Quantum Machine Learning Foundations and Applications: A Succinct Literature Review

J. Orduz[1], P. Rivas[2], and E. Baker[3]

School of Engineering and Computer Science
Department of Computer Science
Baylor University, Texas, USA
[1]Javier_OrduzDucuara@baylor.edu [2]Pablo_Rivas@baylor.edu
[3]Erich_Baker@baylor.edu

Abstract. The advantages of leveraging machine learning with quantum computing theory are still under study and very promising. The new field of quantum machine learning has great, transformative potential in our field. However, one of the major difficulties presented for those in either the machine learning community and the quantum community is the knowledge gap that naturally poses a barrier preventing both fields from joining forces without a significant amount of preparation, time, and space for professional development. With this in mind, we prepared this succinct literature review that provides an introduction to Quantum Machine Learning, involving selected topics in quantum mechanics, mathematics, and computing. We focus on fundamental concepts and a few interesting applications. Upon reading this review, the reader will be exposed to interesting topics about the technological revolution that Quantum Computing and Machine Learning have brought.

1 Introduction

Machine learning (ML) algorithms have been an object of study and scrutiny in recent years. The idea that algorithms can be improved for better generalization accuracy and efficiency has opened many sub-disciplines within ML [66]. One of such disciplines studies the possibility of using quantum theory to gain an edge on learning algorithms and achieve what is known as *quantum advantage* [53]. One way to achieve quantum advantage is through learning from massive amounts of data at the same time by representing data through quantum tools, e.g., using Hamiltonian theory [30]; by inferring from multiple data inputs and multiple models, [60]. A quantum advantage can also be achieved by using recent advances in quantum numerical optimization that may be usable in gradient descent-like calculations [7]. However, for the machine learner scientist transitioning into the field of quantum computing, or even understanding and applying some of the concepts behind quantum computing, it might pose particular difficulty if there is no additional training or preparation.

Our aim with this paper is to collect and coherently introduce some of the most relevant and fundamental concepts in Quantum Mechanics [39,40] and the necessary mathematical concepts behind it, and also a few ML concepts to ease the reader into making the necessary connections while identifying some applications, algorithms, and other areas surrounding this new discipline known as Quantum Machine Learning [5].

This review paper is organized as follows: Section 2 presents basic concepts on Quantum mechanics, Quantum Computing, ML, and, on the paradigm, Quantum Machine Learning. Section 3, we leave some applications which this QML field will be developing next years; and perspectives and comments. And section 4 exposes conclusions.

2 Foundations on Quantum Computing and ML

This section addresses fundamental aspects of Quantum Mechanics and the related mathematical concepts and ML, leading to identifying potential applications and research areas.

2.1 Basics of Quantum Mechanics and Quantum Computing

In the early 80s, Feynman proposed the idea of a quantum computer [13]; then Deutsch proposed a Turing machine in a quantum computer context [11]. Nowadays, we have at least six main paradigms to do quantum computing: (I) adiabatic quantum computation [12], (II) one-way quantum computer [50], (III) quantum Turing machine [4], (IV) topological quantum computer [15], (V) Quantum circuit (Quantum logic gate [1], an example is shown in Figure 1) and (VI) quantum machine learning (Differentiable quantum computing) [8,67], which manipulates states at a lower level of abstraction through Hamiltonian theory. The Hamiltonian can be defined as: $H = T + V$. In terms of energy, T represents the kinetic energy and V the potential energy. This representation has an interesting concept from a theoretical physical view [18]. A deeper analysis and discussion on this area is out of the scope of this document.

If one decides to work with either paradigm, there are several things we can do. For example, suppose we pursue the construction of quantum circuits. In that case, these can be designed to do certain operations at the level of quantum bits, a.k.a. quantum bits, qubits or *qbits* (we will use the latter notation), and the assumption is that input information is already available as *qbits*. On the other hand, dealing with Hamiltonian theory will enable modeling in terms of

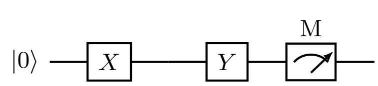

Fig. 1. An example of a typical quantum circuit commonly used in quantum computing with universal gates. In this circuit $|0\rangle$ is interpreted as qbit, lines below we will discuss about this concept; H, X, Y, Z represent the Hadamard and Pauli gates, and M is used to denote a measurement, respectively [32].

energy functions to optimize numerical problems or data treatment. In the next section, we will dive deeper into the basic concepts that form QML.

Many works already discuss the classical models in ML in an appropriate length, e.g., deep learning, supervised or unsupervised, and even adversarial learning [6,19,67,55]; some of the particular algorithms they normally expose include K-Means, K-nearest neighbors, Support Vector Machines (SVMs), and random forests. Some textbooks implement the algorithms with tools like Jupyter, Colab, or python scripts. In this classical information's view, we think about data as zeros (0's) and ones (1's), which represent other data (this document is represented by zeros and ones, for example). Those discretized and finite values can be used to express any other character.

