Provably efficient machine learning for quantum many-body problems

Presenter: Hsin-Yuan (Robert) Huang

Joint work with Richard Kueng, Giacomo Torlai, Victor V. Albert, John Preskill

arXiv:2106.12627, 2021

IQIM INSTITUTE FOR QUANTUM INFORMATION AND MATTER Caltech UNIVERSITY OF MARYLAND aws JKU JOHANNES KEPLER UNIVERSITY LINZ

Motivation

- Machine learning (ML) has received great attention in the quantum community these days.
- Yet, many fundamental questions remain to be answered.

Classical ML
for quantum physics/chemistry

The goal 🎯:
Solve challenging
quantum many-body problems
better than
traditional classical algorithms

“Solving the quantum many-body problem with artificial neural networks.” *Science* 355.6325 (2017): 602-606.
"Learning phase transitions by confusion." *Nature Physics* 13.5 (2017): 435-439.

Motivation

- Machine learning (ML) has received great attention in the quantum community these days.
- Yet, many fundamental questions remain to be answered.

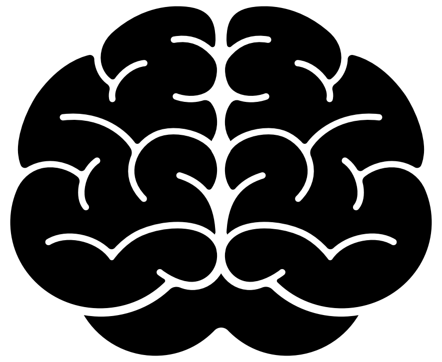
Classical ML
for quantum physics/chemistry

The question 👁:
Can ML algorithms be more useful
than non-ML algorithms in
physically relevant problems?

“Solving the quantum many-body problem with artificial neural networks.” *Science* 355.6325 (2017): 602-606.
"Learning phase transitions by confusion." *Nature Physics* 13.5 (2017): 435-439.

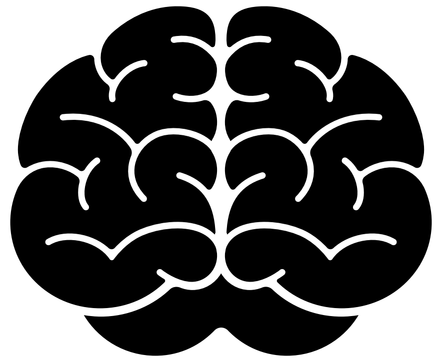
Outline

- Review on classical shadow formalism
- Training machines to predict ground states (theory+experiments)
- Training machines to classify quantum phases of matter (theory+experiments)



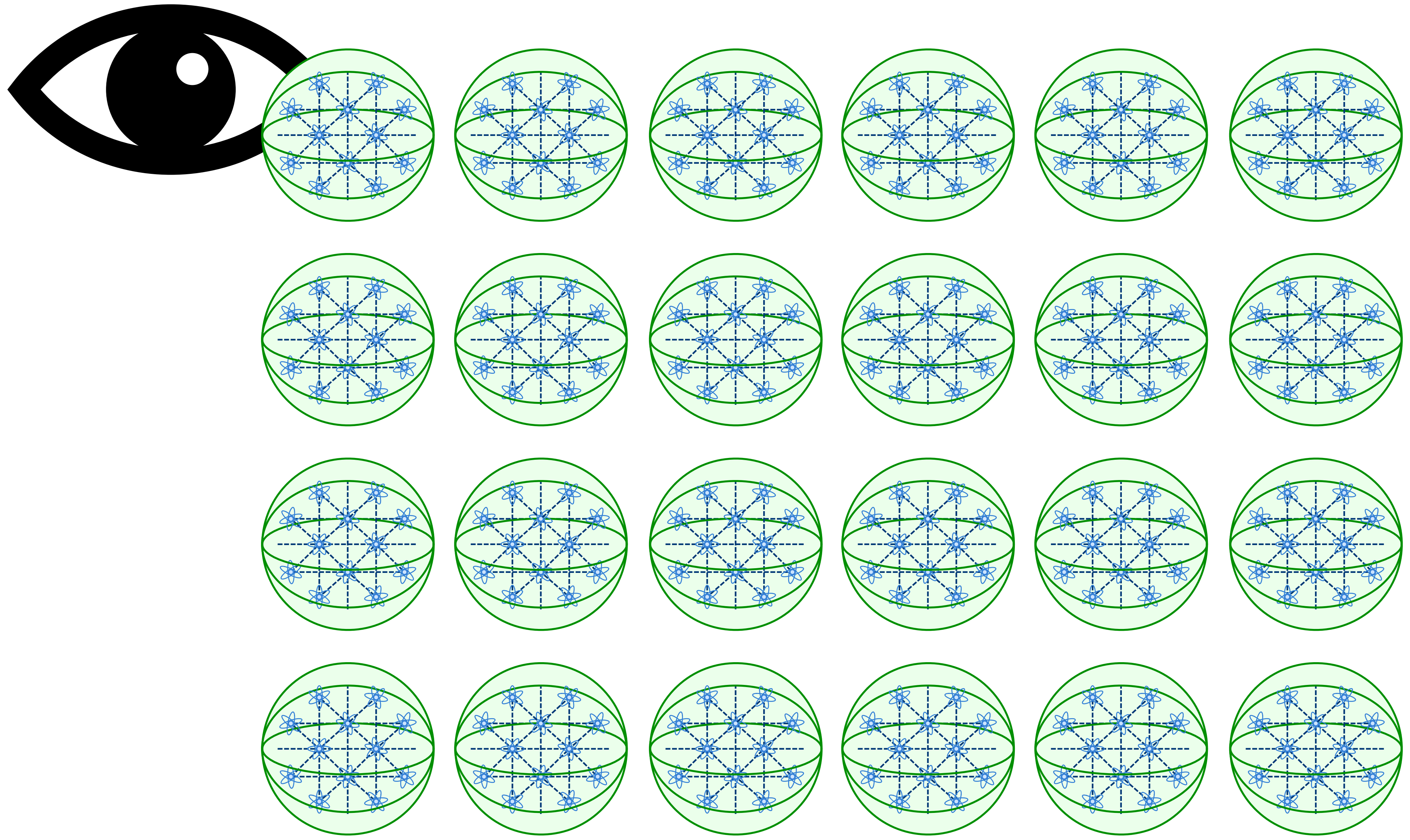
Outline

- Review on classical shadow formalism
- Training machines to predict ground states (theory+experiments)
- Training machines to classify quantum phases of matter (theory+experiments)



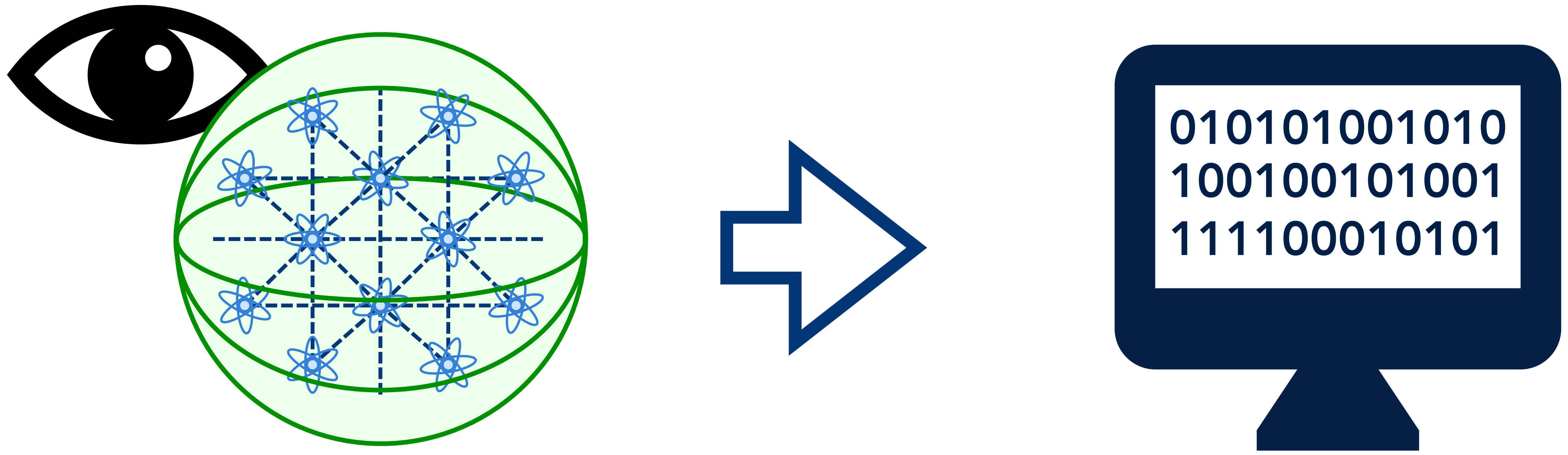
Classical shadow formalism

- How can classical machines “see” quantum many-body systems?



Classical shadow formalism

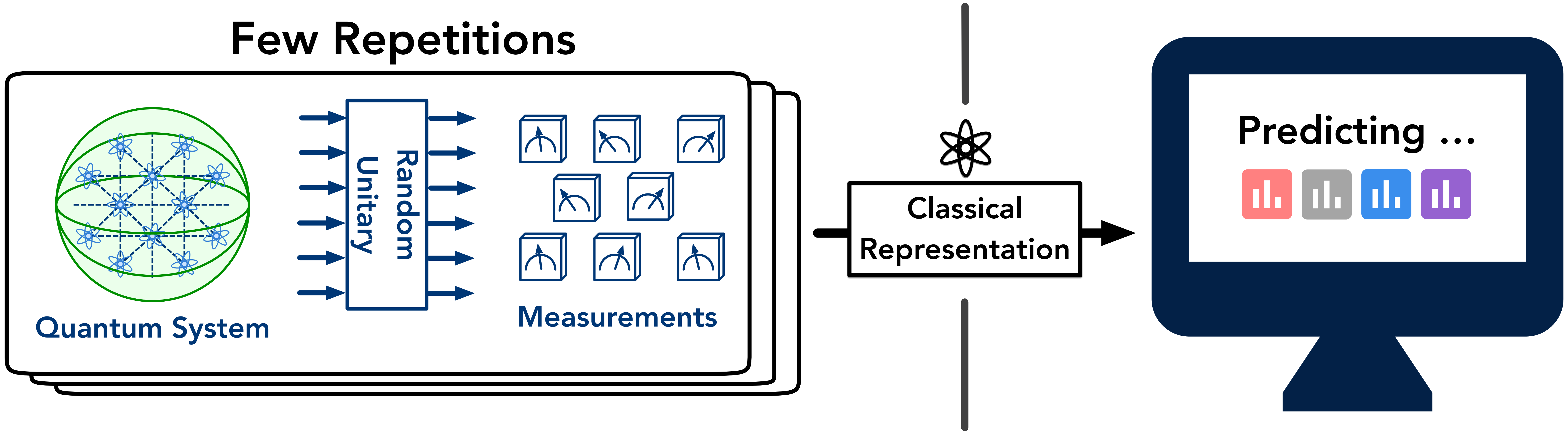
- What do we mean by “seeing” a quantum system?
- Converting the quantum system to a classical form that accurately captures many properties of the quantum system.



Unknown
Quantum System

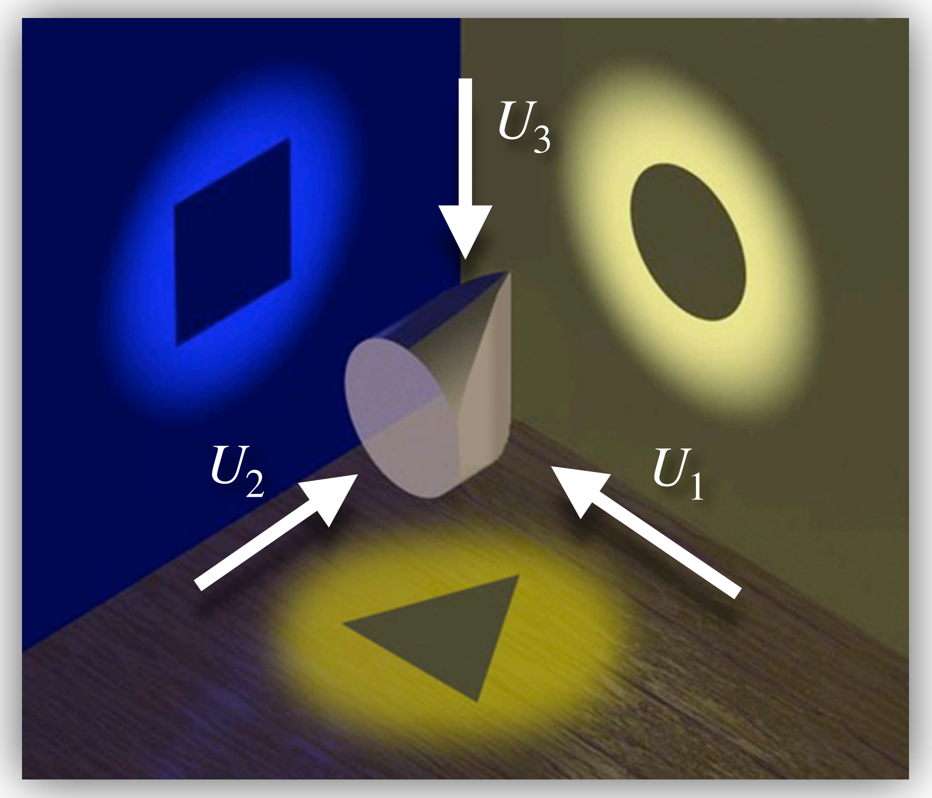
Efficient
Classical Representation

Classical shadow formalism



Data Acquisition Phase

Prediction Phase



Algorithm for predicting $\mathrm{tr}(O\rho)$: (median-of-means)

Compute $X_i = \mathrm{tr}(O\mathcal{M}^{-1}(|s_i\rangle\langle s_i|)), \forall i = 1,\ldots,T.$

Predict $\hat{o} = \text{median}\left(\frac{1}{T/K}\sum_{i=1}^{T/K} X_i, \ldots, \frac{1}{T/K}\sum_{i=T-T/K+1}^{T} X_i\right).$

[HKP20] Hsin-Yuan Huang, Richard Kueng, John Preskill. *Predicting many properties of a quantum system from very few measurements*, Nature Physics, 2020.

Classical shadow formalism

Theorem 1 [HKP20]

The procedure guarantees the following.

1. Given hyperparamers $B, M, \epsilon > 0$, the procedure learns a classical representation of an unknown quantum state ρ from

$$T = \mathcal{O}(B \log(M)/\epsilon^2) \text{ measurements.}$$

2. Subsequently, given any $O_1, \ldots, O_M$ with $\max \|O_i\|^2_{\text{shadow}} \leq B$, the procedure can use the classical representation to predict $\hat{o}_1, \ldots, \hat{o}_M$, where $|\hat{o}_i - \text{tr}(O_i \rho)| < \epsilon$, for all i.

[HKP20] Hsin-Yuan Huang, Richard Kueng, John Preskill. *Predicting many properties of a quantum system from very few measurements*, Nature Physics, 2020.

Classical shadow formalism

Theorem 1 [HKP20]

The procedure guarantees the following.

1. Given hyperparamers $B, M, \epsilon > 0$, the procedure learns a classical representation of an unknown quantum state ρ from

$$T = \mathcal{O}(B \log(M)/\epsilon^2) \text{ measurements.}$$

2. Subsequently, given any $O_1, \ldots, O_M$ with $\max \|O_i\|^2_{\text{shadow}} \leq B$, the procedure can use the classical representation to predict $\hat{o}_1, \ldots, \hat{o}_M$, where $|\hat{o}_i - \operatorname{tr}(O_i \rho)| < \epsilon$, for all i.

For example:

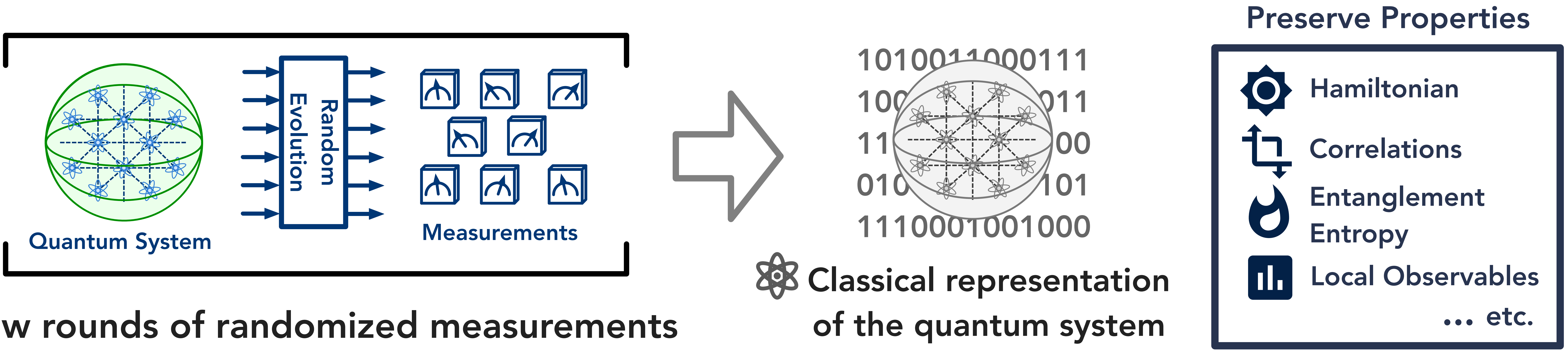
- $M = 10^6$, $B = 1$, then naively we need $10^6/\epsilon^2$ measurements.
- This theorem shows that we only need $6 \log(10)/\epsilon^2$ measurements.

The observables can be chosen after the measurements.

[HKP20] Hsin-Yuan Huang, Richard Kueng, John Preskill. *Predicting many properties of a quantum system from very few measurements*, Nature Physics, 2020.

Classical shadow with randomized Pauli measurements

- One randomized Pauli measurement: Measure every qubit in a random X/Y/Z bases.
- After T randomized Pauli measurements on an n-qubit system ρ, we construct classical shadow $\sigma_T(\rho) = \frac{1}{T}\sum_{t=1}^{T} \sigma_1^{(t)} \otimes \ldots \otimes \sigma_n^{(t)}$, where $\sigma_i^{(t)} = 3|s_i^{(t)}\rangle\langle s_i^{(t)}| - I$ and $s_i^{(t)} \in \{0, 1, +, -, y+, y-\}$.
- $\sigma_T(\rho)$ is a $2^n \times 2^n$ random matrix with $\mathbb{E}\sigma_T(\rho) = \rho$ and takes $\mathcal{O}(nT)$ classical bits to represent.



Few rounds of randomized measurements

Classical representation of the quantum system

Preserve Properties

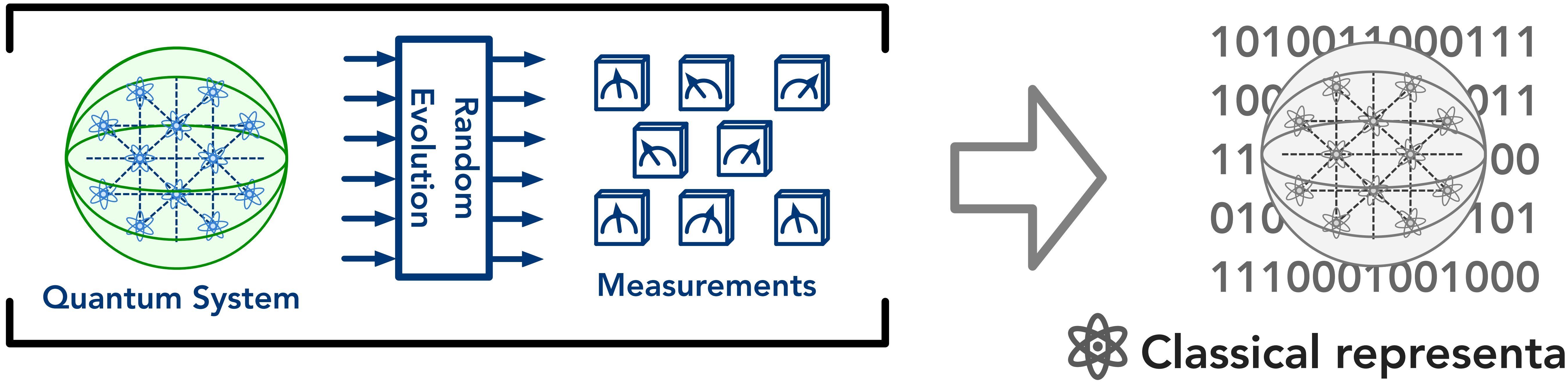
- Hamiltonian
- Correlations
- Entanglement Entropy
- Local Observables
- … etc.

[HKP20] Hsin-Yuan Huang, Richard Kueng, John Preskill. *Predicting many properties of a quantum system from very few measurements*, Nature Physics, 2020.

Classical shadow with randomized Pauli measurements

See Richard Kueng's QIP tutorial

- One randomized Pauli measurement: Measure every qubit in a random X/Y/Z bases.
- After T randomized Pauli measurements on an n-qubit system ρ, we construct classical shadow $\sigma_T(\rho) = \frac{1}{T}\sum_{t=1}^{T} \sigma_1^{(t)} \otimes \ldots \otimes \sigma_n^{(t)}$, where $\sigma_i^{(t)} = 3|s_i^{(t)}\rangle\langle s_i^{(t)}| - I$ and $s_i^{(t)} \in \{0, 1, +, -, y+, y-\}$.
- $\sigma_T(\rho)$ is a $2^n \times 2^n$ random matrix with $\mathbb{E}\sigma_T(\rho) = \rho$ and takes $\mathcal{O}(nT)$ classical bits to represent.



Few rounds of randomized measurements

Classical representation of the quantum system

Preserve Properties

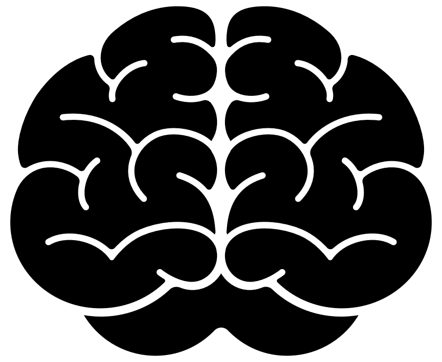
- Hamiltonian
- Correlations
- Entanglement Entropy
- Local Observables

… etc.

[HKP20] Hsin-Yuan Huang, Richard Kueng, John Preskill. *Predicting many properties of a quantum system from very few measurements*, Nature Physics, 2020.

