



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE QUÍMICA

CALIBRACIÓN DE OCUPACIONES FRACCIONALES
TIPO FERMI-DIRAC ORIENTADAS A DESCRIBIR
FENÓMENOS QUÍMICOS ALTAMENTE
CORRELACIONADOS

T E S I S

QUE PARA OPTAR POR EL GRADO DE:

Químico

PRESENTA:

José Alberto Cabrera Jaime

TUTOR DE TESIS:

Dr. José Marco Antonio Franco Pérez



Ciudad Universitaria, Cd. Mx., 2020



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Agradecimientos

Me gustaría expresar sinceramente mi agradecimiento a todas las personas que me ayudaron en el proceso de realización de esta tesis.

A mi tutor de tesis, el Dr. Marco Franco, por su disponibilidad, consejo y experiencia que fueron muy valiosos durante el desarrollo de este trabajo. A mis sinodales, el Dr. Marco Franco, el Dr. Miguel Castro y el Dr. Raúl Álvarez, por tomarse el tiempo de leer esta tesis y hacer las correcciones necesarias.

A mis padres, Luis y Virginia, por criarme de una manera digna de admiración y respeto. Les agradezco desde el fondo de mi corazón su sacrificio, amor y apoyo durante toda mi vida.

A mi hermana, Marian, por animarme en mis momentos de estrés.

A mi familia y amigos ya que todos ellos han contribuido a la realización de mi tesis, con su ayuda moral y material.

A todos los profesores y personal de la Facultad de Química, por su ayuda de diversas maneras para poder completar mi formación académica y profesional. Así mismo, agradezco el apoyo financiero brindado por el Subprograma 127 - Formación Básica e Investigación - y por parte del proyecto DGAPA-PAPIIT IA207719.

Índice general

Agradecimientos	II
Introducción	1
1. Marco teórico	2
1.1. Métodos <i>ab initio</i>	2
1.1.1. Método Hartree-Fock	3
1.1.2. Correlación electrónica	5
1.1.3. Teoría del funcional de la densidad	6
1.2. Números de ocupación	9
1.2.1. Números de ocupación fraccionales	9
1.3. Redes Neuronales	10
1.3.1. Inspiración en la biología	10
1.3.2. Estructura	12
1.3.3. Función de activación	13
1.3.4. Función de costo	13
1.3.5. Entrenamiento	15
1.3.6. El problema de la parametrización	17
1.3.7. Implementación de una red neuronal	18
1.3.8. Aprendizaje máquina en química cuántica	18
2. Desarrollo Teórico	19
2.1. Procedimiento iterativo para obtener E_θ	20
2.2. Procedimiento iterativo para obtener b_{opt}	21
2.3. Implicaciones	22
3. Objetivos	26
3.1. Objetivo general	26
3.2. Objetivos particulares	26
4. Metodología	27
4.1. Detalles computacionales	27
4.2. Preparación de la base de datos	27
4.3. Entradas	34
4.4. Entrenamiento	37
5. Resultados y Discusión	41
5.1. Resultados del entrenamiento	41
5.2. Cálculo de la correlación	46
5.2.1. Disociación de la molécula de hidrógeno	46
5.2.2. Disociación de la molécula de nitrógeno	48
5.2.3. Disociación de la molécula de carbono	49
5.2.4. Rotación de la molécula de etileno	51

5.2.5. Moléculas de entre 80 y 300 electrones	54
5.2.6. Isómeros de C ₆₀	56
6. Conclusiones	59
A. Códigos en Python	61
B. Base de datos	68

Introducción

La correcta descripción de los diversos fenómenos químicos, mediante los métodos de la química cuántica, ha encontrado como inconveniente principal la incapacidad de resolver la ecuación de Schrödinger para sistemas constituidos por N electrones interactuantes. Una de las aproximaciones de partida más importantes en la actualidad es el denominado método de Hartree-Fock (HF). Aunque HF recupera gran parte de la energía exacta de un sistema electrónico, este resulta insuficiente para describir fenómenos (ruptura, formación y disociación de enlaces, etc) y propiedades (momentos dipolares, polarizabilidades, energías de ionización, etc.) de naturaleza química, de manera confiable. En este contexto, se ha introducido el concepto de *energía de correlación electrónica* como la diferencia entre la energía exacta del sistema y la energía HF, y es uno de los principales ejes de investigación en la química teórica actual.

Recientemente, se ha sugerido que los diferentes estados electrónicos pueden mezclarse utilizando una distribución tipo Fermi-Dirac. Dentro de este enfoque, los números fraccionales indicarán la importancia ponderada de un estado mono-electrónico en particular (incluyendo el espectro completo de estados virtuales definido por el espacio de funciones base seleccionado), en la descripción de la energía de correlación electrónica total. No obstante, se desconoce la manera en que dicha distribución debe de optimizarse para tal fin. En este proyecto, la optimización de las ocupaciones se realizará mediante un algoritmo basado en inteligencia artificial, particularmente en redes neuronales secuenciales. Si la propiedad de transferibilidad se mantiene, la red así entrenada proporcionará la mejor distribución ocupacional para alguna especie en particular, lo que permitirá realizar un predicción precisa de las propiedades de estructura electrónica correspondientes al mismo costo computacional que el método HF.

El presente proyecto está orientado a la calibración de una red neuronal con la cual sea posible construir una distribución de Fermi-Dirac que permita una predicción confiable de la energía de correlación electrónica de las especies químicas. Esta red será aplicada particularmente a sistemas cuya descripción mecánico-cuántica es altamente dependiente de la correlación electrónica. Se reproducirán los perfiles de disociación de sistemas moleculares diatómicos y el perfil de rotación de la molécula de etileno. Posteriormente aplicaremos la red neuronal para calcular los números de ocupación de sistemas grandes como isómeros de fullereno.

1.1. Métodos *ab initio*

El estudio de la Química Cuántica y Computacional se enfoca en la obtención de la estructura electrónica de sistemas atómicos y moleculares para determinar sus propiedades físicas y químicas. Se puede escoger de entre una amplia variedad de métodos para este fin, en función de la naturaleza del sistema, de la propiedad que se busca obtener y del poder computacional con el que se cuente. *The Goldbook* [1] define a los métodos *ab initio* como aquellos que hacen uso de la ecuación de Schrödinger para tratar al sistema de estudio.

Cualquier problema de estructura electrónica se puede abordar con la ecuación de Schrödinger [2-4]. Para un sistema de átomos aislados o moléculas aisladas con N electrones, en la aproximación Born-Oppenheimer no relativista, se tiene

$$\hat{H}\Psi = E\Psi \quad (1.1)$$

donde E es la energía electrónica, $\Psi = \Psi(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_N)$ es la función de onda, y \hat{H} es el Hamiltoniano

$$\hat{H} = \sum_{i=1}^N \left(-\frac{1}{2} \nabla_i^2 \right) + \sum_{i=1}^N v(\bar{r}_i) + \sum_{i<j}^N \frac{1}{\bar{r}_{ij}} \quad (1.2)$$

donde

$$v(\bar{r}_i) = - \sum_A \frac{Z_A}{\bar{r}_{iA}} \quad (1.3)$$

es el potencial externo actuando sobre el electrón i , potencial debido a los núcleos A con carga Z_A . Las coordenadas \bar{x}_i corresponden a coordenadas espaciales continuas \bar{r}_i y coordenadas de espín discretas ω_i .

Funciones de onda precisas para sistemas moleculares de muchas partículas son extremadamente difíciles de expresar por términos de correlación, o interdependencia, entre las partículas. Esto se debe a que el operador Hamiltoniano contiene términos de atracción y repulsión, lo que

significa que ninguna partícula se mueve independientemente de las demás [5]. La clave para la resolución de la ecuación (1.1) aplicada a átomos y moléculas está en las posibles simplificaciones o aproximaciones que se pueden hacer. En estas aproximaciones se basan los distintos métodos computacionales existentes.

1.1.1. Método Hartree-Fock

Debido al término correspondiente a la repulsión electrónica $1/\bar{r}_{ij}$ en la ecuación (1.2), la ecuación de Schrödinger no es separable en ningún sistema de coordenadas y es necesario emplear métodos aproximados. Uno de ellos es aproximar el problema de N electrones a N problemas independientes de 1 electrón. Al considerar electrones no interactuantes, el operador Hamiltoniano sólo contiene a los términos de energía cinética y energía potencial de atracción electrón núcleo,

$$\hat{H} = \sum_i^N \hat{h}(i) = \sum_i^N \left(-\frac{1}{2} \nabla_i^2 + v(\bar{r}_i) \right) \quad (1.4)$$

Las funciones propias del operador \hat{h} pueden ser los espín-orbitales $\psi(\bar{r}, \omega) = \psi(\bar{x})$ del nuevo sistema de electrones independientes

$$\hat{h}(i)\psi_j(\bar{x}_i) = \varepsilon_j \psi_j(\bar{x}_i) \quad (1.5)$$

El *producto de Hartree* Ψ^{PH} es la función de onda producto de los espín-orbitales de cada electrón

$$\Psi^{PH}(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_N) = \psi_i(\bar{x}_1)\psi_j(\bar{x}_2) \dots \psi_k(\bar{x}_N) \quad (1.6)$$

y es una función propia del operador \hat{H} para electrones no interactuantes,

$$\hat{H}\Psi^{PH} = E\Psi^{PH} \quad (1.7)$$

con el valor propio E , que es la suma de las energías de cada espín-orbital en Ψ^{PH}

$$E = \varepsilon_i + \varepsilon_j + \dots + \varepsilon_k \quad (1.8)$$

Dado que los electrones son fermiones, la función de onda debe ser antisimétrica respecto al intercambio de coordenadas (espaciales y de espín) de 2 electrones cualesquiera. La función de onda antisimétrica más simple se construye con un determinante de Slater. Para un sistema de N electrones, el determinante tiene la forma

$$\Psi(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_N) = (N!)^{-\frac{1}{2}} \begin{vmatrix} \psi_i(\bar{x}_1) & \psi_j(\bar{x}_1) & \dots & \psi_k(\bar{x}_1) \\ \psi_i(\bar{x}_2) & \psi_j(\bar{x}_2) & \dots & \psi_k(\bar{x}_2) \\ \vdots & \vdots & & \vdots \\ \psi_i(\bar{x}_N) & \psi_j(\bar{x}_N) & \dots & \psi_k(\bar{x}_N) \end{vmatrix} \quad (1.9)$$

de modo que hay N electrones ocupando N orbitales. Una función de onda presentada de esta manera es una función de onda Hartree-Fock (HF).

El principio variacional estipula que la mejor función de onda es la que proporciona la menor energía como valor propio al aplicar el operador Hamiltoniano. En la ecuación (1.4) se escribió el operador \hat{h} dentro de la aproximación de electrones independientes. No obstante, de manera alternativa, también puede ser escrito como un operador de 1 electrón que incluye efectos de repulsión de manera promediada. Este nuevo operador tiene como nombre *operador de Fock*,

$$\hat{f}(i) = \hat{h}(i) + v^{HF}(i) \quad (1.10)$$

donde $v^{HF}(i)$ es la interacción promedio del potencial que experimenta el i -ésimo electrón por la presencia del resto de los $N - 1$ electrones. Siguiendo el desarrollo al aplicar el operador de Fock \hat{f} sobre la función de onda HF [6], se obtiene la expresión de la energía

$$E = \sum_a^N \langle \psi_a | \hat{h} | \psi_a \rangle + \frac{1}{2} \sum_{ab}^N \left(\langle \psi_a \psi_b | \psi_a \psi_b \rangle - \langle \psi_a \psi_b | \psi_b \psi_a \rangle \right) \quad (1.11)$$

donde

$$\langle \psi_a \psi_b | \psi_a \psi_b \rangle = \int |\psi_a(\bar{x}_1)|^2 \frac{1}{r_{12}} |\psi_b(\bar{x}_2)|^2 d\bar{x}_1 d\bar{x}_2 \quad (1.12)$$

es la integral coulombica y

$$\langle \psi_a \psi_b | \psi_b \psi_a \rangle = \int \psi_a^*(\bar{x}_1) \psi_b^*(\bar{x}_2) \frac{1}{r_{12}} \psi_b(\bar{x}_1) \psi_a(\bar{x}_2) d\bar{x}_1 d\bar{x}_2 \quad (1.13)$$

es la integral de intercambio.

La integral coulombica es la repulsión coulombica clásica entre las nubes electrónicas $|\psi_a(\bar{x}_1)|^2$ y $|\psi_b(\bar{x}_2)|^2$. La integral de intercambio no tiene una interpretación clásica sencilla, es resultado de la naturaleza antisimétrica en la función de onda HF y su valor está relacionado con el intercambio de los electrones 1 y 2 entre los espín-orbitales ψ_i y ψ_j . Como ambas integrales son positivas y la integral de intercambio da 0 para electrones con espín opuesto, la integral de intercambio puede interpretarse como una “corrección” debido a la interacción de electrones con espines iguales (paralelos). Esto significa que el movimiento de 2 electrones con espines paralelos está correlacionado, pero no así el de espines iguales.

Se puede pensar en la energía total de un sistema de N electrones, descrito por un determinante de Slater, como la suma de energías mono-electrónicas más la suma de energías de interacción. Sumando todas las interacciones posibles en la ecuación (1.11), se obtiene el potencial promedio total actuando sobre cada electrón, debido a los $N - 1$ electrones restantes.

Las ecuaciones de HF son resultado de aplicar el operador de Fock sobre cada uno de los espín-orbitales del sistema. Es evidente que para obtener la expresión del espín-orbital que describe al i -ésimo electrón, es necesario conocer el resto de los espín-orbitales para calcular el potencial promedio, pues el potencial $v^{HF}(i)$ en la ecuación (1.10) depende de los otros electrones. Por lo

tanto, las ecuaciones de HF son no lineales y se deben resolver de manera iterativa con el método de *campo autoconsistente*: obtener la función propia de \hat{f} para cada uno de los espín-orbitales y repitiendo hasta que los espín-orbitales ya no cambien, asegurando que el campo efectivo experimentado con cada electrón sea “consistente” con el del resto. Este procedimiento es el denominado *método Hartree-Fock* (HF).

La importancia del método HF radica en que es posible mejorar su exactitud de manera sistemática y se utiliza como punto de partida para métodos de estructura electrónica más complejos [7]. En la actualidad sigue siendo utilizado al momento de probar nuevos algoritmos [8, 9] y lenguajes de programación [10] para química cuántica.

1.1.2. Correlación electrónica

A pesar de que el movimiento de pares de electrones con espines paralelos está correlacionado dentro de la aproximación HF, no se puede decir lo mismo con electrones de espines opuestos. Es posible demostrar [6] que la función de onda HF presenta una probabilidad $P = 0$ para encontrar a 2 electrones del mismo espín en la misma región del espacio, este “agujero” de probabilidad en electrones con espín paralelo se le conoce como *agujero de Fermi*. El agujero de Fermi no está presente en electrones de espines opuestos dentro de la aproximación HF, lo que indica que su movimiento es independiente y no está correlacionado uno con el otro. Esto es contradictorio con el término $1/\bar{r}_{ij}$ del Hamiltoniano, indeterminado en $\bar{r}_{ij} = 0$ a pesar de que E tiene un valor finito. La función de onda necesita tener un agujero de probabilidad $P = 0$ con cualquier par de electrones cuando $\bar{r}_{ij} = 0$ para la completa descripción del sistema, este agujero de probabilidad es llamado *agujero de Coulomb*.

El agujero de Coulomb considera el movimiento correlacionado de los electrones debido a la repulsión instantánea, provocando también una mejora en la descripción del sistema. La correlación electrónica debida al movimiento de los electrones recibe el nombre de *correlación dinámica*. Además, existe la llamada *correlación estática*, que se origina debido a las interacciones entre estados electrónicos casi degenerados [11].

Tanto la correlación dinámica como la correlación estática son necesarias para obtener resultados cuantitativos y poder describir al fenómeno químico de manera completa. El método HF no proporciona este nivel de exactitud: las energías que calcula son del orden de 1% más altas que la energía exacta. Esta diferencia de energía es conocida como *energía de correlación* [12], y es definida como la diferencia entre la energía exacta (no relativista) E y la energía obtenida por el método HF E_{HF} ,

$$E_{corr} = E - E_{HF} \quad (1.14)$$

Incluir la superposición de determinantes en la función de onda tiene el doble propósito de considerar los efectos de repulsión Coulómbica instantánea y la degeneración de estados electrónicos [13], resolviendo así el problema de correlación electrónica en su totalidad.

Un método flexible y que puede ser utilizado extensivamente por su simplicidad conceptual para el cálculo de la correlación electrónica es el método de interacción de configuraciones (CI, por sus siglas en inglés). La idea detrás de CI es tomar a la función de onda como una combinación lineal de varios determinantes de N electrones. Cada uno de estos determinantes es una configuración posible de los N electrones en los $2K$ espín-orbitales definidos para el sistema y se describen

considerando como referencia al determinante Ψ resultado de HF. Estos métodos son variacionales, pero no son consistentes en tamaño porque es necesario truncar la expansión CI para hacerlos computacionalmente aplicables [14]. Si se consideran todas las configuraciones con una doble sustitución sobre la referencia HF, el método es llamado CISD; si se consideran también las sustituciones triples, se le llama CISDT, etc.; cuando se consideran todas las posibles sustituciones, se le llama *full CI*.

Otro grupo de métodos para calcular correlación electrónica está basado en la teoría de perturbaciones, introducidos originalmente por Møller Plesset (MP) [15]. Estos métodos tratan a la correlación electrónica como una perturbación en el Hamiltoniano de HF y toman a la energía como una expansión truncada hasta cierto orden [16]. Aunque estos métodos son consistentes en tamaño, no son variacionales y convergen lentamente.

También están los métodos que utilizan la teoría de Cúmulos Acoplados (CC, por sus siglas en inglés) [17]. Los métodos CC introducen la sustitución de configuraciones de manera multiplicativa, a diferencia de la manera aditiva en CI. Se utiliza la misma terminología que en CI para especificar el tipo de sustitución que se realiza: cuando se limita el método a sustituciones dobles, se le llama CCSD, etc. Este procedimiento es consistente en tamaño, pero no es variacional.

Existe también la serie de métodos llamados interacción cuadrática de configuraciones (QCI, por sus siglas en inglés). La idea en estos métodos es una modificación a las ecuaciones de CI para recuperar consistencia en tamaño, a cambio de una pérdida en el carácter variacional [18-20]. En 1987, Pople y Head-Gordon concluyen que la introducción de las contribuciones por sustituciones triples con el método QCISD(T) resulta en una mejora sustancial a la energía electrónica de hasta 1.2 mhartrees de diferencia por debajo de la energía obtenida con *full CI* (para algunos sistemas selectos donde este cálculo CI es posible), además de adquirir una exactitud de 0.17 mhartrees mayor que CCSD(T). Esto sugiere que la no-variacionalidad de QCISD(T) es un bajo precio a pagar a cambio de la consistencia en tamaño que adquiere, además de una pequeña mejora con respecto a CCSD(T). A pesar de esto, es importante mencionar que el método ha demostrado ser inexacto en algunas disociaciones si se incluye la corrección de excitaciones triples [21].

Estos métodos para el cálculo de correlación electrónica, que aplican una corrección utilizando de punto de partida a la aproximación HF (métodos *post-HF*), son de costo computacional superior y su aplicación se limita a sistemas con pocos electrones.

1.1.3. Teoría del funcional de la densidad

A pesar de la corrección que se logra con los métodos post-HF, la función de onda tiene la desventaja de ser prácticamente no-interpretable. La introducción de la *densidad electrónica* presenta la ventaja de utilizar un observable físico como el ente matemático del cuál obtener la estructura electrónica [22].

En un sistema electrónico, el número de electrones por unidad de volumen en un estado dado es la *densidad electrónica* de ese estado [23]. La densidad electrónica en términos de Ψ es

$$\rho(\bar{r}) = \sum_i^N |\psi(\bar{x}_i)|^2 \quad (1.15)$$

tal que $\rho(\bar{r})d\bar{r}$ es la probabilidad de encontrar a cualquier electrón del sistema en la volumen $d\bar{r}$

alrededor de \bar{r} . La integral de la densidad electrónica sobre todo el espacio da como resultado el número total de electrones,

$$\int \rho(\bar{r}) d\bar{r} = \sum_i^N \int |\psi(\bar{x}_i)|^2 d\bar{r} = \sum_i^N 1 = N \quad (1.16)$$

En la teoría del funcional de la densidad (TFD), la energía del estado basal se escribe en términos de $\rho(\bar{r})$ y la idea detrás de esta es realizar cálculos de estructura electrónica representando al potencial externo como un funcional de la densidad electrónica. El nombre “funcional” indica una función de otra función, que en este caso es la densidad electrónica.

La idea original de la TFD comenzó en los años 1920 con el modelo de Thomas y Fermi [24], aproximando la densidad electrónica a la de un gas homogéneo de electrones. A pesar de las diversas mejoras introducidas al modelo, este no puede reproducir resultados siquiera cualitativos, provocando que sea visto como una sobre-simplificación de poca relevancia para que tenga verdadera utilidad. Esta situación cambió cuando Hohenberg y Kohn presentaron la publicación [25] que sirvió como punto de referencia para una teoría variacional. Ellos demostraron (i) que el potencial externo que experimentan los electrones por la interacción electrón-núcleo está determinado unívocamente por la densidad electrónica y (ii) que el principio variacional está siempre establecido para cualquier densidad. Dado que estos teoremas pueden ser demostrados matemáticamente [23], estableciendo la validez del concepto, pueden ser interpretados como los teoremas fundamentales de la TFD.

La introducción de los funcionales de la densidad en la obtención de la energía electrónica es más clara si se escribe el Hamiltoniano de la ecuación (1.2) como

$$\hat{H} = \hat{T} + \hat{V}_{ee} + \hat{V} \quad (1.17)$$

donde el operador de energía cinética es

$$\hat{T} = \sum_{i=1}^N \left(-\frac{1}{2} \nabla_i^2 \right) \quad (1.18)$$

el operador de repulsión electrón-electrón es

$$\hat{V}_{ee} = \sum_{i<j}^N \frac{1}{r_{ij}} \quad (1.19)$$

y el operador de atracción electrón-núcleo es

$$\hat{V} = \sum_{i=1}^N v(\bar{r}_i) = \int v(\bar{r}) \rho(\bar{r}) d\bar{r} \quad (1.20)$$

Además, el principio variacional establece que la energía del estado basal E_0 satisface

$$E_0 = \min_{\Psi} \langle \Psi | \hat{H} | \Psi \rangle \quad (1.21)$$

donde la minimización es sobre todas las posibles Ψ . La TFD moderna comienza al dividir la minimización de la ecuación (1.21) en dos pasos [24]. Primero, minimizar sobre todas las funciones de onda que producen una determinada densidad, luego sobre todas las densidades. Dado que el operador \hat{V} ya depende únicamente de $\rho(\bar{r})$, es posible definir [23] por separado

$$F[\rho] = \min_{\Psi \rightarrow \rho} \langle \Psi | \hat{T} + \hat{V}_{ee} | \Psi \rangle \quad (1.22)$$

Este es el denominado *funcional universal de la densidad*, llamado así porque es “universal” en el sentido de ser independiente del potencial externo $v(\bar{r})$. Después

$$E_0 = \min_{\rho} \left\{ F[\rho] + \int v(\bar{r})\rho(\bar{r})d\bar{r} \right\} = \min_{\rho} E[\rho] \quad (1.23)$$

donde

$$E[\rho] = F[\rho] + \int v(\bar{r})\rho(\bar{r})d\bar{r} \quad (1.24)$$

Del mismo modo, es posible definir al funcional universal de la densidad en términos de los funcionales exactos de energía cinética y energía potencial de repulsión

$$F[\rho] = \langle \Psi_{\rho}^{min} | \hat{T} + \hat{V}_{ee} | \Psi_{\rho}^{min} \rangle = T[\rho] + V_{ee}[\rho] \quad (1.25)$$

donde Ψ_{ρ}^{min} es la función de onda optimizada que satisface la ecuación (1.21).

La teoría de Hohenberg y Kohn muestra que toda la información que contiene el Hamiltoniano también está contenida en la densidad electrónica, pero no indica cómo obtenerla. Dentro de esta teoría no hay ninguna explicación de cómo es que la energía total depende de la densidad, pues se desconoce la forma de $T[\rho]$ y $V_{ee}[\rho]$; la teoría no puede utilizarse directamente para calcular la energía y requeriría hacer uso de la función de onda.

En 1965, Kohn y Sham [26] desarrollaron un método autoconsistente, basado en los teoremas de Hohenberg y Kohn, que permiten calcular la energía de un sistema al mapear el sistema real de electrones interactuantes en un sistema ficticio de electrones no-interactuantes de la misma densidad electrónica, mediante la introducción de orbitales (orbitales distintos a los de HF). El funcional universal de la densidad se escribe:

$$F[\rho] = T_s[\rho] + J[\rho] + E_{xc}[\rho] \quad (1.26)$$

donde $T_s[\rho]$ es la energía cinética del sistema no-interactuante, $J[\rho]$ es la interacción clásica electrón-electrón y

$$E_{xc}[\rho] = T[\rho] - T_s[\rho] + V_{ee}[\rho] - J[\rho] \quad (1.27)$$

es la llamada energía de intercambio y correlación; contiene la diferencia entre $T[\rho]$ y $T_s[\rho]$ y la parte no clásica de la interacción electrón-electrón de $V_{ee}[\rho]$. La energía electrónica se escribe

$$E[\rho] = T_s[\rho] + \int v(\bar{r})\rho(\bar{r})d\bar{r} + J[\rho] + E_{xc}[\rho] \quad (1.28)$$

1.2. Números de ocupación

La teoría de Kohn y Sham gira en torno al funcional de energía cinética no interactuante $T_s[\rho]$ [23] que puede ser expresado en términos de N orbitales mono-electrónicos. No obstante, es posible construir una generalización a la teoría con la introducción de los números de ocupación n_i para cada estado. De esta manera, la densidad electrónica se define como:

$$\rho(\bar{r}) = \sum_i n_i |\psi(\bar{x}_i)|^2 \quad (1.29)$$

donde la suma abarca todos los estados de una partícula, mientras que los números de ocupación n_i se encuentran acotados en ($0 \leq n_i \leq 1$). El número de orbitales permitidos es arbitrario, no obstante debe satisfacerse que

$$N = \sum_i n_i \quad (1.30)$$

En 1978, Janak [27] demostró además que la derivada de la energía electrónica total con respecto al número de ocupación de un orbital es igual a la energía ϵ de dicho orbital,

$$\frac{\partial E}{\partial n_i} = \epsilon_i \quad (1.31)$$

1.2.1. Números de ocupación fraccionales

Los números de ocupación pueden expresarse con funciones de distribución. Considerando la de Fermi-Dirac, los números de ocupación n_i de la ecuación (1.29) tienen la forma

$$n_i = \{1 + \exp[\theta(\epsilon_i - \mu)]\}^{-1} \quad (1.32)$$

donde ϵ_i es la energía del i -ésimo orbital de Kohn-Sham, μ es el potencial químico y $\theta \equiv 1/k_B T$ es una temperatura ficticia (donde k_B es la constante de Boltzmann, T es la temperatura absoluta y θ es la temperatura medida en unidades de energía).

Incluyendo explícitamente a los números de ocupación fraccionales en $E[\rho]$, se hace necesaria la inclusión de un término adicional E_θ para que la energía total siga siendo variacional [28]. En contraste a la partición original de Kohn-Sham, $F[\rho]$ queda dividido en los términos

$$F[\rho] = T_s[\rho] + J[\rho] + E_{xc}[\rho] - E_\theta[\{n_i\}] \quad (1.33)$$

La forma que adopta este nuevo término E_θ , llamada posteriormente como *energía de termo-correlación*, depende de la forma que adoptan los números de ocupación. Cuando se considera una función de distribución Fermi-Dirac, se tiene que

$$E_\theta[\{n_i\}] = \frac{1}{\theta} \sum_i [n_i \ln n_i + (1 - n_i) \ln(1 - n_i)] \quad (1.34)$$

Esta expresión tiene la misma forma que la contribución de la entropía de Von Neumann a la energía para electrones no interactuantes a la temperatura θ [29],

$$S_\theta[\{n_i\}] = \sum_i [n_i \ln n_i + (1 - n_i) \ln(1 - n_i)] \quad (1.35)$$

por lo que la energía de termo-correlación puede reescribirse como

$$E_\theta[\{n_i\}] = \theta^{-1} S_\theta[\{n_i\}] \quad (1.36)$$

Introducir este funcional $F[\rho]$ en cálculos de TFD tiene el propósito de resolver la correlación estática [30-34] a un mínimo costo computacional, comparado con cálculos de correlación electrónica post-HF.

El formalismo anterior sugiere que la correlación electrónica puede ser expresada en términos de la densidad para un sistema auxiliar con N electrones no-interactuantes sujetos a la temperatura ficticia θ .

1.3. Redes Neuronales

Los modelos de redes neuronales artificiales están inspiradas por la manera en la que funciona el cerebro humano [35]. El cerebro consiste en millones de neuronas, conectadas entre sí, formando una red compleja. Una neurona se comunica con otra enviando una señal eléctrica a través del axón, que es una larga fibra nerviosa que conecta a la segunda neurona por medio de una sinapsis. Cada neurona actúa como un elemento que procesa información porque las señales eléctricas mandadas por una neurona dependen de la fuerza de cada una de las señales de entrada en sus sinapsis.

1.3.1. Inspiración en la biología

Las redes neuronales son un algoritmo de Inteligencia Artificial desarrollado en el siglo XX que fue originalmente motivado para tener máquinas que puedan emular al cerebro con el fin de desarrollar sistemas que aprenden por sí solos.

Las neuronas son células del cerebro formadas, entre otras cosas, por dendritas, cuerpo celular y axón. Si se piensa en las dendritas como “entradas” y en el axón como “salida”, es posible ver a la neurona como una unidad que toma una serie de entradas, las procesa y obtiene una salida que

manda hacia otras neuronas. Este procedimiento se repite para cualquier clase de pensamiento y movimiento que realizamos.

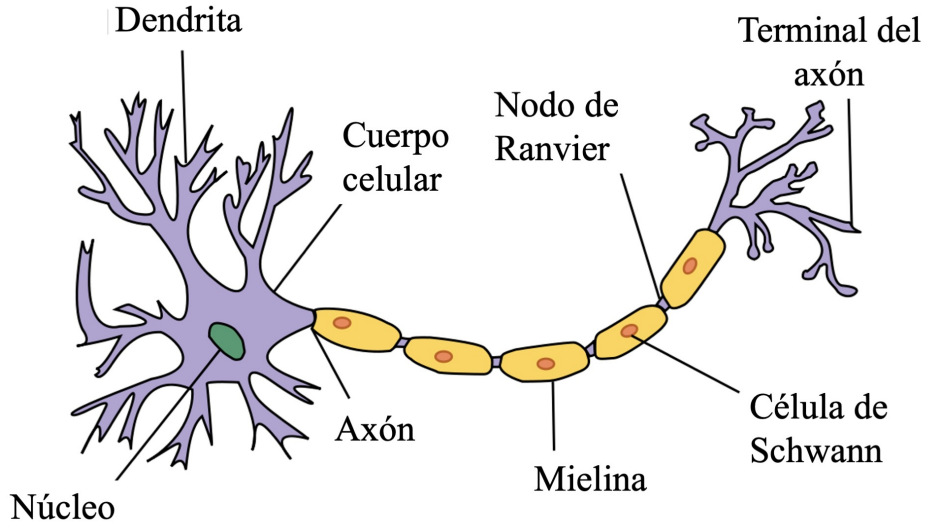


Fig. 1.1: Estructura de una neurona (tomada de <https://es.wikipedia.org/wiki/Neurona>)

Simulando este funcionamiento de manera simplificada, se puede tomar como ejemplo una neurona artificial que recibe información de 3 neuronas en la entrada x y devuelve el resultado (o salida) del procesamiento en $h_{\Theta}(x)$. El procesamiento de la neurona tiene la forma

$$h_{\Theta}(x) = g(\Theta^T x) \quad (1.37)$$

donde $g(z)$ es una función que realiza una transformación en la entrada z y

$$x = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad \Theta = \begin{bmatrix} \Theta_0 \\ \Theta_1 \\ \Theta_2 \\ \Theta_3 \end{bmatrix} \quad (1.38)$$

de modo que el producto interno de la transpuesta de Θ y x tiene la forma

$$\Theta^T x = \Theta_0 x_0 + \Theta_1 x_1 + \Theta_2 x_2 + \Theta_3 x_3 \quad (1.39)$$

Aunque este ejemplo supone que la neurona principal recibe información de tres neuronas, el vector x está conformado por cuatro elementos. Esto es porque se incluye $x_0 = 1$ para que sea posible la inclusión del parámetro Θ_0 en el modelo; este término x_0 tiene el nombre de unidad de “bias” o sesgo y cabe mencionar que no se acostumbra esquemizarla como una neurona más.

Finalmente, es común encontrar en la literatura que los parámetros contenidos en Θ son también referidos como *pesos*. Una red neuronal es un conjunto de unidades de procesamiento como esta.

1.3.2. Estructura

Un ejemplo de estructura de una red neuronal puede ser una con 3 entradas, 3 neuronas y 1 salida, como se muestra en la Figura 1.2. Aunque no está esquematizada, es necesario recordar que va implícita la unidad de sesgo $x_0 = 1$. Las tres neuronas intermedias están denotadas como $a_i^{(2)}$, además de la unidad de sesgo $a_0^{(2)} = 1$. Finalmente, el procesamiento de la neurona $a_1^{(3)}$ da como resultado $h_{\Theta}(x)$.

