Quantum Machine Learning: A Gentle Introduction

ECAI 2025 – Tutorial 17

Dr. Simon Caton & Dr. Steve Campbell
simon.caton@ucd.ie, steve.campbell@ucd.ie

School of Computer Science, University College Dublin, Ireland
School of Physics, University College Dublin, Ireland
UCD Centre for Quantum Engineering, Science, and Technology (C-QuEST)

General Comments

- We will brush over a lot of the theoretical background
- To go further you'll need to catch up on this background (we'll list some resources at the end)
- We're aiming to be quite hands on: so you can continue yourself afterwards
- Please interrupt as much as you like
- This content comes from a graduate course on QML at UCD, were going to cover the first $\approx$15 hours of content (and $\approx$10-20 hours of the prerequisite content)

Common Discussions Points, for Scoping

Academic Perspective

Do I [who works in domain X] need to pay attention to this quantum stuff, and if so, where do I start?

Industry Perspective

We have a project sponsor [in domain X] with some budget, we want to do something with quantum, but we don't know what that should be...

Overview

Anchoring QML within the Standard ML Process

Quantum Computing – a brief minimum

Training a Variational Model or QNN

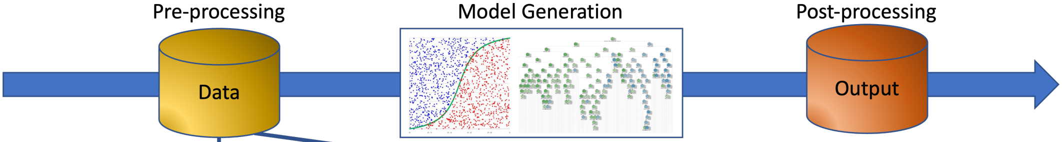
Applying a QNN to the Kaggle Titanic dataset

Tutorial Summary

Getting into Quantum

Anchoring QML within the Standard ML Process

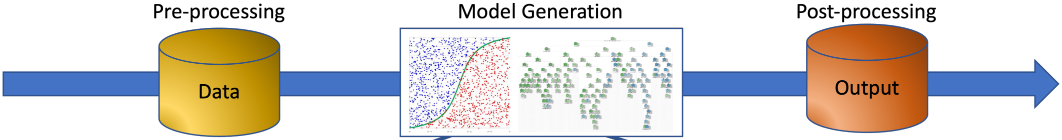
The Machine Learning Process



Generally preparing and understanding the data

Typical activities:
- Data Cleaning
- Feature Engineering
- Dimensionality Reduction
- Sampling the Data: training, validation and testing

The Machine Learning Process



Fitting the model to the input data

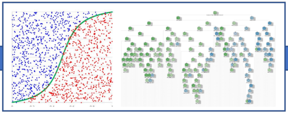
Often:
- optimization to improve the "fit" of the model
- loss- or error-based (supervised); metric-based (unsupervised)

The Machine Learning Process

Pre-processing

Data

Model Generation



Post-processing

Output

Typical activities:
- evaluating Performance
- classifying something
- emitting values

Output tends to be:
- groups of data,
- a category or class,
- forecasted values

Why add Quantum Computing?

- ML is the process of making computers learn from data
- Quantum computing describes information processing with devices based on quantum theory
- The expectation (hope?) is that ML will benefit from speed-up through quantum technologies
- This hasn't happened yet, but QNNs are a truly novel form of ML, which makes their study interesting for that reason alone.

What is Quantum Machine Learning?

In its broadest form: Approaches that use synergies between machine learning and quantum information.

To be a little more precise

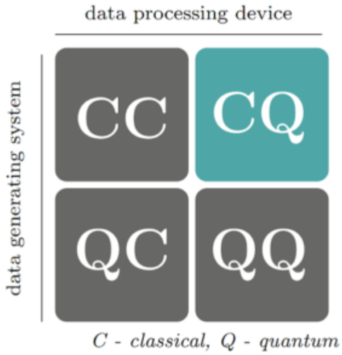
Quantum Machine Learning is the development of Machine Learning approaches with the use of (or assistance from) Quantum Computers [5].

QML Challanges

QML is facing an identity crisis at the moment:

- The barren plateau problem [3] is inhibiting the ability to build big models
- The challenge of good data representations [1]
- This obsession of forcing classical data into a quantum representation: absence of real quantum data

Combining QC & ML

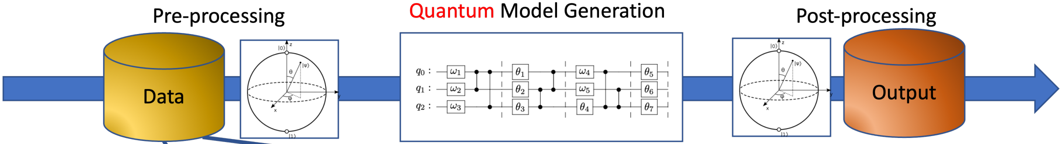


CQ: Classical data, Quantum computation

What we mean when we say QML today.

