

Impacts and Potentials of Quantum Computing for Artificial Intelligence: A Comprehensive Review

Yet Another AI Scientist^{1*}

^{1*}AI Research Division, Generated by AI and should not be trusted,
Nowhere, Earth.

Corresponding author(s). E-mail(s): chenc273@outlook.com;

Abstract

The intersection of quantum computing and artificial intelligence represents one of the most anticipated technological convergences of the twenty-first century. This comprehensive review examines the impacts and potentials of quantum computing for artificial intelligence, synthesizing developments from 2020 to 2025 across seven major themes: variational quantum machine learning, quantum kernel methods, quantum optimization, barren plateaus and trainability, quantum advantage and speedup, NISQ-era constraints and error mitigation, and practical applications. We present a central thesis that quantum computing offers genuine but constrained potential to enhance artificial intelligence, characterized by three defining features: a fundamental tension between trainability and quantum advantage, a shift from universal speedup claims to domain-specific advantages, and a practical trajectory dominated by hybrid quantum-classical approaches on near-term hardware.

Our analysis reveals that the field has matured significantly from early optimistic claims of universal quantum speedup. Rigorous theoretical work has established both the existence of provable exponential speedups for specific problems and the fundamental limits constraining such advantages. The trainability-simulability tradeoff, established by Cerezo and colleagues, demonstrates that quantum circuits avoiding barren plateaus may be efficiently classically simulable, posing a fundamental challenge to variational quantum algorithm approaches. Quantum advantage appears robust for learning from quantum data, as demonstrated by Huang et al., while advantage for classical data remains elusive and may require fault-tolerant quantum computing to materialize.

The practical landscape is defined by noisy intermediate-scale quantum (NISQ) hardware with 50-1000+ qubits and error rates of 0.1-1% per two-qubit gate.

Error mitigation techniques including zero-noise extrapolation, probabilistic error cancellation, and learning-based methods extend the practical reach of quantum algorithms but face fundamental exponential overhead bounds. Hybrid quantum-classical architectures, where quantum components serve as specialized co-processors within classical pipelines, represent the dominant paradigm for near-term implementations.

We identify quantum chemistry and drug discovery as the most promising near-term application domains due to the inherent quantum nature of molecular systems. No production quantum AI application exists as of 2025, but integration into pharmaceutical discovery pipelines may emerge within 3-5 years. Optimization applications in finance and logistics show competitive performance on small instances without demonstrating clear advantage over classical methods. For general-purpose AI tasks such as computer vision and natural language processing, quantum approaches do not currently compete with classical methods.

This review provides researchers, practitioners, and funding agencies with a rigorous framework for evaluating quantum AI research and investment decisions, emphasizing the importance of fair benchmarking, end-to-end cost analysis, and realistic timeline projections for the transition to fault-tolerant quantum computing.

Keywords: quantum computing, quantum machine learning, variational quantum circuits, quantum kernels, barren plateaus, NISQ algorithms, quantum advantage, error mitigation

1 Introduction

The convergence of quantum computing and artificial intelligence represents one of the most anticipated technological developments of the twenty-first century. Quantum computers exploit the principles of quantum mechanics—superposition, entanglement, and interference—to perform certain computations with fundamentally different resources than classical computers [1]. Artificial intelligence, particularly machine learning, has transformed computing over the past decade through algorithms that learn from data to make predictions, generate content, and optimize decisions [2]. The natural question arises: can quantum computers accelerate or enhance artificial intelligence capabilities?

This question has generated intense research activity, substantial private investment, and considerable hype. Early theoretical results suggested that quantum computers could exponentially accelerate machine learning subroutines such as linear algebra operations, principal component analysis, and clustering [3, 4]. The prospect of exponential speedups for computationally intensive AI workloads drove optimistic projections about quantum machine learning revolutionizing industries from drug discovery to finance [2]. However, the reality has proven more nuanced, and the field has matured from broad optimism to careful analysis of specific advantage conditions.

The period from 2020 to 2025 has been transformative for quantum machine learning research. Several developments have fundamentally reshaped understanding of when and how quantum computing can benefit AI. First, the phenomenon of barren plateaus—exponentially vanishing gradients in quantum circuit training—was recognized as a central obstacle to scaling variational quantum algorithms [5, 6]. Second, “dequantization” results demonstrated that many presumed quantum speedups could be matched by classical algorithms with similar data access assumptions [7]. Third, rigorous proofs established both the existence and conditions for genuine quantum advantage in machine learning [8, 9]. Fourth, the unifying perspective that all supervised quantum machine learning models are mathematically equivalent to kernel methods provided theoretical clarity [10].

These developments motivate a comprehensive review that synthesizes the current state of quantum computing for AI. Unlike earlier reviews that surveyed algorithms and applications [2], this work emphasizes the fundamental constraints, tradeoffs, and conditions that determine when quantum approaches can provide genuine benefit. We adopt a critical perspective that acknowledges both the real potential and significant limitations of current and near-term quantum machine learning.

1.1 Scope and Focus

This review focuses specifically on how quantum computing can enhance artificial intelligence—the direction from quantum computing to AI rather than the reverse direction of using AI to improve quantum computing (which represents an active but distinct research area). We consider quantum machine learning in the broad sense: any approach that uses quantum computers to perform or accelerate machine learning tasks, whether through variational circuits, kernel methods, optimization algorithms, or hybrid architectures.

Our scope encompasses gate-based quantum computing and, where relevant, quantum annealing. We emphasize the noisy intermediate-scale quantum (NISQ) era that characterizes current hardware, while also considering the transition to fault-tolerant quantum computing projected for the late 2020s to early 2030s. We focus on developments from 2020 to 2025, the period during which the field achieved significant theoretical and practical maturation.

Several topics fall outside our scope. We do not comprehensively cover quantum simulation of physical systems (a distinct application of quantum computing), quantum cryptography, or quantum communication. We also do not attempt to survey all proposed quantum machine learning algorithms; rather, we focus on the paradigms and results that shape understanding of quantum advantage conditions.

1.2 Central Thesis

We advance a central thesis that guides this review:

Quantum computing offers genuine but constrained potential to enhance artificial intelligence, with the path forward characterized by: (1) a fundamental tension between trainability and quantum advantage; (2) a shift from universal speedup claims to domain-specific advantages; and (3)

a practical trajectory dominated by hybrid quantum-classical approaches on near-term hardware.

This thesis emerges from synthesizing developments across seven major themes that structure the review. The trainability-advantage tension reflects the theoretical discovery that circuits avoiding barren plateaus may be classically simulable [11]. The shift to domain-specific advantages reflects the dequantization of many general speedup claims and the robustness of advantage for quantum data [7, 9]. The hybrid trajectory reflects the constraints of NISQ hardware and the success of variational algorithms that combine quantum circuits with classical optimization [12].

1.3 Organization of This Review

The review is organized into seven thematic sections following this introduction and background:

Variational Quantum Machine Learning (Section 3) examines parameterized quantum circuits as trainable function approximators, covering data encoding, circuit architectures, and training methods. This paradigm dominates near-term quantum machine learning implementations.

Quantum Kernel Methods (Section 4) presents the theoretical framework showing that all supervised quantum models are kernel methods, with implications for understanding advantage conditions and expressivity limits.

Quantum Optimization for AI (Section 5) covers the Quantum Approximate Optimization Algorithm (QAOA) and quantum annealing approaches to combinatorial optimization problems arising in AI applications.

Barren Plateaus and Trainability (Section 6) analyzes the central challenge of exponentially vanishing gradients, their multiple causes, and strategies for mitigation, including the fundamental trainability-simulability tradeoff.

Quantum Advantage and Speedup (Section 7) examines the evolution from early optimism through dequantization to rigorous advantage results, clarifying when quantum computers can genuinely outperform classical methods.

NISQ Era and Error Mitigation (Section 8) describes the hardware constraints that define practical quantum machine learning and the error mitigation techniques that extend useful quantum computation.

Applications and Future Directions (Section 9) surveys current application domains including quantum chemistry, drug discovery, optimization, and finance, with projections for near-term and long-term development.

The review concludes with a Discussion synthesizing insights across themes, an Outlook projecting future developments, and Conclusions summarizing key takeaways for researchers, practitioners, and funding agencies.

1.4 Key Developments Surveyed

Figure 1 presents a timeline of seminal developments in quantum machine learning from 2020 to 2025. Several milestones merit particular emphasis:

2020: Zhou et al. provided comprehensive analysis of QAOA performance, mechanism, and implementation, establishing understanding of this central algorithm [13]. Chen et al. demonstrated variational quantum circuits for reinforcement learning [14].

2021: This year saw multiple foundational results. Cerezo et al. proved that cost function locality determines barren plateau behavior even in shallow circuits [6]. Wang et al. discovered noise-induced barren plateaus as a distinct phenomenon [15]. Huang et al. established the “power of data” framework showing advantage depends on data structure [16]. Liu et al. proved the first rigorous exponential speedup in supervised machine learning [8]. Schuld unified supervised quantum ML as kernel methods [10]. Saggio et al. provided experimental demonstration of quantum speedup in reinforcement learning [17].

Seminal Developments in Quantum Machine Learning (2020-2025)

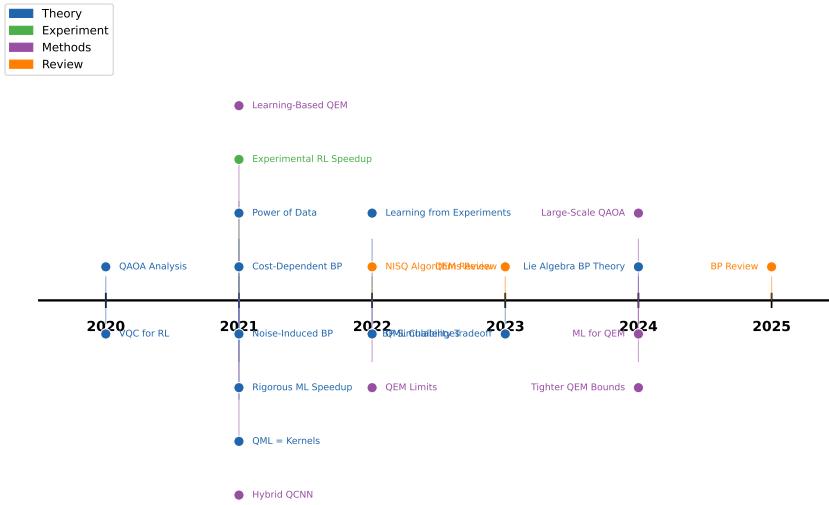


Fig. 1 Timeline of seminal developments in quantum machine learning (2020–2025). Key milestones include foundational theoretical results establishing trainability constraints and advantage conditions (blue), experimental demonstrations (green), methodological advances in error mitigation and hybrid architectures (purple), and comprehensive reviews consolidating the field (orange). The period shows evolution from initial algorithm proposals through critical theoretical analysis to mature understanding of quantum advantage conditions.

2022: Bharti et al. and Cerezo et al. published comprehensive reviews of NISQ algorithms and quantum ML challenges respectively [12, 18]. Huang et al. proved quantum advantage for learning from quantum experiments [9]. Takagi et al. established fundamental exponential overhead bounds for error mitigation [19].

2023: Cai et al. published the definitive review of quantum error mitigation [20]. Cerezo et al. revealed the trainability-simulability tradeoff [11].

2024-2025: Ragone et al. developed the Lie algebraic theory unifying barren plateau understanding [21]. Sack and Egger demonstrated 100+ qubit QAOA with machine learning error mitigation [22]. Larocca et al. published the comprehensive barren plateau review [23].

1.5 The Quantum Machine Learning Landscape

Before proceeding to detailed analysis, we briefly situate quantum machine learning within the broader context of both quantum computing and artificial intelligence.

Quantum Computing Context: Quantum machine learning is one of several proposed applications of quantum computers. Quantum simulation of physical systems—originally Feynman’s motivation for quantum computing—remains perhaps the most natural application [24]. Quantum cryptography and communication exploit quantum mechanical properties for security. Optimization and search algorithms offer potential speedups for NP-hard problems. Quantum machine learning intersects with these areas, particularly quantum simulation for chemistry and optimization for combinatorial AI problems.

Artificial Intelligence Context: Modern AI is dominated by deep learning approaches that train large neural networks on massive datasets using gradient descent optimization on specialized hardware (GPUs, TPUs). The scale of contemporary AI systems—billions of parameters trained on trillions of tokens—far exceeds current quantum computing capabilities. Quantum machine learning must ultimately compete with or complement these classical approaches, which themselves continue advancing rapidly [18].

The Competition Question: A recurring theme throughout this review is the competition between quantum and classical approaches. Early quantum speedup claims were often made against weak classical baselines or under unrealistic assumptions about data access. As the field has matured, comparison against state-of-the-art classical methods with fair resource accounting has become essential. This competition is not static: classical machine learning advances alongside quantum, and dequantization results show that quantum-inspired classical algorithms can sometimes match presumed quantum speedups.

This landscape motivates careful analysis of specific advantage conditions rather than broad claims about quantum acceleration of AI. The following sections provide that detailed analysis, organized around the seven themes that capture the essential structure of the field.

2 Background and Fundamentals

This section provides the foundational concepts necessary for understanding quantum machine learning, targeting readers with familiarity in either quantum computing or machine learning but not necessarily both. We establish the essential quantum computing principles, review relevant machine learning fundamentals, introduce the core concepts of quantum machine learning, describe the NISQ hardware context, and present a taxonomy of quantum AI approaches.

2.1 Quantum Computing Fundamentals

2.1.1 Qubits and Quantum States

The fundamental unit of quantum information is the qubit, a two-level quantum system that exists in superposition of basis states. Unlike a classical bit that takes value 0 or 1, a qubit occupies a quantum state

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \quad (1)$$

where α and β are complex amplitudes satisfying $|\alpha|^2 + |\beta|^2 = 1$. The squared amplitudes represent probabilities of measuring the respective basis states. Geometrically, a single qubit state maps to a point on the Bloch sphere, with the poles representing $|0\rangle$ and $|1\rangle$ and superposition states occupying the surface [1].

For n qubits, the state space grows exponentially: a general state is a superposition over 2^n computational basis states

$$|\psi\rangle = \sum_{x \in \{0,1\}^n} \alpha_x |x\rangle \quad (2)$$

with $\sum_x |\alpha_x|^2 = 1$. This exponential state space is central to quantum computing's potential power—and its challenges for classical simulation.

2.1.2 Quantum Entanglement

Entanglement describes quantum correlations between qubits that have no classical analog. A two-qubit state is entangled if it cannot be written as a product of single-qubit states. The canonical example is the Bell state

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \quad (3)$$

where measuring one qubit instantaneously determines the other's outcome regardless of separation. Entanglement is essential for quantum computational advantage—circuits without entanglement can be efficiently simulated classically [12].

2.1.3 Quantum Gates and Circuits

Quantum computation proceeds through unitary operations (quantum gates) applied to qubit states. Common single-qubit gates include:

- Pauli gates X, Y, Z : rotations by π about respective Bloch sphere axes
- Hadamard H : creates superposition, mapping $|0\rangle \mapsto (|0\rangle + |1\rangle)/\sqrt{2}$
- Rotation gates $R_X(\theta), R_Y(\theta), R_Z(\theta)$: parameterized rotations by angle θ

Two-qubit entangling gates include the CNOT (controlled-NOT) that flips the target qubit conditioned on the control, and the CZ (controlled-Z) applying a phase

conditioned on both qubits being $|1\rangle$. Universal quantum computation can be achieved with single-qubit rotations plus any entangling two-qubit gate.

A quantum circuit is a sequence of gates applied to an initial state (typically $|0\rangle^{\otimes n}$), followed by measurement. Circuit depth counts the sequential gate layers; circuit width counts the number of qubits. These parameters determine computational requirements and noise sensitivity.

2.1.4 Measurement

Quantum measurement extracts classical information from quantum states but destroys superposition. Measuring a qubit in the computational basis projects it to $|0\rangle$ or $|1\rangle$ with probabilities $|\alpha|^2$ and $|\beta|^2$ respectively. After measurement, the qubit is in the measured state—the original superposition is irreversibly lost.

This measurement limitation has profound implications for quantum machine learning: extracting classical output from quantum computations requires measurement, but measurement destroys quantum information. Clever algorithm design navigates this constraint by computing useful quantities as expectation values of measurement operators.

2.1.5 Computational Complexity

Quantum computers define the complexity class BQP (bounded-error quantum polynomial time): problems solvable by quantum circuits with polynomial depth and bounded error probability. The relationship between BQP and classical complexity classes (P, NP, PSPACE) remains open, but BQP is believed to contain problems intractable for classical computers.

Shor’s algorithm for integer factorization and Grover’s algorithm for unstructured search provide canonical examples of quantum speedup. Shor’s algorithm achieves exponential speedup over known classical methods for factoring, while Grover’s provides quadratic speedup for searching unstructured databases. These results establish

that quantum computers can be faster than classical for specific problems, motivating the search for quantum speedups in machine learning.

2.2 Machine Learning Fundamentals

2.2.1 Learning Paradigms

Machine learning encompasses algorithms that improve performance on tasks through experience (data). Three primary paradigms structure the field:

Supervised learning: Given labeled training data $\{(x_i, y_i)\}$, learn a function f that predicts labels for new inputs. Classification predicts discrete labels; regression predicts continuous values. The goal is minimizing expected loss on unseen test data (generalization).

Unsupervised learning: Given unlabeled data $\{x_i\}$, discover structure such as clusters, principal components, or generative distributions. Applications include dimensionality reduction, anomaly detection, and density estimation.

Reinforcement learning: An agent learns to maximize cumulative reward through interaction with an environment. The agent observes states, takes actions, and receives rewards, learning a policy mapping states to actions. Applications include game playing, robotics, and optimization.

2.2.2 Neural Networks

Modern machine learning is dominated by neural networks—parameterized function approximators composed of layers of linear transformations and nonlinear activations.

A feedforward network computes

$$f(x; \theta) = \sigma_L(W_L \cdots \sigma_1(W_1 x + b_1) \cdots + b_L) \quad (4)$$

where W_i , b_i are trainable weights and biases, σ_i are nonlinear activation functions, and θ collectively denotes all parameters.

Deep networks with many layers can represent highly complex functions. Training optimizes parameters via gradient descent on a loss function

$$\theta_{t+1} = \theta_t - \eta \nabla_{\theta} \mathcal{L}(\theta_t) \quad (5)$$

where η is the learning rate. Backpropagation computes gradients efficiently through the chain rule.

Specialized architectures include convolutional neural networks (CNNs) for images, recurrent networks for sequences, and transformer networks that dominate natural language processing. The scale of modern networks—billions of parameters trained on massive datasets using specialized hardware—presents both the opportunity and challenge for quantum approaches.

2.2.3 Kernel Methods

Kernel methods provide an alternative to neural networks grounded in statistical learning theory. A kernel $k(x, x')$ measures similarity between inputs, implicitly computing inner products in a high-dimensional feature space. The kernel support vector machine finds the maximum-margin hyperplane separating classes:

$$f(x) = \sum_i \alpha_i y_i k(x_i, x) + b \quad (6)$$

where the sum is over support vectors (training points on the margin). Kernel methods provide theoretical guarantees and work well with limited data, making them attractive for quantum approaches with limited circuit sizes.

2.3 Quantum Machine Learning Overview

Quantum machine learning encompasses approaches that use quantum computers to perform or accelerate machine learning tasks. The field subdivides into several paradigms [2, 18, 25]:

2.3.1 Variational Quantum Algorithms

Variational quantum algorithms (VQAs) use parameterized quantum circuits optimized by classical computers. A variational quantum circuit (VQC) implements a unitary $U(\theta)$ depending on trainable parameters θ . The circuit output is measured to compute expectation values $\langle O \rangle = \langle 0 | U^\dagger(\theta) O U(\theta) | 0 \rangle$ that define a cost function optimized classically.

This hybrid quantum-classical approach is well-suited to NISQ hardware: quantum circuits perform operations intractable classically, while classical optimization handles gradient computation and parameter updates. The Variational Quantum Eigensolver (VQE) for chemistry and QAOA for optimization pioneered this paradigm [26, 27].

2.3.2 Data Encoding

A fundamental challenge in quantum ML is encoding classical data into quantum states. Common strategies include:

Angle encoding: Features x_i become rotation angles: $|x\rangle = \bigotimes_i R_Y(x_i)|0\rangle$. Simple but uses one qubit per feature.

Amplitude encoding: $N = 2^n$ features encoded in n qubit amplitudes: $|x\rangle = \sum_i x_i|i\rangle$. Exponentially compact but requires complex state preparation.

Feature maps: Parameterized circuits $U_\phi(x)$ create nonlinear embeddings. The feature map determines model expressivity and potential advantage [28].

2.3.3 Quantum Kernels

Quantum circuits naturally define kernels through state overlap:

$$k(x, x') = |\langle \phi(x) | \phi(x') \rangle|^2 = |\langle 0 | U^\dagger(x) U(x') | 0 \rangle|^2 \quad (7)$$

where $U(x)$ encodes data x into quantum state $|\phi(x)\rangle$. This kernel can be estimated by quantum circuit execution and used with classical kernel methods (SVMs). Schuld proved that all supervised quantum ML models are mathematically equivalent to kernel methods, providing a unifying theoretical framework [10].

2.3.4 Quantum Neural Networks

The term “quantum neural network” (QNN) refers to various architectures including:

- Parameterized circuits treated as trainable layers
- Quantum versions of specific classical architectures (quantum CNNs, quantum autoencoders)
- Hybrid networks with quantum layers embedded in classical networks

Despite the naming, QNNs differ fundamentally from classical neural networks: they operate in exponentially large Hilbert space, are constrained by unitarity, and face distinct trainability challenges (barren plateaus) [18].

2.4 The NISQ Hardware Context

Current quantum computers operate in the noisy intermediate-scale quantum (NISQ) era, a term coined by Preskill to describe devices with sufficient qubits to potentially exceed classical simulation but too noisy for fault-tolerant computation [1].

2.4.1 Hardware Platforms

Multiple physical platforms implement quantum computing with different characteristics:

Superconducting qubits: Leading platform (IBM, Google) using Josephson junctions cooled to millikelvin temperatures. Advantages: fast gates (10-100 ns), scalable fabrication. Disadvantages: short coherence times (50-200 μ s), limited connectivity.

Trapped ions: Atoms confined by electromagnetic fields (IonQ, Quantinuum). Advantages: long coherence times (seconds to minutes), high-fidelity gates, all-to-all connectivity. Disadvantages: slower gates (μ s scale), scaling challenges.

Neutral atoms: Arrays of optically trapped atoms (QuEra, Pasqal). Advantages: large qubit counts, reconfigurable connectivity. Disadvantages: lower gate fidelities, slower operation.

Photonics: Photons as qubits (Xanadu). Advantages: room temperature operation, natural for communication. Disadvantages: probabilistic gates, loss.

2.4.2 Error Sources and Rates

NISQ devices face multiple error sources:

- **Gate errors:** Imperfect implementation of quantum gates, typically 0.1-1% for two-qubit gates on leading platforms
- **Measurement errors:** Incorrect readout, typically 1-5%
- **Decoherence:** Loss of quantum information to environment, characterized by T1 (energy relaxation) and T2 (dephasing) times
- **Crosstalk:** Unwanted interactions between qubits during operations

These errors accumulate with circuit depth, fundamentally limiting useful computation. The practical circuit depth on NISQ devices ranges from tens to hundreds of gates depending on error rates and error mitigation.

2.4.3 Implications for Machine Learning

NISQ constraints profoundly shape quantum ML algorithm design:

- Circuits must be shallow (limited depth) to avoid noise accumulation
- Algorithms must be hybrid, offloading optimization to classical computers
- Error mitigation techniques are essential to extract useful results
- Problem size is limited by available qubits and connectivity

These constraints motivate the variational paradigm: use quantum circuits for computations intractable classically while keeping circuits shallow and relying on classical optimization for training.

2.5 Taxonomy of Quantum AI Approaches

Figure 2 presents a taxonomy organizing quantum AI approaches into four major categories, with connections indicating methodological relationships.

2.5.1 Variational Algorithms

This category encompasses parameterized quantum circuits optimized by classical algorithms:

- **Variational Quantum Circuits (VQC)**: General parameterized circuits for classification and regression
- **Variational Quantum Eigensolver (VQE)**: Chemistry-focused algorithm for ground state estimation
- **Quantum Approximate Optimization Algorithm (QAOA)**: Combinatorial optimization through alternating operators

Variational algorithms dominate NISQ-era quantum ML due to their flexibility and noise tolerance.

