QUANTUM TECHNOLOGY INITIATIVE

Quantum Machine Learning and Optimisation

Carla Rieger (CERN, TU Munich)

CERN Openlab Summer Student Lectures, 30.7.2024

Recap: the key features of Quantum Computing

$$\frac{|0\rangle + |1\rangle}{\sqrt{2}}$$

Quantum Superposition State

$$\frac{|00\rangle + |11\rangle}{\sqrt{2}}$$

Quantum Entanglement
(here: Bell state)

Recap: the key features of Quantum Computing

$$\frac{|0\rangle + |1\rangle}{\sqrt{2}}$$

Quantum Superposition State

Can enable speed-up though **highly parallel** computations

$$\frac{|00\rangle + |11\rangle}{\sqrt{2}}$$

Quantum Entanglement
(here: Bell state)

Also, **non-classical correlations** may speed-up computations

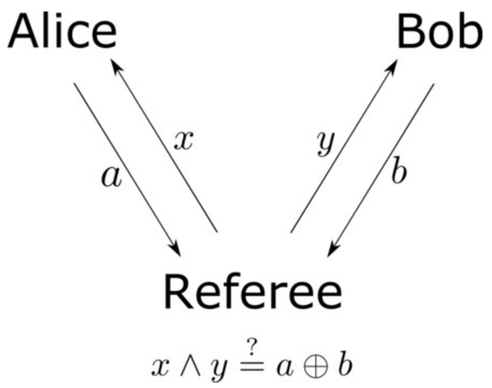
Interlude: Bell at CERN

John Stewart Bell commenting on the famous Bell's inequalities at CERN in 1982.

Source: https://physicsworld.com/a/saved-by-bell/

Interlude: Bell inequality

CHSH-Game



1. Alice and Bob may agree on a strategy before the game starts but cannot communicate once the game has started. They act cooperatively.

2. The referee prepares (binary) bits x and y independently and at rand . om

3. Alice and Bob win if their return answers $a \in \{0,1\}$ and $b \in \{0,1\}$ satisfy: $xy = a \bigoplus b$.

John Stewart Bell commenting on the famous Bell's inequalities at CERN in 1982.

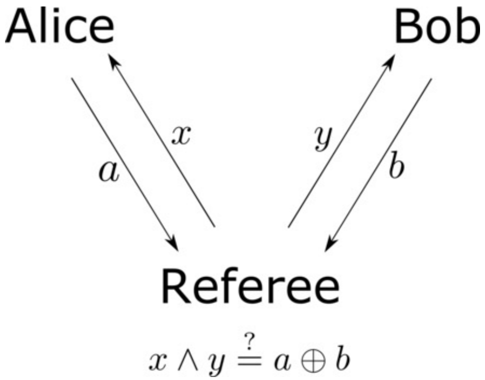
Image source: https://physicsworld.com/a/saved-by-bell/ and Wikipedia.

Scarani, Valerio. *Bell nonlocality*. Oxford University Press, 2019.

Interlude: Bell inequality

John Stewart Bell commenting on the famous Bell's inequalities at CERN in 1982.

CHSH-Game



1. Alice and Bob may agree on a strategy before the game starts but cannot communicate once the game has started. They act cooperatively.
2. The referee prepares (binary) bits x and y independently and at rand . om
3. Alice and Bob win if their return answers $a \in \{0,1\}$ and $b \in \{0,1\}$ satisfy: $xy = a \oplus b$.

→ **Upper bound on winning probability:**

Bell's inequality

$$\underbrace{3/4}_{\text{Local hidden-variable theory}}$$

Violation of Bell's inequality

$$\underbrace{\frac{2+\sqrt{2}}{4} \approx 0.85}_{\text{Shared entanglement and no communication}}$$

Image source: https://physicsworld.com/a/saved-by-bell/ and Wikipedia.

Scarani, Valerio. *Bell nonlocality*. Oxford University Press, 2019.

Interlude: Bell inequality

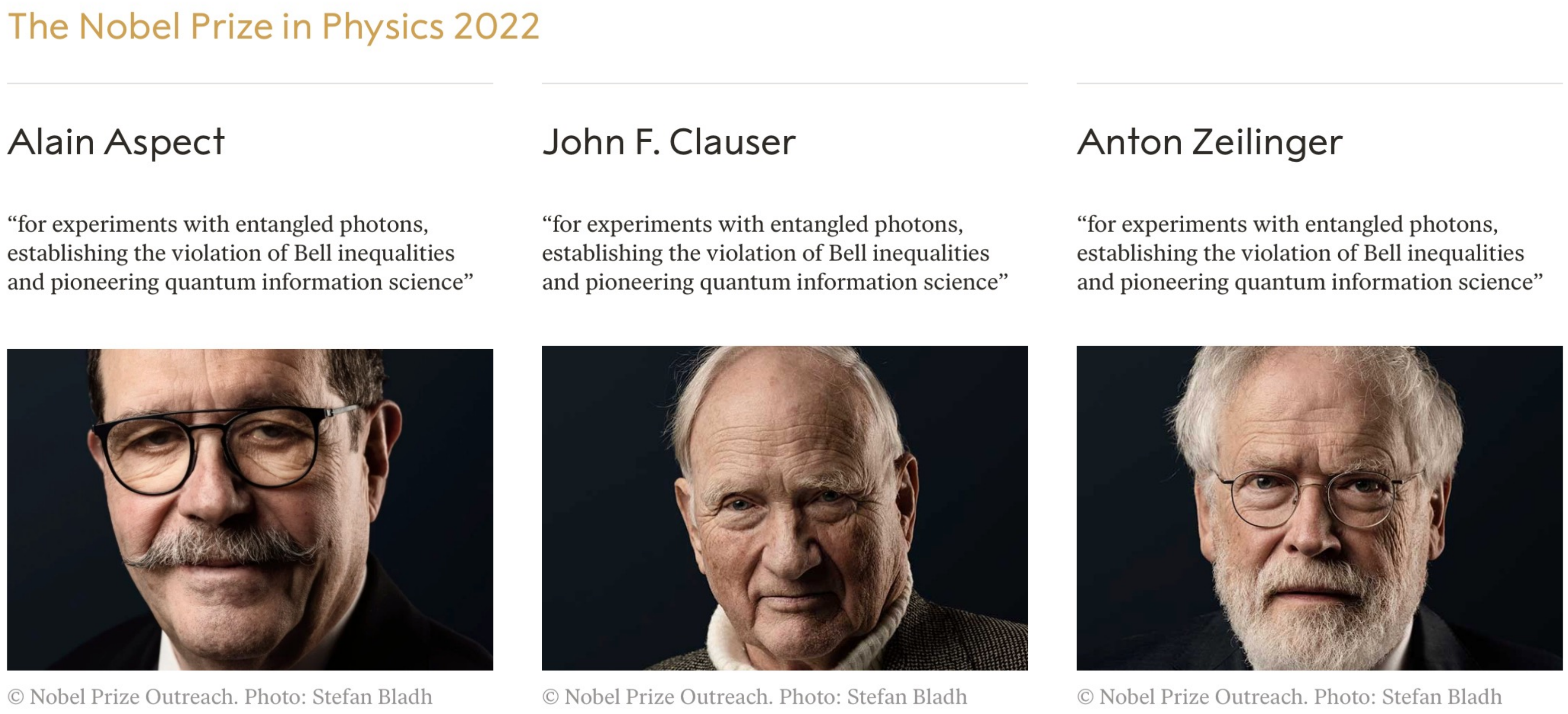
John Stewart Bell commenting on the famous Bell's inequalities at CERN in 1982.

This implies that the ***predictions of quantum theory cannot be accounted for by any local theory.***

Image source: https://physicsworld.com/a/saved-by-bell/

Scarani, Valerio. *Bell nonlocality*. Oxford University Press, 2019.

Interlude: Bell inequality



The Nobel Prize in Physics 2022

Alain Aspect

"for experiments with entangled photons, establishing the violation of Bell inequalities and pioneering quantum information science"

© Nobel Prize Outreach. Photo: Stefan Bladh

John F. Clauser

"for experiments with entangled photons, establishing the violation of Bell inequalities and pioneering quantum information science"

© Nobel Prize Outreach. Photo: Stefan Bladh

Anton Zeilinger

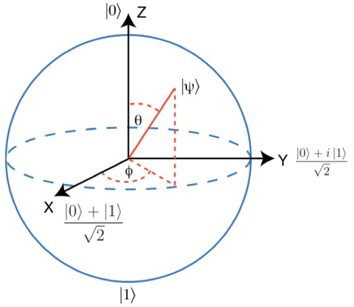
"for experiments with entangled photons, establishing the violation of Bell inequalities and pioneering quantum information science"

© Nobel Prize Outreach. Photo: Stefan Bladh

John Stewart Bell commenting on the famous Bell's inequalities at CERN in 1982.

Image sources: https://www.nobelprize.org/all-nobel-prizes-2022/ and https://physicsworld.com/a/saved-by-bell/

Recap: Basic one qubit gates



Quantum Theory is unitary — gates are represented by unitary matrices U →

$$U^\dagger U = \mathbb{1}$$

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

Bit flip

$$\sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

Phase flip

$$\sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$$

Bit + phase flip

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

Hadamard gate

- *Pauli matrices (together with identity matrix) form basis of 2x2 matrices*
- *any 1-qubit rotation can be written as a linear combination of Pauli gates*

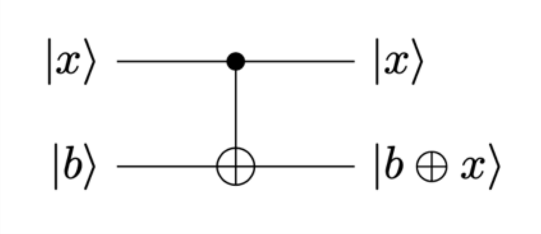
$$H|x\rangle = \frac{1}{\sqrt{2}}(|0\rangle + (-1)^x|1\rangle), \; x \in \{0,1\}$$

Apply H on computational basis state

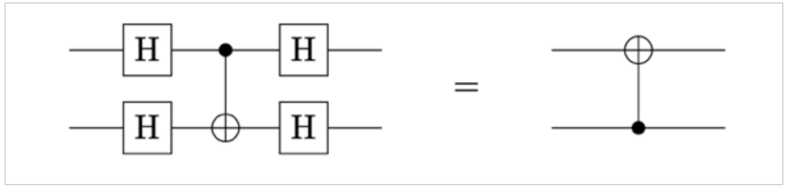
Nielsen, Michael A., and Isaac L. Chuang. *Quantum computation and quantum information*. Cambridge university press, 2010.

