Introduction to Quantum Machine Learning

Jack Y. Araz
Thomas Jefferson National Accelerator Facility

2nd COFI Advanced Instrumentation and Analysis Techniques School
December 12, 2023

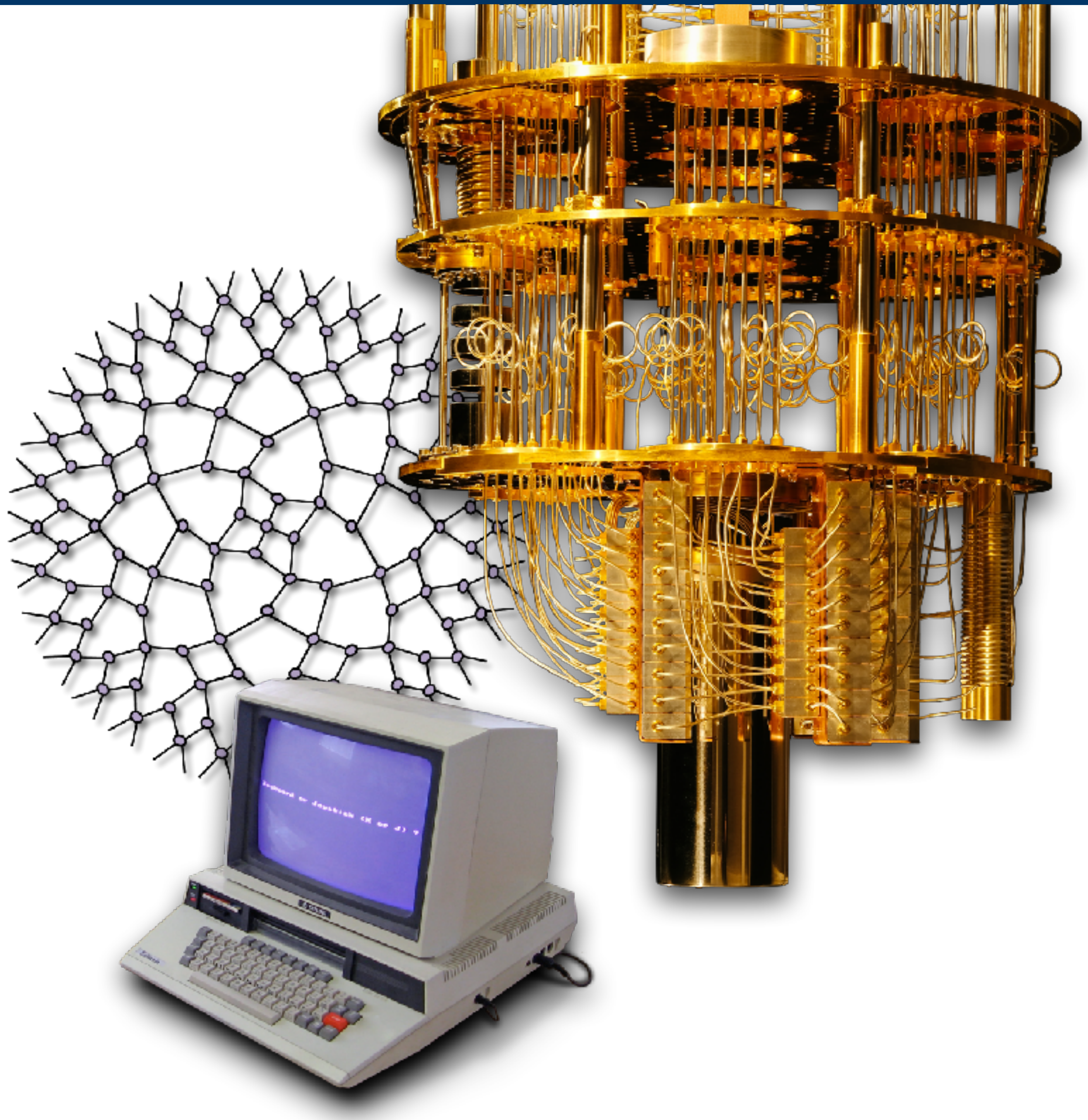
Jefferson Lab
EXPLORING THE NATURE OF MATTER

U.S. DEPARTMENT OF ENERGY | Office of Science

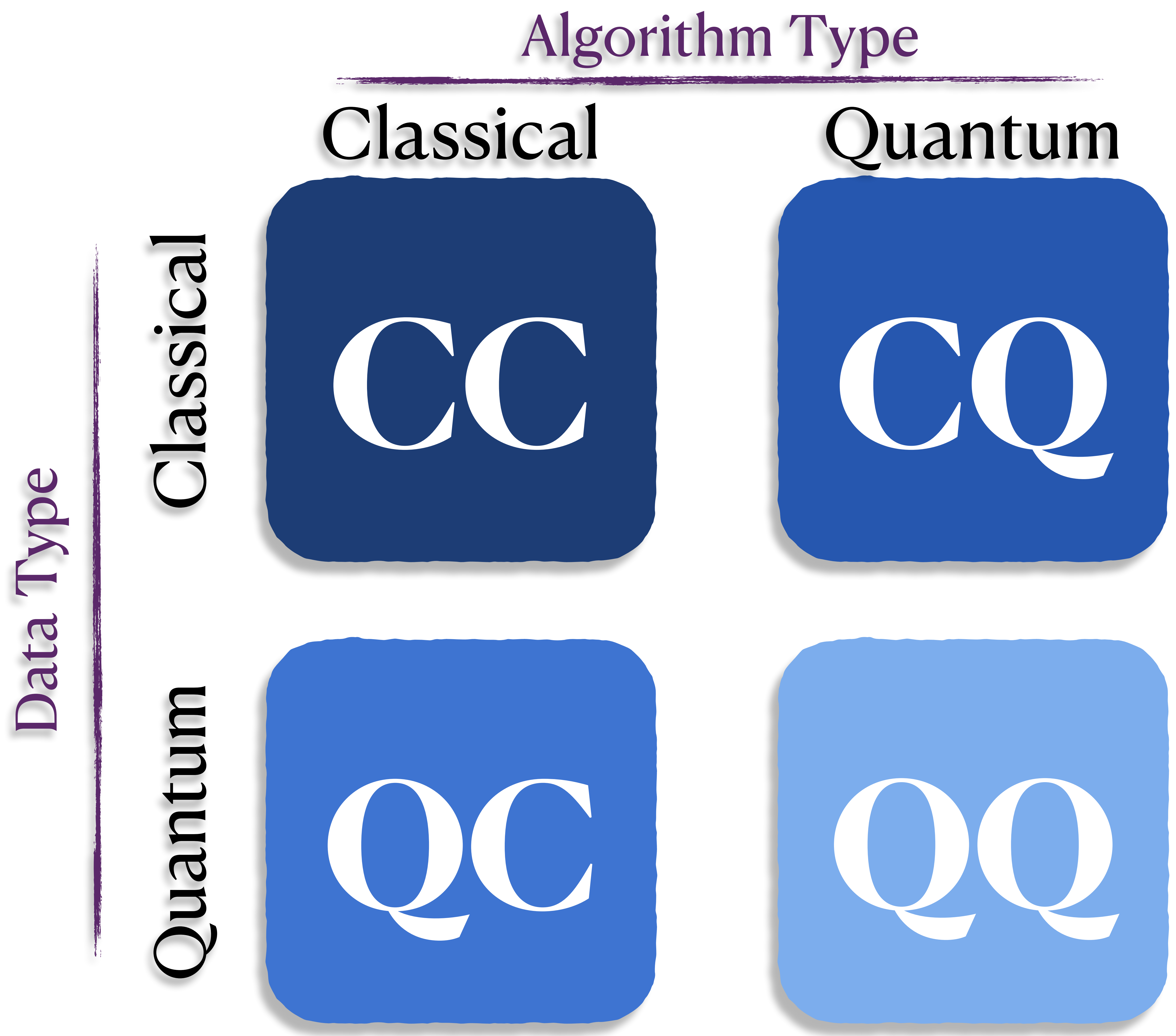
MCnet

Outline

- Introduction to Quantum Computing
- Connection to Machine Learning
- QML for field theories (code examples)
- QML for data analysis (code example)
- Limitations of QML
- Using theory for data analysis with QML
- Conclusion



Computational Paradigms



Computational Paradigms

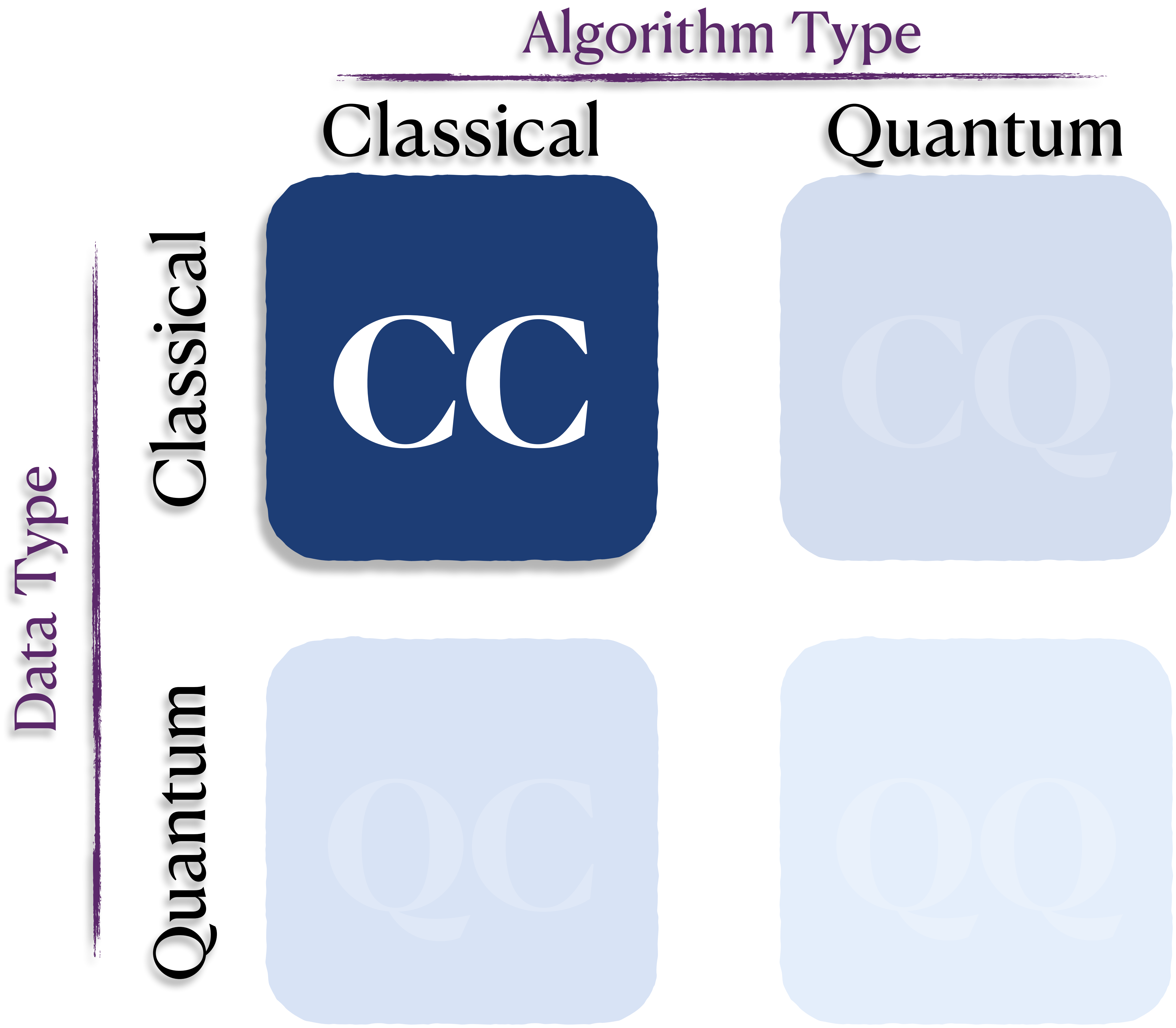
Algorithm Type

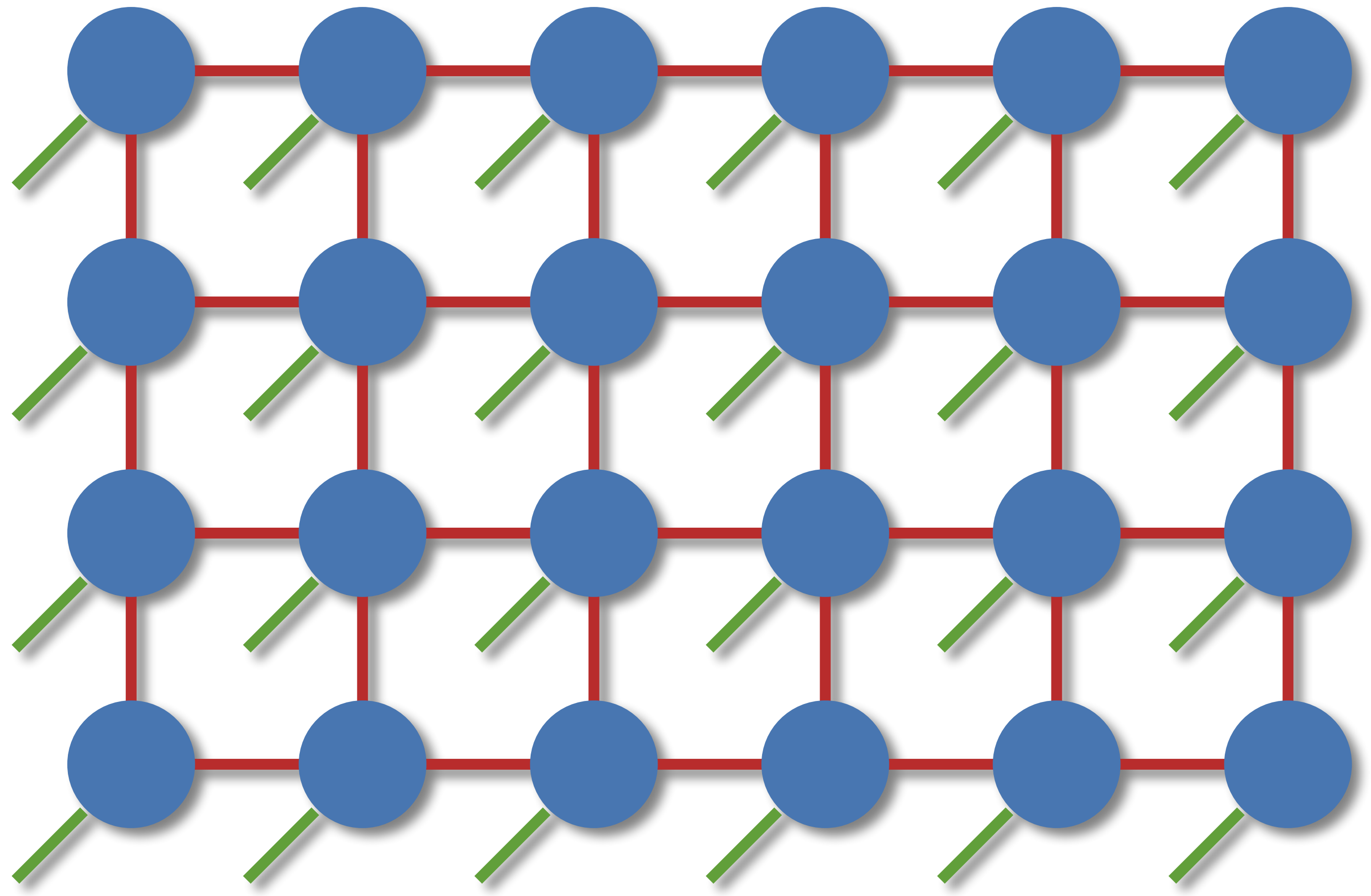
Data Type	Classical	Quantum
Classical	CC	CQ
Quantum	QC	QQ