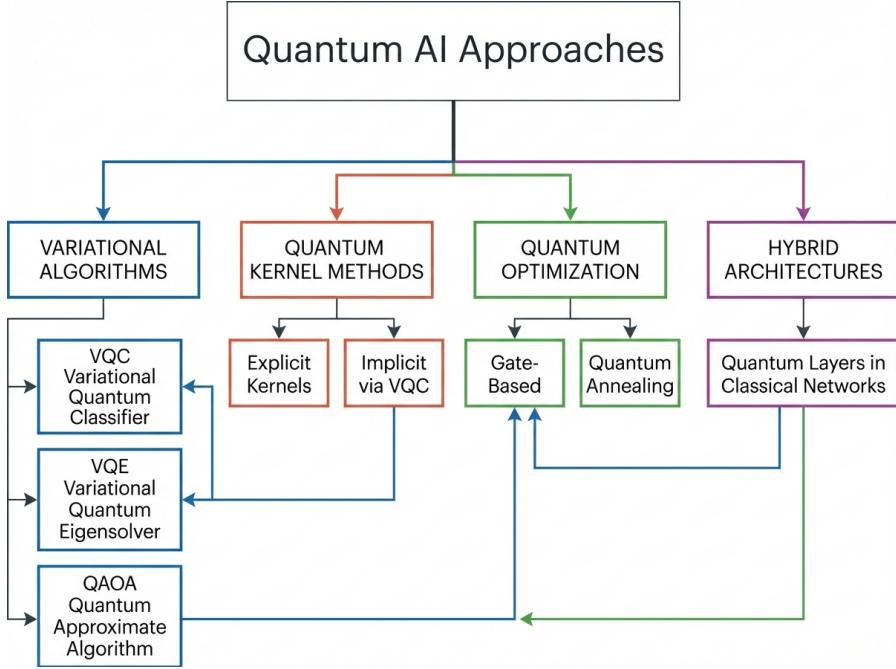


Fig. 2 Taxonomy of quantum computing approaches for artificial intelligence. The field encompasses variational algorithms (parameterized quantum circuits trained via classical optimization), quantum kernel methods (leveraging quantum feature spaces for classical learning), quantum optimization (QAOA and quantum annealing), and hybrid architectures (quantum components within classical neural networks). Arrows indicate methodological connections; notably, the kernel perspective unifies all supervised variational methods [10].

2.5.2 Quantum Kernel Methods

Quantum kernels provide the most rigorous theoretical framework:

- **Explicit quantum kernels:** Directly estimate kernel matrix via quantum circuit execution
- **Implicit kernels via VQC:** Variational classifiers viewed through kernel perspective

The kernel viewpoint clarifies advantage conditions: quantum must implement kernels intractable for classical computation.

2.5.3 Quantum Optimization

Quantum approaches to combinatorial optimization include:

- **Gate-based optimization:** QAOA and variants on universal quantum computers
- **Quantum annealing:** Adiabatic evolution on specialized hardware (D-Wave)

Optimization connects to ML through hyperparameter tuning, feature selection, and combinatorial problems arising in AI applications.

2.5.4 Hybrid Architectures

Hybrid quantum-classical approaches integrate quantum components into classical pipelines:

- **Quantum layers in neural networks:** Replace specific layers with quantum circuits
- **Quantum feature extraction:** Use quantum circuits as preprocessing
- **Quantum-classical pipelines:** Classical preprocessing, quantum computation, classical post-processing

Hybrid approaches acknowledge that quantum processors serve as co-processors enhancing classical systems rather than replacing them entirely.

2.6 Summary

This section established the foundations for understanding quantum machine learning: quantum computing principles including qubits, gates, and measurement; machine learning fundamentals including neural networks and kernel methods; the NISQ hardware context defining practical constraints; and a taxonomy organizing the diverse approaches in the field. With these foundations in place, subsequent sections examine each major theme in depth.

3 Variational Quantum Machine Learning

Variational quantum machine learning (VQML) represents the dominant paradigm for implementing machine learning algorithms on near-term quantum computers. This approach combines parameterized quantum circuits (PQCs), also known as variational quantum circuits (VQCs), with classical optimization to create hybrid quantum-classical learning systems. The fundamental architecture consists of three components: a quantum feature map that encodes classical data into quantum states, a trainable quantum circuit with adjustable gate parameters, and measurements whose expectation values form the model output [12, 18].

The appeal of variational approaches lies in their flexibility and hardware compatibility. Unlike fault-tolerant quantum algorithms that require error-corrected qubits, VQCs can operate on current noisy intermediate-scale quantum (NISQ) devices with tens to hundreds of physical qubits. The quantum circuit serves as a parameterized function approximator analogous to classical neural networks, with rotation angles serving as trainable weights. Training proceeds via gradient descent or gradient-free optimization, where gradients can be computed using the parameter-shift rule that exploits properties of quantum gates [29, 30].

This section examines the theoretical foundations, circuit architectures, training methods, and current performance assessment of variational quantum machine learning.

3.1 Theoretical Foundations

3.1.1 Parameterized Quantum Circuits as Function Approximators

A variational quantum circuit implements a parameterized unitary transformation $U(\theta)$ on an input quantum state, where $\theta = (\theta_1, \dots, \theta_m)$ denotes the trainable

parameters. For a classification or regression task, the model computes

$$f(x; \theta) = \langle 0 |^{\otimes n} U_\phi^\dagger(x) U^\dagger(\theta) O U(\theta) U_\phi(x) | 0 \rangle^{\otimes n} \quad (8)$$

where $U_\phi(x)$ encodes the input data x and O is a measurement observable. The circuit output is an expectation value that depends on both input data and trainable parameters.

The theoretical foundation for VQCs as function approximators draws from quantum computing theory and approximation theory. Perez-Salinas et al. demonstrated that circuits using data re-uploading—interleaving data encoding with trainable layers—can approximate any continuous function to arbitrary precision given sufficient depth [31]. This universal approximation property parallels results for classical neural networks and establishes VQCs as theoretically expressive function classes.

The expressivity of VQCs can be characterized through their effective dimension, which measures the number of independent functions a model can represent [32]. Studies show that expressivity grows with circuit depth and entanglement, but this growth also correlates with trainability challenges. Deep circuits with high expressivity tend to exhibit barren plateaus, the exponentially vanishing gradients that prevent effective training (see Section 6).

3.1.2 Connection to Kernel Methods

A foundational theoretical insight, established by Schuld in 2021, shows that all supervised VQC models are mathematically equivalent to kernel methods [10]. The quantum model output can be expressed as

$$f(x) = \sum_i \alpha_i k(x_i, x) \quad (9)$$

where $k(x, x') = |\langle \phi(x) | \phi(x') \rangle|^2$ is the quantum kernel defined by the feature map, and α_i are coefficients determined by training.

This kernel perspective has profound implications. First, it provides theoretical tools from kernel learning theory to analyze VQC generalization and expressivity. Second, it clarifies that quantum advantage must come from the kernel being classically intractable—if the kernel can be efficiently computed or approximated classically, the VQC provides no computational benefit. Third, it reveals that the choice of data encoding (feature map) fundamentally determines model capabilities, independent of the variational ansatz structure.

3.1.3 Power of Data Framework

Huang et al. developed the “power of data” framework that analyzes when quantum models can outperform classical alternatives [16]. Their analysis shows that quantum and classical models have distinct inductive biases—preference for certain types of functions. Quantum advantage emerges when the target function aligns better with quantum model biases than classical model biases.

Critically, this framework demonstrates that quantum advantage depends on data structure, not just algorithmic capability. For data without inherent “quantum structure,” classical machine learning can match or exceed quantum model performance. This result tempered early claims of universal quantum advantage and redirected research toward identifying data distributions and problems with favorable quantum characteristics.

3.2 Data Encoding Strategies

The encoding of classical data into quantum states is a fundamental challenge with significant implications for model expressivity and potential advantage. Several encoding strategies have been developed, each with distinct characteristics [25, 28].

3.2.1 Angle Encoding

Angle encoding maps each classical feature to a rotation angle of a single qubit:

$$|x\rangle = \bigotimes_{i=1}^d R_Y(x_i)|0\rangle \quad (10)$$

where $x = (x_1, \dots, x_d)$ is the input vector and $R_Y(\theta) = e^{-i\theta Y/2}$ is a Pauli-Y rotation.

This encoding uses one qubit per feature and creates product (unentangled) states.

Advantages include simplicity and direct implementation on hardware. The primary limitation is that product states cannot exploit entanglement, and linear features per qubit restrict expressivity. Variations use multiple rotation axes (e.g., $R_X R_Y R_Z$) per qubit to increase feature richness.

3.2.2 Amplitude Encoding

Amplitude encoding stores $N = 2^n$ classical features as amplitudes of an n -qubit state:

$$|x\rangle = \frac{1}{\|x\|} \sum_{i=0}^{N-1} x_i |i\rangle \quad (11)$$

This exponentially compact representation encodes high-dimensional data in few qubits, making it attractive for large-scale problems.

However, amplitude encoding requires complex state preparation circuits that may negate efficiency gains. General amplitude encoding has circuit depth $O(N)$, eliminating the exponential compression benefit. Efficient amplitude encoding exists only for specific data structures, and the data loading bottleneck remains a fundamental challenge for quantum speedup claims [7].

3.2.3 Quantum Feature Maps

Quantum feature maps $U_\phi(x)$ are parameterized circuits that create nonlinear embeddings of data into quantum states. The ZZ feature map introduced by Havlicek et al. exemplifies this approach [28]:

$$U_\phi(x) = \exp\left(i \sum_{i < j} \phi_{ij}(x) Z_i Z_j\right) \prod_i \exp(i\phi_i(x) Z_i) \quad (12)$$

where $\phi_i(x)$ and $\phi_{ij}(x)$ are functions of the input data.

Feature maps determine the quantum kernel $k(x, x') = |\langle 0 | U_\phi^\dagger(x) U_\phi(x') | 0 \rangle|^2$. Well-designed feature maps create kernels that capture complex data relationships while remaining intractable for classical computation. However, many feature maps suffer from exponential concentration—kernel values concentrate around 2^{-n} for random inputs—limiting discriminative power at scale [33].

3.3 Circuit Architectures

The structure of variational circuits significantly impacts expressivity, trainability, and hardware compatibility. Research has explored numerous architectures, from general-purpose designs to problem-specific constructions.

3.3.1 Hardware-Efficient Ansatzes

Hardware-efficient ansatzes (HEAs) are circuits tailored to the native gate sets and qubit connectivity of specific quantum devices [34]. A typical HEA consists of alternating layers of single-qubit rotations and two-qubit entangling gates:

$$U(\theta) = \prod_{l=1}^L \left[\prod_{\langle i,j \rangle} \text{CZ}_{ij} \prod_i R_Y(\theta_{l,i}^Y) R_Z(\theta_{l,i}^Z) \right] \quad (13)$$

where the two-qubit gates respect hardware connectivity constraints.

HEAs minimize circuit depth by using native operations, maximizing success probability on noisy devices. However, their generic structure provides no guarantee of expressivity for specific problems and may exhibit barren plateaus when deep. The tradeoff between hardware efficiency and problem-specific structure remains an active research topic.

3.3.2 Problem-Inspired Ansatzes

Problem-inspired ansatzes incorporate domain knowledge into circuit structure. For chemistry applications, the Unitary Coupled Cluster (UCC) ansatz constructs circuits from chemically meaningful excitation operators [26]. For optimization, QAOA uses problem-specific cost Hamiltonians to define circuit layers [27].

These ansatzes have several advantages: they respect problem symmetries, avoid redundant parameters, and may avoid barren plateaus by constraining the circuit's dynamical Lie algebra [35]. The disadvantage is that problem-specific design requires domain expertise and may not transfer across applications.

3.3.3 Data Re-Uploading

The data re-uploading strategy interleaves data encoding with trainable layers throughout the circuit [31]:

$$U(x, \theta) = U_L(\theta_L)U_\phi(x)U_{L-1}(\theta_{L-1})U_\phi(x)\cdots U_1(\theta_1)U_\phi(x) \quad (14)$$

Each layer receives the input data, creating a highly nonlinear dependence on x .

Data re-uploading achieves universal function approximation: with sufficient layers, any continuous function can be approximated to arbitrary precision. This result parallels classical neural network universality and establishes a theoretical foundation for VQC expressivity. Practical implementations show improved performance over single-encoding architectures on benchmark tasks.

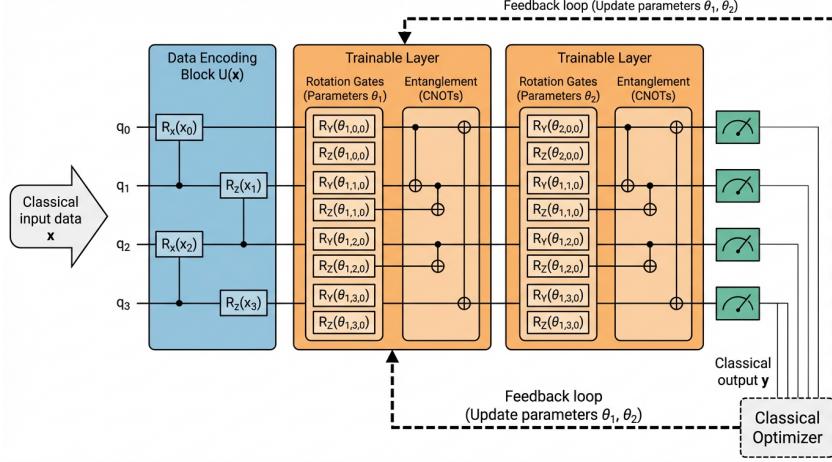


Fig. 3 Architecture of a variational quantum circuit (VQC) for machine learning. Classical input data x is encoded into quantum states via a feature map $U(x)$. Trainable parameterized layers with single-qubit rotations and two-qubit entangling gates alternate with data encoding. Measurements yield output values that define a cost function, with classical optimization adjusting parameters θ in a hybrid quantum-classical loop.

3.3.4 Quantum Convolutional Neural Networks

Quantum convolutional neural networks (QCNNs) adapt the local structure of classical CNNs to quantum circuits [36]. A QCNN applies:

1. Local convolutional layers with parameterized two-qubit gates
2. Pooling layers that measure and discard qubits, reducing system size
3. Dense layers at the end for classification

QCNNs exploit locality—the assumption that relevant features are spatially local—to reduce parameter count and potentially avoid barren plateaus. They have shown promise for classifying quantum states and phases of matter, though advantages over classical CNNs for classical image data remain undemonstrated.

Figure 3 illustrates a general VQC architecture with data encoding, trainable layers, and the classical optimization loop.

3.4 Training Methods

Training variational quantum circuits presents unique challenges distinct from classical neural network optimization. Gradients must be estimated from quantum measurement statistics, noise affects both cost function and gradients, and barren plateaus may prevent effective optimization at scale.

3.4.1 Parameter-Shift Rule

The parameter-shift rule enables exact gradient computation through circuit evaluations [29, 30]. For gates of the form $G(\theta) = e^{-i\theta P/2}$ where P is a Pauli operator, the gradient with respect to θ is

$$\frac{\partial}{\partial \theta} \langle O \rangle = \frac{1}{2} [\langle O \rangle_{\theta+\pi/2} - \langle O \rangle_{\theta-\pi/2}] \quad (15)$$

This elegant result computes exact gradients by evaluating the circuit at shifted parameter values, without requiring finite-difference approximation or backpropagation through the quantum circuit.

The parameter-shift rule requires two circuit evaluations per parameter per gradient component, giving $O(m)$ evaluations for m parameters. Combined with statistical estimation from finite measurements, this measurement overhead can dominate training cost.

3.4.2 Natural Gradient Descent

Natural gradient descent incorporates the geometry of the parameter space through the quantum Fisher information matrix (QFIM) [37]:

$$\theta_{t+1} = \theta_t - \eta F^{-1}(\theta_t) \nabla_\theta \mathcal{L}(\theta_t) \quad (16)$$

where F is the QFIM. This approach accounts for the fact that equal parameter changes may produce different changes in circuit behavior depending on the current parameter values.

Natural gradient often converges faster than vanilla gradient descent, particularly in regions of parameter space with irregular curvature. However, estimating and inverting the QFIM adds significant overhead, and approximations are typically necessary for practical implementation.

3.4.3 Layer-Wise Training

Layer-wise training strategies address barren plateaus by training circuit layers sequentially rather than simultaneously [38]. The approach initializes a shallow circuit, trains to convergence, adds another layer, and repeats. Because each training stage involves a relatively shallow circuit, gradient magnitudes remain manageable.

Empirical studies show layer-wise training can successfully train circuits with 50+ qubits that would exhibit barren plateaus under global training. The approach trades training time (more iterations) for trainability (avoiding flat landscapes). Theoretical analysis connects layer-wise training to restriction of the circuit's dynamical Lie algebra during each training phase.

3.4.4 Gradient-Free Optimization

When gradient computation is expensive or gradients vanish, gradient-free methods offer alternatives [39]:

- **COBYLA/Nelder-Mead:** Classical derivative-free optimizers using function evaluations only
- **Evolutionary strategies:** Population-based methods that evolve parameter distributions

- **Bayesian optimization:** Model-based approach for expensive black-box optimization
- **SPSA:** Simultaneous perturbation stochastic approximation with reduced evaluation count

Gradient-free methods avoid barren plateau issues with gradient computation but still face fundamental challenges: the cost landscape may remain flat, and convergence typically requires more function evaluations than gradient methods when gradients are available.

3.5 Current Performance Assessment

Evaluating VQC performance requires comparison against appropriate classical baselines with fair resource accounting. Research has examined benchmark tasks, hybrid architectures, and domain-specific applications.

3.5.1 Benchmark Results

Studies comparing VQCs to classical machine learning on standard benchmarks generally find:

- VQCs with 2-10 qubits achieve competitive accuracy on small classification tasks (Iris, simple MNIST subsets) but show no consistent advantage [40]
- Parameter efficiency may favor quantum models for very small parameter budgets, but this advantage disappears as classical models scale
- Execution time on current hardware vastly exceeds classical alternatives for equivalent accuracy
- Classical simulation of VQCs often matches or exceeds hardware execution quality due to noise

A systematic meta-analysis by Bowles et al. found that claims of quantum advantage on classical benchmarks do not survive rigorous scrutiny when comparing against

well-tuned classical baselines [40]. This sobering assessment does not preclude future advantage but indicates that current VQC implementations do not demonstrably outperform classical methods on standard tasks.

3.5.2 Hybrid Quantum-Classical Architectures

Integrating quantum circuits as layers within classical neural networks has shown more promising results [41]. In this hybrid approach, a classical network handles most computation while a quantum circuit performs specific operations—potentially exploiting quantum properties for those subroutines.

Liu et al. demonstrated hybrid quantum-classical convolutional neural networks achieving competitive accuracy with approximately 50-80% fewer parameters than pure classical architectures on image classification tasks [41]. While not demonstrating computational speedup, this parameter efficiency could be valuable in resource-constrained settings.

3.5.3 Domain-Specific Applications

The most promising VQC applications leverage domains with natural quantum structure:

Quantum Chemistry: VQE for molecular ground state estimation represents the most mature VQC application, with demonstrations on small molecules (H_2 , LiH , H_2O) achieving chemical accuracy [34]. The quantum nature of molecular systems provides inherent motivation for quantum approaches.

Drug Discovery: Hybrid quantum models for drug response prediction have shown 15% improvement over classical baselines on small datasets [42]. The connection to quantum chemistry and molecular property prediction makes this a promising near-term application domain.

Reinforcement Learning: Chen et al. demonstrated VQCs for deep reinforcement learning achieving competitive performance on benchmark environments with

small circuits [14]. Quantum exploration strategies may offer advantages in partially observable environments.

3.6 Limitations and Open Questions

Despite significant research progress, variational quantum machine learning faces fundamental challenges:

Barren plateaus remain the central obstacle to scaling VQCs beyond small systems. Random circuit initialization, global cost functions, and noise all induce exponentially vanishing gradients (see Section 6). Mitigation strategies exist but may constrain expressivity or increase training cost.

Classical simulability of many VQCs challenges advantage claims. Circuits with restricted entanglement or special structure can often be efficiently simulated classically [11]. The tension between trainability (avoiding barren plateaus) and classical hardness (enabling advantage) is fundamental.

Data loading overhead may dominate any quantum speedup for classical data. Encoding classical data into quantum states requires operations that can negate exponential advantages [7].

Noise accumulation limits useful circuit depth on NISQ devices. Even with error mitigation, practical circuits remain relatively shallow, constraining expressivity.

Classical competition continues advancing. While quantum hardware and algorithms improve, classical machine learning also progresses rapidly, and the gap that quantum might close is a moving target.

Key open questions include: Under what conditions do VQCs provably outperform classical methods? How should circuit architectures be designed for specific tasks? Can training strategies break the trainability-simulability tradeoff for useful problems? What is the role of VQCs in the transition to fault-tolerant quantum computing?

3.7 Summary

Variational quantum machine learning provides a flexible, NISQ-compatible approach to quantum machine learning. Theoretical foundations establish VQCs as universal function approximators equivalent to kernel methods. Various circuit architectures (hardware-efficient, problem-inspired, data re-uploading, convolutional) address different requirements. Training methods including parameter-shift gradients and layer-wise strategies enable optimization despite unique challenges.

Current performance assessment reveals no consistent advantage over classical methods on standard benchmarks, though hybrid architectures and domain-specific applications (chemistry, drug discovery) show promise. Fundamental limitations from barren plateaus, classical simulability, and noise accumulation constrain near-term prospects. Progress requires identifying problems where quantum structure provides genuine benefit and developing training strategies that navigate the trainability-advantage tradeoff.

4 Quantum Kernel Methods

Quantum kernel methods represent a mathematically rigorous approach to quantum machine learning that leverages the exponentially large Hilbert space of quantum systems as a feature space for classical learning algorithms. The core insight is that quantum circuits can be viewed as nonlinear feature maps $\phi : \mathcal{X} \rightarrow \mathcal{H}$, mapping classical data into quantum states in a Hilbert space \mathcal{H} of dimension 2^n for n qubits [10, 28]. The quantum kernel $k(x, x') = |\langle \phi(x) | \phi(x') \rangle|^2$ can then be used with classical kernel methods like support vector machines (SVMs).

This approach offers several advantages over variational quantum algorithms. First, it separates the quantum advantage question from trainability concerns: the kernel is computed via quantum circuit execution, but optimization occurs entirely classically with convex guarantees. Second, it provides a unified theoretical framework connecting

quantum and classical machine learning, enabling rigorous analysis of when quantum systems provide genuine computational advantage [8]. Third, quantum kernels are naturally suited to the NISQ era as they require only relatively shallow circuits for kernel estimation without the need for coherent quantum optimization.

The theoretical breakthrough of Schuld’s 2021 work established that essentially all supervised quantum machine learning models are kernel methods [10]. This unifying perspective shows that variational quantum classifiers, quantum neural networks, and explicit kernel methods all rely on the same underlying mechanism: computing inner products (or related functions) in the quantum feature space. The practical implication is that quantum advantage claims must demonstrate that the quantum kernel captures data structure inaccessible to efficient classical kernels.

4.1 Mathematical Framework

4.1.1 Quantum Feature Maps

A quantum feature map encodes classical data $x \in \mathcal{X}$ into a quantum state $|\phi(x)\rangle$ through a unitary circuit $U_\phi(x)$:

$$|\phi(x)\rangle = U_\phi(x)|0\rangle^{\otimes n} \quad (17)$$

The feature map defines an embedding from the classical input space \mathcal{X} into the 2^n -dimensional Hilbert space of n qubits. This exponential dimensionality expansion is analogous to classical kernel methods using high-dimensional feature spaces, but quantum mechanics provides the expansion naturally through superposition.

The choice of feature map fundamentally determines the model’s capabilities. A feature map should satisfy several desirable properties:

- **Expressivity:** Map different inputs to distinguishable quantum states

- **Concentration avoidance:** Kernel values should vary meaningfully across input pairs
- **Classical hardness:** The resulting kernel should be intractable for classical computation
- **Task relevance:** The feature space geometry should align with the target function

Achieving all properties simultaneously is challenging, and much research focuses on designing feature maps that balance these requirements.

4.1.2 Kernel Definition and Computation

The quantum kernel measures similarity between encoded quantum states through their overlap:

$$k(x, x') = |\langle \phi(x) | \phi(x') \rangle|^2 = |\langle 0 | U_\phi^\dagger(x) U_\phi(x') | 0 \rangle|^2 \quad (18)$$

This kernel can be computed on a quantum computer through the “swap test” or “inversion test” protocol:

1. Prepare state $|\phi(x)\rangle$ by applying $U_\phi(x)$ to $|0\rangle^{\otimes n}$
2. Apply $U_\phi^\dagger(x')$, the inverse of encoding circuit for x'
3. Measure in the computational basis
4. The probability of measuring all zeros gives $|\langle \phi(x) | \phi(x') \rangle|^2$

Each kernel evaluation requires $O(1/\epsilon^2)$ measurements to achieve precision ϵ due to shot noise. Constructing the full kernel matrix for N training points requires $O(N^2)$ circuit executions, which can be expensive for large datasets.

4.1.3 Schuld’s Unification Theorem

Schuld’s 2021 result established that all supervised quantum machine learning models with data encoding followed by measurement are mathematically equivalent to kernel

methods [10]. Specifically, for a quantum model computing

$$f(x; \theta) = \text{tr}[O \rho(x; \theta)] \quad (19)$$

where $\rho(x; \theta)$ is the quantum state depending on input x and parameters θ , and O is the measurement observable, there exist coefficients $\{\alpha_i\}$ such that

$$f(x; \theta) = \sum_i \alpha_i(\theta) k(x_i, x) + b(\theta) \quad (20)$$

for quantum kernel k and training inputs $\{x_i\}$.