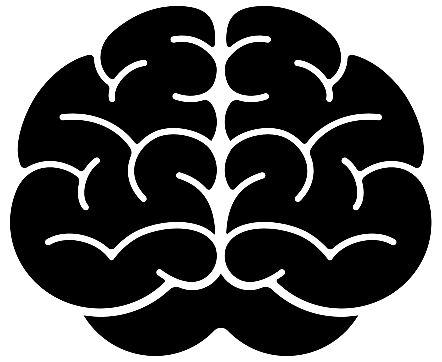
Outline

- Review on classical shadow formalism
- Training machines to predict ground states (theory+experiments)
- Training machines to classify quantum phases of matter (theory+experiments)



Outline

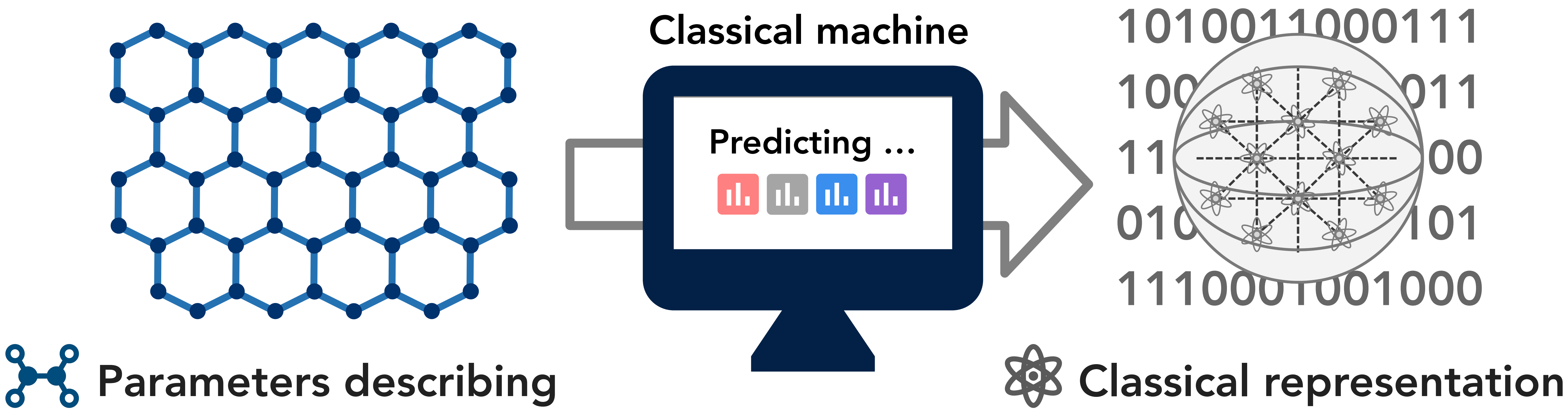
- Review on classical shadow formalism
- Training machines to predict ground states (theory+experiments)
- Training machines to classify quantum phases of matter (theory+experiments)



Predicting ground states: Task

For ML, $x \rightarrow H(x)$ can be unknown

- Given parameters x that describes an n-qubit Hamiltonian $H(x)$, the machine predicts a classical representation (e.g., classical shadow) of the ground state $\rho(x)$ of $H(x)$.
- $x \in [-1,1]^m$ describes laser intensities, few-body interactions, magnetic fields, etc.



Parameters describing a physical Hamiltonian

Classical representation of the ground state

Computational hardness

- This is a ***very hard*** problem.

Computational hardness

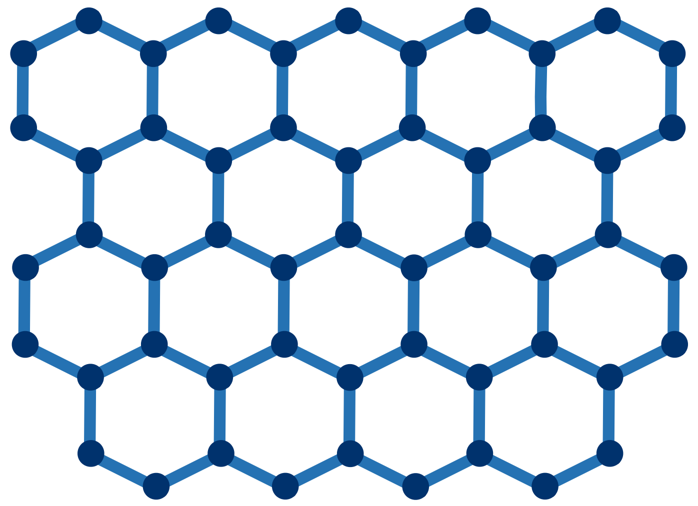
- This is a ***very hard*** problem.
- Consider a smooth class of n-qubit **2D** Hamiltonians $H(x)$ with **spectral gap 1**, and the machine only predicts **1-body** observable O in ground state $\rho(x)$.

Computational hardness

- This is a ***very hard*** problem.
- Consider a smooth class of n-qubit **2D** Hamiltonians $H(x)$ with **spectral gap 1**, and the machine only predicts **1-body** observable O in ground state $\rho(x)$.

1D

2D

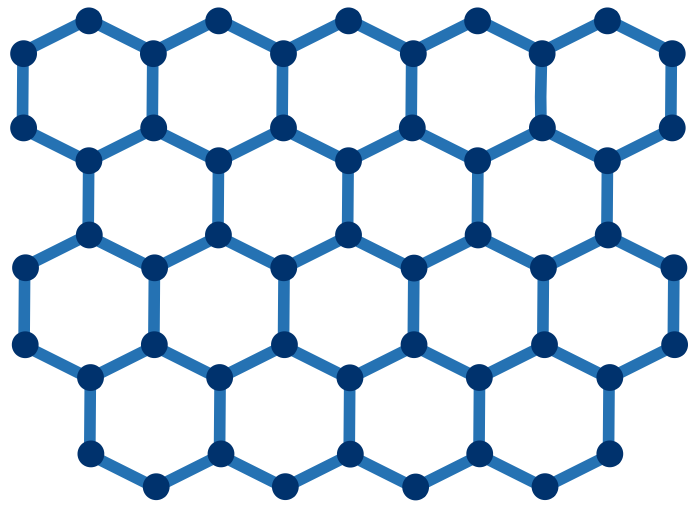


Computational hardness

- This is a ***very hard*** problem.
- Consider a smooth class of n-qubit **2D** Hamiltonians $H(x)$ with **spectral gap 1**, and the machine only predicts **1-body** observable O in ground state $\rho(x)$.
- Furthermore, we only care about average prediction error $\mathbb{E}_{x\sim[-1,1]^m}|\mathscr{A}(x,O)-\mathrm{Tr}(O\rho(x))|^2$.

1D

2D

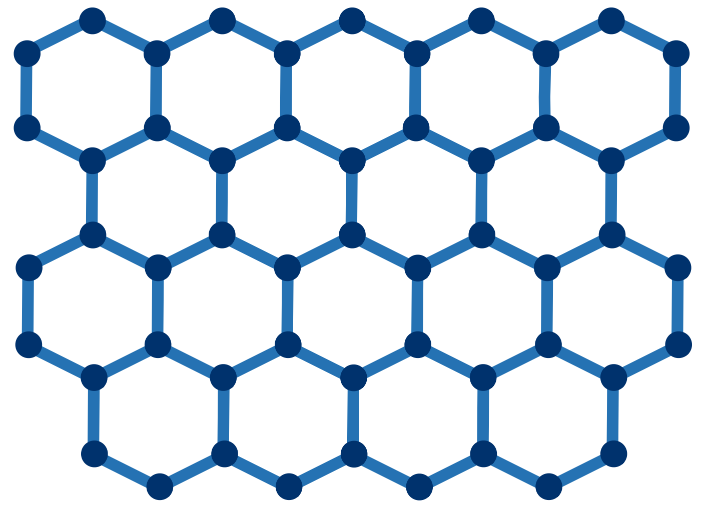


Computational hardness

- This is a ***very hard*** problem.
- Consider a smooth class of n-qubit **2D** Hamiltonians $H(x)$ with **spectral gap 1**, and the machine only predicts **1-body** observable O in ground state $\rho(x)$.
- Furthermore, we only care about average prediction error $\mathbb{E}_{x\sim[-1,1]^m}|\mathscr{A}(x,O)-\mathrm{Tr}(O\rho(x))|^2$.

1D

2D



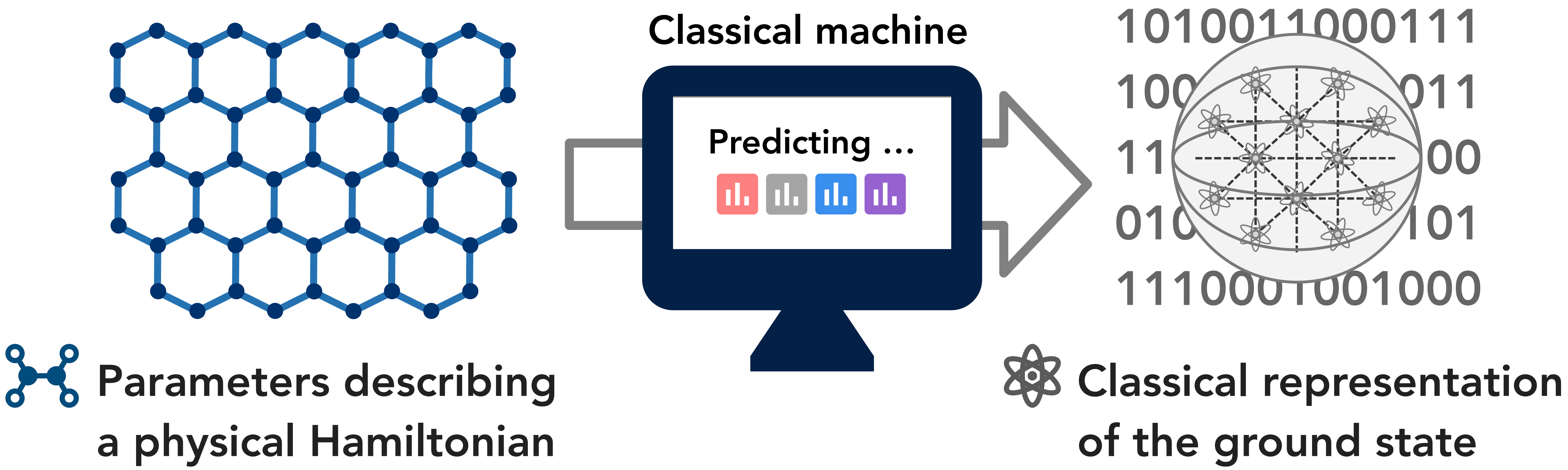
Proposition 1

Assuming RP $\neq$ NP, then no randomized classical algorithm can achieve an average prediction error $\leq 1/4$ within $\mathrm{poly}(m,n)$ time.

RP $\neq$ NP: NP-complete problems cannot be solved in randomized polynomial time.

Predicting ground states: Task

- Can classical ML algorithms do something useful for this very challenging problem?
- E.g., can they even solve these problems in polynomial time?



Classical machine

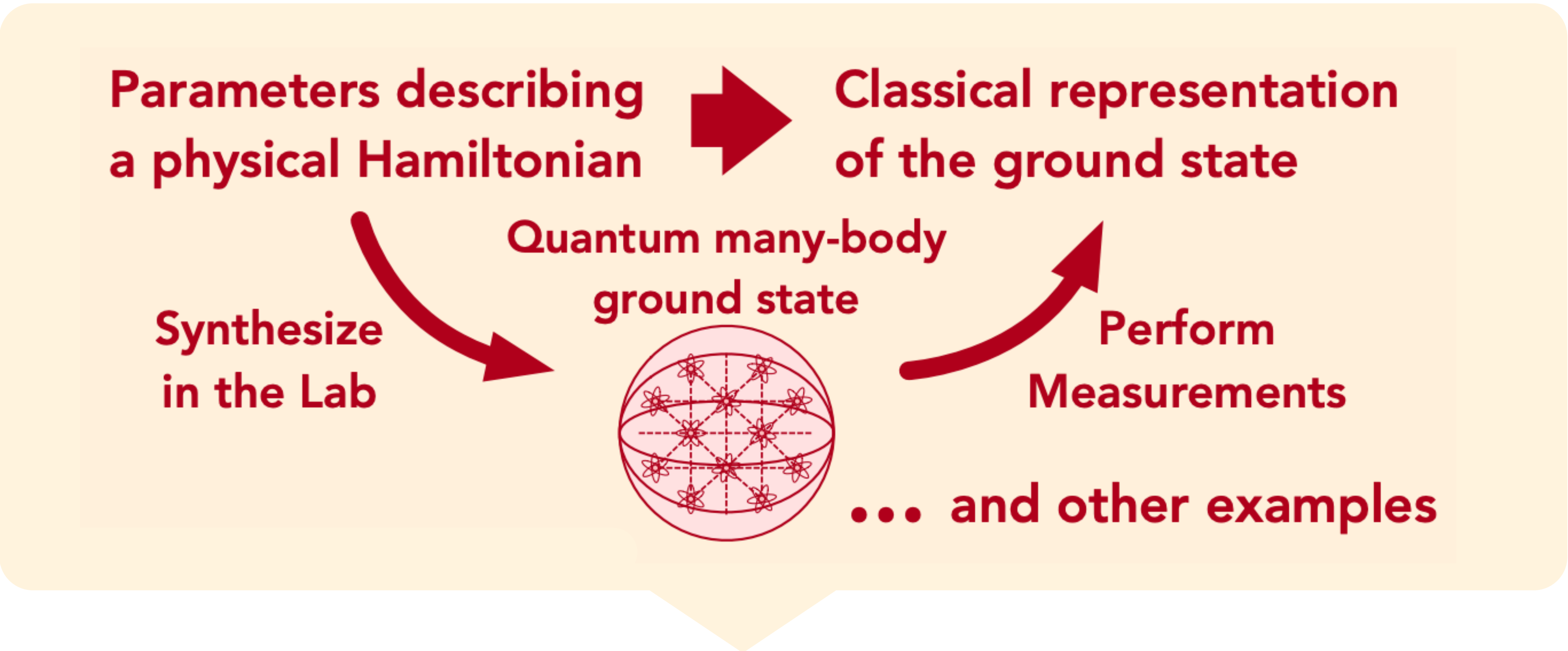
Predicting …

Parameters describing a physical Hamiltonian

Classical representation of the ground state

Predicting ground states: Task

Training data: $\{x_\ell \to \sigma_T(\rho(x_\ell))\}_{\ell=1}^{N}$



Parameters describing a physical Hamiltonian

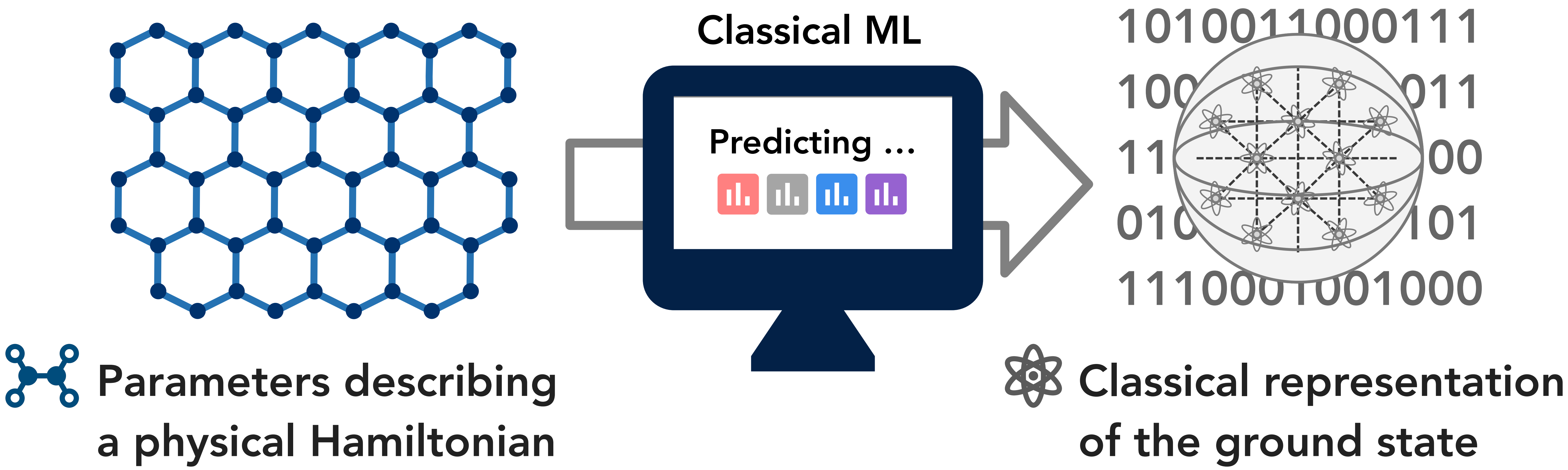
Classical representation of the ground state

Quantum many-body ground state

Synthesize in the Lab

Perform Measurements

… and other examples



Classical ML

Predicting …

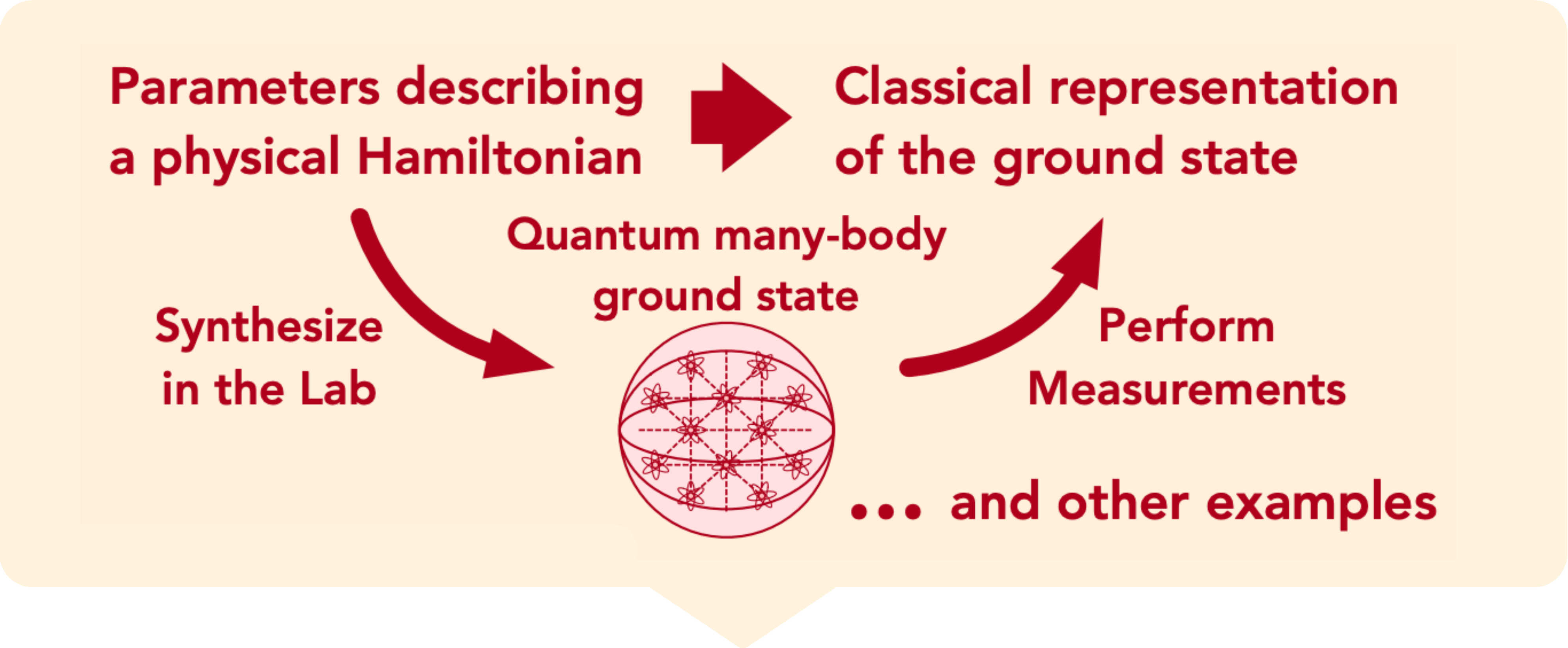
Parameters describing a physical Hamiltonian

Classical representation of the ground state

Predicting ground states: Task

Classical shadow: an efficient representation for $2^n \times 2^n$ matrix $\rho(x_\ell)$

Training data: $\{x_\ell \to \boxed{\sigma_T(\rho(x_\ell))}\}_{\ell=1}^{N}$



Parameters describing a physical Hamiltonian → Classical representation of the ground state

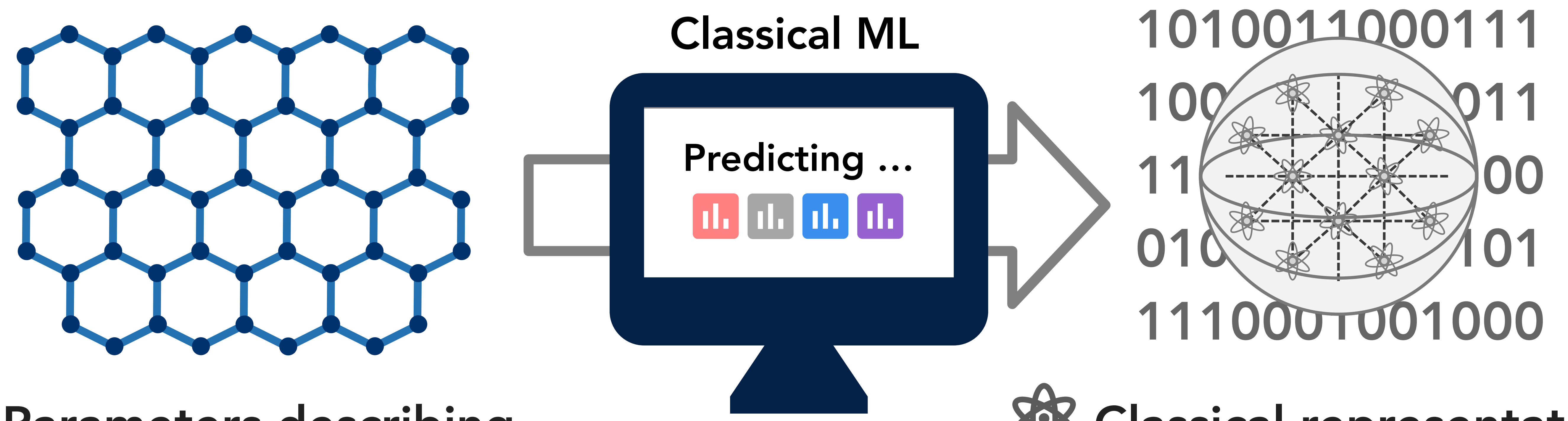
Synthesize in the Lab

Quantum many-body ground state

Perform Measurements

… and other examples

Classical ML



Parameters describing a physical Hamiltonian

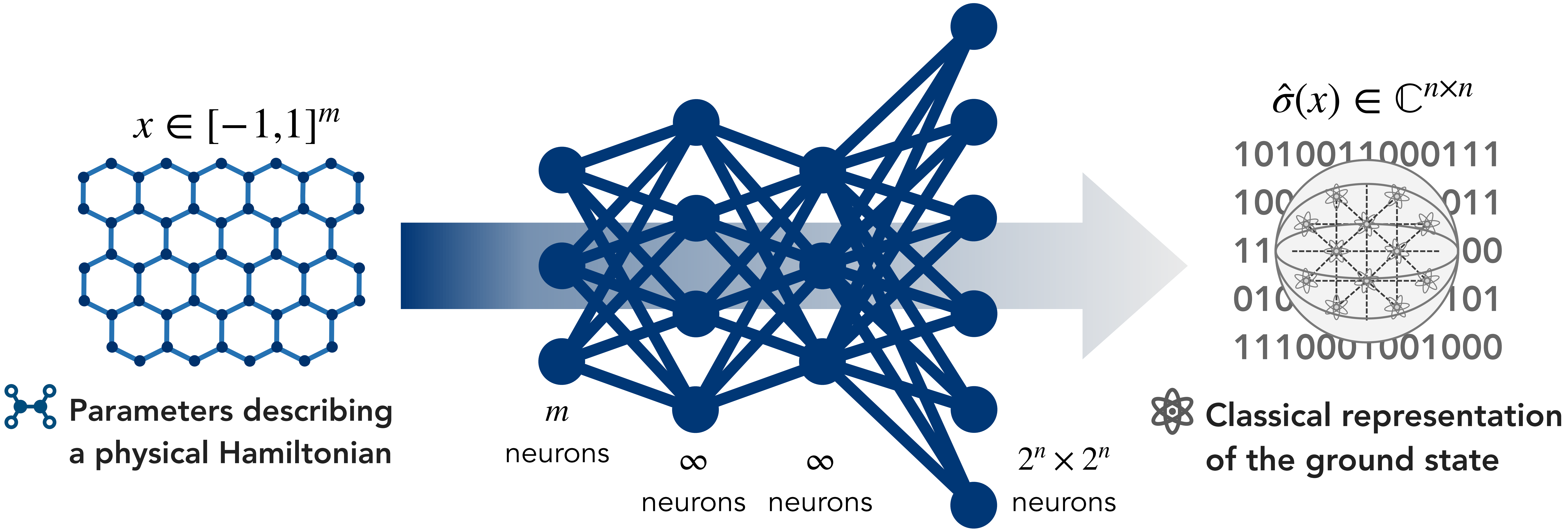
Classical representation of the ground state

Predicting ground states: ML

$x_\ell \in [-1,1]^m$

Training data: $\{x_\ell \rightarrow \sigma_T(\rho(x_\ell))\}_{\ell=1}^{N}$

- We consider training an ML model that takes in an m-dim vector x and outputs an efficient representation of a $2^n \times 2^n$-size matrix $\hat{\sigma}(x)$.
- The ML model needs to be trained within time polynomial in n, m.



