
**Universidad de Valladolid**

**ESCUELA TÉCNICA SUPERIOR DE INGENIEROS DE TELECOMUNICACIÓN**

**GRADO EN INGENIERÍA DE TECNOLOGÍAS DE TELECOMUNICACIÓN**

---

# ANÁLISIS DE SEÑALES DE TOS PARA DETECCIÓN TEMPRANA DE ENFERMEDADES RESPIRATORIAS

---

**Alumno: DIEGO ASAY PÉREZ ALONSO**

**Tutores: JUAN PABLO CASASECA DE LA HIGUERA**
**CARLOS ALBEROLA LÓPEZ**

**Valladolid, 2 de Septiembre de 2019**

# ANÁLISIS DE SEÑALES DE TOS PARA DETECCIÓN TEMPRANA DE ENFERMEDADES RESPIRATORIAS

Diego Pérez Alonso

# ÍNDICE GENERAL

# ÍNDICE DE FIGURAS

# ÍNDICE DE TABLAS

*Somos como enanos a los hombros de gigantes.*
*Podemos ver más, y más lejos que ellos,*
*no por la agudeza de nuestra vista ni por la altura de nuestro cuerpo,*
*sino porque somos levantados por su gran altura.*
Bernardo de Chartres

*Lo que nos hace grandes es*
*el hecho de poder ver lo pequeños que somos.*
Martí Monteferrer

*Lo que sabemos es una gota de agua;*
*lo que ignoramos es el océano.*
Isaac Newton

*En algún lugar, alguna cosa increíble*
*aguarda a ser descubierta.*
Carl Sagan

# AGRADECIMIENTOS

En primer lugar, quiero expresar mi gratitud a mis tutores Juan Pablo Casaseca de la Higuera y Carlos Alberola López por darme la posiblidad de realizar este Trabajo Fin de Grado y por su esfuerzo y tiempo dedicado a guiarme en el transcurso del mismo.

También me gustaría hacer especial mención al Laboratorio de Procesado de Imagen de la Universidad de Valladolid por haber puesto a mi disposición los medios necesarios para el desarrollo de este Trabajo Fin de Grado.

Gracias a mi familia, compañeros y amigos, por el apoyo y el ánimo que me han brindado, especialmente a mis padres y a mi hermano, que me han apoyado en los momentos más difíciles.

# RESUMEN

*Antecedentes*: La tos es un movimiento sonoro y convulsivo del aparato respiratorio. Hasta ahora, el análisis de la tos como síntoma informativo de la evolución de una enfermedad se limita a herramientas de medición subjetivas, o incómodos monitores de la tos. Otro limitante actual, se debe a que los métodos de procesamiento de audio implementados en dichos monitores no pueden hacer frente a entornos ruidosos, como en el caso en que el dispositivo de adquisición sea un *smartphone* que el paciente pueda llevar en su bolsillo.

*Objetivo*: El objetivo de este Trabajo de Fin de Grado (TFG) es diseñar diseñar un sistema de "audición máquina" (*Machine Hearing*) mediante una arquitectura de aprendizaje profundo (*Deep Learning*) para realizar la detección de tos, así como la detección de enfermedades respiratorias con carácter temprano a partir de señales de audio ruidosas.

*Métodos*: Para realizar el proyecto, se han utilizado señales de audio ruidosas de veinte pacientes con diferentes enfermedades respiratorias, 18433 señales de audio grabadas durante episodios de tos y 18433 señales de audio grabadas durante episodios sin tos. Dichas señales de audio son preprocesadas en tres pasos. Primero, se segmentan las señales de audio originales (señales de tos y no tos) para que todas tengan una duración de un segundo. En segundo lugar, se realiza un espectrograma logarítmico a cada audio para transformar las señales 1D temporales en imágenes (señales 2D) tiempo-frecuencia. Finalmente, se normalizan los datos para poder alimentar a una red neuronal convolucional (*Convolutional Neural Network*, CNN), que realiza automáticamente la extracción de características en los espectrogramas de los audios para identificar "firmas" espectrales o temporales. De esta forma en primer lugar se detecta si dicho audio contiene una tos o no, y en caso de que la contenga, se pasaría al diagnóstico de la enfermedad respiratoria.

*Resultados*: El sistema de detección de audios con toses tiene una sensibilidad del 85,64 % y una especificidad del 92,81 %. Con respecto a la detección temprana de enfermedades respiratorias, se ha alcanzado una tasa de acierto del 77,78 % cuando el sistema diagnostica si un paciente tiene tos aguda o enfermedad pulmonar obstructiva crónica (*Chronic Obstructive Pulmonary Disease*, COPD), superando a los métodos más modernos.

*Conclusiones*: Los resultados de este TFG allanan el camino para crear un dispositivo cómodo y no intrusivo, con una interrupción mínima en las actividades diarias, que pueda detectar con carácter temprano enfermedades respiratorias, beneficiando a pacientes, profesionales sanitarios y sistemas nacionales de salud.

**Palabras claves:** *Deep Learning*, Tos, Detector, Análisis tiempo-frecuencia, Enfermedades respiratorias.

# ABSTRACT

*Background*: Cough is a sound and convulsive movement of the respiratory system. So far, cough analysis as an informative symptom of the evolution of a disease is limited to subjective measurement tools, or uncomfortable cough monitors. Another current limitation resides in the audio processing methods implemented in such monitors, as they cannot cope with noisy environments such as those where the acquisition device is a smartphone carried in a pocket.

*Objective*: The objective of this Final Year Project is to design a "Machine Hearing" system using a Deep Learning architecture to perform cough detection, as well as cough analysis for early detection of respiratory diseases from noisy audio signals.

*Methods*: We acquired 36866 audio signals from 20 respiratory patients. Half of them corresponded to cough episodes, whereas the other half did not contain any cough sound. These audio signals are preprocessed in three steps. First, the original audio signals (cough and non-cough signals) are segmented so that they all last one second. Secondly, a logarithmic spectrogram is computed over each audio clip to transform the temporary 1D signals into time-frequency images (2D signals). Finally, the data is normalized to be able to feed a Convolutional Neural Network (CNN), which automatically performs the extraction of features in audio spectrograms to identify time-frequency "signatures". This way, we first detect whether the audio clip contains a cough or not for further analysis aiming at detecting the underlying disease.

*Results*: The cough audio detection system presents 85.64 % sensitivity and 92.81 % specificity. In terms of early detection of respiratory diseases, a success rate of 77.78 % has been obtained when the diagnostic system is a patient has acute cough or Chronic Obstructive Pulmonary Disease (COPD), outperforming modern methods.

*Conclusions*: The results of this Final Year Project pave the way to create a comfortable and non-intrusive device, with minimal disruption in daily activities, which can be detected with the early nature of respiratory diseases, benefiting patients , health professionals and national health systems.

**Keywords:** *Deep Learning*, Cough, Detector, Time-frequency analysis, Respiratory diseases.

# *Capítulo* 1
# INTRODUCCIÓN

En este capítulo, buscamos establecer el contexto del estudio realizado en este Trabajo de Fin de Grado, donde se emplean técnicas de aprendizaje profundo (*Deep Learning*, DL) para clasificar espectrogramas de señales acústicas. Además, se pretende diferenciar los espectrogramas que contienen tos y los que no contienen tos, y una vez es conocido que un espectrograma contiene una tos, conseguir conocer la enfermedad subyacente.

## 1.1. Motivación

La tos es un movimiento sonoro y convulsivo del aparato respiratorio [1], que transmite información sobre el estado del sistema respiratorio, el cual es un reflejo inherente a la protección del aparato respiratorio con un sonido característico y un movimiento corporal asociado. También podemos definir la tos como un reflejo respiratorio protector que permite la expulsión de material extraño y secreciones de la vía aérea y la laringe. Consiste en una salida de aire brusca y fulminante a través de la glotis, componiendo su mecanismo un complejo arco reflejo, en donde participan el centro de la tos cerebral y el nervio laríngeo superior, rama del nervio vago. Cuando se activan los mecanorreceptores y quimiorreceptores de la mucosa de las vías aéreas y de otras localizaciones, se produce un estímulo que, por vía aferente, llega al centro de la tos en el cerebro a través del nervio vago. Desde este centro, se produce una respuesta por vía eferente con modulación del córtex cerebral que activa la musculatura de la tráquea, la faringe y los bronquios [2].

Según [3], las enfermedades pulmonares son causantes de una sexta parte de las muertes que se producen en el mundo, su impacto continúa siendo tan grande como en el siglo pasado y el pronóstico es que siga siéndolo durante varias décadas, de hecho, se proyecta que las enfermedades pulmonares causarán una de cada 5 muertes en el mundo. El problema de las enfermedades pulmonares causan cada año aproximadamente un millón de muertes en la región europea de la Organización Mundial de la Salud (OMS), ocurriendo dos tercios de estas en los 28 países de la Unión Europea (UE). Cada año mueren 600.000 personas en la UE de enfermedad respiratoria, es decir, una de cada ocho muertes se deben a enfermedades respiratorias, y al menos 6 millones de ingresos hospitalarios son causados por trastornos pulmonares.

El cáncer de pulmón es la principal causa de muerte por enfermedades respiratorias, seguido de la Enfermedad Pulmonar Obstructiva Crónica (EPOC), infecciones de las vías respiratorias bajas y la tuberculosis. Unos 5,4 millones de años potenciales de vida perdidos se calcula que son debidos a enfermedades respiratorias como el asma. La tos es un síntoma que deteriora gravemente la calidad de vida del paciente y que ocasiona un alto consumo de medicamentos. El tabaco causa más de la mitad de las muertes a causa de patologías respiratorias. Además es preocupante la reaparición de la tuberculosis multirresistente y el incremento de enfermedades como la EPOC.

Las enfermedades pulmonares causan discapacidad y muerte prematura. Tienen un enorme coste relacionado con la atención primaria, la atención hospitalaria y tratamientos, así como la pérdida de productividad de los que no pueden trabajar y las personas que mueren antes de tiempo debido a su condición. Lo cual da lugar, a que cada año se pierdan 5,2 millones de años de vida ajustados por discapacidad debido a enfermedades respiratorias en la UE, por un valor de 300.000 millones de euros.

El coste total de las enfermedades respiratorias en la UE supera los 380.000 millones de euros, siendo este un coste inaceptable al que debe enfrentarse la UE, proponiendo soluciones como la reducción del uso del tabaco de manera activa, ya que el tabaquismo produce la mitad de la carga económica de las enfermedades pulmonares. La carga más grande corresponde a la EPOC y el asma, con más de 200.000 millones de euros, seguido del cáncer de pulmón que alcanza un coste total de 100.000 millones de euros. Las cifras actuales son probablemente subestimaciones considerables [3].

Hasta ahora, el análisis de la tos como síntoma informativo de la evolución de una enfermedad se limita a herramientas de medición subjetivas, o incómodos monitores de la tos [1]. Los métodos subjetivos se basan en diarios o cuestionarios de calidad de vida en los que los pacientes pueden expresar su apreciación de la gravedad de la tos. Por un lado, estos métodos son baratos y fácilmente aplicables en la atención primaria, pero, por otro lado, pueden estar sesgados debido a la comorbilidad física y psicológica de la tos (*e.g.*, incontinencia, dolor de pecho o vergüenza social), variabilidad entre expertos y otros factores como la personalidad o el estado de ánimo. La aparición de dispositivos de atención médica para monitorizar objetivamente la tos han sido fomentados debido al desarrollo de tecnologías digitales. Estos dispositivos fundamentalmente funcionan gracias a motores de reconocimiento de patrones basados principalmente en características extraídas de los sonidos de la tos y señales complementarias como la electromiografía del movimiento del pecho. Sin embargo, la mayoría de estos sistemas solo han sido probados en entornos controlados donde los pacientes no realizaron ningún movimiento, ni actividad física, ni obligaron a los usuarios a usar sistemas de registro complejos. Por lo tanto, un limitante actual se debe a que los métodos de procesamiento de audio implementados en dichos monitores no pueden hacer frente a entornos ruidosos, como en el caso en que el dispositivo de adquisición sea un *smartphone* que el paciente pueda llevar en su bolsillo [1].

Un monitor portátil para la tos podría implementarse en un teléfono inteligente para la monitorización continua en tiempo real de pacientes con enfermedades respiratorias, mediante grabaciones de audio procedentes de entornos ruidosos de la vida real. Esto constituiría un dispositivo fiable para los profesionales, de modo que se pueda fomentar el potencial de la telemedicina en el contexto de las enfermedades respiratorias. Además, desde el punto de vista del paciente, este sistema de monitorización sería más cómodo y causaría una interrupción mínima en sus actividades diarias. De esta manera, serían menos conscientes de su medicalización.

Los eventos de tos de audio son señales no estacionarias con una duración promedio de alrededor de 300 ms y compuestas de tres fases: explosiva, intermedia y sonora. Estas señales no tienen una estructura formante clara y se caracterizan por un contenido espectral disperso. Asimismo, los eventos de tos pueden tener: fases intermedias fuerte y débiles, fase vocal ausente y fase vocal fuerte y débiles. A pesar de que los eventos de tos presentan una forma de onda similar, existe una variabilidad inter e intra-paciente que afecta tanto la duración como la intensidad de las tres fases [4].

## 1.2. Objetivos

El objetivo perseguido con la elaboración de este trabajo es, por lo expuesto en la sección anterior, diseñar un monitor robusto para su implementación futura. Esta se hará en el futuro mediante un teléfono inteligente para poder monitorizar en tiempo real a pacientes respiratorios. De esta manera, los pacientes tendrán una interrupción mínima en las actividades diarias, ya que serían monitorizados mediante un dispositivo cómodo y no intrusivo. Este dispositivo, mediante la monitorización de los pacientes, podría detectar con carácter temprano enfermedades respiratorias, beneficiando a pacientes, profesionales sanitarios y sistemas nacionales de salud.

El objetivo global se desglosa en los siguientes subobjetivos:

- Adecuar los clips de audio que ofrece nuestra base de datos para introducirlos en una red neuronal convolucional (*Convolutional Neural Network*, CNN). Para llevar a cabo este proceso, debemos segmentar los clips de audio con el objetivo de que tengan una duración adecuada. Además, posteriormente realizamos un espectrograma adecuado para los clips de audio segmentados que maximiza la eficiencia de la red.

- Diseñar un sistema de "audición máquina" (*Machine Hearing*) mediante una arquitectura de *Deep Learning* utilizando una red neuronal convolucional que sea capaz de clasificar espectrogramas basados en clips de audio. En primer lugar, el sistema de "audición máquina" debe detectar los espectrogramas que contengan

tos frente a espectrogramas que no los contengan. En segundo lugar, el sistema de "audición máquina" debe clasificar la enfermedades respiratoria del paciente con carácter temprano, en base a los espectrogramas que han sido detectados anteriormente con tos.

- Obtener un sistema de mejores prestaciones que los propuestos hasta ahora en la literatura en materia de detección de tos y realizar una primera aproximación a la clasificación de enfermedades en base a los clips de audio con tos.

## 1.3. Fases y métodos

Para alcanzar los objetivos planteados, las labores a desarrollar se han llevado a cabo en dos etapas:

I. **Etapa de formación**, comprende el estudio de los conceptos básicos en varias áreas necesarias para la elaboración de este trabajo. Se pueden distinguir las siguientes fases:

   a) Estudio de los fundamentos de aprendizaje automático (*Machine Learning*, ML), *Deep Learning* y reconocimiento de patrones, con el objetivo de detectar y reconocer objetos mediante técnicas de aprendizaje profundo.

   b) Aprendizaje en el uso de la herramienta de software Praat, utilizada para el etiquetado de clips de audio.

   c) Estudio de los fundamentos de Python, así como de las bibliotecas:

      - `TensorFlow`, `Keras` y `Scikit-learn` fundamentales para implementar técnicas de *Deep Learning*.
      - `NumPy`, `SciPy` y `Matplotlib`, esenciales para realizar procesado de imagen en Python.
      - `Pandas`, básica para la manipulación y análisis de grandes cantidades de datos.
      - `tgt`, necesaria para leer los ficheros que nos proporciona Praat en Python.

   d) Estudio del análisis de audio con particularización al análisis de episodios de tos, además del estudio de los diferentes tipos de tos.

II. **Etapa de diseño, implementación y validación**, compuesta por las siguientes fases:

   a) Segmentación de las señales de audio, presentes en las bases de datos, con el objetivo de que todos los clips de audio tengan una duración de un segundo o que solo contengan el sonido de la tos a analizar.

   b) Análisis de la información, estudiando la distribución de energías de las señales segmentadas para la comprobación de ausencia de sesgo en el estudio.

   c) Adaptación de los datos con el objetivo de que puedan ser utilizados en la entrada de una red neuronal convolucional, ya que este tipo de redes, están pensadas para trabajar con imágenes. Para ello se ha utilizado el espectrograma de las señales segmentadas de audio, obteniendo una imagen de cada una de ellas.

   d) Diseño y definición de los parámetros de la red neuronal convolucional que va a funcionar como detector de audios de que contienen tos y núcleo central del sistema de diagnóstico.

   e) Entrenamiento de la red de forma independiente, con diferentes conjuntos de entrenamiento, validación y test.

   f) Analizar las prestaciones que nos ofrece nuestra red en términos de exactitud, sensibilidad, especificidad, área bajo la curva Característica Operativa del Receptor (Receiver Operating Characteristic, ROC), valor predictivo positivo y valor predictivo negativo en la detección de sonidos de tos y clasificación de enfermedades subyacentes según la tos.

## 1.4. Medios necesarios

En la realización de este proyecto, han sido necesarias diferentes herramientas de *hardware*, *software* y bibliográficas, siendo estas:

***Hardware***

- Ordenador de sobremesa con las siguientes características:
  - 8 procesadores Intel® Core™ i7-7700 CPU @ 3,60GHz
  - 16 GB de memoria RAM
  - Tarjeta gráfica NVIDIA GeForce GTX 1050 con 2GB de memoria RAM
- También se realiza una conexión remota a los servidores de cálculo disponibles en el Laboratorio de Procesado de Imagen (LPI) de la Universidad de Valladolid (UVa). Utilizandose el servidor *tebas*, con las características:
  - 70 procesadores Intel® Xeon® CPU E5-2697 v4 @ 2,30GHz
  - 500 GB de memoria RAM
  - Tarjeta gráfica NVIDIA GeForce GTX 1070 con 8 GB de memoria RAM

***Software***

- Kubuntu 18.04 : Sistema operativo, el cual es una distribución Linux basada en Ubuntu que utiliza como entorno de escritorio KDE.
- Anaconda : Distribución libre y abierta1 de los lenguajes Python y R, utilizada en ciencia de datos, y aprendizaje automático.
  - Python : Lenguaje de programación interpretado y multiparadigma, ya que soporta orientación a objetos, programación imperativa y, en menor medida, programación funcional.
  - Spyder : Entorno de desarrollo integrado (*Integrated Development Environment*, IDE) multiplataforma de código abierto para programación científica en el lenguaje Python.
  - Jupyter notebook : Entorno computacional interactivo basado en la web.
  - TensorFlow [5] : Biblioteca de código abierto para aprendizaje automático, y desarrollado por Google.
  - Keras [6] : Biblioteca de Redes Neuronales de Código Abierto escrita en Python capaz de ejecutarse sobre TensorFlow, Microsoft Cognitive Toolkit o Theano.
  - NumPy [7] : Extensión de Python, que le agrega mayor soporte para vectores y matrices, constituyendo una biblioteca de funciones matemáticas de alto nivel para operar con esos vectores o matrices.
  - SciPy [8] : Biblioteca libre y de código abierto para Python que contiene módulos para optimización, álgebra lineal, integración, interpolación, funciones especiales, transformada rápida de Fourier (*Fast Fourier Transform*, FFT), procesamiento de señales y de imagen, resolución de Ecuaciones Diferenciales Ordinarias (EDOs) y otras tareas para la ciencia e ingeniería.
  - Matplotlib [9] : Biblioteca para la generación de gráficos a partir de datos contenidos en listas o arrays en el lenguaje de programación Python y su extensión matemática NumPy.
  - Pandas [10] : Biblioteca de software escrita como extensión de NumPy para manipulación y análisis de datos para el lenguaje de programación Python.
  - Scikit-learn [11] : Biblioteca para aprendizaje de máquina de software libre para el lenguaje de programación Python, está diseñada para interoperar con las bibliotecas numéricas y científicas NumPy y SciPy.
  - tgt [12] : Paquete de Python para procesar, consultar y manipular los archivos TextGrid de Praat.

- Praat [13] : Software gratuito para el análisis científico del habla usado en lingüística.
- LaTeX[14] : Sistema de composición de textos, orientado a la creación de documentos escritos que presenten una alta calidad tipográfica.

***Bibliografía***

- Hemerotecas de la Escuela Técnica Superior de Ingenieros de Telecomunicación (ETSIT).
- Bases de datos electrónicas suscritas por la UVa.

En cuanto a los registros sobre los que se realizará el análisis y evaluación de los métodos, se hará uso de las bases de datos disponible dentro del grupo en el que se realiza el proyecto, el LPI. Estas bases de datos se describen con detalle en la sección 6.1.

## 1.5. Estructura del documento

El documento se encuentra estructurado en 6 capítulos, entre los que se incluye el presente capítulo. Su contenido se describe brevemente a continuación:

**Capítulo 1 - Introducción**. Se expone la temática del documento, así como la motivación, los objetivos perseguidos, las fases y medios necesarios para alcanzarlos y la estructura del presente documento.

**Capítulo 2 - Reconocimiento de patrones**. Introduce el termino de reconocimiento de patrones, Inteligencia Artificial y *Machine Learning*. Explica las diferentes técnicas de *machine hearing* y los principales clasificadores que se pueden utilizar.

**Capítulo 3 - *Deep Learning***. Se presenta una visión global del *Deep Learning* para posteriormente describir los diferentes tipos de redes que se pueden plantear con esta filosofía, haciendo hincapié en las redes neuronales convolucionales.

**Capítulo 4 - Técnicas para evaluar el rendimiento de un sistema**. Se realiza la descripción de las principales métricas para la evaluación del rendimiento del sistema y se explica la curva ROC, la validación cruzada y las formas de promediar los resultados.

**Capítulo 5 - Detección y análisis de la señal de tos**. En este capítulo se presentan los monitores de tos anteriores y las técnicas existentes de *machine hearing* para la detección de eventos de audio y, en particular, para la detección de eventos de tos de audio. Además, se muestran trabajos anteriores sobre la detecciones de enfermedades respiratorias.

**Capítulo 6 - Metodología**. En este capitulo se procede a la descripción tanto de las bases de datos utilizadas como de la composición de los *folds* en los entrenamientos y test de la red. Asimismo lo largo de este capítulo se detallan los procedimientos realizados para el preprocesado de las señales, referido este tanto a la adecuación de la información y preparación de la misma para la utilización de la red como a la definición, diseño y posterior entrenamiento de la red convolucional.

**Capítulo 7 - Resultados y discusión**. En este capítulo se procede al análisis y estudio de los resultados obtenidos tanto de la detección de clips de audios con tos como de la clasificación de audios con tos según la enfermedad subyacente y se realizan las correspondientes comparativas con lo conocido hasta el momento.

**Capítulo 8 - Conclusiones y líneas futuras**. Este capítulo final recoge las principales conclusiones extraídas con la elaboración de este Trabajo de Fin de Grado y se proponen posibles líneas de trabajo futuro manifiestas a partir de su realización.

# *Capítulo 2*

# RECONOCIMIENTO DE PATRONES

El objetivo principal de un sistema de reconocimiento automático de patrones es descubrir la naturaleza subyacente de un fenómeno u objeto, describiendo y seleccionando características fundamentales que permitan clasificarlo en una categoría determinada. Sistemas automáticos de reconocimiento de patrones permiten abordar problemas en informática, en ingeniería y en otras disciplinas científicas, por lo tanto el diseño de cada etapa requiere de criterios de análisis conjuntos para validar los resultados. Las principales áreas de aplicación son: reconocimiento remoto, reconocimiento óptico de caracteres y escritura manuscrita, identificación de patrones en imágenes médicas, sistemas de clasificación en bioinformática, sistemas de identificación biométrica y clasificación de series temporales.

Un modelo general clásico de un sistema automático esta constituido por tres etapas, sensor, extractor/selector de características, y clasificador. La primera etapa puede ser considerada a su vez como la que trata de obtener la representación más fiel del fenómeno estudiado, y un módulo que permite generar características del mismo. La etapa de generación de características aplica desde métodos simples directos hasta características obtenidas en forma indirecta de distintos tipos de señales como imágenes digitales 2D, 2,5D y 3D.

En este capitulo vamos a tratar de adentrarnos en el concepto de *Machine Learning*, debido a que en trabajos anteriores se han empleado estos algoritmos. Pero para esto primero debemos explicar que es la Inteligencia Artificial. En primer lugar debemos de tener claro que la Inteligencia Artificial es el concepto más general y una rama de la Inteligencia Artificial es el *Machine Learning*. Además en este capitulo también veremos que es el *machine hearing* y que características de los sonidos son las más relevantes.

## 2.1. Inteligencia Artificial

La Inteligencia Artificial (IA) esta destinada a marcar uno de los grandes avances en la Sociedad Digital de los próximos años siendo la IA una de las disciplinas más nuevas junto con la genética moderna. El termino de Inteligencia Artificial fue acuñado por primera vez en 1956 durante la conferencia convocada por John McCarthy y celebrada en el *Darmouth College* de New Hampshire (USA), en la cual participaron científicos de diversas ramas del saber. Aunque es en 1950 cuando el matemático británico Alan Mathison Turing, consolidó el campo de la Inteligencia Artificial con su artículo *Computing Machinery and Intelligence* [15], en el que propuso una prueba concreta para determinar si una máquina era inteligente o no, su famoso Test de Turing, por lo que se le considera el padre de la Inteligencia Artificial. Sin perjuicio de aproximaciones rigurosas procedentes de la Psicología se puede tomar como definición de lo que es inteligencia, la que se deriva implícitamente del Test de Turing, el cual postuló que una máquina presenta comportamiento inteligente si un observador es incapaz de distinguir, para un problema determinado, entre las sucesivas respuestas proporcionadas por la máquina y un ser humano.

La Inteligencia Artificial se define como una disciplina que se dedica a la construcción de programas informáticos capaces de realizar trabajos inteligentes. Sus objetivos básicos son dos: (1) estudiar el comportamiento inteligente de las personas humanas, y (2) hacer programas de ordenador inteligentes capaces de imitar el comportamiento humano [16]. Asimismo, cabe mencionar la definición atribuida a Marvin Minsky y recogida en [17],

que establece que la Inteligencia Artificial es el arte de construir máquinas capaces de hacer cosas que requerirían inteligencia en el caso de que fueran hechas por seres humanos [18].

La Inteligencia Artificial también se puede definir como procesos por los que una máquina imita las funciones cognitivas que los humanos asocian con otras mentes humanas, como por ejemplo aprender y resolver problemas [19]. No obstante, debido al gran desarrollo que está teniendo y las nuevas aplicaciones de la Inteligencia Artificial se ha generado una revisión de la definición en la que, según Stuart J. Russell y Peter Norvig [20], en la actualidad podemos establecer diferentes categorías de IA y distinguir entre:

- **Sistemas que piensan como humanos**: Estos sistemas tratan de imitar los procesos del pensamiento humano, por ejemplo las redes neuronales artificiales, y automatizan actividades que vinculamos con procesos de pensamiento humano, incluyendo aspectos como la toma de decisiones, la resolución de problemas o el aprendizaje automático.

- **Sistemas que actúan como humanos**: Estos sistemas informáticos imitan el comportamiento humano, mediante por ejemplo la robótica, siendo el objetivo del estudio de esta categoría lograr que los ordenadores realicen tareas que los humanos realizamos mejor.

- **Sistemas que piensan racionalmente**: Hablamos de un aprendizaje mucho más complejo, ya que estos sistemas, con lógica (idealmente), tratan de imitar o emular el pensamiento lógico racional del ser humano, un ejemplo son los sistemas expertos, siendo el objetivo del estudio de esta categoría los cálculos que hacen posible percibir, razonar y actuar.

- **Sistemas que actúan racionalmente**: Un paso mucho más allá en la evolución de la Inteligencia Artificial ya que hablamos de máquinas que tratan de emular de forma racional el comportamiento humano y son capaces de tomar decisiones en función de esos razonamientos, lo cual esta relacionado con conductas inteligentes en artefactos y siendo un ejemplo en esta categoría los agentes inteligentes.

La Inteligencia Artificial es multidisciplinar y se apoya en los conceptos y técnicas de otras disciplinas, tales como: la Informática, la Ingeniería, la Sociología, la Ciencia del Comportamiento, la Psicología Cognoscitiva, la Investigación Operativa, la Economía, la Teoría General de Sistemas, etc. También cabe destacar que la tecnología del Big Data se ha convertido en un pilar fundamental, ya que a partir del análisis, procesamiento e interpretación de millones de datos se lleva a cabo el desarrollo de la Inteligencia Artificial diariamente. Dentro del desarrollo de la Inteligencia Artificial cabe distinguir tres etapas [18]:

- **Etapa inicial (1956-1970)**: Etapa de creación de las técnicas básicas para representar el comportamiento inteligente tanto en el ámbito de los métodos de representación como de los lenguajes de programación.

- **Etapa de prototipos (1970-1981)**: En esta etapa se estudiaron las posibilidades de aplicación industrial de la Inteligencia Artificial.

- **Etapa de difusión (1981-??)**: A lo largo de los últimos años, la característica principal ha venido siendo la aceptación de la Inteligencia Artificial como una actividad informática profesional, de tal forma que desde 1981 las aplicaciones técnicas no son sólo desarrolladas por los laboratorios especializados, sino también por empresas que realizan aplicaciones para la industria y el mundo empresarial en general.

El desarrollo de la Inteligencia Artificial, en especial en los últimos años, ha sido exponencial, incluyendo aspectos como la predicción de comportamientos a partir de una serie de pautas programadas, siendo este el caso de la supercomputador Deep Blue [21] desarrollada por el fabricante estadounidense IBM que el 10 de febrero de 1996 venció al ajedrez a un campeón del mundo vigente, Garri Kímovich Kaspárov. En el momento actual la Inteligencia Artificial se aplica a numerosas actividades humanas, se han desarrollado procesos industriales y de fabricación, y como líneas de investigación más explotadas destacan el razonamiento lógico, la capacidad de las máquinas para el Procesamiento del Lenguaje Natural (PLN), la traducción automática y comprensión del lenguaje natural, las respuestas automatizadas (chatbots), el cuidado y detección de enfermedades, la educación, la robótica, la visión artificial y, especialmente, las técnicas de aprendizaje y de ingeniería del conocimiento [18].

## 2.2. *Machine Learning*

Ahora que tenemos una idea de que es la Inteligencia Artificial, en esta sección trataremos de explicar que es el *Machine Learning* (ML) o aprendizaje automático, ya que es la rama de la Inteligencia Artificial en la que se encuentran los algoritmos de *Deep Learning* en los que profundizaremos en el capitulo 4 y en donde el objetivo principal es que un sistema o máquina sea capaz de aprender y analizar información suministrada de ejemplo con el objetivo de predecir o generalizar ejemplos futuros, sin ningún tipo de intervención humana en el proceso. Dicho de otra forma, se trata de un conjunto de algoritmos que buscan los patrones por los cuales una información de origen se transforma en una información de destino [22].

La evolución de la Inteligencia Artificial hacia el concepto de *Machine Learning* se puede representar como el paso del reto de Deep Blue [21] de IBM para jugar al ajedrez, al de AlphaGo [23] de Google DeepMind para jugar al juego de mesa Go, que en octubre de 2015 se convirtió en la primera máquina de Go en ganar a un jugador profesional de Go. AlphaGo es capaz de imaginar mundos y proyectarlos a partir del análisis de millones de datos y variables del comportamiento humano en cuestión de segundos y por este motivo es por lo que representa el paso al concepto de *Machine Learning*.

Esta capacidad de aprender y de anticipar comportamientos tiene múltiples usos que van desde los más generalistas como sistemas de reconocimiento facial o la capacidad para la respuesta y el aprendizaje de idiomas hasta usos más especializados como la posibilidad de generar diagnósticos médicos o los sistemas que servirán de base a los vehículos autónomos [19].

El campo de aplicación del *Machine Learning* cada vez es mayor, aunque con frecuencia utilizamos aplicaciones que implementan algoritmos de *Machine Learning* sin saber que estos algoritmos están debajo. Así, está presente en la gran mayoría de motores de búsqueda de Internet, que se personalizan automáticamente según las preferencias del usuario; a la hora de explorar una base de datos de historiales médicos para predecir qué tipo de pacientes podrían responder a un cierto tratamiento; en el mundo de las redes sociales, donde gigantes como Facebook nos ofrecen determinada información en nuestro muro en función de nuestros gustos y el de nuestros amigos; o de los asistentes de voz virtuales como Siri, Cortana o Google Now, capaces de aplicar este tipo de patrones de aprendizaje para mostrarnos la información requerida a través del reconocimiento de voz [22].

## 2.3. *Machine hearing*

El objetivo general en varias disciplinas de ingeniería y ciencias de la computación ha sido dotar a las máquinas con capacidades de detección similares a las de los seres humanos, a través del aprendizaje automático. Si bien el aprendizaje automático aplicado a la visión (*i.e.*, la visión por computadora) ha recibido una atención creciente por parte de la comunidad de investigadores en los últimos años, su progresión para detectar señales acústicas aún está en su infancia [24].

El procesamiento de señales de audio como lo hacen los humanos se conoce con el nombre de "audición máquina" o *machine hearing* [24]. La "audición máquina" se enfoca en aplicaciones que involucran mezclas de sonido realistas en entornos reales. Por lo tanto, es una tarea compleja y desafiante debido a la amplia diversidad de posibles entradas de audio y escenarios de aplicaciones [25].

Las etapas que tiene un sistema de *machine hearing* son equivalentes a las que tiene un sistema de reconocimiento automático de patrones. Primero, un sensor recoge el flujo de audio y lo procesa previamente para facilitar el análisis posterior. Esto puede incluir encuadre, ecualización, eliminación de valores atípicos, eliminación de intervalos silenciosos, entre otras tareas.

El siguiente bloque es el extractor/selector de características y puede considerarse el corazón del sistema debido a su importancia primordial. Cualquier sistema de "audición máquina" debe extraer parámetros apropiados de la señal de audio que informen sobre sus rasgos notables, con el objetivo de aprovechar al máximo sus características particulares. La tarea de extracción se identifica como cálculo de características.

Las características calculadas se alimentan entonces de un módulo de decisión que suele ser un clasificador entrenable. Aprende una asignación de las características calculadas en la etapa anterior a las decisiones que necesita la aplicación. Finalmente, algunos sistemas incluyen un módulo de procesamiento posterior que mejora la salida del clasificador. Este módulo a menudo aprovecha ciertos conocimientos *a priori* sobre el comportamiento

del sistema, por ejemplo, la interpretación física de los patrones de audio o una distribución estadística *a priori* de las clases. Algunos ejemplos de tareas de posprocesamiento son la combinación ingenua de etiquetas de clase de salida o la fusión basada en Viterbi de probabilidades de salida para mejorar las capacidades de segmentación [26].

Derivado de la complejidad mencionada anteriormente, el *machine hearing* se subdivide típicamente en subproblemas más pequeños, y los esfuerzos de investigación se centran en resolver tareas específicas. Esta división se realiza generalmente en base a la señal de audio de interés. Desarrollar un sistema capaz de manejar señales de una naturaleza particular es más factible que crear un sistema genérico de *machine hearing* que pueda hacer frente con éxito a diferentes tipos de sonidos [25].

La primera señal que llamó la atención en la audición de máquina fue el habla. Estos sistemas, comúnmente conocidos como detectores de actividad de voz, se enfrentaron a un problema de clasificación binaria para diferenciar los eventos de habla y no habla. Sin embargo, esta actividad sigue siendo un área de investigación activa [27]. Su complejidad y rendimiento aumentaron con el tiempo y la literatura actualmente contiene un montón de trabajo relacionado con las señales del habla: diarización del hablante [28], verificación del hablante [29], mejora del habla [30], desverberación del habla [31] o análisis patológico del habla [32], por ejemplo.

Más adelante, la generalización de otras fuentes de información, como la transmisión de audio/vídeo o los repositorios musicales, introdujo la música como la señal dirigida a la audición de la máquina [33]-[36]. El tipo de aplicaciones se ha diversificado desde que se puede encontrar una amplia variedad de datos en la actualidad, incluida la identificación de canciones [37], la clasificación de género [38], el reconocimiento de instrumentos [39], la consulta por zumbido [40], o los sistemas de recomendación [41], entre otros.

A pesar de la gran cantidad de técnicas y metodologías para el procesamiento de voz y música, estas señales presentan características distintivas que complican la aplicación de sus métodos a otras señales de audio. Algunos ejemplos de estas características bastante únicas son la distribución espectral y la estructura fonética del habla o las estructuras de patrones estacionarios repetidos como melodía y ritmo en la música. Por otro lado, otro tipo de fuentes de sonido provenientes del medio ambiente o los sistemas biológicos no exhiben tales particularidades, o al menos no de una manera tan clara [25]. El análisis de estos sonidos lleva a otras aplicaciones de audición de máquina como el reconocimiento de sonidos ambientales [42], la clasificación de escenas acústicas [43], el análisis de sonidos pulmonares [44] o los sistemas de vigilancia basados en audio [45].

El diseño de las características de acuerdo con la naturaleza de la señal y la aplicación específica es el punto crítico para garantizar el éxito de un sistema de *machine hearing*. Estas características deben proporcionar una parametrización compacta pero descriptiva de la señal. Se pueden usar muchos criterios para categorizar las características de audio, pero una primera división amplia es entre aproximaciones físicas y perceptivas [25], [46]. Posteriormente, los subdividimos según el dominio empleado y la propiedad que se caracteriza. Los principales clasificadores empleados en el *machine hearing* se presentan en la Sección 2.4.

### 2.3.1. Características físicas del audio

Estas características operan en varios dominios y caracterizan los parametros físicos de la entrada de audio. A continuación hablamos de cada uno de los dominios.

#### Dominio del tiempo

Por un lado, el dominio clásico del tiempo tiene la ventaja de que no requiere una transformación de la señal, pero, por otro lado, cualquier contaminación leve cambia la forma de onda y, por lo tanto, afecta la caracterización. El parámetro físico en el que se basan es lo que diferencia las características físicas del dominio del tiempo:

- **Basado en tasa de cruce por cero**: se basan en el análisis de la tasa de cambios en los signos. Aunque se calculan en el dominio del tiempo, proporcionan un estimador aproximado de la componente de frecuencia dominante de la señal. Se han utilizado para múltiples propósitos, desde la discriminación de voz/música [47] hasta la vigilancia de audio [45].

- **Basado en amplitud**: simplemente analizan la envolvente temporal de la señal. El estándar MPEG-7 (*Moving Picture Experts Group*) describe algunas de estas características que se han utilizado para el reconocimiento del sonido ambiental en el hogar [48], por ejemplo. Además, la característica de brillo calcula la

variación ciclo a ciclo de la amplitud de la forma de onda, que ha sido útil para la clasificación de regiones vocales y no vocales en las canciones [49] o en el análisis patológico del habla [50].

- **Basado en la energía**: emplean la potencia de la señal. La energía de tiempo corto, el volumen (valor cuadrático medio de la magnitud de la forma de onda dentro de un *frame*), el centroide temporal MPEG-7 o el tiempo de ataque de registro MPEG-7 son algunos ejemplos [51]. Se han utilizado para la detección de inicio de música [25] o la detección de transición de *frames* sin voz a con voces y viceversa [52].

- **Basado en el ritmo**: el ritmo representa las organizaciones estructurales de los eventos acústicos a lo largo del eje temporal. Existen varias formas de explotar estos parámetros físicos. La claridad del pulso es un descriptor de alto nivel que transmite información sobre la facilidad con que un oyente aprecia la pulsación rítmica subyacente. La periodicidad de la banda mide la fuerza de las estructuras repetitivas en las señales de audio. El espectro de ritmo cíclico proporciona una representación compacta del período de ritmo fundamental. El espectro rítmico refleja los cambios temporales del tempo. Estas características se han aplicado ampliamente para el análisis de la música [25].

**Dominio de la transformada de Fourier**

Estas características describen las propiedades físicas del contenido de frecuencia de la señal. Hay varios subtipos:

- **Basado en autocorrelación**: se derivan del análisis de predicción lineal de la señal, que captura formas o resonancias espectrales que aparecen en el tracto vocal. Las frecuencias espectrales de línea (*Line Spectral Frequencies*, LSF) son una versión robusta de LPC (*Linear Prediction Cepstral*). Finalmente, las funciones de predicción lineal con código excitado combinan características espectrales como la LSF y los coeficientes del libro de códigos relacionados con el tono de la señal. Estas características las hacen adecuadas para el análisis del habla y la música [25].

- **Relacionado con la forma del espectro**: este conjunto de características intenta describir la forma del espectro [25]. El ancho de banda espectral se entiende como la dispersión del espectro. Ayuda a discriminar el sonido tonal de los sonidos similares al ruido. El estándar MPEG-7 define también una medida llamada propagación del espectro de audio [51]. La reducción espectral se define como el percentil 85 de la densidad espectral de potencia (*Power Spectral Density*, PSD). La planitud espectral es una medida de la uniformidad de la PSD. El factor de cresta espectral mide el pico del espectro de potencia. El centroide espectral describe el centro de gravedad de la energía espectral. De nuevo, una medida basada en el centroide espectral se define en MPEG-7 [51]. El flujo espectral se define como la norma L2 de las diferencias de amplitud espectral *frame* a *frame*, que describe cambios repentinos en la distribución de energía de frecuencia. La entropía del pico espectral es una medida de entropía basada en los picos y valles del espectro [53]. La relación f50 vs f90 define la relación entre las frecuencias a las que se incluye el 50 % y el 90 % de la energía [53]. La entropía espectral de Renyi es una medida generalizada de aleatoriedad [54]. Las estadísticas como la desviación estándar, la curtosis o la asimetría también se pueden calcular [25]. La relación de energía de subbanda es una medida de la energía de señal normalizada a lo largo de bandas de frecuencia predefinidas [25]. Dada la amplia aplicabilidad de estos descriptores, se han utilizado en una variedad de aplicaciones de audición de máquina y otras áreas como el análisis de señales biomédicas [53], [54].

- **Relacionado con la envolvente del espectro**: estas características se calculan a partir del análisis de la envolvente del espectro. NASE (*Normalized Audio Spectral Envelope*) [51], [55] se define en el estándar MPEG-7 como un espectro de potencia de frecuencia logarítmica normalizado que se puede usar para generar un espectrograma reducido de la señal original. El contraste espectral por octava es la diferencia entre los picos, que en general corresponde al contenido armónico y los valles (componentes no armónicos o de ruido) medidos en subbandas con un filtro de escala de octava utilizando un criterio de vecindad en su cálculo. Su uso principal es para análisis de música [56].

- **Relacionado con la tonalidad**: la tonalidad se relaciona con la noción de armonicidad, que describe la estructura de los sonidos que están constituidos principalmente por una serie de frecuencias relacionadas

armónicamente [25]. Su caracterización es muy útil en el análisis musical. La frecuencia fundamental, denominada como F0, se define en el estándar MPEG-7 como el primer pico prominente de la función de autocorrelación normalizada. Existen muchos algoritmos para calcularla [47]. El tono es la estimación de la F0 percibida de la señal. En la literatura científica, el tono a menudo se usa indistintamente con F0. El histograma de tono proporciona una representación compacta del contenido de tono, mientras que el perfil de tono es una representación más precisa de la estructura tonal. Se han utilizado para la detección del tono en el análisis musical, entre otros usos [57]. La armonía es una característica útil para distinguir entre sonidos tonales y sonidos similares al ruido. En el estándar MPEG-7, la relación armónica es una medida de la proporción de componentes armónicos en el espectro de potencia [25]. La forma más sencilla de calcularlo es mediante una estimación F0 basada en autocorrelación [47], aunque existen algoritmos para señales específicas como la voz [58]. El estándar MPEG-7 proporciona otros descriptores de timbres como el centroide espectral armónico, la desviación espectral armónica o la dispersión espectral armónica [51]. Los descriptores de timbre espectral MPEG-7 se han utilizado en el reconocimiento del sonido ambiental [59] o en el reconocimiento de la emoción del habla [60]. Finalmente, la fluctuación de fase es la variación ciclo a ciclo de la frecuencia fundamental [25]. Ha sido útil en el análisis patológico del habla [50] o el reconocimiento del hablante [61], por ejemplo.

- **Relacionados con el croma**: en el análisis de música, dos notas que están separadas una o más octavas tienen el mismo croma y producen un efecto similar en el sistema auditivo humano [25]. El cromagrama es una representación de 12 elementos de la energía espectral [47] relacionada con el tono. Tiene en cuenta las doce clases de tono dentro de una octava según la teoría musical. Constituye por tanto una representación muy compacta del espectro energético. El cromagrama básico se mejoró posteriormente para aplicaciones de audición de máquinas específicas, como la coincidencia de similitud musical [62].

#### Dominio de transformación de wavelet

Una wavelet es una función corta en forma de onda que puede ser escalada y trasladada. Las transformadas de wavelet toman cualquier señal y la expresan en términos de wavelets escalados y traducidos para que la señal se represente en diferentes escalas. Esta transformación permite manipular entidades a diferentes escalas de forma independiente. Esta transformación se realiza de manera diferente dependiendo de la función de wavelet empleada. Por ejemplo, las wavelets de Daubechies se han empleado para mejorar el habla [63], la marca de agua de audio robusta [64] o el reconocimiento de emoción musical [65]. La wavelet de Haar se ha utilizado para indexar secuencias melódicas [66] o para la compresión con pérdida de señales de ecolocación [67]. Otras aplicaciones de la teoría de wavelets en audición automática son la descomposición del habla [68] o la codificación de audio [69] a través de paquetes de wavelets. Finalmente, el parámetro Hurst es una representación estadística de la frecuencia vocal de la fuente vocal derivada de una transformación multidimensional basada en wavelet [25], [70]. La mejora del habla [71] o la localización de la fuente de sonido en entornos ruidosos [72] se encuentran entre sus aplicaciones.

#### Búsqueda coincidente

La búsqueda coincidente tiene como objetivo descomponer cualquier señal en una expansión lineal de formas de onda que se seleccionan de un diccionario de funciones redundantes (también llamadas átomos) [73]. Para seleccionar los "mejores" átomos, la energía extraída del residuo se maximiza en cada paso del algoritmo. Esta descomposición dispersa, permite obtener una aproximación razonable de la señal con unas pocas funciones básicas, lo que proporciona una interpretación de la estructura de la señal [25]. Hay varios conjuntos de características basadas en este enfoque:

- **Características basadas en Gabor**: la idea fue propuesta por Wolfe *et al.* y consiste en utilizar el análisis de Gabor para crear diccionarios adecuados para señales de voz y música [74]. Se utilizaron principalmente para el procesamiento del habla [75] y, más recientemente, para la clasificación del sonido ambiental en domótica [76].

- **Representación del tensor de codificación dispersa**: es una versión mejorada del cálculo de la característica de búsqueda coincidente propuesta por Chu *et al.* [25], [42]. Las tres dimensiones del tensor representan

la frecuencia, el tiempo y la escala de los componentes de frecuencia de tiempo transitorios. El método permite preservar el carácter distintivo de los átomos seleccionados por el algoritmo de búsqueda coincidente. Se ha utilizado para realizar la clasificación de efectos de sonido [77].

#### Dominio de la imagen

La idea subyacente de este enfoque es el uso de métodos de procesamiento de imágenes para caracterizar representaciones en 2D de una señal de audio, por lo general, una representación de tiempo-frecuencia. Uno de los trabajos pioneros [78] utilizó características de imagen del espectrograma (*Spectrogram Image Features*, SIF) para la clasificación de eventos de sonido. En el método SIF, el espectrograma se normaliza en escala de grises y su rango dinámico se cuantifica en regiones antes de dividirlo en bloques cuyas estadísticas de distribución (momentos de segundo y tercer orden) se extraen para construir el conjunto de características correspondiente. La principal desventaja de este enfoque es la gran dimensión del conjunto de características (486). Este inconveniente fue parcialmente resuelto por Sharan y Moir, quienes mejoraron el enfoque básico de SIF para una vigilancia de audio robusta. Redujeron la dimensión del espacio de la característica a 216 al calcular la media y la desviación estándar de las estadísticas de distribución en filas y columnas [79].

#### Dominio cepstral

Estas características proporcionan una aproximación suave del espectro basado en la magnitud logarítmica. Se han utilizado en gran medida en audición de máquina, desde en la identificación de la persona que habla en un audio hasta en la obtención del contexto de un audio [25]. La característica más básica del dominio cepstral es el cepstrum complejo. Se define como la transformada de Fourier inversa del logaritmo del espectro. Los LPCC (*Linear Prediction Cepstral Coefficients*) [80] se diseñaron para el reconocimiento de voz y la identificación del hablante [81], aunque también se han utilizado para otros fines, como la clasificación de música [82], la clasificación de escenas acústicas [43] o el reconocimiento de sonido ambiental [42].

#### Dominio del espectro de alto orden

La estimación del espectro de potencia ha resultado esencial en muchos problemas de procesamiento de señales (*e.g.*, radar y sonar, análisis de señales biomédicas o habla). Sin embargo, los métodos de estimación del espectro "tipo Fourier" explotan principalmente la información presente en la secuencia de autocorrelación y, por lo tanto, descartan las relaciones de fase entre los componentes de frecuencia. Dicha estimación es suficiente para la descripción estadística completa de una señal gaussiana, pero hay situaciones prácticas en las que se requiere un análisis más allá del espectro de potencia de una señal [83].

Los espectros de alto orden resuelven este problema extrayendo información con respecto a: (1) desviaciones de gaussianidad; (2) presencia de relaciones de fase; (3) propiedades no lineales [83]. Las características basadas en el espectro de alto orden han resultado útiles para el análisis patológico del habla [32].

#### Espacio de fase

Se usaron dinámicas no lineales en un intento de modelar los efectos no lineales que ocurren durante la producción del habla. Lindgren *et al.* propusieron un conjunto de características que proviene de un espacio de fase reconstruido generado a partir de versiones retrasadas de la serie de tiempo original [84]. Se utilizó un enfoque similar para la eliminación de ruido en el habla en [85] o para la evaluación patológica de los nódulos de las cuerdas vocales del habla en [86].

#### Espacio propio

Este enfoque se basa en algoritmos de reducción de dimensionalidad como el análisis de componentes principales, la descomposición de valores singulares o el análisis de componentes independientes. Se utilizan para obtener una representación compacta de la información de la señal principal. El estándar MPEG-7 contiene dos descriptores de *eigenspace*: proyección de espectro de audio y base de espectro de audio [25], [51].

### 2.3.2. Características perceptivas del audio

Las características perceptivas o prosódicas intentan extraer información con significado semántico en el contexto de los oyentes humanos [46]. El desarrollo de las características perceptivas se realiza a través de una parametrización que describe la percepción humana, o a través del cálculo de características que son capaces de extraer aspectos perceptualmente relevantes. Como las características físicas, están presentes en diferentes dominios [25].

#### Dominio del tiempo

Hay dos subtipos en el dominio temporal:

- **Amplitud pico de cruce por cero**: la señal se descompone primero en varias bandas psicoacústicas y luego se calculan los cruces por cero para cada subbanda. Finalmente, se construye un histograma utilizando las longitudes inversas de cruce por cero en todas las señales de subbanda. Fueron diseñados para un reconocimiento de voz robusto [87]. Más tarde, se mejoraron al explotar un banco de filtros espaciados por frecuencia de Mel [88] para simular sistemas nerviosos auditivos, lo que resultó en amplitudes de pico de cruce de punto sincrónico [89].

- **Función de autocorrelación**: la función de autocorrelación es una medida de auto-similitud en el dominio del tiempo y se calcula mediante un conjunto de parámetros basados en la percepción relacionados con fenómenos acústicos [25]. Estas características se utilizaron para la representación del habla [90]. El concepto se mejoró más tarde aplicando un banco de filtros Mel o un banco de filtros Gammatone [91].

#### Dominio de la transformada de Fourier

Los diferentes subtipos se basan en las propiedades de percepción que están representadas por cada característica:

- **Basado en la modulación**: el contenido de la modulación es información de baja frecuencia (alrededor de 20 Hz) que produce variaciones tanto de amplitud como de frecuencia. Estas variaciones son más claras en las señales rítmicas y pueden reflejar la evolución estructural a lo largo del tiempo del contenido de frecuencia de un sonido. El espectrograma de modulación es una de las características más representativas de este grupo. Describe la señal en términos de la distribución de modulación lenta en tiempo y frecuencia [25]. Fue diseñado para la representación de voz [92] y se ha utilizado para diferentes tareas de procesamiento de audio, por ejemplo, clasificación mediante la relación señal a ruido (*Signal to Noise Ratio*, SNR) de la voz [93], clasificación de llantos patológicos infantiles [94] o separación ciega de mezclas de audio [95].

- **Análisis de modulación a largo plazo de las características del timbre a corto plazo**: este enfoque fue pensado para explotar las propiedades acústicas musicales. Consiste en un análisis espectral de modulación de descriptores de corto plazo como los Coeficientes de Cepstral de Frecuencia de Mel (*Mel Frequency Cepstral Coefficients*, MFCC) [96] y características de timbre. El análisis de la música es su principal aplicación [56].

- **Relacionado con el brillo**: se puede considerar la contraparte perceptiva de los centroides espectrales. La nitidez es una medida de la intensidad de la señal para altas frecuencias que resulta útil para la clasificación según la similitud del audio [97].

- **Relacionado con la tonalidad**: la tonalidad explica la percepción subjetiva de los armónicos de la señal. Permite distinguir los sonidos similares a los ruidos de los sonidos relacionados con la armonía. A diferencia de las características físicas basadas en el tono, las características de percepción basadas en el tono se basan en modelos auditivos [25]. Un banco de filtros de paso de banda basado en Gammatone enfatiza las bandas de frecuencia más relevantes para la percepción del tono [98]. Este banco de filtros imita la selectividad de frecuencia de la cóclea [99].

- **Relacionado con la sonoridad**: la sonoridad es una noción de la impresión subjetiva de la intensidad del sonido [25]. Basado en un análisis de frecuencia a escala de Bark y un efecto de enmascaramiento espectral que emula el sistema auditivo humano, esta característica proporciona una sensación de sonoridad. Ha sido útil para el procesamiento de música [100]. Otros enfoques miden la sensación de carga humana mediante la integración de la sonoridad en varios grupos de frecuencias, lo que permitió discriminar los sonidos de primer plano y de fondo en las escenas de video [101].

- **Relacionadas con la rugosidad**: la rugosidad se define como la sensación psicoacústica para una rápida variación de amplitud que reduce el agrado sensorial [25]. Matemáticamente, es la percepción de modulaciones de envolvente temporal en el rango [20, 50] Hz. Las características de rugosidad generalmente dependen de la envolvente temporal a través de la transformación de Hilbert y los bancos de filtros de Gammatone. La clasificación musical está entre sus usos [102].

#### Dominio de transformación de wavelet

Hay varios subtipos:

- **Coeficientes de orientación del flujo de potencia del *kernel***: es una caracterización del flujo de potencia de un espectrograma auditivo basado en Gammatone a través de *kernels* 2D [25]. Se usó para el reconocimiento automático de voz, con un buen rendimiento especial en escenarios ruidosos y reverberantes [103]. Dado que realiza un análisis 2D de una representación de tiempo-frecuencia, este conjunto de características también podría considerarse en el dominio de la imagen.

- **Coeficientes de wavelets discretos de frecuencia de Mel**: este conjunto de características explica la respuesta perceptiva del oído humano al aplicar la transformada de wavelet discreta a las energías del banco de filtros de registro de escala Mel de la señal de audio. Originalmente fue creado para mejorar el reconocimiento de voz [25] pero a lo largo del tiempo se ha aplicado a otros problemas de audición de la máquina [45]. Este conjunto de características también podría incluirse en el dominio cepstral.

- **Características de Gammatone wavelet**: las funciones madre típicas de wavelet son reemplazadas por las funciones de Gammatone, que modelan el sistema auditivo. Estas características han mostrado un mejor rendimiento que las wavelets tradicionales tanto en condiciones silenciosas como ruidosas [25]. Se han utilizado en la vigilancia de audio [104], entre otros usos.

- **Paquetes wavelet perceptuales**: es una versión optimizada perceptualmente de paquetes wavelet clásicos. Se han empleado en diferentes aplicaciones, desde el reconocimiento de voz hasta el reconocimiento de eventos de sonido de llanto [25].

#### Dominio espectro-temporal multiescala

La hipótesis de estos enfoques es que al imitar las mediciones en la corteza auditiva primaria de diferentes animales, lo que revela la organización espectro-temporal de los campos receptivos, podemos mejorar el análisis de audio. En las características de modulación espectro-temporal multiescala, el espectrograma auditivo, que simula la representación neuronal de la señal de audio de entrada, se analiza para estimar el contenido de sus modulaciones espectrales y temporales utilizando un banco de filtros selectivos de modulación, que imitan a los descritos en el modelo de la corteza auditiva primaria de los mamíferos. Estas características se han aplicado principalmente a las señales de voz y música [25].

#### Dominio de la imagen

Las características de audio basadas en imágenes se amplían para incluir aspectos de los modelos perceptivos. Los diferentes subtipos son:

- **Característica de imagen del espectrograma**: estas características se derivan de las parametrizaciones de *front-end* que hacen uso de modelos psicoacústicos. Por ejemplo, en [105] un espectrograma de filtro Mel de 40 dimensiones se caracteriza por detectar picos de alta energía que se correlacionan con un diccionario de libro de códigos para entrenar una red neuronal en la clasificación de eventos de sonido. Otro ejemplo sería el trabajo [106], en el que se calculan los momentos de Hu sobre subbloques de energía obtenidos del espectrograma en la escala de frecuencias de Mel.
- **Imágenes de tiempo-croma**: es una representación 2D que traza la distribución de croma a lo largo del tiempo. La definición de croma se basa en los tonos que están separados por n octavas [107]. Se ha utilizado en la identificación de canciones, por ejemplo.

#### Dominio cepstral

Existen dos subtipos dentro de las características de percepción cepstral:

- **Basado en el banco de filtros**: este enfoque se basa en el uso de la descomposición específica del banco de filtros con algunos criterios de percepción posibles. El subtipo comprende MFCC y sus variantes, que a menudo modifican la escala de frecuencia. Por ejemplo, Coeficientes Cepstral de Gammatone (*GammaTone Cepstral Coefficients*, GTCC) [96], [108] emplea un banco de filtros basado en funciones de Gammatone lineales de orden 4 que modelan la respuesta espectral auditiva humana [108]. Otros, como el histograma centroide de subbanda espectral (*Spectral Subband Centroid Histogram (SSCH)*, SSCH) [109], se basan en la escala de Bark para explotar la información de frecuencia dominante de una manera simple y computacionalmente eficiente para la detección de voz robusta [109].
- **Basada en la autorregresión**: estas características incorporan análisis predictivo lineal dentro del marco basado en cepstral. Un ejemplo representativo de este grupo es la predicción lineal espectral-perceptiva relativa. Este conjunto de características incorpora habilidades similares a las humanas para ignorar el ruido del habla al filtrar cada canal de frecuencia con un filtro de paso de banda que mitiga las variaciones de tiempo lento. También utiliza compresión estática no lineal y bloques de expansión antes y después del procesamiento de paso de banda. Se ha utilizado principalmente para el procesamiento del habla [110].

## 2.4. Clasificadores

Como se describió en las secciones anteriores, la creación de características adecuadas es el corazón de la investigación en *machine hearing* [47]. Sin embargo, tales características son necesarias pero insuficientes para lograr el objetivo final del sistema de *machine hearing*. Por lo general, alimentan un sistema entrenable para realizar algún tipo de clasificación [26], [34], [111], por ejemplo, segmentación de eventos de audio, indexación de canciones o recuperación de contenido de audio.

La clasificación de audio es otra etapa importante en el procesamiento de la señal de audio. En particular, la primera tarea de clasificación de este TFG se denomina detección de eventos de audio o segmentación de eventos de audio [46]. La detección de eventos de audio está dirigida a detectar señales de audio específicas dentro de un flujo de audio largo y desestructurado. La clasificación, que es la segunda tarea de este TFG, consiste en estimar la etiqueta de clase de un segmento de audio que está representado por un vector de características [111] dentro de un espacio de características. Los enfoques de clasificación se pueden dividir en tres tipos: supervisados, semisupervisados y no supervisados [46].

La metodología supervisada se basa en la utilización de etiquetas de clase predefinidas para crear un modelo de clasificación que asigna automáticamente una etiqueta de clase a observaciones desconocidas. Khunarsal *et al.* propuso un algoritmo de clasificación de sonido ambiental utilizando el patrón de espectrograma que coincide con la red neuronal y los clasificadores de k vecinos más cercanos (*K-Nearest Neighbours*, K-NN) [112]. Mitra y Wang propusieron una técnica de clasificación de audio basada en el análisis de contenido de audio y perceptrones multicapa (*MultiLayer Perceptrons*, MLP). Resolvieron un problema de clasificación de género musical [113]. Costa *et al.* se enfrentó igualmente a un problema de clasificación de género musical, pero utilizaron características de las texturas de los patrones binarios locales y una máquina de vectores de soporte (*Support Vector Machine*,

SVM) [114]. Se utilizaron clasificadores más simples, como el análisis discriminante lineal (*Linear Discriminant Analysis*, LDA) en [115] para el reconocimiento automático de vocalizaciones de animales. Los clasificadores de conjunto también están presentes en la audiencia de máquina. Por ejemplo, Dafna *et al.* usó un clasificador AdaBoost para detectar eventos de ronquidos [116].

En el aprendizaje no supervisado (también denominado agrupamiento), no hay etiquetas de clase predefinidas disponibles. El objetivo es explorar los datos y detectar similitudes entre las observaciones del espacio de características. Pomponi y Vinogradov crearon un algoritmo de agrupación en tiempo real en el que el número de agrupaciones y sus elementos se deducen de la distribución de datos. Lo aplicaron para la clasificación de emisiones acústicas [117]. Saki y Kehtarnavaz aplicaron un algoritmo de agrupamiento novedoso a los sonidos ambientales [118]. Choi y Chang mejoraron la mejora del habla basándose en la clasificación del entorno acústico [119].

El aprendizaje semisupervisado es el punto medio entre los dos últimos enfoques. La capacitación se basa en una cantidad reducida de datos etiquetados y una gran cantidad de datos sin etiquetar. La combinación de ambos tipos de datos (etiquetados y no etiquetados) tiene como objetivo mejorar el rendimiento de clasificación y superar la limitada disponibilidad de datos etiquetados. Neiberg *et al.* usó métodos semisupervisados para explorar la acústica de retroalimentación productiva simple [120]. Song y Zhang realizaron una clasificación de género musical, basada en un *framework* de fusión de información de contenido que alimentaba un clasificador de distancia semisupervisado [121]. Deng *et al.* propusieron auto-codificadores semi-supervisados para mejorar el reconocimiento de la emoción del habla [122].

El resto de esta sección está dedicado a proporcionar una descripción de alto nivel de los clasificadores supervisados más comúnmente utilizados en "audición máquina", exceptuando los MLP de los cuales hablaremos en la sección 4.2.1, ya que es un algoritmo que está más relacionado con *Deep Learning*.

#### Máquina de vectores de soporte (*Support Vector Machine*, SVM)

Las SVM son herramientas fundamentales en sistemas de aprendizaje automático [111], que se han empleado con éxito en varios campos, permitiendo el tratamiento de problemas actuales en reconocimiento de patrones y minería de datos tales como, reconocimiento y caracterización de texto manuscrito, detección ultrasónica de fallas en materiales, clasificación de imágenes médicas, sistemas biométricos, clasificación en bioinformática y en física de altas energías [123].

