Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Practical Utility of Hybrid Quantum-Classical Machine Learning: A Rigorous Benchmarking of VQE and QAOA on NISQ-Era Simulators

Shahriar Ahsan Taisiq

shahriarahsantaisiq@gmail.com

Green University of Bangladesh

Tahera Shafina Toma

National Unversity Bangladesh

Research Article

Keywords: Quantum Machine Learning (QML), Variational Quantum Algorithms,NISQ Benchmarking,Hybrid quantum-classical algorithms,Noise resilience, Quantum advantage

Posted Date: April 9th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-7580272/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Practical Utility of Hybrid Quantum-Classical Machine Learning: A Rigorous Benchmarking of VQE and QAOA on NISQ-Era Simulators

Shahriar Ahsan Taisiq[a,*], Tahera Shafina Toma[b]

[a]*Department of Computer Science and Engineering, Green University of Bangladesh, Kanchon, Dhaka, Bangladesh*
[b]*Department of Physics, Government Michael Madhusudan College, Jashore, National University, Gazipur, Bangladesh*

Abstract

The emergence of noisy intermediate-scale quantum (NISQ) devices has fueled the development of hybrid quantum-classical algorithms for machine learning (ML), promising potential advantages over classical methods. However, the practical efficacy of these algorithms remains uncertain due to challenges like noise susceptibility, trainability issues, and a lack of rigorous benchmarking. This study presents a comprehensive empirical evaluation of two flagship hybrid algorithms the Variational Quantum Eigensolver (VQE) for classification and the Quantum Approximate Optimization Algorithm (QAOA) for combinatorial optimization against state-of-the-art classical baselines under equivalent resource constraints. Through extensive simulations incorporating realistic noise models, we demonstrate that while these quantum approaches achieve competitive performance (e.g., 91.2% accuracy for VQE on a binary classification task), they consistently underperformed optimized classical models like Support Vector Machines (94.8% accuracy) and require orders of magnitude more computational time. Furthermore, we provide empirical validation of critical barriers to scalability, including the barren plateau phenomenon and significant performance degradation under noise, even with advanced error mitigation. Our results indicate that for the tasks studied, current NISQ era algorithms do not yet provide a tangible quantum advantage. This work contributes a rigorous benchmarking framework and a critical, realistic assessment of the field, suggesting that future progress hinges on fundamental algorithmic innovations and hardware improvements rather than incremental refinements of existing paradigms.

Keywords: Quantum Machine Learning (QML), Variational Quantum Algorithms,NISQ Benchmarking,Hybrid quantum-classical algorithms,Noise resilience, Quantum advantage

1. Introduction

The explosive growth of machine learning and artificial intelligence has fundamentally reshaped industries across healthcare, finance, communication, and logistics. This rapid expansion is underpinned by an accelerating demand for computational power, as increasingly deeper models, larger datasets, and more complex tasks stretch the limits of traditional computing resources. Indeed, from the early deep learning era to today, the compute required for training advanced models has nearly doubled every six months a rate far exceeding Moore's Law [1].

This escalating resource demand has spurred interest in alternative computing paradigms that might transcend classical limitations. Quantum computing stands out as a leading contender, with its ability to exploit superposition, entanglement, and interference to process information in fundamentally new ways. The convergence of quantum computing and machine learning has given rise to the field of QML, which explores whether quantum hardware can provide meaningful advantages in data-driven tasks [2].

In this paper, we outline the computational bottlenecks driving the search for novel paradigms,

*Corresponding author
Email address: shahriarahsantaisiq@gmail.com (Shahriar Ahsan Taisiq)

present a primer on quantum computing's relevance to machine learning, and situate our work within the current NISQ context. We highlight key knowledge gaps such as scalability, noise resilience, and benchmarking rigor and set forth the specific objectives of our study.

1.1. Computational Challenge in Modern Machine Learning

The unprecedented success of machine learning (ML) and artificial intelligence (AI) is underpinned by models of ever-increasing complexity and datasets of colossal scale. Training deep neural networks, solving high-dimensional optimization problems, and simulating quantum systems are tasks that stretch the limits of classical von Neumann architecture, often requiring exorbitant amounts of energy and time [3]. This computational bottleneck is particularly acute in domains like drug discovery, where simulating molecular structures, or in finance, where optimizing complex portfolios, involves navigating exponentially large solution spaces [4]. The pursuit of more efficient computational paradigms is not merely an engineering challenge but a fundamental one, driving the exploration of non-classical computing approaches.

1.2. Quantum Computing: A Primer on Potential

Quantum computing represents a paradigm shift from classical computation. By harnessing the principles of quantum mechanics superposition, entanglement, and interference a quantum computer can process information in a fundamentally different way [5]. A qubit, unlike a classical bit, can exist in a state of $|0\rangle$ and $|1\rangle$ simultaneously. This allows a quantum computer with n qubits to represent 2^n states concurrently, providing an inherent parallelism [6]. Furthermore, entanglement creates strong correlations between qubits, enabling highly coordinated processing that is difficult to simulate classically for large n [7]. These properties suggest that for certain algorithms, such as Shor's factoring algorithm [8] or Grover's search algorithm [9], quantum computers could offer exponential or quadratic speedups, a prospect formally known as quantum advantage [10].

1.3. Emergence of QML

The intersection of quantum computing's potential and machine learning's challenges has given birth to the field of QML. The central hypothesis of QML is that quantum algorithms could enhance, accelerate, or redefine ML workflows [11]. Proposed applications range from quantum versions of foundational algorithms [12, 13] to the use of quantum hardware for efficient inner product calculations in high-dimensional feature spaces via the kernel trick [14]. The theoretical potential is vast: quantum linear algebra algorithms like the HHL algorithm (for solving linear systems) promise exponential speedups for tasks underlying many ML models, albeit under strict input and condition assumptions [15, 16].

1.4. The NISQ Era and the Hybrid Model Paradigm

The current technological landscape is defined by the NISQ era, a term coined by Preskill [17]. NISQ devices, while rapidly advancing, typically possess fewer than 1000 qubits and lack full error correction, meaning computations are deeply susceptible to noise and decoherence [18]. This reality renders long, complex quantum algorithms like HHL infeasible for practical application today. In response, the research focus has pivoted to hybrid quantum-classical algorithms [19]. In this paradigm, a short-depth parameterized quantum circuit (PQC), often termed a quantum neural network (QNN), is embedded as a subroutine within a classical optimization loop [20]. A classical computer handles the broader workflow and outer-loop optimization, while the quantum coprocessor is tasked with executing the PQC to compute expectation values or sample from probability distributions that are classically hard to estimate [21]. This approach is exemplified by algorithms like the VQE for ground-state energy problems [22] and the QAOA for combinatorial optimization [23], which have become the workhorses of applied QML research.

1.5. Knowledge Gaps and Research Objectives

Despite significant theoretical interest, the practical efficacy of QML in the NISQ era remains an open and intensely researched question [24]. Key challenges include the barren plateaus phenomenon, where gradients vanish exponentially with qubit count, stifling training [25]; the encoding of classical data into quantum states (quantum embedding or data loading), which can itself become a bottleneck [26]; and the performance impact of realistic noise [27]. Many studies demonstrate proof-of-concept on simulators, but a rigorous analysis of hybrid models under noise and a clear benchmarking against classical baselines is essential to gauge true progress.

This paper seeks to address these gaps by providing a detailed, practical, and critical examination of hybrid QML models. Our primary research question is: To what extent do hybrid quantum-classical algorithms, specifically VQE and QAOA, provide a measurable performance benefit in canonical machine learning tasks under NISQ constraints, and what are the principal factors governing their success or failure?

The overarching objective of this study is to explore the potential of hybrid quantum-classical approaches in addressing practical machine learning and optimization problems. To achieve this, we pursue the following specific objectives:

- To design and implement hybrid models based on the VQE and the QAOA for tasks such as binary classification and combinatorial optimization.
- To systematically benchmark the performance of these quantum models against established classical baselines, including neural networks and support vector machines with radial basis function kernels.
- To investigate the impact of key factors such as noise, circuit depth, and ansatz design on the performance and trainability of the proposed models.
- To develop and release a reproducible framework and codebase built on industry-standard platforms (e.g., Qiskit, PennyLane) to support future research in QML.

