Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin

Aaron Fitzpatrick

17327368

Machine Learning Density Functional Theory For The Spinless Hubbard Model

Supervised by:
Stefano Sanvito

April 2021

School of Physics
Trinity College Dublin.

Abstract

According to the Hohenberg-Kohn theorems, the energy of a many body system can be expressed as a functional of the particle densities. For lattice models such as the spinless Hubbard model, the on-site occupation is sufficient to determine the ground-state energy. In this report I demonstrate that a simple neural network, trained on exact results obtained by diagonalising the Hamiltonian matrix, can correctly learn predict the ground state Helmholtz free energy of the spinless Hubbard model in the canonical ensemble. I then show that the model can satisfy both of the Hohenberg-Kohn theorems, namely that there exists a one to one map between the external potential (on-site energies) of the problem and the ground state density, and that the theory is variational.

Acknowledgements

I would like to extend my thanks to Prof. Stefano Sanvito for his supervision throughout the course of this project, including his help with the overall direction of the workload. The advice, support and initial information provided to me helped immensely in the completion of the project.

I also thank Dr. Rajarshi Tiwari for his useful input and suggestions, particularly for his help with the beginning of the project and helping to find any mistakes that I may have made.

Finally I would like to acknowledge all of my course mates who have made this year much easier than it may have been otherwise. I look forward to seeing you all in better times.

Contents

1 Introduction

1.1 Aims and Objectives:

Successfully modelling systems of strongly correlated electrons is one of the most important goals in condensed matter physics. However, due to the well known difficulties associated with solving the Schrodinger equation for an arbitrarily large system, numerical methods are almost always necessary. At present, the most widely used method for modelling such interactions is density functional theory (DFT), the foundations of which were laid down by Hohenberg and Kohn in their highly consequential papers in 1964 [1]. One model often used to describe systems of this nature is the Hubbard model [2]. The aim of this project was to successfully write python scripts to generate the Hamiltonian matrices for varying parameters within the spinless Hubbard model. The ground-state energy of such systems would then be found via exact diagonalisation of the matrices and used to train a machine learning algorithm to predict the ground-state energies of other such systems. This is made possible due to the work of Hohenberg and Kohn in devising their famous theorems in which the existence of a universal functional for describing the energy of interacting many body systems was proposed. This will be discussed in more detail in the following sections.

1.2 The Spinless Hubbard Model

The Hubbard Model was first introduced by physicist John Hubbard in 1963 as an approximate model for the interaction of electrons in narrow energy bands. The Hamiltonian of the model is simply the same as that of the well known tight-binding model with an added term for the interaction between the electrons on the lattice, which is assumed to be 0 in the tight-binding model. For a chain of L atoms the Hamiltonian is given by (in the spinless case),

$$\hat{H} = \sum_{i=1}^{L} \epsilon_i \hat{n}_i - t \sum_{i=1}^{L} \left(\hat{c}_i^\dagger \hat{c}_{i+1} + \hat{c}_{i+1}^\dagger \hat{c}_i \right) + V \sum_{i=1}^{L} \hat{n}_{i+1} \hat{n}_i \quad , \tag{1}$$

where

1. $\epsilon_i =$ on site energy at site i.
2. $\hat{c}_i^\dagger$, $\hat{c}_i$ are the creation/annihilation operators.
3. $\hat{n}_i$ the number operator $= \hat{c}_i^\dagger \hat{c}_i$

As can be seen the Hamiltonian is parametrised by $t, V$and $\{\epsilon_i\} = \vec{\epsilon}$, which represent the hopping parameter, the interaction potential strength and the on-site energies respectively.

The model shows many qualitative improvements on the description of electrons in solids compared to that of the tight-binding model, due to the inclusion of the interaction term. For example, the

Hubbard model correctly predicts the existence of Mott insulators [§] due to its inclusion of the electron-electron repulsion term, something not accounted for in the free electron tight-binding model.

It is worth noting that in his original work, Hubbard considered spin 1/2 fermions and the electron repulsion interaction was between electrons of opposite spins occupying the same atomic site. The case of spinless fermions is also well studied in a similar context and will be the case referred to throughout the course of this report [3, 4]. The particles are still assumed to obey Pauli's exclusion principle and as such a maximum of 1 particle may inhabit each atomic site at any given time.

1.3 The Hamiltonian Matrix

As is standard in quantum mechanical problems a matrix representation of the various operators appearing in the Hamiltonian of the system must be found. From Eq.(1), there are three terms appearing which I refer to from here on as:

1. The on-site term or $\hat{H}_{os}$
2. The hopping term or $\hat{T}$
3. The many body term or $\hat{V}$

such that

$$\hat{H} = \hat{H}_{os} + \hat{T} + \hat{V} \quad . \tag{2}$$

Each term can be understood as follows

1. $\hat{H}_{os}$ is the contribution to the total energy associated to a particle depending on its position. This $\vec{\epsilon}$ term therefore can be seen to play a similar role to an external potential, as it is simply a contribution to the total energy of the system due to the locations of its constituent parts.
2. $\hat{T}$ represents the kinetic energy of the system. By looking at Eq.(1), we can see the $\hat{T}$ term uses creation and annihilation operators to transfer particles between adjacent atomic sites. The constant t (known as the hopping parameter) parameterises how easy it is for particles to travel between sites. A value of $t = 0$ would result in 0 kinetic energy contribution and so one should see no movement of particles.
3. $\hat{V}$ is the contribution to total energy of the system from the interaction of individual particles on adjacent sites to one another.

To construct the Hamiltonian matrix explicitly, a set of basis states is first defined. I do this below for the case of a 1D chain of 4 atoms occupied by 2 fermions, as provided as an example by Prof Sanvito when introducing me to the model.

[§] Materials predicted to be conductors in conventional band theory, but are measured in the lab to be insulators.

First I define the basis states of a given site i:

$$|0\rangle_i \quad |1\rangle_i \ ,$$

representing a site i that is either unoccupied, $|0\rangle_i$, or occupied, $|1\rangle_i$. The basis for a system of four sites is then :

$$|\alpha_1\rangle_1 \otimes |\alpha_2\rangle_2 \otimes |\alpha_3\rangle_3 \otimes |\alpha_4\rangle_4 = |\alpha_1\alpha_2\alpha_3\alpha_4\rangle \ ,$$

where α_i can take the values {0,1} as above. It is this tensor product structure that is responsible for the scaling problems associated with solving the theory analytically for many particles.

In order to fully describe the system all possible permutations of α_i must be considered and so the total number of basis states, or equivalently the dimension of the Hilbert space $\mathcal{H}$, is given by,

$$dim(\mathcal{H}) = \left(\begin{array}{c} L \\ N \end{array} \right) = \frac{L!}{N!\,(L-N)!} \quad , \tag{3}$$

with L the length of the atom chain and N the number of fermions present. For the system specified above the basis states obtained are:

$$\begin{aligned} &|1\rangle = |1100\rangle\,, |2\rangle = |1010\rangle\,, |3\rangle = |1001\rangle \\ &|4\rangle = |0110\rangle\,, |5\rangle = |0101\rangle\,, |6\rangle = |0011\rangle \end{aligned} . \tag{4}$$

$\hat{H}$ has matrix elements given by,

$$\hat{H}_{ij} = \langle i|\,\hat{H}_{os}\,|j\rangle + \langle i|\,\hat{T}\,|j\rangle + \langle i|\,\hat{V}\,|j\rangle \quad , \tag{5}$$

that can be simply calculated as follows:

1. $\langle i|\,\hat{H}_{os}\,|j\rangle = \delta_{ij}\sum_{l=1}^{L}\epsilon_l\alpha_l^i$
2. $\langle i|\,\hat{T}\,|j\rangle = t$ [*]
3. $\langle i|\,\hat{V}\,|j\rangle = \delta_{ij}\sum_{l=1}^{L}V\alpha_l^i\alpha_{l+1}^i$ [‡]

For the $L = 4$, $N = 2$ example with basis states above the Hamiltonian is

$$\begin{pmatrix} \epsilon_1+\epsilon_2+V & t & 0 & 0 & t & 0 \\ t & \epsilon_1+\epsilon_3 & t & t & 0 & t \\ 0 & t & \epsilon_1+\epsilon_4+V & 0 & t & 0 \\ 0 & t & 0 & \epsilon_2+\epsilon_3+V & t & 0 \\ t & 0 & t & t & \epsilon_2+\epsilon_4 & t \\ 0 & t & 0 & 0 & t & \epsilon_3+\epsilon_4+V \end{pmatrix}$$

[*] this is equal to t if states differ by moving a single fermion between adjacent sites, 0 otherwise

[‡] Add V for each time two fermions occupy adjacent sites simultaneously

It is important to note here that periodic boundary conditions are in use. i.e site 1 is a nearest neighbour of site 4.

1.4 Density Functional Theory

As mentioned, one of the greater challenges faced when attempting to solve many-body quantum mechanical systems is that of diagonalising Hamiltonian matrices whose dimensions grow exponentially with the number of particles involved. From Eq.(3) we can see this to be true immediately for the Hubbard model. In order to diagonalise the Hamiltonian of the Hubbard model in the case of $L = 20$, $N = 10$ would involve working with a matrix of dimension 184,756. As a result we are constrained to studying smaller systems when we attempt to solve exactly.

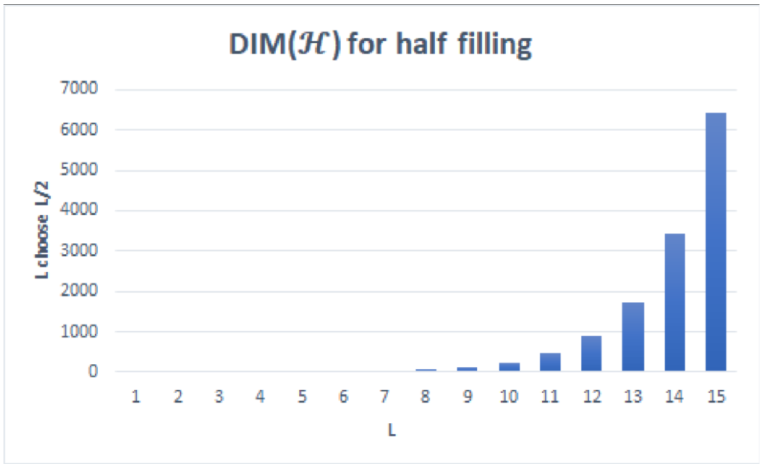


Figure 1: Dim($\mathcal{H}$) as number of sites increases for $N = L/2$. The exponential growth of Dim($\mathcal{H}$) is easy to see.

In order to get around this issue we turn to DFT, the main ideas of which are contained in the Hohenberg Kohn theorems. In the original papers it is shown that for an interacting electron gas in an external potential, there exists a unique universal functional of the density $F[n(\mathbf{r})]$, independent of the potential such that the energy of the system is given by ,

$$E = \int v(\mathbf{r})n(\mathbf{r})d\mathbf{r} + F[n(\mathbf{r})] \quad . \tag{6}$$

Moreover, it is shown that this expression has at its minimum the correct ground state energy associated to the potential $v(\mathbf{r})$ and that there exists a one to one relationship between $v(\mathbf{r})$ and $n(\mathbf{r})$. I give the proof below for completeness.

Proof: We begin by proving that the external potential is uniquely determined by the fermion density n($\mathbf{r}$). The proof is by contradiction as follows.

The generic Hamiltonian for such a problem is of the form $\hat{H} = \hat{T} + \hat{U} + \hat{V}$, where

$$\begin{aligned}
&\hat{T} \equiv \tfrac{1}{2}\int \boldsymbol{\nabla}\psi^*(\mathbf{r})\boldsymbol{\nabla}\psi(\mathbf{r})d\mathbf{r}\\
&\hat{V} \equiv \int v(\mathbf{r})\psi^*(\mathbf{r})\psi(\mathbf{r})d\mathbf{r}\\
&\hat{U} = \tfrac{1}{2}\int \tfrac{1}{|\mathbf{r}-\mathbf{r}'|}\psi^*(\mathbf{r})\psi^*\left(\mathbf{r}'\right)\psi\left(\mathbf{r}'\right)\psi(\mathbf{r})d\mathbf{r}d\mathbf{r}'
\end{aligned} \tag{7}$$

Now assume two external potentials $v_1(\mathbf{r})$ and $v_2(\mathbf{r})$, each necessarily giving rise to two distinct Hamiltonians and therefore ground states, denoted by $\hat{H}_1, \hat{H}_2, \Psi_1, \Psi_2$, result in identical ground state densities.

By the variational principal the following relationship must hold for non-degenerate ground states [5]

$$E_1 = \langle\Psi_1|\,\hat{H}_1\,|\Psi_1\rangle < \langle\Psi_2|\,\hat{H}_1\,|\Psi_2\rangle \ .$$

The form of $\hat{H}$ allows us to re-write the above term as,

$$\langle\Psi_2|\,\hat{H}_1\,|\Psi_2\rangle = \langle\Psi_2|\,\hat{H}_2 - \hat{V}_2 + \hat{V}_1\,|\Psi_2\rangle \ ,$$

so that that inequality now becomes

$$E_1 < E_2 + \langle\Psi_2|\,(\hat{V}_1 - \hat{V}_2)\,|\Psi_2\rangle \ .$$

Now, since by assumption the two ground-state densities are identical we can write the above as

$$E_1 < E_2 + \int dr[V_1(\mathbf{r}) - V_2(\mathbf{r})]n(\mathbf{r}).$$

Exchanging the 1 and 2 labels above yields

$$E_2 < E_1 + \int dr[V_2(\mathbf{r}) - V_1(\mathbf{r})]n(\mathbf{r}).$$

The sum of the two above equations yields

$$E^1 + E^2 < E^2 + E^1,$$

which is trivially false, therefore we can see that the external potential is uniquely fixed by $n(\mathbf{r})$.

Since the external potential is uniquely determined by the fermion density, and it is the potential which fixes the ground-state, all other observables are also determined by the density. This means one can write the energy for example as a functional of the density (as in Eq.(6)), if we simply define the value of the functional as

$$F[n(\mathbf{r})] = \langle\Psi|\,\hat{T} + \hat{U}\,|\Psi\rangle\,. \tag{8}$$

To show that the minimum of the energy functional is indeed realised by the ground state density, we consider the energy as a functional of the state Ψ,

$$E[\Psi] = \langle\Psi|\, \hat{T} + \hat{U} + \hat{V}\, |\Psi\rangle \,. \tag{9}$$

Clearly this has its minimum given by the correct ground state Ψ. Let $\tilde{\Psi}$ be the ground state associated to a different external potential $\tilde{v}(\mathbf{r})$. By Eq.(8) and Eq.(9)

$$E[\tilde{\Psi}] = \int v(\mathbf{r})\tilde{n}(\mathbf{r})d\mathbf{r} + F[\tilde{n}(\mathbf{r})] > E[\Psi] = \int v(\mathbf{r})n(\mathbf{r})d\mathbf{r} + F[n(\mathbf{r})]$$

So we can see the variational nature of Eq.(6) is established.