Classical ⟷ Correlation

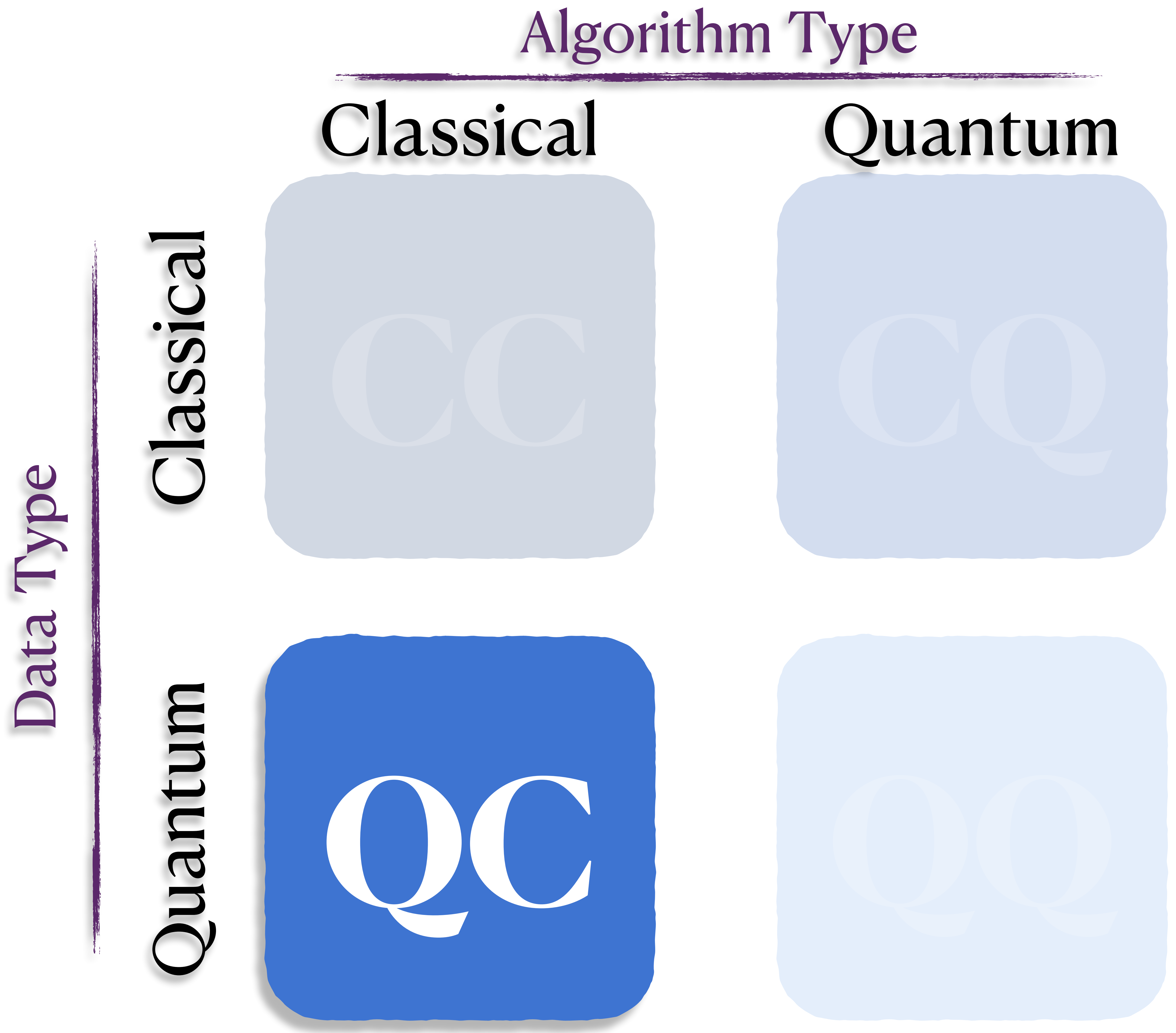
Quantum ⟷ Entanglement

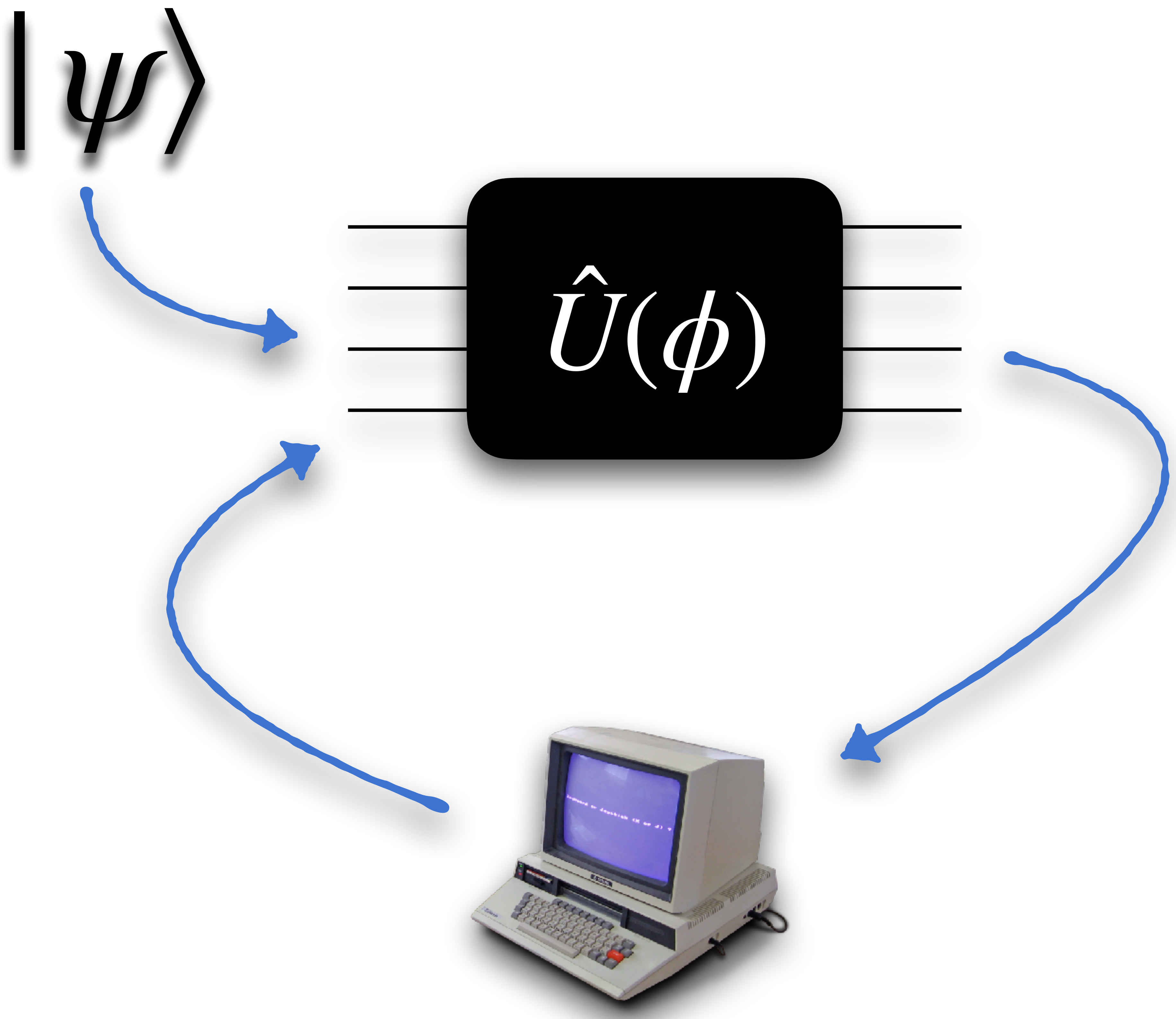
Computational Paradigms





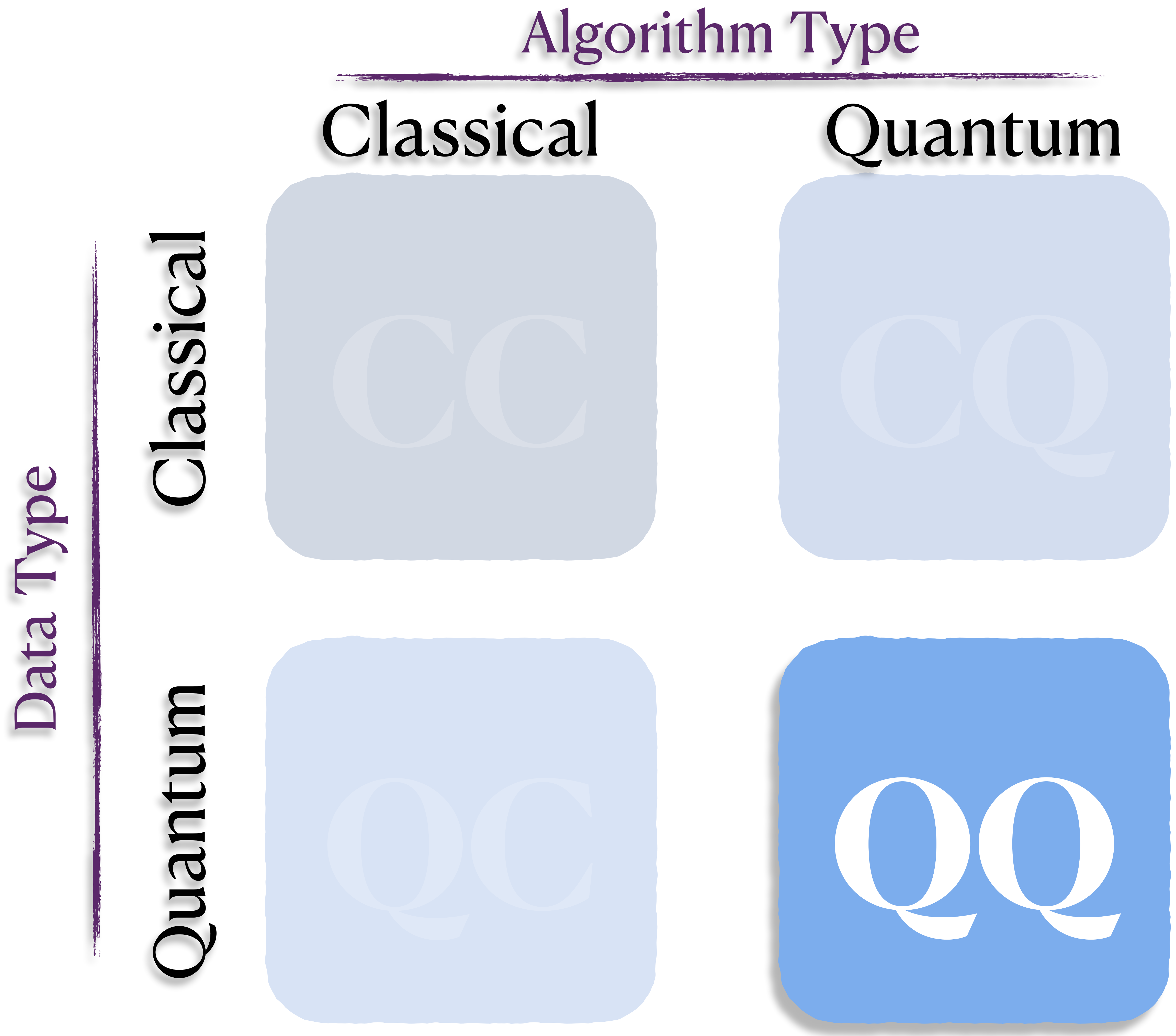
Computational Paradigms

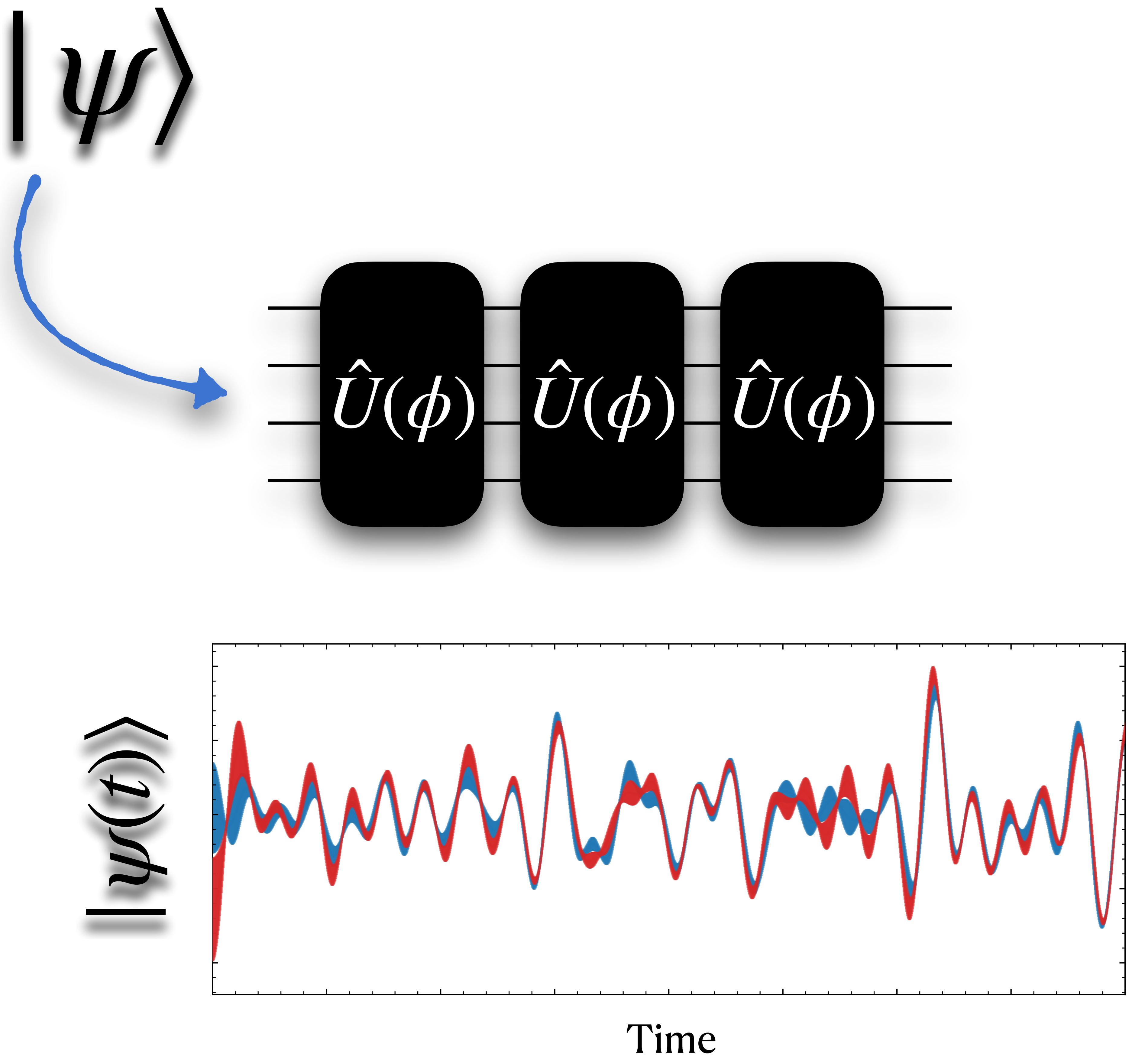




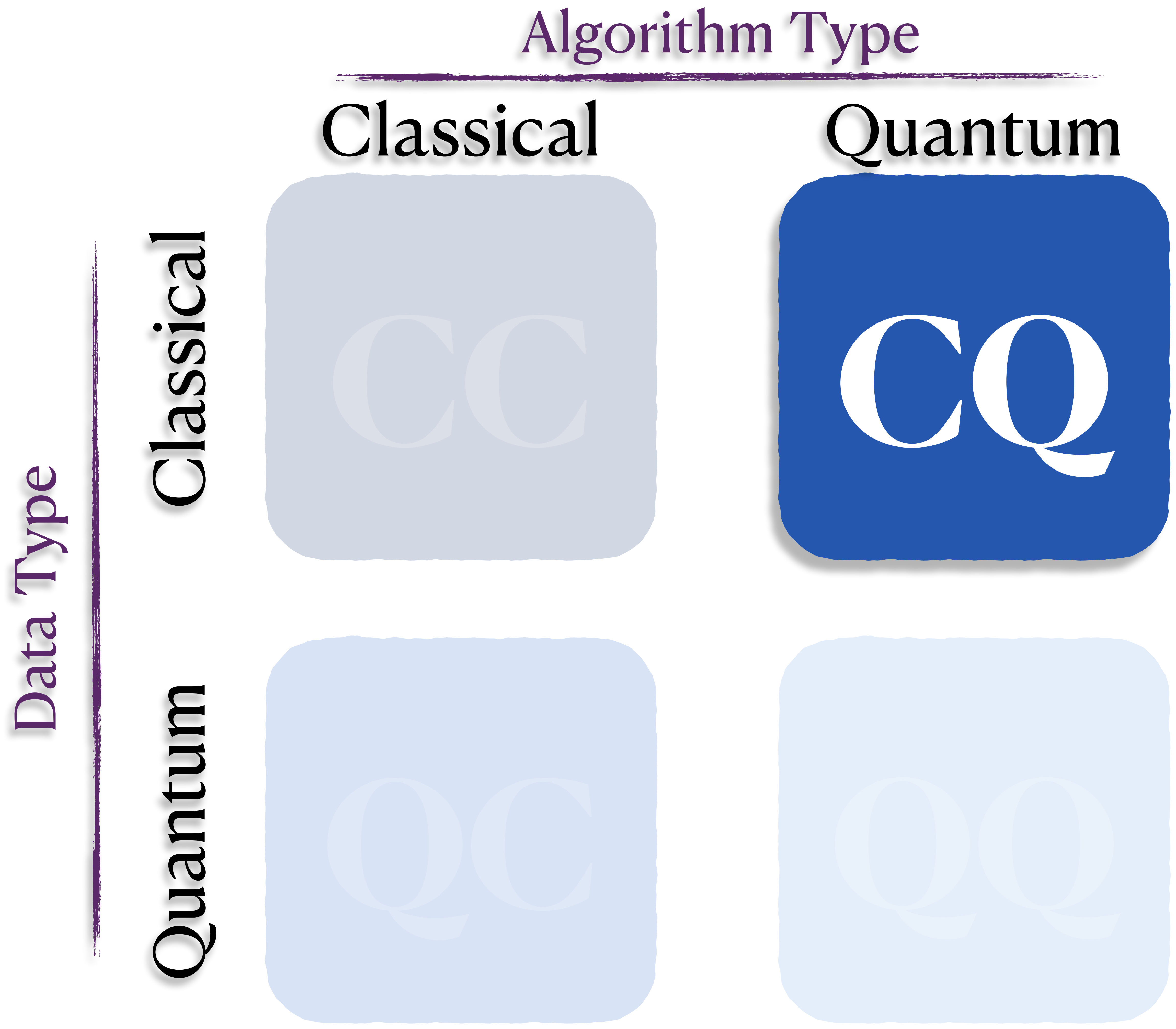
- Also Tensor Networks

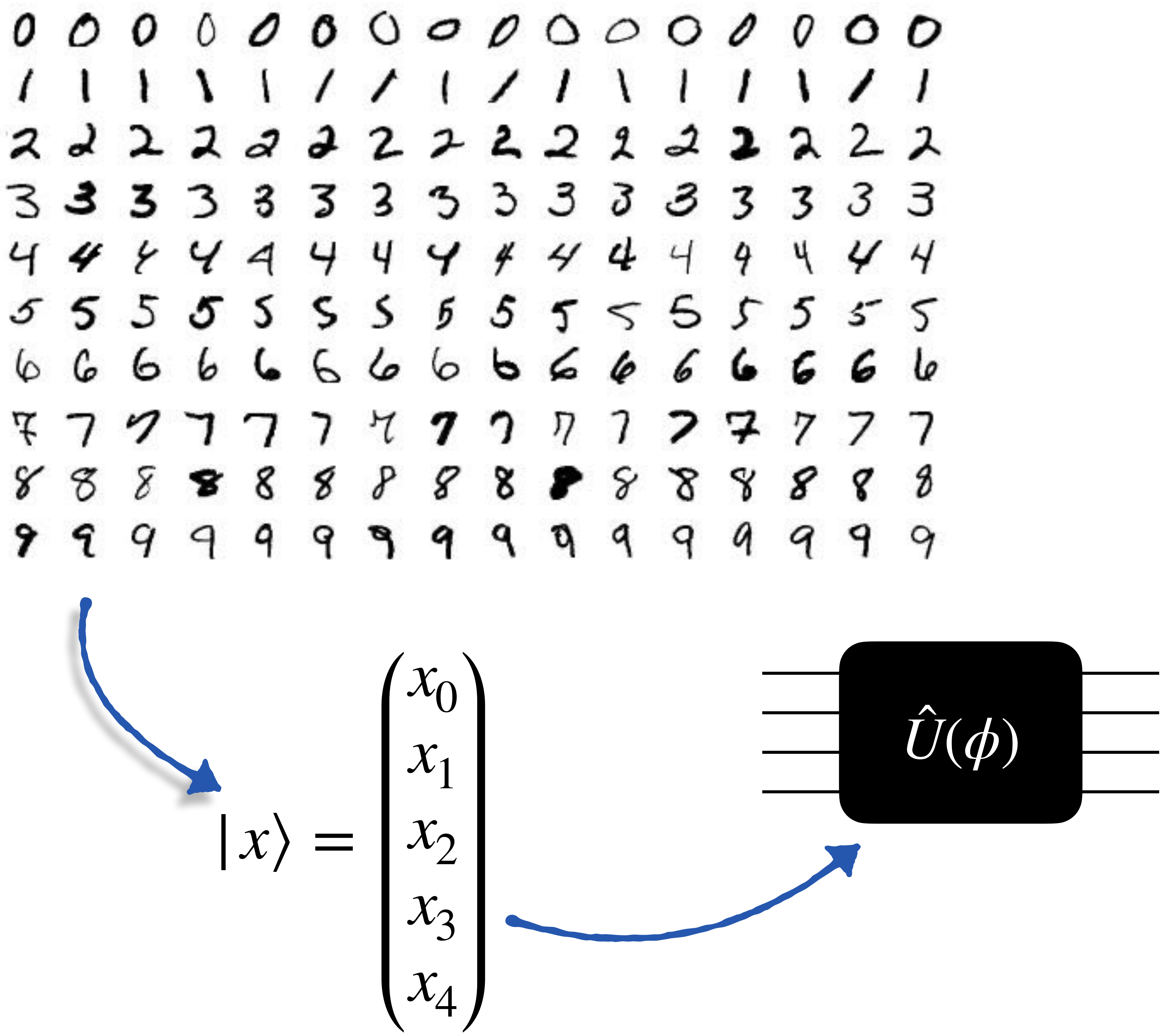
Computational Paradigms





Computational Paradigms





How does the Quantum Computer work (theoretically)?

$$\underbrace{\uparrow\downarrow\uparrow\uparrow\downarrow\cdots\downarrow\uparrow\downarrow\uparrow\uparrow}$$

Recall Quantum Mechanics 101

$$\left.\begin{aligned} a^{+}|\downarrow\rangle &= |\uparrow\rangle \\ a^{-}|\uparrow\rangle &= |\downarrow\rangle \end{aligned}\right\}$$

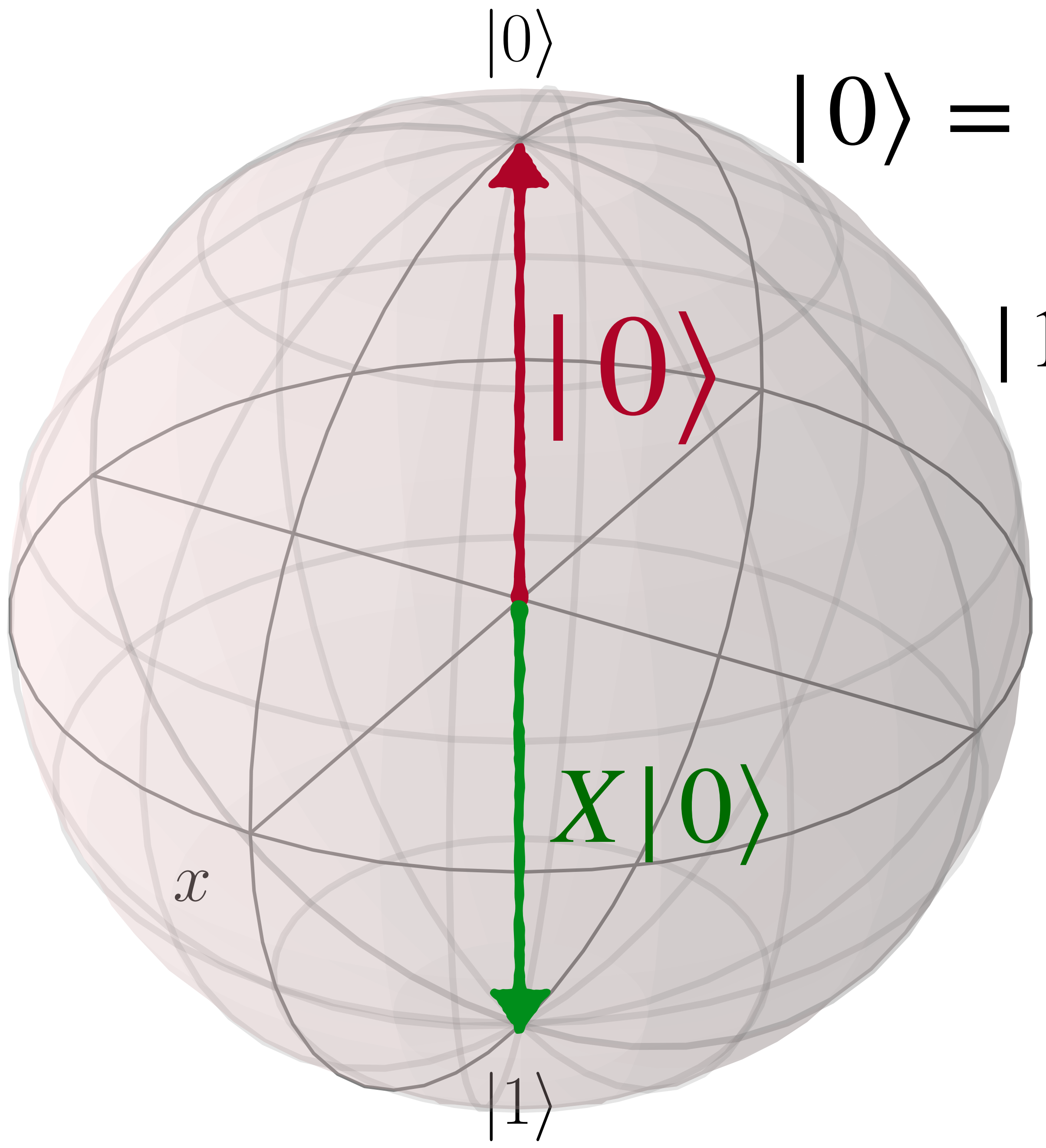
$$|\downarrow\rangle \rightarrow |0\rangle$$
$$|\uparrow\rangle \rightarrow |1\rangle$$
$$2a^{\pm} = X \pm iY$$

$SU(2)$ Generators

$$\{X,\ Y,\ Z,\ I\}$$

How does the Quantum Computer work (theoretically)?

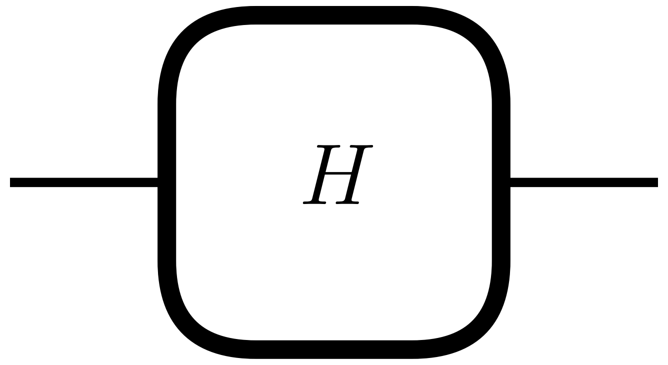
Recall Quantum Mechanics 101



$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$

$|1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$

- $X|0\rangle = |1\rangle,\ X|1\rangle = |0\rangle$
- $Y|0\rangle = i|1\rangle,\ Y|1\rangle = -i|0\rangle$
- $Z|0\rangle = |0\rangle,\ Z|1\rangle = -|1\rangle$



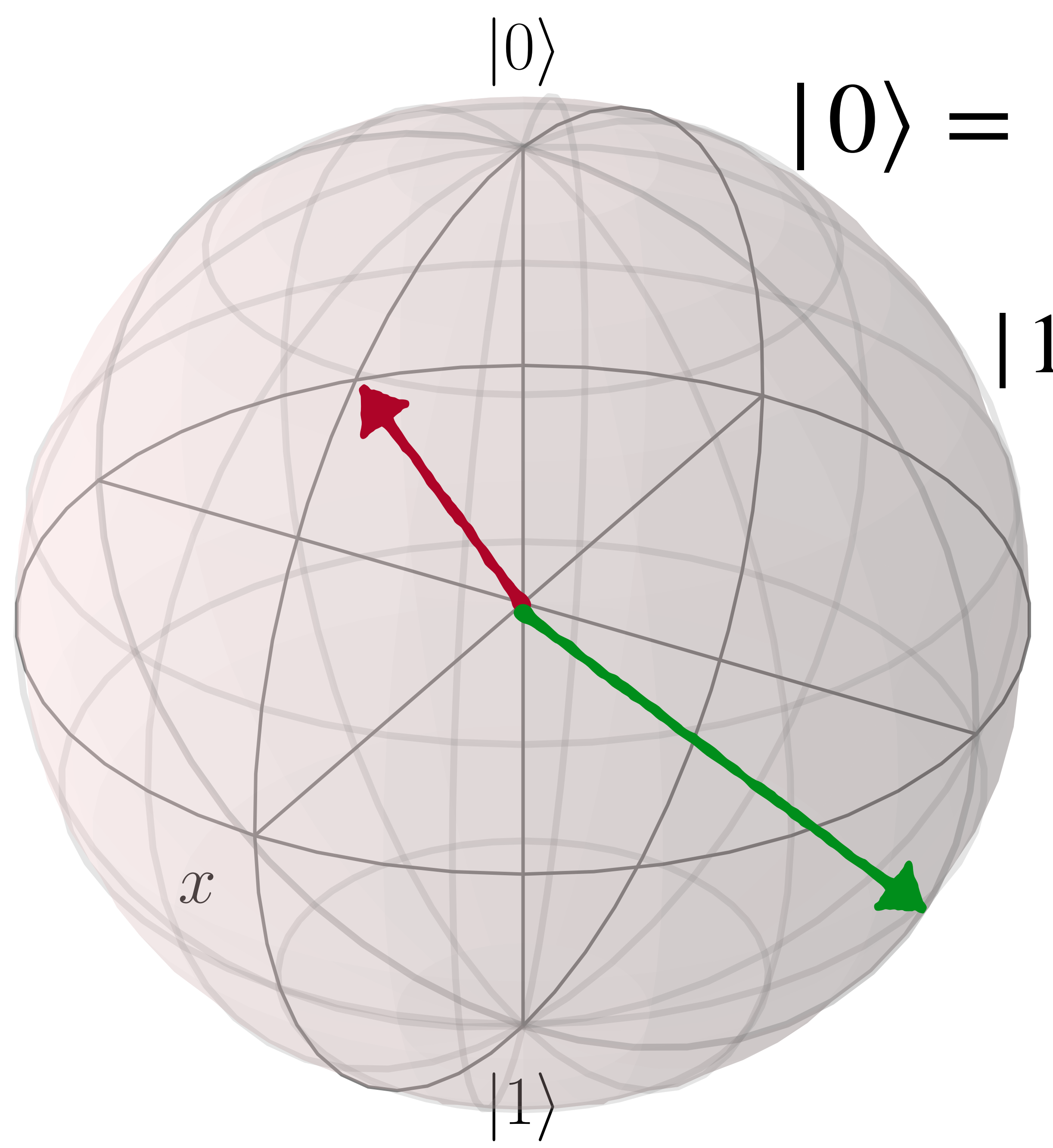
- $H|0\rangle = |+\rangle,\ H|1\rangle = |-\rangle$
- $Y|+\rangle = -i|-\rangle,\ Y|-\rangle = i|+\rangle$
- $Z|+\rangle = |-\rangle,\ Z|-\rangle = |+\rangle$

$|\pm\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle \pm |1\rangle\right)$

Hadamard gate (H) : $\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \to X = HZH$

How does the Quantum Computer work (theoretically)?

