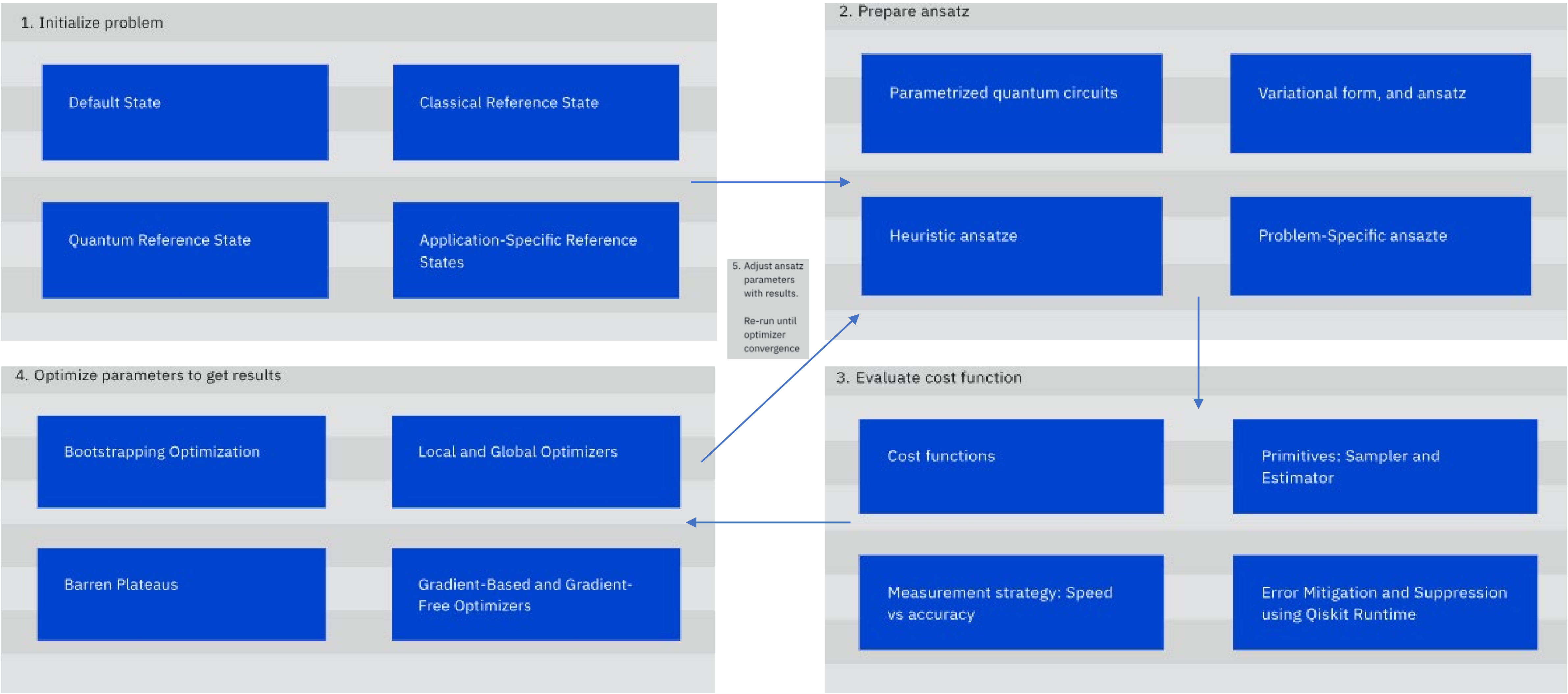
Variational algorithm design

Variational quantum algorithms (VQA)

- Near-term hybrid quantum-classical algorithms based on the variational theorem of quantum mechanics
- These algorithms can leverage the utility provided by today's non-fault-tolerant quantum computers, making them ideal candidates to achieve quantum advantage
- Variational algorithms are very commonly used in near term quantum optimization and quantum machine learning (QML) algorithms in various forms.
- Variational algorithms include several modular components that can be combined and optimized based on algorithm, software, and hardware advancements

