Machine Learning Accelerated DQMC

The Hubbard Model

THE UNIVERSITY OF TENNESSEE
KNOXVILLE

DQMC – Beyond the Ising Model

- Typical Hamiltonians contain non-commuting terms
  - Path Integral Formalism in Imaginary Time

$$\langle \Psi_L \,|\, e^{-\beta H} \,|\, \Psi_R \rangle = \sum_{\Psi_n} \langle \Psi_L \,|\, e^{-\Delta\tau H} \,|\, \Psi_{N-1} \rangle \cdots \langle \Psi_2 \,|\, e^{-\Delta\tau H} \,|\, \Psi_1 \rangle \langle \Psi_1 \,|\, e^{-\Delta\tau H} \,|\, \Psi_R \rangle$$

- Hamiltonian Undergoes Trotter Decomposition

$$e^{-\Delta\tau H} = e^{-\Delta\tau H_1} e^{-\Delta\tau H_2} \cdots e^{-\Delta\tau H_n} + \mathcal{O}((\Delta\tau)^2)$$

Hubbard Model

$$H^{Hub} = -t \sum_{\langle ij \rangle, \sigma} (a^\dagger_{i\sigma} a_{j\sigma}) + U \sum_i \left(n_{i\uparrow} - \frac{1}{2} \right) \left(n_{i\downarrow} - \frac{1}{2} \right)$$

- Trotter Decomposition and Hubbard-Stratonovich Transformation

$$e^{-\Delta\tau H} \approx e^{-\Delta\tau H_1} e^{-\Delta\tau U \left(n_\uparrow - \frac{1}{2}\right)\left(n_\downarrow - \frac{1}{2}\right)}$$

$$e^{-\Delta\tau U \left(n_\uparrow - \frac{1}{2}\right)\left(n_\downarrow - \frac{1}{2}\right)} = \frac{1}{2} e^{-\frac{1}{4}\Delta\tau U} \sum_{h=\pm} e^{\alpha h (n_\uparrow - n_\downarrow)}$$

DQMC – Calculating Next Step (simplified)

- Propagator: $B_{ij}(\tau_2, \tau_1) = \left\langle 0 \middle| c_i \left[\mathcal{T} \exp\left(- \int_{\tau_1}^{\tau_2} H(\tau) d\tau \right) \right] c_j^\dagger \middle| 0 \right\rangle$

- Green's Functions: $G^\sigma(\tau, \tau) = [I + B^\sigma(\tau, 0) B^\sigma(\beta, \tau)]^{-1}$

$$G^\sigma(\tau', \tau') = B^\sigma(\tau', \tau) G^\sigma(\tau, \tau) B^\sigma(\tau', \tau)^{-1}$$

- Accept/Reject Ratio:

$$\mathcal{R}^\sigma \equiv \frac{w^\sigma_{new}}{w^\sigma_{old}} = \det[I + \Delta^\sigma(i, \tau)(I - G^\sigma(\tau, \tau))] \qquad \mathcal{R} = \mathcal{R}^\uparrow \mathcal{R}^\downarrow$$

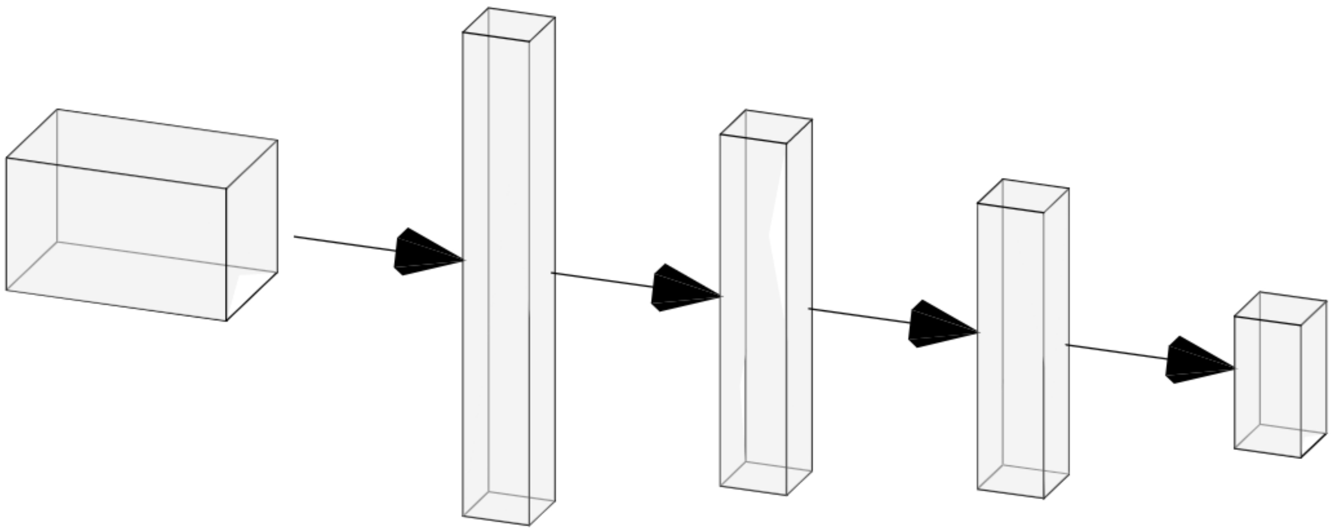
Can we throw away large share of rejects using ML?

