etapa de validación para el modelo VGG16 utilizando transfer learning con la 5 capa convolucional reentrenada y top model 'GMP': (a) Kfold1, (b) Kfold2, (c) Kfold3, (d) Kfold4 y (e) Kfold5.....	58
Figura 5.6: Matriz de confusión obtenida de la etapa de predicción para el modelo VGG16 utilizando transfer learning con la 5 capa convolucional reentrenada y top model 'GMP'.....	59
Figura 5.7: Representación de las curvas de aprendizaje durante el entrenamiento para el modelo From scratch. Valor de la exactitud (Acc) en cada época: (a) Kfold1, (b) Kfold2, (c) Kfold3, (d) Kfold4 y (e) Kfold5.....	60
Figura 5.8: Representación de las curvas de aprendizaje durante el entrenamiento para el modelo From scratch. Valor de las pérdidas en cada época: (a) Kfold1, (b) Kfold2, (c) Kfold3, (d) Kfold4 y (e) Kfold5. ....	60
Figura 5.9: Matrices de confusión obtenidas de la etapa de validación del modelo From scratch: (a) Kfold1, (b) Kfold2, (c) Kfold3, (d) Kfold4 y (e) Kfold5. ....	61
Figura 5.10: Matriz de confusión obtenida de la etapa de predicción del modelo From scratch. ....	62



# Índice de tablas

## Índice de contenido

Tabla 2.1: Número de nuevos casos de cáncer en el mundo (ambos sexos, sobre una muestra de 19.292.789 personas). Datos obtenidos del Globocan. Tabla modificada [2].	9
Tabla 2.2: Nuevos casos y mortalidad de los 12 tipos de cáncer. Datos obtenidos de la Global Cancer Observatory (GCO), año 2020 [2]. Tabla modificada.	10
Tabla 2.3: Prevalencia (5 años) de los 12 tipos de cáncer más frecuentes alrededor del mundo. Datos obtenidos del GCO [2]. Tabla modificada.	11
Tabla 2.4: Estimaciones de incidencia, mortalidad y prevalencia a cinco años. España 2018. Mujeres. Fuente [3]: Tabla modificada.	11
Tabla 2.5: Estimaciones de incidencia, mortalidad y prevalencia a cinco años. España 2018. Hombres. Fuente [3]: Tabla modificada.	12
Tabla 2.6: Incidencia del cáncer de vejiga alrededor del mundo. Fuente Globocan 2020. Modificada [12]	15
Tabla 2.7: Índice de mortalidad del cáncer de vejiga alrededor del mundo. Fuente [12]	17
Tabla 2.8: Tipos de tumor. Información obtenida de referencia [11].	20
Tabla 2.9: Tipos de nodos información obtenida de referencia [11].	20
Tabla 2.10: Tipos de metástasis. Información obtenida de referencia [11].	21
Tabla 2.11: Agrupación por etapas o estadios del cáncer de vejiga. Información obtenida de referencia [11].	22
Tabla 4.12: Número de parches obtenidos de las WSI.	34
Tabla 5.1: Métricas cuantitativas obtenidas de la etapa de validación para el modelo VGG16 utilizando transfer learning con la 5 capa convolucional reentrenada y top model 'GMP'.	58
Tabla 5.2: Métricas cuantitativas obtenidas de la etapa de predicción.	59
Tabla 5.3: Métricas cualitativas obtenidas de la etapa de validación.	62
Tabla 5.4: Métricas cualitativas obtenidas de la etapa de predicción.	62

**Parte 1**

# **Memoria**



# Capítulo 1

## Introducción

### ÍNDICE DE CONTENIDO:

1.1	Motivación y descripción del problema .....	4
1.2	Objetivos del proyecto .....	5
1.3	Estructura del documento.....	6

## 1.1 Motivación y descripción del problema

Como se describió en el resumen de esta memoria, el cáncer de vejiga es una de las enfermedades más frecuentes en el mundo, ocupando el décimo primer puesto a nivel global, y el quinto en España, según los datos recogidos por el *Global Cancer Conservatory* (GCO), durante el año 2020.

Los patrones microscópicos tumorales más comunes que nos podemos encontrar en una muestra histopatológica de carcinoma urotelial músculo-infiltrante son 3: el patrón nodular, el trabecular y el infiltrativo o tumor budding, siendo este último el de peor pronóstico. Todos estos patrones se describirán con más detalle en el punto 2.3.3 del capítulo 2 de este documento académico.

El sistema tradicional para la observación, evaluación y diagnóstico de este tipo de patrones es mediante el uso de un microscopio electrónico de alta definición. Este instrumento está bajo la supervisión y control de un especialista experto en muestras histopatológicas: el patólogo.

El volumen de muestras que un patólogo puede analizar a nivel visual durante una jornada laboral de 8 horas es indeterminado, teniendo en más de un caso, volver a repasar la muestra debido al desgaste mental y agotamiento visual que supone el identificar y a su vez gradar el tipo de patrón que está visualizando, para así poder emitir un diagnóstico correcto.

Aprovechando el auge de las nuevas tecnologías, la evolución de los sistemas visuales computarizados y el emergente mundo de la inteligencia artificial, con este trabajo final de máster, se pretende aportar una herramienta de soporte diagnóstico basada en algoritmos de deep learning que tenga un valor (no decisivo a nivel diagnóstico, pero si a considerar) para que el patólogo pueda resolver aquellos casos en los que, este tipo de patrón no estén claramente diferenciado.

Es conveniente aclarar que, con la elaboración de estos algoritmos, no se pretende, en ningún caso, reemplazar la figura del profesional experto en este campo. No obstante, si lo es la de mejorar y agilizar el proceso diagnóstico, así como también la de mejorar la calidad de vida laboral, en la medida de lo posible, al profesional experto.

## 1.2 Objetivos del proyecto

El objetivo final de este trabajo fin de máster (TFM) es el diseño y desarrollo de algoritmos basados en inteligencia artificial para la identificación de los distintos grados existentes en células tumorales de carcinoma vesical músculo infiltrante y de esta manera poder ofrecer una herramienta de ayuda al diagnóstico computarizada al patólogo especialista.

Con la finalidad de alcanzar la consecución del objetivo principal de este TFM, se han abordado una serie de objetivos específicos, los mismos que se describen a continuación:

1. Revisar el estado del arte sobre diseños de algoritmos que aborden el problema de identificar este tipo de patrón (tumor budding) en células de carcinoma vesical músculo infiltrante en muestras de imágenes histopatológicas.
2. Estudiar, conocer e identificar los distintos grados que se pueden encontrar sobre las muestras histopatológicas del tipo de carcinoma antes mencionado.
3. Realizar el acondicionamiento de la base de datos proporcionada para poder generar los patches o tiles de imágenes que servirán como datos de entrada para entrenar y validar el modelo de red diseñado.
4. Realizar la partición de los datos utilizando técnicas de validación cruzada, teniendo en cuenta el número total de las imágenes.
5. Implementar los algoritmos desarrollados para la clasificación supervisada de los patrones, objeto de nuestro estudio.
6. Entrenar, validar y testear nuestro modelo desarrollado.
7. Presentar las métricas de evaluación obtenidas de los modelos de clasificación realizados.
8. Identificar y describir los problemas que se encuentren en cualesquiera de las fases de la realización de este proyecto y proponer posibles mejoras que puedan resultar útiles en futuras investigaciones.

### 1.3 Estructura del documento

Este proyecto se ha dividido en 2 partes. La primera parte se corresponde a la Memoria y la segunda parte al presupuesto o coste económico que ha supuesto poner en marcha este trabajo final de máster.

La Memoria está formada por 7 capítulos, de los cuales, el primero de ellos se ha expuesto en los puntos anteriores, el resto de los capítulos resumen lo siguiente:

En el capítulo 2, se presenta un marco teórico sobre el cáncer y sus tipos, así como su incidencia a nivel mundial y local. Nos centramos en el cáncer de vejiga, en concreto en el tipo de cáncer músculo-invasivo (dado que sus patrones predictores son nuestro objeto de estudio), así como también en su proceso de diagnóstico y estadios de la enfermedad.

En el capítulo 3, se describen los materiales que han sido necesarios para el desarrollo de los algoritmos: tanto a nivel de software como a nivel hardware.

En el capítulo 4, se describe la metodología empleada para abordar el diseño de la red neuronal, también se introducen breves conceptos teóricos de aprendizaje profundo o *deep learning*, sin la intención de profundizar en ellos (ya que no es el objetivo principal de este trabajo), así como la justificación del por qué en este proyecto se utilizan.

En el capítulo 5, se describe el procedimiento que se siguió, el valor de los parámetros utilizados y los resultados obtenidos de la experimentación realizada. Se exponen las métricas obtenidas, así como también se analizan estos resultados de forma cuantitativa y cualitativa.

Por último, en los capítulos 6 y 7, se exponen las conclusiones a las que se han llegado a partir de los resultados obtenidos, así como también las líneas futuras, es decir, las recomendaciones a seguir si se quiere continuar trabajando en esta área de estudio y la bibliografía consultada para la realización de este TFM.

En la parte 2 de este trabajo final de máster, que se corresponde a la del presupuesto (como se anticipada en las primeras líneas de este apartado), se describe el coste económico que ha supuesto poner en marcha este TFM. Este presupuesto está desglosado en 2 partes: el presupuesto parcial y el presupuesto total.

# Marco teórico

### ÍNDICE DE CONTENIDO:

<b>2.1 CÁNCER: DEFINICIÓN, FACTOR DE RIESGO (FR), PREVENCIÓN Y TRATAMIENTO.</b>	<b>8</b>
2.1.1. Definición .....	8
2.1.2. Factor de riesgo (FR) .....	8
2.1.3. Prevención.....	8
2.1.4. Tratamiento.....	8
<b>2.2 TIPOS DE CÁNCER: 12 TIPOS DE CÁNCER MÁS COMUNES, INCIDENCIA Y MORTALIDAD.</b>	<b>9</b>
2.1.5. Tipos de cáncer .....	9
<b>2.3 CÁNCER DE VEJIGA: INICIO, CRECIMIENTO Y PROPAGACIÓN, INCIDENCIA Y MORTALIDAD.</b>	<b>12</b>
2.1.6. Cáncer de vejiga .....	12
2.1.7. Inicio, crecimiento y propagación .....	12
2.1.8. Crecimiento y propagación .....	13
2.1.9. Incidencia .....	14
2.1.10. Mortalidad.....	16
<b>2.4 PROCESO DIAGNÓSTICO, ESTADIO DE LA ENFERMEDAD, IMAGEN HISTOLÓGICA Y PATRONES NODULARES MÁS COMUNES.</b>	<b>18</b>
2.1.11. Proceso diagnóstico: Exploración física, citoscopia, citología urinaria y marcadores urinarios.	18
2.1.11.1. Exploración física .....	18
2.1.12. Cistoscopia .....	18
2.1.12.1.1. Biopsia .....	18
2.1.12.2. Citología urinaria .....	19
2.1.12.3. Marcadores urinarios .....	19
2.1.13. Estadio de la enfermedad .....	19
2.1.14. Imagen histológica. ....	22
2.1.14.1. Fijación. ....	22
2.1.14.2. Inclusión. ....	22
2.1.14.3. Microtomo. ....	23
2.1.14.4. Montaje y tinción. ....	23
2.1.15. Patrones morfológicos. ....	24
2.1.16. El “tumor Budding” (TB) y el cáncer vesical .....	25



## 2.1 Cáncer: Definición, factor de riesgo (FR), prevención y tratamiento.

### 2.1.1. Definición

El cáncer, según la Organización Mundial de la Salud (OMS) es un término genérico que se designa a un amplio conjunto de enfermedades que pueden afectar a cualquier parte del organismo. Su principal característica es la multiplicación rápida e incontrolada de células anormales que se extienden más allá de sus límites habituales, pudiendo invadir partes adyacentes del cuerpo y propagarse a otros órganos o tejidos, a este proceso se le conoce como <<metástasis>> [1].

#### Factor de riesgo (FR)

Los factores de riesgos asociados a esta enfermedad son principalmente: el consumo no moderado de sustancias (como el tabaco y el alcohol), la exposición a agentes carcinógenos físicos (como las radiaciones ultravioletas *UV* e ionizantes), una dieta poco saludable (factores epigenómicos), la inactividad física y algunas infecciones crónicas (especialmente causados por la *Helicobacter pylori*, los papilomavirus humanos, los virus de la hepatitis B y de la hepatitis C y el virus de Epstein-Barr), que interactúan con los factores genéticos que predisponen al paciente. A este último factor de riesgo, en 2018 se le atribuyó el 13% de los casos de cáncer diagnosticados alrededor del mundo [1].

### 2.1.2. Prevención

Para reducir la carga de morbilidad por cáncer, es necesario, reducir los factores de riesgo, así como el de aplicar estrategias preventivas que tengan base científica.

Entre las estrategias que se emplean para su prevención, la OMS recomienda:

La detección temprana: ya que consigue que el tratamiento sea eficaz y más económico, además de que con esto se consigue que la probabilidad de supervivencia aumente, así como la calidad de vida de la persona que lo padece y;

Los programas de cribado: que se realizan a la población en función del riesgo y la edad [1].

En España, estas estrategias de prevención se realizan mediante: Prevención primaria y prevención secundaria [3]

### 2.1.3. Tratamiento

El tratamiento para el cáncer dependerá de: del tipo, la fase o estadio en el que se encuentre y de su localización. De forma generalizada, cuando el cáncer se encuentra en etapas tempranas su tratamiento consiste en radioterapia, quimioterapia y/o cirugía. En etapas más avanzadas o cuando la enfermedad ya no se puede tratar lo que se realiza en estos casos; son tratamientos paliativos cuyo objetivo principal es prolongar la vida e intentar mejorar la calidad de vida del paciente que lo padece [1].

Es por esto, que se necesita proveer al especialista de más herramientas para un diagnóstico temprano ya que le permitirá a su vez, seleccionar un tratamiento adecuado y personalizado al paciente.

Tipos de cáncer: 12 tipos de cáncer más comunes, incidencia y mortalidad.

#### 2.1.4. Tipos de cáncer

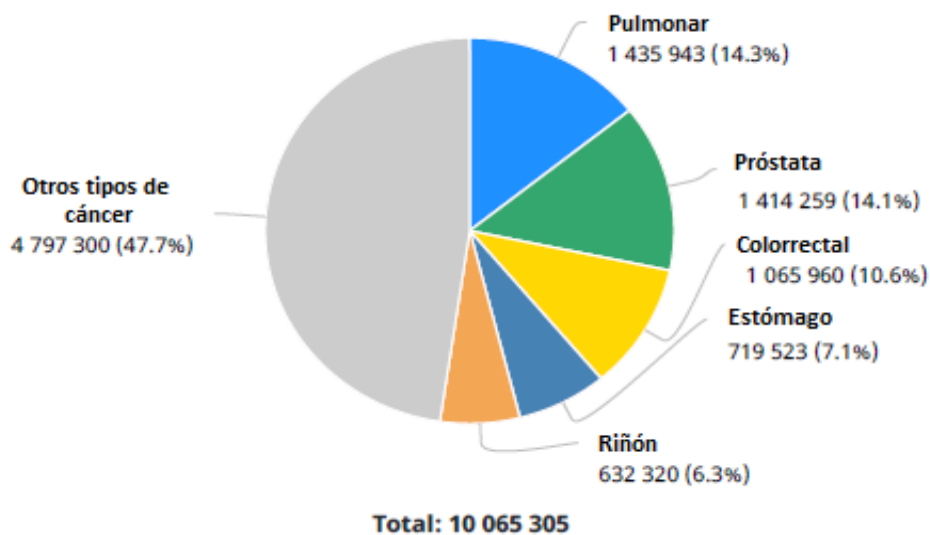
El cáncer es una de las principales causas morbilidad y mortalidad en el mundo, con casi 10 millones de fallecimientos en 2020 [2].

Durante ese año, los más comunes fueron:

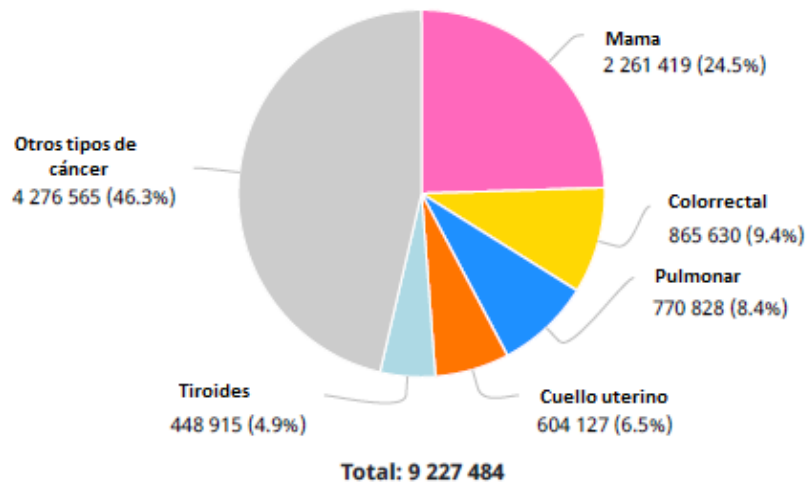
Tipo de cáncer	Número de casos (millones)	Porcentaje (%)
De mama	2,26	11,7
Pulmonar	2,21	11,4
Colorrectal	1,93	10
Próstata	1,41	7,3
De Piel (no melanoma)	1,2	6,2
Gástrico	1,09	5,6
Otros tipos de cáncer	10,39	47,8

**Tabla 2.1:** Número de nuevos casos de cáncer en el mundo (ambos sexos, sobre una muestra de 19.292.789 personas). Datos obtenidos del Globocan. Tabla modificada [2].

En las siguientes gráficas se muestra el porcentaje correspondiente al número de nuevos casos de cáncer: en hombres Figura 2.1 y mujeres Figura 2.2, en el rango de todas las edades.



**Figura 2.1:** Número de nuevos casos de cáncer en 2020; Hombres (todas las edades). Fuente [2].



**Figura 2.2:** Número de nuevos casos de cáncer en 2020; Mujeres (todas las edades). Fuente [2].

A continuación, en la siguiente Tabla 2.2 se muestra un resumen estadístico de la Incidencia y Mortalidad de los 12 tipos de cáncer más frecuentes alrededor del mundo.

Tipo de cáncer	Nuevos casos				Muertes			
	Número	Posición	(%)	Comunidad en riesgo (habitantes)	Número	Posición	(%)	Comunidad en riesgo (habitantes)
Mama	2 261 419	1	11.7	520	684 996	4	6.9	149
Pulmón	2 206 771	2	11.4	274	1 796 144	1	18.0	218
Próstata	1 414 259	3	7.3	386	375 304	8	3.8	63
Colon	1 148 515	4	6.0	130	576 858	5	5.8	55
Estómago	1 089 103	5	5.6	131	768 793	3	7.7	90
Riñón	905 677	6	4.7	111	830 180	2	8.3	101
Rectal	732 210	7	3.8	91	339 022	10	3.4	37
Cuello uterino	604 127	8	3.1	139	341 831	9	3.4	82
Esófago	604 100	9	3.1	78	544 076	6	5.5	68
Tiroides	586 202	10	3.0	68	43 646	25	0.44	5
Vejiga	573 278	11	3.0	64	212 536	14	2.1	18
Linfoma de Hodgkin	544 352	12	2.8	62	259 793	12	2.6	27

**Tabla 2.2:** Nuevos casos y mortalidad de los 12 tipos de cáncer. Datos obtenidos de la Global Cancer Observatory (GCO), año 2020 [2]. Tabla modificada.

Asimismo, en la Tabla 2.3 se muestra su prevalencia (en 5 años) de estos tipos de cáncer es la siguiente:

Diseño y desarrollo de software para el análisis de patrones microscópicos de crecimiento tumoral en pacientes con carcinoma vesical.

Tipo de cáncer	Prevalencia, 5 años (todas las edades)	
	Número	Por cada 100 000 habitantes
Mama	7 790 717	20158
Pulmón	2 604 791	3342
Próstata	4 956 901	12613
Colon	3 045 225	3907
Estómago	1 805 968	2317
Riñón	994 539	1276
Rectal	2 066 732	2651
Cuello uterino	1 495 211	3869
Esófago	666 388	855
Tiroides	1 984 927	2546
Vejiga	1 720 625	2207
Linfoma de Hodgkin	1 544 488	1981

**Tabla 2.3:** Prevalencia (5 años) de los 12 tipos de cáncer más frecuentes alrededor del mundo. Datos obtenidos del GCO [2]. Tabla modificada.

En España, el cáncer es la segunda causa de muerte por detrás de las enfermedades cardiovasculares y representa el 26,4% del total de defunciones; siendo la primera causa de muerte en hombres y la segunda en mujeres [3].

Distinguiendo por sexos, los 5 tipos de cáncer más común son:

En mujeres: Mama, colorrectal (CCR), pulmón, endometrio y páncreas. Ver Tabla 2.4.

Y, en los hombres: próstata, CCR, pulmón, vejiga urinaria y riñón. Ver Tabla 2.5.

Incidencia				Mortalidad				Prevalencia	
Tipo tumor (ránking)	n	Tasa cruda	Tasa ajustada	Tipo tumor (ránking)	n	Tasa cruda	Tasa ajustada	Tipo tumor (ránking)	5 años
Mama	32825	138,8	75,4	Colorrectal	6645	28,1	8	Mama	129928
Colorrectal	14428	61	23,3	Mama	6421	27,2	10,6	Colorrectal	42121
Pulmón	6914	29,2	14	Pulmón	5337	22,6	10	Cuerpo útero	25723
Cuerpo útero	6784	28,7	13,7	Páncreas	3571	15,1	4,8	Tiroides	15332
Páncreas	3777	16	5,4	Estómago	2196	9,3	2,8	Linfoma no Hodgkin	11183

**Tabla 2.4:** Estimaciones de incidencia, mortalidad y prevalencia a cinco años. España 2018. Mujeres. Fuente [3]: Tabla modificada.

Incidencia				Mortalidad				Prevalencia	
Tipo tumor (r�anking)	n	Tasa cruda	Tasa ajustada	Tipo tumor (r�anking)	n	Tasa cruda	Tasa ajustada	Tipo tumor (r�anking)	5 a�os
Pr�stata	31728	139,4	73,1	Pulm�n	17559	77,2	34,4	Pr�stata	106941
Colorrectal	22744	99,9	45,2	Colorrectal	10038	44,1	16,8	Colorrectal	63593
Pulm�n	20437	89,8	42,1	Pr�stata	5793	25,5	7,4	Vejiga	48571
Vejiga	14793	65	27,5	Vejiga	4576	20,1	6,6	Pulm�n	20472
Ri�n�n	5773	25,4	13,2	H�gado	3872	17	7,6	Ri�n�n	15229

**Tabla 2.5:** Estimaciones de incidencia, mortalidad y prevalencia a cinco a os. Espa a 2018. Hombres. Fuente [3]: Tabla modificada.

## 2.2 C ncer de Vejiga: Inicio, crecimiento y propagaci n, incidencia y mortalidad.

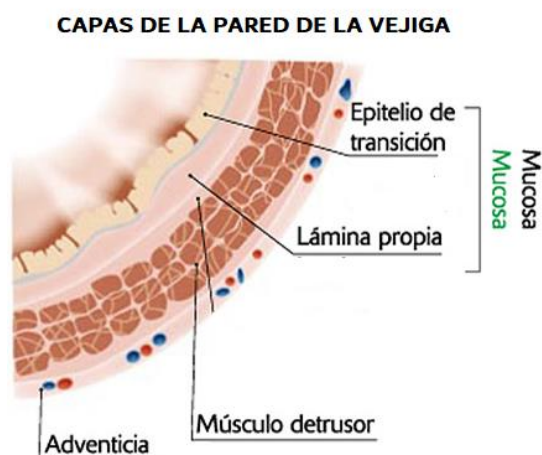
### 2.2.1. C ncer de vejiga

El carcinoma urotelial o tambi n conocido como c ncer de vejiga, se caracteriza por un crecimiento anormal e incontrolado de las c lulas que conforman la vejiga y que da lugar a la formaci n de uno o m s tumores; llegando esta alteraci n a afectar a otros  rganos en estadios m s avanzados. Es uno de los m s frecuentes en el mundo, seg n los datos recogidos por el Global Cancer Observatory (GCO) ocupando el d cimo primer puesto, como se puede observar en los datos ofrecidos por la fuente [2] y reflejados en la **Tabla 2.2** de este mismo documento.