Some of the most recent frameworks consider quantum computing (QC) [14,11] and ML as one integrated system; we refer to such systems as Quantum Machine Learning (QML). These three words load with relevant scientific meaning and are associated with some interesting applications. QML invokes the following Dirac's Algebra and related concepts: bra, ket, superposition, teleportation, and entanglement; which is what physicists normally use in quantum mechanics to describe the universe at some energy scales [48,41].

In the QML field, we use a quantum word to label the new bits: *qbits*, which stands for the minimum information quantity, and it is represented by:

$$|\psi\rangle = \sum_i \alpha_i |k_i\rangle\,, \tag{1}$$

which is a state vector, where $\{|k_i\rangle\}$ is a complete basis of elements in Hilbert space, the vertical-line and right-angle, $|\;\;\rangle$, means *ket*, bracket or Dirac's notation; and α's are $i-$complex coefficients, those so-called probability amplitudes. In quantum mechanics, under Dirac's notation, the state of a system is represented by a wave function: $|\psi\rangle$. The Schrödinger equation, $i\hbar\frac{\partial}{\partial t}|\psi\rangle = H|\psi\rangle$ shows the evolution of the quantum system represented by $|\psi\rangle$. Where $\hbar$ is the Planck's constant and H is the Hamiltonian operator. In quantum mechanics, observable refers to features that we can measure after experimenting and analyzing, are operators, which act on wave functions, and are represented by matrices acting on vectors, in terms of linear algebra. Hilbert space is a complex vector space where wave functions live [21].

QC uses the mathematical formalism of quantum mechanics as a tool since quantum mechanics describes discrete state changes. With regard to (1), we express the linear combinations of states, $|\psi\rangle = \alpha_0|0\rangle + \alpha_1|1\rangle$, as a qbit that lives in the Hilbert space, and amplitude probabilities respect the following condition $|\alpha_0|^2 + |\alpha_1|^2 = 1$. Which describes what is known as the normalization condition. Note that we use $\{|1\rangle, |0\rangle\}$ as a basis to represent the system. In general, we can express any state through (1) (superposition) as an addition of probabilities amplitudes in the basis $|k_i\rangle$, namely, $|1\rangle = \begin{pmatrix}0 & 1\end{pmatrix}^T$, and $|0\rangle = \begin{pmatrix}1 & 0\end{pmatrix}^T$. We can find various bases; in particular, this is called computational basis states. In general, a state vector can be written as $|\psi\rangle = \cos\frac{\theta}{2}|0\rangle + e^{i\phi}\sin\frac{\theta}{2}|1\rangle$, where $0 \leq \theta \leq \pi$, and $0 \leq \phi \leq 2\pi$ are real numbers. Now, to illustrate this, consider

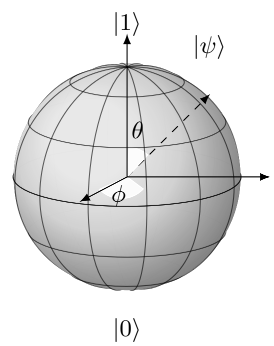

Fig. 2. Bloch sphere and the representation of $|\psi\rangle$ state vector with their angles and their (computational) basis vectors.

Figure 2, which shows the orthogonal vectors $|1\rangle$ and $|0\rangle$, and their angle which is π radians.

All the above concepts present the basic concepts about QC coming from quantum mechanics, in particular, Dirac's notation, linear algebra, and computing. Two interesting concepts coming from quantum mechanics are entanglement and teleportation. The first one means that it cannot be written as a product of a single-particle state [21], since, in a multiple-particle system, each particle interacts with others. The second one, teleportation, refers to a property that transfers information from a quantum state (source) to another one (destination). Sometimes, quantum computer scientists refer to entanglement and teleportation as *quantum protocols*.

Protocols are used in computing, in particular, as a defined set of rules to facilitate information exchange. With this in mind, let us go ahead and present some QC algorithms which are studied under the lenses of quantum protocols; these are summarized in Table 1. The table shows some problems tackled in QC (more documentation can be found in ref.[29]), where $f(x)$ can be either a constant or balanced function, which means that it is equal to 1 for half of all possible values of x, and 0 for the other half [43]; $U(x)$, sometimes called oracle, is

Table 1. Some quantum algorithms and their application, mapping, and a brief description function. D means Deutsch with $f(x)$ balanced and constant; DJ is Deutsch-Jozsa with $U(x)$ as a black box oracle function; BV is for Bernstein-Vazirano; and S is for Simon.

