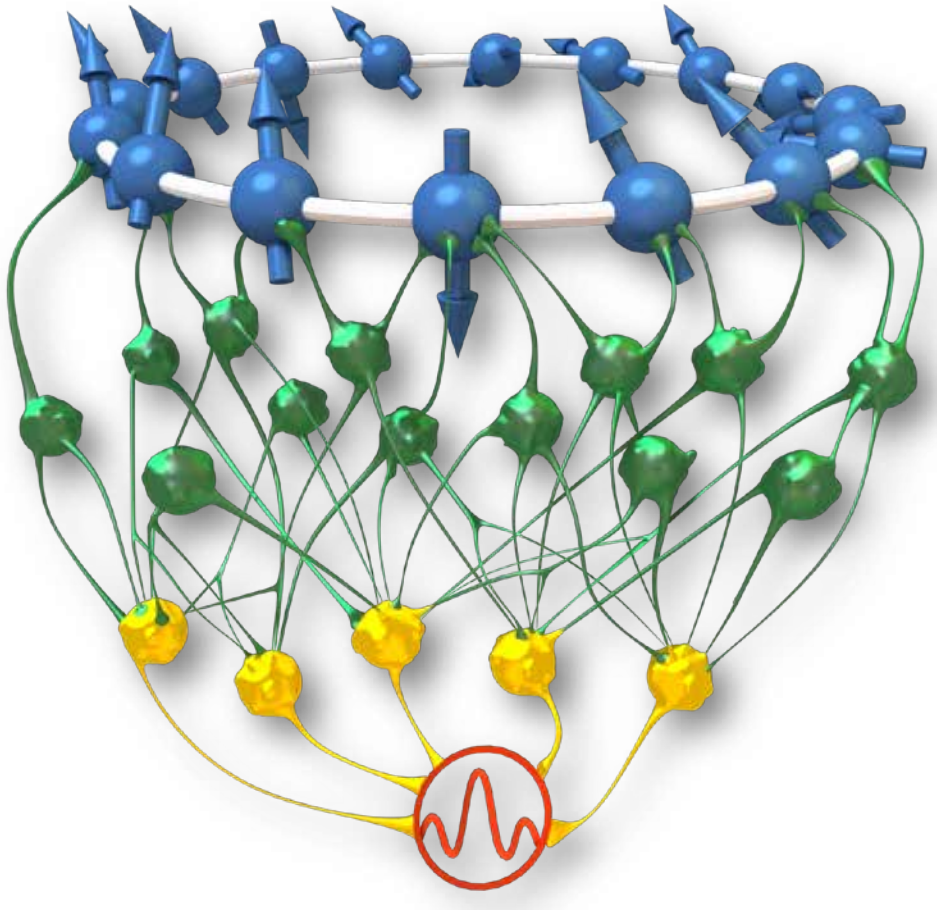
Machine Learning Many-Body Quantum Physics



Filippo Vicentini

Computational quantum Science Lab, EPFL,

Lausanne, Switzerland

(virtually) ICTP
27 January 2021
Trieste

EPFL

Machine Learning for Many-body Systems

Introduction (now)

Coincise introduction to the field

Topics:

- Neural Quantum States
- Sampling
- Optimisation problems (ground state, time-evolution..)

Seminar (13:45)

Given by Prof. Giuseppe Carleo (EPFL)

Giuseppe Carleo

Exciting, new stuff

(please attend, overwise it's obvious I was too boring)

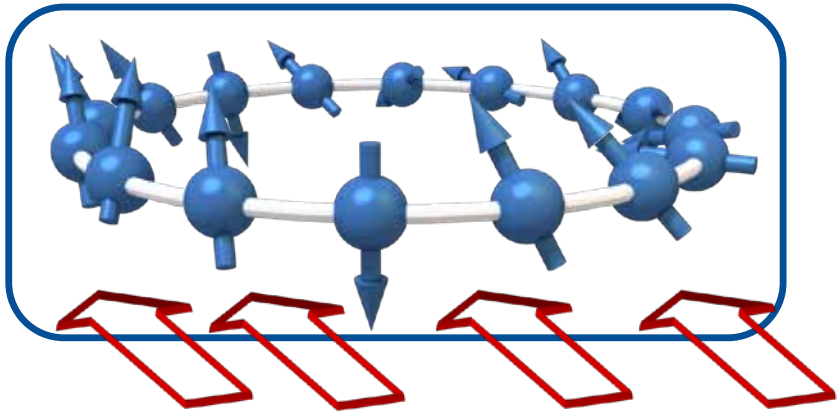
Chapter 1: The tool

Neural Quantum States

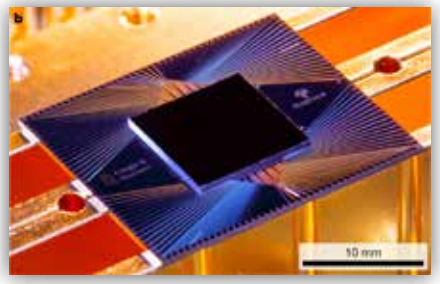
1. Neural Quantum States
2. Variational Monte Carlo