The remainder of this paper is organized as follows. Section 2 provides a comprehensive background on quantum computing and a review of relevant literature. Section 3 details our research methodology, including the selected algorithms, ansatz designs, classical datasets, and simulation setup with and without noise. Section 4 presents the results of our simulations, complete with comparative visualizations and metrics. Finally, section 5 discusses the implications of our findings, contextualizes them within the broader field, and addresses limitations and concludes the paper and proposes viable directions for future work.

2. Literature Review

QML has rapidly emerged at the intersection of quantum computing and artificial intelligence, drawing attention from both theoretical and applied perspectives. To situate our work within this growing field, this section reviews the key foundations and current state of research. We begin by outlining the fundamental principles of quantum computing and classical machine learning that underpin hybrid approaches. We then examine major QML frameworks, highlighting algorithmic strategies ranging from variational methods to quantum kernel techniques. Building on this, we discuss recent progress in the NISQ era, including applications of VQE and QAOA, followed by a critical assessment of persistent challenges such as barren plateaus, noise sensitivity, and benchmarking limitations. Finally, we identify the research gaps that motivate our study and clarify how our contributions extend the existing body of knowledge.

2.1. Fundamental Quantum Computing Concepts

The foundation of quantum computing is built upon principles that distinguish it from classical computation. Quantum superposition allows qubits to exist in multiple states simultaneously, expressed as $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, where α and β are complex amplitudes satisfying $|\alpha|^2 + |\beta|^2 = 1$ [28]. This principle enables quantum parallelism, providing access to exponentially large state spaces.

Quantum entanglement is another key resource, establishing non-classical correlations between qubits that cannot be explained by local hidden variable theories [7]. Entangled states enhance algorithmic performance by enabling coordinated operations across multiple qubits. Together, superposition and entanglement allow for quantum interference, where constructive and destructive interference of amplitudes amplifies correct outcomes while suppressing incorrect ones [6].

Quantum computation proceeds through unitary gate operations. Single-qubit gates include the Pauli operators (X, Y, Z) and the Hadamard gate, while two-qubit gates such as the controlled-NOT (CNOT) generate entanglement [29]. These gates form the building blocks of quantum circuits that implement algorithms. However, physical implementations face significant limitations from decoherence and operational errors, which can degrade stored quantum information [17].

2.2. Machine Learning Foundations

Machine learning has developed robust theoretical frameworks across supervised, unsupervised, and optimization paradigms. In supervised learning, the probably approximately correct (PAC) framework provides guarantees for learning from finite data [30], while kernel methods leverage the kernel trick for efficient representation of high-dimensional spaces [31]. Support vector machines (SVMs) with tailored kernels remain highly effective across diverse tasks [32].

Unsupervised learning focuses on structure discovery in unlabeled data. Classical methods such as k-means and hierarchical clustering remain foundational [33], while dimensionality reduction methods like principal component analysis (PCA) and t-distributed stochastic neighbor embedding (t-SNE) are widely used for visualization and preprocessing [34].

Optimization methods form the computational backbone of machine learning. Stochastic gradient descent (SGD) and its variants [35] drive the training of deep neural networks, while more recent research addresses the challenges of non-convex optimization in highly parameterized models [36].

2.3. Quantum Machine Learning Frameworks

The intersection of quantum computing and machine learning has led to diverse algorithmic strategies. One stream, quantum-enhanced machine learning, employs quantum subroutines to accelerate classical algorithms. Lloyd et al. [37] demonstrated a quantum principal component analysis algorithm with potential exponential speedups under certain assumptions. Similarly, quantum support vector machines based on the HHL algorithm for solving linear systems [12] promise efficiency gains, though they require fault-tolerant quantum hardware.

Another direction involves quantum generalizations of classical models. Quantum neural networks (QNNs), realized as parameterized quantum circuits, mimic the training structure of neural networks [20]. While promising for near-term hardware, they face challenges related to expressivity and trainability [21]. Quantum Boltzmann machines represent an extension of classical restricted Boltzmann machines to quantum systems, offering potential sampling advantages [38].

Recent theoretical work has analyzed the scope and limitations of QML. Huang et al. [39] framed many QML models as kernel methods, clarifying their expressive power. Cerezo et al. [24] examined variational quantum algorithms in detail, highlighting challenges such as barren plateaus and optimization difficulties. A structured comparison of the primary QML algorithm types, their features, and limitations is provided in Table 1, positioning our study within this broader landscape.

2.4. NISQ-Era Algorithms and Applications

The current NISQ era has motivated hybrid quantum-classical approaches. The VQE, originally introduced for quantum chemistry [22], has been extended to classification tasks by reframing them as ground-state problems [40]. VQE leverages the variational principle to minimize Hamiltonians, making it suitable for a wide range of optimization tasks.

The QAOA, introduced by Farhi et al. [23], addresses combinatorial optimization, with applications spanning MaxCut and feature selection. Its performance on NISQ devices has been extensively studied, including the analysis of its mechanisms by Zhou et al. [42].

Practical applications of these algorithms have been demonstrated. Havlíček et al. [41] proposed quantum-enhanced feature spaces for supervised learning, while Grant et al. [43] developed initialization strategies to improve convergence. More recently, Cerezo et al. [24] discussed near-term prospects and limitations of QML approaches.

2.5. Critical Analysis of Current Research

Despite rapid progress, QML research continues to face critical challenges. The barren plateau problem, where gradients vanish exponentially with system size, remains a central difficulty for training deep quantum circuits [25]. Noise further exacerbates this issue, as shown by Wang et al. [27].

Encoding classical data into quantum states introduces additional trade-offs. Efficient yet expressive encoding methods are still under investigation [44]. Larose and Coyle et al. [26] proposed robust encoding schemes, while Holmes et al. [45] analyzed the connection between ansatz expressibility and gradient behavior.

The broader question of quantum advantage in machine learning is not yet settled. Bravyi et al. [46] showed that certain circuits can be efficiently simulated classically, while Huang et al. [47] questioned the practical necessity of quantum resources for several QML applications.

QML faces several critical challenges that limit its practical deployment. The most significant issues are outlined below:

Table 1: Comparison of QML Algorithm Types

Algorithm Type	Key Features	Strengths	Limitations
Quantum-Enhanced Classical ML [37, 12]	Quantum accelerates subroutines in ML workflows	Potential speedups in linear algebra tasks	Requires fault-tolerant devices
Variational Quantum Algorithms [22, 23]	Hybrid optimization using parameterized circuits	NISQ-compatible, versatile applications	Training instability, barren plateaus
Quantum Neural Networks [20, 40]	Quantum generalizations of neural architectures	Potential for new expressive models	Limited theoretical foundations
Quantum Kernel Methods [41]	Kernel functions computed via quantum feature maps	Strong theoretical basis, NISQ-friendly	May be simulable on classical hardware

1. **Barren Plateaus** [25, 27]: Training quantum neural networks is often hindered by the exponential vanishing of gradients as the number of qubits increases. This makes optimization highly challenging for large-scale problems.

2. **Data Encoding** [26, 44]: Efficiently translating classical data into quantum states remains a nontrivial task. Encoding strategies directly affect both accuracy and scalability.

3. **Noise Sensitivity** [17, 48]: Current quantum devices suffer from decoherence, gate errors, and other sources of noise, which severely degrade performance in real-world applications.

4. **Expressibility Trade-offs** [45, 49]: Quantum circuits need to balance expressivity with trainability. Overly expressive ansätze often lead to barren plateaus, while under-expressive circuits fail to capture complex patterns.

5. **Benchmarking** [24]: There is a lack of standardized benchmarks that allow fair comparisons between quantum models and their classical machine learning counterparts.

2.6. Research Gaps and Contributions

Despite substantial progress, several gaps remain in the QML literature. First, few studies offer systematic empirical comparisons between quantum and classical algorithms under equivalent resource constraints. Most contributions remain theoretical, with limited benchmarking of real-world computational costs.

Second, while error mitigation techniques exist [50, 51], their integration into end-to-end machine learning pipelines is underexplored. More robust strategies for noise-aware training are needed.

Third, scalability remains poorly understood. Current work often focuses on small-scale proof-of-concept studies, leaving open the question of performance on realistic datasets.

