Neural Network Representations of Quantum Many-Body States

Brandon Euverman
Dr. Jean-Sébastien Bernier
Dr. David Casperson

January 2025

Abstract

Quantum many-body systems are challenging to solve due to the exponential growth of the Hilbert space with the number of degrees of freedom. Here, we investigate a method for approximating the wavefunctions describing these systems using neural-network quantum states (NQS), as introduced by Carleo and Troyer [1]. We explore the accuracy of this method, and how to improve it via enforcing physical symmetries.

Contents

Introduction

The challenge of obtaining the wavefunction of a system is one of the most common problems in quantum mechanics. In general, an exponential amount of information is required to express the wavefunction of an arbitrary quantum many-body state. However, in physical systems it is often possible to significantly compress this information below the maximum capacity of the relevant Hilbert space.

There exist many approximation methods for obtaining a tractable form of the wavefunction, such as the density matrix renormalization group (DMRG) algorithm [2]. There are also methods capable of computing their associated expectation values directly, like quantum Monte Carlo (QMC) methods. None of these methods are without their downsides however: QMC suffers from the sign problem [3], and DMRG performs poorly in highly-entangled systems and is most effective when considering quasi-one-dimensional systems [4].

Here we focus on the variational method, of which matrix product states (MPS) are an example [2]. Consider a "trial wavefunction" $\Psi_\alpha(x) = \langle x|\Psi_\alpha\rangle$ where $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_p)$ is the set of p parameters describing our wavefunction. We seek a choice of α that best describes the state we desire. Often, that is the system's lowest energy state (i.e., its ground state).

The challenge of variational methods is choosing a representation of Ψ_α that best describes the true wavefunction coefficients. The value in using a neural network is in its expressivity. It has been shown that neural networks are universal approximators, and with a sufficient number of hidden parameters are capable of representing arbitrary (sufficiently smooth) functions [5]. Further, they do not suffer from the sign problem that plagues Quantum Monte Carlo methods.

One advantage of this representation is that its accuracy can, in principle, be improved simply by increasing the number of hidden variables. Another feature is that this representation invites a symmetry-conserving formulation with relative ease.

Neural Quantum States

Consider a system with N discrete-valued degrees of freedom $S = (\vec{S_1}, \vec{S_2}, \ldots, \vec{S_N})$. For the sake of concreteness we consider spin-$\frac{1}{2}$ particles with each taking on a value from $\sigma^z = \{-1, 1\}$. For further simplicity (but with no loss of generality), we consider the Pauli matrices σ^α in place of the proper spin operators $S^\alpha = \frac{\hbar}{2}\sigma^\alpha$. The wave function is then a mapping from this N-dimensional set S to (exponentially many) complex numbers which fully specify the phase and amplitude of the state.

We provide this mapping by way of a restricted Boltzmann machine (RBM). An RBM is a particularly simple neural network, consisting of one visible layer of N nodes and a single hidden layer of M nodes. They are a variant of Boltzmann

machines, with the constraint that their neurons must form a bipartite graph (refer to fig. 1). This corresponds to the following variational expression for the state:

$$\Psi(S;\mathcal{W}) = \frac{1}{2^M} \sum_{h_i \in \{-1,1\}} \exp\left(\sum_{j=1}^{N} a_j \sigma_j^z + \sum_{i=1}^{M} b_i h_i + \sum_{i=1}^{M} \sum_{j=1}^{N} W_{ij} h_i \sigma_j^z \right), \quad (1)$$

where $h_i \in \{-1, 1\}$ is a set of M hidden spin variables, and the weights $\mathcal{W} = \{a_i, b_j, W_{ij}\}$ fully specify the response of the network to a given input S. In particular, a dictates the visible layer biases, b the hidden layer biases, and W the inter-layer connections. Since this architecture does not include any intra-layer interactions, the hidden layer can be explicitly traced out to read

$$\Psi(S;\mathcal{W}) = \exp\left(\sum_{j=1}^{N} a_j \sigma_j^z \right) \times \prod_{i=1}^{M} \cosh[\theta_i(S)], \quad (2)$$

where the *effective angles* θ_i are given by $\theta_i(S) = b_i + \sum_{j=1}^{N} W_{ij} \sigma_j^z$. It is convenient for computation to also consider the log of the wave function:

$$\ln[\Psi(S;\mathcal{W})] = \sum_{j=1}^{N} a_j \sigma_j^z + \sum_{i=1}^{M} \ln(\cosh[\theta_i(S)]). \quad (3)$$

