

New Trends in Quantum Machine Learning

Lorenzo Buffoni^{1,2} and Filippo Caruso^{1,3}

¹*Dipartimento di Fisica e Astronomia, Università di Firenze, I-50019 Sesto Fiorentino, Italy*

²*Dipartimento di Ingegneria dell'Informazione, Università di Firenze, I-50139 Firenze, Italy*

³*LENS, QSTAR and CNR-INO, I-50019 Sesto Fiorentino, Italy*

Here we will give a perspective on new possible interplays between Machine Learning and Quantum Physics, including also practical cases and applications. We will explore the ways in which machine learning could benefit from new quantum technologies and algorithms to find new ways to speed up their computations by breakthroughs in physical hardware, as well as to improve existing models or devise new learning schemes in the quantum domain. Moreover, there are lots of experiments in quantum physics that do generate incredible amounts of data and machine learning would be a great tool to analyze those and make predictions, or even control the experiment itself. On top of that, data visualization techniques and other schemes borrowed from machine learning can be of great use to theoreticians to have better intuition on the structure of complex manifolds or to make predictions on theoretical models. This new research field, named as Quantum Machine Learning, is very rapidly growing since it is expected to provide huge advantages over its classical counterpart and deeper investigations are timely needed since they can be already tested on the already commercially available quantum machines.

I. INTRODUCTION

Machine learning (ML) [1–4] is a broad field of study, with multifaceted applications of cross-disciplinary breadth. ML ultimately aims at developing computer algorithms that improve automatically through experience. The core idea of artificial intelligence (AI) technology is that systems can learn from data, so as to identify distinctive patterns and make consequently decisions, with minimal human intervention. The range of applications of ML methodologies is extremely vast [5–8], and still growing at a steady pace due to the pressing need to cope with the efficiently handling of big data [9]. Training and deployment of large-scale machine learning models faces computational challenges [10] that are only partially met by the development of special purpose classical computing units such as GPUs. This has led to an interest in applying quantum computing to machine learning tasks [11–15] and to the development of several quantum algorithms [16–19] with the potential to accelerate training. Most quantum machine learning algorithms need fault-tolerant quantum computation [20–22], which requires the large-scale integration of millions of qubits and is still not available today. It is however possible that quantum machine learning (QML) will provide the first breakthrough algorithms to be implemented on commercially available Noisy Intermediate Scale Quantum (NISQ) devices [23–26]. Indeed, a number of interesting breakthroughs have already been made at the interface of Quantum Physics and Machine Learning [27]. For example in many-body quantum physics Machine Learning has been successfully employed to speed up simulations [28], predict phases of matter [29] or find variational ansatz for many body quantum states [30]. Similarly in quantum computation Machine Learning has been recently found success in quantum control [31] and to provide error correction [32].

In this paper we will give our perspective on new possible

trends of machine learning in the quantum domain, covering all the main learning paradigms and listing some very recent results. Thus we will try to briefly define the scope of each one of these ML domains, referring to the literature for a complete overview of the specific topics. The three main classes of learning algorithms are the following ones:

- **Supervised learning:** In supervised learning [1, 33] one deals with an annotated dataset $\{(x_i, y_i)\}_{i=1}^N$. Each element x_i is called an input or feature vector. It can be the vector of pixel values of an image or a feature such as height, weight and gender and so on. All input data x_i of the same dataset share the same features (with different values). The label y_i is the ground truth upon which we build the knowledge of our learning algorithm. It can be a discrete class in a set of possible objects or a real number representing some property we want to predict, or even some complex data structure. For example if we want to build a spam classifier the labels will be $y_i = 1$ (spam) or $y_i = 0$ (not spam). The goal of a supervised learning algorithm is to use the dataset to produce a model that, given an input vector x , can predict the correct label y .
- **Unsupervised learning:** In unsupervised learning [34–36] the dataset is a collection of unlabeled vectors $\{x_i\}_{i=1}^N$. The goal of unsupervised learning is to take this input vector and extract some useful property from the single data or the overall data distribution of the dataset. Examples of unsupervised learning are: clustering where the predicted property is the cluster assignment; dimensionality reduction where the distribution of data is mapped in a lower dimensional manifold; outlier detection where the property predicted is the 'typicality' of the data with respect to its distribution; and gen-

erative models where we want to learn to generate new points from the same distribution of the dataset.

- **Reinforcement learning:** The ML subfield of reinforcement learning [5] assumes that the machine ‘lives’ in an environment and can probe the state of the environment as a feature vector. The machine can perform different actions at different states bringing to different rewards. The goal of this machine (or agent) is to learn a policy/strategy. A policy is a function that associates to a particular feature vector, representing the state of the environment, the best action to execute. The optimal policy maximizes the expected average reward. Reinforcement learning has been widely employed in scenarios where decision making and long-term goals are crucial, for example in playing chess, controlling robots or logistics in complex environments.

