



UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE INGENIERÍA MECÁNICA

**ALGORITMOS DE APRENDIZAJE DE MÁQUINAS PARA LA PREDICCIÓN DE BAND
GAPS EN PANELES TIPO SÁNDWICH**

MEMORIA PARA OPTAR AL TÍTULO DE INGENIERO CIVIL MECÁNICO

CARLOS LUIS POBLETE CANALES

PROFESORA GUÍA:
VIVIANA MERUANE NARANJO

MIEMBROS DE LA COMISIÓN:
RAFAEL RUIZ GARCÍA
RUBÉN FERNÁNDEZ URRUTIA

Este trabajo ha sido parcialmente financiado por los proyectos Núcleo Milenio en Soft Smart
Mechanical Metamaterials y Fondecyt 1210442

SANTIAGO DE CHILE
2022

RESUMEN DE LA MEMORIA PARA OPTAR AL
TÍTULO DE INGENIERO CIVIL MECÁNICO
POR: CARLOS POBLETE CANALES
FECHA: 2022
PROF. GUÍA: VIVIANA MERUANE NARANJO

ALGORITMOS DE APRENDIZAJE DE MÁQUINAS PARA LA PREDICCIÓN DE BAND GAPS EN PANELES TIPO SÁNDWICH

Los band gaps corresponden a rangos de frecuencia para los cuales un material celular bloquea la propagación de ondas mecánicas, esta propiedad también puede ser exhibida por paneles tipo sándwich y sus características: la frecuencia media y ancho de banda, son dependientes de la arquitectura del núcleo.

El desarrollo que han tenido las técnicas de manufactura aditiva, que permiten fabricar estructuras cada vez más intrincadas, sumado a la capacidad de variar las características de una propiedad, en materiales celulares, alterando su estructura, llevan a la necesidad de desarrollar métodos eficientes para la optimización de propiedades en los materiales.

Se propone evaluar el desempeño que tienen ciertos modelos de aprendizaje de máquinas al predecir las características de band gaps en paneles tipo sándwich, a partir de los parámetros que definen la arquitectura del núcleo. Esto, debido a que estos modelos podrían reemplazar al método en métodos finitos en el proceso de optimización topológica. Reduciendo así los costos computacionales.

Se plantean dos enfoques para entrenar los modelos de aprendizajes de máquinas. Cada uno de estos es probado con los datos de tres estructuras diferentes de núcleos de paneles. Y se emplea el método de búsqueda de grilla para optimizar los parámetros de cada modelo.

Con todos los modelos se logra obtener buenas predicciones de frecuencia media y ancho de band gaps negativos. Pero ninguno logra predecir correctamente los valores de gaps positivos.

Luego de analizar los resultados, no es posible concluir que sea posible emplear técnicas de aprendizaje de máquinas para la predicción de band gaps en diferentes topologías de paneles tipo sándwich. Debido a esto se proponen alternativas que podrían ayudar a obtener mejores resultados en la predicción de band gaps positivos.

*¡Di algo o van a creer
que eres idiota!*

A lo hecho, pecho

Agradecimientos

En primer lugar, agradezco enormemente a mi familia que me han dado amor y compañía inmensurable: Mamá y Papá, ustedes me han entregado un apoyo infinito, y no puedo no dedicarles todos mis logros los cuales, probablemente, no habría podido alcanzar sin ustedes. Y Cata, erí una hermana de pana, gracias por compartir buenos momentos y carretes y por ayudarme con el uso del punto y coma.

Karen Valeska, Pincheira, Piccolis y Rodri. Han sido excelente amigos y compañeros universitarios. Conservo muy buenos recuerdos con ustedes, estando sobrios y a veces no tanto, terraceando, jugando a la caja y visitando Valhalla. Aunque quizás he sido yo el que más lo ha visitado...

Mención destaca a ti, Tito, por ser un gran amigo y tener siempre una gran disposición para explicarme cosas sobre temas computines y de memes. Me enseñaste a usar la consola de comando y me has instruidos en un sinfín de datos inservibles del internet. Y por si eso fuera poco, durante todo el periodo que trabajé en esta tesis me compartiste una cantidad increíble de videos de animalitos, sin los cuales este trabajo habría sido terminado un par de semanas antes.

Diego, Claudio Paolo y Nico. Ustedes son compas de vida. Su apañe ha sido desde fuera del entorno beauchefiano y se agradece un montón, porque quizás sin eso me hubiese vuelto loco. Las conversaciones que compartimos son siempre gratas y he aprendido mucho de ellas.

No quiero dejar de agradecerle, Profesora Viviana, por confiar en mí para realizar este trabajo y por su siempre buena disposición para aconsejar y resolver dudas.

Y, por último, voy a aprovechar estas últimas líneas para mandarle agradecimientos a mis perritas, quienes no saben nada de esto, pero se lo merecen por ser buenas compañeras en los *breaks* del telestudio: Kira, Manchita, Martina, Daga (no la quiero tanto, pero igual), Dominga, ah, y el Benito (que hace poco lo trajo la Cata).

La Mia, por ser un gato, se merece un párrafo especial: Gracias Mia.

Tabla de contenido

Capítulo 1 Introducción	1
1.1. Objetivos	2
1.1.1. Objetivo general.....	2
1.1.2. Objetivos específicos.....	2
1.2. Alcances.....	2
Capítulo 2 Antecedentes	4
2.1. Materiales celulares	4
2.1.1. Paneles tipo sándwich	5
2.1.2. Band Gaps.....	7
2.2. Aprendizaje de máquinas.....	8
2.2.1. K-Vecinos más Cercanos	9
2.2.2. Redes Neuronales Artificiales.....	10
2.2.3. Estandarización de datos	11
2.2.4. Ajustes de hiperparámetros.....	11
2.2.5. Métricas de evaluación	12
Capítulo 3 Aproximación al problema.....	13
3.1. Estructuras utilizadas	13
3.1.1. Caracterización de los datos	15
3.2. Cálculo de band gap	17

Capítulo 4 Metodología	20
4.1. Enfoques de resolución.....	20
4.2. Inspección de datos	21
4.3. Procedimiento	24
4.3.1. Preprocesamiento.....	25
4.3.2. Implementación y ajuste de modelos.....	25
4.3.3. Predicción y posprocesamiento	25
4.3.4. Evaluación	26
4.4. Modificación de la proporción de datos con band gap	27
Capítulo 5 Resultados	28
5.1. Enfoque 1	29
5.2. Enfoque 2.....	31
5.3. Variación de la proporción de muestras con band gap	33
Capítulo 6 Análisis y discusión	36
Capítulo 7 Conclusión	39
Bibliografía	41
Anexos	43
Anexo A Gráficos para la inspección de datos	43
A.1 Histogramas de valores mínimos y máximos de bandas	44
A.2 Histogramas de bandas que producen band gaps	52
A.3 Histogramas de frecuencia media y ancho de band gaps positivos.....	53
A.4 Histogramas de frecuencia media y ancho de band gaps negativos y nulos	54
Anexo B Códigos.....	55
B.1 Código enfoque 1	56

B.2	Código enfoque 2.....	61
B.3	Código variación proporción de band gaps.....	65
Anexo C Hiperparámetros seleccionados		68
C.1	Hiperparámetros del enfoque 1	68
C.2	Hiperparámetros del enfoque 2.....	69
Anexo D Gráficos de los resultados		70
D.1	Enfoque 1	71
D.2	Enfoque 2.....	77
D.3	<i>R2</i> y <i>MAE</i>	83

Capítulo 1

Introducción

Los paneles tipo sándwich son estructuras que se conforman por dos láminas delgadas separadas por un núcleo de estructura ligera. El empleo de estos materiales se hizo común luego de que en la Segunda Guerra Mundial se utilizaran en aviones de combate. Desde entonces, se ha intensificado su uso debido, principalmente, a la buena relación peso-resistencia que presentan, siendo la estructura tipo panal una de las más utilizada [1].

Se ha demostrado que las propiedades mecánicas que presenta un panel tipo sándwich son altamente dependientes de la estructura de su núcleo [2], esto ha abierto un campo de investigación en torno al estudio de distintas topologías y propiedades de los paneles, con el fin de conseguir materiales que se adapten mejor a determinadas aplicaciones.

Entre las distintas propiedades que pueden presentar los materiales celulares, los que incluyen al tipo de estructura ya mencionada, está la exhibición de band gaps, que corresponden a rangos de frecuencias para los cuales el material impide la propagación de ondas mecánicas. Esta propiedad puede ser aprovechada para el desarrollo de bloqueadores y filtros acústicos, concentradores de onda, etc.

Los avances que se han alcanzado en los últimos años en técnicas de manufactura aditiva permiten una gran flexibilidad sobre la geometría que se quiera dar al núcleo de un panel tipo sándwich, y no es errado anticipar que en un futuro próximo habrá pocas limitaciones, en términos de topología o arquitectura, para las estructuras que deseen fabricarse. Esto lleva a la necesidad de desarrollar métodos eficientes para el diseño y optimización de arquitecturas de modo que tenga un buen desempeño para una aplicación determinada.

Uno de los métodos que se ha empleado para la optimización de band gaps es la optimización topológica. Y, si bien este método ha demostrado ser efectivo en la optimización sobre diferentes topologías, se ve limitado al momento de evaluar

arquitecturas con mayores grados de libertad, debido en parte al gran costo computacional que conllevan los cálculos del método de elementos finitos. Este es un inconveniente debido a que el dominio de diseño para los materiales celulares puede tener muchos grados de libertad: materiales constituyentes, tipo de topología, tamaño de celda, grados de simetría, etc.

Una alternativa que permite reducir el costo computacional en la optimización topológica es el empleo de modelos subrogados, los cuales introducen una representación del modelo original, pero de menor dimensionalidad. Se ha demostrado que, utilizando modelos subrogados, es posible alcanzar resultados similares a los logrados con el método de optimización topológica tradicional, pero en un menor tiempo [3].

Los algoritmos de aprendizaje de máquina son una opción para estudiar como posibles modelos subrogados, debido a que permiten trabajar con problemas de regresión no lineales. Para evaluar el desempeño de estos algoritmos como modelos subrogados, es necesario primero determinar si pueden aproximarse a los modelos de elementos finitos, entregando valores de salida similares a partir de los mismos datos de entrada.

Esta memoria propone el estudio de modelos regresivos de aprendizaje de máquinas para la predicción de frecuencia media y ancho de banda en paneles tipo sándwich, a partir de los parámetros que definen la arquitectura de su núcleo.

1.1. Objetivos

1.1.1. Objetivo general

Evaluar el desempeño de diferentes algoritmos de aprendizaje de máquinas, en la predicción de la frecuencia media y ancho de band gap de distintas topologías de paneles tipo sándwich, a partir de los parámetros que definen la arquitectura de su celda unitaria.

1.1.2. Objetivos específicos

- Desarrollar métodos que permitan manipular e inspeccionar los datos y visualizar resultados de manera ágil.
- Implementar y entrenar, de manera iterativa, diferentes modelos de aprendizaje de máquinas para la obtención de las propiedades del band gaps, ajustando los hiperparámetros buscando obtener los mejores resultados de predicción posible.
- Generar predicciones con cada modelo, evaluar los resultados y comparar el desempeño de cada uno.

1.2. Alcances

En este trabajo se estudia la factibilidad de emplear modelos de aprendizaje de máquinas para la predicción de band gaps. Para lo cual se prueban distintos algoritmos ajustándolos de manera iterativa, con el fin de obtener los mejores resultados posibles en términos de precisión en la predicción.

Sin embargo, no se cuenta con una métrica que determine el nivel necesario de precisión para establecer que el modelo es suficientemente bueno para el fin esperado. Es a partir de los resultados obtenidos al final de los experimentos, que se desarrolla un análisis para determinar si es satisfactoria la precisión alcanzada por los modelos.

Por otro lado, para comenzar el trabajo, se cuenta con los datos para entrenar y evaluar los modelos de aprendizaje de máquinas los cuales fueron generados, previamente, utilizando modelos en elementos finitos desarrollados en un trabajo anterior.

La generación de estos datos por medio de los modelos en elementos finitos se encuentra fuera de los alcances de esta memoria.

Capítulo 2

Antecedentes

Para contextualizar el estudio que se lleva a cabo, es necesario presentar los aspectos teóricos y antecedentes en los que se sustenta este estudio.

La presente sección se divide en dos partes: en la primera se aborda lo relativo al objeto en estudio, es decir, los materiales celulares, paneles tipo sándwich y *band gaps*; y, en la segunda, se habla sobre el aprendizaje de máquinas y los algoritmos que se utilizan para el desarrollo experimental.

2.1. Materiales celulares

Un material celular puede definirse como aquel que se constituye por celdas unitarias que se unen a través de sus bordes y caras relleno el espacio [4]. Se pueden enumerar varios ejemplos de materiales celulares: los huesos y corcho, son ejemplos que pueden encontrarse en la naturaleza; mientras que las espumas y las estructuras tipo panel, son ejemplos de materiales celulares fabricados por el Ser Humano. En la Figura 2-1 se muestran la estructura de dichos materiales.

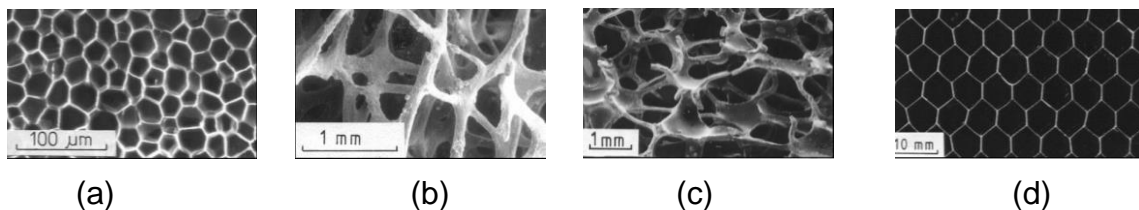


Figura 2-1 Ejemplos de materiales celulares: (a) corcho, (b) hueso trabecular, (c) espuma y (d) estructura tipo panel. Modificado de [2].

La primera característica que destaca en un material celular es la baja densidad que posee en relación con la densidad de su(s) material(es) constituyentes, esto hace que sean atractivos para aplicaciones donde sea importante minimizar el peso.

También, dada la variedad de materiales y estructuras que se pueden lograr, hay una amplia gama de aplicaciones que pueden darse a los materiales celulares tales como estructurales, absorción de energía y como aislantes térmicos, acústicos y eléctrico [2].

Las propiedades que presenta un material celular son dependientes de la estructura que éste posee y se ha demostrado que es posible emplear técnicas para seleccionar la arquitectura y materiales constituyentes que maximizan las propiedades objetivo [5].

Los avances que se han logrado en los últimos años en técnicas de manufactura aditiva permiten fabricar estructuras cada vez más intrincadas, esto ha fortalecido el concepto de materiales celulares diseñados (*Architected Cellular Materials*) [6]. Esta idea consiste en diseñar la estructura de un material celular de modo tal que se ajuste a los requerimientos de una aplicación determinada.

2.1.1. Paneles tipo sándwich

Una estructura conformada por dos láminas delgadas y resistentes, separadas por un núcleo ligero se denomina panel tipo sándwich [4]. Usualmente, las láminas son del mismo material y espesor y soportan la mayor parte de las cargas de flexión, dejando al núcleo bajo la acción de esfuerzos de corte [2]. En la Figura 2-2 se presenta un ejemplo de panel tipo sándwich.

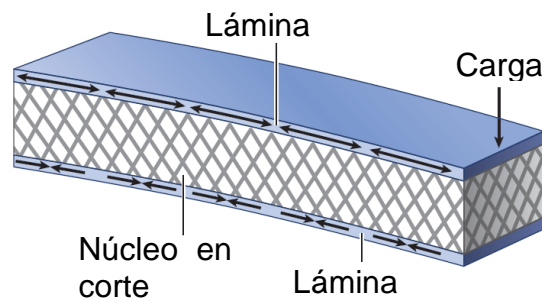


Figura 2-2 Panel tipo sándwich bajo carga [6].

Históricamente, el empleo de paneles tipo sándwich se remonta al siglo XIX, pero fue durante el siglo XX que se comenzó a potenciar el uso de este tipo de estructuras para ser empleadas en aplicaciones aeroespaciales, náutica, estructuras en trenes, etc. [1].

Típicamente, el núcleo del panel es un material celular, siendo las estructuras más usadas hasta ahora: espuma o sólido continuo, estructura tipo panel, estructura de barras (*truss-core*) y *web-core* [1]. En la Figura 2-3 se presentan esquema de estas estructuras.

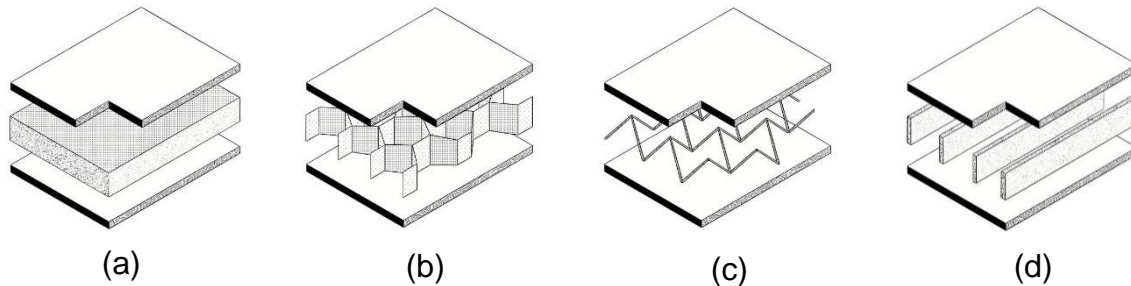


Figura 2-3 Tipos de estructuras más comunes en núcleos de paneles tipo sándwich: (a) espuma o sólido continuo, (b) estructura tipo panel, (c) estructura de barras y (d) *web-core*.

Los paneles que se estudian en esta memoria son del tipo con núcleo de estructura de barras y, además, cumplen con la característica de poseer una estructura periódica. Esto último es relevante, ya que permite imponer condiciones de borde que representan periodicidad infinita y también aprovechar las distintas simetrías en la celda unitaria, lo que reduce los costos computacionales al momento de realizar cálculos de la estructura a través del método de elementos finitos.

Si bien a esta tesis no le concierne la obtención de propiedades en estructuras a través del método de elementos finitos, es necesario comprender la idea de la simetría de la celda unitaria para entender la estructura de los datos con los que se trabaja.

En un material periódico el dominio de diseño es discretizado por un mallado de elementos finitos, donde cada pixel del mallado corresponde a una variable de diseño. Dado un mallado de $N \times N$ en una celda cuadrada, el dominio de diseño puede definirse a partir de una celda con simetría respecto al eje de 45° y los ejes X e Y, Figura 2-4 (a); una celda con simetría respecto únicamente a los ejes X e Y, Figura 2-4 (b); o una celda asimétrica, Figura 2-4 (c).

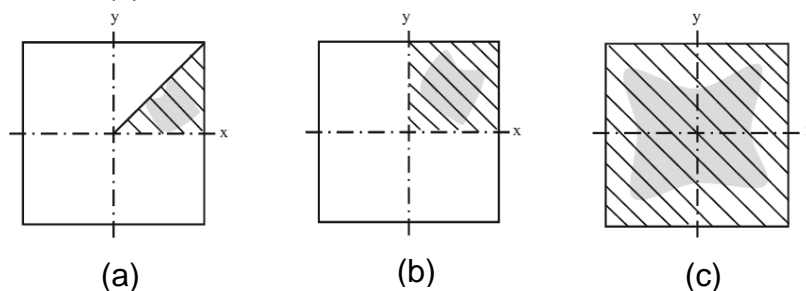


Figura 2-4 Ejemplos de condiciones de simetría: (a) dominio de diseño con $1/8$ de la celda unitaria, (b) dominio de diseño con $1/4$ de la celda unitaria y (c) dominio de diseño con la celda completa [9].

2.1.2. Band Gaps

Al exponer un material a una onda mecánica, lo normal es que esta se propague a través del material. Sin embargo, existen ciertos tipos de materiales que presentan un comportamiento atípico frente a las ondas mecánicas de ciertas frecuencias: bloquean totalmente su propagación. Los materiales que exhiben esta propiedad se denominan fonónicos.

El fonón representa un *quantum* de energía elástica vibracional y gobierna varias propiedades físicas en materiales sólidos, tales como la conductividad térmica y propagación de sonido. Se denomina band gap al rango de frecuencias para el cual la onda mecánica no puede propagarse a través del material [7].

Para obtener el band gap generado por una estructura, es necesario conocer el diagrama de bandas que éste genera. Un ejemplo de diagrama de bandas se presenta en la Figura 2-5.

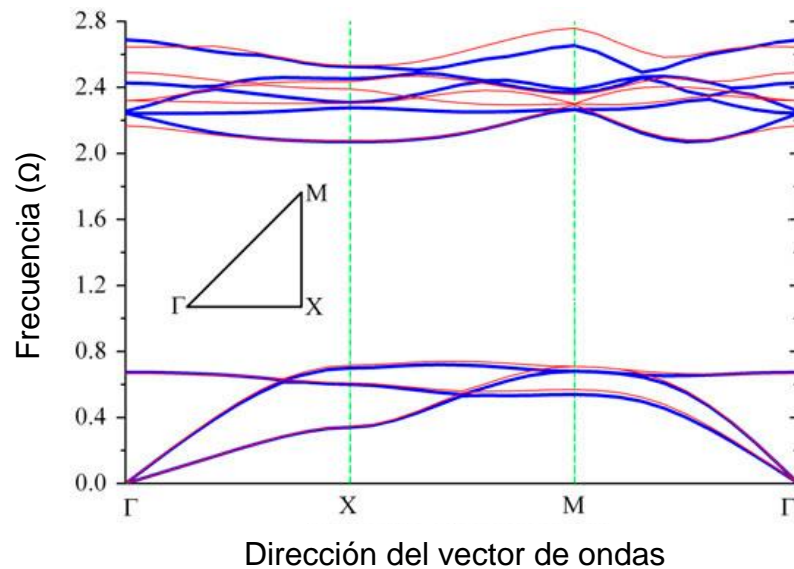


Figura 2-5 Diagrama de bandas [23].

En el diagrama se puede observar una zona no interceptada por ni una de las bandas, esta corresponde al band gap. Γ , X y M definen el perímetro de la representación en el espacio recíproco de la zona reducida de la celda unitaria, denominado también como la Zona Irreducible de Brillouin (ZIB) [8].

Al igual que otras propiedades, el band gap es dependiente de las características del material. En la Figura 2-6 se presenta el diagrama de banda de dos materiales diferentes: un material monofásico (a) y un material bifásico (b). En esta figura se puede notar que el diagrama de bandas del material bifásico presenta un *band gap* en torno a los 60 kHz.

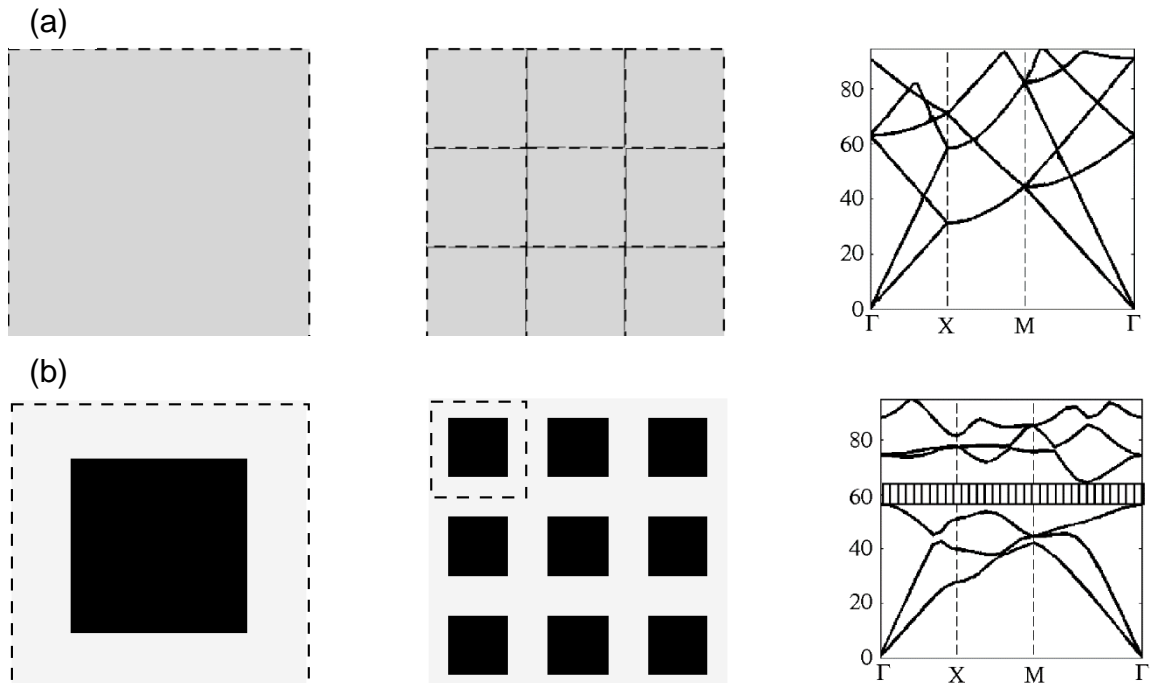


Figura 2-6 Izquierda, celda unitaria; medio, arreglo 3 x 3 de celdas; derecha diagrama de bandas de (a) Material monofásico, (b) Material bifásico [23].

Entre las aplicaciones que pueden tener las estructuras que exhiben band gaps, se encuentran guía-ondas de frecuencia selectiva, filtros y bloqueadores acústicos, y concentradores de onda. Lo importante de destacar es que, independiente de la aplicación para la que se quieran emplear, es fundamental, para su éxito, lograr que el ancho de banda sea lo más grande posible. Es de ahí que surge la necesidad de desarrollar técnicas para maximizar band gaps que puedan superar las dificultades asociadas al alto costo computacional debido a la resolución de cálculos por medio de método de elementos finitos [9].

2.2. Aprendizaje de máquinas

El aprendizaje de máquinas (ML, por sus siglas en inglés) es un área de estudio dentro del campo de la inteligencia artificial que se caracteriza por la capacidad de ciertos sistemas de aprender la solución a problemas de manera automática, sin necesidad de que una persona programe la secuencia de instrucciones para llegar a dicha solución.

Las técnicas de ML son aplicables en problemas donde se desconocen los pasos necesarios para convertir los valores de entrada en los valores de salida esperados, pero se tiene un volumen de datos suficientes para entrenar un algoritmo capaz de detectar patrones en dicha información. Entre los campos de aplicación del aprendizaje de

máquinas se pueden mencionar la clasificación de imágenes, reconocimiento de voz, detección de fraudes, etc. [10].

A grandes rasgos, los problemas de aprendizaje de máquinas pueden categorizarse en problemas de clasificación y de regresión; mientras que, entre los métodos de aprendizaje que pueden emplearse, se encuentran el aprendizaje supervisado, aprendizaje no supervisado y aprendizaje reforzado.

El enfoque de esta memoria se encuentra en un problema de regresión: a partir de un conjunto de atributos que caracterizan a la celda unitaria de un panel tipo sándwich, se espera obtener valores reales positivos, asociados a frecuencias en el diagrama de bandas.

Mientras que el método de aprendizaje que se utiliza es el entrenamiento supervisado. Esto quiere decir que se emplean conjuntos de datos de entrenamiento con los cuales, los distintos modelos implementados, se ajustan para mapear el camino desde el conjunto de entrada X al conjunto de salida Y .

Existen diferentes algoritmos para la implementación del aprendizaje de máquina, cada uno tiene diferentes hiperparámetros que son seleccionados buscando que el modelo se ajuste de la mejor manera posible a las características del problema.

Para el desarrollo de este estudio, se seleccionan dos algoritmos: K-Vecinos más Cercanos y Redes Neuronales Artificiales.

2.2.1. K-Vecinos más Cercanos

El algoritmo de los K-Vecinos más cercanos (KNN, por sus siglas en inglés), es un método de aprendizaje muy sencillo. En éste se supone que, si se tiene conjunto de objetos independientes, idénticamente distribuidos, para una distribución y categorías dadas; es probable que dos objetos cercanos posean la misma categoría o etiqueta [11].

Cuando se emplea este algoritmo, se almacena la información de los datos de entrenamientos y sus etiquetas. Luego, al ingresar un nuevo dato sin etiqueta, el algoritmo busca los k puntos más cercanos y genera la etiqueta respectiva a partir de una interpolación.

Los parámetros para escoger al momento de implementar este algoritmo son la cantidad de vecinos a considerar, la métrica de distancia y el peso que cada vecino tiene asociado al momento de ponderar los valores de cada uno.

Usualmente, la métrica de distancia que se emplea es la de Minkowski y se pueden evaluar distintos valores de potencia (p).

Por último, la interpolación puede ser configurada para que cada vecino aporte de igual manera al promedio o que haya un peso asociado inverso a la distancia del vecino con el punto que se está evaluando.