Our resulting wavefunction is then a mapping from spin configurations to their overlap with the state encoded by the RBM. That is,

$$\Psi(S;\mathcal{W}) = \langle S | \Psi(\mathcal{W}) \rangle . \quad (4)$$

For a given Hamiltonian $\mathcal{H}$, we then want to adapt the weights to provide the best possible representation of its ground state. We do this by unsupervised reinforcement learning through minimizing the expectation value of the energy $E(\mathcal{W}) = \langle \Psi | \mathcal{H} | \Psi \rangle \,/\, \langle \Psi | \Psi \rangle$, with respect to the weights $\mathcal{W}$.

At each iteration k, a Markov chain Monte Carlo (MCMC) sampling of $|\Psi(S;\mathcal{W}_k)|^2$ is obtained for the current set of parameters $\mathcal{W}_k$. Here we apply the Metropolis-Hastings algorithm, which proceeds as follows:

1. Select a spin-configuration $S^{(k)}$ at random.
2. Propose a new sample S' by flipping the ith spin at random.
3. Calculate the acceptance ratio

$$A = \min\left(1, \left| \frac{\Psi(S')}{\Psi(S^{(k)})} \right|^2 \right). \quad (5)$$

The effective angle of this sample can be easily computed by

$$\theta_j(S') = \theta_j(S^{(k)}) - 2W_{ij}\sigma_s^z. \quad (6)$$

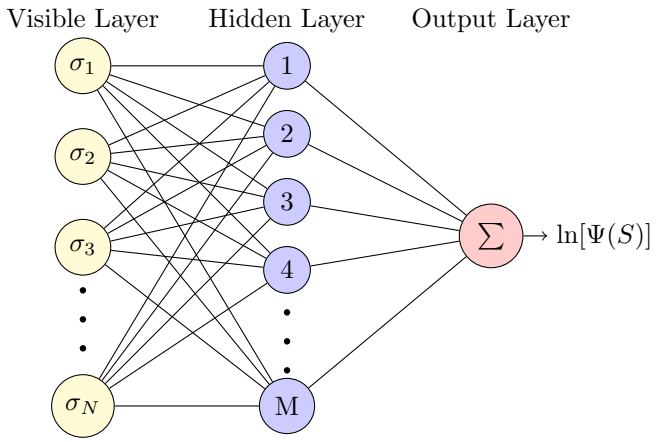

Figure 1: Illustration of the neural network described by a Restricted Boltzmann Machine (RBM).

4. Generate a random number $r \in [0, 1)$.
   - If $r \leq A$, accept the sample by setting $S^{(k+1)} = S'$.
   - If $r > A$, reject the sample by setting $S^{(k+1)} = S^{(k)}$.

In this fashion a "chain" of samples is developed that approaches the target probability distribution. We proceed by attempting to move randomly through the sample space, sometimes accepting the moves and sometimes remaining in place. If we ever consider moving to a point that is more probable than our current point, we always accept the move. If we ever consider moving to a less probable point, we will sometimes reject the move depending on the relative drop in probability. So, we will tend to stay in (and return samples from) regions with a high probability while only occasionally visiting regions with a low probability.

This is an example of rejective sampling and comes with a number of drawbacks. Firstly, the samples are autocorrelated. Any small set of samples from the chain will be correlated and not properly reflect the desired distribution. This means that the effective sample size may be much smaller than the actual length of the chain. Further, although the Markov chain will eventually follow the target distribution, the initial portion of the chain may not. This necessitates a burn-in period, in which some number of initial samples are discarded. We follow the practice of discarding the first tenth of the chain.