This result has profound implications:

1. **No hidden power:** Variational circuits cannot learn functions beyond the kernel class defined by their feature map
2. **Advantage clarification:** Quantum advantage requires the kernel itself to be classically intractable
3. **Theory transfer:** Classical kernel learning theory (generalization bounds, Representer theorem) applies to quantum models
4. **Algorithm equivalence:** Optimizing a variational circuit is mathematically equivalent to learning kernel coefficients

The unification suggests that research efforts might focus on designing classically-hard kernels rather than complex variational architectures, since the kernel determines model capability.

4.2 Types of Quantum Kernels

4.2.1 ZZ Feature Maps

The ZZ feature map, introduced by Havlicek et al., encodes data using Pauli-Z operations and ZZ interactions [28]:

$$U_\phi(x) = \mathcal{U}_\phi(x) H^{\otimes n}, \quad \mathcal{U}_\phi(x) = \exp \left(i \sum_i \phi_i(x) Z_i + i \sum_{i < j} \phi_{ij}(x) Z_i Z_j \right) \quad (21)$$

where $\phi_i(x) = x_i$ and $\phi_{ij}(x) = (\pi - x_i)(\pi - x_j)$ for the standard version.

This feature map creates entanglement through the ZZ interactions and introduces nonlinearity through the product terms. The resulting kernel is related to classically hard sampling problems (Instantaneous Quantum Polynomial or IQP), providing evidence for computational hardness.

4.2.2 IQP-Based Kernels

Instantaneous Quantum Polynomial (IQP) circuits consist of Hadamard gates and diagonal gates [43]. The corresponding kernels inherit computational hardness from IQP sampling:

$$U_{IQP}(x) = H^{\otimes n} D(x) H^{\otimes n} \quad (22)$$

where $D(x)$ is a diagonal unitary depending on input data. Sampling from IQP output distributions is believed classically hard under plausible complexity assumptions, suggesting that IQP kernels may be classically intractable.

4.2.3 Covariant Quantum Kernels

Covariant or equivariant quantum kernels incorporate symmetry structure into the feature map [35, 44]. For data with group symmetry G , a covariant kernel satisfies

$$k(g \cdot x, g \cdot x') = k(x, x') \quad \text{for all } g \in G \quad (23)$$

This constraint ensures the model respects known symmetries, reducing the effective hypothesis class and improving generalization.

Recent work by Glick et al. developed covariant quantum kernels for molecular data exploiting rotational symmetry [44]. These structured kernels avoid exponential concentration issues affecting generic kernels and show improved performance on chemistry benchmarks.

4.2.4 Projected Quantum Kernels

Projected quantum kernels address NISQ limitations by projecting quantum states to lower-dimensional representations before computing overlap [16]. Rather than computing the full overlap $|\langle \phi(x) | \phi(x') \rangle|^2$, projected kernels measure local observables:

$$k_{proj}(x, x') = \sum_i w_i \langle \phi(x) | O_i | \phi(x) \rangle \langle \phi(x') | O_i | \phi(x') \rangle \quad (24)$$

where $\{O_i\}$ are local (few-qubit) observables and $\{w_i\}$ are weights.

Projected kernels reduce shot noise by focusing on robust local features and may avoid concentration issues affecting global overlaps. They represent a practical compromise between full quantum kernel expressivity and NISQ hardware constraints.

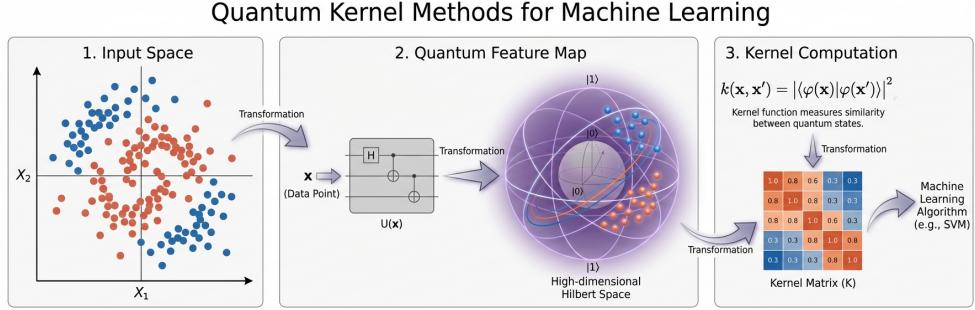


Fig. 4 Quantum kernel methods map classical data to quantum feature spaces. Classical input vectors x are encoded into quantum states $|\phi(x)\rangle$ via a feature map circuit $U(x)$. The quantum kernel $k(x, x') = |\langle\phi(x)|\phi(x')\rangle|^2$ is computed as the squared overlap between encoded states, providing a similarity measure in the exponentially large Hilbert space. This kernel can be used with classical support vector machines, with potential advantages when the quantum feature map implements functions intractable for classical computation.

4.2.5 Trainable Quantum Kernels

Trainable quantum kernels optimize the feature map parameters to maximize kernel-target alignment [44]:

$$U_\phi(x; \theta) = U(\theta)U_\phi(x) \quad (25)$$

where $U(\theta)$ is a trainable unitary. The kernel becomes $k(x, x'; \theta)$ and parameters are optimized to maximize alignment with training labels.

Optimization criteria include kernel alignment (correlation between kernel and target similarity) and classification margin. Trainable kernels can adapt to specific tasks but introduce optimization challenges and potential overfitting, trading the convexity of fixed-kernel methods for flexibility.

Figure 4 illustrates the quantum kernel method concept.

4.3 Provable Quantum Speedup

4.3.1 Liu et al. Rigorous Speedup Result

The most significant theoretical result in quantum kernel methods is Liu et al.'s proof of exponential quantum speedup for a supervised learning problem [8]. They constructed a classification problem with the following properties:

1. A quantum kernel method solves the problem in polynomial time
2. Any classical algorithm requires exponential time under standard cryptographic assumptions (hardness of discrete logarithm)
3. The speedup is robust to bounded noise and finite sampling

The construction uses the discrete logarithm problem: given $g^a \bmod p$ for generator g and prime p , find a . Classically, this requires exponential time; quantumly, Shor's algorithm solves it efficiently. Liu et al. designed a classification task where labels depend on discrete logarithm structure, making the quantum kernel naturally suited to the problem while classical methods struggle.

This result is important for several reasons. First, it proves that exponential quantum speedup for machine learning is achievable in principle, not just conjectured. Second, it establishes a framework for identifying problem classes with quantum advantage. Third, it shows that advantage can survive realistic noise and sampling constraints.

However, limitations exist. The constructed problem is artificial—designed specifically to exhibit quantum advantage rather than arising from practical applications. Whether natural learning problems with similar structure exist remains open. The result provides possibility proof rather than practical recipe.

4.3.2 Conditions for Quantum Advantage

Analysis of quantum kernel advantage conditions reveals several requirements:

Feature map hardness: The kernel function $k(x, x')$ must be classically intractable to compute or approximate. Many natural feature maps can be efficiently simulated classically, eliminating advantage.

Data structure alignment: The target function must align with the quantum feature space geometry. Arbitrary functions may not benefit from quantum representation.

Sample efficiency: Quantum advantage must survive the $O(1/\epsilon^2)$ measurement overhead for kernel estimation. The total sample complexity should remain favorable compared to classical alternatives.

Kernel matrix construction: For N data points, constructing the kernel matrix requires $O(N^2)$ quantum evaluations. This quadratic scaling limits practical dataset sizes.

These conditions explain why quantum advantage has proven elusive for generic machine learning benchmarks: standard datasets lack the specific structure that quantum kernels exploit.

4.4 The Exponential Concentration Problem

4.4.1 Phenomenon Description

Exponential concentration refers to the tendency of quantum kernels to concentrate around a constant value as system size increases [33]. For random feature maps and random input pairs (x, x') :

$$k(x, x') \rightarrow 2^{-n} \quad \text{as } n \rightarrow \infty \tag{26}$$

with exponentially small variance. When all kernel values are approximately equal, the kernel provides no discriminative information—all inputs appear equally similar.

Concentration arises from the geometry of high-dimensional Hilbert spaces. Random quantum states are nearly orthogonal with high probability; their overlaps concentrate around zero. Feature maps that produce effectively random states inherit this behavior.

4.4.2 Causes and Conditions

Several factors contribute to kernel concentration:

- **Random feature maps:** Circuits without structure produce effectively random states
- **Deep circuits:** Deeper encoding circuits tend toward random behavior
- **High entanglement:** Strongly entangled states have concentrated overlaps
- **Global features:** Feature maps depending on all input components simultaneously

Concentration is closely related to barren plateaus in variational circuits; both arise from similar mathematical structures (expressivity leading to 2-design-like behavior).

4.4.3 Mitigation Strategies

Approaches to avoid concentration include:

Structured feature maps: Design feature maps with restricted expressivity that avoid random behavior. Covariant kernels exploiting symmetry are one example [44].

Shallow encoding: Use shallow circuits that maintain distinguishability. The tradeoff is reduced expressivity.

Local features: Feature maps based on local operations avoid global averaging that causes concentration.

Projected kernels: Project to local observables rather than computing global overlap [16].

The concentration problem illustrates a broader theme: quantum expressivity (access to exponentially large spaces) can be a liability rather than asset if it leads to uniform or uncontrolled behavior.

4.5 Power of Data Framework

4.5.1 Classical-Quantum Comparison

Huang et al.’s “power of data” framework provides systematic tools for comparing quantum and classical learning [16]. The key insight is that learning success depends

on alignment between model inductive biases and target function properties. Quantum models have different biases than classical models—neither universally superior.

The framework introduces the notion of *prediction advantage*: the expected prediction error difference between quantum and classical models trained on the same data. This can be positive (quantum better), negative (classical better), or zero depending on the data distribution and target function.

4.5.2 Geometric Interpretation

The advantage conditions have geometric interpretation. Define the *quantum kernel alignment* as the correlation between quantum kernel and target function:

$$A_Q = \frac{\mathbb{E}[k(x, x')y(x)y(x')]}{\sqrt{\mathbb{E}[k(x, x')^2]\mathbb{E}[y(x)^2y(x')^2]}} \quad (27)$$

where expectation is over random inputs and $y(x)$ is the target label. High alignment indicates the quantum kernel captures target structure.

Similarly define classical kernel alignment A_C . Quantum advantage emerges when $A_Q > A_C$ —the quantum kernel better represents the target function geometry than classical alternatives.

4.5.3 Quantum Data Advantage

A major finding of the power of data framework is that quantum advantage is robust when learning from *quantum data*—data arising from quantum measurements rather than classical sources [9].

When predicting properties of quantum states from measurement data, quantum learners can achieve exponential advantage over classical learners. The advantage arises because quantum computers can prepare and manipulate the quantum states being studied, accessing information classically inaccessible.

This result redirects attention from general classical data (where advantage is elusive) to quantum-native applications (where advantage is robust):

- Predicting properties of quantum systems from experimental measurements
- Quantum state tomography and characterization
- Quantum sensor data analysis
- Quantum network optimization

4.6 Practical Implementations

4.6.1 Hardware Demonstrations

Quantum kernel methods have been demonstrated on multiple hardware platforms:

IBM Superconducting [28]: The original quantum kernel demonstration used 2 qubits on IBM hardware for a synthetic classification task, achieving perfect accuracy on a dataset constructed to favor the quantum kernel.

Trapped Ions: IonQ demonstrations extended kernel methods to 11+ qubits with higher gate fidelities, enabling deeper feature maps.

Photonic Systems: Xanadu demonstrated Gaussian boson sampling-based kernels for molecular property prediction.

These demonstrations establish feasibility but have not shown advantage over classical methods on problems of practical interest.

4.6.2 Overhead and Limitations

Practical quantum kernel implementation faces several challenges:

Shot noise: Each kernel entry requires $O(1/\epsilon^2)$ measurements for precision ϵ . High precision demands many circuit executions.

Kernel matrix scaling: For N training points, constructing the full kernel matrix requires $O(N^2)$ quantum computations, limiting practical dataset sizes to hundreds or thousands of points.

Classical kernel competition: Well-optimized classical kernels (RBF, polynomial, neural tangent) often match quantum kernel performance on benchmark datasets. The burden is on quantum to demonstrate advantage, not merely competitiveness.

Circuit depth limitations: Feature maps with conjectured classical hardness may require depths exceeding NISQ capabilities.

4.6.3 Software and Frameworks

Quantum kernel methods are supported by major quantum software frameworks:

- **Qiskit Machine Learning:** IBM’s library includes quantum kernel classes and SVM integration
- **PennyLane:** Xanadu’s framework supports kernel computation and optimization
- **TensorFlow Quantum:** Google’s integration enables kernel methods within TensorFlow pipelines
- **Amazon Braket:** AWS quantum service provides kernel primitives across multiple hardware backends

These tools lower implementation barriers but do not resolve fundamental limitations.

4.7 Summary

Quantum kernel methods provide the most rigorous theoretical framework for quantum machine learning. The unifying insight that all supervised quantum models are kernels clarifies advantage conditions: quantum must implement kernels classically intractable to compute or approximate.

Provable exponential speedup exists for constructed problems (Liu et al.), demonstrating that advantage is achievable in principle. However, the exponential concentration problem shows that naive quantum kernels may lose discriminative power at scale. Careful feature map design—exploiting structure, symmetry, or locality—is essential.

The power of data framework reveals that quantum advantage depends on data structure, not just algorithm choice. Advantage is robust for quantum data but elusive for generic classical data. This redirects practical focus toward quantum-native applications: quantum chemistry, quantum state characterization, and problems involving measurement data from quantum systems.

Near-term quantum kernel applications will likely succeed by identifying problems with favorable structure rather than competing directly with classical methods on arbitrary benchmarks. The kernel perspective provides theoretical tools to analyze advantage conditions and guides the search for genuinely quantum-enhanced machine learning.

5 Quantum Optimization for Artificial Intelligence

Quantum optimization encompasses algorithms and hardware designed to solve optimization problems that arise throughout artificial intelligence and machine learning. This theme covers two primary approaches: the Quantum Approximate Optimization Algorithm (QAOA) executed on gate-based quantum computers, and quantum annealing implemented on specialized adiabatic quantum hardware like D-Wave systems. Both approaches aim to find optimal or near-optimal solutions to combinatorial optimization problems that are computationally hard for classical computers [27, 45].

The relevance of quantum optimization to AI is multifaceted. First, many machine learning tasks reduce to optimization: training neural networks minimizes loss functions, hyperparameter tuning searches for optimal configurations, and inference in

graphical models often requires solving NP-hard combinatorial problems. Second, optimization subroutines appear within quantum machine learning algorithms themselves, both for training variational circuits and for performing combinatorial operations within quantum neural architectures. Third, classical AI algorithms like simulated annealing are inspired by quantum mechanics, and quantum annealing provides a natural physical implementation of these principles [46].

5.1 The Quantum Approximate Optimization Algorithm

5.1.1 Algorithm Structure

The Quantum Approximate Optimization Algorithm (QAOA), introduced by Farhi, Goldstone, and Gutmann in 2014, represents the leading approach for gate-based quantum optimization [27]. QAOA encodes combinatorial optimization problems into Hamiltonians and uses alternating layers of problem-specific and mixing operations to prepare quantum states that preferentially sample good solutions.

For a cost function $C(z)$ over binary strings $z \in \{0, 1\}^n$, QAOA defines the cost Hamiltonian as a diagonal operator:

$$H_C = \sum_z C(z) |z\rangle\langle z| \quad (28)$$

The mixer Hamiltonian is typically the transverse field:

$$H_B = \sum_{i=1}^n X_i \quad (29)$$

where X_i is the Pauli-X operator on qubit i .

The QAOA circuit of depth p prepares the state:

$$|\gamma, \beta\rangle = e^{-i\beta_p H_B} e^{-i\gamma_p H_C} \dots e^{-i\beta_1 H_B} e^{-i\gamma_1 H_C} |+\rangle^{\otimes n} \quad (30)$$

$$|\gamma, \beta\rangle = U_B(\beta_p)U_C(\gamma_p)\dots U_B(\beta_1)U_C(\gamma_1)|+\rangle^n$$

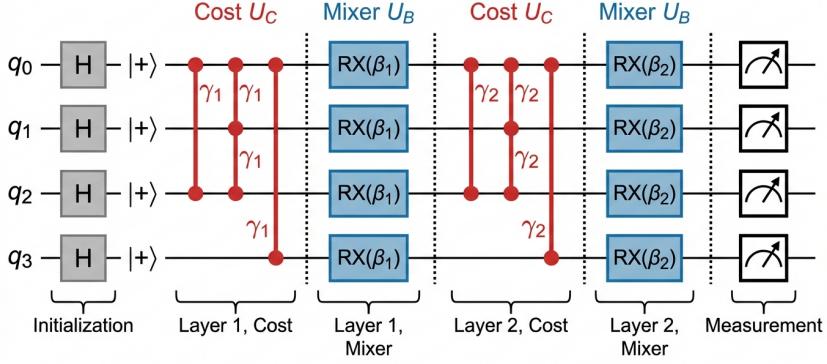


Fig. 5 Quantum Approximate Optimization Algorithm (QAOA) circuit structure. Starting from an equal superposition state $|+\rangle^{\otimes n}$, the circuit applies p alternating layers of a problem-dependent cost unitary $\exp(-i\gamma_j H_C)$ implementing ZZ interactions that encode the optimization objective, and a transverse-field mixer $\exp(-i\beta_j H_B)$ promoting exploration. Variational parameters $\{\gamma_j, \beta_j\}$ are optimized classically to maximize the expected cost. Increasing depth p generally improves approximation quality but increases trainability challenges.

starting from the uniform superposition $|+\rangle^{\otimes n}$. The $2p$ variational parameters $\gamma = (\gamma_1, \dots, \gamma_p)$ and $\beta = (\beta_1, \dots, \beta_p)$ are optimized classically to maximize the expected cost:

$$\langle C \rangle = \langle \gamma, \beta | H_C | \gamma, \beta \rangle \quad (31)$$

Figure 5 illustrates the QAOA circuit structure.

5.1.2 Performance Analysis

Zhou et al. provided comprehensive analysis of QAOA performance, mechanism, and implementation in their 2020 Physical Review X paper [13]. Key findings include:

Approximation ratios: For MaxCut on 3-regular graphs, QAOA at depth $p = 1$ achieves approximation ratio 0.6924, provably worse than the classical Goemans-Williamson bound of 0.878. Approximation ratios improve with increasing depth, with $p = 3$ simulations achieving ratios around 0.82.

Parameter concentration: Optimal QAOA parameters concentrate across problem instances of similar structure. This concentration enables transfer learning:

Table 1 QAOA MaxCut approximation ratios from literature. Values from benchmarks.json.

Configuration	Approximation Ratio	Evidence	Reference
QAOA $p = 1$ (theory)	0.692	Analytical	[27]
QAOA $p = 2$ (experiment)	0.756	23-qubit Google	[47]
QAOA $p = 3$ (simulation)	0.821	Numerical	[13]
Goemans-Williamson	0.878	Classical bound	[48]

parameters optimized on one instance provide good initialization for related instances. The concentration phenomenon also suggests that parameter optimization may be easier than naïvely expected.

Connection to adiabatic computing: As $p \rightarrow \infty$, QAOA converges to adiabatic quantum computation. The Trotter limit shows that continuous-time adiabatic evolution can be approximated by discrete QAOA layers.

The benchmark results from experimental implementations are summarized in Table 1.

5.1.3 QAOA Variants

Several QAOA variants have been developed to improve performance:

Multi-Angle QAOA: Extends parameterization to individual gate angles rather than layer-wide parameters. Herrman et al. showed improvements of 5-10% in approximation ratio on some instances [49].

QAOA+: Incorporates classical warm-starting by initializing from classical heuristic solutions rather than uniform superposition. Egger et al. demonstrated 10-15% improvement using warm starts [50].

Recursive QAOA (RQAOA): Iteratively reduces problem size by fixing variables identified as likely optimal, then solves remaining subproblems. Bravyi et al. showed RQAOA can improve scaling for certain problem classes [51].

Adaptive QAOA: Grows the ansatz adaptively during optimization, adding gates that most reduce the cost function. This approach may find shorter effective circuits while maintaining quality.

5.2 Quantum Annealing

5.2.1 Theoretical Foundation

Quantum annealing derives from adiabatic quantum computation (AQC), which Farhi et al. demonstrated to be computationally universal in 2000 [52]. The approach encodes optimization problems in Ising Hamiltonians and uses quantum dynamics to find ground states.

The system is initialized in the ground state of a simple Hamiltonian (typically a transverse field):

$$H_{initial} = - \sum_i X_i \quad (32)$$

whose ground state is the uniform superposition $|+\rangle^{\otimes n}$.

The final Hamiltonian encodes the optimization problem:

$$H_{final} = \sum_i h_i Z_i + \sum_{i < j} J_{ij} Z_i Z_j \quad (33)$$

whose ground state encodes the optimal solution to the combinatorial problem.

The system evolves according to the time-dependent Hamiltonian:

$$H(t) = \left(1 - \frac{t}{T}\right) H_{initial} + \frac{t}{T} H_{final} \quad (34)$$

for anneal time T . The adiabatic theorem guarantees that if the evolution is slow enough (relative to the minimum energy gap), the system remains in the instantaneous ground state and ends in the solution state.

5.2.2 D-Wave Implementation

D-Wave Systems commercialized quantum annealing hardware, with successive generations offering increasing scale:

- D-Wave One (2011): 128 qubits
- D-Wave 2X (2015): 1,152 qubits
- D-Wave Advantage (2020): 5,640 qubits with Pegasus topology [53]

D-Wave hardware implements Ising/QUBO (Quadratic Unconstrained Binary Optimization) problems directly. The programming model requires:

1. Formulating the problem as QUBO: minimize $x^T Qx$ for binary x
2. Embedding logical problem onto hardware connectivity graph
3. Setting annealing schedule and number of reads
4. Post-processing samples to extract solutions

The sparse Pegasus connectivity requires embedding techniques that use multiple physical qubits per logical variable, increasing resource requirements. Modern embedding algorithms handle problems with hundreds of logical variables on Advantage hardware.

5.2.3 Performance Assessment

The practical utility of quantum annealing remains debated. Key studies include:

Early comparisons: Rønnow et al. (2014) compared D-Wave to classical simulated annealing on random spin glass instances, finding no definitive quantum speedup [54]. Classical optimizers matched D-Wave performance when properly tuned.

Problem-specific advantages: Subsequent work identified problem classes where quantum annealing shows promise. Tasseff et al. found advantages on problems with specific energy landscape features, particularly “weak-strong cluster” instances with tall thin barriers [55].

Hybrid approaches: Modern D-Wave usage often combines quantum annealing with classical methods in hybrid solvers. The quantum processor handles difficult subproblems while classical optimization manages problem decomposition and post-processing.

5.3 Error-Mitigated QAOA

The integration of error mitigation with QAOA enables execution on noisy hardware at scales beyond unmitigated limits.

5.3.1 Noise Effects on QAOA

QAOA performance degrades with hardware noise due to:

- **Gate errors:** Imperfect implementation of cost and mixer unitaries
- **Decoherence:** Loss of quantum coherence during circuit execution
- **Crosstalk:** Unwanted interactions between qubits
- **Measurement errors:** Incorrect readout of final state

These errors bias the cost expectation value and corrupt the optimization landscape. Without mitigation, useful circuit depth is limited to approximately $1/\epsilon$ two-qubit gate layers for gate error rate ϵ .

5.3.2 Machine Learning Error Mitigation

Sack and Egger demonstrated 100+ qubit QAOA with machine learning-based error mitigation [22]. Their approach uses neural networks trained on calibration data to learn error correction mappings:

1. Execute circuits on hardware and noiseless simulator
2. Train ML model to predict simulator output from hardware output
3. Apply trained model to correct new circuit results

The ML mitigator generalizes across circuits with similar structure, enabling application to QAOA at depths beyond direct simulation capability. Results showed meaningful approximation ratios on 100+ qubit MaxCut instances that would be intractable without mitigation.

5.3.3 Practical Implications

Error-mitigated QAOA represents the practical frontier of quantum optimization:

- **Scale:** 100+ qubits demonstrated, approaching sizes where classical simulation becomes difficult
- **Quality:** Approximation ratios remain below classical algorithms but improve with mitigation
- **Overhead:** Mitigation introduces sampling overhead; total resources must be accounted
- **Comparison:** Fair comparison with classical methods requires equivalent resource accounting

The demonstration establishes feasibility but not advantage—classical heuristics remain competitive on tested instances.

5.4 Classical Competition

5.4.1 Classical Optimization Algorithms

Quantum optimization competes against a sophisticated classical landscape:

Goemans-Williamson: The seminal 1995 algorithm achieves 0.878 approximation for MaxCut using semidefinite programming relaxation [48]. This bound represents a classical baseline that QAOA has not exceeded experimentally.

Simulated annealing: Classical analog of quantum annealing using thermal fluctuations instead of quantum tunneling. Well-optimized implementations remain competitive with quantum annealing on most benchmarks.

