# Machine Learning of Many Body Localization

Philipp Krüger
(Dated: July 16, 2020)

Exact diagonalization was used to find the reduced density matrices of the lowest energy eigenstate of the Heisenberg Model with an additional field in z-direction at low and high disorder strength. The resulting dataset representing extended and localized phases was used to train a neural network. Afterwards, the trained network was applied on intermediate disorder strengths to deduct the critical disorder strength for a phase transition. The phase transition occurred for all system sizes at $W_c = J$.

## I. INTRODUCTION

Firstly, the physical model is introduced. Secondly, the concept of exact diagonalization is briefly presented. As we use the reduced density matrices as the feature for the neural network, we state briefly their derivation.

### A. Physical model

#### 1. Hamiltonian of the Heisenberg model

The hamiltonian of the Heisenberg model is shown in equation 1. In the course of further analysis, we choose $J = 1$ and sample $h$ from a uniform distribution such that $h_i \in [-W, W]$.

$$H = \underbrace{J \sum_i \vec{S}_i \cdot \vec{S}_{i+1}}_{\text{Exchange Energy}} - \underbrace{\sum_i h_i S_i^z}_{\text{Random Field}} \tag{1}$$

#### 2. Expectations for the ground state

The expectation for the ground state is dependent on the ratio of the coupling and the local random field.

For $\frac{W}{J} \ll 1$, we expect an extended, delocalized phase, since the exchange energy dominates over the small external field. Therefore, the system can relax to thermal equilibrium serving as its own heat bath in the limit of large system size $L \to \infty$. Here, the reduced density operator of a finite subsystem converges to the equilibrium thermal distribution for $L \to \infty$.[1]

For $\frac{W}{J} \gg 1$, we can expect a localized phase, since the $h_i$ factors dominate over the exchange energy. The resulting states are expected to be product states of spins "up" or "down", as the external field points in z-direction. Also an infinite system cannot equilibrate itself. The local configurations are set by the initial conditions at all times and are adiabatically connected to the trivial state.[1]

### B. Exact diagonalization

Exact diagonalization (ED) is a numerical technique we can use to solve the Schrödinger Equation $H|\psi\rangle = E|\psi\rangle$ for the eigenvalues $E$ and eigenvectors $|\psi\rangle$. This only works of the Hamiltonian $H$ represents a discrete and finite system. Most quantum many-particle problems lead to a sparse matrix representation of the Hamiltonian, where only a very small fraction of the matrix elements is non-zero.[2] An example for a successful sparse matrix method can be found in the Lanczos algorithm, which is in particulary useful to find ground states of a Hamiltonian.[3]

Introduce concepts: Exact Diagonalization,

### C. Areal (reduced??) Density Matrix

areal Density Matrix: `http://www.thphys.nuim.ie/staff/jvala/Lecture_9.pdf`

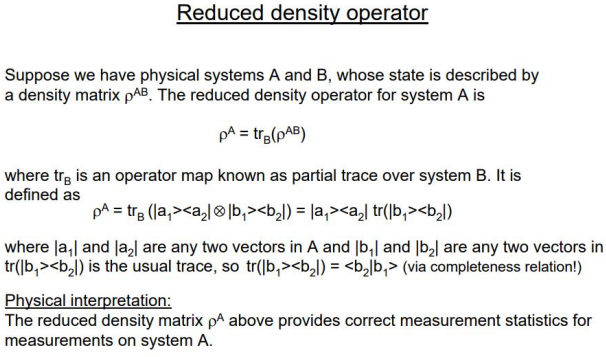



Figure 1: Example of a figure [**?** ].

vs

direct density matrix: (DOI: 10.1103/PhysRevB.99.054208 says Instead of dividing the system into two subsystems A and B to calculate the reduced density matrix of an eigenstate rho a and using the entanglement spectrum as the training data set 34,35, we directly feed the probability density of the eigenstate psi i computed in the spin basis to the machines as the training data set. The reason for doing so is that, although by preprocessing the training data can reduce the dimension and filter out redundant information,

useful information contained in the wavefunction of the entire system can also be lost.)

Conclusion: =¿ areal

### D. Neural Networks and Convolutional Neural Networks

Neural Network, CNN

## II. MATERIALS AND METHODS

### A. Code Protocol

1. Function: Generate random disorder strength using a uniform distribution
2. Function: Generate Hamiltonian from disorder strength and system size
3. Define and understand phase transition from extended phase to localized phase
4. Function: Picks a number M of lowest eigenstates near Energy E = 0
5. Function: Generate density matrix for an eigenstate
6. Function: Visualize density matrices
7. Function: Set up machine learning model that takes density matrices of different W as an input, and predicts whether the state represents an extended or a localized phase.
8. Function: Make predictions for different system sizes L and block sizes n.
9. Function: Plot the predicitons over W.
10. Function: Extract $W_c$ from the data.