### 2.2.2. Inicio, crecimiento y propagaci n

La vejiga es un  rgano hueco que se encuentra en la parte inferior del abdomen y se caracteriza por ser una bolsa o globo que tiene la capacidad de expandirse (cuando almacena la orina producida por los ri ones) y contraerse (cuando expulsa la orina al exterior a trav s de la uretra).

Est  compuesta por varias capas: la serosa, la muscular y la mucosa [4].



**Figura 2.3:** Capas de la pared vesical. Imagen obtenida del blog "Understand to learn", fuente [5].

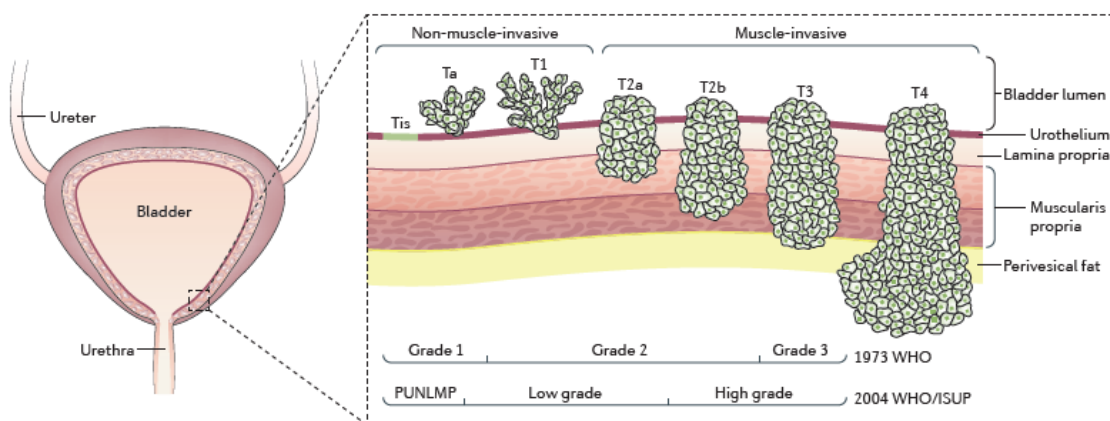
La capa serosa o adventicia: es la que recubre la parte exterior de la vejiga y se encuentra rodeada por el peritoneo por su cara superior, posterior y lateral cuando está llena.

La capa muscular: está formada por músculo liso y consta de 3 capas formadas cada una de ellas por fibras musculares (FM); la capa externa o superficial (por FM longitudinales), la capa media (por FM circulares) y la interna o profunda (también formada por FM longitudinales). Estas 3 capas musculares forman el músculo detrusor, que cuando se contrae expulsa la orina y que está controlado por los esfínteres de la uretra.

Por último, la capa mucosa: está formada por el epitelio de transición urinario (también conocido como urotelio) que es un epitelio estratificado (hasta 8 capas de células) e impermeable (en contacto con la orina) y la lámina propia que está formada por tejido conjuntivo.

### Crecimiento y propagación

Se distinguen dos tipos de crecimiento tumoral vesical: el músculo no invasivo y el músculo invasivo.



**Figura 2.4:** Crecimiento tumoral vesical (músculo no invasivo y músculo invasivo). Imagen obtenida del artículo de Oner S et al. (2017), "Bladder cancer". Fuente [8].

El cáncer de vejiga músculo no invasivo (o en inglés, Non-Muscle -Invasive Bladder cancer, NMIBC) es el tipo más frecuente de cáncer de vejiga y se denomina así porque el crecimiento tumoral es superficial, es decir, crece en la cavidad interior de la vejiga sin llegar a afectar al tejido muscular. Estos tumores superficiales se presentan en los primeros estadios de la enfermedad.

El cáncer de vejiga músculo invasivo (o en inglés, Muscle-Invasive Bladder cancer, MIBC) se denomina así porque las células anormales de la vejiga tienden a invadir la parte muscular de la pared de la vejiga (estadios T2 y T4). Este tipo de cáncer tiene mayor probabilidad de causar metástasis, es por ello, por lo que necesita una forma de tratamiento diferente y más radical [11].

En los casos en donde el MIBC presente metástasis (4% de los nuevos casos), la tasa de supervivencia global va a depender de dónde se encuentre localizado el tumor: la variabilidad de esta tasa es del 5% (distante) y del 36% (regional) [6].

Cuando estas células tumorales malignas se encuentran en el interior de la vejiga, el crecimiento tumoral se puede dar de 3 formas: local, por diseminación linfática y por diseminación hematógena.

Local: invade en profundidad de las células uroteliales. Traspasa la lámina propia e infiltra la

capa muscular. Puede producir afección a órganos cercanos como el útero o la vagina (en mujeres) y próstata (en hombres).

Diseminación linfática: la afectación se produce a los ganglios linfáticos más próximos. Esto se produce después de haber afectado la capa muscular de la vejiga.

Diseminación hematógena: la propagación se realiza a través de los vasos sanguíneos, con preferencia hacia el hígado, los pulmones y el hueso [4].

Una vez presentados estos dos tipos de crecimiento tumoral, el presente trabajo se enfocará en el estudio del crecimiento tumoral de músculo invasivo, dada su radicalidad y capacidad de diseminación celular anormal hacia otros órganos o tejidos adyacentes.

### 2.2.3. Incidencia

El cáncer de vejiga ocupa el décimo primer puesto de los cánceres más comunes, tal y como se puede observar en la Tabla 2.2 del punto 2.1.4; y es una enfermedad que afecta más a hombres que a mujeres.

Según los datos proporcionados por Globocan [12], el número de casos nuevos en el año 2020 fue de 573.278 de los cuales 440.864 fueron diagnosticados en hombres (lo que representa al 4,4% del total de los casos nuevos) y 132.414 en mujeres, estos datos se encuentran distribuidos en el mundo de la siguiente manera. Ver Tabla 2.6.

Diseño y desarrollo de software para el análisis de patrones microscópicos de crecimiento tumoral en pacientes con carcinoma vesical.

<b>Incidencia</b>						
	<b>Ambos sexos</b>		<b>Hombres</b>		<b>Mujeres</b>	
	Nuevos casos	Riesgo acumulado 0-74(%)	Nuevos casos	Riesgo acumulado 0-74(%)	Nuevos casos	Riesgo acumulado 0-74(%)
África del este	6 894	0.35	4 020	0.45	2 874	0.26
África media-central	1 366	0.19	841	0.25	525	0.13
Norte de África	18 589	1.08	15 131	1.83	3 458	0.37
Sur de África	2 208	0.47	1 605	0.82	603	0.21
África occidental	4 139	0.24	2 399	0.30	1 740	0.19
Caribe	2 562	0.44	1 830	0.70	732	0.22
América Central	4 119	0.26	2 963	0.42	1 156	0.13
América del Sur	27 159	0.53	19 618	0.85	7 541	0.26
América del Norte	89 997	1.26	69 080	2.03	20 917	0.55
Asia del este	132 316	0.49	102 408	0.80	29 908	0.20
Sureste de Asia	18 911	0.30	14 620	0.51	4 291	0.12
Sur y centro de Asia	35 949	0.23	29 000	0.39	6 949	0.08
Asia Occidental	20 915	1.05	17 498	1.85	3 417	0.31
Centro y Este de Europa	51 027	1.08	38 764	2.04	12 263	0.40
Europa occidental	68 143	1.53	51 854	2.47	16 289	0.67
Sur de Europa	61 480	1.86	48 889	3.15	12 591	0.67
Norte de Europa	23 333	1.02	17 151	1.58	6 182	0.49
Australia y Nueva Zelanda	3 923	0.59	2 995	0.94	928	0.26
Melanesia	201	0.35	163	0.60	38	0.12
Polinesia	33	0.63	24	0.91	9	0.35
Micronesia	14	0.21	11	0.43	3	0
Bajo HDI	13 025	0.30	8 076	0.40	4 949	0.20
Medio HDI	38 712	0.22	30 178	0.37	8 534	0.09
Alto HDI	164 626	0.48	126 996	0.78	37 630	0.20
Muy alto HDI	356 601	1.20	275 369	2.02	81 232	0.48
<b>Alrededor del mundo</b>	<b>573 278</b>	<b>0.64</b>	<b>440 864</b>	<b>1.05</b>	<b>132 414</b>	<b>0.26</b>

**Tabla 2.6:** Incidencia del cáncer de vejiga alrededor del mundo. Fuente Globocan 2020. Modificada [12].

En España, según los datos ofrecidos por la misma fuente [2] durante el año 2020, el número de



nuevos casos de cáncer de vejiga diagnosticados en los hombres fue de 14.838 que se corresponde al 9.1% de un total de 163.730 nuevos casos de todos los tipos de cáncer.

La edad estándar establecida para el cálculo de la ratio de la incidencia del cáncer de vejiga en el todo mundo es de 9.5 años en hombres y de 2.4 años en mujeres respectivamente siendo la edad media estándar en el continente europeo de 19.6 años en hombres y 4.8 años en mujeres. En España esta edad media se establece en 15.8 años en hombres y 3.5 años en mujeres.

Según la información proporcionada por la que se puede llegar a deducir que esta edad estándar es directamente proporcional al nivel de riqueza de los países que conforman el continente; ya que esta edad es visiblemente menor en aquellos continentes que están conformados por países que están considerados como subdesarrollados.

En el cálculo de la tasa estandarizada por edad, una población se ajusta matemáticamente para tener la misma estructura de edad que la otra; o ambas poblaciones se ajustan matemáticamente para tener la misma estructura de edad que una tercera población, llamada población estándar. De esta manera, a los dos grupos se les da la misma estructura de distribución por edad para que se proporcione una imagen más representativa de la característica en cuestión.

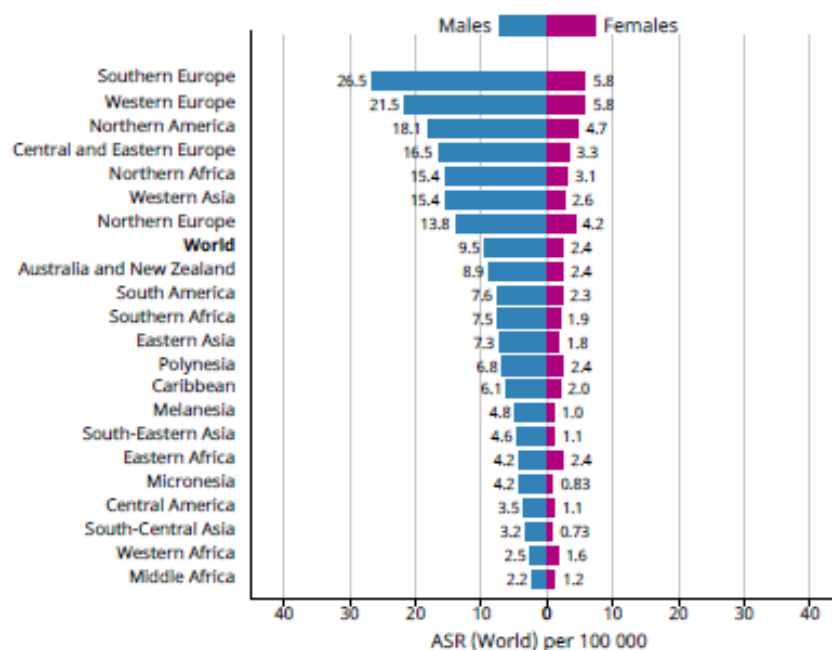


Figura 2.5: Edad estándar (mundial) para el cálculo de la ratio de la incidencia del cáncer de vejiga. Fuente [2].

#### 2.2.4. Mortalidad

El número de fallecidos por cáncer de vejiga durante el año 2020 fue de 212.536 alrededor del mundo, (estos datos se recogen en **Tabla 2.2**), situándose de esta forma en el puesto décimo tercero de todas las muertes por cáncer.

La distribución a nivel mundial de la mortalidad por continentes es la siguiente:

<b>Mortalidad</b>						
	<b>Ambos sexos</b>	<b>Riesgo acumulado</b>	<b>Hombres</b>	<b>Riesgo acumulado</b>	<b>Mujeres</b>	<b>Riesgo acumulado</b>
	Deaths	0-74 (%)	Deaths	0-74 (%)	Deaths	0-74 (%)
África del este	3 926	0.19	2 298	0.25	1 628	0.15
África media-central	784	0.10	483	0.14	301	0.07
Norte de África	10 787	0.53	8 769	0.91	2 018	0.18
Sur de África	853	0.17	566	0.28	287	0.09
África occidental	2 397	0.15	1 386	0.18	1 011	0.11
Caribe	1 302	0.18	917	0.28	385	0.09
América Central	1 690	0.08	1 207	0.13	483	0.04
América del Sur	10 108	0.16	7 067	0.25	3 041	0.08
América del Norte	21 045	0.19	15 129	0.30	5 916	0.10
Asia del este	54 206	0.15	39 717	0.23	14 489	0.06
Sureste de Asia	10 327	0.14	7 857	0.23	2 470	0.05
Sur y centro de Asia	18 243	0.11	14 838	0.19	3 405	0.04
Asia Occidental	7 834	0.34	6 472	0.61	1 362	0.10
Centro y Este de Europa	18 929	0.32	14 797	0.66	4 132	0.09
Europa occidental	20 866	0.29	15 341	0.46	5 525	0.13
Sur de Europa	17 931	0.33	14 000	0.56	3 931	0.11
Norte de Europa	9 563	0.25	6 678	0.37	2 885	0.14
Australia y Nueva Zelanda	1 615	0.14	1 160	0.19	455	0.08
Melanesia	104	0.18	85	0.32	19	0.07
Polinesia	19	0.35	12	0.42	7	0.28
Micronesia	7	0.16	6	0.33	1	0
Bajo HDI	7 693	0.17	4 812	0.24	2 881	0.12
Medio HDI	20 719	0.11	16 414	0.19	4 305	0.04
Alto HDI	75 297	0.17	57 133	0.29	18 164	0.07
Muy alto HDI	108 713	0.24	80 336	0.41	28 377	0.10
<b>Alrededor del mundo</b>	<b>212 536</b>	<b>0.18</b>	<b>158 785</b>	<b>0.30</b>	<b>53 751</b>	<b>0.08</b>

**Tabla 2.7:** Índice de mortalidad del cáncer de vejiga alrededor del mundo. Fuente [12].

En España, las muertes provocadas por el cáncer de vejiga fueron de 5.585 muertes (en 2020) que representa al 4.9% de todas las muertes por cáncer [2].

## 2.3 Proceso diagnóstico, estadio de la enfermedad, imagen histológica y patrones nodulares más comunes.

### 2.3.1. Proceso diagnóstico: Exploración física, cistoscopia, citología y marcadores urinarios.

#### ▪ *Exploración física*

Un síntoma de la posible presencia de carcinoma urotelial es la hematuria, ésta puede ser indolora además de que puede presentarse de manera macroscópica o microscópica.

La hematuria macroscópica es más fácil de detectar puesto que la presencia de sangre en la orina es lo que alerta al paciente (y al especialista) y, que además de esto puede presentar o no más síntomas, su incidencia se sitúa entre el 10-20% de los casos de carcinoma urotelial diagnosticados.

La hematuria microscópica se detecta a través de un examen de orina, en donde previamente el paciente al acompañar (o no) otros síntomas como: almacenamiento deficiente de orina, el incremento de la frecuencia y la disuria urinaria. Su incidencia es del 2-5% de los casos de cáncer de vejiga diagnosticados [13].

#### ▪ *Cistoscopia*

Antes de realizar una cistoscopia, es recomendable primero realizar un estudio cito-bacteriológico de la orina, que debe estar estéril.

La cistoscopia diagnóstica se efectúa con anestesia local y puede estar indicada para los siguientes casos:

- ✓ Se sospeche de un tumor vesical.
- ✓ Si tiene una ecografía con signos muy orientadores de un tumor vesical, ésta es opcional cuando se va a realizar una resección endoscópica.
- ✓ La fibroendoscopia permite precisar: el número, el tamaño, la topografía y el aspecto del tumor y de la mucosa vesical [13].

Su sensibilidad es de 71% y la especificidad del 72%.

#### ▪ *Biopsia*

La biopsia es un procedimiento que consiste en extraer pequeñas muestras del tejido anómalo, en ese caso, la vejiga para poder determinar el tipo de tumor, el estado en el que se encuentra y el grado de este.

Este tejido, en su amplio espectro incluye formas superficiales invasivas y metastásicas, cada una de ellas con su propio comportamiento clínico, pronóstico y tratamiento.

Una vez se le hayan realizado las pruebas pertinentes al paciente y se tenga constancia de la presencia de carcinoma urotelial, se procede a realizar la biopsia del tumor. Normalmente esta biopsia se realiza a través de una cistoscopia y cuando el tumor está en estadios iniciales se suele recurrir a ella para realizar una extracción de estos. A esta intervención se le conoce como resección transuretral de tumor de vejiga (TURBT).

- *Citología urinaria*

La citología urinaria tiene una sensibilidad elevada para la detección de las células tumorales de alto grado y más de forma particular para el carcinoma in situ (CIS), su sensibilidad puede variar entre 28-100%, siendo poco sensible para tumores de bajo grado [13].

Es importante añadir que una citología negativa no excluye la presencia de un tumor en las vías excretoras altas, así como también, una citología positiva no es indicativo solo de carcinoma urotelial ya que puede tratarse de un tumor del tracto urinario.

- *Marcadores urinarios*

Son menos invasivos que la cistoscopia y en algunos casos son buenos detectores de la presencia de cáncer de vejiga. Estos marcadores se suelen utilizar para medir la eficacia del tratamiento que se le esté realizando a la persona que lo padece. Se distinguen 2 grupos principales de marcadores urinarios: los basados en el sedimento celular (marcadores celulares) y los basados en el análisis del sobrenadante (marcadores solubles).

Los primeros requieren de una técnica inmunohistoquímica o de hibridación in situ (FISH) y los segundos se realizan a partir de un análisis de inmuno-absorción ligado a enzimas (ELISA).

Los marcadores más comunes son las pruebas BTA stat, BTA TRAK y NMP22 que son marcadores solubles y por otro lado están las pruebas Urovysion (FISH) e InmunoCyt que son marcadores celulares [13].

No son del todo recomendables porque existen tumores que pueden no segregar la sustancia que es lo que se conoce como marcador tumoral.

- *Estadio de la enfermedad*

Con la estadificación lo que se quiere conseguir es determinar la elección de la terapia adecuada para el paciente, es por esto por lo que es considerado un punto diagnóstico importante.

En cuanto a conceptos de gradación histológica (por similitud con la arquitectura histológica normal del urotelio) y el grado de invasión, en la actualidad se distinguen entre neoplasias de alto (este tipo de cáncer tiene más probabilidades de reaparecer y crecer) y bajo grado (este tipo de cáncer puede reaparecer). Esto remplazaría al sistema previo de clasificación en las que estos grados se designaban por: bajo (G1), medio (G2) y alto (G3) [15].

Una herramienta que resulta de gran ayuda para el especialista es el Sistema Tumor-Metástasis-Nodo (TMN). Este sistema le permite al especialista recopilar la siguiente información:

*Tumor (T):* Indica su tamaño y ubicación;

- Va acompañada de una letra y/o número (0-4), ejemplo: T0
- Si hubiese más de un tumor, se añade la letra "m" (múltiple), ejemplo: Tm
- Si fuese acompañado de la letra "c" significa que el estadio ha sido determinado de forma clínica.
- Si fuese acompañado de la letra "p" significa que el estadio ha sido determinado de forma patológica.
- Si se extirpa el tumor, la información de su estadio con más detalle se indica en la Tabla 2.8.

<b>Tipos de tumor (T)</b>
<b>TX:</b> no se puede evaluar el tumor primario.
<b>T0:</b> No hay evidencia de tumor primario en la vejiga
<b>Ta:</b> Carcinoma papilar no invasivo. Se puede extirpar fácilmente con TURBT.
<b>Tis:</b> Carcinoma in situ (CIS) o un tumor plano. Es conocido como cáncer de vejiga no invasivo de los músculos o carcinoma plano no invasivo. Suele reaparecer después del tratamiento de forma general como otro cáncer de vejiga no músculo invasivo.
<b>T1:</b> El tumor se ha diseminado hacia la lámina propia, pero no afecta al músculo de la pared de la vejiga.
<b>T2:</b> el tumor se ha diseminado hasta el músculo de la pared de la vejiga. Se distinguen 2 tipos dentro de esta clasificación y son: el <b>T2a:</b> que se disemina hacia el músculo superficial (mitad-interna) y el <b>T2b:</b> que hace referencia a que el tumor se ha diseminado al músculo profundo de la vejiga (mitad-exterior).
<b>T3:</b> el tumor ha crecido hacia el tejido perivesical y al igual que el anterior se distinguen 2 tipos: el <b>T3a:</b> el crecimiento de este tumor se puede ver de forma microscópica y el T3b: en donde el tumor es lo suficientemente grande y apreciable a través de pruebas de imagen médica o palpación del especialista.
<b>T4:</b> El tumor se ha invadido cualquiera de los siguientes lugares: pared abdominal, la pared pélvica, la próstata o vesícula seminal ( <i>en hombres</i> ) o el útero o la vagina ( <i>en mujeres</i> ). Se distinguen dos tipos el <b>T4a</b> en donde la extirpación quirúrgica es posible y el <b>T4b</b> en donde no lo sea.[11]

**Tabla 2.8:** Tipos de tumor. Información obtenida de referencia [11].

El *Ganglio (Node, N)*: hace referencia al alcance del tumor hacia los ganglios linfáticos y si éste se ha diseminado, a cuáles o a cuántos de ellos están afectados. Se denominan ganglios linfáticos regionales aquellos que se encuentran cercanos a la vejiga (es decir los que están dentro de la pelvis) y son: los hipogástricos, obturadores, ilíacos, perivesicales, pélvicos, sacros y presacra. Los ganglios linfáticos que se encuentran en otras partes del cuerpo se denominan ganglios linfáticos distantes. En la **Tabla 2.9** se ofrece más información acerca de estos.

<b>Tipos de nodos (N)</b>
<b>NX:</b> Los ganglios regionales no se pueden evaluar.
<b>N0:</b> el cáncer no se ha diseminado a los ganglios linfáticos regionales.
<b>N1:</b> Un ganglio de los ganglios linfáticos regionales está afectado
<b>N2:</b> 2 o más ganglios comprometidos de los ganglios linfáticos regionales.
<b>N3:</b> El cáncer se ha diseminado hacia los ganglios linfáticos ilíacos comunes.

**Tabla 2.9:** Tipos de nodos información obtenida de referencia [11].

Y la *Metástasis*: ayuda a determinar si el cáncer se ha diseminado hacia otros órganos y si lo ha hecho nos permite precisar dónde y en qué medida lo ha hecho. En la Tabla 2.10 se ofrece más información para su mejor comprensión.

<b>Tipo de metástasis</b>
<b>M0:</b> la enfermedad no ha hecho metástasis.
<b>M1:</b> Hay metástasis a distancia. Dentro de este tipo se distinguen 2 clases: el <b>M1a</b> (se ha diseminado solo hacia los ganglios linfáticos fuera de la pelvis) y el <b>M1b</b> (el carcinoma urotelial se ha diseminado a otras partes del cuerpo).

**Tabla 2.10:** Tipos de metástasis. Información obtenida de referencia[11].

Estos resultados se combinan y ayudan a determinar el estadio de afección de cada persona. Hay 5 estadios: el estadio 0 (cero) y los estadios I a IV [11]. Esta determinación de los estadios puede ser clínica o patológica y se basa en lo que se encuentra durante la cirugía; incluye resultados de los exámenes físicos, las exploraciones por imágenes y las biopsias.

<b>Agrupación por etapas del cáncer</b>	<b>Sistema TMN</b>
<b>Etapa 0a:</b> Cáncer temprano, no invasivo	Ta, N0, M0
<b>Etapa 0is:</b> CIS, se encuentra en el revestimiento interno de la vejiga. Si no ha crecido hacia la parte hueca y no se ha extendido hacia el músculo o tejido conectivo de la vejiga. Este cáncer es considerado de alto grado y es considerado como agresivo.	Tis, N0, M0
<b>Etapa I:</b> El tumor se ha diseminado hasta la lámina propia, pero no hacia la capa gruesa de músculo en la pared, ni a los ganglios u otros órganos.	T1, N0, M0
<b>Estadio II:</b> es el conocido como MIBC y es el objeto de estudio.	T2, N0, M0.
<b>Etapa III:</b> El tumor se ha diseminado hasta el tejido perivesical o la próstata en un hombre o el útero y la vagina en una mujer o se ha diseminado a los ganglios linfáticos regionales.	<b>(IIIa):</b> solo si afecta a los explicados en la etapa III → T3a, T3B o T4a, N1, M0 <b>(IIIb):</b> El cáncer se ha diseminado a los ganglios linfáticos ilíacos comunes (T1 a T4a, N2 o N3, M0).