Las SVM implementan reglas de decisión complejas, por medio de una función de *kernel* no lineal que permite mapear los puntos de entrenamiento a un espacio de mayor dimensión. En el nuevo espacio de características las clases son separadas por un hiperplano, siendo este el que maximiza la distancia entre el mismo y los puntos de entrenamiento. La distancia se denomina margen y esos vectores son los vectores de soporte. El enfoque matemático que utiliza los *kernels* puede usar algoritmos de solución y cálculo idénticos, y el método computacional de los hiperplanos solo se basa en productos de puntos. En este caso, los límites de decisión resultantes son hipersuperficies en algún espacio de alta dimensión.

Las SVM cumplen un rol muy importante en teoría de aprendizaje estadístico y cuando es necesario entrenar un clasificador no lineal en un espacio de características de considerable dimensión con un número limitado de muestras. Podemos diferenciar dos aspectos importantes que en general reciben la denominación de maquinas de soporte vectorial, el uso de SVM en clasificación SVC y el uso de las mismas en regresión SVR [123].

#### K vecinos más cercanos (*K-Nearest Neighbours*, K-NN)

Este es el otro gran clasificador de audición de máquina [111]. Además, es uno de los algoritmos de aprendizaje automático más simples. A diferencia de SVM, está bien adaptado para problemas binarios y de clases múltiples. La razón detrás de K-NN es que un vector de características desconocidas se asigna a la clase más común entre sus k-vecinos más cercanos. Es un método no paramétrico sin una fase de entrenamiento en sentido estricto. Las observaciones del entrenamiento son almacenadas y utilizadas directamente por el algoritmo en la etapa de clasificación [46] que estima la probabilidad a posteriori de que un elemento pertenezca a una clase concreta. El K-NN funciona de la siguiente manera para clasificar un patrón desconocido $v$:

1. Los vectores de características para entrenamiento y sus etiquetas correspondientes son almacenados.

2. El algoritmo calcula la distancia entre $v$ y los vectores de características en el conjunto de entrenamiento.
3. Los valores de distancia resultantes se clasifican en orden ascendente. Los primeros valores de k corresponden a los k vecinos más cercanos del vector de características desconocido.
4. El vector desconocido se clasifica a la clase más común entre estos k vecinos. Este paso puede entenderse como si los k vecinos más cercanos tuvieran 1/k de peso y los otros tuvieran peso nulo.

Desde una perspectiva Bayesiana, el algoritmo K-NN asigna el patrón desconocido $v$ de prueba a la clase que corresponde a la probabilidad posterior máxima estimada. A medida que el número de observaciones de entrenamiento y los vecinos más cercanos se acercan al infinito, el error de clasificación se acerca al error Bayesiano [111]. En consecuencia, cuanto mayor sea el conjunto de datos, mejor será el rendimiento del clasificador K-NN.

**Análisis discriminante lineal (*Linear Discriminant Analysis*, LDA)**

El LDA es una técnica que permite transformar la observación de entrada en un nuevo espacio de características en el que la clasificación es más robusta [46]. LDA maximiza la relación (separación entre clases)/(separación dentro de clases). Se ha utilizado tanto para la clasificación [124] como para la reducción de dimensionalidad [115] en aplicaciones de audición de máquina.

**Árboles de clasificación y regresión (*Classification And Regression Trees*, CART)**

Los CART, que son algoritmos de árboles de decisión, se han utilizado ampliamente en los campos del aprendizaje automático y la minería de datos a lo largo de los años. Los árboles de decisión, donde la variable de destino puede tomar valores continuos (por lo general números reales) se llaman árboles de regresión. Los modelos de árbol, donde la variable de destino puede tomar un conjunto finito de valores se denominan árboles de clasificación. Nos centramos en estos últimos, y vemos como funciona el algoritmo para crear dichos arboles [111]:

1. Cada nodo interno (nodo no hoja) o nodos de decisión en el árbol denota una prueba en un atributo, cada rama representa el resultado de una prueba. En consecuencia, el espacio de características se divide a medida que se construye el árbol. El nodo superior en un árbol es el nodo raíz.
2. El punto anterior se repite para generar nodos internos secundarios. El proceso se detiene cuando la división de los nodos ya no mejora la clasificación.
3. El último nodo es nodo hoja, también denominados nodos respuesta o terminal se le asigna una etiqueta de clase que esta asociada a la clasificación que se quiere proporcionar, por lo que devuelve la decisión respecto al ejemplo de entrada.

En el momento de la prueba, una muestra se clasifica navegando por el árbol hasta que se alcanza una hoja (decisión de clasificación final), en función del resultado de las "respuestas" en los nodos internos [111]. La principal ventaja de este algoritmo de aprendizaje automático es la falta de suposiciones con respecto a las distribuciones de características (un enfoque no paramétrico). Además, los árboles de clasificación simples son fáciles de interpretar al convertir su estructura en un conjunto de reglas. Pueden manejar problemas de clase múltiple también. Los dendrogramas son muy útiles en este sentido. Por otro lado, exhiben una alta sensibilidad a las entradas ruidosas [111].

# *Capítulo 3*

# TÉCNICAS PARA EVALUAR EL RENDIMIENTO DE UN SISTEMA

En este capítulo vamos a ver las técnicas para evaluar el funcionamiento de un sistema, que hemos utilizado en este trabajo para evaluar el sistema de detección y clasificación propuesto. Este capitulo es importante, ya que, cuando se diseña un sistema de reconocimiento de patrones es muy importante evaluar el funcionamiento de dicho sistema.

## 3.1. Descripción de las métricas

En esta sección hablaremos de los parámetros con los que hemos medido de forma cuantificable nuestra red neuronal para rastrear y evaluar su comportamiento a la hora de predecir las clases con las que previamente ha sido entrenada.

La base para cuantificar el comportamiento de nuestra red neuronal es la matriz de confusión, siendo esta, una herramienta excelente que nos permite visualizar el desempeño de un algoritmo de Inteligencia Artificial de aprendizaje supervisado como es nuestro caso. El número de predicciones de cada clase es representado en las columnas y las instancias de la clase real en las filas. Uno de los beneficios de las matrices de confusión es que facilitan ver si el sistema está confundiendo dos clases.

En un sistema multiclase para cuantificar el comportamiento de nuestro clasificador en cada clase, debemos dividir nuestra matriz de confusión en regiones. Para entender estas regiones nos vamos a apoyar en la Figura 3.1, donde podemos apreciar con claridad como se divide la matriz en cuatro regiones, siendo estas:

- **Verdaderos positivos (*True Positive*, TP)**, en esta región encontraremos cuantas veces un clasificador ha predicho que era de una determinada clase y ha tenido éxito en su predicción.

- **Falsos negativos (*False Negative*, FN)**, en esta región encontraremos cuantas veces un clasificador ha predicho que era de una determinada clase pero no ha tenido éxito en su predicción.

- **Falsos positivos (*False Positive*, FP)**, en esta región encontraremos cuantas veces un clasificador ha predicho que no era de una determinada clase pero no ha tenido éxito en su predicción, y por lo tanto sí era de esa clase. Los falsos positivos son "falsas alarmas" de un clasificador.

- **Verdaderos negativos (*True Negative*, TN)**, en esta región encontraremos cuantas veces un clasificador ha predicho que no era de una determinada clase y ha tenido éxito en su predicción. Los verdaderos negativos son "rechazos correctos" de un clasificador.

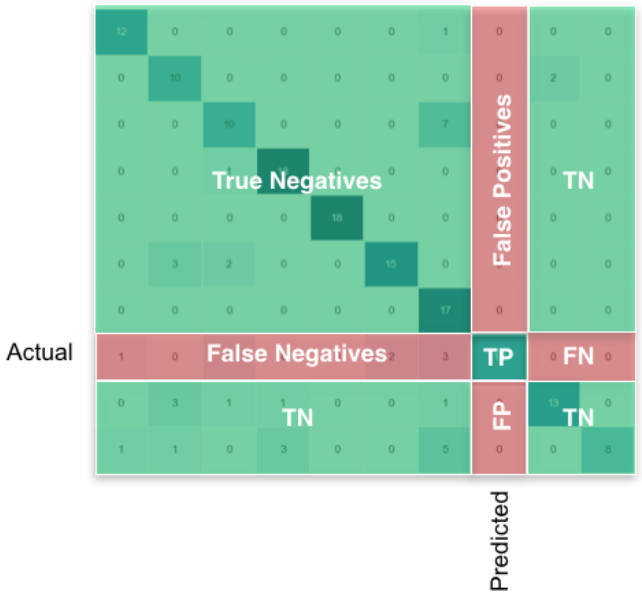



Figura 3.1: Regiones de la matriz de confusión [125].

Antes de proseguir debemos introducir el termino *Gold Standard* (GS) o test de referencia, que observamos en la Figura 3.2. Este termino en medicina es utilizado para definir aquellas pruebas de diagnóstico que tienen la máxima fiabilidad a la hora de diagnosticar una determinada enfermedad. Esto no conlleva que la prueba posea la máxima fiabilidad en términos absolutos, por tanto, podría decirse que el estatus de *Gold Standard* se aplica a aquellos tests de fiabilidad máxima dentro de una serie de condiciones específicas.

| | | CONDITION determined by "Gold Standard" | | | |
|---|---|---|---|---|---|
| | TOTAL POPULATION | CONDITION POS | CONDITION NEG | PREVALENCE = CONDITION POS / TOTAL POPULATION | |
| TEST OUT-COME | TEST POS | True Pos<br>TP | *Type I Error*<br>False Pos<br>FP | *Precision*<br>Pos Predictive Value<br>PPV = TP / TEST P | False Discovery Rate<br>FDR = FP / TEST P |
| | TEST NEG | *Type II Error*<br>False Neg<br>FN | True Neg<br>TN | False Omission Rate<br>FOR = FN / TEST N | Neg Predictive Value<br>NPV = TN / TEST N |
| | ACCURACY<br>ACC<br>ACC = (TP + TN) / TOT POP | *Sensitivity (SN), Recall*<br>Total Pos Rate<br>TPR<br>TPR = TP / CONDITION POS | *Fall-Out*<br>False Pos Rate<br>FPR<br>FPR = FP / CONDITION NEG | Pos Likelihood Ratio<br>LR +<br>LR + = TPR / FPR | Diagnostic Odds Ratio<br>DOR<br>DOR = LR + / LR - |
| | | *Miss Rate*<br>False Neg Rate<br>FNR<br>FNR = FN / CONDITION POS | *Specificity (SPC)*<br>True Neg Rate<br>TNR<br>TNR = TN / CONDITION NEG | Neg Likelihood Ratio<br>LR -<br>LR - = TNR / FNR | |

Figura 3.2: Tabla de contingencia y métricas habituales [126].

A continuación entendiendo lo que representa una matriz de confusión y conociendo las regiones de interés de esta, vamos a ver como cuantificar el comportamiento de un clasificador con cada clase en un sistema multiclase. Esto lo conseguimos a través de realizar una tabla de contingencia como la de la Figura 3.2 para cada clase. Esta tabla de contingencia nos permite calcular las métricas que deseemos para cuantificar el comportamiento de un clasificador con cada clase, también podemos observar en la Figura 3.2 como se calculan y cuales son las métricas habituales.

De todas las métricas que podemos calcular con la tabla de contingencia de cada clase, nosotros solo vamos a utilizar las cinco más usadas en la literatura médica sobre estudios diagnósticos, siendo estas:

- **Exactitud (*Accuracy*, ACC)**, es el grado de cercanía de las mediciones de una cantidad a la verdadera cantidad de ese valor o también se puede entender como el porcentaje de acierto global del clasificador o la fracción de predicciones que el modelo realizó correctamente.
- **Sensibilidad (*Sensitivity*, SEN)**, también denominada razón o ratio de verdaderos positivos (*True Positive Rate*, TPR), se define como la razón entre los individuos que tienen un resultado del test positivo y aquellos que tienen la condición o enfermedad de interés (los verdaderos positivos sobre el total de *Gold Standard* positivos), es decir, es la fracción de observaciones de clase positiva correctamente clasificadas. En un paciente determinado, si aplicamos un examen altamente sensible (identifica muy bien a los enfermos) y obtenemos un resultado negativo, podemos descartar razonablemente la enfermedad.
- **Especificidad (*Specificity*, SPE)**, también denominada razón o ratio de verdaderos negativos (*True Negative Rate*, TNR), se define como la razón entre los individuos que tienen un resultado del test negativo y aquellos sin la enfermedad de interés (verdaderos negativos sobre los *Gold Standard* negativos), es decir, es la fracción de observaciones de clase negativa correctamente clasificadas. Si un paciente tiene un resultado positivo en un test altamente específico podemos confirmar la enfermedad. En sentido estricto nos dice qué tan bueno es el test en excluir a los sanos.
- **Valor predictivo positivo (*Positive Predictive Value*, PPV)**, también denominado precisión, es la proporción de resultados de clase positivos que son verdaderos positivos y se corresponde a la probabilidad de que un individuo con un resultado positivo, tenga la enfermedad.
- **Valor predictivo negativo (*Negative Predictive Value*, NPV)**, es la proporción de resultados negativos de clase que son verdaderos negativos y se corresponde a la probabilidad de que un individuo con un resultado negativo, no tenga la enfermedad.

Si bien los valores predictivos, a diferencia de la sensibilidad y especificidad, nos entregan información clínicamente relevante (la probabilidad de que la condición esté o no presente dado el resultado del test), ésta sólo es utilizable si nos enfrentamos a pacientes similares a aquellos en que se realizó el estudio. Los valores predictivos varían enormemente dependiendo de la prevalencia o riesgo basal de la condición, por lo que si nuestro paciente tiene un riesgo mayor o menor, no podemos aplicarlos. Lo anterior no ocurre con la sensibilidad y especificidad, ya que su cálculo no depende de la prevalencia de la condición (al menos desde el punto de vista matemático). Esto ha hecho que constituyan una de las formas más frecuentes de expresar el rendimiento de un test.

En resumen podemos decir que la sensibilidad y especificidad no varían con la prevalencia de la condición, pero no nos hablan de la probabilidad que tiene un paciente de presentar la enfermedad de interés y los valores predictivos nos hablan de la probabilidad que tiene un paciente de presentar la enfermedad de interés, pero varían enormemente dependiendo de la prevalencia de la condición [127].

## 3.2. Curva ROC

En esta sección hablaremos de otro parámetro con el que hemos medido de forma cuantificable nuestro clasificador para rastrear y evaluar su comportamiento a la hora de predecir las clases con las que previamente ha sido entrenada, en concreto de la curva característica operativa del receptor (*Receiver Operating Characteristic*, ROC).

La curva ROC es la representación gráfica de la TPR frente a la razón o ratio de falsos positivos (*False Positive Rate*, FPR), una curva ROC genérica la podemos ver en la Figura 3.3. Un gráfico de curva ROC también ilustra

la sensibilidad y especificidad de cada uno de los posibles puntos de corte de un test diagnóstico cuya escala de medición es continua. La curva ROC se construye en base a la unión de distintos puntos de corte, correspondiendo el eje Y a la sensibilidad y el eje X a (1-especificidad) de cada uno de ellos. Ambos ejes incluyen valores entre 0 y 1 (0 % a 100 %). A modo de referencia, en todo gráfico de curva ROC se traza una línea desde el punto [0, 0] al punto [1, 1], llamada diagonal de referencia o línea de no-discriminación (concepto a abordar más adelante) [128]. El análisis de la curva ROC, proporciona herramientas para seleccionar los modelos posiblemente óptimos y descartar modelos subóptimos independientemente de (y antes de específicar) el coste de la distribución de las dos clases sobre las que se decide. La curva ROC es también independiente de la distribución de las clases en la población (en diagnóstico, la prevalencia de una enfermedad en la población).


Figura 3.3: Curva ROC.

La realidad nos indica que una amplia gama de tests diagnósticos reportan sus resultados cuantitativamente, utilizando escalas continuas (*e.g.*, recuento de leucocitos, proteína C reactiva). En ellas, para formular el diagnóstico de una determinada enfermedad se establece un punto de corte, sobre el cual se apoya la presencia del diagnóstico y bajo el cual se rechaza, o viceversa [128]. El análisis en base a curvas ROC constituye un método estadístico para determinar la exactitud diagnóstica de tests que utilizan escalas continuas, siendo utilizadas con tres propósitos específicos: determinar el punto de corte en el que se alcanza la sensibilidad y especificidad más alta, evaluar la capacidad discriminativa del test diagnóstico, es decir, su capacidad de diferenciar sujetos sanos *versus* enfermos, y comparar la capacidad discriminativa de dos o más tests diagnósticos que expresan sus resultados como escalas continuas.

El punto de corte de una escala continua que determina la sensibilidad y especificidad más alta es aquel que presenta el mayor índice de Youden, calculado según la fórmula $(sensibilidad + especificidad - 1)$. Gráficamente, éste corresponde al punto de la curva ROC más cercano al ángulo superior-izquierdo del gráfico (punto 0,1), es decir, más cercano al punto del gráfico cuya sensibilidad = 100 % y especificidad = 100 %. En este momento es preciso hacer una aclaración: el índice de Youden identifica el punto de corte que determina la sensibilidad y especificidad más alta conjuntamente (*i.e.*, para un mismo punto), sin embargo, dicho punto de corte no necesariamente determina la sensibilidad ni la especificidad más alta que podría alcanzar el test (generalmente la sensibilidad más alta es determinada por un punto de corte, mientras que la especificidad más alta es determinada por otro).

Existen situaciones en las que se requiere disponer de un test diagnóstico altamente sensible (*e.g.*, tamizaje

de enfermedades) o bien altamente específico (*e.g.*, confirmación de enfermedades). En tales circunstancias, no es aconsejable utilizar el punto de corte identificado por el índice de Youden; por el contrario, resulta más útil conocer los valores de sensibilidad y especificidad determinados por diferentes puntos de corte, y optar por aquel que determine la mayor sensibilidad, o la mayor especificidad, según sea el objetivo [128].

La capacidad discriminativa de un test diagnóstico se refiere a su habilidad para distinguir pacientes sanos *versus* enfermos. Para ello, el parámetro a estimar es el área bajo la curva ROC (*Area Under the Curve*, AUC), medida única e independiente de la prevalencia de la enfermedad en estudio, este área en la Figura 3.3 es de color rojo. El AUC refleja qué tan bueno es el test para discriminar pacientes con y sin la enfermedad a lo largo de todo el rango de puntos de corte posibles [128]. Esta área posee un valor comprendido entre 0,5 y 1, donde 1 representa un valor diagnóstico perfecto y 0,5 es una prueba sin capacidad discriminatoria diagnóstica. Es decir, si el AUC para una prueba diagnóstica es 0,8 significa que existe un 80 % de probabilidad de que el diagnóstico realizado a un enfermo sea más correcto que el de una persona sana escogida al azar. Por esto, siempre se elige la prueba diagnóstica que presente un mayor área bajo la curva.

Para la elección entre dos pruebas diagnósticas distintas, se recurre a las curvas ROC, ya que es una medida global e independiente del punto de corte. Por esto, en el ámbito sanitario, las curvas ROC también se denominan curvas de rendimiento diagnóstico. La elección se realiza mediante la comparación del AUC de ambas pruebas. A modo de guía para interpretar las curvas ROC se han establecido los siguientes intervalos para los valores de AUC [129]:

- $AUC \in [0, 5]$: Es como lanzar una moneda.
- $AUC \in [0, 5, 0, 6)$: Test malo
- $AUC \in [0, 6, 0, 75)$: Test regular
- $AUC \in [0, 75, 0, 9)$: Test bueno.
- $AUC \in [0, 9, 0, 97)$: Test muy bueno.
- $AUC \in [0, 97, 1)$: Test excelente.

## 3.3. Validación cruzada

La validación cruzada es el diseño experimental más utilizado entre los investigadores en aprendizaje automático. En este método, los datos disponibles se dividen aleatoriamente en un conjunto de entrenamiento y un conjunto de test. El conjunto de entrenamiento se subdivide, a su vez, en dos conjuntos disjuntos.

- El conjunto de estimación, usado para seleccionar el algoritmo.
- El conjunto de validación, usado para probar o validar el algoritmo.

La motivación de esta división está en validar el algoritmo sobre un conjunto de datos diferente del empleado para estimar sus parámetros.

Existen numerosas variantes de la validación cruzada. La que se se ha mencionado es conocida como el método *hold out*, y es menos utilizada en la actualidad que la *multifold cross validation* o *k-fold cross validation* [130]. Esta última, está basado en el método anterior, pero con mayor utilidad cuando el conjunto de datos es pequeño. En este caso, el total de los datos se dividen en k subconjuntos, de manera que aplicamos el método *hold-out* k veces, utilizando cada vez un subconjunto distinto para validar el modelo entrenado con los otros k-1 subconjuntos como vemos en la Figura 3.4 basada en la Tabla 6.8. El error medio obtenido de los k análisis realizados nos proporciona el error cometido por el método, permitiendo así evaluar su validez.

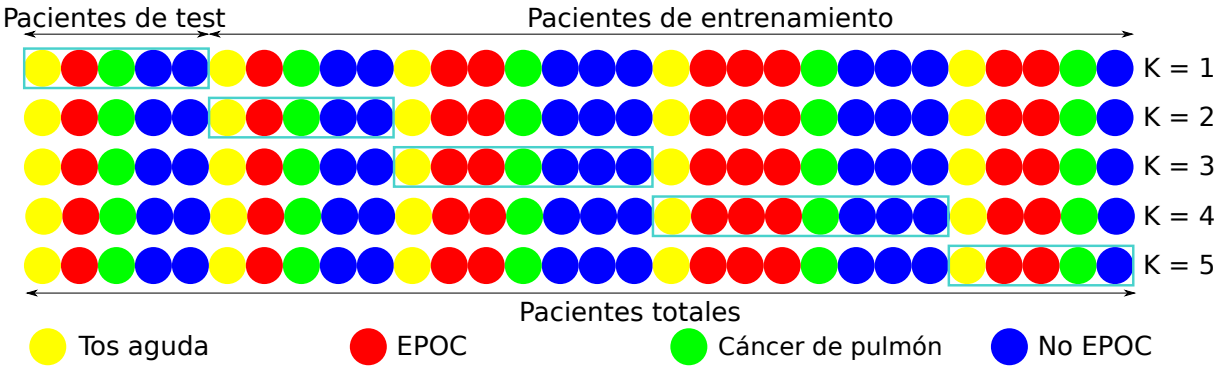



Figura 3.4: Ejemplo de *5-fold cross validation*.

Si comparamos los dos métodos, el método *k-fold* tiene la ventaja de que todos los datos son utilizados para entrenar y validar, por lo que se obtienen resultados más representativos *a priori*. Mientras que, para el método *hold-out*, se realiza el proceso n veces de manera aleatoria, lo que no garantiza que los casos de entrenamiento y validación no se repitan [131].

La mayor ventaja de este diseño experimental es que las estimaciones del error sobre los conjuntos de test son independientes (los conjuntos de *test* no se solapan). Sin embargo, sí existe un cierto solapamiento en lo que se refiere al conjunto de entrenamiento, ya que cada pareja de conjuntos de entrenamiento comparte una alta fracción de los ejemplos. Por este motivo, este diseño experimental no estudia de forma adecuada la variabilidad inducida por la utilización de distintos ejemplos para el entrenamiento. Adicionalmente, existe un claro desequilibrio entre el número de ejemplos utilizado para *test* y para *train* cuando k >3. Esta circunstancia tiene dos efectos: por una parte, los algoritmos cuyo error decrece cuanto mayor sea el número de ejemplos utilizados para el *train* verán estimada de forma optimista su error producido. Por otra parte, esta estimación del error tendrá una mayor variabilidad. Algunos autores proponen utilizar una estrategia determinista para realizar las particiones del conjunto de ejemplos, con objeto de que las particiones contengan ejemplos lo más diversos que sea posible, dentro de cada una de ellas y, paralelamente, que las particiones sean similares entre sí. Con esto se consigue eliminar la variabilidad en la estimación del error que se produce en determinados algoritmos (los llamados “inestables”). Adicionalmente, la alternativa determinista permite repetir una experimentación sin necesidad de conocer las particiones del conjunto de ejemplos.

Existen más variaciones de la validación cruzada. La técnica *complete cross validation* utiliza todas las posibles particiones del conjunto de ejemplos con un tamaño dado, lo que mejora la estimación del error de generalización. Como el número de particiones sólo es abordable en problemas de dimensión reducida, es posible seleccionar un número menor de particiones, con la ayuda de diferentes criterios. El *leave one out* es el caso extremo en que cada conjunto de test contiene un único elemento [130].

## 3.4. Tipos de promediado de resultados

Cuando tenemos múltiples etiquetas de clase, promediar las métricas o medidas de evaluación puede dar una visión de los resultados generales. Hay dos nombres para referirse a los resultados promediados: micro-promediados y macro-promediados.

Los resultados macro-promediados se pueden calcular mediante:

$$B_{macro} = \frac{1}{q}\sum_{\lambda=1}^{q} B(TP_\lambda, FP_\lambda, TN_\lambda, FN_\lambda) \tag{3.1}$$

Sabiendo que $L = \{\lambda_j : j = 1, ..., q\}$ es el conjunto de todas las etiquetas y considerando una medida de evaluación $B(TP, TN, FP, FN)$ que se calcula en función del número de verdaderos positivos (TP), verdaderos negativos (TN), falsos positivos (FP) y falsos negativos (FN). Sean $TP_\lambda$, $FP_\lambda$, $TN_\lambda$ y $FN_\lambda$ el número de

verdaderos positivos, falsos positivos, verdaderos negativos y falsos negativos después de una evaluación para una etiqueta $\lambda$.

Mientras que los resultados micro-promediados se puede calcular de la siguiente manera:

$$B_{micro} = B\left(\sum_{\lambda=1}^{q} TP_{\lambda}, \sum_{\lambda=1}^{q} FP_{\lambda}, \sum_{\lambda=1}^{q} TN_{\lambda}, \sum_{\lambda=1}^{q} FN_{\lambda}\right) \tag{3.2}$$

Observando la diferencia entre macro y micro-promediado, el macro-promediado otorga el mismo peso a cada clase, mientras que el micro-promediado otorga el mismo peso a cada decisión de clasificación por documento. Para tener una idea de la eficacia en clases pequeñas, debe calcular los resultados de macro-promediado. No hay un acuerdo completo entre los autores sobre cuál es mejor. Algunos creen que el rendimiento de micro-promedio es un tanto engañoso porque los temas más frecuentes son más pesados en el promedio y, por lo tanto, favorecen el macro-promediado. Otros (en realidad, la mayoría de los investigadores) creen que los temas deberían contar de manera proporcional a su frecuencia y, por lo tanto, inclinarse hacia el micro-promediado [132].

En nuestro caso considerando una configuración de validación cruzada de *K-fold*. Para cada uno de los k pliegues, hay una salida del sistema y por lo visto en esta sección tenemos dos estilos principales para resumir las salidas del sistema en una evaluación final. El estilo de micro-promediado y el estilo de macro-promediado, en nuestro caso nosotros elegimos el macro-promedio.

# *Capítulo* 4
# DEEP LEARNING

En este capitulo vamos a tratar de adentrarnos en el concepto de *Deep Learning*, debido a que en este trabajo vamos a emplear estos algoritmos y será de vital importancia su correcta comprensión. En el capitulo anterior ya se explicaron los conceptos de Inteligencia Artificial y *Machine Learning*. Esto es importante recordarlo, ya que debemos de tener claro que la Inteligencia Artificial es el concepto más general, una rama de la Inteligencia Artificial es el *Machine Learning* y dentro del *Machine Learning* podemos encontrar un conjunto de algoritmos denominado *Deep Learning*.

## 4.1. Introducción al *Deep Learning*

De la Inteligencia Artificial al *Machine Learning* y de ahí al *Deep Learning* o aprendizaje profundo. Un nuevo paso en los procesos y la evolución de los sistemas de Inteligencia Artificial que se acerca, cada vez más, al reto de conseguir sistemas informáticos que, de forma autónoma y sin apenas intervención humana, sean capaces de imitar comportamientos y razonamientos propios del ser humano. Con este concepto nos estamos refiriendo al conjunto de algoritmos que, a modo de las redes neuronales de cerebro humano, son capaces de generar respuestas y actuar en función de las conclusiones a las que se llega por su combinación, yuxtaposición o contradicción a partir de un sistema de capas que se ordenan en función de una jerarquía.

Estamos hablando de un nuevo nivel en el desarrollo de los sistemas de Inteligencia Artificial y *Machine Learning* y, por el momento, de lo más parecido al pensamiento humano que podemos encontrar. De hecho, estas redes neuronales son capaces de generar procesos que incluyen el pensamiento abstracto (lenguaje propio y razonado) o, incluso, según algunos expertos, en un futuro no muy lejano, de la capacidad de crear [19].

Una red de neuronas, llamado comúnmente red neuronal artificial, es una herramienta matemática que modela -de forma muy simplificada- el funcionamiento de las neuronas en el cerebro. Otra forma de verlas, es como un procesador de datos que recibe información entrante, codificada como números, hace una serie de operaciones y produce como resultado información saliente, codificada también como otros números.

Para ser más precisos, hablamos de una familia de algoritmos cuyo propósito es simular el comportamiento que lleva a cabo nuestro cerebro para reconocer imágenes, palabras o sonidos. Son algoritmos que funcionan en base a "un proceso por capas". El aprendizaje profundo simula el funcionamiento básico del cerebro, que se realiza a través de las neuronas. En *Deep Learning*, esas neuronas serían las capas.

La idea del aprendizaje profundo no es una idea de trabajo novedosa. Ésta surgió de la mano del investigador japonés Kunihiko Fukushima, alrededor de los años ochenta, que propuso un modelo neuronal de entre cinco y seis capas al que denominó *neocognitrón*. Sin embargo, las dificultades para desarrollar alternativas a la propuesta de Fukushima han sido muy complejas y el coste para su investigación era sumamente elevado. Estos motivos hicieron que las técnicas de aprendizaje profundo no se hayan vuelto a retomar con fuerza hasta hace escasamente una década, reactivándose el interés y la inversión por parte de las empresas [22].

Las principal diferencia entre *Machine Learning* y *Deep Learning* es que la extracción de características a partir de los datos de entrada se hace típicamente de forma manual en *Machine Learning*, mientras que en *Deep Learning*

se obtienen directamente del entrenamiento de la red, para lo cual necesitamos grandes volúmenes de datos a modo de ejemplos, por lo que requerirá sistemas más complejos y equipos con una capacidad de procesamiento mayor. Esto quiere decir que realiza de forma automática el proceso de extracción de características, y a cambio necesitará utilizar una mayor cantidad de datos para el entrenamiento, mejorando el rendimiento del sistema cuanto mayor sea la cantidad de datos de la que disponemos. Esto lleva también a que *Deep Learning* sea capaz de soportar y trabajar con *big data*, lo que supone una gran ventaja con respecto al *Machine Learning*.

Además en *Machine Learning* se emplean redes neuronales sencillas, como el perceptrón simple que solo tiene una capa oculta, mientras que en *Deep Learning* necesitaremos redes neuronales multicapa, con más de una capa oculta, llegando a existir sistemas que soporten cientos de capas ocultas, de las cuales hablaremos más tarde. Esto significa que tendremos un grado mayor de complejidad, debido a que no podremos ver las salidas en todos los niveles [133].

### 4.1.1. Evolución histórica

Los primeros trabajos para la construcción de modelos matemáticos que imitasen el comportamiento de las neuronas del cerebro se deben al neurofisiólogo Warren McCulloch y al matemático Walter Pitts, que presentaron en 1943 uno de los primeros modelos de una neurona artificial [22] en su artículo *A logical calculus of the ideas immanent in nervous activity* [134] y donde demostraron que un programa de la máquina de Turing podría ser implementado en una red finita de neuronas convencionales, además de que la neurona era la unidad lógica básica del cerebro. Pitts y McCulloch también escribieron el ensayo *How we know universals the perception of auditory and visual forms* [135] de 1947 en el que ofrecieron aproximaciones para diseñar "redes nerviosas" para el reconocimiento de entradas visuales a pesar de los cambios en orientación y tamaño. Aunque Alan Turing, del que ya hemos hablado anteriormente como padre de la Inteligencia Artificial, fue el primero en estudiar el cerebro desde el punto de vista computacional en 1936.

Poco después, en 1949, Donald Hebb propuso el siguiente principio "Cuando el axón de una célula A está lo suficientemente cerca de una célula B como para excitarla y participa repetida o persistentemente en su disparo, ocurre algún proceso de crecimiento o cambio metabólico, en una o ambas células, de tal modo que la eficacia de A en disparar a B se ve aumentada" [136], esta ley, conocida como "regla de Hebb", explica a grandes rasgos el aprendizaje neuronal y se convirtió en la precursora de las técnicas de entrenamiento de redes neuronales artificiales de hoy en día.

A partir de estas aportaciones iniciales, durante la década de los años 50 y 60 surgieron nuevos desarrollos, destacando los trabajos de Marvin Minsky y Frank Rosenblatt, quienes desarrollaron el conocido como perceptrón, o perceptrón simple, un modelo sencillo capaz de generalizar el conocimiento y que, tras aprender de antemano una serie de patrones, podía reconocer otros similares aunque anteriormente no se le hubieran presentado [22].

Después del trabajo de Minsky y Rosenblatt sobre el perceptrón simple, Bernard Widrow y Ted Hoff desarrollaron en 1960, la denominada "Ley de Widrow-Hoff" [137], una importante variación del algoritmo de aprendizaje del perceptrón, que dio lugar al modelo ADALINE (*ADAptive LINear Elements*), que constituyó la primera red neuronal artificial aplicada a un problema real: la eliminación de los ecos de las líneas telefónicas por medio de filtros adaptativos.

A pesar de los brillantes inicios de la investigación usando redes neuronales, el interés de la comunidad científica por éstas disminuyó enormemente al publicarse el libro: *Perceptrons: An introduction to Computational Geometry* [138] de los investigadores del MIT ya mencionados Marvin Minsky y Seymour Papert. Estos autores demostraron importantes limitaciones teóricas en el aprendizaje de los modelos neuronales artificiales desarrollados hasta entonces, en particular de la red perceptrón, lo que las convertía en juguetes matemáticos sin aplicabilidad práctica real [22].

Este trabajo, propició que abandonaran el ámbito neuronal numerosos autores para centrarse en el análisis de los sistemas basados en el conocimiento, mucho más prometedor en aquel momento. Aunque, otros autores continuaron investigando en el campo de las redes neuronales artificiales, destacando el Asociador Lineal desarrollado por James Anderson en 1977 y su extensión conocida como red *Brain-State-in-a-Box*, que permitieron modelizar funciones arbitrariamente complejas [139].

En 1982 coincidieron numerosos eventos que hicieron resurgir el interés en las redes neuronales artificiales. John Hopfield presentó su trabajo sobre redes neuronales describiendo con claridad y precisión una variante del

Asociador Lineal inspirada en la minimización de la energía presente en los sistemas físicos, conocida como "red de Hopfield" [140]. También, en ese mismo año Fujitsu comenzó el desarrollo de "computadoras pensantes" para diversas aplicaciones en robótica.

Werbos [141], Parker [142] y LeCun [143] formularon una nueva regla, la denominada "Regla Delta Generalizada", que dió lugar a un avance muy significativo en el aprendizaje supervisado. Asimismo, el desarrollo por Rumelhart del algoritmo de aprendizaje supervisado para redes neuronales artificiales conocido como *backpropagation* [144], [145] ofreció en 1986 una solución muy potente para la construcción de redes neuronales más complejas al evitar los problemas observados en el aprendizaje del perceptrón simple. Este algoritmo constituye desde entonces una de las reglas de aprendizaje de mayor utilización para el entrenamiento de la red conocida como perceptrón multicapa [22].

En 1988, los esfuerzos de la IEEE y de la INNS se unieron para formar la *International Joint Conference on Neuronal Networks* (IJCNN) y, tres años más tarde, surgió la *International Conference on Artificial Neural Networks* (ICANN), organizada por la Sociedad Europea de Redes Neuronales (*European Neural Network Society*, ENNS). Igualmente, desde 1987 se viene celebrando la reunión anual *Neural Information Processing Systems* (NIPS), que constituye uno de los referentes de más alto nivel en este campo de investigación.

En los últimos años, como consecuencia de estos esfuerzos, las redes neuronales artificiales han experimentado un importante desarrollo, llegando a conseguir que el paradigma conexionista supere a las aplicaciones basadas en modelos simbólicos. Las investigaciones se centran en la combinación de ambos paradigmas de aprendizaje, con el fin de conseguir una mayor unión entre la capacidad de procesamiento y aproximación de las redes neuronales artificiales, que pueden llegar a soluciones sorprendentemente buenas con rapidez y poca información de partida, y el potencial de los sistemas basados en el conocimiento, como demuestran los trabajos de Tomas Hrycej [146] en Inteligencia Artificial, Paul McNelis [147], [148] en finanzas y Bart Baesens [149] en minería de datos, entre otros [22]. Un resumen de la cronología del *Deep Learning* incluyendo los últimos avances la podemos ver en la Figura 4.1

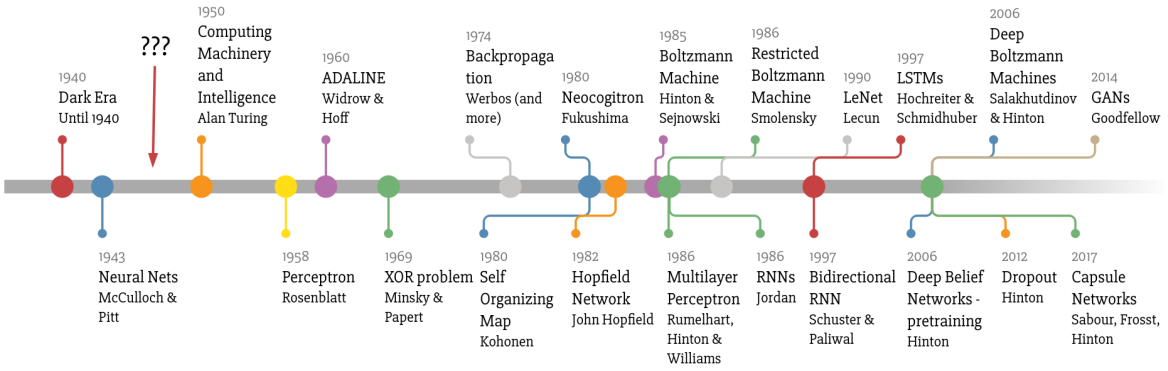
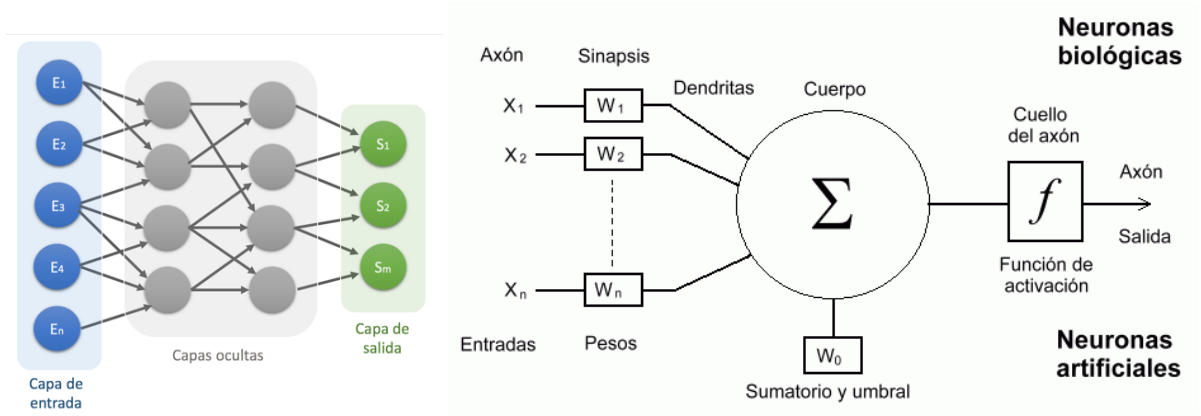



Figura 4.1: Cronología del *Deep Learning*.

### 4.1.2. Funcionamiento básico de una red neuronal

Podemos observar en la Figura 4.2a como es la arquitectura genérica de una red neuronal artificial y en la Figura 4.2b como es el esquema de cada una de las neuronas que la componen, siendo representada cada neurona como un círculo.




(a) Arquitectura de una red neuronal básica [150].

(b) Esquema de una neurona artificial [151]

Figura 4.2: Red neuronal.

Normalmente, todas las neuronas de cada capa tienen una conexión con cada neurona de la siguiente capa, estas conexiones tienen asociado un número, que se llama peso. Lo que hacemos con el peso asociado es multiplicar los valores de entrada de nuestra neurona por dicho peso. Dado que nuestra neurona tendrá conexión con varias neuronas, lo siguiente que realizara es la suma de todas las conexiones entrantes con los pesos aplicados. Posteriormente cada neurona añade a la suma de productos un nuevo término constante, llamado habitualmente *bias* o umbral.

Una vez realizadas las operaciones anteriores otra operación que realizan todas las capas, salvo la capa de entrada, se trata de la función de activación. Esta función $f$, transforma el valor producido por las operaciones anteriores mediante una fórmula que da lugar a un nuevo número. Existen varias opciones, pero las funciones más habituales son las funciones sigmoide [22], tangente hiperbólica y unidad lineal rectificada (*Rectified Linear Unit*, ReLU), las cuales las podemos observar en la Figura 4.3. Estando en boga actualmente la utilización de la función ReLU por su mayor parecido al modelo biológico, los entrenamientos de las redes son mejores y por lo tanto, los resultados que produce son excelentes.

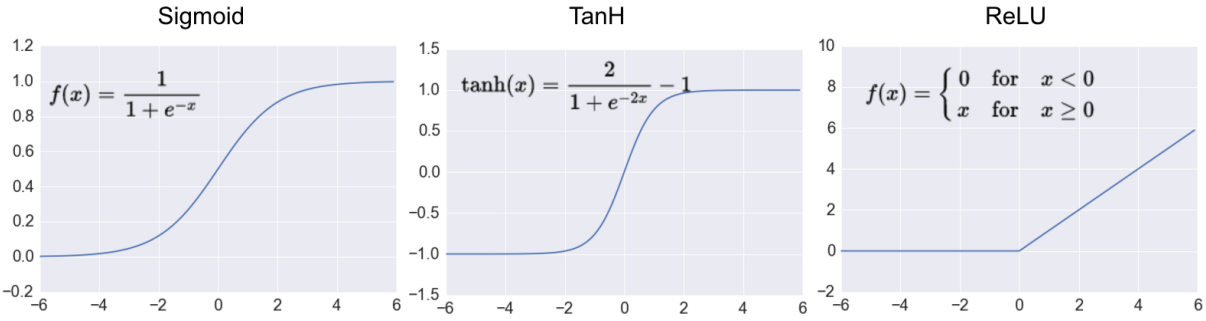



Figura 4.3: Funciones de activación [152].

Además, no nos podemos olvidar de la función *softmax*, o función exponencial normalizada, que es utilizada habitualmente en la última capa para "comprimir" el vector K-dimensional que recibe en la entrada y producir en la salida un vector N-dimensional, de valores reales en el rango [0, 1], que resulta útil para obtener probabilidades

en tantos por uno. Mediante la expresion:

$$f(x)_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \qquad \forall i \in [1, N] \tag{4.1}$$

Matemáticamente y a modo de resumen de lo expuesto anteriormente, podemos expresar la salida de una neurona como:

$$Y = f(\sum_{i=1}^{n} X_i W_i + W_0) \tag{4.2}$$

Siendo $X_i$ la entrada o estímulo $i$ que recibe la neurona, $W_i$ el peso asociado a la entrada $i$ que recibe la neurona, equivalente a las conexiones sinápticas de una neurona real, $W_0$ el *bias* aplicado por la neurona y $f$ la función de activación.

### 4.1.3. Aprendizaje de una red neuronal

Las redes neuronales requieren de entrenamiento para poder ser utilizadas. El proceso de aprendizaje busca un conjunto de pesos que permitan a la red desarrollar correctamente una determinada tarea, a partir de un conjunto de pesos sinápticos aleatorio. La solución se va afinando iterativamente durante el proceso de aprendizaje hasta alcanzar un nivel de operación suficientemente bueno [22]. Este proceso de aprendizaje no siempre es igual y se pueden dividir los tipos de aprendizaje en tres, siendo estos:

- **Aprendizaje supervisado**: Consiste en presentar a la red un conjunto de patrones de entrada y su salida esperada. El objetivo del algoritmo de aprendizaje es modificar los pesos $W_i$ de la red de forma proporcional al error que se produce entre la salida real de la red y la esperada con el objetivo de conseguir a la salida una solución lo más parecida posible a la verdadera [22]. Este tipo de aprendizaje es el que realizamos nosotros, ya que en tareas de clasificación como la nuestra, es el mejor tipo de entrenamiento.

- **Aprendizaje no supervisado**: Se presenta a la red un conjunto de patrones de entrada, ajustándose el modelo a partir de dichas observaciones. No hay información disponible sobre la salida esperada, lo cual es su principal diferencia con respecto al aprendizaje supervisado, ya que no tenemos conocimiento *a priori*. En este caso el proceso de entrenamiento ajustará sus pesos mediante la correlación existente entre los datos de entrada [22]. El objetivo suele ser aprender la distribución de probabilidad completa que generó un conjunto de datos, ya sea explícitamente, como en la estimación de la densidad, o implícitamente, para tareas como síntesis o eliminación de ruido, otro objetivo habitual para este tipo de aprendizaje es la agrupación en *clusters*, que consiste en dividir el conjunto de datos en grupos de ejemplos similares [153].

- **Aprendizaje por refuerzo**: Este tipo de aprendizaje se ubica en el medio de los dos anteriores y es basado en el aprendizaje a partir del entorno, siendo el entorno el conjunto de datos de entrada y formulado generalmente como un proceso de decisión de Markov. Este conjunto de patrones de entrada se le presenta a la red y se le indica a la misma si la salida obtenida es o no correcta, generando algún tipo de realimentación o recompensa acerca de la idoneidad de la respuesta dada. En aquellos casos en los que se desconoce cuál es la salida exacta que debe proporcionar la red, este tipo de aprendizaje es muy útil, ya que no se le proporciona el valor de la salida esperada [154]. La empresa DeepMind de Google sacó a la luz el 5 de diciembre de 2017 la Inteligencia Artificial AlphaZero [155] que únicamente con aprendizaje por refuerzo logró superar en 24 horas un nivel de juego sobrehumano en ajedrez, shogi y Go al derrotar a los campeones del mundo, Stockfish, Elmo y AlphaGo Zero, sin acceso a libros de apertura o base de datos de tablas de finales.

Para entender mejor estos tipos de aprendizaje y ver que no son compartimentos estancos tenemos el ejemplo actual de AlphaStar [156] que es la Inteligencia Artificial de DeepMind capaz de jugar al *StarCraft II*. La cual en diciembre de 2018 en una serie de partidas de prueba venció al jugador profesional Grzegorz “MaNa” Komincz por 5–0, esto se dio en condiciones de partidas oficiales entre jugadores profesionales. AlphaStar fue entrenado inicialmente utilizando aprendizaje supervisado sobre partidas anónimas liberadas por Blizzard, esto permitió que

aprendiera micro y macro estrategias de jugadores reales y luego el sistema comenzó a jugar contra si mismo para mejorar sus estrategias por medio de aprendizaje por refuerzo [157].

Una vez finalizada la fase de aprendizaje, la red puede ser utilizada para realizar la tarea para la que fue entrenada [22]. Debido a que la red aprende la relación existente entre los datos, adquiriendo la capacidad de generalizar conceptos, esta puede tratar con información que no le fue presentada durante la fase de entrenamiento.

### 4.1.4. Entrenamiento de una red neuronal

El proceso de aprendizaje es un proceso iterativo, en el que la solución se va refinando hasta alcanzar un nivel de operación suficientemente bueno, buscando un conjunto de pesos que permitan a la red desarrollar una determinada tarea, a partir de un conjunto de pesos aleatorios. La manera de realizar el aprendizaje consisten en proponer una función de error que mida el rendimiento actual de la red en función de los pesos. El objetivo del método de entrenamiento es encontrar el conjunto de pesos que minimizan la función de error. El método de optimización proporciona una regla de actualización de los pesos que modifica iterativamente los pesos en función de los patrones de entrada hasta alcanzar el punto óptimo de la red neuronal [22].

#### Gradiente Descendente

El método de optimización más utilizado para el entrenamiento es el método del gradiente descendente. Este método define una función $f(\theta)$ que proporciona el error que comete la red en función del conjunto de pesos $W$, que en este caso llamaremos $\theta$. El objetivo ideal del aprendizaje será encontrar la configuración de pesos que corresponda al mínimo global de la función de error, aunque normalmente es suficiente con encontrar un mínimo local lo bastante bueno.

Ésta, es una expresión matemática de carácter iterativo, que comienza con un conjunto de pesos $\theta_0$ para el instante $t = 0$, con el que se calcula la dirección de máxima variación del error. La dirección de máximo crecimiento de la función $f(\theta)$ en $\theta_0$ viene dado por el gradiente $\nabla_\theta f(\theta)$. Luego, se actualizan los pesos siguiendo el sentido contrario al indicado por el gradiente $\nabla_\theta f(\theta)$, dirección que indica el sentido de máximo decrecimiento. De este modo, se va produciendo un descenso por la superficie de error hasta alcanzar un mínimo local [22]. La formula matemática en el gradiente descendente es:

$$\theta_t = \theta_{t-1} - \alpha \cdot \nabla_\theta f(\theta) \tag{4.3}$$

Donde $\alpha$ indica tasa de aprendizaje que tiene el algoritmo, la cual idealmente debería ser infinitesimal. El tamaño de la tasa de aprendizaje es un factor importante a la hora de diseñar un método de estas características. El proceso de entrenamiento resultará muy lento si se toma una tasa de aprendizaje muy pequeña, mientras que se producirán oscilaciones en torno al punto mínimo si el tamaño de la tasa de aprendizaje es muy grande [22], [158].

#### Adam

Adam es un algoritmo para la optimización, basada en gradientes de primer orden, de funciones objetivas estocásticas, fundamentado en estimaciones adaptativas de momentos de orden inferior. El método es sencillo de implementar, es computacionalmente eficiente, tiene pocos recursos de memoria, es invariante a la reescala diagonal de los gradientes y es adecuado para problemas que son grandes en términos de datos y/o parámetros. El método también es apropiado para objetivos no estacionarios y problemas con gradientes muy ruidosos y/o dispersos. Los hiper-parámetros tienen interpretaciones intuitivas y típicamente requieren poco ajuste. Los resultados empíricos demuestran que Adam funciona bien en la práctica y se compara favorablemente con otros métodos de optimización estocástica. Posteriormente, en la sección 6.4.1 hablaremos de AdaMax que es una variante de Adam y es el optimizador que utilizaremos.

El algoritmo iterativo matemático de Adam es bastante complejo por lo que trataremos de desarrollarlo a lo largo de los siguientes puntos:

1) Se obtienen los gradientes con respecto al objetivo estocástico en el *timestep* t con la formula:

$$g_t = \nabla_\theta f_t(\theta_{t-1}) \tag{4.4}$$

2) Se estima el primer momento sesgado actualizándolo mediante la expresión:

$$m_t = \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t \tag{4.5}$$

3) Se estima el segundo momento sesgado actualizándolo mediante la igualdad:

$$v_t = \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot g_t^2 \tag{4.6}$$

4) Se calcula la estimación del primer momento corregido por el sesgo a través de la formula:

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t} \tag{4.7}$$

5) Se calcula la estimación del segundo momento corregido por el sesgo a traves de la igualdad:

$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t} \tag{4.8}$$

6) Se actualizan los parametros mediante la expresion:

$$\theta_t = \theta_{t-1} - \frac{\alpha \cdot \hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon} \tag{4.9}$$

Para comenzar a calcular este algoritmo necesitamos un *stepsize* que hemos denominado $\alpha$, las tasas de decaimiento exponencial para las estimaciones del momento, a las que las hemos denominado como $\beta_1, \beta_2 \in [0, 1)$ y la función objetivo estocástica con parámetros $\theta$ que se denomina como $f(\theta)$. Además se requiere inicializar los parametros $\theta_0$, $m_0 = 0$, $v_0 = 0$ y $t = 0$, que son respectivamente el parámetro inicial vector, el primer vector de momento, el segundo vector de momento y el *timestep* [159], [160].

**AdaMax**

AdaMax es una variante de Adam basada en la norma del infinito. En AdaMax, la tasa de aprendizaje (*learning rate*) es $\alpha/(1 - \beta_1^t)$ con el término de corrección de sesgo para el primer momento, siendo $\beta_1^t$, $\beta_1$ a la potencia de $t$.

El algoritmo iterativo matemático de AdaMax es bastante complejo por lo que trataremos de desarrollarlo a lo largo de los siguientes puntos:

1) Se obtienen los gradientes con respecto al objetivo estocástico en el *timestep* t con la formula:

$$g_t = \nabla_\theta f_t(\theta_{t-1}) \tag{4.10}$$

2) Se estima el primer momento sesgado actualizándolo mediante la expresión:

$$m_t = \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t \tag{4.11}$$

3) Se actualiza la norma de infinito ponderada exponencialmente mediante la igualdad:

$$u_t = max(\beta_2 \cdot u_{t-1}, |g_t|) \tag{4.12}$$

4) Se actualizan los parametros mediante la expresion:

$$\theta_t = \theta_{t-1} - \frac{\alpha}{1 - \beta_1^t} \cdot \frac{m_t}{u_t} \tag{4.13}$$

Para comenzar a calcular este algoritmo necesitamos un *stepsize* que hemos denominado $\alpha$, las tasas de decaimiento exponencial a las que las hemos denominado como $\beta_1, \beta_2 \in [0, 1)$ y la función objetivo estocástica con parámetros $\theta$ que se denomina como $f(\theta)$. Además se requiere inicializar los parametros $\theta_0$, $m_0 = 0$, $u_0 = 0$ y $t = 0$, que son respectivamente el parámetro inicial vector, el primer vector de momento, la norma del infinito ponderada exponencialmente y el *timestep* [158], [159].

***Backpropagation***

El algoritmo *Backpropagation* [144] o algoritmo de propagación hacia atrás de errores o retropropagación es un método de cálculo del gradiente utilizado en algoritmos de aprendizaje supervisado utilizados para entrenar redes neuronales artificiales, siendo el método de entrenamiento más utilizado en redes con conexión hacia delante. En este método se aplica inicialmente un patrón de entrada que se propaga por las distintas capas que componen la red hasta producir la salida de la misma, siendo esta salida comparada con la salida deseada y sobre la cual se calcula el error cometido por cada neurona de salida. Los errores que han sido calculados son transmitidos hacia atrás, partiendo de la capa de salida, llegando a todas las neuronas de las capas intermedias [22]. Debido a la forma de propagarse el error, cada una de las neuronas recibe un error que es proporcional a su contribución sobre el error total de la red y basándose en el error recibido, se ajustan los errores de los pesos de cada neurona.

El principal problema de este algoritmo es que el error se va diluyendo de forma exponencial a medida que atraviesa capas, hasta llegar al principio de la red. Este problema se agranda en una red muy profunda, es decir, con muchas capas ocultas, debido a que sólo las últimas capas se entrenan, mientras que las primeras apenas llegan a sufrir cambios. Por este motivo compensa utilizar redes con pocas capas ocultas que contengan muchas neuronas, en lugar de redes con muchas capas ocultas que contengan pocas neuronas [22].

La aplicación del algoritmo *backpropagation* se produce durante, el aprendizaje de nuestra red que es realizado mediante la presentación sucesiva de un conjunto de entrenamiento, donde cada presentación completa que se realiza sobre nuestra red neuronal del *set* de entrenamiento se denomina época. De este modo, el proceso de aprendizaje se repite época tras época hasta que los pesos se estabilizan y el rendimiento de la red converge en un valor aceptable. En función de como se actualizan los pesos tenemos dos modos de entrenamiento distintos :

- **Modo Secuencial**: La actualización de los pesos en este modo de entrenamiento se produce tras la presentación de cada ejemplo de entrenamiento, por lo que es conocido también como modo por patrón. Cuando los patrones de entrenamiento se presentan a la red de manera aleatoria, el modo de entrenamiento secuencial hace estocástica la búsqueda en el espacio de pesos, y disminuye la probabilidad de que quede atrapado en un mínimo local el algoritmo *backpropagation*. Sin embargo, la naturaleza estocástica del modo de entrenamiento secuencial dificulta el establecimiento de condiciones teóricas para la convergencia del algoritmo [22].
- **Modo *Batch***: La actualización de los pesos en este modo de entrenamiento se produce tras la presentación de todo el *set* de entrenamiento, por lo que para cada época se calcula el error, habitualmente como el error cuadrático medio, producido por la red [22]. El uso del modo de entrenamiento *batch* nos ofrece una estimación precisa del vector gradiente, garantizando de esta manera la convergencia hacia un mínimo local.

## 4.2. Modelos de redes redes neuronales

En *Deep Learning* podemos encontrar numerosos modelos de redes de neuronas a la horas de resolver un determinado problema, y ahora vamos a hablar sobre los principales:

### 4.2.1. Perceptrón Multicapa (*Multilayer Perceptron*, MLP)

El perceptrón simple o perceptrón, fue desarrollado en 1957 por Frank Rosenblatt, con el objetivo de ilustrar algunas propiedades fundamentales de los sistemas inteligentes. Pero como dijimos anterormente, el interés de la comunidad científica por éstas bajó enormemente en 1969 al publicarse el libro: *Perceptrons: An introduction to Computational Geometry* [138] de los ya mencionados Marvin Minsky y Seymour Papert, significando para muchos el fin de las redes neuronales. Debido a que este libro analizaba las capacidades y limitaciones de un perceptrón al detalle, haciendo hincapié en las restricciones que existen para los problemas que una red de tipo perceptrón puede resolver. Dicho libro indicó que el mayor inconveniente del perceptrón es la nula capacidad para solucionar problemas que no sean linealmente separables, hasta tal extremos que en este libro se demostró que el perceptrón no era capaz de aprender una función XOR [22].

Ese mismo año Minsky y Papert mostraron que varios perceptrones simples combinados podrían ser una solución interesante para tratar ciertos problemas no lineales, de lo cual surgió la idea del perceptrón multicapa. Pero

estos autores no solucionaron el problema de como adaptar los pesos de la capa de entrada a la capa oculta, pues la regla de aprendizaje del perceptrón simple no puede aplicarse en nuevo este escenario.

Posteriormente varios autores demostraron que el perceptrón multicapa es un aproximador universal, ya que cualquier función continua en un espacio $\Re^N$ puede aproximarse con un perceptrón multicapa con al menos una capa oculta de neuronas. Por este motivo fue adoptado el perceptrón multicapa como un modelo matemático útil aproximando o interpolando relaciones no lineales entre datos de entrada y salida.

Debido al fácil uso y aplicabilidad del perceptrón multicapa y a su capacidad como aproximador universal, está es una de las arquitecturas más importantes y utilizadas, además de constituir la estructura típica de los modelos de aprendizaje y tomarse como base para el desarrollo del resto de redes. Aunque no es una de las redes más potentes, ni con mejores resultados en sus diferentes áreas de investigación, debido a que tiene varias limitaciones. El largo proceso de aprendizaje para problemas complejos dependientes de un gran número de variables o la dificultad para realizar un análisis teórico de la red debido a la presencia de componentes no lineales son algunas de sus limitaciones. Pero además, su aprendizaje puede adquirir una gran complejidad ajustando los parámetros de la red para encontrar la mejor aproximación de la función que relacione las variables de entrada y salida del problema. Esto se debe a que la red busca en un amplio espacio de funciones y puede reducir su efectividad en determinadas aplicaciones [22].

Como hemos visto en la Figura 4.2a, en la que podemos ver un perceptrón multicapa, esta arquitectura se caracteriza por tener diferentes niveles de capas formadas por agrupaciones de neuronas, siendo estas capas:

- **Capa de entrada**: Capa inicial de la red, en la que las neuronas pertenecientes a esta capa solamente se encargan de recibir señales o patrones del exterior y propagar lo recibido a todas las neuronas de la siguiente capa.
- **Capas ocultas**: Capas intermedias de la red, en las que se realiza un procesamiento no lineal de los patrones recibidos.
- **Capa de salida**: Capa final de la red, en la cual se proporciona la respuesta de la red al exterior para cada uno de los patrones de entrada.

Debido a que las neuronas de una capa se conectan con las neuronas de la siguiente capa, las conexiones del perceptrón multicapa siempre están dirigidas hacia adelante, por lo que también reciben el nombre de redes alimentadas hacia adelante o redes neuronales prealimentadas (*Feedforward Neural Network*, FNN) [22]. Además cuando existe conexión total entre las neuronas de una capa y la siguiente, se dice que la red está totalmente conectada o tiene conectividad total [161].

### 4.2.2. *Autoencoders*

Los auto-codificadores, o *autoencoders* son una de las herramientas más utilizadas en *Deep Learning*. Su estructura más básica tiene tres capas, siendo la primera capa la de entrada, una capa oculta denominada "cuello de botella" y una capa de salida. Este tipo de redes aprende a producir en la salida exactamente el mismo tipo información que recibe a la entrada, por lo que las capas de entrada y salida siempre deben tener el mismo número de neuronas [22].

Normalmente los *autoencoders* tienen menos neuronas en las capas ocultas que en las capas de entrada y salida, por este motivo se denominan "cuellos de botella". Esta característica es muy importante, ya que la red se verá obligada a encontrar una representación intermedia de la información en sus capas ocultas usando menos números, es decir, comprimirá los datos, pero además posteriormente deberá descomprimirla para recuperar la versión "original" a la salida. Esto hace que sean capaces de descubrir por sí mismos una forma alternativa de codificar la información en sus capas ocultas sin necesidad de ejemplos proporcionados por un supervisor.

Este tipo de redes neuronales por lo explicado anteriormente se pueden dividir en dos, siendo la primera red un compresor o codificador y la segunda red un descompresor o decodificador, tal y como vemos en la Figura 4.4. Un *autoencoder* en este proceso puede encontrar características fundamentales en la información de entrada, como las rectas y curvas que son las características más primitivas y simples que se pueden extraer de una imagen [22], o puede encontrar características más complejas como rostros utilizando mas potencia computacional.

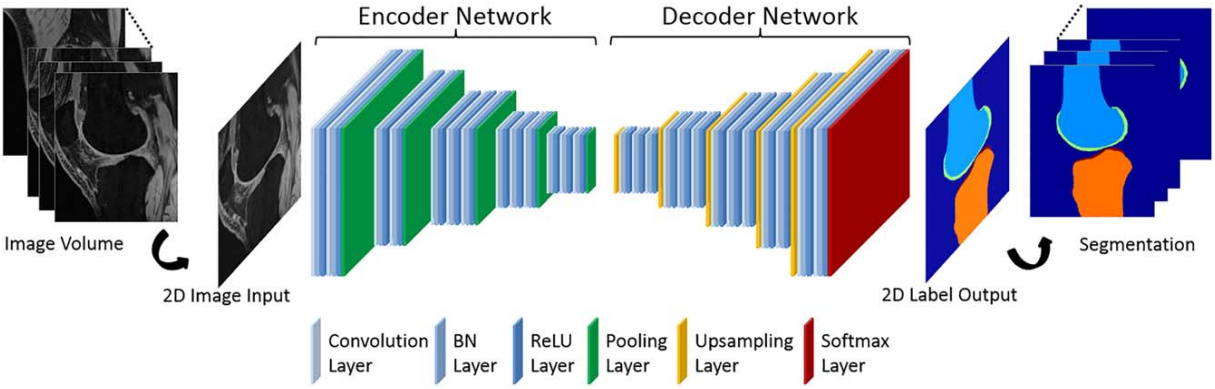



Figura 4.4: Arquitectura de un *autoencoder* [162].