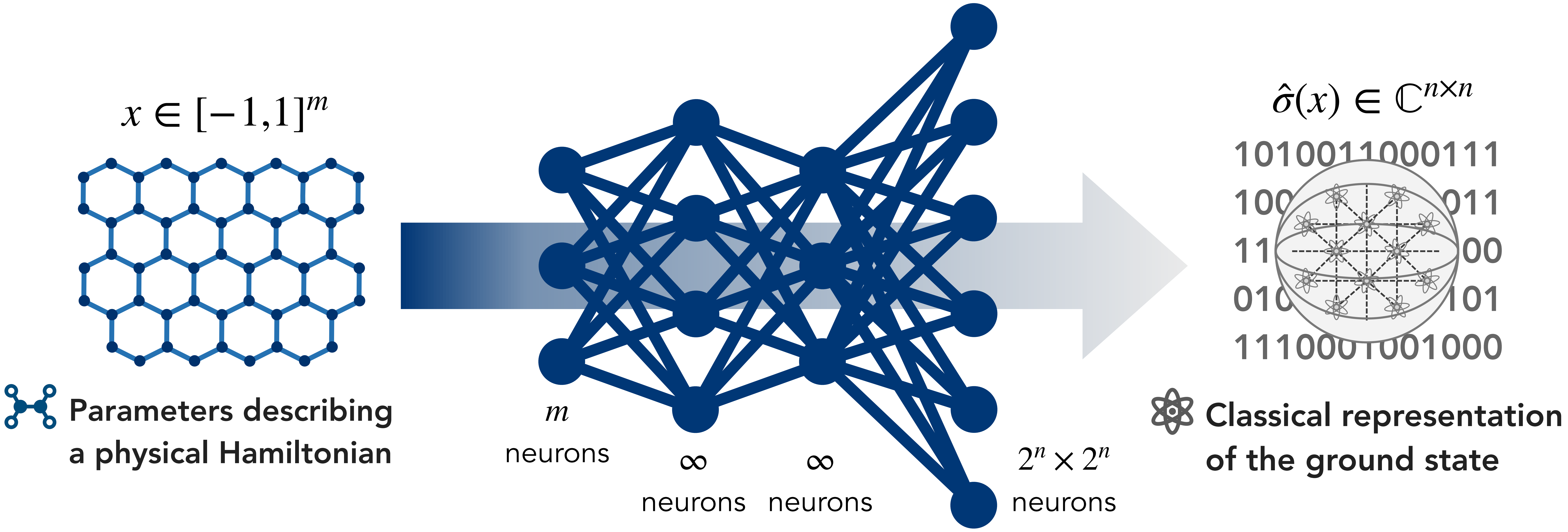
Predicting ground states: ML

$x_\ell \in [-1,1]^m$

Training data: $\{x_\ell \to \sigma_T(\rho(x_\ell))\}_{\ell=1}^N$

- Suppose we train a neural network $\hat{\sigma}_W(x)$ (illustrated below) with infinitely many neurons in the hidden layers and exponentially many neurons in the output layer.

- We can train the bizarrely large model in time polynomial in n, m.



$x \in [-1,1]^m$

Parameters describing a physical Hamiltonian

m neurons

∞ neurons

∞ neurons

$2^n \times 2^n$ neurons

$\hat{\sigma}(x) \in \mathbb{C}^{n\times n}$

Classical representation of the ground state

Predicting ground states: ML

$x_\ell \in [-1,1]^m$

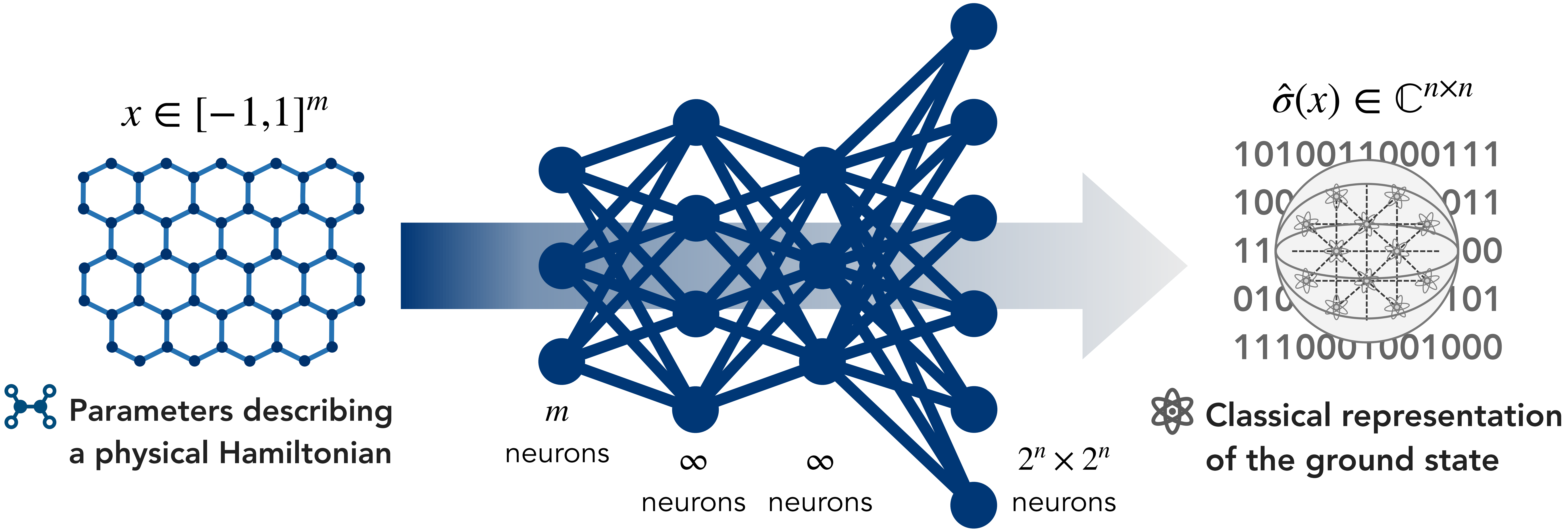
Training data: $\{x_\ell \to \sigma_T(\rho(x_\ell))\}_{\ell=1}^{N}$

- We show that the neural network after training actually has an analytical form given by

$$\hat{\sigma}^{\mathrm{NN}}(x) = \underset{\hat{\sigma}_W}{\operatorname{argmin}} \sum_{\ell=1}^{N} \|\hat{\sigma}_W(x_\ell) - \sigma_T(\rho(x_\ell))\|_2^2 = \sum_{\ell=1}^{N} \kappa^{\mathrm{NN}}(x, x_\ell)\sigma_T(\rho(x_\ell))$$

where the learned function $\kappa^{\mathrm{NN}}(x, x_\ell) \in \mathbb{R}$ can be obtained efficiently; based on [JGH18].

[JGH18] "Neural tangent kernel: Convergence and generalization in neural networks." arXiv preprint arXiv:1806.07572 (2018).



$x \in [-1,1]^m$

Parameters describing a physical Hamiltonian

m neurons

∞ neurons

∞ neurons

$2^n \times 2^n$ neurons

$\hat{\sigma}(x) \in \mathbb{C}^{n \times n}$

Classical representation of the ground state

Predicting ground states: ML

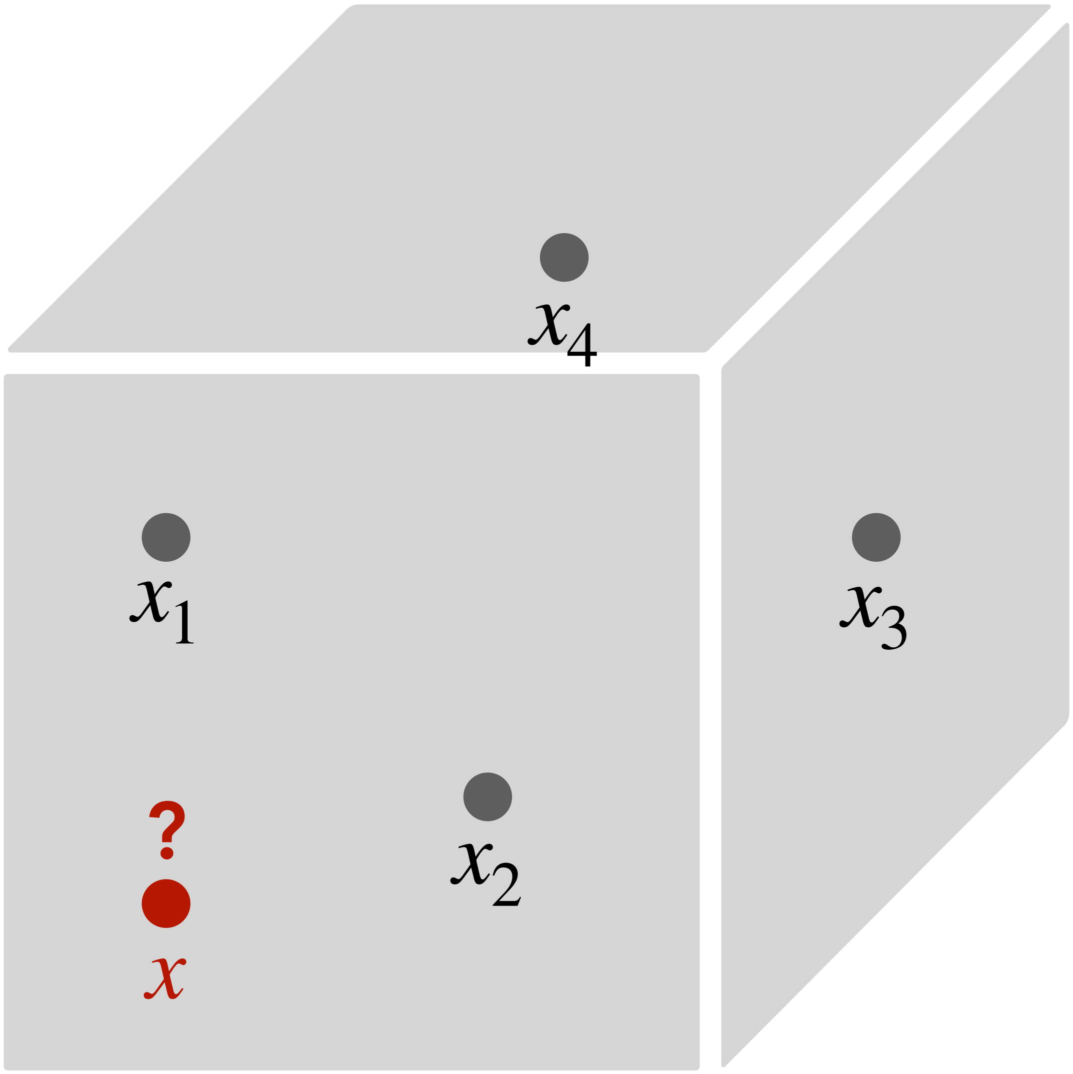
$x_\ell \in [-1,1]^m$

Training data: $\{x_\ell \to \sigma_T(\rho(x_\ell))\}_{\ell=1}^N$

- Many machine learning models can be shown to yield the following analytical form as the global minimum of the optimization (training):

$$\hat{\sigma}(x) = \sum_{\ell=1}^{N} \kappa(x, x_\ell)\sigma_T(\rho(x_\ell))$$

where $\kappa(x, x_\ell) \in \mathbb{R}$ is a learned function for how to extrapolate the known examples to the full space.



Predicting ground states: ML

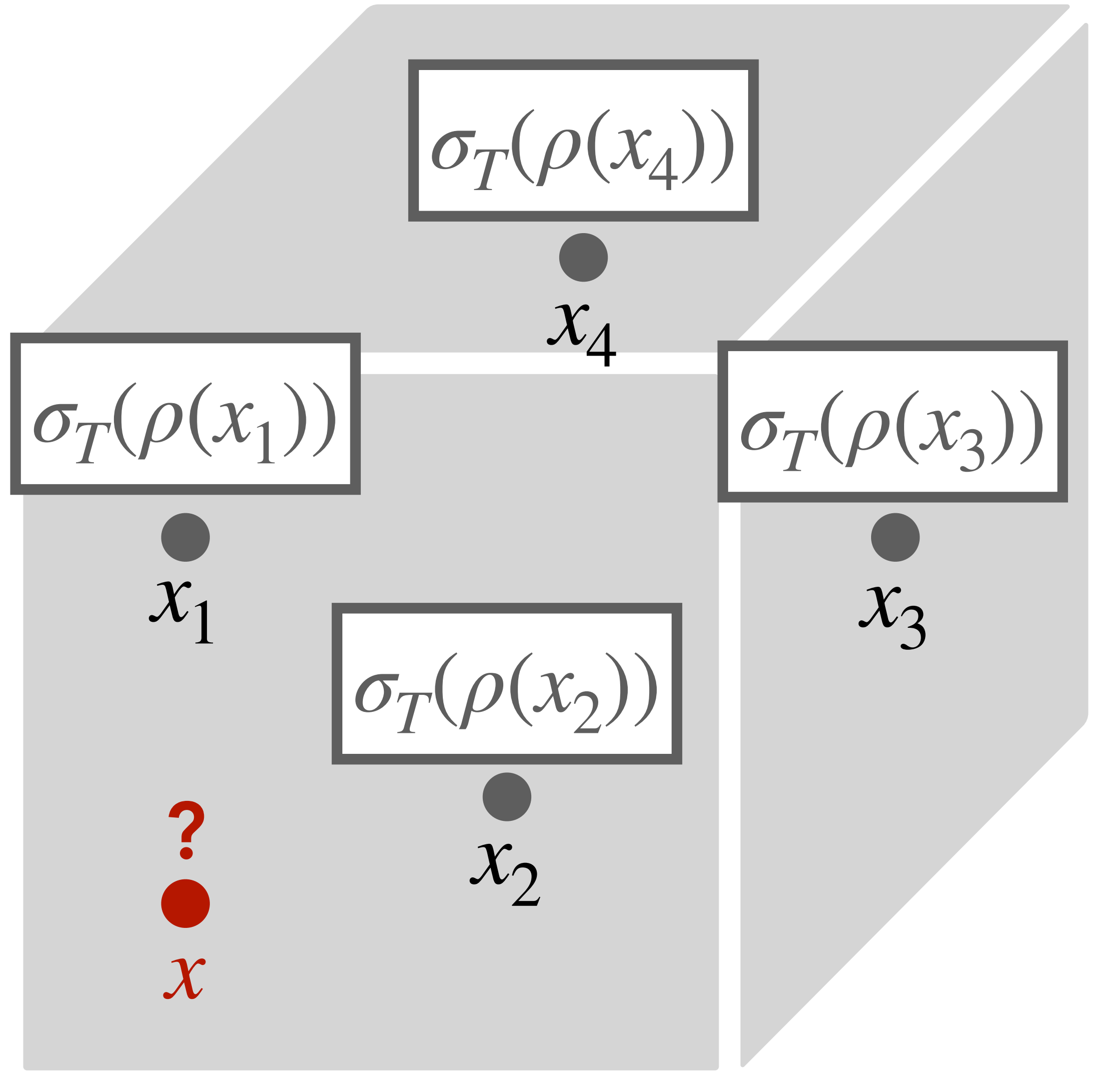
$x_\ell \in [-1,1]^m$

Training data: $\{x_\ell \to \sigma_T(\rho(x_\ell))\}_{\ell=1}^N$

- Many machine learning models can be shown to yield the following analytical form as the global minimum of the optimization (training):

$$\hat{\sigma}(x) = \sum_{\ell=1}^{N} \kappa(x, x_\ell)\sigma_T(\rho(x_\ell))$$

where $\kappa(x, x_\ell) \in \mathbb{R}$ is a learned function for how to extrapolate the known examples to the full space.



Predicting ground states: ML

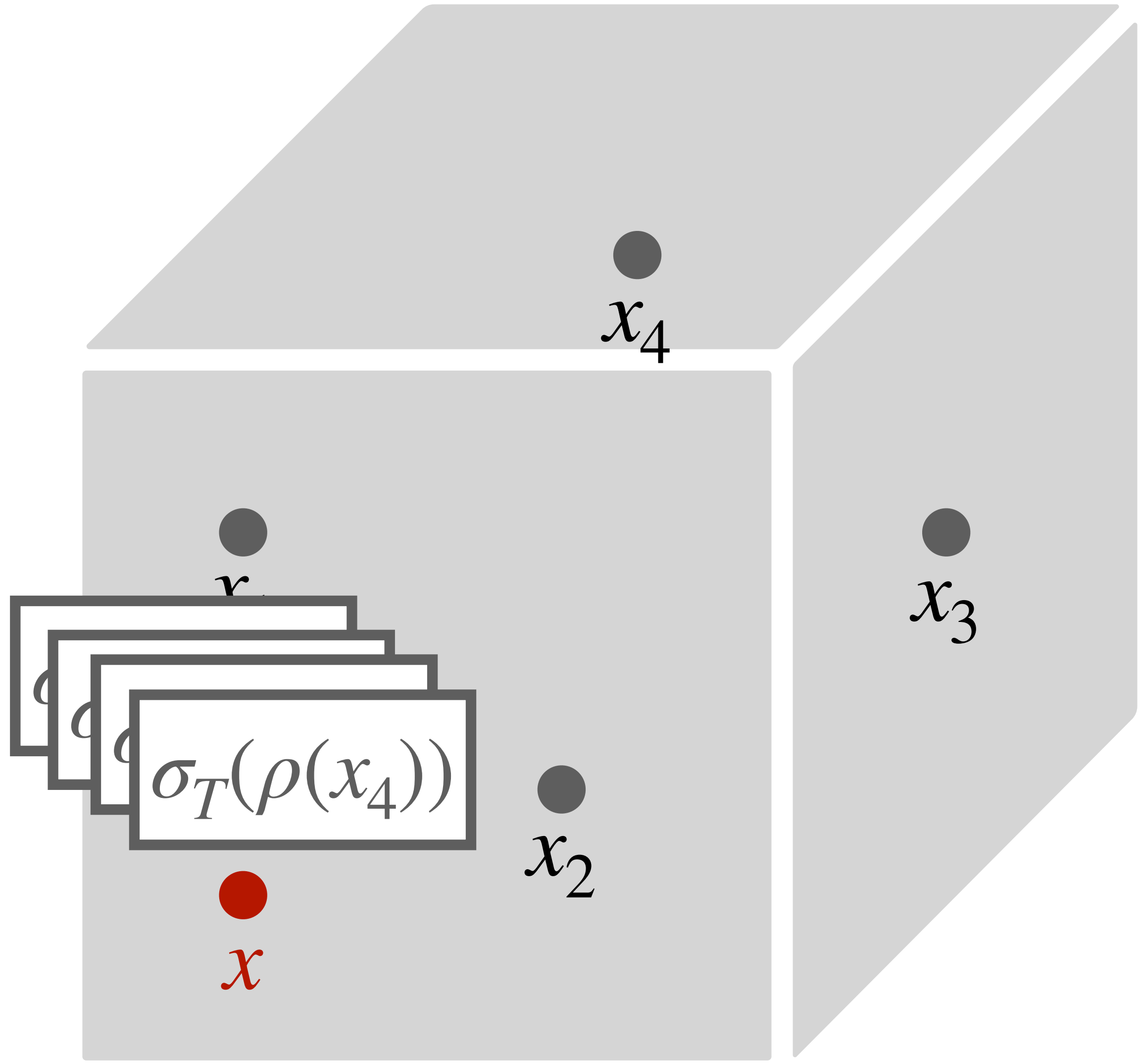
$x_\ell \in [-1,1]^m$

Training data: $\{x_\ell \rightarrow \sigma_T(\rho(x_\ell))\}_{\ell=1}^{N}$

- Many machine learning models can be shown to yield the following analytical form as the global minimum of the optimization (training):

$$\hat{\sigma}(x) = \sum_{\ell=1}^{N} \kappa(x, x_\ell)\sigma_T(\rho(x_\ell))$$

where $\kappa(x, x_\ell) \in \mathbb{R}$ is a learned function for how to extrapolate the known examples to the full space.