<p><b>Estadio IV:</b> El tumor se ha diseminado a la pared pélvica o a la pared abdominal o a los ganglios linfáticos fuera de la pelvis o a otras partes del cuerpo.</p>	<p><b>IVa:</b> Solo a la pared pélvica o abdominal → T4b, cualquier N, M0. A los ganglios linfáticos ubicados fuera de la pelvis → cualquier T, cualquier N, M1a. <b>IVb:</b> El cáncer se ha diseminado a otras partes del cuerpo (cualquier T, cualquier N, M1b)</p>
---	--

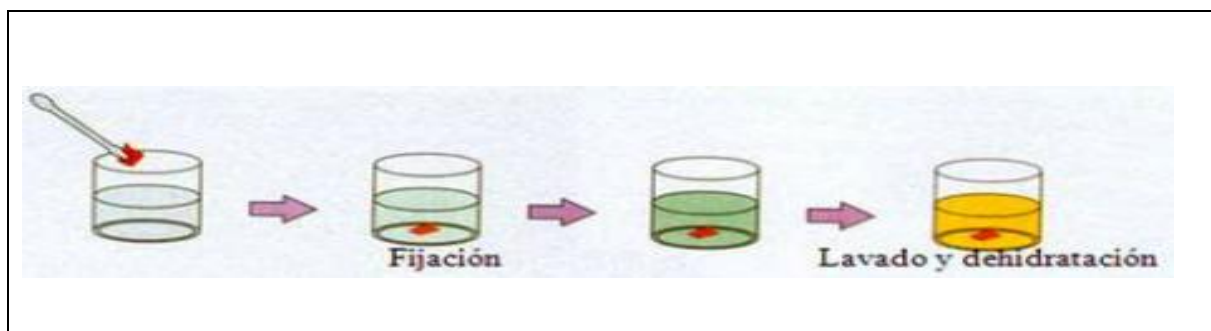
**Tabla 2.11:** Agrupación por etapas o estadios del cáncer de vejiga. Información obtenida de referencia [11].

### 2.3.2. Imagen histológica.

Una vez recogida la muestra, el especialista tiende a examinarla a través de la imagen microscópica. Para la correcta visualización de esta imagen, la muestra pasa por una serie de procedimientos; Estos se describen a continuación, en los apartados 2.3.2.1 al 2.3.2.4.

#### 2.3.2.1. Fijación.

Con la fijación, lo que se quiere conseguir es evitar la lisis de las células que conforman el tejido que es objeto de estudio, de esta manera se consigue conservar su estructura. En este paso se utiliza como sustancia fijadora la formalina al 10%, seguidamente se deshidrata el tejido y se elimina el alcohol existente en el mismo, con la finalidad de que la muestra se torne transparente. Este procedimiento se muestra en la Figura 2.6.



**Figura 2.6:** Fijación, lavado y deshidratación de la muestra. Figura modificada de referencia [16].

#### 2.3.2.2. Inclusión.

Una vez fijada la muestra, se procede a darle cierta dureza y consistencia (*que en condiciones normales no tiene*), para ello lo que se hace es infiltrarle sustancias de inclusión. Entre las sustancias de inclusión más utilizadas se encuentra la parafina, con ella se consigue que la muestra tenga un buen soporte (*ya que se forman bloques y dentro de estos bloques están las muestras a estudiar*, tal y como se ve en la Figura 2.7) para poder llevar a cabo el siguiente procedimiento.

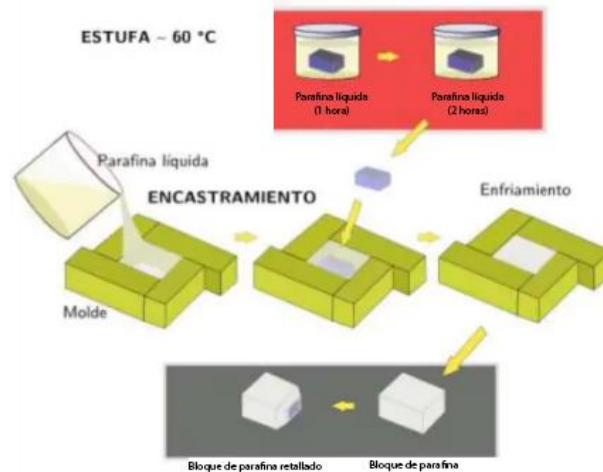


Figura 2.7: Inclusión de una muestra histológica. Imagen modificada de referencia [17].

### 2.3.2.3. Microtomo.

Una vez realizada la inclusión en la muestra, se proceden a realizar los cortes histológicos. Para conseguirlo se hace uso del microtomo. El microtomo es una herramienta semi automática que permite realizar secciones de la muestra con un grosor homogéneo ( $5\mu\text{m}$ ).

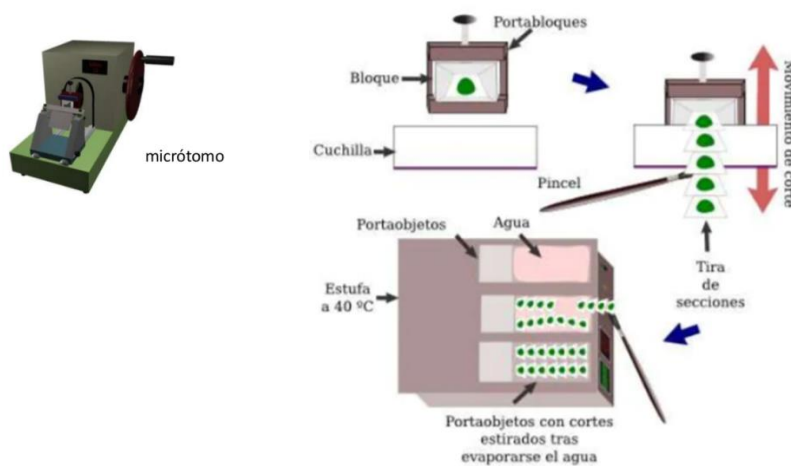


Figura 2.8: Corte de las muestras utilizando el microtomo. Imagen modificada de referencia [17].

### 2.3.2.4. Montaje y tinción.

En este paso se procede a colocar la muestra en un portaobjetos, seguidamente se le aplica la técnica de tinción más apropiada para el objeto de estudio. En este caso la técnica de tinción empleada a la muestra es la inmunohistoquímica CK-AE1/3 ya que es la que facilita la localización del frente tumoral al producirse la unión antígeno-anticuerpo. Un apunte importante es que la muestra debe desparafinarse previamente y pasar por una serie de soluciones para finalmente deshidratarse y montarse en el portaobjetos.



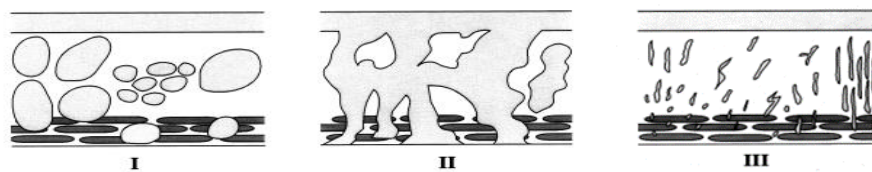


**Figura 2.9:** Procedimiento de tinción de una muestra histopatológica. Figura modificada de referencia [17].

### 2.3.3. Patrones morfológicos.

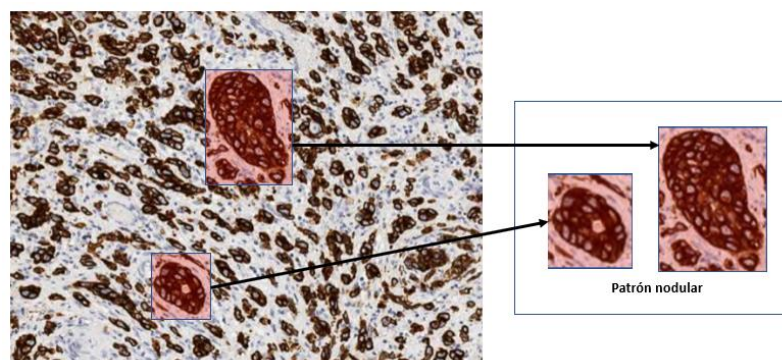
Aunque apenas esté explorado y, por tanto, reconocido el tipo de patrones morfológicos que se pueden encontrar en el MIBC, según Jiménez et al [18] se distinguen 3 tipos de patrones: el nodular (I), el trabecular (II) y el infiltrativo (III).

Estos 3 tipos de patrones se muestran en la Figura 2.10



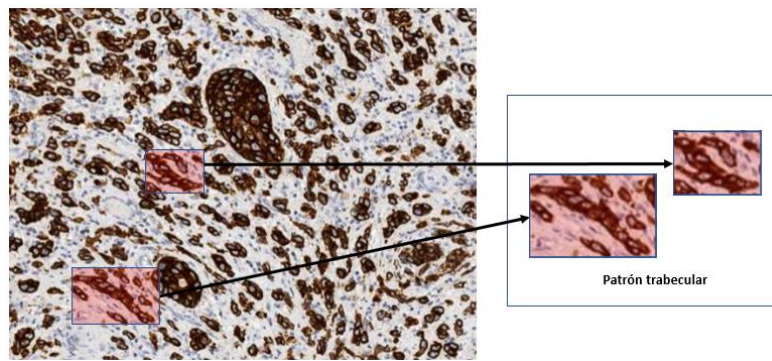
**Figura 2.10:** Patrones morfológicos; nodular (I), trabecular(II) e infiltrativo(III). Imagen obtenida de referencia [18].

*Patrón nodular o tipo I:* está compuesto por nidos de células tumorales bien delineadas. Su morfología es redondeada, tal como se observa en la Figura 2.11



**Figura 2.11:** Patrón nodular en imagen histológica con MIBC. Elaboración propia.

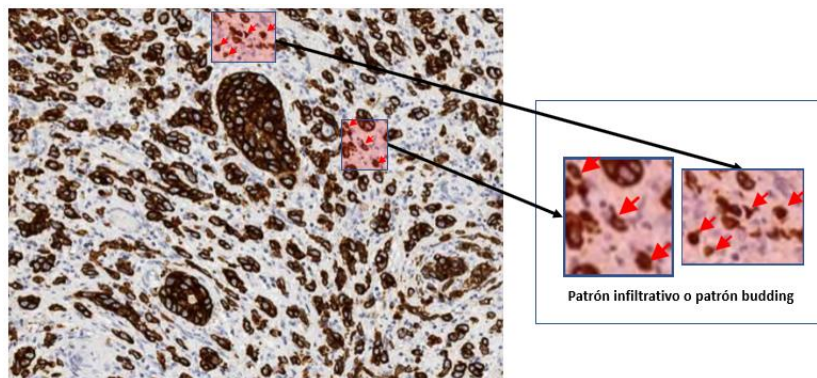
*Patrón trabecular o tipo II:* está compuesto por células tumorales dispuestas en bandas interconectadas entre sí. Su morfología es alargada, tal y como se observa en la Figura 2.12:



**Figura 2.12:** Patrón trabecular en imagen histológica con MIBC. Elaboración propia.

Y por último el;

*Patrón infiltrativo:* que está compuesto por estrechos cordones de células tumorales o células aisladas, forman grupos menores a 4 buds y es también conocido como tumor budding [19].



**Figura 2.13:** Patrón trabecular en imagen histológica con MIBC. Elaboración propia.

#### 2.3.4. El “tumor Budding” (TB) y el cáncer vesical

El valor pronóstico del TB es ampliamente conocido en otros tipos de carcinoma ya que un hallazgo en alto grado de este patrón es un factor de mal pronóstico para el paciente, por su alta capacidad invasora y, en consecuencia, una alta probabilidad de desarrollar metástasis, ya que estas células actúan a modo de avanzadilla preparando el terreno para que el frente invasor progrese. Entre los estudios más relevantes de TB se encuentra el del cáncer colorrectal (CCR) [21]

En la literatura actual, se ha escrito poco sobre la relación de éste y el carcinoma urotelial, aunque se empiezan a desarrollar algoritmos [20][22][33] partiendo de la hipótesis de que pueden resultar de gran ayuda como agente predictor para aquellos carcinomas que se detectan en estadios más tempranos.

El objetivo de este trabajo final de Máster es la de proporcionar nuevos algoritmos que puedan predecir la presencia de este tipo de patrón (TB) en muestras histológicas de pacientes con carcinoma vesical de alto grado (G2b-G3), y de esta manera poder facilitar al patólogo especialista un sistema de ayuda al diagnóstico computacional (CAD) basado en inteligencia artificial (AI).

## Capítulo 3

# Materiales

### ÍNDICE DE CONTENIDO:

3.1. Base de datos.....	27
3.2. Software utilizado .....	27
3.3. Hardware utilizado.....	30

### 3.1 Base de datos

Para la realización de este trabajo final de máster, el grupo de investigación Computer Vision Behavior Laboratory (CVBLab) que pertenece a la Universidad Politécnica de Valencia (UPV) me ha proporcionado una base de datos con 136 imágenes de cortes histológicos en formato (.tif).

La obtención de dichas imágenes, así como la conversión al formato adecuado se llevaron a cabo en: ref. [20]. Para ello se contó con la participación del Departamento de Anatomía patológica del Hospital Universitari i Politènic La Fe, que permitió la digitalización de las imágenes histológicas provenientes de biopsias de tejido vesical con presencia de carcinoma. El procedimiento que siguieron estas muestras para la obtención de las imágenes histológicas fue el explicado en el punto 2.3.2 y el escaneado de éstas fue a 40 aumentos (40px).

La digitalización del tejido histológico se realizó con el escáner Aperio CS2 (Leica Biosystems, San Diego, Estados Unidos) que además de crear portaobjetos digitales de alta calidad, tiene una capacidad de escaneo de 20px y 40px con una alta fiabilidad. Para el manejo de este escáner es necesario el software Aperio ImageScope. El formato de imagen que proporciona este software es el “.svs” [20].



**Figura 3.1:** (a) soporte para los portaobjetos analógicos, (b) Interfaz del software Aperio ImageScope y (c) Escáner Aperio CS2. Imagen modificada de la referencia [46].

Dado que estamos trabajando con imágenes histológicas que provienen de biopsias a pacientes con carcinoma urotelial, es necesario garantizar la privacidad de los datos de estos pacientes, por ello se ha optado por anonimizar estas imágenes y se ha firmado un contrato de confidencialidad antes de realizar el correspondiente preprocesado con ellas.

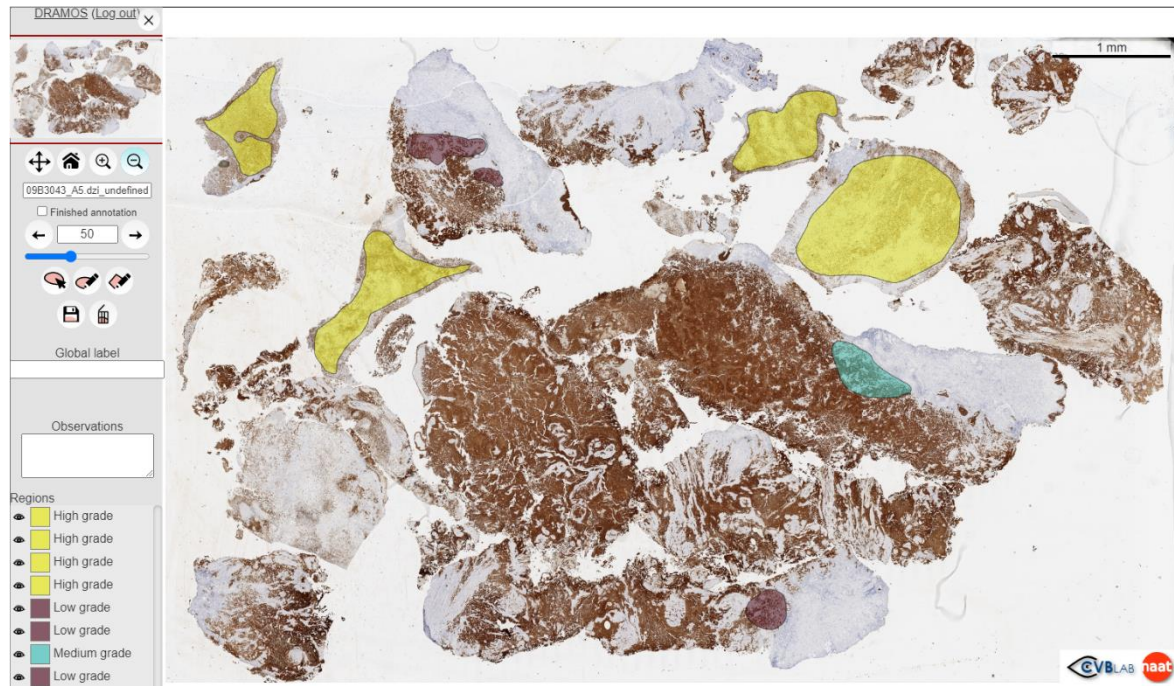
#### 3.1.1. Etiquetado de los datos.

Para la realización del etiquetado de las imágenes histológicas (*Whole slide image WSI*) se hace uso de la herramienta *Microdraw* (Figura 3.2).

## Diseño y desarrollo de software para el análisis de patrones microscópicos de crecimiento tumoral en pacientes con carcinoma vesical.

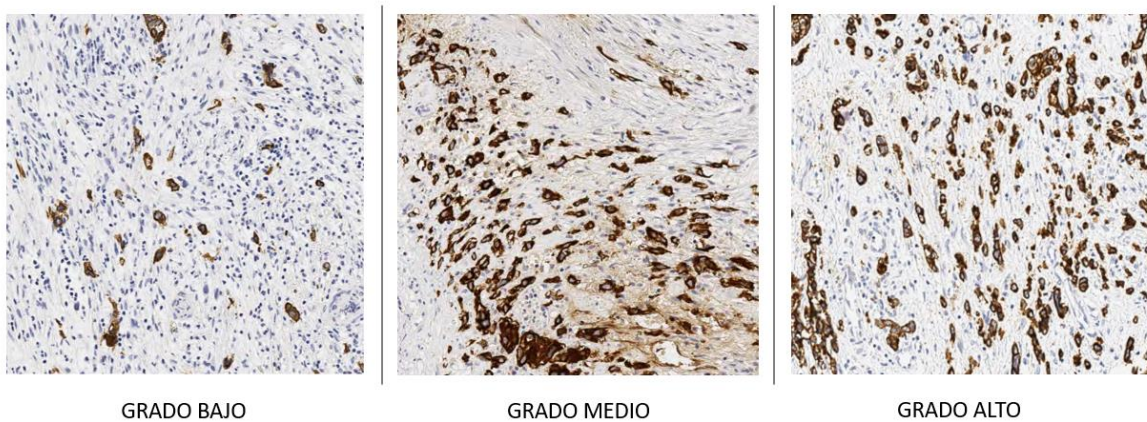
Esta herramienta está creada sobre una librería de código abierto llamada OpenSeandragon que consigue hacer un zoom de las WSI sin perder resolución y, para la visualización de las WSI se hace uso de la librería Openslide que es la que soluciona el problema de leer imágenes de alta resolución espacial [20].

El etiquetado de estas muestras se realizó bajo la supervisión del Doctor Don David Ramos Soler, patólogo experto del Hospital Universitario La Fe, en Valencia, en donde, dentro del grado alto ya existente en las muestras, se realizó una clasificación más refinada.



**Figura 3.2:** Captura de pantalla de la herramienta de etiquetado Microdraw que pertenece al grupo de investigación CVB Lab de la UPV.

Esta clasificación consistió en asignar un grado a aquellas partes de la WSI en donde el patrón budding o patrón infiltrativo tenía más o menos presencia, es decir, en donde su densidad era más representativa o no. Se asignaron 3 grados: bajo, medio y alto, los mismos que se encuentran representados en la siguiente Figura 3.3.



**Figura 3.3:** Clasificación de los grados de TB según su densidad encontrada en la WSI. Las imágenes se corresponden a patches de tamaño 512x512x3 píxeles. Elaboración propia.

Concluido el refinamiento del etiquetado de todas las WSI se procedió a la obtención de las máscaras.

Las máscaras son el resultado de las zonas asignadas previamente con la herramienta Microdraw, y lo que se obtiene a través de ellas son las coordenadas de dichas zonas, que es donde se encuentra el patrón de estudio en mayor o menor densidad. Para ello se han asignado 3 niveles de gris para la distinción de dichas zonas, correspondiéndose el nivel de gris 50 para la zona de grado bajo, 100 de grado medio y 150 de grado alto, respectivamente.

Para el preprocesado de estas máscaras se siguió el mismo procedimiento que el que se realizó con las WSI y que se explicará en el punto 4.1.1 de esta misma memoria.

## 3.2 Software utilizado

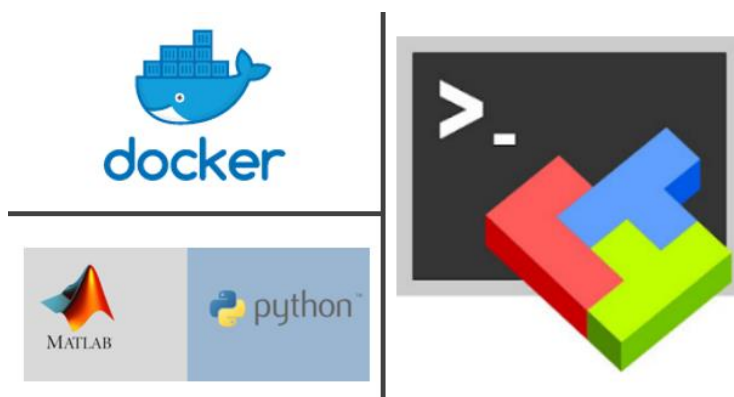


Figura 3.4: Logotipos de las herramientas software utilizadas; Docker, Matlab, Python y MobaXterm.

Para el análisis y la implementación de los algoritmos propuestos se utilizaron los entornos de programación MATLAB® R2019b [29] y Python 3.8 [28]. Estos dos entornos se encuentran embebidos por un contenedor de imágenes: Docker[27].

*Docker*: es una plataforma abierta que permite construir y ejecutar aplicaciones distribuidas basada en contenedores que se ejecutan en Linux y funcionan tanto en máquinas físicas como virtuales, haciendo uso de runtime. Estos contenedores aportan características y herramientas concretas tales como: la portabilidad, escalabilidad, alta disponibilidad y los microservicios en aplicaciones distribuidas [31].

*Python*: es un entorno de programación de propósito general y de alto nivel que permite la integración de módulos y paquetes de forma libre, a través de su cabecera. Dentro de este entorno haremos uso de la librería TensorFlow como *backend*.

Para la etapa de entrenamiento de los diferentes algoritmos de clasificación basados en aprendizaje profundo, se ha hecho uso de la librería NVIDIA Cuda® Deep Neural Network (cuDNN).

Se ha utilizado *PyCharm* como entorno de desarrollo integrado compatible con Python y, para dar soporte al servidor remoto en el que se han entrenado los algoritmos, se ha hecho uso del programa MobaXterm[25], ya que es una herramienta que permite generar una conexión SSH (Secure Shell) facilitando la comunicación segura entre el cliente y el servidor.

Por último, para la preparación de los datos y la validación de los resultados obtenidos, se ha utilizado Matlab que es un software de cómputo matemático con lenguaje de programación propio. En él podemos encontrar funciones y aplicaciones incorporadas que resultan de mucha utilidad

para el programador, además de estar disponible para las plataformas Unix, Windows, Mac OS X y GNU/Linux

### **3.3 Hardware utilizado**

Para el desarrollo y la ejecución de este trabajo final de máster se ha utilizado un ordenador modelo DELL 97FGJ64 con un sistema operativo Windows 10 de 64 bits, un procesador Intel® Core(TM) i7-3770 CPU (@3.4 GHz, 4 núcleos), un controlador gráfico Intel(R) HD Graphics 4000, un disco duro de 500 GB HDD y una memoria RAM de 8GB.

