

TRABAJO DE FIN DE GRADO DE FÍSICA

Redes neuronales para aprender Hamiltonianos

JESÚS BARTOLOMÉ SARSA

Director:
DAVID ZUECO LÁINEZ



Universidad
Zaragoza



Facultad de Ciencias
Universidad Zaragoza

Departamento de Física de la Materia Condensada
Facultad de Ciencias, Universidad de Zaragoza

Junio de 2023

Índice

1. Introducción	1
2. Sistema Físico: Hamiltoniano de espín molecular	1
2.1. Estados fundamentales	2
2.2. Hamiltoniano de espín efectivo	2
2.3. Capacidad calorífica a campo cero	3
2.3.1. Aproximación de c a bajas temperaturas	4
2.3.2. Contribución de la red: modelo de Debye	5
3. Redes Neuronales	7
3.1. La neurona	7
3.2. Red neuronal: Capas sucesivas	8
3.3. Entrenamiento de la red	8
3.4. Base de datos: datos sintéticos	10
3.5. Desarrollo del código	10
4. Desarrollo del modelo	10
4.1. Dependencia con el tamaño de la base de datos y el tamaño de la red	11
4.2. Estudio del aprendizaje de la red: Aproximación a bajas temperaturas	14
4.3. Introducción de distorsiones sintéticas	17
5. Medidas experimentales	18
6. Conclusión	22
7. Referencias	23

1. Introducción

Al describir un sistema físico complejo con el objetivo de estudiar sus propiedades es muy útil utilizar un modelo que a partir de una serie de suposiciones sobre el comportamiento del sistema pueda predecir propiedades en base a unos parámetros. Esto facilita mucho la caracterización del sistema físico, ya que con obtener una lista de valores es posible determinar con precisión las propiedades del sistema original. En concreto, definir un sistema mediante su hamiltoniano es particularmente útil, ya que permite encontrar los estados permitidos junto a sus respectivas energías si son conocidas las interacciones presentes en el sistema.

Estos hamiltonianos son muy útiles a la hora de describir sistemas físicos. Sin embargo, cuando el sistema es complejo, el hamiltoniano necesario para describirlo está compuesto por la suma de varios términos de interacción y obtener soluciones analíticamente se complica enormemente. En muchos casos es imposible obtener las soluciones exactas, y resulta necesario realizar aproximaciones numéricas. Si los parámetros de interacción son conocidos, obtener los autoestados con sus respectivos autovalores siempre es posible mediante métodos numéricos, aunque sistemas de dimensión grande pueden requerir de métodos numéricos particulares y un gran poder computacional.

En física de la materia condensada, sin embargo, el problema habitual es el inverso: se busca caracterizar el hamiltoniano a partir de unas medidas experimentales. Ya que estas medidas sólo suelen tener información sobre los autoestados y sus respectivos autovalores, reconstruir el hamiltoniano original es muy difícil. Este problema no se puede abordar por tanto con ajustes numéricos tradicionales, y sólo es posible estimar algunos de estos parámetros mediante muchas medidas distintas del mismo sistema en condiciones diferentes.

Pese a que no se puede estimar numéricamente estos valores en muchos casos, sí se puede intuir “a ojo” los rangos de valores en los que se pueden encontrar si se tiene la suficiente experiencia en sistemas de este tipo. Este aprendizaje del investigador es muy difícil de programar en un ordenador, pero existen herramientas que pueden hacer uso de él: la inteligencia artificial.

Últimamente ha habido una explosión en el uso de las inteligencias artificiales en distintos ámbitos, ya que es un área de la informática que ha avanzado enormemente en los últimos años. Estas técnicas de inteligencia artificial y en concreto las redes neuronales son una herramienta muy poderosa en el análisis de datos, como por ejemplo en el reconocimiento de letras manuscritas o imágenes. Lo particulariza que son sistemas capaces de aprender a partir de una base de datos con la que se los entrena. En este trabajo se explora la posibilidad de aprovechar la capacidad de aprender de las redes neuronales para interpretar las medidas de un sistema y obtener así una estimación de los parámetros del hamiltoniano de espín. En particular se estudia la posibilidad de obtener estos parámetros a partir de unas medidas de capacidad calorífica. Este análisis puede considerarse como “Hamiltonian Learning” [1], donde se utilizan medidas del sistema en equilibrio para aprender el hamiltoniano del sistema con “machine learning”.

2. Sistema Físico: Hamiltoniano de espín molecular

Al describir las características de una molécula con momento magnético a bajas temperaturas es conveniente considerar únicamente los estados de menor energía, ya que son los únicos pobla-

dos. De este modo, el sistema estará muy bien descrito por un hamiltoniano de espín efectivo, que permite caracterizar la estructura de los niveles fundamentales y los estados de espín.

2.1. Estados fundamentales

Las moléculas magnéticas están compuestas por un átomo principal con un momento magnético elevado ya sea un metal de transición como el hierro o una tierra rara rodeado de ligandos diamagnéticos.

Para poder describir el sistema es necesario tener en cuenta las interacciones presentes en el ion magnético libre, como la atracción Coulombiana entre los electrones y el núcleo, la repulsión entre electrones y el acoplo espín-órbita LS . Además, los ligandos del entorno del ion magnético generan un campo eléctrico llamado campo cristalino, que desdobra parcialmente los estados degenerados y altera los autoestados del sistema.

La contribución del campo cristalino y la del acoplo espín-órbita pueden dar lugar a distintas clases de sistemas que muestran comportamientos radicalmente distintos.

En los átomos del grupo del hierro (3d) la contribución del campo cristalino es muy grande, ya que los electrones de la capa incompleta, que son los que generan el momento magnético orbital y de espín, se encuentran en capas muy externas del átomo. Esto provoca el llamado quenching del momento orbital, haciendo que el valor de la tercera componente sea nulo. Por ello el carácter de estos sistemas viene determinado por \mathbf{S} .

En los lantánidos (4f) el momento orbital es mayor y la contribución del campo cristalino es mucho menor. De este modo la contribución del acoplo espín-órbita es mayor, y el sistema está caracterizado por el momento angular total $\mathbf{J} = \mathbf{S} + \mathbf{L}$.

2.2. Hamiltoniano de espín efectivo

En esta clase de moléculas, el hamiltoniano total vendrá determinado por la suma de las contribuciones del efecto del acoplo espín-órbita, el campo cristalino y el efecto Zeeman. Por último si el núcleo tiene momento magnético no nulo será necesario considerar las contribuciones hiperfinas, así como Zeeman nuclear y cuadrupolar. En un ion (3d) el campo cristalino es mucho mayor que el acoplo espín órbita, lo que provoca que el campo cristalino actúe sólo sobre los estados de espín al estar congelado el momento orbital. La actuación conjunta de ambos términos se describe fenomenológicamente como la actuación de una serie de operadores adaptados en simetría actuando únicamente sobre el espín. A esta contribución la denominaremos hamiltoniano de anisotropía magnética H_{am} :

$$H_S = H_{e,Z} + H_{am} + H_{hf} + H_{n,Z} + H_{qp} \quad (2.1)$$

$$H_S = \mu_B \mathbf{B} \cdot \mathbf{g} \cdot \hat{\mathbf{S}} + \sum_{k,q} B_{kq} \hat{O}_k^q + \hat{\mathbf{S}} \cdot \mathbf{A} \cdot \hat{\mathbf{I}} - \mu_N g_I \mathbf{B} \cdot \hat{\mathbf{I}} + P_{\parallel} \left(\hat{I}_Z^2 - I(I-1) \right) \quad (2.2)$$

Donde \mathbf{B} es el campo magnético, $\hat{\mathbf{S}}$ el espín del átomo magnético, \mathbf{g} el tensor giromagnético, $\hat{\mathbf{I}}$ el espín del núcleo.

En este trabajo se utilizarán los datos de una molécula que contiene un ion Ni(II). Debido a que sólo uno de los cinco isótopos naturales del níquel tiene espín nuclear (^{61}Ni) y tiene una

abundancia relativa del 1% se considera que el espín nuclear del sistema es cero, por lo que todas las componentes hiperfinas se anulan y la expresión del hamiltoniano quedará por tanto:

$$H_S = \mu_B \mathbf{B} \cdot \mathbf{g} \cdot \hat{\mathbf{S}} + \sum_{k,q} B_{kq} \hat{O}_k^q. \quad (2.3)$$

La contribución correspondiente al efecto Zeeman se corresponde con la interacción entre un campo magnético con un electrón, donde el efecto Zeeman tendría una contribución $\mu_B g_e \mathbf{B} \hat{\mathbf{S}}$, con $g_e \simeq 2,0$. Al no estar el electrón libre sino en orbitales atómicos el efecto del campo magnético no es isótropo. Esta anisotropía se puede describir fácilmente sustituyendo el término g_e por un tensor \mathbf{g} . En el caso general:

$$H_{e,Z} = \mu_B \mathbf{B} \cdot \mathbf{g} \cdot \hat{\mathbf{S}} = \mu_B \begin{pmatrix} B_x & B_y & B_z \end{pmatrix} \begin{pmatrix} g_{xx} & g_{xy} & g_{xz} \\ g_{yx} & g_{yy} & g_{yz} \\ g_{zx} & g_{zy} & g_{zz} \end{pmatrix} \begin{pmatrix} \hat{S}_x \\ \hat{S}_y \\ \hat{S}_z \end{pmatrix} \quad (2.4)$$

Usualmente la contribución de los términos cruzados es muy pequeña con respecto a la diagonal, por lo que se puede simplificar la expresión, obteniendo finalmente:

$$H_{e,Z} = \mu_B \left(g_x B_x \hat{S}_x + g_z B_z \hat{S}_z + g_z B_z \hat{S}_z \right) \quad (2.5)$$

La contribución del término de anisotropía magnética se expresa como una combinación de los operadores de Stevens \hat{O}_k^q (que tiene en cuenta las simetrías del sistema y cómo afectan a los espines), junto a sus coeficientes B_{kq} .

$$H_{am} = \sum_{k,q} B_{kq} \hat{O}_k^q = \sum_{k=2,4,6} \left(\sum_{-k \leq q \leq k} B_{kq} \hat{O}_k^q \right) \quad (2.6)$$

Los operadores de Stevens \hat{O}_k^q son combinaciones de operadores de espín de orden k . Para un espín efectivo S sólo los operadores con $k \leq 2S$ son distintos de $\mathbf{0}$.