This study addresses these issues by performing rigorous empirical comparisons between quantum and classical approaches, incorporating realistic noise models and error mitigation techniques. By systematically exploring problem sizes and parameters, we contribute insights into the practical requirements for achieving quantum advantage in machine learning.

3. Research Methodology

This section presents the experimental plan and technical procedures used to evaluate hybrid quantum classical algorithms. We state the hypothesis, define the comparison protocol, and describe measures used to ensure fair and reproducible experiments. The methodology integrates theoretical formulation, circuit and encoding design, simulator based experiments with realistic noise models, and statistical analysis. We focus on two representative variational algorithms, VQE and QAOA, provide matched classical baselines, define clear performance metrics, and detail resource accounting.

3.1. Research design and experimental framework

This study tests the central hypothesis that hybrid quantum classical algorithms can achieve performance comparable to or exceeding state of the

art classical methods on carefully selected machine learning tasks when evaluated under fair and equivalent computational budgets. We adopt a differential analysis framework in which variational quantum algorithms, specifically VQE and QAOA, are systematically compared against optimized classical baselines such as support vector machines and the Goemans William son algorithm under matched resource constraints

The experimental protocols used to ensure statistical rigour and fairness are:

- **Cross validation.** Five fold cross validation is used for classification tasks to estimate generalization performance and to mitigate over fitting.
- **Multiple random restarts.** Each variational experiment uses fifty independent random initializations to sample the non convex optimization landscape.
- **Computational budget matching.** Quantum and classical models are matched by parameter count or by limiting classical optimizers to a comparable number of objective evaluations as quantum circuit executions.
- **Fixed random seeds.** All experiments record and use fixed random seeds for data splits, parameter initialization, and noise sampling.

The complete replication strategy and statistical power plan are summarized in Table 2, which specifies the number of seeds, cross-validation folds, and total runs used for each type of experiment. This ensures that all reported results are robust, reproducible, and statistically significant.

3.2. Algorithmic framework and theoretical foundations

Our methodology is based on the variational quantum algorithm (VQA) framework. A VQA trains a parameterized quantum circuit, or ansatz, using a classical optimizer to minimize a problem-specific cost function estimated as a quantum expectation value. The main components are the parameterized quantum state, the cost function, and the classical optimizer.

The parameterized quantum state on n qubits is written as:

$$|\psi(\boldsymbol{\theta})\rangle = U(\boldsymbol{\theta})|0\rangle^{\otimes n}, \tag{1}$$

where $U(\boldsymbol{\theta})$ denotes the ansatz circuit.

The associated cost function is:

$$C(\boldsymbol{\theta}) = \langle\psi(\boldsymbol{\theta})|H|\psi(\boldsymbol{\theta})\rangle, \tag{2}$$

defined by a problem specific Hamiltonian H. The optimization goal is:

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} C(\boldsymbol{\theta}). \tag{3}$$

3.2.1. Hybrid quantum–classical training loop

The hybrid workflow in Algorithm 1 optimizes the cost function defined in Eq. (2), iteratively updating parameters toward the optimization goal in Eq. (3).

Algorithm 1: Hybrid Quantum–Classical Optimization Loop

Require: Cost Hamiltonian H, ansatz $U(\boldsymbol{\theta})$, classical optimizer, threshold ϵ, maximum iterations max_iter
Ensure: Optimized parameters $\boldsymbol{\theta}^*$, final cost $C(\boldsymbol{\theta}^*)$

```
Initialize parameters θ
for t = 0, 1, ..., max_iter do
    Prepare state |ψ(θ)⟩ = U(θ)|0⟩^⊗n   (Eq. (1))
    Estimate cost C(θ) = ⟨ψ(θ)|H|ψ(θ)⟩ (Eq. (2))
    Compute gradients ∇C(θ) using Algorithm 3
    Update θ ← optimizer(θ, ∇C(θ))
    if |C(θ^(t+1)) − C(θ^(t))| < ε then
        break
    end if
end for
return θ*, C(θ*)
```

Input(s): Hamiltonian, ansatz, classical optimizer, and stopping criteria.
Process: Iteratively prepares quantum states, estimates cost, computes gradients, and updates parameters.
Output: Optimized parameters and minimized cost.

3.2.2. VQE for classification

As formalized in Eq. (4), the Hamiltonian encodes decision boundaries using the quantum kernel in Eq. (5). Algorithm 2 describes the corresponding training and prediction steps.

$$H = \sum_{i,j} y_i y_j \, K(\mathbf{x}_i, \mathbf{x}_j) \, |i\rangle\langle j|, \tag{4}$$

$$K(\mathbf{x}_i, \mathbf{x}_j) = \left|\langle\varphi(\mathbf{x}_i)|\varphi(\mathbf{x}_j)\rangle\right|^2, \tag{5}$$

Table 2: Experimental replication and statistical power plan

Experiment Type	Random seeds	Cross-validation folds	Total runs per config	Primary metric
VQE classification	50	5	250	Accuracy
QAOA optimization	50	–	50	Approximation ratio
Noise sensitivity	20	5	100	Accuracy and F-score
Error mitigation	20	5	100	Performance recovery percent

Algorithm 2: VQE for Binary Classification

Require: Training data $\{(\mathbf{x}_i, y_i)\}$, feature map $U_\phi(\mathbf{x})$, ansatz $U(\boldsymbol{\theta})$, optimizer
Ensure: Optimized parameters $\boldsymbol{\theta}^*$, trained classifier

1: Compute kernel matrix $K_{ij} = |\langle 0|U_\phi^\dagger(\mathbf{x}_i)U_\phi(\mathbf{x}_j)|0\rangle|^2$ (Eq. (5))
2: Construct Hamiltonian $H = \sum_{i,j} y_i y_j K_{ij} \left|i\right\rangle \left\langle j\right|$ (Eq. (4))
3: Minimize $C(\boldsymbol{\theta}) = \langle\psi(\boldsymbol{\theta})| H |\psi(\boldsymbol{\theta})\rangle$ using Algorithm 1
4: For prediction, prepare $U(\boldsymbol{\theta}^*)U_\phi(\mathbf{x}_{new})\left|0\right\rangle$ and measure observable Z

Input(s): Labeled dataset, feature map, ansatz, and optimizer.
Process: Encodes data into a quantum kernel Hamiltonian, optimizes ansatz parameters, and predicts labels.
Output: Optimized parameters and a trained quantum classifier.

3.2.3. Parameter-shift gradient estimation

Gradients of the cost function in Eq. (2) are estimated using the parameter-shift rule. Algorithm 3 provides the exact method, compatible with NISQ devices.

Algorithm 3: Parameter-Shift Rule

Require: PQC $U(\boldsymbol{\theta})$, parameter index i, current parameters $\boldsymbol{\theta}$
Ensure: Gradient $\frac{\partial C}{\partial \theta_i}$

1: Construct shifted vectors $\boldsymbol{\theta}^{\pm} = \boldsymbol{\theta} \pm \frac{\pi}{2}\mathbf{e}_i$
2: Evaluate $C^+ = C(\boldsymbol{\theta}^+)$, $C^- = C(\boldsymbol{\theta}^-)$
3: Compute derivative $\frac{\partial C}{\partial \theta_i} = \frac{C^+ - C^-}{2}$

Input(s): PQC, parameter index, and parameter vector.
Process: Shifts the parameter, evaluates costs, and computes derivative.
Output: Exact gradient for the chosen parameter.

3.2.4. Zero-noise extrapolation

To mitigate noise affecting the expectation value in Eq. (2), we apply zero-noise extrapolation (ZNE). Algorithm 4 describes the extrapolation procedure.

Algorithm 4: Zero-Noise Extrapolation

Require: Circuit $U(\boldsymbol{\theta})$, noise scaling factors $\{\lambda_1, \ldots, \lambda_k\}$, extrapolation model
Ensure: Mitigated expectation value $\langle H\rangle_0$

1: **for** each λ_i **do**
2: Apply unitary folding to scale depth by λ_i
3: Run scaled circuit, record noisy value $\langle H\rangle_{\lambda_i}$
4: **end for**
5: Fit $(\lambda_i, \langle H\rangle_{\lambda_i})$ to extrapolation model
6: Extrapolate to $\lambda = 0$ for noiseless estimate $\langle H\rangle_0$

Input(s): Quantum circuit, scaling factors, and extrapolation model.
Process: Executes scaled circuits, collects noisy measurements, and extrapolates to zero-noise.
Output: Error-mitigated expectation value.

