UNIVERSIDAD SAN FRANCISCO DE QUITO

Colegio de Ciencias e Ingenierías

NeuralRho: un modelo de red neuronal para estimar la densidad electrónica

Alisson Dayana Cabrera Felicita

Ingeniería Química

TRABAJO DE FIN DE CARRERA PRESENTADO COMO REQUISITO
PARA LA OBTENCIÓN DEL TÍTULO DE INGENIERA QUÍMICA

Quito, 9 de diciembre de 2024

UNIVERSIDAD SAN FRANCISCO DE QUITO

Colegio de Ciencias e Ingenierías

HOJA DE CALIFICACIÓN
DE TRABAJO DE FIN DE CARRERA

NeuralRho: un modelo de red neuronal para estimar la densidad electrónica

Alisson Dayana Cabrera Felicita

Nombre del profesor, Título académico **F. Javier Torres, Ph. D**

Quito, 9 de diciembre de 2024

© DERECHOS DE AUTOR

A través de este documento, certifico que he revisado todas las Políticas y Manuales de la Universidad San Francisco de Quito (USFQ), incluyendo la Política de Propiedad Intelectual de la USFQ, y estoy de acuerdo con su contenido, por lo que los derechos de propiedad intelectual de este trabajo estarán sujetos a lo establecido en dichas políticas.

De igual forma, autorizo a la USFQ a proceder con la digitalización y publicación de este trabajo en el repositorio virtual, conforme a lo estipulado por la Ley Orgánica de Educación Superior del Ecuador.

Nombres y apellidos: Alisson Dayana Cabrera Felicita

Código: 00212956

Cédula de identidad: 1727316596

Lugar y fecha: Quito, 9 de diciembre de 2024

ACLARACIÓN PARA LA PUBLICACIÓN

Nota: Este trabajo, en su conjunto o en cualquiera de sus secciones, no debe ser tratado como una publicación, aunque esté accesible sin restricciones a través de un repositorio institucional. Esta afirmación sigue las directrices y sugerencias del Committee on Publication Ethics (COPE), según lo expuesto por Barbour et al. (2017) en el documento de discusión sobre las mejores prácticas en la publicación de tesis, disponible en http://bit.ly/COPETheses.

UNPUBLISHED DOCUMENT

Note: The following capstone project is available through Universidad San Francisco de Quito USFQ institutional repository. Nonetheless, this project– in whole or in part– should not be considered a publication. This statement follows the recommendations presented by the Committee on Publication Ethics COPE described by Barbour et al. (2017) Discussion document on best practice for issues around these publishing available on http://bit.ly/COPETheses.

RESUMEN

La densidad electrónica es una propiedad molecular observable que permite definir precisamente la energía de un sistema y conocer el comportamiento electrostático de una molécula. El cálculo de la densidad electrónica de moléculas complejas como péptidos y proteínas tiene un alto costo computacional. Por lo tanto, el presente proyecto se propone diseñar una herramienta en base a métodos de *machine-learning* que permita la obtención de la densidad electrónica sin recurrir al cálculo de la función de onda. En primer lugar, se construye la red neuronal mediante la biblioteca de código abierto TensorFlow. La red neuronal requiere de un valor de partida, conocida como densidad promolecular, calculada por el método de Roothaan-Hartree-Fock (RHF), de acuerdo con en el trabajo previo de Clementi E. y Roetti C., que en el presente caso se obtiene a partir de la función de onda electrónica y el programa. Asimismo, se requiere de un valor de referencia, la densidad molecular obtenida por Multiwfn. La red neuronal se encarga de aproximar la densidad promolecular a la de referencia, para moléculas constituidas por C, H, O y N en su posición de equilibrio. En segundo lugar, se evalúa el rendimiento de la red neuronal mediante el cálculo del error cuadrático medio (MSE) de moléculas fuera de la etapa de entrenamiento. Finalmente, se aplica la red neuronal en el cálculo de la densidad electrónica de aminoácidos como paso preliminar para la estimación de la densidad molecular de proteínas y péptidos más complejos. Se obtiene que la mejor configuración de la red neuronal consta de tres capas de 100, 50 y 25 neuronas respectivamente con una función de activación tangente hiberbólica. La etapa de entrenamiento obtiene un MSE de 1.7372e-6, mientras que en la etapa de validación este mismo valor es de 1.0990e-6. Asimismo, el error de divergencia de Kullback–Leibler establece que existe sobreestimación de la red neuronal en zonas asociadas a electrones libres de átomos como N y O, en grupos hidroxilos y aminas secundarias, así como en grupos CN, pero subestima en las cercanías de los núcleos y enlaces covalentes. En conclusión, se establece que la red neuronal presenta un buen rendimiento y es transferible, pero requiere de trabajo adicional para incluir una mayor variedad de sistemas moleculares.

***Palabras clave**: Densidad molecular, Machine-Learning, Aminoácidos, Redes Neuronales*

ABSTRACT

Electron density is an observable molecular property that allows to precisely define the energy of a system and to know the electrostatic behavior of a molecule. Calculating the electron density of complex molecules such as peptides and proteins has a high computational cost. Therefore, the present project aims to design a tool based on *machine-learning* methods that allows obtaining the electron density without resorting to the calculation of the wave function. First, the neural network is built using the open-source library TensorFlow. The neural network requires a starting value, known as the promolecular density, calculated by the Roothaan-Hartree-Fock (RHF) method, according to the previous work of Clementi E. and Roetti C., which in the present case is obtained from the electron wave function and the program. Likewise, a reference value is required, the molecular density obtained by Multiwfn. The neural network is responsible for approximating the promolecular density to the reference density for molecules consisting of C, H, O and N in their equilibrium position. Secondly, the performance of the neural network is evaluated by calculating the mean square error (MSE) of molecules outside the training stage. Finally, the neural network is applied to calculate the electron density of amino acids as a preliminary step for estimating the molecular density of more complex proteins and peptides. It is obtained that the best configuration of the neural network consists of three layers of 100, 50 and 25 neurons respectively with a hyperbolic tangent activation function. The training stage obtains an MSE of 1.7372e-6, while in the validation stage this same value is 1.0990e-6. Likewise, the Kullback–Leibler divergence error establishes that there is an overestimation of the neural network in areas associated with free electrons of atoms such as N and O, in hydroxyl groups and secondary amines, as well as in CN groups, but an underestimation in the vicinity of nuclei and covalent bonds. In conclusion, it is established that the neural network presents a good performance and is transferable but requires additional work to include a greater variety of molecular systems.

Keywords: *Molecular density, Machine-Learning, Amino acids, Neural Networks*

TABLA DE CONTENIDO

TABLA DE ILUSTRACIONES

1. INTRODUCCIÓN

1.1 Antecedentes

La densidad electrónica es una propiedad fundamental de átomos, moléculas y fases condensadas de la materia, ya que permite estimar otras propiedades del estado fundamental, como el momento electrostático, potencial electrostático, y energías de interacción electrostática de átomos y moléculas (Grisafi, et al., 2019). A partir de las anteriores, la densidad electrónica permite definir precisamente la energía del sistema, y se puede asociar a la distribución de los electrones en el espacio.

La densidad electrónica se obtiene al resolver la ecuación de Schrödinger mediante el cálculo previo de la función de onda. A este respecto se conoce la solución exacta solamente del átomo de hidrogeno. Por lo que la solución de moléculas más complejas se logra mediante métodos computacionales como *ab initio*. Algunos de los métodos ab initio más comunes incluyen el Método de Hartree-Fock (HF) y el Método de Configuración Interactiva (CI). Asimismo, existen múltiples programas de software que implementan métodos ab initio para realizar cálculos de mecánica cuántica, como Gaussian, VASP (Vienna Ab initio Simulation Package), Quantum ESPRESSO, CASTEP y ORCA (González Forero, 2013).

