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

ENTRENAMIENTO DE ALGORITMOS DE APRENDIZAJE DE MÁQUINAS PARA PREDECIR LOS BAND GAPS EN PANELES DE METAMATERIALES

MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL MECÁNICO

RICARDO ANDRÉS JADUE ABUAUAD

PROFESORA GUÍA:
VIVIANA MERUANE NARANJO

MIEMBROS DE LA COMISIÓN:
RAFAEL RUIZ GARCIA
RUBÉN FERNÁNDEZ URRUTIA

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

SANTIAGO DE CHILE
2021

RESUMEN DE LA MEMORIA PARA OPTAR
AL TÍTULO DE INGENIERO CIVIL MECÁNICO
POR: RICARDO ANDRÉS JADUE ABUAUAD
FECHA: 26 DE JULIO, 2021
PROFESORA GUÍA: VIVIANA MERUANE NARANJO

ENTRENAMIENTO DE ALGORITMOS DE APRENDIZAJE DE MÁQUINAS PARA PREDECIR LOS BAND GAPS EN PANELES DE METAMATERIALES

Los metamateriales son un tipo de materiales compuestos cuyas propiedades dependen de la topología en la que se encuentran sus componentes dentro de la estructura. Una de las propiedades que estos materiales pueden presentan los band gaps, los cuales son supresiones de ondas mecánicas en ciertos rangos de frecuencias. Esta propiedad que exhiben algunos metamateriales a sido la raíz de porque estos han sido objeto de estudio en el último tiempo, ya que ha partir de ellos se han podido desarrollar filtros acústicos, protectores de vibraciones y en algunos casos para recolección de energía. Cabe destacar que modelar de manera numérica los metamateriales para luego calcular las bandas de frecuencia toma bastante tiempo por lo que se plantea este trabajo de investigación como una posible solución en cuanto a la eficiencia con respecto al tiempo que toma este proceso.

Debido a que generar los band gaps mediante modelos numérico resulta poco eficiente en cuanto a tiempo de computación, se propone utilizar algoritmos de machine learning para predecir los anchos de banda. Para poder realizar esto se utilizan diferentes algoritmos los cuales son: Support Vector Machines, K Nearest Neighbors, Random Forest, Adaptive Boosting y Gradient Boositng. Estos algoritmos se entrenan utilizando 3 bases de datos diferentes los cuales fueron generados a partir de una panel de metamaterial tipo enrejado de vigas interconectadas, una estructura periódica con resonadores y una combinación de los dos anteriores el cual fue una enrejado con masas puntuales en las vigas. Además de esto se utilizaron dos enfoques diferentes en la metodología, el primer enfoque fue centrado en predecir directamente el ancho de banda y la frecuencia media utilizando las tres bases de datos para cada uno de los algoritmos mencionados. El segundo enfoque consistió en predecir 8 bandas de frecuencias las cuales se utilizaron para calcular finalmente el ancho de banda y la frecuencia media.

Tras entrenar los algoritmos se pudo observar que el desempeño de estos tuvo una fuerte relación en como los datos de entrenamiento se ajusta a los algoritmos, cabe destacar que algoritmo Random Forest entregó predicciones bastante certeras mediante el segundo enfoque.

Agradecimientos

Agradezco a mis padres por apoyarme en todo momento de este largo camino. Gracias infinitas a todos mis amigos del colegio, Pablo, Max, Jaime, Henry, Nico y Tata por todos los años y buenos momentos que me han regalado. Muchas gracias a mis compañeros de generación y a los cabros, Cayuya, Cristóbal, Diego Bueno, Gonzalo, Nachon, JP y Pacha, por haberme adoptado los últimos años.

Muchísimas gracias a la Pati, por todo el cariño y apañe en estos años juntos, me facilitaste la vida en la universidad con tus resúmenes y clases magistrales por eso estaré eternamente agradecido.

Quiero darle las gracias a mis profesores de mi comisión por todos sus feedbacks y me gustaría destacar la infinita buena disposición de la profesora Viviana que siempre estuvo presente para apoyarme y ayudarme, no hubiera podido realizar esto sin usted.

Finalmente me gustaría dedicar este trabajo de título a mi abuelo Sami, fuiste y serás por siempre un modelo de intelecto y paz para mi. Te recuerdo con mucho cariño y siempre te tengo en mi corazón.

Tabla de Contenido

Capítulo 1

Introducción

Los metamateriales son materiales artificiales los cuales han sido objeto de estudio debido a que presentan propiedades electromagnéticas las cuales dependen de la topología en la que se encuentran sus componentes dentro de la estructura. En otras palabras las propiedades de los metamateriales son diferentes a la de sus componentes. Debido a esto se pueden diseñar de forma estratégica los metamateriales de tal forma que se logre controlar las propiedades deseadas. De cierto modo los meta materiales tienen un parecido a los materiales compuestos comúnmente conocidos debido a que están formados por una matriz con inserciones de otro material, pero existe una diferencia en que las inserciones ocupan posiciones las cuales definen patrones que se repiten de forma periódica.

Dentro de las propiedades que pueden poseer los metamateriales se encuentran los llamados "band gaps", esta propiedad consiste en inhibir la propagación de ondas mecánicas de un cierto rango de frecuencias en el material. Esta propiedad viene siendo el principal objeto de estudio en este trabajo de título debido a que se planea lograr predecir este comportamiento en modelos de metamateriales.

Aprendizaje de máquinas es un área de la inteligencia artificial en el cual un algoritmo recibe información mediante una base de datos y mediante esta información logra identificar patrones de conducta y de esta forma predice comportamientos futuros. Dentro de lo que es aprendizaje de máquinas existen diferentes tipos de algoritmos los cuales predicen comportamientos de diferentes formas, dentro de estos se encuentran "Support Vector Machinez "Nearest Neighbors".

La construcción de diagramas de banda utilizando modelos de elementos finitos para determinar los band gaps requiere muchas evaluaciones del modelo numérico, por lo tanto es lento de implementar. Debido a eso implementar una metodología que optimice la topología para diseñar un metamaterial con un cierto band gap no es factible. Para solucionar esto se propone implementar algoritmos de aprendizaje de maquinas que permitas predecir los band gaps con precisión de forma más rápida, los que se podrían utilizar posteriormente para la optimización de la topología.

Debido a las inusuales propiedades de los metamateriales, estos han sido objeto de estudio

en el último tiempo logrando así aplicaciones en las áreas de electromagnetismo y óptica. De aquí surge la necesidad de lograr implementar un método que permita predecir el comportamiento de las ondas electromagnéticas en paneles de metamateriales. En la actualidad se estudian a partir de simulaciones numéricas para así obtener muestras de estudio, este método tiene un costo de computación bastante alto, de aquí nace la motivación de este trabajo de investigación para lograr reducir los costos de computación a partir de algoritmos de support vector machine.

Capítulo 2

Objetivos y Recursos

2.1. Objetivo General

Predecir band gaps en metamateriales a través de algoritmos de aprendizaje de máquinas.

2.2. Objetivos específicos

Los objetivos específicos son los siguientes:

- Generar bases de datos con indicadores de información sobre las bandas de frecuencia, anchos de bandas y frecuencia media a partir de los siguientes modelos: modelo tipo enrejado, modelo panel con resonadores internos y modelo enrejado con masas puntuales. resonadores
- Utilizar Kernel PCA para comprimir las bases de datos que contengan las bandas de frecuencia.
- Entrenar los siguientes algoritmos: Support Vector Machine, K-Nearest Neighbors, Random Forest, AdaBoost, GradientBoost. Para esto se utilizan las bases de datos generadas.
- Comparación y análisis de los resultados de los algoritmos utilizados.

2.3. Recursos

Para poder realizar esta investigación se cuenta con lo siguiente:

- Computador con Matlab R2018A en adelante y Python3.8(Anaconda).
- Acceso a biblioteca y bases de dato de la Universidad de Chile.

2.4. Alcances

Dentro de los alcances para llevar acabo este trabajo de título hay que considerar que existe diferentes métodos de aprendizaje de máquinas, pero en esta investigación se trabajará utilizando solamente "Support Vector Machines, Nearest Neighbors, Random Forest, AdaBoost y GradientBoost". Además, se debe mencionar que el modelo numérico del panel no será realizado como parte del trabajo por lo tanto se utilizará un modelo desarrollado en un trabajo de título previo.

Capítulo 3

Antecedentes

3.1. Metamateriales

Los metamateriales han sido objeto de estudio por diferentes disciplinas debido a que estos presentan propiedades físicas difíciles de encontrar en la naturaleza por si misma. El estudió de estos materiales tan inusuales comenzó a desarrollarse con mayor fuerza a comienzos del año 2000, en donde los científicos estudiaban las interacciones de las ondas electromagnéticas con estos tipos de materiales.

Estos materiales se generan de manera artificial mediante una arquitectura arbitraria la cual es diseñada de forma inteligente. Los metamateriales generalmente se constituyen a partir de una matriz de un cierto tipo de material con inserciones de otro material al igual que los materiales compuestos, una de las formas más comunes de desarrollarlos es mediante paneles de un material con un núcleo de otro material entremedio, a esto se le conoce como panel tipo sándwich. A diferencia de los materiales compuestos las inserciones de la segunda fase en los metamateriales ocupan posiciones específicas las cuales se repiten a los largo de la matriz, esta construcción de forma inteligente es lo que se conoce como estructuras periódicas. De esta forma los metamateriales logran obtener propiedades globales las cuales son diferentes a las propiedades de cada componente, obteniendo así propiedades inusuales.

Como se explicó anteriormente a partir de la geometría que poseen los metamateriales se pueden obtener diferentes propiedades, dentro de las propiedades que se pueden controlar están las interacciones con las ondas electromagnéticas y mecánicas. Cuando una onda electromagnética o mecánica ingresa a un medio las propiedades de esta se ven afectadas debido a las interacciones de la onda con los electrones y moléculas del medio. Debido a esto en el estudio de los metamateriales se diseñan estructuras de tal forma que se logren ciertas interacciones con ondas obteniendo metamateriales con diversas aplicaciones como por ejemplo en las áreas de óptica y comunicaciones.

Existen diferentes tipos de metamateriales los cuales se diferencian mediante las propiedades que estos presentan, dentro de los diferentes tipos de metamateriales se encuentran los llamados cristales fonónicos.

3.1.1. Paneles Sándwich

Estas estructuras están formadas por varias capaz y poseen láminas en las caras las cuales son particularmente delgadas pero con alta resistencia y rigidez. Además de esto, los paneles sándwich poseen un núcleo ancho pero de baja densidad los cuales tiene múltiples aplicaciones en la ingeniería. En la Figura 3.1.1 se observa un panel tipo sándwich y las partes que lo componen. Dentro de los parámetros que se deben considerar para diseñar estas estructuras se encuentran el material de las laminas de las caras y el núcleo, el grosor de las láminas y el núcleo, la topología del núcleo y finalmente el espesor relativo y la disponibilidad de volumen [1].

Este tipo de estructuras cuentan con una resistencia a la flexión bastante más grande que la de una placa rígida solida hecha del mismo materiales que las capaz y del mismo peso que un panel sándwich. En un panel tipo sándwich las placas se posicionan con una separación con el objetivo de incrementar el momento de inercia y de este modo la rigidez de flexión [2].

Las placas en estas estructuras son las que cumplen la función de resistir los esfuerzos de tracción y comprensión, cabe destacar que la resistencia a la flexión es bastante pequeña por lo que es despreciable. El núcleo del panel debe tener una rigidez en la normal tal que sea capaz de mantener constante la distancia entre placas y para lograr asegurar que al doblarse el panel no ocurra deslizamiento el núcleo debe tener una rigidez de corte alta [1].

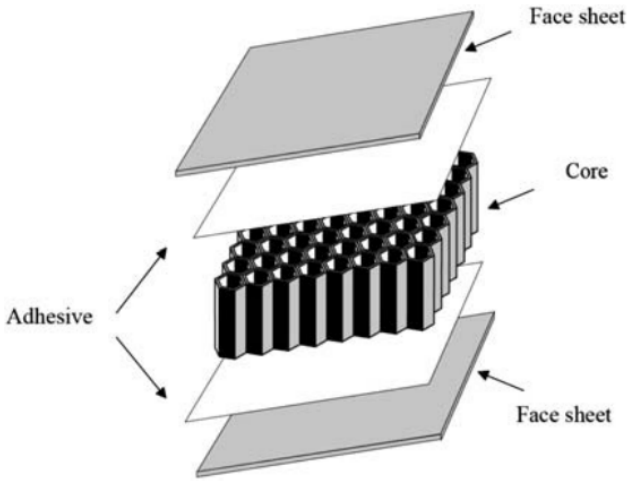

Figura 3.1.1: Panel tipo sándwich [1]

3.1.2. Estructuras Periódicas

Los materiales periódicos se definen como la repetición periódica de la porción más simple de la estructura, a esta porción se le conoce como celda. La celda puede repetirse a lo largo de la estructura en una, dos o tres dimensiones. Por lo general la celda unitaria esta formada por un arreglo de vigas o barras formando una estructura enrejada y pueden tener configuraciones de celda tanto de dos como tres dimensiones [3]. En la figura 3.1.2 se observa un material en tres dimensiones con periodicidad de celda unitaria en una, dos y tres dimensiones [4]. Estas estructuras son utilizadas en la construcción de metamateriales debido a que poseen alta resistencia específica, rigidez específica, densidad relativa baja, buen control de

la transferencia de calor y gran absorción de energía mecánica.[5]

Dentro de los factores que influyen en las propiedades de las estructuras celulares se encuentra la topología de las celdas, el número de celdas, cantidad de uniones de elementos, geometría, espesor de los elementos, tamaño de la celda y características del material [6]. Mediante la manipulación de estos factores es posible variar las propiedades características del material celular para obtener diferentes propiedades dependiendo de lo que se solicite. Cabe destacar que mediante a estudios se ha logrado determinar que ciertos tipos de celdas periódicas han mostrado poseer band gaps fónicos completos, lo que hace que estas estructuras sean bastante llamativas para ser aplicadas en aislantes de sonido y supresores de vibraciones.

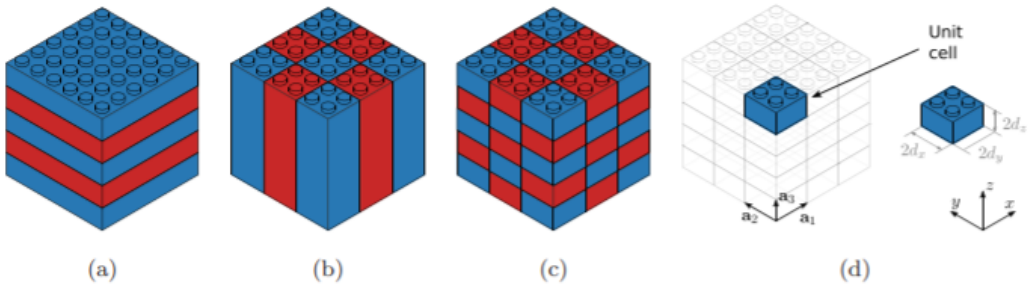

Figura 3.1.2: Material 3D con diferentes periodicidades [4]

3.1.3. Cristales Fonónicos

Los cristales fonónico también llamados metamateriales acústicos, son materiales dotados de una arquitectura periódica los cuales están diseñados de tal forma que les permite suprimir ondas acústicas o electromagnéticas en ciertos rangos de frecuencias dependiendo de la dirección de propagación. Cabe destacar que estos metamateriales también se pueden diseñar para que no solo supriman ondas si no que también las transmitan o amplifiquen. A través de estudios se logró identificar que el comportamiento de las ondas de sonido se puede controlar mediante la manipulación de algunos parámetros como por ejemplo la densidad, compresibilidad, tipo de material, el tamaño y el espaciado [7]. Los cristales fonónicos están construidos generalmente por combinaciones de fases tanto solido-solido o solido-fluido. Tomando en cuenta esto debido a que las ondas se comportan de forma diferente dependiendo el medio por la cual viaja esta dependerá de como esta constituido el material, siendo acústico un cristal fonónico con matriz fluida y elástico para uno con matriz sólida. Cabe destacar que estos cristales pueden ser estructuras periódicas de una, dos o tres dimensiones como se puede observar en la figura 3.1.3, además estás estructuras cubren ordenes de magnitud desde los metros hasta nanómetros dependiendo de las características de propagación de ondas y las aplicaciones que se requieran [8].

Las redes periódicas formadas a partir de cristales fonónicos presentan una propiedad llamada band gaps fonónicos, la cual suprime la propagación de ondas acústicas y elásticas

en un cierto rango de frecuencias. Debido a esta propiedad este tipo de metamateriales se utilizan en aislantes acústicos y supresores de vibraciones.

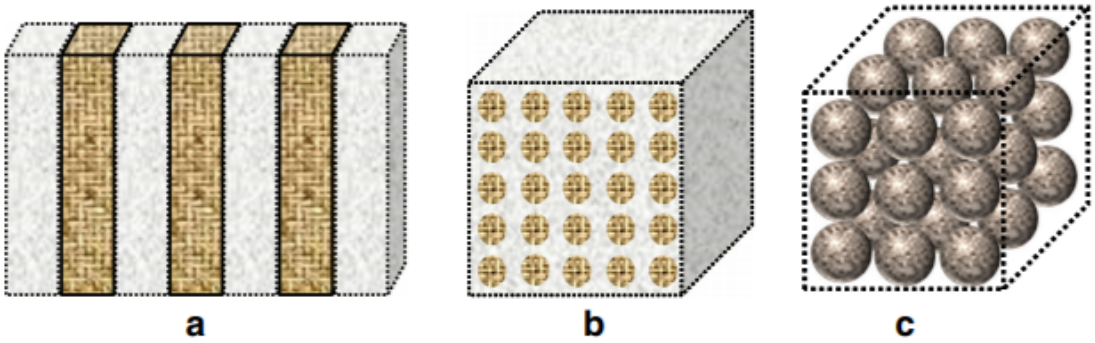

Figura 3.1.3: Cristales fonónicos con periodicidad en (a) una, (b) dos, (c) tres dimensiones. [8]

3.1.4. Band Gaps Fonónicos

Los band gap son una propiedad que poseen los materiales la cual suprime las ondas mecánicas en un cierto rango de frecuencias. En los metamateriales las inserciones periódicas generan una dispersión y reflexión de las ondas, de esta forma en la interfaz del metamaterial ocurren interferencias. La periodicidad de la red dentro de la matriz de los metamateriales permiten que en ciertas direcciones rangos de frecuencias en donde la onda se puede propagar y otros rangos en donde la onda no se puede propagar. Existen diferentes mecanismos para que ocurran band gaps fonónicos, dentro de estos mecanismos el mas recurrente es Bragg. Las bandas de Bragg causan interferencias destructivas en las ondas que se propagan en las inserciones del metamaterial de esta forma se impide la propagación de las ondas. Las bandas fonónicas en los metamateriales producto de Bragg dependen de la ubicación espacial de la periodicidad de la estructura.

Cada metamaterial va a presentar diferentes estructuras y características de band gap fonónicos. Debido a esto se intenta tener un band gap lo mas ancho posible para las aplicaciones en relación a aislantes de sonido y filtros. Se aprecia que los band gaps, tanto su tamaño como ubicación dependen de la topología de las fases que conforman al metamaterial, de las propiedades elásticas de los materiales y la simetría dentro de la estructura. En la figura 3.1.4 (imagen (f))se aprecia un diagrama de band gap para una celda de cristal fonónico con diseño cuadrado. En este gráfico se representan diferentes rangos de frecuencias normalizadas y las diferentes direcciones en las que se pueden propagar. Se observa la presencia de un band gap entre 0.6 y 0.8 (rango de frecuencias normalizadas), esto indica que las ondas no se puede propagar en ese rango de frecuencias.

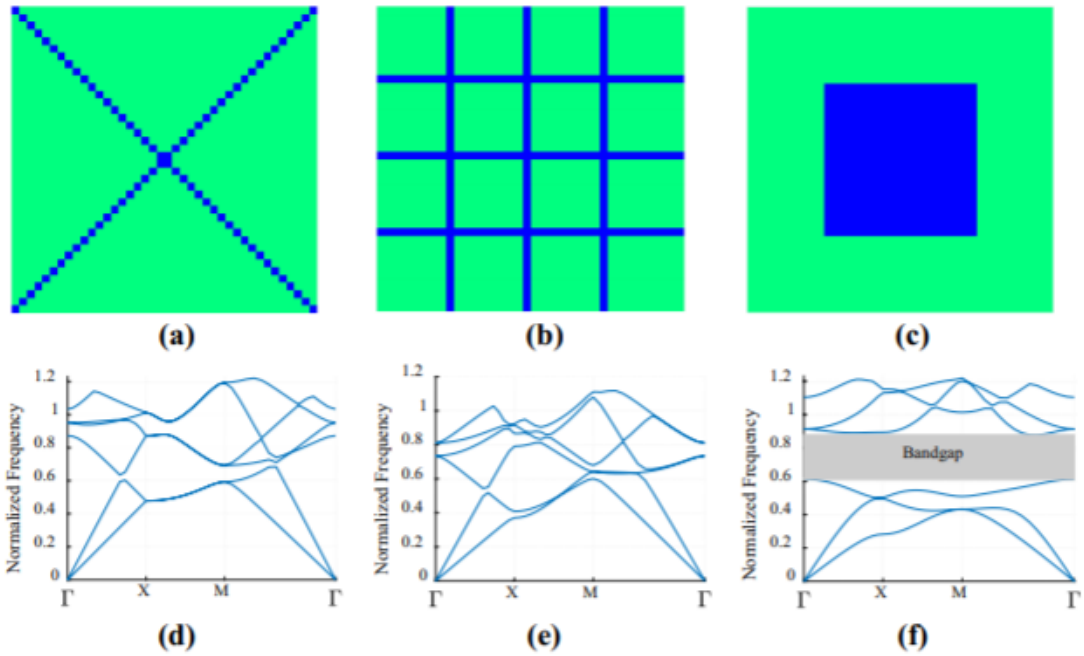

Figura 3.1.4: Tres diferentes diseños de celda unitaria y sus gráficos de bandas respectivos [9].

