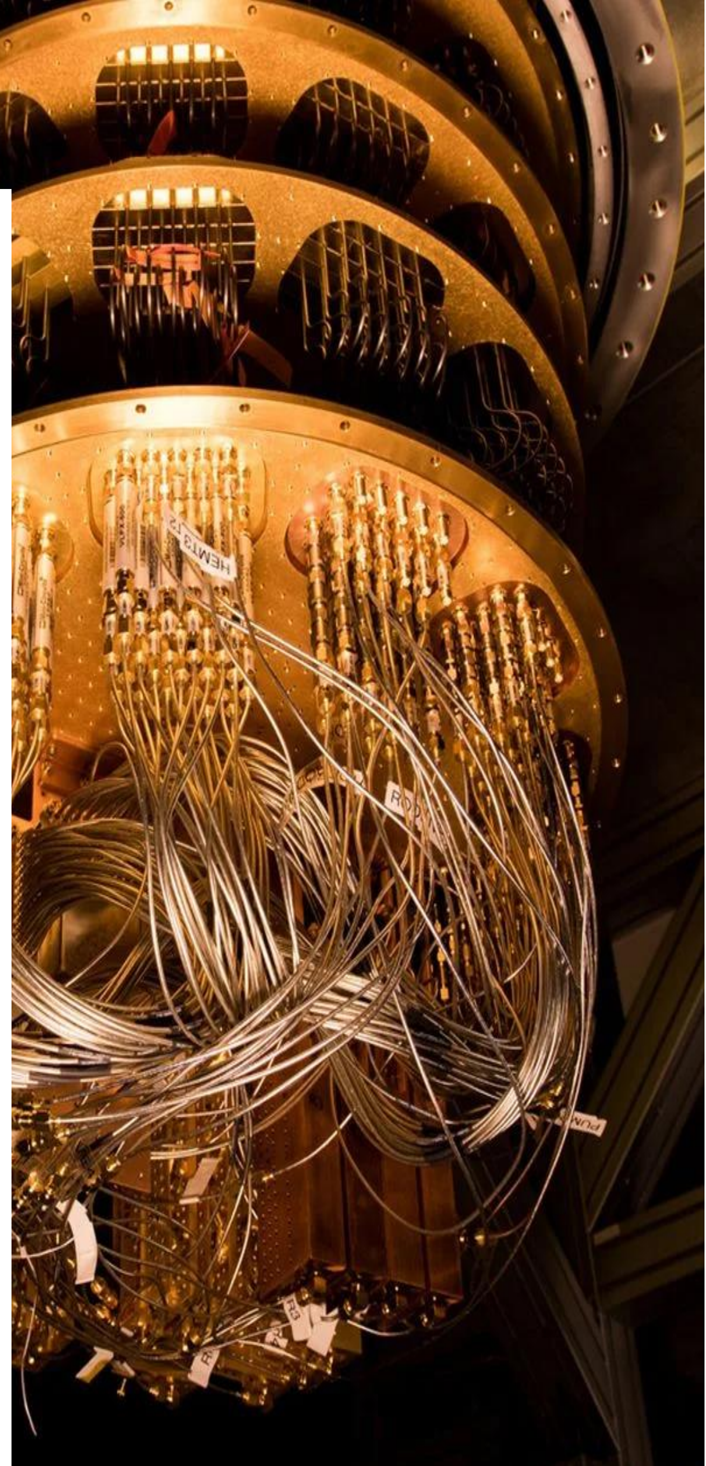
Quantum Machine Learning



Authored by: Anishkumar Dhamelia

STEVENS
INSTITUTE of TECHNOLOGY
THE INNOVATION UNIVERSITY®

Abstract

Quantum Machine Learning is the amalgamation of the two most ambitious fields in Computer Science today; Machine Learning and Quantum Computing.

Machine learning is the study of software applications that can improve their predictions given the appropriate data and experience.

