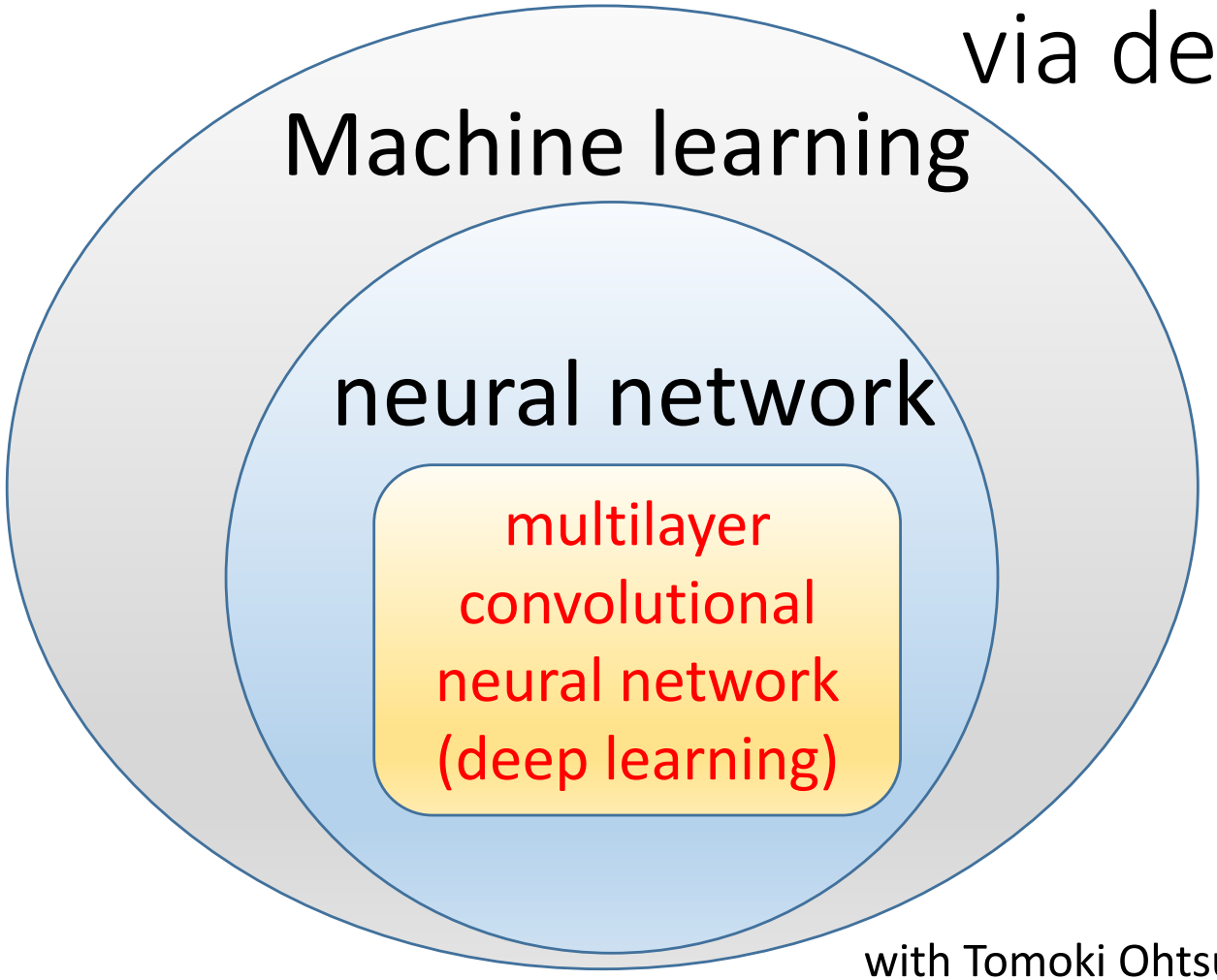
Anderson localization study via deep learning



Physics division
Sophia University
Tomi Ohtsuki

with Tomoki Ohtsuki, NTT data mathematical systems and T. Mano, Sophia univ.

J. Phys. Soc. Jpn. **85**, 123706 (2016)→2D, open access
J. Phys. Soc. Jpn. **86**, 044708 (2017)→3D, open access
T. Mano and T. Ohtsuki: in preparation

Machine learning method in condensed matter physics

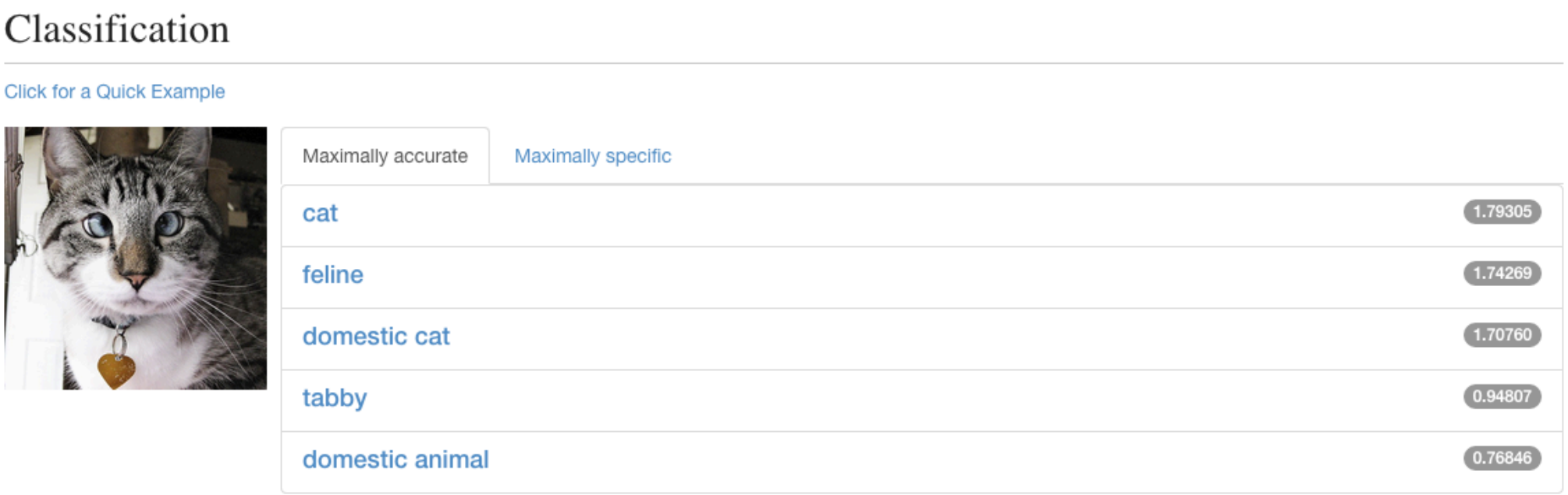
- Though still restricted to standard simplified models, machine learning methods seem to be gradually accepted among solid state physics community from 2016.
  - arXiv1605.01735 (Nat. Phys. '17) : 2D Ising model
  - arXiv1606.02318 (Science '17): Heisenberg model
  - arXiv:1610.02048 (Nat. Phys. '17): Kitaev chain, Ising, disordered quantum spin chain
  - arxiv1609.09087 (JPSJ '17): 2D Ising model, the most beautiful
  - arXiv1608.07848: 2D Hubbard model, $G(E, x, y)$
  - arXiv1609.02552 : 3D Hubbard model, temperature transition
  - arXiv1611.01518: Chern insulator, bulk (PRL '17)
  - arXiv1609.03705 (PRB '17): DFT with machine learning
  - JPSJ 86, 093001 (2017): Bose-Hubbard model

Classification of wave functions in random systems

- Random electron systems show various phases with specific wave functions->Regard $|\psi(\boldsymbol{x})|^2$ as image
- Use multilayer (four-weight layer) convolutional neural network (deep learning) to classify the pattern of $|\psi(\boldsymbol{x})|^2$ and draw the phase diagram of quantum phase transition
  - 2D metal (delocalization)-insulator (localization) transition (Anderson transition)
  - 2D topological insulator-Anderson insulator transition
  - 3D Anderson transition
  - 3D strong/weak topological insulator-ordinary insulator transition
  - Weyl semimetal
  - magnon Hall effect
  - Quantum percolation

Image recognition by deep learning

A Practical Introduction to Deep Learning with Caffe and Python
http://adilmoujahid.com/posts/2016/06/introduction-deep-learning-python-caffe/



In short, we don't teach machine the features of cat. The machine captures the features by supervised learning.