Recap: Basic two qubit gate

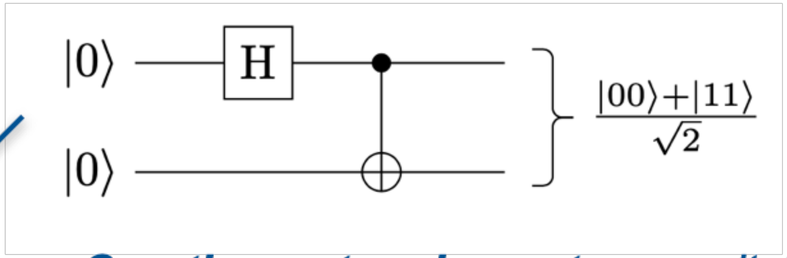
- since Quantum Theory is *unitary*, gates must be *reversible*
- *CNOT gate:* reversible XOR gate



CNOT gate

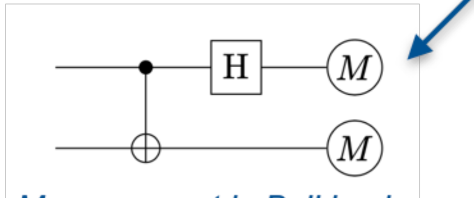


Switched CNOT gate



Creating entanglement *—> switch computational to Bell basis*

$$\begin{cases} |\Psi^-\rangle = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle) \\ |\Psi^+\rangle = \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle) \\ |\Phi^-\rangle = \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle) \\ |\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \end{cases}$$



Measurement in Bell basis

Nielsen, Michael A., and Isaac L. Chuang. *Quantum computation and quantum information*. Cambridge university press, 2010.

Aim of Quantum Computing

→ Do **classically intractable** computations **efficiently** on a Quantum Computer leveraging Quantum Effects

Applications of Quantum Computing

One may successfully leverage quantum effects for:

- Efficient sampling, search and optimization (e.g., Grover's search algorithm)
- Linear algebra, matrix computations and machine learning (e.g., HHL-algorithm)
- Algorithms and protocols for Cryptography and Communication (e.g., Shor's algorithm, Quantum Key distribution)

Based on previous year's talk

What is Quantum Machine Learning?