Problem	Maps	Function
D	$f : \{0,1\} \to \{0,1\}$	$f(x)$
DJ	$f : \{0,1\}^n \to \{0,1\}$	$U(x)$
BV	$f : \{0,1\}^n \to \{0,1\}$	$f(x) = a \bullet x$
S	$f : \{0,1\}^n \to \{0,1\}^n$	$f(x) = a \oplus x$

a function that maps $U_f |x\rangle |y\rangle = |x\rangle |y \otimes f(x)\rangle$ where $\otimes$ represents the tensorial product; and in problem S, $\oplus$ is the XOR-gate application. In problem BV, $\bullet$ is the dot-product in modulo two. This concludes with a brief summary of the elemental pieces of QC. The next section briefly discusses some ML fundamental concepts that are applicable to QC.

2.2 Machine Learning Fundamentals

Artificial intelligence is a broad field that encompasses many areas, including ML algorithms. These algorithms can be used for classification or regression tasks, and those can be done in supervised, semi-supervised, with reinforcement learning, and others. One of the primeval tasks of ML is about finding a pattern in data and learning from it, for example, from a preprocessed, labeled dataset.

A typical preprocessed dataset can be formally defined as $\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=0}^{N}$, where y is the desired output corresponding to the input vector $\mathbf{x} \in \mathbb{R}^d$. So, one of the motivations of ML is to use the data to find linear and non-linear transformations over $\mathbf{x}$ using highly complex tensor (vector) multiplications and additions, or to simply find ways to measure similarities or distances among data points, with the ultimate purpose of predicting y given $\mathbf{x}$. These calculations can be done with traditional or quantum computing techniques.

In the feature space of $\mathbf{x}$, patterns can be also studied through either statistical learning theory [64] or from information theory [52]. Foundationally, ML requires data, technical mathematics, and computing skills. With those elements, scientists can tackle problems considering data manifolds and their relationship to the desired output. Such as cases where the aim is to find structures in the data, then unsupervised learning might be employed by scientists. In such case, labeled data may not be strictly necessary, yet, the algorithm must be able to identify structures on the dataset or to model the data distribution rather than outputting a label prediction [67,6]. Many applications can be easily found in computer vision [56], and even astronomical data analysis [58].

A common way of thinking about a generalized supervised learning problem is to try to approximate some unknown function f over $\mathbf{x}$ as: $f(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + b = y$. Where $\mathbf{w}$ is an unknown vector that facilitates the transformation of $\mathbf{x}$ along with b. This formulation is very basic, linear, and is simply an illustration of what a simple learning model would look like. In this simple case, the ML algorithms revolve around finding the best $\mathbf{w}$ and b that yields the closest (if not perfect) approximation to y, the desired output. Very simple algorithms such as the perceptron [57], try different values for $\mathbf{w}$ and b using past mistakes in the choices of $\mathbf{w}$ and b to make the next selection in proportion to the mistakes made.

There are other supervised learning algorithms that depart from this trend in the search for the optimal separating hyperplane using statistical learning theory and numerical optimization, e.g., Support Vector Machines [59]. However, other methods such as K-nearest neighbors (KNN), or K-means, for example, use parallelizable distance-based measures to associate existing data to new data or to simply find natural data clusters [24]. Other algorithms can use different methods of optimization, such as least-squares, gradient descent, genetic algorithms, stochastic gradient descent, quadratic programming, or linear programming [54].

Some of the areas of opportunity for quantum advantage are currently within optimizing calculations, transforming data, or parallelizing certain aspects of the learning algorithms. We will see examples of these in the next few paragraphs.

2.3 Quantum Machine Learning

ML contains algorithms for classification as Support Vector Machines, KNN and have their quantum counterpart. In this subsection, we expose some algorithms from ML in the QC context.

A Support Vector Machine (SVM) is an algorithm that finds a separating hyperplane between two classes of data points [26]. Classical version: This hyperplane can be in the original feature space or higher-dimensional kernel space. The time complexity of this algorithm is $O(\log(\epsilon^{-1}) poly(N, M))$. Quantum SVM counterpart was labeled QSVM, and one of its features includes QSVM can perform this task with a time complexity of $O(\log NM)$ [51]. Here, N is the dimension of the feature space, M the number of training vectors, and ϵ the accuracy.

The Boltzmann Machine is a type of recurrent neural network that is based on the Hopfield Network [3].

- The network architecture is quite simple, and it consists of two layers, namely the visible nodes and the hidden nodes.
- Each node from both the layers is connecting to every other node in the network.
- It is also called an energy-based model since the Hamiltonian is used to define the network.

The quantum Boltzmann machine has shown some promise of outperforming classical Boltzmann machines in predictive tasks.

q-Means, a variant of the k-means algorithm is used to divide an unlabelled dataset of points into clusters based on a distance metric (e.g., Euclidean distance) [33], in this context,

- The running time of this algorithm $O(k\ d\ N)$ where N is the number of entities to be clustered and k and d are the number of clusters and the dimensions, respectively. In the quantum k-means algorithm, the euclidean distance the calculation is done using a quantum superposition [33,36].
- Hence the quantum k-means algorithm is theoretically shown to be computationally more efficient. This is very useful if clustering needs to be done on particle physics data sets containing millions of records. Further, the number of clusters, k, can be determined following traditional, non-quantum-based methods [25].

