



# Challenges and opportunities in quantum machine learning

M. Cerezo<sup>1,2,3</sup>, Guillaume Verdon<sup>4,5,6</sup>, Hsin-Yuan Huang<sup>7,8</sup>, Lukasz Cincio<sup>3,9</sup> and Patrick J. Coles<sup>3,9</sup>✉

**At the intersection of machine learning and quantum computing, quantum machine learning has the potential of accelerating data analysis, especially for quantum data, with applications for quantum materials, biochemistry and high-energy physics. Nevertheless, challenges remain regarding the trainability of quantum machine learning models. Here we review current methods and applications for quantum machine learning. We highlight differences between quantum and classical machine learning, with a focus on quantum neural networks and quantum deep learning. Finally, we discuss opportunities for quantum advantage with quantum machine learning.**

The recognition that the world is quantum mechanical has allowed researchers to embed well established, but classical, theories into the framework of quantum Hilbert spaces. Shannon's information theory, which is the basis of communication technology, has been generalized to quantum Shannon theory (or quantum information theory), opening up the possibility that quantum effects could make information transmission more efficient<sup>1</sup>. The field of biology has been extended to quantum biology to allow for a deeper understanding of biological processes such as photosynthesis, smell and enzyme catalysis<sup>2</sup>. Turing's theory of universal computation has been extended to universal quantum computation<sup>3</sup>, potentially leading to exponentially faster simulations of physical systems.

One of the most successful technologies of this century is machine learning (ML), which aims to classify, cluster and recognize patterns for large datasets. Learning theory has been simultaneously developed alongside of ML technology to understand and improve upon its success. Concepts such as support vector machines, neural networks and generative adversarial networks have impacted science and technology in profound ways. ML is now ingrained into society to such a degree that any fundamental improvement to ML leads to tremendous economic benefit.

Similarly to other classical theories, ML and learning theory can in fact be embedded into the quantum-mechanical formalism. Formally speaking, this embedding leads to the field known as quantum machine learning (QML)<sup>4–6</sup>, which aims to understand the ultimate limits of data analysis allowed by the laws of physics. Practically speaking, the advent of quantum computers, with the hope of achieving a so-called quantum advantage (as defined below) for data analysis, is what has made QML so exciting. Quantum computing exploits entanglement, superposition and interference to perform certain tasks with substantial speedups over classical computing, sometimes even exponentially faster. Indeed, while such speedup has already been observed for a contrived problem<sup>7</sup>, reaching it for data science is still uncertain even at the theoretical level, but this is one of the main goals for QML.

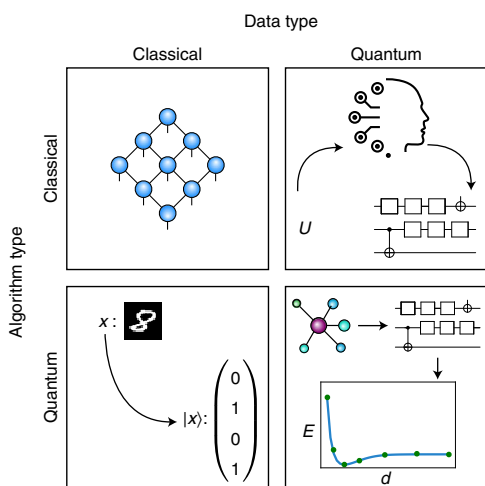
In practice, QML is a broad term that encompasses all of the tasks shown in Fig. 1. For example, ML can be applied to quantum applications such as discovering quantum algorithms<sup>8</sup> or optimizing quantum experiments<sup>9,10</sup>, or a quantum neural network (QNN) can be used to process either classical or quantum information<sup>11</sup>. Even classical tasks can be viewed as QML when they are quantum inspired<sup>12</sup>. We note that the focus of this Perspective will be on QNNs, quantum deep learning and quantum kernels, even though the field of QML is quite broad and goes beyond these topics.

After the invention of the laser, it was called a solution in search of a problem. To some degree, the situation with QML is similar. The complete list of applications of QML is not fully known. Nevertheless, it is possible to speculate that all the areas shown in Fig. 2 will be impacted by QML. For example, QML will likely benefit chemistry, materials science, sensing and metrology, classical data analysis, quantum error correction and quantum algorithm design. Some of these applications produce data that are inherently quantum mechanical, and hence it is natural to apply QML (rather than classical ML) to them.

While there are similarities between classical and quantum ML, there are also some differences. Because QML employs quantum computers, noise from these computers can be a major issue. This includes hardware noise such as decoherence as well as statistical noise (that is, shot noise) that arises from measurements on quantum states. Both of these noise sources can complicate the QML training process. Moreover, nonlinear operations (for example, neural activation functions) that are natural in classical ML require more careful design of QML models due to the linearity of quantum transformations.

For the field of QML, the immediate goal for the near future is demonstrating quantum advantage, that is, outperforming classical methods, in a data science application. Achieving this goal will require keeping an open mind about which applications will benefit most from QML (for example, it may be an application that is inherently quantum mechanical). Understanding how QML methods scale to large problem sizes will also be required, including analysis

<sup>1</sup>Information Sciences, Los Alamos National Laboratory, Los Alamos, NM, USA. <sup>2</sup>Center for Nonlinear Studies, Los Alamos National Laboratory, Los Alamos, NM, USA. <sup>3</sup>Quantum Science Center, Oak Ridge, TN, USA. <sup>4</sup>X, Mountain View, CA, USA. <sup>5</sup>Institute for Quantum Computing, University of Waterloo, Waterloo, Ontario, Canada. <sup>6</sup>Department of Applied Mathematics, University of Waterloo, Waterloo, Ontario, Canada. <sup>7</sup>Institute for Quantum Information and Matter, California Institute of Technology, Pasadena, CA, USA. <sup>8</sup>Department of Computing and Mathematical Sciences, California Institute of Technology, Pasadena, CA, USA. <sup>9</sup>Theoretical Division, Los Alamos National Laboratory, Los Alamos, NM, USA. ✉e-mail: [pcoles@lanl.gov](mailto:pcoles@lanl.gov)



**Fig. 1 | QML tasks.** QML is usually considered for four main tasks. These include tasks where the data are either classical or quantum, and where the algorithm is either classical or quantum. Top left: tensor networks are quantum-inspired classical methods that can analyze classical data. Top right: unitary time-evolution data  $U$  from a quantum system can be classically compiled into a quantum circuit. Bottom left: handwritten digits can be mapped to quantum states for classification on a quantum computer. Bottom right: molecular ground-state data can be classified directly on a quantum computer. The figure shows the dependence of ground-state energy  $E$  on the distance  $d$  between the atoms.

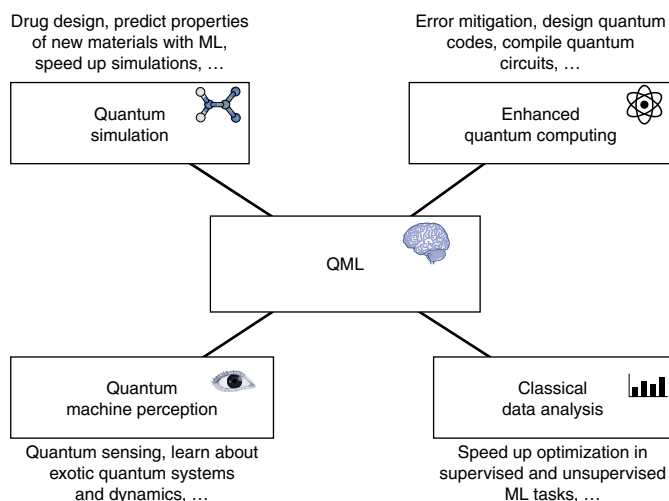
of trainability (gradient scaling) and prediction error. The availability of high-quality quantum hardware<sup>13,14</sup> will also be crucial.

Finally, we note that QML provides a new way of thinking about established fields, such as quantum information theory, quantum error correction and quantum foundations. Viewing such applications from a data science perspective will likely lead to new breakthroughs.

### Framework

**Data.** As shown in Fig. 3, QML can be used to learn from either classical or quantum data, and thus we begin by contrasting these two types of data. Classical data are ultimately encoded in bits, each of which can be in a 0 or 1 state. This includes images, texts, graphs, medical records, stock prices, properties of molecules, outcomes from biological experiments and collision traces from high-energy physics experiments. Quantum data are encoded in quantum bits, called qubits, or higher-dimensional analogs. A qubit can be represented by the states  $|0\rangle$ ,  $|1\rangle$  or any normalized complex linear superposition of these two. Here, the states contain information obtained from some physical process such as quantum sensing<sup>15</sup>, quantum metrology<sup>16</sup>, quantum networks<sup>17</sup>, quantum control<sup>18</sup> or even quantum analog–digital transduction<sup>19</sup>. Moreover, quantum data can also be the solution to problems obtained on a quantum computer: for example, the preparation of various Hamiltonians' ground states.

