

Title: Quantum machine learning

URL: https://en.wikipedia.org/wiki/Quantum_machine_learning

PageID: 44108758

Categories: Category:Machine learning, Category:Quantum information science, Category:Quantum programming, Category:Theoretical computer science

Source: Wikipedia (CC BY-SA 4.0).

Introduction

Glossary

History

Classical mechanics

Old quantum theory

Bra–ket notation

Hamiltonian

Interference

Complementarity

Decoherence

Entanglement

Energy level

Measurement

Nonlocality

Quantum number

State

Superposition

Symmetry

Tunnelling

Uncertainty

Wave function Collapse

Collapse

Bell's inequality

CHSH inequality

Davisson–Germer

Double-slit

Elitzur–Vaidman

Franck–Hertz

Leggett inequality

Leggett–Garg inequality

Mach–Zehnder

Popper
Quantum eraser Delayed-choice
Delayed-choice
Schrödinger's cat
Stern–Gerlach
Wheeler's delayed-choice
Overview
Heisenberg
Interaction
Matrix
Phase-space
Schrödinger
Sum-over-histories (path integral)
Dirac
Klein–Gordon
Pauli
Rydberg
Schrödinger
Bayesian
Consciousness causes collapse
Consistent histories
Copenhagen
de Broglie–Bohm
Ensemble
Hidden-variable
Many-worlds
Objective-collapse
Quantum logic
Superdeterminism
Relational
Transactional
Relativistic quantum mechanics
Quantum field theory
Quantum information science
Quantum computing
Quantum chaos
EPR paradox
Density matrix

Scattering theory

Quantum statistical mechanics

Quantum machine learning

Aharonov

Bell

Bethe

Blackett

Bloch

Bohm

Bohr

Born

Bose

de Broglie

Compton

Dirac

Davisson

Debye

Ehrenfest

Einstein

Everett

Fock

Fermi

Feynman

Glauber

Gutzwiller

Heisenberg

Hilbert

Jordan

Kramers

Lamb

Landau

Laue

Moseley

Millikan

Onnes

Pauli

Planck

Rabi

Raman
Rydberg
Schrödinger
Simmons
Sommerfeld
von Neumann
Weyl
Wien
Wigner
Zeeman
Zeilinger
v
t
e

Quantum machine learning (QML), pioneered by Ventura and Martinez [1] and by Trugenberger in the late 1990s and early 2000s [2] [3] [4] , is the study of quantum algorithms which solve machine learning tasks. [5] [6] [7] [8]

The most common use of the term refers to quantum algorithms for machine learning tasks which analyze classical data, sometimes called quantum-enhanced machine learning. [9] [10] [11] QML algorithms use qubits and quantum operations to try to improve the space and time complexity of classical machine learning algorithms. [12] This includes hybrid methods that involve both classical and quantum processing, where computationally difficult subroutines are outsourced to a quantum device. [13] [14] [15] These routines can be more complex in nature and executed faster on a quantum computer. [7] Furthermore, quantum algorithms can be used to analyze quantum states instead of classical data. [16] [17]

The term "quantum machine learning" is sometimes used to refer classical machine learning methods applied to data generated from quantum experiments (i.e. machine learning of quantum systems), such as learning the phase transitions of a quantum system [18] [19] or creating new quantum experiments. [20] [21] [22]

QML also extends to a branch of research that explores methodological and structural similarities between certain physical systems and learning systems, in particular neural networks. For example, some mathematical and numerical techniques from quantum physics are applicable to classical deep learning and vice versa. [23] [24] [25]

Furthermore, researchers investigate more abstract notions of learning theory with respect to quantum information, sometimes referred to as "quantum learning theory". [26] [27]

Machine learning with quantum computers

Quantum-enhanced machine learning refers to quantum algorithms that solve tasks in machine learning, thereby improving and often expediting classical machine learning techniques. Such algorithms typically require one to encode the given classical data set into a quantum computer to make it accessible for quantum information processing. Subsequently, quantum information processing routines are applied and the result of the quantum computation is read out by measuring the quantum system. For example, the outcome of the measurement of a qubit reveals the result of a binary classification task. While many proposals of QML algorithms are still purely theoretical and require a full-scale universal quantum computer to be tested, others have been implemented on small-scale or special purpose quantum devices.