The q-Means algorithm is a version of δ-k-means, which is k-means with noise.

A Generative Adversarial Network (GAN) is an algorithm that is trained in a supervised fashion to generate unsupervised data [20,37].

- It uses two competing neural networks called the generator and discriminator to perform various classification and denoising tasks.
- A quantum GAN would be capable of performing supervised classification tasks and generating new data, following the dynamics of a quantum system.

A Quantum GAN was first proposed in ref. [38], and consider the discriminator, the generator, and the system generating coming from quantum computing.

QRAM is the quantum random access model, coming from the classical RAM, and others can be found QNN [27] and QRAM [17]. Other papers about neural networks, Quantum annealing and models related are in ref. [31].

Previous algorithms and concepts can be implemented in a phenomenological and theoretical view, to analyze and get information in areas as particle physics, finance, social models, or even in basic sciences.

The next section subsection shows tools, frameworks, and some applications and potential technologies on QML.

2.4 Tools for QML and Applications of QML

To the best of our knowledge, these are the most popular QML frameworks: PennyLane [65] and Strawberry Fields [34] and Qiskit [49]; in addition, other initiative are coming from companies, such as Q# [42]. These frameworks contain a variety of concepts, algorithms, and techniques developed to improve our knowledge about Quantum Computing. In this context, QML depends on engineering advances. Namely, fig. 3 shows the backend stage, where we need to boost the current investigations. In particular, quantum compilers and programming languages and, on the other hand, in new materials technology requires continuous work and advances.

In this tool, in the backend stage runs the quantum circuit, even though the code must be compiled. On the other hand, we have a provider who allows access to a group of different backends. This backend is a simulator of either classical or quantum computers that runs the quantum circuit and returns outputs.

We currently have some applications in different areas. In the following paragraphs, we discuss some of them, and leave some comments to motivate upcoming applications. We also provide several companies investing in this kind of technology.

In High Energy Physics, ML can be implemented in a theoretical and experimental view [2,23]:

1. Higher order computational methods: OneLoop, QCDLoop, LoopTools; parton level generators NNLO, DYNNLO, N3LO
2. Monte Carlo event generators and deep inelastic inclusive cross-sections: MadGraph, POWHEG and HERA
3. Analysis of data produced by LHC
4. Discovery of new particles
5. Studying the interactions and behaviours of different particles

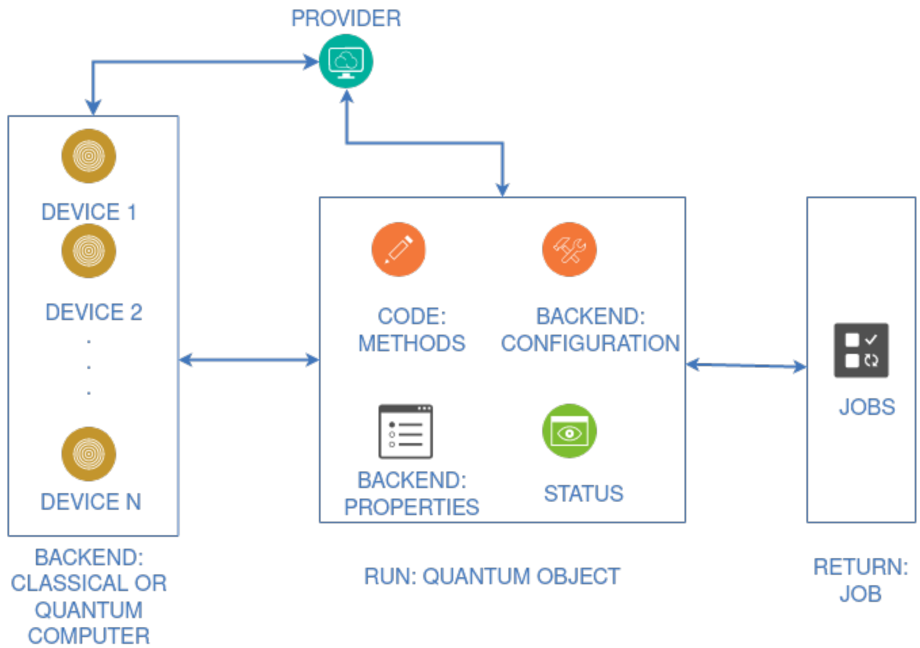

Fig. 3. Back-end architecture in the quantum context. The backend contains a quantum (classical) processor and quantum simulators which can be symbolic or numeric, or both [34].