En el caso de moléculas de gran tamaño como péptidos y proteínas, se requiere de un gran esfuerzo computacional, para lo cual se utilizan técnicas de escalamiento linear como el método de ensamblaje lego de densidad electrónica molecular (MEDLA), el método de ensamblaje de matriz de densidad ajustable (ADMA) o el método del orbital molecular extremadamente localizado (EMLO).

Finalmente, dentro del campo de la inteligencia artificial, métodos de *machine learning* se presentan también como una alternativa. Siguiendo esta premisa, el presente trabajo de titulación tiene como objetivo diseñar una herramienta en base a métodos de *machine-learning* que permita la obtención de la densidad

electrónica $\rho_{(r)}$ sin recurrir al cálculo de la función de onda $\psi(r)$ a partir de la geometría de la molécula en estado de equilibrio.

Existen trabajos previos que se han propuesto con el mismo objetivo de encontrar la densidad electrónica de diferentes compuestos mediante el uso de herramientas de machine learning, los cuales se resumen a continuación.

- Grisafi et al. desarrollaraon un modelo de machine-learning, centrado y adaptado a la simetría del átomo, para obtener la densidad electrónica de la capa de valencia de hidrocarburos complejos como octano y tetraoctano a partir del entrenamiento con butano y butadieno (2019).
- Fabrizio et al. desarrollaron un modelo de machine learning para obtener la densidad electrónica de sistemas con interacciones no covalentes, caracterizadas por bajas densidades y gradientes de densidades, a partir del entrenamiento con dímeros de cadena lateral (2019).
- Brockherde et al. desarrollaron un modelo de machine learning para estudiar la densidad potencial y la densidad de energía en un malonaldehído y capturar el proceso de transferencia intramolecular de protones (2017).
- Sinitskiy & Pande desarrollaron un modelo de machine learning para predecir la densidad electrónica y energía de moléculas orgánicas en la database QM9, a partir del entrenamiento con soluciones de DFT con un conjunto de base (PBE0/pcS-3) (2018).
- Cuevas-Zuviría & Pacios desarrollaron un modelo de machine learning que predice los parámetros utilizados en un modelo analítico anisotrópico de densidad electrónica, a partir del entrenamiento con parámetros obtenidos mediante cálculos con ab initio (2020).

1.2 Justificación del proyecto

La densidad electrónica juega un papel fundamental en el enfoque descrito en el libro *"Atoms in Molecules: A Quantum Theory"* de Richard Bader, publicado en 1990. Este trabajo es clave en el campo de la química computacional y la teoría cuántica aplicada a la química de los átomos y moléculas. De esta forma, la densidad electrónica en el libro de Bader proporciona una base para entender los enlaces químicos y las interacciones atómicas de manera más natural y visual. A través de la topología de la densidad electrónica, se demuestra cómo las estructuras electrónicas pueden ser divididas en cuencas atómicas bien definidas, lo cual facilita el estudio de la reactividad química, la interpretación de la polaridad y la naturaleza de los enlaces y la descripción precisa de interacciones no covalentes (Bader, 1990).

Por otro lado, el conocimiento de la densidad electrónica permite la visualización en tiempo real de descriptores moleculares, que son la representación de la estructura molecular junto con descripciones de propiedades fisicoquímicas y actividades biológicas, cálculo de intensidades en el espectro infrarrojo y el conocimiento exacto del comportamiento electrostático en simulaciones moleculares (Grisafi, et al., 2019).

El presente proyecto utiliza métodos de *machine learning* para la estimación de la densidad electrónica, mediante el diseño de una red neuronal que sea transferible. Con este objetivo en mente, el modelo se entrena con una amplia variedad de moléculas orgánicas, en vez de grupos de moléculas específicos como dímeros, bencenos, malonaldehídos, entre otros.

1.3 Objetivos

1.3.1 Objetivo general

Diseñar una herramienta en base a métodos de *machine-learning* que permita la obtención de la densidad electrónica

1.3.2 Objetivos específicos

- Construir y optimizar la red neuronal para generar una densidad molecular aproximada.
- Evaluar la validez de la densidad molecular aproximada.
- Aplicar la red neuronal para estimar la densidad molecular de aminoácidos.

2. MARCO TEÓRICO

2.1 Densidad electrónica

La química cuántica es una rama de la mecánica cuántica que se encarga de estudiar sistemas de interés químico, la cual a la vez se fundamenta en varios postulados.

La función de onda es un elemento matemático que describe totalmente un sistema mecano-cuántico y depende de las coordenadas espaciales, del tiempo y de ser necesario de las coordenadas del spin.

$$\Psi = \Psi(x, y, z, t) \qquad (1)$$

$$\Psi(x, y, z, t) = \psi(x, y, z) f(t) \qquad (2)$$

Esta función debe ser bien comportada, para lo cual debe cumplir con los requisitos de ser finita en todos sus valores, continua y univaluada (Forero, 2013).

La función de onda tiene la importante propiedad de que el producto punto entre la función de onda y su conjugado $\psi^*(x)\psi(x)dx$ es la probabilidad de que la partícula se encuentre en el intervalo dx, localizado en la posición x (McQuarrie & Simon, 1997).

Para un sistema con N electrones, la función de onda se define de la siguiente forma.

$$\psi = \psi(\boldsymbol{r_1}, \dots, \boldsymbol{r_N}, \tau_1, \dots, \tau_N) \qquad (6)$$

Igualmente, la densidad electrónica se define de la siguiente forma.

$$\rho(\boldsymbol{r}) = N \sum_{\tau_1} \dots \sum_{\tau_1} \int d\boldsymbol{r_2} \dots \int d\boldsymbol{r_N} |\psi(\boldsymbol{r_1}, \dots, \boldsymbol{r_N}, \tau_1, \dots, \tau_N)|^2 \qquad (7)$$

En donde N es el número de electrones, $\boldsymbol{r_1}, \dots, \boldsymbol{r_N}$ son las coordenadas espaciales, $\tau_1, \dots, \tau_N$ son las coordenadas spin.

Es evidente que la densidad electrónica depende únicamente del vector $\boldsymbol{r}$, al contrario de la función de onda, que depende de 4N variables como posición y espín por cada electrón. Por lo tanto, el uso de la densidad electrónica resulta adecuado desde varios puntos de vista, incluyendo la apreciación intuitiva, la visualización gráfica y expansiones del conjunto base (Sinitskiy & Pande, 2018).

Con respecto a su descripción física, la densidad electrónica demuestra como los electrones están deslocalizados en la molécula, es decir, establece donde la probabilidad de encontrar un electrón en el espacio es mayor y donde es menor. De acuerdo con la teoría de Hohenberg-Kohn, la energía del estado basal *E* de un sistema molecular puede ser calculada mediante la minimización del funcional de densidad $E[\rho(\boldsymbol{r})]$. Sin embargo, la expresión del funcional de densidad no se conoce, pero se puede aproximar con diferentes aproximaciones con diferente grado de exactitud y costo computacional (Sinitskiy & Pande, 2018).

2.2 Método de Roothan-Hartree-Fock

El método de Roothaan-Hartree-Fock (RHF) proporciona una formulación matricial de las ecuaciones de Hartree-Fock, que se utilizan para describir las funciones de onda de los electrones en átomos y moléculas. La función de onda en el método de Hartree-Fock se representa como una combinación lineal de orbitales atómicos. En 1974, el método RHF fue la técnica más precisa disponible para aproximar las ecuaciones HF, gracias a Clementi y Roetti, quienes realizaron tablas completas de funciones de onda RHF para el estado basal y ciertos estados excitados de átomos neutros e ionizados (Bunge et al., 1993).