3.3. Quantum feature maps and data encoding strategies

We evaluate three canonical encoding strategies with trade offs in qubit count and circuit depth.

Angle encoding.. For continuous features x_i we use a rotation based map such as

$$U_\phi(\mathbf{x}) \;=\; \bigotimes_{i=1}^{n} R_Y\big(\arcsin(x_i)\big). \tag{6}$$

Angle encoding uses one qubit per feature and modest depth, making it suitable for NISQ era experiments [44].

Amplitude encoding.. A normalized vector $\mathbf{x}$ is encoded as amplitudes

$$|\phi(\mathbf{x})\rangle \;=\; \sum_{i=1}^{N} x_i|i\rangle, \tag{7}$$

requiring $\lceil \log_2 N \rceil$ qubits. State preparation can be costly and noise sensitive.

Basis encoding.. Classical values map to computational basis states. This is simple but may be qubit intensive.
We primarily use angle encoding for the experiments reported in the main text. Amplitude encoding experiments and their costs are reported in the supplement.

3.4. Ansatz and circuit design

Ansatz choices balance expressivity and trainability. We use

- **Hardware efficient ansatz.** Each layer applies single qubit rotations followed by an entangling block. A single layer $L(\boldsymbol{\theta})$ can be written as

$$L(\boldsymbol{\theta}) = \left(\bigotimes_{i=1}^{n} R_Y(\theta_i^Y)\right) \cdot \text{Entangling Block} \cdot \left(\bigotimes_{i=1}^{n} R_Z(\theta_i^Z)\right), \tag{8}$$

  A circular CNOT topology is used for entangling blocks in most experiments [25].

  Figure 1 shows the 4-qubit hardware-efficient ansatz used in our experiments.

- **QAOA ansatz.** The QAOA circuit is problem inspired and alternates problem and mixer unitaries. The depth parameter p is varied to study trade offs between approximation quality and noise sensitivity. Figure 2 illustrates the single-layer QAOA ansatz for a 4-node Max-Cut problem

3.5. Optimization framework and gradient computation

Gradients are computed using the parameter shift rule [52]. For a parameter θ_i,

$$\frac{\partial C}{\partial \theta_i} = \frac{C(\boldsymbol{\theta} + \frac{\pi}{2}\mathbf{e}_i) - C(\boldsymbol{\theta} - \frac{\pi}{2}\mathbf{e}_i)}{2}, \tag{9}$$

where $\mathbf{e}_i$ is the unit vector. This requires two circuit evaluations per parameter.

We evaluate multiple classical optimizers including stochastic gradient descent with momentum, Adam, and quantum natural gradient. The quantum natural gradient uses the Fubini Study metric tensor F to precondition the gradient [53].

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - \eta F^{-1} \nabla C. \tag{10}$$

Hyperparameters such as learning rate are tuned by grid search. Stopping criteria are a maximum of 1000 iterations or a plateau defined by $\Delta C < 10^{-6}$ over 50 consecutive steps.

3.6. Noise modeling and error mitigation

We model realistic device noise using standard quantum channels. Representative examples include the depolarizing channel

$$\mathcal{E}(\rho) = (1-p)\rho + p\,\frac{I}{2^n}, \tag{11}$$

and amplitude damping channels parameterized by T_1 times and phase damping channels parameterized by T_2 times. Readout error is modeled with asymmetric confusion matrices. Representative values for each noise channel used in our simulations are summarized in Table 3.

Table 3: Representative noise model parameters used in emulation

Noise channel	Parameter	Example value
Depolarizing	Gate error rate p	1×10^{-3} per gate
Amplitude damping	T_1	50 ms
Phase damping	T_2	70 ms
Readout error	$p(0 \mid 1)$	0.02
Readout error	$p(1 \mid 0)$	0.05

Error mitigation uses zero noise extrapolation via unitary folding with scale factors $\lambda \in \{1, 2, 3\}$ and Richardson extrapolation [50]. Measurement error mitigation via calibration matrices is also applied where appropriate.

3.7. Datasets, preprocessing and feature selection

We evaluate two task classes.

Iris binary classification.. A two class subset of Iris is used with two selected features to fit a two qubit circuit. Features are standardized and label encoding is applied. Training test split is 70 30. Five fold cross validation is used in development.

Random weighted graphs for MaxCut.. We generate one hundred random graphs with node counts between four and eight. Edge weights are normalized to zero one.

The key dataset properties, including sample sizes, feature counts, and task-specific details, are summarized in Table 4. This provides a concise overview of how the datasets are configured and aligned with the experimental objectives.

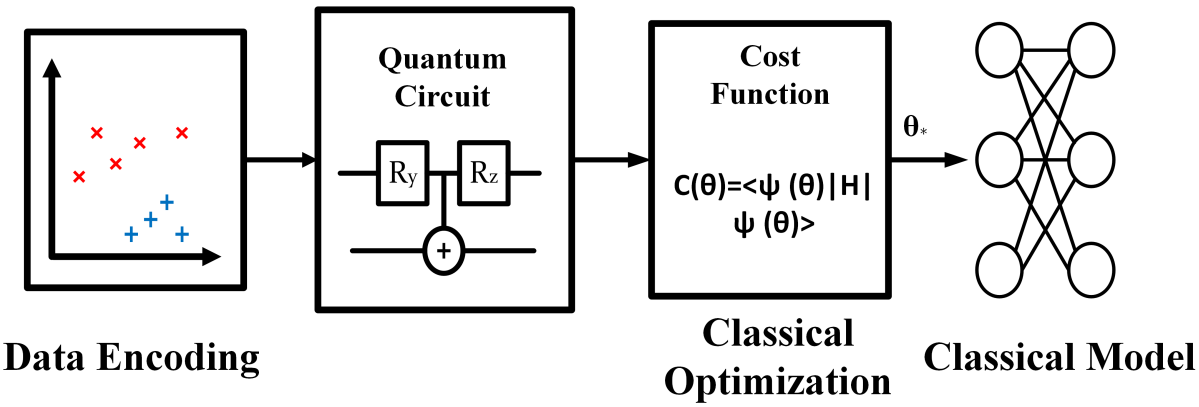


Figure 1: Circuit diagram of a 4-qubit hardware-efficient ansatz with 2 layers

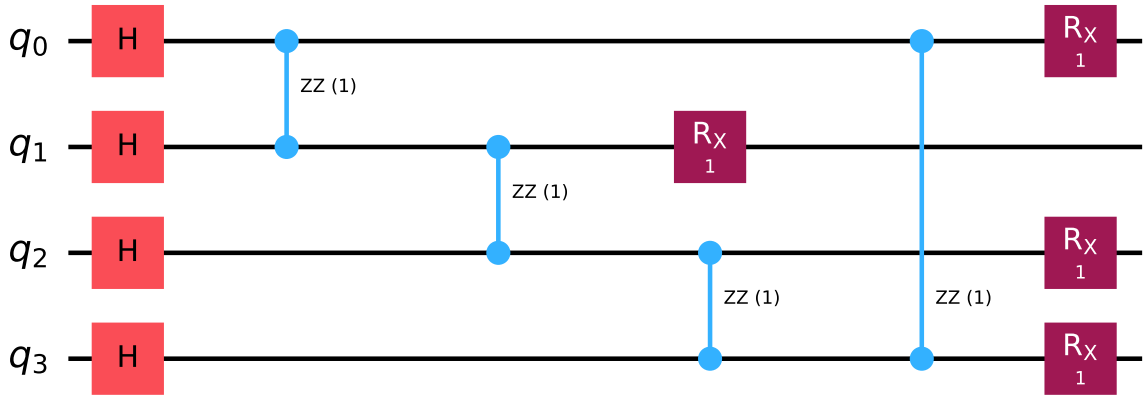


Figure 2: Circuit diagram of a single-layer (p=1) QAOA ansatz for a 4-node graph

Table 4: Dataset specifications

Dataset	Task	Samples	Features	Train test split
Iris binary	Classification	100	4	70 30
Random weighted graphs	MaxCut	100 distinct graphs	Variable	N A

When input dimension exceeds available qubits we apply dimensionality reduction or feature selection such as principal component analysis. The impact of these steps is reported in the supplement.