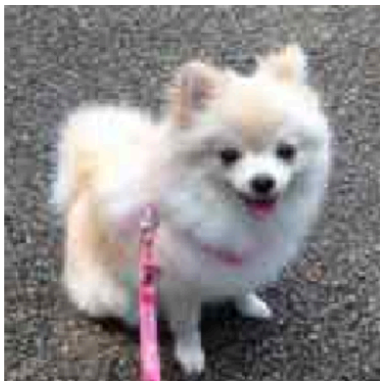
http://demo.caffe.berkeleyvision.org/classify_upload

Caffe Demos

The Caffe neural network library makes implementing state-of-the-art computer vision systems easy.

Classification

Click for a Quick Example

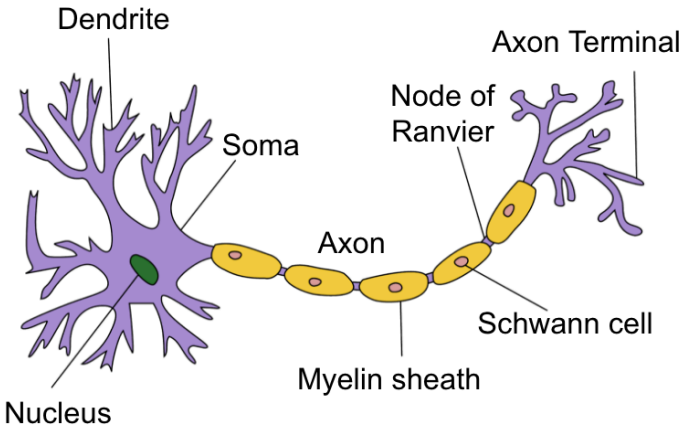


Maximally accurate | Maximally specific

spitz	2.07032
Pomeranian	2.03635
dog	1.17822
domestic animal	1.12034
canine	1.05956

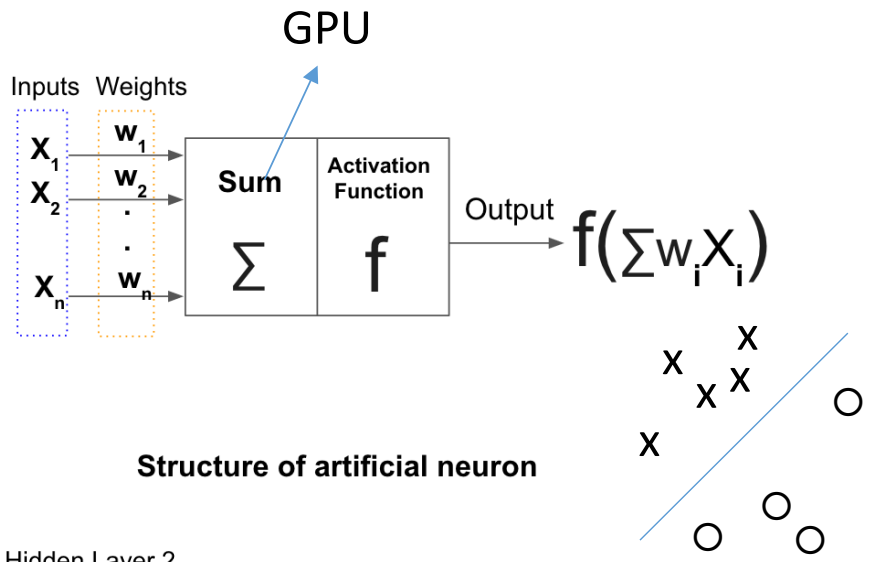
CNN took 0.062 seconds

Let's use image recognition for physics.



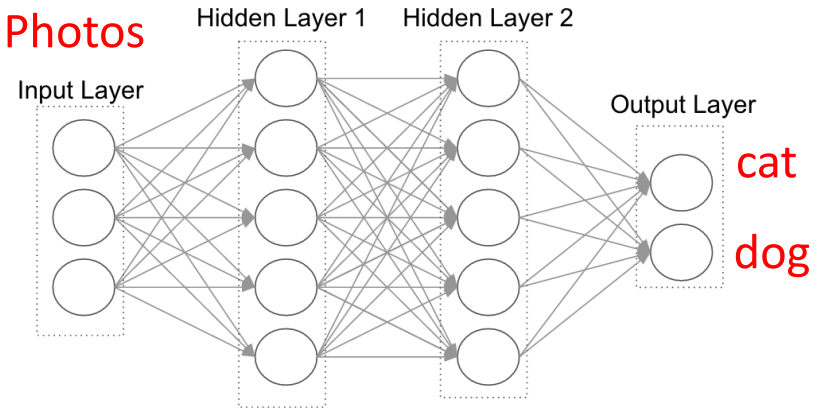
Structure of a typical neuron

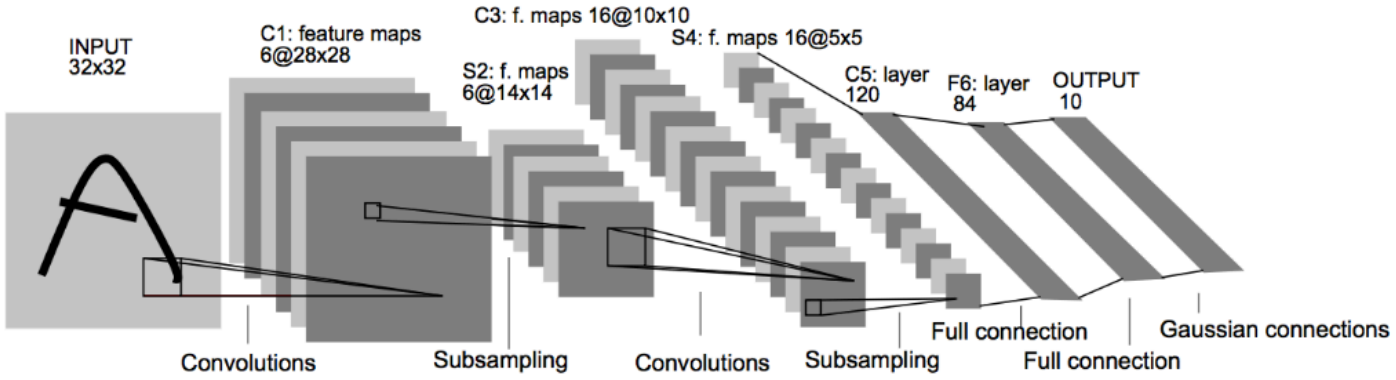
(source: Wikipedia)



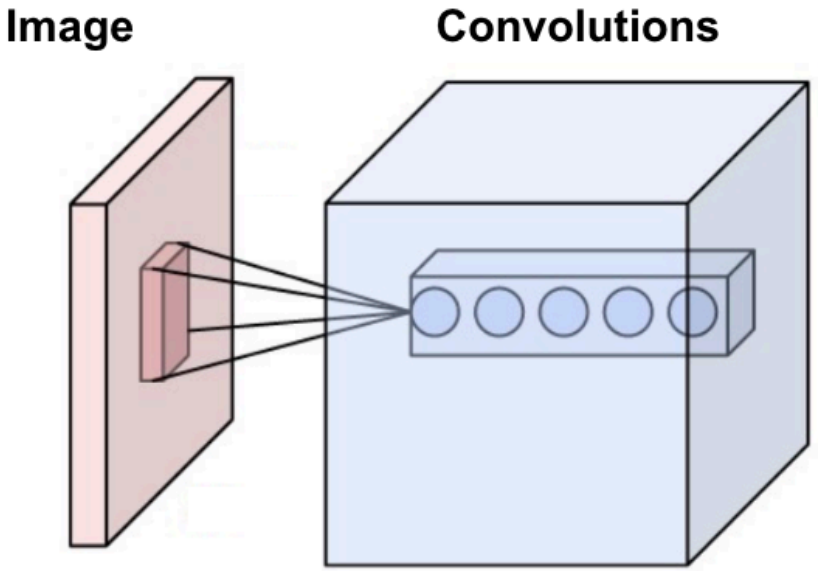
Structure of artificial neuron

works fine only when the classification is simple

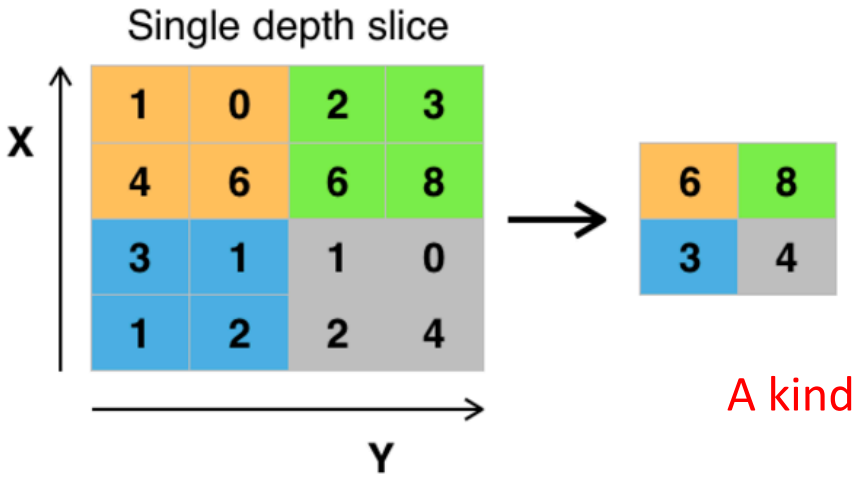




CNN called LeNet by Yann LeCun (1998)



