Neural Network Ansatze for Bosonic and Fermionic Quantum Systems in One Dimension

Hersh Kumar
Department of Physics, University of Maryland, College Park, MD 20742

We use Variational Monte Carlo methods to approximate the ground state energies of many-body quantum systems, using an ansatz with a chosen functional form. However, the accuracy of the approximation is dependent on the choice of functional form, as well as the number and form of variational parameters. We describe neural network ansatze for bosonic and fermionic quantum systems in one dimension, leveraging symmetric functions in order to enforce ground state (anti)symmetries. We apply the neural network ansatze to several bosonic and fermionic systems, and compare our results against the exact solutions. We discuss the generality of the functional form of the neural network wavefunctions, as well as the scalability in the number of variational parameters and particle counts.

I. INTRODUCTION

Quantum many-body systems are of interest across a broad set of subfields in modern physics, but obtaining analytic solutions to many-body models is difficult. For this reason, there are few models for which analytic solutions have been found, and numerical methods have been developed to provide approximate solutions to these complex systems.

One such method is known as variational Monte Carlo, in which a variational ansatz for the wavefunction is chosen, and an estimate for the ground state energy is obtained by the variation of parameters in the ansatz. This method has been applied to a wide range of quantum many-body systems, such as Fermi and Bose gases and quantum chemistry[1–3].

In recent years, machine learning techniques such as neural networks have been applied to solving quantum mechanical problems [4, 5]. When studying the ground states of quantum systems, neural networks provide appealing ansatze for numerical estimations of ground state energies. Minimization of the energy expectation value can be done efficiently using backpropagation and gradient descent algorithms. Neural network techniques have previously been applied to systems such as the Calogero-Sutherland model [6], as well as nuclear systems of few particles [7, 8].

Systems of indistinguishable particles have ground state wavefunctions that are (anti)symmetric under the exchange of particle coordinates. Enforcing these symmetries for neural network ansatze presents a challenge. Work done by Pfau et. al. [9] develops the FermiNet neural network structure to describe systems containing electron-electron and electron-ion interactions in atoms. The FermiNet structure uses permutation-equivariant neural network functions, enforced through neural network layers that are permutation-equivariant. This method produces a neural network ansatz with the antisymmetry necessary to describe fermionic ground state wavefunctions.

In this paper, we propose neural network ansatze for bosonic[1] and fermionic systems, where we select symmetric inputs in order to impose exchange symmetry in the bosonic ansatz, and carefully construct a fermionic ansatz employing the same symmetrization method to produce exchange antisymmetry in the fermionic ansatz. We discuss the benefits of our constructed neural network ansatze, and benchmark them using several bosonic and fermionic systems. We extend our methodology to systems with no known analytic solutions, and we compute the ground state energies of systems with dozens of particles.

In Section II, we outline the four models for which we apply our neural network ansatze. In Section III, we provide an overview of the Variational method, importance sampling, and variational Monte Carlo. We then explicitly construct the bosonic and fermionic neural network functions, discuss the extension of importance sampling to delta function observables, and outline our procedure for determining ground state energies. In Section IV, we discuss our results, we find that the neural network ansatze are able to accurately compute the ground state energy of the many-body systems considered. We conclude with Section V by discussing the extension of the neural network ansatze to higher dimensions.

II. MODELS

We consider several one-dimensional systems in order to leverage our neural network ansatze to approximate the ground state wavefunctions and energies. We consider two bosonic systems, modified forms of the Lieb-Liniger model [11, 12]. The Lieb-Liniger model, a gas of indistinguishable bosons interacting with a contact delta-function potential, has been shown to be analytically solvable in the one-dimensional case [11]. These two models provide rich phase structure, and serve as

[1] The bosonic ansatz, as well as the method for delta function sampling were originally presented in Phys. Rev. C 109, 034004.

testbeds for the comparison of quantitative and qualitative ground state results.

We also consider two fermionic models, starting with a simple system of non-interacting fermions in a harmonic trap, and a system of fermions in a harmonic trap with a delta function contact potential, both of which allow for the comparison of exact and numerical results.

A. Harmonically Trapped Bosons with Contact Potential

The first model that we consider is the Hamiltonian for spinless bosons trapped in a harmonic well, with a contact potential that approximates the behavior of the condensate. In units where $\hbar = 1$, the Hamiltonian is

$$\hat{H} = \sum_{i=1}^{N} \left(-\frac{1}{2m}\frac{\partial^2}{\partial x_i^2} + \frac{1}{2}m\omega^2 x_i^2 \right) + \sum_{i<j}^{N} g\delta\left(x_i - x_j\right) \quad (1)$$

In our implementation, with no loss of generality, we set $m = \omega = 1$. g serves as the interaction strength parameter for the contact potential, and the interaction is repulsive when $g > 0$, and attractive when $g < 0$. While this model is not analytically solvable for a general number of particles when including a harmonic trap[13], the two particle case has been solved exactly[13, 14].

B. Trapped Bosons with Short and Long-Range Interactions

The second bosonic model that we consider is similar to the first, with the addition of a long-range interaction term. The Hamiltonian for this model is

$$\begin{aligned}\hat{H} = &\sum_{i=1}^{N} \left(-\frac{1}{2m}\frac{\partial^2}{\partial x_i^2} + \frac{1}{2}m\omega^2 x_i^2 \right) \\ &+ \sum_{i<j}^{N} \left(g\delta\left(x_i - x_j\right) + \sigma|x_i - x_j| \right)\end{aligned} \quad (2)$$

The strength of this linear interaction is parameterized by σ, and in this one-dimensional case, the interaction corresponds to gravitational attraction in the $\sigma > 0$ regime and Coulomb repulsion in the $\sigma < 0$ regime. This system was found to be exactly solvable in the general case of N bosons [15], and integrable in the case where $\sigma = -m\omega g/2$[16]. In this regime, $g > 0$ indicates that both interaction terms are repulsive, and $g < 0$ indicates that both are attractive. The exact value of the ground state energy was found to be given by

$$E_0 = \frac{N\omega}{2} - mg^2\frac{N\left(N^2-1\right)}{24} \quad (3)$$

with ground state wavefunction

$$\psi_0\left(x_1, \ldots, x_N\right) = \prod_{i<j} e^{-|x_i - x_j|/a_s} \prod_i e^{-x_i^2/\left(2a_{\mathrm{ho}}^2\right)} \quad (4)$$

where $a_{\mathrm{ho}} = \sqrt{\frac{1}{m\omega}}$ denotes the harmonic oscillator length, and $a_s = -\frac{2}{mg}$ is the one-dimensional s-wave scattering length.