Predicting ground states: ML

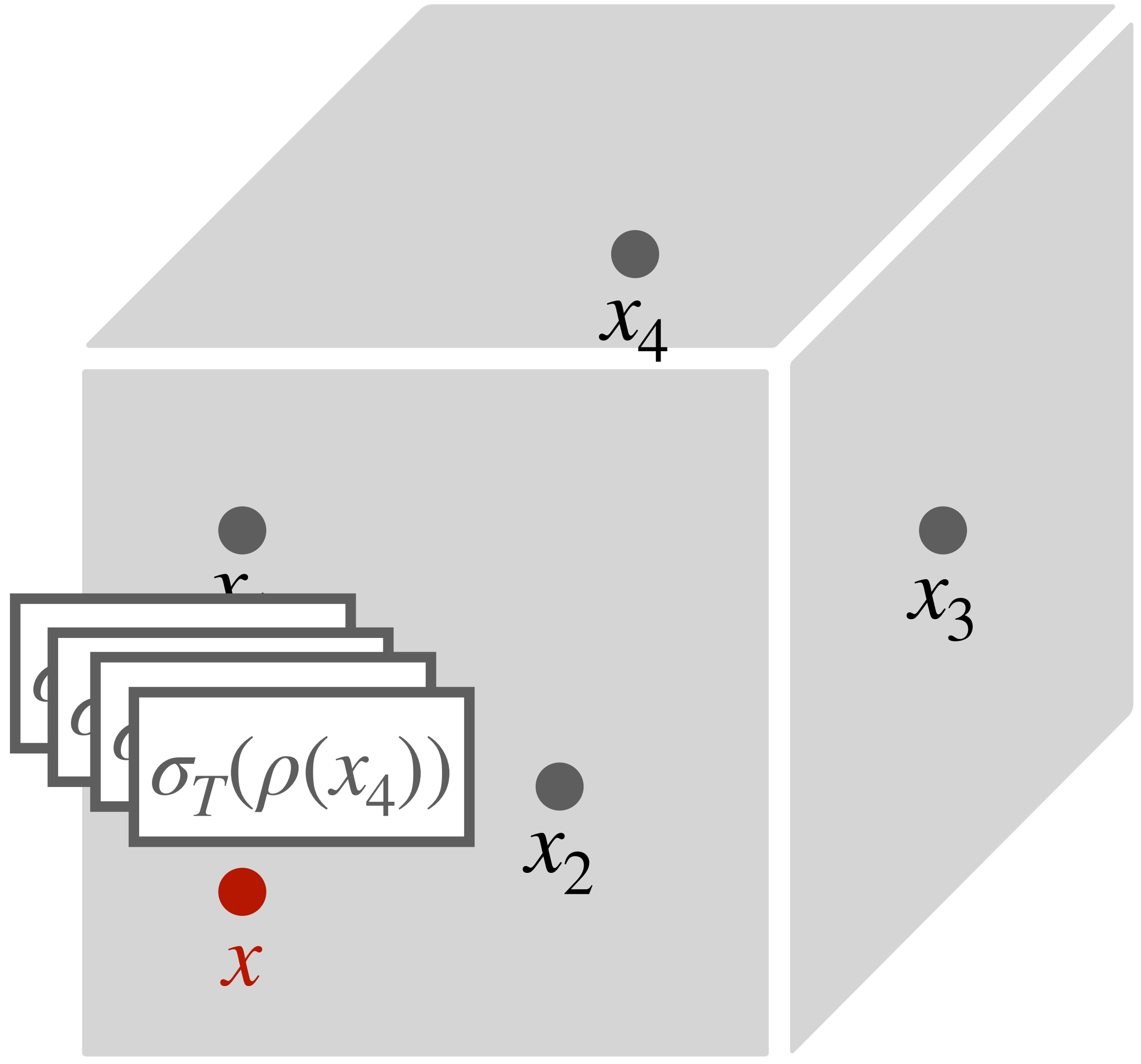
$x_\ell \in [-1,1]^m$

Training data: $\{x_\ell \rightarrow \sigma_T(\rho(x_\ell))\}_{\ell=1}^N$

- The ML model's prediction

$$\hat{\sigma}(x) = \sum_{\ell=1}^{N} \kappa(x, x_\ell)\sigma_T(\rho(x_\ell))$$

can be stored efficiently with $\mathcal{O}(nTN)$ bits; recall $\sigma_T(\rho(x_\ell))$ only require $\mathcal{O}(nT)$ bits.



Predicting ground states: Theorem

$x_\ell \in [-1,1]^m$

Training data: $\{x_\ell \to \sigma_T(\rho(x_\ell))\}_{\ell=1}^N$

- What can be prove about classical ML algorithm?

Proposition 1

Assuming RP $\neq$ NP, then no randomized classical algorithm can achieve an average prediction error $\leq 1/4$ within $\mathrm{poly}(m, n)$ time.

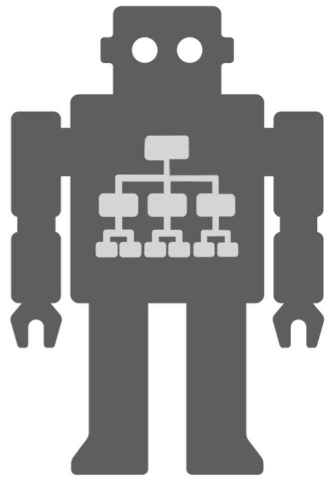
Classical algorithm

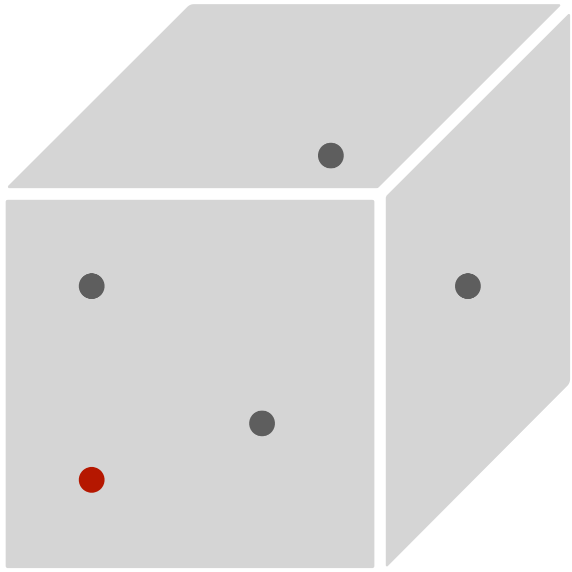
2D ⛔

spectral gap 1

1-body observable

average prediction error 1/4





Predicting ground states: Theorem

$x_\ell \in [-1,1]^m$

Training data: $\{x_\ell \to \sigma_T(\rho(x_\ell))\}_{\ell=1}^N$

- What can be prove about classical ML algorithm?

Theorem 1

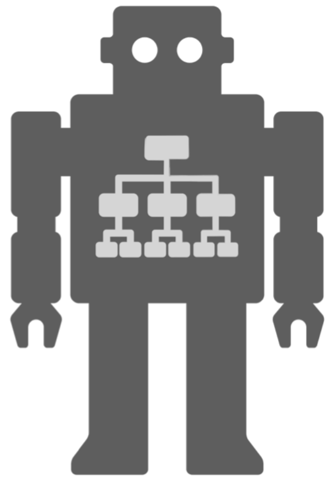
With $N = \mathrm{poly}(m)$ training data and $T = 1$ measurement on each $\rho(x_\ell)$, the ML model $\hat{\sigma}(x)$ can achieve an average prediction error $\leq \epsilon$. The training and prediction time are $\mathcal{O}(n\,\mathrm{poly}(m))$.

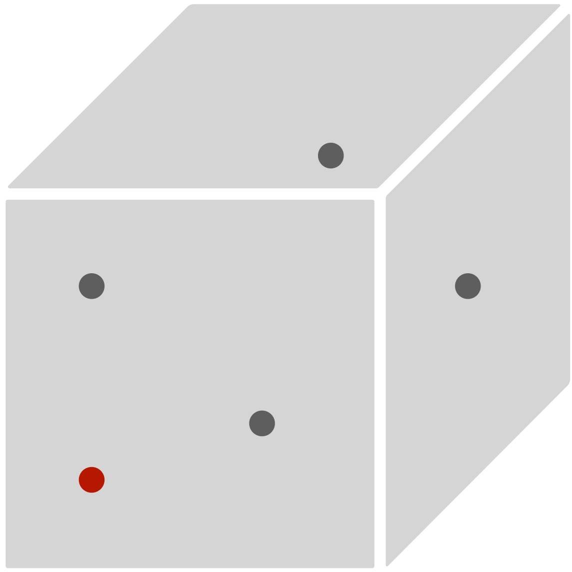
Classical algorithm

2D ⛔
spectral gap 1
1-body observable
average prediction error 1/4

Classical ML (trained with data)

any constant dimension ✓
any constant spectral gap
any few-body observable
any average prediction error $\epsilon = \mathcal{O}(1)$





Predicting ground states: Theorem

$x_\ell \in [-1,1]^m$

Training data: $\{x_\ell \to \sigma_T(\rho(x_\ell))\}_{\ell=1}^N$

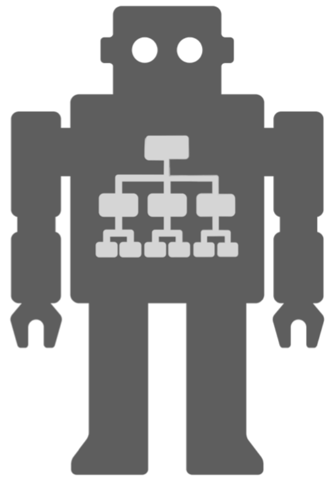
- What can be prove about classical ML algorithm?

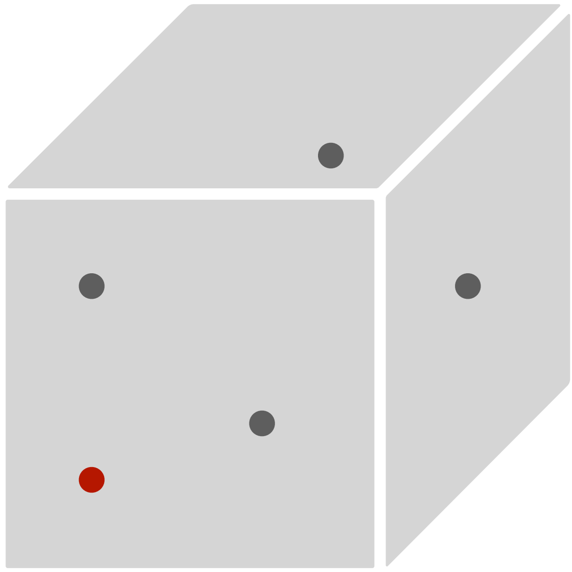
Theorem 1

With $N = \mathrm{poly}(m)$ training data and $T = 1$ measurement on each $\rho(x_\ell)$, the ML model $\hat{\sigma}(x)$ can achieve an average prediction error $\leq \epsilon$. The training and prediction time are $\mathcal{O}(n\,\mathrm{poly}(m))$.

Key steps in the proof of Theorem 1:

1. Constant spectral gap implies the ground state $\rho(x)$ varies "smoothly" with x (spectral flow + Lieb-Robinson bounds).
2. Learning-theoretic analysis to bound prediction error under "smoothness" guarantee (statistical analysis + #lattices in a m-dim. sphere).





Predicting ground states: Theorem

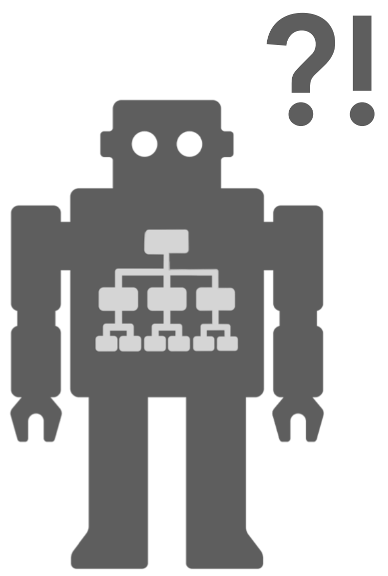
$x_\ell \in [-1,1]^m$

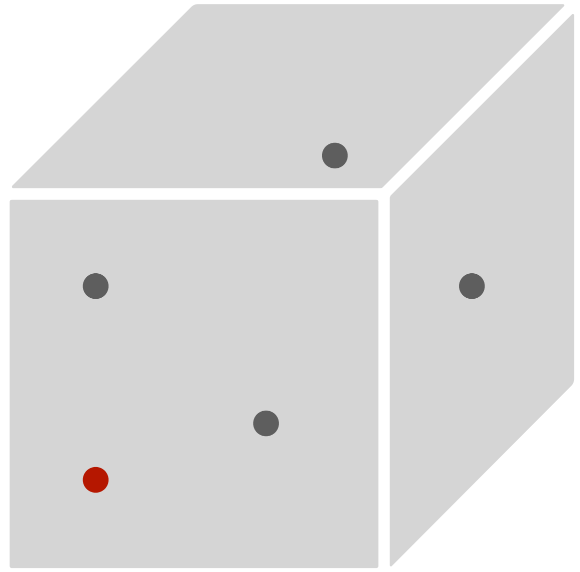
Training data: $\{x_\ell \to \sigma_T(\rho(x_\ell))\}_{\ell=1}^N$

- What can be prove about classical ML algorithm?

Classical algorithm		**Classical ML (trained with data)**	
2D	⛔	any constant dimension	✓
spectral gap 1		any constant spectral gap	
1-body observable		any few-body observable	
average prediction error 1/4		any average prediction error $\epsilon = \mathcal{O}(1)$	

Prop.1+Thm.1 says that classical poly-time ML (trained with data) can predict ground states better than **any** poly-time classical randomized algorithm.





Predicting ground states: Theorem

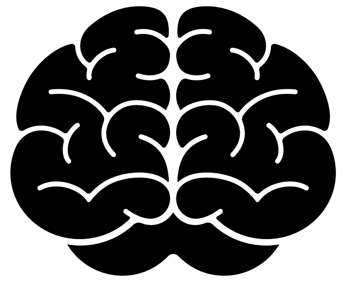
$x_\ell \in [-1,1]^m$

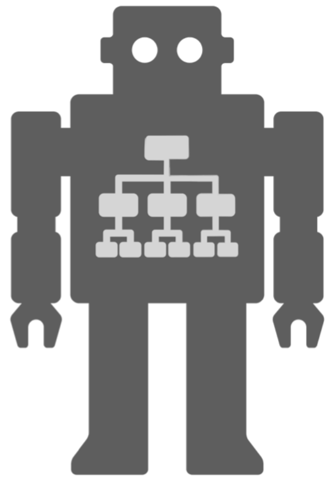
Training data: $\{x_\ell \to \sigma_T(\rho(x_\ell))\}_{\ell=1}^{N}$

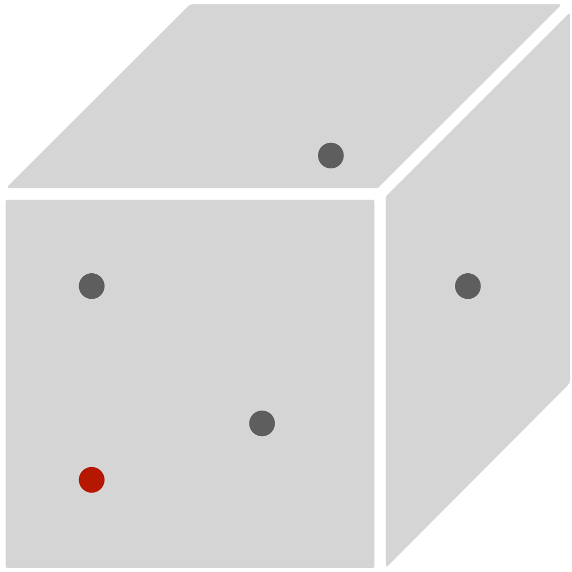
- What can be prove about classical ML algorithm?

Classical algorithm	**Classical ML (trained with data)**
2D ⛔	any constant dimension ✓
spectral gap 1	any constant spectral gap
1-body observable	any few-body observable
average prediction error 1/4	any average prediction error $\epsilon = \mathcal{O}(1)$

The question 👁:
Why ML can be more powerful than
non-ML algorithms?







Predicting ground states: Theorem

$x_\ell \in [-1,1]^m$

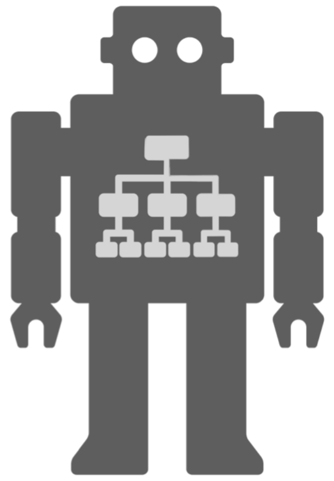
Training data: $\{x_\ell \to \sigma_T(\rho(x_\ell))\}_{\ell=1}^N$

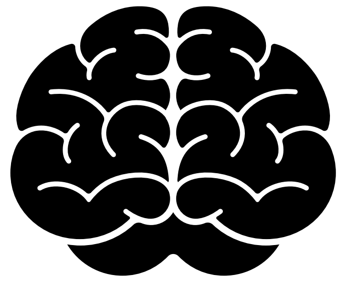
- What can be prove about classical ML algorithm?

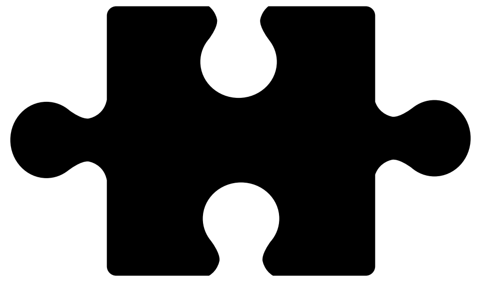
Classical algorithm ⛔	**Classical ML (trained with data)** ✓
2D	any constant dimension
spectral gap 1	any constant spectral gap
1-body observable	any few-body observable
average prediction error 1/4	any average prediction error $\epsilon = \mathcal{O}(1)$

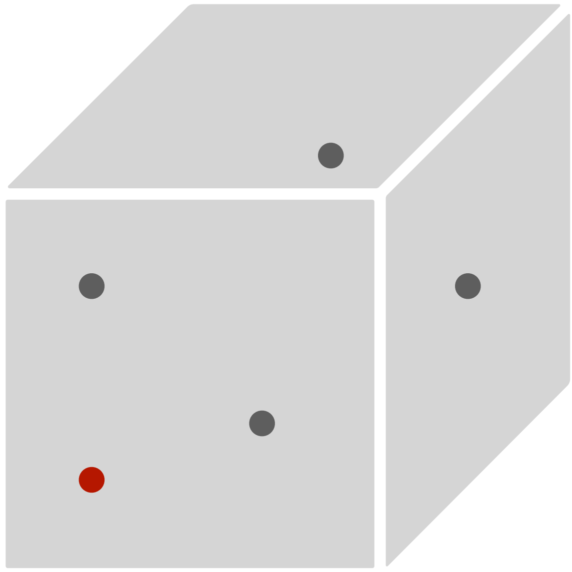
The question 👁:
Why ML can be more powerful than non-ML algorithms?

The answer ✪:
Generalizing from data can be easier than computing everything









Predicting ground states: Theorem

$x_\ell \in [-1,1]^m$

Training data: $\{x_\ell \to \sigma_T(\rho(x_\ell))\}_{\ell=1}^N$

- What can be prove about classical ML algorithm?

Classical algorithm ⛔

2D
spectral gap 1
1-body observable
average prediction error 1/4

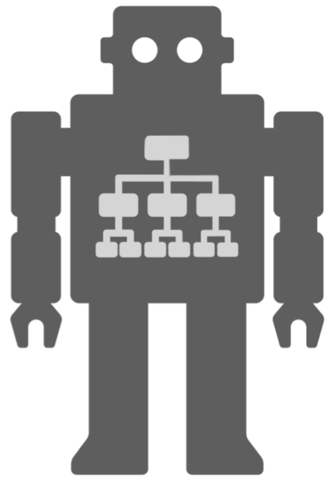
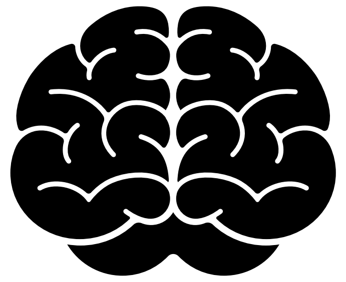
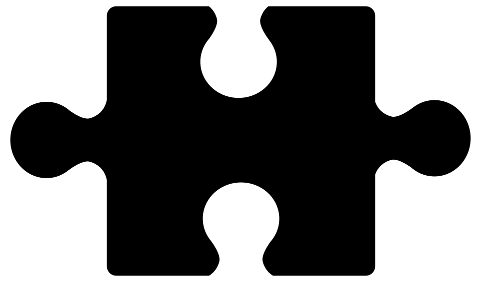
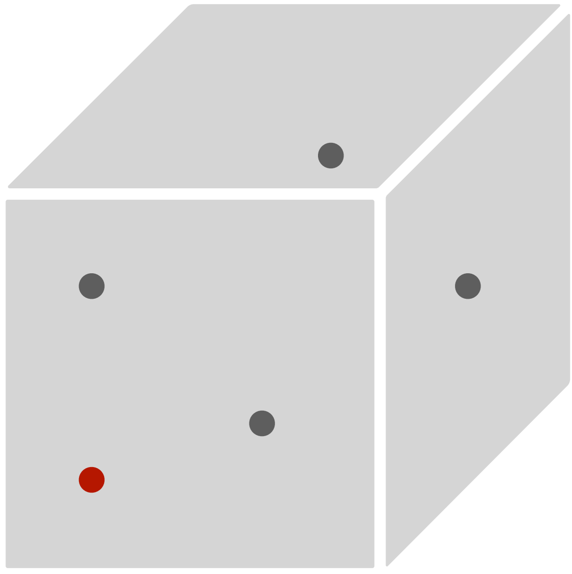
Classical ML (trained with data) ✓

any constant dimension
any constant spectral gap
any few-body observable
any average prediction error $\epsilon = \mathcal{O}(1)$

The question 👁:
Why ML can be more powerful than
non-ML algorithms?

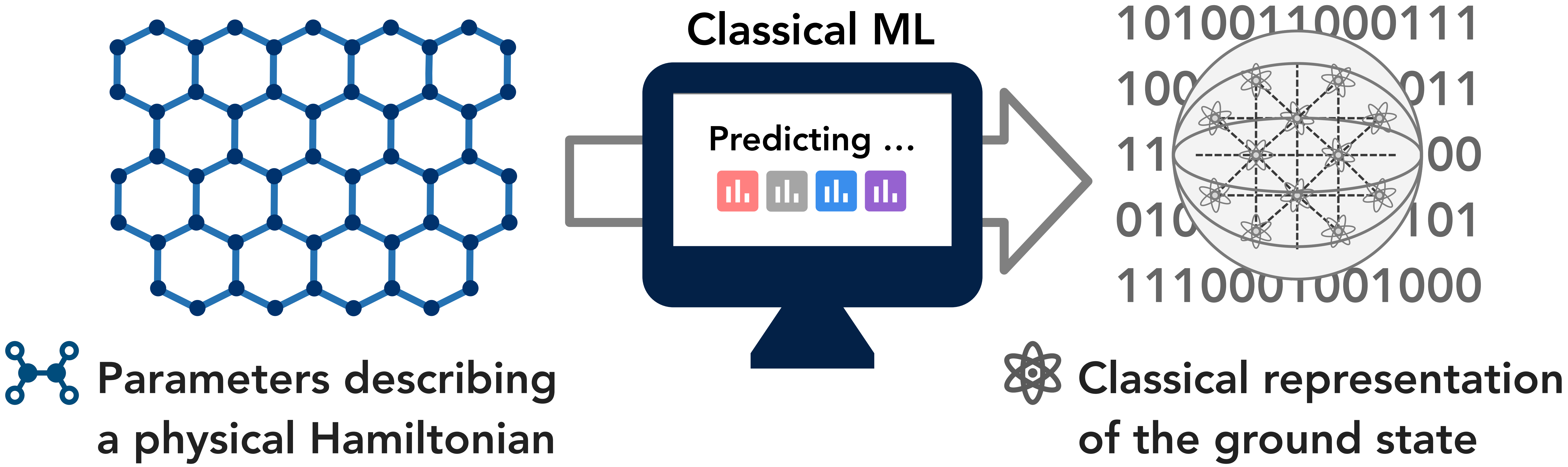
The answer ✪:
Generalizing from data can be
easier than computing everything

Data contain computational power
that enhance those that learn

Predicting ground states: Numerics

- How well does classical ML algorithm perform in actual physical systems?



