NEURAL QUANTUM STATES: FERMIONS ON D-DIMENSIONS

MASTER THESIS
MASTER IN QUANTUM SCIENCE AND TECHNOLOGY
BARCELONA

MIGUEL CARRASCO ARANGO

SUPERVISED BY ARNAU RIOS HUGUET &
JAVIER ROZALÉN SARMIENTO.

SEPTEMBER 2025

Neural Quantum States: Simulating fermions on D-dimensional systems

Miguel Carrasco Arango

Supervised by: Arnau Rios Huguet, Javier Rozalén Sarmiento

Institute of Cosmos Sciences, University of Barcelona

September 2025

In this work, we explore the use of Neural Quantum States to approximate the ground-state wavefunctions of fully polarized fermionic systems confined in a D-dimensional harmonic trap. Building on the architecture introduced in [1], we generalize the input representation and network structure to handle arbitrary spatial dimensionality, extending the applicability of the method beyond one-dimensional systems. The antisymmetric nature of the fermionic wavefunction is preserved through the use of equivariant neural layers, and a generalized Slater determinant is constructed from learned single-particle orbitals modulated by a Gaussian envelope. Training is carried out in two stages: first, a supervised pretraining phase based on analytical solutions of the non-interacting system, which is then followed by variational Monte Carlo optimization of the network parameters using the energy as the loss function.

We validate our approach on non-interacting systems with up to 4 particles in 2D and 3 particles in 3D, where analytical solutions are available for benchmarking. Results show excellent agreement in terms of mean energy, one-body density, and the one-body density matrix, with observed spatial symmetries and degeneracy patterns matching theoretical expectations. While the training protocol has been generalized to incorporate finite-range interactions, this study focuses on non-interacting systems to establish a solid baseline. The framework developed here provides a flexible and scalable foundation for future exploration of interacting quantum systems in higher dimensions using neural variational methods.

Keywords: Machine Learning, Neural Quantum States, Fermionic Systems, Deep Neural Networks, Variational Monte Carlo

Miguel Carrasco Arango: miguelcarrascoarango@gmail.com

Acknowledgements

I would like to express my gratitude to Dr. Arnau Rios, Javier Rozalén Sarmiento and Dr. Alejandro Romero-Ros for their guidance and supervision throughout this project. Their comments and suggestions were valuable during key moments of the work.

I also wish to thank Dr. James Keeble for making his original implementation available, which served as a starting point and inspiration for the present study.

Contents

1 Introduction

Simulating quantum many-body systems on classical hardware is notoriously demanding, as the computational cost of representing the full wavefunction grows exponentially with system size. Although sophisticated techniques such as tensor–network factorizations can alleviate this scaling for certain classes of states, such as large one-dimensional models with short-range interactions [2, 3, 4, 5, 6, 7] they remain insufficient for generic, strongly entangled systems [8, 9]. This computational bottleneck has motivated the search for alternative methods capable of tackling strongly correlated, higher-dimensional systems or systems defined on unstructured lattices, as often encountered in quantum chemistry and quantum algorithms [10].

Concurrently, modern artificial intelligence has demonstrated an undeniable ability to tackle high-dimensional optimization problems by discovering compact representations of complex data distributions. In particular, deep neural networks can learn intricate correlation structures that are prohibitively expensive to encode explicitly. Leveraging this capability, *Neural Quantum States* (NQS) [11] have emerged as a powerful variational ansatz in which the many-body wavefunction is parametrised by a neural network. NQS frameworks have achieved state-of-the-art accuracy in ground-state energy estimation, all while maintaining favourable scaling on classical hardware accelerators [12, 13, 14, 15, 16, 17, 18].

In this work, we generalize the NQS architecture introduced in Ref. [1]. In it, is provided a minimal implementation for solving the many-body Schrödinger equation in one-dimensional systems of potential relevance to condensed matter physics. That study focused on fully polarized (spinless) fermions confined within a harmonic trap and interacting via finite-range pairwise potentials. Our contribution extends this framework to arbitrary spatial dimensions, thereby broadening its applicability to higher-dimensional and continuous quantum systems. Crucially, our generalization retains the expressive capacity of the original model while enabling the investigation of more complex systems within a unified and scalable framework.

We also present a detailed theoretical formulation of the extended NQS model, describe the variational training procedure employed, and benchmark the resulting simulations against known analytical solutions. Our results demonstrate competitive accuracy and computational scalability, highlighting the potential of NQS as a versatile and complementary tool to both tensor-network techniques and nascent quantum computing platforms. This work underscores the practical utility of neural network-based variational methods for advancing classical simulations of complex quantum phenomena.

This report is structured as follows. In Section 2, we introduce the physical system under study and outline the methodology employed. In Section 3, we describe the structure of the model and detail the generalisations introduced. Section 4 presents and discusses the results of our simulations, including a comparison with the findings reported in the original work. Finally, Section 5 summarises the main conclusions and outlines future directions for research.

The implementation developed in this work is available at
https://github.com/mcarraar39/SpinlessFermionsJames---varias-dimensiones.

2 Theoretical Background

2.1 Describing the system

We consider a D-dimensional system of A identical fermions confined within a harmonic trap, which, for simplicity, is assumed to be rotationally invariant. The particles interact through a finite-range Gaussian potential, and the corresponding Hamiltonian is given by:

$$\hat{H} = -\frac{\hbar}{2m}\sum_{i=1}^{A}\nabla_i^2 + \frac{1}{2}m\omega^2\sum_{i=1}^{A}\sum_{d=1}^{D}(x_i^d)^2 + \frac{V}{\sqrt{2\pi}\sigma}\sum_{i<j}\exp\left[-\frac{\sum_{d=1}^{D}(x_i^d - x_j^d)^2}{2\sigma^2}\right], \quad (1)$$

where m is the mass of the particles, ω is the angular frequency of the harmonic trap, and $\vec{x}_i$ denotes the position vector of particle i, with individual spatial components labelled by superscripts. The Gaussian interaction is characterised by two parameters: the interaction strength V and the range σ. In the limit $\sigma \to 0$, the interaction approaches a contact potential, reducing to $V\delta(\vec{x}_i - \vec{x}_j)$.

By introducing Harmonic Oscillator units, defined as

$$a_{ho} = \sqrt{\hbar/m\omega}, \quad \vec{x} \to \vec{x}/a_{ho}, \quad (2)$$

the Hamiltonian can be expressed in a dimensionless form that is more convenient to handle,

$$\hat{H} = -\frac{1}{2}\sum_{i=1}^{A}\nabla_i^2 + \frac{1}{2}\sum_{i=1}^{A}\sum_{d=1}^{D}(x_i^d)^2 + \frac{V_0}{\sqrt{2\pi}\sigma_0}\sum_{i<j}\exp\left[-\frac{\sum_{d=1}^{D}(x_i^d - x_j^d)^2}{2\sigma_0^2}\right], \quad (3)$$

where the dimensionless parameters are $\sigma_0 = \sigma/a_{ho}$ and $V_0 = V/(a_{ho}\hbar w)$.

Since the particles in the system are fermions, the wavefunction must be antisymmetric. Moreover, as the fermions are fully polarized, all of them have the same spin orientation. Consequently, the wavefunction vanishes whenever two or more particles occupy the same position:

$$\Psi(\vec{x}_1, \ldots, \vec{x}_A) = 0 \quad \text{if} \quad \vec{x}_i = \vec{x}_j, \quad \forall i, j \in \{1, \ldots, A\}, \quad (4)$$

and it changes sign under the exchange of any two particles:

$$\Psi(\vec{x}_1, \ldots, \vec{x}_i, \ldots, \vec{x}_j, \ldots, \vec{x}_A) = -\Psi(\vec{x}_1, \ldots, \vec{x}_j, \ldots, \vec{x}_i, \ldots, \vec{x}_A), \quad \forall i, j \in \{1, \ldots, A\}. \quad (5)$$

In the non-interacting case, $V_0 = 0$, the Hamiltonian describes a system of A non-interacting particles in a D-dimensional Harmonic Oscillator. The solutions can be ob-

tained analytically and they are expressed as products of the one-dimensional harmonic oscillator eigenfunctions (with proper accounting for degeneracies introduced by higher dimensionality),

$$\phi_n(x_i^d) = \left(\frac{1}{\sqrt{\pi}2^n n!}\right)^{1/2} H_n(x_i^d) e^{-\frac{(x_i^d)^2}{2}}, \tag{6}$$

where H_n are the physicist's Hermite polynomials,

$$H_n(x_i^d) = (-1)^n e^{(x_i^d)^2} \frac{d^n}{dx^n} e^{-(x_i^d)^2}. \tag{7}$$

Then, the single-particle solutions in D dimensions take the separable form

$$\phi_{n_i^1,\ldots,n_i^D}(x_i^1,\ldots,x_i^D) = \Pi_{j=1}^D \phi_{n_i^j}(x_i^j), \tag{8}$$

where n_i^j denotes the quantum number associated with the j-th coordinate of particle i. The many-body wavefunction, Ψ, is constructed as a Slater determinant built from these single-particle orbitals, ensuring the required antisymmetry under fermion exchange. Each row of the determinant corresponds to a different particle, while the columns are filled with the lowest-energy orbitals available, respecting the Pauli exclusion principle. Degeneracies arising from different combinations of quantum numbers that yield the same total energy must be carefully accounted for when selecting the occupied orbitals.