C. Harmonically Trapped Fermions

The first fermionic model that we consider is a system of $N = N_\uparrow + N_\downarrow$ fermions, $N_\uparrow$ of which are spin up, and $N_\downarrow$ of which are spin down. The fermions are subject to a harmonic potential, and no other interaction terms.

$$\hat{H} = \sum_{i}^{N} \left(-\frac{1}{2m}\frac{\partial^2}{\partial x_i^2} + \frac{1}{2}m\omega^2 x_i^2 \right) \quad (5)$$

This model is exactly solvable, which allows for a benchmark for the correctness of the fermionic neural network ansatz. Trapped Fermi gases are also of theoretical [17–20] and experimental interest [21, 22].

D. Harmonically Trapped Fermions with Contact Potential

The final model that we consider is a system of $N = N_\uparrow + N_\downarrow$ fermions in a harmonic well, with a delta function contact potential, parameterized by an interaction strength g.

$$\hat{H} = \sum_{i}^{N} \left(-\frac{1}{2m}\frac{\partial^2}{\partial x_i^2} + \frac{1}{2}m\omega^2 x_i^2 \right) + \sum_{i<j}^{N} g\delta\left(x_i - x_j\right) \quad (6)$$

Models of this form are of interest in condensed matter, such as studying the properties of ^{6}Li gases [23, 24]. In particular, the case of a single impurity, where all but one fermions are of the same spin, has been of theoretical [25, 26] and experimental [27] interest.

III. METHODS

A. The Variational Method

Consider a Hamiltonian H with unknown energy eigenstates $|\psi_n\rangle$. Arbitrary states in the Hilbert space can be written as linear superpositions of the energy eigenstates:

$$|\psi\rangle = \sum_n c_n |\psi_n\rangle \quad (7)$$

The expectation value of H for arbitrary state can then be written in terms of the eigenenergies:

$$\langle\psi|H|\psi\rangle = \sum_n |c_n|^2 E_n \quad (8)$$

Now we can apply the fact that $E_0 \leq E_1, E_2, \ldots$, and therefore the expectation value will always be greater than or equal to the ground state energy

$$\langle\psi|H|\psi\rangle \geq E_0 \tag{9}$$

The variational method leverages this property in order to approximate the ground state energy of a system with unknown $|\psi_n\rangle$. By providing an ansatz ψ that depends on a set of variational parameters θ, a gradient descent minimization of $\langle\psi|H|\psi\rangle$ in the space of parameters can provide an estimate for E_0. Note that the variational method does not provide a bound on how accurate the approximation of the ground state energy is, it provides an upper bound on E_0. In many-body systems, the computation of $\langle\psi|\hat{H}|\psi\rangle$ is generally analytically intractable, and instead Monte Carlo methods are used.

B. Importance Sampling

Suppose we wish to compute an expectation value of the form

$$\langle\mathcal{O}\rangle = \frac{\int dx\, f(x)\, \mathcal{O}}{\int dx\, f(x)} \tag{10}$$

Looking at the ratio (where $f(x) \geq 0$)

$$\pi(x) = \frac{f(x)}{\int dx\, f(x)} \tag{11}$$

we see that this can be considered a probability density. Using this interpretation, we can approximate the expectation value of $\mathcal{O}$ as an average over samples $(x_1, \ldots, x_N)$ that follow the distribution $\pi(x)$.

$$\langle\mathcal{O}\rangle \approx \frac{1}{N}\sum_{i=1}^{N} \mathcal{O}_i \tag{12}$$

Where $\mathcal{O}_i$ is $\mathcal{O}$ evaluated at the sample x_i.

1. *Metropolis-Hastings Algorithm*

To generate a set of N samples from the probability distribution $\pi(x)$, we can use the Metropolis-Hastings algorithm, shown in Algorithm 1.

In this pseudocode, v is a constant parameter that is chosen to maintain the ratio between accepted samples and rejected samples at roughly $\frac{1}{2}$. In this implementation, the first $N_{\text{thermalization}}$ samples in the chain are removed. The steps removed for thermalization allow for the chain of samples to be suitably sampled from the intended probability distribution. In order to mitigate autocorrelation between adjacent samples, the algorithm skips N_{skip} points in the chain before selecting another sample.

Algorithm 1: Metropolis-Hastings Algorithm

Input: N_{samples}, $N_{\text{thermalization}}$, N_{skip}, $\pi(x)$, v
Output: N_{samples} samples from the distribution $\pi(\vec{x})$
samples = [ ]
$\vec{x} = \text{uniform}(-1, 1)$
for $i = 0$ *to* $N_{samples} \cdot N_{skip} + N_{thermalization}$ **do**
  $\vec{x}' = \vec{x} + \text{uniform}(-v, v)$
  $\text{prob} = \frac{\pi(\vec{x}')}{\pi(\vec{x})}$
  if $prob \geq uniform(0, 1)$ **then**
    $\vec{x} = \vec{x}'$
  if $i \,\%\, N_{skip} == 0$ *and* $i \geq N_{thermalization}$ **then**
    samples.append $(\vec{x})$
return *samples*

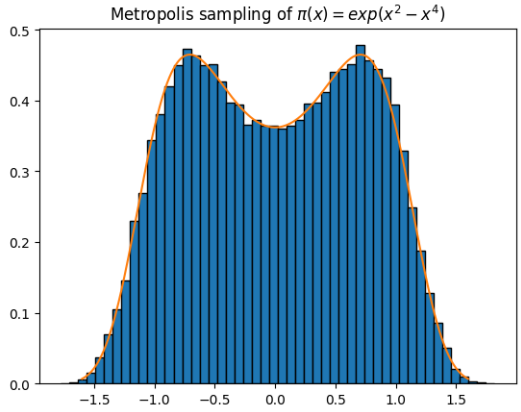

FIG. 1. Shows a histogram generated by the Metropolis-Hastings algorithm with $\pi(x) \sim e^{x^2-x^4}$, using 20×10^4 samples.

This algorithm produces a Markov[2] chain of coordinates that are distributed according to $\pi(x)$. These samples can then be used to compute statistical averages via Equation 12. The Metropolis algorithm can be generalized to higher dimensional probability densities, by changing the individual coordinates to vectors, allowing for the sampling of probability densities with arbitrary numbers of particles. An example of a set of samples following the normalized probability distribution $e^{x^2-x^4}$ is shown in Figure 1.