### 4.2.3. Redes neuronales recurrentes (*Recurrent Neural Networks*, RNN)

Las redes neuronales recurrentes son un tipo de redes neuronales artificiales basadas en el perceptrón multicapa, aunque no tienen una estructura de capas, sino que permiten conexiones arbitrarias entre todas las neuronas, incluso creando ciclos. Esto permite incorporar a la red el concepto de temporalidad, y permite que la red tenga memoria, porque los números que introducimos en un momento dado en las neuronas de entrada son transformados, y continúan circulando por la red incluso después de cambiar los números de entrada por otros diferentes [22].

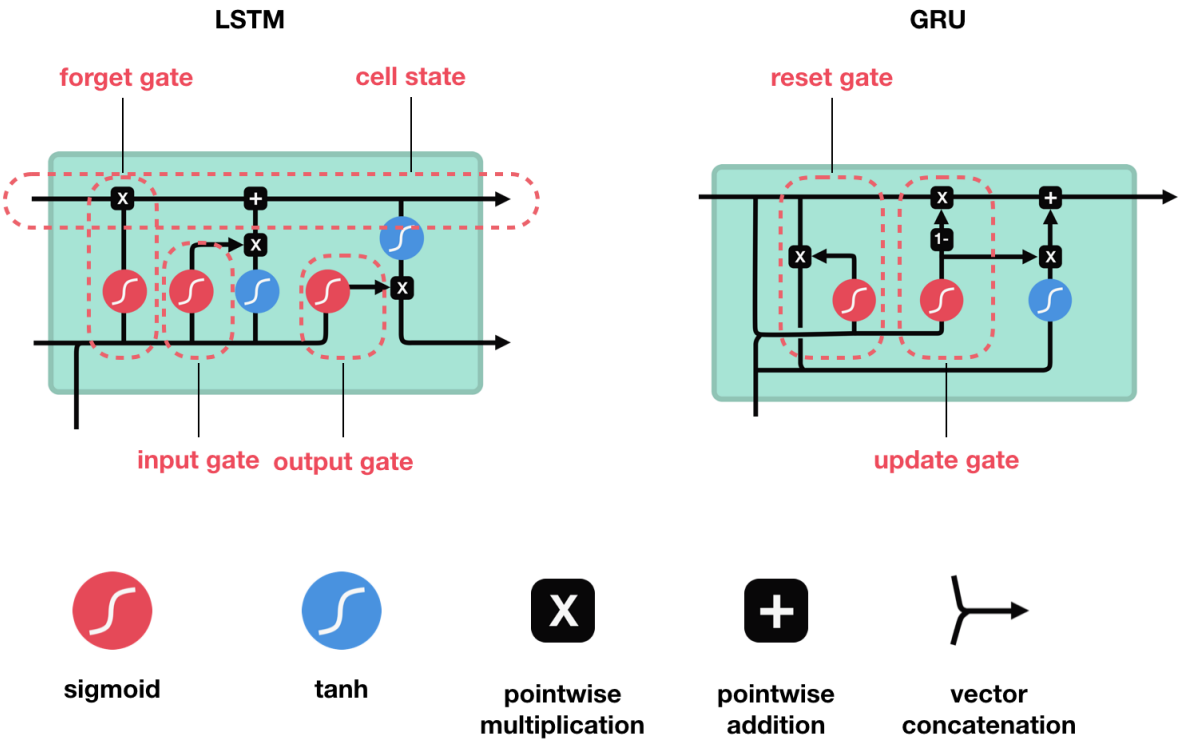



Figura 4.5: Celdas especiales para redes neuronales recurrentes [163].

Las RNN más simples no tienen un estado celular, solo tienen estados ocultos y dichos estados ocultos sirven

como memoria para RNNs, debido a que tienen bucles de retroalimentación en la capa recurrente. Este tipo de redes, debido a que tienen un estado interno que hemos denominado memoria, se utiliza para procesar entradas secuenciales, lo que hace de ellas una opción de red aplicable a tareas como el reconocimiento de voz o el reconocimiento de patrones de escritura manuscrita. Pero, puede ser difícil capacitar a las RNN estándar para resolver problemas que requieren el aprendizaje de dependencias temporales a largo plazo. Esto se debe al problema del gradiente de fuga lo que significa que el gradiente de la función de pérdida decae exponencialmente con el tiempo y por lo tanto debamos recurrir a las memorias largas a corto plazo (*Long Short-Term Memory*, LSTM) [164] o las unidades recurrentes cerradas (*Gated Recurrent Unit*, GRU) [165].

Las redes con LSTM son un tipo de RNN que a menudo se denominan RNN de lujo y tiene estados celulares y estados ocultos mediante la utilización de unidades especiales además de las unidades estándar. Las unidades LSTM a través del estado de la "celda de memoria" tienen la capacidad de eliminar o agregar información a la celda, regulada por "puertas". Estas compuertas denominadas a menudo como las puertas de entrada y olvido, se utilizan para controlar cuándo la información es ingresada a la memoria, cuándo se envía y cuándo se olvida, lo cual permite un mejor control sobre el flujo de gradiente y permiten una mejor conservación de las "dependencias de largo plazo". Debido a esta "célula" o "celda de memoria", puede mantener la información en la memoria por largos períodos de tiempo, y por ende esta arquitectura les permite aprender "dependencias a largo plazo". Las GRUs son similares a las LSTM, pero usan una estructura simplificada. También usan un conjunto de puertas para controlar el flujo de información, pero no usan celdas de memoria separadas, y usan menos puertas [166]. Estos dos tipos de celdas especiales las podemos ver en la Figura 4.5.

### 4.2.4. Redes neuronales convolucionales (*Convolutional Neural Networks*, CNN)

Las redes neuronales convolucionales mantienen el concepto de capas, pero cada neurona de una capa no recibe conexiones entrantes de todas las neuronas de la capa anterior, sino sólo de algunas. Esto favorece que una neurona se especialice en una región de la lista de números de la capa anterior, y reduce drásticamente el número de pesos y de multiplicaciones necesarias. Lo habitual es que dos neuronas consecutivas de una capa intermedia se especialicen en regiones solapadas de la capa anterior [22].

Son redes que se usan para procesar imágenes simulando el comportamiento de las neuronas en la corteza visual primaria de un cerebro biológico, tienen la capacidad de aprender relaciones entrada-salida, donde la entrada es una imagen y están basadas en operaciones de convolución. Las tareas comunes son la detección/categorización de objetos, la clasificación de escenas y la clasificación de imágenes en general [167]. Los tipos de capas que presento son:

- **Capa convolucional**: Esta capa se basa en la operación de convolución que consiste en filtrar una imagen usando una máscara, dando lugar a distintos resultados en función de las mascaras utilizadas. En la convolución, cada píxel de salida es una combinación lineal de los píxeles de entrada como podemos ver en la Figura 4.6, ya que se realizan operaciones de suma y multiplicación entre la señal de entrada de dicha capa y los n filtros o *kernels*, para generar un mapa de características correspondientes a la posible ubicación del filtro en la imagen original. Las máscaras representan la conectividad entre las capas sucesivas, siendo cada capa un volumen de neuronas en 3D con las dimensiones de alto, ancho y profundidad de la capa [167]. Una de las principales ventajas que ofrece esta capa es la posibilidad de reducir el número de conexiones y el número de parámetros a entrenar en comparación con el perceptrón multicapa totalmente conectado [133].

  Después de haber realizado la operación de convolución y normalmente con el objetivo de aumentar las propiedades no lineales, se suelen emplear capas de activación como las vistas en la sección 4.1.2.

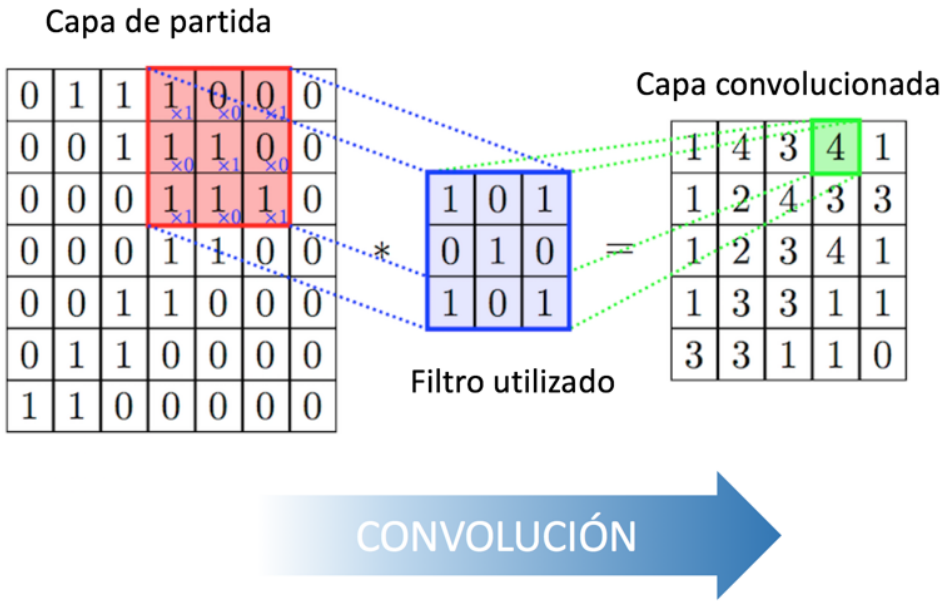



Figura 4.6: Funcionamiento de la operación de convolución [168].

- **Capa de *pooling* o reducción**: La salida es el máximo de la entrada en una ventana en el caso de *Max Pooling* y la media en el caso de *Average Pooling*, desplazando la ventana por la matriz de datos según una configuración del paso o *stride* como vemos en el ejemplo de la Figura 4.7. Se puede usar para submuestrear mediante la reducción del tamaño espacial de los datos, con el objetivo de quedarse solo con aquellas características que sean más comunes.

  Al realizar esta operación, el sistema se vuelve ligeramente menos preciso, pero presenta muchos beneficios como la prevención de un posible sobreajuste y mejora de su compatibilidad, debido a la reducción de las características a analizar, favoreciendo la obtención de una representación invariable a pequeñas traslaciones en la entrada

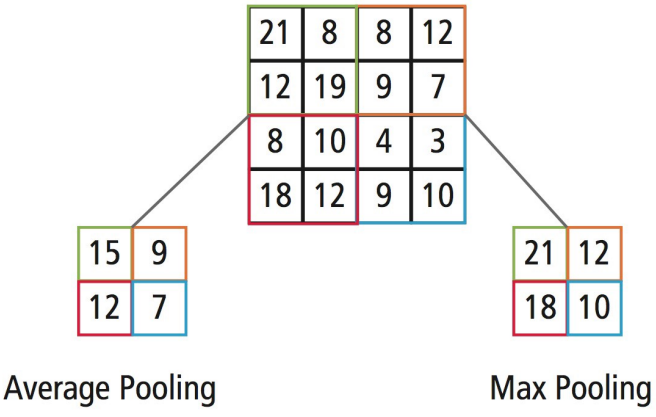



Figura 4.7: Funcionamiento de la operación de *Max Pooling* y *Average Pooling* con paso 2 y ventana de 2x2 [169].

- **Capa *fully connected* o totalmente conectada**: Este tipo de capas se aplican al final cuando la información espacial previamente se ha perdido con una capa *flatten* que aplana los datos. Además combina las carac-

terísticas aprendidas por las capas anteriores para identificar patrones mas grandes. Tiene una función de activación, normalmente la función de activación ReLU, aunque si es la última capa se aplica la función *softmax* vista en la sección 4.1.2 en problemas de clasificación para obtener las probabilidades de cada clase posible.

Además podríamos decir que hay dos arquitecturas básicas de CNN. La primera de ellas sería la CNN básica que entrega una salida para toda la imagen, siendo este tipo de redes muy similares a los perceptrones multicapa o redes neuronales prealimentadas y su principal diferencia radica en que realiza el aprendizaje de características utilizando la operación de convolución [133]. La segunda sería, la de las redes totalmente convolucionales (*Fully Convolutional Networks*, FCN) que tienen un *encoder* y un *decoder*, comprimen la información y entregan una salida por píxel [167], ya que es un tipo de *autoencoder*, como el visto en la Figura 4.4.

## 4.3. *Data augmentation*

La técnica de *data augmentation* es una técnica utilizada en las redes neuronales para aumentar el número de ejemplos con los que alimentamos a la red. Cuando se entrena un modelo de aprendizaje automático, lo que realmente se está haciendo es ajustar sus parámetros para que pueda asignar una entrada particular (*e.g.*, una imagen) a alguna salida (una etiqueta). Para optimizar nuestro modelo buscamos el punto óptimo donde la pérdida de nuestro modelo es baja, lo que sucede cuando sus parámetros se ajustan de la manera correcta.

Debido a que las redes neuronales suelen tener muchos parametros, debemos mostrar a dicha red una cantidad proporcional de ejemplos para que el rendimiento sea óptimo. Además la cantidad de parametros que se necesita es proporcional a la complejidad de la tarea que debe realizar su modelo.la complejidad de la tarea que debe realizar su modelo.

A partir de ahora nos centraremos en como realizar *data augmentation* en una CNN. Para agregar nuevas imágenes a nuestro conjunto de datos no es necesario siempre buscar nuevas imágenes. Debido a que, una red neuronal mal entrenada observaría a las estatuas que se muestran en la Figura 4.8 y pensaría que todas las estatuas son distintas. Pero solo se ha variado la iluminación de la estatua, posteriormente veremos otras pequeñas modificaciones o pequeños cambios como la translación, rotación, cambio del punto de vista o variación de tamaño.


Figura 4.8: Estatua con diferentes iluminaciones

Una red neuronal convolucional que puede clasificar de manera robusta los objetos incluso si se coloca en diferentes orientaciones tiene la propiedad llamada invariancia [170]. Más específicamente, una CNN puede ser invariable para translación, punto de vista, tamaño o iluminación (o una combinación de lo anterior).

Nosotros solo podemos obtener un conjunto de datos de imágenes con un conjunto limitado de condiciones. Pero, nuestra aplicación puede experimentar una gran variedad de condiciones diferentes, como orientación, ubicación, escala, brillo, etc. Podemos introducir estas situaciones al entrenar nuestra red neuronal con datos adicionales modificados sintéticamente.

Incluso si tenemos una gran cantidad de datos puede ser necesario el uso de la técnica de *data augmentation*, ya que al realizar el aumento de datos, puede evitar que la red neuronal aprenda patrones irrelevantes. Lo que

esencialmente aumenta el rendimiento general, ya que sino la red neuronal encuentra las características más obvias que distinguen una clase de otra.

Podemos realizar *data augmentation* en dos momentos diferentes:

- **Aumento sin conexión**: Se realizan todas las transformaciones necesarias de antemano, aumentando el tamaño de su conjunto de datos. Este método se prefiere para conjuntos de datos relativamente pequeños, ya que se aumenta el tamaño del conjunto de datos en un factor igual al número de transformaciones que realiza.
- **Aumento en línea o aumento sobre la marcha**: Se realizan estas transformaciones en un mini lote, justo antes de alimentarlo al modelo de aprendizaje automático. Este método es preferido para conjuntos de datos más grandes, en los que no se puede permitir el aumento explosivo de tamaño. En su lugar, se realizan transformaciones en los mini lotes que alimentan a su modelo. Algunos *frameworks* de aprendizaje automático tienen soporte para el aumento en línea, que se puede acelerar en la GPU.

### 4.3.1. Técnicas básicas de *data augmentation*

Para esta sección vamos a suponer que no necesitamos considerar qué hay más allá del límite de la imagen. Ya que si utilizando *data augmentation* necesitamos saber que hay más allá del limite de la imagen, necesitaremos interpolar parte de la información. Discutiremos esto en detalle después de cubrir los tipos de *data augmentation* en la sección 4.3.3. Las técnicas básicas de *data augmentation* son:

- **Girar o voltear**: Se pueden girar las imágenes horizontal y verticalmente. Algunos *frameworks* no proporcionan función para giros verticales. Pero, un giro vertical es equivalente a rotar una imagen 180 grados y luego realizar un giro horizontal.
- **Rotar**: Con la operación de rotación debemos de tener en cuenta que las dimensiones de la imagen pueden no conservarse después de la rotación. Si la imagen es un cuadrado, al rotarla en ángulo recto se conservará el tamaño de la imagen. Si es un rectángulo, al rotarlo 180 grados conservaría el tamaño. Girar la imagen en ángulos más finos también cambiará el tamaño final de la imagen. Veremos cómo podemos tratar este problema en la sección 4.3.3. Podemos ver ejemplos de rotaciones y cambios de punto de vista en la Figura 4.9.


Figura 4.9: Estatua con diferentes rotaciones y puntos de vista.

- **Escalar o cambiar el tamaño**: La imagen se puede escalar hacia afuera o hacia adentro como observamos en la Figura 4.10. Cuando se escala hacia afuera, el tamaño de la imagen final será mayor que el tamaño de la imagen original. La mayoría de los *frameworks* de imágenes recortan una sección de la nueva imagen, con un tamaño igual a la imagen original. En la sección 4.3.3, trataremos de escalar hacia adentro, ya que reduce el tamaño de la imagen, lo que nos obliga a hacer suposiciones sobre lo que se encuentra más allá del límite.


Figura 4.10: Estatua con diferentes escalas.

- **Recortar**: A diferencia de la escala, muestreamos al azar una sección de la imagen original. Posteriormente cambiamos el tamaño de esta sección al tamaño de la imagen original. Este método se conoce popularmente como recorte aleatorio. A veces, puede ser difícil de notar la diferencia entre este método y la escala.
- **Trasladar**: La traslación solo implica mover la imagen a lo largo de la dirección X o Y (o ambas) como vemos en la Figura 4.11. Este método de aumento es muy útil ya que la mayoría de los objetos se pueden ubicar en casi cualquier lugar de la imagen. Esto obliga a la red neuronal convolucional a buscar en todas partes.


Figura 4.11: Estatua con diferentes traslaciones.

- **Ruido gaussiano**: El sobreajuste generalmente ocurre cuando la red neuronal intenta aprender características de alta frecuencia (patrones que ocurren mucho) que pueden no ser útiles. El ruido gaussiano, que tiene una media cero, esencialmente tiene puntos de datos en todas las frecuencias, distorsionando efectivamente las características de alta frecuencia. Esto también significa que las componentes de baja frecuencia (generalmente, los datos deseados) también están distorsionados, pero su red neuronal puede aprender a mirar más allá. Agregar la cantidad justa de ruido puede mejorar la capacidad de aprendizaje.
- **Ruido sal y pimienta**: Una versión de diferente naturaleza al ruido gaussiano, que se presenta como píxeles aleatorios en blanco y negro diseminados por la imagen. Esto es similar al efecto producido al agregar ruido gaussiano a una imagen, pero puede tener un nivel de distorsión de información más bajo.

### 4.3.2. Técnicas avanzadas de *data augmentation*

Los datos que presentan las imágenes de la naturaleza pueden existir en una variedad de condiciones que no pueden explicarse por los métodos simples anteriores. Por ejemplo si nuestra red neuronal busca clasificar tundras heladas, praderas, bosques, etc., aunque parece una tarea de clasificación sencilla, en realidad no lo es, ya que afectaría el rendimiento la temporada en la que se tomó la fotografía.

Si la red neuronal no comprende el hecho de que ciertos paisajes pueden existir en una variedad de condiciones (nieve, humedad, brillo, etc.), puede etiquetar erróneamente las lagunas congeladas como glaciares o los campos húmedos como pantanos.

Una forma de mitigar esta situación es agregar más imágenes para que tengamos en cuenta todos los cambios estacionales. Pero esa es una tarea ardua. Pero podemos utilizar el concepto de aumento de datos utilizando redes generativas adversarias (*Generative Adversarial Net*, GAN) condicionales.

Las GANs condicionales pueden transformar una imagen de un dominio a una imagen en otro dominio. Un ejemplo de uso es transformar fotografías de paisajes de verano en paisajes de invierno como se muestra en la Figura 4.12.

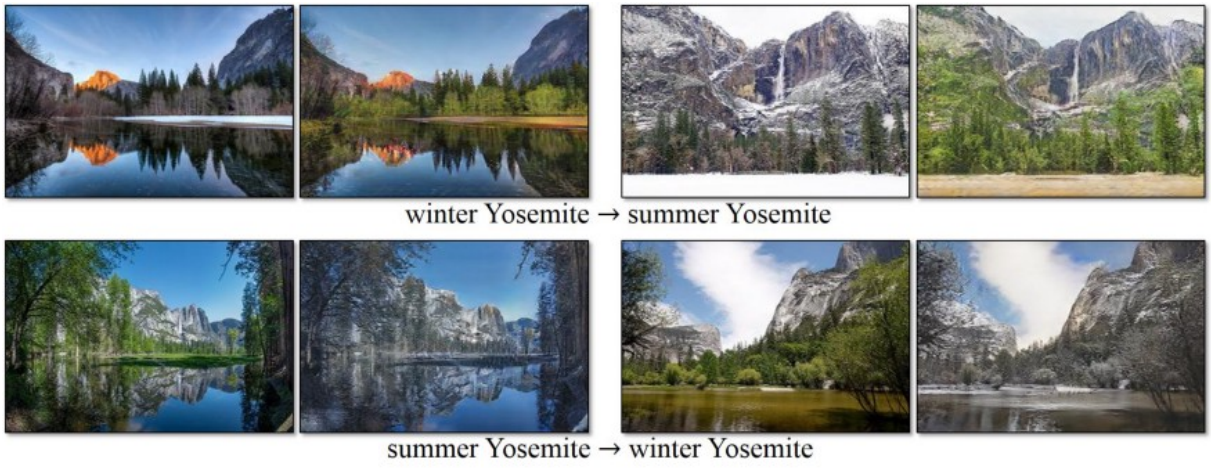



Figura 4.12: Cambio de estaciones usando un CycleGAN [171].

Este método es robusto pero computacionalmente costoso. Una alternativa más liviana sería utilizar la transferencia de estilo neuronal. Coge la textura/ambiente/apariencia de una imagen (también conocido como el “estilo”) y la mezcla con el contenido de otra. Usando esta poderosa técnica, producimos un efecto similar al de nuestra GAN condicional.

El único inconveniente de este método es que la salida tiende a parecer más artística en lugar de realista. Sin embargo, hay ciertos avances, como *Deep Photo Style Transfer* [172], cuyo resultado es impresionante y se puede comprobar en la Figura 4.13.


Figura 4.13: Transferencia de estilo de foto [172].

### 4.3.3. Interpolación

Dado que nuestra imagen no tiene ninguna información sobre cosas fuera de sus límites, debemos hacer algunas suposiciones si se desea trasladar una imagen que no tiene un fondo negro, si quisieras escalar hacia adentro o si se desea rotar en ángulos más finos. Esto se debe a que, debemos preservar nuestro tamaño de imagen original. Por lo general, se supone que el espacio más allá del límite de la imagen es la constante 0 en cada punto. Por lo tanto, cuando realiza estas transformaciones, obtiene una región negra donde la imagen no está definida.

Pero esta suposición solo se puede hacer en algunos casos. El procesamiento de imágenes y los marcos de Machine Learning tienen algunas formas estándar con las que puede se puede decidir cómo llenar el espacio desconocido. Se definen de la siguiente manera:

- **Constante**: El método de interpolación más simple es llenar la región desconocida con algún valor constante (*e.g.*, cero). Esto puede no funcionar para imágenes naturales, pero puede funcionar para imágenes tomadas en un fondo monocromático.
- **Borde**: Los valores de borde de la imagen se extienden después del límite. Este método puede funcionar para traslaciones leves.
- **Reflexión**: Los valores de píxeles de la imagen se reflejan a lo largo del límite de la imagen. Este método es útil para fondos continuos o naturales que contienen árboles, montañas, etc.
- **Simetría**: Este método es similar a la reflexión, excepto por el hecho de que, en el límite de la reflexión, se realiza una copia de los píxeles del borde. Normalmente, reflexión y simetría se pueden usar indistintamente, pero las diferencias serán visibles al tratar con imágenes o patrones muy pequeños.

- **Envolver**: La imagen solo se repite más allá de su límite como si estuviera en mosaico. Este método no se usa tan popularmente como el resto, ya que no tiene sentido para muchos escenarios.

Además de estos, se pueden utilizar otros métodos propios para tratar con espacios indefinidos, pero por lo general estos métodos funcionan bien para la mayoría de los problemas de clasificación.

## 4.4. *Transfer learning* y arquitecturas de CNNs pre-entrenadas

En esta sección hablaremos sobre arquitecturas de redes neuronales que con un entrenamiento previo se pueden reutilizar como punto de partida para un modelo en una segunda tarea, normalmente similar a la que han sido entrenadas, mediante la técnica del aprendizaje por transferencia (*Transfer Learning*, TL). En las tareas de procesamiento de lenguaje natural y visión computacional es un enfoque popular en el aprendizaje profundo o *Deep Learning* utilizar modelos pre-entrenados como punto de partida y por lo tanto usar la técnica de *Transfer Learning*. Esto se debe a que para desarrollar modelos de redes neuronales en estos problemas, se requieren vastos recursos de cómputo, datos y tiempo, y gracias a esta técnica se obtienen grandes saltos de rendimiento sobre problemas relacionados.

En el aprendizaje por transferencia, primero entrenamos una red base en un conjunto de datos y una tarea base, y luego reutilizamos las funciones aprendidas, o las transferimos, a una segunda red de destino para recibir las capacidades aprendidas en un conjunto de datos y una tarea objetivo. Este proceso tenderá a funcionar si las características son generales, lo que significa que son adecuadas para las tareas base y de destino, en lugar de ser específicas para la tarea base. Por lo tanto aquí es donde adquieren mayores beneficios los modelos, al utilizar un ajuste de modelo en una tarea diferente pero relacionada, denominándose esta forma de aprendizaje por transferencia utilizada en el aprendizaje profundo, transferencia inductiva.

Hay que tener en cuenta que el *Transfer Learning* tiene tres posibles beneficios y lo ideal sería ver los tres beneficios cuando se realiza *Transfer Learning*, ya que significaría que ha sido exitoso este método, siendo estos tres beneficios:

- **Comienzo superior**: El *accuracy* inicial del modelo de origen antes de entrenarlo es mayor de lo que sería de otra manera.

- **Mayor pendiente**: La tasa de mejora del *accuracy* durante el entrenamiento del modelo original es más pronunciada de lo que sería de otra manera.

- **Asíntota superior**: El *accuracy* del modelo entrenado converge mejor de lo que lo haría de otra manera.

Aunque no se de alguno de estos beneficios, puede que el *Transfer Learning* sea igualmente existoso, siempre que su asíntota sea superior, debido a que es el beneficio más importante [173].

Tenemos varias arquitecturas de CNNs pre-entrenadas disponibles para usar la técnica de *Transfer Learning*, aunque solo vamos a hablar de algunas de las más relevantes, del desafío de reconocimiento visual de gran escala de ImageNet (*ImageNet Large Scale Visual Recognition Challenge*, ILSVRC) [174], las cuales las podemos ver en la Figura 4.14. El objetivo de este desafío de clasificación de imágenes es entrenar un modelo que pueda clasificar correctamente una imagen de entrada en mil categorías de objetos independientes. Estas mil categorías de imágenes representan clases de objetos que encontramos en nuestra vida cotidiana, como especies de perros, gatos, diversos objetos del hogar, tipos de vehículos y mucho más. Los modelos se entrenan con 1.2 millones de imágenes, 50 mil imágenes para validación y 100 mil imágenes para pruebas.

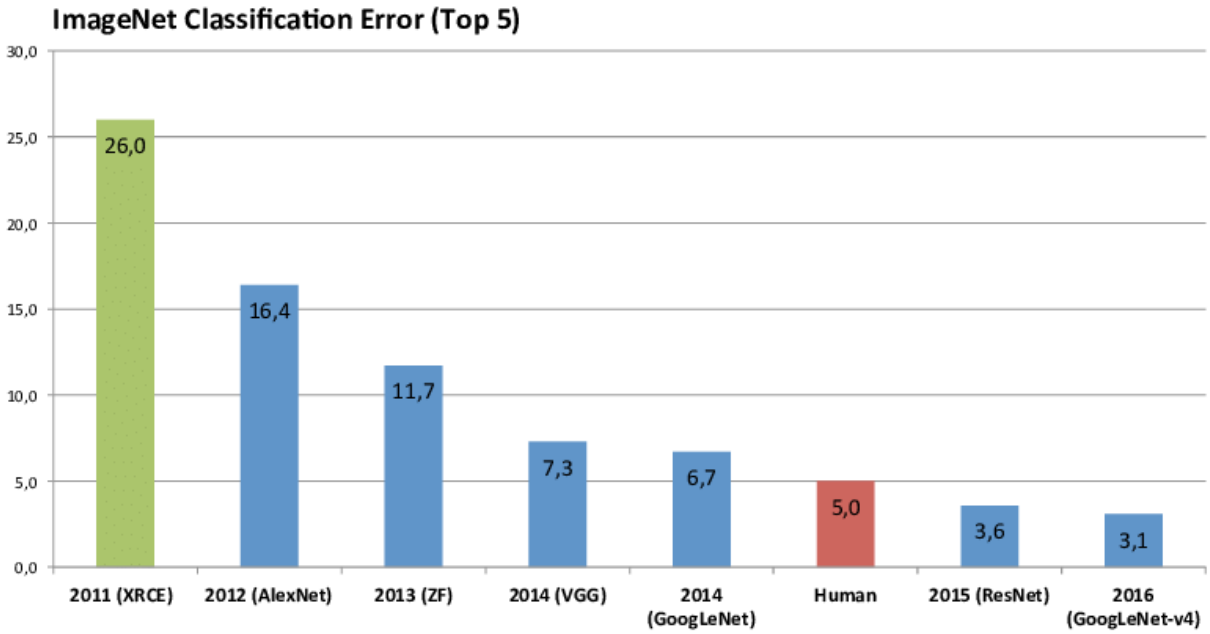



Figura 4.14: *ImageNet Large Scale Visual Recognition Challenge* (ILSVRC).

Cuando escuchamos el término "ImageNet" en el contexto de aprendizaje profundo y redes neuronales convolucionales, es probable que nos estemos refiriendo al ILSVRC, de manera coloquial, aunque ImageNet es el proyecto destinado a etiquetar (manualmente) y categorizar imágenes en casi 22,000 categorías de objetos separados con el fin de investigar la visión por computadora. Cuando se trata de la clasificación de imágenes, el desafío de "ImageNet" es el punto de referencia de facto para los algoritmos de clasificación de visión artificial, y la clasificación para este desafío ha estado dominada por las redes neuronales convolucionales y las técnicas de aprendizaje profundo desde 2012. Estas redes demuestran una gran capacidad para generalizar imágenes fuera del conjunto de datos de "ImageNet" a través del *Transfer Learning*, como la extracción de características y el ajuste fino.

AlexNet-8 [175] superó significativamente a todos los competidores anteriores y ganó el desafío de 2012, la cual redujo el error (top-5) del 26 % al 16,4 %. Esta red tenía una arquitectura muy similar a la red LeNet, una red convolucional pionera de 1998 con 7 niveles, que clasifica los dígitos para reconocer números escritos a mano en cheques digitalizados en imágenes de escala de grises de 32x32 píxeles, realizada por Yann LeCun et al, pero era más profunda, con más filtros por capa y con capas convolucionales apiladas. Consistió en *data augmentation* intensivo con capas convolucionales 11x11, 5x5 y 3x3, *max pooling*, *dropout* de 0,5 y funciones de activación ReLU. Tenía el optimizador de descenso de gradiente estocástico (*Stochastic Gradient Descent*, SGD) con momentum de 0,9, *learning rate* de 0,01 (reducido por un factor de 10 cuando no mejoraba desempeño en conjunto de validación) y regularizador para reducción de peso buscando reducir el sobreajuste de la red neuronal con la suma de los pesos al cuadrado (L2) de 0,0005. Después de cada capa convolucional y totalmente conectada tenia funciones de activación ReLU. AlexNet fue entrenada durante 6 días simultáneamente en dos GPU Nvidia Geforce GTX 580, lo que explica la razón por la cual su red se divide en dos canales y fue diseñada por el grupo SuperVision, que consiste en Alex Krizhevsky, Geoffrey Hinton e Ilya Sutskever.

ZFNet-8 fue la ganadora en 2013, y que fuera una CNN ya no fue sorprendente. Redujo el error (top-5) del 16,4 % al 11,7 % y fue principalmente un logro conseguido al ajustar los hiper parámetros de AlexNet.

VGGNet-19 [176] llegó a ser subcampeón en 2014 y fue desarrollado por Simonyan y Zisserman. Similar a AlexNet, con capas convolucionales 3x3 de paso 1, pero muchos filtros, *max pooling* 2x2 de paso 2, *dropout* de 0,5 y capa de salida *softmax*. Se utilizan filtros de convolución más pequeños ya que, que el campo receptivo efectivo de tres convoluciones de 3x3 es igual al de una convolución de 7x7, además se introducen no linealidades intermedias y utiliza menos parámetros $[3 \cdot (3^2 \cdot C^2) < 7^2 \cdot C^2]$. Aplicó también técnicas de *data augmentation* y su entrenamiento se llevo a cabo mediante el optimizador SGD con momentum de 0,9, *learning rate* de 0,1

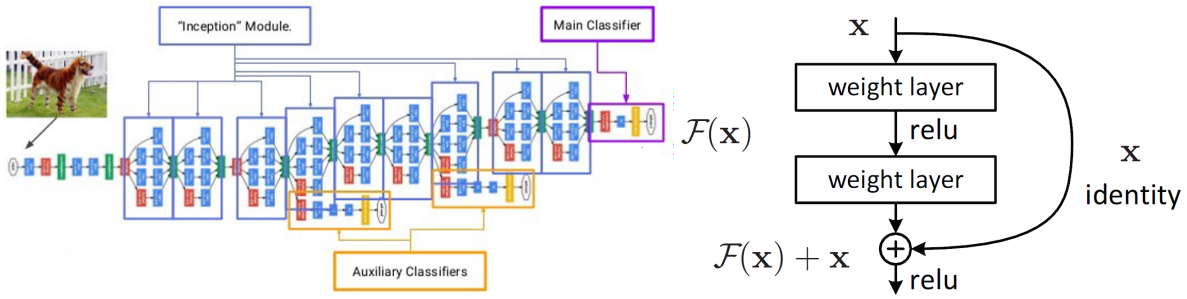
(reducido 3 veces por un factor de 10) y regularizador para reducción de peso buscando reducir el sobreajuste de la red neuronal con la suma de los pesos al cuadrado (L2) de 0,0005. Además realizó una inicialización aleatoria con distribución normal de media cero y varianza 0,01, 74 *epochs* (370000 iteraciones) y *batch size* de 128.

Simonyan y Zisserman consideraron que el entrenamiento de VGGNet era desafiante (específicamente en relación con la convergencia en las redes más profundas), por lo que para facilitar el entrenamiento, primero entrenaron versiones más pequeñas de VGGNet con menos capas de peso. Las redes más pequeñas convergieron y luego se utilizaron como inicializaciones para las redes más grandes y profundas, este proceso se denomina pre-entrenamiento . Si bien tiene sentido lógico, el entrenamiento previo es una tarea tediosa y que requiere mucho tiempo, ya que requiere que toda la red esté capacitada antes de que pueda servir como una inicialización para una red más profunda.

Desafortunadamente, hay dos grandes inconvenientes con VGGNet, es dolorosamente lento para entrenar (para el desafío se entrenó la red en 4 GPUs por 2-3 semanas) y los pesos de la arquitectura de la red en sí son bastante grandes (en términos de ancho de banda/disco). La configuración de peso de VGGNet está disponible públicamente y se ha utilizado en muchas otras aplicaciones y desafíos. Aunque debido a su profundidad y el número de nodos totalmente conectados, VGG supera los 574 MB para VGG19 y consta de 138 millones de parámetros, esto hace que la implementación de esta red sea una tarea pesada y pueda llegar a ser un poco difícil de manejar.

GoogLeNet-22 [177], [178] fue el ganador del concurso ILSVRC 2014, también se conoce a esta red como Inception-v1 y fue desarrollada por Google. Logró una tasa de error (top-5) de 6,67 %, lo cual fue un desempeño muy cercano al nivel humano que los organizadores del desafío se vieron obligados a evaluar. Resulta que, en realidad, esto era bastante difícil de realizar y requería algún entrenamiento humano para poder superar la precisión de GoogLeNets. Después de unos días de entrenamiento, el experto humano (Andrej Karpathy) logró una tasa de error (top-5) del 5,1 % (modelo único) y el 3,6 % (conjunto). La red utilizó una CNN inspirada en LeNet, pero implementó un elemento novedoso que se denomina módulo de inicio.

Utilizó una neurona con un modelo más complejo NIN, que realizaba varios cálculos en paralelo y *pooling* sin reducir la imagen. Esta red fue diseñada para ser eficiente en términos de memoria y cómputo, y consiguió tener 1500 millones de sumas y multiplicaciones, además aunque su arquitectura consistía en una CNN de 22 capas de profundidad, redujo el número de parámetros de 60 millones (AlexNet) a 4 millones, es decir, consiguió tener 15 veces menos parámetros que AlexNet. Esta red tenía un modulo especial que se basaba en varias convoluciones muy pequeñas para reducir drásticamente el número de parámetros mediante un filtrado multi-escala con capas convolucionales de 1x1, 3x3 y 5x5 y pooling de 3x3, cuyas salidas se concatenan en un solo volumen, por lo tanto la arquitectura final no es simplemente una sucesión de capas convolucionales [167], ya que se usan varias resoluciones. La primera etapa es una arquitectura clásica con fuerte reducción de resolución espacial por razones de eficiencia computacional y en la etapa de clasificación utiliza una capa *Global Average Pooling*. También tiene clasificadores auxiliares diseñados para combatir el *vanishing gradient* durante el entrenamiento que son descartados después del entrenamiento. Se entrenó realizando distorsiones de imagen, normalización por lotes y el optimizador RMSprop. Actualmente tenemos las nuevas versiones basadas en esta red denominadas Inception-v3 [179] e Inception-v4 [180].




(a) Arquitectura de GoogLeNet

(b) *Skip connections* de ResNet [181].

Figura 4.15: Avances importantes de GoogLeNet y ResNet.

ResNet-152 [181] ganó el concurso ILSVRC de 2015, ResNet significa Red Neuronal Residual, fue desarrollada por Kaiming He *et al.* e introdujo una arquitectura novedosa con *skip connections*, además de presentar una normalización de lotes pesados. Alcanza una tasa de error del (top-5) de 3,57 %, lo que supera el rendimiento a nivel humano en este conjunto de datos. Pudieron entrenar a una CNN con 152 capas y al mismo tiempo tener una menor complejidad que VGGNet, pero el mayor problema es que al agregar más de 30 capas, el error comienza a aumentar en vez de disminuir y esta es la causa de que haya que usar modelos complejos como GoogleNet. La solución a este problema son dichas conexiones de salto o *skip connections* que también se conocen como unidades cerradas o unidades recurrentes cerradas y tienen una gran similitud con las conexiones que se aplican en una RNN. La idea es que las capas extras reduzcan efectivamente el error trabajando con residuales, y por lo tanto ResNets más profundas tengan un error de entrenamiento más bajo y también un error de prueba más bajo. A diferencia de GoogLeNet, el diseño es simple, sólo se usan capas convolucionales y *max poolings*. El entrenamiento se realizó mediante redes residuales entrenadas desde cero, utilizando normalización por lotes, *data augmentation* e hiperparámetros estándar. El uso de residuales permite usar muchas capas (actualmente 1001 capas) [167] pudiendo ser entrenadas sin dificultades.

Por lo tanto, las principales diferencias son que AlexNet tiene dos líneas CNN paralelas entrenadas en dos GPU con conexiones cruzadas, GoogleNet tiene módulos de inicio, ResNet tiene conexiones residuales [182], y podemos ver la comparativa del rendimiento de estas redes, además de otras también importantes en la Figura 4.16.

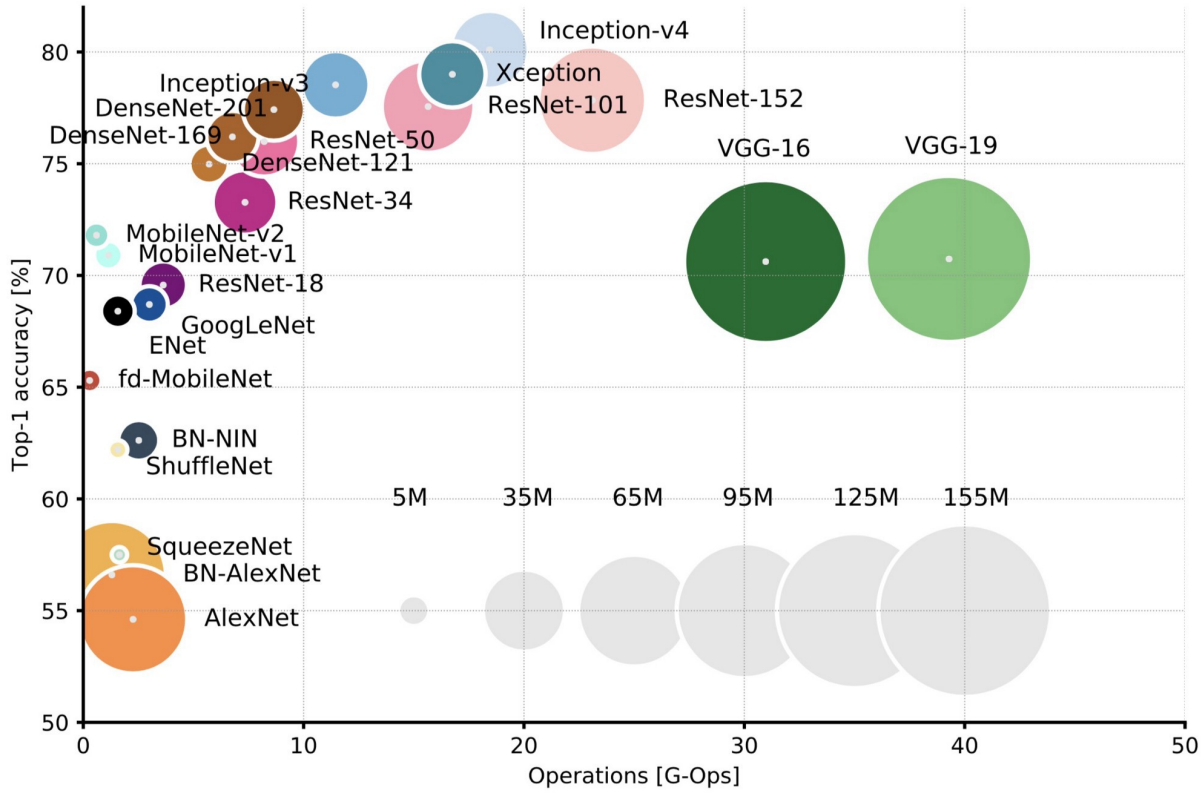



Figura 4.16: Comparativa del rendimiento de las redes neuronales más importantes [183].

# *Capítulo 5*

# DETECCIÓN Y ANÁLISIS DE LA SEÑAL DE TOS

## 5.1. Segmentación del audio de la tos con *machine hearing*

En esta sección se presenta una descripción de los motores de reconocimiento de patrones recientes para la detección de tos basados en grabaciones de audio. Dicha descripción para entender su funcionamiento incluye módulos como los de preprocesamiento, diseño de características, esquema de clasificación y posprocesamiento.

**Detección de tos mediante el conjunto de características de subbanda de frecuencia múltiple**

En marzo de 2017, Mingyu You *et al.* [184] desarrollaron un detector de tos, basado en las siguientes etapas:

- **Preprocesado**: las señales originales de 44,1 kHz son diezmadas recibiendo un submuestreo o *downsampling* para tener la nueva tasa de muestreo de 16 kHz y se segmentan utilizando *frames* de 20 ms con un desplazamiento de 10 ms. Por lo tanto, es un enfoque basado en *frames* a corto plazo.

- **Cálculo de características**: para explotar las características únicas de la distribución espectral de la tos en comparación con el habla, los autores definieron un banco de filtros de Gammatone en el que el ancho de banda de cada filtro es igual a 300 Hz. Además, las frecuencias centrales están espaciadas uniformemente entre 80 Hz y 6 kHz y el número de filtros es 20. La diferencia principal entre esta propuesta y un banco de filtros Gammatone común utilizado en los GTCC es que en este último el ancho de banda a bajas frecuencias es menor que en altas frecuencias. Por lo tanto, enfatiza las características de la región de baja frecuencia, que es útil en el análisis del habla. Después de aplicar el banco de filtros a las señales, tienen 20 señales basadas en subbanda. Esta es la segunda diferencia con respecto a los enfoques clásicos (*e.g.*, MFCC o GTCC), donde el banco de filtros se usa directamente para la extracción de características en lugar de crear varias versiones de la señal basadas en subbandas, una para cada filtro utilizado. En este punto, cada versión puede caracterizarse por cualquier método común de extracción de características. En particular, You *et al.* emplearon GTCC común [108].

- **Esquema de clasificación**: los autores entrenan 20 clasificadores SVM lineales basados en el GTCC de cada señal basada en subbandas. Más tarde, dos métodos tienen como objetivo combinar la salida de estos clasificadores. Ambos se basan en una metodología de votación.

    - El primer método se basa en las etiquetas de clase que ofrece el clasificador SVM en su salida. Si T o más SVM han clasificado la observación desconocida como un evento de tos, la clase final es "tos", de lo contrario, es "no tos". Experimentaron utilizando diferentes valores de T, que oscilaron entre 1 y 20. Los mejores resultados se obtuvieron para T = 9.

- El segundo explota la distancia (p) entre los vectores de soporte y el hiperplano. Si esta distancia es positiva, el SVM clasifica la entrada como tos y viceversa. La salida de distancia SVM de las mejores K subbandas se selecciona y se calcula su suma. Si esta suma es positiva, la observación finalmente se clasifica como "tos" y viceversa. El valor de K se estableció experimentalmente en 14.

Los autores utilizan tanto datos sintéticos como datos de pacientes reales en el desarrollo del detector de tos. Los datos sintéticos se generan mezclando eventos de tos limpia con diferentes tipos de ruido en diversas relaciones SNR. Los tipos de sonidos ruidosos son el piar de las aves, las sirenas, el ruido del tráfico, el ronquido y el habla, mientras que las SNR seleccionadas son -10, -5, 0, 5 y 10 dB. El 66 % de las muestras se utilizan para entrenar el modelo en cada condición ruidosa. La base de datos de señales del paciente está compuesta por nueve pacientes con neumonía, asma bronquial y/o EPOC. Las grabaciones se adquirieron mientras los pacientes repetían sus rutinas diarias en el entorno hospitalario. El dispositivo de grabación era una grabadora digital portátil (*SONY ICD-LX30*) y un micrófono (*ECM-CS10*) pegado al cuello de los pacientes. Aunque la duración total de la base de datos es de 1.440 minutos, solo se emplean 120.000 *frames* en los experimentos, que corresponden aproximadamente a 20 minutos. Además, el desequilibrio existente entre clases se compensa manualmente, de modo que la mitad de ellos son *frames* de la clase positiva. En este experimento se utiliza un *10-fold cross-validation*.

Los resultados ofrecían una sensibilidad del 79,2 % y una especificidad del 86,8 %. Además, los autores utilizaron datos sintéticos para comparar su propuesta con otros conjuntos de características: MFCC, predicción lineal perceptiva (*Perceptual Linear Prediction*, PLP), y transformada espectral relativa y predicción lineal perceptiva (*Relative Spectral Transform and Perceptual Linear Prediction*, RASTA-PLP). Los resultados mostraron que su enfoque es generalmente mejor en las cinco condiciones de ruido analizadas.

**Redes neuronales profundas para identificar los sonidos de la tos**

En septiembre de 2016, Amoh y Odame [185] analizaron dos arquitecturas de aprendizaje profundo para la detección de la tos, siendo estas, una CNN y una RNN. La CNN proyecta el problema como una tarea de reconocimiento visual, donde la entrada es el espectrograma de los clips de audio pre-segmentados. En el caso de RNN, se enfrenta a un problema secuencial. El rendimiento en sujetos vistos por primera vez fue generalmente mejor para el modelo CNN, por lo que este será el que se describe en esta sección y se usará para la comparación que se realizará en la sección 7.3. El funcionamiento para este caso es:

- **Preprocesamiento**: los autores primero aplican un protocolo de admisión de *frames* para rechazar los *frames* de silencio y de baja energía. Después de este proceso, un audio con una duración mínima de 320 ms es seleccionado para el siguiente paso. La Transformada de Fourier de Tiempo Corto (*Short-Time Fourier Transform*, STFT), se calcula utilizando una longitud de FFT de 128 puntos y ventanas de 4 ms sin superposición. Luego, estos espectrogramas se dividen utilizando regiones de 64 ms con una superposición entre regiones de 16 ms. Estas regiones STFT y las etiquetas correspondientes son la entrada del modelo CNN.

- **Arquitectura CNN**: Amoh y Odame basaron su arquitectura en modelos CNN anteriores para el reconocimiento de dígitos manuscritos. Debido a la falta de una gran base de datos de información, simplificaron esta arquitectura. Su arquitectura se compone de dos capas convolucionales, dos capas de *max pooling* y dos capas totalmente conectadas. Cada capa convolucional tiene 16 filtros con funciones de activación ReLU. El tamaño de los *kernels* en la primera capa convolucional es 9x3, por lo que la resolución en la escala de tiempo es más precisa que la de la escala de frecuencia. La segunda capa convolucional tiene *kernels* de tamaño 5x3, con paso 1. Las capas de *max pooling* no afectan el eje de tiempo, ya que su tamaño es 2x1. El tamaño de la representación más profunda generada por las capas es 12x12. Esta representación alimenta las dos capas completamente conectadas y, finalmente, una capa clasificadora con función de activación *softmax* que proporciona la salida de clasificación para cada región de la STFT. Las capas totalmente conectadas tienen 256 neuronas con funciones de activación ReLU y emplean la regularización de *drop out* para controlar el sobreajuste.

El conjunto de datos empleado en este estudio está compuesto por señales de 14 voluntarios sanos, que proporcionaron un total de 627 ejemplos de tos en la base de datos. Se pidió a los sujetos que hablaran y produjeran

otros sonidos, tales como respiración o crepitación. Los autores desarrollaron un sensor de tos *ad-hoc* para adquirir las señales. Este sensor es un dispositivo portátil pegado al pecho por medio de un adhesivo de espuma de grado médico. Consiste en un transductor piezoeléctrico y un sistema electrónico de acondicionamiento de señales adicional que amplifica los sonidos respiratorios, atenúa el habla y elimina los sonidos ambientales. La frecuencia de muestreo fue originalmente de 44,1 kHz, pero las señales fueron posteriormente diezmadas a 16 kHz.

En cuanto a los hiper-parámetros de entrenamiento, se usó SGD con *momentum* de Nesterovs de 0,9, una tasa de aprendizaje (*learning rate*) de 0,01 y un tamaño de lote (*batch size*) de 20. Se usó *10-fold cross-validation* para la evaluación. Algunos experimentos se centraron solo en el habla frente a la tos, mientras que otros se enfrentaron a una clasificación de varias clases (tos, habla y otros sonidos). Los autores compararon su propuesta con el clásico MFCC más un clasificador con función de activación *softmax*, un SVM y un modelo oculto de Márkov (*Hidden Markov Model*, HMM). Ambos modelos, CNN y RNN, mostraron un rendimiento equivalente o mejor. Finalmente, la capacidad de generalización se evalúa clasificando las observaciones de dos sujetos que la red ve por primera vez. En este experimento, la CNN informó de una sensibilidad del 82 % y una especificidad del 93,2 %, mientras que la RNN tuvo un peor desempeño (84,2 % y 75,2 %, respectivamente).

**Detección robusta de eventos de tos de audio utilizando momentos Hu locales**

En enero de 2019, Jesús Monge Álvarez *et al.* [106] propone el uso de momentos Hu locales como un conjunto de características robustas para la detección automática de tos en señales de audio adquiridas en teléfonos inteligentes. El sistema final alimentaba con las características extraídas un clasificador K-NN. Para evaluar adecuadamente el sistema en una diversidad de entornos ruidosos, se contaminaron los datos de audio de la tos real con una variedad de sonidos que incluyen ruido de ambientes interiores y exteriores y eventos que no son de tos (estornudos, risas, discursos, etc.). La base de datos creada permitió configuraciones flexibles de niveles de SNR entre sonidos de fondo y eventos (tos y no tos). Esta evaluación se complementó con datos reales de pacientes de una clínica ambulatoria. El sistema pudo detectar eventos de tos con alta sensibilidad (hasta 88,51 %) y especificidad (hasta 99,77 %) en una variedad de entornos ruidosos.

**Un sistema de audición de máquina para la detección robusta de tos basado en una representación de alto nivel de características de audio específicas de la banda**

En agosto de 2019, Jesús Monge Álvarez *et al.* [4] presentaron un sistema de audición de máquina para una segmentación de tos robusta basada en audio que se puede implementar fácilmente en escenarios móviles. La detección de la tos se realizó en dos pasos. Primero, un conjunto de características espectrales a corto plazo se calcularon por separado en cinco bandas de frecuencia predefinidas: [0; 0,5), [0,5; 1), [1; 1,5), [1,5; 2), [2; 5,5125] kHz. La selección y combinación de características se aplicaron para que el conjunto de características a corto plazo fuera lo suficientemente robusto en diferentes escenarios ruidosos. En segundo lugar, la representación de datos de alto nivel se logró al calcular la media y la desviación estándar de los descriptores a corto plazo en marcos de 300 ms a largo plazo. Finalmente, la detección de la tos se realizó mediante una SVM entrenada con datos de diferentes escenarios ruidosos. El sistema se evaluó utilizando una base de datos de señales de pacientes que emula tres escenarios de la vida real en términos de contenido de ruido. El sistema alcanzó una sensibilidad del 92,71 % y una especificidad del 88,58 %.

**Segmentación automática de la tos a partir de grabaciones de sonido sin contacto en salas pediátricas**

En agosto de 2015, Yusuf A. Amrulloh *et al.* [186] desarrollaron un motor de reconocimiento de patrones para realizar la detección de eventos de tos en ficheros de audio. Siguieron un enfoque a corto plazo basado en medidas no gaussianas, frecuencias que conformaban el audio, cruces por cero, entropía de Shannon y características cepstral. La clasificación se lleva a cabo por una red neuronal. Su base de datos estaba compuesta por diez sujetos para entrenamiento y catorce sujetos para prueba. La edad de los sujetos abarcó desde tres meses hasta setenta y un meses. Sin embargo, las secuencias de audio originales no se utilizaron para el entrenamiento. En su lugar, los autores seleccionaron manualmente eventos de audio representativos, tanto de tos como de no tos, de cada sujeto del grupo de entrenamiento. Estos eventos de audio fueron concatenados más tarde. Antes del cálculo de características, también aplican un procedimiento de reducción de ruido basado en el filtro de paso alto y la sustracción

espectral de potencia. El procesamiento posterior también se llevó a cabo concatenando *frames* adyacentes de 20 ms de largo para recomponer los eventos de audio detectados. Los autores utilizan un proceso de suavizado que consiste en un filtro de promedio móvil y un umbral. El rendimiento a nivel de *frame* fue de alrededor del 85 % de sensibilidad y del 99 % de especificidad para una red neuronal con retardo de tiempo (*Time Delay Neural Network*, TDNN). La evaluación se basó en *leave-one-out cross-validation*.

**Nuevo algoritmo informático para el control de la tos basado en octoniones**

En marzo de 2018, Peter Klco *et al.* [187] presentaron un algoritmo novedoso para la detección de la tos en el que los sonidos de la tos se organizan como números de 8 dimensiones: octoniones. El objetivo es maximizar la sensibilidad del sistema. El conjunto de datos estaba compuesto por 5.200 sonidos de tos y 90 mil sonidos de no tos de dieciocho pacientes respiratorios. El sistema de grabación se basa en una grabadora de voz digital portátil y un micrófono de condensador en miniatura pegado al pecho del sujeto. Una membrana de espuma plástica cubrió el micrófono para suprimir los sonidos del micrófono y disminuir la intensidad de los sonidos externos. Los sujetos realizaron actividades de la vida diaria en un entorno clínico. Se utilizó una frecuencia de muestreo de 11.025 Hz y archivos WAV de 16 bits. La longitud del *frame* fue de 92,88 ms con una superposición del 50 %. Cada *frame* se caracterizó usando MFCC. Una red neuronal octoniónica (*Octonionic Neural Network*, ONN) realizó la clasificación. La entrada y los parámetros de la ONN se expresan como octoniones. Este modelo puede proporcionar un enfoque general para tratar con datos que varían en el tiempo. El modelo ONN se entrenó utilizando 10 mil épocas y ocho conjuntos de pruebas de entrenamiento diferentes seleccionados al azar.Se seleccionaron para el entrenamiento una tercera parte de los eventos de tos y una hora de varios eventos sin tos de cada sujeto. Los eventos de no tos se segmentaron previamente para tener una duración máxima de un segundo. Finalmente, 1.730 sonidos de tos y 6,460 sonidos sin tos estuvieron presentes en cada conjunto de entrenamiento. Sin embargo, el conjunto de pruebas incluyó la información de todos los pacientes. La sensibilidad y especificidad fueron superiores al 90 %.

**Uso de información mutua en la detección de eventos temporales supervisados: aplicación a la detección de tos**

En marzo de 2014, Thomas Drugman [188] propuso una técnica que consiste en un proceso iterativo para: encontrar el segmento más discriminante dentro de cada evento de audio y sincronizar características. Ambos pasos se realizan utilizando un criterio basado en información mutua. Este proceso mejora la selección de características y, por lo tanto, la clasificación posterior. La detección de tos de audio se presenta como un caso de aplicación de estudio. El conjunto de datos incluyó treinta y dos sujetos sanos. El cálculo de características se basa en tramas de 30 ms de largo con un desplazamiento de 12 ms. MFCC y otras características que describen el contenido espectral, como la energía de subbanda de frecuencia o el centroide espectral, se utilizaron junto con las características que miden el contenido de ruido (*e.g.*, relación armónica a ruido, planitud espectral o tasa de cruce por cero). El conjunto inicial de características tenía 222 componentes (incluyendo la primera y la segunda derivada). Después del proceso de selección de características, la dimensión se redujo a 50. La tasa de detección de eventos notificada fue del 92 %.

**Monitorización respiratoria inalámbrica y detección de tos mediante una red de sensores de parche portátil**

En junio de 2017, Elfaramawy *et al.* [189] desarrollaron un sistema respiratorio inalámbrico en tiempo real para medir la frecuencia respiratoria y la frecuencia de la tos. El sistema emplea dos sensores de baja potencia: una unidad de medición inercial de 9 ejes para medir la frecuencia respiratoria y un micrófono MEM para realizar la detección de la tos. Estos sensores envían la información a una estación base y un ordenador procesa los datos. Uno de los sensores se coloca en el tórax y el otro en el abdomen. Se utilizó la tasa de cruce por cero como una característica.

**Un sistema automatizado y no intrusivo para la detección de la tos**

En diciembre de 2017, Di Perna *et al.* [190] introdujeron un estudio que analiza cómo hacer frente al desequilibrio de clase en un monitor de conteo de tos para pacientes con EPOC. Siete pacientes con EPOC constituyeron

el grupo de estudio. Los audios se grabaron en la casa de cada paciente mediante un micrófono remoto instalado en la habitación donde duermen los pacientes. Dichos audios se segmentan previamente en fragmentos de audio de un segundo. MFCC fue empleado como conjunto de características. Los autores analizaron diferentes técnicas para compensar el desequilibrio de clase. En particular, compararon los enfoques de remuestreo (*e.g.*, la técnica de sobre muestreo de minorías sintéticas, SMOTE [191]) y los métodos de ensamblaje (*e.g.*, XGBoost). La clasificación fue realizada por un SVM en el caso de los enfoques de remuestreo. Todos los métodos se realizaron de manera similar, con un AUC promedio de 0,85.

#### Clasificación de eventos de tos por red neuronal profunda pre-entrenada

En noviembre de 2015, Jia-Ming Liu *et al.* [192] desarrollaron un modelo de red neuronal profunda que se construye a partir de un proceso de entrenamiento de dos pasos: pre-entrenamiento sin supervisión y ajuste fino. Finalmente, se utiliza un HMM para capturar información temporal. La red predice la probabilidad de observación asociada con cada estado HMM. Tres HMM están diseñados para la clase de tos y uno para la clase de no tos. La clasificación final se basa en el algoritmo de decodificación Viterbi. La representación de características se basa en MFCC. El sistema de grabación es una grabadora digital portátil con un micrófono pegado al cuello del paciente. Las señales fueron adquiridas en un entorno hospitalario natural. Las secuencias de audio se segmentaron previamente en clips largos de 10 segundos antes de su análisis. El desequilibrio de clase se compensó manualmente para reducir la proporción de no tos y tos a 2. En un experimento independiente del paciente (dieciséis pacientes para entrenamiento y seis pacientes para prueba), el modelo dio lugar a una sensibilidad del 83,6 % y un 90,9 % de especificidad.

## 5.2. Monitores comerciales para la tos

La tecnología comercial para el control de la tos se ha basado hasta ahora en dispositivos de diseño y fabricación *ad-hoc*. Adoptan señales de audio (micrófono y/o micrófono de contacto), solo o en combinación con otros sensores como acelerómetros, electrodos de electromiografía u oxímetros de pulso.

### 5.2.1. Monitores para la tos que solo usan señales de audio

Entre los monitores para la tos que solo usan señales de audio, hay tres ejemplos representativos [193].

#### Contador automático de tos de Hull (*Hull Automatic Cough Counter*, HACC)

Este monitor para la tos fue desarrollado en la Universidad de Hull en Reino Unido y utiliza un micrófono de campo libre para grabar sonidos de tos durante períodos de 24 horas. El sistema procesa previamente la entrada para separar los período de silencio de los sonidos fuertes. Luego, estos sonidos se caracterizan utilizando características espectrales (LPC y MFCC), y se aplica el análisis de componentes principales (*Principal Component Analysis*, PCA) para reducir la dimensión del espacio de características. Este análisis espectral se basa en señales muestreadas a 11.025 Hz de frecuencia de muestreo y calculadas en *frames* de 256 muestras, es decir, un análisis a corto plazo. Finalmente, se utiliza una red neuronal probabilística (*Probabilistic Neural Network*, PNN) para la clasificación [194].

Los autores entrenaron el sistema con un conjunto de datos que incluía a veintitrés sujetos fumadores con una tos crónica problemática durante una hora y lo probaron con diez sujetos desconocidos. El sistema alcanzó una sensibilidad promedio del 80 % (con sensibilidades entre el 55 % y el 100 %) y especificidad del 96 % [193], [195].

#### El monitor de la tos de Leicester (*The Leicester Cough Monitor*, LCM)

Este sistema fue desarrollado por los Hospitales Universitarios de Leicester en Reino Unido y se basa en un micrófono de campo libre con un dispositivo de grabación digital MP3, pudiendo grabar 24 horas. Aplica un enfoque de localización de palabras clave basado en MFCC y HMMs. Se crean tres HMM para modelar estadísticamente los patrones de tos (topología de izquierda a derecha con 10, 15 y 20 estados, respectivamente) y se generan 128

"rellenos" para representar los sonidos restantes (topología de izquierda a derecha con 3 estados) . Los HMM se alimentan con 18 MFCC calculados en *frames* de 8 ms con un desplazamiento de 6 ms. Utiliza una frecuencia de muestreo de 16 kHz y guarda el audio utilizando un formato de grabación comprimido de 64 kbps para su posterior análisis en una computadora personal.