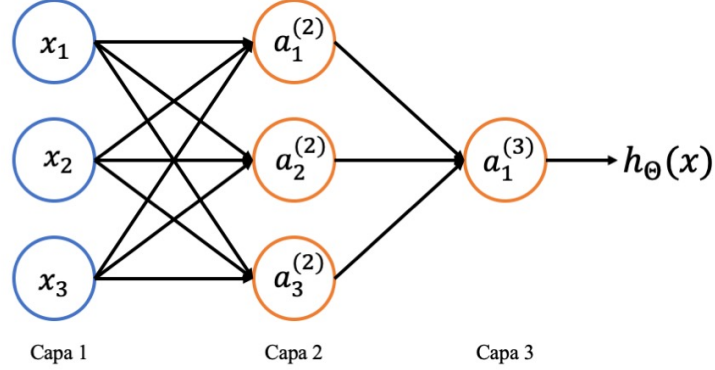


Fig. 1.2: Estructura de una red neuronal con 3 capas; las unidades de azul son entradas y las unidades de anaranjado son de procesamiento

La capa 1 también recibe el nombre de *capa de entrada*, al mismo tiempo que la última capa recibe el nombre de *capa de salida*. La capa 2 de este ejemplo es la *capa escondida*, llamada así porque en el modelo se logran ver únicamente los valores de entrada y los valores de salida, pero no los valores “escondidos” que se calculan en el proceso.

Para analizar lo que ocurre en esta red neuronal se utilizará la notación $a_i^{(j)}$ para la “activación” de la unidad i en la capa j , y la matriz de pesos $\Theta^{(j)}$ que controla la función para pasar de la capa j a la capa $j + 1$.

La manera en la que procesan información las distintas unidades en la capa escondida es

$$\begin{aligned} a_1^{(2)} &= g(\Theta_{10}^{(1)} x_0 + \Theta_{11}^{(1)} x_1 + \Theta_{12}^{(1)} x_2 + \Theta_{13}^{(1)} x_3) \\ a_2^{(2)} &= g(\Theta_{20}^{(1)} x_0 + \Theta_{21}^{(1)} x_1 + \Theta_{22}^{(1)} x_2 + \Theta_{23}^{(1)} x_3) \\ a_3^{(2)} &= g(\Theta_{30}^{(1)} x_0 + \Theta_{31}^{(1)} x_1 + \Theta_{32}^{(1)} x_2 + \Theta_{33}^{(1)} x_3) \end{aligned}$$

La dimensión de $\Theta^{(1)}$ es $\mathbb{R}^{3 \times 4}$. De manera general, si una red neuronal tiene s_j unidades en la capa j y $s_{(j+1)}$ unidades en la capa $j + 1$, la matriz $\Theta^{(j)}$ será de dimensión $s_{(j+1)} \times (s_j + 1)$.

En la capa de salida tenemos la última unidad $a_1^{(3)}$ que procesa la información de la capa 2 con la ecuación

$$h_{\Theta}(x) = a_1^{(3)} = g(\Theta_{10}^{(2)} a_0^{(2)} + \Theta_{11}^{(2)} a_1^{(2)} + \Theta_{12}^{(2)} a_2^{(2)} + \Theta_{13}^{(2)} a_3^{(2)})$$

Si definimos además

$$a^{(1)} = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad a^{(j)} = \begin{bmatrix} a_0^{(j)} \\ a_1^{(j)} \\ a_2^{(j)} \\ a_3^{(j)} \end{bmatrix} \quad (1.40)$$

es fácil comprobar que, de manera general, el procesamiento de cada capa sigue la ecuación

$$a^{(j+1)} = g(\Theta^{(j)} a^{(j)}) \quad (1.41)$$

tomando en cuenta que para toda j , $a_0^{(j)} = 1$. A este proceso de calcular $h_{\Theta}(x)$ se le llama *propagación hacia adelante* [36], porque se comienza desde las entradas y se “propaga” la activación de cada capa escondida hacia la capa de salida.

El objetivo de una red neuronal de este tipo es proporcionar una función $h_{\Theta}(x)$ que “mapea” de la entrada x a la salida y . Para lograrlo es necesario ajustar los pesos $\Theta^{(j)}$ y que, dada una x , se pueda aproximar $h_{\Theta}(x) \approx y$, por medio de un proceso llamado *entrenamiento*.

Cabe mencionar que la red neuronal del ejemplo anterior no es el único arreglo permitido: dependiendo de la aplicación que se le vaya a dar, puede haber distintas maneras de conectar una red neuronal con un número distinto de neuronas o de capas. A la estructura que adquiere una red neuronal también se le conoce como *arquitectura*.

1.3.3. Función de activación

Ya se mencionó que en cada neurona se realiza una activación con la función $g(z)$, donde z era una combinación lineal de pesos y entradas. Esta función es la llamada *función de activación* [36] y su propósito es limitar la amplitud de la señal de salida en cada neurona, obligando así a que todas las neuronas trabajen en un mismo intervalo. Así mismo, con la función de activación se puede reproducir un modelo no-lineal, frecuentemente necesario para obtener la relación entre x y y . La Figura 1.3 muestra algunos ejemplos de funciones de activación.

1.3.4. Función de costo

Para que un modelo de red neuronal pueda operar y entregar buenos resultados, es necesario ajustar los pesos $\Theta^{(j)}$ de cada capa j para que $h_{\Theta}(x) \approx y$. La selección de los valores que conforman estas matrices se realizan con ayuda de la *función de costo*.

La función de costo no es más que la función utilizada para “penalizar” el desempeño de la red cuando $h_{\Theta}(x) \neq y$. Una función de costo $J(\Theta)$ ampliamente utilizada es la función error de mínimos cuadrados. Al igual que en una regresión lineal, los pesos $\Theta^{(j)}$ óptimos se obtienen cuando se minimiza $J(\Theta)$ para un conjunto dado de datos. La función costo de mínimos cuadrados realizada sobre un conjunto de m datos para una red neuronal con un número L de capas está dada por

$$J(\Theta) = \frac{1}{2m} \left[\sum_{i=1}^m \sum_{k=1}^K \left((h_{\Theta}(x^{(i)}))_k - y_k^{(i)} \right)^2 \right] + \lambda \sum_{l=1}^{L-1} \sum_{i=1}^{s_l} \sum_{j=1}^{s_{l+1}} (\Theta_{ji}^{(l)})^2 \quad (1.42)$$

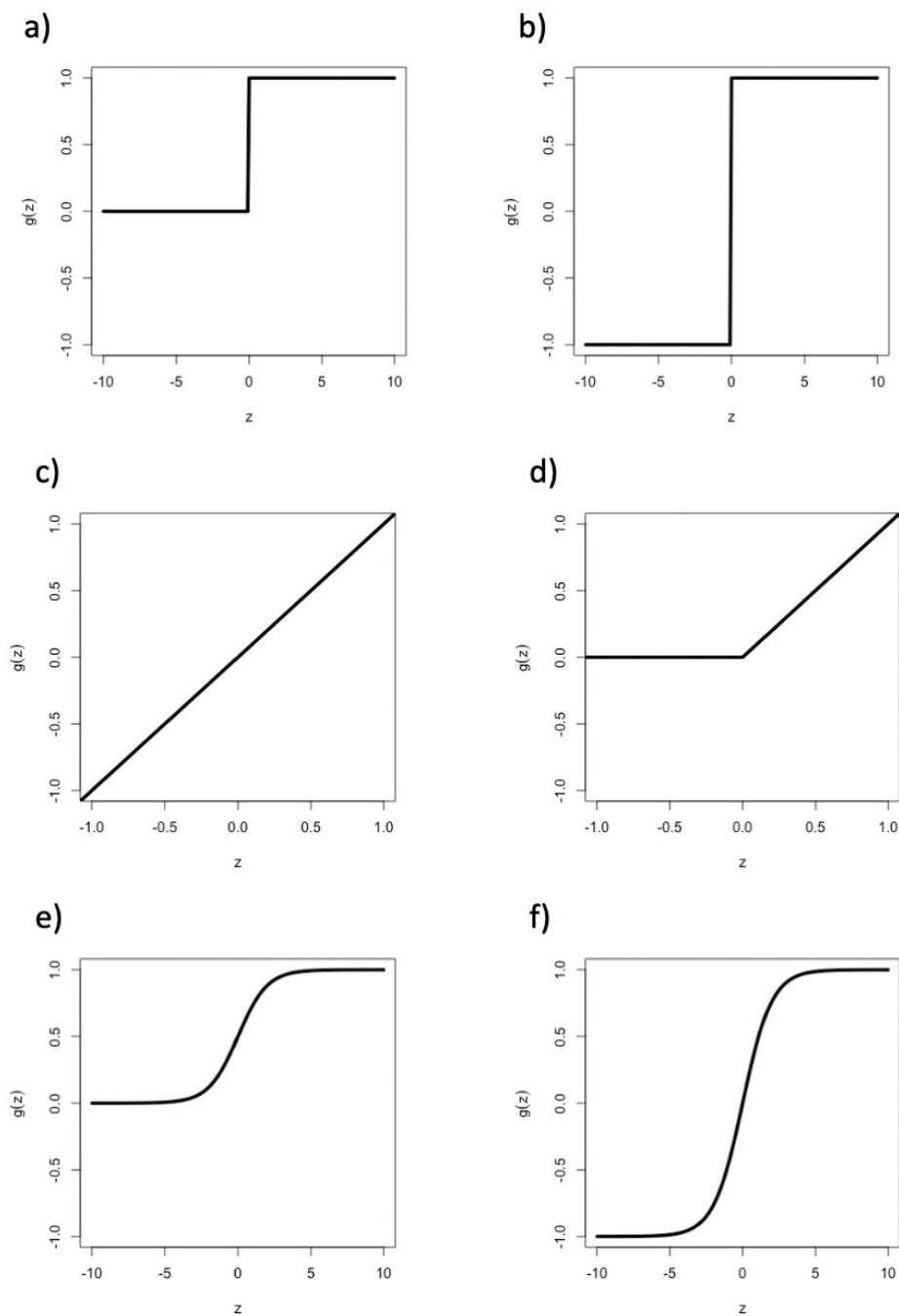


Fig. 1.3: Algunos ejemplos de funciones de activación $g(z)$: a) función escalón, b) función signo, c) función identidad, d) función rectificador (ReLU), e) función sigmoide y f) función tangente hiperbólica

donde s_l es el número de neuronas en la capa l , K es el número de neuronas en la capa de salida y λ es el llamado *parámetro de regularización*. El parámetro de regularización es utilizado para moderar el valor de los pesos en cada capa de salida y evitar tener un modelo sobre-parametrizado o sub-parametrizado (problema de la parametrización descrito en la Sección 1.3.6).

1.3.5. Entrenamiento

En la Figura 1.4 se muestra un ejemplo de $J(\Theta)$ en función de los dos pesos $\Theta_{11}^{(1)}$ y $\Theta_{12}^{(1)}$ para un determinado modelo. Esta gráfica es una simplificación del problema, la función $J(\Theta)$ siempre tiene dimensionalidad alta por la gran cantidad de pesos que se manejan. Debido a que no es posible determinar analíticamente el mínimo global, se utiliza un proceso iterativo. En cada iteración, se calcula el valor de la función de costo al igual que su gradiente, y los pesos son ajustados de acuerdo al algoritmo implementado en la red [37].

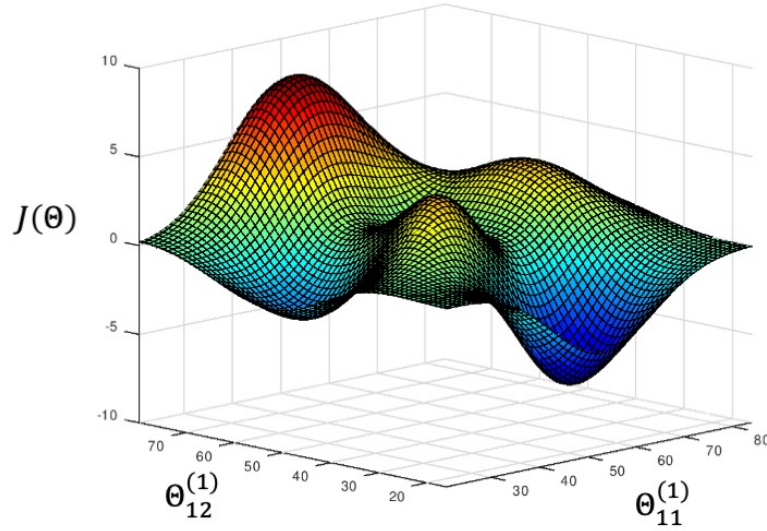


Fig. 1.4: Representación de $J(\Theta)$ como función de los pesos $\Theta_{11}^{(1)}$ y $\Theta_{12}^{(1)}$. Las zonas azules corresponden a $h_{\Theta}(x) \approx y$, mientras que las zonas rojas corresponden a $h_{\Theta}(x)$ bastante alejado de y

Un algoritmo de entrenamiento comúnmente utilizado para calcular los gradientes de la función de costo es el denominado *propagación hacia atrás*. En este trabajo no se presenta la deducción de las expresiones para determinar los gradientes, para más detalles al respecto véase [38] y [39].

El algoritmo consiste en calcular $\delta_j^{(l)}$ como métrica de qué tan responsable es el nodo $a_j^{(l)}$ del error total en el modelo $h_{\Theta}(x) - y$. El error total está dado por

$$\delta^{(L)} = a^{(L)} - y \quad (1.43)$$

para la capa de salida, mientras que para el resto de las capas es

$$\delta^{(l)} = (\Theta^{(l)})^T \delta^{(l+1)} .* g'(\Theta^{(l)} a^{(l)}) \quad (1.44)$$

donde $.*$ es el operador *pointwise* para multiplicar vectores o matrices elemento a elemento y $g'(z)$ es la derivada de la función de activación evaluada en z . El nombre de propagación hacia atrás se debe a que es necesario calcular primero $\delta^{(l+1)}$ para posteriormente calcular “hacia atrás” $\delta^{(l)}$.

El gradiente de la función de costo está dado por

$$\frac{\partial}{\partial \Theta^{(l)}} J(\Theta) = \frac{1}{m} \left(\delta^{(l+1)} (a^{(l)})^T + \lambda \Theta^{(l)} \right) \quad (1.45)$$

El algoritmo de propagación hacia atrás consiste entonces en:

1. Definir una matriz $\Delta^{(l)} = 0$, donde posteriormente se guardarán los gradientes.
2. Para $i = 1$ hasta m :
 - a) Realizar la propagación hacia adelante para calcular $a^{(l)}$
 - b) Calcular $\delta^{(L)} = a^{(L)} - y$
 - c) Calcular $\delta^{(L-1)}, \delta^{(L-2)}, \dots, \delta^{(2)}$
 - d) Actualizar $\Delta^{(l)} := \Delta^{(l)} + \delta^{(l+1)} (a^{(l)})^T$
3. Actualizar los gradientes

$$D_{ij}^{(l)} := \begin{cases} \frac{1}{m} \left(\Delta_{ij}^{(l)} + \lambda \Theta_{ij}^{(l)} \right) & \text{si } j \neq 0 \\ \frac{1}{m} \Delta_{ij}^{(l)} & \text{si } j = 0 \end{cases}$$

donde

$$D_{ij}^{(l)} = \frac{\partial}{\partial \Theta_{ij}^{(l)}} J(\Theta) \quad (1.46)$$

Con el cálculo de los gradientes de $J(\Theta)$, es posible utilizar un algoritmo de optimización. Operacionalmente, es necesario fijar los valores iniciales antes de realizar la optimización de los pesos Θ [40]. No es útil fijar a $\Theta = 0$ para todos los pesos porque la red neuronal adquiere una simetría que resulta en todas las neuronas procesando la misma función y podría ocurrir que se quede estancada en algún punto de silla durante el entrenamiento. Para resolver este problema y romper la simetría de los parámetros iniciales [41], se acostumbra fijarlos a valores aleatorios seleccionados dentro de un intervalo $[-\epsilon, \epsilon]$, evitando además que los pesos tomen valores iniciales muy distintos entre sí.

Uno de los algoritmos más importantes para la optimización de pesos en una red neuronal es el *descenso por el gradiente*, que consiste en minimizar la función $J(\Theta)$ al mover Θ en “pasos pequeños”. La definición del algoritmo aplicada a la función de costo es repetir

$$\Theta_{ij}^{(l)} := \Theta_{ij}^{(l)} - \alpha \frac{\partial}{\partial \Theta_{ij}^{(l)}} J(\Theta) \quad (1.47)$$

hasta converger, donde α es el índice de aprendizaje utilizado para definir el tamaño de los “pasos” en el algoritmo.

Existen variantes del método de descenso por el gradiente más rápidas para optimizar una función [36, 42], como lo son los métodos Adam y RMSprop. Estos algoritmos utilizan estrategias más sofisticadas, a partir de los mismos valores de entrada $J(\Theta)$ y sus derivadas, para obtener Θ .

Por lo general se prefiere utilizar estos algoritmos antes que utilizar el descenso por el gradiente debido a que no necesitan la selección manual del índice de aprendizaje α y tienden a converger más rápido.

En la literatura se utilizan dos términos al momento de reportar entrenamientos: *épocas* y *batch*. El número de épocas es el número de iteraciones realizadas durante el algoritmo de entrenamiento. Al entrenar, se acostumbra agrupar los datos de entrenamiento en “lotes” más pequeños, iterando sobre ellos en cada época para actualizar los pesos en lugar de actualizar sobre todos los datos simultáneamente, provocando que se necesiten menos épocas durante el proceso. Al número de datos que contiene cada lote se le conoce como *batch*.

1.3.6. El problema de la parametrización

Para ajustar los parámetros en Θ y que el modelo pueda predecir satisfactoriamente $h_{\Theta}(x) \approx y$ es necesario resolver el problema de la parametrización, esto es, asegurarse de que el modelo no esté sobre/sub-parametrizado.

Un modelo sobre-parametrizado se caracteriza porque tiene un buen desempeño al predecir datos con los que fue entrenado pero un desempeño pobre con datos que no se utilizaron para su entrenamiento. Esto significa que los parámetros fueron ajustados de manera perfecta con los datos de entrenamiento, pero no es posible generalizar a otros datos.

Un modelo sub-parametrizado se caracteriza por tener un desempeño pobre al predecir cualquier tipo de dato, incluyendo los datos con los que fue entrenado. Esto se puede deber a que los parámetros que tiene el modelo no son suficientes para predecir y y es aconsejable añadir más (agrandar las matrices Θ).

Para poder descartar o identificar estos problemas se utilizan las técnicas llamadas *diagnósticos de aprendizaje*. Para cada una de estas técnicas se utilizan dos tipos de conjuntos de datos: datos de entrenamiento y datos de validación; los primeros son los que se utilizan para ajustar los valores Θ y los últimos son de control.

La primera de estas técnicas es trazar las gráficas $J(\Theta)$ para los datos de entrenamiento y $J(\Theta)$ para datos de validación, ambas en función del número de *épocas* durante el proceso de entrenamiento. Un modelo sobre-parametrizado se caracterizará porque el valor de $J(\Theta)$ de entrenamiento disminuirá conforme aumenta el número de épocas, pero el valor de $J(\Theta)$ de validación aumentaría en algún momento del proceso, dejando en evidencia que los parámetros Θ están siendo ajustados de manera perfecta con los datos de entrenamiento y que no sería posible generalizar a otros datos. Por otra parte, un modelo sub-parametrizado tendría un valor muy alto tanto para $J(\Theta)$ de entrenamiento como para $J(\Theta)$ de validación y el desempeño general del modelo sería pobre.

Otra técnica es la curva de validación o curva de aprendizaje, que consiste en trazar la gráfica $J(\Theta)$ en función de la fracción de datos de entrenamiento utilizada, e.g., si se tienen 1000 datos de entrenamiento, entrenar $m = \{1, 2, 3, \dots, 1000\}$ modelos y trazar $J(\Theta)$ en función de m . Al igual que en la técnica anterior, un modelo sobre-parametrizado tendría un valor pequeño en $J(\Theta)$ de entrenamiento y un modelo sub-parametrizado tendría un valor grande para el mismo. No obstante, la diferencia de esta técnica se encuentra en que un modelo sobre-parametrizado tendría una gran diferencia o “*gap*” entre $J(\Theta)$ de entrenamiento y $J(\Theta)$ de validación, mientras que en un modelo sub-parametrizado ambos valores de $J(\Theta)$ estarían muy cercanos entre sí.

Si se tiene el problema de sobre-parametrización, una posible solución es obtener más datos para entrenar el modelo. Esto haría que el dominio en la curva de validación sea mayor y es posible que se reduzca el *gap* entre $J(\Theta)$ de entrenamiento y $J(\Theta)$ de validación conforme aumenta m . Una posible solución para resolver un problema de sub-parametrización, aparte de añadir más neuronas, es añadir más entradas al vector x , agrandando así las matrices Θ y dándole más flexibilidad al modelo para que $h_{\Theta}(x) \approx y$.

1.3.7. Implementación de una red neuronal

Para entrenar una red neuronal primero se escoge la arquitectura. Esta arquitectura va en función del número de entradas x y el número de salidas y . Basándose en el teorema de Kolmogorov [43], se puede decir que una red neuronal con una única capa escondida que contiene un número finito de neuronas puede aproximar casi cualquier función lineal o no-lineal entre una entrada y una salida si se escogen los pesos apropiadamente [44]. Como regla general, el número de pesos Θ no debe exceder de $m/2$ o se corre el riesgo de que el modelo caiga en una sobre-parametrización.

Con la arquitectura de la red ya definida, la implementación es más directa:

1. Inicializar los pesos aleatoriamente
2. Calcular $h_{\Theta}(x^{(i)})$ para toda $x^{(i)}$ utilizando el procedimiento de propagación hacia delante
3. Calcular $J(\Theta)$
4. Calcular las derivadas $\partial J(\Theta)/\partial \Theta_{ij}^{(l)}$ utilizando el procedimiento de propagación hacia atrás
5. Utilizar una optimización con la propagación hacia atrás para minimizar $J(\Theta)$ como función de los pesos Θ

1.3.8. Aprendizaje máquina en química cuántica

En la actualidad existen gran variedad de métodos de inteligencia artificial, a parte de las redes neuronales, usados en distintas áreas de las ciencias e ingenierías. Como es de esperarse, muchos de estos métodos han encontrado aplicaciones en problemas de química cuántica [45, 46].

Los usos de Inteligencia Artificial en química cuántica pueden ser clasificados en 2 grupos: aquellos que complementan métodos *ab initio* para alcanzar una mayor exactitud, y aquellos que se basan únicamente en métodos aprendizaje máquina. La ventaja del primer grupo es que basarse en un método de química cuántica hace menos dependiente al resultado de la red neuronal, aunque tienden a ser de mayor costo computacional. El segundo grupo depende completamente en que el aprendizaje máquina logre *deducir* las leyes físicas que rigen los fenómenos a estudiar y pierden rigurosidad teórica, pero tienden a ser los de menos costo computacional.

Dentro del primer grupo se encuentra el modelo de Hu et. al., quienes utilizaron una red neuronal para reducir el error de energías de formación calculadas a partir de TFD [47]. Li et. al. utilizaron un método de aprendizaje máquina para predecir energías de absorción de moléculas orgánicas [48]. Hermann et. al. presentaron recientemente una red neuronal para predecir parámetros de la función de onda de tipo Slater-Jastrow como solución a la ecuación de Schrödinger [49]. En el segundo grupo se encuentra por ejemplo el trabajo de Rupp et. al., quienes reportaron un algoritmo de aprendizaje máquina basado únicamente en las coordenadas nucleares relativas de las moléculas para predecir su energía de atomización [50].

Desarrollo Teórico

En la demostración de Weinert y Davenport [28] queda señalado que, para los números de ocupación que obedecen una distribución de Fermi-Dirac, el funcional de energía es idéntico en forma al potencial gran canónico Ω . Siendo $\rho(\bar{r})$ la densidad electrónica de equilibrio dependiente de θ , μ y $v(\bar{r})$, la ecuación variacional del gran potencial se define como

$$\delta\Omega[\rho] = \int \frac{\delta\Omega[\rho]}{\delta\rho(\bar{r})} \delta\rho(\bar{r}) \delta\bar{r} \quad (2.1)$$

Para establecer un esquema de optimización para θ se reescribe de la siguiente forma

$$\delta\Omega[\rho] = \int \left(\frac{\delta\Omega[\rho]}{\delta\rho(\bar{r})} \right)_{\theta} \delta\rho(\bar{r}) \delta\bar{r} + \left(\frac{\partial\Omega[\rho]}{\partial\theta} \right)_{\rho} d\theta \quad (2.2)$$

en donde se ha agregado un término adicional a la ecuación variacional orientado a establecer una condición de equilibrio térmico en el gran potencial, para lo cual se exige que:

$$\left(\frac{\delta\Omega[\rho]}{\delta\theta} \right)_{\rho} = 0 \quad (2.3)$$

Después de resolver la derivada variacional se llega a la siguiente condición de conservación

$$\frac{S_{\theta}[\{n_i\}]}{N} + \sum_i \left(\frac{\partial n_i}{\partial\theta} \right) = C \quad (2.4)$$

donde C es una constante. Desarrollando el segundo término de esta ecuación y haciendo uso de la definición de n_i dada en la ecuación (1.32), se tiene

$$\sum_i \left(\frac{\partial n_i}{\partial\theta} \right)_{\rho} = \sum_i n_i(1 - n_i) \left[\theta \left(\frac{\partial\mu}{\partial\theta} \right)_{\rho} + \mu - \varepsilon_i \right] \quad (2.5)$$

Después de algunas manipulaciones en la ecuación (2.5), puede escribirse como

$$\sum_i \left(\frac{\partial n_i}{\partial \theta} \right)_\rho = \frac{\theta}{N} \sum_i n_i (1 - n_i) [b\mu - \varepsilon_i] \quad (2.6)$$

donde N es el número total de electrones y b es una constante adimensional parametrizada por θ , llamada a partir de ahora *parámetro* b . Sustituyendo la ecuación (2.6) en la ecuación (2.4) y definiendo $\bar{S}_\theta[\{n_i\}]$ como la contribución promedio por electrón a $S_\theta[\{n_i\}]$, se obtiene

$$a + \frac{\theta}{2N} \sum_i n_i (1 - n_i) [b\mu - \varepsilon_i] = \bar{S}_\theta[\{n_i\}] \quad (2.7)$$

donde a es una constante adimensional definida negativa. Resolviendo esta igualdad se obtiene $S_\theta[\{n_i\}]$ que determina E_θ en la ecuación (1.36). A un valor constante a , existe un conjunto infinito de valores para b que pueden resultar en distintos conjuntos $\{n_i\}$; el valor de b que reproduce E_θ se llamará a partir de ahora b_{opt} .

Si se escoge E_θ para que sea igual al valor de E_{corr} , b_{opt} proporciona los números de ocupación $\{n_i\}$ que determinan la energía exacta E y la ecuación (1.14) se puede reescribir

$$E = E_{HF} + E_\theta \quad (2.8)$$

Dicho de otro modo, esto es calcular el valor de θ que resulte en una ocupación de orbitales tal que la energía de termo-correlación reproduzca la energía de correlación.

Este desarrollo deja en evidencia que la ecuación (2.7) es dependiente de los parámetros a , b_{opt} y θ . Encontrar un modo de optimización para estos valores resultaría en una corrección a la energía E_{HF} que proporcione energías electrónicas equivalentes a métodos post-HF.

2.1. Procedimiento iterativo para obtener E_θ

Para obtener una densidad electrónica auto-consistente con los números de ocupación a partir de b se siguen los siguientes pasos: (i) Obtener la función de onda HF; (ii) escoger θ y μ de prueba y calcular n_i con la energía HF ε_i de cada espín-orbital, utilizando la ecuación (1.32); (iii) determinar nueva μ resolviendo (1.30); (iv) determinar una nueva θ resolviendo la ecuación (2.7), utilizando b . Este proceso garantiza la auto-consistencia del conjunto $\{n_i\}$ con el valor de b y, por consiguiente, la auto-consistencia de los valores de θ y μ con la densidad electrónica. La densidad obtenida con $\{n_i\}$ es distinta a la densidad electrónica HF y será tan buena como lo sea el valor de b : si se utiliza el valor b_{opt} , se obtiene la densidad electrónica que reproduce de manera exacta E_{corr} . A este procedimiento se le llamará posteriormente *cálculo de termo-correlación electrónica* (CTCE).

A continuación se presenta el algoritmo para realizar el procedimiento CTCE.

Data: $\{\varepsilon_i\}$, a , b_{opt} , N

Result: E_{corr}

1. Definir función $f_i(\mu, \theta) = \{1 + e^{\theta(\varepsilon_i - \mu)}\}^{-1}$;
2. Definir función $\zeta_f(\mu, \theta) = a + \frac{\theta}{2N} \sum_i f_i(\mu, \theta)(1 - f_i(\mu, \theta))(b_{opt}\mu - \varepsilon_i)$;
3. Definir función $S_f(\mu, \theta) = \sum_i [f_i(\mu, \theta) \ln f_i(\mu, \theta) + (1 - f_i(\mu, \theta)) \ln(1 - f_i(\mu, \theta))]$;
4. Inicializar variables θ y μ ;
5. Definir nuevo valor θ resolviendo $\left\{ \min_{\theta} |F(\mu, \theta)| \right\}$ de la siguiente manera:
 - a) Definir un valor para θ ;
 - b) Definir nuevo valor μ resolviendo $\left\{ \min_{\mu} |N - \sum_i f_i(\mu, \theta)| \right\}$;
 - c) $F(\mu, \theta) = N\zeta_f(\mu, \theta) - S_f(\mu, \theta)$;
6. $S := S_f(\mu, \theta)$;
7. $E_{corr} := \theta^{-1}S$;

return E_{corr}

Algoritmo 1: Procedimiento CTCE

En el Apéndice A se encuentra el código en Python con la implementación del procedimiento CTCE.

2.2. Procedimiento iterativo para obtener b_{opt}

Para obtener b_{opt} a partir de la energía exacta E se siguen los siguientes pasos: (i) Obtener la función de onda HF; (ii) escoger θ y μ de prueba y calcular n_i con la energía HF ε_i de cada espín-orbital, utilizando la ecuación (1.32); (iii) determinar nueva μ resolviendo (1.30); (iv) determinar nueva θ resolviendo la ecuación (2.8), utilizando $S_{\theta}[\{n_i\}]$ de la ecuación (1.35); (v) escoger b de prueba; (vi) determinar nueva b resolviendo la ecuación (2.8), utilizando $S_{\theta}[\{n_i\}]$ de la ecuación (2.7). A este procedimiento se le llamará posteriormente *cálculo de b óptima* (CBO).

A continuación se presenta el algoritmo para realizar el procedimiento CBO.

Data: $\{\varepsilon_i\}$, a , N , E_{corr}

Result: b_{opt}

1. Definir función $f_i(\mu, \theta) = \{1 + e^{\theta(\varepsilon_i - \mu)}\}^{-1}$;
2. Definir función $S_f(\mu, \theta) = \sum_i [f_i(\mu, \theta) \ln f_i(\mu, \theta) + (1 - f_i(\mu, \theta)) \ln(1 - f_i(\mu, \theta))]$;
3. Inicializar variables de prueba θ y μ ;
4. Definir nuevo valor θ resolviendo $\left\{ \min_{\theta} |G(\mu, \theta)| \right\}$ de la siguiente manera:
 - a) Definir un valor para θ ;
 - b) Definir nuevo valor μ resolviendo $\left\{ \min_{\mu} |N - \sum_i f_i(\mu, \theta)| \right\}$;
 - c) $G(\mu, \theta) = E_{corr} - \theta^{-1} S_f(\mu, \theta)$;
5. Definir función $\zeta_f(b) = a + \frac{\theta}{2N} \sum_i f_i(\mu, \theta) (1 - f_i(\mu, \theta)) (b\mu - \varepsilon_i)$;
6. Inicializar variable de prueba b ;
7. Definir nuevo valor b resolviendo $\left\{ \min_b |E_{corr} - \theta^{-1} N \zeta_f(b)| \right\}$;
8. $b_{opt} := b$;

return b_{opt}

Algoritmo 2: Procedimiento CBO

En el Apéndice A se encuentra el código en Python con la implementación del procedimiento CBO.

2.3. Implicaciones

Con el fin de ver claramente la convergencia en $E_{corr} = \theta^{-1} S_{\theta}[\{n_i\}]$, se ha trazado en la Figura 2.1 la superficie de valores θ y μ con respecto al valor $E(\Delta) = |E_{corr, QCI} - \theta^{-1} S_{\theta}[\{n_i\}]|$ para la molécula N_2 en su geometría de equilibrio, donde se ha modificado la ecuación (1.14) para que

$$E_{corr, QCI} = E_{QCI} - E_{HF} \tag{2.9}$$

donde $E_{corr, QCI}$ y E_{QCI} son la energía de correlación y la energía total obtenidas con el nivel de teoría *full* QCISD(T)/6-311+G(3df,2p) y E_{HF} es la energía total obtenida con el nivel de teoría HF/6-311+G(3df,2p), calculados con los programas Gaussian y Gaussview en su paquetería 2009.

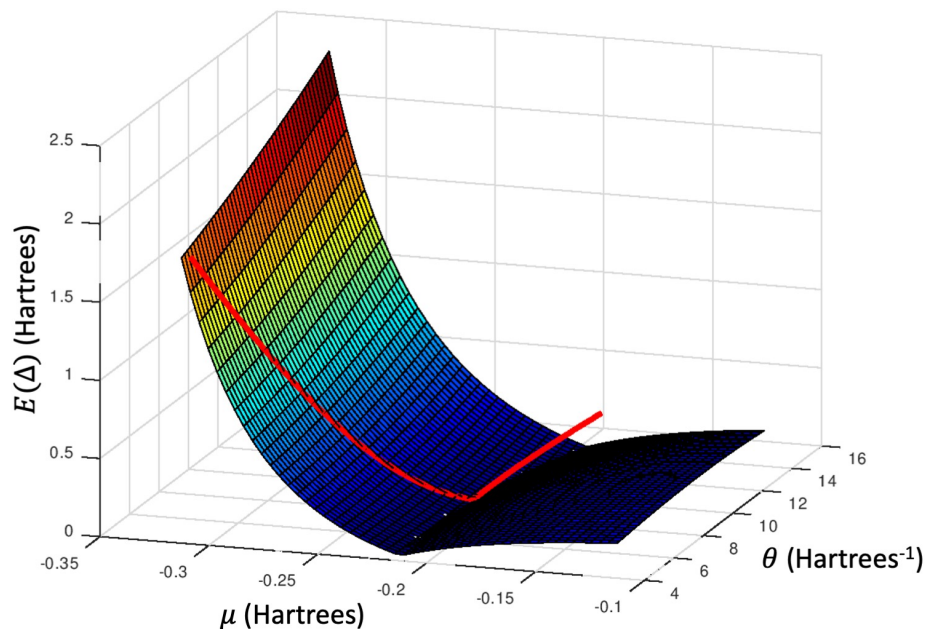


Fig. 2.1: Relación entre la convergencia $E(\Delta)$, potencial químico μ y temperatura θ para la molécula N_2 , en unidades atómicas. La superficie corresponde a un conjunto de valores $\{\mu, \theta\}$ sin restricciones, mientras que la curva en rojo brillante corresponde al conjunto de valores $\{\mu, \theta\}$ que cumplen con la condición de la ecuación (1.30)

Los valores de $E(\Delta)$ se calcularon con los pasos (i) y (ii) del procedimiento CTCE y calculando E_θ con la ecuación (1.36). Se observa que hay un conjunto infinito de valores $\{\mu, \theta\}$ para obtener $E(\Delta) = 0$, pero sólo hay un valor unívoco $\{\mu = -0.215 \text{ Hartrees}, \theta = 9.412 \text{ Hartrees}^{-1}\}$ que además cumple con la condición de la ecuación (1.30).

