

Machine learning approach to trapped many-fermion systems

Paulo F. Bedaque,^{1,*} Hersh Kumar,^{1,†} and Andy Sheng^{1,‡}

¹*Department of Physics, University of Maryland, College Park, MD 20742*

(Dated: August 19, 2024)

We do many important things.

I. INTRODUCTION

- Develop general neural net ansatz for systems of fermions with 2 species in 1D
- Apply ansatz to systems of harmonically trapped fermions with δ function interactions
- Have results for 5+5 with strongly repulsive interactions
- Have results for 4+4, 4+5, 5+5 with attractive interactions
- Andy's results for the impurity cases
- Transfer learning to increase the coupling
- Parameter extrapolation to make guess of starting parameters

II. METHODS

Given the Hamiltonian of a system of $N = N_{\uparrow} + N_{\downarrow}$ fermions, we would like to solve the time-independent Schrödinger equation for the ground state wavefunction and energy. To do so, we utilize the variational method; any arbitrary wavefunction ψ must have an expectation value of the energy that is greater than or equal to the ground state energy. In general, the wavefunction is given by a chosen $\psi(\vec{x}, \theta)$, where \vec{x} are the positions of the fermions and θ are variational parameters. Typically, the energy $\langle E(\theta) \rangle$ is then minimized with respect to the variational parameters θ to obtain an estimate for the ground state of the system.

A. Neural Networks

The quality of the estimate provided by the variational method is dependent on the choice of ansatz, in particular the functional form of the ansatz as well as the number of parameters used. Good guesses for the functional form of the ground state wavefunction can be obtained via expectations or knowledge of the system, but the application of neural networks allows for a more general search through the space of wavefunctions.

Talk
about
the
ben-
efits
of NN
VMC

B. Fermionic Ansatz

An ansatz for the ground state wavefunction for a system of N_{\uparrow} spin up fermions and N_{\downarrow} spin down fermions should enforce antisymmetry under the exchange of fermions of the same spin. For a system of $N = N_{\uparrow} + N_{\downarrow}$ fermions, we have particle coordinates from 1 to N :

$$x_1, \dots, x_{N_{\uparrow}}, x_{N_{\uparrow}+1}, \dots, x_N \quad (1)$$

* bedaque@umd.edu

† hekumar@umd.edu

‡ asheng@umd.edu

Where $x_1, \dots, x_{N_\uparrow}$ denote the positions of the spin up fermions, and $x_{N_\uparrow+1}, \dots, x_N$ denote the positions of the spin down fermions. In order to construct an ansatz that ensures the necessary exchange antisymmetry, we construct a Slater determinant of functions $\{\phi_i\}$

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

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

$$\begin{aligned} & \phi_i(x_j, x_1, \dots, x_k, \dots, x_{j-1}, x_{j+1}, \dots, x_l, \dots, x_N) \\ &= \phi_i(x_j, x_1, \dots, x_l, \dots, x_{j-1}, x_{j+1}, \dots, x_k, \dots, x_N) \end{aligned} \quad (3)$$

Similarly, we define a function Φ^\downarrow in terms of the $\phi_{i,j}$ functions

$$\Phi^\downarrow(\vec{x}) = \frac{1}{\sqrt{N!}} \begin{vmatrix} \phi_{N_\uparrow+1, N_\uparrow+1} & \phi_{N_\uparrow+1, N_\uparrow+2} & \cdots \\ \phi_{N_\uparrow+2, N_\uparrow+1} & \phi_{N_\uparrow+2, N_\uparrow+2} & \\ \vdots & & \ddots \end{vmatrix} \quad (4)$$

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

$$\psi_F(\vec{x}) = \Phi^\uparrow(\vec{x}) \Phi^\downarrow(\vec{x}) e^{-\sum_i^N x_i^2} \quad (5)$$

The Slater determinants are antisymmetric with respect to their set of coordinates, that is, Φ^\uparrow is antisymmetric under interchange of the up spin particles, and Φ^\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 weights and biases of the neural network functions $\{\phi_i\}$, which are implemented to be identical in the number of layers and nodes.

C. Neural Network Architecture

D. Systems

In order to demonstrate the neural network solution, we apply our methodology to solve for the ground state wavefunctions and energies for a particular one-dimensional system of fermions. We consider a system of harmonically trapped fermions, consisting of two species, interacting via a contact δ -function potential.

III. TRAINING

A. Transfer Learning

When probing systems with large values of the coupling g , numerical instabilities in the gradient descent process can arise when the value of g is increased rapidly. We utilize transfer learning in order to decrease the necessary number of training steps, as well as instabilities in the optimization. The neural networks are trained on a particular value of g , and then reused for systems with larger values of g , which starts the gradient descent algorithm at a point in the space of parameters that is closer to the ground state configuration.

- For probing systems with larger g , increasing g immediately can lead to numerical instabilities in the gradient descent algorithm
- Instead, we train the model by stepping g up slowly
- After training on a particular value of g , we can use the parameters of the trained model on a system with slightly larger g , which decreases the training time and provides a smoother gradient descent

This needs to be explained somewhere

Is there a model that this is related to?

Why is this model important

Give an explanation for this?

B. Parameter Extrapolation

After the neural networks are trained on several values of the coupling constant g , we extrapolate the evolution of each parameter to systems with higher g , which decreases the training time by providing a starting point that is closer to the ground state parameters.

Given a set of previous g values, and the value of the parameter after the model is trained for each respective g , we attempt a linear fit, and compute the coefficient of determination, R^2 . If the R^2 value is below a chosen threshold, we instead attempt a linear fit on the last 3 values of the parameter. If the R^2 value of the second fit is again below the threshold, the value of the parameter for the last value of g is chosen, otherwise, the chosen linear fits are used to estimate the value of the parameter for the value of g that we are training the model on.

Explain why this was chosen

Clunky. Re-word.

Get proof of the speedup generated by the parameter evolution technique.

C. Hyperparameters

IV. RESULTS

A. Impurity

B. Attractive Regime

C. Nonperturbative Repulsive Regime

V. DISCUSSION