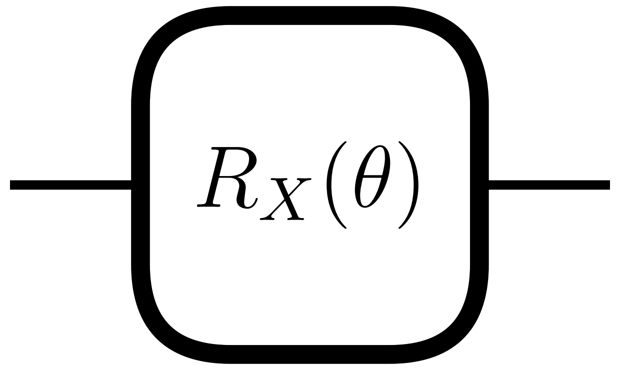
Recall Quantum Mechanics 101



$$|0\rangle = \begin{pmatrix}1\\0\end{pmatrix}$$

$$|1\rangle = \begin{pmatrix}0\\1\end{pmatrix}$$

- $R_X(\theta) = e^{iX\theta/2} = \begin{pmatrix}\cos\theta/2 & -i\sin\theta/2\\ i\sin\theta/2 & \cos\theta/2\end{pmatrix}$
- $R_Y(\theta) = e^{iY\theta/2} = \begin{pmatrix}\cos\theta/2 & -\sin\theta/2\\ \sin\theta/2 & \cos\theta/2\end{pmatrix}$
- $R_Z(\theta) = e^{iZ\theta/2} = \begin{pmatrix}e^{-i\theta/2} & 0\\ 0 & e^{i\theta/2}\end{pmatrix}$

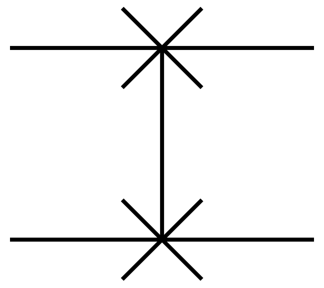


$$R_Y(\pi/2)\,|0\rangle = |-\rangle$$

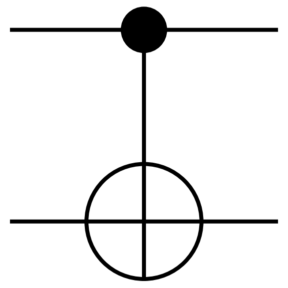
How does the Quantum Computer work (theoretically)?

$$|00\rangle = \begin{pmatrix}1\\0\\0\\0\end{pmatrix},\quad |10\rangle = \begin{pmatrix}0\\1\\0\\0\end{pmatrix},\quad |01\rangle = \begin{pmatrix}0\\0\\1\\0\end{pmatrix},\quad |11\rangle = \begin{pmatrix}0\\0\\0\\1\end{pmatrix}$$

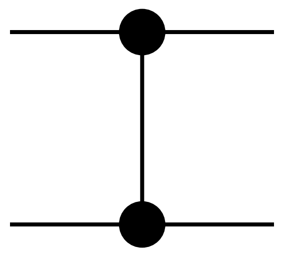
SWAP gate: $\begin{pmatrix}1&0&0&0\\0&0&1&0\\0&1&0&0\\0&0&0&1\end{pmatrix}$



CNOT gate: $\begin{pmatrix}1&0&0&0\\0&1&0&0\\0&0&0&1\\0&0&1&0\end{pmatrix}$



Controlled Z gate: $\begin{pmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&-1\end{pmatrix}$



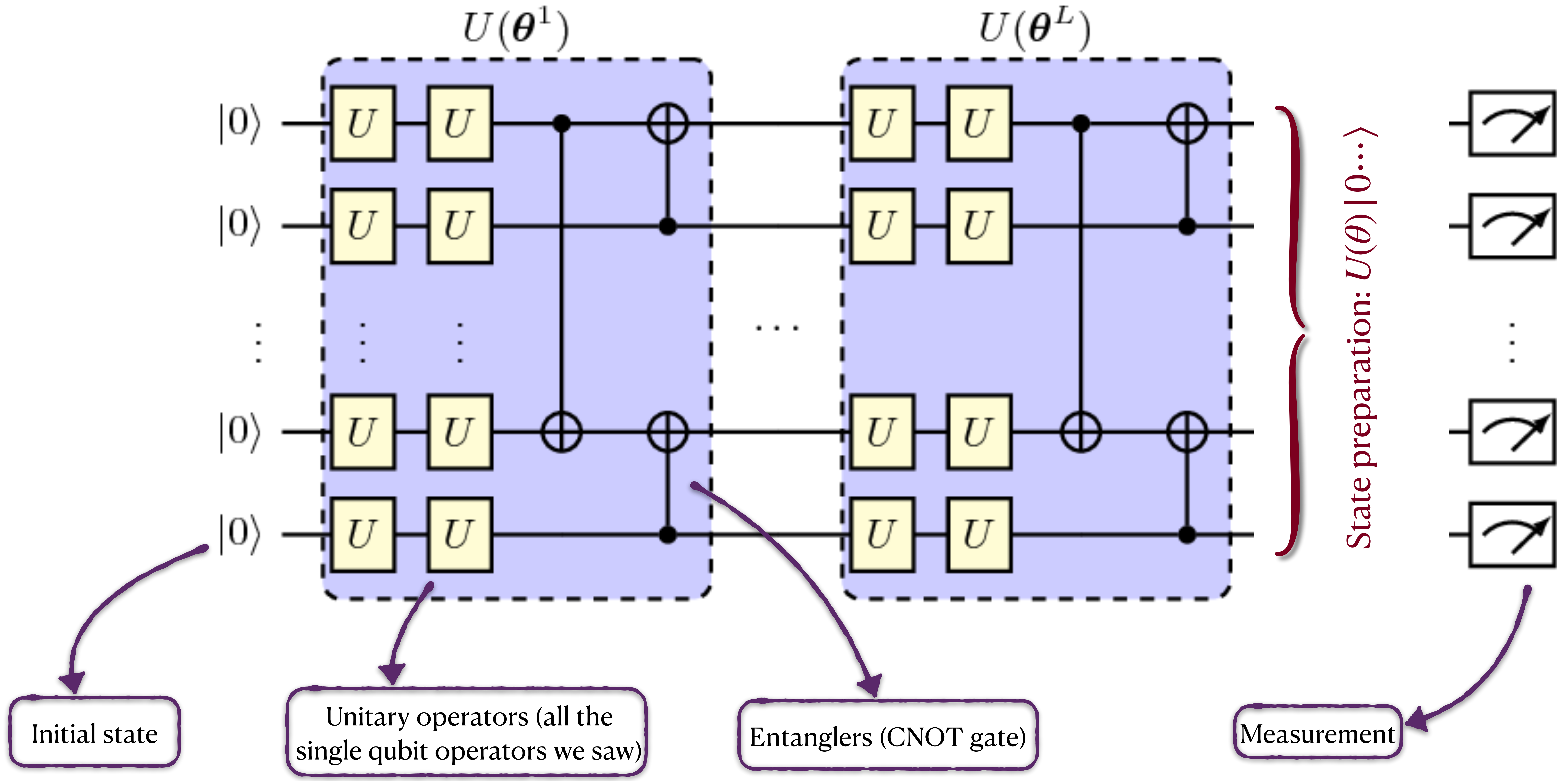
$$\text{CNOT}\,|xy\rangle = |x\rangle \otimes |x+y\rangle$$

$\text{CNOT}\,|0y\rangle = 0$

We can construct new operators by combining these

How does the Quantum Computer work (theoretically)?

Du, Hsieh, Liu, Tao; Phys. Rev. `20



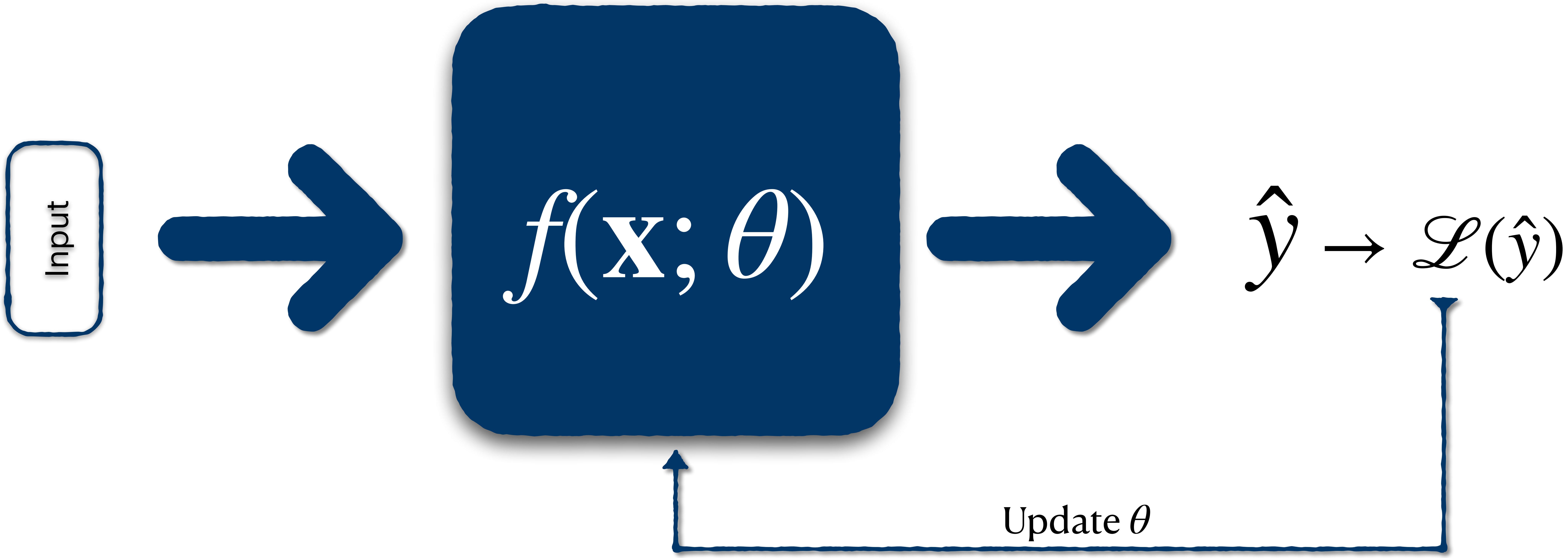
$\langle 0\,|\,U^\dagger(\theta)\mathcal{O}U(\theta)\,|\,0\rangle$

Make many measurements to construct the expectation value.

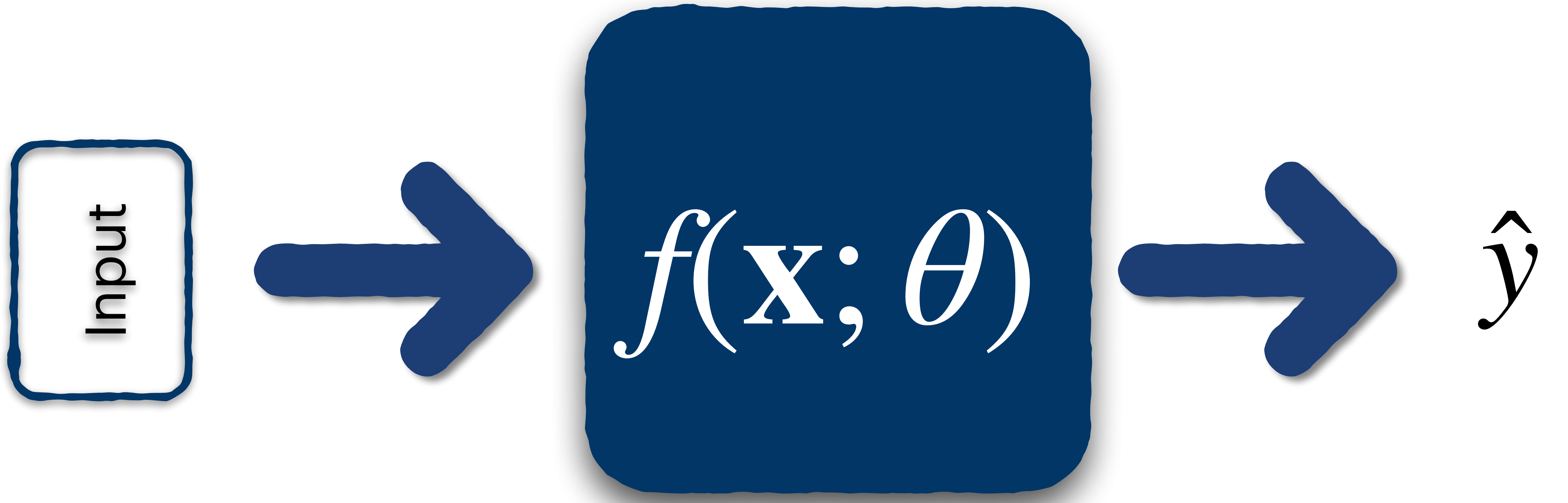
But what does all this have to do with ML?

But what does all this have to do with ML?

The name of the game is "optimisation"



But what does all this have to do with ML?

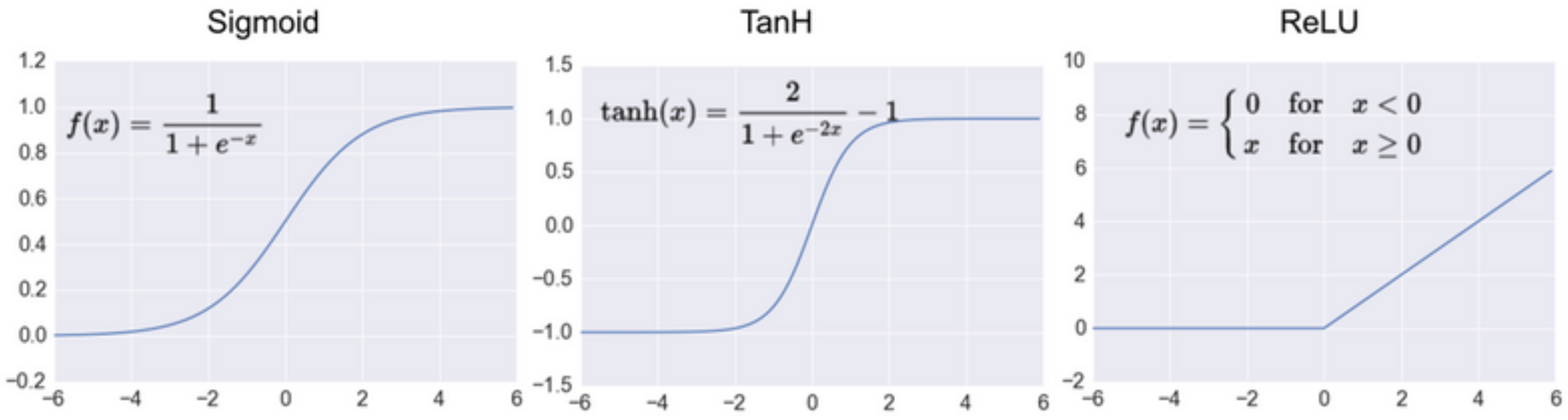


e.g. FC NN with 2 layers

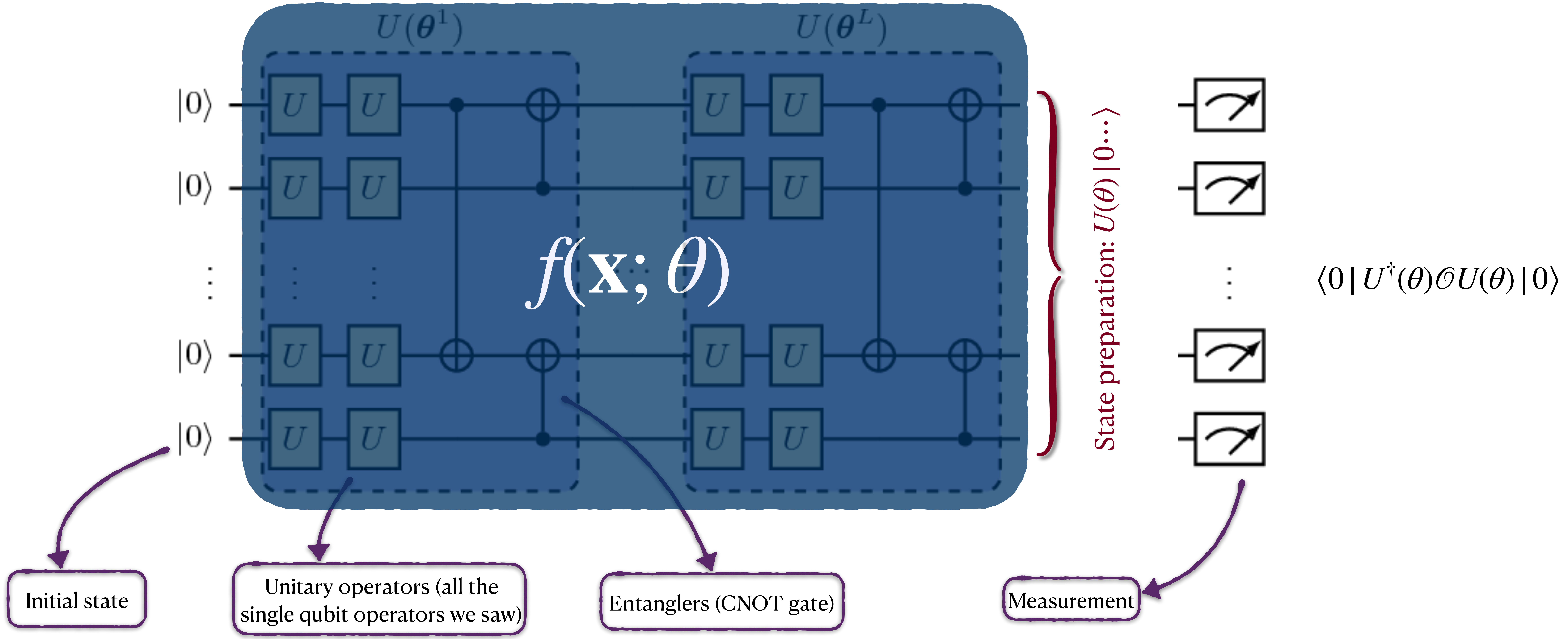
$$f_{\rm FC}(\mathbf{x};\theta) = \sigma_2\left(\mathscr{W}_2\cdot\sigma_1\left(\mathscr{W}_1\cdot\mathbf{x}+\mathscr{B}_1\right)+\mathscr{B}_2\right)$$

$$\forall\theta\in\mathscr{W}_i \text{ or } \mathscr{B}_i$$

$\sigma_i :=$ Activation Function



But what does all this have to do with ML?



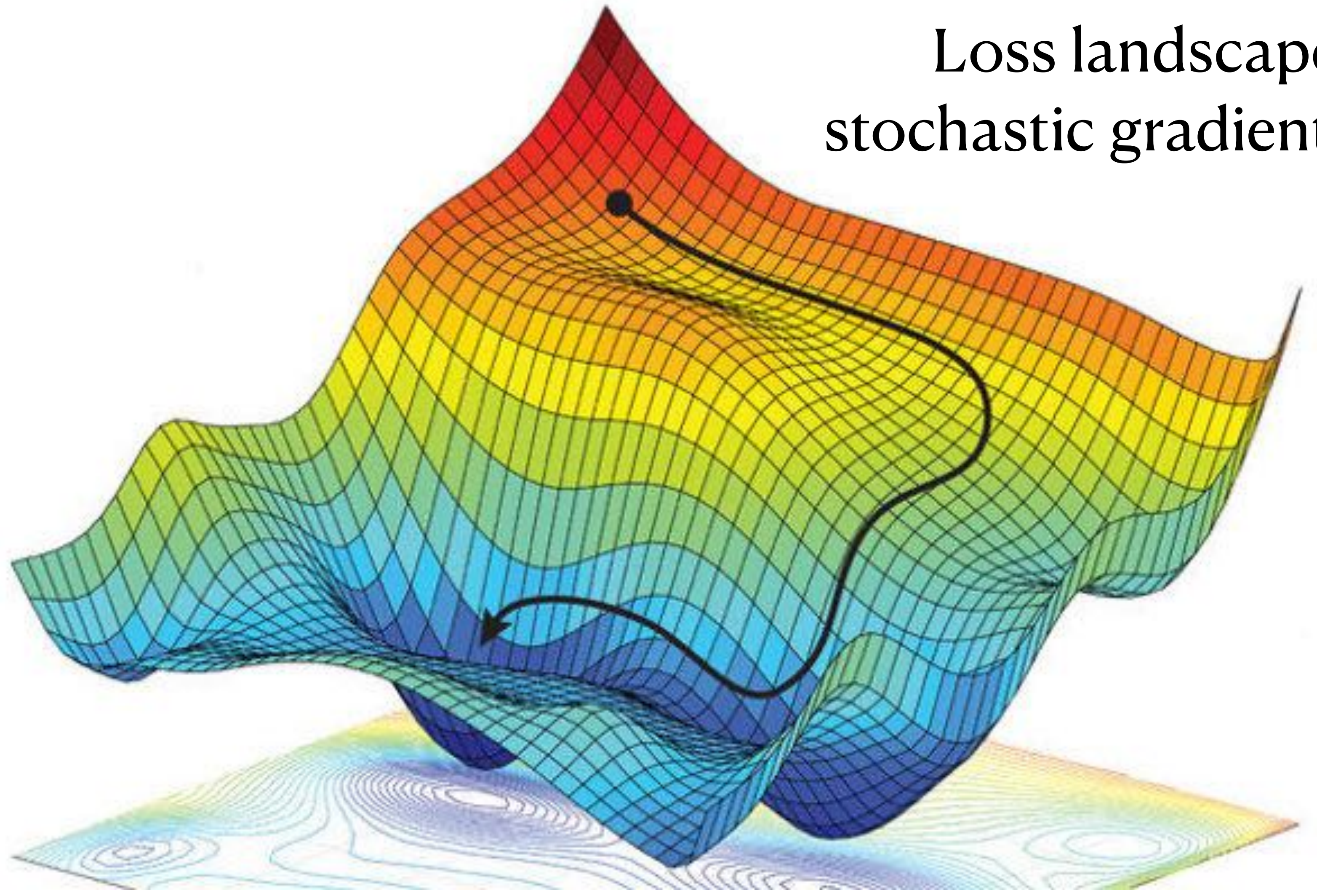
But what does all this have to do with ML?