Image credit: [5]

The Quantum Machine Learning Process

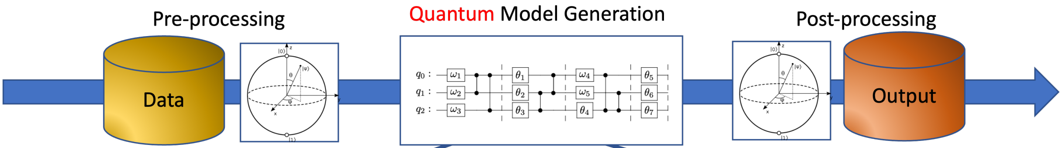


Generally preparing and understanding the data

Typical activities:

- Data Cleaning
- Feature Engineering
- Dimensionality Reduction
- Sampling the Data: training, validation and testing
- Transforming the data into a quantum representation

The Quantum Machine Learning Process



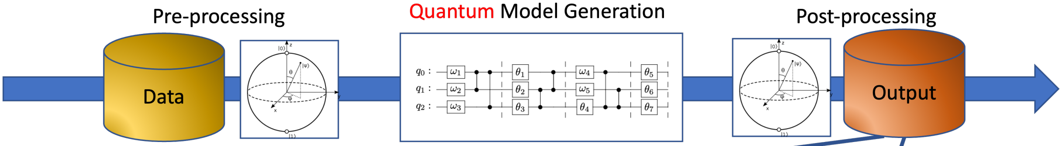
Fitting the quantum model to the input quantum data

Often:

- some form of classical optimization to tune the quantum algorithm
- loss- or error-based (supervised); metric-based (unsupervised)
- multiple algorithm observations per instance to generate a probability distribution

Common use of variational algorithms: parameterized quantum gates

The Quantum Machine Learning Process



Typical activities:
- evaluating performance
- classifying something
- emitting values

Output a bit string that needs to be mapped from a quantum representation to a classical one

High-level Summary

The development of Machine Learning (or Optimisation) approaches with the use of (or assistance from) Quantum Computers [5].

This typically means:

- preparing the input data in a "quantumly appropriate" manner
- developing a quantum representation of the problem: a quantum circuit (i.e. program)
- time-evolving the quantum mechanical system represented by the circuit such that its measurement is the (optimal) solution

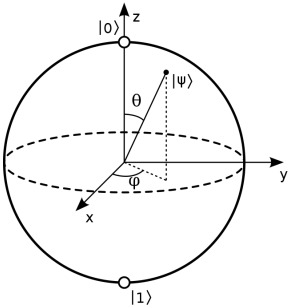
Quantum Computing – a brief minimum

Notion of Data

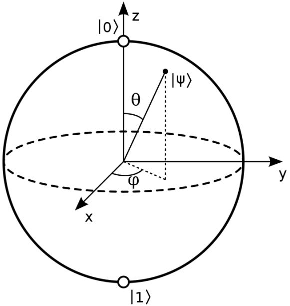
Can come from two sources or contexts:

1. Experimental data already has a quantum representation: QQ
2. Classical data to be transformed into a quantum representation: CQ

We need a "good" quantum representation for any data or the model cannot learn.



Qubits



We represent the logical space with two vectors: $|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ and $|1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$

These vectors form a basis over which we can describe any quantum state

$$|\Psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle \;\; (|\alpha|^2 + |\beta|^2 = 1; \alpha, \beta \in \mathbb{C})$$

Quantum superposition $\rightarrow$ classical bit cannot be in such a state.

Quantum Gates & Circuits (I)

A quantum gate corresponds to a unitary operation (rotation) of the quantum state.

$$|\Psi\rangle \;-\!\!\!-\!\!\!-\!\!\!-\boxed{U}-\!\!\!-\!\!\!-\!\!\!-\; |\Psi'\rangle$$

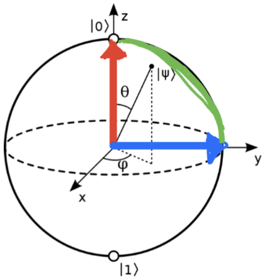
input vector—gate—output vector

Quantum Gates & Circuits (II)

