Research Square

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

Quantum-Assisted Recommender Algorithms for Financial Asset Recommendation: Theoretical Bounds and Empircal Validation

W. Bernard Lee

bernard.lee@hedgespa.com

HedgeSPA Private Limited

Chenxi Liu

University of Cambridge

Anthony G. Constantinides

Imperial College London

Research Article

Keywords: Quantum computing · QAOA · Max-Cut · spectral graph compression · financial recommender systems · cuQuantum · Cirq

Posted Date: November 3rd, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11183328/v1

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

Additional Declarations: No competing interests reported.

Human-Centered Intelligent Systems manuscript No.
(will be inserted by the editor)

Quantum-Assisted Recommender Algorithms for Financial Asset Recommendation

Theoretical Bounds and Empirical Validation

W. Bernard Lee · Chenxi Liu · Anthony G. Constantinides

Received: date / Accepted: date

Abstract We present a unified, practical pipeline for a quantum-assisted financial recommender system. Investor similarity graphs are constructed from high-dimensional profiles, compressed via normalized Laplacian spectral embedding, and optimized using the Quantum Approximate Optimization Algorithm (QAOA) on GPU-based quantum simulators (cuQuantum, Cirq-GPU, Cirq with IonQ simulation). We consolidate theoretical derivations (QUBO/Ising mapping, spectral compression correctness, QAOA parameter optimization) and implementation details (feature selection, Nyström extension, QAOA parameter optimization). The key contribution is a unified theoretical framework that directly connects QAOA optimization to classification performance, eliminating disjoint post-processing steps. We provide complete, rigorous proofs of all error bounds, including a variance bound for QAOA concentration, an explicit treatment of the degree-matrix discrepancy, unbalanced misclassification, effective VC dimension, and the QAOA approximation ratio. Two explicit caveats are documented: (i) the overcomplete feature regime is rank-deficient, with generalization controlled solely by the downstream bottleneck and dropout; and (ii) the unconstrained PAC bound is vacuous for target accuracies above roughly 15%, which formally motivates the Laplacian-constrained effective VC dimension. Results show that spectral compression enables tractable quantum optimization for mid-sized problems and that Cirq-GPU offers a favorable speed-accuracy trade-off. We discuss deployment considerations for financial institutions, including explainability, compliance, and workflows under the likely hybrid classical-quantum architecture.

W. B. Lee
HedgeSPA Private Limited, Levels 45-47, Singapore Land Tower, 50 Raffles Place, Singapore 048623
bernard.lee@hedgespa.com or bernardl@cs.stanford.edu

C. Liu
Computing Laboratory, Cambridge University, Cambridge CB3 0FD, UK
cl2089@cam.ac.uk

A. G. Constantinides
AI & Data Analytics Laboratory, Imperial College London, London SW7 2AZ, UK
a.constantinides@imperial.ac.uk

Keywords Quantum computing · QAOA · Max-Cut · spectral graph compression · financial recommender systems · cuQuantum · Cirq.

Mathematics Subject Classification (2020) 68Q12 · 81P68 · 91G10 · 68T05 · 90C27

1 Introduction

Financial recommender systems must match complex, high-dimensional (at least 50 fields, as compared to typical 3 to 5 in e-commerce) investor profiles to suitable products while satisfying regulatory suitability and auditability. Classical clustering and optimization methods struggle with scalability and nonlinear relationships in large datasets. Quantum computing offers a novel computational paradigm for combinatorial optimization problems such as Max-Cut, which can be used to cluster investors and derive product recommendations.

This paper draws from three interconnected studies to present a unified understanding of theoretical foundations, algorithmic design, and practical experiments as a fully integrated whole. Its primary aims are:

1. To develop a rigorous mapping from regulation-driven financial investor profiles to a weighted graph and formulate the clustering problem as a Max-Cut problem suitable for QAOA.
2. To design a spectral graph compression method based on the normalized Laplacian that reduces the problem size to a number of qubits compatible with near-term quantum processors while preserving essential community structure.
3. To provide a unified theoretical framework that directly connects QAOA optimization on compressed graphs to classification accuracy without requiring disjoint post-processing steps.
4. To implement and benchmark the entire pipeline on state-of-the-art quantum simulators (cuQuantum, Cirq-GPU, Cirq-IonQ) and compare performance against exact brute-force enumeration.
5. To validate the approach on a real-world dataset of 150 investor profiles, including out-of-sample prediction using the Nyström extension, and to report detailed in-sample (97.33%) versus leave-one-out (LOO) out-of-sample (94.67%) results.
6. To discuss practical deployment considerations: explainability, regulatory compliance, and a roadmap toward quantum advantage in financial services.

2 Notation and Problem Definition

2.1 Notation

We summarize the mathematical primitives used throughout the paper and fix notation. Our pipeline is informed by the comprehensive survey of quantum computing in finance by Lee and Constantinides [19], which formalizes the transition from classical Markowitz optimization to QAOA-based implementation.

2.1.1 Graph notation

Let $G = (V, E)$ be an undirected weighted graph with $n = |V|$ nodes. The adjacency matrix is $A \in \mathbb{R}^{n\times n}$ with $A_{ij} = w_{ij} \geq 0$. The degree matrix is $D = \mathrm{diag}(d_1, \ldots, d_n)$ with $d_i = \sum_j A_{ij}$. The (symmetric) normalized Laplacian is

$$L_{\mathrm{norm}} = I - D^{-1/2} A D^{-1/2}.$$

The unnormalized Laplacian is $L = D - A$.

2.1.2 Max-Cut and QUBO

The *maximum cut (Max-Cut)* problem seeks a partition $S \cup T = V$ maximizing the sum of weights across the cut. Using binary variables $x_i \in \{0, 1\}$ or spin variables $z_i \in \{\pm 1\}$, the Max-Cut objective is

$$\mathrm{MaxCut}(z) = \frac{1}{2} \sum_{i<j} w_{ij}(1 - z_i z_j).$$

This is a special case of *Quadratic Unconstrained Binary Optimization (QUBO)*, in which a quadratic function of binary variables is minimized without constraints; any QUBO instance can be mapped to a Max-Cut instance and vice versa.