Classical ML

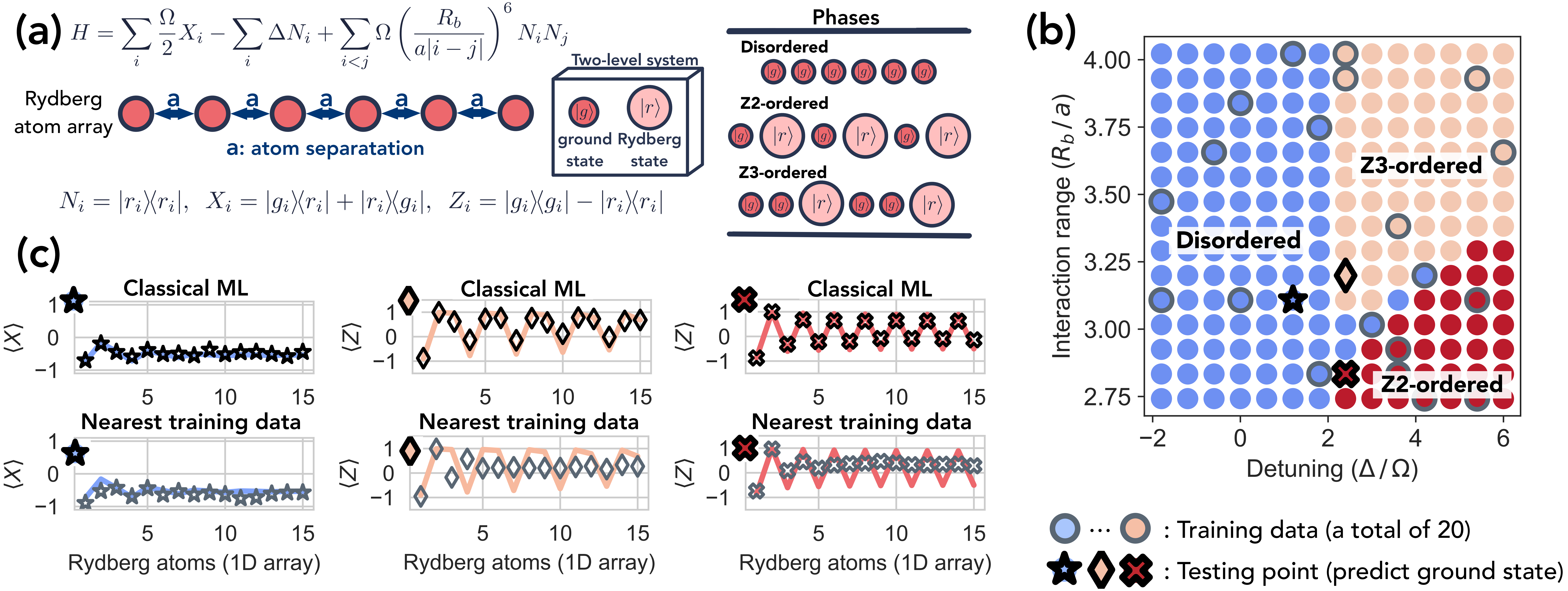
Predicting ...

Parameters describing
a physical Hamiltonian

Classical representation
of the ground state

1D Rydberg atom array

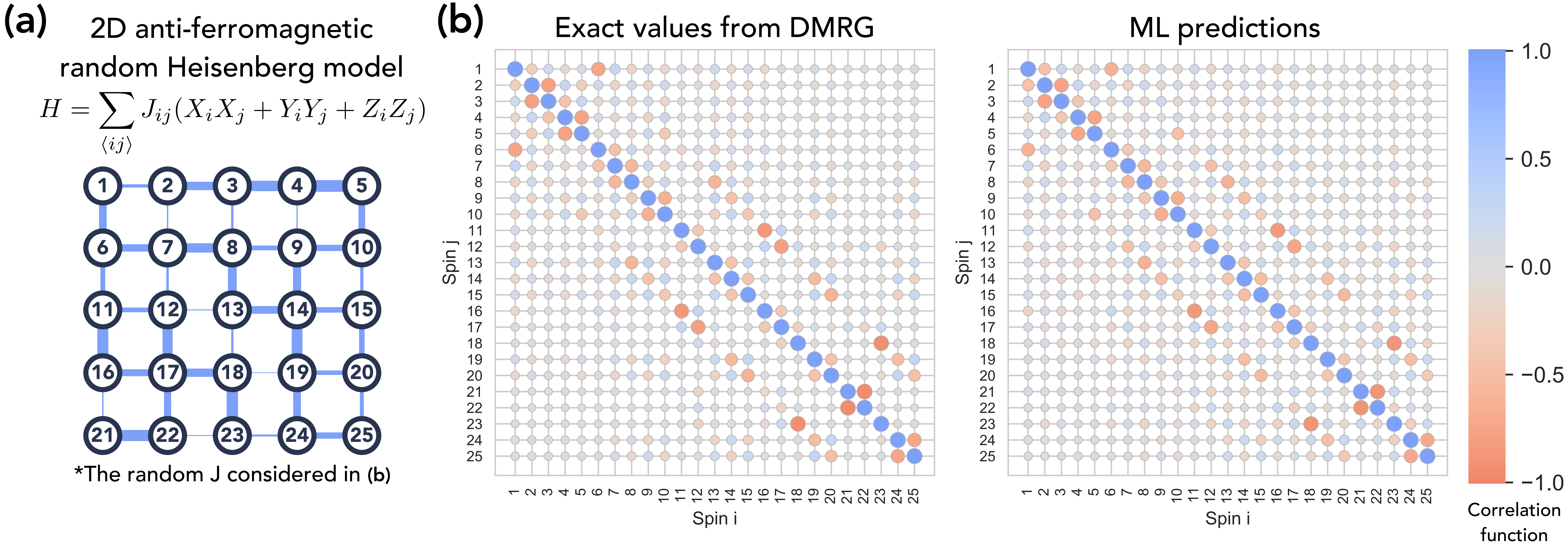
We consider training data size $N = 20$, $T = 500$ randomized measurements for constructing classical shadows.
The best ML model is chosen from Gaussian kernel method, infinite-width neural networks, and l_2-Dirichlet kernel.



*Solid lines in the six line plots indicate exact values from DMRG

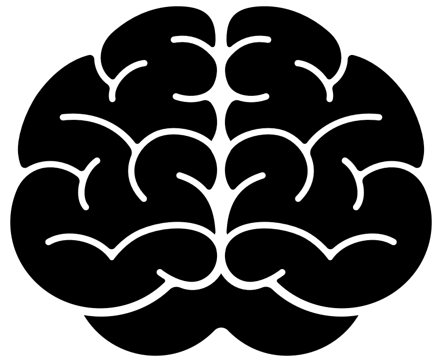
2D random Heisenberg model

We consider training data size $N = 100$, $T = 500$ randomized measurements for constructing classical shadows.
The best ML model is chosen from Gaussian kernel method, infinite-width neural networks, and l_2-Dirichlet kernel.



Outline

- Review on classical shadow formalism
- Training machines to predict ground states (theory+experiments)
- Training machines to classify quantum phases of matter (theory+experiments)

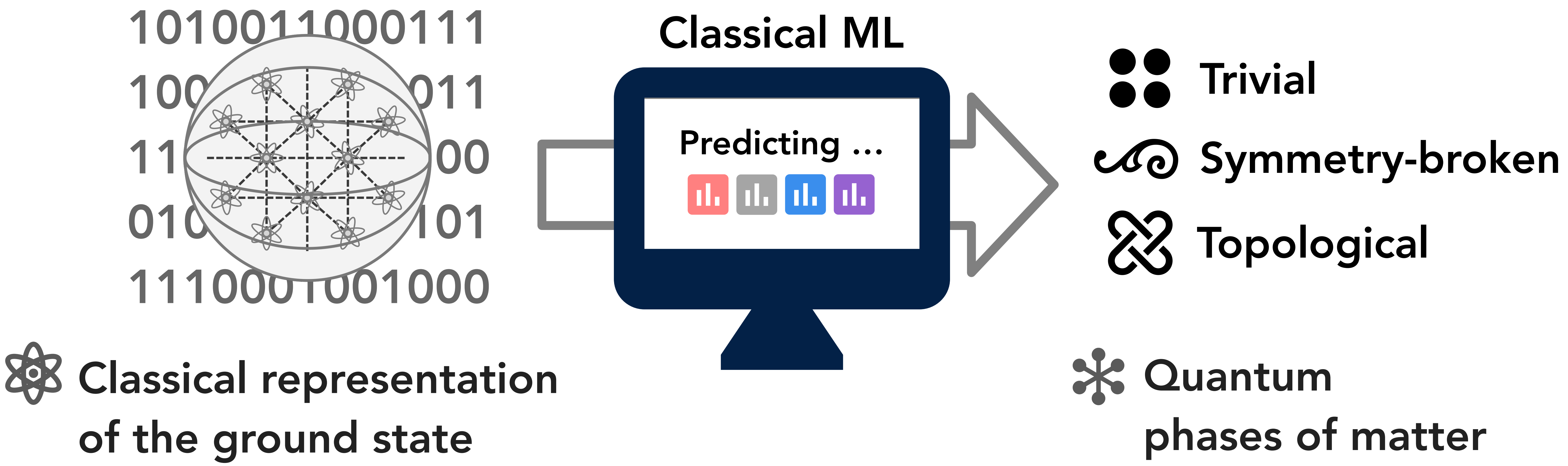


Outline

- Review on classical shadow formalism
- Training machines to predict ground states (theory+experiments)
- Training machines to classify quantum phases of matter (theory+experiments)

Classifying quantum phases: Task

- Given a quantum state ρ, predict which quantum phases of matter the state ρ is in.
- We represent the quantum state ρ using classical shadow, which is a 2D array of measurement outcomes $S_T(\rho) = \{\sigma_i^{(t)}\}_{i=1\sim n, t=1\sim T}$ with $\sigma_T(\rho) = \frac{1}{T}\sum_{t=1}^{T} \sigma_1^{(t)} \otimes \ldots \otimes \sigma_n^{(t)} \approx \rho$.



Classical representation of the ground state

Classical ML

Predicting ...

Trivial

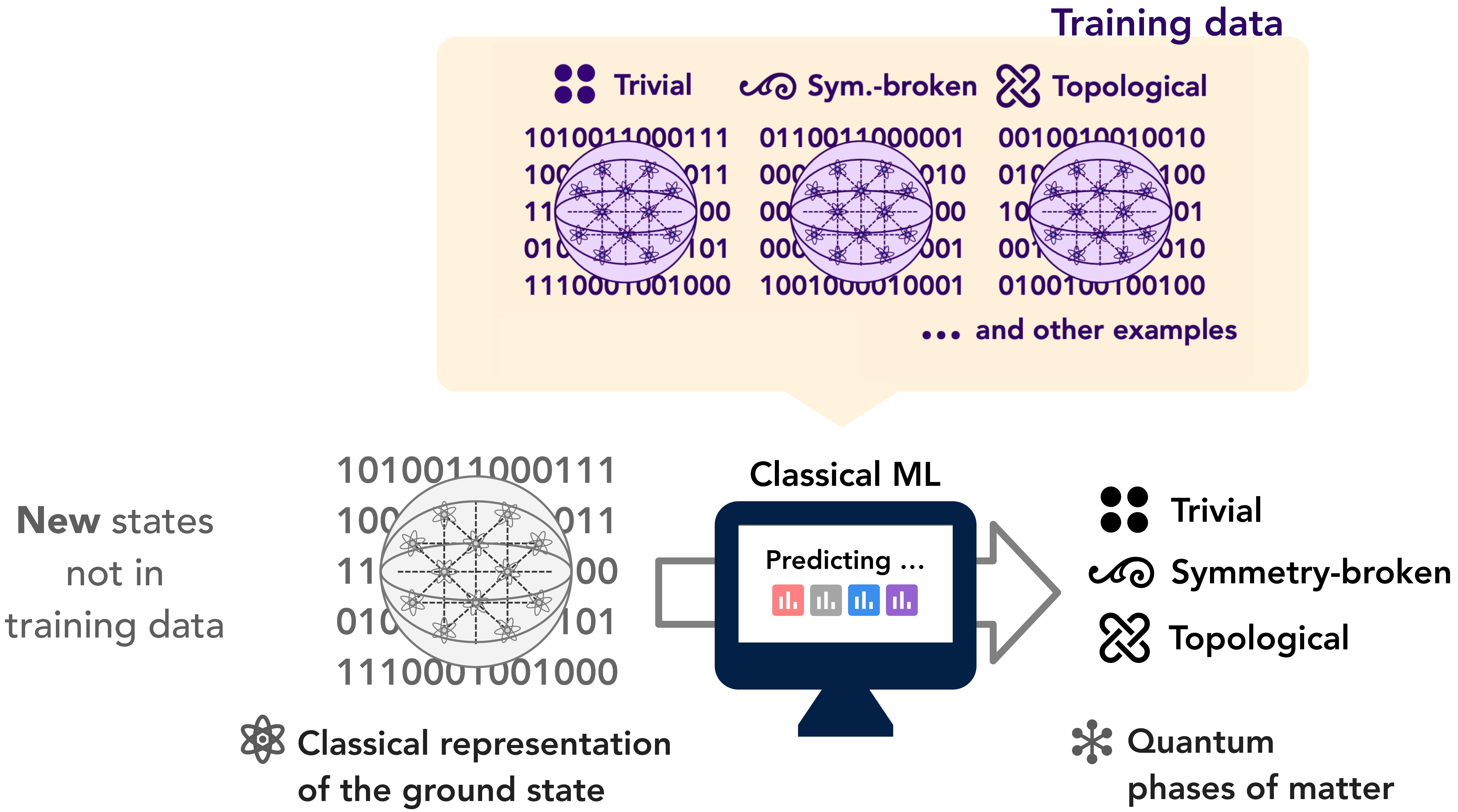
Symmetry-broken

Topological

Quantum phases of matter

Classifying quantum phases: Task

- Given a quantum state ρ, classify which quantum phases of matter the state ρ is in.
- Training data: examples of states and associated phase.

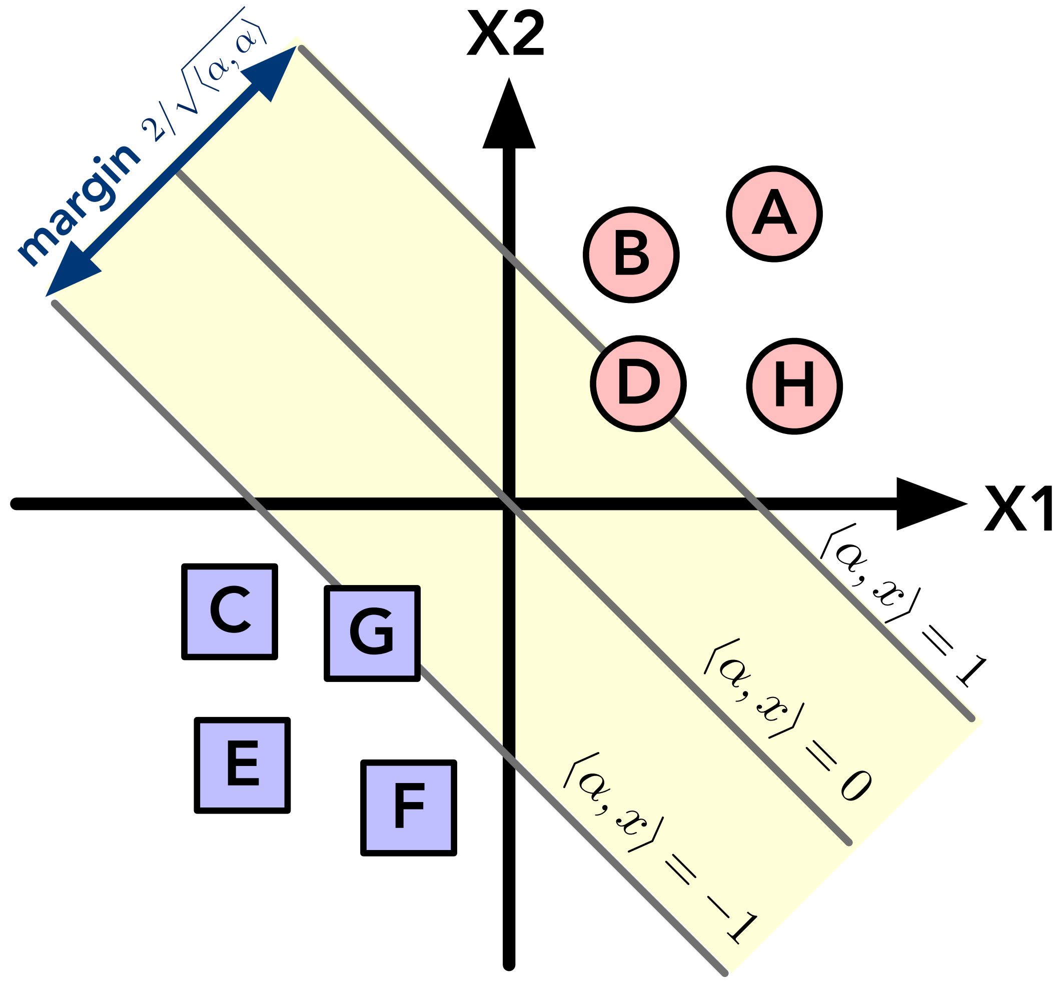


Classifying quantum phases: ML

- The ML model tries to find a classifying function that separates the phases of matter well.
- For symmetry-broken phases, there is typically a local observable O with

$$\mathrm{Tr}(O\rho_A) > 0, \forall \rho_A \in \text{phase A}, \qquad \mathrm{Tr}(O\rho_B) \leq 0, \forall \rho_B \in \text{phase B}.$$

- Proposition 2 shows that no observable (i.e., linear function) can classify topological phases.



Proposition 2

Consider two distinct topological phases A and B.

No (local/global) observable O exists such that

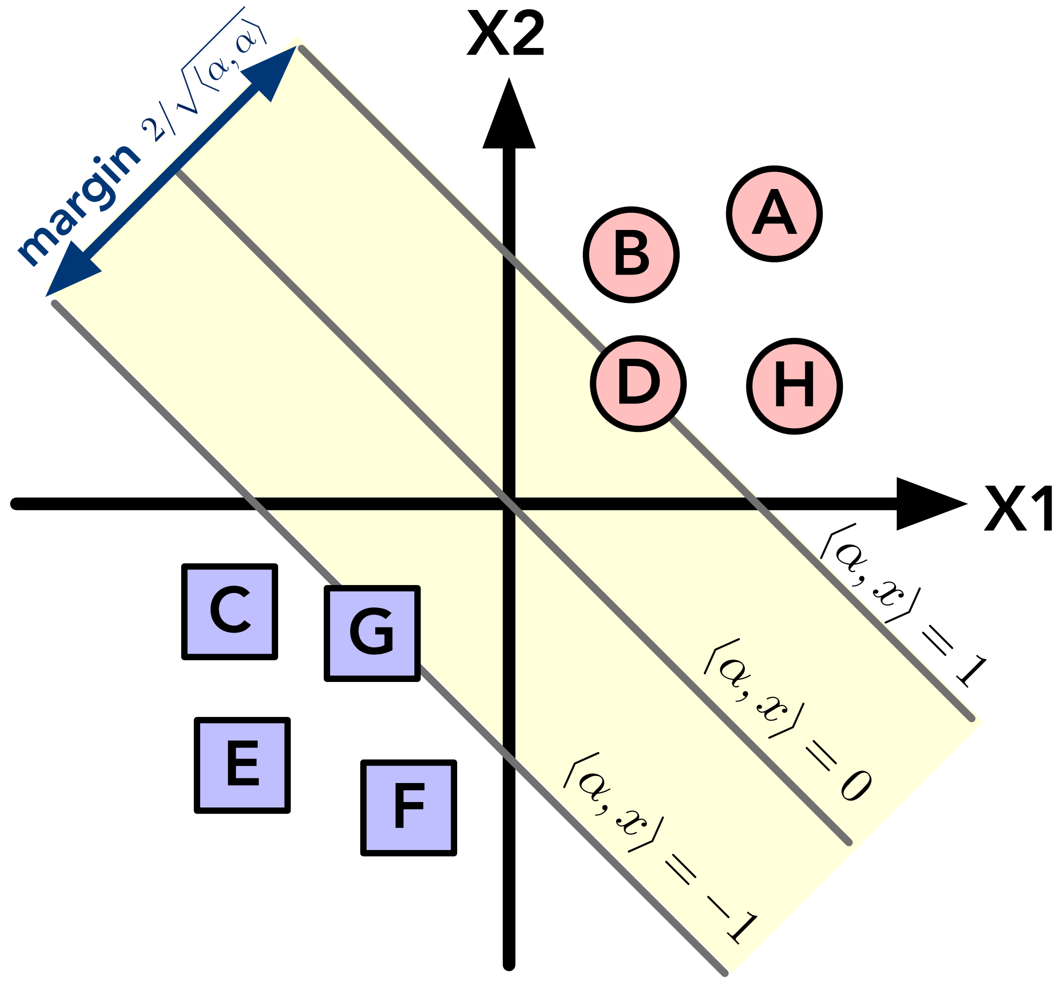
$$\mathrm{Tr}(O\rho_A) > 0, \forall \rho_A \in \text{phase A}, \qquad \mathrm{Tr}(O\rho_B) \leq 0, \forall \rho_B \in \text{phase B}.$$

Classifying quantum phases: ML

- The ML model tries to find a classifying function that separates the phases of matter well.
- For symmetry-broken phases, there is typically a local observable O with

$$\mathrm{Tr}(O\rho_A) > 0, \forall \rho_A \in \text{phase A}, \qquad \mathrm{Tr}(O\rho_B) \leq 0, \forall \rho_B \in \text{phase B}.$$

- Proposition 2 shows that no observable (i.e., linear function) can classify topological phases.



Two states are in the same topological phase if they are connected by const. depth circuit

Proposition 2

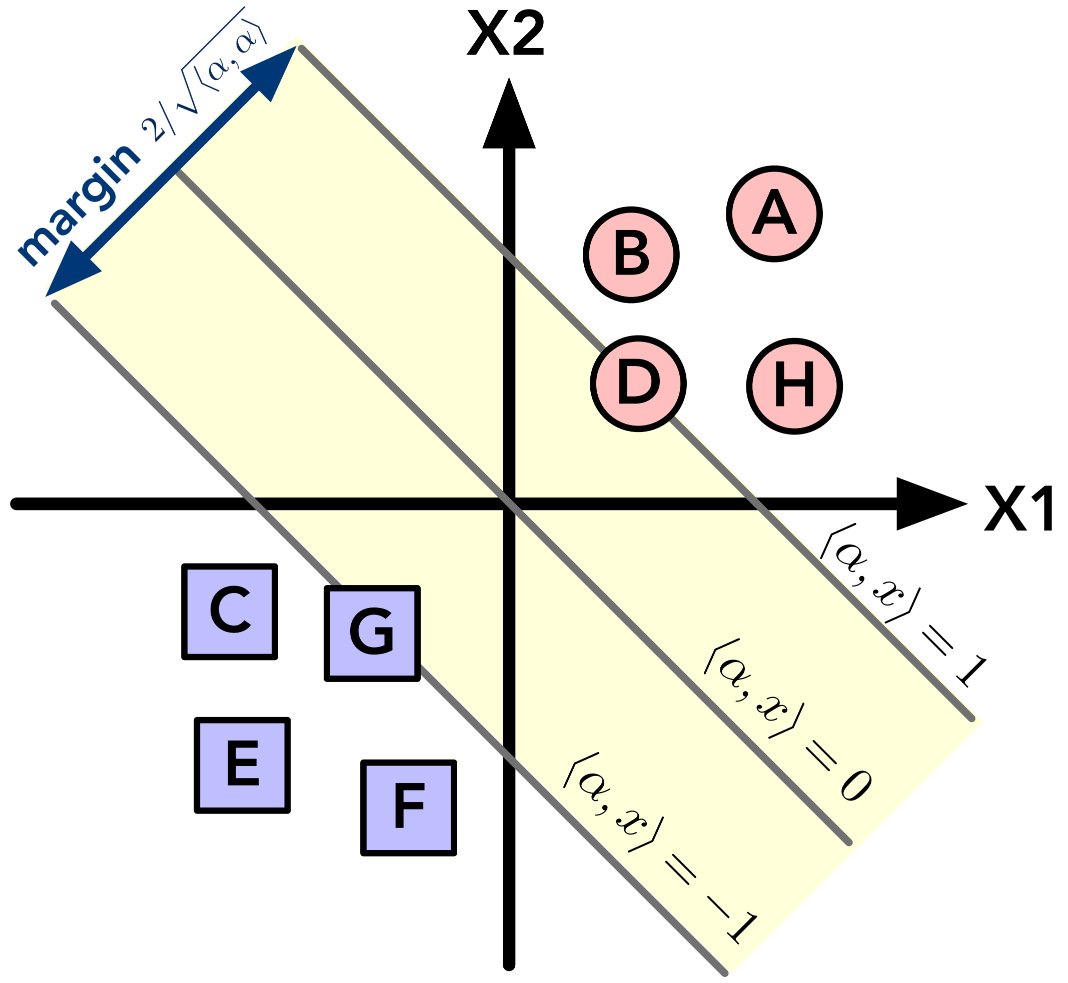
Consider two distinct topological phases A and B.

