NEURAL NETWORKS AS QUANTUM STATES

2023 Perimeter-SAIFR Journeys into Theoretical Physics

Juan Felipe Carrasquilla Álvarez, July 17th, 2023
Vector Institute

VECTOR INSTITUTE | INSTITUT VECTEUR CIFAR NSERC CRSNG compute canada | calcul canada UNIVERSITY OF WATERLOO UNIVERSITY OF TORONTO

Can we represent quantum states in terms of neural networks?

Yes and we will look into the Toric code Hamiltonian

Can we deal with disordered and topological phases not described by order parameters?

Phases of matter without an order parameter at T=0

- Topological phases of matter. Examples: Fractional quantum hall effect, quantum spin liquids, Ising gauge theory. Potential applications in topological quantum computing. These phases defy the Landau symmetry breaking classification.

- Coulomb phases = Highly correlated “spin liquids” described by electrodynamics. Examples: Common water ice and spin ice materials ($Ho_2Ti_2O_7$ and $Dy_2Ti_2O_7$)

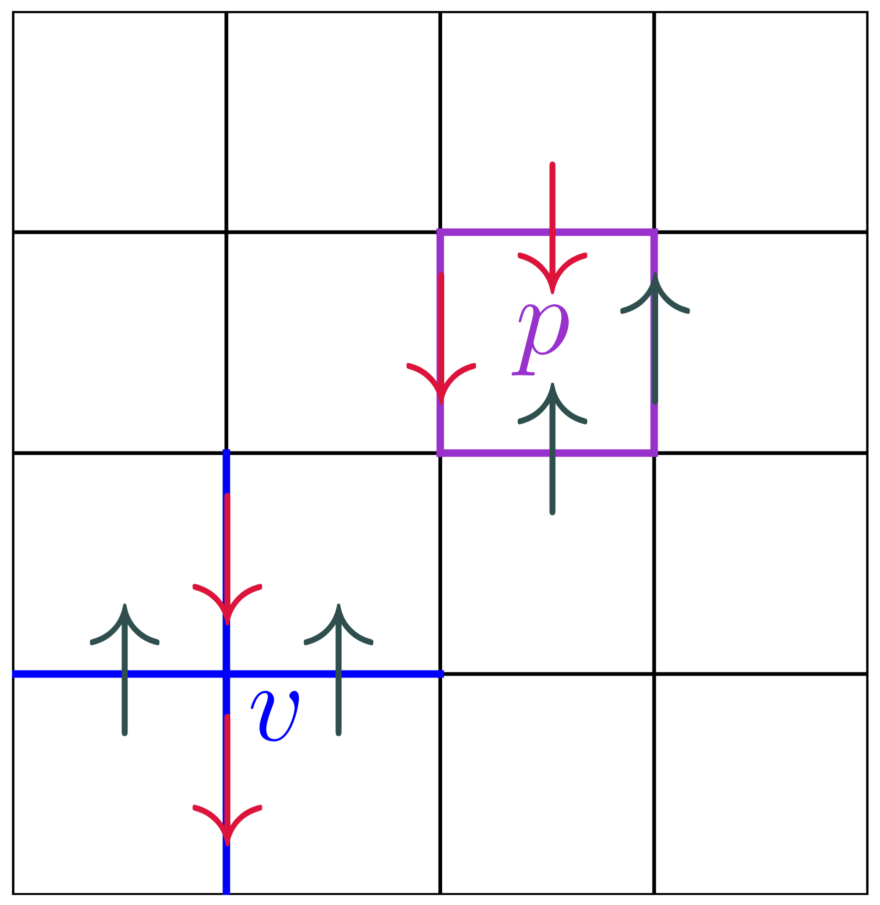
Phases of matter without an order parameter at T=0,∞

Wegner's Ising gauge theory:

$$H = -J \sum_p \prod_{i \in p} \sigma_i^z$$

F.J. Wegner, J. Math. Phys. 12 (1971) 2259

(Kogut Rev. Mod. Phys. 51, 659 (1979))

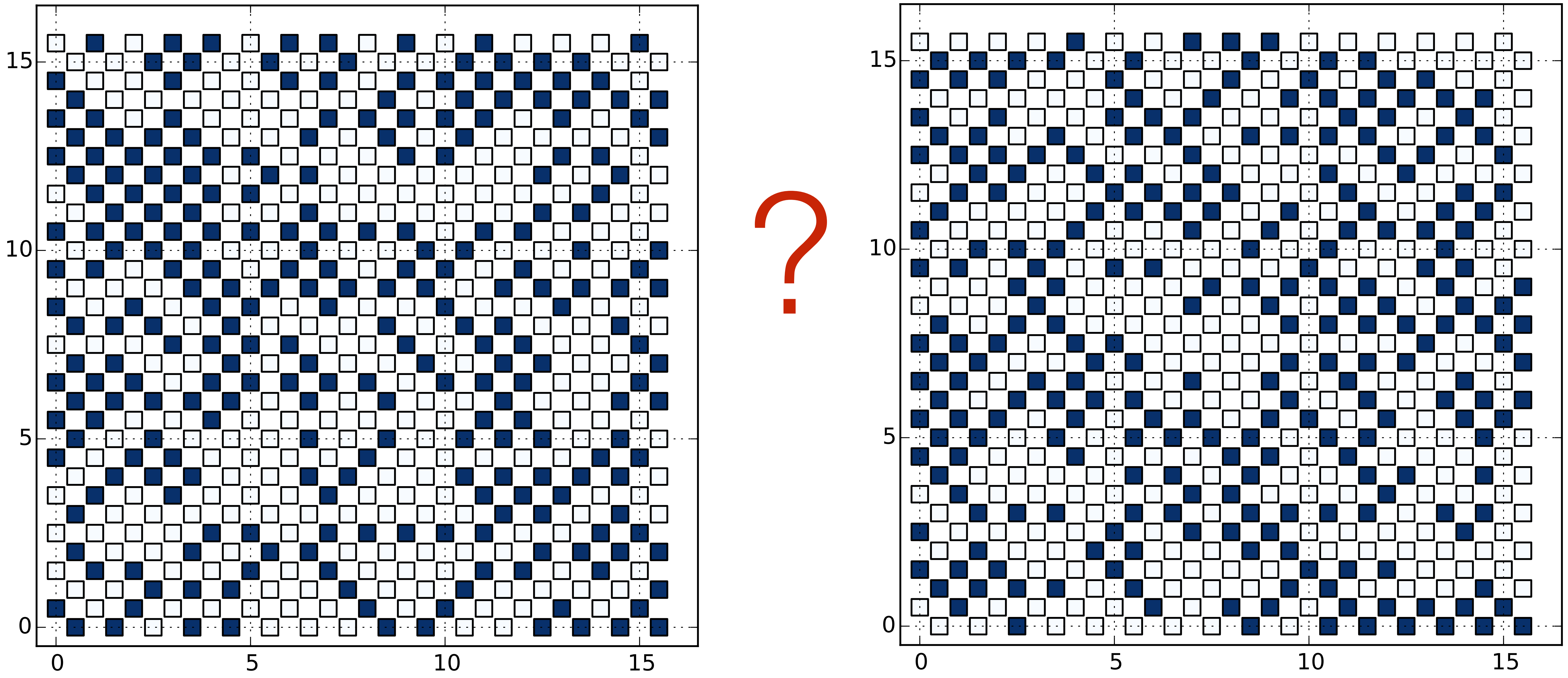


The grandmother of most lattice models for topological quantum computation

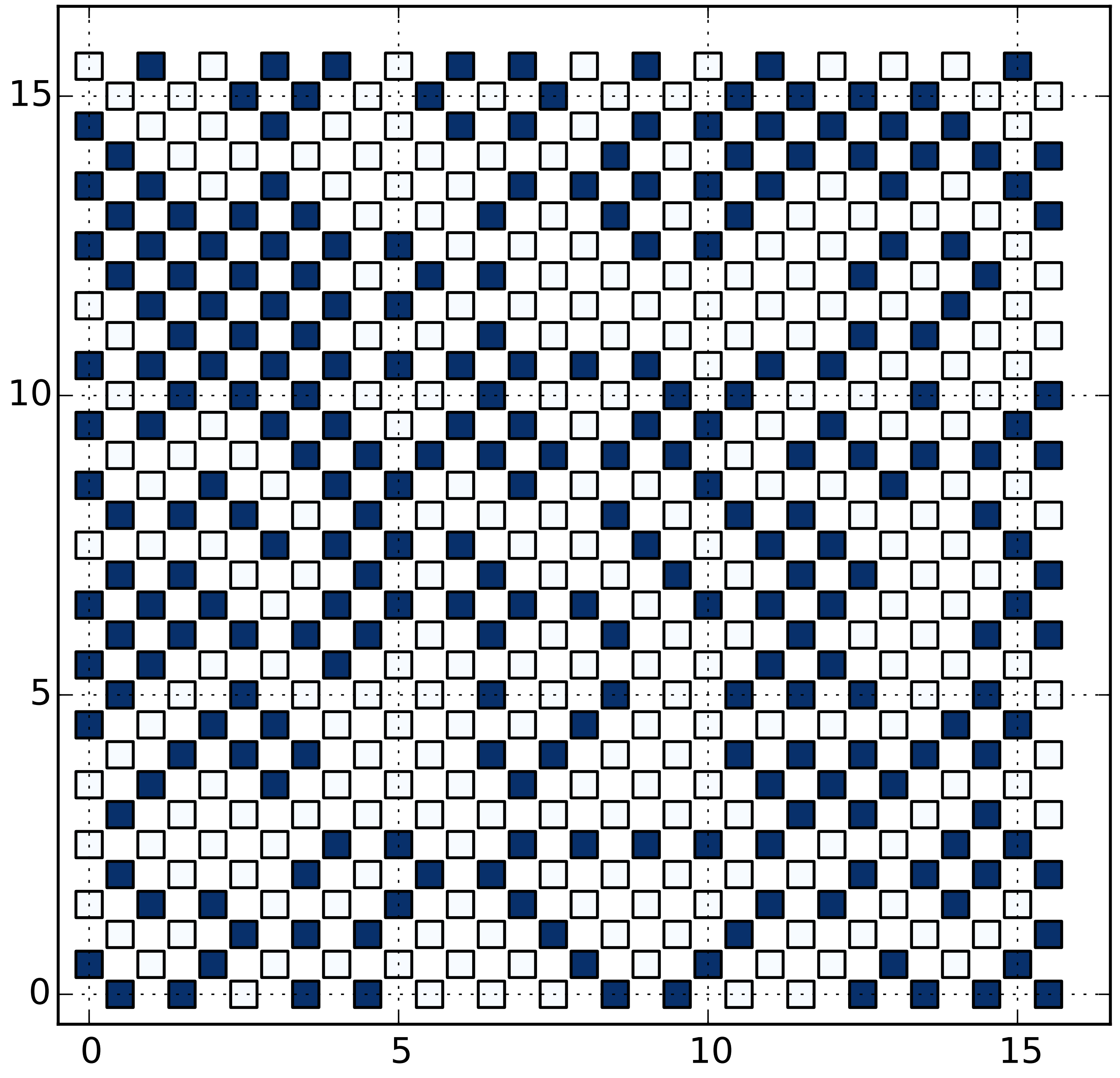
The ground state is a highly degenerate manifold with exponentially decaying spin–spin correlations. Ground state is a disordered topologically ordered phase

Castelnovo and Chamon Phys. Rev. B 76, 174416 (2007)

For two configurations

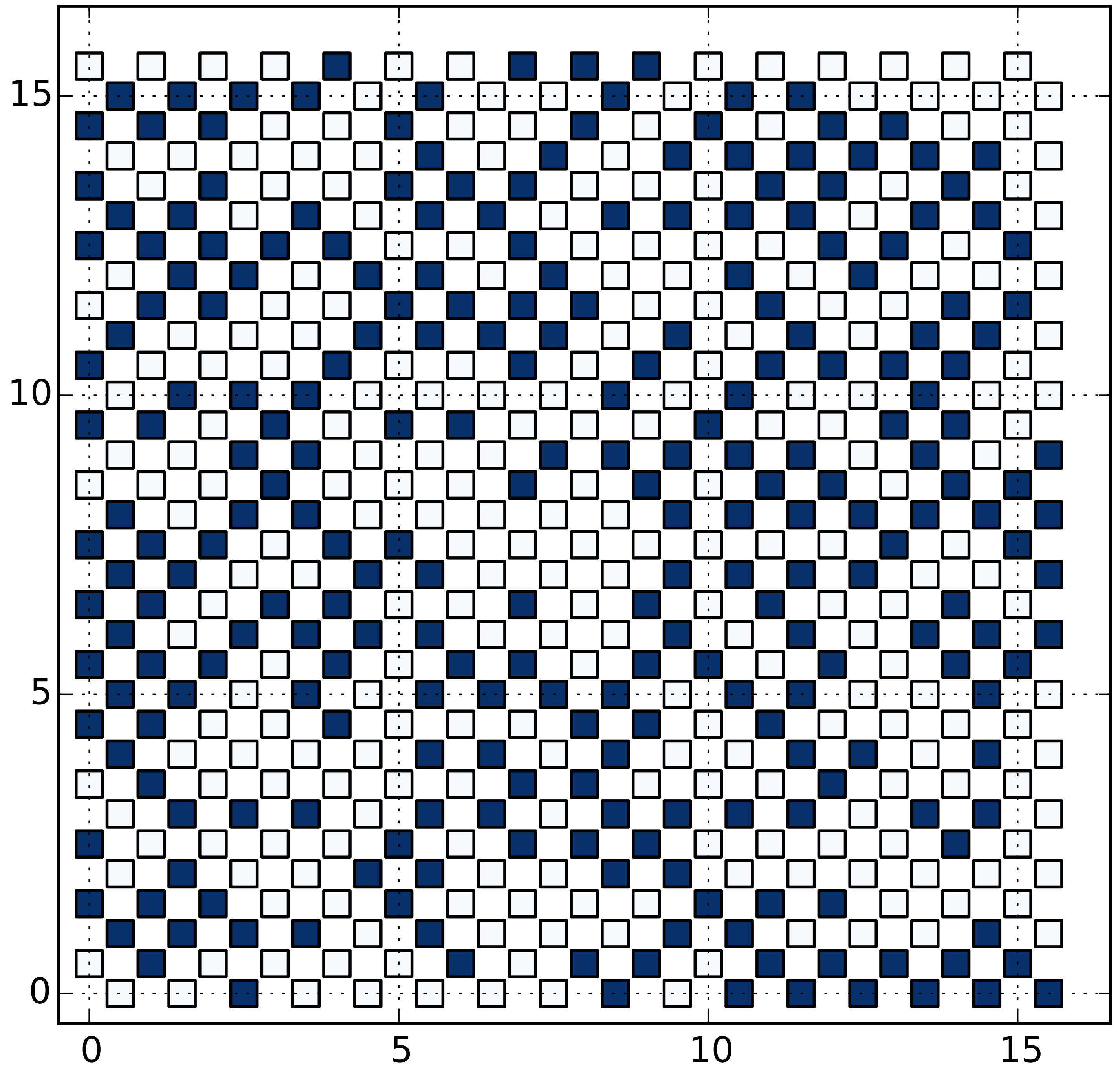


For two configurations



Ground state

?



high-temperature state

Feedforward NN are difficult to apply to this problem and lead to 50% accuracy

Blackboard CNNs

2-D convolution

- The thing we convolve by is called a kernel, or filter. What does this convolution kernel do?

$*$

0	1	0
1	4	1
0	1	0

2-D convolution

- The thing we convolve by is called a kernel, or filter. What does this convolution kernel do?

$*$

0	1	0
1	4	1
0	1	0

Blur/ weighted average

2-D convolution

- What does this convolution kernel do?

$*$

0	-1	0
-1	8	-1
0	-1	0

2-D convolution

- What does this convolution kernel do?

$*$

0	-1	0
-1	8	-1
0	-1	0

Sharpen (image + laplacian edge detection)

2-D convolution

- What does this convolution kernel do?

$*$

0	-1	0
-1	4	-1
0	-1	0

2-D convolution

- What does this convolution kernel do?

$*$

0	-1	0
-1	4	-1
0	-1	0

- the Laplacian edge detector

2-D convolution

- What does this convolution kernel do?

$$* \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$$

2-D convolution

- What does this convolution kernel do?

