Predicción de Propiedades Moleculares

Linda Vanessa Ramos.
Julián Andrés Rosero
Santiago Salazar Correa

Introducción a la Inteligencia Artificial

Raúl Ramos Pollán.

Universidad de Antioquia Sede Medellín

Programa de Bioingeniería.

Facultad de Ingeniería.

Mayo 28, 2023

Contenido.

1. Introducción.

La constante escalar de acoplamiento es un parámetro importante en varias áreas de la ciencia y la ingeniería, incluyendo la física de partículas, la física cuántica, la teoría de campos y la teoría de la relatividad, esta hace una descripción cuantitativa de las interacciones entre átomos de una molécula, así como realizar predicciones y cálculos precisos para la corroboración teorías físicas. También permite analizar la interacción entre los átomos, mostrando la relación magnética entre los átomos de una molécula. La utilidad de una adecuada predicción de esta constante permite maximizar aplicaciones como por ejemplo, el procesamiento de imágenes con medios de contraste, como la Resonancia Magnética Nuclear (RMN), arrojando información sobre interacción proteica y composición molecular y estructural de tejidos. entre otros. En el caso de la RMN, que es una aplicación directamente ligada a la bioingeniería, la información del espectro de resonancia se da por desplazamiento químico y acoplamiento escalar, en el caso de éste último, analizando si la interacción es homonuclear (dos átomos del mismo elemento) o heteronuclear (dos átomos de elemento diferente), dando un espectro de frecuencia específico. [1] [2]

En el presente proyecto se pretende realizar la predicción de la constante escalar de acoplamiento correspondiente a la interacción entre dos átomos en una molécula determinada. Para dicho propósito se hará uso de los datos entregados en la competencia de kaggle "*Predicting Molecular Properties*" *(https://www.kaggle.com/c/champs-scalar-coupling)*. A partir de los datasets entregados se extraerá y analizará las distintas características sobre las moléculas y sus respectivos átomos con el objetivo de encontrar una relación entre las mismas, el cual permita dar con un modelo que logre capturar adecuadamente la naturaleza de los datos y la forma en que estos se distribuyen.

2. Planteamiento del problema.

A pesar de la importancia de la constante de acoplamiento escalar en la física, su determinación precisa en diferentes sistemas presenta desafíos y preguntas abiertas. En este contexto, surge la necesidad de investigar y analizar los siguientes aspectos relacionados con la constante de acoplamiento escalar:

1. Caracterización y medición precisa: ¿Cómo se puede caracterizar y medir con precisión la constante de acoplamiento escalar en diferentes sistemas físicos? ¿Existen métodos experimentales o teóricos específicos para determinar su valor?

2. Dependencia de las condiciones: ¿Cómo varía la constante de acoplamiento escalar en función de las condiciones del sistema, como la temperatura, la presión, el campo magnético u otras variables relevantes? ¿Existen relaciones o leyes que describen estas dependencias?

3. Relaciones con otras constantes: ¿Existen relaciones o conexiones entre la constante de acoplamiento escalar y otras constantes físicas fundamentales, como la constante de Planck, la velocidad de la luz o la carga elemental? ¿Cómo se pueden investigar y explorar estas relaciones?

4. Efectos en el comportamiento físico: ¿Cómo influye la constante de acoplamiento escalar en el comportamiento y las propiedades de los sistemas físicos, como las interacciones de partículas, las transiciones de fase o la propagación de ondas? ¿Cómo se puede modelar y estudiar su efecto en estos fenómenos?

2.1. Exploración del DataSet.

El dataset *"Predicting Molecular Properties"*, encontrado en kaggle cuenta con 131 mil archivos y 47 columnas con información relacionada a propiedades de los átomos iterados. Dentro de estas propiedades se encuentra el índice atómico, momento dipolar, descompuesto como vectores en ejes coordenados, las cargas de Mulkien, energía potencial y campos tensores magnéticos.

Las propiedades mencionadas anteriormente generan contribuciones a la constante de acoplamiento escalar. Así mismo, el dataset cuenta con datos de constante de acoplamiento reportadas propiedades de otras moléculas para para moléculas comunes y se ofrece las realizar la predicción.