Fields in Quantum Computing

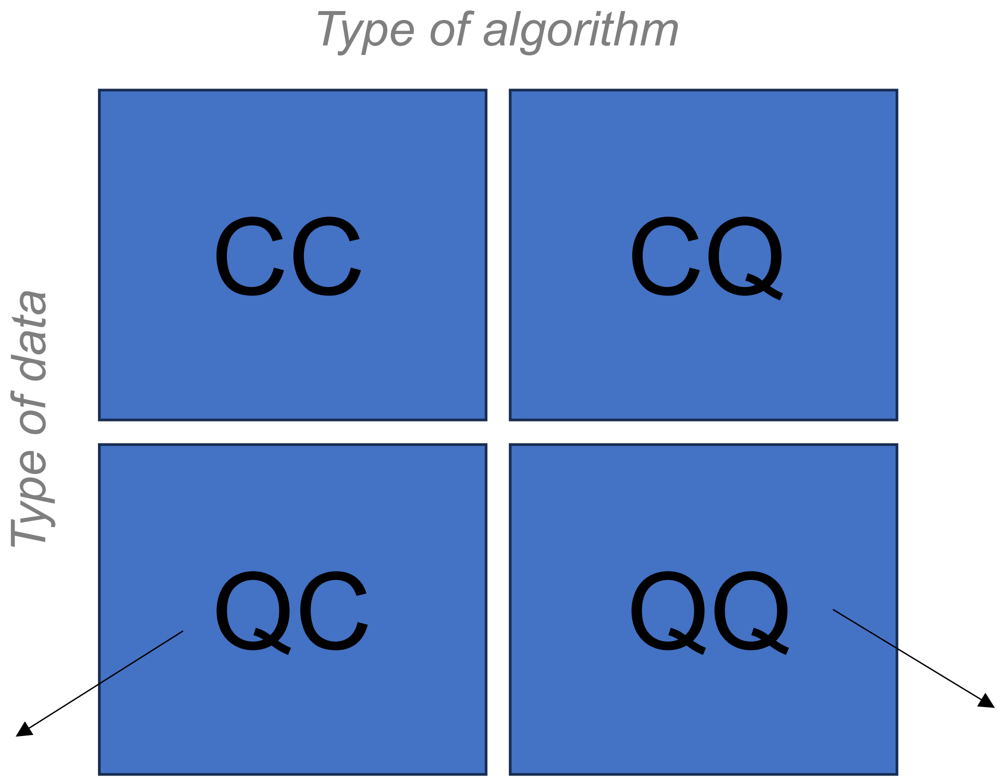
Type of algorithm

Type of data

CC	CQ
QC	QQ

Source: Qiskit Textbook

Fields in Quantum Computing

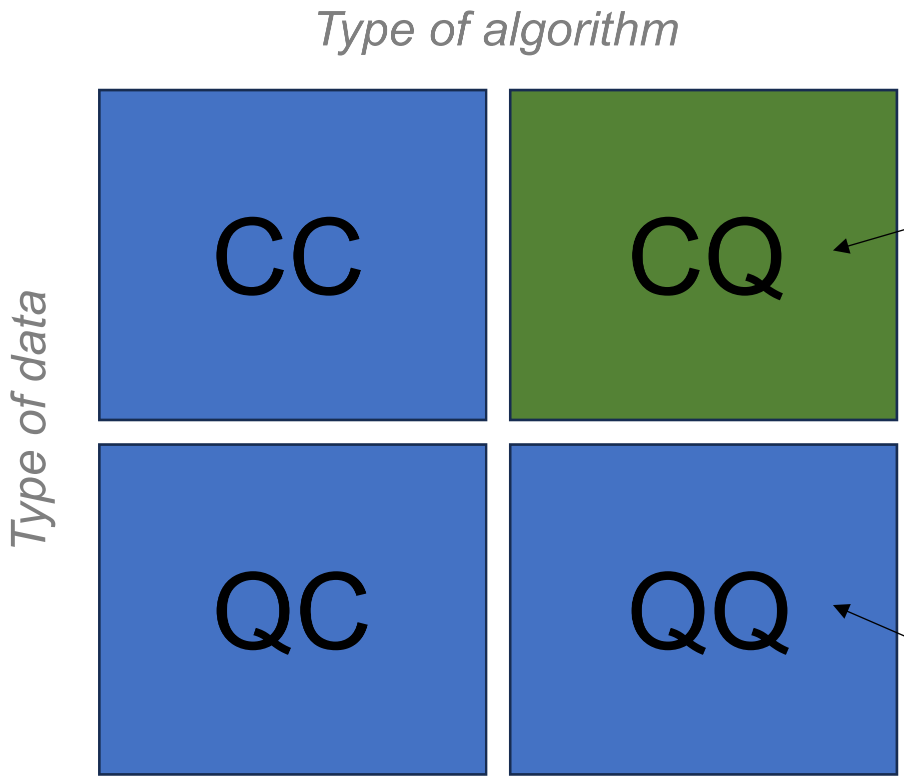


Simulation of Quantum Systems using classical ML

Simulation of Quantum Systems using a Quantum Computer

Source: Qiskit Textbook

Fields in Quantum Computing

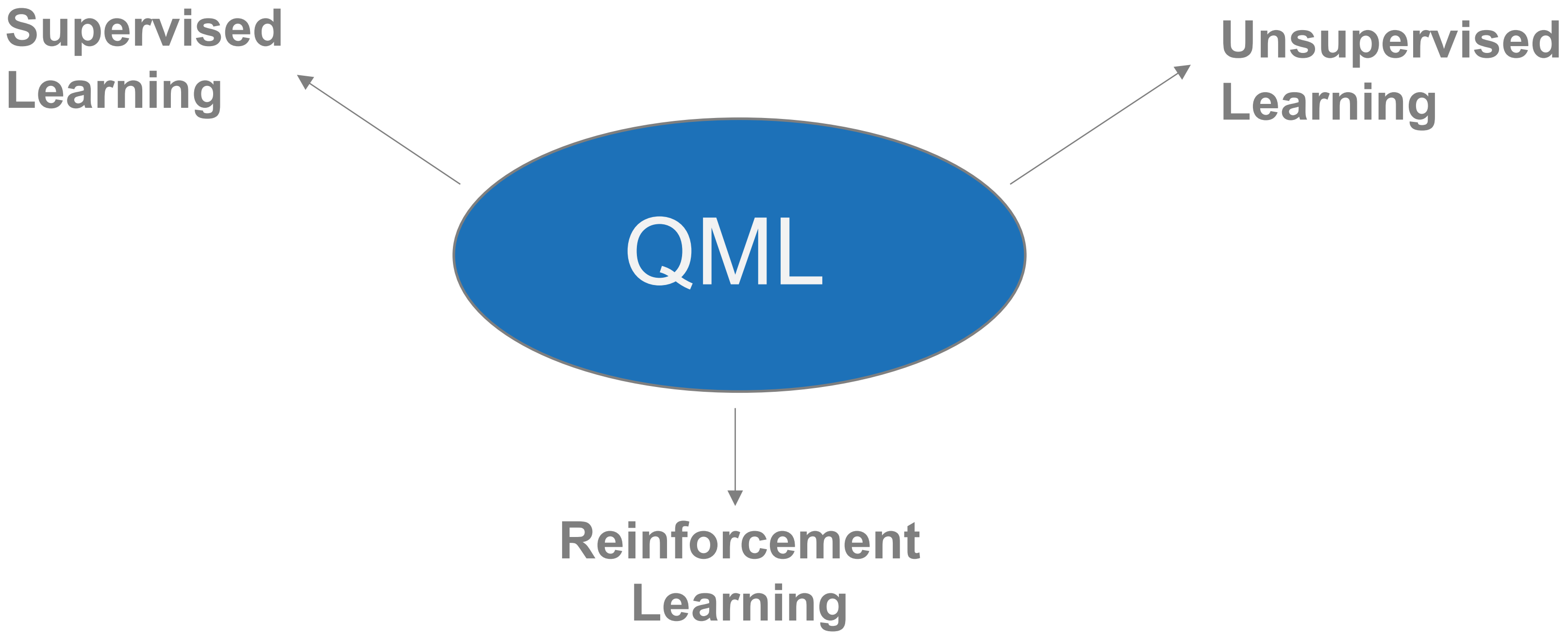


This talk mainly focuses on quantum algorithms with classical input data to foster a simple understanding

But: expect more advantages here

Source: Qiskit Textbook

Fields in Quantum Machine Learning (QML)



QML

Supervised Learning

Unsupervised Learning

Reinforcement Learning

Source: Qiskit Textbook

Fields in Quantum Machine Learning (QML)

Find underlying structure in *labeled* data

Supervised Learning

QML

Unsupervised Learning

Find underlying structure in *unlabeled* data

Reinforcement Learning

Train agent to optimize its environment-based decisions

Source: Qiskit Textbook

Fields in Quantum Machine Learning (QML)

QML

Find underlying structure in *labeled* data

Supervised Learning

- Risk minimization task: minimize the discrepancy between the model's prediction and the target output;
- the best model has minimal expected loss over all data
- model should generalize well

Find underlying structure in *unlabeled* data

Unsupervised Learning

- No loss function based on labels but formulate distance between true probability distribution and the model distribution
- Model should generalize in order to produce samples from true probability distribution

Reinforcement Learning

- Train an agent to make decisions that maximize rewards
- Rewards are given through interaction with the environment

Train agent to optimize its environment-based decisions

Source: Qiskit Textbook

Fields in Quantum Machine Learning (QML)

QML

Find underlying structure in *labeled* data

Supervised Learning

e.g., **Quantum Classifier**: aims to learn *input-output relation* of labeled dataset $f: x_{in} \mapsto x_{out}$ by optimizing quantum network

Find underlying structure in *unlabeled* data

Unsupervised Learning

e.g., **Quantum Born machine**: aims to *learn to sample from underlying probability distribution* $\pi(y)$ of a given data set and *generates samples* via probabilistic measurement

Reinforcement Learning

e.g., **Quantum Reinforcement Learning**: find *policy* for agent that *maximizes reward* (expected reward computed using QC)

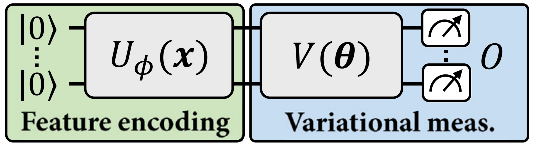
Train agent to optimize its environment-based decisions

Source: Qiskit Textbook

Further differentiating Quantum Machine Learning Models

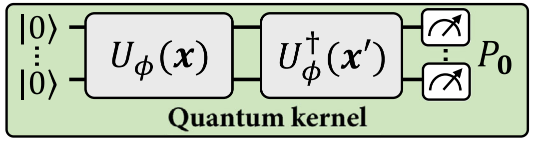
Variational algorithms (e.g., QNN)

- Utilize gradient-based or gradient-free optimization procedure
- May learn data embedding
- The design of the Ansatz circuit can leverage inherent symmetries of data



Kernel methods (e.g., QSVM)

- Based on similarity measures between data points
- Choose kernel function based on inherent data structure
- Quantum kernel functions correspond to feature maps
- Encode data in high-dimensional Hilbert space and use inner products evaluated in the feature space to model data properties



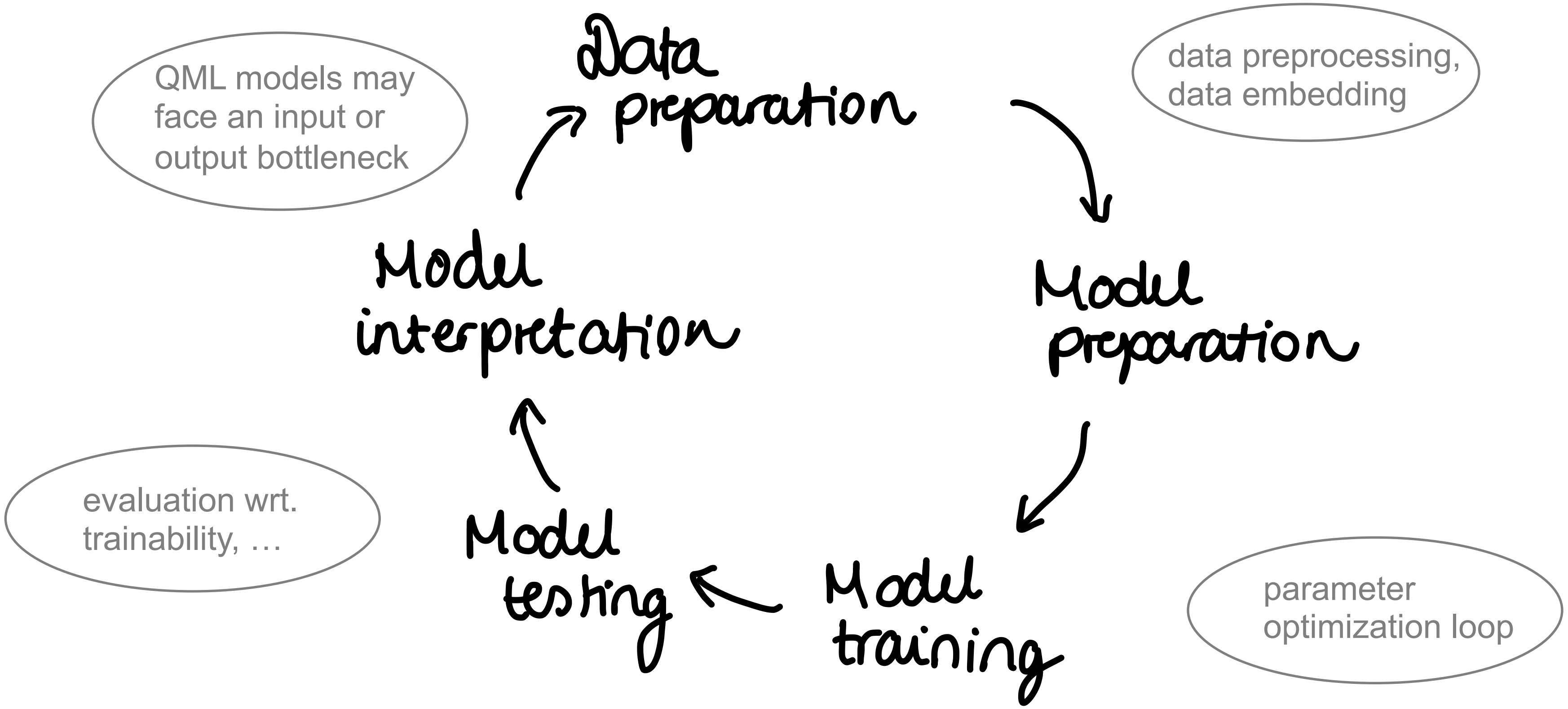
Energy-based models (e.g., QBM)

- Networks of stochastic binary units are optimized wrt. to their energy
- Inspired conceptually by statistical mechanics

Jerbi, Sofiene, et al. "Quantum machine learning beyond kernel methods." *Nature Communications* 14.1 (2023): 1-8.

Schuld, Maria, Ilya Sinayskiy, and Francesco Petruccione. "An introduction to quantum machine learning." *Contemporary Physics* 56.2 (2015): 172-185.