3.2. Machine Learning

Los algoritmos han sido muy importantes en el desarrollo tecnológico de la vida humana, estos son una serie de reglas las cuales permiten solucionar problemas. Bajo esta definición se puede decir que las actividades humanas han sido regidas por algoritmos y han impulsado su desarrollo. La computación ha sido uno de las áreas que ha desarrollado con mayor profundidad los algoritmos con el objetivo de facilitar o agilizar las tareas realizadas por el ser humano. Dentro de los algoritmos computacionales encontramos machine learning, este tipo de algoritmos es de los principales modelos de investigación utilizados en la actualidad. Esta categoría de algoritmos utiliza diferente modelos con los cuales le permite al computador mejorar sus resultados de forma progresiva.

En términos mas crudos machine learning es una disciplina en el cual se desarrollan algoritmos los cuales están representan modelos matemáticos con los que se utilizan muestras de datos para realizar predicciones o decisiones de forma independiente [10]. Estos algoritmos quedan definidos por hiper parámetros los cuales tienen la función de controlar el proceso de aprendizaje de los algoritmos, cabe destacar que existen diversos métodos que permiten optimizar los hiper parámetros para obtener mejores resultados. El aprendizaje se realiza utilizando conjuntos de datos los cuales se separan en dos categorías, los llamados datos de entrenamiento y los datos de prueba. Tal y como su nombre lo dice los datos de entrenamiento son los que se utilizan para entrenar el algoritmo con el propósito de que este logre predecir futuros datos. Por otra parte los datos de prueba son los que se utilizan para analizar y comprobar que tan bien logra predecir los datos el algoritmo implementado. Dentro de el área de machine learning existen diferentes algoritmos de aprendizaje individuales como por ejemplo k-Nearest Neighbors, Support Vector Machines (SVN) y Arboles de Decisión. Estos algorit-

mos individuales pueden ser combinados unos con otros o utilizar de forma repetida uno en específico para conseguir algoritmos más complejos que posean resultados más precisos, ha esto se le conoce como Ensamble Learning, algunos ejemplos de este tipo de algoritmos son Random Forest y Redes Neuronales.

Como se mencionó los algoritmos en machine learning se entrenan utilizando bases de datos por lo que se han desarrollado diversos métodos que permiten perfeccionar las bases de datos para luego así facilitar el proceso de entrenamiento obteniendo mejores resultados, a esto se le conoce como preprocesamiento de datos.

Uno de los procesos más importantes a la hora de trabajar utilizando algoritmos de machine learning es poder evaluar el rendimiento de estos, en otras palabras poder determinar si estos están aprendiendo de forma correcta. Para esto se han desarrollado diferentes métodos llamados métricas.

3.2.1. Algoritmos de Aprendizaje Individuales

Como se mencionó anteriormente existen diversos algoritmos que entran en la categoría de machine learning, estos algoritmos logran generar resultados a partir de bases de datos de forma individual, es decir no necesitan ser combinados unos con otros para obtener los resultados. A continuación se mostraran algunos de estos algoritmos de aprendizaje.

Nearest k Neighbors

En el área de machine learning K-nearest-neighbor(kNN) es de los algoritmos más simples y fundamentales para clasificar objetos, sobre todo para casos en los que hay poco conocimiento previo de la distribución de los datos. Como su nombre lo dice kNN clasifica objetos según las distancias entre estos, en otras palabras toma un conjunto de datos y clasifica los objetos según el objeto más cercano. Debido a que en los casos reales los límites entre datos no están definidos de forma exacta como para clasificarlos se puede utilizar kNN para predecir el valor de los datos de una regresión.[11]

En su forma de regresión este algoritmo funciona de forma similar a la clasificación pero en este caso el algoritmo predice la regresión que se ajusta a la base de datos entregada para entrenar el algoritmo, permitiendo resultados continuos. Dentro de los parámetro importantes en primer lugar se encuentra k, el cual indica la cantidad de vecinos con los que se quiere comparar el valor a predecir, cabe destacar que no existe una relación entre el tamaño de k y la exactitud de la predicción. Otro parámetro importante para este algoritmo es el peso de cada dato dentro de la regresión, en otras palabras se le puede otorgar un valor de importancia a los datos para que tengan prioridad en comparación a otros datos. En la figura 3.2.1 que se encuentra a continuación se observa una representación de este método de regresión. En este diagrama se utiliza el algoritmo kNN para predecir nuevos datos que pertenecen a la regresión representada por una linea roja a partir de una base de datos representada por los puntos azules [12]. Cabe destacar que este algoritmo tiene un costo bastante alto en relación a tiempo de computación y si las variables de entrada tienen muchas características tiende a bajar su rendimiento [13].

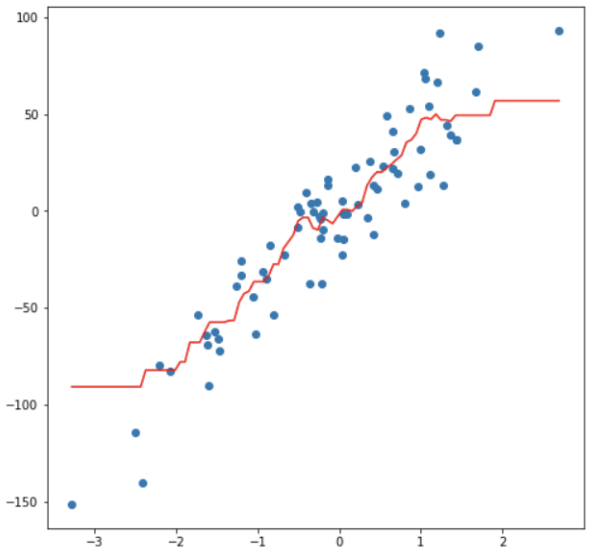

Figura 3.2.1: Representación de algoritmo kNN [12]

Este algoritmo esta basado comúnmente en la distancia Euclidiana entre los datos de muestra y los datos de entrenamiento, esto se puede ver en la siguiente ecuación:

$$\mathrm{d}(x_{\mathrm{i}}, x_l) = \sqrt{(x_{\mathrm{i}1} - x_{l1})^2 + (x_{\mathrm{i}2} - x_{l2})^2 + ... + (x_{\mathrm{i}p} - x_{lp})^2)} \tag{3.1}$$

En donde x_{i} es un objeto de que se quiere clasificar, x_l es un objeto con el cual se entreno el algoritmo y cada uno pose una cantidad p de características.

Este algoritmo cuenta con los siguientes hiper parámetros que controlan el proceso de aprendizaje:

- k: número de vecinos con los cuales se compara utilizando la norma euclidiana para determinar la predicción.

Support Vector Machine

Support vector machine es un modelo versátil de aprendizaje de maquinas, el cual es capaz de realizar clasificación, regresión y detección de valores atípicos, siendo de los modelos más populares en esta área. Este algoritmo es particularmente adecuado para clasificar datos complejos pero para conjuntos de tamaño pequeño o mediano [14].

Este algoritmo de entrenamiento se utiliza principalmente para clasificar las características de objetos en un espacio de k dimensiones mediante una separación de grupos utilizando un hyperplano de corte. Cabe destacar que el caso de 2 dimensiones es ideal y el mas sencillo para calcular el hiperplano de corte. El algoritmo de SVM puede ser adaptado para regresión, en esencia el algoritmo para ambos casos es el mismo pero la principal diferencia entre SVM como mecanismo de clasificación y regresión es que para la clasificación el resultado es categórico, en otra palabras el dato que se quiere predecir o es uno o es otro. En cambio para la regresión

existe un margen de tolerancia el cual permite mayor libertad y resultados menos categóricos. A continuación se observa gráficamente el algoritmo SVM en la figura 3.2.2 la cual posee un conjunto de datos representados por estrellas que están separados por un plano de corte en dos dimensiones, se logra apreciar que existes una tolerancia de épsilon la cual le otorga mayor holgura a los datos.

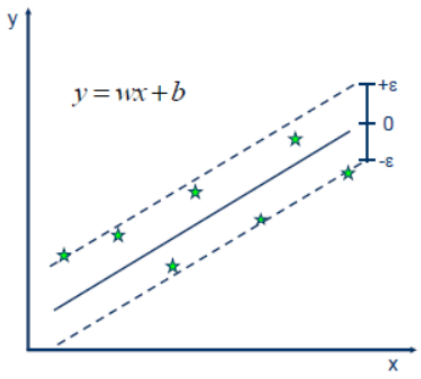

Figura 3.2.2: Representación de algoritmo SVM para regresión

Por lo general no se trabaja en casos de dos dimensiones o con hiperplanos lineales como en la figura 3.2.2, debido a esto y a otras complicaciones del aprendizaje lineal se utilizan funciones de Kernel. Estas funciones se encargan de transformar los datos añadiendo dimensiones manteniendo las características de los objetos permitiendo así encontrar un hiperplano que se adapte mejor al problema. Se puede ver un ejemplo de esta transformación en la figura 3.2.3. Cabe destacar que existen diferentes funciones de Kernel y se debe buscar la que se adapte mejor a los datos, dentro de estas se encuentran:

- Kernel Lineal: Viene dada por el producto entre x,y más una constante opcional c. Como se puede observar en la ecuación 3.2 [15].

$$k(x, y) = x^T y + c \tag{3.2}$$

- Kernel Polinomial: Este tipo de función es bastante útil cuando los set de datos han sido normalizados. Cuenta con parámetros ajustables como la pendiente α, la constante c y el grado d del polinomio. Se puede observar esta función kernel en la ecuación 3.3 [15].

$$k(x, y) = (\alpha x^T y + c)^{\mathrm{d}} \tag{3.3}$$

- Kernel Radial: Esta función utiliza el cuadrado la distancia euclidiana entre vectores y un parámetro θ que varía entre 0 y 1, el cual permite modificar la función. A continuación en la ecuación 3.4 se puede observar esta función [15].

$$k(x, y) = \exp \frac{||x - y||^2}{2\theta^2} \tag{3.4}$$

Dentro de los hiper parámetros de SVM que permiten controlar el proceso de aprendizaje se encuentran los siguientes:

- Kernel: específica las función kernel a usar.
- c: define el rango en aceptable de valores antes de ser considerado erróneos.
- ε: define un margen de tolerancia en el cual no se penalizan los errores.

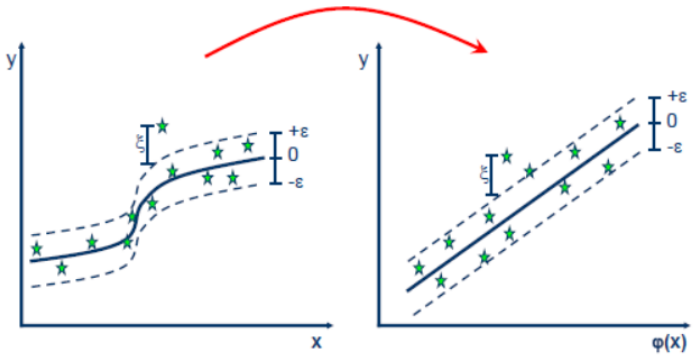

Figura 3.2.3: Transformación de hyperplano utilizando kernel

Arboles de Decisión

Los arboles de decisión son algoritmo bastante versátiles y potentes que pueden ser usados tanto para clasificación como regresión. Estos son capaces de ajustar conjuntos de datos complejos por lo que son bastante utilizados en problemas de alta dificultad [14]. Este algoritmo crea un modelo el cual logra predecir el valor de una variable objetivo, para lograr esto el algoritmo utiliza la representación del árbol para así resolver el problema. Cada hoja del árbol es un nodo y su valor dependerá de si se esta usando clasificación o regresión. Para clasificación el valor de las hojas son valores discretos entregando resultados categóricos, por otra parte para el algoritmo de regresión las hojas son valores numéricos continuos entregando así resultados numéricos.

A continuación se observan dos imágenes de arboles de decisión, una para clasificación en la figura 3.2.4 y para regresión en la figura 3.2.5. Se puede observar que en la figura 3.2.4, el algoritmo hace una predicción siguiendo una regla la cual en este caso es el tamaño del pétalo y dependiendo de esto se mueve por la raíces a los nodos hijos hasta encontrar un resultado que para este ejemplo es un tipo de flor, es decir un resultado categórico. Por otra parte la figura 3.2.5 se observa que es bastante similar pero que a diferencia de la figura 3.2.4 la predicción da como resultado un valor numérico [14].

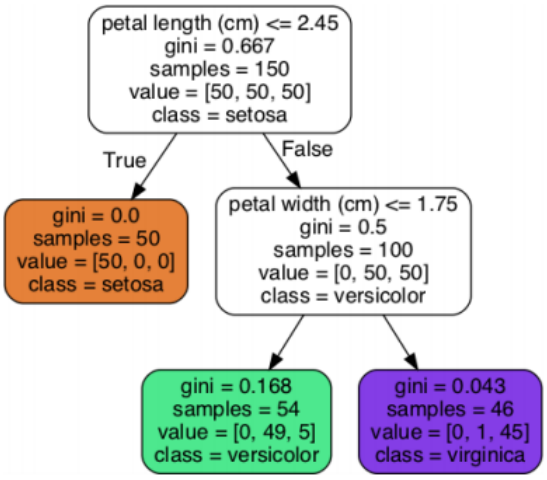

Figura 3.2.4: Árbol de decisión para clasificación [14]

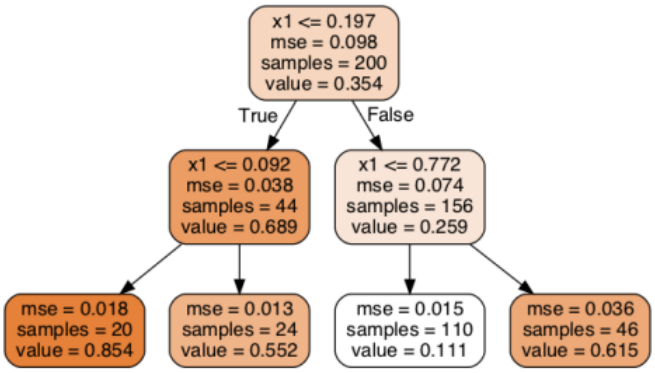

Figura 3.2.5: Árbol de decisión para regresión [14]

Dentro de los hiper parámetros que rigen a este algoritmo se encuentran:

- Profundidad: Define la profundidad del árbol.

3.2.2. Ensamble Learning

Tal y como lo dice su nombre este tipo de entrenamiento es cuando se utiliza un conjunto de algoritmos de aprendizaje para realizar las predicciones deseadas. Una de las ventajas que se obtiene de combinar modelos individuales es que el conjunto tiende a ser menos sesgado y menos sensible a los datos, es decir con menor varianza. Dentro de los métodos de ensamblaje de algoritmos los más populares son llamados Bagging y Boosting.

- Bagging: Se entrenan modelos individuales de forma paralela y cada unos de estos modelos se entrena con un subconjunto aleatorio de los datos. En la figura3.2.6 se puede observar este método de forma esquemática.
- Boosting: Este método se basa en entrenar modelos individuales de algoritmos de forma secuencial. En otras palabras los modelos se entrelazan aprendiendo de los de los anteriores de forma secuencial. Se puede observar en la figura 3.2.7 este método de forma esquemática.

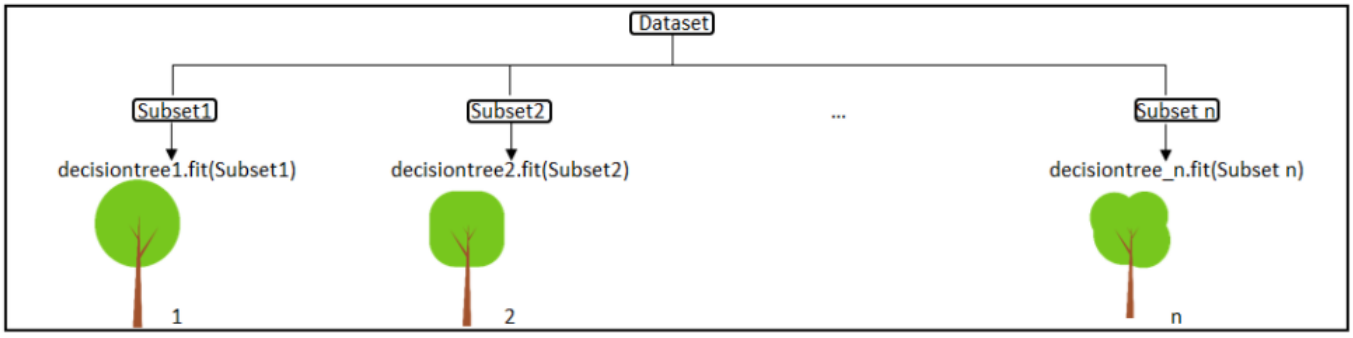

Figura 3.2.6: Ejemplo esquemático del método Bagging [16]

Existen diversos modelos que utilizan métodos de Ensamble Learning para realizar predicciones, dentro de estos se encuentran los siguientes.

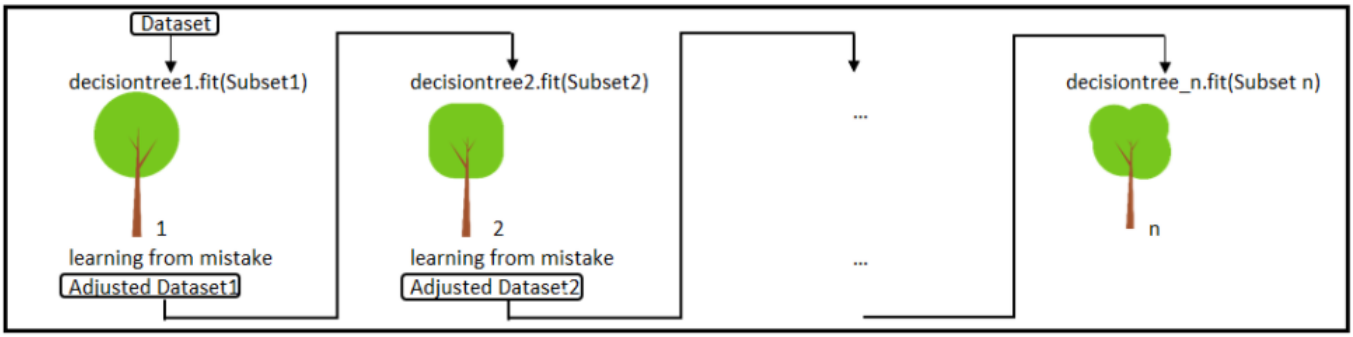

Figura 3.2.7: Ejemplo esquemático del método Boosting [16]

Random Forest

Random forest es un modelo de ensamble learning el cual es usado tanto para regresión como para clasificación. Este modelo es un ensamble de tipo bagging el cual utiliza arboles de decisión como modelo individual. Para clasificación este modelo entrega un resultado el cual es discreto (categórico) y es el que fue elegido por la mayoría de los arboles de decisión. Por otra parte en su versión de regresión este modelo entrega resultados continuos, este resultado es el promedio de las predicciones de cada árbol de decisión. En la figura 3.2.8 se puede observar este modelo de forma esquemática. Uno de los problemas que tienen estos algoritmos es que tienden a presentar overfitting.