Neurons of a convolutional layer, connected to their receptive field
(source: Wikipedia)



A kind of renormalization group

Max pooling with a 2x2 filter and stride = 2
(source: Wikipedia)

Each phase of random electron systems exhibits specific wave functions

- non-interacting random systems
  - nontrivial phases
  - large scale samples by changing the seed of random number
- Extended
- Localized (Anderson localization)
- Edge states (2d topological insulator)
- Dirac Surface states (3d topological insulator), STI, WTI
- Fermi arc (3d Weyl semimetal)

2D Localization-Delocalization transition

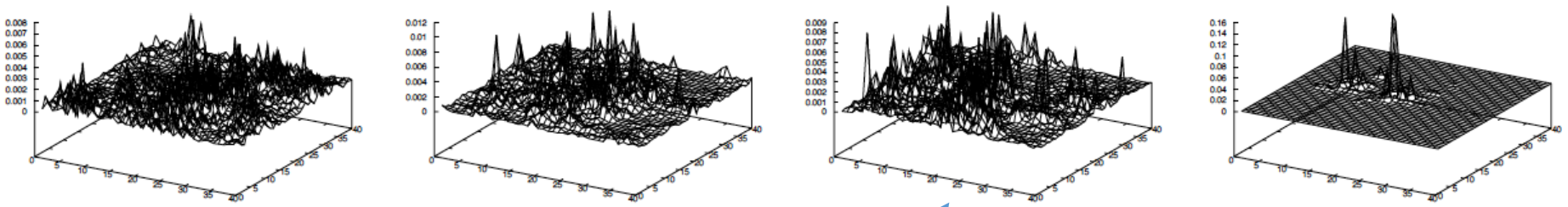
- SU(2) model

Asada, Slevin, TO, PRL '02

$$H = \sum_{i,\sigma} \epsilon_i c^{\dagger}_{i,\sigma} c_{i,\sigma} - \sum_{\langle i,j\rangle,\sigma,\sigma'} R(i,j)_{\sigma,\sigma'} c^{\dagger}_{i,\sigma} c_{j,\sigma'} ,$$

ε_i=[-*W*/2,*W*/2]

$$R(i,j) = \begin{pmatrix} e^{i\alpha_{i,j}} \cos\beta_{i,j} & e^{i\gamma_{i,j}} \sin\beta_{i,j} \\ -e^{-i\gamma_{i,j}} \sin\beta_{i,j} & e^{-i\alpha_{i,j}} \cos\beta_{i,j} \end{pmatrix},$$



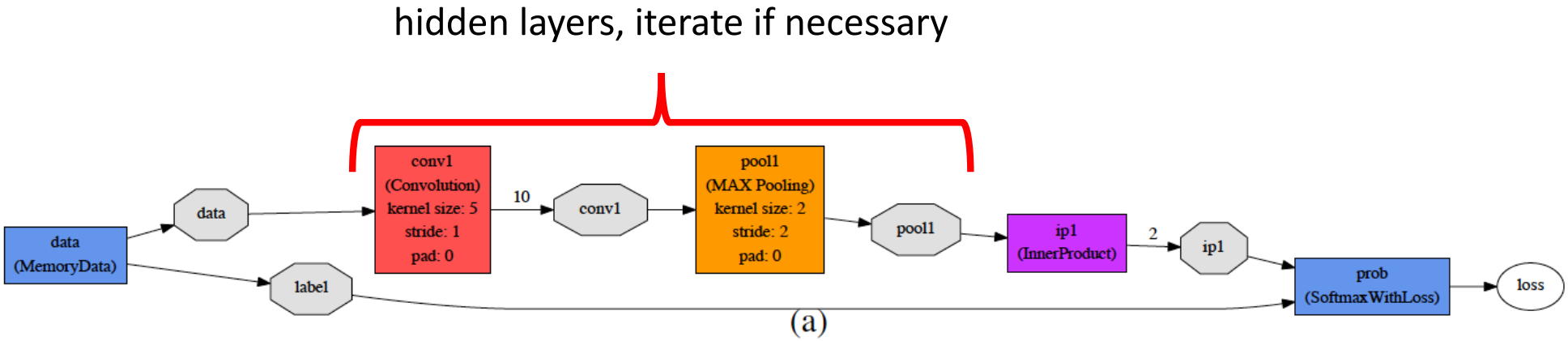
(a) (b) (c) (d)

metal metal insulator insulator

Indistinguishable at first sight

Deep learning

- Training phase
  - Prepare 1000 eigenstates for $W<W_c$, teach the machine that the eigenfunction is “delocalized (metal phase)”
  - Prepare 1000 eigenstates for $W>W_c$, teach the machine that the eigenfunction is “localized (insulator phase)”
  - minimize $-\Sigma_i p_i' \log p_i = -\Sigma' \log p_i$
  - 1800 for training, 200 for testing.
- Prediction phase
  - Diagonalize independent 100 samples and let the machine judge whether they are delocalized or localized with probability P_{deloc} and $P_{loc}=1-P_{deloc}$.

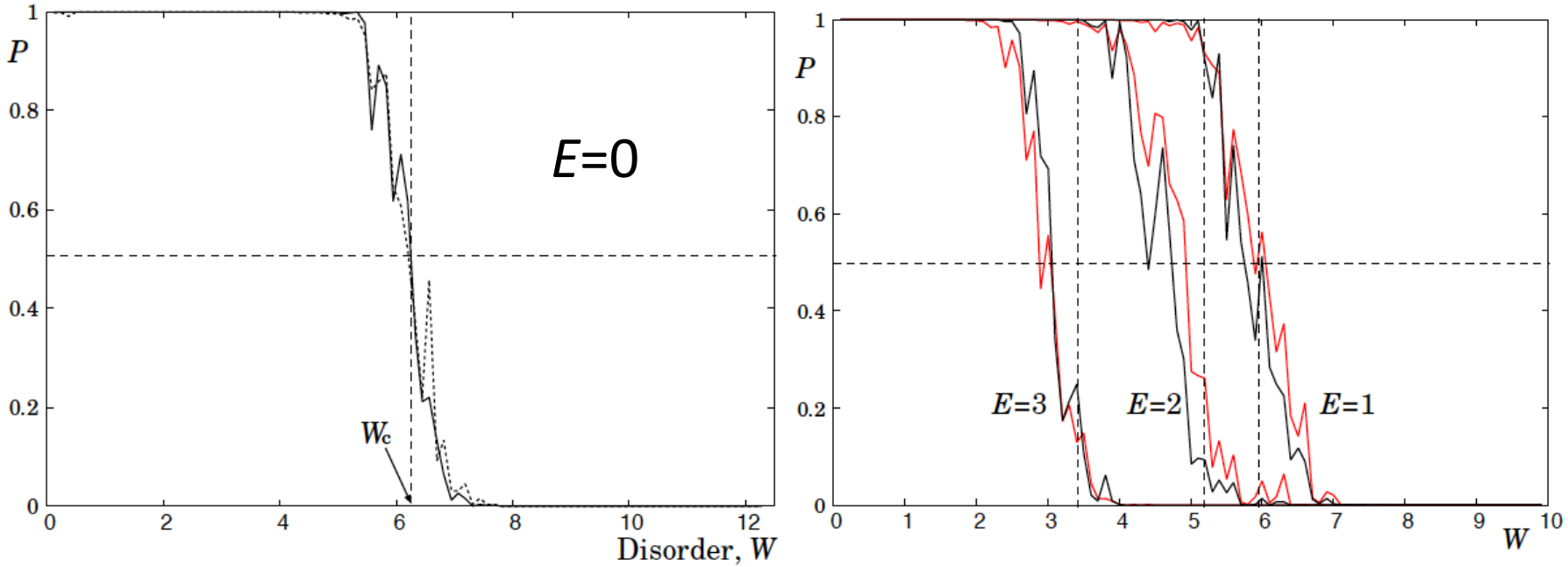


technical note

- no supercomputers
- needs a lot of eigenfunctions for training and testing→cluster workstations
  - sparse matrix diagonalization algorithm is sometimes better
  - typically 10-100 GB storage is required → binary format to read and write
- single workstation with GPU (Tesla K40 or Geforce 1080/1080 ti) is enough for training and testing. Training takes at most an hour with GPU, but a few days without GPU.
- deep learning tools:
  - Caffe
  - Keras (front end) + tensorflow(backend)