A typical simplified hybrid workflow



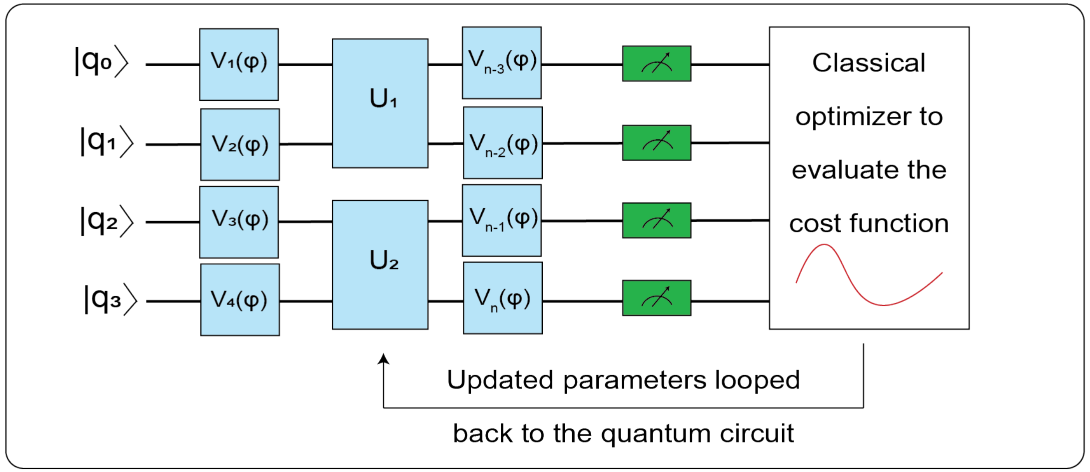
Components of VQA

Most common components are:

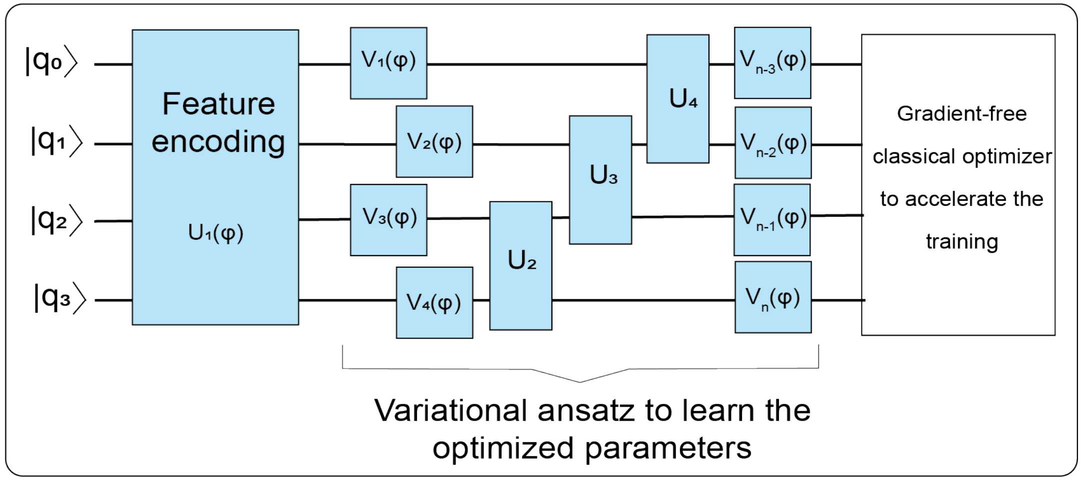
- A **cost function** that describes a specific problem with a set of parameters,
- An **ansatz** to express the search space with these parameters,
- An **optimizer** to iteratively explore the search space.
- During each iteration, the optimizer evaluates the cost function with the current parameters and selects the next iteration's parameters until it converges on an optimal solution. The hybrid nature of this family of algorithms comes from the fact that the cost functions are evaluated using quantum resources and optimized through classical ones.

Variational quantum algorithms

A typical variational circuit setup for optimization problems



A typical variational circuit setup for machine learning problems



Why does this work?