Some interesting contributions on HEP have been reported by [46], and other talks and information on Quantum Machine Learning is promoted by [45,44].

In this area, the QSVM can take in a large feature space and perform the task with good efficiency, in a classification problem where particles are to be divided into two different sets (e.g., kind of fermions, bosons, couplings, or other parameters)

In Cryptography, all systems need security, an review and some ideas can be found in ref. [63]. Our systems must have high cybersecurity is seeking advances through QC and QML. Some advances in this area coming from various sources, P. Shor proposed an algorithm to reduce the factorization which is used in cryptography. This is a quantum algorithm [61,62]. Grover proposed a search algorithm that uses quantum ideas and notation to speed up the search time $O(\sqrt{n})$[22].

After implementing quantum technologies in different devices, security has to be improved to avoid threats; this is called post-quantum cryptographic algorithms (PQCA), also called quantum-resistant cryptography. Its goal is to develop cryptographic systems that are secure against both quantum and classical computers, and can interoperate with existing communications protocols and networks [9]. Some examples of post-quantum primitives have been proposed: families based on lattices, codes, and multivariate polynomials, as well as a handful of others.

In health, we have huge information coming from our own bodies, those data can be used to develop treatments to tackle cancer and other illnesses [28].

In general, QML is being applied on new materials, in wearable tech; new medicines and chemical combinations or genetic science; and pattern recognition as biometrics; and in sciences like mathematics, physics, astronomy, among others.

Big companies around the world are investigating applications about QC and QML. Some of those companies are: Google (technology), Barclays (finance), Airbus (transportation), US Navy and NASA (tech and applied science), Alibaba (commerce), IBM (technology) and more [35].

3 Discussion

The following list includes some exciting potential applications of QML and QC in different areas:

1. Cryptography. QC, QML, and PQCA can be used in new standards for secure communications. There are many possibilities in the aerospace industry to implement secure communications using quantum-based algorithms. Fig. 4 shows relevant cryptographic implications. We currently have cryptographic systems with RSA algorithm and n= 2048 [16]. R_S^G and R_{EH}^{RSA} have similar behavior throughout the plot; in particular, when $m_n = 2048$ and under particular conditions one RSA cipher could be broken in 8h.

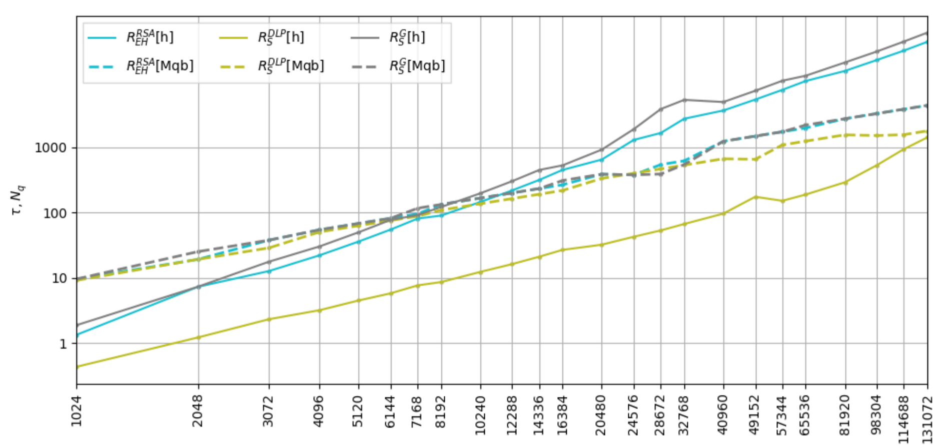

Fig. 4. Estimated space and expected-time costs with three models: R_{EH}^{RSA} is RSA via Ekerå-Håstad, R_S^{DLP} is Short DLP or Schnorr DLP via Ekerå-Håstad, and R_S^G is general DLP via Shor; h means hours, and Mqb is megaqubits. τ labels the expected time (h), and N_q numbers of physical qubit in Mqb; and m_n is the modulus length n (bits). Figure inspired and produced by ancillary files coming from [16].

2. Optimization theory. There are many possible applications in finance and healthcare where the distribution of resources needs to be constrained. The

transportation industry can also be positively transformed by finding optimal routes for vehicles via quantum-based numerical optimization techniques, e.g., with quantum annealing [10].
3. Image recognition. Given the applicability of QNN in a wide variety of image recognition tasks, the area of biometrics can be directly improved with faster convolutions and processing and extending these benefits to the areas of medical imaging and security [27].

There are many more applications besides those listed above that have been directly discussed through this paper, including some specific areas referenced in section 2. We expect to observe many more applications and further development of QML and QC after new materials, theories, and technological advances arise within the next two or three decades.