C. Variational Monte Carlo

For a system with an arbitrary number of particles, we can write the expectation value of the energy via an

[2] Meaning that sample number $i+1$ relies solely on sample number i, not any of the previous samples.

integral of the Hamiltonian and the ansatz

$$\mathcal{E} = \frac{\int d\vec{x}\, \psi^{\dagger}(\vec{x})\, \hat{H}\psi(\vec{x})}{\int d\vec{x}\, \psi^{\dagger}(\vec{x})\, \psi(\vec{x})} \tag{13}$$

where $\vec{x}$ are the particle coordinates. When the number of particles in the system is large, Monte Carlo methods can be used to compute $\mathcal{E}$. To do this, we rewrite the expectation value so as to make it computable via importance sampling.

$$\mathcal{E} = \frac{\int d\vec{x}\, |\psi(\vec{x})|^2 \psi^{-1}(\vec{x})\, \hat{H}\psi(\vec{x})}{\int d\vec{x}\, |\psi(\vec{x})|^2} \tag{14}$$

From this, we see that the expectation value of the energy can be computed via importance sampling, with the probability density function

$$\pi(\vec{x}) = \frac{|\psi(\vec{x})|^2}{\int d\vec{x}\, |\psi(\vec{x})|^2} \tag{15}$$

And observable

$$\mathcal{O} = \frac{1}{\psi(\vec{x})} \hat{H}\psi(\vec{x})$$

In order to perform a gradient descent minimization in the space of variational parameters to estimate the ground state energy, the gradient of the energy with respect to the variational parameters $\frac{\partial \mathcal{E}}{\partial \theta}$ must also be computed. Once again, this is computed via importance sampling

$$\begin{aligned}\left\langle \frac{\partial \mathcal{E}}{\partial \theta} \right\rangle = &\frac{\int d\vec{x}\, \left(\frac{\partial \psi}{\partial \theta} \hat{H}\psi + \psi \hat{H} \frac{\partial \psi}{\partial \theta} \right)}{\int d\vec{x}\, \psi^2} \\ &- 2 \frac{\int d\vec{x}\, \psi \hat{H}\psi \int d\vec{x}\, \psi \frac{\partial \psi}{\partial \theta}}{\left(\int d\vec{x}\, \psi^2 \right)^2}\end{aligned} \tag{16}$$

We can then write these as averages over samples drawn from $\psi(\vec{x})$

$$\left\langle \frac{\partial \mathcal{E}}{\partial \theta} \right\rangle = 2 \left\langle \frac{1}{\psi} \frac{\partial \log(\psi)}{\partial \theta} \hat{H}\psi \right\rangle - 2\mathcal{E} \left\langle \frac{\partial \log(\psi)}{\partial \theta} \right\rangle \tag{17}$$

where we use the fact that $\hat{H}$ is Hermitian.

With the ability to compute these expectation values, we can apply the variational principle in conjunction with a gradient descent minimization in order to approximate the ground state energy, by finding the configuration of the variational parameters that produces the minimum average energy. A broad overview of the Variational Monte Carlo algorithm is shown in Algorithm 2.

The learning rate is a hyperparameter that defines the speed of adjustment of the variational parameters θ. Lower values of the learning rate decrease the change in the variational parameters per iteration of the gradient descent, and higher values of the learning rate provide larger jumps per iteration. The convergence criteria depend on the system being studied, for those with analytic solutions, we can consider the gradient descent minimization to be complete when the average energy is within a predetermined percent difference or threshold away from the analytic solution. For systems that have no analytic solution, we can halt the gradient descent when the magnitude of the components of the gradient are below a chosen threshold value.

Algorithm 2: Variational Monte Carlo

Input: Wavefunction ansatz $\psi(\vec{x}, \theta)$, learning rate
Result: Configuration of θ that produces the minimum $\mathcal{E}$
while *convergence criteria are not met* **do**
  obtain N samples from $\pi(x)$ using Metropolis-Hastings
  Using the samples, compute $\langle \mathcal{E} \rangle$ and $\left\langle \frac{\partial \mathcal{E}}{\partial \theta} \right\rangle$
  Using $\left\langle \frac{\partial \mathcal{E}}{\partial \theta} \right\rangle$, modify θ

D. Neural Network Ansatze

When applying the variational method, the quality of the estimate of the ground state energy depends on both the number of variational parameters used, as well as the functional form of the ansatz. While one can intuit features of the functional form of the ground state wavefunction based on the physical properties of the system in question, neural networks provide a more general tool for searching the space of possible ground state wavefunctions.

1. *Neural Network Function*

We define a neural network with L layers and n_i nodes at layer i, which takes n_1 inputs and produces n_L outputs, as a function of the form

$$\begin{aligned} &I_{i+1} = O_i = f(\mathbf{W}_i \vec{I}_i + \vec{b}_i), \qquad 1 \le i \le L-1 \\ &O_L = \mathbf{W}_L \vec{I}_L + \vec{b}_L \end{aligned} \tag{18}$$

Where $\mathbf{W}_i$ is an $n_{i+1} \times n_i$ matrix of weights, $\vec{b}_i$ is bias vector of n_{i+1} elements, and f is an activation function. The output of the ith layer O_i becomes the input to the $(i+1)$th layer, I_{i+1}. The CELU activation function is used for all but the last layer of the network

$$\mathrm{CELU}(x) = \begin{cases} x & x > 0 \\ e^x - 1 & x \le 0 \end{cases} \tag{19}$$

The last layer uses no activation function, as the neural network should be able to produce outputs across the entire range of real values.

We denote the neural network in our ansatz as $\mathcal{A}(I_1, \ldots, I_N)$. It takes N inputs and produces 1 output value, $n_L = 1$. The variational parameters of the

ansatz are given by the weights and biases of the neural network function, $\vec{\theta} = \{\mathbf{W}_i, \vec{b}_i\}$.

There are several benefits to the use of neural networks as variational ansatze, the first being the scalability in the number of variational parameters. Instead of modifying the functional form of the ansatz to insert additional variational parameters, the structure of the neural network allows for straightforward increases in the number of parameters: by adding more nodes to the network, the sizes of the weights and biases can be increased, increasing the "resolution" of the search of the space of possible ground state wavefunctions. The convergence of the gradient descent to the true functional form of the ground state stems from the fact that feedforward neural networks have been found to be universal function approximators, and can therefore describe arbitrary wavefunctions [28].

2. Bosonic Ansatz

In order to enforce the condition that the wavefunction be symmetric under particle exchange in bosonic systems, we implement a symmetrization function that provides a bijection between the set of particle coordinates, $\{x_i\}$, and a set of coordinates that are fully symmetric under interchange, $\{\xi_i\}$:

$$\xi_i = \sum_{k=1}^{N} (x_k)^i \tag{20}$$

These symmetric coordinates $\xi_1, \xi_2, \ldots, \xi_N$ are then used as the inputs to the neural network, generating a Bose symmetric ansatz. For large values of N, $|\xi_i|$ can grow large, and for this reason we scale the values of $\{x_i\}$ by a width factor w

$$\xi_i' = \sum_{k}^{N} \left(\frac{x_k}{w}\right)^i \tag{21}$$

which is chosen to ensure that $|\xi_i| \sim \mathcal{O}(1)$. This symmetrization transformation keeps all information regarding the original particle coordinates, but eliminates any information about the ordering.