- A small digression into mathematics and quantum mechanics:

$$
\begin{aligned}
|0\rangle \xrightarrow{U_R} U_R|0\rangle = |\rho\rangle &\xrightarrow{U_V(\vec{\theta})} U_A(\vec{\theta})|0\rangle \\
&= U_V(\vec{\theta})U_R|0\rangle \\
&= U_V(\vec{\theta})|\rho\rangle \\
&= |\psi(\vec{\theta})\rangle
\end{aligned}
$$

Initialized at all 0 state, a reference state is constructed with unitary U_R. Then the parametrized ansatz is applied to get to the target state $|\psi(\theta)\rangle$.

- Variational theorem ensures that the we can sample the lowest eigenvalue of the Hamiltonian of the system to approximate the solution.
- We can start by writing the Hamiltonian of the system via spectral decomposition:

$$\hat{\mathcal{H}} = \sum_{k=0}^{N-1} \lambda_k |\phi_k\rangle\langle\phi_k|$$

where λ_k are eigenvalues and $|\phi_k\rangle$ are eigenstates.

Why does this work?

- We can calculate the expected energy of a system (expectation value of the observable):

$$\begin{aligned}\langle\psi|\hat{\mathcal{H}}|\psi\rangle &= \langle\psi|\left(\sum_{k=0}^{N-1}\lambda_k|\phi_k\rangle\langle\phi_k|\right)|\psi\rangle \\ &= \sum_{k=0}^{N-1}\lambda_k\langle\psi|\phi_k\rangle\langle\phi_k|\psi\rangle \\ &= \sum_{k=0}^{N-1}\lambda_k|\langle\psi|\phi_k\rangle|^2\end{aligned}$$

- We can show that the calculated expectation value is always higher than the ground state energy of the system:

$$\begin{aligned}\langle\psi|\hat{\mathcal{H}}|\psi\rangle &= \sum_{k=0}^{N-1}\lambda_k|\langle\psi|\phi_k\rangle|^2 \\ &\geq \sum_{k=0}^{N-1}\lambda_0|\langle\psi|\phi_k\rangle|^2 \\ &= \lambda_0\sum_{k=0}^{N-1}|\langle\psi|\phi_k\rangle|^2 \\ &= \lambda_0\end{aligned}$$

$$\langle\psi|\hat{\mathcal{H}}|\psi\rangle \geq \lambda_0.$$

Assuming $\lambda_0 \leq \lambda_k$ for all k.

This is the sum of all probabilities of measuring $|\phi_k\rangle$ which adds up to 1.

Variational theorem of quantum mechanics

- Then considering the cost function, we can minimize the parameters to get close to the ground state solution:

$$\min_{\vec{\theta}} C(\vec{\theta}) = \min_{\vec{\theta}} \langle \psi(\vec{\theta}) | \hat{\mathcal{H}} | \psi(\vec{\theta}) \rangle \geq \lambda_0.$$

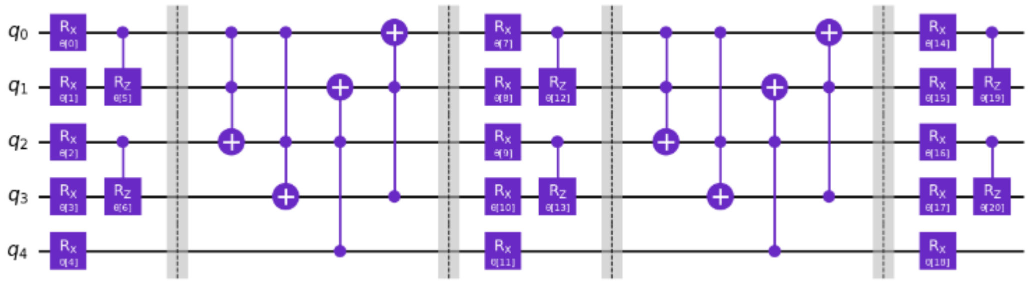
- If the normalized state $|\psi\rangle$ of a quantum system depends on a parameter vector $\vec{\theta}$, then the optimal approximation of the ground state (i.e. eigenstate $|\phi_0\rangle$ with the minimum eigenvalue λ_0) is the one that minimizes the expectation value of the Hamiltonian $\widetilde{H}$:

$$\langle \hat{\mathcal{H}} \rangle (\vec{\theta}) := \langle \psi(\vec{\theta}) | \hat{\mathcal{H}} | \psi(\vec{\theta}) \rangle \geq \lambda_0$$