Los autores informaron una sensibilidad del 85,7 % y una especificidad del 94,7 % en grabaciones de 6 a 24 h de 26 sujetos (18 pacientes con tos crónica y 8 controles sanos) [196].

#### El sistema VitaloJAK

Este dispositivo desarrollado conjuntamente por Vitalograph Ltd., Buckingham y el Hospital Universitario del Sur de Manchester ambos en Reino Unido. Utiliza un micrófono de contacto colocado en la pared torácica y un dispositivo de grabación digital hecho a medida para detectar sonidos de tos. Se utiliza un algoritmo basado en un umbral de frecuencia mediana para comprimir las grabaciones de sonido de la tos de 24 h. Ofreció una sensibilidad del 98 % y una especificidad del 99 % en un conjunto de datos compuesto por 10 pacientes [193], [195].

### 5.2.2. Monitores para la tos que usan múltiples señales

Los siguientes sistemas comerciales emplean múltiples señales para detectar la tos [193].

#### El sistema Lifeshirt

Se lanzó en 2005 y fue desarrollado por VivoMetrics, Inc., Ventura, California en Estados Unidos pero ya no está disponible porque la empresa se liquidó en 2009. Es un sistema de monitorización fisiológico ambulatorio basado en varios sensores integrados en una camisa que lleva el usuario: electrocardiograma, pletismografía de inducción, acelerómetro de 3 ejes y un micrófono de contacto colocado en la garganta. Solo se publicó un pequeño estudio de validación que utilizó información de ocho pacientes con antecedentes documentados de EPOC con diez o más años de tabaquismo en condiciones de laboratorio. La sensibilidad informada es del 78,2 %, pero la tasa de falsos positivos no está clara [195], [197].

#### El Pulmo Track-CC

Fue lanzado en 2010 por KarmelSonix Ltd., Haifa en Israel y utiliza una combinación de sonidos grabados en el cuello y un sensor de movimiento (cinturón piezoeléctrico) colocado en la pared torácica. El dispositivo ha sido probado utilizando 12 voluntarios sanos (25 minutos de registro de cada uno) simulando la tos en diferentes situaciones (*e.g.*, caminar, sentarse o subir escaleras). Proporcionó valores de sensibilidad y especificidad en torno al 95 % [198].

#### BodyScope

Se desarrollaron otras tecnologías con un objetivo más amplio que la detección de la tos. Yatani y Truong crearon un sensor acústico portátil, llamado BodyScope, para grabar los sonidos producidos en el área de la garganta del usuario y clasificarlos en actividades del usuario, como comer, beber, hablar, reír y toser. Consiste en un auricular Bluetooth, un micrófono (incorporado en el auricular) y el estuche de un estetoscopio. La caracterización se basa en características del dominio de la frecuencia, como la velocidad de cruce cero, la potencia de subbanda, la reducción espectral o el flujo espectral y MFCC. Se analizaron tres clasificadores: un SVM, K-NN y HMM. Se reportó un 62 % de sensibilidad para eventos de tos [199].

## 5.3. Detección y análisis de enfermedades

En esta sección hablaremos sobre los principales estudios realizados para la detección de enfermedades pulmonares, cabe destacar que la mayoría de estudios se inclinan por el análisis de radiografías en vez del análisis de los sonidos pulmonares para este fin.

**El análisis del sonido de la tos puede diagnosticar rápidamente la neumonía infantil**

Este estudio fue lanzado en 2013 y se realizo conjuntamente en la Universidad de Queensland, Brisbane en Australia y en la Universidad Gadjah Mada, Yogyakarta en Indonesia. Se grabaron sonidos de tos de 91 pacientes de los cuales se sospechaba que tenían una enfermedad respiratoria aguda como neumonía, bronquiolitis y asma. Los micrófonos utilizados para la adquisición de los datos se colocaban junto a la cama del paciente. Se extrajeron características como la no gaussianidad y cepstrum de frecuencia de Mel (*Mel-Frequency Cepstrum*, MFC) de los sonidos de la tos y las usaron para entrenar a un clasificador de Regresión logística. Utilizaron el diagnóstico clínico proporcionado por el clínico pediátrico de respiración como el estándar de oro para entrenar y validar el clasificador. Con este método se informó de que podían separar la neumonía de otras enfermedades con una sensibilidad y especificidad del 94 % y 75 % respectivamente. La inclusión de otras medidas simples, como la presencia de fiebre, aumentó aún más el rendimiento [200].

**CheXNet: Detección de neumonía a nivel de radiólogo en radiografías de tórax con aprendizaje profundo**

Este estudio se realizó en la Universidad de Stanford de California en Estados Unidos, donde se desarrolló un algoritmo con la capacidad de detectar la neumonía a partir de radiografías de tórax a un nivel superior al de cuatro radiólogos en ejercicio con los que se comparó. Este algoritmo denominado CheXNet, es una red neuronal convolucional de 121 capas, entrenada con el *dataset* ChestX-ray14. Este *dataset* actualmente es el mayor conjunto de datos de rayos X de tórax disponible públicamente, y contiene más de 100 mil imágenes de rayos X de vista frontal con 14 enfermedades. CheXNet obtuvo 43,5 % en la métrica de F1, mejorando así el rendimiento promedio de los cuatro radiólogos (38,7 %). Además se extendió CheXNet para detectar las 14 enfermedades etiquetadas en ChestX-ray14 y lograr los resultados más avanzados en las 14 enfermedades [201].

**Estadificación de la enfermedad y pronóstico en fumadores que utilizan el aprendizaje profundo en la tomografía computarizada de tórax**

Este estudio fue lanzado en 2017 y se realizo conjuntamente en Sierra Research, Alicante en España y en Brigham and Women's Hospital (BWH), Boston (Massachusetts) en Estados Unidos. Se entrenó a una CNN utilizando tomografías computadas de 7983 participantes de COPDGene y se evaluó utilizando mil participantes de COPDGene no superpuestos y 1672 participantes de evaluaciones de EPOC longitudinalmente para identificar puntos finales sustitutos predictivos (*Evaluations of COPD Longitudinally to Identify Predictive Surrogate End-points*, ECLIPSE). Se utilizó la regresión logística (estadística C y la prueba de Hosmer-Lemeshow) para evaluar el diagnóstico de EPOC y la predicción de ERA. Se utilizó la regresión de Cox (índice de C y la prueba de Agnostino de Greenwood-Nam-D) para evaluar la mortalidad. COPDGene es un estudio longitudinal observacional financiado por el NHLBI de 10300 fumadores cuyo objetivo es definir las asociaciones epidemiológicas y los factores de riesgo genéticos para el desarrollo de la EPOC. Los participantes con enfermedades pulmonares activas distintas de la EPOC y el asma fueron excluidos de la participación y todos los participantes se sometieron a una prueba de referencia, incluida una entrevista extensa, una tomografía computarizada (*Computed Tomography*, CT) volumétrica de alta resolución del tórax y una espirometría. Los fumadores con y sin EPOC se inscribieron y ahora regresan para su visita de seguimiento a intervalos de 5 años. ECLIPSE fue un estudio multicéntrico longitudinal de 3 años con 2164 sujetos de iniciativa mundial para la EPOC en estadio 2–4 y 582 sujetos control finalizado en 2011. Los participantes fueron excluidos si tenían enfermedades respiratorias conocidas distintas de la EPOC o una deficiencia grave de antitripsina alfa-1. Los procedimientos del estudio se realizaron al inicio del estudio, 3 meses, 6 meses, y luego cada 6 meses por un total de 3 años. La espirometría se realizó al inicio del estudio, y las CT se realizaron al inicio, al año y a los 3 años.

Debido a las restricciones causadas por las capacidades de procesamiento de las unidades de procesamiento gráfico existentes, la CNN no puede utilizar las imágenes de CT de alta resolución completas de un individuo. Por lo tanto, se usó un detector de objetos para extraer automáticamente cuatro cortes canónicos de CT en puntos de referencia anatómicos preseleccionados. Siendo estos un corte axial en el nivel de la válvula mitral, un corte coronal tomado en el nivel de la aorta ascendente y dos cortes sagitales al nivel de la hila derecha e izquierda. Este paso de reducción de la dimensionalidad "normaliza" los datos de CT utilizando información anatómica. Estas imágenes se unieron en un solo montaje e incluían un corte axial centrado en el corazón a nivel de la

válvula mitral, dos cortes sagitales reformateados centrados en el hilio izquierdo y derecho, y un corte coronario reformateado centrado en la aorta ascendente. La CNN consistía en tres capas convolucionales alternadas con ReLU y seguidas de operaciones de *max polling*, cada una de las cuales reduce cuatro veces el tamaño de la imagen en cada dirección. Las dos capas finales de la CNN estaban completamente conectadas, la primera de 1024 neuronas y la segunda capa completamente conectada variaba (dos, número de clases y una, respectivamente) según la tarea (clasificación binaria, clasificación categórica o regresión).

En el COPDGene, el estadístico C para la detección de la EPOC fue de 0,856. Un total de 51,1 % de los participantes en el COPDGene se estatificaron con exactitud y el 74,95 % se encontraban en una etapa. En ECLIPSE, el 29,4 % se estadificó con precisión y el 74,6 % en una etapa. Los eventos de dificultad respiratoria aguda (*Acute Respiratory Distress*, ARD) ocurren en fumadores con y sin EPOC, y son aumentos temporales de los síntomas respiratorios que incluyen tos, producción de esputo y disnea que justifican un cambio en la terapia. En COPDGene y ECLIPSE, las estadísticas de C para los eventos ARD fueron 0,64 y 0,55, respectivamente, y los valores de Hosmer-Lemeshow P fueron 0,502 y 0,380, respectivamente, lo que sugiere que no hay evidencia de calibración deficiente. En COPDGene y ECLIPSE, la CNN predijo la mortalidad con una discriminación justa (índices C, 0,72 y 0,60, respectivamente) y sin evidencia de calibración deficiente (valores de Greenwood-Nam-D'Agnostino P, 0,307 y 0,331, respectivamente) [202].

**Aprendizaje profundo con segmentación pulmonar y técnicas de exclusión de sombra ósea para el análisis de rayos X de tórax del cáncer de pulmón**

Este estudio fue lanzado en 2018 y se realizo conjuntamente en Universidad Técnica Nacional de Ucrania "Instituto Politécnico Igor Sikorsky de Kiev", Kiev en Ucrania y en la Universidad de Huizhou, ciudad de Huizhou en China. Aquí se demuestra la eficacia de la segmentación pulmonar y las técnicas de exclusión de la sombra ósea para el análisis de los *Chest X-Ray* (CXR) 2D mediante un enfoque de aprendizaje profundo para ayudar a los radiólogos a identificar lesiones sospechosas y nódulos en pacientes con cáncer de pulmón. El entrenamiento y la validación se realizaron en el conjunto de datos JSRT original [203] (conjunto de datos 1), el conjunto de datos BSE-JSRT [204], es decir, el mismo conjunto de datos JSRT, pero sin clavícula y sombras en las costillas (conjunto de datos 2), el conjunto de datos original JSRT después de la segmentación (conjunto de datos 3), y conjunto de datos BSE-JSRT después de la segmentación (conjunto de datos 4). Los resultados demuestran la alta eficiencia y utilidad de las técnicas de preprocesamiento incluso consideradas en la configuración simplificada. El conjunto de datos preprocesados sin huesos (conjunto de datos 2) muestra resultados de pérdida y precisión mucho mejores en comparación con otros conjuntos de datos preprocesados después de la segmentación pulmonar (conjuntos de datos 3 y 4) [205].

# *Capítulo* 6
# METODOLOGÍA

El objeto de este trabajo es diseñar un sistema de detección de tos robusto que permita distinguir entre audios que contengan tos y audios que no contengan tos. Por otra parte también se busca una primera aproximación al diagnóstico de enfermedades basándonos en las toses detectadas. Para ello, hemos trabajado con señales de audio de diversos pacientes que padecían diferentes enfermedades respiratorias, primero realizando un preprocesado y posteriormente alimentando a una red que aplique algoritmos de *Deep Learning*, en concreto una CNN. En este capítulo nos centraremos en detallar el procedimiento que se ha empleado para el desarrollo del sistema, concretamente:

- Descripción de las señales obtenidas en las bases de datos utilizadas y preprocesado necesario para poder utilizar las señales de audio como datos efectivos de entrada para una red neuronal convolucional.
- Diseño de la **red neuronal convolucional** utilizada en cada caso de estudio y explicación del proceso de entrenamiento de la misma.
- Diagnóstico de los pacientes a partir de las predicciones que ofrece nuestra red.

## 6.1. Bases de datos

En esta sección procedemos a describir las tres bases de datos que se han utilizado para nuestro estudio.

### 6.1.1. Palencia

La base de datos "Palencia" está compuesta por grabaciones de audio de algo más de 24 horas de veinte pacientes del Complejo Asistencial Universitario de Palencia con diferentes afecciones respiratorias (ver Tabla 6.1). Utilizamos un teléfono inteligente *Android Sony Xperia Z2* para recopilar los datos, almacenando los archivos en formato WAV de 16 bits a 44,1 kHz [206]. Se instruyó a los pacientes para que pusieran el dispositivo en sus bolsillos o carteras como lo harían normalmente, para obtener muestras de entornos reales y ruidosos que son hostiles a la detección de la tos.

| ID | Enfermedad | Edad | Sexo | # toses | Tipo de tos |
|---|---|---|---|---|---|
| 0 | Enfermedad Respiratoria Aguda | 36 | M | 265 | Aguda |
| 1 | Bronquiectasias & Asma | 23 | F | 0 | Crónica que no es EPOC |
| 2 | Sarcoidosis | 37 | M | 163 | Crónica que no es EPOC |
| 3 | Enfermedad Respiratoria Aguda | 53 | M | 352 | Aguda |
| 4 | Enfermedad Respiratoria Aguda | 18 | F | 602 | Aguda |
| 5 | EPOC | 79 | M | 767 | EPOC |
| 6 | Cáncer de pulmón | 84 | M | 921 | Cáncer de pulmón |

| ID | Enfermedad | Edad | Sexo | # toses | Tipo de tos |
|---|---|---|---|---|---|
| 7 | Neumonía bilateral | 44 | M | 2401 | Aguda |
| 8 | Cáncer de pulmón | 62 | M | 809 | Cáncer de pulmón |
| 9 | Apnea Obstructiva del Sueño | 70 | F | 1268 | |
| 10 | Bronquiectasias | 63 | F | 508 | Crónica que no es EPOC |
| 11 | EPOC & Apnea Obstructiva del Sueño | 48 | F | 586 | |
| 12 | EPOC | 69 | F | 901 | EPOC |
| 13 | Neumonía unilobular | 31 | F | 1579 | Aguda |
| 14 | Neumonía atípica | 83 | M | 504 | Aguda |
| 15 | Neumonía & EPOC | 87 | M | 1166 | |
| 16 | Tromboembolismo pulmonar | 70 | F | 917 | |
| 17 | EPOC | 50 | F | 2033 | EPOC |
| 18 | Asma | 70 | F | 697 | Crónica que no es EPOC |
| 19 | EPOC & Hamartoma pulmonar | 60 | M | 615 | |
| 20 | EPOC & Fibrosis pulmonar | 67 | M | 1379 | |

Tabla 6.1: Detalles de la base de datos “Palencia”.

Usamos la herramienta de software Praat [13] y creamos un fichero de subtítulos para etiquetar manualmente las toses, que usamos para establecer la verdad fundamental o *ground truth*. Si no era posible determinar con certeza si un segmento de audio debía marcarse como tos o no, lo eliminamos de la base de datos [206]. Utilizamos este archivo de subtítulos y Praat para crear clips de audio separados para cada evento de sonido como vemos en las Figuras 6.1, 6.2 y 6.3.

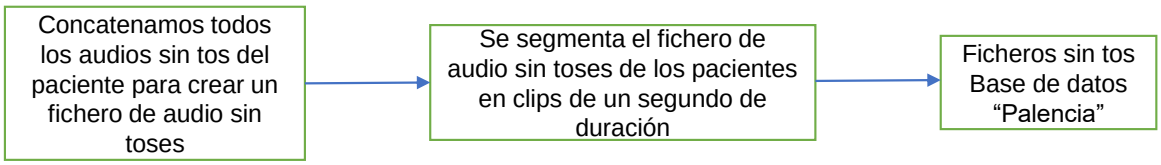



Figura 6.1: Metodologia para la segmentación de los clips de audio sin tos (detección).

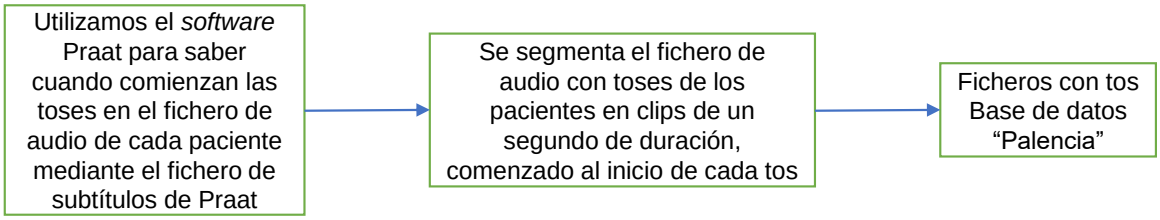



Figura 6.2: Metodologia para la segmentación de los clips de audio con tos (detección).

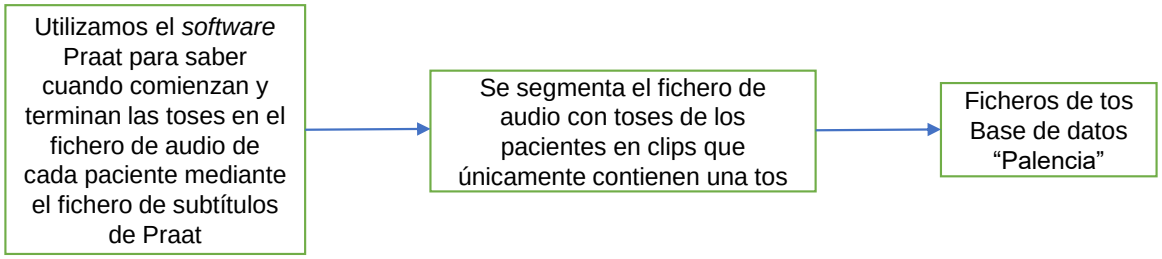



Figura 6.3: Metodologia para la segmentación de los clips de audio de tos (clasificación).

### 6.1.2. Glasgow

La base de datos "Glasgow" ha sido confeccionada recolectando prospectivamente grabaciones de audio ambulatorias de 24 horas de pacientes atendidos en clínicas de medicina respiratoria en el Hospital Universitario Queen Elizabeth (*Queen Elizabeth University Hospital*, QEUH) y en el Hospital New Victoria (*New Victoria Hospital*, NVH) de Glasgow (Escocia) en Reino Unido. Los participantes recibieron un micrófono de solapa de campo libre (*Philips LFH9173*) y una grabadora de MP3 (*Sony ICD-PX333*) durante las 24 horas. La calidad de los ficheros MP3 fue de 192 kbps con una frecuencia de muestreo de 44,1 kHz. El monitor de tos de Leicester (*Leicester Cough Monitor*, LCM) [207] se utilizó para extraer los sonidos de tos de las grabaciones de 24 horas en el *King's College* de Londres y debido a estas circunstancias sólo pudimos disponer de los ficheros extraidos que contenían sonidos de tos. El LCM es un sistema automatizado de detección de tos que identifica automáticamente los sonidos de tos de las grabaciones de audio y ensambla clips de sonido de 1 segundo para todas las partes de las grabaciones de audio que se identifican como un sonido de tos. Los clips de sonido se muestrearon a 16 kHz y se almacenaron como ficheros WAV. El estudio se llevó a cabo de acuerdo con la Declaración de Helsinki y fue aprobado por el Comité de Ética de Investigación de QEUH.

El grupo de estudio incluyó individuos con pulmones "normales", individuos con alto riesgo de cáncer de pulmón (EPOC y otras enfermedades pulmonares crónicas) e individuos con cáncer de pulmón. Los pacientes, de 50 años o más, fueron reclutados y asignados a una de los siguientes 4 grupos:

- **Fumadores normales**: Pacientes que presentaban tos pero que parecían tener pulmones "sanos", es decir, que después de la evaluación clínica se descartó la presencia de EPOC, otras enfermedades pulmonares crónicas o cáncer de pulmón.
- **EPOC**: Pacientes con un diagnóstico confirmado de EPOC según los criterios establecidos.
- **Otras enfermedades pulmonares crónicas (sin EPOC)**: Pacientes con un diagnóstico confirmado de enfermedad pulmonar crónica sin EPOC (*e.g.*, fibrosis pulmonar, asma, etc).
- **Cáncer de pulmón**: Pacientes con diagnóstico confirmado de cáncer de pulmón que incluye enfermedad en los pulmones y tos activa.

| ID | Tipo de enfermedad | Número de toses |
|---|---|---|
| 1 | Tos crónica que no es EPOC | 205 |
| 2 | EPOC | 1191 |
| 3 | EPOC | 535 |
| 4 | Tos crónica que no es EPOC | 794 |
| 5 | Cáncer de pulmón | 784 |
| 6 | No grabado | 0 |
| 7 | EPOC | 493 |
| 8 | Tos crónica que no es EPOC | 443 |

| ID | Tipo de enfermedad | Número de toses |
|---|---|---|
| 9 | EPOC | 9 |
| 10 | EPOC | 509 |
| 11 | Cáncer de pulmón | 334 |
| 12 | Tos crónica que no es EPOC | 341 |
| 13 | Fumador normal | 81 |
| 14 | Fumador normal | 48 |
| 15 | Cáncer de pulmón | 44 |
| 16 | Fumador normal (retirado) | 0 |
| 17 | Fumador normal | 37 |

Tabla 6.2: Detalles de la base de datos “Glasgow”.

Este estudio exploratorio inicial incluyó datos de 3 fumadores normales, 5 pacientes con EPOC, 4 pacientes con otras enfermedades pulmonares crónicas (sin EPOC o cáncer de pulmón) y 3 pacientes con cáncer de pulmón como podemos ver en la Tabla 6.2.

### 6.1.3. Edimburgo

La base de datos “Edimburgo” está compuesta por grabaciones ambulatorias de trece pacientes adultos adquiridos en la *Outpatient Chest Clinic, Royal Infirmary of Edinburgh* de Edimburgo (Escocia) en Reino Unido, que presentan la tos como un síntoma de su afección subyacente (ver la Tabla 6.3). El siguiente protocolo de adquisición fue desarrollado para simular situaciones ruidosas de la vida real.

| ID | Enfermedad | Edad | Sexo | Número de toses | Tipo de tos |
|---|---|---|---|---|---|
| 1 | Bronquiectasias | 70 | F | 74 | Crónica que no es EPOC |
| 2 | Asma | 45 | M | 96 | Crónica que no es EPOC |
| 3 | EPOC | 69 | F | 103 | EPOC |
| 4 | EPOC | 48 | M | 86 | EPOC |
| 5 | Bronquiectasias | 48 | F | 100 | Crónica que no es EPOC |
| 6 | Asma | 72 | F | 97 | Crónica que no es EPOC |
| 7 | EPOC | 66 | F | 32 | EPOC |
| 8 | Bronquiectasias | 66 | F | 50 | Crónica que no es EPOC |
| 9 | EPOC | 61 | F | 134 | EPOC |
| 10 | Bronchiectasis | 68 | F | 110 | Crónica que no es EPOC |
| 11 | EPOC | 65 | F | 70 | EPOC |
| 12 | Asma & Enfisema | 72 | F | 244 | |
| 13 | EPOC & Bronquiectasias | 67 | M | 97 | |

Tabla 6.3: Detalles de la base de datos “Edimburgo”.

La primera parte emula un ambiente de bajo ruido. En esta situación, el paciente está sentado y se le pide que hable o lea en voz alta. De vez en cuando, le pedimos al paciente que produzca otros eventos que no sean tos, como el aclaramiento de la garganta, tragar (bebiendo un vaso de agua), soplar la nariz, estornudar, respirar sin aliento o reír (leyendo un chiste o un cómic de humor).

La segunda parte del protocolo emula un entorno ruidoso con una fuente externa de contaminación (el paciente no produce los ruidos de fondo ruidosos). Para hacerlo, repetimos el experimento en la primera parte con un televisor o un reproductor de radio encendido, y también permitiendo que se grabe también el ruido del pasillo del hospital (*e.g.*, ruido de balbuceo, ruido de mecanografía o un carro en movimiento). Esta segunda parte es un ambiente moderadamente ruidoso.

Finalmente, la tercera parte del protocolo fue diseñada para representar ambientes ruidosos donde los propios pacientes también se convierten en una fuente de contaminación debido a sus movimientos y otras actividades. En este caso, el paciente podía moverse libremente por la habitación mientras le pedíamos que realizara algunas actividades, como abrir o cerrar la ventana, abrir o cerrar un cajón, mover una silla, lavarse las manos, acostarse en la cama y levantarse de inmediato, escribiendo, poniéndose el abrigo y quitándoselo inmediatamente después o, recogiendo un objeto del suelo, entre otros. Al igual que en la segunda parte, un televisor o radio estaba encendido y la puerta se dejó abierta para grabar también ruidos en el pasillo. Igualmente, mientras el paciente estaba realizando estas actividades, le pedimos que produjera otros eventos de primer plano sin tos como en la primera y segunda parte. La tercera parte representa así un escenario de alto ruido [4].

Cada parte del protocolo duró veinte minutos, por lo que se adquirieron un total de trece horas de grabación. La grabadora digital se colocó en una mesa en el centro de la habitación. Todos los eventos de tos registrados como resultado de este protocolo fueron espontáneos. Las señales se registraron en formato WAV utilizando un teléfono inteligente *Android Samsung S6 Edge*, a una frecuencia de muestreo de 44,1 kHz, con 16 bits por muestra. El protocolo de adquisición es similar al utilizado en otros estudios [196], [208] aunque es más diverso, en términos de tipos de sonidos ruidosos, y presenta un mayor grado de contaminación que los utilizados en [185], [186].

## 6.2. Preprocesado

Las bases de datos anteriormente descritas nos ofrecen clips de audio.

- **Con tos**, que comienzan al inicio de la tos y finalizan al finalizar la misma, estos clips habitualmente tienen una duración inferior a un segundo.
- **Sin tos**, que comienzan al finalizar una tos y terminan al comenzar la siguiente tos, estos clips habitualmente tienen una duración superior a un segundo.

Estos datos que aparentan ser triviales van a jugar un papel determinante en el preprocesado.

### 6.2.1. Adecuación de los clips de audio para la detección de tos

Para nuestro primer estudio en el que sólo utilizaremos la base de datos "Palencia" (descrita en la sección 6.1.1), debemos tener en cuenta que la diferencia de duración de los clips de audio con tos y sin tos, debido a que la diferencia de duración entre ambos, hará que posteriormente nuestra CNN sea capaz de diferenciar esta característica y no el contenido del clip de audio. La solución inmediata es recortar todos los clips de audio a un segundo, ya que de esta manera, en los clips de audio con tos pocas veces perdíamos información de la tos, garantizando siempre que la parte explosiva de la tos esta incluida, y los clips de audio sin tos adquirían una duración similar a los clips de audio con tos. Posteriormente realizamos un análisis estadístico para comprobar si habíamos eliminado dicho sesgo.

Claramente observamos en la Figura 6.4a que la duración de los clips de audio sin tos tiende a ser cercana a un segundo o a valores muy bajos, mientras que los clips de audio con tos tienen duraciones intermedias. También podemos comprobar en la Figura 6.4b como estas características se ven reflejadas en la energía de los espectrogramas obtenidos a partir de los clips de audio.

Con intención de subsanar el sesgo observable en la Figura 6.4, vamos a buscar que todos los clips de audio tengan una duración de un segundo. Esta adecuación de los datos a estudiar sólo se puede realizar con la base de datos "Palencia", ya que en está base de datos tenemos los ficheros de subtítulos de Praat, por este motivo no se han utilizado las bases de datos "Glasgow" y "Edimburgo".

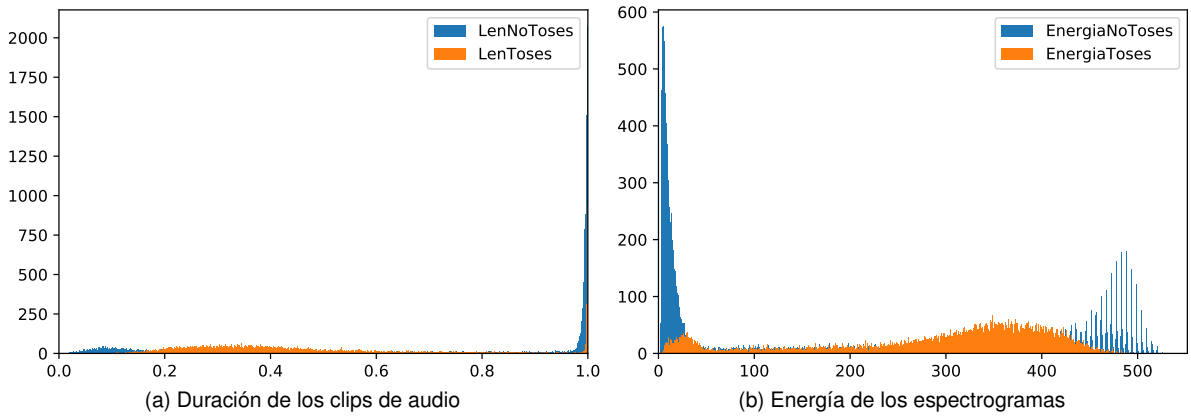



Figura 6.4: Histogramas realizados para observar el sesgo de los datos estudiados.

Para realizar esta adecuación de los datos utilizamos los ficheros de subtítulos de Praat de la siguiente forma para obtener los nuevos clips de audio.

- Los nuevos clips de audio con tos comenzarán en el momento en el que el fichero de subtítulos de Praat indique que comienza una tos y finalizará al termino de un segundo. Estos clips de audio pueden contener toses adyacentes, lo cual nos es de gran interés para realizar *data augmentation*.
- Los nuevos clips de audio sin tos se obtendrán a partir de un fichero de audio que obtenemos con Praat, en el que concatenamos todos los clips de audio sin tos. Este fichero de audio posteriormente lo seccionamos en clips de audio de un segundo, que serán los nuevos clips de audio sin tos.

Ahora sabemos que hemos eliminado el sesgo producido por las duraciones de los clips de audio, ya que todos los clips de audio duran un segundo, pero debemos volver a realizar un análisis estadístico para comprobar si hemos eliminado el sesgo observable en la energía de los espectrogramas.

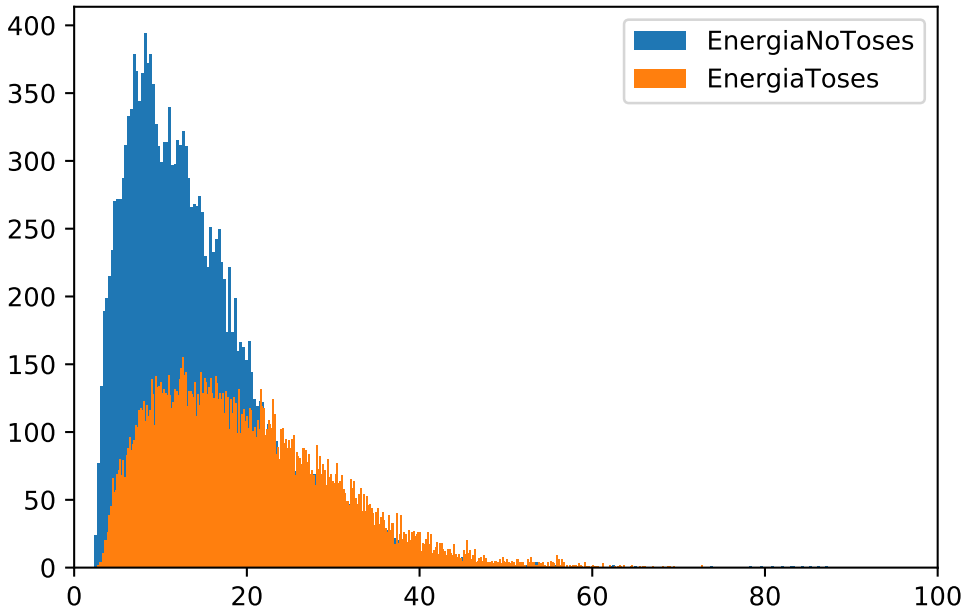



Figura 6.5: Energía de los espectrogramas en el que observamos la eliminación del sesgo.

Podemos comprobar en la Figura 6.5 que hemos eliminado dicho sesgo. Aunque el histograma de energías de los clips de audio con tos es diferenciable del obtenido de los clips de audio sin tos, esto lo entendemos como una característica de los audios sin tos frente a los audios con tos.

Una vez obtenidos estos nuevos clips de audio, realizamos un *downsampling* o diezmado de la señal de audio, bajando su frecuencia de muestreo de 44,1 kHz a una quinta parte de su valor, es decir, 8,82 kHz. Esta operación es realizada debido a que de esta manera, se reducirá la carga computacional necesaria para entrenar nuestra red neuronal.

### 6.2.2. Adecuación de los clips de audio para el diagnóstico de enfermedades

En el caso de nuestro segundo estudio, el procedimiento es mas sencillo, ya que todos los clips de audio que vamos a utilizar son clips de audio con tos. Esto hace que no tengamos que tener clips de audio de una determinada duración, ya que tendrán la duración que presente la tos a analizar. Para entender este problema, primero debemos hacer una breve, pero no menos importante introducción, acerca del tipo de toses que existen y además un poco de información de cada una de las enfermedades que vamos a tratar.

Se pueden clasificar tres tipos de tos según su duración, siendo estos, tos aguda, tos subaguda y tos crónica. La tos aguda es aquella con una duración inferior a tres semanas. Es benigna y se autolimita en el tiempo, pudiéndose deber a causas infecciosas, como los virus que afectan a a las vías respiratorias altas y los que producen bronquitis. También puede estar causada por bacterias, como *Mycoplasma pneumoniae*, *Chlamydia pneumoniae* y *Bordetella pertussis* [2]. Por otro lado la sinusitis es causa de tos aguda en un 10 % de los casos y la neumonía en un 8 %. La tos subaguda es aquella que se dilata de tres a ocho semanas, la tos ferina es una causa emergente altamente contagiosa, que provoca una tos espasmódica con estridor respiratorio que se puede dilatar de cuatro a seis semanas e incluso puede cronificarse. La tos crónica es un síntoma que persiste mas de ocho semanas. La tos aguda o subaguda, en algunas ocasiones, puede cronificarse. Clásicamente, se atribuía la etiología de la tos crónica a un goteo o Drenaje Nasal Posterior (DNP), a Reflujo GastroEsofágico (RGE) o a inflamación crónica de la vía aérea, como ocurre en el asma o en la bronquitis eosinofílica. También son causa de tos crónica algunos fármacos, como los Inhibidores de la Enzima Convertidora de la Angiotensina (IECA) o los $\beta$-bloqueantes, así como causas ambientales u ocupacionales. Además, la mayoría de pacientes con RGE, asma o DNP no padecen tos. Estas situaciones han obligado a reflexionar y a introducir un nuevo concepto: el de *síndrome de hipersensibilidad laríngea* [2].

A la hora de realizar nuestra clasificación también debemos conocer las enfermedades que vamos a estudiar. Por importancia en nuestro estudio comenzaremos hablando sobre el cáncer de pulmón. El cáncer de pulmón es el cáncer de la tráquea, bronquios (vías respiratorias) y alvéolos pulmonares. El cáncer de pulmón era una enfermedad rara a inicios del siglo XX, pero el aumento de la exposición al humo del tabaco y otros factores han contribuido a una expansión pandémica durante los siglos XX y XXI. Se necesitan nuevas técnicas que ayuden a los médicos a determinar la fase de desarrollo del cáncer de pulmón, de modo que puedan reducirse los ingresos hospitalarios y se acelere la toma de decisiones relacionadas con el tratamiento. Se necesita más investigación para refinar las técnicas de radioterapia e identificar los marcadores del cáncer de pulmón para, de este modo, garantizar un diagnóstico precoz. Se necesita una base de datos de casos de cáncer de pulmón, bien organizada y fiable, para permitir la identificación de las tendencias y la investigación a nivel de la salud pública, y poder observar las diferencias de supervivencia en los diferentes países. El cáncer de pulmón es en la actualidad la principal causa de muerte por cáncer en todo el mundo en hombres y mujeres. Sólo una de cada ocho personas con cáncer de pulmón sobrevive cinco años después del diagnóstico. El cáncer de pulmón es en la actualidad la principal causa de muerte por cáncer en todo el mundo, con 1,38 millones de fallecidos en 2008 [3].

La siguiente enfermedad que también vamos a intentar identificar a través de los clips de audio de tos es la EPOC. La EPOC es un proceso a largo plazo que causa inflamación pulmonar, daños en el tejido pulmonar y obstrucción de las vías respiratorias, dificultando la respiración. La enfermedad se presenta de varias formas, aunque se sabe poco de las causas de esta variación y la mejor manera de controlar las diferentes expresiones de esta enfermedad. Aproximadamente el 40-50 % de las personas que fuman durante toda su vida desarrollarán EPOC, frente al 10 % de los no fumadores. Las 300.000 muertes en Europa por EPOC cada año son el equivalente de 3 bombas de Hiroshima.

Con objeto de profundizar más en la neumonía y las Infecciones Respiratorias Agudas (IRA), vamos a hablar de

las infecciones respiratorias de vías bajas. Las infecciones respiratorias de vías bajas incluyen neumonía (infección del pulmón o los alvéolos), así como infecciones que afectan a las vías respiratorias tales como bronquitis aguda y bronquiolitis, gripe y tos ferina. Es una de las principales causas de enfermedad y muerte en niños y adultos en todo el mundo. La importancia de las infecciones respiratorias de vías bajas puede subestimarse porque no están bien definidas. En la UE, se estima que ocurren unos 3.370.000 casos de neumonía cada año. En la población infantil, las infecciones respiratorias agudas suponen aproximadamente el 50 % de las visitas al médico y de las hospitalizaciones [3].

Por último debemos hablar de tres enfermedades de tos crónica que vamos a tratar en nuestro estudio, estas son bronquiectasias, asma y sarcoidosis. Las bronquiectasias describen la dilatación ("ectasia") de algunas vías respiratorias. Esto ocurre de forma intermitente debido a los daños causados por la infección. Se evita la eliminación efectiva de la mucosidad y aumentan las posibilidades de posteriores infecciones e inflamación. Las vías respiratorias más pequeñas se hacen más gruesas y se obstruyen debido a la inflamación, produciendo dificultades respiratorias. La bronquiectasia es una de las enfermedades respiratorias más olvidadas. No existe una clasificación general consensuada de la enfermedad, hay pocos servicios especializados y poca información sobre los efectos a largo plazo de esta enfermedad. El 50 % de las personas con bronquiectasias tienen un problema subyacente relacionado con la enfermedad. Las bronquiectasias suelen confirmarse con un TAC. El reconocimiento y tratamiento precoz es clave para un mejor resultado a largo plazo.

El asma es una enfermedad crónica común que puede afectar a personas de todas las edades. Produce inflamación de las vías respiratorias. El término asma del adulto hace referencia a: Asma infantil que continúa en la edad adulta; reaparición del asma tras haberla padecido en la infancia con posterior desaparición; o Asma que se ha desarrollado sólo en la edad adulta. Con frecuencia, el asma del adulto está relacionada con alergias y viene acompañada de otros estados alérgicos, como la rinitis/fiebre del heno. En Europa, casi 10 millones de niños y adultos menores de 45 años padecen asma [3].

La sarcoidosis es una enfermedad granulomatosa sistémica de etiología desconocida que afecta fundamentalmente al pulmón y ganglios linfáticos del tórax, con menor frecuencia a ojos y piel, y en ocasiones a otros órganos. Conceptualmente además de su etiología desconocida, se incluyen cinco datos relevantes, que permiten sugerir una definición operativa: es un proceso multisistémico, con predominio de la afectación pulmonar, de carácter granulomatoso, en ausencia de vasculitis y mediado inmunológicamente [209].

En nuestro caso, aún utilizando las tres bases de datos conjuntamente, no es una base de datos suficientemente extensa como para analizar cada enfermedad de forma individual, por tanto reduciremos nuestro estudio a pretender distinguir entre cuatro tipos de tos diferentes, siendo estos:

- Tos aguda, en esta clase albergamos pacientes con IRA, neumonía bilateral , neumonía unilobular, neumonía atípica y neumonía.
- Tos producida por pacientes con cáncer de pulmón.
- Tos producida por pacientes con EPOC.
- Tos crónica que no es EPOC, en esta clase albergamos pacientes con bronquiectasias, asma y sarcoidosis.

Por otra parte para utilizar las tres bases de datos conjuntamente debemos tener la misma frecuencia de muestreo en todas las bases de datos, por este motivo vamos a escoger la frecuencia de muestreo de 16 kHz que es la frecuencia de muestreo de la base de datos "Glasgow". Debido a este motivo realizamos un *downsampling* o diezmado de la señal de audio de las bases de datos de Palencia y Edimburgo, bajando su frecuencia de muestreo de 44,1 kHz a 16 kHz.

Además recortamos los clips de audio que tengan una duración superior a un segundo y realizamos un *padding* de ceros en los clips de audio que tengan una duración inferior a un segundo.

### 6.2.3. Espectrograma

La *transformada de Fourier dependiente del tiempo*, denominada también Transformada de Fourier de Tiempo Corto (*Short-Time Fourier Transform*, STFT) de una señal $x[n]$ se define como

$$X[n,\lambda] = \sum_{m=-\infty}^{\infty} x[n+m]w[m]e^{-j\lambda m} \tag{6.1}$$

siendo $w[n]$ una secuencia de ventana. En la representación de Fourier dependiente del tiempo, la secuencia unidimensional $x[n]$, función de una única variable discreta, se convierte en una función bidimensional de la variable temporal $n$, que es discreta, y de la variable de frecuencia $\lambda$ , que es continua. Nótese que la STFT es periódica en $\lambda$ con periodo $2\pi$ y, por tanto, sólo es necesario considerar valores de $\lambda$ en el intervalo $0 \leq \lambda < 2\pi$ o en cualquier otro intervalo de longitud $2\pi$ .

Denominaremos $\lambda$ a la variable de frecuencia de la STFT para diferenciarla de la variable de frecuencia de la Transformada de Fourier en Tiempo Discreto (TFTD) convencional, que se denomina $\omega$ . Utilizamos la notación de corchete-paréntesis $X[n,\lambda)$ como recordatorio de que $n$ es una variable discreta y $\lambda$ una variable en continua.

La Ecuación 6.1 se puede interpretar como la transformada de Fourier de la señal desplazada $x[n+m]$, vista a través de la ventana $w[m]$. La ventana tiene un origen estacionario y, a medida que $n$ cambia, la señal se desliza por la ventana de forma que en cada valor de $n$ se ve una parte diferente de la señal. Como ilustración consideremos el siguiente ejemplo.

El espectrograma es una representación en función del índice $n$ y la frecuencia $\lambda/(2\pi)$ del módulo de la STFT. Esta técnica la vamos a utilizar para terminar de adecuar nuestros clips de audio a la red neuronal convolucional, debido que los clips de audio son señales 1D y una red neuronal convolucional requiere de una señal de entrada de dos dimensiones, es decir, imágenes. Por ende, alimentaremos nuestra CNN con espectrogramas de los clips de audio de tos.

En la creación de espectrogramas debemos tener muy en cuenta el tamaño y la forma de la ventana, ya que, influye directamente en la resolución temporal y frecuencial del mismo. Las ventanas largas producen mayor resolución en frecuencia, pero en aplicaciones prácticas de modelos de señales sinusoidales, las propiedades de la señal (*i.e.*, amplitudes, frecuencia) a menudo cambian con el tiempo. Sabiendo que, una señal no estacionaria es una señal cuyas propiedades varían con el tiempo y, por ejemplo, podría ser una suma de componentes sinusoidales cuyas amplitudes, frecuencias o fases variaran con el tiempo, son necesarios modelos de señal no estacionarios de este tipo para describir señales radar, sonar, de voz y de comunicación de datos. En nuestro caso al tratar con señales de voz, tendremos un modelo de señal no estacionario. Los espectrogramas se pueden utilizar como una forma de visualizar el cambio del contenido de frecuencia de una señal no estacionaria a lo largo del tiempo.

El principal propósito de la ventana en la STFT es limitar la extensión de la secuencia que se va a transformar, de forma que las características espectrales sean razonablemente estacionarias en el intervalo de duración de la ventana. Cuanto más rápidamente cambien las características de la señal, más corta deberá ser la ventana. Hay que tener en cuenta que, a medida que la longitud de la ventana decrece, la resolución en frecuencia también decrece. Por otra parte, a medida que decrece la longitud de la ventana, aumenta la potencialidad de resolver cambios entre en el tiempo. En consecuencia de aparece un compromiso en la selección de la longitud de la ventana, entre resolución en el tiempo y en la frecuencia.

En el caso de una señal que consiste en una secuencia de segmentos sonoros cuasi periódicos intercalados con segmentos sordos con aspecto de ruido. Se sugiere que si la longitud de la ventana L no es demasiado grande, las propiedades de la señal no cambiarán apreciablemente desde el comienzo del segmento hasta el final. Por tanto, la transformada discreta de Fourier (*Discrete Fourier Transform*, DFT) de un segmento de voz enventanado debería poner de manifiesto las propiedades en el dominio de la frecuencia de la señal en el instante correspondiente a la posición de la ventana. Por ejemplo, si la longitud de la ventana es suficiente para poder resolver los armónicos, la DFT de un segmento de voz enventanado debería mostrar una serie de picos en múltiplos enteros de la frecuencia fundamental de la señal en ese intervalo. Esto requiere normalmente que la ventana abarque varios periodos de la forma de onda. Si la ventana es demasiado corta, los armónicos no se resolverán, pero la forma general del espectro será todavía evidente. Esto es un caso típico del compromiso entre resolución temporal y resolución en frecuencia que aparece en al análisis de señales no estacionarias. Si la ventana es demasiado grande, las propiedades de la

señal pueden cambiar demasiado en la duración de la ventana. Si la ventana es demasiado corta, la resolución de las componentes de banda estrecha será sacrificada.

El espectrograma de banda ancha se produce utilizando una ventana que es relativamente corta en el dominio del tiempo, y se caracteriza por tener una resolución pobre en el dominio de la frecuencia y una buena resolución en la dimensión temporal. En el espectrograma de banda estrecha, se utiliza una ventana más larga, para proporcionar una mayor resolución en frecuencia, con la correspondiente disminución de resolución temporal.

Se sugiere que si estamos utilizando la STFT para obtener una estimación dependiente del tiempo del espectro de frecuencias de una señal, es deseable suavizar la ventana para reducir los lóbulos laterales y utilizar una ventana tan larga como sea posible para aumentar la resolución en frecuencia [210].

Nosotros vamos a escalar la longitud de la ventana en función de la frecuencia de muestreo, de la siguiente manera

$$nperseg = \left\lceil \frac{WindowSize \cdot SampleRate}{10^3} \right\rceil \tag{6.2}$$

siendo el valor del parámetro $WindowSize$, una cifra utilizada para realizar una correcta escala. En nuestro caso el valor óptimo fue $WindowSize = 10$, donde la longitud de ventana es 160 para una frecuencia de muestreo de 16000 Hz y 89 para una frecuencia de muestreo de 8820 Hz.

Con respecto a la forma de la ventana, debemos saber que los extremos de la ventana caen a cero suavemente en las ventanas de Hamming, Hanning y Blackman. En dichas ventanas los lóbulos laterales reducen enormemente su amplitud. Sin embargo, se paga el precio de un lóbulo principal mucho más ancho [211]. Esto hace que nos decantemos por una de estas tres ventanas, en concreto la de Hanning.

El tamaño del *overlap* también es de capital importancia, debido a que determina cuanto se solapan las ventanas. Cuanto más se solapen las ventanas, mayor será la resolución temporal y también aumentará el tamaño de la imagen en el eje temporal. Nosotros vamos a escalar el *overlap* de la ventana en función de la frecuencia de muestreo, según

$$noverlap = \left\lceil \frac{StepSize \cdot SampleRate}{10^3} \right\rceil \tag{6.3}$$

siendo el valor del parámetro $StepSize$, una cifra utilizada para realizar una correcta escala. En nuestro caso el valor óptimo fue $StepSize = 0$, donde el *overlap* es nulo.

Por último, el tamaño de la DFT, es el número de puntos que utilizamos para hacer la DFT, por ende, contra más grande es el tamaño de la DFT, mayor resolución frecuencial tendremos y más grande será la imagen en el eje frecuencial. Nosotros hemos puesto el tamaño de la DFT igual a la longitud de la ventana.

Los espectrogramas resultantes tienen un tamaño de 45x100 píxeles para una frecuencia de muestreo de 8820 Hz y un tamaño de 81x100 píxeles para una frecuencia de muestreo de 16000 Hz. Una vez realizado dicho espectrograma calculamos su logaritmo, ya que, la Naturaleza ha dispuesto que la sensación que un estímulo acústico produce sea, aproximadamente, proporcional al logaritmo decimal de la intensidad de este y de esta forma nos adaptamos al oido humano [212]. Además debemos normalizar el espectrograma logaritmico entre 0 y 1, para alimentar nuestra CNN de manera óptima.

## 6.3. Diseño de la CNN

Como hemos comentado anteriormente, las redes neuronales convolucionales son muy eficientes en la clasificación de imágenes, por lo tanto, crearemos una CNN para cada uno de los estudios que vamos a realizar.

### 6.3.1. Detección de tos

En el estudio de la detección de tos hemos utilizado dos arquitecturas de redes neuronales convolucionales, cada una de las cuales con características interesantes en nuestro estudio. Pero antes presentamos el esquema de la Figura 6.6 que resume el proceso para generar todas las imágenes que nos servirán como entrada de las CNNs que utilizaremos en detección de tos para los diferentes entrenamientos.

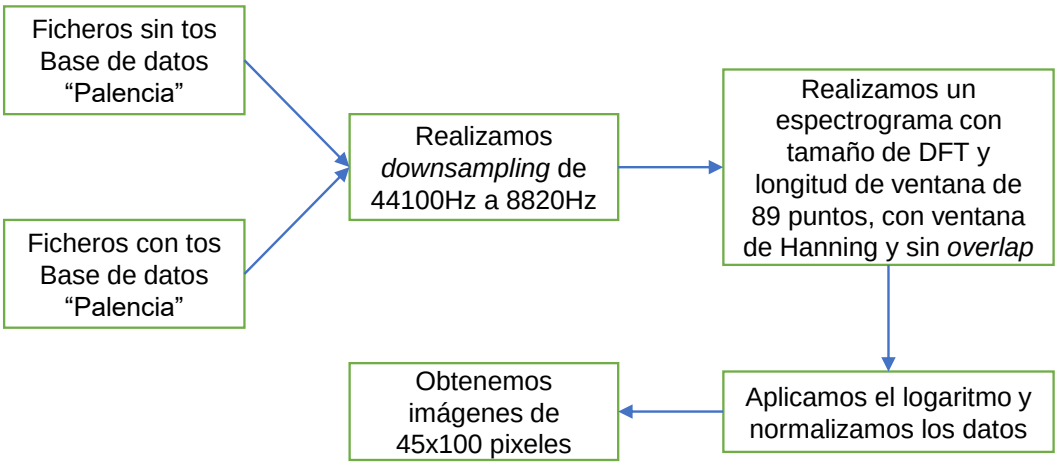



Figura 6.6: Esquema de la generación de las imágenes que nos servirán de entrada para las CNNs de detección de tos.

A continuación explicaremos como es la primera arquitectura que denominaremos CNN A es la que vemos en la Figura 6.7, esta arquitectura como veremos posteriormente tiene la característica de tener más especificidad que sensibilidad. La primera capa es una capa convolucional de 32 filtros con un kernel (2, 2), la función de activación es una ReLU y admite una entrada con la forma (45, 100, 1), ya que, el tamaño de los espectrogramas logarítmicos normalizados es de (45, 100), pero la capa convolucional necesita una tercera dimensión. Seguidamente tenemos una capa de *MaxPoolling2D*, con un tamaño de *pool* de (2, 2), esta capa reduce la dimensionalidad de los datos para no tener una excesiva carga computacional. A continuación, conectamos una capa de *dropout*, la cual tiene la capacidad de reducir el sobreajuste de las redes neuronales, durante el entrenamiento, cierto número de salidas de capa se ignoran aleatoriamente o se “eliminan”, en nuestro caso ignoramos aleatoriamente el 10 % de las salidas.

Todas las capas de *MaxPoolling2D* y *dropout* que vemos posteriormente tienen las mismas características que las descritas anteriormente. El número de capas óptimo, lo hemos determinado experimentalmente, viendo que con el número de capas que vemos en la Figura 6.7, es el que mejor resultados nos proporciona. También hemos comprobado experimentalmente que lo óptimo es ir doblando el número de filtros en las capas convolucionales utilizadas, esta configuración permite realizar entrenamientos de menor duración ofreciendo buenos resultados.

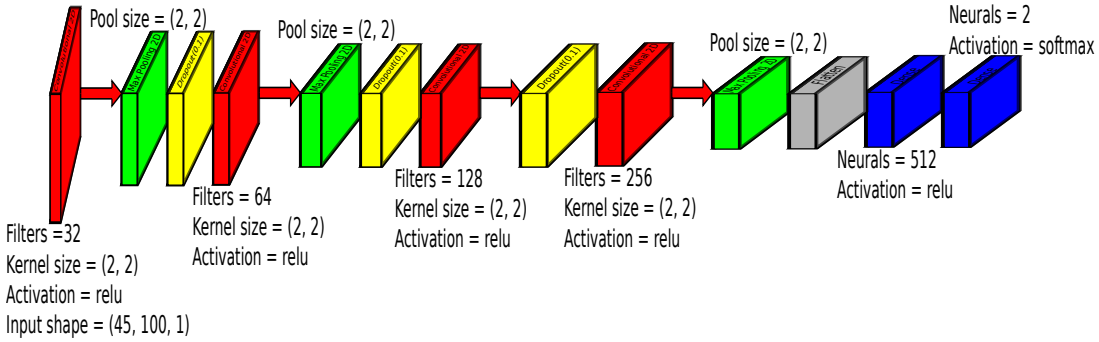



Figura 6.7: Arquitectura de la CNN A utilizada para detectar clips de audios con tos. La entrada a la red es un espectrograma STFT de un segundo. La red consta de 4 capas convolucionales donde las capas convolucionales las vemos en rojo, una capa *flatten* en gris, una capa densa en azul y una capa de clasificación softmax en azul. Además vemos 3 capas de *MaxPoolling2D* en verde y 3 capas de *dropout* en amarillo.

La capa *flatten* es una capa que aplana los datos para las capas posteriores, siendo estas capas densas. Como últimas capas, tenemos una capa densa de 512 neuronas, con la función de activación ReLU y una capa densa de

2 neuronas, ya que es el número de clases que tenemos que clasificar, con una función de activación *softmax*, la salida de la función *softmax* puede ser utilizada para representar una distribución categórica– la distribución de probabilidad sobre las diferentes posibles salidas.

La segunda arquitectura que denominaremos CNN B es la que vemos en la Figura 6.8, donde vemos que desaparecen las capas de *dropout* con respecto a la arquitectura anterior. De manera teórica podríamos pensar que al eliminar estas capas tenderíamos a un mayor sobreajuste en la red neuronal, pero como comprobaremos posteriormente el efecto que provoca es conseguir tener más balanceada la sensibilidad y la especificidad de la red. Esta característica diferenciadora con respecto a nuestra primera red puede ser interesante para nuestro estudio.

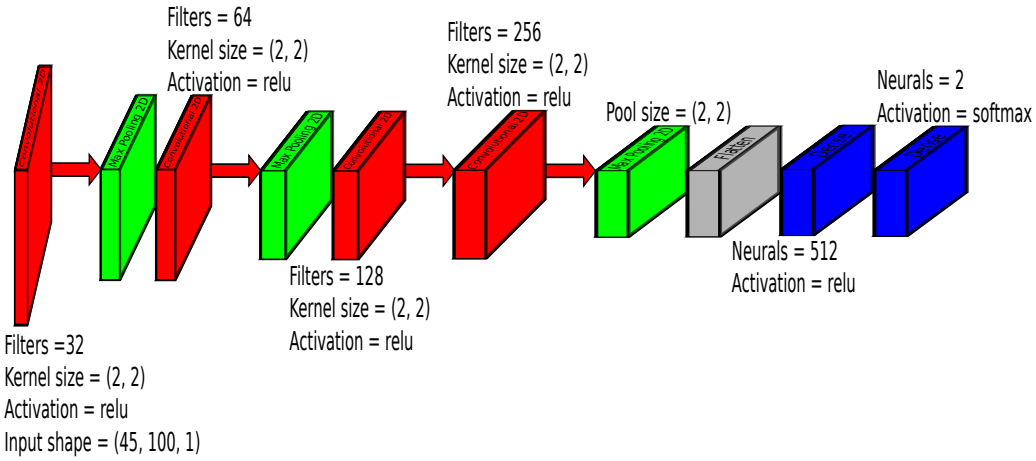



Figura 6.8: Arquitectura de la CNN B utilizada para detectar clips de audios con tos. La entrada a la red es un espectrograma STFT de un segundo. La red consta de 4 capas convolucionales donde las capas convolucionales las vemos en rojo, una capa *flatten* en gris, una capa densa en azul y una capa de clasificación softmax en azul. Además vemos 3 capas de *MaxPoolling2D* en verde.

### 6.3.2. Enfermedades

En el esquema de la Figura 6.9 se resume el proceso para generar todas las imágenes que nos servirán como entrada de la CNN que utilizaremos en clasificación de tos según la enfermedad subyacente para los diferentes entrenamientos.

Para este estudio las características diferenciadoras son mucho menos evidentes, debido a que todos los clips de audio son clips de audio de tos, por lo que ha sido necesario añadir dos capas convoluciones con respecto al anterior caso de estudio.

La arquitectura de red utilizada la podemos ver en la Figura 6.10. La primera capa es una capa convolucional de 32 filtros con un kernel (2, 2), la función de activación es una ReLU y admite una entrada con la forma (81, 100, 1), ya que, el tamaño de los espectrogramas logarítmicos normalizados es de (81, 100), pero la capa convolucional necesita una tercera dimensión. Seguidamente tenemos una capa de *MaxPoolling2D*, con un tamaño de *pool* de (2, 2), esta capa reduce la dimensionalidad de los datos para no tener una excesiva carga computacional. También tenemos una capa de *batch normalization*, la cual vuelve a normalizar los datos en el eje Z después de pasar por varias capas convolucionales, esta capa consigue que la red converja con mayor celeridad.

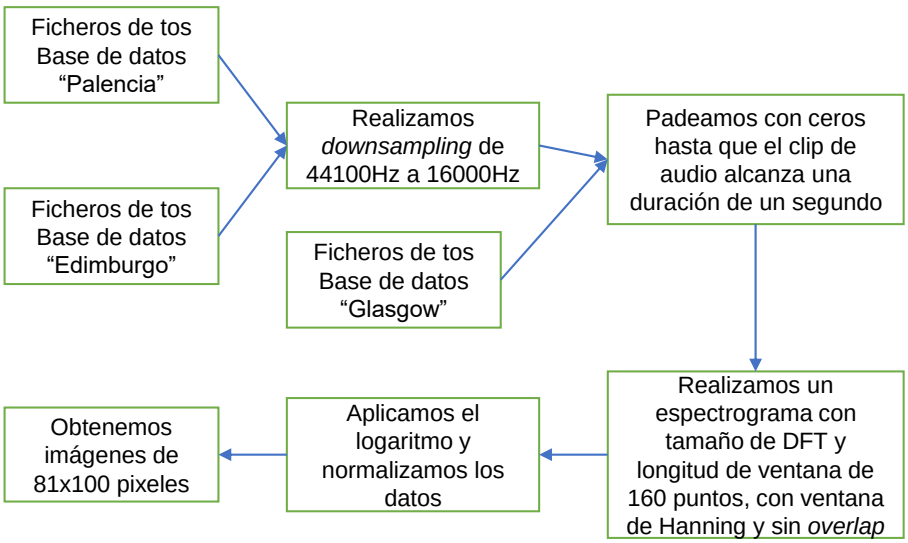



Figura 6.9: Esquema de la generación de las imágenes que nos servirán de entrada para la CNN de clasificación de tos según la enfermedad subyacente.

Todas las capas de *MaxPoolling2D* que vemos posteriormente tienen las mismas características que las descritas anteriormente. El número de capas óptimo, lo hemos determinado experimentalmente, viendo que con el número de capas que vemos en la Figura 6.10, es el que mejor resultados nos proporciona. También hemos comprobado experimentalmente que lo más óptimo es ir doblando casi siempre el número de filtros en las capas convolucionales utilizadas, esta configuración permite realizar entrenamientos de menor duración ofreciendo buenos resultados.

La capa *flatten* es una capa que aplana los datos para las capas posteriores, siendo estas capas densas. Como últimas capas, tenemos una capa densa de 1024 neuronas, con la función de activación ReLU y una capa densa de 4 neuronas, ya que es el número de clases que tenemos que clasificar, con una función de activación *softmax*, la salida de la función *softmax* puede ser utilizada para representar una distribución categórica– la distribución de probabilidad sobre las diferentes posibles salidas.

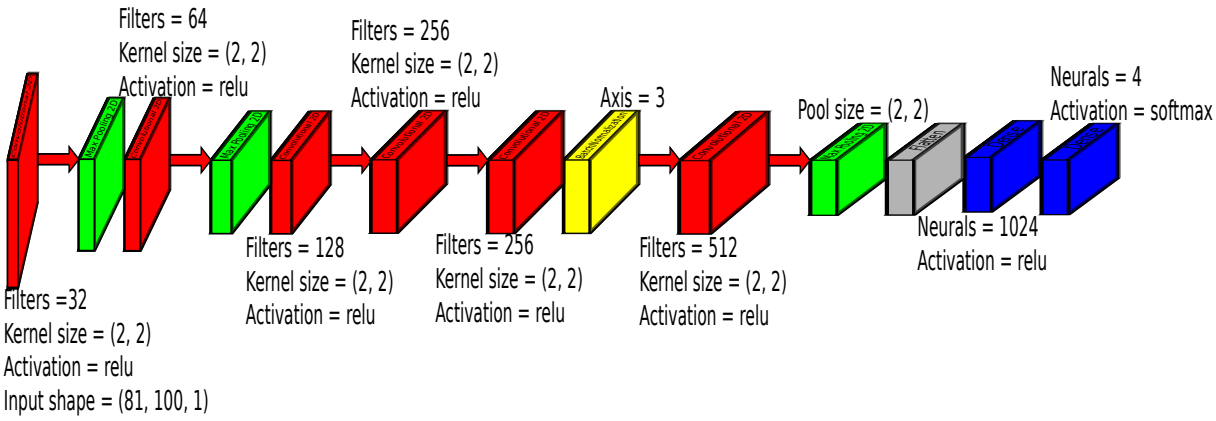



Figura 6.10: Arquitectura de la CNN utilizada para diagnosticar enfermedades. La entrada a la red es un espectrograma STFT de un segundo. La red consta de 6 capas convolucionales donde las capas convolucionales las vemos en rojo, una capa *flatten* en gris, una capa densa en azul y una capa de clasificación softmax en azul. Además vemos 3 capas de *MaxPoolling2D* en verde y una capa de *batch normalization* en amarillo.