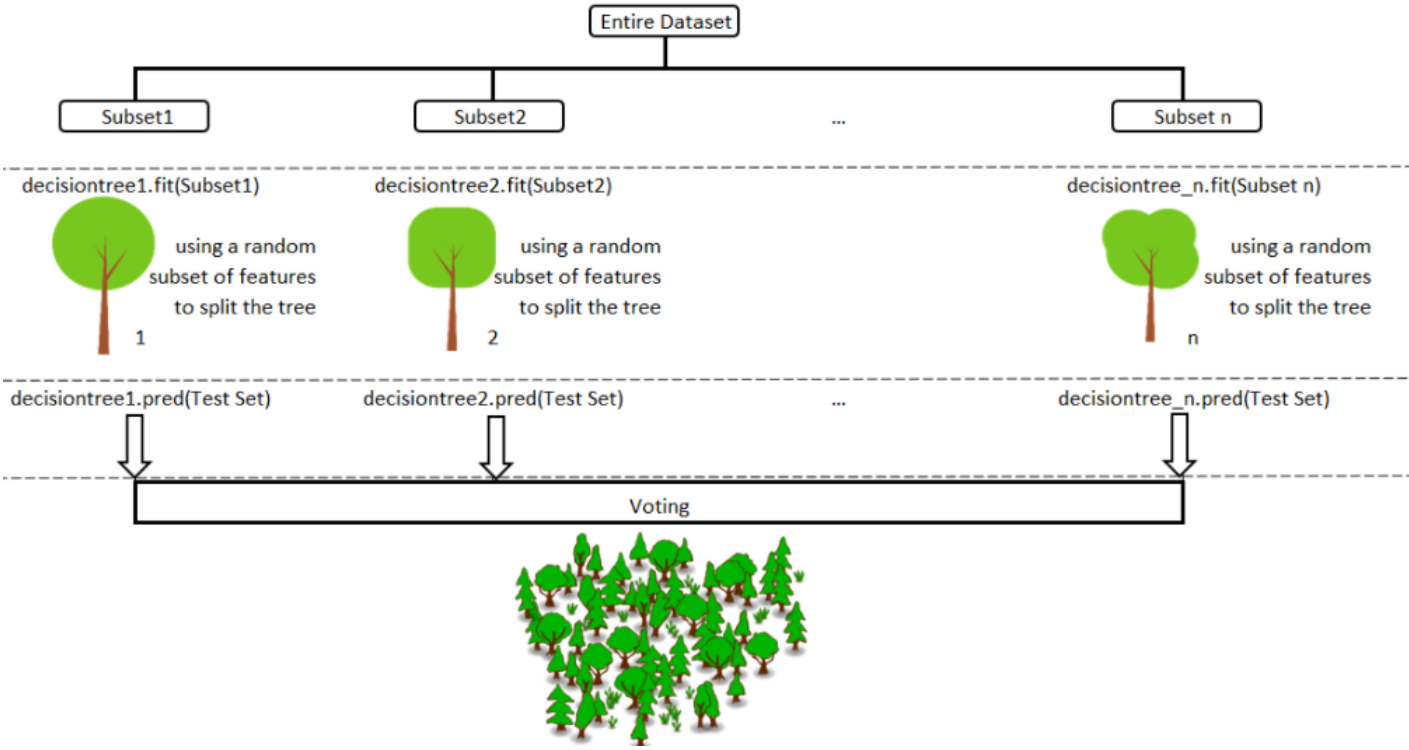

Figura 3.2.8: Ejemplo esquemático del modelo Random Forest [16]

Adaptive Boosting

Adaptative Boosting tambien llamado Adaboost es un modelo de ensamble learning de tipo boosting. Este modelo se utiliza para regresión y clasificación además al ser del tipo boosting este modelo aprende de forma secuencial de los errores cometidos por el modelo individual de entrenamiento, que por lo general son arboles de decisión. Este modelo se basa en entregar un valor de peso o importancia a los datos. A continuación se puede observar como funciona paso a paso este modelo [16]:

- Paso inicial: Se distribuye de forma uniforme el peso para cada dato, es decir para n datos el peso inicial de cada dato es $\frac{1}{n}$.
- Paso 1: Se entrena el árbol de decisión.
- Paso 2: A partir de la cantidad de predicciones erróneas del total se calcula un el peso del error e.
- Paso 3: Se calcula el peso del árbol P mediante la tasa de aprendizaje multiplicada por $log(\frac{1-\mathrm{e}}{\mathrm{e}})$.
- Paso 4: Se actualizan los pesos de los datos, si este se predijo de forma correcta mantiene el peso, en caso de haber sido erróneo su nuevo peso es igual al peso anterior multiplicado por e^P.
- Paso Final: Se repite desde el paso 1 hasta completar todos los arboles en la secuencia.

Gradient Boosting

Este es un método de aprendizaje que al igual que Adaboost es útil tanto para regresiones como para clasificación y también aprende de los errores en la secuencia de entrenamiento. A diferencia de Adaboost este método aprende del error residual de forma directa y no recopila la información en forma de pesos. En otras palabras este algoritmo calcula los errores residuales como la resta entre el valor Y real y el valor Y predicho, guarda esta información como el nuevo set de datos(Y real) y los utiliza para predecir en el siguiente árbol de la secuencia. Se puede ver este algoritmo de forma esquemática en la figura 3.2.9 [16].

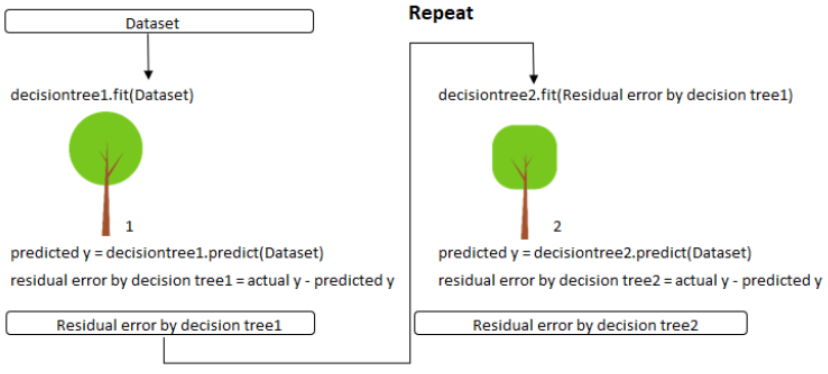

Figura 3.2.9: Ejemplo esquemático de Gradient Boosting [16]

Estos tres modelos de aprendizaje al utilizar como algoritmo individual arboles de decisión poseen hiperparametros similares. Para efectos de esta investigación se trabajo optimizando

el parámetro de número de estimadores el cual indica la cantidad de arboles de decisión que se usan en los ensambles para estos tres modelos.

3.2.3. Optimización de Hiper Parámetros

Como se explicó anteriormente todos los sistemas de aprendizaje de maquinas tienen hiper parámetros, los cuales se encargan de modelar el proceso de aprendizaje. Debido a que estos parámetros juegan un rol fundamental en el rendimiento, al variarlos dentro de un mismo algoritmo se obtendrán diferentes resultados en la predicciones. Con el fin de encontrar los mejores hiper parámetros para los algoritmos de aprendizaje se desarrollan métodos automáticos para optimizarlos. Estos métodos permiten disminuir el esfuerzo y error humano que viene dado por probar diferentes hiper parámetros de forma manual, mejorar el rendimiento de los algoritmos de aprendizaje de máquinas y finalmente permite facilitar la reproducción de investigaciones científicas y la validación de estas mismas [17]. Existen diversos métodos de optimización de hiper parámetros, pero para efectos de esta investigación se trabajó con Grid Search y Random Search.

Grid Search

Grid search es el algoritmo más básico y más utilizado en la optimización de hiperparametros, el cual se basa en una búsqueda exhaustiva en donde el usuario especifica un set finito de valores para cada hiper parámetro y luego el algoritmo de grid search evalúa el producto cartesiano de cada uno de estos parámetros. El rendimiento de este parámetro es usualmente evaluado utilizando validación cruzada la cual consiste en calcular de forma reiterada la media aritmética obtenida en las predicciones del algoritmo [17].

Random Search

Random search es una alternativa a grid search, la cual remplaza la búsqueda exhaustiva de los hiper parámetros escogidos por el usuario seleccionándolos de forma aleatoria. Este método presenta mejores resultado que grid search cuando algunos de los hiper parámetros afectas más en los resultados y el rendimiento de las predicciones [17].

En la figura 3.2.10 se puede ver una comparación esquemática entre random y grid search para una función la cual se quiere minimizar. Estas figuras representan gráficos (X,Y) en donde la variable X son hiper parámetros importantes y la variable Y representa hiper parámetros de poca importancia. Arriba de cada figura se describe la curva objetivo que estos gráficos determinan y se puede observar que para las 9 combinaciones o puntos en grid search solo 3 son importante en cambio para random search las 9 son importantes.

3.2.4. Preprocesamiento de datos

Tal y como su nombre lo indica estos son métodos que se utilizan para trabajar con los datos antes de ser utilizados en los procesos de aprendizajes con el fin de obtener mejores resultados. Existen varios métodos que se utilizan para trabajar los datos antes de ser procesados por los algoritmos de aprendizaje de máquinas, para efectos de esta investigación se trabajó utilizando análisis de componentes principales y normalización de datos.

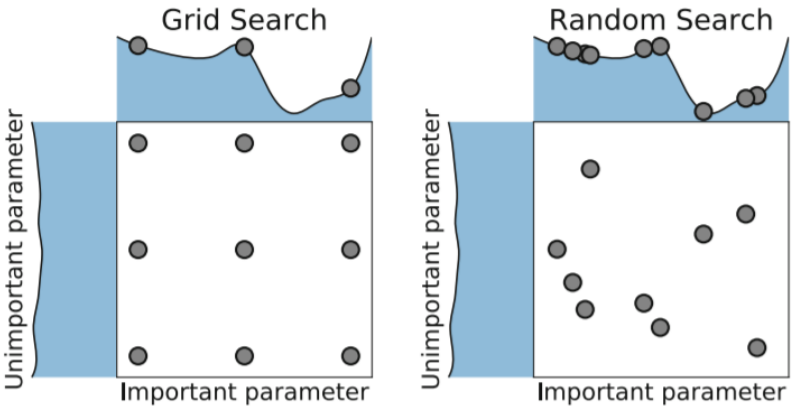

Figura 3.2.10: Comparación de grid search y random search para minimizar una curva.[17]

Análisis de Componentes Principales

El análisis de componentes principales también llamado PCA es de los métodos de preprocesamiento de datos más utilizados en estadística y aprendizaje supervisado. Este método analiza los conjuntos de datos los cuales están descritos por variables independientes interrelacionadas. El objetivo principal del método PCA es lograr extraer la información más importante de un conjunto de datos y representar estos datos como un nuevo conjunto de variables las cuales son ortogonales y reciben el nombre de componentes principales. Este método es comúnmente utilizado en casos que la dimensionalidad de los datos a trabajar es grande y se necesita reducirla [18].

Los objetivos de este método se pueden describir de forma simplificada de la siguiente forma [18]:

- Extraer la información más importante.
- Comprimir el tamaño del conjunto de datos.
- Simplificar el conjunto de datos que describen el problema.
- Analizar la estructura de los datos y las variables.

Normalización de Datos

La normalización de datos es uno de los métodos para preprocesar información más antiguos utilizados en estadística. Este método permite crear versiones escaladas y desplazada de los conjuntos de datos, con el objetivo de comparar de forma más prolija los nuevos valores normalizados del conjunto de datos. Dentro del área de inteligencia artificial se ha demostrado que al utilizar el recurso de normalización de datos conlleva a obtener un mejor rendimiento en las predicciones realizadas por los algoritmos [19].

Existen diversas formas para normalizar los datos, una de las formas más comunes de estandarizar muestras de datos es restándole a estos la media y dividiéndolos por la desviación standart, esto se puede observar en la ecuación 3.5. Dentro de los métodos de normalización en aprendizaje de máquinas se encuentra el cambio de escala (normalización mínima-máxima), el cual es de los más utilizados. Este método normaliza los datos utilizando la ecuación 3.6

que se puede ver a continuación en donde x es valor original y x' es el valor normalizado [20].

$$x' = \frac{x - \bar{x}}{\sigma} \tag{3.5}$$

$$x' = \frac{x - min(x)}{max(x) - min(x)} \tag{3.6}$$

3.2.5. Métrica

Muchos tipos de métricas son utilizadas en machine learning debido a que son los métodos que se utilizan para evaluar la precisión o rendimiento de los algoritmos [21]. Para regresión se implementan funciones de perdida, puntuación y utilidad las cuales permiten medir el rendimiento de esta. Estas funciones utilizan un método que especifica la forma en la cual se deben promediar las puntuaciones o perdidas de cada objetivo individual [22].

Como se dijo anteriormente existen varias métricas que permiten evaluar los rendimientos de los algoritmos, pero dentro de los algoritmos para regresiones la más utilizada es el coeficiente de determinación R^2.

Coeficiente de Determinación

Esta es de las métricas más utilizadas en estadística, esta representa una proporción de varianza de la variable Y la cual ha sido explicada por las variables independientes del modelo. En palabras mas simples describe que tan bien se ajusta el modelo a los datos [23]. La mejor puntuación que se puede obtener con este método es 1 y la peor es -1, puede ser negativa debido a que el modelo puede ser arbitrariamente peor. El coeficiente de determinación R^2 queda definido por la ecuación 3.7 en donde y son los valores reales, $\hat{y}$ son los valores predichos e $\bar{y}$ es la media de los valores reales [22].

$$R^2(y, \hat{y}) = \frac{\sum_{\mathrm{i}=1}^{n}(y_\mathrm{i} - \hat{y_\mathrm{i}})^2}{\sum_{\mathrm{i}=1}^{n}(y_\mathrm{i} - \bar{y})^2} \tag{3.7}$$

3.3. Latin Hipercube Sampling

A medida que se ha desarrollado la tecnología los experimentos computacionales se han hecho cada vez más populares, debido a esto se han desarrollado diferentes metodologías y recursos para realizar simulaciones computacionales. Uno de los métodos de muestreo más utilizados en simulaciones computaciones es latin hipercube sampling.

Latin hipercube sampling es un método estadístico de muestreo casi aleatorio que se utiliza para generar valores de entrada para la estimación de variables de salida para ciertas funciones. Este método tiene una menor varianza relativa en comparación a otros métodos comúnmente utilizados como el muestreo aleatorio, debido a estas razones este método se utiliza bastante en áreas de computación como machine learning. [24]

Capítulo 4

Formulación

En esta sección se detalla la formulación de las bases de datos para luego ser utilizadas en los procesos de entrenamiento mediante algoritmos de aprendizaje de máquinas. Las bases de datos se generan a partir de tres modelos numéricos realizados en otros trabajo de títulos. Estos modelos son representaciones de placas de metamateriales con las siguientes estructuras: enrejado, panel con resonadores internos y un enrejado con masas puntuales. A partir de estos modelos se obtienen los datos necesarios, los cuales son frecuencia media, ancho de banda y bandas de frecuencias. Cabe destacar que existen dos enfoques diferentes en cuanto a la generación de bases de datos los cuales se explicarán a continuación.

4.1. Primer Enfoque

El propósito de este enfoque es realizar una base de datos a partir de entregar a diferentes modelos numéricos variables de entrada los cuales son características del material y área, estos parámetros se generan de forma casi aleatoria utilizando el método latin hipercube sample. A partir de estos datos de entrada los modelos entregan como variables de salida la frecuencia media y el ancho del band gap. para luego entrenar los algoritmos de aprendizaje. Dentro de este enfoque se realizaron tres bases de datos a partir de tres modelos numéricos diferentes los cuales se detallarán a continuación.

4.1.1. Base de Datos I

Este conjunto de datos es generado a partir de un modelo tipo sándwich en el cual el núcleo del panel esta formado por un enrejado de elementos interconectados de tal forma que generan un cuadrado en el plano x-y, se puede observar el plano x-y de este modelo en la figura 4.2.1.

A continuación se presenta la dimensionalidad de esta base de datos generada a partir del primer enfoque.

- Variables de entrada: 24 (Parámetros del material y área).
- Variables de Salida : 2 (Frecuencia media y ancho del band gap).

- Tamaño de Muestra: 21445.

4.1.2. Base de Datos II

Esta base de datos se genera a partir de un cristal fonónico que se modela como un panel bidimensional el cual esta conformado por una matriz con resonadores internos. Se puede observa la celda unitaria de este modelo en la figura 4.2.2, esta celda unitaria tiene una geometría cuadrada y en su interior cuenta con vigas en voladizo, de esta forma se obtiene un sistema con masas en voladizo.

A continuación se observa la dimensionalidad de esta base de datos.

- Variables de entrada: 15 (Parámetros del material y área).
- Variables de Salida : 2 (Frecuencia media y ancho del band gap).
- Tamaño de Muestra: 11242.

4.1.3. Base de Datos III

Esta base de datos es realizada a partir de un modelo el cual nace de la combinación de los dos modelos anteriormente mencionados. en otras palabras este modelo es un panel bidimensional con una matriz enrejada que presenta masas puntuales en las vigas las cuales están interconectadas. Se puede observar la celda unitaria de este modelo en la figura 4.2.3.

A continuación se presenta la dimensionalidad de esta base de datos generada a partir de este modelo.

- Variables de entrada: 15 (Parámetros del material y área).
- Variables de Salida : 2 (Frecuencia media y ancho del band gap).
- Tamaño de Muestra: 11161.

4.2. Segundo Enfoque

Para poder calcular el ancho de banda y la frecuencia media se necesita primero obtener la diferentes bandas presentes en los modelos, las cuales son generadas a partir de las variables de entrada. En este enfoque a diferencia del anterior se pretende generar una base de datos que contenga como variables de salida las bandas las cuales se representaran de forma compacta mediante componentes principales obtenidos por medio de Kernel PCA. Al igual que el primer enfoque se utiliza hiper cube sample para generar casi aleatoriamente las variables de entrada las cuales son área y características del material. Se generaron dos bases de datos utilizando este enfoque las cuales se detallan a continuación.

4.2.1. Base de Datos II

Esta base de datos se genera utilizando el modelo de cristal fonónico con resonadores internos representado por la figura 4.2.2. Utilizando el segundo enfoque se logra generar una base de datos la cual cuenta con la siguiente dimensionalidad.

- Variables de entrada: 15 (Parámetros del material y área).
- Variables de Salida : 8 (Componentes Principales).
- Tamaño de Muestra: 11242.

4.2.2. Base de Datos III

Esta base de datos se genera utilizando el modelo enrejado con masas puntuales el cual se puede observar en la figura 4.2.3. Utilizando el segundo enfoque se logra generar una base de datos la cual cuenta con la siguiente dimensionalidad.

- Variables de entrada: 15 (Parámetros del material y área).
- Variables de Salida : 8 (Componentes Principales).
- Tamaño de Muestra: 11161.

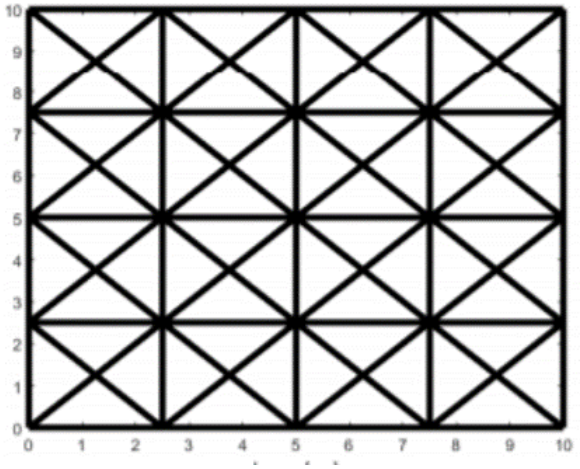

Figura 4.2.1: Plano x-y modelo enrejado.

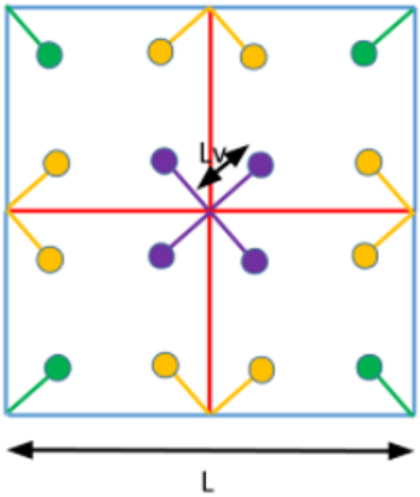

Figura 4.2.2: Celda unitaria de cristal fonónico con resonadores internos.

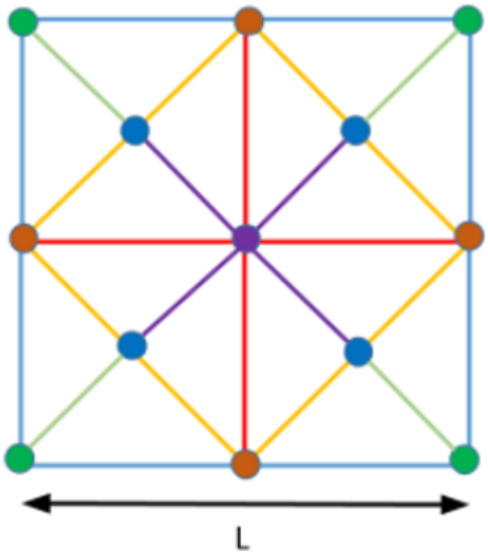

Figura 4.2.3: Celda unitaria de sistema enrejado con masas puntuales.

Capítulo 5

Metodología

El desarrollo de este trabajo de investigación se puede desglosar en dos etapas para luego analizar los resultados que arrojen. La primera etapa consiste en la generación de las bases de datos utilizando los enfoques descritos en el Capitulo 4. La segunda etapa corresponde al entrenamiento de diferentes algoritmos de aprendizaje de máquinas utilizando las bases de datos generadas en la etapa anterior. Una vez realizadas estas etapas se procede a analizar los resultados obtenidos y a concluir el trabajo de investigación. A continuación se explican de forma más detallada estas etapas mediante descripciones y diagramas de flujo.

5.1. Primera Etapa

En esta etapa se crean las bases de datos que se van a utilizar para luego entrenar los algoritmos en la etapa siguiente, cabe destacar que esta etapa depende del enfoque en el cual se esté trabajando. La generación de datos se realiza utilizando modelos numéricos de paneles tipo sándwich de materiales los cuales fueron desarrollados en MATLAB. Para realizar esta etapa se deben seguir los pasos que se nombran a continuación.