The other constraint imposed on bosonic wavefunctions is that ψ_B must be real-valued and positive. This can be imposed via real-valued weights and biases in the neural network. We write our general bosonic ansatz as

$$\psi_B(x_1, \ldots, x_N) = e^{-\mathcal{A}(\xi_1', \ldots, \xi_N')} \cdot e^{-\Omega \sum_{i=1}^{N} x_i^2} \tag{22}$$

The Gaussian factor helps the ansatz vanishes at at infinity, regardless of the values of the weights and biases of the neural network function. This allows for the random selection of initial weights and biases, which otherwise would provide an ansatz that is numerically unstable. In this case, where the bosons are confined in a harmonic trap with $m = \omega = 1$, we choose $\Omega = \frac{1}{2}$, separating out the non-interacting and interacting components of the ground state wavefunction. With this choice of Ω, the neural network function $\mathcal{A}$ captures the boson interactions.

3. Fermionic Ansatz

Fermionic systems are subject to the constraint of antisymmetry under identical particle exchange

$$\psi_F(\ldots, x_i, \ldots, x_j, \ldots) = -\psi_F(\ldots, x_j, \ldots, x_i, \ldots) \tag{23}$$

which requires a more involved (anti)symmetrization process than the bosonic case. For a system of $N_\uparrow$ spin up fermions, and $N_\downarrow$ spin down fermions, with a total of $N = N_\uparrow + N_\downarrow$ particles, we have particle coordinates denoted from 1 to N:

$$x_1, \ldots, x_{N_\uparrow}, x_{N_{\uparrow+1}}, \ldots, x_N$$

Where $x_1, \ldots, x_{N_\uparrow}$ denote the positions of the up spin fermions, and $x_{N_{\uparrow+1}}, \ldots, x_N$ denote the down spin fermions. For systems of this form, we must enforce that the interchange of particles of identical spin must be antisymmetric. In order to construct such an ansatz, we construct a Slater determinant of functions $\{\phi_i\}$

$$\Phi^{\uparrow}(\vec{x}) = \frac{1}{\sqrt{N!}} \begin{vmatrix} \phi_{1,1} & \phi_{1,2} & \cdots \\ \phi_{2,1} & \phi_{2,2} & \\ \vdots & & \ddots \end{vmatrix} \tag{24}$$

Where $\phi_{i,j}$ denotes a neural network function $\phi_i(x_j, x_1, \ldots x_{j-1}, x_{j+1}, \ldots x_N)$, which is symmetric (using the Newton-Girard identities) with respect to all coordinates other than x_j:

$$\begin{aligned} &\phi_i(x_j, x_1, \ldots, x_k, \ldots x_{j-1}, x_{j+1}, \ldots, x_l, \ldots, x_N) \\ &= \phi_i(x_j, x_1, \ldots, x_l, \ldots x_{j-1}, x_{j+1}, \ldots, x_k, \ldots, x_N) \end{aligned} \tag{25}$$

Similarly, we define a function $\Phi^{\downarrow}$ in terms of the ϕ_{ij} functions

$$\Phi^{\downarrow}(\vec{x}) = \frac{1}{\sqrt{N!}} \begin{vmatrix} \phi_{N_\uparrow+1,N_\uparrow+1} & \phi_{N_\uparrow+1,N_\uparrow+2} & \cdots \\ \phi_{N_\uparrow+2,N_\uparrow+1} & \phi_{N_\uparrow+2,N_\uparrow+2} & \\ \vdots & & \ddots \end{vmatrix} \tag{26}$$

We can then construct the fermionic ansatz, via the product of the two Slater determinants, and a Gaussian to minimize the wavefunction at infinity.

$$\psi_F(\vec{x}) = \Phi^{\uparrow}(\vec{x})\, \Phi^{\downarrow}(\vec{x})\, e^{-\sum_i^N x_i^2} \tag{27}$$

The Slater determinants are antisymmetric with respect to their set of coordinates, that is, $\Phi^{\uparrow}$ is antisymmetric under interchange of the up spin particles, and $\Phi^{\downarrow}$ is antisymmetric under interchange of the down spin particles. The Φ functions have no other constructed symmetries,

and therefore the ansatz is antisymmetric under interchange of fermions of the same spin, as desired.

The variational parameters of the ansatz are the set of weights and biases for the $\{\phi_i\}$ neural networks, which are implemented to be identical in the number of layers and nodes.

E. Delta Function Observables

In several of the models that we consider, particles experience delta function interactions of the form $\delta(x_i - x_j)$, which cannot be computed using the Monte Carlo methods previously presented. Consider a set of coordinates obtained by the Metropolis algorithm, $\{x_1, x_2, \ldots, x_N\}$. If the delta function $\delta(x_i - x_j)$ is then computed using this sample, we see that this term will always be zero, the probability of obtaining samples such that $x_i = x_j$ exactly is vanishingly small. This is a zero-overlap problem, when attempting to compute the energy expectation value, none of the sampled sets of coordinates will provide non-zero contributions to the delta function interaction term. In order to remedy this overlap problem and accurately compute delta function observables, we leverage the properties of bosonic and fermionic wavefunctions under particle exchange. As an example, consider the model discussed in Section II A

$$\hat{H} = \sum_{i=1}^{N} \left(-\frac{1}{2m}\frac{\partial^2}{\partial x_i^2} + \frac{1}{2}m\omega^2 x_i^2 \right) + \sum_{i<j} g\delta(x_i - x_j) \tag{28}$$

where we apply our bosonic ansatz

$$\psi(\vec{x}) = e^{-\mathcal{A}(x_1,\ldots,x_N)} \tag{29}$$

We would like to compute the energy expectation value,

$$\langle E \rangle = \frac{\int dx_1 \ldots dx_N \, \psi \hat{H} \psi}{\int dx_1 \ldots dx_N \, \psi^2} \tag{30}$$

We can split this expectation value into a kinetic term and a potential term

$$\begin{aligned} K &= \frac{\int dx_1 \ldots dx_N \, e^{-2\mathcal{A}(\vec{x})} \sum_i \frac{1}{2}\left(\frac{\partial \mathcal{A}}{\partial x_i^2} - \left(\frac{\partial \mathcal{A}}{\partial x_i} \right)^2 \right)}{\int dx_1 \ldots dx_N \, e^{-2\mathcal{A}(\vec{x})}} \\ V &= \frac{\int dx_1 \ldots dx_N \, e^{-2\mathcal{A}(\vec{x})} \sum_{i<j} g\delta(x_i - x_j)}{\int dx_1 \ldots dx_N \, e^{-2\mathcal{A}(\vec{x})}} \end{aligned} \tag{31}$$