Explain Flow with figure
Fig. 2

### B. Machine Learning Models and Error Metrics

Two different approaches:
regression like: Optimizer RMSprop, Loss MAE
classification focused: Optimizer Adam

Loss: BinaryCrossentropy from `https://keras.io/api/losses/probabilistic_losses/#binary_crossentropy-function`: Computes the cross-entropy loss between true labels and predicted labels.

Use this cross-entropy loss when there are only two label classes (assumed to be 0 and 1). For each example, there should be a single floating-point value per prediction.

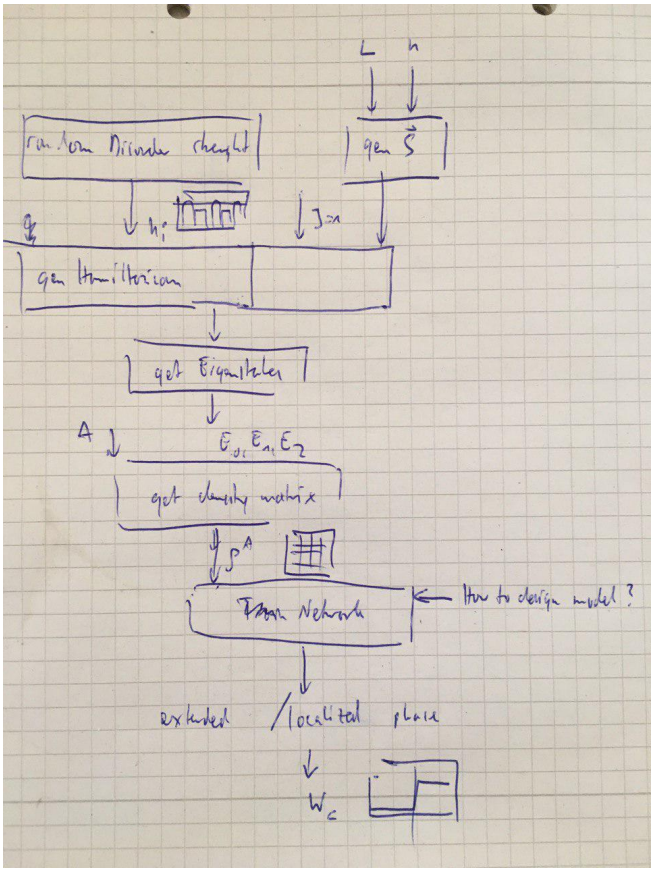



Figure 2: Example of a figure [**?** ].

Optimizer: Stochastic Gradient Descent, Learning rate? Formula? Principle?

Explain metrics and errors and why they are used. Which ml models are used and why?

Use drop out to reduce overfitting. Low sample size techniques:

## III. RESULTS

### A. Generation of density matrix training set

Chosen parameters: $L \in \{10, 11, 12\}$. Repetitions: 500. Measure of variation in the test set??

Plots: What is computationally realizable in 1h concerning time? The training set was sufficiently large enough

We only need M Eigenstates

This is how corresponding density matrices look like

This will be our parameter space for n, L

### B. Prediction of extended vs localized phase

Training and validation scores:

### C. $W_c$ analysis

Now we generate testing set with $W_{max} \in [0, 4]$. We suspect $W_c$ to be at 1 ??We fit a logistic curve and extract $W_c$ as a parameter.

These are our $W_c$ depending on n, L.

## IV. CONCLUSION

$W_c$ depends on n, L (yes/no).

$W_c$ prediction coincides with the expectation (yes/no)

$W_c$ is dependent on these and that effects =¿ scaling analysis? (yes/no)

Citations are numerical[**?** ], some more citations [**? ? ? ? ?** ].


[1] A. Pal and D. A. Huse, Many-body localization phase transition, Phys. Rev. B **82**, 174411 (2010).

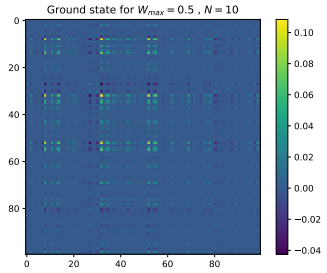
[2] A. Weiße and H. Fehske, Exact diagonalization techniques, in *Computational Many-Particle Physics*, edited by H. Fehske, R. Schneider, and A. Weiße (Springer Berlin Heidelberg, Berlin, Heidelberg, 2008) pp. 529–544.

[3] C. Lanczos, An iteration method for the solution of the eigenvalue problem of linear differential and integral operators, Journal of Research of the National Bureau of Standards **45**, 255 (1950).

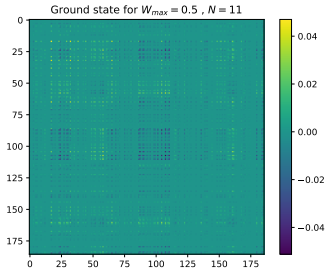

## Appendix A: Code listing

Please copy your code in the appendix.

