La investigación reportada en esta tesis es parte de los programas de investigación del CICESE (Centro de Investigación Científica y de Educación Superior de Ensenada, Baja California).

La investigación fue financiada por el CONAHCYT (Consejo Nacional de Humanidades Ciencias y Tecnologías).

Centro de Investigación Científica y de Educación Superior de Ensenada, Baja California

Maestría en Ciencias
en Tecnologías Avanzadas e Integradas

Análisis acústico de la tos para detección de COVID con énfasis en el balanceo de clases

Tesis
para cubrir parcialmente los requisitos necesarios para obtener el grado de
Maestro en Ciencias

Presenta:

Arley Magnolia Aquino García

Ensenada, Baja California, México
2023

Tesis defendida por

Arley Magnolia Aquino García

y aprobada por el siguiente Comité

Dr. Humberto Pérez Espinosa
Codirector de tesis

Dr. Javier Andreu Pérez
Codirector de tesis

Dr. Ansel Rodríguez González

Dr. Armando Trasviña Castro

Dr. Víctor Manuel Coello Cárdenas
Coordinador de la Maestría en Tecnologías Avanzadas e Integradas

Dra. Ana Denise Re Araujo
Directora de Estudios de Posgrado

Resumen de la tesis que presenta **Arley Magnolia Aquino García** como requisito parcial para la obtención del grado de Maestro en Ciencias en Tecnologías Avanzadas e Integradas.

Análisis acústico de la tos para detección de COVID con énfasis en el balanceo de clases

Resumen aprobado por:

Dr. Humberto Pérez Espinosa Codirector de tesis	**Dr. Javier Andreu Pérez** Codirector de tesis

El desarrollo de métodos computacionales que permitan actuar rápidamente en el diagnóstico de enfermedades nuevas, como el COVID-19, utilizando datos acústicos, como grabaciones de toses, es de suma importancia. Este enfoque ofrece la posibilidad de detectar patrones y características distintivas en el sonido de la tos de los pacientes, lo que puede ser utilizado como una herramienta no invasiva y de fácil acceso para identificar enfermedades respiratorias lo cual puede ayudar a contener su propagación, implementar medidas preventivas y proporcionar un tratamiento adecuado en etapas tempranas. Este documento propone un enfoque acústico para la detección de COVID-19 a través de análisis de tos Investigamos la eficacia de las técnicas de segmentación y equilibrio de clases para mejorar el rendimiento de nuestro modelo de clasificación. Nuestro enfoque implica preprocesamiento y extracción de características de señales de audio de tos, seguida de segmentación para obtener muestras que incluyan periodos de tos para análisis. Evaluamos la efectividad de diferentes técnicas de segmentación y demostramos que segmentar las señales de audio en función de los intervalos de silencio da como resultado el máximo rendimiento. Además, abordamos el problema del desequilibrio de clases, que es un problema común en los conjuntos de datos médicos, aplicando técnicas de submuestreo y sobremuestreo. Comparamos nuestro enfoque de detección de COVID-19 propuesto con técnicas actuales de detección de COVID-19 y demostramos que nuestro enfoque ofrece un análisis más amplio para la clasificación de COVID-19 a través del uso de modelos de aprendizaje automático, segmentación automática de señales acústicas de la tos, y balanceo de clases. Permitiendo la posible detección automática de enfermedades respiratorias a través de la tos. Nuestros resultados muestran que el análisis de la tos puede servir como un método no invasivo, rentable, y herramienta confiable para la detección de COVID-19, especialmente en entornos de recursos limitados.

Palabras clave: Análisis acústico, COVID-19, Segmentación, Balanceo de clases, Clasificación

Abstract of the thesis presented **by Arley Magnolia Aquino García** as a partial requirement to obtain the Master of Science degree in Advanced and Integrated Technologies.

Acoustic analysis of cough for COVID detection with emphasis on class balance

Abstract approved by:

Dr. Humberto Pérez Espinosa	**Dr. Javier Andreu Pérez**
Thesis Co-Director	Thesis Co-Director

The development of computational methods that allow rapid action in the diagnosis of new diseases, such as COVID-19, using acoustic data, such as cough recordings, is of the utmost importance. This approach offers the possibility of detecting distinctive patterns and characteristics in the sound of patients' coughs, which can be used as a non-invasive and easily accessible tool to identify respiratory diseases which can help contain their spread, implement measures preventive measures and provide adequate treatment in early stages. This paper proposes an acoustic approach to detecting COVID-19 through cough analysis. We investigated the efficacy of class balancing and segmentation techniques in improving the performance of our classification model. Our approach involves preprocessing and feature extraction from audio cough signals, followed by segmentation to obtain samples that include cough periods for analysis. We evaluate the effectiveness of different segmentation techniques and show that segmenting audio signals based on intervals of silence results in maximum performance. In addition, we address the problem of class imbalance, which is a common problem in medical data sets, by applying undersampling and oversampling techniques. We compare our proposed COVID-19 detection approach with current COVID-19 detection techniques and demonstrate that our approach offers broader analysis for COVID-19 classification through the use of machine learning models, automatic signal segmentation cough acoustics, and class swinging. Allowing the possible automatic detection of respiratory diseases through coughing. Our results show that cough testing can serve as a non-invasive, cost-effective, and reliable tool for the detection of COVID-19, especially in resource-limited settings.

Keywords: Acoustic analysis, COVID-19, Segmentation, Class balancing, Classification

Dedicatoria

A mi madre por ser mi luz de todos mis días, por acompañarme incondicional, nunca dejarme sola y motivarme a seguir adelante ante cualquier circunstancia. A mi padre por ser mi pilar y mi soporte principal. A mi hermano Enrique por ser mi compañero y guía. A mi hermano David por ser mi amigo y mi compañero. A mi hermana Estrella por ser la estrella de mi vida. A mi familia por su amor y apoyo, incluso desde el cielo. Y a la persona que me apoyó sin importar los miles de kilómetros de distancia.

Agradecimientos

Al Centro de Investigación Científica y de Educación Superior de Ensenada, Baja California por la oportunidad de mi formación de maestría.

Al Consejo Nacional de Humanidades Ciencias y Tecnologías (CONAHCYT) por brindarme el apoyo económico para realizar mis estudios de maestría. No. de becario: 20264680

A mis directores de tesis Dr. Humberto Pérez Espinosa y Dr. Javier Andreu Pérez. Por brindarme todas las posibilidades para realizar este trabajo de investigación. Por su esfuerzo y dedicación para aportar su conocimiento, sus orientaciones y forma de trabajar. Por su paciencia y su motivación. Gracias Dr. Humberto por su acompañamiento y compromiso hacia este trabajo de tesis. Gracias Dr. Javier por su conocimiento y seguimiento continuo. Para ambos, mi admiración agradecimiento por siempre. No me pude haber sentido más feliz del trabajo que hemos logrado y todo es gracias a su enseñanza, me quedo con todos sus consejos y admiración ante su trabajo.

A mi comité de tesis Dr. Ansel Rodríguez González, Dr. Armando Trasviña Castro por su compromiso a este trabajo de tesis, por sus comentarios y observaciones, para que este trabajo fuese terminado y formado íntegramente.

Al Dr. Ansel Rodríguez González por su acompañamiento en todo mi proceso de formación de maestría.

A mis padres Andrea García y Ramiro Aquino por estar al pie del cañon, y motivarme, son mi admiración y motivación de mi vida.

A mis hermanos, Enrique, David y Estrella por su apoyo incondicional.

A Efren Bernal por estar al pendiente de mi cada momento, guiarme y motivarme todos los días.

A mi compañero y amigo Antonio Rivas Navarrete por ayudarme y acompañarme en todo este proceso de investigación.

Gracias Arley por no rendirte.

Tabla de contenido

Página

Lista de figuras

Lista de tablas

Capítulo 1. Introducción

En los últimos años la humanidad pasó por una pandemia que afectó a toda la población mundial. Durante esta crisis miles de profesionales e investigadores de todas las áreas de las ciencias se sumaron a la investigación para combatir y comprender los efectos, el desarrollo y la sintomatología de la enfermedad y sus consecuencias. Sin embargo, 3 años después de los esfuerzos, queda claro que las enfermedades respiratorias representan un gran peligro y debemos estar preparados ante la aparición de nuevos virus. Ahora que contamos con mayor información y comprensión sobre el virus del Síndrome Respiratorio Agudo Severo (SARS-CoV-2) se debe aprovechar el conocimiento para tratar de analizar e identificar enfermedades respiratorias y agudas.

En este trabajo de investigación se presenta una metodología integral para la detección de enfermedades respiratorias y agudas con aprendizaje automático, a través de señales acústicas de la tos. Se utilizan toses de pacientes positivos al COVID-19 y de pacientes negativos, con la finalidad de aprovechar la información disponible y abordar el problema que recientemente impactó en toda la humidad.

Las muestras de toses utilizadas en este estudio son de pacientes positivos y negativos al COVID-19 y son clínicamente validades por la prueba qRT-PCR (Reacción en Cadena de la Polimerasa de Transcripción Inversa Cuantitativa en Tiempo Real). El objetivo de esta tesis es investigar y analizar el efecto de diversas técnicas de segmentación y balanceo de clases en el contexto del diagnóstico de enfermedades respiratorias a través del análisis de grabaciones de tos. Se propone un método integral de diagnóstico que aprovecha las características acústicas de la tos para identificar y clasificar enfermedades respiratorias. Para lograr esto, se exploran diferentes enfoques de segmentación de tos y se evalúa su efectividad en la identificación de información útil relacionada con el COVID-19. Además, se investigan técnicas de balanceo de clases para abordar los desequilibrios en los datos y mejorar la capacidad del modelo para reconocer la clase minoritaria. El objetivo final es proponer un método de tamizaje integral basado en la tos que tenga la capacidad de ser eficaz, accesible y no invasivo, ofreciendo así una herramienta prometedora para el prediagnóstico temprano y la gestión de enfermedades respiratorias.

Aunque esta tesis se enfoca al caso específico del COVID-19, es importante enfatizar que la metodología propuesta es aplicable a otras enfermedades respiratorias agudas que también pueden ser detectadas mediante la tos, ya que, en todas estas enfermedades, el primer paso crucial es la segmentación de los periodos de interés en las grabaciones, lo que permite un análisis más preciso. Además, es muy probable

que el estudio de otras enfermedades se disponga también de un mayor número de individuos sanos en comparación con los enfermos. Por lo tanto, esta investigación presenta un valioso aporte para el abordaje rápido y preciso en la detección de este tipo de enfermedades, permitiendo un enfoque más eficaz en su diagnóstico y tratamiento.

En este capítulo se presenta una perspectiva general de la investigación. Se exponen los antecedentes del objeto de estudio, y la propuesta de investigación. Posteriormente se presenta la hipótesis, seguida de los objetivos a alcanzar con el desarrollo de este trabajo. Por último, se describe la metodología general propuesta.

1.1 Antecedentes

El COVID-19 se presentó por primera vez en un humano a finales de noviembre del 2019 en Wuhan, provincia de Hubei (China) provocado por el virus del Coronavirus de tipo 2 causante del SARS-CoV-2 (Gaviria, 2023). El brote de COVID-19, se esparció rápidamente entre las personas, las cuales presentaban síntomas parecidos a la de una neumonía de tipo viral, tales como, la fiebre, tos seca, fatiga, vómito y diarrea, entre los más importantes (Pullen M. F., 2020) (Andrews P. L., 2021) (Escudero, 2020). Para el 30 de enero del 2020 la Organización Mundial de la Salud (OMS) declaró una emergencia sanitaria mundial (Velavan, 2020) debido a que el brote de la enfermedad presentaba crecientes tasas de contagio en personas de todo el mundo, convirtiéndose en una pandemia. En la última actualización del sitio web proporcionado por la Universidad Johns Hopkins correspondiente al mes de marzo del 2023 ya habían 676,609,955 casos de COVID-19 por todo el mundo y 6,881,955 muertes. (Hopkins, 2023)

El creciente número de casos confirmados de COVID-19 trajo consigo una crisis sanitaria mundial, la cual repercutió en casi todos los sectores de la población; en la economía, educación, turismo, social y en la globalización (Kumar, 2021; Das, 2022; Harper, 2020). Esto se debe a la tasa de reproducción básica (R0) del virus SARS-CoV-2 de humano a humano de forma acelerada. En 2020 se estimó un R0 de 1.5 a 6.68 (Liu, 2020), debido a la propagación del virus por aerosoles, siendo 3.28 y 2.79 el número promedio de casos nuevos que una persona infectada contagia dentro del periodo infeccioso de la enfermedad, además se observó que progresa más rápido en las personas adultas que en las jóvenes (Kadambari, 2020). Las personas enfermas de COVID-19 generalmente comienzan a mostrar síntomas dentro de los 8.2 a 15.6 días, con un promedio de 11.2 días (Sharma A. A., 2021). Es por ello, que resultó indispensable la generación de métodos de detección rápida de COVID-19.

Los métodos actuales de detección de COVID-19 incluyen pruebas de Reacción en Cadena de la Polimerasa (PCR), Tomografías Computarizadas (TC) y pruebas de anticuerpos. Sin embargo, estos métodos son costosos, requieren tiempo para ver los resultados, además pueden requerir equipo y personal especializado (Zhang, 2023; Dong, 2020; Ji T. L., 2020). Resulta indispensable generar métodos que sean asequibles, oportunos y eficientes para mitigar el contagio de COVID-19 entre la sociedad. Si bien, al día de hoy se encuentran disponibles vacunas para prevenir el contagio, sólo han sido administradas 13,338,833,198 dosis de vacunas a personas por todo el mundo hasta el marzo del 2023 (Hopkins, 2023).

El avance de las herramientas que ofrecen las Tecnologías de la Información y Comunicación (TICS) parecen jugar un papel importante para la detección rápida de COVID-19. Conforme infectaba a personas de todo el mundo, a la par se publicaban estudios como un esfuerzo de aportar información relevante para su rápida detección y conceptualización. Sin embargo, los trabajos se centraron en generar soluciones rápidas a través de modelos de Inteligencia Artificial (IA), invirtiendo esfuerzos únicamente en detectar rápidamente el COVID-19, dejando de lado la investigación de metodologías eficientes y completas. Resulta importante detectar el virus con un método integral y adecuado. En los últimos años existe un interés en el uso de enfoques acústicos para la detección de enfermedades respiratorias a través del análisis de los sonidos de la tos, ofreciendo nuevas herramientas de detección no invasivas rentables (Laguarta, 2020; Deshpande, 2020; Bagad, 2020; Brown, 2020).

1.2 Propuesta

Los trabajos actuales abordan la detección de COVID-19 a través de sonidos acústicos de la tos con metodologías centradas únicamente en la clasificación de la representación acústica de la misma. Sin embargo, resulta importante proponer metodologías integrales que se centren utilizar técnicas de segmentación, representación y clasificación óptima para la detección de toses, sanas o enfermas. En la revisión del estado del arte se encontró que, de 5 trabajos revisados, únicamente los trabajos de (Pahar, 2021; Andreu-Perez, 2021) abordan un análisis de segmentación de tos para obtener sólo la información necesaria para la entrenar modelos de aprendizaje automático, sin embargo, estos trabajos no realizan una comparativa de distintas técnicas de segmentación ni evaluación de las mismas. De los trabajos que resuelven el problema de desequilibrio de clases (Pahar, 2021; Xia, 2021; Mouawad, 2021), ninguno de ellos evidencia el uso de técnicas de generación sintética de audio; además, no realizan una comparativa de diversos tipos de balanceo de clases. Por ejemplo, comparar balanceo de clases por aumento de datos, o por generación sintética de audio; las técnicas que se utilizan en los trabajos son de tipo convencionales.

En este trabajo de investigación se propone un enfoque novedoso para la detección de COVID-19 a través del análisis acústico de los sonidos de la tos. Este enfoque implica técnicas de segmentación y equilibrio de clases para mejorar el rendimiento de nuestro modelo de clasificación.

La segmentación es el proceso de dividir una señal de audio en segmentos significativamente más pequeños, y puede mejorar la precisión de la clasificación al analizar sólo los segmentos relevantes. Se utilizaron técnicas de segmentación debido a que la tos emitida por personas se captura en un intervalo de tiempo, sin embargo, en este intervalo se identifican pequeños espacios en ausencia de tos, lo cual no es relevante para la detección de COVID-19. Es por esto que, se divide la señal acústica en espacios con tos y ausencia de tos, para posteriormente tomar en cuenta sólo aquellos que contienen tos asegurando considerar segmentos de señal que, sí contengan información de interés para el objeto de estudio, en contraste con los trabajos actuales dónde se toman en cuenta la señal acústica completa de la tos. Realizar la segmentación de audio, es una tarea indispensable para detección de enfermedades respiratorias, ya que permite extraer de una grabación de audio, sonidos de tos, considerando que cuando se realizan las grabaciones de las muestras no sólo existe tos, sino que también ruido ambiental, voz, etc. Es imprescindible considerar entrenar algoritmos de aprendizaje automático con información útil, esto minimiza los sesgos de predicción (Cohen-McFarlane, 2019; V. Swarnkar, 2013).

El desequilibrio de clases es un problema común en los conjuntos de datos médicos (Fotouhi, 2019; Larrazabal, 2020), esto se debe a que la cantidad de casos positivos (COVID-19) es mucho menor que los casos negativos (no COVID-19), lo que lleva a un rendimiento deficiente de los modelos de clasificación, (Alsaif, 2021; Kuo, 2022) ocasionando sesgos en predicciones de clases minoritarias. Abordamos este problema aplicando técnicas de submuestreo (Nguyen, 2020), sobremuestreo (Bhattacharjee, 2020), aumento de datos (data augmentation) (Mushtaq, 2020) y algoritmos generativos (Codificadores automáticos variacionales, VAE) (Saldanha, 2022) para equilibrar las clases y mejorar el rendimiento de nuestro modelo. Se evalúa la efectividad de diferentes técnicas de segmentación y equilibrio de clases. Considerando que no sólo es importante detectar el COVID-19, sino que, también sea a través de una metodología completa y eficiente dónde se contemplen algunos de los problemas más importantes.

En un artículo reciente (Coppock, 2021), se presenta una crítica sobre trabajos que se publicaron durante la pandemia con relación a la detección automática de COVID-19 a partir de la tos. Los autores resaltan una disyuntiva acerca de las propuestas actuales para la detección de COVID-19 dónde enfatizan 7 problemas que no se abordan, los cuales son:

- Los algoritmos sólo detectan entre individuos sanos y enfermos en lugar de detectar COVID-19.

- En los conjuntos de datos los participantes conocen el estado de COVID-19 en el momento que realizan la grabación, esto es un problema porque la información de las toses puede verse afectada por la emoción en la voz y el comportamiento del participante.

- La validez del conjunto de datos para el entrenamiento de los algoritmos de aprendizaje.

- El difícil acceso a las bases de datos.

- La comorbilidad, los factores geográficos, étnicos y socioeconómicos son una preocupación potencial en el contexto del uso del aprendizaje automático para detectar COVID-19, la influencia de estos factores en la propagación de COVID-19 es compleja.

- Incluir información innecesaria en la grabación de un audio, es decir, que no sólo contiene tos, resulta en entrenamientos de algoritmos automáticos sesgados.

- Por último, otro de los problemas encontrados en investigaciones actuales es el nivel de control que se tiene sobre la población de participantes al desarrollar modelos de aprendizaje automático, los modelos de inteligencia artificial son competentes en la identificación de hablantes, por lo tanto, es imperativo que los conjuntos de entrenamiento, desarrollo y prueba sean poblaciones de participantes demográficamente distintas.

En este trabajo de investigación de tesis se abordan 5 de los 7 problemas, cómo se analizará en el capítulo 4, subtema 4.1.1 el conjunto de datos no sólo se conforma de muestras de personas sana o enfermas de COVID-19, las muestras recopiladas por el conjunto de datos a utilizar, se conforma por grabaciones de toses de personas sanas con y sin síntomas de COVID-19, y grabaciones de personas sanas con y sin síntomas de COVID-19. El segundo problema que se aborda radica en que, en el proceso de recopilación de la base de datos, los participantes no conocían su estado de salud, las grabaciones de toses se realizaron antes de recibir el diagnóstico, dejando de lado el sesgo de incluir emociones que afecten los sonidos de tos grabadas. Las muestras de la base de datos son clínicamente validadas, reduciendo el sesgo de aprendizaje de los algoritmos automáticos, con datos reales validados. Y, por último, en este trabajo se incluyen métodos de segmentación para considerar sólo la información necesaria de las grabaciones de los participantes, la cual es sólo la tos.

Nuestros resultados muestran que el análisis de la tos puede servir como una herramienta prometedora para la detección de COVID-19 con muestras clínicamente validadas por la prueba qRT-PCR. La detección se hace a través del uso de técnicas de segmentación para obtener información valiosa para que el aprendizaje de los algoritmos sea realmente de la tos, y no de segmentos donde existen silencios, abordando así el problema de no incluir información necesaria a los algoritmos de aprendizaje. Cómo se analizará en el capítulo 3, subtema 3.1. Los trabajos encontrados que abordan la segmentación, no presentan evidencia que utilizaron distintas técnicas de segmentación y que estas fuesen evaluadas. Como propuesta de este trabajo se pretende abordar esta tarea, comparando técnicas de segmentación y utilizando un método de evaluación para identificar el rendimiento de las técnicas. Además, se aborda el equilibrio de los datos de entrenamiento para minimizar los sesgos de predicción de los algoritmos de aprendizaje automático.

Esta propuesta de metodología para detección de COVID-19 se propone para la detección de enfermedades respiratorias donde uno de sus principales síntomas sea la tos y se puede aplicar en cualquier ámbito, especialmente en lugares geográficos con recursos limitados, dónde es más difícil el acceso a pruebas clínicas de laboratorio por su accesibilidad de cercanía y asequibilidad. Contribuyendo a los esfuerzos para contener enfermedades respiratorias.

1.3 Motivación

La detección automática de enfermedades respiratorias a través de la tos en los últimos años ha resultado en una tarea importante (Alqudaihi, 2021; Ijaz, 2022; Pramono, 2019; Gabaldón-Figueira, 2022). Específicamente en la detección automática de una de las enfermedades que afecto a todo el mundo, el COVID-19, varios estudios muestran que utilizar la tos para su detección es una herramienta prometedora (Tena, 2022; Lella, 2022; Andreu-Perez, 2021; Islam R. A.-R., 2022; Chowdhury, 2022). Sin embargo, la principal motivación de este trabajo es abordar aspectos que no se abordan en los más recientes trabajos de investigación tal como lo mencionó (Coppock, 2021) en su trabajo, y como se analizó en el subtema 1.3 de este capítulo. Es imperativo detenerse y analizar en profundidad un método que permita en aplicaciones reales maximizar los resultados del desempeño de los algoritmos de aprendizaje automático para detectar COVID-19 y enfermedades respiratorias a través de la tos.

En este trabajo se hace énfasis en la utilización de técnicas de segmentación de los datos, para la extracción útil de las grabaciones que conforma la base de datos para detección de COVID-19. Esto es imprescindible

puesto que, los algoritmos de aprendizaje automático deben entrenarse información necesaria. Además del énfasis de unos de los problemas que comúnmente presentan las bases de datos, el cual es el desbalance de clases (Wan, 2014; Khalilia, 2011; Mazurowski, 2008). Que las bases de datos estén desequilibradas, significa que los datos se inclinan más a una clase, dicho de otra forma, una clase presenta un número de instancias mayor que otra. El desbalance de clases ocasiona sesgos en la clasificación de las clases (Sun, 2007; Alsaif, 2021; Ji X. P., 2023).

Es imprescindible presentar métodos integrales de detección de COVID-19, que no sólo se enfoquen en una tarea específica, esto permite aportar información invaluable para detectar otras enfermedades respiratorias a través de la tos, con algoritmos de aprendizaje automático. Este trabajo se centra en una metodología integral para detección automática de COVID-19, sin embargo, podría utilizarse para detectar otro tipo de enfermedades respiratorias a través de la tos.

1.4 Hipótesis

La elección adecuada de técnicas de segmentación de la señal, balanceo de clases y representación de información en función de las características del conjunto de datos y del problema específico de tamizaje de enfermedades respiratorias agudas resultará en un enfoque adecuado que permita evaluar su rendimiento y capacidad de tamizaje.

1.5 Objetivos

1.5.1 Objetivo general

El objetivo principal de la investigación es desarrollar un método para el tamizaje de COVID-19 que aborde las problemáticas de segmentación de la señal, balanceo de clases y representación de información evaluando su rendimiento y capacidad de identificar la enfermedad.

1.5.2 Objetivos específicos

- Identificar técnicas de segmentación y evaluación de las mismas que resulten adecuadas para el conjunto de datos con el que se cuenta.

- Identificar técnicas de representación de los datos adecuadas para los datos acústicos con los que se cuentan.

- Identificar técnicas de balanceo de clases adecuadas para los datos con los que se cuenta.

- Desarrollar un método de clasificación automática evaluando diferentes algoritmos de clasificación y aplicando las mejores técnicas de balanceo identificadas.

- Evaluar el método de clasificación desarrollada.

1.6 Metodología

La metodología propuesta para este trabajo de investigación se describe a continuación:

1. Revisión del estado del arte en clasificación automática de la tos: Realizar búsqueda de trabajos relevantes y/o actuales que aborden la clasificación automática de la tos de enfermedades respiratorias.

2. Familiarización con la base de datos creada por la Unidad Tepic del Centro de Investigación Científica y de Educación Superior de Ensenada (CICESE-UT3): Identificar las muestras positivas y negativas de toses disponibles, además de las características demográficas de los pacientes.

3. Revisión del estado del arte sobre técnicas de balanceo de clases y generación de datos sintéticos: Realizar búsqueda de trabajos relevantes y/o actuales que aborden el problema de desequilibrio de clasificación de COVID a partir de la tos. Identificar trabajos relacionados a la generación sintética de señales acústicas.

4. Búsqueda de bases de datos complementarias: Identificar bases de datos disponibles de muestras de toses de pacientes positivos y negativos a COVID-19.

5. Preprocesamiento: Dividir señal acústica de tos con técnicas de segmentación para identificar fragmentos de silencios y tos, obteniendo sólo los segmentos de tos.

6. Representación: Caracterizar la señal acústica en un formato en que los modelos de aprendizaje automático admitan.

7. Implementación de técnicas de balanceo de clases: Probar técnicas de balanceo de clases.

8. Clasificación automática: Entrenar modelos de aprendizaje automático.

9. Evaluación y análisis de resultados: Obtener resultados de segmentación, caracterización y clasificación para su posterior análisis.

La arquitectura de la metodología integral propuesta en este trabajo de investigación se muestra en la figura 1.

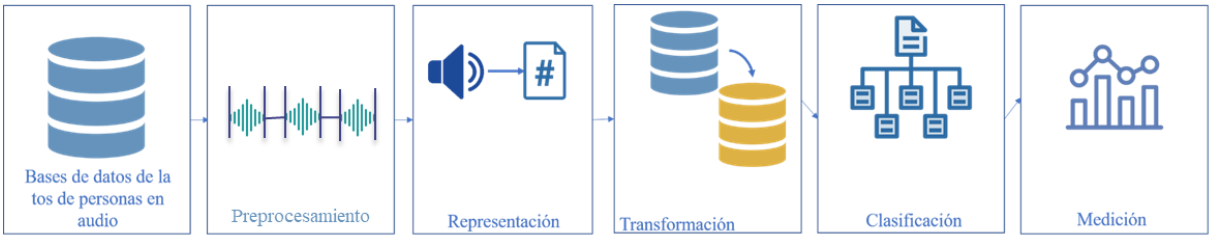


Figura 1 Propuesta de metodología integral

1.7 Contribuciones esperadas

Como resultados de éste trabajo se esperan las siguientes contribuciones:

- Comparativa de diversas técnicas de segmentación de la señal para el tamizaje de enfermedades respiratorias a partir de la tos.

- Análisis del impacto del balanceo de clases en el diagnóstico de enfermedades respiratorias a partir de la tos.

- Análisis del impacto de la combinación de diferentes técnicas de segmentación de la señal, balanceo de clases y representación de información en el diagnóstico de enfermedades respiratorias a partir de la tos.

- Desarrollo de un enfoque integral para el diagnóstico de enfermedades respiratorias agudas al combinar estratégicamente diferentes técnicas de segmentación de la señal, balanceo de clases y representación de información.

- Contribución al avance de la investigación en el campo del diagnóstico de enfermedades respiratorias agudas, al proporcionar resultados y conclusiones significativos que pueden servir de base para futuros estudios y desarrollos en esta área.

1.8 Organización de la tesis

El resto del documento de tesis se conforma con los siguientes capítulos:

Capítulo 2 (Fundamentos teóricos) se introducen los conceptos relacionados al COVID-19, los conjuntos de datos, técnicas de segmentación de datos y las técnicas de balanceo de clases utilizadas a lo largo del presente trabajo de tesis.

Capítulo 3 (Trabajo relacionado) se revisan los trabajos relacionados a la clasificación de COVID-19 a través de datos acústicos como lo es la tos, con atención en el balance de clases.

Capítulo 4 (Metodología) se describe la metodología utilizada en este trabajo de tesis para la clasificación de COVID-19 con énfasis en el balanceo de clases.

Capítulo 5 (Resultados) se aborda el conjunto de resultados derivados de los experimentos realizados en la metodología propuesta.

Capítulo 6 (Conclusiones y trabajo futuro) se exponen las conclusiones identificadas a lo largo del desarrollo de este trabajo de tesis, así mismo se abordan las contribuciones obtenidas, las limitaciones y el trabajo futuro.

Capítulo 2. Fundamentos teóricos

En este capítulo se introducen los conceptos relacionados al objeto de investigación de este trabajo de tesis, el cual se centra en la clasificación automática de COVID-19 a través de señales acústicas de la tos, abordando la problemática del desbalanceo de clases que se presenta en las bases de datos médicas y la segmentación de la señal acústica para la identificación y extracción de tos. A continuación, se describen los conceptos relacionados a las técnicas de segmentación y balanceo de clases utilizadas en este trabajo, así como las técnicas de aprendizaje automático y la representación de los datos utilizadas para la clasificación de la tos de pacientes sanos y enfermos de COVID-19.

2.1 COVID-19

El COVID-19 provocado por el virus del Coronavirus de tipo 2 causante del Síndrome Respiratorio Agudo Severo (SARS-CoV-2) (Zapatero Gaviria, 2023), tiene una transmisión fácil y contagiosa de humano a humano a través de aerosoles especialmente en espacios cerrados o mal ventilados (transmisión de aerosol). La transmisión puede ser directa por la convivencia física con personas infectadas, o por contacto indirecto, dónde se comparte un espacio el cual las áreas o superficies hayan sido comprometidas con el virus a través de las gotitas de saliva que esparcen los portadores del virus (fómites) (Leung, 2021).

La transmisión por aerosol podría contribuir a los eventos de superpropagación de las enfermedades respiratorias (Xiao, 2018; Fennelly, 2020). La tos es un síntoma característico de las enfermedades de las vías respiratorias y se define como un reflejo del tracto respiratorio que es provocado por la irritación mecánica y química de la mucosa de las vías respiratorias (Widdicombe, 2006). Uno de los principales medios de transmisión por aerosol es la tos, la cual se hace presente como síntoma principal de la enfermedad del COVID-19 de acuerdo a los trabajos de la tabla 1, la cual hace referencia a la sintomatología que se presenta en pacientes infectados por el virus del SARS-CoV-2.

De acuerdo a los trabajos presentados en la tabla 1 se observa que uno de los síntomas del COVID-19 es la tos. Una de las principales características de la tos asociada con COVID-19 es que es aguda o seca (no productiva) (Ijaz, 2022). Esto permite que las distintas características de la tos se puedan utilizar para el diagnóstico preciso de las enfermedades respiratorias basado en IA (Infante, 2017; Miranda, 2019; Soliński, 2020).

Tabla 1 Síntomas de COVID-19

Síntomas	Trabajo
Tos, fiebre, fatiga y dolor de cabeza	2020 (Pullen M. F., 2020)
Nauseas, **tos**, vómito y diarrea	2021 (Andrews P. L., 2021)
Fiebre, **tos** seca y dificultad respiratoria	2020 (Escudero, 2020)
Fiebre, **tos**, disnea, dolor muscular, dolor de garganta, dolor de cabeza, dolor en el pecho y dolor abdominal	2021 (Weng, 2021)
Fatiga, disnea, migraña, **tos**, dolor de cabeza, dolor en las articulaciones, dolor en el pecho, alteración del sueño, diarrea, alteración del ánimo.	2021 (Aiyegbusi, 2021)

2.2 Segmentación de datos acústicos de la tos

La segmentación de archivos de audio tiene como objetivo identificar cambios en la señal acústica para identificar y extraer información relevante en la información acústica de la señal. Es la técnica por la cual una señal acústica es dividida en segmentos o partes más pequeñas, donde cada una contiene información de audio según el objeto de interés (Theodorou, 2014).

La segmentación automática se basa en dividir las señales de audio en las categorías acústicas de interés con el uso de algoritmos automatizados para conjuntos de datos acústicos. En este trabajo se utilizó la categoría de silencio y tos. En donde, la etiqueta “silencio” se refiere al segmento de señal de audio en dónde existe ausencia de tos, por el contrario, la etiqueta “tos” se refiere al segmento de audio de interés.

