

Ground state energy of hard sphere Bose gas in Elliptical HO potential by VMC methods

Adrian Martinsen Kleven

Autumn 2020

Contents

| | |
|--|----------|
| 1 Abstract | 2 |
| 2 Introduction | 2 |
| 3 Theory | 2 |
| 3.1 The wave function | 2 |
| 3.2 The Hamiltonian | 3 |
| 3.3 The Local energy | 3 |
| 3.4 Optimization | 3 |
| 4 Implementation | 3 |
| 4.1 Post Analysis and Error Estimation | 3 |
| 4.2 Parallelization | 3 |
| 5 Results | 4 |
| 5.1 Non-interacting electrons | 4 |
| 5.2 Interacting electrons | 4 |
| 5.3 Electron Locations | 5 |
| 5.3.1 Metropolis Sampling | 5 |
| 5.3.2 Importance Sampling | 6 |
| 5.4 Numerical instability | 6 |
| 6 Conclusion | 8 |
| 7 Appendix A | 8 |
| 8 Appendix B | 9 |

List of Figures

| | |
|--|---|
| 1 as | 4 |
| 2 as | 4 |
| 3 as | 5 |
| 4 as | 6 |
| 5 numerically unstable run- electron positions | 7 |
| 6 numerically unstable run- electron positions | 8 |

List of Tables

1 Abstract

systems of two interacting electrons in a quantum dot to benchmark against the closed form expression provided by M. Taut. [5]. Because of the nature of this implementation, I have chosen to focus on qualitative aspects of the results in a hope to gain some insight into its problems.

2 Introduction

The quantum wave function has been successfully modelled using a restricted Boltzmann machine (RBM) in many- body lattice spin- systems [1]. The purpose of this paper is to examine its applicability in estimating the upper bound for the ground state energy of interacting two-electron systems confined in a harmonic oscillator potential.

3 Theory

We can approximate the upper bound for the ground state of a quantum mechanical system by using the variational principle. We propose a trial wave function with some variational parameters, then optimize the energy- expectation value with respect to those parameters. The Expectation value for the energy is given by

$$\langle E \rangle = \frac{\langle \psi | H | \psi \rangle}{\langle \psi | \psi \rangle}. \quad (1)$$

The probability distribution for the locations of the electrons are estimated using the Metropolis and Metropolis- Hastings algorithms. A discussion of these can be found here [3].

3.1 The wave function

The quantum mechanical wave function will be represented by a RBM. The joint probability distribution for a RBM is defined by the Boltzmann distribution

$$F_{rbm}(\mathbf{X}, \mathbf{H}) = \frac{1}{Z} e^{-\frac{1}{T_0} E(\mathbf{X}, \mathbf{H})} \quad (2)$$

where Z is the partition function

$$Z = \iint \frac{1}{Z} e^{-\frac{1}{T_0} E(\mathbf{x}, \mathbf{h})} d\mathbf{x} d\mathbf{h}. \quad (3)$$

We set $T_0 = 1$ for simplicity. Here, E is the configuration energy of the network. The configuration energy expression is unique to different problems. The one of interest to us is the Gaussian- Binary RBM whose joint probability distribution is given by equations (2), (3) and

$$E(\mathbf{X}, \mathbf{H}) = \frac{\|\mathbf{X} - \mathbf{a}\|^2}{2\sigma^2} - \mathbf{b}^T \mathbf{H} - \frac{\mathbf{X}^T \mathbf{W} \mathbf{H}}{\sigma^2} \quad (4)$$

where \mathbf{X} are the visible nodes corresponding to the coordinates of each particle, \mathbf{H} are the hidden nodes. \mathbf{a} and \mathbf{b} are the visible and hidden biases respectively. \mathbf{W} is the matrix of weights connecting nodes between layers. To elucidate the connection between the joint probability distribution and the wave function, the hidden nodes are summed over, giving a distribution solely dependent on \mathbf{X} .