Input → $f(\mathbf{x}; \theta)$ → $\hat{y}$

ELVET

$\mathcal{F}(\mathbf{x}) = 0$

JYA, Criado, Spannowsky; arXiv: 2103.14575



Loss landscape for stochastic gradient descent

Training

- Objective function: cross-entropy, mean squared error or a differential equation or variational problem

$$\underset{\theta \in \mathscr{W}}{\arg\min}\ \mathscr{L}(\,\cdot\,,\,\cdot\,)$$

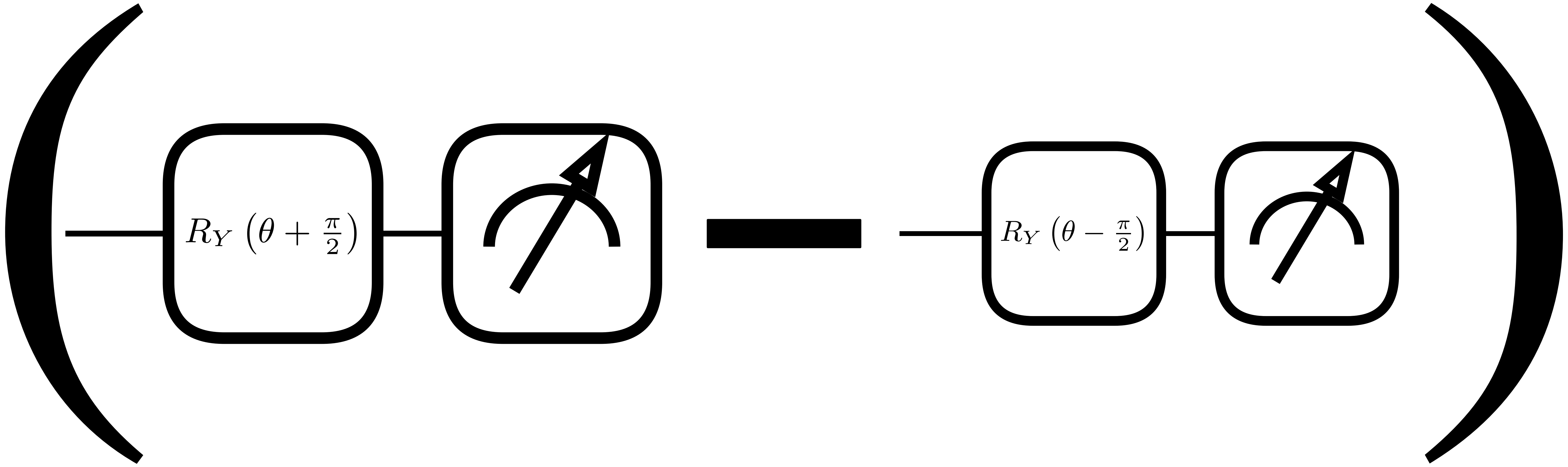
What is the gradient of a quantum circuit?

Quantum computer can not compute gradients!

Parameter Shift

$$\nabla_\theta f(\theta) = \frac{1}{2}\left[f(\theta + \pi/2) - f(\theta - \pi/2)\right]$$

$$\nabla_\theta f(\theta) = \frac{1}{2}$$



How to find ground & thermal states with QML?

How can we find the ground state of a Hamiltonian?

- Given a quantum many-body Hamiltonian, one can use the following methods to compute the ground state:
  - Exact diagonalisation (limited number of sites)
  - Monte Carlo techniques (sign problem)
  - Tensor Networks (limited entanglement structure)
- Beyond classical vs quantum, this is important for ground state preparation, which is theoretically expensive.

$$E_0 \leq \frac{\langle \Psi | H | \Psi \rangle}{\langle \Psi | \Psi \rangle}$$

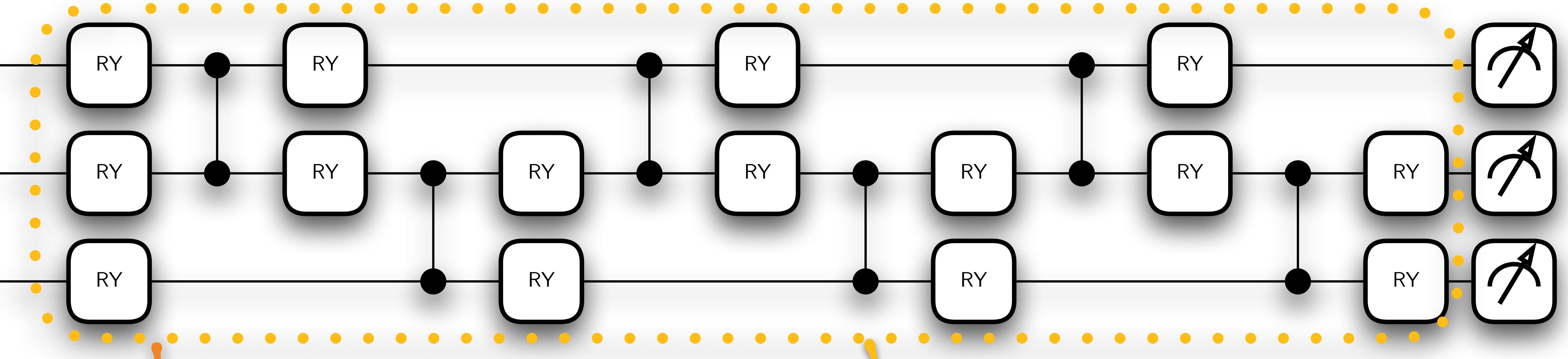
$$E_{VQE} = \min_{\phi} \langle 0 | U^{\dagger}(\phi)\, H\, U(\phi) | 0 \rangle$$

Variational Quantum Circuit

$$H = \sum_i a_i P_i \quad , \quad E_{VQE} = \min_{\phi} \sum_i a_i \langle 0 | U^{\dagger}(\phi) P_i U(\phi) | 0 \rangle$$

Ex I: Variational Quantum Eigensolver

$$H = \sum_i \left(X_i X_{i+1} + Y_i Y_{i+1}\right) + \eta \sum_i Z_i \quad \rightarrow \quad X_0 \otimes X_1 + Y_0 \otimes Y_1 + X_1 \otimes X_2 + Y_1 \otimes Y_2 + \eta(Z_0 + Z_1 + Z_2)$$

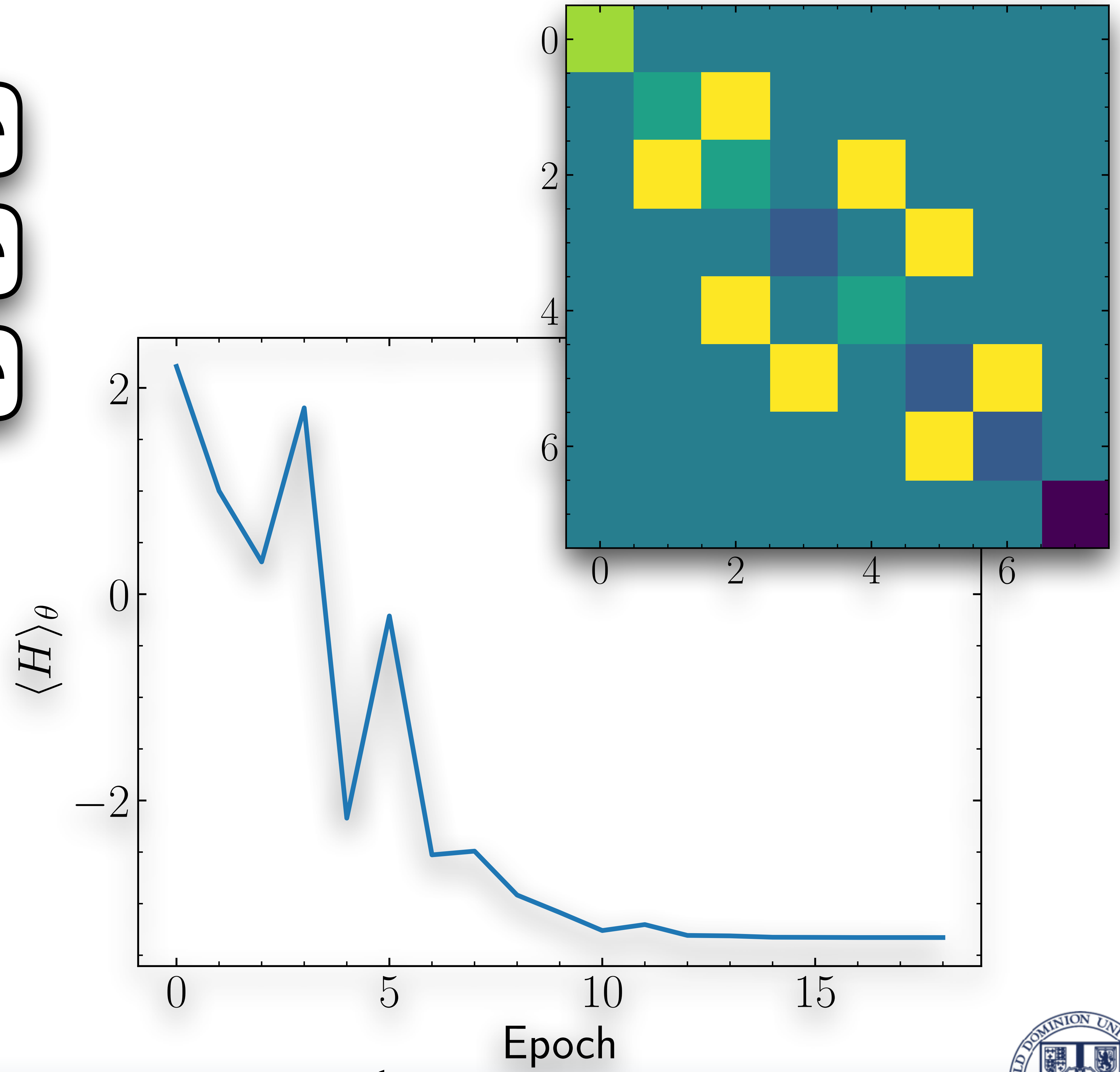


15 parameters

$$\langle H \rangle_\theta = \langle 0 | U^\dagger(\theta)\, H\, U(\theta) | 0 \rangle$$

VQC

Energy from exact diagonalisation: -3.328
Reconstructed ground state energy: -3.328

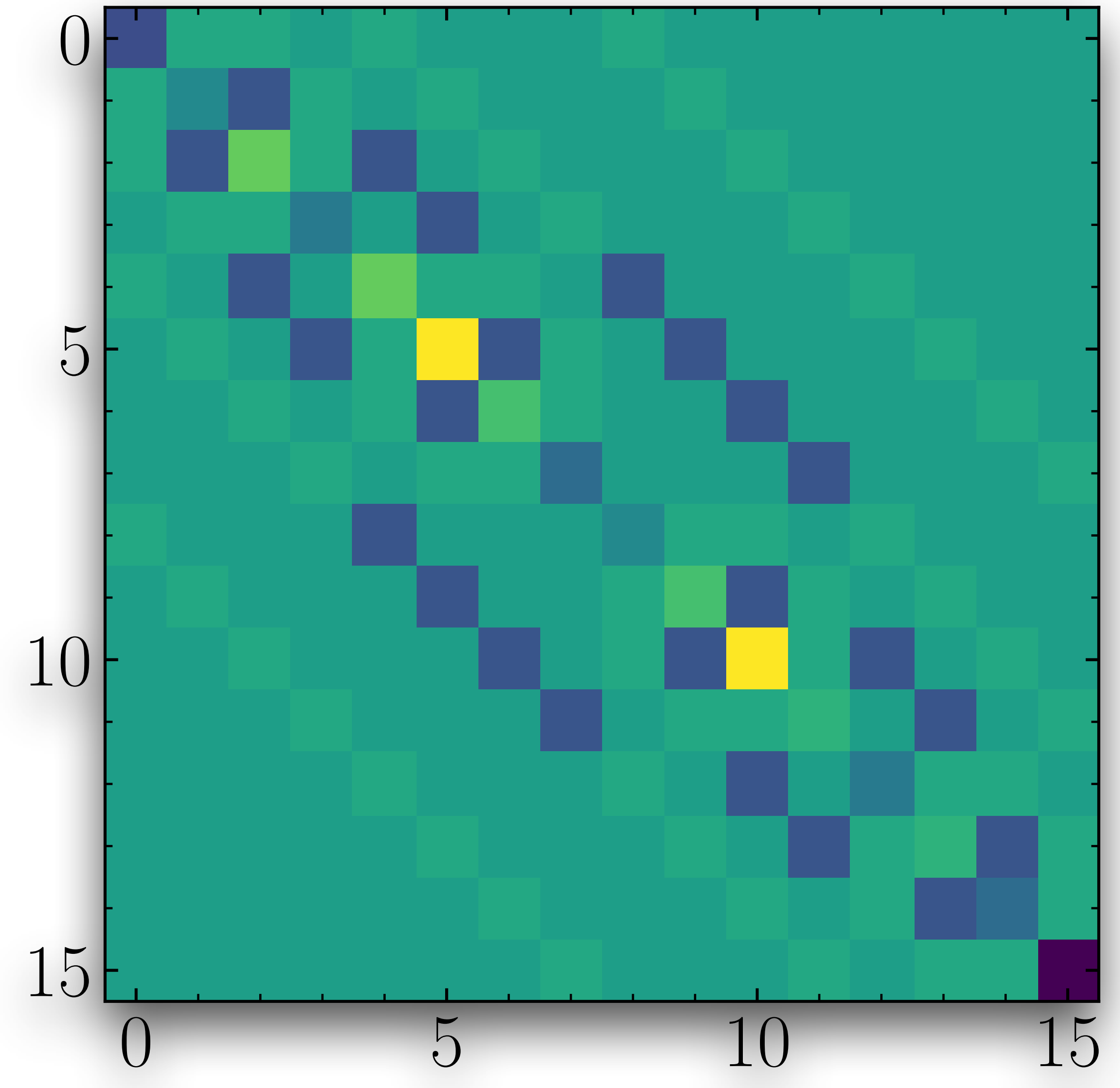


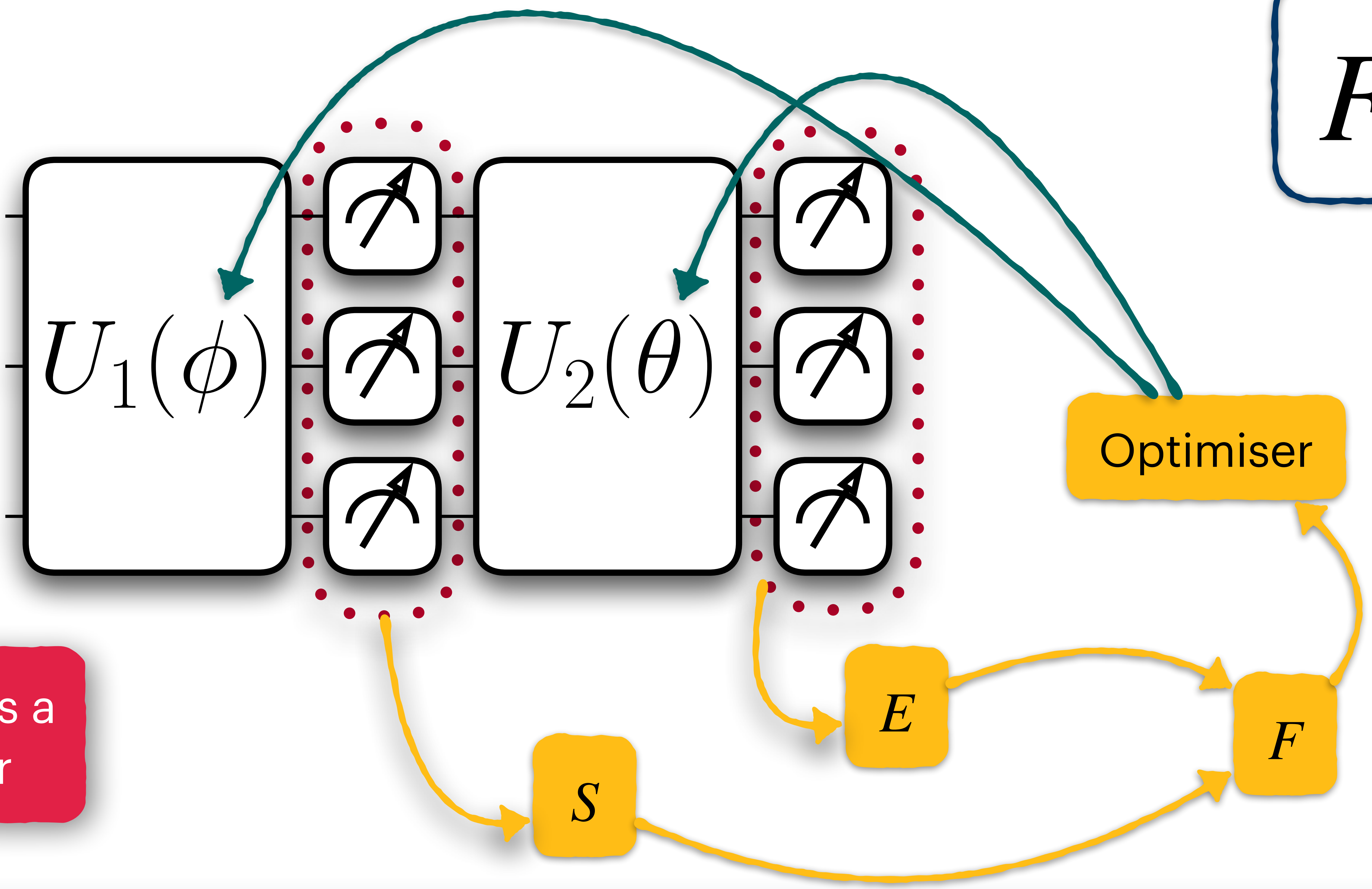
Ex II: Thermal state preparation

$$\rho_{th} = \frac{e^{-\beta H}}{Z} \quad , \quad Z = \mathrm{Tr}\left[e^{-\beta H}\right]$$

$$\beta = 1/T$$

$$H = \sum_{\langle i,j \rangle} J(X_i X_j + Y_i Y_j + Z_i Z_j) + \sum_i (J_x X_i + J_z Z_i)$$