While Machine learning has been around for over half a century, Quantum Computing is a comparatively new topic that aims to replace the classical bits with Quantum bits (Qubit) and use Superposition and Entanglement to perform calculations.

Why is this Useful?

Although there has been a significant increase in the computational power of modern machines, it cannot hold a candle to the increase in data over the same period.

Therefore, the lack of computational power becomes the bottleneck for Machine learning algorithms. Many of which rely on calculations over Big Data.

Since Quantum Computing uses Qubits, which are the linear Superposition of |1> and |0>. Due to this, the Qubits can be at both states simultaneously, allowing true parallelism. Hence, a Qubit register is exponentially faster than a classical register with each new additional bit. This advantage over Classical Computers can be essential to a Machine learning algorithm that has to crunch large entries of raw data over multiple attributes.

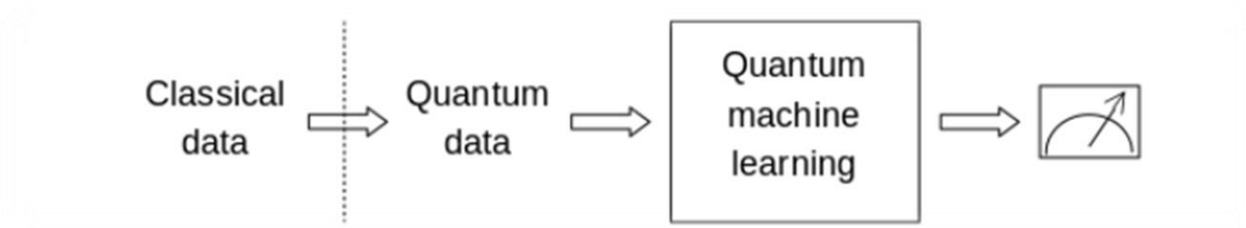
Introduction

Quantum computing is based on phenomena of quantum mechanics, such as superposition and entanglement. Because of the paramount feature of high-speed computing, parallelism can be designed into specific algorithms to solve specific problems. These classical problems usually cannot be solved as efficient as they are in the quantum system.

Quantum computing has an efficient advantage in multi-dimensional systems and multi-variable statistical analysis. Classical systems with a large number of degrees of freedom are difficult to model because of the curse of dimensionality. But quantum parallelism effect allows us to avoid this problem. Therefore, quantum computational resources are very useful to solve different problems with an extra-large number of dimensions (machine learning, for example).

There are many methods associated with data processing and machine learning procedure. Quantum approaches can be applied to basic methods (k-nearest neighbor algorithm, k-means algorithm, principal component analysis and etc.) as well as to sophisticated methods (variational autoencoders, associative adversarial networks and etc.). Here, we demonstrate the applicability of quantum information techniques to both classes of algorithms.

This is the basic structure of Quantum Machine learning.



Background

1) Machine Learning:

Machine learning has mainly three canonical categories of learning—supervised, unsupervised and reinforcement learning. Fundamentally, supervised and unsupervised learning are based on data analysis and data mining tasks. Whereas reinforcement learning is an interaction-based learning, where learning enhances sequentially at every step.

1.1) Supervised Learning:

Supervised learning, also known as supervised machine learning, is a subcategory of machine learning and artificial intelligence. It is defined by its use of labeled datasets to train algorithms that to classify data or predict outcomes accurately. As input data is fed into the model, it adjusts its weights until the model has been fitted appropriately, which occurs as part of the cross-validation process.

1.2) Unsupervised Learning:

Unsupervised learning, also known as unsupervised machine learning, uses machine learning algorithms to analyze and cluster unlabeled datasets. These algorithms discover hidden patterns or data groupings without the need for human intervention. Its ability to discover similarities and differences in information make it the ideal solution for exploratory data analysis, cross-selling strategies, customer segmentation, and image recognition.