1.5 Applications to Lattice Models

While the original formulation of the theory applied to electrons in an external potential interacting via the usual Coulomb repulsion, it has also been shown that the theory can be correctly applied to lattice models by means by defining an appropriate density to realise the problem [6]. The simplest density one could consider is the vector of site occupations $\{n_i\} = \vec{n}$. In this formulation Eq.(6) takes the form

$$E[\{n_i\}] = \sum_i \epsilon_i n_i + F[\{n_i\}], \tag{10}$$

by identifying $\vec{\epsilon}$ with the external potential $v(\mathbf{r})$ as mentioned in the beginning. The form of this equation allows us to easily define the exact value of $F[\{n_i\}]$ identically to how it is defined by Eq.(8)

$$F[\{n_i\}] = \langle\Psi|\hat{T} + \hat{V}|\Psi\rangle^{\S}$$

By using DFT we can restrict the complexity of the problem to solve as the functional we will search for remains a functional of site occupations and so scales linearly with chain length L as opposed to exponentially as in the exact diagonalisation case.

The functional $F[\{n_i\}]$ is universal for a given configuration of atomic sites L and number of fermions N, with an interaction strength V and hopping parameter t, that is to say it does not depend on the on site energies $\{\epsilon_i\}$ found in the first term of the Hamiltonian. It must be noted that the geometry of the problem also matters, as the kinetic term of the Hamiltonian will vary. For example in a ring geometry there will be a matrix element for the hopping term between atom L and atom 1, whereas for the chain this will remain 0. All that is known about the functional is that it exists, it is not known how one finds it exactly, as such various methods such as the Kohn-Sham [7] scheme are used to approximate it and many of its properties are known.

[§] Due to an unfortunate clash of notation, the seen in the Hohenberg Kohn proof is associated with the I define Eq.(2), whilst the seen in the Hohenberg Kohn proof is associated to the $_{os}IdefineinEq.(2)(11)$

1.6 Different ensembles

The above DFT formulation of the Hubbard problem has all been carried out with no consideration of the temperature or particle number conservation i.e. it has been carried out in the microcanonical ensemble. It has been shown, however, that the theory can be extended to analyse the theory of interacting electrons in thermal equilibrium at a temperature $T \neq 0$, namely in the canonical and grand canonical ensembles [8]. In the results section of the report the data will, in fact, be in relation to the canonical ensemble, which has a similar relation to that of Eq.(6).

$$\begin{aligned} H &= \int v(\mathbf{r}) n(\mathbf{r}) d\mathbf{r} + G[n(\mathbf{r})] \\ \rightarrow H[\{n_i^{C,eq}\}] &= \sum_i \epsilon_i n_i^{C,eq} + G[\{n_i^{C,eq}\}], \end{aligned} \tag{12}$$

where now the $n_i^{C,eq}$ represent the ground state densities in thermal equilibrium and H is the Helmholtz free energy of the system.

For the grand canonical ensemble the relevant statements are

$$\begin{aligned} \Omega &= \int v(\mathbf{r}) n(\mathbf{r}) d\mathbf{r} + K[n(\mathbf{r})] \\ \rightarrow \Omega[\{n_i^{GC,eq}\}] &= \sum_i \epsilon_i n_i^{GC,eq} + K[\{n_i^{GC,eq}\}], \end{aligned} \tag{13}$$

with Ω the grand potential of the system and the $n_i^{GC,eq}$ similarly defined as the ground state occupation at thermal equilibrium.

1.7 Machine learning

Machine learning (ML) is a branch of artificial intelligence (AI) focused on building applications that learn to predict outcomes from data and improve their accuracy without being programmed to do so [9]. The standard problem of the field is to take some set of input vectors ,$\vec{v_i}$, and their associated outputs, α_i, and find an implicit representation of the map between them, namely it is intended to find the function M that acts as

$$M : \vec{v_i} \rightarrow \alpha_i. \tag{14}$$

The ML algorithm finds this relationship by taking in often vast quantities of input data and interpolating between the data points to ”learn” the patterns existing within. The problem of finding the universal functional is an ideal one for such methods. Due to the relationships established by Hohenberg and Kohn, we are searching for a functional that maps in a very similar fashion:

$$M : \vec{n}_{GS} \rightarrow e_{GS} \tag{15}$$

where $\vec{n}_{GS}$ is a given fermion density vector in the ground state with components given by

$$n_i = \left\langle \Psi^{GS} \right| \hat{c}_i^\dagger \hat{c}_i \left| \Psi^{GS} \right\rangle , \tag{16}$$

and e_{GS} is the ground state energy of such a system. In practice we actually search for the map between the input parameters and the universal functional, namely

$$M : n\vec{_{GS}} \to E[\{n_i\}] - \textstyle\sum_i \epsilon_i n_i = F[\{n_i\}] \ .$$

Note that the density as defined in Eq.(16) is specific to the spinless Hubbard model in the microcanonical ensemble. Due to the first Hohenberg Kohn theorem stating the bijection between the external potential and density functions, one would expect a machine learning algorithm could also learn to map directly

$$M : \vec{n}_{GS} \to \vec{\epsilon}$$

and indeed this has been achieved by others [4] and will be demonstrated in this report also.

For the case of the canonical ensemble the "density" one must use is the site occupations at thermal equilibrium, given by

$$n_i^{C,eq} = \sum_n \left\langle \Psi_n \right| \hat{c}_i^\dagger \hat{c}_i \left| \Psi_n \right\rangle \frac{\mathrm{e}^{-\beta E_n}}{Z} \quad , \tag{17}$$

where $|\Psi_n\rangle$ is the n'th eigenstate of the system and $Z = Tr(\mathrm{e}^{-\beta \hat{H}})$ is the partition function. E_n is the energy of the n'th state. The grand grand canonical equilibrium density is then given by

$$n_i^{GC,eq} = \sum_N \sum_n \left\langle \Psi_n^N \right| \hat{c}_i^\dagger \hat{c}_i \left| \Psi_n^N \right\rangle \frac{\mathrm{e}^{-\beta(E_n^N - \mu N)}}{\mathcal{Z}} \quad , \tag{18}$$

with $\left|\Psi_n^N\right\rangle$ the n'th eigenvector of the system with N fermions present. μ is the chemical potential of the system, and $\mathcal{Z} = Tr(\mathrm{e}^{-\beta(\hat{H} - \mu \hat{N})})$.

Using the definitions above, the data set required to train the simple neural network can be generated and the problem solved.

2 Methods

The three main goals to be achieved throughout the course of the project were:

1. Successfully write python code to generate the Hamiltonian of the spinless Hubbard model for given input parameters, diagonalise this exactly and record the ground state electron density distributions, ground state energies and value of the universal functional F.
2. Train a machine learning algorithm to be able to successfully predict the value of the universal functional for a given input generated in step 1.
3. Demonstrate that the results were consistent with the Hohenberg-Kohn theorems.

2.1 Generation of the Hamiltonian matrix

An important consideration when generating the Hamiltonian matrix for the problem at hand was efficiency. Since the data that would be used to train the ML model would be obtained via exact diagonalisation, it was important not only to write efficient code to perform the diagonalisation, but also to generate the matrices in the first place. PC memory would be a scarce resource and so ways around storing large arrays of numbers would have to be considered. The generation occurred as follows.

1. All basis states were encoded in the bit representation of integers. E.g the state $|0011\rangle$ was represented by the number 3. For the sample states given in Eq.(4) this translates to a single array containing all of the basis states as

   $$\text{States} = [12{,}10{,}9{,}6{,}5{,}3]$$

   Assuming a Hubbard model chain of length L, this representation of the basis states requires far less memory, requiring we store only a single array of $dim(\mathcal{H})$ integers as opposed to $dim(\mathcal{H})$ arrays of L integers valued 0 or 1. Though admittedly negligible in the sample provided, this would become important if one wished to scale up to larger problems.

