Introduction to Quantum Machine Learning

Cenk Tüysüz
28 February 2024

Outline

- Introduction to Quantum Computing
  - What is a qubit?
  - Gate based Quantum Computing
  - Using a Quantum Computer

- Quantum Machine Learning
  - Examples of models
  - Arguments for QML
  - Problems in QML

What is a qubit?

<u>Classical Bits</u>

Classical bits store binary information (0 or 1). We call this a “state”. A classical bit can only be on one of these states.

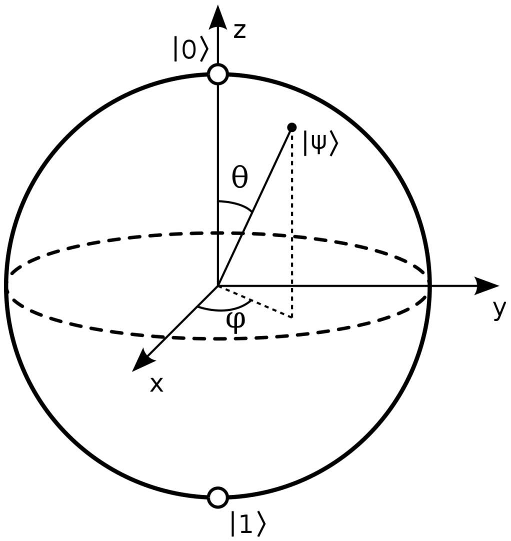
<u>Qubits</u>

Qubits can store quantum mechanical states. This allows them to show quantum mechanical properties such as *superposition* and *entanglement*.

In quantum mechanics we are not limited to two level systems. We can have multi-level systems. These are named qutrits, ququart and qudrit (for d dimensional case)

The bloch sphere:

This representation is limited to a 1 qubit states and do not generalize.



What is a

Classical Bits

Classical bits st
call this a “state
these states.

Qubits

Qubits can store
allows them to s
such as *superpo*

In quantum mec
systems. We ca
are named qutri
dimensional cas

ıbit states and

y

How to represent quantum states?

There are many ways to represent quantum states. We can use matrices, the braket notation from quantum mechanics or even invent new graphical representations. Let’s have a look at these while we introduce some tools of quantum computing. The most generic single-qubit quantum state can be expressed as,

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle\,, \text{ where } \alpha,\ \beta \in \mathbb{C} \text{ and } ||\alpha||^2 + ||\beta||^2 = 1$$

In matrix form:

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix},\ |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix},\ \text{and } |\psi\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}$$

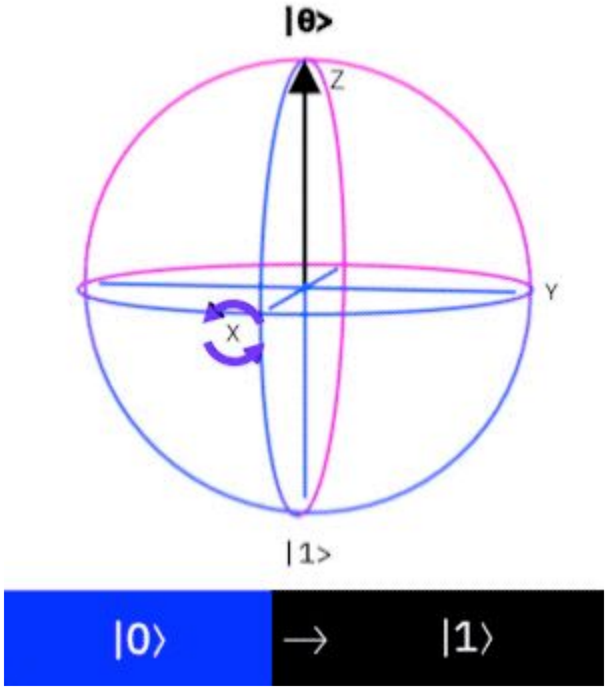
Single qubit gates

X-gate

Now that we have single-qubit states, we can define some operations on them. Some of the very useful single-qubit gates are the Pauli matrices (or gates as we call them in gate based Quantum Computing).

$$\mathbf{X} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

$$X|\psi\rangle = \beta|0\rangle + \alpha|1\rangle$$



Reminder:

$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$, $|1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$, and $|\psi\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}$

Source: https://lewisla.gitbook.io/learning-quantum/quantum-circuits/single-qubit-gates

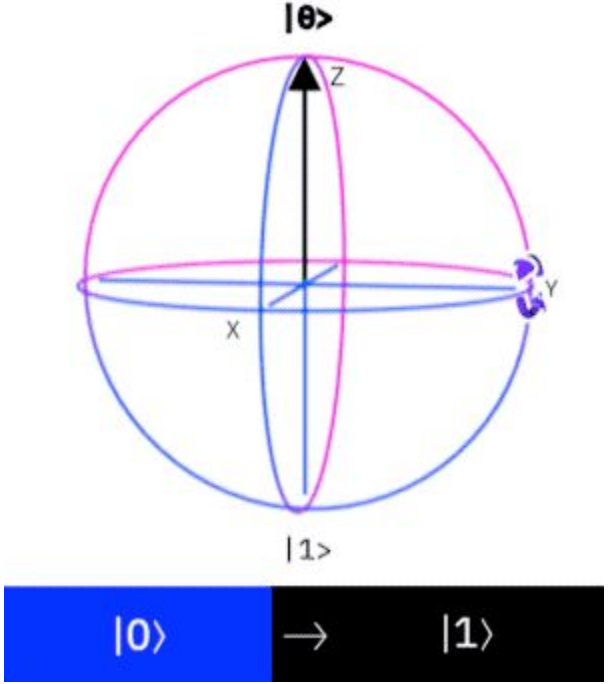
Single qubit gates

Y-gate

Now that we have single-qubit states, we can define some operations on them. Some of the very useful single-qubit gates are the Pauli matrices (or gates as we call them in gate based Quantum Computing).