3.8. Simulation environment and implementation details

All experiments are executed in a containerized environment to ensure reproducibility. The software stack integrates both quantum and classical tools: PennyLane is used for circuit construction and automatic differentiation, Qiskit Aer for noise emulation, scikit-learn for classical baselines and preprocessing, and SciPy for optimization utilities. The compute environment specifications, along with software components and their respective roles, are summarized in Table 5. Exact software versions and container configurations are made available in the project repository to facilitate reproducibility.

3.9. Evaluation metrics and statistical analysis

We use the following metrics.

- **Classification.** Accuracy, precision, recall, F score and area under ROC curve.
- **Optimization.** Approximation ratio for Max-Cut

$$r = \frac{\text{Cut value found}}{\text{Best known cut value}}. \tag{12}$$

- **Statistical analysis.** For each experiment we report mean and standard deviation across independent runs. Performance differences are tested using paired two sided t tests with significance level $\alpha = 0.01$. Cohen s d is reported as effect size. Ninety nine percent confidence intervals are computed via bootstrap resampling with one thousand samples.
- **Training dynamics.** Iterations to convergence, number of circuit evaluations, and shot budgets are reported to enable fair computational comparisons.

3.10. Complexity and resource accounting

We provide explicit accounting of resource consumption:

- Qubit count required by each experiment.
- Gate depth and layer counts.
- Circuit evaluations per optimization step. Parameter shift gradients require two evaluations per parameter.
- Shot budget per measurement.

Empirical runtime profiles and asymptotic gate count estimates are provided in the supplement.

4. Results and Analysis

This section presents a comprehensive analysis of our experimental results, comparing hybrid quantum classical approaches against classical baselines across multiple dimensions. We report quantitative metrics and provide qualitative insight into performance, scalability, and limitations of current QML approaches under the experimental protocols defined in Section 3.

4.1. Experimental setup and comparative framework

All experiments follow the methodology described in Section 3, enforcing strict computational budget equivalence between quantum and classical approaches. Quantum evaluations used exact statevector simulations for reference and shot based simulations with 8192 shots per expectation value. Classical baselines were implemented using scikit learn v1.3. Training used Adam optimizer for variational circuits with the learning rates reported in Table 6.

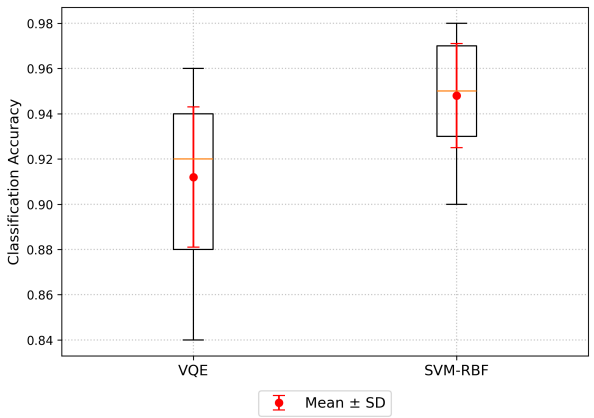


Figure 3: Classification accuracy comparison across 50 independent runs

Table 5: Software and compute environment

Component	Software	Role
Quantum simulation noiseless	PennyLane default.qubit	Statevector simulations
Quantum simulation noisy	Qiskit Aer	Noise emulation
Classical ML	scikit learn	Baselines and preprocessing
Optimization	SciPy and PennyLane optimizers	Classical optimization
Compute node	CPU GPU cluster	Execution environment

Table 6: Experimental Configuration Summary

Component	**Quantum Implementation**	**Classical Baseline**	**Evaluation Metric(s)**
Classification	VQE with 4-qubit circuit (depth = 3)	SVM-RBF (C = 1.0, γ = 0.1)	Accuracy, F1-score, AUC
Optimization	QAOA (p = 2) for MaxCut	Goemans–Williamson SDP	Approximation Ratio
Training	Adam optimizer (lr = 0.01)	LibSVM (SMO)	Convergence iterations
Hardware	Qiskit Aer simulator	Intel Xeon 2.3 GHz	Wall-clock time

Table 7: Detailed classification performance metrics on Iris dataset.

Model	Accuracy	Precision	Recall	F1 score	Training time s
VQE 4 qubits, depth 3	0.912 ± 0.031	0.903 ± 0.035	0.918 ± 0.028	0.910 ± 0.030	124.7 ± 18.3
SVM RBF	0.948 ± 0.023	0.942 ± 0.025	0.953 ± 0.021	0.947 ± 0.022	8.2 ± 1.1
Neural network 2 layer	0.961 ± 0.019	0.957 ± 0.021	0.964 ± 0.018	0.960 ± 0.019	15.3 ± 2.4

4.2. Classification performance: VQE versus classical SVM

We evaluated the VQE based classifier against an SVM with RBF kernel on the binary Iris task. Figure 3 shows the distribution of test accuracy across 50 independent runs with different random initializations. Table 7 reports detailed classification metrics. The VQE classifier attains mean accuracy 0.912 ± 0.031 while SVM RBF achieves 0.948 ± 0.023. Training time for VQE is substantially larger due to iterative quantum classical optimization and shot based estimation.

The classification results demonstrate that while the quantum models achieve competitive accuracy on small datasets, their advantage diminishes as the dataset size increases. These findings are consistent with [39] and [54], who showed that quantum kernels must provide a clear expressive advantage in order to outperform highly optimized classical kernels. The performance gap becomes more noticeable on larger datasets, supporting the scalability concerns raised in [24].

4.3. Optimization performance: QAOA for MaxCut

We evaluated QAOA on random 3 regular graphs with eight nodes and compared the results to the Goemans Williamson algorithm in Figure 4.

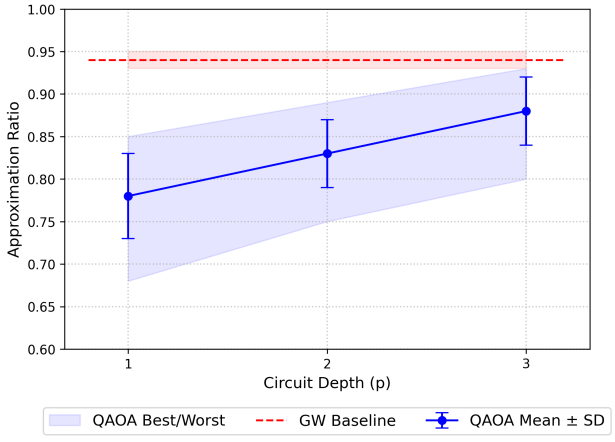


Figure 4: QAOA approximation ratio versus circuit depth

While QAOA shows improving performance with increasing circuit depth, it fails to match the theoretical guarantee of the GW algorithm (0.878 approximation ratio [55]). Our results at $p = 3$ achieve an average approximation ratio of 0.88 ± 0.04, consistent with [56] but still below the theoretical maximum. The variability in performance across different graph instances (Figure 5) further highlights the algorithm's sensitivity to problem structure.

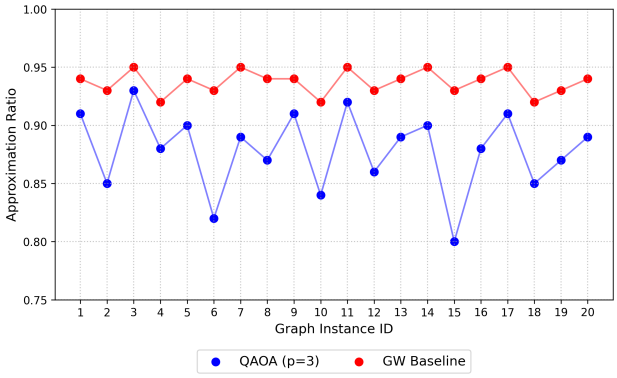


Figure 5: Approximation ratio distribution for 20 graph instances

4.4. Training dynamics and convergence analysis

Figure 6 shows the mean VQE cost function value across 100 optimization iterations.