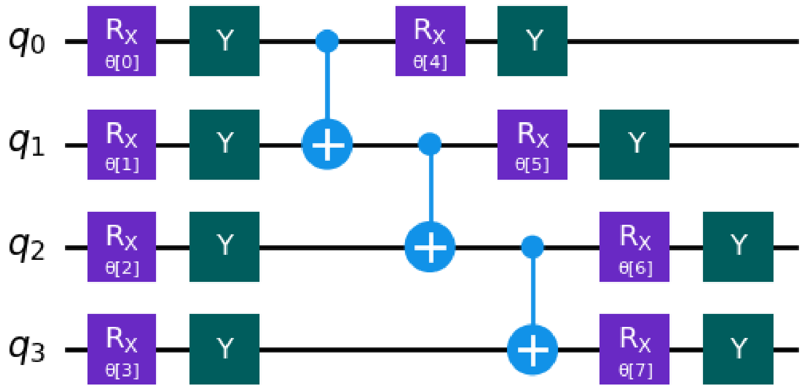
- We also make the following mathematical assumptions:
  - A finite lower bound to the energy $E \geq \lambda_0 > -\infty$ needs to exist
  - Upper bounds do not generally exist

How to set up a variational algorithm

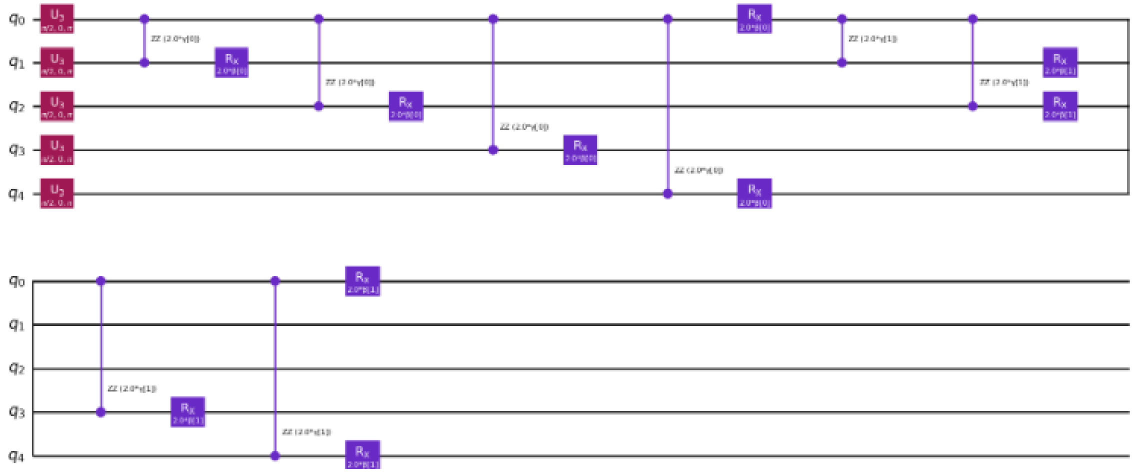
N-local ansatz



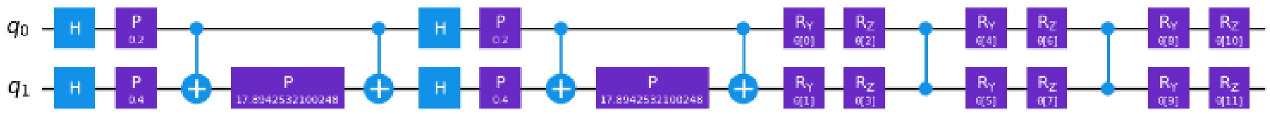
Hardware efficient ansatz



Problem specific ansatz for optimization

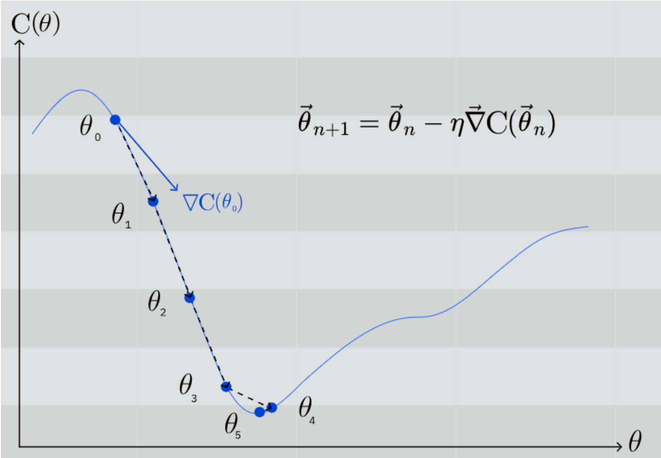


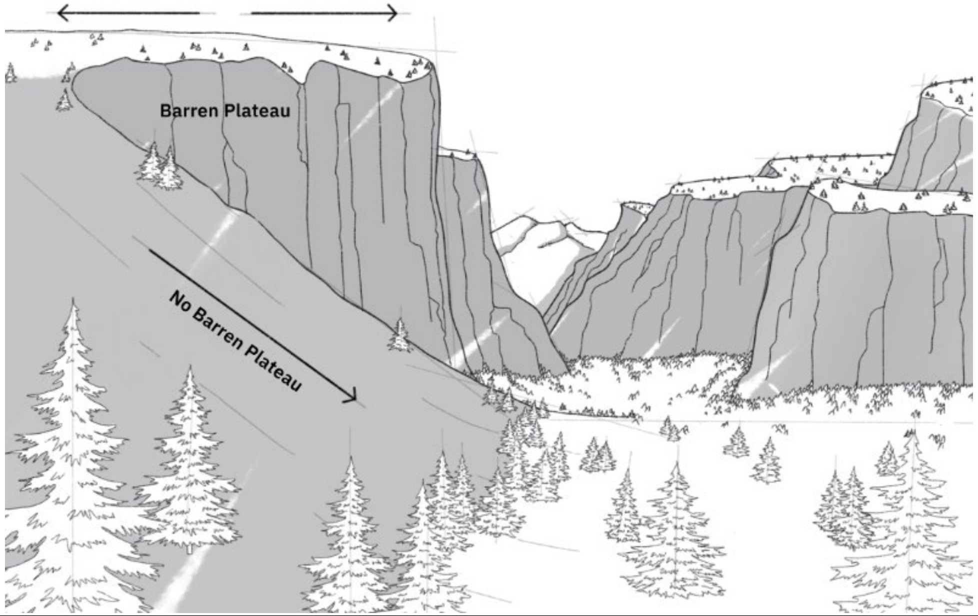
Problem specific ansatz for quantum machine learning



How to set up a variational algorithm

- Qiskit library offers various ways to implement Cost functions as a sum of Pauli operators.
- The primitives Estimator and Sampler can be used to compute expectation values of the operators in the variational loop.
- Supports local and global classical optimizers. Depending on the application instance, you can choose gradient-based or gradient-free optimizers to update the parameters in the variational form.