2.1.3 QAOA

QAOA is a hybrid quantum-classical algorithm alternating between the problem Hamiltonian H_C and mixer Hamiltonian H_M. Throughout, X_i and Z_i denote the Pauli-X and Pauli-Z operators acting on qubit i, i.e.,

$$X_i = I^{\otimes(i-1)} \otimes \sigma_x \otimes I^{\otimes(n-i)}, \qquad Z_i = I^{\otimes(i-1)} \otimes \sigma_z \otimes I^{\otimes(n-i)},$$

where $\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ and $\sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$. For Max-Cut,

$$H_C = \sum_{i<j} w_{ij} \frac{1 - Z_i Z_j}{2}, \qquad H_M = \sum_i X_i.$$

An *Ising Hamiltonian* is a quantum Hamiltonian expressed in terms of Pauli-Z operators, of the form $H = \sum_{i<j} J_{ij} Z_i Z_j + \sum_i h_i Z_i$; H_C above is of this form. The parameterized QAOA state at depth p is

$$|\psi(\boldsymbol{\gamma}, \boldsymbol{\beta})\rangle = \prod_{\ell=1}^{p} e^{-i\beta_\ell H_M} e^{-i\gamma_\ell H_C} \, |+\rangle^{\otimes n}.$$

Classical optimization tunes $(\boldsymbol{\gamma}, \boldsymbol{\beta})$ to minimize $\langle H_C \rangle$.

2.1.4 Label space

The classification task has C product classes. In the investor case study of Section 5, $C = 4$. The theoretical derivations are stated for general C; the empirical evaluation uses $C = 4$, with per-class metrics reported for the three classes with sufficient support.

2.1.5 Learning-theoretic notation

Throughout the theoretical sections, $\mathcal{H}$ denotes a class of binary hypotheses $h : \mathcal{X} \to \{\pm 1\}$, $\mathrm{VCdim}(\mathcal{H})$ denotes its VC dimension (Definition 7), $\mathrm{Err}(h)$ denotes the population error of h, and $\widehat{\mathrm{Err}}_n(h)$ denotes its empirical error on an i.i.d. sample of size n. The terms *PAC learning*, *VC dimension*, and *effective VC dimension* are defined in Section 4.4.

2.2 Problem Definition and Modeling

2.2.1 Investor graph construction

Each investor i is represented by a pre-selection feature vector $\mathbf{p}_i \in \mathbb{R}^{m_0}$, where $m_0 = 54$ is obtained by one-hot encoding the 21 raw investor attributes listed in Appendix C. We compute pairwise Pearson correlation

$$\rho_{ij} = \frac{\sum_{k=1}^{m_0}(p_{ik} - \bar{p}_i)(p_{jk} - \bar{p}_j)}{\sqrt{\sum_k (p_{ik} - \bar{p}_i)^2}\sqrt{\sum_k (p_{jk} - \bar{p}_j)^2}}.$$

To ensure positivity, we set adjacency weights $A_{ij} = 1 - \rho_{ij}$ (smaller = higher similarity). Self-loops are zeroed: $A_{ii} = 0$.

2.2.2 Max-Cut for clustering and recommendation

We interpret a Max-Cut partition as a binary clustering: nodes assigned $z_i = +1$ or -1. After obtaining a partition on the compressed graph, labels are propagated back to original investors via spectral inverse mapping; clusters are associated with dominant product choices in the training set to produce recommendations.

2.2.3 Practical constraints

Financial recommender systems must satisfy:

- **Suitability**: recommendations align with risk profiles and regulations;
- **Explainability**: outputs must be auditable and interpretable;
- **Scalability**: handle large numbers of investors and products.

Our pipeline addresses these via feature selection, spectral compression, and explicit inverse mapping.

3 Background and Literature Review

3.1 Classical Approaches to Financial Recommendation

Traditional systems rely on rule-based engines, collaborative filtering, or factor models. These face three limitations: high dimensionality (50-100 features), non-linearity, and regulatory constraints. Recent advances in *graph neural networks (GNNs)* [20, 21] still face scalability issues. The survey by Lee and Constantinides [19] provides an extensive overview of quantum finance applications.

3.2 Quantum Computing for Optimization

QAOA [15] is a standard hybrid algorithm for combinatorial optimization, applied to Max-Cut, portfolio optimization, and other financial problems [14, 38]. Recent work shows QAOA can exceed classical *semidefinite programming (SDP) relaxations* (i.e., convex relaxations of combinatorial problems whose solutions provide polynomial-time upper bounds on the optimal cut) for certain graphs [5, 9].

3.3 Spectral Graph Compression

Spectral clustering [23] partitions a graph by embedding its nodes into the low-dimensional space spanned by the eigenvectors of the graph Laplacian, then applying a standard clustering algorithm (e.g., k-means) in that space. The *Nyström extension* [36] approximates the eigenvectors of a kernel matrix on new points by projecting them onto the eigenvectors of a subsampled kernel matrix. However, combining spectral compression with QAOA for financial recommendation has not been extensively studied.

3.4 Related Work in Financial Quantum Computing

Studies have applied quantum computing to finance, including:

- Portfolio optimization via QAOA [14];
- Derivative pricing with amplitude estimation [24, 37];
- Risk management [37].

Most focus on algorithmic theory rather than practical deployment. Our work bridges this gap.

4 Methodology

4.1 Graph-Aware Feature Expansion

Let $x \in \mathbb{R}^{m_0}$ denote the pre-selection feature vector, where $m_0 = 54$ is obtained by one-hot encoding the 21 raw investor attributes listed in Appendix C. Let L be the

transaction graph Laplacian. We expand to $d_{\text{out}} = 1024$ dimensions via

$$X_{\text{expanded}} = [x, Lx, L^2x, L^3x] \cdot W,$$

where $W \in \mathbb{R}^{(4m_0)\times d_{\text{out}}}$ is a learned projection matrix. With $m_0 = 54$ and $k = 3$ diffusion steps, the pre-projection dimension is $4m_0 = 216$. This captures multi-hop neighbourhood influence.

