



# Assessing the advantages and limitations of quantum neural networks in regression tasks

Gubio Gomes de Lima<sup>1</sup> · Tiago de S. Farias<sup>1</sup> · Alexandre C. Ricardo<sup>1</sup> · Celso Jorge Villa Boas<sup>1</sup>

Received: 4 September 2025 / Accepted: 19 February 2026  
© The Author(s) 2026

## Abstract

The development of quantum neural networks (QNNs) has attracted considerable attention due to their potential to surpass classical models in certain machine learning tasks. Nonetheless, it remains unclear under which conditions QNNs provide concrete benefits over classical neural networks (CNNs). This study addresses this question by performing both qualitative and quantitative analyses of classical and quantum models applied to regression problems, using two target functions with contrasting properties. Additionally, the work explores the methodological difficulties inherent in making fair comparisons between QNNs and CNNs. The findings reveal a distinct advantage of QNNs in a specific quantum machine learning context. In particular, QNNs excelled at approximating the sinusoidal function, achieving errors up to seven orders of magnitude lower than their classical counterparts. However, their performance was limited in other cases, emphasizing that QNNs are highly effective for certain tasks but not universally superior. These results reinforce the principles of the “No Free Lunch” theorem, highlighting that no single model outperforms all others across every problem domain.

**Keywords** Quantum neural network · Comparative study · Regression tasks

## 1 Highlights

- Quantum neural networks achieve errors up to  $10^7 \times$  smaller than classical neural networks.
- A fair comparison requires clear assumptions and attention to different perspectives.
- Results support the No Free Lunch theorem, showing no universal advantage for quantum machine learning.

## 2 Introduction

Classical artificial neural networks have become a cornerstone of modern machine learning and are applied across numerous scientific and engineering domains (Schalkoff 1997; Pedrycz and Chen 2019; Shinde and Shah 2018; Zhou et al. 2019; Khan and Yairi 2018; Bakator and

Radosav 2018; Hoang and Wiegatz 2023; Marugán et al. 2018). Their power stems from their proven ability as universal function approximators (Liang and Srikant 2016; Adcock and Dexter 2021; Elbrächter et al. 2021), which, when combined with sophisticated optimization techniques (Gudise and Venayagamoorthy 2003; Zhang and Qu 2021; Leonard and Kramer 1990), allows them to model complex, non-linear relationships directly from data (Jain et al. 2020; Gupta et al. 2021). Despite their widespread success, significant challenges remain. These include the “black box” nature of deep models, which hinders interpretability, the need for substantial labeled datasets, catastrophic forgetting, vanishing, and exploding gradient problems, and the escalating computational and energy costs of state-of-the-art architectures (Mishra et al. 2020; Alzubaidi et al. 2021). The latter presents a significant barrier for applications in resource-constrained environments and for researchers without access to large-scale computing infrastructure.

In this context, quantum neural networks have emerged as a promising alternative, attempting to leverage properties of quantum mechanics to overcome the limitations of classical models in certain tasks (Abbas et al. 2021; Kwak et al. 2021; Zhao and Wang 2021; Cerezo et al. 2022). Nevertheless, the exact quantum properties that may contribute

✉ Gubio Gomes de Lima  
gubiofisika@gmail.com

<sup>1</sup> Universidade Federal de São Carlos, Rodovia Washington Luís, km 235 - SP-310, São Carlos 13565-905, SP, Brazil

to these improvements, as well as the nature of the tasks in which such advantages manifest, remain open questions in the scientific community. Despite the interest in QNNs, their definition and practical benefits remain under active debate (Schuld et al. 2014; Altaisky et al. 2014; Qi et al. 2023). A particularly relevant approach involves the use of continuous-variable quantum computing as a model of execution of QNNs (Killoran et al. 2019). This model presents significant advantages compared to the qubit or qudit-based quantum computing paradigms, such as the ease of encoding floating-point numbers and the inherent inclusion of nonlinear operations within quantum circuits (Shi et al. 2020; Choe and Perkowski 2022; Bangar et al. 2023, 2025; Panichi et al. 2025; Markidis 2022).

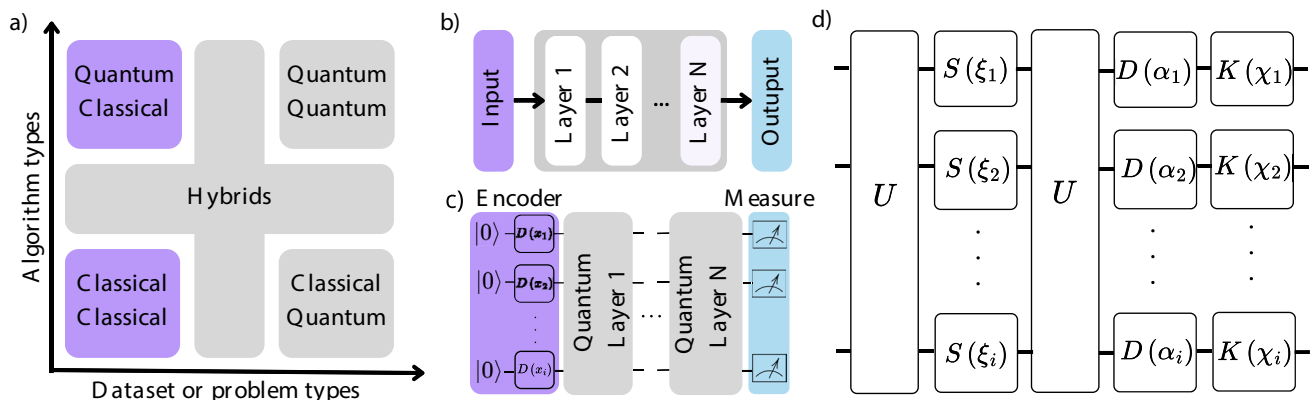
Several studies have focused on comparing classical and quantum algorithms (Wendlinger et al. 2024; West et al. 2023; Suzuki et al. 2024; Fellner et al. 2025; Kea et al. 2024; Hidalgo 2025; Nadim et al. 2025; Ghosh et al. 2024; Nadim et al. 2025; Genç 2024), with different conclusions and perspectives. Abbas et al. (2021) evaluated the expressibility and training efficiency of a QNN, employing information geometry to define the expressibility of quantum versus classical models, providing evidence that well-designed QNNs can surpass classical neural networks in terms of effective dimension and training speed, with validation on real quantum hardware. In the work of Qian et al. (2024) they conducted numerical experiments and observed that, although QNNs can perform well on specific datasets, they generally fail to outperform classical models across broader tasks, indicating that the practical benefits of QNN are still under investigation and not yet universally established. Among these studies, only a limited number investigate CNNs and QNNs (Zaman et al. 2025; Tasnim et al. 2025; Gupta et al. 2022). Within this subset, the majority of applications focus on classification problems, whereas only a few address regression-like tasks (Suzuki et al. 2024; Fellner et

al. 2025; Kea et al. 2024). Regression constitutes a central problem in machine learning, as it involves estimating feature mappings to real-valued outcomes, thereby enabling quantitative predictions that directly inform decision-making. Numerous real-world applications rely on regression analysis, including forecasting (Masini et al. 2023), control (Olivier 2021), energy systems (Ko and Lee 2013), finance (Gu et al. 2020), and scientific research (Rupp et al. 2012)

Establishing a fair comparison between QNNs and their classical counterparts is notoriously difficult, not only due to fundamental differences in hardware architectures but also because of the absence of universally accepted criteria for such an analysis (Zaman et al. 2025; Wendlinger et al. 2024; West et al. 2023; Tasnim et al. 2025). This challenge extends beyond the fundamental differences, as even within classical machine learning, comparing models with disparate architectures and hyperparameter spaces lacks a standardized methodology.