In principle, all classical data can be efficiently encoded in systems of qubits: a classical bitstring of length  $n$  can be easily encoded onto  $n$  qubits. However, the same cannot be said for the converse, since one cannot efficiently encode quantum data in bit systems; that is, the state of a general  $n$ -qubit system requires  $(2^n - 1)$  complex numbers to be specified. Hence, systems of qubits (and more generally the quantum Hilbert space) constitute the ultimate data representation medium, as they can encode not only classical information but also quantum information obtained from physical processes.



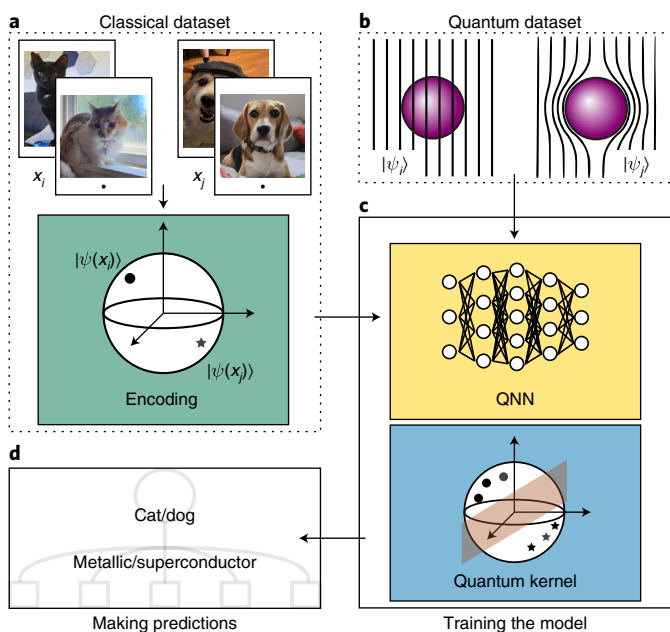
**Fig. 2 | Key applications for QML.** QML has been envisioned to bring a computational advantage in many applications. QML can enhance quantum simulation for chemistry (for example, molecular ground states<sup>110</sup>, equilibrium states<sup>47</sup> and time evolution<sup>112</sup>) and materials science (for example, quantum phase recognition<sup>11</sup> and generative design with a target property in mind<sup>130</sup>). QML can enhance quantum computing by learning quantum error correction codes<sup>11,109</sup> and syndrome decoders, performing quantum control, learning to mitigate errors and compiling and optimizing quantum circuits. QML can enhance sensing and metrology<sup>46,104–107</sup> and extract hidden parameters from quantum systems. Finally, QML may speed up classical data analysis, including clustering and classification.

In a QML setting, the term quantum data refers to data that are naturally already embedded in a Hilbert space  $\mathcal{H}$ . When the data are quantum, they are already in the form of a set of quantum states  $\{|\psi_j\rangle\}$  or a set of unitaries  $\{U_j\}$  that could prepare these states on a quantum device (via the relation  $|\psi_j\rangle = U_j|0\rangle$ ). On the other hand, when the data  $x$  are classical, they first need to be encoded in a quantum system through some embedding mapping  $x_j \rightarrow |\psi(x_j)\rangle$ , with  $|\psi(x_j)\rangle$  in  $\mathcal{H}$ . In this case, the hope is that the QML model can solve the learning task by accessing the exponentially large dimension of the Hilbert space<sup>20–23</sup>.

One of the most important and reasonable conjectures to make is that the availability of quantum data will substantially increase in the near future. The mere fact that people will use the quantum computers that are available will logically lead to more quantum problems being solved and quantum simulations being performed. These computations will produce quantum datasets, and hence it is reasonable to expect the rapid rise of quantum data. Note that, in the near term, these quantum data will be stored on classical devices in the form of efficient descriptions of quantum circuits that prepare the datasets.

Finally, as our level of control over quantum technologies progresses, coherent transduction of quantum information from the physical world to digital quantum computing platforms may be achieved<sup>19</sup>. This would quantum mechanically mimic the main information acquisition mechanism for classical data from the physical world, this being analog–digital conversion. Moreover, we can expect that the eventual advent of practical quantum error correction<sup>24</sup> and quantum memories<sup>25</sup> will allow us to store quantum data on quantum computers themselves.

**Models.** Analyzing and learning from data requires a parameterized model, and many different models have been proposed for QML applications. Classical models such as neural networks and tensor networks (as shown in Fig. 1) are often useful for analyzing



**Fig. 3 | Classification with QML.** **a**, The classical data  $x$ , that is, images of cats and images of dogs, is encoded into a Hilbert space via some map  $x \rightarrow |\psi(x)\rangle$ . Ideally, data from different classes (here represented by dots and stars) are mapped to different regions of the Hilbert space. **b**, Quantum data  $|\psi\rangle$  can be directly analyzed on a quantum device. Here the dataset is composed of states representing metallic or superconducting systems. **c**, The dataset is used to train a QML model. Two common paradigms in QML are QNNs and quantum kernels, both of which allow for classification of either classical or quantum data. In kernel methods we fit a decision hyperplane that separates the classes. **d**, Once the model is trained, it can be used to make predictions.

data from quantum experiments. However, due to their novelty, we will focus our discussion on quantum models using quantum algorithms, where one applies the learning methodology directly at the quantum level.

Similarly to classical ML, there exist several different QML paradigms: supervised learning (task based)<sup>26–28</sup>, unsupervised learning (data based)<sup>29,30</sup> and reinforced learning (reward based)<sup>31,32</sup>. While each of these fields is exciting and thriving in itself, supervised learning has recently received considerable attention for its potential to achieve quantum advantage<sup>26,33</sup>, resilience to noise<sup>34</sup> and good generalization properties<sup>35–37</sup>, which makes it a strong candidate for near-term applications. In what follows we discuss two popular QML models: QNNs and quantum kernels, shown in Fig. 3, with a particular emphasis on QNNs as these are the primary ingredient of several supervised, unsupervised and reinforced learning schemes.

**Quantum neural networks.** The most basic and key ingredient in QML models is parameterized quantum circuits (PQCs). These involve a sequence of unitary gates acting on the quantum data states  $|\psi_i\rangle$ , some of which have free parameters  $\theta$  that will be trained to solve the problem at hand<sup>38</sup>. PQCs are conceptually analogous to neural networks, and indeed this analogy can be made precise: that is, classical neural networks can be formally embedded into PQCs<sup>39</sup>.

This has led researchers to refer to certain kinds of PQC as QNNs. In practice, the term QNN is used whenever a PQC is employed for a data science application, and hence we will use the term QNN in what follows. QNNs are employed in all three QML paradigms mentioned above. For instance, in a supervised classification task, the goal of the QNN is to map the states in different

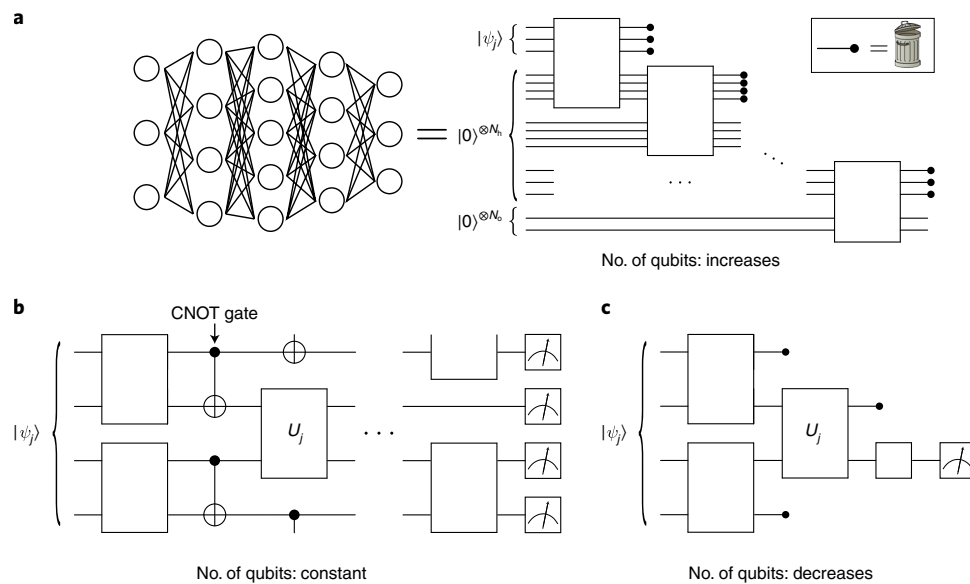
classes to distinguishable regions of the Hilbert space<sup>26</sup>. Moreover, in the unsupervised learning scenario of ref. <sup>29</sup>, a clustering task is mapped onto a MAXCUT problem and solved by training a QNN to maximize distance between classes. Finally, in the reinforced learning task of ref. <sup>32</sup>, a QNN can be used as the  $Q$ -function approximator, which can be used to determine the best action for a learning agent given its current state.