No (local/global) observable O exists such that

$$\mathrm{Tr}(O\rho_A) > 0, \forall \rho_A \in \text{phase A}, \qquad \mathrm{Tr}(O\rho_B) \leq 0, \forall \rho_B \in \text{phase B}.$$

Classifying quantum phases: ML

- We need a more powerful ML model that can learn nonlinear functions, such as $\mathrm{Tr}(O\rho \otimes \rho)$, $\mathrm{Tr}(O\rho^{\otimes d})$, or a general analytic function $f(\rho)$.



Proposition 2

Consider two distinct topological phases A and B.

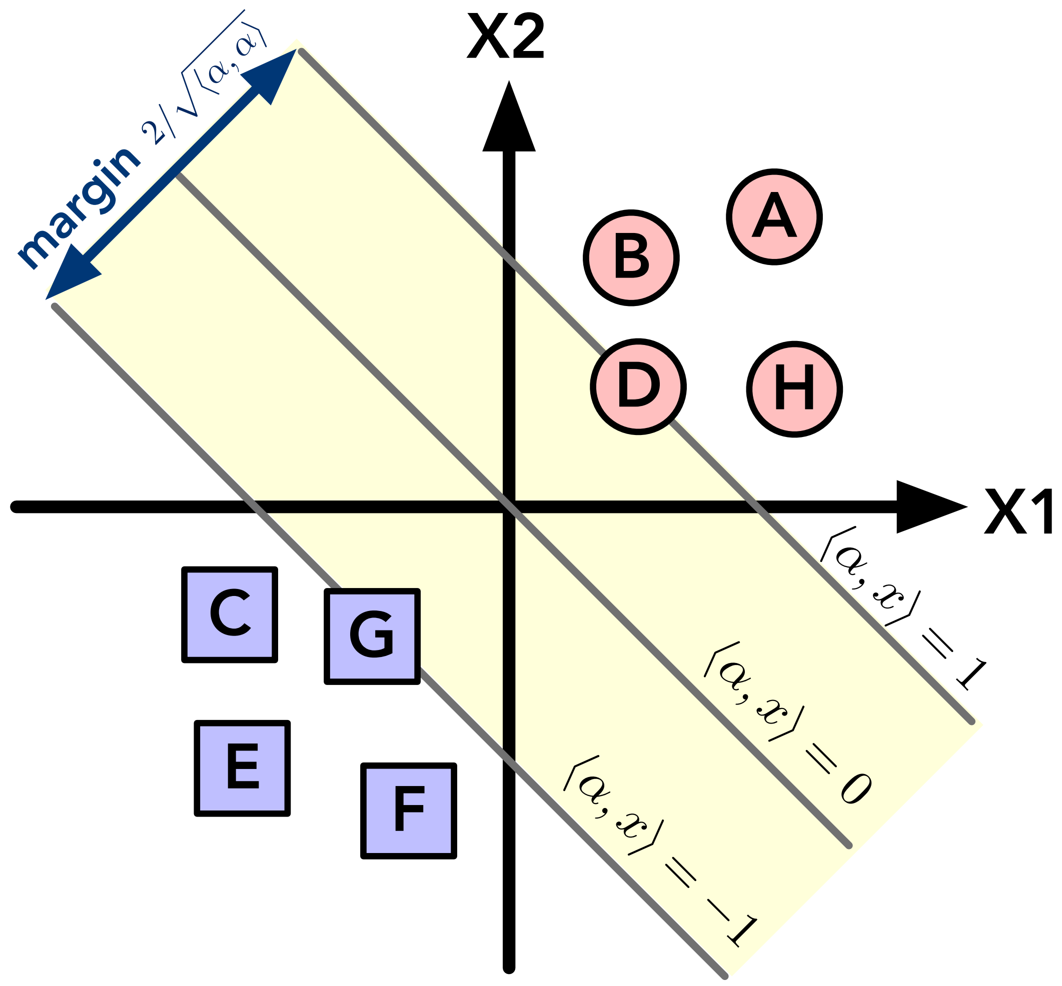
No (local/global) observable O exists such that

$\mathrm{Tr}(O\rho_A) > 0, \forall \rho_A \in$ phase A, $\quad \mathrm{Tr}(O\rho_B) \leq 0, \forall \rho_B \in$ phase B.

Classifying quantum phases: ML

- We need a more powerful ML model that can learn nonlinear functions, such as $\mathrm{Tr}(O\rho \otimes \rho)$, $\mathrm{Tr}(O\rho^{\otimes d})$, or a general analytic function $f(\rho)$.
- To do so, we consider learning a linear function in an ∞-dim space, where each state ρ is mapped to

$$\phi^{(\mathrm{shadow})}(S_T(\rho)) = \lim_{D,R\to\infty} \bigoplus_{d=0}^{D} \sqrt{\frac{\tau^d}{d!}} \left(\bigoplus_{r=0}^{R} \sqrt{\frac{1}{r!}\left(\frac{\gamma}{n}\right)^r} \bigoplus_{i_1=1}^{n} \cdots \bigoplus_{i_r=1}^{n} \mathrm{vec}\left[\frac{1}{T}\sum_{t=1}^{T} \bigotimes_{\ell=1}^{r} \sigma_{i_\ell}^{(t)}\right]\right)^{\otimes d}.$$



Proposition 2

Consider two distinct topological phases A and B.

No (local/global) observable O exists such that

$\mathrm{Tr}(O\rho_A) > 0, \forall \rho_A \in$ phase A, $\quad \mathrm{Tr}(O\rho_B) \leq 0, \forall \rho_B \in$ phase B.

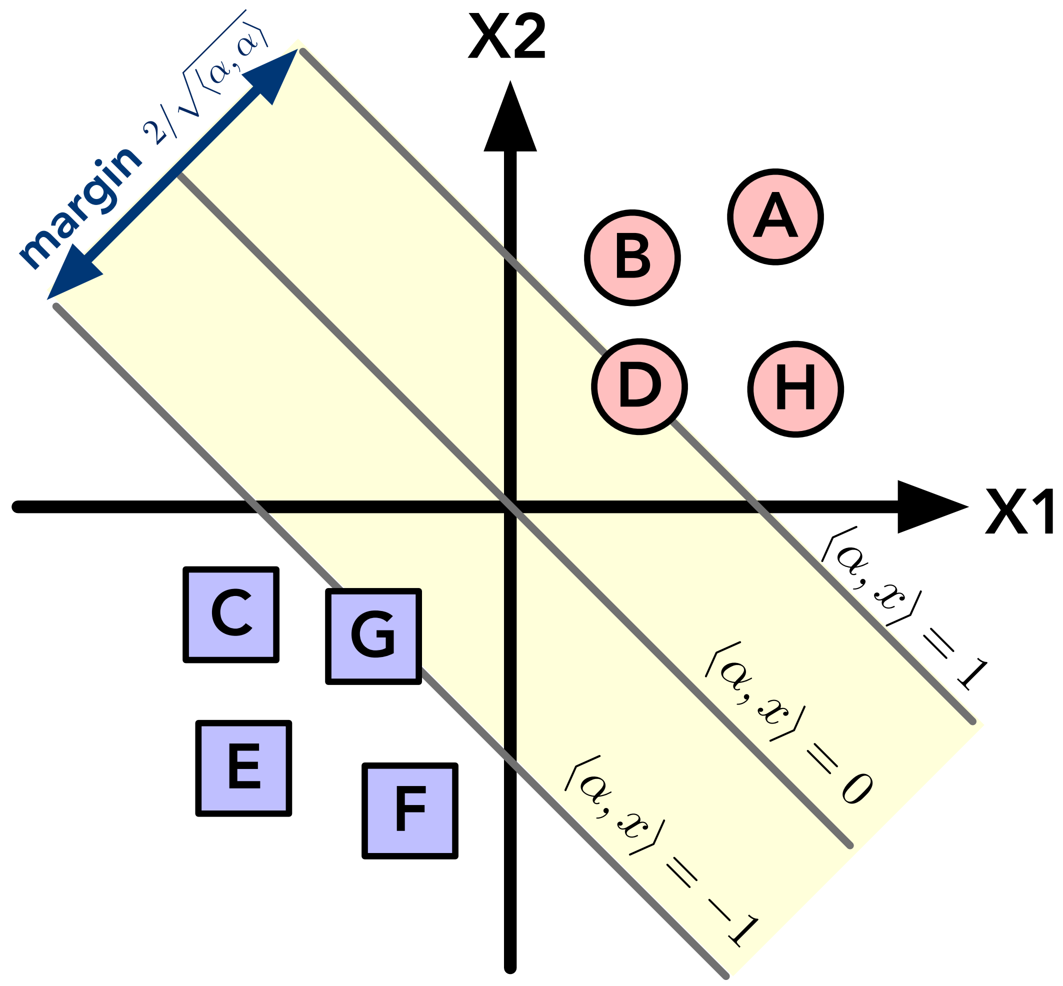
Classifying quantum phases: ML

- We need a more powerful ML model that can learn nonlinear functions, such as $\mathrm{Tr}(O\rho \otimes \rho)$, $\mathrm{Tr}(O\rho^{\otimes d})$, or a general analytic function $f(\rho)$.
- To do so, we consider learning a linear function i

Classical shadow formalism

Approximate reduced density matrix for subsystem $i_1, \ldots, i_r$

$$\phi^{(\mathrm{shadow})}(S_T(\rho)) = \lim_{D,R\to\infty} \bigoplus_{d=0}^{D} \sqrt{\frac{\tau^d}{d!}} \left(\bigoplus_{r=0}^{R} \sqrt{\frac{1}{r!}\left(\frac{\gamma}{n}\right)^r} \bigoplus_{i_1=1}^{n} \cdots \bigoplus_{i_r=1}^{n} \boxed{\mathrm{vec}\left[\frac{1}{T}\sum_{t=1}^{T}\bigotimes_{\ell=1}^{r}\sigma_{i_\ell}^{(t)}\right]} \right.^{\otimes d} .$$



Proposition 2

Consider two distinct topological phases A and B.

No (local/global) observable O exists such that

$\mathrm{Tr}(O\rho_A) > 0, \forall \rho_A \in$ phase A, $\quad \mathrm{Tr}(O\rho_B) \leq 0, \forall \rho_B \in$ phase B.

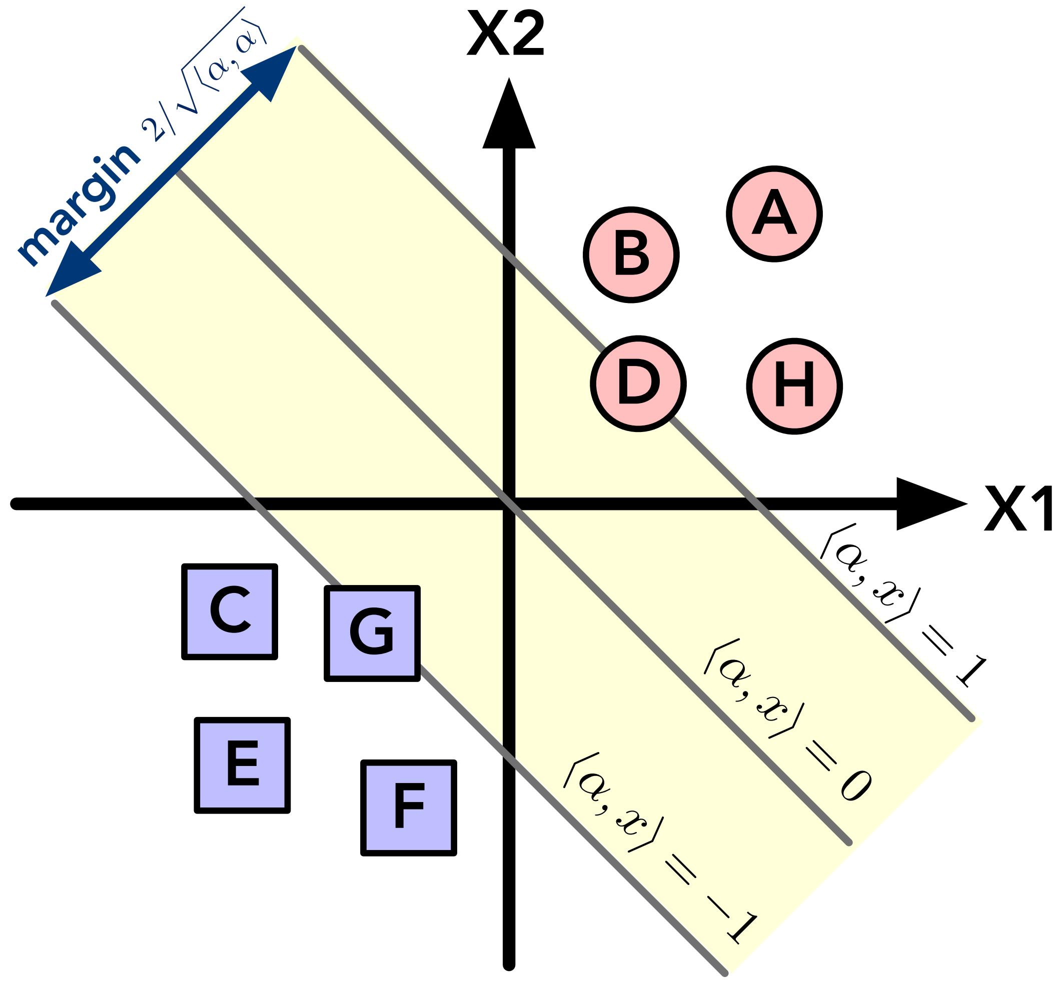
Classifying quantum phases: ML

- We need a more powerful ML model that can learn nonlinear functions, such as
  $\mathrm{Tr}(O\rho \otimes \rho)$, $\mathrm{Tr}(O\rho^{\otimes d})$, or a general analytic function $f(\rho)$.

- To do so, we consider learning a linear function in an ∞-dim ed to

Concatenate all r-body subsystems

$$\phi^{(\text{shadow})}(S_T(\rho)) = \lim_{D,R\to\infty} \bigoplus_{d=0}^{D} \sqrt{\frac{\tau^d}{d!}} \left(\bigoplus_{r=0}^{R} \sqrt{\frac{1}{r!}\left(\frac{\gamma}{n}\right)} \boxed{\bigoplus_{i_1=1}^{n} \cdots \bigoplus_{i_r=1}^{n} \mathrm{vec}\left[\frac{1}{T}\sum_{t=1}^{T}\bigotimes_{\ell=1}^{r}\sigma_{i_\ell}^{(t)}\right]} \right)^{\otimes d}.$$



Proposition 2

Consider two distinct topological phases A and B.

No (local/global) observable O exists such that

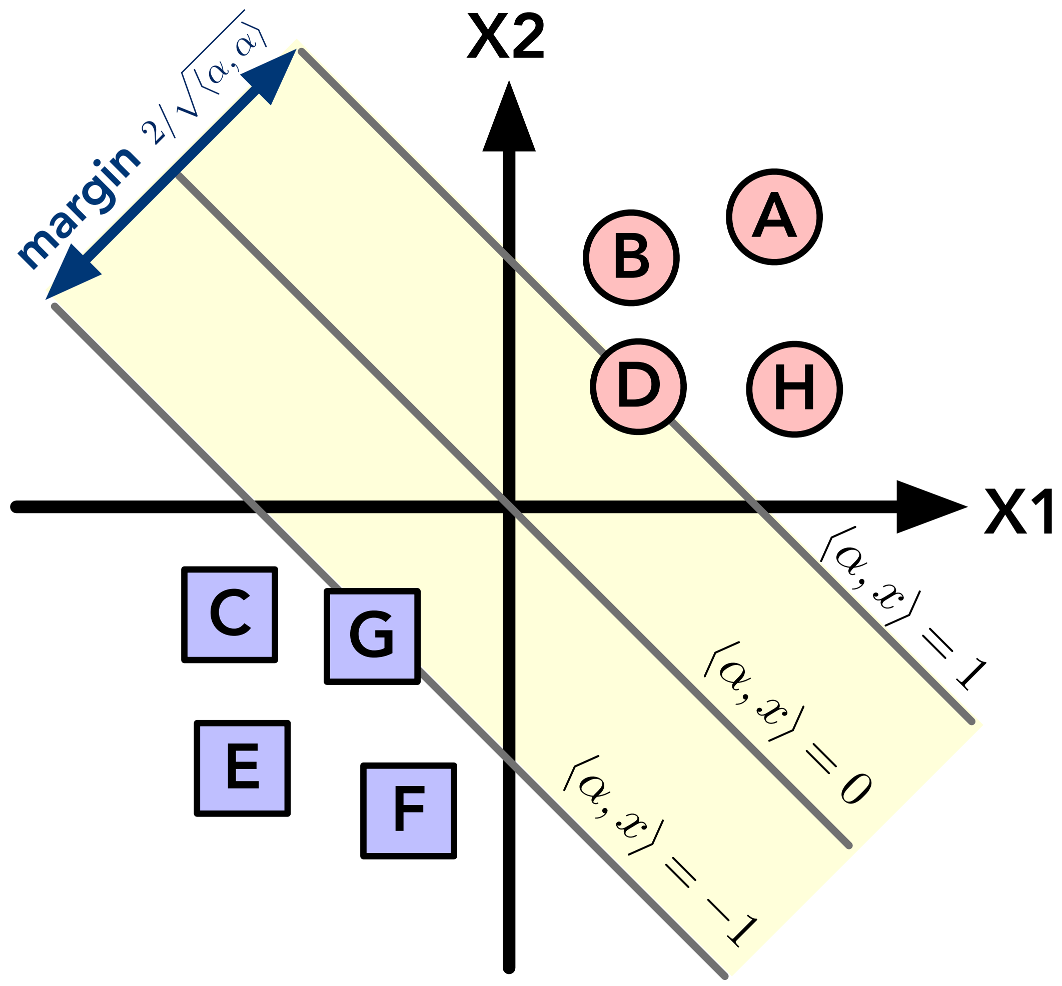
$\mathrm{Tr}(O\rho_A) > 0, \forall \rho_A \in$ phase A, $\mathrm{Tr}(O\rho_B) \leq 0, \forall \rho_B \in$ phase B.

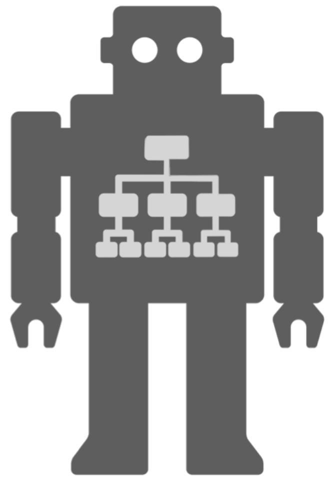
Classifying quantum phases: ML

- We need a more powerful ML model that can learn nonlinear functions, such as $\mathrm{Tr}(O\rho\otimes\rho)$, $\mathrm{Tr}(O\rho^{\otimes d})$, or a general analytic function $f(\rho)$.
- To do so, we consider learning a linear function in an s mapped to

Concatenate all sizes of subsystems

$$\phi^{(\text{shadow})}(S_T(\rho)) = \lim_{D,R\to\infty} \bigoplus_{d=0}^{D} \sqrt{\frac{\tau^d}{d!}} \boxed{\bigoplus_{r=0}^{R} \sqrt{\frac{1}{r!}\left(\frac{\gamma}{n}\right)^r} \bigoplus_{i_1=1}^{n}\cdots\bigoplus_{i_r=1}^{n} \mathrm{vec}\left[\frac{1}{T}\sum_{t=1}^{T}\bigotimes_{\ell=1}^{r}\sigma_{i_\ell}^{(t)}\right]}^{\otimes d}.$$





Proposition 2

Consider two distinct topological phases A and B.

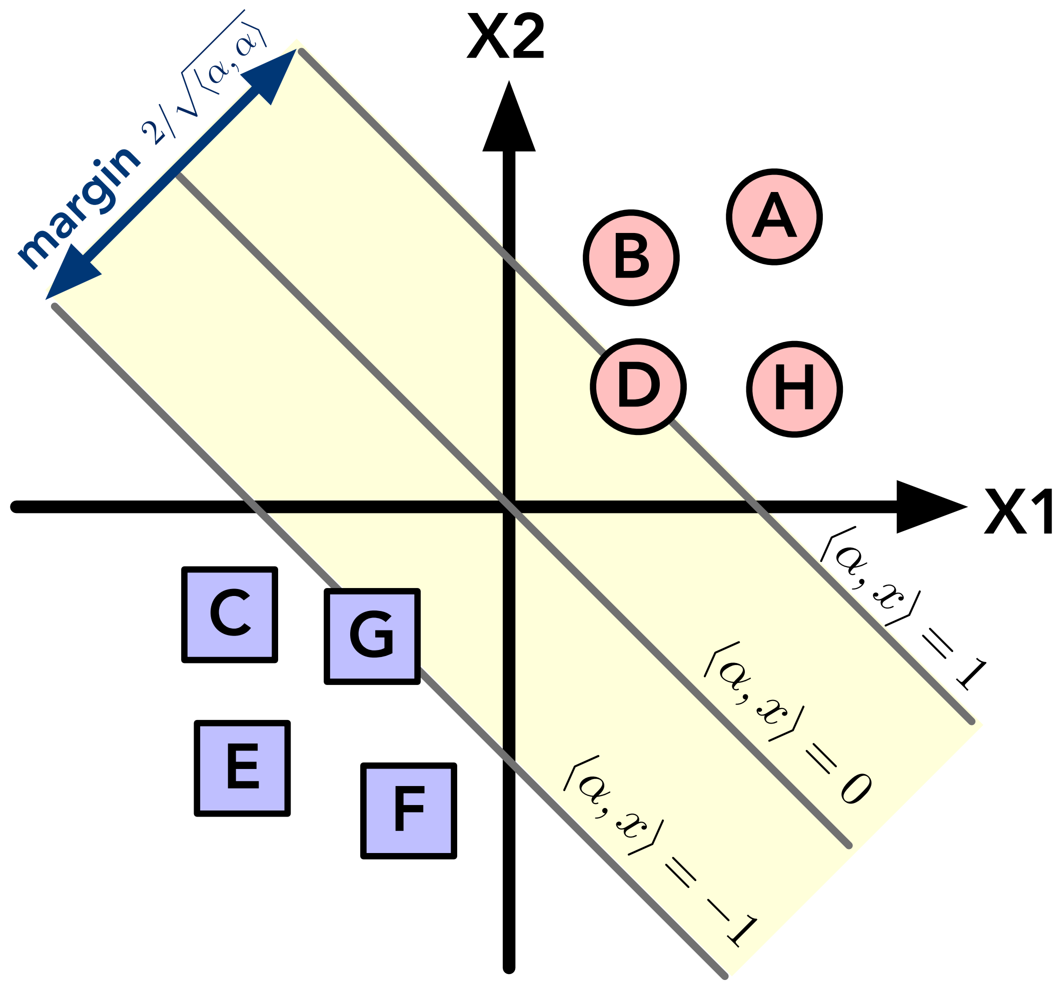
No (local/global) observable O exists such that

$\mathrm{Tr}(O\rho_A) > 0, \forall\rho_A \in$ phase A, $\mathrm{Tr}(O\rho_B) \leq 0, \forall\rho_B \in$ phase B.