The domain of quantum machine learning (QML) encompasses a wide landscape of potential algorithm and data-type combinations, as illustrated in Fig. 1a. This work situates itself within the hybrid quantum-classical regime, where classical problems are tackled using both quantum and classical algorithms, but the data is classical (highlighted in purple in the Fig. 1a). Beyond this classification, emerging gray areas can be identified in which algorithms and data cannot be strictly categorized as purely classical or purely quantum. An example of such hybrid approaches is the use of classical convolutional neural networks as a preprocessing step to reduce data dimensionality before encoding it into a quantum circuit (Killoran et al. 2019). In a broader sense, the hybrid data/problem regime encompasses scenarios in which a problem involves both classical and quantum information.

To establish a well-defined comparison and benchmark the performance of CNNs and QNNs, we restrict



**Fig. 1** (a) Algorithm versus problems addressed by quantum and classical ML; cases highlighted in purple represent those explored in this work. (b) Schematic representation of the neural network model. (c) Representation of the variational quantum algorithm (QNN), with each

line representing a quantum mode. (d) Detailed representation of the “Quantum layer” described in (c), illustrating the sequence of quantum logic gates employed

our analysis to supervised regression tasks of a sinusoidal function and a Heaviside step function. The selection of these specific functions is motivated by their contrasting mathematical properties, which serve to probe the limits of function approximation in both architectures. The sinusoidal function represents a smooth, continuous, and periodic target, serving as a baseline to evaluate how well the models capture regular, well-behaved patterns. In contrast, the Heaviside step function introduces a sharp discontinuity, posing a significant challenge for gradient-based optimization and testing the models' ability to resolve non-periodic targets. Furthermore, one-dimensional problems naturally align with the one-input, one-output architecture of the continuous-variable QNN, which makes it suitable for this investigation.

Our analysis is conducted using two strategies: first, varying the number of layers, and second, matching the number of trainable parameters across models. We evaluate prediction accuracy using the Mean Squared Error (MSE) as a quantitative metric. The primary objective is to identify specific scenarios in which QNNs demonstrate a concrete advantage over CNNs, while also discussing the inherent challenges and limitations of such a comparative analysis.

The article is structured as follows. In Section 2, we present the core component of the classical neural network algorithm: the neuron, or perceptron. Section 3 introduces the continuous-variable quantum neuron used in the comparison, along with the encoding process and measurement scheme. In Section 4, we discuss the challenges and complexities involved in developing a fair and meaningful comparison between the algorithms, given their intrinsic differences. Section 5 presents and analyzes the main results of the comparative study. Finally, in Section 6, we summarize the conclusions and outline future research directions in this area.

### 3 Classical neural networks

A neural network is a computational model composed of interconnected units known as artificial neurons, here named as classical neurons, organized in different layers. The mathematical function defines the  $k$ -th classical neuron as

$$G_k^{[l]} \left( \sum_i w_{i,k}^{[l]} \cdot x_i^{[l-1]} + b_k^{[l]} \right), \quad (1)$$

which depends on the trainable parameters  $w_{i,k}^{[l]}$  and  $b_k^{[l]}$ , known as weights and biases, for the  $k$ -th neuron in the  $l$ -th layer. These parameters determine the response of the neuron to the input data  $x_i^{[l-1]}$  from the preceding layer. The function  $G_k^{[l]}$  represents the non-linear activation function.

A neural network is formed by organizing these neurons into layers, where their collective interactions process information.

A common neural network architecture, particularly in supervised learning, is the fully connected feedforward neural network (Sazlı 2006). In this architecture, each layer processes the information and passes it to the next layer, as illustrated in Fig. 1b. The performance and versatility of this model stem not only from its structural design but also from the learning algorithms that allow its parameters to be efficiently updated. Training is typically achieved by minimizing a loss function using gradient-based optimization algorithms (Soydancer 2020; Fatima 2020). The required gradients are efficiently calculated via the backpropagation algorithm (Robert 1992), which is implemented in modern software frameworks using automatic differentiation techniques (Ketkar and Moolayil 2021; Baydin et al. 2018).

According to the universal approximation theorem (Augustine 2024), a neural network can approximate any continuous function, provided it has at least one hidden layer and a sufficient number of neurons. However, the theorem is non-constructive, since it does not specify the required number of neurons, layers, or the optimal choice of activation function. These architectural choices thus become critical hyperparameters. Consequently, designing an effective neural network for a specific problem often requires extensive hyperparameter tuning or an architecture search process to find a suitable configuration.

## 4 Quantum neural networks

The exact definition of a quantum neural network is an ongoing debate in the scientific community, with proposals ranging from models that integrate classical and quantum elements to purely quantum implementations, reflecting the emerging nature of this field of research (Kwak et al. 2021; Ezhov 2021; Zaman et al. 2025; Abel et al. 2022; Zhou et al. 2023). A common framework, and the one adopted here, is the variational quantum algorithm, which uses parameterized quantum circuits as trainable models analogous to classical networks.

In this work, we employ a continuous-variable (CV) QNN based on the variational circuit model introduced in Ref. (Killoran et al. 2019). This paradigm is particularly well-suited for our comparative analysis in QML due to two key advantages over qubit-based approaches: the natural encoding of real-valued data and the ability to implement non-linear operations directly within the quantum circuit. Our implementation, shown schematically in Fig. 1c and d,<sup>1</sup>

<sup>1</sup> Note that, for the single-mode case, the operator  $\hat{U}$  in Fig. 1d is given by a rotation operator.

operates in three stages: data encoding, parameterized processing, and measurement.

1. **Data encoding:** To input a classical value  $x$  into the network, we apply a displacement operator  $\hat{D}(x)$  to the vacuum state  $|0\rangle$ , preparing the initial state  $|x\rangle = \hat{D}(x)|0\rangle$ . This method directly maps a real number to the phase space of a quantum mode, avoiding the significant overhead required to represent continuous data in qubit-based (Ranga et al. 2024; Pande 2024; Rath and Date 2024).
2. **Parameterized Quantum Layers:** Following encoding, the state is processed by a Quantum Layer. For a single-mode system, this layer is composed of a sequence of quantum gates that we define as a quantum neuron:

$$\text{Quantum Neuron} : \hat{K}(\chi)\hat{D}(\alpha)\hat{R}(\theta_2)\hat{S}(\xi)\hat{R}(\theta_1). \quad (2)$$

This sequence consists of the Gaussian operation as displacement  $\hat{D}(\alpha)$ , rotation  $\hat{R}(\theta)$  and squeezing  $\hat{S}(\xi)$  (Weedbrook et al. 2012). A critical component for expressive power is a nonlinear transformation. In this case, we use a validated approximation of the Kerr gate  $\hat{K}(\chi)$  tailored for trapped-ion systems (Ricardo et al. 2024). A deep QNN is constructed by repeating this entire quantum layer  $L$  times, as depicted in Fig. 1c. Each layer adds five trainable parameters  $(\chi, \alpha, \theta_2, \xi, \theta_1)$ , resulting in a total of  $5L$  parameters for the model.