De este modo, para S pequeños el sistema puede describirse con dos términos $k = 2$ de la forma:

$$H_{am} = B_{20} \hat{O}_2^0 + B_{22} \hat{O}_2^2 = D \left(\hat{S}_z^2 - \frac{1}{3} S(S+1) \right) + E \left(\hat{S}_x^2 - \hat{S}_y^2 \right) \quad (2.7)$$

donde se ha utilizado la expresión de los operadores de Stevens en función de los operadores de espín [2]. De esta forma, el hamiltoniano de la molécula quedará determinado por:

$$H_S = \mu_B \left(g_x B_x \hat{S}_x + g_z B_z \hat{S}_z + g_z B_z \hat{S}_z \right) + D \left(\hat{S}_z^2 - \frac{1}{3} S(S+1) \right) + E \left(\hat{S}_x^2 - \hat{S}_y^2 \right) \quad (2.8)$$

2.3. Capacidad calorífica a campo cero

La capacidad calorífica de un sistema se define como la derivada de la energía con respecto a la temperatura.

Los estados posibles de los electrones vienen determinados por los autoestados del hamiltoniano (2.8), y su energía por sus respectivos autovalores. La ocupación de cada uno de los autoestados viene dada por la mecánica estadística, de la forma:

$$N_m = \frac{1}{Z} e^{-\beta \epsilon_m} \quad \text{con } Z = \sum e^{-\beta \epsilon_m} \quad (2.9)$$

donde N_m es la ocupación del autoestado m , $\beta = \frac{1}{k_B T}$, ϵ_m es el autovalor del hamiltoniano correspondiente y Z es la función de partición.

Sumando las ocupaciones de cada uno de los estados posibles, multiplicados por la energía de cada uno (ϵ_m) se obtiene una expresión de la energía del sistema:

$$U = \frac{1}{Z} \sum e^{-\beta \epsilon_m} \epsilon_m = \frac{1}{Z} \left(-\frac{d}{d\beta} Z \right) \quad (2.10)$$

La expresión de la capacidad calorífica será por tanto:

$$\begin{aligned} c &= \frac{\partial U}{\partial T} = \frac{\partial}{\partial T} \frac{1}{Z} \sum e^{-\beta \epsilon_m} \epsilon_m = \frac{\partial \beta}{\partial T} \frac{\partial}{\partial \beta} \frac{1}{Z} \sum e^{-\beta \epsilon_m} \epsilon_m = \\ &= -k_B \beta^2 \left(-\frac{1}{Z} \sum e^{-\beta \epsilon_m} \epsilon_m^2 + \frac{1}{Z^2} \sum e^{-\beta \epsilon_m} \epsilon_m \sum e^{-\beta \epsilon_m} \epsilon_m \right) = \\ c &= k_B \beta^2 \left(\frac{1}{Z} \sum e^{-\beta \epsilon_m} \epsilon_m^2 - \frac{(\sum e^{-\beta \epsilon_m} \epsilon_m)^2}{Z^2} \right) = k_B \beta^2 (\langle \epsilon_m^2 \rangle - \langle \epsilon_m \rangle^2) \end{aligned} \quad (2.11)$$

Si estudiamos el caso particular de campo nulo, los valores ϵ_m son independientes de la forma de los autoestados y por tanto también lo son U y c . Esto hace particularmente sencillo calcular las curvas de capacidad calorífica para toda T a partir de los valores D y E del hamiltoniano (2.8), siendo únicamente necesario calcular los autovalores de H_S .

2.3.1. Aproximación de c a bajas temperaturas

En el desarrollo general de la expresión de la capacidad calorífica se han utilizado los valores ϵ_m como parámetros iniciales. Pese a que están totalmente determinados por los parámetros D y E , escribir explícitamente estos valores en función de ellos es muy complicado, siendo lo más práctico calcular numéricamente su valor.

Sin embargo, si nos encontramos en un régimen de bajas temperaturas y fijamos $S = 1$, ya que es el espín del ion Ni(II) utilizado, es posible obtener $c(D, E, T)$ explícitamente.

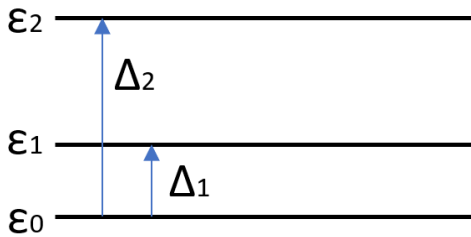


Fig. 1: Diagrama de las energías permitidas en un sistema de $S = 1$

En un sistema de espín 1 existen tres posibles autoestados en los que se pueden encontrar los electrones. En los casos en los que T sea muy pequeña (límite de bajas temperaturas) haremos esta aproximación y podremos asumir que los únicos niveles poblados serán los 2 de menor energía, ya que $k_B T \ll \Delta_2$.

Como un estado totalmente desocupado no afecta a las propiedades globales, el sistema se puede describir con únicamente dos niveles de energía, que serán:

$$\epsilon_0 = 0 \quad \epsilon_1 = \Delta \longrightarrow Z = 1 + e^{-\beta \Delta} \quad (2.12)$$

Utilizando los resultados anteriores para la capacidad calorífica se obtiene una nueva expresión para este sistema simplificado, de la forma:

$$c = \beta^2 \left(\frac{1}{Z} e^{-\beta\Delta} \Delta^2 - \frac{1}{Z^2} e^{-2\beta\Delta} \right) = \beta^2 \Delta^2 \frac{1}{Z^2} \left((1 + e^{-\beta\Delta}) e^{-\beta\Delta} - e^{-2\beta\Delta} \right) = \quad (2.13)$$

$$= \beta^2 \Delta^2 \frac{1}{Z^2} e^{-\beta\Delta} \quad (2.14)$$

Con $S = 1$, a campo magnético nulo y con las aproximaciones realizadas, H_S queda únicamente como la componente de anisotropía magnética, de la forma:

$$H_S = DS_z^2 + E(S_x^2 - S_y^2) \quad (2.15)$$

Desarrollando los operadores $S_{x,y,z}$ en la base $|S^2, S_z\rangle$ podemos obtener analíticamente los autovalores de H_S .

$$S_x = \frac{\hbar}{\sqrt{2}} \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} \quad S_y = \frac{\hbar}{\sqrt{2}} \begin{pmatrix} 0 & i & 0 \\ -i & 0 & i \\ 0 & -i & 0 \end{pmatrix} \quad S_z = \hbar \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -1 \end{pmatrix}$$

$$H_S = \hbar \begin{pmatrix} D & 0 & E \\ 0 & 0 & 0 \\ E & 0 & D \end{pmatrix}$$

Por tanto, los autovalores del hamiltoniano serán las soluciones del polinomio característico:

$$(D - \lambda)^2 \lambda - E^2 \lambda = 0 \longrightarrow \lambda_1 = 0 \quad \lambda_2 = D - E \quad \lambda_3 = D + E \quad (2.16)$$

De este modo si suponemos D y E positivos tendremos que la diferencia de energía entre los dos primeros niveles siempre será $\Delta = |D - E|$.

Con esto sustituyendo en (2.14) podemos obtener la expresión de $c(D, E, T)$:

$$c(D, E, T) = \beta^2 |D - E|^2 \frac{e^{-\beta|D-E|}}{(1 + e^{-\beta|D-E|})^2} \quad (2.17)$$

2.3.2. Contribución de la red: modelo de Debye

Otra contribución importante a la capacidad calorífica de un material viene dada por la excitación de fonones en la red. Para calcular analíticamente esta contribución se utiliza el modelo de Debye, que es el modelo realista más simple de vibraciones cristalinas que predice la existencia de fonones [3].

Este modelo se basa en la existencia de fuerzas interatómicas que tienden a mantener constante la distancia entre átomos. Se puede aproximar esta fuerza recuperadora como la fuerza elástica de un muelle de longitud natural igual a la distancia interatómica promedio a . En una red de una dimensión con M átomos el hamiltoniano del sistema quedaría como:

$$H_{Red} = \sum_{n=0}^{M-1} C \|\vec{x}_n - \vec{x}_{n+1} + a\|^2 \quad (2.18)$$

Los estados permitidos del sistema vienen determinados por los autovalores de este hamiltoniano, que se corresponden con ondas planas de longitud de onda aM/m , con m entero entre 1 y M tanto transversales como longitudinales, de la forma:

$$\Delta x_i = \cos(k_{i,m}x - \omega_{i,m}t) \quad k_{i,m} = \frac{m}{aM} \quad E_{i,m} = \hbar\omega_{i,m} \quad (2.19)$$

donde i es la rama de vibración (una longitudinal y dos transversales en tres dimensiones) y m es el modo de oscilación. A estas soluciones se las denomina fonones. Si consideramos la velocidad de propagación de estos fonones constante e independiente de su energía (y por tanto de su frecuencia) se puede obtener la relación de dispersión del sistema, obteniendo así la relación:

$$v_{prop} = \frac{\Delta x}{\Delta t} = \frac{\omega}{k} \Rightarrow \omega = v_{prop}k. \quad (2.20)$$

Una vez obtenida la relación de dispersión es necesario obtener la densidad de estados de los fonones en un cristal tridimensional. Para ello se procede a integrar todos los estados posibles en la primera zona de Brillouin, que es equivalente a integrar los posibles $k_{i,m}$ en una esfera de radio k_D en el espacio recíproco. El valor k_D es un número de onda elegido de forma que la esfera contenga M vectores diferentes de vibración permitidos, siendo M el número total de átomos en el cristal. Cada una de las soluciones ocupa un volumen en el espacio recíproco de $(2\pi)^3/V$, con V el volumen de la primera zona de Brillouin, por lo que el volumen de las M soluciones distintas ha de ser igual al de la esfera de radio k_B .

$$V_{esfera} = \frac{4\pi}{3}r^3 \quad \frac{V_{esp.reciproco}}{Vec.Solucion} = \frac{(2\pi)^3}{V_{esp.real}} \rightarrow \frac{(2\pi)^3 \cdot M}{V} = \frac{4\pi}{3}k_D^3 \quad (2.21)$$

La densidad de estados D del sistema definida como M/V en el espacio recíproco será:

$$D = \frac{k_D^3}{6\pi^2}. \quad (2.22)$$

Como resultado de estas simplificaciones se puede expresar la capacidad calorífica del sistema como:

$$c_v = \frac{\partial}{\partial T} \frac{3\hbar c}{2\pi^2} \int_0^{k_D} \frac{k^3 dk}{e^{\beta\hbar ck} - 1}. \quad (2.23)$$