Quantum associative memories and quantum pattern recognition

Associative (or content-addressable) memories are able to recognize stored content on the basis of a similarity measure, while random access memories are accessed by the address of stored information and not its content. As such they must be able to retrieve both incomplete and corrupted patterns, the essential machine learning task of pattern recognition.

Typical classical associative memories store p patterns in the $O(n^2)$ interactions (synapses) of a real, symmetric energy matrix over a network of n artificial neurons. The encoding is such that the desired patterns are local minima of the energy functional and retrieval is done by minimizing the total energy, starting from an initial configuration.

Unfortunately, classical associative memories are severely limited by the phenomenon of cross-talk. When too many patterns are stored, spurious memories appear which quickly proliferate, so that the energy landscape becomes disordered and no retrieval is anymore possible. The number of storable patterns is typically limited by a linear function of the number of neurons, $p \leq O(n)$.

Quantum associative memories [2][3][4] (in their simplest realization) store patterns in a unitary matrix U acting on the Hilbert space of n qubits. Retrieval is realized by the unitary evolution of a fixed initial state to a quantum superposition of the desired patterns with probability distribution peaked on the most similar pattern to an input. By its very quantum nature, the retrieval process is thus probabilistic. Because quantum associative memories are free from cross-talk, however, spurious memories are never generated. Correspondingly, they have a superior capacity than classical ones. The number of parameters in the unitary matrix U is $O(pn)$. One can thus have efficient, spurious-memory-free quantum associative memories for any polynomial number of patterns. If the matrix U is encoded as a unique operator (as opposed as to a sequence of gates as in the circuit model), e.g. by an optical interferometer, the retrieval becomes efficient even for an exponential number of patterns.

Linear algebra simulation with quantum amplitudes

A number of quantum algorithms for machine learning are based on the idea of amplitude encoding, that is, to associate the amplitudes of a quantum state with the inputs and outputs of computations. [30][31][32] Since a state of n qubits is described by 2^n complex amplitudes, this information encoding can allow for an exponentially compact representation. Intuitively, this corresponds to associating a discrete probability distribution over binary random variables with a classical vector. The goal of algorithms based on amplitude encoding is to formulate quantum algorithms whose resources grow polynomially in the number of qubits n , which amounts to a logarithmic time complexity in the number of amplitudes and thereby the dimension of the input.

Many QML algorithms in this category are based on variations of the quantum algorithm for linear systems of equations [33] (colloquially called HHL, after the paper's authors) which, under specific conditions, performs a matrix inversion using an amount of physical resources growing only logarithmically in the dimensions of the matrix. One of these conditions is that a Hamiltonian which entry wise corresponds to the matrix can be simulated efficiently, which is known to be possible if the matrix is sparse [34] or low rank. [35] For reference, any known classical algorithm for matrix inversion requires a number of operations that grows more than quadratically in the dimension of the matrix (e.g. $O(n^{2.373})$), but they are not restricted to sparse matrices.

Quantum matrix inversion can be applied to machine learning methods in which the training reduces to solving a linear system of equations, for example in least-squares linear regression, [31][32] the least-squares version of support vector machines, [30] and Gaussian processes. [36]

A crucial bottleneck of methods that simulate linear algebra computations with the amplitudes of quantum states is state preparation, which often requires one to initialise a quantum system in a state whose amplitudes reflect the features of the entire dataset. Although efficient methods for state preparation are known for specific cases, [37][38] this step easily hides the complexity of the task. [39][40]

Variational Quantum Algorithms (VQAs)

In a variational quantum algorithm, a classical computer optimizes the parameters used to prepare a quantum state, while a quantum computer is used to do the actual state preparation and measurement. VQAs are considered promising candidates for noisy intermediate-scale quantum computers. Variational quantum circuits (or parameterized quantum circuits) are a popular class of VQAs where the parameters are those used in a fixed quantum circuit. [41] [42] [43] Researchers have studied VQCs to solve optimization problems and find the ground state energy of complex quantum systems, which were difficult to solve using a classical computer. [44]