Quantum Machine Learning life cycle



Data preparation

data preprocessing, data embedding

Model preparation

Model training

parameter optimization loop

Model testing

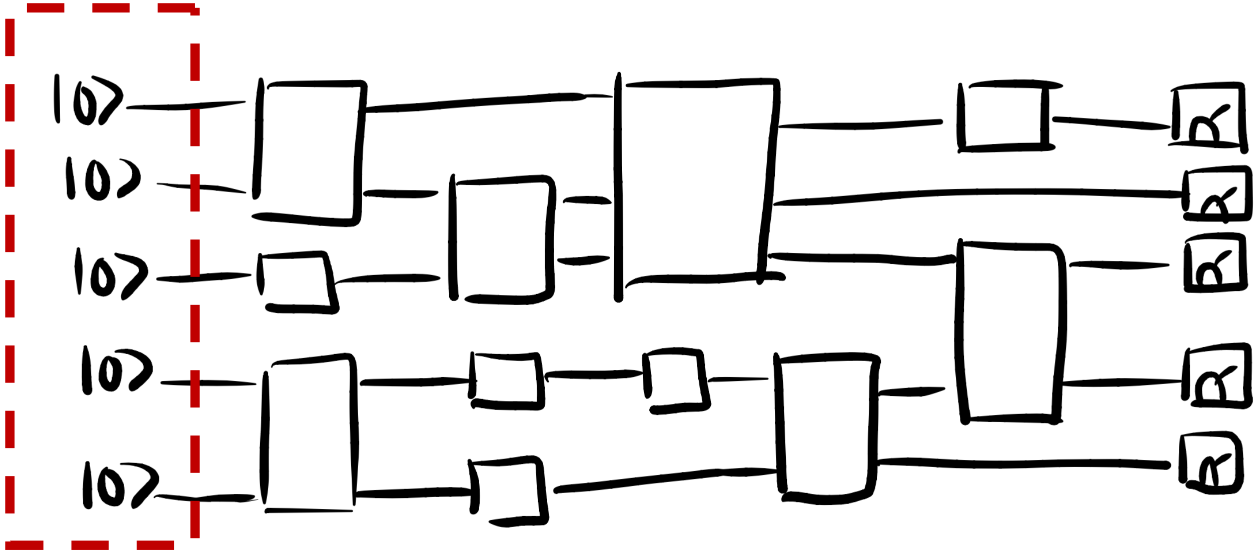
evaluation wrt. trainability, …

Model interpretation

QML models may face an input or output bottleneck

Source: S. Vallecorsa

Quantum Circuits and *the Born rule*

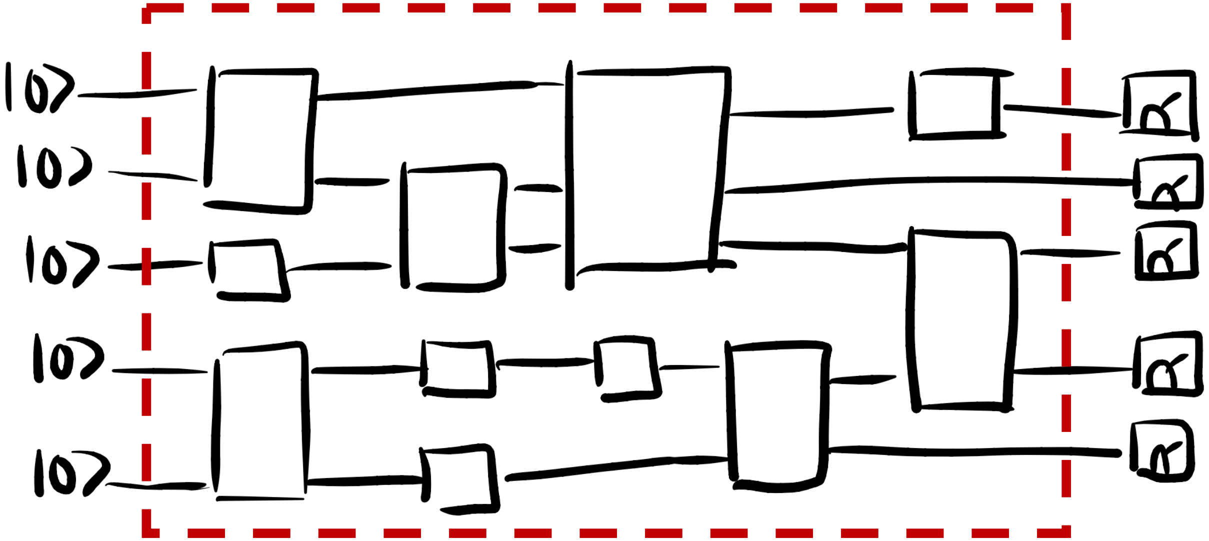


An arbitrary quantum circuit generating the state $|\Psi\rangle$

Initialization:

→ initialize qubits in computational basis state

Quantum Circuits and *the Born rule*

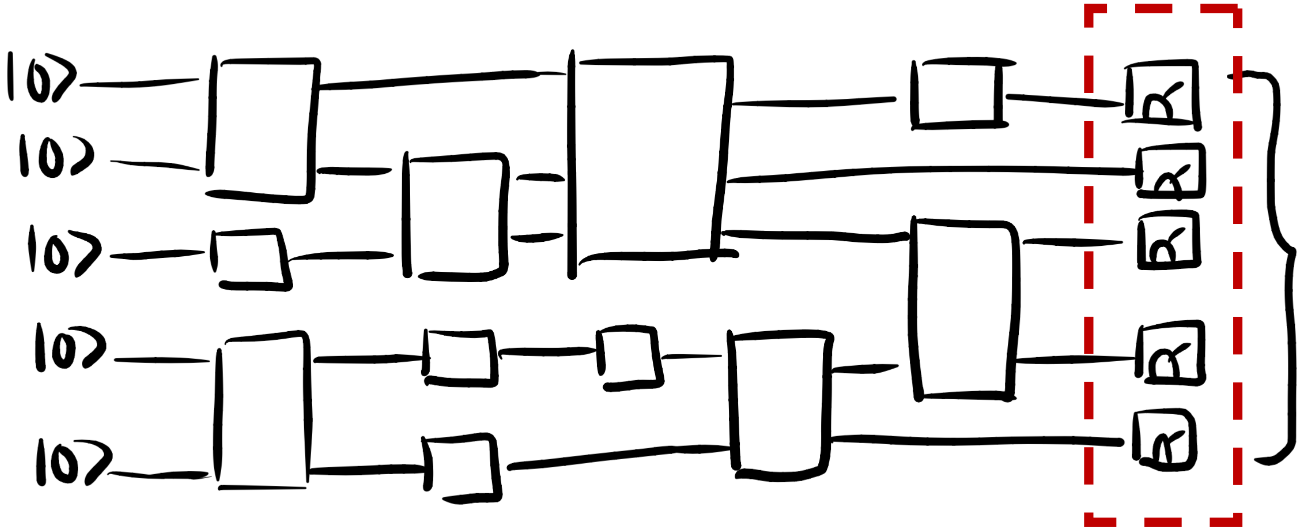


An arbitrary quantum circuit generating the state $|\Psi\rangle$

Evolve initial state:

→ Apply set of **unitary** gates that may **encode classical input data x** and include **parametrized gates**

Quantum Circuits and *the Born rule*



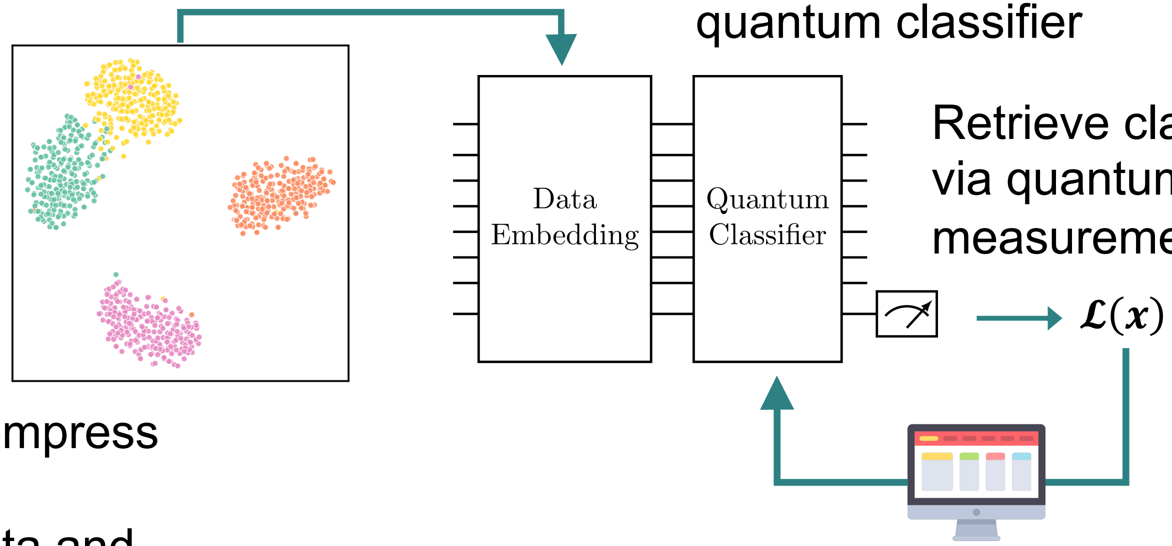
An arbitrary quantum circuit generating the state $|\Psi\rangle$

Quantum Measurement

→ retrieve a classical output distribution $|\langle x|\Psi\rangle|^2$ of classical output states

(with $x \in \{0,1\}^n$) according to Born rule

Parametrized Quantum Circuits – *the data processing pipeline*



Encode (classical) data in Quantum state

Apply a variational quantum circuit as e.g., quantum classifier

Retrieve classical output via quantum measurement

Compute Loss function

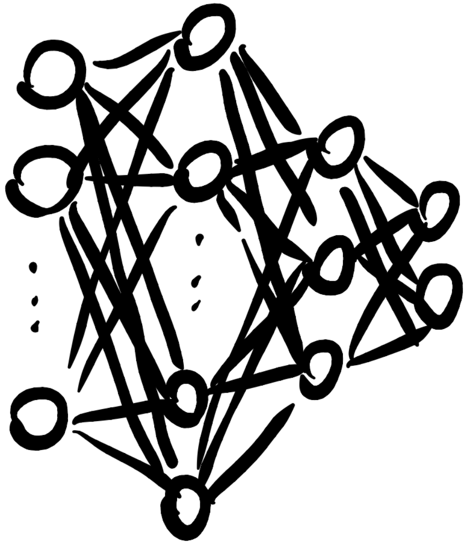
(Classically) compress possible high-dimensional data and extract features