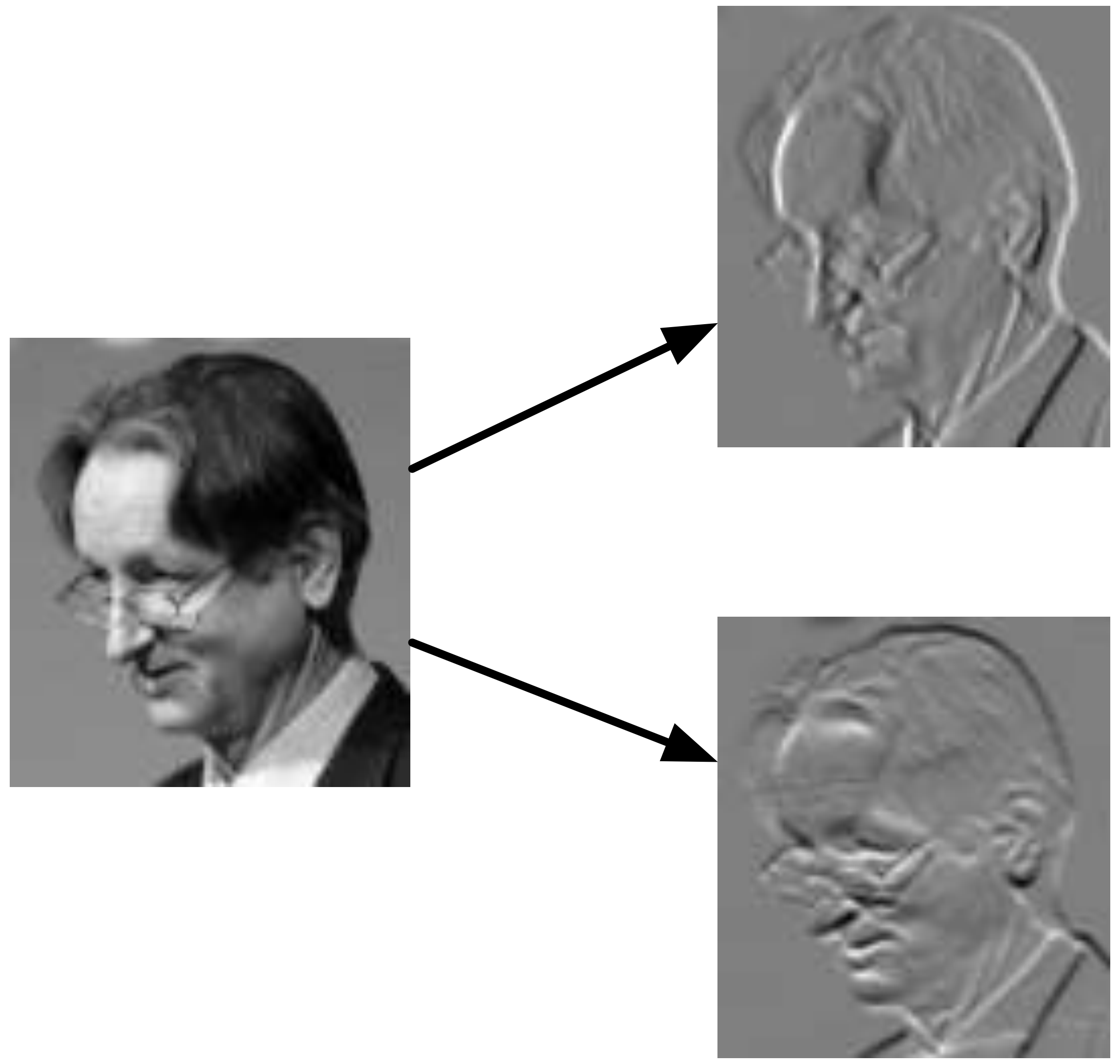
*

1	0	-1
2	0	-2
1	0	-1

Sobel operator/ weighted averaged derivative wrt x

Convolutional neural networks

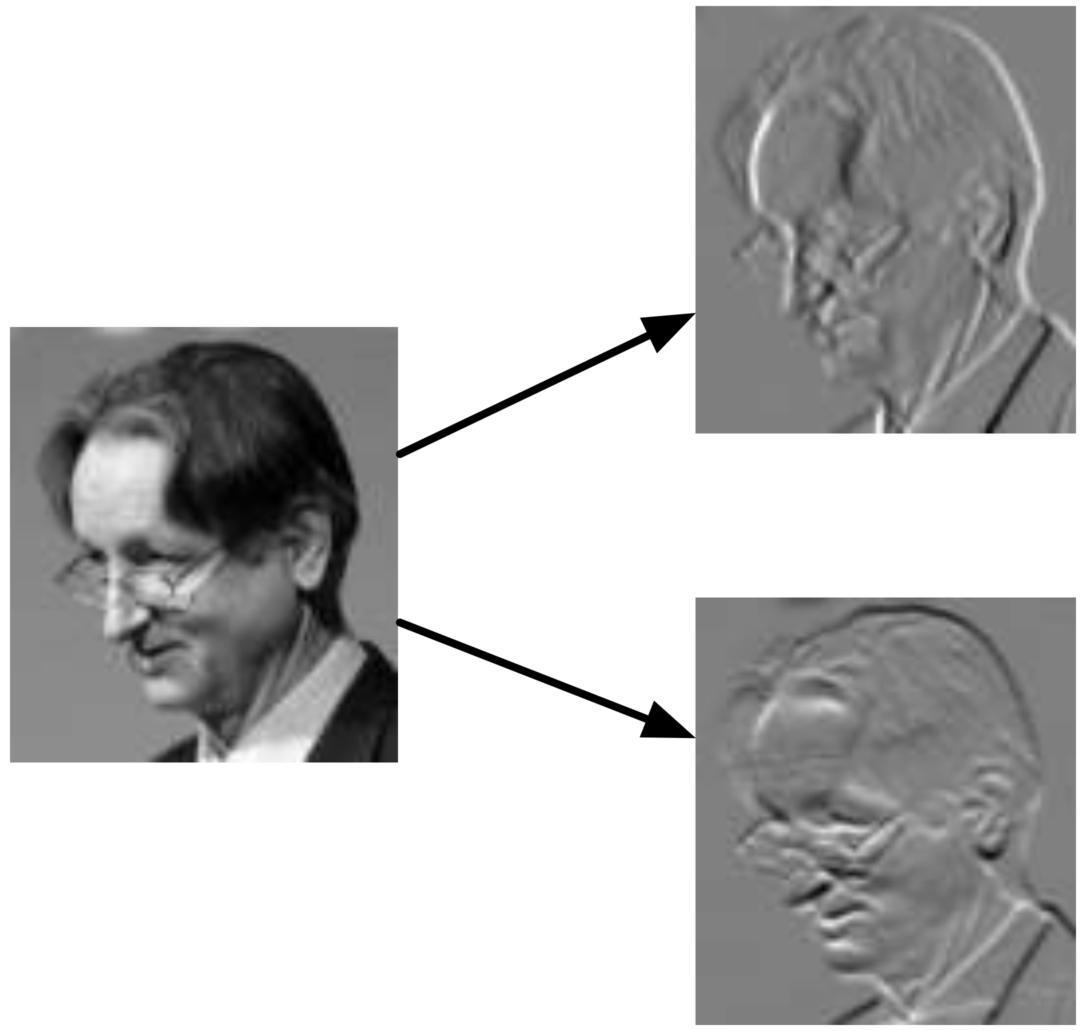
- Let's finally turn to convolutional networks. These have two kinds of layers: detection layers (or convolution layers), and pooling layers.
- The convolution layer has a set of filters. Its output is a set of feature maps, each one obtained by convolving the image with a filter.



convolution

Convolutional neural networks

- Let's finally turn to convolutional networks. These have two kinds of layers: detection layers (or convolution layers), and pooling layers.
- The convolution layer has a set of filters. Its output is a set of feature maps, each one obtained by convolving the image with a filter.



convolution

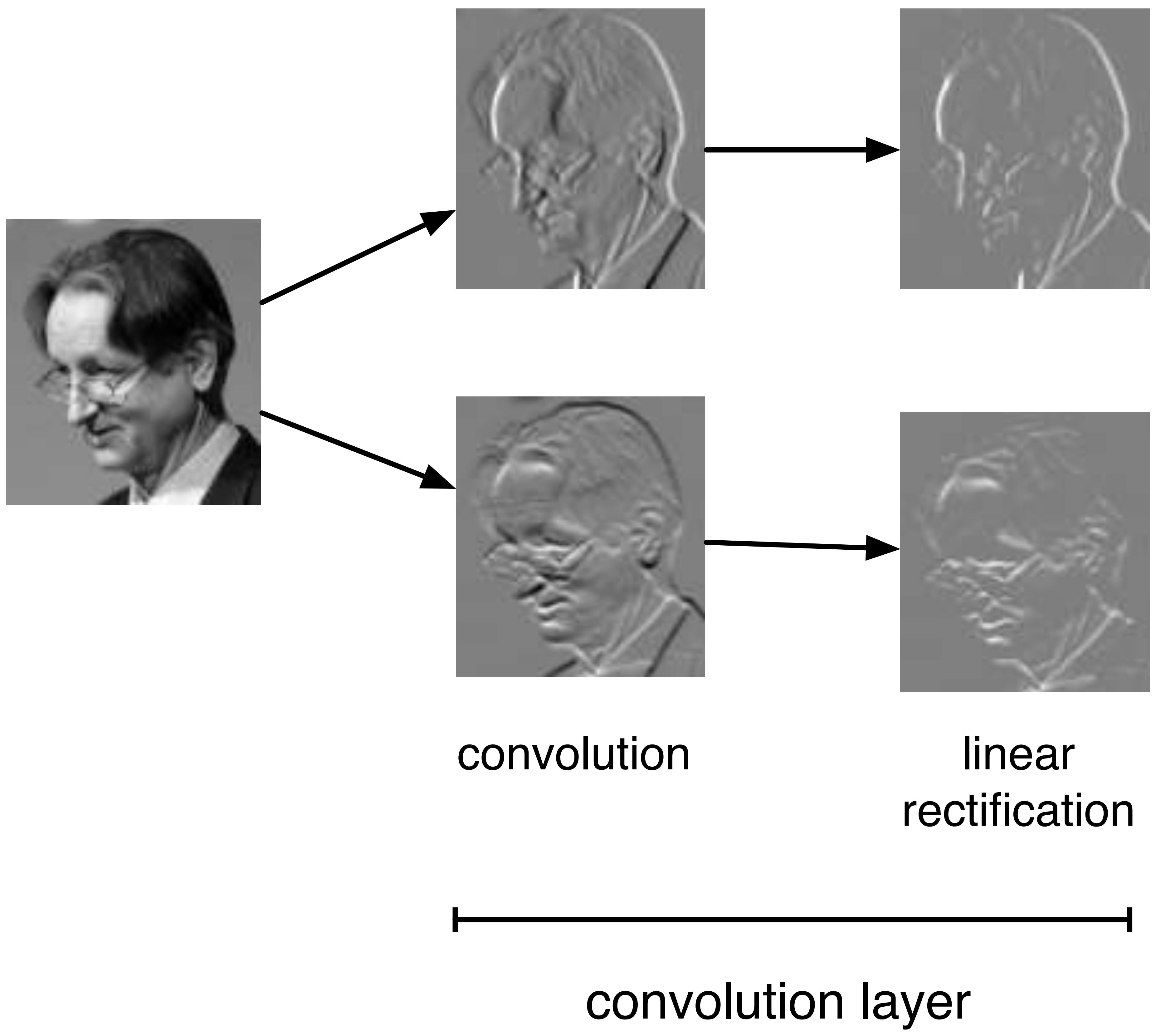
Example first-layer filters

(Zeiler and Fergus, 2013, Visualizing and understanding convolutional networks)

<—— **Highly recommended reading !**

Convolutional neural networks

- It's common to apply a linear rectification nonlinearity: $y_i = \max(z_i, 0)$

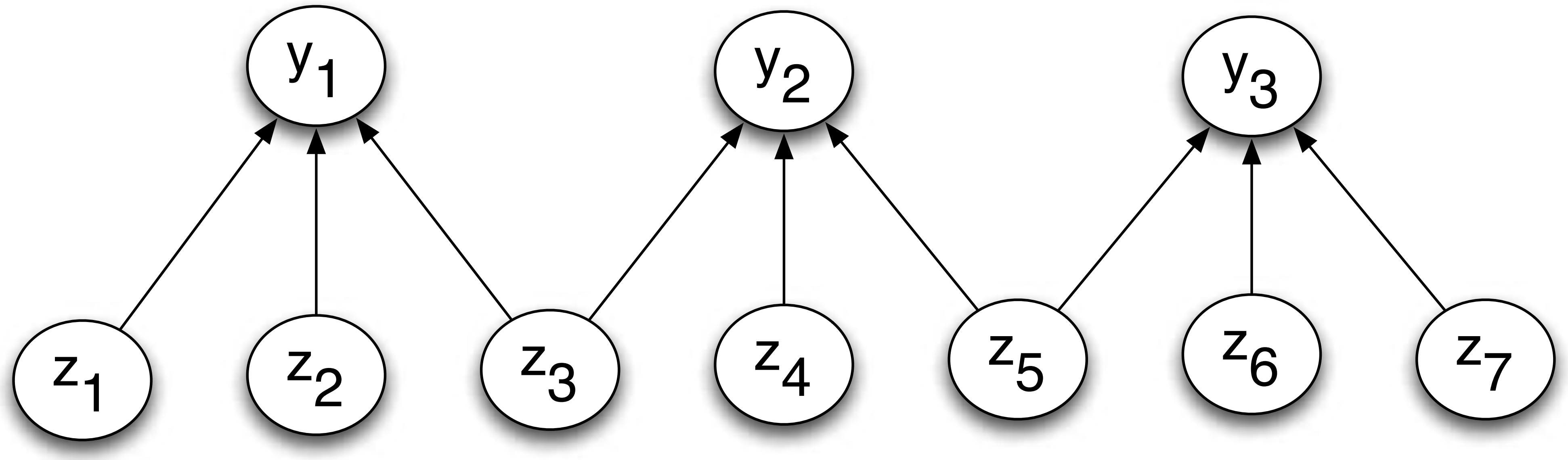


Why might we do this?

- Convolution is a linear operation. Therefore, we need a nonlinearity, otherwise 2 convolution layers would be no more powerful than 1.
- Two edges in opposite directions shouldn't cancel.

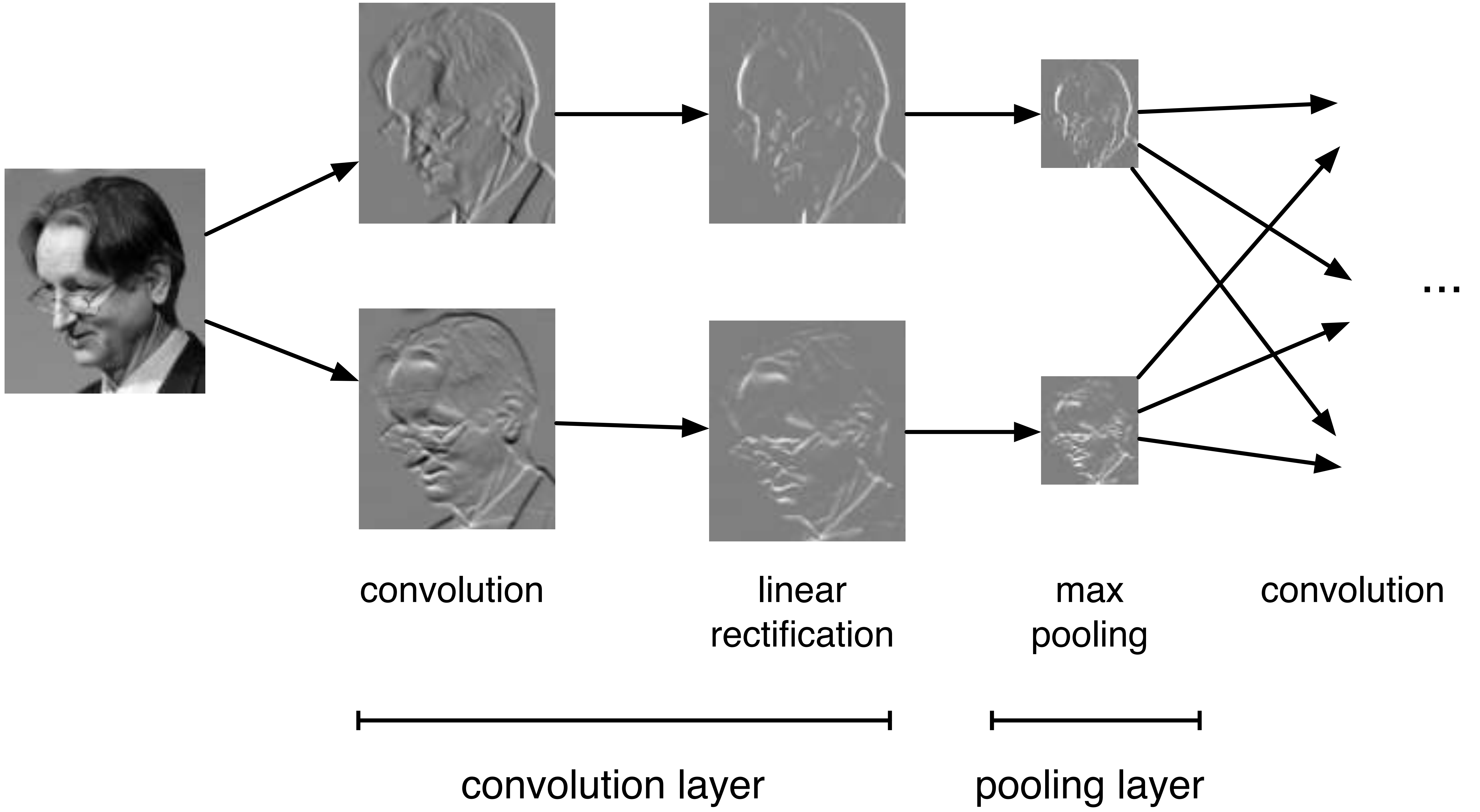
Pooling layers

- The other type of layer in a pooling layer. These layers reduce the size of the representation and build in invariance to small transformations.



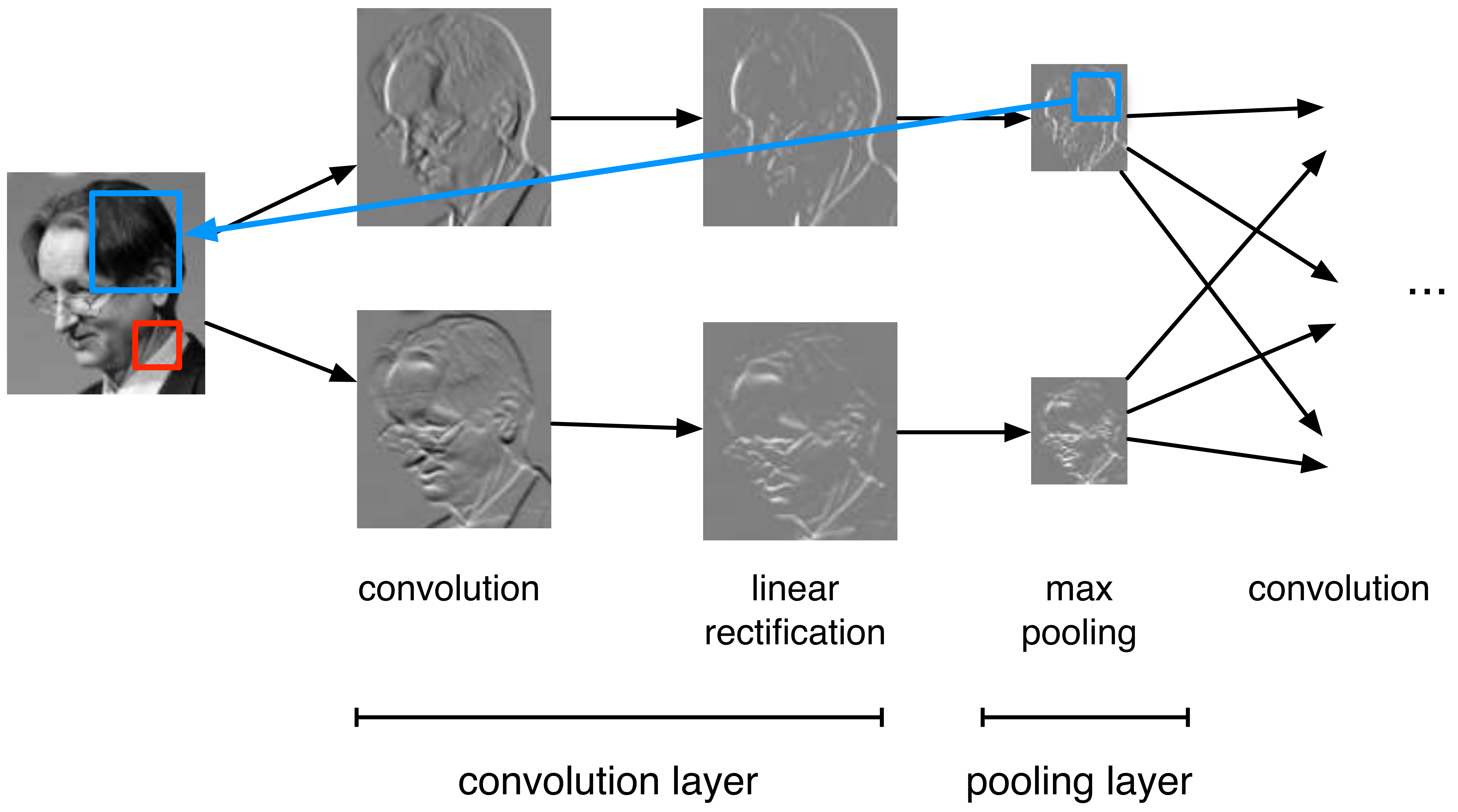
- Most commonly, we use max-pooling, which computes the maximum value of the units in a pooling group:
- $y_j = \max_j(z_j)$ with $j \in$ the pooling group

Convolutional neural networks



Convolutional neural networks

- Because of pooling, higher-layer filters can cover a larger region of the input than equal-sized filters in the lower layers.