Other exciting areas are related to ethics. For example, with the increase of technology and its applications in our life, it is crucial to avoid social conflicts. Researchers have protocols to maintain fairness in classical machine learning; as Quantum Machine Learning gains attention, we shall have fairness protocols in this kind of technology. We expect this area provides significant concepts in subsequent years [47].

We similarly anticipate fruitful and exciting applications in health, cryptography, and the sciences. Such improvements will have a positive impact on our society and will represent the beginning of the quantum revolution for future centuries. Thus far, we have functional algorithms; however, we need more development in the general STEM areas, and we hope this paper is perceived as a call to pursue research in these areas to expand this field.

4 Conclusions

Quantum computing is one topic to investigate and strengthen, some recent results support this technology. This field requires particular knowledge of mathematics, physics, and computing. Some areas as Group Theory, Quantum Mechanics, Linear Algebra; in addition to reviewing some other topics further advanced in computation.

However, in this document, we left succinct ideas on machine learning, quantum computing, and basics mathematics in order to give an introduction to Quantum Machine Learning. Our document includes discussions on High Energy Physics, since it is a big source of data, and it provides pioneer technology and most recent techniques in storage and management of data. We further show the back-end architecture in the quantum context, we highlight the relevant contribution on the quantum simulators, quantum programming language, and contributions on the engineering to boost quantum computing.

We share the most relevant literature related to QML, and several applications in a variety of different fields. The reader can find important papers on Quantum Computing and its paradigms, such as, Quantum Machine Learning.

Acknowledgements

The authors thank the Department of Computer Science at Baylor University for their support. This research was also funded, in part by the Baylor AI lab.

References

1. A Yu Kitaev: Quantum computations: algorithms and error correction. Russian Mathematical Surveys **52**(6), 1191–1249 (1997). DOI {10.1070/rm1997v052n06abeh002155}. URL `https://doi.org/10.1070/rm1997v052n06abeh002155`
2. Albertsson, K., Altoe, P., Anderson, D., Andrews, M., Espinosa, J.P.A., Aurisano, A., Basara, L., Bevan, A., Bhimji, W., Bonacorsi, D., et al.: Machine learning in high energy physics community white paper (2018)
3. Amin, Mohammad H. and Andriyash, Evgeny and Rolfe, Jason and Kulchytskyy, Bohdan and Melko, Roger: Quantum Boltzmann Machine. Physical Review X **8**(2) (2018). DOI {10.1103/physrevx.8.021050}. URL `http://dx.doi.org/10.1103/PhysRevX.8.021050`
4. Benioff, P.: The computer as a physical system: A microscopic quantum mechanical hamiltonian model of computers as represented by turing machines. Journal of statistical physics **22**(5), 563–591 (1980)
5. Biamonte, J., Wittek, P., Pancotti, N., Rebentrost, P., Wiebe, N., Lloyd, S.: Quantum machine learning. Nature **549**(7671), 195–202 (2017)
6. Bishop, C.M.: Pattern recognition and machine learning. springer (2006)
7. Brandão, F.G., Kalev, A., Li, T., Lin, C.Y.Y., Svore, K.M., Wu, X.: Quantum sdp solvers: Large speed-ups, optimality, and applications to quantum learning. arXiv preprint arXiv:1710.02581 (2017)
8. Bshouty, N.H., Jackson, J.C.: Learning dnf over the uniform distribution using a quantum example oracle. SIAM Journal on Computing **28**(3), 1136–1153 (1998)
9. Chen, L., Chen, L., Jordan, S., Liu, Y.K., Moody, D., Peralta, R., Perlner, R., Smith-Tone, D.: Report on post-quantum cryptography, vol. 12. US Department of Commerce, National Institute of Standards and Technology (2016)
10. Das, A., Chakrabarti, B.K.: Quantum annealing and related optimization methods, vol. 679. Springer Science & Business Media (2005)
11. Deutsch, D.: Quantum theory, the Church-Turing principle and the universal quantum computer. Proceedings of the Royal Society of London Series A **400**(1818), 97–117 (1985). DOI 10.1098/rspa.1985.0070
12. Farhi, E., Goldstone, J., Gutmann, S., Sipser, M.: Quantum computation by adiabatic evolution. arXiv preprint quant-ph/0001106 (2000)
13. Feynman, R.P.: Simulating physics with computers. Int. J. Theor. Phys **21**(6/7) (1982)
14. Feynman, R.P.: Simulating Physics with Computers, p. 133–153. Perseus Books, USA (1999)
15. Freedman, Michael H and Kitaev, Alexei and Wang, Zhenghan: Simulation of topological field theories by quantum computers. Communications in Mathematical Physics **227**(3), 587–603 (2002)
16. Gidney, C., Ekerå, M.: How to factor 2048 bit rsa integers in 8 hours using 20 million noisy qubits. Quantum **5**, 433 (2021). DOI 10.22331/q-2021-04-15-433. URL `http://dx.doi.org/10.22331/q-2021-04-15-433`