Quantum Physics

Understand the laws governing quantum systems H

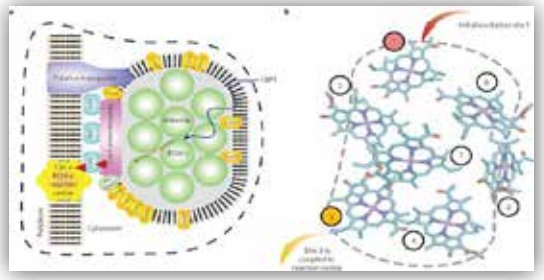


Quantum Computing



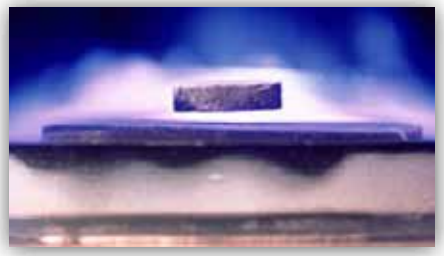
F. Arute et Al.
Nature **574**, 505 (2019)

Quantum Biology



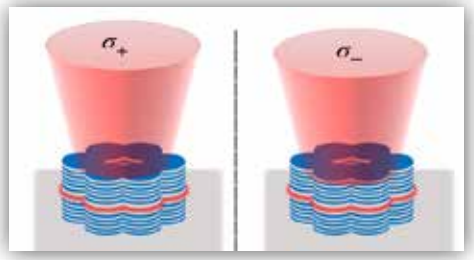
N. Lambert et Al.
Nat Phys **9**, 10 (2013)

Superconductivity



A. Mann
Nature **475**, 280 (2011)

Optical Devices

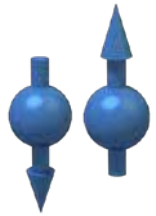


N. Carlon Zambon et Al.
Nature Phot. **13**, 283 (2019)

Quantum State: the wavefunction

A spin: a binary particle which be either $\downarrow$ (0) or $\uparrow$ (1)

Consider a system composed of N-Spins, and its state



Classical State

The state is completely specified by the configuration of each individual spin.

i.e: 3 spins -> $\uparrow\uparrow\downarrow$

For N spins -> N bits of informations

Memory requirements grow linearly $\mathcal{O}(N)$

Quantum State

The state is described by the wavefunction, a probability distribution (+ phase) over all possible configurations.

i.e: 3 spins -> $|\psi\rangle = \begin{bmatrix} \psi(\uparrow\uparrow\uparrow) \\ \psi(\uparrow\uparrow\downarrow) \\ \psi(\uparrow\downarrow\uparrow) \\ \psi(\uparrow\downarrow\downarrow) \\ \psi(\downarrow\uparrow\uparrow) \\ \psi(\downarrow\uparrow\downarrow) \\ \psi(\downarrow\downarrow\uparrow) \\ \psi(\downarrow\downarrow\downarrow) \end{bmatrix}$

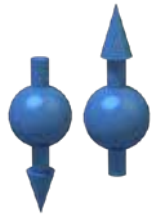
For N spins -> 2^N bits of informations

Memory requirements grow exponentially $\mathcal{O}(2^N)$

Quantum State: the wavefunction

A spin: a binary particle which be either ↓ (0) or ↑ (1)

Consider a system composed of N-Spins, and its state



More Formally…

$$\psi(\sigma_1, \ldots, \sigma_N) = \langle \sigma_1, \ldots, \sigma_N | \psi \rangle$$

$$|\psi\rangle = \sum_{i}^{N} \sum_{\{\sigma_i\}} \psi(\sigma_1, \ldots, \sigma_N) \, |\sigma_1, \ldots, \sigma_N\rangle$$

2^N elements

Quantum State

The state is described by the wavefunction, a probability distribution (+ phase) over all possible configurations.

i.e: 3 spins -> $|\psi\rangle = \begin{bmatrix} \psi(\uparrow\uparrow\uparrow) \\ \psi(\uparrow\uparrow\downarrow) \\ \psi(\uparrow\downarrow\uparrow) \\ \psi(\uparrow\downarrow\downarrow) \\ \psi(\downarrow\uparrow\uparrow) \\ \psi(\downarrow\uparrow\downarrow) \\ \psi(\downarrow\downarrow\uparrow) \\ \psi(\downarrow\downarrow\downarrow) \end{bmatrix}$

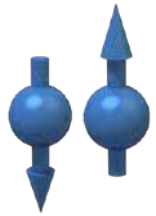
For N spins -> 2^N numbers of informations

Memory requirements grow exponentially $\mathcal{O}(2^N)$

Quantum State: the wavefunction

A spin: a binary particle which be either ↓ (0) or ↑ (1)