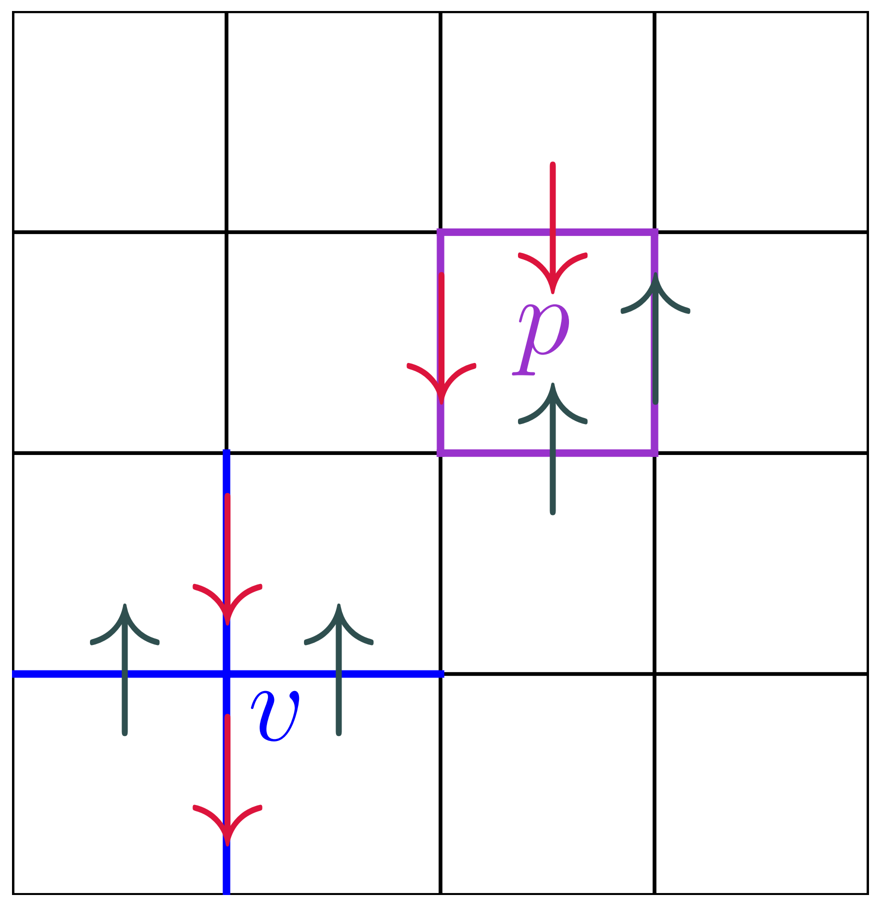
Phases of matter without an order parameter at T=0,∞

Wegner's Ising gauge theory:

$$H = -J \sum_p \prod_{i \in p} \sigma_i^z$$

F.J. Wegner, J. Math. Phys. 12 (1971) 2259

(Kogut Rev. Mod. Phys. 51, 659 (1979))



The grandmother of most lattice models for topological quantum computation

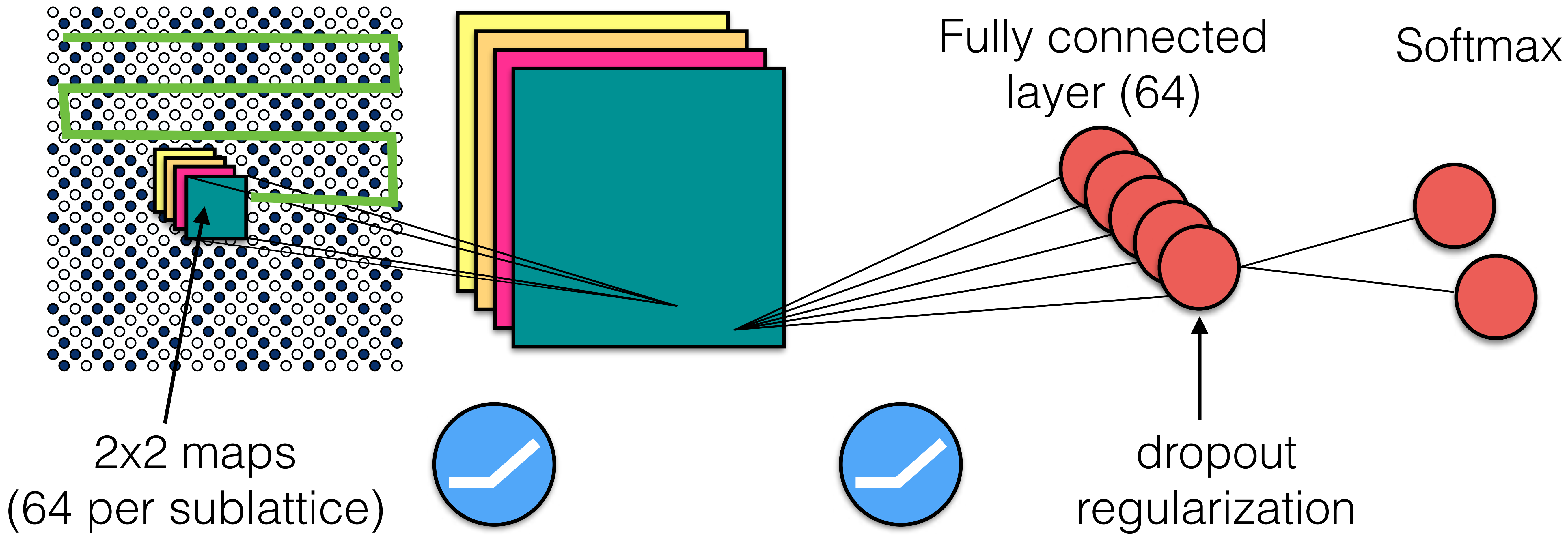
The ground state is a highly degenerate manifold with exponentially decaying spin–spin correlations. Ground state is a disordered topologically ordered phase

Castelnovo and Chamon Phys. Rev. B 76, 174416 (2007)

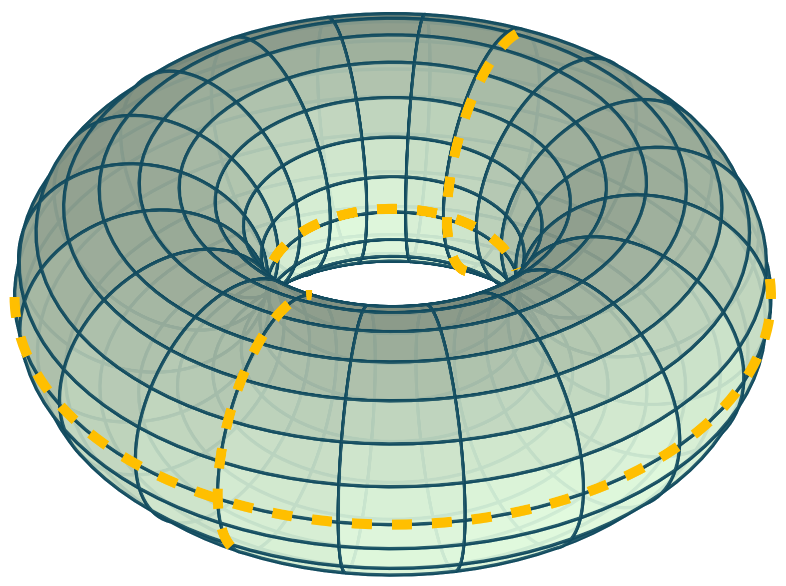
Ising gauge theory

F.J. Wegner, J. Math. Phys. 12 (1971) 2259

$$H = -J \sum_p \prod_{i \in p} \sigma_i^z$$



99% accuracy
easy to train

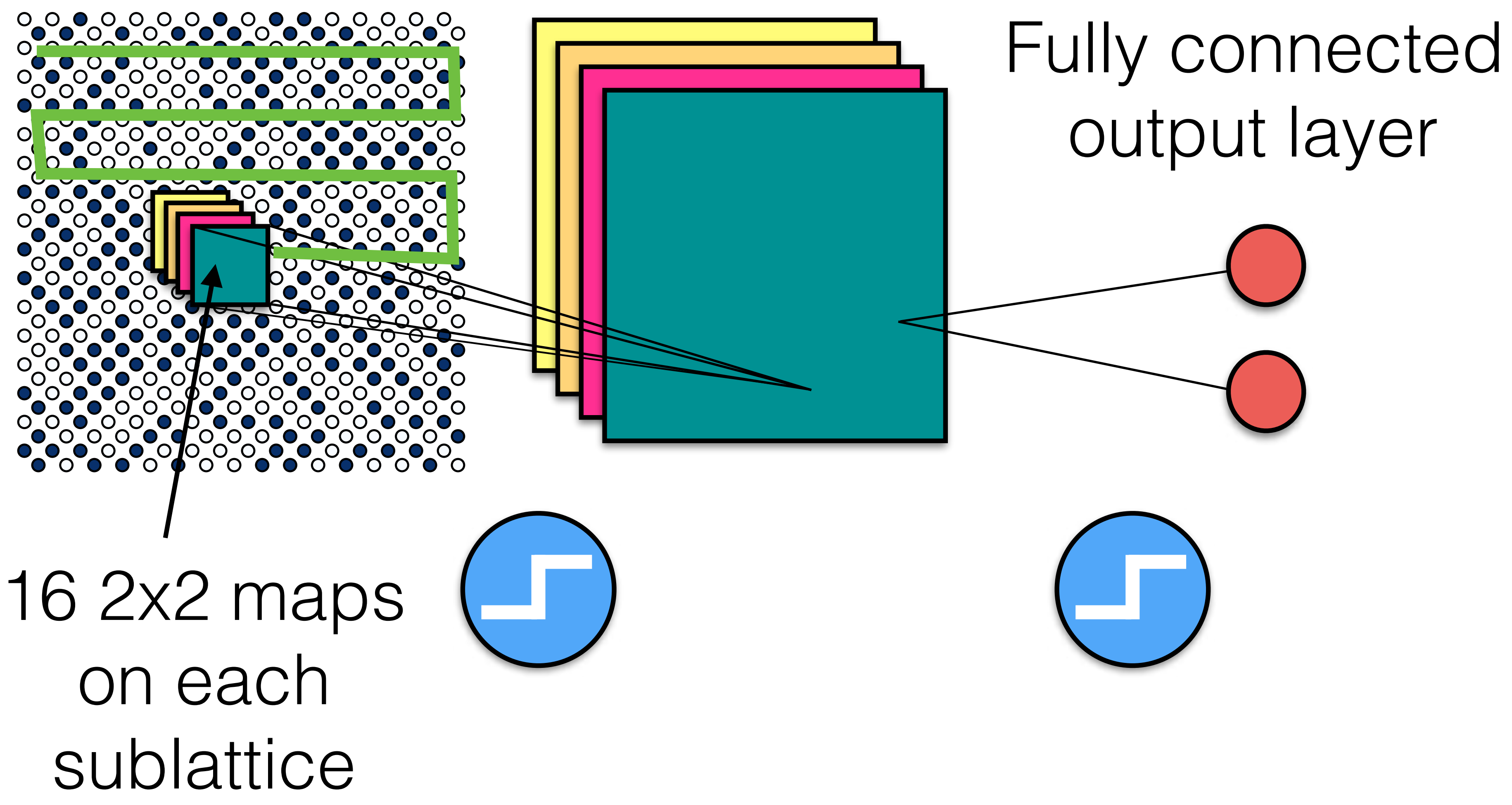


The picture we draw for what the CNN is using to distinguish the phases is that of the detection of satisfied local constraints. In few words, the neural network figures out the energy and uses it to classify states

Analytical understanding: What does the CNN use to make predictions?

- Based on this observation we derived the weights of a streamlined convolutional network analytically designed to work well on our test sets.

$$O_{\text{cold}}(\sigma_1, \ldots, \sigma_N) \propto \lim_{\beta \to \infty} \exp \beta J \sum_p \prod_{i \in p} \sigma_i^z$$



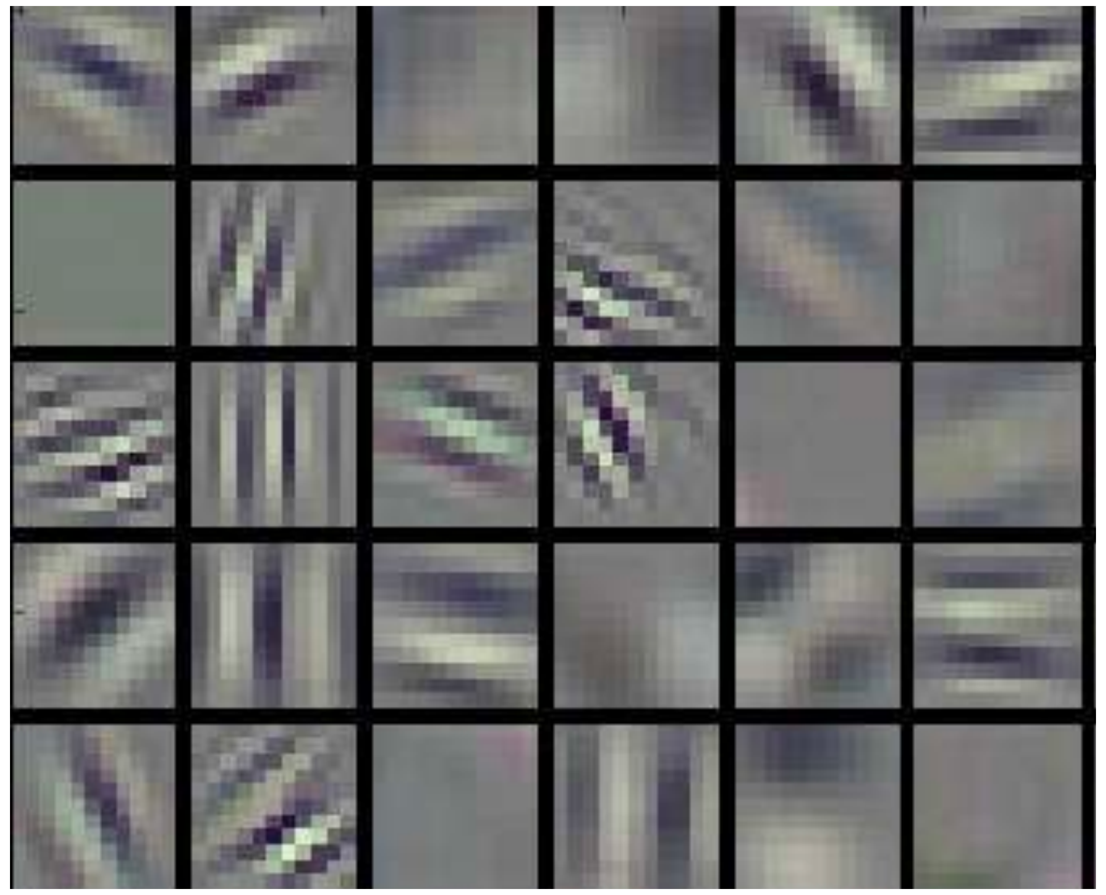
100% accuracy on test sets

Sub-lattices act as color channel in computer vision applications

J. Carrasquilla and R. G. Melko. Nature Physics 13, 431–434 (2017)

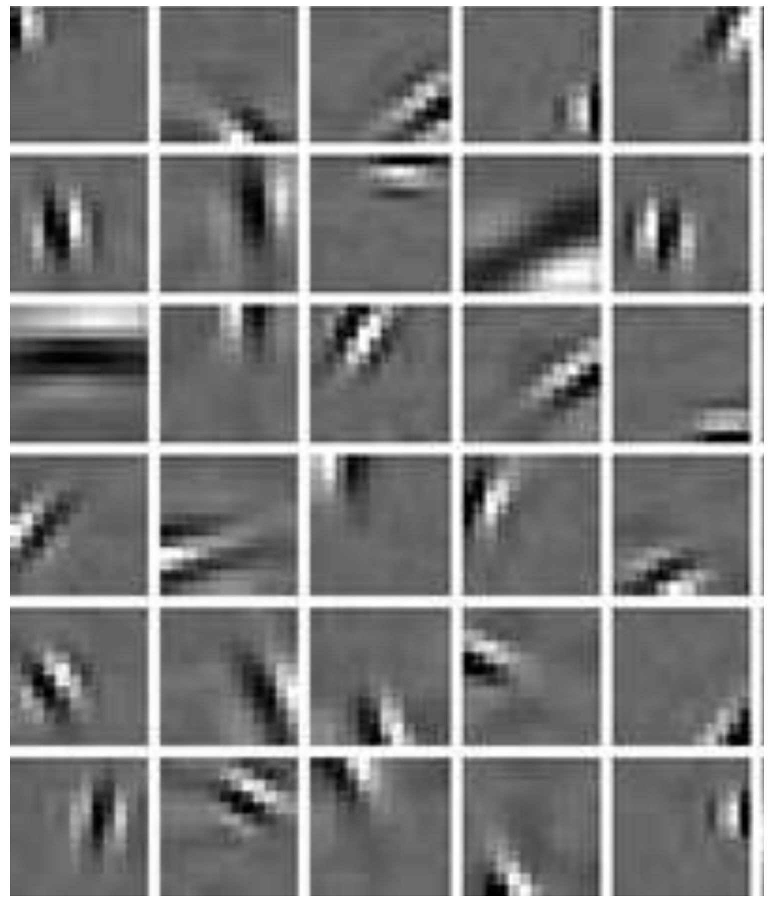
Connection with computer vision

- Convolutional neural networks revolutionized computer vision — beat humans at classifying images since 2015

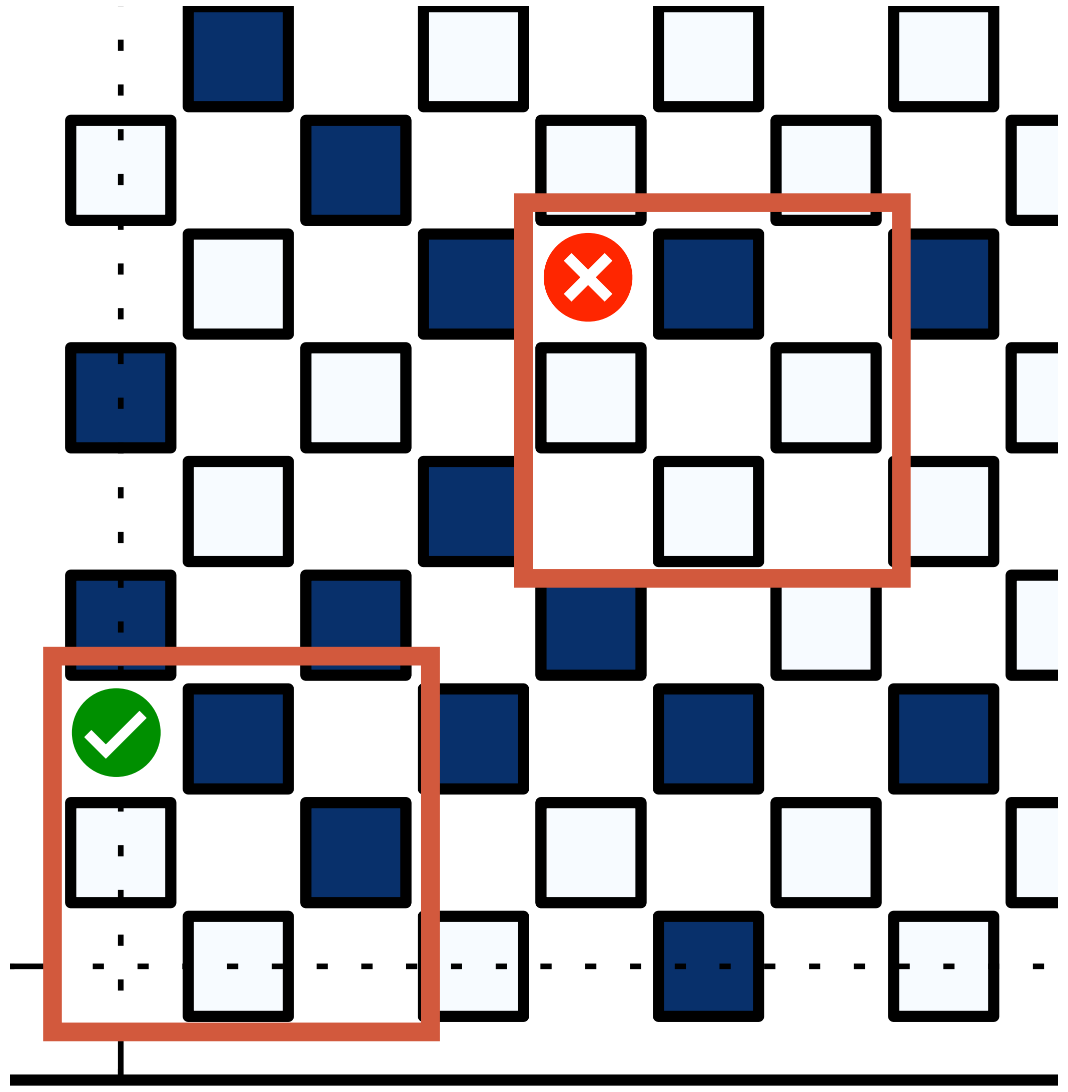


Filters used by the 1st convolutional 1ayer.