E.g. the Hadamard Gate: $-\boxed{H}-$

$$= \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \to H|0\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}\begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

$$= \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ 1 \end{pmatrix} = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

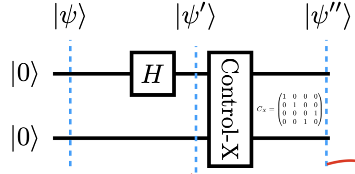


Multi-Qubit Circuits

Four-dimensional column vectors

The Hilbert space for n-qubits has: $\dim = 2^n$

e.g. for two-qubits: $\dim = 2^2 = 4 = \{|00\rangle,\ |01\rangle,\ |10\rangle,\ |11\rangle\}$



$$
\begin{aligned}
|\psi\rangle &= |0\rangle \otimes |0\rangle \\
&= \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} \\
&= \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}
\end{aligned}
$$

$$
\begin{aligned}
|\psi'\rangle &= (H \otimes \mathbb{1})|\psi\rangle \\
&= \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes |0\rangle \\
&= \frac{1}{\sqrt{2}}(|00\rangle + |10\rangle)
\end{aligned}
$$

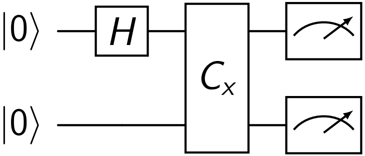
$$
\begin{aligned}
|\psi''\rangle &= C_X|\psi'\rangle \\
&= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix} = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)
\end{aligned}
$$

(Entangled state)

Read out (Measurement)

Measurements are special in quantum mechanics

$\rightarrow$ “collapse of the wavefunction”

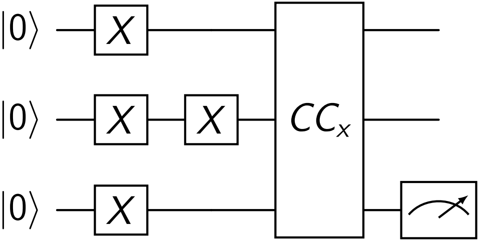


Since $|\Psi''\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ the only possible outcomes are finding both qubits in the same state.

But we get the same outcomes if we only measure one qubit due to the entanglement this circuit creates.

Not all circuits will lead to entangled states.

A Quantum "if": hardcoding a Titanic Prediction



$= \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ (qu)bit flip

CC_x gate: controlled-control-X gate

- First layer of X-gates: flips all qubits to $|1\rangle$
- Second layer: encodes the "if" and flips qubit 2 again
- CC_x: encodes the outcome onto qubit 3 (no entanglement)

Quantum Programs – expressed as a circuit (I)

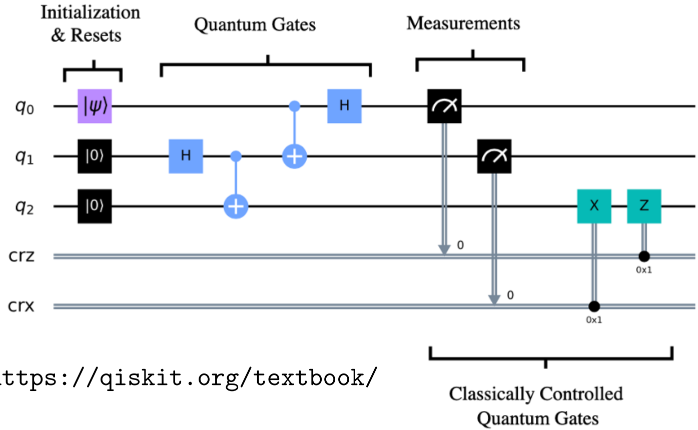


Image Source: `https://qiskit.org/textbook/`

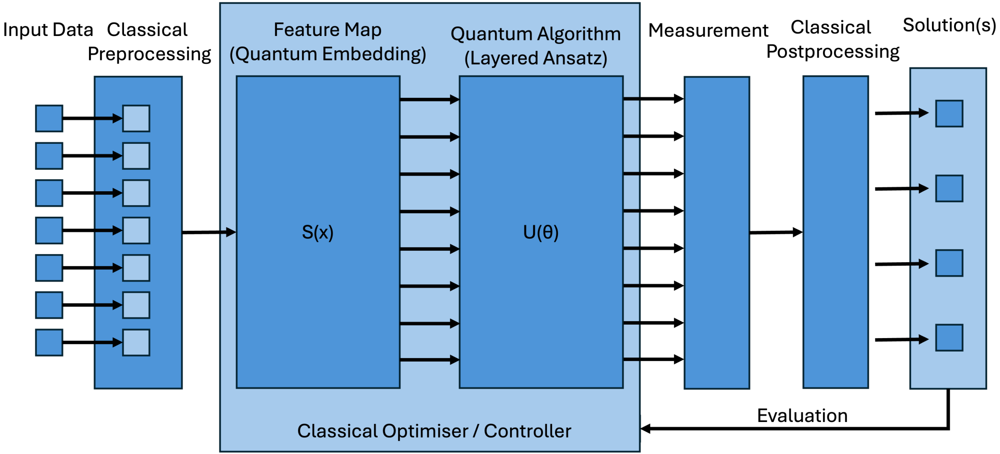
Quantum Programs – expressed as a circuit (II)

The time-evolution (left to right) of the quantum system.