Consider a system composed of N-Spins, and its state



More Formally…

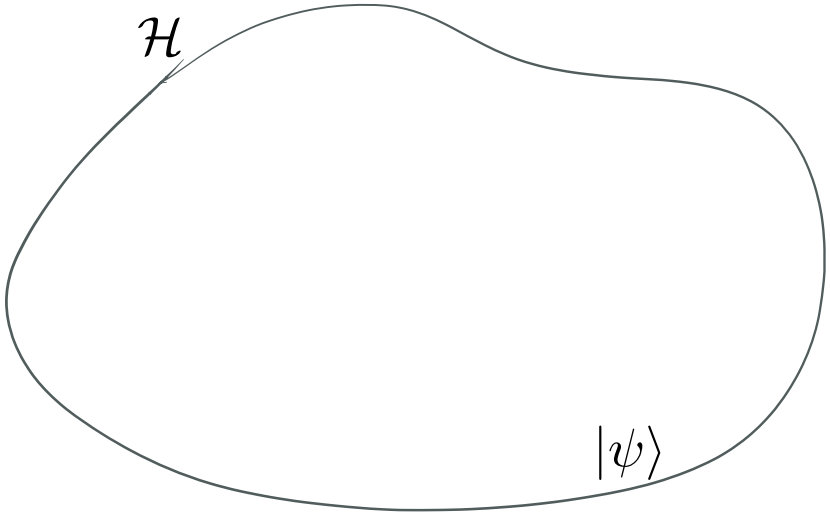
$$\psi(\sigma_1, \ldots, \sigma_N) = \langle \sigma_1, \ldots, \sigma_N | \psi \rangle$$

$$|\psi\rangle = \sum_{i}^{N} \sum_{\{\sigma_i\}} \psi(\sigma_1, \ldots, \sigma_N) \, |\sigma_1, \ldots, \sigma_N\rangle$$

2^N elements

$\mathcal{H}$ Vector space of all possible wavefunctions

$\tilde{\mathcal{H}} \subset \mathcal{H}$ Subset of Physically relevant states



Quantum State: the wavefunction

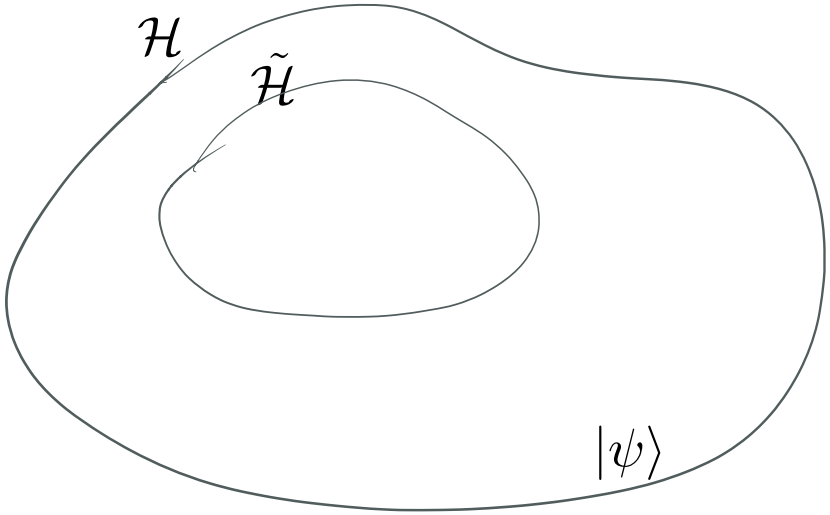
$$\begin{pmatrix} \mathcal{W}_1 \\ \mathcal{W}_2 \\ \vdots \\ \mathcal{W}_M \end{pmatrix} \rightarrow \boxed{\psi} \rightarrow \begin{pmatrix} \langle\downarrow,\downarrow,\ldots\downarrow|\psi(\mathcal{W})\rangle \\ \langle\downarrow,\downarrow,\ldots\uparrow|\psi(\mathcal{W})\rangle \\ \vdots \\ \vdots \\ \vdots \\ \langle\uparrow,\uparrow,\ldots\uparrow|\psi(\mathcal{W})\rangle \end{pmatrix}$$

$$|\psi(\mathcal{W})\rangle = \sum_{i}^{N} \sum_{\{\sigma_i\}} \psi_{\mathcal{W}}(\sigma_1,\ldots,\sigma_N)\,|\sigma_1,\ldots,\sigma_N\rangle$$

$$\mathcal{W} \in \mathbb{C}^M \text{where } M \ll \dim \mathcal{H} \sim 2^N$$

$\mathcal{H}$ Vector space of all possible wavefunctions

$\tilde{\mathcal{H}} \subset \mathcal{H}$ Subset of Physically relevant states



Variational States

$$|\psi(\mathcal{W})\rangle = \sum_{i}^{N} \sum_{\{\sigma_i\}} \psi_{\mathcal{W}}(\sigma_1, \ldots, \sigma_N) \, |\sigma_1, \ldots, \sigma_N\rangle$$

Memory Requirements

$$\mathcal{W} \in \mathbb{C}^M \text{where } M \ll \dim \mathcal{H} \sim 2^N$$

$$M \sim \text{poly}(N)$$

Polynomial memory

Runtime Requirements

2 classes:

- Computationally Tractable States
  - Given the parameters, it is possible to efficiently compute observables with arbitrary precision.
  - Mean Field
  - Gutzwiller Mean Field
  - 1D MPS
  - …
- Computationally Efficient States

Computationally Tractable Variational States

2 Requirements:

Effficiently evaluable

$$\psi_{\mathcal{W}}(\boldsymbol{\sigma}) = \langle \boldsymbol{\sigma} | \psi(\mathcal{W}) \rangle$$

In Poly(N)

Efficiently samplable

$$p_{\mathcal{W}}(\boldsymbol{\sigma}) = \frac{|\langle \boldsymbol{\sigma} | \psi(\mathcal{W}) \rangle|^2}{\sum_{\boldsymbol{\sigma}} |\langle \boldsymbol{\sigma} | \psi(\mathcal{W}) \rangle|^2}$$

In Poly(N)

Theorem

If those two requirements are satisfied, any expectation-value of a K-local operator can be estimated with polynomial accuracy.