Classifying quantum phases: ML

- We need a more powerful ML model that can learn nonlinear functions, such as $\mathrm{Tr}(O\rho \otimes \rho)$, $\mathrm{Tr}(O\rho^{\otimes d})$, or a general analytic function $f(\rho)$.
- To do so, we consider learning a linear function in an … mapped to

$$\phi^{(\text{shadow})}(S_T(\rho)) = \lim_{D,R\to\infty} \bigoplus_{d=0}^{D} \sqrt{\frac{\tau^d}{d!}} \left(\overset{\text{degree-}d\text{ polynomial expansion}}{\boxed{\bigoplus_{r=0}^{R} \sqrt{\frac{1}{r!}\left(\frac{\gamma}{n}\right)^r} \bigoplus_{i_1=1}^{n} \cdots \bigoplus_{i_r=1}^{n} \mathrm{vec}\left[\frac{1}{T}\sum_{t=1}^{T} \bigotimes_{\ell=1}^{r} \sigma_{i_\ell}^{(t)}\right]}} \right)^{\otimes d}.$$



Proposition 2

Consider two distinct topological phases A and B.

No (local/global) observable O exists such that

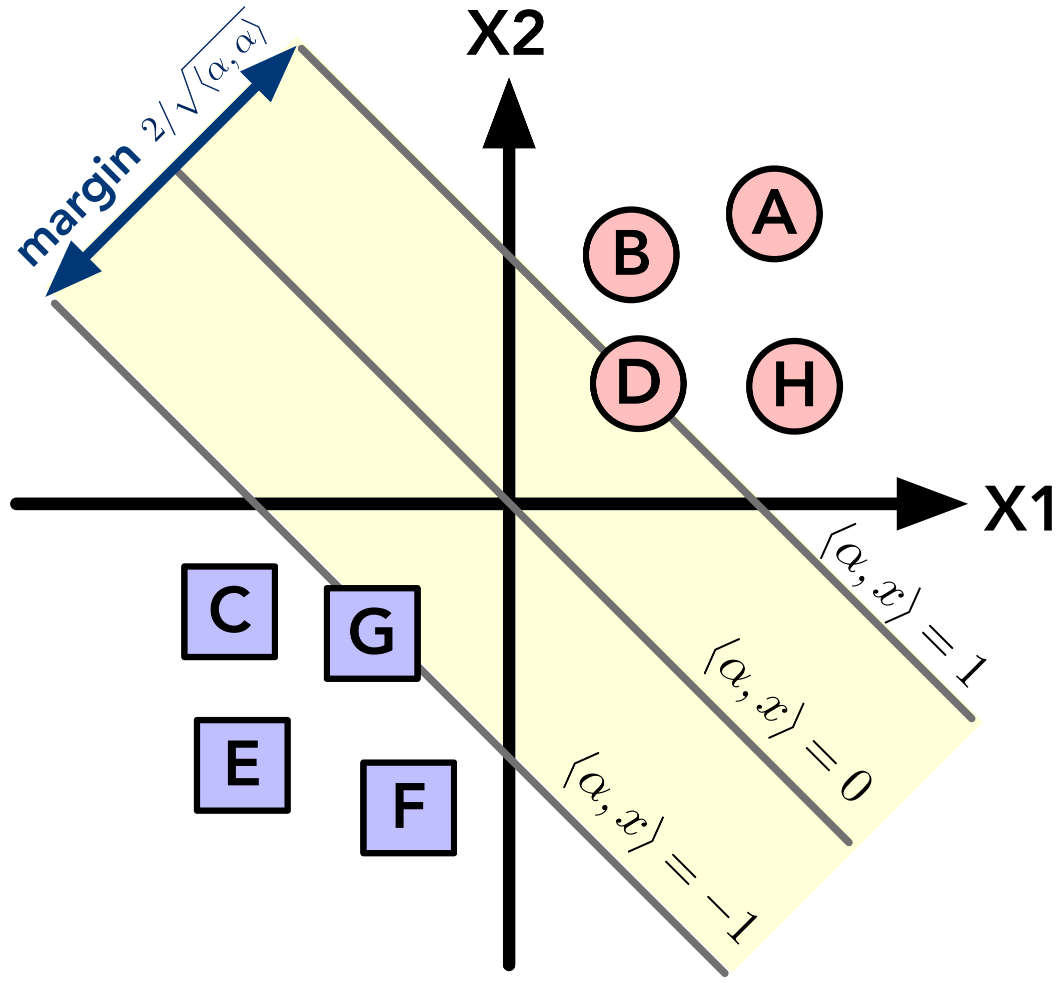
$\mathrm{Tr}(O\rho_A) > 0, \forall \rho_A \in$ phase A, $\mathrm{Tr}(O\rho_B) \leq 0, \forall \rho_B \in$ phase B.

Classifying quantum phases: ML

- We need a more powerful ML model that can learn nonlinear functions, such as $\mathrm{Tr}(O\rho \otimes \rho)$, $\mathrm{Tr}(O\rho^{\otimes d})$, or a general analytic function $f(\rho)$.
- To do so, we consider learning a linear function i e ρ is mapped to

Concatenate all degree

$$\phi^{(\mathrm{shadow})}(S_T(\rho)) = \lim_{D,R\to\infty} \boxed{\bigoplus_{d=0}^{D} \sqrt{\frac{\tau^d}{d!}} \left(\bigoplus_{r=0}^{R} \sqrt{\frac{1}{r!}\left(\frac{\gamma}{n}\right)^r} \bigoplus_{i_1=1}^{n} \dots \bigoplus_{i_r=1}^{n} \mathrm{vec}\left[\frac{1}{T}\sum_{t=1}^{T} \bigotimes_{\ell=1}^{r} \sigma_{i_\ell}^{(t)}\right]\right)^{\otimes d}}.$$



Proposition 2

Consider two distinct topological phases A and B.

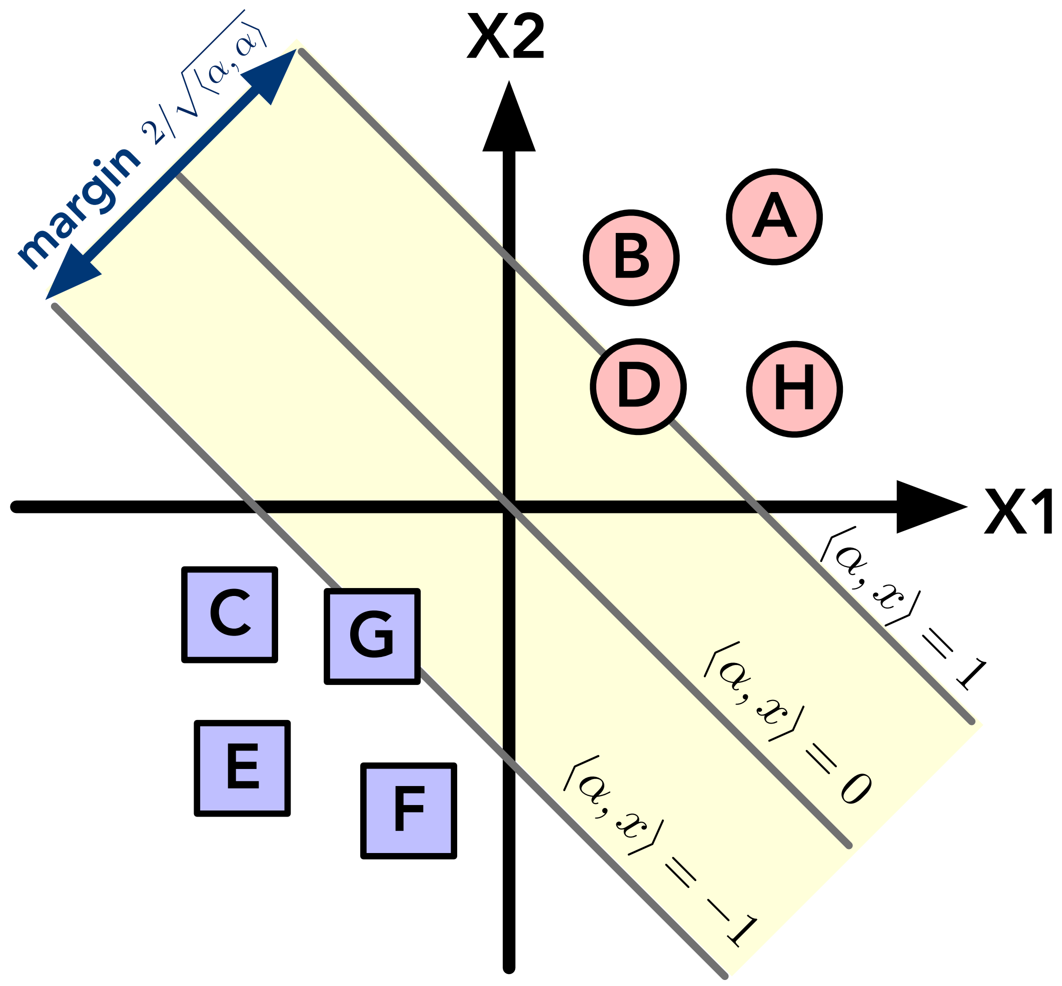
No (local/global) observable O exists such that

$\mathrm{Tr}(O\rho_A) > 0, \forall \rho_A \in$ phase A, $\quad \mathrm{Tr}(O\rho_B) \leq 0, \forall \rho_B \in$ phase B.

Classifying quantum phases: ML

- We need a more powerful ML model that can learn nonlinear functions, such as $\mathrm{Tr}(O\rho \otimes \rho)$, $\mathrm{Tr}(O\rho^{\otimes d})$, or a general analytic function $f(\rho)$.
- To do so, we consider learning a linear function in an ∞-dim space, where each state ρ is mapped to

$$\phi^{(\mathrm{shadow})}(S_T(\rho)) = \lim_{D,R\to\infty} \bigoplus_{d=0}^{D} \sqrt{\frac{\tau^d}{d!}} \left(\bigoplus_{r=0}^{R} \sqrt{\frac{1}{r!}\left(\frac{\gamma}{n}\right)^r} \bigoplus_{i_1=1}^{n} \cdots \bigoplus_{i_r=1}^{n} \mathrm{vec}\left[\frac{1}{T}\sum_{t=1}^{T}\bigotimes_{\ell=1}^{r}\sigma_{i_\ell}^{(t)}\right]\right)^{\otimes d}.$$

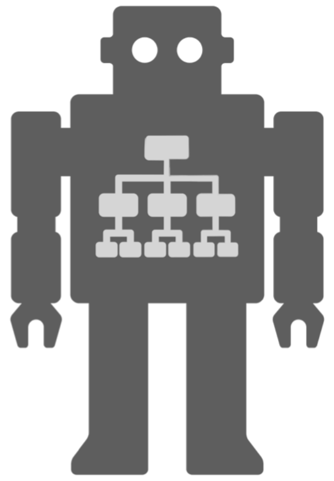


Proposition 2

Consider two distinct topological phases A and B.

No (local/global) observable O exists such that

$\mathrm{Tr}(O\rho_A) > 0, \forall \rho_A \in$ phase A, $\quad \mathrm{Tr}(O\rho_B) \leq 0, \forall \rho_B \in$ phase B.



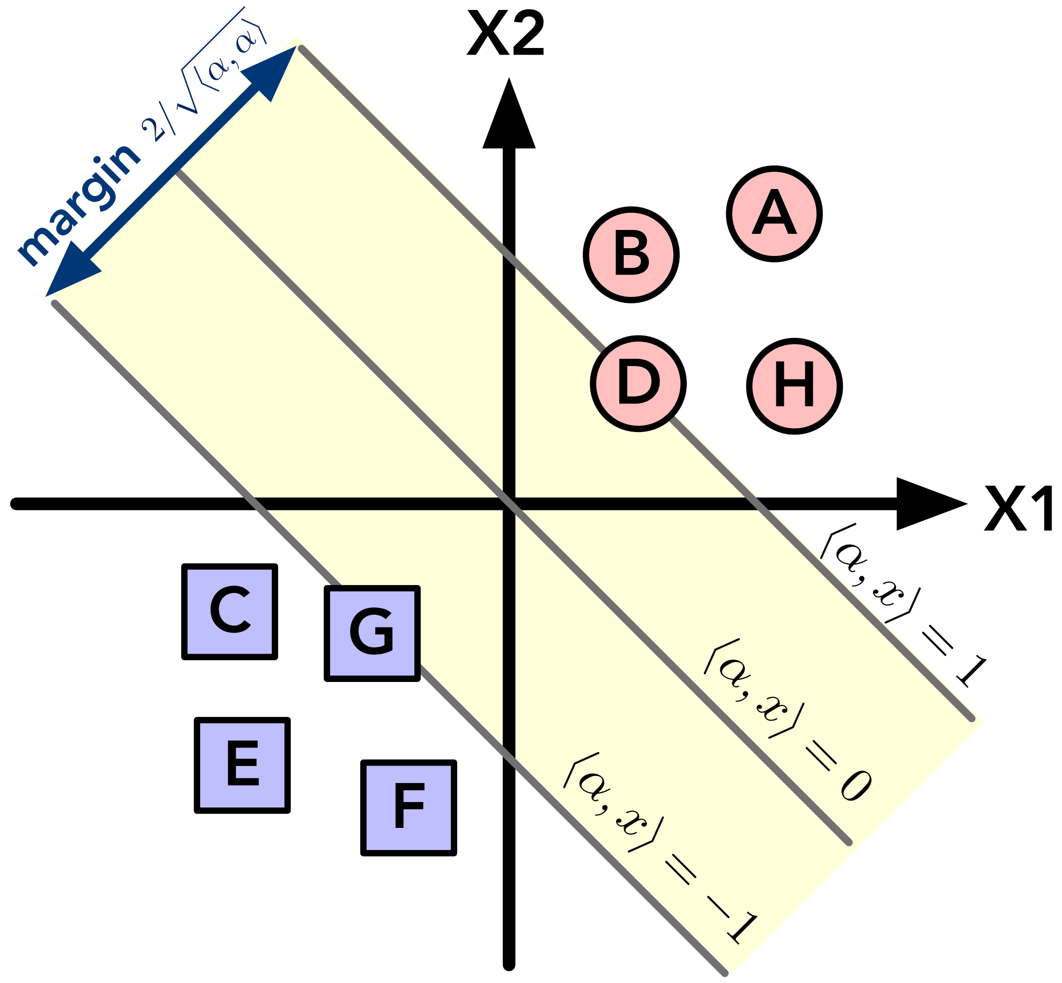
Classifying quantum phases: ML

- We need a more powerful ML model that can learn nonlinear functions, such as
  $\mathrm{Tr}(O\rho \otimes \rho)$, $\mathrm{Tr}(O\rho^{\otimes d})$, or a general analytic function $f(\rho)$.

- To do so, we consider learning a linear function in an ∞-dim space, where each state ρ is mapped to

$$\phi^{(\text{shadow})}(S_T(\rho)) = \lim_{D,R\to\infty} \bigoplus_{d=0}^{D} \sqrt{\frac{\tau^d}{d!}} \left(\bigoplus_{r=0}^{R} \sqrt{\frac{1}{r!}\left(\frac{\gamma}{n}\right)^r} \bigoplus_{i_1=1}^{n} \dots \bigoplus_{i_r=1}^{n} \mathrm{vec}\left[\frac{1}{T}\sum_{t=1}^{T}\bigotimes_{\ell=1}^{r}\sigma_{i_\ell}^{(t)}\right]\right)^{\otimes d}.$$

- It consists of arbitrarily-large r-body reduced density matrices and arbitrarily-high-degree expansion.



Proposition 2

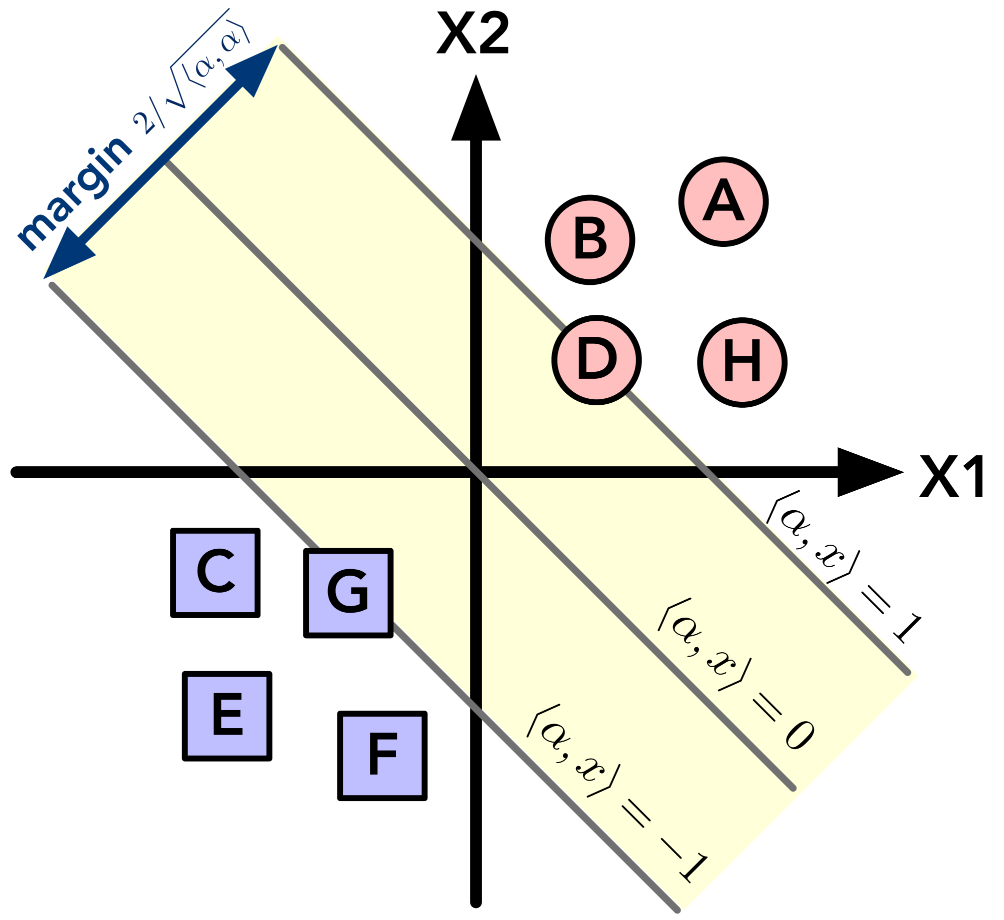
Consider two distinct topological phases A and B.

No (local/global) observable O exists such that

$\mathrm{Tr}(O\rho_A) > 0, \forall \rho_A \in$ phase A, $\quad \mathrm{Tr}(O\rho_B) \leq 0, \forall \rho_B \in$ phase B.

Classifying quantum phases: ML

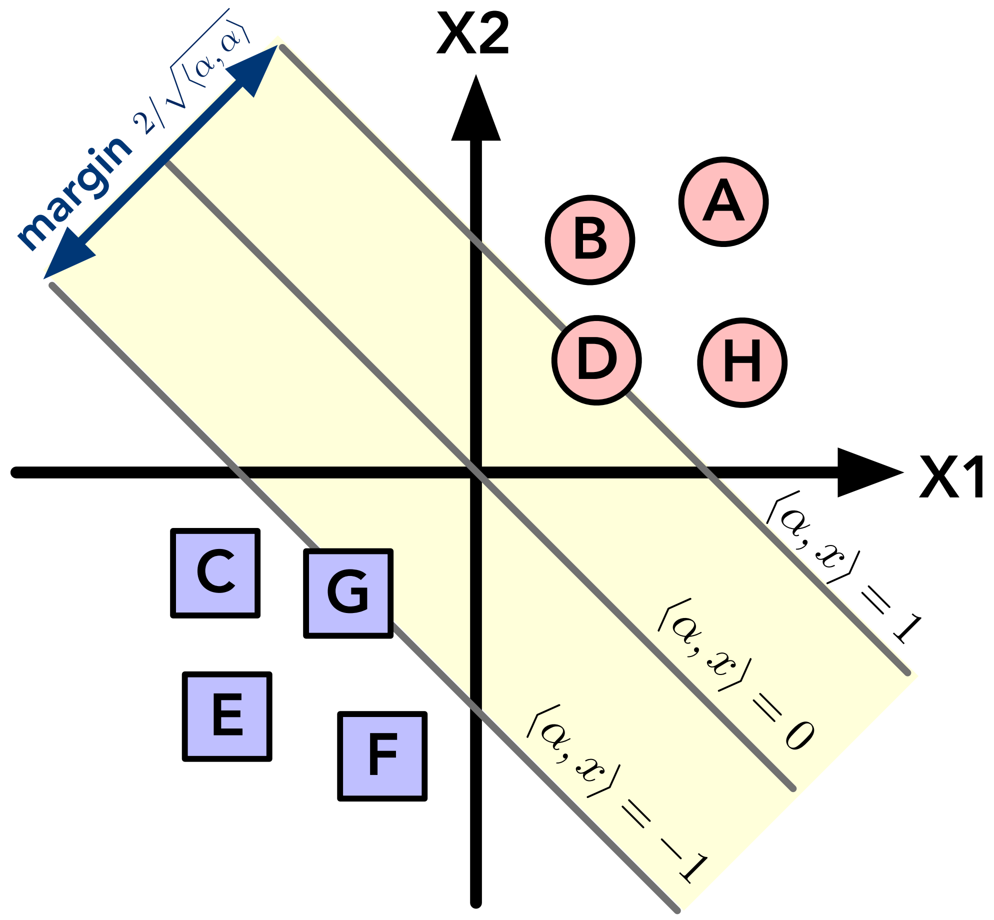
- Classical ML model learns a linear function in $\phi^{(\mathrm{shadow})}(S_T(\rho))$ (nonlinear in ρ).