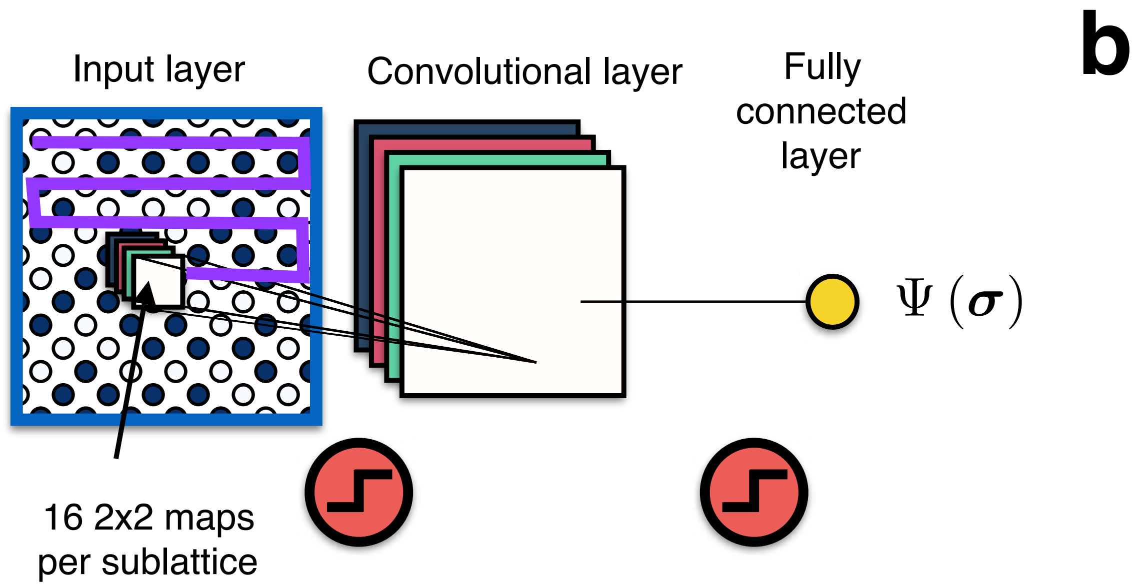
https://cs231n.github.io/understanding-cnn/



Gabor filters, believed to exist in visual processing in the brain (Olshausen & Field, 1997)



In our examples, they see magnetic monopoles or in spin ice, they see the ice rules



$$H = -J \sum_p \prod_{i \in p} \sigma_i^z$$

Analytical model for the Ising gauge theory

Convolutional layer

f	s=A	s=B	f	s=A	s=B
1	$\begin{pmatrix}1 & 0\\ 1 & 0\end{pmatrix}$	$\begin{pmatrix}1 & 1\\ 0 & 0\end{pmatrix}$	9	$\begin{pmatrix}1 & 0\\ 1 & 0\end{pmatrix}$	$\begin{pmatrix}1 & -1\\ 0 & 0\end{pmatrix}$
2	$\begin{pmatrix}-1 & 0\\ -1 & 0\end{pmatrix}$	$\begin{pmatrix}-1 & -1\\ 0 & 0\end{pmatrix}$	10	$\begin{pmatrix}1 & 0\\ 1 & 0\end{pmatrix}$	$\begin{pmatrix}-1 & 1\\ 0 & 0\end{pmatrix}$
3	$\begin{pmatrix}1 & 0\\ 1 & 0\end{pmatrix}$	$\begin{pmatrix}-1 & -1\\ 0 & 0\end{pmatrix}$	11	$\begin{pmatrix}1 & 0\\ -1 & 0\end{pmatrix}$	$\begin{pmatrix}1 & 1\\ 0 & 0\end{pmatrix}$
4	$\begin{pmatrix}1 & 0\\ -1 & 0\end{pmatrix}$	$\begin{pmatrix}-1 & 1\\ 0 & 0\end{pmatrix}$	12	$\begin{pmatrix}-1 & 0\\ 1 & 0\end{pmatrix}$	$\begin{pmatrix}1 & 1\\ 0 & 0\end{pmatrix}$
5	$\begin{pmatrix}-1 & 0\\ -1 & 0\end{pmatrix}$	$\begin{pmatrix}1 & 1\\ 0 & 0\end{pmatrix}$	13	$\begin{pmatrix}-1 & 0\\ -1 & 0\end{pmatrix}$	$\begin{pmatrix}-1 & 1\\ 0 & 0\end{pmatrix}$
6	$\begin{pmatrix}-1 & 0\\ 1 & 0\end{pmatrix}$	$\begin{pmatrix}1 & -1\\ 0 & 0\end{pmatrix}$	14	$\begin{pmatrix}-1 & 0\\ -1 & 0\end{pmatrix}$	$\begin{pmatrix}1 & -1\\ 0 & 0\end{pmatrix}$
7	$\begin{pmatrix}1 & 0\\ -1 & 0\end{pmatrix}$	$\begin{pmatrix}1 & -1\\ 0 & 0\end{pmatrix}$	15	$\begin{pmatrix}-1 & 0\\ 1 & 0\end{pmatrix}$	$\begin{pmatrix}-1 & -1\\ 0 & 0\end{pmatrix}$
8	$\begin{pmatrix}-1 & 0\\ 1 & 0\end{pmatrix}$	$\begin{pmatrix}-1 & 1\\ 0 & 0\end{pmatrix}$	16	$\begin{pmatrix}-1 & 0\\ -1 & 0\end{pmatrix}$	$\begin{pmatrix}-1 & -1\\ 0 & 0\end{pmatrix}$

$$b_c = -(2+\epsilon)\begin{bmatrix}1\\ \vdots \\ 1\end{bmatrix}$$

fully-connected layer

$$W_{\mathrm{o}} = \begin{pmatrix} \overbrace{1 \;\; \dots \;\; 1}^{8L^2 \text{ terms}} & \overbrace{-L^2 \dots - L^2}^{8L^2 \text{ terms}} \\ -1 \;\; \dots \;\; -1 & L^2 \dots L^2 \end{pmatrix}, \quad \text{and } b_{\mathrm{o}} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

The purpose of the filters is to individually process each plaquette in the spin configuration and determine whether its energetic constraints are satisfied or not. The non linearities are step functions. The last layer counts the satisfied plaquettes.

https://github.com/carrasqu/IsingGaugeTensorFlow

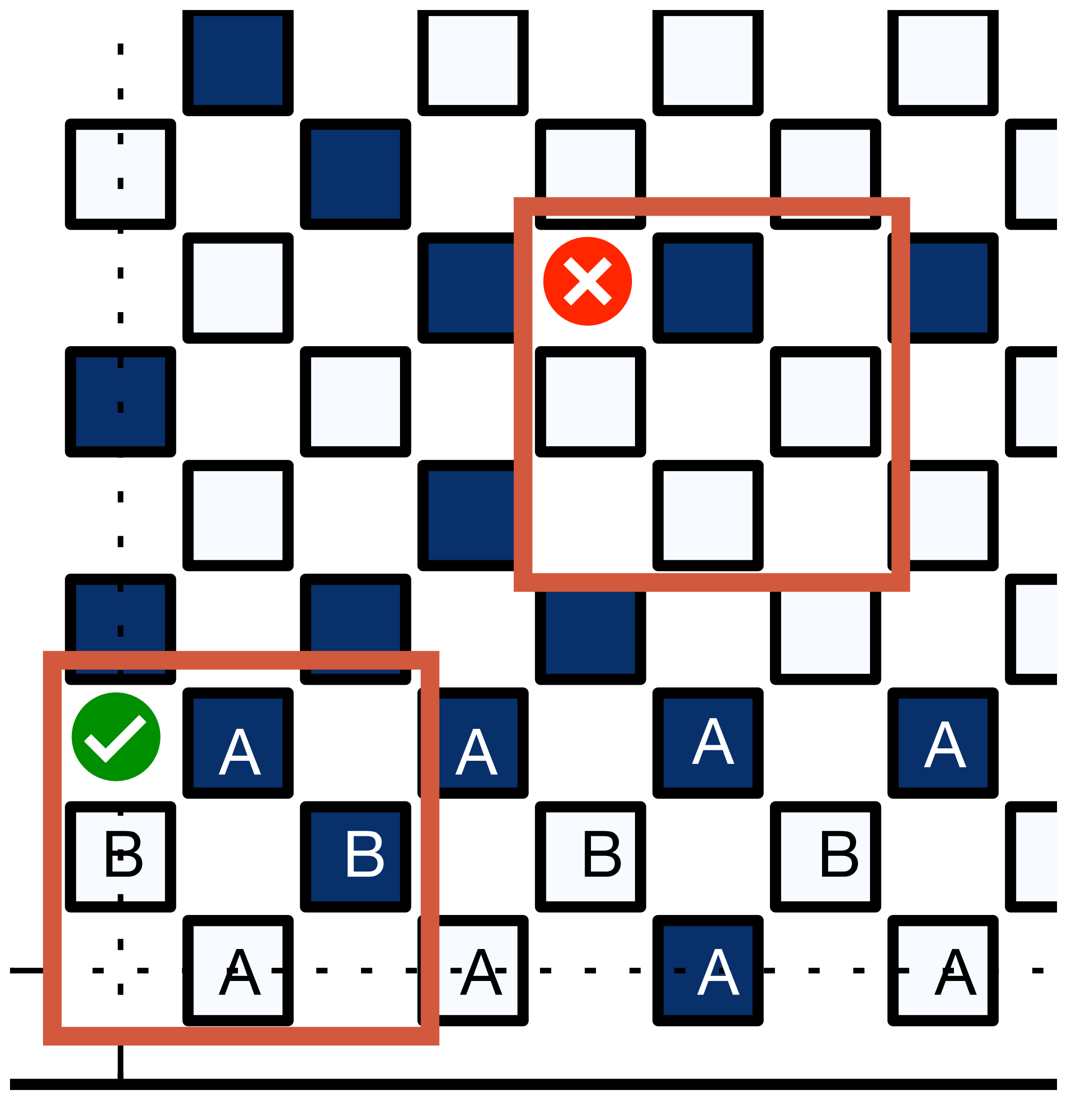
Analytical model for the Ising gauge theory

Convolutional layer

f	s=A	s=B	f	s=A	s=B
1	$\begin{pmatrix}1 & 0\\ 1 & 0\end{pmatrix}$	$\begin{pmatrix}1 & 1\\ 0 & 0\end{pmatrix}$	9	$\begin{pmatrix}1 & 0\\ 1 & 0\end{pmatrix}$	$\begin{pmatrix}1 & -1\\ 0 & 0\end{pmatrix}$
2	$\begin{pmatrix}-1 & 0\\ -1 & 0\end{pmatrix}$	$\begin{pmatrix}-1 & -1\\ 0 & 0\end{pmatrix}$	10	$\begin{pmatrix}1 & 0\\ 1 & 0\end{pmatrix}$	$\begin{pmatrix}-1 & 1\\ 0 & 0\end{pmatrix}$
3	$\begin{pmatrix}1 & 0\\ 1 & 0\end{pmatrix}$	$\begin{pmatrix}-1 & -1\\ 0 & 0\end{pmatrix}$	11	$\begin{pmatrix}1 & 0\\ -1 & 0\end{pmatrix}$	$\begin{pmatrix}1 & 1\\ 0 & 0\end{pmatrix}$
4	$\begin{pmatrix}1 & 0\\ -1 & 0\end{pmatrix}$	$\begin{pmatrix}-1 & 1\\ 0 & 0\end{pmatrix}$	12	$\begin{pmatrix}-1 & 0\\ 1 & 0\end{pmatrix}$	$\begin{pmatrix}1 & 1\\ 0 & 0\end{pmatrix}$
5	$\begin{pmatrix}-1 & 0\\ -1 & 0\end{pmatrix}$	$\begin{pmatrix}1 & 1\\ 0 & 0\end{pmatrix}$	13	$\begin{pmatrix}-1 & 0\\ -1 & 0\end{pmatrix}$	$\begin{pmatrix}-1 & 1\\ 0 & 0\end{pmatrix}$
6	$\begin{pmatrix}-1 & 0\\ 1 & 0\end{pmatrix}$	$\begin{pmatrix}1 & -1\\ 0 & 0\end{pmatrix}$	14	$\begin{pmatrix}-1 & 0\\ -1 & 0\end{pmatrix}$	$\begin{pmatrix}1 & -1\\ 0 & 0\end{pmatrix}$
7	$\begin{pmatrix}1 & 0\\ -1 & 0\end{pmatrix}$	$\begin{pmatrix}1 & -1\\ 0 & 0\end{pmatrix}$	15	$\begin{pmatrix}-1 & 0\\ 1 & 0\end{pmatrix}$	$\begin{pmatrix}-1 & -1\\ 0 & 0\end{pmatrix}$
8	$\begin{pmatrix}-1 & 0\\ 1 & 0\end{pmatrix}$	$\begin{pmatrix}-1 & 1\\ 0 & 0\end{pmatrix}$	16	$\begin{pmatrix}-1 & 0\\ -1 & 0\end{pmatrix}$	$\begin{pmatrix}-1 & -1\\ 0 & 0\end{pmatrix}$

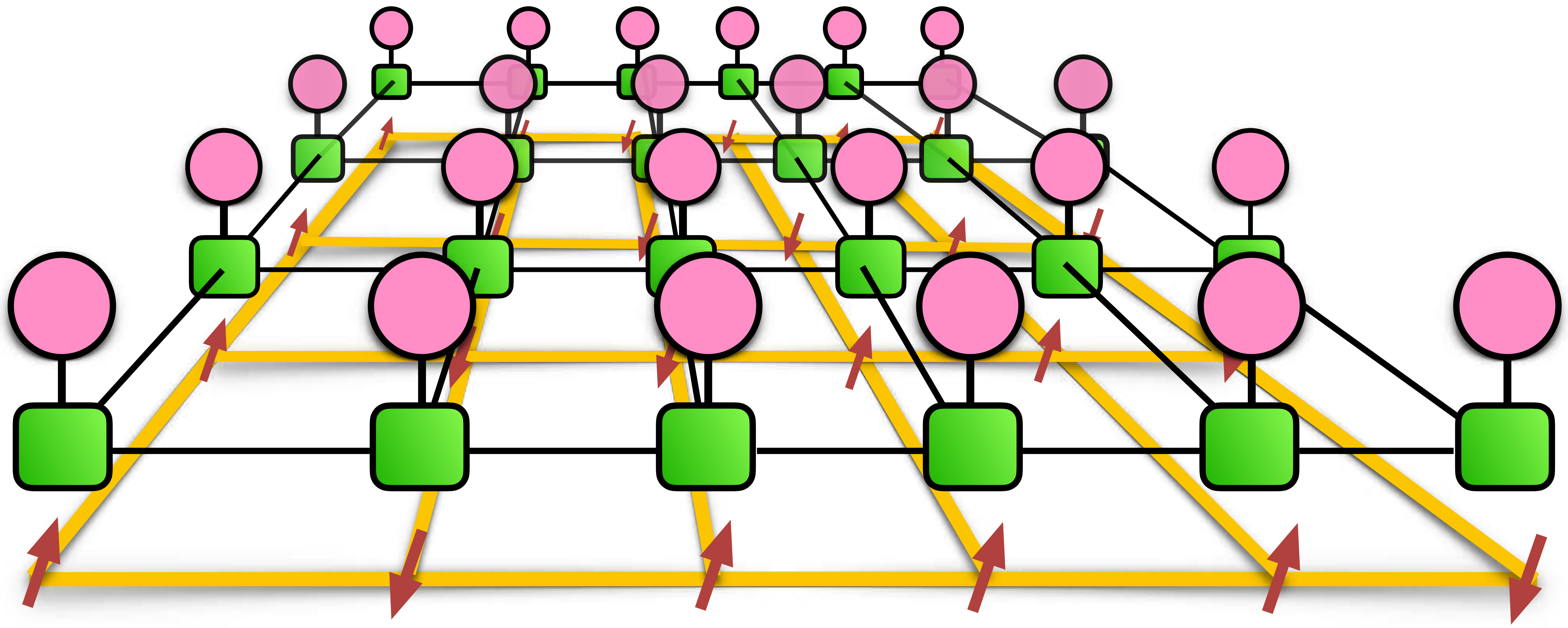
$$b_c = -(2+\epsilon)\begin{bmatrix}1\\ \vdots\\ 1\end{bmatrix}$$

fully-connected layer

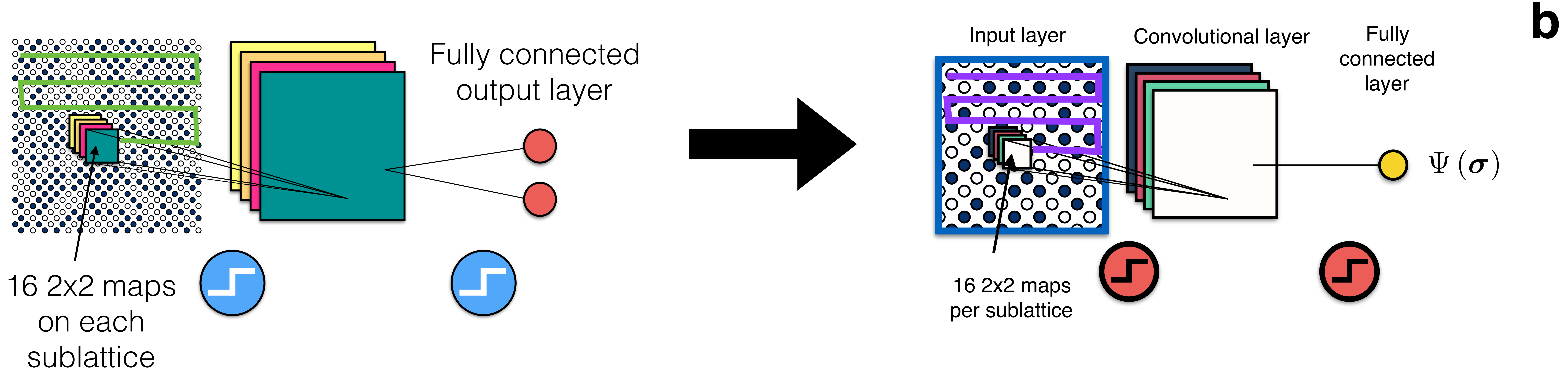


The purpose of the filters is to individually process each plaquette in the spin configuration and determine whether its energetic constraints are satisfied or not. Basically the Conv. layer encodes the Hamiltonian

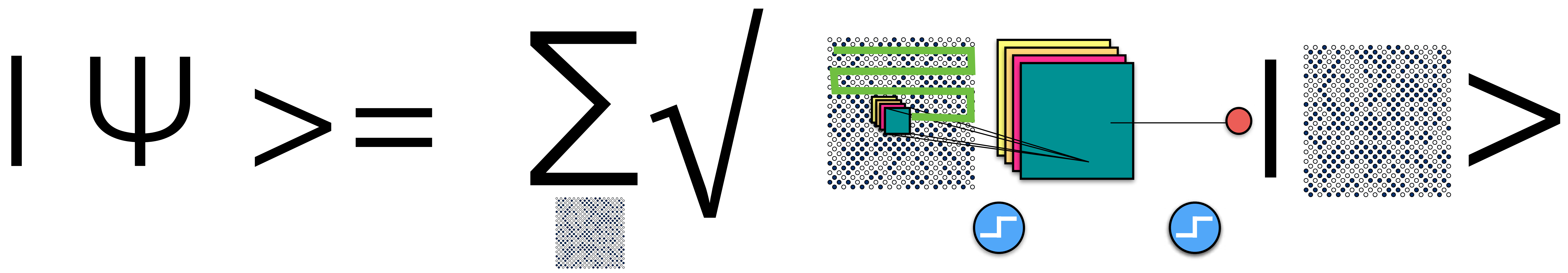
Neural networks as quantum states



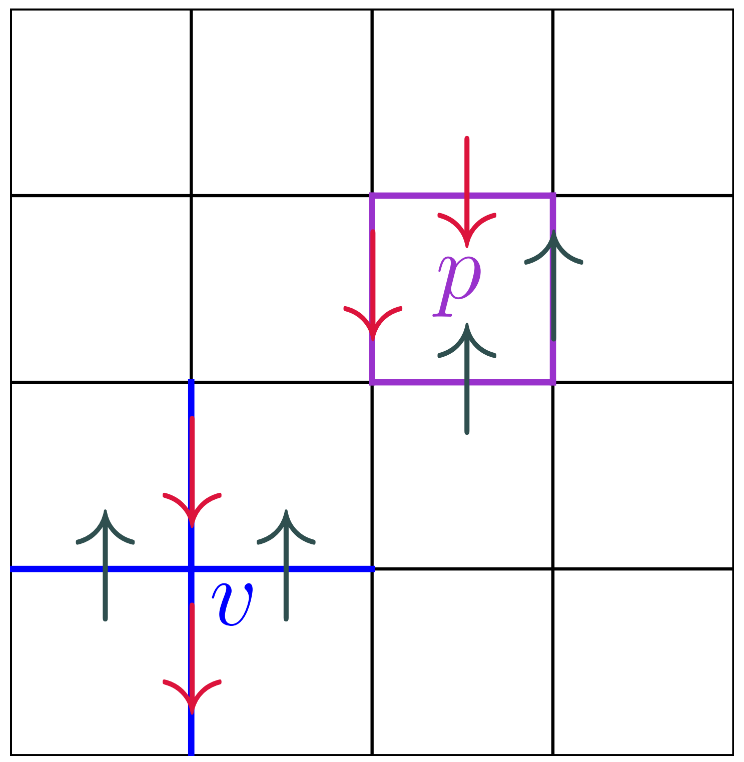
Remove the cold neuron



Cold neuron in our model: Ground state of Kitaev's toric code with convolutional neural networks