$$\mathbf{Y} = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$$

$$Y\,|\psi\rangle = -i\beta\,|0\rangle + i\alpha\,|1\rangle$$



Source: https://lewisla.gitbook.io/learning-quantum/quantum-circuits/single-qubit-gates

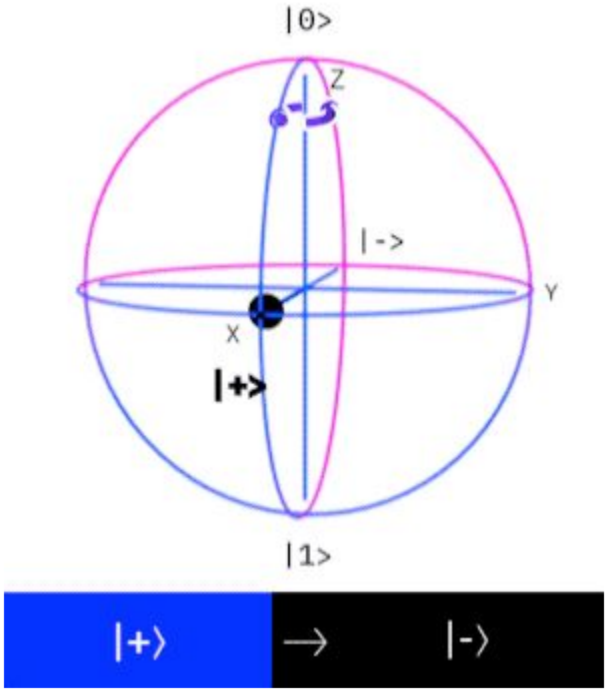
Single qubit gates

Z-gate

Now that we have single-qubit states, we can define some operations on them. Some of the very useful single-qubit gates are the Pauli matrices (or gates as we call them in gate based Quantum Computing).

$$\mathbf{Z} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

$$Z\,|\psi\rangle = \alpha\,|0\rangle - \beta\,|1\rangle$$



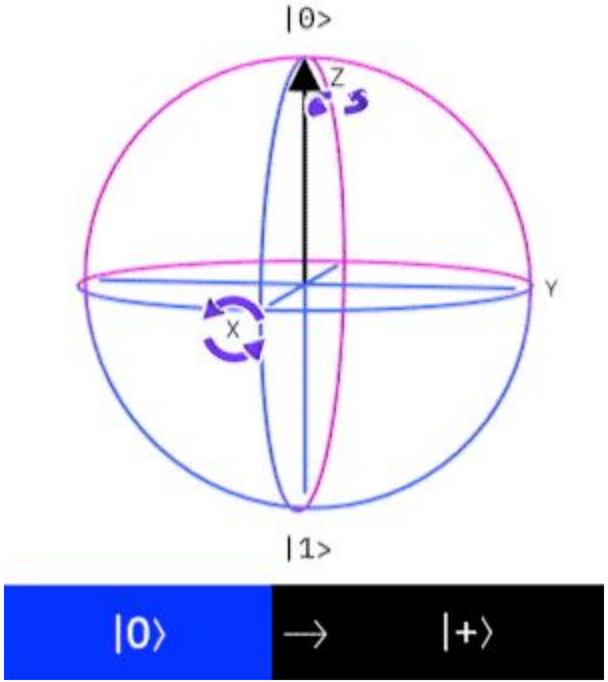
Source: https://lewisla.gitbook.io/learning-quantum/quantum-circuits/single-qubit-gates

Single qubit gates

Hadamard gate

Another example to single-qubit gates is the Hadamard gate:

$$\mathbf{H} = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1 \\ 1 & -1\end{bmatrix}, \; H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$



Source: https://lewisla.gitbook.io/learning-quantum/quantum-circuits/single-qubit-gates

Single qubit gates

Summary

Now that we have single-qubit states, we can define some operations on them. Some of the very useful single-qubit gates are the Pauli matrices (or gates as we call them in gate based Quantum Computing).

$$\mathbf{X} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \qquad \mathbf{Y} = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \qquad \mathbf{Z} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

$$X|\psi\rangle = \beta|0\rangle + \alpha|1\rangle \qquad Y|\psi\rangle = -i\beta|0\rangle + i\alpha|1\rangle \qquad Z|\psi\rangle = \alpha|0\rangle - \beta|1\rangle$$

Another example to single-qubit gates is the Hadamard gate:

$$\mathbf{H} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix},\ H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

Parametrized gates

We can also define gates that are parametrized. These gates will be very useful for some algorithms.