Como se abordó en la el capítulo 1 subtema 1.2, realizar técnicas de segmentación en señales acústicas para detectar enfermedades respiratorias a través de algoritmos de aprendizaje automático con tos, es imprescindible para entrenar algoritmos con información valiosa, es decir con segmentos de tos y no de silencios, ruido, voces u otros tipos de sonido, dependiendo el origen de grabación de la señal acústica. El caso más sencillo de segmentación es tener tos y silencios. Sin embargo, puede existir una señal conformada de distintos tipos de sonidos que no son de interés, esto aumenta la dificulta del proceso de segmentación, pero la finalidad es la misma, detectar y extraer sonidos de tos únicamente.

Segmentar una señal acústica simplifica el procesamiento y permite capturar información localizada, reduciendo el costo computacional del entrenamiento de los modelos de aprendizaje automático enfocando el análisis a características relevantes.

A continuación, se describen las técnicas de segmentación automática de señales acústicas, utilizadas en este trabajo para la categorización de tos y silencio de un conjunto de archivos de audio, para su posterior identificación de fragmentos de toses. Se decidió utilizar la técnica por descomposición de modo empírico puesto que en el trabajo de (Andreu-Perez, 2021) la implementan como parte de su metodología propuesta para detección automática de COVID-19. La técnica de segmentación por comparador de histéresis digital, se utilizó porque el trabajo de (Orlandic, 2021) lo propone como una herramienta de análisis de tos de COVID-19. Y, por último, se utilizó la técnica de segmentación por cambio de intensidad e intervalos de detección de silencio, puesto que en el trabajo de (Zealouk, 2021) proponen el uso de esta técnica a través del software de Praat (Sauder, 2017) para análisis de tos de COVID-19.

2.2.1 Segmentación: Descomposición de Modo Empírico (EMD)

La Descomposición de Modo Empírico (EMD) identifica una serie de oscilaciones, en dónde cada una se componen a partir de los valores mínimos y máximos de la oscilación anterior. Cada una de estas oscilaciones cuenta con su propia escala de tiempo. Cada oscilación se deriva empíricamente de los datos de la observación del algoritmo y se conoce como Función Modal Intrínseca (IFM) (Rilling, 2003). En procesamiento de señales, una función empírica se refiere a una función obtenida de manera práctica o experimental a partir de los datos de una señal. A diferencia de las funciones teóricas o analíticas que se definen mediante una fórmula matemática, las funciones empíricas se construyen a partir de los datos de la señal sin hacer suposiciones previas sobre su forma matemática.

La EMD se puede utilizar para analizar señales no lineales y no estacionarias dividiéndolas. Se trata de descomponer señales acústicas (datos) en un número más pequeño de forma adaptativa (Huang N. E., 1998).

EMD se puede utilizar para localizar los modos que mejor reflejan los periodos de tos. Estos periodos se seleccionan con base en la “experiencia de la información que se obtiene del algoritmo” para detectar el sonido de la tos. El algoritmo se basa en la generación de envolventes de ajuste conformadas por picos máximos y mínimos de la señal, así como la media de los envolventes. En la figura 2 se muestra la señal acústica en color azul, y en color rojo el máximo y en azul el mínimo local de la señal para determinar los envolventes los cuales se identifican con verde.

Estos envolventes resultan en la extracción de la función empírica necesaria en la primera aproximación (IMF), para obtener el resto de funciones se identifica iterativamente nuevos máximos y mínimos de la señal obteniendo descomposiciones de la misma. Para la señal acústica de la tos, algunas IMF contiene información que son relacionada con los picos que se asocian a la tos, estas amplitudes o picos se calculan mediante la Transformada de Hilbert (Huang N. E., 1998), los modos seleccionados son 5° y 6° y estos dan el umbral para determinar si es tos o silencio dividiendo la señal original en segmentos de tos.

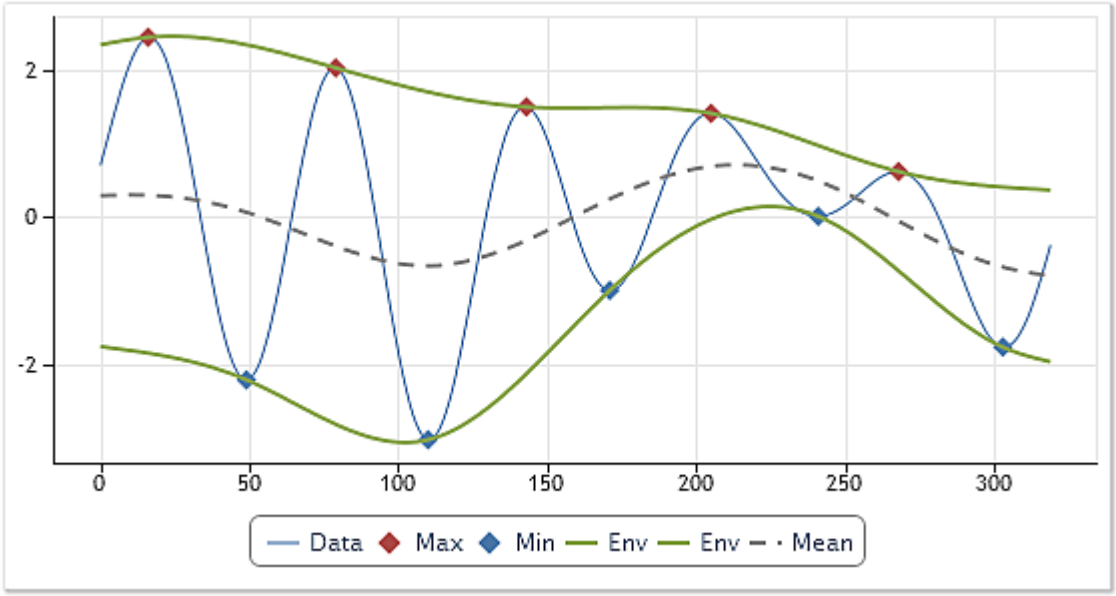


Figura 2 Envolventes de la señal acústica y sus medias tomada de (Ltd, 2012)

2.2.2 Segmentación: Comparador de histéresis digital

Esta técnica de segmentación se basa en un comparador de potencia de señales acústicas digitales, para encontrar los cambios dentro de un intervalo de tiempo de acuerdo a un umbral superior e inferior. Se define como potencia acústica, a la energía sonora que atraviesa una determinada sección por unidad de tiempo (Miyara, 2003).

Los comparadores de histéresis digitales, de una señal oscilan entre un umbral mínimo y máximo, cuando la señal (amplitud de la señal) está por debajo del mínimo la salida es un estado posible. Para este trabajo, se considera como silencio y el resto de la señal se considera como tos. Estos umbrales se definen de acuerdo a cada objeto de estudio. Para la tos se considera un intervalo de duración de 230 a 550 ms (Chang A. B., 2006). En la figura 3 se muestra un ejemplo de la señal acústica en donde el comparador identificaría tos y silencios en una señal.

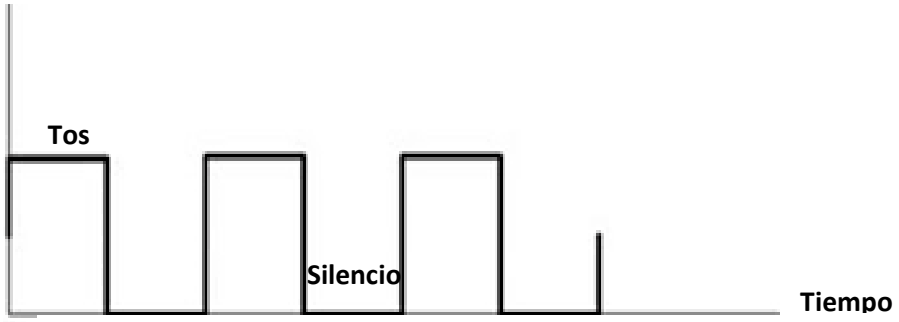


Figura 3 Señal digital acústica de la tos tomada de (Martínez, 2017)

2.2.3 Segmentación: Cambio de intensidad e intervalos de detección de silencio

La segmentación automática por cambio de intensidad e intervalos de detección de silencio se basa en el ajuste de parámetros para detectar los cambios en la señal acústica de la tos, a través de un software de análisis acústico de la voz, Praat (Correa Duarte, 2014). Del cual se obtiene un script que determina la tos y el silencio en intervalos de tiempo definidos, este script lleva por nombre TexGrid, éste puede ser utilizado como referencia para posteriormente segmentar de forma automática un conjunto de datos acústicos de toses. Dicho de otro modo, un TexGrid es el script con las instrucciones de inicio y fin de los segmentos de silencio o tos de una señal de audio. Los parámetros son los siguientes:

Parámetros de intensidad

- Tono mínimo (Hz): La frecuencia de periodicidad mínima en su señal. Si lo configura demasiado alto, terminará con una modulación de intensidad sincronizada con el tono. Si lo configura demasiado bajo, el contorno de la intensidad de la señal puede aparecer manchado, por lo que debe configurarlo tan alto como lo permita la señal si desea una señal sin ruido ni superposiciones de otros sonidos. El contorno es el nivel de intensidad de la señal de la onda de audio.

- Paso(s) de tiempo: El paso del tiempo del contorno de intensidad resultante. Si lo establece en cero, el paso de tiempo se calcula como un cuarto de la longitud efectiva de la ventana, es decir, como 0.8 / (*tono mínimo*).

Intervalos de detección de silencio

- Umbral de silencio (dB): determina el valor máximo de intensidad del silencio en dB respecto a la intensidad máxima. Por ejemplo, si imax es la intensidad máxima en dB, entonces la intensidad máxima del silencio se calcula como imax - silentThreshold; los intervalos con una intensidad inferior a este valor se consideran intervalos silenciosos.

- Intervalo mínimo de silencio (s): determina la duración mínima para que un intervalo se considere silencioso. Utilice un valor lo suficientemente grande.

- Intervalo mínimo de sonoridad (s): determina la duración mínima para que un intervalo no se considere silencioso. Esto ofrece la posibilidad de filtrar pequeñas ráfagas intensas de duración relativamente corta.

- Etiqueta de intervalo de silencio: determina la etiqueta para un intervalo de silencio en TextGrid.

- Etiqueta de intervalo de sondeo: determina la etiqueta para un intervalo de sonoridad en TextGrid.

2.3 Desbalanceo de clases

El desbalanceo de clases es la desproporción de datos que existe en los conjuntos de datos especialmente en los médicos, ya que existe menor número de datos para la clase positiva a la enfermedad (pacientes enfermos) en comparación con el mayor número de datos de clase negativa (pacientes sanos) (Fotouhi, 2019).

El desequilibrio de clases en un conjunto de datos es una de las características más comunes en las bases de datos médicas. La base de datos que se utilizarán en esta investigación se presenta en el capítulo 4 subtema 4.1 y esta se conforma de dos clases, una de muestras de tos de pacientes positivos y otra de muestras negativas a COVID-19. Aquí se observará que el número instancias por clase es mayor para la clase negativa que el número de instancias para la clase positiva. Esto es el desequilibrio de clases, el cual ocasiona sesgos en los modelos de aprendiza automático. Como se abordó en el capítulo 1 subtema 1.2 una de la propuesta de este trabajo de investigación es la utilización de técnicas de balanceo de clases para minimizar este problema.

Las técnicas de balanceo de clases utilizadas en este trabajo de investigación se describen a continuación. Las cuales se dividen en técnicas convencionales, aumento de datos y generación sintética de audio. Se decidió utilizar estas técnicas debido a la variedad de algoritmos distintos para balancear la base de datos. Las técnicas convencionales son las de los trabajos del estado de arte presentados en capítulo 3 subtema 3.1. El aumento de datos se basa en la generación de audio a partir de modificaciones de la onda de la señal, y la generación automática genera muestras nuevas sintéticas a partir de audios originales. Estas técnicas son utilizadas en los trabajos presentados en el capítulo 3 subtema 3.2.

La finalidad de probar tres tipos de técnicas diferentes es evaluar cuál obtiene mejores resultados de sensibilidad y especificidad para detección automática de COVID-19, considerando las técnicas que se utilizan en el estado del arte.

2.3.1 Técnicas convencionales de balanceo de clases

Las técnicas de balanceo de clases en conjuntos de datos desequilibrados que se consideran como base de diversas técnicas son las siguientes (García Abad, 2021) , las cuales se utilizan en este trabajo de investigación.

- **ROS (Random Oversampling):** El muestreo aleatorio, replica de aleatoriamente los datos de clase minoritaria para igualar la clase mayoritaria.

- **RUS (Random undersampling):** El submuestreo aleatorio, elimina de forma aleatoria, datos de la clase mayoritaria.

- **SMOTE (Synthetic Minority Oversampling Technique):** El sobremuestreo de minorías sintéticas, agrega datos a la clase minoritaria tomando en cuenta los datos originales y sus k vecinos más cercanos (número de instancias cercanas).

- **ADASYN (Adaptative Synthetic Sampling)**: El muestreo sintético adaptativo se basa en crear datos nuevos a partir de un numero definido de datos cercanos (k vecinos) añadiendo una distorsión a los datos para que estos sean más parecidos a los reales.

2.3.2 Aumento de datos

El aumento de datos o data augmentation es una técnica utilizada en el aprendizaje automático para aumentar el tamaño y la diversidad del conjunto de datos de entrenamiento. Consiste en aplicar transformaciones y manipulaciones a las muestras existentes para crear nuevas instancias que sean variaciones realistas de los datos originales (Ko, 2015). Esto con el objetivo de agregar muestras que cubran más un espacio de entrenamiento. Para este objeto de estudio se utilizan las siguientes técnicas de aumento de datos en la representación de forma de onda.

- **Noise**: Adición de ruido, agrega un tipo de ruido blanco o similar a la señal acústica. Agregar ruido blanco como muestras para el entrenamiento hace que el modelo sea robusto ante ruidos ambientales o ruido de fonda en las muestras.

- **Pitch scaling**: El escalamiento de tono, cambia la frecuencia o el tono del audio original sin cambiar la velocidad del sonido que percibe de la muestra original, este tono puede aumentar o disminuir.

- **Time strech**: El estiramiento de tiempo, cambia la forma de onda del audio original cambiando la velocidad del sonido sin modificar el tono, se puede acelerar o ralentizar el sonido percibido sin afectar la frecuencia de la muestra original.

- **Random gain**: La ganancia aleatoria, toma la forma de onda de la muestra original y la multiplica por un factor aleatorio, cambiando la amplitud, por ende, el volumen percibido.

- **Polarity inversión**: En inversión de polaridad, la señal original es multiplicada por -1, invirtiendo la polaridad de la onda, en donde, se intercambia el tipo positivo de la señal para su igual en negativo y viceversa.

2.3.3 Generación sintética

En el campo del aprendizaje automático y específicamente en redes neuronales profundas (deep learning) existen métodos para generar datos nuevos y realistas a partir de un conjunto de datos existente. Estos métodos se centran en el modelado y generación de datos. Los modelos generativos de deep learning se

entrenan para aprender la distribución subyacente de los datos de entrenamiento y luego son capaces de generar nuevas muestras que son similares a los datos originales.

Generación sintética de sonido por Autocodificadores Variacionales (VAE): Son modelos de aprendizaje automático que combinan redes neuronales con distribuciones de probabilidad (Doersch, 2016). Para generar datos sintéticos a partir de patrones de los datos originales. Con los VAE se pueden generar cualquier tipo de datos sintéticos tomando en cuenta que el conjunto de datos originales para entrenamiento de los modelos sigue un tamaño y contenido uniforme. En audio y acústica los datos originales y los datos generados sintéticos son espectrogramas o representaciones basadas en el análisis de la frecuencia de la señal a lo largo del tiempo (Largo, 2022).

La arquitectura de los VAE está constituida por dos arquitecturas de red, una para ser codificador el cual se encarga de comprimir la información, minimizando el número de neuronas por capa. Otra arquitectura de red que funciona como decodificador, en cada capa de red se aumenta el número de neuronas hasta que la última capa iguala el número de neuronas que la primera capa del codificador, entre estas dos arquitecturas se encuentra un espacio latente que se conceptualiza como el espacio mínimo en el cual los datos son representados dentro de una red neuronal.

En los VAE del codificador se obtiene una media y desviación estándar, esto permite generar una distribución Gaussiana que funciona como entrara al decodificador para poder generar muestras sintéticas similares a las reales. La arquitectura general de los VAE se observa en la figura 4.

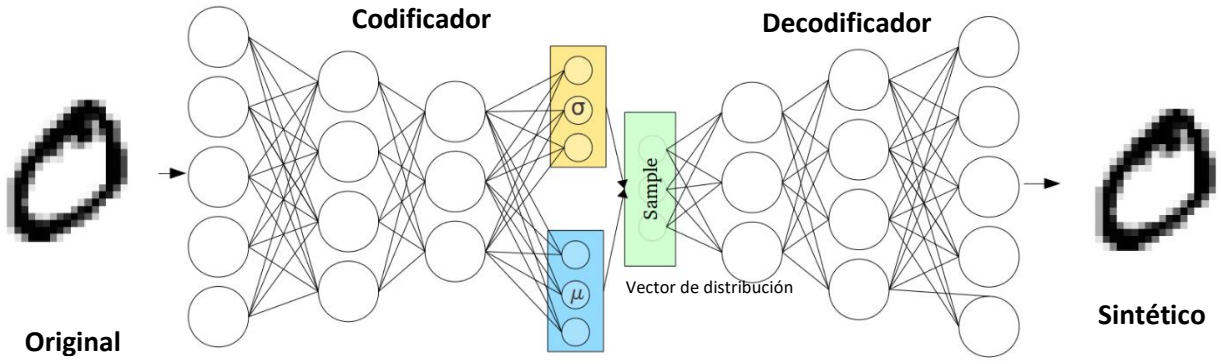


Figura 4 Arquitectura general de VAE tomada de (Caparrini, 2022)

2.4 Caracterización de la señal de audio: Representación

La caracterización de la señal de audio es la extracción de características, las cuales resultan ser las más dominantes y discriminatorias de una señal. Las características adecuadas expresan las propiedades de una señal de audio (Sharma G. U., 2020).

Los algoritmos de aprendizaje automático aprenden de la caracterización de los datos. En este trabajo se utilizan tres tipos de caracterización de la señal acústica de la tos. A continuación, se describen las técnicas utilizadas en este trabajo, en el siguiente capítulo se encuentra la búsqueda que permitió determinar la caracterización que obtiene mejor resultados al representar tos estudiando detección de COVID-19 por tos.

2.4.1 Técnicas de caracterización

Chroma: El espectro completo se asigna a 12 contenedores que representan los 12 semitonos (o Chroma) de la octava musical. Se puede calcular a partir de la transformada logarítmica de Fourier de tiempo corto de la señal de sonido. Las características Chroma se basan en añadir la información espectral que se relaciona con una clase de tono dada en un solo coeficiente.

Por ejemplo, en la percepción humana el tono es periódico, es decir, dos tonos se perciben casi iguales en "color", (que dos tonos sean parecidos en color significa que son armónicamente similares) sin embargo, son diferentes en una o varias octavas (en la escala que percibimos, se define como octava al grupo de 12 tonos, que van desde el Do hasta el Si) (Müller, 2015).

En la figura 5 se muestra un ejemplo en donde, en (a) se representan las octavas en un piano, en (b) su representación en un espectrograma, en (c) la representación en espectrograma de frecuencia logarítmica basado en tono, y en (d) su representación cromática, se considera la octava C3 en un el t=30 s el cual se observa enmarcado en el rectángulo verde.

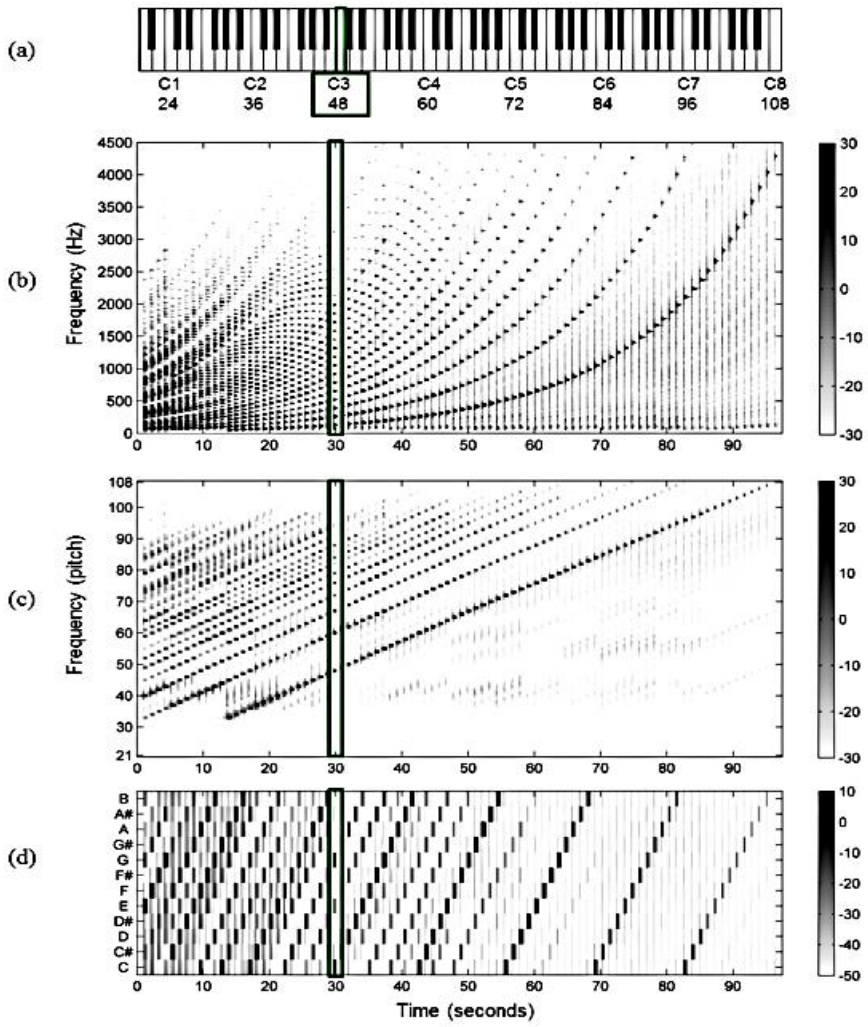


Figura 5 Representación para una grabación de piano de la escala cromática que va desde A0 (p=21) a C8 (p=108). (a) Teclas de piano que representan la escala cromática. (b) Espectrograma representación. (c) Espectrograma de frecuencia logarítmica basado en tono. (d) Representación del Chroma tomada de (Müller, 2015)

MFCC: Los Coeficientes Cepstrales de Frecuencia Mel representan el espectro de potencia a corto plazo de una señal de audio basado en la transformada de coseno discreta del espectro de potencia logarítmica en una escala de Mel no lineal (Rojo, 2011). El algoritmo es el siguiente:

- Enmarcar la señal en ventanas pequeñas.
- Para cada ventana, calcular la estimación del periodo del espectro de potencia.
- Aplicar el banco de filtros Mel al espectro de potencia, sumar la energía en cada filtro.

- Calcular el logaritmo de las energías del banco de filtros.

- Calcular la transformada de coseno discreta (DCT) de las energías del banco de filtros logarítmicos.

- Considerar n coeficientes DCT, descarte el resto.

Los MFCC son coeficientes para la representación del habla los cuales están basados en la percepción auditiva humana. Puesto que en la percepción humana las señalas de la frecuencia de los sonidos no siguen una escala lineal. Los MFCC representan las características locales de la señal de voz asociadas al tracto vocal (Rincón, 2007).

Espectrogramas de Mel: Un espectrograma es una representación gráfica del espectro de frecuencias de la emisión sonora (Mascorro, 2013). La escala Mel imita cómo funciona el oído humano.

El espectrograma de Mel tiene el objetivo de calcular las variaciones en frecuencias alejadas al rango vocal para que impacten menos en la imagen final del espectrograma (Elkfury, 2021). El algoritmo general para representar espectrogramas de Mel es el siguiente (leland, 2020):

- A una señal de audio se mapea desde el dominio del tiempo al dominio de la frecuencia con la Transformada Rápida de Fourier en segmentos de ventana de la señal superpuestos.

- Se convierte el eje y frecuencia a una escala logarítmica y la dimensión de color (amplitud) a decibelios para obtener el espectrograma.

- Se mapea el eje (frecuencia) en la escala de Mel para representar el espectrograma de Mel.

Combinación extensa de descriptores acústicos de bajo nivel para emociones (Emolarge): Es un conjunto de características acústicas que representan una señal de audio. Este conjunto consta de 6552 características (Schuller B. S., 2009). Las características se derivan de 25 descriptores de voz de bajo nivel incluyen intensidad, volumen, 12 coeficientes de la frecuencia de Mel, tono (F0) probabilidad de sonorización, envolvente F0, tasa de cruce por cero y 8 líneas frecuencias espectrales (Pérez-Rosas, 2017).

Combinación de descriptores acústicos de bajo nivel para fenómenos paralingüísticos (Is10_paraling): Es un conjunto de características que representan una señal de audio (Schuller B. S., 2010) utilizando 21

funciones estadísticas (como desviación estándar, media aritmética, asimetría, curtosis, etc.) Y 34 descriptores de bajo nivel (LLD) incluidas las características de MFCC, potencia logarítmica de bandas de frecuencia Mel, y el volumen como la intensidad normalizada, etc.) y se obtiene junto con su delta. Además, se añaden características relacionadas con el tono. Este conjunto de características conforma un total de 1582 descriptores de una señal acústica (Yazdani, 2021).

Conjunto Mínimo de Parámetros Acústicos de Ginebra (Gemapsvb01): Es un conjunto de 63 características que representan una señal de acústica (Eyben F. S., 2015). Es conformado por la frecuencia fundamental de la señal, el brillo, relación de armónicos con ruido, longitud del segmento sonoro, etc.

2.4.2 Técnicas de caracterización complementarias

La caracterización de la señal es importante puesto que permite la representación de las señales acústicas que contienen información específica. A continuación, describen las técnicas de caracterización de audio las cuales se mencionan en el capítulo 3 subtema 3.3. Se nombraron complementarias debido a que no son utilizadas como parte de la metodología de esta investigación. Sin embargo, se describen con la finalidad de presentar un contexto de las técnicas que podrían utilizarse para representar audio.

Tasa de cruce por cero (ZCR): Este caracterizador es del tipo temporal, a cada instante de tiempo se le asigna un valor obtenido por un micrófono llamándose “muestra” que tiene valores positivos y negativos, que serán utilizados para calcular el número de cruces por cero (Cantero, 2018). Número de veces que una señal cambia de signo dentro de un cuadro, lo que indica la variabilidad presente en la señal (Pahar, 2021).

Códigos predictivos lineales (LPC): El LPC calcula un espectro de potencia de la señal. Los coeficientes LPC obtenidos describen los formantes. Las frecuencias a las que los picos resonantes se producen se denominan frecuencias formantes (Dave, 2013).

Entropía de espectral: Es el cálculo de la distribución de potencia espectral junto con la previsibilidad de la señal de serie temporal (Devi, 2021).

Centroide espectral: Es el punto central de la distribución del espectro. En términos de percepción de audio humana, a menudo se asocia con el brillo del sonido (Xie, 2016).

Flujo espectral: Describe la variación del espectro en el tiempo, segmento a segmento. Esta se calcula como la diferencia al cuadrado entre las magnitudes normalizadas de los espectros consecutivos (Ozerov, 211).

Curtosis: Indica la cola de una densidad de probabilidad (DeCarlo, 1997).. Para las muestras de una señal de audio, indica la prevalencia de amplitudes más altas (Pahar, 2021).

2.5 Algoritmos de aprendizaje Automático

El aprendizaje automático (ML) es una rama en constate evolución de los algoritmos computaciones. El ML tiene como objetivo imitar la inteligencia humana a través del aprendizaje de la información del entorno circundante (Naqa, 2015). El ML permite que un sistema "aprenda" considerando los datos de entrenamiento para una determinada tarea. El entrenamiento de un modelo de ML es el aprendizaje por el cual un algoritmo aprende con base en los datos que recibe como entrada para tener respuestas que no son programadas, estas son calculadas y estimadas.

Los algoritmos de ML generalmente siguen un enfoque no lineal y no paramétrico, donde la complejidad del modelo se controla a través de uno o más hiperparámetros (Molnar, 2021)

En ML existen tres tipos de aprendizaje cómo se observa la figura 6 tomando en cuenta los datos de entrada a los algoritmos de ML para realizar el entrenamiento:

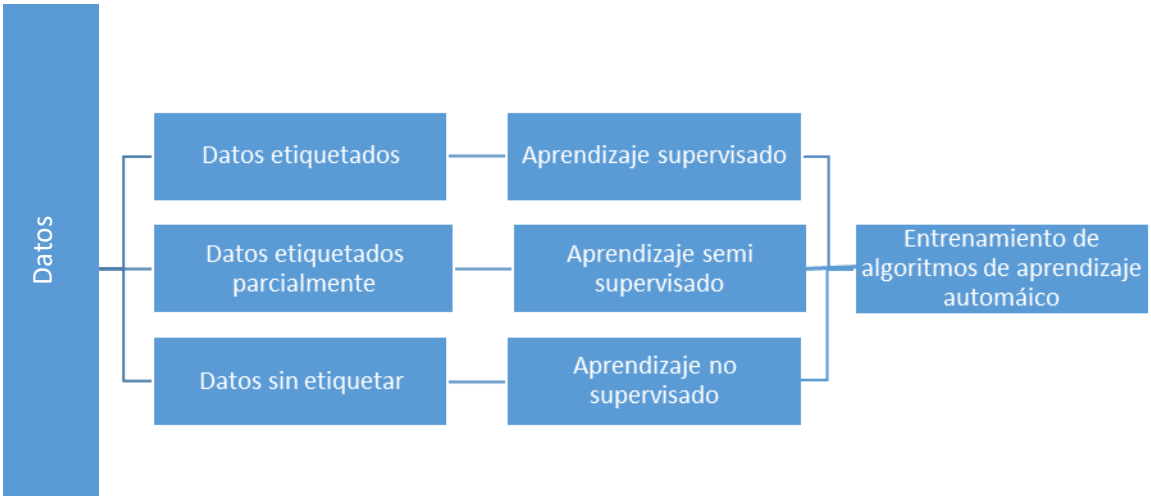


Figura 6 Categorías de algoritmos de aprendizaje automático

El aprendizaje supervisado se basa en estimar una respuesta desconocida a partir de muestras que el algoritmo conoce para realizar el entrenamiento de acuerdo a la etiqueta de las observaciones de entrada. Los datos con valores de resultado especificados se denominan "datos etiquetados". El aprendizaje semisupervisado ajusta los modelos a los datos etiquetados y no etiquetados. (Bi, 2019). En el aprendizaje no supervisado, el algoritmo identifica las relaciones y agrupaciones naturales dentro de los datos sin hacer referencia a ningún resultado o la "respuesta correcta" (Duda, 1973).

A continuación, se describen los algoritmos de aprendizaje automático que se utilizaron en este trabajo, considerados dentro de la rama de aprendizaje supervisado, ya que, los datos de entrenamiento están etiquetados. En el capítulo 4 se aborda la metodología en donde Random forest y la Red de Convolución Profunda (CNN) se utilizan para detectar COVID-19 previamente de haber realizado la segmentación y caracterización de la señal acústica de la tos.

Random forest: El algoritmo de bosque aleatorio son una combinación de predictores de árboles de modo que cada árbol depende de los valores de un vector aleatorio muestreado de forma independiente y con la misma distribución para todos los árboles del bosque (Breiman, 2001). Random forest permite utilizar conjuntos de datos que contiene variables continuas, como pasa en la regresión y variables categóricas como en el caso de la clasificación.

El algoritmo de bosque aleatorio es una colección de árboles de decisión, y cada árbol se compone de una muestra de datos que se extrae del total de conjunto de entrenamiento con reemplazo. De la muestra que se extrajo un parte se guarda como datos de prueba. Esto para hacer predicciones para cada árbol creado combinando N arboles de decisión.

En la figura 7 se muestra un ejemplo de la construcción de un bosque aleatorio combinando N arboles de decisión, para hacer predicciones para cada árbol creado.

CNN: Las redes neuronales de convolución están diseñada para procesar datos, que se toman de los datos multidimensionales, es decir, una imagen en color compuesta por tres datos 2D que incluyen la densidad de píxeles en los canales 3D (F. Demir, 2020).

La arquitectura de las CNN se conforma de un conjunto de capas o bloques, la cual cumple una función determinada, las cuales son:

1. Capa convolucional: conjunto de filtros convolucionales (núcleos). La imagen de entrada, expresada como métricas N-dimensionales, se convoluciona con estos filtros para generar el mapa de características de salida. Una convolución funciona como el filtrado de una imagen a través de una máscara. Diferentes máscaras obtienen resultados diferentes. Las máscaras son las conexiones entre neuronas de capas anteriores, así se aprenden características progresivamente.

2. Capa de agrupación: Es el submuestreo de los mapas de las características: reduce los mapas de características de gran tamaño a mapas de características más pequeños.