$$\begin{aligned}
\Psi(\mathbf{X}) &= \frac{1}{Z} \sum_{\mathbf{h}} e^{-E(\mathbf{X}, \mathbf{h})} \\
&= \frac{1}{Z} \sum_{\{h_j\}} e^{-\sum_i^M \frac{(x_i - a_i)^2}{2\sigma^2} + \sum_j^N b_j h_j + \sum_{i,j}^{M,N} \frac{x_i w_{ij} h_j}{\sigma^2}} \\
&= \frac{1}{Z} e^{-\sum_i^M \frac{(x_i - a_i)^2}{2\sigma^2}} \prod_j^N \left(1 + e^{b_j + \sum_i^M \frac{x_i w_{ij}}{\sigma^2}} \right)
\end{aligned} \tag{5}$$

3.2 The Hamiltonian

The Hamiltonian of the system is given by a standard harmonic oscillator Hamiltonian with an included, idealized interaction term

$$\hat{H} = \sum_{i=1}^N \left(-\frac{1}{2} \nabla_i^2 + \frac{1}{2} \omega^2 r_i^2 \right) + \sum_{i < j} \frac{1}{r_{ij}} \tag{6}$$

using natural units ($\hbar = c = e = m_e = 1$).

3.3 The Local energy

The local energy

$$E_L = \frac{1}{\Psi} \hat{\mathbf{H}} \Psi \tag{7}$$

can be determined analytically (see Appendix A), giving the expression (8). The local energy has to be minimized with respect to the network parameters. The gradient for the expectation value of the local energy is given by equation (9) (see Appendix B).

3.4 Optimization

The optimization of the network parameters were done using SGD with the local energy as the cost function.

4 Implementation

4.1 Post Analysis and Error Estimation

The post analysis involved printing the energy at each sampling stage to file, then analysing it using Python. The error estimates for the expectation value of the local energy were evaluated using Marius Jonsson's Blocking code [2].

4.2 Parallelization

Building a solution for c++ in Visual Studio in x- 64 release mode, automatically vectorizes and parallelizes many loops that don't have dependencies that can't be parallelized. The only loop to be explicitly parallelized in the program was a grid search for the number of hidden nodes and the learning rate for the SGD. This loop was parallelized using OpenMP [4].

5 Results

5.1 Non- interacting electrons

Other than the network parameters that are optimized by SGD, there are

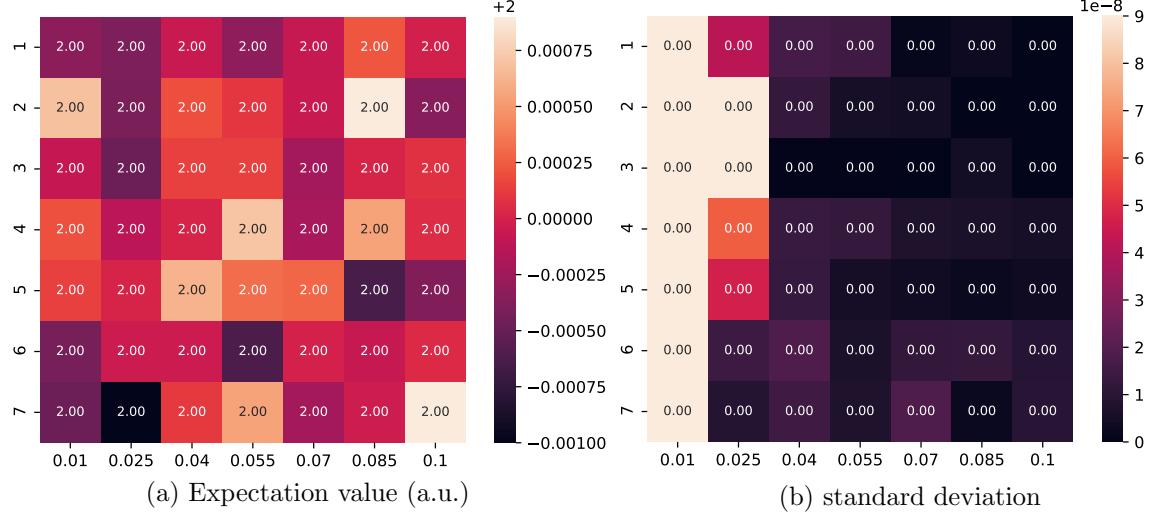


Figure 1: Two interacting electrons in two dimensions. Computed using importance sampling with 2^{14} equilibration steps, 2^{18} sampling steps and 200 learning steps. The number of hidden nodes on the y- axis and the SGD- learning rate on the x- axis. Expectation values and standard deviations were computed using blocking on the energies from the final learning step.