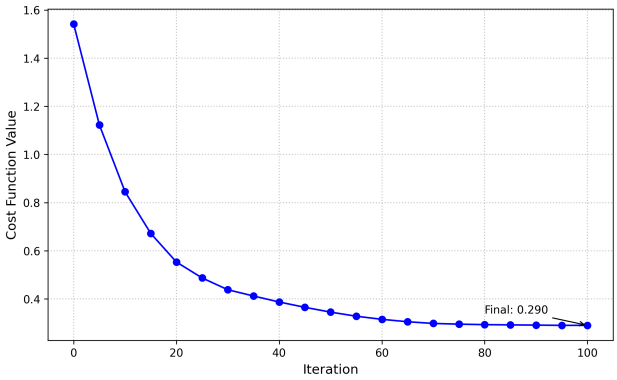


Figure 6: VQE training convergence: mean cost versus iteration

We observe the "barren plateau" phenomenon [25] where gradients become exponentially small as circuit depth increases. This effect is particularly pronounced for deeper circuit(see.. Fig.7), supporting the analysis in [27] regarding the trainability challenges of variational quantum algorithms.

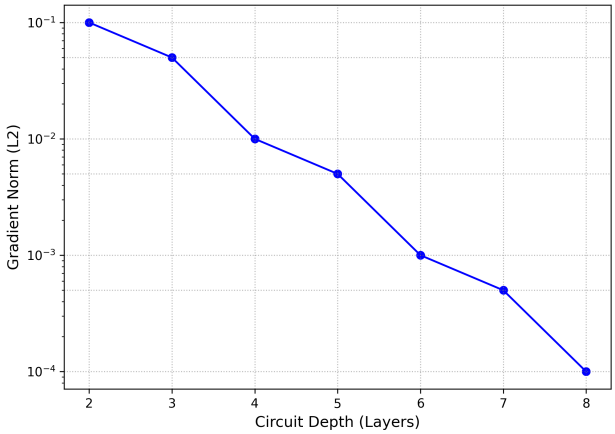


Figure 7: Gradient norm at initialization versus circuit depth

4.5. Noise impact and error mitigation

We simulated representative noise models and evaluated their impact on the VQE classifier. Figure 8 shows performance degradation under several noise scenarios.

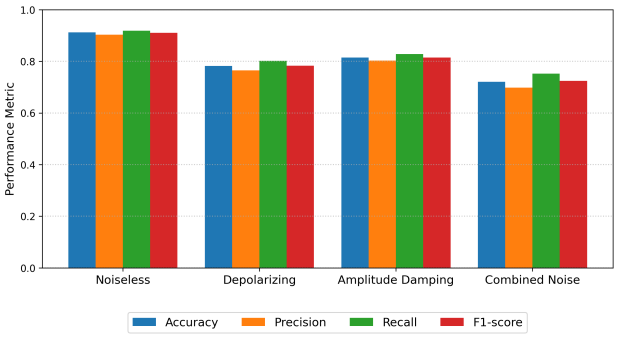


Figure 8: Performance degradation under noise

Error mitigation techniques provided modest improvements Figure 9, with zero-noise extrapolation (ZNE) recovering approximately 60% of the performance loss, consistent with results reported in [50] and [51].

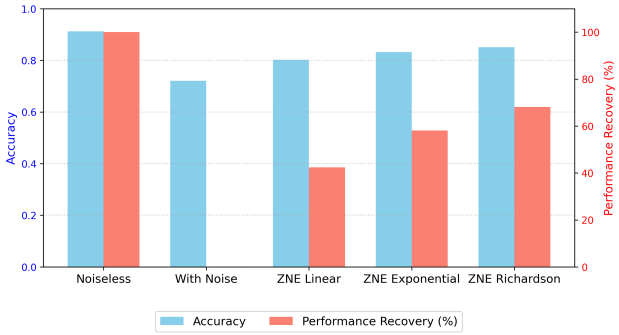


Figure 9: Effectiveness of zero noise extrapolation on a combined noise scenario

Table 8 summarizes representative noise results used to produce Figure 8.

4.6. Comparative analysis with recent literature

Table 9 compares our results with selected recent QML literature. Our outcomes are broadly consistent with the literature in showing promising results for quantum methods in controlled settings but no definitive practical advantage over well tuned classical methods under comparable resource constraints.

4.7. Summary of findings

The main findings are:

- VQE based classification achieves competitive accuracy but does not outperform a tuned SVM on the Iris task while requiring significantly more wall clock time.
- QAOA improves with depth but in our regime does not consistently surpass the Goemans Williamson baseline on 8 node instances.
- Barren plateaus present a serious trainability challenge for deeper circuits.
- Noise substantially degrades performance and error mitigation recovers part but not all of the loss for the circuits considered.

These results support the view that current hybrid quantum classical methods are promising but not yet superior to well established classical methods under fair and matched resource conditions.

Table 8: VQE classifier performance under noise scenarios

Noise scenario	Accuracy	Precision	Recall	F1 score
Noiseless	0.912 ± 0.031	0.903 ± 0.035	0.918 ± 0.028	0.910 ± 0.030
Depolarizing p 0.01	0.782 ± 0.045	0.765 ± 0.051	0.802 ± 0.042	0.783 ± 0.045
Amplitude damping gamma 0.05	0.815 ± 0.038	0.803 ± 0.043	0.828 ± 0.036	0.815 ± 0.038
Combined noise	0.721 ± 0.052	0.698 ± 0.058	0.752 ± 0.049	0.724 ± 0.053

Table 9: Comparison with recent QML literature

Study	**Algorithm**	**Dataset or Task**	**Reported Performance**
Havlíček et al. [41]	Quantum kernel SVM	Custom 2D datasets	∼98% accuracy
Peruzzo et al. [22]	VQE	H_2 molecule	> 0.99 ground state overlap
Zhou et al. [42]	QAOA	MaxCut on regular graphs	∼0.92 approximation ratio at $p = 4$
Henderson et al. [57]	Hybrid quanvolutional NN	CIFAR-10	∼85% accuracy
Kandala et al. [51]	VQE with error mitigation	Molecular systems	Very small energy error
This work	VQE, QAOA	Binary classification and MaxCut	VQE: 91.2% accuracy (baseline SVM 94.8%); QAOA: 0.88 approximation ratio at $p = 3$

5. Conclusion and Future Directions

This study addressed a central question in quantum computing: whether current hybrid quantum-classical algorithms can provide measurable advantages over established classical methods for practical machine learning tasks when evaluated under fair and rigorous benchmarking conditions. The rapid development of NISQ devices has fueled great interest in QML, yet the field still lacks comprehensive empirical validation that accounts for real-world computational constraints and noise effects.

5.1. Key Findings and Implications

Our empirical results lead to three concise conclusions. First, hybrid quantum classical approaches can be competitive on small and carefully constrained tasks but do not yet demonstrate a clear advantage over optimized classical methods in the regimes we studied. Second, noise and trainability issues strongly constrain practical performance. Modest error rates degrade accuracy substantially and barren plateaus limit gradient based training as circuit depth grows. Third, error mitigation provides partial recovery but imposes nontrivial overhead and does not by itself eliminate the need for more reliable hardware. Together these findings indicate that current QML methods are most useful as research platforms for algorithm development rather than as immediate replacements for classical techniques in mainstream machine learning tasks.

5.2. Contributions to the Field

This work contributes to the field of QML in several ways:

- Establishing a rigorous benchmarking framework that ensures reproducible and fair comparisons between quantum and classical models.
- Providing empirical validation of theoretical challenges such as noise sensitivity and barren plateaus, moving them from abstract concepts to quantified obstacles.
- Delivering a fully documented, open-source implementation with containerized environments, fostering transparency and reproducibility.
- Offering a balanced perspective that tempers optimism with rigorous assessment, supporting a more realistic roadmap for QML research.