Now it is important to point out that there is no rigorous universal definition of QML and instead it has assumed different meanings in the context where it has been studied in literature. The main subjects are the data that can be either classical or quantum, and the algorithm used to perform ML on the data itself, which can also be classical or quantum. One indeed can define four macro categories of algorithms [37] as depicted in Fig.1, three of which are generally considered QML. There are also a lot of gray areas where the algorithms are hybrid quantum-classical or only a subroutine or an optimization task is carried out by the quantum processor. An exhaustive list of all possible interplays between quantum and classical ML algorithms is outside the scope of this paper, so we will stick to this simplified representation for the sake of clarity.

II. SUPERVISED LEARNING

There are already in literature several examples of supervised learning applications for NISQ devices. For instance, small gate-model devices and quantum annealers have been used to perform quantum heuristic optimization [26, 38–42] and to solve classification problems [43–46]. An interesting approach to classification was devised in Ref. [47], where classical data are embedded into a larger quantum (Hilbert) space describing the state of a quantum system. The idea, which is similar in spirit to classical Support Vector Machines (SVMs) [1], is to map classical data into an high dimensional space where the classes can be well separated by an hyperplane. **In the quantum case the embedding is done by a quantum circuit composed of single- and multi-qubit gates, effectively mapping the classical data into the Hilbert space of the qubits.** For example, an embedding of a classical point x

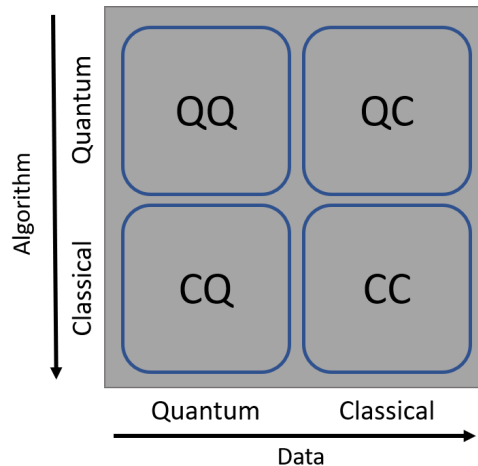


FIG. 1: Scheme of the various types of QML. The data to be processed can be either classical or quantum and the algorithm that process the data itself can also be either classical or quantum. That gives four possible combinations of algorithms, three of which (QQ, QC, CQ) usually fall under the umbrella of QML.

into a single qubit state $|x\rangle$ could be the following:

$$R_X(x)R_Y(\theta_3)R_X(x)R_Y(\theta_2)R_X(x)R_Y(\theta_1)R_X(x)|0\rangle \longrightarrow |x\rangle. \quad (1)$$

where R_X and R_Y are rotation operator along the axes X and Y , respectively, **while the rotation angles $\{\theta_1, \theta_2, \theta_3\}$ are the trainable parameters of our learning model.** Once a dataset $\{x_i\}_{i=1}^N$ is embedded, one can compute (using a SWAP test) the overlaps $|\langle x_i | x_j \rangle|^2$. Points (states) belonging to the same class will have an overlap close to 1 while points from different classes will have an overlap close to 0, hence enabling the classification of the dataset. The training can be done using software such as PennyLane [48] that **computes the gradients of the parametric quantum gates.** In Fig. 2 we show an embedding circuit for an one-dimensional dataset that can be trained by small quantum processors with the circuit proposed above.

In Ref. [49] we train an embedding of that dataset and test it with 10 validation points on a real quantum processor. The process has been carried out in the IBM quantum platform using the Valencia QPU (Quantum Processing Unit) composed of 5 qubits. The number of samples was rather small, namely 100 samples per point. That is due to the queues to access the system and to the fact that we need to compute 100 overlaps to build the Gram matrix for our 10 test points. As one can see in Fig. 3 our results [49], even if they contain some noise, are good and able to achieve a good classification boundary between the two classes. That is quite remarkable as this scheme is implemented on a real NISQ device via the cloud. Let us also point out that the error on the overlap depends on the number of samples we take and it could be the case that our 100 samples do not contain enough statistics. This method has then been proved ef-