Quantum binary classifier

Pattern reorganization is one of the important tasks of machine learning, binary classification is one of the tools or algorithms to find patterns. Binary classification is used in supervised learning and in unsupervised learning. In QML, classical bits are converted to qubits and they are mapped to Hilbert space; complex value data are used in a quantum binary classifier to use the advantage of Hilbert space. [45] [46] By exploiting the quantum mechanic properties such as superposition, entanglement, interference the quantum binary classifier produces the accurate result in short period of time. [47]

Quantum machine learning algorithms based on Grover search

Another approach to improving classical machine learning with quantum information processing uses amplitude amplification methods based on Grover's search algorithm, which has been shown to solve unstructured search problems with a quadratic speedup compared to classical algorithms. These quantum routines can be employed for learning algorithms that translate into an unstructured search task, as can be done, for instance, in the case of the k-medians [48] and the k-nearest neighbors algorithms. [9] Other applications include quadratic speedups in the training of perceptrons. [49]

An example of amplitude amplification being used in a machine learning algorithm is Grover's search algorithm minimization. In which a subroutine uses Grover's search algorithm to find an element less than some previously defined element. This can be done with an oracle that determines whether or not a state with a corresponding element is less than the predefined one. Grover's algorithm can then find an element such that our condition is met. The minimization is initialized by some random element in our data set, and iteratively does this subroutine to find the minimum element in the data set. This minimization is notably used in quantum k-medians, and it has a speed up of at least $O\left(\sqrt{\frac{n}{k}}\right)$ compared to classical versions of k-medians, where n is the number of data points and k is the number of clusters. [48]

Amplitude amplification is often combined with quantum walks to achieve the same quadratic speedup. Quantum walks have been proposed to enhance Google's PageRank algorithm [50] as well as the performance of reinforcement learning agents in the projective simulation framework. [51]

Quantum-enhanced reinforcement learning

In quantum-enhanced reinforcement learning, a quantum agent interacts with a classical or quantum environment and occasionally receives rewards for its actions. [52] [51] [53] In some situations, either because of the quantum processing capability of the agent, [51] or due to the possibility to probe the environment in superpositions, [29] a quantum speedup may be achieved. Implementations of these kinds of protocols have been proposed for systems of trapped ions [54] and superconducting circuits. [55] A quantum speedup of the agent's internal decision-making time [51] has been experimentally demonstrated in trapped ions, [56] while a quantum speedup of the learning time in a fully coherent ('quantum') interaction between agent and environment has been experimentally realized in a photonic setup. [57]

Quantum annealing

Quantum annealing is an optimization technique used to determine the local minima and maxima of a function over a given set of candidate functions. This is a method of discretizing a function with many local minima or maxima in order to determine the observables of the function. The process

can be distinguished from Simulated annealing by the Quantum tunneling process, by which particles tunnel through kinetic or potential barriers from a high state to a low state. Quantum annealing starts from a superposition of all possible states of a system, weighted equally. Then the time-dependent Schrödinger equation guides the time evolution of the system, serving to affect the amplitude of each state as time increases. Eventually, the ground state can be reached to yield the instantaneous Hamiltonian of the system.

Quantum sampling techniques

Sampling from high-dimensional probability distributions is at the core of a wide spectrum of computational techniques with important applications across science, engineering, and society. Examples include deep learning , probabilistic programming , and other machine learning and artificial intelligence applications.

A computationally hard problem, which is key for some relevant machine learning tasks, is the estimation of averages over probabilistic models defined in terms of a Boltzmann distribution . Sampling from generic probabilistic models is hard: algorithms relying heavily on sampling are expected to remain intractable no matter how large and powerful classical computing resources become. Even though quantum annealers, like those produced by D-Wave Systems, were designed for challenging combinatorial optimization problems, it has been recently recognized as a potential candidate to speed up computations that rely on sampling by exploiting quantum effects. [58]

Some research groups have recently explored the use of quantum annealing hardware for training Boltzmann machines and deep neural networks . [59] [60] [61] The standard approach to training Boltzmann machines relies on the computation of certain averages that can be estimated by standard sampling techniques, such as Markov chain Monte Carlo algorithms. Another possibility is to rely on a physical process, like quantum annealing, that naturally generates samples from a Boltzmann distribution. The objective is to find the optimal control parameters that best represent the empirical distribution of a given dataset.