Para la etapa de entrenamiento, validación y testeo de los modelos es necesario el uso de un servidor de computación (NAS de datos) que tenga la capacidad suficiente para llevar a cabo este proyecto y dado que las imágenes con las que se ha trabajado en este proyecto ocupan mucho espacio en disco; se ha facilitado el uso del servidor NAS Synology DS918 de 32 TB de capacidad para almacenarlas. Ambos servidores pertenecen al Departamento del CVBLab de la Universidad Politécnica de Valencia.

Respecto a las características del servicio, este servidor está compuesto por un procesador amd64-038 @4.20GHz, 32GB de RAM y tarjetas gráficas NVIDIA Titan V de 12 GB de RAM.

## Capítulo 4

# Metodología

### ÍNDICE DE CONTENIDO:

4.1	Preparación de los datos: Etiquetado, generación de patches y partición de datos.....	33
4.2	Algoritmos implementados.....	39
4.3	Algoritmos de clasificación supervisada para las imágenes histopatológicas de los grados del carcinoma urotelial .....	44



Diseño y desarrollo de software para el análisis de patrones microscópicos de crecimiento tumoral en pacientes con carcinoma vesical.

Con el objetivo de presentar la metodología empleada correspondiente al proceso de diseño y desarrollo software para la detección de patrones microscópicos se expone un diagrama de flujo en donde se pueden ver representadas todas las fases seguidas en esta sección.

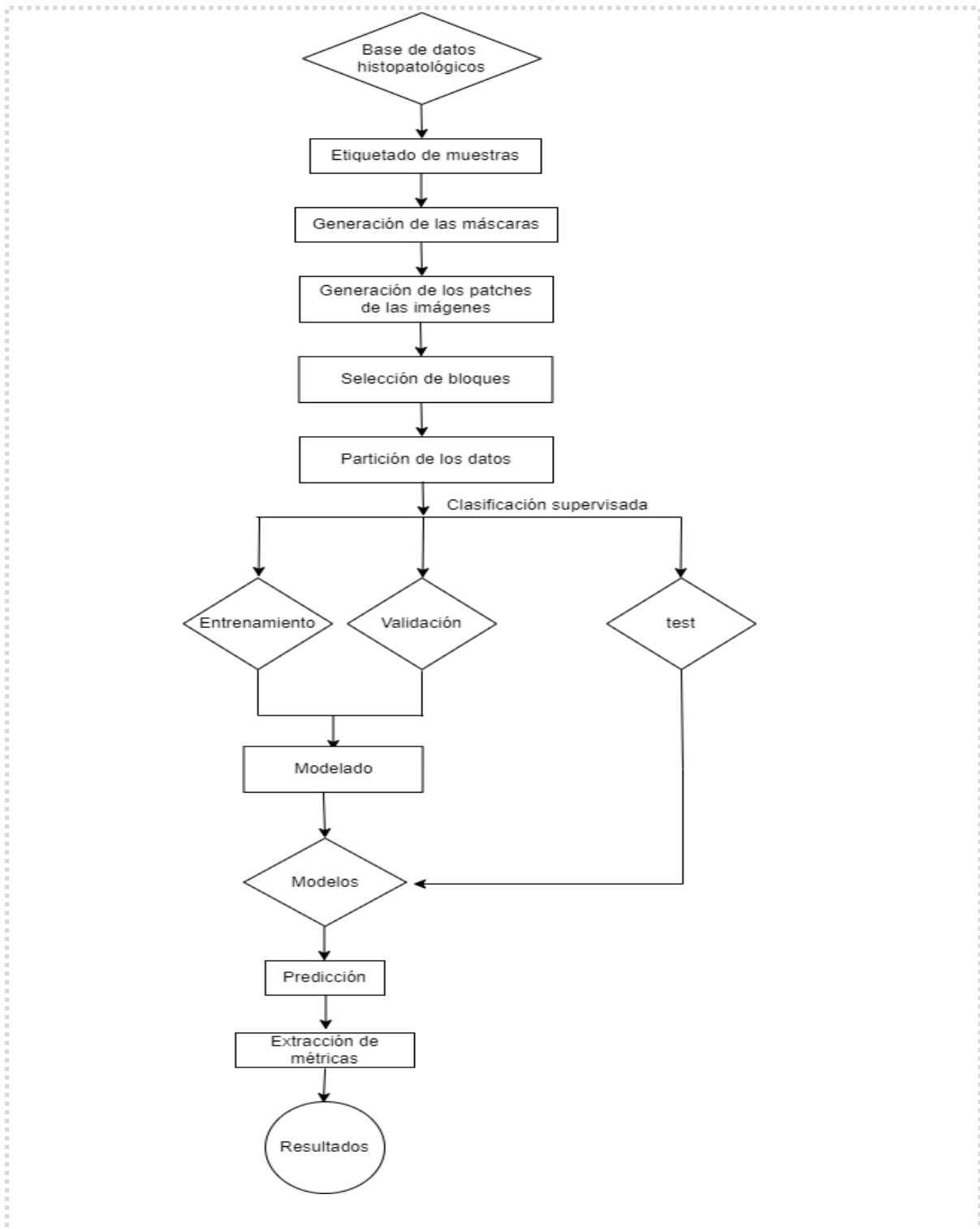


Figura 4.1: Diagrama de flujo correspondiente a la metodología empleada en este trabajo.

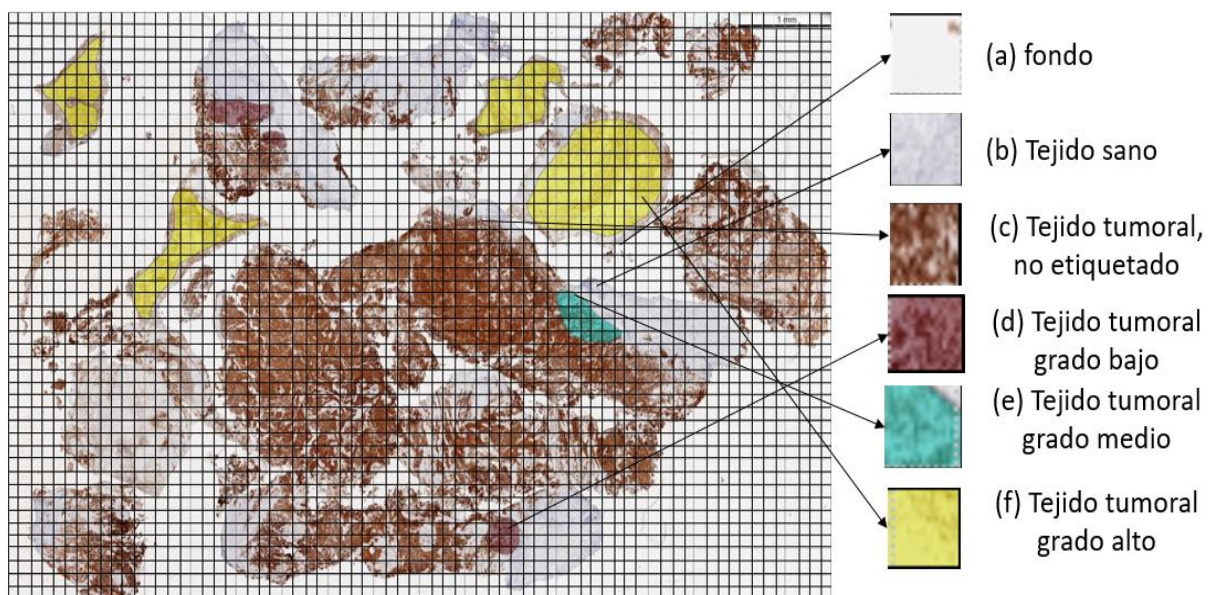
## 4.1 Preparación de los datos: Generación de patches y partición de datos.

### 4.1.1. Generación de patches.

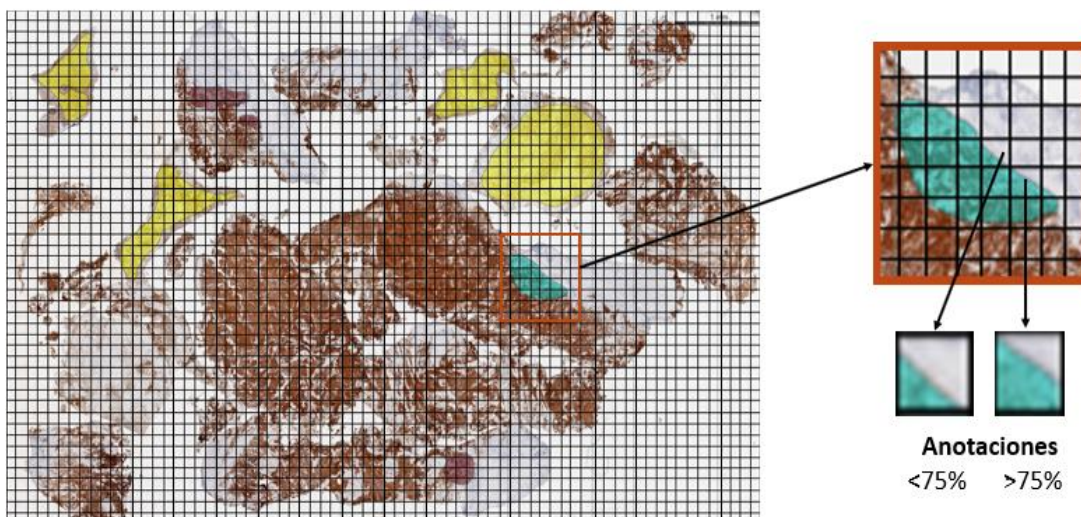
Para la realización de este preprocesado de datos se utilizó la herramienta software de cómputo matemático Matlab, la misma que fue explicada en el punto 0 de la sección anterior.

Este preprocesado consistió en la generación de sub-divisiones de la WSI y de sus respectivas máscaras, en tamaños de 512x512x3 píxeles. Estas sub-divisiones de la imagen y sus máscaras se conocen en *computer vision* (CV) como “patches” y tienen como finalidad; utilizar imágenes que tengan dimensiones aceptables para poder lidiar con los problemas de memoria a la hora de entrenar algoritmos de deep learning y, por otro lado, disponer de información de alta resolución de los patrones morfológicos a buscar.

Como se puede observar en la Figura 4.2 con la generación de estos patches se obtiene mucha información de: fondo <sup>(a)</sup>, tejido sano <sup>(b)</sup> y tejido tumoral no etiquetado <sup>(c)</sup>, que no es relevante para nuestro objeto de estudio. Es por ello, que de estos patches solo se seleccionaron aquellos que contenían más del 75% de información del etiquetado.



**Figura 4.2:** Sub-división de la WSI. Patches obtenidos: a) Fondo, b) Tejido sano, c) Tejido tumoral no etiquetado, d) Tejido tumoral en grado bajo, e) Tejido tumoral en grado medio y f) Tejido tumoral en grado alto. Elaboración propia.



**Figura 4.3:** Selección de parches obtenidos de la WSI. Elaboración Propia.

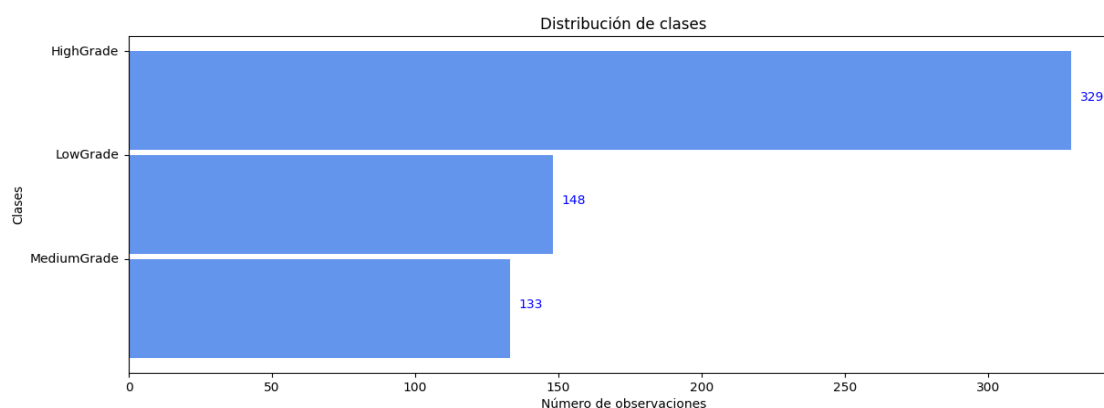
De las 136 WSI proporcionadas por la base de datos, 40 fueron descartadas a petición del patólogo experto puesto que no contenían información clara de los patrones a buscar.

Las 96 WSI restantes una vez etiquetadas, se procedió a la obtención de sus patches correspondientes, de las cuales solo 50 de ellas proporcionaron patches válidos que nos servirán para la generación de nuestro "ground truth".

En la Tabla 4.1 se resume el número total de patches obtenidos por las 50 WSI.

Número total de WSI ( <i>whole slide images</i> )	Número de patches obtenidos		
	Grado bajo	Grado medio	Grado alto
50	148	133	329

**Tabla 4.1:** Número de patches obtenidos de las WSI.



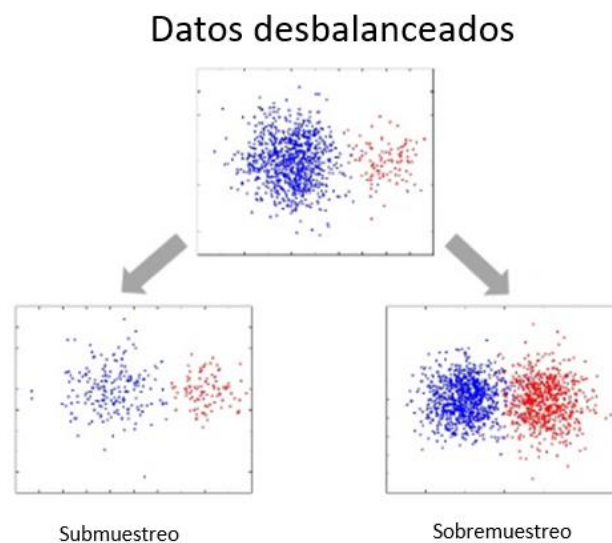
**Figura 4.4:** Distribución de clases del Ground truth. Elaboración propia.

A la vista de los datos presentados en Figura 4.4, el número de patches varía de forma considerable entre los distintos grados en los que el TB se encuentra presente en las WSI, por lo que será necesario implementar técnicas que nos ayuden a resolver el problema de desbalanceo evidente entre los tipos de grados existentes.

#### 4.1.2. Balanceo y creación de muestras sintéticas para la base de datos de imágenes histológicas

Un problema muy común que se suele presentar al momento de querer entrenar un modelo en aprendizaje profundo (AP), con una base de datos propia, es el desbalance de datos entre clases.

El remuestreo de datos, es uno de los enfoques más comúnmente preferidos para lidiar con un conjunto de datos que presente estas características; se distinguen dos tipos: *el submuestreo o undersampling y el sobremuestreo u oversampling*.



**Figura 4.5:** Ejemplo de submuestreo o under-sampling y sobremuestreo u over-sampling. Imagen modificada de referencia [35].

De estas dos técnicas se prefiere la de **sobremuestreo** que consiste en el aumento de datos, es decir, el número de muestras. La razón por la que se opta por ella es porque con el submuestreo se tiende a eliminar instancias de los datos que pueden contener información importante.

En este proyecto para solventar este desbalanceo o desequilibrio de datos, se hace uso de la técnica: ponderación de la función de pérdidas, que consiste en introducir un factor de ponderación en la función de pérdidas proporcionando un peso mayor a la clase minoritaria por lo que la función de pérdidas asignará un valor más alto a estas instancias.

La función de pérdidas utilizada fue, la entropía cruzada ponderada que se utiliza cuando el clasificador es multiclase.

Su fórmula es la siguiente:

$$ECP = -\frac{1}{N} \sum_{n=1}^N w r_n \log(p_n) + (1 - r_n) \log(1 - p_n),$$

Donde  $w$  es el peso atribuido a la clase mayoritaria, definida como  $w = \frac{N - \sum_n p_n}{\sum_n p_n}$ .

Skicit-learn [37] a través de la biblioteca sklearn-utils incorpora las clases: `Class_weight_compute_class_weight` y `class_weight`. La primera para calcular los pesos de las

clases y la segunda para modificarlos.

En nuestro caso los pesos calculados por la clase `Class_weight_compute_class_weight` son los siguientes:

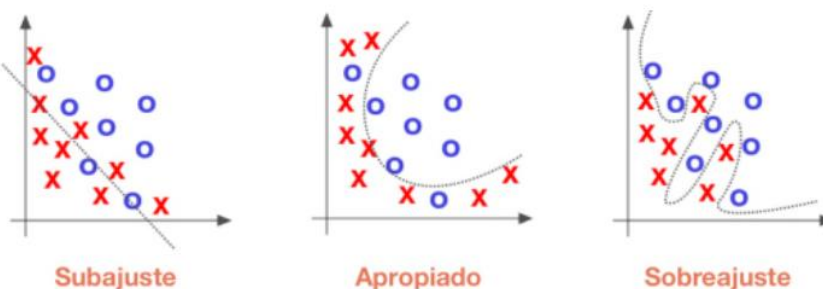
`Class_weight = {0: 0.9823, 1: 1.3922, 2: 0.7913}`, donde las clases 0: MediumGrade, 1: LowGrade y 2: HighGrade, respectivamente.

```
None
[INFO]: Compilando el modelo...
1111
276
[INFO]: Entrenando la red...
{0: 0.982316534040672, 1: 1.3922305764411027, 2: 0.7913105413105413}
WARNING:tensorflow:From Clasificador_Aguilar1.py:98: Model.fit_generator (from tensorflow.python.keras.engine.training) is deprecated and will be removed in a future version
Instructions for updating:
```

**Figura 4.6:** Captura de pantalla de los pesos proporcionados por la función `Class_weight_compute_class_weight` de Skicit-learn, en la terminal MovaXterm. Ref [37]. Elaboración propia.

Una vez realizado el balanceo de datos, se procede a aumentar el número de estas muestras para evitar que en nuestra red se produzca el *overfitting* o sobreajuste.

El *overfitting* o sobreajuste se produce cuando el modelo obtenido se ajusta tanto a los ejemplos etiquetados de entrenamiento que no puede realizar las predicciones correctas cuando se introducen datos nuevos en la red porque no generaliza bien.



**Figura 4.7:** Diferencias entre subajuste (underfitting) y sobreajuste (overfitting). Referencia [47].

En general, hay dos formas básicas de evitar el *overfitting*: la primera es la de obtener más datos y la segunda es la regularización [38]. De estas dos, la primera es normalmente la mejor solución.

Es por ello, que se recurre al aumento de datos. Para el aumento de datos, en este trabajo se ha utilizado la técnica *Data augmentation*, que se explicará con más detalle en el próximo punto de esta memoria.

### 2.1.1. Data augmentation

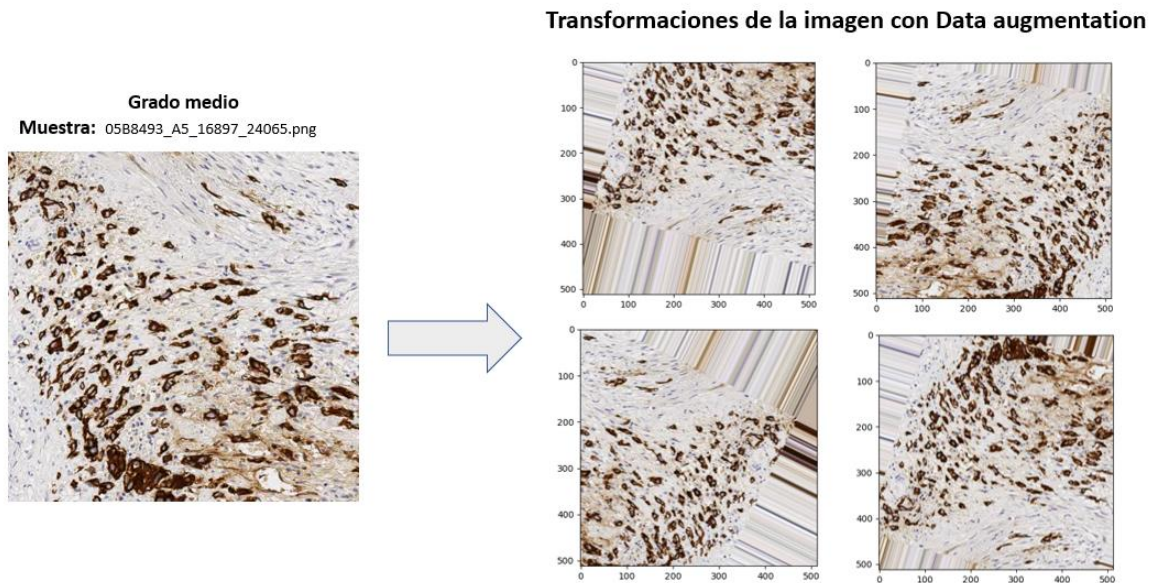
Es una de las técnicas más populares para mitigar el *sobreajuste (overfitting)* en los modelos de redes neuronales y que es altamente recomendable, cuando el dato a tratar es de tipo imagen.

Esta técnica es muy simple y de fácil entendimiento visual, ya que aplica transformaciones sencillas a la imagen, como: rotación, re-escalado, translación, zoom, flips (horizontal y vertical) y perturbación de color.

Estas transformaciones se realizan en tiempo real (durante el entrenamiento) por lo que, no guarda de forma automática las imágenes aumentadas. Las nuevas imágenes toman la clase de la imagen origen, con lo cual, estas transformaciones no pueden ser muy exageradas, para no ocasionar confusión en la red neuronal.

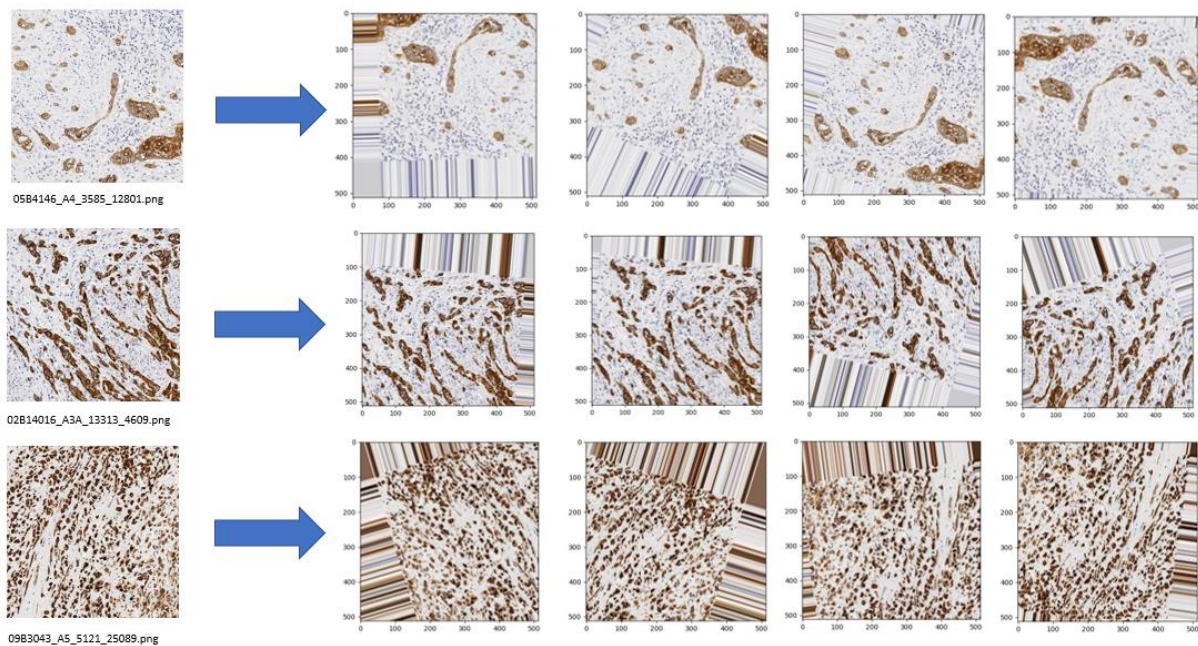
Diseño y desarrollo de software para el análisis de patrones microscópicos de crecimiento tumoral en pacientes con carcinoma vesical.

Tensorflow [36] a través de la biblioteca de redes neuronales de AP Keras, proporciona la capacidad de ajustar modelos utilizando el aumento de datos de imágenes a través de la clase ImageDataGenerator.



**Figura 4.8:** Transformación de un patch obtenido de una WSI (perteneciente a la clase grado medio) con la técnica Data augmentation de tensorflow[36]. Elaboración propia.