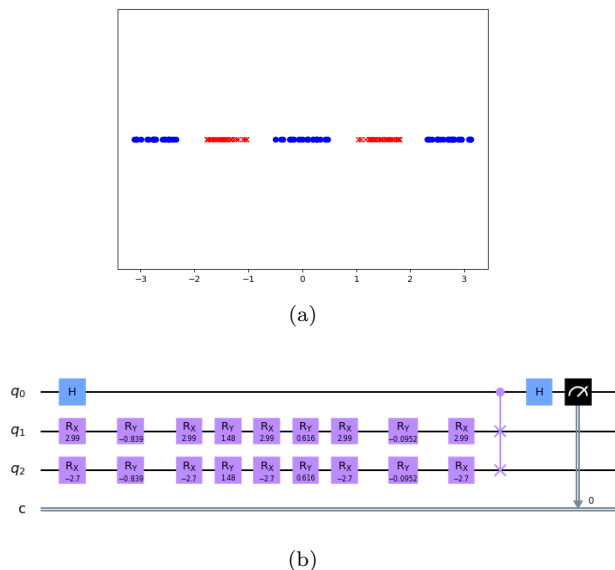


FIG. 2: In panel (a) the 1D syntetic dataset used as classification benchmark. Class A are the blue dots while class B are the red crosses. Note that in the 1D space the dataset cannot be linearly classified, i.e. there is not a simple threshold allowing us to separate the two classes. In panel (b) an example of the embedding circuit including the final SWAP test to compute the overlap between the two embeddings [49].

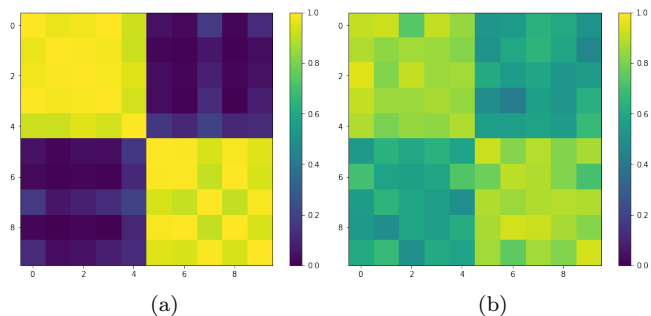


FIG. 3: (a) Theoretical Gram matrix, representing the overlaps between 10 test embeddings; (b) Experimental Gram matrix on the IBM Valencia QPU. The achieved results are pretty consistent with the theory albeit a bit noisier than other setups that are exploited in Ref. [49].

fective on small datasets (embedded in 1 or 2 qubits) and should be quite robust to experimental noise, yielding some promises for the future of NISQ devices. Indeed, as argued in Ref. [47], we can map high-dimensional data in a n -qubit state using the same protocol. And, if we could build a circuit of 100 qubits with circuit depth 100 and a decoherence time of 10^{-3} s, it could be capable to embed $\mathcal{O}(10^{10})$ bits of classical information, i.e. a task which is classically unattainable. Therefore, high-dimensional embeddings of large data sets for (QC-type) QML could be accessible in future on NISQ devices.

Once classical data are embedded on quantum states,

one has to deal with a fundamental problem in quantum physics that is the discrimination amongst a set of non-orthogonal quantum states of a system. This can be addressed for instance by following a recent approach [50] based on the quantum stochastic walk model [51, 52] and inspired by the structure of classical neural networks. In particular, the quantum states to discriminate can be encoded on the input nodes of a network, while the discrimination is achieved on the output nodes. Then, the parameters of the underlying network (e.g. hopping rates or link weights) are optimized to obtain the highest probability of correct discrimination. Interestingly enough, this probability approaches the theoretical optimal quantum limit known as *Helstrom bound* [53]. Our numerical and analytical results in Ref. [50] also show the robustness and reconfigurability of the network for different set of quantum states. Our proposed (QQ-type) architecture can pave the way for experimental realizations of QML protocols as well as novel quantum generalizations of deep learning algorithms running over larger complex networks.

Finally, let us mention a CQ-type QML application we are proposing in Ref. [54] where artificial neural network models [1, 55] are exploited for quantum noise discrimination in stochastic quantum dynamics. In particular some **Hamiltonian parameters are affected by noise, which is described as a stochastic process associated to a specific probability distribution. Our aim is to discriminate among different noise probability distributions and correlation parameters, by measuring the quantum state at discrete time instants.** In particular, we proposed the use of classical ML algorithms such as SVMs and Recurrent Neural Networks (RNN), i.e. Long Short Term Memory (LSTM) [56] and Gated Recurrent Unit (GRU) [57], to perform supervised classification tasks on simulated quantum data. We believe that these are preliminary steps towards QML algorithms for quantum sensing [58], e.g. to detect Markovian vs. non-Markovian noise in real quantum systems as optical and atomic platforms that are very promising candidates for NISQ devices.

III. UNSUPERVISED LEARNING

Unlike supervised learning [33], unsupervised learning is a much harder, and still largely unsolved, problem. And yet, it has the appealing potential to learn the hidden statistical correlations of large unlabeled datasets [34–36], which constitute the vast majority of data being available today.