2.2.2. Redes Neuronales Artificiales

Las redes neuronales artificiales (ANN o NN, por sus siglas en inglés) son modelos bioinspirados cuyo estudio inició en la década de los cuarenta con una publicación de Warren McCulloch y Walter Pitts titulada “*A logical calculus of the ideas immanent in nervous activity*” [12].

Son diversas las arquitecturas de redes neuronales que se conocen hoy en día, cada una se adapta de mejor forma a determinados tipos de datos y/o complejidades de problemas. Sin embargo, el principio subyacente en todos los tipos de arquitecturas es formar un modelo no lineal complejo, a partir de la interconexión densa de elementos computacionales sencillos [13].

El perceptrón multicapa es el ejemplo más sencillo de NN, esta arquitectura se compone de una capa de entrada, una cantidad arbitraria de capas ocultas y una capa de salida. La unidad básica de esta arquitectura es la neurona o nodo, la cual suma las N entradas multiplicadas por los pesos asociados a cada una. Luego, el resultado de esta suma ponderada se entrega a una función no lineal. En la Figura 2-7 se muestra una representación de una neurona del perceptrón multicapa.

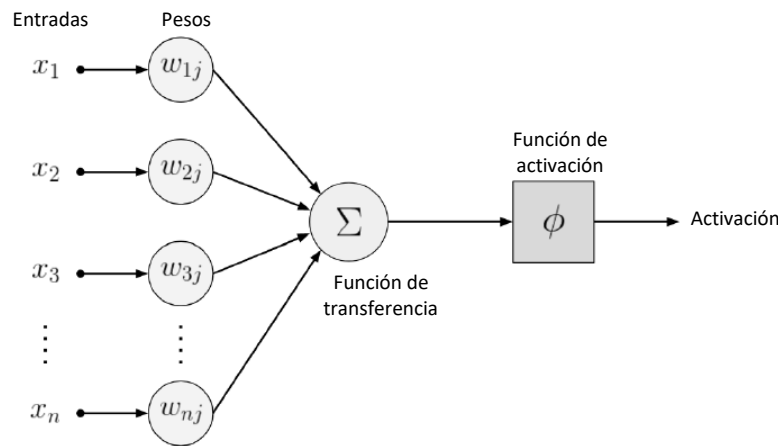


Figura 2-7 Neurona del perceptrón multicapa. Modificado de [14].

En esta arquitectura cada capa posee una cantidad determinada de neuronas, todas estas se encuentran conectadas a las neuronas de la capa siguiente, de modo que los valores de entrada de un nodo corresponden a los valores de activación de los nodos de la capa anterior, a excepción de la capa de entrada.

El entrenamiento de las redes neuronales consiste en ajustar los valores de los pesos sinápticos de cada neurona, para esto se llevan a cabo dos etapas iterativas: *Forward Propagation* y *Back Propagation*.

Durante el *Forward Propagation*, se entrega un valor de entrada a la red y se evalúa su salida, a partir de esto se calcula el error de la evaluación respecto a un valor esperado. Luego, con *Back Propagation*, se distribuye la contribución del error por cada uno de los

pesos y estos se actualizan de modo que se reduzca el valor de evaluación en la próxima iteración [14].

Los parámetros que definen la arquitectura de una red neuronal son la cantidad de capas, el número de nodos por capas, y las funciones de activación. Por otro lado, los hiperparámetros son valores que definen el modo en que se entrena la red, entre estos se encuentran la tasa de aprendizaje, el método de optimización y la función de pérdida.

En el caso de un problema de regresión, para garantizar que el valor obtenido en la predicción sea positivo, puede utilizarse la función de activación ReLU en la capa de salida. Y usualmente, la función de pérdida empleada para el entrenamiento es el error cuadrático medio o el error absoluto medio [15].

2.2.3. Estandarización de datos

Es común que los datos con los que se entrenan los modelos de aprendizaje de máquinas posean distribuciones y ordenes de magnitud diferentes entre las diferentes variables. Un ejemplo sencillo de esto es el caso de un modelo que prediga el precio de una casa a partir de los metros cuadrados del terreno y cantidad de baños.

Se puede anticipar que la variable “metros cuadrados” tenga valores que van de las decenas a los miles de metros cuadrados, distribuidos de una manera más o menos uniforme. Mientras que la segunda variable podría tener la mayor cantidad de datos concentrados en 1 y 2, y a medida que crece este valor, disminuye la cantidad de muestras que las presentan.

Eso puede generar problemas en el entrenamiento debido a la posibilidad de que los algoritmos tiendan a reconocer mejor a las variables con mayores órdenes de magnitud, y también puede haber distribuciones más fáciles de aprender que otras.

La estandarización es una práctica común de preprocesamiento que tiene por objetivo hacer que todas las variables de los datos se distribuyan según una normal estándar. De esta forma disminuye el efecto causado por la disimilitud entre las diferentes variables.

Los valores estandarizados se obtienen a partir de la siguiente ecuación:

$$z = \frac{x - \mu}{\sigma} \quad 2-1$$

Donde μ es la media y σ la desviación estándar de las variables de entrenamiento [16].

2.2.4. Ajustes de hiperparámetros

Como ya se ha mencionado los hiperparámetros son valores que se entregan al modelo como argumento. Y estos no se ajustan de manera automática por lo que parte del proceso de implementar modelos de aprendizaje de máquinas, es buscar las combinaciones de hiperparámetros que generan los mejores resultados posibles.

Debido a que realizar esta tarea de manera manual puede resultar tedioso y aún así no alcanzar los mejores resultados, es común implementar búsquedas automáticas de las combinaciones de hiperparámetros.

Existen dos aproximaciones para realizar esto: La búsqueda exhaustiva, o de grilla, y la búsqueda aleatoria. En la primera se prueban todas las combinaciones posibles dentro de un espacio de búsqueda predefinido. Mientras que la segunda toma una cantidad de combinaciones generadas automáticamente dentro de dicho espacio [17].

2.2.5. Métricas de evaluación

Para evaluar el desempeño que tiene un modelo de aprendizaje de máquinas al predecir valores, es necesario contar con métricas que indiquen la cercanía de los valores predichos con los valores reales.

Entre las métricas usadas para los modelos de regresión, son dos las que se toman en esta memoria: el coeficiente de determinación, R^2 , y el error absoluto medio, MAE, por sus siglas en inglés. Que se definen con las siguientes ecuaciones.

$$R^2(y, \hat{y}) = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad 2-2$$

$$MAE(y, \hat{y}) = \frac{1}{n_{samples}} \sum_{i=0}^{n-1} |y_i - \hat{y}_i| \quad 2-3$$

y e \hat{y} corresponden a los arreglos que contienen los valores reales y predichos, respectivamente; y_i se refiere al valor real asociado a la i -ésima muestra (etiqueta) mientras que \hat{y}_i es la predicción para la misma. Finalmente, n es la cantidad de muestras empleadas para la evaluación del modelo [18].

Capítulo 3

Aproximación al problema

En este capítulo se describen aspectos necesarios para comprender el problema que se busca resolver. Se comienza describiendo las estructuras estudiadas y se dan las características de los datos empleados. Luego, se explica el método para calcular los band gaps a partir de la información del diagrama de bandas.

3.1. Estructuras utilizadas

Los diferentes modelos de aprendizaje de máquinas son entrenados y evaluados utilizando los datos de tres estructuras de paneles tipo sándwich, obtenidos a partir los modelos en elementos finitos desarrollados por V. Gálvez [19].

Las tres estructuras estudiadas en la memoria de Gálvez poseen núcleo de barra y fueron modeladas utilizando elementos tipo *shell* para las placas y elementos tipo *beam*, de dos nodos, para las barras.

Las diferentes topologías fueron diseñadas definiendo una celda unitaria cuadrada para el panel tipo sándwich, la cual se subdividió en celdas básicas. Y luego, imponiendo condiciones de simetría en la celda unitaria, se determinó la Zona Irreducible de Brillouin.

Se modelaron las topologías base para cada estructura considerando celdas unitarias con elementos de barra de área transversal, altura y material definidas, y estableciendo factores de diseño que modificarán el valor del radio de cada elemento de barra en la ZIB y el largo de la celda unitaria. De este modo, se tiene que los valores de entrada de los modelos en elementos finitos corresponden a los factores que modifican la geometría de la celda unitaria.

Las estructuras utilizadas en esta memoria se denominan pirámide de base cuadrada, kagome de base cuadrada y pirámide central. En adelante llamadas estructura 1, estructura 2 y estructura 3, respectivamente. En la Figura 3-1, Figura 3-2 y Figura 3-3 se muestran los esquemas de la celda unitaria y ZIB de cada una; los colores que se muestran en los esquemas de la ZIB indican el diámetro que posee cada elemento, los cuales son modificados por medio de los factores de radio.

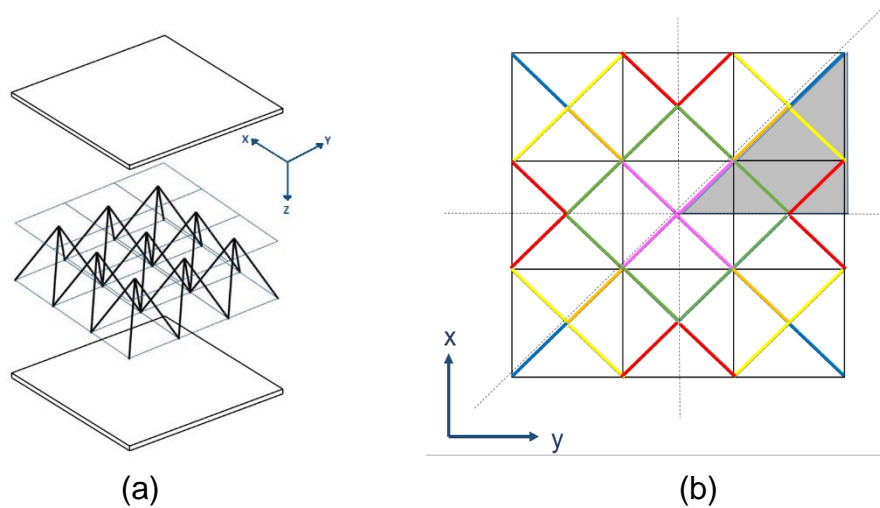


Figura 3-1 (a) esquema de la estructura 1 y (b) simetrías y ZIB de la misma.

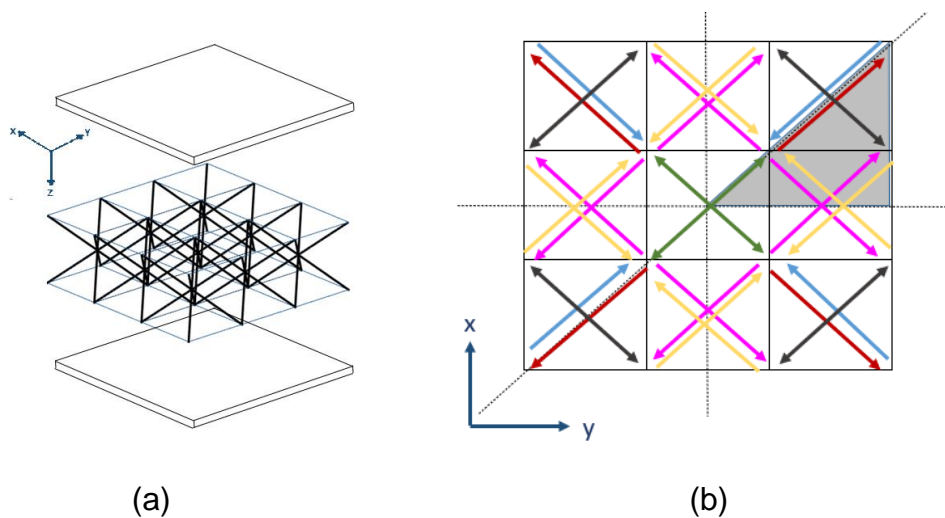


Figura 3-2 (a) esquema de la estructura 2 y (b) simetrías y ZIB de la misma.

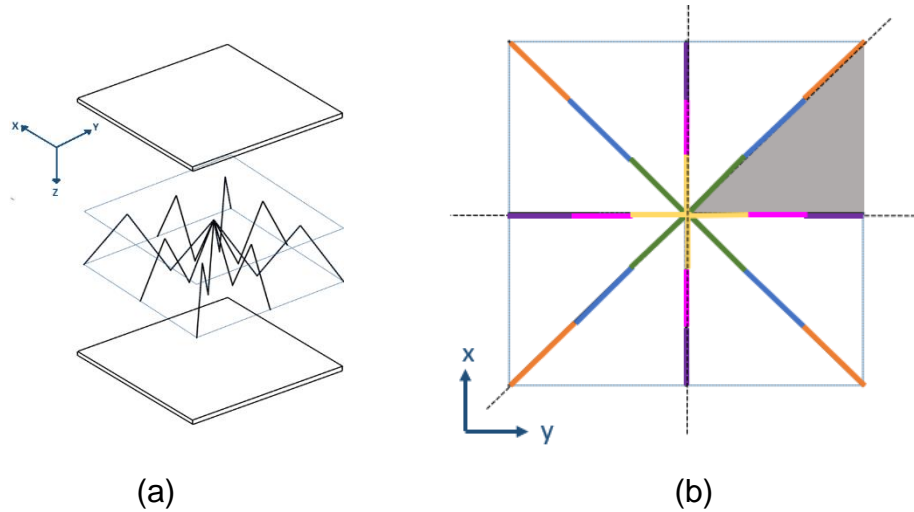


Figura 3-3 (a) esquema de la estructura 3 y (b) simetrías y ZIB de la misma.

En estas figuras se puede notar que las tres estructuras poseen seis factores que modifican los radios de los elementos de barra, por lo que, en total, considerando el factor que modifica el largo de la celda unitaria, se tienen siete factores que determinan la geometría de cada estructura de panel tipo sándwich.

El resultado que se obtiene con cada modelo en elementos finitos, ingresando diferentes combinaciones de los siete factores, corresponde a las primeras bandas del diagrama de bandas de la estructura.

3.1.1. Caracterización de los datos

Como ya se ha mencionado, para cada una de las estructuras se cuenta con un par de conjuntos de datos correspondiente a los datos de entrada y salida del modelo. En adelante, a estos conjuntos se les denomina \bar{X} e \bar{Y} , respectivamente.

\bar{X} es un arreglo bidimensional de $n_{samples}$ filas y $n_{attributes}$ columnas. Este conjunto de datos se generó utilizando el método de muestreo de hipercubo latino, tomando puntos dentro de espacio definido en el diseño de los modelos en elementos finitos. En el caso de la estructura 1, el conjunto \bar{X} posee valores reales entre 0,2 y 5, mientras que en la estructura 2 y 3 los valores están entre 0 y 1. Estos valores corresponden a los factores que modifican la geometría de las estructuras.

Por otro lado, \bar{Y} es un arreglo tridimensional que posee $n_{samples}$ arreglos bidimensionales, cada uno de ellos con n_{bands} filas y $n_{features}$ columnas. Este conjunto es generado evaluando todos los puntos de \bar{X} en los modelos en elementos finitos correspondientes a cada estructura.

Cada uno de los arreglos bidimensionales almacenados en \bar{Y} corresponde al diagrama de bandas generado por el punto en \bar{X} correspondiente, por lo que los valores que se hallan en dicho arreglo son valores reales no-negativos que representa frecuencias en Hertz. n_{bands} representa la cantidad de bandas que se toman del diagrama y $n_{features}$ es la cantidad de puntos con el que se discretiza el vector de ondas.

En la Figura 3-4 se puede ver un esquema representativo de los conjuntos de datos de entrada y salida. Y en la Tabla 3-1 se entregan los valores que toman $n_{samples}$, $n_{attributes}$, n_{bands} y $n_{features}$. Se debe destacar que para los conjuntos de las tres estructuras utilizadas en esta memoria se tienen los mismos valores.

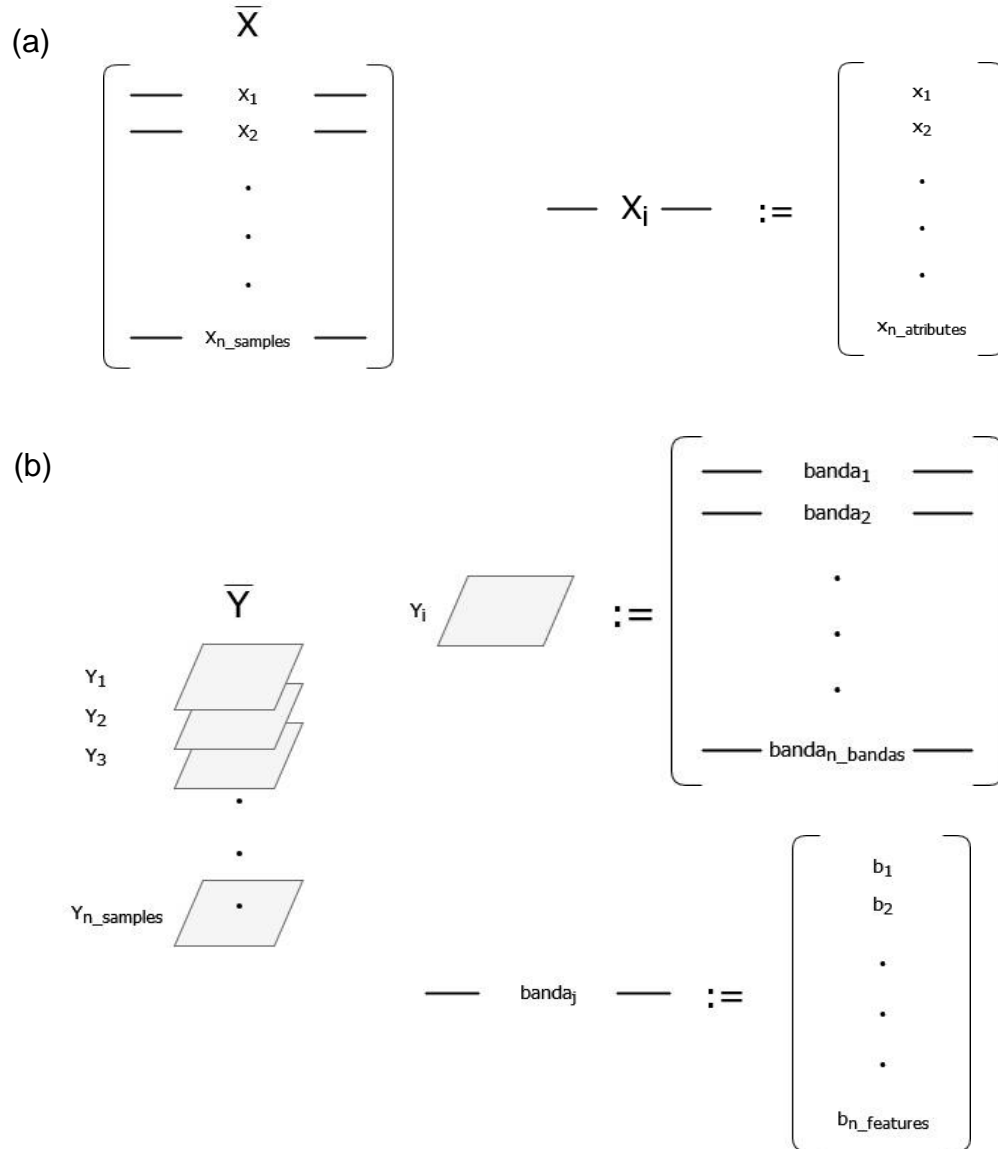


Figura 3-4 Estructura de los datos generados por modelos en elementos finitos. (a) conjunto de entrada y (b) conjunto de salida.

Tabla 3-1 Dimensiones de los conjuntos de entrada y salida.

$n_{samples}$	20.000
$n_{attributes}$	7
n_{bands}	8
$n_{features}$	28

3.2. Cálculo de band gap

El band gap generado por una estructura, puede calcularse a partir de los datos que se tienen en el conjunto \bar{Y} . Para explicar el algoritmo con el que se obtiene el band gap de cada muestra, en la Figura 3-5 se presenta el diagrama de banda asociado a uno de los puntos pertenecientes al conjunto de la estructura 3.

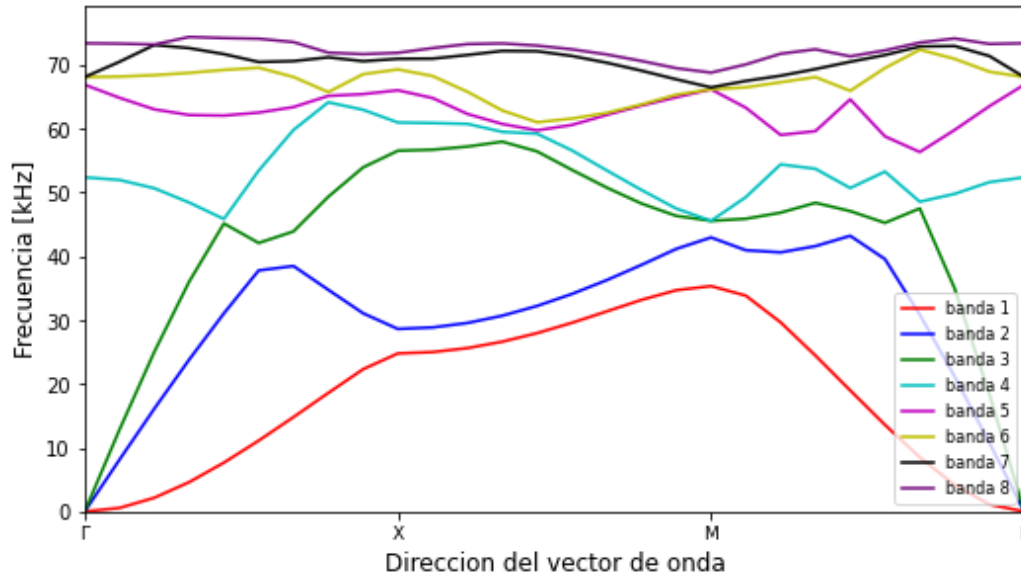


Figura 3-5 Diagrama de bandas de muestra de la estructura 3.

Son dos supuestos los que se consideran para diseñar el algoritmo. El primero es que el band gap puede encontrarse entre cualquier par de bandas sucesivas y el segundo es que, como se puede notar en el diagrama presentado, las bandas se encuentran ordenada de manera ascendente, es decir, $Max(banda_i) \leq Max(banda_{i+1})$.

Tomando dichos supuestos, sean $banda_i$ y $banda_{i+1}$ dos bandas sucesivas entre las que hay un band gap, entonces, los valores de la frecuencia media y ancho de banda de dicho band gap están dados por:

$$freq = \frac{Min(banda_{i+1}) + Max(banda_i)}{2} \quad 3-1$$

$$gap = Min(banda_{i+1}) - Max(banda_i) \quad 3-2$$

Puede suceder que una muestra presente más de un band gap. En ese caso se considera únicamente el band gap más ancho.

A continuación, se explica el seudocódigo con el que se calcula el band gap de cada muestra. Para facilitar la comprensión del procedimiento, se explica cómo calcular el band gap de una sola muestra, sin embargo, esto puede realizarse de manera simultánea para todas las muestras, aprovechando la librería NumPy. En la Figura 3-6 se presentan esquemas que acompañan la explicación del pseudocódigo.

Algoritmo 1: Cálculo de Band Gap.

Input: Arreglo de $(n_{bands}, n_{features})$, perteneciente a \bar{Y} .

Output: Propiedades del band gap.

1. Obtener los valores mínimos y máximos de cada banda.
 2. Restar la frecuencia mínima de la $banda_{i+1}$ con la máxima de la $banda_i$ ¹.
 3. Seleccionar el máximo gap y guardar el valor i para ubicar la banda inferior.
 4. Calcular la frecuencia media.
-

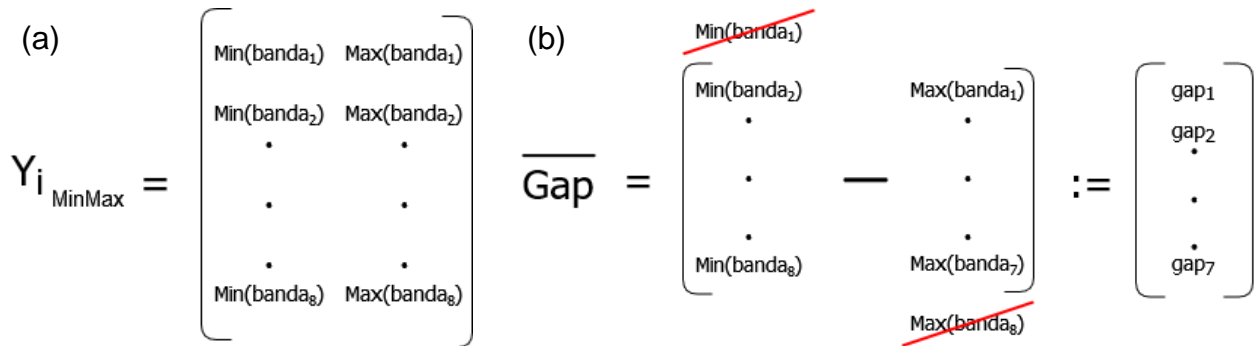


Figura 3-6 Diagramas cálculo de band gaps (a) paso 1 y (b) paso 2.

De acuerdo con el seudocódigo y lo mostrado en la Figura 3-6, el gap y frecuencia media se obtienen con las siguientes ecuaciones:

$$gap := gap_i = \text{Max}(\overline{Gap}): gap_i \geq threshold \quad 3-3$$

$$freq = \frac{\text{Max}(banda_i) + \text{Min}(banda_{i+1})}{2} \quad 3-4$$

¹ Se puede notar que la frecuencia mínima de la primera banda y la frecuencia máxima de la última banda son descartadas. Esto debido a que dichos valores nunca influyen en el cálculo de los band gaps.

threshold es un valor entero positivo y arbitrario utilizado para descartar band gaps con anchos de banda cercanos a cero. En el caso de que no se cumpla la condición del umbral, tanto la frecuencia media como el ancho de banda se vuelven cero.

Capítulo 4

Metodología

Todo el trabajo es desarrollado usando el lenguaje de programación Python en su versión 3.8.8 y el código es escrito utilizando Spyder 5.1.5.

Las principales librerías empleadas para el desarrollo de los experimentos son NumPy 1.21.2, para la manipulación de los datos; scikit-learn 1.0.1, para la implementación de modelos de aprendizaje de máquinas, técnicas de preprocesamiento y métricas de evaluación; y, por último, Matplotlib para generar todos los gráficos y figuras necesarios para la inspección de datos y visualización de resultados.

Como ya se mencionó, para el desarrollo de los experimentos se cuenta con los tres pares de conjuntos de datos, \bar{X} e \bar{Y} . En las próximas secciones se describen los procedimientos llevados a cabo para lograr la implementación de los diferentes modelos de aprendizaje de máquinas, la obtención de resultados y posterior análisis de los mismo.

4.1. Enfoques de resolución

Se busca evaluar el desempeño de modelos de aprendizaje de máquinas en la obtención de las propiedades del band gap, a partir de los parámetros que definen la arquitectura de la celda unitaria.

Teniendo esto presente, se plantean dos alternativas para obtener la frecuencia media y ancho de band gap empleando modelos de aprendizaje de máquinas. La primera consiste en entrenar modelos que predigan los valores mínimos y máximos de cada banda.

Según se presentó en la sección 3.2, si se conocen los valores mínimos y máximos de todas las bandas de una muestra, es posible obtener el band gap que esta genera. Por lo que, si se entrenan modelos capaces de predecir los valores mínimos y máximos de cada banda, luego puede utilizarse un método que calcule el band gap generado por la muestra a partir de los valores predichos.

Por otro lado, la segunda alternativa consiste en entrenar modelos que predigan directamente las propiedades del band gap a partir de los parámetros de la geometría de la celda.

Una consideración que se toma para la evaluación de estos enfoques consiste en aceptar como válidos los band gaps con gaps negativos. Es decir que el ancho de banda calculado a partir del diagrama de bandas puede ser un valor positivo, nulo o negativo².

La razón para esta consideración se encuentra en que al momento de realizar un proceso de optimización topológica de band gap entorno a una frecuencia media determinada, el algoritmo, inicialmente, podría encontrarse en un punto con gap negativo. Por lo que con cada iteración se iría reduciendo este gap negativo, acercándose a cero, hasta eventualmente alcanzar los gaps positivos.

4.2. Inspección de datos

Para facilitar la optimización de los hiperparámetros de los modelos de aprendizaje de máquinas y el análisis de los resultados, es útil conocer características básicas de los datos con los que se está trabajando.

Debido a que se sabe que los datos en los conjuntos de entrada son valores reales generados por el método de muestreo de hipercubo latino, la inspección de datos se enfoca únicamente en los conjuntos de salida para los tres enfoques.

En primer lugar, es relevante saber el número de datos que generan band gaps positivos, negativos y nulos en cada estructura. Esto se muestra en la Tabla 4-1. Se destaca la poca cantidad de datos con band gap positivos: menor al 2% para todas las estructuras.