Results

- *P*: probability that the wave function is delocalized

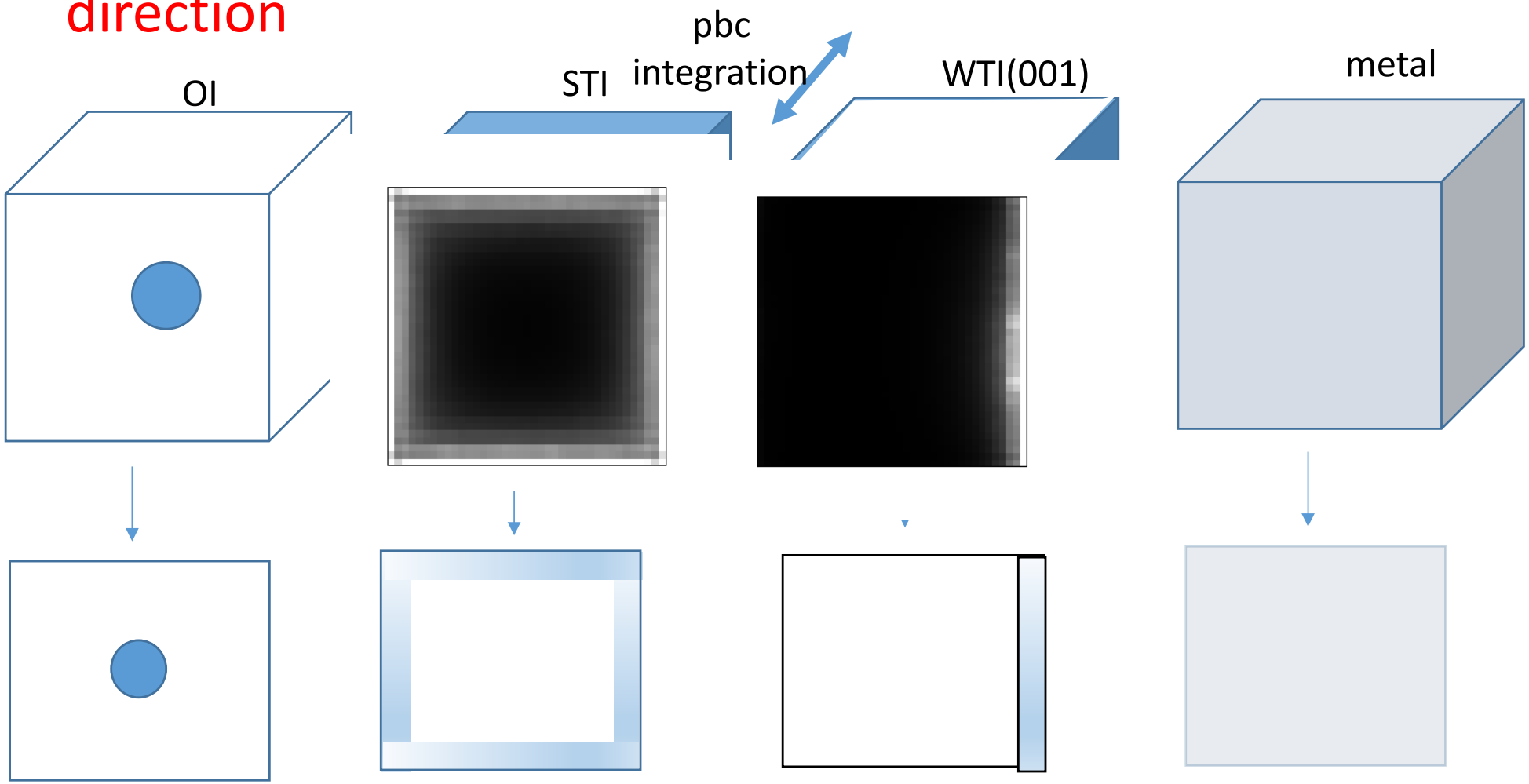


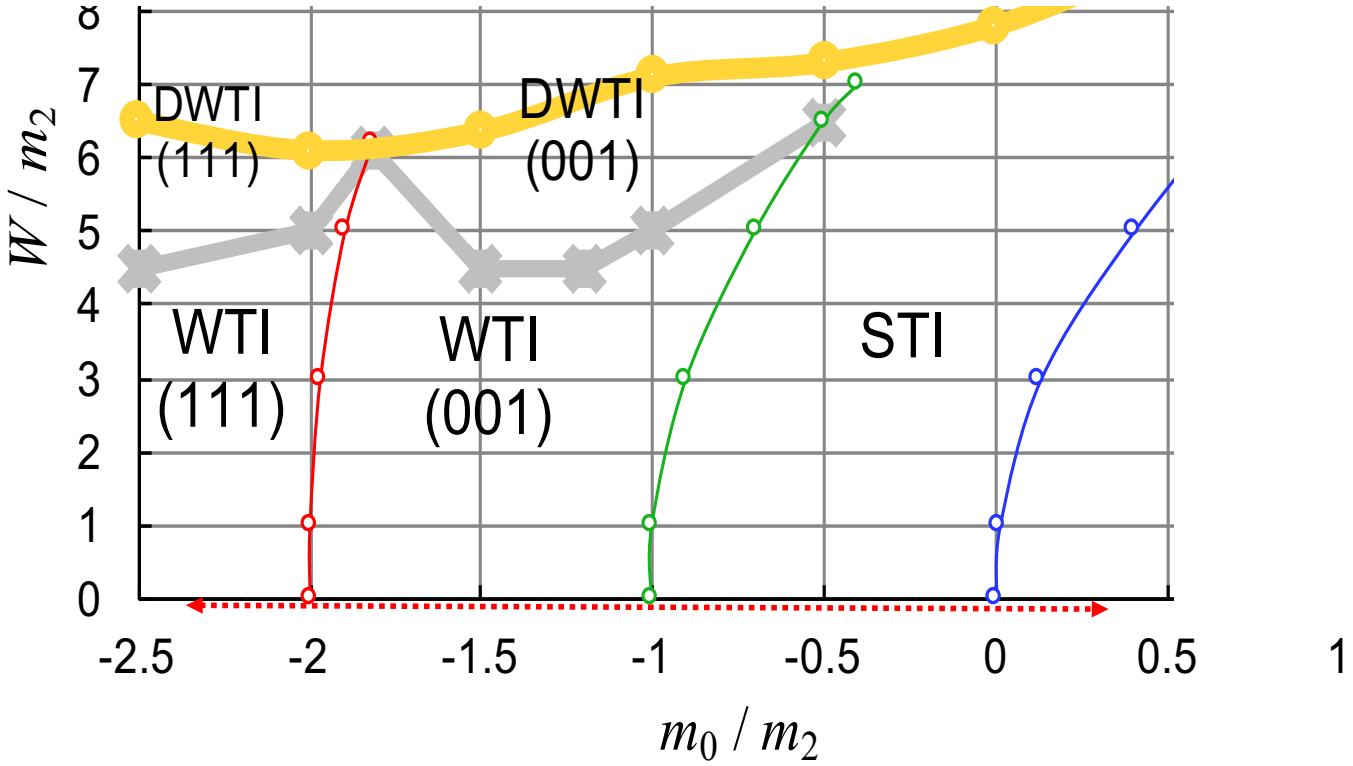
Applications to 3D topological system (3D TI and 3D WSM)

- focus on the existence of surface states
- choose proper boundary conditions
- integrate over one direction to have 2D image