Así mismo, el documento cuenta con dataframes de entrenamiento (Train) el cual contiene información sobre las propiedades mencionadas anteriormente y valor estimado de la constante de acoplamiento; otro dataframe de prueba (Test), que contiene las propiedades moleculares, pero no el valor de la constante, puesto que es el valor que se desea encontrar en este proyecto; y un último dataframe de estructura (Structure) con información adicional sobre las moléculas de estudio. lo anteriormente mencionado se encuentra descrito a continuación, entre información adicional encontrada en el dataset *'champs-scalar-coupling*':

- **Momento dipolar *(dipole_moments.csv)***
- Contribución al escalar coupling *(scalar_coupling_contributions.csv)*
- Cargas mulliken *(mulliken_charges.csv)*

- **Energía potencial *(potential_energy.csv)***
- Tensores de blindaje magnético *(magnetic_shielding_tensors.csv)*
- **Datos de entrenamiento *(Train.csv)***
- **Datos de testeo *(Test.csv)***
- **Estructuras *(structures.csv)***

Cuando se realizó el análisis de cada uno de los data frames existentes se encontró que:

- **Train:** Contiene información como el nombre de la molécula, el tipo de interacción que tienen estas, es decir interacción átomo-átomo, estás presentan todas las posibles combinaciones de iteraciones entre los átomos que se están estudiando en las diferentes moléculas, los índices de los átomos y el valor de la constante de acoplamiento, siendo este el valor relevante para el estudio, como se puede observar en la figura 1.

	id	molecule_name	atom_index_0	atom_index_1	type	scalar_coupling_constant
0	0	dsgdb9nsd_000001	1	0	1JHC	84.80760
1	1	dsgdb9nsd_000001	1	2	2JHH	-11.25700
2	2	dsgdb9nsd_000001	1	3	2JHH	-11.25480

Figura 1. *Dataframe de Train.*

- **Test:** Contiene la misma información que *Train*, pero con la diferencia de que no está considerada la constante coupling, puesto que, está es quien estamos buscando.

- **Structure:** Posee información como nombre de molécula, el tipo de átomo presente en la molécula y sus coordenadas respectivamente, como lo indica la figura 2.

	molecule_name	atom_index	atom	x	y	z
0	dsgdb9nsd_000001	0	C	-0.012698	1.085804	0.008001
1	dsgdb9nsd_000001	1	H	0.002150	-0.006031	0.001976
2	dsgdb9nsd_000001	2	H	1.011731	1.463751	0.000277
3	dsgdb9nsd_000001	3	H	-0.540815	1.447527	-0.876644
4	dsgdb9nsd_000001	4	H	-0.523814	1.437933	0.906397

Figura 2. Dataframe de structure.

- **Dipole Moments:** Posee información de nombre de molécula e información del vector de momento dipolar de la molécula, descompuesto en coordenadas cartesianas (x,y,z) para las respectivas interacciones

átomo-átomo. Este momento dipolar es importante porque habla del comportamiento químico y físico que puede tener la interacción de átomos en una molécula, haciendo propiedades como su solubilidad.

	molecule_name	X	Y	Z
0	dsgdb9nsd_000001	0.0000	0.0000	0.0000
1	dsgdb9nsd_000002	-0.0002	0.0000	1.6256
2	dsgdb9nsd_000003	0.0000	0.0000	-1.8511
3	dsgdb9nsd_000004	0.0000	0.0000	0.0000

Figura 3. Dataframe de dipole moment.

- **Potential Energy:** Este data frame contiene información acerca del nombre de la molécula, la energía potencial de la molécula.

- **Mulliken Charges:** Posee información como el valor de mulliken, el cual entrega información de la distribución de la carga electrónica en una molécula.

- **Magnetic Shielding Tensors:** Aquí se tiene información como la interacción de átomo-átomo en diferentes coordenadas.

2.2. Métrica.

Para cuantificar el desempeño del modelo predictivo se utiliza el logaritmo medio del error absoluto, es decir, se calcula el logaritmo del error absoluto para cada acoplamiento y posteriormente se promedia entre los tipos de acoplamiento, descritos en el dataset.

$$SCORE = \frac{1}{T}\sum_{t=1}^{t} log(\frac{1}{nt}\sum_{i=1}^{nt}(|yi - \hat{y}i|))$$

Donde:

- T es el número de tipos de acoplamiento escalar.
- n_t es el número de observaciones de tipo.
- y_i es la constante de acoplamiento escalar real para la observación.
- $\hat{y}_i$ es la constante de acoplamiento escalar predicha para la observación

2.3. Variable objetivo.

Como se mencionó anteriormente, la variable objetivo que se desea predecir en este caso "*la constante escalar de acoplamiento*", la cual nos mostrará de forma útil el comportamiento de ciertas moléculas, que posteriormente se analizarán con el código proporcionado, *dt.score(X, y)* se utiliza para calcular la precisión del modelo de clasificación dt en base a los datos de entrada 'X' y las etiquetas objetivo 'y'. La función *score()* es un

método disponible en muchos modelos de aprendizaje automático en la biblioteca *scikit-learn* de Python.

En particular, para un modelo de clasificación como 'Decision Tree Classifier', la función *score()* calcula la precisión del modelo, que es la proporción de muestras correctamente clasificadas sobre el total de muestras. Para calcular la precisión, el modelo dt toma las características en 'X', realiza predicciones utilizando el método *predict()*, y luego compara esas predicciones con las etiquetas reales en 'y'.