La densidad electrónica de una molécula $\rho(\boldsymbol{r})$ se descompone en contribuciones centradas en el átomo $\rho_i(\boldsymbol{r})$, que son funciones definidas para átomos aislados $\rho^0(\boldsymbol{r})$, denominada como densidad promolecular, de la siguiente manera:

$$\rho(\boldsymbol{r}) = \rho^0(r) + \delta\rho(\boldsymbol{r}) = \sum_{i=1}^{N} \rho_i(\boldsymbol{r}) = \sum_{i=1}^{N} (\rho_i{}^0(\boldsymbol{r}) + \delta\,\rho_i(\boldsymbol{r})) \qquad (8)$$

Este modelo consiste en una suma de funciones de tipo Slater:

$$\rho^0(\boldsymbol{r}) = \sum_k C_{ki}\, STO_k = \sum_k C_{ki}\, N_k r^{n_k-1} \exp(-\xi_k r) \qquad (9)$$

donde la constante de normalización es,

$$N_k = \frac{(2\xi_k)^{n_k+1/2}}{\sqrt{(2n_k)!}} \qquad (10)$$

En general, la densidad promolecular es muy cercana a la densidad de entrenamiento. Aquí, $\rho_i(\boldsymbol{r})$ se calcula utilizando la partición del espacio real de Hirshfeld.

$$\rho_i(\boldsymbol{r}) = \frac{\rho_i(\boldsymbol{r})}{\rho^0(\boldsymbol{r})}\rho(\boldsymbol{r}) \qquad (11)$$

Por lo tanto, $\delta\rho_i(\boldsymbol{r})$ se calcula utilizando una red neuronal.

El presente proyecto utiliza las funciones de onda de RHF para estados basales reportados en tablas para átomos desde He hasta Xe con energías de error que no sobrepasan 0.6meV. Las tablas contienen los valores de número atómico, orbitales tipo Slater (STO), exponentes y constantes requeridas.

La entrada de la red neuronal, conocida como *NeuralRho*, utiliza los siguientes descriptores:

- Z_i: número atómico del átomo i
- $\rho_i{}^0(\boldsymbol{r})$: densidad promolecular del átomo i en $\boldsymbol{r}$

Finalmente, la densidad molecular aproximada se obtiene de la siguiente manera:

$$\sum_{i=1}^{N} \delta\rho_i(r) = \Omega(r) \qquad (13)$$

$$\rho^{Approx}(r) = \rho^0(r) + \Omega(r) \qquad (14)$$

2.3 Redes neuronales

Las redes neuronales son una parte fundamental del campo de la inteligencia artificial y el aprendizaje automático (*machine learning*), y se utilizan para resolver una amplia variedad de problemas como el reconocimiento de patrones, clasificación, regresión, predicción, entre otros. Una red neuronal está compuesta por nodos o neuronas organizados en capas, específicamente en capas de entrada, ocultas y de salida, como se indica en la Figura 1. Cada neurona de una red neuronal realiza una operación matemática básica, que es una combinación lineal de las entradas multiplicadas por los pesos asociados, seguida de una función de activación. Todos estos son parámetros que afectan el desempeño de una red neuronal.

De esta forma, un elemento importante son las funciones de activación, que son cruciales en las redes neuronales para introducir no linealidad en el modelo. Algunas de las funciones de activación más comunes son ReLU (Rectified Linear Unit) $(f(x) = max(0, x)$, y Tangente hiperbólica (tanh) $\left(f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}\right)$, que produce salidas entre -1 y 1.

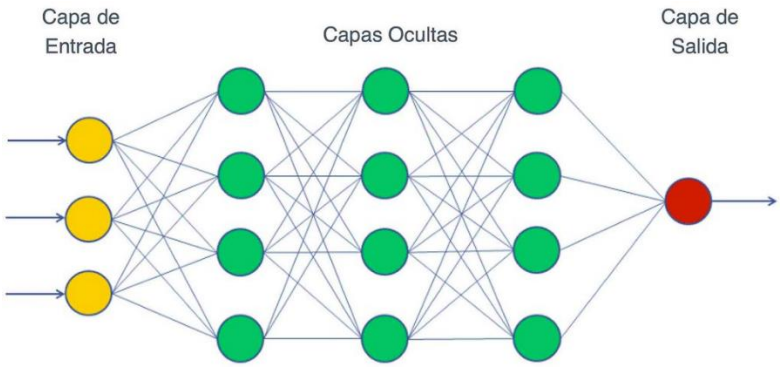

Figura 1. Esquema de una red neuronal

2.4 Divergencia de Kullback-Leibler

La divergencia de *Kullback-Leibler* es una medida de la diferencia entre dos distribuciones de probabilidad. Aunque no es una distancia en sentido estricto, ya que no es simétrica y no satisface la desigualdad triangular, es una herramienta fundamental en diversas áreas como el aprendizaje automático, la teoría de la información y la estadística. Se define de la siguiente forma:

$$D_{KL}(P||Q) = \sum_{x \in X} P(x)\, log\left(\frac{P(x)}{Q(x)}\right) \qquad (3)$$

En donde P y Q son dos distribuciones de probabilidad sobre un espacio de probabilidad *X*.

La divergencia de *Kullback-Leibler* mide la cantidad de información adicional que se pierde cuando se usa la distribución Q, en lugar de P, para aproximar los eventos de la distribución P (Cover & Thomas, 2005).

3. METODOLOGÍA

Obtención de la geometría de la molécula en su posición de equilibrio

Se utiliza una base de datos de 150 moléculas orgánicas constituidas por C, H, O Y N (Ramakrishnan, 2014) calculadas con el nivel de teoría B3LYP/6-31G(2df, p) y cuya geometría en coordenadas cartesianas se encuentra en un archivo.xyz. El conjunto de moléculas se definió aleatoriamente y tienen la siguiente distribución: 45 moléculas contienen C y H, 45 contienen C, H y O, 25 contienen C, H y N, 31 contienen C, H, O y N, 2 contienen C y N, una molécula contiene N y H y una molécula contiene O y H.

Cálculo de la densidad molecular mediante el programa Multiwfn

Se utiliza un archivo.wfn de entrada que contiene las coordenadas atómicas, energías orbitales, números de ocupación, entre otros. El archivo.cube de salida contiene coordenadas atómicas y un conjunto de datos que corresponden a la densidad electrónica de cada punto en el espacio, en unidades atómicas.

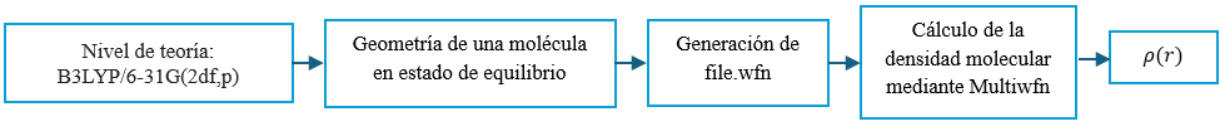

Figura 2. Esquema del cálculo de la densidad molecular

Cálculo de la densidad promolecular mediante el método de RHF

Se utiliza un archivo.dat que contiene el número atómico, momento angular máximo, número de orbitales atómicos, número de orbitales tipo Slater, números ocupaciones, datos de constantes, obtenidos a partir de las funciones de onda de Roothaan-Hartree-Fock en el estado basal, y los cuales se basan en el trabajo previo de Clementi y Roetti (Bunge et al., 1993), el cual se requiere en el programa desarrollado en Pyhton3 que realiza el cálculo de la densidad promolecular.