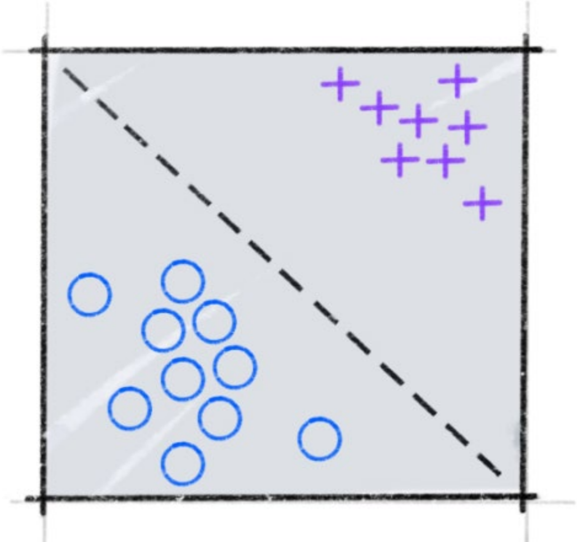




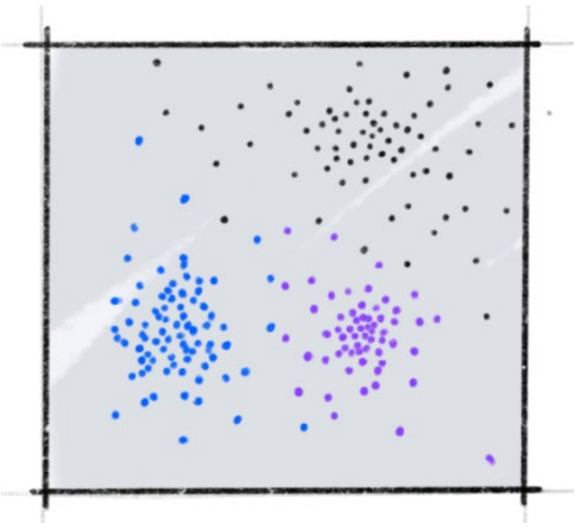
Quantum machine learning

- Machine learning has established itself as a successful interdisciplinary field which seeks to find patterns in data. Throwing in quantum computing gives rise to interesting areas of research that aim to use the principles of quantum mechanics to augment machine learning, or vice-versa.