The D-Wave 2X system hosted at NASA Ames Research Center has been recently used for the learning of a special class of restricted Boltzmann machines that can serve as a building block for deep learning architectures. [60] Complementary work that appeared roughly simultaneously showed that quantum annealing can be used for supervised learning in classification tasks. [59] The same device was later used to train a fully connected Boltzmann machine to generate, reconstruct, and classify down-scaled, low-resolution handwritten digits, among other synthetic datasets. [62] In both cases, the models trained by quantum annealing had a similar or better performance in terms of quality. The ultimate question that drives this endeavour is whether there is quantum speedup in sampling applications. Experience with the use of quantum annealers for combinatorial optimization suggests the answer is not straightforward. Reverse annealing has been used as well to solve a fully connected quantum restricted Boltzmann machine. [63]

Inspired by the success of Boltzmann machines based on classical Boltzmann distribution, a new machine learning approach based on quantum Boltzmann distribution of a transverse-field Ising Hamiltonian was recently proposed. [64] Due to the non-commutative nature of quantum mechanics, the training process of the quantum Boltzmann machine can become nontrivial. This problem was, to some extent, circumvented by introducing bounds on the quantum probabilities, allowing the authors to train the model efficiently by sampling. It is possible that a specific type of quantum Boltzmann machine has been trained in the D-Wave 2X by using a learning rule analogous to that of classical Boltzmann machines. [62] [61] [65]

Quantum annealing is not the only technology for sampling. In a prepare-and-measure scenario, a universal quantum computer prepares a thermal state, which is then sampled by measurements. This can reduce the time required to train a deep restricted Boltzmann machine, and provide a richer and more comprehensive framework for deep learning than classical computing. [66] The same quantum methods also permit efficient training of full Boltzmann machines and multi-layer, fully connected models and do not have well-known classical counterparts. Relying on an efficient thermal state preparation protocol starting from an arbitrary state, quantum-enhanced Markov logic networks exploit the symmetries and the locality structure of the probabilistic graphical model

generated by a first-order logic template. [67] [19] This provides an exponential reduction in computational complexity in probabilistic inference, and, while the protocol relies on a universal quantum computer, under mild assumptions it can be embedded on contemporary quantum annealing hardware.

Quantum neural networks

Quantum analogues or generalizations of classical neural nets are often referred to as quantum neural networks . The term is claimed by a wide range of approaches, including the implementation and extension of neural networks using photons, layered variational circuits or quantum Ising-type models .

Quantum Convolution Neural Network

A novel design for multi-dimensional vectors that uses circuits as convolution filters [68] is QCNN. It was inspired by the advantages of CNNs [69] [70] and the power of QML. It is made using a combination of a variational quantum circuit (VQC) [71] and a deep neural network [72] (DNN), fully utilizing the power of extremely parallel processing on a superposition of a quantum state with a finite number of qubits. The main strategy is to carry out an iterative optimization process in the NISQ [73] devices, without the negative impact of noise, which is possibly incorporated into the circuit parameter, and without the need for quantum error correction. [74]

The quantum circuit must effectively handle spatial information in order for QCNN to function as CNN. The convolution filter is the most basic technique for making use of spatial information. One or more quantum convolutional filters make up a quantum convolutional neural network (QCNN), and each of these filters transforms input data using a quantum circuit that can be created in an organized or randomized way. Three parts that make up the quantum convolutional filter are: the encoder, the parameterized quantum circuit (PQC), [75] and the measurement. The quantum convolutional filter can be seen as an extension of the filter in the traditional CNN because it was designed with trainable parameters.

Quantum neural networks take advantage of the hierarchical structures, [76] and for each subsequent layer, the number of qubits from the preceding layer is decreased by a factor of two. For n input qubits, these structure have $O(\log(n))$ layers, allowing for shallow circuit depth. Additionally, they are able to avoid "barren plateau," one of the most significant issues with PQC-based algorithms, ensuring trainability. [77] Despite the fact that the QCNN model does not include the corresponding quantum operation, the fundamental idea of the pooling layer is also offered to assure validity. In QCNN architecture, the pooling layer is typically placed between succeeding convolutional layers. Its function is to shrink the representation's spatial size while preserving crucial features, which allows it to reduce the number of parameters, streamline network computing, and manage over-fitting. Such process can be accomplished applying full Tomography on the state to reduce it all the way down to one qubit and then processed it in subway. The most frequently used unit type in the pooling layer is max pooling, although there are other types as well. Similar to conventional feed-forward neural networks, the last module is a fully connected layer with full connections to all activations in the preceding layer. Translational invariance, which requires identical blocks of parameterized quantum gates within a layer, is a distinctive feature of the QCNN architecture. [78]