- qubits as wires in the figure
- single qubit gates manipulate the Hilbert space (rotations around x, y, or z) they have classical parameters that either load or manipulate data
- two qubit gates (often) create entanglement to increase the functional expressivity of the model
- measurements collapse the quantum wave function to observe the state $|0\rangle$ or $|1\rangle$ with some probability (amplitude) α

Training a Variational Model or QNN

General Process



State Preparation

- We always starts with putting the quantum computer into a specific state: data loading for QML
- It's hard to do well, but do it badly and the data is unusable [5]
- The quantum representation of the data has significant leverage over program design / model architecture
- This "input problem" [1] is key to "unlocking" quantum technologies

“Quantum Neural Networks” (I)

Name stems from:

- the multi-layer perception of the Ansatz being similar to layers in a neural network
- the borrowing of some of the tools from deep learning
- the parameters (θ_k) are analogous to model weights
- quantum gates being non-linear transformations in a feature space

"Quantum Neural Networks" (II)

- the idea of the encoding strategy is something analogous to an embedding
- the time-evolution (through parameterised Pauli rotations) is similar to the idea of single hidden layer NN
- we can think of the non-linear (feature) transformations as something similar to an activation function: $v \to \psi(v)$

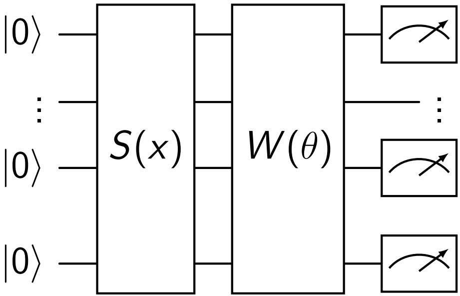
Despite all this, QNNs and NNs are fundamentally different forms of machine learning

Key steps

Three main stages to the ML process

1. Input encoding circuit – often (but not always) as parameterised Pauli Rotations – often called a Feature Map
2. Processing circuit – the Ansatz (or template of computation) defines the circuit architecture: repeated layers of parameterised rotation and entanglement operations
3. Output decoding – turning measurements into a probability distribution by executing the circuit k times – called shots

Visual Interpretation



Where:

- $S(x)$ is our feature map or quantum embedding
- $W(\theta)$ is our parameterised processing circuit (ansatz)

We can define our circuit as $U(x,\theta) = W(\theta)S(x)$

We train end-to-end either in simulation, or via the cloud.

Model Architecture (I)

Just like regular machine learning, we need to design our model architecture, encoding and sampling strategies.

The idea of an ansatz, is that of a template, i.e. some architecture that has previously shown success.

Thus, there are libraries of ansätze to start with.

The no free lunch theory (or *ansatz lottery*) of machine learning applies when making design choices, and the design space is huge.

Model Architecture (II)

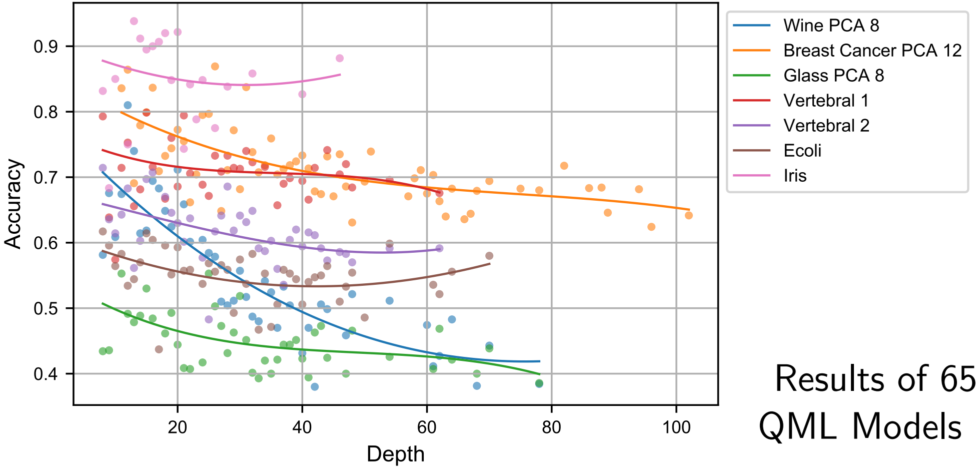
The emerging area of (neural) quantum architecture search seeks to automate this somewhat. (It's kind of like AutoML for quantum)

In [6] we saw how hard it is to randomly design a "good" QNN, even with a fixed structural guide.