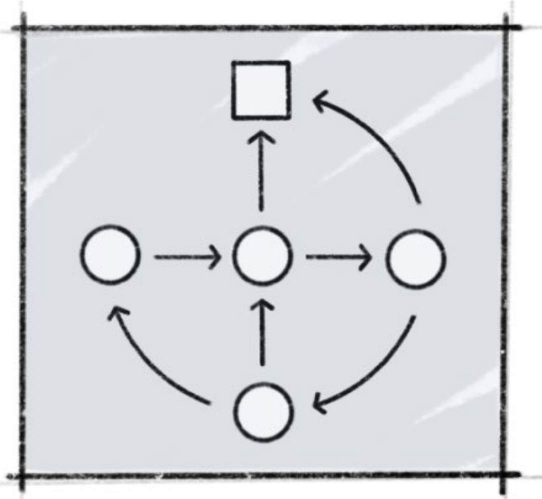
Supervised learning



Unsupervised learning



Reinforcement learning

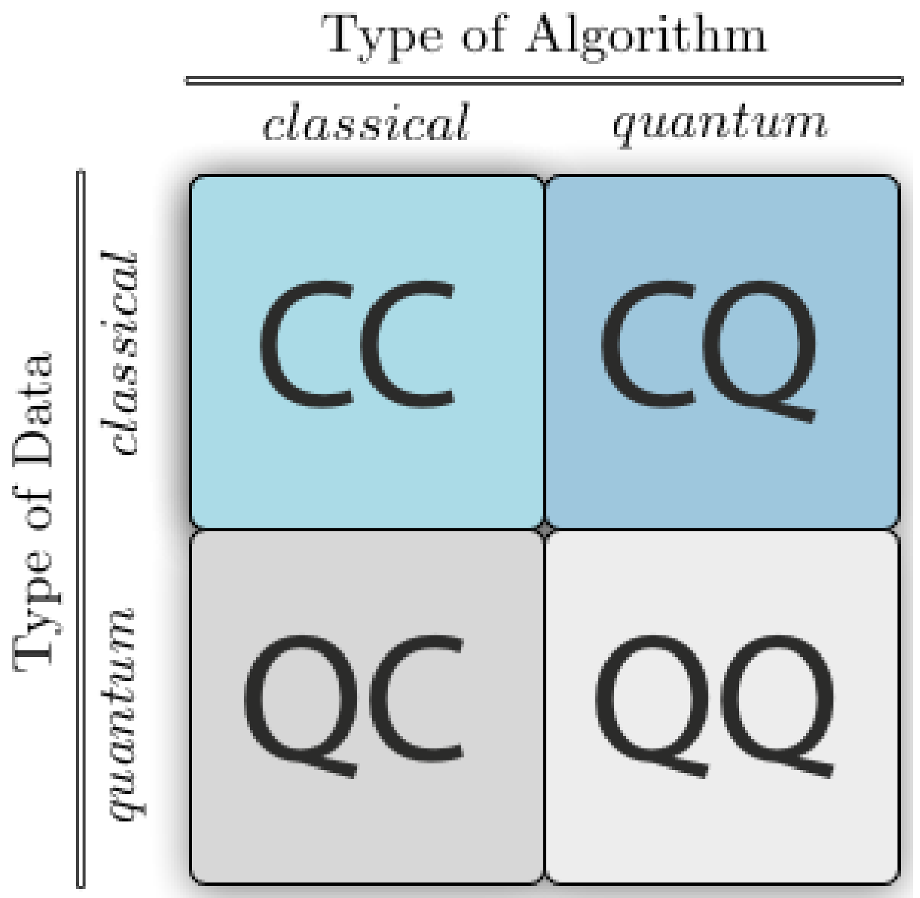


Quantum machine learning

- Recently, most of the focus of **CQ** approaches to machine learning has been on near-term algorithms that can be executed on the current quantum devices.

- Note that QML is still an active research area where we are already seeing results competitive with classical ML methods.

- Many quantum algorithms have been proposed for QML, spanning all supervised, unsupervised and reinforcement learning models.



Data encoding

- Data representation is crucial for the success of machine learning models. For classical machine learning, the problem is how to represent the data numerically, so that it can be best processed by a classical machine learning algorithm.

- For quantum machine learning, this question is similar, but more fundamental: how to represent and efficiently input the data into a quantum system, so that it can be processed by a quantum machine learning algorithm. This is usually referred to as data encoding but is also called data *embedding* or *loading.*

- Some common methods are:

    - Basis encoding
    - Amplitude encoding
    - Angle encoding
    - Arbitrary encoding

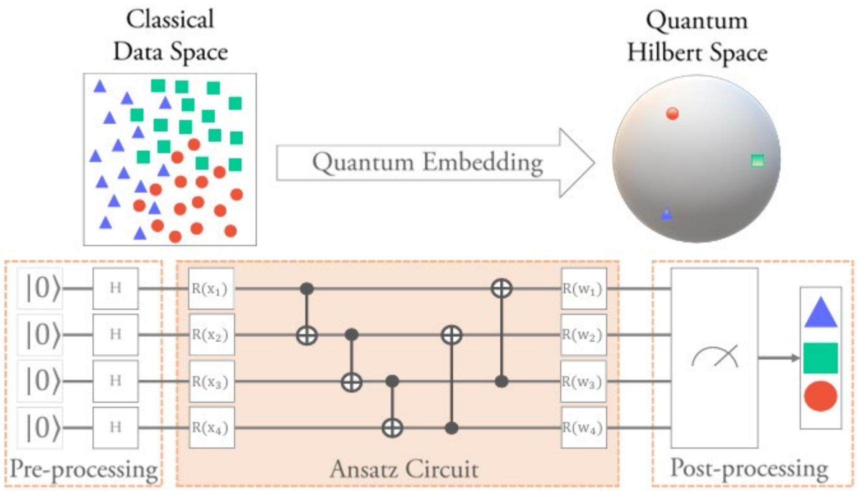


Image from: Quantum Embedding Search for Quantum Machine Learning, NamNguyen and Kwang-Chen Chen, IEEE

Data encoding

- Given a data set with M samples, each with N features:

$D = \{ x^1, x^2, \dots, x^M \}$ where each x^i is an N dimensional vector.