Fully quantum machine learning

In the most general case of QML, both the learning device and the system under study, as well as their interaction, are fully quantum. This section gives a few examples of results on this topic.

One class of problem that can benefit from the fully quantum approach is that of 'learning' unknown quantum states, processes or measurements, in the sense that one can subsequently reproduce them on another quantum system. For example, one may wish to learn a measurement that discriminates between two coherent states, given not a classical description of the states to be discriminated, but instead a set of example quantum systems prepared in these states. The naive approach would be to first extract a classical description of the states and then implement an ideal discriminating measurement based on this information. This would only require classical learning. However, one can show that a fully quantum approach is strictly superior in this case. [79] (This

also relates to work on quantum pattern matching. [80]) The problem of learning unitary transformations can be approached in a similar way. [81]

Going beyond the specific problem of learning states and transformations, the task of clustering also admits a fully quantum version, wherein both the oracle which returns the distance between data-points and the information processing device which runs the algorithm are quantum. [82] Finally, a general framework spanning supervised, unsupervised and reinforcement learning in the fully quantum setting was introduced in, [29] where it was also shown that the possibility of probing the environment in superpositions permits a quantum speedup in reinforcement learning. Such a speedup in the reinforcement-learning paradigm has been experimentally demonstrated in a photonic setup. [57]

Explainable quantum machine learning

The need for models that can be understood by humans emerges in QML in analogy to classical machine learning and drives the research field of explainable QML (or XQML [83] in analogy to XAI/XML). These efforts are often also referred to as Interpretable Machine Learning (IML, and by extension IQML). [84] XQML/IQML can be considered as an alternative research direction instead of finding a quantum advantage. [85] For example, XQML has been used in the context of mobile malware detection and classification. [86] Quantum Shapley values have also been proposed to interpret gates within a circuit based on a game-theoretic approach. [83] For this purpose, gates instead of features act as players in a coalitional game with a value function that depends on measurements of the quantum circuit of interest. Additionally, a quantum version of the classical technique known as LIME (Linear Interpretable Model-Agnostic Explanations) [87] has also been proposed, known as Q-LIME. [88]

Quantum Kernel Methods and Generative Models

Quantum kernel methods have emerged as particularly promising approaches for near-term applications. Large-scale benchmarking studies encompassing over 20,000 trained models have provided comprehensive insights into the effectiveness of fidelity quantum kernels (FQKs) and projected quantum kernels (PQKs) across diverse classification and regression tasks. These studies have revealed universal patterns that guide effective quantum kernel method design. [89]

In the generative modeling domain, quantum generative adversarial networks and quantum circuit Born machines have shown particular promise for tabular data synthesis. Novel quantum generative models for tabular data have demonstrated performance improvements of 8.5% over leading classical models while using only 0.072% of the parameters, indicating significant potential for parameter-efficient learning. [90]

Classical learning applied to quantum problems

The term "quantum machine learning" sometimes refers to classical machine learning performed on data from quantum systems. A basic example of this is quantum state tomography , where a quantum state is learned from measurement. Other applications include learning Hamiltonians [91] and automatically generating quantum experiments. [20]

Quantum learning theory

Quantum learning theory pursues a mathematical analysis of the quantum generalizations of classical learning models and of the possible speed-ups or other improvements that they may provide. The framework is very similar to that of classical computational learning theory , but the learner in this case is a quantum information processing device, while the data may be either classical or quantum. Quantum learning theory should be contrasted with the quantum-enhanced machine learning discussed above, where the goal was to consider specific problems and to use quantum protocols to improve the time complexity of classical algorithms for these problems. Although quantum learning theory is still under development, partial results in this direction have been obtained. [92]

The starting point in learning theory is typically a concept class, a set of possible concepts. Usually a concept is a function on some domain, such as $\{0, 1\}^n$. For example, the concept class could be the set of disjunctive normal form (DNF) formulas on n bits or the set of

Boolean circuits of some constant depth. The goal for the learner is to learn (exactly or approximately) an unknown target concept from this concept class. The learner may be actively interacting with the target concept, or passively receiving samples from it.