Cabe destacar que en el diagnóstico de enfermedades, para el caso de clasificar entre dos enfermedades hemos modificado la ultima capa densa reduciéndola a 2 neuronas, debido a que esta capa siempre debe tener tantas neuronas como clases que clasificar.

## 6.4. Entrenamiento

El entrenamiento de nuestra red neuronal tiene el objetivo de que sea capaz de clasificar los datos introducidos en las clases deseadas. En un caso clasificaremos los clips de audio en clips de audio con tos y sin tos y en el otro caso de estudio que hemos descrito anteriormente clasificaremos clips de audio con tos según la enfermedad subyacente.

### 6.4.1. Parámetros de entrenamiento

La red neuronal que hemos definido en la sección 6.3, debe completarse agregando una función de pérdida, un optimizador y las métricas de rendimiento. Esto se llama modelo de "compilación".

Una función de pérdida (o función objetivo, o función de puntuación de optimización) es uno de los tres parámetros (el primero, en realidad) requerido para compilar un modelo. La función de pérdida utilizada en este caso es *categorical crossentropy*, esta es la función de pérdida más utilizada en tareas de clasificación multiclase, máxime si sus objetivos (*targets*) están codificados de forma instantánea (*one-hot encoded*).

El optimizador implementado en nuestro caso es AdaMax (del cual hablamos en la sección 4.1.4), cuyos parámetros configurables son $\alpha$, $\beta_1$ y $\beta_2$, y los valores utilizados para nuestro estudio son $\alpha = 0,002$, $\beta_1 = 0,9$ y $\beta_2 = 0,999$ [158], [159].

La métrica de rendimiento que nos interesa es la exactitud (*accuracy*) de la clasificación, para de esta forma ver en cada época del entrenamiento el *accuracy* de la validación. Además también veremos de forma predeterminada la pérdida (*loss*) de la validación en cada época.

Hemos determinado experimentalmente que para obtener resultados óptimos 50 es número de épocas apropiadas. Por otro lado, el tamaño de *batch* es 128 en el caso de clasificación de clips de audio con tos y sin tos, y 32 en el caso de clasificación de clips de audio con tos según la enfermedad subyacente. Además utilizamos el 20 % del conjunto de entrenamiento para realizar la validación.

### 6.4.2. Características de los entrenamientos

En esta sección vamos a determinar como son los conjuntos de entrenamiento y test en cada uno de los estudios realizados. Es de capital importancia destacar que las clases han sido balanceadas previamente con el objetivo de evitar sobre entrenar la red para alguna de las clases a clasificar. Además para saber si la red aprendía correctamente los pacientes que están en el conjunto de test nunca están en el conjunto de entrenamiento, debido a la necesidad de eliminar el sesgo producido si el paciente estuviese en ambos conjuntos. A continuación, mostramos un caso en el que los pacientes están en el conjunto de entrenamiento y en el conjunto de test.

| | SEN/recall/TPR | SPE/TNR | ACC | PPV/precision | NPV |
|---|---|---|---|---|---|
| **Acute** | 0,86988 | 0,920229 | 0,908116 | 0,77551 | 0,957126 |
| **COPD** | 0,862827 | 0,902204 | 0,891304 | 0,771536 | 0,945004 |
| **Lung cancer** | 0,721845 | 0,93955 | 0,893043 | 0,764368 | 0,925563 |
| **non COPD** | 0,672414 | 0,948057 | 0,873913 | 0,82649 | 0,887199 |
| **macro avg** | 0,781741 | 0,92751 | 0,891594 | 0,784476 | 0,928723 |
| **weighted avg** | 0,794619 | 0,924637 | 0,892386 | 0,783188 | 0,931703 |
| **micro avg** | 0,783188 | 0,927729 | 0,891594 | 0,783188 | 0,927729 |

Tabla 6.4: Media de las métricas sin *k-folds* en la clasificación de enfermedades.

Podemos observar que si entrenamos un clasificador de enfermedades en el que los pacientes están en el conjunto de entrenamiento y en el conjunto de test, los resultados obtenidos apreciables en la Tabla 6.4 y en la Figura 6.11 son bastante buenos, pero estos resultados son engañosos, ya que la red neuronal esta aprendiendo como es la "firma vocal" de los pacientes para ayudarse en el diagnóstico de la enfermedad que deduce a partir de cada clip de tos.

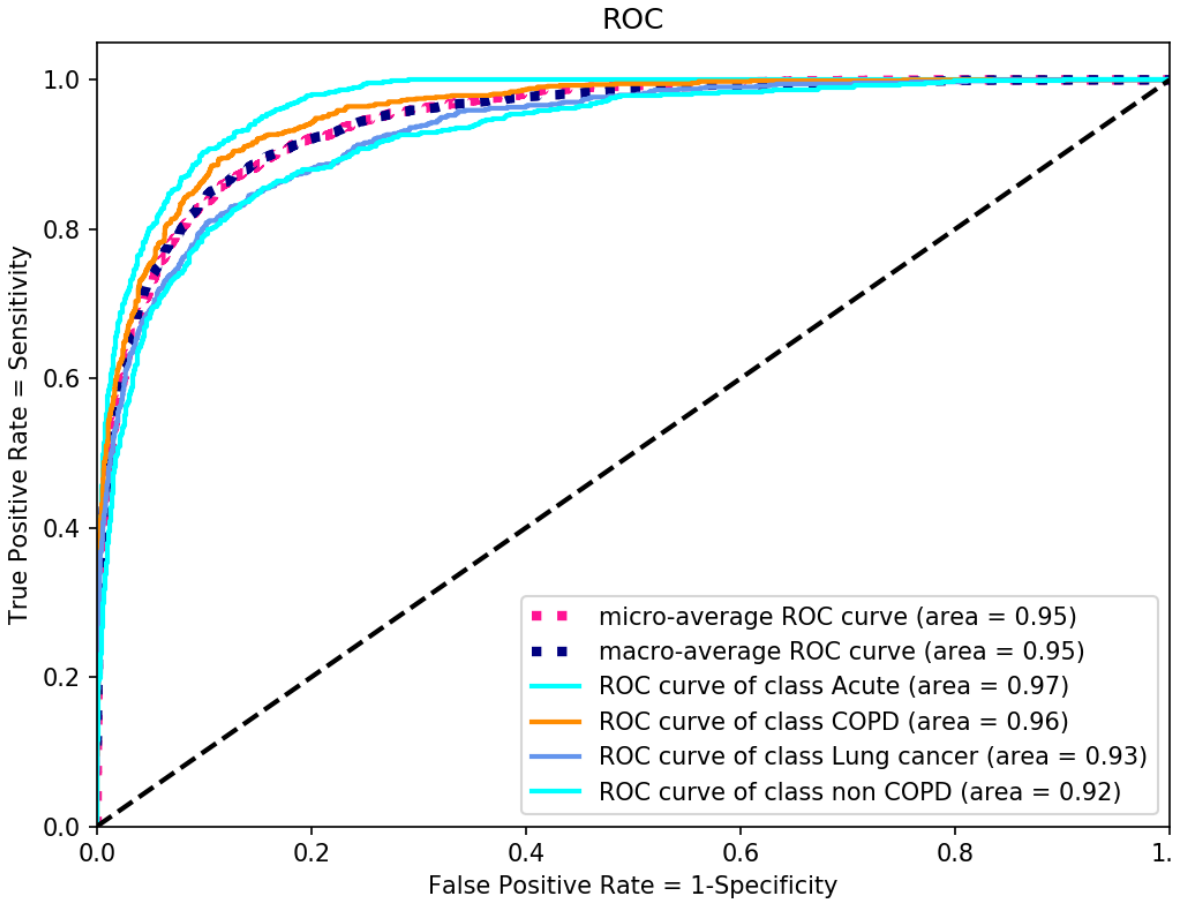



Figura 6.11: Curvas ROC sin *k-folds* en la clasificación de enfermedades.

Posteriormente en la sección 7.2, veremos este mismo estudio, pero realizando la debida separación de pacientes entre los conjuntos de entrenamiento y test.

### Detección de tos

En el caso de la detección de tos vamos a realizar una separación de pacientes entre los conjuntos de entrenamiento y test, en el que buscamos únicamente que todos los conjuntos de test tengan el mismo número de clips de tos y no tos. Recordamos que para este estudio solo utilizamos la base de datos "Palencia", por lo que nos referiremos a los pacientes de la Tabla 6.1 simplemente con su ID.

A continuación se plantean diferentes estructuras de *cross-validation*, siendo estas *10-fold cross-validation*, *5-fold cross-validation* y *3-fold cross-validation*.

| **Ficheros** | 4806 | 4066 | 3158 | 3766 | 3552 | 3504 | 3748 | 3594 | 3196 | 3474 |
|---|---|---|---|---|---|---|---|---|---|---|
| ***Fold* 0** | **7** | 17 | 13 | 14, 20 | 9, 10 | 11, 15 | 3, 4, 6 | 0, 16, 19 | 12, 18 | 2, 5, 8 |
| ***Fold* 1** | 7 | **17** | 13 | 14, 20 | 9, 10 | 11, 15 | 3, 4, 6 | 0, 16, 19 | 12, 18 | 2, 5, 8 |
| ***Fold* 2** | 7 | 17 | **13** | 14, 20 | 9, 10 | 11, 15 | 3, 4, 6 | 0, 16, 19 | 12, 18 | 2, 5, 8 |
| ***Fold* 3** | 7 | 17 | 13 | **14, 20** | 9, 10 | 11, 15 | 3, 4, 6 | 0, 16, 19 | 12, 18 | 2, 5, 8 |
| ***Fold* 4** | 7 | 17 | 13 | 14, 20 | **9, 10** | 11, 15 | 3, 4, 6 | 0, 16, 19 | 12, 18 | 2, 5, 8 |
| ***Fold* 5** | 7 | 17 | 13 | 14, 20 | 9, 10 | **11, 15** | 3, 4, 6 | 0, 16, 19 | 12, 18 | 2, 5, 8 |
| ***Fold* 6** | 7 | 17 | 13 | 14, 20 | 9, 10 | 11, 15 | **3, 4, 6** | 0, 16, 19 | 12, 18 | 2, 5, 8 |
| ***Fold* 7** | 7 | 17 | 13 | 14, 20 | 9, 10 | 11, 15 | 3, 4, 6 | **0, 16, 19** | 12, 18 | 2, 5, 8 |
| ***Fold* 8** | 7 | 17 | 13 | 14, 20 | 9, 10 | 11, 15 | 3, 4, 6 | 0, 16, 19 | **12, 18** | 2, 5, 8 |
| ***Fold* 9** | 7 | 17 | 13 | 14, 20 | 9, 10 | 11, 15 | 3, 4, 6 | 0, 16, 19 | 12, 18 | **2, 5, 8** |

Tabla 6.5: Descripción de los *folds* en el *10-fold cross-validation* utilizado para entrenar la red en la detección de tos.

La Tabla 6.5 describe la composición de los *folds*, del *10-fold cross-validation*, que hemos realizado. Resaltamos la celda de la tabla en la que están los pacientes del conjunto de test, con un color de fondo naranja, y vemos en la fila superior cuantos clips de audio de tos tiene este conjunto de test. Por ende, el resto de pacientes pertenecen al conjunto de entrenamiento de dicho *fold*.

| **Ficheros** | 7572 | 7394 | 7238 | 7390 | 7270 |
|---|---|---|---|---|---|
| ***Fold* 0** | **0, 3, 5, 7** | 2, 4, 12, 17 | 10, 13, 16, 19 | 6, 8, 11, 20 | 9, 14, 15, 18 |
| ***Fold* 1** | 0, 3, 5, 7 | **2, 4, 12, 17** | 10, 13, 16, 19 | 6, 8, 11, 20 | 9, 14, 15, 18 |
| ***Fold* 2** | 0, 3, 5, 7 | 2, 4, 12, 17 | **10, 13, 16, 19** | 6, 8, 11, 20 | 9, 14, 15, 18 |
| ***Fold* 3** | 0, 3, 5, 7 | 2, 4, 12, 17 | 10, 13, 16, 19 | **6, 8, 11, 20** | 9, 14, 15, 18 |
| ***Fold* 4** | 0, 3, 5, 7 | 2, 4, 12, 17 | 10, 13, 16, 19 | 6, 8, 11, 20 | **9, 14, 15, 18** |

Tabla 6.6: Descripción de los *folds* en el *5-fold cross-validation* utilizado para entrenar la red en la detección de tos.

La Tabla 6.6 describe la composición de los *folds*, del *5-fold cross-validation*, y la Tabla 6.7 describe la composición de los *folds*, del *3-fold cross-validation*, que hemos realizado. Estas tablas proporcionan la información de manera análoga a como la proporciona la Tabla 6.5. Aunque en este caso resaltamos la celda de la tabla en la que están los pacientes del conjunto de test, con un color de fondo cían.

| **Ficheros** | 12210 | 12178 | 12476 |
|---|---|---|---|
| ***Fold* 0** | **3, 7, 11, 12, 15, 18** | 4, 5, 9, 14, 16, 17 | 0, 2, 6, 8, 10, 13, 19, 20 |
| ***Fold* 1** | 3, 7, 11, 12, 15, 18 | **4, 5, 9, 14, 16, 17** | 0, 2, 6, 8, 10, 13, 19, 20 |
| ***Fold* 2** | 3, 7, 11, 12, 15, 18 | 4, 5, 9, 14, 16, 17 | **0, 2, 6, 8, 10, 13, 19, 20** |

Tabla 6.7: Descripción de los *folds* en el *3-fold cross-validation* utilizado para entrenar la red en la detección de tos.

En adelante, para no sobrecargar el texto se describirán de forma explicita los conjuntos de test, mientras que los conjuntos de entrenamiento se pueden inferir de forma implícita.

### Enfermedades

En esta subsección, hablaremos de cómo hemos realizado la configuración de los *folds* para la clasificación de enfermedades, además de mostrar cual es la configuración de los *folds*, explicando cual es el conjunto de entrenamiento y el conjunto de test en cada caso.

Con respecto a la identificación de los pacientes, en este caso es más complicado que en el caso anterior, ya que utilizamos las tres bases de datos descritas en la sección 6.1. Para el caso en el que un paciente pertenezca a la base de datos "Palencia", su identificación será palXX, siendo XX el ID asignado en la Tabla 6.1. Si un paciente pertenece a la base de datos "Edimburgo", su identificación será ediXX, siendo XX el ID asignado en la Tabla 6.3. Por último, si un paciente pertenece a la base de datos "Glasgow", su identificación será glaXX, siendo XX el ID asignado en la Tabla 6.2.

La configuración de los *folds* se ha realizado teniendo en cuenta que en cada conjunto de test debía haber un paciente de cada clase y además que los clips de audio con tos de cada clase estuviera lo más balanceado posible.

| | | **Pacientes** | | | |
|---|---|---|---|---|---|
| | **# Ficheros** | Tos aguda | EPOC | Cáncer de pulmón | no EPOC |
| ***Fold* 0** | 4305 | pal13 | pal12 | pal06 | edi10, gla04 |
| ***Fold* 1** | 2973 | pal04 | pal05 | pal08 | pal18, edi05 |
| ***Fold* 2** | 2680 | pal14 | edi11, gla03 | gla05 | pal10, edi01, gla01 |
| ***Fold* 3** | 1364 | pal03 | edi03, edi04, edi09 | gla11 | pal02, edi02, edi06 |
| ***Fold* 4** | 400 | pal00 | edi07, gla09 | gla15 | edi08 |

Tabla 6.8: Descripción de los conjuntos de test en los *folds* del *5-fold cross-validation* utilizado para entrenar la red en la clasificación de enfermedades.

En este caso de estudio en el que buscamos clasificar los clips de audio con tos según enfermedades subyacentes, aún juntando las tres bases de datos, no tenemos suficientes pacientes como para poder realizar un *10-fold cross-validation*. Esto se debe a que en cada *fold*, debemos tener al menos un paciente de cada una de las clases que queremos clasificar, por lo tanto, en cada *fold* al menos debemos de tener cuatro pacientes siendo cada uno de estos pacientes de diferente clase.

Podemos ver en la Tabla 6.8 la composición de los conjuntos de test de los diferentes *folds*, cuando realizamos un *5-fold cross-validation*. Observando en la primera columna el número de los clips de audio con tos y en la segunda columna los pacientes que alberga cada *fold* en su conjunto de test. De manera implícita se describe el número de clips de audio con tos que componen el entrenamiento de dicho *fold*, ya que este es la suma del número de clips de audio con tos del resto de *folds*. Análogamente a la forma de saber número de clips de audio con tos del entrenamiento en cada *fold* podemos deducir los pacientes que alberga cada conjunto de entrenamiento.

| | # Ficheros | Pacientes |
|---|---|---|
| ***Fold* 0** | 4305 | pal06, pal12, pal13, edi10, gla04 |
| ***Fold* 1** | 3373 | pal00, pal04, pal05, pal08, pal18, edi05, edi07, edi08, gla09, gla15 |
| ***Fold* 2** | 4044 | pal02, pal03, pal10, pal14, edi01, edi02, edi03, edi04, edi06, edi09, edi11, gla01, gla03, gla05, gla11 |

Tabla 6.9: Descripción de los conjuntos de test en los *folds* del *3-fold cross-validation* utilizado para entrenar la red en la clasificación de enfermedades.

La Tabla 6.9 describe la composición de los *folds*, cuando realizamos un *3-fold cross-validation*. Esta tabla proporciona la información e igual manera a como la proporciona la Tabla 6.8.

Debido a los pocos datos que tenemos, con el objetivo de mejorar los resultados, en vez de buscar clasificar los clips de audio con tos entre las cuatro clases, vamos a fragmentar nuestro estudio buscando clasificar solamente entre dos clases y de esta manera viendo que clases son mas diferenciables para nuestra red neuronal convolucional.

### Tos aguda frente a EPOC

Lo primero que haremos es clasificar entre tos aguda y EPOC, ya que, son las clases en las que más clips de audio con tos tenemos. Esto significa que son las dos clases de las que más información disponemos y por tanto que mejor podrá clasificar nuestra red. Con el objetivo de buscar aprovechar los datos de los que disponemos con mayor eficiencia realizaremos primeramente un *10-fold cross-validation*.

| | # Ficheros | Pacientes |
|---|---|---|
| ***Fold* 0** | 2401 | pal07 |
| ***Fold* 1** | 2033 | pal17 |
| ***Fold* 2** | 1579 | pal13 |
| ***Fold* 3** | 1191 | gla02 |
| ***Fold* 4** | 766 | pal05 |
| ***Fold* 5** | 687 | pal04, edi04 |
| ***Fold* 6** | 708 | edi03, edi11, gla03 |
| ***Fold* 7** | 684 | edi07, edi09, gla09, gla10 |
| ***Fold* 8** | 769 | pal00, pal14 |
| ***Fold* 9** | 845 | pal03, gla07 |

Tabla 6.10: Descripción de los conjuntos de test en los *folds* del *10-fold cross-validation* utilizado para entrenar la red en la clasificación de tos aguda frente a EPOC.

Podemos ver en la Tabla 6.10 la composición de los conjuntos de test de los diferentes *folds*, cuando realizamos un *10-fold cross-validation*. El inconveniente de realizar los tests con esta composición de *folds* es que no podemos ver como se comportan todas las clases en cada *fold* aunque si que podemos hacer que al finalizar el entrenamiento diagnostique a los pacientes que pertenecen al conjunto de test. Pero dado que nos interesa saber como se comportan todas las clases en cada *fold*, vamos a realizar un *5-fold cross-validation*.

| | # Ficheros | Pacientes | |
|---|---|---|---|
| | | Tos aguda | EPOC |
| ***Fold* 0** | 4927 | pal07 | pal17, gla07 |
| ***Fold* 1** | 3246 | pal13 | pal05, pal12 |
| ***Fold* 2** | 1136 | pal04 | gla03 |
| ***Fold* 3** | 1013 | pal14 | gla10 |
| ***Fold* 4** | 1051 | pal00, pal03 | edi03, edi04, edi07, edi09, edi11, gla09 |

Tabla 6.11: Descripción de los conjuntos de test en los *folds* del *5-fold cross-validation* utilizado para entrenar la red en la clasificación de tos aguda frente a EPOC.

La composición de los conjuntos de test de los diferentes *folds* para realizar un *5-fold cross-validation*, la podemos ver en la Tabla 6.11 y para realizar un *3-fold cross-validation*, la podemos ver en la Tabla 6.12. Se puede observar que en cada *fold* hay pacientes de las dos clases que queremos clasificar. De esta manera vemos como se comportan todas las clases en cada *fold*, y de esta manera ver de que pacientes la red neuronal es capaz de obtener mejores características. Además de poder ver en cada *fold* las métricas necesarias para evaluar el rendimiento de nuestra red neuronal después de haber sido entrenada.

| | # Ficheros | Pacientes | |
|---|---|---|---|
| | | Tos aguda | EPOC |
| **Fold 0** | 4927 | pal07 | pal17, gla07 |
| **Fold 1** | 3246 | pal13 | pal05, pal12 |
| **Fold 2** | 3200 | pal00, pal03, pal04, pal14 | edi03, edi04, edi07, edi09, edi11, gla03, gla09, gla10 |

Tabla 6.12: Descripción de los conjuntos de test en los *folds* del *3-fold cross-validation* utilizado para entrenar la red en la clasificación de tos aguda frente a EPOC.

### Tos aguda frente a cáncer de pulmón

Las Tablas 6.13 y 6.14 son análogas a las Tablas 6.11 y 6.12 para la ver la composición de los *folds* en el estudio de la clasificación de clips de audio con tos aguda frente a cáncer de pulmón.

| | # Ficheros | Pacientes | |
|---|---|---|---|
| | | Tos aguda | Cáncer de pulmón |
| ***Fold* 0** | 2500 | pal13 | pal06 |
| ***Fold* 1** | 1410 | pal04 | pal08 |
| ***Fold* 2** | 1288 | pal14 | gla05 |
| ***Fold* 3** | 686 | pal03 | gla11 |
| ***Fold* 4** | 309 | pal00 | gla15 |

Tabla 6.13: Descripción de los conjuntos de test en los *folds* del *5-fold cross-validation* utilizado para entrenar la red en la clasificación de tos aguda frente a cáncer de pulmón.

| | # Ficheros | Pacientes | |
|---|---|---|---|
| | | Tos aguda | Cáncer de pulmón |
| ***Fold* 0** | 2500 | pal13 | pal06 |
| ***Fold* 1** | 1719 | pal00, pal04 | pal08, gla15 |
| ***Fold* 2** | 1974 | pal03, pal14 | gla05, gla11 |

Tabla 6.14: Descripción de los conjuntos de test en los *folds* del *3-fold cross-validation* utilizado para entrenar la red en la clasificación de tos aguda frente a cáncer de pulmón.

### Tos aguda frente a tos crónica que no es EPOC

Las Tablas 6.15 y 6.16 son análogas a las Tablas 6.11 y 6.12 para la ver la composición de los *folds* en el estudio de la clasificación de clips de audio con tos aguda frente a tos crónica que no es EPOC.

| | # Ficheros | Pacientes | |
|---|---|---|---|
| | | Tos aguda | no EPOC |
| ***Fold* 0** | 3166 | pal13 | pal18, edi02, gla04 |
| ***Fold* 1** | 1204 | pal04 | edi08, edi10, gla08 |
| ***Fold* 2** | 1012 | pal14 | pal10 |
| ***Fold* 3** | 693 | pal03 | gla12 |
| ***Fold* 4** | 527 | pal00 | pal02, edi05 |

Tabla 6.15: Descripción de los conjuntos de test en los *folds* del *5-fold cross-validation* utilizado para entrenar la red en la clasificación de tos aguda frente a tos crónica que no es EPOC.

| | # Ficheros | Pacientes | |
|---|---|---|---|
| | | Tos aguda | no EPOC |
| ***Fold* 0** | 3166 | pal13 | pal18, edi02, gla04 |
| ***Fold* 1** | 1731 | pal00, pal04 | pal02, edi05, edi08, edi10, gla08 |
| ***Fold* 2** | 1705 | pal03, pal14 | pal10, gla12 |

Tabla 6.16: Descripción de los conjuntos de test en los *folds* del *3-fold cross-validation* utilizado para entrenar la red en la clasificación de tos aguda frente a tos crónica que no es EPOC.

### EPOC frente a cáncer de pulmón

Las Tablas 6.17 y 6.18 son análogas a las Tablas 6.11 y 6.12 para la ver la composición de los *folds* en el estudio de la clasificación de clips de audio con EPOC frente a cáncer de pulmón.

| | # Ficheros | Pacientes | |
|---|---|---|---|
| | | EPOC | Cáncer de pulmón |
| ***Fold* 0** | 1822 | pal12 | pal06 |
| ***Fold* 1** | 1575 | pal05 | pal08 |
| ***Fold* 2** | 1389 | edi11, gla03 | gla05 |
| ***Fold* 3** | 657 | edi03, edi04, edi09 | gla11 |
| ***Fold* 4** | 85 | edi07, gla09 | gla15 |

Tabla 6.17: Descripción de los conjuntos de test en los *folds* del *5-fold cross-validation* utilizado para entrenar la red en la clasificación de EPOC frente a cáncer de pulmón.

| | # Ficheros | Pacientes | |
|---|---|---|---|
| | | EPOC | Cáncer de pulmón |
| ***Fold* 0** | 1822 | pal12 | pal06 |
| ***Fold* 1** | 1660 | pal05, edi07, gla09 | pal08, gla15 |
| ***Fold* 2** | 2046 | edi03, edi04, edi09, edi11, gla03 | gla05, gla11 |

Tabla 6.18: Descripción de los conjuntos de test en los *folds* del *3-fold cross-validation* utilizado para entrenar la red en la clasificación de EPOC frente a cáncer de pulmón.

### EPOC frente a tos crónica que no es EPOC

Las Tablas 6.19 y 6.20 son análogas a las Tablas 6.11 y 6.12 para la ver la composición de los *folds* en el estudio de la clasificación de clips de audio con EPOC frente a tos crónica que no es EPOC.

| | # Ficheros | Pacientes | |
|---|---|---|---|
| | | EPOC | no EPOC |
| ***Fold* 0** | 2690 | pal12, edi03, edi04, edi07, edi09, edi11, gla09 | edi02, edi05, edi08, edi10, gla01, gla04 |
| ***Fold* 1** | 1537 | pal05 | pal18, edi01 |
| ***Fold* 2** | 1075 | gla03 | edi06, gla08 |
| ***Fold* 3** | 1017 | gla10 | pal10 |
| ***Fold* 4** | 996 | gla07 | pal02, gla12 |

Tabla 6.19: Descripción de los conjuntos de test en los *folds* del *5-fold cross-validation* utilizado para entrenar la red en la clasificación de EPOC frente a tos crónica que no es EPOC.

| | # Ficheros | Pacientes | |
|---|---|---|---|
| | | EPOC | no EPOC |
| ***Fold* 0** | 2690 | pal12, edi03, edi04, edi07, edi09, edi11, gla09 | edi02, edi05, edi08, edi10, gla01, gla04 |
| ***Fold* 1** | 2533 | pal05, gla07 | pal02, pal18, edi01, gla12 |
| ***Fold* 2** | 2092 | gla03, gla10 | pal10, edi06, gla08 |

Tabla 6.20: Descripción de los conjuntos de test en los *folds* del *3-fold cross-validation* utilizado para entrenar la red en la clasificación de EPOC frente a tos crónica que no es EPOC.

#### Cáncer de pulmón frente a tos crónica que no es EPOC

Las Tablas 6.21 y 6.22 son análogas a las Tablas 6.11 y 6.12 para la ver la composición de los *folds* en el estudio de la clasificación de clips de audio con cáncer de pulmón frente a tos crónica que no es EPOC.

| | # Ficheros | Pacientes | |
|---|---|---|---|
| | | Cáncer de pulmón | no EPOC |
| ***Fold* 0** | 1825 | pal06 | edi10, gla04 |
| ***Fold* 1** | 1606 | pal08 | pal18, edi05 |
| ***Fold* 2** | 1571 | gla05 | pal10, edi01, gla01 |
| ***Fold* 3** | 689 | gla11 | pal02, edi02, edi06 |
| ***Fold* 4** | 94 | gla15 | edi08 |

Tabla 6.21: Descripción de los conjuntos de test en los *folds* del *5-fold cross-validation* utilizado para entrenar la red en la clasificación de cáncer de pulmón frente a tos crónica que no es EPOC.

| | # Ficheros | Pacientes | |
|---|---|---|---|
| | | Cáncer de pulmón | no EPOC |
| ***Fold* 0** | 1825 | pal06 | edi10, gla04 |
| ***Fold* 1** | 1700 | pal08, gla15 | pal18, edi05, edi08 |
| ***Fold* 2** | 2260 | gla05, gla11 | pal02, pal10, edi01, edi02, edi06, gla01 |

Tabla 6.22: Descripción de los conjuntos de test en los *folds* del *3-fold cross-validation* utilizado para entrenar la red en la clasificación de cáncer de pulmón frente a tos crónica que no es EPOC.

### 6.4.3. Diagnóstico

En esta sección hablaremos sobre el sistema de diagnóstico empleado para determinar la enfermedad del paciente. Este método de diagnóstico es muy intuitivo y basado en la lógica, y se ha realizado sólo cuando los pacientes estaban en el conjunto de test. Consiste en calcular el tanto por ciento de clips de audio con tos que nuestra red neuronal asigna a cada clase y la clase que obtiene un tanto por ciento superior es el diagnóstico obtenido por el paciente. Por lo tanto el diagnóstico *a priori* será más preciso si tenemos una gran cantidad de clips de audio con tos del paciente.

# *Capítulo 7*

# RESULTADOS Y DISCUSIÓN

En esta sección discutiremos los resultados que hemos obtenido realizando los diferentes experimentos, en los que primeramente se ha entrenado una red neuronal convolucional para detectar clips de audios con tos, para después clasificar tipos de tos dependiendo de la enfermedad subyacente.

Para analizar estos resultados utilizaremos dos criterios principales, siendo estos las métricas descritas en la sección 3.1 y la curva ROC que hemos explicado en la sección 3.2. En el caso en el que clasificamos tipos de tos dependiendo de la enfermedad subyacente también analizamos los resultados sobre el acierto que tiene el sistema diagnosticando a cada paciente.

Con el objetivo de garantizar la independencia de los resultados que se obtienen de las particiones de entrenamiento y test, se ha empleado la técnica de la validación cruzada descrita en la sección 3.3, utilizando diez, cinco y tres *folds* diferentes para posteriormente realizar el macro-promediado explicado en la sección 3.4.

## 7.1. Detección de tos

Para la detección de clips de audio con tos frente a clips de audio sin tos hemos realizado experimentos con dos arquitecturas diferentes de redes neuronales convolucionales.

Vamos a comenzar viendo los resultados que obtenemos al entrenar la primera red neuronal convolucional, que hemos visto en la Figura 6.7 y descrito en la sección 6.3.1. Por otro lado la configuración de los *folds* que se ha usado es descrita en la sección 6.4.2.

| | **SEN** | **SPE** | **AUC** | **PPV/Precision** | **NPV** | **ACC** |
|---|---|---|---|---|---|---|
| ***3-Fold*** | 0,86±0,09 | 0,90±0,02 | 0,95±0,02 | 0,90±0,02 | 0,87±0,07 | 0,88±0,04 |
| ***5-Fold*** | 0,85±0,05 | 0,93±0,02 | 0,95±0,01 | 0,92±0,02 | 0,86±0,03 | 0,89±0,02 |
| ***10-Fold*** | 0,86±0,01 | 0,93±0,04 | 0,96±0,03 | 0,92±0,03 | 0,87±0,08 | 0,89±0,05 |

Tabla 7.1: Media±Desviación estándar de las métricas de los *k-folds* de Palencia con la CNN A en la detección de tos.

Lo primero que observamos es que siempre vemos en la Tabla 7.1 que la especificidad es superior al 90 % y la sensibilidad es en torno al 6 % inferior. También se observa en dicha tabla que la AUC es siempre superior al 95 %, lo que indica que es un test muy bueno, además en la Figura 7.1 se presentan las curvas ROC asociadas.

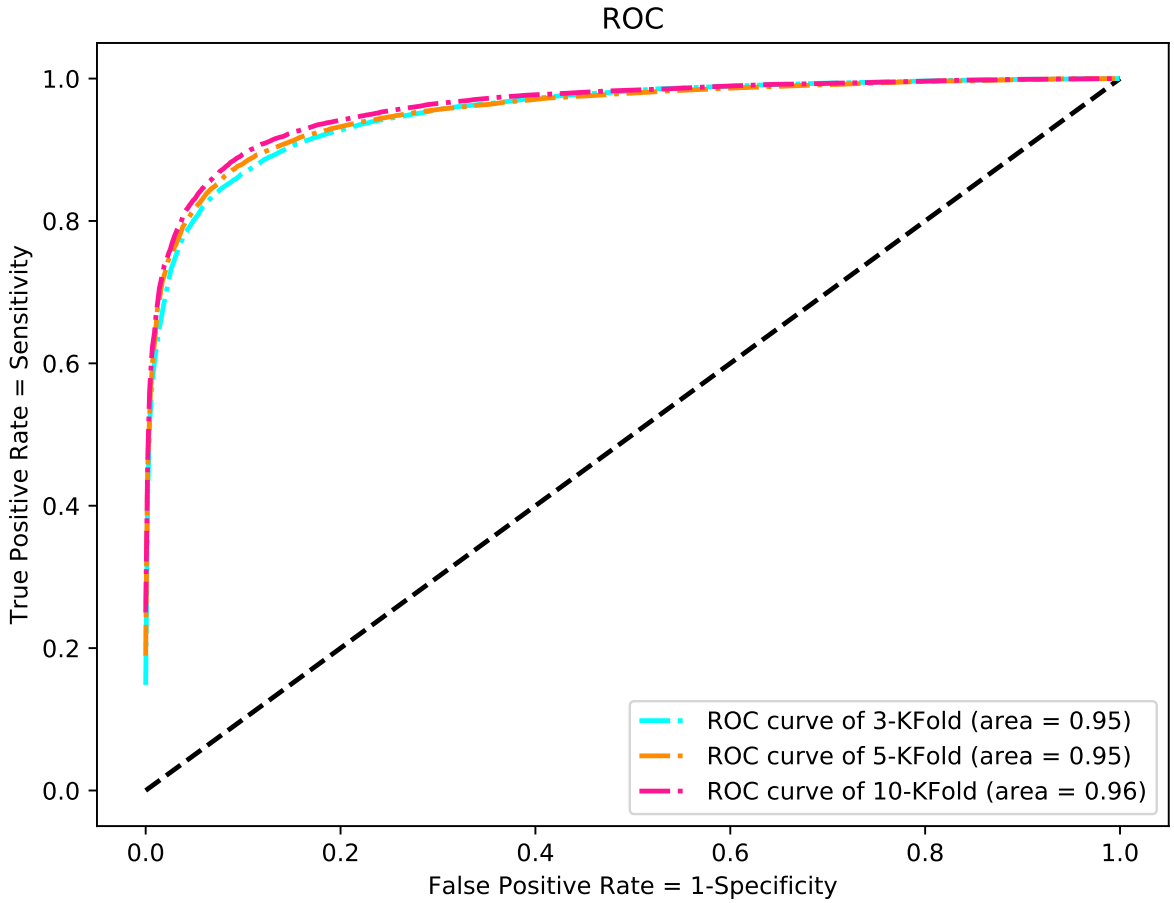



Figura 7.1: Curvas ROC de los *k-folds* de Palencia con la CNN A en la detección de tos.

A continuación vemos los resultados que obtenemos al entrenar la segunda red neuronal convolucional, que hemos visto en la Figura 6.8 y descrito en la sección 6.3.1.

| | **SEN** | **SPE** | **AUC** | **PPV/Precision** | **NPV** | **ACC** |
|---|---|---|---|---|---|---|
| ***3-Fold*** | 0,88±0,07 | 0,86±0,06 | 0,95±0,02 | 0,86±0,04 | 0,89±0,06 | 0,87±0,00 |
| ***5-Fold*** | 0,91±0,03 | 0,87±0,05 | 0,96±0,01 | 0,88±0,04 | 0,90±0,02 | 0,89±0,02 |
| ***10-Fold*** | 0,89±0,08 | 0,90±0,04 | 0,96±0,03 | 0,90±0,04 | 0,89±0,07 | 0,89±0,04 |

Tabla 7.2: Media±Desviación estándar de las métricas de los *k-folds* de Palencia con la CNN B en la detección de tos.

En esta ocasión observamos en la Tabla 7.2 que esta red es mas sensible y menos específica, consiguiendo que en el caso de la validación cruzada *10-Fold* la sensibilidad y especificidad estén prácticamente en el mismo porcentaje. También se observa en dicha tabla que la AUC es siempre superior al 95 % como vimos con la anterior red, lo que indica que también es un test muy bueno, además en la Figura 7.2 se presentan las curvas ROC asociadas.

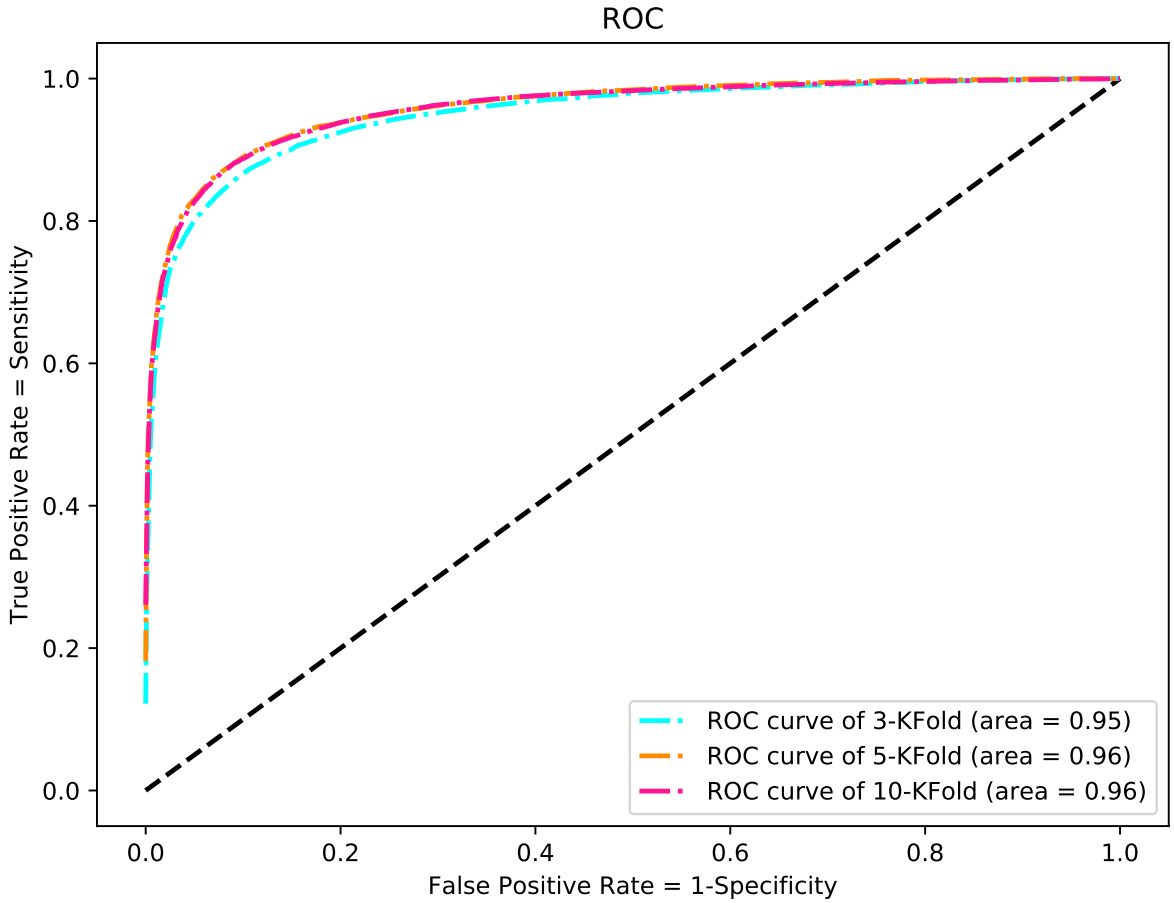



Figura 7.2: Curvas ROC de los *k-folds* de Palencia con la CNN B en la detección de tos.

Los experimentos anteriores se realizaron utilizando la base de datos "Palencia" descrita en la sección 6.1.1, pero dado que esta base de datos no ha sido utilizada en trabajos anteriores, con el objetivo de poder realizar una posterior comparativa hemos testeado también nuestras dos redes neuronales convolucionales con la base de datos "Edimburgo" descrita en la sección 6.1.3.

| | **SEN** | **SPE** | **AUC** | **PPV/Precision** | **NPV** | **ACC** |
|---|---|---|---|---|---|---|
| **CNN A** | 0,75±0,22 | 0,90±0,09 | 0,92±0,07 | 0,87±0,09 | 0,81±0,14 | 0,83±0,12 |
| **CNN B** | 0,96±0,05 | 0,83±0,12 | 0,98±0,02 | 0,85±0,09 | 0,96±0,05 | 0,89±0,06 |

Tabla 7.3: Media±Desviación estándar de las métricas mediante *leave one out* de Edimburgo en la detección de tos.

Antes de nada debemos decir que en este caso no realizamos un *10-fold cross-validation* como anteriormente, sino un *13-fold cross-validation* o lo que es lo mismo en este caso, un *leave one patient out*, esto se debe a que en los trabajos anteriores se realizaba esta metodología. Podemos observar en la Tabla 7.3 que el comportamiento de estas redes neuronales es similar a lo visto anteriormente, ya que la CNN A es mucho más específica que sensible y la CNN B es más sensible que específica, aunque con esta base de datos las características propias de cada red se potencian mucho más.

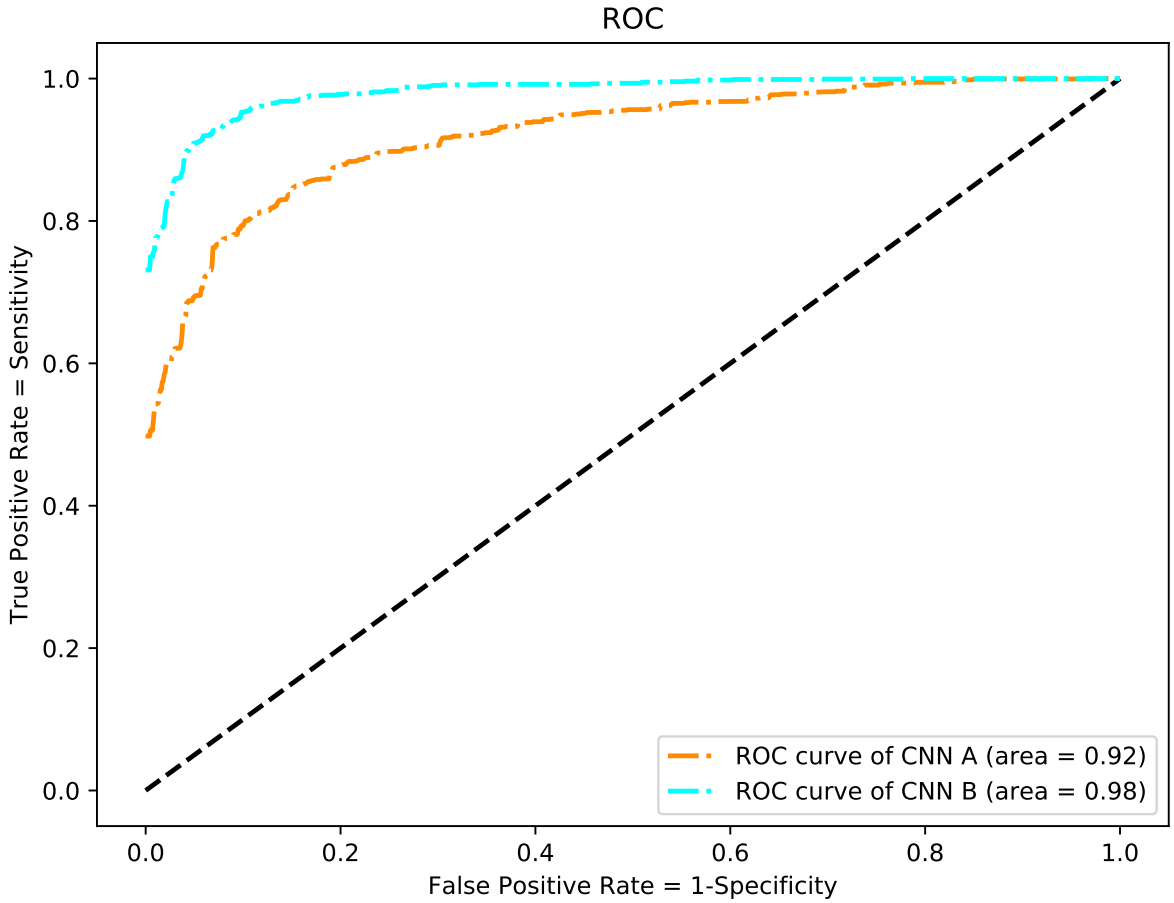



Figura 7.3: Curvas ROC mediante *leave one out* de Edimburgo en la detección de tos.

También vemos que la AUC en el caso de la CNN A denota que es un test muy bueno pero no tan bueno como el que obteníamos con la base de datos "Palencia" y en el caso de la CNN B el AUC denota que es un test excelente, superando lo obtenido con la base de datos "Palencia". Las curvas ROC asociadas a estas AUC se pueden ver en la Figura 7.3.

## 7.2. Clasificación de tipos de tos según enfermedades

Con respecto al problema de clasificación de clips de audio con tos según la enfermedad subyacente debemos decir que el problema es notablemente más complejo, debido a que aquí todos los clips de audio contienen toses y por lo tanto su similitud es mucho mayor al caso anterior, por este motivo se tuvo que aumentar la complejidad de la red neuronal, siendo la red neuronal utilizada en este caso la red neuronal convolucional descrita en la sección 6.3.1.

Además debido a que requerimos de una mayor cantidad de datos se han utilizado las tres bases de datos de pacientes que hemos descrito en la sección 6.1. Para el caso en el que un paciente pertenezca a la base de datos "Palencia", su identificación será palXX, siendo XX el ID asignado en la Tabla 6.1. Si un paciente pertenece a la base de datos "Edimburgo", su identificación será ediXX, siendo XX el ID asignado en la Tabla 6.3. Por último, si un paciente pertenece a la base de datos "Glasgow", su identificación será glaXX, siendo XX el ID asignado en la Tabla 6.2.

Por otro lado la configuración de los *folds* que se han usado para obtener los resultados descritos a continuación, se ha descrito en la sección 6.4.2. Empezamos con los resultados que nos arroja realizar el *5-fold cross-validation*,

cuyos *folds* se describen en la Tabla 6.8.

| | SEN | SPE | AUC | PPV | NPV | ACC |
|---|---|---|---|---|---|---|
| **Acute** | 0,25±0,36 | 0,82±0,30 | 0,77±0,17 | 0,28±0,09 | 0,68±0,22 | 0,57±0,19 |
| **COPD** | 0,48±0,47 | 0,58±0,43 | 0,52±0,27 | 0,21±0,15 | 0,86±0,13 | 0,53±0,25 |
| **Lung cancer** | 0,46±0,48 | 0,65±0,35 | 0,69±0,19 | 0,41±0,36 | 0,86±0,11 | 0,62±0,19 |
| **non COPD** | 0,00±0,00 | 1,00±0,00 | 0,40±0,32 | 0,05±0,07 | 0,77±0,07 | 0,77±0,07 |
| **macro avg** | 0,30±0,07 | 0,76±0,02 | 0,60±0,15 | 0,25±0,04 | 0,79±0,03 | 0,62±0,05 |
| **weighted avg** | 0,81±0,17 | 0,33±0,27 | 0,61±0,28 | 0,33±0,13 | 0,93±0,07 | 0,43±0,19 |
| **micro avg** | 0,25±0,11 | 0,75±0,04 | 0,51±0,11 | 0,25±0,11 | 0,75±0,04 | 0,62±0,05 |

Tabla 7.4: Media±Desviación estándar de las métricas del *5-fold cross-validation* en la clasificación de enfermedades.

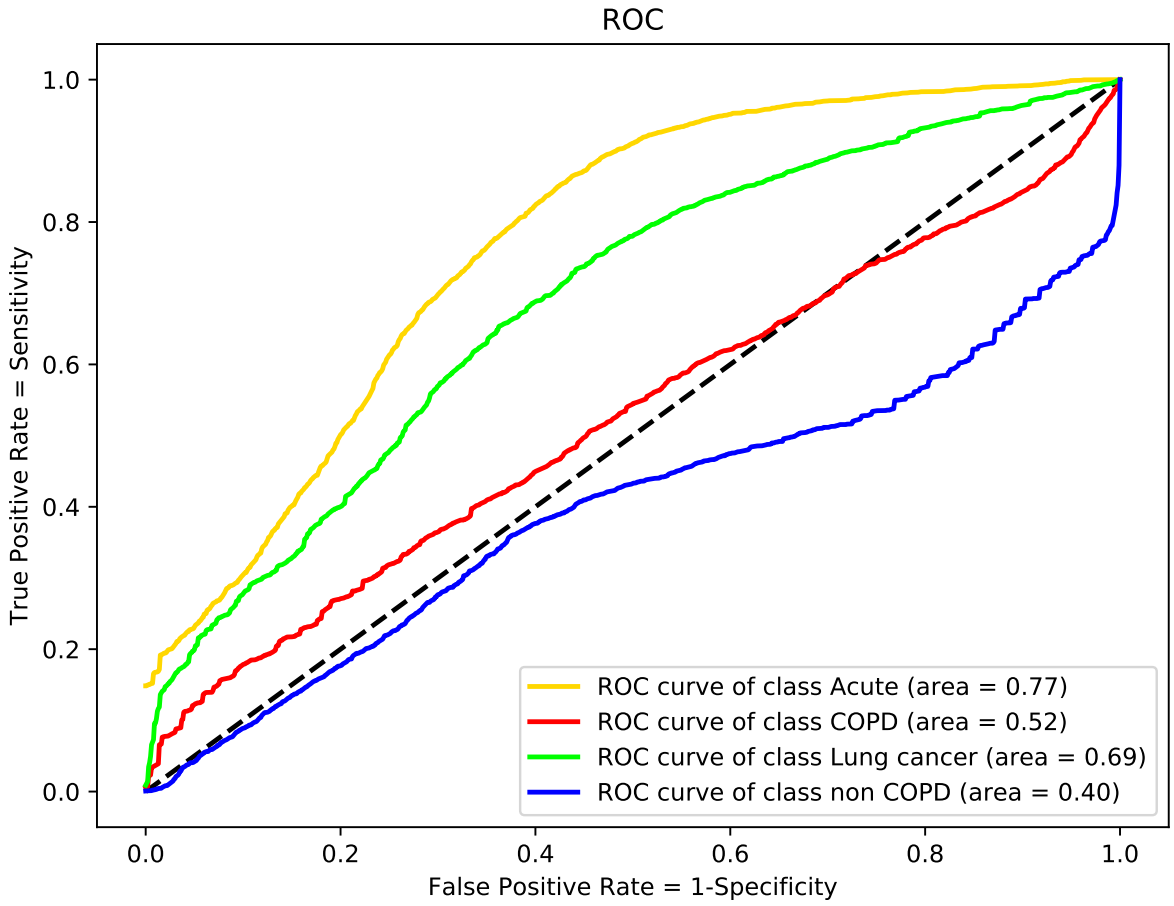



Figura 7.4: Curvas ROC del *5-fold cross-validation* en la clasificación de enfermedades.

Podemos observar con que clase nuestra red neuronal se comporta mejor de manera general observando la curva ROC de la Figura 7.4, lo cual nos indica que la clase que mejor es capaz de clasificar es la de toses agudas (*acute*). También debemos tener en cuenta la Tabla 7.4 para saber que tan bueno es dicho comportamiento vemos

que la sensibilidad es muy baja para todas las clases y la especificidad es la que hace que el AUC aumente, de tal manera que la clase de tos aguda indique que pueda ser un buen test.

| Sujeto | Ficheros | Enfermedad | Porcentaje de ficheros de toses clasificadas | | | | Diagnóstico |
|---|---|---|---|---|---|---|---|
| | | | Acute | COPD | Lung cancer | non COPD | |
| **pal06** | 921 | Lung cancer | 0 | 99,8914 | 0,1086 | 0 | COPD |
| **pal12** | 901 | COPD | 0 | 100 | 0 | 0 | COPD |
| **pal13** | 1579 | Acute | 0 | 100 | 0 | 0 | COPD |
| **edi10** | 110 | non COPD | 0 | 100 | 0 | 0 | COPD |
| **gla04** | 794 | non COPD | 0 | 100 | 0 | 0 | COPD |
| **pal04** | 601 | Acute | 76,0399 | 15,4742 | 7,8203 | 0,6656 | Acute |
| **pal05** | 766 | COPD | 43,6031 | 21,5405 | 34,8564 | 0 | Acute |
| **pal08** | 809 | Lung cancer | 84,796 | 5,3152 | 9,2707 | 0,618 | Acute |
| **pal18** | 697 | non COPD | 86,944 | 3,8737 | 9,0387 | 0,1435 | Acute |
| **edi05** | 100 | non COPD | 1 | 81 | 18 | 0 | COPD |
| **pal10** | 508 | non COPD | 87,5984 | 4,7244 | 7,6772 | 0 | Acute |
| **pal14** | 504 | Acute | 50,5952 | 32,5397 | 16,2698 | 0,5952 | Acute |
| **edi01** | 74 | non COPD | 21,6216 | 37,8378 | 40,5405 | 0 | Lung cancer |
| **edi11** | 70 | COPD | 8,5714 | 68,5714 | 22,8571 | 0 | COPD |
| **gla01** | 205 | non COPD | 2,439 | 16,5854 | 80,9756 | 0 | Lung cancer |
| **gla03** | 535 | COPD | 1,4953 | 12,3364 | 86,1682 | 0 | Lung cancer |
| **gla05** | 784 | Lung cancer | 0 | 4,4643 | 95,5357 | 0 | Lung cancer |
| **pal02** | 162 | non COPD | 0 | 7,4074 | 92,5926 | 0 | Lung cancer |
| **pal03** | 352 | Acute | 0 | 19,6023 | 80,3977 | 0 | Lung cancer |
| **edi02** | 96 | non COPD | 0 | 0 | 100 | 0 | Lung cancer |
| **edi03** | 103 | COPD | 0 | 1,9417 | 98,0583 | 0 | Lung cancer |
| **edi04** | 86 | COPD | 0 | 0 | 100 | 0 | Lung cancer |
| **edi06** | 97 | non COPD | 0 | 0 | 100 | 0 | Lung cancer |
| **edi09** | 134 | COPD | 0 | 0 | 100 | 0 | Lung cancer |
| **gla11** | 334 | Lung cancer | 0 | 1,1976 | 98,8024 | 0 | Lung cancer |
| **pal00** | 265 | Acute | 0 | 70,9434 | 29,0566 | 0 | COPD |
| **edi07** | 32 | COPD | 0 | 100 | 0 | 0 | COPD |
| **edi08** | 50 | non COPD | 0 | 100 | 0 | 0 | COPD |
| **gla09** | 9 | COPD | 0 | 88,8889 | 11,1111 | 0 | COPD |
| **gla15** | 44 | Lung cancer | 0 | 75 | 25 | 0 | COPD |

Tabla 7.5: Diagnósticos del *5-fold cross-validation* en la clasificación de enfermedades. Los diagnósticos correctos se han presentado con fondo verde, siendo este diagnóstico la clase que más ha sido detectada en las toses del paciente.

Para este caso vemos como ha diagnosticado nuestra red a cada uno de los pacientes, viendo que solamente ha diagnosticado correctamente 8 de 30 pacientes, lo cual nos ofrece una tasa de acierto en el diagnóstico del 26,67 %.

Continuamos con los resultados que nos arroja realizar el *3-fold cross-validation*, cuyos *folds* se describen en la Tabla 6.9.

| | **SEN** | **SPE** | **AUC** | **PPV** | **NPV** | **ACC** |
|---|---|---|---|---|---|---|
| **Acute** | 0,13±0,05 | 0,68±0,06 | 0,41±0,06 | 0,13±0,04 | 0,67±0,11 | 0,53±0,10 |
| **COPD** | 0,53±0,36 | 0,54±0,18 | 0,52±0,11 | 0,24±0,07 | 0,83±0,11 | 0,54±0,06 |
| **Lung cancer** | 0,21±0,36 | 0,90±0,12 | 0,55±0,23 | 0,27±0,35 | 0,79±0,05 | 0,74±0,01 |
| **non COPD** | 0,08±0,14 | 0,85±0,25 | 0,53±0,15 | 0,13±0,01 | 0,73±0,02 | 0,65±0,14 |
| **macro avg** | 0,24±0,08 | 0,75±0,03 | 0,50±0,08 | 0,12±0,01 | 0,76±0,04 | 0,62±0,04 |
| **weighted avg** | 0,44±0,25 | 0,59±0,13 | 0,51±0,08 | 0,15±0,01 | 0,79±0,09 | 0,55±0,06 |
| **micro avg** | 0,23±0,08 | 0,74±0,03 | 0,49±0,04 | 0,23±0,08 | 0,74±0,03 | 0,62±0,04 |

Tabla 7.6: Media±Desviación estándar de las métricas del *3-fold cross-validation* en la clasificación de enfermedades.

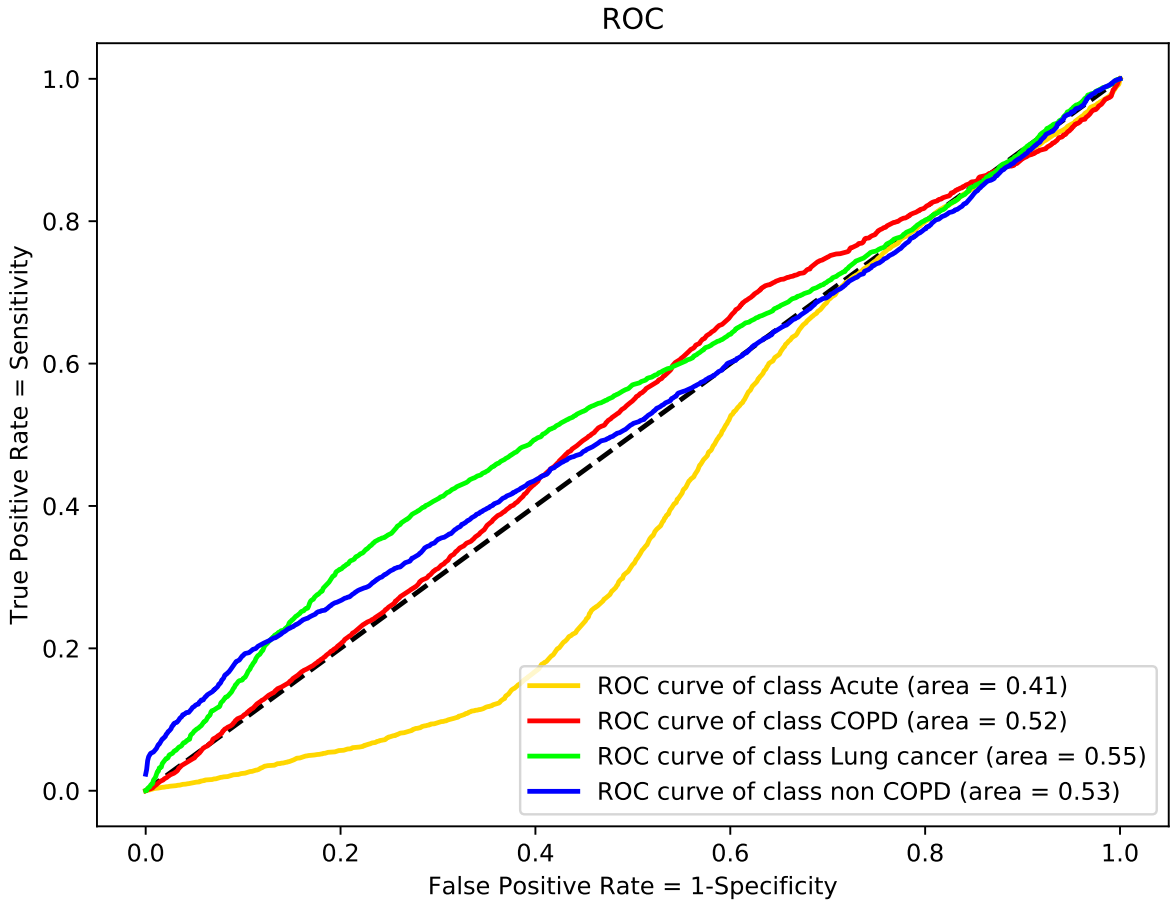



Figura 7.5: Curvas ROC del *3-fold cross-validation* en la clasificación de enfermedades.

En este caso la curva ROC de la Figura 7.5 nos indica que nuestra red neuronal se comporta mejor generalmente con la clase de cáncer de pulmón (*lung cancer*). También observamos la Tabla 7.6 donde se corrobora

que el comportamiento de nuestra red neuronal es notablemente peor que en el caso en el que realizabamos un *5-fold cross-validation*, seguramente debido a que en los entrenamientos alimentamos con menos datos nuestra red neuronal.

| Sujeto | Ficheros | Enfermedad | Porcentaje de ficheros de toses clasificadas | | | | Diagnóstico |
|---|---|---|---|---|---|---|---|
| | | | Acute | COPD | Lung cancer | non COPD | |
| **pal06** | 921 | Lung cancer | 65,798 | 22,0413 | 0,4343 | 11,7264 | Acute |
| **pal12** | 901 | COPD | 41,9534 | 26,5261 | 1,7758 | 29,7447 | Acute |
| **pal13** | 1579 | Acute | 10,4497 | 18,4927 | 0,1267 | 70,931 | non COPD |
| **edi10** | 110 | non COPD | 0,9091 | 84,5455 | 5,4545 | 9,0909 | COPD |
| **gla04** | 794 | non COPD | 0,3778 | 50,8816 | 22,67 | 26,0705 | COPD |
| **pal00** | 265 | Acute | 29,434 | 70,566 | 0 | 0 | COPD |
| **pal04** | 601 | Acute | 0,8319 | 99,1681 | 0 | 0 | COPD |
| **pal05** | 766 | COPD | 2,4804 | 97,5196 | 0 | 0 | COPD |
| **pal08** | 809 | Lung cancer | 39,1842 | 60,8158 | 0 | 0 | COPD |
| **pal18** | 697 | non COPD | 63,1277 | 36,8723 | 0 | 0 | Acute |
| **edi05** | 100 | non COPD | 32 | 68 | 0 | 0 | COPD |
| **edi07** | 32 | COPD | 84,375 | 15,625 | 0 | 0 | Acute |
| **edi08** | 50 | non COPD | 36 | 64 | 0 | 0 | COPD |
| **gla09** | 9 | COPD | 0 | 100 | 0 | 0 | COPD |
| **gla15** | 44 | Lung cancer | 54,5455 | 45,4545 | 0 | 0 | Acute |
| **pal02** | 162 | non COPD | 28,3951 | 44,4444 | 27,1605 | 0 | COPD |
| **pal03** | 352 | Acute | 26,4205 | 57,1023 | 16,4773 | 0 | COPD |
| **pal10** | 508 | non COPD | 54,1339 | 41,1417 | 4,7244 | 0 | Acute |
| **pal14** | 504 | Acute | 13,6905 | 83,9286 | 2,381 | 0 | COPD |
| **edi01** | 74 | non COPD | 64,8649 | 4,0541 | 31,0811 | 0 | Acute |
| **edi02** | 96 | non COPD | 32,2917 | 2,0833 | 65,625 | 0 | Lung cancer |
| **edi03** | 103 | COPD | 23,301 | 23,301 | 53,3981 | 0 | Lung cancer |
| **edi04** | 86 | COPD | 4,6512 | 18,6047 | 76,7442 | 0 | Lung cancer |
| **edi06** | 97 | non COPD | 10,3093 | 9,2784 | 80,4124 | 0 | Lung cancer |
| **edi09** | 134 | COPD | 46,2687 | 12,6866 | 41,0448 | 0 | Acute |
| **edi11** | 70 | COPD | 40 | 5,7143 | 54,2857 | 0 | Lung cancer |
| **gla01** | 205 | non COPD | 34,6341 | 35,122 | 30,2439 | 0 | COPD |
| **gla03** | 535 | COPD | 29,1589 | 55,3271 | 15,514 | 0 | COPD |
| **gla05** | 784 | Lung cancer | 2,9337 | 23,5969 | 73,4694 | 0 | Lung cancer |
| **gla11** | 334 | Lung cancer | 4,7904 | 55,3892 | 39,8204 | 0 | COPD |

Tabla 7.7: Diagnósticos del *3-fold cross-validation* en la clasificación de enfermedades. Los diagnósticos correctos se han presentado con fondo verde, siendo este diagnóstico la clase que más ha sido detectada en las toses del paciente.