The kinetic term can be sampled using the Metropolis-Hastings algorithm as is,

$$K = \left\langle \sum_{i=1}^{N} \frac{1}{2}\left(\frac{\partial \mathcal{A}}{\partial x_i^2} - \left(\frac{\partial \mathcal{A}}{\partial x_i} \right)^2 \right) \right\rangle_{\pi(\vec{x})} \tag{32}$$

where the subscript $\pi(\vec{x})$ denotes that the average is computed using samples from the distribution

$$\pi(\vec{x}) = \frac{e^{-2\mathcal{A}(\vec{x})}}{\int dx_1 \ldots dx_N \, e^{-2\mathcal{A}(\vec{x})}} \tag{33}$$

However, one can see that the Metropolis-Hastings algorithm will fail to compute V, due to the presence of the delta functions. Instead, we restructure the integral in order to make it possible to apply importance sampling. To begin, we integrate out a coordinate x_2 using the delta functions. This choice of coordinate is arbitrary, due to the bosonic symmetry of the system.

$$\begin{aligned} V &= g\frac{N(N-1)}{2} I \\ I &= \frac{\int dx_1 dx_3 \ldots dx_N \, e^{-2\mathcal{A}(\vec{x}')}}{\int dx_1 dx_2 \ldots dx_N \, e^{-2\mathcal{A}(\vec{x})}} \end{aligned} \tag{34}$$

Where $\vec{x}'$ denotes the set of coordinates where x_2 is replaced with x_1, that is, $(x_1, x_1, x_3, \ldots, x_N)$. The factor $N(N-1)/2$ is obtained through the bosonic symmetry, and eliminates the sum over pairs of particles. We now rewrite the integral I as

$$I = \frac{\int dx_1 \ldots dx_N \, \frac{\exp(-2\mathcal{A}(\vec{x}'))}{\exp(-2\mathcal{A}(\vec{x}))} \mathcal{D}(x_2) \exp(-2\mathcal{A}(\vec{x}))}{\int dx_1 \ldots dx_N \exp(-2\mathcal{A}(\vec{x}))} \tag{35}$$

Where $\mathcal{D}(x_2)$ is any function such that

$$\int_{\mathbb{R}} dx_2 \, \mathcal{D}(x_2) = 1 \tag{36}$$

We can now compute the potential term V via importance sampling, using $\pi(\vec{x})$ as the probability distribution from which we draw samples,

$$V = g\frac{N(N-1)}{2} \left\langle \frac{e^{-2\mathcal{A}(\vec{x}')}}{e^{-2\mathcal{A}(\vec{x})}} \mathcal{D}(x_2) \right\rangle_{\pi(\vec{x})} \tag{37}$$

In practice, we choose $\mathcal{D}(x_2)$ to be the Gaussian

$$\mathcal{D}(x_2) = \frac{1}{\alpha\sqrt{\pi}} e^{-x_2^2/\alpha^2} \tag{38}$$

Where α is a parameter dependent on ψ, which we choose to be

$$\alpha = \sqrt{\frac{\max(|\vec{y}|)^2}{-\log(\sqrt{\pi} \times 10^{-10})}} \tag{39}$$

Where $\vec{y}$ is the set of x_2 coordinates taken from the importance sampling of $\pi(\vec{x})$.

This method of computing delta function observables is also applicable to fermionic models in one dimension, such as the model discussed in Section II D, with the multiplicative factor $N(N-1)/2$ becoming $N_\uparrow N_\downarrow$.

F. Finding Ground State Energies

In order to compute the ground state energy for a certain configuration of a given system, we must specify the structure of the neural network, as well as other related optimization parameters: the number of Monte Carlo samples taken for each computation of the observables, the number of thermalization steps, and the number of skipped steps between each Monte Carlo sample.

There is not a systematic method of choosing the optimal number of layers and nodes for any given problem [29]. We arbitrarily choose to construct our neural network functions with two to four layers, and varied the number of layers in order to obtain the desired precision. We also choose to increase the number of layers and nodes per layer for systems with larger numbers of particles, as we expect the ground state wavefunctions to be more complicated, requiring more complexity in the neural network functions.

In the initial stages of the variational Monte Carlo process, we choose to compute the gradient in parameter space using a relatively low number of Monte Carlo samples, on the order of 500 samples. This allows for faster computations of the gradient, and a faster convergence to the region in parameter space that contains the true minimum. As the optimization converges, we increase the number of Monte Carlo samples per iteration, in order to reduce the stochastic noise. After the energy has sufficiently converged, we compute the final ground state energy using $20,000$ Monte Carlo samples.

The bosonic and fermionic neural network ansatze were implemented in Python, using Google's JAX library. The neural network architecture described in Section III D was implemented using `numpy` matrix operations, and the gradient descent was conducted via the Adam optimizer[30]. We use learning rates between 10^{-2} and 10^{-5}, and decrease the learning rate as the optimizer converges, in order to avoid overshooting the minimum. In order to increase efficiency, we make extensive use of vectorization methods such as `vmap`, as well as JAX's just-in-time compilation through the `@jit` decorator. The vectorization methods provide runtime speedups for computations that involve computing quantities for every sample, such as evaluating the kinetic term of the energy expectation value (32). Rather than sequentially computing the same expression for every sample, `vmap` allows for parallel computation across the samples, providing significant speedups. JAX's just-in-time compilation traces the execution flow of a function the first time it is called, and uses an XLA compiler in order to optimize the computation. Subsequent calls to the function will then leverage the compiled function, providing significant performance improvements.

IV. RESULTS

A. Harmonically Trapped Bosons with Contact Potential

Applying our bosonic ansatz to a system of N bosons in a harmonic trap, interacting with a delta function contact potential, parameterized by interaction strength g. For the case of two bosons, we compare our bosonic ansatz against the analytic expression relating the ground state energy to the interaction strength

$$\sqrt{2}\left(\frac{\Gamma\left(1-\frac{E_0}{2}\right)}{\Gamma\left(\frac{1}{2}-\frac{E_0}{2}\right)}\right)=-\frac{2}{g} \tag{40}$$

Figure 2 compares the exact result for two bosons against the neural network ansatz, for various values of g. We see that the neural network ansatz is able to successfully obtain the correct ground state energies across the range of interaction strengths to within error bars. In order to obtain these ground state energies, a neural network structure of 3 layers of 50 nodes each was used, with a total of $5,301$ variational parameters.