Figure 4 gives examples of three distinct QNN architectures where in each layer the number of qubits in the model is increased, preserved or decreased. In Fig. 4a we show a dissipative QNN<sup>40</sup> which generalizes the classical feedforward network. Here, each node corresponds to a qubit, while lines connecting qubits are unitary operations. The term dissipative arises from the fact that qubits in a layer are discarded after the information forward-propagates to the (new) qubits in the next layer. Figure 4b shows a standard QNN where quantum data states are sent through a quantum circuit, at the end of which some or all of the qubits are measured. Here, no qubits are discarded or added as we go deeper into the QNN. Finally, Fig. 4c depicts a convolutional QNN<sup>11</sup>, where in each layer qubits are measured to reduce the dimension of the data while preserving its relevant features. Many other QNNs have been proposed<sup>41–45</sup>, and constructing QNN architectures is currently an active area of research.

To further accommodate the limitation of near-term quantum computers, we can also employ a hybrid approach with models that have both classical and quantum neural networks<sup>46</sup>. Here, QNNs act coherently on quantum states while deep classical neural networks alleviate the need for higher-complexity quantum processing. Such hybridization distributes the representational capacity and computational complexity across both quantum and classical computers. Moreover, since quantum states generally have a mixture of classical correlations and quantum correlations, hybrid quantum–classical models allow for the use of quantum computers as an additive resource to increase the ability of classical models to represent quantum-correlated distributions. Applications of hybrid models include generating<sup>47</sup> or learning and distilling information<sup>46</sup> from multipartite-entangled distributions.

**Quantum kernels.** As an alternative to QNNs, researchers have proposed quantum versions of kernel methods<sup>26,28</sup>. A kernel method maps each input to a vector in a high-dimensional vector space, known as the reproducing kernel Hilbert space. Then, a kernel method learns a linear function in the reproducing kernel Hilbert space. The dimension of the reproducing kernel Hilbert space could be infinite, which enables the kernel method to be very powerful in terms of expressiveness. To learn a linear function in a potentially infinite-dimensional space, the kernel trick<sup>48</sup> is employed, which only requires efficient computation of the inner product between these high-dimensional vectors. The inner product is also known as the kernel<sup>48</sup>. Quantum kernel methods consider the computation of kernel functions using quantum computers. There are many possible implementations. For example, refs. <sup>26,28</sup> considered a reproducing kernel Hilbert space equal to the quantum state space, which is finite dimensional. Another approach<sup>13</sup> is to study an infinite-dimensional reproducing kernel Hilbert space that is equivalent to transforming a classical vector using a quantum computer. It then maps the transformed classical vectors to infinite-dimensional vectors.

**Inductive bias.** For both QNNs and quantum kernels, an important design criterion is their inductive bias. This bias refers to the fact that any model represents only a subset of functions and is naturally biased towards certain types of function (that is, functions relating the input features to the output prediction). One aspect of achieving quantum advantage with QML is to aim for QML models with an inductive bias that is inefficient to simulate with a classical model.



**Fig. 4 | Examples of QNN architectures.** **a**, A classical feedforward neural network has input, hidden and output layers. This can be generalized to the quantum setting with a dissipative QNN, where some qubits are discarded and replaced by new qubits during the algorithm. Here we show a quantum circuit representation for the dissipative QNN. In a circuit diagram each horizontal line represents a qubit, and the logical operations, or quantum gates, are represented by boxes connecting the qubit lines. Circuits are read from left to right. For instance, here the circuit is initialized in a product state  $|\psi_j\rangle \otimes |0\rangle^{\otimes(N_h+N_o)}$ , where  $|\psi_j\rangle$  encodes the input data and  $N_h$  ( $N_o$ ) is the number of ancilla qubits in the hidden (output) layer, which are initialized to the fiducial state  $|0\rangle$ . As logical operations are performed, the information forward-propagates through the circuit into the ancillary qubits. **b**, Another possible QNN strategy is to keep the qubits fixed, without discarding or replacing them. The circuit represents consecutive application of two-qubit gates  $U_j$  and controlled-NOT (denoted by CNOT) gates. **c**, QCNNs measure and discard qubits during the algorithm. The QCNN circuit considered here is built with two-qubit quantum gates  $U_j$  and is initialized in  $|\psi_j\rangle$ .

Indeed, it was recently shown<sup>49</sup> that quantum kernels with this property can be constructed, albeit with some subtleties regarding their trainability.

Generally speaking, inductive bias encompasses any assumptions made in the design of the model or the optimization method that bias the search of the potential models to a subset in the set of all possible models. In the language of Bayesian probabilistic theory, we usually call these assumptions our prior. Having a certain parameterization of potential models, such as QNNs, or choosing a particular embedding for quantum kernel methods<sup>13,14,26</sup> is itself a restriction of the search space, and hence a prior. Adding a regularization term to the optimizer or modulating the learning rate to keep searches geometrically local also adds inherently a prior and focuses the search, and thus provides inductive bias.

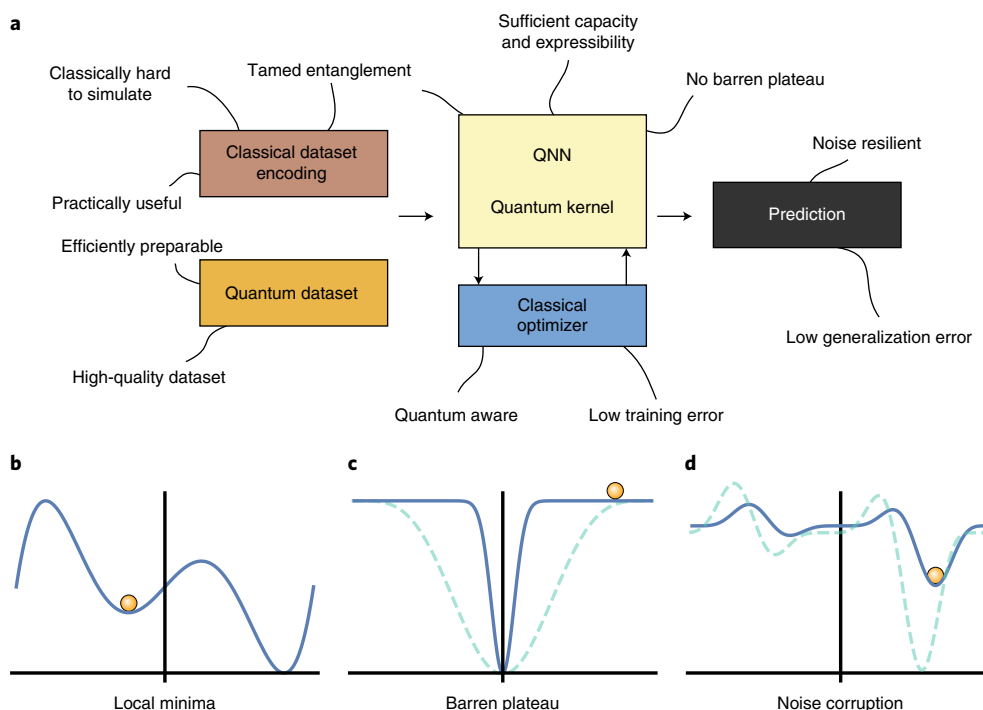
Ultimately, inductive biases from the design of the ML model, combined with a choice of training process, are what make or break an ML model. The main advantage of QML will then be to have the ability to sample from and learn models that are (at least partially) natively quantum mechanical. As such, they have inductive biases that classical models do not have. This discussion assumes that the dataset to be represented is quantum mechanical in nature, and is one of the reasons why researchers typically believe that QML has greater promise for quantum rather than classical data.

**Training and generalization.** The ultimate goal of ML (classical or quantum) is to train a model to solve a given task. Thus, understanding the training process of QML models is fundamental for their success.