Suppose that a set of configurations $\{S_i\}$ with $i = 1, \ldots, N$ have been sampled from the associated probability distribution of our variational state $\Psi(S; \mathcal{W})$. Then the energy of this state can be given as

$$E(\mathcal{W}) = \frac{\langle \Psi | \mathcal{H} | \Psi \rangle}{\langle \Psi | \Psi \rangle} \approx \frac{1}{N} \sum_{i=1}^{N} \frac{\langle S_i | \mathcal{H} | \Psi \rangle}{\langle S_i | \Psi \rangle} = \frac{1}{N} \sum_{i=1}^{N} E_{loc}(S_i), \tag{7}$$

where $E_{loc} \equiv \frac{\langle S_i|\mathcal{H}|\Psi\rangle}{\langle S_i|\Psi\rangle}$ is called the local energy.

Indeed, this holds for any operator with its corresponding "local" quantity. This serves then as a method for approximating the expectation of any operator on our state, provided we have such a set of samples $\{S_i\}$.

With this sampling, stochastic estimates of the energy gradient are computed, with which a new set of weights $\mathcal{W}_{k+1}$ are proposed according to the stochastic reconfiguration (SR) method. See the appendix for a brief overview.

Implementation

This project was implemented in Python using the JAX library. JAX was designed by Google for high-performance numerical computation and machine learning. Its primary feature is Just In Time (JIT) compilation of python code to XLA (Accelerated Linear Algebra), leading to much better performance. This code can then be executed on GPUs (or TPUs) for even greater performance, but this project was not written with this in mind. Code was executed on an AMD 7543 2.8GHz processor.

The variational wavefunction as described above does not necessarily possess the symmetries that an eigenstate of the Hamiltonian should, even after training. In principle, since our RBM has the property of being a universal approximator, it is possible to achieve these symmetries with some large enough number of hidden nodes. Doing so is inefficient however, and it is possible to impose some symmetry operator on our state to do this much more effectively. Such a procedure commonly improves the accuracy of our wavefunction.

Consider a symmetry group of order R defined by a set of linear transformations $\{\hat{T}_r\}$. We can enforce our wavefunction to be invariant under the action of this group by defining a symmetrized wavefunction

$$\tilde{\Psi}(S;\mathcal{W}) = \frac{1}{2^M} \sum_{h_{i,r}\in\{-1,1\}} \exp\left(\sum_{r=1}^{R} \Lambda(S;\mathcal{W},h_{i,r})\right), \tag{8}$$

where we have defined

$$\Lambda(S;\mathcal{W},h_{i,r}) \equiv \sum_{j=1}^{N} a_j \sigma_{j,r}^z + \sum_{i=1}^{M} b_i h_{i,r} + \sum_{i=1}^{M}\sum_{j=1}^{N} W_{ij} h_{i,r} \sigma_{j,r}^z. \tag{9}$$

Here we understand $\sigma_{j,r}^z$ to be the spin of the j-th lattice site under the transformation $\hat{T}_r$. We can explicitly trace out eq. 8 as we did eq. 1 to provide

$$\tilde{\Psi}(S;\mathcal{W}) = \exp\left(\sum_{r=1}^{R}\sum_{j=1}^{N} a_j \sigma_{j,r}^z\right) \times \prod_{r=1}^{R}\prod_{i=1}^{M} \cosh(\theta_i[\hat{T}_r(S)]), \tag{10}$$

where $\hat{T}_r(S) = (\langle S|\,\hat{T}_r)$, and we can again consider the log of the wavefunction

to give

$$\ln[\tilde{\Psi}(S;\mathcal{W})] = \sum_{r=1}^{R}\sum_{j=1}^{N} a_j \sigma_{j,r}^{z} + \sum_{r=1}^{R}\sum_{i=1}^{M} \ln[\cosh(\theta_i[\hat{T}_r(S)])] \tag{11}$$

$$= \sum_{r=1}^{R} \ln[\Psi(\hat{T}_r(S);\mathcal{W})]. \tag{12}$$

In the specific case of translational symmetry, our operators have the effect of translating sites on the lattice and look like permutations on the sites.