Finalmente vemos como ha diagnosticado nuestra red a cada uno de los pacientes, viendo que solamente ha

diagnosticado correctamente 4 de 30 pacientes, lo cual nos ofrece una tasa de acierto en el diagnóstico del 13,33 %. Estos resultados nos hacen pensar que tenemos pocos datos para alimentar nuestra red neuronal y que esta sea capaz de diferenciar cuatro clases diferentes, pero también vemos que hay clases que diferencia del resto de clases con mayor facilidad. Con el objetivo de ver que clases son más diferenciables para nuestra red neuronal utilizando la misma metodología que en este caso, vamos a entrenar nuestra red neuronal solo con dos clases realizando todas las posibles combinaciones.

### 7.2.1. Tos aguda frente a EPOC

Para la clasificación de tos aguda frente a EPOC, es el caso en el que más datos tenemos y que por tanto se espera un mayor rendimiento de la red neuronal. Con el objetivo de sacar el mayor rendimiento a nuestros datos, vamos a realizar un *10-fold cross-validation*, siendo la configuración de los *folds* la descrita en la Tabla 6.10.

| | | | Clasificación ( %) | | |
|---|---|---|---|---|---|
| **Sujeto** | **Ficheros** | **Enfermedad** | **Acute** | **COPD** | **Diagnóstico** |
| **pal07** | 2401 | Acute | 83,1737 | 16,8263 | Acute |
| **pal17** | 2033 | COPD | 96,4092 | 3,5908 | Acute |
| **pal13** | 1579 | Acute | 88,9804 | 11,0196 | Acute |
| **gla02** | 1191 | COPD | 100 | 0 | Acute |
| **pal05** | 766 | COPD | 88,5117 | 11,4883 | Acute |
| **pal04** | 601 | Acute | 83,3611 | 16,6389 | Acute |
| **edi04** | 86 | COPD | 0 | 100 | COPD |
| **edi03** | 103 | COPD | 0,9709 | 99,0291 | COPD |
| **edi11** | 70 | COPD | 0 | 100 | COPD |
| **gla03** | 535 | COPD | 1,6822 | 98,3178 | COPD |
| **edi07** | 32 | COPD | 0 | 100 | COPD |
| **edi09** | 134 | COPD | 1,4925 | 98,5075 | COPD |
| **gla09** | 9 | COPD | 0 | 100 | COPD |
| **gla10** | 509 | COPD | 0 | 100 | COPD |
| **pal00** | 265 | Acute | 72,0755 | 27,9245 | Acute |
| **pal14** | 504 | Acute | 30,3571 | 69,6429 | COPD |
| **pal03** | 352 | Acute | 84,0909 | 15,9091 | Acute |
| **gla07** | 493 | COPD | 0 | 100 | COPD |

Tabla 7.8: Diagnósticos del *10-fold cross-validation* en la clasificación de tos aguda frente a EPOC. Los diagnósticos correctos se han presentado con fondo verde, siendo este diagnóstico la clase que más ha sido detectada en las toses del paciente.

En este caso hemos obtenido 14 de 18 diagnósticos correctos como se puede comprobar en la Tabla 7.8, lo cual ofrece un acierto del 77,78 %. Pero no tenemos suficientes pacientes como para obtener las estadísticas de como se esta comportando la red neuronal, ya que en todos los *folds* no hay pacientes test de todas las clases. Por lo tanto vamos a reducir el número de *folds* para poder obtener dichas estadísticas. En esta ocasión realizaremos un *5-fold cross-validation*, siendo la configuración de los *folds* la descrita en la Tabla 6.11.

| | SEN | SPE | AUC | PPV | NPV | ACC |
|---|---|---|---|---|---|---|
| **Acute** | 0,70±0,22 | 0,42±0,28 | 0,54±0,34 | 0,57±0,18 | 0,55±0,36 | 0,56±0,23 |
| **COPD** | 0,42±0,28 | 0,70±0,22 | 0,54±0,34 | 0,55±0,36 | 0,57±0,18 | 0,56±0,23 |
| **macro avg** | 0,56±0,23 | 0,56±0,23 | 0,55±0,34 | 0,56±0,26 | 0,56±0,26 | 0,56±0,23 |
| **weighted avg** | 0,61±0,22 | 0,51±0,25 | 0,54±0,34 | 0,56±0,23 | 0,56±0,30 | 0,56±0,23 |
| **micro avg** | 0,56±0,23 | 0,56±0,23 | 0,56±0,30 | 0,56±0,23 | 0,56±0,23 | 0,56±0,23 |

Tabla 7.9: Media±Desviación estándar de las métricas del *5-fold cross-validation* en la clasificación de tos aguda frente a EPOC.

Se puede observar en la Tabla 7.9 como al ser solo dos clases los valores de sensibilidad y PPV de una clase son los valores de la especificidad y el NPV de la otra clase y viceversa. Además también observamos que el AUC y el *accuracy* son iguales en ambas clases. Sabiendo esto vemos que la clase tos aguda es mas sensible mientras que la clase EPOC es más específica.

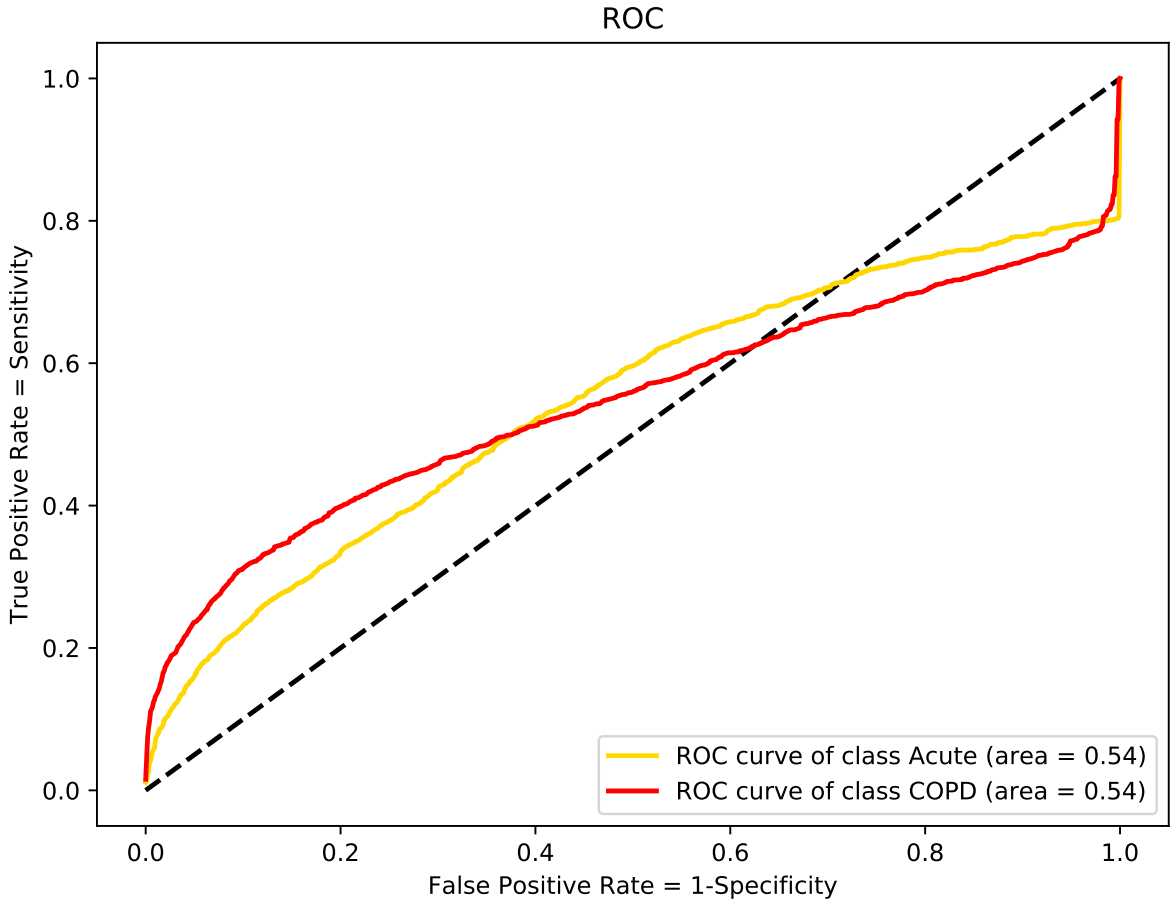



Figura 7.6: Curvas ROC del *5-fold cross-validation* en la clasificación de tos aguda frente a EPOC.

Por otro lado el AUC nos indica que es un test malo siendo las curvas ROC asociadas a dichas AUC, las de la Figura 7.6.

| Sujeto | Ficheros | Enfermedad | Clasificación ( %) | | Diagnóstico |
|---|---|---|---|---|---|
| | | | Acute | COPD | |
| **pal07** | 2401 | Acute | 86,0891 | 13,9109 | Acute |
| **pal17** | 2033 | COPD | 27,152 | 72,848 | COPD |
| **gla07** | 493 | COPD | 0 | 100 | COPD |
| **pal05** | 766 | COPD | 95,8225 | 4,1775 | Acute |
| **pal12** | 901 | COPD | 24,1953 | 75,8047 | COPD |
| **pal13** | 1579 | Acute | 92,0836 | 7,9164 | Acute |
| **pal04** | 601 | Acute | 76,2063 | 23,7937 | Acute |
| **gla03** | 535 | COPD | 53,0841 | 46,9159 | Acute |
| **pal14** | 504 | Acute | 41,4683 | 58,5317 | COPD |
| **gla10** | 509 | COPD | 100 | 0 | Acute |
| **pal00** | 265 | Acute | 64,1509 | 35,8491 | Acute |
| **pal03** | 352 | Acute | 44,3182 | 55,6818 | COPD |
| **edi03** | 103 | COPD | 37,8641 | 62,1359 | COPD |
| **edi04** | 86 | COPD | 66,2791 | 33,7209 | Acute |
| **edi07** | 32 | COPD | 84,375 | 15,625 | Acute |
| **edi09** | 134 | COPD | 56,7164 | 43,2836 | Acute |
| **edi11** | 70 | COPD | 68,5714 | 31,4286 | Acute |
| **gla09** | 9 | COPD | 11,1111 | 88,8889 | COPD |

Tabla 7.10: Diagnósticos del *5-fold cross-validation* en la clasificación de tos aguda frente a EPOC. Los diagnósticos correctos se han presentado con fondo verde, siendo este diagnóstico la clase que más ha sido detectada en las toses del paciente.

Observando los diagnósticos vemos que hemos obtenido 9 de 18 diagnósticos correctos, como se puede comprobar en la Tabla 7.10, lo cual ofrece un acierto del 50 % y por lo tanto baja bastante el acierto, debido a que en ciertos *folds* la red no ha convergido tan bien como debiera. Ahora vamos a reducir más el número de *folds*, en esta ocasión realizaremos un *3-fold cross-validation*, siendo la configuración de los *folds* la descrita en la Tabla 6.12.

| | SEN | SPE | AUC | PPV | NPV | ACC |
|---|---|---|---|---|---|---|
| **Acute** | 0,76±0,20 | 0,62±0,16 | 0,73±0,10 | 0,68±0,06 | 0,75±0,16 | 0,69±0,08 |
| **COPD** | 0,62±0,16 | 0,76±0,20 | 0,73±0,10 | 0,75±0,16 | 0,68±0,06 | 0,69±0,08 |
| **macro avg** | 0,69±0,08 | 0,69±0,08 | 0,73±0,10 | 0,71±0,08 | 0,71±0,08 | 0,69±0,08 |
| **weighted avg** | 0,74±0,09 | 0,65±0,11 | 0,73±0,10 | 0,69±0,08 | 0,74±0,09 | 0,69±0,08 |
| **micro avg** | 0,69±0,08 | 0,69±0,08 | 0,73±0,10 | 0,69±0,08 | 0,69±0,08 | 0,69±0,08 |

Tabla 7.11: Media±Desviación estándar de las métricas del *3-fold cross-validation* en la clasificación de tos aguda frente a EPOC.

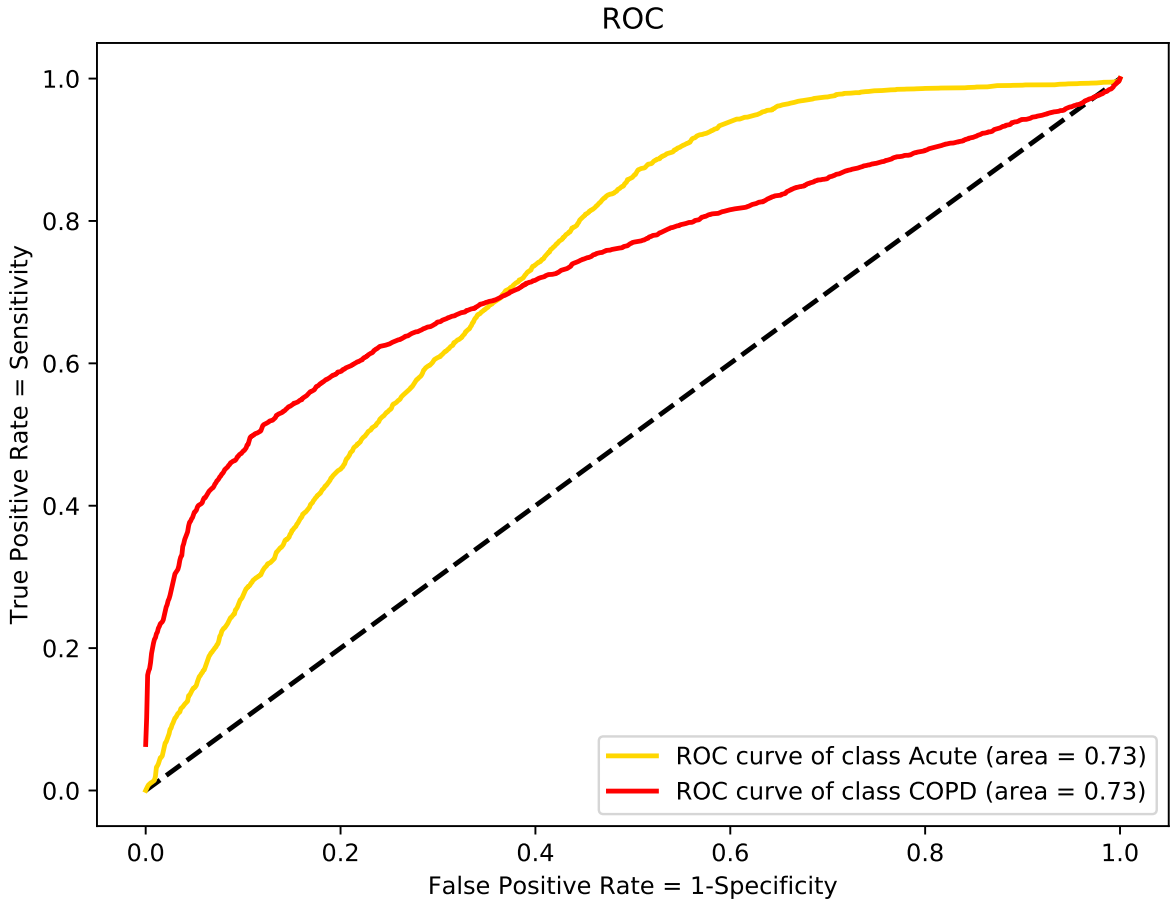



Figura 7.7: Curvas ROC del *3-fold cross-validation* en la clasificación de tos aguda frente a EPOC.

En este caso vemos en la Tabla 7.11 que también la clase tos aguda es mas sensible y la clase EPOC es mas específica, pero aumentan bastante los valores con respecto al experimento anterior. Fijándonos en el AUC vemos que es un test regular cercano a ser un test bueno, además podemos ver las curvas ROC asociadas a dichas AUC en la Figura 7.7.

| Sujeto | Ficheros | Enfermedad | Clasificación ( %) | | Diagnóstico |
|---|---|---|---|---|---|
| | | | Acute | COPD | |
| **pal07** | 2401 | Acute | 82,4656 | 17,5344 | Acute |
| **pal17** | 2033 | COPD | 34,9238 | 65,0762 | COPD |
| **gla07** | 493 | COPD | 0 | 100 | COPD |
| **pal05** | 766 | COPD | 97,5196 | 2,4804 | Acute |
| **pal12** | 901 | COPD | 20,4218 | 79,5782 | COPD |
| **pal13** | 1579 | Acute | 92,9702 | 7,0298 | Acute |
| **pal00** | 265 | Acute | 60,3774 | 39,6226 | Acute |
| **pal03** | 352 | Acute | 53,9773 | 46,0227 | Acute |
| **pal04** | 601 | Acute | 75,0416 | 24,9584 | Acute |
| **pal14** | 504 | Acute | 24,0079 | 75,9921 | COPD |
| **edi03** | 103 | COPD | 29,1262 | 70,8738 | COPD |
| **edi04** | 86 | COPD | 40,6977 | 59,3023 | COPD |

| Sujeto | Ficheros | Enfermedad | Clasificación ( %) | | Diagnóstico |
|---|---|---|---|---|---|
| | | | Acute | COPD | |
| **edi07** | 32 | COPD | 84,375 | 15,625 | Acute |
| **edi09** | 134 | COPD | 41,791 | 58,209 | COPD |
| **edi11** | 70 | COPD | 61,4286 | 38,5714 | Acute |
| **gla03** | 535 | COPD | 45,4206 | 54,5794 | COPD |
| **gla09** | 9 | COPD | 0 | 100 | COPD |
| **gla10** | 509 | COPD | 0,3929 | 99,6071 | COPD |

Tabla 7.12: Diagnósticos del *3-fold cross-validation* en la clasificación de tos aguda frente a EPOC. Los diagnósticos correctos se han presentado con fondo verde, siendo este diagnóstico la clase que más ha sido detectada en las toses del paciente.

Ahora observando los diagnósticos vemos que en este caso hemos obtenido 14 de 18 diagnósticos correctos, como se puede comprobar en la Tabla 7.12, lo cual ofrece un acierto del 77,78 %, este es un resultado bastante interesante, debido a que en este caso los *folds* de la red han entrenado mejor. Esto puede ser debido a que hay pacientes que benefician más a la red que otros.

### 7.2.2. Tos aguda frente a cáncer de pulmón

La segunda combinación de la que vamos a hablar es tos aguda frente a cáncer de pulmón. En este caso vamos a estudiar la clase de la que menos datos tenemos (cáncer de pulmón). Esta clase será nuestro limitante para realizar un entrenamiento balanceado. Con el objetivo de sacar el mayor rendimiento a nuestros datos, vamos a realizar un *5-fold cross-validation*, siendo la configuración de los *folds* la descrita en la Tabla 6.13.

| | SEN | SPE | AUC | PPV | NPV | ACC |
|---|---|---|---|---|---|---|
| **Acute** | 0,59±0,35 | 0,61±0,43 | 0,70±0,26 | 0,73±0,23 | 0,52±0,29 | 0,54±0,21 |
| **Lung cancer** | 0,61±0,43 | 0,59±0,35 | 0,70±0,26 | 0,52±0,29 | 0,73±0,23 | 0,54±0,21 |
| **macro avg** | 0,60±0,15 | 0,60±0,15 | 0,70±0,26 | 0,63±0,16 | 0,63±0,16 | 0,54±0,21 |
| **weighted avg** | 0,80±0,16 | 0,40±0,28 | 0,70±0,26 | 0,54±0,21 | 0,72±0,21 | 0,54±0,21 |
| **micro avg** | 0,54±0,21 | 0,54±0,21 | 0,56±0,25 | 0,54±0,21 | 0,54±0,21 | 0,54±0,21 |

Tabla 7.13: Media±Desviación estándar de las métricas del *5-fold cross-validation* en la clasificación de tos aguda frente a cáncer de pulmón. Los diagnósticos correctos se han presentado con fondo verde, siendo este diagnóstico la clase que más ha sido detectada en las toses del paciente.

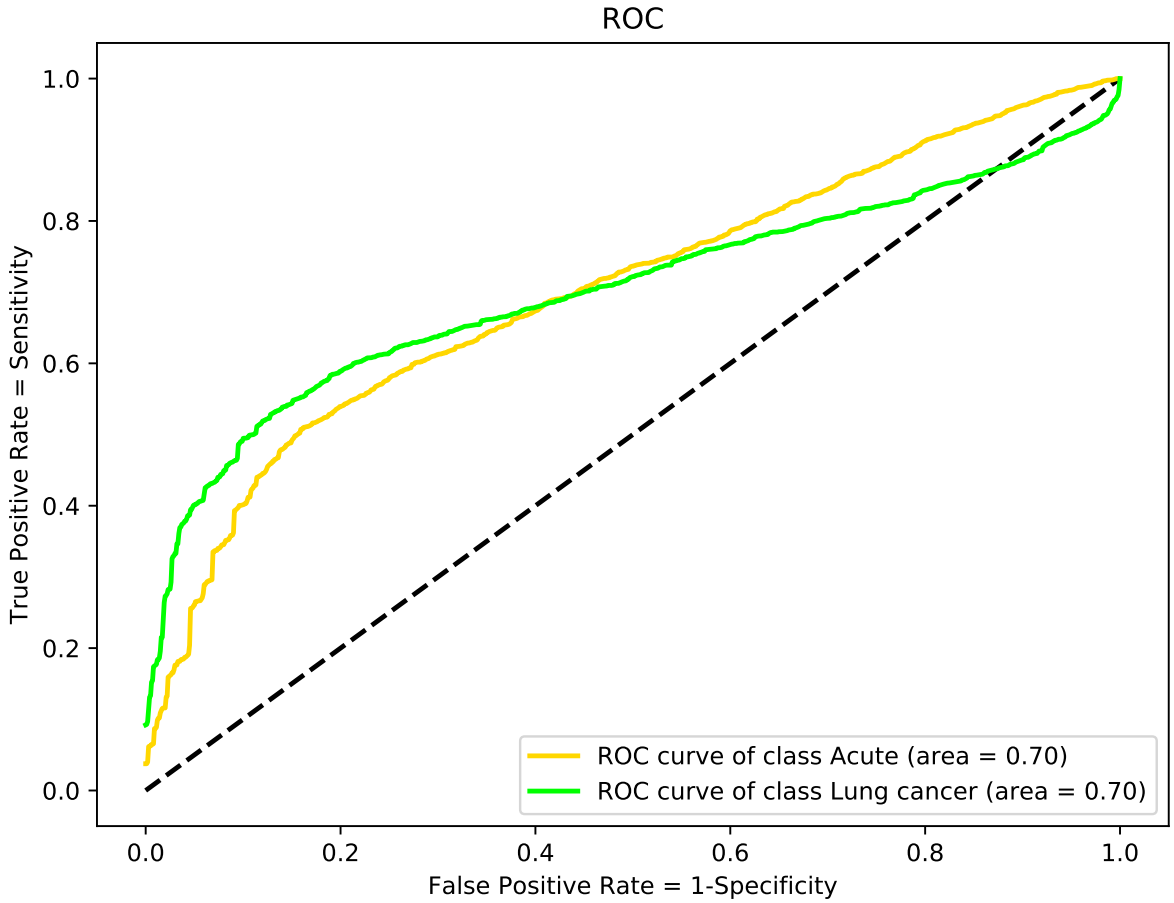



Figura 7.8: Curvas ROC del *5-fold cross-validation* en la clasificación de tos aguda frente a cáncer de pulmón.

Se puede observar en la Tabla 7.13 que la clase cáncer de pulmón es ligeramente más sensible, mientras que la clase tos aguda es ligeramente más específica. Además vemos que es un test regular cercano a ser un test bueno fijándonos en la AUC y en las curvas ROC asociadas que tenemos en la Figura 7.8.

| Sujeto | Ficheros | Enfermedad | Clasificación ( %) | | Diagnóstico |
|---|---|---|---|---|---|
| | | | Acute | Lung cancer | |
| **pal06** | 921 | Lung cancer | 73,6156 | 26,3844 | Acute |
| **pal13** | 1579 | Acute | 63,9645 | 36,0355 | Acute |
| **pal04** | 601 | Acute | 96,3394 | 3,6606 | Acute |
| **pal08** | 809 | Lung cancer | 95,3028 | 4,6972 | Acute |
| **pal14** | 504 | Acute | 88,0952 | 11,9048 | Acute |
| **gla05** | 784 | Lung cancer | 21,301 | 78,699 | Lung cancer |
| **pal03** | 352 | Acute | 32,6705 | 67,3295 | Lung cancer |
| **gla11** | 334 | Lung cancer | 2,0958 | 97,9042 | Lung cancer |
| **pal00** | 265 | Acute | 15,8491 | 84,1509 | Lung cancer |
| **gla15** | 44 | Lung cancer | 2,2727 | 97,7273 | Lung cancer |

Tabla 7.14: Diagnósticos del *5-fold cross-validation* en la clasificación de tos aguda frente a cáncer de pulmón. Los diagnósticos correctos se han presentado con fondo verde, siendo este diagnóstico la clase que más ha sido detectada en las toses del paciente.

Para este caso hemos obtenido 6 de 10 diagnósticos correctos, como vemos en la Tabla 7.14, lo cual ofrece un acierto del 60 %. Esta tasa de acierto es notablemente más baja que en el caso anterior, esto nos reafirma en la hipótesis de que con más datos la red neuronal ofrecería mejores resultados. Ahora vamos a reducir más el número de *folds*, en esta ocasión realizaremos un *3-fold cross-validation*, siendo la configuración de los *folds* la descrita en la Tabla 6.14.

| | **SEN** | **SPE** | **AUC** | **PPV** | **NPV** | **ACC** |
|---|---|---|---|---|---|---|
| **Acute** | 0,71±0,03 | 0,44±0,34 | 0,59±0,24 | 0,62±0,14 | 0,53±0,25 | 0,60±0,17 |
| **Lung cancer** | 0,44±0,34 | 0,71±0,03 | 0,59±0,24 | 0,53±0,25 | 0,62±0,14 | 0,60±0,17 |
| **macro avg** | 0,58±0,18 | 0,58±0,18 | 0,59±0,24 | 0,58±0,18 | 0,58±0,18 | 0,60±0,17 |
| **weighted avg** | 0,65±0,12 | 0,51±0,23 | 0,59±0,24 | 0,60±0,17 | 0,55±0,20 | 0,60±0,17 |
| **micro avg** | 0,60±0,17 | 0,60±0,17 | 0,63±0,21 | 0,60±0,17 | 0,60±0,17 | 0,60±0,17 |

Tabla 7.15: Media±Desviación estándar de las métricas del *3-fold cross-validation* en la clasificación de tos aguda frente a cáncer de pulmón.

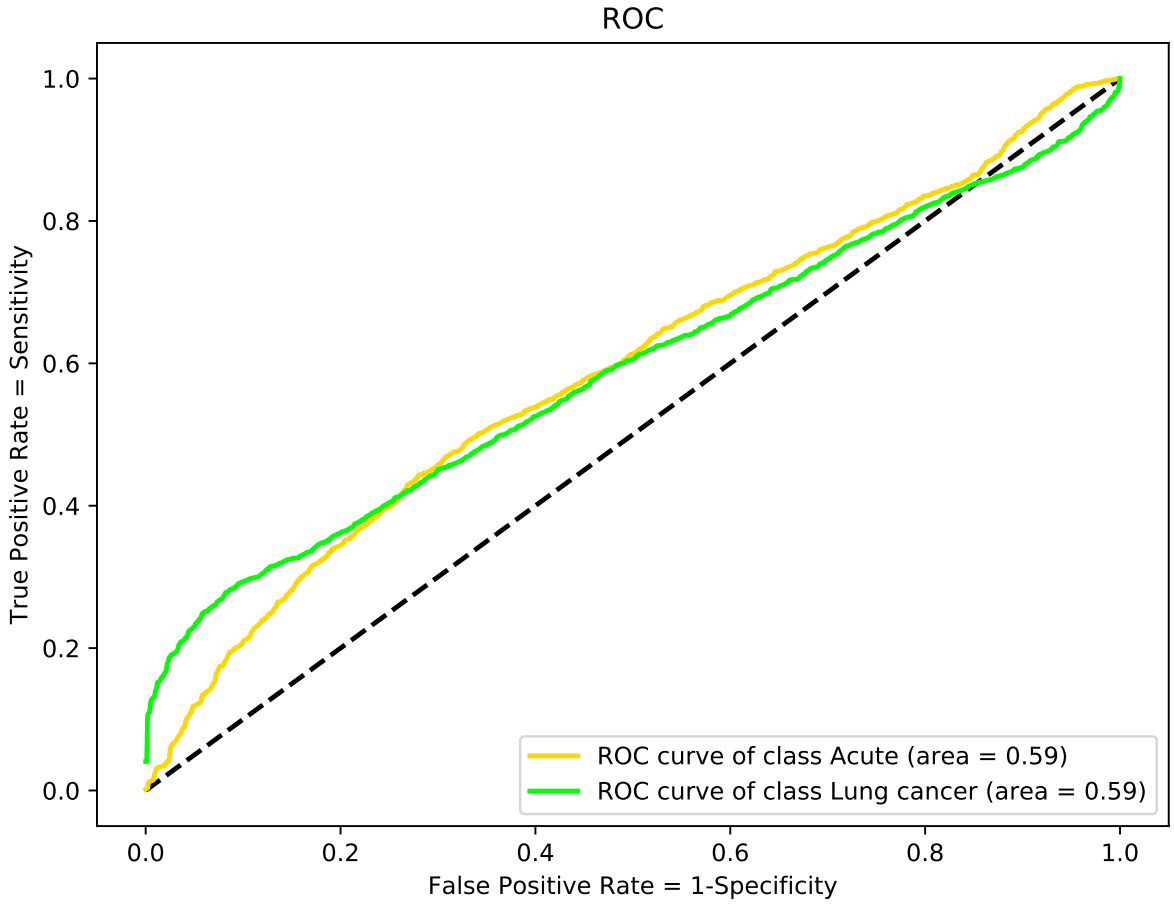



Figura 7.9: Curvas ROC del *3-fold cross-validation* en la clasificación de tos aguda frente a cáncer de pulmón.

Revisando la Tabla 7.15, podemos distinguir que la clase tos aguda es más sensible, mientras que la clase cáncer de pulmón es más específica. Tendiendo en cuenta el AUC observamos que es un test malo cercano a ser un test regular, además podemos examinar las curvas ROC asociadas en la Figura 7.9.

| Sujeto | Ficheros | Enfermedad | Clasificación ( %) | | Diagnóstico |
|---|---|---|---|---|---|
| | | | Acute | Lung cancer | |
| **pal06** | 921 | Lung cancer | 75,8958 | 24,1042 | Acute |
| **pal13** | 1579 | Acute | 72,1976 | 27,8024 | Acute |
| **pal00** | 265 | Acute | 33,2075 | 66,7925 | Lung cancer |
| **pal04** | 601 | Acute | 83,8602 | 16,1398 | Acute |
| **pal08** | 809 | Lung cancer | 78,6156 | 21,3844 | Acute |
| **gla15** | 44 | Lung cancer | 0 | 100 | Lung cancer |
| **pal03** | 352 | Acute | 69,6023 | 30,3977 | Acute |
| **pal14** | 504 | Acute | 76,1905 | 23,8095 | Acute |
| **gla05** | 784 | Lung cancer | 15,1786 | 84,8214 | Lung cancer |
| **gla11** | 334 | Lung cancer | 21,5569 | 78,4431 | Lung cancer |

Tabla 7.16: Diagnósticos del *3-fold cross-validation* en la clasificación de tos aguda frente a cáncer de pulmón. Los diagnósticos correctos se han presentado con fondo verde, siendo este diagnóstico la clase que más ha sido detectada en las toses del paciente.

Ahora hemos obtenido 7 de 10 diagnósticos correctos, según lo observado en la Tabla 7.16, lo cual ofrece un acierto del 70 %. Esta tasa de acierto es superior a la que ofrecía la red neuronal con *5-fold cross-validation*, lo cual puede ser debido a que los pacientes con los que se ha entrenado la red neuronal en cada *fold* son los que más información aportaban a misma.

### 7.2.3. Tos aguda frente a tos crónica que no es EPOC

La tercera combinación de la que vamos a hablar es tos aguda frente a tos crónica que no es EPOC, en este caso la clase limitante para realizar un entrenamiento balanceado es tos crónica que no es EPOC. Con el objetivo de sacar el mayor rendimiento a nuestros datos, vamos a realizar un *5-fold cross-validation*, siendo la configuración de los *folds* la descrita en la Tabla 6.15

| | SEN | SPE | AUC | PPV | NPV | ACC |
|---|---|---|---|---|---|---|
| **Acute** | 0,67±0,12 | 0,62±0,39 | 0,69±0,31 | 0,73±0,25 | 0,59±0,23 | 0,65±0,17 |
| **non COPD** | 0,62±0,39 | 0,67±0,12 | 0,69±0,31 | 0,59±0,23 | 0,73±0,25 | 0,65±0,17 |
| **macro avg** | 0,65±0,17 | 0,65±0,17 | 0,69±0,31 | 0,66±0,22 | 0,66±0,22 | 0,65±0,17 |
| **weighted avg** | 0,74±0,13 | 0,55±0,23 | 0,69±0,31 | 0,65±0,17 | 0,67±0,28 | 0,65±0,17 |
| **micro avg** | 0,65±0,17 | 0,65±0,17 | 0,69±0,23 | 0,65±0,17 | 0,65±0,17 | 0,65±0,17 |

Tabla 7.17: Media±Desviación estándar de las métricas del *5-fold cross-validation* en la clasificación de tos aguda frente a tos crónica que no es EPOC.

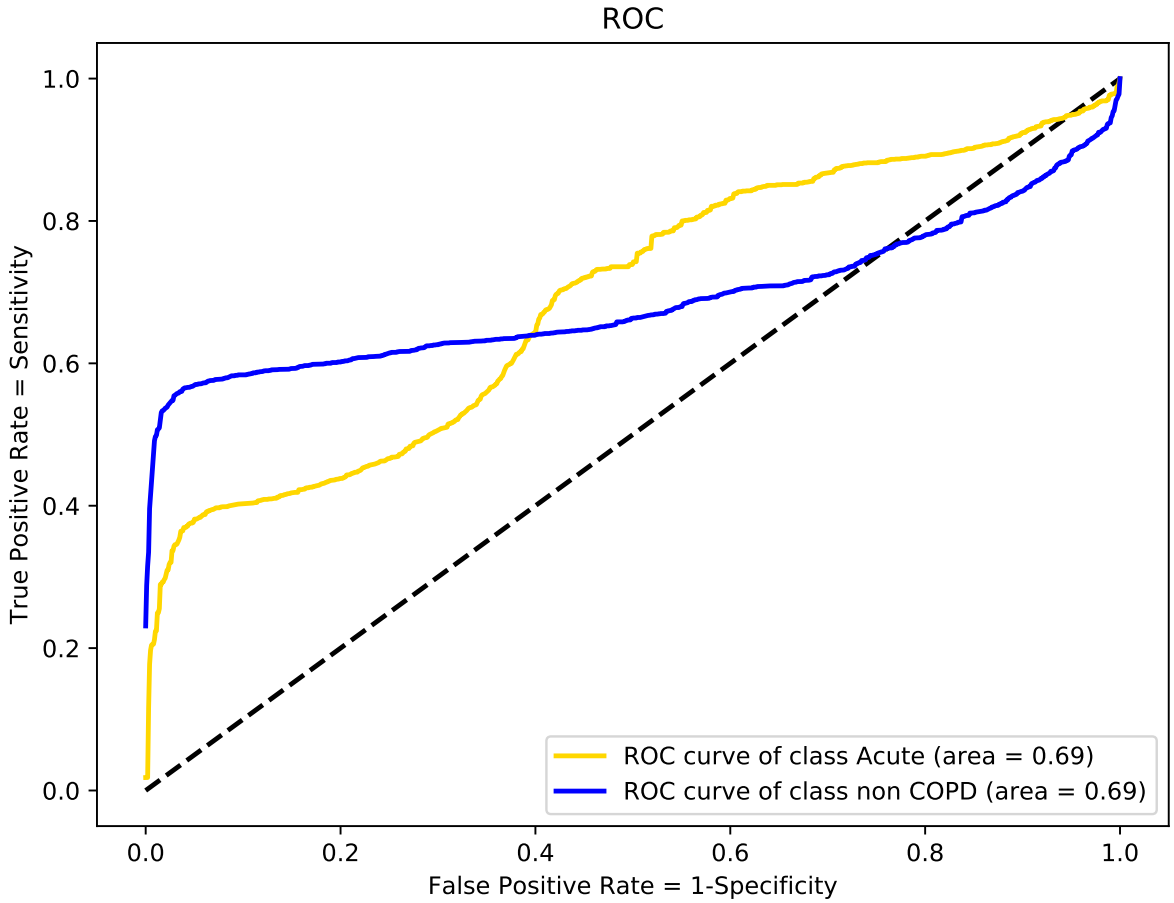



Figura 7.10: Curvas ROC del *5-fold cross-validation* en la clasificación de tos aguda frente a tos crónica que no es EPOC.

Analizando la Tabla 7.17 podemos comprobar que la clase tos aguda es levemente más sensible mientras que la clase tos crónica que no es EPOC es levemente más específica. Además vemos que el AUC señala que es un test regular, pudiendo examinar las curvas ROC asociadas en la Figura 7.10.

| Sujeto | Ficheros | Enfermedad | Clasificación ( %) | | Diagnóstico |
|---|---|---|---|---|---|
| | | | Acute | non COPD | |
| **pal13** | 1579 | Acute | 74,9208 | 25,0792 | Acute |
| **pal18** | 697 | non COPD | 89,3831 | 10,6169 | Acute |
| **edi02** | 96 | non COPD | 0 | 100 | non COPD |
| **gla04** | 794 | non COPD | 0,2519 | 99,7481 | non COPD |
| **pal04** | 601 | Acute | 69,0516 | 30,9484 | Acute |
| **edi08** | 50 | non COPD | 14 | 86 | non COPD |
| **edi10** | 110 | non COPD | 0,9091 | 99,0909 | non COPD |
| **gla08** | 443 | non COPD | 0,4515 | 99,5485 | non COPD |
| **pal10** | 508 | non COPD | 94,0945 | 5,9055 | Acute |
| **pal14** | 504 | Acute | 72,0238 | 27,9762 | Acute |
| **pal03** | 352 | Acute | 45,4545 | 54,5455 | non COPD |
| **gla12** | 341 | non COPD | 0,2933 | 99,7067 | non COPD |
| **pal00** | 265 | Acute | 75,0943 | 24,9057 | Acute |

| Sujeto | Ficheros | Enfermedad | Clasificación ( %) | | Diagnóstico |
|---|---|---|---|---|---|
| | | | Acute | non COPD | |
| **pal02** | 162 | non COPD | 87,6543 | 12,3457 | Acute |
| **edi05** | 100 | non COPD | 0 | 100 | non COPD |

Tabla 7.18: Diagnósticos del *5-fold cross-validation* en la clasificación de tos aguda frente a tos crónica que no es EPOC. Los diagnósticos correctos se han presentado con fondo verde, siendo este diagnóstico la clase que más ha sido detectada en las toses del paciente.

En este caso hemos obtenido 11 de 15 diagnósticos correctos, como se puede comprobar en la Tabla 7.18, lo cual ofrece un acierto del 73,33 %. Esta tasa de acierto es bastante alta e interesante, aunque este resultado puede haberse debido a que la clase tos aguda solo esta en una de las tres bases de datos. Ahora vamos a reducir más el número de *folds*, en esta ocasión realizaremos un *3-fold cross-validation*, siendo la configuración de los *folds* la descrita en la Tabla 6.16.

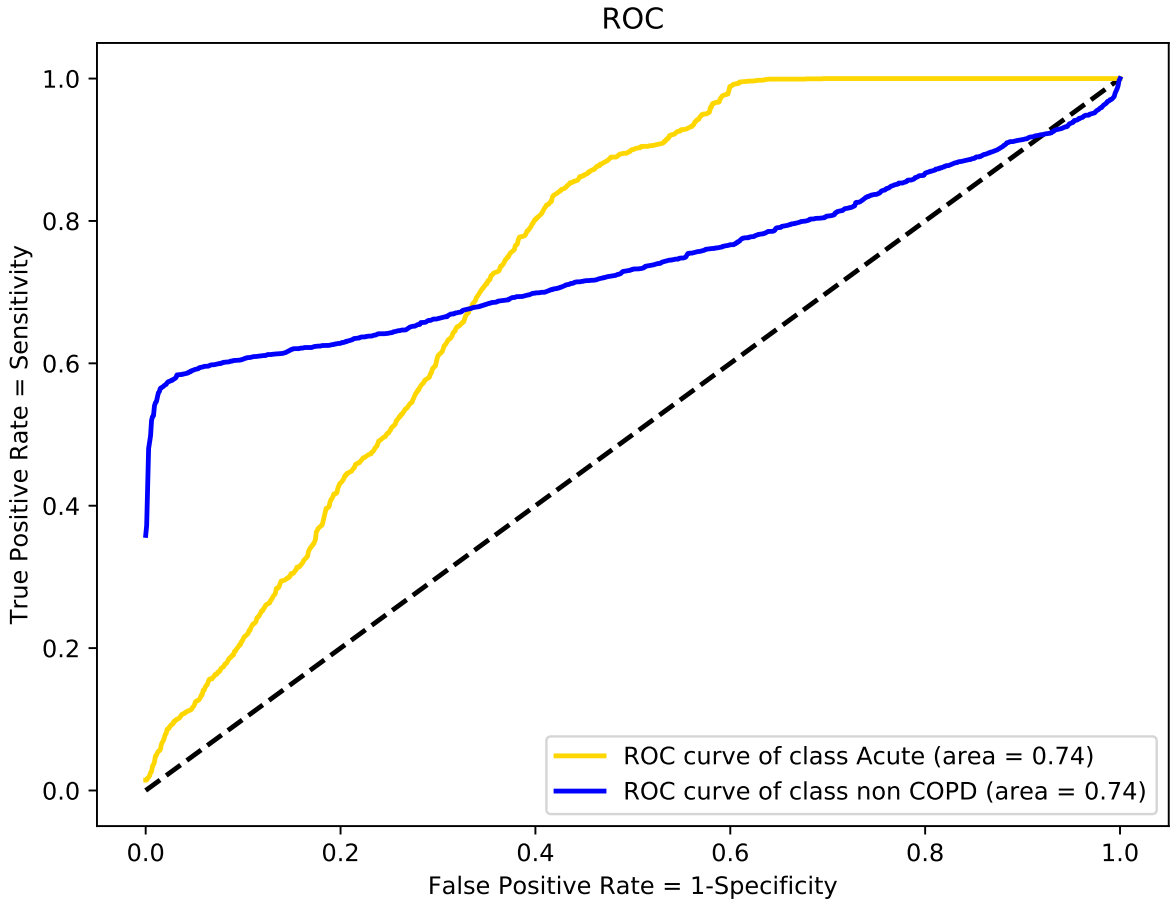



Figura 7.11: Curvas ROC del *3-fold cross-validation* en la clasificación de tos aguda frente a tos crónica que no es EPOC.

| | SEN | SPE | AUC | PPV | NPV | ACC |
|---|---|---|---|---|---|---|
| **Acute** | 0,72±0,05 | 0,66±0,15 | 0,74±0,13 | 0,69±0,11 | 0,70±0,08 | 0,69±0,10 |
| **non COPD** | 0,66±0,15 | 0,72±0,05 | 0,74±0,13 | 0,70±0,08 | 0,69±0,11 | 0,69±0,10 |
| **macro avg** | 0,69±0,10 | 0,69±0,10 | 0,74±0,13 | 0,69±0,10 | 0,69±0,10 | 0,69±0,10 |
| **weighted avg** | 0,70±0,09 | 0,69±0,10 | 0,74±0,13 | 0,69±0,10 | 0,70±0,10 | 0,69±0,10 |
| **micro avg** | 0,69±0,10 | 0,69±0,10 | 0,77±0,12 | 0,69±0,10 | 0,69±0,10 | 0,69±0,10 |

Tabla 7.19: Media±Desviación estándar de las métricas del *3-fold cross-validation* en la clasificación de tos aguda frente a tos crónica que no es EPOC.

Revisando la Tabla 7.19, podemos ver que la clase tos aguda es más sensible mientras que la clase tos crónica que no es EPOC es más específica. Asimismo el AUC nos indica que es un test regular cercano a ser un test bueno y las curvas ROC asociadas se pueden ver en la Figura 7.11.

| | | | Clasificación ( %) | | |
|---|---|---|---|---|---|
| **Sujeto** | **Ficheros** | **Enfermedad** | **Acute** | **non COPD** | **Diagnóstico** |
| **pal13** | 1579 | Acute | 74,2875 | 25,7125 | Acute |
| **pal18** | 697 | non COPD | 81,4921 | 18,5079 | Acute |
| **edi02** | 96 | non COPD | 0 | 100 | non COPD |
| **gla04** | 794 | non COPD | 0,1259 | 99,8741 | non COPD |
| **pal00** | 265 | Acute | 66,0377 | 33,9623 | Acute |
| **pal02** | 162 | non COPD | 87,6543 | 12,3457 | Acute |
| **pal04** | 601 | Acute | 79,7005 | 20,2995 | Acute |
| **edi05** | 100 | non COPD | 1 | 99 | non COPD |
| **edi08** | 50 | non COPD | 2 | 98 | non COPD |
| **edi10** | 110 | non COPD | 5,4545 | 94,5455 | non COPD |
| **gla08** | 443 | non COPD | 0,2257 | 99,7743 | non COPD |
| **pal03** | 352 | Acute | 57,9545 | 42,0455 | Acute |
| **pal10** | 508 | non COPD | 79,1339 | 20,8661 | Acute |
| **pal14** | 504 | Acute | 71,8254 | 28,1746 | Acute |
| **gla12** | 341 | non COPD | 0 | 100 | non COPD |

Tabla 7.20: Diagnósticos del *3-fold cross-validation* en la clasificación de tos aguda frente a tos crónica que no es EPOC. Los diagnósticos correctos se han presentado con fondo verde, siendo este diagnóstico la clase que más ha sido detectada en las toses del paciente.

Ahora hemos obtenido 12 de 15 diagnósticos correctos, pudiéndose contrastar en la Tabla 7.20, lo cual ofrece un acierto del 80 %. La tasa de acierto que nos ofrece nuestra red neuronal en este caso es la más alta distinguiendo entre dos clases de toses según la enfermedad subyacente, pero como dijimos anteriormente este resultado puede haberse debido a que la clase tos aguda solo esta en una de las tres bases de datos.

### 7.2.4. EPOC frente a cáncer de pulmón

La cuarta combinación de la que vamos a hablar es EPOC frente a cáncer de pulmón, en este caso la clase limitante para realizar un entrenamiento balanceado es cáncer de pulmón. Estas dos clases tienen un gran interés, ya que como vimos en la sección 6.2.2 son las dos enfermedades respiratorias que más muertes causan. Con

el propósito de obtener el mayor rédito a nuestros datos, vamos a realizar un *5-fold cross-validation*, siendo la configuración de los *folds* la descrita en la Tabla 6.17.

| | **SEN** | **SPE** | **AUC** | **PPV** | **NPV** | **ACC** |
|---|---|---|---|---|---|---|
| **COPD** | 0,57±0,31 | 0,32±0,34 | 0,43±0,09 | 0,44±0,05 | 0,40±0,20 | 0,44±0,07 |
| **Lung cancer** | 0,32±0,34 | 0,57±0,31 | 0,43±0,09 | 0,40±0,20 | 0,44±0,05 | 0,44±0,07 |
| **macro avg** | 0,45±0,07 | 0,45±0,07 | 0,44±0,09 | 0,42±0,12 | 0,42±0,12 | 0,44±0,07 |
| **weighted avg** | 0,64±0,20 | 0,26±0,16 | 0,43±0,09 | 0,44±0,07 | 0,40±0,17 | 0,44±0,07 |
| **micro avg** | 0,44±0,07 | 0,44±0,07 | 0,42±0,09 | 0,44±0,07 | 0,44±0,07 | 0,44±0,07 |

Tabla 7.21: Media±Desviación estándar de las métricas del *5-fold cross-validation* en la clasificación de EPOC frente a cáncer de pulmón.

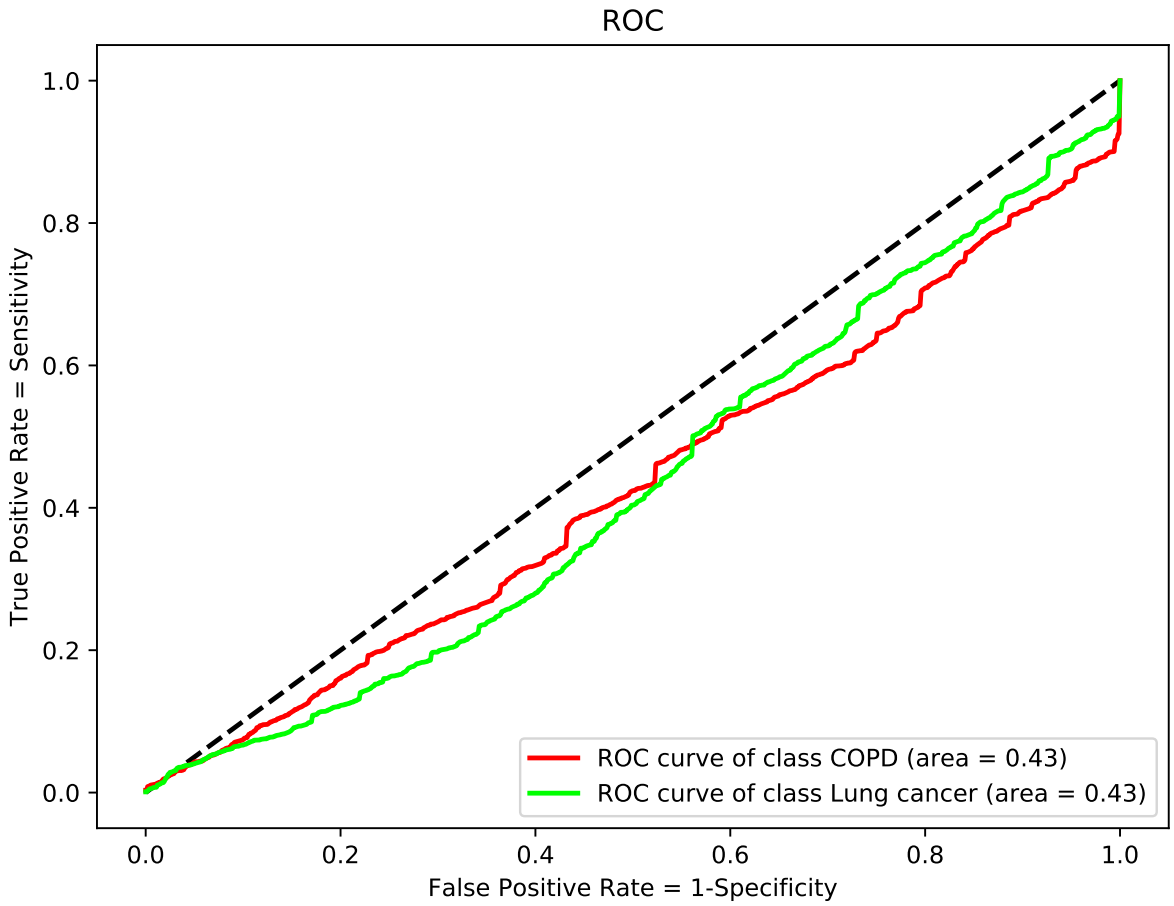



Figura 7.12: Curvas ROC del *5-fold cross-validation* en la clasificación de EPOC frente a cáncer de pulmón.

Observando la Tabla 7.21 podemos ver que la clase EPOC es más sensible, mientras que la clase cáncer de pulmón es más específica. También podemos ver que el AUC nos indica que este test no es fiable, ya que es inferior al 50 %, siendo las curvas asociadas a dichas AUC las que vemos en la Figura 7.12.

| Sujeto | Ficheros | Enfermedad | Clasificación ( %) | | Diagnóstico |
|---|---|---|---|---|---|
| | | | COPD | Lung cancer | |
| **pal06** | 921 | Lung cancer | 72,6384 | 27,3616 | COPD |
| **pal12** | 901 | COPD | 68,5905 | 31,4095 | COPD |
| **pal05** | 766 | COPD | 19,0601 | 80,9399 | Lung cancer |
| **pal08** | 809 | Lung cancer | 17,7998 | 82,2002 | Lung cancer |
| **edi11** | 70 | COPD | 77,1429 | 22,8571 | COPD |
| **gla03** | 535 | COPD | 99,4393 | 0,5607 | COPD |
| **gla05** | 784 | Lung cancer | 97,1939 | 2,8061 | COPD |
| **edi03** | 103 | COPD | 74,7573 | 25,2427 | COPD |
| **edi04** | 86 | COPD | 62,7907 | 37,2093 | COPD |
| **edi09** | 134 | COPD | 66,4179 | 33,5821 | COPD |
| **gla11** | 334 | Lung cancer | 98,503 | 1,497 | COPD |
| **edi07** | 32 | COPD | 18,75 | 81,25 | Lung cancer |
| **gla09** | 9 | COPD | 88,8889 | 11,1111 | COPD |
| **gla15** | 44 | Lung cancer | 52,2727 | 47,7273 | COPD |

Tabla 7.22: Diagnósticos del *5-fold cross-validation* en la clasificación de EPOC frente a cáncer de pulmón. Los diagnósticos correctos se han presentado con fondo verde, siendo este diagnóstico la clase que más ha sido detectada en las toses del paciente.

Para este caso hemos obtenido 8 de 14 diagnósticos correctos, como se puede corroborar en la Tabla 7.22, lo cual ofrece un acierto del 57,14 %. Esta tasa de acierto es más alta de lo que se podía esperar según las métricas analizadas anteriormente, y por tanto hay que ser escéptico con dicho resultado. Posteriormente procedemos a realizar un *3-fold cross-validation*, siendo la configuración de los *folds* la descrita en la Tabla 6.18.

| | SEN | SPE | AUC | PPV | NPV | ACC |
|---|---|---|---|---|---|---|
| **COPD** | 0,32±0,09 | 0,70±0,11 | 0,50±0,16 | 0,50±0,14 | 0,53±0,09 | 0,52±0,10 |
| **Lung cancer** | 0,70±0,11 | 0,32±0,09 | 0,50±0,16 | 0,53±0,09 | 0,50±0,14 | 0,52±0,10 |
| **macro avg** | 0,51±0,10 | 0,51±0,10 | 0,50±0,16 | 0,52±0,12 | 0,52±0,12 | 0,52±0,10 |
| **weighted avg** | 0,59±0,11 | 0,43±0,08 | 0,50±0,16 | 0,52±0,10 | 0,51±0,13 | 0,52±0,10 |
| **micro avg** | 0,52±0,10 | 0,52±0,10 | 0,52±0,15 | 0,52±0,10 | 0,52±0,10 | 0,52±0,10 |

Tabla 7.23: Media±Desviación estándar de las métricas del *3-fold cross-validation* en la clasificación de EPOC frente a cáncer de pulmón.

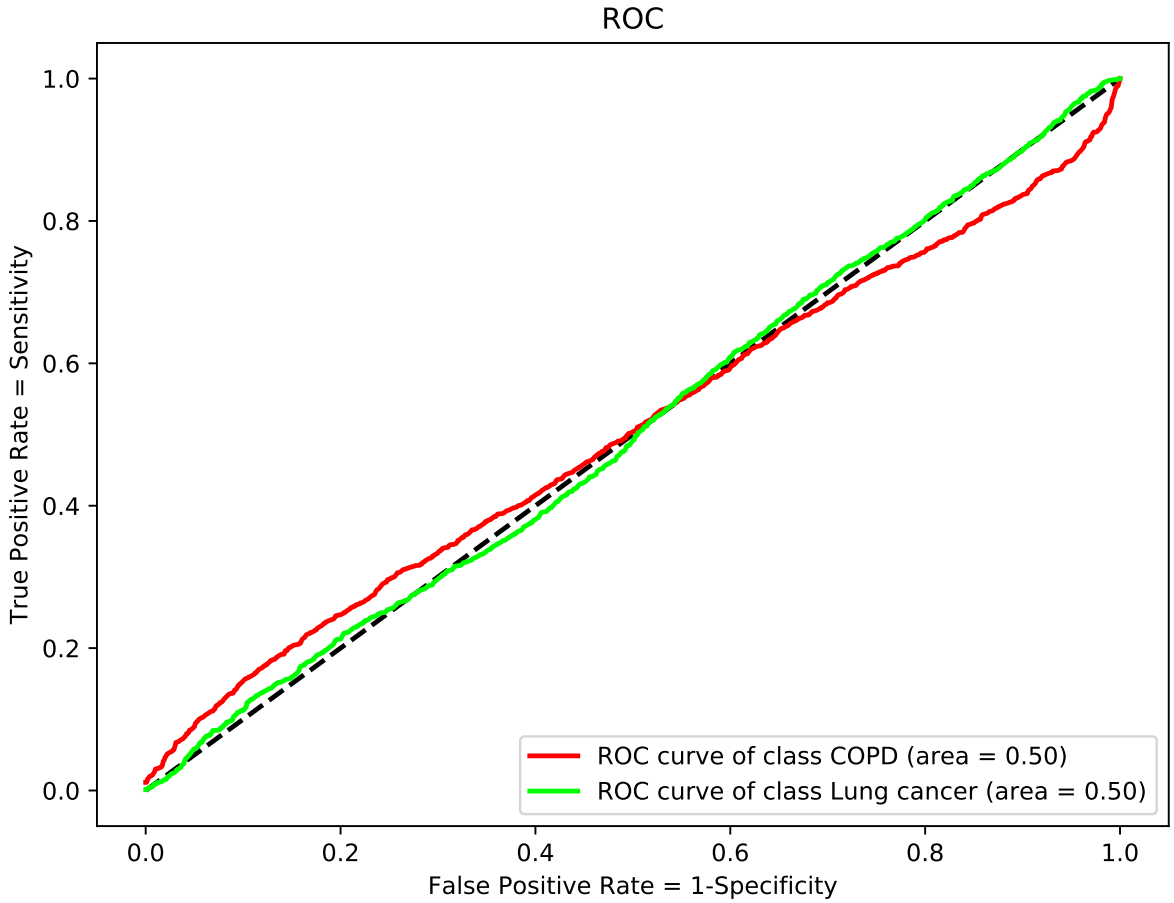



Figura 7.13: Curvas ROC del *3-fold cross-validation* en la clasificación de EPOC frente a cáncer de pulmón.

Analizando la Tabla 7.23 podemos ver esta vez que la clase cáncer de pulmón es más sensible y la clase EPOC es más específica, al contrario de lo que sucedía en el *5-fold cross-validation*. También se puede observar que el AUC es superior al caso anterior, pero sigue sin ser fiable, ya que tenemos un AUC del 50 %, siendo las curvas ROC de la Figura 7.13 las asociadas a dichas AUC.

| Sujeto | Ficheros | Enfermedad | Clasificación ( %) | | Diagnóstico |
|---|---|---|---|---|---|
| | | | COPD | Lung cancer | |
| **pal06** | 921 | Lung cancer | 39,5223 | 60,4777 | Lung cancer |
| **pal12** | 901 | COPD | 22,0866 | 77,9134 | Lung cancer |
| **pal05** | 766 | COPD | 31,7232 | 68,2768 | Lung cancer |
| **pal08** | 809 | Lung cancer | 28,1829 | 71,8171 | Lung cancer |
| **edi07** | 32 | COPD | 100 | 0 | COPD |
| **gla09** | 9 | COPD | 33,3333 | 66,6667 | Lung cancer |
| **gla15** | 44 | Lung cancer | 84,0909 | 15,9091 | COPD |
| **edi03** | 103 | COPD | 21,3592 | 78,6408 | Lung cancer |
| **edi04** | 86 | COPD | 9,3023 | 90,6977 | Lung cancer |
| **edi09** | 134 | COPD | 4,4776 | 95,5224 | Lung cancer |
| **edi11** | 70 | COPD | 10 | 90 | Lung cancer |
| **gla03** | 535 | COPD | 58,8785 | 41,1215 | COPD |

| Sujeto | Ficheros | Enfermedad | Clasificación ( %) | | Diagnóstico |
|---|---|---|---|---|---|
| | | | COPD | Lung cancer | |
| **gla05** | 784 | Lung cancer | 14,2857 | 85,7143 | Lung cancer |
| **gla11** | 334 | Lung cancer | 26,6467 | 73,3533 | Lung cancer |

Tabla 7.24: Diagnósticos del *3-fold cross-validation* en la clasificación de EPOC frente a cáncer de pulmón. Los diagnósticos correctos se han presentado con fondo verde, siendo este diagnóstico la clase que más ha sido detectada en las toses del paciente.

En este caso, en la Tabla 7.24, hemos obtenido 6 de 14 diagnósticos correctos, lo cual ofrece un acierto del 42,86 %. Siendo esta tasa de acierto inferior a la tasa de acierto del *5-fold cross-validation*, aunque como hemos visto las métricas son mejores.

Vamos a realizar ahora otro *3-fold cross-validation* con los mismos datos, pero modificando levemente la metodología con la que realizamos el espectrograma descrito en la sección 6.2.3. Esta modificación busca que el espectrograma tenga mayor resolución temporal y para esto elegimos un *overlap* del 70 %. Buscar esta mayor resolución temporal justamente en este caso es debido a que como hemos dicho anteriormente es un caso muy importante, además de que es el caso en el que menos datos tenemos y por lo tanto tenemos menos limitaciones de *hardware* a la hora de aumentar el tamaño de los espectrogramas.