3. Función de activación: toma la decisión de disparar o no una neurona con referencia a una entrada en particular creando la salida correspondiente.

4. Capa totalmente conectada: Dentro de esta capa, cada neurona está conectada a todas las neuronas de la capa anterior, el llamado enfoque Totalmente Conectado (FC). La salida de esta capa es la salida final de la CNN.

En la figura 8 se muestran un ejemplo de las capas de una CNN.

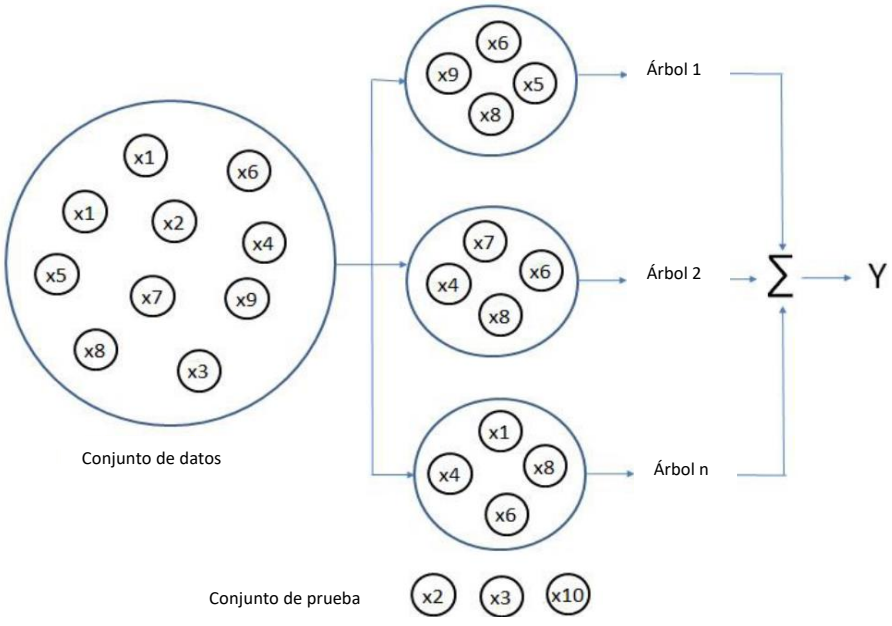


Figura 7 Algoritmo de Random forest tomada de (Espinosa-Zúñiga, 2020)

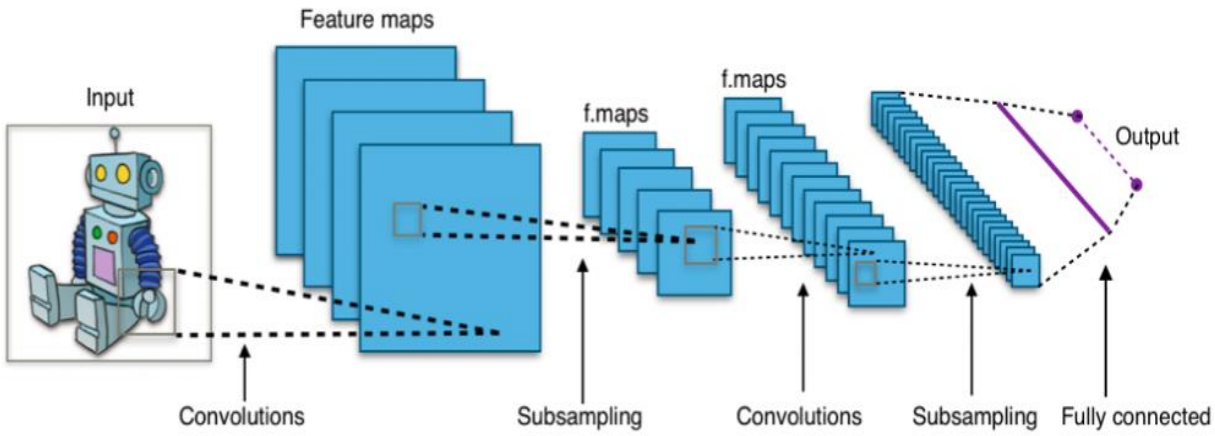


Figura 8 Ejemplo de red convolucional tomada de (Aphex34, 2015)

2.6 Métricas de evaluación

Las métricas de evaluación para medir el desempeño de los modelos probados que se utilizan en este trabajo se describen a continuación, donde:

El falso negativo (FN) ocurre cuando la prueba es negativa para un sujeto que posee el COVID. Falso positivo (FP) se define como el caso en el que el resultado predicho es COVID-positivo, pero el individuo es COVID-negativo. El verdadero positivo (TP) ocurre cuando la prueba predicha es positiva para COVID mientras que el sujeto también es positivo para COVID. El verdadero negativo (TN) ocurre cuando la prueba predicha es negativa para COVID y el sujeto también es negativo para COVID (Islam R. A.-R., 2022).

Estas métricas de evaluación se utilizan en el capítulo 5 para la evaluación de los algoritmos de aprendizaje automático utilizados.

Especificidad: La ecuación 1 representa la especificidad como la probabilidad de clasificar correctamente a un apersona sana. (Pita Fernández, 2003).

$$\boldsymbol{Especificidad} = \frac{\boldsymbol{Verdaderos\ Negativos}}{\boldsymbol{Verdaderos\ Negativos + Falsos\ positivos}} \quad (1)$$

Sensibilidad: La ecuación 1 representa la especificidad como la probabilidad de clasificar correctamente pacientes enfermos con resultado positivo en la prueba diagnóstica. La ecuación 2 representa la sensibilidad (Pita Fernández, 2003).

$$\boldsymbol{Sensibilidad} = \frac{\boldsymbol{Verdaderos\ Positivos}}{\boldsymbol{Verdaderos\ Positivos + Falsos\ Negativos}} \quad (2)$$

Exactitud: La ecuación 3 representa la precisión como la relación entre las observaciones predichas correctamente y el número total de observaciones evaluadas (Hossin, 2015).

$$\boldsymbol{Exactitud} = \frac{\boldsymbol{Verdaderos\ positivos + Verdaderos\ Negativos}}{\boldsymbol{Verdaderos\ Positivos + Falsos\ positivos + Falsos\ Negativos + Verdaderos\ Negativos}} \quad (3)$$

Medida F1: La ecuación 3 representa la Medida f1 tiene en cuenta tanto los falsos positivos como los falsos negativos. (Hossin, 2015). Esta medida se calcula usando la sensibilidad y la precisión de la ecuación 5.

$$\boldsymbol{Medida\ F1} = \frac{\boldsymbol{2 * Precisión * Sensibilidad}}{\boldsymbol{Precisión + Sensibilidad}} \quad \boldsymbol{(4)}$$

$$\boldsymbol{Precisión} = \frac{\boldsymbol{Verdaderos\ Positivos}}{\boldsymbol{Verdaderos\ Positivos + Falsos\ Positivos}} \quad \boldsymbol{(5)}$$

Matriz de confusión: Es la matriz con dimensión (n*n) donde n es el número de clases predictoras. Esta matriz se utiliza para evaluar el rendimiento de un modelo de aprendizaje automático y profundo, comparando los valores objetivo reales con los predichos por el modelo de aprendizaje automático (Borja-Robalino, 2020). En la figura 9 se ejemplifica la matriz de confusión.

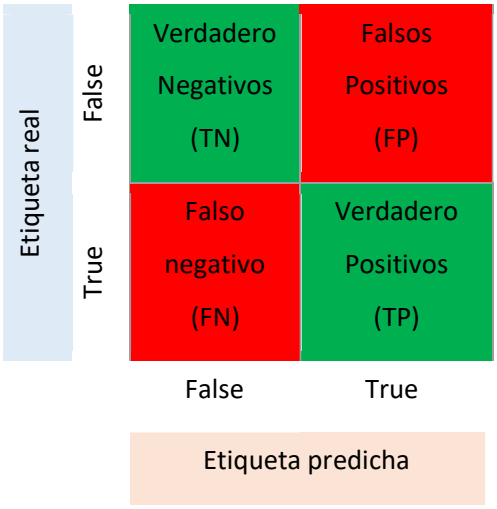


Figura 9 Matriz de confusión

Capítulo 3. Trabajos relacionados

En este capítulo se presenta los trabajos relacionados con la investigación. Se inicia con la búsqueda de la información relacionada con la clasificación automática de la tos, posteriormente, se exponen las técnicas que se implementan en otros trabajos para resolver el problema de desequilibrio de clases.

3.1 Clasificación automática de tos y técnicas de balance de clases

Se realizó una búsqueda de trabajos que abordaran detección automática de COVID-19 con el objetivo de identificar las diferentes técnicas de balanceo de clases, y segmentación que utilizaron.

La búsqueda de trabajos que realizan clasificación automática de la tos se hizo mediante el buscador de Google scholar e IEEE, utilizando las siguientes palabras clave: COVID-19, machine laerning, automatic classification, cough segmentation, data augmentation, class balancing, acoustic analysis of cough.

Se consideraron artículos recientes, del 2020 a la actualidad, artículos que abordarán el COVID-19 a través de análisis acústico de la tos; como criterios de selección.

En la tabla 2 se presentan los trabajos e investigaciones relacionadas a clasificación de COVID a partir de la tos.

Considerando los trabajos mencionados en la tabla 2, el trabajo de (Andreu-Perez, 2021) es uno de los trabajos actuales que aborda la segmentación, de señales acústicas de la tos para obtener solo la información necesaria del objeto de estudio, sin embargo, no se encontró evidencia que mencione el uso de técnicas de balanceo de clases. Cómo lo es en el caso de trabajos cómo (Mouawad, 2021), (Xia, 2021), (Pahar, 2021), en dónde especifican el uso de técnicas de balanceo de clases, las cuales entran en la categoría descrita como técnicas convencionales del capítulo 2, subtema 2.3.1. Esta es una de las principales diferencias entre los trabajos antes mencionados, y el desarrollo de esté, el cual radica en la prueba de diferentes técnicas de balanceo de clases, donde no sólo se centre un enfoque, puesto que no abordan técnicas de generación sintética de audio.

Tabla 2 Trabajos relacionados acerca de clasificación de COVID-19 por señal acústica de la tos.

Trabajo	Técnica de segmentación	Técnica de balanceo	Algoritmo de ML	Datos	Resultados
(Pahar, 2021)	Se eliminaron los períodos de silencio de la señal dentro de un margen de 50 ms utilizando un detector de energía simple.	• SMOTE • Borderline-SMOTE	• Regresión logística • KNN • SVM • CNN • Perceptrón multicapa • LSTM • ResNet50 • DNN	Muestras acústicas de la tos. • Conjunto de datos Coswara: 1079 sujetos sanos y 92 con COVID-19 positivo. • Conjunto de datos de Sarcos: 26 sujetos negativos para COVID-19 y 18 positivos para COVID-19.	• Coswara: Resnet50, sensibilidad del 93 % y una especificidad del 98 %. • Sarcos: selección directa secuencial codicioso (SFS) y un clasificador LSTM con sensibilidad del 91 % y una especificidad del 96 %.
(Xia, 2021)	Sin mencionar	• Sobremuestreo • Submuestreo	• CNN • SVM	Base de datos recopilada del (Departamento de Informática y Tecnología, Universidad de Cambridge, Reino Unido): Tras un control de calidad manual, 330 usuarios positivos con 469 muestras y 919 usuarios sanos con 2021 muestras.	Modelo de ensamble con CNN: Sensibilidad 68%, especificidad 69%.
(Mouawad, 2021)	Sin mencionar	• SMOTE • ADASYN • SMOTE: SMOTE + ENN • SMOTE + TL • Técnica de sobremuestreo de minoría ponderada por mayoría (MWMOTE) • Filtrado de datos sobremuestreados utilizando teoría de	• Arboles de decisión • SVM • RF • XGBoost	Base de datos de toses del proyecto Corona Voice Detect project in partnership with Voca.ai and Carnegie Mellon University: 1895 muestras sanas y 32 enfermas	XG-Boost con submuestreo con IR (Tasa de información): F1-score 0.62.

		juegos no cooperativos (NEATER) • Submuestreo informativo basado en un criterio de tasa de información			
(Islam, 2022)	Sin mencionar	No se aborda	• DNN	Base de datos Virufy, es una organización dirigida por voluntarios, que ha creado una base de datos global para identificar a los pacientes con COVID-19 que utilizan IA. Contiene las toses de los pacientes sanos y enfermos a COVID-19 además de la edad, el sexo y el historial médico. Virufy proporcionó 121 muestras de tos segmentadas de estos 16 pacientes.	Vector de características de dominio de frecuencia: 97.50% exactitud
(Andreu-Perez, 2021)	El algoritmo de detección de tos con las señales de audio filtradas se basa en la descomposición de modo empírico (EMD). Se aplica EMD para encontrar los modos que mejor reflejan los períodos de tos. Para posteriormente eliminar los silencios y obtener fragmentos de audio únicamente con tos.	No se aborda	• CNN con entrada 3D	Base de datos de toses acústicas con pacientes sanos y positivos a COVID-19, los datos son clínicamente validados.	Con entrada 3D en CNN: sensibilidad: 96.43 % ± 1.85 %, y especificidad: 96.20 % ± 1.74 %.

Una de las disyuntivas mencionadas en (Coppock, 2021), es la utilización de metodologías en trabajos de clasificación COVID-19 a través de ML, es que consideran información del entorno para entrenar modelos de aprendizaje automático, así como las características de los silencios que conforma un archivo de tos (por cómo se grabó), sin embargo, esto no aporta información valiosa puesto que se quiere entrenar modelos con información específica que sea sólo de tos. La segmentación es parte fundamental de este trabajo de investigación como se mencionó en el capítulo 1 subtema 1.2.

Los trabajos (Xia, 2021; Pahar, 2021; Andreu-Perez, 2021), utilizan CNN cómo algoritmo de aprendizaje automático, esto da a entender que uno de los modelos más utilizados se basa en representaciones espaciales.

El trabajo realizado en esta tesis es único y se distingue de otros estudios previos por su enfoque integral en la comparación y evaluación de técnicas tanto de segmentación como de balanceo. A diferencia de otros trabajos, donde se elige una única técnica de segmentación o balanceo y se centran principalmente en la etapa de caracterización y clasificación, esta investigación aborda el desafío de considerar ambas etapas críticas en el diagnóstico de enfermedades respiratorias agudas.

3.2 Técnicas de balanceo de clases mediante aumento de datos

Se realizó una búsqueda de trabajos que aborda el balanceo de clases con técnicas de generación de datos sintéticos para datos acústicos.

La búsqueda se hizo en Google scholar e IEEE, utilizando las siguientes palabras clave: data augmentation, signal acoustic, class balancing, augmentation of synthetic data.

Se consideraron artículos recientes, del 2019 a la actualidad, artículos que abordarán el aumento de datos por generación sintética, en problemas de desbalanceo de clases, con datos acústicos, además que fuesen artículos relevantes; como criterios de selección.

En la tabla 3 se presentan los trabajos que abordan generación de datos a partir de muestras de audio sintéticas para abordar el problema de desbalanceo de datos.

Tabla 3 Trabajos de generación de datos acústicos

Trabajo	Técnica de generación sintética	Datos utilizados	Algoritmo	Resultados
(Park, 2019)	Specaugment: un método simple de aumento de datos para el reconocimiento automático de voz. A través de modificación a espectrograma de señal de audio.	Conjuntos de datos de Fisher y Switchboard	SpecAugment, un método de aumento que opera en el espectrograma del audio.	Tasa de error de palabras (WER) del 6.8 %
(Saldanha, 2022)	Aumento de datos utilizando codificadores automáticos variacionales para mejorar la clasificación de enfermedades respiratorias	Base de datos de sonidos respiratorios: contiene 1864 grabaciones con crepitantes, 886 con sibilancias y 506 con sibilancias y crepitantes. De enfermedades respiratorias, obtenidas de 126 pacientes	Variantes de codificadores automáticos variacionales como el perceptrón multicapa VAE (MLP-VAE),	Precisión 98%
(Wei, 2020)	Aumento de datos por alteraciones en la forma de onda de la señal.	Base de datos Freesound Dataset Kaggle2018: contiene 11,073 audios de sonidos del entorno diario, con 41 categorías desequilibradas.	• Ruido blanco (Noise) • Estiramiento de tiempo (Time strech) • Cambio de tono (Pitch shift)	Precisión promedio media (mAP): • Noise: 92.73 % • Time Stretch 92.59 % • Pitch Shift 92.74 %

Considerando los trabajos antes mencionados en la tabla 3 se observa que el trabajo de (Saldanha, 2022), que utiliza los Codificadores Automáticos Variacionales expone el mejor de resultado de precisión con un 98%, considerando que el objeto de estudio habla de clasificación automática de enfermedades respiratorias, puede ser una herramienta prometedora. El trabajo de (Wei, 2020) considera el aumento de datos por la técnica de data augmentation (descritas en el capítulo 2, subtema 2.3.2) para equilibrar las clases de sonidos que se encuentran en el entorno diario, como sirenas de carros, ladridos, etc. Ambas técnicas de generación sintética por VAE y data augmentation se utilizan en este trabajo de investigación.

De los trabajos mencionados dos (Park, 2019; Saldanha, 2022), se basan en aumento de datos por modificación de las muestras existentes para generar muestras para clase minoritaria. Y un trabajo (Wei, 2020) aborda la generación sintética de audio para añadir muestras sintéticas a la clase minoritaria.

3.3 Descriptores acústicos para detección de COVID-19

Se realizó una búsqueda de trabajos que utilizan técnicas de caracterización de señales acústicas de la tos, para detectar a través de algoritmos de aprendizaje automático COVID-19. Con la finalidad de identificar las técnicas que obtienen mejores resultados, ya que es fundamental identificar aquellos descriptores que mejor representen la señal de una tos.

La búsqueda se hizo en Google scholar e IEEE, utilizando las siguientes palabras clave: feature acoustic, COVID-19, machine learning, características acústicas de tos y extraction feature acoustic for COVID-19.

Se consideraron artículos recientes, del 2020 a la actualidad, artículos que técnicas de caracterización acústica de la tos para detección de COVID-19; como criterios de selección.

En la tabla 4 en la primera columna se presentan los trabajos que abordan la caracterización de señales acústicas de la tos para detección automática de COVID-19. Adicionalmente, se exponen las técnicas que utilizaron para caracterizar el audio, y la última columna muestra la técnica y el número de características que se extraen, para tener el mejor resultado de predicción de COVID-19.

Tabla 4 Trabajos de clasificación automática de señal acústica de tos y las representaciones utilizadas

Trabajo	Características comparadas	Características con mejores resultados
(Pahar, 2021)	• Coeficientes cepstrales de frecuencia Mel (MFCC) • Energías del marco logarítmico, la tasa de cruce por cero (ZCR) • Curtosis	• 13 coeficientes MFCC
(Mouawad, 2021)	• Los coeficientes cepstrales de frecuencia de Mel (MFCC)	• 13 coeficientes MFCC
(Xia, 2021)	• Espectrogramas de Mel	• 128 espectrogramas de Mel

(Pahar, 2022)	• Coeficientes cepstrales de frecuencia Mel (MFCC) • Las energías del banco de filtros de registro espaciadas linealmente, junto con sus respectivos coeficientes de velocidad y aceleración • Tasa de cruce por cero de la señal (ZCR) • Curtosis	• El número de MFCC extraídos se encuentra entre 13 y 65, • El número de bancos de filtros espaciados linealmente entre 40 y 200
(Chang J. C., 2021)	• Coeficientes cepstrales de frecuencia Mel (MFCC)	• 39 coeficientes MFCC
(Vijayakumar, 2021)	• Espectogreamas de Mel • Coeficientes Cepstrales de Frecuencia Mel (MFCC)	• 40 MFCC
(Tena, 2022)	• 39 características: espectral instantánea, frecuencia instantánea, frecuencia instantánea de pico e información espectral, frecuencia media del espectro total, las entropías conjuntas, instantánea • La curtosis • 3 momentos conjuntos tiempo-frecuencia • 3 momentos conjuntos de las señales marginales de potencia instantánea y densidad espectral.	• 15 características a partir de la representación de frecuencia temporal de cada muestra de tos: energía espectral instantánea, frecuencia instantánea, frecuencia instantánea de pico e información espectral
(Imran, 2020)	• Coeficientes Cepstrales de Frecuencia Mel (MFCC) • Proyecciones de análisis de componentes principales (PCA) de las funciones de MFCC en todos los fotogramas	-
(Islam R. A.-R., 2022)	• Las características del dominio del tiempo (a) Distribución de energía de tiempo corto, (b) Tasa de cruce por cero de tiempo corto, y (c) Entropía de energía. • Las características del dominio de frecuencia (a) Centroide espectral, (b) Entropía espectral y (c) Flujo espectral • Las características del dominio de frecuencia (a) caída espectral, (b) coeficiente MFCC, (c) vector cromático y (d) armónicos de características.	• Vector de características de dominio de frecuencia.
(Hoang, 2022)	• Coeficientes Cepstrales de Frecuencia Mel (MFCC), • Cromáticos (Chroma), • Espectrograma Mel (Mel)	• 64 coeficientes cepstrales de frecuencia Mel (MFCC), • 12 cromáticos (croma), • 128 espectrograma Mel (Mel)
(Amal, 2022)	• LPC	• 12 LPC
(Islam R. A.-R., 2021)	• Chroma	• Chroma

De los trabajos anteriores de la tabla 4 se observa que existe una variedad de técnicas de caracterización de audio, dónde cada una de ellas resalta información diferente como se describieron en el capítulo 2, subtema 2.4.2, la mayoría de trabajos se centran en utilizar MFCC y Espectrogramas de Mel para caracterizar la tos. En el capítulo 4, subtema 4.4 se presenta un análisis más detallado acercade las técnicas de representación de audio más utilizadas por el estado actual del arte en detección automática de COVID-19 por señales acústicas de la tos. La finalidad de este análisis y búsqueda de trabajos que abordan la caracterización de la tos es diseñar nuestro propio conjunto de descriptores a través de las técnicas de caracterización más utilizadas en el estado del arte.

Capítulo 4. Metodología

En este capítulo se presenta la propuesta de la metodología integral diseñada para detección automática de COVID-19 por tos. Se inicia con la descripción de la base de datos que se utilizó en este trabajo, posteriormente se describen otras bases de datos disponibles de señal acústica de tos para detección de COVID-19. Se continua con la exposición de diferentes técnicas de preprocesamiento, representación y técnicas de balanceo de clases para los datos.

4.1 Conjunto de datos

En esta sección se presenta y describe la Base de Datos (BD) que se utilizó en esta investigación. Además de la descripción de otras bases disponibles acerca de tos para detección de COVID-19.

En este subtema se aborda el punto “Bases de datos de la tos de personas en audio” de la metodología integral propuesta en el capítulo 1, subtema 1.6.

4.1.1 Base de datos utilizada (CICESE)

La base de datos utilizada en este proyecto es proporcionada por la Unidad Tepic del Centro de Investigación Científica y de Educación Superior de Ensenada (CICESE-UT3), la cual, está conformada por datos de tos validados clínicamente. La recopilación se llevó a cabo en colaboración con el Laboratorio Nacional de Investigación en Seguridad Alimentaria (LANIIA) en Nayarit, México. Los datos se comenzaron a recopilar en México el 4 de abril de 2020 y se culminó con la recolección el 21 de septiembre de 2020. Los participantes de las grabaciones que conforman la base de datos, mayormente pertenecen a la comunidad de La Universidad Autónoma de Nayarit.

Los protocolos clínicos y de ética de la investigación se aprobaron por los comités de ética institucionales locales (Código: BIOETIC_HUM_2020_02, México; Código: APP_Covid19_03042020, España). La Unidad de Nayarit y el hospital de Málaga son ambos centros acreditados para el diagnóstico molecular de la Covid-19 y además cuentan con la certificación ISO 9001 (Andreu-Perez, 2021). Las muestras de tos se recolectan de pacientes que acuden a las instituciones antes mencionadas, para una prueba qRT-PCR para la detección de COVID-19.

La base de datos se conforma por muestras de audio de tos de total de 1105 pacientes, de los cuales 378 son pacientes positivos a COVID-19 y 727 son pacientes negativos a COVID-19. En la figura 10 inciso a se presenta el porcentaje del sexo correspondiente a los pacientes positivos. De los cuales se observa que en la mayoría son masculinos con un 53% del total de pacientes. En la figura 10 inciso b se presenta el porcentaje del sexo correspondiente a los pacientes negativos. De los cuales se observa que en la mayoría son masculinos con un 53% del total de pacientes.

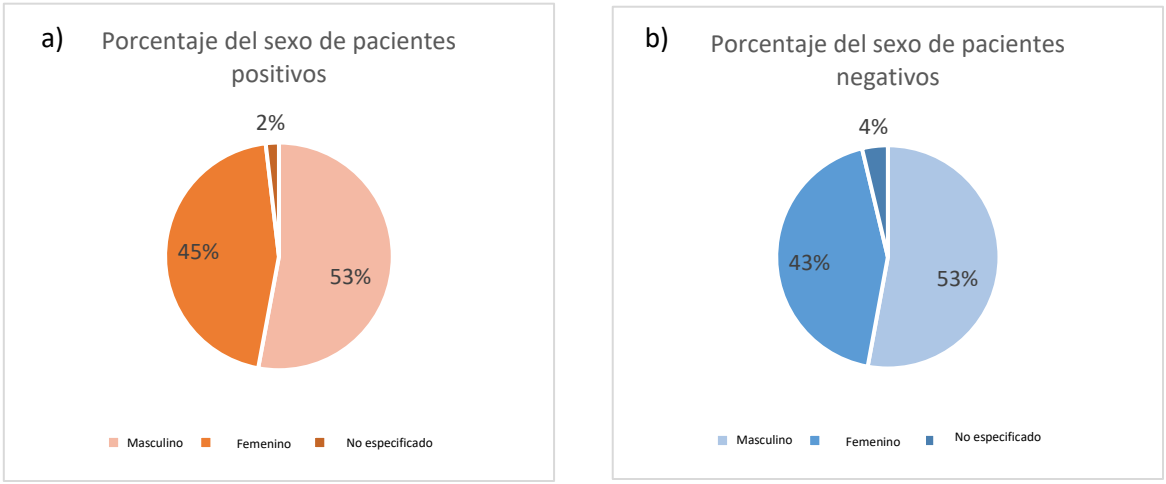


Figura 10 Porcentaje de sexo correspondiente a pacientes, (a) positivos a COVID-19, (b) negativos a COVID-19

La edad promedio de los pacientes negativos es de 38.74 años con una desviación estándar de 13.59. La edad promedio de los pacientes positivos es de 39.44 años con una desviación estándar de 14.24. Se puede corroborar esta información en la figura 11 inciso a, donde se observa histograma de las edades para los pacientes positivos, el paciente más joven tiene 10 años y el mayor 80. En la figura 11 inciso b, el histograma de las edades para los pacientes negativos, el paciente más joven tiene 10 años y el mayor 80.

Esta base de datos, presenta como ventaja la resolución de algunos de los problemas presentados en el capítulo 1 subtema 1.2, que el artículo (Coppock, 2021) aborda como deficientes en las recientes investigaciones de detección automática de COVID-19. Cómo ya se mencionó, las muestras recopiladas son clínicamente validadas por la prueba qRT-PCR. Al momento que se realizaron las grabaciones de las toses, los participantes no conocían su estado de COVID-19, puesto que el resultado se les fue entregado después de realizar la grabación, esto ayuda a que los sonidos de la tos no sean alterados por emociones del participante.

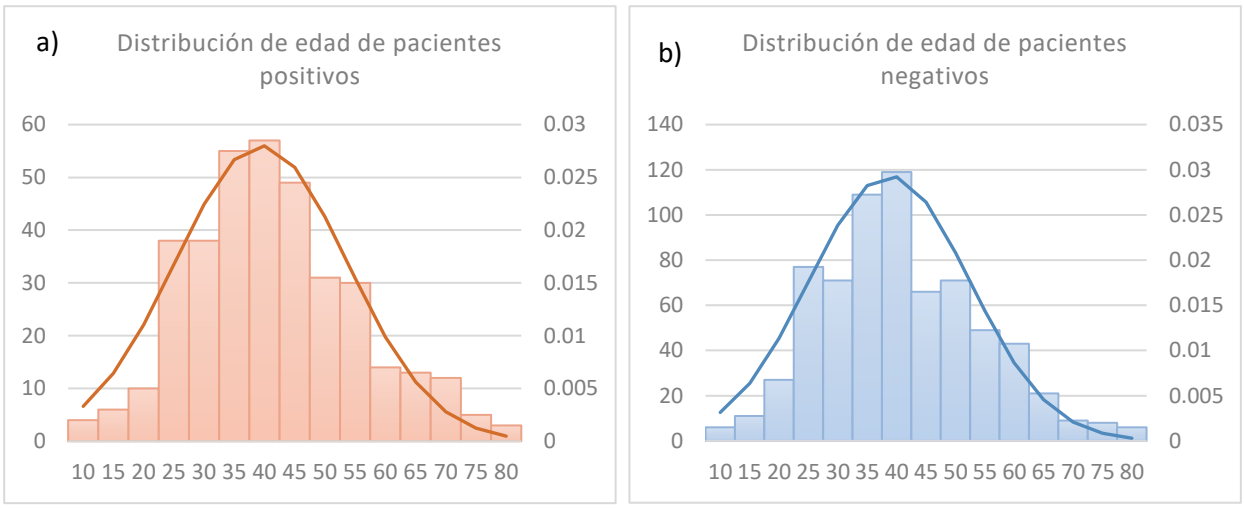


Figura 11 Distribución de edad de los pacientes, a) Positivos a COVID-19, b) Negativos a COVID-19

Esta base de datos proporciona si las personas sanas y enfermas de COVID-19 al momento de hacer la grabación tenían o no síntomas, esto con la finalidad de no descartar muestras, que, aunque las personas sanas presentaran síntomas se considerara esa información, en caso contrario también se consideraron a las personas positivas a COVID-19 que no presentaran síntomas al momento de la grabación. En la figura 12 inciso a, se muestra la proporción de personas enfermas de COVID-19 con y sin tos, y en la figura 12 inciso b, la proporción de personas sanas con y sin tos.

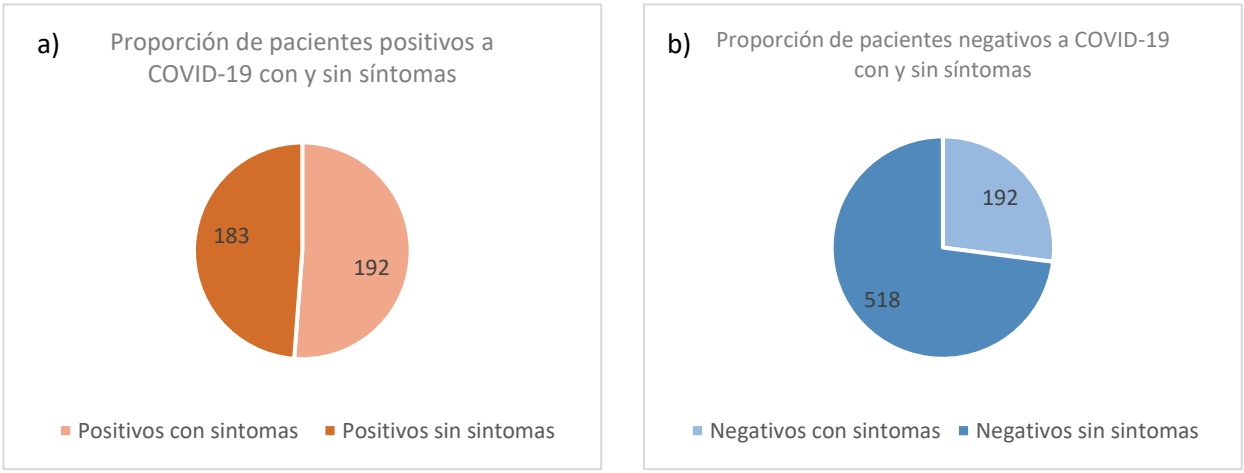


Figura 12 Proporción de pacientes, a) Positivos a COVID-19 con y sin síntomas, b) Negativos a COVID-19 con y sin síntomas

4.1.1.1 Base de datos complementarias

Se realizó una búsqueda de bases de datos disponibles de audio de la tos de personas positivas a COVID-19 y negativas. Con la finalidad de realizar pruebas con la metodología propuesta con otras bases de datos. De lo cual se encontraron las siguientes bases de datos: BD Buenos Aires, Argentina y Kaggle.

4.1.1.2 Base de datos Buenos Aires, Argentina

La base de datos Argentina se encuentra disponible en el portal de la secretaria de Innovación y Transformación Digital. Jefatura de Gabinete de Ministros de Buenas Aires (SITD, 2021). Se conforma por archivos de audio de toses provenientes de grabaciones por celular, categorizadas por COVID-19 positivo y negativo según resultado de test RT-PCR. Los datos se recaudaron en septiembre y octubre del 2020. En total participaron 2,771 personas, de las cuales 1393 son negativas y 1378 son positivas a COVID-19.

En la figura 13 inciso a, se presenta el porcentaje del sexo correspondiente a personas positivas. Y en la figura 13 inciso b, se presenta el porcentaje del sexo correspondiente a personas negativas a COVID-19. El sitio no brida información precisa acerca del protocolo de recolección de datos.