5.2 Interacting electrons

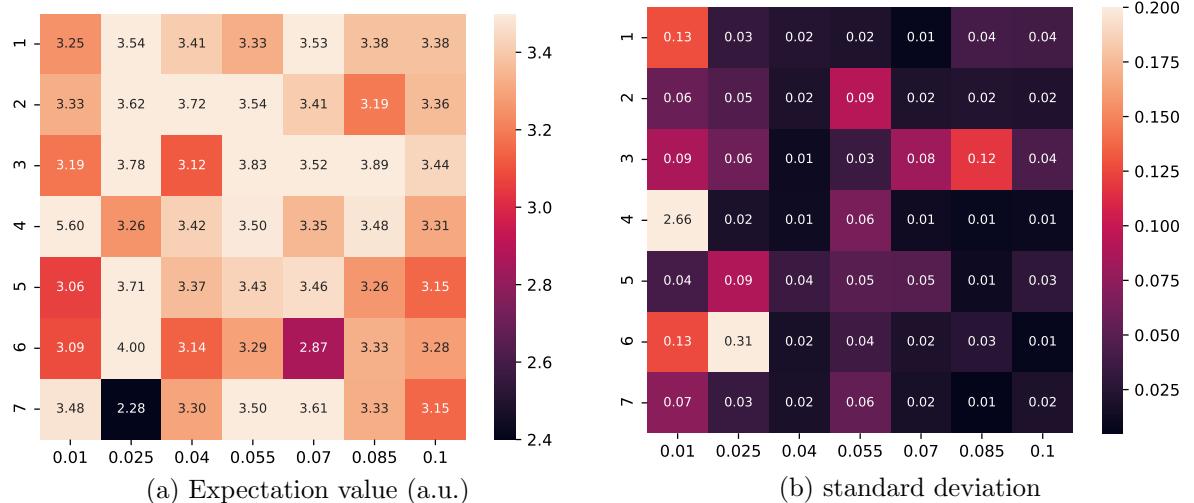


Figure 2: Two non- interacting electrons in two dimensions. Computed using importance sampling with 2^{14} equilibration steps, 2^{18} sampling steps and 200 learning steps. The number of hidden nodes on the y- axis and the SGD- learning rate on the x- axis. Expectation values and standard deviations were computed using blocking on the energies from the final learning step. Simulations that resulted in non- numeric values such as "nan" or "inf" were run again.