It is easy to see that there are N such translations on any grid lattice: each transformation can be uniquely identified by where a particular site finds itself translated to, of which there are N.

Results

To validate our model we apply it to finding the ground state of the transverse-field Ising model, given by

$$\mathcal{H} = -J \sum_{\langle i,j \rangle} \sigma_i^z \sigma_j^z - h_x \sum_i \sigma_i^x \tag{13}$$

where $\langle i,j \rangle$ refers to a sum only over neighbouring sites, and σ^x and σ^z are Pauli matrices. While an arbitrary graph can be made to be a spin-lattice, a

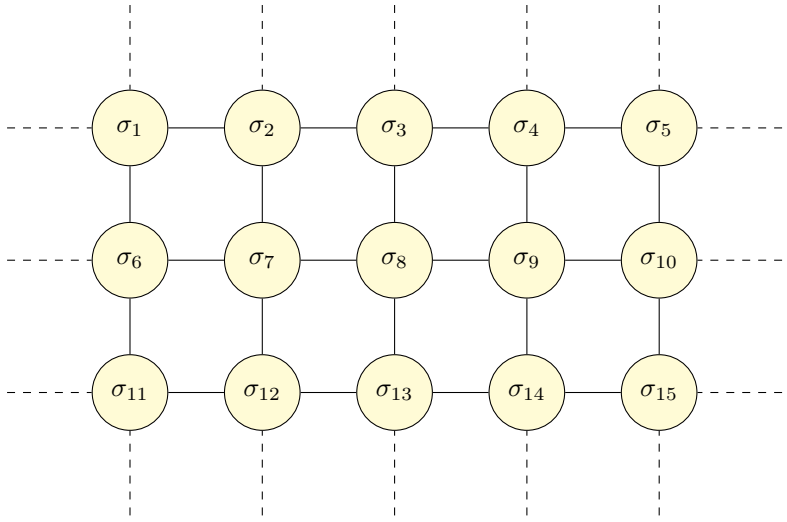

Figure 2: A sample grid spin-lattice, dashed lines representing additional potential sites.

grid is the most common shape and is shown in fig. 2. This model is of interest due to it being one of the simplest models that undergoes a phase transition, from ferromagnetic to polarized at $J = h_x$, $J > 0$. This so-called critical point is also the most difficult place at which to analyze the system.

As an example, here we trained on a 25-site chain with periodic boundary conditions, using $J = h_x = 1$ and a hidden layer of size $M = 90$. The results of the training process are shown in fig. 3, where the energy error at each step is computed against a contemporary Matrix Product States (MPS) method.

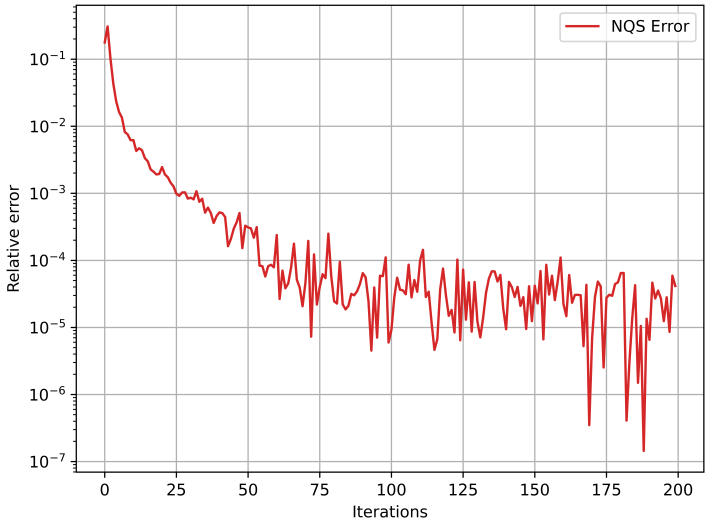

Figure 3: Relative error $\epsilon = \left|\frac{E_{NQS}-E_{MPS}}{E_{MPS}}\right|$ between the approximate energy obtained via NQS and the quasi-exact groundstate energy obtained from MPS.