Este aumento de datos solo se realizó para la etapa de entrenamiento (partimos con un total de 488 patches de imágenes). La clase DataAugmentation realiza 4 transformaciones por cada patch (como se puede observar en la Figura 4.8). Por lo tanto, el volumen de imágenes con las que entrena nuestra red es de 488 patches x 4 transformaciones = 1952 patches de imágenes aproximadamente.



**Figura 4.9:** Transformación de otros patches de la base de datos aplicando la técnica Data augmentation de tensorflow[36]. Elaboración propia.

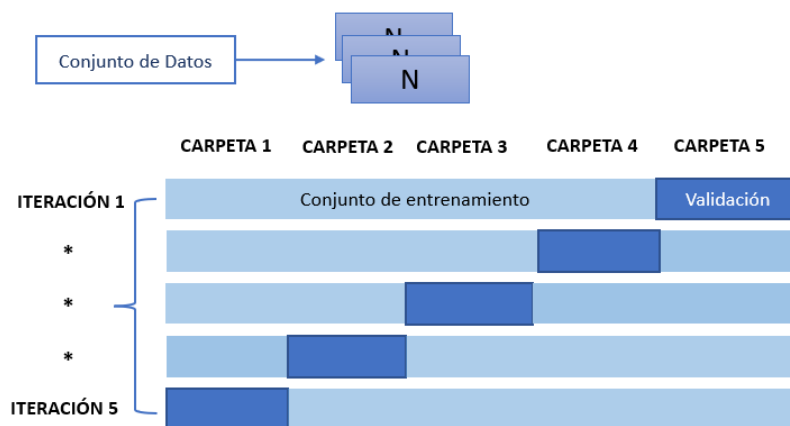
#### 4.1.3. Partición de los datos.

En los algoritmos de clasificación supervisada es importante determinar qué datos son los que se van a utilizar para el entrenamiento y que datos se utilizarán para el testeo de nuestro modelo de red neuronal.

Esto se realiza con la finalidad de que este último conjunto de datos (test) pueda poner a prueba la robustez del modelo empleado ya que, si no realizásemos esta partición previa, habría una alta probabilidad de que nuestro modelo dé resultados inmejorables, pero que estos a su vez sean poco fiables dado que cambiarán a peor cuando se intente predecir con un conjunto de datos jamás vistos por el mismo.

Por esto, en este trabajo se ha realizado una partición previa en donde el 80% de los datos fueron asignados para el conjunto de entrenamiento y el 20% restante se asignaron al conjunto de prueba.

Se utilizaron técnicas de validación cruzada de tipo *K-Fold* para dividir el conjunto de datos de entrenamiento en varios (*K*) subconjuntos de "*training*" y "*validation*". Esto se realizó con el objetivo de encontrar la mejor configuración de hiperparámetros y de poder controlar el overfitting durante el entrenamiento de los modelos.



**Figura 4.10:** Funcionamiento de la técnica K-fold cross validation empleada en este estudio. Elaboración propia.

Partición de datos con K-fold cross validation, k=5 nº observaciones = 607			
nº observaciones totales particionadas	Train	Validation	Test
		391	97
Conjunto de datos	partición n=1:5		
	LowGrade	MediumGrade	HighGrade
Conjunto de entrenamiento:	94	85	211
Conjunto de validación:	24	21	53
Conjunto de test:	29	26	64
nº observaciones parciales	147	132	328
total	607		

**Figura 4.11:** Partición de datos realizada con la técnica K-fold cross validation.

## 4.2 Algoritmos implementados

Antes de tratar detalladamente los algoritmos que en este trabajo se implementan, es recomendable realizar una breve introducción sobre el concepto de red neuronal, así como de sus parámetros e hiperparámetros.

### 2.2.1. Concepto de Red neuronal

Una red neuronal es un método de computación que simula el funcionamiento neuronal humano durante el aprendizaje. Para conseguir emular este funcionamiento se interconectan elementos (neuronas) que reciben información y elaboran una señal que es transmitida hasta generar una respuesta.

Las redes neuronales a diferencia de los métodos estadísticos multivariantes no tienen ninguna condición sobre la distribución de los datos, ya que estos pueden tener una gran variabilidad, ser ambiguos e incluso presentar ruido. Otra de sus características, es que admite relaciones no lineales en la que el valor de una variable puede tener distinto efecto dependiendo de los valores de otras variables[43]. Estos sistemas aprenden a base de una gran cantidad de ejemplos, siendo capaces de predecir patrones y tendencias además de descubrir sutiles relaciones entre los datos. Esto último es lo que las convierte en buenas candidatas para desarrollar algoritmos de computación que sirvan de ayuda al diagnóstico.

### 2.2.2. Estructura de una red neuronal.

Una red neuronal está compuesta por 3 niveles o capas:

Un *primer nivel o capa de entrada*; que es como su nombre lo indica, la entrada del dato o los datos (ya que puede tener más de una entrada), un *nivel o capa de salida*; que es la que nos proporciona los resultados del entrenamiento de la red y un *nivel o capa oculta*; que es en donde se realiza el procesamiento de los datos y que puede estar compuesto por una o varias capas.

En la siguiente figura se muestra la distribución de estas capas en la red neuronal.

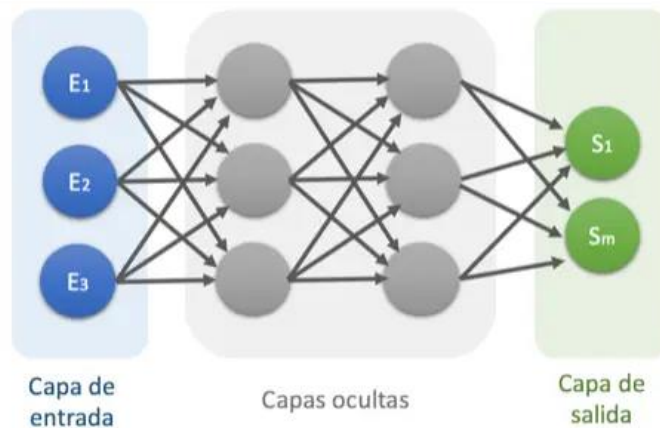


Figura 4.12: Estructura de una Red neuronal. Imagen obtenida de referencia [44].

### 2.2.3. Parámetros e hiperparámetros de una red neuronal

Los parámetros e hiperparámetros son variables que determinarán la estructura de la red o de cómo se entrena dicha red.

Los parámetros se diferencian de los hiperparámetros en que los primeros forman parte del modelo interno de la red, por ejemplo, los pesos de las neuronas. Estos pesos se estiman o aprenden



automáticamente a partir de las muestras de entrenamiento y suelen utilizarse para hacer predicciones en un modelo entrenado que se encuentre en producción.

Los hiperparámetros son externos al modelo de la red y son establecidos por el programador, por ejemplo, la función de activación o el tamaño de lote que se utilizará en el entrenamiento. Una elección adecuada de estos hiperparámetros serán clave para conseguir una buena optimización de la red neuronal.

Es importante añadir que descubrir estos valores no es algo trivial y que dependerá exclusivamente de la experiencia que tenga el programador de la red neuronal, ya que estos resultados se consiguen a través de la experimentación, en otras palabras, mediante ensayos de prueba y error.

Se distinguen 2 tipos de hiperparámetros: Los que están a nivel de estructura y tipología (número de capas, número de neuronas por capas, inicialización de los pesos, etcétera) y los que están a nivel del algoritmo de aprendizaje (*batch size*, *epochs*, *learning rate*, *momentum*, etcétera) [38].

A continuación, se describirán los hiperparámetros *más relevantes* que están a nivel del algoritmo de aprendizaje.

**Batch size:** Esta variable hace referencia al tamaño de los datos particionados (en lotes) de una iteración del entrenamiento. Su función es la de actualizar el gradiente. Su tamaño óptimo dependerá de muchos factores, entre ellos la capacidad de memoria del ordenador que utilicemos para hacer los cálculos.

**Número de épocas (epochs):** Esta variable nos indica la cantidad de veces que los datos de entrenamiento han realizado el proceso de *forward* y *backward propagation* por la red neuronal durante el entrenamiento. Un número alto de épocas puede ocasionar problemas de generalización en el conjunto de los datos de prueba y validación. Del mismo modo, un número bajo de épocas puede llegar a no hacer buenas predicciones cayendo en lo que se conoce como *underfitting* o subajuste (Figura 4.7). El número óptimo de épocas se consigue de manera empírica durante la etapa de validación. No obstante, se pueden utilizar determinados *callbacks* que permiten salvarguardar el modelo en la época.

**Tasa de aprendizaje (Learning rate) y tasa de decaimiento (learning decay):** La tasa de aprendizaje o en inglés (*learning rate*) hace referencia a la dirección y magnitud que tiene un vector de gradiente. Un valor óptimo de este hiperparámetro será muy dependiente del problema a tratar ya que nos indica *el tamaño* de los pasos que da la red durante el proceso de aprendizaje.

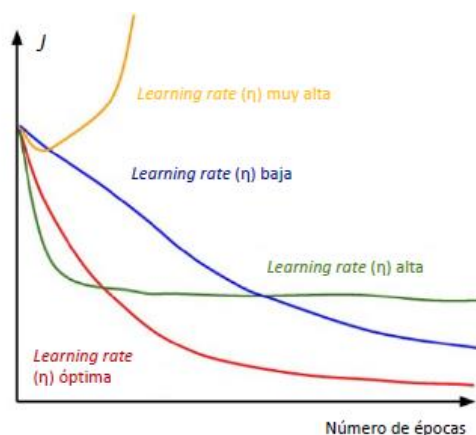


Figura 4.13: Comportamiento del learning rate o tasa de aprendizaje en función del número de épocas.

Un valor muy alto en la tasa de aprendizaje se traduce en que estamos dando pasos muy grandes hasta llegar al número de épocas, lo que puede provocar que el proceso de aprendizaje sea rápido, pero no efectivo dado que nunca llegará al valor mínimo. De manera contraria, una tasa de aprendizaje demasiado pequeña puede provocar que el proceso de aprendizaje sea extremadamente lento.

En líneas generales, podemos decir que la tasa de aprendizaje es aquella que disminuye a medida que el modelo se acerca a una solución. Para conseguir este efecto, se hace uso de la variable *learning decay* o decaimiento en el rango de aprendizaje. Esta variable permite que el aprendizaje avance más rápido al principio con tasas de aprendizajes más grandes y a medida que avanza va ajustando esta tasa para facilitar que converja el proceso de entrenamiento al mínimo de la función de pérdidas.

Un apunte importante, es que su valor depende también del tipo de optimizador que se esté utilizando para entrenar la red.

**Momentum:** es una variable que mejora la tasa de convergencia en las redes neuronales profundas. El momentum se suele utilizar para asistir al optimizador (sgd) en las zonas donde el gradiente es muy pronunciado (normalmente cerca del mínimo local), limitando el salto errático del peso de una iteración a otra [39].

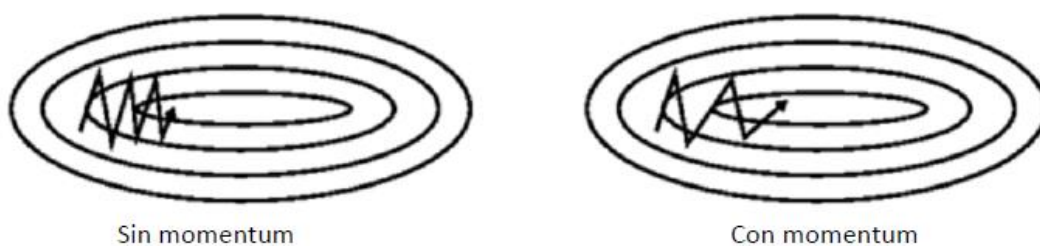


Figura 4.14: Comportamiento del hiperparámetro momentum. Imagen obtenida de referencia [39].

**Funciones de activación:** La función de activación sirve para introducir la no linealidad (propagar hacia adelante la salida de una neurona, incluida la capa de salida) en las capacidades de modelado de la red. En la siguiente figura se muestra la función de activación RELU que es la que ha sido utilizada en este proyecto.

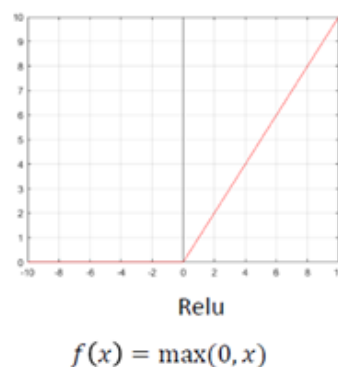


Figura 4.15: Gráfica y ecuación de la función de activación Relu. Imagen obtenida de la referencia [39].

La función de activación RELU suele ser muy común de utilizar ya que converge más rápido que otras funciones de activación. Como desventaja de esta función, se puede decir que es frágil ya que el sistema puede dejar de aprender porque puede hacer que los pesos no se activen[39].

Otra de las funciones de activación que nos podemos encontrar en la capa de salida de la red

neuronal es la *softmax*. Esta función de activación devuelve la distribución de probabilidad de cada una de las clases soportadas en el modelo de tal forma que el sumatorio de todas las probabilidades dan como resultado la unidad  $\sum \sigma(z)_j = 1$ .

Para ello, *softmax* usa el valor exponencial de las evidencias calculadas y, luego, las normaliza de modo que sumen uno, formando una distribución de probabilidad. La probabilidad de pertenencia a la clase  $i$  es:

$$Softmax_i = \frac{e^{evidencia_i}}{\sum_j e^{evidencia_j}}$$

El efecto que se consigue con el uso de exponenciales es que una unidad más de evidencia tiene un efecto multiplicador y una unidad menos tiene el efecto inverso. Teniendo en cuenta esto, una buena predicción tendrá una sola predicción en el vector cercana a 1, mientras que las restantes estarán cerca de 0.

**Inicialización de los pesos:** No es exactamente un hiperparámetro, pero es tan importante como cualesquiera de ellos. Esta inicialización se debe realizar sobre todo cuando se trabaja con redes pre entrenadas con técnicas de *fine-tuning* o *transfer learning* y es recomendable inicializar estos pesos con valores aleatorios y pequeños para que de esta manera se pueda romper con la simetría entre las diferentes neuronas puesto que, si más de una neurona tiene exactamente los mismos pesos quiere decir que en iteraciones subsecuentes no serán capaces de aprender características diferentes de la información que le estamos proporcionando a la red.

#### 2.2.4. Argumentos de compilación del modelo.

**Optimizadores:** Es un argumento muy necesario a la hora de compilar el modelo de nuestra red. La función que realiza es la de optimizar los valores de los parámetros para reducir el error cometido por la red. Esto lo consigue retro propagando (*backward propagation*) el error calculado para así ajustar los parámetros.

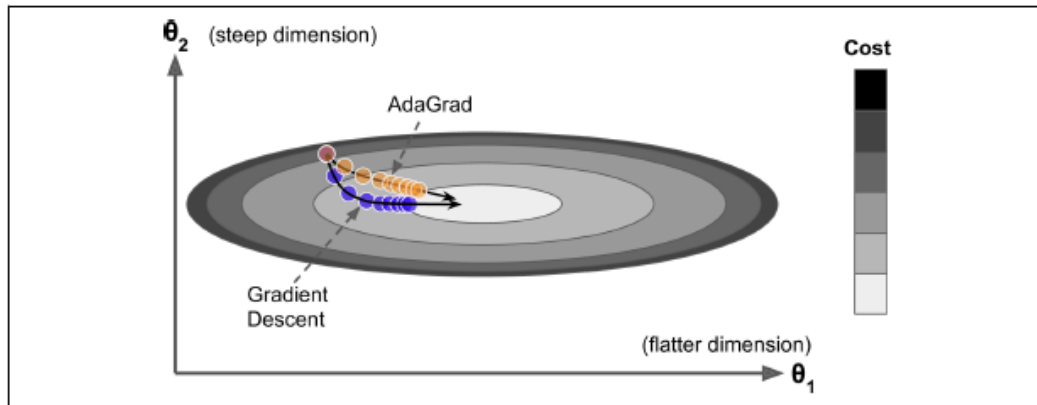
Los optimizadores más comúnmente utilizados son: el gradiente estocástico (*stochastic gradient descent*, sus siglas en inglés: *sgd*), Adadelta, Adam y Adagrad.

En este proyecto hemos utilizado el optimizador Adagrad, ya que tiene tasas de aprendizaje específicas de parámetros que se adaptan en relación con la frecuencia con la que se actualiza un parámetro durante el entrenamiento, es decir, cuantas más actualizaciones recibe un parámetro, más pequeñas son estas actualizaciones [48]. Del mismo modo, estas actualizaciones serán más grandes para parámetros que ocurren con poca frecuencia.

En su regla de actualización, Adagrad modifica la tasa general de aprendizaje  $\eta$  en cada paso de tiempo  $t$  para cada parámetro  $\theta_i$  basado en los gradientes pasados para  $\theta_i$ .

$$\theta_{t+1,i} = \theta_{t,i} - \frac{\eta}{\sqrt{G_{t,i} + \epsilon}} g_{t,i}$$

La principal ventaja de utilizar este optimizador es que elimina la necesidad de ajustar manualmente la tasa de aprendizaje; se suele recomendar un valor predeterminado de 0.01. Como desventaja, se podría apuntar a la acumulación de gradientes al cuadrado en el denominador. Dado que cada término agregado es positivo, la suma acumulada sigue creciendo durante el entrenamiento, lo que hace que la tasa de aprendizaje se reduzca y se vuelva infinitesimalmente pequeña [49].



**Figura 4.16:** Comparativa de los optimizadores Adagrad y el descenso del gradiente: El primero puede corregir su dirección y llegar rápidamente al punto óptimo. Imagen obtenida de referencia [24].

**Función de pérdidas:** este argumento nos sirve para guiar el proceso de entrenamiento de la red y para cuantificar lo cercana que está una determinada red neuronal de su ideal mientras está en el proceso de entrenamiento. La función de pérdidas que utilizaremos en este trabajo es la 'categorical crossentropy' ya que nuestra salida debe de estar en formato categórico, es decir, la variable de salida debe de tomar un valor entre los 3 posibles.

En tareas de clasificación es importante tener en cuenta que además de seleccionar la función de pérdidas como categórica, hemos de poner en la última capa del clasificador, el mismo número de neuronas que de clases (en nuestro caso 3) acompañadas por la función de activación *softmax*.

## 4.3 Algoritmos de clasificación supervisada para las imágenes histopatológicas de los grados del carcinoma urotelial

### 4.3.1. Clasificación supervisada con redes neuronales convolucionales.

En visión por computador o *computer vision*, cuando se trabaja con imágenes es preferible recurrir a las redes neuronales convolucionales (*Convolutional Neural Network*, en inglés sus acrónimos son CNN o *ConvNets*). Se diferencian de las redes neuronales vistas en el punto anterior en que hacen la suposición explícita de que las entradas son imágenes, por lo que permite al programador codificar ciertas propiedades en la arquitectura para reconocer elementos concretos que contengan las imágenes.

Dos propiedades interesantes que tienen las ConvNets son:

- Los patrones que estas redes aprenden son invariantes en translación, esto quiere decir, que después de aprender un determinado patrón, una convnet es capaz de reconocer este patrón en cualquier parte de la imagen.
- Estas redes aprenden jerarquías espaciales de los patrones; esto es, que una primera capa de convolución aprenderá pequeños patrones locales, mientras que una segunda capa de convolución aprenderá patrones más grandes hechos de las características de las primeras capas y así sucesivamente. Esto es lo que las hace muy eficientes ya que aprenden el complejo y abstracto concepto visual [67][32].

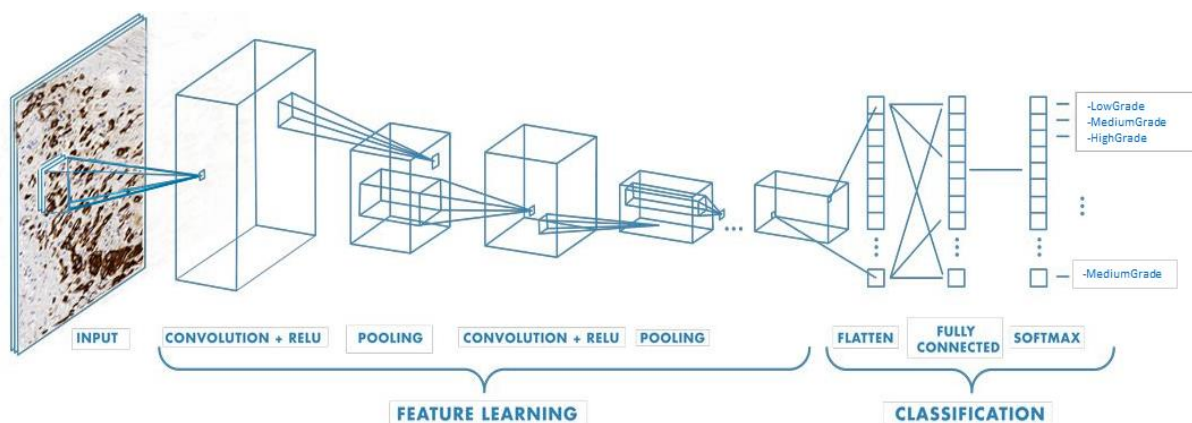
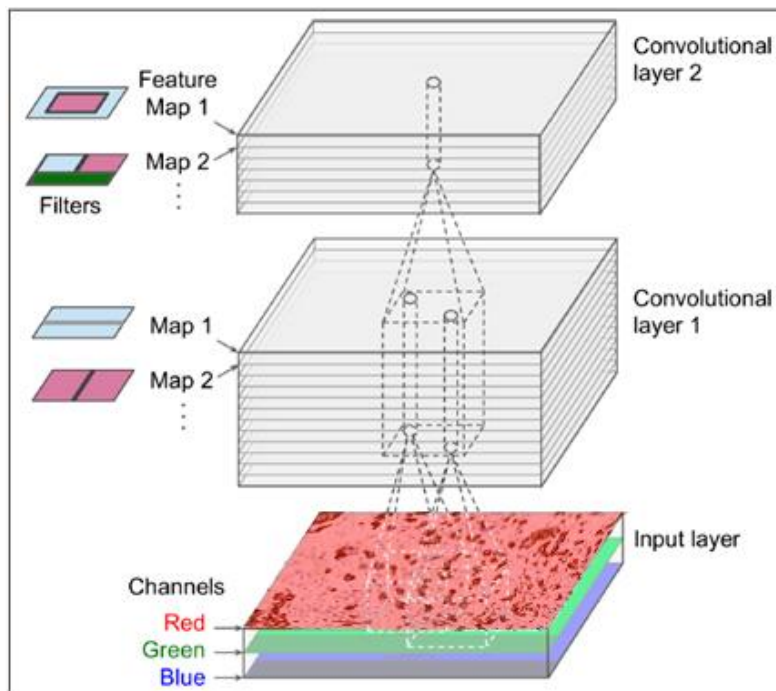


Figura 4.17: Estructura de una CNN. Elaboración propia.

Los *componentes básicos* de una red neuronal convolucional son dos capas que pueden expresarse como grupos de neuronas especializadas en dos operaciones: convolución y *pooling*.

4.3.1.1. Capa de convolución: con la capa convolucional lo que se quiere conseguir es mapear la imagen de entrada a través de los tensores 3D con los que operan y que se conocen como mapas de características (*feature maps, en inglés*). Estos tensores se corresponden a los 2 ejes espaciales de altura y ancho, además de un eje de canal (*channels*) llamado profundidad (*depth*).

En resumen, una capa convolucional aplica simultáneamente múltiples filtros entrenables a sus entradas, lo que la hace capaz de detectar múltiples características en cualquier parte de sus entradas [24].



**Figura 4.18:** Capas convolucionales con múltiples mapas de características en una imagen con 3 canales de colores (RGB). Imagen modificada de la referencia [24].

La ecuación (4.1), resume el cálculo matemático que realiza la salida de una capa convolucional:

$$Z_{i,j,k} = b_k + \sum_{u=0}^{f_h-1} \sum_{v=0}^{f_w-1} \sum_{k'=0}^{f_{n'}-1} x_{i',j',k'} \times w_{u,v,k',k} \quad \text{con} \quad \begin{cases} i' = \times s_h + u \\ j' = \times s_w + v \end{cases} \quad (4.1)$$

donde;

$Z_{i,j,k}$  es la salida de la neurona localizada en la fila  $i$  y la columna  $j$  en el mapa de características  $k$  de la capa convolucional (Capa 1).