4.1.1 Mathematical Foundation and Dimensionality Justification

The expansion leverages *graph signal processing* theory [27]. For a graph signal $x \in \mathbb{R}^{m_0}$, we compute:

$$X_{\text{expanded}} = [x, Lx, L^2x, \ldots, L^kx] \cdot W$$

where $L = D - A$ is the graph Laplacian, and $W \in \mathbb{R}^{(m_0(k+1))\times d_{\text{out}}}$ is a learned projection matrix. Throughout, we take $k = 3$ diffusion steps, so the pre-projection dimension is $m_0 \times 4 = 54 \times 4 = 216$.

Signal-to-Noise Ratio: Information-Theoretic Definition. In information theory, the signal-to-noise ratio (SNR) is defined as the ratio of signal power to noise power [26, 8]:

$$\text{SNR} = \frac{S}{N} = \frac{P_{\text{signal}}}{P_{\text{noise}}}.$$

This is the quantity that appears in the Shannon–Hartley capacity of an additive white Gaussian noise (AWGN) channel with bandwidth B, which is the maximum rate (in bits per second) at which information can be transmitted reliably over the channel:

$$C = B \log_2\left(1 + \frac{S}{N}\right) \quad \text{bits/s}.$$

In the continuous-time formulation, $\text{SNR} = P/(N_0W)$. In practice, SNR is estimated on a held-out split by fitting a ridge regressor (or a shallow classifier for the multi-class case) of Y on X_{expanded} and taking the ratio of explained to residual variance.

Theoretical Justification for $d_{out} = 1024$. The choice $d_{\text{out}} = 1024 = 2^{10}$ is motivated by four considerations, with the caveat that the overcomplete regime $d_{\text{out}} > n$ behaves differently from the undercomplete regime and must be treated carefully.

1. **Johnson–Lindenstrauss Lemma** [18]: For any set of n points in high-dimensional space, there exists a linear projection to $d = O(\log n/\epsilon^2)$ dimensions that preserves pairwise distances up to a factor $(1 \pm \epsilon)$. Formally:

   Lemma 1 (Johnson–Lindenstrauss [18]) *For any n points $x_1, \ldots, x_n \in \mathbb{R}^{d_0}$ and any $\epsilon \in (0, 1)$, there exists a linear map $f : \mathbb{R}^{d_0} \to \mathbb{R}^{d'}$ with $d' = O(\log n/\epsilon^2)$ such that*

   $$(1 - \epsilon)\|x_i - x_j\|_2^2 \le \|f(x_i) - f(x_j)\|_2^2 \le (1 + \epsilon)\|x_i - x_j\|_2^2$$

   for all $i, j \in \{1, \ldots, n\}$.

For $n = 150$ investors, this gives $d \gtrsim O(\log 150/0.1^2) \approx 512$ at $\epsilon = 0.1$. The choice $d_{\text{out}} = 1024$ provides a safety margin.

2. **Universal Approximation Capacity** [10]: A neural network with d_{out} hidden units can approximate any continuous function on a compact set to within a specified error. Formally:

   Theorem 1 (Universal approximation [10, 16]) *Let $K \subset \mathbb{R}^{d_0}$ be compact and $f : K \to \mathbb{R}$ continuous. For every $\epsilon > 0$, there exists a feedforward network with a single hidden layer of width $r = O(1/\epsilon^2)$ and ReLU activations such that*

   $$\sup_{x \in K} |f(x) - \hat{f}(x)| \;\le\; \epsilon.$$

   For $d_{\text{out}} = 1024$,

   $$\|f(x) - \hat{f}(x)\|_\infty \le \frac{C}{\sqrt{1024}} \approx \frac{C}{32},$$

   where C depends on the function's Lipschitz constant (a function f is *L-Lipschitz* if $\|f(x) - f(y)\| \le L\|x - y\|$ for all x, y in its domain; L is called the *Lipschitz constant* of f).

3. **Information-Theoretic Capacity**: We measure information in *nats*, the natural-logarithm unit of information.

   Definition 1 (Nat) A *nat* is the unit of information obtained when information is measured using the natural logarithm. If a random variable Y takes values in a finite set $\mathcal{Y}$ with probability mass function $p(y)$, its Shannon entropy is

   $$H(Y) \;=\; -\sum_{y \in \mathcal{Y}} p(y)\, \ln p(y) \quad \text{[nats]},$$

   where ln denotes the natural logarithm. The more familiar *bit* is obtained by using $\log_2$ instead of ln. The two units are related by

   $$1 \text{ nat} \;=\; \frac{1}{\ln 2} \text{ bits} \;\approx\; 1.4427 \text{ bits}, \qquad 1 \text{ bit} \;=\; \ln 2 \text{ nats} \;\approx\; 0.6931 \text{ nats}.$$

   We use nats because the Gaussian-channel capacity formula below is naturally expressed with the natural logarithm; any statement can be converted to bits by multiplying by $1/\ln 2 \approx 1.4427$.

   Treating the expanded features $X_{\text{expanded}} \in \mathbb{R}^{d_{\text{out}}}$ as a Gaussian channel with signal-to-noise ratio SNR and observing n samples, the mutual information between the expanded features and the label is bounded by the Shannon capacity of a d_{out}-dimensional Gaussian channel in per-sample form [26, 8]:

   $$I(X_{\text{expanded}}; Y) \;\le\; \frac{d_{\text{out}}}{2n} \ln\!\left(1 + \frac{\text{SNR} \cdot n}{d_{\text{out}}}\right) \quad \text{[nats]}. \tag{1}$$

   The factor $d_{\text{out}}/(2n)$ is the number of complex degrees of freedom per sample, and the argument of the logarithm is the effective per-dimension SNR.