2) Qubit and Quantum states:

Like the bit in the classical information, qubit is the fundamental unit of quantum information, which is usually denoted by Dirac notation:

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

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

Unlike classical bits, qubits can exist in a superposition state:

$$|\Psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix},$$

It means that the state $|\Psi\rangle$ is in both $|0\rangle$ and $|1\rangle$ simultaneously, but when it is measured, it will collapse to the state $|0\rangle$ with probability $|\alpha|^2$, or to the state $|1\rangle$ with probability $|\beta|^2$.

2.1) Quantum states:

When we talk about the qubit basis states we implicitly refer to the z-basis states ($|0\rangle$ and $|1\rangle$) as the computational basis states.

The two orthogonal x-basis states are:

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

The two orthogonal y-basis states are:

$$|R\rangle = \frac{|0\rangle + i|1\rangle}{\sqrt{2}} \quad |L\rangle = \frac{|0\rangle - i|1\rangle}{\sqrt{2}}$$

The basis states are located at opposite points on the Bloch sphere representation of the state of a single qubit.

2.2) Quantum Gates:

In the quantum circuit model of computation, the quantum gate is the basic quantum circuit operating on qubits, which is the building block of quantum circuits like the classical logic gate is for conventional digital circuits. Unlike classical logic gates, quantum gates are all reversible. Therefore, quantum gates are represented by unitary matrices. That means the quantum gates in the circuits always have the same number of inputs and outputs. Quantum gates are operators—unitary matrices that act on a quantum state and transform the quantum state into another quantum state, which has the form:

$$
\begin{aligned}
U|\psi\rangle &= \begin{pmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \\
&= \begin{pmatrix} a \\ b \end{pmatrix} \\
&= |\phi\rangle .
\end{aligned}
$$

The list of Quantum gates:

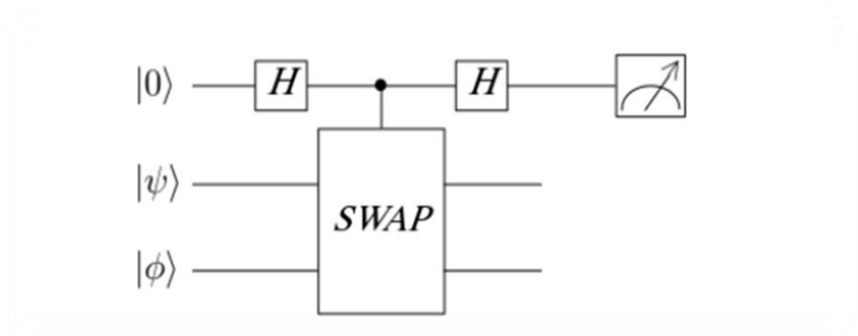
Pauli-X	$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \equiv \sigma_x$	It maps $\|0\rangle$ to $\|1\rangle$ and $\|1\rangle$ to $\|0\rangle$. It is equialent to NOT gate.
Pauli-Y	$Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \equiv \sigma_y$	It maps $\|0\rangle$ to $i\,\|1\rangle$ and $\|1\rangle$ to $-i\,\|0\rangle$.
Pauli-Z	$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \equiv \sigma_z$	It leaves the basis state $\|0\rangle$ unchanged and maps $\|1\rangle$ to $-\|1\rangle$. It is also called phase-flip.
Hadamard	$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$	It creates a superposition by mapping $\|0\rangle$ to $\frac{\|0\rangle+\|1\rangle}{\sqrt{2}}$ and $\|1\rangle$ to $\frac{\|0\rangle-\|1\rangle}{\sqrt{2}}$.

Phase shift	$S = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\phi} \end{pmatrix}$	It leaves the basis state $\|0\rangle$ unchanged and maps $\|1\rangle$ to $e^{i\phi}\,\|1\rangle$.
Controlled NOT	$CNOT = \begin{pmatrix} 1&0&0&0 \\ 0&1&0&0 \\ 0&0&0&1 \\ 0&0&1&0 \end{pmatrix}$	It acts on 2 qubits, and performs the NOT operation on the target qubit only when the control qubit is $\|1\rangle$.
SWAP	$SWAP = \begin{pmatrix} 1&0&0&0 \\ 0&0&1&0 \\ 0&1&0&0 \\ 0&0&0&1 \end{pmatrix}$	It acts on 2 qubits, and swaps these two qubits.