Free Energy

$$F = E - TS$$

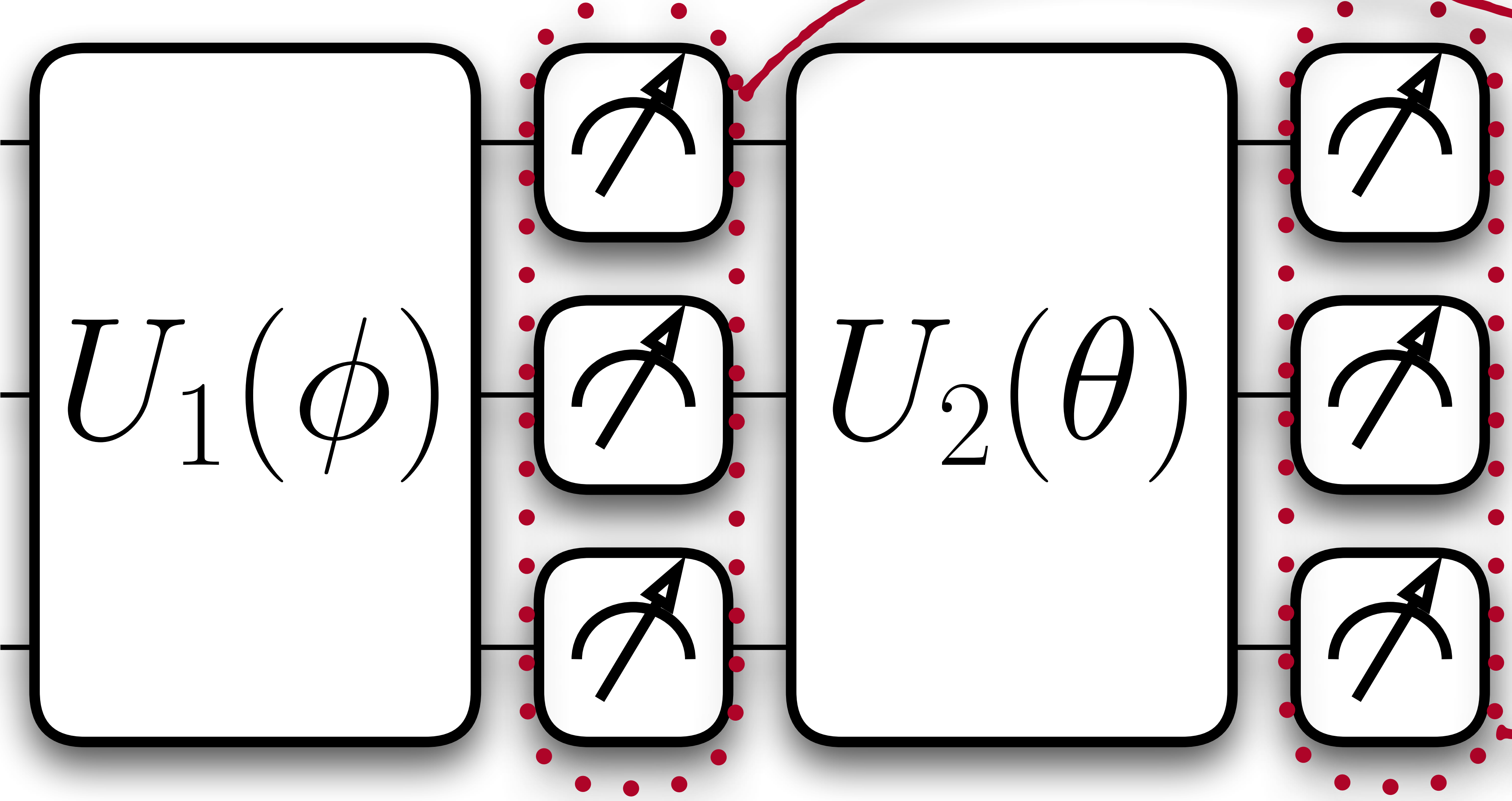
$$\langle H \rangle_\theta$$

$$S = -\mathrm{Tr}\left[\rho_{th} \log \rho_{th}\right]$$

Quantum Computer is a pure state simulator

We don't have access to ρ_{th}

Ex II: Thermal state preparation



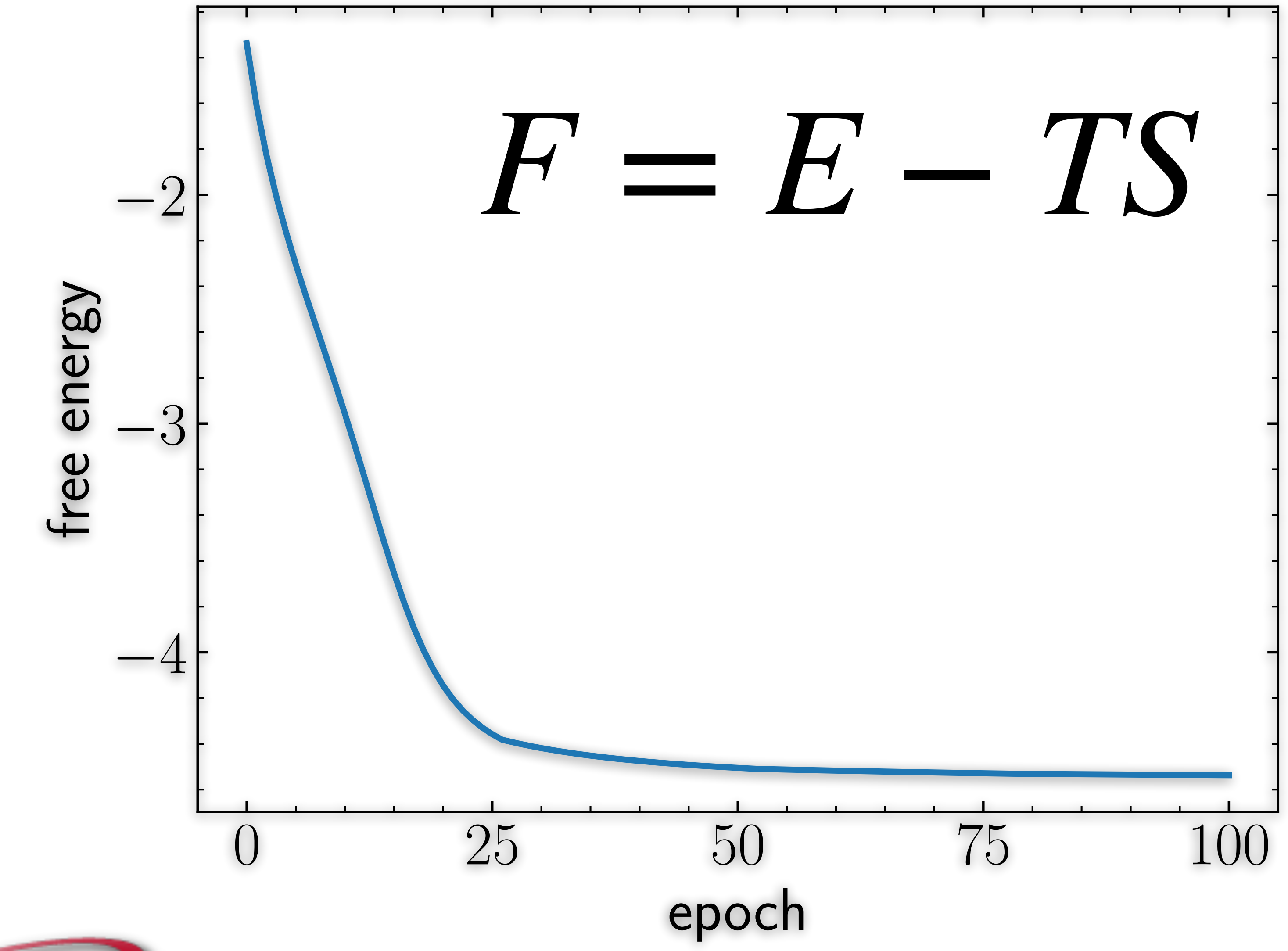
Shannon Entropy

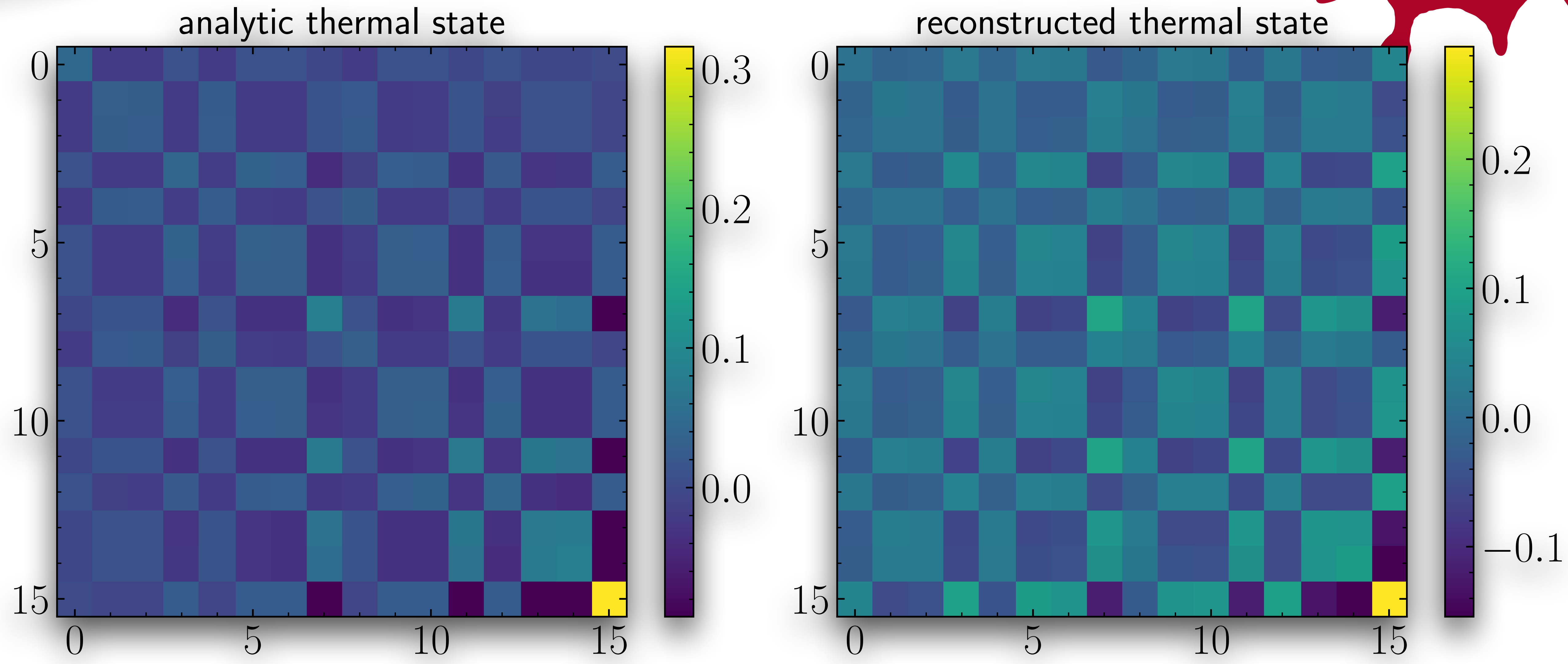
$$S = -\sum_i p_i \log p_i$$

$$\rho_{th} = \sum_i p_i |\psi_i\rangle\langle\psi_i|$$

$$\langle H\rangle_\theta = \sum_i p_i \langle\psi_i| U_2^\dagger(\theta)\ H\ U_2(\theta) |\psi_i\rangle$$

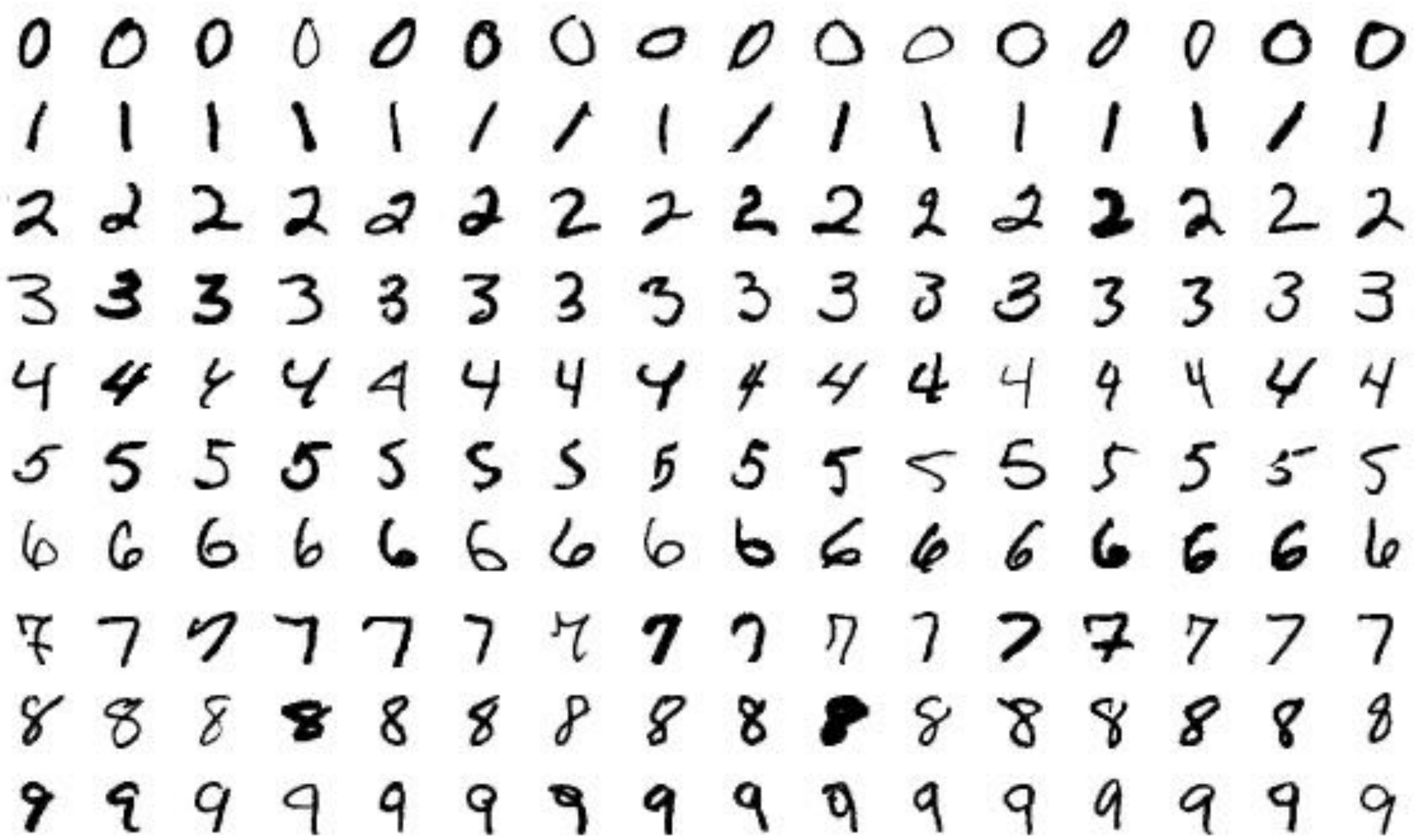
$$X|0\rangle = |1\rangle$$

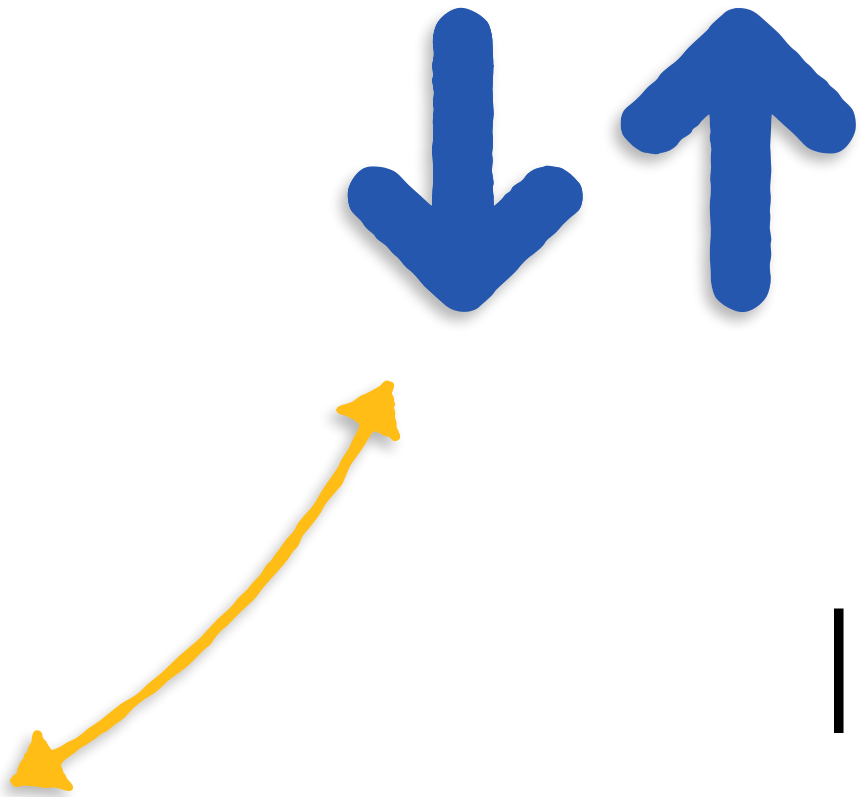




Data Encoding

Data Encoding





$$|X\rangle = R_Y(X\pi/2)\,|0\rangle$$

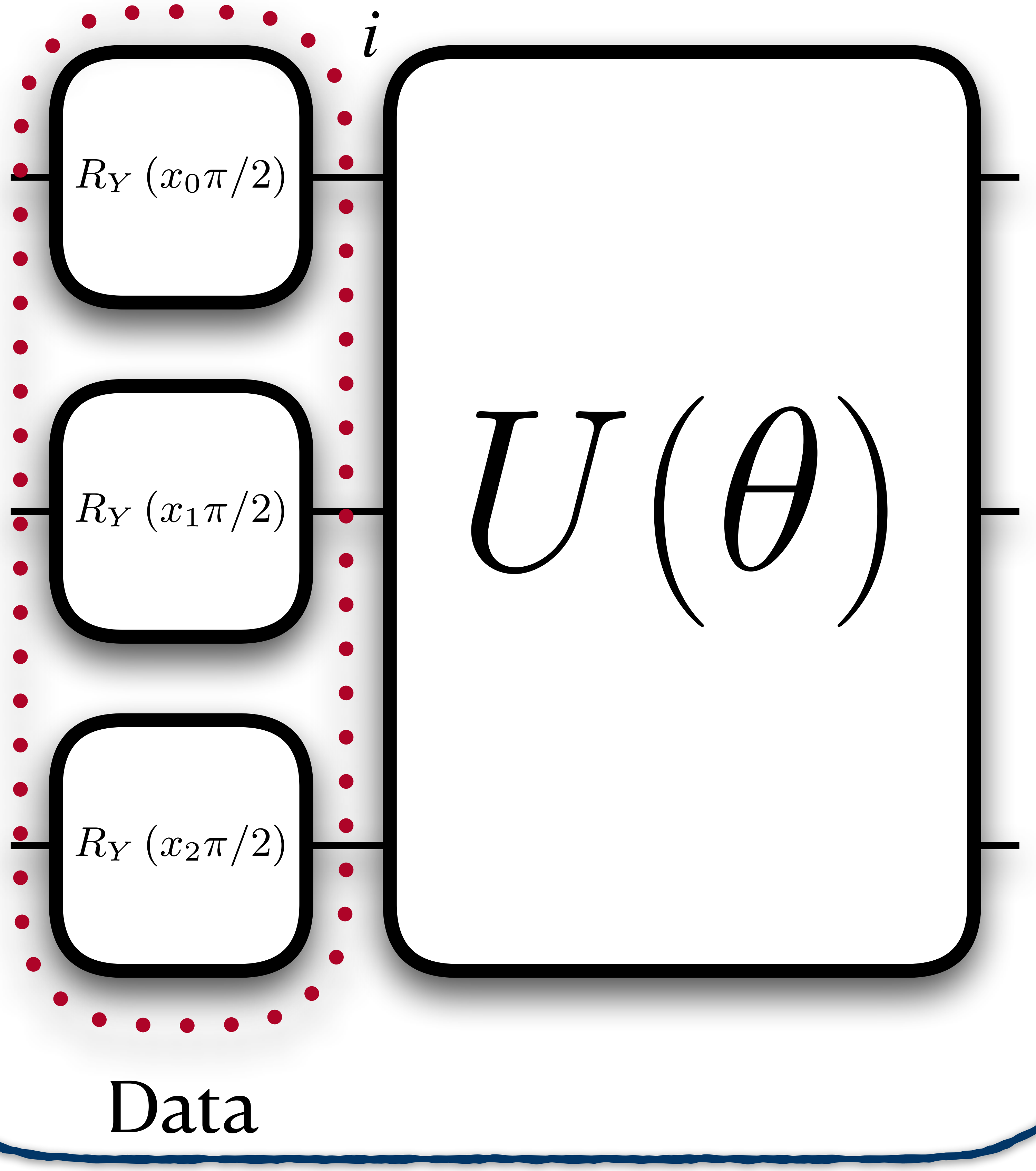
$$|x_i\rangle = \begin{bmatrix}\cos(x_i\pi/2)\\ \sin(x_i\pi/2)\end{bmatrix}$$

Jack Y. Araz

Data Encoding

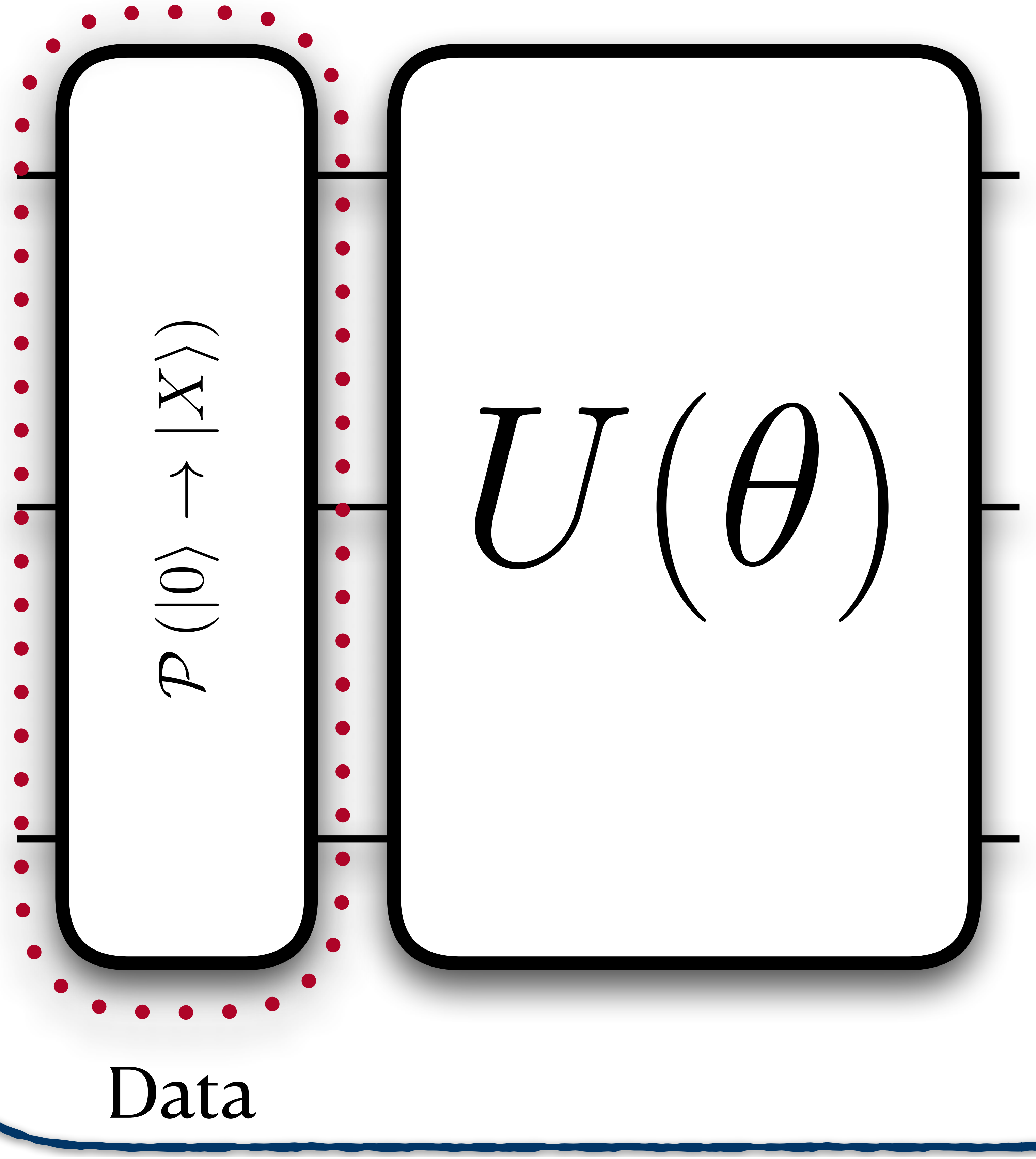
Angle Encoding

$$|X\rangle = \sum_i R_Y\left(x_i\pi/2\right)|0_i\rangle$$

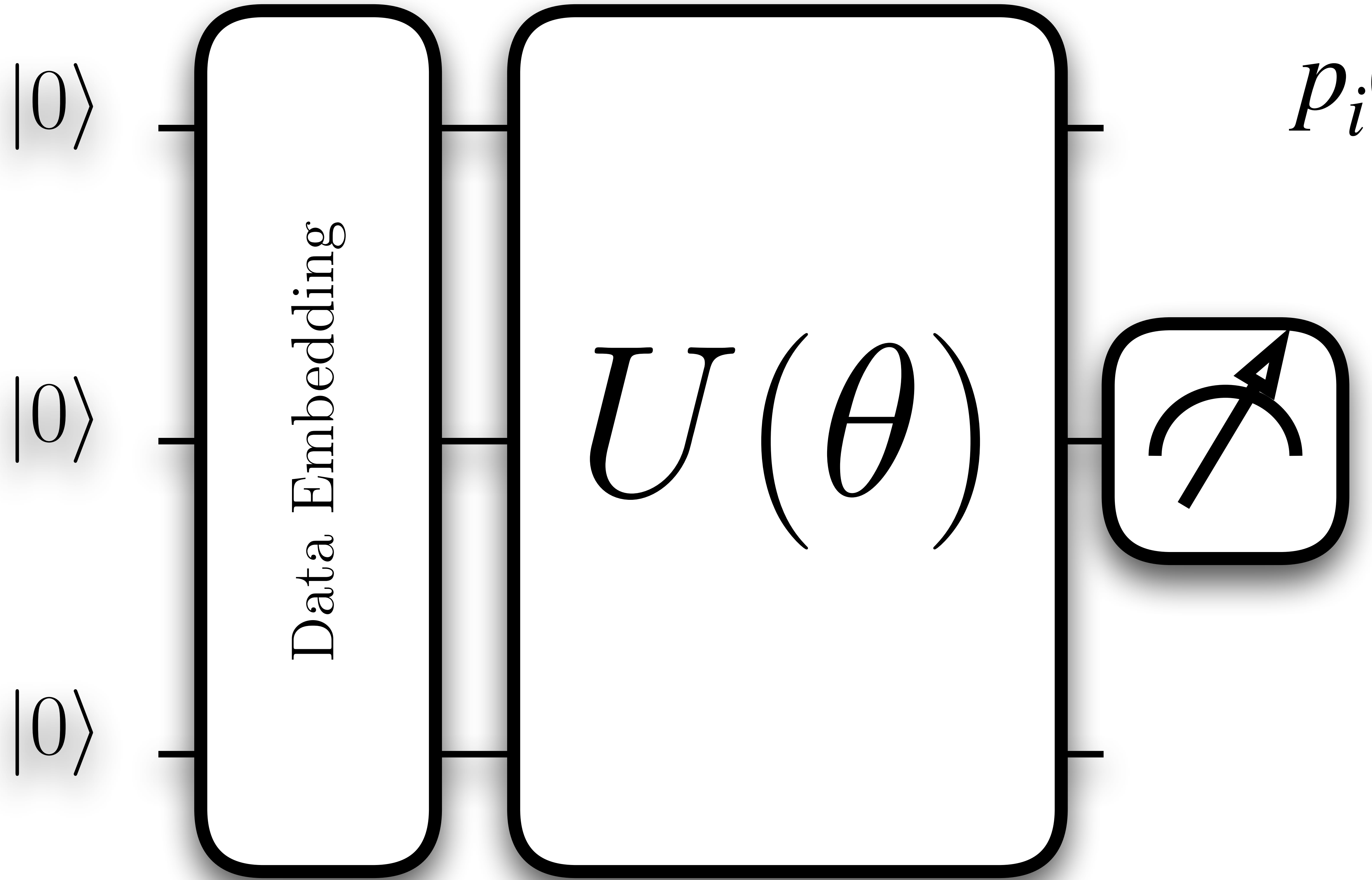


Amplitude Encoding

Normalised vector $|X\rangle = \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{pmatrix}$