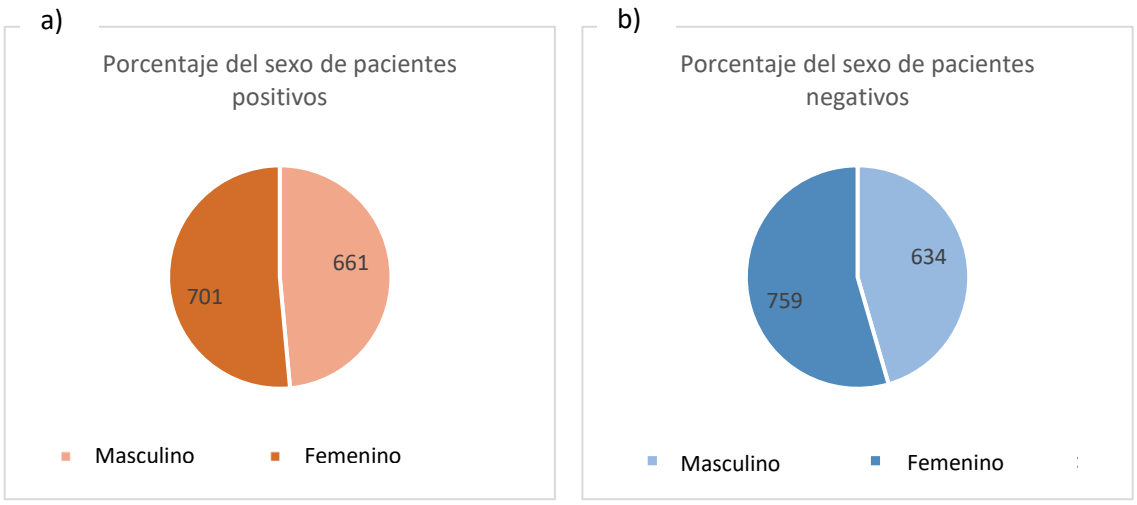


Figura 13 Porcentaje de sexo, a) pacientes positivos a COVID-19, b) pacientes negativos a COVID-19

4.1.1.3 Base de datos Kaggle

La base de datos de Kaggle (Kaggle, 2020) está compuesta por toses de personas que compartieron su diagnóstico a COVID-19 en la plataforma de Kaggle. Hay muestras de un total de 118 personas, 105 negativas y 11 positivas COVID-19. No se presenta mayor información acerca del protocolo de recolección.

La representación de las muestras de toses se debe caracterizar a una representación que sea eficiente y descriptiva para los algoritmos de aprendizaje automático. Por ende, se realizó una búsqueda de trabajos de clasificación automática de sonidos de la tos, los cuales se encuentran descritos en la tabla 4. Para que continuamente se indicaron las representaciones de audio que mejores resultados han obtenido según los trabajos encontrados.

4.1.1.4 Diferencia base de datos CICESE, Buenos Aires, y Kaggle

La principal diferencia de esta base de datos con las bases de datos complementarias descritas, radica en que las muestras que conforman a las BD complementarias no son clínicamente validadas por la prueba qRT-PCR. Al contrario de la BD utilizada en este trabajo, la cual es la de CICESE, que sí son muestras de pacientes positivas a COVID-19 clínicamente validadas por la prueba qRT-PCR, además los pacientes no sabían su estado de COVID-19 al hacer la grabación y también incluyen muestras con y sin síntomas. Las muestras de Buenos Aires y Kaggle no son clínicamente validadas, si no que, la validación de los datos recae en la veracidad de la persona que proporciona las muestras a través de una plataforma online.

Se decidió utilizar el conjunto de datos de CICESE el cual reúne 4 aspectos positivos de los problemas encontrados por (Coppock, 2021) de los conjuntos de datos para detección automática de COVID-19 por tos, vistos en el capítulo 1 del subtema 1.2. Estos aspectos son:

- Muestras clínicamente validades.
- Los pacientes no conocen su estado de salud al momento de hacer la grabación.
- Fácil acceso al conjunto de datos.

- El conjunto de datos no sólo se conforma de muestras de personas sana o enfermas de COVID-19, las muestras recopiladas por el conjunto de datos a utilizar, se conforma por grabaciones de toses de personas sanas con y sin síntomas de COVID-19, y grabaciones de personas sanas con y sin síntomas de COVID-19.

4.2 Preprocesamiento de señal acústica: Segmentación

Con el preprocesamiento se pretende aplicar técnicas que permitan mejorar y homogenizar los diferentes formatos y características de los datos. A continuación, se describen las técnicas de segmentación que se utilizaron en los audios de la base de datos CICESE-UT3. La segmentación se utiliza para identificar en la muestra de tos las partes de interés. Es decir, de los audios completos extraer información necesaria para detectar el COVID-19, es decir los fragmentos en donde sí hay tos.

En este subtema se aborda el punto “Preprocesamiento” de la metodología integral propuesta en el capítulo 1, subtema 1.6.

Segmentación: Comparador de histéresis digital

Se utilizó el código del repositorio detect segment cough (Orlandic, The COUGHVID crowdsourcing dataset, a corpus for the study of large-scale cough analysis algorithms, 2021). El algoritmo que sigue la detección y segmentación de tos es el siguiente:

1. La señal se normaliza al rango [−1, 1], y se reduce la muestra a 12 kHz.

2. Identifica regiones de la señal con picos rápidos de potencia.

3. Dado que la fase respiratoria de la tos tiene una duración de 230-550 ms (Chang A. B., 2006), se descarta cualquier sonido de tos que dure menos de 200 ms.

4. Se considera los 200 ms antes y después de un sonido de tos como parte de la tos (Chang A. B., 2006).

5. Se recortan los segmentos identificados de tos.

Segmentación: Descomposición de modo empírico (EMD)

Para la detección de tos en las muestras de audios de pacientes positivos y negativos a COVID-19 se utilizó el algoritmo basado en Descomposición de Modo Empírico (EMD), que divide una secuencia en un conjunto de secuencias más pequeñas, denominadas funciones de modo intrínseco (IMF) (Andreu-Perez, 2021). Para implementar la técnica de segmentación basada en EMD se utilizó la librería de Matlab EMD (MathWorks, 2023). El algoritmo general de segmentación es el siguiente:

1. Los datos de sonido sin procesar se filtran en paso bajo con una frecuencia de corte de 1kHz.

2. Un filtro Chebyshev tipo 2 de segundo orden con una frecuencia de transición de10Hz se aplica para retener el sonido de tono alto de la tos mientras se atenúan los sonidos de fondo simultáneamente.

3. Se utiliza la librería de EMD (Huang N. E., 1998) de Matlab para detección de tos.

4. Se ignora la primera detección máxima de amplitud máxima en el sonido.

5. Se cortan las posteriores detecciones máximas de amplitud en la onda del sonido.

Segmentación: Cambio de intensidad e intervalos de detección de silencio

Se utilizo la herramienta Praat el cual es un software para el análisis fonético del habla, en la cual se puede analizar el espectrograma de sonidos grabados (Correa Duarte, 2014). El algoritmo para detección toses y segmentación es el siguiente:

1. Selección de muestras para ajuste de parámetros: Seleccionar una muestra de las clases que conforman en conjunto de datos. En este trabajo de investigación se seleccionaron de forma aleatoria 6 muestras, 3 muestras de audio de la clase negativas y 3 muestras de audio de clase positiva.

2. Ajuste de parámetros: En la interfaz de Praat se cargaron las muestras de sonido y se detectaron los parámetros de cada uno de los audios en los que se detectaba los segmentos de silencio y de sonido para generar un script que determina los fragmentos de silencios y

sonidos. Este script es llamado como TextGrid, el cual se define como las instrucciones de inicio y fin de un segmento de silencio o tos en una muestra acústica.

3. Registro y promedio de parámetros de muestras: Registrar los parámetros de cada muestra, en la tabla 5 se encuentran los parámetros correspondientes a cada muestra. Posteriormente se calculó la media de cada uno de los parámetros.

Tabla 5 Parámetros personalizados de ajuste para generar TextGrid de silencios

Archivo	Tono mínimo (HZ)	Paso mínimo (S)	Umbral de silencio (DB)	Intervalo mínimo de silencio (S)	Intervalo mínimo de sondeo(S)
202_13072020_Negativo	110	0.0=auto	-15	0.1	0.1
196_13072020_Positivo	101	0.0=auto	-20	1	0.1
001_17-06-2020_Negativo	100	0.0=auto	-20	0.1	0.1
183_13072020_Positivo	70	0.0=auto	-23	0.01	0.1
1043_28082020_Negativo	105	0.0=auto	-25	0.1	0.1
921_18092020_Positivo	120	0.0=auto	-30	0.001	0.1
Promedio	101	0.0=auto	-22.166667	0.2185	0.1

4. Extracción de tos: Después de obtener parámetros promedio de la tabla 5, se generó un TextGrid con los mismos, para que con ayuda del lenguaje de programación Python se cortaran los fragmentos en donde en el archivo de audio se detectara sonido de todas las muestras de audio negativas y positivas.

4.2.1 Evaluación de técnicas de segmentación

En este trabajo se propone una técnica de evaluación de segmentación, con la finalidad de comparar a través de distintas métricas como la sensibilidad y especificidad de las técnicas de segmentación utilizadas y así identificar la que obtiene mejores resultados.

A continuación, el procedimiento propuesto para evaluar distintas técnicas de segmentación de toses es el siguiente:

1. Elegir una muestra representativa el conjunto de datos: seleccionar de forma aleatoria 15 audios de la clase positiva y 15 audios seleccionados de forma aleatoria de la clase negativa. En total una muestra de 30 audios.

2. Segmentar de forma manual: con ayuda de la herramienta Praat para análisis de audio, cargar la muestra de los 30 audios. Seleccionar cada audio y abrir ventana de ver y editar. En la interfaz cómo se muestra en la figura 18, seleccionar la señal de audio y reproducirla para escuchar e identificar los segmentos de tos, considerando de igual forma el espectrograma de amplitud de la seña, seleccionar segmento donde se identificó tos y agregar un nivel de intervalo añadir un nivel de intervalo en la pestaña tier. Agregar una etiqueta como identificador que se encontró tos, en este caso sound cuando hay tos y silent cuando no hay tos. Guardar el TextGrid en la pestaña de file, esto hará un scrpit identificando los intervalos de silencio y tos del archivo de audio en cuestión. (Repetir este paso con los 30 audios)

3. Realizar segmentación automática: hacer la segmentación del conjunto de datos de audio con las técnicas vistas en capítulo 4 subtema 4.2. y obtener el espectrograma de amplitud de la señal segmentada. Identificar los audios segmentados seleccionados como muestra del paso 1.

4. Comparar: Después de realizar la segmentación automática y manual, realizar la comparación especifica de cada segmento identificado como tos en el espectrograma de la señal original segmentada de forma manual y automática. Se considera la siguiente terminología para identificar si un segmento de la técnica automática concuerda o no con el segmento identificado de forma manual.

    a. VP: si hay intersección entre la tos segmentada manual y automática

    b. VN: si la segmentación manual y automática identificaron ausencia de tos (silencio)

    c. FP: si la segmentación automática detecto tos cuando hay silencios considerando la manual.

    d. FN: si la segmentación manual no detecto tos en el segmento, cuando en realidad si hay.

5. Por cada segmento de tos identificado como silencio o de tos de forma automática comparar con la detección manual.

6. De las 30 muestras hacer un conteo de VP, VN, FP, y FN, calcular métricas de capítulo 2, subtema 2.6. Realizar evaluaciones y determinar los mejores resultados de técnica de segmentación utilizada.

En la figura 14 se observa la interfaz de Praat para ver y editar un TextGrid de un archivo de audio (.wav). en el inciso a, se considera la segmentación manual que se tomará como punto de referencia para determinar las coincidencias en los segmentos de técnicas automáticas.

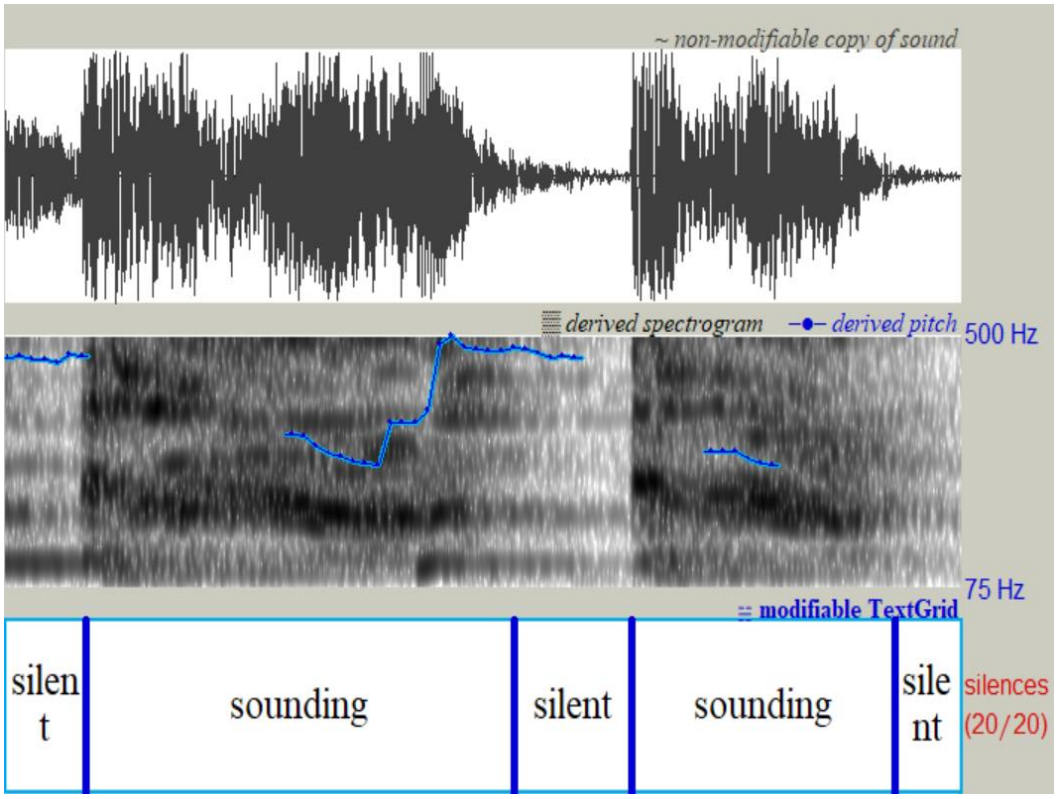
4.3 Marcos de tiempo

Los algoritmos de aprendizaje automático comparan vectores de características con la misma dimensionalidad para realizar el entrenamiento. Los segmentos de toses resultantes del procedimiento de segmentación tienen duraciones variables, es por ello que es importante definir un marco de tiempo específico, o una “ventana” que se considere como favorable para que las señales acústicas caigan entre esta ventana. Esta ventana o marco de tiempo será la pauta de duración de todas las muestras con las que se alimentará a los algoritmos de aprendizaje.

El marco de tiempo tiene que ser específico para cada conjunto de datos puesto que está dado por el fenómeno a estudiar, por ejemplo, si el análisis del habla se hace a nivel de fonema, de sílaba, de palabra o de frase. En este caso debemos encontrar una duración adecuada para la tos.

El cálculo del marco de tiempo va relacionado con la segmentación de las muestras, ya que muestras con tamaño variable hacen que el marco de tiempo englobe muestras poco típicas, es por ello que se busca que la duración de los segmentos presenten una baja desviación estándar.

a)



b)

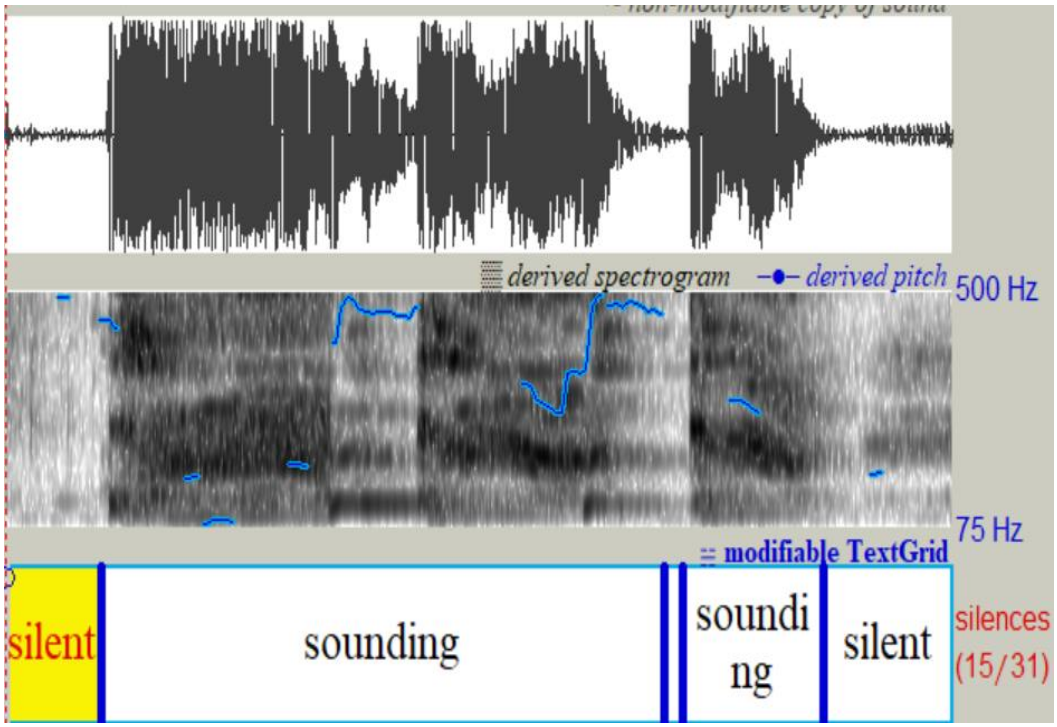


Figura 14 Interfaz de Praat de archivo .wav y TextGrid para identificación de segmentos de silencio y tos. Archivo de audio "017_22-06-2020_Negativo" (a) segmentación manual, (b) segmentación automática.

El procedimiento para encontrar el tamaño de la ventana adecuada para un conjunto de datos acústicos es el siguiente:

1. Calcular duración de audios, detectar máximos y mínimos de duración de audios, además frecuencia de muestreo.

2. Determinar si los datos siguen una distribución normal.

3. Establecer el intervalo de confianza de 5% de significancia.

4. Calcular ventana con ecuación 6. Dónde, max es la duración máxima en segundos de los segmentos de un conjunto de datos, y fs corresponde a la frecuencia de muestreo de los segmentos.

$$\boldsymbol{Marco\ de\ tiempo} = \left(\left|\frac{\boldsymbol{max}*\boldsymbol{fs}}{\mathbf{512}}\right|\right) + \mathbf{1} \tag{6}$$

4.4 Caracterización de señal de audio: Representación

La caracterización del audio es un proceso fundamental para la clasificación de cualquier tarea, ya que es la representación de un archivo de audio en un vector numérico que funciona como entrada a un modelo de ML para que sea posible el entrenamiento.

En este subtema se aborda el punto “Representación” de la metodología integral propuesta en el capítulo 1, subtema 1.6.

En la figura 15 se presenta el conteo de las técnicas de caracterización de audio que se mencionaron en la tabla 4 del capítulo 3 que abordan la detección de COVID-19 con señales acústicas. Se observa en la barra de color azul que las representaciones que más se utilizan en los trabajos son los MFCC, Chroma, y Espectrogramas de Mel.

Los MFCC, Chroma, y Espectrogramas de Mel son las técnicas de caracterización que mejores resultados obtuvieron para la clasificación de tos, es por ello que en este trabajo se utilizaron estas 3 técnicas de caracterización de audio con tosidos de personas positivas y negativas a COVID-19

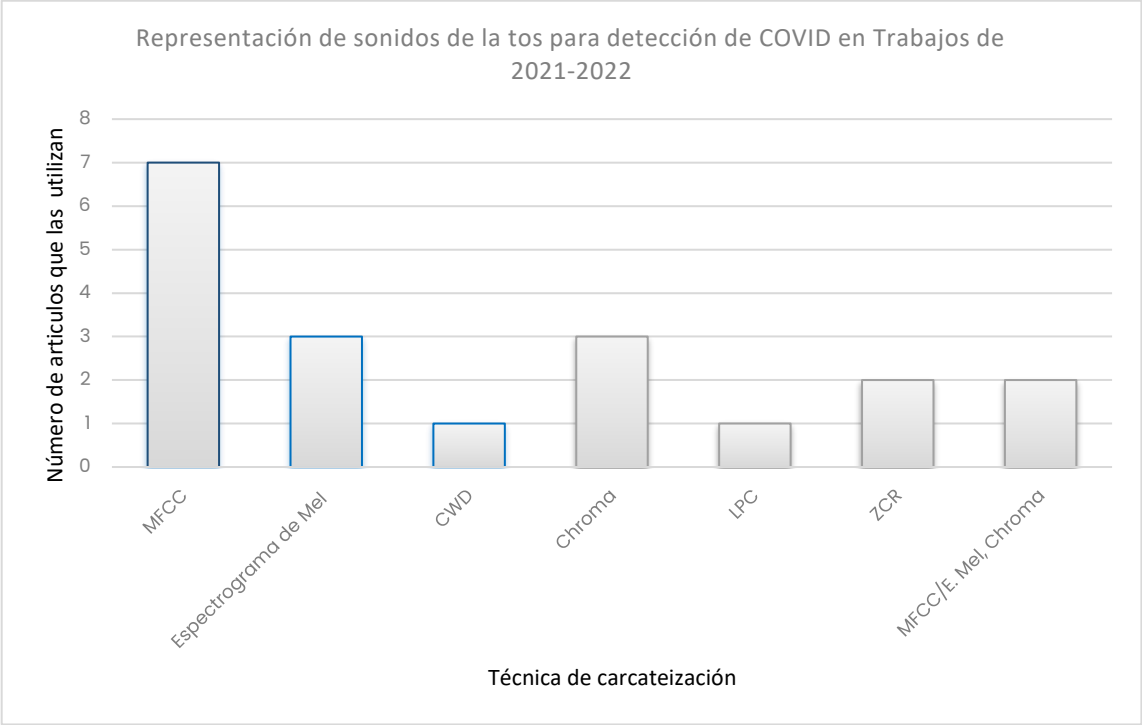


Figura 15 Proporción de representación de señal acústica de tos

En este trabajo se extraen 33 coeficientes por técnica de caracterización de audio (MFCC, Espectrogramas de Mel, Chorma), puesto que en el trabajo de (Andreu-Perez, 2021) se expone que utilizar este número de características obtiene buenos resultados de predicción automática de COVID-19 con sonidos de la tos. Para construir el vector de entrada de la red neuronal se abordan dos tipos de representaciones, bidimensional y tridimensional las cuales se describen a continuación.

4.4.1 Tensor 2D

Abarca dos dimensiones, considerando que se extraen 33 coeficientes por cada técnica de caracterización (MFCC, Chroma, Espectrogramas de Mel) se conforma un tensor de (Tamaño de la ventana × 33 × 1). Tal y como se muestra en la figura 16.

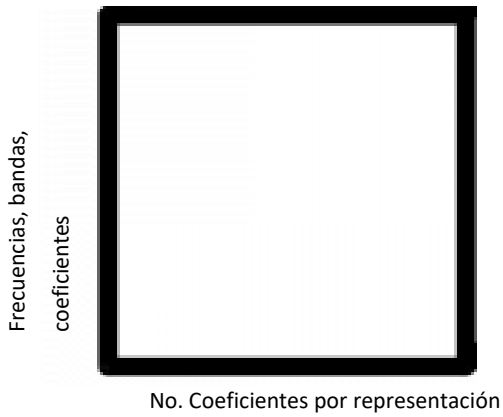


Figura 16 Tensor que abarca dos dimensiones

En donde:

- Tamaño de la ventana= Arreglo de la señal correspondiente al tamaño de la ventana calculada como adecuada.
- 33 = Número de coeficientes que se extraen por cada señal de audio (muestra).
- 1= Representación: MFCC ó Chroma ó Espectrograma de Mel (1 canal).

4.4.2 Tensor 3D

Abarca tres dimensiones, considerando que extraen 33 coeficientes por cada técnica de caracterización (MFCC, Chroma, Espectrogramas de Mel) se conforma un tensor de (Tamaño de la ventana × 33 × 3). Tal y como se muestra en la figura 17.

En donde:

- Tamaño de la ventana= Arreglo de la señal correspondiente al tamaño de la ventana calculada como favorable.
- 33 = Número de coeficientes que se extraen por cada señal de audio (muestra).

- 3= Representación: MFCC, Chroma , Espectrograma de Mel (3 canales).

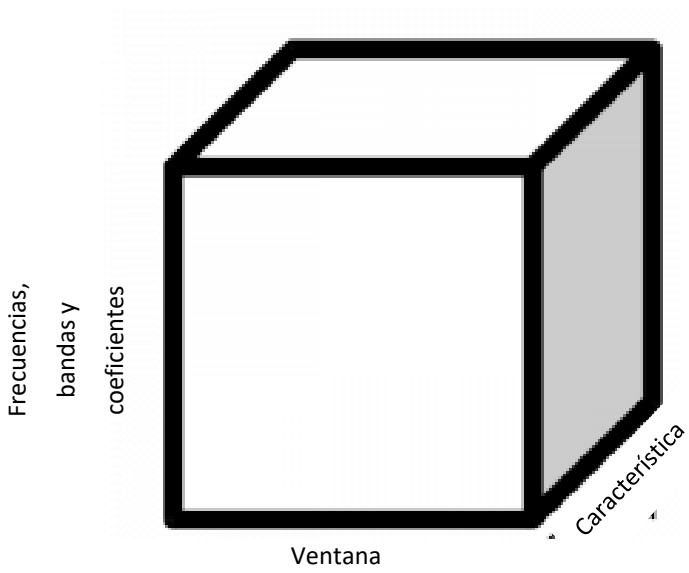


Figura 17 Tensor que abarca tres dimensiones

4.4.3 Representación unidimensional

En el capítulo 2 subtema 2.4.1.1 se describieron otras técnicas de caracterización, las cuales son utilizadas en este trabajo. Emolarge, Is10_paraling y GeMAPSvb01. Esta representación consta de un vector de un conjunto de descriptores de la señal de audio en forma de vector. Esta representación es unidimensional, puesto que cada señal de acústica de un conjunto de audio tiene su equivalente en descriptores acústicos según sea el conjunto que se extraiga. Este proceso de la extracción de características se obtiene por medio de la herramienta de OpenSmile (Eyben F. W., 2010) en el lenguaje de programación Python.

4.5 Técnicas de balanceo de clases

En este subtema se aborda el punto “Transformación” de la metodología integral propuesta en el capítulo 1, subtema 1.6.

Se aplicaron técnicas de balanceo de clases para las clases minoritarias de la base de datos, con la finalidad de disminuir sesgos de clasificación y aplicar técnicas de generación de datos sintéticos.

En la figura 18 se muestra como la base de datos utilizada en este trabajo la cual se describe en el subtema 4.1 presenta desbalanceo de clases, en donde la clase minoritaria es la positiva y la clase mayoritaria es la negativa, es decir que hay más datos de personas negativas a COVID-19. Las técnicas que se utilizaron en este trabajo se definen en el capítulo 2, en el subtema 2.3.

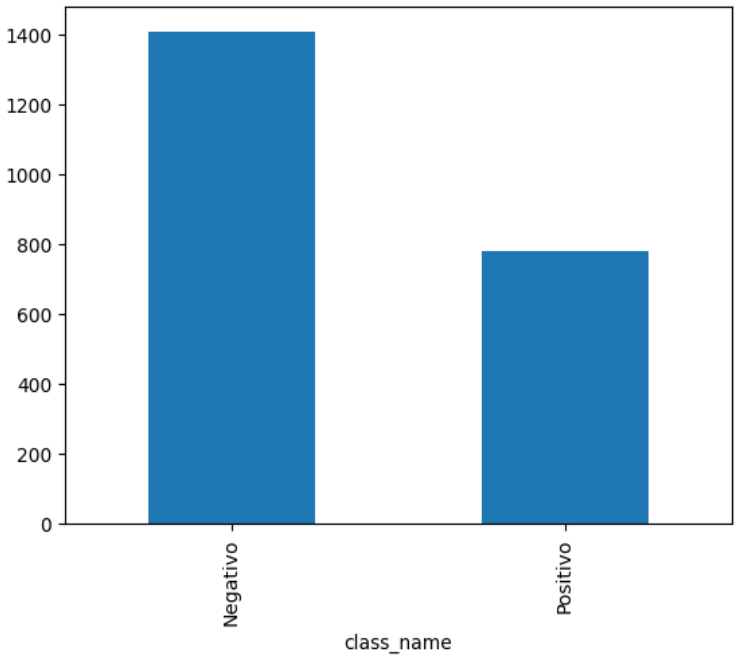


Figura 18 Proporción de instancias por clases de base de datos utilizada en este objeto de estudio (CICESE)

El desbalanceo de clases se pretende abordar con diferentes técnicas de balance de datos, con técnicas convencionales de submuestreo y sobremuestreo que funcionan como base para otros algoritmos de balanceo de clases. Así mismo, se utilizan técnicas de aumento de datos las cuales, consideran los formatos de audio originales de la clase minoritaria para aplicar modificaciones y obtener nuevas muestras.

Por último, se prueba con generación sintética a través de aprendizaje automático, específicamente con autoencoder.

Estas técnicas se utilizaron en este trabajo ya que son las encontradas en la revisión del estado del arte de capítulo 3 subtema 3.1 y 3.2, las cuales corresponden a técnicas de balanceo de clases convencionales, por aumento de datos y por generación sintética. A continuación, se describen detalladamente.

4.5.1 Técnicas convencionales

ROS (Random Oversampling): Es la técnica de sobremuestreo básica en el que su algoritmo se basa en seleccionar de forma aleatoria instancias de la clase minoritaria para igual el número de instancias para la clase mayoritaria. En la figura 19 se muestra la proporción de clases aplicando ROS a la base de datos utilizada.

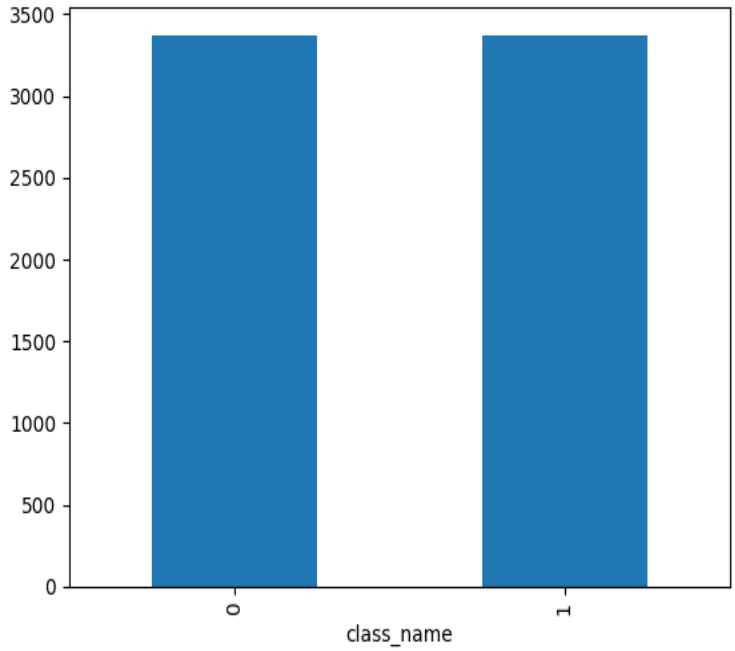


Figura 19 Proporción de instancias aplicando ROS: 0 es clase Negativa y 1 es clase Positiva

RUS (Random undersampling): Es la técnica de submuestreo básica en el que si algoritmo al contrario de ROS elimina instancias aleatoriamente de la clase mayoritaria hasta igualar el número de instancias de la clase minoritaria. En la figura 20 se muestra la proporción de clases aplicando RUS a la base de datos utilizada.

SMOTE (Synthetic Minority Oversampling Technique): Es la técnica de sobre muestreo que genera muestras sintéticas se añaden a la clase minoritaria, o la clase que contiene menor número de muestras.

SMOTE Utiliza el espacio de características para generar nuevas muestras gracias a la interpolación entre las muestras de la clase minoritaria que se encuentra juntas.