A partir de los valores $\{\mu, \theta\}$ que satisfacen la condición de la ecuación (1.30), es posible trazar la gráfica de valores b con respecto a $E(\Delta)$ estableciendo un valor de a constante. La Figura 2.2 muestra la relación entre los valores $E(\Delta)$ y b para 4 valores distintos de a , calculados con el procedimiento CTCE. La Figura 2.2 deja en evidencia que al variar a también varía la relación de $E(\Delta)$ y b , e.g., con $a = -0.05$, calcular E_θ con $b = b_{opt} - 0.25$ resulta en $E(\Delta) \approx 0.4$ Hartrees; con $a = -0.20$, calcular E_θ con $b = b_{opt} - 0.25$ resulta en $E(\Delta) \approx 0.02$ Hartrees. A esta propiedad numérica se le llamará posteriormente *sensibilidad*: la sensibilidad de $E(\Delta)$ aumenta al disminuir a .

La gráfica de a como función de b_{opt} verifica que estas variables tienen una dependencia lineal, como se muestra en la Figura 2.3. Las propiedades entre a y b de dependencia lineal y sensibilidad son intrínsecas del modelo matemático y se cumplen para todos los sistemas. No obstante, vale la pena notar que los valores de θ y μ que determinan $E(\Delta) = 0$ son los mismos para cualquier valor de a , por eso el cambio se tiene que ver reflejado en b_{opt} .

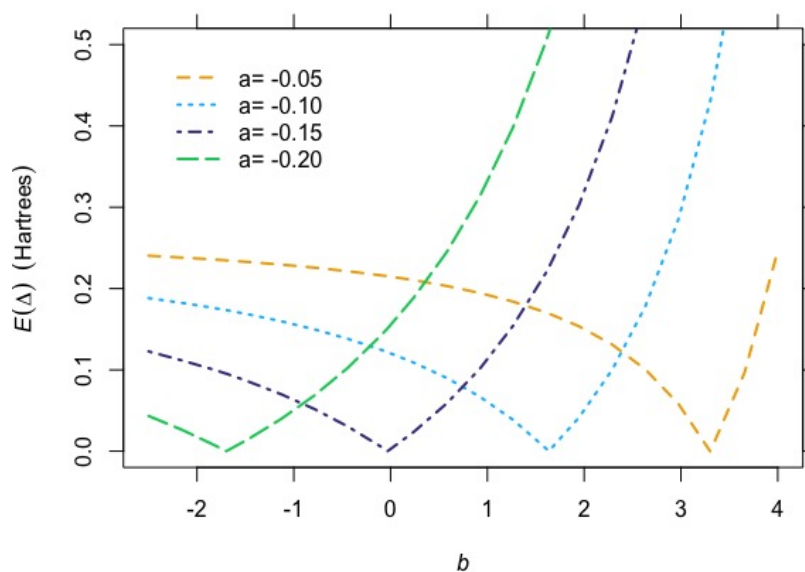


Fig. 2.2: Gráficas de $E(\Delta)$ como función de b para distintos valores de a para la molécula N_2

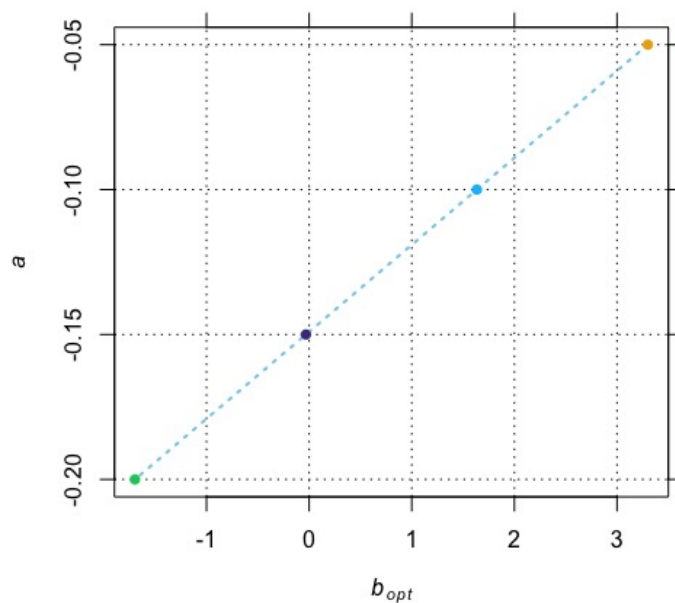


Fig. 2.3: Gráfica de a como función de b_{opt} para la molécula N_2

Como se mencionó anteriormente, la ecuación (2.7) es dependiente de los parámetros a , b_{opt} y θ , i.e., determinar estos valores para cada sistema resultaría equivalente a determinar $E_{corr,QCI}$. Los procedimientos CTCE y CBO demuestran que es posible calcular θ a partir de b_{opt} a un valor de

a constante, reduciendo el problema únicamente a determinar b_{opt} . Es conveniente además que la determinación de b_{opt} sea a partir de propiedades provenientes de la función de onda HF. De esta manera, el costo determinate para obtener la energía electrónica mediante esta metodología lo constituiría el cálculo de la función de onda HF.

Este trabajo tiene el propósito de diseñar y calibrar una red neuronal artificial que proporcione el valor de b_{opt} para sistemas moleculares orgánicos, a partir de información contenida en la función de onda HF.

3.1. Objetivo general

- Calibración de una red neuronal orientada a la predicción de la energía de correlación electrónica para moléculas orgánicas.

3.2. Objetivos particulares

- Generación de una base de datos de energías de correlación para distintos sistemas moleculares orgánicos al nivel de teoría *full* QCISD(T).
- Obtención de b_{opt} para el conjunto de datos. Los datos de entrada para predecir b con la red neuronal serán parámetros obtenidos a partir de la función de onda HF.
- Entrenamiento de una red neuronal para cálculos de estructura electrónica en moléculas pequeñas y medianas empleando la base 6-311+G(3df,2p), utilizando sistemas del conjunto obtenido, hasta obtener un porcentaje de error total no mayor al 15 % para la predicción de la energía de correlación.
- Entrenamiento de una red neuronal para cálculos de estructura electrónica en moléculas grandes empleando la base 6-31+G(d,p), utilizando sistemas del conjunto obtenido, hasta obtener un porcentaje de error total no mayor al 15 % para la predicción de la energía de correlación.

Metodología

El modelo construido en este trabajo tiene el propósito de predecir b_{opt} en moléculas de diversos tamaños. Por este motivo se utilizarán 2 redes neuronales: la red neuronal *Thermal-Triple* (TT), empleada para moléculas pequeñas y medianas calculadas con la base de valencia triple zeta 6-311+G(3df,2p); y la red neuronal *Thermal-Double* (TD), empleada para moléculas grandes calculadas con la base de valencia doble zeta 6-31+G(d,p). Con el fin de emplear la misma arquitectura en ambas redes neuronales, la calibración de los hiperparámetros (tipo de funciones de activación, número de neuronas, número de épocas de entrenamiento, etc) se realizará únicamente en la red neuronal TT. En este Capítulo se presenta la metodología empleada para dicha calibración.

4.1. Detalles computacionales

Para los cálculos de estructura electrónica HF y QCISD(T) se utilizaron los programas Gaussian y Gaussview en su paquetería 2009 [51]. Las herramientas utilizadas para el procesamiento de datos fueron creadas con el lenguaje de programación Python (versión 3.7.1) con las bibliotecas de Numpy (versión 1.18.1) para el manejo de los operadores numéricos, Scipy (versión 1.3.0) para las minimizaciones en los procedimientos CBO/CTCE y Keras (versión 2.3.1) para la creación de la red neuronal.

4.2. Preparación de la base de datos

Antes de escoger una arquitectura para la red, es necesario contar con los valores de entrada y de salida. La salida la definimos como b_{opt} calculada con el procedimiento CBO y la entrada como un conjunto de propiedades obtenidas de la función de onda HF.

Muchos enfoques de Inteligencia Artificial dependen de una base de datos grande para ser entrenada, pero este no puede ser el caso de b_{opt} . La limitante principal en este modelo es la energía E_{QCI} , pues es necesario realizar un cálculo al nivel de teoría *full* QCISD(T)/6-311+G(3df,2p) para obtenerla, siguiendo la definición utilizada en el Capítulo 2. El costo computacional de este método es alto (varios días por cálculo) y se buscó desde el principio realizarlo únicamente para moléculas que no tengan más de 100 electrones.

Una práctica recomendable al desarrollar algoritmos de Inteligencia Artificial es implementar al principio un modelo rápido y sencillo que sirva como base para después implementar un modelo

más complejo. Con esto en mente, el primer algoritmo desarrollado para el presente trabajo se entrenó con 55 sistemas: 18 átomos y 37 moléculas. Utilizando como retroalimentación a los sistemas que más error presentaban, se construyó gradualmente una base de datos más robusta. El conjunto de datos final tuvo un total de 179 sistemas, que se decidió dividir en 23 subconjuntos en función de su naturaleza:

1. **A0** contiene 17 átomos neutros en su estado de mínima energía. Elementos presentes: Al, Ar, As, B, Be, Br, C, Ca, Cl, F, He, Kr, Mg, N, Ne, O y P.

2. **AGD** contiene las moléculas

- Biciclobutano
- Cubano
- Tetraedrano

Elementos presentes: C y H.

3. **AM** contiene 6 átomos neutros con multiplicidades distintas a las de mínima energía. Elementos presentes: As (doblete), C (singulete), O (singulete), P (doblete), S (singulete) y Se (singulete).

4. **AP** contiene 5 átomos con 1 carga positiva en su estado de mínima energía. Elementos presentes: C, N, O, P y S.

5. **CMOL** contiene las moléculas

- Benceno
- 1,3-Butadieno
- Etano
- Eteno
- Etino
- H₂
- Metano
- Propano
- Propeno
- Propino

Elementos presentes: C y H.

6. **CYCP** contiene las moléculas

- Carbo-catión bencilo
- Carbo-catión terbutilo
- Carbo-catión (H₃C-CH-CH₂-CH₃)⁺
- Ciclobutano

- Ciclohexano conformación de silla
- Ciclohexano conformación de bote
- Ciclopentano
- Ciclopropano

Elementos presentes: C y H.

7. **DS1** contiene 6 moléculas diatómicas con distancia de enlace distinta a la de mínima energía, i.e., distancias de enlace seleccionadas empíricamente para mejorar el desempeño de la red. La corrección en las excitaciones triples presentó una sobrestimación en el límite de la disociación infinita para C_2 , por lo que se tomó el valor QCISD para el cálculo de b_{opt} . Elementos presentes: C, H y He.
8. **DS2** contiene 9 moléculas diatómicas con distancia de enlace distinta a la de mínima energía, i.e., distancias de enlace en las que el perfil de disociación E_{HF} presenta anomalías con respecto a E_{QCI} . La corrección en las excitaciones triples presentó una sobrestimación en el límite de la disociación infinita para C_2 y F_2 , por lo que se tomó el valor QCISD para el cálculo de b_{opt} . Elementos presentes: Be, C, F, H y He.
9. **EM** contiene 2 moléculas C_2 a distancias de enlace donde se observó una anomalía en los perfiles de disociación HF singlete y triplete. Elementos presentes: C.
10. **H2OC** contiene 4 cúmulos de H_2O . Elementos presentes: H y O.
11. **H2SC** contiene 5 cúmulos de H_2S , cuyas geometrías de mínima energía se obtuvieron de [52]. Elementos presentes: H y S.
12. **HALO** contiene las moléculas
 - CCl_4
 - $CHCl_3$
 - HCl
 - HFElementos presentes: C, Cl, F, y H.
13. **HB** contiene los sistemas moleculares
 - $H_2O \cdots HCH_3$
 - $HOH \cdots NH_3$Elementos presentes: C, H, N y O.
14. **HETC** contiene las moléculas
 - Pirrolidina
 - Piperidina
 - 2,2-dimetiltiociclobutano
 - 3-etilazaciclobutano

- 2-fluorotiociclopropano
- Furano
- Indol
- 3-metiloxaciclohexano
- N-metilazaciclopropano
- Oxaciclobutano
- Oxaciclopropano
- Pirrol
- Quinoleína
- Tiociclopentano
- Tiofeno
- 2-indanona
- Óxido de indeno

Elementos presentes: C, H, F, N, O y S.

15. **HETM** contiene las moléculas

- Dimetilamina
- Metilfosfina
- Dimetilfosfina
- Ácido metilfosfónico
- Ácido metilsulfónico
- Dietilamina
- H_3O^+
- H_3PO_4
- H_3S^+
- PH_3
- PH_4^+

Elementos presentes: C, H, P, N, O y S.

16. **NCOV** contiene 24 sistemas moleculares neutros con distintas interacciones no-covalentes, cuyas geometrías de mínima energía se obtuvieron de [53]. Elementos presentes: C, Cl, H, F, N, Ne, O y S.

17. **NH3C** contiene 4 cúmulos de NH_3 , cuyas geometrías de mínima energía se obtuvieron de [54]. Elementos presentes: H y N.

18. **NMOL** contiene las moléculas

- Etilamina
- HNO_3
- Metilamina
- N_2
- NH_3
- NH_4^+
- Piridina

Elementos presentes: C, H, N y O.

19. **OMOL** contiene las moléculas

- Ácido acético
- Acetona
- Dimetil éter
- Etanol
- Formaldehído
- Ácido Fórmico
- H_2O
- H_2O_2
- Metanol
- O_2

Elementos presentes: C, H y O.

20. **RAD** contiene las moléculas

- Radical bencilo
- Radical terbutilo
- Radical $\text{H}_3\text{C}-\text{CH}-\text{CH}_2-\text{CH}_3\bullet$
- Radical $\text{CH}_3\bullet$
- Radical $\text{OH}\bullet$
- Radical $\text{PH}_2\bullet$
- Radical $\text{SH}\bullet$

Elementos presentes: C, H, O, P y S.

21. **RESO** contiene las moléculas

- Propionitrilo
- 3-buten-2-ona
- Ciclohexeno
- Ciclopentadieno
- Ciclopenteno
- Anhídrido maleico
- Tetracianoetileno

Elementos presentes: C, H, O y N.

22. **ROHC** contiene los sistemas moleculares

- $\text{OH}_2 \cdots \text{OHCH}_3$
- $\text{CH}_3\text{OH} \cdots \text{OH}_2$
- $(\text{CH}_3\text{OH})_2$
- $(\text{CH}_3\text{OH})_3$
- $\text{CH}_3\text{OH} \cdot (\text{H}_2\text{O})_2$
- $\text{H}_2\text{O} \cdot (\text{CH}_3\text{OH})_3$

Elementos presentes: C, H y O.

23. **SMOL** contiene las moléculas

- Dimetilsulfuro
- Metanotiol
- H_2SO_4
- SH_2
- SO_2

Elementos presentes: C, H, O y S.

Se pueden encontrar más detalles sobre los subconjuntos en el Apéndice B. Todas las geometrías moleculares fueron obtenidas de la optimización a nivel de teoría HF/6-311+G(3df,2p), excepto en los casos donde se indicó lo contrario.

Se calculó el valor b_{opt} para cada sistema, a 3 valores distintos de $a = \{-0.05, -0.10, -0.15\}$, siguiendo el procedimiento CBO; los valores se muestran en la Figura 4.1.

Tendencias interesantes a analizar:

- Los sistemas que tienen carga positiva son los que presentan un valor de b_{opt} menor al del resto dentro de su subconjunto, e.g, la molécula NH_4^+ (índice 143) tiene una b_{opt} menor que el resto de **NMOL** a cualquier valor de a .

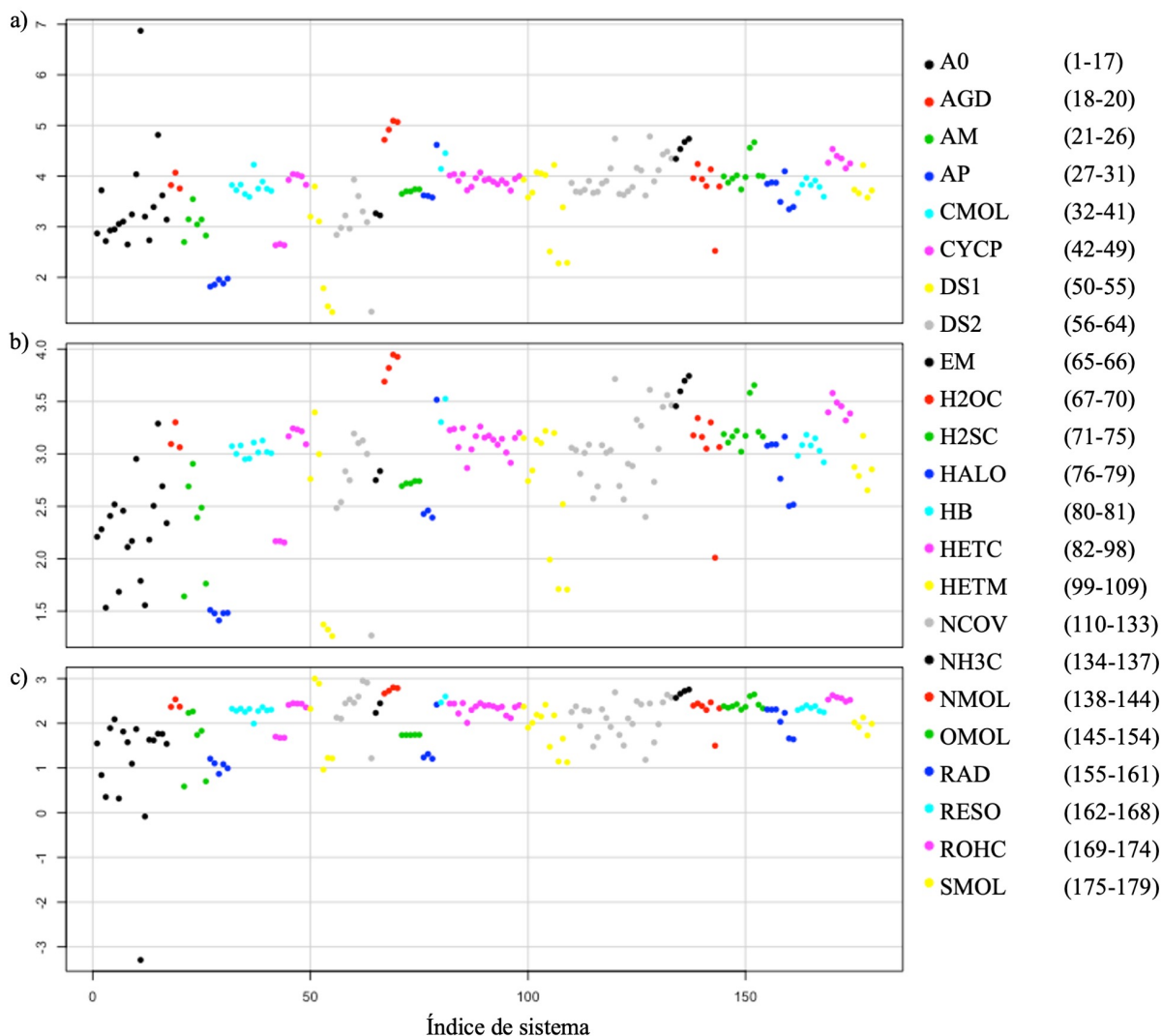


Fig. 4.1: Valores de b_{opt} (en u.a.) a distintos valores de a para los 23 subconjuntos ordenados por índice de sistema: a) -0.05 , b) -0.10 , c) -0.15

- A pesar de ser sistemas distintos, los subconjuntos que contienen cúmulos de la misma naturaleza presentan una tendencia en b_{opt} , e.g., hay una tendencia casi lineal en b_{opt} como función del número de moléculas en los cúmulos de **H2OC**, **H2SC** y **NH3C**.
- La pendiente más pronunciada para b_{opt} como función de a se observa en el átomo de He (subconjunto **A0**, índice 11).
- Los valores de b_{opt} tienden a agruparse en función de los elementos químicos presentes en los sistemas, e.g., los valores de b_{opt} para **AGD** son muy similares a los de **CMOL** a cualquier valor de a , ambos subconjuntos conformados únicamente por C y H.

Como se mencionó en el Capítulo 2, b_{opt} tiene una relación lineal con a para cada sistema. Esto se ve reflejado en la dispersión de los valores b_{opt} para distintas a , i.e., el intervalo de valores $b_{opt} = [1.31, 6.87]$ para $a = -0.05$, mientras que $b_{opt} = [-3.29, 2.99]$ para $a = -0.15$. Es de esperar que la red neuronal se desempeñe de mejor manera cuando el entrenamiento se da con datos que

estén distribuidos de manera similar a los valores de entrada. Esto es, conviene escoger un valor de a cuya distribución sea similar a la distribución de los valores de entrada.

4.3. Entradas

Uno de los factores más importantes al momento de entrenar un algoritmo de Inteligencia Artificial es la selección de los datos de entrada, lo que hace necesario saber discriminar las Variables Significativas (VS) que condicionan un resultado específico.

A partir de la función de onda HF es posible obtener una gran cantidad de variables de entrada con las que se puede generar la red. Dentro de las propiedades obtenidas directamente del operador de Fock (modificación del operador Hamiltoniano, utilizado en el método HF) se tiene la siguiente lista:

- Energía cinética E_T
- Energía de atracción electrón-núcleo E_V
- Energía del término coulombico E_J
- Energía del término de intercambio E_K
- Energía de repulsión núcleo-núcleo E_{Nuc}
- Energía de los espín-orbitales ε_i
- Energía total E_{HF}

Siendo θ un parámetro energético equivalente a la beta termodinámica $\beta \equiv 1/k_B T$ de la mecánica estadística, es plausible establecer que θ está relacionada con la energía cinética de las partículas, dentro del marco de gas ideal. Esto sugiere que E_T forma parte del conjunto VS que condiciona θ y, al mismo tiempo, a b_{opt} .

La propiedad E_V considera la suma de las atracciones electrónicas con los núcleos y no considera en ningún momento dependencia entre electrones, lo que sugiere que no aporta información para determinar la correlación electrónica y que no forma parte de VS.

Se mencionó en la Sección 1.1.2 que para resolver el problema de la correlación electrónica es necesario considerar los efectos de repulsión Coulombica instantánea y de degeneración de estados electrónicos. El primero no está contenido en E_J porque no considera la repulsión instantánea, pero hay cierta información de esto en E_K con el agujero de Fermi. La segunda puede contenerse en la información de ε_i , en específico de los orbitales HOMO y LUMO. Esto sugiere que E_K o $(E_J + E_K)$ y $\{\varepsilon_{HOMO}, \varepsilon_{LUMO}\}$ forman parte de VS.

Como el valor de E_{Nuc} es igual a cero para los sistemas que son átomos aislados pero no se observa esta tendencia para sus valores b_{opt} ni θ , además de que la repulsión nuclear no considera a los electrones, E_{Nuc} no forma parte de VS.

Dado que las propiedades obtenidas del operador de Fock dependen del tamaño del sistema, es más apropiado considerar el valor promedio sobre el número de electrones. En la Figura 4.2 se muestran las cinco posibles VS \bar{E}_T , $(\bar{E}_J + \bar{E}_K)$, \bar{E}_K , ε_{HOMO} y ε_{LUMO} para los 23 subconjuntos, donde la barra superior indica que la propiedad fue dividida entre los N electrones del sistema.

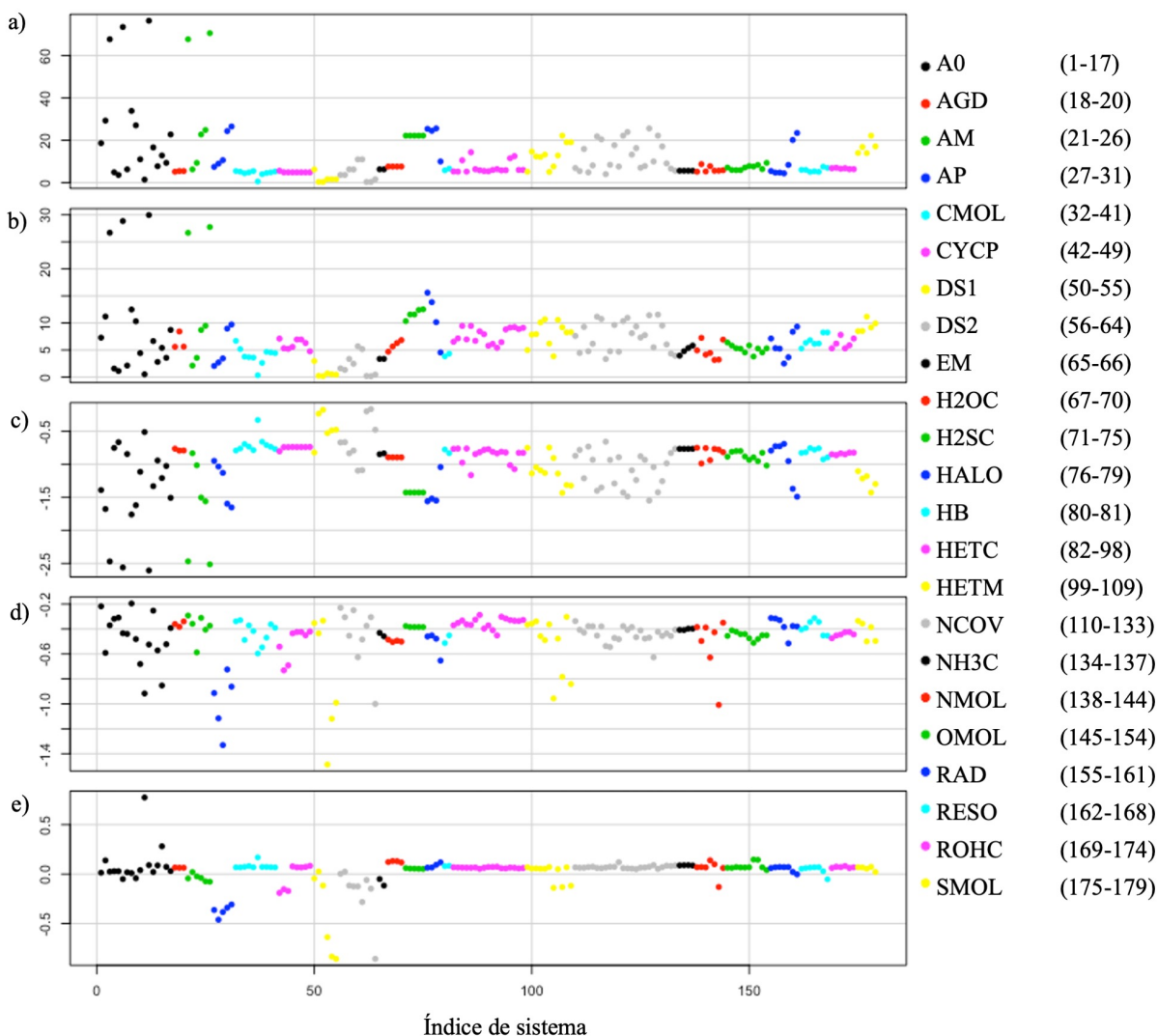


Fig. 4.2: Propiedades calculadas con el operador de Fock (en u.a.) para los 23 subconjuntos ordenados por índice de sistema: a) \bar{E}_T , b) $(\bar{E}_J + \bar{E}_K)$, c) \bar{E}_K , d) ε_{HOMO} y e) ε_{LUMO}

Tendencias interesantes a analizar:

- La energía cinética \bar{E}_T varía en función del potencial nuclear, e.g., los mayores valores de \bar{E}_T corresponden a sistemas con elementos de número atómico grande y a especies con carga positiva en los subconjuntos que corresponda.
- Los subconjuntos que contienen cúmulos de la misma naturaleza presentan una tendencia en $(\bar{E}_J + \bar{E}_K)$, e.g., hay una tendencia casi lineal en $(\bar{E}_J + \bar{E}_K)$ como función del número de moléculas en los cúmulos de **H2OC**, **H2SC** y **NH3C**.
- Los sistemas de naturaleza similar presentan valores similares para las propiedades \bar{E}_T y \bar{E}_K .
- Casi todos los sistemas tienen una variación muy pequeña en sus valores ε_{LUMO} . Los sistemas con el valor más alejado de la media son el átomo de He (subconjunto **A0**, índice

11), los sistemas con carga positiva y las moléculas que no se encuentran en su geometría de mínima energía (subconjuntos **DS1** y **DS2**).

- Los valores ε_{HOMO} varían más que los valores ε_{LUMO} , pero siguen la tendencia de tener valores más alejados de la media en sistemas con carga positiva o con geometría distinta a la de mínima energía.

Las tendencias en estas propiedades siguen un patrón similar a las tendencias observadas en los valores b_{opt} , lo que sugiere que es posible obtener de ellas información sobre la correlación electrónica y que forman parte de VS para calcular b_{opt} .

Es evidente que los valores de las propiedades seleccionadas en VS son de distinto orden de magnitud en los subconjuntos seleccionados, e.g., mientras ε_{LUMO} toma valores no mayores a 1 u.a., \bar{E}_T toma valores alrededor de 10 u.a. El procedimiento para escalar variables resuelve este problema. El escalamiento de variables consiste en realizar para cada propiedad p

$$p_S = \frac{p - \bar{P}}{\sigma(P)} \quad (4.1)$$

donde p_S es el valor escalado de la propiedad p en el sistema i , \bar{P} es la media de la propiedad p en todos los m sistemas de entrenamiento y $\sigma(P)$ es la desviación estándar de la propiedad p en todos los m sistemas de entrenamiento. La media del nuevo valor escalado p_S calculada con todos los m sistemas es 0 y la desviación estándar es 1, lo que asegura que todos los valores escalados de las propiedades en VS tengan el mismo orden de magnitud y el entrenamiento de la red sea eficiente. Por ejemplo, el vector de entrada x formado por \bar{E}_T , $(\bar{E}_J + \bar{E}_K)$, \bar{E}_K , ε_{HOMO} y ε_{LUMO} para la molécula N_2

$$x = \begin{bmatrix} 7.78 \\ 4.45 \\ -0.94 \\ -0.63 \\ 0.14 \end{bmatrix} \quad (4.2)$$

se transforma en el vector escalado x_S

$$x_S = \begin{bmatrix} -0.30 \\ -0.51 \\ 0.14 \\ -0.85 \\ 0.73 \end{bmatrix} \quad (4.3)$$

si se utilizan todos los 179 sistemas para la media y desviación estándar en el proceso de escalamiento.

Antes de realizar el entrenamiento de la red neuronal, es necesario establecer un valor de a . Se escogió de manera artesanal $a = -0.140$, que demostró recuperar las tendencias de manera más óptima en la red neuronal (en la siguiente sección se describe cómo se encontró este valor). En la Figura 4.3 se muestran los valores de b_{opt} con este valor de a para los 23 subconjuntos.

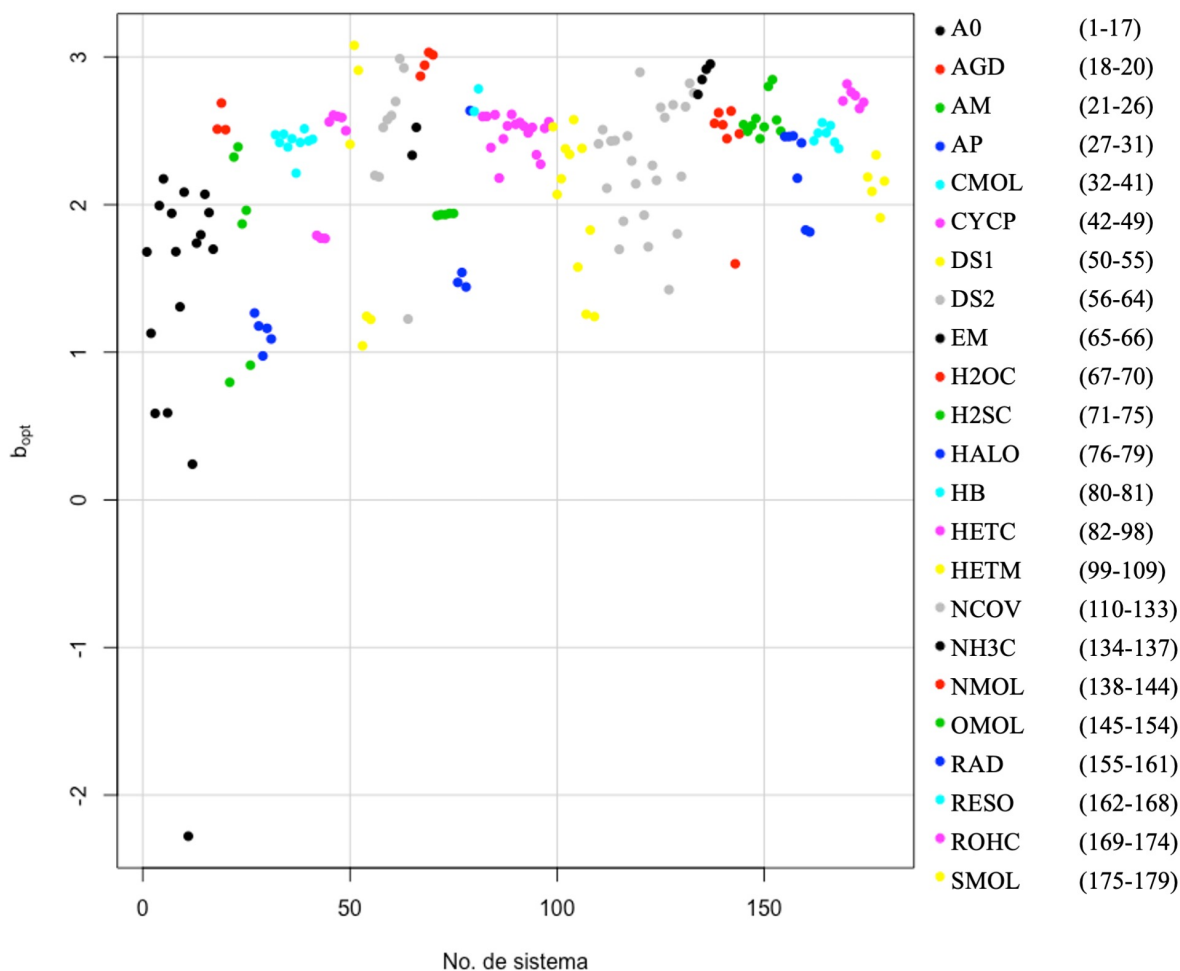


Fig. 4.3: Valores de b_{opt} (en u.a.) para los 23 subconjuntos ordenados por índice de sistema con $a = -0.140$

4.4. Entrenamiento

La red neuronal se entrenó con el conjunto de datos de 179 sistemas que se dividió en conjuntos de entrenamiento (70 %, 125 sistemas), validación (15 %, 27 sistemas) y prueba (15 %, 27 sistemas). Esta división fue de manera aleatoria con la única condición de mantener los sistemas de valores frontera en el conjunto de entrenamiento para que el procedimiento de escalamiento abarque a todos los sistemas, e.g., el conjunto de entrenamiento contiene a los sistemas con mínima y máxima \bar{E}_T , con mínima y máxima \bar{E}_K , etc. En la Figura 4.4 se muestra la distribución de estos conjuntos. El ajuste de los pesos Θ se realiza tomando en cuenta únicamente al conjunto de entrenamiento; los conjuntos de validación y prueba no participan en este proceso. Se realiza esta división de conjuntos porque se acostumbra seleccionar el modelo que proporcione el menor error posible en el conjunto de validación y, para no caer en una generalización errónea, se acostumbra reportar como el error generalizado al error calculado sobre el conjunto de prueba.