If the C product classes are equally likely, the label entropy is

$$H(Y) \;=\; \ln C \quad [\text{nats}]. \tag{2}$$

For $C = 4$ product classes,

$$H(Y) \;=\; \ln 4 \;=\; 2\ln 2 \;\approx\; 1.386 \text{ nats} \;\approx\; 1.386 \times 1.4427 \;\approx\; 2.000 \text{ bits}. \tag{3}$$

So identifying one of four equally likely classes requires $\ln 4 \approx 1.39$ nats (equivalently, exactly 2 bits) of information about the label.
We now evaluate (1) for the configuration used in this paper. Take $n = 150$ investors, $d_{\text{out}} = 1024$, and SNR = 1. Then

$$\frac{d_{\text{out}}}{2n} \;=\; \frac{1024}{300} \;\approx\; 3.413, \qquad \frac{\text{SNR}\cdot n}{d_{\text{out}}} \;=\; \frac{1 \times 150}{1024} \;\approx\; 0.1465.$$

Substituting into (1),

$$I(X_{\text{expanded}}; Y) \;\le\; 3.413{\cdot}\ln(1{+}0.1465) \;=\; 3.413{\cdot}\ln(1.1465) \;\approx\; 3.413{\cdot}0.1367 \;\approx\; 0.467 \text{ nats}. \tag{4}$$

Equivalently, in bits, $0.467 \times 1.4427 \approx 0.674$ bits.
The capacity bound (4) is 0.467 nats, which is *below* the $\ln 4 \approx 1.386$ nats required to identify one of four classes. At SNR = 1, therefore, the Gaussian-channel bound does *not* certify that the expanded features carry enough information to distinguish the four product classes. This is a statement about the *worst-case* Gaussian-channel bound, not about the actual features, which may exploit non-Gaussian structure and the graph-diffusion expansion to exceed the bound.
To have $I(X_{\text{expanded}}; Y) \ge \ln 4$, we require

$$\frac{d_{\text{out}}}{2n}\,\ln\!\left(1 + \frac{\text{SNR}\cdot n}{d_{\text{out}}}\right) \;\ge\; \ln 4.$$

With $d_{\text{out}}/2n \approx 3.413$ and $\ln 4 \approx 1.386$,

$$\ln(1 + 0.1465\,\text{SNR}) \;\ge\; \frac{1.386}{3.413} \;\approx\; 0.406.$$

Exponentiating,

$$1 + 0.1465\,\text{SNR} \;\ge\; e^{0.406} \;\approx\; 1.501,$$

so

$$\text{SNR} \;\ge\; \frac{0.501}{0.1465} \;\approx\; 3.42. \tag{5}$$

Thus, under the Gaussian-channel model, an SNR of at least about 3.4 (equivalently, $10\log_{10} 3.42 \approx 5.3$ dB) is needed for the expanded features to carry enough information to identify one of four product classes.
Finally, because the factor d_{out}/n appears in front of the logarithm in (1), the capacity grows only logarithmically in d_{out} once $d_{\text{out}} \gtrsim \text{SNR}\cdot n$. As $d_{\text{out}} \to \infty$ with n and SNR fixed,

$$I(X_{\text{expanded}}; Y) \;\to\; \frac{\text{SNR}}{2\ln 2} \text{ bits} \quad (\text{i.e., } \frac{\text{SNR}}{2} \text{ nats per sample}),$$

which is finite. This is the precise sense in which increasing d_{out} beyond n yields **diminishing returns**: the capacity saturates, and the statistical cost of the overcomplete representation is not offset by additional information.

4. **Graph Diffusion Completeness**: The expansion with $k = 3$ diffusion steps captures up to 3-hop neighborhood information. With $d_{\text{out}} = 1024$, the representation is overcomplete in the sense $d_{\text{out}} > n = 150$.

The Overcomplete Regime $d_{out} > n$: Explicit Caveat on Rank Deficiency and Generalization Control. When $d_{\text{out}} > n$, the claim that the representation ensures "linear independence of features with high probability" is *not* valid. This is the *overcomplete regime deficit caveat*, and it must be stated explicitly because it is a structural property of the design, not an incidental numerical detail.

Caveat 1 (Overcomplete regime deficit) *With $d_{out} = 1024$ and $n = 150$ investors, the expanded feature matrix $X_{expanded} \in \mathbb{R}^{n \times d_{out}}$ has rank at most $\min(n, d_{out}) = n = 150$, so the $d_{out} = 1024$ columns are necessarily linearly dependent. There is no possibility of achieving feature linear independence in this regime; rank deficiency is guaranteed, not merely possible. Consequently:*

1. *The ordinary least-squares problem $X_{expanded}w = y$ is underdetermined, and a linear readout can perfectly interpolate arbitrary labels on the training set, including random labels.*
2. *The generalization error of any downstream classifier is therefore* not *controlled by the ambient dimension d_{out}, but by (i) the effective rank of the empirical covariance $\Sigma_X = X_{expanded}^{\top} X_{expanded}/n$, and (ii) the regularization imposed by the downstream architecture.*
3. *In this paper, the* only *mechanisms that prevent overfitting are the downstream neural network bottleneck layers $1024 \to 512 \to \cdots \to 8$ and the dropout regularization applied within those layers. Feature independence cannot be invoked as a mitigating factor because it does not hold.*
4. *Practically, this means the reported 97.33% in-sample / 94.67% LOO out-of-sample accuracy is a statement about the* combined *pipeline (expansion + bottleneck + dropout + spectral compression + QAOA), not about the expansion layer alone. Removing or weakening the bottleneck would be expected to degrade out-of-sample performance even if in-sample performance remained high.*

The remaining bullets clarify the practical consequences and the design choices that follow from this caveat.

- **Interpolation and overfitting.** With $d_{\text{out}} > n$, a linear readout can perfectly fit arbitrary labels on the training set. Generalization is controlled by the effective dimension and the regularization applied in the progressive-reduction layers.
- **Role of the progressive reduction.** The subsequent $1024 \to 512 \to \cdots \to 8$ stack acts as a learned, data-dependent bottleneck that compresses the overcomplete representation down to a dimension $8 \ll n$. This is the sole architectural mechanism by which statistical control is restored.

– **Practical guidance.** The choice $d_{\text{out}} > n$ should be accompanied by (i) explicit ℓ_2 regularization or dropout in the expansion layer, (ii) early stopping on a validation split, and (iii) monitoring of the effective rank of X_{expanded} (e.g., via the singular-value spectrum) to detect degenerate expansions. When $n \gg d_{\text{out}}$, the usual undercomplete guarantees on linear independence and unique least-squares solutions apply instead.