Quantum Machine Learning for Data Analysis



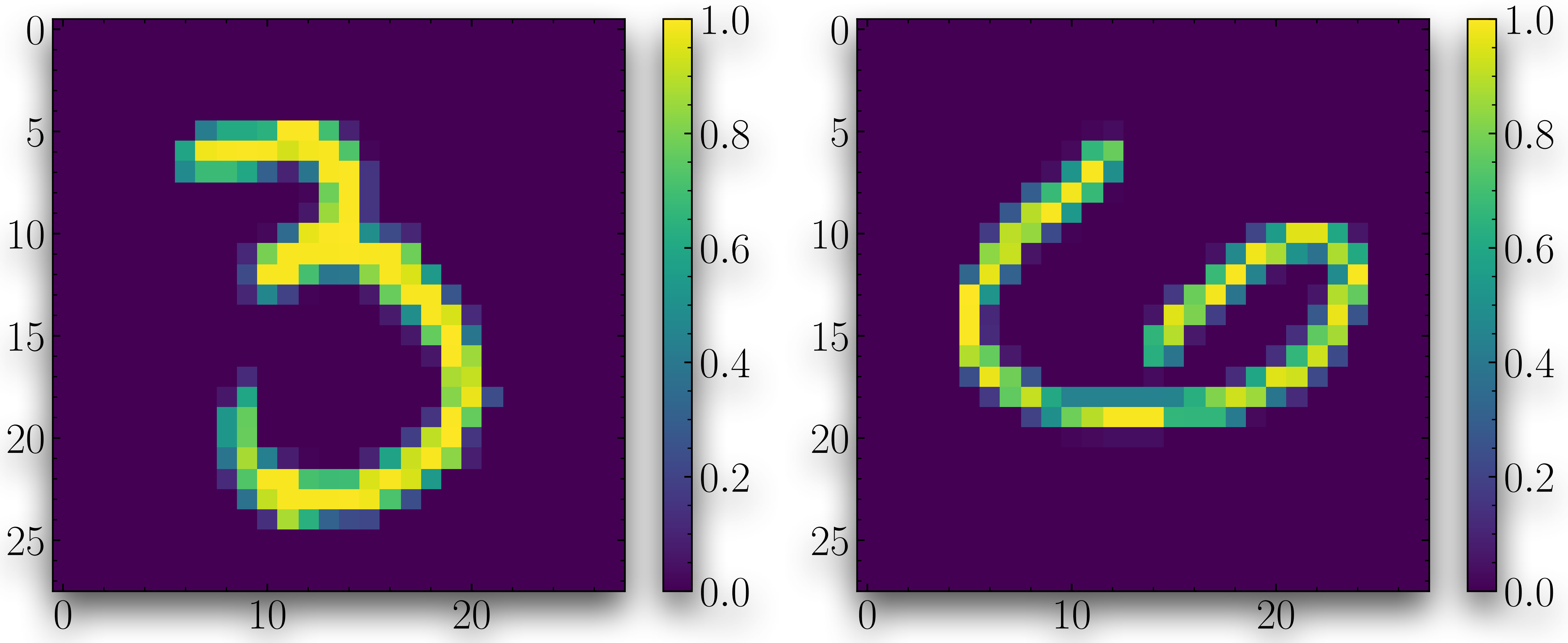
$$p_i(\theta) = \left| \langle 0 | \mathscr{P}^\dagger(x_i) U^\dagger(\theta)\, Z\, U(\theta) \mathscr{P}(x_i) | 0 \rangle \right|^2$$

Ex: Classification

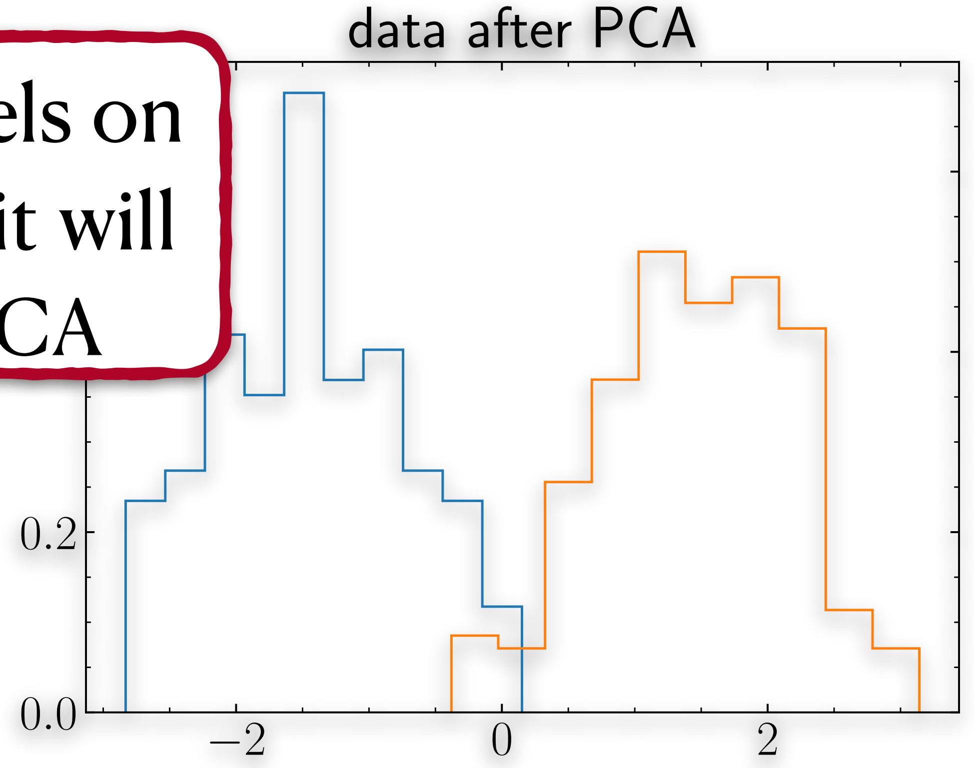
$$\arg\min_\theta \left(\mathscr{L}(\theta) = \frac{1}{N} \sum_i^N q^{\text{truth}} \log p_i(\theta) \right)$$

- Notice that there is no reason to just use Z the operator here.
- There is no clear convention for choosing an operator for ML purposes.
- Why not multiple observations? See next section

Ex III: QML with MNIST Dataset

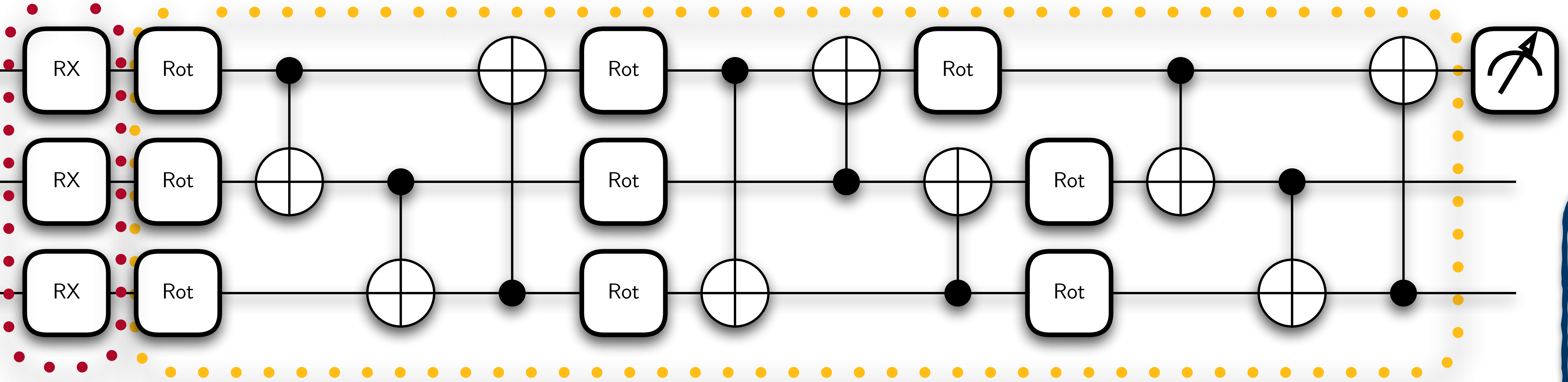


We can not work with 28×28 pixels on a quantum circuit (try if you wish; it will burn your RAM) so we will use PCA



Hinge Loss

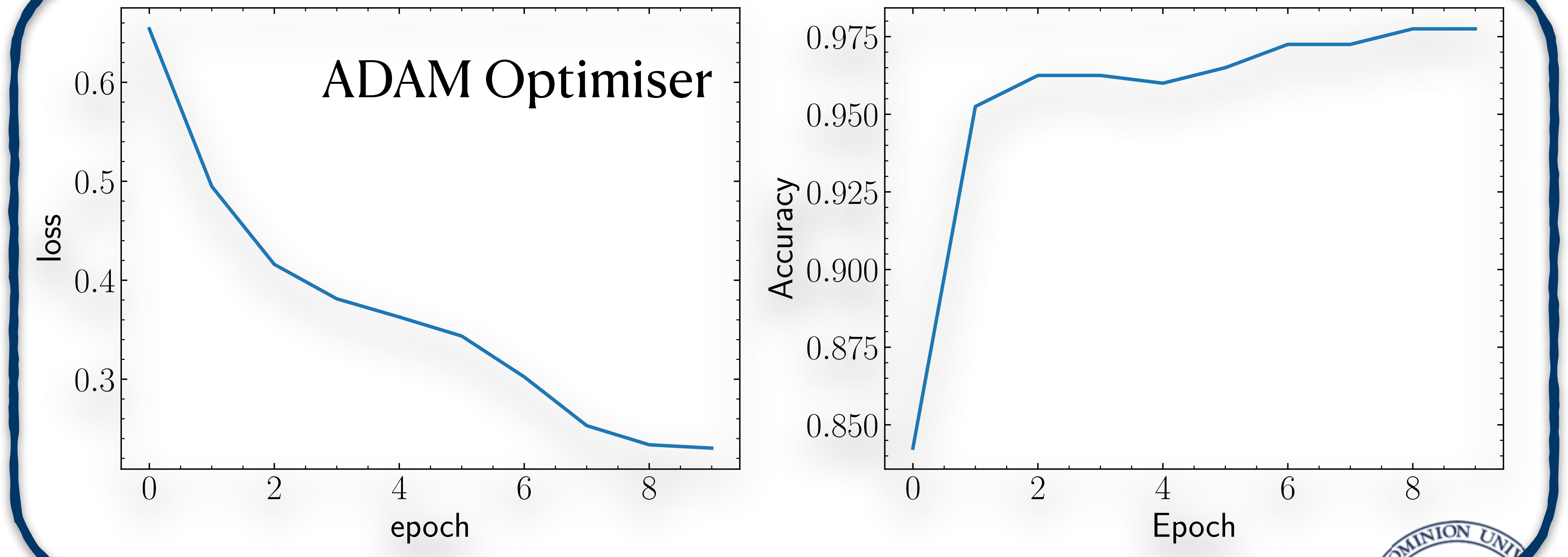
$$\mathscr{L}(\theta) = \max\left(1 - y^{\mathrm{true}} y^{\mathrm{reco}},\ 0\right)$$



Angle Embedding

Ansatz

$$\langle 0 | U^{\dagger}(\theta)\ Z_0\ U(\theta) | 0 \rangle = y^{\mathrm{reco}}$$



Sharp bits

Sharp bits

- Which qubit to measure?
- Barren plateaus!
  - Number of qubits
  - Number of layers
- Which observables to use?
- Which ansatz to use?
- Real data is not "entangled"; it is "correlated"!
- QC are very limited!

Good read: arXiv 2309.09342

A short-term solution can be Tensor Networks!

Barren plateau, according to Dall-E

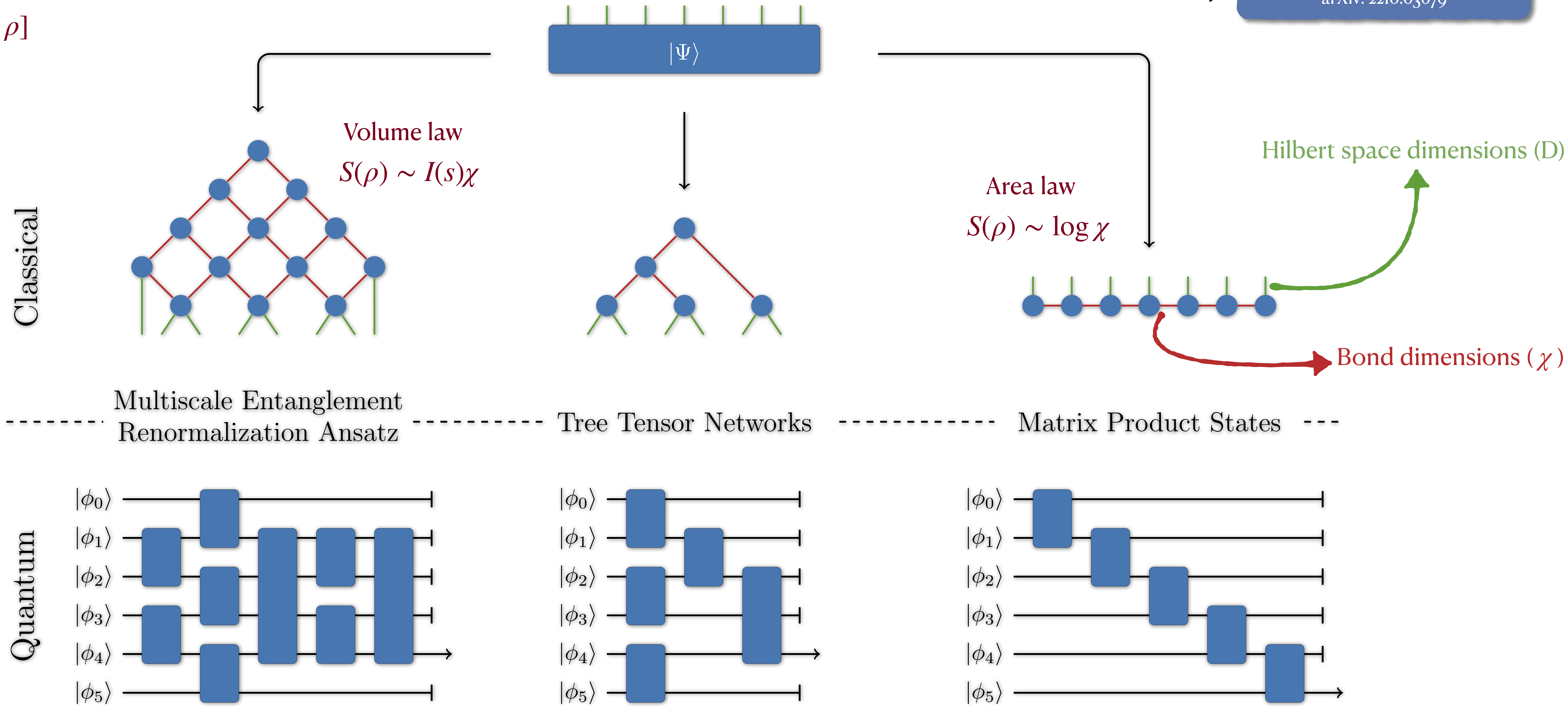
Briefly Tensor Networks

ML: JYA, Spannowsky; JHEP '21, arXiv: 2106.08334

JYA, Spannowsky; PRA '22, arXiv: 2202.10471

field theory: JYA, Schenk, Spannowsky; PRA '23, arXiv: 2210.03679

$S(\rho) = -\operatorname{Tr}[\rho \log \rho]$



Let's put everything together: Quantum-probabilistic Hamiltonian Learning for anomaly detection

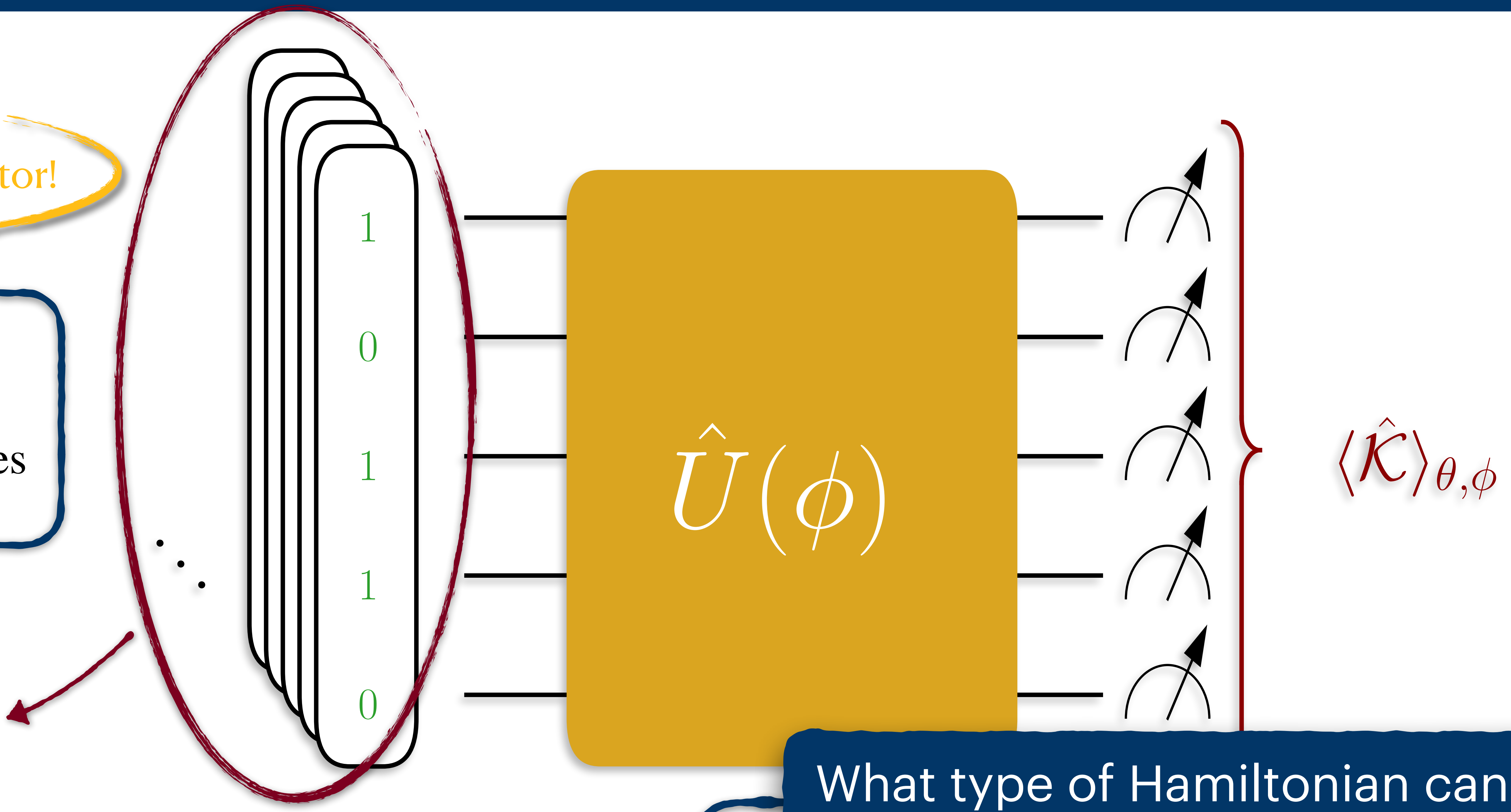
Quantum-probabilistic Hamiltonian Learning

JYA, Spannowsky; arXiv: 2211.03803 ; PRA

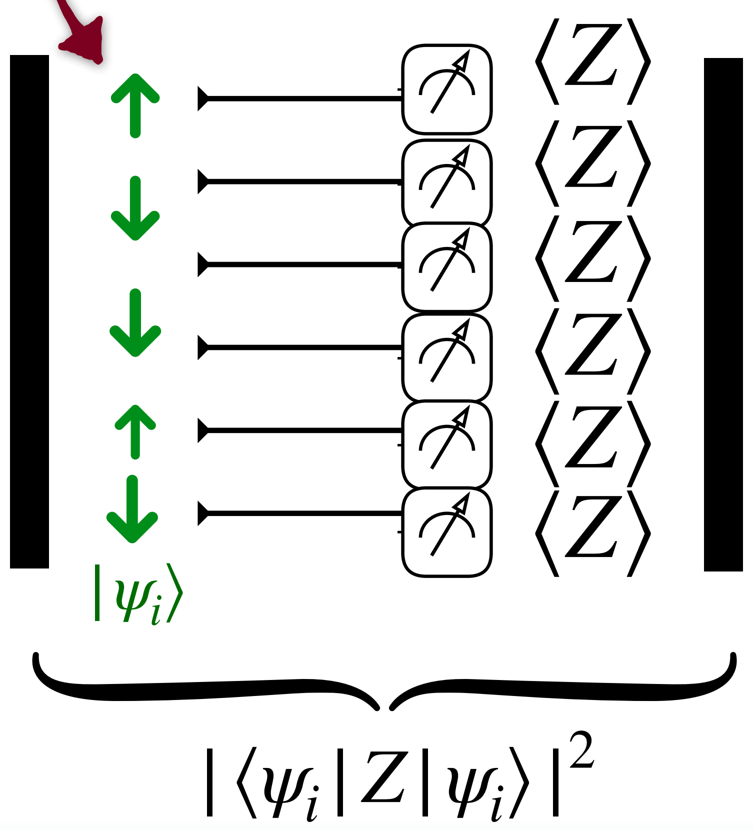
Quantum Circuit is a pure-state simulator!

A data point can be represented as a mixed state

$$\sigma_D = \sum_i p_i |\psi_i\rangle \ , \ |\psi_i\rangle := \text{pure states}$$



Hamiltonian captures the entropic probability distribution of the mixed state.



$$\frac{1}{N_{\rm sample}} \sum_{\psi_i \in \Psi}^{N_{\rm sample}} \Big| \ \cdots \ \Big|^2 \simeq \mathrm{P(Data)}$$

What type of Hamiltonian can we choose?

- Any field theory Hamiltonian, e.g. Ising model
- A generic Hamiltonian, e.g. $\sum_{\langle i,j\rangle} (\alpha_{i,j}\sigma_i^+\sigma_j^- + \mathrm{h.c.})$
- But can these options capture the full complexity of the data? Can we get ambitious?

What has Hamiltonian to do with data?

JYA, Spannowsky; arXiv: 2211.03803 ; PRA

Quantum Circuit is a pure-state simulator!