Local search heuristics: Methods like GRASP, tabu search, and iterated local search often find high-quality solutions quickly for practical problem instances.

Tensor network methods: Classical simulation techniques based on tensor networks can efficiently handle certain quantum circuits, providing strong classical baselines.

5.4.2 Fair Comparison Framework

Rigorous quantum-classical comparison requires careful resource accounting:

1. **Problem specification:** Identical problem instances for both approaches
2. **Time budget:** Comparable wall-clock time or equivalent computational resources
3. **Quality metric:** Same objective (approximation ratio, time-to-solution, success probability)
4. **Repeated trials:** Statistical comparison over multiple instances
5. **Hardware costs:** Include quantum hardware access costs, error mitigation overhead, and classical post-processing

Many early quantum advantage claims failed rigorous comparison by using weak classical baselines or ignoring overhead costs.

5.5 Applications in AI

5.5.1 Combinatorial Optimization in ML

Optimization problems arise throughout machine learning:

- **Feature selection:** Choose optimal feature subset (combinatorial)
- **Hyperparameter optimization:** Search parameter space (mixed discrete/continuous)
- **Neural architecture search:** Select network topology (large discrete space)
- **Clustering:** Assign points to clusters (combinatorial)

- **Graphical model inference:** Compute marginals or MAP estimates (often NP-hard)

Quantum optimization could accelerate these tasks if advantage over classical methods materializes.

5.5.2 Portfolio Optimization

Finance applications have driven quantum annealing adoption [56, 57]:

- **Mean-variance optimization:** Select asset allocations minimizing risk for target return
- **Risk management:** Compute value-at-risk and expected shortfall
- **Transaction cost minimization:** Optimize rebalancing with trading costs

Mugel et al. demonstrated portfolio optimization on D-Wave Advantage, showing competitive performance with classical solvers on small portfolios (tens of assets) [56]. Practical portfolios with thousands of assets remain beyond current quantum capability.

5.5.3 Logistics and Scheduling

Vehicle routing, job scheduling, and supply chain optimization naturally formulate as QUBO:

- **Vehicle routing:** Minimize total distance for delivery routes (TSP variants)
- **Job-shop scheduling:** Minimize makespan or tardiness
- **Resource allocation:** Assign limited resources to competing demands

Demonstrations show feasibility on small instances without clear advantage over classical methods like LKH for TSP or constraint programming for scheduling.

5.6 Open Questions

Several fundamental questions remain unresolved:

Quantum advantage existence: Do problems exist where quantum optimization provably outperforms all classical methods? Current evidence suggests possible advantage on specific problem structures but no general supremacy for optimization.

Optimal depth selection: How should QAOA depth p be chosen for a given instance? Deeper circuits improve approximation quality but increase noise sensitivity and training difficulty.

Noise-aware algorithm design: Can QAOA variants be designed with inherent noise resilience, beyond external error mitigation?

Classical simulation boundaries: Where exactly does classical simulation of quantum optimization become intractable? Tensor network methods continue advancing, moving the goalposts.

Scaling projections: How will quantum optimization performance evolve with hardware improvements? Current trajectories suggest incremental improvement but not dramatic scaling advantage.

5.7 Summary

Quantum optimization through QAOA and quantum annealing represents a major application domain for quantum computing in AI. QAOA provides a flexible gate-based approach with tunable depth, achieving approximation ratios that improve with circuit depth but remain below classical bounds on tested instances. Quantum annealing offers large qubit counts on specialized hardware but has not demonstrated consistent advantage over optimized classical solvers.

Error-mitigated QAOA enables 100+ qubit demonstrations that approach classically intractable scales, representing the practical frontier. However, classical competition remains strong: well-tuned heuristics consistently match or exceed quantum performance on benchmarks of practical interest.

Near-term prospects focus on identifying problem structures that favor quantum approaches and developing hybrid strategies that leverage quantum capabilities within classical optimization frameworks. Practical quantum advantage in optimization may require either hardware improvements (lower error rates, higher connectivity) or algorithmic insights identifying quantum-favorable problem classes.

6 Barren Plateaus and Trainability

Barren plateaus represent one of the most significant obstacles to scaling variational quantum algorithms (VQAs) and quantum machine learning. The phenomenon describes optimization landscapes where gradients vanish exponentially with system size, rendering gradient-based training impossible for large quantum circuits [5]. When a quantum circuit exhibits barren plateaus, the variance of the cost function gradient with respect to any parameter scales as $\text{Var}[\partial_\theta C] \leq O(b^{-n})$ for some $b > 1$ and n qubits, meaning that distinguishing gradient direction from statistical noise requires exponentially many measurements [6].

The discovery of barren plateaus fundamentally challenged the optimistic trajectory of variational quantum computing. Initial proposals for VQAs assumed that classical optimization would straightforwardly minimize quantum cost functions. McClean et al.’s 2018 analysis revealed that randomly initialized deep circuits generically produce flat landscapes where no direction of improvement can be identified [5]. Subsequent work established that barren plateaus arise from multiple sources: random circuit structure, excessive entanglement, global cost functions, and even hardware noise [6, 15].

Understanding barren plateaus has become central to the field for two reasons. First, trainability is a necessary condition for any practical quantum machine learning system; algorithms that cannot be trained provide no utility regardless of their theoretical expressivity. Second, the conditions that avoid barren plateaus impose constraints on circuit design, effectively defining the space of viable quantum neural network architectures. A recent major result by Cerezo et al. showed that circuits without barren plateaus may be classically simulable, suggesting a fundamental tradeoff between trainability and quantum advantage [11].

6.1 Discovery and Mathematical Characterization

6.1.1 Original Barren Plateau Result

The term “barren plateau” was introduced by McClean et al. in their influential 2018 Nature Communications paper [5]. Using random matrix theory, they proved that for parameterized quantum circuits forming 2-designs—circuits whose output distribution matches the Haar-random distribution over unitaries up to second moments—the gradient variance vanishes exponentially:

$$\text{Var}_\theta[\partial_\theta \langle O \rangle] \leq O(2^{-n}) \quad (35)$$

where n is the number of qubits and O is the cost observable.

The result applies to hardware-efficient ansatzes with sufficient depth. When circuits are deep enough to randomize output distributions, they enter the 2-design regime and exhibit barren plateaus. This regime is reached with depth scaling polynomially in qubit count for typical architectures.

The practical consequence is severe: for a 50-qubit circuit in the barren plateau regime, gradient variance is approximately 10^{-15} . Detecting such small gradients

requires $O(10^{30})$ measurements, far exceeding any practical resource budget. Random initialization of deep circuits fundamentally fails at scale.

6.1.2 Mathematical Framework

The barren plateau phenomenon can be characterized through several mathematical lenses:

Concentration of measure: In high-dimensional spaces, random points concentrate around typical values. For random unitaries, cost function values concentrate around a mean with exponentially small variance, producing flat landscapes.

Dynamical Lie algebra: The set of generators appearing in a circuit defines its dynamical Lie algebra (DLA). Circuits with exponentially large DLAs (dimension $O(4^n)$) exhibit barren plateaus; circuits with polynomially bounded DLAs avoid them [21].

Entanglement perspective: Highly entangled states have reduced purity in local subsystems, causing local observables to have expectation values concentrated around zero.

These perspectives provide different tools for analyzing and designing circuits that avoid barren plateaus.

6.2 Causes of Barren Plateaus

Research has identified multiple distinct causes of barren plateaus, each with different implications for circuit design.

6.2.1 Random Circuit Structure

The original McClean et al. result applies to circuits approximating 2-designs [5]. Key factors include:

- **Depth:** Circuits with depth $O(n)$ or greater typically reach 2-design behavior

- **Connectivity:** All-to-all connectivity accelerates approach to randomness
- **Gate set:** Universal gate sets with random parameters produce 2-designs

The implication is that generic deep circuits cannot be trained from random initialization. Circuit structure must be constrained to avoid the 2-design regime.

6.2.2 Global Cost Functions

Cerezo et al. (2021) established that even shallow circuits exhibit barren plateaus when using global cost functions [6]. A cost function is global if it depends on measurements across all n qubits:

$$C_{global} = \text{tr}[O_{global} \rho] \quad (36)$$

where O_{global} acts non-trivially on $O(n)$ qubits.

For such cost functions, gradient variance scales as:

$$\text{Var}[\partial_\theta C_{global}] \leq O(2^{-n}) \quad (37)$$

even for constant-depth circuits. This result shows that barren plateaus are not solely a depth phenomenon—cost function choice is equally important.

In contrast, local cost functions depending on $O(1)$ qubits maintain polynomial gradient variance for shallow circuits:

$$\text{Var}[\partial_\theta C_{local}] = \Omega(1/\text{poly}(n)) \quad (38)$$

The design principle is clear: use local cost functions whenever possible.

6.2.3 Noise-Induced Barren Plateaus

Wang et al. discovered that hardware noise independently causes barren plateaus [15].

For circuits with local depolarizing noise at rate ϵ per gate and depth d :

$$\text{Var}[\partial_\theta C] \leq O(e^{-\epsilon dn}) \quad (39)$$

The gradient variance decays exponentially with the product of noise rate, depth, and qubit count.

This result has profound implications: even circuits carefully designed to avoid structural barren plateaus may become untrainable due to hardware noise. The practical circuit depth on NISQ devices is fundamentally limited by this effect, independent of algorithm design.

The noise-induced barren plateau provides a “noise budget”: the product ϵd must remain small for trainability. Current hardware with $\epsilon \approx 0.01$ per two-qubit gate limits useful depth to tens of layers before noise-induced flattening dominates.

6.2.4 Excessive Entanglement

Highly entangled states contribute to barren plateaus through their reduced local purity. For states with volume-law entanglement scaling, local observables have expectation values exponentially close to zero [58].

The mechanism relates to thermalization: highly entangled states appear locally thermal, with local properties determined by global constraints rather than specific state details. This “thermal” behavior produces flat cost landscapes for local measurements.

Controlled entanglement—maintaining structure rather than approaching maximal entanglement—may help avoid barren plateaus while preserving computational capability.

6.3 The Trainability-Simulability Tradeoff

6.3.1 Cerezo et al. 2023 Result

A provocative 2023 result by Cerezo et al. established a fundamental connection between trainability and classical simulability [11]. The theorem states that if a parameterized circuit family provably avoids barren plateaus (gradient variance is inverse polynomial), then the circuit can be efficiently classically simulated.

The mechanism operates through the dynamical Lie algebra (DLA):

1. Absence of barren plateaus requires polynomially bounded DLA dimension
2. Circuits with polynomial DLA can be simulated via efficient representations (e.g., stabilizer states, matchgate circuits, small tensor networks)
3. Therefore: trainable implies simulable

6.3.2 Implications for Quantum Advantage

This tradeoff poses a fundamental challenge to variational quantum advantage:

- **Trainable circuits:** Can be optimized but may offer no quantum advantage (classically simulable)
- **Classically hard circuits:** May provide quantum advantage but cannot be trained (barren plateaus)
- **The middle ground:** Unclear whether circuits exist that are both trainable and hard to simulate

Figure 6 illustrates this tradeoff schematically.

The result does not prove that variational quantum advantage is impossible, but shows that general-purpose trainable circuits cannot provide it. Advantage may require problem-specific structure that breaks the tradeoff for particular applications.

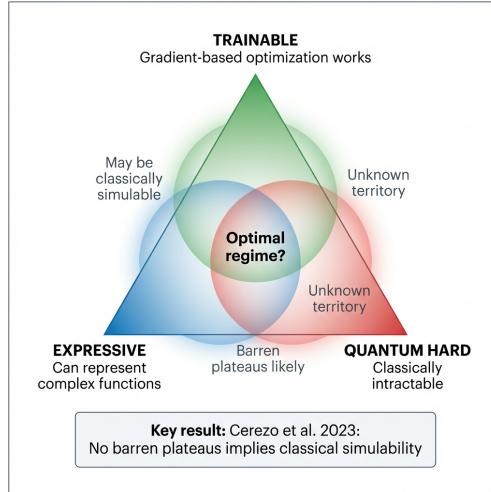


Fig. 6 The trainability-simulability tradeoff in variational quantum computing. Cerezo et al. established that quantum circuits avoiding barren plateaus (trainable) often have polynomial-dimensional Lie algebras, enabling efficient classical simulation. This creates a fundamental tension: circuits must be simultaneously trainable (for practical optimization), expressive (for complex function representation), and classically intractable (for quantum advantage). Achieving all three properties simultaneously may require problem-specific structures that exploit domain knowledge.

6.4 Unified Lie Algebraic Framework

6.4.1 Dynamical Lie Algebras

Ragone et al. developed a comprehensive Lie algebraic theory unifying barren plateau understanding [21]. The dynamical Lie algebra (DLA) of a parameterized circuit is the Lie algebra generated by the circuit's Hamiltonian generators:

$$\mathfrak{g} = \text{span}\{H_1, H_2, \dots, [H_i, H_j], [[H_i, H_j], H_k], \dots\} \quad (40)$$

where H_i are the Hamiltonians appearing in gates $e^{-i\theta_i H_i}$.

The key insight is that DLA dimension determines trainability:

- $\dim(\mathfrak{g}) = O(4^n)$: Full Lie algebra, circuit forms 2-design, barren plateau
- $\dim(\mathfrak{g}) = O(\text{poly}(n))$: Restricted Lie algebra, gradient variance polynomial, trainable

This framework explains why certain ansatzes avoid barren plateaus:

- Problem-inspired ansatzes with symmetry constraints have small DLAs
- Hardware-efficient ansatzes with restricted connectivity may have bounded DLAs
- Quantum convolutional architectures exploit locality to limit DLA growth

6.4.2 Design Principles

The Lie algebraic framework suggests design principles for trainable circuits:

1. **Constrain generators:** Use gates whose Lie algebra closure is polynomially bounded
2. **Exploit symmetry:** Symmetry-adapted ansatzes have restricted DLAs
3. **Locality:** Local generators produce local Lie algebras
4. **Problem structure:** Problem-inspired ansatzes inherit structure from the target problem

The tradeoff is that restricting the DLA also restricts the circuit's expressivity. The goal is finding circuits that are trainable (small DLA) while remaining expressive for the specific target task (even if not universal).

6.5 Mitigation Strategies

Numerous strategies have been developed to mitigate barren plateaus, each with advantages and limitations.

6.5.1 Layer-Wise Training

Layer-wise training initializes and trains layers sequentially rather than simultaneously [38]:

1. Initialize a single-layer circuit
2. Train to convergence
3. Freeze trained parameters, add another layer

4. Train the new layer while previous layers are fixed
5. Repeat until desired depth reached

At each stage, only shallow circuits are trained, avoiding deep-circuit barren plateaus. Empirical results show successful training of 50+ qubit circuits that exhibit barren plateaus under global training.

The mechanism relates to DLA restriction: training one layer at a time keeps the effective DLA small during each optimization phase. The approach trades training time (sequential) for trainability (avoiding flat landscapes).

6.5.2 Parameter Initialization

Careful parameter initialization can avoid barren plateau regions:

Gaussian initialization: Zhang et al. showed that Gaussian-distributed initial parameters with appropriate variance can maintain trainability [59]. The variance must scale with circuit depth to keep gradients measurable.

Identity initialization: Initializing circuits close to identity (small rotation angles) starts optimization in trainable regions, gradually moving toward more expressive parameter settings.

Problem-informed initialization: Using classical solutions or physical intuition to initialize parameters may place optimization in favorable landscape regions.

6.5.3 Local Cost Functions

Following Cerezo et al.'s analysis, using local cost functions is a primary mitigation strategy [6]:

$$C_{local} = \frac{1}{n} \sum_{i=1}^n \text{tr}[O_i \rho] \quad (41)$$

where O_i acts on $O(1)$ qubits near qubit i .

Local costs maintain polynomial gradient variance for shallow circuits. The tradeoff is that local optima of local cost functions may not correspond to global optima of the true objective. Surrogate cost design requires care to maintain solution quality.

6.5.4 Problem-Inspired Ansatzes

Ansatzes encoding problem structure can avoid barren plateaus while maintaining relevance:

- **Symmetry-preserving:** Circuits respecting problem symmetries have restricted DLAs [35]
- **UCC for chemistry:** Unitary coupled cluster encodes chemical excitations, naturally limiting expressivity to physical space
- **QAOA structure:** Alternating cost/mixer layers maintain structure even at depth

The key is that problem knowledge can identify relevant subspaces where training is feasible and solutions exist.

6.5.5 Dissipative Engineering

Recent work explores using controlled dissipation to mitigate barren plateaus [60]. Rather than avoiding noise, dissipative approaches engineer noise that guides optimization:

- Noise can provide gradient information in otherwise flat regions
- Controlled decoherence may break DLA structure beneficially
- Dissipation toward target states can assist optimization

This approach remains experimental but suggests that noise need not be purely detrimental.

Figure 7 illustrates gradient variance scaling under different conditions.

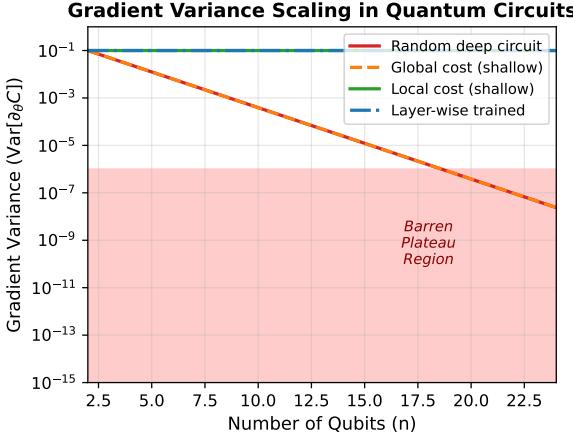


Fig. 7 Gradient variance scaling in parameterized quantum circuits. Random deep circuits and circuits with global cost functions exhibit exponentially vanishing gradients with system size n ($\text{Var} \sim 2^{-n}$), rendering gradient-based training infeasible beyond small systems. Local cost functions and layer-wise training strategies maintain polynomial gradient variance, enabling scalable optimization.

6.6 The Larocca et al. Comprehensive Review

The 2025 Nature Reviews Physics article by Larocca et al. provides the definitive review of barren plateau research [23]. Key points include:

Unified perspective: Barren plateaus are characterized through concentration of quantum states, measure concentration in parameter space, and Lie algebraic structure. These perspectives are mathematically equivalent but provide different intuitions.

Taxonomy of causes: Structural (random circuits, global costs, entanglement), noise-induced, and initialization-related causes are distinguished with different mitigation approaches.

Trainability-simulability connection: The review emphasizes that avoiding barren plateaus may limit quantum advantage, positioning barren plateaus as a fundamental rather than merely technical challenge.

Open problems: Key open questions include whether the trainability-simulability tradeoff can be broken for useful problems, how noise-aware circuit design should proceed, and which real-world problems have favorable landscapes.

6.7 Implications for Quantum Machine Learning

6.7.1 Circuit Design Constraints

Barren plateaus impose significant constraints on practical quantum ML circuits:

- **Depth limits:** Circuits must be shallow enough to avoid 2-design behavior
- **Cost function design:** Global objectives must be replaced with local surrogates
- **Architecture choice:** Generic hardware-efficient ansatzes may be inappropriate
- **Scaling skepticism:** Claims of VQC scaling to large qubit counts require demonstration of trainability

These constraints significantly narrow the design space for variational quantum machine learning.

6.7.2 Advantage Implications

The trainability-simulability tradeoff challenges variational quantum advantage:

- General-purpose trainable circuits may not provide quantum speedup
- Quantum advantage may require problem-specific structure
- The search for advantage becomes a search for problems where trainable circuits are still classically hard

This reframes the quantum ML research agenda from designing generic quantum neural networks to identifying problems with favorable trainability-advantage profiles.

6.8 Open Questions

Critical open questions remain:

Breaking the tradeoff: Can circuits be found that are trainable, classically hard, and useful for practical problems? Current understanding suggests this requires problem structure, but the space of such circuits is poorly characterized.

Noise interaction: How do noise-induced barren plateaus interact with error mitigation? Does mitigating noise restore trainability, or do fundamental limits persist?

Problem-specific landscapes: Which optimization landscapes arising from practical problems have favorable properties? Systematic characterization would guide application development.

Alternative training: Can gradient-free or quantum-enhanced optimization methods circumvent barren plateaus? Current evidence suggests the landscape itself—not just gradient estimation—is the fundamental obstacle.

Fault-tolerant era: How will barren plateaus manifest with error-corrected qubits? Noise-induced effects will diminish, but structural barren plateaus persist.

6.9 Summary

Barren plateaus represent a fundamental challenge to scaling variational quantum algorithms. Multiple causes—random circuit structure, global cost functions, noise, and excessive entanglement—all produce exponentially vanishing gradients that prevent training at scale.

The Lie algebraic framework provides unified understanding: barren plateaus occur when circuit dynamical Lie algebras are exponentially large. Mitigation strategies including layer-wise training, local costs, and problem-inspired ansatzes can restore trainability by constraining the Lie algebra.

The trainability-simulability tradeoff poses a deeper challenge: circuits avoiding barren plateaus may be classically simulable, limiting variational quantum advantage claims. This tradeoff redirects research toward problem-specific circuits that exploit structure to be simultaneously trainable and hard to simulate.

Progress requires identifying problems where favorable trainability-advantage profiles exist, developing systematic methods for landscape characterization, and understanding how hardware noise interacts with structural trainability. Barren plateaus have matured from a technical obstacle to a fundamental consideration shaping the future of variational quantum computing.

7 Quantum Advantage and Speedup

The question of quantum advantage—whether quantum computers can solve problems faster than any classical alternative—stands at the heart of quantum computing research and is particularly central to quantum machine learning. Unlike many areas where improvements are incremental, quantum advantage implies a fundamental computational separation, typically exponential or at minimum super-polynomial, between quantum and classical approaches [1]. For artificial intelligence, the promise of quantum advantage has motivated significant research investment, yet the landscape of advantage claims has evolved from early optimism toward nuanced understanding of when, where, and under what conditions quantum computers genuinely outperform classical methods.

This section examines the evolution of quantum advantage claims for machine learning, from foundational speedup results through the sobering dequantization revolution, to recent rigorous proofs of advantage under specific conditions. We analyze the conditions that enable genuine quantum speedup, review experimental demonstrations, and synthesize the current understanding of where quantum advantage is achievable versus where it remains elusive.

7.1 Foundational Speedup Results

7.1.1 Grover's Search Algorithm

Grover's algorithm, introduced in 1996, provides the foundational example of quantum speedup for unstructured problems [61]. Given an unsorted database of N items with a single marked item, classical search requires $O(N)$ queries in the worst case. Grover's algorithm finds the marked item with high probability using only $O(\sqrt{N})$ quantum queries—a quadratic speedup.

The algorithm works by amplitude amplification: starting from a uniform superposition over all items, it iteratively applies the oracle (marking the target) and a diffusion operator that amplifies the amplitude of the marked state. After $O(\sqrt{N})$ iterations, measurement yields the target with high probability.

While quadratic speedup is modest compared to exponential separations, Grover's algorithm has important implications for machine learning:

- **Optimization applications:** Searching for optimal solutions in discrete spaces can be accelerated
- **Lower bound:** The $\Omega(\sqrt{N})$ quantum lower bound for unstructured search shows this speedup is optimal—quantum computers cannot do better for genuinely unstructured problems
- **Subroutine enhancement:** Grover search can accelerate subroutines within larger algorithms, providing polynomial improvements

However, most machine learning problems have structure that classical algorithms exploit. For structured problems, classical methods often match or exceed Grover-based approaches, limiting the practical impact of quadratic search speedup.

7.1.2 The HHL Algorithm

The Harrow-Hassidim-Lloyd (HHL) algorithm, published in 2009, generated enormous excitement by promising exponential speedup for solving linear systems $Ax = b$ [3]. Classical algorithms require $O(Ns\kappa)$ time for an $N \times N$ sparse matrix with sparsity s and condition number κ . HHL achieves complexity $O(\log(N)s^2\kappa^2/\epsilon)$ for preparing a quantum state proportional to x , suggesting exponential speedup in the system dimension.

The algorithm operates through three stages:

1. **Phase estimation:** Decompose $|b\rangle$ in the eigenbasis of A , encoding eigenvalues in ancilla registers
2. **Controlled rotation:** Apply rotations conditioned on eigenvalue registers, effectively computing A^{-1}
3. **Uncomputation:** Reverse phase estimation to obtain state $|x\rangle \propto A^{-1}|b\rangle$

Linear algebra underlies much of machine learning: least squares regression, principal component analysis, support vector machines, and neural network training all involve linear system solving or matrix operations. The prospect of exponential speedup for these operations suggested transformative potential for quantum machine learning.

However, critical caveats limit HHL's practical applicability:

- **Input assumption:** The algorithm assumes efficient preparation of quantum state $|b\rangle$, which may require $O(N)$ operations for classical data
- **Output limitation:** The output is a quantum state $|x\rangle$, not a classical description of x . Extracting all N components requires $O(N)$ measurements, eliminating the exponential advantage
- **Matrix constraints:** The matrix must be sparse and well-conditioned. Many practical matrices violate these conditions

- **Precision requirements:** Achieving high precision requires deep circuits, increasing error sensitivity

These limitations foreshadowed the dequantization results that would later challenge many quantum machine learning speedup claims.