Tabla 4-1 Cantidad de band gaps positivos, nulos y negativos en los conjuntos de datos de cada estructura.

	Estructura 1	Estructura 2	Estructura 3
B.G. Positivo	248	319	386
B.G. Nulo	397	52	51
B.G. Negativo	19.355	19.629	19.563
Total	20.000	20.000	20.000

² Un band gap con ancho de banda negativo, es realidad un band gap nulo (frecuencia media y ancho de banda iguales a cero). Sin embargo, al reemplazar en la ecuación 3-3 la condición de umbral $gap_i \geq threshold$ por $|gap_i| \geq threshold$ se obtienen los band gaps de frecuencia media positiva y ancho de banda negativo utilizados en el enfoque 3.

Para el enfoque 1, es útil saber cómo se distribuyen los valores mínimos y máximos de cada banda. En la Figura 4-1 se presentan dos histogramas que muestran la distribución de los valores mínimos (histograma superior) y máximos (histograma inferior) de la banda 4 en la estructura 1. En el Anexo A.1 se hallan los histogramas de valores mínimos y máximos de todas las bandas de las tres estructuras.

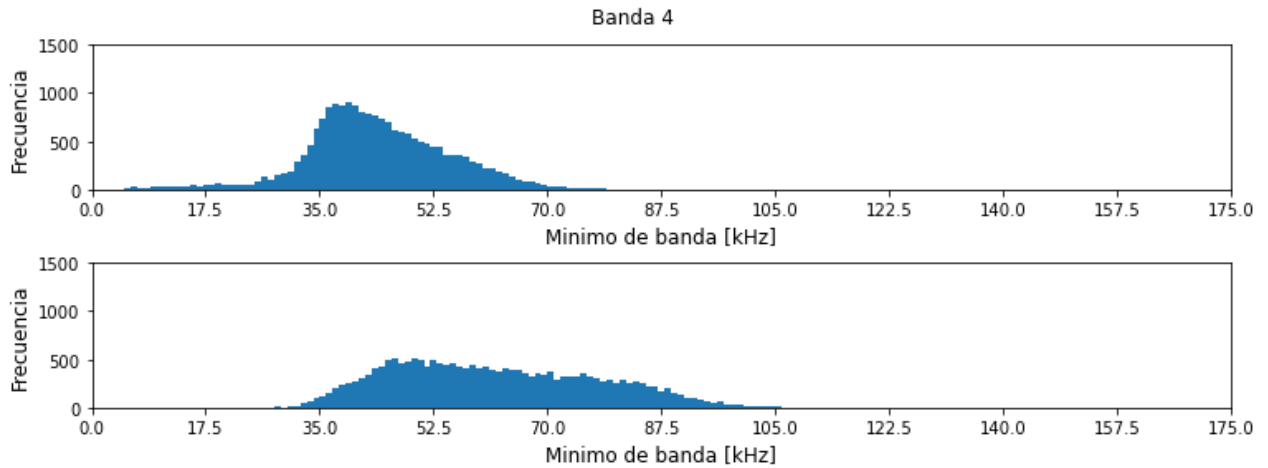


Figura 4-1 Histograma de valores mínimos y máximos de la banda 4, estructura 1.

En general se puede apreciar que en todos los casos los valores máximos están más uniformemente distribuidos que los valores mínimos. Esta diferencia es mayor en las bandas de baja frecuencia y sobre todo en la estructura 1.

En la Tabla 4-2 se entrega el promedio de los valores mínimos y máximos de cada una de las bandas de las estructuras 1, 2 y 3.

Tabla 4-2 Promedio de valores mínimos y máximos de bandas.

	Estructura 1		Estructura 2		Estructura 3	
	$\mu_{mín}$ [Hz]	$\mu_{máx}$ [Hz]	$\mu_{mín}$ [Hz]	$\mu_{máx}$ [Hz]	$\mu_{mín}$ [Hz]	$\mu_{máx}$ [Hz]
Banda 1	85	37.844	0	40.607	2	31.313
Banda 2	2.844	44.872	27	45.741	51	43.495
Banda 3	5.498	57.079	115	56.543	1.050	58.215
Banda 4	43.712	61.907	47.070	61.381	44.416	66.082
Banda 5	52.986	65.166	54.789	65.530	55.812	77.372
Banda 6	58.898	68.176	57.647	68.849	64.194	79.348
Banda 7	63.101	70.830	62.380	70.676	69.007	86.329
Banda 8	65.358	72.827	64.250	73.243	76.368	88.765

Otro dato interesante de analizar es entre cuáles de las ocho bandas se pueden generar band gaps. En la Figura 4-2 se presenta el histograma de bandas que producen band gaps positivos en la estructura 1. En el Anexo A.2 se presentan los histogramas correspondientes a cada estructura.

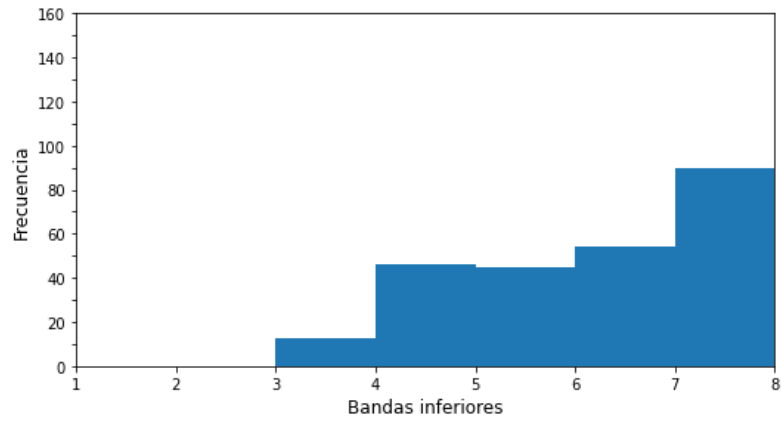


Figura 4-2 Histograma de bandas que producen band gap positivo en la estructura 1.

Se destaca que, en las tres estructuras, las primeras bandas no producen band gaps. Esto es relevante para el enfoque 1 debido a que, si se sabe que hay bandas irrelevantes para el cálculo de band gaps, estas pueden ser descartadas previamente, reduciendo así la dimensionalidad del vector de salida del modelo.

Por último, se cuenta con los histogramas de frecuencia media y ancho de banda de band gaps positivos (Figura 4-3 (a)) y band gaps no-positivos (Figura 4-3 (b)). En los anexos A.3 y A.4 se encuentran estos histogramas para las tres estructuras.

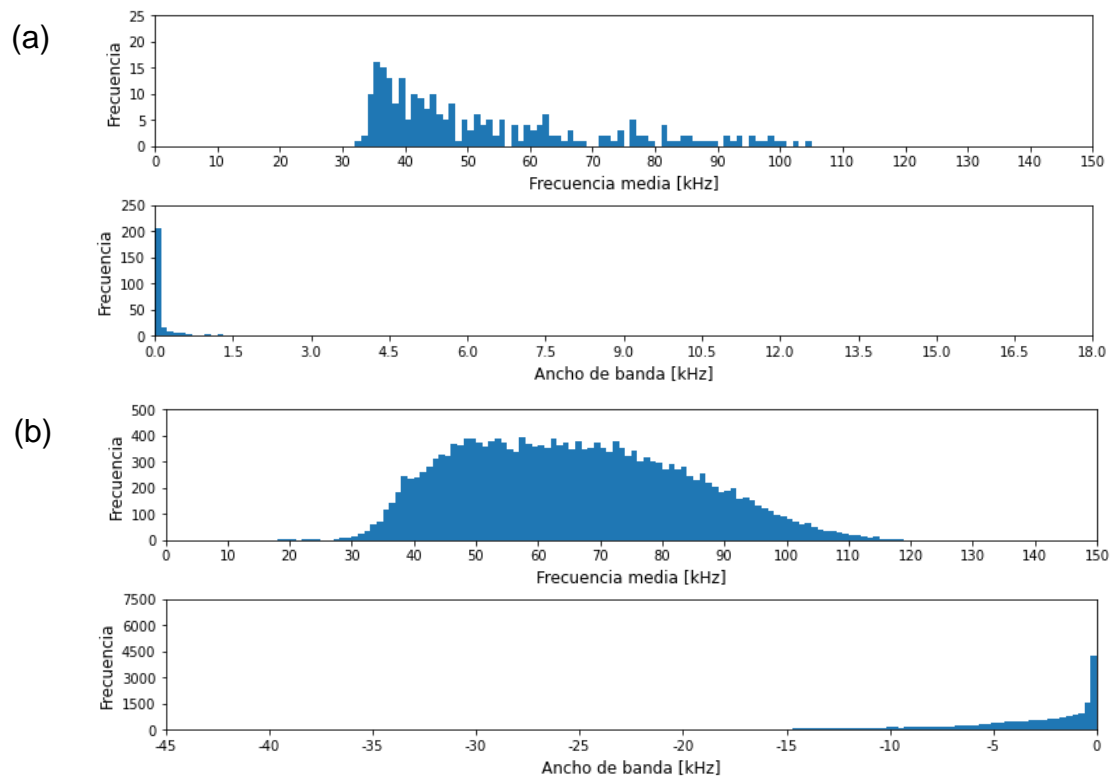


Figura 4-3 Histogramas de band gaps de la estructura 1. (a) band gaps positivos y (b) band gaps no-positivos.

Respecto a los band gaps positivos se puede observar que, para las tres estructuras, los valores de ancho de band gap se encuentran bastante desbalanceados y, además, se tiene una gran concentración de puntos en torno a cero. Esta particularidad, nuevamente, se ve acrecentada en la estructura 1.

Por otra parte, los band gaps negativos y nulos tiene una distribución más cercana a la uniforme. Sobre todo, los valores de frecuencia media.

En la Tabla 4-3 se entrega el valor promedio de la frecuencia media y ancho de banda de band gaps positivos para las tres estructuras. En la Tabla 4-4 se entregan los mismos valores para los band gaps no-positivos.

Tabla 4-3 Promedio de la frecuencia media y ancho de band gaps positivos.

	Estructura 1	Estructura 2	Estructura 3
	μ [Hz]	μ [Hz]	μ [Hz]
freq [Hz]	52.860	59.110	86.740
gap [Hz]	134	300	1.140

Tabla 4-4 Promedio de la frecuencia media y ancho de band gaps no-positivos.

	Estructura 1	Estructura 2	Estructura 3
	μ [Hz]	μ [Hz]	μ [Hz]
freq [Hz]	65.610	62.050	69.820
gap [Hz]	-3.850	-4.510	-6.300

4.3. Procedimiento

Inicialmente, los datos crudos vienen almacenados en archivos *.mat. El primer paso es importar dichos archivos para poder trabajar los datos en Python con arreglos NumPy. Durante el proceso de importación, los datos pertenecientes a \bar{X} se conservan de la forma en que vienen, mientras que, por comodidad y dado el orden de magnitud de las frecuencias con las que se trabaja, los valores que vienen almacenados en \bar{Y} son redondeados a la unidad, de modo que todos los valores de frecuencia de las bandas son números enteros positivos o cero.

Luego de la importación de datos, el desarrollo experimental para la memoria se divide en cinco etapas. Estas son: Preprocesamiento, implementación y ajuste de modelos, predicción y posprocesamiento y, por último, evaluación.

Existen leves variaciones en cada etapa dependiendo del enfoque y algoritmo utilizado. La explicación de cada etapa se entrega en las siguientes secciones y en el Anexo B.1 y B.2 se entregan los códigos implementados para uno de los modelos del enfoque 1 y 2, respectivamente.

4.3.1. Preprocesamiento

El preprocesamiento consiste en el tratamiento de los conjuntos de datos previo a su uso para el entrenamiento y evaluación de los modelos de aprendizaje de máquinas.

El primer paso del preprocesamiento, para el enfoque 1, consiste en obtener los valores mínimos y máximos de cada banda y, luego, eliminar los valores de las bandas que no influyen en el cálculo de band gaps.

Por otro lado, para el enfoque 2, se comienza obteniendo los valores de frecuencia media y ancho de band gap de todas las muestras.

Luego, se realiza la división de los conjuntos de datos en los subconjuntos de entrenamiento y evaluación. Se destina el 90% de los datos para el entrenamiento de todos los modelos implementados.

Además, debido a que en el enfoque 1 son predichos los diagramas de banda, la división de conjuntos en este enfoque se realiza dejando todas las muestras que generan band gap en el subconjunto de evaluación. Mientras que en el enfoque 2, la división se realiza manteniendo, en ambos subconjuntos, la misma proporción de muestras que generan band gaps positivos, respecto a los conjuntos originales.

Por último, los datos de salida son estandarizados utilizando el método StandardScaler de scikit-learn.

4.3.2. Implementación y ajuste de modelos

Inicialmente se implementa un modelo de aprendizaje de máquinas fijando sus hiperparámetros en valores arbitrarios. Luego, se realizan, de manera iterativa, búsquedas de grilla para ajustar los hiperparámetros con el objetivo de optimizar los resultados.

Para KNN, los hiperparámetros que se ajustan son '*n_neighbors*', '*weights*' y '*p*'. Mientras que para RNN son '*hidden_layer_sizes*', '*activation*', '*solver*', '*learning_rate*', '*learning_rate_init*' y '*tol*'. Sin contar '*max_iter*', '*early_stopping*' y '*n_iter_no_change*', para RNN, el resto de los hiperparámetros fueron dejados en sus valores por defecto [20] [21].

Con el fin de obtener resultados replicables, durante el proceso de ajuste de los modelos se establece una semilla *random*.

4.3.3. Predicción y posprocesamiento

Antes de evaluar el desempeño de los modelos en las predicciones realizadas. Es necesario realizar un tratamiento a los datos predichos. Éste consiste, básicamente, en descartar las predicciones que no cumplen con ciertas condiciones.

En el enfoque 1, lo esperado para los valores predichos es que cumplan las condiciones:

$$Max(banda_i) \leq Max(banda_{i+1}) \quad 4-1$$

$$Min(banda_i) \leq Max(banda_i) \quad 4-2$$

Y en el enfoque 2, las condiciones que deben cumplir los valores predichos son:

$$freq > 0 \quad 4-3$$

$$|gap| \geq threshold \quad 4-4$$

En todos los casos, los valores de las muestras predichas que no cumplen con dichas condiciones, se vuelven cero. Resultando en band gaps nulos.

4.3.4. Evaluación

Para evaluar el desempeño de los modelos implementados se realizan diez ciclos de entrenamiento y predicción. En cada ciclo se registran las métricas de evaluación de la predicción, R^2 y MAE , según corresponda, y también se mide el tiempo de ciclo. Luego de los diez ciclos, estos valores son promediados.

El error absoluto medio es empleado en el enfoque 1 para medir la distancia entre valores mínimo y máximos predicción y los valores reales esperados. Mientras que el coeficiente de determinación es usado tanto en el enfoque 1 como en el 2 para medir la calidad de la predicción de los band gaps.

Con el fin de poder analizar de mejor manera la forma en que se comporta la predicción de los band gaps, se evalúa diferentes valores correspondientes al coeficiente de determinación. Estos son:

- R^2_{freq} : Corresponde al coeficiente de determinación entre los valores de frecuencia media reales y predichas. Este valor también se segrega en $R^2_{freq-pos}$ y $R^2_{freq-neg}$ con el objetivo de analizar la predicción de band gaps positivos y no-positivos de manera independiente.
- R^2_{gap} : Corresponde al coeficiente de determinación entre los valores de banda reales y predichos. Al igual que el coeficiente anterior, éste se segrega en $R^2_{gap-pos}$ y $R^2_{gap-neg}$.
- $R^2_{bandgap}$: Este valor se toma como el coeficiente de determinación global de la predicción de band gaps. Corresponde al promedio simple entre R^2_{freq} y R^2_{gap} , y es utilizado para ordenar, según desempeño, los diferentes modelos implementados.

Además de estas métricas se elaboran gráficos de dispersión y matrices de confusión. El primero para constatar visualmente la calidad de la predicción realizada comparándola con los valores reales esperados. Mientras que la segunda sirve para conocer la cantidad de datos que fueron correctamente identificados como band gap positivo o no-positivo.

4.4. Modificación de la proporción de datos con band gap

Debido al bajo porcentaje de muestras que presentan band gaps, se propone como último experimento evaluar cómo influye la proporción de muestras con band gaps en los resultados obtenidos.

Para esto se selecciona el modelo con mejores resultados de acuerdo con el $R^2_{bandgap}$ obtenido. Y se realizan con éste secuencias sucesivas de entrenamiento y predicción, disminuyendo entre cada secuencia quinientas muestras sin band gaps positivos asociados.

En cada secuencia se registran los coeficientes de determinación obtenidos, siguiendo la misma segregación indicada en la sección 4.3.4. De este modo se obtiene un conjunto de valores de R^2 asociados a cada proporción de muestras con band gaps evaluada.

Este procedimiento se realiza cinco veces para obtener un valor medio para cada coeficiente de determinación.

En el Anexo B.3 se adjunta el código implementado para este experimento.

Capítulo 5

Resultados

A continuación, se presentan los resultados obtenidos en este estudio. Todos los resultados son generados con los distintos modelos, luego de los hiperparámetros de estos fuesen optimizados por el método de búsqueda de grilla. Para conocer los hiperparámetros de cada modelo, revisar el Anexo C.

Este capítulo se divide en tres secciones. En las secciones 5.1 y 5.2 se entregan los mejores resultados obtenidos con los enfoques 1 y 2, respectivamente. Estos son presentados con gráficos de dispersión para la predicción de valores mínimo y máximos de las bandas correspondientes y del cálculo de frecuencia media y ancho de band gap. También se entregan los valores del coeficiente de determinación y errores cuadráticos medios obtenidos. Y por último matrices de confusión sobre las etiquetas band gaps positivo y no-positivo.

En la sección 5.3 se entregan los resultados correspondientes a la modificación del porcentaje de muestras con band gaps para el entrenamiento del mejor modelo implementado.

Los resultados obtenidos con los dos enfoques para las tres estructuras y empleando RNN y KNN se encuentran en el Anexo D.

5.1. Enfoque 1

Los mejores resultados para el enfoque 1 son obtenidos con el modelo KNN implementado para la estructura 2. El tiempo de entrenamiento de éste fue de 2 segundos.

En la Figura 5-1 se presentan los gráficos de dispersión de las predicciones de valores mínimo y máximos de las bandas para la estructura 2.

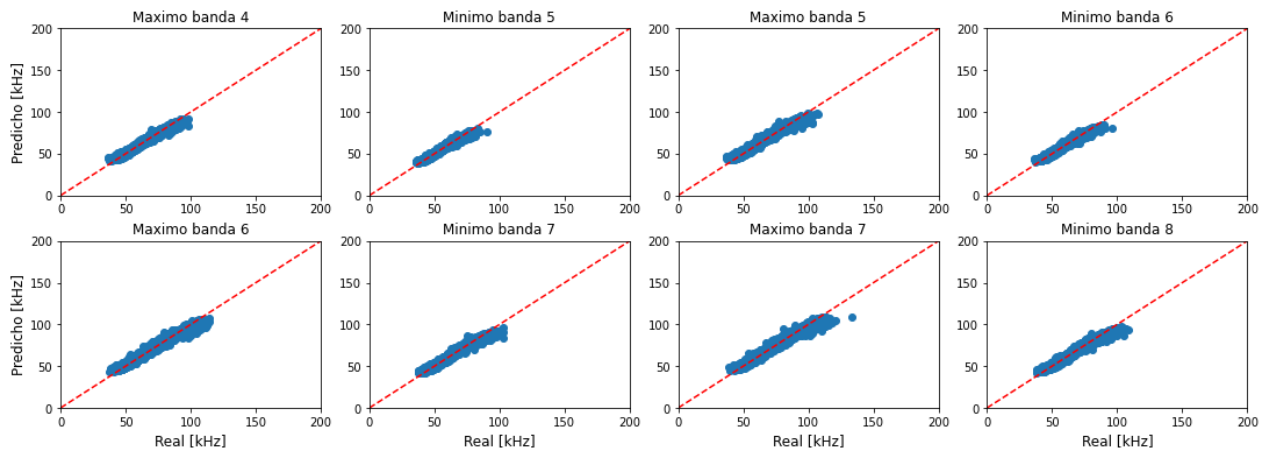


Figura 5-1 Gráficos de dispersión de las predicciones de valores mínimo y máximos de bandas en la estructura 2. Predicción realizada con KNN.

La Figura 5-2 corresponde al gráfico de dispersión de la frecuencia media y ancho de band gap calculados a partir de los valores mínimos y máximos predichos con el modelo KNN.

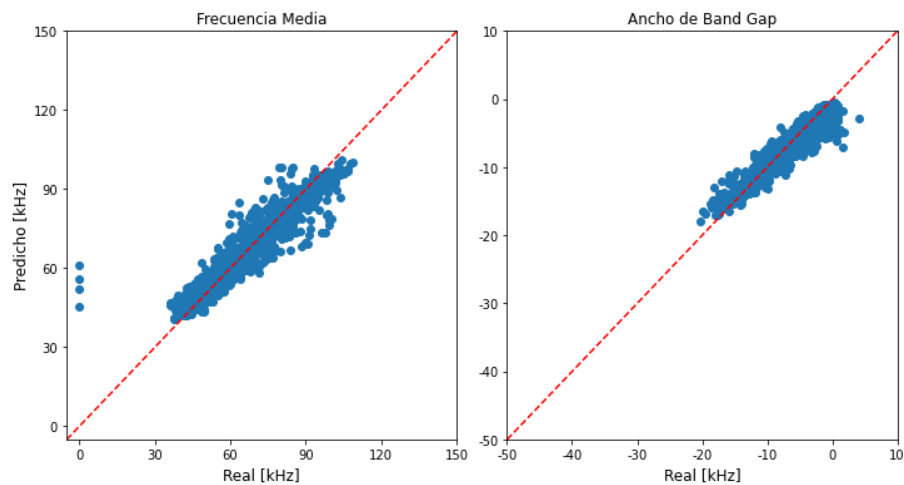


Figura 5-2 Gráficos de dispersión de frecuencia media y ancho de banda real vs. calculada, para la estructura 2. Predicción realizada con KNN.

En la Tabla 5-1 se entrega el error absoluto medio obtenido de la predicción, con KNN, de los valores mínimos y máximos de bandas para la estructura 2. Mientras que en la Tabla 5-2 se entregan los coeficientes de determinación del cálculo de frecuencia media y ancho de banda.

Tabla 5-1 Valores de MAE y correspondientes a la predicción, con KNN, de mínimos y máximos de bandas para la estructura 2.

	MAE	
	μ [Hz]	σ [Hz]
Máx. Banda 4	1.901	41
Mín. Banda 5	1.492	25
Máx. Banda 5	2.148	45
Mín. Banda 6	1.596	32
Máx. Banda 6	2.448	49
Mín. Banda 7	1.887	36
Máx. Banda 7	2.492	63
Mín. Banda 8	1.986	36

Tabla 5-2 Valores de R^2 del cálculo de frecuencia media y ancho de banda para la estructura 2. Predicción realizada con KNN.

	R^2	
	μ	σ
$R^2_{bandgap}$	0,888	0,007
R^2_{freq}	0,896	0,015
$R^2_{freq-neg}$	0,893	0,016
$R^2_{freq-pos}$	0,916	0,010
R^2_{gap}	0,879	0,004
$R^2_{gap-neg}$	0,903	0,005
$R^2_{gap-pos}$	-47,167	0,724

La Figura 5-3 corresponde a la matriz de confusión del cálculo de band gaps de la estructura 2, empleando los valores mínimos y máximos predichos con KNN.

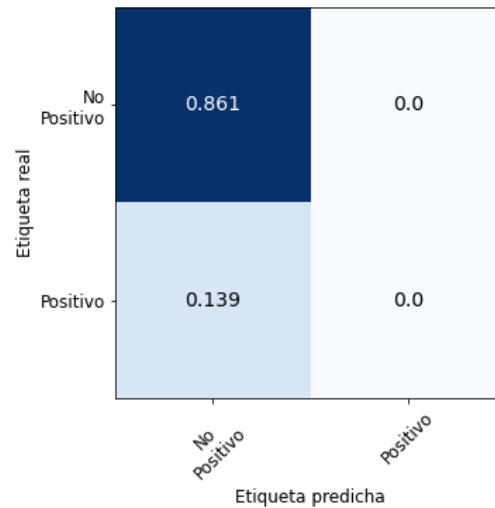


Figura 5-3 Matriz de confusión del cálculo de band gap, estructura 2. Predicciones realizadas con KNN.

5.2. Enfoque 2

Los mejores resultados para el enfoque 2 son obtenidos con el modelo RNN implementado para la estructura 2. El tiempo de entrenamiento de éste fue menor a 5 minutos.

En la Figura 5-4 se presentan los gráficos de dispersión de la frecuencia media y ancho de banda predichos con el modelo ya mencionado.

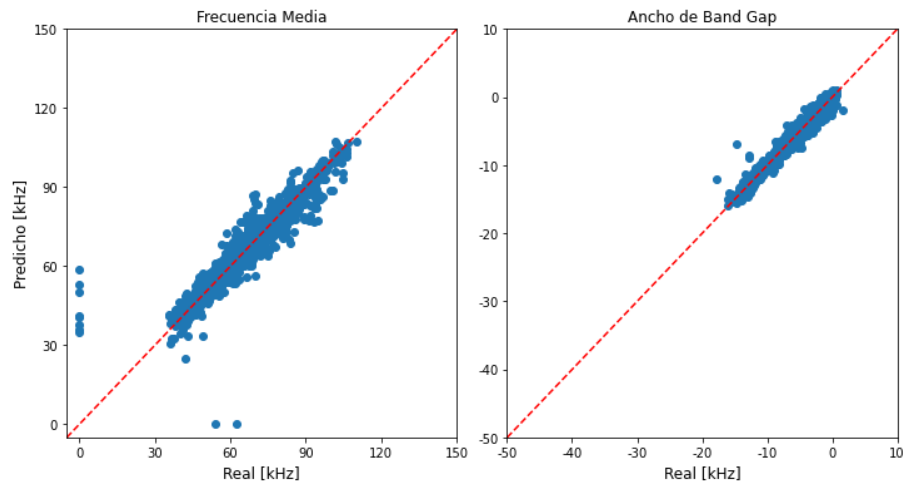


Figura 5-4 Gráficos de dispersión de la predicción, con RNN, de la frecuencia media y ancho de band gap para la estructura 2.

En la Tabla 5-3 se entregan los coeficientes de determinación de la predicción de frecuencia media y ancho de banda, realizada en la estructura 2 con el modelo RNN.

Tabla 5-3 Valores de R^2 de la predicción de frecuencia media y ancho de banda para la estructura 2. Predicción realizada con RNN.

	R^2	
	μ	σ
$R^2_{bandgap}$	0,942	0,007
R^2_{freq}	0,915	0,012
$R^2_{freq-neg}$	0,916	0,013
$R^2_{freq-pos}$	0,783	0,304
R^2_{gap}	0,970	0,002
$R^2_{gap-neg}$	0,970	0,002
$R^2_{gap-pos}$	-7,975	5,923

La Figura 5-5 corresponde a la matriz de confusión generada a partir de la predicción de frecuencia media y ancho de band gap para la estructura 2, utilizando RNN.

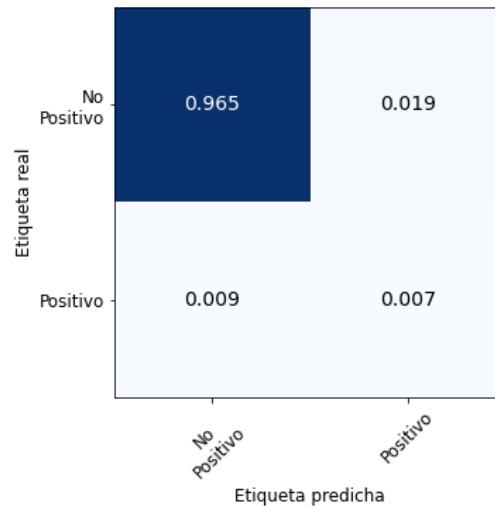


Figura 5-5 Matriz de confusión de la predicción de band gaps, utilizando RNN para la estructura 2.

5.3. Variación de la proporción de muestras con band gap

En la Tabla 5-4 se entregan los valores de coeficientes de determinación obtenidos en la evaluación del modelo RNN implementado para la predicción de frecuencia media y ancho de banda de la estructura 3.

Estos valores corresponden al promedio de cinco rutinas de entrenamiento y evaluación. En la Tabla 5-5 se entrega el valor de la desviación estándar.

Por último, en la Figura 5-6 se grafica la evolución que tiene el coeficiente de determinación al variar la cantidad de muestras, eliminando datos sin band gaps positivos asociados. Por simplicidad se grafican solamente el $R^2_{bandgap}$ y $R^2_{gap-pos}$.

Tabla 5-4 Promedios de coeficientes de determinación obtenido en entrenamientos con diferentes tamaños de conjuntos de datos.