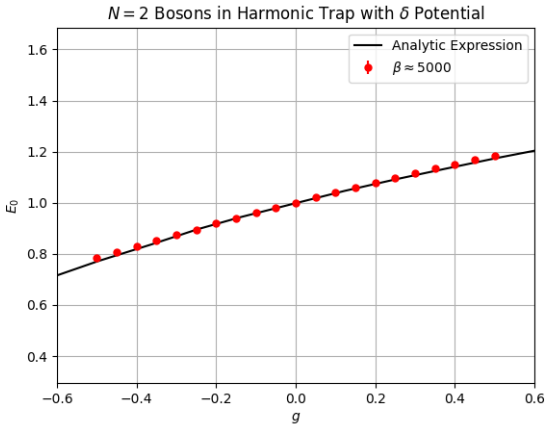

FIG. 2. Shows the ground state energy of the system discussed in Section II A, with $N = 2$, computed by the neural network ansatz in red, along with the exact ground state energy (40) in black, for various values of the contact potential strength. Monte Carlo error bars for the neural network ansatz are too small to be seen. The neural network structure uses roughly 5300 parameters.

B. Trapped Bosons with Short and Long-Range Interactions

Now considering the system discussed in Section II B, N bosons in a harmonic well, with contact interactions (parameterized by g), and long-range interactions (parameterized by σ), we study the exactly solvable region, where $\sigma = -g/2$, and the regime where no analytic solution is known, $\sigma = -g$.

In the exactly solvable regime, we compare our bosonic ansatz against the analytic ground state energy (3). Figure 3 shows the exact ground state energies compared against the neural network ansatz. We see that the ansatz is in agreement with the exact solution across a range of values for g. For values of g less than -0.5, the gradient descent can become sensitive to the minimization hyperparameters, making agreement with the analytic expression more difficult to ascertain.

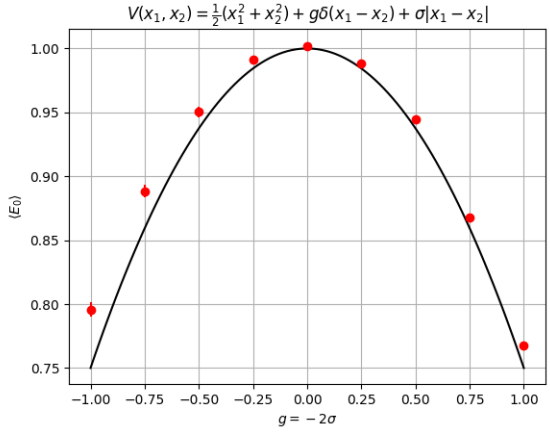

FIG. 3. Shows a comparison of the ground state energies of the system discussed in Section II B with two bosons with interaction strengths given by $\sigma = -g/2$. The black line displays the exact solution for the ground state energy, taken from Beau et. al. [15], and the red points are obtained via the neural network ansatz.

We also apply the ansatz to a system of $N = 50$ bosons in the analytic regime, with both attractive ($g = -0.01$) and repulsive ($g = 0.05$) potentials. Figure IV B compares the local density profiles $n(x) = \int dx_2 \dots dx_N |\psi(x_1, x_2, \dots, x_N)|^2$ of the analytic ground states to those obtained through the neural network ansatz. We see that the neural network density profiles closely match the analytic expressions for both potentials.

We also consider the regime that has no known analytic solution, particularly the case where $g = -\sigma$. In order to determine whether the neural network qualitivatively encodes the correct information about the ground state wavefunction, we plot the local density profile. We compute the profile using samples drawn from the normalized probability distribution $\pi(x) \sim \psi(\vec{x})^2$, and histogramming the x_1 coordinates. Figure 5 shows the local density profile for a system of $N = 3$ bosons with a repulsive interaction, where we can see that the neural network ansatz is able to encode the ground state locations of the bosons.

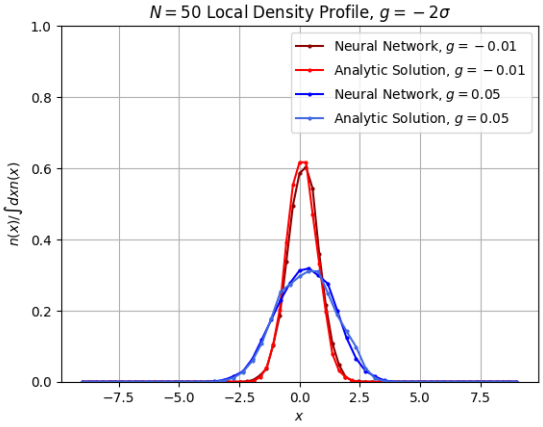

FIG. 4. Shows the local density profiles for $N = 50$ bosons with short and long range interactions in the analytically solvable regime, for a repulsive and an attractive potential. The attractive potential (red) displays the bosons clumping together, while the repulsive potential (blue) has bosons spreading out.

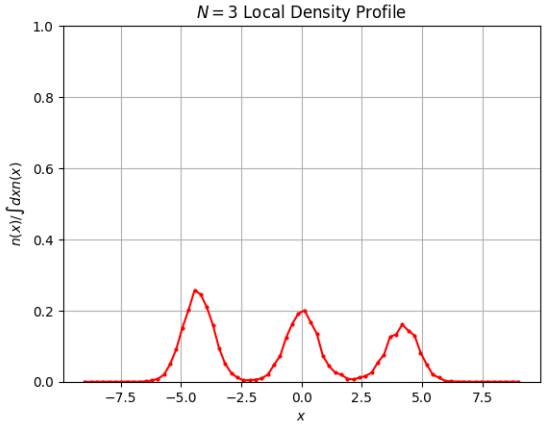

FIG. 5. Shows the local density profile $n(x)$ of the ground state wavefunction for a system of three bosons with repulsive interactions with strengths $g = -\sigma = 2$.

C. Harmonically Trapped Fermions

For the system discussed in Section II C, $N = N_\uparrow + N_\downarrow$ fermions trapped in a harmonic well, we compare our neural network fermionic ansatz against the analytic solution for the ground state energy in the case where $m = \omega = 1$,

$$E_{\text{gs}} = \frac{N_\uparrow^2 + N_\downarrow^2}{2} \tag{41}$$

Figure 6 shows a comparison between the analytic solution for the ground state energy and the fermionic ansatz for low particle counts, $N_\uparrow = N_\downarrow = \frac{N}{2}$. We find that the neural network ansatz is in agreement with the analytic expression to within Monte Carlo error bars. The mean

values of the energy are also found to be within one percent of the exact value.