1. Se utiliza el método latin hypercube sampling para generar de forma casi aleatoria diferentes valores de características del material y área, los cuales serán las variables de entrada que determinaran las variables de salida. Los cuales se guardan en un documento compatible con Python.
2. Utilizando las variables obtenidas en el paso anterior se comienza a iterar el modelo numérico en matlab para que este calcule los diagramas de banda para cada uno de los casos representados por las variables de entrada. Cabe destacar que se dentro de los parámetros que se seleccionan para generar los diagramas, se impone que este genere 8 bandas de frecuencias.
3. Este paso dependerá del enfoque en el cual se esta trabajando, para el enfoque número uno a medida que el código va construyendo las bandas también calcula los anchos de banda y frecuencia media para cada uno de los casos. Para el enfoque número dos solo basta con la construcción de bandas y extracción de componentes principales a partir de estas.
4. Finalmente se guardan las variables de salida dependiendo del enfoque en el cual se

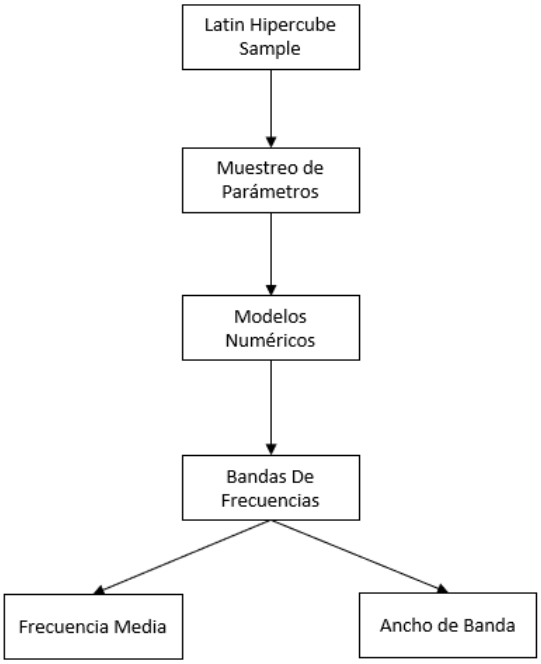

Figura 5.1.1: Diagrama de flujo para la primera etapa del primer enfoque.

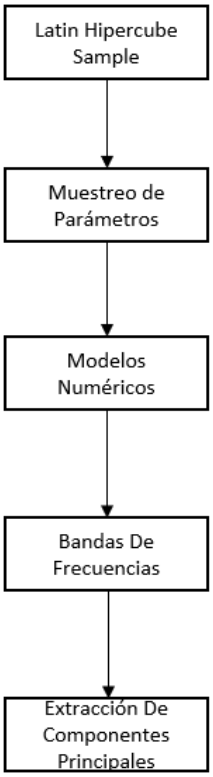

Figura 5.1.2: Diagrama de flujo para la primera etapa del segundo enfoque.

esta trabajando en un documento compatible con Python. Cabe destacar que se filtran algunos resultados de anchos de banda, sobre todo los casos iguales a 0.

Se puede observar la etapa anterior paso a paso de forma más acotada como diagramas de flujo en la figura 5.1.1 la cual representa el primer enfoque y la figura 5.1.2 para el segundo enfoque.

5.2. Segunda Etapa

En esta etapa se utilizan las bases de datos generadas en la etapa anterior para entrenar diferentes algoritmos de machine learning utilizando librerías especializadas de Python. Nuevamente el tipo de enfoque va a afectar en en la etapa debido a que el proceso de aprendizaje será diferente para cada uno de los enfoques. Para realizar esta etapa se deben seguir ciertos pasos, pero a diferencia de la etapa anterior los cambios causados por el tipo de enfoque son mas notorios por lo que se describirán de forma separada.

5.2.1. Pasos Para El Primer Enfoque

1. Se importan las librerías de algoritmos y recursos necesarios para poder trabajar en Python.
2. Se procede a seleccionar y dividir la base de datos que se utilizará para entrenar el

algoritmo seleccionado. Los datos se dividen en el conjunto de entrenamiento que corresponde al 80 por ciento de los datos y el conjunto de prueba que corresponde al 20 por ciento de los datos.

3. Se entrena el algoritmo seleccionado utilizando los recursos correspondientes (pre procesamiento de dato y/o optimización de hiper parámetros).
4. Una vez entrenado el algoritmo se predice los valores del ancho de banda y frecuencia media a partir de las variables de entrada del conjunto de prueba.
5. Se evalúa el rendimiento del algoritmo utilizando el coeficiente de determinación R^2 mediante las variables de salida predichas y las variables de salida del conjunto de prueba.
6. Se repite el proceso utilizando un nuevo algoritmo de aprendizaje de máquinas

5.2.2. Pasos Para El Segundo Enfoque

1. Se importan las librerías de algoritmos y recursos necesarios para poder trabajar en Python.
2. Se procede a seleccionar y dividir la base de datos que se utilizará para entrenar el algoritmo seleccionado. Los datos se dividen en el conjunto de entrenamiento que corresponde al 80 por ciento de los datos y el conjunto de prueba que corresponde al 20 por ciento de los datos.
3. Se entrena el algoritmo seleccionado utilizando los recursos correspondientes (pre procesamiento de dato y/o optimización de hiper parámetros).
4. Una vez entrenado el algoritmo se predice los valores de los componentes principales a partir de las variables de entrada del conjunto de prueba.
5. Se evalúa el rendimiento del algoritmo utilizando el coeficiente de determinación R^2 mediante las variables de salida predichas y las variables de salida del conjunto de prueba.
6. Se reconstruyen las bandas de frecuencias a partir de los componentes principales.
7. Se guardan las bandas reconstruidas predichas y las bandas de prueba como archivos compatibles con MATLAB.
8. A partir de las bandas predichas y de prueba se calculan los anchos de banda y frecuencia media para cada variable de entrada utilizando MATLAB.
9. Finalmente se calcula el coeficiente de determinación R^2 para el ancho de de banda y frecuencia media calculados en MATLAB.

En las figuras 5.2.1 y 5.2.2 se pueden observar los diagramas de flujo que representan de forma sintetizada esta etapa para cada uno de los dos enfoques.

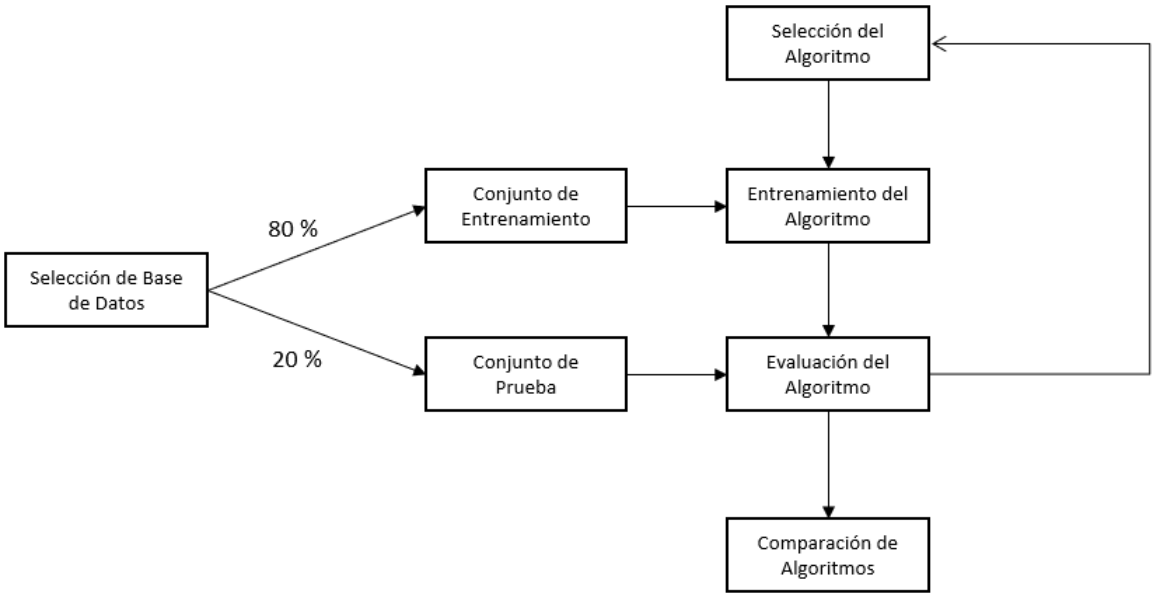

Figura 5.2.1: Diagrama de flujo para la segunda etapa del primer enfoque.

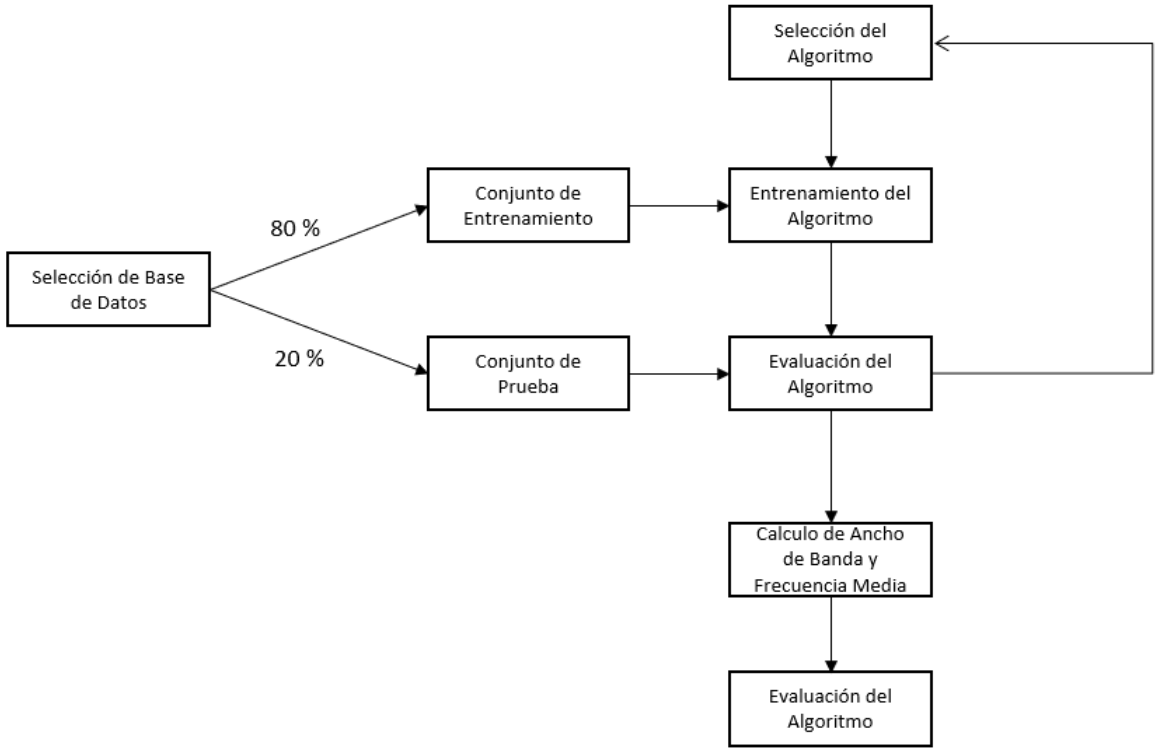

Figura 5.2.2: Diagrama de flujo para la segunda etapa del segundo enfoque.

Capítulo 6

Resultados

En esta sección se muestran los resultados obtenidos tras entrenar diferentes algoritmos mediante los dos enfoques diferentes. Se utilizaron las bases de datos generadas a partir de los modelos numéricos de placas de metamaterial para cumplir con el objetivo de este trabajo de investigación. Para obtener estos resultados se utilizaron 5 tipos de algoritmos de entrenamiento los cuales son support vector machine, k nearest neighbors, random forest, adaptative boosting y gradient boosting. También se utilizan los diversos recursos como gradient y random search para optimizar hiper parámetros y métodos de preprocesamiento de datos con el fin de mejorar los resultados para luego compararlos. Cabe destacar que se mostrarán los gráficos considerando los mejores resultados para cada algoritmo, de esta forma se observará que tanto se ajustan a la linea de tendencia deseada cuando se gráfica los valores predichos con respecto a los valores de prueba del conjunto de datos.

6.1. Primer Enfoque

6.1.1. Support Vector Machine

A continuación se pueden observar diferentes figuras en las cuales se puede observar los resultados obtenidos utilizando support vector machine para predecir tanto la frecuencia media como el ancho de banda utilizando los conjuntos de datos generados.

En las figuras 6.1.1 y 6.1.2 se observan gráficos de barra comparando los R2 para cada data set. De aquí se logra observar que los mejores resultados se obtuvieron utilizando la base de datos número dos para predecir el Band Gap y la base de datos número 3 para predecir la Frecuencia Media.

Las figuras 6.1.3 y 6.1.4 son los resultados que se obtuvieron al graficar las variables de salida predichas y del conjunto de prueba, seleccionando las bases de datos que obtuvieron un R2 más grande.

El gráfico 6.1.3 se obtuvo utilizando los parámetros por defecto de la librería de SVN, los cuales son kernel radial, C=1.0, ε=0.1. Mientras que el gráfico 6.1.4, se realizó utilizando un grid search para optimizar los hiper parámetros lo que dio como resultado un kernel lineal,

C=10, ε=0.5. Además se utilizó PCA para trabajar previamente los datos.

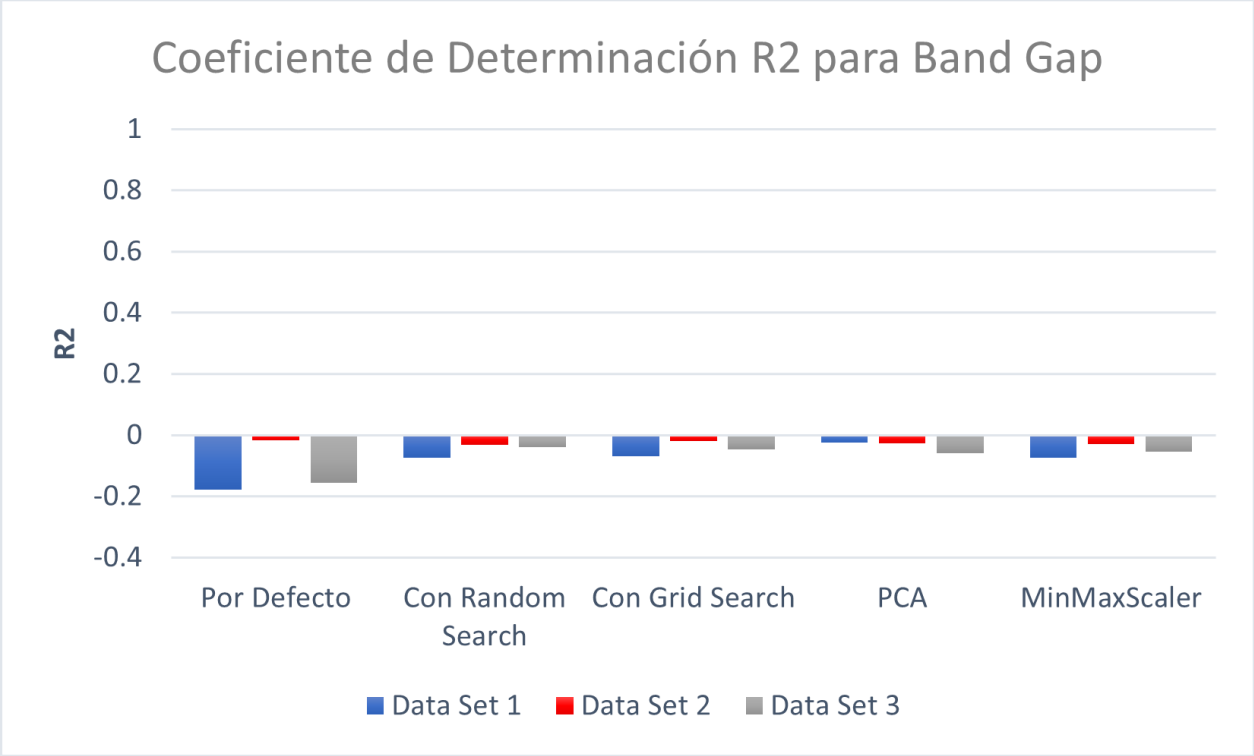

Figura 6.1.1: Coeficiente de Determinación obtenido para cada base de datos utilizando SVN para predecir el ancho de banda.

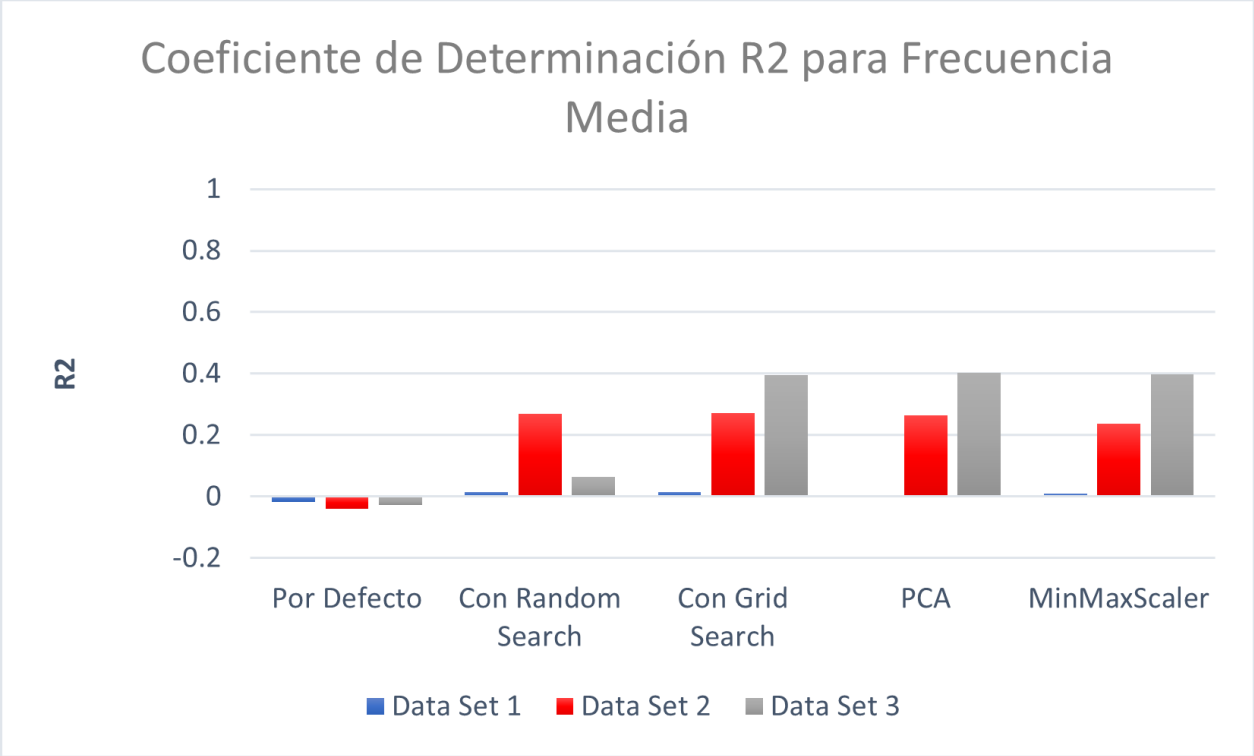

Figura 6.1.2: Coeficiente de Determinación obtenido para cada base de datos utilizando SVN para predecir la frecuencia media.

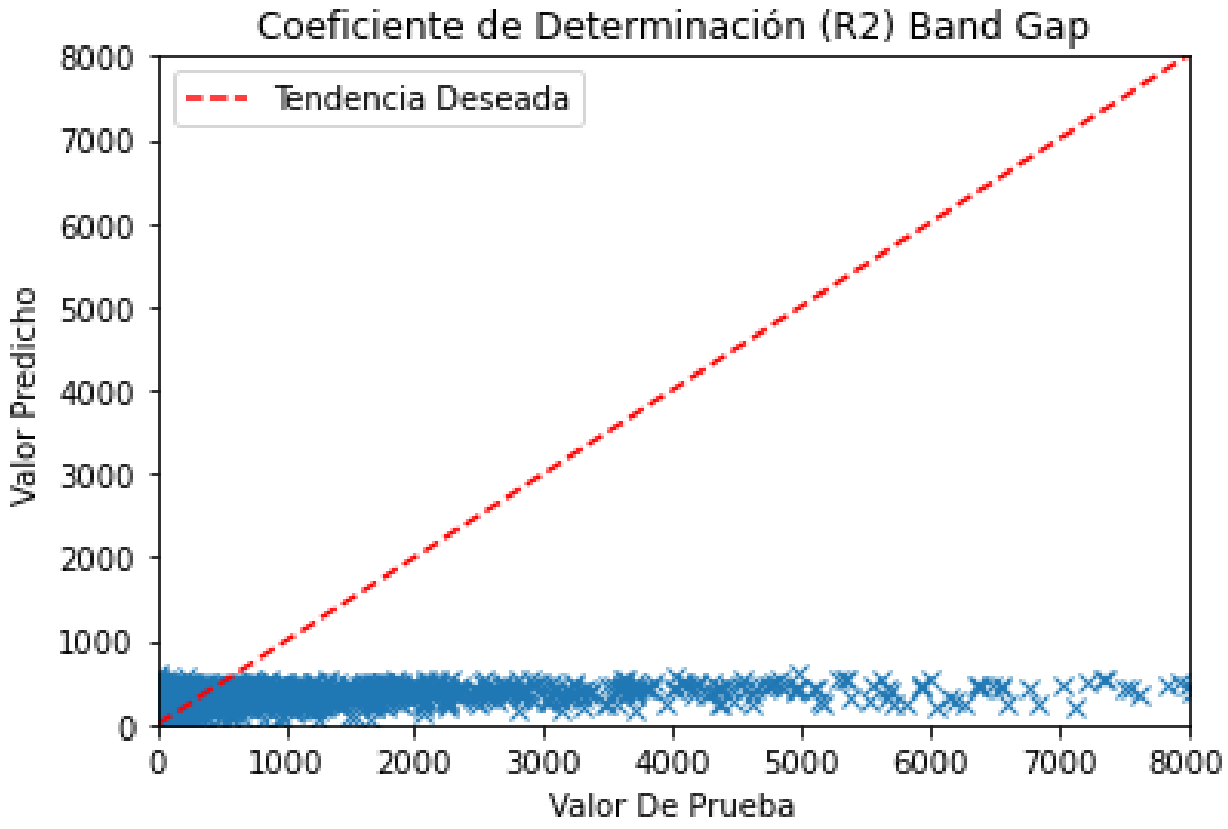

Figura 6.1.3: Rendimiento del algoritmo SVN para predecir el ancho de banda mediante el data set 2.