2.2 Neural Quantum States and Variational Monte Carlo

The wavefunction of a quantum system fully characterizes the state of a closed system. All physically accessible information is encoded in it and can be extracted through the expectation values of suitable observables. However, in the many-body setting, the dimension of the Hilbert space grows exponentially with the number of particles, rendering direct approaches intractable for all but the smallest systems.

To address this challenge, variational methods are commonly employed. These rely on the variational principle of quantum mechanics, which states that for any normalized trial wavefunction $|\Psi_\theta\rangle$, parametrized by a set of variables θ, the expectation value of the Hamiltonian provides an upper bound to the true ground-state energy of the system,

$$E(\theta) = \langle \hat{H} \rangle = \frac{\langle \Psi_\theta | \hat{H} | \Psi_\theta \rangle}{\langle \Psi_\theta | \Psi_\theta \rangle} \geq E_0, \tag{9}$$

where the brackets can be expressed as integrals in position space,

$$\langle \Psi_\theta | \hat{H} | \Psi_\theta \rangle = \int_{-\infty}^{+\infty} \Psi_\theta^*(\vec{x}_1,\ldots,\vec{x}_A) \hat{H} \Psi_\theta(\vec{x}_1,\ldots,\vec{x}_A) d\vec{x}_1 \ldots d\vec{x}_A, \tag{10}$$

and

$$\langle \Psi_\theta | \Psi_\theta \rangle = \int_{-\infty}^{+\infty} |\Psi_\theta(\vec{x}_1,\ldots,\vec{x}_A)|^2 d\vec{x}_1 \ldots d\vec{x}_A. \tag{11}$$

The goal is then to adjust the parameters θ to minimize $E(\theta)$, yielding an optimal approximation to the ground-state energy and wavefunction.

Within this framework, a variational ansatz is a parametrized family of wavefunctions chosen to approximate the true ground state. Traditional ansätze are typically based on physical intuition and may be limited in expressibility. In contrast, neural networks provide a highly flexible alternative: due to their universal approximation capabilities, they can serve as powerful ansätze capable of capturing complex many-body correlations. When a neural network is used to represent the wavefunction, the resulting object is referred to as a NQS. The network is trained by minimizing a suitable cost function, usually the expectation value of the Hamiltonian, using iterative optimization algorithms. This procedure ensures that the network learns a variational approximation to the ground state of the system [12].

A naive approach would be computing these integrals through some grid-based numerical methods, such as the trapezoidal rule. However, due to the curse of dimensionality, the memory needed to compute these integrals scales exponentially as we increase the dimension of the system and/or the number of particles. To overcome this, we can leverage the Variational Monte Carlo (VMC) method, which enables efficient sampling from the probability distribution defined by the wavefunction [13].

To understand the advantage of this approach, consider the following manipulation. The completeness relation in Hilbert space can be expressed as

$$\mathbf{1} = \int d\mathbf{X} \, |\mathbf{X}\rangle \, \langle \mathbf{X}| \, , \tag{12}$$

where $\mathbf{X}$ includes the quantum numbers of all particles. Inserting it twice in the numerator and once in the denominator of Eq. (9), we can rewrite the expectation value of the Hamiltonian as

$$\langle \hat{H} \rangle = \frac{\langle \Psi_\theta | \, \hat{H} \, | \Psi_\theta \rangle}{\langle \Psi_\theta | \Psi_\theta \rangle} = \frac{\int d\mathbf{X} d\mathbf{X}' \, \langle \Psi_\theta | \mathbf{X} \rangle \, \langle \mathbf{X} | \, \hat{H} \, | \mathbf{X}' \rangle \, \langle \mathbf{X}' | \Psi_\theta \rangle}{\int d\mathbf{X} \, \langle \Psi_\theta | \mathbf{X} \rangle \, \langle \mathbf{X} | \Psi_\theta \rangle} \times \frac{\langle \mathbf{X} | \Psi_\theta \rangle}{\langle \mathbf{X} | \Psi_\theta \rangle}. \tag{13}$$

We can reorder the expression in a more meaningful shape,

$$\langle \hat{H} \rangle = \int d\mathbf{X} \frac{\langle \Psi_\theta | \mathbf{X} \rangle \, \langle \mathbf{X} | \Psi_\theta \rangle}{\int d\mathbf{X} \, \langle \Psi_\theta | \mathbf{X} \rangle \, \langle \mathbf{X} | \Psi_\theta \rangle} \int d\mathbf{X}' \frac{\langle \mathbf{X} | \, \hat{H} \, | \mathbf{X}' \rangle \, \langle \mathbf{X}' | \Psi_\theta \rangle}{\langle \mathbf{X} | \Psi_\theta \rangle}. \tag{14}$$

The first term in the product can be interpreted as the probability density of obtaining the configuration $\mathbf{X}$ given the wavefunction Ψ_θ,

$$p_\theta(\mathbf{X}) = \frac{\langle \Psi_\theta | \mathbf{X} \rangle \, \langle \mathbf{X} | \Psi_\theta \rangle}{\int d\mathbf{X} \, \langle \Psi_\theta | \mathbf{X} \rangle \, \langle \mathbf{X} | \Psi_\theta \rangle}. \tag{15}$$

The second term corresponds to the so-called "local energy" when the system has the quantum numbers encoded in $\mathbf{X}$,

$$\int d\mathbf{X}' \frac{\langle \mathbf{X} | \, \hat{H} \, | \mathbf{X}' \rangle \, \langle \mathbf{X}' | \Psi_\theta \rangle}{\langle \mathbf{X} | \Psi_\theta \rangle} := E^{\mathrm{loc}}(\mathbf{X}, \theta). \tag{16}$$

Since Monte Carlo methods allow us to efficiently sample from the distribution given by the network, we have that

$$\langle \hat{H} \rangle = \int d\mathbf{X} P_\theta(\mathbf{X}) E^{\text{loc}}(\mathbf{X}, \theta) = \langle E^{\text{loc}}(\mathbf{X}, \theta) \rangle_{\mathbf{X}}, \tag{17}$$

where the notation $\langle \cdot \rangle_{\mathbf{X}}$ denotes an average over the sampled configurations.

The use of VMC is essential in this context, as it enables the estimation of expectation values from a tractable number of samples drawn from $|\Psi_\theta|^2$, which is orders of magnitude smaller than the full Hilbert space. Provided the Hamiltonian is local, the local energy E^{loc} remains computationally affordable to evaluate at each sampled configuration. Although this derivation has been presented for the Hamiltonian, the same reasoning applies to any other observable (for more details see [19]).

3 Implementation and extension to arbitrary dimensions

3.1 The NQS architecture

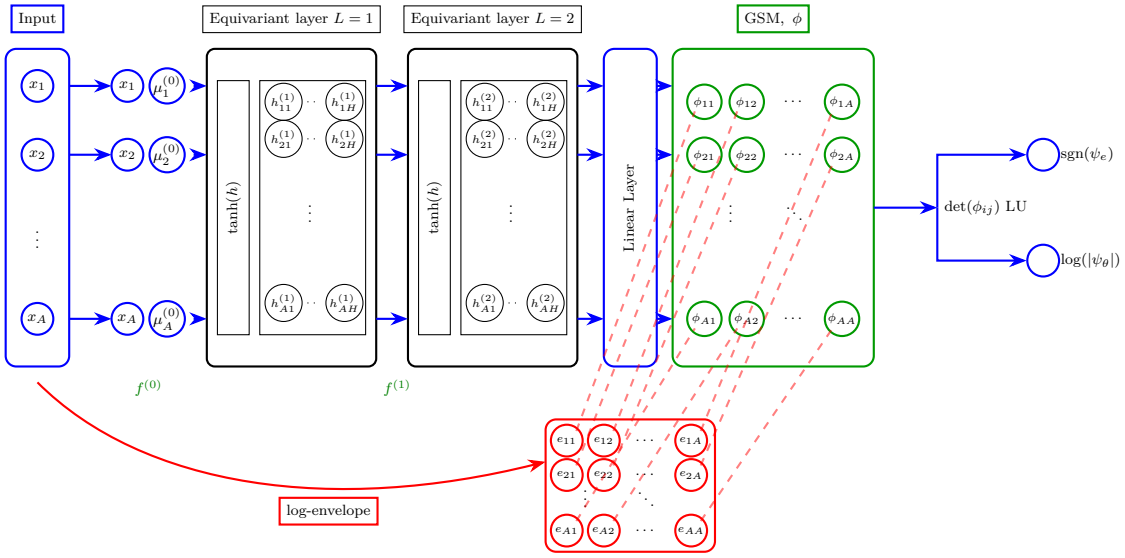

Figure 1: NQS Ansatz of the original work, where A particles were studied using a network with two equivariant layers and a single generalised Slater determinant. The latter outputs the antisymmetric wavefunction of the system, given the generalised single-particle orbitals obtained after applying a log-envelope function to each orbital [1].