Consider the training process, whereby we aim to find the set of parameters  $\theta$  that lead to the best performance. The latter can be accomplished, for instance, by minimizing a loss function  $\mathcal{L}(\theta)$  encoding the task at hand. Some methods for training QML models are leveraged from classical ML, such as stochastic gradient descent. However, shot noise, hardware noise and unique landscape

features often make off-the-shelf classical optimization methods perform poorly for QML training. (This is due to the fact that extracting information from a quantum state requires computing the expectation values of some observable, which in practice need to be estimated via measurements on a noisy quantum computer. Hence, given a finite number of shots (measurement repetitions), these can only be resolved up to some additive errors. Moreover, such expectation values will be subject to corruption due to hardware noise.) This realization led to development of quantum-aware optimizers, which account for the quantum idiosyncrasies of the QML training process. For example, shot-frugal optimizers<sup>50–53</sup> can employ stochastic gradient descent while adapting the number of shots (or measurements) needed at each iteration, so as not to waste too many shots during the optimization. Quantum natural gradient<sup>54,55</sup> adjusts the step size according to the local geometry of the landscape (on the basis of the quantum Fisher information metric). These and other quantum-aware optimizers often outperform standard classical optimization methods in QML training tasks.

For the case of supervised learning, we are interested not only in learning from a training dataset but also in making accurate predictions on (generalizing to) previously unseen data. This translates into achieving small training and prediction errors, with the second usually hinging on the first. Thus, let us now consider prediction error, also known as generalization error, which has been studied only very recently for QML<sup>13,14,35,37,56,57</sup>. Formally speaking, this error measures the extent to which a trained QML model performs well on unseen data. Prediction error depends on the training error as well as the complexity of the trained model. If the training error is large, the prediction error is also typically large. If the training error is small but the complexity of the trained model is large, then the prediction error is likely still large. The prediction error is small only if training error is itself small and the complexity of the trained model is moderate (that is, sufficiently smaller than training data size)<sup>14,35</sup>. The notion of



**Fig. 5 | Challenges for QML.** **a**, There are several ingredients and priors needed to build a QML model: a dataset (and an encoding scheme for classical data), the choice of parameterized model, loss function and classical optimizer. In this diagram, we show some of the challenges of the different components of the model. **b-d**, The success of the QML model hinges on an accurate and efficient training of the parameters. However, there are certain phenomena that can hinder the QML trainability. These include the abundance of low-quality local minimum solutions (**b**), as well as the barren plateau phenomenon (**c**). When a QML architecture exhibits a barren plateau, the landscape becomes exponentially flat (on average) as the number of qubits increases (seen as a transition from dashed to solid line). The presence of hardware noise has been shown to erase the features in the landscape as well as potentially shifting the position of the minima. Here, the dashed (solid) line corresponds to the noiseless (noisy) landscape shown in **d**.

complexity depends on the QML model. We have a good understanding of the complexity of quantum kernel methods<sup>13,14</sup>, while more research is needed on QNN complexity. Recent theoretical analysis of QNNs shows that their prediction performance is closely linked to the number of independent parameters in the QNN, with good generalization obtained when the number of training data is roughly equal to the number of parameters<sup>35</sup>. This gives the exciting prospect of using only a small number of training data to obtain good generalization.

### Challenges in QML

Heuristic fields can face periods of stagnation (or ‘winters’) due to unforeseen technical challenges. Indeed in classical ML, there was a gap between introducing a single perceptron<sup>58</sup> and the multilayer perceptron<sup>59</sup> (that is, neural network), and there was also a gap between attempts to train multiple layers and the introduction of the backpropagation method<sup>60</sup>.

Naturally we would like to avoid these stagnations or winters for QML. The obvious strategy is to try to determine all of the challenges as quickly as possible, and focus research effort on addressing them. Fortunately, QML researchers have adopted this strategy. Figure 5 showcases some of the different elements of QML models, as well as the challenges associated with them. In this section we detail various QML challenges and how they could potentially be avoided.

**Embedding schemes and quantum datasets.** The access to high-quality, standardized datasets has played a key role in advancing classical ML. Hence, one could conjecture that such datasets will be crucial for QML as well.

Currently, most QML architectures are benchmarked using classical datasets (such as MNIST, Dogs vs Cats and Iris). While using classical datasets is natural due to their accessibility, it is still unclear how to best encode classical information onto quantum states. Several embedding schemes have been proposed<sup>22,26,61</sup>, and there are some properties they must possess. One such property is that the inner product between output states of the embedding is classically hard to simulate (otherwise the quantum kernel would be classically simulable). In addition, the embedding should be practically useful: that is, in a classification task, the states should be in distinguishable regions of the Hilbert space. Unfortunately, embeddings that satisfy one of these properties do not necessarily satisfy the other<sup>62</sup>. Thus, developing encoding schemes is an active area of research, especially those that are equipped with an inductive bias containing information about the dataset<sup>49</sup>.

Furthermore, some recent results suggest that achieving a quantum advantage with classical data might not be straightforward<sup>49</sup>. On the other hand, QML models with quantum data have a more promising route towards a quantum advantage<sup>63–66</sup>. Despite this fact, there is still a dearth of truly quantum datasets for QML, with just a few recently proposed<sup>67,68</sup>. Hence, the field needs standardized quantum datasets with easily preparable quantum states, as these can be used to benchmark QML models on true quantum data.

**Quantum landscapes.** Training the parameters of the QML model corresponds in a wide array of cases to minimizing a loss function and navigating through a (usually non-convex) loss function landscape in search of its global minimum. Technically speaking, the loss function defines a map from the model’s parameter space to the real values. The loss function value can quantify, for instance, the

model's error in solving a given task, so our goal is to find the set of parameters that minimizes such error. Quantum landscape theory<sup>69</sup> aims to understand QML landscape properties and how to engineer them. Local minima and barren plateaus have received substantial attention in quantum landscape theory.

**Local minima in quantum landscapes.** As schematically shown in Fig. 5b, similarly to classical ML, the quantum loss landscape can have many local minima. Ultimately, this can lead to the overall non-convex optimization being NP-hard<sup>70</sup>, which is again similar to the classical case. There have been some methods proposed to address local minima. For example, variable structure QNNs<sup>71,72</sup>, which grow and contract throughout the optimization, adaptively change the model's prior and allow some local minima to be turned into saddle points. Moreover, evidence of the overparametrization phenomenon has been seen for QML<sup>73,74</sup>. Here, the optimization undergoes a computational phase transition, due to spurious local minima disappearing, whenever the number of parameters exceeds a critical value.

**Overview of barren plateaus.** Local minima are not the only issue facing QML, as it has been shown that quantum landscapes can exhibit a fascinating property known as a barren plateau<sup>37,75–87</sup>. As depicted in Fig. 5c, in a barren plateau the loss landscape becomes, on average, exponentially flat with the problem size. When this occurs, the valley containing the global minimum also shrinks exponentially with problem size, leading to a so-called narrow gorge<sup>69</sup>. As a consequence, exponential resources (for example, numbers of shots) are required to navigate through the landscape. The latter impacts the complexity of the QML algorithm and can even destroy quantum speedup, since quantum algorithms typically aim to avoid the exponential complexity normally associated with classical algorithms.

**Barren plateaus from ignorance or insufficient inductive bias.** The barren plateau phenomenon has been studied in deep hardware-efficient QNNs<sup>75</sup>, where they arise due to the high expressivity of the model<sup>79</sup>. By making no assumptions about the underlying data, deep hardware-efficient architectures aim to solve a problem by being able to prepare a wide range of unitary evolutions. In other words, the prior over hypothesis space is relatively uninformed. Barren plateaus in this unsharp prior are caused by ignorance or the lack of sufficient inductive bias, and therefore a means to avoid them is to input knowledge into the construction of the QNN—making the design of QNNs with good inductive biases for the problem at hand a key solution.

Fortunately various strategies have been developed to address these barren plateaus, such as clever initialization<sup>88</sup>, pretraining and parameter correlation<sup>80,81</sup>. These are all examples of adding a sharper prior to the search over the overexpressive parameterizations of hardware-efficient QNNs. Below we further discuss how QNN architectures can be designed to further introduce inductive bias.

**Barren plateaus from global observables.** Other mechanisms have been linked to barren plateaus. Simply defining a loss function based on a global observable (that is, observables measuring all qubits) leads to barren plateaus even for shallow circuits with sharp priors<sup>76</sup>, while local observables (those comparing quantum states at the single-qubit level) avoid this issue<sup>76,85</sup>. The latter is due not to bad inductive biases but rather to the fact that comparing objects in exponentially large Hilbert spaces requires an exponential precision, as their overlap is usually exponentially small.

**Barren plateaus from entanglement.** While entanglement is one of the most important quantum resources for information processing tasks in quantum computers, it can also be detrimental for QML

models. QNNs (or embedding schemes) that generate too much entanglement also lead to barren plateaus<sup>82,84,86</sup>. Here, the issue arises when the visible qubits of the QNN (those that are measured at the QNN's output) are entangled with a large number of qubits in the hidden layers. Due to entanglement, the information of the state is stored in non-local correlations across all qubits, and hence the reduced state of the visible qubits concentrates around the maximally mixed state. This type of barren plateau can be solved by taming the entanglement generated across the QNN.