Classically optimize variational circuit parameters

$$\theta_{t+1} \longleftarrow \theta_t - \eta \nabla_x \mathcal{L}(x)$$

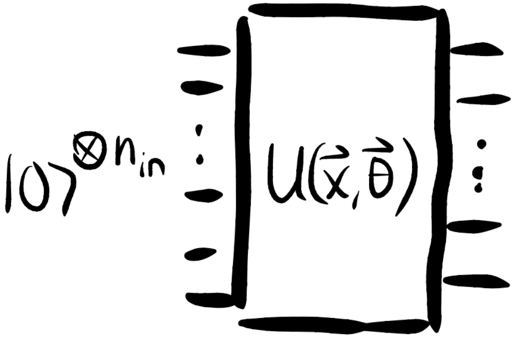
Supervised Learning in Quantum Computing: Quantum Classifiers

→ **Goal:** learn the *input-output relation* of *labeled* data



$$\Psi(\underset{\substack{\uparrow\\ \text{input}}}{\vec{x}}, \underset{\substack{\nwarrow\\ \text{trainable weights}}}{\vec{\theta}}) : \mathbb{R}^{n_{in}} \to \mathbb{R}^{n_{out}}$$

Classical Neural Network



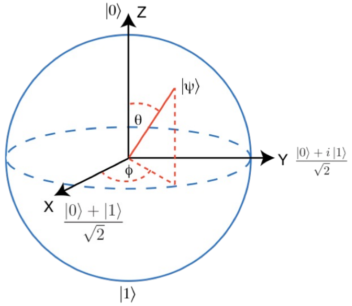
$$|\psi_{out}(\vec{x},\vec{\theta})\rangle = U(\underset{\substack{\nearrow\\ \text{input}}}{\vec{x}}, \underset{\substack{\uparrow\\ \text{trainable weights}}}{\vec{\theta}})|0\rangle^{\otimes n_{in}}$$

$$y(\vec{x},\vec{\theta}) = \langle \psi_{out}(\vec{x},\vec{\theta}) | \hat{O} | \psi_{out}(\vec{x},\vec{\theta}) \rangle$$

Parametrized Quantum Circuit

Embedding classical information in a Quantum Circuit

→ A **tradeoff** between **depth of input** encoding quantum circuit and **exponential compression** of classical input data



Angle encoding:

- Classical input encoded using rotational gates (e.g., $R_x(\theta)$)
- Constant depth wrt. to number of encoded features
- Number of qubits n scales linearly with the number of features N'

$$|\phi(x)\rangle = \bigotimes_{i=1}^{n} (R_x(x_i)|0\rangle) = \bigotimes_{i=1}^{n} (\cos(x_i/2)|0\rangle - i\sin(x_i/2)|1\rangle)$$

$n \propto \mathcal{O}(N)$, $n_{gates} \propto n$

Amplitude encoding:

- Classical input is encoded as amplitudes of the quantum state
- N-dimensional data point x is encoded by a n-qubit quantum state with $N = 2^n$
- Much deeper circuit depth for encoding, see scaling:

$$|\phi(x)\rangle = \frac{1}{||x||} \sum_{i=1}^{2^n} x_i |i\rangle$$

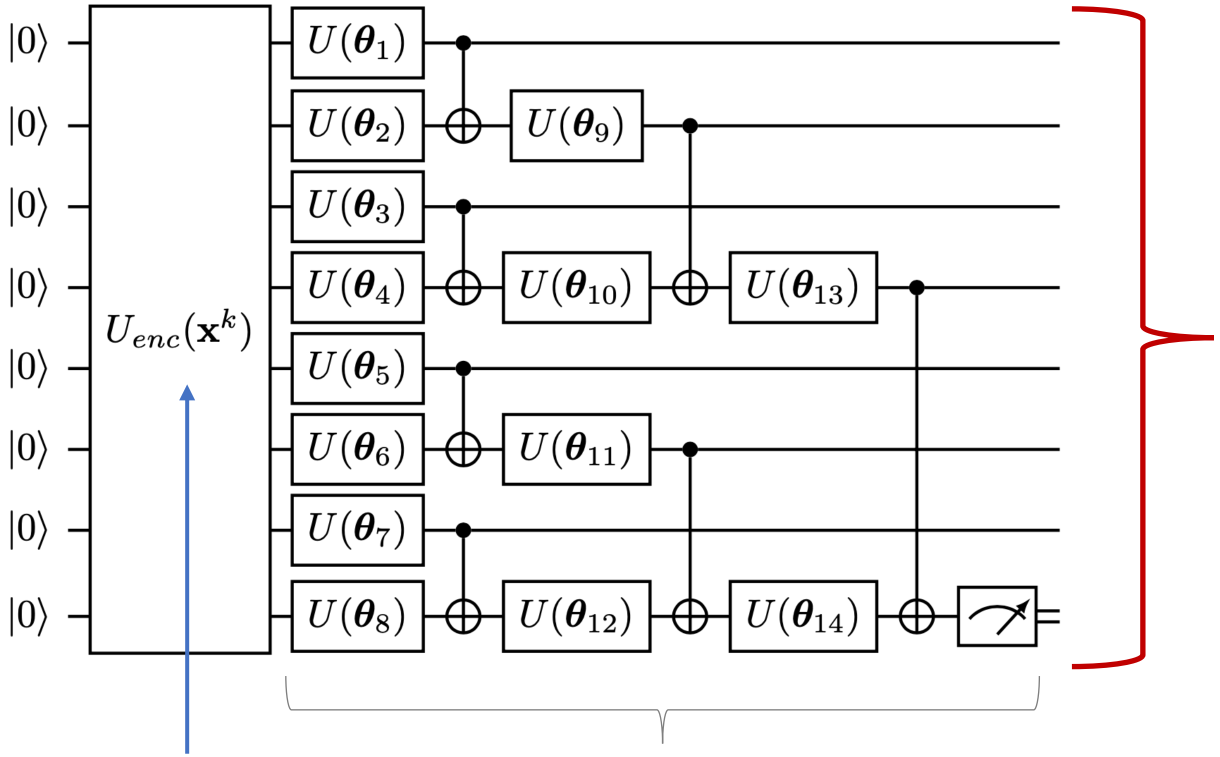
$n \propto \mathcal{O}(\log(N))$, $n_{gates} \propto \mathcal{O}(poly(N)) = \mathcal{O}(poly(2^n))$

Schuld, Maria, and Francesco Petruccione. *Supervised learning with quantum computers*. Vol. 17. Berlin: Springer, 2018.
Image source: Nielsen, Michael A., and Isaac Chuang. "Quantum computation and quantum information." (2002).

Quantum Classifier example: Hierarchical quantum classifiers

Hierarchical quantum classifiers with generic single-qubit unitary gates $U(\theta, \phi, \lambda)$ →

$$U(\theta, \phi, \lambda) = \begin{pmatrix} \cos(\theta/2) & -e^{i\lambda}\sin(\theta/2) \\ e^{i\phi}\sin(\theta/2) & e^{i(\phi+\lambda)}\cos(\theta/2) \end{pmatrix}$$



Apply **Quantum Network as a binary classifier**: measure one qubit

We can encode our classical input features here

Variational part

See: Grant, Edward, et al. "Hierarchical quantum classifiers." *npj Quantum Information* 4.1 (2018): 65.

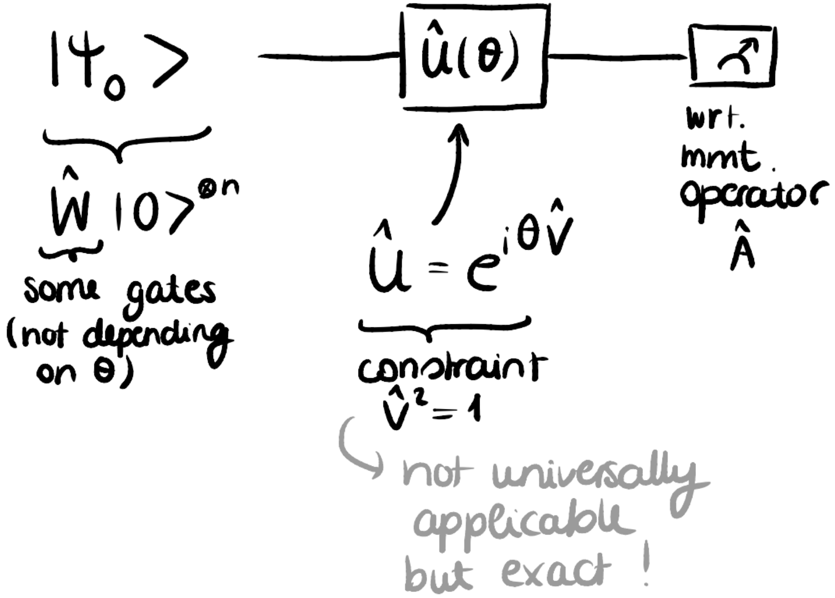
Parameter optimization

The parameter-shift rule (gradient-based)

$$\theta \rightarrow \theta - \eta \nabla_\theta f$$

$$\uparrow \langle \hat{A}(\theta) \rangle$$

→ Compute **partial derivative** of variational circuit parameter θ, alternative to analytical gradient computation and classical finite difference rule (numerical errors and resource cost considerations)



$$\Rightarrow \nabla_\theta \langle \hat{A} \rangle = u\left[\langle \hat{A}(\theta + \frac{\pi}{4u}) \rangle - \langle \hat{A}(\theta - \frac{\pi}{4u} \rangle\right]$$

→ Evaluate Quantum Circuit **twice at shifted parameters** to compute gradient

Source: https://pennylane.ai/qml/demos/tutorial_stochastic_parameter_shift
https://pennylane.ai/qml/demos/tutorial_spsa

Parameter optimization

$$\theta_{k+1} \leftarrow \theta_k - \underbrace{a_k \hat{g}_k(\hat{\theta}_k)}_{\text{stochastic estimate of } \nabla_\theta f}$$

Simultaneous perturbation stochastic approximation (SPSA) (gradient-free)

⟶ If gradient computation is not possible, too resource-intensive, or noise-robustness required (slower convergence but fewer function evaluations)

⟶ The gradient is **approximated by two sampling steps**, and parameters are **perturbed in all directions simultaneously**

$$" \; y(\theta) = f(\theta) + \underset{\text{random output perturbation}}{\varepsilon} \; "$$

$$\hat{g}_{ki}(\hat{\theta}_k) = \frac{y(\hat{\theta}_k + c_k\Delta_k) - y(\hat{\theta}_k - c_k\Delta_k)}{2c_k\Delta_{ki}}$$

Iterative update rule comparable to classical stochastic gradient descent

$c_k \geqslant 0, \; \Delta_k = (\Delta_{k_1}, \Delta_{k_2}, \dots, \Delta_{kp})^T$ perturbation vector (~ randomly sampled from zero-mean distr.)