$n_samples$	$R^2_{bandgap}$	R^2_{freq}	$R^2_{freq-neg}$	$R^2_{freq-pos}$	R^2_{gap}	$R^2_{gap-neg}$	$R^2_{gap-pos}$
20.000	0,948	0,927	0,931	0,423	0,968	0,968	-11,988
19.500	0,942	0,915	0,914	0,939	0,969	0,970	-10,354
19.000	0,942	0,916	0,916	0,889	0,969	0,969	-9,625
18.500	0,945	0,921	0,920	0,935	0,970	0,970	-7,952
18.000	0,941	0,915	0,917	0,638	0,967	0,967	-12,692
17.500	0,941	0,913	0,917	0,531	0,969	0,969	-10,115
17.000	0,943	0,921	0,921	0,953	0,966	0,966	-7,369
16.500	0,939	0,910	0,911	0,875	0,968	0,968	-11,559
16.000	0,943	0,922	0,923	0,841	0,966	0,965	-7,990
15.500	0,939	0,909	0,913	0,707	0,970	0,970	-7,862
15.000	0,944	0,918	0,918	0,940	0,969	0,969	-7,533
14.500	0,944	0,919	0,921	0,783	0,969	0,969	-9,843
14.000	0,937	0,912	0,913	0,853	0,963	0,963	-12,137
13.500	0,941	0,913	0,914	0,796	0,968	0,968	-5,871
13.000	0,934	0,904	0,905	0,835	0,964	0,963	-6,971
12.500	0,937	0,905	0,913	0,547	0,968	0,968	-4,461
12.000	0,938	0,912	0,913	0,883	0,963	0,963	-10,640
11.500	0,935	0,907	0,909	0,816	0,963	0,963	-11,754
11.000	0,936	0,908	0,909	0,812	0,964	0,963	-8,032
10.500	0,943	0,921	0,920	0,961	0,966	0,965	-8,994
10.000	0,939	0,911	0,913	0,810	0,966	0,966	-7,064
9.500	0,929	0,895	0,899	0,617	0,964	0,963	-9,017
9.000	0,936	0,910	0,915	0,572	0,962	0,961	-14,764
8.500	0,925	0,892	0,890	0,951	0,958	0,957	-10,532
8.000	0,929	0,897	0,896	0,938	0,960	0,960	-13,392
7.500	0,933	0,906	0,910	0,699	0,959	0,957	-9,496
7.000	0,936	0,912	0,910	0,946	0,960	0,959	-7,648
6.500	0,926	0,895	0,911	0,377	0,958	0,956	-16,195
6.000	0,926	0,896	0,910	0,459	0,956	0,955	-8,787
5.500	0,935	0,921	0,920	0,944	0,948	0,946	-8,302
5.000	0,927	0,902	0,906	0,791	0,951	0,950	-11,157

Tabla 5-4 (Continuación).

$n_samples$	$R^2_{bandgap}$	R^2_{freq}	$R^2_{freq-neg}$	$R^2_{freq-pos}$	R^2_{gap}	$R^2_{gap-neg}$	$R^2_{gap-pos}$
4.500	0,906	0,859	0,864	0,791	0,952	0,952	-10,401
4.000	0,911	0,878	0,874	0,938	0,943	0,944	-8,493
3.500	0,925	0,902	0,899	0,951	0,947	0,944	-11,455
3.000	0,909	0,869	0,882	0,715	0,949	0,947	-11,074
2.500	0,915	0,893	0,907	0,674	0,937	0,930	-11,236
2.000	0,893	0,861	0,890	0,584	0,926	0,916	-7,828
1.500	0,903	0,874	0,861	0,957	0,932	0,918	-6,622
1.000	0,865	0,802	0,845	0,645	0,928	0,908	-10,515
500	0,875	0,915	0,868	0,949	0,835	0,707	-11,488

Tabla 5-5 Desviación estándar obtenidas con cinco ciclos de entrenamiento y evaluación.

$n_samples$	$R^2_{bandgap}$	R^2_{freq}	$R^2_{freq-neg}$	$R^2_{freq-pos}$	R^2_{gap}	$R^2_{gap-neg}$	$R^2_{gap-pos}$
20.000	0,003	0,006	0,007	0,477	0,003	0,003	3,741
19.500	0,007	0,014	0,014	0,040	0,001	0,000	5,669
19.000	0,009	0,018	0,018	0,132	0,003	0,003	7,158
18.500	0,003	0,007	0,007	0,034	0,001	0,001	2,962
18.000	0,004	0,008	0,007	0,606	0,002	0,002	7,362
17.500	0,006	0,012	0,009	0,817	0,002	0,003	5,588
17.000	0,010	0,021	0,022	0,016	0,005	0,005	1,707
16.500	0,005	0,010	0,009	0,178	0,002	0,002	6,313
16.000	0,007	0,013	0,015	0,217	0,004	0,004	6,303
15.500	0,007	0,015	0,015	0,344	0,001	0,001	4,236
15.000	0,009	0,017	0,017	0,032	0,001	0,001	2,153
14.500	0,006	0,013	0,012	0,316	0,003	0,003	6,951
14.000	0,004	0,009	0,011	0,162	0,001	0,001	5,309
13.500	0,008	0,016	0,015	0,266	0,002	0,002	5,404
13.000	0,009	0,018	0,018	0,183	0,002	0,002	3,998
12.500	0,011	0,021	0,016	0,819	0,003	0,003	3,293
12.000	0,008	0,018	0,018	0,160	0,005	0,005	4,913
11.500	0,015	0,028	0,024	0,267	0,004	0,003	8,199
11.000	0,009	0,017	0,013	0,305	0,002	0,002	2,277
10.500	0,006	0,012	0,012	0,010	0,003	0,003	8,047
10.000	0,003	0,009	0,009	0,286	0,004	0,005	3,557
9.500	0,015	0,029	0,029	0,428	0,001	0,002	4,031
9.000	0,005	0,006	0,009	0,478	0,005	0,005	4,962
8.500	0,020	0,040	0,041	0,013	0,002	0,002	6,034
8.000	0,012	0,022	0,022	0,018	0,002	0,002	6,833
7.500	0,017	0,033	0,025	0,485	0,003	0,003	2,847
7.000	0,018	0,031	0,032	0,026	0,005	0,005	5,519
6.500	0,007	0,014	0,009	0,525	0,004	0,004	12,249
6.000	0,015	0,029	0,008	0,971	0,006	0,007	6,302
5.500	0,007	0,013	0,013	0,013	0,006	0,007	6,479
5.000	0,012	0,023	0,016	0,327	0,005	0,005	5,461

Tabla 5-5 (Continuación).

$n_samples$	$R^2_{bandgap}$	R^2_{freq}	$R^2_{freq-neg}$	$R^2_{freq-pos}$	R^2_{gap}	$R^2_{gap-neg}$	$R^2_{gap-pos}$
4.500	0,031	0,059	0,065	0,333	0,005	0,004	5,355
4.000	0,021	0,041	0,043	0,015	0,002	0,005	8,113
3.500	0,013	0,024	0,026	0,014	0,005	0,005	5,159
3.000	0,039	0,073	0,048	0,481	0,006	0,006	2,106
2.500	0,021	0,042	0,013	0,540	0,004	0,005	5,112
2.000	0,024	0,047	0,043	0,406	0,007	0,008	3,322
1.500	0,037	0,067	0,078	0,010	0,011	0,015	3,684
1.000	0,039	0,075	0,049	0,301	0,006	0,014	8,976
500	0,031	0,059	0,065	0,333	0,005	0,004	5,355

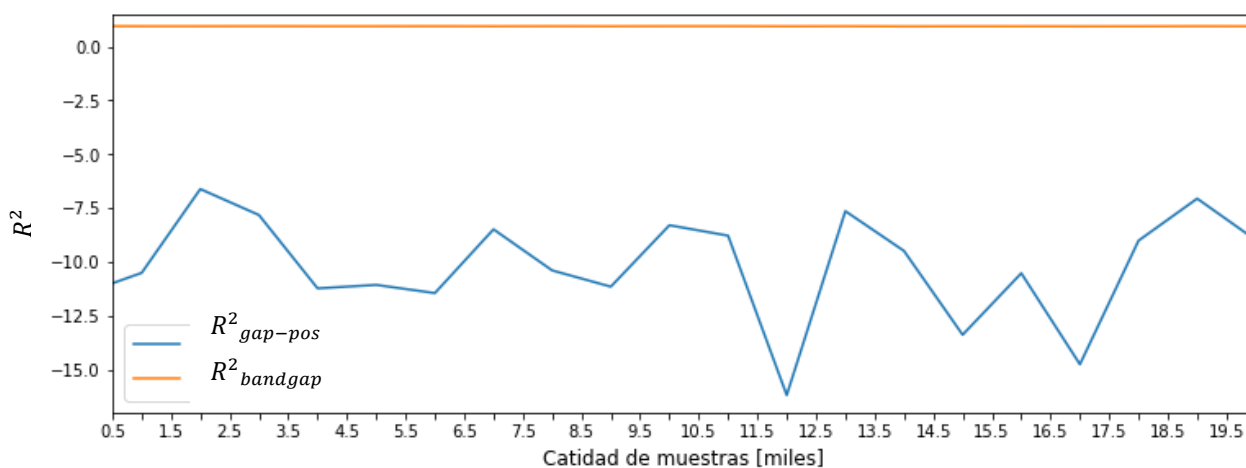


Figura 5-6 Variación en el coeficiente de determinación debido al cambio en la cantidad de muestras sin band gap.

Capítulo 6

Análisis y discusión

Tanto en el enfoque 1 como en el 2 se obtienen buenos valores en el coeficiente de determinación $R^2_{bandgap}$, habiendo obtenido un valor máximo de 0,888 en el enfoque 1 para la estructura 2 empleando KNN, y de 0,942 en el enfoque 2 para la misma estructura, pero empleando RNN.

Esto habla de que sí es posible ajustar modelos de aprendizaje de máquinas para obtener las propiedades de band gaps. Sin embargo, todos los modelos implementados presentan un sesgo que afecta la predicción de band gaps positivos y el efecto es mayor en la predicción de los anchos de banda.

Lo último se evidencia al observar que todos los valores de $R^2_{gap-neg}$ resultan negativos mientras que $R^2_{gap-pos}$ posee valores cercanos al $R^2_{bandgap}$. Por otro lado, se tiene que tanto $R^2_{frec-neg}$ como $R^2_{frec-pos}$ poseen valores cercanos al coeficiente de determinación global, pero también ocurre que el primero tiende a ser menor que el segundo. La explicación de estos resultados debe analizarse de manera separada para cada enfoque.

La razón del mal desempeño en el cálculo de ancho de band gaps positivos, en el enfoque 1, yace en la magnitud del error absoluto medio de la predicción de los valores mínimos y máximos de las bandas.

Tomando como ejemplo los valores de MAE obtenidos en la predicción de bandas de la estructura 2 en la Tabla 5-1 y comparándolo con los promedios de los mínimos y máximos de las bandas de esta estructura, presentados en la Tabla 4-2, se puede notar que los errores absolutos medios son relativamente bajos, están en torno a los 2.000 Hz, mientras que los valores que se buscan predecir son superiores a los 50.000 Hz.

Sin embargo, el error obtenido en la predicción de cada banda es considerablemente mayor a la magnitud del ancho de banda promedio en band gaps positivos, que está en los 300 Hz (ver Tabla 4-3), mientras que el ancho de banda promedio en band gaps

negativos es -4.500 Hz (Tabla 4-4). Esto explica el mal desempeño en la predicción de anchos de banda positivo.

Por otro lado, los promedios de las frecuencias medias y el rango de valores son similares entre los band gaps positivos y negativos. El peor desempeño en la predicción de frecuencias medias de band gaps positivos, se atribuye a que estas se encuentran menos uniformemente distribuidas (Ver Figura 7-10 y Figura 7-11).

Este análisis se refuerza al estudiar el modelo RNN del enfoque 1 implementado para la predicción de bandas de la estructura 3, llama la atención los malos resultados obtenidos con este modelo ($R^2_{bandgap}$ de -0,563). Si bien, aparentemente la predicción de bandas con este modelo es similar a las otras (ver Anexo D.1), el error absoluto medio obtenido es considerablemente más alto: en torno a 15.000 Hz para todas las bandas.

Para analizar los resultados obtenidos con el enfoque 2, hay que concentrarse en las diferencias de distribuciones de frecuencia media y ancho de banda de band gaps positivo y negativos (ver Anexo A.3 y A.4).

El mal desempeño en la predicción de ancho de band gap positivo ($R^2_{gap-pos}$). Se atribuye a dos posibles factores: El primero es el bajo porcentaje de muestras que presentan band gap positivo, menor al 2% en todas las estructuras (ver Tabla 4-1), lo que produce que los modelos se ajusten mejor a los band gaps no-positivos. Mientras que el segundo factor el gran desbalance que se observa en la distribución de ancho de bandas positivos.

Los resultados entregados en la sección 5.3 hablan sobre el efecto de estos factores. En estos se muestra que los valores de $R^2_{gap-pos}$ y $R^2_{gap-neg}$ se mantienen invariables a medida que aumenta el porcentaje de muestras con band gaps positivos. De esto se desprende que el mayor impacto en el desempeño de la predicción, lo tiene el gran desbalance de datos con band gap positivo.

Siguiendo en la línea de la predicción de band gaps positivos y negativos, es interesante analizar el porcentaje de band gaps positivos predichos como tal. Se puede obtener esta información mirando las matrices de confusión.

En el enfoque 1, el que mejor realiza esta tarea es el modelo RNN implementado para la estructura 3, que identifica correctamente el 21% de las muestras que poseen band gap positivo (ver Figura 7-20b). Por otro lado, en el enfoque 2 el modelo RNN implementado para la estructura 3 detecta el 73% de las muestras con band gap positivo (ver Figura 7-26b).

Sobre este último punto, se destaca además el mal desempeño que tienen los modelos KNN para discernir entre band gaps positivos y negativos, en el mejor caso se detectó el 11% de los band gaps positivos (enfoque 2, estructura 3. Ver Figura 7-26a).

No se encuentra una razón válida que explique porque no se obtienen predicciones de band gaps positivos con los modelos KNN en el enfoque 1. Pero en el enfoque 2 esto se atribuye a la sobrerrepresentación que tienen las muestras con anchos de banda

negativos. Recordando cómo trabaja el algoritmo KNN, al momento de predecir una muestra que posee band gap positivo, es probable que más de un vecino corresponda a un band gap negativo y en algunos casos con valores de ancho de banda muy lejanos a cero.

Capítulo 7

Conclusión

Al final de este trabajo, se logra desarrollar una metodología para la implementación y evaluación de modelos de aprendizaje de máquinas capaces de predecir band gaps en distintas topologías de paneles tipo sándwich.

Pese a que el análisis agregado de las métricas de evaluación muestra buenos resultados: el mejor $R^2_{bandgap}$ es 0,942, con el modelo RNN del enfoque 2 utilizado para la estructura 2; mientras que el $R^2_{bandgap}$, positivo, más bajo es 0,369, con el modelo RNN del enfoque 1 utilizado en la estructura 2. Si se examinan de manera desagregada, se puede notar que todos los modelos tienen en común el mal desempeño en la predicción de anchos de bandas de band gaps positivos: Todos obtienen un $R^2_{gap-pos}$ menor a cero. Es este el principal aspecto a mejorar en este estudio.

En otras palabras, ninguno de los modelos implementados muestra un desempeño que permita concluir que efectivamente sea posible emplear técnicas de aprendizaje de máquinas para la obtención de frecuencia media y ancho de band gaps, a partir de los parámetros que definen la topología de la estructura.

Para mejorar los resultados en la predicción de band gaps positivos, utilizando el enfoque 1, se propone buscar topologías con las que se consigan band gaps cuyos anchos de bandas se encuentren, al menos, entre los 1.000 y 10.000 Hz. Esto debiese superar el error absoluto medio obtenido en la predicción de valores mínimos y máximos de cada banda.

Por otro lado, para mejorar los resultados del enfoque 2 se proponen dos alternativas: Trabajar con topologías de paneles que produzcan band gaps distribuidos más uniformemente que los vistos en este trabajo. O, en su defecto, emplear durante el preprocesamiento de datos, técnicas de *oversampling* y/o *undersampling* para problemas de regresión, como los desarrollados por Yang y su equipo [22]. Ambas alternativas

apuntan a superar el problema asociado al gran desbalance en los dantos de anchos de banda positivo.

No obstante, los modelos implementados en esta memoria pueden servir para el desarrollo de una metodología de optimización topológica de band gaps a través de modelos subrogados. Teniendo para esto en consideración que, inicialmente, el estudio debe limitarse a la optimización de band gaps negativos.

Bibliografía

- [1] J. R. Vinson, «Sandwich structures: past, present, and future,» *Sandwich structures 7: advancing with sandwich structures and materials*, pp. 3-12, 2005.
- [2] L. J. Gibson, «Cellular Solids,» *Mrs Bulletin*, vol. 28, nº 4, pp. 270-274, 2003.
- [3] M. Lia, Z. Chengb, G. Jiaa y Z. Shib, «Dimension reduction and surrogate based topology optimization of periodic structure,» *Composite Structures*, vol. 229, p. 111385, 2019.
- [4] L. Gibson y M. Ashby, *Cellular solids. Structure and properties*, Press Syndicate of the University of Cambridge, 1997.
- [5] L. Valdevit, A. J. Jacobsen, J. R. Greer y W. B. Carter, «Protocols for the optimal design of multi-functional cellular structures: from hypersonics to micro-architected materials,» *Journal of the American Ceramic Society*, vol. 94, pp. s15-s34, 2011.
- [6] T. A. Schaedler y W. B. Carter, «Architected cellular materials,» *Annual Review of Materials Research*, vol. 46, pp. 187-210, 2016.
- [7] G. A. Gazonas, D. S. Weile, R. Wildman y A. Mohan, «Genetic algorithm optimization of phononic bandgap structures,» *International journal of solids and structures*, vol. 43, nº 18-19, pp. 5851-5866, 2006.
- [8] L. Quiteros, V. Meruane, E. Lenz y R. Ruiz, «Phononic Bandgap Optimization in Sandwich Panels Using Cellular Truss Cores,» *Materials*, vol. 14, nº 18, p. 5236, 2021.
- [9] G. Yi y B. D. Youn, «A comprehensive survey on topology optimization of phononic crystals,» *Structural and Multidisciplinary Optimization*, vol. 54, nº 5, pp. 1315-1344, 2016.
- [10] E. Alpaydin, *Introduction to Machine Learning*, Tercera ed., MIT press, 2014.
- [11] G. Wilfong, «Nearest neighbor problems,» *International Journal of Computational Geometry & Applications*, vol. 2, nº 04, pp. 383-416, 1992.
- [12] W. S. McCulloch y W. Pitts, «A logical calculus of the ideas immanent in nervous activity,» *The bulletin of mathematical biophysics*, vol. 5, nº 4, pp. 115-133, 1943.
- [13] R. P. Lippmann, «An introduction to computing with neural nets,» *IEEE Assp magazine*, vol. 4, nº 2, pp. 4-22, 1987.
- [14] J. Patterson y A. Gibson, *Deep learning: A practitioner's approach*, O'Reilly Media, Inc., 2017.

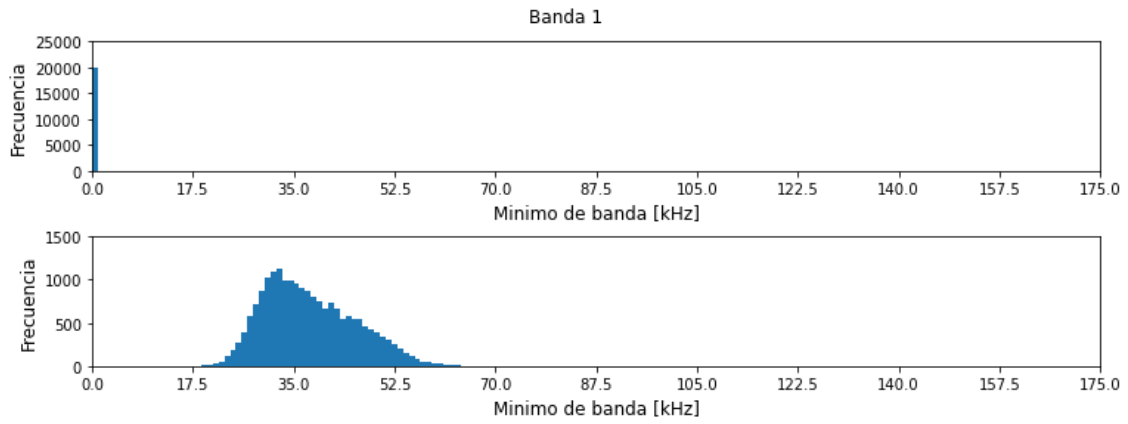
- [15] A. Gerón, *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems*, O'Reilly Media, 2019.
- [16] scikit-learn, «sklearn.preprocessing.StandardScaler,» [En línea]. Available: <https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html#sklearn.preprocessing.StandardScaler>. [Último acceso: Diciembre 2021].
- [17] scikit-learn, «Tuning the hyper-parameters of an estimator,» [En línea]. Available: https://scikit-learn.org/stable/modules/grid_search.html. [Último acceso: Diciembre 2021].
- [18] scikit-learn, «Regression metrics,» [En línea]. Available: https://scikit-learn.org/stable/modules/model_evaluation.html#regression-metrics. [Último acceso: Diciembre 2021].
- [19] V. Gálvez, «Desarrollo de modelos en elementos finitos parametrizados de paneles tipo sándwich con núcleos ultralivianos para predecir band gaps,» Universidad de Chile, 2021.
- [20] scikit-learn, «sklearn.neighbors.KNeighborsRegressor,» [En línea]. [Último acceso: Diciembre 2021].
- [21] scikit-learn, «sklearn.neural_network.MLPRegressor,» [En línea]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPRegressor.html. [Último acceso: Diciembre 2021].
- [22] Y. Yang, Zha, Kaiwen, Chen, Ying-Cong, Wang, Hao y Katabi, Dina, «Delving into Deep Imbalanced Regression,» *International Conference on Machine Learning (ICML)*, 2021.
- [23] O. Sigmund y J. S. Jensen, «Systematic design of phononic band-gap materials and structures by topology optimization,» *Philosophical Transactions of the Royal Society of London*, vol. 361, n° 1806, pp. 1001-1019, 2003.
- [24] L. Zong-fa, W. Bin y H. Cun-fu, «Band-gap optimization of twodimensional phononic crystals based on genetic algorithm and FPWE,» *Waves in Random and Complex Media*, vol. 24, n° 3, pp. 286-305, 2014.

Anexos

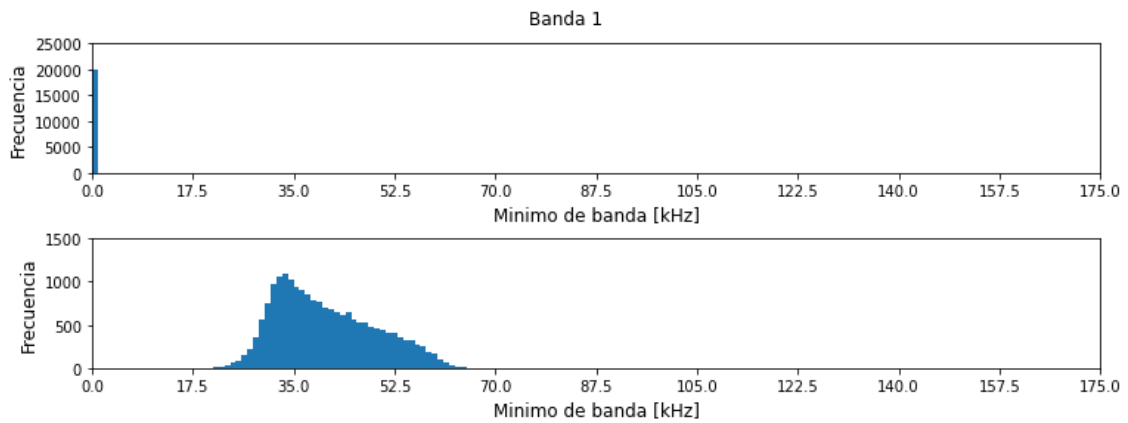
Anexo A

Gráficos para la inspección de datos

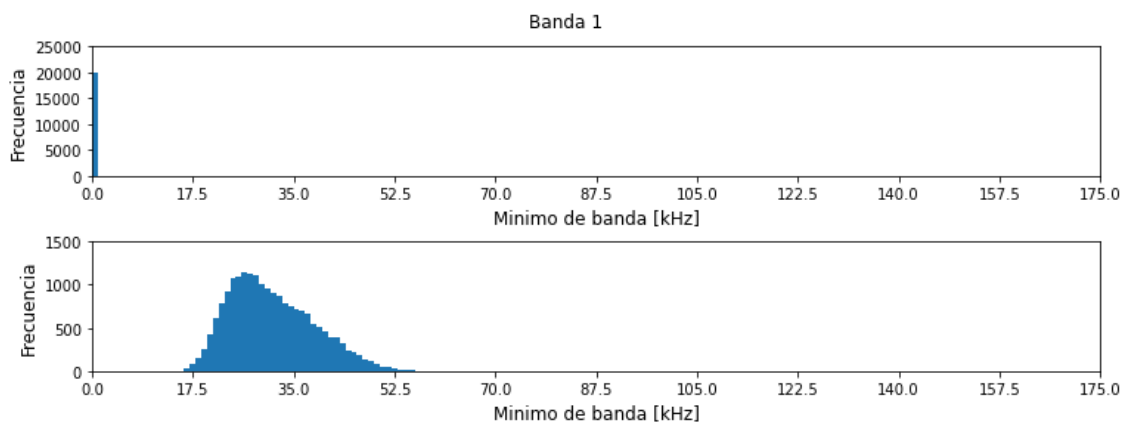
A.1 Histogramas de valores mínimos y máximos de bandas



(a)

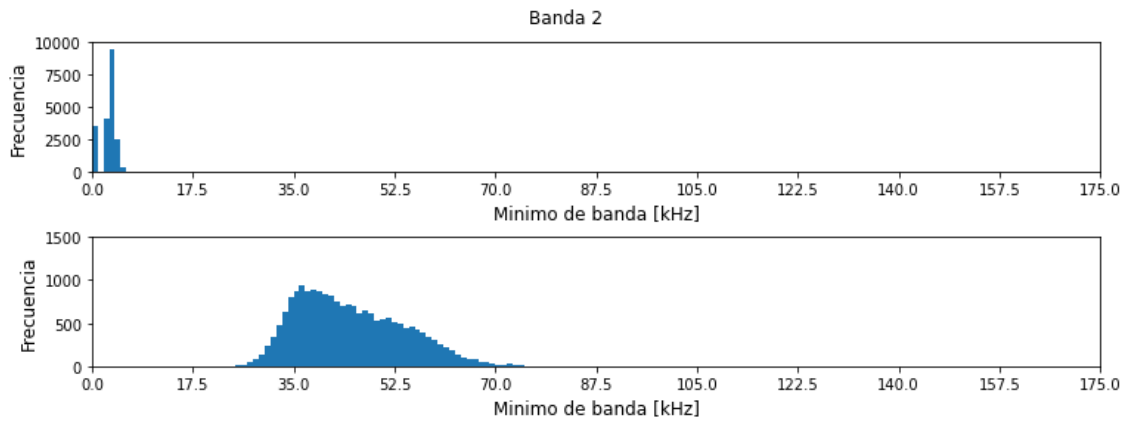


(b)

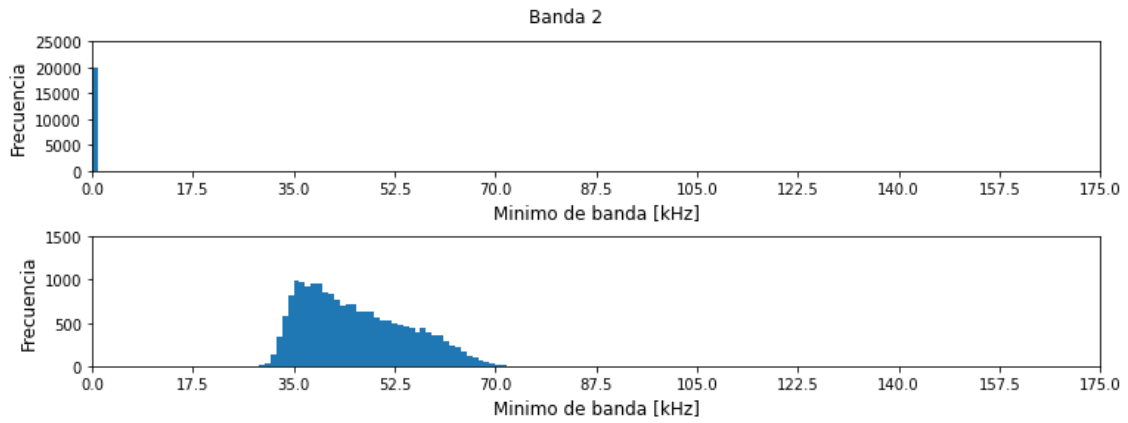


(c)

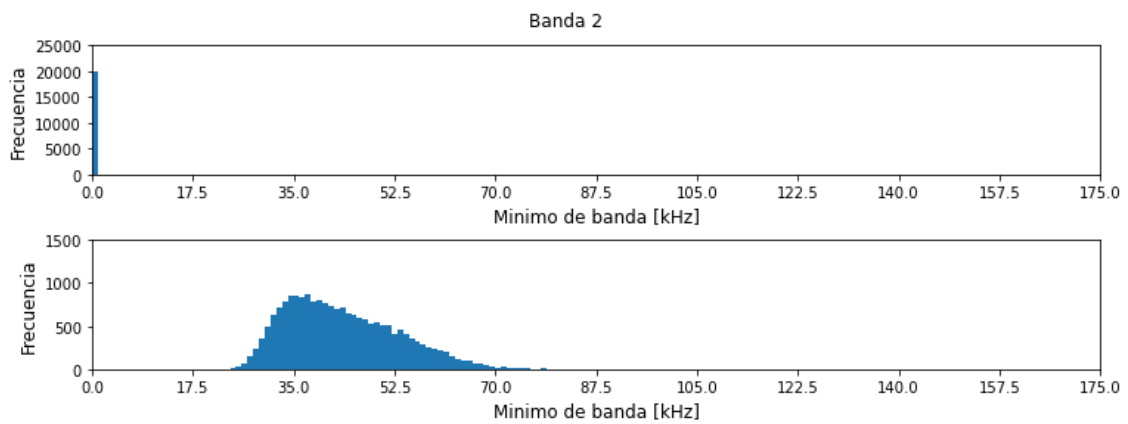
Figura 7-1 Histogramas de valores mínimos y máximos de banda 1. (a) estructura 1, (b) estructura 2 y (c) estructura 3.