A data point can be represented as a mixed state

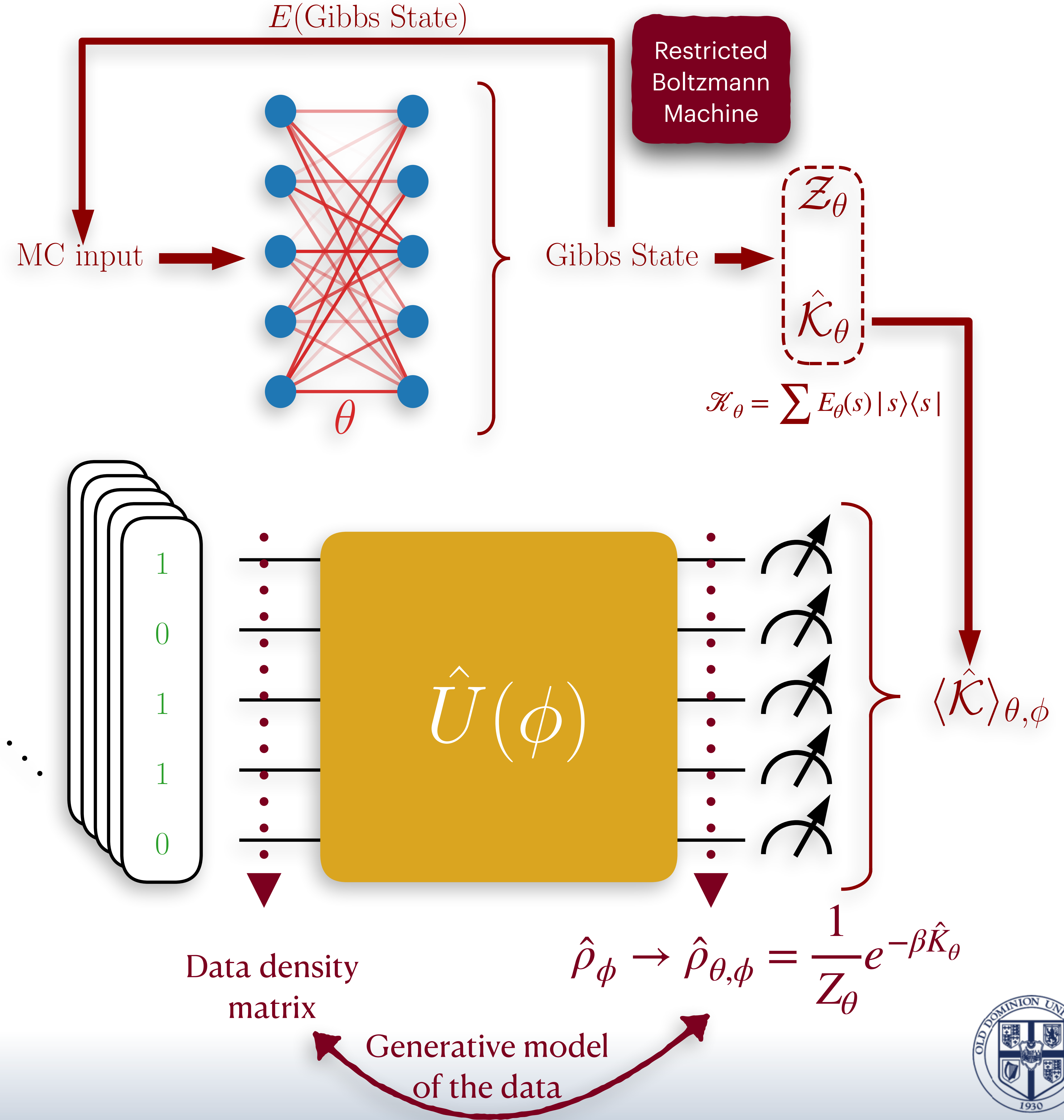
$$\sigma_D = \sum_i p_i |\psi_i\rangle \ , \ |\psi_i\rangle := \text{pure states}$$

See Gibbs-Delbrück-Moliére variational principle

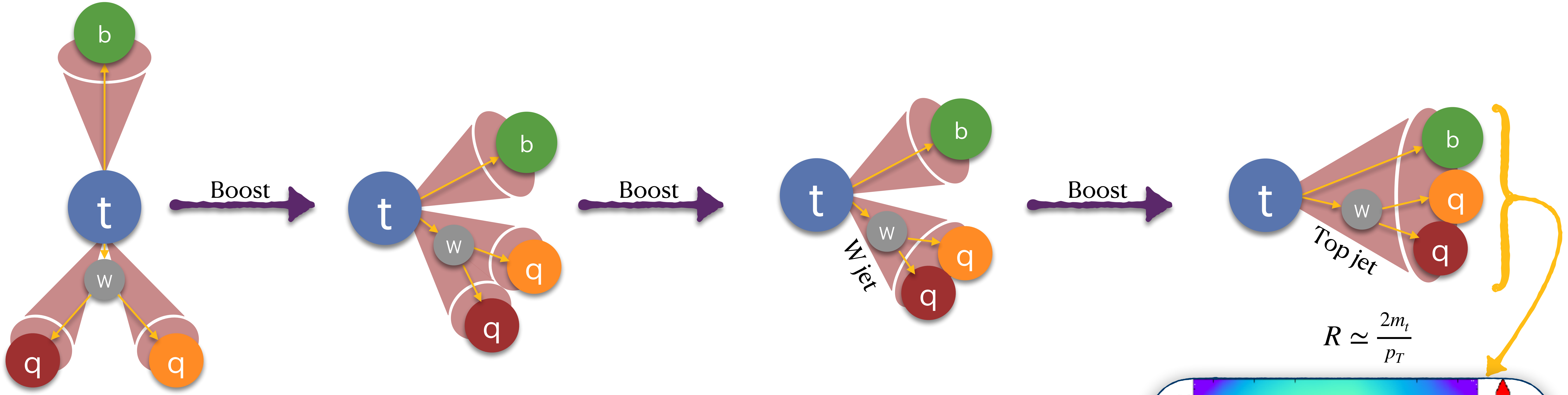
$$F = E - TS = -k_\beta T \ln Z_\theta$$

$$\mathscr{L}_{\theta,\phi}(\sigma_D) = \beta\langle\hat{K}\rangle_{\theta,\phi} + k_\beta \ln Z_\theta \ \geq \ S(\sigma_D)$$

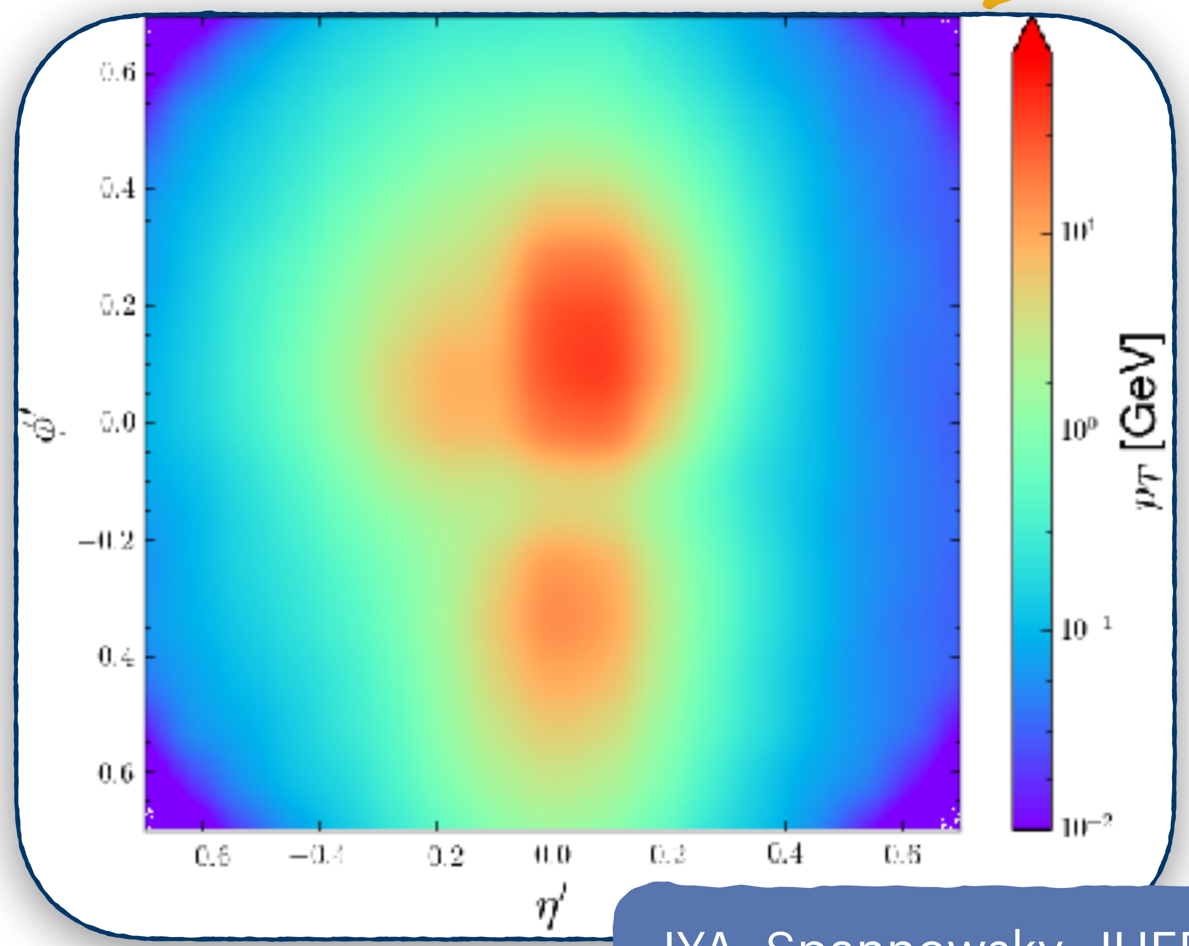
k_β := Boltzmann constant
β := Inverse temperature



Hello world of HEP-ML: Top tagging



$$R \simeq \frac{2m_t}{p_T}$$

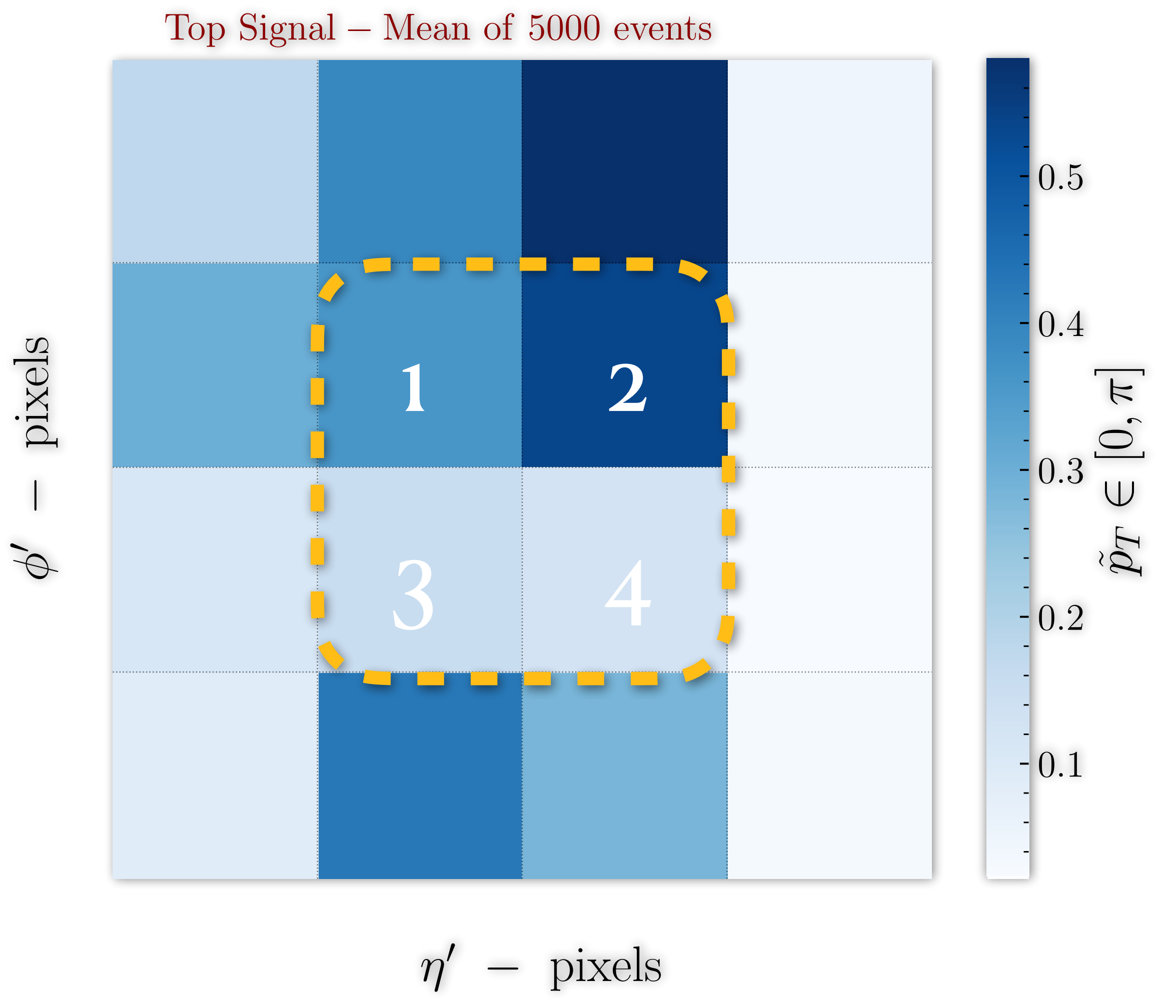


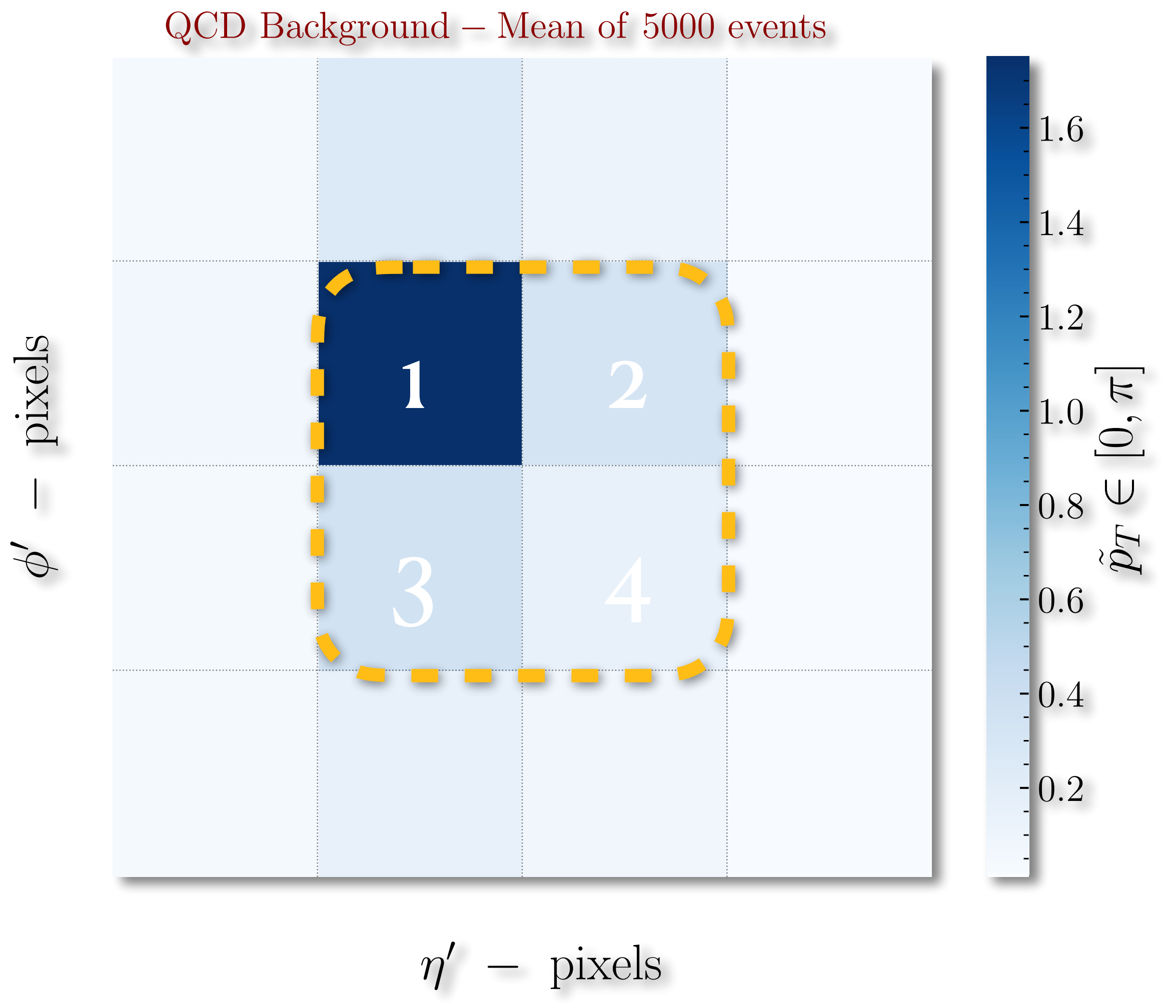
JYA, Spannowsky; JHEP '21

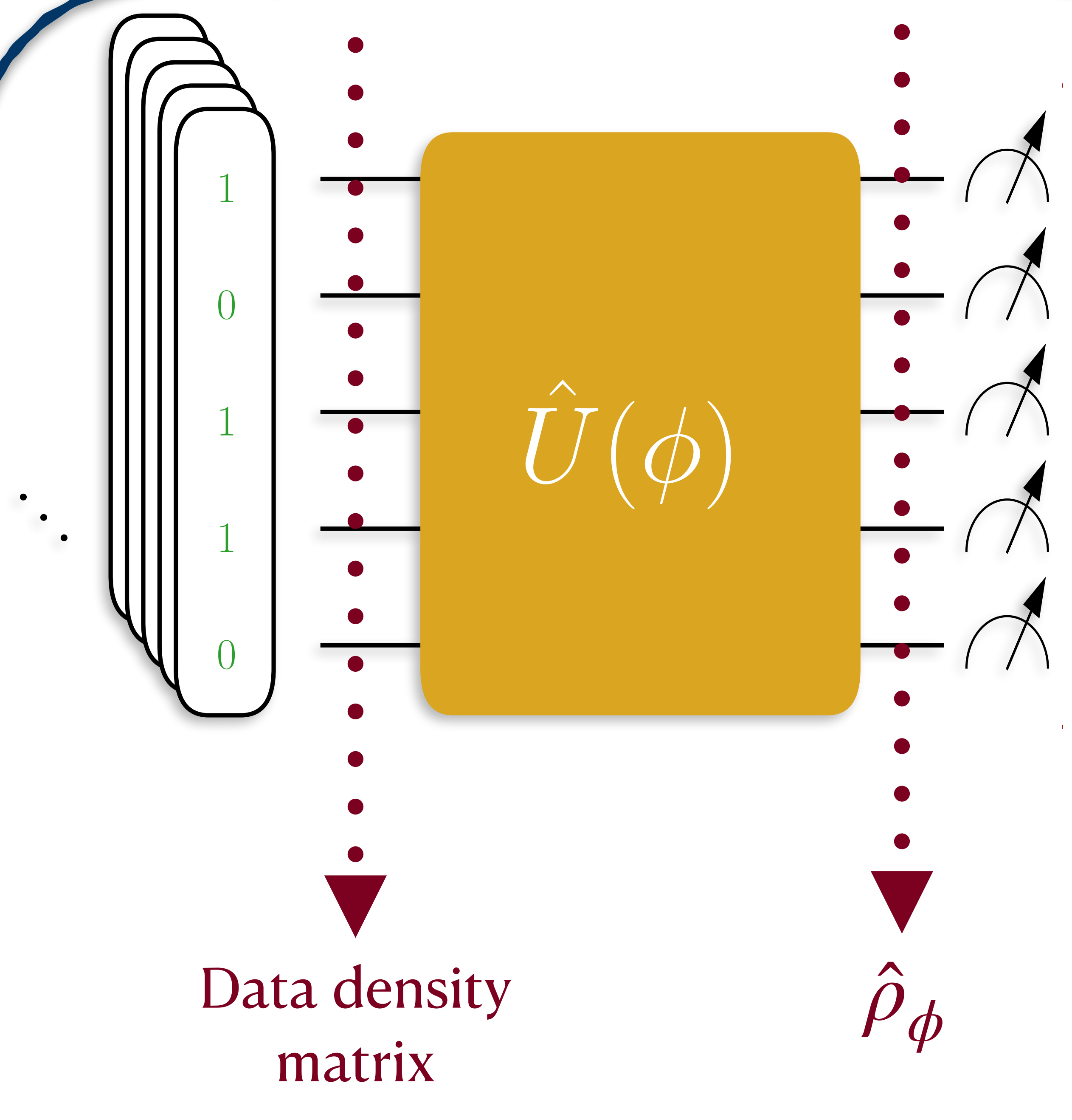
- With the increased boost factor, jets (top decay products) are getting more collimated.
- Hadronic top tagging tools: Mass grooming and filtering, Pruning, Trimming, Soft Drop Tagger, Mass Drop Tagger, HEPTopTagger, Machine Learning

What has Hamiltonian to do with data?

JYA, Spannowsky; arXiv: 2211.03803 ; PRA

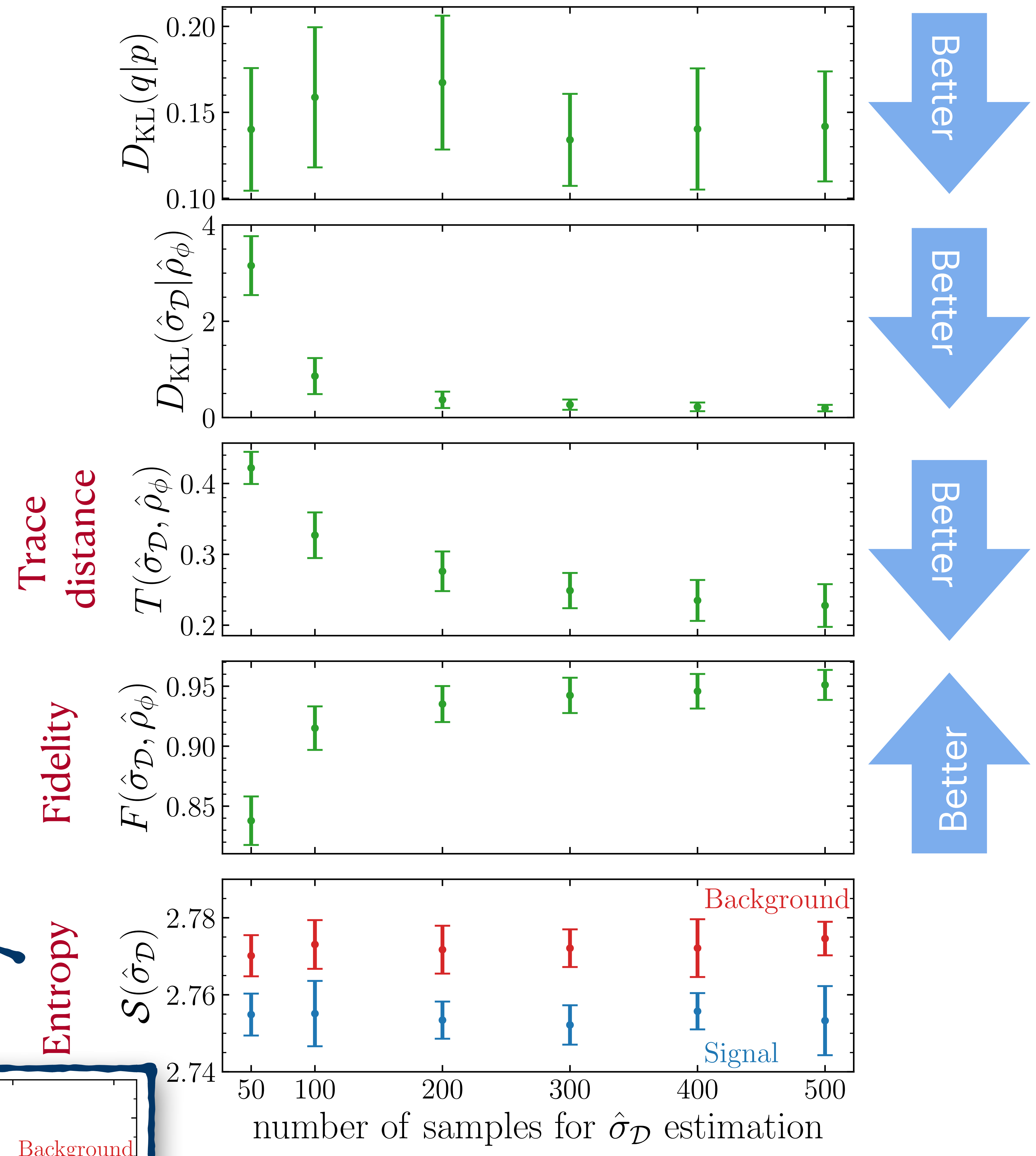


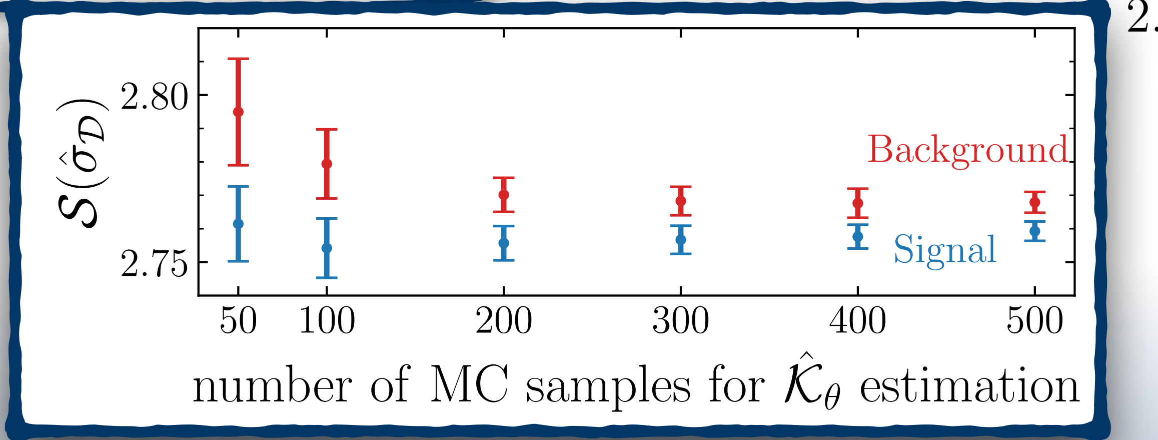




$$\mathscr{L}_{\theta,\phi}(\sigma_D) = \langle \hat{K} \rangle_{\theta,\phi} + \ln Z_\theta \ \geq \ S(\sigma_D)$$

$$\arg\min_{\theta,\phi} \mathscr{L}_{\theta,\phi}(\sigma_D) \simeq S(\sigma_D)$$