Para entrenar la red se utilizó la función costo $J(\Theta)$ de mínimos cuadrados y el algoritmo NAdam

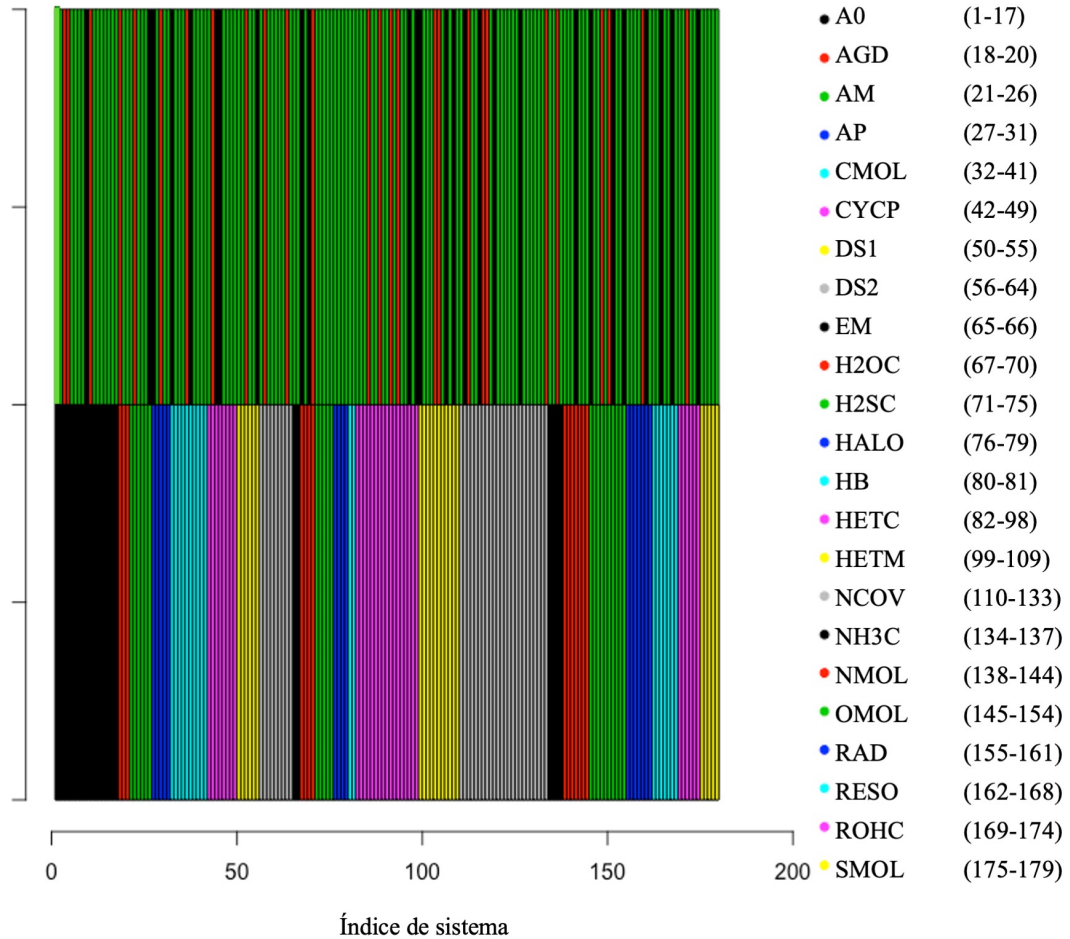


Fig. 4.4: Conjuntos para entrenar la red neuronal: Parte superior: conjuntos de entrenamiento (verde), validación (negro) y prueba (rojo); parte inferior: 23 subconjuntos

en la minimización de Θ . El ajuste de hiper parámetros se hizo de manera artesanal, haciendo varias iteraciones de entrenamiento mientras se hacían variaciones en los hiper parámetros.

Todos los modelos probados tenían en común:

- La misma arquitectura de 1 capa de entrada, 1 capa escondida y 1 capa de salida.
- El número de pesos Θ no mayor a $m/2 = 62$.

Las variaciones que se realizaron en los modelos fueron:

- Valor de a para calcular b_{opt}
- Número de entradas, haciendo distintas combinaciones con el conjunto de VS establecidas
- Número de neuronas en la capa escondida

- Funciones de activación para $a^{(2)}$ y $a^{(3)}$
- Número de épocas
- Tamaño de lote (*batch*)

Se construyeron 432 modelos con distintos valores de a (desde -0.05 hasta -0.25) y distintas combinaciones de los hiperparámetros: el desempeño promedio de los modelos analizados fue mejor con $a = -0.140$. Para ese valor de a , se realizaron además 28 modelos haciendo combinaciones más finas de las variaciones ya mencionadas.

En la Figura 4.5 se muestra un ejemplo de las 179 predicciones de b (llamada a partir de ahora b_{rn}) con distintos modelos. El mejor conjunto de entradas e hiper parámetros que se encontró es el que se muestra en la Tabla 4.1.

Tabla 4.1: Hiper parámetros utilizados para entrenar el modelo 84 de la red neuronal

Hiper parámetro	Descripción	Valor
n	Dimensión del vector de entrada	4
s	Neuronas en la capa escondida	8
<i>batch</i>	Tamaño de lote	10
<i>epoch</i>	Épocas	1000
$a^{(2)}$	Activación para la capa escondida	Sigmoide
$a^{(3)}$	Activación para la capa de salida	SELU

Las entradas para el mejor modelo de red neuronal fueron \bar{E}_T , \bar{E}_K , ε_{HOMO} y ε_{LUMO} . En la Tabla 4.2 se muestra el espacio de trabajo, que es el rango de valores con los que trabaja la red de manera confiable. Este espacio está dado únicamente por el conjunto de entrenamiento, no es confiable predecir b_{opt} de un sistema que se encuentra fuera de este rango de operación. Así mismo, el espacio de trabajo correspondiente a estos mismos sistemas pero calculados con la base 6-31+G(d,p), para entrenar la red neuronal TD, se muestra en la Tabla 4.3.

Tabla 4.2: Rango de valores de entrada para cada conjunto de entrenamiento, calculados con HF/6-311+G(3df,2p)

Entrada (u.a.)	Entrenamiento	Validación	Prueba
E_T/N	[0.330, 76.432]	[0.565, 70.566]	[0.333, 67.690]
E_K/N	[-2.607, -0.165]	[-2.514, -0.330]	[-2.470, -0.177]
ε_{HOMO}	[-1.486, -0.195]	[-1.119, -0.303]	[-1.330, -0.288]
ε_{LUMO}	[-0.858, 0.774]	[-0.834, 0.169]	[-0.857, 0.117]

Tabla 4.3: Rango de valores de entrada para cada conjunto de entrenamiento, calculados con HF/6-31+G(d,p)

Entrada (u.a.)	Entrenamiento	Validación	Prueba
E_T/N	[0.355, 76.076]	[0.567, 70.349]	[0.372, 67.457]
E_K/N	[-2.598, -0.176]	[-2.508, -0.329]	[-2.464, -0.186]
ε_{HOMO}	[-1.486, -0.195]	[-1.119, -0.301]	[-1.330, -0.287]
ε_{LUMO}	[-0.858, 1.399]	[-0.839, 0.241]	[-0.858, 0.123]

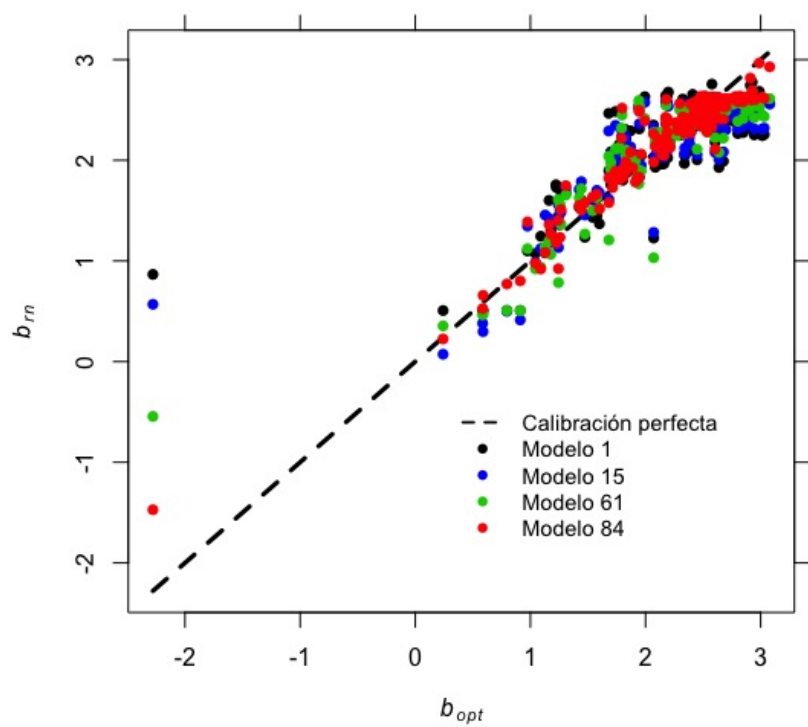


Fig. 4.5: Calibración de los hiper parámetros, siendo el modelo número 84 la mejor combinación encontrada

Resultados y Discusión

5.1. Resultados del entrenamiento

Retomando lo mencionado en el capítulo anterior, se construyeron 2 redes neuronales: la red neuronal TT, entrenada con los 179 sistemas calculados con la base de valencia triple zeta 6-311+G(3df,2p); y la red neuronal TD, entrenada con los mismos 179 sistemas pero calculados con la base de valencia doble zeta 6-31+G(d,p). Además de la base, la única otra diferencia que tienen ambas redes neuronales es el valor para la constante a : la red TT fue entrenada con valores b_{opt} calculados con $a = -0.140$; para la red TD se escogió $a = -0.099$ por ser el valor con el que se obtuvieron los mejores resultados después de hacer distintas evaluaciones con valores de a dentro del rango $[-0.240, -0.060]$. Los hiperparámetros y la división de subconjuntos de entrenamiento fueron calibrados para la red TT en específico, pero se utilizaron los mismos para construir la red TD, es decir, ambas redes comparten los hiperparámetros de la Tabla 4.1 y tienen a los mismos sistemas en los conjuntos de entrenamiento, validación y prueba; por esta razón no se reporta un proceso de calibración de hiperparámetros para TD.

Los resultados del entrenamiento muestran la manera en la que se llegó a los pesos Θ que minimizan la función de costo $J(\Theta)$. En general, entrenar una red neuronal de 8 neuronas escondidas con Keras no lleva mucho tiempo, aunque depende del número de épocas y lote (aproximadamente 40 segundos en una computadora personal para 1000 épocas y tamaño de lote de 10). Debido a que TT y TD son estructuralmente iguales, se puede utilizar el mismo código para entrenarlas. En el Apéndice A se encuentra el código en Python con la implementación del entrenamiento.

Uno de los resultados más relevantes es el número de épocas y el rendimiento de cada una, como se muestra en la Figura 5.1. Es posible ver que no hay sub/sobre-parametrización en ninguna de las 2 redes neuronales, puesto que el valor de $J(\Theta)$ se mantiene bajo para los tres conjuntos y en ningún momento se elevan los valores correspondientes a la validación ni al entrenamiento.

Otro resultado relevante para descartar casos de sub/sobre-parametrización es la curva de validación, que se muestra en la Figura 5.2. A partir de un número de datos de entrenamiento m , $J(\Theta)$ de validación mantiene un valor similar a $J(\Theta)$ de entrenamiento, lo que significa que un mayor número de datos no mejorará significativamente el desempeño de las redes y que no hay una sobre-parametrización. Los valores de $J(\Theta)$ de entrenamiento y validación son además bastante bajos con una mínima diferencia entre ellos, lo que descarta una sub-parametrización.

Keras obtiene las matrices de pesos de la última época registrada, que en este caso es la época con el menor $J(\Theta)$ para el conjunto de entrenamiento. La arquitectura y las matrices de pesos

son el resultado de este trabajo, ya que son las únicas variables que se necesitan para operar la red neuronal. Es decir, no es necesario contar con Keras para operar la red.

Los pesos para pasar de la capa de entrada a la capa escondida son

$$\Theta_{TT}^{(1)} = \begin{bmatrix} 0.497913 & -0.822109 & -2.057387 & 0.993503 & -1.507902 \\ -1.097269 & 0.533072 & 1.796869 & 0.140544 & 2.704808 \\ -1.253098 & 1.321370 & 0.479851 & -1.086221 & 0.255773 \\ 1.633310 & -0.483589 & -0.128530 & -0.126170 & -0.816542 \\ 0.808812 & -1.438255 & 1.476784 & -1.034514 & 0.369751 \\ 1.485057 & 1.464596 & -1.499638 & 0.007738 & -0.393483 \\ -1.844564 & -0.526694 & -0.348819 & 0.834965 & 1.205243 \\ 0.942457 & 0.336135 & 0.066630 & 1.595861 & 0.010933 \end{bmatrix}$$

$$\Theta_{TD}^{(1)} = \begin{bmatrix} 2.476633 & -1.346263 & -0.315505 & -2.795317 & -0.909047 \\ 1.228467 & -0.596272 & -0.147831 & 1.257303 & 0.382971 \\ -0.572251 & -0.348756 & -0.073778 & 0.019509 & 0.243445 \\ 1.763015 & -1.152152 & 0.670407 & -0.476411 & -0.304579 \\ -0.270864 & -0.145610 & 0.110506 & 0.352332 & 0.302635 \\ -0.734281 & -0.039326 & 0.407466 & -0.275177 & 0.119840 \\ 0.323330 & 0.218195 & 0.042308 & -0.484794 & -0.379572 \\ -0.386124 & -0.020658 & 0.479081 & -0.316392 & 0.026799 \end{bmatrix} \quad (5.1)$$

y los pesos para pasar de la capa escondida a la capa de salida son

$$\Theta_{TT}^{(2)} = \begin{bmatrix} -0.172117 \\ -1.024885 \\ 0.971249 \\ -1.379570 \\ 1.645000 \\ 0.474365 \\ 1.419526 \\ -2.466847 \\ 1.025032 \end{bmatrix}^T$$

$$\Theta_{TD}^{(2)} = \begin{bmatrix} -1.003608 \\ 1.684502 \\ 1.014663 \\ 1.008943 \\ 1.255469 \\ 0.634135 \\ 0.934177 \\ -2.160412 \\ 0.785697 \end{bmatrix}^T \quad (5.2)$$

donde los subíndices TT y TD indican a cuál red neuronal pertenecen.

En la Figura 5.3 se muestra la relación entre los valores b predichos por la respectiva red neuronal (b_{rn}) y los valores objetivo b_{opt} , agrupados en los conjuntos de entrenamiento, validación y prueba. Cabe destacar que ninguna red neuronal presenta un mejor desempeño en el conjunto de entrenamiento que en los conjuntos de validación y de prueba, sino que en los 3 conjuntos hay un desempeño similar, dejando en evidencia que no existe una sobre-parametrización y sugiriendo que ambos modelos logran generalizar para sistemas con los que no fueron entrenados.

El histograma de errores, Figura 5.4, es una gráfica que permite conocer el rango de error con el que trabaja la red. A diferencia de $J(\Theta)$, el error calculado para esta gráfica no está elevado al cuadrado, sino que es la resta de $b_{opt} - b_{rn}$. Para la red TT, la mayoría de los sistemas se encuentra en un rango de error entre -0.365 y 0.365 (2σ). En la misma figura se observa que el

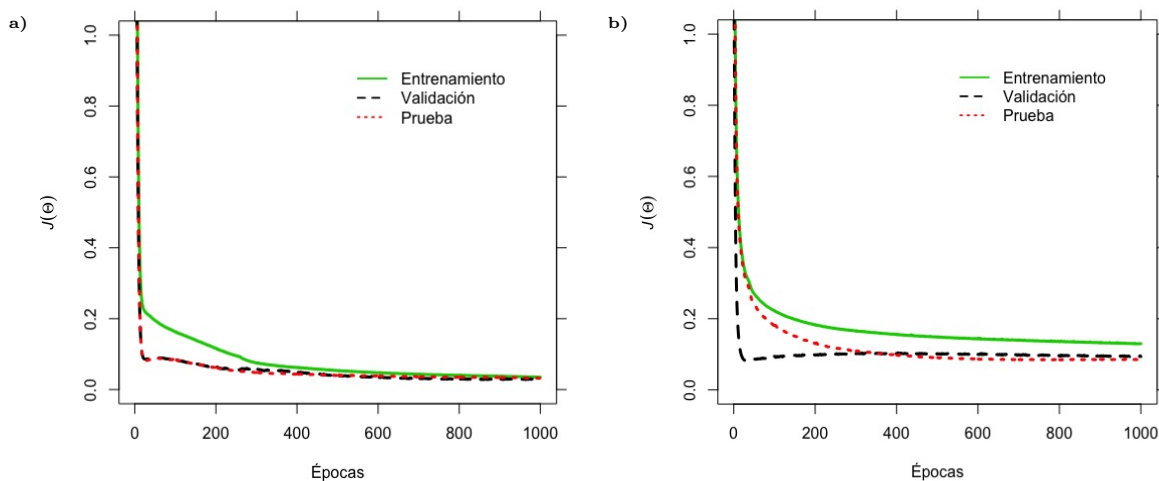


Fig. 5.1: Comportamiento de la función costo durante el proceso de entrenamiento para a) TT y b) TD

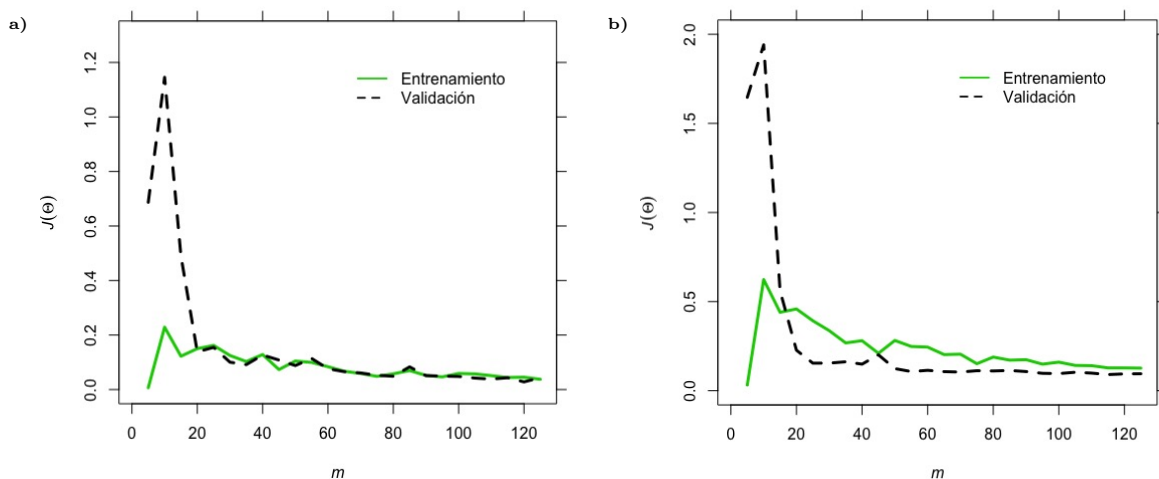


Fig. 5.2: Curvas de validación para a) TT y b) TD

error para b_{opt} mínimo fue de -0.806 y el error máximo fue de 0.493 , correspondientes al átomo de He (índice 11, subconjunto **A0**) y a la molécula F_2 con distancia de enlace 1.84 \AA (índice 60, subconjunto **DS2**), respectivamente. Los resultados son ligeramente menos precisos para el caso de TD, con un error máximo de 1.288 y un error mínimo de -2.364 , correspondientes al átomo de Se en estado singulete (índice 26, subconjunto **AM**) y al átomo de Kr (índice 12, subconjunto **A0**), respectivamente.

Teniendo ya los pesos optimizados Θ , el tiempo de cálculo de b_{rn} sumado al tiempo de cálculo para el procedimiento CTCE es en promedio 10 segundos utilizando una computadora personal, dependiendo el número de orbitales en el sistema. Se podría decir entonces que, utilizando cualquiera de las redes neuronales entrenadas, el tiempo de cálculo de las energías totales por sistema (incluyendo la correlación) es prácticamente igual al tiempo de cálculo HF correspondiente.

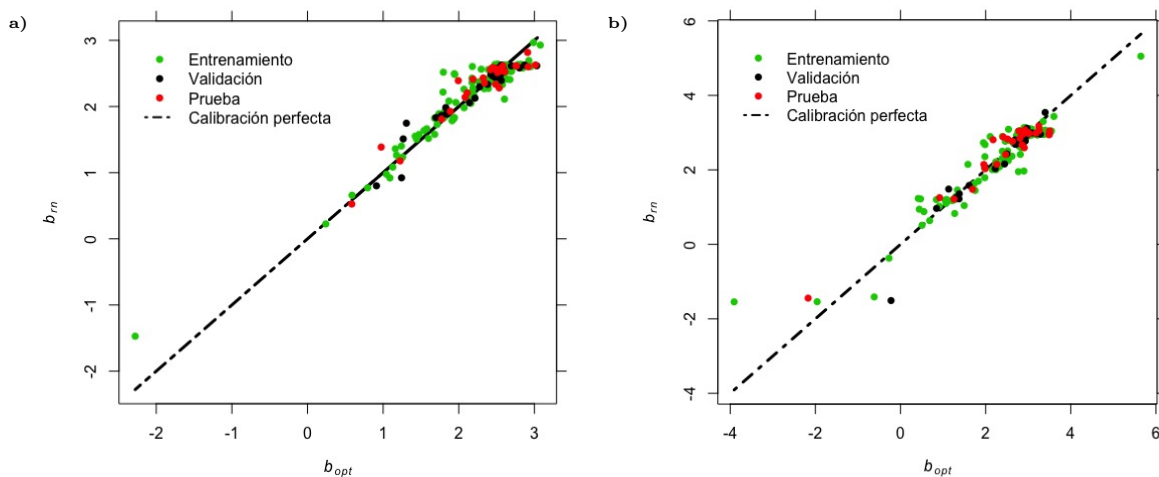


Fig. 5.3: Ajuste de datos predichos b_{rn} vs datos reales b_{opt} para a) TT y b) TD

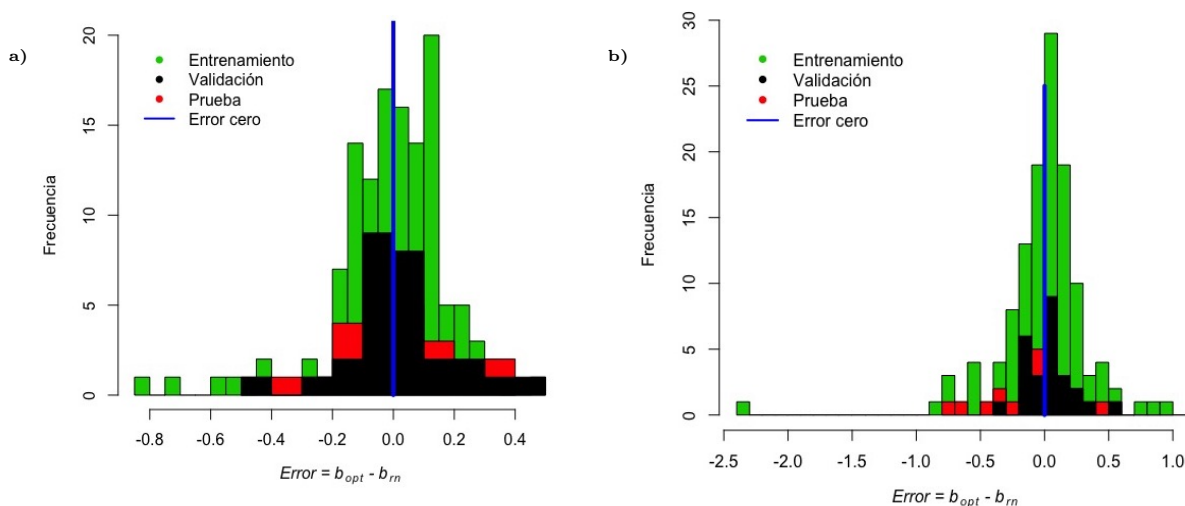


Fig. 5.4: Histograma de error en b_{opt} , objetivos - resultados, para a) TT y b) TD

Para tener una idea más general del desempeño de TT y TD, en la Figura 5.5 se muestra el ajuste de E_θ calculadas con b_{rn} y los valores objetivo $E_{corr,QCI}$ para ambas redes.

En la Figura 5.6, similar a la Figura 5.4, se muestra el histograma de errores con las energías de correlación. Se observa que la mayoría de los sistemas predichos con TT se encuentran dentro del intervalo de error entre -0.177 y 0.177 Hartrees (2σ). El error con el valor mínimo obtenido fue de -0.250 Hartrees y el máximo fue de 0.800 Hartrees, correspondientes a la molécula de indol (índice 89, subconjunto **HETC**) y al catión bencilo (índice 42, subconjunto **CYCP**), respectivamente. El caso de TD es bastante peculiar, a pesar de tener una menor exactitud en b_{rn} , la exactitud en E_θ es similar a la de TT. El error mínimo obtenido fue de -0.328 Hartrees y el máximo fue de 0.592 Hartrees, correspondientes a la molécula de F_2 con distancia de enlace 3.1 \AA (índice 61, subconjunto **DS2**) y al catión bencilo (índice 42, subconjunto **CYCP**), respectivamente. Estos errores en los extremos no corresponden a los mismos sistemas que los errores en la predicción de

b_{opt} , debido a que la sensibilidad (Capítulo 2) es distinta en cada sistema y en cada base.

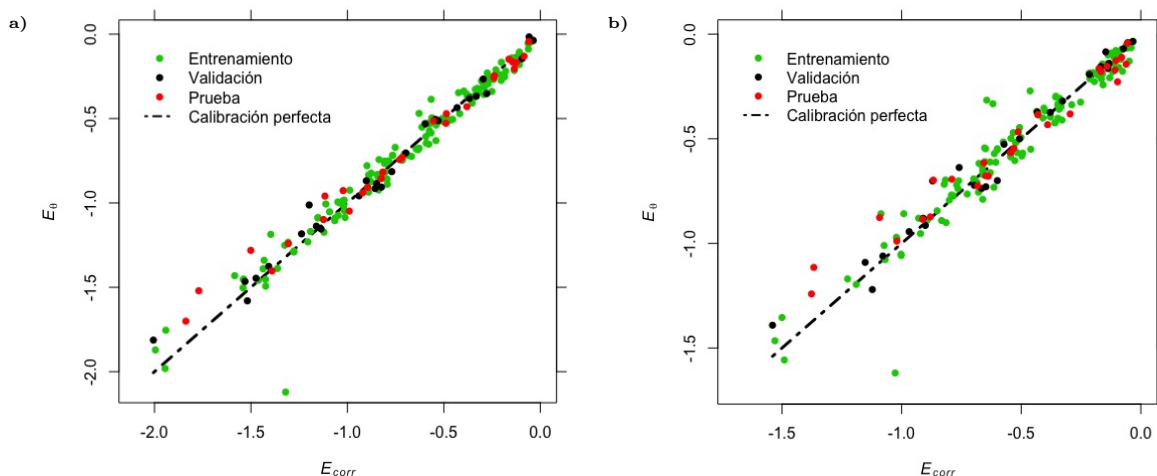


Fig. 5.5: Ajuste de energías de correlación (Hartrees) predichas por la red E_{θ} vs energías de correlación E_{QCI} para a) TT y b) TD

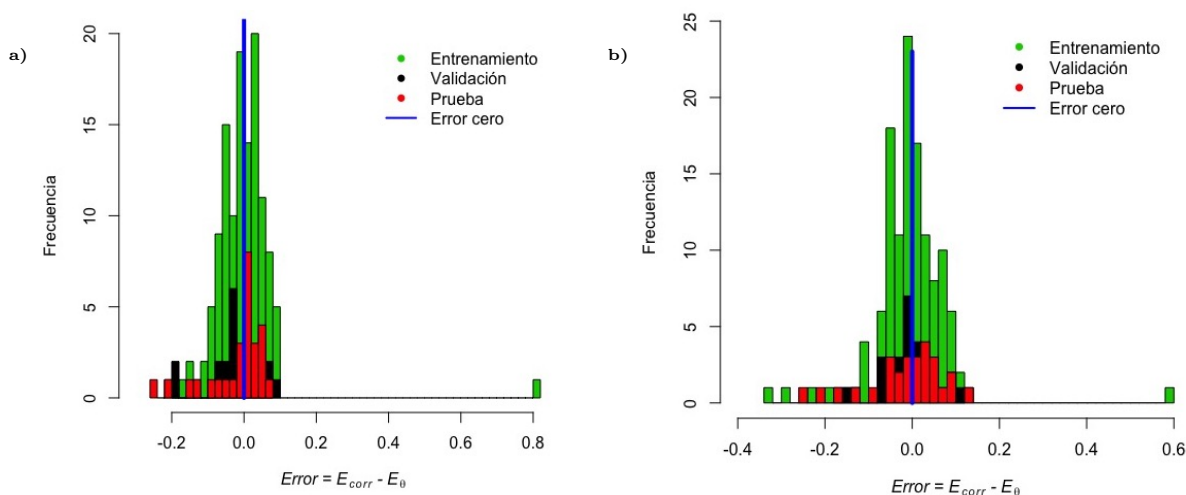


Fig. 5.6: Histograma de error para E_{corr} , objetivos - resultados, para a) TT y b) TD

En la Tablas 5.1 se encuentran los errores absolutos relativos para el cálculo de E_{θ} utilizando las redes TT y TD. Como se indicó en la Sección 4.4, el conjunto de validación tiene el propósito de escoger un modelo y el conjunto de prueba indica el error generalizado de dicho modelo. Se puede observar que la red neuronal TT tiene un error generalizado de 11.76%, lo que sugiere que ya está lista para operarse, es decir, para predecir la correlación electrónica de cualquier sistema que se encuentre dentro del espacio de los valores de entrada. A pesar de que el error generalizado de TD es casi el doble, con 22.58%, cumple con el objetivo propuesto del trabajo con un error total de 15.60% y también está lista para se utilizada en cualquier sistema.

Tabla 5.1: Errores absolutos relativos para las redes neuronales

Conjunto de entrenamiento	Error absoluto relativo $100 \% \times (E_{corr} - E_{\theta})/E_{corr} $	
	TT	TD
Entrenamiento	9.40 %	15.37 %
Validación	10.44 %	9.64 %
Prueba	11.76 %	22.58 %
Total	9.91 %	15.60 %

5.2. Cálculo de la correlación

Como se mencionó en el Capítulo 4, se entrenaron 2 redes neuronales con el fin de poder evaluar el desempeño del modelo en moléculas de diversos tamaños. En esta Sección se presentan los cálculos de correlación utilizando una red neuronal según el tamaño de las moléculas: la red neuronal TT se utilizó para moléculas con una cantidad de electrones menor o igual a 300 (Secciones 5.2.1-5.2.5) mientras que la red neuronal TD se utilizó para moléculas con más de 300 electrones (Sección 5.2.6).

5.2.1. Disociación de la molécula de hidrógeno

La disociación de la molécula de H_2 involucra el rompimiento de un enlace sencillo, donde el estado basal es singulete en el enlace y triplete en el límite de disociación infinita. El cálculo HF de espín restringido no ofrece resultados cualitativos a distancias internucleares grandes debido a que la energía sigue conteniendo un término de repulsión electrónica, resultado de los 2 electrones compartiendo el mismo orbital. La Figura 5.7 muestra la curva de la energía potencial para la disociación de H_2 utilizando el estado singulete en HF. La red neuronal se utilizó de 2 maneras: (i) predecir b_{rn} distinta para cada geometría (ii) fijar el valor b_{rn} de la geometría de mínima energía para todas las geometrías. Se tomó la precaución de no incluir distancias que tengan valores de entrada fuera del espacio de trabajo de la red neuronal. El cálculo de la correlación QCISD(T) es el mismo que QCISD porque las excitaciones triples no tienen lugar en la molécula.