(a)



(b)



(c)

Figura 7-2 Histogramas de valores mínimos y máximos de banda 2. (a) estructura 1, (b) estructura 2 y (c) estructura 3.

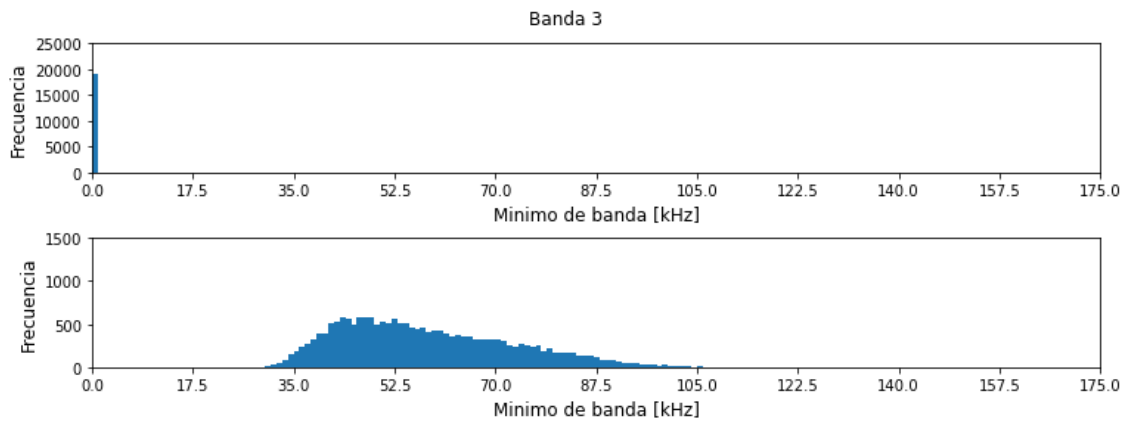
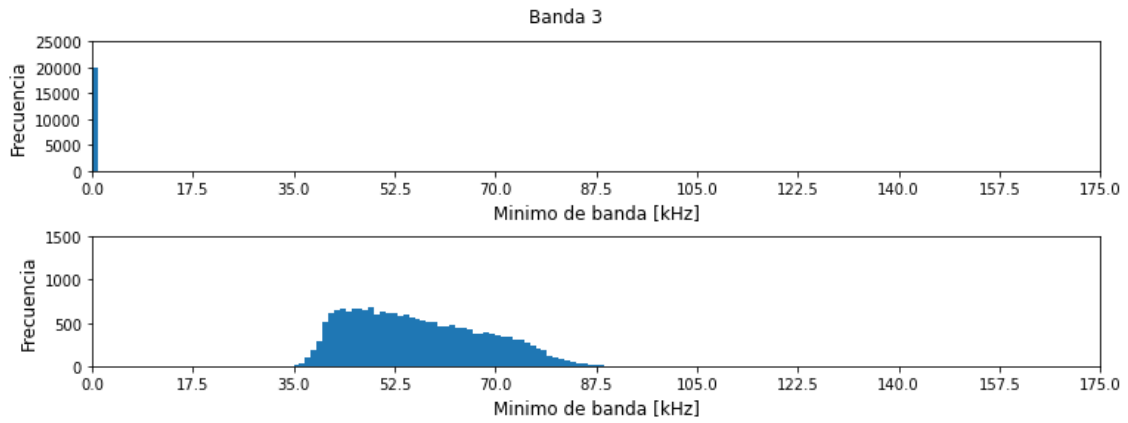
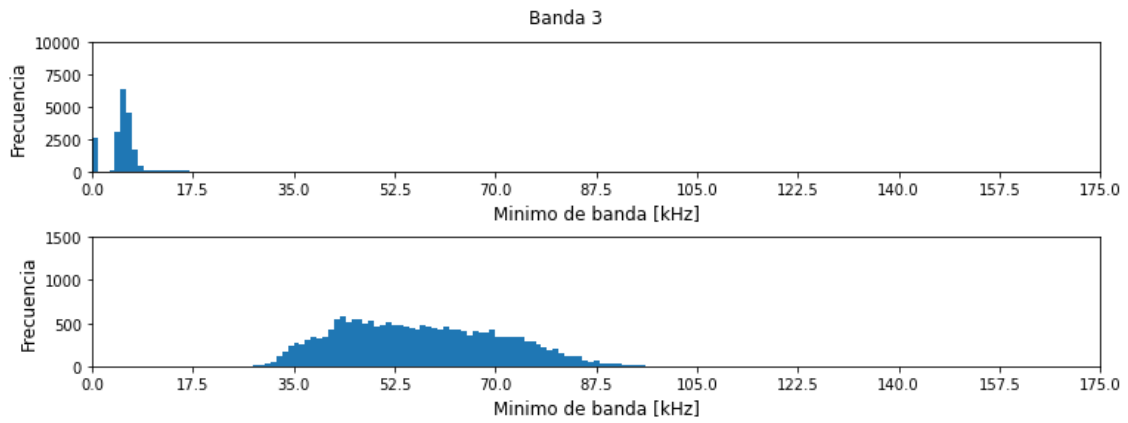
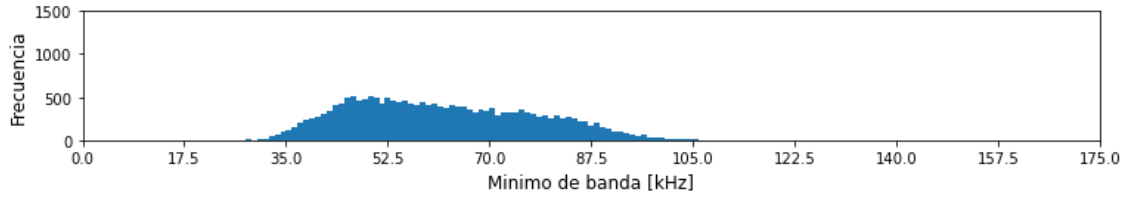
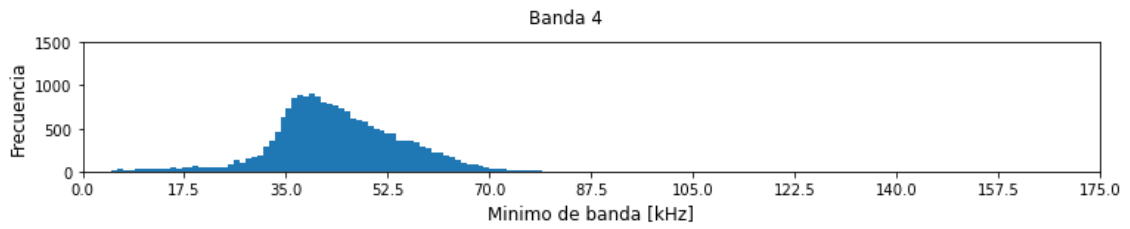
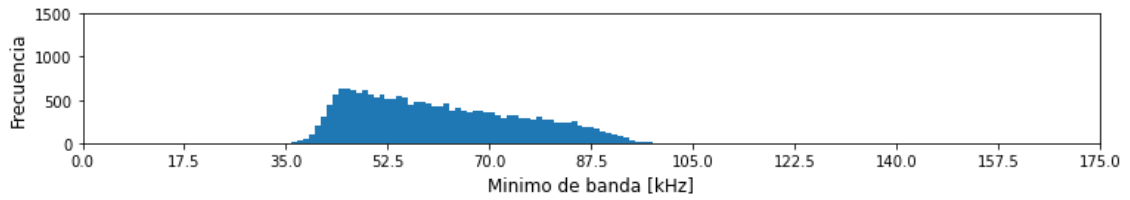
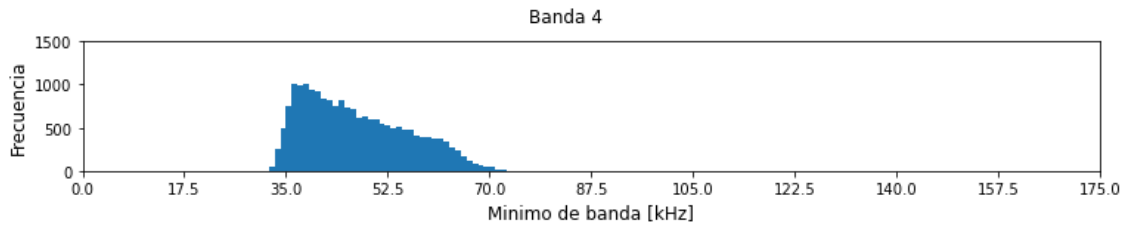


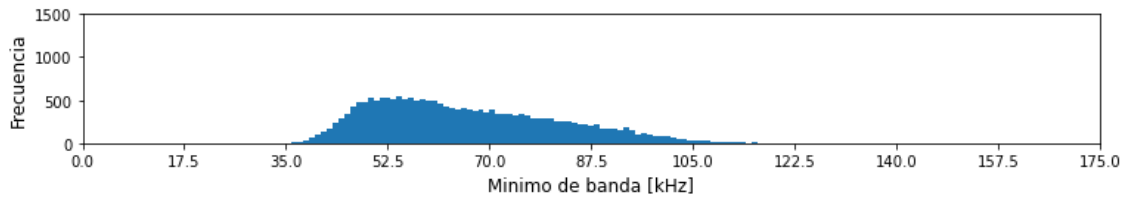
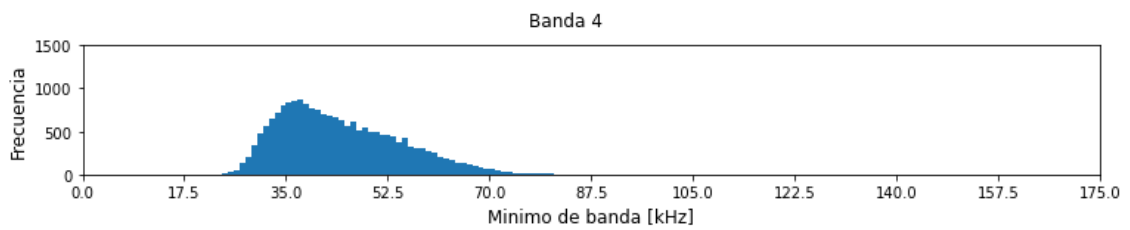
Figura 7-3 Histogramas de valores mínimos y máximos de banda 3. (a) estructura 1, (b) estructura 2 y (c) estructura 3.



(a)

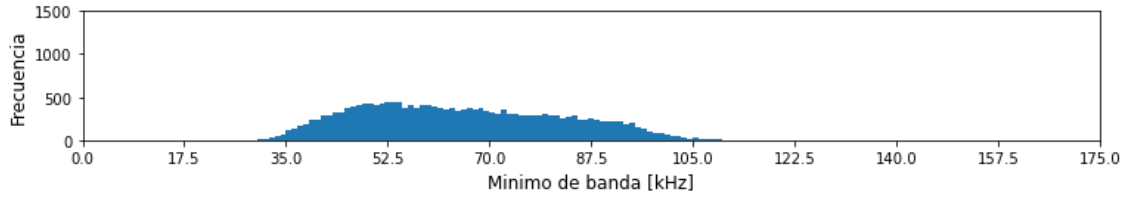
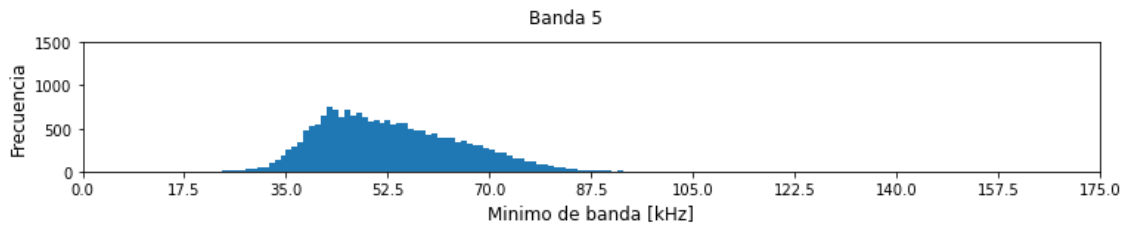


(b)

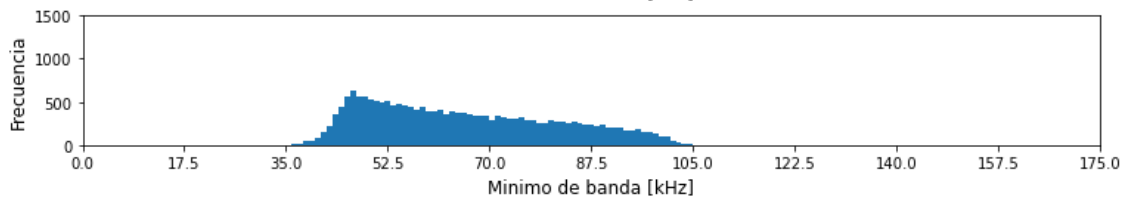
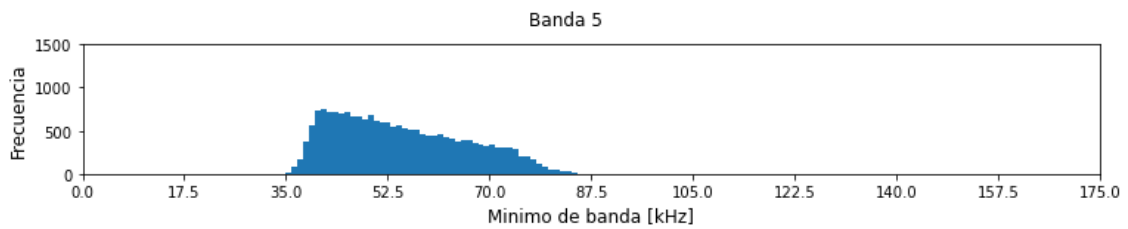


(c)

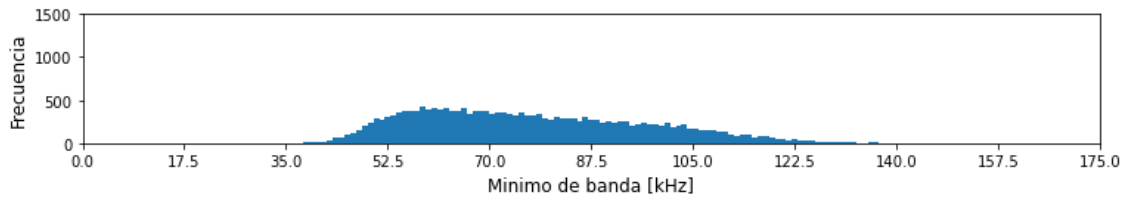
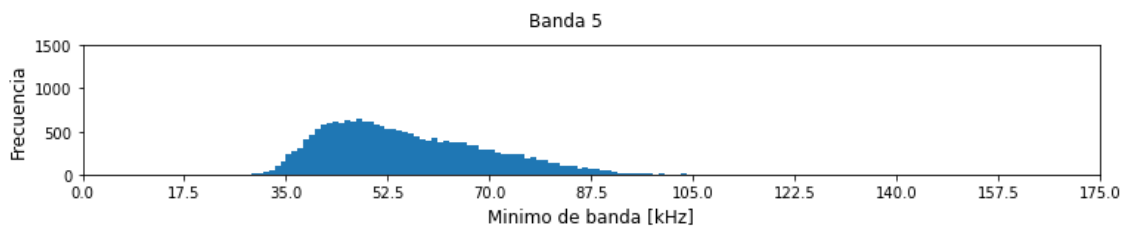
Figura 7-4 Histogramas de valores mínimos y máximos de banda 4. (a) estructura 1, (b) estructura 2 y (c) estructura 3.



(a)

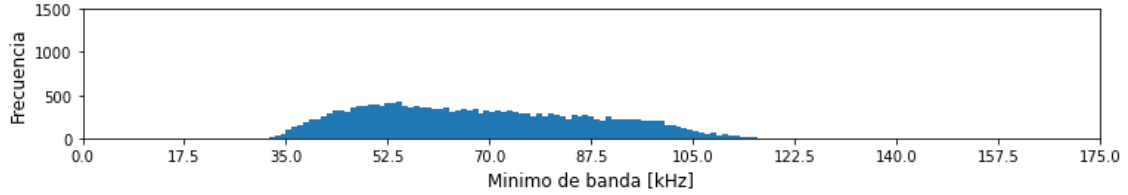
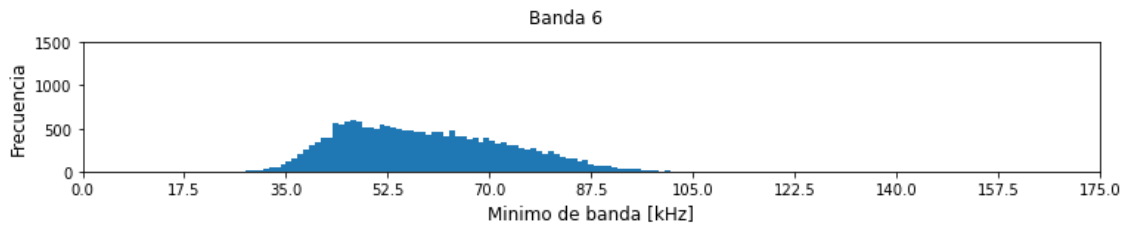


(b)

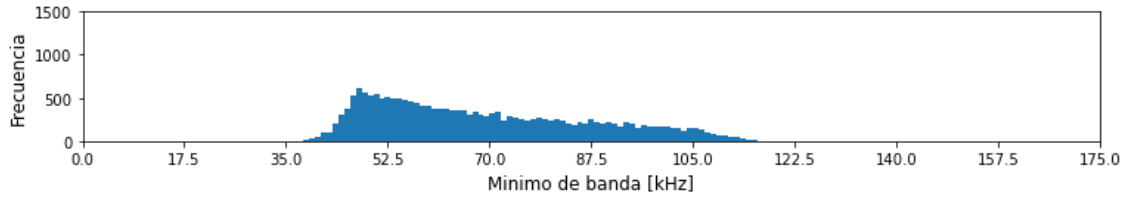
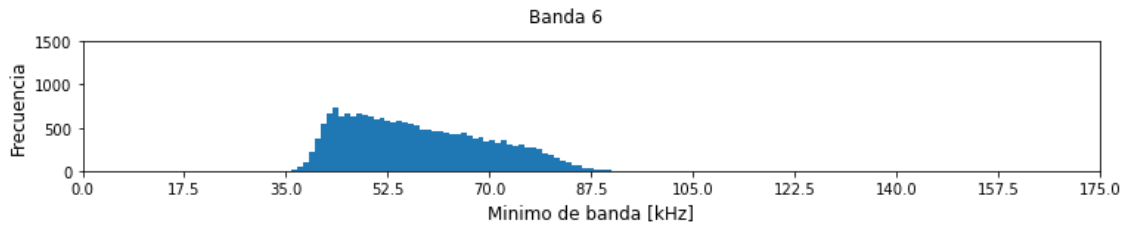


(c)

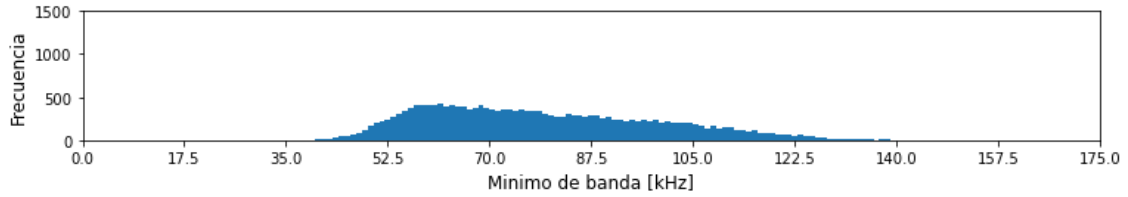
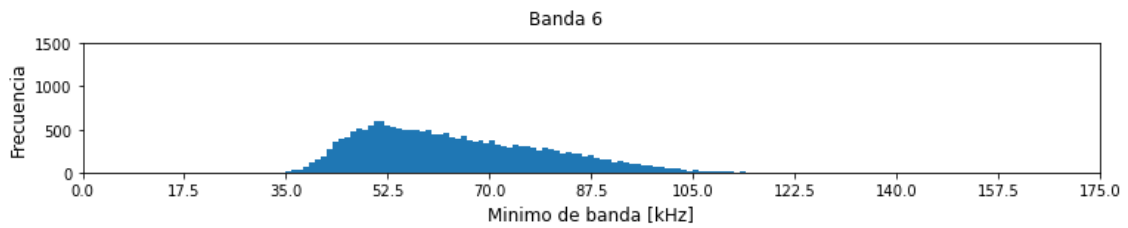
Figura 7-5 Histogramas de valores mínimos y máximos de banda 5. (a) estructura 1, (b) estructura 2 y (c) estructura 3.



(a)

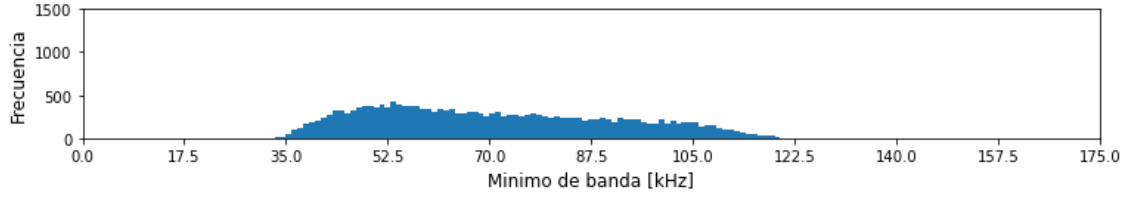
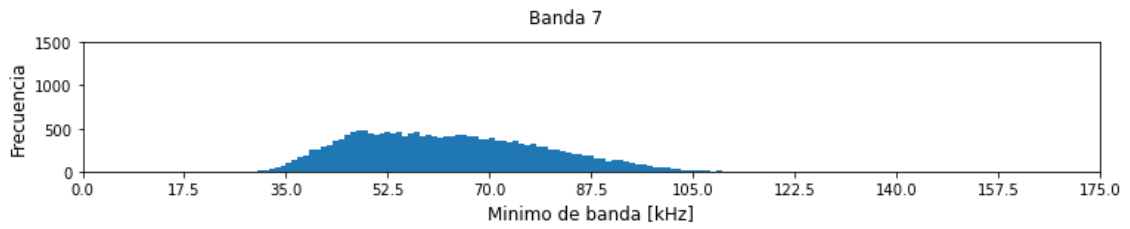


(b)

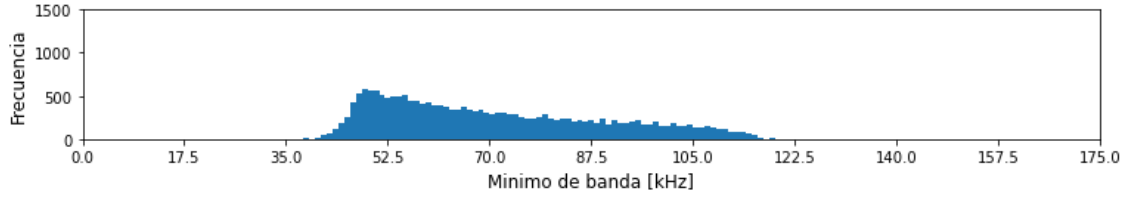
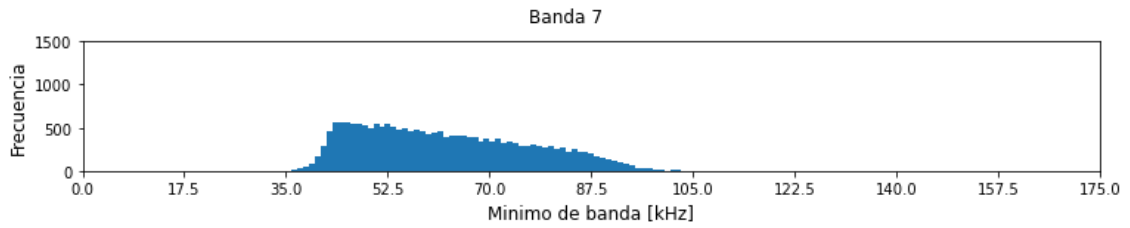


(c)

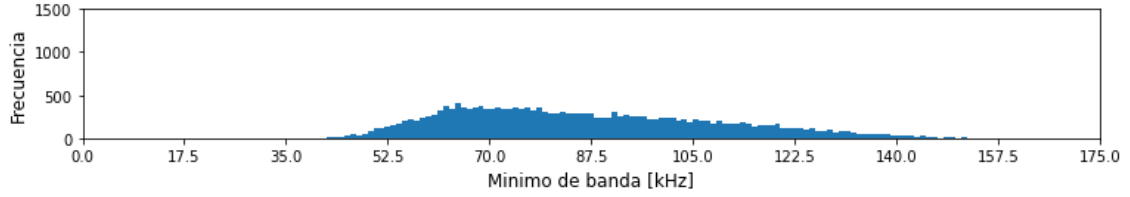
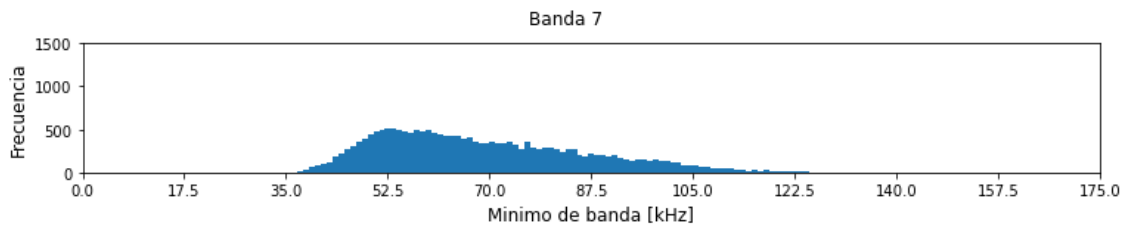
Figura 7-6 Histogramas de valores mínimos y máximos de banda 6. (a) estructura 1, (b) estructura 2 y (c) estructura 3.