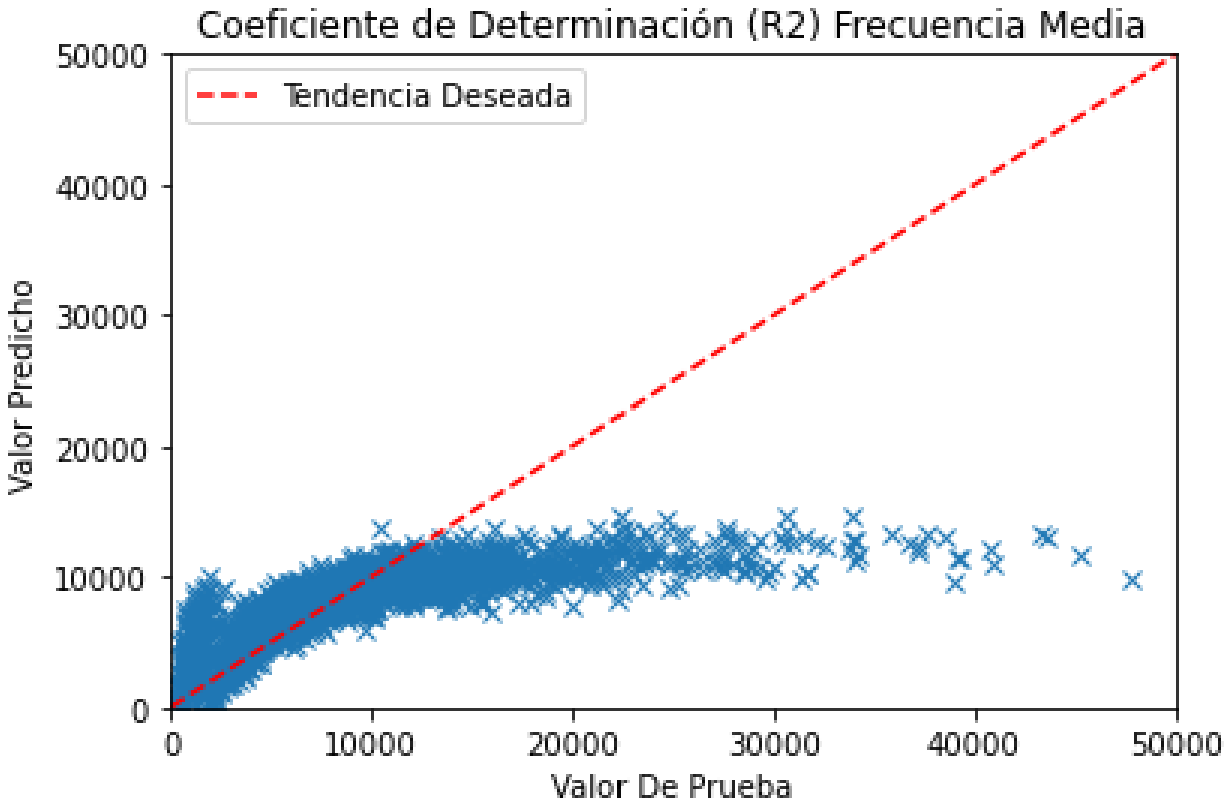

Figura 6.1.4: Rendimiento del algoritmo SVN para predecir la frecuencia media mediante el data set 3.

6.1.2. K Nearest Neighbors

Se puede observar que a diferencia de SVN, para ningún set de datos se obtienen valores de R2 negativos. Se observa de las figuras 6.1.6 y 6.1.5 que las bases de datos que obtuvieron un R2 más alto fueron el data set 1 para predecir el band gap y el data set 2 para predecir la frecuencia media, se puede ver de forma más detallada el rendimiento para estas bases de datos en las figuras 6.1.7 y 6.1.8 . En ambos casos se utilizo grid search para optimizar la cantidad de vecinos cercanos lo cual indicó que 6 vecinos era el hiper parámetro óptimo para ambos set de datos.

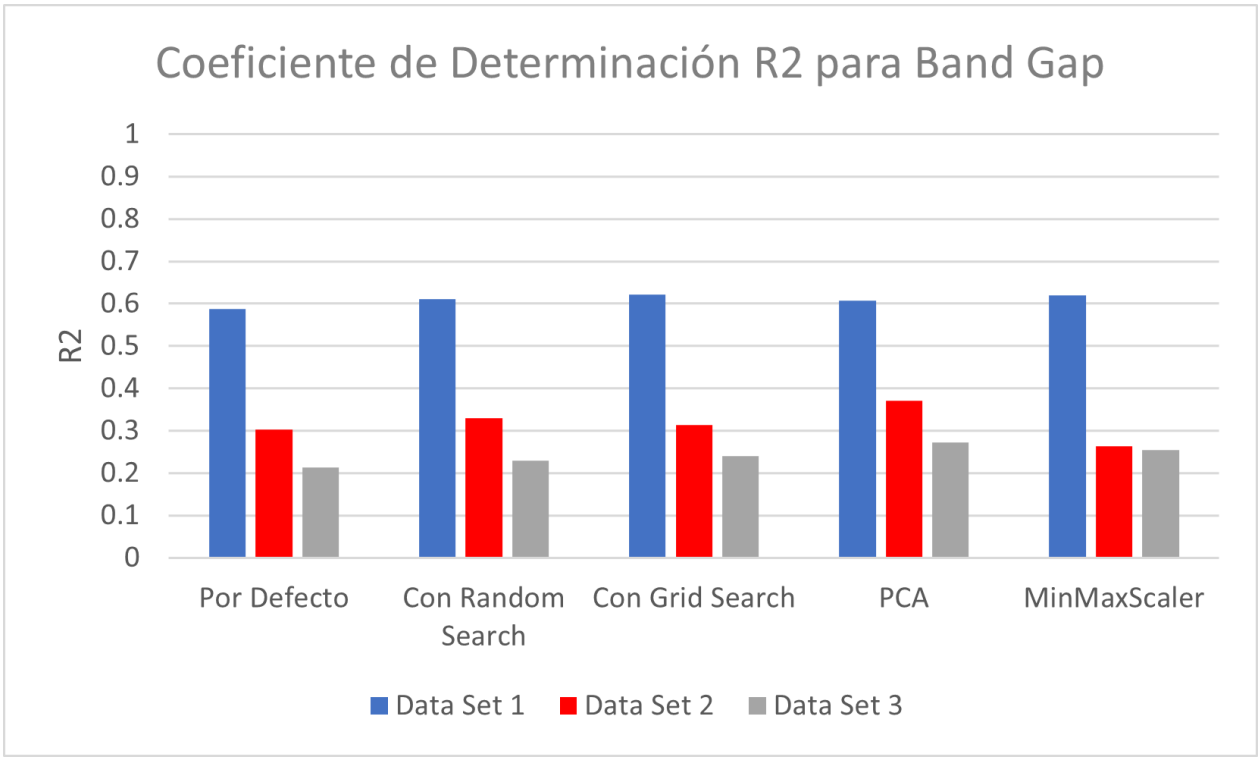

Figura 6.1.5: Coeficiente de Determinación obtenido para cada base de datos utilizando KNN para predecir el ancho de banda.

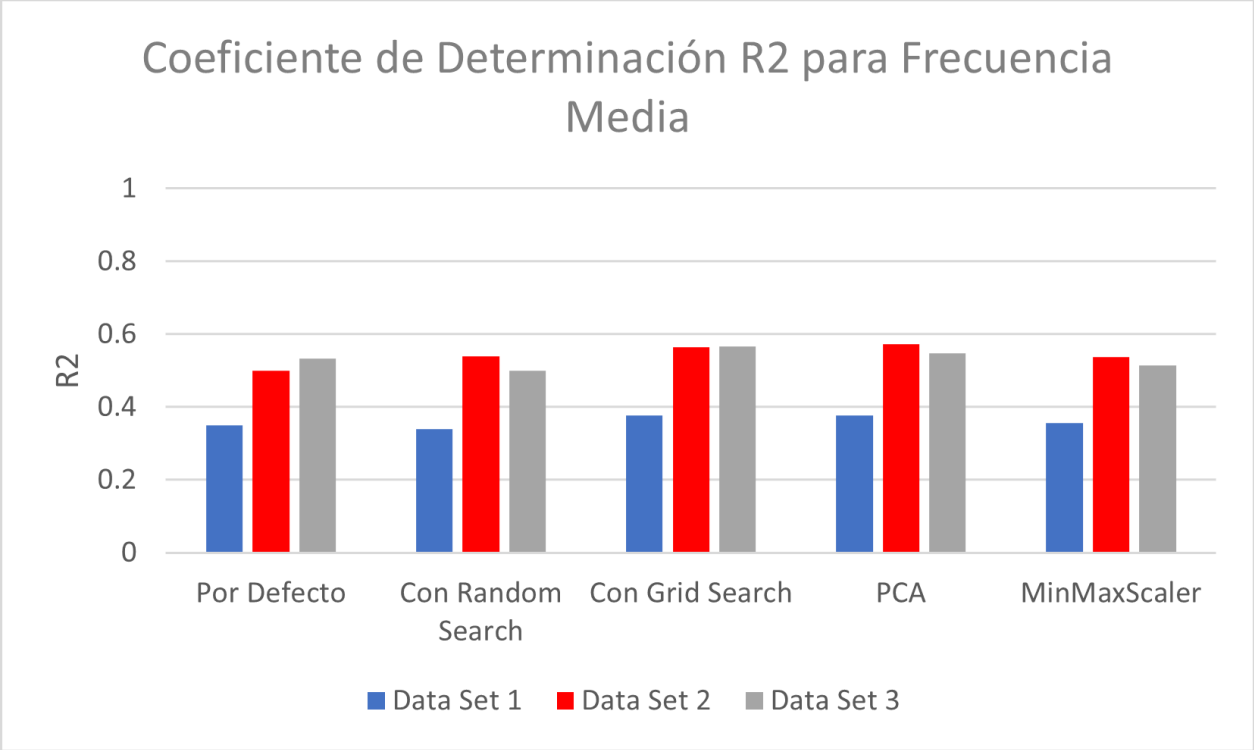

Figura 6.1.6: Coeficiente de Determinación obtenido para cada base de datos utilizando KNN para predecir la frecuencia media.

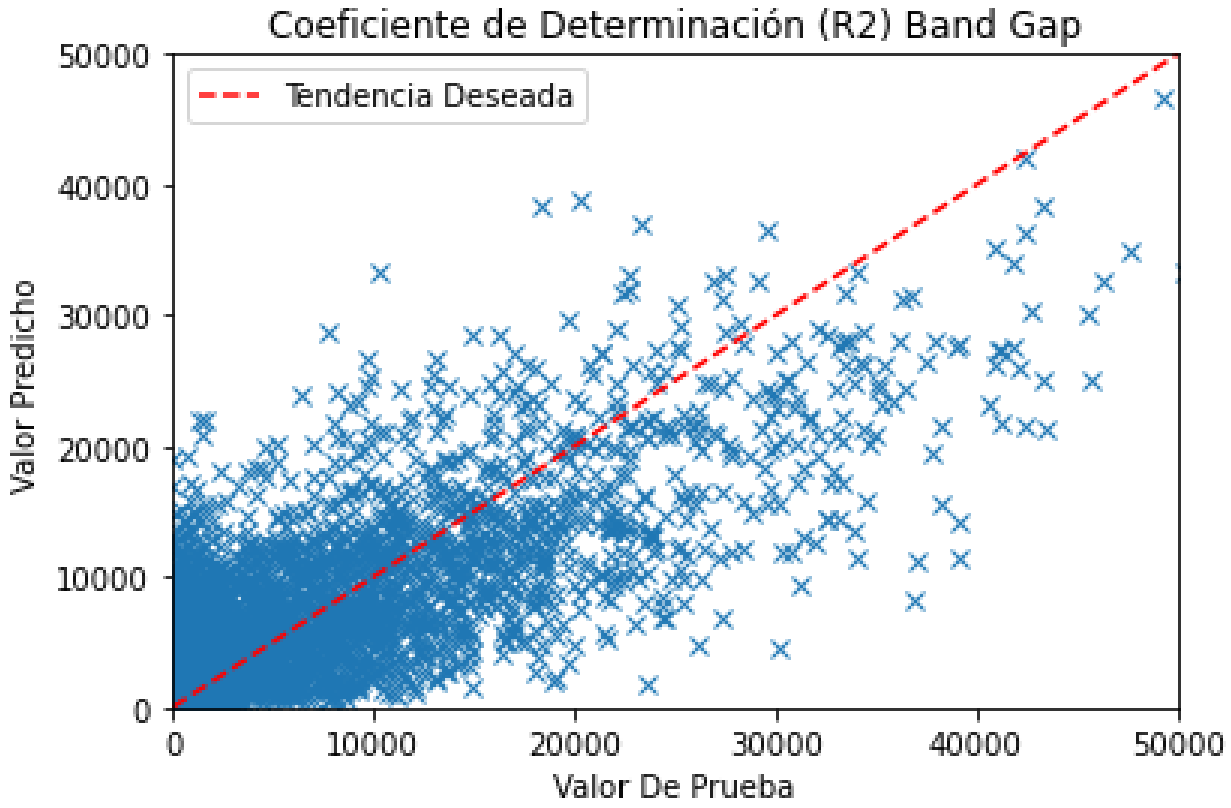

Figura 6.1.7: Rendimiento del algoritmo KNN para predecir el ancho de banda mediante el data set 1.

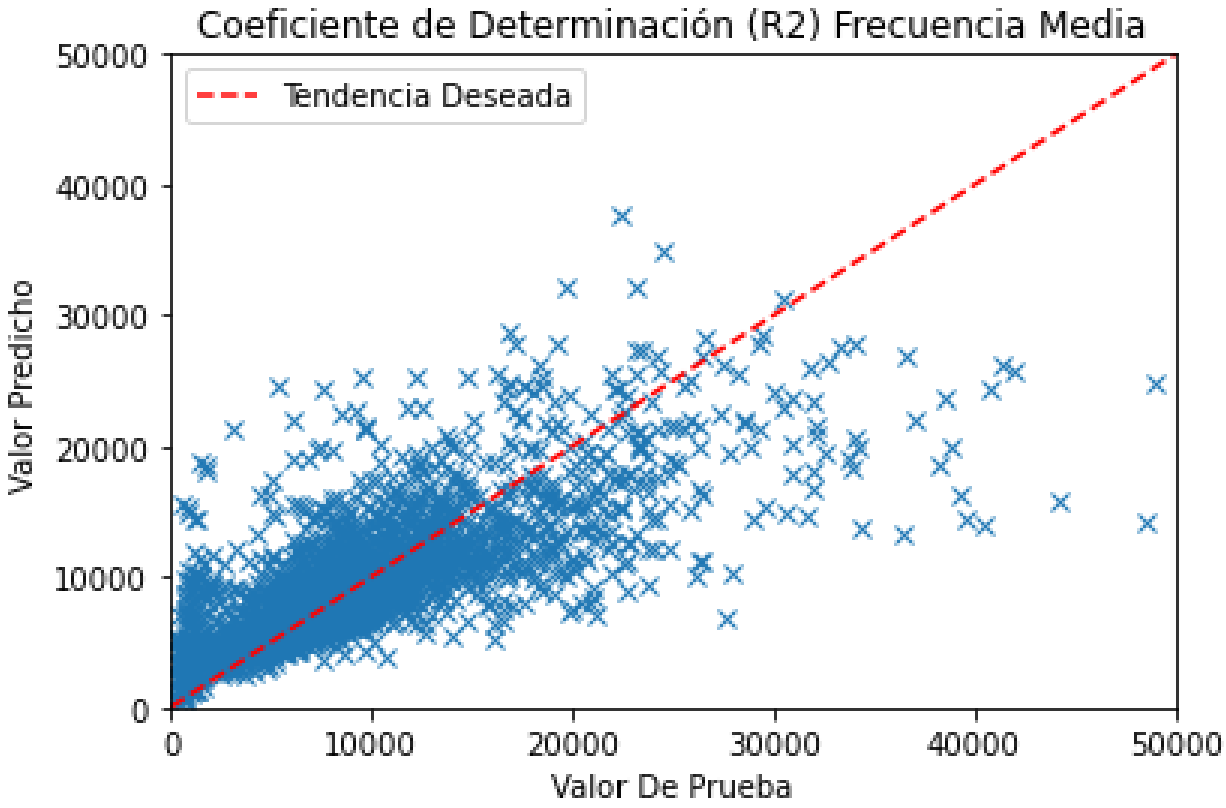

Figura 6.1.8: Rendimiento del algoritmo KNN para predecir la frecuencia media mediante el data set 2.

6.1.3. Random Forest

Los resultados obtenidos mediante random forest se pueden observar en las figuras 6.1.9 y 6.1.10. De aquí se puede observar que los coeficientes de determinación más altos se obtuvieron del data set número dos para predecir la frecuencia media y del data set 1 para el band gap. Cabe destacar que para este algoritmo los mejores resultados se obtuvieron mediante una optimización con el método grid search, con el cual se obtuvo que el número de arboles óptimo es 156 y además utilizando PCA en el caso de la predicción del band gap. En la figura 6.1.12 se logra observar que los datos se acercan bastante a la tendencia deseada.

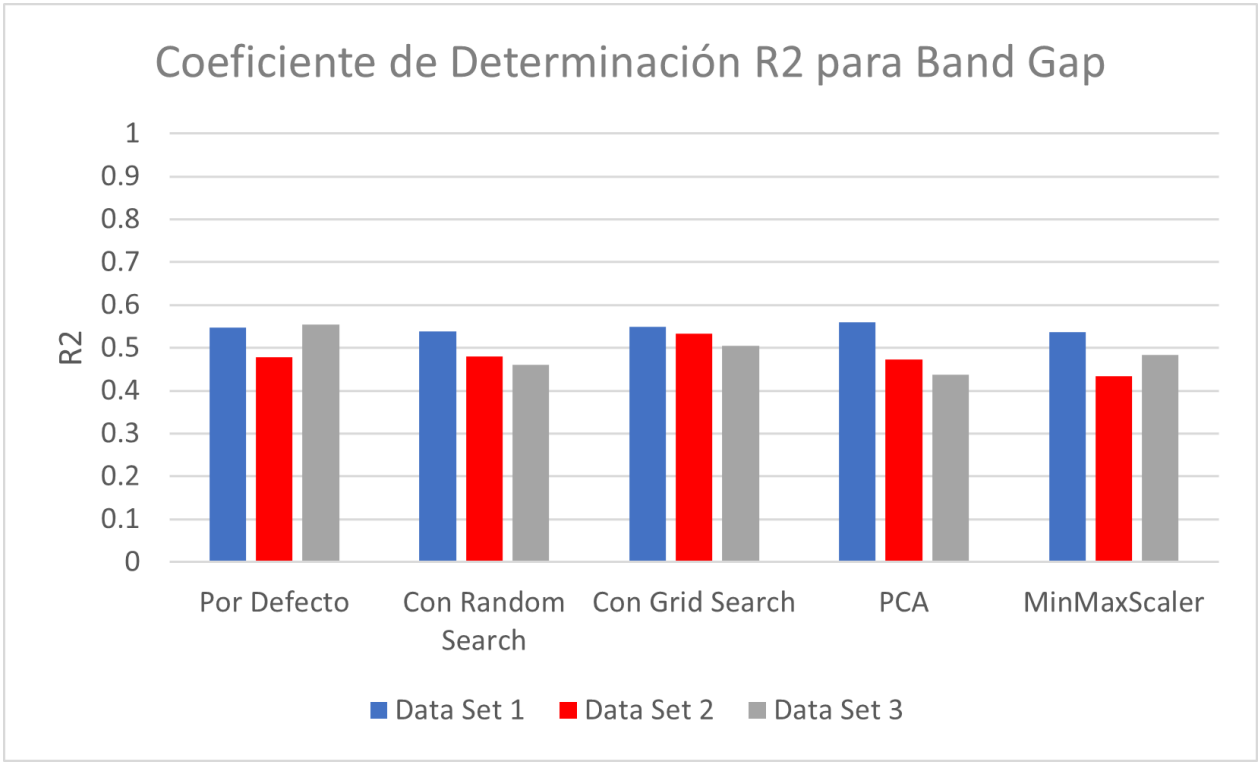

Figura 6.1.9: Coeficiente de Determinación obtenido para cada base de datos utilizando Random Forest para predecir el band gap.