In active learning, a learner can make membership queries to the target concept c , asking for its value $c(x)$ on inputs x chosen by the learner. The learner then has to reconstruct the exact target concept, with high probability. In the model of quantum exact learning, the learner can make membership queries in quantum superposition. If the complexity of the learner is measured by the number of membership queries it makes, then quantum exact learners can be polynomially more efficient than classical learners for some concept classes, but not more. [93] If complexity is measured by the amount of time the learner uses, then there are concept classes that can be learned efficiently by quantum learners but not by classical learners (under plausible complexity-theoretic assumptions). [93]

A natural model of passive learning is Valiant's probably approximately correct (PAC) learning . Here the learner receives random examples $(x, c(x))$, where x is distributed according to some unknown distribution D . The learner's goal is to output a hypothesis function h such that $h(x)=c(x)$ with high probability when x is drawn according to D . The learner has to be able to produce such an 'approximately correct' h for every D and every target concept c in its concept class. We can consider replacing the random examples by potentially more powerful quantum examples $\sum_x \sqrt{D(x)} |x\rangle |c(x)\rangle$. In the PAC model (and the related agnostic model), this doesn't significantly reduce the number of examples needed: for every concept class, classical and quantum sample complexity are the same up to constant factors. [94] However, for learning under some fixed distribution D , quantum examples can be very helpful, for example for learning DNF under the uniform distribution. [95] When considering time complexity, there exist concept classes that can be PAC-learned efficiently by quantum learners, even from classical examples, but not by classical learners (again, under plausible complexity-theoretic assumptions). [93]

This passive learning type is also the most common scheme in supervised learning: a learning algorithm typically takes the training examples fixed, without the ability to query the label of unlabelled examples. Outputting a hypothesis h is a step of induction. Classically, an inductive model splits into a training and an application phase: the model parameters are estimated in the training phase, and the learned model is applied an arbitrary many times in the application phase. In the asymptotic limit of the number of applications, this splitting of phases is also present with quantum resources. [96]

Implementations and experiments

The earliest experiments were conducted using the adiabatic D-Wave quantum computer, for instance, to detect cars in digital images using regularized boosting with a nonconvex objective function in a demonstration in 2009. [97] Many experiments followed on the same architecture, and leading tech companies have shown interest in the potential of QML for future technological implementations. In 2013, Google Research, NASA , and the Universities Space Research Association launched the Quantum Artificial Intelligence Lab which explores the use of the adiabatic D-Wave quantum computer. [98] [99] A more recent example trained a probabilistic generative models with arbitrary pairwise connectivity, showing that their model is capable of generating handwritten digits as well as reconstructing noisy images of bars and stripes and handwritten digits. [62]

Using a different annealing technology based on nuclear magnetic resonance (NMR), a quantum Hopfield network was implemented in 2009 that mapped the input data and memorized data to Hamiltonians, allowing the use of adiabatic quantum computation. [100] NMR technology also enables universal quantum computing, [citation needed] and it was used for the first experimental implementation of a quantum support vector machine to distinguish hand written number '6' and '9' on a liquid-state quantum computer in 2015. [101] The training data involved the pre-processing of the image which maps them to normalized 2-dimensional vectors to represent the images as the states of a qubit. The two entries of the vector are the vertical and horizontal ratio of the pixel intensity of the image. Once the vectors are defined on the feature space , the quantum support vector machine was implemented to classify the unknown input vector. The readout avoids costly

quantum tomography by reading out the final state in terms of direction (up/down) of the NMR signal.