An important feature of Neural Quantum States is that this framework enables the explicit encoding of the symmetries of the wavefunction, which facilitates the network's exploration of the parameter space. In the original work, the authors implemented an ansatz inspired by FermiNet [20], represented schematically in Fig. 1. We now describe the architecture of the network, detailing the necessary modifications to generalize it to more than one spatial dimension.

In the original formulation, the input is an A-dimensional vector encoding the positions of the A fermions, where each position is a real, one-dimensional coordinate, $x_i \in \mathbb{R}$ $\forall i \in 1, \ldots, A$. Our first modification is to replace this vector $x \in \mathbb{R}^A$ by a matrix that accounts for the D spatial dimensions, where each column corresponds to one spatial dimension,

$$\begin{pmatrix} x_1 \\ \vdots \\ x_A \end{pmatrix} \in \mathbb{R}^A \quad \longrightarrow \quad \begin{pmatrix} x_1^1 & x_1^2 & \ldots & x_1^D \\ \vdots & \ddots & & \vdots \\ x_A^1 & x_A^2 & \ldots & x_A^D \end{pmatrix} \in \mathbb{R}^{A \times D}. \tag{18}$$

To preserve antisymmetry, the network employs two equivariant layers (corresponding to the black layers in Fig. 1). These layers are designed so that any permutation of the input, x, is reflected in the same permutation of the outputs, h, such that

$$h_i(x_{\pi(1)}, \ldots, x_{\pi(A)}) = h_{\pi(i)}(x_1, \ldots, x_A), \tag{19}$$

where h is the transformation of Eq. (21), and π is a permutation. To achieve this, the methodology of the original work [1] is followed, and a permutation-invariant feature is concatenated with the network input. Thus, the input feature to the first equivariant layer is $f^{(0)} \in \mathbb{R}^{A \times 2}$, constructed as $f^{(0)} = (x \quad \vec{\mu}^{(0)})$, where $x \in \mathbb{R}^{A \times D}$ is the position matrix and $\vec{\mu}^{(0)}$ is a vector of size A with all entries equal to $\mu^{(0)}$. While the computation of this feature changes due to the higher dimensionality, we still preserve the idea of taking a mean over positions across all dimensions,

$$\mu^{(0)} = \frac{1}{A} \sum_{i=1}^{A} x_i \quad \longrightarrow \quad \mu^{(0)} = \frac{1}{A \cdot D} \sum_{i=1}^{A} \sum_{j=1}^{D} (x_i^j). \tag{20}$$

The first equivariant layer then performs a linear transformation followed by a non-linear activation function. Here, we use the hyperbolic tangent activation, as it is both continuous and differentiable. For each row, the computation is

$$h_i^{(1)} = \tanh\left(f_i^{(0)} W^{(1)^T} + b^{(1)} \right), \tag{21}$$

where the weight matrix, which originally had dimensions $H \times 2$, is expanded to $W^{(1)} \in \mathbb{R}^{H \times (D+1)}$ and the bias vector remains $b^{(1)} \in \mathbb{R}^A$. Before feeding the output of this layer into the next equivariant layer, a permutation-invariant feature is needed, so we concatenate the column-wise averages $\mu^{(1)} \in \mathbb{R}^H$ and we obtain $f^{(1)} \in \mathbb{R}^{A \times 2H}$.

The second equivariant layer follows a similar procedure, with the addition of a residual connection, such that the output is

$$h_i^{(2)} = \tanh\left(f_i^{(1)} W^{(2)^T} + b^{(2)} \right) + h_i^{(1)}, \tag{22}$$

where the weight matrix is of the shape $W^{(2)} \in \mathbb{R}^{H \times 2H}$, and the biases are $b^{(2)} \in \mathbb{R}^H$.

The output of the equivariant block is then passed through another linear layer with weights $W^{(M)} \in \mathbb{R}^{A \times H}$ and biases $b^{(M)} \in \mathbb{R}^A$, producing a matrix $M \in \mathbb{R}^{A \times A}$, with rows

computed as

$$M_i = h_i^{(2)} W^{(M)^T} + b^{(M)}. \tag{23}$$

To ensure the correct boundary conditions, we impose a Gaussian envelope on the wavefunction, which enforces decay to zero at infinity. For numerical stability, this envelope is implemented in the log-domain. Due to the updated position structure, we modify this term as

$$e_{ij} = \exp\Big[-(x_i \cdot W_j^{(e)})^2\Big] \quad \longrightarrow \quad e_{ij} = \exp\left[-(\sum_{d=1}^{D} x_i^d \cdot W_j^{d(e)})^2\right], \tag{24}$$

where the weights $W^{(e)} \in \mathbb{R}^{A \times D}$ match the shape of the position matrix. Taking the element-wise product of the output of the linear layer and the Gaussian envelope, we obtain a generalized Slater matrix (GSM), formed by the elements

$$\phi_{ij} = M_{ij} e_{ij}. \tag{25}$$

Each element ϕ_{ij} can be interpreted as a generalized single-particle orbital on state j, constructed to preserve the permutation invariance introduced by the equivariant layers. As discussed in the original article, this construction, where all orbitals depend on all particle positions, enables an efficient encoding of backflow correlations in the wavefunction [21, 20, 22].

Finally, we compute the generalized Slater determinant (GSD) of the GSM, which produces an antisymmetric wavefunction consistent with the description of fully polarized fermions. For numerical stability, the determinant is computed in the log-domain using a LU decomposition. As in the original work, our network can be modified to have a different number of equivariant layers or GSM, but for the systems studied, this combination, with $H = 64$, 2 equivariant layers and one determinant, performed well and a study of how these changes would affect the training is out of the scope of this work.

3.2 Training and sampling

Having defined our variational ansatz, we have to adjust its parameters so that it returns the wavefunction of the ground state of the system. The process is divided into two different stages: a pretraining phase, in which supervised learning is performed, and the proper training phase, which is an unsupervised learning procedure to find the ground state wavefunction.

For the pretraining, the solutions of the non-interacting system, given by Eq. (8), are taken as the target wavefunctions. We compare the generalized single-particle orbitals of the GSM with these analytical solutions. This is done by minimizing the loss function

$$\mathcal{L}^{(0)}(\theta) = \left\langle \sum_{ij} \left[\phi_i(x_j) - \varphi_i(x_j)\right]^2 \right\rangle_{\mathbf{X}}, \tag{26}$$

where, in the original work, φ_i were the i-th eigenstates of the harmonic oscillator. For the

generalized case, we have to increase the number of quantum numbers in each combination. Instead of having one eigenstate per energy level, we work with tuples of D quantum numbers and take into account the energy degeneracy. In cases where different combinations of quantum numbers yield the same energy, the ordering of these tuples will be random.

For the second phase, the variational principle is used, Eq. (9), as it states that the ground-state energy is a lower bound of the energy of the NQS. Therefore, the statistical average of the local energies is minimized as the loss function. This local energy is computed with the kinetic energy evaluated in the log domain for stability reasons, and it has the expression

$$E^{\text{loc}}(\mathbf{X},\theta) = -\frac{1}{2}\sum_{i=1}^{A}\left[\nabla_i^2 \ln|\psi_\theta| + \left(\nabla_i \ln|\psi_\theta|\right)^2\right]\Big|_X + \sum_{i=1}^{A}\sum_{j=1}^{D}\frac{(x_i^j)^2}{2} + \frac{V_0}{\sqrt{2\pi}\sigma_0}\sum_{i<j} e^{-\frac{\sum_{d=1}^{D}(x_i^d - x_j^d)^2}{2\sigma_0^2}}. \tag{27}$$

However, due to numerical instabilities that arise when computing gradients for the kinetic energy, it is better to use an auxiliary function as the loss function, which has the same gradient. The auxiliary expression reads

$$\mathcal{L}^{\text{aux}}(\theta) = 2\left\langle \perp\left[E^{\text{loc}}(\mathbf{X},\theta) - \left\langle E^{\text{loc}}(\mathbf{X},\theta)\right\rangle_{\mathbf{X}}\right] \times \ln|\Psi_\theta(\mathbf{X},\theta)|\right\rangle_{\mathbf{X}}, \tag{28}$$

where $\perp$ denotes the detach function, which acts as the identity but sets the derivative to zero [23].

