Solving Many-Body Quantum Mechanics with Neural Networks

Emily J. Davis, Kevin A. Fischer, and Alexander T. Hristov
{emilyjd, kevinf, hristov}@stanford.edu

Motivation

- Neural Networks have been successful representing complex nonlinear functions for tasks like image recognition and speech processing. We use such models for representing many-body quantum states.
- Quantum mechanics underlies numerous open challenges in materials science and quantum engineering; simulating these systems would enable improved material synthesis or device design
- The number of parameters necessary to represent a quantum state scales exponentially with system size, precluding many simple numerical techniques. We use neural networks to efficiently reduce the number of parameters.
- Quantum states can be represented by complex valued functions over the basis vectors of a Hilbert space. The square modulus gives probability distribution over the basis states.

Technical Approach

- Given a system with known Hamiltonian H (operator that specifies the energy and dynamics of the state of the system), we seek the ground state, which is the probability distribution $P_\theta(x) = \frac{|\psi_0(x;\theta)|^2}{\langle\psi_0|\psi_0\rangle}$ that minimizes the expectation value of the Hamiltonian. Formally:

$$\mathcal{E}[\psi] = E\left[\sum_{x'} H_{xx'}\frac{\psi(x')}{\psi(x)}\right] = E[\mathcal{E}_{loc}(x)]$$

$$\psi_0 = \mathrm{argmin}_\psi\, \mathcal{E}[\psi]$$

- For any variational wave function $\psi_0(x;\theta)$ (neural network or otherwise), training can be achieved through the stochastic reconfiguration algorithm of Sorella, et al.

$$\mathrm{d}\psi_j(x;\theta) \equiv \frac{1}{\psi(x;\theta)}\frac{\partial\psi(x;\theta)}{\partial\theta_j}$$

$$S_{ij}^{-1} = E\left[\mathrm{d}\psi_i^*(x;\theta)\mathrm{d}\psi_j(x;\theta)\right] - E[\mathrm{d}\psi_i^*(x;\theta)]E[\mathrm{d}\psi_j(x;\theta)]$$

$$\frac{\partial\mathcal{E}[\psi(x;\theta)]}{\partial\theta_j} \approx E\left[\mathcal{E}_{loc}(x)\,\mathrm{d}\psi_j(x;\theta)\right] - E[\mathcal{E}_{loc}(x)]E[\mathrm{d}\psi_j(x;\theta)]$$

$$\theta_i \leftarrow \theta_i - \alpha\sum_j S_{ij}\frac{\partial\mathcal{E}[\psi]}{\partial\theta_j}$$

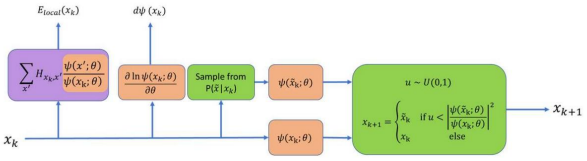
Model and Training

- Our model is a restricted Boltzman machine, which has a probability distribution over both visible and latent variables. Integrating over latent variables yields the distribution of visible variables.

$$\psi(x,h;\theta) = \exp(-a^\intercal x - b^\intercal h - h^\intercal W x)$$

$$\psi(x;\theta) = \sum_h \psi(x,h;\theta) = \exp(-a^\intercal x)\prod_{i=1}^{M} 2\cosh\left(b_i + \sum_{j=1}^{N} W_{ij}x_j\right)$$

- The sampling process can be understood as a stochastic sequential model, with iterative step given by the following diagram.

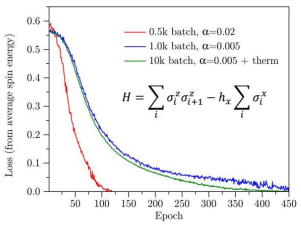


Sampling

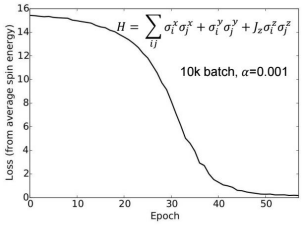
- The algorithm stochastically samples the state space using a Metropolis-Hastings algorithm with the probability of transitioning to a new state x' from a previous state x is given by $\left|\frac{\psi(x';\theta)}{\psi(x;\theta)}\right|^2$.
- The process can be tuned by adjusting both thermalization time and the length of the sampling process. Both must be set above the estimated autocorrelation time of this Markov Chain technique.

Training results

1D Ising Model with $h_x = 0.5$

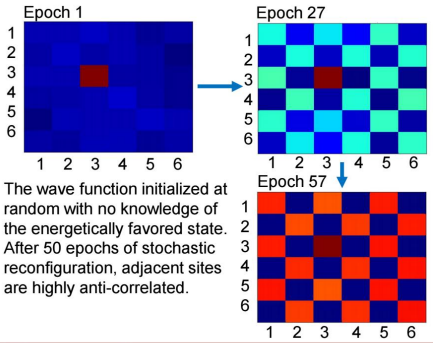


2D Heisenberg Model with J_z =10



Correlations emerge from optimizing the wave function

Sample results for an anisotropic Heisenberg model ($J_z \gg J_x, J_y$). Plot show the spin-spin correlators $\langle x^{(i)}x^{(j)}\rangle$ of $P_\theta(x)$.



The wave function initialized at random with no knowledge of the energetically favored state. After 50 epochs of stochastic reconfiguration, adjacent sites are highly anti-correlated.

Discussion

- Stochastic reconfiguration of a restricted Boltzmann machine successfully optimizes the wave-function represented by the network.
- Stochastic Reconfiguration algorithm is exponentially sensitive to the learning rate, making this learning algorithm very hard to optimize. Frequently overflows during training for improper configuration.
- Thermalization of the Metropolis-Hastings algorithm plays a minimal role in controlling gradient descent.

Future Directions

- Infrastructure we developed works for arbitrary networks.
- These techniques would benefit from implementing weight sharing for systems with symmetries, particularly convolutional architectures.

References

- G. Carleo and M. Troyer, Science **355**, 602 (2017)
- S. Sorella, M. Casula, and D. Rocca, The Journal of Chemical Physics **127**, 014105 (2007)