Photonic implementations are attracting more attention, [102] not the least because they do not require extensive cooling. Simultaneous spoken digit and speaker recognition and chaotic time-series prediction were demonstrated at data rates beyond 1 gigabyte per second in 2013. [103] Using non-linear photonics to implement an all-optical linear classifier, a perceptron model was capable of learning the classification boundary iteratively from training data through a feedback rule. [104] A core building block in many learning algorithms is to calculate the distance between two vectors: this was first experimentally demonstrated for up to eight dimensions using entangled qubits in a photonic quantum computer in 2015. [105]

Recently, based on a neuromimetic approach, a novel ingredient has been added to the field of QML, in the form of a so-called quantum memristor, a quantized model of the standard classical memristor . [106] This device can be constructed by means of a tunable resistor, weak measurements on the system, and a classical feed-forward mechanism. An implementation of a quantum memristor in superconducting circuits has been proposed, [107] and an experiment with quantum dots performed. [108] A quantum memristor would implement nonlinear interactions in the quantum dynamics which would aid the search for a fully functional quantum neural network.

Since 2016, IBM has launched an online cloud-based platform for quantum software developers, called the IBM Q Experience . This platform consists of several fully operational quantum processors accessible via the IBM Web API. In doing so, the company is encouraging software developers to pursue new algorithms through a development environment with quantum capabilities. New architectures are being explored on an experimental basis, up to 32 qubits, using both trapped-ion and superconductive quantum computing methods.

In October 2019, it was noted that the introduction of Quantum Random Number Generators (QRNGs) to machine learning models including Neural Networks and Convolutional Neural Networks for random initial weight distribution and Random Forests for splitting processes had a profound effect on their ability when compared to the classical method of Pseudorandom Number Generators (PRNGs). [109] However, in a more recent publication from 2021, these claims could not be reproduced for Neural Network weight initialization and no significant advantage of using QRNGs over PRNGs was found. [110] The work also demonstrated that the generation of fair random numbers with a gate quantum computer is a non-trivial task on NISQ devices, and QRNGs are therefore typically much more difficult to use in practice than PRNGs.

A paper published in December 2018 reported on an experiment using a trapped-ion system demonstrating a quantum speedup of the deliberation time of reinforcement learning agents employing internal quantum hardware. [56]

In March 2021, a team of researchers from Austria, The Netherlands, the US and Germany reported the experimental demonstration of a quantum speedup of the learning time of reinforcement learning agents interacting fully quantumly with the environment. [111] [57] The relevant degrees of freedom of both agent and environment were realized on a compact and fully tunable integrated nanophotonic processor.

Breakthrough Solutions to Barren Plateaus

The barren plateau problem [1] where quantum algorithms encounter flat optimization landscapes—has seen significant theoretical and practical advances in 2025. Los Alamos National Laboratory researchers have provided the first mathematical characterization of why and when barren plateaus occur in variational quantum algorithms, establishing theoretical guarantees for algorithm scalability. [112] [113]

This work solves a key usability problem for quantum machine learning by providing rigorous theorems that predict whether a given architecture will remain trainable as it scales to larger quantum systems. The breakthrough eliminates the previous trial-and-error approach that had led to researcher fatigue in the field. [114]

Skepticism

While machine learning itself is now not only a research field but an economically significant and fast growing industry and quantum computing is a well established field of both theoretical and experimental research, QML remains a purely theoretical field of studies. Attempts to experimentally demonstrate concepts of QML remain insufficient. [citation needed] Further, another obstacle exists at the prediction stage because the outputs of quantum learning models are inherently random. [115] This creates an often considerable overhead, as many executions of a quantum learning model have to be aggregated to obtain an actual prediction.

Many of the leading scientists that extensively publish in the field of QML warn about the extensive hype around the topic and are very restrained if asked about its practical uses in the foreseeable future. Sophia Chen [116] collected some of the statements made by well known scientists in the field:

"I think we haven't done our homework yet. This is an extremely new scientific field," - physicist Maria Schuld of Canada-based quantum computing startup Xanadu.

"When mixing machine learning with 'quantum,' you catalyse a hype-condensate." [117] - Jacob Biamonte a contributor to the theory of quantum computation.

"There is a lot more work that needs to be done before claiming quantum machine learning will actually work," - computer scientist Iordanis Kerenidis, the head of quantum algorithms at the Silicon Valley-based quantum computing startup QC Ware.

"I have not seen a single piece of evidence that there exists a meaningful [machine learning] task for which it would make sense to use a quantum computer and not a classical computer," - physicist Ryan Sweke of the Free University of Berlin in Germany.