For both pretraining and training we perform 5000 epochs, each consisting of three phases. First, we perform 10 sweeps of the sampler (it has been adapted to work with D-dimensional positions) that runs the Metropolis-Hastings algorithm [24, 25] over 10096 walkers, so that they are distributed according to the Born probability of the network. Then, the loss function is computed from each position sampled by the walkers, and finally, the Adam optimizer (with learning rate 10^{-4}) is used to update the parameters [26].

During the development, we discovered that for systems with $D > 1$, the wavefunction was unable to properly reproduce the probability distribution outside a neighbourhood of the origin. Initial efforts to address this issue focused on increasing the expressiveness of the network by adding more neurons and verifying the correctness of the generalizations introduced in the sampler and other parts of the code. However, we ultimately identified that the root of the problem laid in the boundary behaviour learned by the network. Specifically, the log-envelope structure, which in principle constrains the decay of the wavefunction, becomes insufficient in higher-dimensional settings. The network discovers that it can minimize the loss function more effectively by suppressing this envelope, reducing its contribution to the generalized single-particle orbitals. As a result, the wavefunction artificially decays too rapidly with distance, vanishing entirely beyond a certain radius.

This behaviour is unphysical. In the systems we aim to simulate, harmonically trapped fermions, the probability density should decay gradually and asymptotically vanish only at infinity. This means that, although the likelihood of finding particles far from the origin decreases, it never becomes exactly zero. Instead, the network learns a degenerate solution in which all regions beyond a certain distance become effectively equiprobable with zero

probability, causing walkers that fall into these regions to drift without guidance rather than being drawn back towards the physically relevant configurations. This undermines the statistical sampling and deteriorates the learned representation of the wavefunction.

To solve this, we decided to enforce the potential of the harmonic trap when computing the probability distribution in the Metropolis-Hastings by adding a Gaussian envelope weighted by a strength constant. Since this modification is applied only when comparing the proposed and existing configurations, it does not affect to the energy or other observables of the system, while it ensures the walkers follow the proper distribution. As in the original work, the Metropolis configuration is adjusted at each step to accept approximately the 55% of the proposed configurations, following the methodology of [27]. Additionally, ℓ_1 norm clipping is used for numerical stability, as in [20]. The "coherent" local energy window is defined by computing the local energy median $\langle E_L \rangle$ and an associated ℓ_1 norm deviation σ_{ℓ_1}. Local energies are only accepted if they fall within the interval $[\langle E_L \rangle - 5\sigma_{\ell_1}, \langle E_L \rangle + 5\sigma_{\ell_1}]$.

3.3 Obtaining the results

3.3.1 Energy

Once the training is completed, we employ a handful of procedures to obtain the final results. We begin by computing a final estimation of the energy using the optimized configuration of the parameters θ. This is done following the methodology described in [28], as in the original work, by employing multiple batches through the "blocking" method (i.e., partitioning the data into smaller blocks called batches to limit comparisons to within each block). We use 10^4 batches, as in the original article, but we increase the number of walkers per batch to 10096 (as in the training phase), resulting in a total of 1.61×10^8 samples that are not affected by the energy clipping applied during training.

The error of the mean energy is computed as

$$\Delta \left\langle \hat{H} \right\rangle_{\mathbf{x}} = \sqrt{\frac{\sigma_B^2}{N_B}}, \tag{29}$$

where N_B is the number of batches used and σ_B^2 is the variance of the batch samples.

3.3.2 One-body density

The next property we analyse is the fermion spatial distribution. Since the fermions are indistinguishable, it is sufficient to consider the marginal distribution of a single particle, which already encodes the behaviour of the ensemble. To this end, we again use 10^4 batches, each employing 10096 walkers to explore the feature space and sample the distribution learned by the network.

Once a sufficiently large number of samples is collected, the one-body density is estimated using a D-dimensional histogram. The spatial domain is discretized into 250^D voxels,

and the density is accumulated cell by cell. Although the final histogram has a fixed size of $\mathcal{O}(n_{\text{bins}}^D)$, its generation is computationally demanding. This is because NumPy needs to build and manipulate intermediate arrays that hold all sampled positions before binning. Each walker contributes A positions in D dimensions, i.e., data of shape $[A, D]$, so the total number of input points is

$$N = B \cdot A \cdot W, \tag{30}$$

where B is the number of batches and W the number of walkers per batch, and the memory required for temporary storage scales as $\mathcal{O}(N \cdot D)$.

As a consequence, with our current computational resources we were only able to obtain the density for up to 4 fermions in a bidimensional system, and only up to 3 fermions in a three-dimensional setup, as 3.01 GiB of RAM were required to store the tensors for the case of a three-dimensional system with 4 fermions.

3.3.3 One-Body Density Matrix

Finally, the One-Body Density Matrix (OBDM) is estimated. It is defined as

$$\rho(\vec{x}, \vec{x}') = A \int d\vec{x}_2 \dots d\vec{x}_A \Psi^*(\vec{x}, \vec{x}_2 \dots, \vec{x}_A)\Psi(\vec{x}', \vec{x}_2, \dots, \vec{x}_A). \tag{31}$$

and it corresponds to the reduced one-body density matrix, obtained by tracing out all but one particle. It can be interpreted as the amplitude of annihilating a particle at position $\vec{x}$ and creating it at $\vec{x}'$. The diagonal elements, $\rho(\vec{x}, \vec{x})$, yield the one-body density, while the off-diagonal terms quantify spatial coherence and correlations.

To estimate it, the Ghost Particle method is used [29]. It works as follows. Once we have the positions of the walkers following the NQS distribution, we perform a random displacement of the first fermion's position, $\vec{x}$, moving it to a new position, $\vec{x}'$. Then, we evaluate the network at both configurations. Since the network returns its output in logarithmic form, we compute the ratio

$$\frac{\Psi(\vec{x}', \dots)}{\Psi(\vec{x}, \dots)} = \text{sgn}_1 \text{sgn}_0 \exp(\log|\Psi_1| - \log|\Psi_0|). \tag{32}$$

Next, we compute some weights defined as $A \cdot (\vec{x}_{\max} - \vec{x}_{\min})^D$, where $\vec{x}_{\min}$ and $\vec{x}_{\max}$ are the spatial bounds in which the OBDM is being calculated. These weights are multiplied by the previously computed ratios. We then compute a $2D$ histogram where the first coordinate corresponds to the original positions, $\vec{x}$, and the second coordinate corresponds to the "ghost" positions $\vec{x}'$. The ratios, multiplied by the weights, are used as the weights of the histogram. Finally, the histogram is normalized so its trace matches the number of particles,

$$\text{Tr}[\rho] = A. \tag{33}$$

In the D-dimensional case, this procedure is generalized by computing a separate histogram for each dimension, as these projections of the OBDM are easier to interpret.

4 Results

4.1 Energy

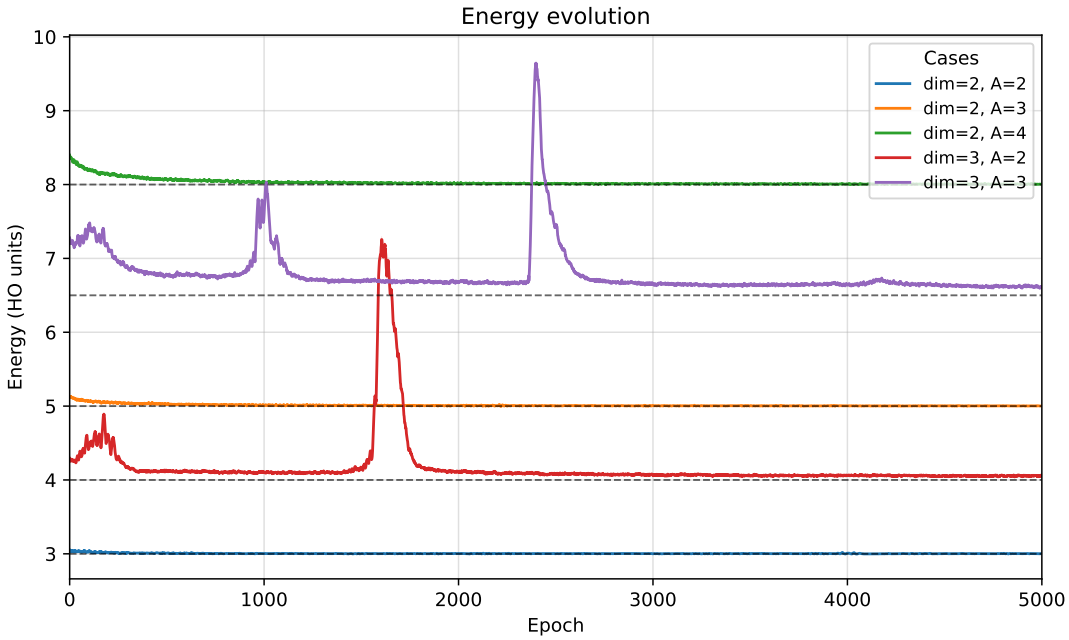

Figure 2: Training evolution of the ground-state energy for systems with $A = 2, 3, 4$ fermions in two dimensions and $A = 2, 3$ fermions in three dimensions. Shaded areas represent the statistical uncertainty, while dashed horizontal lines indicate the exact ground-state energies to which the simulations converge.