For instance, clustering algorithms have the ability to separate unlabeled data in different classes (or clusters) without the need for external labeling and supervision. Hybrid approaches have been tried to solve clustering [59] by applying a quantum-based optimization scheme to a classical clustering algorithm. **In order to represent the clusters, we need to define a distance measure $d(x_i, x_j)$ between two data elements x_i and x_j .** For example, a

choice could be the Euclidean distance, but specific applications may naturally lead to very different metrics. You can then calculate the distance matrix C with elements $C_{ij} = d(x_i, x_j)$. Now this matrix can also be interpreted as adjacency matrix of a weighted graph G and, by a suitable choice of metric and doing some coarse graining, the problem of clustering reduces to the MAXCUT optimization problem [60] on the graph G . **The MAXCUT problem is an example of the class of NP-complete problems, which are notoriously hard to be solved.** In Ref. [59] an hybrid scheme is devised where the MAXCUT optimization is solved by the QAOA algorithm [41] and applied to a syntetic dataset composed of 2 clusters of 10 points living in a 2-dimensional space. The model was also successfully tested on a real 19-qubit chip produced by Rigetti as demonstration of a QC-type application.

Another opportunity to use quantum technologies on unsupervised learning tasks comes from Variational Autoencoders (VAEs) [61]. Variational Autoencoders are generative models in which the goal is to learn a complex data distribution (e.g. the pixels of images representing cats) to later sample from it 'generating' new objects. In particular VAEs learn a probability distribution $p_\theta(x, z) = p_\theta(x|z)p_\theta(z)$, where usually the a *posterior* distribution $p_\theta(x|z)$ is implemented by a Deep Neural Network and $p_\theta(z)$ is simple (prior) distribution (e.g., i.i.d Gaussian or Bernoulli variables). The idea here is to replace the distribution $p_\theta(z)$ with a complex distribution sampled from a quantum device [62, 63]. This setup allows for the construction of quantum-classical hybrid generative models that can be scaled to large, realistic datasets. For example in Ref. [63] a quantum-classical hybrid VAE was trained using a D-Wave 2000Q quantum annealer on the popular MNIST [64] dataset of hand-written digits. This quantum-classical hybrid VAE still employs a large amount of classical computing power performed on modern GPUs. The computational task that we offloaded to the quantum annealer (sampling from a complex distribution) can still be performed classically at a fraction of the overall computational cost. To achieve any form of quantum advantage in this framework, we need to offload generative capacity to the prior distribution, by exploiting large graphs capable of representing complex probability distributions **from which classical sampling becomes too expensive. We have evidence that this path to quantum advantage is possible by deploying annealers with denser connectivities and lower noise, engineering classical neural networks that better exploit physical connectivities and by working with more complex datasets. All these improvements should be achievable in the near future, and represent possible interesting lines of research in the area of QC-type QML models.**

Furthermore, one of the most remarkable results in unsupervised ML is provided by generative adversarial networks (GANs) [65, 66], where game-theory models aim to learn how to faithfully reproduce some given distribution of data. **In particular, the idea is that two agents, named as the generator and the discriminator, compete against**

each other in a zero-sum game where on one side the generator aims to generate 'fake' data that the discriminator is not able to distinguish from the real generated ones, while on the other side the latter optimizes its discrimination strategy. Under some reasonable assumptions, it is possible to demonstrate that the game reaches the unique (Nash) equilibrium point where the generator is able to exactly reproduce the desired (real) data distribution. GANs have been generalized to the quantum case leading to the so-called Quantum Generative Adversarial Networks (QGANs) [67, 68]. They are an example of QQ-type QML. Again the goal is to learn of reproducing the state of a physical system that is now quantum, e.g. a register of qubits. This can be for instance implemented by exploiting parametrized quantum circuits (where quantum gates are controlled by real tunable parameters) [69] allowing to realize, among others, quantum approximation optimization algorithms [41], VAEs [70], and eigensolvers [42]. However, recent efforts have been mainly focused on learning pure states [68, 71], while the scenario of mixed quantum states is currently under investigation and some preliminary results are appearing in literature [72, 73]. In particular, in Ref. [73] we show how the emergence of limit cycles may considerably extend the convergence time in the case of mixed quantum (generated data) states that of course play a key role for practical implementations on commercially available NISQ devices.