5.3. Limitations and Future Work

While comprehensive within scope, this study has limitations that highlight promising directions for future work:

- **Scaling studies:** The current experiments were limited to small-scale problems (2–8 qubits). Larger-scale studies on more reliable quantum hardware are essential to understand scaling behavior.
- **Algorithmic innovation:** Promising future directions include layerwise training, quantum convolutional neural networks, and non-gradient-based optimization techniques that may mitigate barren plateaus.
- **Error handling:** More advanced error mitigation and correction methods, integrated directly into the training process, are needed to improve resilience.
- **Application focus:** Rather than seeking general-purpose advantage, research should prioritize domains with inherent structural benefits for quantum approaches, such as quantum chemistry, optimization, or quantum-native data analysis.
- **Standardization:** The development of community benchmarks for datasets, metrics, and reporting will accelerate progress and support meaningful comparisons across studies.

The path toward practical quantum-enhanced machine learning should be viewed as a long-term research effort rather than an immediate breakthrough. The current NISQ era is best understood as a crucial prototyping phase: a time to stress-test algorithms, explore fundamental limitations, and build robust evaluation frameworks. By combining rigorous benchmarking, realistic expectations, and cross-disciplinary collaboration, the community can lay the groundwork for the eventual realization of quantum advantage once fault-tolerant hardware becomes available. In this way, progress in QML today will form the foundation for the transformative potential of quantum technologies in the future.

Declaration Regarding the Utilization of Generative AI

The authors used Generative AI model to enhance language and readability while preparing this work. Following the utilization of this tool, the authors conducted a review and made necessary edits to the content, thereby taking full responsibility for the publication's content.

Data Availability Statement

All data, materials, and outputs generated or used in the course of this research are available from the corresponding author upon reasonable request. This includes raw datasets, processed data, analysis scripts, and any other supporting materials that contributed to the results presented in this study. Access may be granted under conditions that ensure proper use and protection of sensitive or confidential information where applicable.

Funding

This research received no external funding.

References

[1] J. Sevilla, L. Heim, A. Ho, T. Besiroglu, M. Hobbhahn, P. Villalobos, Compute trends across three eras of machine learning, in: 2022 International Joint Conference on Neural Networks (IJCNN), 2022, pp. 1–8. doi:10.1109/IJCNN55064.2022.9891914.

[2] K. Zaman, A. Marchisio, M. A. Hanif, M. Shafique, A survey on quantum machine learning: Current trends, challenges, opportunities, and the road ahead, arXiv:2310.10315arXiv:2310.10315.

[3] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, D. Amodei, Language models are few-shot learners, in: Proceedings of the 34th International Conference on Neural Information Processing Systems, NIPS '20, Curran Associates Inc., Red Hook, NY, USA, 2020, p. 25.

[4] Y. Cao, J. Romero, J. Olson, M. Degroote, P. D. Johnson, M. Kieferová, I. D. Kivlichan, T. Menke, B. Peropadre, N. P. D. Sawaya, S. Sim, L. Veis, A. Aspuru-Guzik, Quantum chemistry in the age of quantum computing., Chemical reviews.
URL https://api.semanticscholar.org/CorpusID:119417908

[5] M. A. Nielsen, I. L. Chuang, Quantum Computation and Quantum Information: 10th Anniversary Edition, 10th Edition, Cambridge University Press, USA, 2011.

[6] S. Aaronson, Read the fine print, Nature Phys 11 (2015) 291–293. doi:10.1038/nphys3272.

[7] R. Horodecki, P. Horodecki, M. Horodecki, K. Horodecki, Quantum entanglement, Rev. Mod. Phys. 81 (2009) 865–942. doi:10.1103/RevModPhys.81.865.
URL https://link.aps.org/doi/10.1103/RevModPhys.81.865

[8] P. W. Shor, Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer, SIAM Rev. 41 (1995) 303–332.
URL https://api.semanticscholar.org/CorpusID:2337707

[9] L. K. Grover, A fast quantum mechanical algorithm for database search, in: Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing, STOC '96, Association for Computing Machinery, New York, NY, USA, 1996, p. 212–219. doi:10.1145/237814.237866.
URL https://doi.org/10.1145/237814.237866

[10] A. W. Harrow, A. Montanaro, Quantum computational supremacy, Nature 549 (7671) (2017) 203–209. doi:10.1038/nature23458.

[11] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, S. Lloyd, Quantum machine learning, Nature 549 (7671) (2017) 195–202. doi:10.1038/nature23474.

[12] P. Rebentrost, M. Mohseni, S. Lloyd, Quantum support vector machine for big data classification, Phys. Rev. Lett. 113 (2014) 130503. doi:10.1103/PhysRevLett.113.130503.
URL https://link.aps.org/doi/10.1103/PhysRevLett.113.130503

[13] N. Wiebe, A. Kapoor, K. M. Svore, Quantum algorithms for nearest-neighbor methods for supervised and unsupervised learning, Quantum Info. Comput. 15 (3–4) (2015) 316–356.

[14] M. Schuld, N. Killoran, Quantum machine learning in feature hilbert spaces, Phys. Rev. Lett. 122 (2019) 040504. doi:10.1103/PhysRevLett.122.040504.
URL https://link.aps.org/doi/10.1103/PhysRevLett.122.040504

[15] A. W. Harrow, A. Hassidim, S. Lloyd, Quantum algorithm for linear systems of equations, Phys. Rev. Lett. 103 (2009) 150502. doi:10.1103/PhysRevLett.103.150502.
URL https://link.aps.org/doi/10.1103/PhysRevLett.103.150502

[16] D. Dervovic, M. Herbster, P. Mountney, S. Severini, N. Usher, L. Wossnig, Quantum linear systems algorithms: a primer, ArXiv abs/1802.08227.
URL https://api.semanticscholar.org/CorpusID:3486225

[17] J. Preskill, Quantum computing in the nisq era and beyond, Quantum 2 (2018) 79. arXiv:1801.00862, doi:10.22331/q-2018-08-06-79.

[18] K. Bharti, A. Cervera-Lierta, T. H. Kyaw, T. Haug, S. Alperin-Lea, A. Anand, M. Degroote, H. Heimonen, J. S. Kottmann, T. Menke, W.-K. Mok, S. Sim, L.-C. Kwek, A. Aspuru-Guzik, Noisy intermediate-scale quantum algorithms, Rev. Mod. Phys. 94 (2022) 015004. doi:10.1103/RevModPhys.94.015004.
URL https://link.aps.org/doi/10.1103/RevModPhys.94.015004

[19] M. Cerezo, A. Arrasmith, R. Babbush, P. J. Coles, L. Fitzgerald, J. Leyton-Ortega, A. Fefferman, M. B. Ruskai, P. J. Coles, J. Larocque, et al., Variational quantum algorithms, Nature Reviews Physics 3 (2021) 625–644. doi:10.1038/s42254-021-00348-9.

[20] E. Farhi, H. Neven, Classification with quantum neural networks on near term processors, arXiv: Quantum Physics.
URL https://api.semanticscholar.org/CorpusID:119037649

[21] M. Benedetti, E. Lloyd, S. Sack, M. Fiorentini, Parameterized quantum circuits as machine learning models, Quantum Science and Technology 4 (4) (2019) 043001. doi:10.1088/2058-9565/ab4eb5.

[22] A. Peruzzo, J. McClean, P. Shadbolt, L. Zhou, J. Love, J. L. O'brien, A. Kandala, K. Temme, L. Cincio, A. Mezzacapo, A variational eigenvalue solver on a photonic quantum processor, Nat Commun 5 (2014) 4213. doi:10.1038/ncomms5213.

[23] E. Farhi, J. Goldstone, S. Gutmann, A quantum approximate optimization algorithm, arXiv quant-ph/1411.4028 (2014) –. arXiv:1411.4028, doi:10.48550/arXiv.1411.4028.
URL https://arxiv.org/abs/1411.4028

[24] M. Cerezo, G. Verdon, H.-Y. Huang, L. Cincio, P. J. Coles, Challenges and opportunities in quantum machine learning, Nature Computational Science 2 (2022) 567 – 576.
URL https://api.semanticscholar.org/CorpusID:252323115

[25] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, H. Neven, Barren plateaus in quantum neural network training landscapes, Nat Commun 9 (1) (2018) 4812. doi:10.1038/s41467-018-07090-4.