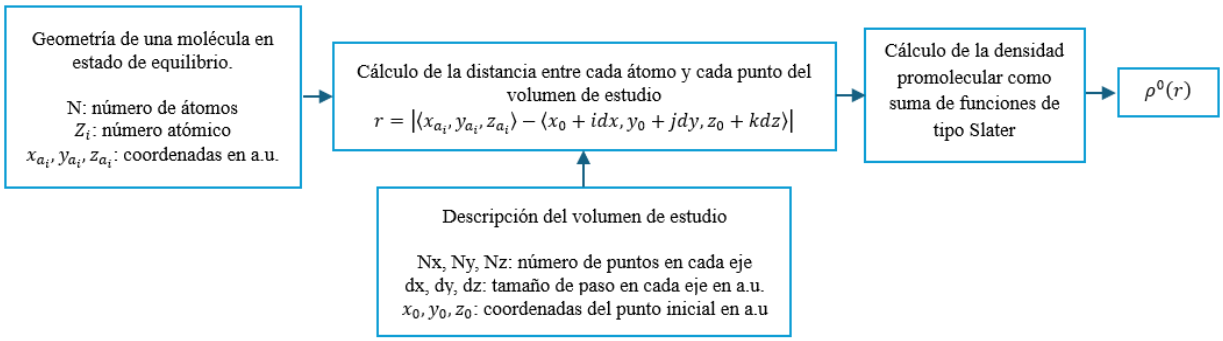

Figura 3. Esquema del cálculo de la densidad promolecular

Construcción y optimización de la red neuronal

La red neuronal se construye mediante la biblioteca de código abierto *TensorFlow*, y la optimización se realiza mediante el cálculo de error generado por diferentes configuraciones de la red neuronal, primero en la etapa de entrenamiento y después en la etapa de validación.

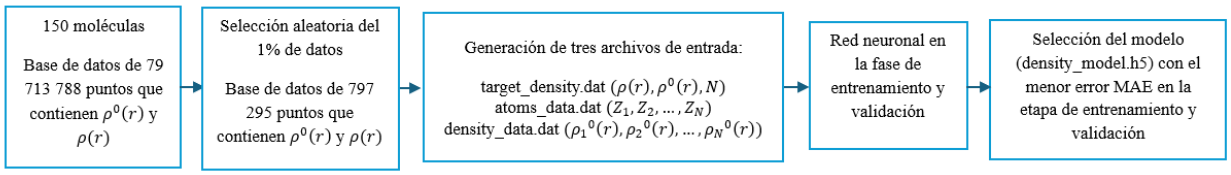

Figura 4. Esquema de selección del modelo de red neural óptimo

La configuración de la red neuronal de se basa en los siguientes parámetros.

- Función de activación
- Numero de capas internas
- Numero de neuronas por capa

Construcción de la herramienta computacional para la estimación de la densidad molecular

Se requiere de un programa que, a partir de la geometría en estado de equilibrio de cualquier molécula constituida por C, H, O y N, sin la necesidad de conocer su densidad molecular de referencia calculada por Multiwfn se pueda estimar su densidad molecular. El funcionamiento del programa o herramienta computacional que incluye la red neuronal se describe a continuación.

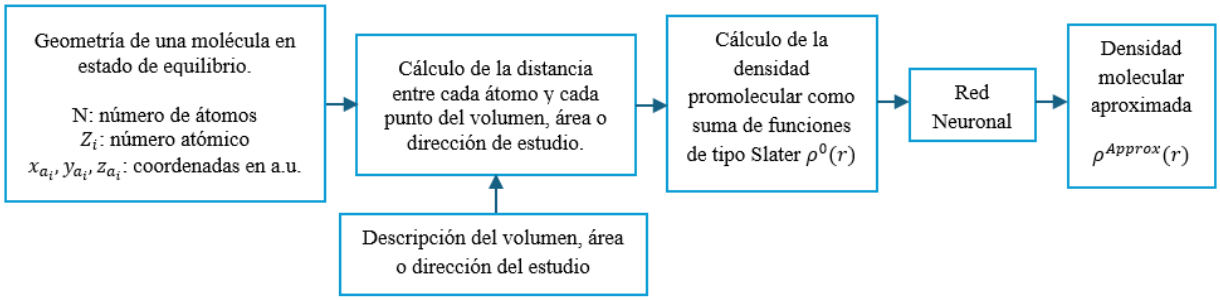

Figura 5. Esquema del funcionamiento de la herramienta computacional para la estimación de la densidad molecular

Evaluación del rendimiento de la red neuronal

Se escogen 20 moléculas fuera del conjunto de entrenamiento para determinar su densidad molecular mediante la red neuronal y compararla con la densidad de referencia por Multiwfn. La diferencia entre ambas densidades moleculares se evalúa mediante la divergencia de de *Kullback-Leibler* punto por punto.

4. RESULTADOS Y DISCUSIÓN

Se determina que la red neuronal óptima tiene un error calculado de 5.7759e-7 en la etapa de entrenamiento, 4.1226e-7 en la fase validación, y con el menor tiempo en cálculo computacional, y corresponde a la siguiente configuración:

- Función de activación: Tangente hiperbólica
- Numero de capas: 3
- Numero de neuronas: 50,25,12

Las tablas que contienen los errores obtenidos de las diferentes configuraciones de la red neuronal se encuentran en Anexos.

Una vez que se selecciona el modelo de la red neuronal más optimo, se procede evaluar su rendimiento en moléculas fuera del entrenamiento, primeramente, mediante el cálculo de MSE.

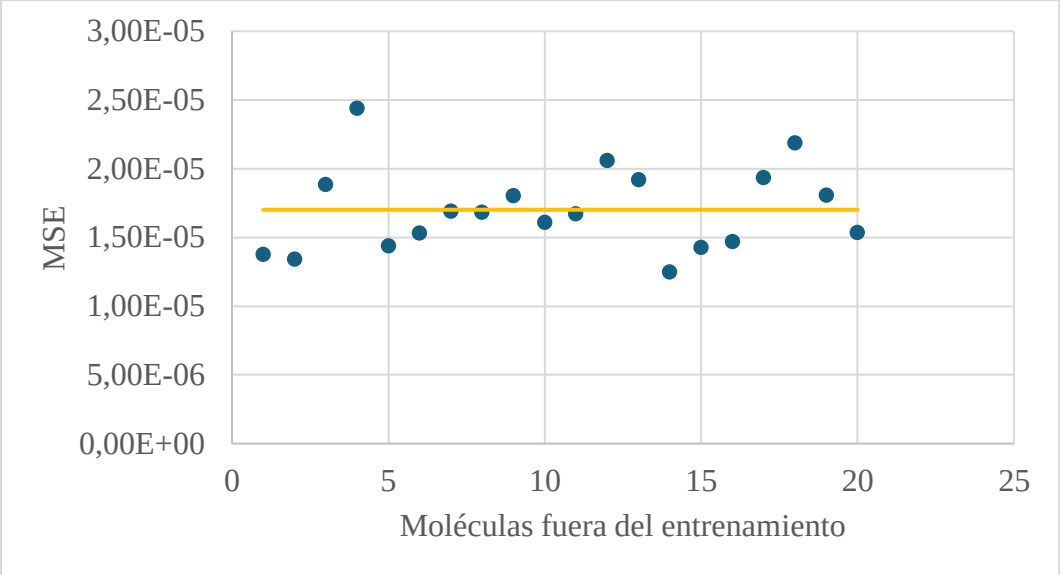

Figura 6. Error generado por la red neuronal en el Entrenamiento 1 en moléculas fuera del entrenamiento

La molécula con el mayor error igual a 2.4389e-5 corresponde a la molécula 1,3-propanodiol, la cual contiene dos grupos hidroxilo en los extremos, lo cual se debe principalmente a que la red neuronal no fue entrenada en mayor parte con moléculas constituidas por H y O, de la cual solo se encuentra una. Por otro lado, el menor error es igual a 1.2470e-5.