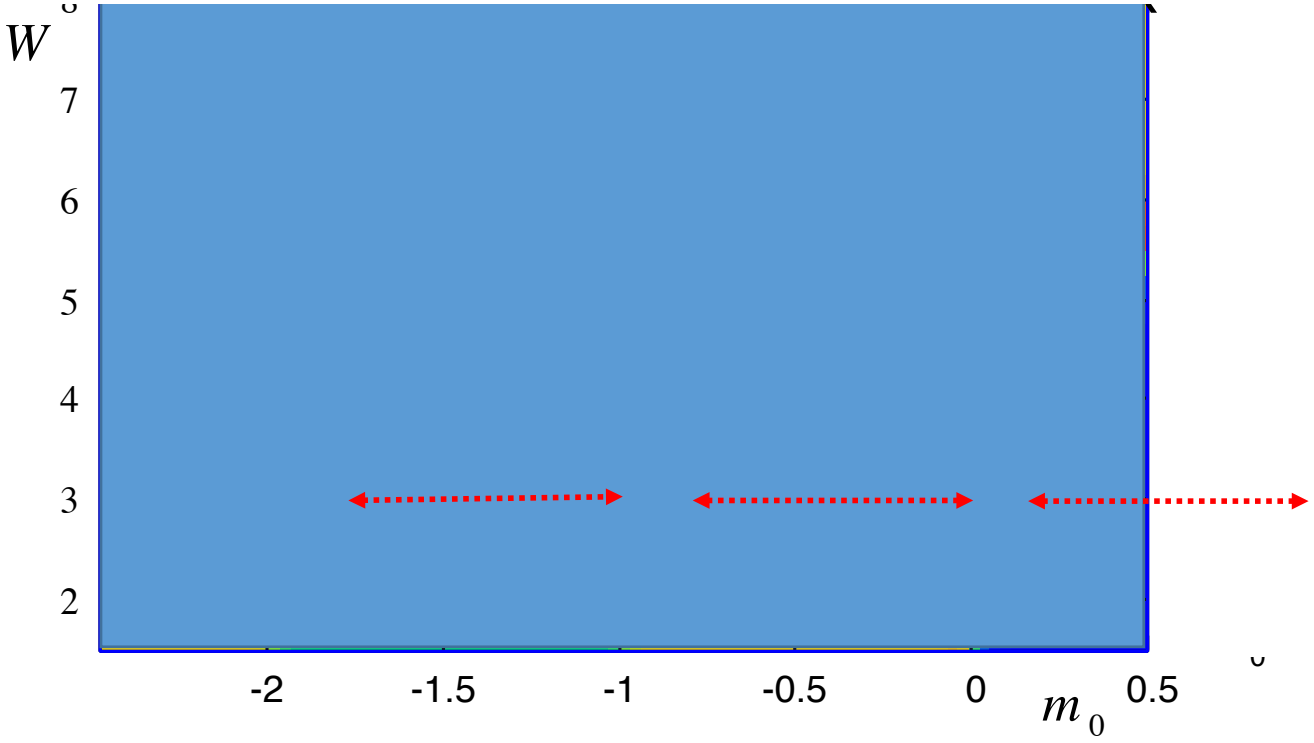
How we teach what TI's are.

- Topological insulator is characterized by surface states
- Choose proper boundary condition, integrate over one direction

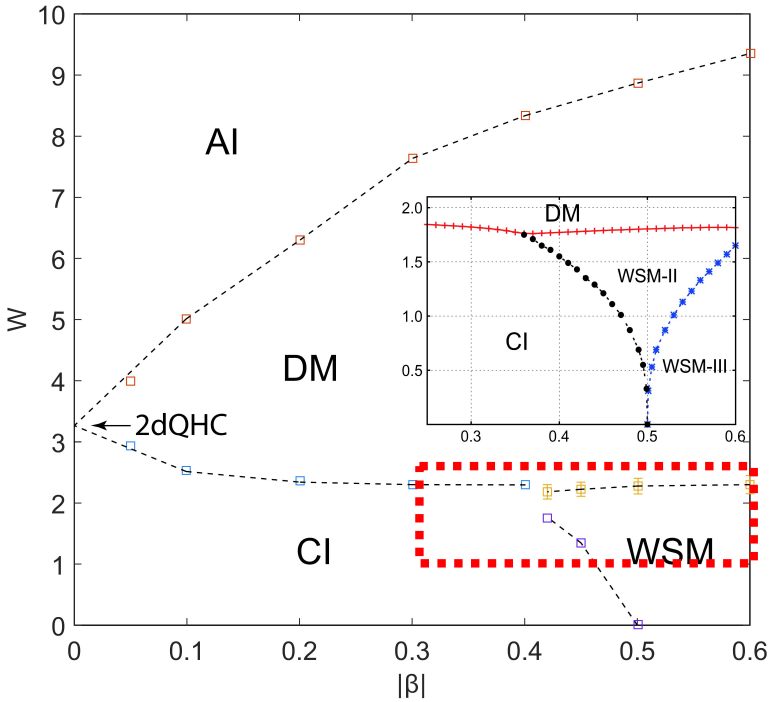




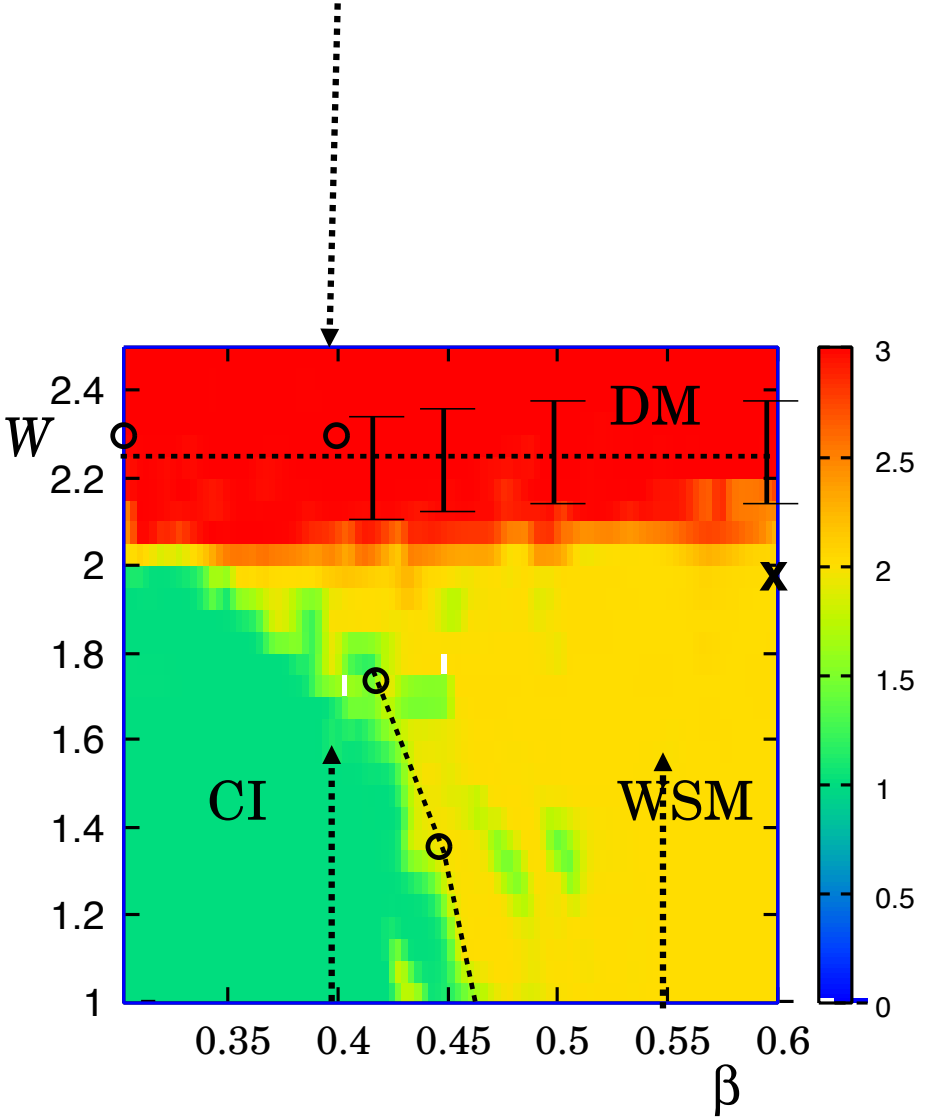
Koji Kobayashi,
unpublished



$0 \times P_{OI} + 1 \times P_{WTI} +$
$2 \times P_{STI} + 3 \times P_{Metal}$



Liu, Ohtsuki, Shindou, PRL16



$$1 \times P_{\mathrm{CI}} + 2 \times P_{\mathrm{WSM}} + 3 \times P_{\mathrm{DM}}$$