Para simplificar esta integral se definen los valores de la frecuencia de Debye ω_D y la temperatura de Debye Θ_D , de la forma:

$$\omega_D = k_D v_{prop} \quad k_B \Theta_D = \hbar\omega_D = \hbar v_{prop} k_D. \quad (2.24)$$

Realizando así el cambio de variable $x = \hbar v_{prop} k / k_B T$ en (2.23) la integral puede reescribirse en términos de la temperatura de Debye:

$$c_v = 9Dk_B \left(\frac{T}{\Theta_D}\right)^3 \int_0^{\Theta_D/T} \frac{x^4 e^x dx}{(e^x - 1)^2} \quad (2.25)$$

El valor de esta integral satura a altas temperaturas, alcanzando un valor del calor específico constante $c_v = 3k_B$. Este resultado está en acuerdo con el teorema de equipartición de la energía

de la mecánica clásica. Sin embargo, ya que la señal del sistema que se estudiará es medida a bajas temperaturas, para calcular la contribución de los fonones a las medidas basta con calcular c_v en aproximación de bajas temperaturas, donde $k_B T \ll \hbar v_{prop} k \rightarrow x \gg 1$. El resultado de esta aproximación será así una contribución proporcional a T^3 :

$$c_v = \frac{12\pi^4}{5} D k_B \left(\frac{T}{\Theta_D} \right)^3. \quad (2.26)$$

3. Redes Neuronales

Con el fin de obtener los parámetros D y E del hamiltoniano es necesario diseñar una función que pueda tomar como datos de entrada la curva de capacidad calorífica a estudiar y devuelva los dos valores deseados.

$$c(T) \xrightarrow{f_0} (D, E) \quad (3.1)$$

Este proceso sin embargo resulta muy complicado en funciones de esta naturaleza, y realizar ajustes numéricos tradicionales (como minimización de cuadrados, por ejemplo) no es posible, debido a la naturaleza matricial del hamiltoniano. Al necesitar obtener los autovalores de una matriz no se puede expresar el calor específico mediante una fórmula analítica, haciendo imposible así su ajuste numérico.

Para resolver este problema en este trabajo se proponen las redes neuronales. Las redes neuronales son sistemas que permiten desarrollar una inteligencia artificial mediante un entrenamiento con una base de datos. Tras un entrenamiento con datos similares a los que se desea analizar posteriormente, el sistema es capaz de aprender, e idealmente alcanzaría el comportamiento deseado.

3.1. La neurona

Las redes neuronales son funciones complejas formadas por la unión de múltiples operadores simples, a los que llamaremos neuronas [4].

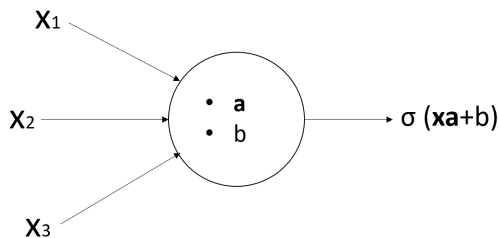


Fig. 2: Neurona, parte mínima de la red.

Cada una de estas neuronas toma una serie de valores como entrada y devuelve un único valor como respuesta.

Esta dependencia funcional de cada una de las neuronas puede ser descrita por un vector de activación (\vec{a}), un bias b y una función de activación $\sigma(*)$, que ha de ser monótona no lineal para facilitar el aprendizaje.

$$y = \sigma(\vec{x}\vec{a} + b) \quad (3.2)$$

El comportamiento de esta neurona es muy simple, pero con un conjunto grande de ellas puede llegar a mostrar comportamientos muy complejos, siendo estos sistemas un sistema de computación universal.

3.2. Red neuronal: Capas sucesivas

Las neuronas se pueden organizar en multitud de configuraciones distintas, permitiendo comportamientos de la red arbitrariamente complejos.

Los modelos más sencillos están constituidos por múltiples capas de neuronas, donde cada una de las neuronas de una capa está unido a todas las neuronas de la capa anterior. Esta configuración es particularmente sencilla de entrenar, aunque los modelos con muchas capas pueden aprender lentamente y requieren un mucho mayor poder computacional.

Por ello se utilizará el modelo más simple no trivial de red neuronal, una disposición de una única capa oculta.

Hay que tener en cuenta que, pese a que este es un modelo sencillo de red neuronal, va a ser utilizado para interpretar un hamiltoniano muy sencillo. De este modo la sencillez del modelo sirve como prueba de concepto al mismo tiempo de que no requiere de un poder computacional desorbitadamente alto.

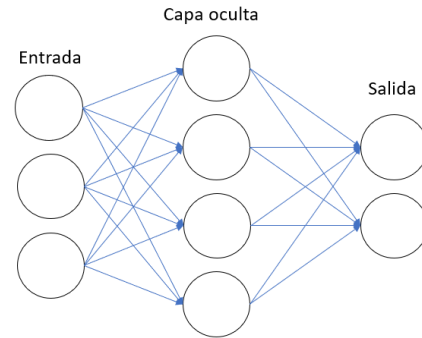


Fig. 3: Red neuronal utilizada: una única capa oculta.

3.3. Entrenamiento de la red

Para conseguir que la red neuronal adopte el comportamiento deseado es necesario entrenarla

con una base de datos que se asemeje a los datos que posteriormente se quieren procesar.

Este proceso de entrenamiento es complicado y puede ser llevado a cabo de múltiples formas en función de la base de datos de la que disponemos, las características que esperamos de la red o del poder computacional disponible.

Las dos formas de entrenamiento más comunes son:

-No supervisado: la base de datos de entrenamiento sólo contiene la información que se quiere procesar, y no se tiene una etiqueta asociada a cada fichero que contenga la información esperada de la red. La red aprende a agrupar ficheros que considera similares, pese a que no se ha aportado información externa. Es muy útil en reconocimiento de imágenes o en análisis de datos, ya que descubre patrones en la información que pueden ayudar a entender mejor los datos [5].

-Supervisado: junto a la información de la base de datos se tiene una etiqueta que indica el valor que se espera recibir de la red. De este modo a red aprende a encontrar una unión entre la información suministrada y su etiqueta correspondiente. El funcionamiento de este entrenamiento es más sencillo, pero requiere de una base de datos donde se conoce perfectamente la respuesta que se espera de cada fichero. En este trabajo utilizaremos el aprendizaje supervisado, ya que al interpretar las medidas se pretende obtener los valores D y E , por lo que es necesario entrenar a la red para que los proporcione. Si utilizásemos aprendizaje no supervisado la red aprendería a distinguir curvas de susceptibilidad, pero sería imposible determinar qué parámetros utiliza para ello.

Para entrenar a la red es necesario definir una función de coste que cuantifique lo buena o

lo mala que es la predicción de la red en base a la etiqueta asociada a los datos. Una función de coste muy sencilla que se utilizará en el entrenamiento es la suma de las diferencias al cuadrado de los parámetros reales y los proporcionados por la red:

$$C(x_0) = (D_0 - D_{Red})^2 + (E_0 - E_{Red})^2 \quad (3.3)$$

donde x_0 es una curva de capacidad calorífica, D_0 y E_0 son los parámetros reales de la curva proporcionada y D_{Red} y E_{Red} son los predichos por la red.

El objetivo del entrenamiento es que esta función de coste sea mínima para toda la base de datos. Esto se lleva a cabo minimizando la función de coste mediante un proceso llamado backpropagation, que permite determinar el gradiente de esta función con respecto a los parámetros de la red. Este gradiente se calcula para un conjunto pequeño de datos, se promedia y se modifican los valores para reducir el valor de la función de coste lo más rápidamente posible. El tamaño del conjunto de datos utilizado para calcular el gradiente se denomina “batch size”.

Existen sin embargo una serie de problemas que pueden ocurrir durante el entrenamiento que reducirían considerablemente la eficacia de la red neuronal. Uno de ellos es el sobreentrenamiento u “overfitting”, que ocurre cuando se dispone de una base de datos demasiado pequeña. Si se entrena demasiado a la red con poca diversidad de datos es posible que la red termine “memorizando” los datos de entrenamiento, lo que empeora considerablemente el rendimiento.

Para poder conocer el rendimiento de la red parte de los datos disponibles se dedican a evaluar a la red. Estos datos no pueden ser los mismos con los que la red se entrena, ya que para que la evaluación sea fiable es necesario probar a la red sobre datos que no haya visto antes. Para ello, la base de datos original se divide en dos grupos, el set de entrenamiento y el set de validación. El tamaño del set de validación N no puede ser muy pequeño ya que se desean obtener evaluaciones precisas y de no haber muestras suficientes las desviaciones estadísticas en la evaluación del rendimiento serían muy grandes. Tampoco queremos que sea muy extenso, ya que realizar las evaluaciones lleva tiempo y requeriría de muchos datos que no serían usados en el entrenamiento. Por ello se utiliza típicamente en torno a un quinto del total de la base de datos como set de validación y el resto como set de entrenamiento.

Con el fin de ajustar la velocidad de aprendizaje de la red se introduce un hiperparámetro η . Este valor permite determinar cuánto se han de modificar los parámetros de la red en base al valor del gradiente de la función de coste. Este valor junto al “batch size” y el número de veces que se reutiliza la base de datos completa (epochs) constituyen el llamado conjunto de hiperparámetros de la red. El valor de estos hiperparámetros es muy importante ya que permiten ajustar el funcionamiento de la red, siendo posible que la red no aprenda en absoluto si el valor de los hiperparámetros es muy lejano al óptimo.

En principio no hay un límite al tamaño de estas redes neuronales, y cuanto mayor sea el tamaño de ésta y su complejidad, mayor potencia de cálculo se podrá alcanzar. Sin embargo, entrenar a la red es un proceso muy lento, y cuanto más grande es mayor cantidad de datos son necesarios. Esto supone un aumento muy significativo del poder computacional necesario para entrenar la red.

3.4. Base de datos: datos sintéticos

Ya hemos hecho énfasis en la necesidad de tener una base de datos extensa para poder entrenar correctamente a la red neuronal. Sin embargo, las medidas experimentales son lentas de producir y requieren mucho esfuerzo, además de que es complicado asignar los valores de D y E a estas medidas (es el problema que se desea abordar en este trabajo). Esto hace totalmente inviable entrenar a la red con datos reales.

Para poder entrenar a la red neuronal se plantea una solución alternativa, que es utilizar datos sintéticos. Ya que calcular numéricamente la curva de capacidad calorífica es un proceso poco costoso (como se ha visto anteriormente 2.11) es posible generar aleatoriamente los parámetros D y E , calcular su curva correspondiente y utilizar estos datos para entrenar a la red.