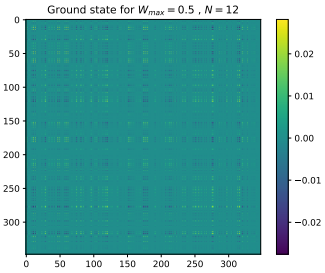
```
"""

Description

"""

import numpy as np

code
```



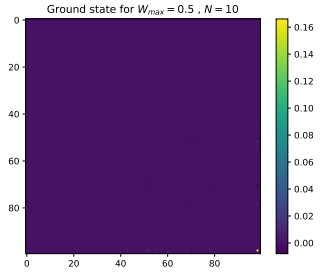

**a)** Ergodic phase L = 10.



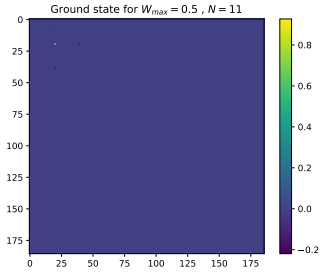

**b)** Ergodic phase L = 11.



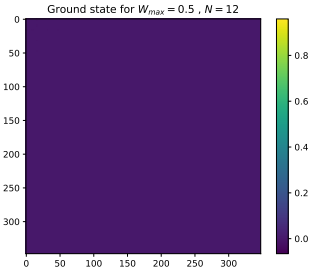

**c)** Ergodic phase L = 12.




**d)** Localized phase L = 10.

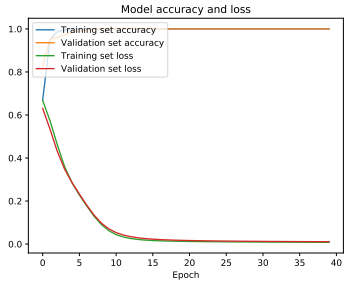



**e)** Localized phase L = 11.

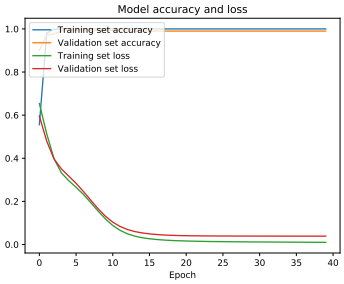



**f)** Localized phase L = 12.

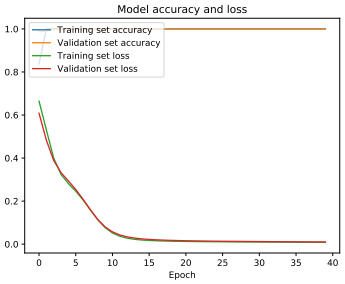
Figure 3: Real part of the density matrix of an ergodic/localized phase for different system sizes L.




**a)** $N = 10$

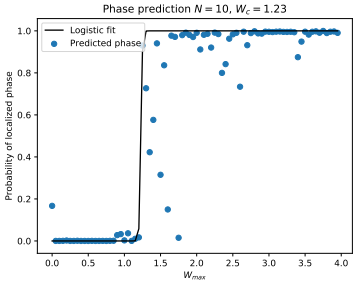



**b)** $N = 11$

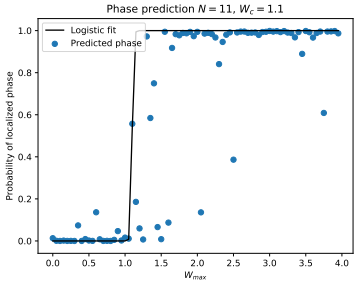



**c)** $N = 12$

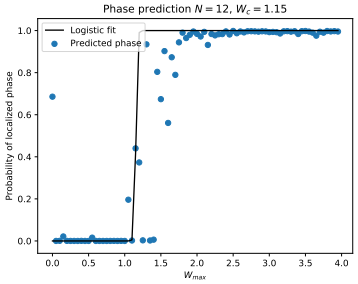
Figure 4: Accuracy and loss of neuronal network plotted over training epochs.




**a)** $N = 10$




**b)** $N = 11$




**c)** $N = 12$

Figure 5: Phase prediction with localized and ergodic phase defined as 1, 0.

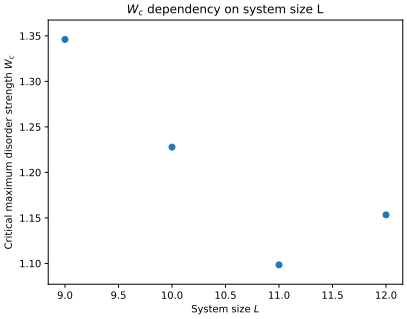



Figure 6

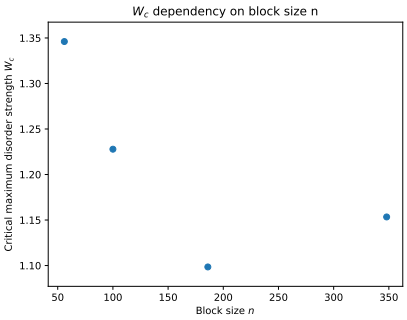



Figure 7