IV. REINFORCEMENT LEARNING

Reinforcement learning has been already applied to closed quantum systems, i.e. following a unitary dynamical evolution [37]. Yet, the setting of an agent acting on an environment has a natural analogue in open quantum systems [74]. In Ref. [75] we propose to generalize RL into the quantum domain in order to solve a Quantum Maze problem. It is an example of QQ-type QML. For Quantum Maze we mean a network whose topology is represented by a perfect maze, i.e. there is always one unique path between any two points in the maze, and the dynamical evolution of the physical state (described by its adjacency matrix ρ) of this network is quantum. The paths of the maze are defined by links between pair of nodes, which are described by an adjacency matrix A . The coefficient $A_{i,j} = 1$ indicates the presence of the link, while $A_{i,j} = 0$ indicates its absence. The evolution of the quantum system has been based on the quantum stochastic walk model [51, 52] that can be described by a Lindblad master equation [76] where the Hamiltonian of the system H corresponds to the adjacency matrix itself, i.e. $H = A$, and with a set of Lindblad operators acting as noise for the quantum walker, i.e.

$$\dot{\rho} = -(1-p) i[H, \rho] + p \mathcal{L}_{CRW}(\rho) + \mathcal{L}_{sink}(\rho) \quad (2)$$

with a term describing (for $p = 1$) the regime of a classical random walk (CRW), as

$$\mathcal{L}_{CRW}(\rho) = \sum_{i,j} L_{ij} \rho L_{ij}^\dagger - \frac{1}{2} \{L_{ij}^\dagger L_{ij}, \rho\}, \quad (3)$$

with the Lindblad operators being defined as $L_{ij} = (A_{ij}/d_j) |i\rangle \langle j|$, where $\{d_j\}$ are the node degrees. When $p = 0$ one recovers the pure quantum walker regime, while $p = 1$ corresponds to the classical scenario for a random walker. For the intermediate values of p in the range $]0, 1[$, one has a quantum stochastic walker with an interplay between coherent evolution and noise effects. In addition, there is a Lindblad operator that irreversibly transfers the population from the “exit” node n to the sink S with a rate Γ , as follows

$$\mathcal{L}_{sink}(\rho) = \Gamma [2 |S\rangle \langle n| \rho |n\rangle \langle S| - \{|n\rangle \langle n|, \rho\}]. \quad (4)$$

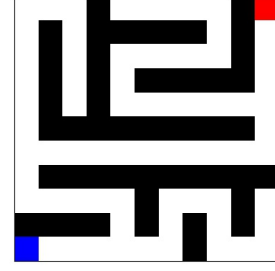
Moreover, we assume that all population is initially at the entrance node of the maze. The exit of the maze corresponds to the sink S which irreversibly traps the population. Equation 2 gives leads to the following expression for the escaping probability from the maze

$$p_{sink}(t) = 2\Gamma \int_0^t \rho_{n,n}(t') dt'. \quad (5)$$

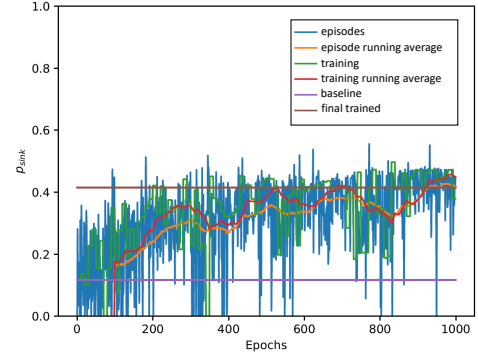
The goal is to maximize this escaping probability in the shortest amount of time. Since we consider a perfect maze, there is a single path to exit the maze from the entrance node, in presence of several dead ends. In this scenario, an external controller (the computer user or the quantum system itself) is the agent that has some information about the quantum state of the system, while the maze is the RL environment. The available actions for the agent, during the evolution, are:

- 1 – Building walls. During the evolution, at given (periodic) time instants, the system can change the adjacency matrix of the environment removing a link (changing from 1 to 0 an entry $A_{i,j}$). This emulates the closing of a door through a passage that for instance leads to a dead end or that does not bring efficiently the walker to the exit.
- 2 – Breaking walls. The system can change the adjacency matrix of the environment adding a link (changing from 0 to 1 an entry $A_{i,j}$). This mimics the creation of an hole in a maze wall, hence opening a shortcut.