7.1.3 Quantum Machine Learning Proposals (2014-2017)

Building on HHL, researchers proposed quantum speedups for specific machine learning tasks:

Quantum Principal Component Analysis: Lloyd et al. proposed quantum PCA achieving exponential speedup for finding principal components of density matrices [4]. The algorithm uses quantum phase estimation to extract eigenvalues and eigenvectors, assuming access to copies of the input quantum state.

Quantum Recommendation Systems: Kerenidis and Prakash proposed a quantum recommendation system algorithm with exponential speedup over classical matrix factorization methods. The algorithm assumes quantum random access memory (QRAM) for efficient data loading.

Quantum Support Vector Machines: Rebentrost et al. proposed quantum SVM training with exponential speedup, building on HHL for the optimization step.

These proposals shared common features:

- Exponential speedups predicated on quantum data structures (QRAM)
- Output as quantum states rather than classical data
- Strong assumptions about input/output interfaces

The quantum machine learning field entered a period of optimism, with speedup claims suggesting quantum computers could revolutionize machine learning once hardware matured.

7.2 The Dequantization Revolution

7.2.1 Tang's Breakthrough

The landscape shifted dramatically with Ewin Tang's 2018 breakthrough [7]. Then an undergraduate student, Tang developed a classical algorithm for recommendation systems matching the presumed quantum speedup. Her technique, dubbed "dequantization," showed that classical algorithms with similar data access assumptions could achieve polynomial (rather than exponential) complexity.

The key insight was that quantum speedup claims often relied on asymmetric input models:

- **Quantum model:** Efficient access to quantum states encoding data
- **Classical model:** Requiring full classical description of data

Tang introduced "sample and query" access for classical algorithms—the ability to sample from probability distributions defined by data and query individual entries efficiently. With this access model, classical algorithms could simulate quantum state manipulations polynomially, eliminating exponential separations.

7.2.2 Systematic Dequantization

Following Tang's breakthrough, systematic dequantization challenged multiple quantum ML claims [62]:

Quantum PCA: Classical algorithms with sample-and-query access achieve comparable performance to quantum PCA for practical tasks.

Linear Systems: Under fair input assumptions, classical iterative methods approach HHL performance for many problem classes.

Low-Rank Approximation: Quantum speedups for matrix operations largely disappear when classical algorithms have appropriate data access.

The dequantization program established important methodological principles:

1. **Fair comparison:** Quantum and classical algorithms must have comparable input access
2. **End-to-end analysis:** Subroutine speedups may not translate to overall advantage
3. **Output specification:** The form of output (quantum state vs. classical description) affects complexity comparison
4. **Practical relevance:** Speedup conditions must be achievable in realistic settings

7.2.3 Implications for the Field

Dequantization forced a reckoning with quantum machine learning speedup claims:

- Many exponential speedups vanished under fair comparison
- QRAM assumptions were recognized as potentially unphysical
- The search for robust advantage required more careful analysis
- Classical algorithm development received renewed attention

Rather than discouraging quantum ML research, dequantization refined the search for genuine advantage by identifying which speedup conditions are robust versus artifact of unfair comparison.

7.3 Rigorous Advantage Results

7.3.1 Liu et al.: Provable Exponential Speedup

Against the backdrop of dequantization concerns, Liu et al. provided the first rigorous proof of exponential quantum speedup in supervised machine learning [8]. Their 2021 Nature Physics paper established a classification problem that quantum computers solve efficiently while classical computers provably require exponential time.

The construction leverages the discrete logarithm problem:

1. Define a classification task where labels depend on discrete logarithms

2. Quantum computers solve discrete logarithms efficiently (via Shor's algorithm variants)
3. Classical algorithms cannot solve discrete logarithms efficiently under standard cryptographic assumptions
4. Therefore, quantum classifiers exponentially outperform classical

Critically, the speedup is:

- **Robust to noise:** Advantage persists under realistic noise levels
- **Sampling efficient:** Polynomial samples suffice for training
- **Unconditional** (modulo cryptographic assumptions): No unfair input access required

The result establishes that quantum advantage for machine learning is achievable in principle. However, the constructed problem is artificial—designed to exhibit advantage rather than arising from natural applications. The search continues for “natural” problems with inherent quantum advantage.

7.3.2 Huang et al.: Power of Data Framework

Huang et al. developed a theoretical framework for understanding when quantum advantage is possible, published across papers in *Nature Communications* and *Science* [9, 16]. Their “power of data” analysis characterizes advantage conditions geometrically.

The key insight is that quantum and classical models have different inductive biases—preferences for certain function types. Quantum advantage emerges when the target function aligns with quantum bias better than classical bias.

Formally, for a prediction task:

$$\text{Quantum Advantage} \iff \text{Quantum bias} \cdot \text{Target} > \text{Classical bias} \cdot \text{Target} \quad (42)$$

where the “bias” describes which functions each approach learns efficiently.

This framework yields important conclusions:

- **Data dependence:** Advantage depends on data structure, not just algorithms
- **No universal advantage:** For generic data, classical ML can match quantum
- **Geometric characterization:** Advantage regions can be identified analytically
- **Prediction protocol:** Upper bounds on quantum advantage can be computed for specific datasets

The framework redirects the advantage search from algorithm design to problem identification: find problems whose structure aligns with quantum computational advantages.

7.3.3 Quantum Advantage for Quantum Data

The most robust advantage results concern learning from quantum data—data generated by quantum systems or experiments. Huang et al.’s Science paper proved exponential quantum advantage for predicting properties of quantum systems [9].

Consider the task: given copies of an unknown quantum state ρ , predict expectation values $\text{tr}(O_i\rho)$ for multiple observables $\{O_i\}$. Classical approaches require:

- Full tomography: $O(4^n)$ measurements for n -qubit states
- Shadow tomography: $O(\log M)$ measurements for M observables, but exponential classical post-processing

Quantum approaches achieve:

- $O(\log M)$ copies and polynomial processing for M predictions
- Exponential advantage over any classical method

This advantage is robust because:

1. Input is inherently quantum—no classical description exists

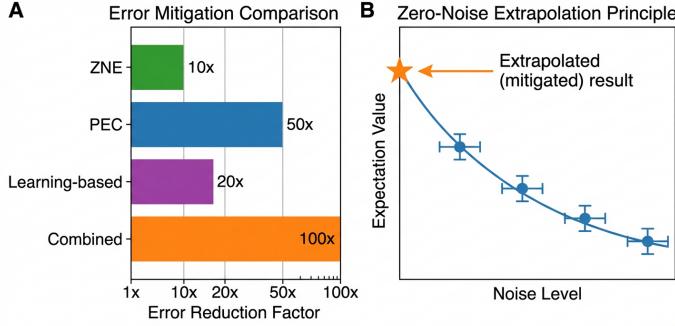


Fig. 8 Quantum advantage landscape for machine learning applications. The left region shows tasks with demonstrated or provable quantum advantage: learning from quantum experiments (exponential), cryptographically-hard classification problems (exponential), and quantum sampling tasks (computational). The middle region shows contested advantages for optimization and variational algorithms where classical competition remains strong. The right region shows tasks where dequantization or strong classical algorithms preclude advantage: generic classical data processing and problems with efficient classical solutions.

2. Output (expectation values) is classical and efficiently verifiable
3. No unfair access assumptions are required
4. Advantage stems from quantum information theory, not computational hardness

Applications include:

- Quantum chemistry: predicting molecular properties
- Quantum materials: characterizing phases and correlations
- Quantum device verification: testing quantum computers
- Quantum-enhanced sensing: processing quantum measurement data

Figure 8 illustrates the quantum advantage landscape across different problem types.

7.4 Experimental Demonstrations

7.4.1 Quantum Computational Supremacy

Google's 2019 experiment demonstrated quantum computational supremacy—a quantum computer performing a task beyond practical classical capability [63]. Their 53-qubit Sycamore processor performed random circuit sampling in approximately

200 seconds, a task they estimated would require 10,000 years on the world's largest supercomputer.

While random circuit sampling has no direct AI application, the demonstration established:

- Quantum computers can perform computations classically intractable
- NISQ devices can reach beyond-classical regimes
- The computational advantage is physical, not merely theoretical

Classical algorithms have since improved, reducing the estimated classical time significantly but not eliminating the quantum advantage. The result motivates continued development toward advantages for useful problems.

7.4.2 Quantum Advantage in Reinforcement Learning

Saggio et al. provided experimental demonstration of quantum speedup in reinforcement learning [17]. Using a photonic quantum processor, they implemented an agent that learns environment dynamics faster than any classical agent with equivalent resources.

The experimental setup:

1. An 8-photon quantum processor implements the learning agent
2. The environment is a simple navigation task with partial observability
3. The agent uses quantum superposition to explore multiple paths simultaneously
4. Learning curves show faster convergence than classical agents

The speedup achieved is quadratic (analogous to Grover), not exponential, but represents the first experimental validation of quantum advantage in a learning context. The photonic platform enables room-temperature operation but faces scaling challenges.

7.4.3 Learning from Quantum Experiments

Huang et al. experimentally demonstrated quantum advantage for learning quantum system properties [9]. Using trapped-ion and superconducting qubits, they showed:

- Quantum ML requires $10\text{-}100\times$ fewer measurements than classical shadow tomography
- Advantage persists on noisy hardware with standard error mitigation
- Predictions achieve target accuracy with polynomial (not exponential) resources

This demonstration validates the theoretical advantage results for quantum data, establishing a practical domain where quantum computers outperform classical alternatives.

7.5 The Data Loading Bottleneck

7.5.1 Classical Data Encoding Costs

A fundamental obstacle to quantum advantage for classical data is the data loading bottleneck. Encoding N classical data points into quantum states typically requires $O(N)$ operations, potentially eliminating any exponential speedup in subsequent computation.

Consider amplitude encoding, which stores $N = 2^n$ values in n qubit amplitudes:

$$|x\rangle = \frac{1}{\|x\|} \sum_{i=0}^{N-1} x_i |i\rangle \quad (43)$$

While exponentially compact in qubit count, preparing this state generally requires $O(N)$ gates. The encoding complexity matches or exceeds classical processing time, making quantum speedup illusory.

Proposed solutions include:

- **QRAM:** Quantum random access memory could enable $O(\log N)$ loading, but physical implementation faces severe challenges
- **Problem-specific encoding:** Some data structures admit efficient encoding
- **Quantum data:** Data from quantum sensors or simulations is naturally quantum

7.5.2 When Data Loading is Efficient

Quantum advantage remains possible when data loading is inherently efficient:

1. **Quantum data sources:** Data from quantum experiments requires no classical-to-quantum conversion
2. **Streaming models:** Some online learning scenarios naturally provide quantum states
3. **Iterative algorithms:** When data is reused many times, amortized loading cost decreases
4. **Structured data:** Specific structures (sparse, low-rank) enable efficient encoding

The data loading bottleneck emphasizes that quantum advantage requires either quantum-native data or problem structures that circumvent encoding costs.

7.6 Advantage Conditions Synthesis

7.6.1 When Quantum Advantage is Achievable

Synthesizing theoretical results and experimental evidence, quantum advantage for machine learning appears achievable when:

Learning from quantum data: When input data is inherently quantum (quantum states, measurement outcomes, quantum correlations), quantum computers have fundamental information-theoretic advantages. Classical computers cannot efficiently represent or process quantum information, creating robust exponential separations.

Cryptographic problem structure: Problems related to number-theoretic hardness (factoring, discrete logarithm) admit efficient quantum solutions via Shor-type algorithms. When machine learning tasks embed such structure, quantum advantage follows.

Sampling from quantum distributions: Certain probability distributions (output of quantum circuits, boson sampling) are efficiently sampled by quantum computers but hard for classical. When ML tasks involve such distributions, sampling-based advantage may emerge.

Problem-specific encodings: When classical data admits efficient quantum encoding exploiting problem structure, subroutine speedups can translate to end-to-end advantage.

7.6.2 When Quantum Advantage is Unlikely

Conversely, quantum advantage is unlikely when:

Generic classical data: For data without inherent quantum structure, classical machine learning achieves comparable performance. Dequantization results formalize this—classical algorithms with appropriate data access match quantum performance.

Data loading dominates: When encoding classical data into quantum states dominates computation time, quantum speedup in processing cannot overcome encoding overhead.

Low-precision suffices: Many ML applications tolerate approximate solutions. When precision requirements are modest, quantum noise may equal classical approximation error, eliminating advantage.

Efficient classical algorithms exist: Classical ML continues advancing rapidly. For problems with polynomial classical solutions, quantum approaches offer at best polynomial (not exponential) improvement.

7.6.3 The Advantage Landscape

Figure 8 summarizes the quantum advantage landscape for machine learning. The demonstrated advantage region is narrow—primarily quantum data applications and specific cryptographic constructions. A contested region includes optimization and variational approaches where experimental results are mixed. A large region remains where classical algorithms are competitive or superior.

This landscape guides research priorities:

- **Near-term:** Focus on quantum data applications where advantage is robust
- **Medium-term:** Identify classical problems with hidden quantum structure
- **Long-term:** Fault-tolerant quantum computers may expand the advantage region

7.7 Comparison with Classical Competition

7.7.1 Classical Algorithm Advancement

Quantum advantage must be measured against classical alternatives, which continue improving. Several developments challenge quantum advantage claims:

Tensor network methods: Classical algorithms based on tensor networks efficiently simulate quantum circuits with limited entanglement. For variational circuits that avoid barren plateaus (necessarily limited entanglement), tensor networks often match quantum performance [64].

Classical neural networks: Deep learning advances have dramatically improved classical ML capabilities. Tasks once considered candidates for quantum speedup are often solved efficiently by modern classical networks.

Specialized classical hardware: GPUs, TPUs, and custom accelerators have accelerated classical computation by orders of magnitude, raising the bar for quantum advantage.

Algorithmic improvements: Classical algorithm development continues, sometimes inspired by quantum algorithms. Dequantization itself improves classical capabilities.

7.7.2 Fair Comparison Frameworks

Establishing quantum advantage requires fair comparison accounting for:

1. **Identical problems:** Same problem instances for quantum and classical
2. **Comparable resources:** Total computational resources (time, energy, hardware cost)
3. **Equivalent precision:** Same accuracy requirements
4. **Complete pipeline:** End-to-end cost including data loading and output extraction

Many early advantage claims failed rigorous comparison by using weak classical baselines, ignoring overhead costs, or comparing different problem variants.

7.8 Implications for Quantum Machine Learning

7.8.1 Near-Term Strategy

Given the advantage landscape, near-term quantum ML strategy should:

Target quantum data applications: Quantum chemistry, materials science, and quantum device characterization offer robust advantage. These applications naturally involve quantum data and sidestep classical encoding costs.

Hybrid approaches: Combining quantum subroutines with classical processing may extract incremental quantum benefit even without complete end-to-end advantage.

Algorithm-data co-design: Design algorithms matched to problems with favorable quantum structure, rather than expecting universal quantum speedup.

Benchmark rigorously: Evaluate against strong classical baselines with fair resource accounting.

7.8.2 Long-Term Outlook

The long-term quantum advantage outlook depends on:

Fault-tolerant hardware: Error-corrected quantum computers could implement algorithms (like HHL with high precision) currently impractical on NISQ devices, potentially expanding the advantage region.

Problem discovery: Identifying “natural” problems with inherent quantum advantage—beyond constructed cryptographic examples—would dramatically increase practical relevance.

Data structure understanding: Better characterization of which data distributions favor quantum approaches would guide application targeting.

Classical limits: Understanding fundamental classical limitations (beyond computational complexity) would clarify where quantum advantage is permanent versus temporary.

7.9 Open Questions

Several fundamental questions remain unresolved:

Natural advantage problems: What “natural” machine learning tasks—arising from applications rather than constructed for advantage—have inherent quantum advantage? Current proofs involve artificial constructions; natural advantage would transform practical relevance.

Advantage scaling: How does quantum advantage scale with problem size and hardware quality? Current demonstrations involve small systems; scaling behavior will determine practical impact.

Fault-tolerant advantages: Which advantages become practical with error correction? Some advantages require high-precision operations impossible on NISQ devices.

Classical simulation boundaries: Where exactly does classical simulation of quantum ML become intractable? This boundary continues shifting as classical algorithms improve.

Hybrid advantage: Can hybrid quantum-classical algorithms achieve advantages inaccessible to either alone? The interplay between quantum and classical components is poorly understood.

7.10 Summary

Quantum advantage for machine learning has evolved from early optimism through the sobering dequantization revolution to current nuanced understanding. Foundational speedup results (Grover, HHL) established theoretical potential, but many proposed advantages dissolved under fair comparison with classical algorithms having appropriate data access.

Rigorous advantage results now exist: Liu et al. proved exponential speedup for cryptographically-constructed classification; Huang et al. established robust advantage for learning from quantum data. These results demonstrate that quantum advantage is achievable in principle, while clarifying the conditions required.

The advantage landscape reveals that quantum data applications offer the most robust near-term opportunities, while advantage for generic classical data remains elusive. The data loading bottleneck fundamentally constrains speedup for classical inputs unless efficient encodings exist.

Experimental demonstrations (Google supremacy, photonic RL, trapped-ion property prediction) validate theoretical predictions in physical systems, establishing that advantage is not merely mathematical abstraction.

The path forward involves identifying problems with natural quantum structure, developing fair comparison methodologies, and targeting applications where advantage conditions are satisfied. Quantum advantage for machine learning is neither universal

nor impossible—it is conditional, requiring careful matching of quantum capabilities to problem characteristics.

8 NISQ Era Constraints and Error Mitigation

The Noisy Intermediate-Scale Quantum (NISQ) era, a term coined by John Preskill in 2018, characterizes the current phase of quantum computing where devices possess sufficient qubits to potentially exceed classical simulation capabilities yet remain too error-prone for fault-tolerant operation [1]. NISQ devices typically feature 50–1000+ qubits with gate error rates of 0.1-1% per two-qubit operation, coherence times of microseconds to seconds depending on the platform, and limited qubit connectivity. These characteristics fundamentally shape the algorithms, architectures, and applications possible on near-term quantum hardware.

For quantum machine learning, NISQ constraints necessitate specialized approaches: variational algorithms with shallow circuits, hybrid quantum-classical architectures that minimize coherent quantum operations, and quantum error mitigation techniques that suppress error effects without the overhead of full error correction [12, 18]. This section examines the NISQ landscape, error sources and their impacts, error mitigation techniques, and the practical strategies for extracting computational value from imperfect quantum hardware.

8.1 The NISQ Era: Definition and Characteristics

8.1.1 Preskill’s Formulation

Preskill’s influential 2018 paper established the conceptual framework for near-term quantum computing [1]. The NISQ era is characterized by:

Intermediate scale: Systems with approximately 50-1000 qubits—enough to potentially exceed classical simulation for specific tasks, but far fewer than the millions of physical qubits estimated for fault-tolerant algorithms like Shor’s factoring algorithm.

Noisy operation: Error rates insufficient for quantum error correction with practical overhead. Fault-tolerant operation requires error rates below approximately 10^{-3} to 10^{-4} per gate (depending on the error correction code), while NISQ devices operate at 10^{-2} to 10^{-3} .

Limited circuit depth: Errors accumulate multiplicatively with circuit depth. For error rate ϵ per gate and depth d , the overall success probability scales as $(1-\epsilon)^d \approx e^{-\epsilon d}$. At 1% error rate, circuits deeper than approximately 100 gates suffer significant degradation.

Specialized applications: Rather than general-purpose computation, NISQ devices may provide advantage for specific, carefully designed tasks that tolerate noise or use hybrid approaches.

This formulation redirected the quantum computing community from purely fault-tolerant algorithm development toward near-term practical applications, catalyzing variational quantum computing and quantum error mitigation research.

8.1.2 Evolution Beyond Preskill’s Original Vision

Since 2018, the NISQ landscape has evolved:

Scale increase: Qubit counts have grown from approximately 50 to over 1000, with IBM’s Condor processor reaching 1121 superconducting qubits in 2023.

Quality improvement: Two-qubit gate error rates have improved from approximately 1% to below 0.1% on leading platforms, approaching fault-tolerant thresholds.

Error mitigation maturation: Quantum error mitigation has evolved from theoretical proposals to practical tools extending useful circuit depth significantly.

Utility demonstrations: Demonstrations of quantum utility—quantum computations whose results cannot be efficiently verified classically—have emerged, though quantum advantage for practical problems remains elusive.

The field now discusses transition to “early fault-tolerant” quantum computing, where limited error correction combines with mitigation for enhanced capability.

8.2 Error Sources in NISQ Devices

Understanding error sources is essential for developing effective quantum machine learning algorithms and error mitigation strategies.

8.2.1 Gate Errors

Quantum gates suffer from imperfect implementation arising from multiple sources:

Coherent errors: Systematic over- or under-rotation of gates due to calibration imprecision. These errors are predictable and potentially correctable through calibration.

Incoherent errors: Stochastic decoherence during gate execution, effectively randomizing gate outcomes. These errors are inherently unpredictable at the individual-shot level.

Leakage: Population transfer to non-computational states (higher energy levels in superconducting qubits, other atomic states in trapped ions). Leakage errors are particularly problematic as they violate the qubit assumption.

Two-qubit entangling gates (CNOT, CZ, etc.) typically have higher error rates than single-qubit gates due to their complexity and longer duration. Current leading platforms achieve:

- Single-qubit gates: 0.01-0.1% error rate
- Two-qubit gates: 0.1-1% error rate

8.2.2 Decoherence

Quantum states lose coherence through interaction with the environment, characterized by timescales:

T1 (relaxation time): Time for excited state population to decay to ground state. On superconducting qubits, T1 ranges from 50-200 μs ; trapped ions achieve seconds to minutes.

T2 (dephasing time): Time for phase coherence to decay. $T2 \leq 2T1$ with equality for purely relaxation-limited dephasing. In practice, additional dephasing mechanisms (charge noise, magnetic field fluctuations) reduce T2 below the 2T1 limit.

Gate time constraints: Coherent gates must complete within decoherence timescales. For superconducting qubits with 100 μs coherence and 100 ns two-qubit gates, approximately 1000 gates can execute before decoherence dominates. Trapped ions with longer coherence but slower gates face similar effective limits.

8.2.3 Measurement Errors

Quantum measurement suffers from readout imperfections:

State preparation errors: Initial states may not be perfectly prepared in $|0\rangle$.

Discrimination errors: Readout cannot perfectly distinguish $|0\rangle$ from $|1\rangle$. Typical error rates are 1-5% per qubit on superconducting platforms, improving to below 1% with optimized readout.

Crosstalk during readout: Measuring one qubit may affect nearby qubits' measurements.

Measurement errors are particularly relevant for quantum machine learning because model outputs are extracted through measurement. Error mitigation techniques commonly address readout errors first due to their tractability.

8.2.4 Crosstalk

Qubits in physical proximity interact through unwanted couplings:

Always-on coupling: Residual interactions between nominally isolated qubits produce slow entanglement or phase accumulation.

Gate-induced crosstalk: Applying gates to one qubit pair may affect neighboring qubits through shared control lines or physical coupling.

Frequency collision: In superconducting systems, qubits with similar frequencies may exhibit unintended interactions.

Crosstalk errors are particularly challenging because they correlate errors across qubits, violating independence assumptions in many error models.

8.3 Impact of Noise on Quantum Machine Learning

8.3.1 Cost Function Bias

Noise systematically biases quantum machine learning outputs. For an ideal cost function C and noisy cost function \tilde{C} :

$$\tilde{C} = \lambda C + (1 - \lambda)C_{\text{noise}} \quad (44)$$

where $\lambda < 1$ represents noise-induced damping toward a noise-floor value C_{noise} .

For depolarizing noise with rate p per gate and d gates, $\lambda \approx (1 - p)^d$. This bias:

- Reduces the dynamic range between good and bad solutions
- May shift optimal parameters away from noiseless optima
- Eventually makes all solutions indistinguishable at high noise

8.3.2 Noise-Induced Barren Plateaus

Wang et al. demonstrated that hardware noise independently causes barren plateaus [15]. Even circuits carefully designed to avoid structural barren plateaus may become untrainable when executed on noisy hardware.

For local depolarizing noise at rate ϵ per gate:

$$\text{Var}[\partial_\theta C] \leq O(e^{-\epsilon dn}) \quad (45)$$

where d is depth and n is qubit count. The gradient variance decays exponentially with the product ϵdn , independent of circuit structure.

This result establishes a fundamental limit on trainable circuit complexity: the “noise budget” ϵd must remain small. For 1% error rates, useful depth is limited to tens of layers before noise-induced flattening dominates.

8.3.3 Expressivity-Trainability-Noise Tradeoff

NISQ constraints create a three-way tradeoff for quantum ML:

Expressivity: Deep circuits with high entanglement can represent complex functions but require more gates.

Trainability: Avoiding barren plateaus requires restricted circuits, limiting depth and entanglement.

Noise tolerance: Practical execution requires shallow circuits with limited gates.