Rotation around the y-axis,

$$R_Y(\theta) = e^{(-i\frac{\theta}{2}Y)} = \begin{bmatrix} cos(\theta/2) & -sin(\theta/2) \\ sin(\theta/2) & cos(\theta/2) \end{bmatrix}$$

$$R_Y(\theta)\,|0\rangle = cos(\theta/2)\,|0\rangle + sin(\theta/2)\,|1\rangle$$

or a generic rotation,

$$U3(\theta, \phi, \lambda) = \begin{bmatrix} cos(\theta/2) & -e^{i\lambda} sin(\theta/2) \\ e^{i\phi} sin(\theta/2) & e^{i(\phi+\lambda)} cos(\theta/2) \end{bmatrix}$$

Multi qubit states

Two or more qubit basis states are expressed with tensor products. e.g.,

$$|00\rangle = |0\rangle \otimes |0\rangle = \begin{bmatrix}1\\0\end{bmatrix} \otimes \begin{bmatrix}1\\0\end{bmatrix} = \begin{bmatrix}1\\0\\0\\0\end{bmatrix}$$

or more generally,

$$|\psi_1\rangle \otimes |\psi_2\rangle = \begin{bmatrix}\alpha\\\beta\end{bmatrix} \otimes \begin{bmatrix}\gamma\\\lambda\end{bmatrix} = \begin{bmatrix}\alpha\gamma\\\alpha\lambda\\\beta\gamma\\\beta\lambda\end{bmatrix}$$

Two qubit gates

Controlled-NOT (CX, CNOT) gate is one of the most common two-qubit gates. It is defined with a 4x4 matrix, since it is a 2 qubit gate.

$$\text{CX} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

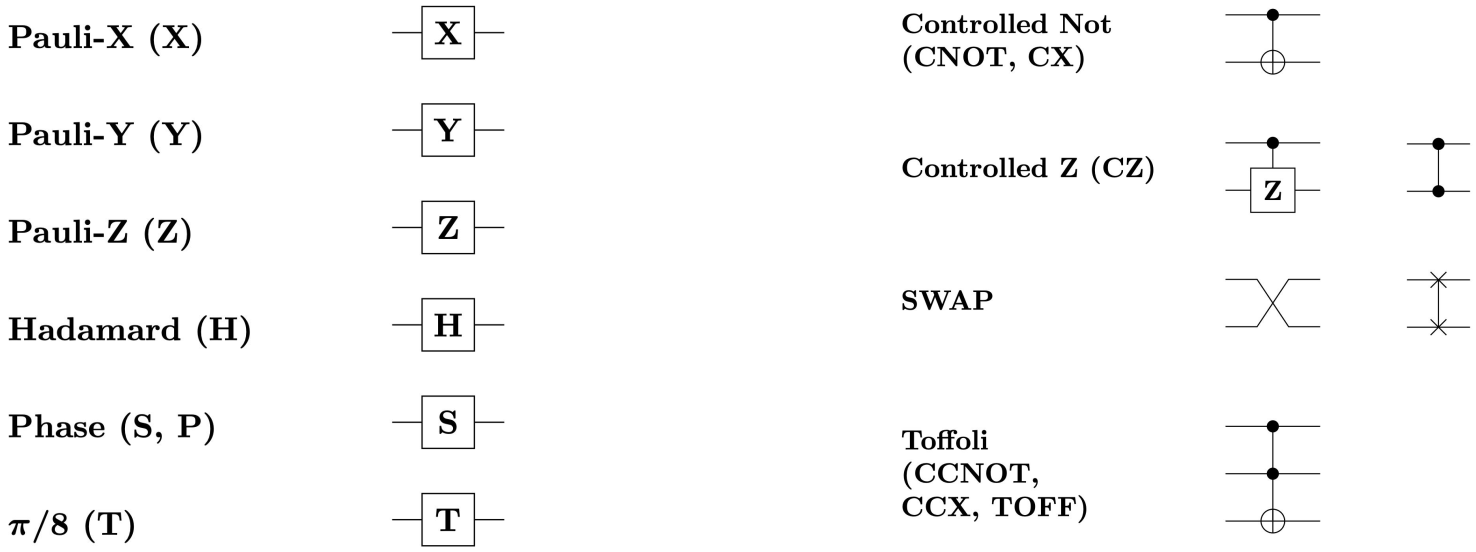
Then let's apply the CX gate to $|01\rangle$ and $|10\rangle$ states as an example;

$$\text{CX}\,|01\rangle = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} = |01\rangle$$

$$\text{CX}\,|10\rangle = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = |11\rangle$$