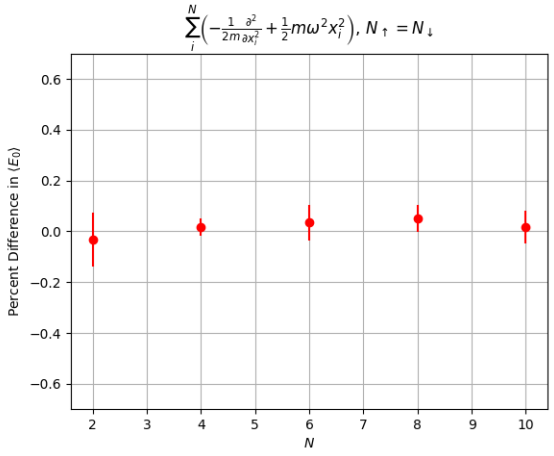

FIG. 6. Shows the percent difference between the ground state energy computed by the neural network ansatz and the analytic expression (41) for the system described in Section II C. Fermions are equally spin up and spin down, $N_\uparrow = N_\downarrow = \frac{N}{2}$.

D. Harmonically Trapped Fermions with Contact Potential

Finally, we apply our fermionic ansatz to the system of harmonically trapped fermions with a contact potential, discussed in Section II D. We compare our neural network ansatz against the ground state energies discussed by Astrakharchik et. al., for the case where $N_\downarrow = 1$ and $N_\uparrow$ ranges from 1 to 10 [31]. We find that the neural network ansatz agrees with the diffusion Monte Carlo results presented in Astrakharchik et. al [31]. A plot of the percent difference from the exact values is shown in Figure 7.

Figure 8 shows the ground state wavefunction of the $N_\uparrow = 1$ and $N_\downarrow = 1$ case, for both the $g > 0$ and $g < 0$ cases. We find that the ground state wavefunction found by the neural network ansatz produces the correct qualitative behavior, with the attractive case where $g = -1.5$ showing the fermions in the same region, and the repulsive case where $g = 3$ showing the fermions separating.

V. CONCLUSIONS

The quantum mechanical variational method allows for the approximation of ground state energies of quantum systems given an ansatz for the functional form of the wavefunction. For systems with arbitrary numbers of particles, expectation values of observables can be intractable analytically, and therefore must be computed via numerical methods. We discuss the traditional method of variational Monte Carlo, using the Metropolis-Hastings algorithm to sample from arbitrary probability

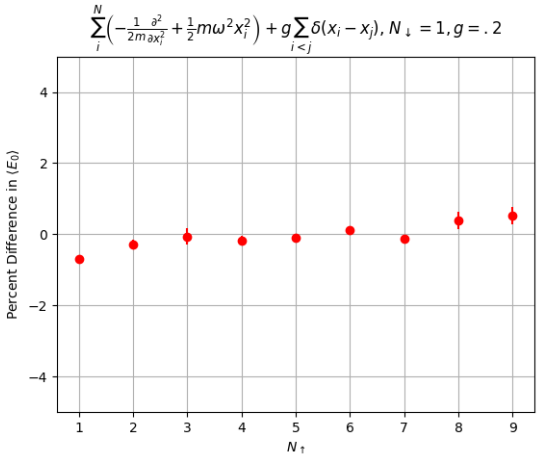

FIG. 7. Shows the percent difference between the ground state energy computed by the neural network ansatz and Astrakharchik et. al. [31] for the system discussed in Section II D. The system contains 1 spin down fermion, and N spin up fermions. The delta function interaction strength is set to $g = .2$.

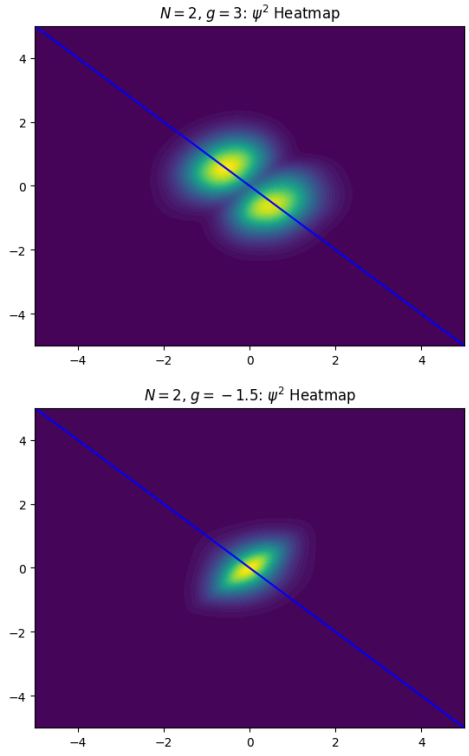

FIG. 8. Shows the ground state wavefunction for a spin up fermion and a spin down fermion interacting according to the model discussed in Section II D. We plot the probability density in position space, where brighter points denote higher probabilities (larger values of $\psi\left(x_\uparrow, x_\downarrow\right)^2$) of the fermions being there. The top plot displays the case where $g = 3$ (repulsive potential), and the bottom plot displays the case where $g = -1.5$ (attractive potential).

densities, which are then used to compute statistical averages for the energy of the system. We then briefly note the drawbacks of the traditional variational Monte Carlo approach, namely the dependence on the choice of ansatz.

We then discuss the applications of neural network functions as general wavefunction ansatze to bosonic and fermionic systems, across a range of particle counts. The neural network ansatze provide several important advantages, such as being able to describe general configurations in the space of possible wavefunctions, as well as providing ease of scalability, without the need to change the functional form of the ansatz, the number of variational parameters can be meaningfully increased. The neural network ansatze also allow for straightforward scaling of particle counts, codes for computing the ground state energy of a system with 2 particles can easily be adapted to a system of 50 particles by changing the structure of the neural network. Due to these benefits, these ansatze are able to more easily probe systems for which there is no known analytic solution.

Wavefunctions of models that describe bosons and fermions must satisfy Bose and Fermi exchange symmetries respectively. We demonstrate methods to enforce such symmetries in neural network functions in one dimension, using a set of bijective coordinates in the case of Bose symmetries, and via Slater determinants with careful coordinate symmetrization in the case of Fermi symmetries. We also discuss a method of computing delta function observables in one dimension.

The method of enforcing Bose symmetries in neural network functions can be generalized to higher dimensional systems via the use of a bijection between the set of coordinates $\{\vec{r}_i\}$, $i = 1, \dots N$, and symmetric functions $\{\xi_i\left(r_1^x, r_1^y, r_1^z, \dots, r_N^x, r_N^y, r_N^z\right)\}$, $i = 1, \dots M$, where M can be larger than $3N$. These functions must have certain exchange symmetries, they must be symmetric under the exchange of entire particle vectors, $\vec{r}_n \leftrightarrow \vec{r}_m$, but not under the exchange of individual components, $r_n^x \leftrightarrow r_m^x$. From the set of functions $\{\xi_i\}$, we can reconstruct the individual sets of particle coordinates. In order to extend the Fermi (anti)symmetry, we can apply the symmetric functions in a similar manner as in the one dimensional case, in order to enforce the exchange antisymmetry in higher dimensions.