In this work, the network has been trained for five different systems, varying the number of fermions and spatial dimensions in order to evaluate its effectiveness. During the generalization of the network, all one-dimensional systems from 2 up to 6 fermions were configured to replicate the results of [1], ensuring that the network behaves as we expected. Therefore, we focus here on the two- and three-dimensional systems without particle interactions, as these allow comparison with analytical results.

As introduced in Section 2.1, the analytical solutions of the non-interacting harmonic trap correspond to products of the one-dimensional harmonic oscillator eigenfunctions. The total energy can thus be computed analytically as

$$E = \sum_{i=1}^{A} \sum_{d=1}^{D} \left(n_i^d + \frac{1}{2} \right). \tag{34}$$

Table 1 shows the mean energies obtained for each case. As expected, we observe that as the system grows in complexity, the network finds it more challenging to reproduce the target state. This is somewhat expected, as the number of parameters or the number of pretraining and training steps are not increased accordingly. Nevertheless, thanks to the variational principle Eq. (9), the true mean energy of the system will always be a lower bound to the mean energy obtained by the NQS, and a sufficiently optimized network

could approximate this bound with high accuracy.

	2 fermions $2D$	3 fermions $2D$	4 fermions $2D$	2 fermions $3D$	3 fermions $3D$
Quantum numbers	(0,0),(0,1)	(0,0),(0,1) (1,0)	(0,0),(0,1) (1,0),(1,1)	(0,0,0),(0,0,1)	(0,0,0),(0,0,1) (0,0,1)
Analytical	3	5	8	4	6.5
NQS	3.0149 ± 0.0008	5.013 ± 0.001	8.026 ± 0.002	4.181 ± 0.005	6.744 ± 0.005

Table 1: Comparison between the ground-state energies obtained by the NQS and the corresponding analytical values for each studied system. The listed quantum numbers correspond to one possible configuration; due to energy degeneracy, alternative equivalent configurations also exist.

This behaviour is illustrated more clearly in Fig. 2, which shows the energy evolution during training for both the two- and three-dimensional systems. For the bidimensional cases with $A = 2, 3, 4$ fermions, the energy exhibits a smooth, monotonic decrease and asymptotically approaches the corresponding analytical values reported in Table 1. In contrast, the three-dimensional systems with $A = 2$ and $A = 3$ fermions display occasional spikes in the energy trajectory. Although these fluctuations are quickly suppressed by the optimization algorithm, they reflect the increased instability and complexity associated with higher-dimensional systems. Nevertheless, in all cases, the energy stabilizes after approximately 5000 training epochs, indicating convergence towards the ground-state configuration.

Despite this convergence, the mean energies obtained do not reach values where the analytical solutions fall within the estimated uncertainties (see Table 1). This discrepancy is most likely due to the limited expressive power of the chosen model in its present configuration, which may not capture the finer details of the system. Alternatively, it could indicate that a more extensive training protocol, either by increasing the number of parameters, pretraining steps, or training epochs, is required to achieve closer agreement with the analytical benchmarks.

4.2 One-body density

4.2.1 $D = 2$ systems

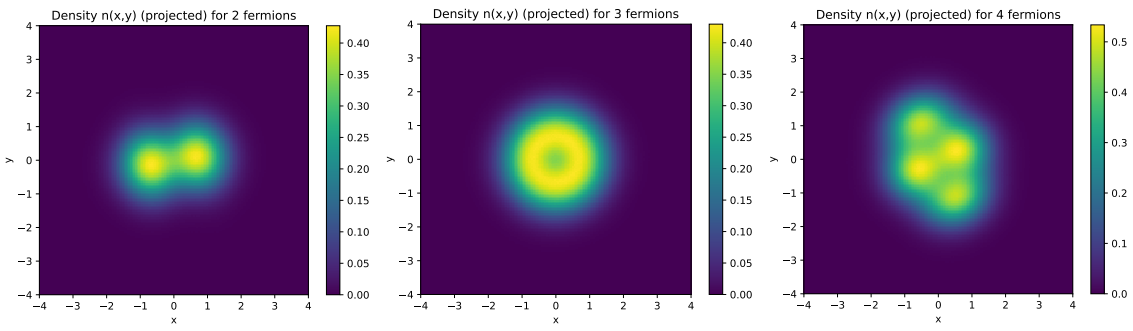

Figure 3: Density $n(x, y)$ of 2 (left panel), 3 (central panel) and 4 (right panel) fermions in $D = 2$ systems.

The next step is to study how the particles are distributed in space. In Fig. 3, we display the probability density of finding a particle, shown as heat maps for the cases of 2, 3 and 4 fermions in $D = 2$ dimensions.

In the cases of 2 and 4 fermions, we observe that the densities are tilted. This occurs because, although the system is equivariant under the SO(2) group, when the energy level is not fully occupied, the system is not fully symmetric and there is a degree of freedom in the rotation angle. This choice in the rotation angle is influenced by the initial weights of the network, which causes a preferred direction to emerge. This direction can be altered by changing the random seed or the initial weights. In contrast, for the system of 3 fermions, the symmetry is preserved, as all the degenerate energetic levels are occupied.

To construct the analytical solutions, the Hermite orbitals used during pretraining are implemented as callable objects. Then, the centres of the grid used to computed the NQS one-body density are taken, and the A Hermite orbitals are evaluated at each point, yielding $\phi_i(\vec{x})$.

The one-body density is computed as

$$n(\vec{x}) = N \int d\vec{x}_2 \dots d\vec{x}_A |\Psi(\vec{x}, \vec{x}_2, \dots, \vec{x}_A)|^2, \tag{35}$$

where the total wavefunction is a Slater determinant,

$$\Psi(\vec{x}_1, \dots, \vec{x}_A) = \frac{1}{\sqrt{A!}} \det \left[\phi_i(\vec{x}_j)\right]_{i,j=1}^{A}. \tag{36}$$

Expanding the squared determinant into a double sum over permutations π and σ yields:

$$|\Psi|^2 = \frac{1}{A!} \sum_{\pi,\sigma} (-1)^{\mathrm{sgn}(\pi)+\mathrm{sgn}(\sigma)} \Pi_{k=1}^{A} \phi^*_{\pi(k)}(\vec{x}_k) \phi_{\sigma(k)}(\vec{x}_k). \tag{37}$$

Due to orthonormality of the orbitals, $\langle \phi_i | \phi_j \rangle = \delta_{ij}$, all cross terms vanish, and only the diagonal terms remain. Thus, the analytical density is computed as

$$n_{\mathrm{exact}}(\vec{x}) = \sum_{i=1}^{A} |\phi_i(\vec{x})|^2. \tag{38}$$

In Fig. 4 we compare the densities computed by the NQS with the analytical solutions. In systems with energy degeneracy, we must select an ordering for the quantum numbers used to build the eigenstates. To avoid introducing any bias, this ordering was chosen randomly. In these cases, the X axis appears to be the preferred direction. Since this choice does not affect the energy or the properties of the system (unless an external field is applied), we perform a rotation of the system to align the preferred directions of both the NQS and the analytical solutions.