Ground state of the toric code

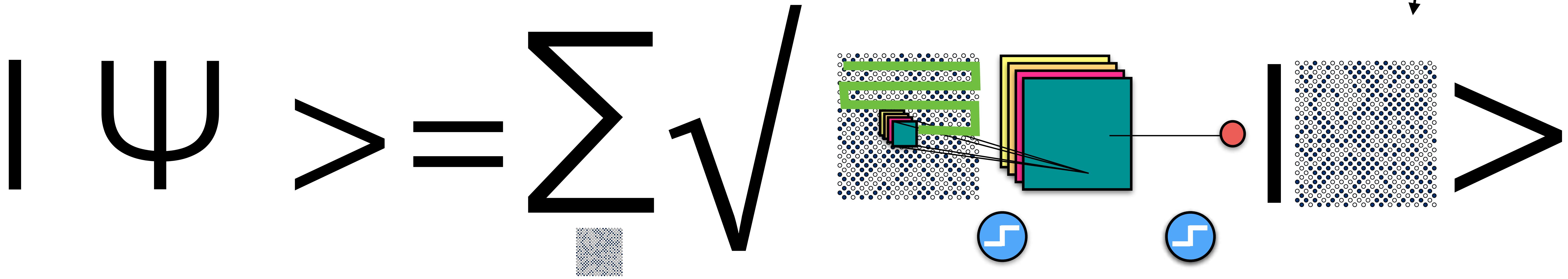


$$H = -J_p \sum_p \prod_{i \in p} \sigma_i^z - J_v \sum_v \prod_{i \in v} \sigma_i^x$$

$$|\Psi_{\mathrm{TC}}\rangle \propto \lim_{\beta \to \infty} \sum_{\sigma_1, \ldots, \sigma_N} e^{\frac{\beta}{2} J \sum_p \prod_{i \in p} \sigma_i^z} |\sigma_1, \ldots, \sigma_N\rangle$$

PEPS : *F. Verstraete, M. M. Wolf, D. Perez-Garcia, J. I. Cirac Phys. Rev. Lett. 96, 220601 (2006).*

$$O_{\mathrm{cold}}(\sigma_1, \ldots, \sigma_N) \propto \lim_{\beta \to \infty} \exp \beta J \sum_p \prod_{i \in p} \sigma_i^z$$



J. Carrasquilla and R. G. Melko. Nature Physics 13, 431–434 (2017)
Dong-Ling Deng et al Phys. Rev. X 7, 021021 (2017)
Jing Chen, Song Cheng, Haidong Xie, Lei Wang, Tao Xiang arXiv:1701.04831 RBMs

Neural network quantum states

ELSEVIER

Computer Physics Communications 104 (1997) 1–14

Computer Physics Communications

Artificial neural network methods in quantum mechanics

I.E. Lagaris[1], A. Likas, D.I. Fotiadis

Department of Computer Science, University of Ioannina, P.O. Box 1186, GR 45110 Ioannina, Greece

Received 17 March 1997; revised 22 April 1997

3.5. Two-dimensional Schrödinger equation

We consider here the well-studied [2] example of the Henon–Heiles potential. The Hamiltonian is written as

$$H = -\frac{1}{2}\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right) + V(x,y)\,,$$

with $V(x,y) = \frac{1}{2}(x^2+y^2) + \frac{1}{4\sqrt{5}}(xy^2 - \frac{1}{3}x^3)$.

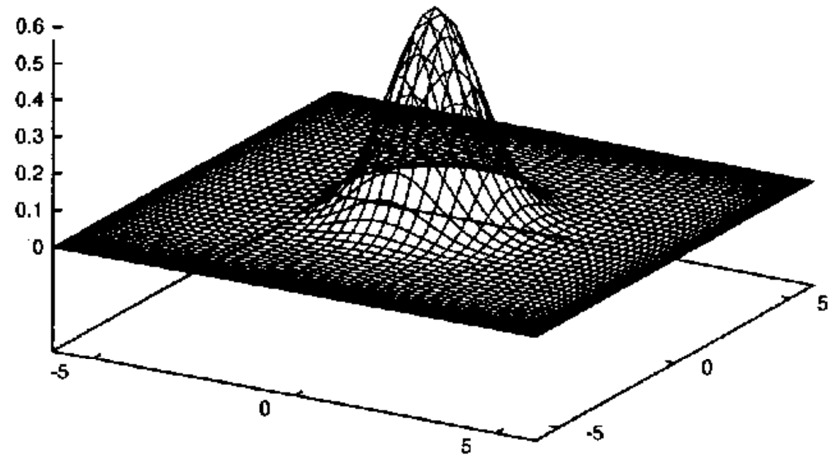


Fig. 4.

Fig. 4. Ground state of the Henon–Heiles problem (ϵ = 0.99866).

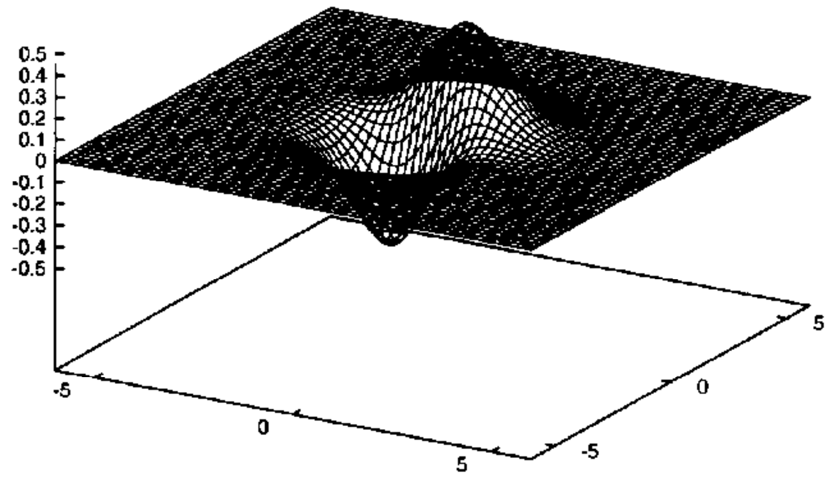
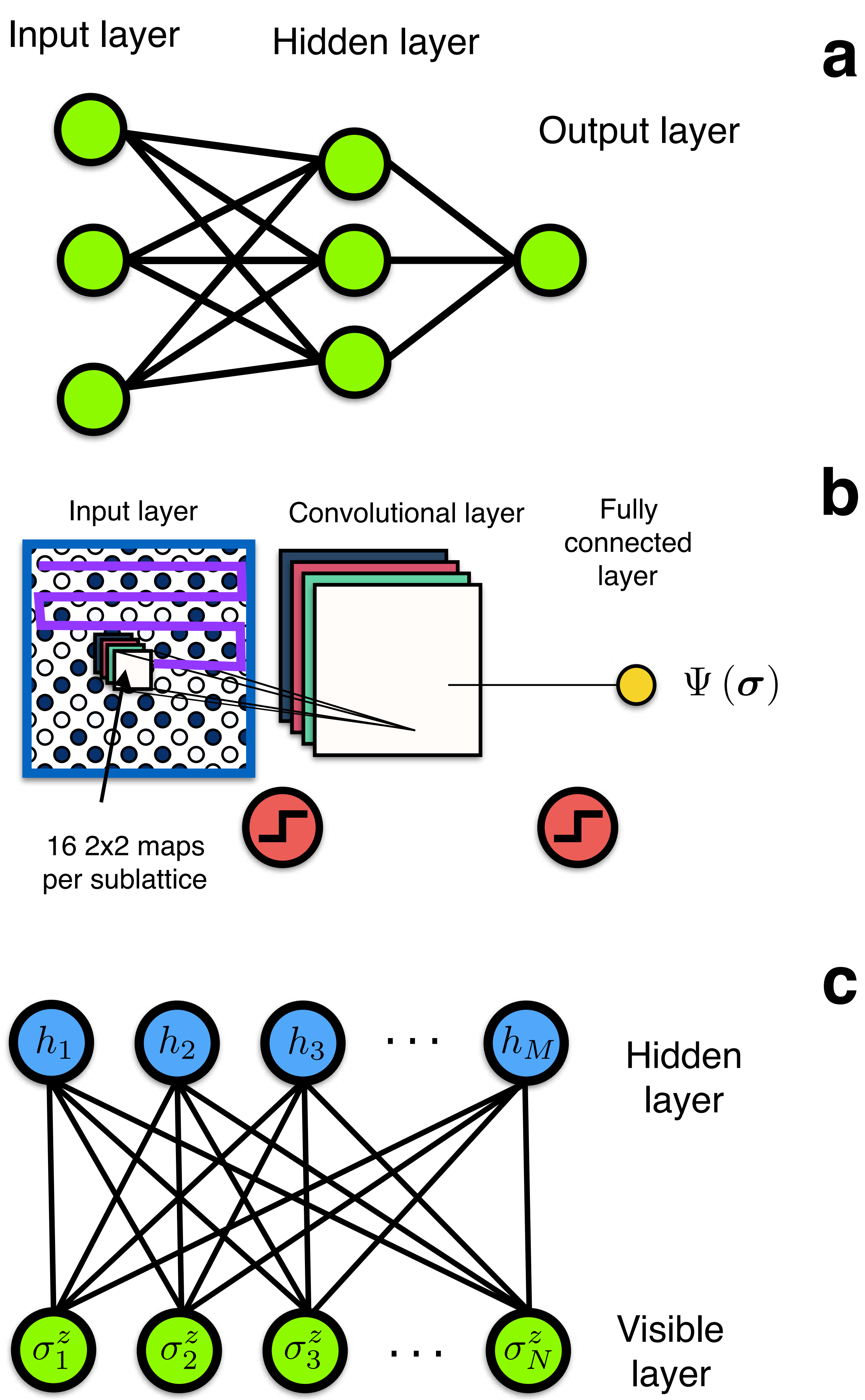


Fig. 5.



Neural network quantum states

nature physics

LETTERS

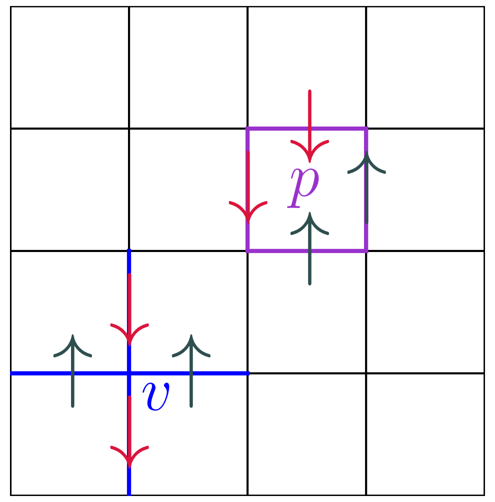
PUBLISHED ONLINE: 13 FEBRUARY 2017 | DOI: 10.1038/NPHYS4035

Machine learning phases of matter

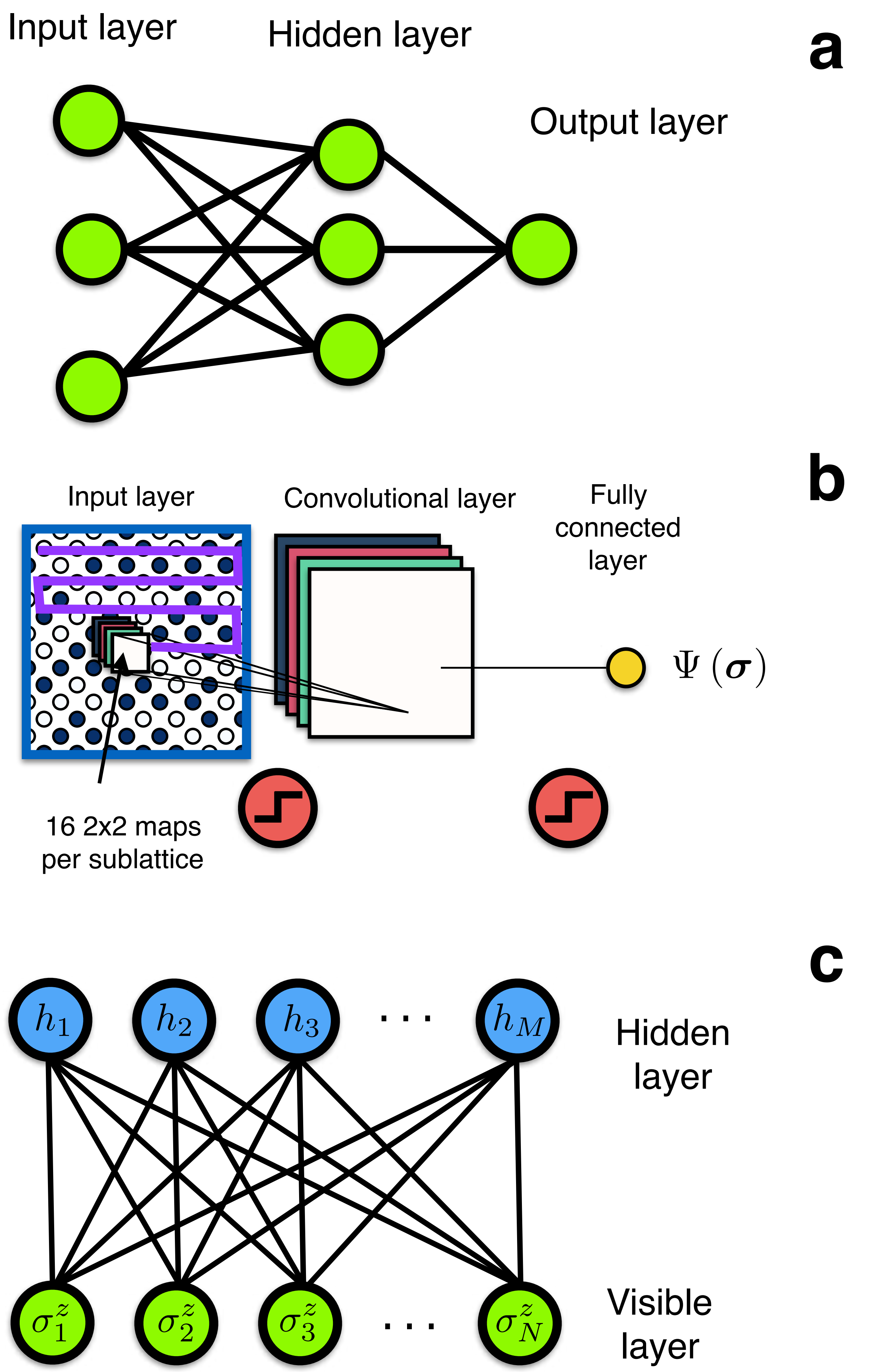
Juan Carrasquilla[1]* and Roger G. Melko[1,2]

KITAEV'S TORIC CODE GROUND STATE

$$H = -J_p \sum_p \prod_{i \in p} \sigma_i^z - J_v \sum_v \prod_{i \in v} \sigma_i^x$$



$$|\Psi_{\mathrm{TC}}\rangle \propto \lim_{\beta \to \infty} \sum_{\sigma_1,\ldots,\sigma_N} e^{\frac{\beta}{2} J \sum_p \prod_{i \in p} \sigma_i^z} |\sigma_1, \ldots, \sigma_N\rangle$$



J. Carrasquilla and R. G. Melko. Nature Physics 13, 431–434 (2017)

Neural network quantum states

RESEARCH

RESEARCH ARTICLE

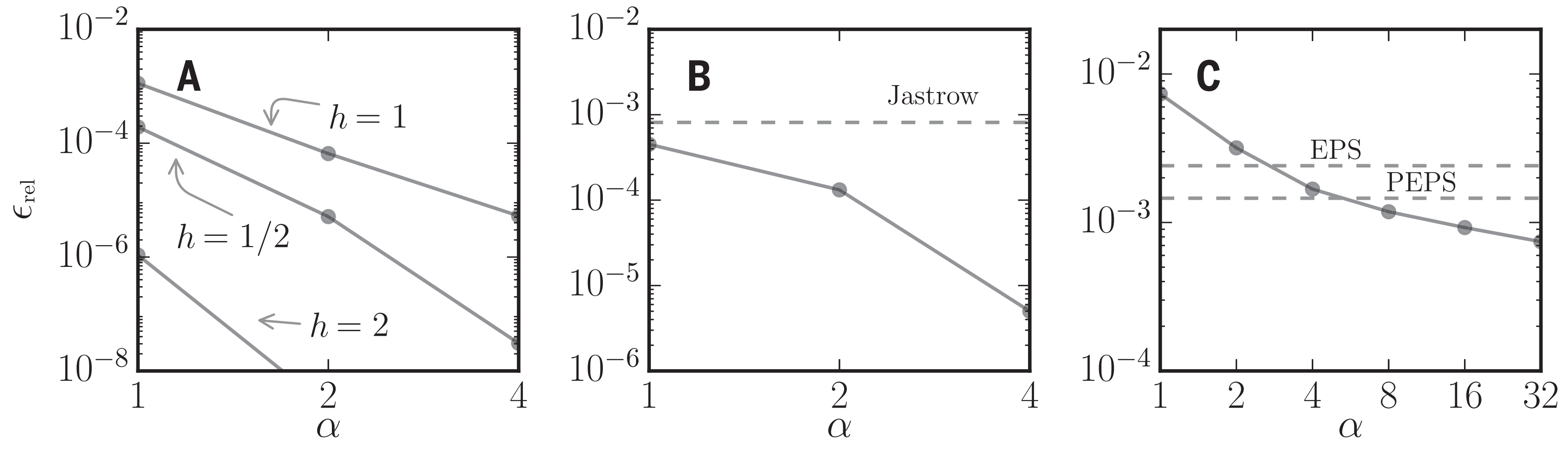
MANY-BODY PHYSICS

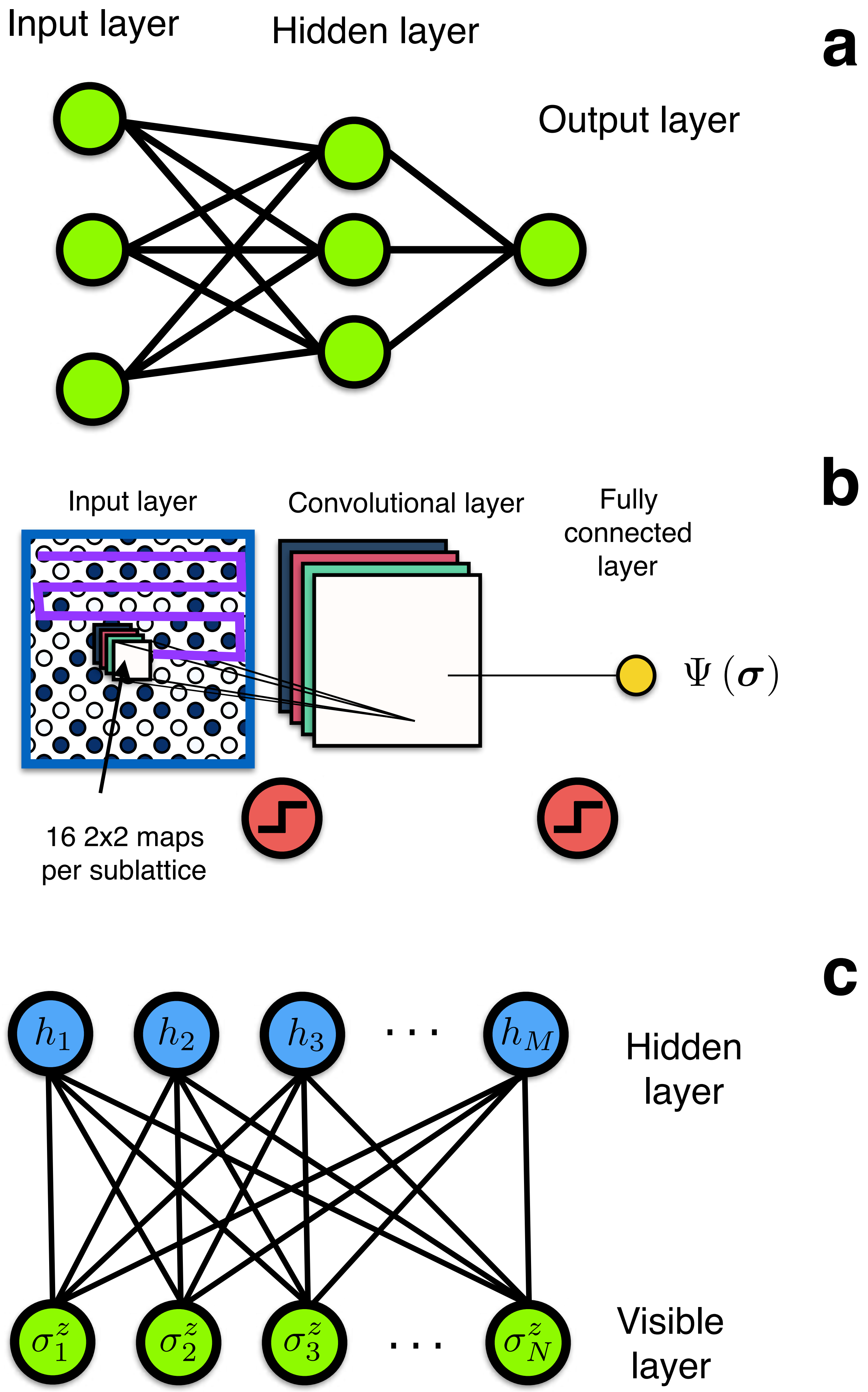
Solving the quantum many-body problem with artificial neural networks