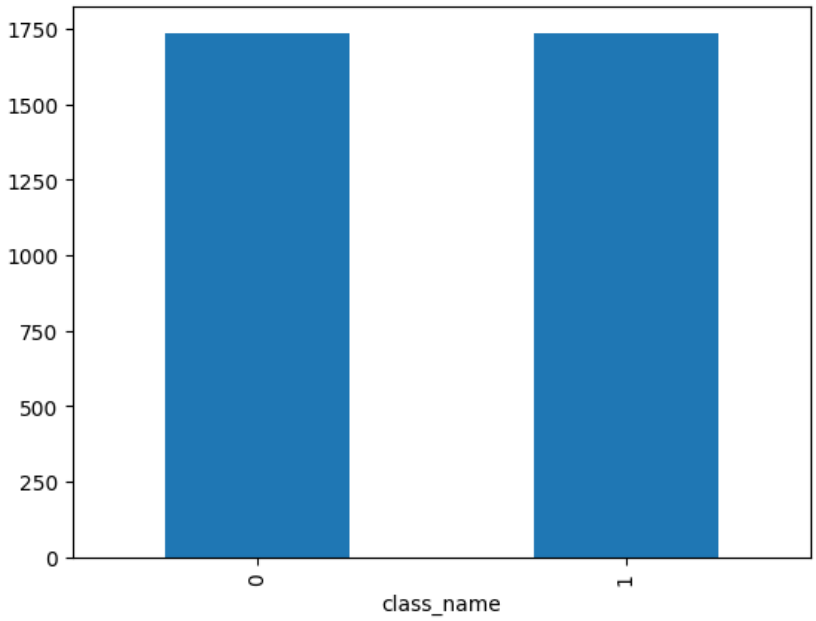


Figura 20 Proporción de instancias aplicando RUS: 0 es clase Negativa y 1 es clase Positiva

El algoritmo base de SMOTE se describe a continuación:

1. Se obtiene el número de instancias totales de las clases predictoras, en este caso objeto de estudio es la clase positiva y negativa a COVID-19.

2. Se selecciona de forma aleatoria una instancia de la clase minoritaria.

3. Se calcula la diferencia de distancia entre el vector de características de la instancia minoritaria seleccionada y sus k vecinos más cercanos (instancias vecinas más cercanas). La diferencia se multiplica por un valor aleatorio entre 0 y 1 (0,1] y se suma el vector de características anterior, como se expresa en la ecuación 7 (Chawla, 2002).

$$\boldsymbol{x' = x + rand(0,1) * | x - k_k |} \quad \textbf{(7)}$$

Donde x' es la nueva instancia sintética generada por SMOTE, x es la instancia minoritaria seleccionada aleatoriamente, rand (0,1) es el número aleatorio entre 0 y 1, el subíndice k representa los k vecinos más cercanos. En la figura 21 se representa gráficamente lo anterio

En la figura 22 se muestra la proporción de clases aplicando SMOTE a la base de datos utilizada.

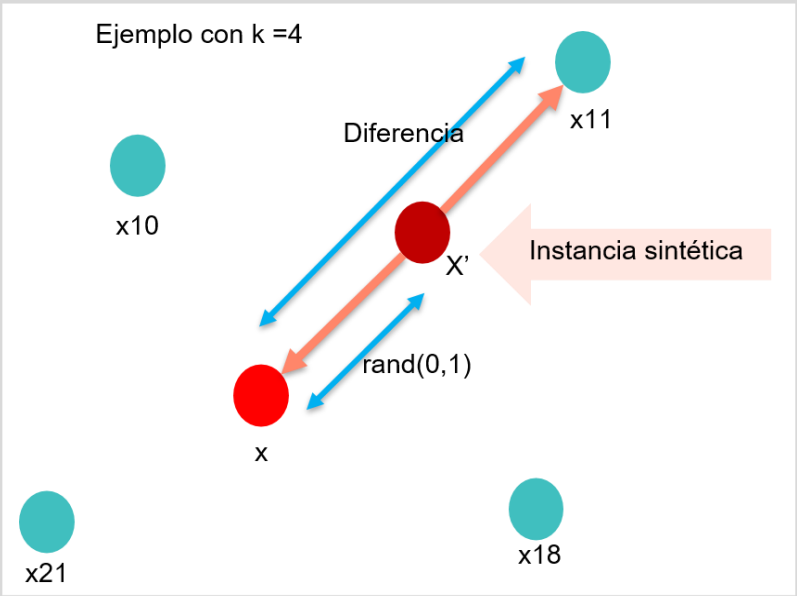


Figura 21 Representación gráfica de SMOTE con k=4

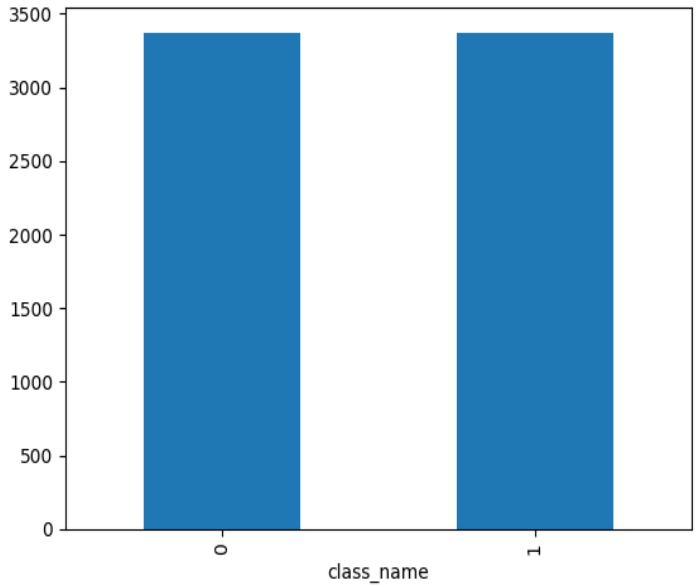


Figura 22 Proporción de instancias aplicando SMOTE: 0 es clase Negativa y 1 es clase Positiva

ADASYN (Adaptative Synthetic Sampling): Esta técnica agrega instancias sintéticas a la clase minoritaria como SMOTE, sin embargo, esta técnica se decide el número de instancias sintéticas generadas para, muestras de difícil aprendizaje, es decir, que se puede cambiar de forma adaptativa la frontera de decisión

para centrarse en los ejemplos difíciles de aprender (Hinojosa Cardenas, 2015). El algoritmo base (He, 2008) se describe a continuación:

1. Se determinan el número de instancias de la clase mayoritaria N^- y la clase minoritaria, N^+. Se establece un umbral para decidir el grado máximo de desequilibrio de clases. El número total de muestras sintéticas a generar se determina por la ecuación 8.

$$\boldsymbol{G = (N^- - N^+) * \left(\frac{N^+}{N^-}\right)} \tag{8}$$

2. Se selecciona de forma aleatoria una instancia minoritaria y se calcula la distancia entre los k vecinos más cercanos y la relación r_i se calcula como Δ_i / k, posteriormente se normaliza como en la ecuación 9.

$$\boldsymbol{r_x <= \frac{r_i}{\sum r_i}} \tag{9}$$

Continuamente las muestras sintéticas totales para cada x_i serán $g_i = r_x * G$, después se itera de 1 a g_i para generar muestras como SMOTE tomando en cuenta la ecuación 6. En la figura 23 se representa gráficamente. Donde x' es la nueva instancia sintética generada por ADASYN, x es la instancia minoritaria seleccionada aleatoriamente, rand (0,1) es el número aleatorio entre 0 y 1. Y el subíndice k representa los k vecinos más cercanos.

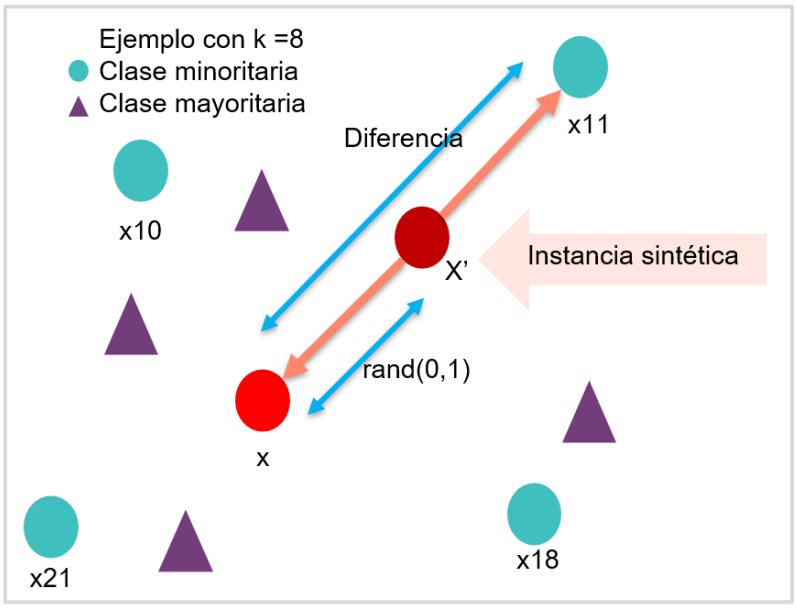


Figura 22 Representación gráfica de ADASYN

En la figura 24 se muestra la proporción de clases aplicando ADASYN a la base de datos utilizada

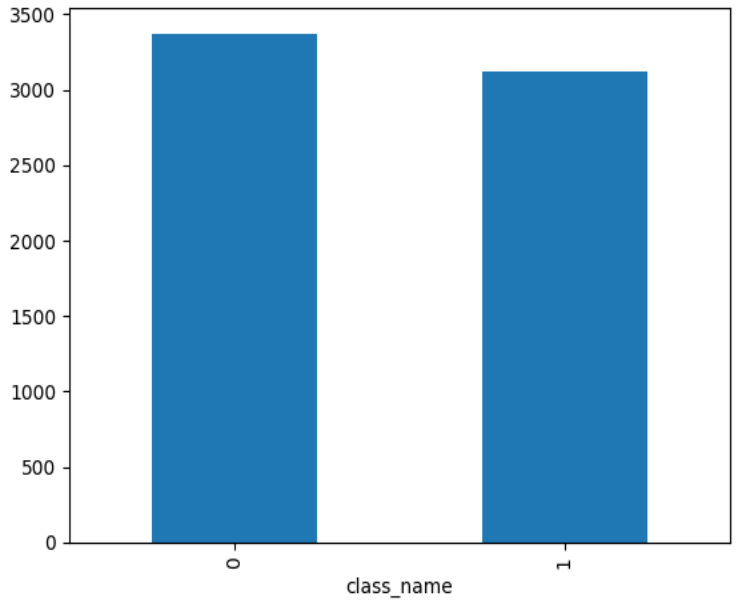


Figura 23 Proporción de instancias aplicando ADASYN: 0 es clase Negativa y 1 es clase Positiva

4.5.2 Aumento de datos

Noise: En esta técnica se agrega ruido blanco a muestras de audio de la clase minoritaria para generar muestras de audio nuevas e igualar el número de instancias de la clase minoritaria. Se agrego un factor de ruido de 0.5. El algoritmo para calcular la señal aumentada de una muestra original con ruido blanco es el siguiente:

1. Obtener el vector numérico de la señal de audio con la librería Librosa (McFee, 2015) en Python.

2. Calcular el ruido de la señal, que es igual a la desviación estándar del vector de numérico de la señal original.

3. Calcular la señal aumentada con ecuación 10.

$$\boldsymbol{noise} = \boldsymbol{signal} + \boldsymbol{noise} * \boldsymbol{factor} \quad (10)$$

Dónde signal es el vector número de la señal original, Noise corresponde al paso dos, y factor es igual a 0.5. En la figura 25 se observa un ejemplo de agregar ruido a una señal de audio de una muestra positiva a COVID-19.

a)

b)

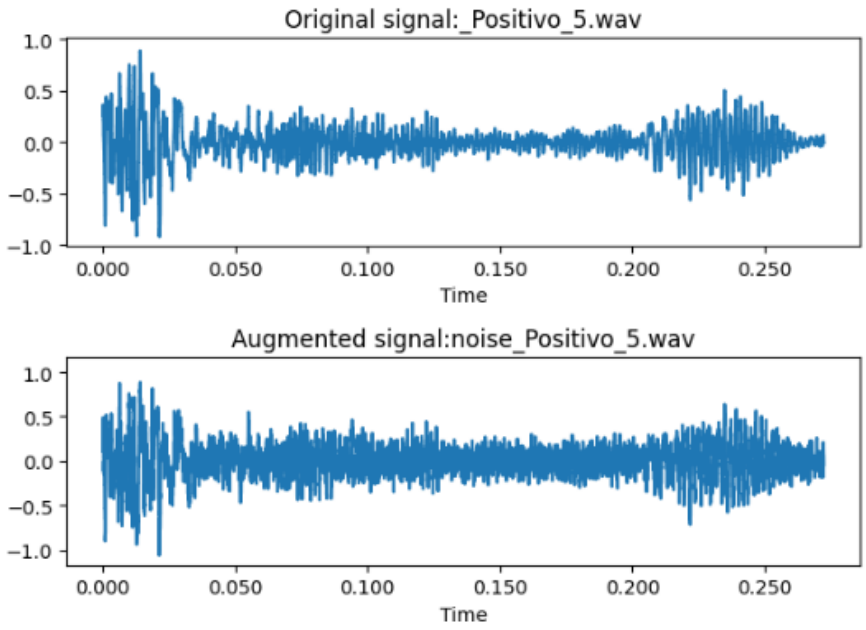


Figura 24 Ejemplo de agregar ruido blanco a una señal de audio de una muestra positiva. a) Señal original. b) Señal con ruido blanco factor 0.5

Pitch scaling: Esta técnica cambia la frecuencia de la señal para generar una nueva muestra para la clase minoritaria, con la finalidad de igualar número de muestras con la mayoritaria. Se agregaron 2 semitonos en la señal original para lograr el cambio de frecuencia en muestras originales. El algoritmo para calcular la señal aumentada de una muestra original con cambio en escala de tono es el siguiente:

1. Obtener el vector numérico y la frecuencia de muestreo de la señal de audio con la librería Librosa (McFee, 2015) en Python.

2. Agregar los semitonos definidos a la señal original tomando en cuenta su frecuencia de muestreo, con la función pitch_shift de la sublibrería effects de Librosa (McFee, 2015).

En la figura 26 se observa un ejemplo de agregar 2 semitonos a una señal de audio de una muestra positiva a COVID-19.

a)

b)

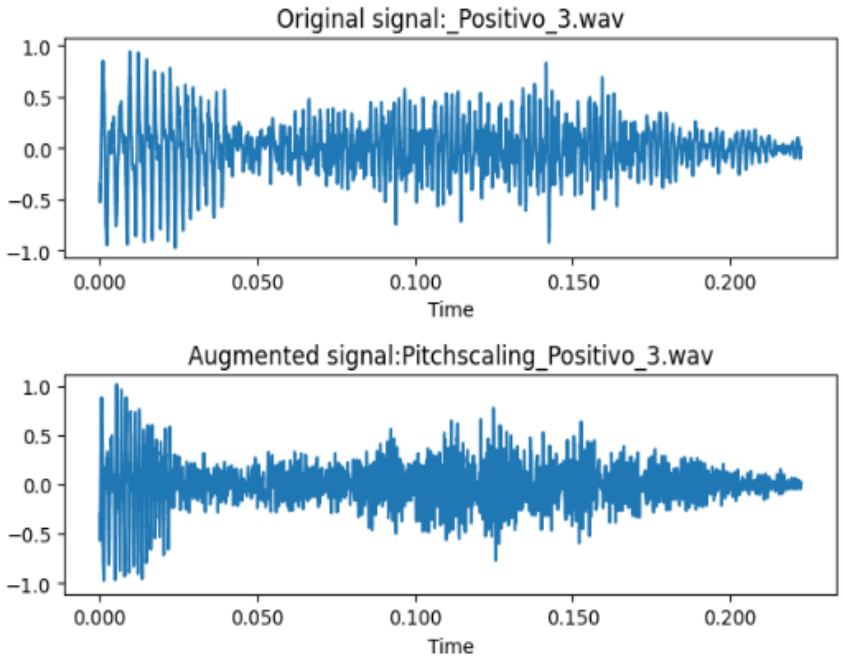


Figura 25 Ejemplo de agregar 2 semitonos a una señal de audio de una muestra positiva. a) Señal original. b) Señal con cambio de escala de tono

Time strech: El estiramiento de tiempo cambia la velocidad del sonido, al contrario que Pitch scaling esta técnica no cambia el tono de la señal original. Se estira el tiempo de la señal de audio al 20% es decir con un factor de estiramiento de 0.2. Para generan muestras para la clase minoritaria, con la finalidad de igualar número de muestras con la mayoritaria. El algoritmo para calcular la señal aumentada de una muestra original con estiramiento de tiempo es el siguiente:

1. Obtener el vector numérico de la señal de audio con la librería Librosa (McFee, 2015) en Python.

2. Agregar el estiramiento de tiempo con factor 0.2, con la función Time_strech de la sublibrería effects de Librosa (McFee, 2015).

En la figura 27 se observa un ejemplo de agregar estiramiento de tiempo en un 20% de una señal de audio de una muestra positiva a COVID-19.

a)

b)

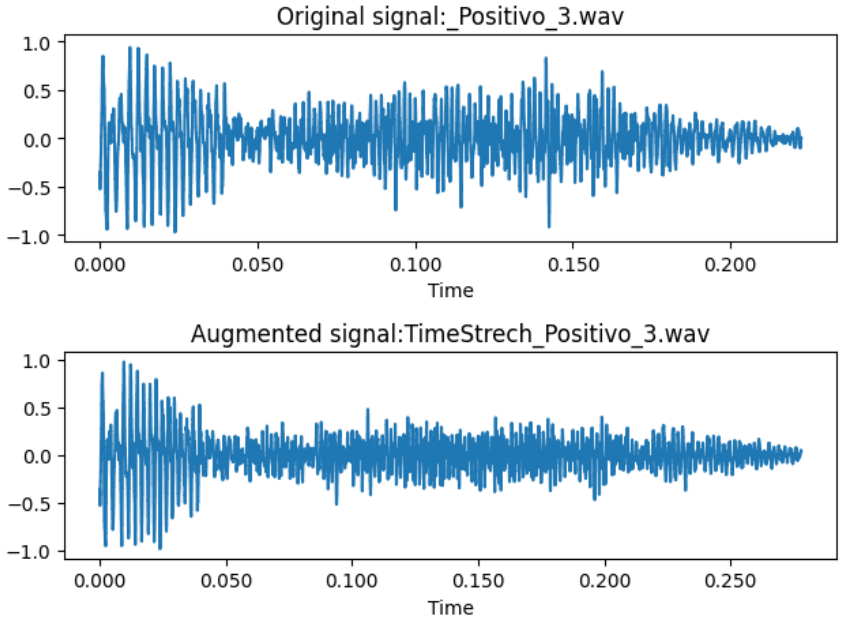


Figura 26 Ejemplo de agregar un estiramiento de tiempo con factor 0.2 a la señal de audio de una muestra positiva. a) Señal original. b) Señal con estiramiento de tono

Random gain: La ganancia aleatoria toma la forma de onda de la señal y la multiplica con un factor aleatorio entre un rango establecido, en este caso se utilizó de 0 a 1, esto cambia la amplitud de la señal, por ende, el volumen que las personas perciben en el audio generado. Se generan muestras para la clase minoritaria, con la finalidad de igualar número de muestras con la mayoritaria. El algoritmo para calcular la señal aumentada de una muestra original con ganancia aleatoria es el siguiente:

1. Obtener el vector numérico de la señal de audio con la librería Librosa (McFee, 2015) en Python.

2. Multiplicar el factor aleatorio entre (0 y 1) por la señal original.

En la figura 28 se observa un ejemplo de multiplicar un factor aleatorio entre (0 y 1) a una señal de audio de una muestra positiva a COVID-19

a)

b)

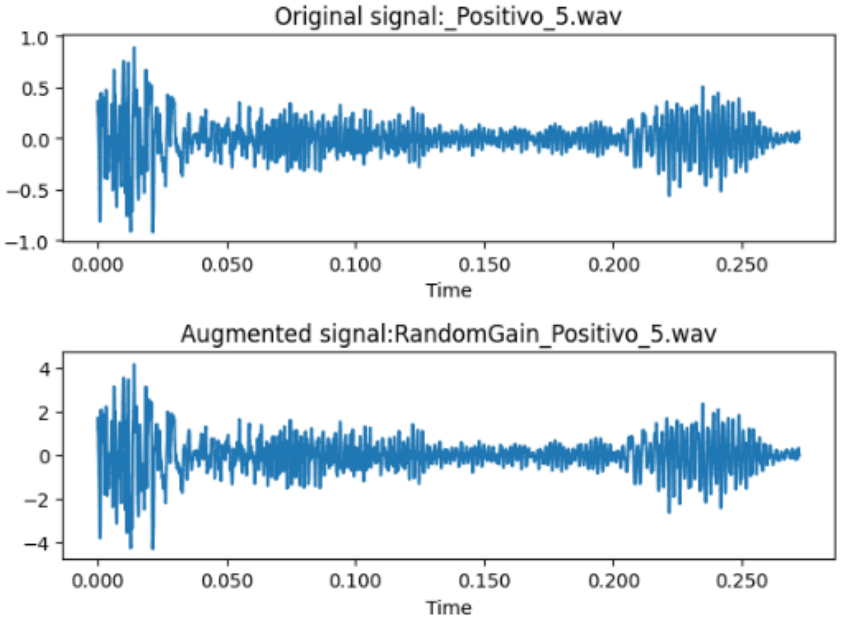


Figura 27 Ejemplo de multiplicar una ganancia aleatoria entre (0,1) por la señal de audio de una muestra positiva. a) Señal original. b) Señal con ganancia aleatoria

Polarity inversion: La inversión de polaridad toma la forma de onda de la señal y la multiplica por -1, esto cambia la polaridad de la señal, por ende, se intercambia el tipo positivo de la señal para su igual en negativo y al revés. Se generan muestras para la clase minoritaria, con la finalidad de igualar número de muestras con la mayoritaria. El algoritmo para calcular la señal aumentada de una muestra original con ganancia aleatoria es el siguiente:

1. Obtener el vector numérico de la señal de audio con la librería Librosa (McFee, 2015) en Python.

2. Multiplicar -1 por la señal original.

En la figura 29 se observa un ejemplo de multiplicar por -1 una señal de audio de una muestra positiva a COVID-19.

a)

b)

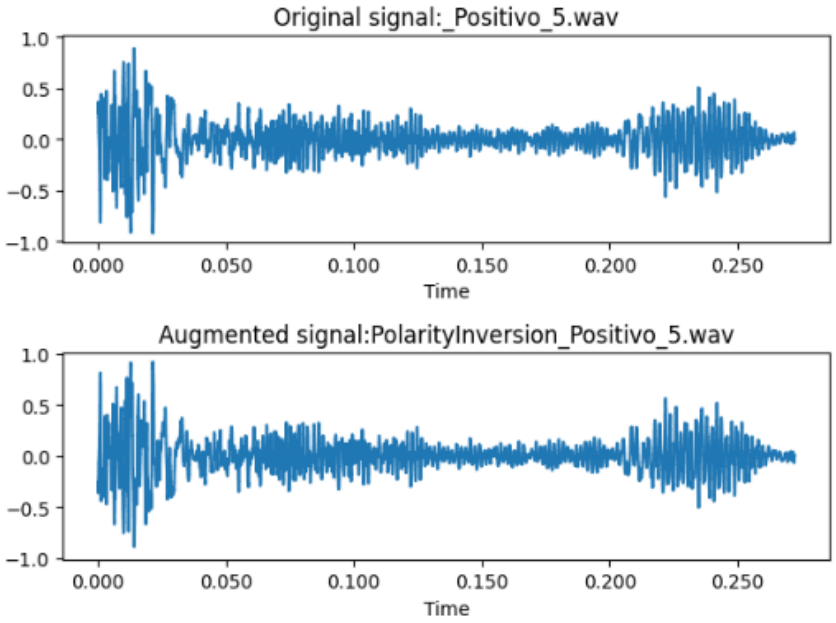


Figura 28 Ejemplo de invertir la polaridad de la señal de audio de una muestra positiva. a) Señal original. b) Señal con polaridad invertida

4.5.3 Generación sintética

VAE: El autocodificador variacional se utilizó para generar muestras sintéticas para la clase minoritaria a través de aprendizaje automático, específicamente se utilizaron dos arquitecturas de red, un codificador y un decodificador. En la figura 30 se presenta el VAE general utilizado en este trabajo. En la figura 31 se muestra la arquitectura de codificador, y en la figura 32 el decodificador. Esta arquitectura de VAE se tomó de (Velardo, 2021), puesto que presenta un curso completo para la generación sintética de audio con VAE, el cual se analizó durante el desarrollo de este trabajo de investigación.

```
_________________________________________________________________
Model: "autoencoder"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 encoder_input (InputLayer)  [(None, 28, 28, 1)]       0

 encoder (Functional)        (None, 2)                 106116

 decoder (Functional)        (None, 28, 28, 1)         121537

=================================================================
Total params: 227,653
Trainable params: 226,821
Non-trainable params: 832
```

Figura 29 Arquitectura general del VAE utilizado para generar muestras sintéticas de clase minoritaria

```
Model: "encoder"
_________________________________________________________________________________________________
 Layer (type)                Output Shape              Param #   Connected to
=================================================================================================
 encoder_input (InputLayer)  [(None, 28, 28, 1)]       0         []

 encoder_conv_layer_1 (Conv2D)  (None, 28, 28, 32)     320       ['encoder_input[0][0]']

 encoder_relu_1 (ReLU)       (None, 28, 28, 32)        0         ['encoder_conv_layer_1[0][0]']

 encoder_bn_1 (BatchNormalizati  (None, 28, 28, 32)    128       ['encoder_relu_1[0][0]']
 on)

 encoder_conv_layer_2 (Conv2D)  (None, 14, 14, 64)     18496     ['encoder_bn_1[0][0]']

 encoder_relu_2 (ReLU)       (None, 14, 14, 64)        0         ['encoder_conv_layer_2[0][0]']

 encoder_bn_2 (BatchNormalizati  (None, 14, 14, 64)    256       ['encoder_relu_2[0][0]']
 on)

 encoder_conv_layer_3 (Conv2D)  (None, 7, 7, 64)       36928     ['encoder_bn_2[0][0]']

 encoder_relu_3 (ReLU)       (None, 7, 7, 64)          0         ['encoder_conv_layer_3[0][0]']

 encoder_bn_3 (BatchNormalizati  (None, 7, 7, 64)      256       ['encoder_relu_3[0][0]']
 on)

 encoder_conv_layer_4 (Conv2D)  (None, 7, 7, 64)       36928     ['encoder_bn_3[0][0]']

 encoder_relu_4 (ReLU)       (None, 7, 7, 64)          0         ['encoder_conv_layer_4[0][0]']

 encoder_bn_4 (BatchNormalizati  (None, 7, 7, 64)      256       ['encoder_relu_4[0][0]']
 on)

 flatten (Flatten)           (None, 3136)              0         ['encoder_bn_4[0][0]']

 mu (Dense)                  (None, 2)                 6274      ['flatten[0][0]']

 log_variance (Dense)        (None, 2)                 6274      ['flatten[0][0]']

 encoder_output (Lambda)     (None, 2)                 0         ['mu[0][0]',
                                                                  'log_variance[0][0]']

=================================================================================================
Total params: 106,116
Trainable params: 105,668
Non-trainable params: 448
_________________________________________________________________________________________________
```

Figura 30 Arquitectura de red de codificador de la señal de audio

```
_________________________________________________________________
Model: "decoder"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 decoder_input (InputLayer)  [(None, 2)]               0

 decoder_dense (Dense)       (None, 3136)              9408

 reshape (Reshape)           (None, 7, 7, 64)          0

 decoder_conv_transpose_laye  (None, 7, 7, 64)         36928
 r_1 (Conv2DTranspose)

 decoder_relu_1 (ReLU)       (None, 7, 7, 64)          0

 decoder_bn_1 (BatchNormaliz  (None, 7, 7, 64)         256
 ation)

 decoder_conv_transpose_laye  (None, 14, 14, 64)       36928
 r_2 (Conv2DTranspose)

 decoder_relu_2 (ReLU)       (None, 14, 14, 64)        0

 decoder_bn_2 (BatchNormaliz  (None, 14, 14, 64)       256
 ation)

 decoder_conv_transpose_laye  (None, 28, 28, 64)       36928
 r_3 (Conv2DTranspose)

 decoder_relu_3 (ReLU)       (None, 28, 28, 64)        0

 decoder_bn_3 (BatchNormaliz  (None, 28, 28, 64)       256
 ation)

 decoder_conv_transpose_laye  (None, 28, 28, 1)        577
 r_4 (Conv2DTranspose)

 sigmoid_layer (Activation)  (None, 28, 28, 1)         0

=================================================================
Total params: 121,537
Trainable params: 121,153
Non-trainable params: 384
```

Figura 31 Arquitectura de red de decodificador de la señal de audio

En este trabajo de tesis como ya se analizó anteriormente la clase mayoritaria es la clase negativa y la minoritaria es la positiva, para abordar el problema de desequilibrio de clases a través de esta técnica de generación sintética de audio por VAE se utilizaron dos enfoques, el primero, aumentar la clase positiva

con datos sintéticos para posteriormente entrenar con CNN y el segundo enfoque, aumentar clase positiva y clase negativa. A continuación, se describe las implicaciones de cada enfoque.

Aumentar clase positiva

- Datos de entrenamiento de VAE: Este conjunto de datos se conformó el 70% las muestras de audio positivas de la BD CICESE las cuales pertenecen a la clase minoritaria, segmentadas por la técnica de comparador de histéresis digital.

  Estos datos se caracterizaron con espectrogramas de Mel, los cuales funcionaron como datos de entrada del codificador de la arquitectura de VAE obteniendo como salida del decodificador espectrogramas sintéticos aprendidos por el modelo variacional.

- Tiempo de entrenamiento del VAE: El tiempo que se tardó entrenando el modelo con los espectrogramas de la clase positiva fue de 6 horas.

- Generación sintética: Se consideraron el 70 % de muestras del conjunto de datos CICESE, para determinar el número de muestras faltantes de la clase minoritaria para igualar clase mayoritaria. Sabiendo el número de muestras faltantes se seleccionaron aleatoriamente datos de prueba de los espectrogramas originales para obtener a través del modelo entrenado el número exacto de espectrogramas sintéticos.

- Datos de entrenamiento de CNN con tensor 3D: De los espectrogramas generados sintéticamente por VAE se convirtieron en audio a través del uso de la librería de Librosa. Posteriormente se conformó una nueva BD la cual contiene audios sintéticos y originales de la clase positiva, además del 70% de los audios originales de clase negativa segmentos con la técnica de comparador de histéresis digital.

- Clasificación con CNN con tensor 3D: Una vez que se obtuvo el conjunto de entrenamiento conformado por datos sintéticos y originales para clase positiva, y el 70% de los audios negativos, se caracterizó la señal de audio para realizar la clasificación automática con una CNN con tensor 3D. Esta arquitectura y clasificación (CNN con tensor 3D) se detalla en el siguiente subtema de este capítulo.

Aumentar clase positiva y negativa

- Datos de entrenamiento de VAE: Este conjunto de datos se conformó el 40% las muestras de audio negativas de la BD CICESE las cuales pertenecen a la clase mayoritaria, segmentadas por la técnica de comparador de histéresis digital.

  Estos datos se caracterizaron con espectrogramas de Mel, los cuales funcionaron como datos de entrada del codificador de la arquitectura de VAE obteniendo como salida del decodificador espectrogramas sintéticos aprendidos por el modelo variacional.

- Tiempo de entrenamiento del VAE: El tiempo que se tardó entrenando el modelo con los espectrogramas de la clase negativa fue de 8 horas.

- Generación sintética: La finalidad de generar audio sintético es igualar el número de muestras para la clase positiva y negativa, sin embargo, en este enfoque sólo se tomó el 30% de los datos negativos puesto que se desea generar el 30% más de los datos para considerar el audio sintético que pertenece a clase positiva generado con el enfoque anterior y así obtener el mismo número de muestras en ambas clases. Sabiendo el número de muestras faltantes se seleccionaron aleatoriamente datos de prueba de los espectrogramas originales para obtener a través del modelo entrenado el número exacto de espectrogramas sintéticos.

- Datos de entrenamiento de CNN con tensor 3D: De los espectrogramas generados sintéticamente por VAE se convirtieron en audio a través del uso de la librería de librosa. Posteriormente se conformó una nueva BD la cual contiene audios sintéticos y originales de la clase positiva, además del 40% de los audios originales de clase negativa y el 30% de datos sintéticos negativos generados con este enfoque.

- Clasificación con CNN con tensor 3D: Una vez que se obtuvo el conjunto de entrenamiento conformado por datos sintéticos y originales para clase positiva y negativa, se caracterizó la señal de audio para realizar la clasificación automática con una CNN con tensor 3D.

  Esta arquitectura y clasificación (CNN con tensor 3D) se detalla en el siguiente subtema de este capítulo.

4.6 Clasificación automática

La clasificación automática es la tarea de determinar si una muestra de tos es positiva o negativa a COVID-19 a través del entrenamiento de algoritmos de aprendizaje automático con datos, para este objeto de estudio a partir del preprocesamiento y caracterización de la señal de audio de muestras de toses de pacientes sanos y enfermos de COVID-19.

En este subtema se aborda el punto “Clasificación” de la metodología integral propuesta en el capítulo 1, subtema 1.6.