- **Basis encoding**

Basis encoding associates a classical N-bit string with a computational basis state of an N-qubit system. Essentially, each data point must be an N-bit string $x^i = (b_1, b_2, \dots, b_N)$ which will be mapped to a quantum state $|x^m\rangle = |b_1 b_2 \dots b_N\rangle$ with $b_j \in \{0,1\}$. Then we can represent the entire data set as superpositions of computational basis states:

$$|D\rangle = \frac{1}{\sqrt{M}} \sum_{m=1}^{M} |x^m\rangle$$

- **Amplitude encoding**

Amplitude encoding encodes data into the amplitudes of a quantum state. To encode the data set D, we concatenate all M vectors (each of dimension N) into one amplitude vector of length $M \times N$:

$$\alpha = A_{norm}(x_1^1, \dots, x_N^1, \dots x_1^m, \dots x_N^m)$$

Where A_{norm} is a normalization factor such that $|\alpha|^2 = 1$. Then we have:

$$|D\rangle = \sum_{i=1}^{N} \alpha_i |i\rangle$$

Data encoding

- Given a data set with M samples, each with N features:

$D = \{ x^1, x^2, \dots, x^M \}$ where each x^i is an N dimensional vector.

- **Angle encoding**

Angle encoding encodes N features into the rotation angles of *n* qubits for $N \leq n$. For example, the data point $\mathrm{x} = (x_1, x_2, \dots, x_N)$ can be encoded as follows:

$$|x\rangle = \otimes_{i=1}^{N} \cos(x_i)\,|0\rangle + \sin(x_i)|1\rangle$$

This is different from the previous two encoding methods, as it only encodes one data point at a time, rather than a whole dataset. It does, however, only require N qubits or less and a constant depth quantum circuit, making it amenable to current quantum hardware.

We can specify the angle encoding as a unitary:

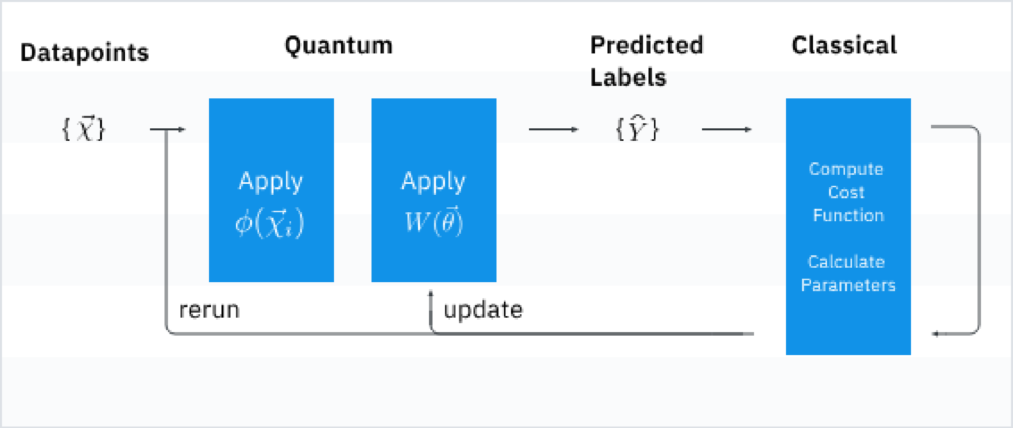
$$S_{x_j} = \otimes_{i=1}^{N} U(x_j^i) \longrightarrow U(x_j^{(i)}) = \begin{bmatrix} \cos(x_j^{(i)}) & -\sin(x_j^{(i)}) \\ \sin(x_j^{(i)}) & \cos(x_j^{(i)}) \end{bmatrix}$$

Quantum Support Vector Classifier (QSVC)

- Variational quantum classifier (VQC) is the simplest classifier available in Qiskit Machine Learning and is a good starting point for newcomers to quantum machine learning who have a background in classical machine learning.

- Two of its central elements are the feature map and ansatz.

- Our data is classical, meaning it consists of a set of bits, not qubits. We need a way to encode the data as qubits. This process is crucial if we want to obtain an effective quantum model. This is the role of the feature map. While feature mapping is a common ML mechanism, this process of loading data into quantum states does not appear in classical machine learning as that only operates in the classical world.



Variational quantum classifier

- The variational quantum classifier is a variational algorithm where the measured expectation value is interpreted as the output of a classifier, introduced by multiple groups in 2018.

- Very commonly used for classification problems (especially binary classification), hence it is a supervised learning algorithm.

- Like the classical vector classifiers, after encoding the data as a quantum state, we train the variational ansatz to converge to optimal weights using the cost function.