3. **Measurement:** To extract a classical output from the circuit, a measurement is performed. For the tasks in this study, we measure the expectation value of the quadrature operator  $\hat{X} = \frac{1}{\sqrt{2}}(\hat{a}^\dagger + \hat{a})$ , where  $\hat{a}$  and  $\hat{a}^\dagger$  denote the annihilation and creation operators. This approach makes it easier to obtain a real-valued floating-point number in the output.

Our analysis focuses on single-variable problems, restricting the QNN implementation to a single quantum mode. This decision is pragmatic, as simulating multi-mode CV systems on classical hardware is computationally demanding due to the exponential growth of the Hilbert space. In particular, the space complexity scales as  $N^D$ , where  $D$  is the number of modes and  $N$  is the size of the basis. This constitutes a major limitation of classical simulations of CV quantum algorithms, since  $N$  must be sufficiently large to accurately represent the probability distribution of the quantum states. Qubit-based models exhibit similar exponential scaling, but with  $N = 2$ , making classical simulation of systems with multiple qubits comparatively more feasible. Then, for the numerical simulation, quantum states are represented in the Fock basis, which consists of photon number states  $|n\rangle$ . Any

pure state  $|\psi\rangle$  is expanded as an arbitrary superposition of the Fock states as  $|\psi\rangle = \sum_n^{N=\infty} c_n |n\rangle$ . Because the full basis is infinite-dimensional, we must truncate it at a finite cutoff dimension for numerical calculations. This is a valid approximation provided that the probability amplitudes  $c_n$  for  $n$  near the cutoff are negligible. In this work, we set the cutoff dimension to  $N = 30$ , which ensures simulation fidelity while maintaining computational tractability.

While most quantum neural networks are formulated within a qubit-based, gate-model framework, where quantum advantage is often associated with multi-qubit entanglement, our approach adopts a CV architecture operating in a single-mode setting. In this regime, entanglement is not the primary computational resource. Instead, expressivity and potential quantum advantage arise from the infinite-dimensional Hilbert space of CV systems and, crucially, from the inclusion of non-Gaussian operations, which introduce effective nonlinear transformations inaccessible to purely Gaussian or qubit Clifford circuits.

## 5 Challenges in comparison

Establishing a fair comparison between classical and quantum neural networks is intrinsically complex. The challenge stems not only from their disparate computational architectures but also from the absence of universally accepted benchmarking criteria. The “No Free Lunch” theorem (Wolpert 2021) underscores this, asserting that no single algorithm is optimal across all possible problems. An algorithm superior in one problem class may be suboptimal in another (Adam et al. 2019; Gómez and Rojas 2016). Therefore, any meaningful comparison requires a clearly defined context, including a specific problem scope, suitable evaluation metrics, and a careful accounting of structural differences between the models.

To create this well-defined context, we first narrow our focus to supervised regression tasks. This choice is motivated by several factors. First, supervised learning is a foundational paradigm in both classical and quantum machine learning, providing common ground for comparison. Second, for single-variable regression, estimating an unknown function  $f(x)$  from input-output pairs  $\{(x^{(i)}, y^{(i)})\}_{i=1}^n$ , aligns directly with the one-input, one-output architecture of the continuous-variable QNN model used in this work. The goal is to train a model  $\hat{f}$  that accurately predicts an output  $\hat{y} = \hat{f}(x)$  for a new input  $x$ .

Next, we must select an evaluation metric. Performance can be measured from multiple perspectives, including accuracy, training time, computational cost, or robustness. Since our primary goal is to evaluate the expressive power and solution quality of the models, we adopt the MSE as

both the training loss function and the final performance metric. The MSE is defined as

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^N (\hat{y}_i - y_i)^2, \quad (3)$$

where  $N$  is the number of data points,  $y_i$  is the true value, and  $\hat{y}_i$  is the prediction of the model. Finally, to navigate the structural disparities between the models, we devise two complementary comparison strategies. Each strategy is designed to isolate different aspects of performance and provide a nuanced view of the capabilities of each model.

The first approach involves using one classical neuron and one quantum neuron per layer while keeping as many hyperparameters as possible fixed, varying only the number of layers. This design ensures that the network architectures remain as comparable as possible. However, this approach has inherent limitations. In particular, it raises questions about the validity of the comparison since classical networks offer a variety of activation functions, whereas QNNs rely on a specific nonlinear function, controlled by the parameter  $\chi$ . Moreover, each QNN layer includes five trainable parameters, whereas the simplified classical neuron has only two. This gives the QNN a greater representational capacity for the same number of layers. This structure artificially limits the classical model, which typically leverages wider layers (more neurons per layer) to achieve universal approximation. This discrepancy raises concerns about the validity of comparing models that differ in their parameter counts, given that the number of parameters influences the optimizer and, consequently, the algorithm's accuracy and precision.

Our second strategy aims to provide a fairer comparison of expressive power by ensuring both models have the same number of trainable parameters. To achieve this, we fix the QNN architecture (and thus its parameter count) and adjust the width and depth of the classical network to match that count. Although this equalizes model capacity, it reveals a fundamental architectural trade-off: in classical networks, parameters can be flexibly distributed between width and depth, whereas in our QNN the parameterization is strictly determined by its depth (number of layers). This approach allows for a more direct evaluation of which model makes more effective use of its parametric budget.

## 6 Results and discussion

In this section, we present a comparative analysis of the classical and quantum neural networks on supervised regression tasks. To evaluate the models' performance, we selected two target functions with distinct properties over the interval

$[-1, 1]$ . The first is the sine function  $y = \sin(\pi x)$ , a common choice for regression tasks due to its continuous and periodic nature, which appears in various physical and natural phenomena. The second function is the Heaviside function, given by,

$$\text{Heaviside}(x) = \begin{cases} 0 & \text{if } x < 0, \\ 1 & \text{if } x \geq 0. \end{cases}$$

Unlike the sine function, the Heaviside function exhibits a discontinuity at  $x = 0$  and is therefore non-differentiable at this point. This property imposes an additional challenge on the regression model, as it requires the ability to handle abrupt transitions in the target function values. Both functions are commonly used in testing numerical methods of approximating functions (Stulp and Sigaud 2015; Gajny et al. 2017; Fernández-Delgado et al. 2019).

### 6.1 Implementation details

The data used for training and testing were generated through a uniform discretization of the domain of the selected functions. A total of  $N_{\text{train}} = 20$  points was used for the training set, and, to ensure a robust evaluation of the models,  $N_{\text{test}} = 200$  points were used for the test set.