In this setup, the objective function is the amount of time required to exit the maze (to be minimized) or the amount of population that exits to the sink in a given amount of time (to be maximized). If the adjacency matrix can change – either intrinsically by random flips or as a result of actions taken by the agent – we are in the canonical scenario for RL but in a quantum domain (quantum RL). We thus want to learn a policy by which,



(a)



(b)

FIG. 4: In panel (a) a sample of a 6×6 perfect maze. In white the possible paths, in black the walls. In the lower left corner, the node in blue is the entrance to the maze, corresponding to the initial quantum state, while the node in red is the exit, i.e., the node connected to the sink. In panel (b) an example of a training curve for an agent doing actions at periodic time instants, in the regime of $p = 0.8$ on the 6×6 maze in (a). The curves show the rewards from the single episodes and their running average over 100 episodes as well as the training curve of the agent with the running average again over 100 episodes. The two constant lines are the baseline quantum walker with no actions performed by the agent and the final trained policy. [75]

dynamically changing the topology of the maze, even by just adding or removing a small number of walls, we can significantly improve the transfer rate of the walker to the sink. We can also imagine the action of modifying the adjacency matrix as a sort of additional, engineered *noise* that, if carefully tuned, can improve the exit performance of the stochastic quantum walker. An example of such an improvement is shown in Fig.4 where an agent was trained to perform the described actions on a 6×6 perfect maze. We can see how the performance improves from the baseline stochastic quantum walker as the agent learns how to get better rewards, i.e. transferring more population into the sink. The number of actions made by the agent was quite small (8 in this particular case) but carefully planned in order to yield a great boost to the efficiency of the walker. Given the key role of transport of energy or information over complex networks, this ap-

proach (see more details in Ref. [75]) might represent an interesting and promising step towards novel QML schemes for NISQ devices and quantum technologies in general.

V. CONCLUSIONS

Quantum Machine Learning is where nowadays machine learning is going to meet quantum information science in order to realize more powerful quantum technologies. However, a much deeper understanding of their underlying mechanisms is still required in order to develop new algorithms and especially to apply them to address real problems and then to lead to commercial applications. This is a very young but very rapidly developing research field where the first results pave the way for new experimental demonstrations of such hybrid

classical-quantum protocols allowing to evaluate the potential advantages of exploiting them over their classical ML counterparts and then to exploit them on commercially available or coming soon NISQ devices and quantum technologies. We have hereby presented a perspective on new recent algorithms covering the main areas of ML (supervised, unsupervised and reinforcement learning) and different combinations of quantum-classical data and algorithms (QQ, QC, CQ models). We have also pointed out that there is still plenty of work to do, as it is definitely necessary to scale up the algorithms listed here (and all the other models that have been proposed recently) to the limits of existing devices performing an accurate scaling analysis of performances and corresponding errors. Ultimately, the crucial question we will need to answer in this field is whether a quantum speedup is theoretically and experimentally feasible via Quantum Machine Learning models running on NISQ machines.