- https://pennylane.ai/qml/demos/tutorial_stochastic_parameter_shift
- https://pennylane.ai/qml/demos/tutorial_spsa

Challenges when using *Parametrized Quantum Circuits*

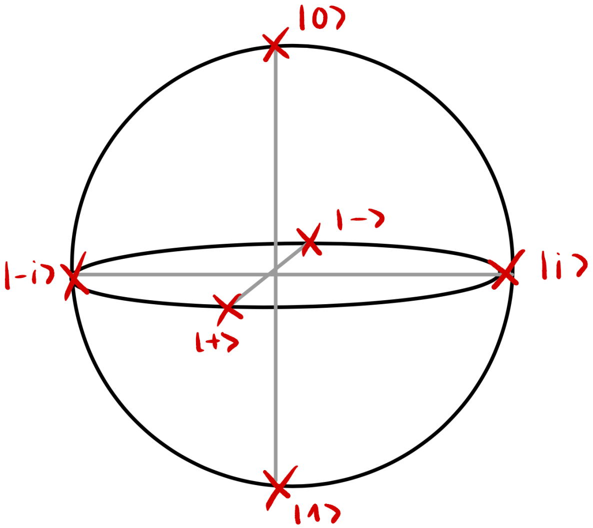
- Efficient **data handling** and data **embedding**
- Find balance: **Generalization** and **representational power** vs. **Convergence**
  - Problem of barren plateaus and vanishing gradients in optimization landscape
  - How well can we survey the Hilbert space (expressibility)?

- **Current hardware limitations**
  - Limited number of qubits and connectivity
  - **Quantum Noise Effects** (decoherence, measurement errors or gate-level errors)
  - Efficient interplay between a classical and a quantum computer
- ….

What is Quantum Advantage in QML?

Multiple considerations:

1. Runtime speed-up
2. Sample complexity
3. Representational power



Bloch sphere: only the marked points are produced by the Clifford operators acting on a computational basis state

This includes considerations regarding **classical intractability**:

Focus on Quantum Circuits that are **not efficiently simulable classically**

Nielsen, Michael A., and Isaac Chuang. "Quantum computation and quantum information." (2002).
Gottesman, Daniel. "The Heisenberg representation of quantum computers." *arXiv preprint quant-ph/9807006* (1998).
See also: - Kübler, Jonas, Simon Buchholz, and Bernhard Schölkopf. "The inductive bias of quantum kernels." *Advances in Neural Information Processing Systems* 34 (2021): 12661-12673.
- Huang, HY., Broughton, M., Mohseni, M. *et al.* Power of data in quantum machine learning. *Nat Commun* **12**, 2631 (2021). https://doi.org/10.1038/s41467-021-22539-9

Interlude: Efficient classical simulation of Clifford circuits

The Gottesman-Knill theorem

*A quantum circuit build up of Clifford gates can be **efficiently simulated** on a **classical computer**.*
(Qubit preparation and measurement in computational basis.)

There are more detailed considerations of cases with different computational complexities.

→ Even **highly entangled states** can be simulated efficiently classically.

Generating set of the Clifford group: $\langle \boldsymbol{H}, \boldsymbol{S}, \boldsymbol{CNOT} \rangle$

Nielsen, Michael A., and Isaac Chuang. "Quantum computation and quantum information." (2002).
Gottesman, Daniel. "The Heisenberg representation of quantum computers." *arXiv preprint quant-ph/9807006* (1998).

What is Quantum Advantage in QML?

Multiple considerations:

- Runtime speed-up
- Sample complexity
- Representational power

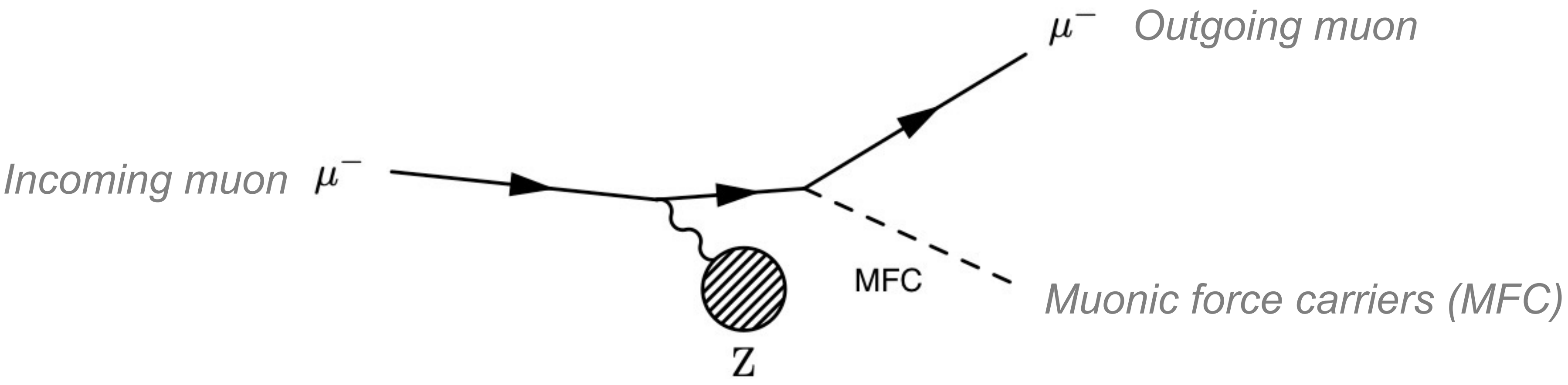
Practical advantage → Practical implementations on NISQ devices

→ *Need for performance metrics and fair comparisons to classical models*

Nielsen, Michael A., and Isaac Chuang. "Quantum computation and quantum information." (2002).
Gottesman, Daniel. "The Heisenberg representation of quantum computers." *arXiv preprint quant-ph/9807006* (1998).
See also: - Kübler, Jonas, Simon Buchholz, and Bernhard Schölkopf. "The inductive bias of quantum kernels." *Advances in Neural Information Processing Systems* 34 (2021): 12661-12673.
- Huang, HY., Broughton, M., Mohseni, M. *et al.* Power of data in quantum machine learning. *Nat Commun* **12**, 2631 (2021). https://doi.org/10.1038/s41467-021-22539-9

Examples and use-cases

Quantum Circuit Born Machine for Event Generation



Muon fixed target scattering experiment

→ MFCs are **bosons** which appear in beyond-the-standard-model theoretical frameworks and are **candidates for dark matter**

→ Monte Carlo calculations are **expensive** in time and CPU consumption

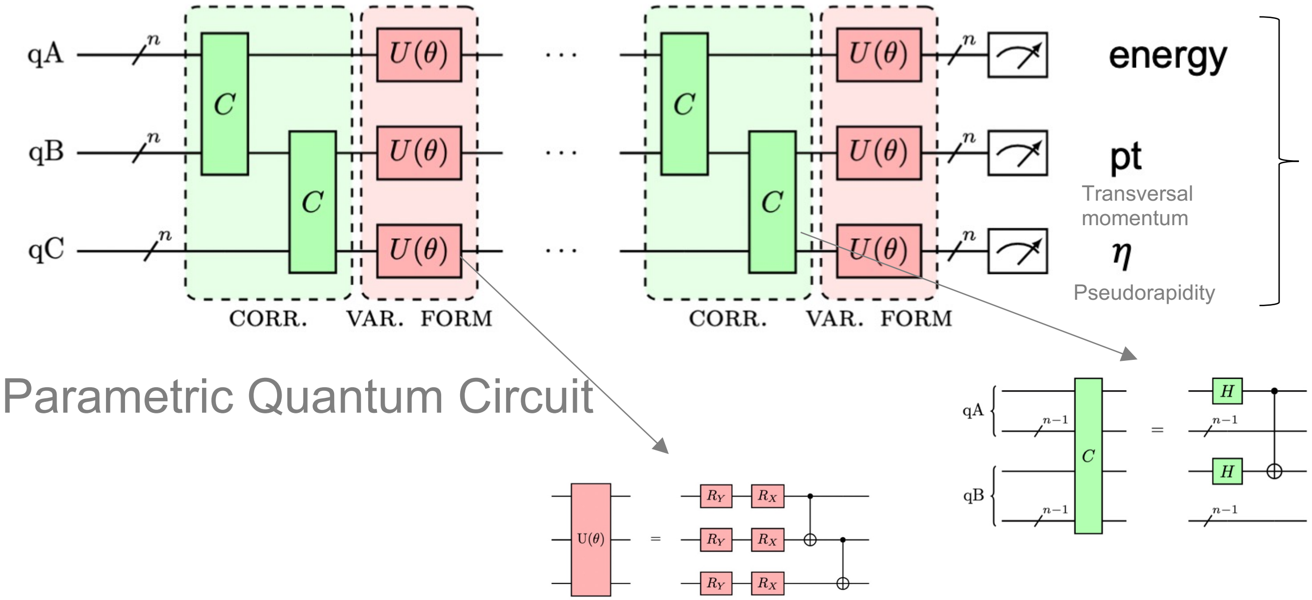
Kiss O., Grossi M. et all.,
Conditional Born machine for Monte Carlo events generation,
Phys. Rev. A ***106****, 022612 (2022)*