In summary, $d_{\text{out}} = 1024$ is chosen as a deliberate over-parameterization that provides distance preservation (JL), sufficient width for universal approximation, and a capacity that saturates gracefully once $d_{\text{out}} \gtrsim n$. The statistical cost of this over-parameterization is absorbed *entirely* by the progressive-reduction stage, which restores a low effective dimension before the graph and QAOA stages. An alternative, more conservative setting $d_{\text{out}} \approx n = 150$ avoids the overcomplete regime entirely and is a reasonable choice when n is small and regularization is difficult to tune; we report ablations on $d_{\text{out}} \in \{128, 256, 512, 1024\}$ in Section 5.

4.2 Progressive Dimensionality Reduction

We apply a 7-layer neural network:

$$1024 \to 512 \to 256 \to 128 \to 64 \to 32 \to 16 \to 8.$$

Each layer applies, in order: a linear transformation, a ReLU activation, batch normalization, and dropout. We define each component before describing the layer-wise construction.

Component definitions. For a layer with input $h^{(l)} \in \mathbb{R}^{d_l}$ and output dimension d_{l+1}, the four operations applied in sequence are called, respectively, a *linear transformation*, the *rectified linear unit (ReLU) activation*, *batch normalization*, and *dropout*:

– **Linear transformation:** $z^{(l)} = W^{(l)} h^{(l)} + b^{(l)}$.
– **ReLU activation [22]:** $\mathrm{ReLU}(z) = \max(0, z)$.
– **Batch normalization [17]:** $\mathrm{BN}(z_i) = \gamma \frac{z_i - \mu_{\mathcal{B}}}{\sqrt{\sigma^2_{\mathcal{B}} + \epsilon}} + \delta$.
– **Dropout [28]:** during training, each unit is independently set to zero with probability p, and the surviving units are scaled by $1/(1-p)$.

Layer-wise construction. The forward pass through layer l is

$$h^{(l+1)} = \mathrm{Dropout}_p\Big(\mathrm{BN}\Big(\mathrm{ReLU}\Big(W^{(l)} h^{(l)} + b^{(l)}\Big)\Big)\Big),$$

with $\dim(h^{(l+1)}) < \dim(h^{(l)})$ for every l.

Information bottleneck principle. The sequence of decreasing widths implements the information bottleneck principle [30]:

Definition 2 (Information bottleneck principle) Given an input X, a target Y, and a joint distribution $p(X,Y)$, the *information bottleneck* principle [30] seeks a compressed representation T of X that solves

$$\min_{p(T|X)} \; I(X;T) \quad \text{subject to} \quad I(T;Y) \;\geq\; I_c,$$

where $I(\cdot;\cdot)$ denotes mutual information and I_c is a target level of predictive information. The principle trades off compression (small $I(X;T)$) against predictive power (large $I(T;Y)$).

Why progressive reduction is used instead of a single step. We do *not* claim that the progressive stack uses fewer parameters than a single-step projection; in fact, as shown in Table 1, the progressive stack uses *more* total parameters. The rationale rests on three points.

1. **Non-linear representational capacity.** A single $1024 \to 8$ linear map cannot represent non-linear functions of the expanded features.
2. **Optimization and conditioning.** Reducing dimension in stages, with batch normalization at each stage, keeps the intermediate representations well-conditioned.
3. **Effective parameter count.** Dropout with rate p at layer l means that only a fraction $(1-p)$ of the units are active on any given forward pass, and the L2 penalty drives many weights toward zero. Crucially, in the overcomplete regime of Section 4.1.1, this effective-capacity reduction is the *only* mechanism preventing the expanded representation from interpolating the training set.

Table 1 Parameter counts: single-step projection vs. progressive reduction. The progressive total is 699,008, the exact sum of the seven weight matrices.

Architecture	Weight matrices	Total weights
Single step $1024 \to 8$	1024×8	8,192
Progressive $1024 \to 512 \to \cdots \to 8$	$\sum_l d_l d_{l+1}$	699,008

4.3 Spectral Compression and Encoding

Directly mapping n investors to n qubits is infeasible for large n. We compress to $k \ll n$ virtual nodes using normalized Laplacian spectral embedding.

4.3.1 Normalized Laplacian embedding

Compute the $k+1$ smallest eigenpairs of L_{norm}. Discard the trivial eigenvector (eigenvalue zero) and retain $U \in \mathbb{R}^{n \times k}$ with columns as eigenvectors $2 \ldots k+1$. Node i maps to the i-th row of U.

Table 2 Per-layer breakdown of the progressive architecture's weight count.

Layer	Weights	Cumulative
$1024 \rightarrow 512$	524,288	524,288
$512 \rightarrow 256$	131,072	655,360
$256 \rightarrow 128$	32,768	688,128
$128 \rightarrow 64$	8,192	696,320
$64 \rightarrow 32$	2,048	698,368
$32 \rightarrow 16$	512	698,880
$16 \rightarrow 8$	128	699,008

4.3.2 Compressed adjacency

Project original adjacency into the spectral basis:

$$B = U^\top A U \in \mathbb{R}^{k \times k}.$$

QAOA is applied to B (each virtual node = one qubit).

4.3.3 Unified Inverse Mapping and Classification

After solving Max-Cut on compressed graph B with binary labels $\ell_B \in \{\pm 1\}^k$, we reconstruct original labels directly:

$$\hat{y} = \mathrm{sgn}_\tau(U \ell_B),$$

where $U \in \mathbb{R}^{n \times k}$ is the spectral embedding matrix, $\ell_B \in \{\pm 1\}^k$ are the QAOA-optimized compressed labels, and $\hat{y} \in \{\pm 1\}^n$ are the reconstructed labels for all n investors. This is not a separate post-processing step but the direct output of the QAOA-optimized spectral embedding, yielding continuous scores for explainability.