Classifying quantum phases: ML

- Classical ML model learns a linear function in $\phi^{(\text{shadow})}(S_T(\rho))$ (nonlinear in ρ).
- Kernel trick shows we only need to efficiently compute the inner product (referred to as *shadow kernel*)

$$\langle \phi^{(\text{shadow})}(S_T(\rho)), \phi^{(\text{shadow})}(S_T(\tilde{\rho}))\rangle = \exp\left(\frac{\tau}{T^2}\sum_{t,t'=1}^{T}\exp\left(\frac{\gamma}{n}\sum_{i=1}^{n}\mathrm{Tr}\left(\sigma_i^{(t)}\tilde{\sigma}_i^{(t')}\right)\right)\right) \equiv k^{(\text{shadow})}\left(S_T(\rho), S_T(\tilde{\rho})\right).$$

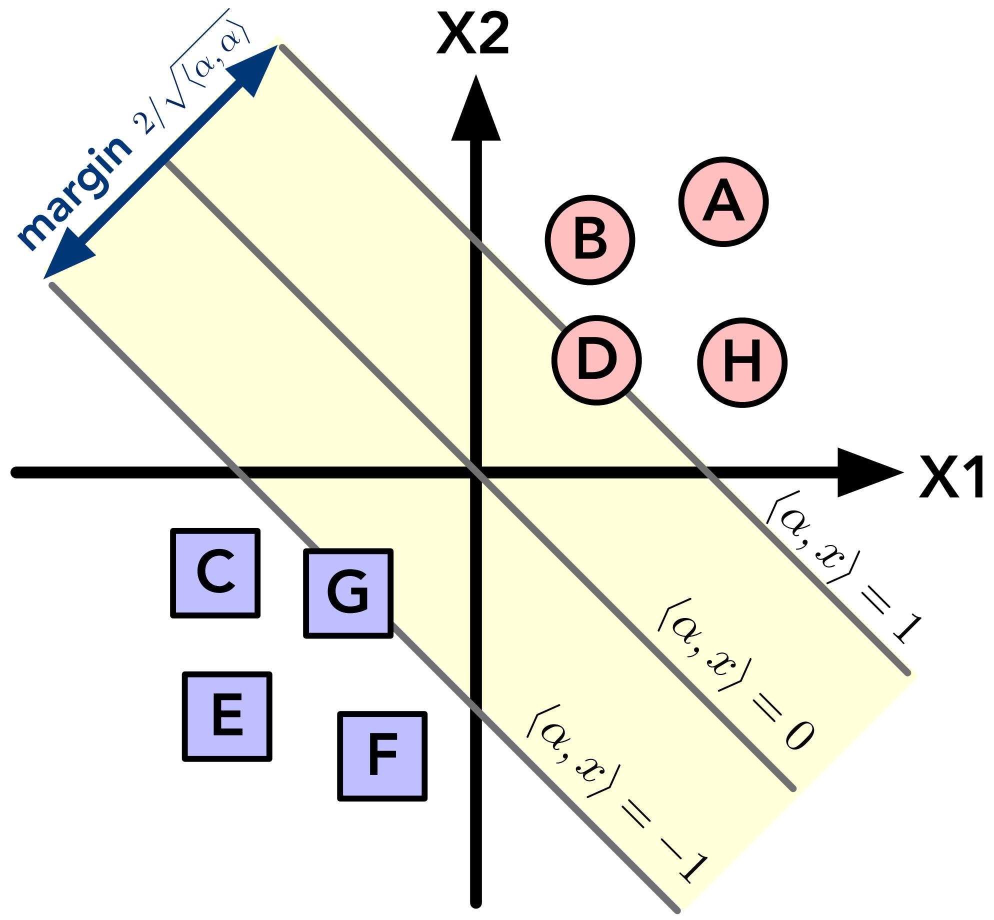


Classifying quantum phases: ML

- Classical ML model learns a linear function in $\phi^{(\text{shadow})}(S_T(\rho))$ (nonlinear in ρ).
- Kernel trick shows we only need to efficiently compute the inner product (referred to as *shadow kernel*)

$$\langle \phi^{(\text{shadow})}(S_T(\rho)), \phi^{(\text{shadow})}(S_T(\tilde{\rho}))\rangle = \exp\left(\frac{\tau}{T^2}\sum_{t,t'=1}^{T}\exp\left(\frac{\gamma}{n}\sum_{i=1}^{n}\mathrm{Tr}\left(\sigma_i^{(t)}\tilde{\sigma}_i^{(t')}\right)\right)\right) \equiv k^{(\text{shadow})}\left(S_T(\rho), S_T(\tilde{\rho})\right).$$

- Computing *shadow kernels* only takes classical computational time $\mathcal{O}(nT^2)$.

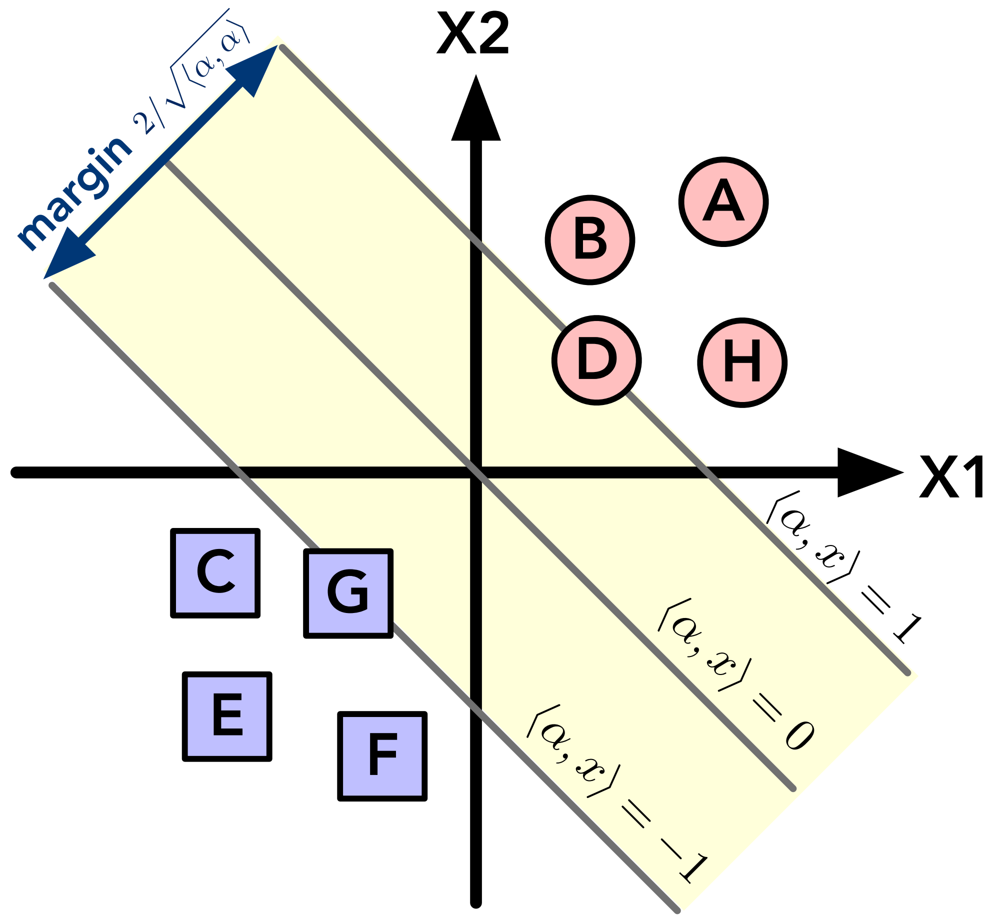


Classifying quantum phases: ML

- Classical ML model learns a linear function in $\phi^{(\text{shadow})}(S_T(\rho))$ (nonlinear in ρ).
- Kernel trick shows we only need to efficiently compute the inner product (referred to as *shadow kernel*)

$$\langle \phi^{(\text{shadow})}(S_T(\rho)), \phi^{(\text{shadow})}(S_T(\tilde{\rho}))\rangle = \exp\left(\frac{\tau}{T^2}\sum_{t,t'=1}^{T}\exp\left(\frac{\gamma}{n}\sum_{i=1}^{n}\mathrm{Tr}\left(\sigma_i^{(t)}\tilde{\sigma}_i^{(t')}\right)\right)\right) \equiv k^{(\text{shadow})}\left(S_T(\rho), S_T(\tilde{\rho})\right).$$

- Computing *shadow kernels* only takes classical computational time $\mathcal{O}(nT^2)$.
- Hence, training the classical ML model only take time polynomial in n, T, N (and is practically efficient).



Classifying quantum phases: Theorem

Theorem 2

If there is a **nonlinear** function of **few-body** reduced density matrices that classifies phases, then the classical algorithm can learn to classify these phases accurately.

Training data size and computation time scale polynomially in system size.

- The ML model constructs the classifying function explicitly.
- As long as the classifying function exists, the ML model with shadow kernel is guaranteed to find it.

Classifying quantum phases: Theorem

Theorem 2

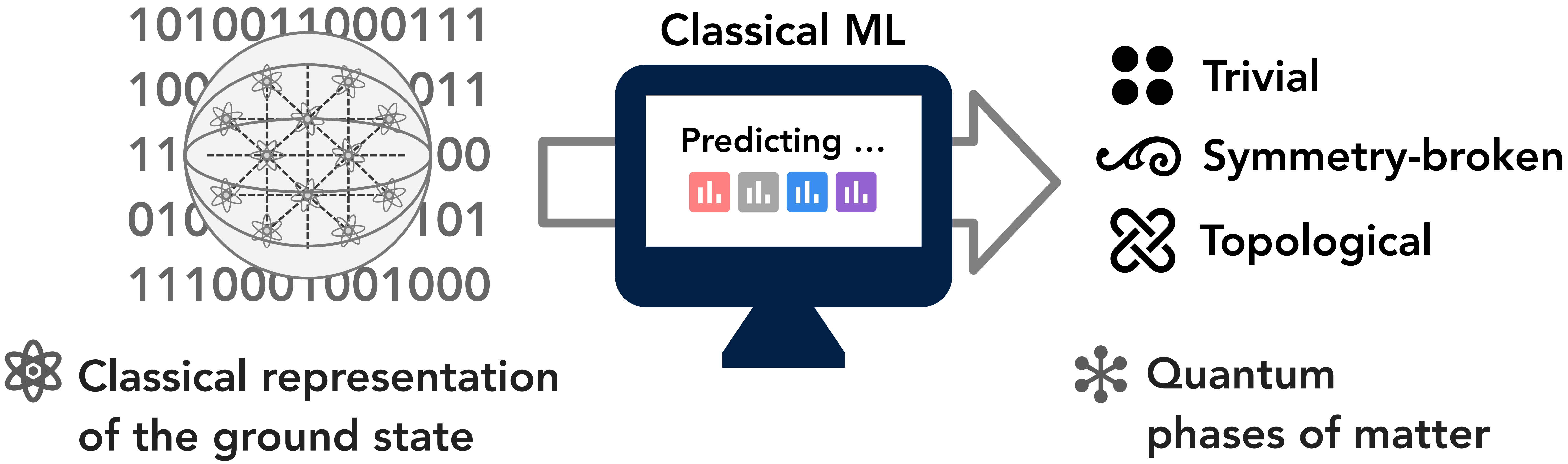
If there is a **nonlinear** function of **few-body** reduced density matrices that classifies phases, then the classical algorithm can learn to classify these phases accurately.

Training data size and computation time scale polynomially in system size.

- The ML model constructs the classifying function explicitly.
- As long as the classifying function exists, the ML model with shadow kernel is guaranteed to find it.
- Examples of classifying functions on few-body reduced density matrices (assuming const. spectral gap) include:
  1. Twist operators for 1D Haldane phase with $O(2)$-symmetry (linear function)
  2. Hall conductivity for systems adiabatically connected to free fermion (low-degree polynomial)
  3. Topological entanglement entropy in a constant region (nonlinear function)

Classifying quantum phases: Numerics

- How well does the classical ML algorithm perform in actual physical systems?



Classical representation of the ground state

Classical ML

Predicting ...

Trivial

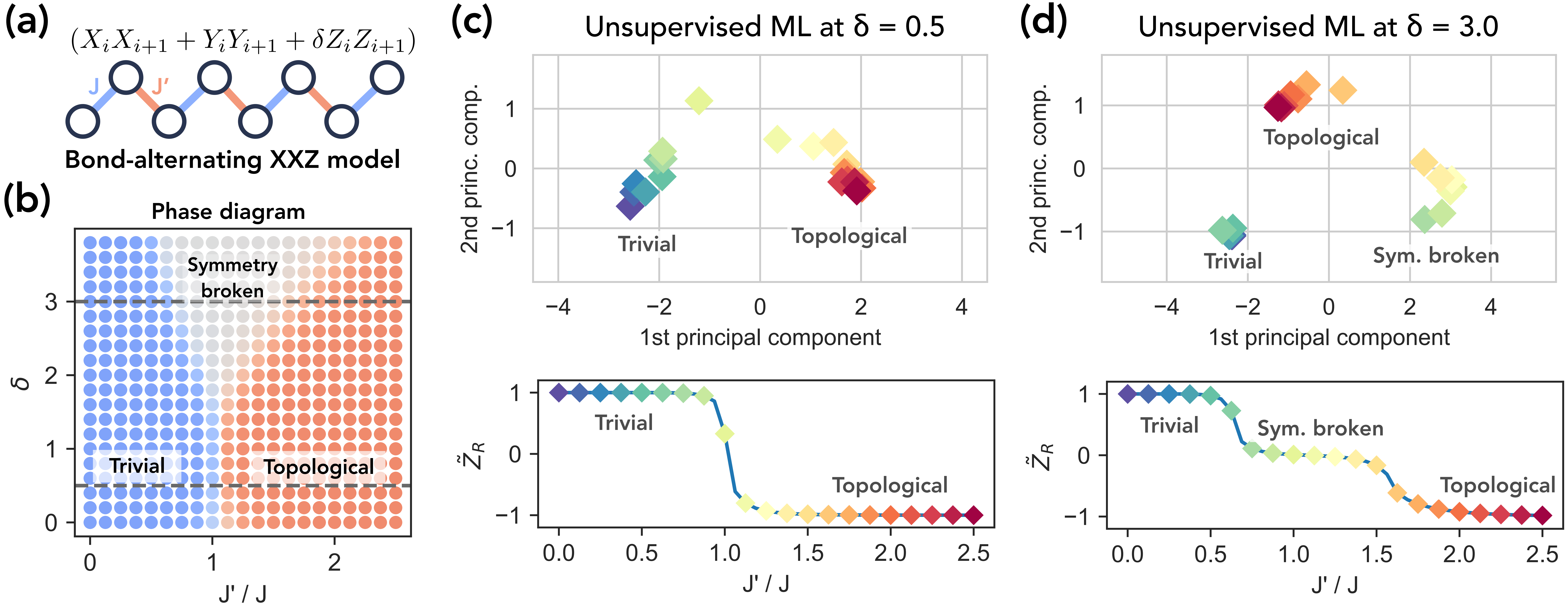
Symmetry-broken

Topological

Quantum phases of matter

1D Symmetry protected topological phases

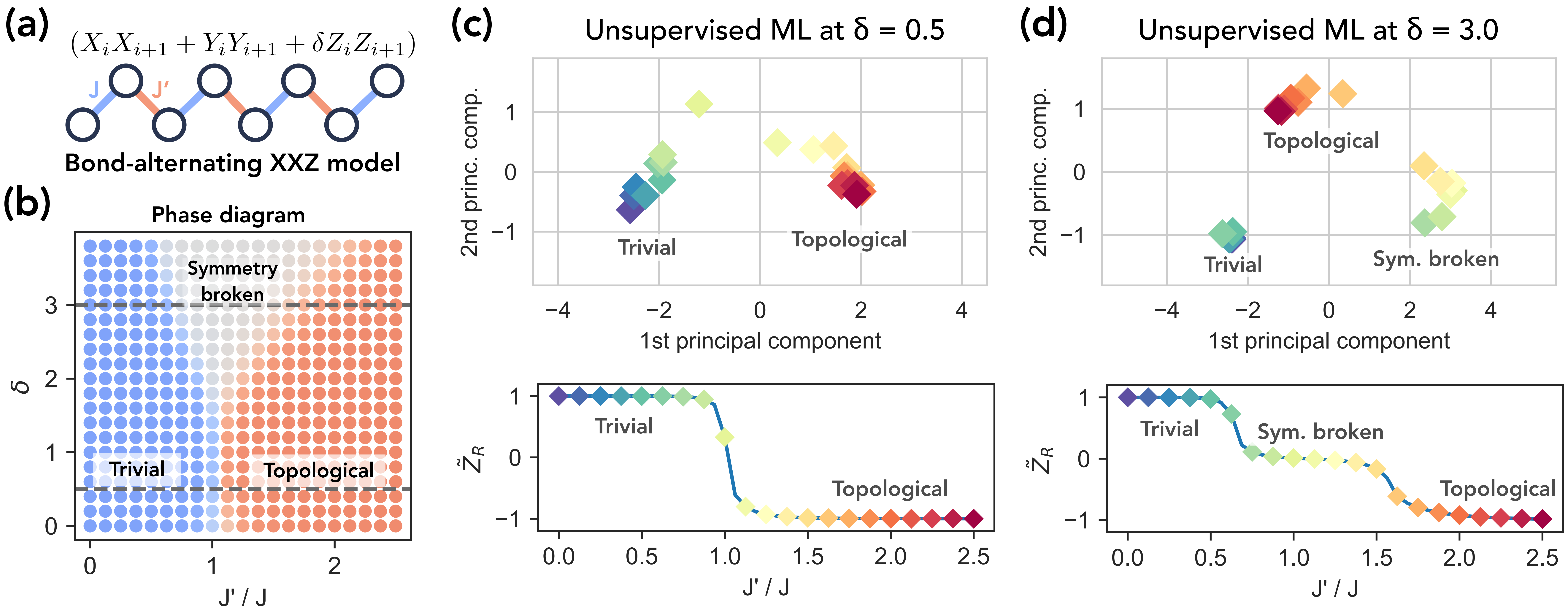
We consider $T = 500$ randomized measurements to construct classical shadows for each state.
The classical **unsupervised** ML model is a kernel PCA using the shadow kernel.



1D Symmetry protected topological phases

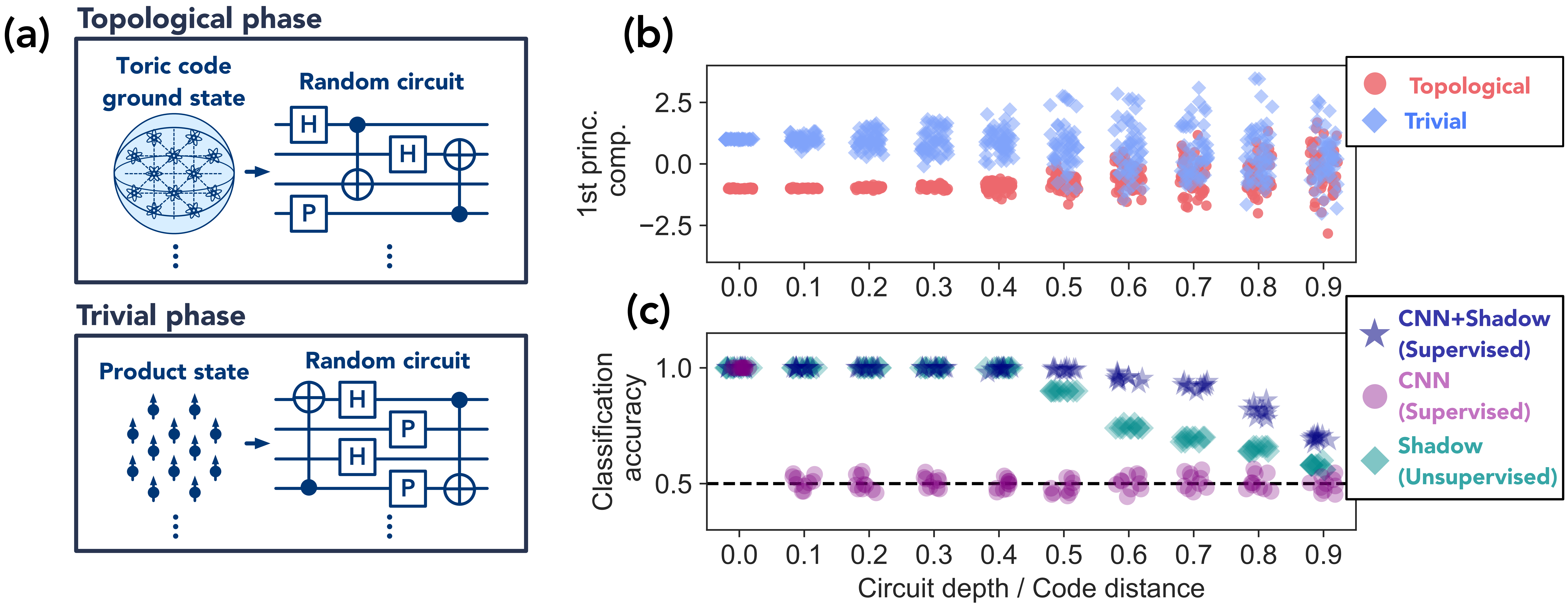
No labeled training data

We consider $T = 500$ rando ruct classical shadows for each state.
The classical **unsupervised** ML model is a kernel PCA using the shadow kernel.



2D topologically-ordered phases

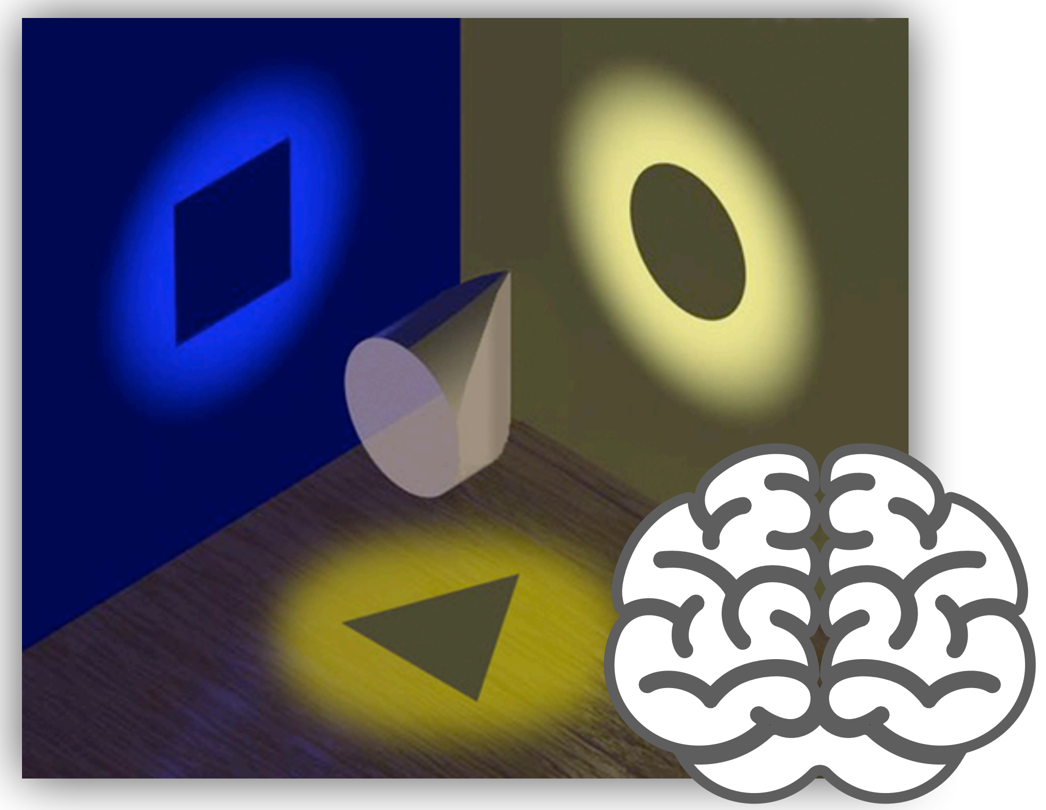
We consider $T = 500$ randomized measurements to construct classical shadows for each state.
The classical **unsupervised** ML model is a kernel PCA using the shadow kernel.



Code available at: https://github.com/hsinyuan-huang/provable-ml-quantum

Conclusion

- We prove that classical ML algorithms, informed by data from physical experiments, can address some challenging quantum many-body problems.
- As a consequence, we establish the advantage of classical ML models that learn from data over classical non-ML algorithms in a physically relevant task.
- **Open questions:**
  Advantage of ML over non-ML algorithms in more problems?
  Rigorous guarantee for other quantum many-body problems with classical/quantum ML?



Classical shadows enhanced with ML