To this end, we run an optimization loop that searches for the rotation angle minimizing the absolute difference between the analytical and NQS densities. After applying this optimal rotation to the NQS density, we compute and plot the absolute error. As shown

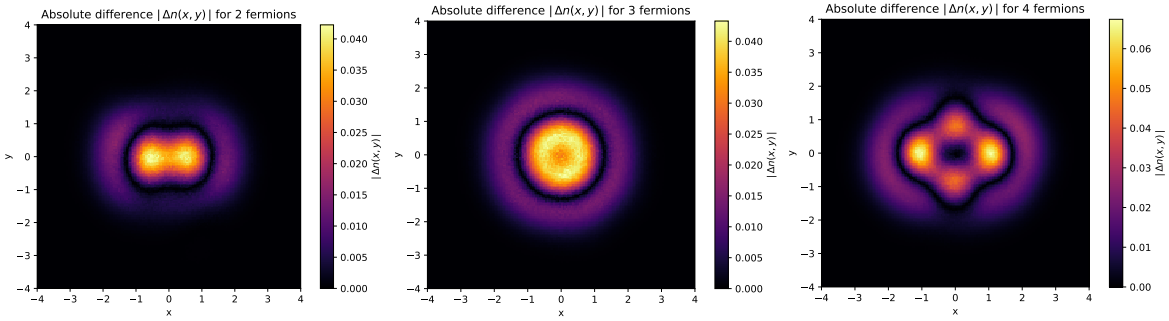

Figure 4: Absolute error in the density of 2 (left panel), 3 (central panel) and 4 (right panel) fermions in $D = 2$ systems.

in Fig. 4, the error maps follow the underlying one-body density profiles, which confirms that the network captures the correct spatial structure. The discrepancies are localized in the regions where the density is nonzero, and the relative error remains below 10% in all cases. This demonstrates that the spatial distribution is accurately reproduced, while the remaining deviations are mainly due to the precise adjustment of the density values rather than to its overall shape.

4.2.2 $D = 3$ systems

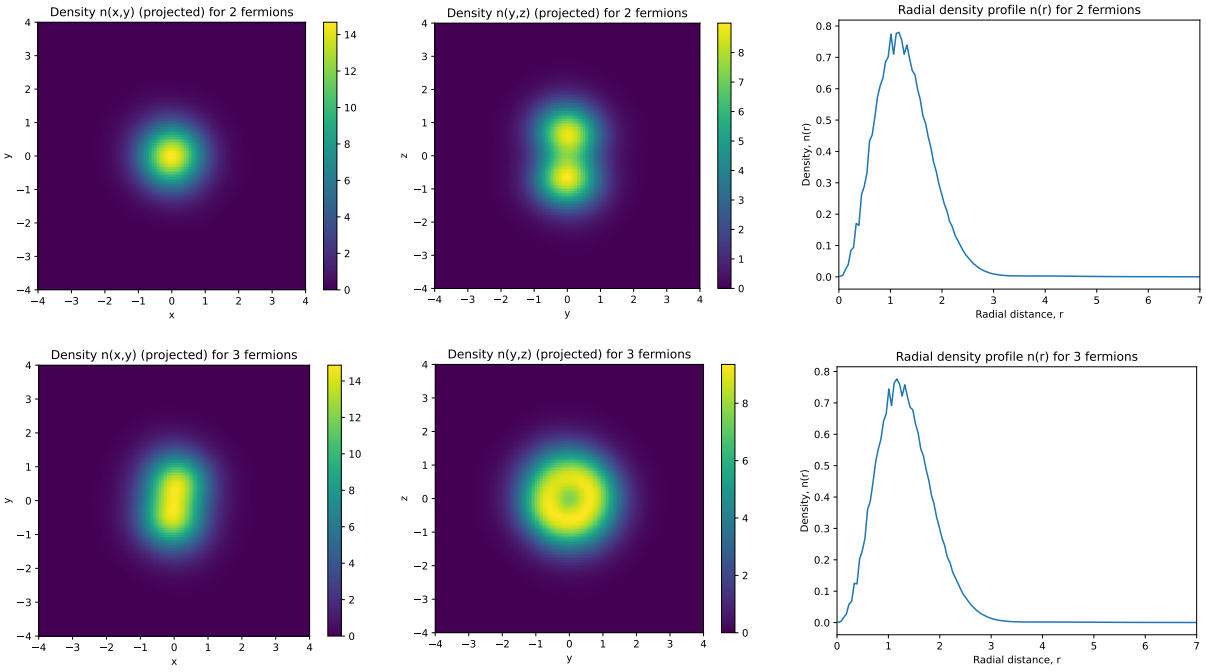

Figure 5: Densities of 2 (upper row) and 3 (bottom row) fermions in $D = 3$ systems.

The analysis of three-dimensional systems is more complex, as the density must be evaluated through lower-dimensional projections. In Fig. 5, we present the projected one-body densities $n(x, y)$ and $n(y, z)$ for systems of 2 and 3 fermions, along with the corresponding radial density profiles. The latter reveal that the average size of the system is approximately 3 harmonic oscillator units in radius for both cases. Notably, the $n(x, y)$ projection for 2

fermions does not exhibit any clear spatial separation between the particles, whereas the $n(y, z)$ projection shows that their separation occurs predominantly along the z-axis in our chosen coordinate system. In the case of 3 fermions, the projected density in the y-z plane resembles the spatial distribution observed in the bidimensional system with 3 fermions, suggesting that the three-dimensional generalization retains key structural features of the lower-dimensional configuration. Overall, the projected densities remain qualitatively consistent with their bidimensional counterparts, providing further validation of the network's behaviour across dimensions.

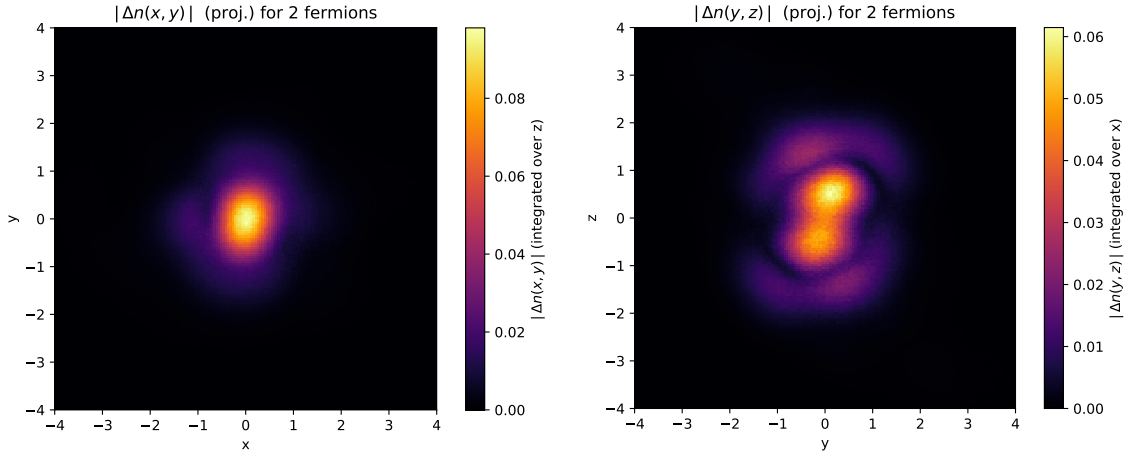

Figure 6: Absolute error in the density of 2 fermions in a $D = 3$ system

In this case, the system is invariant under the SO(3) group, meaning that three degrees of freedom must be considered. To reduce the difference between the theoretical and the obtained densities, we optimize a three-dimensional rotation. This rotation is performed using the Euler angles in the $Z - Y - Z$ convention,

$$R(\alpha, \beta, \gamma) = R_z(\gamma)R_y(\beta)R_z(\alpha). \tag{39}$$

However, performing a full optimization to find the optimal angles is computationally

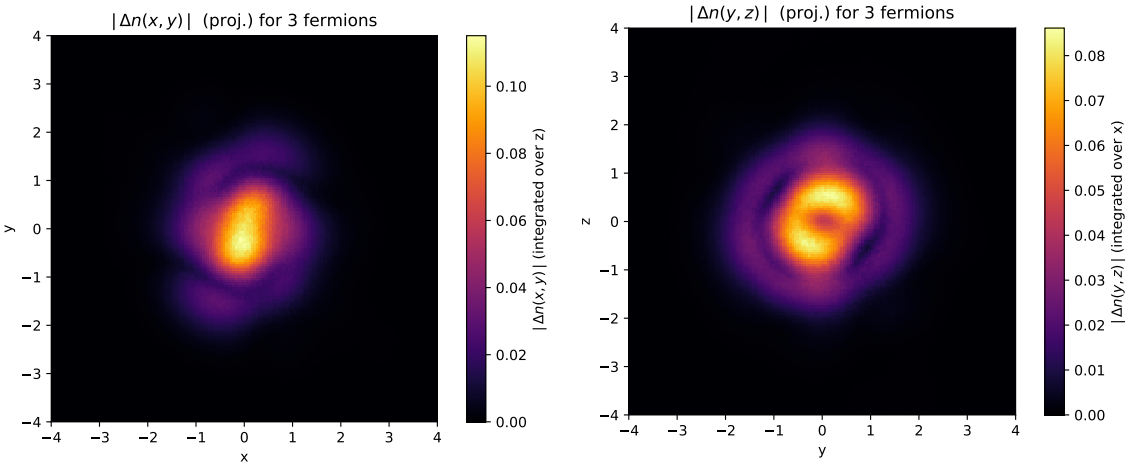

Figure 7: Absolute error in the density of 3 fermions in a $D = 3$ system

costly, so we restrict the search to Euler angles that are multiples of 30°. As shown in Fig. 6 and Fig. 7, the shape of the error plots remains qualitatively consistent with those of the densities, although the discrepancy between these plots increases with the complexity of the system. In these cases the match is not perfect, but the shapes are very similar, so it is likely that further fine-tuning of the rotation angles could reduce the discrepancy.