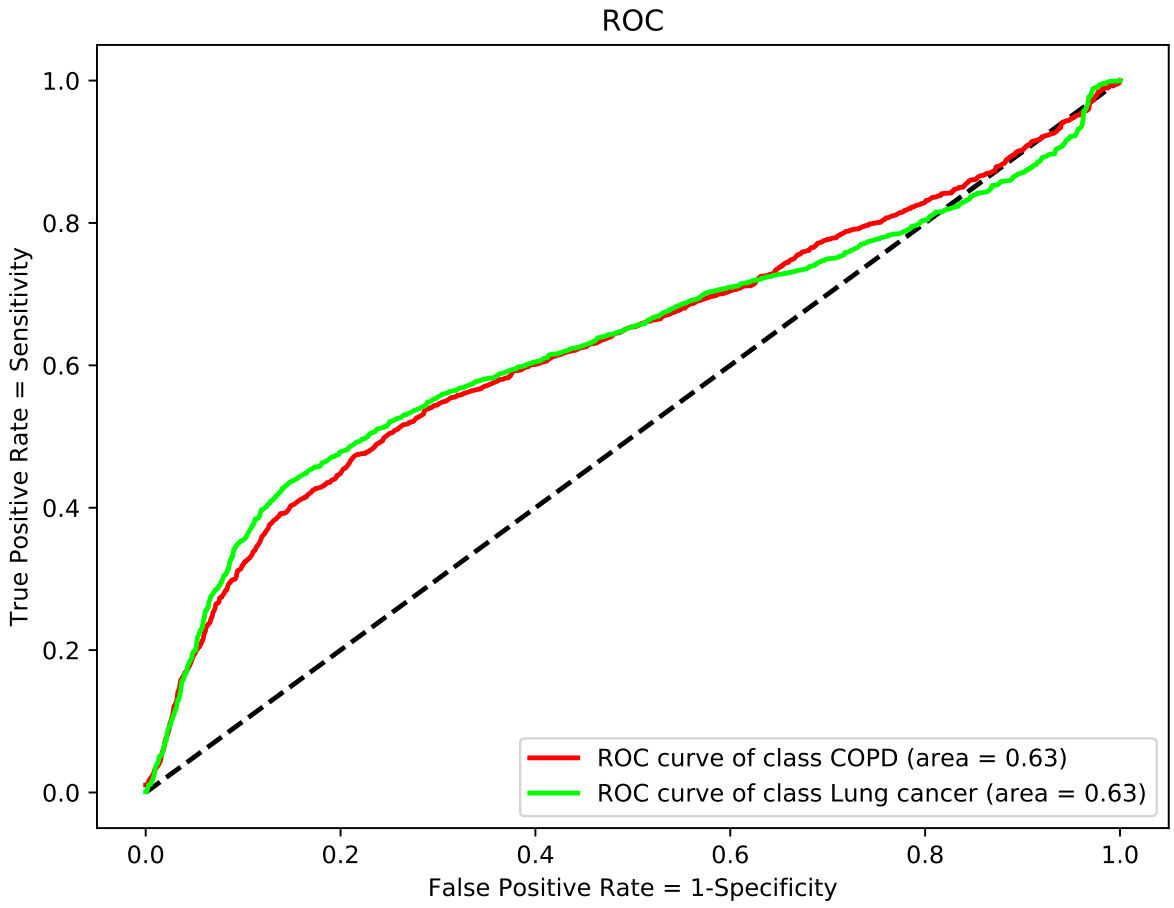



Figura 7.14: Curvas ROC del *3-fold cross-validation* en la clasificación de EPOC frente a cáncer de pulmón.

| | SEN | SPE | AUC | PPV | NPV | ACC |
|---|---|---|---|---|---|---|
| **COPD** | 0,32±0,39 | 0,81±0,18 | 0,63±0,29 | 0,59±0,31 | 0,59±0,18 | 0,58±0,21 |
| **Lung cancer** | 0,81±0,18 | 0,32±0,39 | 0,63±0,29 | 0,59±0,18 | 0,59±0,31 | 0,58±0,21 |
| **macro avg** | 0,56±0,22 | 0,56±0,22 | 0,63±0,29 | 0,59±0,24 | 0,59±0,24 | 0,58±0,21 |
| **weighted avg** | 0,75±0,23 | 0,38±0,37 | 0,63±0,29 | 0,58±0,21 | 0,60±0,28 | 0,58±0,21 |
| **micro avg** | 0,58±0,21 | 0,58±0,21 | 0,62±0,26 | 0,58±0,21 | 0,58±0,21 | 0,58±0,21 |

Tabla 7.25: Media±Desviación estándar de las métricas del *3-fold cross-validation* en la clasificación de EPOC frente a cáncer de pulmón.

Claramente las métricas observadas en la Tabla 7.25 han mejorado con respecto a los casos anteriores y por lo tanto con menos limitaciones de *hardware* para este estudio se podrían analizar el resto de casos mejorando la resolución temporal de los espectrogramas. En este caso vemos que la clase cáncer de pulmón es mas sensible mientras que la clase EPOC es más específica. Además vemos que el AUC es notablemente mejor a los casos anteriores teniendo en este caso un test regular y por lo tanto mucho más fiable que en los casos anteriores. La Figura 7.14 nos muestra como son las curvas ROC asociadas a dichas AUC.

| | | | Clasificación ( %) | | |
|---|---|---|---|---|---|
| **Sujeto** | **Ficheros** | **Enfermedad** | **COPD** | **Lung cancer** | **Diagnóstico** |
| **pal06** | 921 | Lung cancer | 17,3724 | 82,6276 | Lung cancer |
| **pal12** | 901 | COPD | 77,0255 | 22,9745 | COPD |
| **pal05** | 766 | COPD | 8,7467 | 91,2533 | Lung cancer |
| **pal08** | 809 | Lung cancer | 34,9815 | 65,0185 | Lung cancer |
| **edi07** | 32 | COPD | 96,875 | 3,125 | COPD |
| **gla09** | 9 | COPD | 11,1111 | 88,8889 | Lung cancer |
| **gla15** | 44 | Lung cancer | 90,9091 | 9,0909 | COPD |
| **edi03** | 103 | COPD | 9,7087 | 90,2913 | Lung cancer |
| **edi04** | 86 | COPD | 2,3256 | 97,6744 | Lung cancer |
| **edi09** | 134 | COPD | 7,4627 | 92,5373 | Lung cancer |
| **edi11** | 70 | COPD | 15,7143 | 84,2857 | Lung cancer |
| **gla03** | 535 | COPD | 4,1121 | 95,8879 | Lung cancer |
| **gla05** | 784 | Lung cancer | 1,148 | 98,852 | Lung cancer |
| **gla11** | 334 | Lung cancer | 3,2934 | 96,7066 | Lung cancer |

Tabla 7.26: Diagnósticos del *3-fold cross-validation* en la clasificación de EPOC frente a cáncer de pulmón. Los diagnósticos correctos se han presentado con fondo verde, siendo este diagnóstico la clase que más ha sido detectada en las toses del paciente.

Observando la Tabla 7.26 vemos que en este caso hemos obtenido 6 de 14 diagnósticos correctos, lo cual ofrece un acierto del 42,86 %. Esta tasa de acierto es muy baja pero es más fiable debido a las métricas analizadas anteriormente.

### 7.2.5. EPOC frente a tos crónica que no es EPOC

La quinta combinación de la que vamos a hablar es EPOC frente a tos crónica que no es EPOC, siendo limitante para realizar un entrenamiento balanceado la clase tos crónica que no es EPOC. Estas dos clases son posiblemente las más similares de las clases que estudiamos, debido a que todas las toses son toses crónicas, añadiendo otra

dificultad añadida debido a que la clase tos crónica que no es EPOC alberga muchas enfermedades respiratorias. Con el propósito de obtener el mayor rédito a nuestros datos, vamos a realizar un *5-fold cross-validation*, siendo la configuración de los *folds* la descrita en la Tabla 6.19.

| | **SEN** | **SPE** | **AUC** | **PPV** | **NPV** | **ACC** |
|---|---|---|---|---|---|---|
| **COPD** | 0,86±0,14 | 0,25±0,26 | 0,49±0,16 | 0,54±0,11 | 0,61±0,19 | 0,55±0,14 |
| **non COPD** | 0,25±0,26 | 0,86±0,14 | 0,49±0,16 | 0,61±0,19 | 0,54±0,11 | 0,55±0,14 |
| **macro avg** | 0,56±0,14 | 0,56±0,14 | 0,49±0,16 | 0,58±0,15 | 0,58±0,15 | 0,55±0,14 |
| **weighted avg** | 0,78±0,18 | 0,33±0,28 | 0,49±0,16 | 0,55±0,14 | 0,60±0,16 | 0,55±0,14 |
| **micro avg** | 0,55±0,14 | 0,55±0,14 | 0,52±0,13 | 0,55±0,14 | 0,55±0,14 | 0,55±0,14 |

Tabla 7.27: Media±Desviación estándar de las métricas del *5-fold cross-validation* en la clasificación de EPOC frente a tos crónica que no es EPOC.

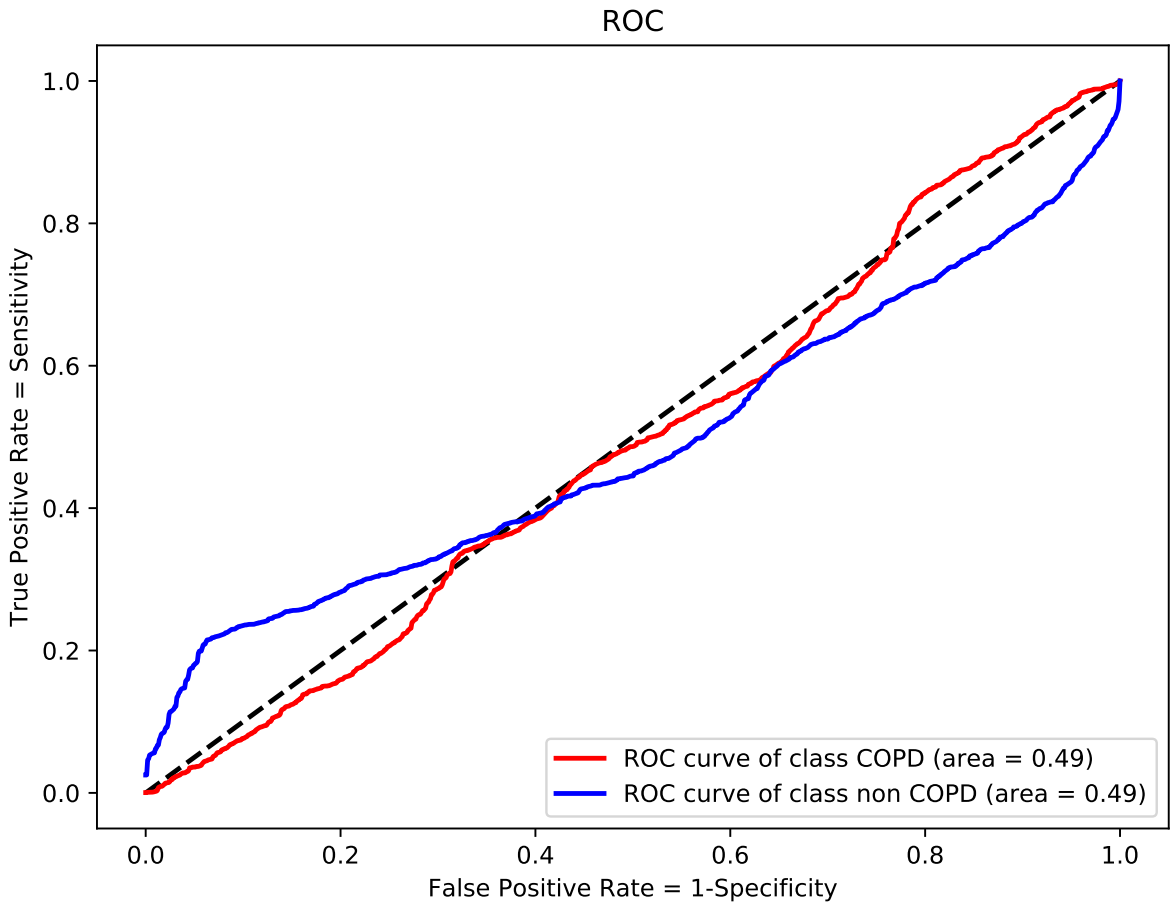



Figura 7.15: Curvas ROC del *5-fold cross-validation* en la clasificación de EPOC frente a tos crónica que no es EPOC.

Analizando la Tabla 7.27 vemos que la clase EPOC es más sensible mientras que la clase tos crónica que no es EPOC es más específica. Por otra parte también se observa que el AUC es inferior al 50 %, indicándonos que no es un test fiable, pudiéndose ver las curvas ROC asociadas en la Figura 7.15.

| Sujeto | Ficheros | Enfermedad | Clasificación ( %) | | Diagnóstico |
|---|---|---|---|---|---|
| | | | COPD | non COPD | |
| **pal12** | 901 | COPD | 83,9068 | 16,0932 | COPD |
| **edi02** | 96 | non COPD | 63,5417 | 36,4583 | COPD |
| **edi03** | 103 | COPD | 78,6408 | 21,3592 | COPD |
| **edi04** | 86 | COPD | 66,2791 | 33,7209 | COPD |
| **edi05** | 100 | non COPD | 58 | 42 | COPD |
| **edi07** | 32 | COPD | 6,25 | 93,75 | non COPD |
| **edi08** | 50 | non COPD | 38 | 62 | non COPD |
| **edi09** | 134 | COPD | 79,8507 | 20,1493 | COPD |
| **edi10** | 110 | non COPD | 73,6364 | 26,3636 | COPD |
| **edi11** | 70 | COPD | 50 | 50 | COPD |
| **gla01** | 205 | non COPD | 97,0732 | 2,9268 | COPD |
| **gla04** | 794 | non COPD | 84,7607 | 15,2393 | COPD |
| **gla09** | 9 | COPD | 77,7778 | 22,2222 | COPD |
| **pal05** | 766 | COPD | 93,0809 | 6,9191 | COPD |
| **pal18** | 697 | non COPD | 29,1248 | 70,8752 | non COPD |
| **edi01** | 74 | non COPD | 67,5676 | 32,4324 | COPD |
| **edi06** | 97 | non COPD | 61,8557 | 38,1443 | COPD |
| **gla03** | 535 | COPD | 95,514 | 4,486 | COPD |
| **gla08** | 443 | non COPD | 98,1941 | 1,8059 | COPD |
| **pal10** | 508 | non COPD | 72,8346 | 27,1654 | COPD |
| **gla10** | 509 | COPD | 65,4224 | 34,5776 | COPD |
| **pal02** | 162 | non COPD | 96,2963 | 3,7037 | COPD |
| **gla07** | 493 | COPD | 98,5801 | 1,4199 | COPD |
| **gla12** | 341 | non COPD | 98,827 | 1,173 | COPD |

Tabla 7.28: Diagnósticos del *5-fold cross-validation* en la clasificación de EPOC frente a tos crónica que no es EPOC. Los diagnósticos correctos se han presentado con fondo verde, siendo este diagnóstico la clase que más ha sido detectada en las toses del paciente.

En este caso hemos obtenido 12 de 24 diagnósticos correctos, como se puede comprobar en la Tabla 7.28, lo cual ofrece un acierto del 50 %, siendo esta tasa de acierto baja. Posteriormente procedemos a realizar un *3-fold cross-validation*, siendo la configuración de los *folds* la descrita en la Tabla 6.20.

| | SEN | SPE | AUC | PPV | NPV | ACC |
|---|---|---|---|---|---|---|
| **COPD** | 0,67±0,33 | 0,19±0,24 | 0,41±0,27 | 0,44±0,06 | 0,33±0,11 | 0,43±0,06 |
| **non COPD** | 0,19±0,24 | 0,67±0,33 | 0,41±0,27 | 0,33±0,11 | 0,44±0,06 | 0,43±0,06 |
| **macro avg** | 0,43±0,06 | 0,43±0,06 | 0,41±0,27 | 0,37±0,04 | 0,37±0,04 | 0,43±0,06 |
| **weighted avg** | 0,65±0,31 | 0,21±0,20 | 0,41±0,27 | 0,39±0,01 | 0,34±0,08 | 0,43±0,06 |
| **micro avg** | 0,43±0,06 | 0,43±0,06 | 0,42±0,16 | 0,43±0,06 | 0,43±0,06 | 0,43±0,06 |

Tabla 7.29: Media±Desviación estándar de las métricas del *3-fold cross-validation* en la clasificación de EPOC frente a tos crónica que no es EPOC.

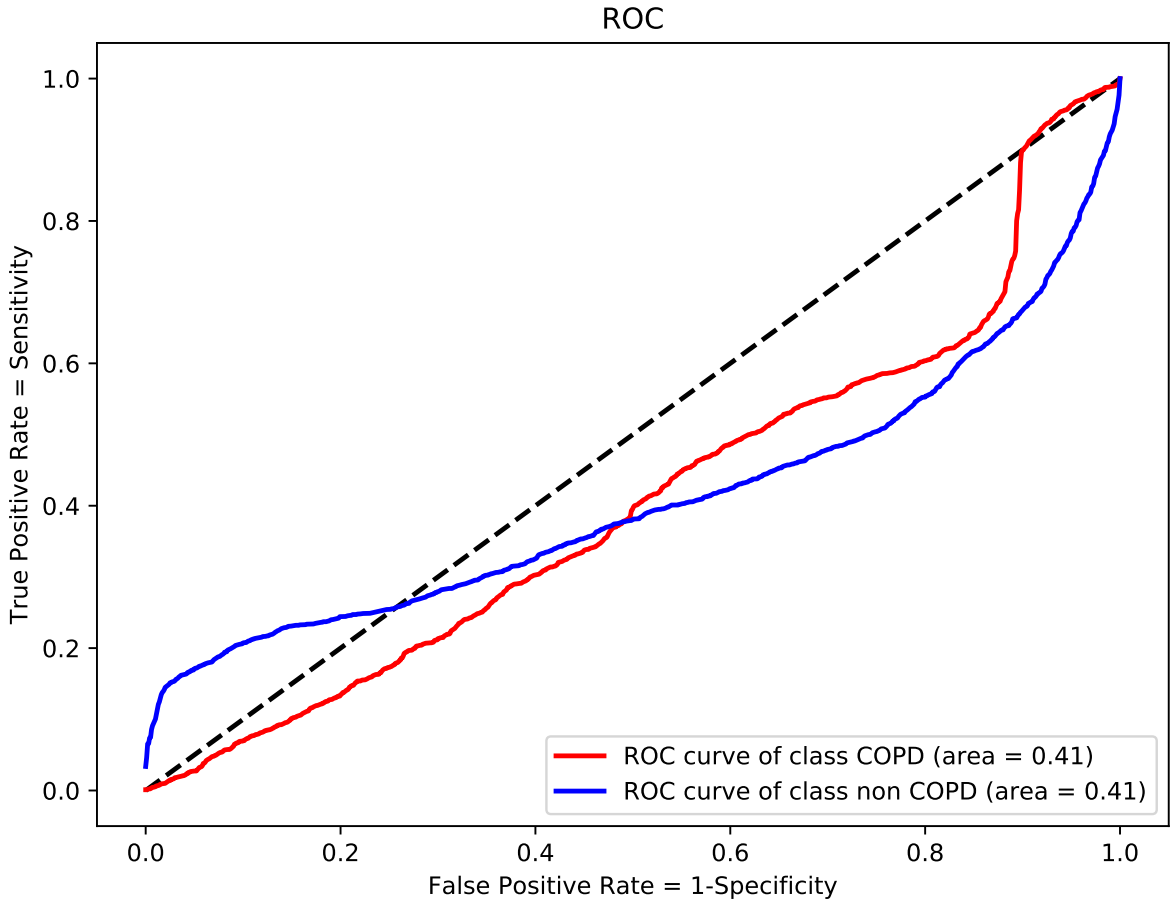



Figura 7.16: Curvas ROC del *3-fold cross-validation* en la clasificación de EPOC frente a tos crónica que no es EPOC.

Examinando la Tabla 7.29 se puede observar que nuestra red neuronal es más sensible respecto a la clase EPOC y más específica respecto a la clase tos crónica que no es EPOC. Asimismo el AUC es inferior al 50 % lo que denota que el test es poco fiable, siendo estas AUC las obtenidas a prtir de las curvas ROC de la Figura 7.16.

| Sujeto | Ficheros | Enfermedad | Clasificación ( %) | | Diagnóstico |
|---|---|---|---|---|---|
| | | | COPD | non COPD | |
| **pal12** | 901 | COPD | 65,3718 | 34,6282 | COPD |
| **edi02** | 96 | non COPD | 75 | 25 | COPD |
| **edi03** | 103 | COPD | 88,3495 | 11,6505 | COPD |
| **edi04** | 86 | COPD | 69,7674 | 30,2326 | COPD |
| **edi05** | 100 | non COPD | 68 | 32 | COPD |
| **edi07** | 32 | COPD | 15,625 | 84,375 | non COPD |
| **edi08** | 50 | non COPD | 66 | 34 | COPD |
| **edi09** | 134 | COPD | 82,8358 | 17,1642 | COPD |
| **edi10** | 110 | non COPD | 77,2727 | 22,7273 | COPD |
| **edi11** | 70 | COPD | 61,4286 | 38,5714 | COPD |
| **gla01** | 205 | non COPD | 99,5122 | 0,4878 | COPD |
| **gla04** | 794 | non COPD | 94,7103 | 5,2897 | COPD |
| **gla09** | 9 | COPD | 100 | 0 | COPD |

| Sujeto | Ficheros | Enfermedad | Clasificación ( %) | | Diagnóstico |
|---|---|---|---|---|---|
| | | | COPD | non COPD | |
| **pal02** | 162 | non COPD | 100 | 0 | COPD |
| **pal05** | 766 | COPD | 100 | 0 | COPD |
| **pal18** | 697 | non COPD | 100 | 0 | COPD |
| **edi01** | 74 | non COPD | 100 | 0 | COPD |
| **gla07** | 493 | COPD | 100 | 0 | COPD |
| **gla12** | 341 | non COPD | 100 | 0 | COPD |
| **pal10** | 508 | non COPD | 39,5669 | 60,4331 | non COPD |
| **edi06** | 97 | non COPD | 59,7938 | 40,2062 | COPD |
| **gla03** | 535 | COPD | 10,8411 | 89,1589 | non COPD |
| **gla08** | 443 | non COPD | 70,4289 | 29,5711 | COPD |
| **gla10** | 509 | COPD | 58,7426 | 41,2574 | COPD |

Tabla 7.30: Diagnósticos del *3-fold cross-validation* en la clasificación de EPOC frente a tos crónica que no es EPOC. Los diagnósticos correctos se han presentado con fondo verde, siendo este diagnóstico la clase que más ha sido detectada en las toses del paciente.

En este caso, como vemos en la Tabla 7.30, hemos obtenido 10 de 24 diagnósticos correctos, lo cual ofrece un acierto del 41,67 %, siendo esta tasa de acierto es muy baja.

### 7.2.6. Cáncer de pulmón frente a tos crónica que no es EPOC

La sexta y última combinación de la que vamos a hablar es cáncer de pulmón frente a tos crónica que no es EPOC, siendo limitante para realizar un entrenamiento balanceado la clase cáncer de pulmón. Estas dos clases son las dos clases de las que menor cantidad de datos tenemos. Con la finalidad de obtener el mayor rendimiento de nuestros datos, vamos a realizar un *5-fold cross-validation*, siendo la configuración de los *folds* la descrita en la Tabla 6.21.

| | SEN | SPE | AUC | PPV | NPV | ACC |
|---|---|---|---|---|---|---|
| **Lung cancer** | 0,66±0,33 | 0,34±0,33 | 0,48±0,09 | 0,51±0,09 | 0,61±0,24 | 0,50±0,06 |
| **non COPD** | 0,34±0,33 | 0,66±0,33 | 0,48±0,09 | 0,61±0,24 | 0,51±0,09 | 0,50±0,06 |
| **macro avg** | 0,50±0,06 | 0,50±0,06 | 0,48±0,08 | 0,56±0,13 | 0,56±0,13 | 0,50±0,06 |
| **weighted avg** | 0,72±0,24 | 0,28±0,15 | 0,48±0,09 | 0,50±0,06 | 0,63±0,22 | 0,50±0,06 |
| **micro avg** | 0,50±0,06 | 0,50±0,06 | 0,48±0,07 | 0,50±0,06 | 0,50±0,06 | 0,50±0,06 |

Tabla 7.31: Media±Desviación estándar de las métricas del *5-fold cross-validation* en la clasificación de cáncer de pulmón frente a tos crónica que no es EPOC.

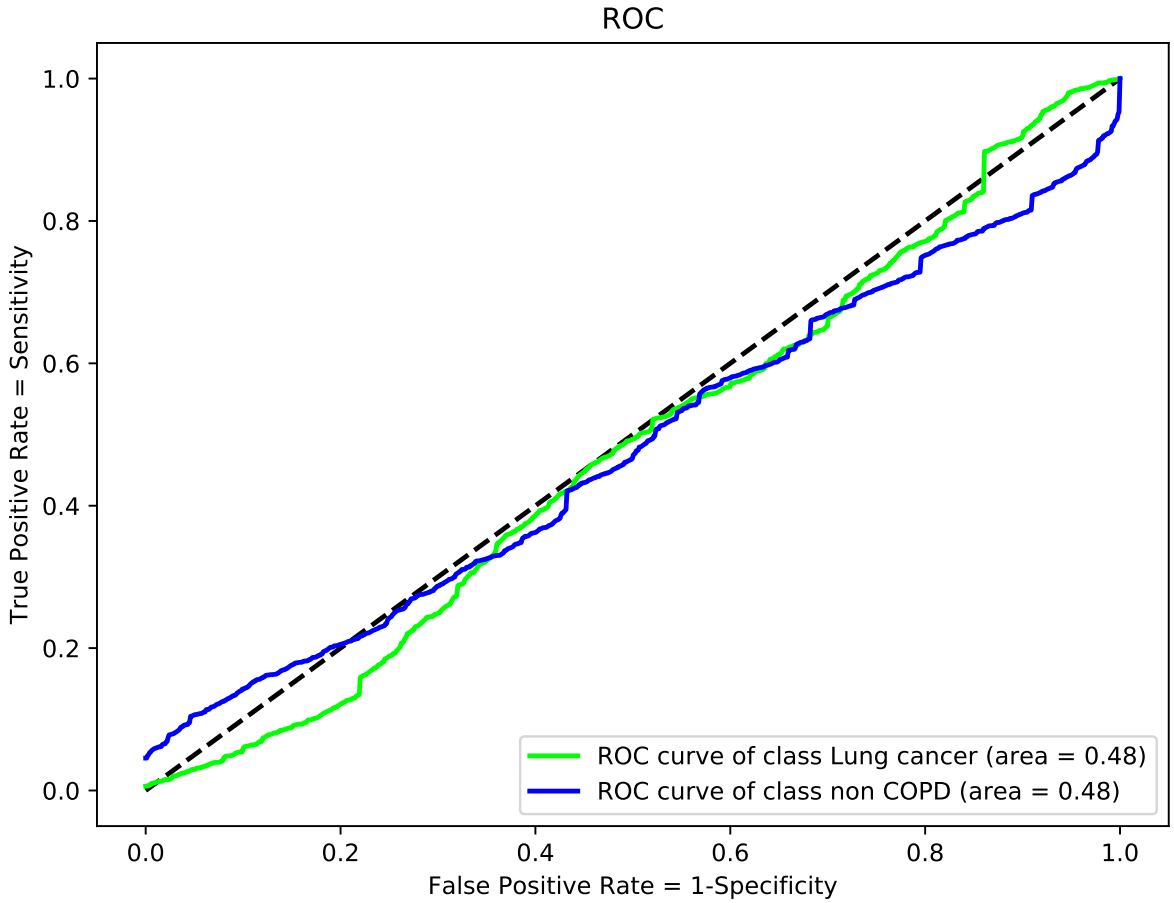



Figura 7.17: Curvas ROC del *5-fold cross-validation* en la clasificación de cáncer de pulmón frente a tos crónica que no es EPOC.

Analizando la Tabla 7.31 podemos ver nuestra red neuronal es más sensible a la clase cáncer de pulmón mientras que es mas específica con la clase tos crónica que no es EPOC. Por otro lado vemos que el AUC, asociado a las curvas ROC de la Figura 7.17, es inferior al 50 % lo cual indica que no es un test fiable.

| Sujeto | Ficheros | Enfermedad | Clasificación ( %) | | Diagnóstico |
|---|---|---|---|---|---|
| | | | Lung cancer | non COPD | |
| **pal06** | 921 | Lung cancer | 97,937 | 2,063 | Lung cancer |
| **edi10** | 110 | non COPD | 83,6364 | 16,3636 | Lung cancer |
| **gla04** | 794 | non COPD | 97,4811 | 2,5189 | Lung cancer |
| **pal08** | 809 | Lung cancer | 20,89 | 79,11 | non COPD |
| **pal18** | 697 | non COPD | 10,7604 | 89,2396 | non COPD |
| **edi05** | 100 | non COPD | 18 | 82 | non COPD |
| **pal10** | 508 | non COPD | 86,811 | 13,189 | Lung cancer |
| **edi01** | 74 | non COPD | 25,6757 | 74,3243 | non COPD |
| **gla01** | 205 | non COPD | 18,0488 | 81,9512 | non COPD |
| **gla05** | 784 | Lung cancer | 57,398 | 42,602 | Lung cancer |
| **pal02** | 162 | non COPD | 98,7654 | 1,2346 | Lung cancer |
| **edi02** | 96 | non COPD | 83,3333 | 16,6667 | Lung cancer |
| **edi06** | 97 | non COPD | 74,2268 | 25,7732 | Lung cancer |

| Sujeto | Ficheros | Enfermedad | Clasificación ( %) | | Diagnóstico |
|---|---|---|---|---|---|
| | | | Lung cancer | non COPD | |
| **gla11** | 334 | Lung cancer | 100 | 0 | Lung cancer |
| **edi08** | 50 | non COPD | 70 | 30 | Lung cancer |
| **gla15** | 44 | Lung cancer | 52,2727 | 47,7273 | Lung cancer |

Tabla 7.32: Diagnósticos del *5-fold cross-validation* en la clasificación de cáncer de pulmón frente a tos crónica que no es EPOC. Los diagnósticos correctos se han presentado con fondo verde, siendo este diagnóstico la clase que más ha sido detectada en las toses del paciente.

Para este caso hemos obtenido 8 de 16 diagnósticos correctos, según observamos en la Tabla 7.32, lo cual ofrece un acierto del 50 %, siendo esta tasa de acierto es baja y poco fiable. A continuación, procedemos a realizar un *3-fold cross-validation*, siendo la configuración de los *folds* la descrita en la Tabla 6.22.

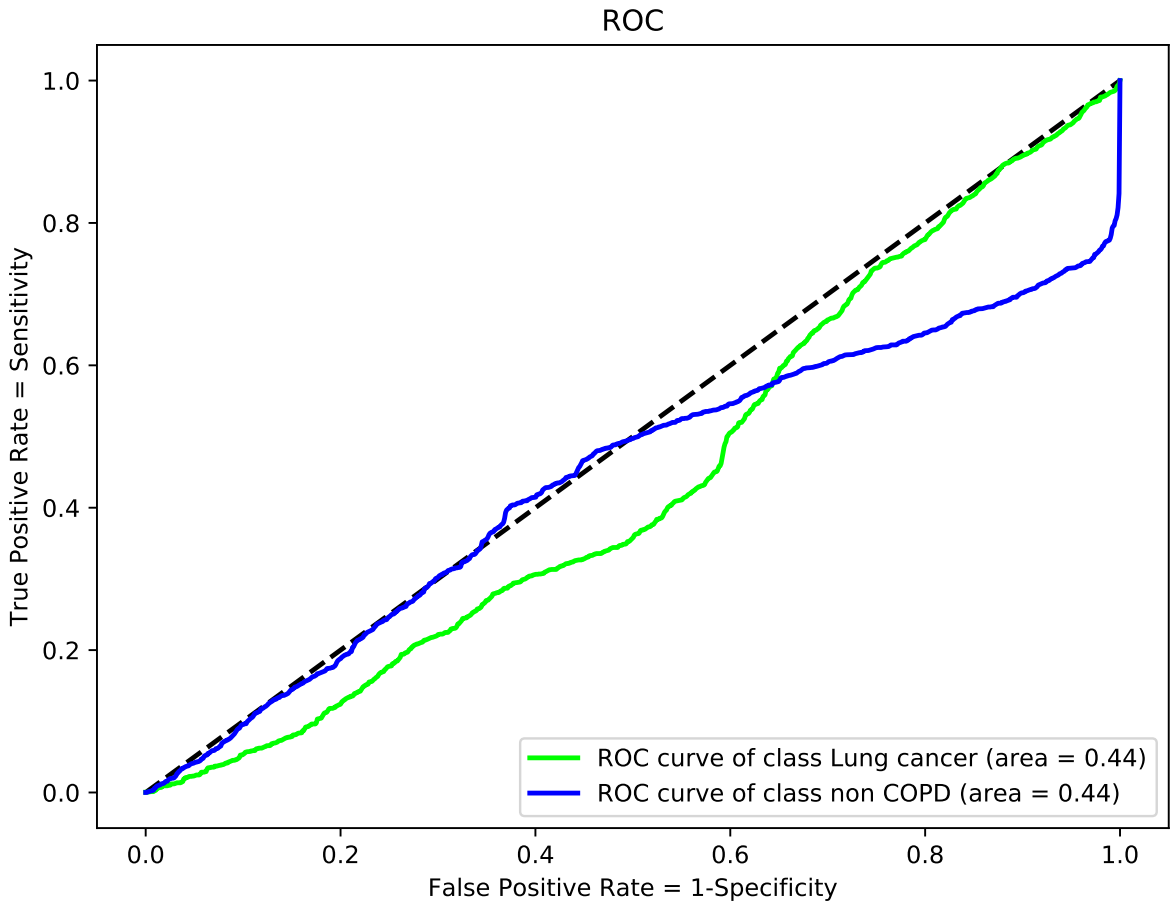



Figura 7.18: Curvas ROC del *3-fold cross-validation* en la clasificación de cáncer de pulmón frente a tos crónica que no es EPOC.

| | SEN | SPE | AUC | PPV | NPV | ACC |
|---|---|---|---|---|---|---|
| **Lung cancer** | 0,38±0,54 | 0,47±0,45 | 0,44±0,15 | 0,37±0,32 | 0,44±0,08 | 0,43±0,15 |
| **non COPD** | 0,47±0,45 | 0,38±0,54 | 0,44±0,15 | 0,44±0,08 | 0,37±0,32 | 0,43±0,15 |
| **macro avg** | 0,43±0,15 | 0,43±0,15 | 0,44±0,15 | 0,40±0,20 | 0,40±0,20 | 0,43±0,15 |
| **weighted avg** | 0,73±0,31 | 0,13±0,11 | 0,44±0,15 | 0,43±0,15 | 0,38±0,26 | 0,43±0,15 |
| **micro avg** | 0,43±0,15 | 0,43±0,15 | 0,44±0,13 | 0,43±0,15 | 0,43±0,15 | 0,43±0,15 |

Tabla 7.33: Media±Desviación estándar de las métricas del *3-fold cross-validation* en la clasificación de cáncer de pulmón frente a tos crónica que no es EPOC.

Examinando la Tabla 7.33, se puede observar que nuestra red neuronal es más sensible con respecto a la clase tos crónica que no es EPOC, mientras que es más específica con la clase cáncer de pulmón. Asimismo vemos que el AUC asociado a las curvas ROC de la Figura 7.18, es inferior al 50 % lo cual nos indica que dicho test no es fiable.

| | | | Clasificación ( %) | | |
|---|---|---|---|---|---|
| **Sujeto** | **Ficheros** | **Enfermedad** | **Lung cancer** | **non COPD** | **Diagnóstico** |
| **pal06** | 921 | Lung cancer | 99,3485 | 0,6515 | Lung cancer |
| **edi10** | 110 | non COPD | 98,1818 | 1,8182 | Lung cancer |
| **gla04** | 794 | non COPD | 99,6222 | 0,3778 | Lung cancer |
| **pal08** | 809 | Lung cancer | 14,0915 | 85,9085 | non COPD |
| **pal18** | 697 | non COPD | 7,3171 | 92,6829 | non COPD |
| **edi05** | 100 | non COPD | 20 | 80 | non COPD |
| **edi08** | 50 | non COPD | 22 | 78 | non COPD |
| **gla15** | 44 | Lung cancer | 20,4545 | 79,5455 | non COPD |
| **pal02** | 162 | non COPD | 88,8889 | 11,1111 | Lung cancer |
| **pal10** | 508 | non COPD | 78,937 | 21,063 | Lung cancer |
| **edi01** | 74 | non COPD | 1,3514 | 98,6486 | non COPD |
| **edi02** | 96 | non COPD | 0 | 100 | non COPD |
| **edi06** | 97 | non COPD | 9,2784 | 90,7216 | non COPD |
| **gla01** | 205 | non COPD | 0 | 100 | non COPD |
| **gla05** | 784 | Lung cancer | 0,5102 | 99,4898 | non COPD |
| **gla11** | 334 | Lung cancer | 0 | 100 | non COPD |

Tabla 7.34: Diagnósticos del *3-fold cross-validation* en la clasificación de cáncer de pulmón frente a tos crónica que no es EPOC. Los diagnósticos correctos se han presentado con fondo verde, siendo este diagnóstico la clase que más ha sido detectada en las toses del paciente.

Para este último caso hemos obtenido 8 de 16 diagnósticos correctos, según lo observado en la Tabla 7.34, lo cual ofrece un acierto del 50 %, siendo esta una tasa baja de acierto.

## 7.3. Comparativa con los trabajos de vanguardia

En esta sección buscaremos hacer una comparativa entre los resultados obtenidos en publicaciones previas y los presentados en este trabajo.

### 7.3.1. Detección de tos

Para los resultados obtenidos en esta parte de nuestro trabajo realizaremos la comparativa de nuestro trabajo con cuatro trabajos previos. Dado que estos trabajos anteriores han sido probados en la base de datos "Edimburgo" primero compararemos los resultados obtenidos en sus respectivas publicaciones para posteriormente compararlos en dicha base de datos.

Vale la pena señalar que el rendimiento reportado en [184] y [185] fue superior al obtenido en la base de datos "Edimburgo". Además de advertir que la base de datos "Palencia" con la que se ha hecho nuestro estudio, tiene un ruido real según las actividades realizadas por el paciente durante el día y las otras bases de datos eran de pacientes en entornos hospitalarios con ruidos sintéticos [184] o que producía el paciente a voluntad siguiendo el protocolo de la prueba [185].

#### Detección de tos mediante el conjunto de características de subbanda de frecuencia múltiple

En lo que respecta al trabajo en [184], su principal inconveniente es el alto número de clasificadores que deben utilizarse. Igualmente, se debe realizar el cálculo de características para cada señal de subbanda. Estos factores conducen a una alta carga computacional, que se confirma cuando los autores solo usan 120.000 *frames* de datos reales (aproximadamente 20 minutos). Sin embargo, se registraron y etiquetaron un total de 1440 minutos de datos de pacientes.

| | **SEN** | **SPE** | **PPV** | **ACC** |
|---|---|---|---|---|
| ***5-Fold* CNN A** | 0,85 | 0,93 | 0,92 | 0,89 |
| ***10-Fold* CNN A** | 0,86 | 0,93 | 0,92 | 0,89 |
| ***5-Fold* CNN B** | 0,91 | 0,87 | 0,88 | 0,89 |
| ***10-Fold* CNN B** | 0,89 | 0,90 | 0,90 | 0,89 |
| ***Threshold Voting* (T = 9) [184]** | 0,79 | 0,87 | 0,86 | 0,83 |
| ***Weighted Voting* (k = 14) [184]** | 0,78 | 0,88 | 0,87 | 0,83 |

Tabla 7.35: Métricas obtenidas en nuestro estudio frente a las obtenidas en [184].

Vemos en la Tabla 7.35, donde podemos comparar las métricas obtenidas en nuestro estudio con respecto a [184], que nuestro método es mejor en cualquiera de las métricas observadas, por lo que la capacidad de reconocimiento de patrones de nuestro sistema se mostró mejor.

| | **SEN** | **SPE** | **AUC** |
|---|---|---|---|
| **CNN A** | 0,75 | 0,90 | 0,92 |
| **CNN B** | 0,96 | 0,83 | 0,98 |
| **[184]** | 0,76 | 0,75 | 0,76 |

Tabla 7.36: Métricas obtenidas en nuestro estudio con la base de datos "Edimburgo" frente a las obtenidas con el método de [184] con la misma base de datos.

En la Tabla 7.36, podemos comparar las métricas obtenidas en nuestro estudio con respecto a [184], utilizando en ambos métodos la misma base de datos [4]. En este caso observamos que nuestro método es mejor en cualquiera de las métricas observadas utilizando la CNN B, pero utilizando la CNN A su sensibilidad es levemente superior.

**Redes neuronales profundas para identificar los sonidos de la tos**

La propuesta en [185] tiene dos diferencias metodológicas notables: (1) emplearon un sensor piezoeléctrico unido al tórax mediante un adhesivo de grado médico para registrar las señales; (2) cuando los autores enfrentan un problema de clasificación de dos clases, solo usan el habla frente a la tos. Además en [185] trabajan con audio con una duración mínima de 320 ms, mientras que nosotros basamos nuestro estudio en audio con una duración mínima de un segundo.

Cabe destacar por último que la arquitectura de las dos CNN empleadas en este estudio tienen unas arquitecturas bastante diferentes a las utilizadas en [185].

| | **SEN** | **SPE** | **ACC** |
|---|---|---|---|
| ***5-Fold* CNN A** | 0,85 | 0,93 | 0,89 |
| ***10-Fold* CNN A** | 0,86 | 0,93 | 0,89 |
| ***5-Fold* CNN B** | 0,91 | 0,87 | 0,89 |
| ***10-Fold* CNN B** | 0,89 | 0,90 | 0,89 |
| **CNN [185]** | 0,82 | 0,93 | 0,88 |
| **RNN [185]** | 0,84 | 0,75 | 0,80 |

Tabla 7.37: Métricas obtenidas en nuestro estudio frente a las obtenidas en [185].

Vemos en la Tabla 7.37, donde podemos comparar las métricas obtenidas en nuestro estudio con respecto a la CNN de [185], que nuestro método es mejor en cualquiera de las métricas observadas utilizando la CNN A, pero utilizando la CNN B su especificidad es levemente superior.

Si comparamos nuestro estudio con la RNN de [185] nuestro método es mejor en cualquiera de las métricas observadas.

| | **SEN** | **SPE** | **AUC** |
|---|---|---|---|
| **CNN A** | 0,75 | 0,90 | 0,92 |
| **CNN B** | 0,96 | 0,83 | 0,98 |
| **[185]** | 0,73 | 0,89 | 0,81 |

Tabla 7.38: Métricas obtenidas en nuestro estudio con la base de datos “Edimburgo” frente a las obtenidas con el método de [185] con la misma base de datos.

Por otro lado, en la Tabla 7.36, podemos comparar las métricas obtenidas en nuestro estudio con respecto a la CNN de [185], utilizando en ambos métodos la misma base de datos [4]. En este caso observamos que nuestro método es mejor en cualquiera de las métricas observadas utilizando la CNN A.

Aunque utilizando la CNN B su especificidad es superior un 6 %, pero nuestra sensibilidad es superior un 23 %, por lo que superaron ligeramente nuestra propuesta en términos de especificidad a costa de que la sensibilidad sea significativamente más baja. En consecuencia, la pérdida asociada de información clínica (patrones de tos) es mayor en este sistema.

**Detección robusta de eventos de tos de audio utilizando momentos Hu locales**

La propuesta en [106], tiene una diferencia capital y es que todos los sonidos cuyo espectro es superior a 2 kHz se descartan directamente, mientras que en nuestro caso se consideran todos los sonidos grabados por el *smartphone*.

| | SEN | SPE | AUC |
|---|---|---|---|
| **CNN A** | 0,75 | 0,90 | 0,92 |
| **CNN B** | 0,96 | 0,83 | 0,98 |
| **[106]** | 0,60 | 0,90 | 0,75 |

Tabla 7.39: Métricas obtenidas en nuestro estudio con la base de datos “Edimburgo” frente a las obtenidas con el método de [106] con la misma base de datos.

En la Tabla 7.39, podemos comparar las métricas obtenidas en nuestro estudio con respecto a [106], utilizando en ambos métodos la misma base de datos [4]. En este caso observamos que nuestro método es mejor en cualquiera de las métricas observadas utilizando la CNN A.

Aunque utilizando la CNN B su especificidad es superior un 7 %, pero nuestra sensibilidad es superior un 36 %, por lo que superaron ligeramente nuestra propuesta en términos de especificidad a costa de que la sensibilidad sea significativamente más baja.

**Un sistema de audición de máquina para la detección robusta de tos basado en una representación de alto nivel de características de audio específicas de la banda**

En [4], el rango de frecuencia empleado era [0, 4] o [0, 5,125] kHz, dependiendo de los conjuntos de características. Mientras que en nuestro caso se realiza el espectrograma en todo el rango de frecuencias. Este punto confirma de alguna manera la mejor robustez del ruido de este enfoque en comparación con [4]. Cuanto mayor sea el rango de frecuencia, mejor será la caracterización de la tos.

| | SEN | SPE | AUC |
|---|---|---|---|
| **CNN A** | 0,75 | 0,90 | 0,92 |
| **CNN B** | 0,96 | 0,83 | 0,98 |
| **AvgSD [4]** | 0,89 | 0,87 | 0,88 |
| **Supervised BoAW [4]** | 0,83 | 0,79 | 0,81 |

Tabla 7.40: Métricas obtenidas en nuestro estudio con la base de datos “Edimburgo” frente a las obtenidas con el método de [4] con la misma base de datos.

En la Tabla 7.40, podemos comparar las métricas obtenidas en nuestro estudio con respecto a [4], utilizando en ambos métodos la misma base de datos [4]. En este caso observamos que nuestro método utilizando la CNN A tiene mayor especificidad y menor sensibilidad que cualquiera de los métodos de [4].

Mientras que utilizando la CNN B podemos ver que nuestro método es mejor en cualquiera de las métricas observadas con respecto a *Supervised BoAW* [4]. Pero si realizamos la comparativa con *AvgSD* [4] observamos que su especificidad es superior un 4 %, pero nuestra sensibilidad es superior un 7 %, por lo que superaron ligeramente nuestra propuesta en términos de especificidad a costa de que la sensibilidad sea significativamente más baja.

### 7.3.2. Clasificación de tos según la enfermedad subyacente

Para los resultados obtenidos en esta parte de nuestro trabajo realizaremos la comparativa de nuestro trabajo con un trabajo previo en el que se utiliza el sonido de la tos para el diagnóstico precoz de enfermedades respiratorias al igual que buscamos hacer en la segunda parte de nuestro estudio.

**El análisis del sonido de la tos puede diagnosticar rápidamente la neumonía infantil**

En [200] los micrófonos utilizados para la adquisición de los datos se colocaban junto a la cama del paciente, por lo que el ruido adquirido en las señales de audio grabadas es muy bajo con respecto al ruido que tienen los audios de nuestra base de datos. Lo cual puede explicar que su método sea mejor en cualquiera de las métricas observadas.

Dado que en su estudio lo que se busca es diferenciar la neumonía de otras enfermedades respiratorias como bronquiolitis o asma, nosotros solo podemos buscar una "comparativa" cuando intentamos discriminar la tos aguda con respecto a otras enfermedades subyacentes de nuestro estudio. Esto se debe a que la neumonía es un tipo de tos aguda.

| | **SEN** | **SPE** | **ACC** | **PPV** | **NPV** |
|---|---|---|---|---|---|
| **Tos aguda vs EPOC** | 0,76 | 0,62 | 0,69 | 0,68 | 0,75 |
| **Tos aguda vs cáncer de pulmón** | 0,71 | 0,44 | 0,60 | 0,62 | 0,53 |
| **Tos aguda vs no EPOC** | 0,72 | 0,66 | 0,69 | 0,69 | 0,70 |
| **Sólo tos (neumonía vs otras enfermedades) [200]** | 0,94 | 0,75 | 0,88 | 0,89 | 0,86 |
| **Tos y fiebre (neumonía vs otras enfermedades) [200]** | 0,94 | 1 | 0,96 | 1 | 0,89 |

Tabla 7.41: Métricas obtenidas en nuestro estudio frente a las obtenidas en [200].

Como podemos ver en la Tabla 7.41 generalmente su estudio tiene un 20 % más de sensibilidad y un 10 % más de especificidad cuando solo utiliza los audios de tos. Mientras que cuando utiliza además la fiebre como dato para la clasificación su método consigue mejorarse a si mismo en términos de especificidad un 25 %, alcanzando de esta manera el 100 %.

# *Capítulo* 8

# CONCLUSIONES Y LÍNEAS FUTURAS

## 8.1. Conclusiones

El objetivo principal de este trabajo fin de grado fue investigar un nuevo método de audición de máquina para la segmentación automática de eventos de audio de tos, por lo que en este documento, se propone un sistema de audición de máquina para una segmentación robusta de la tos basada únicamente en grabaciones de audio.

El diseño de las características de acuerdo con la naturaleza de la señal y su contexto es un punto crítico para lograr un sistema auditivo de máquina exitoso. Por lo que se optó por el análisis 2D de las representaciones de tiempo y frecuencia de la señal, lo cual mejoró tanto la capacidad de reconocimiento de patrones como la solidez frente al ruido. De esta forma, el sistema caracteriza los patrones de tos extrayendo dichos patrones mediante *Deep Learning* a partir de espectrogramas logarítmicos normalizados.

El sistema de detección de tos se evalúa utilizando una base de datos de veinte pacientes grabados con un *smartphone* que se ubicaba en sus bolsillos o carteras como lo harían normalmente, para obtener muestras de entornos reales y ruidosos que son hostiles a la detección de la tos.

Los resultados en detección de tos confirman que nuestro sistema supera los métodos propuestos hasta ahora en este ámbito. Además, la capacidad de generalización del sistema se evalúa utilizando una estrategia de validación cruzada. Nuestro sistema está alineado con una monitorización del paciente menos perturbadora y más cómoda, que puede beneficiar a los pacientes al permitir el autocontrol de los síntomas de la tos. Además, nuestro sistema tiene potencial para brindar apoyo en la evaluación de tratamientos y una mejor comprensión clínica de los patrones de tos. Los patrones de audio de la tos se pueden detectar y analizar más a fondo para este propósito. Sin embargo, esto podría requerir un paso de preprocesamiento donde los efectos del ruido y otros eventos de audio se minimizarían. Finalmente, los sistemas nacionales de salud y las economías también se beneficiarían con un número reducido de hospitalizaciones y un aumento de la productividad.

Teniendo en cuenta los resultados obtenidos se han podido extraer que las redes neuronales convolucionales pueden resultar una herramienta útil a la hora de encontrar patrones en espectrogramas logarítmicos normalizados diferenciando de esta forma si el audio contiene una tos o no.

Además también tenemos un segundo objetivo que es el sistema de clasificación de enfermedades. Este se evalúa utilizando tres base de datos de 48 pacientes. En este caso al ser de tres bases de datos diferentes los audios fueron adquiridos de diferente forma dependiendo de la base de datos a la que pertenecían. Para dos bases de datos, 35 pacientes fueron grabados con un *smartphone* que se ubicaba en sus bolsillos o carteras como lo harían normalmente. Mientras que en la tercera base de datos había 13 pacientes grabados en tres escenarios ruidosos diferentes.

En este caso los resultados de clasificación de enfermedades no dieron buenos resultados. Pero en el caso de la clasificación de tos aguda frente a otros tipos de tos, vemos una interesante primera aproximación para la clasificación de toses según la enfermedad subyacente.

## 8.2. Líneas futuras

En el desarrollo de este Trabajo Fin de Grado, hemos encontrado ciertas limitaciones, principalmente en el *hardware* empleado y en la extensión de la base de datos que teníamos para realizar este estudio. Por este segundo motivo se buscará aumentar la base de datos principal que es la base de datos "Palencia", para tener más pacientes que aporten sus datos para nuestro estudio. Es importante tener en cuenta que, aunque la cantidad de imágenes que analiza la red es elevada, en realidad el número global de pacientes es limitado.

La limitación de *hardware* hizo que sólo pudiéramos trabajar con espectrogramas con mayor resolución temporal, aumentando el *overlap*, en el estudio de EPOC frente a cáncer de pulmón que es donde menos datos teníamos, observando que gracias a esta modificación las métricas resultantes mejoraban ligeramente. Esto indica que con más capacidad *hardware* podríamos mejorar algunos experimentos, ya que al aumentar la resolución temporal de los espectrogramas nuestra red neuronal sería capaz de extraer más características de interés.

Otra posible modificación de los espectrogramas sería utilizar espectrograma auditivos basado en Gammatone a través de kernels 2D. Este banco de filtros imita la selectividad de frecuencia de la cóclea, por lo que nos parece una línea futura interesante para abordar en una próxima investigación.

Las redes convolucionales realizan la extracción de características de forma automática y oculta, por lo que resultaría interesante ver qué filtros se están activando en cada capa de la red para las diferentes señales de tos y no tos. Estos datos se podía utilizar para intentar dar a los resultados una interpretación médica. Esta interpretación médica se obtendría mediante la búsqueda de explicaciones biológicas y físicas para la diferenciación de los datos a tratar. Esta información podría utilizarse para realizar un preprocesado más eficiente.

La salida de estos detectores de tos podría emplearse fácilmente para detectar el número de eventos de tos (conteo de la tos). Dado el hecho de que la fase explosiva es generalmente más poderosa que otros eventos de primer plano, el conteo de eventos de tos podría reducirse potencialmente al conteo de la fase explosiva. Por otro lado, el análisis de la tos puede requerir una mejora adicional o un proceso de separación de la fuente para evitar que el ruido de fondo afecte los resultados del análisis. Sin embargo, estos procesos adicionales no afectan negativamente las propiedades de valor agregado de las soluciones anteriores.

En definitiva, los resultados obtenidos en este Trabajo Fin de Grado han resultado prometedores para el objetivo que se perseguía, pero aún queda trabajo por hacer para conseguir que la detección de tos mediante esta técnica, pueda ser implantada en nuestros sistemas públicos de salud.

# GLOSARIO

**crepitación** Las crepitaciones son sonidos producidos en distintas situaciones médicas y que permiten el diagnóstico de diversas enfermedades. Se dice de ellos que son similares al ruido que se hace al pisar la nieve, al restregar los cabellos entre los dedos o al echar sal al fuego. Se detectan normalmente mediante el tacto en lugar del oído, debido a su baja intensidad 51

**dendrograma** Un dendrograma es un tipo de representación gráfica o diagrama de datos en forma de árbol, en griego déndron, que organiza los datos en subcategorías que se van dividiendo en otros hasta llegar al nivel de detalle deseado (asemejándose a las ramas de un árbol que se van dividiendo en otras sucesivamente) 18

# ACRÓNIMOS

**ACC** Accuracy 21, 71, 79–81, 83, 85, 88, 89, 91, 93, 94, 97–99, 102–104, 106, 109–111, 113

**ADALINE** ADAptive LINear Elements 28

**ARD** Acute Respiratory Distress 56

**AUC** Area Under the Curve 23, 53, 79–85, 88–95, 97–100, 102–107, 109–112

**BSE-JSRT** Bone Shadow Eliminated - Japanese Society of Radiological Technology 56

**BWH** Brigham and Women's Hospital 55

**CART** Classification And Regression Trees 18

**CNN** Convolutional Neural Network 2, 39, 44–47, 50, 51, 55–57, 61, 65–68, 81, 82, 110–112

**COPD** Chronic Obstructive Pulmonary Disease 71, 83–91, 94–109

**CPU** Central Processing Unit 4

**CT** Computed Tomography 55

**CXR** Chest X-Ray 56

**DFT** Discrete Fourier Transform 65, 66

**DL** Deep Learning 1–3, 5, 9, 17, 27–29, 34, 35, 44, 57, 115

**DNP** Drenaje Nasal Posterior 63

**e.g.** exempli gratia 2, 13, 22, 23, 39, 43, 49, 52–54, 59, 61

**ECLIPSE** Evaluations of COPD Longitudinally to Identify Predictive Surrogate End-points 55, 56

**EDOs** Ecuaciones Diferenciales Ordinarias 4

**ENNS** European Neural Network Society 29

**EPOC** Enfermedad Pulmonar Obstructiva Crónica 1, 2, 50, 52, 54–60, 63, 64, 73–78, 87, 88, 90, 94, 95, 97, 98, 100, 102, 103, 105–107, 109, 113, 116

**ETSIT** Escuela Técnica Superior de Ingenieros de Telecomunicación 5

**FCN** Fully Convolutional Networks 39

**FFT** Fast Fourier Transform 4, 50

**FN** False Negative 19, 24, 25

**FNN** Feedforward Neural Network 35

**FP** False Positive 19, 24, 25

**FPR** False Positive Rate 21

**GAN** Generative Adversarial Net 42

**GPU** Graphics Pprocessing Unit 40, 45–47

**GRU** Gated Recurrent Unit 37

**GS** Gold Standard 20, 21

**GTCC** GammaTone Cepstral Coefficients 16, 49

**HMM** Hidden Markov Model 51, 53, 54

**i.e.** id est 9, 22, 65

**IA** Inteligencia Artificial 5, 7–9, 19, 27–29, 31

**IBM** International Business Machines Corporation 8, 9

**ICANN** International Conference on Artificial Neural Networks 29

**IDE** Integrated Development Environment 4

**IECA** Inhibidores de la Enzima Convertidora de la Angiotensina 63

**IEEE** Institute of Electrical and Electronics Engineers 29

**IJCNN** International Joint Conference on Neuronal Networks 29

**ILSVRC** ImageNet Large Scale Visual Recognition Challenge 44–47

**INNS** International Neural Network Society 29

**IRA** Infecciones Respiratorias Agudas 63, 64

**JSRT** Japanese Society of Radiological Technology 56

**K-NN** K-Nearest Neighbours 16–18, 51, 54

**LCM** Leicester Cough Monitor 59

**LDA** Linear Discriminant Analysis 17, 18

**LPC** Linear Prediction Cepstral 11, 53

**LPCC** Linear Prediction Cepstral Coefficients 13

**LPI** Laboratorio de Procesado de Imagen 4, 5

**LSF** Line Spectral Frequencies 11

**LSTM** Long Short-Term Memory 37

**MFC** Mel-Frequency Cepstrum 55

**MFCC** Mel Frequency Cepstral Coefficients 14, 16, 49–54

**MIT** Massachusetts Institute of Technology 28

**ML** Machine Learning 3, 5, 7, 9, 27, 28, 43

**MLP** MultiLayer Perceptrons 16, 17

**MPEG** Moving Picture Experts Group 10–13

**NASE** Normalized Audio Spectral Envelope 11

**NHLBI** National Heart Lung and Blood Institute 55

**NIN** Network In Network 46

**NIPS** Neural Information Processing Systems 29

**NPV** Negative Predictive Value 21, 71, 79–81, 83, 85, 88, 89, 91, 93, 94, 97–99, 102–104, 106, 109, 113

**NVH** New Victoria Hospital 59

**OMS** Organización Mundial de la Salud 1

**ONN** Octonionic Neural Network 52

**PCA** Principal Component Analysis 53

**PLN** Procesamiento del Lenguaje Natural 8

**PLP** Perceptual Linear Prediction 50

**PNN** Probabilistic Neural Network 53

**PPV** Positive Predictive Value 21, 71, 79–81, 83, 85, 88, 89, 91, 93, 94, 97–99, 102–104, 106, 109, 110, 113

**PSD** Power Spectral Density 11

**QEUH** Queen Elizabeth University Hospital 59

**RAM** Random Access Memory 4

**RASTA-PLP** Relative Spectral Transform and Perceptual Linear Prediction 50

**ReLU** Rectified Linear Unit 30, 39, 45, 50, 56, 67–69

**RGE** Reflujo GastroEsofágico 63

**RNN** Recurrent Neural Networks 36, 37, 47, 50, 51, 111

**ROC** Receiver Operating Characteristic 3, 5, 21–23, 79, 80, 82, 83, 85, 88, 90, 92, 94, 95, 97, 100, 102, 103, 105, 107, 109

**SEN** Sensitivity 21, 71, 79–81, 83, 85, 88, 89, 91, 93, 94, 97–99, 102–104, 106, 109–113

**SGD** Stochastic Gradient Descent 45, 51

**SIF** Spectrogram Image Features 13

**SNR** Signal to Noise Ratio 14, 50, 51

**SPE** Specificity 21, 71, 79–81, 83, 85, 88, 89, 91, 93, 94, 97–99, 102–104, 106, 109–113

**SSCH** Spectral Subband Centroid Histogram 16

**STFT** Short-Time Fourier Transform 50, 65, 66

**SVC** Support Vector Classification 17

**SVM** Support Vector Machine 16, 17, 49–51, 53, 54

**SVR** Support Vector Regression 17

**TAC** Tomografía Axial Computarizada 64

**TDNN** Time Delay Neural Network 52

**TFG** Trabajo de Fin de Grado 1, 5, 16

**TFTD** Transformada de Fourier en Tiempo Discreto 65

**TL** Transfer Learning 44, 45

**TN** True Negative 19, 24, 25

**TNR** True Negative Rate 21

**TP** True Positive 19, 24, 25

**TPR** True Positive Rate 21

**UE** Unión Europea 1, 2, 64

**USA** United States of America 7

**UVa** Universidad de Valladolid 4, 5

# BIBLIOGRAFÍA


[1] M. G. Gadañón, C. G. Peña, J. P. Crespo, R. H. Sánchez, D. Á. González, G. C. G. Tobal, J. G. Pilar, F. V. Villar y P. N. Novo, *Libro de Actas de la "I Jornada para Alumnos de Trabajo Fin de Grado y Trabajo Fin de Máster: Uso Efectivo de Herramientas TIC"*. Universidad de Valladolid, mar. de 2019, ISBN: 978-84-09-10918-0. dirección: http://uvadoc.uva.es/handle/10324/37116.

[2] C. D. Ribas y A. S. Sagardía, «Tos crónica: viejos problemas, nuevas perspectivas», *Revista de asma*, vol. 1, n.º 3, 2016. dirección: http://www.separcontenidos.es/revista3/index.php/revista/article/view/107.

[3] N. Ambrosino, W. Aniwidyaningsih, I. Annesi-Maesano, P. Bakke, F. Blasi, S. Borg, K. Bracke, G. Brusselle, G. Burge, A. Bush, N. Cano, K.-H. Carlsen, N. H. Chavannes, L. Clancy, J.-F. Cordier, U. Costabel, V. Cottin, P. Cullinan, M. Decramer, M. Demedts, S. Drysdale, J. S. Elborn, K. F. Chung, M. Fletcher, J. Gerritsen, G. J. Gibson, A. Greenough, A. Gulsvik, R. P. Gupta, G. Hardavella, D. Heederik, M. Hecker, M. Humbert, N. Künzli, T. Lerut, E. S. Limb, Robert, K. Mayer, G. B. Migliori, N. Navani, B. Nemery, L. Nicod, D. Nowak, S. Olofsson, P. Palange, P. Pelosi, L. Perez, U. Persson, C. Pison, R. Rapp, R. L. Riha, W. Seeger, Y. Sibille, A. K. Simonds, M. Simoni, G. Sotgiu, S. Spiro, I. Steenbruggen, R. Stevenson, D. P. Strachan, J. Svenson, P. Tønnesen, J. Townsend, T. Troosters, S. Turner, P. V. Schil, R. Varraso, G. Viegi, J. A. Wedzicha, T. Welte y S. Williams, *La salud pulmonar en Europa - Hechos y cifras*. European Lung Foundation, 2014, ISBN: 978-1-84984-058-3. dirección: https://www.ers-education.org/home/browse-all-content.aspx?idParent=132227.

[4] J. Monge-Álvarez, C. Hoyos-Barceló, L. M. San-José-Revuelta y P. Casaseca-de-la-Higuera, «A Machine Hearing System for Robust Cough Detection Based on a High-Level Representation of Band-Specific Audio Features», *IEEE Transactions on Biomedical Engineering*, vol. 66, n.º 8, págs. 2319-2330, ago. de 2019, ISSN: 0018-9294. DOI: 10.1109/TBME.2018.2888998.