Ohtsuki, JPSJ 17

3D Anderson localization and 3D quantum percolation

- New challenge: 3D image recognition
- Method: 3D deep convolutional neural network (6 weight layer network)
- global phase diagram in *W*-*E* plane. (conventionally, we fix the energy *E* and vary *W*.)
- Byproduct: phase diagram for 3D quantum percolation (QP), i.e., geometrically random systems
  - transfer matrix method is not applicable for QP

model and method

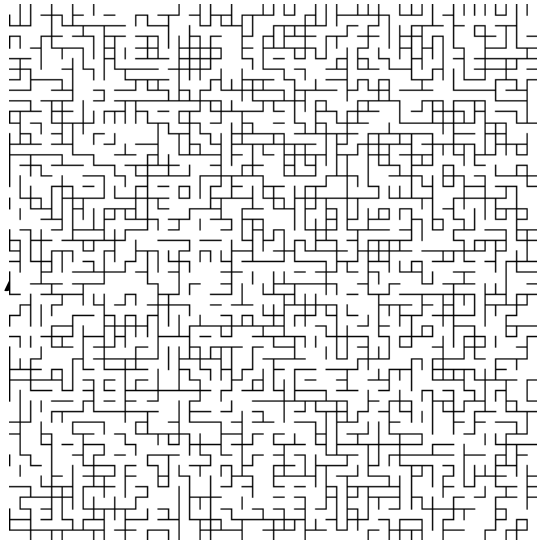
- Anderson model

$$H=\sum_{j}\varepsilon_{j}|j\rangle\langle j|+\sum_{j',j}V_{j',j}|j'\rangle\langle j| \quad , \quad -\frac{W}{2}<\varepsilon_{j}<\frac{W}{2}$$

- quantum percolation, bond percolation.
  - $H=\sum \; V_{j',j}|j'\rangle\langle j|$, $V_{j',j}$=0 or 1 with probability p.
- for site percolation, site is randomly deleted with probability p.
- We first train the neural network for Anderson model, E=0, then use the trained network to obtain the phase diagrams in W-E (Anderson model), p-E (quantum percolation) planes.
- deeper the network, the better the phase diagram.

Brief introduction to percolation

- Bonds or sites are randomly connected with probability p.
- For $d>1$, an infinite cluster appears $p>p_c$, with p_c the percolation threshold.
- If the particle is quantum, $p>p_q>p_c$
  - p_q is the quantum percolation threshold.
  - Due to Anderson localization, $p_q>p_c$
- Due to the random structure of lattices,
  - transfer matrix is not applicable
  - spiky density of states

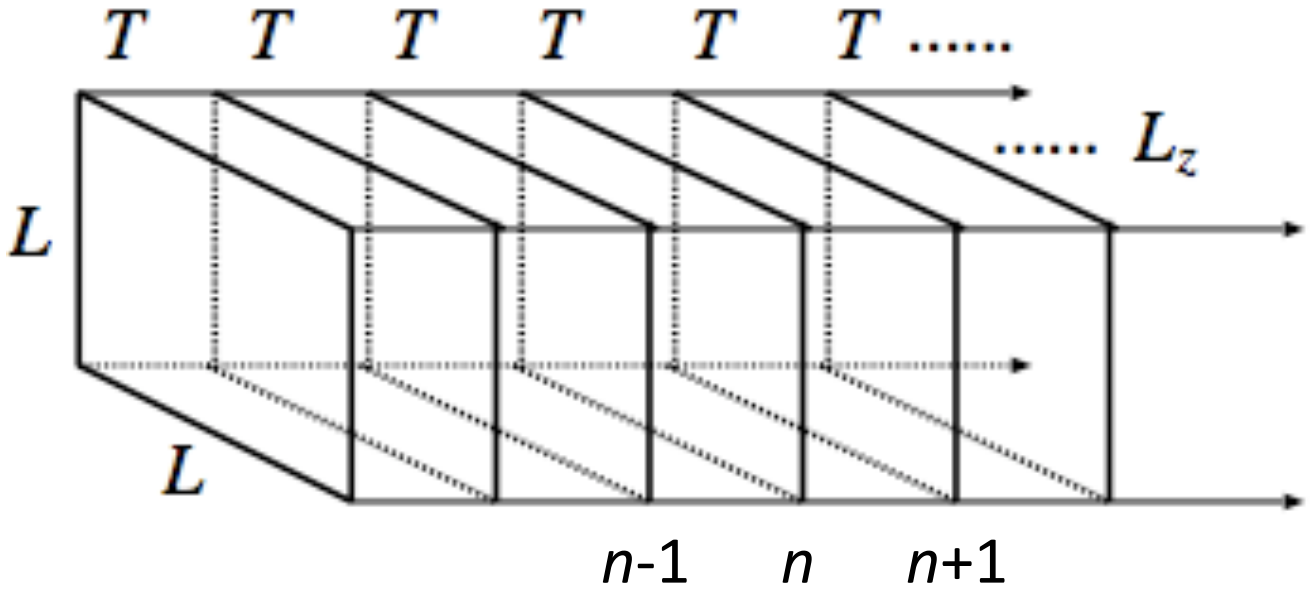


https://en.wikipedia.org/wiki/Percolation_theory

transfer matrix

$$E\psi_n = H_n\psi_n + V_{n,n+1}\psi_{n+1} + V_{n,n-1}\psi_{n-1}$$

$$\begin{pmatrix} \psi_{n+1} \\ V_{n+1,n}\psi_n \end{pmatrix} = T_n \begin{pmatrix} \psi_n \\ V_{n,n-1}\psi_{n-1} \end{pmatrix} \qquad T_n = \begin{pmatrix} V_{n,n+1}^{-1} & 0 \\ 0 & V_{n,n+1}^{\dagger} \end{pmatrix} \begin{pmatrix} EI - H_n & -I \\ I & 0 \end{pmatrix}$$



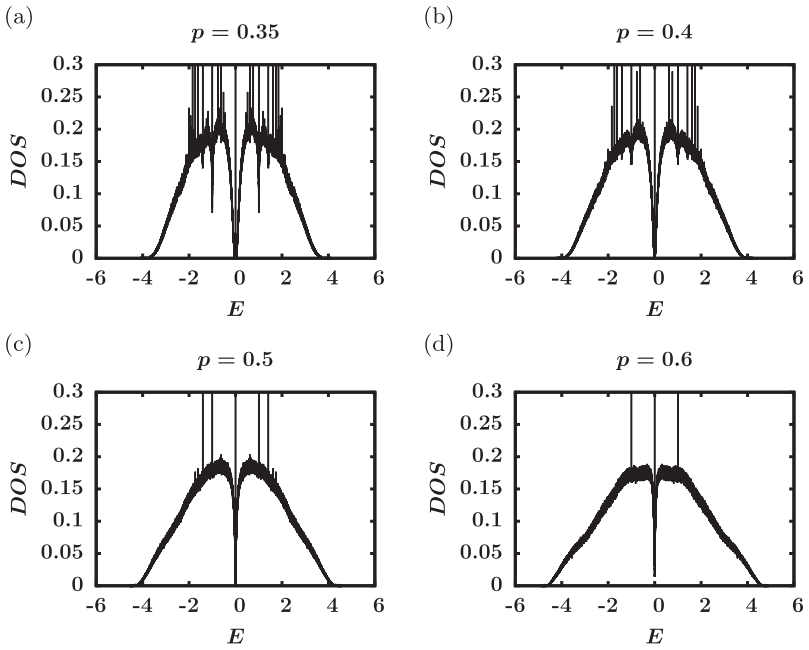
$$\psi_n \approx \exp\left(-\frac{n}{\xi_{q1D}(L)}\right)$$

Energy level statistics or multifractal + finite size scaling

- even when the transfer matrix is not applicable, we can diagonalize the Hamiltonian
- energy level statistics for the nearest energy spacing $P(s)$
  - localized states: Poisson, $P(s)=\exp(-s)$
  - delocalized states: Wigner-Dyson, $P(s) \doteqdot A\, s^{\beta}\exp(-s^2)$, β=1, 2, and 4 depending on the universality class
- multifractal analysis (A. Rodriguez et al., PRL (‘10), PRB (‘11), Ujfalusi and Varga, PRB ‘14)