2. Bit-wise operations on these integers could then be used to mimic the operations of the Hamiltonian given in Eqs.(1,5). This allowed for swifter manipulation of the basis states than would be possible if integer arithmetic were used. E.g. counting the number of set bits in an integer is quicker than summing over all values in an array of 1 and 0 integers.

3. Once the Hamiltonian was generated, it was diagonalised using the *numpy.linalg* routine, which is already highly optimised and as such suitable for the problem.

The code was written in a class structure with the following parameters

Variable	**Description**
sites = 8	Length of the chain L
filling = 4	The number of fermions present
pbcs = true	Boolean indicating use of periodic boundary conditions
$t = 1$	Float for the hopping parameter
eps = random	Array containing all on site energies
$V = 3$	Interaction potential strength
gridtype = 1D	Dimension of the problem.

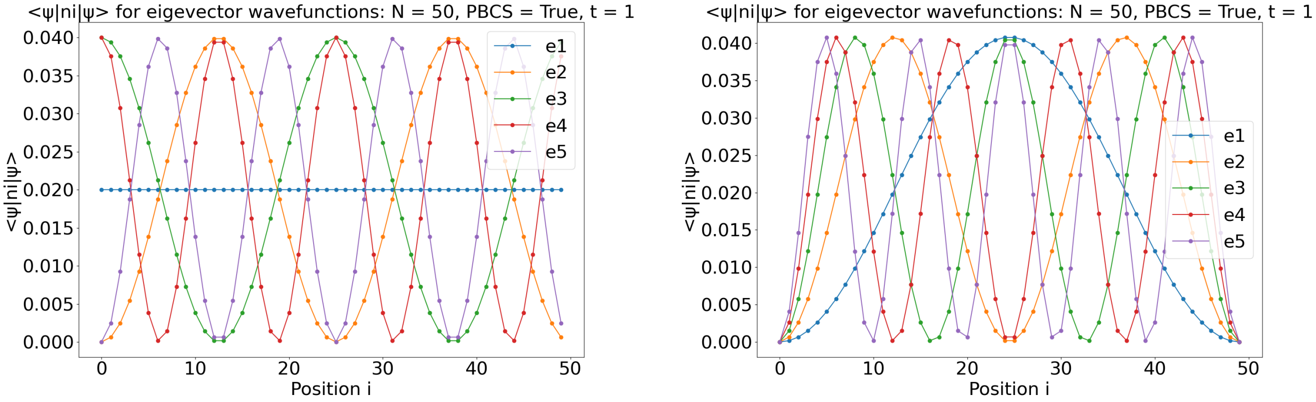
This way by simply importing the module ”hubbard.py” into any other script one could specify completely the problem at hand and immediately generate a Hamiltonian matrix. The temperature scale of the problem would be defined externally to the instance of the class.

2.2 Data generation and testing

Once the "hubbard.py" module script was complete, data could be obtained by producing sets of random on-site energies and performing the necessary computations. From each instance of the class the on site energies, the resulting density values determined by Eq.(17) and value of the functional $G\{n_i\}$ were recorded and stored in a .csv file for feeding to the neural network. A number of tests were carried out to determine whether or not the Hubbard class was generating sensible data based on the following requirements:

1. The homogeneous case (i.e where all on site energies were equal) should result in a similarly homogeneous density distribution, i.e $n_i^{eq} = L/N \qquad \forall i$
2. A particularly large on-site energy for a given site i should result in $n_i \to 0$
3. Given a hopping parameter of $t = 0$, values of n_i should be strictly 0 or 1 (i.e particles are completely localised and cannot move between sites)
4. The sum $\sum_i^L n_i = N$ should hold i all cases.[†]

Below are some plots made for the case of a single fermion to test these conditions.



(a) $< \hat{n}_i >$ at each site for the first 5 eigenvectors of the $L = 50$, homogeneous case.

(b) $< \hat{n}_i >$ at each site for the first 5 eigenvectors of the $L = 50$, inhomogeneous case with large on site energies at the end points.

Figure 2: Figures demonstrating the fulfilment of the first two testing conditions for the case of a single fermion. In fig.(a) it can be seen that the ground state eigenvector shows $< \hat{n}_i >= 0.020 = L/N$ as predicted in the homogeneous case. In fig.(b) the same graph is produced for the inhomogeneous case, where the on site energies at the end points are set to be very large. This results in $< \hat{n}_1 >=< \hat{n}_L >\approx 0$ for all eigenvectors as expected.

[†] This may not be true when considering the problem in the grand canonical ensemble, where the particle number N is allowed to vary.

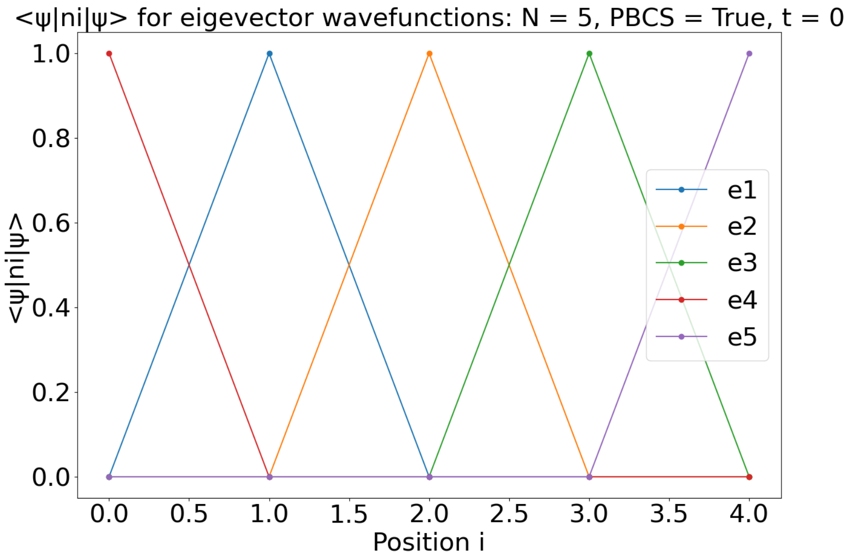


Figure 3: With $t = 0$, the particles become completely localised. Each eigenvector shows an expectation value $< n_i >= 1$ at exactly one point, with it being $= 0$ at all other points. This is the effect of the fermion not being able to 'hop' between neighbouring sites. In fact all of the eigenstates are degenerate in this case.

Condition 4 was also found to hold in all cases, and can be easily appreciated from the form of both the ground state eigenvector in Figure 2(a) and Figure 3.