4.3 One-Body Density Matrix

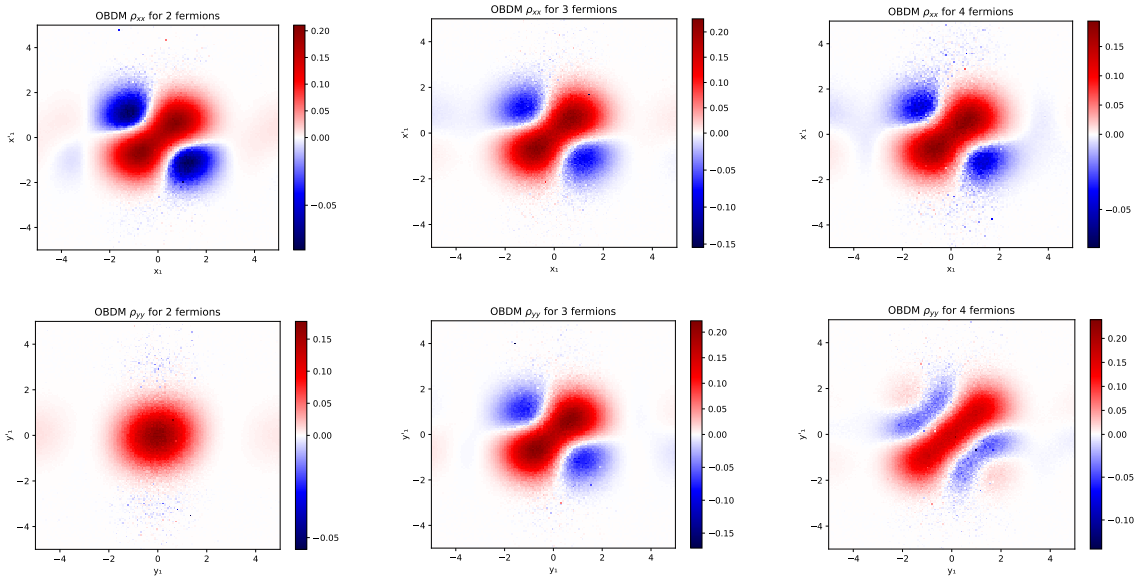

Figure 8: OBDM projections (ρ_{xx}, ρ_{yy}, ρ_{zz}) for 2 (left column), 3 (central column) and 4 (right column) fermions in $D = 2$ systems.

To conclude the analysis, we study the one body density matrix . In multidimensional systems, the OBDM becomes a tensor of rank $2D$ that encodes not only the spatial density, but also coherence and correlation properties of the quantum state. Its eigenvectors represent the natural orbitals of the system, and its eigenvalues provide their respective occupation numbers.

In our case, since the many-body wavefunction factorizes into the products of unidimensional harmonic oscillator eigenstates, it is sufficient to study the coordinate projections of the OBDM. In an ideal scenario, these projections should be diagonalizable, and their eigenvectors should match the $1D$ analytical orbitals.

However, the OBDM is constructed numerically using histograms, and this introduces limitations. The binning resolution is critical for accurate reconstruction, but memory constraints restrict the number of bins that can be used. Although increasing the bin count improves precision, it was not feasible here due to hardware limitations. Even in the unidimensional case, special care was needed to extract eigenvectors with sufficient accuracy.

Fig. 8 displays the projections ρ_{xx} and ρ_{yy} for the $D = 2$ systems with 2, 3 and 4 fermions. We observe that their qualitative features resemble those reported in the

unidimensional benchmarks of [1]. For 2 fermions, ρ_{xx} exhibits a two-lobed structure, as expected from the 1D OBDM for two particles, while ρ_{yy} shows a single, centred cluster. When the number of fermions increases to 3, ρ_{xx} retains the same structure, while ρ_{yy} now acquires the two-lobed shape, suggesting that the additional fermion "fills" a new orbital in the y direction. Finally, for 4 fermions, it is ρ_{yy} that now resembles the OBDM of 3 fermions, while ρ_{xx} remains unchanged.

This observation suggests a sort of directional promotion: when the energy shell is not completely filled and all spatial directions are symmetric, one projection "inherits" the structure of the next higher unidimensional OBDM. When the energy level is completely filled (e.g., 3 fermions in 2D), all directions exhibit the same projection structure.

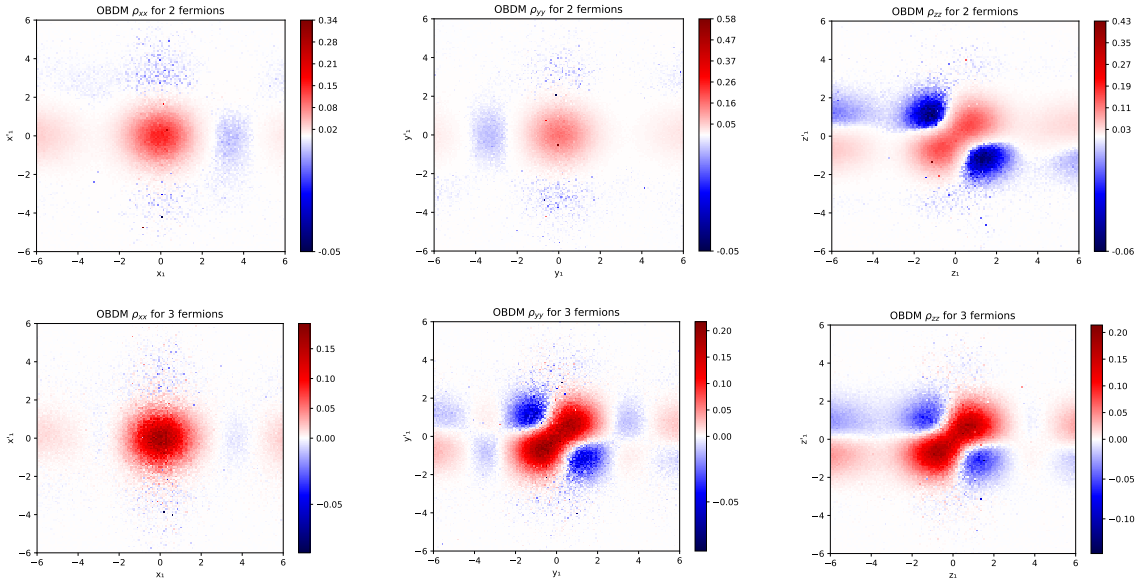

Figure 9: OBDM projections (ρ_{xx}, ρ_{yy}, ρ_{zz}) for 2 (upper row) and 3 (bottom row) fermions in three-dimensional systems.

In Fig. 9, we extend this analysis to the three-dimensional systems, now including the third projection ρ_{zz}. The same pattern of promotion seems to hold: for 2 fermions, one projection (in this case ρ_{zz}) resembles the unidimensional OBDM of 2 fermions, while the others remain simple and centralized. Upon increasing to 3 fermions, another projection (ρ_{yy}) gains structure, also mimicking the unidimensional OBDM of 2 fermions.

However, numerical limitations become more pronounced in 3D. The histogram-based construction of the OBDM introduces noise, especially given the coarse binning resolution used here. This affects the smoothness and eigenstructure of the computed matrices. While some of this noise could be mitigated by increasing the number of bins and improving sampling accuracy, doing so requires more computational memory and time.

A more refined numerical treatment, possibly involving kernel density estimation or variational methods, would likely improve the quality of the OBDM. Nonetheless, such improvements fall beyond the scope of this work.

5 Conclusions

In this work, we have explored the use of machine learning, and in particular, Neural Quantum States (NQS), as a computational tool for the study of quantum many-body systems. Specifically, we have generalized the architecture proposed in [1] to describe fully polarized fermions confined in a D-dimensional harmonic trap. The particles were considered to interact via a Gaussian potential, a form commonly used in nuclear physics to model finite-range interactions. The combination of deep learning techniques with Monte Carlo sampling methods enables us to construct a scalable and flexible framework for variational quantum simulations, capable of learning physically meaningful wavefunctions with minimal prior assumptions beyond the required antisymmetry for fermionic systems. This is made possible by the universal approximation properties of neural networks.