5.3 Electron Locations

5.3.1 Metropolis Sampling

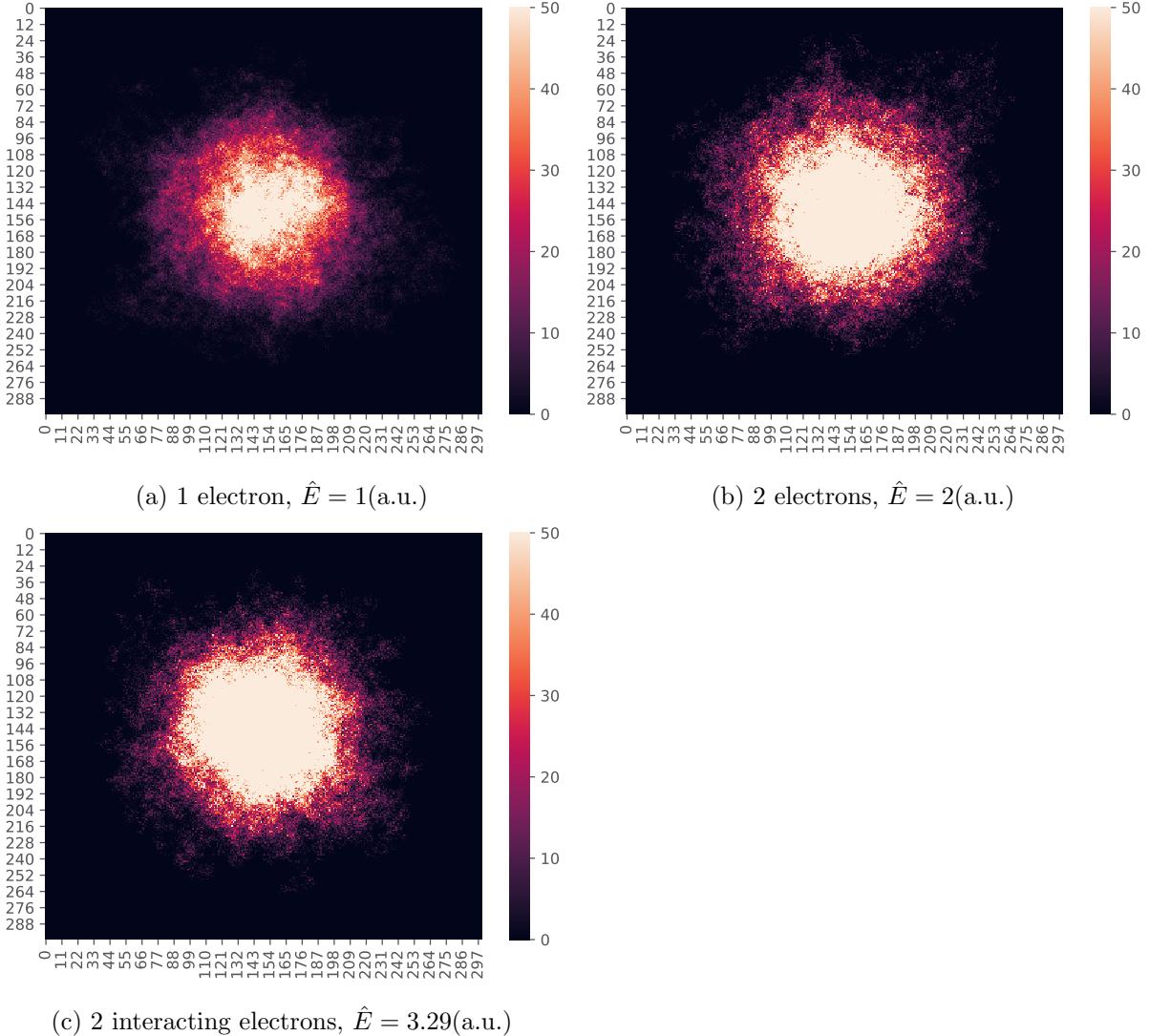


Figure 3: Electrons in two dimensions. Computed using importance sampling with 2^{18} equilibration steps, 2^{19} sampling steps and 200 learning steps. The entire interval in the x and y positions are six times the natural length scale of the system, centered at $x, y = 0$. Those intervals are split into 300 zones that record each time an electron is located in that zone in the course of the sampling steps.