Sin embargo los datos obtenidos de esta forma no se parecen a una medida real, ya que son curvas de capacidad calorífica “perfectas” que carecen de otras contribuciones, como puede ser la red o los términos hiperfinos. Una medida real tiene tanto ruido inherente a la medida como otras contribuciones sumados a la señal, y para interpretar correctamente las medidas sería necesario entrenar a la red incluyendo estas contribuciones en los datos de entrenamiento.

El ruido aleatorio y las otras contribuciones introducidas dependerán de qué componentes se observen en las medidas realizadas, y será necesario calibrar la amplitud e importancia de cada una para asemejarse lo más posible a las medidas que se desea estudiar.

3.5. Desarrollo del código

Los procesos que se han descrito son las bases para implementar una red neuronal y entrenarla correctamente. Sin embargo escribir el código siguiendo los pasos descritos resulta en un aprendizaje muy poco optimizado y lento.

Con el fin de minimizar la potencia computacional necesaria para entrenar a la red se ha decidido utilizar bibliotecas con estas funciones predefinidas, ya que han sido desarrolladas por profesionales y están muy bien optimizadas. En concreto, para desarrollar el código de esta red neuronal se ha utilizado la librería de Google TensorFlow [6], que presenta herramientas muy útiles en el desarrollo de redes con estas características.

4. Desarrollo del modelo

Una vez explorados los conceptos teóricos sobre el sistema físico que se desea estudiar y las redes neuronales se procede a comprobar la eficacia de la inteligencia artificial, más concretamente de las redes neuronales, en la interpretación de las curvas de calor específico. Para interpretar estas curvas se plantea una red neuronal con una capa oculta, donde la primera capa tiene los valores del calor específico calculados y la última capa consta de 2 neuronas, cuyas salidas serán los valores estimados de D y E .

En la parte puramente teórica de este trabajo se utilizará un sistema de unidades donde $k_B = 1$, $\hbar = 1$ para simplificar los cálculos. Por ello, pese a que los parámetros D y E tengan magnitud de energía no se indicará en qué unidades se miden, ya que no es un sistema de unidades convencional y todos los sistemas que respeten el valor de las constantes \hbar y k_B son

igualmente válidos. Cuando tratemos con medidas experimentales se reescalarán las constantes k_B y \hbar , de modo que las unidades de energía cm^{-1} y Kelvin (K) como unidad de temperatura.

Con el fin de ajustar los hiperparámetros de la red neuronal y comprobar si el sistema es capaz de interpretar los datos sintéticos se realiza un análisis preliminar, entrenando a una red neuronal muy simple con una base de datos pequeña, de forma que se pueda estimar rápidamente la validez de cada hiperparámetro. Para ello se genera una base de datos con 20000 curvas de calor específico en un sistema de espín 1. Se utilizan 4000 de ellas como set de validación y las 16000 restantes como set de entrenamiento. Se utiliza un rango de energías entre $k_B T = 0$ y $k_B T = 10$ con 100 puntos y 30 neuronas en la capa oculta.

Tras optimizar los hiperparámetros estudiando el valor de la función de coste tras el entrenamiento se encontró que el sistema aprendía de forma satisfactoria con $\eta = 5$, “batch size”=10, “epochs”=5. También se encontró que la función de activación de las neuronas que mejores resultados daba es la función ReLU:

$$\sigma(x) = ReLU(x) = \begin{cases} x & \text{si } x > 0 \\ 0 & \text{si } x \leq 0 \end{cases} \quad (4.1)$$

Una vez entrenada a la red definiremos el error en la predicción del sistema como el promedio del valor absoluto de la diferencia entre los parámetros reales y las predicciones de la red:

$$\Delta D = \frac{1}{N} \sum_{i=0}^N |D_{Dat,i} - D_{Red,i}| \quad \Delta E = \frac{1}{N} \sum_{i=0}^N |E_{Dat,i} - E_{Red,i}| \quad (4.2)$$

donde D y E son los parámetros del hamiltoniano y N el tamaño del set de validación.

Generando los valores D y E aleatoriamente, acotados ambos en el rango (0,5) y usando sus curvas de calor específico correspondientes se entrena a la red y se comparan los valores de los parámetros predichos del set de validación con los reales, obteniendo los errores:

$$\boxed{\Delta D = 0,678 \quad \Delta E = 0,235}. \quad (4.3)$$

Se puede observar en las figuras 4 que la estimación de la red es bastante buena en la mayoría de estos casos. Sin embargo, se observan problemas consistentes en la región de bajas temperaturas, así como una predicción bastante desacertada para la sexta curva.

Estos resultados demuestran que las redes neuronales se pueden utilizar para ajustar este tipo de curvas, obteniendo resultados satisfactorios en algunos casos.

4.1. Dependencia con el tamaño de la base de datos y el tamaño de la red

Ahora se estudiará cómo evolucionan los resultados al cambiar distintos parámetros de la red como el tamaño de la capa oculta, el número de puntos de la curva o el número total de datos. Esto permitirá optimizar la utilización de la red neuronal, ya que un mayor número de neuronas supone una mayor capacidad de cálculo, pero también requiere de mayores recursos de computación, ralentizando enormemente el proceso de aprendizaje.

Para estudiar estas dependencias se entrenará a la red con distintas bases de datos, variando en tamaño entre 5000 y 500000 curvas distintas. También se modificará la resolución de las curvas de capacidad calorífica, utilizando curvas de 50, 100 y 200 puntos a intervalos iguales de

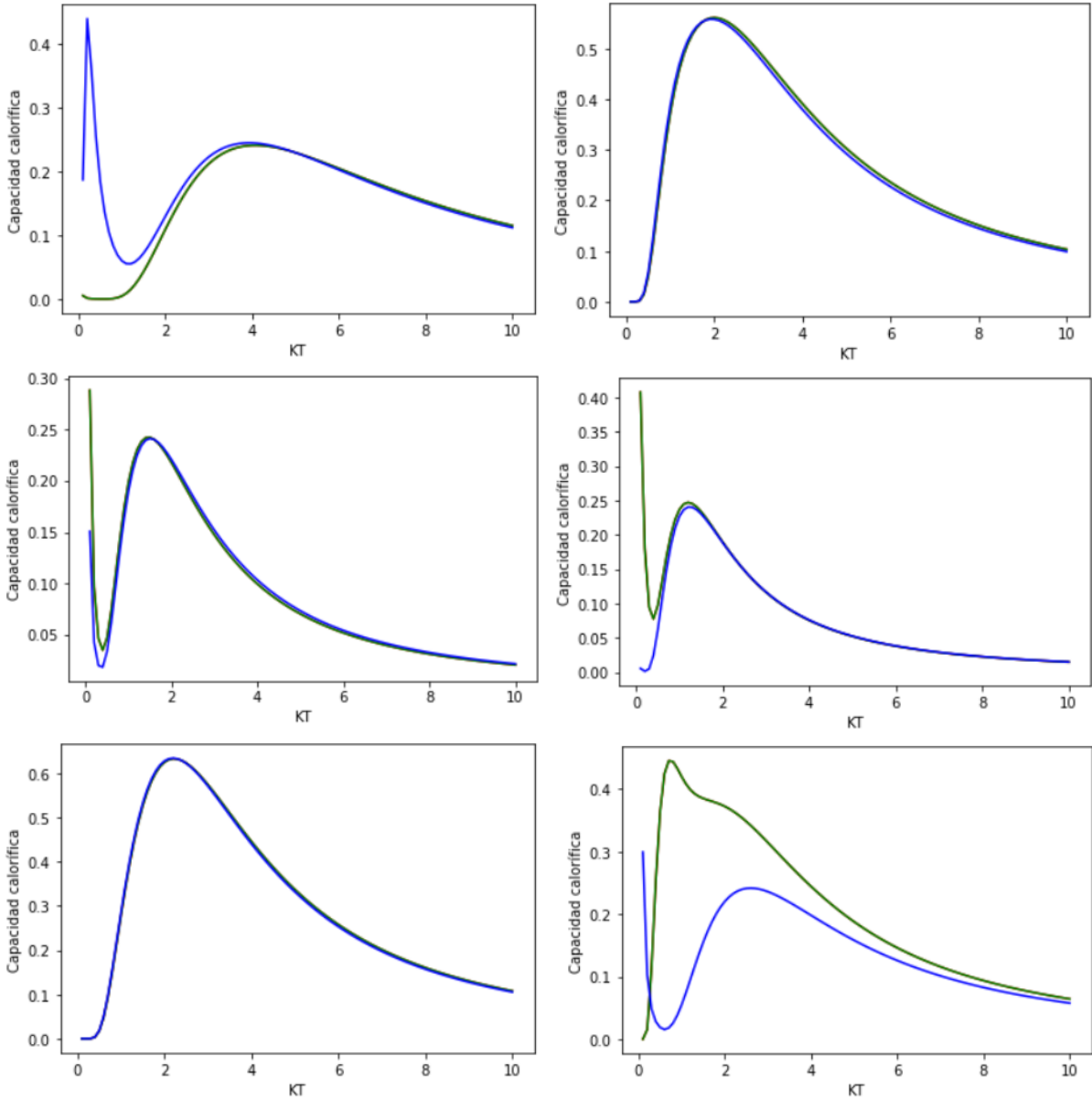


Fig. 4: Seis primeras curvas del set de validación (verde) y la predicción de la red (azul). Este es una muestra de los resultados del entrenamiento de la red simple como análisis preliminar.

temperatura en el rango $k_B T = (0, 10)$. Por último se utilizarán redes neuronales distintas con una capa intermedia de tamaños 15, 30 y 50.

Como estimador de la bondad de las estimaciones de las distintas redes se utilizará el promedio de la suma de las diferencias entre los valores predichos y los reales al cuadrado: el valor de la función de coste final (3.3).

En este análisis se utilizan los mismos hiperparámetros que en el análisis preliminar anterior, a excepción del número de veces que se entrena a la red con la base de datos. Ahora se entrena utilizando 20 veces la base de datos en lugar de 5 (epochs=20) ya que se desea observar el rendimiento óptimo de estas redes por lo que se las entrena durante más tiempo. Los rendimientos obtenidos son expresados en la figura 5.