4.3.4 Multi-Class Classification

For C classes, we extend the binary classification via one-vs-all or recursive bisection:
Option 1: One-vs-All

$$\hat{y}_i = \arg \max_{c \in \{1,\ldots,C\}} \mathrm{sgn}(U_c \ell_B^{(c)})$$

Option 2: Recursive Bisection

Initialize: all nodes in set $S = \{1, \ldots, n\}$
function RecursiveCut(S, $depth$)
  if $depth = \log_2 C$ **then return** labels for S
  end if
  Compute spectral embedding U_S for subgraph S
  Run QAOA to obtain ℓ_B
  Partition: $S_1 = \{i \in S : (U_S \ell_B)_i > 0\}$, $S_2 = S \setminus S_1$
  return RecursiveCut(S_1, $depth + 1$) $\cup$ RecursiveCut(S_2, $depth + 1$)
end function

4.3.5 Classification Loss Function

The classification loss combines the Max-Cut objective with a supervised component:

$$\mathcal{L} = \lambda_1 \underbrace{\left(-\frac{1}{2} \sum_{i<j} A_{ij}(1 - \hat{y}_i \hat{y}_j) \right)}_{\text{Max-Cut Objective}} + \lambda_2 \underbrace{\sum_{i=1}^{n} \mathbb{1}[\hat{y}_i \neq y_i]}_{\text{Classification Error}}$$

or in differentiable form using cross-entropy:

$$\mathcal{L}_{\text{CE}} = -\sum_{i=1}^{n} \sum_{c=1}^{C} y_{ic} \log \left(\frac{\exp((U\ell_B)_i)}{\sum_{c'=1}^{C} \exp((U\ell_B)_{ic'})} \right).$$

4.3.6 Nyström Extension for Out-of-Sample Classification

For a new investor with adjacency vector $\mathbf{e}_{\text{new}}$, the Nyström extension [36] (which approximates the eigenvectors of a kernel matrix on new points by projecting them onto the eigenvectors of a subsampled kernel matrix) projects into spectral space:

$$u_{\text{new}} = \mathbf{e}_{\text{new}}^{\top} U \Lambda^{-1},$$

where Λ is the diagonal matrix of retained eigenvalues. The predicted label is then:

$$\hat{y}_{\text{new}} = \text{sgn}(u_{\text{new}}^{\top} \ell_B).$$

For multi-class:

$$\hat{y}_{\text{new}} = \arg \max_{c \in \{1,\ldots,C\}} \text{sgn}(u_{\text{new}}^{\top} \ell_B^{(c)}).$$

4.4 Unified Theoretical Framework

We now formalize the pipeline as a single end-to-end operator and derive non-asymptotic guarantees that connect QAOA optimization on the compressed graph to classification accuracy on the original investor graph. All statements are stated with explicit assumptions and, where applicable, constants. Full proofs are deferred to Appendix A.

4.4.1 Standing Assumptions and Setup

Let $G = (V, E)$ be a connected, undirected, weighted graph with $n = |V|$ nodes, adjacency matrix $A \in \mathbb{R}_{\geq 0}^{n \times n}$, and degree matrix $D = \text{diag}(A\mathbf{1})$. Define the symmetric normalized Laplacian

$$\mathcal{L} = I - D^{-1/2} A D^{-1/2} \in \mathbb{R}^{n \times n},$$

with eigenvalues $0 = \lambda_1 < \lambda_2 \leq \cdots \leq \lambda_n \leq 2$ and orthonormal eigenvectors $v_1, \ldots, v_n$. The trivial eigenvector is $v_1 = D^{1/2}\mathbf{1}/\sqrt{\mathrm{vol}(G)}$. Write

$$U = [v_2, v_3, \ldots, v_{k+1}] \in \mathbb{R}^{n \times k}$$

for the matrix of the k smallest nontrivial eigenvectors, and

$$\Lambda_k = \mathrm{diag}(\lambda_2, \ldots, \lambda_{k+1}) \in \mathbb{R}^{k \times k}.$$

We also work with the unnormalized Laplacian $L = D - A$, with eigenvalues $0 = \mu_1 < \mu_2 \leq \cdots \leq \mu_n \leq 2\Delta$, where $\Delta = \max_i d_i$.

Assumption 1 (Connectedness) *G is connected, so $\mu_2 > 0$ and $\lambda_2 > 0$.*

Assumption 2 (Spectral gap at k) *There exists $\gamma_k > 0$ such that $\mu_{k+1} - \mu_k \geq \gamma_k$.*

Assumption 3 (Bounded spectrum) $\|A\|_2 \leq \Delta$.

Definition 3 (QAOA approximation ratio) For a graph G with adjacency matrix B, depth p, and QAOA-optimized parameters $(\boldsymbol{\gamma}^\star, \boldsymbol{\beta}^\star)$, the *QAOA approximation ratio* is

$$\rho_p(B) = \frac{\langle \psi(\boldsymbol{\gamma}^\star, \boldsymbol{\beta}^\star) | H_C | \psi(\boldsymbol{\gamma}^\star, \boldsymbol{\beta}^\star) \rangle}{\max_{\ell \in \{\pm 1\}^k} F_B(\ell)}.$$

Assumption 4 (QAOA approximation ratio bound) *For depth p, the QAOA-optimized cut on the compressed graph B satisfies*

$$F_B(\ell_B^{\mathrm{QAOA}}) \geq \rho_p \, F_B(\ell_B^\star),$$

where $\ell_B^\star = \arg\max_{\ell \in \{\pm 1\}^k} F_B(\ell)$ and $\rho_p \in (0, 1]$ is the depth-p approximation ratio (Definition 3).

Remark 1 (On Assumption 4) Assumption 4 is standard in the QAOA literature [15, 9, 38]. For Max-Cut on 3-regular graphs, $\rho_1 \geq 0.6924$ (Farhi et al.). For general graphs, $\rho_p \to 1$ as $p \to \infty$ under conditions characterized by Brandão et al. [5]. We keep ρ_p abstract so the bound applies to any simulator or hardware backend that achieves a certified ratio. Section 4.4.6 provides the precise statement and known bounds.