It is also interesting to consider the computation of spin-spin correlations in this system. Consider some direction α in spin space. Then the spin-spin correlations between two sites i and j along this axis is simply $\langle \sigma_i^\alpha \sigma_j^\alpha \rangle$. In some systems, this can be used to verify that they possess the expected symmetries. Consider a spin chain of odd length and periodic boundary conditions and pick integers i, j, and d. Then, since such a system possesses both translational and reflectional symmetry, for any eigenstate of $\mathcal{H}$, we expect

$$\langle \sigma_i^\alpha \, \sigma_{i+j}^\alpha \rangle = \langle \sigma_i^\alpha \, \sigma_{i-j}^\alpha \rangle \,, \tag{14}$$

$$\langle \sigma_i^\alpha \sigma_{i+j}^\alpha \rangle = \langle \sigma_{i+d}^\alpha \sigma_{i+d+j}^\alpha \rangle \,, \tag{15}$$

respectively. The reflection symmetry of our state is shown in fig. 4.

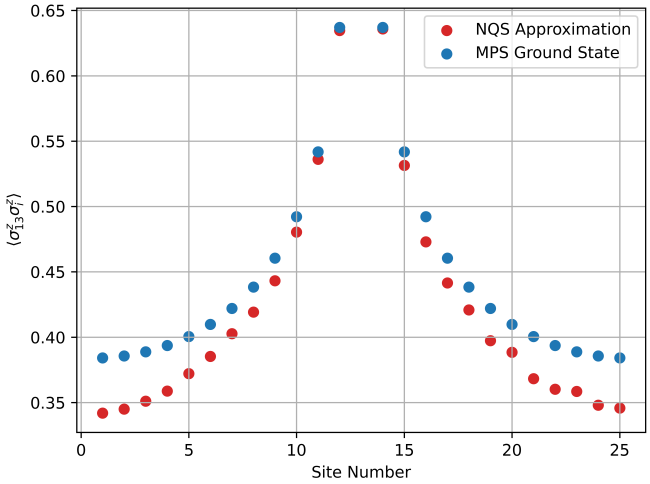

Figure 4: Spin-spin correlations $\langle \sigma^z_{13}\sigma^z_i \rangle$. We expect such a graph to be symmetric, which is demonstrated by the true ground state (blue). The result of the NQS is shown in red. Note that there is a discrepancy between the two that has not yet been accounted for. Future work will include applying a Jordan-Wigner transformation to this system, allowing us to obtain an exact solution.

Conclusion

Neural-network quantum states are a relatively recent advancement that have been shown to be very promising and lacking many of the downsides that come with other contemporary variational techniques. We have demonstrated here very accurate results for the ground states of spin models, despite the relative simplicity of the restricted Boltzmann machines used here. We have shown further that it is possible to enforce various physical symmetries on the state to achieve better training results.

One potential advancement to be made is with the implementation of the minSR algorithm [6], which may massively reduce training costs while maintaining numerical precision. Possible architectures have expanded to include (among others) Recursive Neural Networks [7] and Convolutional Neural Networks [8], and have been applied to excited states and finite temperature states, to name a few [8].

References

[1] Giuseppe Carleo and Matthias Troyer. "Solving the quantum many-body problem with artificial neural networks". In: *Science* 355.6325 (Feb. 2017), pp. 602–606. ISSN: 1095-9203. DOI: 10.1126/science.aag2302. URL: http://dx.doi.org/10.1126/science.aag2302.

[2] Ulrich Schollwöck. "The density-matrix renormalization group in the age of matrix product states". In: *Annals of Physics* 326.1 (Jan. 2011), pp. 96–192. ISSN: 0003-4916. DOI: 10.1016/j.aop.2010.09.012. URL: http://dx.doi.org/10.1016/j.aop.2010.09.012.