"Don't fall for the hype!" - Frank Zickert, who is the author of probably the most practical book related to the subject beware that "quantum computers are far away from advancing machine learning for their representation ability", and even speaking about evaluation and optimization for any kind of useful task quantum supremacy is not yet achieved. Furthermore, nobody among the active researchers in the field make any forecasts about when it could possibly become practical. [citation needed]

See also

Differentiable programming

Quantum computing

Quantum algorithm for linear systems of equations

Quantum annealing

Quantum neural network

Quantum image

References

v

t

e

DiVincenzo's criteria

NISQ era

Quantum computing timeline

timeline

Quantum information

Quantum programming

Quantum simulation

Qubit physical vs. logical
physical vs. logical
Quantum processors cloud-based
cloud-based
Bell's
Eastin–Knill
Gleason's
Gottesman–Knill
Holevo's
No-broadcasting
No-cloning
No-communication
No-deleting
No-hiding
No-teleportation
PBR
Quantum speed limit
Threshold
Solovay–Kitaev
Schrödinger-HJW
Classical capacity entanglement-assisted quantum capacity
entanglement-assisted
quantum capacity
Entanglement distillation
Entanglement swapping
Monogamy of entanglement
LOCC
Quantum channel quantum network
quantum network
State purification
Quantum teleportation quantum energy teleportation quantum gate teleportation
quantum energy teleportation
quantum gate teleportation
Superdense coding
Post-quantum cryptography
Quantum coin flipping
Quantum money
Quantum key distribution BB84 SARG04 other protocols

BB84
SARG04
other protocols
Quantum secret sharing
Algorithmic cooling
Amplitude amplification
Bernstein–Vazirani
BHT
Boson sampling
Deutsch–Jozsa
Grover's
HHL
Hidden subgroup
Magic state distillation
Quantum annealing
Quantum counting
Quantum Fourier transform
Quantum optimization
Quantum phase estimation
Shor's
Simon's
VQE
BQP
DQC1
EQP
QIP
QMA
PostBQP
Quantum supremacy
Quantum volume
QC scaling laws
Randomized benchmarking XEB
XEB
Relaxation times T_1 T_2
 T_1
 T_2
Adiabatic quantum computation
Continuous-variable quantum information

One-way quantum computer cluster state

cluster state

Quantum circuit quantum logic gate

quantum logic gate

Quantum machine learning quantum neural network

quantum neural network

Quantum Turing machine

Topological quantum computer

Hamiltonian quantum computation

Codes 5 qubit CSS GKP quantum convolutional stabilizer Shor Bacon–Shor Steane Toric gnu

5 qubit

CSS

GKP

quantum convolutional

stabilizer

Shor

Bacon–Shor

Steane

Toric

gnu

Entanglement-assisted

Cavity QED

Circuit QED

Linear optical QC

KLM protocol

Neutral atom QC

Trapped-ion QC

Kane QC

Spin qubit QC

NV center

NMR QC

Charge qubit

Flux qubit

Phase qubit

Transmon

OpenQASM – Qiskit – IBM QX

Quil – Forest/Rigetti QCS

Cirq

Q#

libquantum

many others...

Quantum information science

Quantum mechanics topics

v

t

e

Differentiable programming

Information geometry

Statistical manifold

Automatic differentiation

Neuromorphic computing

Pattern recognition

Ricci calculus

Computational learning theory

Inductive bias

IPU

TPU

VPU

Memristor

SpiNNaker

TensorFlow

PyTorch

Keras

scikit-learn

Theano

JAX

Flux.jl

MindSpore

Portals Computer programming Technology

Computer programming

Technology

v

t

e

algorithms

amplifier

bus
cellular automata
channel
circuit
complexity theory
computing
cryptography post-quantum
post-quantum
dynamics
electronics
error correction
finite automata
image processing
imaging
information
key distribution
logic
logic clock
logic gate
machine
machine learning
metamaterial
network
neural network
optics
programming
sensing
simulator
teleportation
Acoustic levitation
Anti-gravity
Cloak of invisibility
Digital scent technology
Force field Plasma window
Plasma window
Immersive virtual reality
Magnetic refrigeration
Phased-array optics

Thermoacoustic heat engine

List