Gate-based Quantum Computing

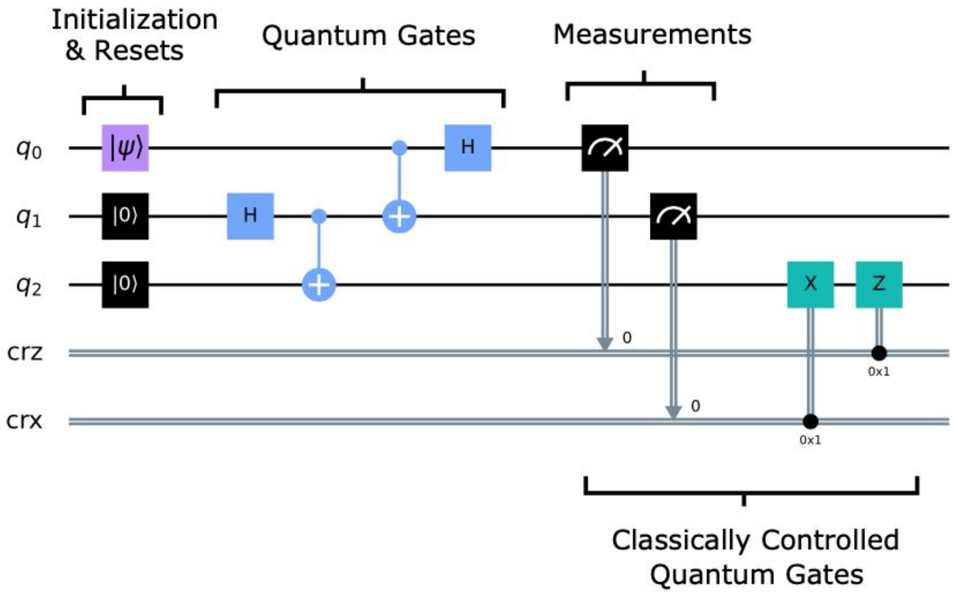
In gate based quantum computing operations are defined using quantum gates. This allows us to construct quantum circuits, which can be visualized easily.



source: https://en.wikipedia.org/wiki/Quantum_logic_gate

Quantum Circuits

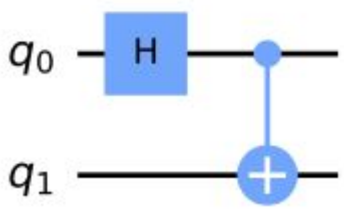
An example Quantum Circuit



Source: https://qiskit.org/textbook/ch-algorithms/defining-quantum-circuits.html

Constructing some well known states

Bell state

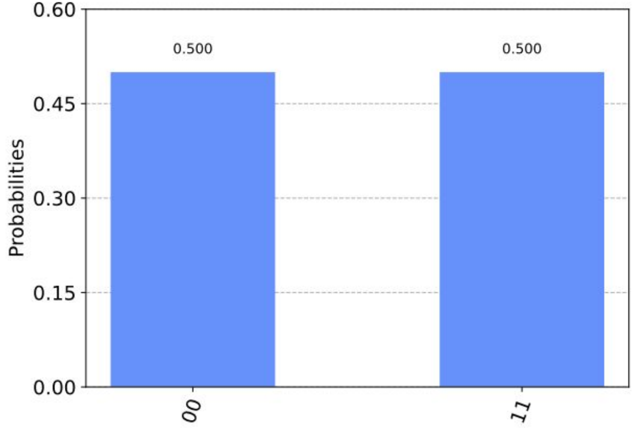


$$|\psi_1\rangle\,(H \otimes \mathbb{1})\,|00\rangle = \frac{1}{\sqrt{2}}\left(|00\rangle + |10\rangle\right)$$

$$|\psi_2\rangle = CX\,|\psi_1\rangle = \frac{1}{\sqrt{2}}\left(|00\rangle + |11\rangle\right)$$

Ideal expectation:

P(00) = 0.5, P(11) = 0.5

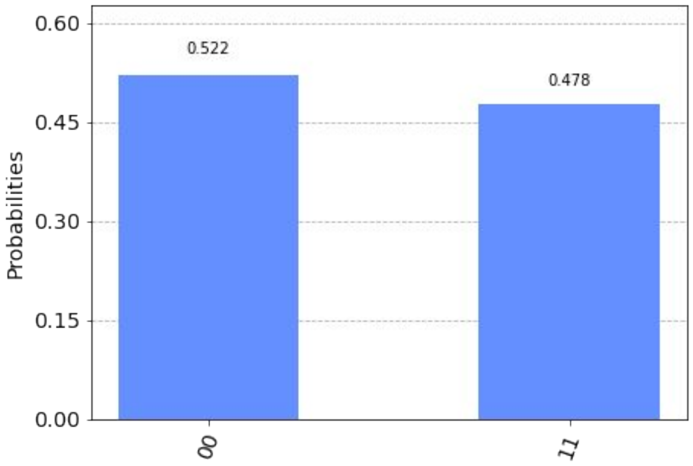


Constructing some well known states

Bell state

Using a Quantum Computer

Results from an *ideal* Quantum Computer simulation:

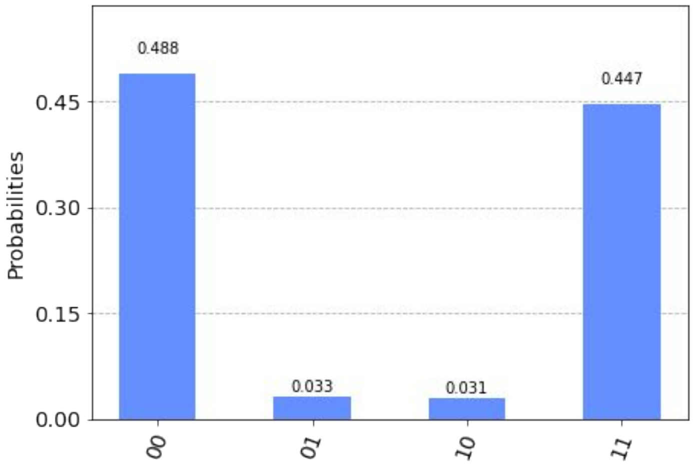


To reconstruct state probabilities we need to take a *finite* number of measurements. We call this number "shots". Then the standard error defines a precision (ϵ) such that,

$$\epsilon = \frac{1}{\sqrt{shots}}$$

Using a *real* Quantum Computer

Results from hardware:



Now we are not only observing finite sampling error but also errors caused by the not perfect hardware.