Giuseppe Carleo[1*] and Matthias Troyer[1,2]

The challenge posed by the many-body problem in quantum physics originates from the difficulty of describing the nontrivial correlations encoded in the exponential complexity of the many-body wave function. Here we demonstrate that systematic machine learning of the wave function can reduce this complexity to a tractable computational form for some notable cases of physical interest. We introduce a variational representation of quantum states based on artificial neural networks with a variable number of hidden neurons. A reinforcement-learning scheme we demonstrate is capable of both finding the ground state and describing the unitary time evolution of complex interacting quantum systems. Our approach achieves high accuracy in describing prototypical interacting spins models in one and two dimensions.

$$\mathcal{H}_{\rm TFI} = -h\sum_i \sigma_i^x - \sum_{ij} \sigma_i^z \sigma_j^z \qquad \mathcal{H}_{\rm AFH} = \sum_{ij} \sigma_i^x\sigma_j^x + \sigma_i^y\sigma_j^y + \sigma_i^z\sigma_j^z$$





Giuseppe Carleo, Matthias Troyer, Science 355, 602 (2017)

Useful references

- Topological phases and quantum computation. Alexei Kitaev, Chris Laumann. https://arxiv.org/abs/0904.2771
- https://topocondmat.org/w12_manybody/topoorder.html
- Machine learning phases of matter. Juan Carrasquilla, Roger Melko. Nature Physics volume 13, pages 431–434 (2017)

Exploration areas

- Condensed matter physics
- Quantum chemistry
- Materials science
- Atomic physics
- High energy physics and field theory
- Quantum information
- Nuclear physics
- Combinatorial optimization

Juan Carrasquilla (2020) Machine learning for quantum matter, Advances in Physics: X, 5:1, DOI: 10.1080/23746149.2020.1797528

Juan Carrasquilla and Giacomo Torlai. Neural networks in quantum many-body physics: a hands-on tutorial. https://arxiv.org/abs/2101.11099

Giuseppe Carleo, Ignacio Cirac, Kyle Cranmer, Laurent Daudet, Maria Schuld, Naftali Tishby, Leslie Vogt-Maranto, and Lenka Zdeborová. Machine learning and the physical sciences*. Rev. Mod. Phys. 91, 045002 (2019)

Questions?

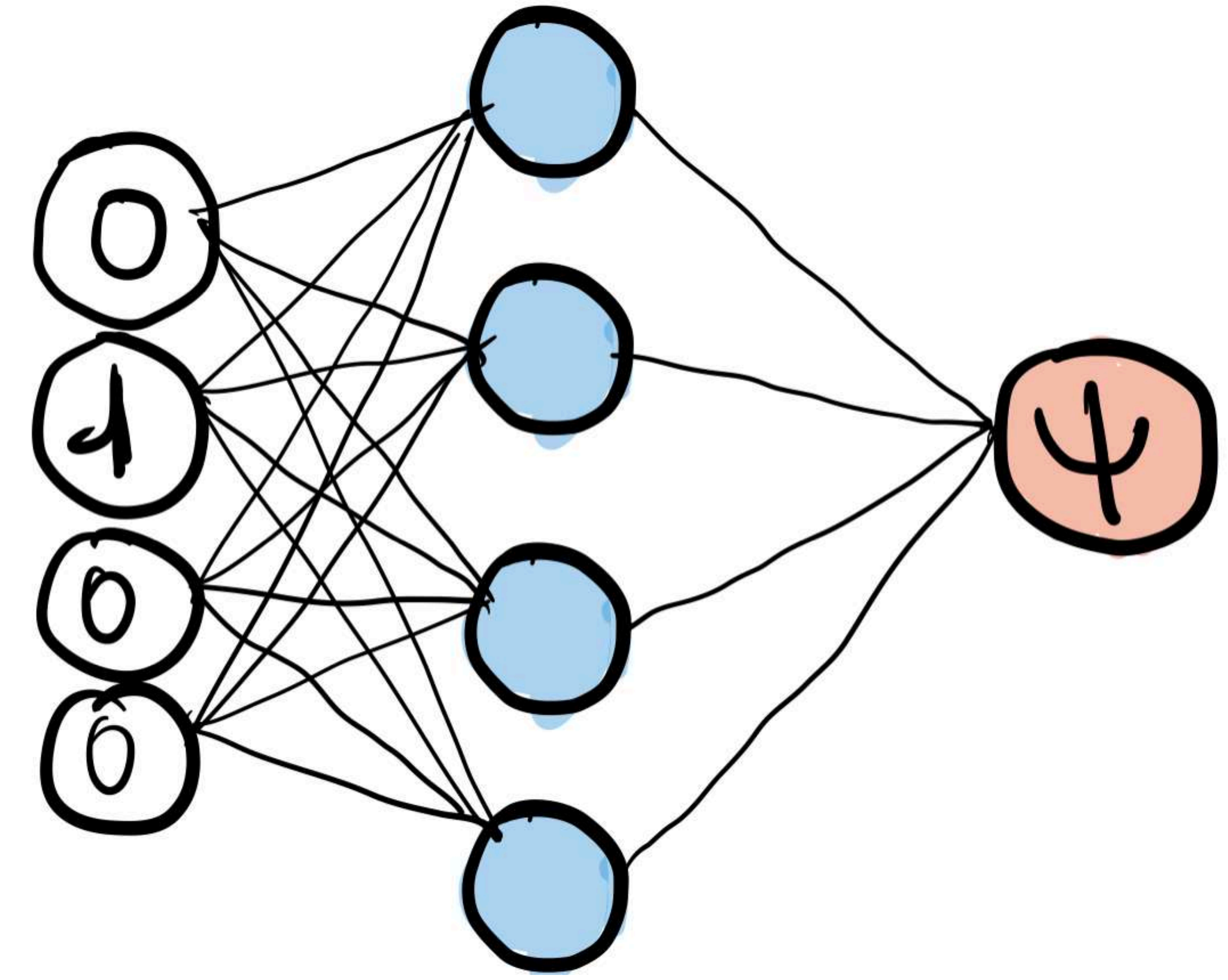
Numerical approach based on Variational Monte Carlo

- Ground state search reframed as an optimization problem with an appropriate cost function.
- The variational theorem in quantum physics $E_\theta = \langle \Psi_\theta | H | \Psi_\theta \rangle \geq E_0$ where E_0 is the lowest energy eigenvalue of Hamiltonian matrix H.
- $|\Psi_\theta\rangle$ is a neural network parameterizing the quantum state.
- It is possible to evaluate E_θ and its gradients $\nabla_\theta E_\theta$ via Monte Carlo — use gradient descent techniques
- No data from the exact solution is needed— gradient signal comes from H

F Becca, S Sorella. Quantum Monte Carlo Approaches for Correlated Systems. Cambridge University Press (2017)

Numerical approach based on Variational Monte Carlo

- Ground state search reframed s an optimization problem with an appropriate cost function.
- The variational theorem in quantum physics $E_\theta = \langle \Psi_\theta | H | \Psi_\theta \rangle \geq E_0$ where E_0 is the lowest energy eigenvalue corresponding to the ground
matrix H.
- $|\Psi_\theta\rangle$ is a neural network parameterizing the quantum state
- It is possible to evaluate E_θ and its gradients $\nabla_\theta E_\theta$ via Mo
descent techniques
- No data from the exact solution is needed— gradient signa

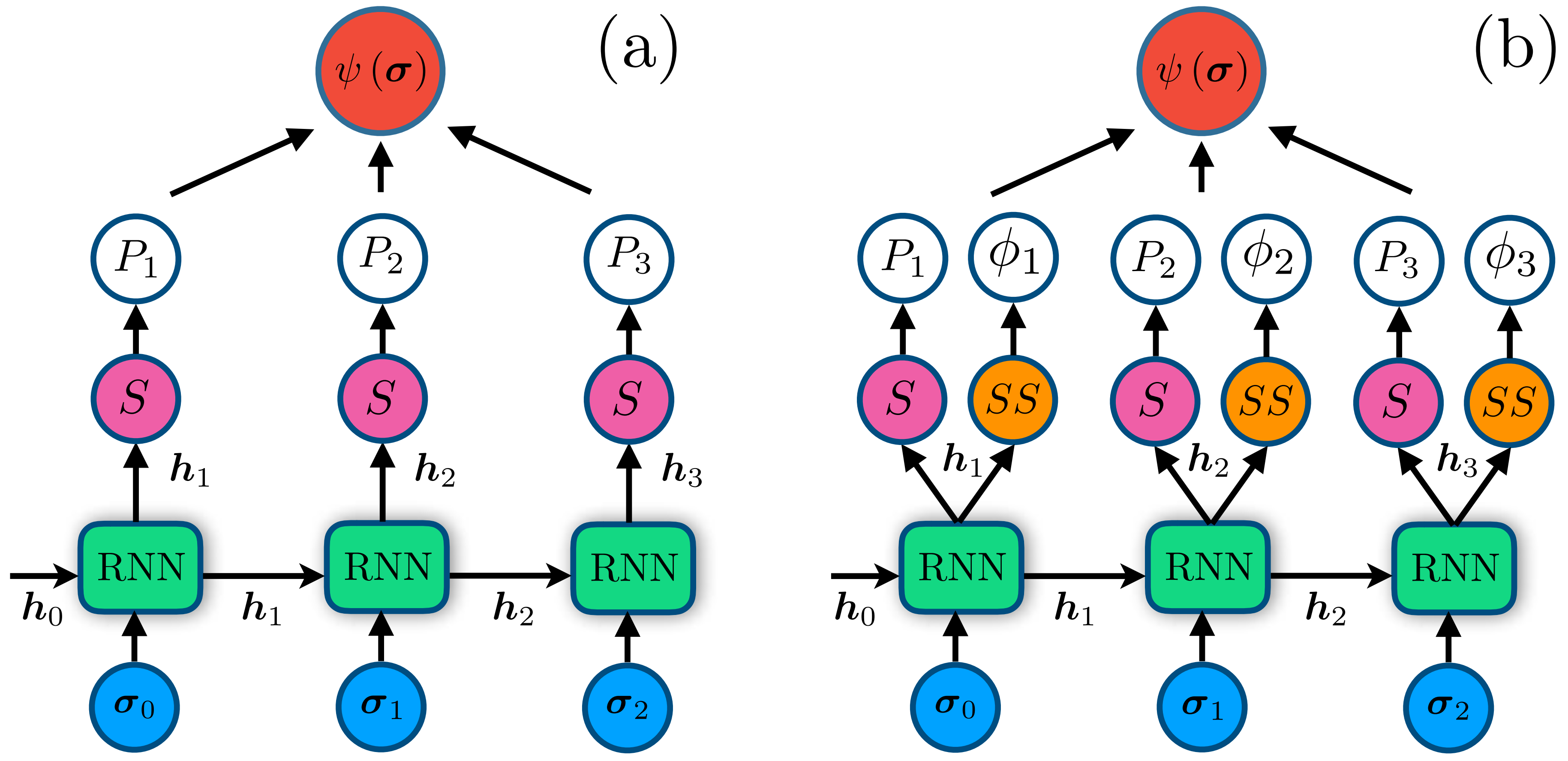


F Becca, S Sorella. Quantum Monte Carlo Approaches for Correlated Systems. Cambridge University Press (2017)

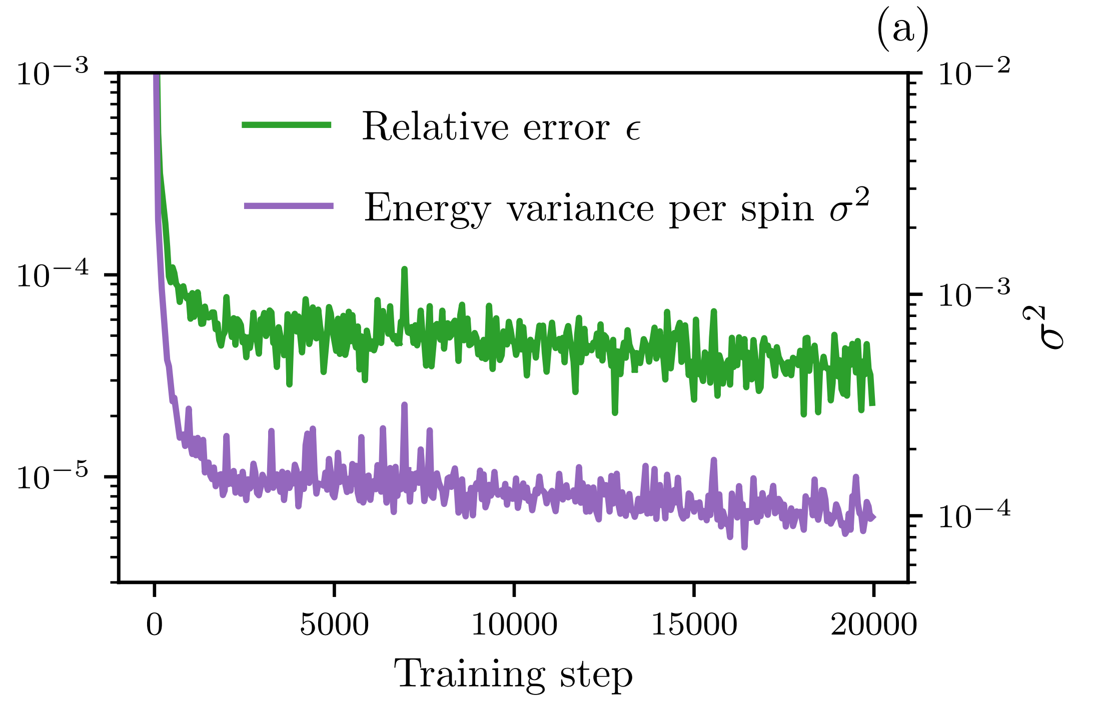
Training the models

- Both E_θ and its gradients available through sampling.
- $E = \langle \Psi_\theta | \hat{H} | \Psi_\theta \rangle = \sum_{\boldsymbol{\sigma}} |\psi_\theta(\boldsymbol{\sigma})|^2 \sum_{\boldsymbol{\sigma}'} H_{\boldsymbol{\sigma}\boldsymbol{\sigma}'} \frac{\psi_\theta(\boldsymbol{\sigma}')}{\psi_\theta(\boldsymbol{\sigma})}$ We interpret $|\psi_\theta(\boldsymbol{\sigma})|^2$ as a probability distribution
- $\equiv \sum_{\boldsymbol{\sigma}} |\psi_\theta(\boldsymbol{\sigma})|^2 E_{loc}(\boldsymbol{\sigma}) \approx \frac{1}{N_S} \sum_{\boldsymbol{\sigma} \sim |\psi_\theta(\boldsymbol{\sigma})|^2} E_{loc}(\boldsymbol{\sigma})$ Evaluate via Monte Carlo average
- Gradients
- $\partial_{\theta_j} E = \sum_{\boldsymbol{\sigma}} |\psi_\theta(\boldsymbol{\sigma})|^2 \frac{\partial_{\theta_j} \psi_\theta^*(\boldsymbol{\sigma})}{\psi_\theta^*(\boldsymbol{\sigma})} E_{loc}(\boldsymbol{\sigma}) + \text{c.c}$
- $\partial_{\theta_j} E \approx \frac{2}{N_S} \mathfrak{Re} \left(\sum_{i=1}^{N_S} \frac{\partial_{\theta_j} \psi_\theta^*(\boldsymbol{\sigma}^{(i)})}{\psi_\theta^*(\boldsymbol{\sigma}^{(i)})} E_{loc}(\boldsymbol{\sigma}^{(i)}) \right)$ Evaluate via Monte Carlo average