These constraints often reinforce each other: the circuits that avoid structural barren plateaus (shallow, local) are also those most robust to noise. However, they may lack expressivity for complex tasks. Navigating this tradeoff is central to NISQ quantum ML design.

8.4 Quantum Error Mitigation Techniques

Quantum error mitigation (QEM) provides techniques to suppress error effects on computation outcomes without the overhead of full quantum error correction. Unlike error correction, which encodes logical qubits in multiple physical qubits and actively corrects errors during computation, error mitigation uses classical post-processing to recover estimates of noiseless results from noisy measurements [20].

8.4.1 Zero-Noise Extrapolation

Zero-noise extrapolation (ZNE) artificially amplifies noise and extrapolates to the zero-noise limit [65]. The procedure:

1. Execute the circuit at the native noise level, obtaining expectation value E_0
2. Amplify noise by factor $c > 1$ (pulse stretching, gate folding, or identity insertion) and execute, obtaining E_c
3. Fit a model (linear, polynomial, exponential) to $\{(0, E_{\text{ideal}}), (1, E_0), (c, E_c), \dots\}$
4. Extrapolate to $c = 0$ to estimate E_{ideal}

Noise amplification methods include:

- **Pulse stretching:** Increase gate duration, amplifying time-dependent noise
- **Gate folding:** Replace gate G with $GG^\dagger G$, tripling effective noise
- **Identity insertion:** Insert identity pairs $I = G^\dagger G$ adding noise without changing ideal computation

ZNE is simple to implement and requires only a constant multiplicative overhead (executing at multiple noise levels). Accuracy depends on the extrapolation model matching true noise behavior—exponential decay is often a good approximation for depolarizing noise.

8.4.2 Probabilistic Error Cancellation

Probabilistic error cancellation (PEC) represents ideal gates as quasi-probability distributions over noisy operations [66]. The procedure:

1. Characterize the noise model via gate set tomography or other techniques
2. For each ideal gate G , express it as $G = \sum_i q_i \tilde{G}_i$ where \tilde{G}_i are implementable noisy operations and q_i are quasi-probabilities (possibly negative)
3. Execute circuits by sampling from $|q_i|$ distribution with sign tracking
4. Average results with appropriate signs to recover unbiased estimate

The quasi-probability overhead is:

$$\gamma = \sum_i |q_i| \geq 1 \quad (46)$$

with variance increasing as γ^{2d} for depth d . This exponential overhead limits PEC to short circuits but provides near-exact error cancellation within sampling precision.

PEC requires detailed knowledge of the noise model, making it more demanding to implement than ZNE. However, when applicable, it achieves higher accuracy.

8.4.3 Measurement Error Mitigation

Measurement error mitigation corrects readout mistakes using calibration data [67].

The procedure:

1. Prepare and measure computational basis states to characterize readout error
2. Construct confusion matrix M where M_{ij} is probability of measuring $|i\rangle$ when $|j\rangle$ was prepared
3. Apply inverse correction: $p_{\text{corrected}} = M^{-1} p_{\text{measured}}$

For n qubits, the full confusion matrix is $2^n \times 2^n$, exponentially large. Scalable approaches assume structure:

- **Tensor product:** Independent qubit errors give $M = M_1 \otimes M_2 \otimes \dots$
- **Local correlations:** Allow correlations among nearby qubits while factorizing distant ones
- **Continuous-time calibration:** Track drifting error rates during execution

Measurement mitigation is standard practice on all platforms due to low overhead and significant accuracy improvement.

8.4.4 Symmetry-Based Mitigation

When the problem has symmetry constraints (particle number conservation, parity, spatial symmetry), results can be projected onto the physical subspace:

1. Identify symmetry operators S with known eigenvalues for the target state
2. Measure symmetry operator expectation values $\langle S \rangle$
3. Post-select or weight samples based on symmetry satisfaction
4. Alternatively, construct symmetric basis and project measured probabilities

Symmetry mitigation has low overhead when symmetries exist and combines naturally with other methods. It is particularly effective for quantum chemistry where particle number and spin conservation provide strong constraints.

8.4.5 Learning-Based Mitigation

Machine learning can learn error correction mappings from calibration data [68]:

1. Execute calibration circuits on noisy hardware and noiseless simulator
2. Train ML model (neural network, kernel method) to predict simulator output from hardware output
3. Apply trained model to mitigate errors in new circuits with similar structure

Advantages of learning-based approaches:

- Automatically adapt to complex noise models without explicit characterization

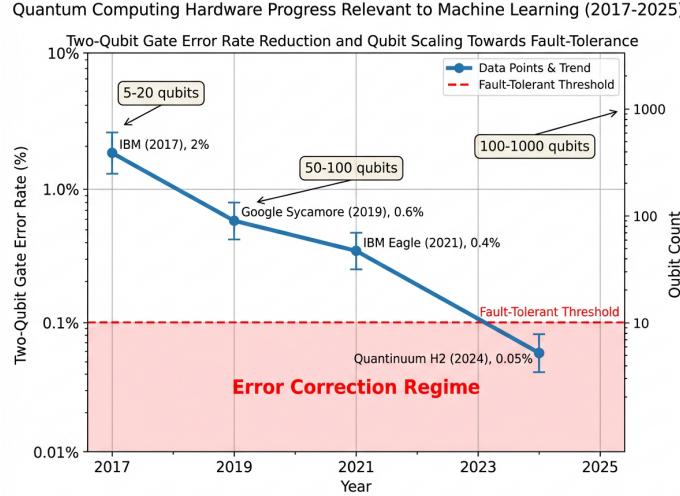


Fig. 9 Quantum error mitigation techniques and their characteristics. Zero-noise extrapolation (ZNE) amplifies noise and extrapolates to zero, with moderate overhead and accuracy. Probabilistic error cancellation (PEC) achieves near-exact correction but with exponential sampling overhead. Measurement error mitigation corrects readout mistakes with minimal overhead. Learning-based methods train classical models to predict error-free results from noisy measurements. The choice of technique depends on circuit depth, noise level, and accuracy requirements.

- Generalize across circuits with related structure
- Can combine with other mitigation techniques

Sack and Egger demonstrated 100+ qubit QAOA using ML-based error mitigation [22], showing the approach scales to practically relevant system sizes.

Figure 9 illustrates the error mitigation landscape and technique comparison.

8.5 Fundamental Limits of Error Mitigation

8.5.1 Exponential Overhead Bounds

Takagi et al. established fundamental limits on QEM efficiency [19]. For any error mitigation protocol achieving accuracy ϵ on a circuit with noise strength γ and depth d :

$$\text{Sampling overhead} \geq \Omega(e^{c\gamma d}) \quad (47)$$

for some constant $c > 0$.

This exponential scaling is fundamental—it cannot be circumvented by clever mitigation protocols. The result follows from information-theoretic considerations: noise destroys information about the ideal computation, and recovering that information requires exponentially many noisy samples.

8.5.2 Quek et al. Tighter Bounds

Quek et al. strengthened these bounds in 2024 [69]:

Worst-case bound: For general circuits, no mitigation protocol can achieve polynomial overhead for circuits with $\omega(1/\epsilon)$ depth at error rate ϵ .

Average-case bound: Even typical (rather than adversarial) circuits exhibit exponential overhead requirements.

Comparison to error correction: The bounds show that full error correction overhead is necessary for arbitrary-depth computation—mitigation cannot circumvent this for general circuits.

These results clarify that error mitigation is not a general-purpose solution but a tool for specific regimes where circuits are shallow enough that exponential overhead remains manageable.

8.5.3 Practical Implications

The fundamental limits imply:

- **Depth limits:** QEM is effective for circuits with depth $d \lesssim 1/\epsilon$ where overhead remains practical
- **No free lunch:** Achieving arbitrary precision with fixed depth requires exponential resources
- **Sweet spot:** A “practical window” exists where QEM provides significant benefit before overhead becomes prohibitive

- **Transition planning:** Long-term quantum computing requires error correction; mitigation is a bridge technology

8.6 Combining Mitigation Techniques

State-of-the-art error mitigation combines multiple techniques targeting different error sources:

8.6.1 Standard Practice Pipeline

A typical mitigation pipeline includes:

1. **Measurement error mitigation:** Always applied due to low overhead and significant benefit
2. **Symmetry projection:** Applied when problem symmetries exist
3. **ZNE or PEC:** Applied to address gate errors, choice depending on noise level and precision requirements
4. **Post-processing:** Additional classical processing to improve estimates

8.6.2 Unified Frameworks

Recent work develops unified perspectives on error mitigation. Techniques that appear different (ZNE, PEC, virtual distillation) can be understood as different strategies within a common framework based on quasi-probability decomposition or channel inversion.

This unification enables:

- Systematic comparison of technique efficiency
- Identification of optimal combined strategies
- Understanding of fundamental limits across techniques

8.7 NISQ Hardware Platforms

Different physical platforms implement NISQ quantum computing with distinct characteristics relevant for quantum machine learning.

8.7.1 Superconducting Qubits

Superconducting qubits, the leading platform by market adoption (IBM, Google), use Josephson junctions cooled to millikelvin temperatures:

Advantages:

- Fast gates (10-100 ns two-qubit gates)
- Scalable fabrication using semiconductor techniques
- Mature control systems and software ecosystems
- Rapidly improving error rates

Challenges:

- Short coherence times (50-200 μ s)
- Limited connectivity (typically nearest-neighbor on 2D grid)
- Dilution refrigerator infrastructure requirements
- Frequency crowding at scale

For quantum ML, superconducting platforms offer rapid iteration and good software support but require shallow circuits due to coherence limits.

8.7.2 Trapped Ions

Trapped ion systems (IonQ, Quantinuum) use electromagnetic fields to confine individual atoms:

Advantages:

- Long coherence times (seconds to minutes)
- High-fidelity gates (>99.9% demonstrated)

- All-to-all connectivity through ion shuttling or photonic links
- Identical qubits (same atomic species)

Challenges:

- Slower gates (microseconds for two-qubit operations)
- Scaling challenges for large numbers (ion heating, control complexity)
- Lower qubit counts than superconducting (tens vs. hundreds+)

For quantum ML, trapped ions enable deeper circuits per coherence limit but slower execution. All-to-all connectivity simplifies circuit compilation.

8.7.3 Neutral Atoms

Neutral atom arrays (QuEra, Pasqal) trap atoms in optical tweezers:

Advantages:

- Large qubit counts (hundreds demonstrated)
- Reconfigurable connectivity
- Long coherence times
- Native multi-qubit gates

Challenges:

- Lower gate fidelities than ions
- Slower operation
- Less mature software ecosystem

Neutral atoms excel for simulation-type applications with geometric structure.

8.7.4 Photonic Systems

Photonic quantum computing (Xanadu) uses photons as information carriers:

Advantages:

- Room temperature operation
- Natural for communication and networking
- High-dimensional encoding possible (continuous variables)

Challenges:

- Probabilistic gates requiring heralding
- Photon loss
- Different computational model (often Gaussian boson sampling)

Photonic systems have demonstrated quantum advantage for specific sampling tasks and may enable quantum ML for naturally photonic data.

8.8 Practical NISQ Quantum Machine Learning

8.8.1 Algorithm Design Principles

Successful NISQ quantum ML algorithms incorporate hardware constraints from the design stage:

Shallow circuits: Limit depth to approximately $1/\epsilon$ for error rate ϵ , typically tens to hundreds of gates.

Hardware-native gates: Use native gate sets to minimize decomposition overhead. Compilation to non-native gates multiplies effective depth.

Connectivity-aware: Design ansatzes respecting hardware connectivity to avoid SWAP gate overhead. On superconducting hardware, linear or grid connectivity is natural; all-to-all connectivity is expensive.

Noise-aware training: Incorporate noise effects in training simulation when possible. Noise-aware optimization can find parameters robust to hardware errors.

Error mitigation integration: Design circuits amenable to mitigation. Circuits with symmetry enable symmetry-based mitigation; circuits with repeated structure enable learning-based approaches.

8.8.2 Resource Estimation

Practical NISQ quantum ML requires careful resource accounting:

Circuit executions: Each expectation value requires many measurements (shots).

Variational training requires many circuit evaluations (parameters \times gradient samples \times optimization iterations).

Classical processing: Error mitigation, optimization, and post-processing require classical computation. For learning-based mitigation, training data generation adds overhead.

Calibration: Hardware calibration consumes time and must be repeated as drift occurs.

Queue time: Cloud quantum computing involves queue delays that dominate wall-clock time.

Total resources often exceed naive estimates by orders of magnitude, affecting practical utility assessments.

8.8.3 Experimental Demonstrations

Notable NISQ quantum ML demonstrations include:

IBM 127-qubit utility: Kim et al. demonstrated quantum circuits producing results that cannot be reliably computed classically, using error mitigation [70]. While not directly ML-focused, the work establishes utility-scale quantum computation.

100+ qubit QAOA: Sack and Egger demonstrated error-mitigated QAOA on 100+ qubit graphs [22], showing optimization-relevant quantum ML scales to practical sizes.

VQE chemical accuracy: Multiple groups achieved chemical accuracy for small molecules using VQE with error mitigation on real hardware [34].

Quantum kernel classification: Demonstrations of quantum kernel methods achieving competitive accuracy on benchmark datasets with careful error handling.

8.9 Transition to Fault-Tolerant Computing

8.9.1 Early Fault-Tolerant Systems

As error rates approach fault-tolerant thresholds (approximately 10^{-3} for surface codes), hybrid approaches emerge:

Partial error correction: Use error correction for critical circuit sections while leaving others unprotected.

Error detection without correction: Detect errors and post-select valid runs without full correction overhead.

Concatenated mitigation-correction: Apply mitigation to residual errors after limited correction.

These approaches may provide an intermediate regime with enhanced capability before full fault tolerance.

8.9.2 Implications for Quantum ML

The fault-tolerant transition changes the quantum ML landscape:

Deeper circuits: Error correction enables circuits far deeper than NISQ limits, accessing more expressive models.

Different algorithms: Fault-tolerant algorithms (HHL, quantum gradient descent) become practical, potentially offering advantages inaccessible to variational approaches.

Higher overhead: Error correction overhead (thousands of physical qubits per logical qubit) means fewer effective qubits for computation. Small fault-tolerant systems may have similar effective size to large NISQ systems.

Timeline uncertainty: Achieving practical fault tolerance remains 5-10+ years away on most projections. NISQ approaches remain relevant in the interim.

8.10 Open Questions

Several fundamental questions remain:

Optimal mitigation strategy selection: Given a circuit and hardware, how should optimal mitigation techniques be automatically selected? Current practice relies on heuristics and expert judgment.

Structure exploitation: Can problem structure reduce QEM overhead below worst-case bounds? Structured problems may have favorable mitigation properties.

Noise model accuracy: How accurately must noise be characterized for effective mitigation? Mismatched models can degrade rather than improve results.

Scaling behavior: How does practical QEM effectiveness scale with qubit count? Large-scale demonstrations are limited.

Integration with error correction: How will QEM integrate with partial error correction in early fault-tolerant systems?

8.11 Summary

The NISQ era defines the practical context for near-term quantum machine learning. Devices with tens to thousands of qubits and 0.1-1% error rates enable meaningful quantum computation but require specialized approaches to manage noise.

Error sources including gate errors, decoherence, measurement mistakes, and crosstalk accumulate with circuit complexity, fundamentally limiting executable circuit depth. Noise-induced barren plateaus compound structural trainability challenges, further constraining viable quantum ML architectures.

Quantum error mitigation techniques—zero-noise extrapolation, probabilistic error cancellation, measurement mitigation, symmetry projection, and learning-based methods—extend the practical reach of NISQ devices by factors of 5-10x in effective circuit depth. Combining techniques addresses multiple error sources simultaneously.

However, fundamental limits establish that QEM overhead scales exponentially with circuit depth times noise rate. Error mitigation is not a general-purpose solution but a tool for a specific practical regime where circuits are shallow enough that overhead remains manageable.

Different hardware platforms (superconducting, trapped ion, neutral atom, photonic) offer distinct tradeoffs for quantum ML. Superconducting systems provide fast iteration with many qubits; trapped ions offer higher fidelity with slower operation; neutral atoms enable large arrays with geometric structure.

The transition to fault-tolerant computing will eventually enable deeper circuits and different algorithms, but remains years away. NISQ quantum machine learning represents the practical frontier, requiring careful algorithm-hardware co-design, comprehensive error mitigation, and honest assessment of capabilities and limitations.

9 Applications and Emerging Directions

The ultimate objective of quantum machine learning research is developing practical applications where quantum computing provides genuine value for artificial intelligence tasks. While theoretical developments have been substantial and algorithmic frameworks increasingly mature, translating these advances into real-world applications poses distinct challenges: problem encoding overhead, data access constraints, hardware limitations, and competition with rapidly advancing classical methods. This section examines current application domains, evaluates their near-term prospects, and projects future directions as quantum hardware evolves toward fault-tolerant operation [2, 18].

The application landscape for quantum AI divides broadly into domains with natural quantum structure—where quantum advantages may be inherent—and domains where quantum computing is applied to classically-defined problems with less clear advantage paths. Quantum chemistry and materials science exemplify the former:

molecular systems are fundamentally quantum mechanical, and quantum computers naturally represent their complexity. Optimization, image recognition, and natural language processing exemplify the latter: these are classical problems where quantum approaches must demonstrate advantage against sophisticated classical alternatives.

9.1 Quantum Chemistry and Drug Discovery

9.1.1 Fundamental Motivation

Quantum chemistry represents the most compelling near-term application domain for quantum computing, and by extension, quantum machine learning [24, 71]. The connection is fundamental: molecules are quantum mechanical systems governed by the Schrödinger equation, and their properties emerge from quantum correlations among electrons that grow exponentially difficult to simulate classically.

For a molecule with N electrons in M orbitals, the exact quantum state requires specifying $\binom{M}{N}$ amplitudes in the configuration interaction expansion, scaling exponentially with system size. Classical algorithms (density functional theory, coupled cluster) achieve polynomial scaling through approximations, but these approximations break down for strongly correlated systems common in transition metal chemistry, enzyme active sites, and unconventional materials.

Quantum computers address this exponential scaling directly: n qubits can represent 2^n configurations, enabling polynomial-resource simulation of quantum chemistry within controlled error bounds. The variational quantum eigensolver (VQE) exploits this representation for ground state energy estimation, the most fundamental quantum chemistry task.

9.1.2 Variational Quantum Eigensolver

VQE has become the primary quantum chemistry approach on NISQ hardware [34]. The algorithm:

1. Encodes the molecular Hamiltonian H in qubit operators via Jordan-Wigner or Bravyi-Kitaev transformation
2. Prepares a parameterized ansatz state $|\psi(\theta)\rangle$ (typically unitary coupled cluster or hardware-efficient)
3. Measures the energy expectation value $E(\theta) = \langle\psi(\theta)|H|\psi(\theta)\rangle$
4. Optimizes parameters classically to minimize energy
5. The final energy approximates the ground state energy

Demonstrations have achieved chemical accuracy (1 kcal/mol, approximately 1.6 mHa) for small molecules:

- H_2 : The simplest molecule, demonstrated across multiple platforms
- LiH , BeH_2 : Demonstrated on 6-8 qubits with error mitigation
- H_2O : Demonstrated at the limit of current capability

However, these molecules are easily simulated classically. Practical quantum advantage requires addressing larger systems: drug molecules with dozens of atoms, enzyme active sites with transition metals, or materials with extended correlation.

9.1.3 Quantum Machine Learning for Chemistry

Beyond VQE, quantum machine learning techniques apply to molecular property prediction—relating molecular structure to physical, chemical, or biological properties:

Quantum neural networks for QSAR: Sagingalieva et al. demonstrated hybrid quantum-classical neural networks for drug response prediction, achieving 15% improvement over classical baselines on small datasets [42]. The quantum component provides feature extraction that may capture chemical information inaccessible to classical encodings.

Quantum kernels for molecular similarity: Quantum feature maps can define molecular similarity kernels potentially capturing quantum effects. Virtual screening

for drug discovery relies on similarity metrics; quantum kernels may provide novel similarity notions.

Quantum-enhanced molecular generation: Generative models for de novo drug design could leverage quantum sampling to explore chemical space. Born machines and quantum GANs have been proposed for molecular generation, though demonstrations remain limited.

9.1.4 Drug Discovery Pipeline Integration

The pharmaceutical industry has identified quantum computing as a strategic technology with specific integration points:

Lead optimization: Once a promising drug candidate is identified, quantum simulation could optimize its properties (binding affinity, selectivity, ADMET properties) more accurately than classical methods.

Binding affinity prediction: Accurately predicting protein-ligand binding strength remains challenging classically. Quantum approaches may improve accuracy for strongly correlated binding sites.

Retrosynthesis planning: Finding synthetic routes to target molecules is a combinatorial optimization problem amenable to quantum approaches.

Major pharmaceutical companies (Roche, Boehringer Ingelheim, Merck) have established quantum computing research programs, though production deployment remains years away.

9.1.5 Challenges and Timeline

Significant challenges remain:

System size scaling: Interesting drug molecules have 50-100+ atoms requiring hundreds of qubits. Current demonstrations address systems with fewer than 10 atoms.

Encoding overhead: Transforming molecular Hamiltonians to qubit operators introduces significant overhead. For N spin-orbitals, $O(N^4)$ terms arise, requiring many measurements.

Classical competition: Tensor network methods and machine learning potentials continue advancing, improving classical capabilities for strongly correlated systems.

Accuracy requirements: Drug discovery requires 1 kcal/mol accuracy. Achieving this with error-mitigated NISQ circuits is challenging.

Timeline projections suggest:

- Near-term (1-3 years): Enhanced property prediction for small molecules
- Medium-term (3-7 years): Integration into drug discovery pipelines for specific tasks
- Long-term (7+ years): Routine quantum simulation for large biomolecules

9.2 Optimization and Finance

9.2.1 Combinatorial Optimization

Many AI applications involve combinatorial optimization: finding optimal configurations from discrete possibility spaces. Machine learning training, feature selection, neural architecture search, and inference in graphical models all involve optimization. Quantum approaches (QAOA, quantum annealing) aim to accelerate these searches.

Applications explored include:

Feature selection: Selecting optimal feature subsets for classification can be formulated as QUBO and solved via quantum annealing or QAOA.

Hyperparameter tuning: Discrete hyperparameter optimization maps to combinatorial problems.

Graph neural network training: Certain GNN optimization tasks have been mapped to quantum systems.

However, as discussed in Section 5, quantum optimization has not demonstrated consistent advantage over classical heuristics on benchmark problems.

9.2.2 Financial Applications

Finance represents a major target domain for quantum optimization and sampling:

Portfolio optimization: The classic mean-variance optimization—selecting asset allocations minimizing risk for target return—formulates naturally as QUBO. Demonstrations on D-Wave and gate-based systems show feasibility for small portfolios (tens of assets) [56, 72]. Production portfolios with thousands of assets remain beyond current capability.

Risk analysis: Monte Carlo simulation underlies value-at-risk, expected shortfall, and scenario analysis. Quantum speedup for Monte Carlo (via amplitude estimation) could accelerate these calculations, though end-to-end advantage remains undemonstrated.

Option pricing: Pricing complex derivatives requires high-dimensional integration. Quantum approaches could provide speedup, though classical methods remain competitive.

Fraud detection: Anomaly detection for fraud identification could leverage quantum machine learning, though classical approaches are highly developed.

9.2.3 Financial Industry Status

Major financial institutions actively research quantum computing:

- JPMorgan: Quantum research lab, publications on optimization and option pricing
- Goldman Sachs: QC research team, Monte Carlo and optimization applications
- BBVA: European bank with active quantum research program
- PayPal: Fraud detection and optimization exploration

Despite this activity, no production quantum application exists in finance. Industry consensus suggests fault-tolerant quantum computing (5-10+ years) will be required for practical advantage. Near-term applications focus on learning and preparation rather than deployment.

9.2.4 Challenges

Financial applications face specific challenges:

Problem scale: Real portfolios involve thousands of assets; current quantum systems handle tens effectively.

Speed requirements: High-frequency trading requires microsecond latency, incompatible with current quantum access models involving queue times and classical communication.

Data privacy: Financial data is sensitive; quantum cloud computing raises security and compliance concerns.

Regulatory environment: Financial regulations require auditability and explainability that quantum systems currently lack.

Classical competition: Financial firms have invested heavily in classical optimization; the bar for quantum improvement is high.

9.3 Computer Vision and Image Processing

9.3.1 Quantum Approaches to Vision

Computer vision—processing and understanding visual information—has been explored through several quantum approaches:

Quantum convolutional neural networks: Quanvolutional layers replace classical convolutions with quantum operations. The quantum layer processes image patches through parameterized quantum circuits, producing feature maps for subsequent classical processing [41]. Demonstrations show competitive accuracy with reduced parameter counts on small datasets.

Quantum image encoding: Various schemes encode images into quantum states:

- FRQI (Flexible Representation of Quantum Images): Encodes pixel values as rotation angles

- NEQR (Novel Enhanced Quantum Representation): Encodes pixels as basis states
- Amplitude encoding: Pixel values as state amplitudes

These encodings enable quantum image processing but introduce significant overhead.