En resumen, *dt.score(X, y)* devuelve un valor entre 0 y 1, donde 1 representa una precisión perfecta, es decir, todas las muestras se clasificaron correctamente, y valores más bajos indican una menor precisión en la clasificación de las muestras.

3. Exploración de variables.

Como se mencionó anteriormente, la constante de acoplamiento escalar depende de varias propiedades de los átomos, sin embargo, por cuestiones de simplicidad y tiempo, no se tuvo en cuenta todas las mencionadas, sino las enlistadas a continuación.

3.1. Índice atómico: en el dataset, el índice atómico funciona como elemento identificador de los átomos, estableciendo las relaciones entre un átomo fijo, y los demás átomos de la molécula estudiada.

3.2. Momento dipolar: el momento dipolar es la medida de la fuerza de atracción entre un par de átomos, por lo cuál tiene gran significancia en la cuantificación de la constante que se busca determinar. Por otra parte, al estar definido en coordenadas rectangulares, es más sencilla su interpretación y su manejo de datos.

3.3. Energía potencial: la energía potencial también tiene una alta contribución en el acoplamiento atómico, puesto que es la energía que define la interacción atómica, es decir, la atracción o repulsión de estos átomos. [3]

Una disminución del 1 % en el error absoluto medio para un tipo, representa una mejora del 1 % para los demás tipos de acoplamiento presentes. La idea es alcanzar el score más negativo posible (-20,7232 de acuerdo a la predicción del dataset), puesto que esto representaría un acercamiento casi exacto al valor de la constante de acoplamiento.

En la competencia de kaggle donde se encuentra el dataset, el score ganador hasta el momento es de -3,23968.

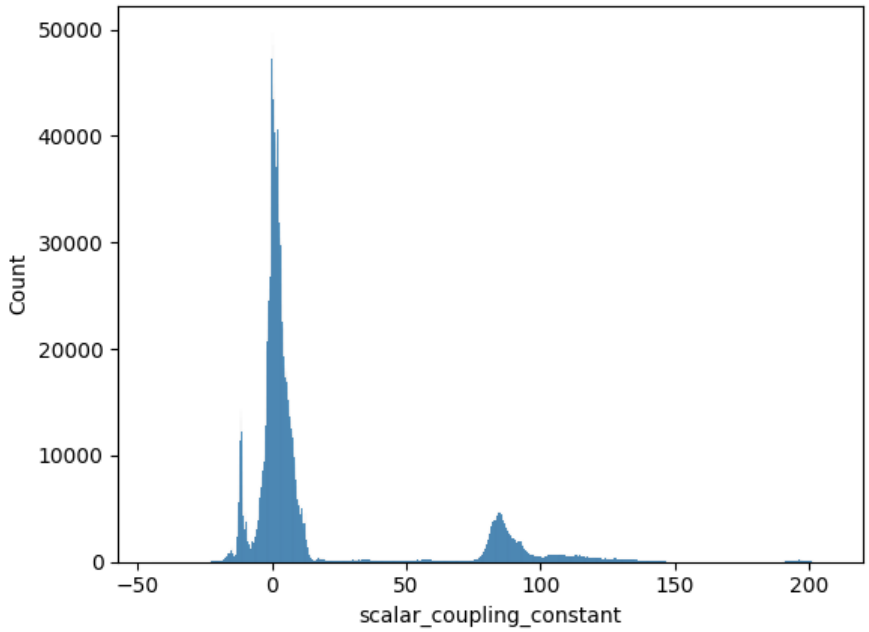

Figura 4. *Conteo de la constante de acoplamiento escalar de los datos de entrenamiento.*

4. Tratamiento de datos.

A partir de toda la información entregada en los múltiples datasets, y las características anteriormente mencionadas debido a que de esta forma se obtiene información relevante sobre cada molécula y las interacciones entre cada par de átomos en la misma, dado que, el objetivo es predecir la constante de acoplamiento que existe entre cada par de átomos presentes en determinada molécula. Se descartó hacer uso de las cargas mulkien y tensores de blindaje magnético, porque las filas entregadas no corresponden con las filas que se tienen en los anteriores dataframes. Además, con las características anteriormente descritas se espera tener suficiente información para llevar a cabo la predicción sin complejizar demasiado dicho propósito.

Una vez se tuvo claro qué valores se iban a usar se procedió a unir toda esta información en un mismo dataframe haciendo uso de la librería pandas.

En primer lugar, se cargan los datos de los parámetros seleccionados, es decir, la energía potencial y el momento dipolar del dataset, esto con el fin de condensar toda la información en un solo dataframe. En primer lugar se cargan las columnas de energía potencial, que como se puede ver en la *Figura 5*, tiene una tendencia a los -400 kJ/mol, pero con variaciones en un rango entre los -600 y -200 kJ/mol.