Quantum Circuit Born Machine for Event Generation

Born machine:

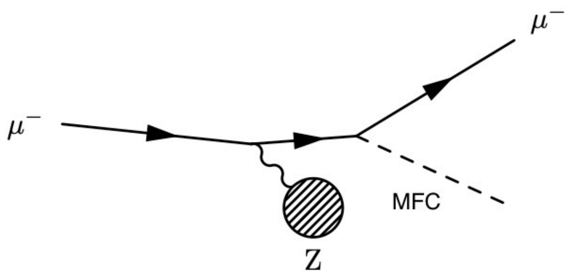
Produces statistics according to Born's measurement rule using parametrized quantum circuit $|\psi(\theta)\rangle$

$$p_\theta(x) = |\langle x|\psi(\theta)\rangle|^2,\ x \in \{0,1\}^{3n}$$



Parametric Quantum Circuit

Generate discrete PDFs
(continuous in the limit increasing no. of qubits)



Muon fixed target scattering experiment

Kiss O., Grossi M. et all.,
Conditional Born machine for Monte Carlo events generation,
Phys. Rev. A ***106****, 022612 (2022)*

Coyle, B., Mills, D. et al, **The Born supremacy.** In: *npj Quantum Inf 6, 60* (2020)

Quantum Circuit Born Machine for Event Generation

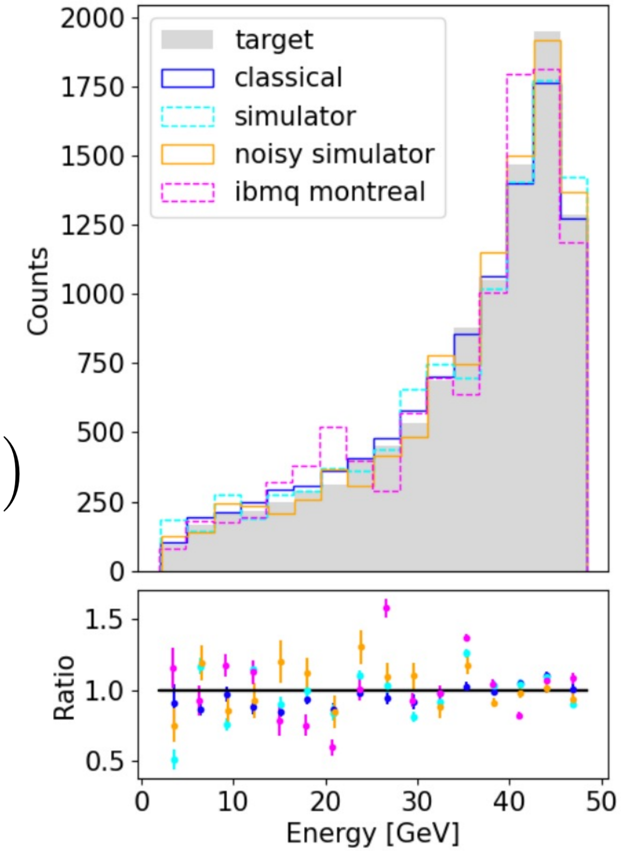
- **Generate samples of discrete PDFs with Born machine**
- Train using Maximum Mean Discrepancy loss function:

$$\mathrm{MMD(P,Q)} = \mathbb{E}_{\substack{X\sim P\\ Y\sim P}}[K(X,Y)] + \mathbb{E}_{\substack{X\sim Q\\ Y\sim Q}}[K(X,Y)] - 2\mathbb{E}_{\substack{X\sim P\\ Y\sim Q}}[K(X,Y)]$$

Gaussian kernel

$$K(x,y) = \exp\left(-\frac{(x-y)^2}{2\sigma}\right)$$

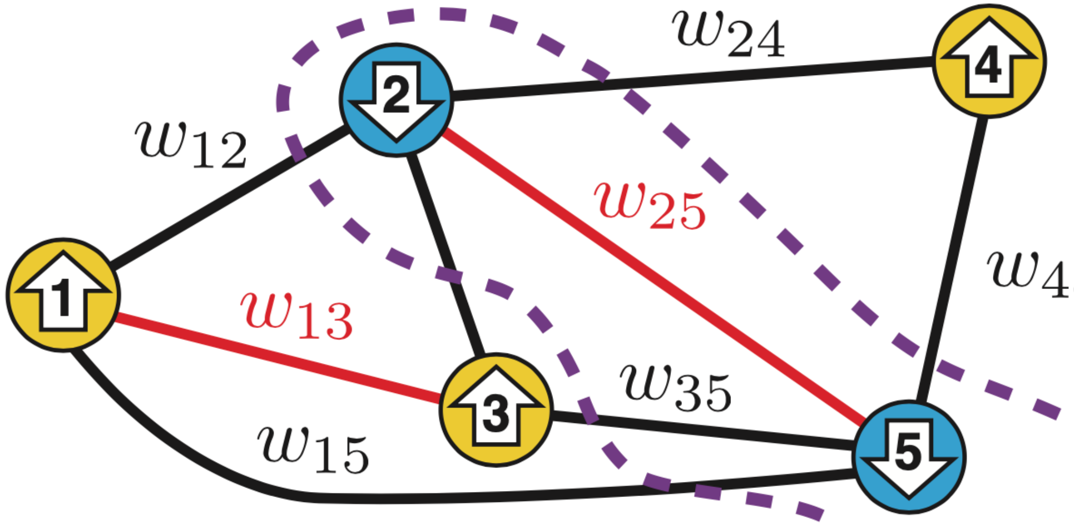
→ **efficient way to generate multivariate (and conditional) distributions** for NISQ devices (suggested by numerical evidence)



Kiss O., Grossi M. et all., **Conditional Born machine for Monte Carlo events generation**, *Phys. Rev. A* ***106****, 022612 (2022)*

Coyle, B., Mills, D. et al, **The Born supremacy.** In: *npj Quantum Inf 6, 60* (2020)

Solving combinatorial problems using QAOA



$w_{i,j} = 1 \text{ for} \, \forall \, i, j$

The MaxCut problem (NP-complete)

Goal: partition the graph into two groups and maximize the number of edges connecting both partitions

⟶ assign binary variables to nodes

Zhou, Leo, et al. "Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices." *Physical Review X* 10.2 (2020): 021067.

Solving combinatorial problems using QAOA

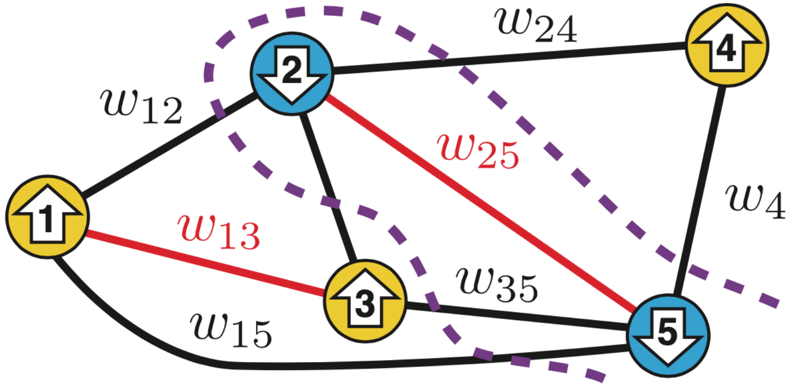
Aiming to solve a **QUBO problem**
of the form: $f_Q(x) = x^T Q\, x,\ \ x \in \{0,1\}^N$

↓

Map to an **Ising Hamiltonian**
of the general form $H = \sum_{i,j} J_{i,j} Z_i Z_j$

MaxCut problem

$w_{i,j} = 1 \text{ for } \forall\, i, j$



$$H_G = \frac{1}{2} \sum_{\{u,v\} \in E} (I - Z_u Z_v) \qquad G = (V, E)$$

Zhou, Leo, et al. "Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices." *Physical Review X* 10.2 (2020): 021067.

Solving combinatorial problems using QAOA

$$f_Q(x) = x^T Q x = \sum_{i,j=1}^{N} Q_{i,j}\, x_i\, x_j \qquad x^* = \underset{x\in\{0,1\}^N}{\arg\min}\ f_Q(x)$$

QUBO problem

$$\Big\downarrow \begin{array}{l} x_i \in \{0,1\} \\ z_i \in \{1,-1\} \end{array}$$

$$H_c\,(z) = \sum_{i=1}^{N-1} \sum_{j>i}^{N} Q_{i,j}\,(1 - z_i)\,(1 - z_j)$$

Ising-type Hamiltonian

$$\Big\downarrow$$

$$H(t) = A(t)\, H_M + B(t)\, H_c \qquad H_M = \sum_i \sigma_i^x$$

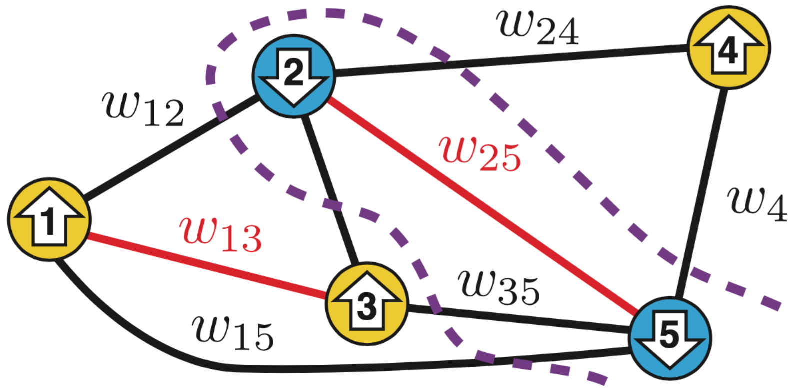
$$A(t),\ B(t) : [0,T] \mapsto \mathbb{R} \qquad A(0) = B(T) = 1 \quad A(T) = \ B(0) = 0$$