[3] Gaopei Pan and Zi Yang Meng. "The sign problem in quantum Monte Carlo simulations". In: *Encyclopedia of Condensed Matter Physics*. Elsevier, 2024, pp. 879–893. ISBN: 9780323914086. DOI: 10.1016/b978-0-323-90800-9.00095-0. URL: http://dx.doi.org/10.1016/B978-0-323-90800-9.00095-0.

[4] Samuel Moukouri and Eytan Grosfeld. "Behavior of entanglement entropy in interacting quasi-one-dimensional systems and its consequences for their efficient numerical study". In: *Phys. Rev. B* 90 (3 July 2014), p. 035112. DOI: 10.1103/PhysRevB.90.035112. URL: https://link.aps.org/doi/10.1103/PhysRevB.90.035112.

[5] Kurt Hornik. "Approximation capabilities of multilayer feedforward networks". In: *Neural Networks* 4.2 (1991), pp. 251–257. ISSN: 0893-6080. DOI: https://doi.org/10.1016/0893-6080(91)90009-T. URL: https://www.sciencedirect.com/science/article/pii/089360809190009T.

[6] Ao Chen and Markus Heyl. "Empowering deep neural quantum states through efficient optimization". In: *Nature Physics* 20.9 (July 2024), pp. 1476–1481. ISSN: 1745-2481. DOI: 10.1038/s41567-024-02566-1. URL: http://dx.doi.org/10.1038/s41567-024-02566-1.

[7] Mohamed Hibat-Allah et al. "Recurrent neural network wave functions". In: *Phys. Rev. Res.* 2 (2 June 2020), p. 023358. DOI: 10.1103/PhysRevResearch.2.023358. URL: https://link.aps.org/doi/10.1103/PhysRevResearch.2.023358.

[8] Hannah Lange et al. *From Architectures to Applications: A Review of Neural Quantum States*. 2024. arXiv: 2402.09402 [cond-mat.dis-nn]. URL: https://arxiv.org/abs/2402.09402.

Quantum Mechanics Primer

Quantum mechanics is done formally in a complex Hilbert space, which is a vector space equipped with an inner product. We restrict ourselves here to considering finite-dimensional Hilbert spaces, which are in all cases isometric to $\mathbb{C}^n$. We make use of Dirac notation to denote elements of these spaces. Let $\mathcal{H}$ be an n-dimensional Hilbert space. We denote some element $\psi \in \mathcal{H}$ by $|\psi\rangle$, and an element ϕ in the dual space $\mathcal{H}^\dagger$ by $\langle\phi|$. This leads to a natural understanding of the inner product as

$$\langle\phi|\psi\rangle = \phi_1^*\psi_1 + \phi_2^*\psi_2 + \cdots + \phi_n^*\psi_n. \tag{16}$$

where we have taken $\langle\phi| = \begin{bmatrix}\phi_1^* & \phi_2^* & \cdots & \phi_n^*\end{bmatrix}$ and $|\psi\rangle = \begin{bmatrix}\psi_1 \\ \psi_2 \\ \vdots \\ \psi_n\end{bmatrix}$.

Now let $\{|\phi_i\rangle\}_{i=1}^n$ be an orthonormal basis of $\mathcal{H}$. A wavefunction is some element $|\Psi\rangle$ of $\mathcal{H}$ and is a mathematical description of a quantum state. To each element of the basis, $|\Psi\rangle$ assigns a complex number c_i and we can decompose our wavefunction in this basis using the completeness relation:

$$|\Psi\rangle = \left(\sum_{i=1}^n |\phi_i\rangle\langle\phi_i|\right)|\Psi\rangle = \sum_{i=1}^n \langle\phi_i|\Psi\rangle\,|\phi_i\rangle = \sum_{i=1}^n c_i\,|\phi_i\rangle \tag{17}$$