At most k-body interactions

[Formalized by Van Der Nest, ArXiv:0911.1624]

Expectation Values

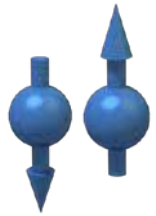
$$\left\langle \hat{O} \right\rangle = \frac{\langle \psi | \hat{O} | \psi \rangle}{\langle \psi | \psi \rangle}$$

$$= \frac{\sum_{\boldsymbol{\sigma}\boldsymbol{\eta}} \langle \psi | \boldsymbol{\sigma} \rangle \langle \boldsymbol{\sigma} | \hat{O} | \boldsymbol{\eta} \rangle \langle \boldsymbol{\eta} | \psi \rangle}{\left| \sum_{\boldsymbol{\sigma}} \langle \boldsymbol{\sigma} | \psi \rangle \right|^2}$$

$$= \frac{\sum_{\boldsymbol{\sigma}} |\langle \psi | \boldsymbol{\sigma} \rangle|^2 \left(\sum_{\boldsymbol{\eta}} \langle \boldsymbol{\sigma} | \hat{O} | \boldsymbol{\eta} \rangle \frac{\langle \boldsymbol{\eta} | \psi \rangle}{\langle \boldsymbol{\sigma} | \psi \rangle} \right)}{\left| \sum_{\boldsymbol{\sigma}} \langle \boldsymbol{\sigma} | \psi \rangle \right|^2}$$

$$= \sum_{\boldsymbol{\sigma}} p(\boldsymbol{\sigma}) \left(\sum_{\boldsymbol{\eta}} O_{\boldsymbol{\sigma},\boldsymbol{\eta}} \frac{\psi(\boldsymbol{\eta})}{\psi(\boldsymbol{\sigma})} \right)$$

$$= \sum_{\boldsymbol{\sigma}} p(\boldsymbol{\sigma}) O^{\mathrm{loc}}(\boldsymbol{\sigma})$$



$$p(\boldsymbol{\sigma}) = \frac{|\langle \psi | \boldsymbol{\sigma} \rangle|^2}{\sum_{\boldsymbol{\sigma}} |\langle \psi | \boldsymbol{\sigma} \rangle|^2}$$

$$O^{\mathrm{loc}}(\boldsymbol{\sigma}) = \sum_{\boldsymbol{\eta}} O_{\boldsymbol{\sigma},\boldsymbol{\eta}} \frac{\psi(\boldsymbol{\eta})}{\psi(\boldsymbol{\sigma})}$$

[Formalized by Van Der Nest, ArXiv:0911.1624]

[Becca, Sorella (2017) Book, chapters 5-6]

METROPOLIS SAMPLING

$$\left\langle \hat{O} \right\rangle = \sum_{\boldsymbol{\sigma}} p(\boldsymbol{\sigma}) O^{\mathrm{loc}}(\boldsymbol{\sigma}) = \left\langle O^{\mathrm{loc}} \right\rangle_{p(\boldsymbol{\sigma})}$$

We have shown that expectation values of an observable O can be computed as averages of O^{loc} over the distribution $p(\sigma)$.

To avoid an exponential cost, we need to define a set

$$\{\boldsymbol{\sigma} \in \mathcal{H}\}, \qquad \dim[\{\boldsymbol{\sigma}\}] \sim \mathrm{poly}(N) \ll \dim[\mathcal{H}] \sim 2^N$$

$$p(\boldsymbol{\sigma}) = \frac{|\langle \psi | \boldsymbol{\sigma} \rangle|^2}{\sum_{\boldsymbol{\sigma}} |\langle \psi | \boldsymbol{\sigma} \rangle|^2}$$

$$\left\langle \hat{O} \right\rangle = \left\langle O^{\mathrm{loc}} \right\rangle_{p(\boldsymbol{\sigma})} = \frac{1}{N_s} \sum_{\{\boldsymbol{\sigma}\}}^{N_s} O^{\mathrm{loc}}(\boldsymbol{\sigma}) \pm \sqrt{\frac{\mathbb{V}\mathrm{ar}(O^{\mathrm{loc}})}{N_s}}$$

$$O^{\mathrm{loc}}(\boldsymbol{\sigma}) = \sum_{\boldsymbol{\eta}} O_{\boldsymbol{\sigma},\boldsymbol{\eta}} \frac{\psi(\boldsymbol{\eta})}{\psi(\boldsymbol{\sigma})}$$