Quantum Machine learning Algorithms

1) SWAP test:

Swap-test is a simple and basic algorithm that is able to evaluate the overlap of two states. The overlap is a measure of similarity between two quantum states, and it is noted as $\langle \Psi \mid \Phi \rangle$.



It is necessary to get a lot of measurements to reach correct value of scalar product between two states.
The probability of measuring control qubit being in state $|0\rangle$ is given by:

$$(\boldsymbol{P}(|\mathbf{0}\rangle) = \frac{\mathbf{1}}{\mathbf{2}} + \frac{\mathbf{1}}{\mathbf{2}}\boldsymbol{F}(|\boldsymbol{a}\rangle, |\boldsymbol{b}\rangle))$$

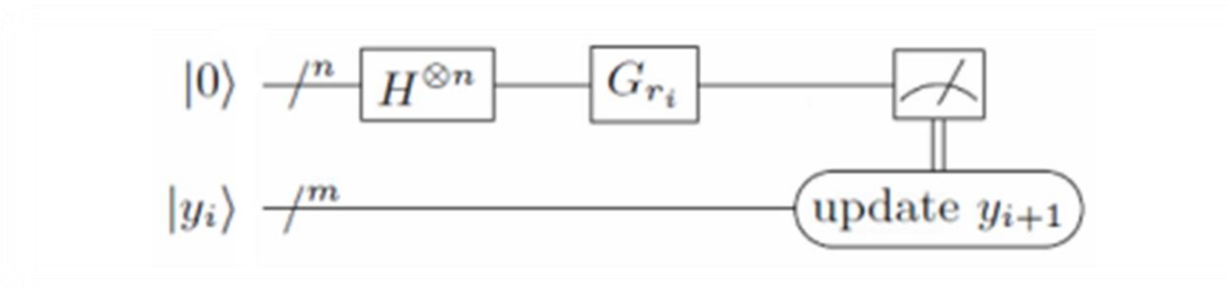
Where, $\boldsymbol{F}(|\boldsymbol{a}\rangle, |\boldsymbol{b}\rangle)$ =$|\boldsymbol{a}|\boldsymbol{b}\rangle|^{\mathbf{2}}$ – is the fidelity between the two input states.

The probability $P(|0\rangle) = 0.5$ means that the states $|a\rangle$ and $|b\rangle$ are orthogonal. And the probability $P(|0\rangle) = 1$ indicates that the states are identical. The routine should be repeated several times ($\geq 10^4$) to obtain a good estimation of fidelity.

SWAP-test can be used to calculate Euclidian distance between vectors in multidimensional space. It is very useful for various classical algorithms.

2) Quantum minimization algorithm (QMA):

It applies Grover's algorithm with quantum oracles that tells which items are smaller than the threshold and performs it several times to find out the solution. In addition, some researchers made significant improvements in the quantum search algorithm based on Grover's algorithm.



Quantum algorithm for finding the minimum:

Let T [0 → N −1] be an unsorted table of N items, each holding a value from an ordered set. For simplicity, assume that all values are distinct. The minimum searching problem is to find the index y such that T [y] is minimum.
Our algorithm calls the quantum exponential searching algorithm as a subroutine to find the index of a smaller item than the value determined by a particular threshold index. The result is then chosen as the new threshold. This process is repeated until the probability that the threshold index selects the minimum is sufficiently large.

Steps:

1. Choose threshold index $0 \leq y \leq N-1$ uniformly at random.
2. Repeat the following and interrupt it when the total running time is more than $22.5 \sqrt{N} + 1.4 \lg^2 N$. Then go to stage 2(2c).
    (a) Initialize the memory as $\sum_j \frac{1}{\sqrt{N}} |j\rangle|y\rangle$. Mark every item j for which T [j] < T [y].