2.3 Creating the ML Model

In order to implement the machine learning algorithm , the Keras API [10] for Tensorflow 2.0 was used. This allowed for swift implementation with an easy to use interface and powerful pre-established libraries. The neural network created consisted of an input layer, followed by 3 fully connected layers with 32, 64 and 128 nodes and finally output layer, which would be configured to output either a scalar if trained to predict the value of $F[\{n_i\}]$ or vector of length L if trained to predict $\vec{\epsilon}$. Note for the results graphs presented in the following section, the quantities predicted are in the canonical ensemble, where the functional to be learned is defined as :

$$g[\{n_i\}] = G[\{n_i\}]/L$$

where $G[\{n_i\}]$ is the functional appearing in Eq.(12). In effect all values in Eq.(12) have been divided by L before feeding to the network. The temperature is taken to be $T = 1$ and similarly the for simplicity the Boltzmann constant is also taken as $k = 1$

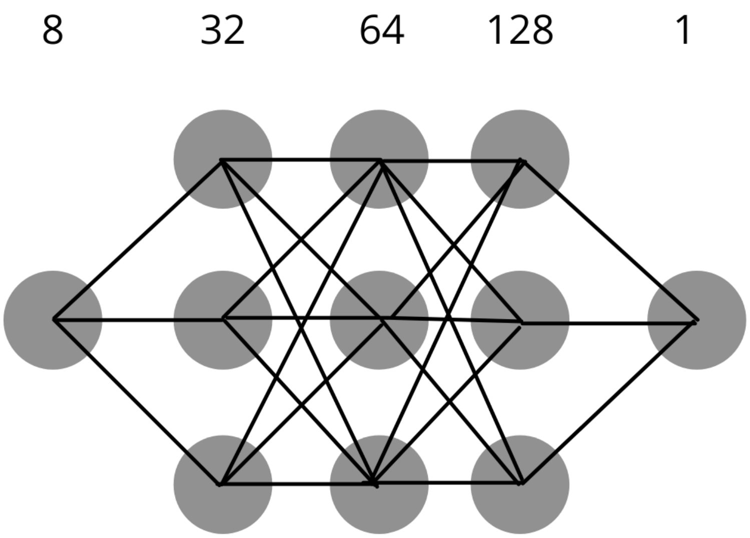


Figure 4: Visual representation of the structure of the neural network used to predict the value of $g[\{n_i\}]$ for the case of 8 atomic sites. The network taks the 8 component vector $\vec{n}$ as an input, passes it through 3 fully connected layers with 32, 63 and 128 nodes, and outputs a single scalar value.

Once the network was trained fresh data sets could be fed to the network which would predict the value of the functional. This is possible due to the fact the functional is one of the density alone, and so ground state densities obtained by diagonalising matrices with different on site energies would be of no effect.

The model was trained on 50,000 samples with the following parameters.

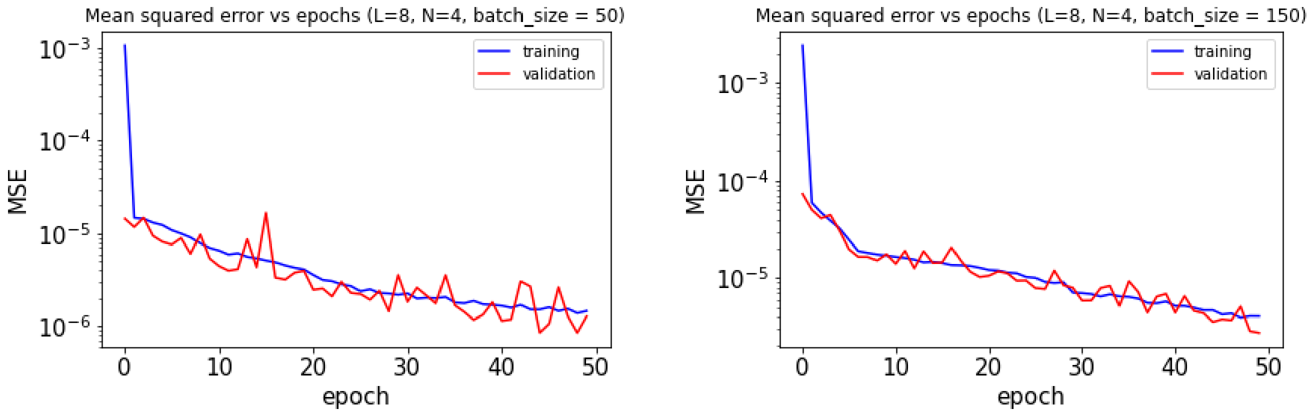
Samples	50,000
Training	40,000
Validation	10,000
Optimiser	ADAM algorithm
Loss Function	Mean-Squared Error
Activation	Rectified Linear Unit
Batch Size	50 / 150
Epochs	50

To make predictions an entirely separate data set of 1000 samples was used.

3 Results

3.1 Neural Network Performance

It was found the network could form a sufficient representation of the functional. Included below are plots showing a decreasing value of the loss vs the number of epochs. The data set for these particular figures was for a ring of 8 atoms with 4 fermions in one dimension, in the canonical ensemble.



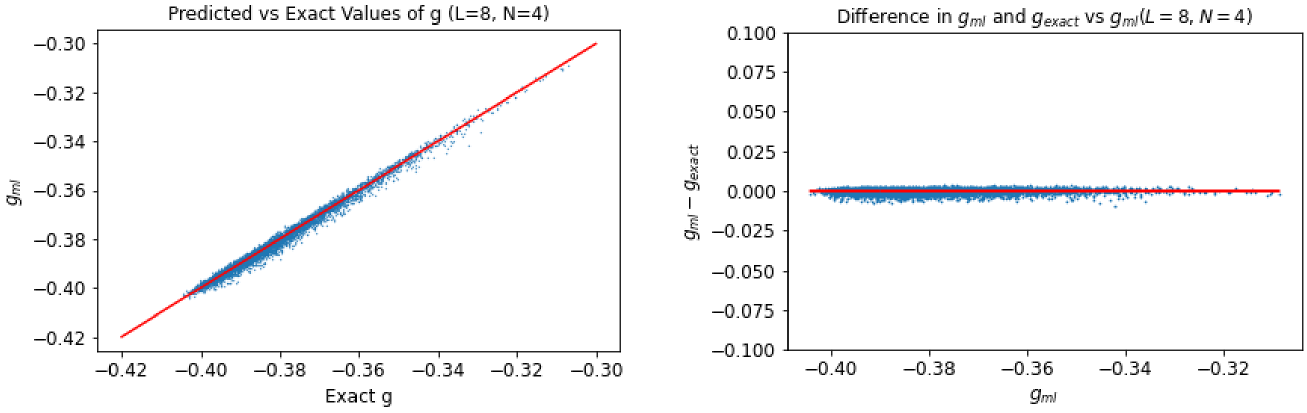
(a) MSE vs Epochs for a batch size of 50.

(b) MSE vs Epochs for a natch size of 150.

Figure 5: One can appreciate from the quick descent of the loss function that the neural network had correctly learned to interpolate the given data. The batch size of 50 led to a somewhat more erratic descent compared to that of the 150 case, however, both cases converged to similar values of the order MSE $\approx 1 * 10^{-5}$

One can clearly see that the network begins to learn how to map the data quite quickly. The network was not run for any more than 50 epochs over the training set to avoid over-fitting, a scenario where the network becomes over optimised to the training set and is not able to generalise well to other randomly generated sets.

In order to gain a better understanding of the predictive power of the network; a plot of the values of the functional predicted for randomly selected densities vs the exact value of the functional obtained through diagonalisation was made. All graphs are taken from a model trained over 50 epochs with a batch size of 50.



(a) XY graph of the exact values of $g[\{n_i\}]$ vs the values predicted by the model. The red diagonal indicates perfect correlation between the two.

(b) A plot of the difference in g_{ml} and g_{exact} for each g_{ml}. A higher accuracy would correspond to all values hovering around the 0 value.

Figure 6: Graphs showing the numerical accuracy of the model in predicting the true value of the universal functional. It is clear to see the predicted values are almost identical to those obtained through exact diagonalisation.

3.2 Satisfying the Hohenberg Kohn theorems

To show the model was able to satisfy the first Hohenberg Kohn theorem (i.e. the existence of the one the one map between $\vec{\epsilon}$ and $\vec{n}^{GS}$) the network was modified to take the on site energies $\vec{\epsilon}$ as the input with the resulting ground state densities as output. Should this bijection exist one would expect the network to easily find the relationship between the two, this was in fact found to be the case. An example of such a prediction with the learning performance of the network is given below.