Las distancias R en H_2 que se utilizaron para entrenar la red (sistemas que pertenecen al conjunto de entrenamiento) fueron a 0.73, 1.54, 2.17 y 3.70 angstroms. Se observa que la E_{θ} calculada a partir de b_{rn} , tanto fija como variable, provee una mejora con respecto al punto de partida HF, incluso para distancias que no fueron consideradas en el entrenamiento de la red neuronal.

Para vislumbrar el efecto que tienen \bar{E}_T , \bar{E}_K , ε_{HOMO} y ε_{LUMO} sobre la predicción de b_{rn} (variable), en la Figura 5.8 se muestran las entradas y las salidas. Las entradas \bar{E}_T y \bar{E}_K son casi constantes al aumentar R , las variaciones en b_{rn} podrían atribuirse principalmente a las variaciones de ε_{HOMO} y ε_{LUMO} . Aunque en la Figura 5.8a no se observa a simple vista, la Figura 5.8b deja en evidencia que E_{θ} es ligeramente más profunda que $E_{corr,QCI}$ a 0.53 angstroms porque b_{rn} es mayor que b_{opt} . La curva de b_{opt} muestra un máximo en los valores cercanos de R cuando los valores escalados de ε_{HOMO} y ε_{LUMO} se invierten. Aunque la red neuronal reproduce cualitativamente este comportamiento, presenta el mayor error para la distancia de 3.70 angstroms. Este error se ve reflejado en el límite de disociación infinita, que se ve bastante similar a HF. A pesar de esto, E_{θ} presenta una mejora con respecto a HF en todos los puntos de la Figura 5.7. Vale la pena mencionar que no es posible calcular de manera confiable E_{θ} a distancias mayores que 3.70 angstroms porque los valores de entrada no se encuentran dentro del espacio de trabajo

de la red neuronal.

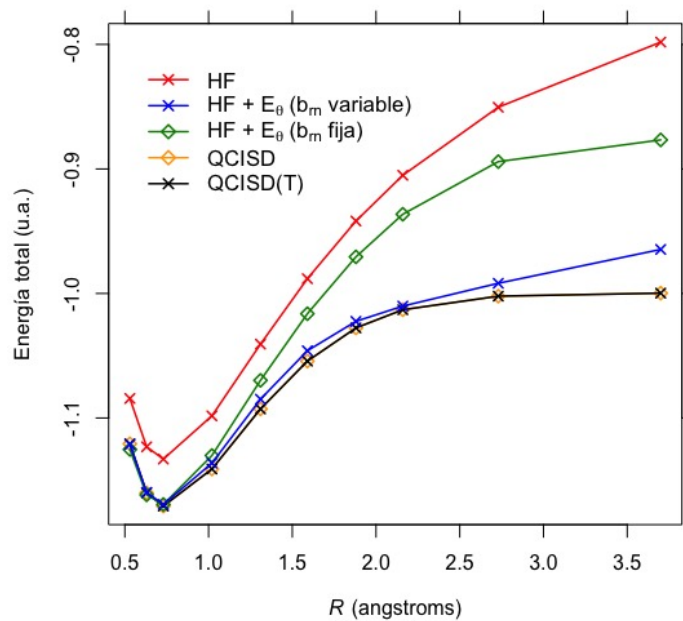


Fig. 5.7: Curvas de energía potencial de H₂, calculados con los niveles de teoría HF de espín restringido, QCISD, QCISD(T) y con E_{θ} predicha por la red neuronal TT a partir de valores entrada obtenidos con el nivel de teoría HF de espín restringido

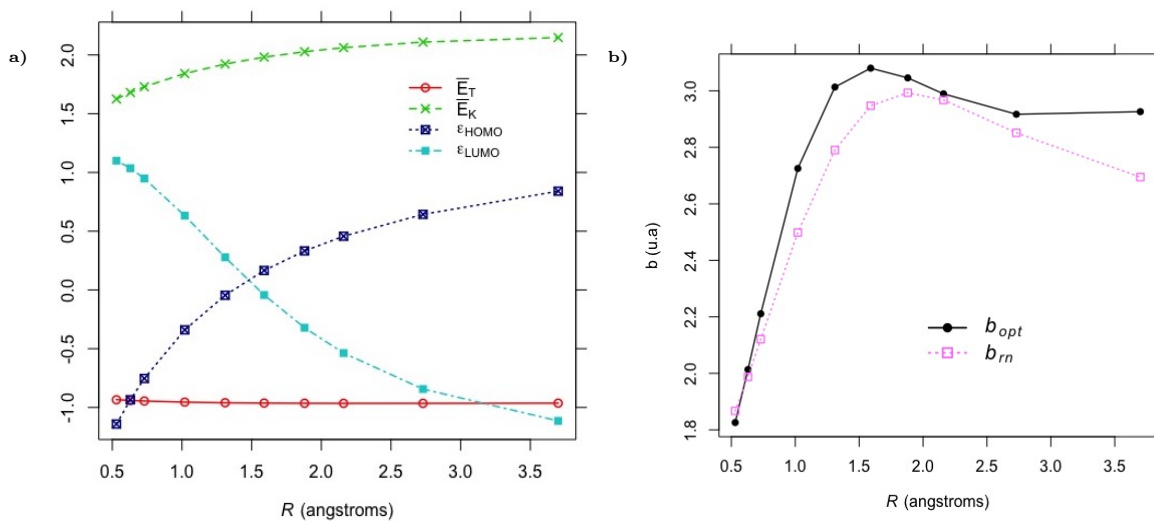


Fig. 5.8: a) Valores de entrada escalados (Sección 4.3) y b) valores de salida para H₂ como función de la distancia internuclear R

5.2.2. Disociación de la molécula de nitrógeno

La curva de energía potencial para la disociación de N_2 en el estado basal es un ejemplo frecuentemente utilizado para poner a prueba el desempeño de nuevos métodos orientados a la determinación precisa de la energía de correlación [29, 55], siendo un caso más desafiante de reproducir que H_2 con métodos ab initio por involucrar el rompimiento de un enlace triple. La Figura 5.9 muestra la curva de la energía potencial para la disociación de N_2 utilizando el estado singlete en HF. Se tomó la precaución de no incluir distancias que tengan valores de entrada fuera del espacio de trabajo de la red neuronal. En este caso, el cálculo de la correlación QCISD(T) resulta en una mejora con respecto a QCISD en distancias cercanas a la de mínima energía, pero no así con distancias mayores. Así mismo, se muestran resultados utilizando b_{rn} variable y b_{rn} fija.

Es muy importante resaltar que la red neuronal no fue entrenada con ningún sistema N_2 porque incluso la molécula N_2 de mínima energía (índice 141, subconjunto **NMOL**) pertenece al conjunto de validación. Es decir, la red neuronal no ajustó sus pesos Θ tomando en cuenta N_2 a ninguna distancia R , la Figura 5.9 es resultado de la generalización que tiene la red para predecir especies con las que no fue entrenada. Aunque el método no es variacional y con él se obtuvieron energías ligeramente más profundas que $E_{corr,QCI}$ para distancias menores a la de mínima energía, la E_θ calculada a partir de b_{rn} sí proporciona una mejora destacable con respecto al punto de partida HF. Un punto a resaltar es que, como la red no fue entrenada en ningún momento con valores erróneos QCISD(T), se logra reproducir el comportamiento asintótico del límite de disociación al igual que QCISD.

Al igual que con H_2 , la Figura 5.10 muestra las entradas y las salidas (b_{rn} variable). La entrada que menos variación presenta en función de R es \bar{E}_T . Los primeros 3 valores de b_{rn} son mayores que b_{opt} , lo que explica las energías más profundas a estas distancias. A partir de 2.2 angstroms la curva b_{rn} logra reproducir b_{opt} cualitativamente, la diferencia entre ambas se debe a la diferencia que hay entre QCISD y QCISD(T). Resulta interesante que la variación de b_{rn} sea muy baja a diferencia de b_{opt} y aún así se calcule correctamente E_θ . El error en b_{rn} con respecto a b_{opt} afecta poco en el cálculo de E_θ porque la sensibilidad (Capítulo 2) en función de la distancia internuclear R es baja. Esto también se ve reflejado también en la curva de disociación utilizando b_{rn} fija, pues es muy similar al resultado con b_{rn} variable.

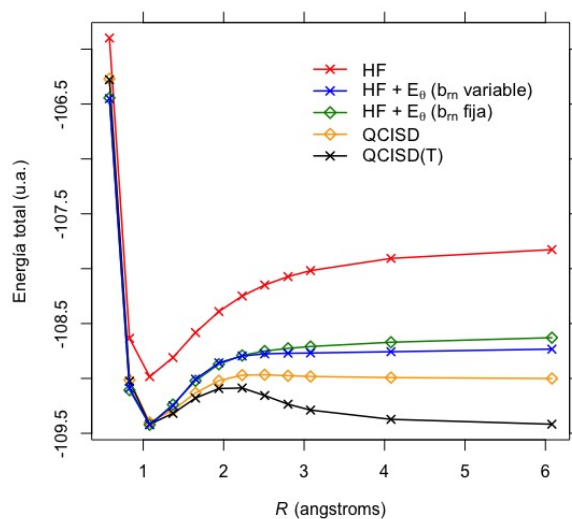


Fig. 5.9: Curvas de energía potencial de N_2 , calculados con los niveles de teoría HF de espín restringido, QCISD, QCISD(T) y con E_θ predicha por la red neuronal TT a partir de valores entrada obtenidos con el nivel de teoría HF de espín restringido

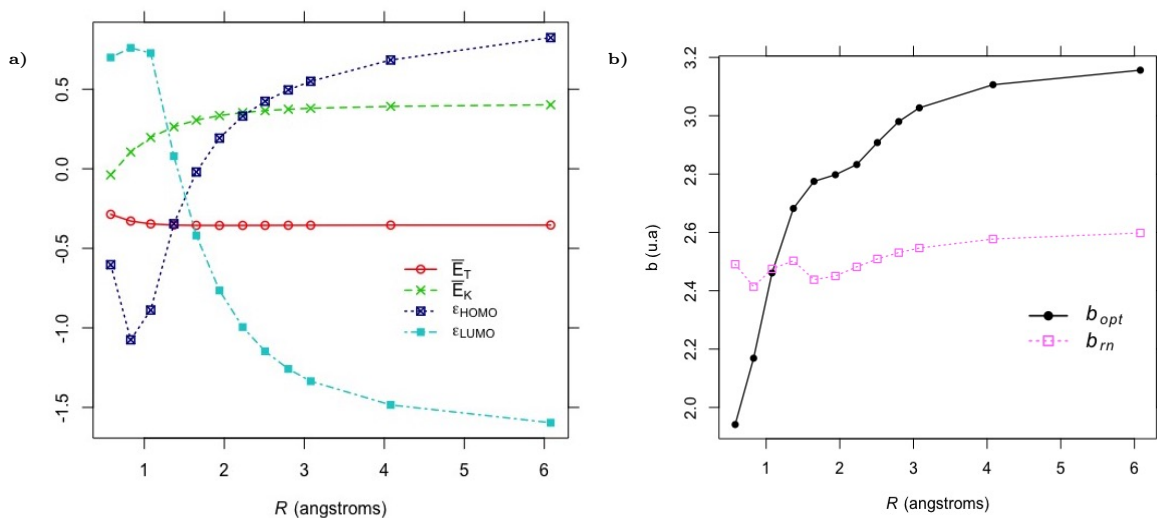


Fig. 5.10: a) Valores de entrada escalados (Sección 4.3) y b) valores de salida para N_2 como función de la distancia internuclear R

5.2.3. Disociación de la molécula de carbono

Debido a que múltiples orbitales moleculares participan en el enlace en cambios pequeños de distancias intermoleculares y al entrecruzamiento de estados a 1.7 angstroms [55, 56], la curva de disociación de C_2 suele ser un desafío para muchos métodos de estructura electrónica. En la Figura 5.11 se muestra la curva de la energía potencial para la disociación de C_2 utilizando el estado singulete en HF. Se tomó la precaución de no incluir distancias que tengan valores de entrada fuera del espacio de trabajo de la red neuronal. Al igual que con la disociación de N_2 , el cálculo de la correlación QCISD(T) da un comportamiento erróneo en el límite de disociación

infinita, a diferencia de la correlación QCISD.

Las distancias R en C_2 que se utilizaron para entrenar la red (sistemas que pertenecen al conjunto de entrenamiento) fueron 1.26, 1.83, 4.26 angstroms en estado singulete y 1.24 angstroms en estado triplete. Debido a la precaución que se tomó de entrenar la red con valores QCISD (Sección 4.2), E_θ corrige a HF incluso en los valores de disociación infinita, a diferencia de QCISD(T). Aunque no fue posible corregir por completo la discontinuidad de HF para reproducir el comportamiento más suavizado de QCISD, sí fue posible disminuirla considerablemente. Al igual que con H_2 y N_2 , E_θ proporciona una mejora incluso para distancias con las que la red neuronal no fue entrenada.

En la Figura 5.12 se muestran las entradas y las salidas. A diferencia de H_2 y N_2 , los valores escalados de ε_{HOMO} y ε_{LUMO} no se invierten en ningún momento: la curva de ε_{HOMO} presenta un mínimo en los alrededores de la distancia de mínima energía y ε_{LUMO} es la propiedad que presenta la discontinuidad más evidente a la misma distancia que HF. Vale la pena mencionar que, aunque \bar{E}_T parece constante como función de R en los 3 dímeros hasta ahora analizados, su importancia parece radicar en los valores promedio que predice para b_{rn} y juega un papel más importante cuando se comparan sistemas con distintos elementos entre sí. En la Figura 5.12b se observa con mayor claridad a qué distancias se obtienen E_θ más profundas que E_{corr} , pues coinciden con las distancias que tienen b_{rn} mayores a b_{opt} . En el caso de b_{opt} , hay una discontinuidad muy marcada en los alrededores de 1.7 angstroms, debido a que es necesario igualar la discontinuidad en HF para igualar las energías de correlación QCISD(T) con un comportamiento más suavizado. Esto deja en evidencia que el dímero C_2 tiene la misma ventaja que en los casos anteriores: el error aparentemente grande en la predicción b_{rn} no se ve reflejado del mismo modo en el cálculo de E_θ . Al igual que en N_2 , la curva de disociación con b_{rn} fija es muy similar a la de b_{rn} variable. Esto sugiere que para pequeños cambios en la geometría, el mismo valor de b sigue ofreciendo buenos resultados al momento de calcular la correlación electrónica.

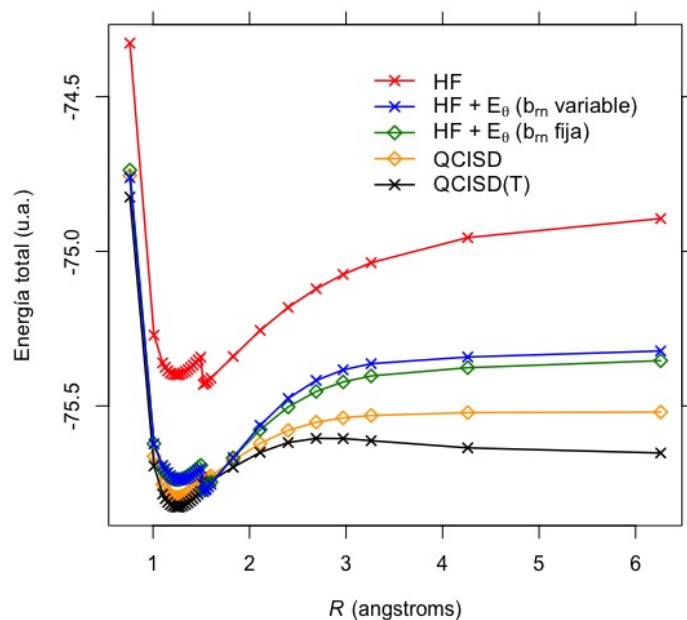


Fig. 5.11: Curvas de energía potencial de C_2 , calculados con los niveles de teoría HF de espín restringido, QCISD, QCISD(T) y con E_0 predicha por la red neuronal TT a partir de valores entrada obtenidos con el nivel de teoría HF de espín restringido

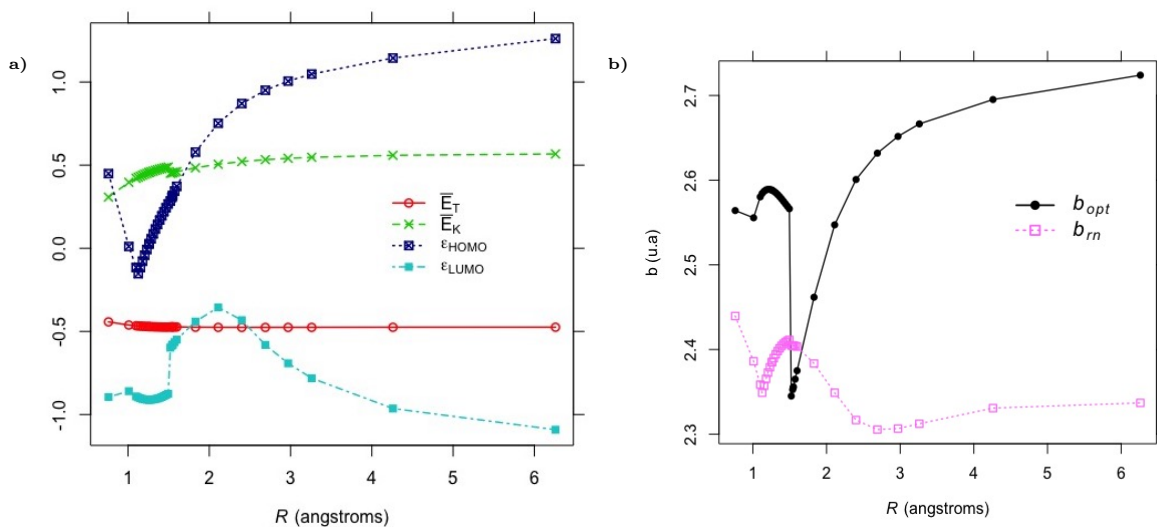


Fig. 5.12: a) Valores de entrada escalados (Sección 4.3) y b) valores de salida para C_2 como función de la distancia internuclear R

5.2.4. Rotación de la molécula de etileno

La molécula de etileno con un ángulo de torsión de 90° tiene un estado triplete de mínima energía [57] debido a que los orbitales π y π^* están degenerados. Los cálculos de espín restringido son insuficientes para describir de manera apropiada esta degeneración, resultando en una superficie

de energía potencial con una cúspide en los alrededores del ángulo 90° [29]. En la Figura 5.13 se muestra la superficie de energía potencial relativa de torsión para la molécula de etileno.

La molécula de etileno de mínima energía (índice 35, subconjunto **CMOL**) pertenece al conjunto de entrenamiento, el resto de las geometrías en la Figura 5.13 son sistemas con los que la red neuronal no había sido entrenada. Esta curva resulta ser un caso particularmente complicado para la red: a pesar de que recupera el 88.05 % de la energía de correlación en el ángulo 90° , el error se hace más evidente al compararlo con el 112.84 % de energía de correlación que predice para el ángulo 0° .

En la Figura 5.14 se muestran las entradas y las salidas. Las variaciones de b_{rn} se deben principalmente a las variaciones de ε_{HOMO} y ε_{LUMO} , pues \bar{E}_T y \bar{E}_K permanecen prácticamente constantes. Los valores b_{rn} mayores a b_{opt} dejan en evidencia los ángulos que corresponden a E_θ más profundas que $E_{corr,QCI}$. Se puede observar que la curva b_{opt} tiene una cúspide, necesaria para igualar la cúspide de HF y así igualar el comportamiento QCISD(T). Por otra parte, la curva b_{rn} tiene un comportamiento cualitativo completamente opuesto a b_{opt} . Vale la pena mencionar que, mientras que en las disociaciones parecía que b_{rn} adoptaba principalmente la forma cualitativa de ε_{HOMO} en la ruptura del enlace, este no es el caso para el etileno, lo que refleja la naturaleza no lineal de la red neuronal para resolver el problema.

Lamentablemente, corregir la cúspide es un desafío demasiado grande para la red neuronal. En este caso, el mejor desempeño se obtiene utilizando como valor fijo $b_{rn} = 2.56$, valor correspondiente a la predicción para la geometría del etileno en mínima energía. Este resultado se muestra en la Figura 5.15.

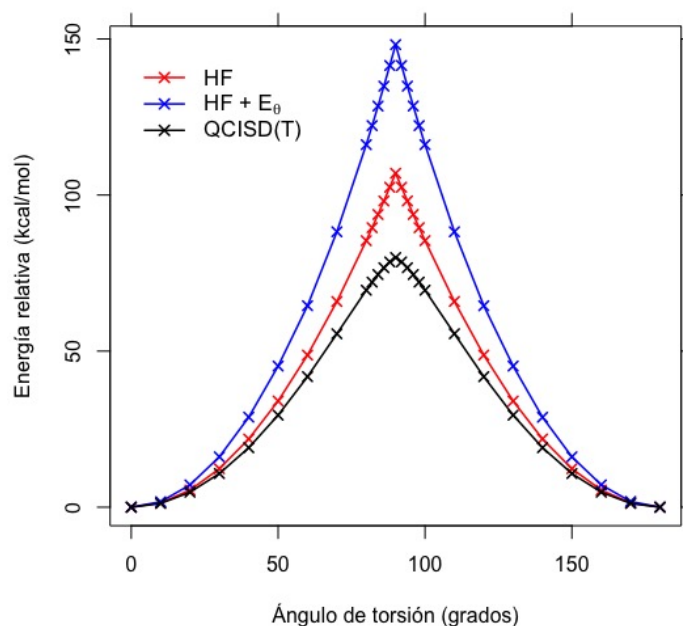


Fig. 5.13: Curvas de energía potencial relativa de etileno, calculados con los niveles de teoría HF de espín restringido, QCISD(T) y con E_θ predicha por la red neuronal TT a partir de valores entrada obtenidos con el nivel de teoría HF de espín restringido.

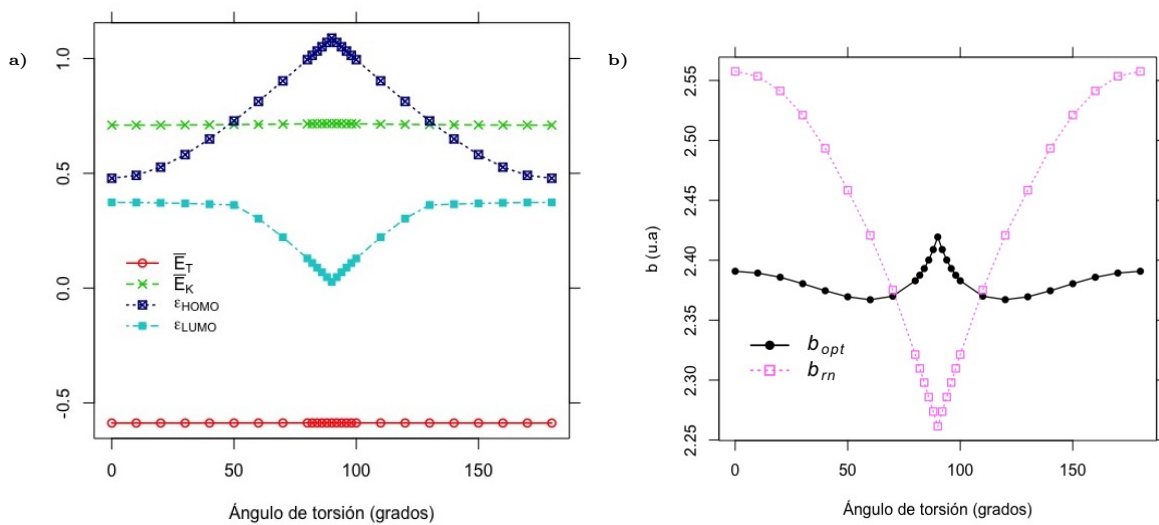


Fig. 5.14: a) Valores de entrada escalados (Sección 4.3) y b) valores de salida para el etileno como función del ángulo de torsión

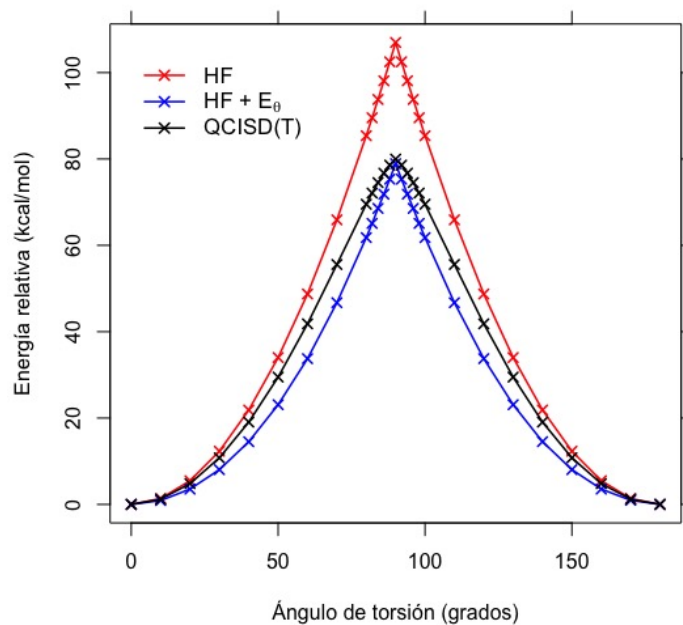


Fig. 5.15: Curvas de energía potencial relativa para el estado singlete de etileno, calculados con los niveles de teoría HF de espín restringido, QCISD(T) y con E_θ calculada con $b_{rn} = 2.56$ constante

5.2.5. Moléculas de entre 80 y 300 electrones

Se consideraron las energías de correlación CCSD(T) de 21 moléculas pertenecientes al conjunto de prueba CEMS26, descrito por Nagy y colaboradores [58], para examinar el desempeño de la red neuronal con moléculas más grandes que las que se utilizaron para entrenarla. Aunque los resultados reportados fueron con un nivel de teoría distinto al utilizado para entrenar la red TT, es posible analizar el desempeño cualitativo del modelo. En la Figura 5.16 se muestra la energía de correlación CCSD(T) y E_θ en función del número de electrones.

Tomando en cuenta que la red neuronal proporciona la correlación electrónica a partir de parámetros obtenidos de HF, su desempeño general es destacable. No obstante, hay 2 moléculas que presentan una diferencia muy grande entre la correlación CCSD(T) y E_θ : Octametil-silsesquioxano (OMS) y el complejo de Li^+ con dos moléculas de 12-corona-4 éter ($[\text{Li}(\text{corona})]^+$). En la Tabla 5.2 se muestran los valores de entrada correspondientes a estas moléculas. Aunque los valores de entrada se encuentran dentro del espacio de trabajo de la red, la gran diferencia en la correlación podría deberse a que contienen Li y Si, elementos que no están presentes en ningún subconjunto de los 179 sistemas utilizados en este trabajo. Esta peculiaridad en ambas moléculas provoca que presenten combinaciones de valores $\{\bar{E}_T, \bar{E}_K, \varepsilon_{\text{HOMO}}, \varepsilon_{\text{LUMO}}\}$ que no tiene ninguno de los 179 sistemas. Las más evidentes corresponden a las relaciones $\{\bar{E}_T, \varepsilon_{\text{HOMO}}\}$ para la molécula $[\text{Li}(\text{corona})]^+$ y $\{\bar{E}_T, \bar{E}_K\}$ para la molécula OMS, representadas en la Figura 5.17. Dado que los pesos Θ no fueron ajustados dentro del espacio de estos valores de entrada, está ocurriendo algo similar a una extrapolación y resulta en un error equivalente al de predecir sistemas que no se encuentran dentro del espacio de trabajo. Vale la pena revisar los valores de entrada con atención si se quiere utilizar la red neuronal para predecir sistemas con elementos distintos a los utilizados

para entrenarla.

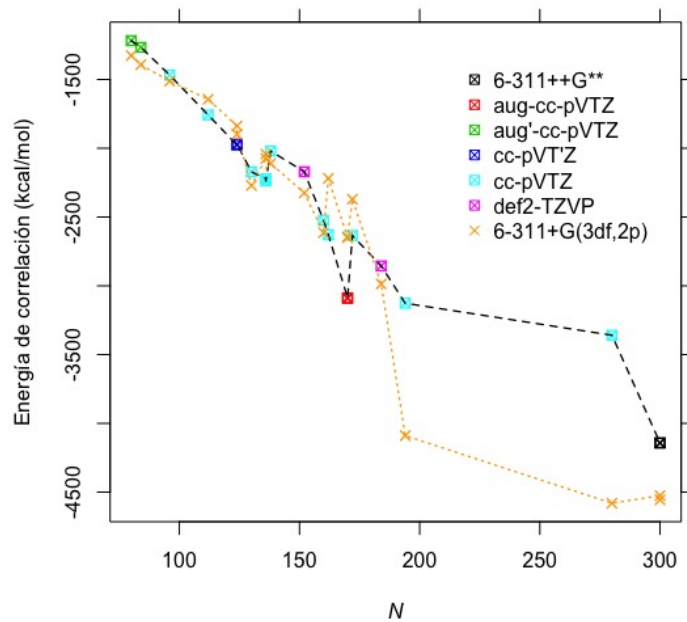


Fig. 5.16: Energía de correlación CCSD(T) (línea punteada negra) y E_θ obtenida a partir de TT (línea punteada naranja) en función del número de electrones N para el conjunto CEMS26

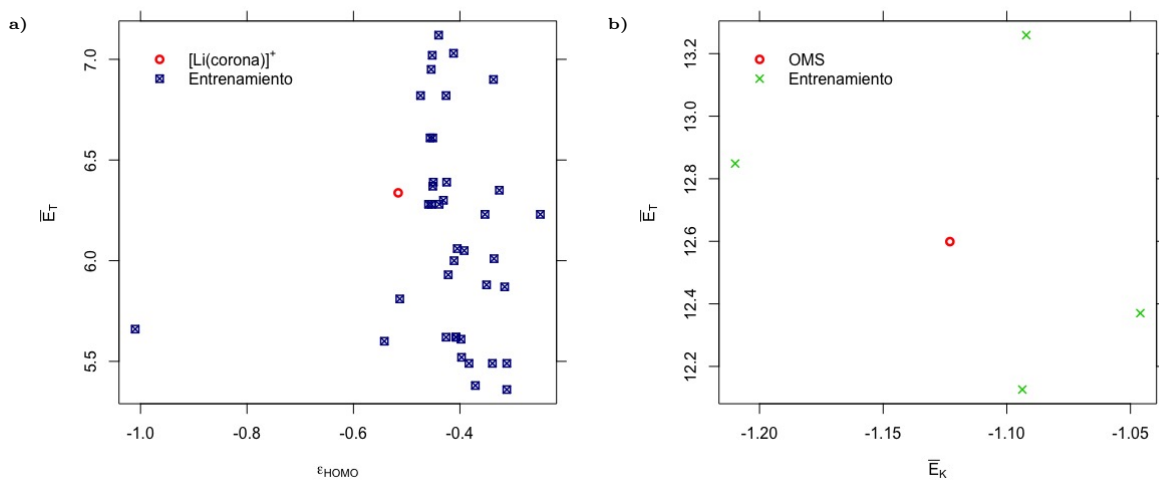


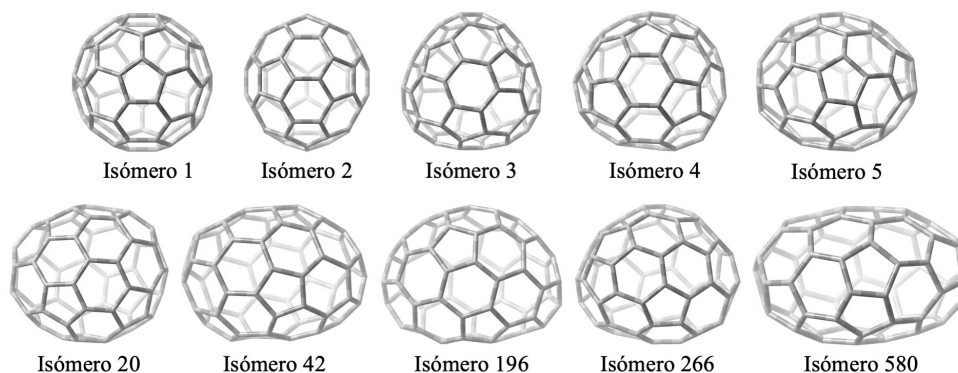
Fig. 5.17: Comparación de los valores de entrada de a) $[\text{Li}(\text{corona})]^+$ y b) OMS con los valores del conjunto de entrenamiento

Tabla 5.2: Número de electrones y valores de entrada (u.a.) correspondientes a [Li(corona)]⁺ y OMS

Molécula	N	\bar{E}_T	\bar{E}_K	ε_{HOMO}	ε_{LUMO}
[Li(corona)] ⁺	194	6.337	-0.837	-0.516	-0.053
OMS	280	12.599	-1.123	-0.459	0.045

5.2.6. Isómeros de C₆₀

El buckminsterfullereno (C₆₀) es una molécula esférica neutra constituida por 60 átomos de carbono en hibridación sp² y en años recientes ha sido de interés, al igual que otros nanomateriales de carbono, por sus propiedades electrónicas con posibles aplicaciones en almacenamiento de energía eléctrica y fotoconversión, entre otras [59-61]. En el 2017, Sure y colaboradores [62] realizaron cálculos al nivel de teoría DLPNO-CCSD(T)/CBS* [63] para 10 de los 1812 isómeros totales de C₆₀, presentados en la Figura 5.18.

**Fig. 5.18:** Geometrías de los 10 isómeros de C₆₀ calculados al nivel de teoría DLNPO-CCSD(T)/CBS* en [62]

Un cálculo de C₆₀ al nivel de teoría CISD/6-31+G(d,p), con 360 electrones y 1140¹ funciones base, resulta en una función de onda con aproximadamente cien mil millones² de determinantes, por lo que no es un problema tratable. No obstante, calcular E_θ utilizando la red neuronal TD toma únicamente el tiempo de cálculo HF/6-31+G(d,p) (3-19 horas, dependiendo el isómero) porque el procedimiento CTCE se efectúa en poco menos de 30 segundos.