- Hubbard DQMC derived Green's functions
  - Half-filled Hubbard Model ($U=6$, $t=1$, $\beta=1$, $\Delta\tau=0.1$, 2x2 Lattice)
- Input Data:
  - 2,000,000 sets of 2x2x10 Green's Functions
- Estimate Output
  - 2x2 New Green' functions

Neural Net Architecture

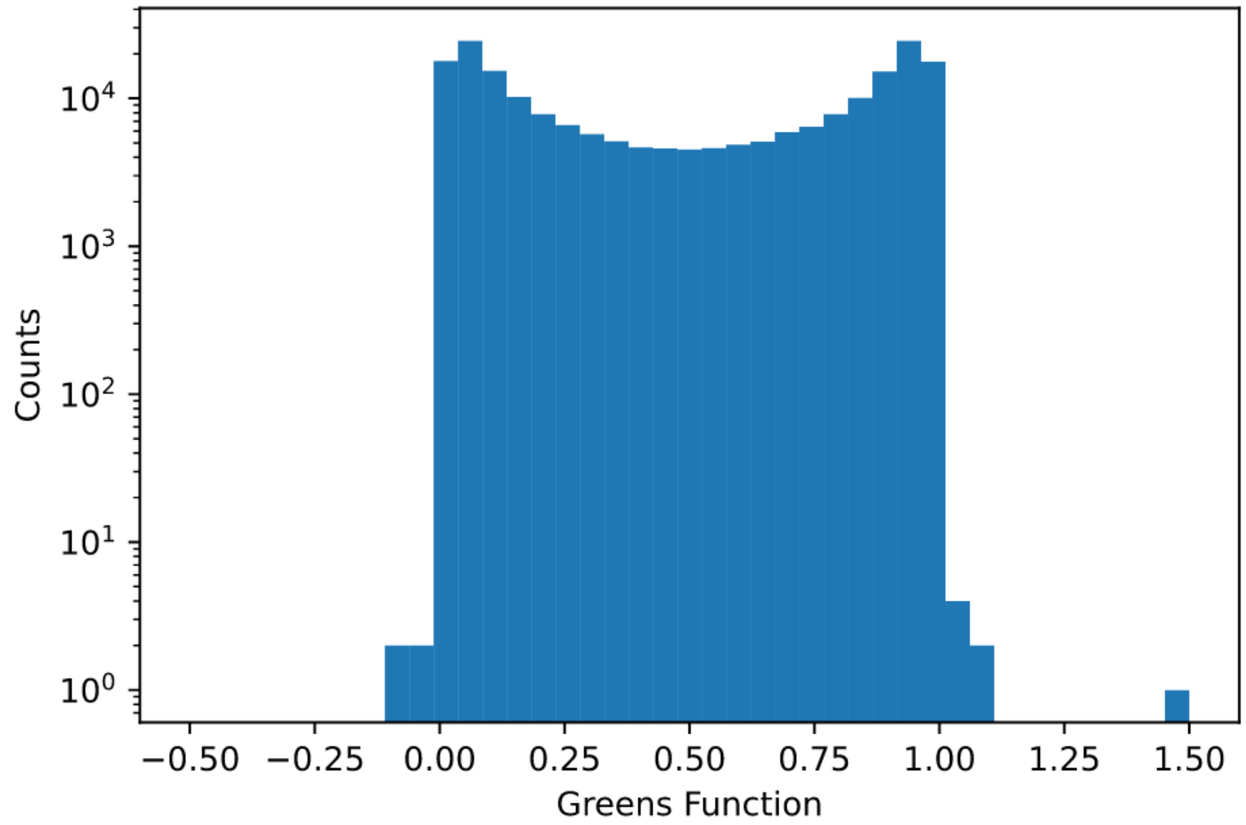
- Final Network DNN with 10% dropout between layers