Quantum image classification: Variational circuits classify encoded images. Demonstrations on MNIST subsets achieve high accuracy but do not exceed classical performance.

9.3.2 Adversarial Robustness

Recent work suggests potential quantum advantages in adversarial robustness [73]. Classical neural networks are notoriously vulnerable to adversarial perturbations—small input changes causing misclassification. Quantum classifiers may exhibit different vulnerability profiles:

- Quantum noise may mask adversarial perturbations
- Quantum measurement collapse may disrupt adversarial attacks
- Quantum feature spaces may be geometrically different from classical

West et al. demonstrated that quantum classifiers showed enhanced robustness to certain attack types, though comprehensive assessment remains ongoing.

9.3.3 Assessment

Current quantum vision approaches do not outperform classical convolutional neural networks on standard benchmarks. Classical CNNs benefit from:

- Massive scale (billions of parameters)
- Extensive architecture optimization
- Specialized hardware (GPUs, TPUs)
- Large training datasets

Potential quantum advantages may emerge for:

- Processing quantum sensor data (quantum imaging, quantum radar)
- Privacy-preserving image classification
- Specific image types with quantum structure

Standard image recognition is unlikely to be an early quantum advantage domain.

9.4 Natural Language Processing

9.4.1 Quantum NLP Approaches

Natural language processing presents both opportunities and challenges for quantum approaches:

Compositional distributional semantics: The DisCoCat framework represents linguistic structure as tensor networks, with words as vectors and grammatical rules as tensor operations. This representation maps naturally to quantum processes, enabling grammar-aware quantum NLP.

Quantum language models: Tensor network representations of probability distributions connect to quantum states. Quantum language models could represent syntactic structure more efficiently than classical alternatives.

Quantum text classification: Standard quantum classification applied to text embeddings. Demonstrations on sentiment analysis and topic classification show feasibility without clear advantage.

9.4.2 Classical Competition

NLP is dominated by transformer architectures (GPT, BERT, LLaMA) that achieve remarkable performance through scale:

- Billions to trillions of parameters
- Training on vast text corpora

- Sophisticated attention mechanisms

Quantum approaches cannot currently compete with these systems for general NLP tasks. The gap between small quantum demonstrations and state-of-the-art classical NLP is enormous.

9.4.3 Potential Niches

Quantum NLP may contribute to:

- Linguistically-structured tasks leveraging DisCoCat
- Reasoning tasks where compositional structure matters
- Quantum-native text (quantum program descriptions, quantum physics papers)

General-purpose quantum NLP remains a distant prospect.

9.5 Generative Modeling

9.5.1 Quantum Generative Approaches

Generative modeling—learning to generate samples from data distributions—may be a natural quantum strength due to native sampling capabilities:

Quantum GANs: Adversarial training with quantum generators and classical or quantum discriminators. The generator produces quantum states; measurement yields samples. Demonstrations have generated simple images (handwritten digits) and molecular structures [74, 75].

Born machines: Parameterized quantum circuits define probability distributions through Born’s rule: $p(x) = |\langle x|\psi(\theta)\rangle|^2$. Training adjusts parameters to match target distributions. Born machines naturally implement probability distributions difficult to sample classically.

Quantum Boltzmann machines: Energy-based models implemented on quantum annealing hardware. D-Wave systems naturally sample from Boltzmann distributions, enabling quantum-native training of restricted Boltzmann machines.

9.5.2 Potential Advantages

Quantum generative models may offer advantages for:

Distributions with quantum structure: If target distributions arise from quantum processes, quantum generators naturally match them.

High-dimensional sampling: Quantum circuits can efficiently sample from distributions hard to sample classically (certain graph distributions, quantum circuit outputs).

Mode coverage: Classical generators (GANs) often suffer from mode collapse— failing to cover the full data distribution. Quantum sampling mechanisms may have different mode coverage properties.

Molecular generation: Drug design requires generating novel molecules with desired properties. Quantum generative models for molecular graphs are actively explored.

9.5.3 Current Limitations

Current demonstrations remain limited:

- Small system sizes (typically < 10 qubits)
- Simple target distributions (handwritten digits, toy molecules)
- Training challenges (barren plateaus affect generative training)
- Unclear advantage over classical generative models (diffusion models, flow-based)

Figure 10 summarizes the quantum AI application landscape.

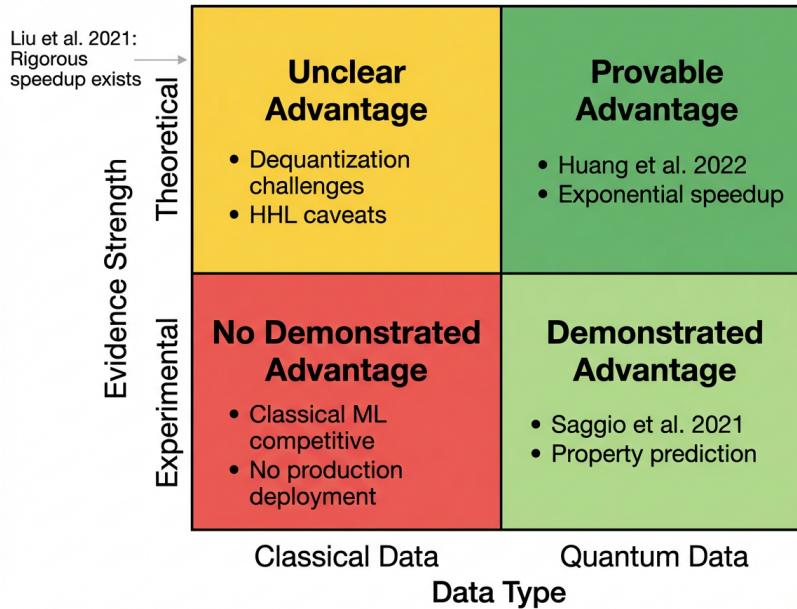


Fig. 10 Quantum AI application domains and their readiness levels. Quantum chemistry/drug discovery shows the highest near-term potential due to natural quantum structure. Optimization and generative modeling have medium-term prospects with active research. Computer vision and NLP face strong classical competition with unclear quantum advantage paths. Finance requires fault-tolerant systems for practical impact. The vertical axis indicates technology readiness level; horizontal axis indicates estimated years to practical advantage.

9.6 Reinforcement Learning

9.6.1 Quantum RL Paradigms

Reinforcement learning—agents learning through environment interaction—offers unique quantum opportunities:

Quantum exploration: Agents using quantum superposition can explore multiple action sequences simultaneously, potentially accelerating environment learning. Saggio et al. demonstrated quadratic speedup in learning environment models using photonic quantum processing [17].

Variational quantum policy: Quantum circuits as policy functions, mapping states to action probabilities [14]. The circuit parameters are trained via policy gradient methods.

Quantum value estimation: Quantum circuits estimating state or action values for Q-learning approaches.

Quantum environment simulation: When environments are quantum systems (quantum control, chemistry), quantum processors naturally simulate them.

9.6.2 Demonstrated Advantages

The Saggio et al. demonstration represents the clearest quantum RL advantage to date:

- Photonic quantum processor with 8 modes
- Navigation task with partial observability
- Quadratic speedup in environment learning
- First experimental validation of quantum RL theory

The speedup is analogous to Grover's—quadratic rather than exponential—but demonstrates that quantum advantage in learning is physically achievable.

9.6.3 Open Challenges

Quantum RL development faces challenges:

- Scaling to complex environments (current demos are toy problems)
- Credit assignment in hybrid quantum-classical systems
- Sample efficiency compared to classical RL (which has also advanced rapidly)
- Hardware requirements for real-time interaction

9.7 Future Directions

9.7.1 Near-Term Roadmap (1-3 Years)

Near-term developments will likely include:

Quantum utility demonstrations: Problems where quantum computation provides measurable benefit—not necessarily speedup, but enhanced capability, accuracy, or insight. IBM’s 127-qubit utility demonstration provides a template.

Hybrid integration frameworks: Software frameworks enabling seamless quantum-classical ML pipelines. PennyLane, Qiskit Machine Learning, and TensorFlow Quantum continue maturing.

Specialized applications: Focus on narrow domains where quantum advantages are clearest—quantum chemistry for specific molecule classes, quantum data applications, structured optimization problems.

Benchmark standardization: Development of fair benchmarks for quantum ML claims, enabling rigorous comparison with classical alternatives.

9.7.2 Medium-Term Prospects (3-7 Years)

With early fault-tolerant systems (100-1000 logical qubits), new possibilities emerge:

Quantum linear algebra: HHL and related algorithms become practical, enabling quantum speedup for linear regression, principal component analysis, and recommendation systems under appropriate conditions.

Larger-scale chemistry: Molecules with 50-100+ atoms become accessible, enabling practically relevant quantum chemistry for drug discovery and materials design.

Practical optimization: Portfolio optimization, logistics, and scheduling at meaningful scale, though classical competition remains strong.

Quantum data centers: Integration of quantum processors into classical data center infrastructure, enabling routine quantum subroutine calls.

9.7.3 Long-Term Vision (7+ Years)

Full fault-tolerant quantum computing enables:

Unlimited circuit depth: Algorithms no longer constrained by noise accumulation, enabling complex quantum computations.

Novel algorithmic paradigms: Applications not yet conceived—quantum computing for AI may evolve in unexpected directions as capability increases.

Quantum-native AI: AI systems designed from the ground up for quantum substrates, rather than quantum versions of classical approaches.

Clear advantage domains: Definitive identification of problems where quantum approaches are the optimal solution.

9.8 Cross-Cutting Considerations

9.8.1 Benchmarking and Evaluation

Rigorous evaluation of quantum AI applications requires:

Fair comparison protocols: Identical problems, equivalent resources (time, energy, cost), comparable precision requirements.

End-to-end accounting: Including data loading, error mitigation overhead, classical post-processing, and output extraction.

Strong classical baselines: Comparison against state-of-the-art classical methods, not strawman alternatives.

Statistical rigor: Multiple problem instances, confidence intervals, and reproducibility requirements.

9.8.2 Integration Challenges

Deploying quantum AI applications requires addressing:

Data movement: Transferring data between classical and quantum systems introduces latency and potential bottlenecks.

Hybrid workflow management: Coordinating quantum and classical computation within ML pipelines.

Error handling: Gracefully managing quantum computation failures within larger applications.

Tooling maturity: Software tools for quantum ML remain less mature than classical ML frameworks.

9.8.3 Economic Considerations

Practical quantum AI must be economically justified:

Hardware costs: Quantum computer access is expensive; applications must provide sufficient value to justify costs.

Time-to-solution: Total time including queue waits, compilation, execution, and post-processing must compare favorably to classical.

Energy efficiency: Quantum systems have significant energy requirements (cryogenics for superconducting); energy costs matter at scale.

Expertise requirements: Quantum ML requires specialized skills currently in short supply.

9.9 Summary

The application landscape for quantum AI ranges from compelling near-term prospects to speculative long-term visions. Quantum chemistry and drug discovery represent the most promising near-term domain, leveraging the fundamental connection between quantum computation and molecular simulation. Demonstrations of VQE for small molecules and hybrid quantum-classical property prediction establish feasibility, while practical impact awaits hardware scaling.

Optimization and finance applications are actively explored but face strong classical competition and scale challenges. Current demonstrations remain proof-of-concept; production deployment requires fault-tolerant systems. Computer vision and NLP face enormous classical dominance from deep learning; quantum advantages, if they exist, likely lie in specialized niches rather than general capability.

Generative modeling and reinforcement learning show intriguing quantum possibilities exploiting native sampling and exploration capabilities. Experimental demonstrations validate theoretical predictions, though scaling to practical problems remains challenging.

The path from current demonstrations to practical applications involves:

- Hardware scaling toward hundreds of high-quality qubits
- Algorithm development addressing specific application requirements
- Software framework maturation enabling quantum-classical integration
- Benchmark standardization enabling rigorous evaluation
- Economic justification of quantum approaches versus classical alternatives

No production quantum AI application exists today. The honest assessment is that practical quantum AI impact is years away—1-3 years for specialized utility demonstrations, 3-7 years for early practical applications in chemistry and optimization, and 7+ years for broad quantum AI deployment. The trajectory is positive but expectations should be calibrated to physical and algorithmic realities.

10 Discussion

This review has examined the current state of quantum computing for artificial intelligence across seven major themes: variational quantum machine learning, quantum kernel methods, quantum optimization, barren plateaus and trainability, quantum advantage and speedup, NISQ-era constraints and error mitigation, and applications. Several cross-cutting insights emerge from this analysis that illuminate both the promise and the challenges facing the field.

10.1 The Central Tradeoff: Trainability, Expressivity, and Simulability

Perhaps the most significant insight emerging from recent research is the fundamental tension between trainability, expressivity, and classical simulability. Variational quantum circuits must be expressive enough to represent complex functions, trainable via gradient-based optimization, and classically intractable to provide quantum advantage. The trainability-simulability tradeoff established by Cerezo et al. demonstrates that achieving two of these properties may preclude the third [11].

Circuits with large dynamical Lie algebras are expressive and potentially hard to simulate but suffer from barren plateaus that prevent training. Circuits with small dynamical Lie algebras avoid barren plateaus but can be efficiently simulated classically, negating quantum advantage. This tradeoff is not merely a technical obstacle but may represent a fundamental feature of quantum computational advantage.

Resolution may come through problem-specific structure. Ansatzes designed for particular applications may break the general tradeoff by exploiting domain knowledge to maintain trainability while remaining classically intractable for the specific target problem. Quantum chemistry with symmetry-preserving circuits provides a template: encoding physical constraints reduces the dynamical Lie algebra (enabling training) while the resulting computation may still be hard for classical algorithms lacking equivalent structure exploitation.

10.2 The Moving Target of Classical Competition

A recurring theme across application domains is the strength of classical competition. Quantum machine learning does not develop in isolation; classical methods continue advancing rapidly. Several patterns emerge:

Dequantization: Tang’s quantum-inspired classical algorithms demonstrated that many presumed quantum speedups relied on unfair input/output assumptions.

When classical algorithms receive similar data access, exponential speedups often vanish. This pattern suggests caution in claiming quantum advantage without careful resource accounting.

Classical scaling: Deep learning has achieved remarkable capability through scaling to billions of parameters trained on massive datasets. Quantum approaches with tens of qubits cannot compete on parameter count; advantage must come from different sources than scale alone.

Algorithmic innovation: Classical algorithm development continues, including tensor network methods that efficiently simulate certain quantum circuits. The boundary of classical tractability is not fixed.

These observations suggest that quantum AI advantage requires demonstrating value beyond what classical methods can achieve with equivalent resources—not merely matching classical performance with quantum implementations.

10.3 Hardware Constraints Shape Algorithmic Design

NISQ hardware constraints pervasively influence quantum machine learning design.

The cascade of constraints includes:

1. Error rates limit useful circuit depth to approximately $1/\epsilon$ gates
2. Shallow circuits face expressivity limitations
3. Deep circuits encounter noise-induced barren plateaus
4. Error mitigation extends the practical regime but with exponential overhead limits
5. Qubit connectivity restricts efficient circuit compilation

These constraints do not merely limit performance—they fundamentally shape which algorithms are viable. The variational paradigm emerged specifically to work within NISQ limitations: hybrid quantum-classical approaches that minimize coherent quantum operations while leveraging classical optimization.

Practical quantum machine learning must be “noise-aware” from design inception, not adapted to noise after development. Algorithms designed for ideal systems typically fail when executed on real hardware. This requirement favors shallow, local architectures over deep, highly entangled circuits.

10.4 The Kernel Perspective Provides Theoretical Unity

Schuld’s demonstration that all supervised variational quantum models are mathematically equivalent to kernel methods provides theoretical unity to an otherwise fragmented field [10]. This kernel perspective clarifies that:

- VQC expressivity corresponds to kernel richness
- VQC trainability relates to kernel smoothness
- VQC advantage requires classically intractable kernels

The variational versus kernel distinction becomes an implementation choice rather than fundamental difference. Research can freely translate between perspectives, using kernel analysis to understand VQC properties while employing variational techniques to implement and train kernel models.

This unification also clarifies the advantage question: quantum kernels provide advantage if and only if they are classically intractable to evaluate or approximate. Feature map design thus becomes central to achieving quantum benefit.

10.5 Quantum Data Represents the Clearest Advantage

Regime

Across theoretical results and experimental demonstrations, the clearest quantum advantages emerge for learning from quantum data—data generated by quantum systems or experiments. Huang et al.’s work establishing exponential speedup for quantum property prediction exemplifies this regime [9].

For quantum data, advantage is robust because:

- Input is inherently quantum, avoiding classical data loading bottlenecks
- Classical computers face fundamental information-theoretic limits in representing quantum states
- Output (expectation values) is classical and efficiently verifiable
- No unfair access assumptions are required

In contrast, quantum advantage for classical data remains contested. Dequantization challenges many speedup claims, and no practical demonstration has established clear advantage for processing generic classical datasets.

This observation redirects the search for near-term quantum AI applications toward domains with natural quantum data: quantum chemistry, quantum materials, quantum sensing, and quantum device characterization. Applications involving purely classical data may require fault-tolerant quantum computing for clear advantage.

10.6 Error Mitigation Extends but Does Not Solve NISQ Limitations

Quantum error mitigation has matured from theoretical proposals to practical tools, enabling demonstrations at scales previously considered intractable. Techniques including zero-noise extrapolation, probabilistic error cancellation, and machine learning-based mitigation extend useful circuit depth by factors of 5-10.

However, fundamental limits constrain error mitigation capability. The exponential overhead scaling established by Takagi et al. and strengthened by Quek et al. shows that mitigation cannot enable arbitrary circuit depth [19, 69]. Error mitigation is a bridge technology for the NISQ era, not a substitute for fault-tolerant error correction.

The practical implication is a “mitigation window”—a regime of circuit complexity where mitigation provides significant benefit with manageable overhead. Beyond this window, circuits require either error correction or redesign to reduce depth. Current

estimates place this window at approximately 50-500 total gates depending on error rates.

10.7 Application Domains Have Different Advantage Profiles

The review of applications reveals that different domains have distinctly different quantum advantage profiles:

Quantum chemistry and drug discovery: Most promising near-term, leveraging natural quantum structure. Early demonstrations exist; practical impact awaits hardware scaling.

Optimization: Actively researched but without clear advantage over classical heuristics. QAOA and quantum annealing show competitive but not superior performance.

Finance: Requires scale and precision beyond current NISQ capability. Fault-tolerant systems likely needed for practical advantage.

Computer vision and NLP: Dominated by classical deep learning with no identified quantum advantage path. Quantum approaches may contribute to specialized niches rather than general capability.

Generative modeling and reinforcement learning: Intriguing possibilities exploiting quantum sampling and exploration, with early demonstrations validating theoretical predictions.

This domain diversity suggests that quantum AI will not provide universal acceleration but rather specific improvements in particular application areas. Research should prioritize domains with natural quantum structure rather than seeking general-purpose quantum enhancement.

10.8 The Timeline Reality

Honest assessment of timelines is essential. Despite significant progress, no production quantum AI application exists today. Timeline projections include:

Near-term (1-3 years): Quantum utility demonstrations for specialized tasks, enhanced property prediction for small molecules, proof-of-concept hybrid applications.

Medium-term (3-7 years): Early fault-tolerant systems enabling new algorithms, integration of quantum chemistry into drug discovery pipelines, practical optimization at modest scale.

Long-term (7+ years): Full fault-tolerant quantum computing, routine quantum-classical integration, definitive identification of quantum-advantage domains.

These timelines are uncertain and depend on hardware development trajectories that may accelerate or encounter unexpected obstacles. However, expectations should be calibrated to physical and algorithmic realities rather than marketing optimism.

10.9 Implications for Research Direction

The synthesized findings suggest several research priorities:

Focus on problem-specific approaches: Universal quantum machine learning appears limited by fundamental tradeoffs. Problem-specific ansatzes exploiting domain structure may break these tradeoffs for particular applications.

Rigorous benchmarking: The field requires standardized benchmarks with fair resource accounting against state-of-the-art classical methods. Claims of advantage must survive rigorous comparison.

Quantum data applications: Near-term advantage is clearest for quantum data. Applications involving quantum chemistry, sensing, and characterization deserve priority.

Noise-aware design: Algorithms should incorporate noise from design inception rather than adaptation. Understanding noise effects on trainability and advantage is essential.

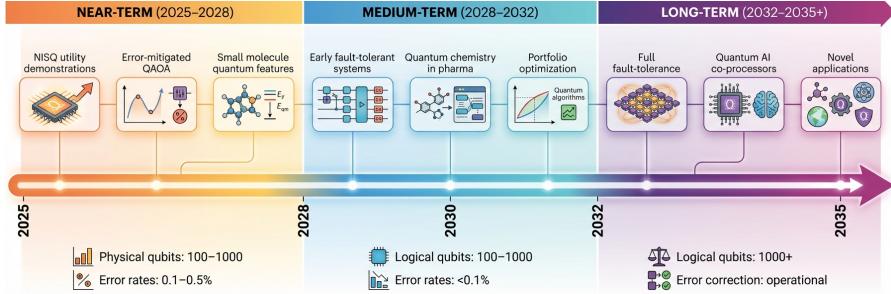


Fig. 11 Synthesis of cross-cutting themes in quantum machine learning. The central trainability-expressivity-simulability tradeoff connects theoretical limitations (barren plateaus, dequantization) with practical constraints (NISQ noise, circuit depth). The kernel perspective unifies variational and explicit kernel methods. Application success depends on matching problem domains to quantum capabilities, with quantum data applications showing clearest advantage. Error mitigation extends the practical regime but fundamental limits remain. Progress requires problem-specific approaches, rigorous benchmarking, and honest timeline assessment.

Integration strategies: Practical deployment requires quantum-classical integration frameworks, not quantum replacement of classical systems. Hybrid architectures that leverage quantum components within classical pipelines may provide the most practical near-term value.

Figure 11 summarizes the cross-cutting themes and their relationships.

10.10 Limitations of This Review

This review has several limitations that should inform interpretation:

Rapidly evolving field: Quantum computing advances quickly; some findings may be superseded by the time of publication.

Publication bias: Published work may emphasize positive results; failed approaches are underreported.

Hardware specificity: Results on specific platforms may not generalize across quantum technologies.

Classical baseline evolution: Classical methods continue improving; static comparisons become outdated.

Theoretical vs. practical gap: Many theoretical results assume conditions (perfect gates, unlimited coherence) not achieved in practice.

These limitations suggest that conclusions should be held with appropriate uncertainty and updated as the field progresses.

11 Outlook

The future trajectory of quantum computing for artificial intelligence depends on developments across multiple fronts: hardware capability, algorithmic innovation, theoretical understanding, and practical integration. This section projects the evolution of the field and identifies key milestones that will shape its development.

11.1 Hardware Evolution

11.1.1 Near-Term Hardware (2025-2028)

Current hardware trends project continued improvement in the near term:

Qubit counts: Systems with 1000+ physical qubits are available today; 5000-10000 qubit systems are projected within 3 years across multiple platforms.

Error rates: Two-qubit gate error rates have improved from approximately 1% (2019) to below 0.1% on leading platforms (2024). Continued improvement toward 10^{-4} is projected.

Coherence times: Superconducting qubit coherence has improved steadily; trapped ion and neutral atom platforms offer inherently longer coherence.

Connectivity: Hardware-specific topologies are expanding, with improved connectivity enabling more efficient circuit compilation.

These improvements will expand the useful circuit complexity achievable with error mitigation, enabling more sophisticated quantum machine learning demonstrations.

11.1.2 Early Fault-Tolerant Systems (2028-2035)

The transition to fault-tolerant quantum computing represents the most significant upcoming milestone:

Threshold crossing: Leading platforms are approaching the error thresholds required for surface code error correction (approximately 10^{-3}). Google’s Willow processor has demonstrated below-threshold operation.

Logical qubits: First systems with tens of logical qubits are projected within 5-7 years, expanding to hundreds of logical qubits over the following decade.

New algorithmic regimes: Fault-tolerant operation enables algorithms requiring deep circuits, including HHL and quantum linear algebra methods currently impractical on NISQ devices.

Hybrid operation: Early fault-tolerant systems will likely combine limited error correction with mitigation, exploiting both approaches.

11.1.3 Full Fault-Tolerance (2035+)

Full fault-tolerant quantum computing with millions of physical qubits remains a long-term goal:

Universal computation: Arbitrary circuit depth with controlled error enables the full power of quantum algorithms.

New applications: Applications not yet conceived will become possible as hardware capability removes current constraints.

Integration: Quantum processors may become standard co-processors in data center infrastructure.

Timeline uncertainty is substantial; these projections may prove optimistic or conservative depending on technological breakthroughs and engineering challenges.

11.2 Algorithmic Development

11.2.1 Near-Term Algorithms

Algorithmic research will address current limitations:

Trainability solutions: Methods to circumvent barren plateaus while maintaining expressivity and advantage remain a primary research focus. Problem-specific ansatzes, layer-wise training, and noise-aware design will continue developing.

Error mitigation advances: Learning-based mitigation, automated strategy selection, and integration with partial error correction will extend the practical NISQ regime.

Kernel methods: Efficient kernel estimation and geometric understanding of quantum feature spaces will clarify advantage conditions.