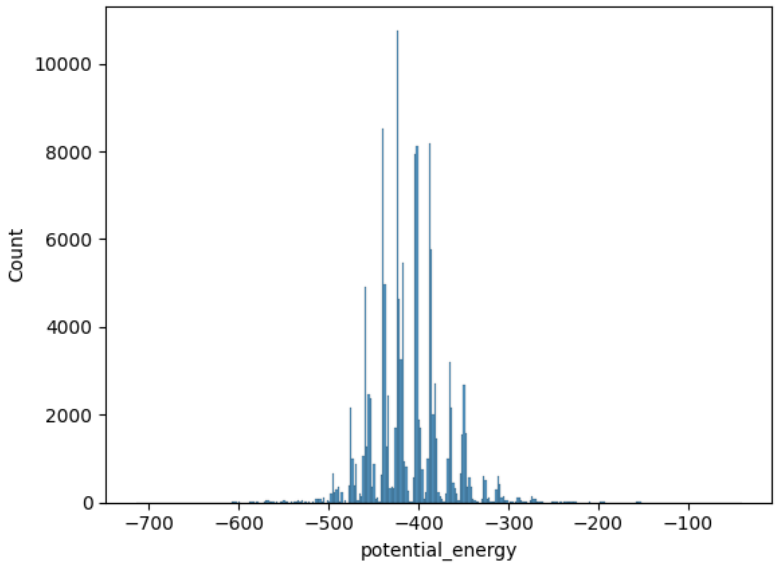

Figura 5. *Distribución de energía potencial en el acoplamiento atómico de los datos reportados.*

El valor negativo es debido a que se reporta la diferencia de energía entre el átomo enlazado y un valor de referencia, que suele tomarse como 0.

Posteriormente, se cargan los datos de momento dipolar, el cual arroja una gran cantidad de valores como 0, lo cual da a entender que el momento dipolar no es tridimensional en un gran número de casos. No obstante, el vector de momento dipolar es una suma vectorial de las componentes mostradas en el dataframe, por tal razón, su magnitud final no va a ser 0 y va a apuntar hacia la zona más negativa de la molécula estudiada.

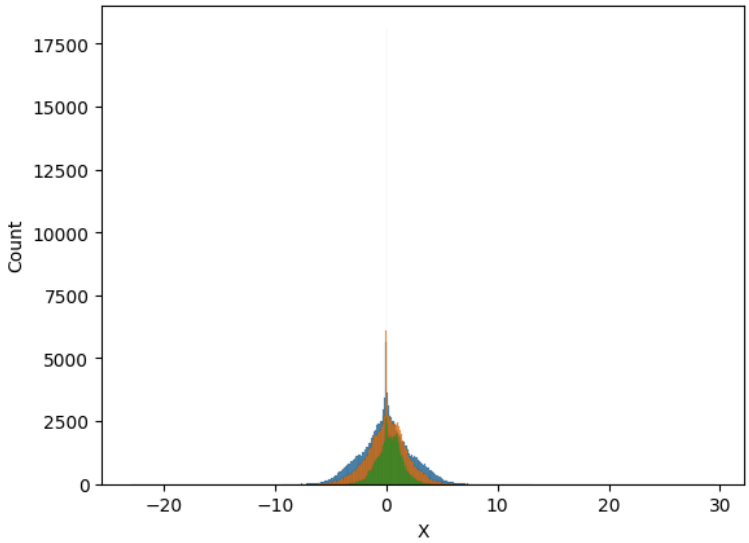

Figura 6. *Distribución de coordenadas de momento dipolar.*

Luego se procedió a hacer las siguientes modificaciones en los datos: debido a que los datos correspondientes al tipo de interacción son presentados con una nomenclatura que describe la interacción entre los tipos de átomos como una variable categórica, se procede a separar dicha información en columnas que denotan el tipo de interacción con un 1 si se trata de dicha interacción o un 0 si no se trata de ella. Se realiza el mismo procedimiento para el índice químico tanto para el index_atom_0 como para el índex_atóm_1. Se procede a convertir el 5% de los datos de las tres columnas correspondientes al momento dipolar en datos faltantes tipo NAN.

Las modificaciones anteriormente expuestas se realizaron con el principal objetivo de cumplir con las condiciones establecidas en el proyecto de contar con al menos 30 columnas y un 5% de datos faltantes en al menos tres columnas.

Posteriormente se procedió a reemplazar los datos faltantes en las tres columnas correspondientes al momento dipolar por el promedio de la columna y a eliminar las dos columnas correspondientes a index_atom_0 e index_atom_1, puesto que, ya están siendo descritas por las nuevas columnas creadas. Para realizar los modelos se decidió extraer del data frame train y el creado por el equipo, con el propósito de sacar unos datos de entrenamiento y otros de testeo, posteriormente a esto se realizó un modelo de machine learning mediante el método de regresión lineal. Lo que se obtuvo fue errores mínimos. Esta prueba se realizó solo utilizando el archivo de train.csv preprocesado, también se evaluaron pruebas de valor absoluto medio, así como error cuadrático medio, Los errores fueron aproximadamente 0 y los score fueron aproximadamente el 100% como se ve en la figura 7.