5.3.2 Importance Sampling

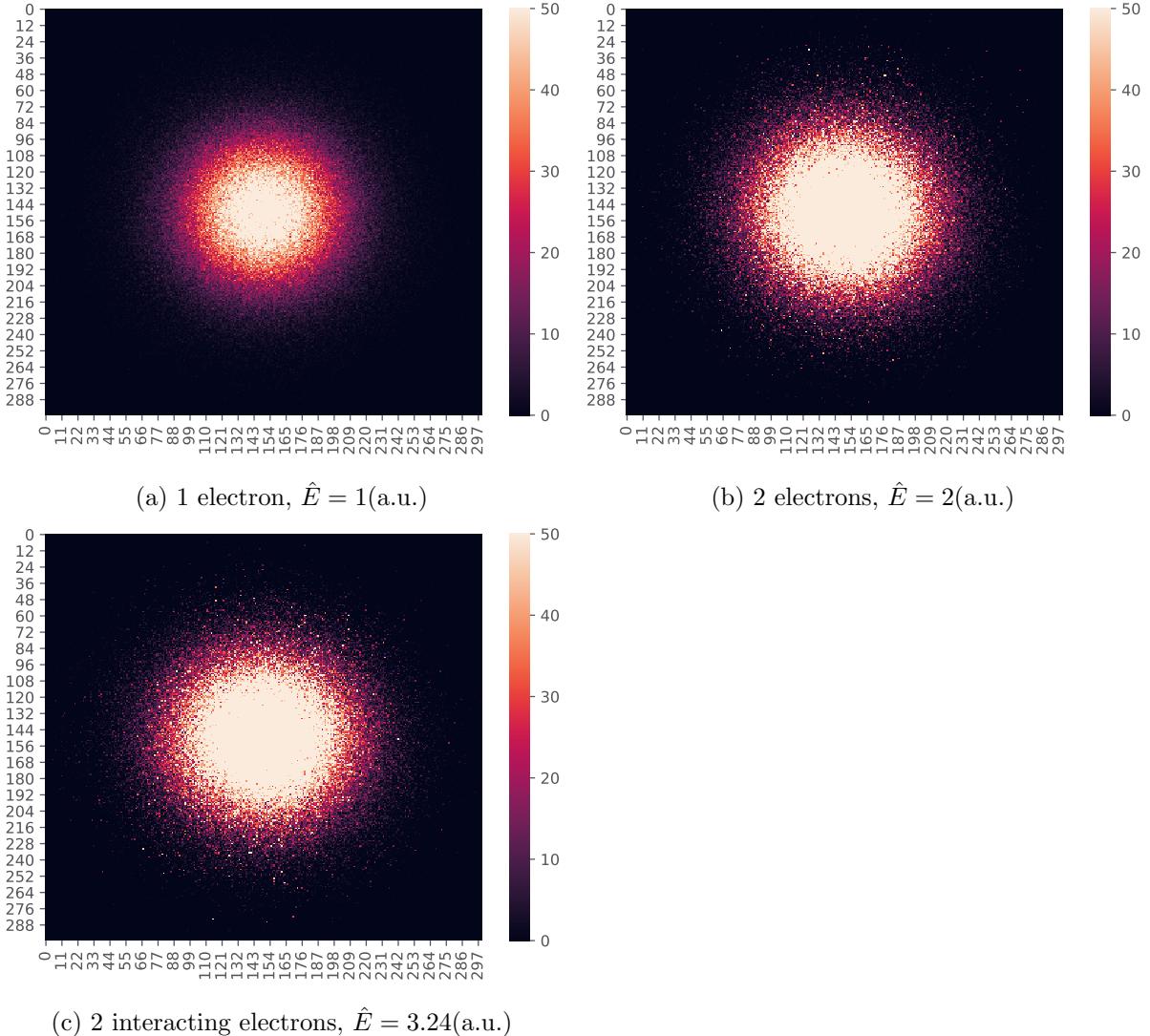


Figure 4: Electrons in two dimensions. Computed using importance sampling with 2^{18} equilibration steps, 2^{19} sampling steps and 200 learning steps. The entire interval in the x and y positions are six times the natural length scale of the system, centered at $x, y = 0$. Those intervals are split into 300 zones that record each time an electron is located in that zone in the course of the sampling steps.

The vertical line is notably absent in figure 4a, indicating that the faulty code does not lie with the interaction term, but with another function using the particles' x- position as an argument.