Definition 4 (Reconstruction operator) Fix a threshold $\tau \in \mathbb{R}$. Given compressed labels $\ell_B \in \{\pm 1\}^k$, define the continuous score vector

$$s = U\ell_B \in \mathbb{R}^n,$$

and the reconstructed label vector

$$\hat{y} = \mathrm{sgn}_\tau(s), \qquad (\mathrm{sgn}_\tau(s))_i = \begin{cases} +1, & s_i \geq \tau, \\ -1, & s_i < \tau. \end{cases}$$

When the classes are balanced we take $\tau = 0$; otherwise τ is the empirical median of s.

Definition 5 (Cut functional) For any $x \in \{\pm 1\}^n$ define

$$F_A(x) \;=\; \frac{1}{4} x^\top L x \;=\; \frac{1}{2} \sum_{i<j} A_{ij}\,(1 - x_i x_j).$$

Similarly, for $y \in \{\pm 1\}^k$,

$$F_B(y) \;=\; \frac{1}{4} y^\top L_B\, y, \qquad L_B = D_B - B, \quad D_B = \operatorname{diag}(B\mathbf{1}).$$

Definition 6 (Classification accuracy) For true labels $y^\star \in \{\pm 1\}^n$,

$$\operatorname{Acc}(\hat{y}, y^\star) \;=\; \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}\{\hat{y}_i = y_i^\star\} \;=\; 1 - \delta(\hat{y}, y^\star),$$

where $\delta(x, y) = \frac{1}{2n}\|x - y\|_2^2$ is the disagreement fraction.

Definition 7 (VC dimension) Let $\mathcal{H}$ be a class of binary functions $h : \mathcal{X} \to \{\pm 1\}$ on a domain $\mathcal{X}$. We say that $\mathcal{H}$ *shatters* a finite set $S = \{x_1, \ldots, x_m\} \subset \mathcal{X}$ if, for every labelling $(y_1, \ldots, y_m) \in \{\pm 1\}^m$, there exists $h \in \mathcal{H}$ with $h(x_i) = y_i$ for all $i = 1, \ldots, m$. The *VC dimension* of $\mathcal{H}$, denoted $\operatorname{VCdim}(\mathcal{H})$, is the largest cardinality of a set that $\mathcal{H}$ can shatter:

$$\operatorname{VCdim}(\mathcal{H}) \;=\; \max\Big\{ m \in \mathbb{N} \;:\; \exists\, S \subset \mathcal{X},\ |S| = m,\ \mathcal{H} \text{ shatters } S \Big\},$$

with the convention $\operatorname{VCdim}(\mathcal{H}) = 0$ if $\mathcal{H}$ shatters no set of size 1, and $\operatorname{VCdim}(\mathcal{H}) = \infty$ if $\mathcal{H}$ shatters sets of every finite size.

The VC dimension is the standard combinatorial measure of the capacity of a binary hypothesis class. It governs the uniform convergence of empirical error to population error via the Vapnik–Chervonenkis inequality (Theorem 2), which is the tool used in Section 4.4.8 and Theorem 7.

Definition 8 (Linear threshold function and linear threshold class) A *linear threshold function* on $\mathbb{R}^d$ is a function of the form

$$x \;\mapsto\; \operatorname{sgn}(w^\top x + b)$$

for some weight vector $w \in \mathbb{R}^d$ and bias $b \in \mathbb{R}$. The *linear threshold class* $\mathcal{H}_{\text{lin}}^d$ is the set of all such functions:

$$\mathcal{H}_{\text{lin}}^d \;=\; \big\{ x \mapsto \operatorname{sgn}(w^\top x + b) \;:\; w \in \mathbb{R}^d,\ b \in \mathbb{R} \big\}.$$

Remark 2 (VC dimension of a linear threshold class) The canonical example, which underlies all the VC bounds in this paper, is the linear threshold class $\mathcal{H}_{\text{lin}}^d$ on $\mathbb{R}^d$ (Definition 8). It is a classical result [33, 34] that

$$\operatorname{VCdim}\big(\mathcal{H}_{\text{lin}}^d\big) \;=\; d + 1.$$

The pipeline's hypothesis class $\mathcal{H}_{k,m}$ is a composition of such a linear threshold class on $\mathbb{R}^k$ with a linear map from $\mathbb{R}^m$ to $\mathbb{R}^k$, which is why the VC dimension of $\mathcal{H}_{k,m}$ is $\Theta(km)$ rather than $\Theta(k)$; see Theorem 7.

Definition 9 (PAC learning) A hypothesis class $\mathcal{H}$ is *PAC-learnable* (Probably Approximately Correct learnable) [32] if there exists a learning algorithm $\mathcal{A}$ and a function $m_{\mathcal{H}} : (0,1)^2 \to \mathbb{N}$ such that, for every $\epsilon, \delta \in (0,1)$, every distribution $\mathcal{D}$ over $\mathcal{X} \times \{\pm 1\}$, and every sample size $n \geq m_{\mathcal{H}}(\epsilon, \delta)$, the algorithm $\mathcal{A}$ outputs, with probability at least $1 - \delta$ over the i.i.d. sample, a hypothesis $h \in \mathcal{H}$ with population error $\mathrm{Err}(h) \leq \epsilon$. The function $m_{\mathcal{H}}$ is called the *sample complexity* of $\mathcal{H}$.

A sufficient condition for PAC-learnability is that $\mathcal{H}$ has finite VC dimension (Definition 7); this is the route used in Theorem 7, where the sample complexity is expressed in terms of $\mathrm{VCdim}(\mathcal{H})$.