Circuits need to be "lean": not too deep, not too many qubits

- due to NISQ noise (error)
- to stay "simulatable", and
- to avoid issues with gradients (see [3])

Model Architecture (III)



Results of 6500 QML Models [6]

Applying a QNN to the Kaggle Titanic dataset

Tutorial Summary

Summary of Tutorial

Example-driven Notebook of Applying QNNs to the Titanic, supported by:

- Minimum of Quantum Information Theory
- General Background into QNNs
- Exploring the Training Process
- Small excursion into the effects of Quantum Noise and Cloud Accessible Devices
- Mini discussion on baselines

Quantum AI Projects @UCD

- Exploring Applications of Deep Learning for Quantum Architecture Search [7, 8]
- Reinforcement Learning for the Precise Control of Quantum Systems [2]
- Designing Efficient Formulations of Quantum QUBO Algorithms for Bioinformatics Applications [4]
- Design and Optimisation of Quantum Machine Learning Approaches [6]
- 13.7M Euro QUBIC project announced yesterday: see here

Getting into Quantum

Getting into Quantum

Depends on your background, and time commitment.

Basic topics are:

- Mechanics (essentially high-school maths and physics)
- Quantum Mechanics (undergrad physics)
- Quantum Computing and Quantum Information – both theory and core computer science
- Applied Quantum Computing – Machine Learning / Optimisation etc.

Mechanics and Quantum Mechanics

The theoretical minimum (Stanford): video lectures and corresponding books here

James Binny's Lectures and Recordings (Oxford): here

Strang's Linear Algebra (and related videos): here

Khan Academy (e.g. here)

Feynman Lectures (especially Vol III, but also others) here

Quantum Computing

Quantum Computing for Everyone (Bernhardt) – here

IBM Qiskit Book and YouTube Channel – book, book replacement, and YouTube Channel

Quantum Computation and Quantum Information (Nielsen and Chuang) – here

Introduction to Classical and Quantum Computing (Wong) – here

Quantum AI

Machine Learning with Quantum Computers (Schuld and Petruccione) – here

PennyLane Demos (QAOA for example) – here

Qiskit QML Demos – here

D-Wave's Annealing and General Intro – here, but costly!

References (I)

[1] Jacob Biamonte et al. "Quantum machine learning". In: *Nature* 549.7671 (2017), pp. 195–202 (cit. on pp. 11, 29).

[2] Haftu W Fentaw, Steve Campbell, and Simon Caton. "Exploring quantum control landscape and solution space complexity through optimization algorithms and dimensionality reduction". In: *Scientific Reports* 15.1 (2025), p. 14605 (cit. on p. 40).

[3] Jarrod R McClean et al. "Barren plateaus in quantum neural network training landscapes". In: *Nature communications* 9.1 (2018), p. 4812 (cit. on pp. 11, 35).

References (II)

[4] Namasi G. Sankar, Georgios Miliotis, and Simon Caton. "Scalable Quantum Optimisation using HADOF: Hamiltonian Auto-Decomposition Optimisation Framework". In: *3rd International Workshop on AI for Quantum and Quantum for AI (AIQxQIA 2025), at the 28th European Conference on Artificial Intelligence (ECAI)*. 2025 (cit. on p. 40).

[5] Maria Schuld and Francesco Petruccione. *Machine learning with quantum computers*. Springer, 2021 (cit. on pp. 10, 12, 16, 29).

[6] Patrick Selig et al. "A Case for Noisy Shallow Gate-based Circuits in Quantum Machine Learning". In: *2021 International Conference on Rebooting Computing (ICRC)*. IEEE. 2021, pp. 24–34 (cit. on pp. 35, 36, 40).

References (III)

[7] Patrick Selig et al. "DeepQPrep: Neural Network Augmented Search for Quantum State Preparation". In: *IEEE Access* (2023) (cit. on p. 40).

[8] Patrick Selig et al. "On the Challenges of Quantum Circuit Encoding using Deep and Reinforcement Learning". In: *IEEE Access* (2025) (cit. on p. 40).

UCD
DUBLIN