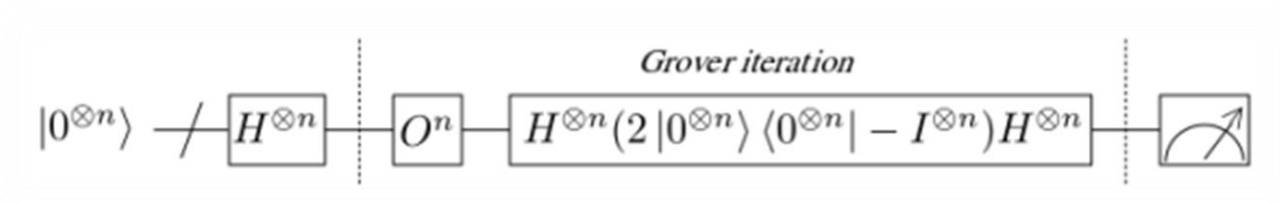
(b) Apply the quantum exponential searching algorithm of [2].

(c) Observe the first register: let y ′ be the outcome. If T [y′] < T [y], then set threshold index y to y ′.

3. Return y.

By convention, we assume that stage 2(2a) takes lg(N) time steps and that one iteration in the exponential searching algorithm takes one time step. The work performed in the stages 1, 2(2c), and 3 is not counted.

Resulting structure of Grover's Algorithm:



Such algorithm allows us to find global optimum as it is always searching through all possible input's states.

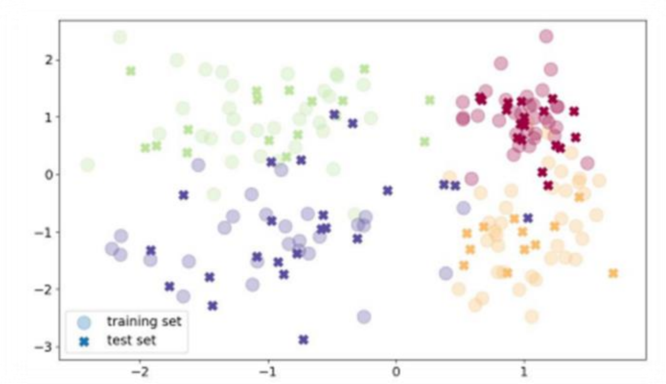
Use of these algorithms:

We use the above Quantum Machine learning algorithms in order to replace the existing algorithms used in Machine learning to accept and calculate the state of qubits.

We will show the use by creating Quantum K-Nearest Neighbors algorithm using SWAP test and QMA.

Quantum K-nearest Neighbor

KNN is a very popular and simple classification algorithm. Given a training dataset T of feature vectors and corresponding class labels. The idea of algorithm is to classify new data (test dataset D) by choosing the class label for each new input vector that appears most often among its k nearest neighbors. Such idea is based on the simple rule: close feature vectors have the similar class labels. It is true for many applications.



Quantum version of KNN is based on using SWAP-test (to calculate all scalar products between feature vectors encoded in quantum states) and QMA (to get k closest neighbors for each test vector). Therefore, quantum algorithm efficiency depends on number of SWAP-test iterations (n_{ST}) and number of QMA iterations (n_{QMA}).

Our modeling results (for datasets presented above) are shown below. Classification accuracy is demonstrated using confusion matrix for different values n_{ST} and n_{QMA}.

Confusion matrix is a performance measurement for machine learning classification problem where output can be two or more classes.

Let n - training set size, m - test set size, k - number of clusters and d - space dimension. Then, classical KNN computational complexity is $O(nm(k+d))$. Quantum approaches can improve this value. The quantum KNN complexity is $O(nm(k+\log d))$. Here we can see the computational advantage.