Además, se conoce que la aproximación de la red neuronal en los espacios más alejados de la molécula es muy alta, con una diferencia en magnitud de 10^{10}, lo cual crea un sesgo sobre los valores de densidad más cercanos al núcleo. De esta forma, se establece un límite inferior de densidad promolecular para un segundo entrenamiento de datos, en donde valores menores a 1e-5 bohr^{-3} se mantienen como densidad promolecular y valores iguales o mayores se aproximan a la densidad molecular. Esto se debe a que la densidad promolecular se considera que describe suficientemente bien las regiones de baja densidad y bajo gradiente, típicas de las regiones no covalentes (Fabrizio, Grisafi, Meyer, Ceriotti, & Corminboeuf, 2019). Asimismo, las regiones no covalentes o con interacciones de dispersión más débiles se establecen en valores de densidad electrónica menor a 5e-3 bohr^{-3} (Johnson, Keinan, Sánchez, & Contreras-García, 2010). Por lo tanto, se puede decir que la densidad promolecular se puede utilizar sin problema sobre esta región.

En el segundo entrenamiento se establece un conjunto de moléculas iniciales más grande que cumplen con la condición antes descrita, y se procede con la misma metodología que en el entrenamiento 1.

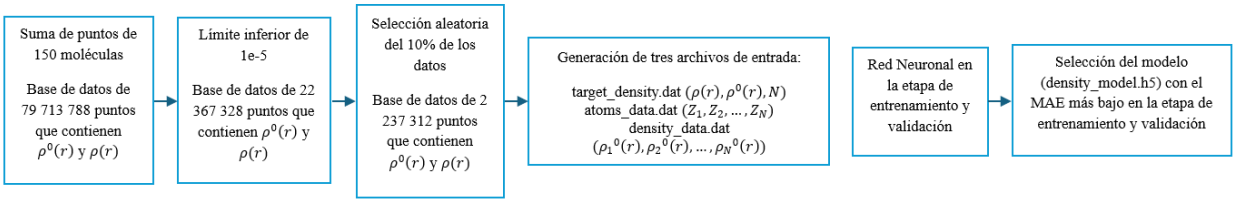

Figura 7. Esquema de selección del modelo de red neural óptimo en Entrenamiento 2

Se determina que la red neuronal óptima tiene un error calculado de 1.7372e-6 en la etapa de entrenamiento, 1.0990e-6 en la fase validación, y con el menor tiempo en cálculo computacional, corresponde a la siguiente configuración:

- Función de activación: Tangente hiperbólica Tanh(x)
- Numero de capas: 3
- Numero de neuronas: 100,50,25

Las tablas que contienen los errores obtenidos de las diferentes configuraciones de la red neuronal se encuentran en Anexos.

De igual forma, una vez que se selecciona el modelo de la red neuronal más optimo, se procede evaluar su rendimiento en moléculas fuera del entrenamiento, primeramente, mediante el cálculo de MSE.

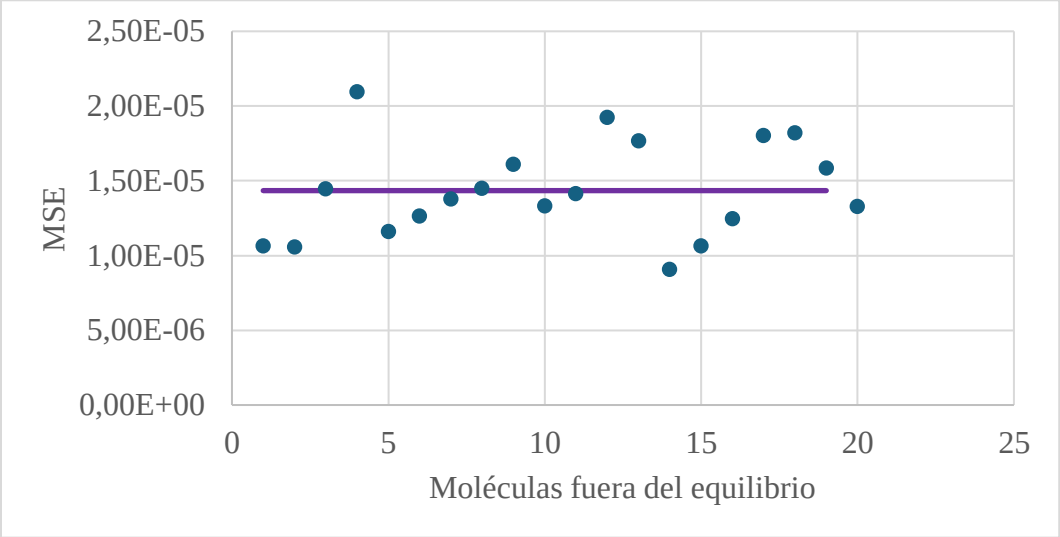

Figura 8. Error generado por la red neuronal en el Entrenamiento 2 en moléculas fuera del entrenamiento

En este caso, es evidente que el error generado por la segunda red neuronal en la etapa de entrenamiento es mayor, lo cual se debe al mayor número de puntos a evaluar que genera saturación en la red neuronal y

mantiene el error en un valor de equilibrio. Sin embargo, en la evaluación de su rendimiento en moléculas fuera del entrenamiento, el error generado es mucho menor, casi la mitad en algunos casos. El mayor error corresponde a 2.0938e-5 y el menor error a 1.057e-5.

Finalmente, se analiza la densidad electrónica en una representación tridimensional resultante de ambos entrenamientos, en donde se incorpora la divergencia *de Kullback-Leibler* punto por punto.

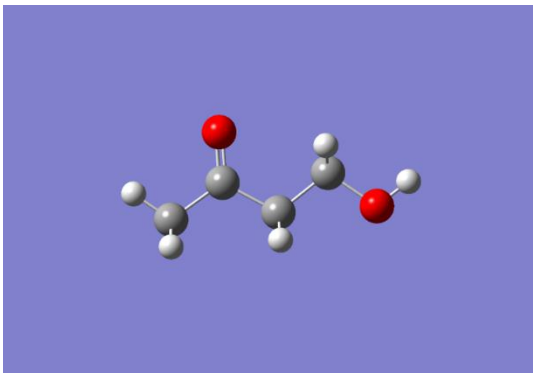

Figura 9. Molécula 4-Hidroxi-2-Butanona

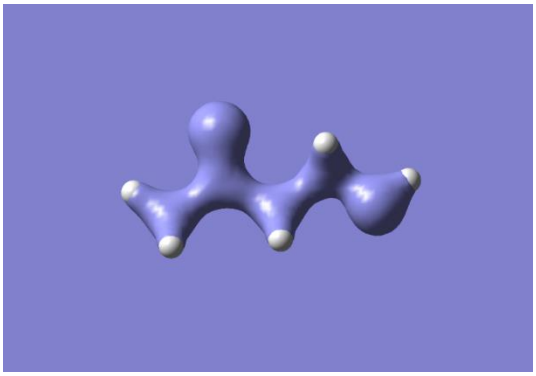

Figura 10. Densidad molecular obtenida por Multiwfn de la molécula 4-Hidroxi-2-Butanona, con un isovalor de 0.15 Bohr^{-3}.

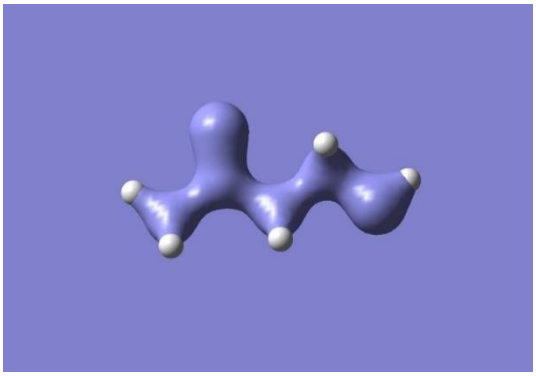

Figura 11. Densidad molecular aproximada (Entrenamiento 1) optimizada de la molécula 4-Hidroxi-2-Butanona, con un isovalor de 0.15 Bohr^{-3}.