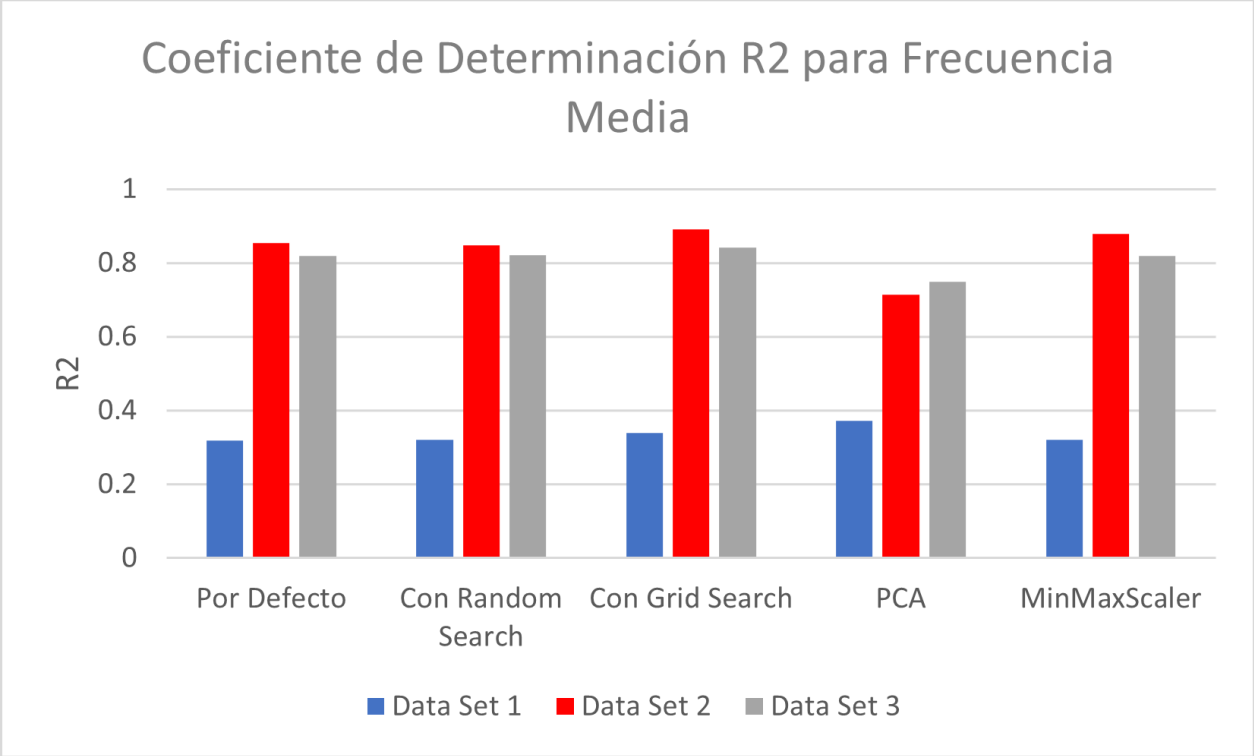

Figura 6.1.10: Coeficiente de Determinación obtenido para cada base de datos utilizando Random Forest para predecir la frecuencia media.

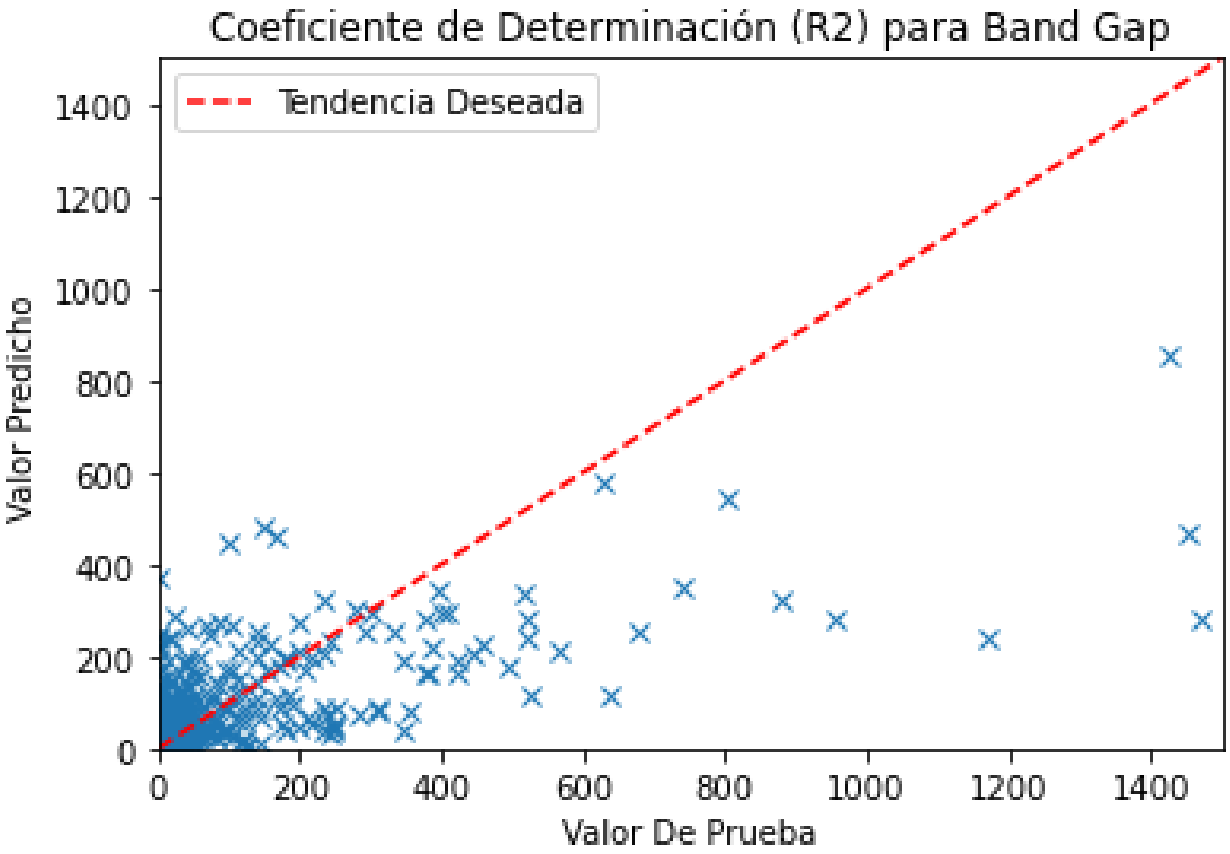

Figura 6.1.11: Rendimiento del algoritmo Random Forest para predecir el ancho de banda mediante el data set 1.

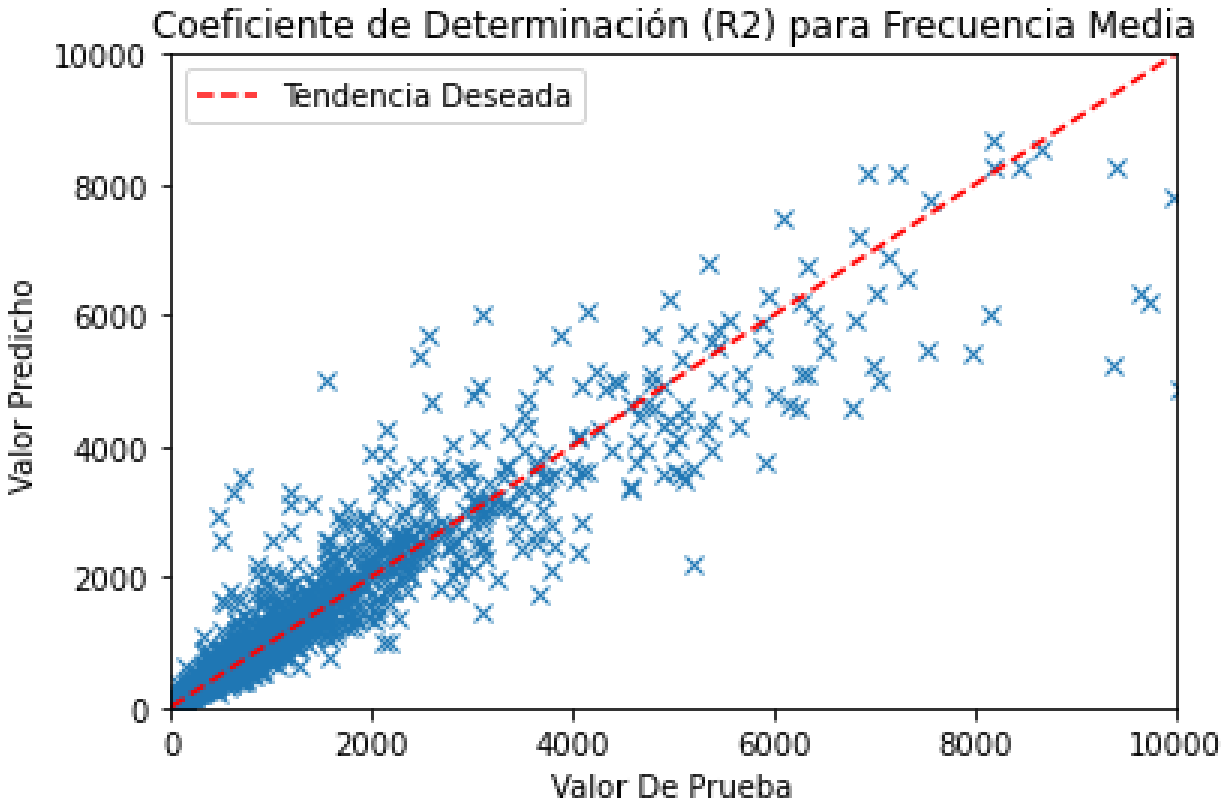

Figura 6.1.12: Rendimiento del algoritmo Random Forest para predecir la frecuencia media mediante el data set 2.

6.1.4. Adaptative Boosting

Debido al R2 obtenido utilizando Random Forest se decide utilizar Adaboost al ser otro algoritmo de ensamble de arboles de decisión. En las figuras 6.1.13 y 6.1.14 se logra observar que la base de datos que obtuvo mejor R2 fue el data set 3. El mejor resultado de R2 se obtuvo tras realizar grid search con lo cual se encontró que el número de estimadores óptimo es 2.

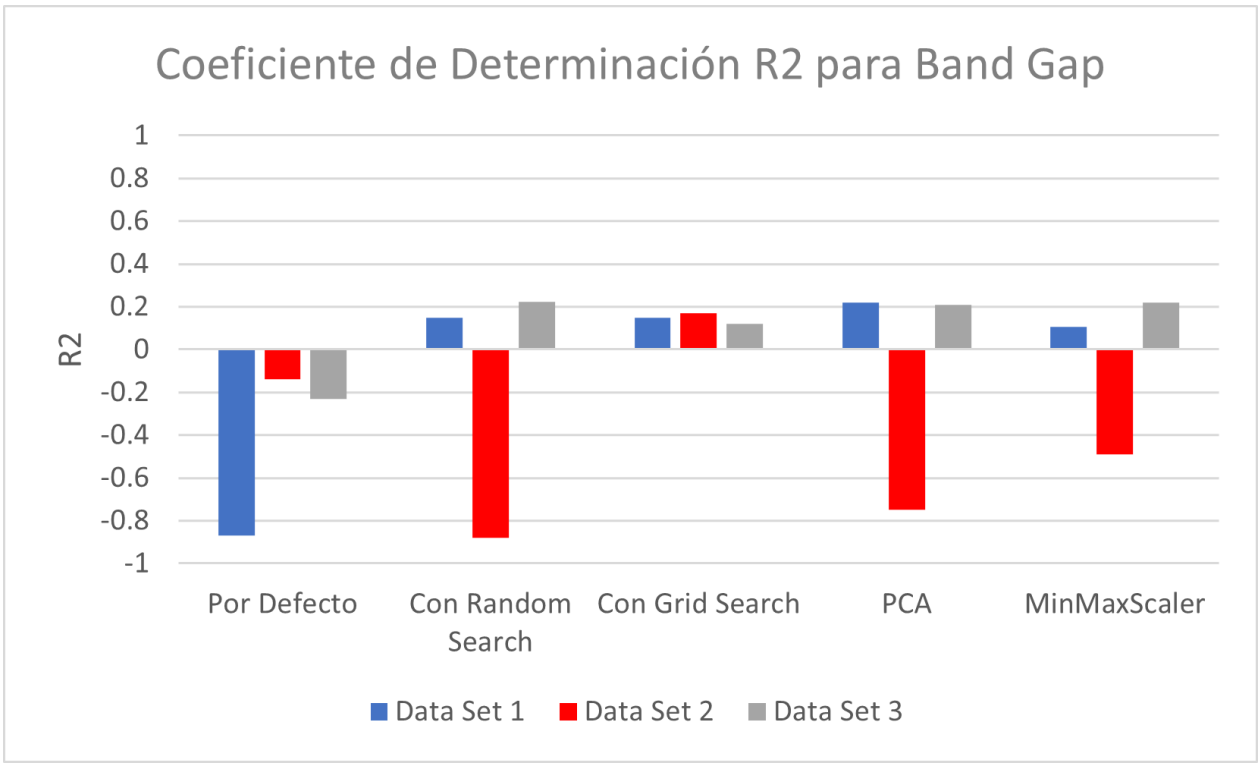

Figura 6.1.13: Coeficiente de Determinación obtenido para cada base de datos utilizando AdaBoosting para predecir el band gap.

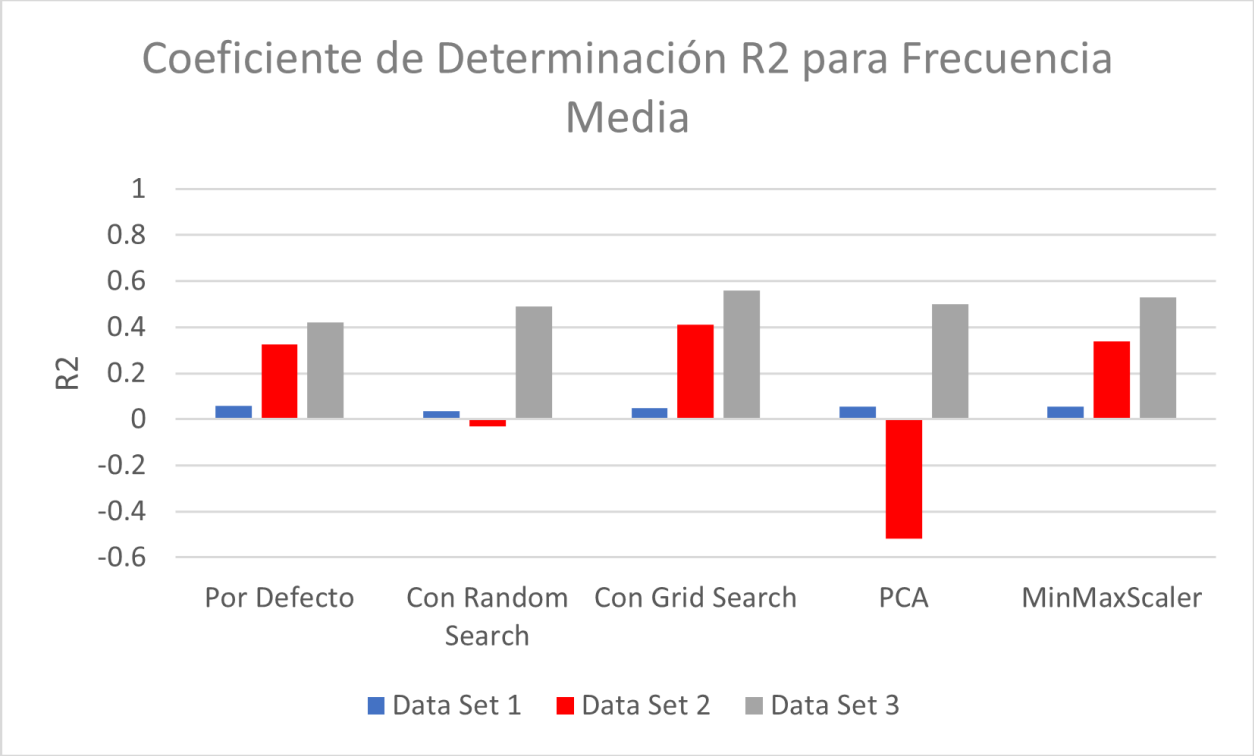

Figura 6.1.14: Coeficiente de Determinación obtenido para cada base de datos utilizando AdaBoosting para predecir la frecuencia media.

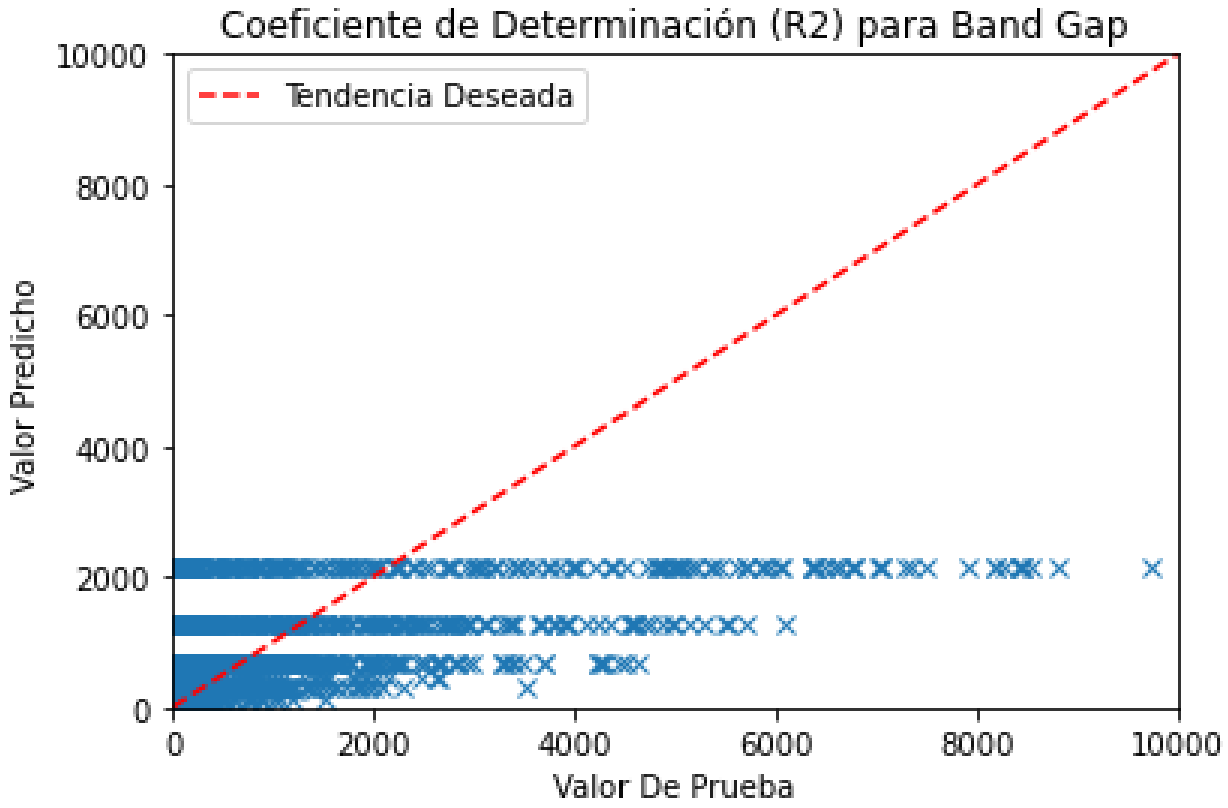

Figura 6.1.15: Rendimiento del algoritmo AdaBoost para predecir el ancho de banda mediante el data set 3.

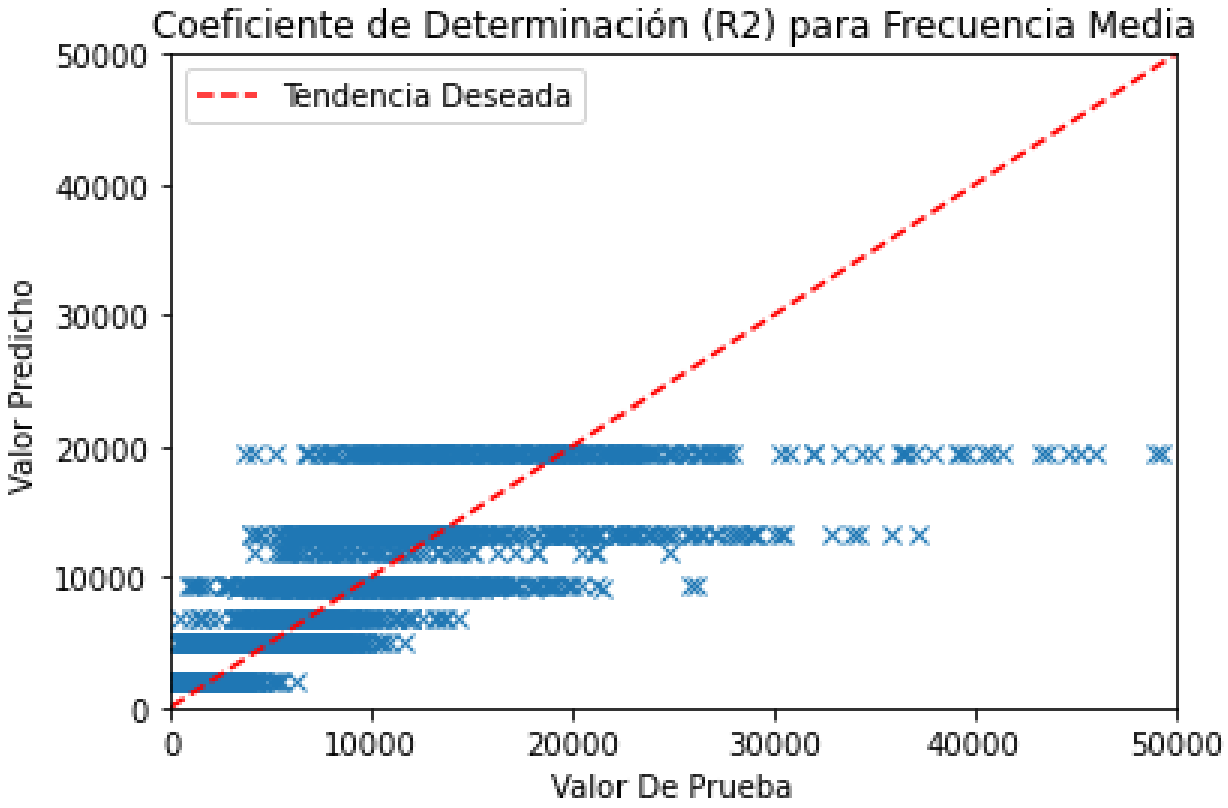

Figura 6.1.16: Rendimiento del algoritmo AdaBoost para predecir la frecuencia media mediante el data set 3.