For the optimization strategy, we employed the Adam algorithm (Kingma and Ba 2014) with a learning rate of 0.01 over  $10^4$  epochs in all cases. To account for the stochasticity of parameter initialization, all reported results are averaged over 100 independent training runs with different random seeds. The choice of learning rate and number of epochs was guided by prior experience with similar models, as well as preliminary tests conducted on the current setup. These exploratory runs indicated that a learning rate of 0.01 provides a good balance between convergence speed and training stability across both models considered. Furthermore, the number of training epochs was constrained by the computational cost of simulating the quantum model.

The simulation time for the entire quantum system for one of the functions took approximately one week, even when parallelized across a machine with two AMD EPYC™ 7452 processors, 32 CPU cores (128 threads), and 256 GB of RAM (3200 MHz). This limitation imposed a practical upper bound on the number of epochs feasible within our computational budget.

### 6.2 Comparison in terms of the number of layers

We first analyze the models under our structural strategy, where performance is evaluated as a function of network depth with a single neuron per layer. This test is designed to probe how each architecture leverages increasing depth

under a highly constrained, narrow structure. As shown in Fig. 2a, the quantum model (black dotted line) exhibits a decreasing trend in error as the number of layers increases. Notably, the quantum algorithm achieved the best performance, reaching an MSE on the order of  $10^{-8}$ . This behavior indicates that the quantum architecture effectively captures the smooth structure of the sine function, reducing the error by several orders of magnitude while increasing network depth.

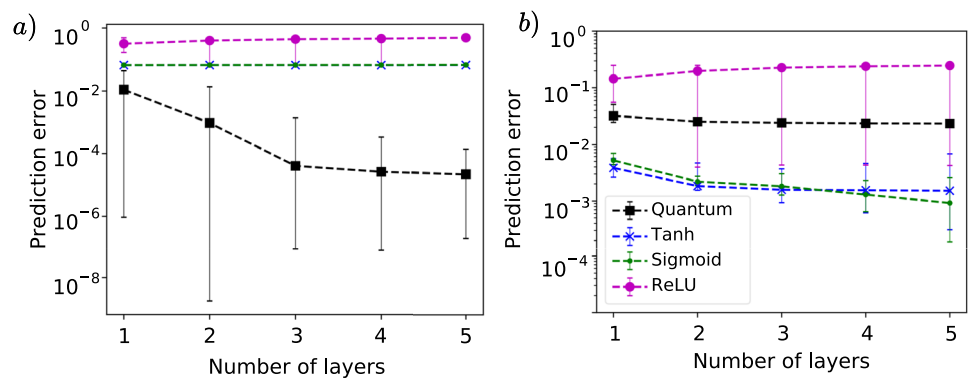
In contrast, all classical models stagnate with high error, regardless of depth. This poor performance is an expected consequence of the single neuron per layer constraint. A narrow classical network has severely limited representational capacity, as it cannot form the complex combinations of features that wider layers enable. The QNN, whose parameters are inherently tied to its depth via a sequence of unitary operations, does not suffer from this specific limitation in the same way, highlighting a fundamental difference in how the two architectures use their parameters. Our hypothesis for the observed quantum advantage is attributed to the higher expressiveness associated with the number of parameters. Specifically, in the case of CNNs, the parameter count scales as  $2L$ , whereas in QNNs it scales as  $5L$ .

Figure 2b presents the results for approximating the Heaviside function. In this case, a markedly different behavior is observed compared to the previous result. The

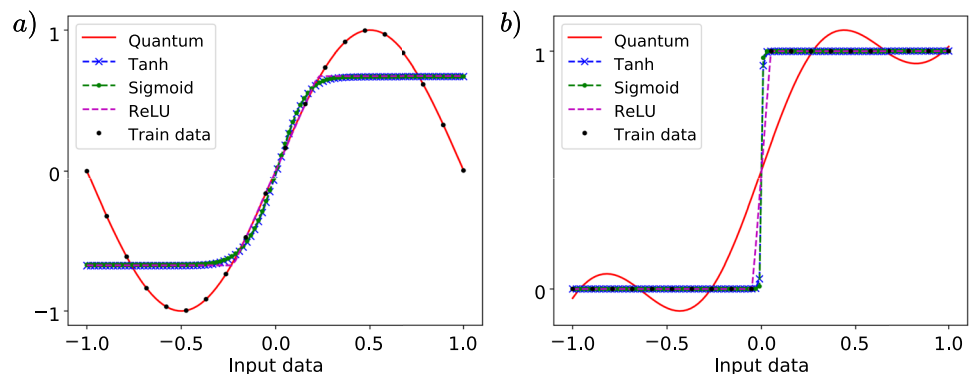
quantum model maintains a relatively high and nearly constant error as the number of layers increases, suggesting a fundamental limitation in its ability to represent abrupt discontinuities. In contrast, classical models with Tanh and Sigmoid activations show a progressive reduction in error as the number of layers increases. Their inherent sigmoidal shape is well-suited to approximating a step-like transition, and adding depth allows for a steeper and more accurate fit. The ReLU-based model, however, continues to exhibit the worst performance, as its piecewise-linear nature is ill-suited for this task.

Figure 3 provides a qualitative comparison of the predictive capabilities of the best results of classical and quantum neural networks in modeling the a) sine and b) Heaviside functions. For the sine function, the quantum model (red curve) exhibits good fitting and generalization in the training region, and the classical networks with tanh, sigmoid, and ReLU activation functions fit the data well in the central region, where the input data ranges over  $[-0.25, 0.25]$ , but struggle at the extremes. The three classical cases have good results for the Heaviside function, although ReLU, on average, presents poor results compared to the quantum model, as we see in Fig. 3b. The failure of the QNN is also clear, as its prediction is characterized by the smooth, oscillatory behavior intrinsic to its variational form, which prevents it from capturing the sharp transition.

**Fig. 2** Prediction error with MSE in logarithmic scale of QNN and CNN, as a function of the number of layers. The curves represent different activation functions used in CNN: Tanh (blue), Sigmoid (green), and ReLU (magenta). The black squares indicate results obtained by the quantum model. Error bars represent statistical variation across 100 different initializations per layer. Subplot **a**) presents the results for the  $\sin(\pi x)$  and **b**) the Heaviside(x)



**Fig. 3** Best results obtained after training the algorithms applied to the (a) sine and (b) Heaviside functions. The curves represent the quantum neural network (red) and classical neural networks with activation functions: Tanh (blue), Sigmoid (green), and ReLU (magenta); each curve was generated with 200 points. The black points indicate the training data with 20 points