(a)

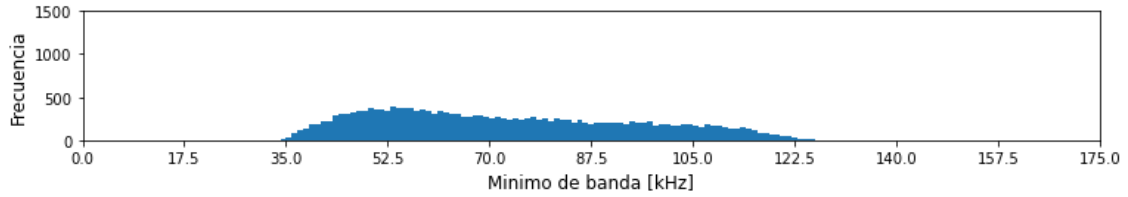
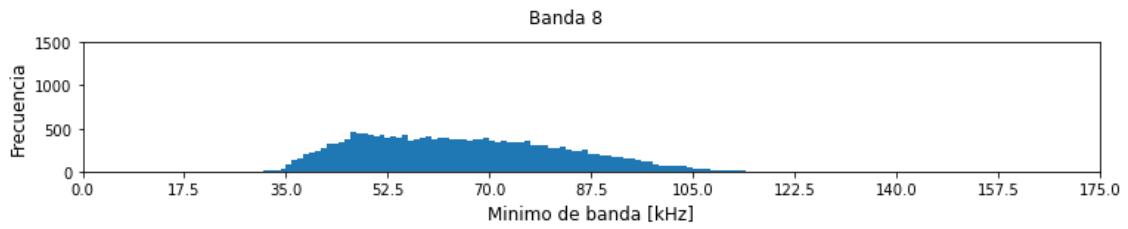


(b)

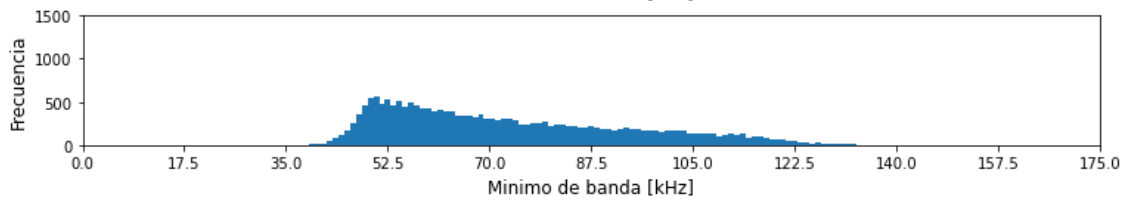
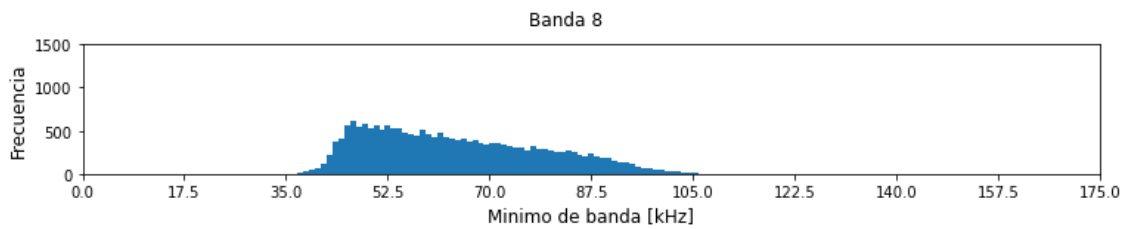


(c)

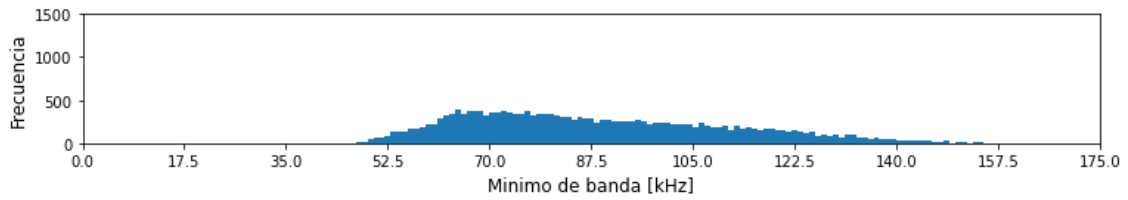
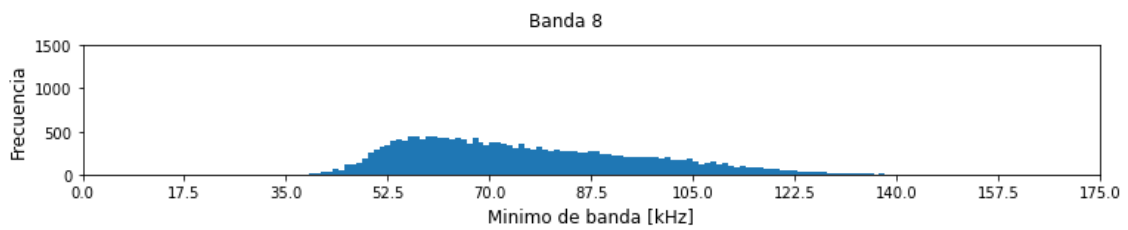
Figura 7-7 Histogramas de valores mínimos y máximos de banda 7. (a) estructura 1, (b) estructura 2 y (c) estructura 3.



(a)



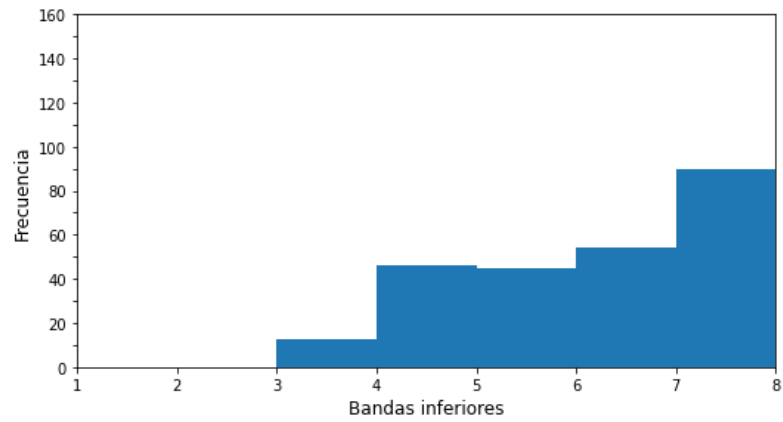
(b)



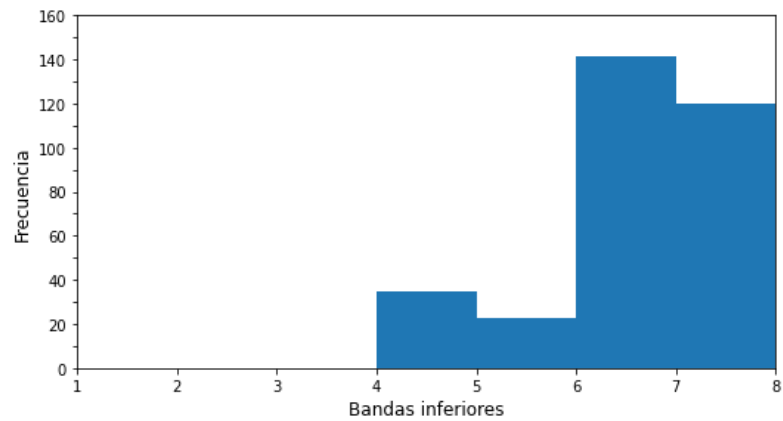
(c)

Figura 7-8 Histogramas de valores mínimos y máximos de banda 8. (a) estructura 1, (b) estructura 2 y (c) estructura 3.

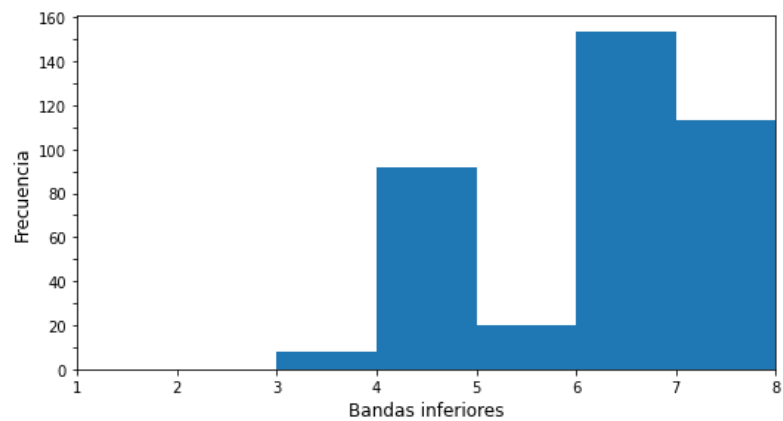
A.2 Histogramas de bandas que producen band gaps



(a)



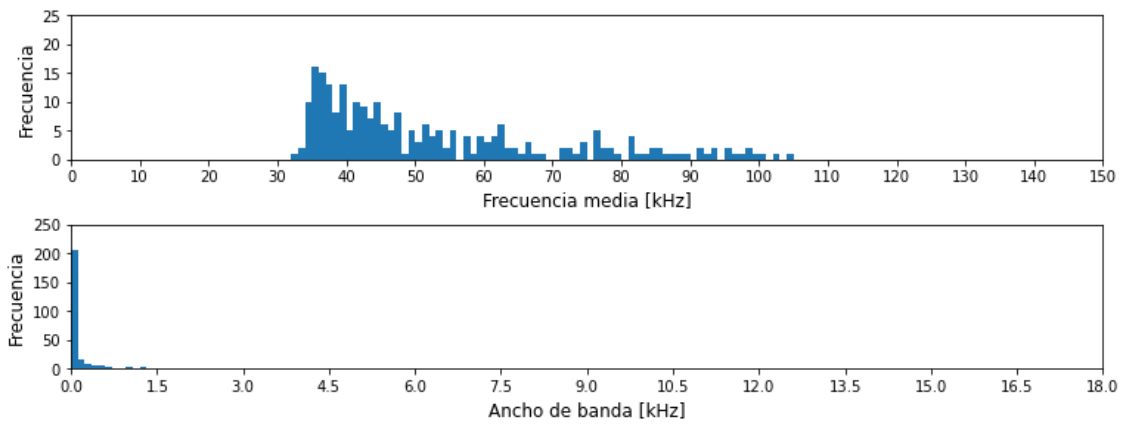
(b)



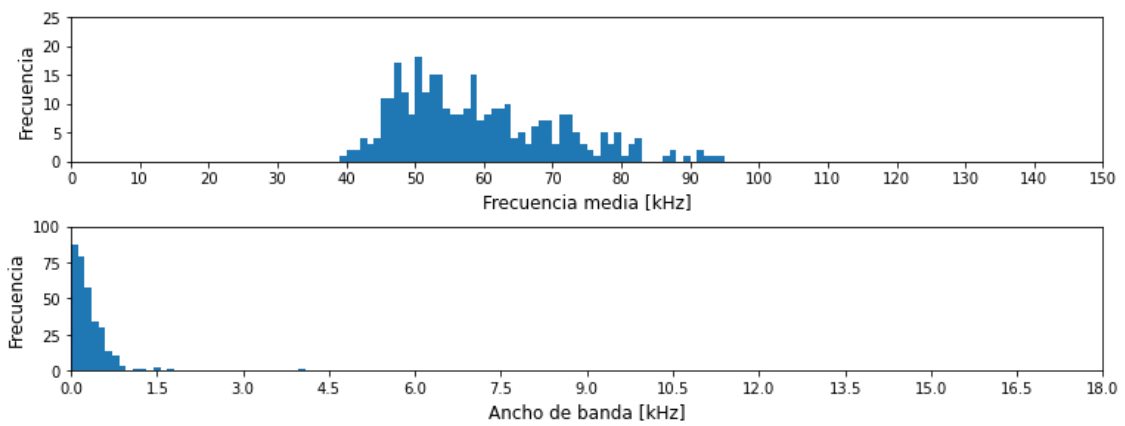
(c)

Figura 7-9 Histogramas de bandas que producen band gap. (a) estructura 1, (b) estructura 2 y (c) estructura 3

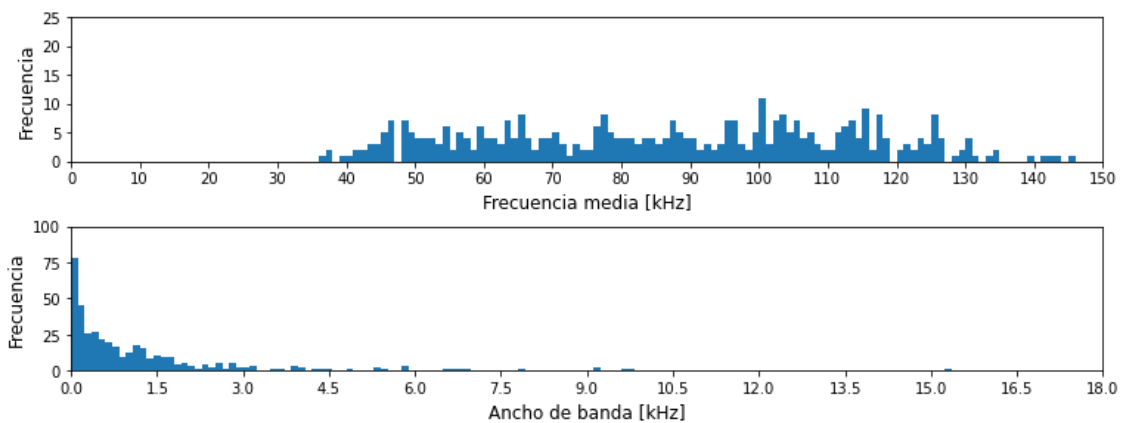
A.3 Histogramas de frecuencia media y ancho de band gaps positivos



(a)



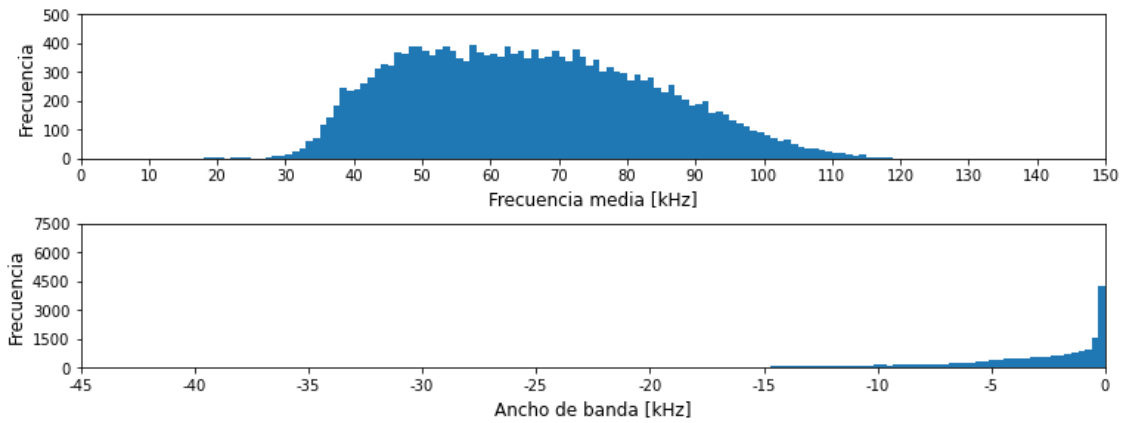
(b)



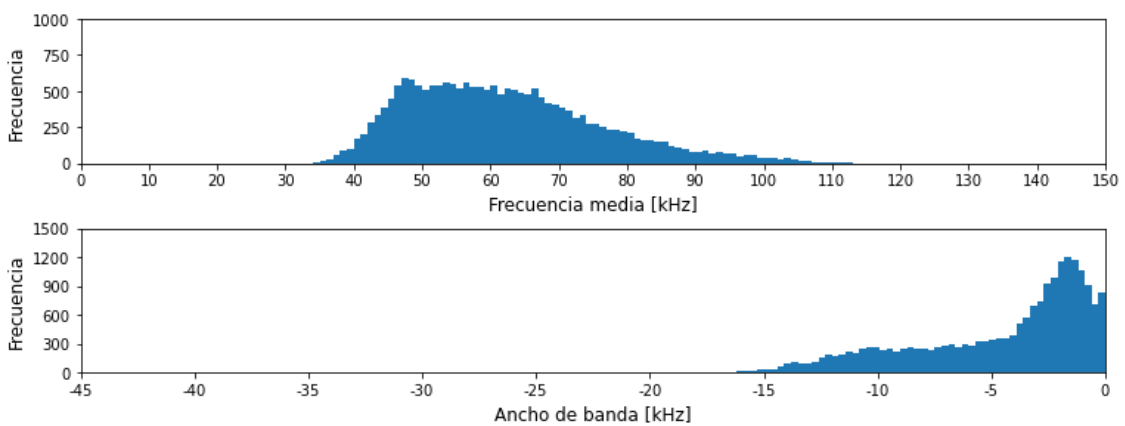
(c)

Figura 7-10 Histogramas de frecuencia media y ancho de band gaps positivos. (a) estructura 1, (b) estructura 2 y (c) estructura 3.

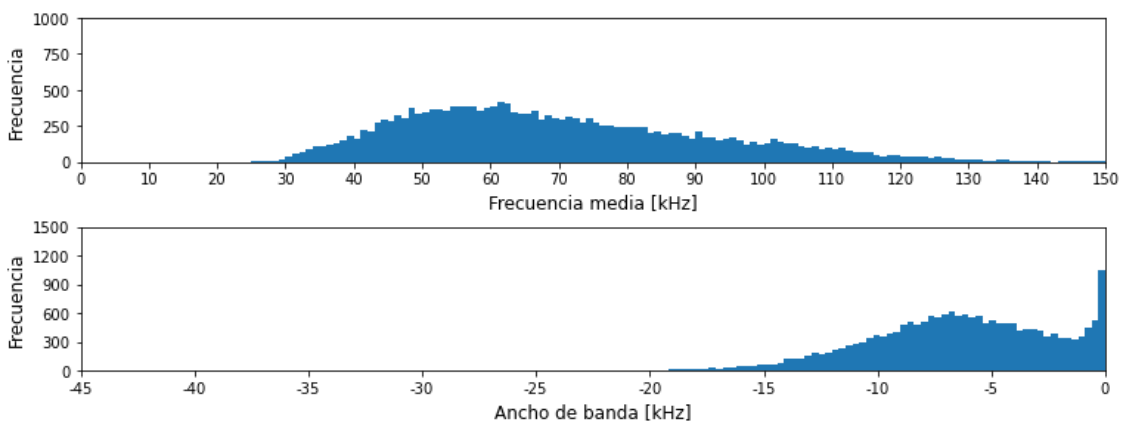
A.4 Histogramas de frecuencia media y ancho de band gaps negativos y nulos



(a)



(b)



(c)

Figura 7-11 Histogramas de frecuencia media y ancho de band gaps no-positivos. (a) estructura 1, (b) estructura 2 y (c) estructura 3.

Anexo B

Códigos

B.1 Código enfoque 1

```
import numpy as np
import pandas as pd
import time

from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.neighbors import KNeighborsRegressor
from sklearn.metrics import r2_score
from sklearn.metrics import mean_absolute_error

from toolbox.save_data import bring_data_in, save_pkl

from toolbox.calculate_bandgaps import (calculate_bandgaps_1,
                                         calculate_bandgaps_2)

from toolbox.plots_informe import (plot_confusion_2,
                                   pred_MinMax_scatt,
                                   pred_bandgap_scatt)

from toolbox.bands_processing import (get_MinMax, reduce_Bands,
                                      postprocessing_1)

##### GENERAR DATOS A PARTIR DE ARCHIVOS *.MAT

# nombre de estructura ('Estructura i, Nombre')
estructure = 'Estructura 1, Piramides'

# enfoque
version = 'enfoque_1_KNN_V1'

# prueba
gs = 'grid_search_4'

# obtener datos
X, Bands, metadata = bring_data_in(estructure, version)

# directorio donde guardar resultados
loc = metadata['directory_out']
loc2 = metadata['directory_out']+'/1_resultados/'+gs

##### PREPROCESAMIENTO

# calcular bandgaps
Bandgaps = calculate_bandgaps_1(Bands, positive_only=False)

# obtener MinMax
Bands_MinMax = get_MinMax(Bands)

# reducir bandas
obj = [0, 1, 2, 3, 4, 15]
reduced_Bands = reduce_Bands(Bands_MinMax, obj)
```

```

# convertir Bandgaps a np.array
Bandgaps = Bandgaps[['freq', 'gap']].to_numpy(na_value=0)

# separar datos con bandgaps de los sin bandgaps 0 = sin bandgap
# (De esta forma se aprovechan todos los datos con bandgap para el test)
X_0 = X[np.where(Bandgaps[:,1] <= 0)]
X_1 = X[np.where(Bandgaps[:,1] > 0)]

reduced_Bands_0 = reduced_Bands[np.where(Bandgaps[:,1] <= 0)]
reduced_Bands_1 = reduced_Bands[np.where(Bandgaps[:,1] > 0)]

# iniciar ciclos de entrenamiento y predicción
n_cycles = 10

# DataFrames para guardar resultados
resultados_MAE = pd.DataFrame(columns=['max3', 'min4', 'max4', 'min5', 'max5',
                                       'min6', 'max6', 'min7', 'max7',
                                       'min8'])

resultados_R2 = pd.DataFrame(columns=['time',
                                       'r2',
                                       'r2_freq', 'r2_freq_neg', 'r2_freq_pos',
                                       'r2_gap', 'r2_gap_neg', 'r2_gap_pos'])

for j in range(n_cycles):
    start_time = time.time()
    # data split
    train_size = 0.9
    seed = None

    X_train, X_rest, Y_train, Y_rest = train_test_split(X_0, reduced_Bands_0,
                                                         train_size=train_size,
                                                         random_state=seed)

    X_test = np.concatenate((X_rest, X_1))
    Y_test = np.concatenate((Y_rest, reduced_Bands_1))

    # aplicar Scaler
    n_bands = reduced_Bands.shape[1]

    Scalers = {}

    for i in range(n_bands):
        scaler = StandardScaler()
        Y_train[:,i,:] = scaler.fit_transform(Y_train[:,i,:])
        Scalers[i] = scaler

    # guardar dataset
    np.savez('{}data_split.npz'.format(loc),
            X_train=X_train,
            Y_train=Y_train,
            X_test=X_test,
            Y_test=Y_test)

##### IMPLEMENTAR Y ENTRENAR MODELO

# reshape al conjunto de salida
Y_train = np.reshape(Y_train, (-1, n_bands*2))

```



```

#parametros de modelo
params = {'n_neighbors': 12,
          'weights': 'distance',
          'p': 1}

# instanciar modelo
Model = KNeighborsRegressor()

# ajustar parametros
Model.set_params(**params)

# entrenar
Model.fit(X_train, Y_train)

#### EVALUACION DE MODELO

# generar prediccion
Y_pred = Model.predict(X_test)

# registrar tiempo
run_time = time.time() - start_time

# convertir a valores no-escalados
Y_pred = np.reshape(Y_pred, (-1, n_bands, 2))

for i in range(n_bands):
    scaler = Scalers[i]
    Y_pred[:,i,:] = scaler.inverse_transform(Y_pred[:,i,:])

# postprocesamiento
Y_pred = np.around(Y_pred, decimals=0)

# eliminar predicciones invalidas
Y_pred = postprocessing_1(Y_pred)

# obtener bandgaps
Bandgaps_test = calculate_bandgaps_2(Y_test,
positive_only=False).to_numpy(na_value=0)
Bandgaps_pred = calculate_bandgaps_2(Y_pred,
positive_only=False).to_numpy(na_value=0)

# R2 Bandgaps
R2 = r2_score(Bandgaps_test, Bandgaps_pred, multioutput='raw_values')

# R2 Bandgaps positivos
test_positive = Bandgaps_test[np.where(Bandgaps_test[:,1] > 0)]
pred_positive = Bandgaps_pred[np.where(Bandgaps_test[:,1] > 0)]

R2_pos = r2_score(test_positive, pred_positive, multioutput='raw_values')

# R2 Bandgaps negativos
test_negative = Bandgaps_test[np.where(Bandgaps_test[:,1] <= 0)]
pred_negative = Bandgaps_pred[np.where(Bandgaps_test[:,1] <= 0)]

R2_neg = r2_score(test_negative, pred_negative, multioutput='raw_values')

```

```

# MAE Bandas
Y_test = np.reshape(Y_test, (-1, n_bands*2))
Y_pred = np.reshape(Y_pred, (-1, n_bands*2))

MAE = mean_absolute_error(Y_test, Y_pred, multioutput='raw_values')

# guardar scores
data = {'max3': MAE[0], 'min4': MAE[1], 'max4': MAE[2], 'min5': MAE[3],
        'max5': MAE[4], 'min6': MAE[5], 'max6': MAE[6], 'min7': MAE[7],
        'max7': MAE[8], 'min8': MAE[9]}

row = pd.Series(data=data)
row = np.around(row, decimals=0)

resultados_MAE = resultados_MAE.append(row, ignore_index=True)

data = {'time': run_time, 'r2': np.mean(R2), 'r2_freq': R2[0],
        'r2_freq_neg': R2_neg[0], 'r2_freq_pos': R2_pos[0],
        'r2_gap': R2[1], 'r2_gap_neg': R2_neg[1], 'r2_gap_pos': R2_pos[1]}

row = pd.Series(data=data)
row = np.around(row, decimals=3)

resultados_R2 = resultados_R2.append(row, ignore_index=True)

# verbose
print('iteracion {} completada'.format(j+1))

# guardar ultimos Scalers
save_pkl(Scalers, 'Scalers', loc)

# guardar modelo y parametros
save_pkl(Model, 'Model', loc2)
save_pkl(params, 'Model_params', loc2)

# guardar ultima prediccion
np.savez('{} /prediction.npz'.format(loc2),
        Y_test=Y_test,
        Y_pred=Y_pred)

##### GRAFICAR

# parametros para texto en las figuras
text_params = {'fig_fontsize': 12,
               'title_fontsize': 12,
               'xlabel_fontsize': 12,
               'ylabel_fontsize': 12,
               'ticks_fontsize': 10,
               'text_fontsize': 14}

# Scater de prediccion Bandas
titles = []
band = int(len(obj)/2)

for i in range(n_bands*2):
    if i%2 == 0:
        titles.append('Maximo banda {}'.format(band))
        band += 1

```

```

else:
    titles.append('Minimo banda {}'.format(band))

figsize = (15, 5.3)
pred_MinMax_scatt((2, 5), Y_test, Y_pred, titles, metadata, figsize=figsize,
                  save=loc2, text_params=text_params)

# Scatter plot de frecuencia media y ancho de banda
figsize = (10, 5.3)
pred_bandgap_scatt(Bandgaps_test, Bandgaps_pred, metadata, figsize=figsize,
                   save=loc2, text_params=text_params)

# matriz de confusion
text_params['ticks_fontsize'] = 12

figsize = (5, 5)
plot_confusion_2(Bandgaps_test, Bandgaps_pred, metadata, figsize=figsize,
                 save=loc2, text_params=text_params)

```

B.2 Código enfoque 2

```
import numpy as np
import pandas as pd
import time

from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.neighbors import KNeighborsRegressor
from sklearn.metrics import r2_score

from toolbox.save_data import bring_data_in, save_pkl

from toolbox.calculate_bandgaps import (calculate_bandgaps_1,
                                         postprocessing_2)

from toolbox.plots_informe import (pred_bandgap_scatt,
                                    plot_confusion_2)

##### GENERAR DATOS A PARTIR DE ARCHIVOS *.MAT

# nombre de estructura ('Estructura i, Nombre')
estructura = 'Estructura 1, Piramides'

# enfoque
version = 'enfoque_3_KNN_V1'

# prueba
gs = 'grid_search_3'

# obtener datos
X, Bands, metadata = bring_data_in(estructura, version)

# directorio donde guardar resultados
loc = metadata['directory_out']
loc2 = metadata['directory_out']+'/1_resultados/'+gs
##### PREPROCESAMIENTO

# obtener bandgaps
Bandgaps = calculate_bandgaps_1(Bands, positive_only=False)

# convertir Bandgaps a np.array
Bandgaps = Bandgaps[['freq', 'gap']].to_numpy(na_value=0)

# segregar datos con bandgaps positivos de los negativos, 0 = bandgap negativo
# (Se mantiene la proporción de datos con bandgap en train y test)
X_0 = X[np.where(Bandgaps[:,1] <= 0)]
X_1 = X[np.where(Bandgaps[:,1] > 0)]

Bandgaps_0 = Bandgaps[np.where(Bandgaps[:,1] <= 0)]
Bandgaps_1 = Bandgaps[np.where(Bandgaps[:,1] > 0)]

# iniciar ciclos de entrenamiento y predicción
n_cycles = 10
```

```

# DataFrames para guardar resultados
resultados_R2 = pd.DataFrame(columns=['time',
                                     'r2',
                                     'r2_freq', 'r2_freq_neg', 'r2_freq_pos',
                                     'r2_gap', 'r2_gap_neg', 'r2_gap_pos'])

for j in range(n_cicles):
    start_time = time.time()
    # data split
    train_size = 0.9
    seed = None

    X_0train, X_0test, Y_0train, Y_0test = train_test_split(X_0, Bandgaps_0,
                                                            train_size=train_size,
                                                            random_state=seed)

    X_1train, X_1test, Y_1train, Y_1test = train_test_split(X_1, Bandgaps_1,
                                                            train_size=train_size,
                                                            random_state=seed)

    X_train = np.concatenate((X_0train, X_1train))
    Y_train = np.concatenate((Y_0train, Y_1train))

    X_test = np.concatenate((X_0test, X_1test))
    Y_test = np.concatenate((Y_0test, Y_1test))

    # aplicar scaler
    scaler = StandardScaler()
    Y_train = scaler.fit_transform(Y_train)

    # guardar dataset
    np.savez('{}data_split.npz'.format(loc),
            X_train=X_train,
            Y_train=Y_train,
            X_test=X_test,
            Y_test=Y_test)

##### IMPLEMENTAR Y ENTRENAR MODELO

# parametros de red
params = {'n_neighbors': 18,
         'weights': 'distance',
         'p': 1}

# instanciar modelo
Model = KNeighborsRegressor()

# ajustar parametros
Model.set_params(**params)

# entrenar
Model.fit(X_train, Y_train)

##### EVALUAR MODELO

# generar prediccion
Y_pred = Model.predict(X_test)

# registrar tiempo