where $c_i = \langle\phi_i|\Psi\rangle$ is exactly the overlap of our wavefunction with the state $|\phi_i\rangle$. Further, according to the Born interpretation, $|c_i|^2$ is the probability to find our quantum state in the basis state $|\phi_i\rangle$ upon taking the relevant measurement. In accordance with this interpretation, we also require

$$\langle\Psi|\Psi\rangle = 1. \tag{18}$$

Observables are physical quantities that can be measured. In quantum mechanics, observables are Hermitian operators and possible results of a measurement are the eigenvalues of this operator. The expectation value of an operator $\hat{O}$ is obtained simply by sandwiching the operator in the inner product:

$$\langle\hat{O}\rangle = \langle\Psi|\hat{O}|\Psi\rangle\,. \tag{19}$$

If we were to decompose our wavefunction into the eigenbasis of the operator, we can clearly see this as a weighted average of the possible outcomes of measurement on our superposition of states.

The Hamiltonian of a system is the operator corresponding to the total energy of that system, including both kinetic and potential energy. Its spectrum is the set of possible outcomes one can obtain from measuring the system's total energy. When we are searching for the ground state of a system, we are looking for the eigenvector of the Hamiltonian corresponding to its least eigenvalue.

Consider a spin-1/2 particle. Due to the uncertainty principle, non-commuting observables cannot be measured simultaneously as there does not exist a basis of simultaneous eigenstates. Since the components of angular momentum do not commute, it is impossible to know the exact direction of the spin of this particle. It is convention to consider the projection of the spin along the z-axis, allowing us to choose our basis as $\{|\uparrow\rangle, |\downarrow\rangle\}$. We can draw an isomorphism easily to $\mathbb{C}^2$ by

$$|\uparrow\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad \text{and} \quad |\downarrow\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \tag{20}$$

If we are interested in the spin of our state in any particular axis, we can do so by taking the expectation of the corresponding spin operator. Up to a constant, these are the Pauli matrices. In our chosen basis, these can be given as:

$$\sigma^x = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$
$$\sigma^y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$$
$$\sigma^z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

If we wanted to introduce another particle to this system, we could do so by considering the new space $\mathbb{C}^2 \otimes \mathbb{C}^2$. Indeed, in general we can consider N sites by looking at $\bigotimes_{i=1}^{N} \mathbb{C}^2 \cong \mathbb{C}^{2N}$.

Stochastic Reconfiguration

We detail with justification the application of the stochastic reconfiguration (SR) method to our RBM.

We first introduce the variational derivatives with respect to the k-th network parameter,

$$\mathcal{O}_k(S) = \frac{1}{\Psi(S)} \partial_{\mathcal{W}_k} \Psi(S). \tag{21}$$

For the weights in our RBM, these derivatives read

$$\frac{1}{\Psi(S)} \partial_{a_i} \Psi(S) = \sigma_i^z,$$
$$\frac{1}{\Psi(S)} \partial_{b_j} \Psi(S) = \tanh(\theta_j(S)),$$
$$\frac{1}{\Psi(S)} \partial_{W_{ij}} \Psi(S) = \sigma_i^z \tanh(\theta_j(S)).$$

The SR updates to the weights at the p-th iteration are then

$$\mathcal{W}(p+1) = \mathcal{W}(p) + \gamma S^{-1}(p) F(p), \tag{22}$$

where we have the (positive-semidefinite) covariance matrix

$$S_{kk'}(p) = \langle (\mathcal{O}_{k'} - \langle \mathcal{O}_{k'} \rangle)(\mathcal{O}_k - \langle \mathcal{O}_k \rangle)^* \rangle, \tag{23}$$

and the forces

$$F_k(p) = \langle (E_{loc} - \langle E_{loc} \rangle)(\mathcal{O}_k - \langle \mathcal{O}_k \rangle)^* \rangle, \tag{24}$$

with

$$E_{loc} \equiv \frac{\langle S_i | \mathcal{H} | \Psi \rangle}{\langle S_i | \Psi \rangle} \tag{25}$$