Los 10 isómeros de esta molécula involucran un pequeño cambio en la geometría entre ellos, por lo que la red neuronal podría utilizarse de 2 maneras: (i) b_{rn} variable o (ii) b_{rn} fija. En la Figura 5.19 se muestra la energía E_θ calculada con ambos métodos y, al igual que en la rotación del etileno tratada en la Sección 5.2.4, se obtienen mejores resultados utilizando la misma b_{rn} para todas las geometrías. Así mismo, en la Figura 5.20 se muestran las entradas y salidas de la red neuronal.

¹En la práctica, Gaussian09 reduce este número a aproximadamente 1080 debido a que elimina automáticamente funciones base que son linealmente dependientes.

²La ecuación para calcular el número de excitaciones dobles es $\binom{N}{2} \binom{2K-N}{2} = \binom{360}{2} \binom{1920}{2} \approx 10^{11}$

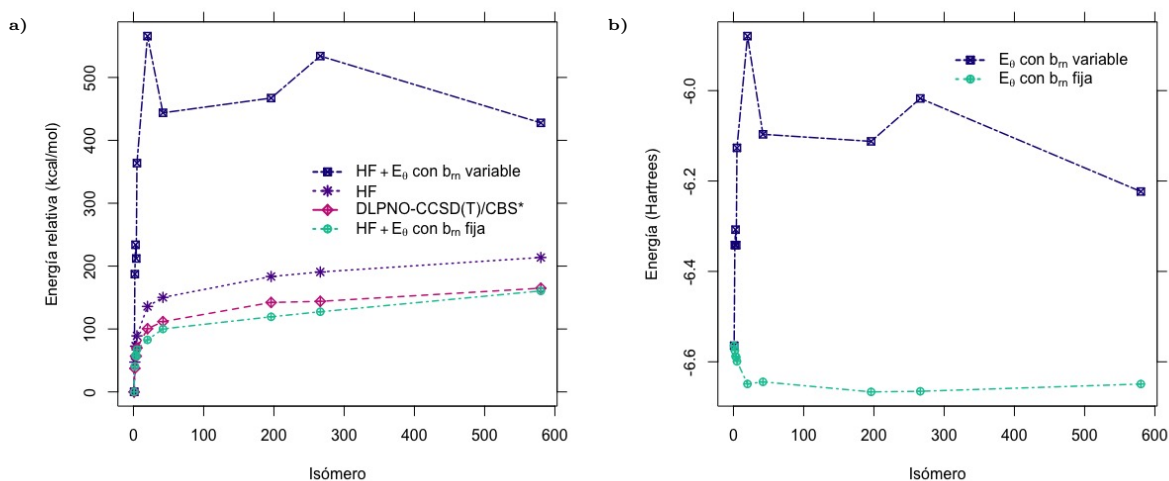


Fig. 5.19: a) Energías relativas y b) energías E_{θ} para los 10 isómeros de C_{60} calculados. Los cálculos HF son de espín restringido con la base 6-31+G(d,p), los cálculos de E_{θ} fueron realizados con valores b_{rn} proporcionados por TD. El valor de b_{rn} fijo fue el predicho para el isómero de mínima energía ($b_{rn} = 2.576$)

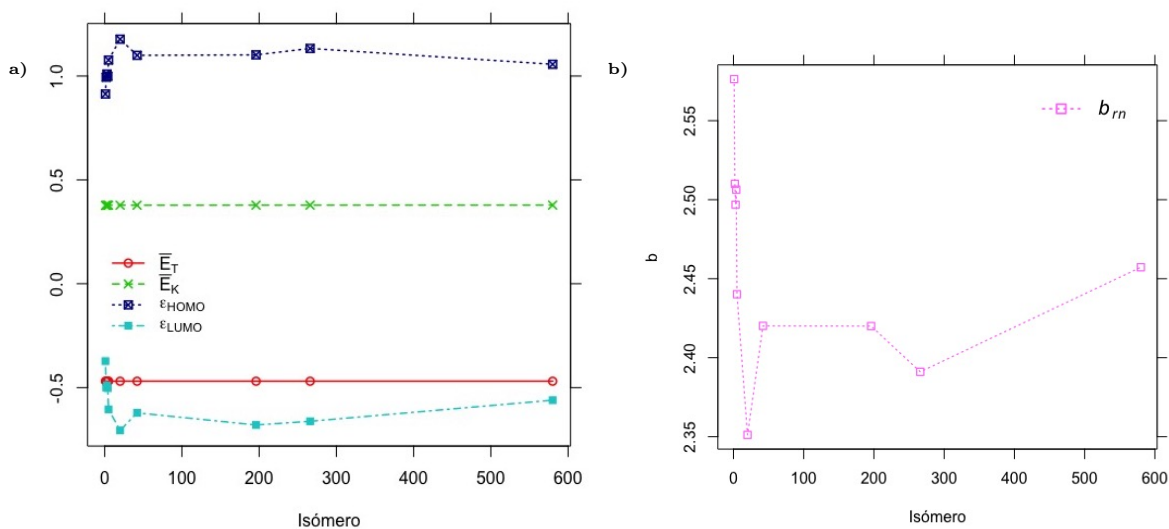


Fig. 5.20: a) Valores de entrada escalados (Sección 4.3) y b) valores de salida para los 10 isómeros de C_{60}

Es evidente que los valores de b_{rn} no reproducen cualitativamente las energías de correlación cuando se predice un valor distinto para cada geometría. Dado que se obtienen energías relativas más precisas con $b_{rn} = 2.576$, es plausible pensar que los valores b_{opt} que reproducen de manera exacta las energías $E_{corr,QCI}$ son mayores que los valores b_{rn} mostrados en la Figura 5.20b. Esto también se puede confirmar de ver la Figura 5.19b, siendo las energías E_{θ} más profundas al calcularlas con $b_{rn} = 2.576$.

En la Figura 5.21 se muestran los números de ocupación calculados con b_{rn} variable y b_{rn} fija. Dado que el cálculo HF fue de espín restringido, los espín-orbitales son degenerados y solo se muestran los orbitales espaciales. Además, se muestran únicamente los orbitales activos, definidos como los orbitales cuyos números de ocupación se encuentran entre 0.2 y 1.8. Definiendo al

HOMO como el orbital número $N/2 = 180$, es interesante ver que los orbitales activos para cualquiera de los isómeros comienzan desde el orbital HOMO-12, tanto con b_{rn} variable como con b_{rn} fija. Sin embargo, los números de ocupación descienden rápidamente después del orbital HOMO y los orbitales activos terminan antes de LUMO+4. Estos resultados sugieren que los isómeros tienen más posibilidades de aceptar electrones que de donarlos, algo ya confirmado por investigaciones anteriores [64, 65]: los isómeros de fullereno en general son moléculas deficientes de electrones.

Vale la pena mencionar que ambos métodos (b_{rn} variable y b_{rn} fija) reproducen números de ocupación bastante similares en el mismo isómero. Esto se debe a que la cantidad de orbitales es bastante grande y la suma de las pequeñas diferencias entre ambos métodos es lo que resulta en una mayor diferencia de energía. Así mismo, es importante recordar que el mismo valor b_{rn} no va a reproducir los mismos números de ocupación entre los 10 isómeros. Los valores de b y de las energías orbitales son lo que define los valores de μ y θ , y estos a su vez definen el valor de E_θ y de los números de ocupación. A pesar de fijar $b_{rn} = 2.576$, la Figura 5.21b muestra distintos números de ocupación entre los isómeros porque cada uno de ellos tiene μ , θ y energías orbitales distintos.

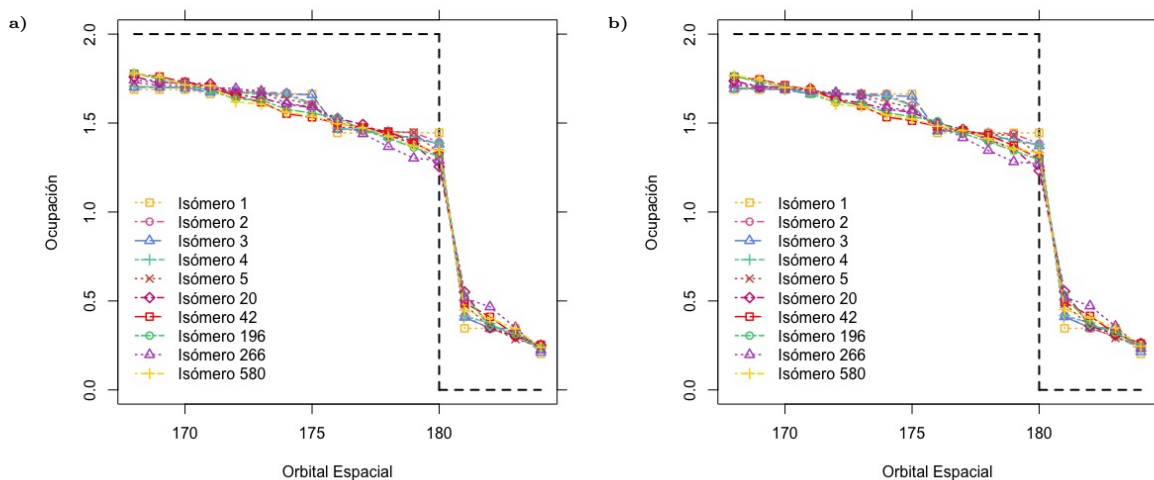


Fig. 5.21: Números de ocupación para los orbitales activos de los isómeros de C_{60} , calculados con **a)** b_{rn} variable y **b)** $b_{rn} = 2.576$

Conclusiones

En este trabajo se construye un modelo para predecir los números de ocupación de tipo Fermi-Dirac en moléculas orgánicas mediante el uso de 2 redes neuronales: *Thermal-Triple* (TT) y *Thermal-Double* (TD). Ambas redes neuronales comparten los mismos hiperparámetros y arquitectura: 4 neuronas de entrada, 8 neuronas en 1 capa escondida y 1 neurona de salida. Los valores de entrada son los componentes de la energía HF \bar{E}_T , \bar{E}_K , ε_{HOMO} y ε_{LUMO} . Las redes están calibradas para predecir el parámetro b_{opt} , a partir del cual es posible calcular E_θ mediante el procedimiento CTCE. La red neuronal *Thermal-Triple* se utiliza con la base 6-311+G(3df,2p), idealmente para moléculas pequeñas y medianas; la red neuronal *Thermal-Double* se utiliza con la base 6-31+G(d,p), idealmente para moléculas grandes.

Los resultados obtenidos durante la etapa de entrenamiento de ambas redes fueron satisfactorios, con un error generalizado del 11.76 % para TT y 22.58 % para TD en el cálculo de E_θ , por lo que recuperan el 88.24 % y 77.42 % de la energía de correlación, respectivamente. Estos resultados son destacables, considerando que la red neuronal parte del método HF. La exactitud obtenida se debe no sólo al entrenamiento de las redes neuronales, sino a la sensibilidad que ofrecen los procedimientos CBO y CTCE con los valores escogidos para la constante a .

Los resultados obtenidos en los cálculos de correlación sugieren que es posible utilizar TT y/o TD para cualquier molécula orgánica dentro del espacio de trabajo. El método aquí presentado es conceptualmente sencillo, de bajo costo computacional y fácil de implementar, por lo que ha demostrado ser una potencial alternativa para el cálculo de estructura electrónica.

No obstante, es importante mencionar las limitaciones del modelo. La más importante y que puede pasar por alto más fácilmente es que se ha sacrificado parte de la rigurosidad teórica por utilizar una red neuronal y al final queda como un método aproximado. Otra limitante es que puede aplicarse sólo en elementos que fueron utilizados para entrenar las redes neuronales. Además, ninguna de las redes demostró reproducir cualitativamente b_{opt} en función de cambios en la geometría de la molécula ni lograron corregir discontinuidades en el método HF, como la discontinuidad en la disociación de C_2 o la cúspide en la rotación del etileno.

Respecto a los errores obtenidos en los distintos casos que involucran un cambio en la geometría de la molécula, una posible solución es utilizar el mismo valor fijo de b_{rn} obtenido para la geometría de mínima energía. Aunque esta alternativa demostró ser no tan buena como usar b_{rn} variable para la disociación de H_2 , demostró mejores resultados tanto en la rotación del etileno como en los isómeros de C_{60} . Dado que el error obtenido en estos últimos casos es mayor que el error obtenido en H_2 , se recomienda utilizar el método de b_{rn} fija para otros casos que involucren un

cambio en la geometría de la molécula.

La diferencia en la exactitud entre TT y TD se debe a que los hiperparámetros fueron calibrados únicamente para TT. Debido a los alcances de este trabajo, solo se realizó la calibración de hiperparámetros para TT porque es la red neuronal que más exactitud puede ofrecer; sin embargo, sería posible en un futuro calibrar TD con hiperparámetros distintos para mejorar su exactitud.

Códigos en Python

Implementación del procedimiento CTCE

```
1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3 # MAIN computes E_theta with CTCE procedure (Chapter 2)
4 #     varepsilon is an array with the orbital energies
5 #     mu_0=-0.5*abs(HOMO-LUMO) is trial value of mu
6 #     theta_0=5 is trial value of theta
7 #     N is the number of electrons
8 #     factor=1 if unrestricted-HF ; factor=2 if restricted-HF
9 #     consA=a/factor
10 #     b is value of b to use (b_opt or b_rn)
11 #     return E_theta
12
13 import numpy as np
14 from scipy.optimize import minimize
15
16 def f_i(e_i, mu, theta):
17     if (theta * (e_i - mu)) < -700:
18         return 1
19     elif (theta * (e_i - mu)) > 700:
20         return 0
21     else:
22         return 1 / (1 + np.exp(theta * (e_i - mu)))
23
24 def zeta_f(varepsilon, mu, theta, consA, b, N):
25
26     zeta = 0
27     for e_i in varepsilon:
28         n_i = f_i(e_i, mu, theta)
29         zeta += n_i * (1 - n_i) * (b * mu - e_i)
30     zeta = consA + zeta * theta / (2*N)
31
32     return zeta
33
34 def S_f(varepsilon, mu, theta):
35
36     S = 0
37     for e_i in varepsilon:
38         n_i = f_i(e_i, mu, theta)
39         if n_i == 1:
40             mi = 0
```

```

41     else:
42         mi = (1 - n_i) * np.log(1 - n_i)
43     if n_i == 0:
44         pi = 0
45     else:
46         pi = n_i * np.log(n_i)
47     S += pi + mi
48
49     return S
50
51 def min_mu(varepsilon, mu_0, theta, N, factor):
52
53     def error(mu):
54         N_calc = 0
55         for e_i in varepsilon:
56             N_calc += f_i(e_i, mu, theta)
57         N_calc *= factor
58         return abs(N - N_calc)
59
60     res = minimize(error, mu_0, method='Nelder-Mead',
61                   options={'xtol': 1e-3})
62     mu = float(res.x)
63
64     return mu
65
66 def min_F(varepsilon, mu_0, theta_0, N, consA, b, factor):
67
68     def F(theta):
69         mu = min_mu(varepsilon, mu_0, theta, N, factor)
70         zeta = zeta_f(varepsilon, mu, theta, consA, b, N)
71         S = S_f(varepsilon, mu, theta)
72         return abs(N*zeta - S)
73
74     res = minimize(F, theta_0, method='Nelder-Mead',
75                   options={'xtol': 1e-10})
76
77     # Optimized values
78     theta = float(res.x)
79     mu = min_mu(varepsilon, mu_0, theta, N, factor)
80
81     return mu, theta
82
83 def main(varepsilon, mu_0, theta_0, N, factor, consA, b):
84
85     # Optimize theta and mu
86     print(' {0} orbitals & {1} electrons loaded'.\
87           format(len(varepsilon), N))
88     print(' Loaded b: {0:.6f}'.format(b))
89     print(' Performing thermal optimization...')
90     mu, theta = min_F(varepsilon, mu_0, theta_0, N, consA, b, factor)
91
92     # Calculate convergences
93     N_calc = 0
94     for e_i in varepsilon:
95         N_calc += f_i(e_i, mu, theta)
96     N_calc *= factor
97     N_conver = abs(N - N_calc)
98     zeta = zeta_f(varepsilon, mu, theta, consA, b, N)

```

```

99     S = S_f(varepsilon, mu, theta)
100     S_conver = abs(N*zeta - S)
101
102     # Check convergences
103     print(' *****Thermal Optimization Complete*****')
104     print(' Convergence achieved in the electrons number {0:.2e}'.\
105           format(N_conver))
106     print(' Convergence achieved in the entropy per electron {0:.2e}'.\
107           format(S_conver))
108     print(' Optimized Mu {0:.6f} Optimized Theta {1:.6f}'.\
109           format(mu, theta))
110
111     # Corrected energy
112     E_corr = S * factor / theta
113     print(' PREDICTED CORRELATION {0:.6e}'.format(E_corr))
114
115     results = {'mu': mu,
116              'theta': theta,
117              'new_correlation': E_corr}
118
119     return results

```

Implementación del procedimiento CBO

```

1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3  # MAIN computes b_opt with CBO procedure (Chapter 2)
4  #     varepsilon is an array with the orbital energies
5  #     mu_0=-0.5*abs(HOMO-LUMO) is trial value of mu
6  #     theta_0=5 is trial value of theta
7  #     N is the number of electrons
8  #     factor=1 if unrestrictedHF ; factor=2 if restrictedHF
9  #     consA=a/factor
10 #     b_0=2 is trial value of b
11 #     E_corr is the correlation energy
12 #     return b_opt
13
14 import numpy as np
15 from scipy.optimize import minimize
16
17 def f_i(e_i, mu, theta):
18     if (theta * (e_i - mu)) < -700:
19         return 1
20     elif (theta * (e_i - mu)) > 700:
21         return 0
22     else:
23         return 1 / (1 + np.exp(theta * (e_i - mu)))
24
25 def zeta_f(varepsilon, mu, theta, consA, b, N):
26
27     zeta = 0
28     for e_i in varepsilon:
29         n_i = f_i(e_i, mu, theta)
30         zeta += n_i * (1 - n_i) * (b * mu - e_i)
31     zeta = consA + zeta * theta / (2*N)
32
33     return zeta

```

```

34
35 def S_f(varepsilon, mu, theta):
36
37     S = 0
38     for e_i in varepsilon:
39         n_i = f_i(e_i, mu, theta)
40         if n_i == 1:
41             mi = 0
42         else:
43             mi = (1 - n_i) * np.log(1 - n_i)
44         if n_i == 0:
45             pi = 0
46         else:
47             pi = n_i * np.log(n_i)
48         S += pi + mi
49
50     return S
51
52 def min_mu(varepsilon, mu_0, theta, N, factor):
53
54     def error(mu):
55         N_calc = 0
56         for e_i in varepsilon:
57             N_calc += f_i(e_i, mu, theta)
58         N_calc *= factor
59         return abs(N - N_calc)
60
61     res = minimize(error, mu_0, method='Nelder-Mead',
62                   options={'xtol': 1e-3})
63     mu = float(res.x)
64
65     return mu
66
67 def min_G(varepsilon, mu_0, theta_0, N, E_corr, factor):
68
69     def G(theta):
70         mu = min_mu(varepsilon, mu_0, theta, N, factor)
71         S = S_f(varepsilon, mu, theta)
72         return abs(E_corr - S * factor / theta)
73
74     res = minimize(G, theta_0, method='Nelder-Mead',
75                   options={'xtol': 1e-10})
76
77     # Optimized values
78     theta = float(res.x)
79     mu = min_mu(varepsilon, mu_0, theta, N, factor)
80
81     return mu, theta
82
83 def min_E(varepsilon, mu, theta, E_corr, consA, b_0, N, factor):
84
85     def E(b):
86         zeta = zeta_f(varepsilon, mu, theta, consA, b, N)
87         return abs(E_corr - zeta * factor * N / theta)
88
89     res = minimize(E, b_0, method='Nelder-Mead',
90                   options={'xtol': 1e-6})
91     b_opt = float(res.x)

```

```

92
93     return b_opt
94
95 def main(varepsilon, mu_0, theta_0, N, factor, consA, b_0, E_corr):
96
97     # Optimize theta and mu
98     print(' {0} orbitals & {1} electrons loaded'.\
99           format(len(varepsilon), N))
100    print(' Performing thermal optimization...')
101    mu, theta = min_G(varepsilon, mu_0, theta_0, N, E_corr, factor)
102
103    # Optimize b
104    b_opt = min_E(varepsilon, mu, theta, E_corr, consA, b_0, N, factor)
105
106    # Calculate convergences
107    N_calc = 0
108    for e_i in varepsilon:
109        N_calc += f_i(e_i, mu, theta)
110    N_calc *= factor
111    N_conver = abs(N - N_calc)
112    zeta = zeta_f(varepsilon, mu, theta, consA, b_opt, N)
113    S = S_f(varepsilon, mu, theta)
114    S_conver = abs(N*zeta - S)
115
116    # Check convergences
117    print(' *****Thermal Optimization Complete*****')
118    print(' Convergence achieved in the electrons number {0:.2e}'.\
119          format(N_conver))
120    print(' Convergence achieved in the entropy per electron {0:.2e}'.\
121          format(S_conver))
122    print(' Optimized Mu {0:.6f} Optimized Theta {1:.6f}'.\
123          format(mu, theta))
124    print(' Optimized b {0:.6f}'.format(b_opt))
125
126    # Calculated energy
127    E_theta = S * factor / theta
128    print(' CALCULATED CORRELATION {0:.6e}'.format(E_theta))
129    assert abs(E_corr - E_theta) <= 1e-4, ' E_corr= {0:.6e}'.\
130          format(E_corr)
131
132    results = {'mu': mu,
133              'theta': theta,
134              'new_correlation': E_theta,
135              'b_optimized': b_opt}
136
137    return results

```

Implementación del proceso de entrenamiento de una red neuronal utilizado en este trabajo

```

1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3 # MAIN trains a neural network (NN) model with the given X and y data
4 # and returns the trained model, mu and sigma.
5 #     X is a matrix with the input data, where the i-th row represents
6 #     the i-th system and each column represents a feature (ET, EK,

```

```

7 # HOMO, LUMO). Therefore, X is a Mx4 matrix being M the number of
8 # systems.
9 # y is a vector where the i-th element represents b_opt of the
10 # i-th system. Therefore, y is a vector of M dimension being M
11 # the number of systems.
12 # model is the NN trained
13 # mu and sigma are the mean and the standard deviation of the
14 # training data
15
16 import numpy as np
17 import matplotlib.pyplot as plt
18
19 import tensorflow as tf
20 from tensorflow import keras
21 from tensorflow.keras import layers
22
23 def main(X_train, y_train, X_val, y_val, X_test, y_test):
24
25     print('\n DIMENSION OF THE TRAINING DATA:')
26     print(' X= {0} \t y= {1}'.format(X_train.shape, y_train.shape))
27
28     # Feature scaling
29     mu = np.mean(X_train, axis=0)           # mean
30     sigma = np.std(X_train, axis=0)        # standard deviation
31     X_train = (X_train - mu) / sigma
32     X_val = (X_val - mu) / sigma
33     X_test = (X_test - mu) / sigma
34
35     # Hyperparameters
36     input_size = X_train.shape[1]
37     hidden_size = 8
38     output_size = 1
39     epochs = 1000
40     batch = 10
41
42     # Create NN
43     print('\n TRAINING NN WITH')
44     print(' {0} INPUTS; {1} NEURONS; {2} OUTPUTS...'.format(input_size,
45         hidden_size, output_size))
46     model = keras.Sequential()
47
48     # Seed to generate always same random numbers
49     seed = 2
50     np.random.seed(seed)
51     tf.set_random_seed(seed)
52
53     # Hidden layer
54     model.add(layers.Dense(hidden_size, input_dim=input_size,
55         activation='sigmoid'))
56
57     # Output layer
58     model.add(layers.Dense(output_size,
59         activation='selu'))
60
61     # Training
62     model.compile(loss='mse', optimizer='nadam')
63     history = model.fit(X_train, y_train, epochs=epochs, batch_size=batch,
64         validation_data=(X_val, y_val), verbose=1)

```

```
65 print('\n TRAINING COMPLETED \n J= {0:.6f}'.\
66       format(history.history['loss'][-1]))
67
68 # Calculate accuracy
69 X_total = np.concatenate([X_train, X_val, X_test], axis=0)
70 y_total = np.concatenate([y_train, y_val, y_test], axis=0)
71 p = model.predict(X_total)
72 precs = np.mean(np.abs( (p - y_total)*100 / y_total ))
73 print(' ACCURACY/ERROR: {0:.2f}% \n'.format(precs) )
74
75 # Plot history J vs epochs
76 plt.plot(history.history['loss'])
77 plt.plot(history.history['val_loss'])
78 plt.ylabel('J')
79 plt.xlabel('Epoch')
80 plt.legend(['train', 'validation'], loc='upper right')
81 plt.show()
82
83 return model, mu, sigma
```

Sistemas para la red neuronal TT

A continuación se presenta la información de los 179 sistemas utilizados para el entrenamiento de la red neuronal *Thermal-Triple*.

Notas:

- La columna **Sistema** contiene el nombre de los sistemas, respetando el orden presentado en la Sección 4.2. Algunos nombres están abreviados para que todas las columnas de la tabla se ajusten en una misma página.
- Todos los valores numéricos están en unidades atómicas.
- Las energías E_{HF} y sus componentes fueron calculados al nivel de teoría HF/6-311+G(3df,2p).
- Las energías E_{QCI} fueron calculadas al nivel de teoría *full* QCISD(T)/6-311+G(3df,2p).
- Los valores b_{opt} fueron calculados con $a = -0.140$, utilizando el procedimiento CBO.
- Los valores b_{rn} son los predichos por la red neuronal, los valores E_{θ} son las energías de correlación calculadas con b_{rn} utilizando el procedimiento CTCE.
- La columna **SUB** indica a cuál de los 23 subconjuntos pertenece el sistema (Sección 4.2).
- La columna **RN** indica el conjunto de entrenamiento: *TRAIN* para entrenamiento, *CV* para validación y *TEST* para prueba.
- Para los subconjuntos **DS1** y **DS2**, los valores entre paréntesis son las distancias de enlace en angstroms.

Índice	Sistema	N	E_T	E_K	ϵ_{HOMO}	ϵ_{LUMO}	E_{HF}	$E_{corr,QCI}$	b_{opt}	b_{rn}	E_θ	SUB	RN
1	Al	13	241.823	-18.090	-0.218	0.014	-241.874	-0.210	1.680	1.824	-0.239	A0	TRAIN
2	Ar	18	526.789	-30.183	-0.591	0.139	-526.807	-0.326	1.128	1.084	-0.321	A0	TRAIN
3	As	33	2233.790	-81.504	-0.370	0.025	-2234.157	-0.487	0.585	0.526	-0.471	A0	TEST
4	B	5	24.535	-3.769	-0.319	0.030	-24.531	-0.084	1.994	2.391	-0.131	A0	TEST
5	Be	4	14.569	-2.666	-0.309	0.030	-14.572	-0.062	2.175	2.428	-0.088	A0	TRAIN
6	Br	35	2571.894	-89.642	-0.434	-0.050	-2572.357	-0.566	0.589	0.659	-0.586	A0	TRAIN
7	C	6	37.696	-5.075	-0.439	0.019	-37.690	-0.109	1.942	2.497	-0.180	A0	TRAIN
8	Ca	20	676.760	-35.211	-0.196	0.010	-676.740	-0.424	1.682	1.579	-0.347	A0	TRAIN
9	Cl	17	459.498	-27.541	-0.481	-0.042	-459.477	-0.278	1.308	1.748	-0.352	A0	CV
10	F	9	99.384	-10.037	-0.681	0.040	-99.402	-0.239	2.085	2.142	-0.246	A0	TEST
11	He	2	2.860	-1.026	-0.917	0.775	-2.860	-0.036	-2.278	-1.472	-0.039	A0	TRAIN
12	Kr	36	2751.569	-93.845	-0.525	0.091	-2751.966	-0.575	0.242	0.225	-0.572	A0	TRAIN
13	Mg	12	199.481	-15.992	-0.253	0.021	-199.607	-0.180	1.739	1.892	-0.219	A0	TRAIN
14	N	7	54.405	-6.604	-0.571	0.088	-54.399	-0.135	1.796	2.518	-0.215	A0	TRAIN
15	Ne	10	128.486	-12.098	-0.853	0.280	-128.527	-0.297	2.071	1.981	-0.289	A0	TRAIN
16	O	8	74.806	-8.212	-0.522	0.073	-74.809	-0.185	1.947	2.484	-0.253	A0	TRAIN
17	P	15	340.717	-22.641	-0.392	0.031	-340.709	-0.270	1.698	1.817	-0.294	A0	TRAIN
18	Biciclobutano	30	154.904	-22.946	-0.361	0.066	-154.923	-0.816	2.512	2.513	-0.817	AGD	TEST
19	Cubano	56	307.449	-44.517	-0.383	0.064	-307.481	-1.584	2.688	2.530	-1.431	AGD	TRAIN
20	Tetraedrano	28	153.629	-22.198	-0.339	0.064	-153.651	-0.786	2.508	2.445	-0.751	AGD	TRAIN
21	As_am	33	2233.705	-81.445	-0.292	-0.044	-2234.074	-0.509	0.796	0.771	-0.500	AM	TRAIN
22	C_am	6	37.604	-5.006	-0.358	0.021	-37.601	-0.144	2.323	2.431	-0.164	AM	TEST
23	O_am	8	74.671	-8.128	-0.587	-0.023	-74.682	-0.230	2.392	2.273	-0.211	AM	TRAIN
24	P_am	15	340.629	-22.580	-0.311	-0.040	-340.623	-0.293	1.870	2.083	-0.357	AM	TRAIN
25	S_am	16	397.410	-24.974	-0.406	-0.074	-397.419	-0.341	1.962	2.061	-0.371	AM	TRAIN
26	Se_am	34	2399.231	-85.483	-0.374	-0.077	-2399.713	-0.544	0.912	0.802	-0.505	AM	CV
27	C+	5	37.298	-4.745	-0.914	-0.362	-37.293	-0.094	1.266	1.511	-0.145	AP	CV
28	N+	6	53.897	-6.212	-1.116	-0.461	-53.888	-0.114	1.177	1.266	-0.128	AP	TRAIN
29	O+	7	74.378	-7.912	-1.331	-0.383	-74.367	-0.135	0.975	1.388	-0.205	AP	TEST
30	P+	14	340.365	-22.368	-0.725	-0.339	-340.345	-0.251	1.162	1.361	-0.339	AP	TRAIN

Índice	Sistema	N	E_T	E_K	ε_{HOMO}	ε_{LUMO}	E_{HF}	$E_{corr,QCI}$	b_{opt}	b_m	E_θ	SUB	RN
31	S+	15	397.174	-24.791	-0.862	-0.307	-397.164	-0.275	1.090	0.920	-0.230	AP	TRAIN
32	Benceno	42	230.770	-33.331	-0.339	0.069	-230.775	-1.161	2.474	2.444	-1.138	CMOL	CV
33	13Butadieno	30	154.963	-22.880	-0.329	0.068	-154.976	-0.804	2.421	2.446	-0.817	CMOL	TRAIN
34	Etano	18	79.254	-12.534	-0.488	0.074	-79.258	-0.454	2.478	2.636	-0.505	CMOL	TRAIN
35	Eteno	16	77.958	-11.753	-0.372	0.081	-78.061	-0.418	2.391	2.558	-0.471	CMOL	TRAIN
36	Etino	14	76.833	-11.014	-0.417	0.069	-76.849	-0.381	2.448	2.589	-0.430	CMOL	TEST
37	H2	2	1.130	-0.661	-0.596	0.169	-1.133	-0.038	2.214	2.127	-0.036	CMOL	CV
38	Metano	10	40.209	-6.604	-0.547	0.073	-40.212	-0.244	2.422	2.568	-0.269	CMOL	TRAIN
39	Propano	26	118.297	-18.462	-0.469	0.073	-118.305	-0.667	2.515	2.641	-0.724	CMOL	TRAIN
40	Propeno	24	117.106	-17.706	-0.363	0.070	-117.114	-0.629	2.432	2.539	-0.676	CMOL	TRAIN
41	Propino	22	115.886	-16.952	-0.390	0.068	-115.907	-0.591	2.444	2.564	-0.646	CMOL	TRAIN
42	Bencilo+	48	268.954	-38.612	-0.542	-0.191	-268.967	-1.320	1.792	2.218	-2.121	CYCP	TRAIN
43	C3H9C+	32	156.490	-23.746	-0.732	-0.154	-156.498	-0.825	1.772	1.809	-0.855	CYCP	TEST
44	CH5CHCH3+	32	156.469	-23.738	-0.692	-0.171	-156.475	-0.824	1.771	1.864	-0.906	CYCP	CV
45	Ciclobutano	32	156.133	-23.680	-0.435	0.078	-156.147	-0.848	2.562	2.626	-0.885	CYCP	CV
46	CiclohexBote	48	234.252	-35.567	-0.425	0.068	-234.270	-1.278	2.606	2.616	-1.286	CYCP	TRAIN
47	CiclohexSilla	48	234.262	-35.570	-0.425	0.068	-234.281	-1.277	2.598	2.615	-1.291	CYCP	TRAIN
48	Ciclopentano	40	195.210	-29.631	-0.450	0.073	-195.225	-1.062	2.591	2.629	-1.088	CYCP	TRAIN
49	Ciclopropano	24	117.086	-17.750	-0.421	0.084	-117.099	-0.633	2.502	2.616	-0.685	CYCP	TRAIN
50	C2(1.83)	12	74.755	-9.887	-0.353	-0.044	-75.340	-0.337	2.409	2.383	-0.327	DS1	TRAIN
51	H2(1.54)	2	0.722	-0.470	-0.435	0.025	-0.997	-0.063	3.078	2.928	-0.055	DS1	TRAIN
52	H2(2.90)	2	0.667	-0.354	-0.333	-0.116	-0.838	-0.163	2.910	2.817	-0.148	DS1	TEST
53	He2+(1.07)	3	4.927	-1.587	-1.486	-0.638	-4.930	-0.052	1.043	0.977	-0.048	DS1	TRAIN
54	He2+(2.16)	3	4.771	-1.462	-1.119	-0.834	-4.853	-0.058	1.244	0.923	-0.016	DS1	CV
55	He2+(4.0)	3	4.833	-1.434	-0.991	-0.858	-4.835	-0.060	1.222	1.199	-0.052	DS1	TRAIN
56	Be2(2.17)	8	29.244	-5.365	-0.230	0.003	-29.121	-0.146	2.198	2.171	-0.140	DS2	TRAIN
57	Be2(5.23)	8	29.138	-5.333	-0.305	0.024	-29.144	-0.124	2.189	2.412	-0.167	DS2	TEST
58	C2(1.26)	12	75.313	-10.022	-0.454	-0.116	-75.399	-0.397	2.523	2.385	-0.340	DS2	TRAIN
59	C2(4.26)	12	74.712	-9.530	-0.249	-0.124	-74.956	-0.566	2.577	2.331	-0.386	DS2	TRAIN
60	F2(1.84)	18	198.100	-19.760	-0.626	-0.124	-198.641	-0.628	2.604	2.111	-0.470	DS2	TRAIN