**QNN architecture design.** One of the most active areas is developing QNN architectures that have sharp priors. Since QNNs are a fundamental ingredient in supervised learning (deep learning, kernel methods), but also in unsupervised learning and reinforced learning, developing good QNN architectures is crucial for the field.

For instance, it has been shown that QNNs with sharp priors can avoid issues such as barren plateaus altogether. One such example is quantum convolutional neural networks (QCNNs)<sup>11</sup>. QCNNs possess an inductive bias from having a prior over the space of architectures that is much sharper than that of deep hardware-efficient architectures, as QCNNs are restricted to be hierarchically structured and translationally invariant. The notable reduction in the expressivity and parameter space dimension from this translational invariance assumption yields the greater trainability<sup>80</sup>.

The idea of embedding knowledge about the problem and dataset into our models (to achieve helpful inductive bias) will be key to improve the trainability of QML models. Recent proposals use quantum graph neural networks<sup>89</sup> for scenarios where quantum subsystems live on a graph, and potentially have further symmetries. For instance, the underlying graph-permutation symmetries of a quantum communication dataset were taken into account by a quantum graph convolutional network. Similarly, a quantum recurrent neural network has been used in scenarios where temporal recurrence of parameters occurs—for example, in the quantum dynamics of a stationary (time-dependent) quantum dynamical process.

To better understand how to go beyond the aforementioned inductive biases from temporal and/or translational invariance in grids and graphs, we can take inspiration from recent advances in the theory of classical deep learning. In classical ML, the study of the group theory behind graph neural networks, namely the concepts of invariance and equivariance to various group actions on the input space, has led to a unifying theory of deep learning architectures based on group theory, called geometric deep learning theory<sup>90</sup>.

To have a prescription to create arbitrary architectures and inductive biases suitable for a given set of quantum physical data, a theory of quantum geometric deep learning could be key to design architectures with the right prior over the transformation space and inductive biases to ensure trainability and generalization. As the study of physics is often about the identification of inherent or emergent symmetries in particular systems, there is great potential for a future unifying theory of quantum geometric deep learning to provide consistent methods to create QML model architectures with inductive biases encoding knowledge of the basic symmetries and principles of the quantum physical system underlying given quantum datasets. This approach has been recently explored in refs. 91–93. Moreover, the works<sup>74,94</sup> have also shown that the Lie algebra obtained from the generators of the QNN can be linked to properties of the QML landscape such as the presence of barren plateaus or the overparametrization phenomenon.

**Effect of quantum noise.** The presence of hardware noise during quantum computations is one of the defining characteristics of noisy intermediate-scale quantum (NISQ) computing. Despite this fact, most QML research neglects noise in the analytical calculations and numerical simulations while still promising that the methods are near-term compatible. Accounting for the effects of hardware noise

should be a crucial aspect of QML analysis if we wish to pursue a quantum advantage with currently available hardware.

Noise corrupts the information as it forward-propagates in a quantum circuit, meaning that deeper circuits with longer run-times will be particularly affected. As such, noise affects all aspects of the model that make use of quantum computers. This includes the dataset preparation scheme as well as circuits used to compute quantum kernels. Moreover, when using QNNs, noise can hinder their trainability as it leads to noise-induced barren plateaus<sup>87,95</sup>. Here, the relevant features of the landscape become exponentially suppressed by noise as the depth of the circuit increases (Fig. 5d). Ultimately, the effects of noise translate into a deformation of the inductive bias of the model from its original one, and an effective reduction of the dimension of the quantum feature space. Despite the critical impact of quantum noise, its effects are still largely unexplored, particularly its impact on the classical simulability of the QML model<sup>96,97</sup>.

Addressing noise-induced issues will likely require either (1) reduction in hardware error rates, (2) partial quantum error correction<sup>98</sup> or (3) employing QNNs that are relatively shallow (that is, whose depth grows sublinearly in the problem size)<sup>87</sup>, such as QCNNs. Error mitigation techniques<sup>99–101</sup> can also improve performance of QML models in the presence of noise, although they may not solve noise-induced trainability issues<sup>95</sup>. A different approach to dealing with noise is to engineer QML models with noise-resilient properties<sup>34,102,103</sup> (such as the position of the minima not changing due to noise).

## Outlook

**Potential for quantum advantage.** The first quantum advantages in QML will likely arise from hidden parameter extraction from quantum data. This can be for quantum sensing or quantum state classification/regression. Fundamentally, we know from the theory of optimal measurement that non-local quantum measurements can extract hidden parameters using fewer samples. Using QML, one can form and search over a parameterization of hypotheses for such measurements.

This is particularly useful when such optimal measurements are not known a priori—for example, identifying the measurement that extracts an order parameter or identifies a particular phase of matter. As the information about this classical parameter is embedded in the structure of quantum correlations between subsystems, it is natural that a trained QML model with good inductive biases can exhibit an advantage over local measurements and classical representations.

Another area of application where classical parameter extraction may yield an advantage is in quantum machine perception<sup>46,63,104–107</sup>, that is, quantum sensing, metrology and beyond. Here, leveraging the variational search over multipartite-entangled states for input to exposure to a quantum signal along with the optimization for optimal control and/or over post-processing schemes can find optimal measurements for the estimation of hidden parameters in the incoming signal. In particular, the variational approach may be able to find the optimal entanglement, exposure and measurement scheme that filters signal from noise<sup>108</sup>, akin to variationally learning the quantum error correcting code that filters signal from noise, but instead applied to quantum metrology.

Beyond classical parameter extraction embedded in quantum data, there may be an advantage for the discovery of quantum error correcting codes<sup>109</sup>. Quantum error correcting codes fundamentally encode data (typically) non-locally into a subsystem or subspace of the Hilbert space. As deep learning is fundamentally about the discovery of submanifolds of data space, identifying and decoding subspaces/subsystems from a Hilbert space that correspond to a quantum error correction subspace/subsystem is a natural area where differentiable quantum computing may yield an advantage.

This is barely explored, mainly due to the difficulty of gaining insights with small-scale numerical simulations. Fundamentally, it is akin to a quantum data version of classical parameter embedding/extraction advantage.

Finally, a quantum advantage for generative modeling may be achieved when ground states<sup>110</sup>, equilibrium states<sup>47,111</sup> or quantum dynamics<sup>112</sup> can be generated using models incorporating QNNs, in a situation where the distribution cannot be sampled classically, and yields more accurate predictions or more extensive generalization compared with classical ML approaches. The nearest-term possibility for demonstrating such an advantage would likely be from variational optimization at the continuous time optimal control level on analog quantum simulators.

**What will quantum advantage look like?** When the data originate from quantum-mechanical processes, such as from experiments in chemistry, material science, biology and physics, it is more likely to see exponential quantum advantage in ML. The quantum advantage could be in sample complexity or time complexity. An exponential advantage in sample complexity always implies an exponential advantage in time complexity, but the reverse is not generally true. It was recently shown<sup>63,65,113,114</sup> that there is an exponential quantum advantage in sample complexity when we can use a quantum sensor, quantum memory and quantum computer to retrieve, store and process quantum information from experiments. Such a sample complexity advantage can be proven rigorously without the possibility of being dequantized<sup>12,64,115</sup> in the future, that is, it is impossible to find improved classical algorithms such that there is no exponential advantage. This substantial quantum advantage has recently been demonstrated on the Sycamore processor<sup>63</sup> raising the hope for achieving quantum advantage using NISQ devices<sup>116</sup>.

The situation for advantage in time complexity is more subtle. Classical simulation of quantum process is intractable in many cases, hence exponential advantage in time complexity would be expected to be prevalent. However, we should be cautious about the availability of data in ML tasks, which makes classical ML algorithms computationally more powerful<sup>13,117</sup>. For instance, ref. <sup>117</sup> shows that in the worst case there is no exponential quantum advantage in predicting ground-state properties in geometrically local gapped Hamiltonians. Furthermore, the emergence of effective classical theory in quantum-mechanical processes could enable classical machines to provide accurate predictions. For example, density functional theory<sup>118,119</sup> allows accurate prediction of molecular properties when we have an accurate approximation to the exchange–correlation functionals by conducting real-world experiments. It is still likely that an exponential advantage is possible in physical systems of practical interest, but there are no rigorous proofs yet.