Flattened(2x2x10) x 160 x 80 x 40 x 4

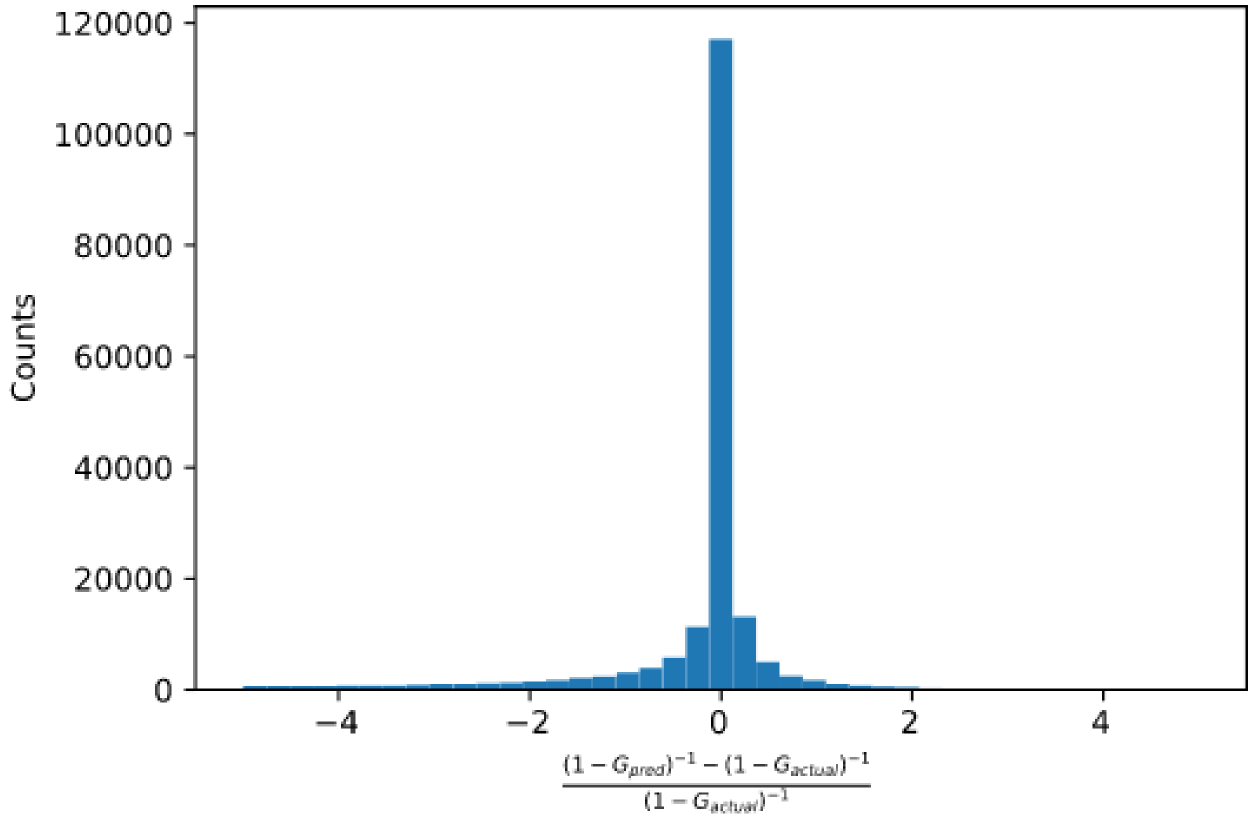


Results

True Green's Functions (DQMC)



Relative effect on accept/reject



Concluding thoughts

- Results promising but not definitive

- To do:
  - Define better metric for success
  - Better MC integration to bypass even more calculatons

Questions?