6.1.5. Gradient Boosting

En las figuras 6.1.17 y 6.1.18 se puede observar que los valores más altos de R2 que se lograron obtener fueron mediante el uso del data set 1 para el band gap y el data set 2 para la frecuencia media. Cabe destacar que para ambos casos la optimización de hiper parámetros mediante grid search entregó resultados más altos de R2 pero además en el caso del ancho de banda tras utilizar PCA el resultado del R2 mejora nuevamente. Se puede observar esto de forma gráfica en las figuras 6.1.19 y 6.1.20

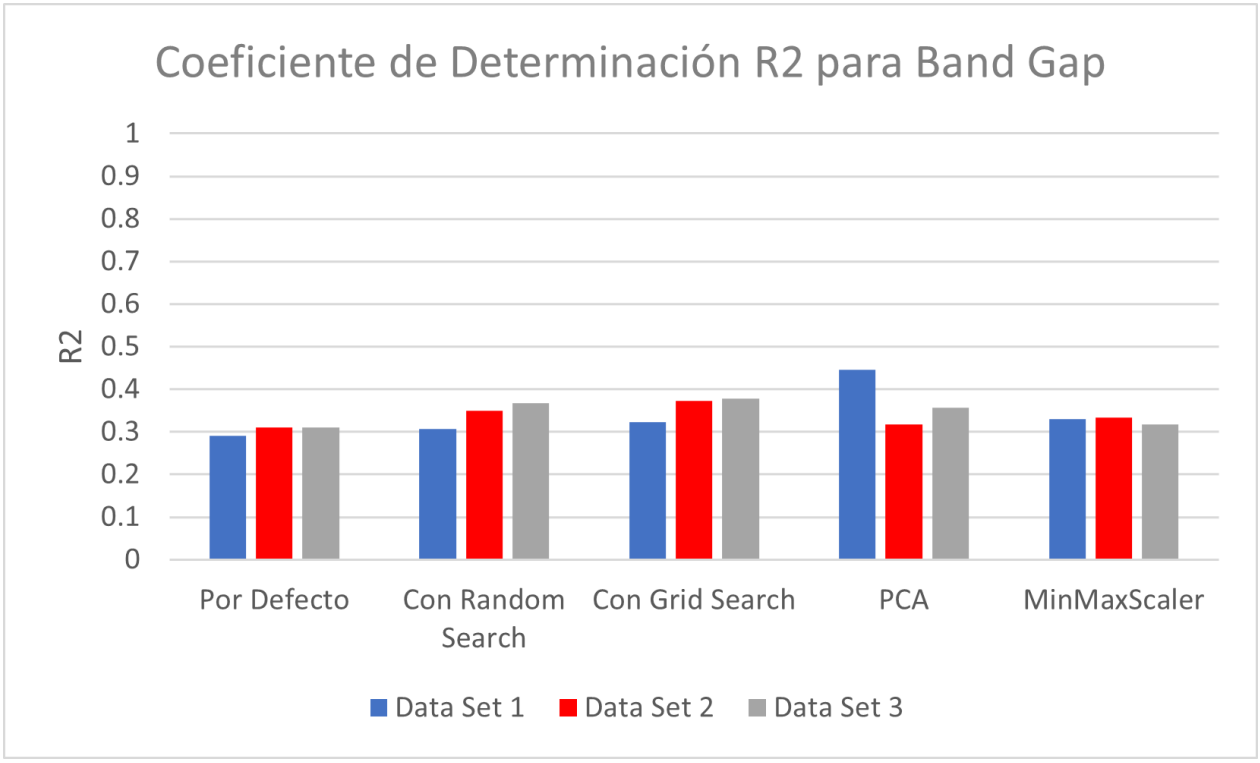

Figura 6.1.17: Coeficiente de Determinación obtenido para cada base de datos utilizando Gradient Boositng para predecir la el ancho de banda.

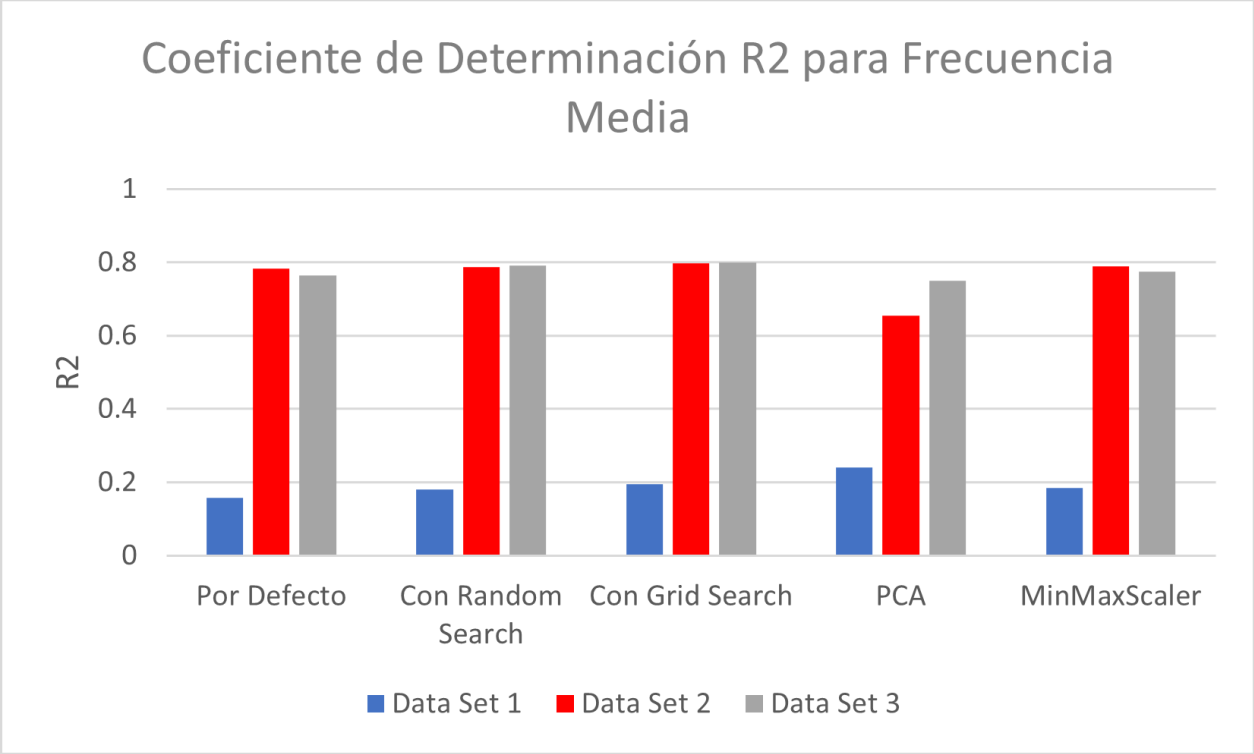

Figura 6.1.18: Coeficiente de Determinación obtenido para cada base de datos utilizando Gradient Boositng para predecir la frecuencia media.

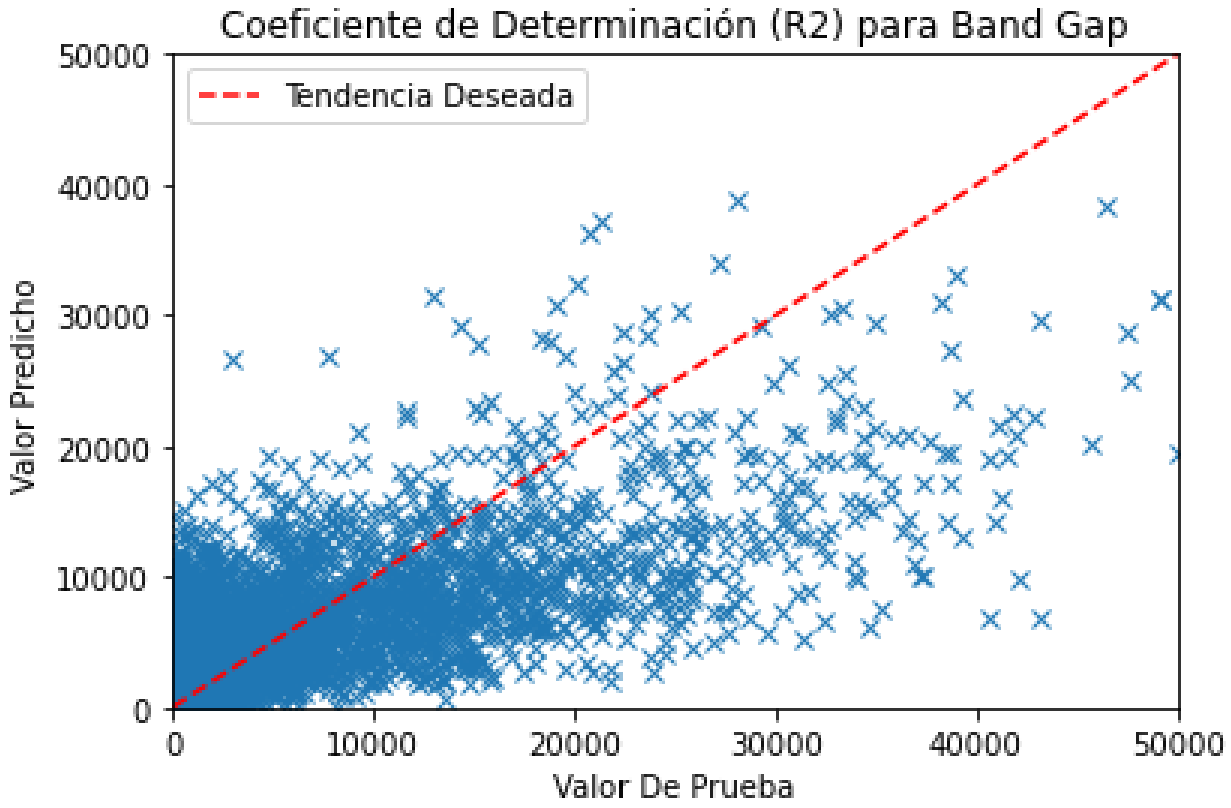

Figura 6.1.19: Rendimiento del algoritmo Gradient Boost para predecir la frecuencia media mediante el data set 1.

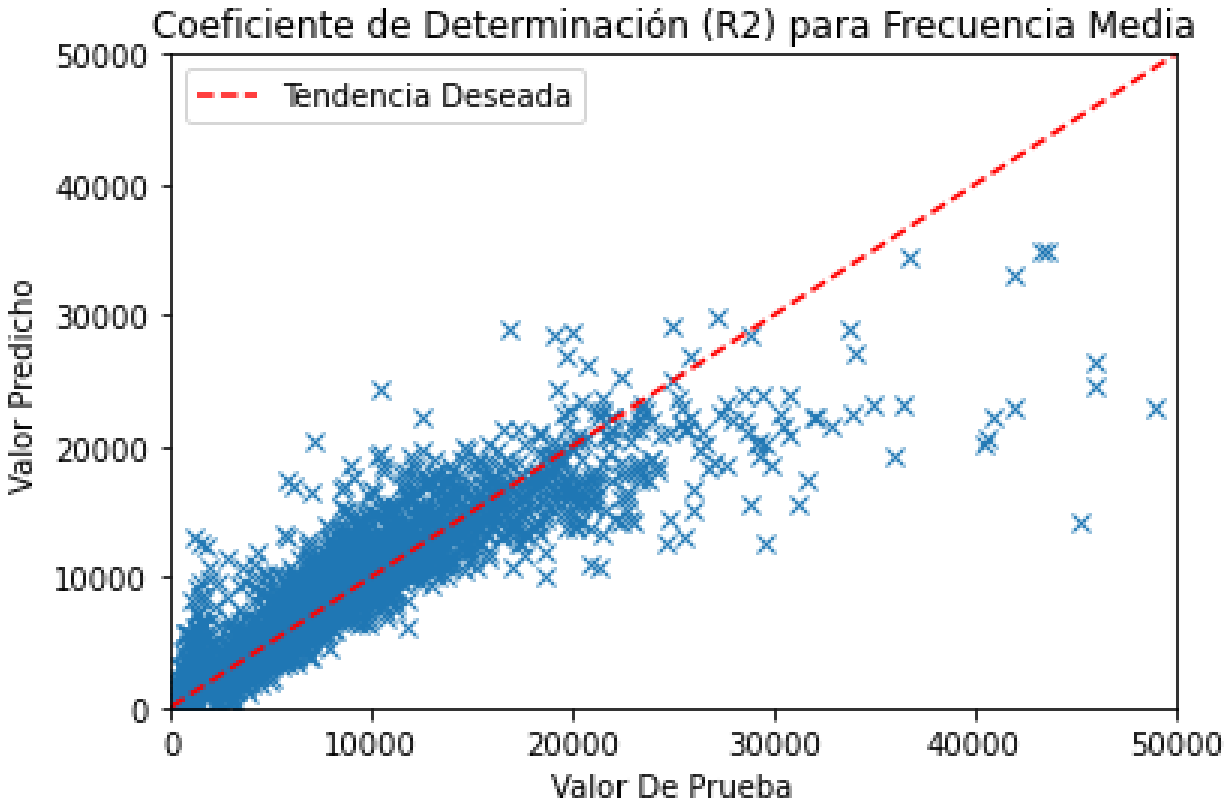

Figura 6.1.20: Rendimiento del algoritmo Gradient Boost para predecir la frecuencia media mediante el data set 2.

6.2. Segundo Enfoque

A continuación se presentan los resultados de R2 que se obtuvieron tras utilizar el algoritmo Random Forest mediante el segundo enfoque. Se utilizó este algoritmo debido a que fue el que presentó mejores resultados para R2. Cabe destacar que se utilizó grid search para optimizar el valor del número de estimadores el cual dio 180 como resultado para el data set 2 y 156 para el data set 3.

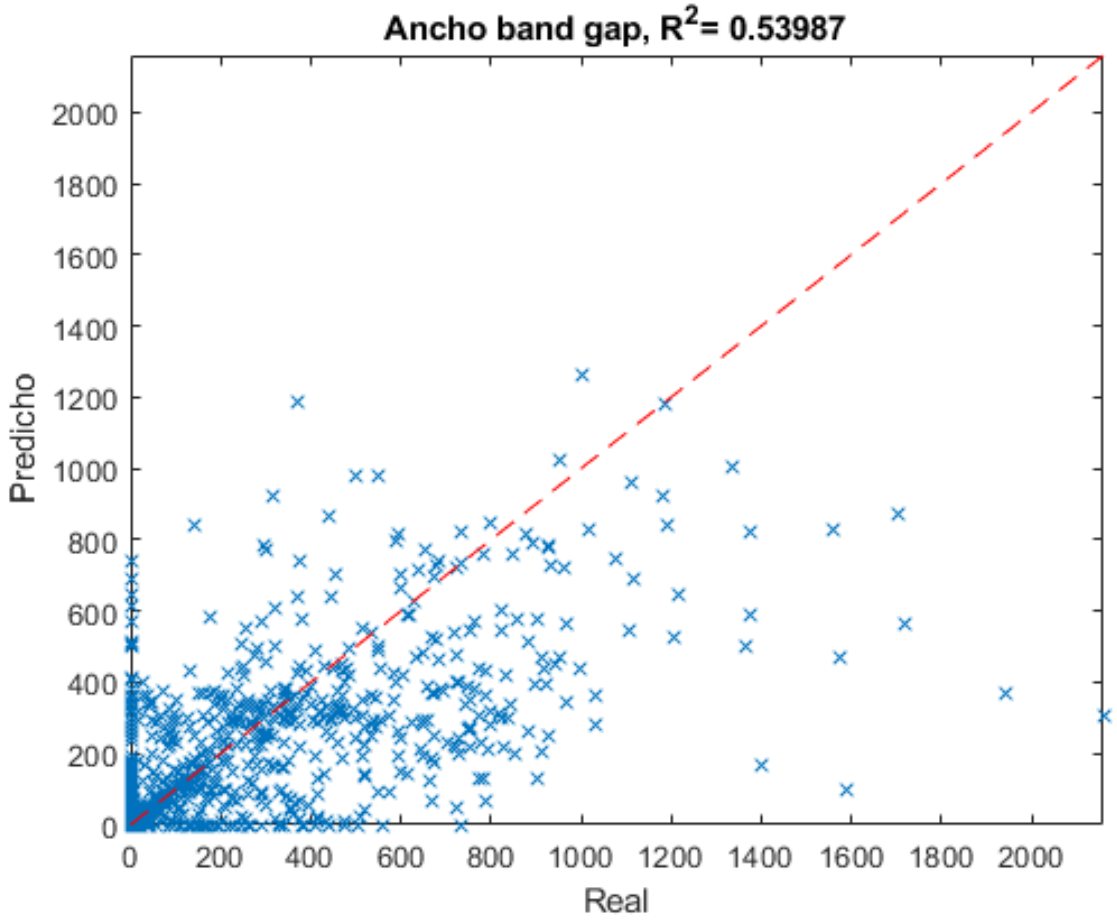

Figura 6.2.1: Coeficiente de determinación mediante el uso del set de datos 2 el para band gap.

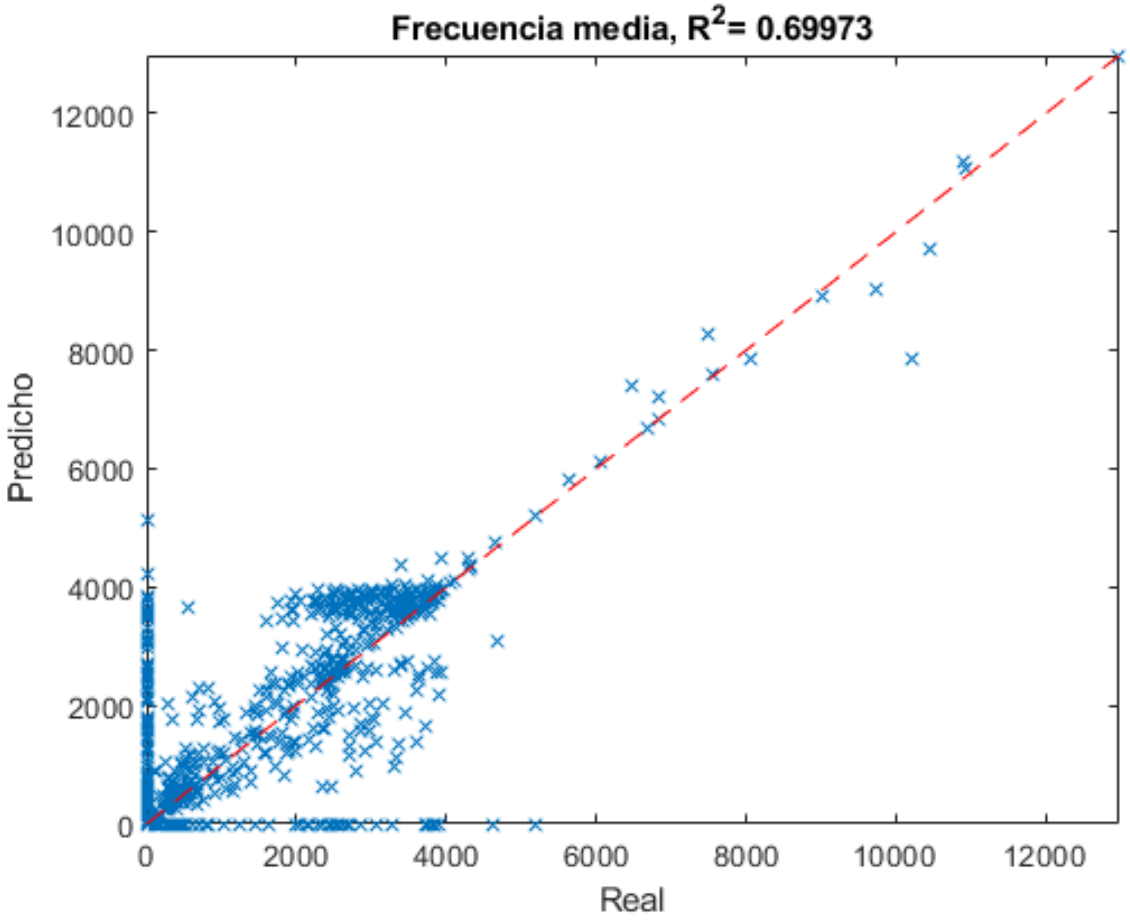

Figura 6.2.2: Coeficiente de determinación mediante el uso del set de datos 2 para la frecuencia media.