[Formalized by Van Der Nest, ArXiv:0911.1624]

[Becca, Sorella (2017) Book, chapters 5-6]

Generating samples: Markov Chains

Metropolis-Hastings Monte carlo sampling

- If you can compute $f(\boldsymbol{\sigma}) \propto p(\boldsymbol{\sigma})$
- We want to generate a chain of states

$$\boldsymbol{\sigma}_0 \to \boldsymbol{\sigma}_1 \to \boldsymbol{\sigma}_2 \to \cdots \to \boldsymbol{\sigma}_{N_s}$$

Easy to compute

$$p(\boldsymbol{\sigma}) = \frac{|\langle\psi|\boldsymbol{\sigma}\rangle|^2}{\sum_{\boldsymbol{\sigma}} |\langle\psi|\boldsymbol{\sigma}\rangle|^2}$$

Exponentially hard (sum over hilb. Space)

[Hastings, W.K. , Biometrika 57, 97-109 (1970)]

[for a simple introduction see: Robert, Arxiv: 1504.01896]

GENERATING SAMPLES: MARKOV CHAINS

METROPOLIS-HASTINGS MONTE CARLO SAMPLING

- If you can compute $f(\boldsymbol{\sigma}) \propto p(\boldsymbol{\sigma})$
- We want to generate a chain of states

  $\boldsymbol{\sigma}_0 \to \boldsymbol{\sigma}_1 \to \boldsymbol{\sigma}_2 \to \cdots \to \boldsymbol{\sigma}_{N_s}$
- We start with $\boldsymbol{\sigma}_0$
- At every iteration i, $\boldsymbol{\sigma}_i$

  A. We propose a new state $\boldsymbol{\sigma}'$ by sampling $T(\boldsymbol{\sigma}'|\boldsymbol{\sigma}_i)$
  B. We compute the 'acceptance probability'

$$A(\boldsymbol{\sigma}'|\boldsymbol{\sigma}) = \min\left[1, \frac{p(\boldsymbol{\sigma}')}{p(\boldsymbol{\sigma}_i)}\frac{T(\boldsymbol{\sigma}'|\boldsymbol{\sigma}_i)}{T(\boldsymbol{\sigma}_i|\boldsymbol{\sigma}')}\right]$$

  C. With probability $A(\boldsymbol{\sigma}'|\boldsymbol{\sigma})$ accept the move, $\boldsymbol{\sigma}_{i+1} = \boldsymbol{\sigma}'$
  otherwise 'reject' it, setting $\boldsymbol{\sigma}_{i+1} = \boldsymbol{\sigma}_i$

[Hastings, W.K. , Biometrika 57, 97-109 (1970)]

Easy to compute

$$p(\boldsymbol{\sigma}) = \frac{|\langle\psi|\boldsymbol{\sigma}\rangle|^2}{\sum_{\boldsymbol{\sigma}}|\langle\psi|\boldsymbol{\sigma}\rangle|^2}$$

Exponentially hard (sum over hilb. Space)

[for a simple introduction see: Robert, Arxiv: 1504.01896]

GENERATING SAMPLES: MARKOV CHAINS

METROPOLIS-HASTINGS MONTE CARLO SAMPLING

- If you can compute $f(\boldsymbol{\sigma}) \propto p(\boldsymbol{\sigma})$
- We want to generate a chain of states

  $\boldsymbol{\sigma}_0 \rightarrow \boldsymbol{\sigma}_1 \rightarrow \boldsymbol{\sigma}_2 \rightarrow \cdots \rightarrow \boldsymbol{\sigma}_{N_s}$
- We start with $\boldsymbol{\sigma}_0$
- At every iteration i, $\boldsymbol{\sigma}_i$

  A. We propose a new state $\boldsymbol{\sigma}'$ by sampling $T(\boldsymbol{\sigma}'|\boldsymbol{\sigma}_i)$
  B. We compute the 'acceptance probability'

$$A(\boldsymbol{\sigma}'|\boldsymbol{\sigma}) = \min\left[1, \frac{p(\boldsymbol{\sigma}')}{p(\boldsymbol{\sigma}_i)} \frac{T(\boldsymbol{\sigma}'|\boldsymbol{\sigma}_i)}{T(\boldsymbol{\sigma}_i|\boldsymbol{\sigma}')}\right]$$

  C. With probability $A(\boldsymbol{\sigma}'|\boldsymbol{\sigma})$ accept the move, $\boldsymbol{\sigma}_{i+1} = \boldsymbol{\sigma}'$
  otherwise 'reject' it, setting $\boldsymbol{\sigma}_{i+1} = \boldsymbol{\sigma}_i$

[Hastings, W.K. , Biometrika 57, 97-109 (1970)]

This technique is derived from the **Detailed balance,** or **microscopic reversibility condition**

At equilibrium, the following is valid:

$$P(\boldsymbol{\sigma}'|\boldsymbol{\sigma})p(\boldsymbol{\sigma}) = P(\boldsymbol{\sigma}|\boldsymbol{\sigma}')p(\boldsymbol{\sigma}')$$

And by defining the transition probability:

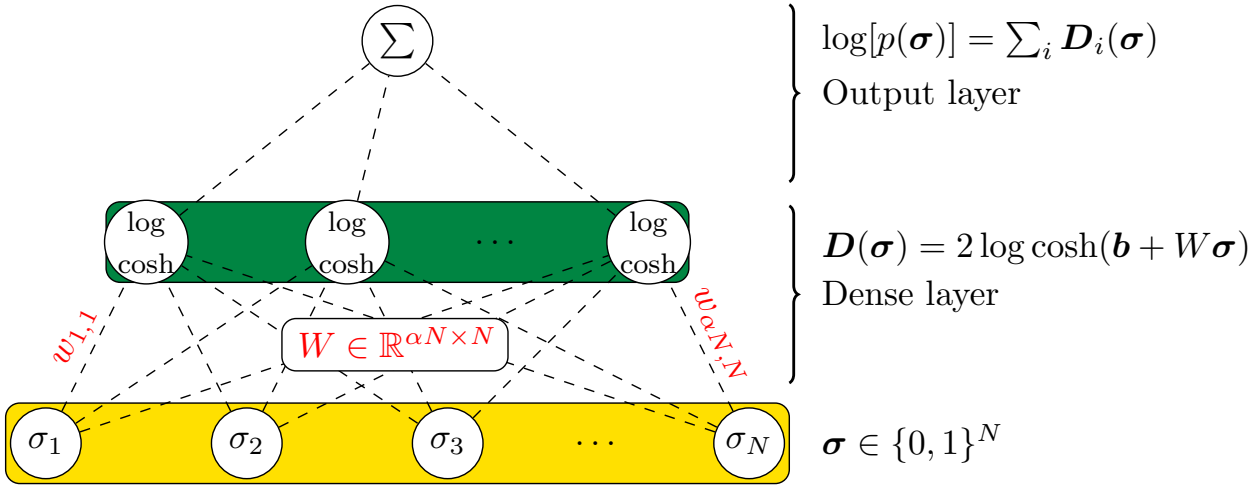
$$P(\boldsymbol{\sigma}'|\boldsymbol{\sigma}) = T(\boldsymbol{\sigma}'|\boldsymbol{\sigma})A(\boldsymbol{\sigma}'|\boldsymbol{\sigma})$$

[for a simple introduction see: Robert, Arxiv: 1504.01896]

Neural Quantum States

Restricted Bolzmann Machine:

$$\psi(\{W, \boldsymbol{b}\}, \boldsymbol{\sigma}) = \exp\left[\sum_i \log\cosh\left(\boldsymbol{W}^{(i)}\boldsymbol{\sigma} + b^{(i)}\right)\right]$$



Chapter 2: The problem

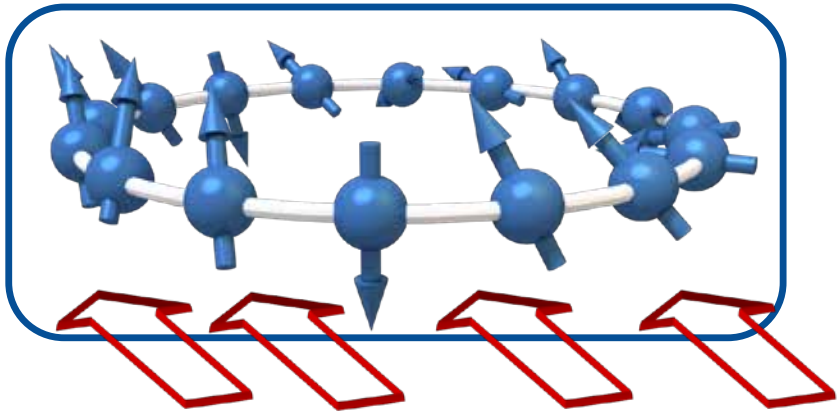
Variational Monte Carlo

1. Neural Quantum States
2. Variational Monte Carlo

QUANTUM PHYSICS: 2 PROBLEMS (AMONG MANY)

SIMULATION: COMPUTING THE GROUND/STEADY STATE

Given a system encoded by the hamiltonian H,
find the fundamental (ground) state



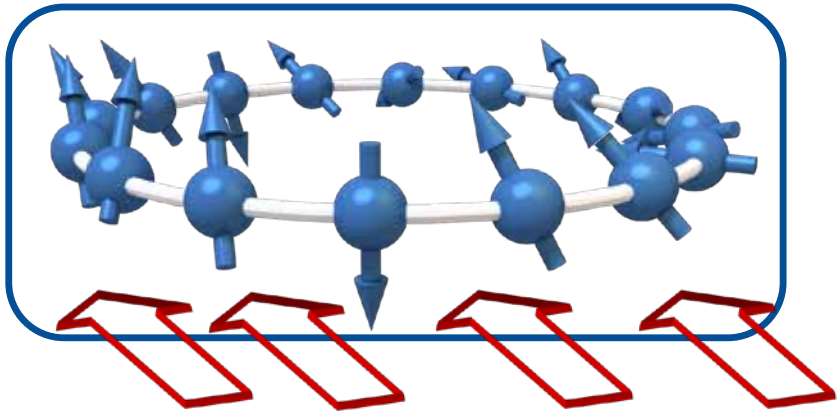
$$\hat{H}\,|\psi_{gs}\rangle = E_{gs}\,|\psi_{gs}\rangle$$