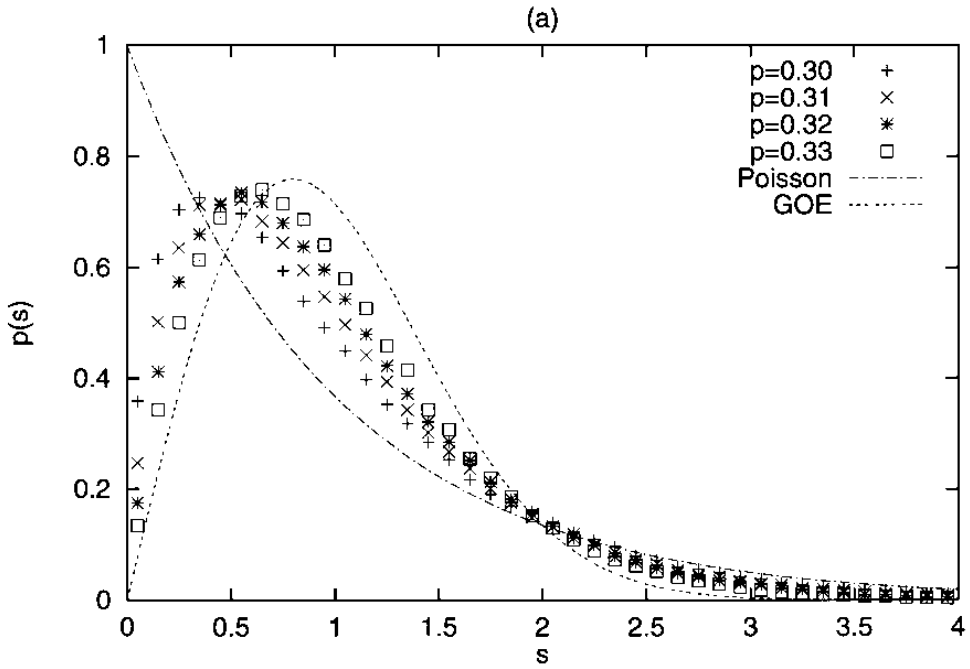
dos and level spacing for quantum percolation