When the data are of a purely classical origin, such as in applications for recommending products to customers<sup>12</sup>, performing portfolio optimization<sup>120,121</sup> and processing human languages<sup>122</sup> and everyday images<sup>123</sup>, there is no known exponential advantage<sup>115</sup>. However, it is still reasonable to expect polynomial advantage. Furthermore, a quadratic advantage can be rigorously proven<sup>124,125</sup> for purely classical problems. Therefore, we likely have a potential impact in the long term when we have fault-tolerant quantum computers, albeit with the speedup notably dampened by the overheads of quantum error correction<sup>126</sup> for currently known fault-tolerant quantum computing schemes.

**Transition to the fault-tolerant era and beyond.** While QML has been proposed as a candidate to achieve a quantum advantage in the near term using NISQ devices, we can still pose a question about its usability in the future. Here, researchers envision two different chronological eras post-NISQ. In the first, which we can refer to as ‘partial error corrected’, quantum computers will have enough

physical qubits (a couple of hundred), and sufficiently small error rates, to allow for a small number of fully error-corrected logical qubits. Since one logical qubit is comprised of multiple physical qubits, in this era we will have the freedom to trade off and split the qubits in the device into a subset of error-corrected qubits along with a subset of non-error-corrected qubits. The next era, that is, the ‘fault-tolerant’ era, will arise when the quantum hardware has a large number of error-corrected qubits.

Indeed, we can easily envision QML being useful in both of these post-NISQ eras. First, in the partial error-corrected era, QML models will be able to execute high-fidelity circuits and thus have an improved performance. This will naturally enhance the trainability of the models by mitigating noise-induced barren plateaus, and also reduce noise-induced classification errors in QML models. Most importantly, QML will likely see its most widespread and critical use during the fault-tolerant era. Here, quantum algorithms such as those for quantum simulation<sup>127,128</sup> will be able to accurately prepare quantum data, and to faithfully store it in quantum memories<sup>129</sup>. Therefore QML will be the natural model to learn, infer and make predictions from quantum data, as here the quantum computer will learn from the data themselves directly.

On the further-term horizon, we anticipate it will be possible to capture quantum data from nature directly via transduction from their natural analog form to one that is quantum digital (for example, via quantum analog–digital interconversion<sup>19</sup>). These data will then be able to be shuttled around quantum networks for distributed and/or centralized processing with QML models, using fault-tolerant quantum computation and error-corrected quantum communication. At this point, QML will have reached a stage similar to that where ML is today, where edge sensors capture data, the data are relayed to a central cloud and ML models are trained on the aggregated data. As the modern advent of widespread classical ML arose at this point of abundant data, one could anticipate that ubiquitous access to quantum data in the fault-tolerant era could similarly propel QML to even greater widespread use.

Received: 5 January 2022; Accepted: 4 August 2022;  
Published online: 15 September 2022

## References

- Nielsen, M. A. & Chuang, I. L. *Quantum Computation and Quantum Information* (Cambridge Univ. Press, 2000).
- Brookes, J. C. Quantum effects in biology: golden rule in enzymes, olfaction, photosynthesis and magnetodetection. *Proc. R. Soc. A* **473**, 20160822 (2017).
- Deutsch, D. Quantum theory, the Church–Turing principle and the universal quantum computer. *Proc. R. Soc. A* **400**, 97–117 (1985).
- Wiebe, N., Kapoor, A. & Svore, K. M. Quantum deep learning. Preprint at <https://arxiv.org/abs/1412.3489> (2014).
- Schuld, M., Sinayskiy, I. & Petruccione, F. An introduction to quantum machine learning. *Contemp. Phys.* **56**, 172–185 (2015).
- Biamonte, J. et al. Quantum machine learning. *Nature* **549**, 195–202 (2017).
- Arute, F. Quantum supremacy using a programmable superconducting processor. *Nature* **574**, 505–510 (2019).
- Cincio, L., Subaşı, Y., Sornborger, A. T. & Coles, P. J. Learning the quantum algorithm for state overlap. *New J. Phys.* **20**, 113022 (2018).
- Tranter, A. D. Multiparameter optimisation of a magneto-optical trap using deep learning. *Nat. Commun.* **9**, 4360 (2018).
- Kaubruegger, R., Vasilyev, D. V., Schulte, M., Hammerer, K. & Zoller, P. Quantum variational optimization of Ramsey interferometry and atomic clocks. *Phys. Rev. X* **11**, 041045 (2021).
- Cong, I., Choi, S. & Lukin, M. D. Quantum convolutional neural networks. *Nat. Phys.* **15**, 1273–1278 (2019).
- Tang, E. A quantum-inspired classical algorithm for recommendation systems. In *Proc. 51st Annual ACM SIGACT Symposium on Theory of Computing* 217–228 (Association for Computing Machinery, 2019).
- Huang, H.-Y. Power of data in quantum machine learning. *Nat. Commun.* **12**, 2631 (2021).
- Banchi, L., Pereira, J. & Pirandola, S. Generalization in quantum machine learning: a quantum information standpoint. *PRX Quantum* **2**, 040321 (2021).
- Degen, C. L., Reinhard, F. & Cappellaro, P. Quantum sensing. *Rev. Mod. Phys.* **89**, 035002 (2017).
- Giovannetti, V., Lloyd, S. & Maccone, L. Advances in quantum metrology. *Nat. Photon.* **5**, 222–229 (2011).
- Chiribella, G., D’Ariano, G. M. & Perinotti, P. Theoretical framework for quantum networks. *Phys. Rev. A* **80**, 022339 (2009).
- D’Alessandro, D. *Introduction to Quantum Control and Dynamics* (Chapman & Hall/CRC Applied Mathematics & Nonlinear Science, Taylor & Francis, 2007).
- Verdon-Akzam, G. Quantum analog–digital interconversion for encoding and decoding quantum signals. US patent application 17,063,595 (2020).
- Rebentrost, P., Mohseni, M. & Lloyd, S. Quantum support vector machine for big data classification. *Phys. Rev. Lett.* **113**, 130503 (2014).
- Schuld, M. & Killoran, N. Quantum machine learning in feature Hilbert spaces. *Phys. Rev. Lett.* **122**, 040504 (2019).
- Lloyd, S., Schuld, M., Ijaz, A., Izaac, J. & Killoran, N. Quantum embeddings for machine learning. Preprint at <https://arxiv.org/abs/2001.03622> (2020).
- Schuld, M., Sweke, R. & Meyer, J. J. Effect of data encoding on the expressive power of variational quantum-machine-learning models. *Phys. Rev. A* **103**, 032430 (2021).
- Roffe, J. Quantum error correction: an introductory guide. *Contemp. Phys.* **60**, 226–245 (2019).
- Shor, P. W. Scheme for reducing decoherence in quantum computer memory. *Phys. Rev. A* **52**, R2493 (1995).
- Havlíček, V. Supervised learning with quantum-enhanced feature spaces. *Nature* **567**, 209–212 (2019).
- Liu, Y., Arunachalam, S. & Temme, K. A rigorous and robust quantum speed-up in supervised machine learning. *Nat. Phys.* **17**, 1013–1017 (2021).
- Schuld, M. Supervised quantum machine learning models are kernel methods. Preprint at <https://arxiv.org/abs/2101.11020> (2021).
- Otterbach, J. S. et al. Unsupervised machine learning on a hybrid quantum computer. Preprint at <https://arxiv.org/abs/1712.05771> (2017).
- Kerenidis, I., Landman, J., Luongo, A. & Prakash, A. q-means: a quantum algorithm for unsupervised machine learning. In *Advances in Neural Information Processing Systems* Vol. 32 (eds Wallach, H. M. et al.) 4136–4146 (Curran, 2019).
- Saggio, V. Experimental quantum speed-up in reinforcement learning agents. *Nature* **591**, 229–233 (2021).
- Skolik, A., Jerbi, S. & Dunjko, V. Quantum agents in the gym: a variational quantum algorithm for deep q-learning. *Quantum* **6**, 720 (2022).
- Huang, H.-Y. et al. Quantum advantage in learning from experiments. *Science* **376**, 1182–1186 (2022).
- LaRose, R. & Coyle, B. Robust data encodings for quantum classifiers. *Phys. Rev. A* **102**, 032420 (2020).
- Caro, M. C. et al. Generalization in quantum machine learning from few training data. *Nat. Commun.* **13**, 4919 (2022).
- Caro, M. C. et al. Out-of-distribution generalization for learning quantum dynamics. Preprint at <https://arxiv.org/abs/2204.10268> (2022).
- Caro, M. C., Gil-Fuster, E., Meyer, J. J., Eisert, J. & Sweke, R. Encoding-dependent generalization bounds for parametrized quantum circuits. *Quantum* **5**, 582 (2021).
- Cerezo, M. Variational quantum algorithms. *Nat. Rev. Phys.* **3**, 625–644 (2021).
- Wan, K. H., Dahlsten, O., Kristjánsson, H., Gardner, R. & Kim, M. S. Quantum generalisation of feedforward neural networks. *npj Quantum Inf.* **3**, 36 (2017).
- Beer, K. Training deep quantum neural networks. *Nat. Commun.* **11**, 808 (2020).
- Schuld, M., Sinayskiy, I. & Petruccione, F. The quest for a quantum neural network. *Quantum Inf. Process.* **13**, 2567–2586 (2014).
- Dallaire-Demers, P.-L. & Killoran, N. Quantum generative adversarial networks. *Phys. Rev. A* **98**, 012324 (2018).
- Farhi, E. & Neven, H. Classification with quantum neural networks on near term processors. Preprint at <https://arxiv.org/abs/1802.06002> (2018).
- Killoran, N. Continuous-variable quantum neural networks. *Phys. Rev. Res.* **1**, 033063 (2019).
- Bausch, J. Recurrent quantum neural networks. *Adv. Neural. Inf. Process. Syst.* **33**, 1368–1379 (2020).
- Broughton, M. et al. TensorFlow Quantum: a software framework for quantum machine learning. Preprint at <https://arxiv.org/abs/2003.02989> (2020).
- Verdon, G., Marks, J., Nanda, S., Leichenauer, S. & Hidary, J. Quantum Hamiltonian-based models and the variational quantum thermalizer algorithm. Preprint at <https://arxiv.org/abs/1910.02071> (2019).
- Cortes, C. & Vapnik, V. Support-vector networks. *Mach. Learn.* **20**, 273–297 (1995).
- Kübler, J. M., Buchholz, S. & Schölkopf, B. The inductive bias of quantum kernels. *Adv. Neural. Inf. Process. Syst.* **34**, 12661–12673 (2021).