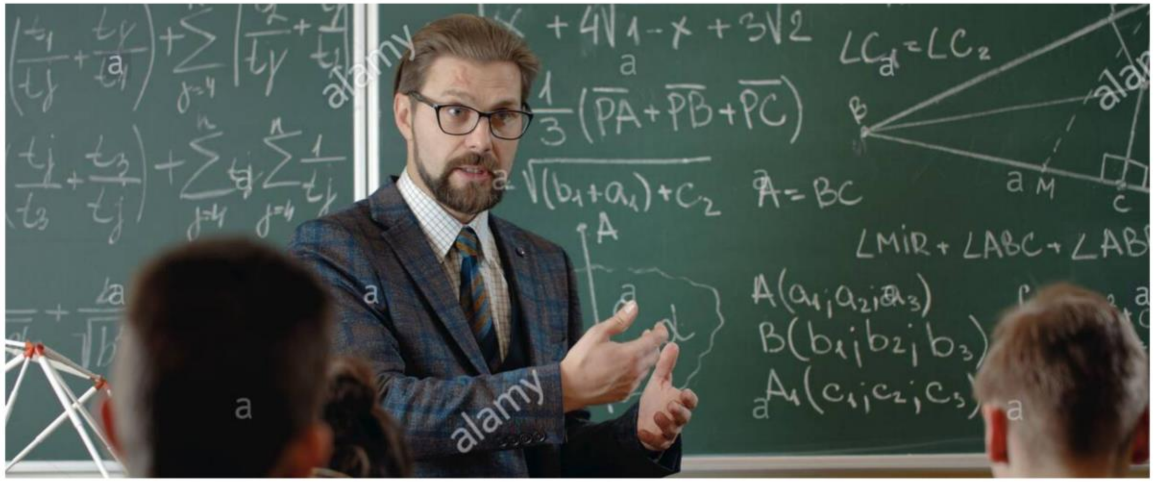
- Freebie question:
  - Why not a CNN?

- References:

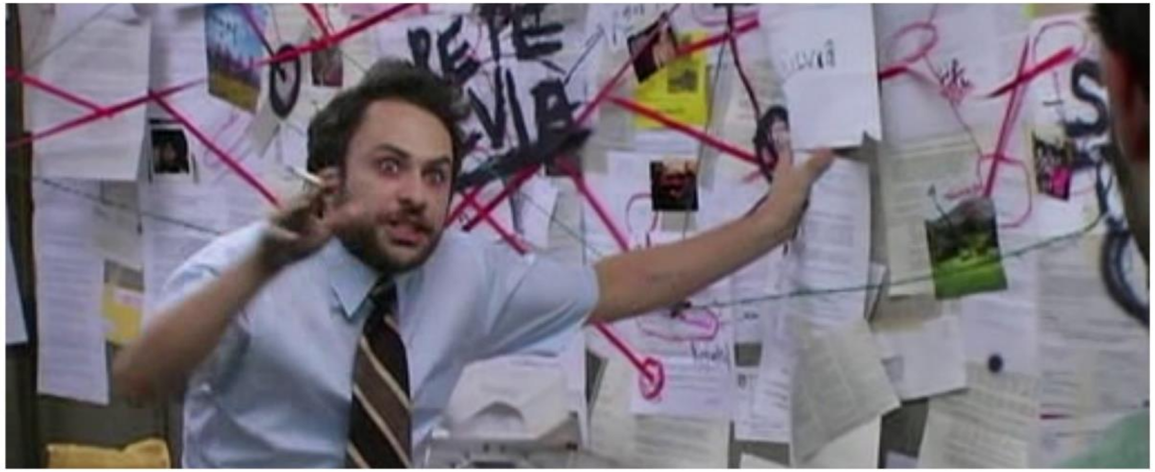
[1] J. Gubernatis, N. Kawashima, and P. Werner, *Quantum monte carlo methods: algorithms for lattice models* (Cambridge University Press, 2016)

[2] S. Johnston, E. A. Nowadnick, Y. F. Kung, B. Moritz, R. T. Scalettar, and T. P. Devereaux, "Determinant quantum monte carlo study of the two-dimensional single-band hubbard-holstein model", Phys. Rev. B **87**, 235133 (2013)

how it feels like explaining my code



how it looks like



r/ProgrammerHumor