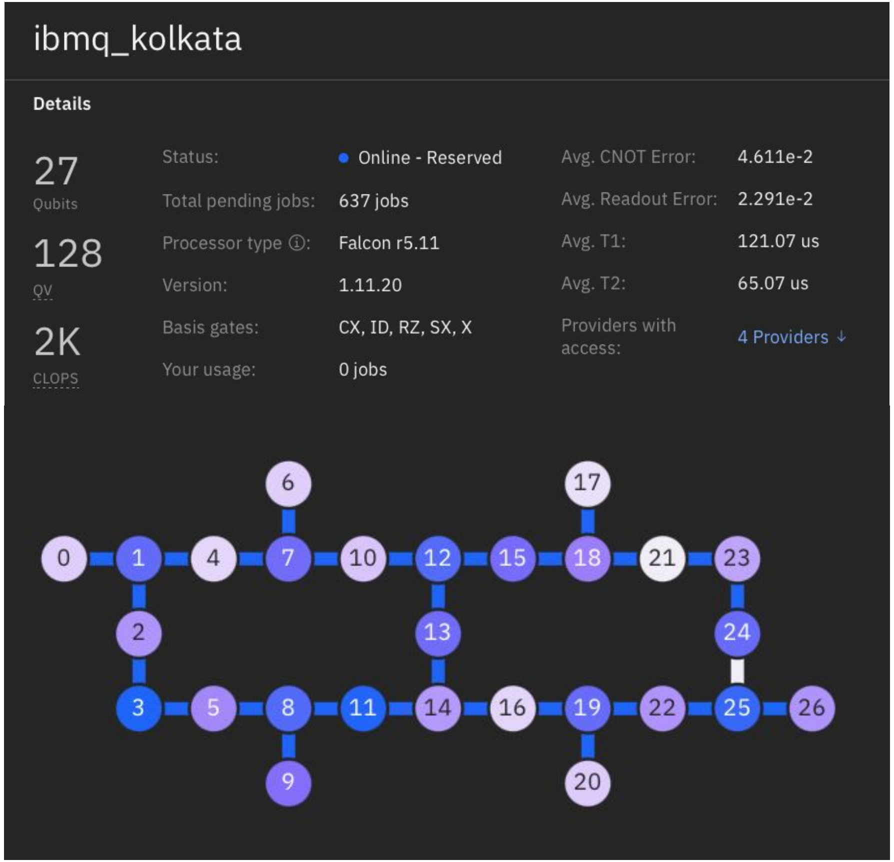
This is a challenge for current Quantum Computers.

What are the challenges?

- Gate execution time
- Coherent and incoherent noise
- Low number of qubits
- Qubit connectivity

Different types of hardware exist, all of them have their pros and cons.

We have privileged access to IBM’s superconducting qubit devices.



Exciting Applications with Quantum Computers

- Grover's search algorithm
- Shor's factoring algorithm

Classical optimization problems:

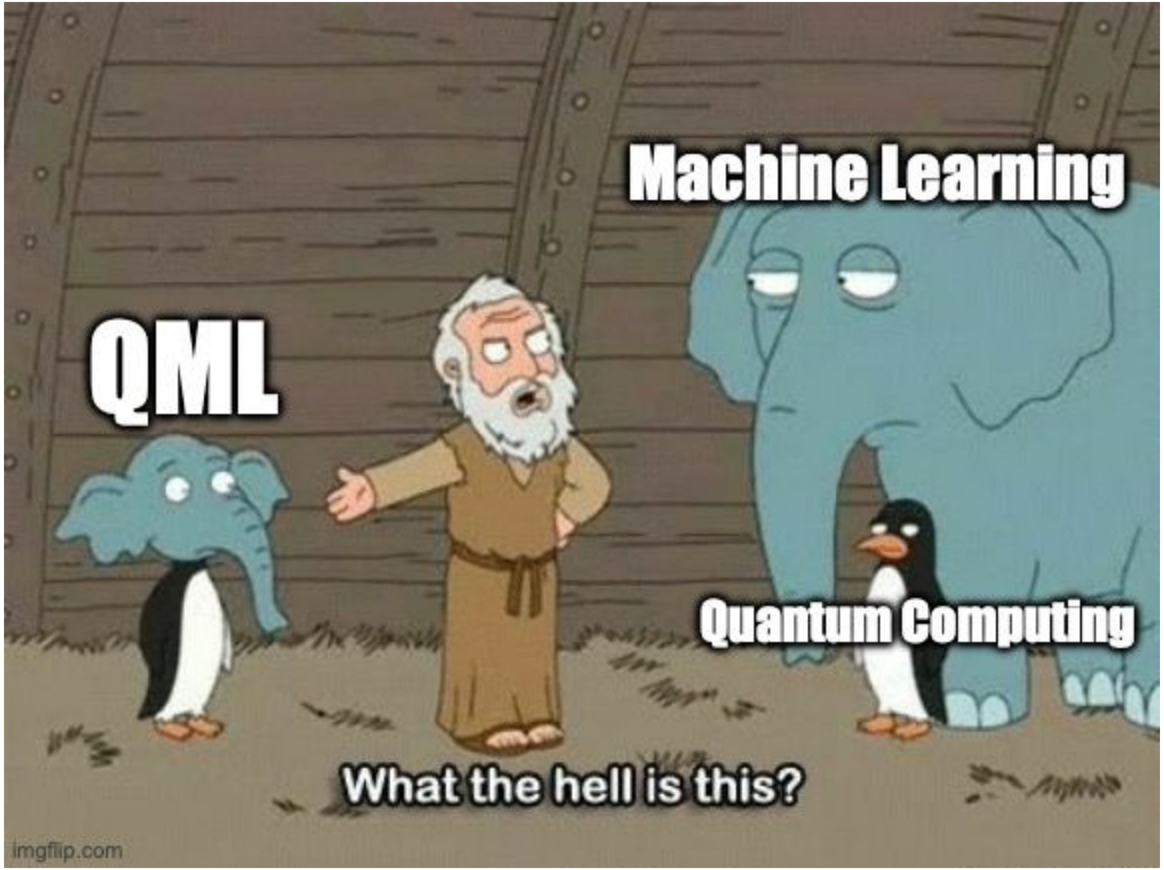
- Quantum Approximate Optimization Algorithm (QAOA)

Simulating Physics:

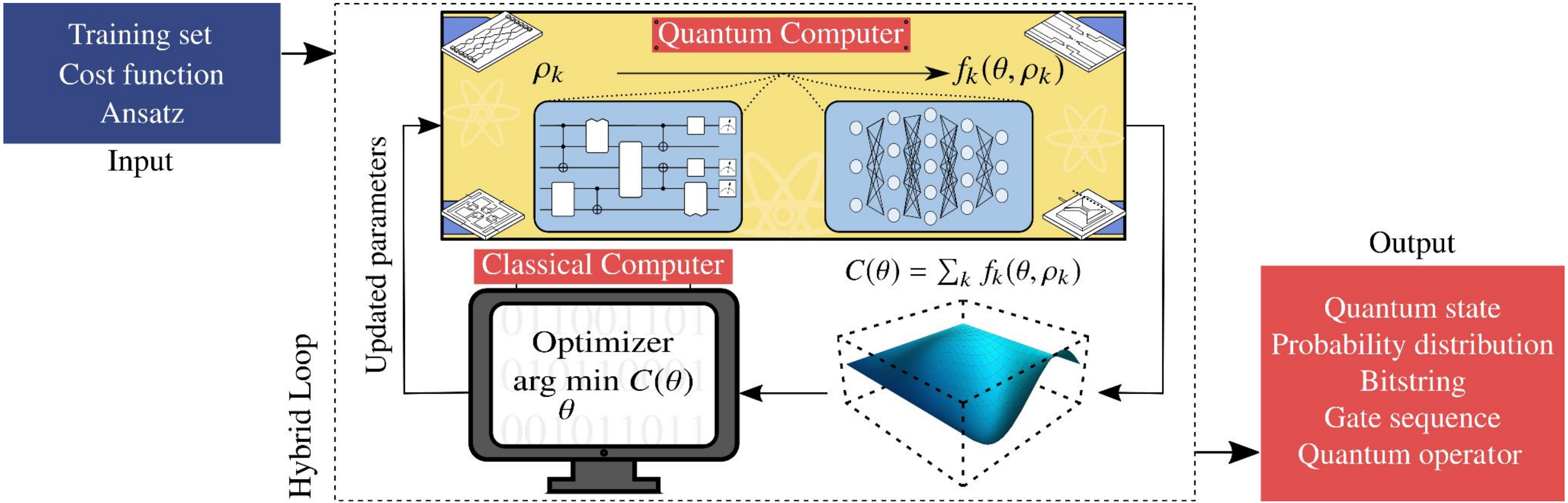
- Variational Quantum Eigensolver (VQE)
- Quantum dynamics with trotterization

Quantum Machine Learning

- Can we improve classical machine learning?
- Can we use Quantum Computers for machine learning of quantum data?