50. Kübler, J. M., Arrasmith, A., Cincio, L. & Coles, P. J. An adaptive optimizer for measurement-frugal variational algorithms. *Quantum* **4**, 263 (2020).
51. Arrasmith, A., Cincio, L., Somma, R. D. & Coles, P. J. Operator sampling for shot-frugal optimization in variational algorithms. Preprint at <https://arxiv.org/abs/2004.06252> (2020).
52. Gu, A., Lowe, A., Dub, P. A., Coles, P. J. & Arrasmith, A. Adaptive shot allocation for fast convergence in variational quantum algorithms. Preprint at <https://arxiv.org/abs/2108.10434> (2021).
53. Sweke, R. Stochastic gradient descent for hybrid quantum–classical optimization. *Quantum* **4**, 314 (2020).
54. Stokes, J., Izaac, J., Killoran, N. & Carleo, G. Quantum natural gradient. *Quantum* **4**, 269 (2020).
55. Koczor, B. & Benjamin, S. C. Quantum natural gradient generalised to non-unitary circuits. Preprint at <https://arxiv.org/abs/1912.08660> (2019).
56. Sharma, K. et al. Reformulation of the no-free-lunch theorem for entangled data sets. *Phys. Rev. Lett.* **128**, 070501 (2022).
57. Abbas, A. The power of quantum neural networks. *Nat. Comput. Sci.* **1**, 403–409 (2021).
58. Rosenblatt, F. *The Perceptron, a Perceiving and Recognizing Automaton (Project PARA)* Report No. 85-460-1 (Cornell Aeronautical Laboratory, 1957).
59. Haykin, S. *Neural Networks: a Comprehensive Foundation* (Prentice Hall, 1994).
60. Rumelhart, D. E., Hinton, G. E. & Williams, R. J. Learning representations by back-propagating errors. *Nature* **323**, 533–536 (1986).
61. Hubregtsen, T. et al. Training quantum embedding kernels on near-term quantum computers. Preprint at <https://arxiv.org/abs/2105.02276> (2021).
62. Thanasilp, S., Wang, S., Nghiem, N. A., Coles, P. J. & Cerezo, M. Subtleties in the trainability of quantum machine learning models. Preprint at <https://arxiv.org/abs/2110.14753> (2021).
63. Huang, H.-Y. Quantum advantage in learning from experiments. *Science* **376**, 1182–1186 (2022).
64. Cotler, J., Huang, H.-Y. & McClean, J. R. Revisiting dequantization and quantum advantage in learning tasks. Preprint at <https://arxiv.org/abs/2112.00811> (2021).
65. Chen, S., Cotler, J., Huang, H.-Y. & Li, J. A hierarchy for replica quantum advantage. Preprint at <https://arxiv.org/abs/2111.05874> (2021).
66. Chen, S., Cotler, J., Huang, H.-Y. & Li, J. Exponential separations between learning with and without quantum memory. In *2021 IEEE 62nd Annual Symp. on Foundations of Computer Science (FOCS)* 574–585 (IEEE, 2022).
67. Perrier, E., Youssry, A. & Ferrie, C. QDataSet: quantum datasets for machine learning. Preprint at <https://arxiv.org/abs/2108.06661> (2021).
68. Schatzki, L., Arrasmith, A., Coles, P. J. & Cerezo, M. Entangled datasets for quantum machine learning. Preprint at <https://arxiv.org/abs/2109.03400> (2021).
69. Arrasmith, A., Holmes, Z., Cerezo, M. & Coles, P. J. Equivalence of quantum barren plateaus to cost concentration and narrow gorges. *Quantum Sci. Technol.* **7**, 045015 (2022).
70. Bittel, L. & Kliesch, M. Training variational quantum algorithms is NP-hard. *Phys. Rev. Lett.* **127**, 120502 (2021).
71. Bilkis, M., Cerezo, M., Verdon, G., Coles, P. J. & Cincio, L. A semi-agnostic ansatz with variable structure for quantum machine learning. Preprint at <https://arxiv.org/abs/2103.06712> (2021).
72. LaRose, R., Tikku, A., O’Neil-Judy, É., Cincio, L. & Coles, P. J. Variational quantum state diagonalization. *npj Quantum Inf.* **5**, 57 (2019).
73. Kiani, B. T., Lloyd, S. & Maity, R. Learning unitaries by gradient descent. Preprint <https://arxiv.org/abs/2001.11897> (2020).
74. Larocca, M., Ju, N., García-Martín, D., Coles, P. J. & Cerezo, M. Theory of overparametrization in quantum neural networks. Preprint at <https://arxiv.org/abs/2109.11676> (2021).
75. McClean, J. R., Boixo, S., Smelyanskiy, V. N., Babbush, R. & Neven, H. Barren plateaus in quantum neural network training landscapes. *Nat. Commun.* **9**, 4812 (2018).
76. Cerezo, M., Sone, A., Volkoff, T., Cincio, L. & Coles, P. J. Cost function dependent barren plateaus in shallow parametrized quantum circuits. *Nat. Commun.* **12**, 1791 (2021).
77. Cerezo, M. & Coles, P. J. Higher order derivatives of quantum neural networks with barren plateaus. *Quantum Sci. Technol.* **6**, 035006 (2021).
78. Arrasmith, A., Cerezo, M., Czarnik, P., Cincio, L. & Coles, P. J. Effect of barren plateaus on gradient-free optimization. *Quantum* **5**, 558 (2021).
79. Holmes, Z., Sharma, K., Cerezo, M. & Coles, P. J. Connecting ansatz expressibility to gradient magnitudes and barren plateaus. *PRX Quantum* **3**, 010313 (2022).
80. Pesah, A. Absence of barren plateaus in quantum convolutional neural networks. *Phys. Rev. X* **11**, 041011 (2021).
81. Volkoff, T. & Coles, P. J. Large gradients via correlation in random parameterized quantum circuits. *Quantum Sci. Technol.* **6**, 025008 (2021).
82. Sharma, K., Cerezo, M., Cincio, L. & Coles, P. J. Trainability of dissipative perceptron-based quantum neural networks. *Phys. Rev. Lett.* **128**, 180505 (2022).
83. Holmes, Zoë Barren plateaus preclude learning scramblers. *Phys. Rev. Lett.* **126**, 190501 (2021).
84. Marrero, C. O., Kieferova, M. & Wiebe, N. Entanglement induced barren plateaus. *PRX Quantum* **2**, 040316 (2020).
85. Uvarov, A. V. & Biamonte, J. D. On barren plateaus and cost function locality in variational quantum algorithms. *J. Phys. A* **54**, 245301 (2021).
86. Patti, T. L., Najafi, K., Gao, X. & Yelin, S. F. Entanglement devised barren plateau mitigation. *Phys. Rev. Res.* **3**, 033090 (2021).
87. Wang, S. Noise-induced barren plateaus in variational quantum algorithms. *Nat. Commun.* **12**, 6961 (2021).
88. Verdon, G. et al. Learning to learn with quantum neural networks via classical neural networks. Preprint at <https://arxiv.org/abs/1907.05415> (2019).
89. Verdon, G. et al. Quantum graph neural networks. Preprint at <https://arxiv.org/abs/1909.12264> (2019).
90. Bronstein, M. M., Bruna, J., Cohen, T. & Velicković, P. Geometric deep learning: grids, groups, graphs, geodesics, and gauges. Preprint at <https://arxiv.org/abs/2104.13478> (2021).
91. Larocca, M. et al. Group-invariant quantum machine learning. Preprint at <https://arxiv.org/abs/2205.02261> (2022).
92. Skolik, A., Cattelan, M., Yarkoni, S., Bäck, T. & Dunjko, V. Equivariant quantum circuits for learning on weighted graphs. Preprint at <https://arxiv.org/abs/2205.06109> (2022).
93. Meyer, J. J. et al. Exploiting symmetry in variational quantum machine learning. Preprint at <https://arxiv.org/abs/2205.06217> (2022).
94. Larocca, M. et al. Diagnosing barren plateaus with tools from quantum optimal control. Preprint at <https://arxiv.org/abs/2105.14377> (2021).
95. Wang, S. et al. Can error mitigation improve trainability of noisy variational quantum algorithms? Preprint at <https://arxiv.org/abs/2109.01051> (2021).
96. Deshpande, A. et al. Tight bounds on the convergence of noisy random circuits to the uniform distribution. Preprint at <https://arxiv.org/abs/2112.00716> (2021).
97. Hakkaku, S., Tashima, Y., Mitarai, K., Mizukami, W. & Fujii, K. Quantifying fermionic nonlinearity of quantum circuits. Preprint at <https://arxiv.org/abs/2111.14599> (2021).
98. Bultrini, D. et al. The battle of clean and dirty qubits in the era of partial error correction. Preprint at <https://arxiv.org/abs/2205.13454> (2022).
99. Temme, K., Bravyi, S. & Gambetta, J. M. Error mitigation for short-depth quantum circuits. *Phys. Rev. Lett.* **119**, 180509 (2017).
100. Czarnik, P., Arrasmith, A., Coles, P. J. & Cincio, L. Error mitigation with Clifford quantum-circuit data. *Quantum* **5**, 592 (2021).
101. Endo, S., Cai, Z., Benjamin, S. C. & Yuan, X. Hybrid quantum–classical algorithms and quantum error mitigation. *J. Phys. Soc. Jpn* **90**, 032001 (2021).
102. Sharma, K., Khatri, S., Cerezo, M. & Coles, P. J. Noise resilience of variational quantum compiling. *New J. Phys.* **22**, 043006 (2020).
103. Cincio, L., Rudinger, K., Sarovar, M. & Coles, P. J. Machine learning of noise-resilient quantum circuits. *PRX Quantum* **2**, 010324 (2021).
104. Ho, A., Verdon, G. & Mohseni, M. Quantum machine perception. US patent application 17,019,564 (2020).
105. Meyer, J. J., Borregaard, J. & Eisert, J. A variational toolbox for quantum multi-parameter estimation. *NPJ Quantum Inf.* **7**, 89 (2021).
106. Beckey, J. L., Cerezo, M., Sone, A. & Coles, P. J. Variational quantum algorithm for estimating the quantum Fisher information. *Phys. Rev. Res.* **4**, 013083 (2022).
107. Wang, J. Experimental quantum Hamiltonian learning. *Nat. Phys.* **13**, 551–555 (2017).
108. Layden, D. & Cappellaro, P. Spatial noise filtering through error correction for quantum sensing. *npj Quantum Inf.* **4**, 30 (2018).
109. Johnson, P. D., Romero, J., Olson, J., Cao, Y. & Aspuru-Guzik, A. QVECTOR: an algorithm for device-tailored quantum error correction. Preprint at <https://arxiv.org/abs/1711.02249> (2017).
110. Peruzzo, A. A variational eigenvalue solver on a photonic quantum processor. *Nat. Commun.* **5**, 4213 (2014).
111. McArdle, S. Variational ansatz-based quantum simulation of imaginary time evolution. *npj Quantum Inf.* **5**, 75 (2019).
112. Cirstoiu, C. Variational fast forwarding for quantum simulation beyond the coherence time. *npj Quantum Inf.* **6**, 82 (2020).
113. Huang, H.-Y., Kueng, R. & Preskill, J. Information-theoretic bounds on quantum advantage in machine learning. *Phys. Rev. Lett.* **126**, 190505 (2021).
114. Aharonov, D., Cotler, J. & Qi, X.-L. Quantum algorithmic measurement. *Nat. Commun.* **13**, 887 (2022).
115. Chia, N.-H. et al. Sampling-based sublinear low-rank matrix arithmetic framework for dequantizing quantum machine learning. In *Proc. 52nd Annual ACM SIGACT Symposium on Theory of Computing* 387–400 (Association for Computing Machinery, 2020).