[5] *Documentación oficial de TensorFlow*, https://www.tensorflow.org/api_docs/python. (visitado 02-09-2019).

[6] *Documentación oficial de Keras*, https://keras.io/. (visitado 02-09-2019).

[7] *Documentación oficial de Numpy*, https://docs.scipy.org/doc/numpy/. (visitado 02-09-2019).

[8] *Documentación oficial de Scipy*, https://docs.scipy.org/doc/scipy/reference/. (visitado 02-09-2019).

[9] *Documentación oficial de Matplotlib*, https://matplotlib.org/api/index.html#the-pyplot-api. (visitado 02-09-2019).

[10] *Documentación oficial de Pandas*, https://pandas-docs.github.io/pandas-docs-travis/reference/index.html. (visitado 02-09-2019).

[11] *Documentación oficial de Scikit-learn*, https://scikit-learn.org/stable/modules/classes.html. (visitado 02-09-2019).

[12] H. Buschmeier y M. Wlodarczak, «TextGridTools: A TextGrid processing and analysis toolkit for Python», en *Tagungsband der 24. Konferenz zur Elektronischen Sprachsignalverarbeitung (ESSV 2013)*, 2013, págs. 152-157. DOI: 10.5281/zenodo.2582132. dirección: https://github.com/hbuschme/TextGridTools.

[13] P. Boersma y D. Weenink, «PRAAT: Doing phonetics by computer (Version 6.0.52)», mayo de 2019. dirección: http://www.fon.hum.uva.nl/praat/.

[14] W. Mora-Flores y A. Borbón-Alpízar, *Edición de Textos Científicos con LATEX. Composición, Diseño Editorial, Gráficos, Inkscape, Tikz y Presentaciones Beamer*, Segunda Edición. Escuela de Matemática, Instituto Tecnológico de Costa Rica., 2012, ISBN: 978-9977-66-227-5.

[15] A. M. Turing, «Computing machinery and intelligence (1950)», *The Essential Turing: The Ideas that Gave Birth to the Computer Age. Ed. B. Jack Copeland. Oxford: Oxford UP*, págs. 433-464, 2004.

[16] R. O. Duda y E. H. Shortliffe, «Expert Systems Research», *Science*, vol. 220, n.º 4594, págs. 261-268, 1983, ISSN: 0036-8075. DOI: 10.1126/science.6340198. eprint: https://science.sciencemag.org/content/220/4594/261.full.pdf. dirección: https://science.sciencemag.org/content/220/4594/261.

[17] P. Lasala Calleja, «Introducción a la Inteligencia Artificial y los Sistemas Expertos», *Ed. Prensas Universitarias de Zaragoza. Zaragoza*, nov. de 1994.

[18] J. Andrés Suárez, «Técnicas de inteligencia artificial aplicadas al análisis de la solvencia empresarial», *Universidad de Oviedo. Facultad de Ciencias Económicas*, 2000. dirección: http://hdl.handle.net/10651/45810.

[19] *Deep Learning, Inteligencia Artificial y Machine Learning, ¿cuáles son las diferencias?*, https://www.blog.andaluciaesdigital.es/deep-learning-inteligencia-artificial-y-machine-learning/, oct. de 2018. (visitado 02-09-2019).

[20] S. J. Russell y P. Norvig, *Artificial Intelligence: A Modern Approach*, ép. Always learning. Pearson, 2016, ISBN: 9781292153964. dirección: https://books.google.es/books?id=XS9CjwEACAAJ.

[21] M. Campbell, A. J. Hoane Jr y F.-h. Hsu, «Deep blue», *Artificial intelligence*, vol. 134, n.º 1-2, págs. 57-83, 2002, ISSN: 0004-3702. DOI: 10.1016/S0004-3702(01)00129-1. dirección: http://www.sciencedirect.com/science/article/pii/S0004370201001291.

[22] B. García Navarro, «Implementación de técnicas de deep learning», Tesis doct., sep. de 2015. dirección: http://riull.ull.es/xmlui/handle/915/1409.

[23] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel y D. Hassabis, «Mastering the game of Go with deep neural networks and tree search», *Nature*, vol. 529, n.º 7587, pág. 484, ene. de 2016. DOI: 10.1038/nature16961. dirección: https://www.nature.com/articles/nature16961#supplementary-information.

[24] R. F. Lyon, «Machine Hearing: An Emerging Field [Exploratory DSP]», *IEEE Signal Processing Magazine*, vol. 27, n.º 5, págs. 131-139, sep. de 2010, ISSN: 1053-5888. DOI: 10.1109/MSP.2010.937498.

[25] F. Alías, J. C. Socoró y X. Sevillano, «A Review of Physical and Perceptual Feature Extraction Techniques for Speech, Music and Environmental Sounds», *Applied Sciences*, vol. 6, n.º 5, 2016, ISSN: 2076-3417. DOI: 10.3390/app6050143. dirección: https://www.mdpi.com/2076-3417/6/5/143.

[26] T. Giannakopoulos y A. Pikrakis, «Chapter 6 - Audio Segmentation», en *Introduction to Audio Analysis*, T. Giannakopoulos y A. Pikrakis, eds., Oxford: Academic Press, 2014, págs. 153-183, ISBN: 978-0-08-099388-1. DOI: 10.1016/B978-0-08-099388-1.00006-6. dirección: http://www.sciencedirect.com/science/article/pii/B9780080993881000066.

[27] C. Kim y R. M. Stern, «Power-normalized Cepstral Coefficients (PNCC) for Robust Speech Recognition», *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 24, n.º 7, págs. 1315-1329, jul. de 2016, ISSN: 2329-9290. DOI: 10.1109/TASLP.2016.2545928.

[28] X. Anguera, S. Bozonnet, N. Evans, C. Fredouille, G. Friedland y O. Vinyals, «Speaker Diarization: A Review of Recent Research», *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, n.º 2, págs. 356-370, feb. de 2012, ISSN: 1558-7916. DOI: 10.1109/TASL.2011.2125954.

[29] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel y P. Ouellet, «Front-End Factor Analysis for Speaker Verification», *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, n.º 4, págs. 788-798, mayo de 2011, ISSN: 1558-7916. DOI: 10.1109/TASL.2010.2064307.

[30] Y. Hu y P. C. Loizou, «Evaluation of Objective Quality Measures for Speech Enhancement», *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, n.º 1, págs. 229-238, ene. de 2008, ISSN: 1558-7916. DOI: 10.1109/TASL.2007.911054.

[31] T. Nakatani, T. Yoshioka, K. Kinoshita, M. Miyoshi y B.-H. Juang, «Speech Dereverberation Based on Variance-Normalized Delayed Linear Prediction», *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, n.º 7, págs. 1717-1731, sep. de 2010, ISSN: 1558-7916. DOI: 10.1109/TASL.2010.2052251.

[32] J. Mekyska, E. Janousova, P. Gomez-Vilda, Z. Smekal, I. Rektorova, I. Eliasova, M. Kostalova, M. Mrackova, J. B. Alonso-Hernandez, M. Faundez-Zanuy y K. López-de-Ipiña, «Robust and complex approach of pathological speech signal analysis», *Neurocomputing*, vol. 167, págs. 94-111, 2015, ISSN: 0925-2312. DOI: 10.1016/j.neucom.2015.02.085. dirección: http://www.sciencedirect.com/science/article/pii/S0925231215007304.

[33] N. Adami, A. Cavallaro, R. Leonardi y P. Migliorati, *Analysis, Retrieval and Delivery of Multimedia Content*, ép. Lecture Notes in Electrical Engineering. Springer New York, 2012, ISBN: 9781461438311. DOI: 10.1007/978-1-4614-3831-1. dirección: https://books.google.es/books?id=Mdp%5C_0zPq3lQC.

[34] T. Giannakopoulos y A. Pikrakis, «Chapter 8 - Music Information Retrieval», en *Introduction to Audio Analysis*, T. Giannakopoulos y A. Pikrakis, eds., Oxford: Academic Press, 2014, págs. 211-231, ISBN: 978-0-08-099388-1. DOI: 10.1016/B978-0-08-099388-1.00008-X. dirección: http://www.sciencedirect.com/science/article/pii/B978008099388100008X.

[35] M. Müller, *Information Retrieval for Music and Motion*. Springer Berlin Heidelberg, 2007, ISBN: 9783540740483. DOI: 10.1007/978-3-540-74048-3. dirección: https://books.google.es/books?id=kSzeZWR2yDsC.

[36] A. Divakaran, *Multimedia Content Analysis: Theory and Applications*, ép. Signals and Communication Technology. Springer Science & Business Media, 2009, ISBN: 9780387765693. dirección: https://books.google.es/books?id=nkjUHeuzQTIC.

[37] A. L.-c. Wang, «An Industrial Strength Audio Search Algorithm.», en *ISMIR 2003, 4th Symposium Conference on Music Information Retrieval*, Washington, DC, vol. 2003, 2003, págs. 7-13.

[38] E. Benetos y C. Kotropoulos, «Non-Negative Tensor Factorization Applied to Music Genre Classification», *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, n.º 8, págs. 1955-1967, nov. de 2010, ISSN: 1558-7916. DOI: 10.1109/TASL.2010.2040784.

[39] D. Giannoulis y A. Klapuri, «Musical Instrument Recognition in Polyphonic Audio Using Missing Feature Approach», *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, n.º 9, págs. 1805-1817, sep. de 2013, ISSN: 1558-7916. DOI: 10.1109/TASL.2013.2248720.

[40] Y. Zhu y D. Shasha, «Warping Indexes with Envelope Transforms for Query by Humming», en *Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data*, ép. SIGMOD '03, San Diego, California: ACM, 2003, págs. 181-192, ISBN: 1-58113-634-X. DOI: 10.1145/872757.872780. dirección: http://doi.acm.org/10.1145/872757.872780.

[41] Z.-S. Chen, J.-S. R. Jang y C.-H. Lee, «A Kernel Framework for Content-Based Artist Recommendation System in Music», *IEEE Transactions on Multimedia*, vol. 13, n.º 6, págs. 1371-1380, dic. de 2011, ISSN: 1520-9210. DOI: 10.1109/TMM.2011.2166380.

[42] S. Chu, S. Narayanan y C.-C. J. Kuo, «Environmental Sound Recognition With Time–Frequency Audio Features», *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, n.º 6, págs. 1142-1158, ago. de 2009, ISSN: 1558-7916. DOI: 10.1109/TASL.2009.2017438.

[43] D. Barchiesi, D. Giannoulis, D. Stowell y M. D. Plumbley, «Acoustic Scene Classification: Classifying environments from the sounds they produce», *IEEE Signal Processing Magazine*, vol. 32, n.º 3, págs. 16-34, mayo de 2015, ISSN: 1053-5888. DOI: 10.1109/MSP.2014.2326181.

[44] R. Naves, B. H. Barbosa y D. D. Ferreira, «Classification of lung sounds using higher-order statistics: A divide-and-conquer approach», *Computer Methods and Programs in Biomedicine*, vol. 129, págs. 12-20, 2016, ISSN: 0169-2607. DOI: https://doi.org/10.1016/j.cmpb.2016.02.013. dirección: http://www.sciencedirect.com/science/article/pii/S0169260716301614.

[45] A. Rabaoui, M. Davy, S. Rossignol y N. Ellouze, «Using One-Class SVMs and Wavelets for Audio Surveillance», *IEEE Transactions on Information Forensics and Security*, vol. 3, n.º 4, págs. 763-775, dic. de 2008, ISSN: 1556-6013. DOI: 10.1109/TIFS.2008.2008216.

[46] E. Babaee, N. B. Anuar, A. W. A. Wahab, S. Shamshirband y A. T. Chronopoulos, «An Overview of Audio Event Detection Methods from Feature Extraction to Classification», *Applied Artificial Intelligence*, vol. 31, n.º 9-10, págs. 661-714, 2017. DOI: 10.1080/08839514.2018.1430469.

[47] T. Giannakopoulos y A. Pikrakis, «Chapter 4 - Audio Features», en *Introduction to Audio Analysis*, T. Giannakopoulos y A. Pikrakis, eds., Oxford: Academic Press, 2014, págs. 59-103, ISBN: 978-0-08-099388-1. DOI: 10.1016/B978-0-08-099388-1.00004-2. dirección: http://www.sciencedirect.com/science/article/pii/B9780080993881000042.

[48] Jhing-Fa Wang, Jia-Ching Wang, Tze-Hsuan Huang y Cheng-Shu Hsu, «Home environmental sound recognition based on MPEG-7 features», en *2003 46th Midwest Symposium on Circuits and Systems*, IEEE, vol. 2, dic. de 2003, 682-685 Vol. 2. DOI: 10.1109/MWSCAS.2003.1562378.

[49] Y. V. S. Murthy y S. G. Koolagudi, «Classification of vocal and non-vocal regions from audio songs using spectral features and pitch variations», en *2015 IEEE 28th Canadian Conference on Electrical and Computer Engineering (CCECE)*, IEEE, mayo de 2015, págs. 1271-1276. DOI: 10.1109/CCECE.2015.7129461.

[50] J. Kreiman y B. R. Gerratt, «Perception of aperiodicity in pathological voice», *The Journal of the Acoustical Society of America*, vol. 117, n.º 4, págs. 2201-2211, 2005. DOI: 10.1121/1.1858351.

[51] H.-G. Kim, N. Moreau y T. Sikora, *MPEG-7 Audio and Beyond: Audio Content Indexing and Retrieval*. John Wiley & Sons, 2006, ISBN: 9780470093351. dirección: https://books.google.es/books?id=eQM%5C_Ip2nN8YC.

[52] T. Zhang y C.-C. J. Kuo, «Audio content analysis for online audiovisual data segmentation and classification», *IEEE Transactions on Speech and Audio Processing*, vol. 9, n.º 4, págs. 441-457, mayo de 2001, ISSN: 1063-6676. DOI: 10.1109/89.917689.

[53] M. Wisniewski y T. P. Zielinski, «Application of tonal index to pulmonary wheezes detection in asthma monitoring», en *2011 19th European Signal Processing Conference*, IEEE, ago. de 2011, págs. 1544-1548.

[54] J. Poza, R. Hornero, J. Escudero, A. Fernández y C. I. Sánchez, «Regional Analysis of Spontaneous MEG Rhythms in Patients with Alzheimer's Disease Using Spectral Entropies», *Annals of Biomedical Engineering*, vol. 36, n.º 1, págs. 141-152, ene. de 2008, ISSN: 1573-9686. DOI: 10.1007/s10439-007-9402-y.

[55] H.-G. Kim, N. Moreau y T. Sikora, «Audio classification based on MPEG-7 spectral basis representations», *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, n.º 5, págs. 716-725, mayo de 2004, ISSN: 1051-8215. DOI: 10.1109/TCSVT.2004.826766.

[56] C.-H. Lee, J.-L. Shih, K.-M. Yu y H.-S. Lin, «Automatic Music Genre Classification Based on Modulation Spectral Analysis of Spectral and Cepstral Features», *IEEE Transactions on Multimedia*, vol. 11, n.º 4, págs. 670-682, jun. de 2009, ISSN: 1520-9210. DOI: 10.1109/TMM.2009.2017635.

[57] Y. Zhu y M. S. Kankanhalli, «Precise pitch profile feature extraction from musical audio for key detection», *IEEE Transactions on Multimedia*, vol. 8, n.º 3, págs. 575-584, jun. de 2006, ISSN: 1520-9210. DOI: 10.1109/TMM.2006.870727.

[58] M. R. Schroeder, «Period Histogram and Product Spectrum: New Methods for Fundamental-Frequency Measurement», *The Journal of the Acoustical Society of America*, vol. 43, n.º 4, págs. 829-834, 1968. DOI: 10.1121/1.1910902.

[59] G. Muhammad y K. Alghathbar, «Environment Recognition from Audio Using MPEG-7 Features», en *2009 Fourth International Conference on Embedded and Multimedia Computing*, IEEE, dic. de 2009, págs. 1-6. DOI: 10.1109/EM-COM.2009.5402978.

[60] A. S. Lampropoulos y G. A. Tsihrintzis, «Evaluation of MPEG-7 Descriptors for Speech Emotional Recognition», en *2012 Eighth International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, IEEE, jul. de 2012, págs. 98-101. DOI: 10.1109/IIH-MSP.2012.29.

[61] M. Farrús, J. Hernando y P. Ejarque, «Jitter and shimmer measurements for speaker recognition», en *Eighth annual conference of the international speech communication association*, 2007.

[62] M. Muller, F. Kurth y M. Clausen, «Chroma-based statistical audio features for audio matching», en *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, 2005.*, IEEE, oct. de 2005, págs. 275-278. DOI: 10.1109/ASPAA.2005.1540223.

[63] D. Arifianto, «Enhancement of speech over wireless network using sinusoidal modeling and synthesis», en *SiPS 2013 Proceedings*, IEEE, oct. de 2013, págs. 301-305. DOI: 10.1109/SiPS.2013.6674523.

[64] S. Dimri, S. Singh, A. Kaur y M. K. Dutta, «A Robust Watermarking Algorithm Based on Multi Resolution Decomposition of Audio Signal», en *2012 Third International Conference on Computer and Communication Technology*, IEEE, nov. de 2012, págs. 299-302. DOI: 10.1109/ICCCT.2012.67.

[65] Y.-H. Yang, Y.-C. Lin, Y.-F. Su y H. H. Chen, «A Regression Approach to Music Emotion Recognition», *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, n.º 2, págs. 448-457, feb. de 2008, ISSN: 1558-7916. DOI: 10.1109/TASL.2007.911513.

[66] A. Pinto, «Indexing melodic sequences via Wavelet transform», en *2009 IEEE International Conference on Multimedia and Expo*, IEEE, jun. de 2009, págs. 882-885. DOI: 10.1109/ICME.2009.5202636.

[67] X. Xu, H. Wang, R. Fan y X. Li, «An application of lossy compression of the echolocation signals of toothed whale», en *2017 International Conference on Wavelet Analysis and Pattern Recognition (ICWAPR)*, IEEE, jul. de 2017, págs. 7-12. DOI: 10.1109/ICWAPR.2017.8076654.

[68] S. Moussa, Z. Hajaiej y A. Garsallah, «Decomposition of a speech signal wavelet packet using an entropy criterion», en *2017 International Conference on Advanced Systems and Electric Technologies (IC_ASET)*, IEEE, ene. de 2017, págs. 351-355. DOI: 10.1109/ASET.2017.7983718.

[69] G. Luo, «An Efficient DSP Implantation of Wavelet Audio Coding for Digital Communication», en *2010 Fourth International Conference on Digital Society*, feb. de 2010, págs. 66-71. DOI: 10.1109/ICDS.2010.20.

[70] R. SantÁna, R. Coelho y A. Alcaim, «Text-independent speaker recognition based on the Hurst parameter and the multidimensional fractional Brownian motion model», *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, n.º 3, págs. 931-940, mayo de 2006, ISSN: 1558-7916. DOI: 10.1109/TSA.2005.858054.

[71] L. Zão, R. Coelho y P. Flandrin, «Speech Enhancement with EMD and Hurst-Based Mode Selection», *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, n.º 5, págs. 899-911, mayo de 2014, ISSN: 2329-9290. DOI: 10.1109/TASLP.2014.2312541.

[72] E. Dranka y R. Coelho, «Robust Maximum Likelihood Acoustic Energy Based Source Localization in Correlated Noisy Sensing Environments», *IEEE Journal of Selected Topics in Signal Processing*, vol. 9, n.º 2, págs. 259-267, mar. de 2015, ISSN: 1932-4553. DOI: 10.1109/JSTSP.2014.2385657.

[73] S. G. Mallat y Z. Zhang, «Matching pursuits with time-frequency dictionaries», *IEEE Transactions on Signal Processing*, vol. 41, n.º 12, págs. 3397-3415, dic. de 1993, ISSN: 1053-587X. DOI: 10.1109/78.258082.

[74] P. J. Wolfe, S. J. Godsill y M. Dorfler, «Multi-Gabor dictionaries for audio time-frequency analysis», en *Proceedings of the 2001 IEEE Workshop on the Applications of Signal Processing to Audio and Acoustics (Cat. No.01TH8575)*, IEEE, oct. de 2001, págs. 43-46. DOI: 10.1109/ASPAA.2001.969538.

[75] B. T. Meyer y B. Kollmeier, «Optimization and evaluation of Gabor feature sets for ASR», en *Ninth Annual Conference of the International Speech Communication Association*, vol. 2, sep. de 2008, págs. 906-909, ISBN: 978-1-615-67378-0. dirección: https://www.isca-speech.org/archive/interspeech_2008/i08_0906.html.

[76] J.-C. Wang, C.-H. Lin, B.-W. Chen y M.-K. Tsai, «Gabor-Based Nonuniform Scale-Frequency Map for Environmental Sound Classification in Home Automation», *IEEE Transactions on Automation Science and Engineering*, vol. 11, n.º 2, págs. 607-613, abr. de 2014, ISSN: 1545-5955. DOI: 10.1109/TASE.2013.2285131.

[77] X.-Y. Zhang y Q.-H. He, «Time–frequency audio feature extraction based on tensor representation of sparse coding», English, *Electronics Letters*, vol. 51, n.º 2, págs. 131-132, ene. de 2015, ISSN: 0013-5194. DOI: 10.1049/el.2014.3333. dirección: https://digital-library.theiet.org/content/journals/10.1049/el.2014.3333.

[78] J. Dennis, H. D. Tran y H. Li, «Spectrogram Image Feature for Sound Event Classification in Mismatched Conditions», *IEEE Signal Processing Letters*, vol. 18, n.º 2, págs. 130-133, feb. de 2011, ISSN: 1070-9908. DOI: 10.1109/LSP.2010.2100380.

[79] R. V. Sharan y T. J. Moir, «Noise robust audio surveillance using reduced spectrogram image feature and one-against-all SVM», *Neurocomputing*, vol. 158, págs. 90-99, 2015, ISSN: 0925-2312. DOI: 10.1016/j.neucom.2015.02.001. dirección: http://www.sciencedirect.com/science/article/pii/S0925231215001113.

[80] R. J. Mammone, X. Zhang y R. P. Ramachandran, «Robust speaker recognition: a feature-based approach», *IEEE Signal Processing Magazine*, vol. 13, n.º 5, págs. 58-, sep. de 1996, ISSN: 1053-5888. DOI: 10.1109/79.536825.

[81] B. S. Atal, «Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification», *The Journal of the Acoustical Society of America*, vol. 55, n.º 6, págs. 1304-1312, 1974. DOI: 10.1121/1.1914702.

[82] C. Xu, N. C. Maddage y X. Shao, «Automatic music classification and summarization», *IEEE Transactions on Speech and Audio Processing*, vol. 13, n.º 3, págs. 441-450, mayo de 2005, ISSN: 1063-6676. DOI: 10.1109/TSA.2004.840939.

[83] C. L. Nikias, «Higher-order spectral analysis», en *Proceedings of the 15th Annual International Conference of the IEEE Engineering in Medicine and Biology Societ*, IEEE, oct. de 1993, págs. 319-319. DOI: 10.1109/IEMBS.1993.978564.

[84] A. C. Lindgren, M. T. Johnson y R. J. Povinelli, «Speech recognition using reconstructed phase space features», en *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03).*, IEEE, vol. 1, abr. de 2003, págs. I-60. DOI: 10.1109/ICASSP.2003.1198716.

[85] R. Hegger, H. Kantz y L. Matassini, «Denoising Human Speech Signals Using Chaoslike Features», *Physical Review Letters*, vol. 84, págs. 3197-3200, 14 abr. de 2000. DOI: 10.1103/PhysRevLett.84.3197. dirección: https://link.aps.org/doi/10.1103/PhysRevLett.84.3197.

[86] R. Behroozmand, F. Almasganj y M. H. Moradi, «Pathological Assesment of Vocal Fold Nodules and Polyp Using Accoustic Perturbation and Phase Space Features», en *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, IEEE, vol. 2, mayo de 2006, págs. II-II. DOI: 10.1109/ICASSP.2006.1660528.

[87] Doh-Suk Kim, Jae-Hoon Jeong, Jae-Weon Kim y Soo-Young Lee, «Feature extraction based on zero-crossings with peak amplitudes for robust speech recognition in noisy environments», en *1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings*, IEEE, vol. 1, mayo de 1996, 61-64 vol. 1. DOI: 10.1109/ICASSP.1996.540290.

[88] S. Stevens, J. Volkmann y E. Newman, «The mel scale equates the magnitude of perceived differences in pitch at different frequencies», *Journal of the Acoustical Society of America*, vol. 8, n.º 3, págs. 185-190, 1937.

[89] M. Ghulam, J. Horikawa y T. Nitta, «A Pitch-Synchronous Peak-Amplitude Based Feature Extraction Method for Noise Robust ASR», en *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, IEEE, vol. 1, mayo de 2006, págs. I-I. DOI: 10.1109/ICASSP.2006.1660068.

[90] Y. Ando, «Autocorrelation-based features for speech representation», *Proceedings of Meetings on Acoustics*, vol. 19, n.º 1, pág. 060 033, 2013. DOI: 10.1121/1.4795143.

[91] X. Valero y F. Alías, «Narrow-band autocorrelation function features for the automatic recognition of acoustic environments», *The Journal of the Acoustical Society of America*, vol. 134, n.º 1, págs. 880-890, 2013. DOI: 10.1121/1.4807807.

[92] S. Greenberg y B. E. D. Kingsbury, «The modulation spectrogram: in pursuit of an invariant representation of speech», en *1997 IEEE International Conference on Acoustics, Speech, and Signal Processing*, IEEE, vol. 3, abr. de 1997, 1647-1650 vol.3. DOI: 10.1109/ICASSP.1997.598826.

[93] W. E. I. Wei, Q.-S. Xie y Q.-J. Chen, «SNR classification based on amplitude modulation spectrogram via deep belief networks», en *2016 IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)*, IEEE, oct. de 2016, págs. 1834-1840. DOI: 10.1109/IMCEC.2016.7867536.

[94] A. Chittora y H. A. Patil, «Classification of pathological infant cries using modulation spectrogram features», en *The 9th International Symposium on Chinese Spoken Language Processing*, IEEE, sep. de 2014, págs. 541-545. DOI: 10.1109/ISCSLP.2014.6936626.

[95] T. Barker y T. Virtanen, «Blind Separation of Audio Mixtures Through Nonnegative Tensor Factorization of Modulation Spectrograms», *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, n.º 12, págs. 2377-2389, dic. de 2016, ISSN: 2329-9290. DOI: 10.1109/TASLP.2016.2602546.

[96] R. V. Sharan y T. J. Moir, «An overview of applications and advancements in automatic sound recognition», *Neurocomputing*, vol. 200, págs. 22-34, 2016, ISSN: 0925-2312. DOI: 10.1016/j.neucom.2016.03.020. dirección: http://www.sciencedirect.com/science/article/pii/S0925231216300406.

[97] G. Richard, S. Sundaram y S. Narayanan, «An Overview on Perceptually Motivated Audio Indexing and Classification», *Proceedings of the IEEE*, vol. 101, n.º 9, págs. 1939-1954, sep. de 2013, ISSN: 0018-9219. DOI: 10.1109/JPROC.2013.2251591.

[98] R. Meddis y L. O'Mard, «A unitary model of pitch perception», *The Journal of the Acoustical Society of America*, vol. 102, n.º 3, págs. 1811-1820, 1997. DOI: 10.1121/1.420088.

[99] M. Slaney y R. F. Lyon, «A perceptual pitch detector», en *International Conference on Acoustics, Speech, and Signal Processing*, IEEE, abr. de 1990, 357-360 vol.1. DOI: 10.1109/ICASSP.1990.115684.

[100] F. Morchen, A. Ultsch, M. Thies e I. Lohken, «Modeling timbre distance with temporal statistics from polyphonic music», *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, n.º 1, págs. 81-90, ene. de 2006, ISSN: 1558-7916. DOI: 10.1109/TSA.2005.860352.

[101] S. Pfeiffer, R. Lienhart y W. Effelsberg, «Scene Determination Based on Video and Audio Features», *Multimedia Tools and Applications*, vol. 15, n.º 1, págs. 59-81, sep. de 2001, ISSN: 1573-7721. DOI: 10.1023/A:1011315803415.

[102] M. Mckinney y J. Breebaart, «Features for Audio and Music Classification», en *Proceedings of the International Symposium on Music Information Retrieval*, Johns Hopkins University, 2003, págs. 151-158. dirección: http://jhir.library.jhu.edu/handle/1774.2/22.

[103] B. Gerazov y Z. Ivanovski, «Kernel Power Flow Orientation Coefficients for Noise-robust Speech Recognition», *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 23, n.º 2, págs. 407-419, feb. de 2015, ISSN: 2329-9290. DOI: 10.1109/TASLP.2014.2384274.

[104] X. Valero y F. Alías, «Gammatone Wavelet features for sound classification in surveillance applications», en *2012 Proceedings of the 20th European Signal Processing Conference (EUSIPCO)*, IEEE, ago. de 2012, págs. 1658-1662.

[105] J. Dennis, H. D. Tran y H. Li, «Combining robust spike coding with spiking neural networks for sound event classification», en *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, abr. de 2015, págs. 176-180. DOI: 10.1109/ICASSP.2015.7177955.

[106] J. Monge-Álvarez, C. Hoyos-Barceló, P. Lesso y P. Casaseca-de-la-Higuera, «Robust Detection of Audio-Cough Events Using Local Hu Moments», *IEEE Journal of Biomedical and Health Informatics*, vol. 23, n.º 1, págs. 184-196, ene. de 2019, ISSN: 2168-2194. DOI: 10.1109/JBHI.2018.2800741.

[107] M. Malekesmaeili y R. K. Ward, «A local fingerprinting approach for audio copy detection», *Signal Processing*, vol. 98, págs. 308-321, 2014, ISSN: 0165-1684. DOI: 10.1016/j.sigpro.2013.11.023. dirección: http://www.sciencedirect.com/science/article/pii/S0165168413004593.

[108] X. Valero y F. Alias, «Gammatone Cepstral Coefficients: Biologically Inspired Features for Non-Speech Audio Classification», *IEEE Transactions on Multimedia*, vol. 14, n.º 6, págs. 1684-1689, dic. de 2012, ISSN: 1520-9210. DOI: 10.1109/TMM.2012.2199972.

[109] B. Gajic y K. K. Paliwal, «Robust speech recognition in noisy environments based on subband spectral centroid histograms», *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, n.º 2, págs. 600-608, mar. de 2006, ISSN: 1558-7916. DOI: 10.1109/TSA.2005.855834.

[110] H. Hermansky y N. Morgan, «RASTA processing of speech», *IEEE Transactions on Speech and Audio Processing*, vol. 2, n.º 4, págs. 578-589, oct. de 1994, ISSN: 1063-6676. DOI: 10.1109/89.326616.

[111] T. Giannakopoulos y A. Pikrakis, «Chapter 5 - Audio Classification», en *Introduction to Audio Analysis*, T. Giannakopoulos y A. Pikrakis, eds., Oxford: Academic Press, 2014, págs. 107-151, ISBN: 978-0-08-099388-1. DOI: 10.1016/B978-0-08-099388-1.00005-4. dirección: http://www.sciencedirect.com/science/article/pii/B9780080993881000054.

[112] P. Khunarsal, C. Lursinsap y T. Raicharoen, «Very short time environmental sound classification based on spectrogram pattern matching», *Information Sciences*, vol. 243, págs. 57-74, 2013, ISSN: 0020-0255. DOI: 10.1016/j.ins.2013.04.014. dirección: http://www.sciencedirect.com/science/article/pii/S0020025513003113.

[113] V. Mitra y C.-J. Wang, «Content based audio classification: a neural network approach», *Soft Computing*, vol. 12, n.º 7, págs. 639-646, mayo de 2008, ISSN: 1433-7479. DOI: 10.1007/s00500-007-0241-4.

[114] Y. M. Costa, L. Oliveira, A. L. Koerich, F. Gouyon y J. Martins, «Music genre classification using LBP textural features», *Signal Processing*, vol. 92, n.º 11, págs. 2723-2737, 2012, ISSN: 0165-1684. DOI: 10.1016/j.sigpro.2012.04.023. dirección: http://www.sciencedirect.com/science/article/pii/S0165168412001478.

[115] C.-H. Lee, C.-H. Chou, C.-C. Han y R.-Z. Huang, «Automatic recognition of animal vocalizations using averaged MFCC and linear discriminant analysis», *Pattern Recognition Letters*, vol. 27, n.º 2, págs. 93-101, 2006, ISSN: 0167-8655. DOI: 10.1016/j.patrec.2005.07.004. dirección: http://www.sciencedirect.com/science/article/pii/S0167865505001959.

[116] E. Dafna, A. Tarasiuk e Y. Zigel, «Automatic Detection of Whole Night Snoring Events Using Non-Contact Microphone», *PLOS ONE*, vol. 8, n.º 12, págs. 1-14, dic. de 2014. DOI: 10.1371/journal.pone.0084139.

[117] E. Pomponi y A. Vinogradov, «A real-time approach to acoustic emission clustering», *Mechanical Systems and Signal Processing*, vol. 40, n.º 2, págs. 791-804, 2013, ISSN: 0888-3270. DOI: 10.1016/j.ymssp.2013.03.017. dirección: http://www.sciencedirect.com/science/article/pii/S0888327013001179.

[118] F. Saki y N. Kehtarnavaz, «Real-Time Unsupervised Classification of Environmental Noise Signals», *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, n.º 8, págs. 1657-1667, ago. de 2017, ISSN: 2329-9290. DOI: 10.1109/TASLP.2017.2711059.

[119] J.-H. Choi y J.-H. Chang, «On using acoustic environment classification for statistical model-based speech enhancement», *Speech Communication*, vol. 54, n.º 3, págs. 477-490, 2012, ISSN: 0167-6393. DOI: 10.1016/j.specom.2011.10.009. dirección: http://www.sciencedirect.com/science/article/pii/S0167639311001579.

[120] D. Neiberg, G. Salvi y J. Gustafson, «Semi-supervised methods for exploring the acoustics of simple productive feedback», *Speech Communication*, vol. 55, n.º 3, págs. 451-469, 2013, ISSN: 0167-6393. DOI: 10.1016/j.specom.2012.12.007. dirección: http://www.sciencedirect.com/science/article/pii/S0167639313000022.

[121] Y. Song y C. Zhang, «Content-Based Information Fusion for Semi-Supervised Music Genre Classification», *IEEE Transactions on Multimedia*, vol. 10, n.º 1, págs. 145-152, ene. de 2008, ISSN: 1520-9210. DOI: 10.1109/TMM.2007.911305.

[122] J. Deng, X. Xu, Z. Zhang, S. Frühholz y B. Schuller, «Semisupervised Autoencoders for Speech Emotion Recognition», *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, n.º 1, págs. 31-43, ene. de 2018, ISSN: 2329-9290. DOI: 10.1109/TASLP.2017.2759338.

[123] J. Giacomantone, M. L. Violini, L. Lorenti, M. Naiouf, O. N. Bria, M. J. Abásolo Guerrero y C. Manresa Yee, «Reconocimiento Automático de Patrones, Análisis de Imágenes y Generación de Características», es, en *XV Workshop de Investigadores en Ciencias de la Computación*, Red de Universidades con Carreras en Informática (RedUNCI), abr. de 2013, págs. 735-739. dirección: http://sedici.unlp.edu.ar/handle/10915/27517.

[124] S. Gergen, A. Nagathil y R. Martin, «Classification of reverberant audio signals using clustered ad hoc distributed microphones», *Signal Processing*, vol. 107, págs. 21-32, 2015, Special Issue on ad hoc microphone arrays and wireless acoustic sensor networks Special Issue on Fractional Signal Processing and Applications, ISSN: 0165-1684. DOI: 10.1016/j.sigpro.2014.04.034. dirección: http://www.sciencedirect.com/science/article/pii/S0165168414002151.

[125] O. Banos, K. Konsolakis, H. op den Akker, C. Pelachaud y R. Bangalore, «Council of Coaches: Methods for inferring short-term behaviour from multimodal sensor data», ago. de 2018. dirección: https://ec.europa.eu/research/participants/documents/downloadPublic?documentIds=080166e5bd4ea421&appId=PPGMS.

[126] Lavender888000, *Confusion Matrix, Table of Confusion, Preventive Medicine*, https://commons.wikimedia.org/wiki/File:Preventive_Medicine_-_Statistics_Sensitivity_TPR,_Specificity_TNR,_PPV,_NPV,_FDR,_FOR,_ACCuracy,_Likelihood_Ratio,_Diagnostic_Odds_Ratio_2_Final_wiki.png?uselang=es, ago. de 2015. (visitado 02-09-2019).

[127] F. Salech, V. Mery, F. Larrondo y G. Rada, «Estudios que evalúan un test diagnóstico: interpretando sus resultados», es, *Revista médica de Chile*, vol. 136, págs. 1208-1208, sep. de 2008, ISSN: 0034-9887. DOI: 10.4067/S0034-98872008000900018. dirección: https://scielo.conicyt.cl/scielo.php?script=sci_arttext&pid=S0034-98872008000900018&nrm=iso.

[128] J. Cerda y L. Cifuentes, «Uso de curvas ROC en investigación clínica: Aspectos teórico-prácticos», es, *Revista chilena de infectología*, vol. 29, n.º 2, págs. 138-141, abr. de 2012, ISSN: 0716-1018. DOI: 10.4067/S0716-10182012000200003. dirección: https://scielo.conicyt.cl/scielo.php?script=sci_arttext&pid=S0716-10182012000200003&nrm=iso.

[129] J. G. Abad, «Calibración local de predicciones numéricas de viento con tecnicas estadísticas no lineales.(Downscaling Estadístico)», Tesis de mtría., oct. de 2012. dirección: http://hdl.handle.net/10902/1006.

[130] F. Herrera, C. Hervás, J. Otero y L. Sánchez, «Un estudio empírico preliminar sobre los tests estadísticos más habituales en el aprendizaje automático», *Tendencias de la Minería de Datos en España, Red Española de Minería de Datos y Aprendizaje (TIC2002-11124-E)*, págs. 403-412, 2004. dirección: https://www.lsi.us.es/redmidas/Capitulos/LMD35.pdf.

[131] L. Pérez Planells, J. Delegido, J. P. Rivera-Caicedo y J. Verrelst, «Análisis de métodos de validación cruzada para la obtención robusta de parámetros biofísicos», *Revista Española de Teledetección*, vol. 44, págs. 55-65, 2015. dirección: https://core.ac.uk/download/pdf/71051261.pdf.

[132] V. Van Asch, «Macro-and micro-averaged evaluation measures [[basic draft]]», *Belgium: CLiPS*, págs. 1-27, sep. de 2013. dirección: https://pdfs.semanticscholar.org/1d10/6a2730801b6210a67f7622e4d192pdf.

[133] P. A. Caballero, «Ayuda al diagnóstico del TDAH en la infancia mediante técnicas de procesado de señal y aprendizaje», Tesis doct., abr. de 2019.

[134] W. S. McCulloch y W. Pitts, «A logical calculus of the ideas immanent in nervous activity», *The bulletin of mathematical biophysics*, vol. 5, n.º 4, págs. 115-133, dic. de 1943, ISSN: 1522-9602. DOI: 10.1007/BF02478259. dirección: https://link.springer.com/article/10.1007/BF02478259.

[135] W. Pitts y W. S. McCulloch, «How we know universals the perception of auditory and visual forms», *The bulletin of mathematical biophysics*, vol. 9, n.º 3, págs. 127-147, sep. de 1947, ISSN: 1522-9602. DOI: 10.1007/BF02478291. dirección: https://link.springer.com/article/10.1007/BF02478291.

[136] D. O. Hebb, *The Organization of Behavior: A Neuropsychological Theory*. Psychology Press, 2005, ISBN: 9781135631901. dirección: https://books.google.es/books?id=ddB4AgAAQBAJ.

[137] B. Widrow y M. E. Hoff, «Adaptive switching circuits», Stanford University California. Stanford Electronics Laboratories et. al., inf. téc., jun. de 1960. dirección: https://apps.dtic.mil/dtic/tr/fulltext/u2/241531.pdf.

[138] M. Minsky, S. Papert y L. Bottou, *Perceptrons: An Introduction to Computational Geometry*, ép. The MIT Press. MIT Press, 2017, ISBN: 9780262534772. dirección: https://books.google.es/books?id=PLQ5DwAAQBAJ.

[139] J. A. Anderson, J. W. Silverstein, S. A. Ritz y R. S. Jones, «Distinctive features, categorical perception, and probability learning: Some applications of a neural model.», *Psychological Review*, vol. 84, n.º 5, págs. 413-451, sep. de 1977, ISSN: 1939-1471(Electronic), 0033-295X(Print). DOI: 10.1037/0033-295X.84.5.413.

[140] J. J. Hopfield, «Neural networks and physical systems with emergent collective computational abilities», *Proceedings of the National Academy of Sciences*, vol. 79, n.º 8, págs. 2554-2558, 1982, ISSN: 0027-8424. DOI: 10.1073/pnas.79.8.2554. eprint: https://www.pnas.org/content/79/8/2554.full.pdf. dirección: https://www.pnas.org/content/79/8/2554.

[141] P. J. Werbos, *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences*. Harvard University, 1975. dirección: https://books.google.es/books?id=z81XmgEACAAJ.

[142] D. B. Parker, *Learning-logic: Casting the Cortex of the Human Brain in Silicon*, ép. Technical report: Center for Computational Research in Economics and Management Science. Massachusetts Institute of Technology, Center for Computational Research in Economics y Management Science, 1985. dirección: https://books.google.es/books?id=2kS9GwAACAAJ.

[143] Y. Le Cun, «Learning Process in an Asymmetric Threshold Network», en *Disordered Systems and Biological Organization*, E. Bienenstock, F. F. Soulié y G. Weisbuch, eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 1986, págs. 233-240, ISBN: 978-3-642-82657-3. DOI: 10.1007/978-3-642-82657-3_24.

[144] D. E. Rumelhart, G. E. Hinton y R. J. Williams, «Learning internal representations by error propagation», Institute for Cognitive Science, University of California San Diego, La Jolla, inf. téc., 1985. dirección: https://apps.dtic.mil/dtic/tr/fulltext/u2/a164453.pdf.

[145] J. L. McClelland y D. E. Rumelhart, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition : Foundations, Psychological and Biological Models*, ép. Bradford Book. MIT press, 1987, vol. 2, ISBN: 9780262631129. dirección: https://books.google.es/books?id=YUhGwgEACAAJ.

[146] T. Hrycej, *Modular Learning in Neural Networks: A Modularized Approach to Neural Network Classification*, 1st. New York, NY, USA: John Wiley & Sons, Inc., 1992, ISBN: 0471571547.

[147] P. D. McNelis, «Neural Networks and Genetic Algorithms: Tools for Forecasting and Risk Analysis in Financial Markets», *Department of Economics, Georgetown University*, 1997.

[148] ——, *Neural Networks in Finance: Gaining Predictive Edge in the Market*, ép. Academic Press Advanced Finance. Elsevier Science, 2005, ISBN: 9780124859678. dirección: https://books.google.es/books?id=QHruQkVnvJwC.

[149] B. Baesens, «Developing intelligent systems for credit scoring using machine learning techniques», K.U.Leuven. Faculteit Economische en toegepaste economische wetenschappen 180, 2003. dirección: http://www.dataminingapps.com/wp-content/uploads/2015/04/Phd-Bart-Baesens.pdf.

[150] D. Calvo, *Definición de red neuronal artificial*, http://www.diegocalvo.es/definicion-de-red-neuronal/, jul. de 2017. (visitado 02-09-2019).

[151] F. Sancho Caparrini, *Redes Neuronales: una visión superficial*, http://www.cs.us.es/~fsancho/?e=72, dic. de 2018. (visitado 02-09-2019).

[152] A. Moujahid, *A Practical Introduction to Deep Learning with Caffe and Python*, http://adilmoujahid.com/posts/2016/06/introduction-deep-learning-python-caffe/, jun. de 2016. (visitado 02-09-2019).

[153] I. Goodfellow, Y. Bengio y A. Courville, *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.

[154] V. F. Zambaldi, D. Raposo, A. S. and Victor Bapst, Y. Li, I. Babuschkin, K. Tuyls, D. P. R. and Timothy P. Lillicrap, E. Lockhart, M. Shanahan, V. Langston, R. P. and Matthew Botvinick y O. V. and Peter W. Battaglia, «Relational Deep Reinforcement Learning», *CoRR*, vol. abs/1806.01830, 2018. arXiv: 1806.01830. dirección: http://arxiv.org/abs/1806.01830.

[155] D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel, T. Lillicrap, K. Simonyan y D. Hassabis, «A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play», *Science*, vol. 362, n.º 6419, págs. 1140-1144, 2018, ISSN: 0036-8075. DOI: 10.1126/science.aar6404. eprint: https://science.sciencemag.org/content/362/6419/1140.full.pdf. dirección: https://science.sciencemag.org/content/362/6419/1140.

[156] O. Vinyals, I. Babuschkin, J. Chung, M. Mathieu, M. Jaderberg, W. M. Czarnecki, A. Dudzik, A. Huang, P. Georgiev, R. Powell, T. Ewalds, D. Horgan, M. Kroiss, I. Danihelka, J. Agapiou, J. Oh, V. Dalibard, D. Choi, L. Sifre, Y. Sulsky, S. Vezhnevets, J. Molloy, T. Cai, D. Budden, T. Paine, C. Gulcehre, Z. Wang, T. Pfaff, T. Pohlen, Y. Wu, D. Yogatama, J. Cohen, K. McKinney, O. Smith, T. Schaul, T. Lillicrap, C. Apps, K. Kavukcuoglu, D. Hassabis y D. Silver, *AlphaStar: Mastering the Real-Time Strategy Game StarCraft II*, https://deepmind.com/blog/alphastar-mastering-real-time-strategy-game-starcraft-ii/, 2019.

[157] M. Silva, *Aprendizaje por Refuerzo: Introducción al mundo del RL*, https://medium.com/aprendizaje-por-refuerzo-introducci%C3%B3n-al-mundo-del/aprendizaje-por-refuerzo-introducci%C3%B3n-al-mundo-del-rl-1fcfbaa1c87, abr. de 2019. (visitado 02-09-2019).

[158] S. Ruder, «An overview of gradient descent optimization algorithms.», *CoRR*, vol. abs/1609.04747, 2016, cite arxiv:1609.04747Comment: Added derivations of AdaMax and Nadam. arXiv: 1609.04747. dirección: http://arxiv.org/abs/1609.04747.

[159] D. P. Kingma y J. Ba, «Adam: A method for stochastic optimization», *arXiv preprint arXiv:1412.6980*, 2014. dirección: https://arxiv.org/pdf/1412.6980.pdf.

[160] S. J. Reddi, S. Kale y S. Kumar, «On the Convergence of Adam and Beyond», en *International Conference on Learning Representations*, 2018. dirección: https://openreview.net/forum?id=ryQu7f-RZ.

[161] S. K. Zhou, H. Greenspan y D. Shen, *Deep Learning for Medical Image Analysis*. Elsevier Science Academic Press, 2017, ISBN: 978-0-12-810408-8. DOI: 10.1016/B978-0-12-810408-8.00032-8. dirección: http://www.sciencedirect.com/science/article/pii/B9780128104088000328.

[162] F. Liu, Z. Zhou, H. Jang, A. Samsonov, G. Zhao y R. Kijowski, «Deep convolutional neural network and 3D deformable approach for tissue segmentation in musculoskeletal magnetic resonance imaging», *Magnetic Resonance in Medicine*, vol. 79, n.º 4, págs. 2379-2391, 2018. DOI: 10.1002/mrm.26841. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/mrm.26841. dirección: https://onlinelibrary.wiley.com/doi/abs/10.1002/mrm.26841.

[163] M. Nguyen, *Illustrated Guide to LSTM's and GRU's: A step by step explanation*, https://towardsdatascience.com/illustrated-guide-to-lstms-and-gru-s-a-step-by-step-explanation-44e9eb85bf21, sep. de 2018. (visitado 02-09-2019).

[164] S. Hochreiter y J. Schmidhuber, «Long Short-Term Memory», *Neural Computation*, vol. 9, n.º 8, págs. 1735-1780, dic. de 1997. DOI: 10.1162/neco.1997.9.8.1735.

[165] K. Cho, B. van Merrienboer, D. Bahdanau e Y. Bengio, «On the Properties of Neural Machine Translation: Encoder-Decoder Approaches», *CoRR*, vol. abs/1409.1259, 2014. arXiv: 1409.1259. dirección: http://arxiv.org/abs/1409.1259.

[166] J. Chung, C. Gulcehre, K. Cho e Y. Bengio, «Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling», *CoRR*, vol. abs/1412.3555, 2014. arXiv: 1412.3555. dirección: http://arxiv.org/abs/1412.3555.

[167] P. Loncomilla, *Deep learning: Redes convolucionales*, https://ccc.inaoep.mx/~pgomez/deep/presentations/2016Loncomilla.pdf, 2016. (visitado 02-09-2019).

[168] D. Calvo, *Red Neuronal Convolucional CNN*, http://www.diegocalvo.es/red-neuronal-convolucional/, jul. de 2017. (visitado 02-09-2019).

[169] S. Hijazi, R. Kumar y C. Rowen, «Using convolutional neural networks for image recognition», *Cadence Design Systems Inc.: San Jose, CA, USA*, 2015. dirección: https://ip.cadence.com/uploads/901/cnn_wp-pdf.

[170] A. Gandhi, *Data Augmentation | How to use Deep Learning when you have Limited Data — Part 2*, https://nanonets.com/blog/data-augmentation-how-to-use-deep-learning-when-you-have-limited-data-part-2/, 2018. (visitado 02-09-2019).

[171] J.-Y. Zhu, T. Park, P. Isola y A. A. Efros, «Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks», en *2017 IEEE International Conference on Computer Vision (ICCV)*, oct. de 2017, págs. 2242-2251. DOI: 10.1109/ICCV.2017.244.

[172] F. Luan, S. Paris, E. Shechtman y K. Bala, «Deep Photo Style Transfer», en *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, jul. de 2017, págs. 6997-7005. DOI: 10.1109/CVPR.2017.740.

[173] J. Yosinski, J. Clune, Y. Bengio y H. Lipson, «How transferable are features in deep neural networks?», en *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence y K. Q. Weinberger, eds., Curran Associates, Inc., 2014, págs. 3320-3328. dirección: http://papers.nips.cc/paper/5347-how-transferable-are-features-in-deep-neural-networks.pdf.

[174] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg y L. Fei-Fei, «ImageNet Large Scale Visual Recognition Challenge», *International Journal of Computer Vision*, vol. 115, n.º 3, págs. 211-252, dic. de 2015, ISSN: 1573-1405. DOI: 10.1007/s11263-015-0816-y. dirección: https://link.springer.com/article/10.1007/s11263-015-0816-y?sa_campaign=email/event/articleAuthor/onlineFirst#.

[175] A. Krizhevsky, I. Sutskever y G. E. Hinton, «ImageNet Classification with Deep Convolutional Neural Networks», en *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou y K. Q. Weinberger, eds., Curran Associates, Inc., 2012, págs. 1097-1105. dirección: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf.

[176] K. Simonyan y A. Zisserman, «Very deep convolutional networks for large-scale image recognition», *CoRR*, vol. abs/1409.1556, 2014. arXiv: 1409.1556. dirección: https://arxiv.org/abs/1409.1556.

[177] M. Lin, Q. Chen y S. Yan, «Network in network», *CoRR*, vol. abs/1312.4400, 2014. arXiv: 1312.4400. dirección: https://arxiv.org/abs/1312.4400.

[178] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke y A. Rabinovich, «Going Deeper with Convolutions», *CoRR*, vol. abs/1409.4842, 2014. arXiv: 1409.4842. dirección: http://arxiv.org/abs/1409.4842.

[179] «Rethinking the Inception Architecture for Computer Vision», *CoRR*, arXiv: 1512.00567. dirección: http://arxiv.org/abs/1512.00567.

[180] C. Szegedy, S. Ioffe, V. Vanhoucke y A. A. Alemi, «Inception-v4, inception-ResNet and the Impact of Residual Connections on Learning», en *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, ép. AAAI'17, San Francisco, California, USA: AAAI Press, 2017, págs. 4278-4284. dirección: http://dl.acm.org/citation.cfm?id=3298023.3298188.

[181] K. He, X. Zhang, S. Ren y J. Sun, «Deep Residual Learning for Image Recognition», en *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, jun. de 2016, págs. 770-778. DOI: 10.1109/CVPR.2016.90.

[182] A. Canziani, A. Paszke y E. Culurciello, «An Analysis of Deep Neural Network Models for Practical Applications», *CoRR*, vol. abs/1605.07678, 2016. arXiv: 1605.07678. dirección: http://arxiv.org/abs/1605.07678.

[183] Y. LeCun, «1.1 Deep Learning Hardware: Past, Present, and Future», en *2019 IEEE International Solid-State Circuits Conference-(ISSCC)*, feb. de 2019, págs. 12-19. DOI: 10.1109/ISSCC.2019.8662396.

[184] M. You, Z. Liu, C. Chen, J. Liu, X.-H. Xu y Z.-M. Qiu, «Cough detection by ensembling multiple frequency subband features», *Biomedical Signal Processing and Control*, vol. 33, págs. 132-140, 2017, ISSN: 1746-8094. DOI: 10.1016/j.bspc.2016.11.005. dirección: http://www.sciencedirect.com/science/article/pii/S1746809416301835.

[185] J. Amoh y K. Odame, «Deep Neural Networks for Identifying Cough Sounds», *IEEE Transactions on Biomedical Circuits and Systems*, vol. 10, n.º 5, págs. 1003-1011, oct. de 2016, ISSN: 1932-4545. DOI: 10.1109/TBCAS.2016.2598794. dirección: https://ieeexplore.ieee.org/abstract/document/7570164.

[186] Y. A. Amrulloh, U. R. Abeyratne, V. Swarnkar, R. Triasih y A. Setyati, «Automatic cough segmentation from non-contact sound recordings in pediatric wards», *Biomedical Signal Processing and Control*, vol. 21, págs. 126-136, ago. de 2015, ISSN: 1746-8094. DOI: 0.1016/j.bspc.2015.05.001. dirección: http://www.sciencedirect.com/science/article/pii/S1746809415000804.

[187] P. Klco, M. Kollarik y M. Tatar, «Novel computer algorithm for cough monitoring based on octonions», *Respiratory physiology & neurobiology*, vol. 257, págs. 36-41, 2018, Cough and Airway Defense - from neurophysiology to therapy, ISSN: 1569-9048. DOI: 10.1016/j.resp.2018.03.010. dirección: http://www.sciencedirect.com/science/article/pii/S1569904817303750.

[188] T. Drugman, «Using mutual information in supervised temporal event detection: Application to cough detection», *Biomedical Signal Processing and Control*, vol. 10, págs. 50-57, 2014, ISSN: 1746-8094. DOI: 10.1016/j.bspc.2014.01.001. dirección: http://www.sciencedirect.com/science/article/pii/S1746809414000020.

[189] T. Elfaramawy, C. Latyr Fall, M. Morissette, F. Lellouche y B. Gosselin, «Wireless respiratory monitoring and coughing detection using a wearable patch sensor network», en *2017 15th IEEE International New Circuits and Systems Conference (NEWCAS)*, jun. de 2017, págs. 197-200. DOI: 10.1109/NEWCAS.2017.8010139.

[190] L. Di Perna, G. Spina, S. Thackray-Nocera, M. G. Crooks, A. H. Morice, P. Soda y A. C. den Brinker, «An automated and unobtrusive system for cough detection», en *2017 IEEE Life Sciences Conference (LSC)*, dic. de 2017, págs. 190-193. DOI: 10.1109/LSC.2017.8268175.

[191] N. V. Chawla, K. W. Bowyer, L. O. Hall y W. P. Kegelmeyer, «SMOTE: synthetic minority over-sampling technique», *Journal of artificial intelligence research*, vol. 16, págs. 321-357, jun. de 2002. DOI: 10.1613/jair.953.

[192] J.-M. Liu, M. You, Z. Wang, G.-Z. Li, X. Xu y Z. Qiu, «Cough event classification by pretrained deep neural network», *BMC medical informatics and decision making*, vol. 15, n.º 4, S2, nov. de 2015, ISSN: 1472-6947. DOI: 10.1186/1472-6947-15-S4-S2.

[193] Y. Shi, H. Liu, Y. Wang, M. Cai y W. Xu, «Theory and Application of Audio-Based Assessment of Cough», *Journal of Sensors*, vol. 2018, 2018. DOI: 10.1155/2018/9845321.

[194] S. J. Barry, A. D. Dane, A. H. Morice y A. D. Walmsley, «The automatic recognition and counting of cough», *Cough*, vol. 2, n.º 1, pág. 8, 2006. DOI: 10.1186/1745-9974-2-8.

[195] J. Smith y A. Woodcock, «New Developments in the Objective Assessment of Cough», *Lung*, vol. 186, n.º 1, págs. 48-54, feb. de 2008, ISSN: 1432-1750. DOI: 10.1007/s00408-007-9059-1.

[196] S. Matos, S. S. Birring, I. D. Pavord y H. Evans, «Detection of cough signals in continuous audio recordings using hidden Markov models», *IEEE Transactions on Biomedical Engineering*, vol. 53, n.º 6, págs. 1078-1083, jun. de 2006, ISSN: 0018-9294. DOI: 10.1109/TBME.2006.873548. dirección: https://ieeexplore.ieee.org/document/1634502.

[197] M. A. Coyle, D. B. Keenan, L. S. Henderson, M. L. Watkins, B. K. Haumann, D. W. Mayleben y M. G. Wilson, «Evaluation of an ambulatory system for the quantification of cough frequency in patients with chronic obstructive pulmonary disease», *Cough*, vol. 1, n.º 1, pág. 3, 2005. DOI: 10.1186/1745-9974-1-3.

[198] E. Vizel, M. Yigla, Y. Goryachev, E. Dekel, V. Felis, H. Levi, I. Kroin, S. Godfrey y N. Gavriely, «Validation of an ambulatory cough detection and counting application using voluntary cough under different conditions», *Cough*, vol. 6, n.º 1, pág. 3, 2010. DOI: 10.1186/1745-9974-6-3.

[199] K. Yatani y K. N. Truong, «BodyScope: a wearable acoustic sensor for activity recognition», en *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, ép. UbiComp '12, ACM, Pittsburgh, Pennsylvania: ACM, 2012, págs. 341-350, ISBN: 978-1-4503-1224-0. DOI: 10.1145/2370216.2370269. dirección: http://doi.acm.org/10.1145/2370216.2370269.

[200] U. R. Abeyratne, A. Swarnkar Vinayak and Setyati y R. Triasih, «Cough Sound Analysis Can Rapidly Diagnose Childhood Pneumonia», *Annals of Biomedical Engineering*, vol. 41, n.º 11, págs. 2448-2462, nov. de 2013, ISSN: 1573-9686. DOI: 10.1007/s10439-013-0836-0.

[201] P. Rajpurkar, J. Irvin, K. Zhu, B. Yang, H. Mehta, T. Duan, D. Y. Ding, A. Bagul, C. L. and Katie S. Shpanskaya and Matthew P. Lungren y A. Y. Ng, «CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning», *CoRR*, vol. abs/1711.05225, 2017. arXiv: 1711.05225. dirección: http://arxiv.org/abs/1711.05225.

[202] G. González, S. Y. Ash, G. Vegas-Sánchez-Ferrero, J. Onieva Onieva, F. N. Rahaghi, J. C. Ross, A. Díaz, R. San José Estépar y G. R. Washko, «Disease Staging and Prognosis in Smokers Using Deep Learning in Chest Computed Tomography», *American Journal of Respiratory and Critical Care Medicine*, vol. 197, n.º 2, págs. 193-203, 2018, PMID: 28892454. DOI: 10.1164/rccm.201705-0860OC.

[203] J. Shiraishi, S. Katsuragawa, J. Ikezoe, T. Matsumoto, T. Kobayashi, K.-i. Komatsu, M. Matsui, H. Fujita, Y. Kodera y K. Doi, «Development of a digital image database for chest radiographs with and without a lung nodule: receiver operating characteristic analysis of radiologists'detection of pulmonary nodules», *American Journal of Roentgenology*, vol. 174, n.º 1, págs. 71-74, ene. de 2000, ISSN: 0361-803X. DOI: 10.2214/ajr.174.1.1740071.

[204] S. Juhász, Á. Horváth, L. Nikházy y G. Horváth, «Segmentation of Anatomical Structures on Chest Radiographs», en *XII Mediterranean Conference on Medical and Biological Engineering and Computing 2010*, P. D. Bamidis y N. Pallikarakis, eds., Springer, Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, págs. 359-362, ISBN: 978-3-642-13039-7. DOI: 10.1007/978-3-642-13039-7_90.

[205] Y. Gordienko, P. Gang, J. Hui, W. Zeng, Y. Kochura, O. Alienin, O. Rokovyi y S. Stirenko, «Deep Learning with Lung Segmentation and Bone Shadow Exclusion Techniques for Chest X-Ray Analysis of Lung Cancer», en *Advances in Computer Science for Engineering and Education*, Z. Hu, I. Petoukhov Sergey and Dychka y M. He, eds., Springer, Cham: Springer International Publishing, 2019, págs. 638-647, ISBN: 978-3-319-91008-6. DOI: 10.1007/978-3-319-91008-6_63.

[206] C. Hoyos-Barceló, J. R. Garmendia-Leiza, M. D. Aguilar-García, J. Monge-Álvarez, D. A. Pérez-Alonso, C. Alberola-López y P. Casaseca-de-la-Higuera, «Evaluation in a Real Environment of a Trainable Cough Monitoring App for Smartphones», en *15th Mediterranean Conference on Medical and Biological Engineering and Computing (MEDICON'2019)*, 2019.

[207] S. Matos, S. S. Birring, I. D. Pavord y D. H. Evans, «An automated system for 24-h monitoring of cough frequency: the leicester cough monitor», *IEEE Transactions on Biomedical Engineering*, vol. 54, n.º 8, págs. 1472-1479, ago. de 2007, ISSN: 0018-9294. DOI: 10.1109/TBME.2007.900811.

[208] S. E. Küçükbay y M. Sert, «Audio-based event detection in office live environments using optimized MFCC-SVM approach», en *Proceedings of the 2015 IEEE 9th International Conference on Semantic Computing (IEEE ICSC 2015)*, feb. de 2015, págs. 475-480. DOI: 10.1109/ICOSC.2015.7050855. dirección: https://ieeexplore.ieee.org/document/7050855.

[209] A. S. Soriano, L. S. Osuna y J. B. Rivera, *Enfermedad pulmonar difusa II: sarcoidosis. Alveolitis alérgica extrínseca. Neumonía organizada criptogénética*, https://www.neumosur.net/files/EB03-33%20EPID%202.pdf.

[210] A. Oppenheim y R. Schafer, «Análisis de Fourier de señales mediante la DFT», en *Tratamiento de señales en tiempo discreto*, 3 ed, ép. Fuera de colección Out of series. Pearson Educación, 2011, cap. 10, págs. 769-864, ISBN: 9788483227183. dirección: https://books.google.es/books?id=9R7bXwAACAAJ.

[211] ——, «Diseño de filtros FIR mediante enventanado», en *Tratamiento de señales en tiempo discreto*, 3 ed, ép. Fuera de colección Out of series. Pearson Educación, 2011, cap. 7.5, págs. 522-534, ISBN: 9788483227183. dirección: https://books.google.es/books?id=9R7bXwAACAAJ.

[212] J. M. Merino y L. Muñoz-Repiso, «La percepción acústica: física de la audición», *Revista de ciencias*, n.º 2, págs. 19-26, 2013. dirección: https://dialnet.unirioja.es/descarga/articulo/4293906.pdf.