### 6.3 Comparison in terms of the number of parameters

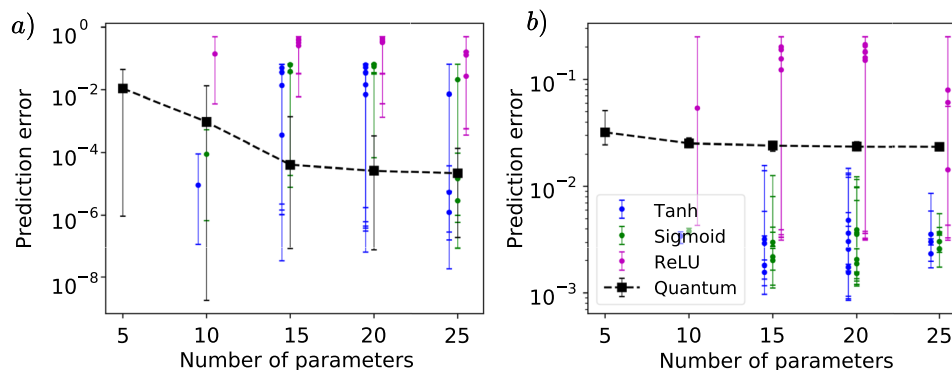
In the previous results, we identified a case in which the quantum algorithm outperformed the classical one. However, this comparison was limited to CNNs, as the number of parameters in the two models differed. To address this, our second strategy offers an alternative comparison. In this approach, we evaluate the models based on their parameter count. Here, the goal is to assess which architecture makes more efficient use of a given number of trainable parameters. For each parameter count defined by the QNN, we explored multiple CNN architectures with different combinations of layers and neurons that yielded the same total parameter count. The full details of these configurations are provided in the Appendix A.

Figure 4 presents the comparison of the prediction errors obtained by each algorithm as a function of the number of parameters. The plot in Fig. 4a displays the MSE for the function  $\sin(\pi x)$  on a logarithmic scale, allowing for a clear visualization of the performance differences among the tested models. The CNN exhibits relatively high errors, with the ReLU function performing particularly poorly, and the other two activation functions produce results with high variance. On average, it shows worse results. Only in specific situations does the CNN obtain good results. In contrast, the quantum model (represented by black squares) demonstrates a steeper decline in error from  $10^{-2}$  to  $10^{-5}$ , reaching average values three orders of magnitude smaller than the average of some classical activation functions. The results of the quantum model used in this second comparison are the same as those in the previous subsection, so we will have the same result. And as previously mentioned, they were the reference for defining the number of parameters in the comparison. Notably, the quantum model with only five

parameters, shown in Fig. 4a, already achieves results that classical models fail to reach. These findings suggest that, on average, the quantum neural network with continuous variables is capable of capturing the function's structure more effectively, even with a limited number of parameters. This indicates a potential advantage in learning smooth and continuous functions.

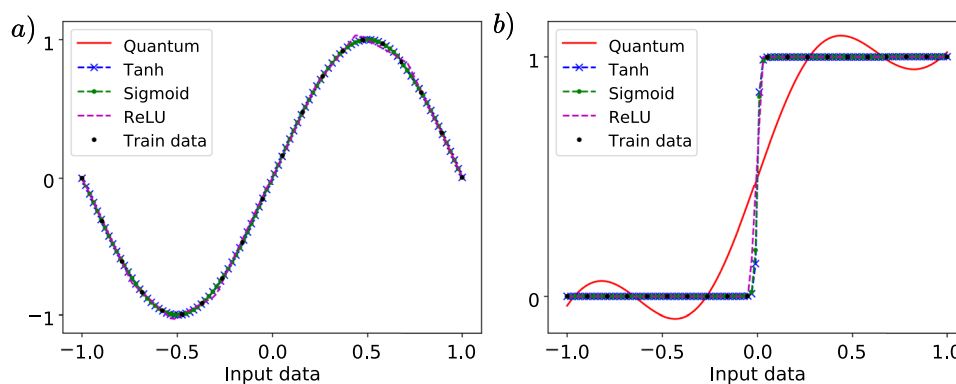
The second subplot Fig. 4b presents results for the Heaviside function. Unlike the previous case, CNN exhibits its highly activation-dependent behavior. Classical neural networks with tanh and sigmoid activation functions achieve lower and consistent errors, whereas ReLU-based models show a much greater dispersion and significantly higher errors, highlighting difficulties in learning the discontinuity. Meanwhile, the quantum model maintains a relatively constant error without the decreasing trend observed for the sine function. This suggests that, for discontinuous functions, the quantum model may not offer the same advantage observed for smooth functions, indicating that its applicability may strongly depend on the characteristics of the target function.

Figure 5 provides a qualitative view of the best-performing models under this parameter-matched comparison. The most striking result is seen in Fig. 5a. Unlike the failed attempt in the first strategy (Fig. 3a), the classical networks are now able to perfectly fit the sine function. This starkly illustrates the central theme of our comparative study: the conclusions drawn about the relative strengths of QNNs and CNNs are fundamentally dependent on the chosen basis for comparison. A comparison by structural analogues suggests classical inferiority on the sine task, whereas a comparison by parameter count demonstrates classical competence. This highlights the critical importance of defining a fair and well-motivated methodology when benchmarking quantum against classical machine learning models.



**Fig. 4** Prediction error results for each algorithm as a function of the number of parameters. The y-axis represents the error with MSE on a logarithmic scale, while the x-axis indicates the number of parameters with different combinations of neurons and layers for each point. The colors represent different activation functions used in CNN: Tanh

(blue), Sigmoid (green), and ReLU (magenta). The black squares indicate results obtained by the quantum model. Error bars represent statistical variation across 100 different initializations per layer. Subplot a) presents the results for the  $\sin(\pi x)$  and b) the Heaviside(x) functions



**Fig. 5** Best results obtained after training the QNN and CNN in the comparison based on the number of parameters. Panel (a) presents the fitting results for the sine function, while panel (b) corresponds to the Heaviside function. The curves represent the quantum neural

network (red) and classical neural networks with activation functions Tanh (blue), Sigmoid (green), and ReLU (magenta); each curve was generated with 200 points. The black points indicate the training data with 20 points

## 6.4 Discussions

The comparison reveals that the performance of the quantum model is highly sensitive to the nature of the function being learned. While for smooth functions, such as the sine function, it exhibits a clear advantage in terms of prediction error, this advantage is not evident for discontinuous functions, such as the Heaviside function. This suggests that, although quantum computing can offer significant improvements in certain scenarios, its effectiveness may depend on the specific problem being addressed, as in the classic case.

The advantage observed in the quantum model for smooth functions in this study may be attributed to the model's ability to effectively represent such functions within its state space. In contrast, the performance with the Heaviside function indicates a limitation in capturing functions with discontinuities. This reinforces the idea that the choice of a learning algorithm should consider the characteristics of the problem and that significant gains in specific scenarios do not imply the algorithm's superiority in a general sense.

Looking at another aspect outside of QNN algorithms, we can relate it to the analysis of the work by Schuld et al. (2021), which characterizes variational quantum models as partial Fourier series where the data-encoding strategy fundamentally determines the accessible frequency spectrum. In alignment with their analysis, the superior performance of the QNN on the sine function can be attributed to the model's inherent ability to represent low-frequency periodic functions, which can be effectively captured even with limited encoding repetitions (in our case, we only have one). Conversely, the bad performance of the Heaviside function underscores the limitations of standard encoding methods in representing the high-frequency components required for non-periodic functions with sharp discontinuities.