$s_h$  y  $s_w$  son el número de pasos o zancadas (*strides*) horizontal y vertical,  $f_h$  y  $f_w$  son el alto y el ancho del campo receptivo y  $f_{n'}$  es el número de características en el mapa de la capa previa ( $\ell-1$ ).

$x_{i',j',k'}$  es la salida de la neurona localizada en la última capa ( $\ell-1$ ), fila  $i'$ , columna  $j'$ , mapa de características  $K'$  (o canal  $k'$  si la capa previa es la capa de entrada).

$b_k$  es el valor del término *bias* del mapa de características  $K$  en la capa  $\ell$  y,

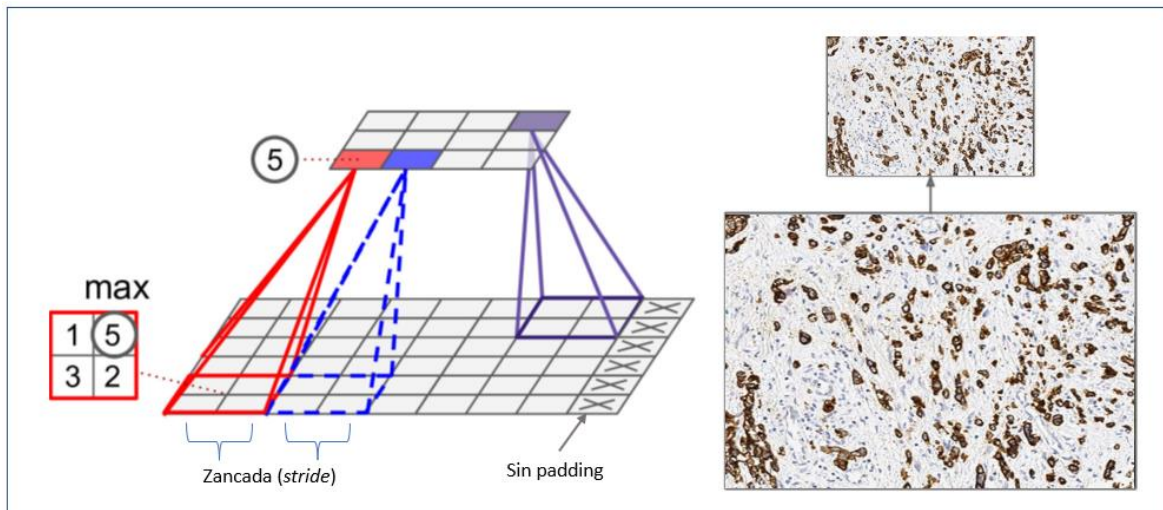
$w_{u,v,k',k}$  es la conexión de los pesos entre cualquier neurona en un mapa de características  $K$  de la capa  $\ell$  y su entrada ubicada en la fila  $u$  y columna  $v$  (relativo al campo receptivo de neuronas) y mapa de características  $k'$ .

**4.3.1.2. Capas de pooling:** las capas de *pooling* hacen una simplificación de la información recogida por la capa convolucional y crean una versión condensada de la información contenida en esta capa.

Su objetivo es sub-sampear la imagen de entrada para de esta manera reducir la carga computacional, el uso de la memoria y el número de parámetros (de este modo evita el riesgo de *overfitting*).

En la Figura 4.19 se utiliza sobre la imagen de entrada un *kernel pooling* (núcleo)  $2 \times 2$ , con un *stride* de 2, sin *padding*. Como se puede observar solo el máximo valor del píxel (dentro del campo

receptivo) forma la siguiente capa, mientras que las otras entradas se eliminan.



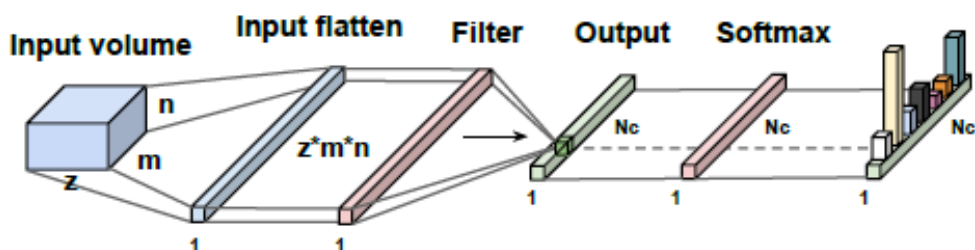
**Figura 4.19:** Representación de una capa de Max pooling con 2x2 de pooling kernel, zancada, no padding. Figura modificada de referencia[24].

Otra de las razones por las que se van a utilizar este tipo de red neuronal en este trabajo, es por la diferencia que existe entre las capas convolucionales de las CNN y las capas densamente conectadas (NN). En las primeras aprenden patrones locales dentro de la imagen en pequeñas ventanas de dos dimensiones mientras que, en las segundas aprenden patrones globales en su espacio global de entrada.

Además de estas capas (*layers*) que forman la estructura principal de una CNN, haremos uso de otras capas como: densa, max pooling2D, flatten y dropout.

#### 4.3.1.3. Capa densa (Layer dense or fully connected FC):

La capa densa es conocida también como capa densamente conectada, ya que todas sus neuronas están conectadas a las de la siguiente capa. Cada nodo de entrada está conectado a cada nodo de salida. La última capa densamente conectada es una capa *softmax* (cuando se tiene más de una neurona de salida, como ocurre en nuestro caso), con tantas neuronas como el número de clase que se tenga, para nuestro caso de estudio, un total de 3.



**Figura 4.20:** Estructura de una capa fully connected o capa densamente conectada. Imagen obtenida de la referencia [39].

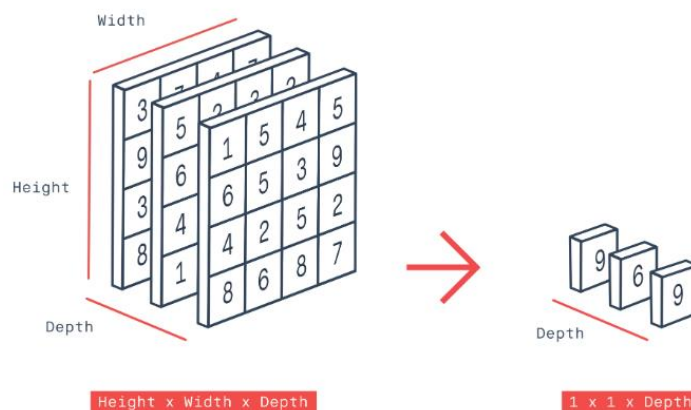
Los parámetros de entrada que normalmente recibe esta capa son: el número de neuronas, la función de activación y el nombre como identificador de la capa. El resto de los parámetros se pueden consultar en la referencia [45].

#### 4.3.1.4. Capa GlobalMaxPooling2D:

El funcionamiento de esta capa es similar a la capa de *pooling* con la diferencia de que el tamaño de la agrupación (el factor de agrupación horizontal por el factor de agrupación vertical) es el tamaño de toda la entrada del bloque, es decir, calcula un solo valor máximo para cada uno de los canales de entrada.

La salida que se obtiene de esta capa es la de un tensor unidimensional con los máximos valores de los canales de entrada: alto (*height*), ancho (*width*) y profundidad (*depth*).

Estas capas se suelen utilizar como alternativa a la capa *flatten* en donde lo que se quiere conseguir es que la salida sea unidimensional (array plano, como se puede observar en la Figura 4.20).

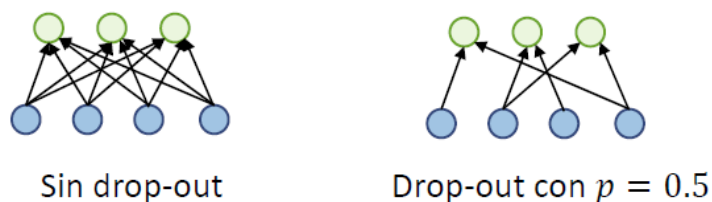


**Figura 4.21:** Funcionamiento de la capa MaxGlobalpooling2D. Imagen obtenida de la referencia [50].

#### 4.3.1.5. Capa dropout:

La función que realiza es la de desconectar aleatoriamente entradas desde una capa a la siguiente con una probabilidad  $p$ : probabilidad de mantener el enlace o de desconectarlo (esto dependerá de la implementación). Esta capa solo se utiliza en la etapa de *training*.

Para un *batch*, los enlaces se desactivan para las etapas de *forward and backward propagation*, luego se reactivan y se vuelven a desactivar aleatoriamente para el siguiente *batch*.



**Figura 4.22:** Comportamiento de capas densas sin y con dropout. Imagen obtenida de la referencia [39].

Desconectar conexiones aleatoriamente asegura que no haya ningún nodo que sea siempre responsable de “activarse” cuando se presente un patrón dado. Se asegura que haya nodos múltiples y redundantes que se activarán cuando se presenten entradas similares, lo que a su vez ayuda a nuestro modelo a generalizar [39].

En el **capítulo 5** se justificará brevemente su uso y los argumentos utilizados en estas capas. Para obtener una información más detallada acerca de su funcionamiento, se puede consultar en la referencia [45].

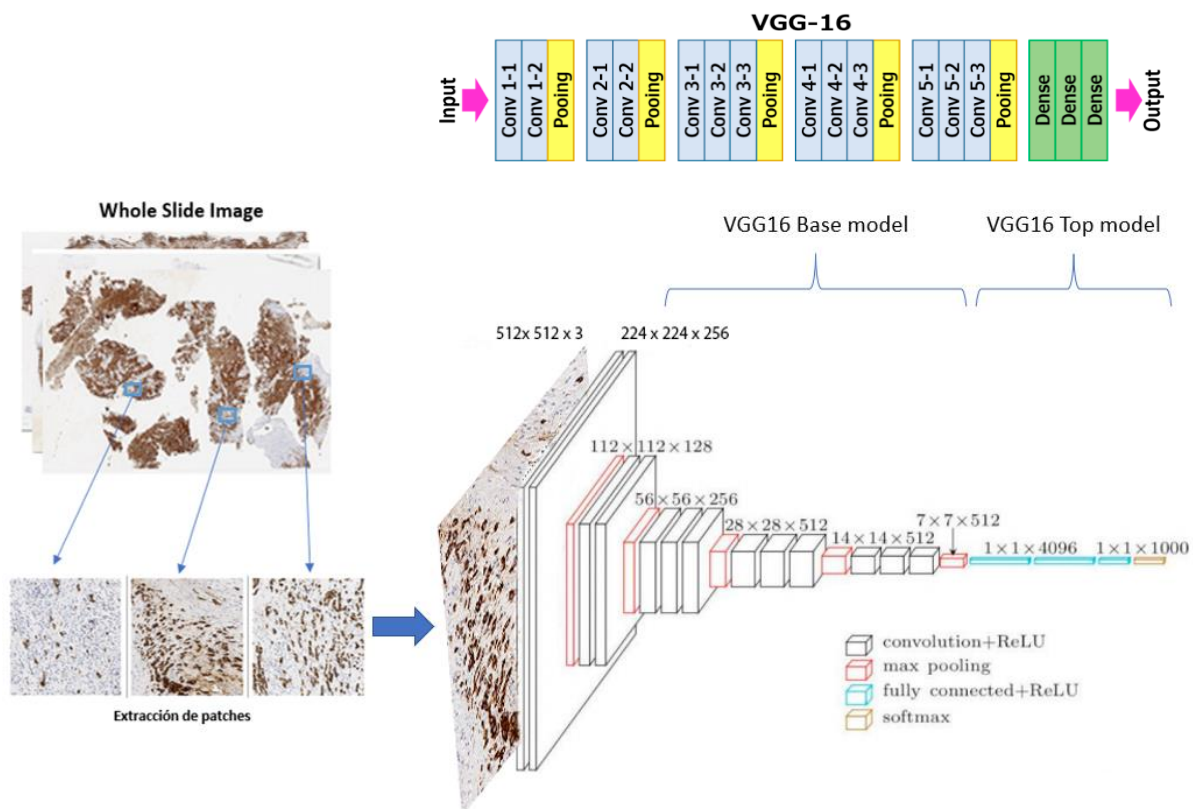


#### 4.3.2. Transferencia del conocimiento o *Transfer Learning*.

Con la transferencia de conocimiento o *transfer learning* lo que se pretende es seleccionar un modelo de red neuronal predefinida, cuyos pesos fueron previamente entrenados con otro tipo de imágenes en tareas similares de clasificación y de la cual se han obtenido buenos resultados. Con esto se evita, el tener que definir su arquitectura desde cero (*from scratch*) así como también realizar un ajuste fino (*fine tuning*) para que los pesos de la red no se inicien de forma aleatoria.

Para nuestro caso de estudio se ha seleccionado el modelo VGG-16, que es una arquitectura de red neuronal convolucional (CNN) y que es ampliamente utilizada por el conjunto de datos ImageNet\*.

Esta arquitectura fue desarrollada e introducida por Karen Simonyan y Andrew Zisserman de la Universidad de Oxford, en el año 2014 [40] y sus siglas 'VGG' se corresponden a que es la abreviatura del grupo de investigación "Visual Geometry Group" del que forman parte sus autores. El número '16' que acompaña al nombre de esta arquitectura implica que está conformada por 16 capas convolucionales (con peso) distribuidas en 5 bloques (13 convolucionales + 5 pooling). Ver Figura 4.23.



**Figura 4.23:** Arquitectura VGG16 diseñada para llevar a cabo tareas de clasificación. Figura modificada de referencia [39].

A este modelo se le aplicará la técnica complementaria de *transfer learning*: *Fine tuning* (4.3.3) que se explicará con mejor detalle en el siguiente apartado.

\* Imagenet es un conjunto de datos de más de 14 millones de imágenes pertenecientes a 1000 clases que están disponibles de manera pública para su descarga y se pueden usar para crear potentes modelos de visión por computador con muy pocos datos.

ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224 × 224 RGB image)					
conv3-64	conv3-64 <b>LRN</b>	conv3-64 <b>conv3-64</b>	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 <b>conv3-128</b>	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 <b>conv1-256</b>	conv3-256 conv3-256 <b>conv3-256</b>	conv3-256 conv3-256 conv3-256 <b>conv3-256</b>
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 <b>conv1-512</b>	conv3-512 conv3-512 <b>conv3-512</b>	conv3-512 conv3-512 conv3-512 <b>conv3-512</b>
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 <b>conv1-512</b>	conv3-512 conv3-512 <b>conv3-512</b>	conv3-512 conv3-512 conv3-512 <b>conv3-512</b>
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

Figura 4.24: Configuración de una CNN modelo VGG-16. Referencia [42].

### 4.3.3. Fine Tuning

El fine-tuning es una técnica complementaria, ampliamente utilizada en la reutilización de modelos (*transfer learning*). Consiste en hacer un ajuste más fino de las representaciones más abstractas que forman parte del modelo que se está reutilizando como base, esto se consigue congelando los valores de los pesos (coeficientes de las primeras capas convolucionales), así como reentrenar únicamente los pesos de las últimas capas para incorporar en la arquitectura la información relativa a las imágenes específicas del problema bajo estudio.

Para entender mejor su funcionamiento hay que tener en cuenta el nivel de generalización y la posición que tengan dichas capas (*de convolución específicas*) en el modelo, ya que dependiendo de esto último nos interesará modificarlas en mayor medida o no.

Aplicando esta definición a nuestro caso de estudio, podríamos asumir que las capas iniciales extraerán los bordes, texturas, colores, etc., de los patrones existentes en nuestro conjunto de datos. Las capas intermedias extraerán conceptos más abstractos como pueden ser la forma. Y las capas que están casi al final, conceptos más abstractos que nos van a permitir construir un concepto del patrón presentado.

Cuando se hace una modificación (*tuneo*) a las capas iniciales e intermedias del modelo reutilizado, a este fine tuning se le conoce como Deep tuning (afinación profunda) y cuando la modificación es en las capas finales se le conoce como Shallow tuning (afinación superficial).

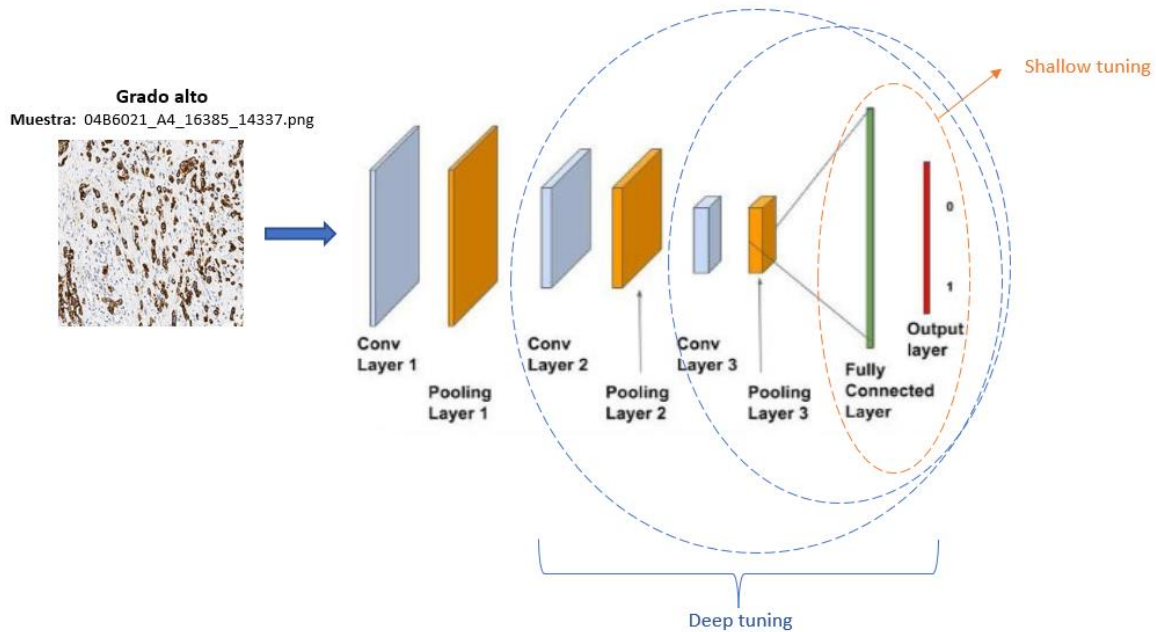


Figura 4.25: Formas de Fine tuning, según la capa que se modifique. Elaboración propia.

Es importante, tener en cuenta el tipo de *fine tuning* que queremos aplicar ya que, si el nuevo conjunto de datos difiere mucho del conjunto de datos en el que se entrenó el modelo original, se deben usar solo las primeras capas del modelo, para de esta manera poder realizar la extracción de características (*en lugar de usar toda la base convolucional*).

#### 4.3.4. Modelo From scratch

Desde cero o *From scratch*, hace referencia a que es el programador el que construye el modelo de red neuronal, esto es, definiendo las capas convolucionales que crea necesarias (tanto en la parte del *base model*, como en el *top model*), con la finalidad de entrenar sus datos desde cero.

Se suele recurrir a este modelo, cuando la base de datos con la que se trabaja es pequeña y se necesita implementar una arquitectura muy sencilla, en donde los hiperparámetros a optimizar sean muy pocos, como ocurre en nuestro caso de estudio. Esto a su vez hace que computacionalmente sea más eficiente.

# Experimentación y Resultados

## ÍNDICE DE CONTENIDO:

<b>EXPERIMENTACIÓN Y RESULTADOS .....</b>	<b>51</b>
5.1 Experimentos realizados .....	52
5.2 Métricas de evaluación .....	54
5.3 Resultados obtenidos .....	56

## 5.1 Experimentos realizados

En este capítulo, se exponen los valores utilizados en los parámetros e hiperparámetros con los que se han obtenido mejores resultados para cada uno de los modelos desarrollados, con el objetivo de llevar a cabo la clasificación de los grados de tumor budding (músculo no invasivo) en las muestras histopatológicas proporcionadas.

Para conseguir estos valores, es importante tener en cuenta que la autora de este proyecto llevó a cabo una rigurosa exploración en busca de una combinación óptima de éstos.

### 5.1.1. Modelos empleados en transferencia del conocimiento

Como se explicó en el capítulo anterior, la transferencia del conocimiento o *transfer learning* además de permitirnos entrenar nuestros datos en una red que ha sido previamente entrenada, nos permite optimizar el tiempo. Esto último es muy conveniente cuando se dispone de un tiempo limitado para la elaboración de un trabajo académico de estas características.

Como se explicó en el punto 4.3.2 para entrenar nuestros datos se ha empleado el modelo VGG16.

El procedimiento desarrollado para entrenar esta arquitectura ha sido el siguiente:

- Las 5 particiones de los datos generadas mediante código de Matlab y con extensión “.csv”, se han cargado en el Dataframe a través de la función `pd.read_csv` de la biblioteca pandas.
- Para el preprocesado previo de las imágenes con las que trabajaremos, se ha hecho uso de la clase `ImageDataGenerator` de la biblioteca Keras. Esta clase nos permite: cargar los dataframes, crear y normalizar los objetos `train_generator` y `val_generator`. Estos objetos son los que contendrán nuestras imágenes que, a su vez representan nuestras variables de entrada X e Y para el entrenamiento de la red.

Otra funcionalidad importante que nos permite esta clase es el aumento de datos, más comúnmente conocido como Data Augmentation. En los experimentos se realizó este aumento sobre el objeto `train_generator`, en donde, el valor de los parámetros utilizados para la generación de estos durante el proceso de entrenamiento fue:

```
width_shift_range=0.25, rotation_range=30, height_shift_range=0.25,  
horizontal_flip=True, fill_mode='nearest' y vertical_flip=True.
```

El parámetro “zoom” fue descartado, por introducir variaciones en el tamaño del patrón a buscar en las muestras.

- Para el balanceo de nuestros datos se ha empleado la función `class_weight` de la biblioteca `sckicit-learn`, explicada en el punto 4.1.2.
- Se selecciona como optimizador: `Adagrad` <sup>[2.2.4]</sup>, con `learning rate` (`lr = 0.005`), `batch_size = 16` (por las pocas muestras de la que se disponen) y como argumentos de compilación: para las pérdidas (`loss = 'categorical_crossentropy'`) por tratarse de una clasificación multiclase y para las métricas se utiliza el `'accuracy'`.
- Dado que el modelo empleado dispone de 5 bloques convolucionales, lo que se ha hecho es lo siguiente:
  - ✓ Entrenar los datos con los pesos de las neuronas establecidos por la arquitectura

(weights= 'imagenet') con todos los bloques congelados de forma inicial. Este paso se ha realizado para todas las particiones de nuestros datos.

- ✓ Se procede a descongelar de forma progresiva cada bloque convolucional, empezando por la última capa. A partir del reentrenamiento del tercer bloque convolucional, se empiezan a notar mejores resultados en las métricas.
- En la etapa clasificadora se ha optado por una capa GlobalMaxpooling2D<sup>[4.3.1.4]</sup>.
- Y para las predicciones se ha utilizado una capa densa compuesta por 3 neuronas (que son el número de clases que queremos clasificar) con activación softmax.
- El número de épocas utilizado para el entrenamiento de esta red fue de 50.

#### 5.1.2. Modelo From Scratch

Las capas empleadas para la extracción de características en el diseño de esta red han sido las siguientes:

- El primer bloque está formado por 3 capas: dos capas convolucionales con 32 y 64 neuronas respectivamente con función de activación relu. Como tercera capa se optó por una Maxpooling2D con pool\_size =2x2.
- El segundo bloque, está formado por 5 capas: 3 convolucionales con 128, 256 y 256 neuronas, con función de activación relu. Entre las dos capas convolucionales (de 256 y 256 neuronas) se opta por poner una capa dropout<sup>[4.3.1.5]</sup>. Y como capa final de este bloque se ha utilizado una capa MaxPooling2D con pool\_size=2x2.
- Como última capa del base model se utiliza una capa convolucional con 512 neuronas y función de activación relu.
- Para todas las capas convolucionales se utilizó un kernel de tamaño 3x3, inicializado con una semilla seed=2. La inicialización del kernel hará que no se produzcan los mismos valores aleatorios en varias llamadas y a su vez producirá la misma secuencia cuando se construyan con el mismo valor de semilla inicial.
- Para el *top model* o la etapa clasificadora se utilizó la capa GlobalMaxpooling2D<sup>[4.3.1.4]</sup>.

Y para las predicciones se ha utilizado una capa densa compuesta por 3 neuronas (que son el número de clases que queremos clasificar) con activación softmax.