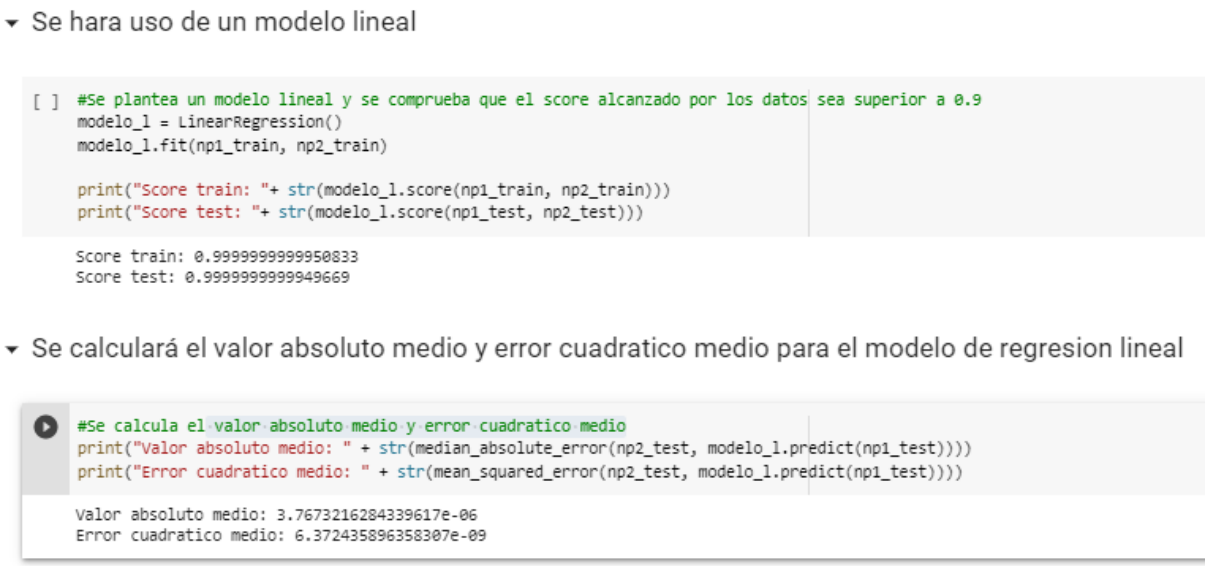

Figura 7. *Modelo de regresión lineal usando machine learning.*

En la limpieza de datos, se seleccionan las primeras 500.023 filas de las más de 4 millones del dataset, puesto que el ejecutar todo el archivo, satura la memoria RAM del entorno Jupyter. El número de filas seleccionado se justifica con los requerimientos solicitados para el presente proyecto (al menos 500.000 datos) y a tomar los datos de una molécula completa, puesto que tomar los 500.000 cortaba información del acoplamiento atómico entre los átomos de una molécula en específico, arrojando información errónea.

5. Métodos supervisados.

Una vez juntados los datos en un solo dataframe, se procede a los modelos iterativos que se clasifican en supervisados.

5.1. Regresión lineal: el primer modelo, y uno de los de mejor comprensión es la regresión lineal, donde se analiza la relación entre las variables de entrada (momento dipolar y energía potencial) y la variable de salida (constante de acoplamiento escalar), la cual, como se ha dicho anteriormente, depende de estas variables de estudio. El modelo se aplica a los datos de *Train* y *Test*, con el fin de evaluar ambos casos. Como resultado, se obtiene un score, el cual se busca que sea lo más cercano al 100 %, y como confirmación del método, se realiza un análisis del error cuadrático medio, el cual, claramente, se busca que sea cercano a 0.

Los resultados obtenidos son de 99,99999999950955 % y 99,99999999949407 % para los data frames de *Train* y *Test* respectivamente, con un error cuadrático medio de $6.464380609479796 \times 10^{-9}$ %, lo cual indica una alta precisión de los datos evaluados. Este alto valor superior al 99% sugiere que el modelo de regresión lineal tiene un ajuste muy bueno a los datos y es altamente predictivo. Sin embargo, es importante tener en cuenta que otros aspectos del modelo, como la interpretación de los coeficientes, la validez de las suposiciones y la presencia de multicolinealidad, también deben tenerse en cuenta al evaluar la calidad y utilidad del modelo de regresión lineal en el contexto específico.