En estas gráficas se puede observar que el rendimiento de todas las redes mejora consistente-

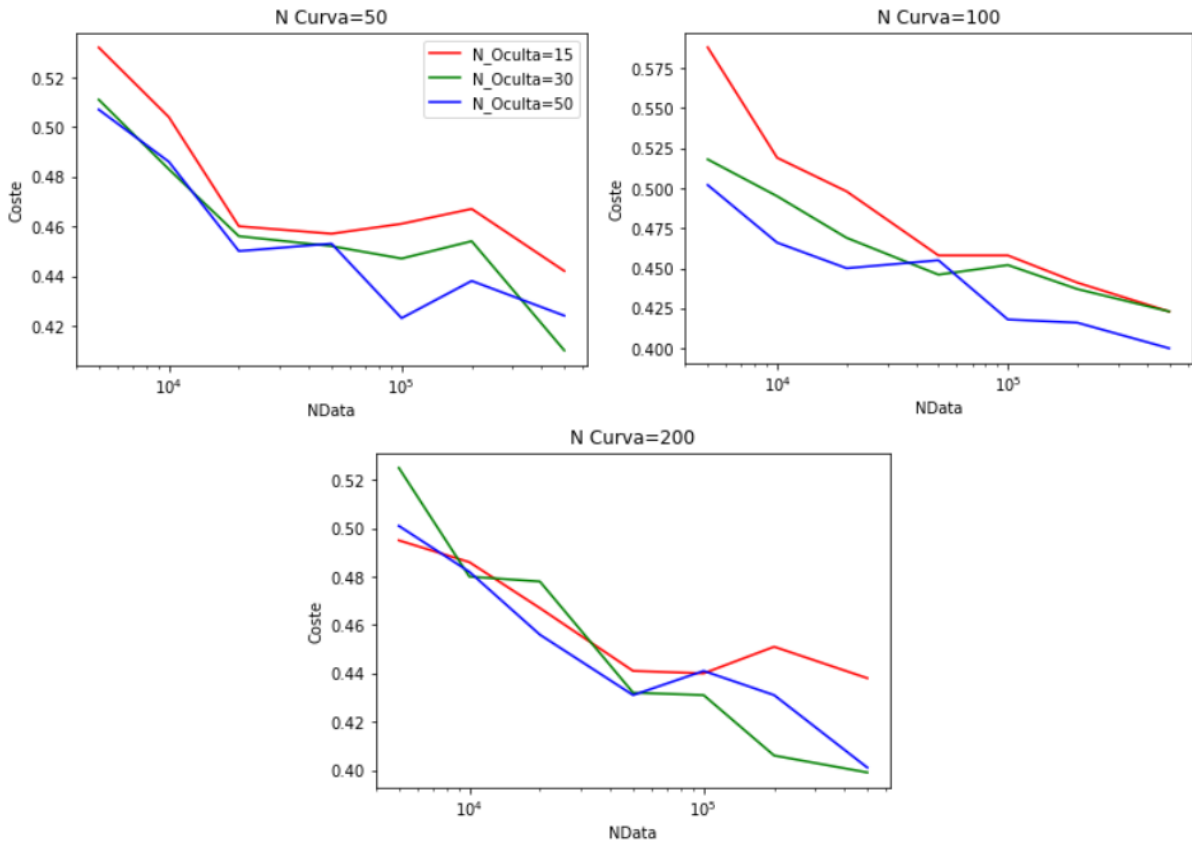


Fig. 5: Dependencia del coste final en función del tamaño de la red, el número de puntos de las curvas de entrenamiento y el tamaño de la base de datos. Las distintas curvas de colores representan los distintos tamaños de la capa oculta, con 15 neuronas en la capa oculta (rojo), 30 (verde) y 50 (azul). Cada una de las gráficas representa el rendimiento de las redes al interpretar curvas con distinta resolución. El número de puntos de las curvas está indicado en la parte superior de cada una de las tres figuras.

mente al aumentar el tamaño de la base de datos. También se observa que los resultados mejoran al añadir puntos a las curvas de susceptibilidad, ya que al suministrarle más información a la red puede obtener resultados más precisos. Sin embargo, se observa que la red necesita un tamaño mínimo para procesar curvas con muchos puntos, ya que la red con 15 neuronas en la capa oculta no presenta mejoras significativas al superar los 100 puntos en la curva de capacidad calorífica, mientras que las otras dos redes sí que muestran mejoras al aumentar la resolución de los datos de entrada.

Teniendo en cuenta este hecho, puede parecer que lo ideal es utilizar una red con el mayor número posible de neuronas en la capa oculta y entrenarla con una base de datos también lo más grande posible. Sin embargo, es necesario tener en cuenta que entrenar una red grande es notablemente más lento que las pequeñas, ya que el número de parámetros a ajustar crece rápidamente al añadir más neuronas a la capa intermedia y además al ser necesarias base de datos muy grandes el tiempo necesario para entrenarlas crece enormemente. Las redes grandes requieren de bases de datos más grandes y de más epochs para ser entrenadas correctamente, por lo que es necesario llegar a un compromiso entre obtener el mejor rendimiento posible y la potencia computacional necesaria para obtenerlo.

Por todo esto a lo largo de este trabajo se utilizarán redes de 50 neuronas en la capa oculta,

una base de datos con 200000 curvas y 100 puntos en cada curva, ya que se ha considerado que una red de estas características obtiene resultados aceptables sin ser excesivamente lenta de entrenar.

Características	
Tamaño base de datos	200000
Resolución	100 puntos
Neuronas en capa oculta	50

Con esta configuración y repitiendo el análisis realizado en 4.3 se obtienen unos nuevos valores de los errores esperados, que pasan a ser:

$$\boxed{\Delta D = 0,622 \quad \Delta E = 0,214}. \quad (4.4)$$

4.2. Estudio del aprendizaje de la red: Aproximación a bajas temperaturas

Como se ha visto anteriormente el calor específico a bajas temperaturas viene totalmente determinado por $|D - E|$ (2.14). Al sólo depender de un único parámetro y estar entrenando a la red con los dos valores D y E se puede observar que se tiene un grado de libertad y la asignación de parámetros deja de ser única.

Un método numérico tradicional de ajuste parametrizaría los valores D y E incorrectamente, pero sería capaz de estimar el valor $|D - E|$ con mucha precisión. Sin embargo, la red neuronal planteada no es un método numérico y los mecanismos exactos que utiliza para determinar ambos valores no pueden ser conocidos con precisión.

Este sistema pone a prueba la capacidad de la red para encontrar correlaciones entre los parámetros finales. Si la inteligencia artificial aprende a parametrizar deduciendo que el valor relevante es $|D - E|$ y calcula D y E de forma que se mantenga su diferencia, la red debería reproducir con mucha exactitud las curvas en un rango de temperaturas más reducido. Por ello se entrena a la red en el rango de temperaturas $k_B T = (0, 1)$, frente al anterior $(0, 10)$.

Para comprobar la bondad de las predicciones en este régimen no sólo se calculan los errores promedio en D y E , sino también el de su diferencia $|D - E|$, que es la que determina el comportamiento del sistema a bajas temperaturas:

$$\boxed{\Delta D = 0,693 \quad \Delta E = 0,258 \quad \Delta|D - E| = 0,763} \quad (4.5)$$

Se puede observar que la red estima los parámetros D y E algo peor que la utilizada anteriormente. Sorprendentemente, estima el módulo de su diferencia peor que cada uno de los parámetros individuales. Esto es un claro indicador de que la red no puede aprender la correlación esperada entre los parámetros finales y en su lugar estima cada uno de ellos de forma independiente. Esta falta de correlación entre ambos parámetros resulta en estimaciones muy lejanas a las curvas del set de validación, como se observa en la figura 6.

Se observa que la red no puede interpretar el comportamiento del sistema correctamente en este rango de temperaturas, por lo que se plantea un método alternativo de entrenamiento. Sabiendo que el parámetro $|D - E|$ describe muy bien el comportamiento del sistema a bajas temperaturas, podemos entrenar a la red con este parámetro en lugar de hacerlo con D y E

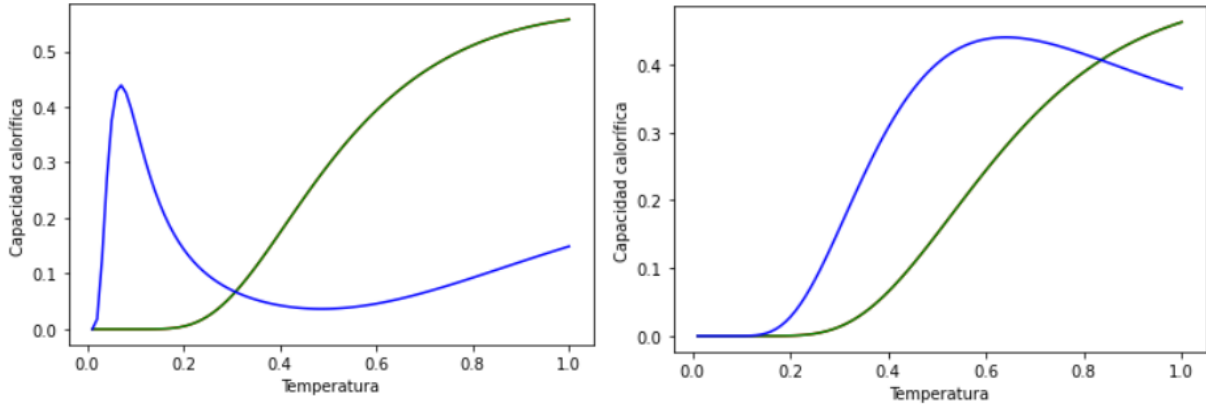


Fig. 6: Representación de las predicciones de la red entrenada para interpretar calor específico a bajas temperaturas. Curvas del set de validación (verde) junto a las predichas por la red 1 (azul).

por separado. De este modo sólo se le suministra a la red el parámetro relevante y no sería necesario que aprendiera esta dependencia en el entrenamiento. Entrenando a la red de este modo sólo se puede obtener la aproximación a bajas temperaturas de la curva a analizar, pero esta aproximación debería ajustarse muy fielmente a la curva en el rango de temperaturas utilizado $k_B T = (0, 1)$. Al entrenar a la red de esta forma se obtienen unos ajustes como los mostrados en la figura (7) y un error en el parámetro $|D - E|$:

$$\Delta|D - E| = 0,0243 \quad (4.6)$$

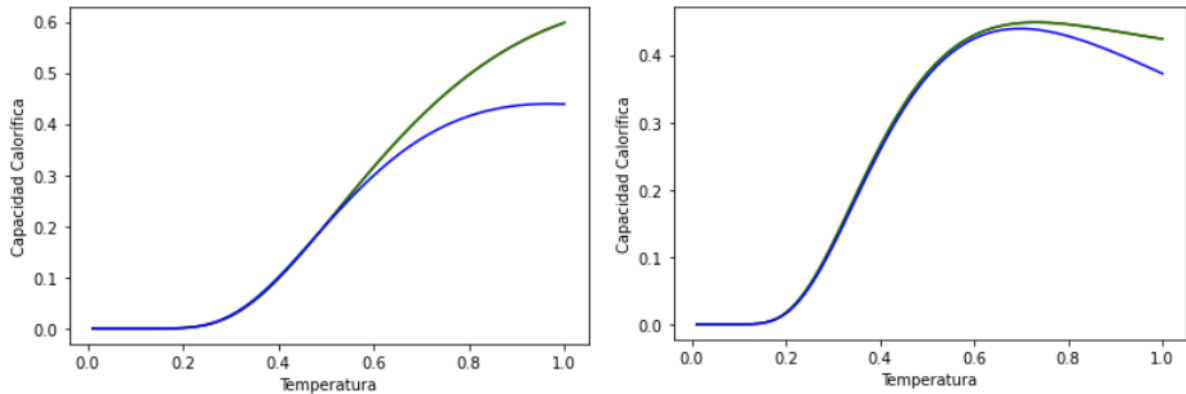


Fig. 7: Representación de las predicciones de la red entrenada a bajas temperaturas utilizando el parámetro $|D - E|$. Curvas del set de validación (verde) junto a la aproximación de bajas temperaturas predicha por la red 2 (azul).