Beyond the performance, our study underscores that the methodology of comparison dictates the outcome. Compared with structural analogues, the constrained architecture of the classical network led to its failure on the sine task, creating the illusion of quantum advantage. However, when comparing by parameter count, the classical network, given sufficient flexibility, proved perfectly capable. This demonstrates that claims of quantum advantage can be misleading if not contextualized within a carefully justified and transparent benchmarking framework. Simply having more parameters per layer, as our QNN did in the first strategy, can easily be mistaken for a more fundamental algorithmic superiority.

This work highlights that the path to practical quantum advantage in machine learning will likely not involve a single QNN algorithm that outperforms all classical models. Instead, progress will come from identifying specific problem domains whose structure aligns with the inductive bias of a particular quantum algorithm. Furthermore, the selection of hyperparameters, both for the model architecture and the optimizer, remains as crucial in QML as it is in classical ML (A Ilembayo et al. 2024; Anand et al. 2020). The need for expert-driven tuning and architecture search is not eliminated by the quantum paradigm but rather adapted to a new set of operations and constraints.

## 7 Conclusion

This study presented a systematic comparison between classical and quantum neural networks in regression tasks, highlighting the conditions under which each model excels. We find a quantum advantage in a specific quantum machine learning scenario. However, the results demonstrate that the performance of QNNs is highly dependent on the nature of the target function: while they exhibited superiority in approximating the sine function, achieving a mean square error of the order of

$10^{-8}$ , their performance was limited in modeling the Heaviside function, which required capturing an abrupt discontinuity. In contrast, CNNs with tanh and sigmoid activation functions showed greater adaptability to discontinuities, although they were less efficient in approximating smooth functions.

The analysis also revealed methodological challenges in comparing these models, such as disparities in the number of parameters and the architectural rigidity of QNNs. Additionally, computational limitations in simulating QNNs on classical hardware restricted the exploration of more complex scenarios (multidimensional regression problems). Therefore, until suitable quantum hardware for this algorithm becomes available, its full potential remains unexplored. Consequently, determining whether quantum or classical neural networks perform better remains an open and challenging problem, raising important questions about the criteria for fair algorithmic comparison. In this work, we presented a way to proceed in this context. This highlights the importance of studies where practical comparative analyses are conducted to enrich the quantum machine learning literature in a manner analogous to what has already been established for classical machine learning algorithms and reinforces the importance of well-defined criteria for adopting QNNs.

With respect to trainability, the barren plateau phenomenon (McClellan et al. 2018; Zhang and Zhuang 2025), widely discussed in the context of deep, highly entangling qubit-based circuits, does not play a significant role in the present study. The single-mode nature of the CV architecture and the limited circuit depth employed in our simulations mitigate the emergence of such effects. Nevertheless, barren plateaus remain a relevant theoretical consideration for future extensions involving deeper circuits or multiple quantum modes, where scaling behavior becomes a central concern.

Looking forward, promising directions for future research include extending QNNs to higher-dimensional settings with two or more quantum modes, as well as systematically investigating alternative encoding strategies to assess their suitability for non-periodic and more complex target functions. The development of hybrid classical-quantum architectures also represents a compelling avenue, with the potential to combine the expressive strengths of neural networks with the structural properties of quantum models. Together, these directions may further clarify the role of QNNs within the broader landscape of machine learning methods.

## Appendix A: Table

Table 1 below presents all configurations used in the classical neural network during the comparison by the number of parameters. Each combination represents the distribution

of neurons per hidden layer, where the values in parentheses indicate the number of neurons in each layer. The input and output layers were omitted, as they always consist of a single input and a single output.

**Table 1** Neural network configurations per hidden layer in the classical neural network

Number of Parameters	Combinations
10	(3)
15	(1,4) (4,1) (1,2,2) (2,2,1)
20	(1,1,5) (1,5,1) (2,1,4) (3,1,3) (4,1,2) (5,1,1)
25	(8) (2,5) (5,2)

**Acknowledgements** This work was supported by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) - Finance Code 001 and São Paulo Research Foundation (FAPESP) grants No. 2022/00209-6, 2023/14831-3, and 2023/15739-3. C.J.V.-B. is also grateful for the support from the National Council for Scientific and Technological Development (CNPq) Grant No. 311612/2021-0. This work is also part of the CNPq Grants No. 140467/2022-0 and No. 140468/2022-6.

**Author Contributions** G.G. de Lima – Writing(Original Draft, Review & Editing), Conception of the research, Methodology, Collection of data, Data analysis/interpretation, Investigation. T.S. Farias – Formal analysis and methodology. A. C. Ricardo – Visualization and Collection of data, C.J.V. Boas – Supervision and Conception of the research. All authors reviewed the manuscript.

**Funding** The Article Processing Charge (APC) for the publication of this research was funded by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) (ROR identifier: 00x0ma614).

**Data Availability** All data supporting the findings of this study are available on GitHub: <https://github.com/GubioGL/Quantum-Neural-Networks-in-Regression-Tasks>.

## Declarations

**Competing interests** The authors declare no competing interests.

**Conflicts of interest** The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: GUBIO GOMES DE LIMA reports financial support was provided by UFScar. Reports a relationship with that includes: Has patent pending to. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted

use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

- A Illembayo J et al (2024) Hyperparameter Tuning in Machine Learning: A Comprehensive Review. *J Eng Res Reports* 26(6):388–395. <https://doi.org/10.9734/jerr/2024/v26i61188>
- Abbas A et al (2021) The power of quantum neural networks. *Nat Comput Sci* 1(6):403–409. <https://doi.org/10.1038/s43588-021-0084-1>
- Abel S et al (2022) Completely Quantum Neural Networks. *Phys Rev A* 106(2):022601. <https://doi.org/10.1103/PhysRevA.106.022601>
- Jain A et al (2020) Overview and importance of data quality for machine learning tasks, In: Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery data mining, pp 3561–3562. <https://doi.org/10.1145/3394486.3406477>
- Adcock B, Dexter N (2021) The gap between theory and practice in function approximation with deep neural networks. *SIAM J Math Data Sci* 3(2):624–655. <https://doi.org/10.1137/20M131309X>
- Baydin AG et al (2018) Automatic differentiation in machine learning: a survey. *J Mach Learn Res* 18(153):1–43
- Ezhov AA (2021) On Quantum Neural Networks. [arXiv:2104.07106](https://arxiv.org/abs/2104.07106)
- Altaiisky MV et al (2014) Quantum neural networks: Current status and prospects for development. *Phys Part Nucl* 45:1013–1032. <https://doi.org/10.1134/S1063779614060033>
- Alzubaidi L et al (2021) Review of deep learning: concepts. *CNN Arch Challenges Appl Future Directions J Big Data* 8(1):53. <https://doi.org/10.1186/s40537-021-00444-8>
- Ghosh A et al (2024) Quantum vs. Classical: A rigorous comparative study on neural networks for advanced satellite image classification. In: 2024 International conference on trends in quantum computing and emerging business technologies, pp 1–5. <https://doi.org/10.1109/TQCEBT59414.2024.10545133>
- Ricardo AC et al (2024) Quantum computation over the vibrational modes of a single trapped ion. [arXiv:2412.15025](https://arxiv.org/abs/2412.15025)
- Bakator M, Radosav D (2018) Deep learning and medical diagnosis: a review of literature. *Multimodal Technol Inter* 2(3). <https://doi.org/10.3390/mti2030047>
- Bangar S et al (2023) Experimentally realizable continuous-variable quantum neural networks. *Phys Rev A* 108(4):042414. <https://doi.org/10.1103/PhysRevA.108.042414>
- Bangar S et al (2025) Continuous-variable quantum Boltzmann machine. *Quantum Mach Intell* 7(1):1–15. <https://doi.org/10.1007/s42484-025-00259-3>
- Zhang B, Zhuang Q (2025) Quantum. *Sci Technol* 10:015009. <https://doi.org/10.1088/2058-9565/ad80bf>
- Cerezo M et al (2022) Challenges and opportunities in quantum machine learning. *Nat Comput Sci* 2(9):567–576. <https://doi.org/10.1038/s43588-022-00311-3>
- Ko C-N, Lee C-M (2013) Short-term load forecasting using SVR (support vector regression)-based radial basis function neural network with dual extended Kalman filter. *Energy* 49:413–422. <https://doi.org/10.1016/j.energy.2012.11.015>
- Gómez D, Rojas A (2016) An empirical overview of the no free lunch theorem and its effect on real-world machine learning classification. *Neural Comput* 28(1):216–228. [https://doi.org/10.1162/NECO\\_a\\_00793](https://doi.org/10.1162/NECO_a_00793)
- Wolpert DH (2021) What is important about the no free lunch theorems? In: Black box optimization, machine learning, and no-free lunch theorems, Springer International Publishing, pp 373–388. [https://doi.org/10.1007/978-3-030-66515-9\\_13](https://doi.org/10.1007/978-3-030-66515-9_13)
- Elbrächter D et al (2021) Deep neural network approximation theory. *IEEE Trans Inf Theory* 67(5):2581–2623. <https://doi.org/10.1109/TIT.2021.3062161>
- Stulp F, Sigaud O (2015) Many regression algorithms, one unified model: A review. *Neural Netw* 69:60–79. <https://doi.org/10.1016/j.neunet.2015.05.005>
- Fatima N (2020) Enhancing performance of a deep neural network: a comparative analysis of optimization algorithms. *ADCAIJ: Adv Distrib Comput Artif Intell J* 9(2):79–90. <https://doi.org/10.14201/ADCAIJ2020927990>
- Fernández-Delgado M et al (2019) An extensive experimental survey of regression methods. *Neural Netw* 111:11–34. <https://doi.org/10.1016/j.neunet.2018.12.010>
- Gajny L, et al (2017) Best L 1 approximation of Heaviside-type functions from Chebyshev and weak-Chebyshev spaces. *Numer Algorithms* 75:827–843. <https://doi.org/10.1007/s11075-016-0222-8>
- Panichi G et al (2025) Quantum physics informed neural networks for multi-variable partial differential equations. [arXiv:2503.12244](https://arxiv.org/abs/2503.12244)
- Gudise VG, Venayagamoorthy GK (2003) Comparison of particle swarm optimization and backpropagation as training algorithms for neural networks. In: Proceedings of the 2003 IEEE swarm intelligence symposium. SIS'03 (Cat. No.03EX706), pp 110–117. <https://doi.org/10.1109/SIS.2003.1202255>
- Gupta H et al (2022) Comparative performance analysis of quantum machine learning with deep learning for diabetes prediction. *Complex Intell Syst* 8(4):3073–3087. <https://doi.org/10.1007/s40747-021-00398-7>
- Hidalgo MC (2025) Comparative Analysis of a Quantum SVM With an Optimized Kernel Versus Classical SVMs. *IEEE Access* 13:82378–82387. <https://doi.org/10.1109/ACCESS.2025.3567352>
- Hoang D, Wiegatz K (2023) Machine learning methods in finance: Recent applications and prospects. *Eur Financ Manag* 29(5):1657–1701. <https://doi.org/10.1111/eufm.12408>
- Zaman K et al (2025) A comparative analysis of hybrid-quantum classical neural networks. In: Grid, cloud, and cluster computing; quantum technologies; and modeling, simulation and visualization methods, Springer Nature Switzerland, pp 102–115. [https://doi.org/10.1007/978-3-031-85884-0\\_9](https://doi.org/10.1007/978-3-031-85884-0_9)
- Anand K et al (2020) Black magic in deep learning: how human skill impacts network training. <https://doi.org/10.48550/arXiv.2008.05981>
- Khan S, Yairi T (2018) A review on the application of deep learning in system health management. *Mech Syst Signal Process* 107:241–265. <https://doi.org/10.1016/j.ymssp.2017.11.024>
- Killoran N et al (2019) Continuous-variable quantum neural networks. *Phys Rev Res* 1(3):033063. <https://doi.org/10.1103/PhysRevResearch.1.033063>
- Kea K et al (2024) A hybrid quantum-classical model for stock price prediction using quantum-enhanced long short-term memory. *Entropy* 26(11):954. <https://doi.org/10.3390/e26110954>
- Kingma DP, Ba J (2014) Adam: A method for stochastic optimization. [arXiv:1412.6980](https://arxiv.org/abs/1412.6980)
- Kwak Y et al (2021) Quantum neural networks: concepts, applications, and challenges. In: Twelfth international conference on ubiquitous and future networks (ICUFN) 2021, pp 413–416. <https://doi.org/10.1109/ICUFN49451.2021.9528698>
- Leonard J, Kramer MA (1990) Improvement of the backpropagation algorithm for training neural networks. *Comput Chem Eng* 14(3):337–341. [https://doi.org/10.1016/0098-1354\(90\)87070-6](https://doi.org/10.1016/0098-1354(90)87070-6)
- Liang S, Srikant R (2016) Why deep neural networks for function approximation? In: International conference on learning representations
- Long C, Huang M, Ye X, Futamura Y, Sakurai T (2025) Hybrid quantum-classical-quantum convolutional neural networks. *Sci Rep* 15(1):31780. <https://doi.org/10.1038/s41598-025-13417-1>