-
- [1] Christopher M. Bishop. Springer, New York, 1st ed. 2006. corr. 2nd printing 2011 edition, April 2011.
 - [2] T. M. Cover and Joy A. Thomas. Wiley series in telecommunications. Wiley, New York, 1991.
 - [3] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. Springer Science & Business Media, 2009.
 - [4] <http://themlbook.com/>.
 - [5] Richard S Sutton and Andrew G Barto. MIT press, 2018.
 - [6] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. In *2013 IEEE international conference on acoustics, speech and signal processing*, pages 6645–6649. IEEE, 2013.
 - [7] Nicu Sebe, Ira Cohen, Ashutosh Garg, and Thomas S Huang. volume 29. Springer Science & Business Media, 2005.
 - [8] Sorin Grigorescu, Bogdan Trasnea, Tiberiu Cocias, and Gigel Macesanu. *Journal of Field Robotics*, 37(3):362–386, 2020.
 - [9] Min Chen, Shiwen Mao, and Yunhao Liu. *Mobile networks and applications*, 19(2):171–209, 2014.
 - [10] Tijmen Tieleman. In *Proceedings of the 25th international conference on Machine learning*, pages 1064–1071. ACM, 2008.
 - [11] Maria Schuld, Ilya Sinayskiy, and Francesco Petruccione. *Contemporary Physics*, 56(2):172–185, 2015.
 - [12] Peter Wittek. Academic Press, 2014.
 - [13] Jeremy Adcock, Euan Allen, Matthew Day, Stefan Frick, Janna Hinchliff, Mack Johnson, Sam Morley-Short, Sam Pallister, Alasdair Price, and Stasja Stanisic. *arXiv preprint arXiv:1512.02900*, 2015.
 - [14] Srinivasan Arunachalam and Ronald de Wolf. *arXiv preprint arXiv:1701.06806*, 2017.
 - [15] Jacob Biamonte, Peter Wittek, Nicola Pancotti, Patrick Rebentrost, Nathan Wiebe, and Seth Lloyd. *Nature*, 549:195–202, 2017.
 - [16] Aram W Harrow, Avinandan Hassidim, and Seth Lloyd. *Physical review letters*, 103(15):150502, 2009.
 - [17] Nathan Wiebe, Daniel Braun, and Seth Lloyd. *Physical review letters*, 109(5):050505, 2012.
 - [18] Andrew M Childs, Robin Kothari, and Rolando D Somma. *arXiv preprint arXiv:1511.02306*, 2015.
 - [19] Seth Lloyd, Masoud Mohseni, and Patrick Rebentrost. *Nature Physics*, 10(9):631–633, 2014.
 - [20] Michael A Nielsen and Isaac Chuang, 2002.
 - [21] Daniel A Lidar and Todd A Brun. Cambridge University Press, 2013.
 - [22] Austin G Fowler, Matteo Mariantoni, John M Martinis, and Andrew N Cleland. *Physical Review A*, 86(3):032324, 2012.
 - [23] Edward Farhi, Jeffrey Goldstone, Sam Gutmann, Joshua Lapan, Andrew Lundgren, and Daniel Preda. *Science*, 292(5516):472–475, 2001.
 - [24] Mark W Johnson, Mohammad HS Amin, Suzanne Gildert, Trevor Lanting, Firas Hamze, Neil Dickson, R Harris, Andrew J Berkley, Jan Johansson, Paul Bunyk, et al. *Nature*, 473(7346):194–198, 2011.
 - [25] C Neill, P Roushan, K Kechedzhi, S Boixo, SV Isakov, V Smelyanskiy, R Barends, B Burkett, Y Chen, Z Chen, et al. *arXiv preprint arXiv:1709.06678*, 2017.
 - [26] Abhinav Kandala, Antonio Mezzacapo, Kristan Temme, Maika Takita, Markus Brink, Jerry M Chow, and Jay M Gambetta. *Nature*, 549(7671):242, 2017.
 - [27] Giuseppe Carleo, Ignacio Cirac, Kyle Cranmer, Laurent Daudet, Maria Schuld, Naftali Tishby, Leslie Vogt-Maranto, and Lenka Zdeborová. *Reviews of Modern Physics*, 91(4):045002, 2019.
 - [28] Louis-François Arsenault, Richard Neuberg, Lauren A Hannah, and Andrew J Millis. *Inverse Problems*, 33(11):115007, 2017.
 - [29] Juan Carrasquilla and Roger G Melko. *Nature Physics*, 13(5):431–434, 2017.
 - [30] Giuseppe Carleo and Matthias Troyer. *Science*, 355(6325):602–606, 2017.
 - [31] Marin Bukov, Alexandre GR Day, Dries Sels, Phillip Weinberg, Anatoli Polkovnikov, and Pankaj Mehta. *Physical Review X*, 8(3):031086, 2018.
 - [32] Savvas Varsamopoulos, Koen Bertels, and Carmen G Almudever. *Quantum Machine Intelligence*, 2(1):1–12, 2020.
 - [33] Ian Goodfellow, Yoshua Bengio, and Aaron Courville.