17. Giovannetti, V., Lloyd, S., Maccone, L.: Quantum random access memory. Physical review letters **100**(16), 160,501 (2008)
18. Goldstein, H., Poole, C., Safko, J.: Classical mechanics addison-wesley. Reading, MA p. 426 (1980)
19. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press (2016). `http://www.deeplearningbook.org`
20. Goodfellow, Ian J. and Pouget-Abadie, Jean and Mirza, Mehdi and Xu, Bing and Warde-Farley, David and Ozair, Sherjil and Courville, Aaron and Bengio, Yoshua : Generative Adversarial Networks (2014)
21. Griffiths, D.J., Schroeter, D.F.: Introduction to quantum mechanics. Cambridge University Press (2018)
22. Grover, L.K.: A fast quantum mechanical algorithm for database search. In: Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing, STOC '96, p. 212–219. Association for Computing Machinery, New York, NY, USA (1996). DOI 10.1145/237814.237866. URL `https://doi.org/10.1145/237814.237866`
23. Guan, W., Perdue, G., Pesah, A., Schuld, M., Terashi, K., Vallecorsa, S., et al.: Quantum machine learning in high energy physics. Machine Learning: Science and Technology (2020)
24. Hamerly, G., Elkan, C.: Alternatives to the k-means algorithm that find better clusterings. In: Proceedings of the eleventh international conference on Information and knowledge management, pp. 600–607 (2002)
25. Hamerly, G., Elkan, C.: Learning the k in k-means. Advances in neural information processing systems **16**, 281–288 (2004)
26. Hearst, M.A., Dumais, S.T., Osuna, E., Platt, J., Scholkopf, B.: Support vector machines. IEEE Intelligent Systems and their applications **13**(4), 18–28 (1998)
27. Henderson, M., Shakya, S., Pradhan, S., Cook, T.: Quanvolutional neural networks: powering image recognition with quantum circuits. Quantum Machine Intelligence **2**(1), 1–9 (2020)
28. Iyer, V., Ganti, B., Hima Vyshnavi, A., Krishnan Namboori, P., Iyer, S.: Hybrid quantum computing based early detection of skin cancer. Journal of Interdisciplinary Mathematics **23**(2), 347–355 (2020)
29. Jordan, S.: Quantum algorithm zoo (2020). Last updated: 2019, Created: 2011. https://quantumalgorithmzoo.org/
30. Kappen, H.J.: Learning quantum models from quantum or classical data. Journal of Physics A: Mathematical and Theoretical **53**(21), 214,001 (2020)
31. Katsnelson, M.I., Vanchurin, V.: Emergent quantumness in neural networks. arXiv preprint arXiv:2012.05082 (2020)
32. Kay, A.: Tutorial on the quantikz package. arXiv preprint arXiv:1809.03842 (2018)
33. Kerenidis, Iordanis, and Landman, Jonas, and Luongo, Alessandro, and Prakash, Anupam: q-means: A quantum algorithm for unsupervised machine learning. In: H. Wallach and H. Larochelle and A. Beygelzimer and F. d'Alché-Buc and E. Fox and R. Garnett (ed.) Advances in Neural Information Processing Systems 32, pp. 4134–4144. Curran Associates, Inc. (2019). URL
34. Killoran, Nathan and Izaac, Josh and Quesada, Nicolás and Bergholm, Ville and Amy, Matthew and Weedbrook, Christian: Strawberry Fields: A Software Platform for Photonic Quantum Computing. Quantum **3**, 129 (2019). DOI 10.22331/q-2019-03-11-129. URL `https://doi.org/10.22331/q-2019-03-11-129`
35. Kommadi, B.: Quantum solutions overview. In: Quantum Computing Solutions, pp. 3–20. Springer (2020)