DoS



Ujfalusi and Varga, PRB ‘14

Level spacing

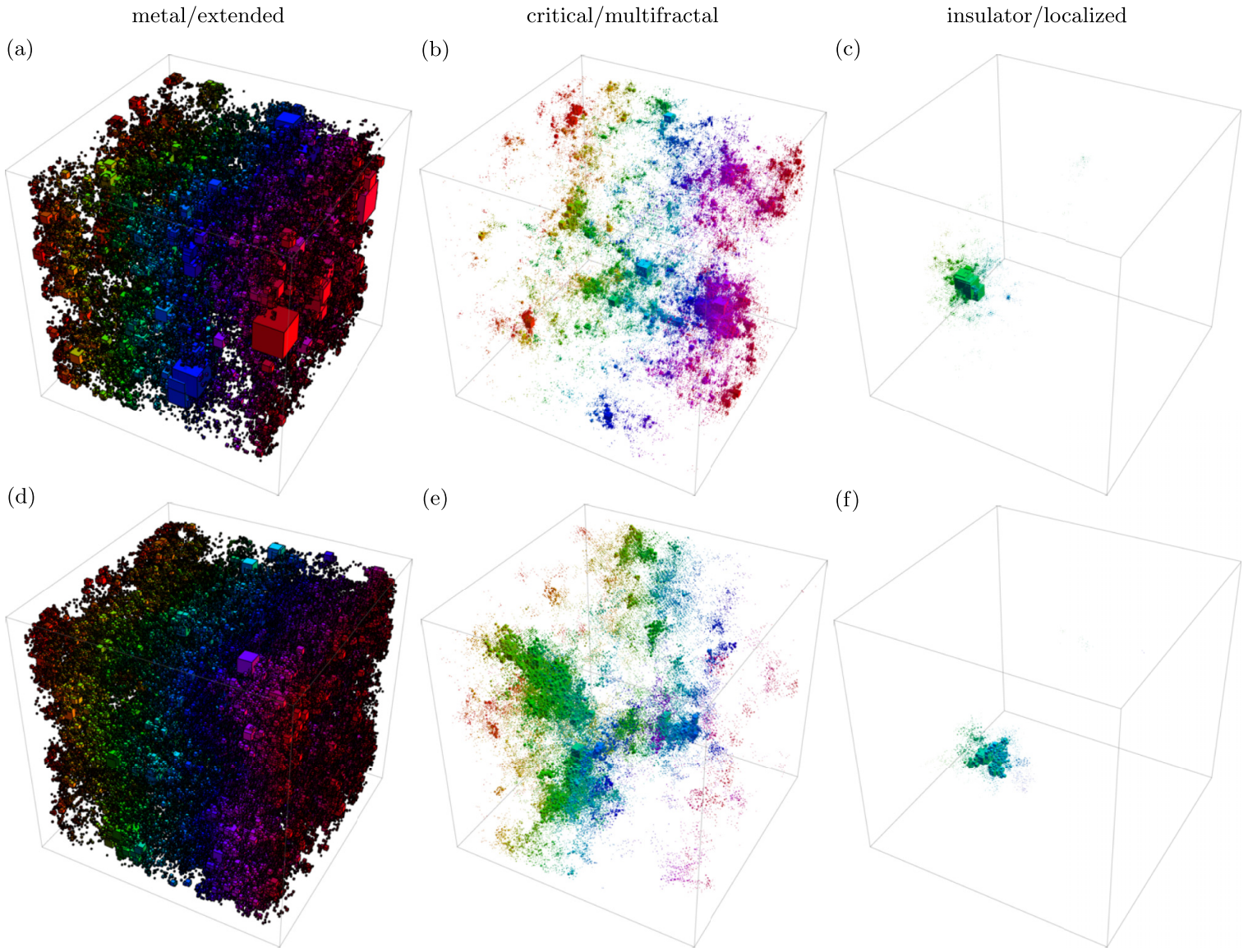


Kaneko and Ohtsuki, JPSJ ‘99

multifractality

Anderson model

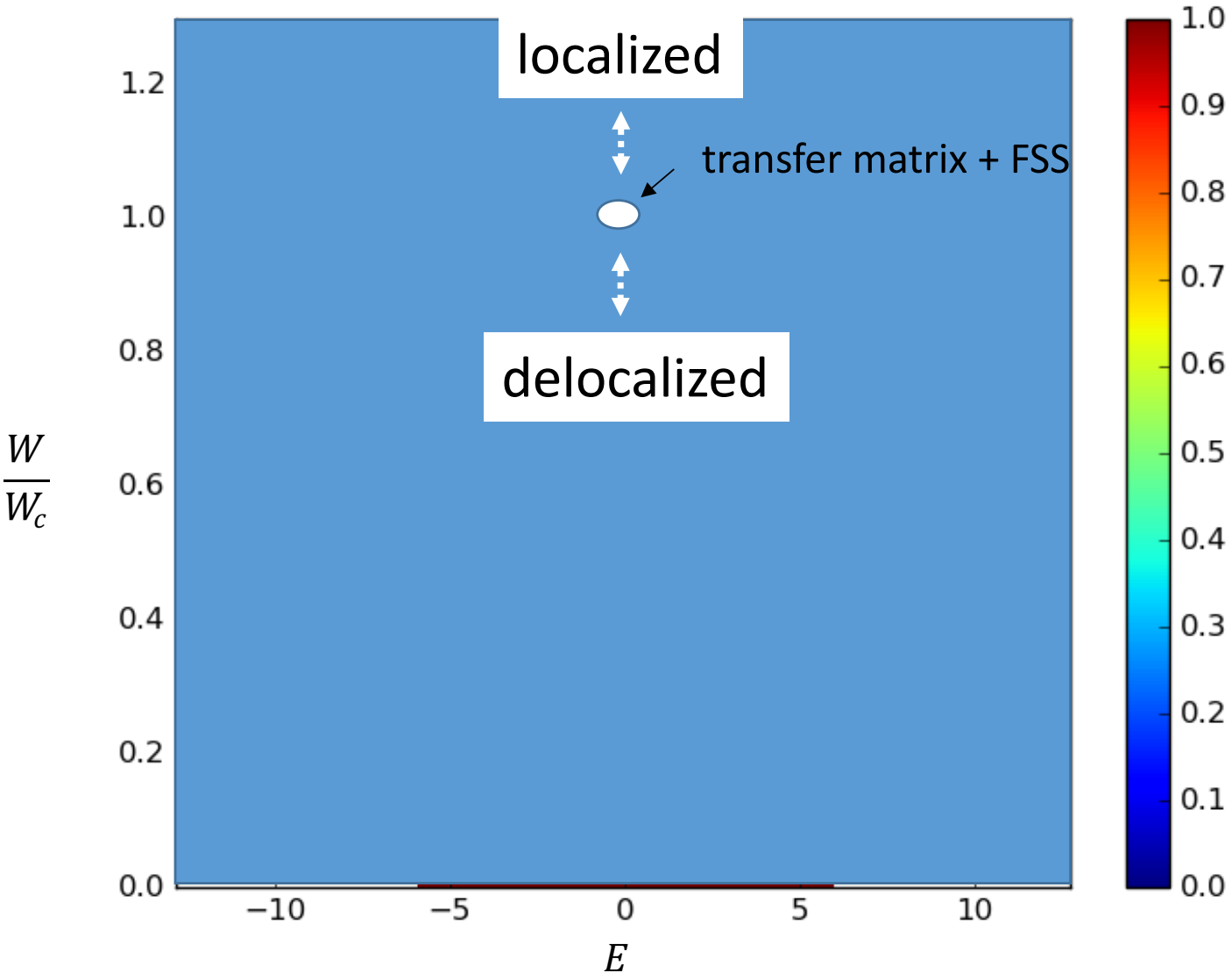
quantum percolation

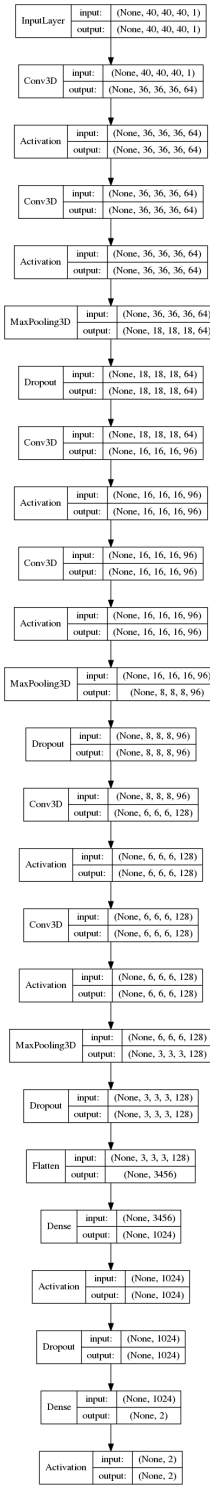


Ujfalusi and Varga, PRB ‘14

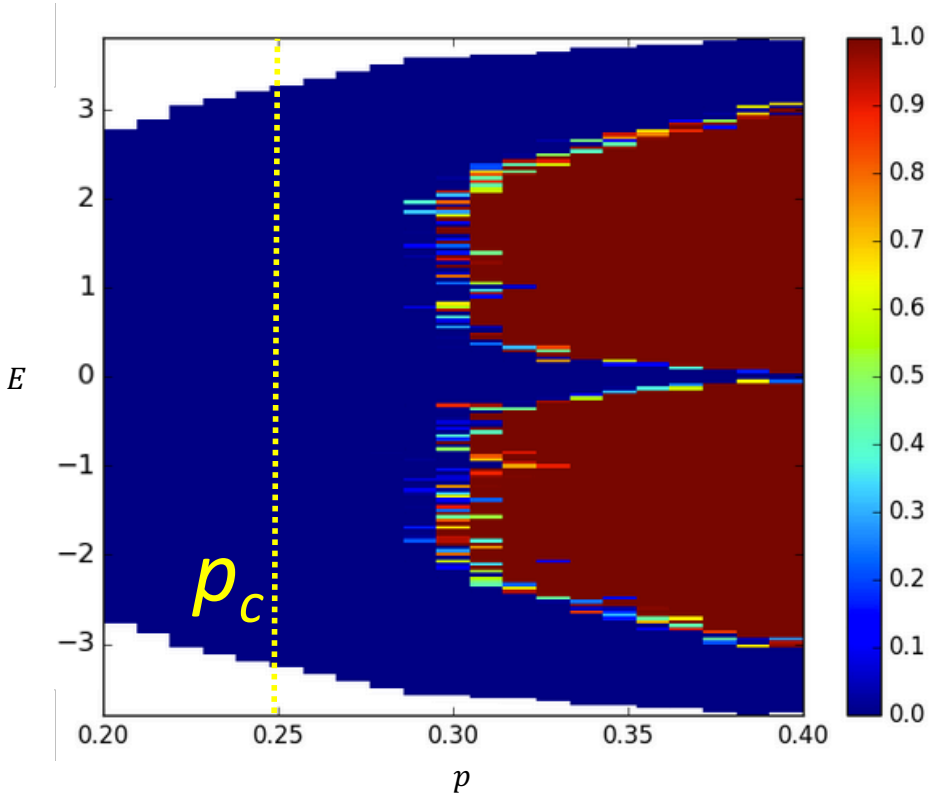
Deep (6 weight layer) CNN

3D Anderson transition

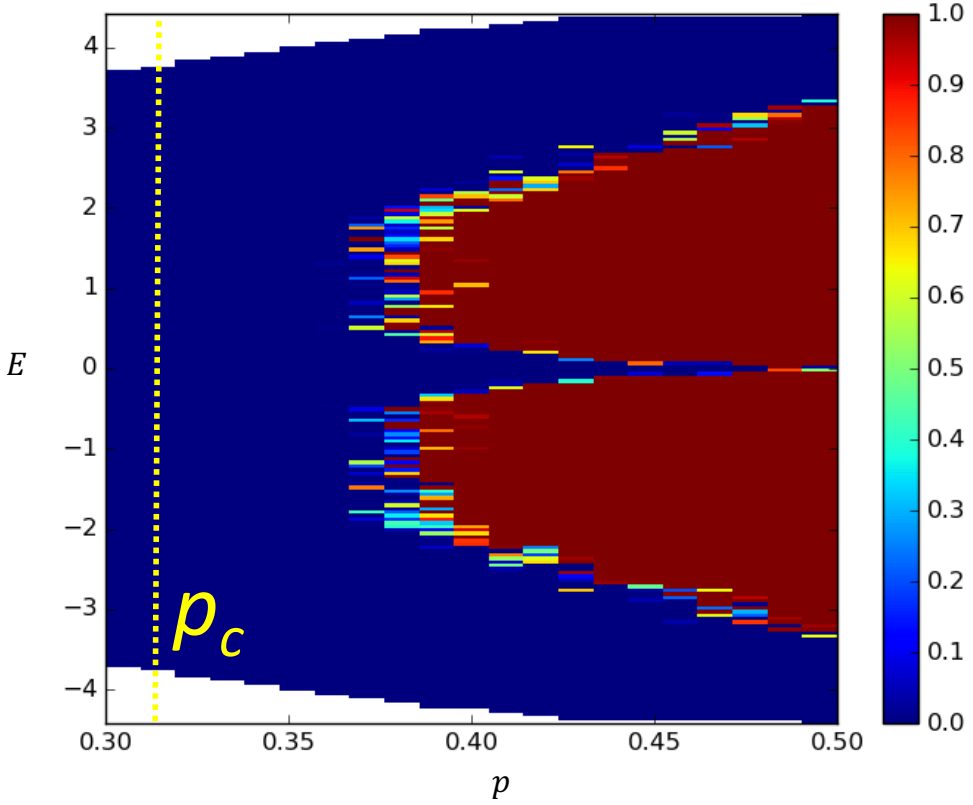




Applying the results of 3D Anderson transition to quantum percolation



Bond percolation



Site percolation

Comparing the methods of drawing phase diagram

- Finite size scaling (Slevin and Ohtsuki, NJP '14)
  - Define a nondimensional quantity $\Lambda(L, E, W, ..)$ such as conductance.
  - Plot $\Lambda(L, E, W, ..)$ as a function *of* E, W, etc. with different system sizes L.
  - Analyze $\Lambda(L, E, W, ..)$. Scaling invariant point is the phase boundary.
  - Precise estimate of critical point and critical exponents.
- Machine learning method: complementary to FSS
  - Simple analysis.
    - [python train.py;] python test.py; python dataArrange.py; python plot.py
  - Wider applicability.
  - Once trained, can draw phase diagrams for different parameters.
  - Detection of states on the phase boundary.
  - Only rough estimate of the phase boundaries. No critical exponent.
  - Too many tuning parameters like number of hidden layers, convolution size, pooling size, bias, padding, ….