[26] R. LaRose, B. Coyle, Robust data encodings for quantum classifiers, Phys. Rev. A 102 (2020) 032420. doi:10.1103/PhysRevA.102.032420.
URL https://link.aps.org/doi/10.1103/PhysRevA.102.032420

[27] S. Wang, E. Fontana, M. Cerezo, I. Kerenidis, P. J. Coles, L. Mross, Noise-induced barren plateaus in variational quantum algorithms, Nat Commun 12 (1) (2021) 6961. doi:10.1038/s41467-021-27045-6.

[28] M. A. Nielsen, I. L. Chuang, Quantum Computation and Quantum Information: 10th Anniversary Edition, Cambridge University Press, 2010.

[29] A. Barenco, C. H. Bennett, R. Cleve, D. P. DiVincenzo, N. Margolus, P. Shor, T. Sleator, J. A. Smolin, H. Weinfurter, Elementary gates for quantum computation, Phys. Rev. A 52 (1995) 3457–3467. doi:10.1103/PhysRevA.52.3457.
URL https://link.aps.org/doi/10.1103/PhysRevA.52.3457

[30] L. G. Valiant, A theory of the learnable, Commun. ACM 27 (11) (1984) 1134–1142. doi:10.1145/1968.1972.
URL https://doi.org/10.1145/1968.1972

[31] B. Schölkopf, A. J. Smola, Learning with kernels: support vector machines, regularization, optimization, and beyond, MIT Press, 2001. doi:10.7551/mitpress/4175.001.0001.

[32] C. Cortes, V. Vapnik, Support-vector networks, Mach Learn 20 (3) (1995) 273–297. doi:10.1007/BF00994018.

[33] A. K. Jain, M. N. Murty, P. J. Flynn, Data clustering: a review, ACM Computing Surveys 31 (3) (1999) 264–323. doi:10.1145/331499.331504.
URL https://doi.org/10.1145/331499.331504

[34] L. van der Maaten, G. E. Hinton, Visualizing data using t-sne, Journal of Machine Learning Research 9 (2008) 2579–2605.
URL https://api.semanticscholar.org/CorpusID:5855042

[35] L. Bottou, Large-scale machine learning with stochastic gradient descent, in: Proceedings of COMPSTAT'2010, Physica-Verlag HD, 2010, pp. 177–186. doi:10.1007/978-3-7908-2604-3_16.

[36] J. Sun, Q. Qu, J. Wright, Complete dictionary recovery over the sphere ii: Recovery by riemannian trust-region method, IEEE Trans. Inf. Theor. 63 (2) (2017) 885–914. doi:10.1109/TIT.2016.2632149.
URL https://doi.org/10.1109/TIT.2016.2632149

[37] S. Lloyd, M. Mohseni, P. Rebentrost, Quantum algorithms for supervised and unsupervised machine learning, arXiv preprint arXiv:1307.0411.

[38] M. H. Amin, E. Andriyash, J. Rolfe, B. Kulchytskyy, R. Melko, Quantum boltzmann machine, Phys. Rev. X 8 (2018) 021050. doi:10.1103/PhysRevX.8.021050.
URL https://link.aps.org/doi/10.1103/PhysRevX.8.021050

[39] H.-Y. Huang, M. Broughton, M. Mohseni, R. Babbush, S. Boixo, R. Cleve, H. Neven, J. R. McClean, Power of data in quantum machine learning, Nat Commun 12 (2021) 2631. doi:10.1038/s41467-021-22539-9.

[40] K. Mitarai, M. Negoro, M. Kitagawa, K. Fujii, Quantum circuit learning, Phys. Rev. A 98 (2018) 032309. doi:10.1103/PhysRevA.98.032309.
URL https://link.aps.org/doi/10.1103/PhysRevA.98.032309

[41] V. Havlíček, A. D. Córcoles, K. Temme, A. W. Harrow, A. Kandala, J. M. Chow, J. M. Gambetta, Supervised learning with quantum-enhanced feature spaces, Nature 567 (7747) (2019) 209–212. doi:10.1038/s41586-019-0980-2.
URL https://doi.org/10.1038/s41586-019-0980-2

[42] L. Zhou, S.-T. Wang, S. Choi, H. Pichler, M. D. Lukin, Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices, Phys. Rev. X 10 (2020) 021067. doi:10.1103/PhysRevX.10.021067.
URL https://link.aps.org/doi/10.1103/PhysRevX.10.021067

[43] E. Grant, L. Wossnig, M. Ostaszewski, M. Benedetti, An initialization strategy for addressing barren plateaus in parametrized quantum circuits, Quantum 3 (2019) 214. arXiv:1903.05076, doi:10.22331/q-2019-12-09-214.

[44] M. Schuld, F. Petruccione, Machine Learning with Quantum Computers, 2nd Edition, Quantum Science and Technology, Springer Cham, 2021. doi:10.1007/978-3-030-83098-4.

[45] Z. Holmes, K. Sharma, M. Cerezo, P. J. Coles, Connecting ansatz expressibility to gradient magnitudes and barren plateaus, PRX Quantum 3 (2022) 010313. doi:10.1103/PRXQuantum.3.010313.
URL https://link.aps.org/doi/10.1103/PRXQuantum.3.010313

[46] S. Bravyi, D. Gosset, R. Movassagh, Classical algorithms for quantum mean values, Nat. Phys. 17 (2021) 337–341. doi:10.1038/s41567-020-01109-8.

[47] H.-Y. Huang, M. Broughton, J. Cotler, S. Chen, J. Li, M. Mohseni, H. Neven, R. Babbush, R. Kueng, J. Preskill, J. R. McClean, Quantum advantage in learning from experiments, Science 376 (6598) (2022) 1182–1186. arXiv:https://www.science.org/doi/pdf/10.1126/science.abn7293, doi:10.1126/science.abn7293.
URL https://www.science.org/doi/abs/10.1126/science.abn7293

[48] Z. Cai, R. Babbush, S. C. Benjamin, S. Endo, W. J. Huggins, Y. Li, J. R. McClean, T. E. O'Brien, Quantum error mitigation, Rev. Mod. Phys. 95 (2023) 045005. doi:10.1103/RevModPhys.95.045005.
URL https://link.aps.org/doi/10.1103/RevModPhys.95.045005

[49] S. Sim, P. D. Johnson, A. Aspuru-Guzik, Expressibility and entangling capability of parameterized quantum circuits for hybrid quantum-classical algorithms, Advanced Quantum Technologies 2 (10) (2019) 1900070. doi:10.1002/qute.201900070.

[50] K. Temme, S. Bravyi, J. M. Gambetta, Error mitigation for short-depth quantum circuits, Phys. Rev. Lett. 119 (2017) 180509. doi:10.1103/PhysRevLett.119.180509.
URL https://link.aps.org/doi/10.1103/PhysRevLett.119.180509

[51] A. Kandala, K. Temme, A. D. Córcoles, A. Mezzacapo, J. M. Chow, J. M. Gambetta, Error mitigation extends the computational reach of a noisy quantum processor, Nature 567 (2019) 491–495. doi:10.1038/s41586-019-1040-7.

[52] A. Mari, T. R. Bromley, N. Killoran, Estimating the gradient and higher-order derivatives on quantum hardware, Phys. Rev. A 103 (2021) 012405. doi:10.1103/PhysRevA.103.012405.
URL https://link.aps.org/doi/10.1103/PhysRevA.103.012405

[53] J. Stokes, J. Izaac, N. Killoran, G. Carleo, Quantum natural gradient, Quantum 4 (2020) 269. doi:10.22331/q-2020-05-25-269.

[54] M. Schuld, Supervised quantum machine learning models are kernel methods, arXiv preprint arXiv:2101.11020.

[55] M. X. Goemans, D. P. Williamson, Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming, Journal of the ACM 42 (6) (1995) 1115–1145. doi:10.1145/227683.227684.

[56] F. G. Brandao, M. Broughton, E. Farhi, S. Gutmann, H. Neven, For fixed control parameters the quantum approximate optimization algorithm's objective function value concentrates for typical instances, arXiv preprint arXiv:1812.04170.

[57] M. Henderson, S. Shakya, S. Pradhan, T. Cook, Quanvolutional neural networks: powering image recognition with quantum circuits, Quantum Mach. Intell. 2 (1) (2020) 2. doi:10.1007/s42484-020-00012-y.