4.6.1 CNN

Se utilizó una red convolucional profunda para la clasificación de muestras de tos positivas y negativas a COVID-19. Una CNN es el algoritmo de aprendizaje automático más utilizado para analizar espectrogramas por su capacidad de aprender aspectos espaciales como lo son las imágenes. Como se describe en el capítulo 2 subtema 2.5, se puede aprovechar tener diversos canales, es por esto que se utiliza esta característica en este trabajo con dos variantes, una de entrada bidimensionales y otra para entrada tridimensionales (2D y 3D) como se describen en el subtema 4.4 de este capítulo.

a) CNN con dos canales

   La arquitectura para la red convolucional profunda para entrada bidimensional se observa en la tabla 6. Esta arquitectura se construyó con la librería disponible en Python de Keras y TensorFlow (Géron, 2022).

a) CNN con tres canales

   La arquitectura para la red convolucional profunda para entrada tridmensional se observa en la tabla 7. Esta arquitectura se construyó con la librería disponible en Python de Keras y TensorFlow (Géron, 2022).

Tabla 6 Arquitectura CNN con tensor 2D

Block Number	Layer Name	Output Shape	Learning Params
Block 1	Convolution + ReLu MaxPooling Dropout (0.2)	(32, 16) (171,16) (171,16)	208 n/n n/n
Block 2	Convolution + ReLu MaxPooling Dropout (0.2)	(170, 32) (85,32) (85,32)	16448 0 0
Block 3	Convolution + ReLu MaxPooling Dropout (0.2)	(84,64) (42, 64) (42, 64)	16448 0 0
Block 4	Convolution + ReLu MaxPooling Dropout (0.2)	(41,128) (20,128) (20,128)	16448 0 0
GA	Global Average	(3648)	0
Out	Dense + Softmax	(2)	1002

Tabla 7 Arquitectura CNN con tensor 3D

Block Number	Layer Name	Output Shape	Learning Params
Block 1	Convolution + ReLu MaxPooling Dropout (0.2)	(32, 343,16) (16, 171,16) (16,171,16)	208 n/n n/n
Block 2	Convolution + ReLu MaxPooling Dropout (0.2)	(15, 170, 32) (7,85,32) (7,85,32)	16448 0 0
Block 3	Convolution + ReLu MaxPooling Dropout (0.2)	(6,84,64) (3,42, 64) (3,42, 64)	16448 0 0
Block 4	Convolution + ReLu MaxPooling Dropout (0.2)	(2,41,128) (1,20,128) (1,20,128)	16448 0 0
GA	Global Average	(3648)	0
Out	Dense + Softmax	(2)	1002

Cada bloque convolucional se compone de las siguientes capas:

- Capas convolucionales con unidades lineales rectificadoras (ReLU): En esta capa se ReLU retienen los valores positivos de la entrada y reduce los valores negativos a cero (Nair, 2010). Las dimensiones que se establecen como entrada corresponde a la fila, columna y canales (ya sea bidimensional o tridimensional).

- Capa de agrupación máxima (Max pooling) (Christlein, 2019): Enseña a la red convolucional identificar patrones similares, aunque existan diferencias, pero no significativa.

- Capa de abandono (Dropout) (Srivastava, 2014): Establecido en un 0.2, el cual significa que el 20% del aprendizaje de las neuronas de la capa anterior será olvidado, esto evita el sobre ajuste en el entrenamiento.

Las capas seguidas aplican transformaciones al resultado de los 4 bloques anteriores de capas CNN para la capa final:

- Capa de agrupación promedio global (GA – Global Average) (Akhtar, 2020). En esta capa de agrupación se reemplazan las capas conectadas por la CNN. Generando un mapa de características para cada clase proveniente de la última capa de la convolución, para calcular el proemdio de cada capa de características y el vector que resulta de esta operación es la entrada de la siguiente capa que es la densa + softmax.

- Capa densa: esta capa tiene como resultado la salida, que son las clases con las que se entrenó el modelo (una salida por clase). Más softmax (Zúñiga-López, 2020): Es función que se encarga de hacer la elección o clasificación de las entradas.

4.6.1.1 Auto ML

Auto machine learning se utiliza para la optimización de parámetros en un modelo de aprendizaje automático. En este trabajo se utilizó la herramienta HyperOpt para la búsqueda de hiperparametros que funcionarán mejor con la CNN. HyperOpt es un paquete de Python que utiliza el algoritmo Tree-based Parzen Esimtors (TPE) para seleccionar hiperparámetros de modelo que optimizan una función objetivo

definida por el usuario. Simplemente definiendo la forma funcional y los límites de cada hiperparámetro, TPE busca de manera exhaustiva pero eficiente a través del hiperespacio complejo para alcanzar valores óptimos (Bergstra, 2013). TPE es un algoritmo secuencial que aprovecha la actualización bayesiana y sigue la siguiente secuencia. Se utilizó la función hyperopt.fmin(): Función objetiva para ejecutar Hyperopt. Sigue el siguiente algoritmo:

1. Entrenar un modelo con un conjunto hiperparámetros.
2. Dividir los valores de nuestra función objetivo observados en grupos "buenos" y "malos".
3. Calcular P(x|bueno) / P(x|malo).
4. Determinar los hiperparámetros que maximizan.
5. Ajuste de modelo usando los hiperparámetros del paso.
6. Repetir los pasos 2 a 5 hasta un criterio de parada.

4.6.2 Random forest

En el algoritmo de Random forest se describe en el capítulo 2 en el subtema 2.5. En te trabajo se utiliza a través de la herramienta de Weka (Hall, 2009). El cual es un software que permite la experimentación con algoritmos de aprendizaje automático (Corso, 2009).

Capítulo 5. Resultados

En este capítulo se exponen los resultados obtenidos en el preprocesamiento, representación, balanceo de clases y clasificación automática de señales acústicas de la tos, el cual en este caso se enfoca en la detección de COVID-19.

En este subtema se aborda el punto “Medición” de la metodología integral propuesta en el capítulo 1, subtema 1.6.

5.1 Preprocesamiento: segmentación

La segmentación es la técnica de dividir y seleccionar la información específica de una señal de audio. Como ya se analizó, realizar la segmnetación permite entrenar a los algoritmos con información útil. En este caso con segmnetos de toses únicamente, dejando de lado segmentos que no contienen información para predecir automáticamente el COVID-19. Los segmentos sin tos, suelen contener ruido, esto propicia sesgos en la predicción. En este trabajo se aplicaron 3 técnicas de segmentación de la señal de audio para la tos con la finalidad de identificar información precisa, como es la tos. Estas técnicas de describieron en capítulo 4 subtema 4.2.

La evaluación y comparativa de las técnicas de segmentación aplicadas a base de datos CICESE-UT3 se presentan a continuación.

Es importante observar la duración de los segmentos resultantes de cada técnica de segmentación, ya que es deseable que la técnica no genere fragmentos de audio cortos o demasiado largos, para que la duración sea consistente y esto se debe reflejar en una duración de estándar pequeña.

En la tabla 8, se presentan las estadísticas de duración de los segmentos de las muestras de audio obtenidas con la segmentación por cambio de intensidad e intervalos de detección de silencio. Además, se muestran las estadísticas de duración utilizando segmentación por descomposición de modo empírico (EMD) y la segmentación con comparador de histéresis digital.

Tabla 8 Estadísticas de duración de segmentos por técnica cambio de intensidad e intervalos de detección de silencio, EMD y comparador de histéresis digital

Segmentación: cambio de intensidad e intervalos de detección de silencio						
Número, muestras	14,023 muestras	**Tiempo de segmento(s)**	**Mínimo(s)**	**Máximo(s)**	**Promedio(s)**	**Desviación estándar(s)**
Positivos	Positivos 3,751	**Total promedio(s)**	0.14	1.23	0.45	0.32
Negativos	Negativos 10,272	**Total desviación estándar(s)**	0.05	0.64	0.17	0.17
Segmentación: EMD						
Número, muestras	6166 Muestras	**Tiempo de segmento(s)**	**Mínimo (s)**	**Máximo(s)**	**Promedio(s)**	**Desviación estándar(s)**
Positivos	Positivos 1.976	**Total promedio (s)**	0.30	0.81	0.51	0.41
Negativos	Negativos 4,190	**Total desviación estándar(s)**	0.08	0.32	0.15	0.56
Segmentación: Comparador de histéresis digital						
Número, muestras	14,596 muestras	**Tiempo de segmento(s)**	**Mínimo(s)**	**Máximo(s)**	**Promedio(s)**	**Desviación estándar(s)**
Positivos	Positivos 5,005	**Total promedio (s)**	0.22	0.48	0.30	**0.12**
Negativos	Negativos 9,591	**Total desviación estándar(s)**	0.04	0.20	0.08	**0.15**

De acuerdo a las estadísticas anteriores de la tabla 8 se observa como la técnica de segmentación del comparador de histéresis digital hace que los segmentos obtenidos de las toses presenten menor desviación estándar (destacado en azul), lo cual significa que el tamaño de los segmentos en duración no estés tan alejados de la media, es decir que no hay tanta diferencia en duración de los segmentos.

Evaluar las técnicas de segmentación es importante para determinar el rendimiento de la técnica, la evaluación que se aplicó en este trabajo se describe en capítulo 4 subtema 4.2.1. Los resultados de la evaluación de las tres técnicas de segmentación se presentan en la tabla 9.

Tabla 9 Evaluación de técnicas de segmentación utilizada en base de datos CICESE

Métrica	Cambio de intensidad e intervalos de detección de silencio	Comparador de histéresis digital	EMD
Exactitud	85.64%	88.64%	86.64%
Precisión	87.6%	93.35%	88.66%
Medida F1	84.95%	89.0%	86.84%
Tiempo de ejecución	8 horas	4 horas	6.9 horas

De acuerdo a la tabla 9 se observa que la segmentación de los audios de la base de datos CICESE, muestra mejores resultados al utilizar la técnica por comparador de histéresis digital en Medida F1 con 89.0% respecto a las técnicas por EMD y por cambio de intensidad e intervalos de detección de silencio. La técnica de segmentación que presenta una Medida F1 más baja con 84.95% es por EMD.

La evaluación de la segmentación manual y automática se presentan en la tabla 10. La segmentación para ambas técnicas se hizo con una muestra de 30 audios para verificar la diferencia en sensibilidad y especificidad.

Tabla 10 Comparación de evaluación de segmentación manual y automática con comparador de histéresis digital

Segmentación manual		
Métrica	**Resultado**	**Tiempo de duración al realizar segmentación**
Sensibilidad	100%	12 horas
Especificada	100%	
Segmentación automática: comparador de histéresis digital		
Sensibilidad	84.38%	1 hora
Especificidad	93.35%	

En la tabla 10 se observa que los mejores resultados de sensibilidad y especificidad se obtienen segmentado manualmente las toses, sin embargo, la duración para segmentar 30 audios es de 12 horas, en comparación de utilizar la segmentación automática, el tiempo es menor de 1 hora. Considerando que el total de muestras es de 1105 se tardaría alrededor de 442 horas para realizar la segmentación de todas las muestras de toses, esto resultaría en un proceso deficiente. Por esto, se optó por utilizar la segmentación automática. Además, que, en ambientes reales, si esta metodología se pretendiera utilizar en dispositivos móviles o sistemas de cómputo actuales, la segmentación manual no podría llevarse a cabo dentro del proceso ya que la idea principal seria automatizar procesos, y hacer segmentación forma manual no resultaría en un proceso eficiente ni automático.

5.1.1 Marco de tiempo

Para obtener el tamaño de la ventana adecuada de los audios de los segmentos obtenidos por la técnica de comparador de histéresis digital se llevó a cabo el proceso descrito en el subtema 4.3 del capítulo 4. La distribución de los datos segmentados con la técnica de comparador de histéresis digital de la base de datos CICESE se observa en la figura 33.

Los datos no siguen una distribución normal estándar de acuerdo a la prueba K^2 de D'Agostino (D'agostino, 1990) que calcula la curtosis y asimetría a partir de los datos. Lo cual significa para este conjunto de datos que, al realizar la prueba de dos colas con un 5% de significancia, no haya datos en la cola del límite inferior.

Sin embargo, de acuerdo al grafico anterior, la línea punteada verde representa el límite inferior aceptado en segundos de los segmentos, siendo el límite inferior de la cola de la distribución de los datos igual al 2.5% de significancia y la línea punteada roja marca el límite superior en segundos de la duración de los segmentos aceptados en un 97.5% de significancia. Y la línea roja con nombre PDF teórico (Probability Density Function), son los valores de que debería seguir la función de la distribución normal.

Para calcular el tamaño de la ventana adecuada se considera la fórmula del capítulo 4 subtema 4.3 en dónde para la BD CICESE, la frecuencia de muestreo de las muestras de audio obtenida por la librería Librosa es de 16,000. El tiempo de duración máxima de las muestras es de 2.4 segundos.

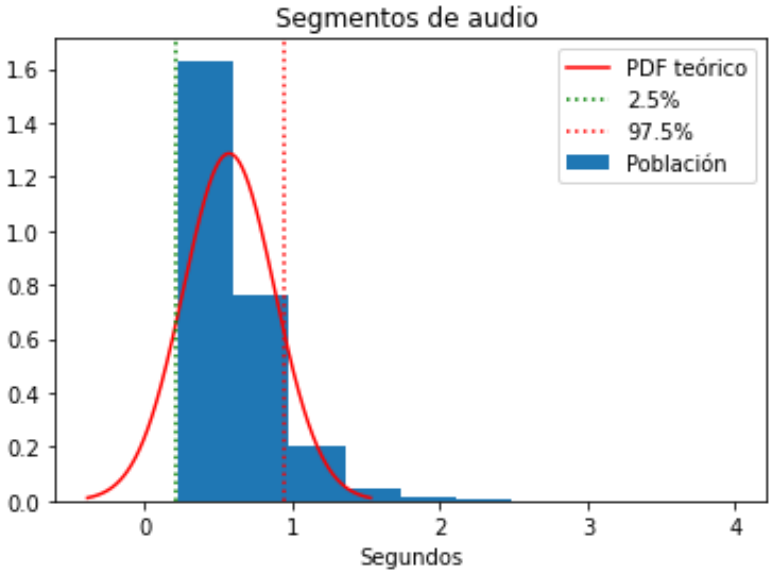


Figura 32 Histograma de duración de los segmentos con comparador de histéresis digital de base de datos CICESE

Considerando lo anterior y el cálculo del tamaño favorable de la ventana es de **344** muestras de la señal, esto equivale a un marco de tiempo de 0.93 segundos para los segmentos por comparador de histéresis digital obtenidos con la base de datos CICESE. Los segmentos que presentan una duración máxima de 0.93 segundos estarán cortadas. Las muestras que duran menos de 0.93 segundos el marco será rellenado a la derecha de los datos con ceros para completar el marco de 344 muestras de la señal.

5.2 Caracterización y clasificación

La caracterización de señales acústicas extrae determinadas propiedades de la señal, como se definió en capítulo 4 subtema 4.4. A continuación, se presenta la evaluación de la representación con una CNN con tensor 2D y 3D.

5.2.1 CNN con tensor 2D

La arquitectura de CNN con tensor 2D se presenta en la tabla 6. Y el formato de entrada de la CNN se constituye de la siguiente forma: (número de coeficientes extraídos, tamaño del marco de tiempo con representación MFCC o Chroma o Espectrograma de Mel) = (33, 344, 1)

Los resultados se presentan en la tabla 11 utilizando segmentos obtenidos con comparador de histéresis digital de la base de datos de CICESE.

Tabla 11 Resultados de arquitectura CNN con tensor 2D

Caracterización y coeficientes utilizados	Exactitud
33 de Mfcc, 33 Mel, 33 Chroma (99 características)	Entrenamiento: 0.65 Prueba: 0.66
33 Mfccs	Entrenamiento: 0.39 Prueba: 0.43
33 Espectrogramas de Mel	Entrenamiento: 0.39 Exactitud de Prueba: 0.39
33 Chroma	Entrenamiento: 0.67 Prueba: 0.67

De acuerdo a los resultados anteriores de la tabla 11, se observa que utilizar 33 descriptores Chroma para los segmentos de CICESE obtenidos por comparador de histéresis digital tiene mejores resultados con una exactitud de 0.67 respecto a utilizar MFCC y Espectrogramas de Mel. Sin embargo, utilizar las 3 características a la vez en un formato bidimensional, obtiene una exactitud de prueba de 0.65. Sigue siendo mejores resultados al sólo utilizar la caracterización Chroma.

5.2.2 CNN con tensor 3D

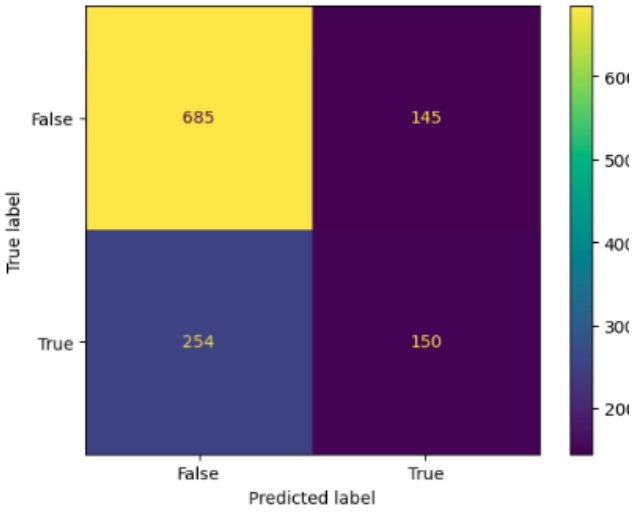
La arquitectura de CNN con 3 tensores se presenta en la tabla 7. Y el formato de entrada de la CNN se constituye de la siguiente forma: (número de coeficientes extraídos, tamaño de la ventana, representación MFCC y Chroma y Espectrograma de Mel) = (33, 344, 3). Los resultados se presentan en la tabla 12 utilizando segmentos obtenidos con comparador de histéresis digital de la base de datos de CICESE.

En la figura 34 se muestran las matrices de confusión de cada experimento, adjuntada de los resultados de sensibilidad y especificidad.

Tabla 12 Resultados de arquitectura CNN tensor 3D

Segmentos	Representación y número de coeficientes	Marco de tiempo adecuado	Partición de datos Entrenamiento/ Prueba	Parámetros de entrenamiento	Resultados de entrenamiento y prueba	Métricas de prueba
(CICESE segmentos EMD)	Mfccs, Espectogrfamas de Mel Chroma. 33 coeficientes	401	20 Prueba 80 Entrenamiento	Épocas: 300 Batch_size =128 Optimizer=adam Metricas de entrenamiento = exactitud, Medida F1	Exactitud de Entrenamiento: 0.94444441 Exactitud de Prueba: 0.676669	Porcentaje de certeza: 67.666 Precision: 0.619 Recall: 0.598 Medida F1: 0.678418
(CICESE segmentos comparador histéresis dogital)	Mfccs, Espectogrfamas de Mel Chroma 33 coeficientes	344	20 Prueba 80 Entrenamiento	Épocas: 300 Batch_size =128 Optimizer=adam Metricas de entrenamiento = exactitud, Medida F1	Exactitud de Entrenamiento: 0.9763617515563965 Exactitud de Prueba: 0.7654109597206116	Porcentaje de certeza: 76.541 Precision: 0.742 Recall: 0.710 Medida F1: 0.720
(CICESE audios completos)	Mfccs, Espectogrfamas de Mel Chroma 33 coeficientes	9999	20 Prueba 80 Entrenamiento	Épocas: 750 Batch_size =12 Optimizer=adam Metricas de entrenamiento = exactitud, Medida F1	Exactitud de Entrenamiento: 0.5700483322143555 Exactitud de Prueba: 0.5723214149475098	Porcentaje de certeza: 58.93719806763285 Precision: 0.575 Recall: 0.561 Medida F1: 0.552

a)



Sensibilidad=37.128%

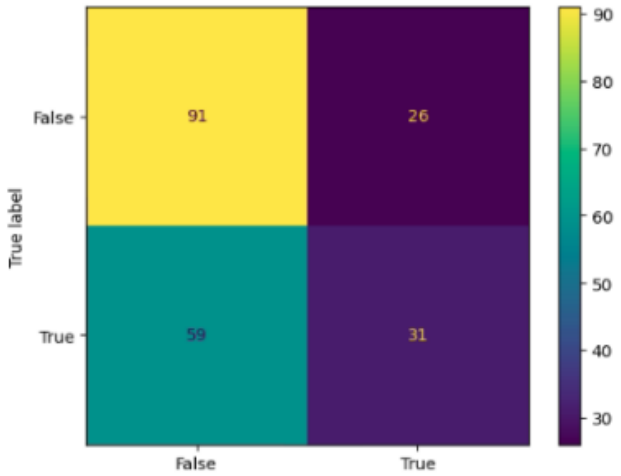
Especificidad=82.53%

b)

Sensibilidad=53.92%

Especificidad= 87.98%

c)



Sensibilidad=34.44%

Especificidad=77.77%

Figura 33 Matriz de confusión y sensibilidad, especificidad de resultados de tabla 14. a) Segmentos con EMD, b) Segmentos con comparador de histéresis digital. C) Segmentos completos

De acuerdo a la tabla 12 y figura 34, segmentar la BD de CICESE con muestras clínicamente validadas por la prueba qRT-PCR y probar en la CNN con tensor 3D es mejor respecto a los resultados de la clasificación con CNN con tensor 2D con la técnica de segmentación por comparador de histéresis digital.

CNN con tensor 3D presenta una sensibilidad de 53.92% y una especificidad de 87.98%. Como ya se analizó antes, esta técnica de segmentación hace que los segmentos de las muestras acústicas de tos no sean tan variables en duración.

La sensibilidad más baja con 34.44% corresponde a utilizar los audios completos, es decir, los audios si utilizar técnicas de segmentación para entrenar CNN con tensor 3D. La especificidad más baja corresponde de igual forma a utilizar audios complementos para entrenar la CNN con tensor 3D, presentado 77.77%.

5.3 Balanceo de clases y clasificación

Las técnicas de balanceo de clases para abordar el problema de desequilibrio de clases son 3 tipos, lo cuales se basan en técnicas convencionales, generación por aumento de datos y generación sintética por VAE. Utilizar técnicas de balanceo de clases se utilizaron con la finalidad de equilibrar las clases, es decir tener el mismo número de instancias para la clase positiva y negativa de la BD CICESE con segmentos obtenidos por comparador de histéresis digital.

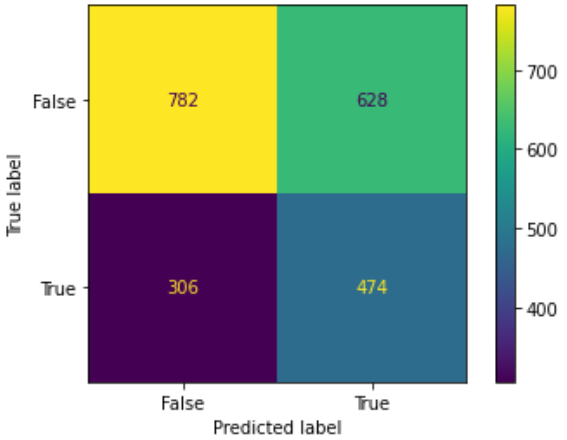
5.3.1 Técnicas convencionales

Las técnicas de balanceo de clases convencionales que se utilizaron en este trabajo fueron: ROS, RUS, SMOTE, ADASYN. Con la finalidad de equilibrar el número de muestras segmentadas por comparador de histéresis digital de las clases del conjunto de datos CICESE. Para la clasificación automática se utilizó la red CNN con 3 canales, con las características de MFCC, Espectrogramas de Mel y Chroma, con un marco de tiempo de 344 muestras de señal calculado en el subtema 5.1.1 de este capítulo. Del total de los datos el 20% fue para prueba y el 80% para entrenamiento en dónde se aplicaron las técnicas de balance. Los resultados de entrenamiento con datos balanceados se presentan en la tabla 13. En la figura 35 se presentan las matrices de confusión de los resultados de conjunto de prueba o test.

Tabla 13 Resultados de entrenamiento de CNN con 3 tensores con datos de BD CICESE segmentados utilizando balance por técnicas convencionales

Técnica de balanceo	Representación y número de coeficientes	Parametrons de entrenamiento	Resultados de entrenamiento y prueba	Métricas de prueba
RUS	Mfccs, Espectogrfamas de Mel Chroma 33 coeficientes	Épocas: 300 Batch_size =128 Optimizer=adam Metricas de entrenamiento = Exactitud, Medida F1	Exactitud de Entrenamiento: 0.93 Exactitud de Prueba: 0.57	Porcentaje de certeza: 57.35 Precision: 0.57 Recall: 0.58 Medida F1: 0.56
ROS	Mfccs, Espectogrfamas de Mel Chroma 33 coeficientes	Épocas: 300 Batch_size =128 Optimizer=adam Metricas de entrenamiento = Exactitud, Medida F1	Exactitud de Entrenamiento: 0.92 Exactitud de Prueba: 0.58	Porcentaje de certeza: 58.26 Precision: 0.56 Recall: 0.56 Medida F1: 0.56
SMOTE	Mfccs, Espectogrfamas de Mel Chroma 33 coeficientes	Épocas: 300 Batch_size =12 Optimizer=adam Metricas de entrenamiento = Exactitud, Medida F1	Exactitud de Entrenamiento: 0.93 Exactitud de Prueba: 0.62	Porcentaje de certeza: 62.37 Precision: 0.57 Recall: 0.56 Medida F1: 0.56
ADASYN	Mfccs, Espectogrfamas de Mel Chroma 33 coeficientes	Épocas: 300 Batch_size =12 Optimizer=adam Metricas de entrenamiento = Exactitud, Medida F1	Exactitud de Entrenamiento: 0.85 Exactitud de Prueba: 0.63	Porcentaje de certeza: 63.19 Precision: 0.57 Recall: 0.55 Medida F1: 0.54

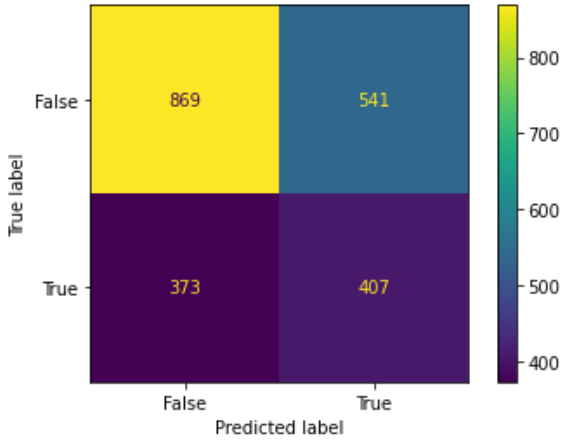
a)



Sensibilidad=60.76%

Especificidad=55.46%

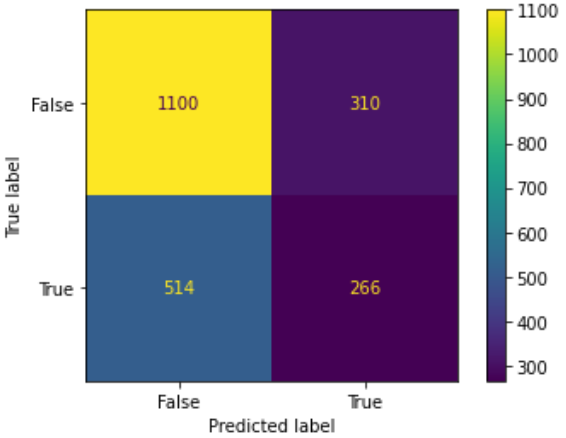
b)



Sensibilidad= 52.17%

Especificidad=61.63%

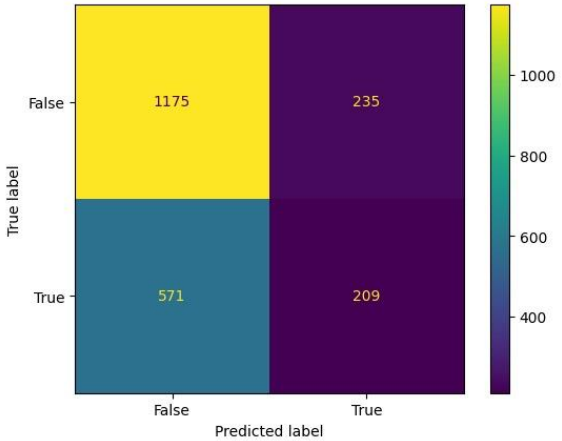
C)



Sensibilidad=34.10%

Especificidad=78.01%

d)



Sensibilidad=26.79%

Especificidad=83.33%

Figura 34 Matriz de confusión de resultados de con conjunto de prueba. CNN con 3 tensores entrada con segmentos de comparador de histéresis digital de BD CICESE. a) RUS. b) ROS. c) SMOTE. d) ADASYN

De acuerdo a la tabla 13 y figura 39. Los resultados de aplicar técnicas convencionales mostraron que, para la BD CICESE segmentada con el comparador de histéresis digital, SMOTE presenta el mejor resultado considerando las demás técnicas aplicadas para balancear los datos, con una sensibilidad de 34.10% y especificidad de 78.01%. La técnica de ADASYN tiene la sensibilidad más baja con 26.79%. RUS presenta la especificidad más baja con 55.46%.

5.3.2 Técnica de aumento de datos (Data augmentation)

Las técnicas que se implementaron son: Noise, Time strech, Pitch Scaling, Random gain, Polarity inversion. Se utilizaron en los segmentos de la BD de CICESE obtenidos por segmentación por comparador de histéresis digital, además del marco de tiempo de 344 muestras de la señal calculado en el subtema 5.1.1 de este capítulo. La caracterización de los datos es por MFCC, Espectrograma de Mel, y Chroma en la representación 3D, utilizando 33 coeficientes por cada técnica de caracterización. Los resultados de balanceo por aumentación se encuentran en tabla 14.

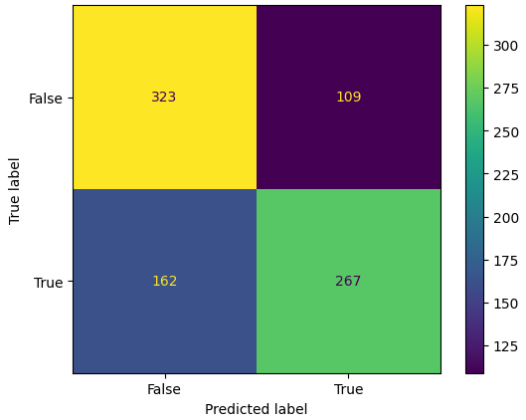
En la figura 40 se presentan las matrices de confusión de los resultados de conjunto de prueba o test.

De acuerdo a la tabla 14 y figura 36. Los resultados de aplicar técnicas de aumento de datos mostraron que, para la BD CICESE segmentada con el comparador de histéresis digital, Pitch scaling, Time strech y Random gain presentan el mejor resultado de sensibilidad y especificidad respecto a las demás técnicas aplicadas para balancear los datos. Picth scaling tiene una sensibilidad de 55.24% y especificidad de 82.40%. Time strech con una sensibilidad de 51.28% y especificidad de 85.18%. Random gain con una sensibilidad de 51.04% y especificidad de 94.67%. La especificidad más baja corresponde a utilizar aumento de datos a clase positiva utilizando Polarity inversion. Aunque la sensibilidad más baja corresponde a Random gain con 51.04% forma parte una de las técnicas que mejores resultados presenta, porque la especificidad es la más alta.