The original NQS ansatz was adapted to account for the spatial dimensionality of the system. Instead of receiving one-dimensional position vectors as input, the generalized network processes $A \times D$ matrices encoding the positions of the A particles in D spatial dimensions. The structure of the subsequent layers is generalized accordingly, while preserving the fundamental features of the original architecture, such as a set of permutation-equivariant layers and a final Generalized Slater Matrix (GSM) determinant calculation, which ensures the correct fermionic antisymmetry. This demonstrates that the original one-dimensional architecture can be extended to higher dimensions without requiring fundamental changes to its conceptual design.

Importantly, the number of parameters in the network was kept constant across all studied systems. While this may result in a loss of precision for more complex configurations, the overall accuracy of the predictions, particularly in terms of mean energy, one-body density, and the one-body density matrix (OBDM), demonstrates the strong expressibility of the ansatz, even without task-specific tuning.

This work has focused on non-interacting systems, which admit analytical solutions to the Schrödinger equation and thus provide a reliable benchmark for validating the variational approach. The results obtained exhibit close agreement with theoretical expectations and confirm the correct generalization of the NQS architecture to arbitrary dimensions. It is worth noting that the training procedure itself has been fully generalized to incorporate interparticle interactions, using a finite-range Gaussian potential. However, no interacting systems were trained in this study, as the focus was placed on validating the method with exactly solvable cases. This ensures that future investigations can directly explore systems with non-zero interaction strength, without requiring further modifications to the architecture or training protocol. Such studies could systematically examine how interactions and dimensionality influence degeneracy lifting, correlation structures, and the overall behaviour of the variational solutions.

Other natural extensions include fine-tuning the architecture for each specific case in order to optimize accuracy and training efficiency. Furthermore, the current setup reveals how the harmonic potential's radial symmetry leads to degenerate energy levels, especially in higher dimensions. This opens the door to exploring symmetry breaking via external

fields and analysing how such perturbations influence the one-body density and OBDM projections. Such studies would be particularly relevant for simulating trapped ultracold atomic gases or nuclear shell models.

Overall, the results presented here illustrate the potential of deep learning–based variational methods to simulate and analyse the quantum behaviour of few-body fermionic systems in arbitrary dimensions. The framework developed provides a robust and flexible foundation for future investigations of more complex, interacting, or experimentally motivated scenarios within the realm of quantum many-body physics.

Bibliography

[1] J. W. T. Keeble, M. Drissi, A. Rojo-Francàs, B. Juliá-Díaz, and A. Rios. Machine learning one-dimensional spinless trapped fermionic systems with neural-network quantum states. *Phys. Rev. A*, 108:063320, 2023. doi:10.1103/physreva.108.063320.

[2] M. C. Banuls. Tensor network algorithms: A route map. *Annual Review of Condensed Matter Physics*, 14:173–191, 2023.

[3] R. Orus. Tensor networks for complex quantum systems. *Nature Reviews Physics*, 1:538–550, 2019. doi:10.1038/s42254-019-0086-7.

[4] U. Schollwöck. The density-matrix renormalization group. *Rev. Mod. Phys.*, 77:259, 2005. doi:10.1103/RevModPhys.77.259.

[5] N. Feldman, A. Kshetrimayum, J. Eisert, and M. Goldstein. Entanglement estimation in tensor network states via sampling. *PRX Quantum*, 3:030312, 2022. doi:10.1103/PRXQuantum.3.030312.

[6] J. Eisert. Entanglement and tensor network states, 2013. URL: https://arxiv.org/abs/1308.3318, arXiv:1308.3318.

[7] A.H. Werner, D. Jaschke, P. Silvi, M. Kliesch, T. Calarco, J. Eisert, and S. Montangero. Positive tensor network approach for simulating open quantum many-body systems. *Phys. Rev. Lett.*, 116:237201, 2016. doi:10.1103/PhysRevLett.116.237201.

[8] F. Verstraete and J.I. Cirac. Renormalization algorithms for quantum-many body systems in two and higher dimensions, 2004. URL: https://arxiv.org/abs/cond-mat/0407066, arXiv:cond-mat/0407066.

[9] M. Lubasch, J.I. Cirac, and M. C. Bañuls. Unifying projected entangled pair state contractions. *New Journal of Physics*, 16:1367–2630, 2014. doi:10.1088/1367-2630/16/3/033014.

[10] L. Gravina, V. Savona, and F. Vicentini. Neural projected quantum dynamics: a systematic study. *Quantum*, 9:1803, 2025. doi:10.22331/q-2025-07-22-1803.

[11] G. Carleo and M. Troyer. Solving the quantum many-body problem with artificial neural networks. *Science*, 355:602,606, 2017. doi:10.1126/science.aag2302.

[12] K. Choo, T. Neupert, and G. Carleo. Two-dimensional frustrated J_1-J_2 model studied with neural network quantum states. *Phys. Rev. B*, 100:125124, 2019. doi:10.1103/PhysRevB.100.125124.

[13] O. Sharir, Y. Levine, N. Wies, G. Carleo, and A. Shashua. Deep autoregressive models for the efficient variational simulation of many-body quantum systems. *Phys. Rev. Lett.*, 124:020503, 2020. doi:10.1103/PhysRevLett.124.020503.

[14] D. Wu, R. Rossi, F. Vicentini, N. Astrakhantsev, et al. Variational benchmarks for quantum many-body problems. *Science*, 386:296–301, 2024. doi:10.1126/science.adg9774.

[15] L.L. Viteritti, R. Rende, and F. Becca. Transformer variational wave functions for frustrated quantum spin systems. *Phys. Rev. Lett.*, 130:236401, 2023. doi:10.1103/PhysRevLett.130.236401.

[16] X. Liang, W. Liu, P. Lin, G. Guo, Y. Zhang, and L. He. Solving frustrated quantum many-particle models with convolutional neural networks. *Phys. Rev. B*, 98:104426, 2018. doi:10.1103/PhysRevB.98.104426.

[17] G. Cassella, H. Sutterud, S. Azadi, N.D. Drummond, D. Pfau, J.S. Spencer, and W.M.C. Foulkes. Discovering quantum phase transitions with fermionic neural networks. *Phys. Rev. Lett.*, 130:036401, 2023. doi:10.1103/PhysRevLett.130.036401.

[18] J. Robledo, G. Carleo, A. Georges, and Stokes J. Fermionic wave functions from neural-network constrained hidden states. *Proceedings of the National Academy of Sciences*, 119:e2122059119, 2022. doi:10.1073/pnas.2122059119.

[19] T. Heightman and M. Płodzień. Deep learning in classical and quantum physics, 2025. URL: https://arxiv.org/abs/2508.10666, arXiv:2508.10666.

[20] D. Pfau, J.S. Spencer, A.G.D.G. Matthews, and W.M.C. Foulkes. Ab initio solution of the many-electron Schrödinger equation with deep neural networks. *Phys. Rev. Res.*, 2:033429, 2020. doi:10.1103/PhysRevResearch.2.033429.

[21] D. Luo and B.K. Clark. Backflow transformations via neural networks for quantum many-body wave functions. *Phys. Rev Lett*, 122:226401, 2019. doi:10.1103/physrevlett.122.226401.

[22] F. Becca and S. Sorella. *Quantum Monte Carlo Approaches for Correlated Systems*. Cambridge University Press, 2017.

[23] S. Zhang, Z. Wan, and H. Yao. Automatic differentiable Monte Carlo: Theory and application. *Phys. Rev. Res.*, 5:033041, 2023. doi:10.1103/physrevresearch.5.033041.

[24] N. Metropolis and S. Ulam. The Monte Carlo method. *Journal of the American Statistical Association*, 44:335–341, 1949. doi:10.1080/01621459.1949.10483310.

[25] W. K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57:97–109, 1970.

[26] D.P. Kingma and J. Ba. Adam: A method for stochastic optimization, 2017. URL: https://arxiv.org/abs/1412.6980, arXiv:1412.6980.

[27] M. Wilson, N. Gao, F. Wudarski, E. Rieffel, and N.M. Tubman. Simulations of state-of-the-art fermionic neural network wave functions with diffusion monte carlo, 2021. URL: https://arxiv.org/abs/2103.12570, arXiv:2103.12570.

[28] A. Scemama, T. Lelièvre, G. Stoltz, E. Cancès, and M. Caffarel. An efficient sampling algorithm for variational Monte Carlo. *The Journal of Chemical Physics*, 125:114105, 2006. doi:10.1063/1.2354490.

[29] W.L. McMillan. Ground state of liquid He^4. *Phys. Rev.*, 138:214507, 1965. doi:10.1103/PhysRev.138.A442.