[Carleo and Troyer, Science **355**, 602 (2017)]

Quantum Physics: 2 problems (among many)

Simulation: Computing the ground/steady state

Given a system encoded by the hamiltonian H,
find the fundamental (ground) state



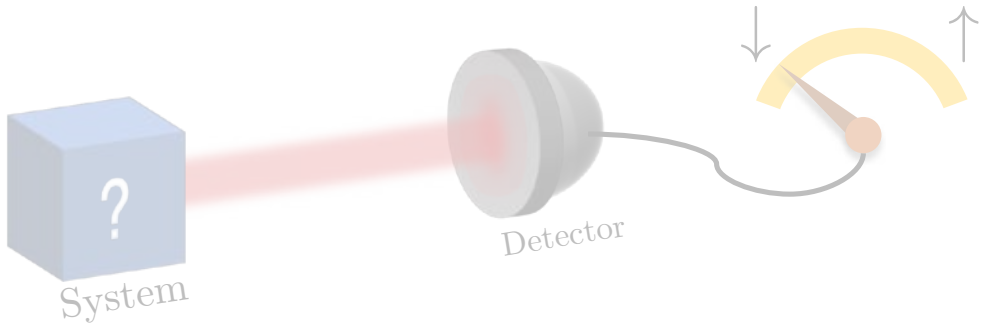
$$\hat{H}\,|\psi_{gs}\rangle = E_{gs}\,|\psi_{gs}\rangle$$

[Carleo and Troyer, Science **355**, 602 (2017)]

Experiments: Reconstruct a quantum state

Given a quantum system in an unknown state
(black box)

Perform some measurement S and reconstruct the
State



Probably next week: Juan Carrasquilla

[Torlai et Al, Nature Physics **14**, 447 (2018)]

Determining the Ground State

Ground State

$$\hat{H}\,|\psi_{gs}\rangle = E_{gs}\,|\psi_{gs}\rangle$$

We want to use a flexible representation of $|\psi\rangle$:

- Neural Quantum States

$$\mathcal{W}_{gs} \quad \text{so that} \quad |\psi(\mathcal{W}_{gs})\rangle \approx |\psi_{gs})\rangle$$

Variational Principle

$$E(\mathcal{W}) = \left\langle \hat{H} \right\rangle = \frac{\langle\psi(\mathcal{W})|\hat{H}|\psi(\mathcal{W})\rangle}{\langle\psi(\mathcal{W})|\psi(\mathcal{W})\rangle} \geq E_{gs}$$

$$E(\mathcal{W}) = E_{gs} \quad \Rightarrow \quad |\psi(\mathcal{W})\rangle = |\psi_{gs}\rangle$$

So we want to determine the

$$\min_{\mathcal{W}}[E(\mathcal{W})]$$

[simple introduction, see Wikipedia: Variational principle (Quantum Mechanics)]

[Landau, Quantum Mechanics (pg. 58)]

VARIATIONAL OPTIMISATION

OPTIMISATION

I start from a set of parameters

$$\mathcal{W}_0 \Rightarrow E(\mathcal{W}_0)$$

I can also compute the gradient of the energy

$$\nabla_{\mathcal{W}} E(\mathcal{W}_0)$$

And use it to optimisee the parameters

$$\mathcal{W}_{i+1} = \mathcal{W}_i - \eta \nabla_{\mathcal{W}} E(\mathcal{W}_i)$$

Can we compute the gradient efficiently?

VARIATIONAL PRINCIPLE

$$E(\mathcal{W}) = \left\langle \hat{H} \right\rangle = \frac{\langle \psi(\mathcal{W}) | \hat{H} | \psi(\mathcal{W}) \rangle}{\langle \psi(\mathcal{W}) | \psi(\mathcal{W}) \rangle} \geq E_{gs}$$

$$E(\mathcal{W}) = E_{gs} \quad \Rightarrow \quad |\psi(\mathcal{W})\rangle = |\psi_{gs}\rangle$$

So we want to determine the

$$\min_{\mathcal{W}} [E(\mathcal{W})]$$

[Landau, Quantum Mechanics (pg. 58)]

VARIATIONAL OPTIMISATION

But we cannot compute the energy and it's gradient exactly:

$$E(\mathcal{W}) = \sum_{\boldsymbol{\sigma}} p_{\mathcal{W}}(\boldsymbol{\sigma}) E^{\mathrm{loc}}_{\mathcal{W}}(\boldsymbol{\sigma}) = \left\langle E^{\mathrm{loc}}_{\mathcal{W}} \right\rangle_p$$

$$E^{\mathrm{loc}}_{\mathcal{W}}(\boldsymbol{\sigma}) = \sum_{\boldsymbol{\sigma}'} \langle \boldsymbol{\sigma} | \hat{H} | \boldsymbol{\sigma}' \rangle \frac{\psi_{\mathcal{W}}(\boldsymbol{\sigma}')}{\psi_{\mathcal{W}}(\boldsymbol{\sigma})}$$

$$p_{\mathcal{W}}(\boldsymbol{\sigma}) = |\psi_{\mathcal{W}}(\boldsymbol{\sigma})|^2 / Z$$