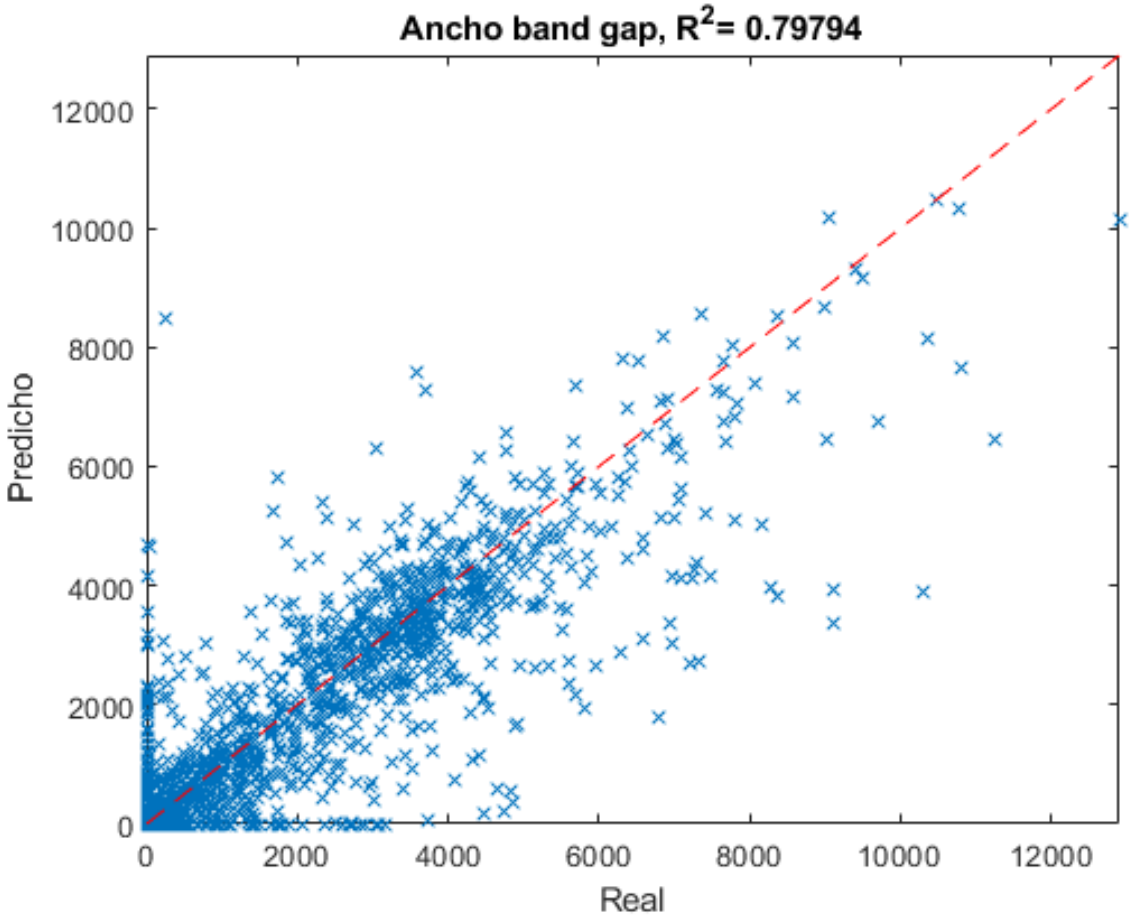

Figura 6.2.3: Coeficiente de determinación mediante el uso del set de datos 3 el para band gap.

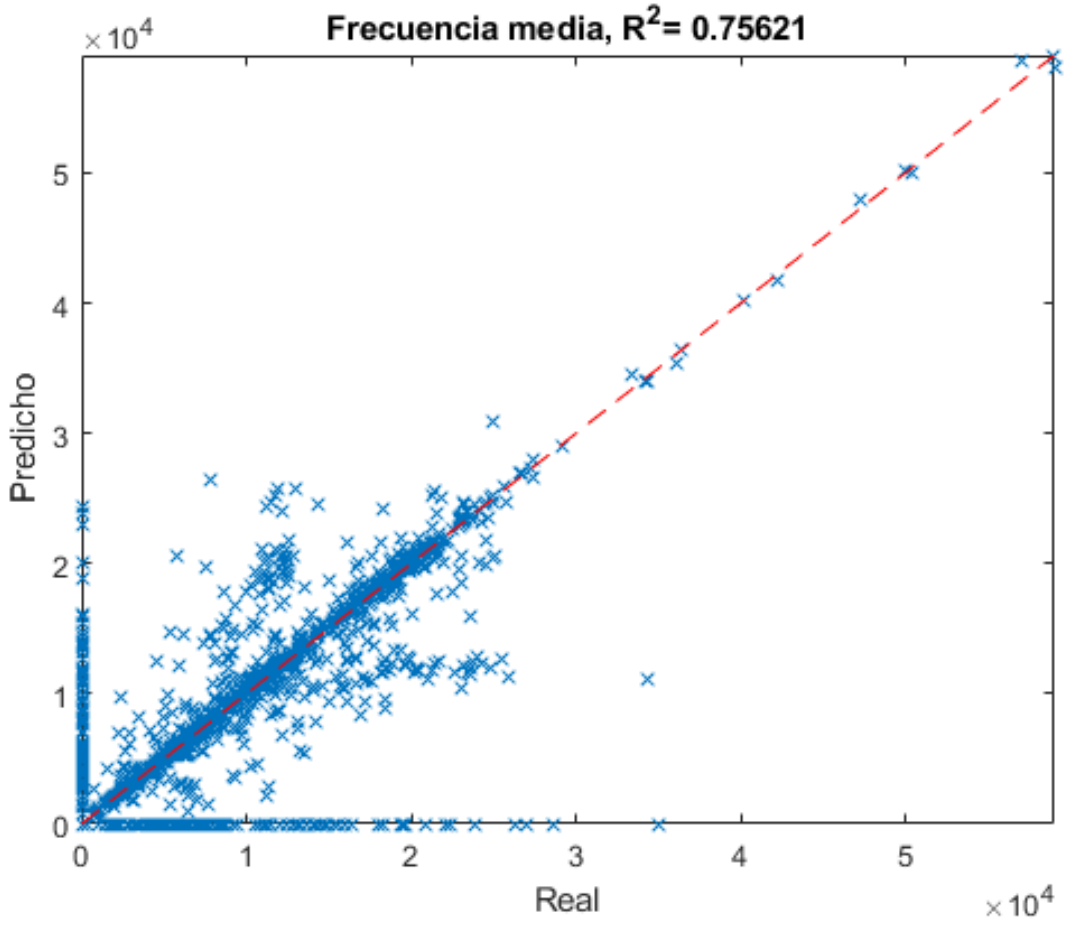

Figura 6.2.4: Coeficiente de determinación mediante el uso del set de datos 2 para la frecuencia media.

Capítulo 7

Discusión y Análisis

A partir de una primer acercamiento a los resultados obtenidos se puede observar que la precisión de los algoritmos según el coeficiente de determinación depende de como se ajuste la base de datos a estos de entrenamiento. Debido a esto se puede observar una disparidad en el resultado de los R2. Tomando el set de datos número uno el cual consistía en un modelo enrejado sin masas puntuales tiende a entregar mejores valores de R2 para la predicción de band gap mientras los data set número 2 y 3 entregan mejores resultados de R2 para las predicciones de la frecuencia media. Otro análisis rápido que se puede observar es que en general para estás bases de datos el método de optimización de parámetros grid search tuvo mejores resultados que el método random search. Esto ocurre debido a que este método toma en consideración todas las posibles combinaciones que se le entregan mientras que random search prueba con muestras aleatoria de la cantidad que se definen esto lleva a que el método sea menos eficiente a la hora de probar con varias dimensiones lo cual fue el caso para las 3 bases de datos las cuales contaban con 15 y 24 parámetros de entrada. Una desventaja que se debe mencionar del método grid search es que exige mayor rendimiento a la CPU que random search por lo que en ciertos casos se tuvo que limitar la cantidad de combinaciones. Se observa una poca consistencia del efecto que tuvo el método PCA y la normalización de datos con MinMaxScaler, por ejemplo si se observa la figura 6.1.10 se puede observar que el algoritmo obtuvo un R2 más alto utilizando solamente Grid Search a diferencia del algoritmo Gradient Boosting el cual para el data set 1 PCA mejoró el resultado como se puede observar en la figura 6.1.17. Esto tiene relación a que el método PCA modifica las variables de entrada, ya sea reduciendo las dimensiones o simplemente cambiando los valores a valores más representativos lo que puede ajustarse bien dependiendo el algoritmo y la base de datos en la cual se esta trabajando, hay que recordar que en apredizaje de máquinas los cambios de los parámetros de entrada afecta bastante en los resultados finales.

7.1. Primer Enfoque: Support Vector Machine y K Nearest Neighbors

Observando las figuras 6.1.1 y 6.1.2 se puede notar que este algoritmo no se ajusta bien a los datos, en particular para predecir los anchos de banda debido a que arroja valores

negativos de R2. Esto ocurre debido a que el hiper plano de corte con el cual SVN realiza las predicciones no se ajusta lo suficientemente bien a los set de datos utilizados. Cabe destacar que mediante el data set número 3 se logra un R2 positivo de aproximadamente 0.43, esto no es un resultado satisfactorio pero como se observa en la figura 6.1.4 parte de los datos sigue la tendencia deseada.

Si se observan las curvas 6.1.7 y 6.1.8 se puede apreciar que este k nearest neighbors, si bien no predice con mucha exactitud, si tiende a formar una regresión que se ajusta a la tendencia. A diferencia de SVN este algoritmo tiene bastante más precisión debido a que trabaja comparando datos para generar las predicciones por lo que la estructura de la base de datos no juega un rol tan importante como en SVN, en el cual si el plano de corte no se ajusta a los datos se obtienen malos resultados. Cabe destacar que si bien este algoritmo sigue la tendencia deseada existen varias predicciones que quedan fuera de esta tendencia, una de las razones que puede provocar esto es que el rendimiento de este algoritmo se ve afectado por la cantidad de parámetros de entrada.

7.2. Primer Enfoque: Ensamble Learning

Como se sabe los arboles de decisión son de los algoritmos con mayor rendimiento a la hora de realizar predicciones, pero tienen un costo en cuanto al tiempo de computación. Debido a esto se decide trabajar con los tres tipos de ensambles de algoritmos más comunes de arboles de decisión, los cuales al ser ensambles tienden a entregar mayor rendimiento. Si se observan desde las figuras 6.1.9 hasta 6.1.20 se puede observar que tanto Random Forest como Gradient Boost son los algoritmos que mejor se ajustan a la linea de tendencia deseada en la predicción de la frecuencia media, de igual modo son de los algoritmos con mayor constancia y rendimiento para predecir el ancho de banda. Esto se debe al carácter de entrenamiento de random forest el cual entrena varios arboles de decisión los cuales de por si son algoritmos potentes y entrega un promedio de estos. En cuanto a Gradient Boost se puede apreciar que debido a que el apredizaje secuencial en el cual utiliza los errores para mejorar los resultados de forma directa juegan un rol importante en entregar buenas predicciones. Finalmente se logra observar que el algoritmo de ensamble Adaptive Boosting deja mucho que desear a diferencia de los otros mencionados anteriormente, entregando resultados negativos y curvas que no siguen la tendencia deseada. Una de las razones por las cuales esto pudo haber sucedido es que este algoritmo no aprende de forma directa de los errores como Gradient Boost, si no que le entrega ciertos valores a los errores los cuales va modificando en el proceso de apredizaje.

7.3. Segundo Enfoque

Para este enfoque se trabajó utilizando las bases de datos que consistían en una estructura periódica con resonadores y un enrejado con masas puntuales debido a que fueron las estructuras que entregaron mejores resultados. Si se comprara en los gráficos 6.2.1 y 6.2.2 con los resultados obtenidos en las figura 6.1.9 y 6.1.10 para las bases de datos 2 mediante el primer enfoque, se logra identificar que el valor de R2 para la frecuencia media disminuyó y para el ancho de banda se mantuvo constante. En primer lugar esto resulta extraño debido a que se esperaba tener un aumento el los desempeños debido a que hay menor perdida de información

al predecir las bandas en lugar de saltarse el paso de calculo entremedio e ir directamente al ancho y frecuencia. Se especula a que esto ocurre debido a que al tener mayor cantidad de datos de salida hay mas tendencia a tener errores en las predicciones. Por otra parte se logra observar en las figuras 6.2.3 y 6.2.4 las cuales se obtuvieron utilizando la tercera base de datos como conjunto de entrenamiento y prueba. Para esto se obtuvieron valores de R2 elevados tanto para el band gap como para la frecuencia media, llegando a obtener un R2 igual a 0.797 para el ancho de banda, el cual fue el más alto de todos los resultados. Esto va de acuerdo a lo esperado debido a que se utilizó random forest el cual es reconocido por tener buen desempeño y mediante el segundo enfoque hubo menor perdida de información por lo que se entrenó de manera exitosa el algoritmo. Cabe destacar que por los resultados obtenidos en este enfoque se puede identificar que la base de datos 3 se ajusta mejor a esta metodología que la base de datos 2.

Capítulo 8

Conclusiones

En este trabajo de investigación se logro desarrollar una metodología favorable para poder predecir el ancho de banda en paneles de metamateriales mediante algoritmos de aprendizaje de máquinas. Se generaron bases de datos a través de un modelo de panel enrejado, modelos periódico con resonadores y la combinación de los dos anteriores el cual dio como resultado un panel enrejado con masas puntuales en las vigas. Utilizando las bases de datos mencionadas se obtuvieron los desempeños medidos utilizando el coeficiente de determinación como métrica para los siguientes algoritmos: support vector machine, k nearest neighbors, random forest, adaptative boosting y gradient boosting. Se logró encontrar algoritmos los cuales se ajustaban bien a las bases de datos utilizadas entregando buenos resultados en relación a las predicciones deseadas, las cuales se lograron justificar mediante los temas tratados en detalle en los antecedentes de este trabajo de investigación entregando así resultados válidos. Como observación general se concluyó que las bases de datos tienen una fuerte influencia en como se ajustan los algoritmos a ellas para realzar las predicciones.

Una de las desventajas que presenta el algoritmo de support vector machine es que es muy sensible a la estructura de datos debido a que no siempre se puede encontrar un hiper plano que se ajuste a ellos. Debido a esto los resultados obtenidos utilizando este algoritmo no tuvieron un buen desempeño llegando a valores negativos de R2. Por otra parte, KNN otro algoritmo básico de machine learning entregó mejor desempeño pero no lo suficiente debido a que no supero un R2 del 0.54, esto quiere decir que lograba predecir con exactitud el 54 por ciento de los datos aproximadamente. El uso de los algoritmos de apredizaje de máquinas mostró entregar resultados bastante favorables para los algoritmos de ensamble Random Forest y Gradient Boosting. Cabe destacar que para el algoritmo de ensamble Adaptative Boositng los resultados de R2 fueron bastante bajos, se especula que el sistema de entregar valor a las predicciones erradas no se ajuste bien a los modelos utilizados a diferencia de Gradient Boosting el cual entrena con el error de forma directa.

Se logró observar que el método grid search para optimizar hiper parámetros se ajusto bastante bien a todos los modelos utilizados. Tomando en cuenta que dentro una de las motivaciones de este trabajo es lograr encontrar los anchos de banda de manera más rápida, es necesario indicar que grid search tiene un costo de tiempo alto, por lo que se debe realizar un trade-off para evaluar si efectivamente es de vitalidad utilizarlo.

Utilizando un segundo enfoque el cual consistía en predecir las bandas en lugar de predecir directamente el ancho de bandas y la frecuencia media, se logró identificar que entregó resultados más constantes y elevados de R2. Esto ocurre debido a que este método trabaja con mayor información y como se puede observar en el marco teórico la perdida de información tiene una efecto negativo en las predicciones realizadas, lo cual ocurre si se predice de forma directa el ancho y la frecuencia media. Mediante este enfoque se logró concluir que la base de datos número 3 se ajustaba mejor a la metodología, entregando los valores más altos de R2 para el ancho de banda.

8.1. Trabajos Futuros

Se propone como trabajo futuro experimentar con diferentes ensambles de algoritmos debido a que este método entrego valores de R2 bastante altos llegando a un rendimiento de 0.89. Como se pudo apreciar en esta investigación el uso de algoritmos de apredizaje de máquinas resulta bastante complejo debido a la gran cantidad de factores que influyen en sus rendimientos, por lo que se propone trabajar con diferentes métricas, bases de datos, métodos de optimización y preprocesamiento de datos.

Bibliografía

[1] Vincenzo D'Alessandro, Giuseppe Petrone, Francesco Franco, and Sergio De Rosa. A review of the vibroacoustics of sandwich panels: Models and experiments. *Journal of Sandwich Structures & Materials*, 15(5):541–582, 2013.

[2] Jack R Vinson and Robert L Sierakowski. *The behavior of structures composed of composite materials*, volume 105. Springer, 2006.

[3] A Srikantha Phani, J Woodhouse, and NA Fleck. Wave propagation in two-dimensional periodic lattices. *The Journal of the Acoustical Society of America*, 119(4):1995–2005, 2006.

[4] Camilo Valencia, Juan Gomez, and Nicolás Guarín-Zapata. A general-purpose element-based approach to compute dispersion relations in periodic materials with existing finite element codes. *Journal of Theoretical and Computational Acoustics*, 28(01):1950005, 2020.

[5] Chengxing Yang and QM Li. Advanced lattice material with high energy absorption based on topology optimisation. *Mechanics of Materials*, 148:103536, 2020.

[6] M Mazur, M Leary, M McMillan, S Sun, D Shidid, and Milan Brandt. Mechanical properties of ti6al4v and alsi12mg lattice structures manufactured by selective laser melting (slm). In *Laser Additive Manufacturing*, pages 119–161. Elsevier, 2017.

[7] Dionisio Del Vescovo and Ivan Giorgio. Dynamic problems for metamaterials: review of existing models and ideas for further research. *International Journal of Engineering Science*, 80:153–172, 2014.

[8] Guilian Yi and Byeng D Youn. A comprehensive survey on topology optimization of phononic crystals. *Structural and Multidisciplinary Optimization*, 54(5):1315–1344, 2016.

[9] Guilian Yi, Yong Chang Shin, Heonjun Yoon, Soo-Ho Jo, and Byeng D Youn. Topology optimization for phononic band gap maximization considering a target driving frequency. *JMST Advances*, 1(1):153–159, 2019.

[10] Xian-Da Zhang. Machine learning. In *A Matrix Algebra Approach to Artificial Intelligence*, pages 223–440. Springer, 2020.

[11] Yunsheng Song, Jiye Liang, Jing Lu, and Xingwang Zhao. An efficient instance selection

algorithm for k nearest neighbor regression. *Neurocomputing*, 251:26–34, 2017.

[12] A Singh. A practical introduction to k-nearest neighbors algorithm for regression (with python code). *Accès à https://www. analyticsvidhya. com/blog/2018/08/k-nearestneighbor-introduction-regression-python*, 2018.

[13] Danny Varghese. Comparative study on classic machine learning algorithms. *Quick summary on various ML algorithms*, 2018.

[14] Aurélien Géron. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems*. O'Reilly Media, 2019.

[15] KP Soman, R Loganathan, and V Ajay. *Machine learning with SVM and other kernel methods*. PHI Learning Pvt. Ltd., 2009.

[16] Lujing Chen. Basic ensemble learning (random forest, adaboost, gradient boosting)-step by step explained. *Towards Data Science*, 2019.

[17] Matthias Feurer and Frank Hutter. Hyperparameter optimization. In *Automated Machine Learning*, pages 3–33. Springer, Cham, 2019.

[18] Hervé Abdi and Lynne J Williams. Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4):433–459, 2010.

[19] Jorge Sola and Joaquin Sevilla. Importance of input data normalization for the application of neural networks to complex industrial problems. *IEEE Transactions on nuclear science*, 44(3):1464–1468, 1997.

[20] Hera Shaheen, Shikha Agarwal, and Prabhat Ranjan. Minmaxscaler binary pso for feature selection. In *First International Conference on Sustainable Technologies for Computational Intelligence*, pages 705–716. Springer, 2020.

[21] Peter A Flach. The geometry of roc space: understanding machine learning metrics through roc isometrics. In *Proceedings of the 20th international conference on machine learning (ICML-03)*, pages 194–201, 2003.

[22] Scikit-Learn Developers. 3.3. metrics and scoring: quantifying the quality of predictions. In *Scikit-Learn 0.22. 1 Documentation. 2019*. 2019.

[23] Luke J Saunders, Richard A Russell, and David P Crabb. The coefficient of determination: what determines a useful r2 statistic? *Investigative ophthalmology & visual science*, 53(11):6830–6832, 2012.

[24] Michael Stein. Large sample properties of simulations using latin hypercube sampling. *Technometrics*, 29(2):143–151, 1987.