```

```

run_time = time.time() - start_time

# convertir a valores no-escalados
Y_pred = scaler.inverse_transform(Y_pred)

# posprocesamiento
Y_pred = postprocessing_2(Y_pred, positive_only=False)

# convertir nan a 0
Y_pred = np.nan_to_num(Y_pred)

# R2 Bandgaps
R2 = r2_score(Y_test, Y_pred, multioutput='raw_values')

# R2 Bandgaps positivos
test_positive = Y_test[np.where(Y_test[:,1] > 0)]
pred_positive = Y_pred[np.where(Y_test[:,1] > 0)]

R2_pos = r2_score(test_positive, pred_positive, multioutput='raw_values')

# R2 Bandgaps negativos
test_negative = Y_test[np.where(Y_test[:,1] <= 0)]
pred_negative = Y_pred[np.where(Y_test[:,1] <= 0)]

R2_neg = r2_score(test_negative, pred_negative, multioutput='raw_values')

# guardar scores
data = {'time': run_time, 'r2': np.mean(R2), 'r2_freq': R2[0],
        'r2_freq_neg': R2_neg[0], 'r2_freq_pos': R2_pos[0],
        'r2_gap': R2[1], 'r2_gap_neg': R2_neg[1], 'r2_gap_pos': R2_pos[1]}

row = pd.Series(data=data)
row = np.around(row, decimals=3)

resultados_R2 = resultados_R2.append(row, ignore_index=True)

# verbose
print('iteracion {} completada'.format(j+1))

# guardar Scalers
save_pkl(scaler, 'Scalers', loc)

# guardar modelo y parametros
save_pkl(Model, 'Model', loc2)
save_pkl(params, 'Model_params', loc2)

# guardar prediccion
np.savez('{} /prediction.npz'.format(loc2),
        Y_test=Y_test,
        Y_pred=Y_pred)

##### GRAFICAR

# parametros para texto en las figuras
text_params = {'fig_fontsize': 12,
               'title_fontsize': 12,
               'xlabel_fontsize': 12,
               'ylabel_fontsize': 12,

```

```
        'ticks_fontsize': 10,  
        'text_fontsize': 14}  
  
# Scatter plot de frecuencia media y ancho de banda  
figsize = (10, 5.3)  
pred_bandgap_scatt(Y_test, Y_pred, metadata, figsize=figsize,  
                  save=loc2, text_params=text_params)  
  
# matriz de confusion  
text_params['ticks_fontsize'] = 12  
  
figsize = (5, 5)  
plot_confusion_2(Y_test, Y_pred, metadata, figsize=figsize,  
                save=loc2, text_params=text_params)
```

B.3 Código variación proporción de band gaps

```
import numpy as np
import pandas as pd

from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.neural_network import MLPRegressor
from sklearn.metrics import r2_score

from toolbox.save_data import bring_data_in

from toolbox.bands_processing import delete_samples

from toolbox.calculate_bandgaps import (calculate_bandgaps_1,
                                         postprocessing_2)

# nombre de estructura ('Estructura i, Nombre')
estructure = 'Estructura 2, Kagome'

resultados = pd.DataFrame(columns=['r', 'r2',
                                   'r2_freq', 'r2_freq_neg', 'r2_freq_pos',
                                   'r2_gap', 'r2_gap_neg', 'r2_gap_pos'])

n_bg = 319

for j in range(5):
    print('Iniciando ciclo {}/5'.format(j+1))

    for i in np.arange(0, 15500, 500, dtype=int):
        # obtener datos
        X, Bands, metadata = bring_data_in(estructure, '', save=False)

        n_samples = X.shape[0] - i

        r = n_bg/n_samples

        ##### PREPOCESAMIENTO

        # obtener bandgaps
        Bandgaps = calculate_bandgaps_1(Bands, positive_only=False)

        # convertir Bandgaps a np.array
        Bandgaps = Bandgaps[['freq', 'gap']].to_numpy(na_value=0)

        # eliminar samples
        X, Bands, Bandgaps = delete_samples(X, Bands, Bandgaps, i)

        # segregar datos con bandgaps positivos de los negativos, 0 = bandgap
        negativo
        # (Se mantiene la proporción de datos con bandgap en train y test)
        X_0 = X[np.where(Bandgaps[:,1] <= 0)]
        X_1 = X[np.where(Bandgaps[:,1] > 0)]

        Bandgaps_0 = Bandgaps[np.where(Bandgaps[:,1] <= 0)]
```



```

Bandgaps_1 = Bandgaps[np.where(Bandgaps[:,1] > 0)]

# data split
train_size = 0.9
seed = None

X_0train, X_0test, Y_0train, Y_0test = train_test_split(X_0,
Bandgaps_0,
train_size=train_size,
random_state=seed)

X_1train, X_1test, Y_1train, Y_1test = train_test_split(X_1,
Bandgaps_1,
train_size=train_size,
random_state=seed)

X_train = np.concatenate((X_0train, X_1train))
Y_train = np.concatenate((Y_0train, Y_1train))

X_test = np.concatenate((X_0test, X_1test))
Y_test = np.concatenate((Y_0test, Y_1test))

# aplicar scaler
scaler = StandardScaler()
Y_train = scaler.fit_transform(Y_train)

##### IMPLEMENTAR Y ENTRENAR MODELO

# parametros de red
params = {'hidden_layer_sizes': (128, 256, 512),
'activation': 'relu',
'solver': 'adam',
'learning_rate': 'constant',
'learning_rate_init': 0.001,
'max_iter': 5000,
'tol': 1e-3,
'early_stopping': True,
'n_iter_no_change': 50,
'verbose': False}

# instanciar modelo
Model = MLPRegressor(random_state=seed)

# ajustar parametros
Model.set_params(**params)

# entrenar
Model.fit(X_train, Y_train)

##### EVALUAR MODELO

# generar prediccion
Y_pred = Model.predict(X_test)

```

```

# convertir a valores no-escalados
Y_pred = scaler.inverse_transform(Y_pred)

# postprocesamiento
Y_pred = postprocessing_2(Y_pred, positive_only=False)

# convertir nan a 0
Y_pred = np.nan_to_num(Y_pred)

# R2 Bandgaps
R2 = r2_score(Y_test, Y_pred, multioutput='raw_values')

# R2 Bandgaps positivos
test_positive = Y_test[np.where(Y_test[:,1] > 0)]
pred_positive = Y_pred[np.where(Y_test[:,1] > 0)]

R2_pos = r2_score(test_positive, pred_positive,
multioutput='raw_values')

# R2 Bandgaps negativos
test_negative = Y_test[np.where(Y_test[:,1] <= 0)]
pred_negative = Y_pred[np.where(Y_test[:,1] <= 0)]

R2_neg = r2_score(test_negative, pred_negative,
multioutput='raw_values')

# guardar scores
data = {'r': r, 'r2': np.mean(R2), 'r2_freq': R2[0],
        'r2_freq_neg': R2_neg[0], 'r2_freq_pos': R2_pos[0],
        'r2_gap': R2[1], 'r2_gap_neg': R2_neg[1], 'r2_gap_pos':
R2_pos[1]}

row = pd.Series(data=data)
row = np.around(row, decimals=3)

resultados = resultados.append(row, ignore_index=True)

#verbose

print('n_samples: {}'.format(n_samples))

```

Anexo C

Hiperparámetros seleccionados

En este Anexo se dan a conocer los parámetros con los que son configurados los diferentes modelos de aprendizaje de máquinas implementados en esta memoria.

Los parámetros que no se indican en las tablas son dejados en sus valores preestablecidos según la documentación de scikit-learn, exceptuando los parámetros para las RNN que se indican en la Tabla 7-1.

Tabla 7-1 Hiperparámetros comunes a todos los modelos RNN.

Hiperparámetros	Valor
max_iter	5000
tol	0.003
early_stopping	True
n_iter_no_change	50

C.1 Hiperparámetros del enfoque 1

Tabla 7-2 Hiperparámetros empleados en modelos KNN para el enfoque 1.

	Estructura 1	Estructura 2	Estructura 3
n_neighbors	12	12	10
weights	distance	distance	distance
p	1	1	1

Tabla 7-3 Hiperparámetros empleados en modelos RNN para el enfoque 1.

	Estructura 1	Estructura 2	Estructura 3
hidden_layer_sizes	(512, 512, 128)	(512, 512)	(512, 512, 256)
activation	relu	relu	relu
solver	adam	adam	adam
learning_rate_init	0.001	0.001	0.001

C.2 Hiperparámetros del enfoque 2

Tabla 7-4 Hiperparámetros empleados en modelos KNN para el enfoque 3.

	Estructura 1	Estructura 2	Estructura 3
n_neighbors	18	13	11
weights	distance	distance	distance
p	1	3	2

Tabla 7-5 Hiperparámetros empleados en modelos RNN para el enfoque 3.

	Estructura 1	Estructura 2	Estructura 3
hidden_layer_sizes	(512, 512, 128, 128)	(128, 256, 512)	(512, 512, 64, 512, 512)
activation	relu	relu	relu
solver	adam	adam	adam
learning_rate_init	0.001	0.001	

Anexo D

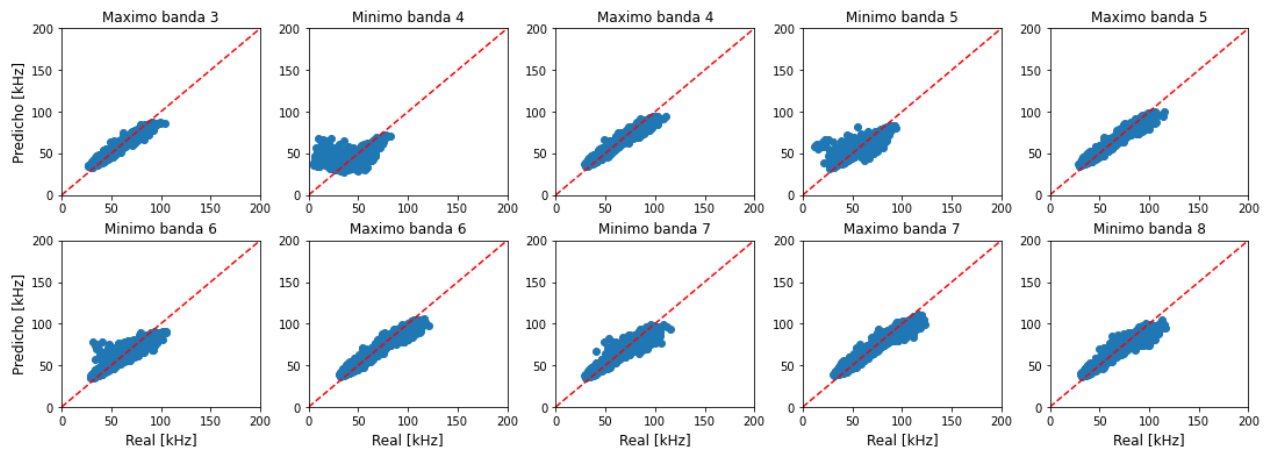
Gráficos de los resultados

A continuación, se presentan todos los gráficos de dispersión y matrices de confusión generados a partir de las predicciones hechas para todas las estructuras, empleando modelos de KNN y RNN.

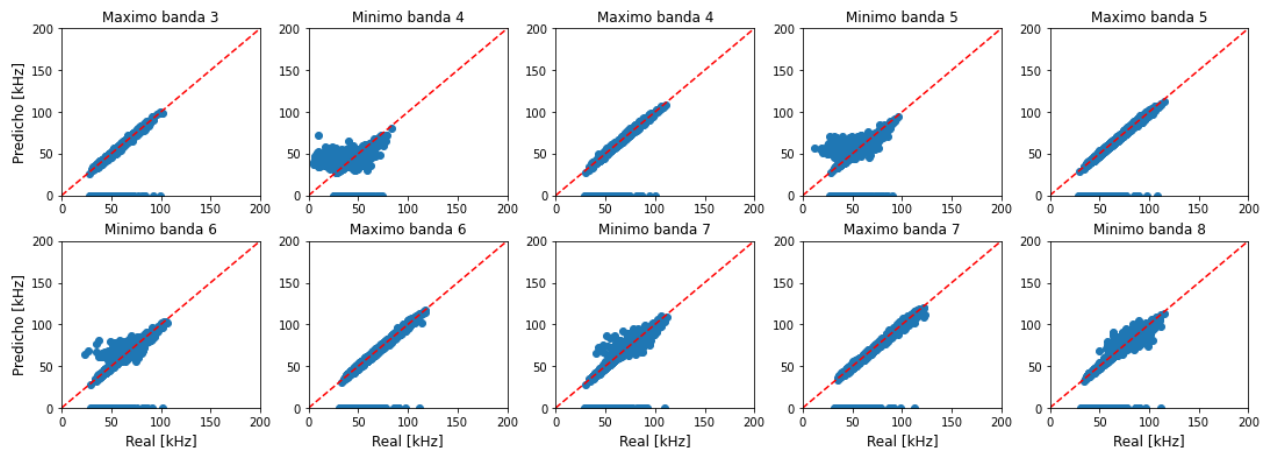
En las siguientes secciones, se presentan los resultados obtenidos correspondientes a cada uno de los enfoques.

D.1 Enfoque 1

Estructura 1

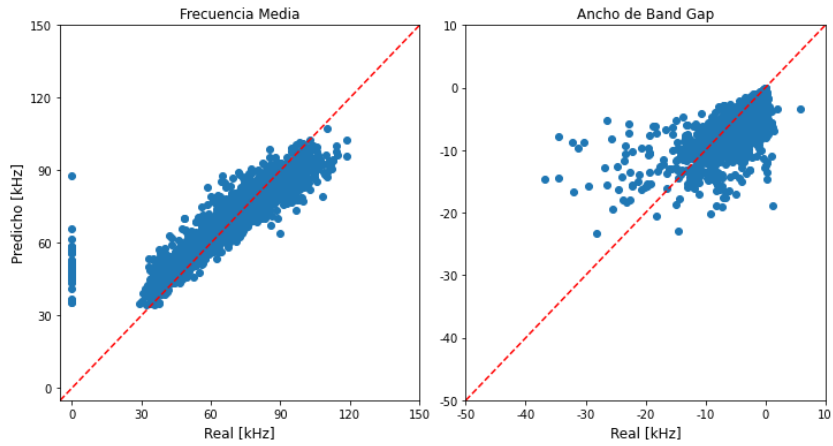


(a)

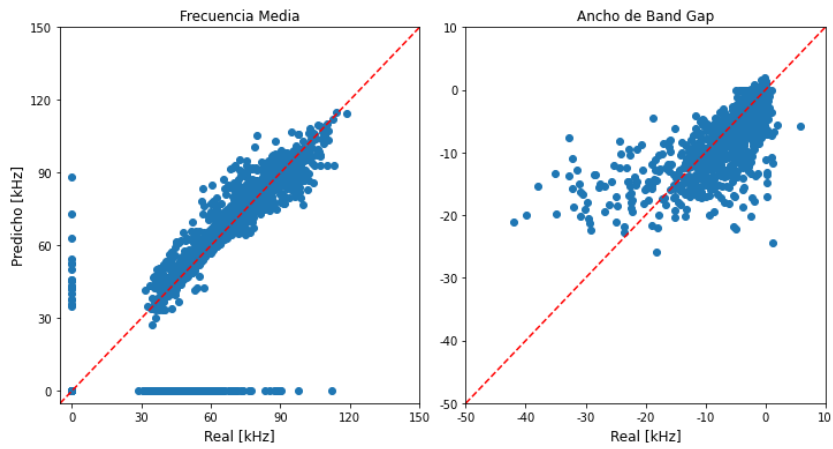


(b)

Figura 7-12 Gráficos de dispersión de la predicción de los valores mínimos y máximos de las bandas. Estructura 1. **(a)** predicción realizada con KNN y **(b)** predicción realizada con RNN.



(a)



(b)

Figura 7-13 Gráficos de dispersión de la frecuencia media y ancho de band gap, calculados a partir de los valores mínimos y máximos predichos para la estructura 1. **(a)** predicción realizada con KNN y **(b)** predicción realizada con RNN.

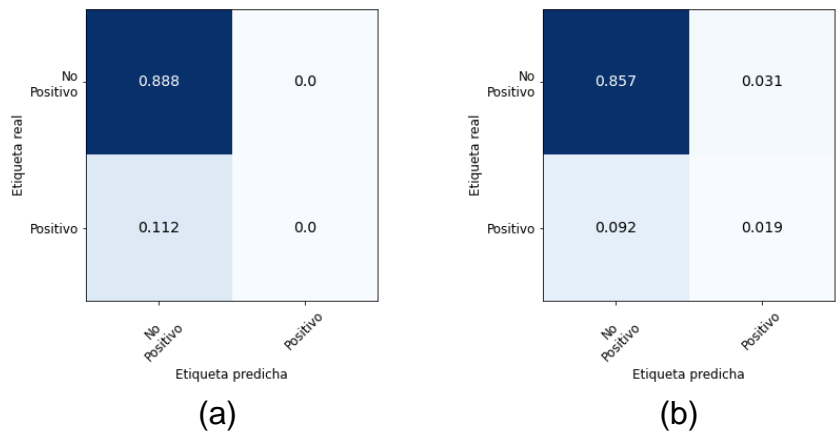
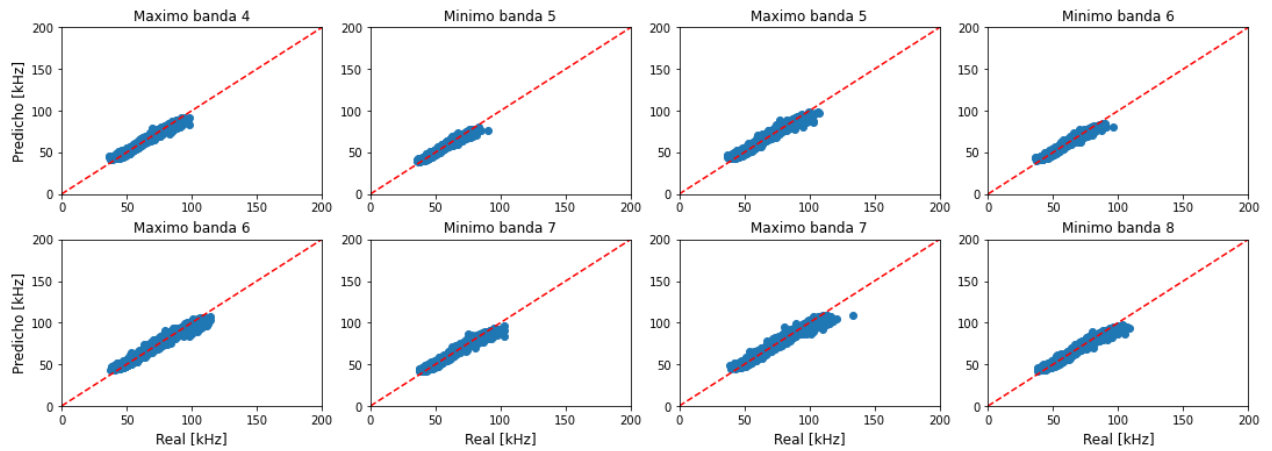
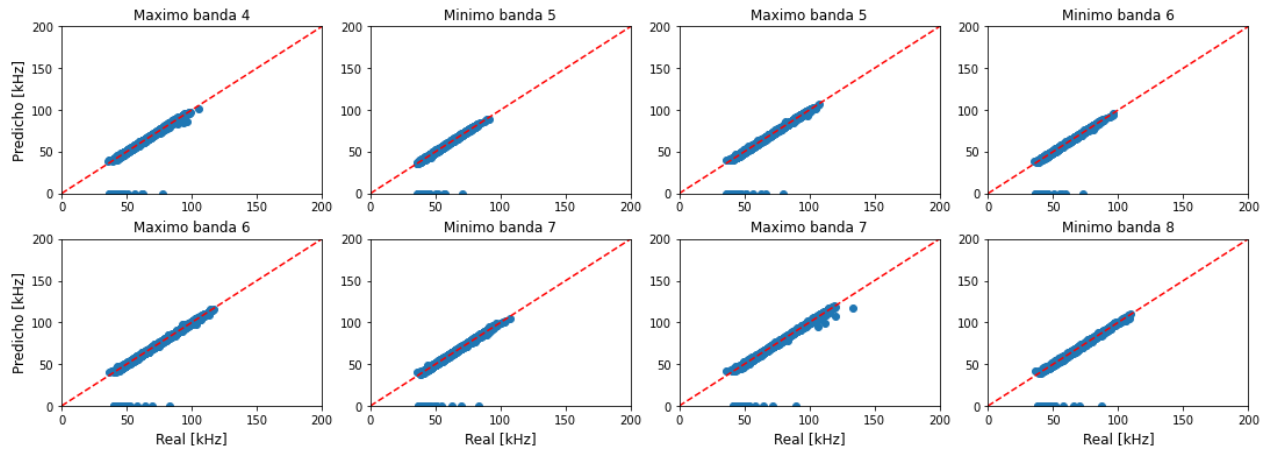


Figura 7-14 Matriz de confusión del cálculo de band gaps en estructura 1. (a) predicción realizada con KNN y (b) predicción realizada con RNN.

Estructura 2



(a)



(b)

Figura 7-15 Gráficos de dispersión de la predicción de los valores mínimos y máximos de las bandas. Estructura 2. **(a)** predicción realizada con KNN y **(b)** predicción realizada con RNN.

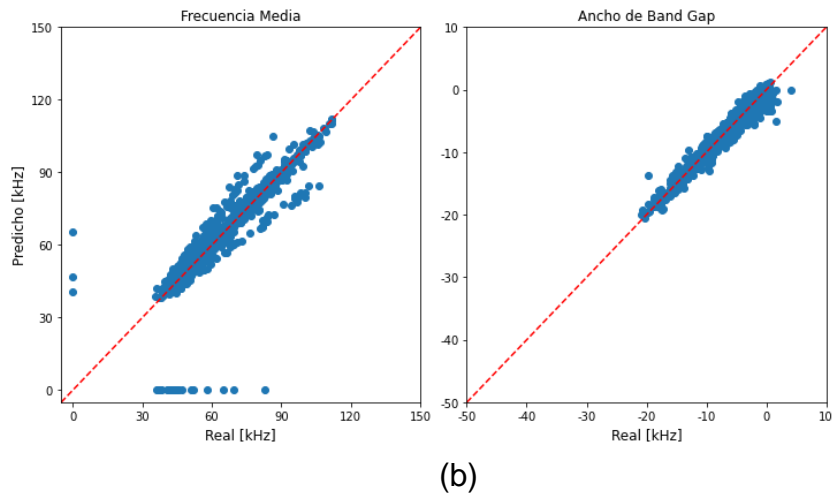
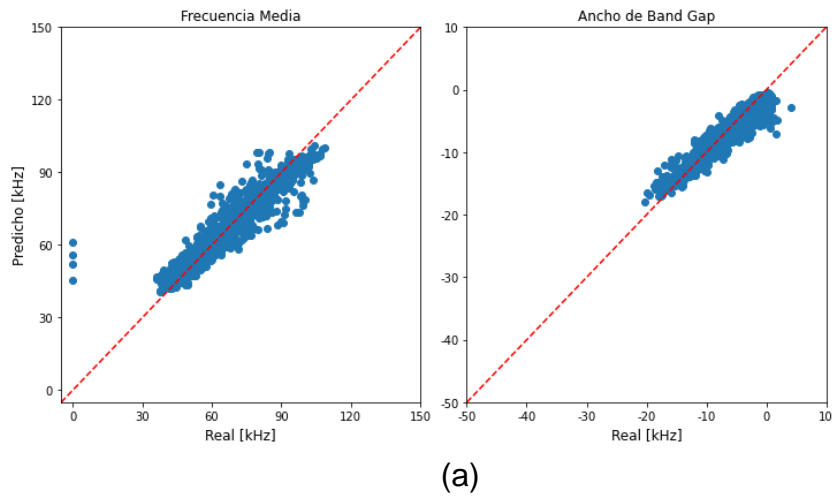


Figura 7-16 Gráficos de dispersión de la frecuencia media y ancho de band gap, calculados a partir de los valores mínimos y máximos predichos para la estructura 2. **(a)** predicción realizada con KNN y **(b)** predicción realizada con RNN.

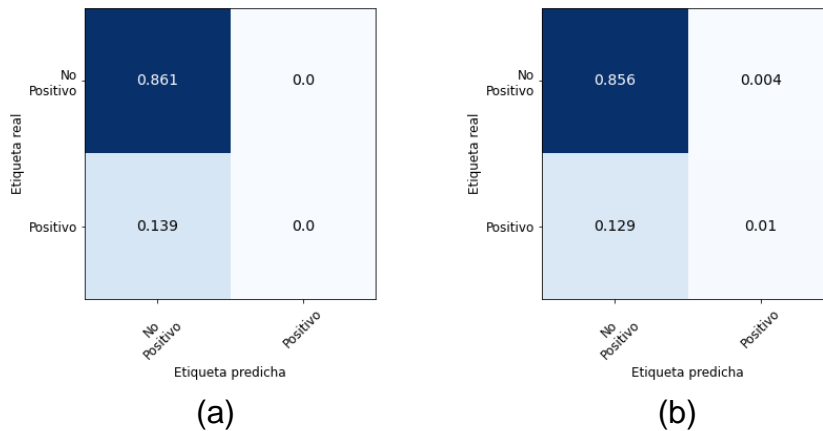
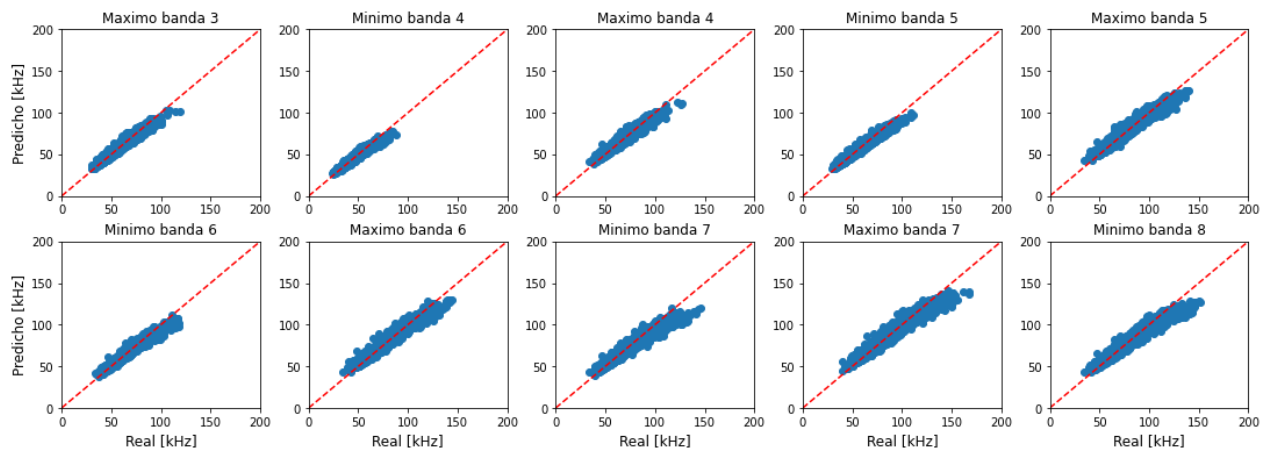
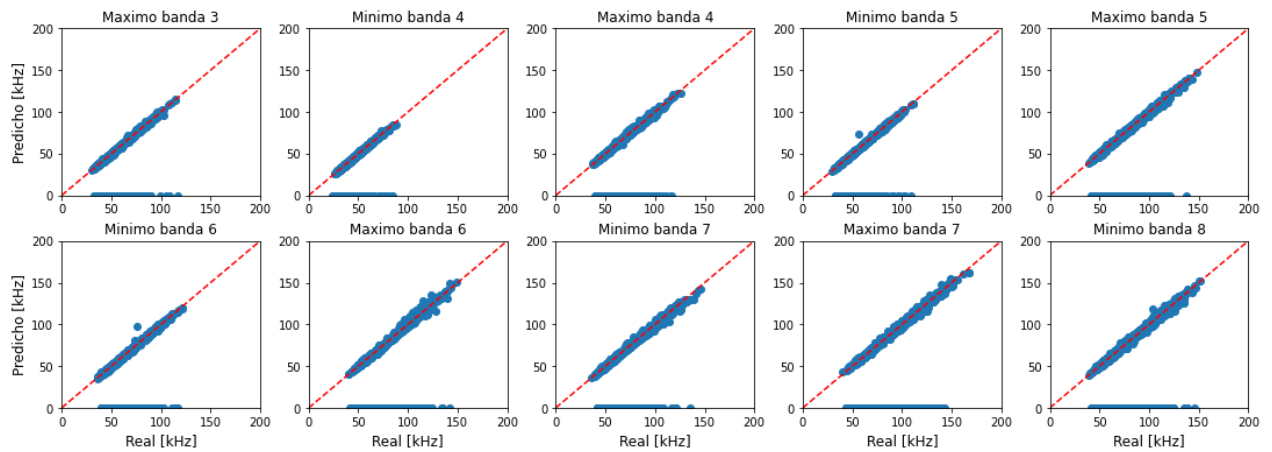


Figura 7-17 Matriz de confusión del cálculo de band gaps en estructura 2. **(a)** predicción realizada con KNN y **(b)** predicción realizada con RNN.