- Markidis S (2022) On physics-informed neural networks for quantum computers. *Front Appl Math Stat* 8:1036711. <https://doi.org/10.3389/fams.2022.1036711>
- Marugán AP et al (2018) A survey of artificial neural network in wind energy systems. *Appl Energy* 228:1822–1836. <https://doi.org/10.1016/j.apenergy.2018.07.084>
- Masini RP et al (2023) Machine learning advances for time series forecasting. *J Econ Surv* 37(1):76–111. <https://doi.org/10.1111/joes.12429>
- Pande MB (2024) A comprehensive review of data encoding techniques for quantum machine learning problems. In: Second international conference on emerging trends in information technology and engineering (ICETITE) 2024, pp 1–7. <https://doi.org/10.1109/ic-ETITE58242.2024.10493306>
- McClean JR et al (2018) Barren plateaus in quantum neural network training landscapes. *Nat Commun* 9(1):4812
- Nadim Md et al (2025) Quantum Vs. Classical machine learning algorithms for software defect prediction: challenges and opportunities. In: 2025 IEEE/ACM international workshop on quantum software engineering (Q-SE), pp 35–42. <https://doi.org/10.1109/Q-SE66736.2025.00012>
- Zhou M-G et al (2023) Quantum neural network for quantum neural computing. *Research* 6
- Sazlı MH (2006) A brief review of feed-forward neural networks. *Commun Faculty Sci Univ Ankara Ser A2–A3 Phys Sci Eng* 50(01). [https://doi.org/10.1501/commua1-2\\_0000000026](https://doi.org/10.1501/commua1-2_0000000026)
- Augustine MT (2024) A survey on universal approximation theorems. [arXiv:2407.12895](https://arxiv.org/abs/2407.12895)
- Nadim Md et al (2025) Comparative analysis of quantum and classical support vector classifiers for software bug prediction: an exploratory study. *Quantum Mach Intell* 7(1):1–17. <https://doi.org/10.1007/s42484-025-00236-w>
- Ketkar N, Moolayil J (2021) Automatic differentiation in deep learning, in deep learning with python: learn best practices of deep learning models with PyTorch, Apress, pp 133–145. [https://doi.org/10.1007/978-1-4842-5364-9\\_4](https://doi.org/10.1007/978-1-4842-5364-9_4)
- Gupta N et al (2021) Data quality for machine learning tasks. In: Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining, pp 4040–4041. <https://doi.org/10.1145/3447548.3470817>
- Olivier LE (2021) Gaussian process based model predictive control to address uncertain milling circuit dynamics *IFAC-PapersOnLine* 54(21):1–6. <https://doi.org/10.1016/j.ifacol.2021.12.001>
- Pedrycz W, Chen SM (2019) *Deep learning: algorithms and applications*. Springer International Publishing
- Qi H et al (2023) The barren plateaus of quantum neural networks: review, taxonomy and trends. *Quantum Inf Process* 22(12):435. <https://doi.org/10.1007/s11128-023-04188-7>
- Qian Y et al (2024) The dilemma of quantum neural networks. *IEEE Trans Neural Netw Learn Syst* 35(4):5603–5615. <https://doi.org/10.1109/TNNLS.2022.3208313>
- Mishra R et al (2020) A survey on deep neural network compression: Challenges, overview, and solutions. [arXiv:2010.03954](https://arxiv.org/abs/2010.03954)
- Ranga D et al (2024) Quantum machine learning: exploring the role of data encoding techniques. *Challenges Future Direct Math* 12(21):3318. <https://doi.org/10.3390/math12213318>
- Rath M, Date H (2024) Quantum data encoding: A comparative analysis of classical-to-quantum mapping techniques and their impact on machine learning accuracy. *EPJ Quantum Technol* 11(1):72. <https://doi.org/10.1140/epjqt/s40507-024-00285-3>
- Zhao R, Wang S (2021) A review of quantum neural networks: methods, models. *Dilemma*. [arXiv:2109.01840](https://arxiv.org/abs/2109.01840)
- Robert H-N (1992) III.3 - Theory of the backpropagation neural network. In: *Neural networks for perception*, Academic Press, pp 65–93. <https://doi.org/10.1016/B978-0-12-741252-8.50010-8>
- Rupp M et al (2012) Fast and accurate modeling of molecular atomization energies with machine learning. *Phys Rev Lett* 108(5):058301. <https://doi.org/10.1103/PhysRevLett.108.058301>
- Schalkoff RJ (1997) *Artificial neural networks*, McGraw-Hill
- Schuld M et al (2014) The quest for a quantum neural network. *Quantum Inf Process* 13:2567–2586. <https://doi.org/10.1007/s11128-014-0809-8>
- Schuld M, Sweke R, Meyer JJ (2021) Effect of data encoding on the expressive power of variational quantum-machine-learning models. *Phys Rev A* 103:032430
- Genç S (2024) Performance analysis of quantum and classical machine learning models for feature selection and classification of the diabetes health indicators dataset. In: 2024 8th International artificial intelligence and data processing symposium (IDAP), pp 1–7. <https://doi.org/10.1109/IDAP64064.2024.10710904>
- Shi J et al (2020) An approach to cryptography based on continuous-variable quantum neural network. *Sci Rep* 10(1):2107. <https://doi.org/10.1038/s41598-020-58928-1>
- Shihao G et al (2020) Empirical asset pricing via machine learning. *Rev Financial Stud* 33(5):2223–2273. <https://doi.org/10.1093/rfs/hhaa009>
- Shinde PP, Shah S (2018) A review of machine learning and deep learning applications. In: Fourth international conference on computing communication control and automation (ICCUBEA) 2018, pp 1–6. <https://doi.org/10.1109/ICCUBEA.2018.8697857>
- Choe S, Perkowski M (2022) Continuous variable quantum mnist classifiers-classical-quantum hybrid quantum neural networks. *J Quantum Inf Sci* 12(2):37–51. <https://doi.org/10.4236/jqis.2022.122005>
- Soydancer D (2020) A comparison of optimization algorithms for deep learning. *Int J Pattern Recognit Artif Intell* 34(13):2052013. <https://doi.org/10.1142/S0218001420520138>
- Adam SP et al (2019) No free lunch theorem: a review. In: *Approximation and optimization : algorithms, complexity and applications*. Springer International Publishing, pp 57–82. [https://doi.org/10.1007/978-3-030-12767-1\\_5](https://doi.org/10.1007/978-3-030-12767-1_5)
- Suzuki T et al (2024) Quantum support vector machines for classification and regression on a trapped-ion quantum computer. *Quantum Mach Intell* 6(1):31. <https://doi.org/10.1007/s42484-024-00165-0>
- Tasnim T et al (2025) Quantum vs Classical: performance benchmarking of cnn and qcnn in binary image classification, In: 2025 IEEE 15th annual computing and communication workshop and conference (CCWC), pp 00203–00208. <https://doi.org/10.1109/CCWC62904.2025.10903816>
- Fellner T et al (2025) Quantum vs. classical: A comprehensive benchmark study for predicting time series with variational quantum machine learning. [arXiv:2504.12416](https://arxiv.org/abs/2504.12416)
- Weedbrook C et al (2012) Gaussian quantum information. *Rev Mod Phys* 84(2):621–669. <https://doi.org/10.1103/RevModPhys.84.621>
- Wendlinger M et al (2024) A comparative analysis of adversarial robustness for quantum and classical machine learning models. In: IEEE International conference on quantum computing and engineering (QCE) 2024, pp 1447–1457. <https://doi.org/10.1109/QCE60285.2024.00171>
- West MT et al (2023) Benchmarking adversarially robust quantum machine learning at scale. *Phys Rev Res* 5(2):023186. <https://doi.org/10.1103/PhysRevResearch.5.023186>
- Zhang J, Qu S (2021) Optimization of Backpropagation Neural Network under the Adaptive Genetic Algorithm. *Complexity* 2021(1):1718234. <https://doi.org/10.1155/2021/1718234>
- Zhou L et al (2019) Application of deep learning in food: a review. *Comprehen Rev Food Sci Food Safety* 18(6):1793–1811. <https://doi.org/10.1111/1541-4337.12492>