5.2. Árbol de Decisión: cada nodo interno representa una prueba o una condición sobre una característica particular del conjunto de datos, y cada rama que sale del nodo representa el resultado de esa prueba. Los nodos hoja representan las etiquetas o las predicciones finales. El modelo fue aplicado a los datos *Train* y *Test*, se hicieron múltiples pruebas con distintos niveles de profundidad, de 2,4,6,8,10 obteniéndose niveles de score superiores a 0.96 a partir de un nivel de profundidad de 2. Un score superior al 96% en un árbol de decisión indica que el modelo está logrando un alto rendimiento en la tarea de clasificación o regresión para la cual fue

entrenado. Esto implica que el modelo es capaz de hacer predicciones correctas en aproximadamente el 96% de los casos. Sin embargo es importante considerar otros factores y métricas de evaluación para tener una comprensión más completa del rendimiento del modelo.

5.3. Random Forest Regressor: el modelo combina las predicciones de varios árboles para obtener una predicción final más precisa y estable. Cada árbol se entrena con una muestra aleatoria de los datos de entrenamiento y utiliza una selección aleatoria de características para realizar divisiones en los nodos. Esta aleatorización ayuda a reducir el sobreajuste y aumentar la generalización del modelo. Para el presente caso de estudio se realizaron múltiples modelos con diferentes parámetros de número de estimadores y profundidad siendo estos: (2,2),(3,4),(4,8),(5,10),(10,10) a partir del primer par de parámetros se obtuvieron scores superiores al 96% , al igual que en el modelo árbol de decisión Un score del 96% indica que el modelo es capaz de explicar aproximadamente el 96% de la variabilidad en los datos de prueba. En otras palabras, el modelo es capaz de hacer predicciones correctas para el 96% de los casos o muestras en los que se evalúa. Sin embargo es importante tener en cuenta que el puntaje por sí solo no proporciona información completa sobre la calidad del modelo al igual que el modelo planteado anteriormente. Es necesario considerar otros factores, como la distribución de los datos, la cantidad de datos utilizados para el entrenamiento, la validación cruzada u otras métricas de evaluación para tener una imagen más completa de la capacidad predictiva del modelo y su rendimiento en situaciones del mundo real.

6. Métodos no supervisados.

Los métodos no supervisados sirven como un modo de tratamiento de los datos antes del entrenamiento de los métodos supervisados, en este caso se usaron dos tipos de métodos no supervisados:

6.1. PCA: El PCA busca transformar un conjunto de variables correlacionadas en un nuevo conjunto de variables no correlacionadas. Para ello se hizo uso de sus respectivas librerías en este caso *"sklearn.decomposition"* se realizó la exploración de de este algoritmo teniendo en cuenta diferentes valores para los hiperparametros en este caso como en el anterior fueron los valores de 2,4,6,8, y 10 respectivamente, y observar cual de estos daban mejor desempeño, se encontró que quien daba un mejor desempeño es el que tiene un *n_componen* de , 4.

6.2. **K-means:** Es un algoritmo de agrupamiento o clustering ampliamente utilizado en el campo del aprendizaje automático y la minería de datos. Su

objetivo principal es dividir un conjunto de datos en grupos o clusters de manera que las observaciones dentro de cada grupo sean similares entre sí y diferentes de las observaciones en otros grupos.

- Selección del número de clusters (K): Se determina previamente el número de grupos en los que se desea dividir los datos.

- Inicialización de los centroides: Se seleccionan aleatoriamente K puntos del conjunto de datos como centroides iniciales. Los centroides son puntos representativos que actúan como el centro de cada clúster.

- Asignación de observaciones a clusters: Cada observación del conjunto de datos se asigna al cluster cuyo centroide esté más cercano a ella.

- Actualización de los centroides: Los centroides se actualizan calculando el centroide de cada cluster como el promedio de todas las observaciones asignadas a ese cluster.

7. Curvas de aprendizaje.

La curva de entrenamiento muestra cómo se desempeña el modelo en el conjunto de entrenamiento a medida que aumenta la cantidad de datos utilizados. Inicialmente, con un tamaño de conjunto de entrenamiento pequeño, el modelo puede ajustarse demasiado (sobreajuste) a los datos y obtener un rendimiento muy alto en el conjunto de entrenamiento. A medida que se agregan más datos, es posible que el modelo tenga más dificultades para ajustarse a ellos y su rendimiento en el conjunto de entrenamiento disminuya.

7.1. Curva de aprendizaje para la regresión lineal: inicialmente, con un tamaño de conjunto de entrenamiento pequeño, el modelo puede tener dificultades para generalizar y obtener un rendimiento bajo en el conjunto de validación, adicional a ello las curvas de entrenamiento y testeo se mantienen a una distancia casi constante, lo que puede ser indicativo de un alto ajuste a los datos de entrenamiento. Sin embargo ambas curvas presentan valores de score muy bajos por lo que tanto la curva de entrenamiento como la de validación tienen un bajo rendimiento, es posible que haya subajuste, lo que indica que el modelo no es lo suficientemente complejo como para capturar los patrones en los datos.