So the gradient is the vector

$$\boldsymbol{\nabla}_{\mathcal{W}} E(\mathcal{W}) = \begin{pmatrix} \nabla_{\mathcal{W}^1} E(\mathcal{W}) \\ \nabla_{\mathcal{W}^2} E(\mathcal{W}) \\ \vdots \\ \nabla_{\mathcal{W}^{N_p}} E(\mathcal{W}) \end{pmatrix}$$

$$O^k_{\mathcal{W}}(\boldsymbol{\sigma}) = \nabla_{\mathcal{W}^k} \log \psi_{\mathcal{W}}(\boldsymbol{\sigma})$$

$$\nabla_{\mathcal{W}^k} E(\mathcal{W}) = \left\langle E^{\mathrm{loc}} O^k_{\mathcal{W}} \right\rangle_p - \left\langle E^{\mathrm{loc}} \right\rangle_p \left\langle O^k_{\mathcal{W}} \right\rangle_p$$

VARIATIONAL OPTIMISATION

OPTIMISATION

I start from a set of parameters

$$\mathcal{W}_0 \Rightarrow E(\mathcal{W}_0)$$

I can also compute the gradient of the energy

$$\boldsymbol{\nabla}_{\mathcal{W}} E(\mathcal{W}_0)$$

And use it to optimisee the parameters

$$\mathcal{W}_{i+1} = \mathcal{W}_i - \eta \boldsymbol{\nabla}_{\mathcal{W}} E(\mathcal{W}_i)$$

Can we compute the gradient efficiently?

STOCHASTIC GRADIENT (DESCENT)

$$\mathcal{W}_{i+1} = \mathcal{W}_i - \eta \left(\langle \nabla_{\mathcal{W}^k} E(\mathcal{W}) \rangle_p + \mathcal{N}(\mu = 0, \sigma^2 = \frac{\mathbb{V}\mathrm{ar}}{N_s}) \right)$$

In the limit $N_s \to \infty$ the stochastic term is 0 and we recover the exact result.

$$x(t + dt) = x(t) - dt \nabla V(t) + \mathcal{N}(\mu = 0, \sigma^2 = 2T dt)$$

Otherwise, notice the similarity to the Langevin process Describing a particle in a potential and temperature T.

$$T \propto {N_S}^{-1}$$

[Benjo, Goodfellow, Deep Learning book]

Time Evolution

Consider the state

$$|\psi_{\mathcal{W}}\rangle$$

And consider an infinitesimal time evolution of the same state (tau real or imaginary)

$$e^{-idt\hat{H}}\,|\psi_{\mathcal{W}}\rangle \approx (\mathbb{I} - idt\hat{H})\,|\psi_{\mathcal{W}}\rangle$$

Expanding to first order in dW

$$|\psi_{\mathcal{W}+\delta\mathcal{W}}\rangle \approx |\psi_{\mathcal{W}}\rangle + \sum_k \delta\mathcal{W}_k O^k\,|\psi_W\rangle = (\mathbb{I} + \sum_k \delta\mathcal{W}_k O^k)\,|\psi_{\mathcal{W}}\rangle$$

Then we want to find dW that matches the two equations above

$$\max_{\delta\mathcal{W}} \left[\frac{\left| \langle\psi_{\mathcal{W}}|\, e^{idt\hat{H}}\,|\psi_{\mathcal{W}+\delta\mathcal{W}}\rangle \right|^2}{|\psi_{\mathcal{W}+\delta\mathcal{W}}| \left| e^{-idt\hat{H}}\,|\psi_{\mathcal{W}}\rangle \right|} \right]$$

[Becca, Sorella (2017) Book, chapters 7]

Time Evolution

Solving that system gives:

$$i \sum_{k'} S_k^{k'} \delta \mathcal{W}'_k = \nabla_{\mathcal{W}_k} E(\mathcal{W})$$

(same equation for imaginary time, except i -> 1)

Quantum Geometric Tensor $$S_k^{k'} = \left\langle O_k^\star O^{k'} \right\rangle_p - \langle O_k \rangle_p^\star \left\langle O^{k'} \right\rangle_p$$

Q.G.T: Local approximation of the metric tensor in variational space

Symbolically it is solved by:

$$\boldsymbol{\delta \mathcal{W}} = S^{-1} \boldsymbol{\nabla}_{\mathcal{W}} E(\mathcal{W})$$

But in general it is best to solve it with a linear solver (CG, GMRES, MINRES…)

Then inserted back into

$$\mathcal{W}_{i+1} = \mathcal{W}_i - \delta \mathcal{W}$$

CONCLUSIONS

Neural Quantum States are efficient encodings of Quantum states

- Can estimate expectation values efficiently through sampling
- Can compute the gradient of the energy efficiently

The Variational Principle recasts the ground state problem into optimisation problem

Also valid for time evolution (Quantum Geometric Tensor)

Those techniques are implemented in Netket

http://netket.org