Machine Learning
QML
Quantum Computing
What the hell is this?
imgflip.com

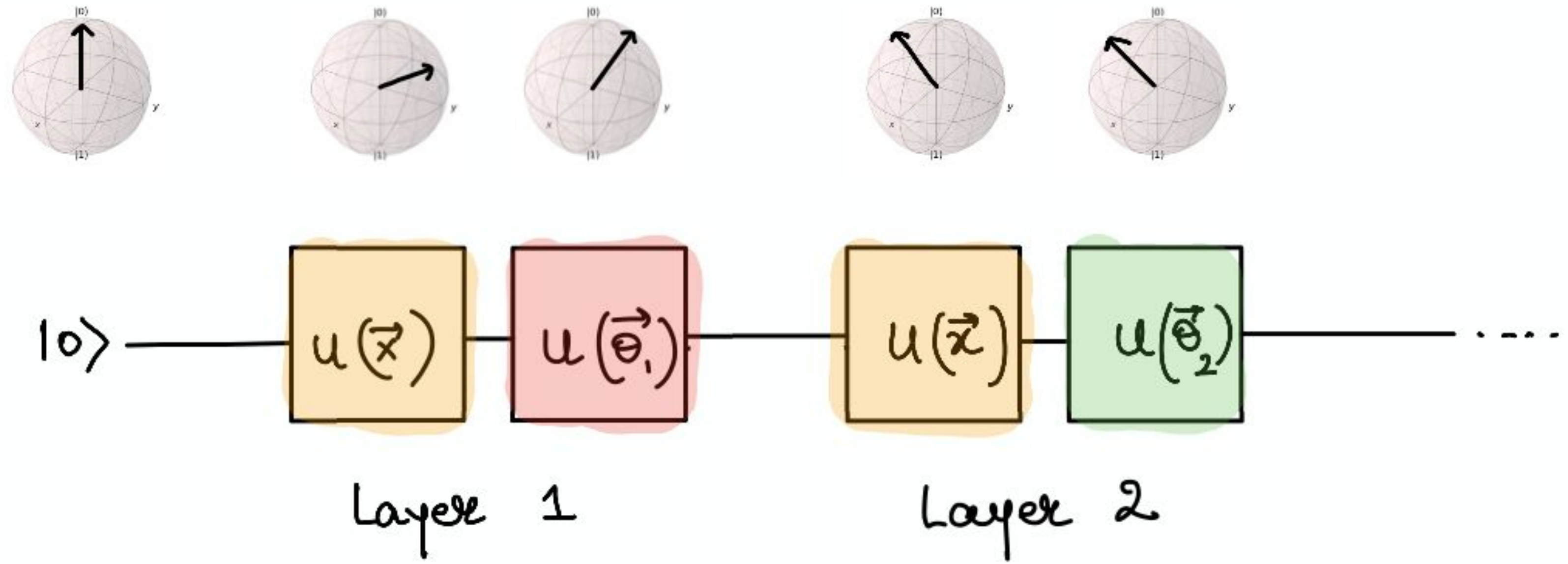
Quantum Machine Learning (QML)



Cerezo et al. Nature Reviews Physics 3, 625–644 (2021)

Quantum Neural Networks

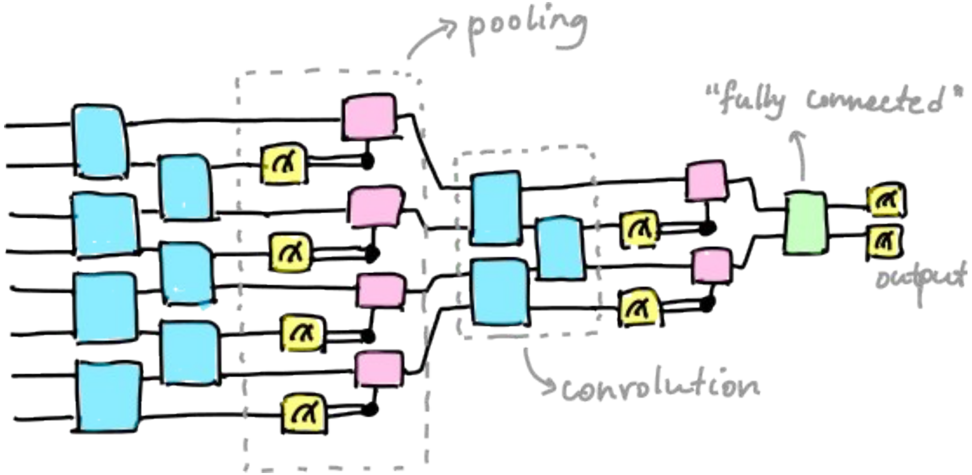
Data re-uploading models



https://pennylane.ai/qml/demos/tutorial_data_reuploading_classifier/

Quantum Neural Networks

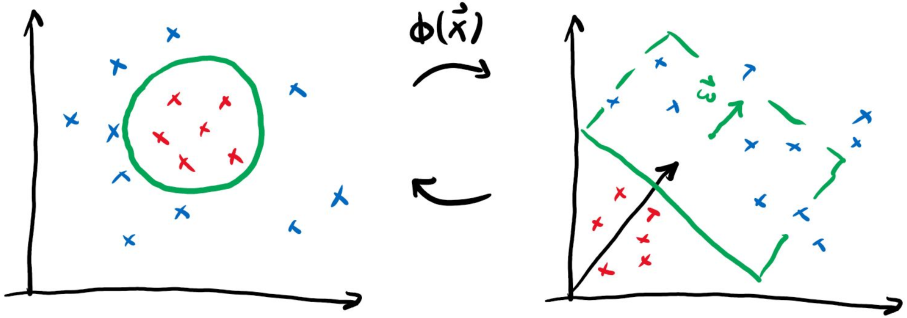
Quantum Convolutional Neural Networks



https://pennylane.ai/qml/glossary/qcnn/

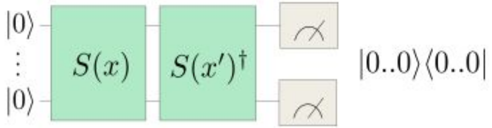
Quantum Machine Learning

Quantum Kernel Methods



https://pennylane.ai/qml/demos/tutorial_kernels_module/

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j) \rangle$$



https://pennylane.ai/qml/demos/tutorial_kernel_based_training/

What QML promises?