5.4 Numerical instability

A bug in an earlier implementation of this program would sometimes cause the energy of the system to explode. Now, I wouldn't normally include details about a bug, except that this one was quite interesting. Because of an indexing error, the y- coordinate of the second particle would never be updated in a metropolis step. The same indexing error also meant that if the

move was rejected, the program would do something like this

```
set  $P_1(y) = P_2(x)$ ;  
set  $P_2(x) = P_2(y) = Constant$ ;
```

effectively constraining the second particle to move along a small vertical line. This meant that every time two moves were rejected in a row, we would get

Result: $P_1(y) = P_2(x) = P_2(y) = Constant$

```
set  $P_1(y) = P_2(x)$ ;  
set  $P_2(x) = P_2(y)$ ;  
set  $P_1(y) = P_2(x)$ ;  
set  $P_2(x) = P_2(y)$ ;
```

This would (less strictly) constrain the first particle to align with the second, horizontally. The only (actual) degree of freedom for the system then became the x- position of the first particle. This created an obvious failure mode where every time the first particle crossed a diagonal ($P_1(x) = P_1(y)$), the interaction term would explode. The chance then, that the only relevant parameter ($P_1(x)$) would be updated in the next cycle was less than 1 in 3. The energy would then aggregate over several cycles and likely cause a memory leak. This problem would become worse, the closer the initial value of $P_2(y)$ was to zero, as the first particle would intercept the diagonal more frequently near the origin.

One example of this failure mode is seen in figure 5 below. The diagonal is marked in red. The x- and y- axes are flipped.

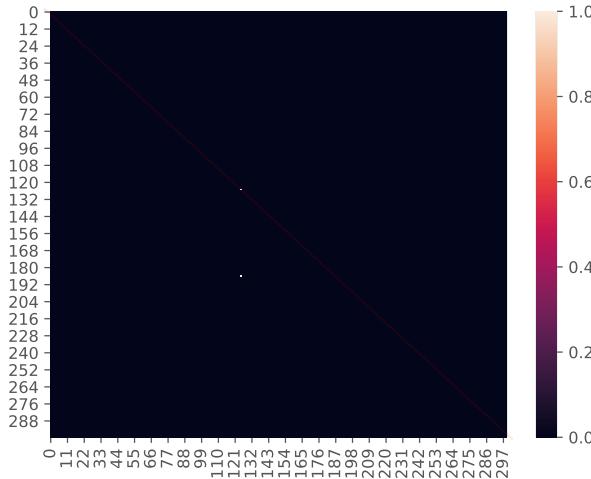


Figure 5: Position sampling for 2 interacting electrons in 2 dimensions using importance sampling. In this case, the energy was calculated to be -1374 a.u . The electrons remained at those locations for every sampling step. The second particle (most likely) is located on a diagonal, and the first (...) is horizontally locked to it.

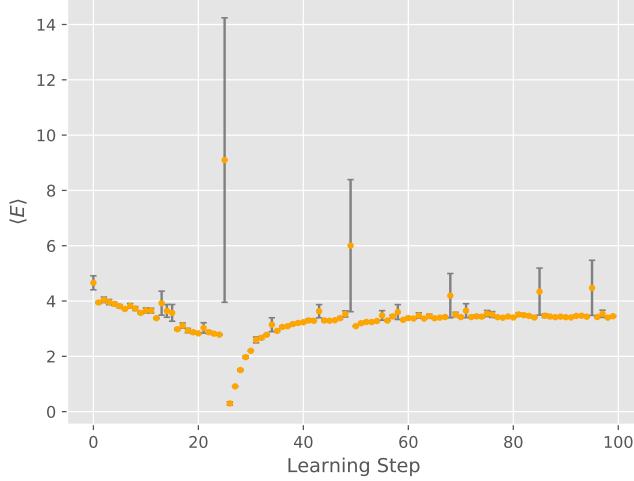


Figure 6: 2 interacting particles in 2 dimensions. Local energy as a function of SGD learning step using a learning rate of 0.05. Sparse adverse events (large errorbars) stop SGD from converging properly