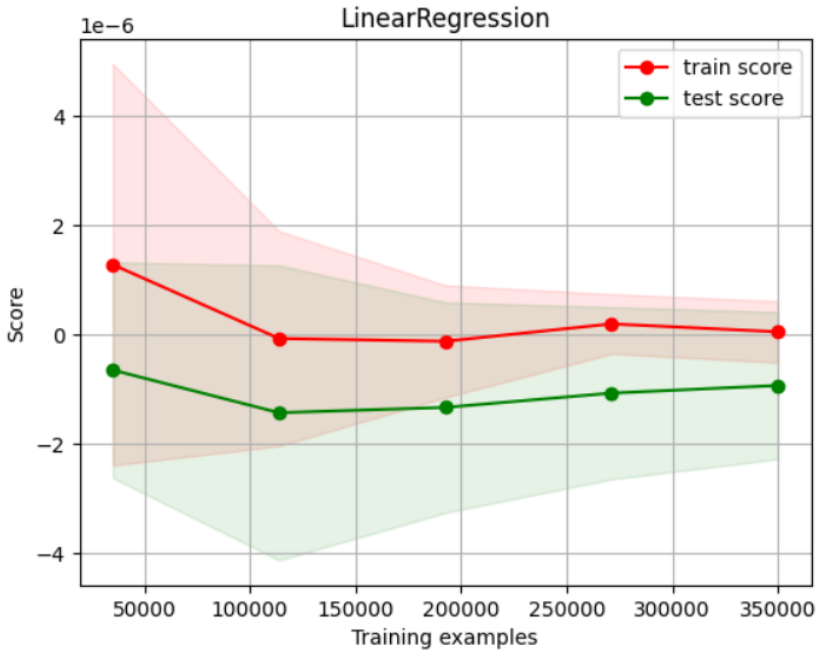

Figura 8. *Curva de aprendizaje para regresión lineal.*

7.2. Curva de aprendizaje para el modelo de árbol de decisión: las curvas comienzan con scores algo similares entre para bajos valores de entrenamiento sin embargo a medida que aumentamos la cantidad de ejemplos de entrenamiento las curvas de test y validación tienden a distanciarse cada vez más, al igual que en la curva de aprendizaje anteriormente expuesta los valores de score obtenidos son bajos por lo que el modelo puede estar presentando un subajuste y sería necesario emplear modelos con parámetros de profundidad más altos para poder capturar de forma más adecuada el comportamiento de los datos.

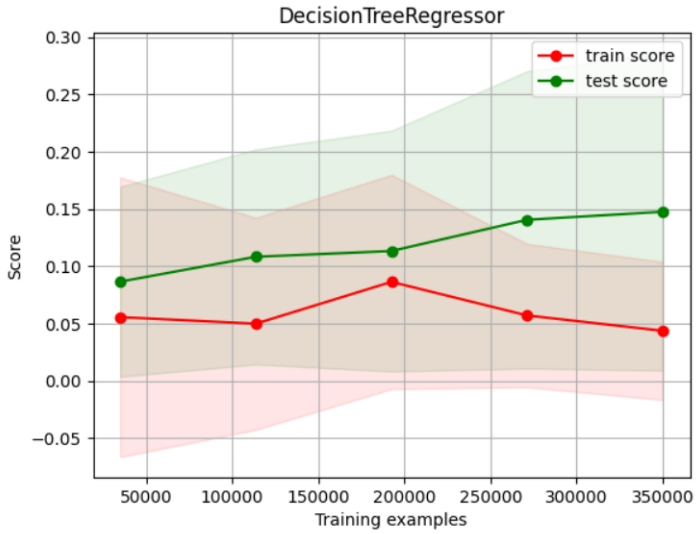

Figura 9. *Curva de aprendizaje de árbol de decisión.*

7.3. **Curva de aprendizaje para el modelo de Random Forest Regressor:** la curva de testeo presenta una pendiente positiva para valores bajos de ejemplos de entrenamiento, lo cual indica que el modelo aprende rápidamente a medida que entregamos mayor cantidad de datos, luego las curvas se estabilizan y superponen , por lo que tienen scores muy similares, lo cual implica que puesto que las curvas de entrenamiento y validación se estabilizan a medida que se agregan más datos.esto indica que el modelo es robusto y no es muy sensible a variaciones en los datos de entrenamiento. Sin embargo el bajo score alcanzado por ambas curvas indica que el modelo puede mejorar su desempeño aumentando los hiperparametros empleados o en otras palabras sería necesario complejizar un poco más el modelo para captar adecuadamente el comportamiento de los datos y así tener un posible score más alto tanto en entrenamiento como en validación.