Índice	Sistema	N	E_T	E_K	ϵ_{HOMO}	ϵ_{LUMO}	E_{HF}	$E_{corr,QCI}$	b_{opt}	b_{rn}	E_θ	SUB	RN
61	F2(3.1)	18	198.202	-19.631	-0.484	-0.282	-198.458	-0.752	2.698	2.533	-0.671	DS2	TRAIN
62	H2(2.17)	2	0.661	-0.398	-0.375	-0.060	-0.904	-0.109	2.988	2.966	-0.106	DS2	TRAIN
63	H2(3.7)	2	0.697	-0.331	-0.305	-0.148	-0.798	-0.202	2.926	2.695	-0.166	DS2	TRAIN
64	He2+(3.73)	3	4.832	-1.437	-1.000	-0.857	-4.835	-0.060	1.225	1.178	-0.044	DS2	TEST
65	3_C2	12	75.658	-10.189	-0.431	-0.050	-75.492	-0.323	2.335	2.407	-0.347	EM	TRAIN
66	C2	12	75.383	-10.040	-0.459	-0.116	-75.400	-0.396	2.524	2.381	-0.338	EM	TRAIN
67	H2Oclust2	20	152.050	-17.915	-0.486	0.122	-152.123	-0.596	2.871	2.625	-0.532	H2OC	CV
68	H2Oclust3	30	228.088	-26.885	-0.504	0.132	-228.192	-0.898	2.944	2.612	-0.779	H2OC	TRAIN
69	H2Oclust4	40	304.106	-35.855	-0.494	0.131	-304.263	-1.198	3.031	2.616	-1.012	H2OC	CV
70	H2Oclust5	50	380.152	-44.821	-0.501	0.118	-380.334	-1.501	3.014	2.627	-1.281	H2OC	TEST
71	H2Sclust2	36	797.377	-51.453	-0.376	0.061	-797.418	-0.765	1.926	1.810	-0.719	H2SC	TRAIN
72	H2Sclust3anti	54	1196.082	-77.185	-0.384	0.056	-1196.126	-1.153	1.933	1.824	-1.087	H2SC	TRAIN
73	H2Sclust3syn	54	1196.082	-77.184	-0.385	0.055	-1196.126	-1.153	1.932	1.826	-1.089	H2SC	TRAIN
74	H2Sclust4Ci	72	1594.780	-102.914	-0.384	0.054	-1594.836	-1.538	1.941	1.830	-1.451	H2SC	TRAIN
75	H2Sclust4S4	72	1594.781	-102.914	-0.385	0.052	-1594.835	-1.539	1.941	1.833	-1.454	H2SC	TRAIN
76	CCl4	74	1876.005	-115.436	-0.460	0.065	-1875.890	-1.424	1.473	1.577	-1.493	HALO	TRAIN
77	CHCl3	58	1417.062	-88.238	-0.451	0.067	-1416.981	-1.122	1.540	1.637	-1.174	HALO	TRAIN
78	HCl	18	460.128	-27.895	-0.478	0.094	-460.100	-0.320	1.442	1.507	-0.329	HALO	TRAIN
79	HF	10	100.019	-10.433	-0.653	0.119	-100.057	-0.305	2.637	2.414	-0.277	HALO	TRAIN
80	H3CH-H2O	20	116.261	-15.567	-0.513	0.080	-116.271	-0.540	2.631	2.625	-0.538	HB	TRAIN
81	NH3-H2O	20	132.272	-16.655	-0.451	0.084	-132.285	-0.574	2.784	2.635	-0.531	HB	TRAIN
82	Pirrolidina	40	211.197	-30.706	-0.370	0.069	-211.217	-1.093	2.597	2.537	-1.053	HETC	TRAIN
83	Piperidina	48	250.247	-36.647	-0.351	0.067	-250.272	-1.307	2.598	2.500	-1.233	HETC	TRAIN
84	22Metiocicb	56	592.725	-54.663	-0.332	0.065	-592.750	-1.425	2.387	2.421	-1.453	HETC	TRAIN
85	3Etilazacicb	48	250.210	-36.615	-0.364	0.064	-250.236	-1.307	2.609	2.525	-1.241	HETC	TEST
86	2Fluotiocicp	40	574.443	-46.614	-0.368	0.064	-574.490	-1.048	2.180	2.221	-1.074	HETC	TRAIN
87	Furano	36	228.664	-30.474	-0.326	0.067	-228.706	-1.041	2.446	2.384	-1.000	HETC	TRAIN
88	Indol	62	361.561	-50.777	-0.288	0.054	-361.584	-1.770	2.534	2.281	-1.521	HETC	TEST
89	3Meoxacicb	56	309.117	-43.847	-0.397	0.064	-309.156	-1.541	2.612	2.570	-1.503	HETC	TRAIN
90	NMeazacicp	32	172.112	-24.740	-0.371	0.072	-172.130	-0.878	2.545	2.538	-0.874	HETC	TRAIN

Índice	Sistema	N	E_T	E_K	ϵ_{HOMO}	ϵ_{LUMO}	E_{HF}	$E_{corr,QCI}$	b_{opt}	b_{rrn}	E_θ	SUB	RN
91	Oxaciclob	32	191.949	-26.029	-0.410	0.072	-191.979	-0.895	2.557	2.581	-0.908	HETC	TEST
92	Oxaciclop	24	152.899	-20.079	-0.451	0.075	-152.926	-0.684	2.532	2.609	-0.719	HETC	TRAIN
93	Pirrol	36	208.863	-29.220	-0.302	0.059	-208.883	-1.022	2.487	2.335	-0.927	HETC	TEST
94	Quinoleina	68	399.445	-55.949	-0.316	0.058	-399.471	-1.941	2.524	2.361	-1.754	HETC	TRAIN
95	Tiociclop	48	553.705	-48.760	-0.329	0.066	-553.725	-1.205	2.338	2.374	-1.231	HETC	TRAIN
96	Tiofeno	44	551.363	-47.272	-0.334	0.064	-551.374	-1.136	2.274	2.299	-1.154	HETC	CV
97	2Indanona	70	420.439	-57.965	-0.336	0.058	-420.480	-1.995	2.518	2.413	-1.872	HETC	TRAIN
98	OxIndeno	70	420.388	-57.977	-0.328	0.062	-420.428	-2.006	2.562	2.393	-1.813	HETC	CV
99	Dimetilam	26	134.279	-19.540	-0.365	0.069	-134.292	-0.697	2.528	2.543	-0.704	HETM	CV
100	Metilfosf	26	381.532	-29.629	-0.357	0.056	-381.535	-0.578	2.069	2.262	-0.652	HETM	TRAIN
101	Dimetilfosf	34	420.587	-35.564	-0.339	0.058	-420.588	-0.793	2.176	2.362	-0.889	HETM	TRAIN
102	AcMefosf	50	606.300	-54.685	-0.458	0.054	-606.349	-1.362	2.379	2.415	-1.388	HETM	TRAIN
103	AcMesulf	50	662.358	-56.681	-0.490	0.053	-662.413	-1.391	2.341	2.357	-1.403	HETM	TEST
104	Dietilamina	42	212.374	-31.397	-0.363	0.072	-212.391	-1.124	2.576	2.538	-1.099	HETM	TEST
105	H3O+	10	76.347	-9.056	-0.957	-0.140	-76.337	-0.291	1.577	1.664	-0.315	HETM	TRAIN
106	H3PO4	50	642.161	-57.083	-0.477	0.050	-642.224	-1.408	2.382	2.335	-1.376	HETM	CV
107	H3S+	18	398.989	-25.835	-0.784	-0.132	-398.990	-0.378	1.257	1.234	-0.370	HETM	TRAIN
108	PH3	18	342.424	-23.667	-0.303	0.068	-342.427	-0.368	1.828	1.883	-0.381	HETM	CV
109	PH4+	18	342.802	-23.845	-0.841	-0.118	-342.804	-0.353	1.241	1.402	-0.408	HETM	TRAIN
110	Benz_Ne	52	358.943	-45.357	-0.337	0.069	-359.300	-1.462	2.412	2.408	-1.458	NCOV	TRAIN
111	C2H2_C2H2	28	153.312	-21.947	-0.406	0.064	-153.695	-0.770	2.509	2.578	-0.814	NCOV	CV
112	C2H2_C1F	40	635.430	-48.545	-0.429	0.065	-635.738	-0.990	2.112	2.207	-1.048	NCOV	TEST
113	C2H4_C2H4	32	155.863	-23.491	-0.377	0.071	-156.123	-0.838	2.431	2.570	-0.923	NCOV	TRAIN
114	C2H4_F2	34	276.419	-31.685	-0.379	0.074	-276.803	-1.016	2.435	2.483	-1.045	NCOV	TRAIN
115	CH3Cl_HCl	44	959.230	-61.705	-0.453	0.062	-959.242	-0.856	1.697	1.833	-0.915	NCOV	CV
116	CH3SH_HCl	44	897.800	-59.539	-0.379	0.060	-897.853	-0.921	1.888	1.927	-0.940	NCOV	TEST
117	CH4_CH4	20	80.337	-13.187	-0.537	0.066	-80.424	-0.489	2.465	2.584	-0.528	NCOV	TEST
118	CH4_Ne	20	168.649	-18.692	-0.545	0.074	-168.739	-0.541	2.297	2.564	-0.631	NCOV	TRAIN
119	H2O_C1F	36	634.733	-46.511	-0.472	0.074	-634.953	-0.902	2.142	2.058	-0.869	NCOV	CV
120	H2O_H2O	20	151.943	-17.885	-0.481	0.120	-152.121	-0.598	2.897	2.628	-0.528	NCOV	TRAIN

Índice	Sistema	N	E_T	E_K	ϵ_{HOMO}	ϵ_{LUMO}	E_{HF}	$E_{corr,QCI}$	b_{opt}	b_{rrn}	E_θ	SUB	RN
121	H2S_H2S	36	797.333	-51.442	-0.377	0.061	-797.417	-0.766	1.929	1.812	-0.719	NCOV	TRAIN
122	H2S_HcI	36	858.781	-53.615	-0.407	0.058	-858.810	-0.706	1.714	1.729	-0.711	NCOV	TRAIN
123	HCN_CH3SH	40	530.361	-43.684	-0.385	0.056	-530.660	-1.012	2.266	2.375	-1.086	NCOV	TRAIN
124	HCN_CIF	40	651.445	-49.614	-0.466	0.050	-651.801	-1.019	2.165	2.160	-1.016	NCOV	TRAIN
125	HCONH2_dimer	48	337.509	-42.250	-0.412	0.065	-338.018	-1.432	2.659	2.544	-1.340	NCOV	TRAIN
126	HCOOH_dimer	48	377.135	-44.777	-0.477	0.067	-377.699	-1.474	2.591	2.558	-1.446	NCOV	CV
127	HCl_HCl	36	920.219	-55.784	-0.469	0.074	-920.201	-0.643	1.424	1.552	-0.680	NCOV	TRAIN
128	HF_HF	20	199.941	-20.848	-0.627	0.091	-200.119	-0.613	2.676	2.407	-0.551	NCOV	TRAIN
129	NH3_Cl2	44	975.103	-62.747	-0.423	0.053	-975.204	-0.895	1.802	1.828	-0.906	NCOV	TRAIN
130	NH3_CIF	36	614.912	-45.243	-0.456	0.070	-615.114	-0.888	2.192	2.131	-0.863	NCOV	TRAIN
131	NH3_F2	28	254.633	-27.603	-0.436	0.085	-254.959	-0.876	2.663	2.535	-0.824	NCOV	TRAIN
132	NH3_H2O	20	132.121	-16.613	-0.456	0.082	-132.283	-0.577	2.822	2.637	-0.525	NCOV	TRAIN
133	NH3_NH3	20	112.291	-15.334	-0.406	0.087	-112.439	-0.554	2.756	2.615	-0.514	NCOV	TEST
134	NH3clust2asym	20	112.336	-15.347	-0.407	0.088	-112.440	-0.554	2.746	2.615	-0.516	NH3C	TRAIN
135	NH3clust3equi	30	168.487	-23.020	-0.408	0.090	-168.664	-0.836	2.848	2.615	-0.743	NH3C	TRAIN
136	NH3clust4boat	40	224.654	-30.697	-0.398	0.090	-224.887	-1.116	2.917	2.601	-0.959	NH3C	TEST
137	NH3clust5ring	50	280.615	-38.313	-0.398	0.087	-281.107	-1.397	2.952	2.604	-1.186	NH3C	TRAIN
138	Etilamina	26	134.289	-19.541	-0.385	0.069	-134.303	-0.698	2.551	2.576	-0.709	NMOL	TRAIN
139	HNO3	32	279.476	-31.634	-0.496	0.072	-279.549	-1.050	2.623	2.524	-0.994	NMOL	TRAIN
140	Metilamina	18	95.242	-13.615	-0.388	0.068	-95.254	-0.485	2.541	2.582	-0.497	NMOL	TRAIN
141	N2	14	108.967	-13.150	-0.629	0.140	-108.985	-0.431	2.448	2.462	-0.435	NMOL	CV
142	NH3	10	56.217	-7.686	-0.426	0.100	-56.219	-0.275	2.634	2.624	-0.273	NMOL	TRAIN
143	NH4+	10	56.566	-7.806	-1.008	-0.131	-56.563	-0.269	1.599	1.517	-0.250	NMOL	TRAIN
144	Piridina	42	246.757	-34.402	-0.350	0.062	-246.775	-1.191	2.480	2.453	-1.169	NMOL	TRAIN
145	AcAcetico	32	227.856	-28.367	-0.455	0.064	-227.903	-0.940	2.543	2.574	-0.958	OMOL	CV
146	Acetona	32	192.003	-25.991	-0.411	0.063	-192.033	-0.890	2.497	2.576	-0.936	OMOL	TRAIN
147	DimetilEter	26	154.102	-20.810	-0.422	0.071	-154.126	-0.715	2.536	2.606	-0.747	OMOL	TRAIN
148	Etanol	26	154.119	-20.816	-0.441	0.067	-154.142	-0.717	2.584	2.616	-0.730	OMOL	TEST
149	Formaldehido	16	113.886	-14.124	-0.440	0.067	-113.913	-0.466	2.447	2.575	-0.509	OMOL	TRAIN
150	AcFormico	24	188.799	-22.431	-0.474	0.069	-188.843	-0.728	2.527	2.560	-0.743	OMOL	TEST

Índice	Sistema	N	E_T	E_K	ϵ_{HOMO}	ϵ_{LUMO}	E_{HF}	$E_{corr:QCI}$	b_{opt}	b_{rrn}	E_θ	SUB	RN
151	H2O	10	76.052	-8.962	-0.512	0.147	-76.058	-0.296	2.801	2.583	-0.266	OMOL	CV
152	H2O2	18	150.816	-17.150	-0.481	0.145	-150.837	-0.562	2.847	2.582	-0.496	OMOL	TRAIN
153	Metanol	18	115.070	-14.887	-0.450	0.069	-115.090	-0.504	2.574	2.616	-0.517	OMOL	TRAIN
154	O2	16	149.562	-16.329	-0.451	0.041	-149.592	-0.529	2.499	2.443	-0.512	OMOL	CV
155	BenzCH2rad	49	269.211	-38.956	-0.312	0.061	-269.222	-1.325	2.461	2.370	-1.251	RAD	TRAIN
156	C3H9Crad	33	156.719	-24.023	-0.316	0.069	-156.728	-0.844	2.462	2.437	-0.831	RAD	TRAIN
157	CH5CHCH3rad	33	156.716	-24.023	-0.330	0.072	-156.724	-0.842	2.466	2.471	-0.845	RAD	TRAIN
158	CH3rad	9	39.575	-6.236	-0.384	0.071	-39.577	-0.203	2.180	2.605	-0.274	RAD	TRAIN
159	OHrad	9	75.413	-8.573	-0.515	0.071	-75.419	-0.240	2.419	2.555	-0.259	RAD	TEST
160	PH2rad	17	341.883	-23.335	-0.376	0.022	-341.883	-0.332	1.828	1.985	-0.368	RAD	CV
161	SHrad	17	398.095	-25.377	-0.381	-0.003	-398.100	-0.343	1.815	1.927	-0.368	RAD	TRAIN
162	Propionitrilo	28	169.799	-23.200	-0.405	0.054	-169.826	-0.794	2.432	2.539	-0.859	RESO	TRAIN
163	3Buten2ona	38	229.854	-31.165	-0.392	0.061	-229.888	-1.066	2.486	2.542	-1.105	RESO	TRAIN
164	Ciclohexeno	46	233.077	-34.818	-0.344	0.066	-233.094	-1.238	2.556	2.483	-1.183	RESO	CV
165	Ciclopentadieno	36	192.847	-28.123	-0.312	0.069	-192.855	-0.988	2.487	2.385	-0.924	RESO	TRAIN
166	Ciclopenteno	38	194.026	-28.878	-0.343	0.070	-194.040	-1.023	2.537	2.480	-0.988	RESO	TRAIN
167	AnhidMaleico	50	377.290	-46.312	-0.452	0.029	-377.368	-1.519	2.424	2.485	-1.580	RESO	CV
168	Tetracianet	64	444.973	-57.428	-0.454	-0.053	-445.058	-1.945	2.380	2.405	-1.981	RESO	TRAIN
169	H2O_MeOH	28	190.925	-23.801	-0.474	0.062	-191.152	-0.807	2.703	2.607	-0.766	ROHC	TRAIN
170	MeOHH2Oclust2	38	266.921	-32.760	-0.452	0.075	-267.220	-1.112	2.817	2.616	-1.006	ROHC	TRAIN
171	MeOHH2Oclust3	64	420.855	-53.543	-0.445	0.071	-421.352	-1.837	2.763	2.614	-1.701	ROHC	TEST
172	MeOH_H2O	28	190.926	-23.801	-0.426	0.082	-191.151	-0.807	2.739	2.606	-0.752	ROHC	TRAIN
173	MeOHclust2	36	229.892	-29.713	-0.425	0.062	-230.183	-1.016	2.652	2.595	-0.984	ROHC	TRAIN
174	MeOHclust3	54	344.852	-44.582	-0.443	0.069	-345.282	-1.531	2.694	2.614	-1.465	ROHC	CV
175	DimetilSulfuro	34	476.790	-37.591	-0.335	0.069	-476.796	-0.808	2.187	2.276	-0.852	SMOL	TRAIN
176	Metanotiol	26	437.745	-31.661	-0.357	0.068	-437.752	-0.593	2.090	2.128	-0.607	SMOL	TRAIN
177	H2SO4	50	698.198	-59.065	-0.500	0.054	-698.261	-1.437	2.337	2.269	-1.390	SMOL	TRAIN
178	SH2	18	398.700	-25.729	-0.385	0.074	-398.709	-0.381	1.911	1.785	-0.355	SMOL	TRAIN
179	SO2	32	547.246	-41.596	-0.496	0.022	-547.298	-0.890	2.160	2.048	-0.834	SMOL	TRAIN

Sistemas para la red neuronal TD

A continuación se presenta la información de los 179 sistemas utilizados para el entrenamiento de la red neuronal *Thermal-Double*.

Notas:

- La columna **Sistema** contiene el nombre de los sistemas, respetando el orden presentado en la Sección 4.2. Algunos nombres están abreviados para que todas las columnas de la tabla se ajusten en una misma página.
- Todos los valores numéricos están en unidades atómicas.
- Las energías E_{HF} y sus componentes fueron calculados al nivel de teoría HF/6-31+G(d,p).
- Las energías E_{QCI} fueron calculadas al nivel de teoría *full* QCISD(T)/6-31+G(d,p).
- Los valores b_{opt} fueron calculados con $a = -0.099$, utilizando el procedimiento CBO.
- Los valores b_{rn} son los predichos por la red neuronal, los valores E_{θ} son las energías de correlación calculadas con b_{rn} utilizando el procedimiento CTCE.
- La columna **SUB** indica a cuál de los 23 subconjuntos pertenece el sistema (Sección 4.2).
- La columna **RN** indica el conjunto de entrenamiento: *TRAIN* para entrenamiento, *CV* para validación y *TEST* para prueba.
- Para los subconjuntos **DS1** y **DS2**, los valores entre paréntesis son las distancias de enlace en angstroms.

Índice	Sistema	N	E_T	E_K	ε_{HOMO}	ε_{LUMO}	E_{HF}	$E_{corr,QCI}$	b_{opt}	b_{rn}	E_θ	SUB	RN
1	Al	13	241.775	-18.086	-0.214	0.014	-241.857	-0.055	0.550	0.879	-0.067	A0	TRAIN
2	Ar	18	526.463	-30.175	-0.592	0.123	-526.775	-0.161	0.441	0.943	-0.188	A0	TRAIN
3	As	33	2226.082	-81.310	-0.371	0.024	-2231.988	-0.104	-2.170	-1.444	-0.126	A0	TEST
4	B	5	24.580	-3.770	-0.313	0.029	-24.526	-0.062	2.175	2.808	-0.144	A0	TEST
5	Be	4	14.637	-2.673	-0.307	0.030	-14.570	-0.049	2.394	2.899	-0.129	A0	TRAIN
6	Br	35	2558.128	-89.339	-0.432	-0.056	-2569.880	-0.130	-1.958	-1.541	-0.146	A0	TRAIN
7	C	6	37.664	-5.071	-0.436	0.017	-37.684	-0.077	2.107	2.894	-0.178	A0	TRAIN
8	Ca	20	676.598	-35.206	-0.196	0.010	-676.708	-0.044	-0.270	-0.368	-0.043	A0	TRAIN
9	Cl	17	459.196	-27.533	-0.478	-0.047	-459.449	-0.133	0.846	0.965	-0.141	A0	CV
10	F	9	99.066	-10.016	-0.680	0.035	-99.372	-0.141	1.958	2.145	-0.154	A0	TEST
11	He	2	2.855	-1.027	-0.914	1.400	-2.855	-0.032	5.646	5.052	-0.034	A0	TRAIN
12	Kr	36	2738.754	-93.562	-0.525	0.078	-2749.258	-0.130	-3.908	-1.545	-0.208	A0	TRAIN
13	Mg	12	199.550	-15.996	-0.253	0.022	-199.596	-0.041	0.416	1.233	-0.065	A0	TRAIN
14	N	7	54.231	-6.590	-0.572	0.088	-54.387	-0.092	1.948	2.729	-0.159	A0	TRAIN
15	Ne	10	128.343	-12.082	-0.854	0.292	-128.484	-0.163	1.584	2.148	-0.192	A0	TRAIN
16	O	8	74.525	-8.193	-0.519	0.071	-74.787	-0.118	1.986	2.677	-0.181	A0	TRAIN
17	P	15	340.538	-22.639	-0.392	0.031	-340.691	-0.085	0.470	1.220	-0.121	A0	TRAIN
18	Biciclobutano	30	154.843	-22.928	-0.359	0.076	-154.885	-0.638	2.885	2.969	-0.679	AGD	TEST
19	Cubano	56	307.505	-44.492	-0.384	0.070	-307.410	-1.226	3.056	2.991	-1.170	AGD	TRAIN
20	Tetraedrano	28	153.557	-22.183	-0.338	0.071	-153.612	-0.606	2.853	2.866	-0.612	AGD	TRAIN
21	As_am	33	2225.944	-81.248	-0.289	-0.045	-2231.902	-0.123	-0.618	-1.412	-0.090	AM	TRAIN
22	C_am	6	37.553	-5.004	-0.355	0.021	-37.594	-0.108	2.514	2.829	-0.172	AM	TEST
23	O_am	8	74.363	-8.114	-0.587	-0.025	-74.660	-0.160	2.505	2.360	-0.143	AM	TRAIN
24	P_am	15	340.401	-22.573	-0.306	-0.042	-340.601	-0.105	1.272	0.829	-0.078	AM	TRAIN
25	S_am	16	397.144	-24.970	-0.405	-0.075	-397.394	-0.137	1.496	1.042	-0.100	AM	TRAIN
26	Se_am	34	2391.876	-85.282	-0.373	-0.080	-2397.492	-0.147	-0.222	-1.510	-0.085	AM	CV
27	C+	5	37.212	-4.737	-0.911	-0.364	-37.287	-0.072	1.380	1.358	-0.070	AP	CV
28	N+	6	53.631	-6.194	-1.116	-0.464	-53.874	-0.080	1.267	1.194	-0.073	AP	TRAIN
29	O+	7	74.038	-7.892	-1.331	-0.385	-74.344	-0.080	0.918	1.253	-0.111	AP	TEST
30	P+	14	340.109	-22.361	-0.724	-0.342	-340.325	-0.079	0.685	0.638	-0.075	AP	TRAIN

Índice	Sistema	N	E_T	E_K	ε_{HOMO}	ε_{LUMO}	E_{HF}	$E_{corr,QCI}$	b_{opt}	b_m	E_θ	SUB	RN
31	S+	15	396.933	-24.788	-0.862	-0.307	-397.141	-0.095	0.506	0.515	-0.095	AP	TRAIN
32	Benceno	42	230.594	-33.307	-0.338	0.084	-230.722	-0.901	2.848	2.867	-0.914	CMOL	CV
33	13Butadieno	30	154.847	-22.860	-0.327	0.082	-154.937	-0.629	2.786	2.856	-0.662	CMOL	TRAIN
34	Etano	18	79.215	-12.521	-0.488	0.094	-79.239	-0.360	2.903	3.135	-0.424	CMOL	TRAIN
35	Eteno	16	77.892	-11.741	-0.370	0.092	-78.041	-0.329	2.762	3.037	-0.411	CMOL	TRAIN
36	Etino	14	76.762	-11.007	-0.415	0.080	-76.827	-0.295	2.783	3.045	-0.381	CMOL	TEST
37	H2	2	1.134	-0.659	-0.597	0.241	-1.131	-0.034	3.397	3.542	-0.036	CMOL	CV
38	Metano	10	40.187	-6.596	-0.547	0.092	-40.202	-0.194	2.849	3.067	-0.227	CMOL	TRAIN
39	Propano	26	118.245	-18.443	-0.469	0.092	-118.277	-0.527	2.940	3.136	-0.604	CMOL	TRAIN
40	Propeno	24	117.028	-17.689	-0.361	0.087	-117.085	-0.494	2.815	3.008	-0.569	CMOL	TRAIN
41	Propino	22	115.809	-16.939	-0.389	0.079	-115.876	-0.460	2.810	3.031	-0.550	CMOL	TRAIN
42	Bencilo+	48	268.762	-38.580	-0.542	-0.192	-268.903	-1.027	1.983	2.355	-1.619	CYCP	TRAIN
43	C3H9C+	32	156.428	-23.721	-0.733	-0.157	-156.459	-0.648	1.990	2.030	-0.677	CYCP	TEST
44	CH5CHCH3+	32	156.403	-23.712	-0.691	-0.175	-156.436	-0.648	1.982	2.086	-0.730	CYCP	CV
45	Ciclobutano	32	156.090	-23.660	-0.435	0.088	-156.111	-0.665	2.969	3.113	-0.738	CYCP	CV
46	CiclohexBote	48	234.180	-35.537	-0.425	0.083	-234.217	-1.002	3.032	3.104	-1.052	CYCP	TRAIN
47	CiclohexSilla	48	234.192	-35.541	-0.425	0.081	-234.227	-1.002	3.019	3.101	-1.058	CYCP	TRAIN
48	Ciclopropano	40	195.151	-29.606	-0.450	0.087	-195.180	-0.833	3.012	3.110	-0.891	CYCP	TRAIN
49	Ciclopropano	24	117.026	-17.735	-0.420	0.086	-117.071	-0.497	2.896	3.106	-0.583	CYCP	TRAIN
50	C2(1.83)	12	74.727	-9.877	-0.352	-0.047	-75.322	-0.289	2.663	2.776	-0.338	DS1	TRAIN
51	H2(1.54)	2	0.747	-0.475	-0.430	0.047	-0.992	-0.061	3.603	3.435	-0.051	DS1	TRAIN
52	H2(2.90)	2	0.745	-0.372	-0.317	-0.100	-0.823	-0.175	3.249	3.176	-0.165	DS1	TEST
53	He2+(1.07)	3	4.949	-1.578	-1.486	-0.641	-4.918	-0.049	1.336	1.461	-0.059	DS1	TRAIN
54	He2+(2.16)	3	4.784	-1.456	-1.119	-0.839	-4.842	-0.054	1.310	1.256	-0.041	DS1	CV
55	He2+(4.0)	3	4.839	-1.429	-0.991	-0.859	-4.825	-0.056	1.251	1.213	-0.043	DS1	TRAIN
56	Be2(2.17)	8	29.361	-5.375	-0.228	0.004	-29.117	-0.114	2.391	2.500	-0.140	DS2	TRAIN
57	Be2(5.23)	8	29.272	-5.345	-0.304	0.024	-29.140	-0.097	2.412	2.883	-0.228	DS2	TEST
58	C2(1.26)	12	75.268	-10.017	-0.454	-0.115	-75.383	-0.356	2.781	2.649	-0.301	DS2	TRAIN
59	C2(4.26)	12	74.614	-9.523	-0.249	-0.125	-74.944	-0.618	2.814	2.414	-0.333	DS2	TRAIN
60	F2(1.84)	18	197.455	-19.728	-0.628	-0.127	-198.584	-0.463	2.770	1.951	-0.271	DS2	TRAIN

Índice	Sistema	N	E_T	E_K	ϵ_{HOMO}	ϵ_{LUMO}	E_{HF}	$E_{corr,QCI}$	b_{opt}	b_{rn}	E_θ	SUB	RN
61	F2(3.1)	18	197.543	-19.597	-0.485	-0.284	-198.401	-0.644	2.899	1.968	-0.316	DS2	TRAIN
62	H2(2.17)	2	0.710	-0.410	-0.364	-0.043	-0.894	-0.114	3.350	3.300	-0.107	DS2	TRAIN
63	H2(3.7)	2	0.791	-0.354	-0.285	-0.131	-0.780	-0.217	3.297	3.074	-0.188	DS2	TRAIN
64	He2+(3.73)	3	4.838	-1.431	-1.001	-0.859	-4.826	-0.056	1.256	1.215	-0.042	DS2	TEST
65	3_C2	12	75.618	-10.181	-0.430	-0.050	-75.473	-0.252	2.561	2.785	-0.325	EM	TRAIN
66	C2	12	75.339	-10.035	-0.458	-0.115	-75.383	-0.355	2.783	2.642	-0.297	EM	TRAIN
67	H2Oclust2	20	151.576	-17.891	-0.483	0.125	-152.070	-0.433	3.288	2.959	-0.372	H2OC	CV
68	H2Oclust3	30	227.382	-26.848	-0.504	0.129	-228.115	-0.653	3.376	2.949	-0.544	H2OC	TRAIN
69	H2Oclust4	40	303.187	-35.805	-0.491	0.132	-304.162	-0.871	3.497	2.963	-0.702	H2OC	CV
70	H2Oclust5	50	378.982	-44.757	-0.501	0.123	-380.208	-1.092	3.492	2.942	-0.877	H2OC	TEST
71	H2Sclust2	36	796.993	-51.438	-0.376	0.066	-797.352	-0.346	1.308	1.227	-0.335	H2SC	TRAIN
72	H2Sclust3anti	54	1195.501	-77.162	-0.386	0.064	-1196.028	-0.523	1.299	1.276	-0.518	H2SC	TRAIN
73	H2Sclust3syn	54	1195.501	-77.162	-0.386	0.062	-1196.028	-0.523	1.299	1.279	-0.519	H2SC	TRAIN
74	H2Sclust4Ci	72	1594.002	-102.883	-0.386	0.062	-1594.706	-0.698	1.299	1.277	-0.692	H2SC	TRAIN
75	H2Sclust4S4	72	1594.001	-102.884	-0.386	0.062	-1594.705	-0.698	1.298	1.278	-0.693	H2SC	TRAIN
76	CCl4	74	1874.998	-115.413	-0.464	0.064	-1875.747	-0.785	1.225	1.181	-0.771	HALO	TRAIN
77	CHCl3	58	1416.309	-88.219	-0.455	0.078	-1416.873	-0.630	1.342	1.266	-0.611	HALO	TRAIN
78	HCl	18	459.845	-27.890	-0.477	0.088	-460.067	-0.170	1.080	1.206	-0.178	HALO	TRAIN
79	HF	10	99.746	-10.419	-0.653	0.194	-100.024	-0.199	2.819	2.624	-0.184	HALO	TRAIN
80	H3CH-H2O	20	116.002	-15.546	-0.511	0.100	-116.234	-0.409	3.057	3.020	-0.401	HB	TRAIN
81	NH3-H2O	20	131.896	-16.633	-0.451	0.099	-132.242	-0.430	3.238	3.006	-0.382	HB	TRAIN
82	Pirrolidina	40	211.044	-30.677	-0.368	0.083	-211.164	-0.851	3.003	2.990	-0.844	HETC	TRAIN
83	Piperidina	48	250.079	-36.613	-0.350	0.081	-250.210	-1.020	3.009	2.940	-0.976	HETC	TRAIN
84	22Metiocicb	56	592.480	-54.633	-0.332	0.075	-592.673	-0.991	2.619	2.366	-0.858	HETC	TRAIN
85	3Etilazacicb	48	250.050	-36.580	-0.362	0.076	-250.173	-1.020	3.022	2.973	-0.989	HETC	TEST
86	2Fluotiocicp	40	573.980	-46.582	-0.367	0.067	-574.405	-0.652	2.229	2.111	-0.610	HETC	TRAIN
87	Furano	36	228.377	-30.442	-0.326	0.079	-228.640	-0.792	2.774	2.734	-0.769	HETC	TRAIN
88	Indol	62	361.254	-50.737	-0.287	0.060	-361.494	-1.367	2.907	2.594	-1.115	HETC	TEST
89	3Meoxacicb	56	308.860	-43.803	-0.397	0.076	-309.074	-1.189	3.011	3.020	-1.196	HETC	TRAIN
90	NMeazacicp	32	171.959	-24.714	-0.370	0.081	-172.084	-0.685	2.936	2.984	-0.708	HETC	TRAIN