6 Conclusion

7 Appendix A

Using the hamiltonian in (6) we get that the local energy is given by

$$\begin{aligned} E_L &= \frac{1}{\Psi} \mathbf{H} \Psi \\ &= -\frac{1}{2} \frac{1}{\Psi} \sum_i^M \frac{\partial^2}{\partial X_i^2} \Psi + \frac{1}{2} \omega^2 \sum_i^M X_i^2 + \sum_{j < k} \frac{1}{r_{jk}} \end{aligned}$$

with M being the number of visible nodes in the network. The only non-trivial contribution comes from the Laplacian. We can simplify the differential term like this

$$\frac{\partial^2}{\partial X_i^2} \ln \Psi = \frac{\partial}{\partial X_i} \left(\frac{1}{\Psi} \frac{\partial}{\partial X_i} \Psi \right) = \frac{1}{\Psi} \frac{\partial^2}{\partial X_i^2} \Psi - \left(\frac{\partial}{\partial X_i} \ln \Psi \right)^2$$

giving us this expression

$$\begin{aligned} E_L &= \frac{1}{\Psi} \mathbf{H} \Psi \\ &= -\frac{1}{2} \sum_i^M \left(\frac{\partial^2}{\partial X_i^2} \ln \Psi + \left(\frac{\partial}{\partial X_i} \ln \Psi \right)^2 \right) + \frac{1}{2} \omega^2 \sum_i^M X_i^2 + \sum_{j < k} \frac{1}{r_{jk}}. \end{aligned} \tag{8}$$

Thankfully, the natural log of the wave function is easily computed to be

$$\ln \Psi = -\ln Z - \sum_i^M \frac{(X_i - a_i)^2}{2\sigma^2} + \sum_j^N \ln \left(1 + e^{b_j + \sum_i^M \frac{X_i W_{ij}}{\sigma^2}} \right).$$

The differentials then become trivial to solve, and we get

$$\frac{\partial}{\partial X_i} \ln \Psi = \frac{1}{\sigma^2} \left(a_i - X_i + \sum_j^N W_{ij} \zeta(u_j) \right)$$

and

$$\frac{\partial^2}{\partial X_i^2} \ln \Psi = -\frac{1}{\sigma^2} + \frac{1}{\sigma^4} \sum_j^N W_{ij}^2 \zeta(u_j) \zeta(-u_j)$$

where $\zeta_j = \frac{1}{e^{-u_j} + 1}$ is the familiar sigmoid function with $u_j = b_j + \frac{1}{\sigma^2} \sum_k^M X_k W_{kj}$. This concludes the calculation.

8 Appendix B

The gradient of the local energy with respect to the network parameters is given by

$$G_i = \frac{\partial \langle E_L \rangle}{\partial \alpha_i} = 2 \left(\left\langle E_L \frac{1}{\Psi} \frac{\partial \Psi}{\partial \alpha_i} \right\rangle - \langle E_L \rangle \left\langle \frac{1}{\Psi} \frac{\partial \Psi}{\partial \alpha_i} \right\rangle \right). \quad (9)$$

From the expression for the wave function in 7 the derivative of the wave function with respect to the weights and biases are easily calculated to be

$$\begin{aligned} \frac{\partial}{\partial a_i} \ln \Psi &= \frac{X_i - a_i}{\sigma^2}, \\ \frac{\partial}{\partial b_i} \ln \Psi &= \zeta_i \end{aligned}$$

and

$$\frac{\partial}{\partial W_{ij}} \ln \Psi = \frac{X_i}{\sigma^2} \zeta_j.$$

this concludes the calculation.

References

- [1] Giuseppe Carleo and Matthias Troyer. Solving the quantum many-body problem with artificial neural networks. *Science*, 355(6325):602–606, 2017.
- [2] Marius Jonsson. Standard error estimation by an automated blocking method. *Phys. Rev. E*, 98:043304, Oct 2018.
- [3] A. Kleven. Ground state energy of hard sphere bose gas in elliptical ho potential by vmc methods. Feb 2020.
- [4] OpenMP Architecture Review Board. OpenMP application program interface version 3.0, May 2008.
- [5] M. Taut. Two electrons in an external oscillator potential: Particular analytic solutions of a coulomb correlation problem. *Phys. Rev. A*, 48:3561–3566, Nov 1993.