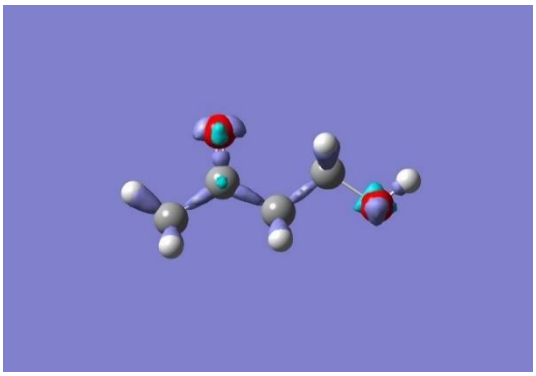

Figura 12. Error generado por la red neuronal (Entrenamiento 1) optimizada de la molécula 4-Hidroxi-2-Butanona, con un isovalor de 0.05 Bohr^{-3}. (Color morado) Subestimación. (Color celeste) Sobreestimación.

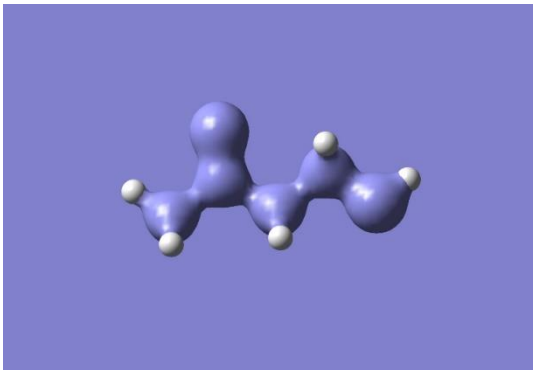

Figura 13. Densidad promolecular de la molécula 4-Hidroxi-2-Butanona, con un isovalor de 0.15 Bohr^{-3}.

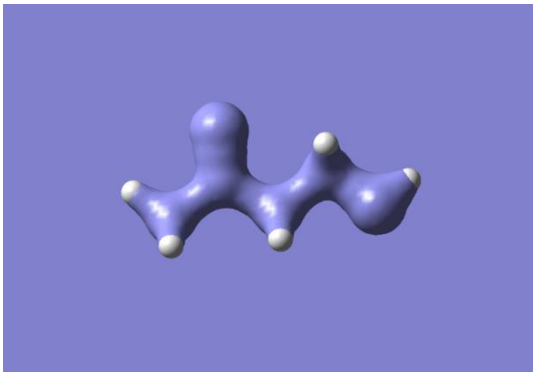

Figura 14. Densidad molecular aproximada (Entrenamiento 2) optimizada de la molécula 4-Hidroxi-2-Butanona, con un isovalor de 0.15 Bohr^{-3}.

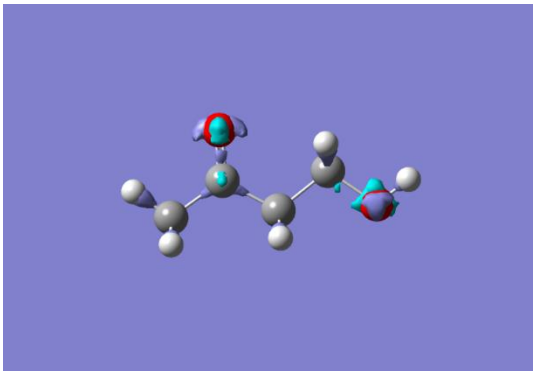

Figura 15. Error generado por la red neuronal (Entrenamiento 2) optimizada de la molécula 4-Hidroxi-2-Butanona, con un isovalor de 0.05 Bohr^{-3}. (Color morado) Subestimación. (Color celeste) Sobreestimación.

En este caso, se puede observar que ambas redes neuronales establecen los pares de electrones libres del oxigeno en la parte contraria del átomo, ya que la sobreestimación compensa la subestimación en las Figuras 12 y 15.

Por otro lado, la segunda red neuronal se acerca más a la densidad molecular de referencia, ya que se genera menor subestimación alrededor de los enlaces C-H, y el doble enlace C=O.

Se conoce que la densidad molecular aproximada es mayor en los electrones libres de átomos como N y O en grupos hidroxilos y aminas secundarias. Asimismo, es aún mayor en el grupo CN. Además, la densidad molecular aproximada es menor en las cercanías de los núcleos y enlaces covalentes (Sinitskiy & Pande, 2018).

Finalmente, se pretende utilizar la red neuronal óptima en aplicaciones a moléculas más complejas como aminoácidos, en este caso, se presentan las moléculas de leucina y glutamina, como ejemplos.

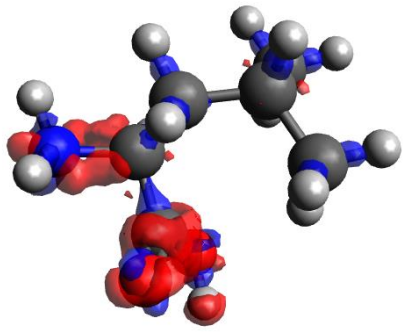

Figura 16. Error generado por la red neuronal (Entrenamiento 1) optimizada de la molécula Leucina, con un isovalor de 0.02 Bohr^{-3}. (Color azul) Subestimación. (Color rojo) Sobreestimación.

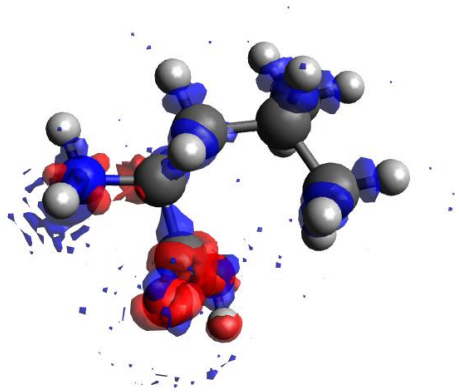

Figura 17. Error generado por la red neuronal (Entrenamiento 2) optimizada de la molécula Leucina, con un isovalor de 0.02 Bohr^{-3}. (Color azul) Subestimación. (Color rojo) Sobreestimación.

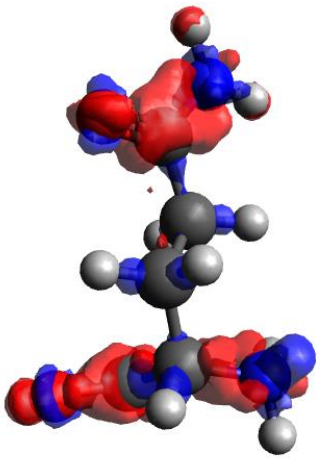

Figura 18. Error generado por la red neuronal (Entrenamiento 1) optimizada de la molécula Glutamina, con un isovalor de 0.02 Bohr^{-3}. (Color azul) Subestimación. (Color rojo) Sobreestimación.

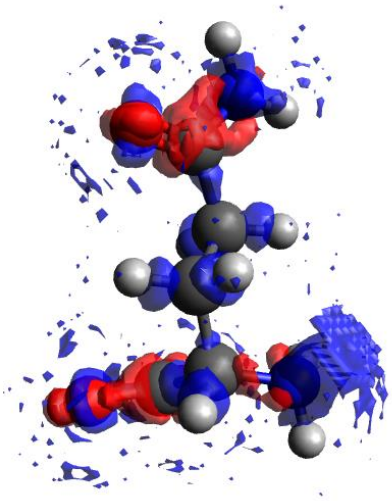

Figura 19. Error generado por la red neuronal (Entrenamiento 2) optimizada de la molécula Glutamina, con un isovalor de 0.02 Bohr^{-3}. (Color azul) Subestimación. (Color rojo) Sobreestimación.