Para el entrenamiento de esta red, se empleó como optimizador el adagrad<sup>[2.2.4]</sup> con un Lr= 0.005, Bach\_size = 2, número de épocas *epochs*= 100 y como argumentos de compilación se mantuvieron los mismos que se emplearon en el modelo anterior.

## 5.2 Métricas de evaluación

Una vez finalizados todos los entrenamientos necesarios para los modelos desarrollados, se procede a analizar el rendimiento de cada uno de ellos desde la perspectiva cuantitativa.

### 5.2.1 Análisis cuantitativo:

- Matriz de confusión:

La matriz de confusión es una herramienta muy útil para valorar cómo de bueno es nuestro modelo de clasificación implementado, ya que sirve para mostrar de forma explícita cuando una clase es confundida con otra, esto a su vez, nos permitirá obtener distintos tipos de error de manera independiente.

		Valores predictivos		
		LowGrade	MediumGrade	HighGrade
Valores actuales	LowGrade	VP (C1)	(C2)	(C3)
	MediumGrade	(C4)	VP (C5)	(C6)
	HighGrade	(C7)	(C8)	VP (C9)

Figura 5.1: Representación gráfica e interpretación de la matriz de confusión. Elaboración propia.

De la matriz de confusión mostrada en la Figura 5., los valores más importantes son los que se encuentran en la diagonal ya que serán los verdaderos positivos (VP) de nuestro clasificador.

El resto de los valores se han obtenido de la siguiente manera:

- ✓ Para la clase LowGrade:

Los falsos negativos (FN) = C2 + C3; Falsos positivos (FP) = C4 + C7 y; los Verdaderos negativos (VN) = C5 + C6 + C8 + C9.

- ✓ Para la clase MediumGrade:

Los falsos negativos (FN) = C4 + C6; Falsos positivos (FP) = C2 + C8 y; los Verdaderos negativos (VN) = C1 + C3 + C7 + C9.

- ✓ Y para la clase HighGrade:

Los falsos negativos (FN) = C7 + C8; Falsos positivos (FP) = C3 + C6 y; los Verdaderos negativos (VN) = C1 + C2 + C4 + C5.

La matriz de confusión ofrece mucha información, de fácil comprensión, pero puede que a veces se prefiera tener métricas más concisas, como lo son: la exactitud, la precisión, la sensibilidad y la especificidad.

- Exactitud (*accuracy*)

La exactitud representa el porcentaje de predicciones correctas frente al total, por tanto, es el cociente entre los casos bien clasificados por el modelo (verdaderos positivos y verdaderos

negativos, es decir, los valores en la diagonal de la matriz de confusión), y la suma de todos los casos.

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

- **Precisión:**

La precisión es una métrica que permite medir la calidad del modelo de nuestra CNN en tareas de clasificación ya que con ella se obtiene el porcentaje de los aciertos que tiene la misma. Para calcular su porcentaje, hay que emplear la siguiente fórmula:

$$\text{Precisión} = \frac{VP}{VP + FP} \times 100 \quad (5.2)$$

Donde se puede decir que, la precisión es el resultado del número total de verdaderos positivos dividido por la suma de los verdaderos positivos y los falsos positivos.

- **Sensibilidad:**

La sensibilidad (o *recall*) representa la tasa de verdaderos positivos (True Positive Rate) o TP. Es la proporción entre los casos positivos bien clasificados por el modelo, respecto al total de positivos. Su fórmula es la siguiente:

$$\text{Sensibilidad} = \frac{VP}{VP + FN} \quad (5.3)$$

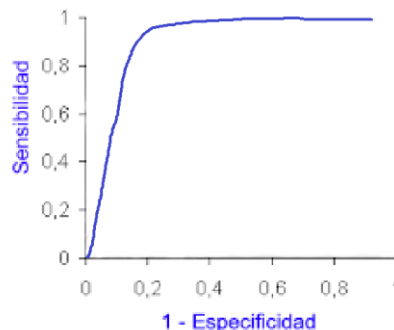
- **Especificidad:**

La especificidad, por su parte, es la tasa de verdaderos negativos, ("true negative rate") o TN. Es la proporción entre los casos negativos bien clasificados por el modelo, respecto al total de negativos. Su fórmula es la siguiente:

$$\text{Especificidad} = \frac{VN}{FP + VN} \quad (5.4)$$

- **Área bajo la curva o en inglés área under curve (AUC):**

Esta métrica nos permite medir el área bajo la curva ROC (Receiver Operating Characteristic). La curva ROC es una representación gráfica de la relación que tiene la sensibilidad frente a la especificidad y cuanto mayor sea el área bajo la curva, mayor será la capacidad discriminadora de nuestro clasificador.



**Figura 5.2:** Gráfica de la curva ROC. Elaboración propia.

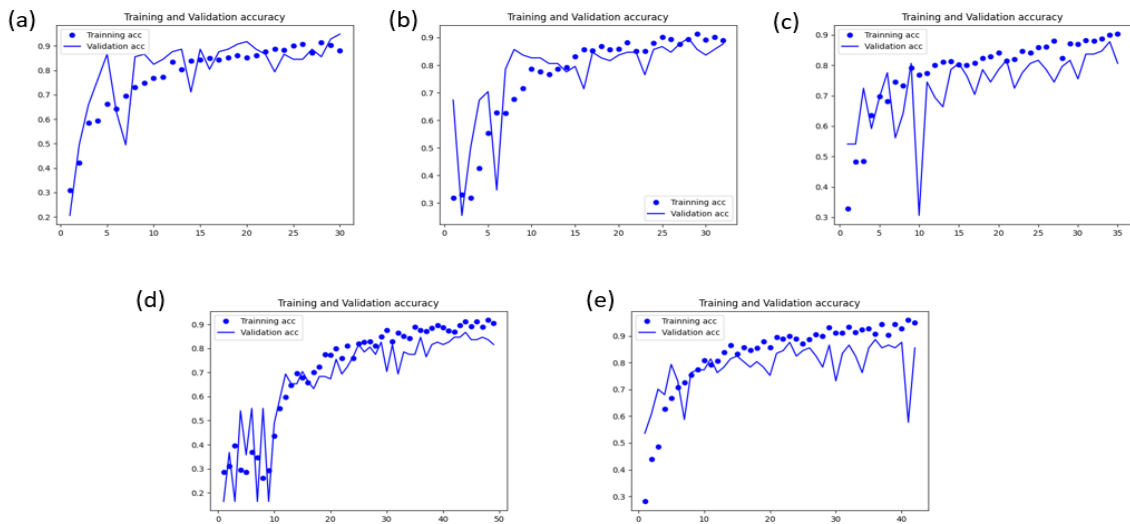


## 5.3 Resultados obtenidos

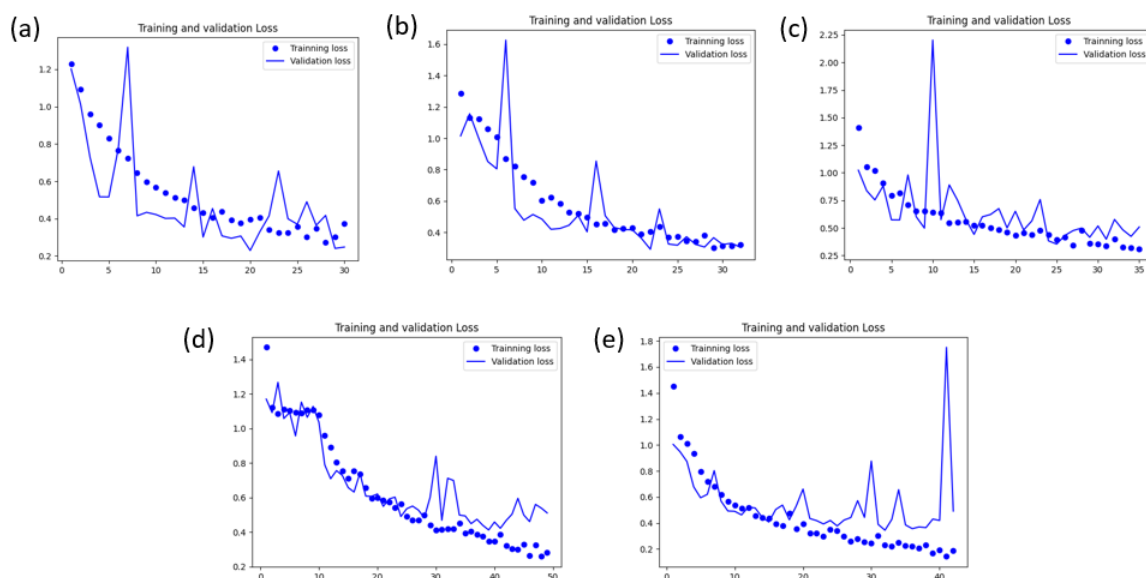
A continuación, se muestran los resultados obtenidos, durante la etapa de validación y la de test, de los modelos desarrollados para la clasificación de los grados de cáncer de vejiga no músculo invasivo.

### 5.3.1 Curvas de aprendizaje obtenidas en la validación cruzada.

#### 5.3.1.1 Top model: 'GMP' transfer Learning



**Figura 5.3:** Representación de las curvas de aprendizaje de la exactitud durante el entrenamiento para el modelo VGG16 utilizando transfer learning con la 5 capa convolucional reentrenada y top model 'GMP'. Valor de la exactitud (Acc) en cada época: (a) Kfold1, (b) Kfold2, (c) Kfold3, (d) Kfold4 y (e) Kfold5.



**Figura 5.4:** Representación de las curvas de aprendizaje de las pérdidas durante el entrenamiento para el modelo VGG16 utilizando transfer learning con la 5 capa convolucional reentrenada y top model 'GMP'. Valor de la exactitud (Acc) en cada época: (a) Kfold1, (b) Kfold2, (c) Kfold3, (d) Kfold4 y (e) Kfold5.

### 5.3.1.2 Discusión sobre los resultados de las curvas de aprendizaje obtenidas en la validación cruzada para el modelo TL con top model GMP.

En las figuras: Figura 5.3 y Figura 5.4 se muestran las curvas de aprendizaje de la exactitud (*accuracy*) y las pérdidas (*loss*) obtenidas de la validación cruzada. Estas curvas se utilizan para monitorear el proceso de aprendizaje que tiene nuestro modelo de red neuronal con el conjunto de datos utilizados: entrenamiento (*train*) y validación (*validation*).

La curva de aprendizaje de *train* (calculada a partir del conjunto de datos de entrenamiento) da una idea de cómo de bien el modelo está aprendiendo mientras que, la curva de aprendizaje de validación (calculada a partir del conjunto de datos de validación) da una idea de qué tan bien está generalizando el modelo.

En el eje y se muestra el valor del *acc* del modelo (su rango va de 0 a 1) y el eje x se corresponde a las épocas utilizadas para el entrenamiento que fue de 50 aunque, en algunas de ellas no se aprecie este valor de forma fija por haber hecho uso del parámetro *early stopping* durante el entrenamiento del modelo. Este parámetro lo que hace es detener el monitoreo de las pérdidas cuando detecta que el modelo ha dejado de aprender o tiende a crear *overfitting* (en un número determinado de épocas fijado con otro parámetro; "*patience*", en nuestro caso su valor fue de 10, esto es que, si en 10 épocas consecutivas el aprendizaje no mejora o hay *overfitting*, detiene el entrenamiento) es por lo que en las gráficas presentadas su valor en el eje varía.

Dicho esto, la interpretación que se puede obtener de estas gráficas (tanto de *training* y *validation* Figura 5.3 como de *training* y *loss* Figura 5.4) es la siguiente:

- (a) En la época  $\approx 27$  se obtiene el valor mínimo de las pérdidas ( $loss \approx 0,38$ ) siendo el valor máximo de la exactitud en la validación de  $acc \approx 0,85$ .
- (b) En la época  $\approx 28$  se obtiene el valor mínimo de las pérdidas ( $loss \approx 0,40$ ) siendo el valor máximo de la exactitud en la validación de  $acc \approx 0,86$ .
- (c) En la época  $\approx 21$  se obtiene el valor mínimo de las pérdidas ( $loss \approx 0,50$ ) siendo el valor máximo de la exactitud en la validación de  $acc \approx 0,81$ .
- (d) En la época  $\approx 26$  se obtiene el valor mínimo de las pérdidas ( $loss \approx 0,44$ ) siendo el valor máximo de la exactitud en la validación de  $acc \approx 0,78$ .
- (e) En la época  $\approx 15$  se obtiene el valor mínimo de las pérdidas ( $loss \approx 0,44$ ) siendo el valor máximo de la exactitud en la validación de  $acc \approx 0,82$ . Para la obtención de esta gráfica se aumentó el valor de "*patience*" a 40, con la intención de ver si el modelo pudiera mostrar mejoras, pero como se puede observar en la gráfica a partir de este valor el modelo tiende a presentar *overfitting*.

Resultados cuantitativos de la etapa de validación

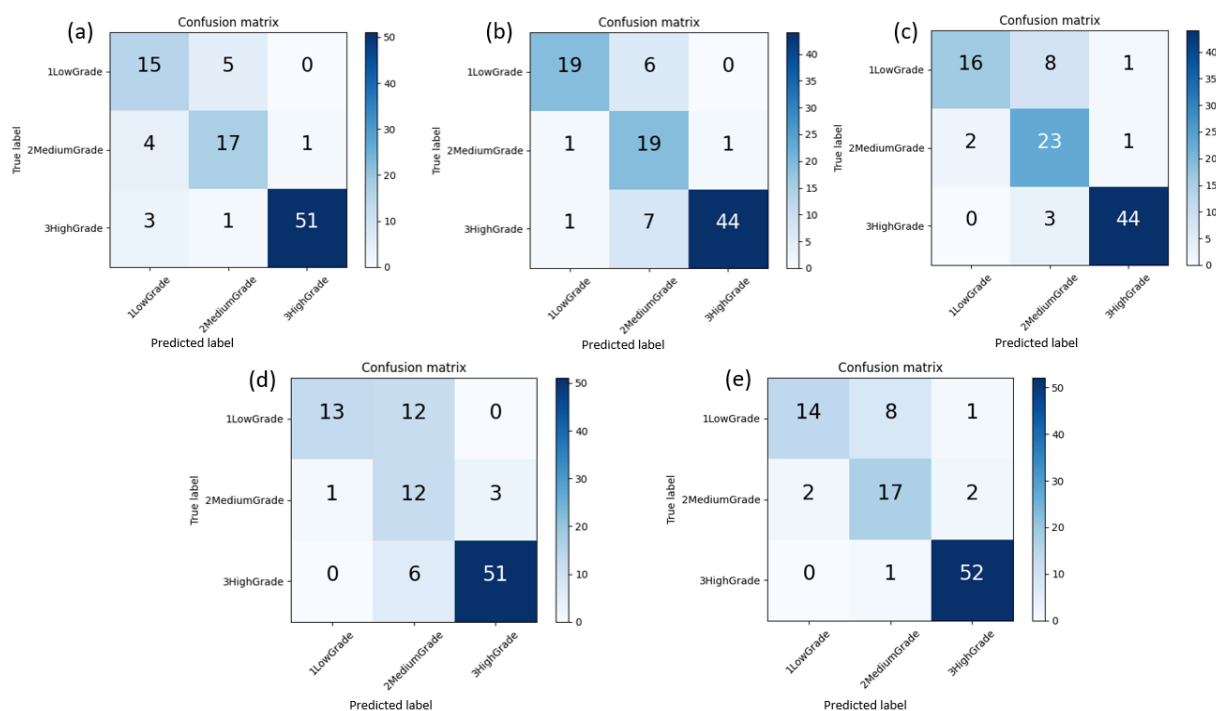


Figura 5.5: Matrices de confusión obtenidas de la etapa de validación para el modelo VGG16 utilizando transfer learning con la 5 capa convolucional reentrenada y top model ‘GMP’: (a) Kfold1, (b) Kfold2, (c) Kfold3, (d) Kfold4 y (e) Kfold5.

K-PARTICION	Kfold1	Kfold2	Kfold3	Kfold4	Kfold5
ACC_TOTAL_MODELO	0,8557	0,8367	0,8469	0,7755	0,8557

ACC_PROMEDIO	0,83
--------------	------

Tabla 5.1: Valores de la exactitud de las matrices de confusión obtenidas en la etapa de validación. Ver Figura 5.5.

En la Tabla 5.1 se presentan los valores de la exactitud (*accuracy*) obtenidos por las matrices de confusión, utilizando la fórmula (5.1) del apartado 5.2.1. El valor promedio de estos valores es de un 83% de exactitud en el modelo.

	LowGrade	MediumGrade	HighGrade
Sensibilidad	0,66 ± 0,1008	0,82 ± 0,0680	0,92 ± 0,0503
Precisión	0,86 ± 0,0993	0,96 ± 0,1298	0,86 ± 0,0174
Especificidad	0,92 ± 0,0305	0,85 ± 0,0526	0,85 ± 0,0243

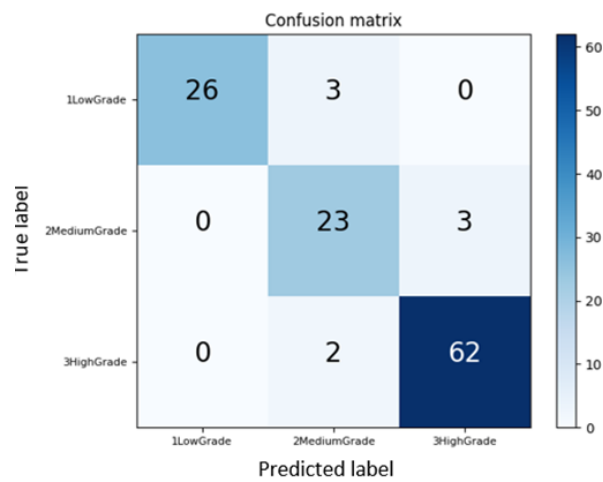
ACC	0,83
-----	------

Tabla 5.2: Métricas cuantitativas obtenidas de la etapa de validación para el modelo VGG16 utilizando transfer learning con la 5 capa convolucional reentrenada y top model ‘GMP’.

En la Tabla 5.2 se presentan los valores promedio individuales (para cada clase) obtenidos por las matrices de confusión de cada partición de datos. Estos valores se han calculado con las fórmulas explicadas en el apartado 5.2.1. El *accuracy* es el mismo que en la Tabla 5.1 ya que es el valor

promedio de todas las particiones.

- Resultados cuantitativos de la etapa de predicción.



**Figura 5.6:** Matriz de confusión obtenida de la etapa de predicción para el modelo VGG16 utilizando transfer learning con la 5 capa convolucional reentrenada y top model 'GMP'.

	LowGrade	MediumGrade	HighGrade
Sensibilidad	0,90	0,88	0,97
Precisión	1,00	0,82	0,95
Especificidad	1,00	0,95	0,95

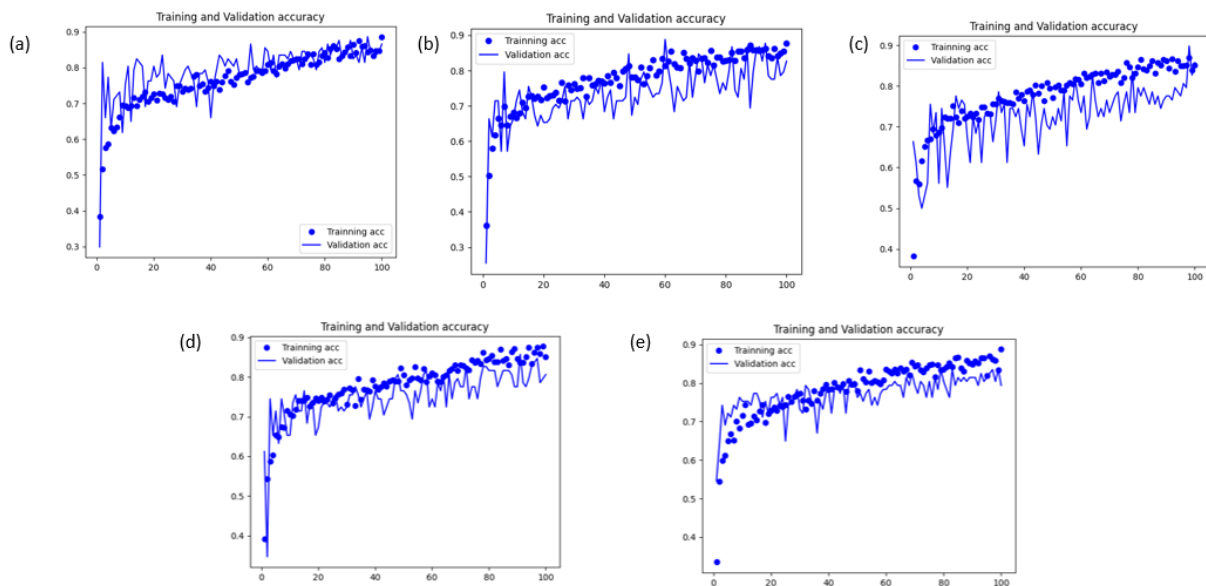
ACC	0,93
-----	------

**Tabla 5.3:** Métricas cuantitativas obtenidas de la etapa de predicción.

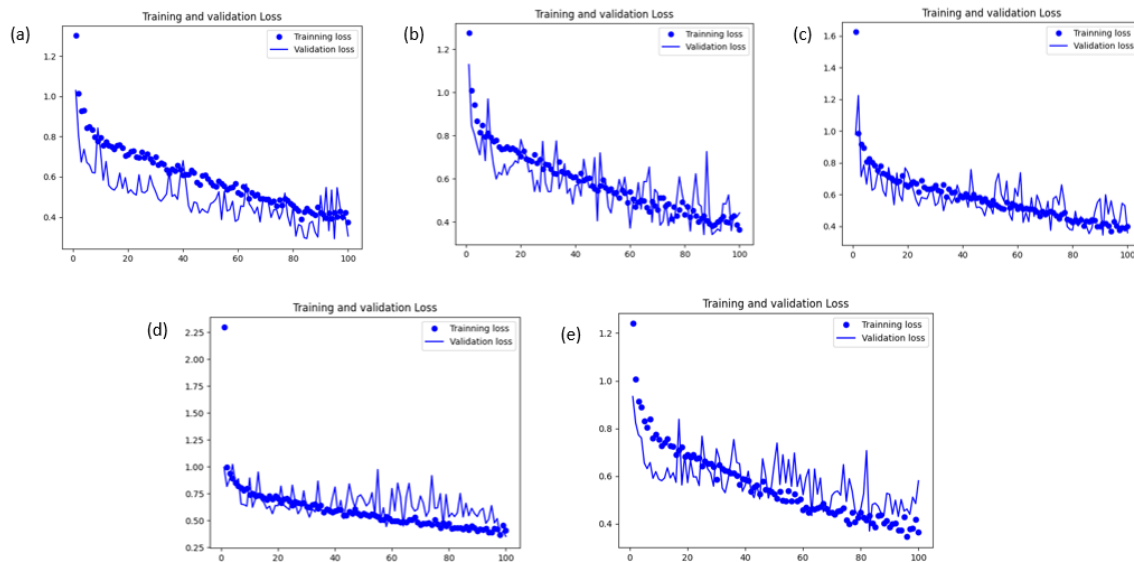
En la Tabla 5.3 se presentan los resultados individuales (para cada clase) obtenidos del mejor modelo para la etapa de predicción. El conjunto de datos que se ha utilizado en esta etapa, son los de test. El accuracy total se obtiene utilizando la fórmula (5.1) del apartado 5.2.1.

### 5.3.2 Curvas de aprendizaje obtenidas en la validación cruzada

#### 5.3.2.1 Modelo From scratch (FS) - Top model: GMP.



**Figura 5.7:** Representación de las curvas de aprendizaje durante el entrenamiento para el modelo From scratch. Valor de la exactitud (Acc) en cada época: (a) Kfold1, (b) Kfold2, (c) Kfold3, (d) Kfold4 y (e) Kfold5.



**Figura 5.8:** Representación de las curvas de aprendizaje durante el entrenamiento para el modelo From scratch. Valor de las pérdidas en cada época: (a) Kfold1, (b) Kfold2, (c) Kfold3, (d) Kfold4 y (e) Kfold5.

#### 5.3.2.2 Discusión sobre los resultados de las curvas de aprendizaje obtenidas en la validación cruzada para el modelo FS con top model GMP.