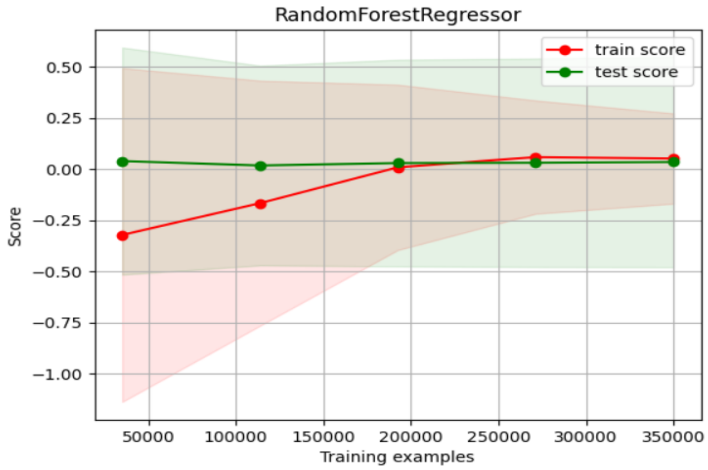

Figura 10. *Curva de aprendizaje de Random Forest Regressor.*

8. Retos y consideraciones de despliegue.

Inicialmente se consideró realizar el proyecto haciendo uso de todas las filas entregadas en el dataset el cual contaba con más de 4 millones de las mismas, sin embargo trabajar con dicha cantidad de datos se tornó en algo sumamente complicado debido a limitaciones de RAM en el entorno (Google Colab); constantemente se reiniciaba el proyecto y los cambios no eran guardados, además de que los tiempos de ejecución de ciertas tareas de limpieza y organización de los datos se tornaba en una tarea de horas complejizando mucho el proyecto.

Debido a las complicaciones anteriormente expuestas se decidió usar una muestra de todos los datos entregados, se procuró entonces tomar alrededor de 500 mil filas teniendo en cuenta que la última molécula en el dataset se encontrara completa y con la información de todos sus átomos, por lo que, finalmente se

usaron exactamente 500023 filas del data frames y con las mismas se continuará el proyecto de predicción.

Consideramos que la cantidad de datos usados para el procesamiento de la información es adecuada, puesto que, se está cumpliendo con el requisito de tener al menos 5 mil filas además de que se está respetando el que todas las moléculas empleadas en el dataset tengan la información sobre sus átomos completa.

Por otra parte, el proceso de limpieza de la información requiere saber adecuadamente qué información se debe conservar y cuál no, con el fin de optimizar los datos conservados. Una vez hecho este proceso se pudo continuar con los métodos iterativos pertinentes, mostrando los resultados previamente discutidos.

9. Conclusiones.

- La correcta limpieza y entendimiento de los dataset a estudiar son los pasos fundamentales para iniciar cualquier investigación con un conjunto de datos puesto que todos los resultados que se obtengan dependen de la calidad de los mismos y orden de los mismos.

- El uso de modelos supervisados necesita de un correcto entendimiento de los datos a estudiar puesto que los modelos predictivos dependen de los hiperparametros ingresados por el investigador lo cual puede incurrir en subajustes o sobre ajustes, Por eso lo que se busca es un hiper parámetro que le permita al modelo ajustarse y captar el comportamiento de los datos sin llegar a "aprender de memoria los datos" puesto que en este caso el modelo no lograria predecir nuevos datos que se le ingresen.

- Los métodos no supervisados tienen importancia, puesto que, permiten una implementación en el preprocesamiento de los datos con los que se trabaja, permitiendo así presentar un dataset más compacto y almacenando las variables que tienen mayor peso en mi entrenamiento. Por otro lado, permiten explorar y analizar datos sin la necesidad de etiquetas o categorías previas. Estos métodos son especialmente útiles cuando no se dispone de información de salida o cuando se desea descubrir patrones y estructuras ocultas en los datos.

- Es necesario realizar un análisis exhaustivo y detallado de todas las variables que tienen mayor influencia en los modelos que se entrenan, puesto que, de esta forma se puede llegar a reducir el error significativamente.

Referencias bibliográficas.

[1] Castañar Acedo, L. (2012). *Medida de Acoplamientos Dipolares Residuales en Moléculas Orgánicas*. Universidad Autónoma de Barcelona. Accedido el 28 de mayo de 2023. [En línea].
Disponible: https://sermn.uab.cat/wp-content/uploads/2012/10/Master-definitivo.pdf

[2] SCAI. (2022). *Resonancia Magnética Nuclear*. [Entrada de Blog]. Accedido el 28 de mayo de 2023.
Disponible: https://www.scai.uma.es/areas/aqcm/rmn/rmn.html

[3] Libretexts. (*S.F.*) *9.4: Energía y Formación de Enlace Covalente*. LibreTexts - Español. [Libro en línea]. Accedido el 28 de mayo de 2023.
Disponible: https://espanol.libretexts.org/Quimica/Química_Introductoria_Conceptual_y_GOB/Química_Introductoria_(CK-12)/09:_Enlace_covalente/9.04:_Energía_y_Formación_de_Enlace_Covalente