36. Lloyd, S., Mohseni, M., Rebentrost, P.: Quantum algorithms for supervised and unsupervised machine learning (2013)
37. Lloyd, S., Weedbrook, C.: Quantum generative adversarial learning. Physical review letters **121**(4), 040,502 (2018)
38. Lloyd, S., Weedbrook, C.: Quantum generative adversarial learning. Physical Review Letters **121**(4) (2018). DOI 10.1103/physrevlett.121.040502. URL `http://dx.doi.org/10.1103/PhysRevLett.121.040502`
39. Mermin, N.D.: From cbits to qbits: Teaching computer scientists quantum mechanics. American Journal of Physics **71**(1), 23–30 (2003). DOI 10.1119/1.1522741. URL `http://dx.doi.org/10.1119/1.1522741`
40. Mermin, N.D.: Quantum Computer Science: An Introduction. Cambridge University Press (2007). DOI 10.1017/CBO9780511813870
41. Meschini, D.: Planck-scale physics: Facts and beliefs. Foundations of science **12**(4), 277–294 (2007)
42. Microsoft: Q# (2017). Go to https://docs.microsoft.com/en-us/azure/quantum/overview-qdk
43. Nielsen, M.A., Chuang, I.: Quantum computation and quantum information (2002)
44. Orduz-Ducuara, J.: Quantum machine learning as a tool in hep (2020). XXXIV Reunión anual de la división de partículas y campos. Otros eventos relacionados: ISSN:2395-9061
45. Orduz-Ducuara, J. and Iyer, V.: Quantum Machine Learning concepts for Physicists (2020). LXAI colocated at NeuRIPS 2020
46. Paganini, M., de Oliveira, L., Nachman, B.: Accelerating science with generative adversarial networks: An application to 3d particle showers in multilayer calorimeters. Physical Review Letters **120**(4) (2018). DOI 10.1103/physrevlett.120.042003. URL `http://dx.doi.org/10.1103/PhysRevLett.120.042003`
47. Perrier, E.: Quantum fair machine learning. CoRR **abs/2102.00753** (2021). URL `https://arxiv.org/abs/2102.00753`
48. Pisano, F. and Reis, N. O.: Natural units, numbers and numerical clusters (2001)
49. Qiskit python api: https://qiskit.org (2017)
50. Raussendorf, R., Browne, D.E., Briegel, H.J.: Measurement-based quantum computation on cluster states. Phys. Rev. A **68**, 022,312 (2003). DOI 10.1103/PhysRevA.68.022312. URL `https://link.aps.org/doi/10.1103/PhysRevA.68.022312`
51. Rebentrost, P., Mohseni, M., Lloyd, S.: Quantum support vector machine for big data classification. Physical Review Letters **113**(13) (2014). DOI 10.1103/physrevlett.113.130503. URL `http://dx.doi.org/10.1103/PhysRevLett.113.130503`
52. Reza, F.M.: An introduction to information theory. Courier Corporation (1994)
53. Ristè, D., Da Silva, M.P., Ryan, C.A., Cross, A.W., Córcoles, A.D., Smolin, J.A., Gambetta, J.M., Chow, J.M., Johnson, B.R.: Demonstration of quantum advantage in machine learning. npj Quantum Information **3**(1), 1–5 (2017)
54. Rivas, P.: Accelerating the training of an lp-svr over large datasets. In: International Conference on Innovative Techniques and Applications of Artificial Intelligence, pp. 123–136. Springer (2020)
55. Rivas, P.: Deep Learning for Beginners: A beginner's guide to getting up and running with deep learning from scratch using Python. Packt Publishing Ltd (2020)
56. Rivas, P., Rivas, E., Velarde, O., Gonzalez, S.: Deep sparse autoencoders for american sign language recognition using depth images. In: 21st International Conference on Artificial Intelligence (ICAI 2019) (2019)
57. Rosenblatt, F.: The perceptron: a probabilistic model for information storage and organization in the brain. Psychological review **65**(6), 386 (1958)

58. Schawinski, K., Zhang, C., Zhang, H., Fowler, L., Santhanam, G.K.: Generative adversarial networks recover features in astrophysical images of galaxies beyond the deconvolution limit. Monthly Notices of the Royal Astronomical Society: Letters p. slx008 (2017). DOI 10.1093/mnrasl/slx008. URL `http://dx.doi.org/10.1093/mnrasl/slx008`
59. Schölkopf, B., Simard, P., Smola, A.J., Vapnik, V.: Prior knowledge in support vector kernels. Advances in neural information processing systems pp. 640–646 (1998)
60. Schuld, M., Sinayskiy, I., Petruccione, F.: Simulating a perceptron on a quantum computer. Physics Letters A **379**(7), 660–663 (2015)
61. Shor, P.W.: Algorithms for quantum computation: discrete logarithms and factoring. In: Proceedings 35th Annual Symposium on Foundations of Computer Science, pp. 124–134 (1994). DOI 10.1109/SFCS.1994.365700
62. Shor, P.W.: Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM review **41**(2), 303–332 (1999)
63. Sooksatra, K., Rivas, P.: A review of machine learning and cryptography applications (2020)
64. Vapnik, V.: The nature of statistical learning theory. Springer science & business media (2013)
65. Ville Bergholm, and Josh Izaac, and Maria Schuld, and Christian Gogolin, and M. Sohaib Alam, and Shahnawaz Ahmed, and Juan Miguel Arrazola, and Carsten Blank, and Alain Delgado, and Soran Jahangiri, and Keri McKiernan, and Johannes Jakob Meyer, and Zeyue Niu, and Antal Száva, and Nathan Killoran: PennyLane: Automatic differentiation of hybrid quantum-classical computations (2020)
66. Wang, M.H.: Artificial intelligence and subfields. Santa Clara University, Department of Computer Engineering (2017)
67. Wittek, P.: Quantum machine learning: what quantum computing means to data mining. Academic Press (2014)