116. Preskill, J. Quantum computing in the NISQ era and beyond. *Quantum* **2**, 79 (2018).
117. Huang, H.-Y., Kueng, R., Torlai, G., Albert, V. V. & Preskill, J. Provably efficient machine learning for quantum many-body problems. Preprint at <https://arxiv.org/abs/2106.12627> (2021).
118. Hohenberg, P. & Kohn, W. Inhomogeneous electron gas. *Phys. Rev.* **136**, B864–B871 (1964).
119. Kohn, W. Nobel lecture: Electronic structure of matter—wave functions and density functionals. *Rev. Mod. Phys.* **71**, 1253–1266 (1999).
120. Alcazar, J., Leyton-Ortega, V. & Perdomo-Ortiz, A. Classical versus quantum models in machine learning: insights from a finance application. *Mach. Learn. Sci. Technol.* **1**, 035003 (2020).
121. Bouland, A., van Dam, W., Joorati, H., Kerenidis, I. & Prakash, A. Prospects and challenges of quantum finance. Preprint at <https://arxiv.org/abs/2011.06492> (2020).
122. Manning C. & Schütze, H. *Foundations of Statistical Natural Language Processing* (MIT Press, 1999).
123. Russ, J. C. *The Image Processing Handbook* (CRC Press, 2006).
124. Grover, L. K. A fast quantum mechanical algorithm for database search. In *Proc. 28th Annual ACM Symposium on Theory of Computing* 212–219 (Association for Computing Machinery, 1996).
125. Bernstein, E. & Vazirani, U. Quantum complexity theory. *SIAM J. Comput.* **26**, 1411–1473 (1997).
126. Babbush, R. Focus beyond quadratic speedups for error-corrected quantum advantage. *PRX Quantum* **2**, 010103 (2021).
127. Georgescu, I. M., Ashhab, S. & Nori, F. Quantum simulation. *Rev. Mod. Phys.* **86**, 153–185 (2014).
128. Berry, D. W., Childs, A. M., Cleve, R., Kothari, R. & Somma, R. D. Simulating Hamiltonian dynamics with a truncated Taylor series. *Phys. Rev. Lett.* **114**, 090502 (2015).
129. Lvovsky, A. I., Sanders, B. C. & Tittel, W. Optical quantum memory. *Nat. Photon.* **3**, 706–714 (2009).
130. Sanchez-Lengeling, B. & Aspuru-Guzik, Alán Inverse molecular design using machine learning: generative models for matter engineering. *Science* **361**, 360–365 (2018).

## Acknowledgements

M.C. acknowledges support from the Los Alamos National Laboratory (LANL) LDRD program under project number 20210116DR. M.C. was also supported by the Center for Nonlinear Studies at LANL. L.C. and P.J.C. were supported by the US Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, under the Accelerated Research in Quantum Computing program. L.C. also acknowledges support from US Department of Energy, Office of Science, National Quantum Information Science Research Centers, Quantum Science Center. P.J.C. was also supported by the NNSA's Advanced Simulation and Computing Beyond Moore's Law Program at LANL. G.V. thanks F. Sbahi, A. J. Martinez and P. Velickovic for useful discussions. X, formerly known as Google[x], is part of the Alphabet family of companies, which includes Google, Verily, Waymo and others ([www.x.company](http://www.x.company)). H.-Y.H. is supported by a Google PhD fellowship.

## Author contributions

P.J.C. drafted the manuscript structure. The manuscript was written and revised by M.C., G.V., H.-Y.H., L.C. and P.J.C.

## Competing interests

The authors declare no competing interests.

## Additional information

**Correspondence** should be addressed to Patrick J. Coles.

**Peer review information** *Nature Computational Science* thanks Dan Browne and Mile Gu for their contribution to the peer review of this work. Primary Handling Editor: Jie Pan, in collaboration with the *Nature Computational Science* team.

**Reprints and permissions information** is available at [www.nature.com/reprints](http://www.nature.com/reprints).

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.

© Springer Nature America, Inc. 2022