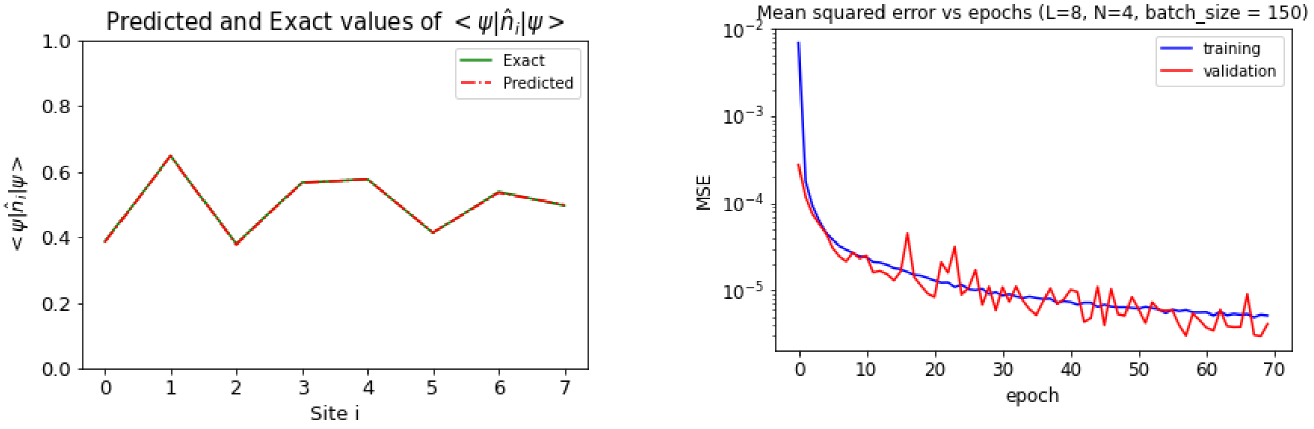


Figure 7: Again the numerical accuracy of the network can be seen from the above graphs. The fact that the network can map directly between $\vec{\epsilon}$ and $\vec{n}_{GS}$ validates theorem one.

As a result of this successful mapping, one could then bypass the need for obtaining ground-state densities via diagonalisation once the network is trained. This would allow for a complete end to end prediction of the value of the functional if, for example, only the on-site energy data is available for a particular problem and not the ground-state densities.

To show that the results were variational, a randomly chosen ground-state $\vec{n}$ with an associated Helmholtz free energy was selected from the new prediction data set and randomly varied. The second theorem predicts that any variations from the ground-state density should increase the value of the predicted Helmholtz free energy. To ensure the conservation of particle number I have only considered variations of the form:

$$n_i \to n_i + \delta$$
$$n_{i+1} \to n_i - \delta$$

Although in principle any variation that conserves particle number should lead to similar results. The following graph shows the change in value of the Helmholtz free energy for 1000 randomly chosen variations on a randomly chosen ground-state density from the prediction set.

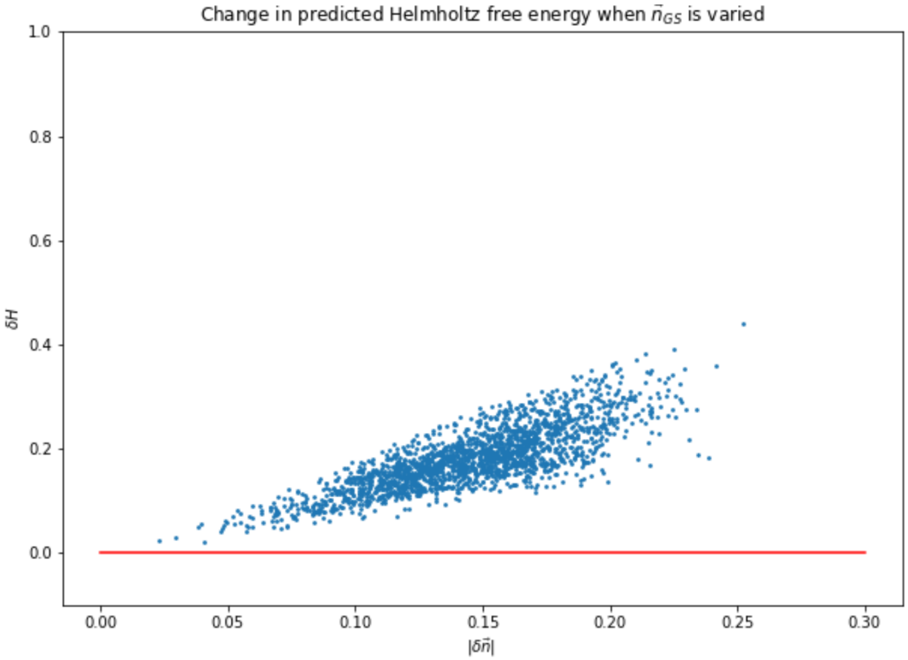


Figure 8: All generated variations of the ground-state density resulted in an increased value of the predicted Helmholtz free energy, showing that the results obtained are indeed variational.

4 Discussion

In conclusion, machine learning was successfully applied to the spinless Hubbard model to:

1. Predict the value of the universal functional from the given ground-state densities

2. Correctly map directly between the on site energies and ground-state densities.

The mappings were compliant with the two Hohenberg Kohn theorems and were were extremely accurate with Mean squared Errors for mapping the functional of the the order $1 * 10^{-5}$. These results could in fact be improved further by considering a larger data set. In fact on a newly generated set of 150,000 samples trained over 250 epochs it was found the MSE fell to the order of $1 * 10^{-7}$, though in practice generating data sets this large is computationally very expensive and takes a long time. One way to improve the size of the data set without performing any other diagonalisations would have been to exploit the symmetries of the model as done by Prof Sanvito et al [11], namely noticing that any cyclic permutations of the on site energies (i.e. $\epsilon_i \rightarrow \epsilon_{i+1}$) in the ring geometry should result in exactly the same values of ground state energy, as should any mirror inversion due to the corresponding symmetry.

An important limitation to note is that the model used was specifically the L=8 problem. There is no easy way to extend the method to larger sites, although again as mentioned in [11] one could potentially consider the functional to be semi-local, and any larger problem could then be considered as being made up of these smaller problems.

For a more definitive proof that the network predictions are variational, one could use a gradient descent scheme to minimise the value of the Helmholtz free energy by varying a randomly chosen density vector $\tilde{\vec{n}}$. This scheme should converge to the value of $\vec{n}$ as given by the model. Work on this method is currently ongoing and is discussed in the appendix.

References

[1] P. Hohenberg and W. Kohn. Inhomogeneous electron gas. *Phys. Rev.*, 136:B864–B871, Nov 1964.

[2] J Hubbard. Electron correlations in narrow energy bands. *Proc. R. Soc. Lond.*, 296, Jan 1967.

[3] M. Michael Denner, Mark H. Fischer, and Titus Neupert. Efficient learning of a one-dimensional density functional theory. *Phys. Rev. Research*, 2:033388, Sep 2020.

[4] Javier Robledo Moreno, Giuseppe Carleo, and Antoine Georges. Deep learning the hohenberg-kohn maps of density functional theory. *Phys. Rev. Lett.*, 125:076402, Aug 2020.

[5] D.J Griffiths. *Introduction to Quantum Mechanics.* Prentice Hall, Upper Saddle River,New Jersey, 1995.

[6] K. Schönhammer, O. Gunnarsson, and R. M. Noack. Density-functional theory on a lattice: Comparison with exact numerical results for a model with strongly correlated electrons. *Phys. Rev. B*, 52:2504–2510, Jul 1995.

[7] W. Kohn and L. J. Sham. Self-consistent equations including exchange and correlation effects. *Phys. Rev.*, 140:A1133–A1138, Nov 1965.