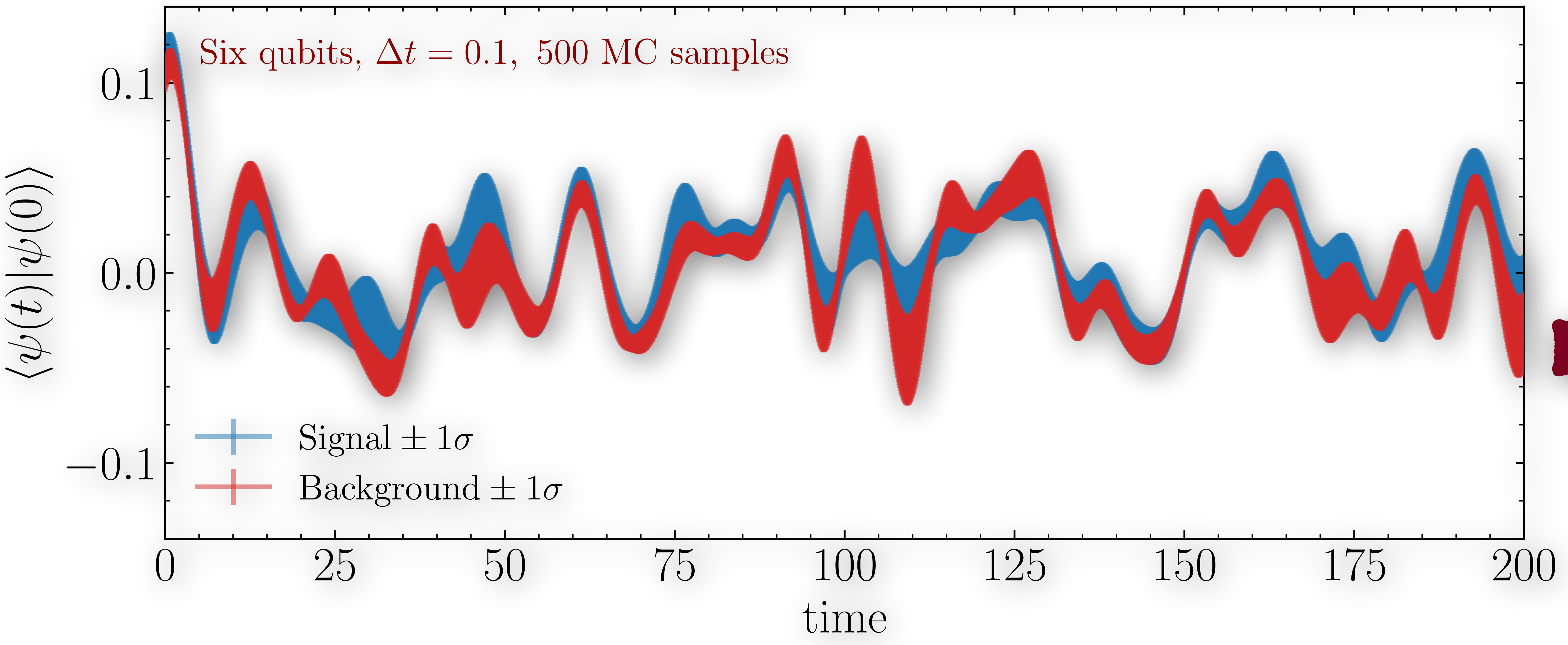


Jack Y. Araz

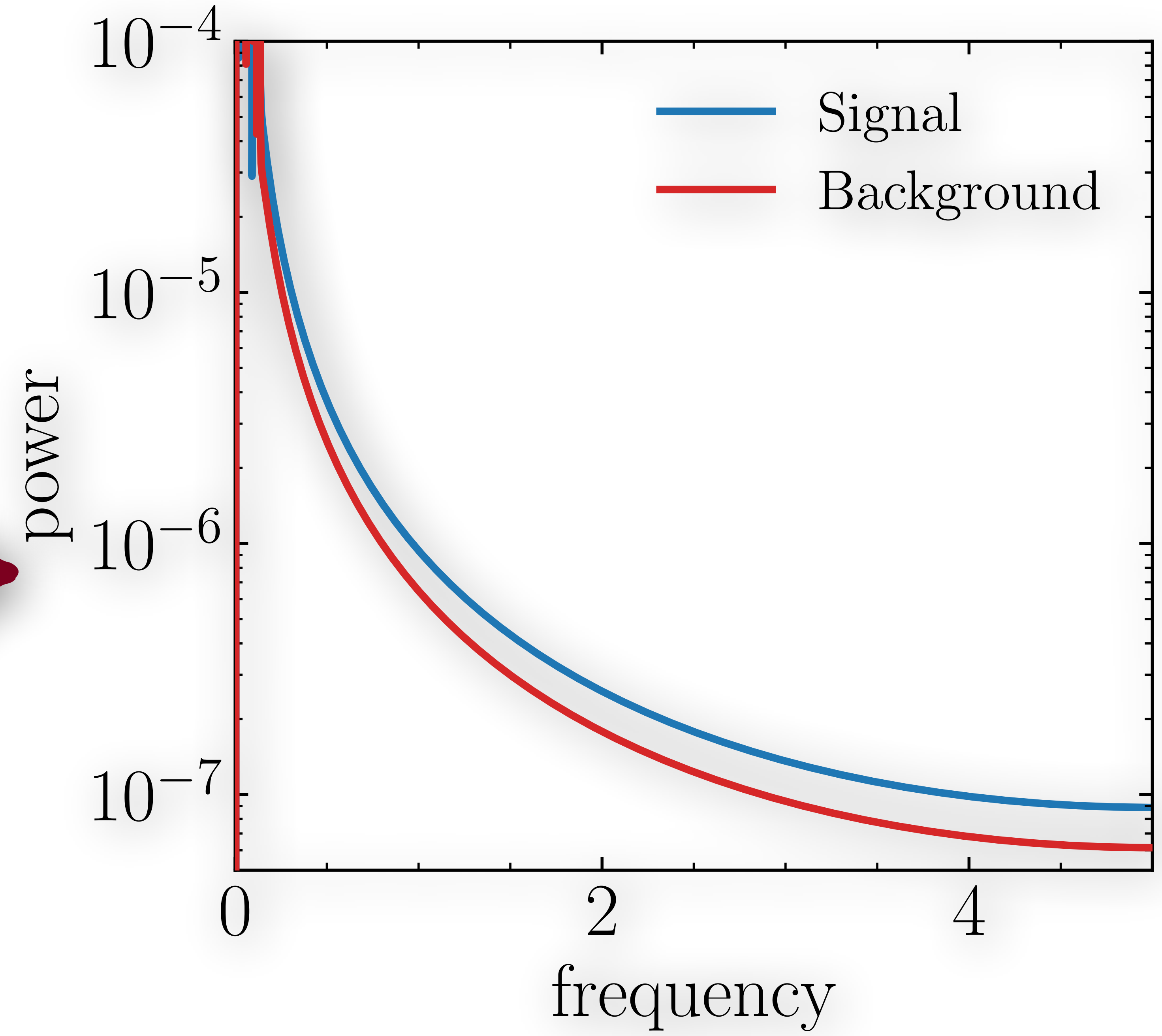
Hamiltonian as a discriminator!

JYA, Spannowsky; arXiv: 2211.03803 ; PRA

Trotter-Suzuki approximation → $e^{-iT\hat{K}_\theta} = \prod^{N} e^{-i\Delta t\hat{K}_\theta}$



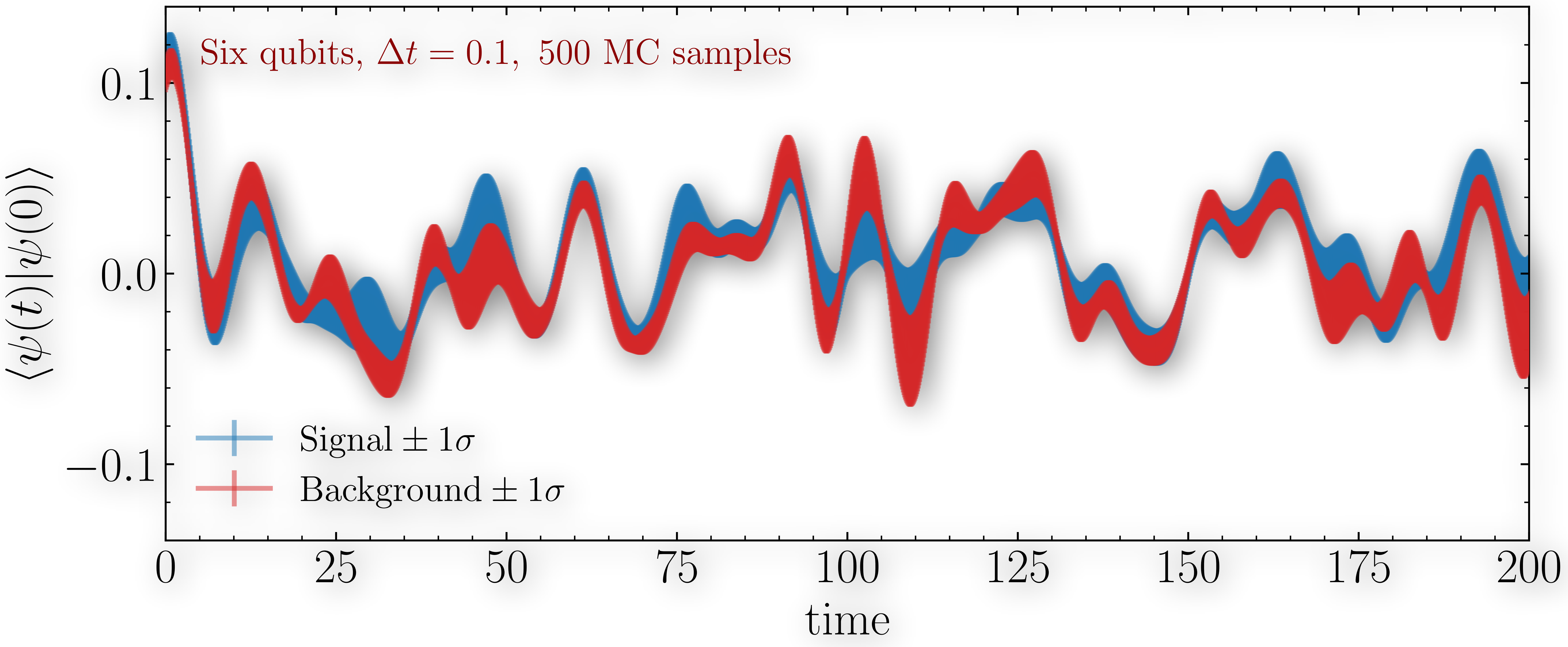
Fast Fourier Transform

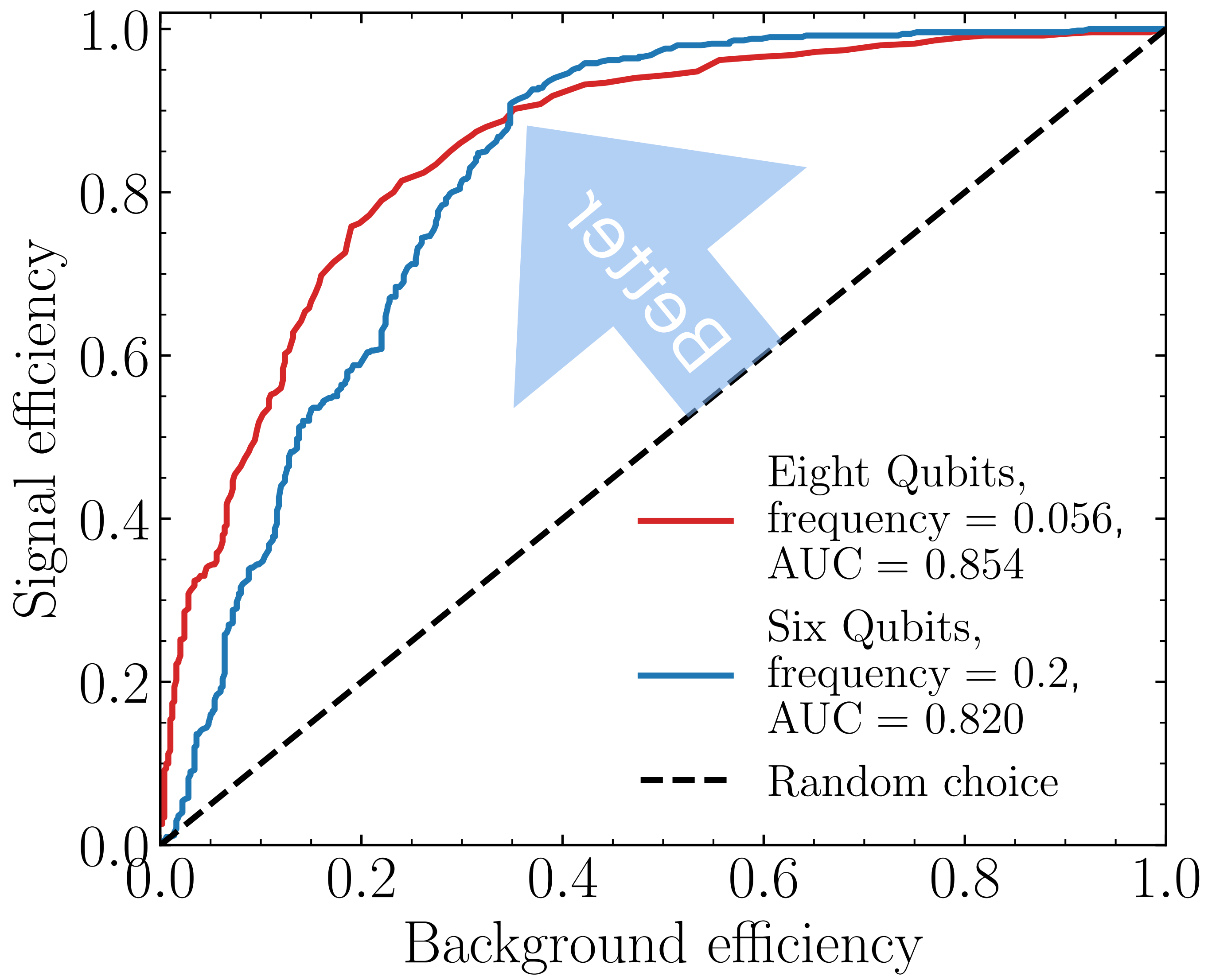


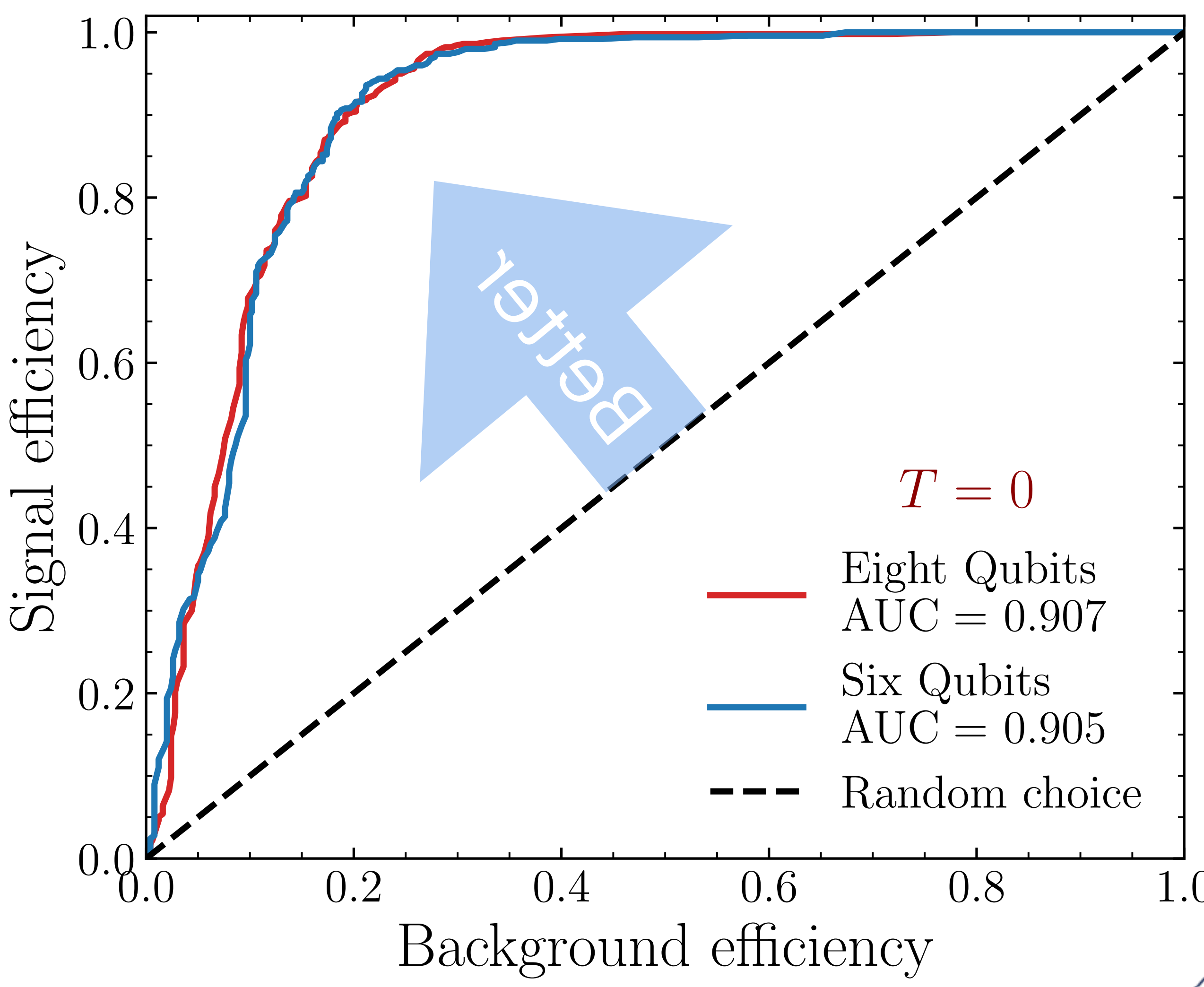
Hamiltonian as a discriminator!

JYA, Spannowsky; arXiv: 2211.03803 ; PRA

Trotter-Suzuki approximation → $e^{-iT\hat{K}_\theta} = \prod^{N} e^{-i\Delta t\hat{K}_\theta}$

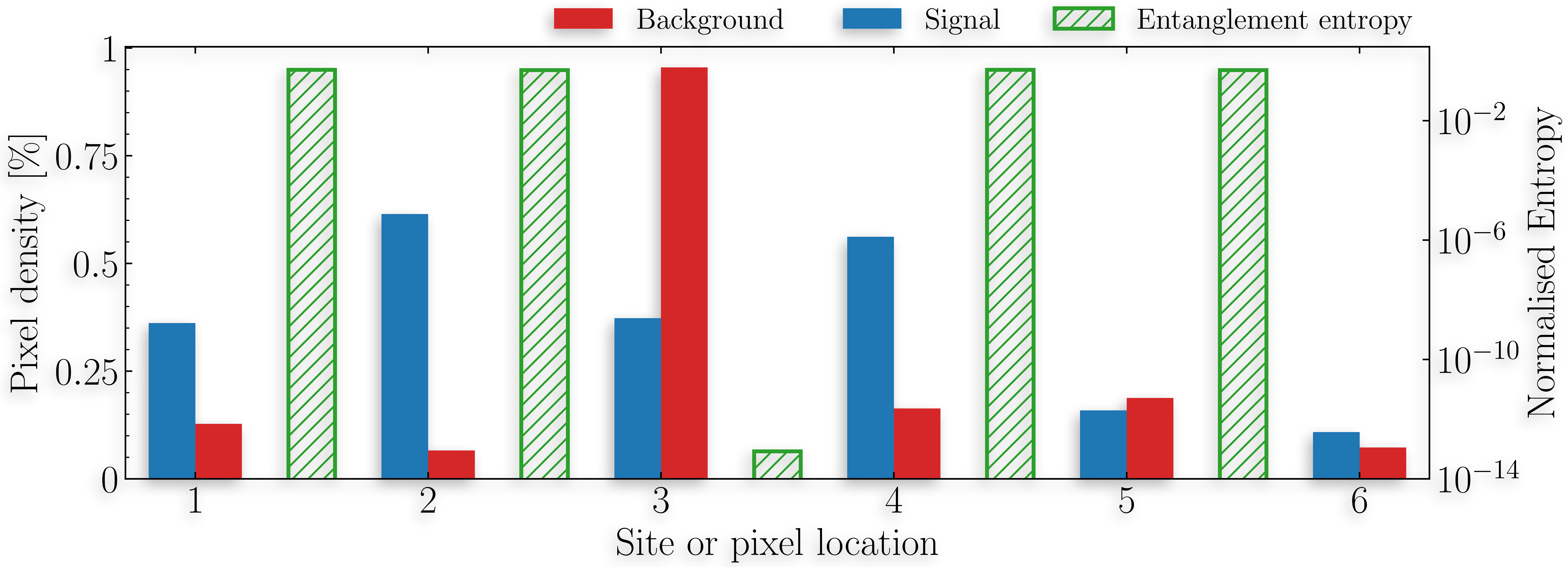






What did the Hamiltonian learn?

JYA, Spannowsky; arXiv: 2211.03803 ; PRA



$$\mathcal{S}(\rho) = -\,\mathrm{Tr}[\rho \log \rho]$$

Conclusion

Conclusion

- There is a wide range of applications for QML, from field theory to data analysis. VQC is significantly more capable compared to classical MC or TN methods.
- Quantum theory is extremely rich and worth exploring the applications designed for theory in data analysis.
- There are significant limitations, i.e. barren plateaus.
- Quantum advantage in QML????????
  - One obvious advantage: Time evolution