5. CONCLUSIONES Y RECOMENDACIONES

La red neuronal más optima presenta un MSE mínimo de 1.057e-5, y máximo de 2.0938e-5. La red neuronal predice la presencia de dobles enlaces y pares de electrones libres, aunque no los coloca en la posición correcta. La red neuronal sobreestima la densidad electrónica en los electrones libres de átomos como N y O en grupos hidroxilos y aminas secundarias, así como en grupos CN, pero subestima en las cercanías de los núcleos y enlaces covalentes.

La red neuronal es transferible a sistemas más complejos como aminoácidos, ya que se puede utilizar en sistemas más generales de moléculas orgánicas, que contengan C, H, O y N.

Se recomienda aumentar el tamaño de la base de datos inicial para aumentar la diversidad química, ya que otros estudios utilizan alrededor de 2000 moléculas en la etapa de entrenamiento (Fabrizio, Grisafi, Meyer, Ceriotti, & Corminboeuf, 2019).

Se recomienda utilizar otro tipo de densidad inicial, en cambio de la densidad promolecular, ya que presenta cierta desventajas como discontinuidad en los puntos sobre los núcleos, y superposición en ciertas regiones del espacio.

Se recomienda utilizar redes neuronales más complejas como aquellas *Non-local neural networks* (NLNN) o implementar una herramienta dentro de la red neuronal que cree subconjuntos de datos para diferentes entrenamientos adecuados para diferentes clases de compuestos químicos (Fabrizio, Grisafi, Meyer, Ceriotti, & Corminboeuf, 2019).

6. REFERENCIAS

Bader, R. F. (1990). *Atoms in Molecules: A Quantum Theory.* Oxford University Press. doi:10.1093/oso/9780198551683.001.0001

Bunge, Carlos & Barrientos, J.A. & Vivier-bunge, Annik. (1993). Roothaan-Hartree-Fock Ground-State Atomic Wave-Functions - Slater-Type Orbital Expansions and Expectation Values for Z=2-54. Atom. Data Nucl. Data Tables. 53. 113-162. 10.1006/adnd.1993.1003

Brockherde, F.; Vogt, L.; Li, L.; Tuckerman, M. E.; Burke, K.; M¨uller, K.-R. (2017) Bypassing the Kohn-Sham Equations with Machine Learning. Nat. Commun.

Cover, T. M., & Thomas, J. A. (2005). Elements of Information Theory. doi:DOI:10.1002/047174882X

Cuevas-Zuviría, B., & Pacios, L. F. (2020, August 4). Analytical Model of Electron Density and Its Machine Learning Inference. Journal of Chemical Information and Modeling. American Chemical Society (ACS). http://doi.org/10.1021/acs.jcim.0c00197

Fabrizio, A., Grisafi, A., Meyer, B., Ceriotti, M., & Corminboeuf, C. (2019). Electron density learning of non-covalent systems. *Chemical Science*, 9424-9432. doi:DOI https://doi.org/10.1039/C9SC02696G

González Forero, D. (2013). *Química Computacional.* Bogotá: UNIVERSIDAD NACIONAL ABIERTA Y A DISTANCIA – UNAD. Retrieved from https://repository.unad.edu.co/bitstream/handle/10596/12499/210116-contenido%20del%20curso.pdf?sequence=1

Grisafi, A., Fabrizio, A., Meyer, B., Wilkins, D. M., Corminboeuf, C., & Ceriotti, M. (2019). Transferable Machine-Learning Model of the Electron Density. *ACS Central Science*, 57-64. doi:doi: 10.1021/acscentsci.8b00551

Johnson, E. R., Keinan, S., Sánchez, P., & Contreras-García, J. (2010). Revealing Non Covalent Interactions. *Jacs Articles*, 6498–6506. doi:doi:10.1021/ja100936w

McQuarrie, D. A., & Simon, J. D. (1997). *Physical Chemistry A Molecular Approach.* California: University Science Books.

R. Ramakrishnan, P. O. Dral, M. Rupp, O. A. von Lilienfeld, Quantum chemistry structures and properties of 134 kilo molecules, Scientific Data 1, 140022, 2014.

Rincón, L., Seijas, L., Almeida, R., & Torres, F. (2023). Towards the construction of an accurate kinetic energy density functional and its functional derivative through physics-informed neural networks. *Journal of Physics Communications*, 061001. doi:10.1088/2399-6528/acd90e

Sinitskiy, A. V., & Pande, V. S. (2018). *Deep Neural Network Computes Electron Densities and Energies of a Large Set of Organic Molecules Faster than Density Functional Theory (DFT).* California: Standford University.

7. ANEXO A

Entrenamiento 1

Tabla 1. Error calculado de la red neuronal en el Entrenamiento 1 y medición del error con MSE

Medición del error	**Error cuadrático medio (MSE)**					
Función de activación	Tanh(x)			ReLu(x)		
Distribución de capas y neuronas	Tiempo [s]	Step-loss	Val-loss	Tiempo [s]	Step-loss	Val-loss
1º (200) 2º (100) 3º (50)	15:38:55	1,7221E-06	5,8854E-06	10:19:16	5,1512E-07	3,8566E-07
1º (100) 2º (50) 3º (25)	10:11:20	6,5453E-07	5,2516E-07	6:45:09	5,8435E-07	4,5610E-07
1º (50) 2º (25) 3º (12)	4:37:05	5,7759E-07	4,1226E-07	5:34:18	4,7393E-07	4,5844E-07
1º (200) 2º (100) 3º (50) 4º (25)	9:57:55	1,0657E-06	4,6150E-07	8:18:28	5,6069E-07	3,5230E-07
1º (100) 2º (50) 3º (25) 4º (12)	6:34:08	6,0947E-07	5,3440E-07	5:14:53	5,3442E-07	3,8482E-07
1º (30) 2º (30)	3:59:08	6,9411E-07	4,1435E-07	3:44:55	5,6965E-07	3,5454E-07
1º (100) 2º (100) 3º (100) 4º (100)	9:37:24	5,3771E-07	4,7334E-07	8:26:58	5,1776E-07	5,1128E-07
1º (100) 2º (100) 3º (100)	8:06:16	6,2670E-07	4,8466E-07	7:12:07	4,8779E-07	3,6570E-07
1º (100) 2º (100)	10:55:53	7,8341E-07	5,7987E-07	5:02:56	6,3213E-07	3,8704E-07
1º (50) 2º (50) 3º (50) 4º (50)	10:05:33	5,5747E-07	4,6760E-07	5:10:48	4,4913E-07	3,8244E-07
1º (50) 2º (50) 3º (50)	9:27:15	7,2905E-07	4,6581E-07	7:35:49	4,8634E-07	4,0106E-07
1º (50) 2º (50)	8:46:56	8,3896E-07	5,5937E-07	5:43:35	6,0038E-07	4,0501E-07
1º (200) 2º (200)	23:01:31	2,8433E-05	9,1356E-05	26:54:35	4,9419E-07	5,3422E-07

3º (200) **4º (200)**						
1º (200) **2º (200)** **3º (200)**	18:11:45	1,4034E-06	1,0471E-06	21:16:45	4,8124E-07	3,6162E-07
1º (200) **2º (200)**	10:50:05	1,5115E-06	7,9471E-07	15:26:46	5,6618E-07	4,1508E-07

Tabla 2. Error calculado de la red neuronal en el Entrenamiento 1 y medición del error con MAE