Article | Open access | Published: 11 May 2021

Power of data in quantum machine learning

Hsin-Yuan Huang, Michael Broughton, Masoud Mohseni, Ryan Babbush, Sergio Boixo, Hartmut Neven & Jarrod R. McClean

Nature Communications **12**, Article number: 2631 (2021) | Cite this article

There exists separations of quantum and classical kernels and these separations are also data dependent.

Article | Published: 24 June 2021

The power of quantum neural networks

Amira Abbas, David Sutter, Christa Zoufal, Aurelien Lucchi, Alessio Figalli & Stefan Woerner

Nature Computational Science **1**, 403–409 (2021) | Cite this article

Quantum Neural Networks have more *capacity* compared to classical Neural Networks with *same* number of parameters.

Article | Open access | Published: 22 August 2022

Generalization in quantum machine learning from few training data

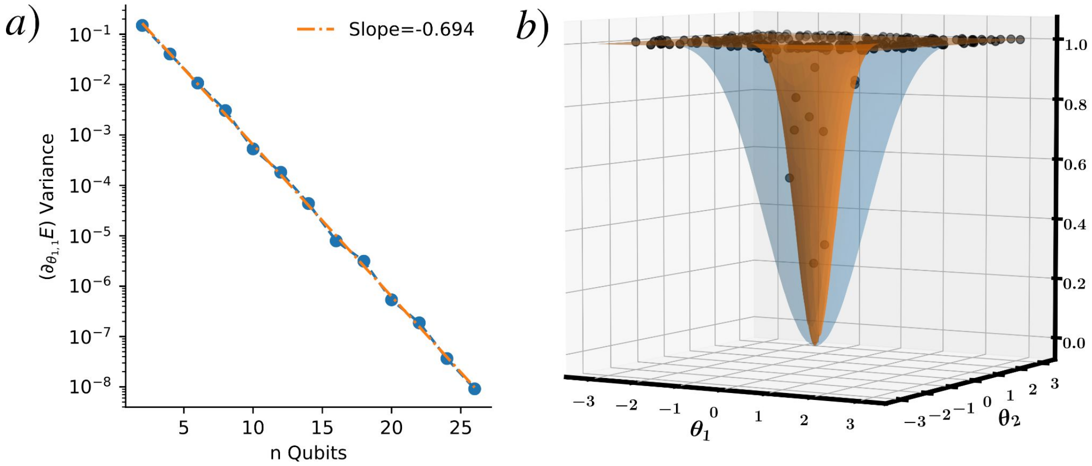
Matthias C. Caro, Hsin-Yuan Huang, M. Cerezo, Kunal Sharma, Andrew Sornborger, Lukasz Cincio & Patrick J. Coles

Nature Communications **13**, Article number: 4919 (2022) | Cite this article

Quantum Neural Networks are good at generalization (needs less data) compared to classical models.

Roadblocks in QML

Barren Plateaus

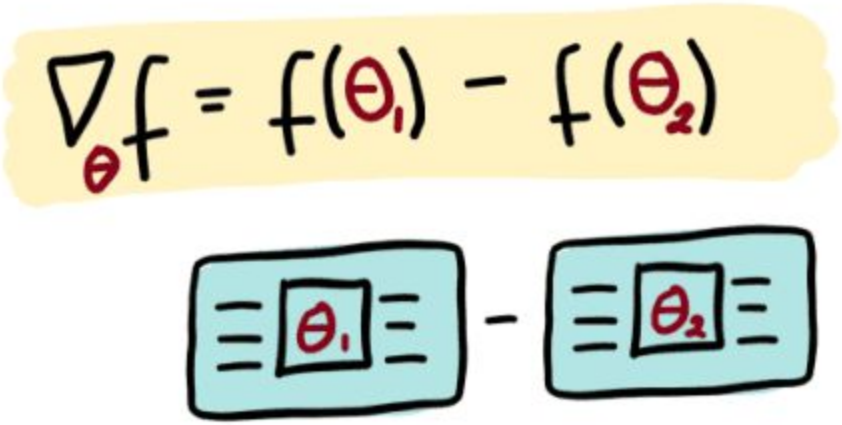


Cerezo et al. Nature Reviews Physics 3, 625–644 (2021)

Gradients of parametrized quantum circuits concentrate around zero and vanish exponentially in system size if the model is too *expressive*.

Roadblocks in QML

Lack of backprop-like gradient computation



https://pennylane.ai/qml/glossary/parameter_shift/

Gradients of parametrized quantum circuits are computed using parameter shift rules, which requires 2*p* times execution of the circuit, where *p* is number of parameters.

Cenk Tüysüz
DESY CQTA

cenk.tueysuez@desy.de

@cenk_tuysuz

Contact

DESY. Deutsches
Elektronen-Synchrotron

www.desy.de