Índice	Sistema	N	E_T	E_K	ϵ_{HOMO}	ϵ_{LUMO}	E_{HF}	$E_{corr,QCI}$	b_{opt}	b_{rrn}	E_θ	SUB	RN
91	Oxaciclob	32	191.735	-25.998	-0.408	0.080	-191.923	-0.682	2.911	3.000	-0.725	HETC	TEST
92	Oxaciclop	24	152.677	-20.053	-0.450	0.085	-152.879	-0.518	2.868	2.995	-0.566	HETC	TRAIN
93	Pirrol	36	208.636	-29.194	-0.301	0.067	-208.828	-0.790	2.856	2.671	-0.693	HETC	TEST
94	Quinoleina	68	399.124	-55.905	-0.316	0.067	-399.373	-1.500	2.883	2.734	-1.354	HETC	TRAIN
95	Tiociclop	48	553.483	-48.736	-0.329	0.075	-553.658	-0.817	2.531	2.248	-0.697	HETC	TRAIN
96	Tiofeno	44	551.073	-47.251	-0.334	0.078	-551.304	-0.759	2.445	2.158	-0.637	HETC	CV
97	2Indanona	70	420.049	-57.913	-0.335	0.069	-420.372	-1.530	2.871	2.806	-1.465	HETC	TRAIN
98	OxIndeno	70	420.016	-57.927	-0.328	0.072	-420.319	-1.540	2.931	2.774	-1.391	HETC	CV
99	Dimetilam	26	134.127	-19.517	-0.364	0.082	-134.255	-0.545	2.937	2.988	-0.563	HETM	CV
100	Metilfosf	26	381.339	-29.612	-0.357	0.060	-381.497	-0.333	1.973	2.047	-0.347	HETM	TRAIN
101	Dimetilfosf	34	420.366	-35.543	-0.339	0.063	-420.540	-0.501	2.239	2.214	-0.494	HETM	TRAIN
102	AcMefosf	50	605.406	-54.618	-0.459	0.056	-606.201	-0.893	2.518	2.509	-0.889	HETM	TRAIN
103	AcMesulf	50	661.425	-56.608	-0.489	0.056	-662.236	-0.912	2.472	2.412	-0.885	HETM	TEST
104	Dietilamina	42	212.196	-31.364	-0.361	0.088	-212.337	-0.880	3.005	2.994	-0.873	HETM	TEST
105	H3O+	10	76.093	-9.041	-0.955	-0.133	-76.311	-0.213	1.732	1.627	-0.192	HETM	TRAIN
106	H3PO4	50	641.063	-57.006	-0.479	0.050	-642.051	-0.910	2.497	2.428	-0.880	HETM	CV
107	H3S+	18	398.830	-25.823	-0.783	-0.135	-398.953	-0.179	0.910	1.024	-0.195	HETM	TRAIN
108	PH3	18	342.282	-23.658	-0.301	0.068	-342.398	-0.167	1.375	1.220	-0.155	HETM	CV
109	PH4+	18	342.683	-23.827	-0.841	-0.120	-342.771	-0.165	0.843	1.202	-0.215	HETM	TRAIN
110	Benz_Ne	52	358.654	-45.316	-0.336	0.082	-359.204	-1.068	2.715	2.728	-1.078	NCOV	TRAIN
111	C2H2_C2H2	28	153.173	-21.932	-0.405	0.074	-153.653	-0.600	2.862	3.030	-0.700	NCOV	CV
112	C2H2_C1F	40	634.751	-48.516	-0.429	0.062	-635.652	-0.656	2.260	2.150	-0.615	NCOV	TEST
113	C2H4_C2H4	32	155.739	-23.469	-0.376	0.085	-156.084	-0.661	2.816	3.047	-0.789	NCOV	TRAIN
114	C2H4_F2	34	275.776	-31.640	-0.377	0.074	-276.722	-0.726	2.673	2.758	-0.764	NCOV	TRAIN
115	CH3Cl_HCl	44	958.682	-61.686	-0.455	0.071	-959.167	-0.507	1.613	1.581	-0.501	NCOV	CV
116	CH3SH_HCl	44	897.318	-59.520	-0.382	0.066	-897.779	-0.513	1.691	1.478	-0.467	NCOV	TEST
117	CH4_CH4	20	80.293	-13.171	-0.537	0.087	-80.403	-0.390	2.920	3.077	-0.433	NCOV	TEST
118	CH4_Ne	20	168.501	-18.667	-0.545	0.094	-168.686	-0.357	2.461	2.767	-0.422	NCOV	TRAIN
119	H2O_C1F	36	633.879	-46.478	-0.473	0.073	-634.863	-0.572	2.225	2.033	-0.526	NCOV	CV
120	H2O_H2O	20	151.470	-17.861	-0.479	0.124	-152.069	-0.436	3.321	2.960	-0.369	NCOV	TRAIN

Índice	Sistema	N	E_T	E_K	ϵ_{HOMO}	ϵ_{LUMO}	E_{HF}	$E_{corr,QCI}$	b_{opt}	b_{rrn}	E_θ	SUB	RN
121	H2S_H2S	36	796.948	-51.426	-0.378	0.066	-797.351	-0.347	1.312	1.235	-0.336	NCOV	TRAIN
122	H2S_HcI	36	858.309	-53.600	-0.410	0.063	-858.745	-0.344	1.194	1.178	-0.342	NCOV	TRAIN
123	HCN_CH3SH	40	530.000	-43.655	-0.385	0.061	-530.591	-0.663	2.391	2.311	-0.632	NCOV	TRAIN
124	HCN_CIF	40	650.672	-49.578	-0.467	0.057	-651.709	-0.676	2.308	2.147	-0.618	NCOV	TRAIN
125	HCONH2_dimer	48	336.795	-42.188	-0.411	0.075	-337.912	-1.073	3.009	2.911	-1.010	NCOV	TRAIN
126	HCOOH_dimer	48	376.267	-44.706	-0.479	0.076	-377.572	-1.078	2.876	2.849	-1.061	NCOV	CV
127	HCl_HCl	36	919.651	-55.772	-0.470	0.073	-920.135	-0.341	1.047	1.184	-0.359	NCOV	TRAIN
128	HF_HF	20	199.397	-20.818	-0.627	0.151	-200.055	-0.401	2.832	2.592	-0.366	NCOV	TRAIN
129	NH3_Cl2	44	974.509	-62.729	-0.426	0.048	-975.118	-0.533	1.754	1.445	-0.471	NCOV	TRAIN
130	NH3_CIF	36	614.163	-45.213	-0.458	0.073	-615.034	-0.575	2.326	2.083	-0.515	NCOV	TRAIN
131	NH3_F2	28	253.906	-27.559	-0.434	0.089	-254.881	-0.612	2.924	2.781	-0.571	NCOV	TRAIN
132	NH3_H2O	20	131.749	-16.590	-0.456	0.097	-132.240	-0.434	3.282	3.003	-0.376	NCOV	TRAIN
133	NH3_NH3	20	112.008	-15.314	-0.403	0.103	-112.404	-0.430	3.261	3.058	-0.385	NCOV	TEST
134	NH3clust2asym	20	112.053	-15.327	-0.404	0.105	-112.405	-0.430	3.251	3.060	-0.387	NH3C	TRAIN
135	NH3clust3equi	30	168.069	-22.989	-0.406	0.108	-168.614	-0.649	3.400	3.066	-0.548	NH3C	TRAIN
136	NH3clust4boat	40	224.104	-30.655	-0.395	0.108	-224.822	-0.867	3.501	3.048	-0.698	NH3C	TEST
137	NH3clust5ring	50	279.924	-38.261	-0.395	0.104	-281.025	-1.086	3.546	3.045	-0.858	NH3C	TRAIN
138	Etilamina	26	134.135	-19.519	-0.383	0.082	-134.267	-0.545	2.964	3.036	-0.571	NMOL	TRAIN
139	HNO3	32	278.782	-31.577	-0.498	0.072	-279.445	-0.761	2.895	2.754	-0.701	NMOL	TRAIN
140	Metilamina	18	95.098	-13.598	-0.386	0.077	-95.226	-0.378	2.954	3.027	-0.396	NMOL	TRAIN
141	N2	14	108.650	-13.125	-0.631	0.138	-108.947	-0.327	2.737	2.707	-0.320	NMOL	CV
142	NH3	10	56.073	-7.677	-0.424	0.114	-56.201	-0.213	3.089	3.093	-0.213	NMOL	TRAIN
143	NH4+	10	56.436	-7.792	-1.009	-0.124	-56.546	-0.212	1.830	1.695	-0.186	NMOL	TRAIN
144	Piridina	42	246.519	-34.371	-0.351	0.073	-246.712	-0.921	2.829	2.875	-0.953	NMOL	TRAIN
145	AcAcetico	32	227.411	-28.326	-0.457	0.071	-227.829	-0.696	2.853	2.912	-0.723	OMOL	CV
146	Acetona	32	191.743	-25.958	-0.411	0.074	-191.977	-0.679	2.841	2.997	-0.756	OMOL	TRAIN
147	DimetilEter	26	153.876	-20.782	-0.421	0.085	-154.079	-0.544	2.910	3.026	-0.587	OMOL	TRAIN
148	Etanol	26	153.882	-20.790	-0.441	0.076	-154.096	-0.545	2.962	3.020	-0.566	OMOL	TEST
149	Formaldehido	16	113.644	-14.100	-0.440	0.076	-113.874	-0.347	2.725	2.925	-0.402	OMOL	TRAIN
150	AcFormico	24	188.357	-22.395	-0.476	0.076	-188.777	-0.531	2.806	2.850	-0.546	OMOL	TEST

Índice	Sistema	N	E_T	E_K	ϵ_{HOMO}	ϵ_{LUMO}	E_{HF}	$E_{corr,QCI}$	b_{opt}	b_{rrn}	E_θ	SUB	RN
151	H2O	10	75.812	-8.950	-0.510	0.146	-76.031	-0.215	3.192	2.972	-0.193	OMOL	CV
152	H2O2	18	150.373	-17.122	-0.481	0.152	-150.784	-0.405	3.213	2.934	-0.354	OMOL	TRAIN
153	Metanol	18	114.845	-14.867	-0.450	0.079	-115.052	-0.378	2.931	2.990	-0.392	OMOL	TRAIN
154	O2	16	149.109	-16.296	-0.451	0.037	-149.537	-0.378	2.702	2.685	-0.374	OMOL	CV
155	BenzCH2rad	49	269.025	-38.929	-0.312	0.072	-269.160	-1.022	2.826	2.751	-0.971	RAD	TRAIN
156	C3H9Crad	33	156.647	-24.000	-0.315	0.083	-156.691	-0.661	2.862	2.842	-0.652	RAD	TRAIN
157	CH5CHCH3rad	33	156.640	-24.000	-0.329	0.087	-156.687	-0.659	2.871	2.902	-0.672	RAD	TRAIN
158	CH3rad	9	39.542	-6.229	-0.383	0.071	-39.567	-0.158	2.537	3.091	-0.242	RAD	TRAIN
159	OHrad	9	75.155	-8.556	-0.513	0.070	-75.393	-0.166	2.631	2.766	-0.179	RAD	TEST
160	PH2rad	17	341.715	-23.325	-0.375	0.020	-341.855	-0.138	1.131	1.484	-0.164	RAD	CV
161	SHrad	17	397.894	-25.368	-0.377	-0.007	-398.069	-0.139	1.070	1.100	-0.141	RAD	TRAIN
162	Propionitrilo	28	169.591	-23.176	-0.404	0.063	-169.779	-0.614	2.754	2.967	-0.732	RESO	TRAIN
163	3Buten2ona	38	229.555	-31.129	-0.392	0.072	-229.823	-0.814	2.822	2.964	-0.901	RESO	TRAIN
164	Ciclohexeno	46	232.972	-34.789	-0.343	0.079	-233.039	-0.968	2.964	2.925	-0.944	RESO	CV
165	Ciclopentadieno	36	192.727	-28.100	-0.312	0.087	-192.809	-0.770	2.859	2.764	-0.720	RESO	TRAIN
166	Ciclopenteno	38	193.936	-28.854	-0.342	0.088	-193.994	-0.801	2.933	2.920	-0.793	RESO	TRAIN
167	AnhidMaleico	50	376.647	-46.247	-0.452	0.026	-377.243	-1.122	2.686	2.803	-1.221	RESO	CV
168	Tetracianet	64	444.360	-57.362	-0.454	-0.053	-444.926	-1.490	2.655	2.708	-1.557	RESO	TRAIN
169	H2O_MeOH	28	190.465	-23.768	-0.474	0.072	-191.090	-0.600	3.082	2.935	-0.553	ROHC	TRAIN
170	MeOHH2Oclust2	38	266.225	-32.714	-0.452	0.088	-267.134	-0.822	3.229	2.956	-0.718	ROHC	TRAIN
171	MeOHH2Oclust3	64	419.941	-53.468	-0.444	0.084	-421.221	-1.377	3.180	2.984	-1.242	ROHC	TEST
172	MeOH_H2O	28	190.461	-23.767	-0.424	0.095	-191.089	-0.600	3.142	2.972	-0.548	ROHC	TRAIN
173	MeOHclust2	36	229.438	-29.670	-0.423	0.072	-230.110	-0.765	3.036	2.977	-0.738	ROHC	TRAIN
174	MeOHclust3	54	344.165	-44.521	-0.443	0.080	-345.175	-1.152	3.090	2.992	-1.091	ROHC	CV
175	DimetilSulfuro	34	476.574	-37.573	-0.335	0.078	-476.746	-0.507	2.245	2.006	-0.446	SMOL	TRAIN
176	Metanotiol	26	437.542	-31.648	-0.357	0.072	-437.710	-0.339	1.981	1.789	-0.307	SMOL	TRAIN
177	H2SO4	50	697.048	-58.982	-0.500	0.054	-698.057	-0.929	2.444	2.330	-0.880	SMOL	TRAIN
178	SH2	18	398.512	-25.722	-0.385	0.074	-398.676	-0.172	1.313	1.268	-0.169	SMOL	TRAIN
179	SO2	32	546.508	-41.547	-0.494	0.017	-547.174	-0.543	2.157	2.005	-0.498	SMOL	TRAIN

Bibliografía

- [1] IUPAC. *Compendium of Chemical Terminology*. Blackwell Scientific Publications, Oxford, 2° edición, 2019. DOI: [10.1351/goldbook](https://doi.org/10.1351/goldbook) (citado en la pág. 2).
- [2] I. Levine. *Química Cuántica*. Prentice Hall, 5° edición, 2001 (citado en la pág. 2).
- [3] D. McQuarrie y J. Simon. *Physical chemistry, a molecular Approach*. University Science books, 1° edición, 1997 (citado en la pág. 2).
- [4] R. Grinter. *The quantum in chemistry: an experimentalist's view*. John Wiley y Sons, 2005 (citado en la pág. 2).
- [5] C. J. Cramer. *Essentials of Computational Chemistry: Theories and Models*. Wiley, 2° edición, 2004 (citado en la pág. 3).
- [6] A. Szabo y N. Ostlund. *Modern Quantum Chemistry*. McGraw-Hill, 1989 (citado en las págs. 4, 5).
- [7] D. B. Cook. *Handbook of Computational Quantum Chemistry*. Oxford University Press, 1° edición, 1998 (citado en la pág. 5).
- [8] A. Aspuru-Guzik, A. D. Dutoi, P. J. Love y M. Head-Gordon. Chemistry: Simulated quantum computation of molecular energies. *Science*, **309**(5741):1704-1707, 2005. DOI: [10.1126/science.1113479](https://doi.org/10.1126/science.1113479) (citado en la pág. 5).
- [9] Google AI Quantum. Hartree-Fock on a superconducting qubit quantum computer. *Science*, **369**(6507):1084-1089, 2020. DOI: [10.1126/science.abb9811](https://doi.org/10.1126/science.abb9811) (citado en la pág. 5).
- [10] D. Poole, J. L. Galvez Vallejo y M. S. Gordon. A New Kid on the Block: Application of Julia to Hartree-Fock Calculations. *Journal of Chemical Theory and Computation*, **16**(8):5006-5013, 2020. DOI: [10.1021/acs.jctc.0c00337](https://doi.org/10.1021/acs.jctc.0c00337) (citado en la pág. 5).
- [11] T. Tsuneda. *Density functional theory in quantum chemistry*. Springer, 1° edición, 2014 (citado en la pág. 5).
- [12] P. Lödwin. Quantum theory of many particle systems. extension of the hartree fock scheme to include degenerate systems and correlation effects. *Phys. Rev.*, **97**(6):1509-1520, 1955. DOI: [10.1103/PhysRev.97.1509](https://doi.org/10.1103/PhysRev.97.1509) (citado en la pág. 5).
- [13] T. Helgaker, P. Jorgensen y J. Olsen. *Molecular electronic structure theory*. Wiley, 2012 (citado en la pág. 5).
- [14] J. A. Pople, J. S. Binkley y R. Seeger. Theoretical models incorporating electron correlation. *International Journal of Quantum Chemistry*, **10**(10 S):1-19, 1976. DOI: [10.1002/qua.560100802](https://doi.org/10.1002/qua.560100802) (citado en la pág. 6).
- [15] C. Møller y M. S. Plesset. Note on an Approximation Treatment for Many-Electron Systems. *Physical Review*, **46**(7):618-622, 1934. DOI: [10.1103/PhysRev.46.618](https://doi.org/10.1103/PhysRev.46.618) (citado en la pág. 6).

- [16] P. J. Knowles, K. Somasundram, N. C. Handy y K. Hirao. The calculation of higher-order energies in the many-body perturbation theory series. *Chemical Physics Letters*, **113**(1):8-12, 1985. DOI: [10.1016/0009-2614\(85\)85002-8](https://doi.org/10.1016/0009-2614(85)85002-8) (citado en la pág. 6).
- [17] J. Cizek. On the Correlation Problem in Atomic and Molecular Systems. Calculation of Wavefunction Components in Ursell-Type Expansion Using Quantum-Field Theoretical Methods. *The Journal of Chemical Physics*, **45**(11):4256-4266, 1966. DOI: [10.1063/1.1727484](https://doi.org/10.1063/1.1727484) (citado en la pág. 6).
- [18] J. Pople, M. Head-Gordon y K. Raghavachari. Quadratic configuration interaction. a general technique for determining electron correlation energies. *The Journal of Chemical Physics*, **87**(10):5968-5975, 1987. DOI: [10.1063/1.453520](https://doi.org/10.1063/1.453520) (citado en la pág. 6).
- [19] D. Cremer y Z. He. Size-extensive QCISDT - implementation and application. **222**(1-2):40-45, 1994. DOI: [10.1016/0009-2614\(94\)00328-9](https://doi.org/10.1016/0009-2614(94)00328-9) (citado en la pág. 6).
- [20] T. J. Lee, A. P. Rendell, K. G. Dyall y D. Jayatilaka. Open-shell restricted Hartree-Fock perturbation theory: Some considerations and comparisons. *The Journal of Chemical Physics*, **100**(10):7400-7409, 1994. DOI: [10.1063/1.466883](https://doi.org/10.1063/1.466883) (citado en la pág. 6).
- [21] M. Böhme y G. Frenking. The CuC bond dissociation energy of CuCH₃. A dramatic failure of the QCISD(T) method. *Chemical Physics Letters*, **224**(1-2):195-199, 1994. DOI: [10.1016/0009-2614\(94\)00526-5](https://doi.org/10.1016/0009-2614(94)00526-5) (citado en la pág. 6).
- [22] R. Bader. *Atoms in Molecules, a quantum theory*. Oxford University Press, 1995 (citado en la pág. 6).
- [23] R. Parr y W. Yang. *Density Functional theory of atoms and molecules*. Oxford University Press, 1989 (citado en las págs. 6-9).
- [24] K. Burke y L. O. Wagner. Dft in a nutshell. *International Journal of Quantum Chemistry*, **113**:96-101, 2013. DOI: [10.1002/qua.24259](https://doi.org/10.1002/qua.24259) (citado en las págs. 7, 8).
- [25] P Hohenberg y W Kohn. Inhomogeneous electron gas. *Physical Review B*, **136**(3):864-871, 1964. DOI: [10.1103/PhysRev.136.B864](https://doi.org/10.1103/PhysRev.136.B864) (citado en la pág. 7).
- [26] W. Kohn y L. J. Sham. Self-consistent equations including the exchange and correlation. *Physical Review A*, **140**(4):1133-1138, 1965. DOI: [10.1103/PhysRev.140.A1133](https://doi.org/10.1103/PhysRev.140.A1133) (citado en la pág. 8).
- [27] J. F. Janak. Proof that $\partial e/\partial n_i = \epsilon_i$ in density functional theory. *Physical Review B*, **18**(12):7165-7168, 1978. DOI: [10.1103/PhysRevB.18.7165](https://doi.org/10.1103/PhysRevB.18.7165) (citado en la pág. 9).
- [28] M. Weinert y J. W. Davenport. Fractional occupations and density functional energies and forces. *Physical Review B*, **45**(23):13709-13712, 1992. DOI: [10.1103/PhysRevB.45.13709](https://doi.org/10.1103/PhysRevB.45.13709) (citado en las págs. 9, 19).
- [29] J. D. Chai. Density functional theory with fractional orbital occupations. *J. Chem. Phys.*, **136**(15):154104, 2012. DOI: [10.1063/1.3703894](https://doi.org/10.1063/1.3703894) (citado en las págs. 10, 48, 52).
- [30] J. D. Chai. Thermally assisted occupation density functional theory with generalized gradient approximations. *The Journal of Chemical Physics*, **140**(18):18A521, 2014. DOI: [10.1063/1.4867532](https://doi.org/10.1063/1.4867532) (citado en la pág. 10).
- [31] C. N. Yeh, C. Wu, H. Su y J. D. Chai. Electronic properties of the coronene series from thermally-assisted-occupation density functional theory. *RSC Advances*, **8**(60):34350-34358, 2018. DOI: [10.1039/C8RA01336E](https://doi.org/10.1039/C8RA01336E) (citado en la pág. 10).
- [32] J. H. Chung y J. D. Chai. Electronic Properties of Möbius Cyclacenes Studied by Thermally-Assisted-Occupation Density Functional Theory. *Scientific Reports*, **9**(1), 2019. DOI: [10.1038/s41598-019-39524-4](https://doi.org/10.1038/s41598-019-39524-4) (citado en la pág. 10).

- [33] H. J. Huang, S. Seenithurai y J. D. Chai. TAO-DFT study on the electronic properties of diamond-shaped graphene nanoflakes. *Nanomaterials*, **10**(6):1-16, 2020. DOI: [10.3390/nano10061236](https://doi.org/10.3390/nano10061236) (citado en la pág. 10).
- [34] S. Seenithurai y J. D. Chai. TAO-DFT investigation of electronic properties of linear and cyclic carbon chains. *Scientific Reports*, **10**(1), 2020. DOI: [10.1038/s41598-020-70023-z](https://doi.org/10.1038/s41598-020-70023-z) (citado en la pág. 10).
- [35] P. Joshi. *Artificial Intelligence with Python*. Packt, 2017 (citado en la pág. 10).
- [36] A. Gulli y S. Pal. *Deep Learning with Keras*. Packt, 2017 (citado en las págs. 13, 16).
- [37] E. Alba y R. Marti. *Metaheuristic procedures for training neural networks*. Springer, 1° edición, 2006 (citado en la pág. 15).
- [38] A. A. Rodríguez-Pié. *Implementación de redes neuronales convolucionales para el meta-análisis de acoplamientos moleculares de complejos proteína-ligando*, UNAM, 2019 (citado en la pág. 15).
- [39] J. Bojórquez-Mora. *Uso de redes neuronales arificiales para estimar la respuesta sísmica de sistemas estructurales*, UNAM, 2011 (citado en la pág. 15).
- [40] I. Goodfellow, Y. Bengio y A. Courville. *Deep Learning*. MIT Press, 2016. <http://www.deeplearningbook.org> (citado en la pág. 16).
- [41] D. E. Rumelhart y J. L. McClelland. *Parallel distributed processing: Explorations in the microstructure cognition. Volume 1. Foundations*. MIT Press, 1986 (citado en la pág. 16).
- [42] L. Y., L. Bottou, Y. Bengio y P. Haffner. Gradient based learning applied to document recognition. *Proceedings of the IEEE*, **86**(11):2278 -2324, 1998. DOI: [10.1109/5.726791](https://doi.org/10.1109/5.726791) (citado en la pág. 16).
- [43] V Kůrková. Kolmogorov's theorem and multilayer neural networks. *Neural Networks*, **5**(3):501-506, 1992. DOI: [10.1016/0893-6080\(92\)90012-8](https://doi.org/10.1016/0893-6080(92)90012-8) (citado en la pág. 18).
- [44] K Hornik. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, **4**(2):251-257, 1991. DOI: [10.1016/0893-6080\(91\)90009-T](https://doi.org/10.1016/0893-6080(91)90009-T) (citado en la pág. 18).
- [45] O. A. von Lilienfeld. Quantum Machine Learning in Chemical Compound Space. *Angewandte Chemie - International Edition*, **57**(16):4164-4169, 2018. DOI: [10.1002/anie.201709686](https://doi.org/10.1002/anie.201709686) (citado en la pág. 18).
- [46] K. T. Butler, D. W. Davies, H. Cartwright, O. Isayev y A. Walsh. Machine learning for molecular and materials science. *Nature*, **559**(7715):547-555, 2018. DOI: [10.1038/s41586-018-0337-2](https://doi.org/10.1038/s41586-018-0337-2) (citado en la pág. 18).
- [47] L. H. Hu, X. J. Wang, L. H. Wong y G. H. Chen. Combined first-principles calculation and neural-network correction approach for heat of formation. *Journal of Chemical Physics*, **119**(22):11501-11507, 2003. DOI: [10.1063/1.1630951](https://doi.org/10.1063/1.1630951) (citado en la pág. 18).
- [48] H. Li, L. L. Shi, M. Zhang, Z. Su, X. J. Wang, L. H. Hu y G. H. Chen. Improving the accuracy of density-functional theory calculation: The genetic algorithm and neural network approach. *Journal of Chemical Physics*, **126**(14), 2007. DOI: [10.1063/1.2715579](https://doi.org/10.1063/1.2715579) (citado en la pág. 18).
- [49] J. Hermann, Z. Schätzle y F. Noé. Deep-neural-network solution of the electronic Schrödinger equation. *Nature Chemistry*, **12**(10):891-897, 2020. DOI: [10.1038/s41557-020-0544-y](https://doi.org/10.1038/s41557-020-0544-y) (citado en la pág. 18).
- [50] M. Rupp, A. Tkatchenko, K. R. Müller y O. A. Von Lilienfeld. Fast and accurate modeling of molecular atomization energies with machine learning. *Physical Review Letters*, **108**(5):1-5, 2012. DOI: [10.1103/PhysRevLett.108.058301](https://doi.org/10.1103/PhysRevLett.108.058301). arXiv: [1109.2618](https://arxiv.org/abs/1109.2618) (citado en la pág. 18).
- [51] M. J. Frisch, G. W. Trucks, H. B. Schlegel, G. E. Scuseria, M. A. Robb, J. R. Cheeseman, G. Scalmani, V. Barone, B. Mennucci, G. A. Petersson, H. Nakatsuji, M. Caricato, X. Li,

- H. P. Hratchian, A. F. Izmaylov, J. Bloino, G. Zheng, J. L. Sonnenberg, M. Hada, M. Ehara, K. Toyota, R. Fukuda, J. Hasegawa, M. Ishida, T. Nakajima, Y. Honda, O. Kitao, H. Nakai, T. Vreven, J. A. Montgomery Jr., J. E. Peralta, F. Ogliaro, M. Bearpark, J. J. Heyd, E. Brothers, K. N. Kudin, V. N. Staroverov, R. Kobayashi, J. Normand, K. Raghavachari, A. Rendell, J. C. Burant, S. S. Iyengar, J. Tomasi, M. Cossi, N. Rega, J. M. Millam, M. Klene, J. E. Knox, J. B. Cross, V. Bakken, C. Adamo, J. Jaramillo, R. Gomperts, R. E. Stratmann, O. Yazyev, A. J. Austin, R. Cammi, C. Pomelli, J. W. Ochterski, R. L. Martin, K. Morokuma, V. G. Zakrzewski, G. A. Voth, P. Salvador, J. J. Dannenberg, S. Dapprich, A. D. Daniels, Farkas, J. B. Foresman, J. V. Ortiz, J. Cioslowski y D. J. Fox. Gaussian09 Revision E.01. Gaussian Inc. Wallingford CT 2009 (citado en la pág. 27).
- [52] S. Sarkar, Monua y B. Bandyopadhyay. Cooperative nature of the sulfur centered hydrogen bond: investigation of $(\text{h}_2\text{s})_n$ ($n = 2 \sim 4$) clusters using an affordable yet accurate level of theory. *Phys. Chem. Chem. Phys.*, **21**(45):25439-25448, 2019. DOI: [10.1039/C9CP05326C](https://doi.org/10.1039/C9CP05326C) (citado en la pág. 29).
- [53] Y. Zhao y D. G. Truhlar. Benchmark databases for nonbonded interactions and their use to test density functional theory. *J. Chem. Theory Comput.*, **1**(3):415-432, 2005. DOI: [10.1021/ct049851d](https://doi.org/10.1021/ct049851d) (citado en la pág. 30).
- [54] P. E. Janeiro-Barral y M. Mella. Study of the structure, energetics and vibrational properties of small ammonia clusters $(\text{nh}_3)_n$ ($n = 2 - 5$) using correlated ab initio methods. *J. Phys. Chem. A*, **110**(39):11244-11251, 2006. DOI: [10.1021/jp063252g](https://doi.org/10.1021/jp063252g) (citado en la pág. 30).
- [55] L. Bytautas, G. E. Scuseria y K. Ruedenberg. Seniority number description of potential energy surfaces: symmetric dissociation of water, n_2 , c_2 , and be_2 . *J. Chem. Phys.*, **143**(9):094105, 2015. DOI: [10.1063/1.4929904](https://doi.org/10.1063/1.4929904) (citado en las págs. 48, 49).
- [56] L. Bytautas y K. Ruedenberg. Correlation energy extrapolation by intrinsic scaling. iv. accurate binding energies of the homonuclear diatomic molecules carbon, nitrogen, oxygen, and fluorine. *J. Chem. Phys.*, **122**(15):154110, 2005. DOI: [10.1063/1.1869493](https://doi.org/10.1063/1.1869493) (citado en la pág. 49).
- [57] X. Wang, W. E. Turner, J. Agarwal y H. F. Schaefer. Twisted triplet ethylene: anharmonic frequencies and spectroscopic parameters for c_2h_4 , c_2d_4 , and $^{13}\text{c}_2\text{h}_4$. *J. Phys. Chem. A*, **118**(35):75607567, 2014. DOI: [10.1021/jp502282v](https://doi.org/10.1021/jp502282v) (citado en la pág. 51).
- [58] P. R. Nagy, G. Samu y M Kállay. Optimization of the linear-scaling local natural orbital ccSD(t) method: improved algorithm and benchmark applications. *J. Chem. Theory Comput.*, **14**(8):4193-4215, 2018. DOI: [10.1021/acs.jctc.8b00442](https://doi.org/10.1021/acs.jctc.8b00442) (citado en la pág. 54).
- [59] S. J. Austin, P. W. Fowler, P. Hansen, D. E. Manolopoulos y M. Zheng. Fullerene isomers of C_{60} Kekulé counts versus stability. *Chemical Physics Letters*, **228**(4-5):478-484, 1994. DOI: [10.1016/0009-2614\(94\)00965-1](https://doi.org/10.1016/0009-2614(94)00965-1) (citado en la pág. 56).
- [60] S. J. Austin, P. W. Fowler, D. E. Manolopoulos, G. Orlandi y F. Zerbetto. Structural motifs and the stability of fullerenes. *Journal of physical chemistry*, **99**(20):8076-8081, 1995. DOI: [10.1021/j100020a035](https://doi.org/10.1021/j100020a035) (citado en la pág. 56).
- [61] A. C. Dillon. Carbon nanotubes for photoconversion and electrical energy storage. *Chemical Reviews*, **110**(11):6856-6872, 2010. DOI: [10.1021/cr9003314](https://doi.org/10.1021/cr9003314) (citado en la pág. 56).
- [62] R. Sure, A. Hansen, P. Schwerdtfeger y S. Grimme. Comprehensive theoretical study of all 1812 C_{60} isomers. *Physical Chemistry Chemical Physics*, **19**(22):14296-14305, 2017. DOI: [10.1039/c7cp00735c](https://doi.org/10.1039/c7cp00735c) (citado en la pág. 56).
- [63] C. Riplinger, B. Sandhoefer, A. Hansen y F. Neese. Natural triple excitations in local coupled cluster calculations with pair natural orbitals. *Journal of Chemical Physics*, **139**(13), 2013. DOI: [10.1063/1.4821834](https://doi.org/10.1063/1.4821834) (citado en la pág. 56).

- [64] P. W. Fowler y A. Ceulemans. Electron deficiency of the fullerenes. *Journal of physical chemistry*, **99**(2):508-510, 1995. DOI: [10.1021/j100002a010](https://doi.org/10.1021/j100002a010) (citado en la pág. 58).
- [65] Fereshteh Naderi y Valera Veryazov. Multiconfigurational Study of the Electronic Structure of Negatively Charged Fullerenes. *Journal of Chemistry and Chemical Engineering*, **11**(1):30-36, 2017. DOI: [10.17265/1934-7375/2017.01.005](https://doi.org/10.17265/1934-7375/2017.01.005) (citado en la pág. 58).