[8] N. David Mermin. Thermal properties of the inhomogeneous electron gas. *Phys. Rev.*, 137:A1441–A1443, Mar 1965.

[9] https://www.ibm.com/cloud/learn/machine-learning.

[10] Francois Chollet et al. Keras, 2015.

[11] James Nelson, Rajarshi Tiwari, and Stefano Sanvito. Machine learning density functional theory for the hubbard model. *Phys. Rev. B*, 99:075132, Feb 2019.

5 Appendix

5.1 Gradient Descent method

As mentioned in the discussion section of the report, a more complete way to prove the variational nature of the theorem would have been to show that a gradient descent scheme applied to Eq.(12) should result in the correct ground state density. I have attempted to do this as follows.

Gradient descent works on finding the minimum point of a given function $f(x)$ by the basic update to the input

$$\tilde{x} = x - \delta \nabla_x f(x), \tag{19}$$

where δ is some small real number. When applied to Eq.(12) this becomes,

$$\tilde{n}_i = n_i - \delta \nabla_{n_i}, H[\{n_i\}] \tag{20}$$

where the derivative of H is given by

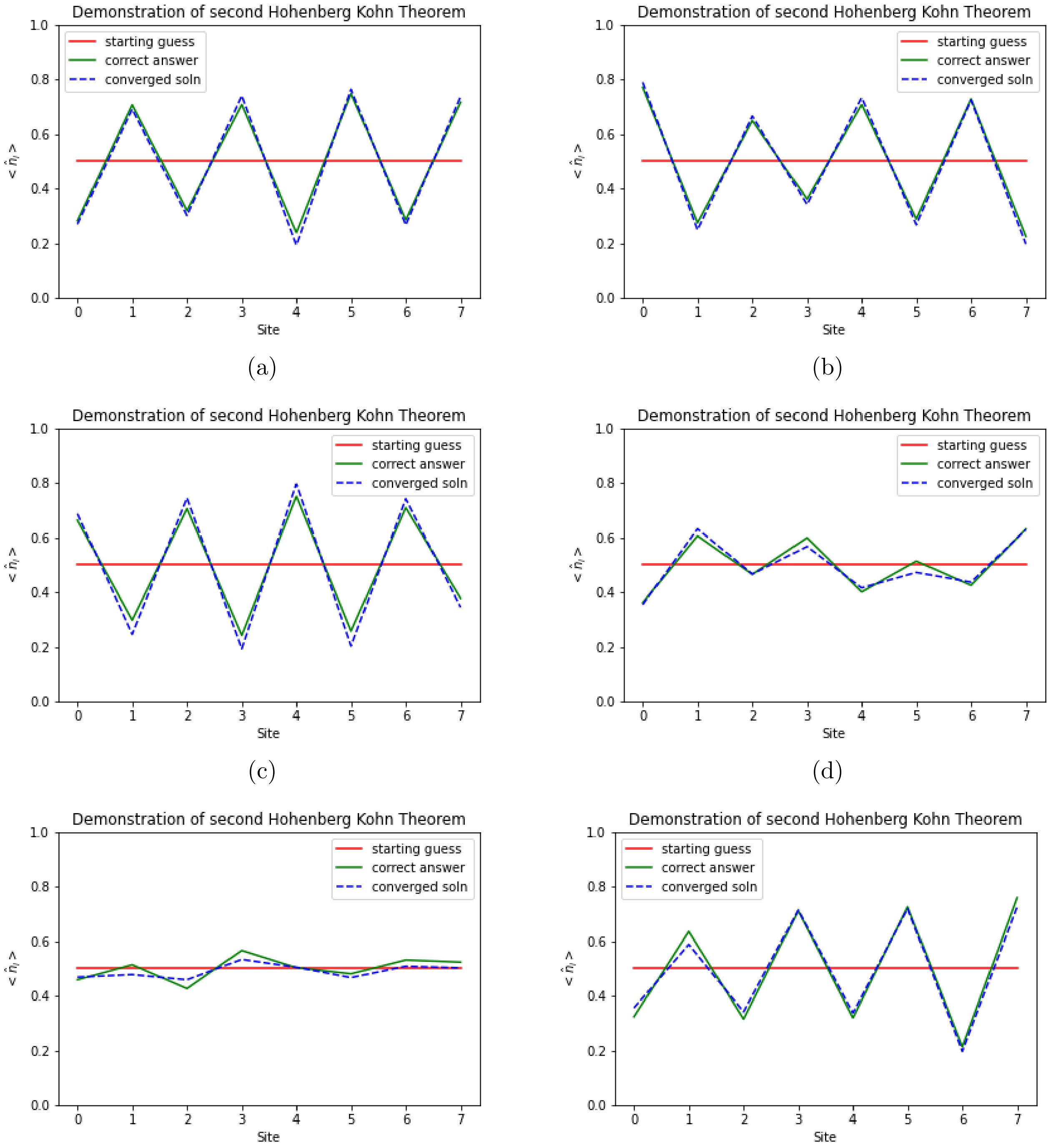
$$\nabla_{n_i} H[\{n_i\}] = \epsilon_i + \frac{G[n_i + \Delta] + G[n_i - \Delta]}{2\Delta} \tag{21}$$

and Δ is also some small real number. The derivative of the functional is approximated by a second order finite difference. When applying the updates care had to be taken that particle number was conserved, therefore after each update the occupation vector was normalised by

$$\vec{n}_i \to \tfrac{4}{k} \vec{n}_i \ ,$$

where $\sum_i n_i = k \neq 4$

The results so far are promising, though I am currently having problems with defining convergence. I am finding ∇H does not go to 0 when the ground-state is reached by the current definition. This is understandable as the derivative will not go to 0 unless both terms in Eq.(21) also go to 0, which is not the case as $\{\epsilon_i\}$ is fixed and non-variable. To get around this I have simply defined the point of convergence as the point at which ∇H ceases to continue decreasing. Some examples of the results being outputted are included on the following page.



(a)

(b)

(c)

(d)

(e)

(f)