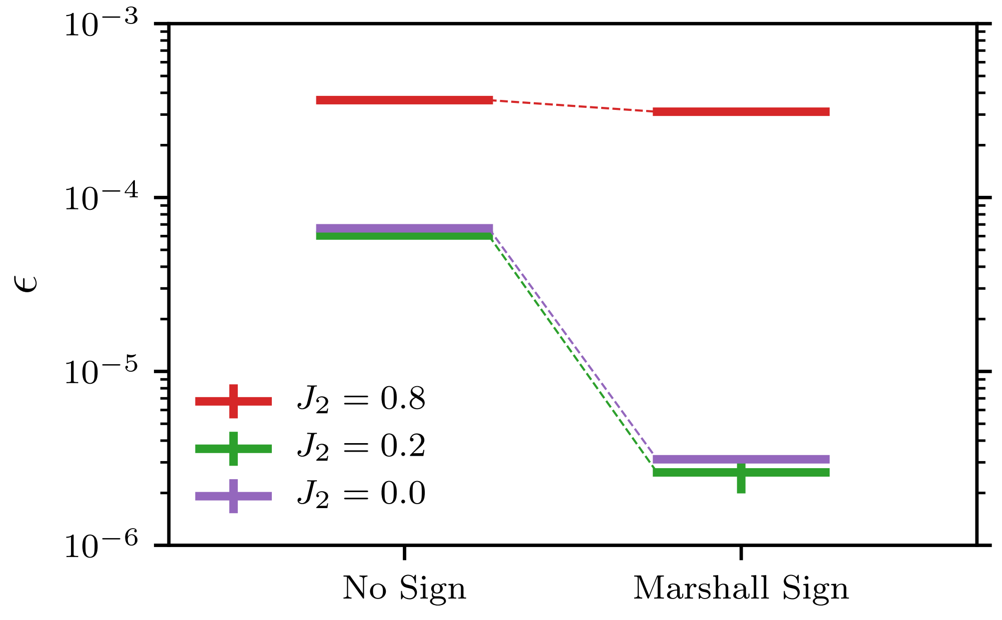
Recurrent neural network wavefunctions



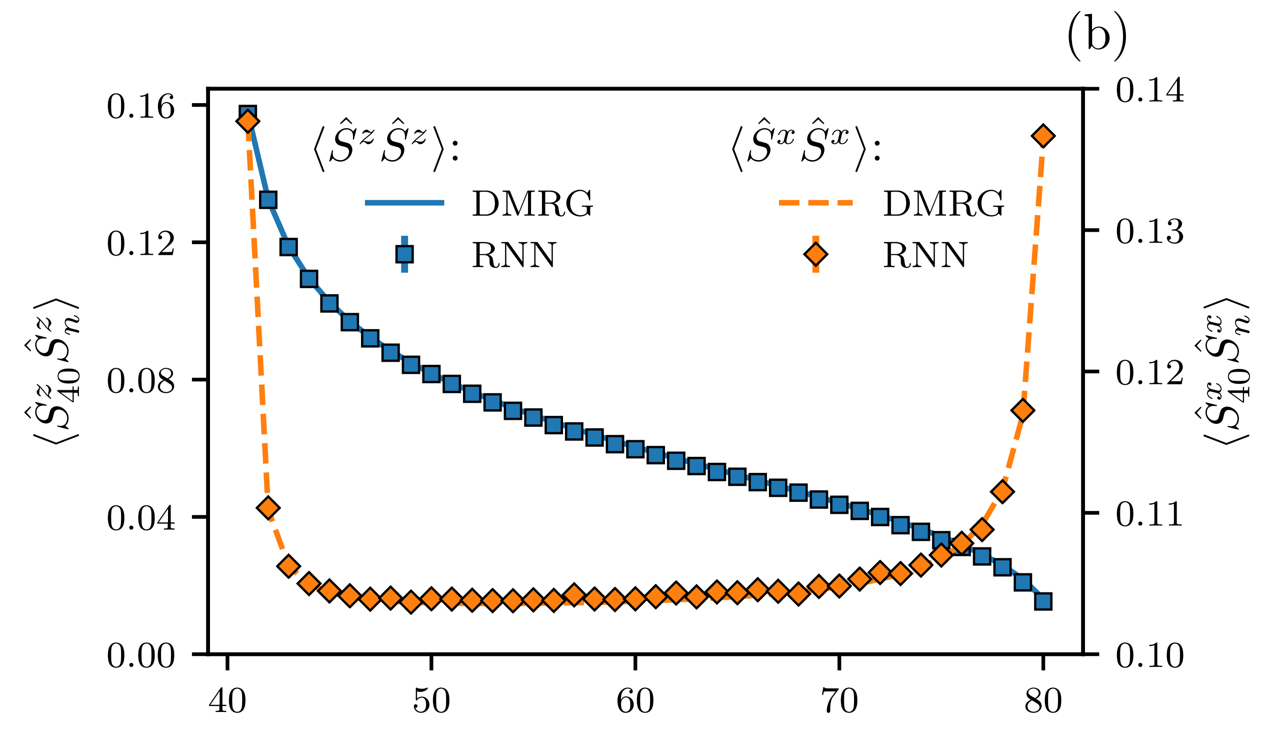
$$|\Psi\rangle = \sum_{\boldsymbol{\sigma}} \psi(\boldsymbol{\sigma})\,|\boldsymbol{\sigma}\rangle = \sum_{\boldsymbol{\sigma}} \sqrt{P(\boldsymbol{\sigma})}\,|\boldsymbol{\sigma}\rangle \qquad |\Psi\rangle = \sum_{\boldsymbol{\sigma}} \exp(\mathrm{i}\phi(\boldsymbol{\sigma}))\sqrt{P(\boldsymbol{\sigma})}\,|\boldsymbol{\sigma}\rangle$$



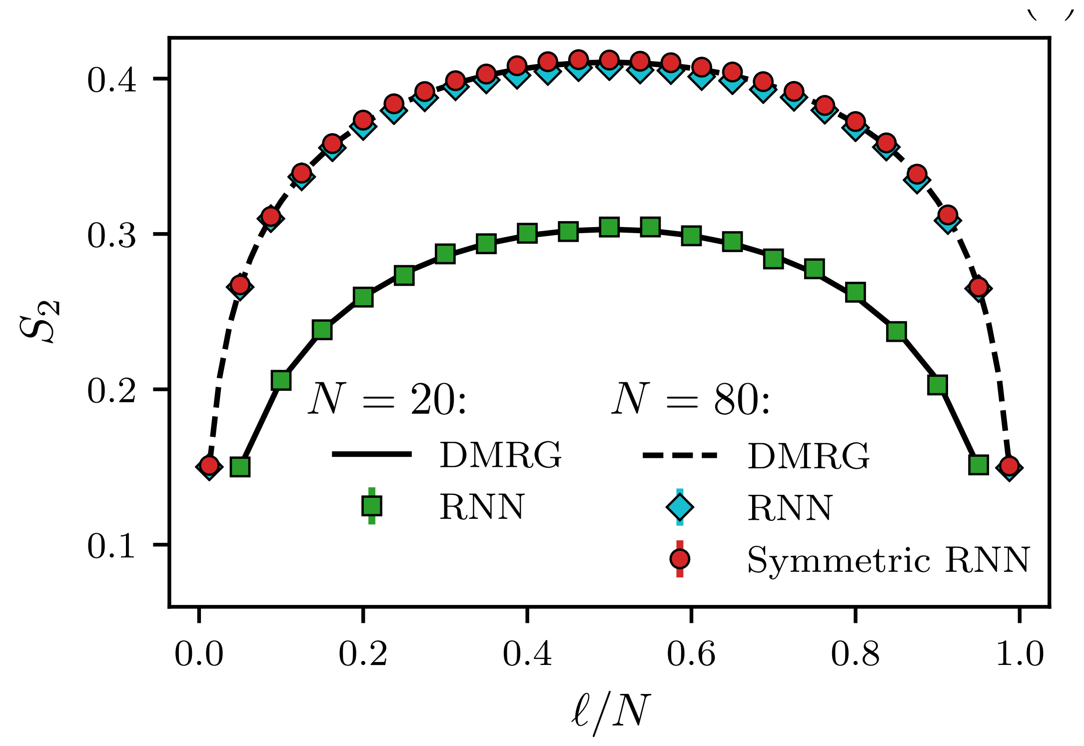
TFIM 1-D N=1000



J1-J2 model in 1-D



TFIM 1-D N=80



TFIM 1-D N=80

Symmetries: Spin inversion, mirror reflection, Sz. Sign: different Marshall signs for the J1-J2 model

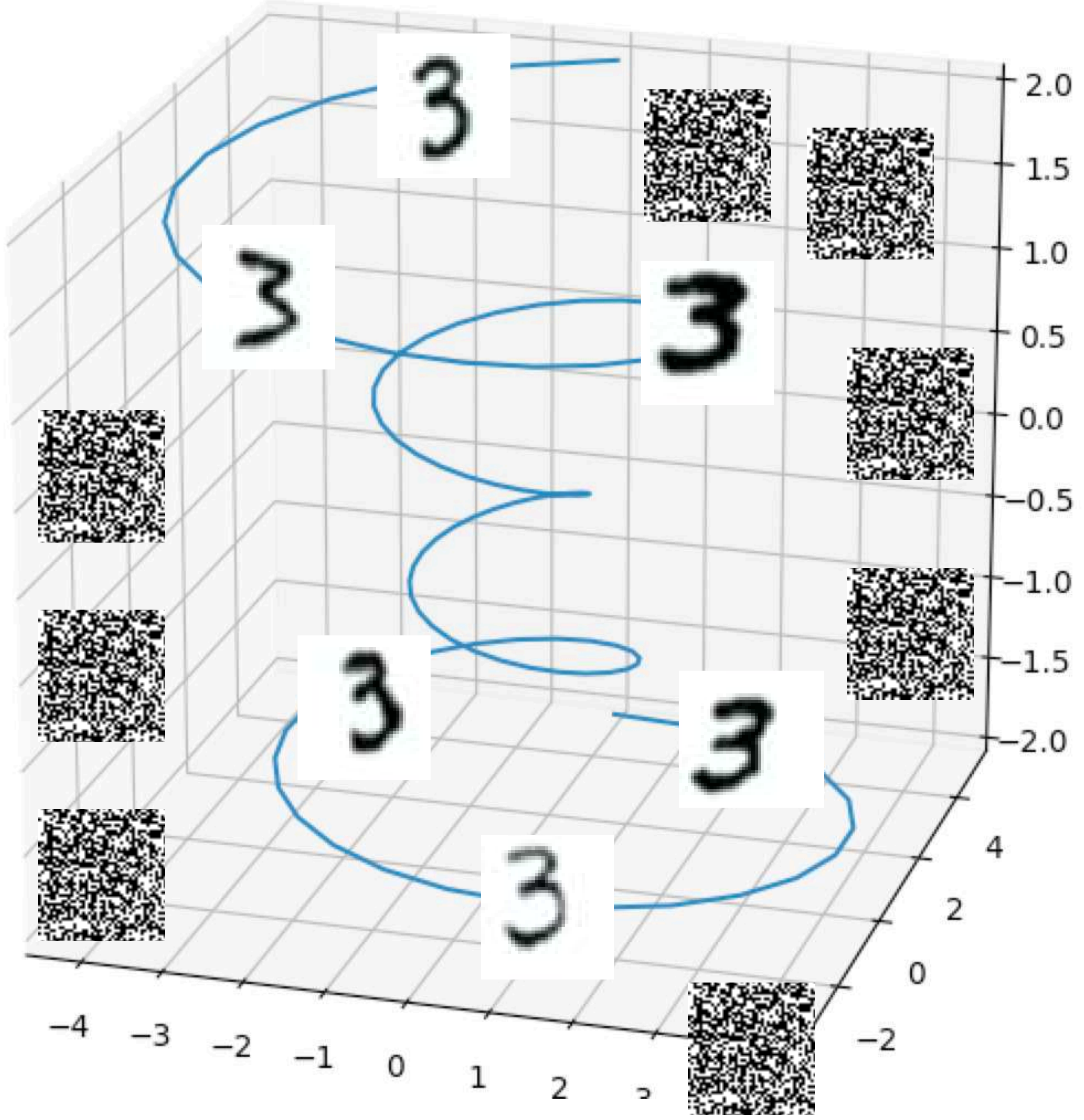
Mohamed Hibat-Allah, Martin Ganahl, Lauren E. Hayward, Roger G. Melko, and Juan Carrasquilla Phys. Rev. Research **2**, 023358 (2020)

Classical topological order

- The peculiar structure of phase space divided into sectors that are connected exclusively by extensive rearrangements of the microscopic degrees of freedom.

Take 28 x 28 binary images

- Size of state space: $2^{28\times 28} = 1.017458 \times 10^{236}$
- Bigger than the number of atoms in the known un noise —> Probability distributions over the image e in low-dimensional subspace of these big spaces.



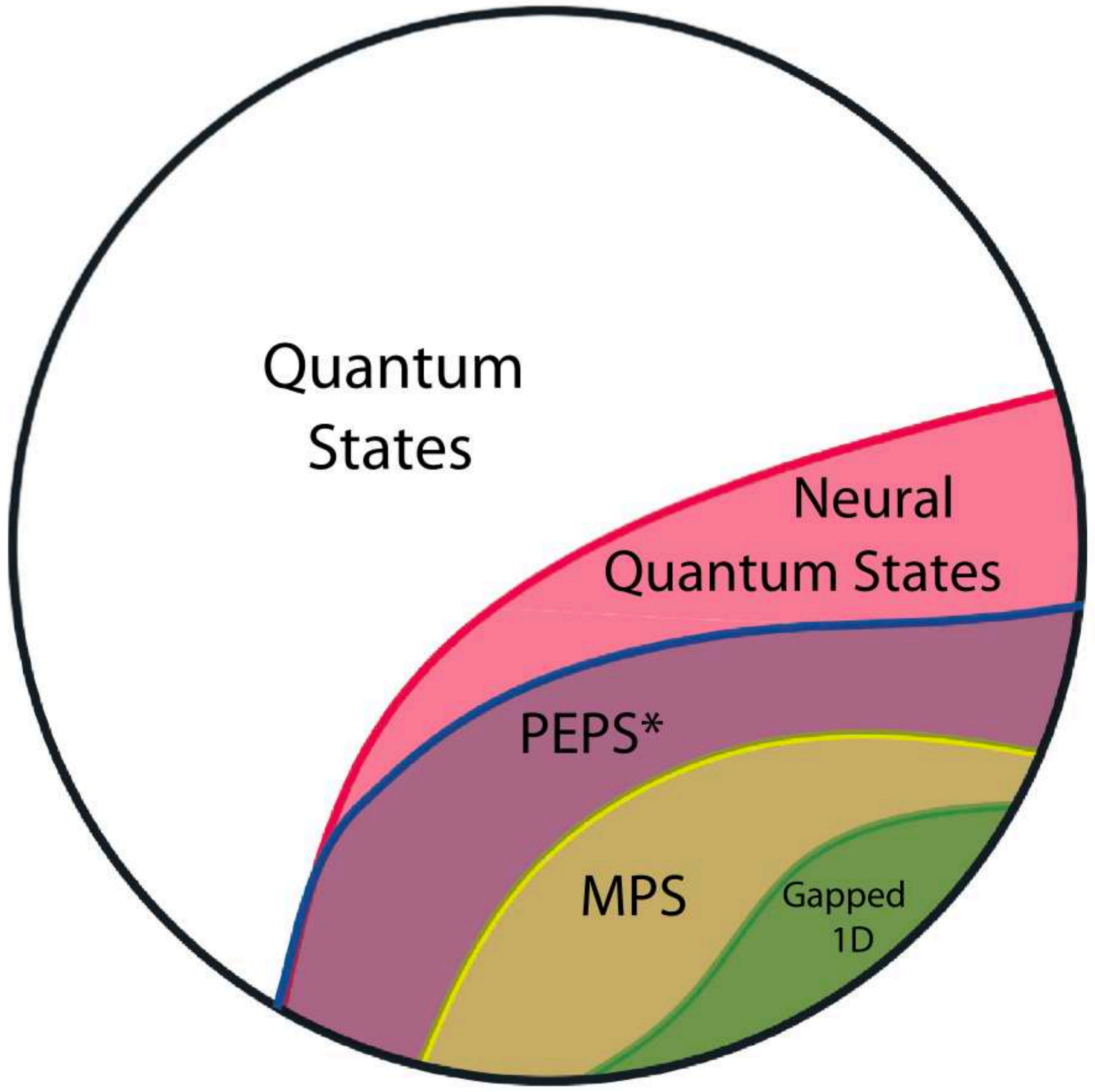


FIG. 3. Expressive power of classically tractable variational quantum states. Different classes of quantum states describing a qudit system with N degrees of freedom and comprising poly(N) variational parameters are compared. MPS can efficiently represent gapped ground states of one-dimensional systems. PEPS* denotes projected entangled pair states of bond dimension χ that are exactly or approximately contracted in poly(N, χ) time on a classical computer. NQS comprise all polynomially tractable TN, thus include MPS, and PEPS∗, while also representing additional states with volume-law entanglement that are not efficiently described by such planar TN.

PHYSICAL REVIEW B 106, 205136 (2022)