Estructura 3



(a)



(b)

Figura 7-18 Gráficos de dispersión de la predicción de los valores mínimos y máximos de las bandas. Estructura 3. **(a)** predicción realizada con KNN y **(b)** predicción realizada con RNN.

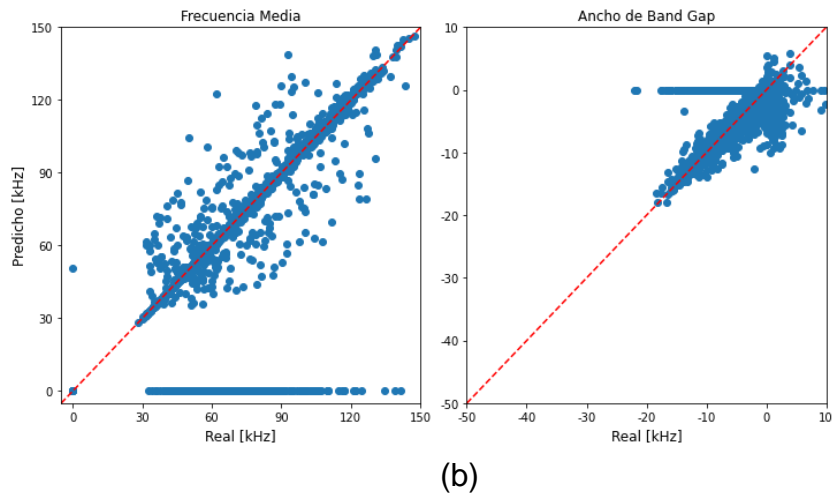
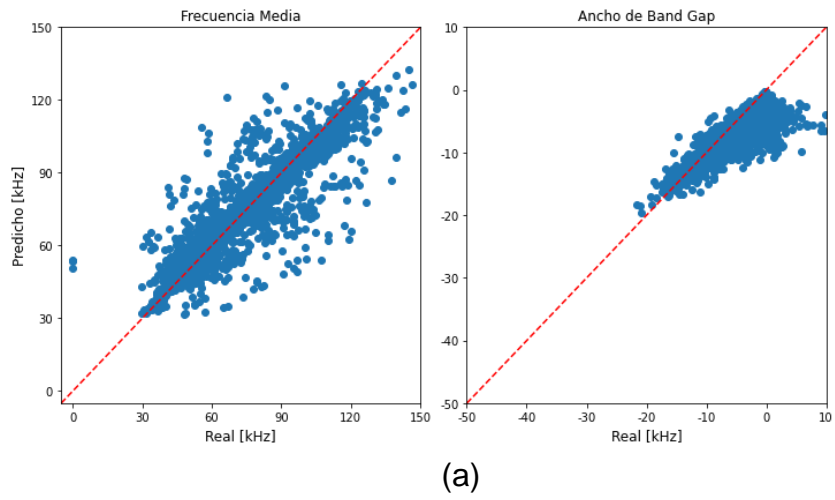


Figura 7-19 Gráficos de dispersión de la frecuencia media y ancho de band gap, calculados a partir de los valores mínimos y máximos predichos para la estructura 3. **(a)** predicción realizada con KNN y **(b)** predicción realizada con RNN.

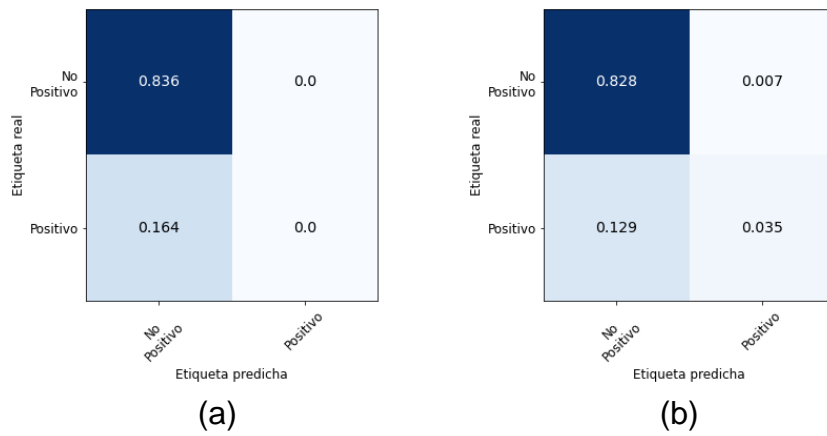
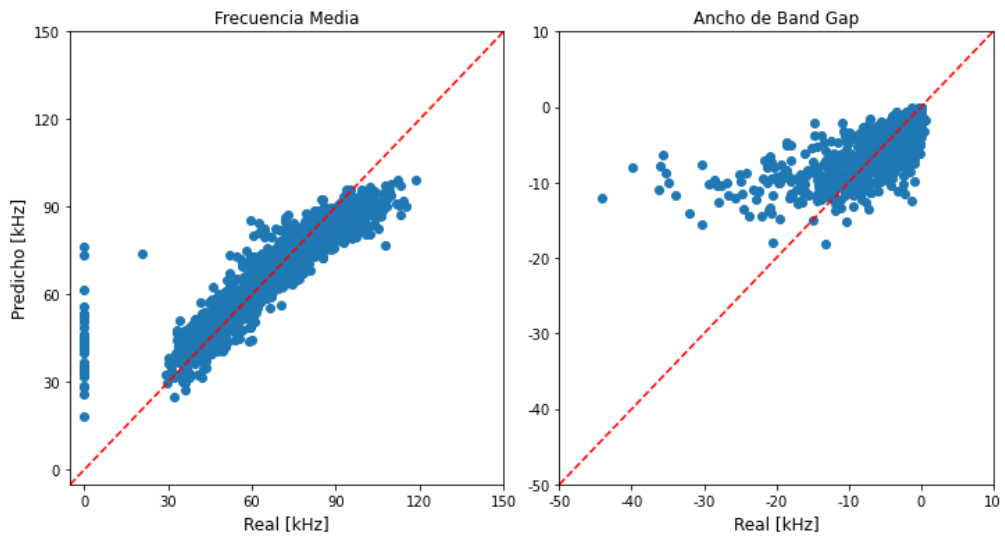


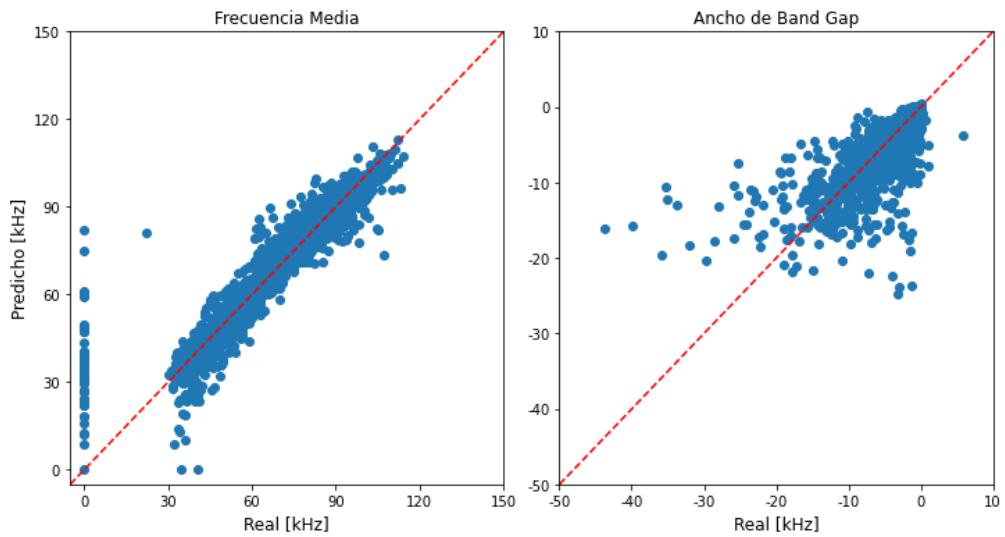
Figura 7-20 Matriz de confusión del cálculo de band gaps en estructura 3. (a) predicción realizada con KNN y (b) predicción realizada con RNN.

D.2 Enfoque 2

Estructura 1

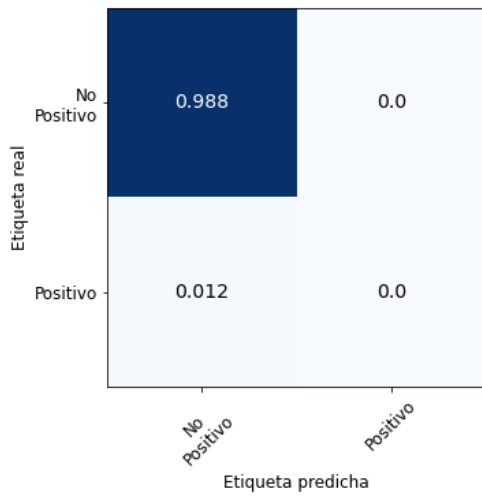


(a)

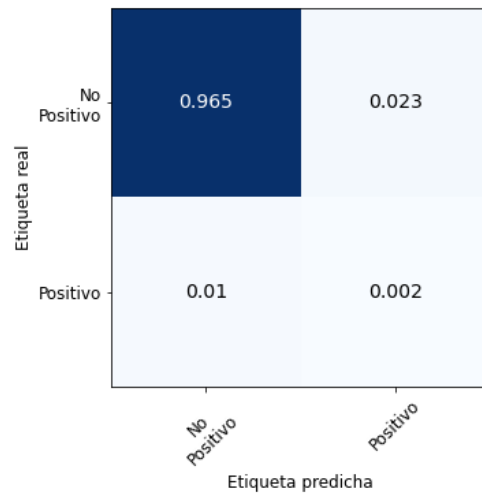


(b)

Figura 7-21 Gráficos de dispersión de la predicción de frecuencia media y ancho de banda para la estructura 1. **(a)** predicción realizada con KNN y **(b)** predicción realizada con RNN.



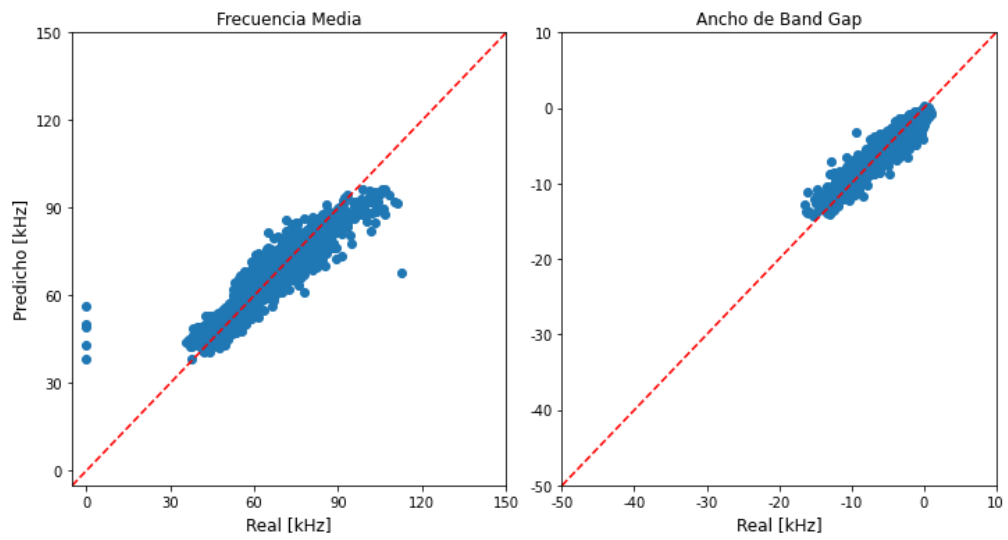
(a)



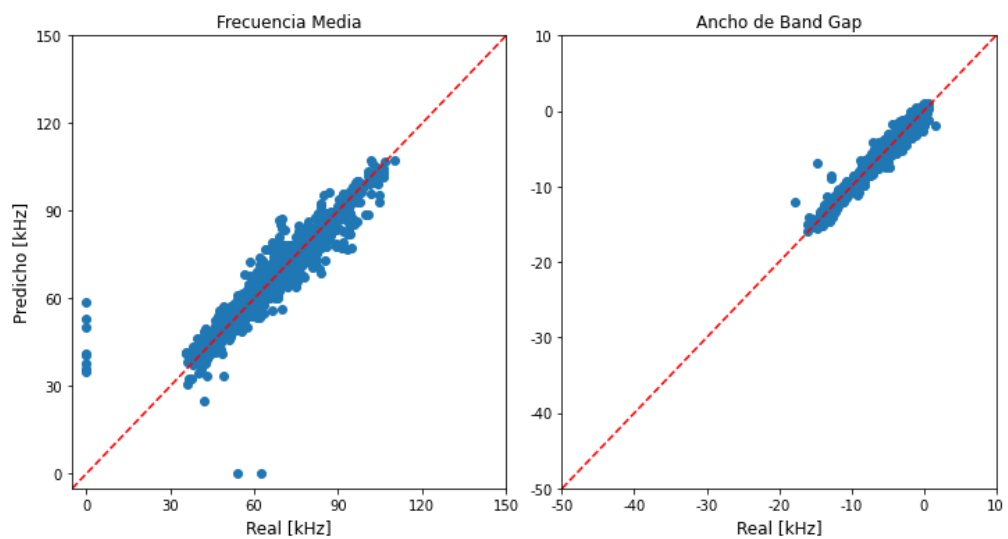
(b)

Figura 7-22 Matriz de confusión de la predicción de band gaps en la estructura 1. **(a)** predicción realizada con KNN y **(b)** predicción realizada con RNN.

Estructura 2

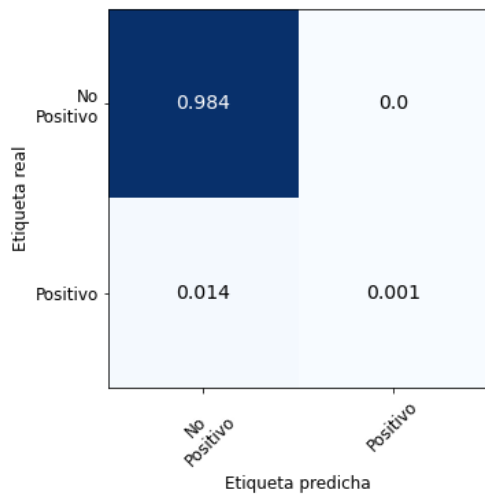


(a)

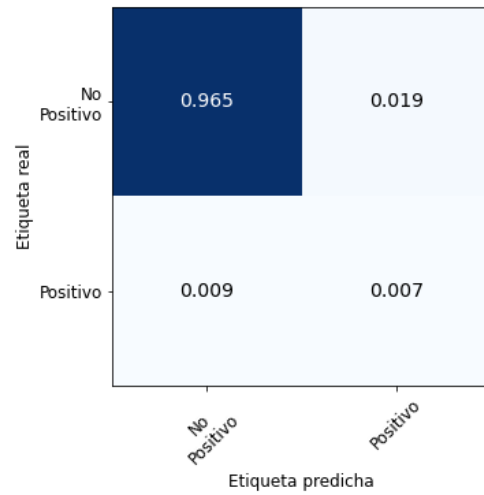


(b)

Figura 7-23 Gráficos de dispersión de la predicción de frecuencia media y ancho de banda para la estructura 2. **(a)** predicción realizada con KNN y **(b)** predicción realizada con RNN.



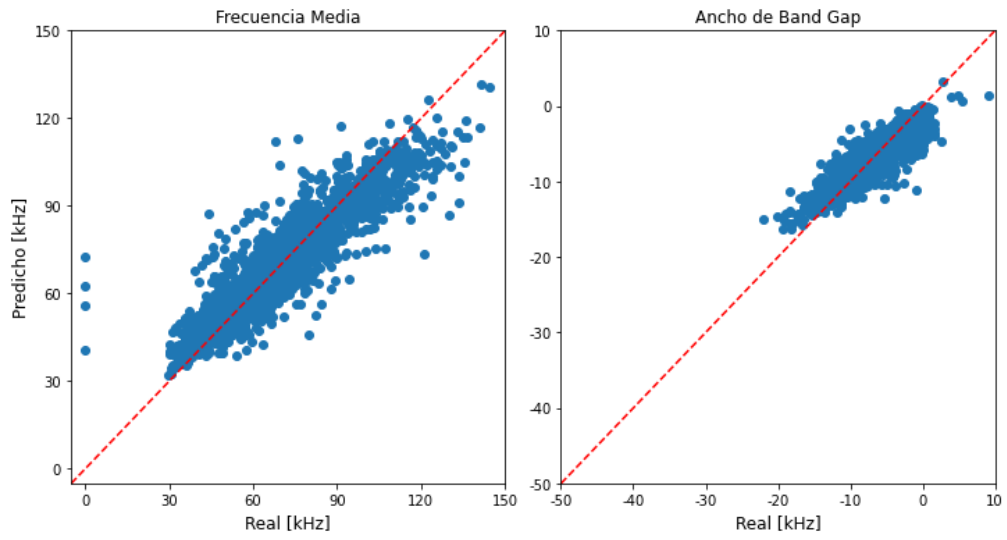
(a)



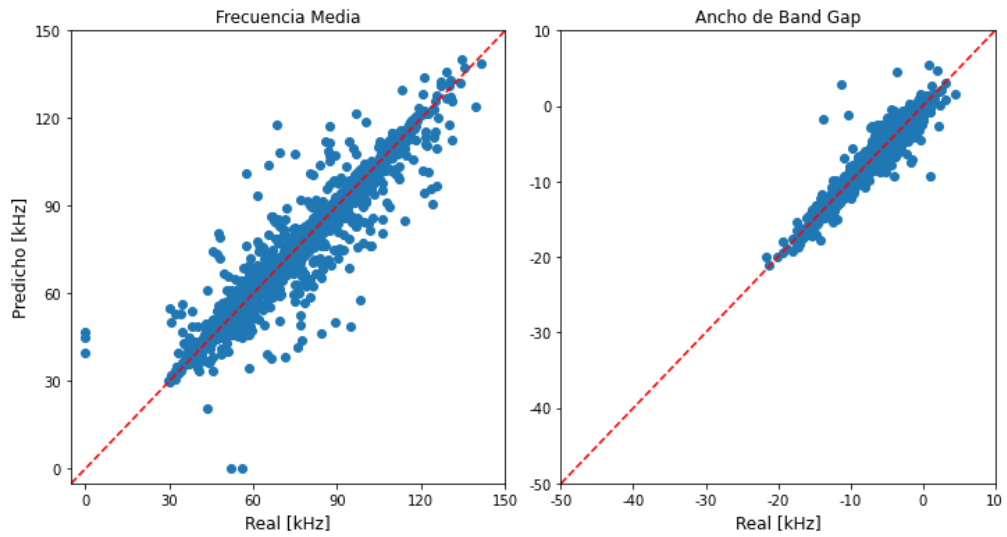
(b)

Figura 7-24 Matriz de confusión de la predicción de band gaps en la estructura 2. **(a)** predicción realizada con KNN y **(b)** predicción realizada con RNN.

Estructura 3

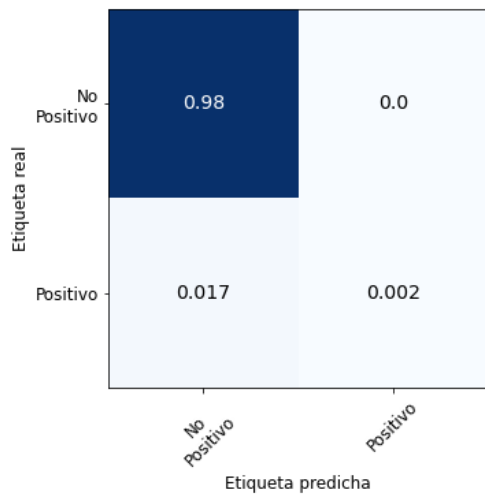


(a)

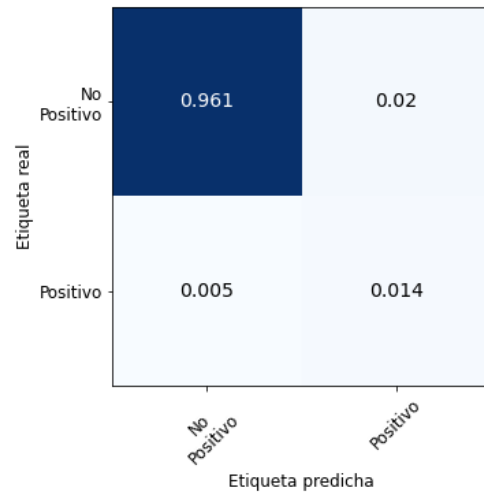


(b)

Figura 7-25 Gráficos de dispersión de la predicción de frecuencia media y ancho de banda para la estructura 3. **(a)** predicción realizada con KNN y **(b)** predicción realizada con RNN.



(a)



(b)

Figura 7-26 Matriz de confusión de la predicción de band gaps en la estructura 3. **(a)** predicción realizada con KNN y **(b)** predicción realizada con RNN.

D.3 R^2 y MAE

A continuación, se entregan todos los valores del coeficiente de determinación y error absoluto medio obtenidos para todas las predicciones.

Enfoque 1

En la Tabla 7-6, Tabla 7-7 y Tabla 7-8 se entregan los valores de MAE obtenidos de la predicción de los valores mínimos y máximos de bandas, para las estructuras 1, 2 y 3 respectivamente.

Tabla 7-6 MAE de la predicción de valores mínimos y máximos de bandas para la estructura 1.

	Predicción KNN		Predicción RNN	
	μ [Hz]	σ [Hz]	μ [Hz]	σ [Hz]
MAE_{\max} banda 3	2.909	34	6.488	1.824
MAE_{\min} banda 4	4.802	148	8.872	1.629
MAE_{\max} banda 4	3.181	52	6.816	1.908
MAE_{\min} banda 5	3.728	103	7.883	1.840
MAE_{\max} banda 5	3.518	58	7.014	1.961
MAE_{\min} banda 6	3.658	69	7.726	1.879
MAE_{\max} banda 6	3.886	57	7.165	2.011
MAE_{\min} banda 7	4.049	76	8.115	1.967
MAE_{\max} banda 7	4.218	68	7.378	2.066
MAE_{\min} banda 8	4.102	77	8.034	1.989

Tabla 7-7 MAE de la predicción de valores mínimos y máximos de bandas para la estructura 2.

	Predicción KNN		Predicción RNN	
	μ [Hz]	σ [Hz]	μ [Hz]	σ [Hz]
MAE_{\max} banda 4	1.901	41	1.239	245
MAE_{\min} banda 5	1.492	25	1.060	227
MAE_{\max} banda 5	2.148	45	1.247	250
MAE_{\min} banda 6	1.596	32	1.066	226
MAE_{\max} banda 6	2.448	49	1.308	280
MAE_{\min} banda 7	1.887	36	1.227	244
MAE_{\max} banda 7	2.492	63	1.399	272
MAE_{\min} banda 8	1.986	36	1.308	245

Tabla 7-8 *MAE* de la predicción de valores mínimos y máximos de bandas para la estructura 3.

	Predicción KNN		Predicción RNN	
	μ [Hz]	σ [Hz]	μ [Hz]	σ [Hz]
$MAE_{\max_banda_3}$	2.170	27	15.093	2.636
$MAE_{\min_banda_4}$	1.722	24	11.485	1.929
$MAE_{\max_banda_4}$	2.438	38	17.046	2.969
$MAE_{\min_banda_5}$	2.107	32	14.422	2.488
$MAE_{\max_banda_5}$	2.959	51	19.458	3.496
$MAE_{\min_banda_6}$	2.372	40	16.202	2.816
$MAE_{\max_banda_6}$	2.997	51	19.938	3.479
$MAE_{\min_banda_7}$	3.173	34	17.459	2.988
$MAE_{\max_banda_7}$	3.771	58	21.621	3.828
$MAE_{\min_banda_8}$	3.548	62	19.404	3.347

En la Tabla 7-9,

Tabla 7-10 y Tabla 7-11 se entrega los R^2 obtenido al calcular la frecuencia y ancho de band gap a partir de los valores mínimos y máximos predichos, para las estructuras 1, 2 y 3 respectivamente.

Tabla 7-9 R^2 del cálculo de frecuencia media y ancho de band gap para la estructura 1.

	Predicción KNN		Predicción RNN	
	μ	σ	μ	σ
$R^2_{bandgap}$	0,665	0,010	0,369	0,116
R^2_{freq}	0,786	0,016	0,208	0,228
$R^2_{freq-neg}$	0,773	0,017	0,277	0,212
$R^2_{freq-pos}$	0,835	0,005	-0,723	0,485
R^2_{gap}	0,545	0,010	0,530	0,077
$R^2_{gap-neg}$	0,568	0,008	0,561	0,072
$R^2_{gap-pos}$	-57,465	1,823	-64,402	9,153

Tabla 7-10 R^2 del cálculo de frecuencia media y ancho de band gap para la estructura 2.

	Predicción KNN		Predicción RNN	
	μ	σ	μ	σ
$R^2_{bandgap}$	0,888	0,007	0,857	0,041
R^2_{freq}	0,896	0,015	0,764	0,071
$R^2_{freq-neg}$	0,893	0,016	0,802	0,077
$R^2_{freq-pos}$	0,916	0,010	0,358	0,288
R^2_{gap}	0,879	0,004	0,948	0,012
$R^2_{gap-neg}$	0,903	0,005	0,957	0,013
$R^2_{gap-pos}$	-47,167	0,724	-17,830	3,378

Tabla 7-11 R^2 del cálculo de frecuencia media y ancho de band gap para la estructura 3.

	Predicción KNN		Predicción RNN	
	μ	σ	μ	σ
$R^2_{bandgap}$	0,663	0,006	-0,563	0,278
R^2_{freq}	0,811	0,012	-1,329	0,403
$R^2_{freq-neg}$	0,790	0,016	-1,759	0,572
$R^2_{freq-pos}$	0,821	0,009	-0,596	0,400
R^2_{gap}	0,515	0,012	0,202	0,163
$R^2_{gap-neg}$	0,701	0,014	-0,139	0,268
$R^2_{gap-pos}$	-12,156	0,173	-3,192	0,717

Enfoque 2

En la Tabla 7-12, Tabla 7-13 y

Tabla 7-14 se entregan los valores de R^2 obtenidos para la predicción de frecuencia media y ancho de band gap, para las estructuras 1, 2 y 3 respectivamente, utilizando el enfoque 2.

Tabla 7-12 R^2 de la predicción de frecuencia media y ancho de band gap para la estructura 1. Enfoque 2.

	Predicción KNN		Predicción RNN	
	μ	σ	μ	σ
$R^2_{bandgap}$	0,699	0,010	0,755	0,016
R^2_{freq}	0,806	0,015	0,858	0,014
$R^2_{freq-neg}$	0,805	0,015	0,858	0,014
$R^2_{freq-pos}$	0,824	0,065	0,816	0,137
R^2_{gap}	0,592	0,012	0,652	0,029
$R^2_{gap-neg}$	0,592	0,010	0,653	0,030
$R^2_{gap-pos}$	-164,991	339,172	-128,663	157,864

Tabla 7-13 R^2 de la predicción de frecuencia media y ancho de band gap para la estructura 2. Enfoque 2.

	Predicción KNN		Predicción RNN	
	μ	σ	μ	σ
$R^2_{bandgap}$	0,889	0,008	0,942	0,007
R^2_{freq}	0,871	0,014	0,915	0,012
$R^2_{freq-neg}$	0,872	0,012	0,916	0,013
$R^2_{freq-pos}$	0,743	0,276	0,783	0,304
R^2_{gap}	0,907	0,005	0,970	0,002
$R^2_{gap-neg}$	0,908	0,005	0,970	0,002
$R^2_{gap-pos}$	-41,106	14,961	-7,975	5,923

Tabla 7-14 R^2 de la predicción de frecuencia media y ancho de band gap para la estructura 3. Enfoque 2.

	Predicción KNN		Predicción RNN	
	μ	σ	μ	σ
$R^2_{bandgap}$	0,813	0,008	0,913	0,008
R^2_{freq}	0,830	0,010	0,897	0,015
$R^2_{freq-neg}$	0,828	0,010	0,895	0,015
$R^2_{freq-pos}$	0,850	0,041	0,924	0,040
R^2_{gap}	0,795	0,016	0,929	0,006
$R^2_{gap-neg}$	0,793	0,014	0,929	0,007
$R^2_{gap-pos}$	-2,845	1,862	-1,838	1,963