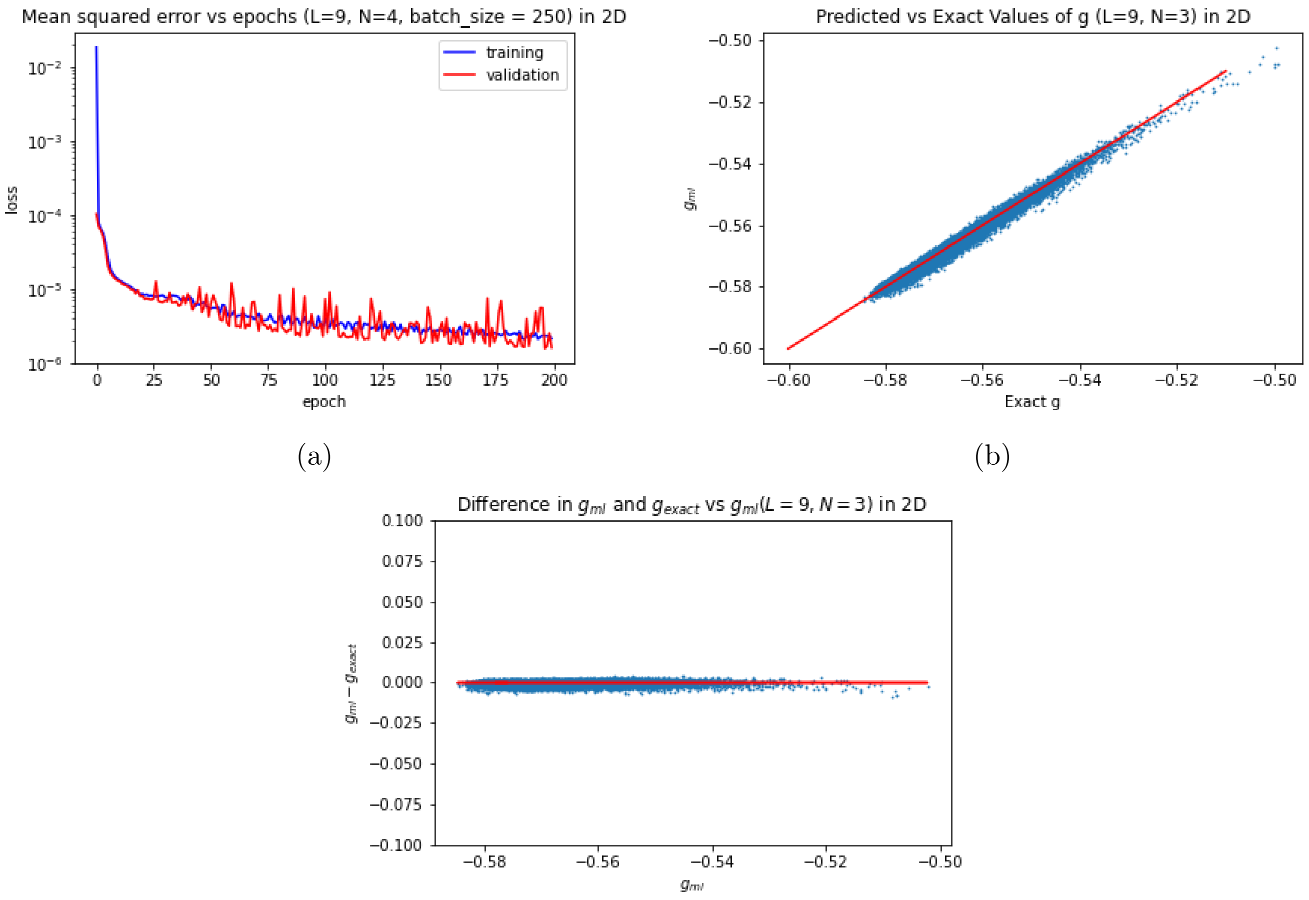
Figure 9: Results obtained by performing a gradient descent scheme on Eq.(12). In each case the homogeneous density was chosen as the starting guess and gradient descent was applied to it.

5.2 2D results (Canonical Ensemble)

Below is the performance results of the network trained on data for the case of the canonical ensemble in 2 dimensions. The network and data information are as below. Again the temperature was taken as $T = 1$. The lattice considered was a simple square lattice with standard nearest-neighbour interactions. The number of particles was reduced to 3 as having 9 sites with 4 particles resulted in large Hamiltonians that took too long to diagonalise.

Samples	100,000
Training	80,000
Validation	20,000
Optimiser	ADAM algorithm
Loss Function	Mean-Squared Error
Activation	Rectified Linear Unit
Batch Size	250
Epochs	200

Variable
sites = 9
filling = 3
pbcs = true
$t = 1$
eps = random
$V = 3$
gridtype = 2D



(a)

(b)

(c)

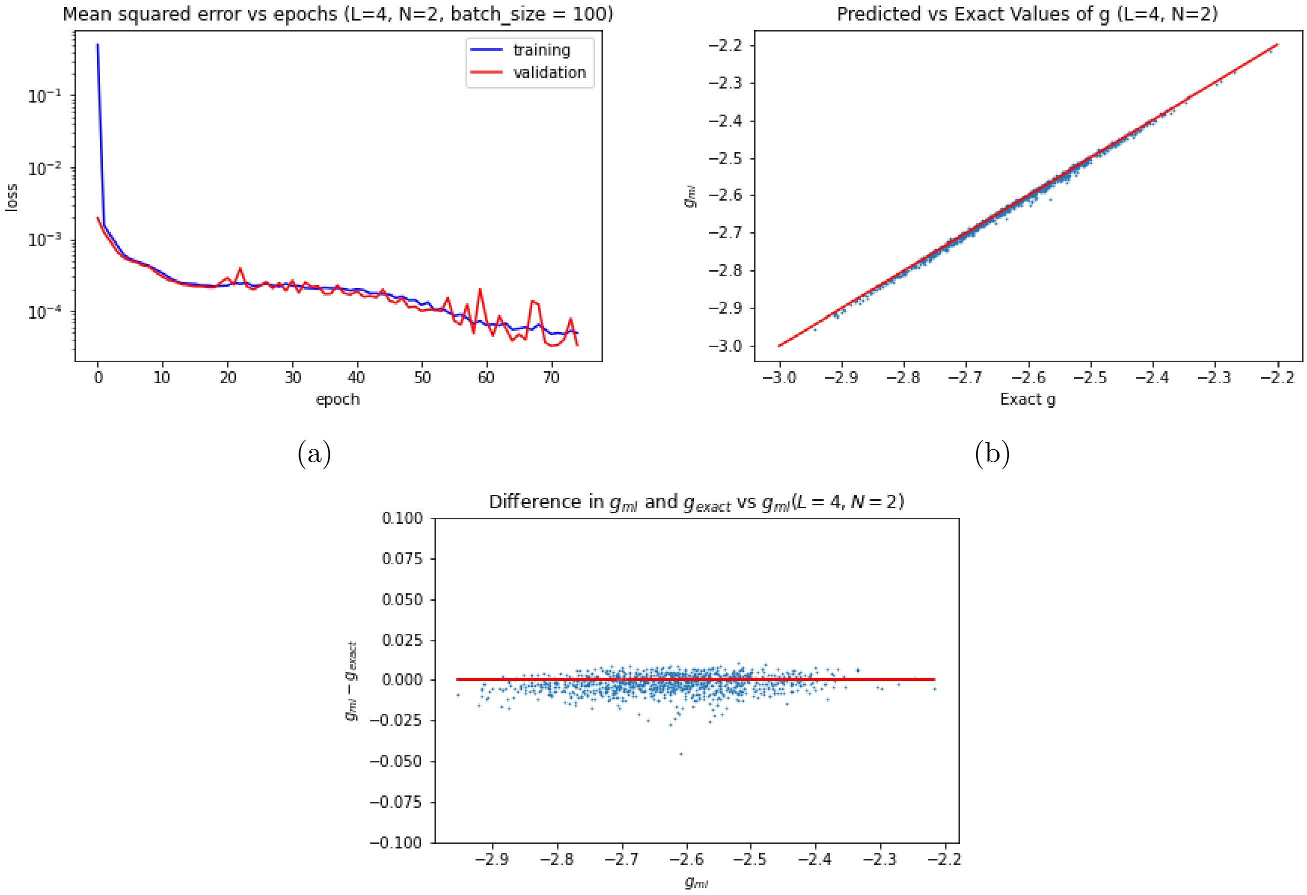
Figure 10: Results obtained for the canonical ensemble in 2D.

5.3 Grand Canonical Results

Data generation was also slower for the canonical ensemble, as all energy eigenstates and all possible particle numbers had to be considered when calculating the grand canonical partition function. Hence only the case of 4 sites in 1D was considered. The model is seen to perform well. Again I include parameters etc for reference.

Samples	50,000
Training	40,000
Validation	10,000
Optimiser	ADAM algorithm
Loss Function	Mean-Squared Error
Activation	Rectified Linear Unit
Batch Size	100
Epochs	75

Variable
sites = 4
filling = N/A
pbcs = true
$t = 1$
eps = random
$V = 3$
gridtype = 1D



(a)

(b)

(c)

Figure 11: Results obtained for the grand canonical ensemble in 1D.