ML broad categories

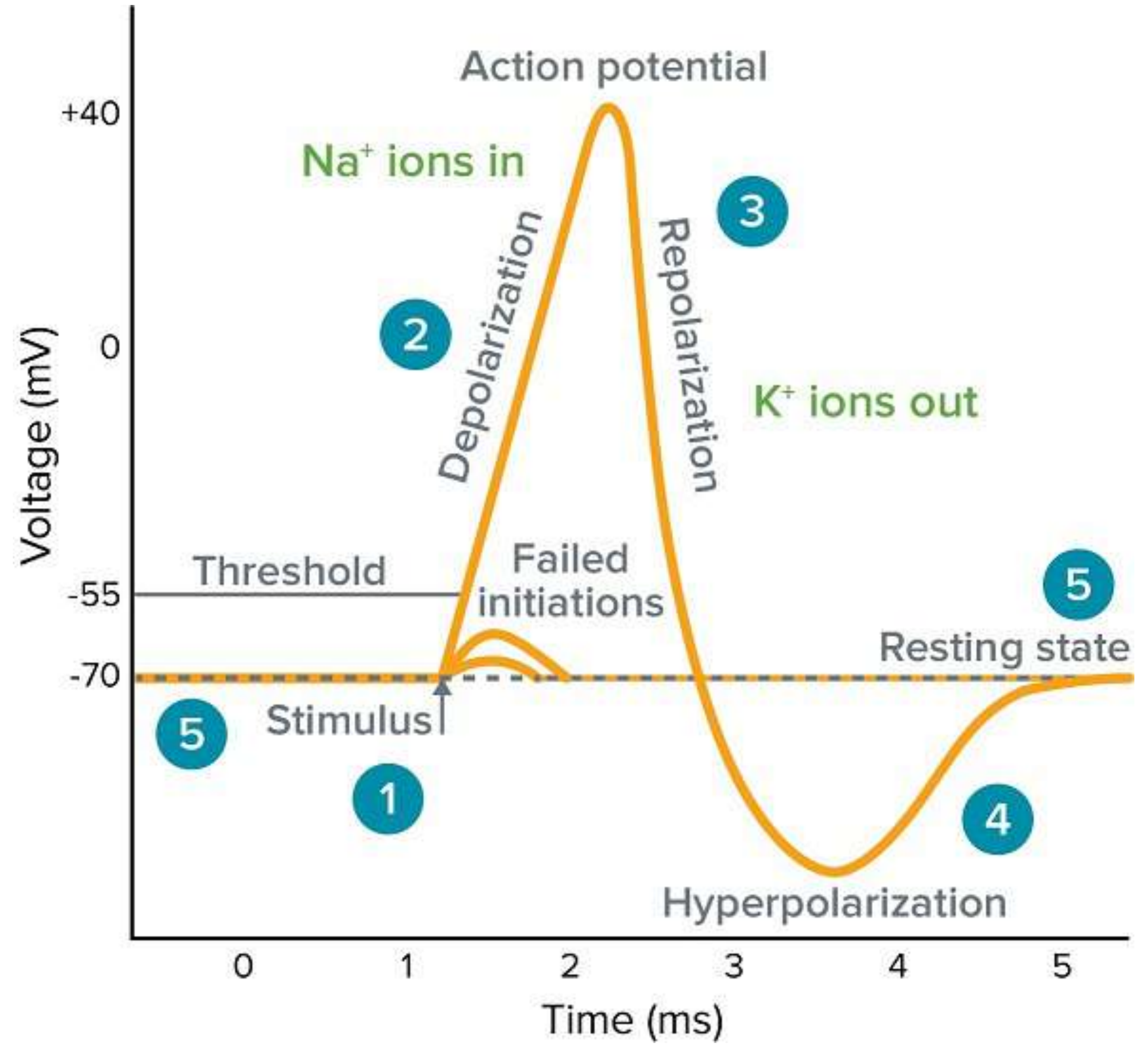
Equation/physical law learning

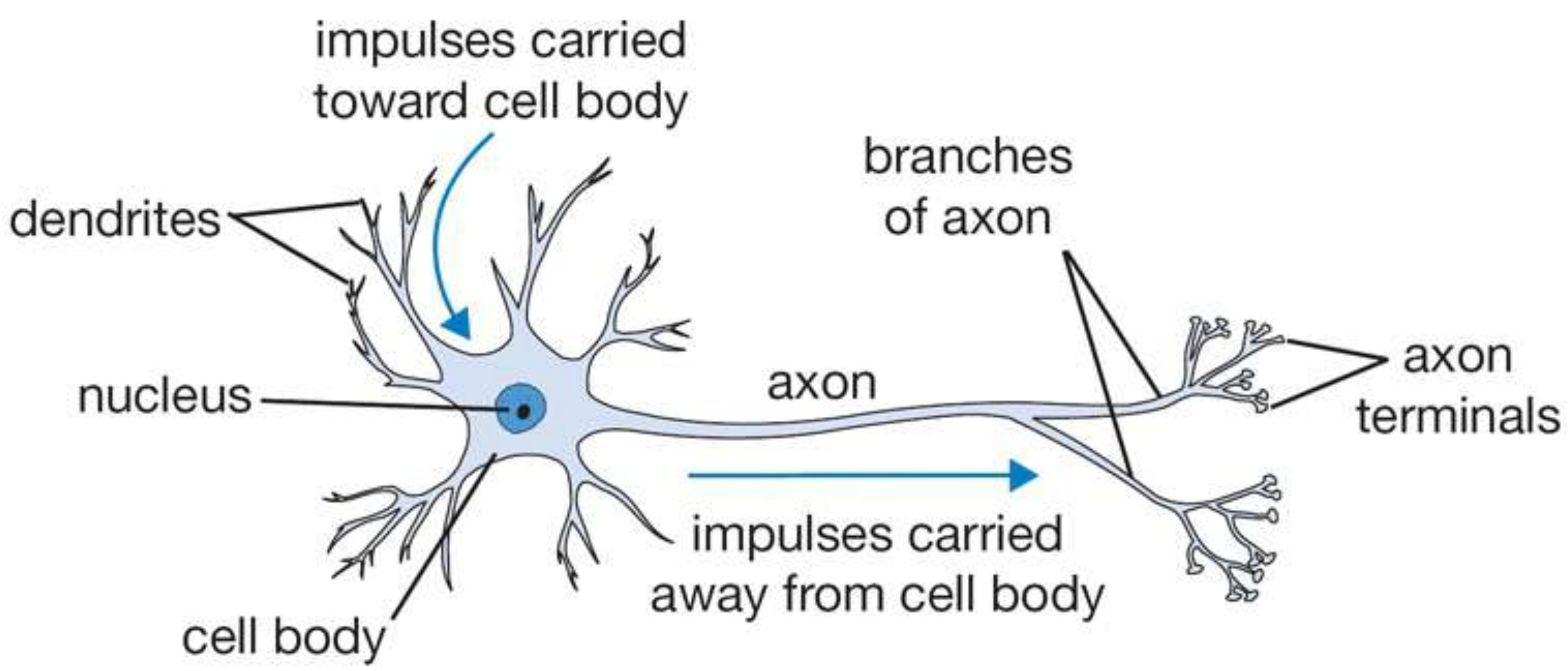
Closed quantum systems	
Time-dependent Schrödinger equation	
Exact	$\mathscr{L}_{\dot{\theta}}$
$i\frac{d}{dt}\|\Psi(t)\rangle = H\|\Psi(t)\rangle$	$\mathscr{D}\left(\|\Psi_{\theta(t)+\delta t\dot{\theta}(t)}\rangle, e^{-iH\delta t}\|\Psi_{\theta(t)}\rangle\right)$
Time-Independent Schrödinger equation	
Exact	$\mathscr{L}_{\theta}$
$H\|\Psi\rangle = E\|\Psi\rangle$	$\langle\Psi_\theta\|H\|\Psi_\theta\rangle$

Open quantum systems	
Master equation	
Exact	$\mathscr{L}_{\dot{\theta}}$
$\dot{\rho} = L\rho$	$\mathscr{D}\left(\rho_{\theta(t)+\delta t\dot{\theta}(t)}, e^{L\delta t}\rho_{\theta(t)}\right)$
Steady state	
Exact	$\mathscr{L}_{\theta}$
$\dot{\rho} = L\rho = 0$	$\|\|\dot{\rho}_\theta\|\|$
Gibbs state	
Exact	$\mathscr{L}_{\theta}$
$\rho(T) = \frac{e^{-H/T}}{Z(T)}$	$\mathrm{Tr}\left[\rho_\theta H\right] - TS\left(\rho_\theta\right)$

Inspiration: The brain

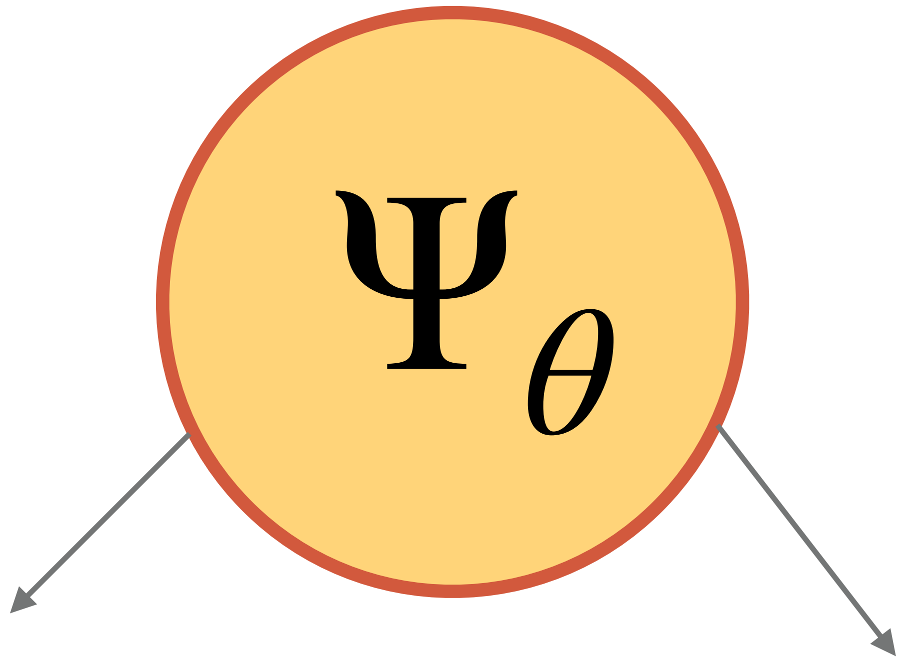
- Our brain has $\sim 10^{11}$ neurons, each of which communicates to other $\sim 10^{4}$ neurons





- Neurons receive input signals and accumulate voltage. After some threshold they will fire spiking responses.
- Pic credit: www.moleculardevices.com, http://cs231n.github.io/neural-networks-1/

Neural networks as quantum states



Hamiltonian driven learning —

Finding ground states. This is just one example, but there are many more tasks that are driven by a "physical" principle.

Data driven learning —

quantum state tomography, approximate reconstruction of quantum devices, quantum simulations and Quantum channels

Closest in spirit to ML tasks

Neural networks as quantum states

$$\Psi_\theta$$

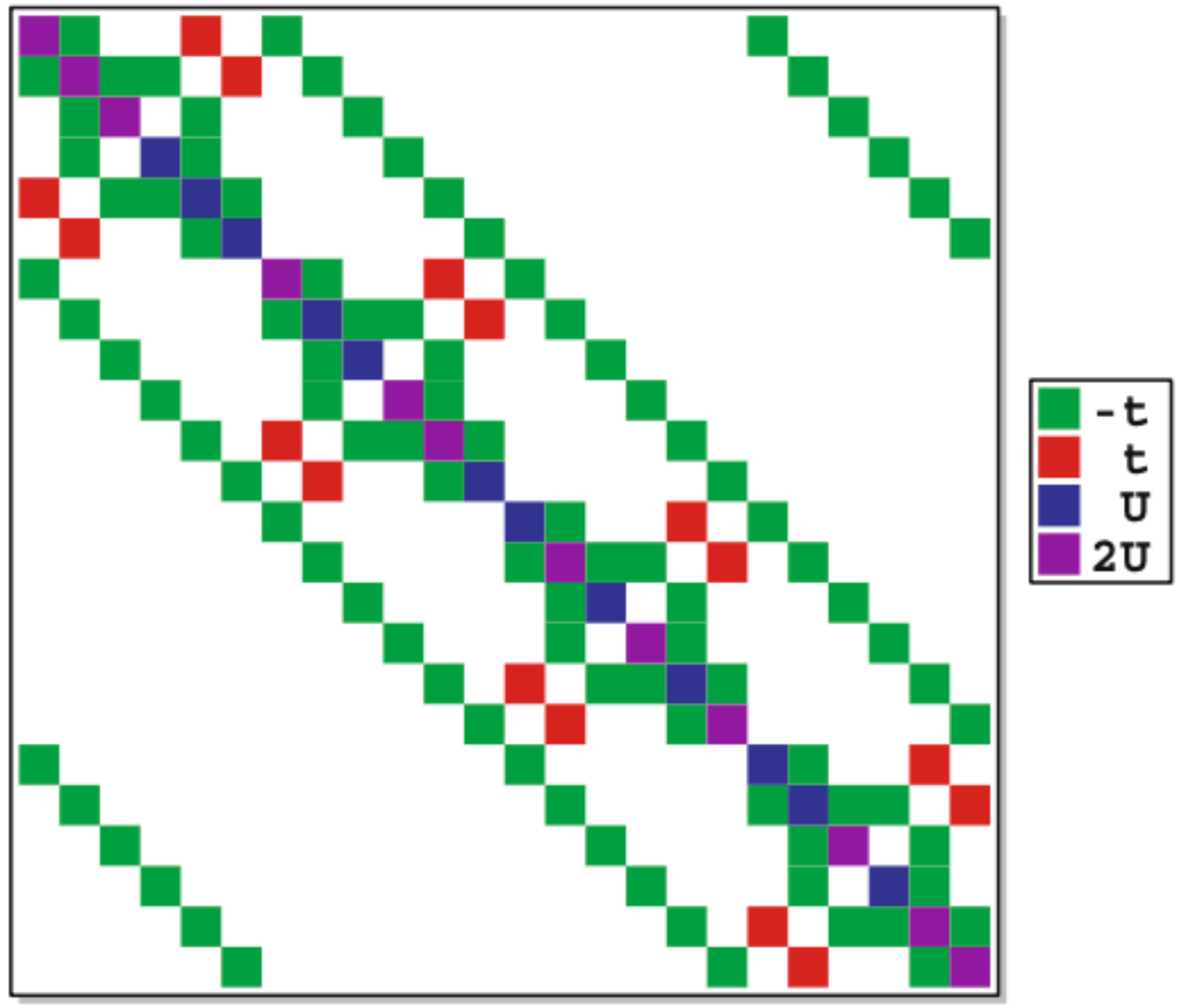
Hamiltonian driven learning —

Finding ground states. This is just one example, but there are many more tasks that are driven by a "physical" principle.

Solving for the low-energy, time-independent solutions of the Schrodinger equation.

Ground states

- Given a problem Hamiltonian H, approximate its ground state.
- H is a very large Hermitian matrix that describes the behaviour of a physical system at the microscopic scale.
- Finding the ground state — the Hamiltonian's lowest energy eigenvector and eigenvalues.
- Typically hard because of exponential complexity.
- Solve the problem **analytically** or propose an Ansatz inspired by machine learning techniques (**neural network**)

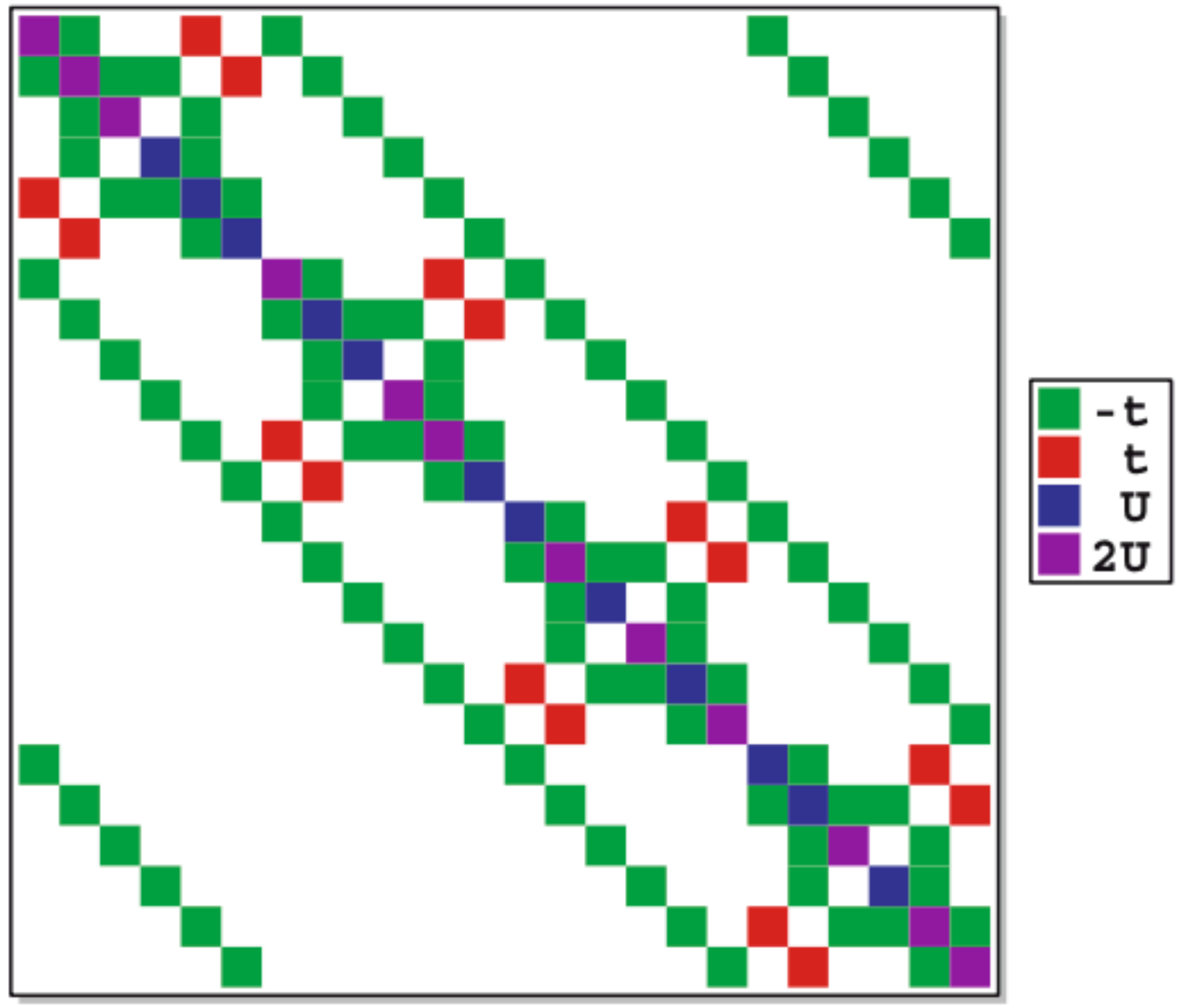


Schematic representation of the Hamiltonian matrix of the Hubbard model with L = 4, $N\uparrow = 3$, $N\downarrow = 2$ (5 particles)

Exact Diagonalization Techniques.
Alexander WeißeHolger Fehske.
Part of the Lecture Notes in Physics book series (LNP, volume 739)

Ground states

- Hamiltonians are very large— problem is computationally difficult.
- Structured and sparse.
- Symmetries (some of which are common to important symmetries and inductive biases in ML).
- The rows and columns indices are usually related to real space configurations of the particles in the system and are usually ordered using bit-strings representations (001010110).



Schematic representation of the Hamiltonian matrix of the Hubbard model with L = 4, N↑ = 3, N↓ = 2 (5 particles)

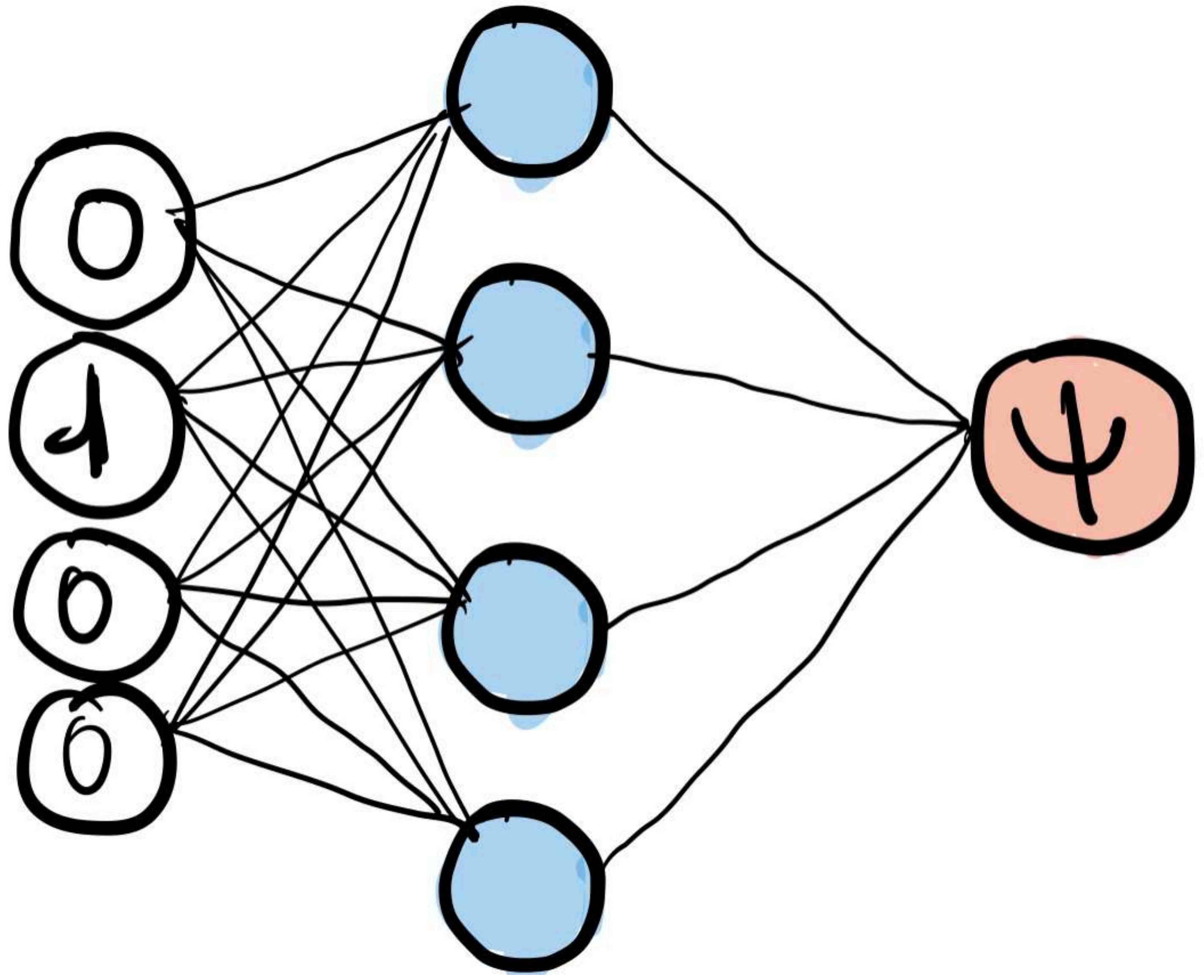
Approximating ground states with neural networks

- Recall that we represent a quantum state as a 2^N-dimensional vector of complex entries
- $$|\psi\rangle = \begin{bmatrix} \psi_{0,0,0,\ldots 0,0} \\ \psi_{1,0,0,\ldots 0,0} \\ \psi_{1,1,0,\ldots 0,0} \\ \vdots \\ \psi_{1,1,1,\ldots 1,1} \end{bmatrix}$$ What does it mean that we represent a quantum state as a neural network?

$$|\psi_\theta\rangle = \begin{bmatrix} \psi_\theta(0,0,0,\ldots 0,0) \\ \psi_\theta(1,0,0,\ldots 0,0) \\ \psi_\theta(1,1,0,\ldots 0,0) \\ \vdots \\ \psi_\theta(1,1,1,\ldots 1,1) \end{bmatrix}$$

Where the complex-valued boolean function

$\psi_\theta(x_1, x_2, \ldots, x_N) =$ **Neural network**$(x_1, x_2, \ldots, x_N)$



As a consequence, we go from an exponential amount of parameters to a neural network with a few parameters at the cost of constraining the type of functions we can represent.