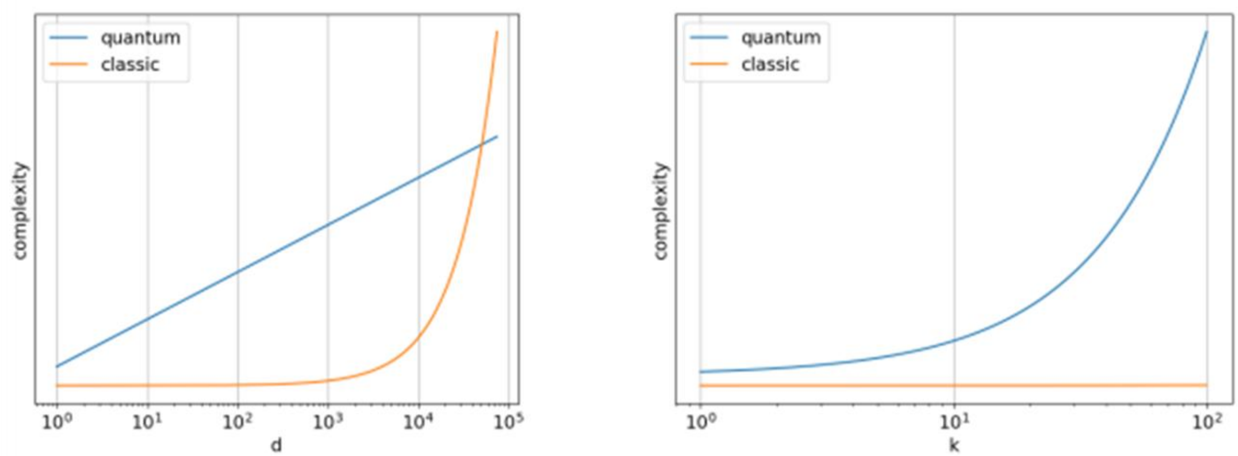


Figure 5. KNN algorithm computational complexities. Left - $k < d$, right - $k > d$.

As we can see, quantum approach leads to efficient computation only for $d > 5.10^4$ (Fig. 5, left picture). Therefore, quantum KNN algorithm can be applied to the task with large space dimension.

Conclusion:

This paper highlights some of the tools that can be used to speed up the classical KNN algorithm on a quantum computer. Quantum parallelism shows that processing speed can be vastly improved using a quantum computer. Addition-ally the linear storage space that qubits allow further enhances this. Using the swap test to compute the fidelity of two quantum states lays the groundwork to determine the distance between two data points. Combining this with a quantum minimization algorithm would allow a quantum computer to calculate the nearest neighbors for all data points.

It is worth noting that this not a complete review on Quantum Machine learning but, it showcases the power of using Quantum computing for high computational algorithm. Therefore, proving the advantage they have over existing systems and with enough development of research in Quantum computing might completely replace our existing Machine learning models.

References:

1) Machine learning methods in quantum computing theory, D.V. Fastovets, Yu.I. Bogdanov, B.I. Bantysh, V.F. Lukichev.
2) Recent Advances in Quantum Machine Learning, Yao Zhang and Qiang Ni.
3) Quantum Machine Learning: A Review and Current Status. Nimish Mishra, Manik Kapil, Hemant Rakesh, Amit Anand, Nilima Mishra, Aakash Warke, Soumya Sarkar, Sanchayan Dutta, Sabhyata Gupta, Aditya Prasad Dash, Rakshit Gharat, Yagnik Chatterjee, Shuvarati Roy, Shivam Raj, Valay Kumar Jain, Shreeram Bagaria, Smit Chaudhary, Vishwanath Singh, Rituparna Maji, Priyanka Dalei, Bikash K. Behera, Sabyasachi Mukhopadhyay, and Prasanta K. Panigrahi.
4) A quantum algorithm for finding the minimum. Christoph Durr and Peter Høyer.
5) Building a quantum kNN classifier with Qiskit: theoretical gains put to practice. D.J. Kok.