Medición del error	**Error absoluto medio (MAE)**					
Función de activación	Tanh(x)			ReLu(x)		
Distribución de capas y neuronas	Tiempo [s]	Step-loss	Val-loss	Tiempo [s]	Step-loss	Val-loss
1º (200) **2º (100)** **3º (50)**	3:00:31	1,1267E-04	1,3804E-04	2:46:38	2,0342E-04	2,3165E-04
1º (100) **2º (50)** **3º (25)**	1:56:40	1,3434E-04	1,8761E-04	1:37:15	1,8422E-04	2,3740E-04
1º (50) **2º (25)** **3º (12)**	1:06:40	9,8721E-04	1,4000E-03	0:50:00	1,4481E-04	1,1347E-04
1º (200) **2º (100)** **3º (50)** **4º (25)**	3:19:59	1,5000E-03	1,8000E-03	3:02:33	1,3837E-04	2,0870E-04
1º (100) **2º (50)** **3º (25)** **4º (12)**	1:56:40	1,6000E-03	4,8542E-04	1:40:00	2,0553E-04	3,2217E-04
1º (30) **2º (30)**	0:50:00	1,3280E-04	2,0189E-04	0:50:00	1,4751E-04	5,7920E-05
1º (100) **2º (100)** **3º (100)** **4º (100)**	3:20:00	2,0543E-04	2,3314E-04	3:06:28	1,5119E-04	1,9145E-04
1º (100) **2º (100)** **3º (100)**	2:30:00	1,9315E-04	1,8123E-04	2:13:21	1,8075E-04	9,2179E-05
1º (100) **2º (100)**	1:40:00	1,1139E-04	7,8133E-05	1:40:00	1,5017E-04	7,6300E-05
1º (50) **2º (50)** **3º (50)** **4º (50)**	2:03:36	1,9434E-04	1,0966E-04	1:49:01	1,3568E-04	3,4277E-05

1º (50) **2º (50)** **3º (50)**	1:40:01	1,8028E-04	8,4518E-05	1:23:20	1,2607E-04	1,6687E-04
1º (50) **2º (50)**	1:22:57	1,3871E-04	4,7865E-05	1:05:56	1,4147E-04	1,0565E-04
1º (200) **2º (200)** **3º (200)** **4º (200)**	7:55:31	2,0746E-04	1,0430E-04	8:07:04	2,0513E-04	1,4923E-04
1º (200) **2º (200)** **3º (200)**	6:51:03	1,9296E-04	3,5898E-04	7:02:33	1,4621E-04	1,6148E-04
1º (200) **2º (200)**	3:48:03	1,4768E-04	1,0252E-04	4:39:36	1,6170E-04	6,7367E-05

8. ANEXO B

Entrenamiento 2

Tabla 3. Error calculado de la red neuronal en el Entrenamiento 2 y medición del error con MSE

Medición del error	**Error cuadrático medio (MSE)**					
Función de activación	Tanh(x)			ReLu(x)		
Distribución de capas y neuronas	Tiempo [s]	Step-loss	Val-loss	Tiempo [s]	Step-loss	Val-loss
1º (200) 2º (100) 3º (50)	23:32:58	3,6998E-06	2,0641E-06	25:11:05	3,5427E-06	2,0022E-06
1º (100) 2º (50) 3º (25)	24:08:31	1,7372E-06	1,0990E-06	15:46:20	3,3659E-06	1,4668E-06
1º (50) 2º (25) 3º (12)	19:33:07	1,6947E-06	1,3444E-06	12:41:54	2,4682E-06	1,5795E-06
1º (200) 2º (100) 3º (50) 4º (25)	25:32:41	2,0632E-06	1,1488E-06	26:01:02	3,2507E-06	1,7015E-06
1º (100) 2º (50) 3º (25) 4º (12)	14:56:01	1,6463E-06	1,3989E-06	19:14:10	5,7493E-06	4,8661E-06
1º (30) 2º (30)	8:46:43	1,7451E-06	1,1090E-06	11:20:57	3,8470E-06	3,5539E-06
1º (100) 2º (100) 3º (100) 4º (100)	27:03:00	1,5984E-06	1,1084E-06	19:47:39	2,2771E-06	1,3010E-06
1º (100) 2º (100) 3º (100)	18:45:45	4,5039E-06	2,8930E-06	15:51:25	2,2204E-06	1,3750E-06
1º (100) 2º (100)	16:46:05	8,1042E-06	9,5235E-06	14:19:01	2,6360E-06	1,5562E-06
1º (50) 2º (50) 3º (50) 4º (50)	13:49:17	9,8093E-06	2,1966E-06	11:32:44	2,1959E-06	1,3201E-06
1º (50) 2º (50) 3º (50)	16:03:43	6,2169E-06	2,8276E-06	6:49:46	2,0618E-06	1,4757E-06
1º (50) 2º (50)	10:20:38	3,9072E-06	3,7551E-06	17:38:00	1,7368E-06	1,2211E-06
1º (200) 2º (200) 3º (200)	41:26:50	1,8979E-05	3,6309E-06	35:25:21	2,6444E-06	1,4183E-06

4º (200)						
1º (200) **2º (200)** **3º (200)**	30:20:32	1,1327E-05	3,7063E-06	28:49:10	2,4400E-06	1,5785E-06
1º (200) **2º (200)**	23:46:27	1,8772E-05	2,9719E-06	37:59:32	1,6694E-06	1,1330E-06

Tabla 4. Error calculado de la red neuronal en el Entrenamiento 2 y medición del error con MAE

Medición del error	**Error absoluto medio (MAE)**					
Función de activación	Tanh(x)			ReLu(x)		
Distribución de capas y neuronas	Tiempo [s]	Step-loss	Val-loss	Tiempo [s]	Step-loss	Val-loss
1º (200) **2º (100)** **3º (50)**	63:27:46	2,1074E-07	2,1306E-04	32:46:35	1,8280E-04	1,4122E-04
1º (100) **2º (50)** **3º (25)**	17:27:33	2,0271E-04	2,0991E-04	18:46:19	1,8146E-04	1,5577E-04
1º (50) **2º (25)** **3º (12)**	14:38:53	6,5960E-04	4,9017E-04	15:53:12	1,9032E-04	1,0558E-04
1º (200) **2º (100)** **3º (50)** **4º (25)**	29:50:54	2,9000E-03	1,7000E-03	34:55:32	1,8663E-04	1,4526E-04
1º (100) **2º (50)** **3º (25)** **4º (12)**	18:12:42	5,0981E-04	5,5951E-04	22:48:15	3,1808E-04	2,1129E-04
1º (30) **2º (30)**	13:16:10	2,2301E-04	2,3949E-04	12:48:11	1,9257E-04	1,6272E-04
1º (100) **2º (100)** **3º (100)** **4º (100)**	43:25:30	1,9147E-04	2,0222E-04	26:45:30	2,2134E-04	2,2747E-04
1º (100) **2º (100)** **3º (100)**	30:11:55	2,0453E-04	1,4767E-04	34:31:58	1,8378E-04	1,0268E-04
1º (100) **2º (100)**	18:17:10	2,3029E-04	2,3037E-04	22:54:07	1,9784E-04	1,6518E-04
1º (50) **2º (50)** **3º (50)** **4º (50)**	18:08:10	1,9093E-04	1,9226E-04	24:32:10	1,9558E-04	1,4949E-04
1º (50) **2º (50)**	15:17:26	2,0273E-04	2,6385E-04	16:49:07	2,0060E-04	1,8391E-04

3º (50)						
1º (50) **2º (50)**	11:53:17	2,1912E-04	2,6412E-04	13:55:31	1,9277E-04	9,2668E-05
1º (200) **2º (200)** **3º (200)** **4º (200)**	83:43:40	2,1001E-04	1,2002E-04	43:33:20	1,9419E-04	2,2899E-04
1º (200) **2º (200)** **3º (200)**	39:35:25	2,0745E-04	1,4101E-04	35:05:30	1,9732E-04	1,9996E-04
1º (200) **2º (200)**	36:55:58	2,3302E-04	2,3523E-04	27:09:39	1,9127E-04	2,1833E-04