ACKNOWLEDGEMENTS

First and foremost, I would like to thank Professor Paulo Bedaque for advising me in this research, as well as for all the guidance he has provided during my time at the University of Maryland. I would also like to thank Dr. Andy Sheng for his collaboration, guidance, and debugging prowess. I would also like to thank Isabelle Alvi for her unwavering love and support, I truly could not have achieved this without her. I am also indebted to my brother for his advice, and for inspiring me to pursue physics. Finally, I would like to thank my parents for their constant support, for which I am truly grateful.

[1] M. Bajdich and L. Mitas, Electronic structure quantum monte carlo, arXiv preprint arXiv:1008.2369 (2010).
[2] F. R. Petruzielo, J. Toulouse, and C. Umrigar, Approaching chemical accuracy with quantum monte carlo, The Journal of chemical physics **136** (2012).
[3] J. E. Deustua, I. Magoulas, J. Shen, and P. Piecuch, Communication: Approaching exact quantum chemistry by cluster analysis of full configuration interaction quantum monte carlo wave functions, The Journal of Chemical Physics **149** (2018).
[4] G. Carleo, I. Cirac, K. Cranmer, L. Daudet, M. Schuld, N. Tishby, L. Vogt-Maranto, and L. Zdeborová, Machine learning and the physical sciences, Rev. Mod. Phys. **91**, 045002 (2019).
[5] S. Das Sarma, D.-L. Deng, and L.-M. Duan, Machine learning meets quantum physics, Physics Today **72**, 48 (2019), https://pubs.aip.org/physicstoday/article-pdf/72/3/48/10121955/48_1_online.pdf.
[6] H. Saito, Method to solve quantum few-body problems with artificial neural networks, Journal of the Physical Society of Japan **87**, 074002 (2018).
[7] C. Adams, G. Carleo, A. Lovato, and N. Rocco, Variational monte carlo calculations of $a \leq 4$ nuclei with an artificial neural-network correlator ansatz, Physical Review Letters **127**, 022502 (2021).
[8] A. Gnech, C. Adams, N. Brawand, G. Carleo, A. Lovato, and N. Rocco, Nuclei with up to $a = 6$ nucleons with artificial neural network wave functions, Few-Body Systems **63**, 7 (2022).
[9] D. Pfau, J. S. Spencer, A. G. Matthews, and W. M. C. Foulkes, Ab initio solution of the many-electron schrödinger equation with deep neural networks, Physical Review Research **2**, 033429 (2020).
[10] P. F. Bedaque, H. Kumar, and A. Sheng, Neural network solutions of bosonic quantum systems in one dimension, Phys. Rev. C **109**, 034004 (2024).
[11] E. H. Lieb and W. Liniger, Exact analysis of an interacting bose gas. i. the general solution and the ground state, Physical Review **130**, 1605 (1963).
[12] E. H. Lieb, Exact analysis of an interacting bose gas. ii. the excitation spectrum, Physical Review **130**, 1616 (1963).
[13] B. Wilson, A. Foerster, C. Kuhn, I. Roditi, and D. Rubeni, A geometric wave function for a few interacting bosons in a harmonic trap, Physics Letters A **378**, 1065 (2014).
[14] T. Busch, B.-G. Englert, K. Rzażewski, and M. Wilkens, Two cold atoms in a harmonic trap, Foundations of Physics **28**, 549 (1998).
[15] M. Beau, S. Pittman, G. Astrakharchik, and A. Del Campo, Exactly solvable system of one-dimensional trapped bosons with short-and long-range interactions, Physical review letters **125**, 220602 (2020).

[16] J. Yang and A. Del Campo, One-dimensional quantum systems with ground state of jastrow form are integrable, Physical Review Letters **129**, 150601 (2022).
[17] D. Butts and D. Rokhsar, Trapped fermi gases, Physical Review A **55**, 4346 (1997).
[18] S. E. Gharashi and D. Blume, Correlations of the upper branch of 1d harmonically trapped two-component fermi gases, Physical review letters **111**, 045302 (2013).
[19] B. Mukherjee, Z. Yan, P. B. Patel, Z. Hadzibabic, T. Yefsah, J. Struck, and M. W. Zwierlein, Homogeneous atomic fermi gases, Physical review letters **118**, 123401 (2017).
[20] D. J. Toms, Ideal fermi gases in harmonic oscillator potential traps, Annals of Physics **320**, 487 (2005).
[21] B. DeMarco and D. S. Jin, Onset of fermi degeneracy in a trapped atomic gas, Science **285**, 1703 (1999), https://www.science.org/doi/pdf/10.1126/science.285.5434.1703.
[22] S. Giorgini, L. P. Pitaevskii, and S. Stringari, Theory of ultracold atomic fermi gases, Reviews of Modern Physics **80**, 1215 (2008).
[23] G. Bruun and K. Burnett, Interacting fermi gas in a harmonic trap, Physical Review A **58**, 2427 (1998).
[24] J. Levinsen, P. Massignan, G. M. Bruun, and M. M. Parish, Strong-coupling ansatz for the one-dimensional fermi gas in a harmonic potential, Science Advances **1**, e1500197 (2015).
[25] C. J. M. Mathy, M. B. Zvonarev, and E. Demler, Quantum flutter of supersonic particles in one-dimensional quantum liquids, Nature Physics **8**, 881 (2012).
[26] Z. Lan and C. Lobo, A single impurity in an ideal atomic fermi gas: current understanding and some open problems, arXiv preprint arXiv:1404.3220 (2014).
[27] J. Catani, G. Lamporesi, D. Naik, M. Gring, M. Inguscio, F. Minardi, A. Kantian, and T. Giamarchi, Quantum dynamics of impurities in a one-dimensional bose gas, Physical Review A **85**, 023623 (2012).
[28] K. Hornik, M. Stinchcombe, and H. White, Multilayer feedforward networks are universal approximators, Neural networks **2**, 359 (1989).
[29] D. Stathakis, How many hidden layers and nodes?, International Journal of Remote Sensing **30**, 2133 (2009).
[30] D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980 (2014).
[31] G. E. Astrakharchik and I. Brouzos, Trapped one-dimensional ideal fermi gas with a single impurity, Phys. Rev. A **88**, 021602 (2013).