Tabla 14 Resultados de entrenamiento de CNN con tensor 3D con datos segmentados de BD CICESE utilizando balance de clases por aumento de datos

Experimentos con segmentos comparador de histéresis digital	Partición de datos Entrenamiento/ Prueba	Parámetros de entrenamiento	Resultados de entrenamiento y prueba	Métricas de prueba
Noise 0.5	Del total de los datos se consideraron el 50% de los datos dónde: 20 Prueba 80 Entrenamiento	Épocas: 100 Batch_size =128 Optimizer=adam Metricas de entrenamiento = Exactitud, Medida F1	Exactitud de Entrenamiento: 0.98 Exactitud de Prueba: 0.68	Porcentaje de certeza: 68.52 Precision: 0.68 Recall: 0.68 Medida F1: 0.68
Random gain 2 a 8	Del total de los datos se consideraron el 50% de los datos dónde: 20 Prueba 80 Entrenamiento	Épocas: 100 Batch_size =128 Optimizer=adam Metricas de entrenamiento = Exactitud, Medida F1	Exactitud de Entrenamiento: 0.73 Exactitud de Prueba: 0.7	Porcentaje de certeza: 72.93 Precision: 0.78 Recall: 0.72 Medida F1: 0.71
Time strech 0.8	Del total de los datos se consideraron el 50% de los datos dónde: 20 Prueba 80 Entrenamiento	Épocas: 100 Batch_size =12 Optimizer=adam Metricas de entrenamiento = Exactitud, Medida F1	Exactitud de Entrenamiento: 0.79 Exactitud de Prueba: 0.6	Porcentaje de certeza: 68.29 Precision: 0.70 Recall: 0.68 Medida F1: 0.67
Pitch Scaling 10	Del total de los datos se consideraron el 50% de los datos dónde : 20 Prueba 80 Entrenamiento	Épocas: 100 Batch_size =12 Optimizer=adam Metricas de entrenamiento = Exactitud, Medida F1	Exactitud de Entrenamiento: 0.78 Exactitud de Prueba: 0.68	Porcentaje de certeza: 68.87 Precision: 0.70 Recall: 0.68 Medida F1: 0.68
Polarity inversion	Del total de los datos se consideraron el 50% de los datos dónde : 20 Prueba 80 Entrenamiento	Épocas: 300 Batch_size =12 Optimizer=adam Metricas de entrenamiento = Exactitud, Medida F1	Exactitud de Entrenamiento: 0.55 Exactitud de Prueba: 0.55	Porcentaje de certeza: 55.05 Precision: 0.55 Recall: 0.55 Medida F1: 0.54

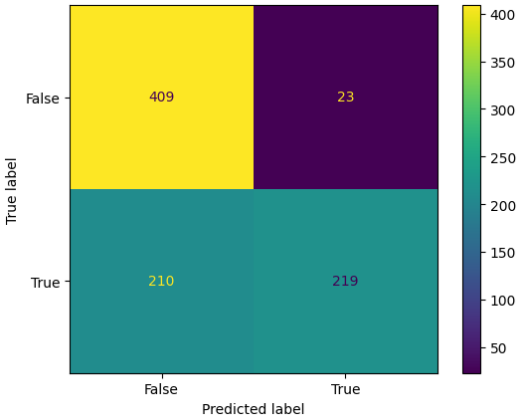
a)



Sensibilidad=62.23%

Especificidad=74.76%

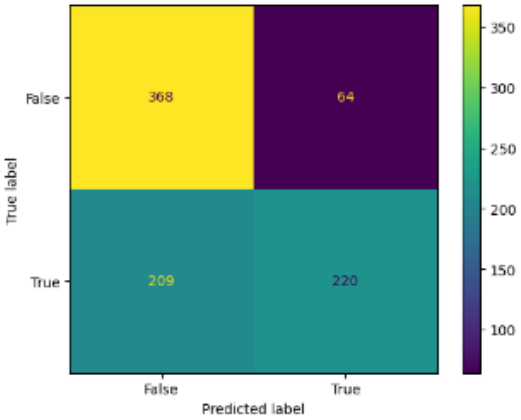
b)



Sensibilidad=51.04%

Especificidad=94.67%

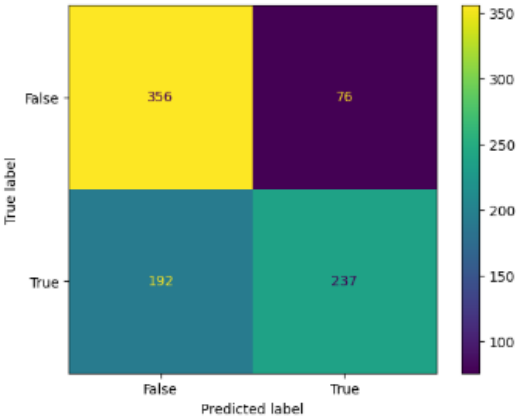
c)



Sensibilidad=51.28%

Especificidad=85.18%

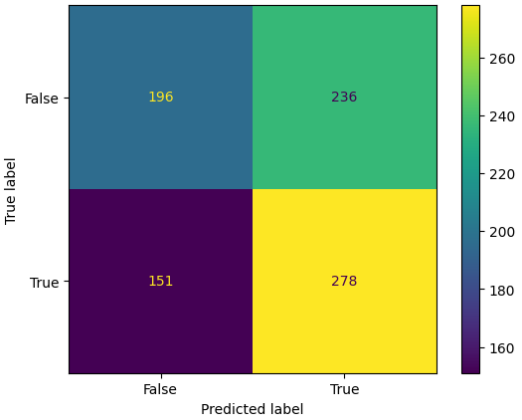
d)



Sensibilidad=55.24%

Especificidad=82.40%

e)



Sensibilidad=64.80%

Especificidad=45.37%

Figura 35 Matriz de confusión de resultados de con conjunto de prueba. CNN con 3 tensores entrada con segmentos de comparador de histéresis digital de BD CICESE. a) Noise. b) Random gain. c) Time strech. d) Pitch Scaling. e) Polarity inversion

5.3.3 Generación sintética por autoencoder (VAE)

Los Autoencoder variacionales permiten generar muestras sintéticas de audio a partir de arquitectura de red como, codificadores y decodificadores. En este trabajo se balancearon las clases generando muestras sintéticas de audio para la clase minoritaria, la cual es la positiva. Sin embargo, también se generaron muestras negativas sintéticas, para observar el comportamiento de entrenamiento de la CNN con tensor 3D cuando se tiene datos generados sintéticamente para ambas clases. Los datos con que se generaron muestras sintéticas con VAE son los segmentos obtenidos por el comparador de histéresis digital de la BD CICESE. Los resultados se observan en la tabla 15.

Tabla 15 Resultados de entrenamiento de CNN con tensor 3D con datos segmentados de BD CICESE utilizando balance de clases por generación sintética por VAE.

Experimentos con segmentos comparador de histéresis digital	Representación	Conjunto de entrenamiento y prueba	Parámetros de entrenamiento	Resultados
Generación sintética a clase positiva para igual el número de instancias en clase negativa	MFCC, Chroma, Espectograma de Mel. 33 características, ventana de 344	20 Prueba 80 Entrenamiento	Épocas: 150 Batch_size =32 Optimizer=adam Metricas de entrenamiento = sensibilidad, especificidad	Especificidad de Entrenamiento: 0.96 Sensibilidad de Entrenamiento: 0.96 Especificidad de Prueba: 0.66 Sensibilidad de Prueba: 0.66
Generación sintética a clase positiva y negativa. Se generaron muestras sintéticas para clase positiva hasta igual clase negativa y posteriormente Se aplico RUS para clase negativa y se agregaron datos faltantes para igualar número de instancias en clase positiva.	MFCC, Chroma, Espectograma de Mel. 33 características, ventana de 344	20 Prueba 80 Entrenamiento	Épocas: 150 Batch_size =32 Optimizer=adam Metricas de entrenamiento = sensibilidad, especificidad	Especificidad de Entrenamiento: 0.93 Sensibilidad de Entrenamiento: 0.93 Especificidad de Prueba: 0.64 Sensibilidad de Prueba: 0.64

De acuerdo a la tabla 15. Los resultados de los experimentos de generación sintética de audio para balancear las clases a través de VAE, mostraron que generar audio sintético solo en la clase positiva para igual el número de muestras de la clase negativa que es la mayoritaria es mejor que agregar datos

sintéticos en clase positiva y negativa, considerando agregar un 30% de datos sintéticos de clase negativa y generar datos sintéticos de clase positiva hasta igual el número de instancias en ambas clases.

5.4 Técnicas de balanceo de clases con Auto ML

Tomando en cuenta los resultados de la clasificación y las distintas técnicas de balanceo de clases sobre el conjunto de datos de CICESE segmentados con el comparador de histéresis digital se aplicó Auto ML, para encontrar los hiperpárametros que optimizarían mejor entrenamiento de la CNN con tres tensores.

El espacio de búsqueda de los parámetros se presenta a continuación:

- Dropout 1: con posibles valores de 0.25 a 0.75 para el primer bloque de capa de abandono CNN.
- Dropout 2: con posibles valores de 0.25 a 0.75 para el segundo bloque de capa de abandono CNN.
- Dropout 3: con posibles valores de 0.25 a 0.75 para el tercer bloque de capa de abandono CNN.
- Dropout 4: con posibles valores de 0.25 a 0.75 para el cuarto bloque de capa de abandono CNN.
- Batch_size: con posibles valores de 64 y 128, el cual es el tamaño del lote de muestras que se toman para entrenar la red para entrenar la red.
- Optimizador: con posible elección de adadelta, adam, rmprop.

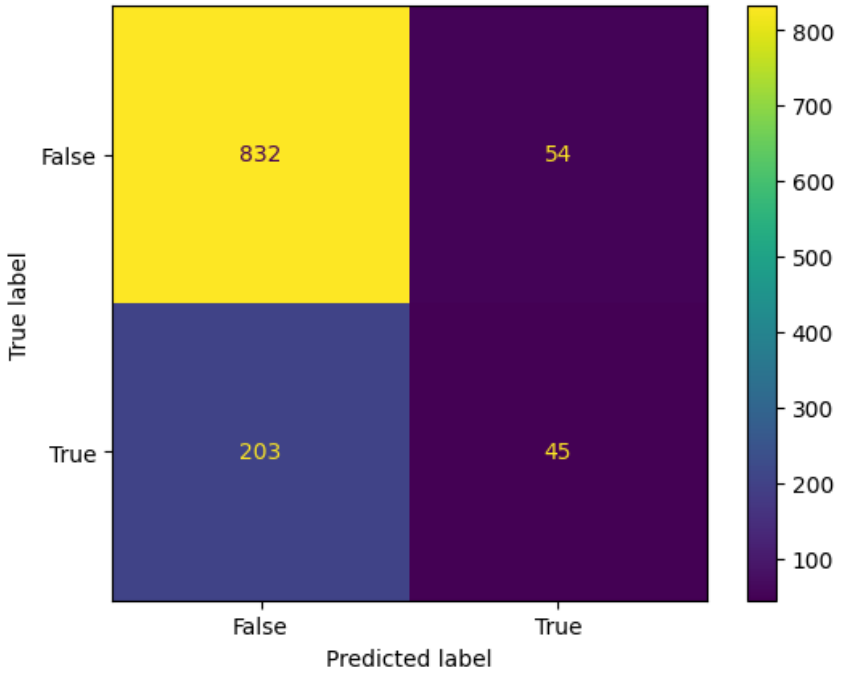
En la tabla 16 se encuentran los resultados de sensibilidad y especificidad recopilados con mejores resultados para cada una de las técnicas de balanceo de clases sin utilizar auto ml y utilizando esta herramienta de optimización en la CNN con tensor 3D. En azul se remarcan el mejor resultado.

Considerando los resultados de la tabla 16, en la figura 37 se encuentra la matriz de confusión de los resultados de especificidad más alto, correspondiente a aplicar la técnica de balanceo de clases SMOTE utilizando los hiperparámetros encontrados con Auto ML. Con un conjunto de prueba con 886 instancias en clase negativa y 248 en clase positiva

Tabla 16 Recopilación de mejores resultados de técnicas de balanceo en clase positiva utilizando CNN con tensor 3D y auto ml, en BD CICESE segmentada por comparador de histéresis digital

	Entrenamiento de CNN			**Entrenamiento de CNN con resultado de búsqueda de Hiperparámetros con AUTO ML**			
Técnica de balance en segmentos con comparador de hitersis digital en CICESE	**Hiperparámetros utilizados en CNN 3D**	**Métricas de entrenamiento**	**Resultados**	**Métricas de entrenamiento**	**Tiempo ejecución de auto ml**	**Hiperparámetros encontrados con auto ml para CNN con tensor 3D**	**Resultados en el conjunto de prueba**
SMOTE	Épocas: 300 Dropout 1:0.2 Dropout 2:0.2 Dropout 3:0.2 Dropout 4:0.2 Batch_size :128 Optimizer:adam	Exactitud, Medida F1	Sensibilidad=34.10% Especificidad=78.01%	sensibilidad, especificidad	23 horas	Épocas: 68 Dropout 1:0.19 Dropout 2:0.2 Dropout 3:0.19 Dropout 4:0.21 Batch_size :64 Optimizer_adam	Sensibilidad=18.14% Especificidad=93.90%
Random gain, Time strech, Pitch scaling (aumento en clase positiva, utilizando las 3 técnicas)	Épocas: 300 Dropout 1:0.2 Dropout 2:0.2 Dropout 3:0.2 Dropout 4:0.2 Batch_size :128 Optimizer:adam	Exactitud, Medida F1	Sensibilidad=52.52% Especificidad=87.41%	sensibilidad, especificidad	27 horas	Épocas: 300 Dropout 1:0.18 Dropout 2:0.19 Dropout 3:0.21 Dropout 4:0.22 Batch_size :128 Optimizer:adam	Sensibilidad=61.14% Especificidad=61.14%
VAE en clase positiva	Épocas: 150 Dropout 1:0.2 Dropout 2:0.2 Dropout 3:0.2 Dropout 4:0.2 Batch_size :128 Optimizer:adam	Exactitud, Medida F1		sensibilidad, especificidad	39 horas	Épocas: 300 Dropout 1:0.476 Dropout 2:0.57 Dropout 3:0.39 Dropout 4:0.53 Batch_size :128 Optimizer:rmsprop	Sensibilidad=1.37% Especificidad=99.75%

Utilizar la herramienta de callbacks para guardar el mejor resultado de entrenamiento funciono mejor que no utilizar, puesto el modelo que guardo la herramienta de callback presenta una sensibilidad de 18.14% y especificidad de 93.90% al entrenarlo con 68 épocas. Si no se utiliza callback para SMOTE se tiene una sensibilidad de 1.9% y una especificidad de 99.02% en el conjunto de prueba.



Sensibilidad=18.14%

Especificidad=93.90%

Figura 36 Matriz de confusión para SMOTE con callback en modelo de entrenamiento, con resultado de 68 épocas

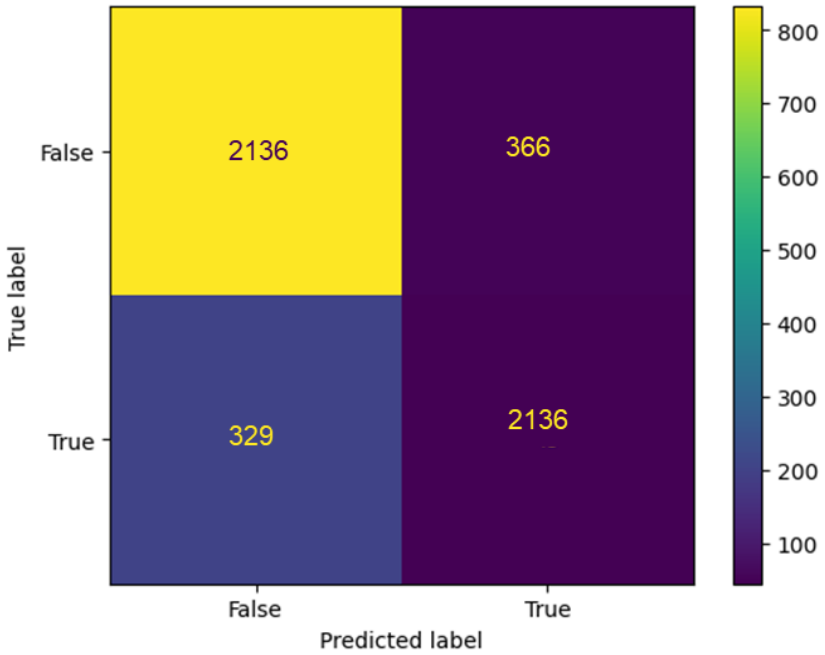
5.5 Clasificación con Random Forest

Conjuntamente a la clasificación con CNN con 3 tensores se realizó el experimento de los conjuntos de características extraído con Open smile mencionados en capítulo 4, subtema 4.4.3 con la clasificación del algoritmo de Random forest en weka.

En la clasificación con Weka primero se extrajo un conjunto de características con ayuda de Open smile, el cual corresponde al archivo de configuración de GEMAPSV01, el contiene 63 características. Al utilizar la herramienta de weka con el archivo de GEMAPSV01b con 63 características por cada instancia de la de base datos CICESE segmentada con la técnica del comparador de histéresis digital, se emplea el método de balanceo de clases resample con un porcentaje de muestra de 70% con el algoritmo de aprendizaje

automático Random Forest a validación cruzada de 10 folds se obtienen como resultado la matriz de confusión de la figura 38. Dónde para calcular sensibilidad y especificidad se considera lo siguiente:

- FN=329
- FP=366
- TP=2136
- TN=2173



Especificidad=86.65%

Sensibilidad=85.58%

Figura 37 Matriz de confusión de weka con características GEMAPSV01b clasificando con Random forest

Continuamente se evaluaron el conjunto de características Emolarge y Is10_paraling extraídos de los segmentos por comparador de historiéis digita de la base de datos CICESE. En la tabla 17. Se observan los resultados de sensibilidad y especificidad utilizando el algoritmo de Random forest de weka.

Tabla 17 Resultados conjuntos de características de Emolarge e Is10_aparaling clasificados con Random forest en weka**.**

Conjunto de características	Número de descriptores acústicos extraídos	Sensibilidad	Especificidad
Emolarge	6552	80.14%	78.34%
Is10_paraling	1582	85.75%	83.13%

Los resultados de la tabla 17 correspondientes a los conjuntos de Emolarge y Is10_parlaing muestran que son menos prometedores que las 63 características del conjunto de datos de GEMAPSvb01 extraídos de los segmentos de la base de datos CICESE con la técnica de comparador de histéresis digital.

5.6 Síntesis de resultados

A continuación, es se presenta en la tabla 18 los resultados obtenidos en la metodología para detección de COVID-19 por CNN con tensor 3D. Con la finalidad de resumir los resultados de los experimentos realizados en este trabajo. Los resultados corresponden al conjunto de prueba. En azul se remarcan los mejores resultados de cada tipo de técnica de balanceo de clases.

La tabla 18 muestra los resultados de sensibilidad y especificidad de las distintas técnicas de balanceo de clases en la BD CICESE segmentada con la técnica de comprador de histéresis digital. Utilizar audios segmentados sin balancear tiene una sensibilidad de 53.92% y especificidad de 87.98%. utilizar técnicas convencionales hace que la sensibilidad y especificidad bajen, SMOTE es la técnica que en promedio de sensibilidad y especificidad respecto a las demás consideradas como convencionales sea la técnica que mejor resultados tiene con una sensibilidad de 34.10% y especificidad de 78.01%. Aplicar técnicas de balanceo de clases con aumento de datos, mejora la sensibilidad y especificidad que al tener la BD desbalanceada. Se observa que Random gain sube la especificidad a 94.67%, y Pitch scaling mejora la sensibilidad a 55.24%. Sin embargo, Time strech y Pitch scaling considerando el promedio de sensibilidad y especificidad son las tecnicas de generación de datos que obtienen mejores resultados a comparación de aplicar Noise y Polarity inversion. Utilizar generación de audio sintético por VAE para la clase positiva hasta igualar el mismo número de instancia en clase negativa se observa que baja la sensibilidad y especificidad que al no utilizar audio sintético en la clase positiva y tener BD desbalanceada.

Tabla 18 Resumen de resultados con CNN tensor 3D

Metodología con CNN con tensor 3D para BD CICESE:	Sensibilidad en conjunto de prueba	Especificidad en conjunto de prueba
Segmentación por histéresis digital CNN con tensor 3D (sin balance de clases)	53.92%	87.98%
Segmentación por histéresis digital CNN con tensor 3D: Balance por técnicas convencionales		
ROS	52.17%	61.63%
RUS	60.76%	55.46%
ADASYN	26.79%	83.33%
SMOTE	34.10%	78.01%
Segmentación por histéresis digital CNN con tensor 3D: Balance por técnicas de generación		
Noise	62.23%	74.76%
Time strech	51.28%	85.18%
Pitch Scaling	55.24%	82.40%
Polarity inversión	64.80%	45.37%
Random gain	51.04%	94.67%
Segmentación por histéresis digital CNN con tensor 3D: Balance por generación sintética		
VAE	66.0%	66.0%
Mejores resultados de cada técnica de balance aplicando Auto ML a entrenamiento de CNN con tensor 3D		
SMOTE	18.14%	93.90%
Random gain, Time strech, Pitch scaling (aumento en clase positiva, utilizando las 3 técnicas)	61.14%	61.14%
VAE en clase positiva	1.37%	99.75%

Considerando estos resultados de cada técnica de balanceo de clases se tomaron las mejores para entrenar el modelo CNN con tensor 3D con parámetros que se encontraron con la herramienta de auto ml como los mejores para entrenamiento. Utilizar SMOTE sin balancear da como resultado 34.10% y especificidad de 78.01% utilizando los parámetros encontrados con auto ml subió la especificidad a un 93.90% y a la sensibilidad a 18.14% esto significa que el modelo es mejor para detectar a los pacientes negativos a COVID. Combinar las tres técnicas de generación de datos que mejores resultados presentaron (Random gain, Time strech y Pitch scaling) y entrenar la CNN con tensor 3D con los parámetros encontrados con auto ml da como resultado sensibilidad y especificidad de 61.14%, aquí se observa que

aplicar auto ml baja el rendimiento del modelo. Balancear los datos con generación sintética y entrenar modelo con parámetros encontrados con auto ml aumenta la especificidad a un 99.75% y disminuye la sensibilidad a un 1.37%.

De lo anterior se resalta que utilizar parámetros ajustados con auto ml únicamente mejora la especificidad con generación sintética, sin embargo, considerando sensibilidad y especificidad los mejores resultados corresponden a SMOTE, puesto que la especificidad es de 93.90% y sensibilidad de 18.14% esto pone en evidencia que el modelo CNN con tensor 3D es más eficiente para detectar a pacientes sanos con datos balanceados con SMOTE, sin embargo la sensibilidad es baja en contraste a que los datos no hubiesen sido balanceados.

A continuación, es se presenta en la tabla 19 los resultados obtenidos en la metodología para detección de COVID-19 por Random forest de weka. En azul se remarcan el mejor resultado.

Tabla 19 Resumen de resultados de Random forest de weka

Metodología con Random Forest de weka para BD CICESE	**Sensibilidad en conjunto de prueba**	**Especificidad en conjunto de prueba**
Segmentación por comprador de histéresis digital con Random forest (sin balance de clases)	70%	71%
Balance por resample de weka con conjunto de datos segmentado por comprador de histéresis digital		
Con descriptores de Emolarge	80.14%	78.34%
Con descriptores Is10_paraling	85.75%	83.13%
Con descriptores GEMAPSvb01	85.58%	86.65%

Segmentar BD CICESE con comparador de histéresis digital y clasificar con el algoritmo de Random forest en weka obtiene 70% en sensibilidad y 71% en especificidad, sin que los datos estén balanceados. Balancear los datos del conjunto de características de GEMAPSvb01 con el algoritmo de RUS en weka llamado “resample” tiene una sensibilidad de 85.58% y sensibilidad de 86%. Estos resultados muestran que aumenta la sensibilidad y especificidad respecto a no balancear la BD. Considerando los resultados de tabla 21, balancear con resample de weka hace que mejore la sensibilidad, sin embargo, la especificidad

baja, respecto a SMOTE con hiperparámetros de auto ml, SMOTE obtiene un 93.90% y sensibilidad de 18.14%.

Considerando que la finalidad de realizar un modelo detección de COVID-19 o cualquier otra enfermedad respiratoria, se desea que sea capaz detectar correctamente a los pacientes enfermos como enfermos y a los sanos como sanos, entonces clasificar con Random forest y las características de GEMAPSvb01 parece que da una sensibilidad (capacidad para detectar correctamente a los enfermos) y especificidad (capacidad para detectar correctamente a los sanos) mejor que entrenar con CNN con tensor 3D y utilizar SMOTE como técnica de balanceo de clases.

Sin embrago, clasificar con Random forest en weka presenta una limitación la cual recae que los algoritmos de la herramienta weka ya están definidos y no se pueden manipular variables del algoritmo para ajustar el entrenamiento.

En la figura 39 se observan los resultados de las técnicas utilizadas en la metodología presentada en el capítulo 4.

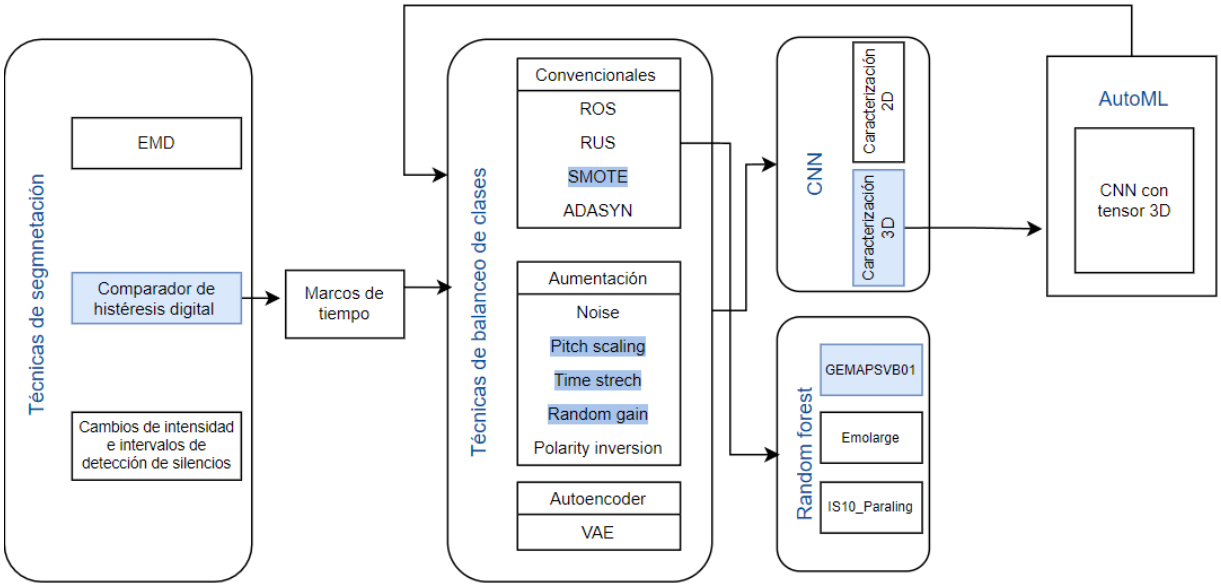


Figura 38 Metodología para detección clasificación automática de tos.

En azul están marcados los procesos que dieron mejor resultado respecto a las demás técnicas que se probaron al realizar la tarea. Primero se segmentaron los datos de la BD, para segmentar los audios, a través de identificar y seleccionar solo la información necesaria de un archivo de audio que es la tos, la

técnica de por comparador de histéresis digital resultó ser mejor. Posteriormente se realiza el cálculo de los marcos de tiempo para balancear los datos con distintas técnicas de balanceo de las cuales en las convencionales obtuvo mejores resultados con SMOTE, en aumentación de datos Pitch scalind, Time strech y Random gain entrenando con CNN con tensor 3D. Continuamente se aplica la búsqueda de hiperparámetros para la red CNN con tensor 3D con auto ml para entrenar de nuevo la CNN con estos parámetros con los datos balanceados con SMOTE.

Clasificar con Random forest el conjunto de características GEMPASvb01 balanceadas con resample de weka es el mejor resultado de sensibilidad y especificidad obtenido para el objeto de estudio de este trabajo. Sin embargo, la figura 39 muestra la metodología que se puede seguir para clasificar una BD de enfermedades respiratorias a través de la tos. Aunque no se haya mejorado los resultados de sensibilidad y especificidad del estado del arte, la aportación principal de este trabajo radica en la metodología integral de detección de enfermedades respiratorias por tos, la cual no se presenta en ningún otro trabajo actual. Ya que solo se centran en clasificar y no en enfatizar en metodología que consideren sólo la información necesaria a través de la segmentación, y abordando problemas comunes de las BD medicas como es el desbalanceo. Además, que no se encontró evidencia de que otros trabajos aborden la comparación de distintos tipos de balanceo de clases determinando que, para la tos, es mejor utilizar algoritmos convencionales en vez de generación sintética de audio para balancear los datos. Este trabajo aborda las deficiencias que otros trabajos presentan como la utilización de segmentación de tos y una propuesta de evaluación de las técnicas de segmentación, y la comparación de balanceo con distintas técnicas.

Capítulo 6. Conclusiones y trabajo futuro

Este trabajo ha pretendido diseñar una metodología integral que permita la detección automática de enfermedades respiratorias en las que una de su sintomatología clave sea la tos. Abordando la principal característica que presentan las bases de datos médicas la cual, es el desbalanceo de clases, no enfatizar soluciones a este problema trae sesgos de clasificación para los modelos de aprendizaje automático. Para este caso de estudio se abordó el COVID-19 ya que ha sido la enfermedad que recientemente a impactado a todo el mundo.

El objetivo general de esta investigación fue desarrollar una metodología de clasificación para identificar la presencia de COVID de tal modo que minimice el efecto negativo de contar con una base de datos desbalanceada mejorando la sensibilidad y especificidad del modelo.

6.1 Conclusiones

Para lograr el objetivo de esta tesis se llevó a cabo un proceso de familiarización del base de datos que se utilizaría en esta investigación, la cual fue recopilada por CICESE-UT3, en dónde las muestras que conforman el conjunto de datos están clínicamente validadas por la prueba qRT-PCR para detección del SARS-Cov-2 (COVID-19). Esto permitió que el entrenamiento de los algoritmos de aprendizaje automático utilizados, aprendan patrones con características reales del diagnóstico de la enfermedad en cuestión.

En el preprocesamiento de la señal, se realizó una segmentación de las señales acústicas de la tos con la finalidad de obtener las partes de los audios que contienen la información útil para realizar la clasificación. La segmentación se realizó con tres diferentes técnicas, comparador de histéresis digital EMD y cambio de intensidad e intervalos de detección de silencio. De las cuales, de acuerdo a los resultados se observa que la mejor técnica para segmentar silencios de toses y mantener la tos, es la técnica de comparador de histéresis digital (ver Capítulo 5). Esta presenta una ventaja sobre las demás, la cual radica en obtener segmentos de tos con duraciones similares entre todos los segmentos generados, lo que permite que los segmentos no sean tan variables, esto beneficia a que la ventana favorable calculada abarque la mayor cantidad de segmentos con base en su duración (ver Capítulo 5, subtema 5.1.1). La técnica de segmentación por comparador de histéresis digital funcionó mejor puesto que los audios originales no contenían tanto ruido de fondo, haciendo que los cambios de amplitud de la señal correspondieran

mayormente a fragmentos de tos. Además, que considera la duración específica de tos y el tiempo de inicio que se debe considerar como tos después que se encuentra un pico de amplitud en la señal, esto hace que los fragmentos se encuentren en un rango similar de duración. A comparación de las demás técnicas de segmentación que resultan en fragmentos muy grandes o muy pequeños, cortando o generalizando duraciones de toses.

En la caracterización de la señal los resultados mostraron que la representación tridimensional en una CNN obtiene mejores resultados aplicando la técnica de segmentación por comparador de histéresis digital. Esto se debe a que la arquitectura de CNN aprende mejor con una representación tridimensional, con técnicas de caracterización de audio las cuales se ha demostrado que son las que obtienen mejores resultados al tratarse de clasificación de señales acústicas, las cuales son los MFCC, Chroma, y Espectrogramas de Mel. Estas tres representaciones parecen aportar información complementaria y organizadas en la estructura de CNN con tensor 3D permitiendo que el algoritmo pueda aprovechar la información en el aprendizaje de cada una de las representaciones (ver Capítulo 5, subtema 5.2.2).

Para abordar el problema de desbalanceo de clases, se utilizaron tres tipos de técnicas de balanceo para entrenar la CNN con 3 tensores con los datos equilibrados. Los resultados arrojan que de las técnicas convencionales SMOTE tiene mejor especificidad y sensibilidad (ver Capítulo 5, subtema 5.3.1). Para las técnicas de aumento de datos considerando la modificación de la forma de onda de la señal acústica, el Random gain, Pitch scaling y Time strech incrementan la sensibilidad y especificidad (ver Capítulo 5, Sección 5.3.2). Por último, utilizar VAE para generación sintética de audio para la clase positiva resulto mejor que agregar datos sintéticos para la clase positiva y negativa (ver Capítulo 5, subtema 5.3.3).

Con base en los resultados, para balancear base de datos de toses, resulta mejor utilizar técnicas convencionales, sí el objetivo es aumentar la especificidad de tamizaje para detección de enfermedades respiratorias. Sí la finalidad es balancear los datos y hacer que la predicción de los algoritmos automáticos sea más robusta siendo capaz de detectar más variedad de datos dentro de una misma clase se puede considerar utilizar técnicas como Random gain, Pitch scaling y Time strech, obteniendo resultados de sensibilidad y especificidad más cercanos entre sí. No se recomienda, utilizar generación sintética a través de VAE para la clase mayoritaria, hace que existan sesgos en predicciones, aun así, cuando se esperaba que aumentar datos a través de audio sintético para clase minoritaria y mayoritaria el aprendizaje fuese más robusto, esto no fue así, la sensibilidad y especificidad es baja.

Al obtener los mejores resultados para balancear clases en cada uno de los tipos de técnicas, se realizó una búsqueda automática de hiperpárametros con auto ml. Con la finalidad de encontrar los que maximicen el rendimiento del modelo de CNN con las técnicas que de acuerdo a los resultados obtienen mejor sensibilidad y especificidad. Los resultados arrojaron que entrenar la CNN con los parámetros encontrados con auto ml y con datos balanceados con SMOTE obteniendo una sensibilidad=18.14% y especificidad de 93.90%, resultan mejores que combinar Random gain, Pitch scaling y Time strech, y que utilizar VAE para generar audio sintético (ver Capítulo 5, subtema 5.3.4). Esto permite analizar que generar datos sintéticos o modificados a partir de los originales incrementa el sesgo de clasificación por las perturbaciones que sufre la forma de onda de la señal acústica de la tos.

Los resultados de utilizar otro conjunto de características como lo es GEMAPS01b entrenado con el algoritmo de Random forest y con la técnica de balanceo de clases de resample arrojan una sensibilidad de 86.65% y sensibilidad de 85.58% (ver Capítulo 5, subtema 5.5). Estos son los resultados de sensibilidad y especificidad más altos en comparación de los resultados de las técnicas convencionales, por aumento de datos, y generación sintética.