- <http://www.deeplearningbook.org>, 2016.
- [34] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. In *Proceedings of the 25th international conference on Machine learning*, pages 1096–1103. ACM, 2008.
 - [35] Geoffrey E Hinton, Peter Dayan, Brendan J Frey, and Radford M Neal. *Science*, 268(5214):1158, 1995.
 - [36] Yoshua Bengio, Pascal Lamblin, Dan Popovici, and Hugo Larochelle. In *Advances in neural information processing systems*, pages 153–160, 2007.
 - [37] Vedran Dunjko, Jacob M. Taylor, and Hans J. Briegel. *Physical Review Letters*, 117(13):130501, September 2016.
 - [38] Tadashi Kadowaki and Hidetoshi Nishimori. *Physical Review E*, 58(5):5355, 1998.
 - [39] Giuseppe E Santoro, Roman Martoňák, Erio Tosatti, and Roberto Car. *Science*, 295(5564):2427–2430, 2002.
 - [40] J Brooke, TF Rosenbaum, and G Aepli. *Nature*, 413(6856):610–613, 2001.
 - [41] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. *arXiv preprint arXiv:1411.4028*, 2014.
 - [42] Alberto Peruzzo, Jarrod McClean, Peter Shadbolt, Man-Hong Yung, Xiao-Qi Zhou, Peter J Love, Alán Aspuru-Guzik, and Jeremy L O’Brien. *Nature communications*, 5, 2014.
 - [43] Hartmut Neven, Vasil S Denchev, Geordie Rose, and William G Macready. *arXiv preprint arXiv:0811.0416*, 2008.
 - [44] Vasil S Denchev, Nan Ding, SVN Vishwanathan, and Hartmut Neven. *arXiv preprint arXiv:1205.1148*, 2012.
 - [45] Kristen L Pudenz and Daniel A Lidar. *Quantum information processing*, 12(5):2027–2070, 2013.
 - [46] Alex Mott, Joshua Job, Jean-Roch Vlimant, Daniel Lidar, and Maria Spiropulu. *Nature*, 550(7676):375, 2017.
 - [47] Seth Lloyd, Maria Schuld, Aroosa Ijaz, Josh Izaac, and Nathan Killoran. *arXiv preprint arXiv:2001.03622*, 2020.
 - [48] Ville Bergholm, Josh Izaac, Maria Schuld, Christian Gogolin, Carsten Blank, Keri McKiernan, and Nathan Killoran. *arXiv preprint arXiv:1811.04968*, 2018.
 - [49] Iliaria Gianani, Ivana Mastroserio, Lorenzo Buffoni, Natalia Bruno, Ludovica Donati, Valeria Cimini, Marco Barbieri, Francesco S Cataliotti, and Filippo Caruso. *arXiv preprint arXiv:2106.13835*, 2021.
 - [50] Nicola Dalla Pozza and Filippo Caruso. *Phys. Rev. Research*, 2:043011, 2020.
 - [51] James D. Whitfield, César A. Rodríguez-Rosario, and Alán Aspuru-Guzik. *Physical Review A*, 81:022323, 2010.
 - [52] Filippo Caruso. *New Journal of Physics*, 16(5):055015, May 2014.
 - [53] C. Helstrom. *Mathematics in Science and Engineering : a series of monographs and textbooks*, 1976.
 - [54] Stefano Martina, Stefano Gherardini, and Filippo Caruso. *arXiv preprint arXiv:2101.03221*, 2021.
 - [55] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. MIT Press, 2016. <http://www.deeplearningbook.org>.
 - [56] K. Cho, B.v. Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, 2014.
 - [57] F.A. Gers, J. Schmidhuber, and F. Cummins. *1999 Ninth International Conference on Artificial Neural Networks ICANN 99*, 2:850–855, 1999.
 - [58] C.L. Degen, F. Reinhard, and P. Cappellaro. *Rev. Mod. Phys.*, 89:035002, 2017.
 - [59] JS Otterbach, R Manenti, N Alidoust, A Bestwick, M Block, B Bloom, S Caldwell, N Didier, E Schuyler Fried, S Hong, et al. *arXiv preprint arXiv:1712.05771*, 2017.
 - [60] Meena Mahajan and Venkatesh Raman. *J. Algorithms*, 31(2):335–354, 1999.
 - [61] Diederik P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. In *Advances in Neural Information Processing Systems*, pages 3581–3589, 2014.
 - [62] Amir Khoshaman, Walter Vinci, Brandon Denis, Evgeny Andriyash, and Mohammad H Amin. *Quantum Science and Technology*, 4(1):014001, 2019.
 - [63] Walter Vinci, Lorenzo Buffoni, Hossein Sadeghi, Amir Khoshaman, Evgeny Andriyash, and Mohammad Amin. *Machine Learning: Science and Technology*, 2020.
 - [64] Yann LeCun. <http://yann.lecun.com/exdb/mnist/>, 1998.
 - [65] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. In *Advances in neural information processing systems*, pages 2672–2680, 2014.
 - [66] Ian Goodfellow. *arXiv preprint arXiv:1701.00160*, 2016.
 - [67] Seth Lloyd and Christian Weedbrook. *Physical review letters*, 121(4):040502, 2018.
 - [68] Pierre-Luc Dallaire-Demers and Nathan Killoran. *Physical Review A*, 98(1):012324, 2018.
 - [69] Marcello Benedetti, Erika Lloyd, Stefan Sack, and Mattia Fiorentini. *Quantum Science and Technology*, 4(4):043001, 2019.
 - [70] Alex Pepper, Nora Tischler, and Geoff J Pryde. *Physical review letters*, 122(6):060501, 2019.
 - [71] Marcello Benedetti, Edward Grant, Leonard Wossnig, and Simone Severini. *New Journal of Physics*, 21(4):043023, 2019.
 - [72] Ling Hu, Shu-Hao Wu, Weizhou Cai, Yuwei Ma, Xianghao Mu, Yuan Xu, Haiyan Wang, Yipu Song, Dong-Ling Deng, Chang-Ling Zou, et al. *Science advances*, 5(1):eaav2761, 2019.
 - [73] Paolo Braccia, Filippo Caruso, and Leonardo Banchi. *Eprint arXiv:2012.05996*, 2020.
 - [74] Heinz-Peter Breuer and Francesco Petruccione. Oxford University Press, 2002.
 - [75] Nicola Dalla Pozza, Lorenzo Buffoni, Stefano Martina, and Filippo Caruso. *arXiv preprint arXiv:2108.04490*, 2021.
 - [76] Goran Lindblad. *Communications in Mathematical Physics*, 48(2):119–130, 1976.