Esta nueva forma de entrenar a la red es claramente más efectiva que utilizar directamente los parámetros D y E sin procesar, como se observa al comparar los errores esperados del parámetro $|D - E|$, llegando a resultados mucho más fiables y consistentes. Esta mejora tan considerable en los ajustes plantea la posibilidad de cambiar los parámetros con los que entrenamos a la red neuronal para obtener mejores resultados, ya que si a partir de los parámetros originales se pueden obtener nuevos parámetros que describan particularmente bien el comportamiento del sistema en algún rango de temperaturas estos parámetros serán mucho más fáciles de aprender, mejorando mucho la eficacia de la red neuronal.

Para aprovechar este buen resultado a bajas temperaturas uno de los parámetros con el que se plantea entrenar a la red será $|D - E|$, que como ya hemos visto puede ser determinado con mucha precisión. Otro parámetro será E , que presentaba un error menor que D en los ajustes anteriores, y un tercer parámetro C , que indica si D es mayor que E o viceversa (de lo contrario sería imposible obtener los parámetros D y E a partir de los anteriores).

Con este nuevo entrenamiento, se puede repetir el análisis original de 4.3, obteniendo los resultados siguientes. En la figura (8) se representa una selección de los ajustes obtenidos con los valores de D y E predichos junto a los del set de validación correspondientes.

$$\Delta D = 0,696 \quad \Delta E = 0,225 \quad (4.7)$$

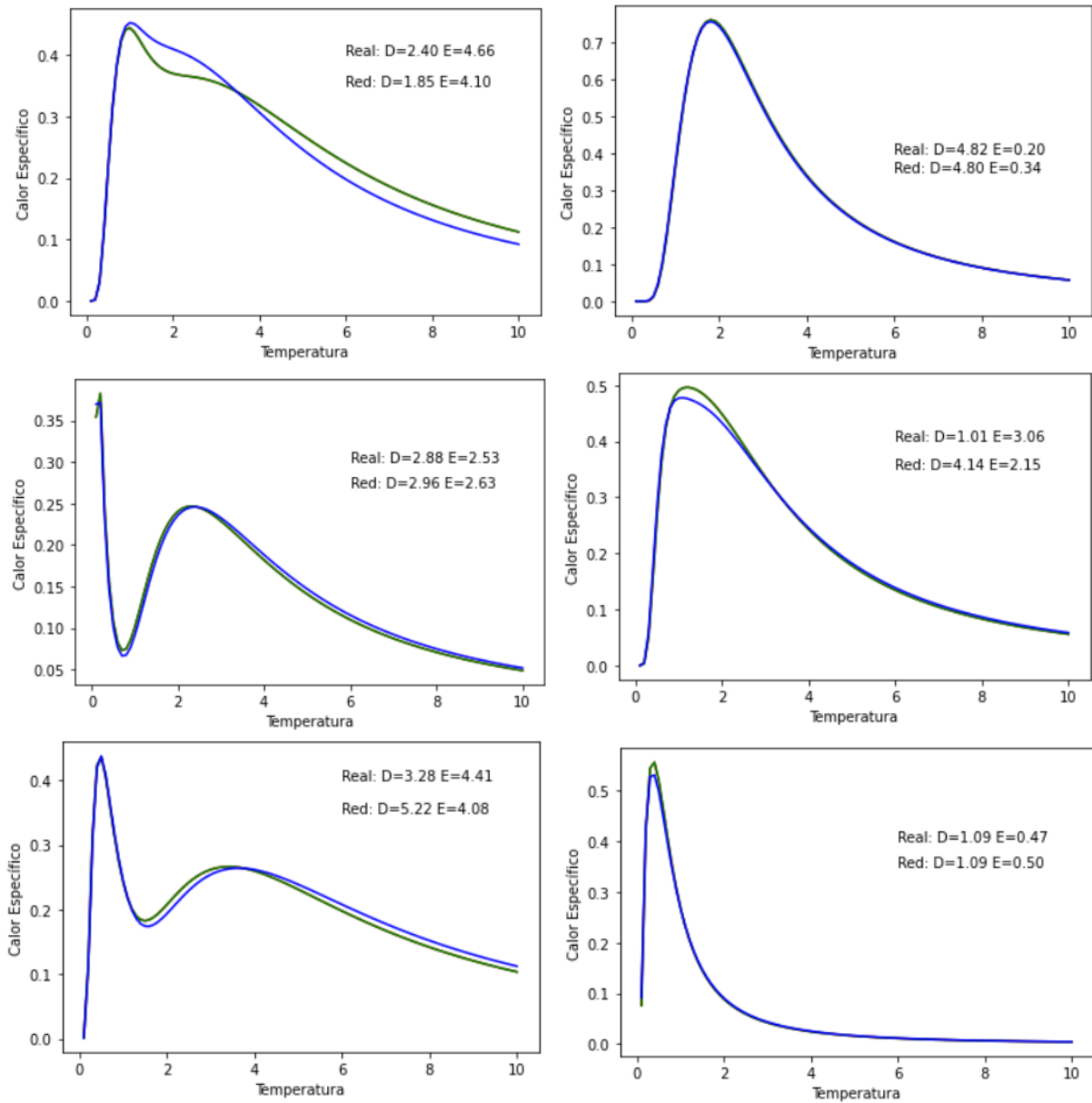


Fig. 8: Selección de curvas del set de validación (verde) y la predicción de la red con los nuevos parámetros (azul) $|D - E|$ y E .

Los errores en los parámetros D y E no se han reducido considerablemente, obteniendo

valores para los errores esperados muy similares a los de las redes anteriores. Sin embargo se observa una mejora muy significativa en la capacidad de la red para reproducir la curva original, obteniendo curvas de capacidad calorífica mucho más similares a las proporcionadas que cualquier otra red anterior.

Estos resultados sugieren que obtener errores más pequeños en los valores de D y E en este sistema es complicado, ya que las curvas de capacidad calorífica son muy similares para valores distintos de estos parámetros.

Esta degeneración de las curvas de calor específico del sistema no puede romperse únicamente con estos datos, por lo que si se pretende obtener valores precisos de D y E será necesario proporcionar más información, como por ejemplo curvas de susceptibilidad magnética.

4.3. Introducción de distorsiones sintéticas

Ya que el objetivo principal de este trabajo es comprobar la capacidad de una inteligencia artificial simple para interpretar datos experimentales, se busca generar bases de datos con curvas sintéticas de capacidad calorífica que se asemejen todo lo posible a datos experimentales reales. Anteriormente se ha comprobado que estas redes pueden reproducir fielmente curvas de calor específico, interpretando curvas muy similares a las originales (aunque los parámetros D y E puedan ser algo distintos). Sin embargo unos datos obtenidos experimentalmente nunca tendrán esta forma, ya que en las medidas aparecerán tanto ruido aleatorio como fondos sistemáticos o contribuciones características de la muestra en cuestión.

El ruido aleatorio inherente a la medida es fácil de caracterizar, y se correspondería con sumar en cada punto de la curva un número real con distribución gaussiana. Para variar la forma del ruido entre cada curva, la dispersión de este ruido se asigna aleatoriamente en un rango de valores $(0, 0.015)$, de forma que sea notablemente inferior a la señal original.

$$\text{Ruido Aleatorio} \longrightarrow \text{Distribución aleatoria} : \left(e^{-Ax^2} \right) \quad (4.8)$$

En las medidas de capacidad calorífica suele aparecer un fondo correspondiente a los fonones de la red (2.26). Esta contribución crece con T^3 en el rango de temperaturas estudiado, y se añade a las curvas de susceptibilidad como una curva BT^3 . Para estimar el valor de B se elige un valor máximo de modo que la contribución de los fonones sea del mismo orden que la de la señal a estudiar. De este modo el valor B se elige aleatoriamente en un rango de valores $(0, 0,00005)$.

$$\text{Contribución de la red} \longrightarrow \text{Dependencia cúbica con } T : c_{red} = (B \cdot T^3) \quad (4.9)$$

Una vez se ha generado una base de datos con estas distorsiones incluidas se procede a entrenar la red de la misma forma en la que se ha hecho previamente. Utilizando los nuevos parámetros $|D - E|$ y E se llega a los resultados 9 con unos errores esperados:

$$\boxed{\Delta D = 0,77 \quad \Delta E = 0,23} \quad (4.10)$$

Se puede observar que en muchos de los casos la señal original es indistinguible de la predicha por la red, lo que indica que este método de ajuste puede ser una herramienta muy útil en la interpretación de estas curvas. Sin embargo, la degeneración presente en las curvas de la sección anterior también está presente aquí, por lo que aunque la forma de las curvas sea muy parecida