En el punto 5.3.1.2 se realiza una breve descripción sobre la utilidad de las curvas de aprendizaje y

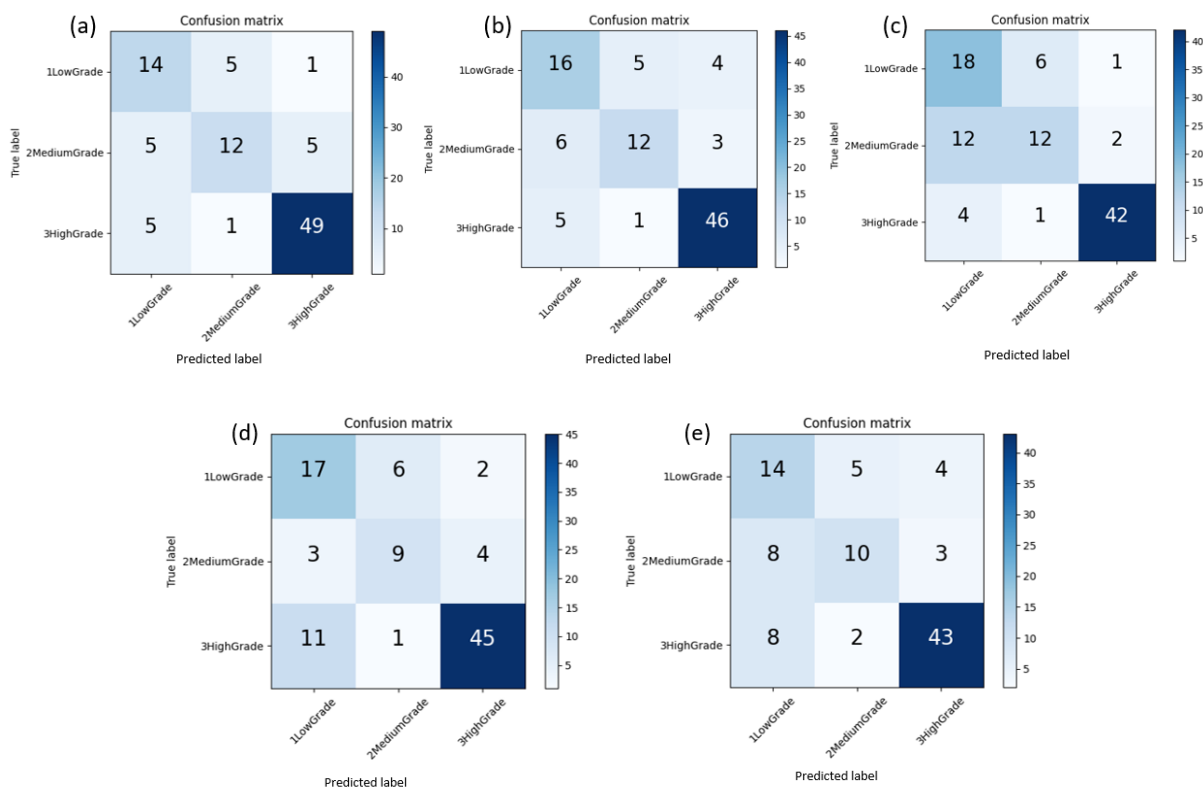
el valor de sus ejes ( $x$  e  $y$ ).

El número de épocas utilizado para este modelo fue de  $epochs=100$ .

La interpretación que se puede obtener de estas gráficas (tanto de *training* y *validation* Figura 5.7 como de *training* y *loss* Figura 5.8) es la siguiente:

- (a) En la época  $\approx 77$  se obtiene el valor mínimo de las pérdidas ( $loss \approx 0,44$ ) siendo el valor máximo de la exactitud en la validación de  $acc \approx 0,82$ .
- (b) En la época  $\approx 85$  se obtiene el valor mínimo de las pérdidas ( $loss \approx 0,44$ ) siendo el valor máximo de la exactitud en la validación de  $acc \approx 0,87$ .
- (c) En la época  $\approx 97$  se obtiene el valor mínimo de las pérdidas ( $loss \approx 0,42$ ) siendo el valor máximo de la exactitud en la validación de  $acc \approx 0,85$ .
- (d) En la época  $\approx 80$  se obtiene el valor mínimo de las pérdidas ( $loss \approx 0,42$ ) siendo el valor máximo de la exactitud en la validación de  $acc \approx 0,82$ .

▪ **Resultados cuantitativos de la etapa de validación:**



**Figura 5.9:** Matrices de confusión obtenidas de la etapa de validación del modelo From scratch: (a) Kfold1, (b) Kfold2, (c) Kfold3, (d) Kfold4 y (e) Kfold5.

K-PARTICION	FOLD1_VAL	FOLD2_VAL	FOLD3_VAL	FOLD4_VAL	FOLD5_VAL
ACC_TOTAL_MODELO	0,7732	0,7551	0,7347	0,7245	0,6907
ACC_PROMEDIO	0,74				

**Tabla 5.4:** Valores de la exactitud individual de cada partición de datos y el promedio de estos. Ver Figura 5.9

En la **Tabla 5.1**: Valores de la exactitud de las matrices de confusión obtenidas en la etapa de validación. Ver Figura 5.5. Tabla 5.4 se presentan los valores de la exactitud (*accuracy*) obtenidos por las matrices de confusión, utilizando la fórmula (5.1) del apartado 5.2.1. El valor promedio de estos valores es de un 74% de exactitud en el modelo.

	LowGrade	MediumGrade	HighGrade
<b>Sensibilidad</b>	0,67 ± 0,0452	0,52 ± 0,0509	0,85 ± 0,0496
<b>Precisión</b>	0,54 ± 0,0503	0,62 ± 0,0468	0,89 ± 0,0286
<b>Especificidad</b>	0,92 ± 0,0398	0,85 ± 0,0081	0,85 ± 0,0413

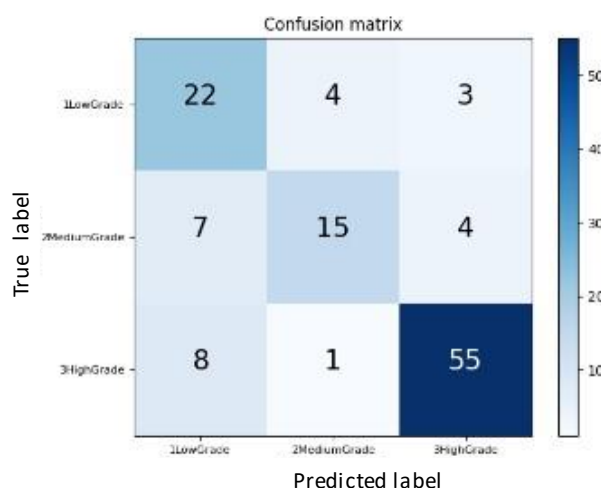
  

<b>ACC</b>	0,74
------------	------

**Tabla 5.5:** Métricas cualitativas obtenidas de la etapa de validación.

En la Tabla 5.5 se presentan los valores promedio individuales (para cada clase) obtenidos por las matrices de confusión de cada partición de datos. Estos valores se han calculado con las fórmulas explicadas en el apartado 5.2.1. El *accuracy* es el mismo que en la Tabla 5.4 ya que es el valor promedio de todas las particiones.

▪ **Resultados cuantitativos de la etapa predicción:**



**Figura 5.10:** Matriz de confusión obtenida de la etapa de predicción del modelo From scratch.

	LowGrade	MediumGrade	HighGrade
<b>Sensibilidad</b>	0,76	0,58	0,86
<b>Precisión</b>	0,59	0,75	0,89
<b>Especificidad</b>	0,83	0,95	0,87

<b>ACC</b>	0,77
------------	------

**Tabla 5.6:** Métricas cualitativas obtenidas de la etapa de predicción.

En la Tabla 5.6 se presentan los resultados individuales (para cada clase) obtenidos del mejor modelo para la etapa de predicción. El conjunto de datos que se ha utilizado en esta etapa, son los de test. El *accuracy* total se obtiene utilizando la fórmula (5.1) del apartado 5.2.1.

## Capítulo 6

# Conclusiones y líneas futuras

### ÍNDICE DE CONTENIDO:

<b>CONCLUSIONES Y LÍNEAS FUTURAS .....</b>	<b>63</b>
6.1 Conclusiones .....	64
6.2 Líneas futuras .....	64
<b>BIBLIOGRAFÍA .....</b>	<b>65</b>



## 6.1 Conclusiones

A la vista de los resultados presentados en el capítulo anterior se puede concluir que, a pesar de haber cumplido de manera satisfactoria la mayor parte de los objetivos propuestos en el punto 1.2 de este mismo documento, el objetivo principal de este trabajo no se llega a cumplir por los siguientes motivos:

- *En relación con el preprocesado de las imágenes: etiquetado*

Resulta evidente que, si los datos de entrenamiento con los que vamos a trabajar no están correctamente preprocesados, al sistema le será más difícil detectar y extraer de manera eficiente las características de los patrones subyacentes, de manera que tiene menos probabilidades de funcionar bien.

Para abordar este problema, se realizó, una exhaustiva limpieza de datos (*data cleaning*) de los patches generados que se filtraron por error (por no contener información relevante, estar mal etiquetados o el patrón etiquetado no estar bien diferenciado).

- *En relación con la base de datos y la experiencia del programador:*

Para lograr tener mejores resultados en la elaboración de un sistema de aprendizaje profundo, además de tener un buen set de datos (con muchas imágenes y en donde los patrones a buscar se puedan diferenciar bien) es muy necesario tener experiencia o al menos partir con una buena base sobre aprendizaje profundo ya que un trabajo de estas características supera el tiempo proporcionado para la realización de una tesis de máster.

- *En relación con la mejora de los modelos:*

Al tener pocas muestras en la base de datos, la autora de este trabajo se vio limitada a la hora de diseñar la red neuronal dado que hiperparámetros como el *batch-size* y el número de épocas a entrenar se vio restringido por falta de almacenamiento en el servidor proporcionado.

## 6.2 Líneas futuras

Dado que este tipo de algoritmos desarrollados para la búsqueda de este determinado patrón (TB) se encuentra en fases tempranas de investigación, no se cuenta aún con una cantidad de muestras suficientes para poder elaborar un buen corpus de datos (ciento de miles o, al menos, superior a 10000 imágenes) que nos permita mejorar los resultados de los modelos presentados en este trabajo académico.

Tan importante como tener datos suficientes es conocer la herramienta que se utiliza en el etiquetado para, de esta manera, poder dotar al profesional experto (en este caso, al patólogo) de la información acerca de su uso y cómo de ajustado se quiere la selección del patrón a identificar. Esto último ayudará bastante a que el sistema aprenda y extraiga las características del patrón de una forma más fiable.

Teniendo en cuenta estas dos observaciones, las líneas futuras se centrarán en la búsqueda de mejoras de los algoritmos implementados para ofrecer una herramienta robusta y con alta sensibilidad en la detección de patrones de crecimiento tumoral en muestras histopatológicas de carcinoma urotelial músculo-infiltrante.

## Bibliografía

- [1] Organización Mundial de la Salud (OMS), URL: <https://www.who.int/es/news-room/fact-sheets/detail/cancer>.
- [2] Global Cancer Observatory (GCO), URL: <https://gco.iarc.fr/today/fact-sheets-populations>
- [3] Marzo-Castillejo M., Vela-Vallespín C., Bellas-Beceiro B., Bartolomé-Moreno C., Ginés-Díaz Y. y Melús-Palazón E., “Grupos de Expertos de Cáncer del PAPPS. Recomendaciones de Prevención del Cáncer. Actualización PAPPS 2020”, Atención Primaria 2020;52(S2):44---69.
- [4] Agencia española contra el cáncer (AECC), URL: <https://www.aecc.es/es/todo-sobre-cancer/tipos-cancer/cancer-vejiga>.
- [5] Understand to learn, “Capas de la pared de la vejiga”, URL: <http://brenuvm.blogspot.com/2013/06/>.
- [6] Andrew T., Patrick M., Karim C., “Bladder cancer.” A Review. JAMA 2020;324(19):1980-1991.
- [7] “Tratamiento del cáncer de vejiga con invasión muscular”, URL: <https://patients.uroweb.org/treatments/muscle-invasive-bladder-cancer/>
- [8] Oner S et al. (2017) “Bladder cancer”, Nat Rev Dis Primers 3, 17022
- [9] Redacción médica, URL: <https://www.redaccionmedica.com/recursos-salud/diccionario-enfermedades/cancer-vejiga>.
- [10] La vanguardia. Prensa online, URL: <https://www.lavanguardia.com/vida/salud/tipos-cancer/20190716/463513609653/cancer-cancer-de-vejiga-carcinoma-adenocarcinoma-sarcoma-sangre-en-la-orina.html>.
- [11] Información al paciente: American Society of Clinical Oncology (ASCO), URL: <https://www.cancer.net/es/tipos-de-c%C3%A1ncer/c%C3%A1ncer-de-vejiga/estad%C3%ADsticas>.
- [12] World Bladder cancer patient, “Globocan 2020” URL: [https://worldbladdercancer.org/news\\_events/globocan-2020-bladder-cancer-10th-most-commonly-diagnosed-worldwide/](https://worldbladdercancer.org/news_events/globocan-2020-bladder-cancer-10th-most-commonly-diagnosed-worldwide/)

- [13] Larré S., Leon P., El Bakri A., “Cáncer de vejiga: diagnóstico y principios terapéuticos”, EMC -Urología, Vol 48, nº 4 diciembre de 2016.
- [14] American Cancer Society, Cáncer de vejiga, URL: <https://www.cancer.org/es/cancer/cancer-de-vejiga/deteccion-diagnostico-clasificacion-por-etapas/deteccion.html>
- [15] Anales de pediatría, Urotelioma vesical, URL: <https://www.analesdepediatria.org/es-urotelioma-vesical-paciente-12-anos-articulo-13101248>
- [16] Técnica para el procesamiento de material para microscopía electrónica, Monografías.com, URL: <https://www.monografias.com/docs110/metodo-preparacion-muestras-histologicas/metodo-preparacion-muestras-histologicas.shtml>
- [17] Procesamiento de Muestras para Anatomía patológica, Slideshare.net, URL: <https://www.slideshare.net/cosaparahacerunblog/procesamiento-de-muestras-para-anatoma-patologica-5969905/6>
- [18] Jimenez RE, Gheiler E, Oskanian P, Tiguert R, Sakr W, Wood DP, et al. Grading the invasive component of urothelial carcinoma of the bladder and its relationship with progression-free survival. Am J Surg Pathol. 2000;24(7):980–7.
- [19] Lorenzo L., Tesis Doctoral, Valor pronóstico de la presencia de un componente tumoral indiferenciado (“Tumor Budding”) en pacientes con carcinoma vesical músculo invasivo.
- [20] Esteve Domínguez, A., Trabajo final de Máster, Diseño y desarrollo de un sistema automático basado en algoritmos de deep learning para identificar distintos grados de tumor budding en cáncer de vejiga
- [21] Labalde, M., Labrador F., Golitsin A., Santiago Pinto S., Carlin J., García-Parreño Jorge.; Tumor budding y el cáncer colorrectal; Rev Chil Cir. Vol 66 - Nº 6, diciembre 2014; pág. 556-561.
- [22] Brieu, N., Gavriel, C., Nearchou, I., Harrison, D., Schmidt, G & Caie, P.; Automated tumour budding quantification by machine learning augments TNM staging in muscle-invasive bladder cancer prognosis; Scientific Reports | (2019) 9:5174.
- [23] Gonzalez, R., Woods R., Eddins, S., “Digital Image Processing using Matlab”, Gatesmark, LLC 2009.

- [24] Aurélien G., “Aprende Machine Learning con Scikit-Learn, Keras y TensorFlow”. O’Reilly 2019.
- [25] MobaXterm, URL: <https://mobaxterm.mobatek.net/>.
- [26] Microdraw, URL: <http://microdraw.pasteur.fr/>.
- [27] Docker, URL: <https://www.docker.com/>
- [28] Python, URL: <https://www.python.org/doc/essays/blurb/>.
- [29] MathWorks, URL: [https://es.mathworks.com/help/matlab/learn\\_matlab/product-description.html](https://es.mathworks.com/help/matlab/learn_matlab/product-description.html).
- [30] NVIDIA cuDNN, URL: <https://developer.nvidia.com/cudnn>.
- [31] Ortega Candel J., “Docker Seguridad y monitorización en contenedores e imágenes”. RC Libros, 2019.
- [32] Chollet F., “Deep Learning con Python”, Ediciones Anaya multimedia (Grupo ANAYA, S.A), 2020.
- [33] García G., Esteve A., Colomer A., Ramos D. and Naranjo V.; A novel Self-Learning Framework for Bladder Cancer Grading Using Histopathological Images, 2106.13559v1 [eess.IV].
- [34] Patch Transformer for Multi-tagging Whole Slide Histopathology Images.
- [35] Wallimbe, R; Handling imbalanced dataset in supervised learning using family of SMOTE algorithm; Data Science Central. URL: <https://www.datasciencecentral.com/profiles/blogs/handling-imbalanced-data-sets-in-supervised-learning-using-family>
- [36] TensorFlow, URL: <https://www.tensorflow.org>
- [37] Skicit-learn. Class weight. URL: [https://scikit-learn.org/stable/modules/generated/sklearn.utils.class\\_weight.compute\\_class\\_weight.html](https://scikit-learn.org/stable/modules/generated/sklearn.utils.class_weight.compute_class_weight.html)
- [38] Python Deep learning. Introducción práctica con Keras y tensorFlow 2. Jordi Torres
- [39] Naranjo V., Colomer A., Curso Deep Learning aplicado al análisis de señales; Sesión\_5.pdf.

- [40] Simonyan, K.; Zisserman A.; Very deep convolutional networks for large-scale image recognition; Visual Geometry Group, Department of Engineering Science, University of Oxford.
- [41] Tinsy J.; What is VGG16?. Introduction to VGG16; 09/2021; enlace web: <https://medium.com/@mygreatlearning/what-is-vgg16-introduction-to-vgg16-f2d63849f615>
- [42] Understanding the VGG-16 model; enlace web: [https://ebookreading.net/view/book/EB9781788831307\\_191.html](https://ebookreading.net/view/book/EB9781788831307_191.html)
- [43] Egea J.M; Redes neuronales: concepto, fundamentos y aplicaciones en el laboratorio clínico; Química clínica 1994;13(5): 221-228.
- [44] Calvo, D.; Clasificación de Redes neuronales; enlace web: <https://www.diegocalvo.es/clasificacion-de-redes-neuronales-artificiales/>
- [45] Keras API reference: <https://keras.io/>
- [46] Aperio CS2 y software Aperio ImageScope de Leica; enlace web: <https://www.medicaexpo.es/prod/leica-biosystems/product-95735-653511.html>
- [47] Azure Machine Learning, sobreajuste y subajuste (Overfitting and Underfitting); enlace web: <https://epicalsoft.com/2019/02/27/azure-machine-learning-sobreajuste-y-subajuste-over%EF%AC%81tting-and-under%EF%AC%81tting/>
- [48] Optimizador Adagrad; Referencia de la API de keras; enlace web: <https://keras.io/api/optimizers/adagrad/>.
- [49] Optimización de lotes grandes. Adagrad; enlace web: <https://paperswithcode.com/method/adagrad>
- [50] Global Max pooling 2D; autores: Min L.; Qiang C.; Shuicheng Y.; enlace web <https://peltarion.com/knowledge-center/documentation/modeling-view/build-an-ai-model/blocks/global-max-pooling-2d>
- [51] Zizhao Z., Pingjun C., Mason M. et al; Pathologist-level interpretable whole-slide cancer diagnosis with deep learning; DOI: <https://doi.org/10.1038/s42256-019-0052-1>

Parte II

# **Presupuestos**

## Capítulo 1

# Presupuesto

### ÍNDICE DE CONTENIDOS

1.1	Objetivo.....	1
1.2	Presupuestos parciales.....	1
	i)Costes de personal.....	1
	ii)Coste de materiales .....	1
	iii)Coste de software .....	2
	iv)Coste de hardware.....	2
1.3	Presupuesto Total .....	3

## 1.1 Objetivo

El objetivo que persigue esta sección es la de aportar una valoración económica *aproximada* de este proyecto basado en el Diseño y desarrollo de software para el análisis de patrones microscópicos de crecimiento tumoral en pacientes con carcinoma vesical.

Dado que para este proyecto se han utilizado **Materiales** (*muestras de imágenes histopatológicas obtenidas de [20]*); estos costes se tendrán en cuenta para la valoración económica final.

## 1.2 Presupuestos parciales

Con la finalidad de poder cuantificar los presupuestos parciales, estos se han desglosado de la siguiente manera: (i) Costes de personal, (ii) coste de materiales, (iii) coste de software y (iv) coste de hardware. De esta manera cualquier modificación adicional *o no*, se va a poder realizar de forma independiente, facilitando de esta manera el incremento o decremento de estos.

### *i) Costes de personal*

En la siguiente tabla se desglosa: el coste de la mano de obra (de los participantes) en base al tiempo invertido en este proyecto.

Descripción	Uds. (horas)	Cantidad	Precio unitario (€/h)	Coste imputable (€)
Director del TFM (Titular de Universidad)	h	25	29,50	737,50
Cotutor del TFM (Doctor)	h	70	28,00	1.960
Profesional encargado del etiquetado (Patólogo Forense)	h	20	28,00	560
Estudiante y autora del TFM	h	600	14,23	8.538
<b>TOTAL</b>				<b>11.795,50€</b>

*Tabla 1.2-i: Desglose del coste de mano de obra.*

### *ii) Coste de materiales*

Como ya se anticipó en el punto **1.1**, para este proyecto se han utilizado materiales que son procedentes de otro trabajo de investigación [20]. Es por esto, por lo que es necesario añadir estos costes para no causar perjuicio alguno sobre el trabajo de investigación antes mencionado.

Aclarado este punto, los costes se desglosan en la siguiente tabla: Tabla **2.11**: Agrupación por etapas



Diseño y desarrollo de software para el análisis de patrones microscópicos de crecimiento tumoral en pacientes con carcinoma vesical.

Descripción	Cantidad (u)	Precio unitario (€/h)	Coste imputable (€)
Obtención de biopsias	120	600	7.200
Preparación de muestras histológicas	140	10	1.400
	<b>TOTAL</b>		8.600

o estadios del cáncer de vejiga. Información obtenida de referencia [11].

*iii) Coste de software*

Descripción	Cantidad (u)	Coste (€) sin IVA	Periodo de amortización (meses)	Intervalo de uso (meses)	Coste imputable (€)
Matlab R2019b	1	800	12	7	466,60
Librerías <i>Tensorflow-Keras</i>	1	0	12	3	0
	<b>TOTAL</b>				466,60

**Tabla 1.2-7:** Desglose del coste del software utilizado para este trabajo final de máster.

*iv) Coste de hardware*

Descripción	Cantidad (u)	Coste (€) sin IVA	Periodo de amortización (meses)	Intervalo de uso (meses)	Coste imputable (€)
Tarjeta gráfica NVIDIA (Titan V)	1	3300,00	48	3	206,25
NAS Synology DS918	1	77,00	48	3	4,81
DELL 97FGJ64, Intel® i7-3770 CPU @3.4 GHz, Disco 500 GB HDD, HD Graphics 4000, 8GB de RAM	1	740,00	60	7	86,30

Diseño y desarrollo de software para el análisis de patrones microscópicos de crecimiento tumoral en pacientes con carcinoma vesical.

	<b>TOTAL</b>	297,39
--	--------------	--------

**Tabla 1.2-8:** Desglose del coste hardware utilizado para este trabajo final de máster.

### 1.3 Presupuesto Total

Una vez realizados todos los cálculos del apartado 1, en este apartado se procederá a obtener (a partir de la suma de ellos), el Presupuesto de Ejecución de Material (PEM).

DESCRIPCIÓN	Coste (€)
Costes de mano de obra	11.795,50
Costes de materiales	8.600,00
Costes de software	466,60
Costes de hardware	297,39
<b>Suma (PEM)</b>	<b>21.159,49</b>

**Tabla 1.3 (i):** Cálculo de Presupuesto de ejecución de material (PEM)

A este presupuesto PEM se les sumarán los impuestos correspondientes a: gastos generales (13%) y al beneficio industrial (6%) y el total de estos será el presupuesto total.

<b>Presupuesto total</b>	
Presupuesto PEM	21.159,49
Gastos generales (13%)	2.750,73
Beneficio industrial (6%)	1.269,57
Suma total	25.179,79
IVA (21%)	5287,76
<b>TOTAL</b>	<b>30.467,54</b>

**Tabla 1.3 (ii):** Cálculo del presupuesto total