Hybrid architectures: Quantum-classical algorithms that optimally partition computation between quantum and classical processors.

11.2.2 Fault-Tolerant Algorithms

With error correction, new algorithmic possibilities emerge:

Quantum linear algebra: HHL and variants become practical, enabling quantum speedup for linear regression, PCA, and related methods under appropriate conditions.

Quantum simulation: Long-time quantum dynamics simulation for chemistry and materials becomes accessible.

Quantum search enhancements: Grover-based algorithms and amplitude amplification provide quadratic speedups for various optimization and search tasks.

Novel paradigms: Algorithms not yet conceived may emerge as hardware capability removes current constraints.

11.3 Theoretical Progress

Key theoretical questions requiring resolution:

Trainability-simulability resolution: Can the fundamental tradeoff be broken for useful problem classes? Characterizing the conditions under which trainable, expressive, classically-hard circuits exist is essential.

Advantage characterization: For which problems do quantum computers provide provable advantage? Beyond constructed examples, identifying natural advantage domains remains critical.

Generalization theory: Rigorous understanding of quantum machine learning generalization, including sample complexity and noise effects.

Complexity connections: Relating quantum machine learning capability to computational complexity theory, clarifying fundamental limits.

11.4 Application Roadmap

11.4.1 Quantum Chemistry and Drug Discovery

The most mature near-term application trajectory:

2025-2027: Enhanced property prediction for small molecules using hybrid quantum-classical models. Validation on pharmaceutically relevant systems.

2028-2030: Integration into drug discovery pipelines for specific tasks (binding affinity, lead optimization). Demonstration of value beyond classical alternatives.

2030-2035: Routine use of quantum simulation for molecular design. Extension to larger systems (proteins, materials).

2035+: Quantum-first molecular simulation for appropriate problems, with classical methods for others.

11.4.2 Optimization

A more uncertain trajectory:

2025-2028: Continued research on QAOA and hybrid optimization. Identification of problem classes with quantum benefit.

2028-2032: Practical demonstrations if favorable problem classes exist. Integration with classical optimization frameworks.

2032+: Potentially routine use for specific optimization problems if advantage materializes.

The optimization trajectory depends heavily on whether problem structures exist where quantum approaches outperform classical alternatives.

11.4.3 Finance

Long-term potential with significant uncertainty:

2025-2030: Research and preparation. Hybrid approaches for small-scale demonstrations.

2030-2040: With fault-tolerant systems, Monte Carlo acceleration and portfolio optimization at meaningful scale may become practical.

2040+: Routine quantum finance if hardware and algorithms mature sufficiently.

Regulatory, privacy, and latency requirements add constraints beyond pure computational considerations.

11.4.4 Other Domains

Generative modeling: Near-term potential for specialized distributions; broader impact depends on overcoming training challenges.

Reinforcement learning: Demonstrated speedups suggest continued development; practical impact depends on scaling to complex environments.

Computer vision and NLP: Long-term prospects remain unclear; classical methods are highly optimized.

11.5 Integration and Infrastructure

Practical deployment requires infrastructure development:

Software frameworks: Maturation of quantum ML frameworks (PennyLane, Qiskit ML, TensorFlow Quantum) with seamless classical integration.

Cloud services: Quantum computing as cloud service with improved accessibility, reduced latency, and enterprise integration.

Hybrid workflows: Tools for designing, debugging, and deploying quantum-classical hybrid applications.

Workforce development: Training programs to build quantum ML expertise in the broader ML community.

11.6 Key Milestones to Watch

Progress toward practical quantum AI can be tracked through specific milestones:

Algorithmic milestones:

- Practical method to break trainability-simulability tradeoff
- First natural problem with demonstrated quantum advantage for classical data
- Quantum ML achieving state-of-the-art on a mainstream benchmark

Hardware milestones:

- First system with 100+ logical qubits
- Error rates enabling 1000+ gate circuits with mitigation
- Quantum cloud latency competitive with local classical computation

Application milestones:

- First quantum chemistry calculation outperforming classical for a drug discovery task
- Optimization problem class where quantum consistently beats classical heuristics
- Production deployment of quantum ML in any domain

Achievement of these milestones would signal significant progress; their timeline remains uncertain.

11.7 Scenarios for the Field

The future may unfold along several possible trajectories:

Optimistic scenario: Hardware improves rapidly, fault tolerance arrives by 2030, and quantum ML achieves clear advantage for multiple applications. Quantum AI becomes a standard tool in computational chemistry, optimization, and specialized ML.

Baseline scenario: Hardware progress continues but more slowly; fault tolerance arrives around 2035. Quantum ML achieves advantage in narrow domains (quantum chemistry, quantum data) but not broadly. Impact is real but limited.

Pessimistic scenario: Hardware challenges prove more severe than expected; classical algorithms continue improving faster than quantum capability grows. Quantum ML remains primarily a research topic without production deployment.

The actual trajectory will likely contain elements of all scenarios, with domain-specific outcomes varying.

11.8 Recommendations

Based on the review findings, recommendations for different stakeholders:

For researchers:

- Focus on domain-specific rather than universal approaches
- Prioritize rigorous benchmarking against strong classical baselines
- Develop noise-aware algorithms from design inception
- Engage with classical ML community for fair comparison
- Consider full resource accounting in advantage claims

For practitioners:

- Monitor quantum chemistry domain for first practical applications
- Consider hybrid integration rather than quantum replacement

- Plan for 5-10 year timeline to production deployment
- Build quantum literacy in technical teams
- Identify potential quantum-native internal problems

For funding agencies:

- Prioritize fundamental research on advantage conditions
- Support rigorous benchmarking infrastructure
- Fund interdisciplinary collaborations (chemistry-quantum, ML-quantum)
- Invest in workforce development
- Maintain long-term perspective despite hype cycles

For the community:

- Establish benchmarking standards and repositories
- Create challenge problems for quantum ML
- Maintain honest dialogue on advantage claims
- Support theoretical-experimental collaboration
- Communicate realistic timelines to stakeholders

12 Conclusion

Quantum computing for artificial intelligence represents a field of substantial scientific interest and potential practical impact, tempered by fundamental challenges and realistic assessment of current capabilities. This comprehensive review has examined the landscape across variational quantum machine learning, quantum kernel methods, quantum optimization, trainability limitations, quantum advantage conditions, NISQ-era constraints, and application domains.

12.1 Principal Findings

The central thesis of this review is that quantum computing offers genuine potential for enhancing artificial intelligence, but this potential is conditional rather than universal. Several principal findings support this assessment:

Variational quantum machine learning provides a flexible framework for near-term quantum AI, with parameterized quantum circuits serving as trainable function approximators. The mathematical equivalence to kernel methods provides theoretical unity, while practical challenges of trainability and noise accumulation constrain viable architectures.

Quantum kernels offer the clearest theoretical foundation for quantum advantage in supervised learning. Provable speedups exist for constructed problems, and the kernel perspective clarifies that advantage requires feature maps yielding classically intractable kernels. The exponential concentration phenomenon poses challenges for generic quantum feature maps at scale.

Quantum optimization through QAOA and quantum annealing has not demonstrated consistent advantage over classical heuristics on benchmark problems. The practical frontier remains competitive with classical methods rather than clearly superior, though ongoing research may identify favorable problem classes.

Barren plateaus represent a fundamental challenge to scaling variational quantum algorithms. The trainability-simulability tradeoff suggests deep tensions between achieving trainable circuits, expressive representations, and quantum advantage. Problem-specific structures may provide escape routes for particular applications.

Quantum advantage for machine learning is conditional: robust for learning from quantum data, achievable for problems with cryptographic hardness structure, but elusive for generic classical data. Dequantization results have eliminated many early speedup claims; advantage requires careful end-to-end analysis.

NISQ hardware constraints shape the entire algorithmic landscape. Error rates, coherence times, and connectivity limitations restrict useful circuit complexity. Error mitigation extends the practical regime significantly but cannot substitute for fault-tolerant error correction for arbitrary computations.

Application domains differ markedly in their quantum advantage profiles. Quantum chemistry and drug discovery show the strongest near-term potential due to natural quantum structure. Finance and general optimization require fault-tolerant systems. Computer vision and NLP face dominant classical competition with unclear quantum advantage paths.

12.2 The State of the Field

As of this writing, quantum machine learning occupies a transitional position. Theoretical foundations are increasingly mature, with clear understanding of expressivity, trainability, and advantage conditions. Algorithmic frameworks exist for implementing quantum ML on available hardware. Experimental demonstrations have validated key theoretical predictions.

However, no production quantum AI application exists. Current demonstrations remain proof-of-concept, limited to small system sizes, and do not clearly outperform optimized classical alternatives on problems of practical interest. The gap between theoretical promise and practical impact remains substantial.

This situation is not unusual for emerging technologies. The path from scientific possibility to practical utility involves hardware maturation, algorithmic refinement, integration infrastructure, and identification of suitable application domains. Quantum machine learning is progressing along this path, but with timeline uncertainty.

12.3 What Has Been Established

Despite remaining challenges, significant progress has been made:

Theoretical clarity: The mathematical foundations of quantum machine learning are now well-understood. The kernel perspective, expressivity characterizations, barren plateau analysis, and advantage conditions provide rigorous frameworks for understanding quantum ML capability.

Algorithmic diversity: Multiple algorithmic approaches—variational circuits, explicit kernels, QAOA, hybrid architectures—offer different tradeoffs and suit different application contexts.

Hardware demonstrations: Real quantum hardware has executed quantum ML algorithms, validating theoretical predictions and identifying practical challenges.

Advantage results: Rigorous proofs of quantum speedup exist for specific problem classes, establishing that quantum advantage for machine learning is achievable in principle.

Application targeting: The most promising near-term application domains (quantum chemistry, quantum data processing) have been identified, directing research toward tractable targets.

12.4 What Remains Open

Critical questions remain unresolved:

Natural advantage domains: What naturally occurring machine learning problems exhibit quantum advantage? Beyond constructed examples, identifying practical applications with inherent quantum benefit is essential for real-world impact.

Trainability-simulability resolution: Can the fundamental tradeoff be broken for useful problem classes? This question determines whether variational quantum advantage is achievable in practice.

Scaling behavior: How do quantum ML algorithms perform as system size increases? Current demonstrations on small systems may not predict large-scale behavior.

Fault-tolerant algorithms: Which quantum ML algorithms become practical with error correction? The transition to fault-tolerant computing may enable qualitatively different approaches.

Classical competition evolution: Where will the classical-quantum boundary settle as both technologies advance? The moving target of classical capability complicates advantage claims.

12.5 Concluding Assessment

Quantum computing for artificial intelligence is neither the revolution sometimes proclaimed nor the impossibility sometimes suggested. It is a developing technology with genuine potential in specific domains, significant remaining challenges, and uncertain timelines. The scientific foundations are solid; the practical impact awaits hardware and algorithmic maturation.

For the near term (1-5 years), quantum ML will likely remain primarily a research endeavor, with specialized demonstrations but limited production deployment. Quantum chemistry applications may achieve earliest practical impact due to natural quantum advantage.

For the medium term (5-15 years), fault-tolerant quantum computing may expand the application landscape significantly. If hardware progress continues and algorithmic challenges are resolved, quantum ML could become a standard tool for appropriate problems.

For the long term, the full potential of quantum computing for AI remains uncertain. Optimistic projections envision quantum processors as routine AI accelerators; pessimistic assessments suggest classical methods will remain superior for most applications. The actual trajectory will emerge from continued research, development, and honest assessment.

This review aims to contribute to that honest assessment by providing comprehensive, balanced coverage of the field's state. Progress in quantum machine learning will come through rigorous research, fair benchmarking, and realistic expectations—not through hype or dismissal. The intersection of quantum computing and artificial intelligence represents a fascinating scientific frontier whose ultimate impact, while uncertain, justifies continued investigation.

Disclaimer

This review article was generated by artificial intelligence and should not be trusted without independent verification. The authors, citations, and conclusions presented here may contain errors, hallucinations, or misrepresentations of the underlying literature. Readers are strongly encouraged to consult primary sources and expert opinions before relying on any information contained herein.

Acknowledgments

This document was generated using Claude, an AI assistant created by Anthropic. The generation process involved synthesis of publicly available research literature, but the accuracy and completeness of this synthesis cannot be guaranteed.

Author contributions

This manuscript was entirely generated by AI with no human author contributions beyond providing the initial prompt and research materials.

Competing interests

The authors declare no competing interests. As an AI-generated document, no financial or personal conflicts exist.

Data availability

This review article synthesizes published literature. No new experimental data were generated. All cited works are available through their respective publishers or preprint servers.

References

- [1] Preskill, J. Quantum computing in the NISQ era and beyond. *Quantum* **2**, 79 (2018).
- [2] Biamonte, J. *et al.* Quantum machine learning. *Nature* **549**, 195–202 (2017).
- [3] Harrow, A. W., Hassidim, A. & Lloyd, S. Quantum algorithm for linear systems of equations. *Physical Review Letters* **103**, 150502 (2009).
- [4] Lloyd, S., Mohseni, M. & Rebentrost, P. Quantum principal component analysis. *Nature Physics* **10**, 631–633 (2014).
- [5] McClean, J. R., Boixo, S., Smelyanskiy, V. N., Babbush, R. & Neven, H. Barren plateaus in quantum neural network training landscapes. *Nature Communications* **9**, 4812 (2018).
- [6] Cerezo, M., Sone, A., Volkoff, T., Cincio, L. & Coles, P. J. Cost function dependent barren plateaus in shallow parametrized quantum circuits. *Nature Communications* **12**, 1791 (2021).
- [7] Tang, E. A quantum-inspired classical algorithm for recommendation systems. *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing* 217–228 (2019).

- [8] Liu, Y., Arunachalam, S. & Temme, K. A rigorous and robust quantum speed-up in supervised machine learning. *Nature Physics* **17**, 1013–1017 (2021).
- [9] Huang, H.-Y. *et al.* Quantum advantage in learning from experiments. *Science* **376**, 1182–1186 (2022).
- [10] Schuld, M. Supervised quantum machine learning models are kernel methods. *arXiv preprint arXiv:2101.11020* (2021).
- [11] Cerezo, M., Larocca, M., Garcia-Martin, D., Diaz, N. L. *et al.* Does provable absence of barren plateaus imply classical simulability? or, why we need to rethink variational quantum computing. *arXiv preprint arXiv:2312.09121* (2023).
- [12] Bharti, K. *et al.* Noisy intermediate-scale quantum algorithms. *Reviews of Modern Physics* **94**, 015004 (2022).
- [13] Zhou, L., Wang, S.-T., Choi, S., Pichler, H. & Lukin, M. D. Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices. *Physical Review X* **10**, 021067 (2020).
- [14] Chen, S. Y.-C. *et al.* Variational quantum circuits for deep reinforcement learning. *IEEE Access* **8**, 141007–141024 (2020).
- [15] Wang, S. *et al.* Noise-induced barren plateaus in variational quantum algorithms. *Nature Communications* **12**, 6961 (2021).
- [16] Huang, H.-Y. *et al.* Power of data in quantum machine learning. *Nature Communications* **12**, 2631 (2021).
- [17] Saggio, V., Asenbeck, B. E., Hamann, A., Strömberg, T. *et al.* Experimental quantum speed-up in reinforcement learning agents. *Nature* **591**, 229–233 (2021).

- [18] Cerezo, M., Verdon, G., Huang, H.-Y., Cincio, L. & Coles, P. J. Challenges and opportunities in quantum machine learning. *Nature Computational Science* **2**, 567–576 (2022).
- [19] Takagi, R., Endo, S., Minagawa, S. & Gu, M. Fundamental limits of quantum error mitigation. *npj Quantum Information* **8**, 114 (2022).
- [20] Cai, Z. *et al.* Quantum error mitigation. *Reviews of Modern Physics* **95**, 045005 (2023).
- [21] Ragone, M. *et al.* A Lie algebraic theory of barren plateaus for deep parameterized quantum circuits. *Nature Communications* **15**, 4137 (2024).
- [22] Sack, S. H. & Egger, D. J. Large-scale quantum approximate optimization on non-planar graphs with machine learning noise mitigation. *Physical Review Research* **6**, 013223 (2024).
- [23] Larocca, M., Thanasilp, S., Wang, S., Sharma, K. *et al.* Barren plateaus in variational quantum computing. *Nature Reviews Physics* (2025).
- [24] Cao, Y. *et al.* Quantum chemistry in the age of quantum computing. *Chemical Reviews* **119**, 10856–10915 (2019).
- [25] Schuld, M. & Killoran, N. Quantum machine learning in feature Hilbert spaces. *Physical Review Letters* **122**, 040504 (2019).
- [26] Peruzzo, A. *et al.* A variational eigenvalue solver on a photonic quantum processor. *Nature Communications* **5**, 4213 (2014).
- [27] Farhi, E., Goldstone, J. & Gutmann, S. A quantum approximate optimization algorithm. *arXiv preprint arXiv:1411.4028* (2014).

- [28] Havlíček, V. *et al.* Supervised learning with quantum-enhanced feature spaces. *Nature* **567**, 209–212 (2019).
- [29] Mitarai, K., Negoro, M., Kitagawa, M. & Fujii, K. Quantum circuit learning. *Physical Review A* **98**, 032309 (2018).
- [30] Schuld, M., Bergholm, V., Gogolin, C., Izaac, J. & Killoran, N. Evaluating analytic gradients on quantum hardware. *Physical Review A* **99**, 032331 (2019).
- [31] Pérez-Salinas, A., Cervera-Lierta, A., Gil-Fuster, E. & Latorre, J. I. Data re-uploading for a universal quantum classifier. *Quantum* **4**, 226 (2020).
- [32] Abbas, A. *et al.* The power of quantum neural networks. *Nature Computational Science* **1**, 403–409 (2021).
- [33] Thanasilp, S., Wang, S., Cerezo, M. & Holmes, Z. Exponential concentration and untrainability in quantum kernel methods. *arXiv preprint arXiv:2208.11060* (2022).
- [34] Kandala, A. *et al.* Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. *Nature* **549**, 242–246 (2017).
- [35] Larocca, M., Czarnik, P., Sharma, K. *et al.* Group-invariant quantum machine learning. *PRX Quantum* **3**, 030341 (2022).
- [36] Cong, I., Choi, S. & Lukin, M. D. Quantum convolutional neural networks. *Nature Physics* **15**, 1273–1278 (2019).
- [37] Stokes, J., Izaac, J., Killoran, N. & Carleo, G. Quantum natural gradient. *Quantum* **4**, 269 (2020).

- [38] Skolik, A., McClean, J. R., Mohseni, M., van der Smagt, P. & Leib, M. Layerwise learning for quantum neural networks. *Quantum Machine Intelligence* **3**, 5 (2021).
- [39] Anand, A., Degroote, M. & Aspuru-Guzik, A. Natural evolutionary strategies for variational quantum computation. *Machine Learning: Science and Technology* **2**, 045012 (2021).
- [40] Bowles, J., Ahmed, S. & Schuld, M. Better than classical? the subtle art of benchmarking quantum machine learning models. *arXiv preprint arXiv:2403.07059* (2024).
- [41] Liu, J. *et al.* Hybrid quantum-classical convolutional neural networks. *Science China Physics, Mechanics & Astronomy* **64**, 290311 (2021).
- [42] Sagingalieva, A., Kordzanganeh, M., Kenbayev, N. *et al.* Hybrid quantum neural network for drug response prediction. *Cancers* **15**, 2705 (2023).
- [43] Shepherd, D. & Bremner, M. J. Temporally unstructured quantum computation. *Proceedings of the Royal Society A* **465**, 1413–1439 (2009).
- [44] Glick, J. R. *et al.* Covariant quantum kernels for data with group structure. *Nature Physics* **20**, 1027–1035 (2024).
- [45] McGeoch, C. C. Theory versus practice in annealing-based quantum computing. *Theoretical Computer Science* **816**, 169–183 (2020).
- [46] Kadowaki, T. & Nishimori, H. Quantum annealing in the transverse Ising model. *Physical Review E* **58**, 5355 (1998).
- [47] Harrigan, M. P., Sung, K. J., Neeley, M. *et al.* Quantum approximate optimization of non-planar graph problems on a planar superconducting processor. *Nature Physics* **17**, 332–336 (2021).

- [48] Goemans, M. X. & Williamson, D. P. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM* **42**, 1115–1145 (1995).
- [49] R Herrman, J. O. T. H., PC Lotshaw. Multi-angle quantum approximate optimization algorithm. *Scientific Reports* (2022). URL <https://www.nature.com/articles/s41598-022-10555-8>.
- [50] Egger, D. J., Mareček, J. & Woerner, S. Warm-starting quantum optimization. *Quantum* **5**, 479 (2021).
- [51] Bravyi, S., Kliesch, A., Koenig, R. & Tang, E. Obstacles to variational quantum optimization from symmetry protection. *Physical Review Letters* **125**, 260505 (2020).
- [52] Farhi, E., Goldstone, J., Gutmann, S. & Sipser, M. Quantum computation by adiabatic evolution. *arXiv preprint quant-ph/0001106* (2000).
- [53] D-Wave Systems Inc. D-wave advantage system overview. <https://www.dwavesys.com/> (2020). Technical documentation.
- [54] Rønnow, T. F. *et al.* Defining and detecting quantum speedup. *Science* **345**, 420–424 (2014).
- [55] Tasseff, B. *et al.* On the emerging potential of quantum annealing hardware for combinatorial optimization. *Journal of Heuristics* **30**, 103–133 (2024).
- [56] Mugel, S., Kuchkovsky, C., Sánchez, E. *et al.* Dynamic portfolio optimization with real datasets using quantum processors and quantum-inspired tensor networks. *Physical Review Research* **4**, 013006 (2022).

- [57] Herman, D. *et al.* A survey of quantum computing for finance. *arXiv preprint arXiv:2201.02773* (2023).
- [58] Patti, T. L., Najafi, K., Gao, X. & Yelin, S. F. Entanglement devised barren plateau mitigation. *Physical Review Research* **3**, 033090 (2021).
- [59] Zhang, K., Liu, L., Hsieh, M.-H. *et al.* *Escaping from the barren plateau via Gaussian initializations in deep variational quantum circuits*, Vol. 35 (2022).
- [60] Sannia, A., Tacchino, F., Tavernelli, I., Giorgi, G. L. & Zambrini, R. Engineered dissipation to mitigate barren plateaus. *npj Quantum Information* **10**, 72 (2024).
- [61] Grover, L. K. A fast quantum mechanical algorithm for database search. *Proceedings of the 28th Annual ACM Symposium on Theory of Computing* 212–219 (1996).
- [62] Tang, E. Dequantizing algorithms to understand quantum advantage in machine learning. *Nature Reviews Physics* (2022). URL <https://www.nature.com/articles/s42254-022-00511-w>.
- [63] Arute, F. *et al.* Quantum supremacy using a programmable superconducting processor. *Nature* **574**, 505–510 (2019).
- [64] M Medvidović, G. C. Classical variational simulation of the quantum approximate optimization algorithm. *npj Quantum Information* (2021). URL <https://www.nature.com/articles/s41534-021-00440-z>.
- [65] Li, Y. & Benjamin, S. C. Efficient variational quantum simulator incorporating active error minimization. *Physical Review X* **7**, 021050 (2017).
- [66] Temme, K., Bravyi, S. & Gambetta, J. M. Error mitigation for short-depth quantum circuits. *Physical Review Letters* **119**, 180509 (2017).

- [67] Nation, P. D., Kang, H., Sundaresan, N. & Gambetta, J. M. Scalable mitigation of measurement errors on quantum computers. *PRX Quantum* **2**, 040326 (2021).
- [68] Strikis, A., Qin, D., Chen, Y., Benjamin, S. C. & Li, Y. Learning-based quantum error mitigation. *PRX Quantum* **2**, 040330 (2021).
- [69] Quek, Y., Stilck Fran  a, D., Khatri, S., Meyer, J. J. & Eisert, J. Exponentially tighter bounds on limitations of quantum error mitigation. *Nature Physics* **20**, 1027–1035 (2024).
- [70] Kim, Y., Eddins, A., Anand, S. *et al.* Evidence for the utility of quantum computing before fault tolerance. *Nature* **618**, 500–505 (2023).
- [71] McArdle, S., Endo, S., Aspuru-Guzik, A., Benjamin, S. C. & Yuan, X. Quantum computational chemistry. *Reviews of Modern Physics* **92**, 015003 (2020).
- [72] D Herman, X. L. Y. S. A. G., C Googin. Quantum computing for finance. *Nature Reviews* ... (2023). URL <https://www.nature.com/articles/s42254-023-00603-1>.
- [73] West, M. T. *et al.* Towards quantum enhanced adversarial robustness in machine learning. *Nature Machine Intelligence* **5**, 581–589 (2023).
- [74] Huang, H.-L., Du, Y., Gong, M. *et al.* Experimental quantum generative adversarial networks for image generation. *Physical Review Applied* **16**, 024051 (2021).
- [75] SA Stein, D. C. Y. M., B Baheri. Qugan: A quantum state fidelity based generative adversarial network. ... *on quantum* ... (2021). URL <https://ieeexplore.ieee.org/abstract/document/9605352/>.