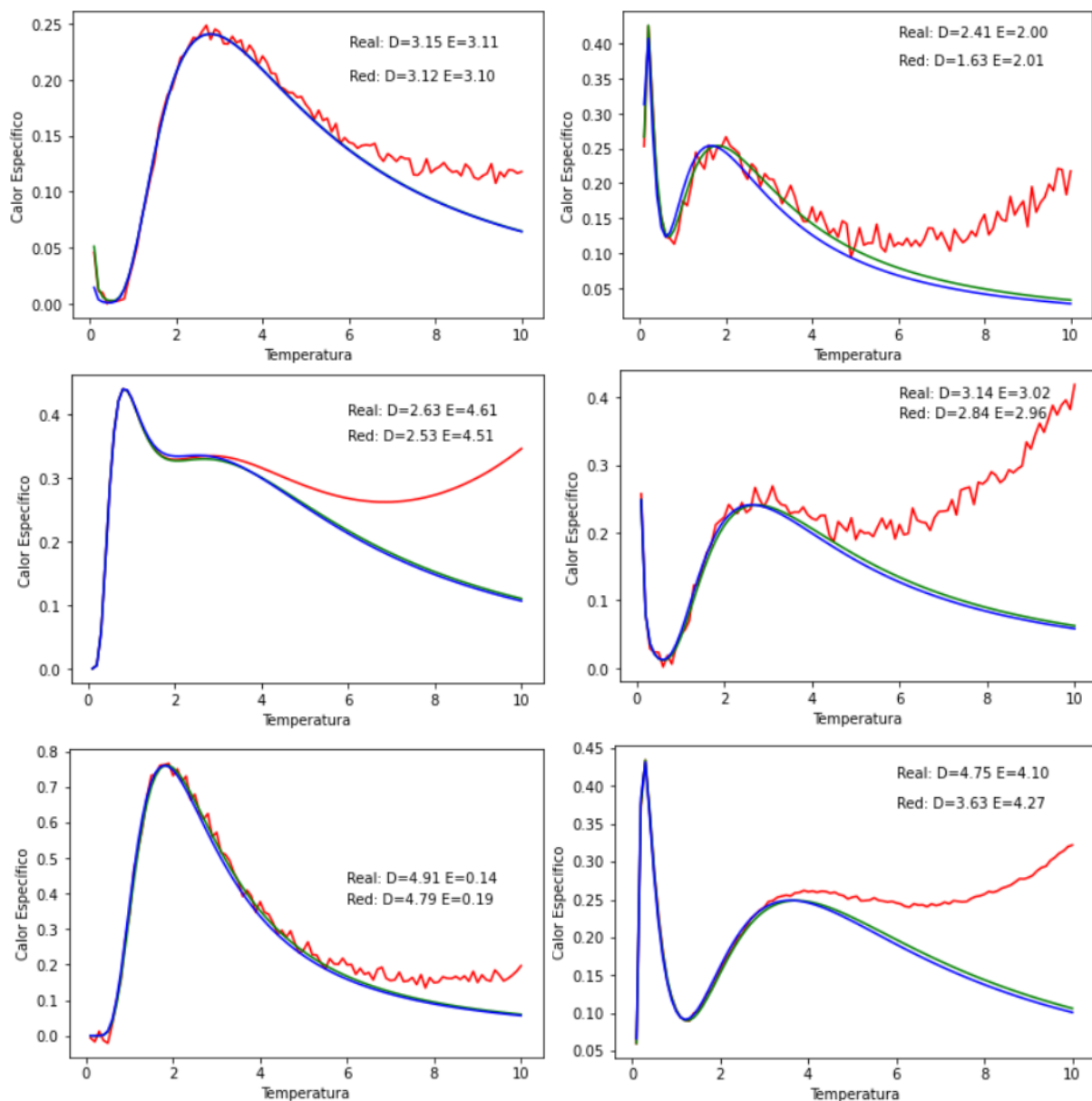


Fig. 9: Selección de curvas del set de validación con ruido y distorsiones añadidas (rojo), la señal original (verde) y la predicción de la red (azul). Esta base de datos es lo más parecido a una señal real que se ha generado de forma sintética.

es imposible obtener valores precisos de D y E con seguridad únicamente con las curvas de calor específico a campo cero.

5. Medidas experimentales

Ahora se utilizará una red neuronal para obtener los valores de D y E esperados en unas medidas experimentales reales [7]. Se utilizarán las medidas de capacidad calorífica de un compuesto de Ni(II) en un complejo molecular, cuyo comportamiento se aproxima muy bien por el hamiltoniano de anisotropía 2.8 de espín 1. Se tomaron dos medidas distintas a campo 0, cada una en un compuesto distinto con el fin de determinar la contribución de la red a las medidas.

Para analizar estas medidas es necesario caracterizar las contribuciones y el ruido que puede haber en ellas de forma que podamos entrenar a la red neuronal correctamente.

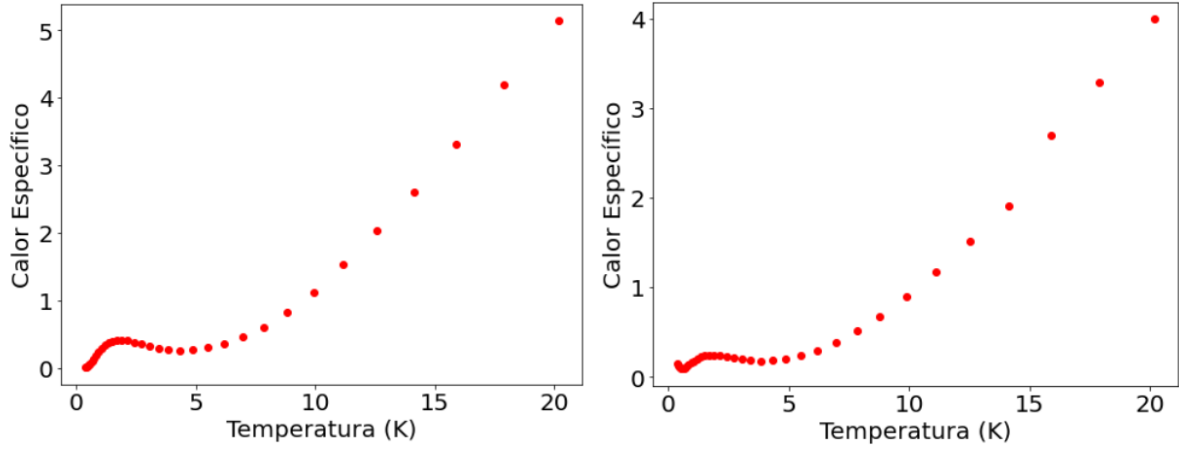


Fig. 10: Medidas de capacidad calorífica de una muestra de una molécula con un ion Ni(II) en dos medios distintos a campo nulo.

La contribución de la red es muy considerable, siendo a altas temperaturas mucho mayor que la señal que nos concierne. Además, apenas aparece ruido aleatorio en las medidas, por lo que se plantea no utilizarlo en el entrenamiento de la red.

Sin embargo, para poder entrenar a la red correctamente es necesario conocer un posible rango de valores en el que se puedan encontrar los parámetros D y E . Para conocer el orden de magnitud de los parámetros correspondientes a estas medidas utilizaremos la parametrización realizada en el artículo original [7], donde asignaba los siguientes valores representados en la figura (11):

$$D = -8,3 \text{ cm}^{-1} \quad E = 1,45 \text{ cm}^{-1} \quad (5.1)$$

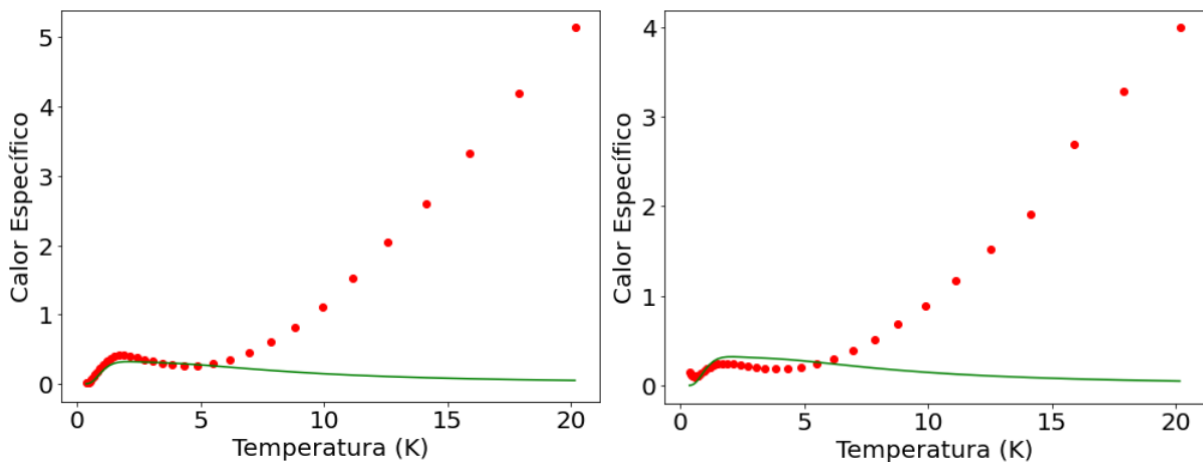


Fig. 11: Medidas de capacidad calorífica de una muestra de una molécula con un ion Ni(II)(rojo) y ajuste aportado en el paper original(verde) [7].