Quantum Adiabatic Algorithm

$\searrow$ *Trotterization* of Temporal Evolution

MaxCut problem

$$w_{i,j} = 1 \text{ for } \forall\, i, j$$

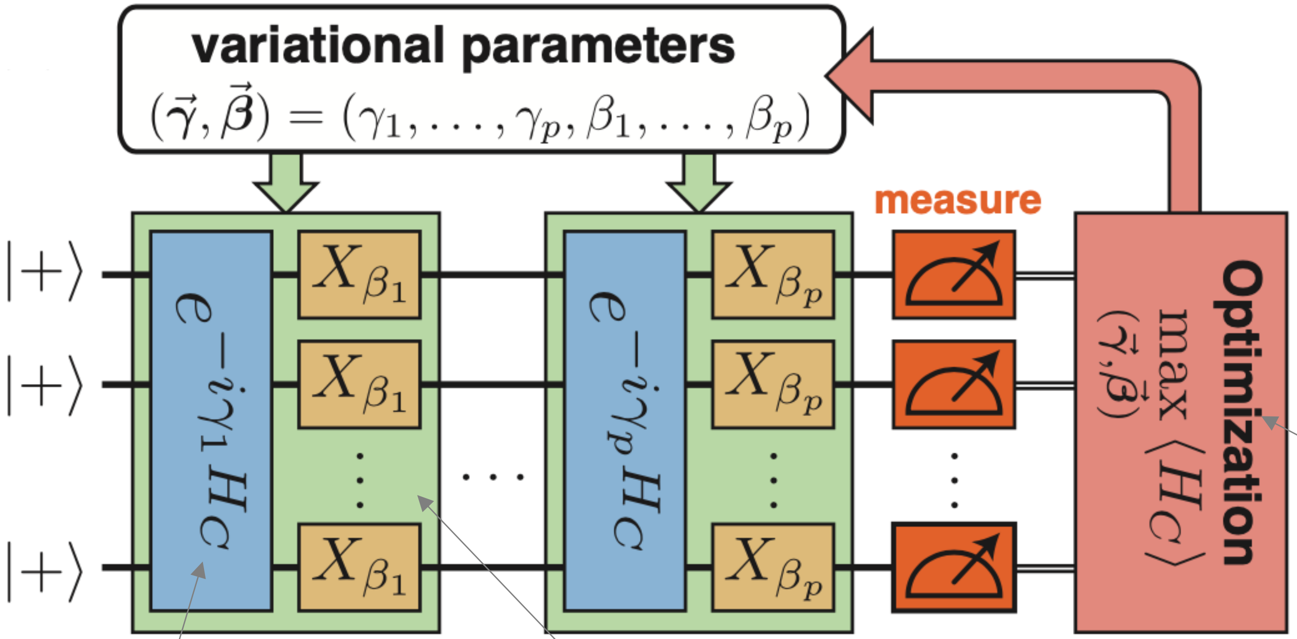


$$H_G = \frac{1}{2} \sum_{\{u,v\}\in E} (I - Z_u Z_v) \qquad G = (V, E)$$

Zhou, Leo, et al. "Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices." *Physical Review X* 10.2 (2020): 021067.

Solving combinatorial problems using QAOA

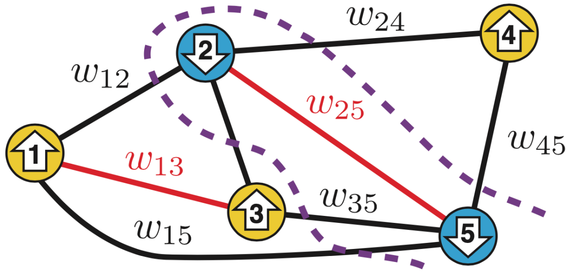
QAOA Ansatz and hybrid optimization procedure



Cost Hamiltonian

Mixer Hamiltonian

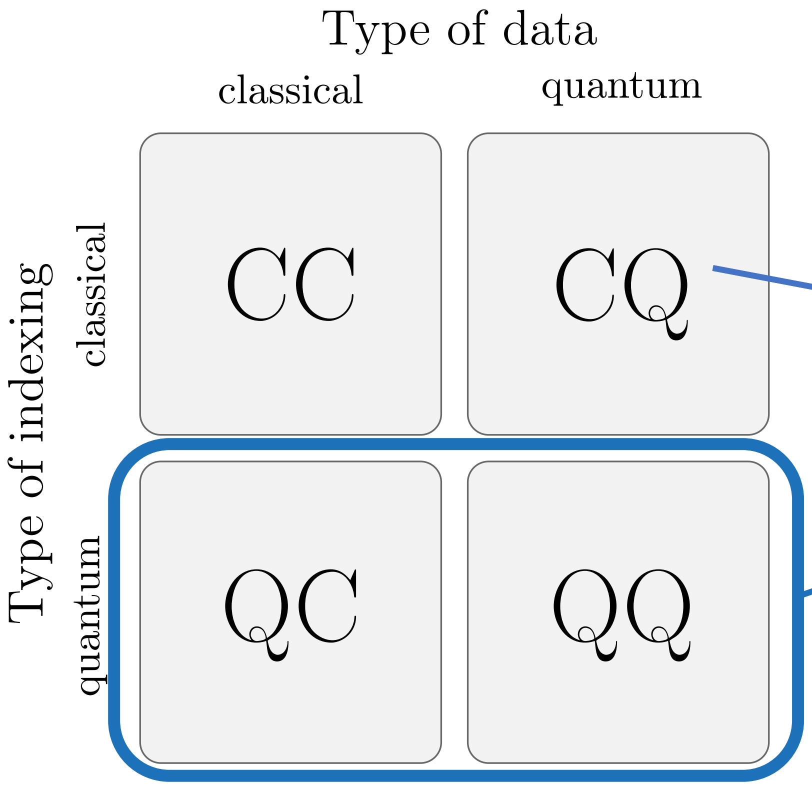
Classically optimize angles:
Hybrid procedure



MaxCut Graph

Zhou, Leo, et al. "Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices." *Physical Review X* 10.2 (2020): 021067.

Quantum Databases in a General Context



$$\bigotimes_{i=0}^{p} |d_i\rangle_D \in (\mathcal{H}_D)^{\otimes p}$$

$$|\mathrm{QDB}^{(k)}\rangle = \sum_{j=0}^{k-1} \psi_j |j\rangle_I |d_j\rangle_D \in \mathcal{H}_I \otimes \mathcal{H}_D$$

Rieger, Carla, et al. **"Operational Framework for a Quantum Database."** *arXiv preprint arXiv:2405.14947* (2024).

Quantum Databases (QDB)

How do we operate on superposition states containing a quantum index register correlated to data registers?

$$|\mathrm{QDB}^{(k)}\rangle = \sum_{j=0}^{k-1} \psi_j \, |j\rangle_I \, |d_j\rangle_D \in \mathcal{H}_I \otimes \mathcal{H}_D$$

→ QDB’s are **relevant for quantum algorithms to operate on** quantum database states and dynamically manipulate them.

→ Make use of **exponential compression** due to the usage of the superposition principle.

→ Manipulation operations are defined to **mimic classical database operations.**

prepare, *extend*, *remove*,
write, *read-out* and *permute*

Rieger, Carla, et al. **"Operational Framework for a Quantum Database."** *arXiv preprint arXiv:2405.14947* (2024).

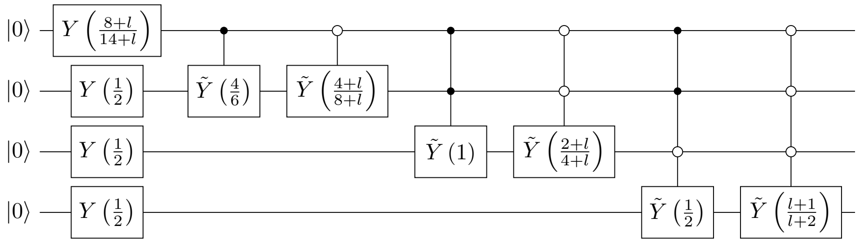
Quantum Databases (QDB)

Example: *QDB Prepare Algorithm with qubits for* ***State Preparation***

$$\mathcal{H}_I \otimes \mathcal{H}_D \to \mathcal{H}_I \otimes \mathcal{H}_D$$

$$|0\rangle_{I\otimes D} \xmapsto{P_{(k)}} \frac{1}{\sqrt{k}} \sum_{j=0}^{k-1} |j\rangle_I |0\rangle_D$$

$\log_2(k) \notin \mathbb{N}$



Linear-depth (in no. qubits) quantum circuit (e.g., $k = 14$)

$\log_2(k) \in \mathbb{N}_{>0}$

$$P_{(k)} = H^{\otimes \log_2(k)} \otimes \mathbb{I}_D$$

Constant-depth quantum circuit

Rieger, Carla, et al. **"Operational Framework for a Quantum Database."** *arXiv preprint arXiv:2405.14947* (2024).

Outlook on QML and summary

Research on QML applications in High Energy Physics is producing a **large number of prototypical algorithms for potential future use-cases:**

- Currently focus on *algorithms for data processing* in a *controlled* environment for current hardware
- Preliminary hints for advantage in terms of *representational power of quantum states*
- Mostly, algorithm performance is *as good as* the classical counterpart
- Need *more robust studies* to relate architecture of quantum computational model and its performance to data sets
- *Identify use-cases* where quantum approach is provably *more efficient* than classical model
- Studying QML algorithms today *links Quantum computing and Learning Theory* and draw separation between classical and quantum learner

Based on previous year‘s talk

Thank you, are there any questions?

carla.sophie.rieger@cern.ch