En la figura 40 se presenta la metodología integral para detección automática de COVID-19 a través de análisis acústico de la tos, considerando las técnicas de segmentación, balanceo de clases, caracterización y clasificación automática que mejores resultados presentaron.

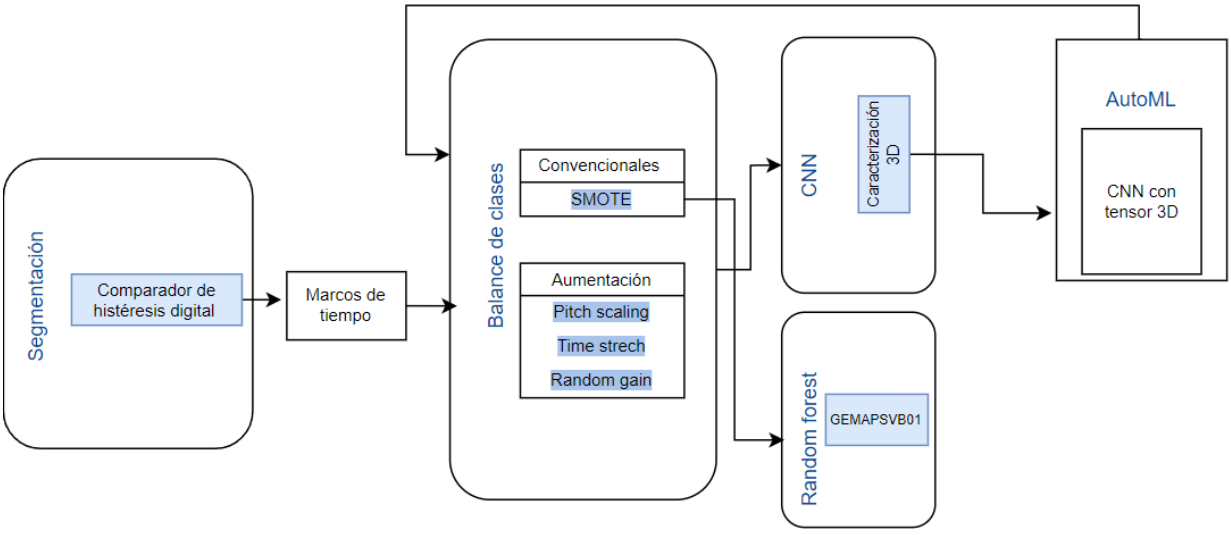


Figura 39 Metodología integral

A diferencia de los trabajos reportados en los trabajos relacionados se decidió abordar la problemática del desbalanceo de clases probando distintas técnicas. Y sobre todo la segmentación de las toses, puesto que de acuerdo a los resultados de la segmentación utilizar toses completas o los archivos de audio sin segmentar minimiza la sensibilidad y especificidad debido a que el modelo aprende características innecesarias de las partes donde hay silencios en los audios completos, metiendo ruido al entrenamiento del modelo de CNN.

6.2 Contribuciones

Tras la realización de este trabajo se realizaron contribuciones al estado actual de la predicción de enfermedades respiratorias a través de la tos.

Usamos técnicas de segmentación de audio para detección y extracción de tos. Esto permite extraer sólo la información necesaria para el aprendizaje en los modelos automáticos.

Presentamos un nuevo método de evaluación de técnicas de segmentación de audio, identificando el método que con mejores resultados para segmentación automática de toses.

Presentamos una forma de calcular marcos de tiempo adecuados para cualquier conjunto de señales acústicas como entrenamiento para algoritmos de aprendizaje automático.

Identificamos las técnicas de caracterización de audio que más se utilizan en el estado del arte, hasta este momento. Abordando dos tipos de representaciones, bidimensional y tridimensional.

Evaluamos distintas técnicas de balanceo de clases, identificando que utilizar generación sintética y modificación de la forma de onda de la señal de la tos no resultan favorables para la clasificación de tos saludable y no saludable.

Presentamos una propuesta de un método integral para detección automática de enfermedades respiratorias a través de sonidos de la tos. Este método se encuentra en la figura 39. Del capítulo 5.

6.3 Limitaciones

Una de las limitaciones importantes de este trabajo de tesis radica en que la base de datos que se utilizó tiene muestras de una población especifica, en este caso de trabajadores y personal cercano a la Universidad Autónoma de Nayarit, perteneciendo a un perfil específico.

6.4 Trabajo futuro

Conforme se desarrolló este trabajo surgieron nuevas vertientes que podrían enriquecerlo, a pesar de que no se desarrollaron quedan como propuesta para trabajo a futuro y se mencionan a continuación:

En este trabajo de tesis se utilizó una base de datos de toses de COVID-19 positivas y negativas clínicamente validadas, con la finalidad de proponer un método integral de clasificación automática de tos, sin embargo, este método también podría funcionar para detectar toses enfermas y sanas de enfermedades respiratorias. Como trabajo futuro se podría implementar otro conjunto de datos de diversas enfermedades respiratorias, por ejemplo, al EPOC (Enfermedad respiratoria obstructiva), influenza, nuevas variantes de COVID, asma, la fibrosis pulmonar, neumonía, cáncer de pulmón, etc.

Otro enfoque que podría abordarse es la incorporación de nuevas clases en el conjunto de datos, como: datos demográficos, en donde se incluya, la edad de las personas, el sexo, los síntomas que presentan al realizar la recolección de datos, condiciones de salud generales. Esto con la finalidad de analizar la influencia de estos datos al clasificar enfermedades, aumentando la posibilidad de establecer relaciones entre las clases, la cuales se pudieran explicar clínicamente.

De los trabajos relacionados en clasificación al parecer no se aborda la transferencia de aprendizaje, esto pudiera permitir aprender de las características de la tos en enfermedades respiratorias específicas.

Se pueden extender la BD con muestras de participantes de otro estado, o país con la finalidad de probar la metodología con poblaciones que no tienen las ismas características demográficas, socioeconómicas, etc.

Literatura citada

Aiyegbusi, O. L. (2021). Symptoms, complications and management of long COVID: a review. Journal of the Royal Society of Medicine, 114(9), 428-442. doi: 10.1177/01410768211032850

Akhtar, N. y. (2020). Interpretation of intelligence in CNN-pooling processes: a methodological survey. *Neural computing and applications, 32*(3), 879-898. doi:https://doi.org/10.1007/s00521-019-04296-5

Alqudaihi, K. S. (2021). Cough Sound Detection and Diagnosis Using Artificial Intelligence Techniques: Challenges and Opportunities. *IEEE Access, 9*, 102327-102344. doi:10.1109/ACCESS.2021.3097559

Alsaif, S. A. (2021). Impact of data balancing during training for best predictions. *Informatica, 45*(2). doi:https://doi.org/10.31449/inf.v45i2.3479

Amal, M. I. (2022). Early Detection of Covid-19 Through Cough Sound Recognition using LPC and K-NN algorithm. *In 7th International Conference on Sustainable Information Engineering and Technology*, 90-97. doi:https://doi.org/10.1145/3568231.3568253

Andreu-Perez, J. P.-E.-P.-T. (2021). A generic deep learning based cough analysis system from clinically validated samples for point-of-need covid-19 test and severity levels. *IEEE Transactions on Services Computing, 15*(3), 1220-1232. doi:doi: 10.1109/TSC.2021.3061402

Andrews, P. L. (2021). COVID-19, nausea, and vomiting. *Gastroenterology and hepatology*, 646-656. doi:https://doi.org/10.1111/jgh.15261

Andrews, P. L. (2021). COVID-19, nausea, and vomiting. *Journal of gastroenterology and hepatology, 36*(3), 646-656. doi:https://doi.org/10.1111/jgh.15261

Aphex34. (16 de diciembre de 2015). *Aphex34, CC BY-SA 4.0 <https://creativecommons.org/licenses/by-sa/4.0>, via Wikimedia Commons.* Typical CNN architecture: https://commons.wikimedia.org/wiki/File:Typical_cnn.png

Bagad, P. D. (2020). Cough Against COVID: Evidence of COVID-19 Signature in Cough Sounds. *ArXiv*. doi:/abs/2009.08790

Bergstra, J. Y. (2013). Hyperopt: A python library for optimizing the hyperparameters of machine learning algorithms. *Proceedings of the 12th Python in science conference, 13*, 20.

Bhattacharjee, M. G. (2020). Multilabel sentiment prediction by addressing imbalanced class problem using oversampling. *Advances in Smart Communication Technology and Information Processing: OPTRONIX, 165*, 239-249. doi:https://doi.org/10.1007/978-981-15-9433-5_23

Bi, Q. G. (2019). What is machine learning? A primer for the epidemiologist. *American journal of epidemiology, 188*(12), 2222-2239. doi:https://doi.org/10.1093/aje/kwz189

Borja-Robalino, R. M.-G. (2020). Estandarización de métricas de rendimiento para clasificadores Machine y Deep Learning. *Revista Ibérica de Sistemas e Tecnologias de Informação*(E30), 184-196.

Breiman, L. (2001). Random forests. *Machine learning, 45*, 5-32. doi:https://doi.org/10.1023/A:1010933404324

Brown, C. C. (2020). Exploring Automatic Diagnosis of COVID-19 from Crowdsourced Respiratory Sound Data. *ArXiv*. doi:https://doi.org/10.1145/3394486.3412865

Cantero, A. T.-M. (2018). Algoritmos de aprendizaje supervisado para la clasificación de géneros musicales caracterizados mediante modelos estadísticos. *Res. Comput. Sci., 147*(5), 119-128.

Caparrini, F. S. (2022). *Variational AutoEncoder.* http://www.cs.us.es/~fsancho/?e=232

Chang, A. B. (2006). The physiology of cough. *Paediatric Respiratory Reviews, 7*(1), 2-8. doi:https://doi.org/10.1016/j.prrv.2005.11.009

Chang, J. C. (2021). DiCOVA-Net: Diagnosing covid-19 using acoustics based on deep residual network for the DiCOVA challenge 2021. *arXiv preprint*. doi:/abs/2107.06126

Chawla, N. V. (2002). SMOTE: synthetic minority over-sampling technique. *Journal of artificial intelligence research, 16*, 321-357. doi:https://doi.org/10.1613/jair.953

Chen, T. y. (2016). Xgboost: A scalable tree boosting system. *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 785-794. doi:https://doi.org/10.1145/2939672.2939785

Chowdhury, N. K. (2022). Machine learning for detecting COVID-19 from cough sounds: An ensemble-based MCDM method. *Computers in Biology and Medicine, 145*, 105405. doi:https://doi.org/10.1016/j.compbiomed.2022.105405

Christlein, V. S. (2019). Deep generalized max pooling. *International conference on document analysis and recognition (ICDAR)*, 1090-1096. doi:10.1109/ICDAR.2019.00177

Cohen-McFarlane, M. G. (2019). Comparison of silence removal methods for the identification of audio cough events. *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 1263-1268. doi:10.1109/EMBC.2019.8857889

Coppock, H. J. (2021). COVID-19 detection from audio: seven grains of salt. *The Lancet Digital Health, 3*(9), e537-e538. doi:https://doi.org/10.1016/S2589-7500(21)00141-2

Correa Duarte, J. A. (2014). Manual de análisis acústico del habla con Praat. doi:http://bibliotecadigital.caroycuervo.gov.co/id/eprint/998

Corso, C. L. (2009). Aplicación de algoritmos de clasificación supervisada usando Weka. *Córdoba: Universidad Tecnológica Nacional, Facultad Regional Córdoba*.

D'agostino, R. B. (1990). A suggestion for using powerful and informative tests of normality. *The American Statistician, 44*(4), 316-321. doi:10.1080/00031305.1990.10475751

Das, K. B. (2022). Socio-economic impact of COVID-19. *COVID-19 in the Environment. Elsevier*, 153-190. doi:https://doi.org/10.1016/B978-0-323-90272-4.00014-2

Dave, N. (2013). Feature extraction methods LPC, PLP and MFCC in speech recognition. *International journal for advance research in engineering and technology, 1*(6), 1-4.

DeCarlo, L. T. (1997). On the meaning and use of kurtosis. *Psychological methods, 2*(3), 292. doi:https://doi.org/10.1037/1082-989X.2.3.292

Deshpande, G. y. (2020). An overview on audio, signal, speech, & language processing for COVID-19. *arXiv*. doi:arXiv preprint arXiv:2005.08579

Devi, D. S. (2021). *Deep learning-based cognitive state prediction analysis using brain wave signal.* Academic Press. doi:https://doi.org/10.1016/B978-0-323-85769-7.00017-3

Doersch, C. (2016). Tutorial on Variational Autoencoders. *arXiv preprint arXiv:1606.05908*. doi:https://doi.org/10.48550/arXiv.1606.05908

Dong, D. T. (2020). The role of imaging in the detection and management of COVID-19: a review. *IEEE reviews in biomedical engineering, 14*, 16-29. doi:doi: 10.1109/RBME.2020.2990959.

Duda, R. O. (1973). *Pattern classification and scene analysis* (Vol. 3). New York: Wiley.

Elkfury, F. y. (2021). Clasificación y representación de emociones en el discurso hablado en español empleando Deep Learning. *RISTI-Revista Ibérica de Sistemas e Tecnologías de Información, versión impresa ISSN*, 1646-9895. doi:DOI: 10.17013/risti.42.78–92

Escudero, X. G.-F.-S.-G. (2020). La pandemia de Coronavirus SARS-CoV-2 (COVID-19): Situación actual e implicaciones para México. *Archivos de cardiología de México, 90*, 7-14. doi:https://doi.org/10.24875/acm.m20000064

Espinosa-Zúñiga, J. J. (2020). Aplicación de algoritmos Random Forest y XGBoost en una base de solicitudes de tarjetas de crédito. *Ingeniería, investigación y tecnología, 21*(3). doi:http://orcid.org/0000-0001-6828-2145

Eyben, F. S. (2015). The Geneva Minimalistic Acoustic Parameter Set (GeMAPS) for Voice Research and Affective Computing. *IEEE Transactions on Affective Computing, 7*(2), 190-202. doi:10.1109/TAFFC.2015.2457417

Eyben, F. W. (2010). Opensmile: the munich versatile and fast open-source audio feature extractor. *Proceedings of the 18th ACM international conference on Multimedia*, 1459-1462. doi:10.1145/1873951.1874246

F. Demir, D. A. (2020). A New Deep CNN Model for Environmental Sound Classification. *in IEEE Access, 8*, 66529-66537. doi:DOI: 10.1109/ACCESS.2020.2984903

Fennelly, K. P. (2020). Particle sizes of infectious aerosols: Implications for infection control. . *The Lancet Respiratory Medicine, 8*(9), 914-924. doi:https://doi.org/10.1016/S2213-2600(20)30323-4

Fotouhi, S. A. (2019). A comprehensive data level analysis for cancer diagnosis on imbalanced data. *Journal of biomedical informatics, 90*, 103089. doi:https://doi.org/10.1016/j.jbi.2018.12.003

Gabaldón-Figueira, J. C. (2022). Acoustic surveillance of cough for detecting respiratory disease using artificial intelligence. *ERJ open research, 8*(2). doi:10.1183/23120541.00053-2022

García Abad, J. (2021). Comparativa de técnicas de balanceo de datos. Aplicación a un caso real para la predicción de fuga de clientes. *Universidad de Oviedo*, 112. doi:http://hdl.handle.net/10651/60629

Gaviria, A. Z. (2023). ¿Qué sabemos del origen del COVID-19 3 años después? *Revista Clínica Española, 223*(4), 240-243. doi:https://doi.org/10.1016/j.rce.2023.02.002.

Géron, A. (2022). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow.*

Hall, M. F. (2009). he WEKA data mining software: an update. *ACM SIGKDD explorations newsletter, 11*(1), 10-18. doi:https://doi.org/10.1145/1656274.1656278

Harper, L. K.-L. (2020). The impact of COVID-19 on research. *Journal of Pediatric Urology, 16*(5), 715-716. doi:https://doi.org/10.1016/j.jpurol.2020.07.002

He, H. B. (2008). ADASYN: Adaptive synthetic sampling approach for imbalanced learning. *2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence)*, 1322-1328. doi:10.1109/IJCNN.2008.4633969

Hinojosa Cardenas, E. (2015). Generación genética multiobjetivo de bases de conocimiento Fuzzi para clasificación en bases de datos no balanceados usando el enfoque iteractivo. Obtenido de http://hdl.handle.net/20.500.12390/355

Hoang, T. P. (2022). A Cough-based deep learning framework for detecting COVID-19. *44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, 3422-3425. doi:doi: 10.1109/EMBC48229.2022.9871179

Hopkins, J. U. (16 de Abril de 2023). *Center for Systems Science and Engineerin.* Obtenido de COVID-19 Dashboard: https://www.arcgis.com/apps/dashboards/bda7594740fd40299423467b48e9ecf6

Hossin, M. y. (2015). A review on evaluation metrics for data classification evaluations. *International journal of data mining & knowledge management process, 5*(2), 1. doi:DOI : 10.5121/ijdkp.2015.5201

Huang, N. E. (1998). The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis. *Proceedings of the Royal Society of London. Series A: mathematical, physical and engineering sciences, 454*(1971), 903-995. doi:https://doi.org/10.1098/rspa.1998.0193

Huang, N. E. (1998). The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis. *Proceedings of the Royal Society of London. Series A: mathematical, physical and engineering sciences, 454*(1971), 903-995. doi:https://doi.org/10.1098/rspa.1998.0193

Ijaz, A. N. (2022). Towards using cough for respiratory disease diagnosis by leveraging Artificial Intelligence: A survey. *Informatics in Medicine Unlocked, 29*, 100832. doi:https://doi.org/10.1016/j.imu.2021.100832

Imran, A. P. (2020). AI4COVID-19: AI enabled preliminary diagnosis for COVID-19 from cough samples via an app. *Informatics in Medicine Unlocked, 20*, 100378. doi:https://doi.org/10.1016/j.imu.2020.100378

Infante, C. C. (2017). Use of cough sounds for diagnosis and screening of pulmonary disease. *IEEE global humanitarian technology conference (GHTC)*, 1-10. doi:DOI: 10.1109/GHTC.2017.8239338

Islam, R. A.-R. (2021). Early detection of COVID-19 patients using chromagram features of cough sound recordings with machine learning algorithms. *International Conference on Microelectronics (ICM)*, 82-85. doi:doi: 10.1109/ICM52667.2021.9664931

Islam, R. A.-R. (2022). A study of using cough sounds and deep neural networks for the early detection of COVID-19. *Biomedical Engineering Advances, 3*, 100025. doi:https://doi.org/10.1016/j.bea.2022.100025

Ji, T. L. (2020). Detection of COVID-19: A review of the current literature and future perspectives. *Biosensors and Bioelectronics, 166*, 112455. doi:https://doi.org/10.1016/j.bios.2020.112455

Ji, X. P. (2023). Imbalanced binary classification under distribution uncertainty. *Information Sciences, 621*, 156-171. doi:https://doi.org/10.1016/j.ins.2022.11.063

Kadambari, S. K. (2020). Why the elderly appear to be more severely affected by COVID-19: the potential role of immunosenescence and CMV. *Reviews in medical virology, 30*(5), e2144. doi:https://doi.org/10.1002/rmv.2144

Kaggle. (2020). *COVID-19 Cough Recordings*. COVID-19 Positive and Negative Patients' Cough Recordings: https://www.kaggle.com/datasets/himanshu007121/coughclassifier-trial?select=cough_trial_extended.csv

Khalilia, M. C. (2011). Predicting disease risks from highly imbalanced data using random forest. *BMC Med Inform Decis Mak, 11*, 1-13. doi:https://doi.org/10.1186/1472-6947-11-51

Ko, T. P. (2015). Audio augmentation for speech recognition. *In Sixteenth annual conference of the international speech communication association*, 3586-3589.

Kumar, V. A. (2021). Evaluating the impact of covid-19 on society, environment, economy, and education. *Sustainability, 13*(24), 13642. doi:https://doi.org/10.3390/su132413642

Kuo, K. M. (2022). The accuracy of machine learning approaches using non-image data for the prediction of COVID-19: A meta-analysis. *International journal of medical informatics*, 104791. doi:https://doi.org/10.1016/j.ijmedinf.2022.104791

Laguarta, J. H. (2020). COVID-19 artificial intelligence diagnosis using only cough recordings. *IEEE Open Journal of Engineering in Medicine and Biology, 1*, 275-281. doi:doi: 10.1109/OJEMB.2020.3026928.

Largo, C. J. (2022). Acercando los autocodificadores variacionales al gran público. *Revista de acústica, 53*(3), 3-11.

Larrazabal, A. J. (2020). Gender imbalance in medical imaging datasets produces biased classifiers for computer-aided diagnosis. *Proceedings of the National Academy of Sciences, 117*(23), 12592-12594. doi:https://doi.org/10.1073/pnas.1919012117

leland, R. (2020). *Understanding the Mel Spectrogram*. (A. Vidhya, Productor) Understanding the Mel Spectrogram: https://medium.com/analytics-vidhya/understanding-the-mel-spectrogram-fca2afa2ce53

Lella, K. K. (2022). Automatic diagnosis of COVID-19 disease using deep convolutional neural network with multi-feature channel from respiratory sound data: cough, voice, and breath. *Alexandria Engineering Journal, 61*(2), 1319-1334.

Leung, N. H. (2021). Transmissibility and transmission of respiratory viruses. *Nature Reviews Microbiology, 19*(8), 528-545. doi:https://doi.org/10.1038/s41579-021-00535-6

Liu, Y. G.-S. (2020). The reproductive number of COVID-19 is higher compared to SARS coronavirus. *Journal of travel medicine*, 1-4. doi:doi: 10.1093/jtm/taaa021

Ltd, M. (28 de junio de 2012). *Introducción al método de descomposición de modo empírico*. (MetaTrader 5) Introducción al método de descomposición de modo empírico: https://www.mql5.com/es/articles/439

Martínez, J. L. (2017). *Señales.* Unidad de Apoyo para el Aprendizaje: https://programas.cuaed.unam.mx/repositorio/moodle/pluginfile.php/824/mod_resource/content/5/contenido/index.html#contenido

Mascorro, G. A. (2013). Reconocimiento de voz basado en MFCC, SBC y Espectrogramas. *Ingenius*(10), 12-20. doi:https://doi.org/10.17163/ings.n10.2013.02

MathWorks. (2023). *Empirical mode decomposition*. Help Center: https://es.mathworks.com/help/signal/ref/emd.html#mw_e1c36026-4029-4ca4-b759-0b454f0a14c0_vh

Mazurowski, M. A. (2008). Training neural network classifiers for medical decision making: The effects of imbalanced datasets on classification performance. *Neural Networks, 21*(2-3), 427-436. doi:https://doi.org/10.1016/j.neunet.2007.12.031

McFee, B. R. (2015). librosa: Audio and music signal analysis in python. *Proceedings of the 14th python in science conference, 8*, 18-25.

Miranda, I. D. (2019). A comparative study of features for acoustic cough detection using deep architectures. *In 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2601-2605. doi:DOI: 10.1109/EMBC.2019.8856412

Miyara, F. (2003). Introducción a la Acústica. *Publicación interna de la Facultad de Ciencias Exactas, Ingeniería y Agrimensura, UNR Rosario (Arg.)*.

Molnar, C. C. (2021). Interpretable Machine Learning – A Brief History, State-of-the-Art and Challenges. En C. i. Science (Ed.), *ECML PKDD 2020 Workshops: Workshops of the European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD 2020): SoGood 2020, PDFL 2020, MLCS 2020, NFMCP 2020, DINA 2020, EDML 2020, XKDD 2020 and INRA 2020. vol 1323*, págs. 417-431. Ghent, Belgium: Cham: Springer International Publishing. doi:https://doi.org/10.1007/978-3-030-65965-3_28

Mouawad, P. D. (2021). Robust detection of COVID-19 in cough sounds: using recurrence dynamics and variable Markov model. *SN Computer Science, 2*(1), 34. doi:https://doi.org/10.1007/s42979-020-00422-6

Müller, M. (2015). Short-time fourier transform and chroma features. *Lab Course, Friedrich-Alexander-Universität Erlangen-Nürnberg*.

Mushtaq, Z. &. (2020). Environmental sound classification using a regularized deep convolutional neural network with data augmentation. *Applied Acoustics, 167*, 107389. doi:https://doi.org/10.1016/j.apacoust.2020.107389

Nair, V. y. (2010). Rectified linear units improve restricted boltzmann machines. *Proceedings of the 27th international conference on machine learning (ICML-10)*, 807-814.

Naqa, I. M. (2015). What Is Machine Learning? *El Naqa, I., Li, R., Murphy, M. (eds) Machine Learning in Radiation Oncology*. doi:https://doi.org/10.1007/978-3-319-18305-3_1

Nguyen, K. D. (2020). Temporal sub-sampling of audio feature sequences for automated audio captioning. *ArXiv*. doi:/abs/2007.02676

Orlandic, L. T. (2021). The COUGHVID crowdsourcing dataset, a corpus for the study of large-scale cough analysis algorithms. *Sci Data, 8*(1), 1-10. doi:https://doi.org/10.1038/s41597-021-00937-4

Orlandic, L. T. (2021). The COUGHVID crowdsourcing dataset, a corpus for the study of large-scale cough analysis algorithms. *Sci Data, 8*(1). doi:https://doi.org/10.1038/s41597-021-00937-4

Ozerov, A. V. (211). A general flexible framework for the handling of prior information in audio source separation. *IEEE Transactions on audio, speech, and language processing, 2*(4), 1118-1133. doi:10.1109/TASL.2011.2172425

Pahar, M. K. (2021). COVID-19 cough classification using machine learning and global smartphone recordings. *Computers in Biology and Medicine, 135*, 104572. doi:https://doi.org/10.1016/j.compbiomed.2021.104572

Pahar, M. K. (2022). COVID-19 detection in cough, breath and speech using deep transfer learning and bottleneck features., *141*, pág. 105153. doi:https://doi.org/10.1016/j.compbiomed.2021.105153

Park, D. S. (2019). Specaugment: A simple data augmentation method for automatic speech recognition. *ArXiv*. doi:https://doi.org/10.21437/Interspeech.2019-2680

Pérez, C. O. (2021). Muestra de saliva para diagnóstico de SARS-CoV-2 por RT-qPCR en población ambulatoria. *Ierta, Revista científica del Instituto Nacional de Salud, 4*(2), 38-45. doi:https://doi.org/10.5377/alerta.v4i2.11476

Pérez-Rosas, V. M. (2017). Understanding and predicting empathic behavior in counseling therapy. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1426-1435. doi:10.18653/v1/P17-1131

Pita Fernández, S. y. (2003). Pruebas diagnósticas. *Cad Aten Primaria, 10*(1), 120-124.

Pramono, R. X.-V. (2019). Automatic cough detection in acoustic signal using spectral features. *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 7153-7156. doi:10.1109/EMBC.2019.8857792

Pullen, M. F. (2020). Symptoms of COVID-19 outpatients in the United States. *Open forum infectious diseases, 7*(7). doi:https://doi.org/10.1093/ofid/ofaa271

Rilling, G. F. (2003). On empirical mode decomposition and its algorithms. *In IEEE-EURASIP workshop on nonlinear signal and image processing, 3*(3), 8-11.

Rincón, C. (2007). Diseño, implementación y evaluación de técnicas de identificación de emociones a través de la voz. *Universidad Politécnica de Madrid*.

Rojo, L. R. (2011). Mejoras en reconocimiento del habla basadas en mejoras en la parametrización de la voz. *Universidad Autónoma de Madrid*.

Saldanha, J. C. (2022). Data augmentation using Variational Autoencoders for improvement of respiratory disease classification. *Plos one, 17*(8), e0266467. doi:https://doi.org/10.1371/journal.pone.0266467

Sauder, C. B. (2017). Predicting voice disorder status from smoothed measures of cepstral peak prominence using Praat and Analysis of Dysphonia in Speech and Voice (ADSV). *Journal of Voice, 31*(5), 557-566. doi:https://doi.org/10.1016/j.jvoice.2017.01.006

Schuller, B. S. (2009). The INTERSPEECH 2009 Emotion Challenge. *Tenth Annual Conference of the International Speech Communication Association*, 312-315.

Schuller, B. S. (2010). The INTERSPEECH 2010 paralinguistic challenge. *INTERSPEECH*, 2794-2797.

Sharma, A. A. (2021). COVID-19: a review on the novel coronavirus disease evolution, transmission, detection, control and prevention. *Viruses, 13*(2), 202. doi:https://doi.org/10.3390/v13020202

Sharma, G. U. (2020). Trends in audio signal feature extraction methods. *Applied Acoustics, 158*, 107020. doi:https://doi.org/10.1016/j.apacoust.2019.107020

SITD, S. d. (27 de mayo de 2021). *Tos COVID-19*. BA Data: https://data.buenosaires.gob.ar/dataset/tos-covid-19

Soliński, M. Ł. (2020). Automatic cough detection based on airflow signals for portable spirometry system. *Informatics in medicine unlocked, 18*, 100313. doi:https://doi.org/10.1016/j.imu.2020.100313

Srivastava, N. H. (2014). Dropout: a simple way to prevent neural networks from overfitting. *he journal of machine learning research, 15*(1), 1929-1958.

Sun, Y. K. (2007). Cost-sensitive boosting for classification of imbalanced data. *Pattern recognition, 40*(12), 3358-3378. doi:https://doi.org/10.1016/j.patcog.2007.04.009

Tena, A. C. (2022). Automated detection of COVID-19 cough. *Biomedical Signal Processing and Control, 71*, 103175. doi:https://doi.org/10.1016/j.bspc.2021.103175

Theodorou, T. M. (2014). An overview of automatic audio segmentation. *International Journal of Information Technology and Computer Science (IJITCS), 6*(11), 1. doi:DOI: 10.5815/ijitcs.2014.11.01

V. Swarnkar, U. R. (2013). Neural network based algorithm for automatic identification of cough sounds. *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 1764-1767. doi:10.1109/EMBC.2013.6609862

Velardo, V. (26 de abril de 2021). *generating-sound-with-neural-networks*. (generating-sound-with-neural networks, Editor) Generating-sound-with-neural-networks: https://github.com/musikalkemist/generating-sound-with-neural-networks/blob/main/LICENSE

Velavan, T. P. (2020). The COVID-19 epidemic. *Tropical medicine & international health, 25*(3), 278. doi:10.1111/tmi.13383

Vijayakumar, D. S. (2021). Low cost Covid-19 preliminary diagnosis utilizing cough samples and keenly intellective deep learning approaches. *Alexandria Engineering Journal, 60*(1), 549-557. doi:https://doi.org/10.1016/j.aej.2020.09.032

Wan, X. L. (2014). Learning to improve medical decision making from imbalanced data without a priori cost. *BMC medical informatics and decision making, 14*, 1-9. doi:https://doi.org/10.1186/s12911-014-0111-9

Wei, S. Z. (2020). A comparison on data augmentation methods based on deep learning for audio classification. *ournal of Physics: Conference Series, 1453*(1), 012085. doi:doi: 10.1088/1742-6596/1453/1/012085

Weng, L. M. (2021). Pain symptoms in patients with coronavirus disease (COVID-19): A literature review. *Journal of Pain Research*, 147-159. doi:doi:10.2147/JPR.S269206

Widdicombe, J. y. (2006). Cough: what's in a name? *European Respiratory Journal, 28*(1), 10-15. doi:https://doi.org/10.1183/09031936.06.00096905

Xia, T. H. (2021). Uncertainty-aware COVID-19 detection from imbalanced sound data. *ArXiv*. doi:/abs/2104.02005

Xiao, S. L. (2018). A study of the probable transmission routes of MERS-CoV during the first hospital outbreak in the Republic of Korea. *Indoor Air, 28*(1), 51-63. doi:https://doi.org/10.1111/ina.12430

Xie, J. T. (2016). Acoustic classification of Australian frogs based on enhanced features and machine learning algorithms. *Applied Acoustics, 113*, 193-201. doi:https://doi.org/10.1016/j.apacoust.2016.06.029

Yazdani, A. S. (2021). Emotion recognition in persian speech using deep neural networks. *021 11th International Conference on Computer Engineering and Knowledge (ICCKE)*, 374-378. doi:10.1109/ICCKE54056.2021.9721504

Yue, S. L. (2003). SVM classification: Its contents and challenges. *Applied Mathematics-A Journal of Chinese Universities, 18*, 332-342. doi:https://doi.org/10.1007/s11766-003-0059-5

Zapatero Gaviria, A. y. (2023). ¿Qué sabemos del origen del COVID-19 tres años después? *Revista Clínica Española, 223*(4), 240-243. doi:https://doi.org/10.1016/j.rce.2023.02.002

Zealouk, O. S. (2021). Analysis of COVID-19 Resulting Cough Using Formants and Automatic Speech Recognition System. *Journal of Voice*. doi:https://doi.org/10.1016/j.jvoice.2021.05.015

Zhang, Y. H. (2023). An updated review of SARS-CoV-2 detection methods in the context of a novel coronavirus pandemic. *Bioengineering & Translational Medicine, 8*(1), e10356. doi:https://doi.org/10.1002/btm2.10356

Zúñiga-López, A. A.-C.-R.-M. (2020). Algoritmo de clasificación basado en la función Softmax. *Research in Computing Science, 148*(8), 95-107.