Estos valores se utilizarán para determinar el orden de magnitud de los valores D y E en los que se entrena a la red. Para realizar el entrenamiento se escoge un rango amplio de

posibles valores, eligiendo así un rango entre -5 y -20cm^{-1} para D y entre 0 y 5cm^{-1} para E . Para caracterizar el fondo se ajusta el rango de valores posibles para la amplitud del fondo dependiente de T^3 de forma que sea similar al de las medidas, y no se utilizará fondo aleatorio ya que no se observa ningún ruido de amplitud significativa. Entrenando a la red en los valores de temperatura de las medidas con el fondo caracterizado de la misma forma que en 9 se obtienen los valores siguientes, representados en la figura 13.

$$\boxed{D_1 = -74 \quad E_1 = 0,24 \quad D_2 = -55 \quad E_2 = 1,5} \quad (5.2)$$

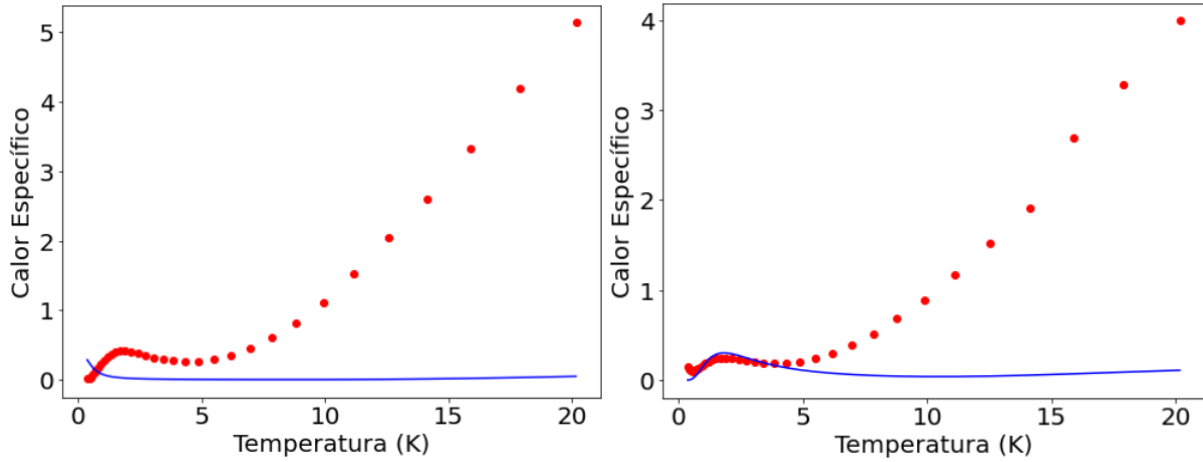


Fig. 12: Medidas de capacidad calorífica de un ion Ni(II)(rojo) y primer ajuste de la red neuronal (azul). La red ha sido entrenada con una contribución de la red y sin ruido aleatorio.

Se observa que los ajustes no son razonables, ya que ambos valores de D deberían estar comprendidos entre -5 y -20 (rango en el que se ha entrenado a la red). Aunque en la segunda medida la predicción es parecida a los datos originales, los resultados no son consistentes y en la primera muestra el ajuste es totalmente distinto a los datos medidos. Por lo que es necesario analizar estas medidas con mayor profundidad modificando el método de entrenamiento.

Un posible factor en esta caracterización errónea puede ser la contribución de la red a altas temperaturas. Si observamos el comportamiento del calor específico a altas temperaturas se puede intuir que la dependencia deja de ser cúbica con la temperatura como predice el modelo de fonones utilizado (2.26), ya que la capacidad calorífica de la red satura en $C = 3k_B$. Esto se puede observar claramente si superponemos las medidas y unos datos sintéticos con un valor de la contribución de la red parecido al observado experimentalmente:

De este modo se puede ver claramente que la dependencia a partir de 10K deja de ser cúbica, y por tanto nuestros datos sintéticos dejarían de ser razonables en este rango de temperaturas. Por ello se plantea no utilizar las medidas correspondientes a temperaturas mayores a 10K , ya que no hemos entrenado a la red para que las interprete correctamente.

Además incluiremos en los datos sintéticos una componente importante de ruido aleatorio, de modo que la red pueda ignorar pequeñas distorsiones en la señal a bajas temperaturas, como una contribución adicional distinta a la de la red o contribuciones nucleares del hamiltoniano (que fueron despreciadas en los desarrollos (2.8) pero pueden aparecer a bajas temperaturas).

Realizando estos cambios generamos unos datos sintéticos de 30 puntos de resolución en el

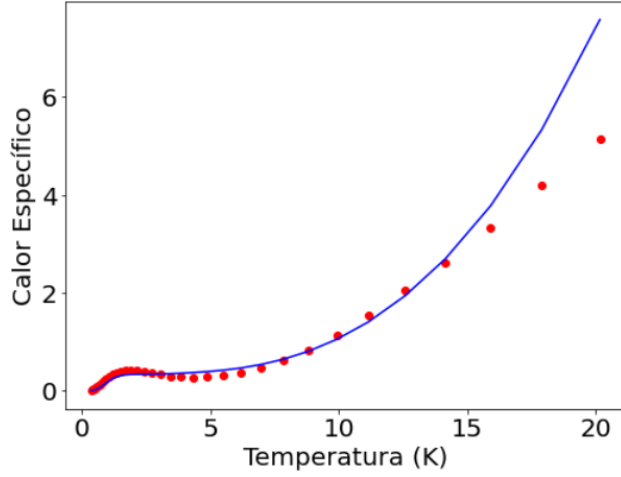


Fig. 13: Medidas de capacidad calorífica de un ion Ni(II)(rojo) y ajuste dado por el paper [7] sumado a un fondo proporcional a T^3 (azul) de amplitud similar al presente en la medida. Se puede observar que a altas temperaturas la forma de la curva azul difiere notablemente de la roja.

rango de temperaturas $T = (0, 10)K$, con una contribución de la red en el rango de valores anterior y un ruido aleatorio con una amplitud $A = (0, 0,04)$. Las nuevas predicciones de la red de las medidas al entrenarla con los nuevos datos sintéticos, representadas en la figura (14), son:

$$\boxed{D_1 = -10,7 \quad E_1 = 1,0 \quad D_2 = -16,4 \quad E_2 = 0,9} \quad (5.3)$$

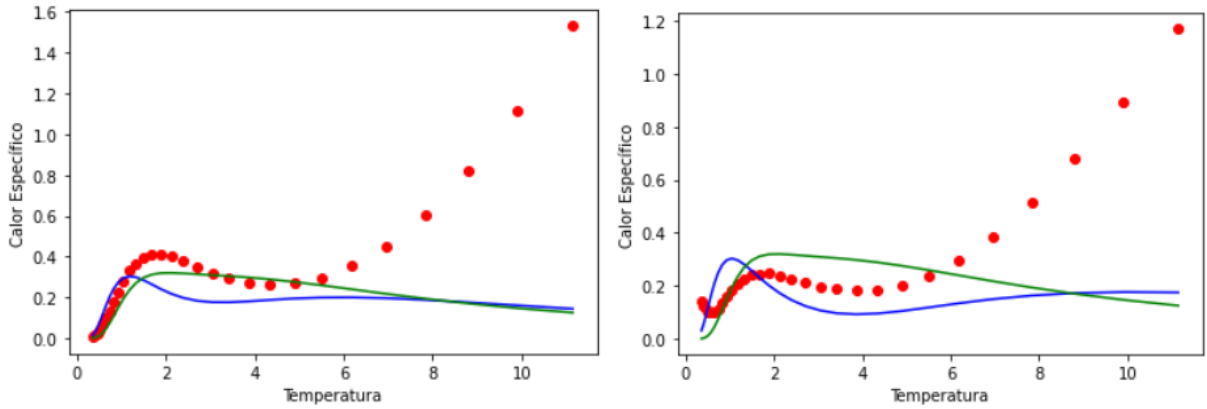


Fig. 14: Medidas de capacidad calorífica de un ion Ni(II)(rojo), segundo ajuste de la red neuronal (azul) y ajuste proporcionado en el paper original (verde). La red ha sido entrenada con una contribución de la red en un rango de temperaturas reducido y con una fuerte componente de ruido aleatorio.

Se puede observar que, aunque las aproximaciones de los parámetros sean más cercanas a los reales que en la primera aproximación, los ajustes no son buenos. El valor de los parámetros sigue siendo muy distinto entre ambas muestras pese a tener ambas parámetros reales muy parecidos y la forma de las curvas predichas sigue siendo considerablemente distinta de las medidas originales. Sin embargo, se puede observar que en el ajuste original (14) tampoco se obtenía un acuerdo perfecto entre los parámetros estimados y la medida de calor específico a campo nulo. Esto probablemente sea debido a la presencia de contribuciones adicionales al hamiltoniano, como se puede observar claramente en el entorno de temperaturas $T < 0,5K$ en

la segunda medida (14), donde se observa un aumento repentino de la capacidad calorífica que no se explica con los parámetros estimados del hamiltoniano de anisotropía.

6. Conclusión

A lo largo de este trabajo hemos estudiado la posibilidad de utilizar redes neuronales como herramienta para caracterizar los parámetros de un hamiltoniano a partir de medidas simples, como puede ser la capacidad calorífica a campo nulo. Se ha observado que estos sistemas son capaces de caracterizar muy bien las señales sintéticas empleadas en su entrenamiento. Esto indica que las redes neuronales son muy efectivas en sistemas perfectamente controlados en los que se conocen todas las posibles contribuciones a la señal.

Sin embargo, en este caso particular hemos observado que en las medidas de la capacidad calorífica suelen aparecer contribuciones que es difícil caracterizar, como contribuciones nucleares o distorsiones provocadas por defectos en los aparatos de medida. Además, la calorimetría a campo cero es muy poco sensible a algunas variaciones particulares de los parámetros D y E , reduciendo enormemente la capacidad de predicción de las redes neuronales.

La inteligencia artificial es una herramienta potente en el análisis de datos que muestra mucho potencial. El análisis realizado en este trabajo es muy rudimentario y tan sólo es una prueba de concepto. Es posible repetir el procedimiento utilizando otros observables más complejos en lugar de la capacidad calorífica, que contengan información acerca de los autoestados del sistema, como puede ser la susceptibilidad magnética, o utilizar distintos campos magnéticos en el entrenamiento. También se podrían utilizar varias medidas distintas de capacidad calorífica en la misma red neuronal, donde las medidas han sido tomadas variando el valor del campo magnético aplicado al sistema, proporcionándole así mucha más información a la red.

7. Referencias

- [1] Fabian Dechert Evert van Nieuwenburg Alex Blania, Sandro Herbig and Florian Marquardt. Deep learning of spatial densities in inhomogeneous correlated quantum systems. *arXiv:2211.09050v1*, 2022.
- [2] A. Abragam y B. Bleaney. *Electron Paramagnetic Resonance of Transition Ions*. 1^a edición, 1970. Editorial: Clarendon Press, Oxford.
- [3] N.W.Ashcroft y N.D.Mermin. *Solid State Physics*. 1^a edición, 1976. Editorial: Brooks/Cole.
- [4] Michael Nielsen. *neural networks and deep learning*. 2019. Accedido al sitio en 2023. <http://neuralnetworksanddeeplearning.com/>.
- [5] IBM. *what is unsupervised learning?* Accedido al sitio en 2023. <https://www.ibm.com/topics/unsupervised-learning>.
- [6] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.
- [7] Marcos Rubín-Osanz, François Lambert, Feng Shao, Eric Rivière, Régis Guillot, Nicolas Suaud, Nathalie Guihéry, David Zueco, Anne-Laure Barra, Talal Mallah, and Fernando Luis. Chemical tuning of spin clock transitions in molecular monomers based on nuclear spin-free ni(ii). *Chem. Sci.*, 12:5123–5133, 2021.