Theorem 2 (VC inequality [33, 34]) *Let $\mathcal{H}$ be a hypothesis class with finite VC dimension $d = \mathrm{VCdim}(\mathcal{H})$, and let $\widehat{\mathrm{Err}}_n(h)$ be the empirical error of h on an i.i.d. sample of size n. Then for any $\delta \in (0,1)$, with probability at least $1 - \delta$ over the sample,*

$$\sup_{h \in \mathcal{H}} \left| \widehat{\mathrm{Err}}_n(h) - \mathrm{Err}(h) \right| \;\leq\; \sqrt{\frac{2}{n}\left(d \log \frac{2en}{d} + \log \frac{2}{\delta} \right)}.$$

4.4.2 Spectral Compression Error

Lemma 2 (Eckart–Young–Mirsky [12, 29]) *Let $A = \sum_{i=1}^{n} \lambda_i^A u_i u_i^\top$ be the spectral decomposition of the symmetric matrix A, ordered so that $|\lambda_1^A| \geq |\lambda_2^A| \geq \cdots$. Then for any $k \leq n$,*

$$\min_{\mathrm{rank}(M) \leq k} \|A - M\|_F \;=\; \sqrt{\sum_{i=k+1}^{n} (\lambda_i^A)^2},$$

and the minimizer is $M^\star = \sum_{i=1}^{k} \lambda_i^A u_i u_i^\top$.

Definition 10 (Relative spectral compression error)

$$\varepsilon_{\mathrm{spec}}(k) \;=\; \frac{\|A - UBU^\top\|_F}{\|A\|_F}, \qquad B = U^\top A U.$$

Lemma 3 (Spectral compression error bound) *Under Assumption 2,*

$$\begin{aligned} \|A - UBU^\top\|_F \;\leq\; & \|A\|_2 \cdot \sqrt{\sum_{i=k+1}^{n} \|u_i\|_2^4} \\ & + \; \frac{2\|A\|_2}{\gamma_k} \cdot \|(\mathcal{L} - \lambda_{k+1} I) U\|_F. \end{aligned}$$

In particular, if the spectrum of A decays so that $\sum_{i>k} \sigma_i(A)^2 \leq \eta_k^2 \|A\|_F^2$, then

$$\begin{aligned} \frac{\|A - UBU^\top\|_F}{\|A\|_F} \;\leq\; & \eta_k \\ & + \; \frac{2\|A\|_2}{\gamma_k \|A\|_F} \cdot \|(\mathcal{L} - \lambda_{k+1} I) U\|_F. \end{aligned}$$

Proof Decompose $A - UBU^\top = (I-P)A(I-P) + PA(I-P) + (I-P)AP$ with $P = UU^\top$. The first term is bounded by the spectral projector of $\mathcal{L}$; the second by the spectral gap γ_k; the cross terms vanish by orthogonality of U. The full proof is given in Appendix A.2.

Remark 3 The bound is tight when the top-k spectral subspace of $\mathcal{L}$ aligns with the top-k singular subspace of A. In the worst case (misaligned subspaces), the factor $1/\gamma_k$ can be large. This is the precise sense in which a small spectral gap hurts compression.

4.4.3 Cut-Value Preservation with Degree-Matrix Discrepancy

Lemma 4 (Cut-value preservation with degree discrepancy) *For any $\ell_B \in \{\pm 1\}^k$, let $x = U\ell_B \in \mathbb{R}^n$ and $\hat{y} = \mathrm{sgn}_0(x) \in \{\pm 1\}^n$. Then*

$$|F_A(\hat{y}) - F_B(\ell_B)| \;\le\; \frac{1}{2}\|L\|_2\|\hat{y}-x\|_2\|\hat{y}+x\|_2 + \frac{1}{2}\|A - UBU^\top\|_2\|x\|_2^2 \;+\; \frac{1}{4}\Delta\|x\|_2^2.$$

If $\hat{y} = x$, the first term vanishes.

Proof Split $F_A(\hat{y}) - F_B(\ell_B)$ into a rounding term and a compression term after adding and subtracting $\frac{1}{4}x^\top Lx$. The rounding term is controlled by the polarization identity; the compression term splits into the A-part (which cancels exactly because $B = U^\top AU$) and the D-part (which yields the degree-matrix discrepancy $\frac{1}{4}\Delta\|x\|_2^2$). The full proof is given in Appendix A.3.

Remark 4 (Role of the degree-matrix discrepancy) The term $\frac{1}{4}\Delta\|x\|_2^2$ is real and cannot always be dropped. In the common case where A is approximately k-rank and D is approximately $UD_BU^\top$, this term is negligible. For the remainder of the paper, we write

$$\varepsilon_{\deg} \;=\; \frac{\Delta\|x\|_2^2}{4F_B(\ell_B^\star)}$$

as the relative degree-discrepancy error and include it in the total error budget.

4.4.4 Cut Gap Implies Label Agreement with Unbalanced Misclassification

We bound the classification error via the spectral gap $\mu_2(L)$ directly, using the variational characterization of $\mu_2(L)$ on the orthogonal complement of $\mathbf{1}$. (For reference, Cheeger's inequality [7, 1] relates the conductance of a graph to $\lambda_2(\mathcal{L})$, but it does not directly bound classification accuracy; we therefore use a self-contained argument based on the variational characterization.)

Lemma 5 (Cut gap implies label agreement) *Let $y^\star = \arg\max_{y\in\{\pm1\}^n} F_A(y)$, and let $\hat{y} \in \{\pm 1\}^n$ satisfy $F_A(y^\star) - F_A(\hat{y}) \le \eta$. Let $d = \delta(\hat{y}, y^\star) = \frac{1}{n}|\{i : \hat{y}_i \ne y_i^\star\}|$. Then*

$$d \;\le\; \frac{4\eta}{n\,\mu_2(L)}.$$

If the misclassifications are balanced between the two classes (i.e., equal numbers of $+1 \to -1$ *and* $-1 \to +1$ *errors), the constant improves to* 2*:*

$$d \;\le\; \frac{2\eta}{n\,\mu_2(L)}.$$

Proof Use the variational characterization of $\mu_2(L)$ on the orthogonal complement of $\mathbf{1}$, decompose $x = \hat{y} - y^\star$ into mean and orthogonal components, and bound the cut deficit η from below by $\frac{1}{2}\mu_2(L)\|x^\perp\|_2^2$. The constant 4 is valid for unbalanced misclassification; 2 requires balanced errors. The full proof, including the treatment of the imbalance ratio $\beta = |p-q|/(nd)$, is given in Appendix A.4.

Remark 5 (Why the constant is 4, not 2) The constant 2 is valid only when the misclassifications are balanced between the two classes (i.e., $p = q$). In the general unbalanced case, the correct constant is 4. The bound is stated with 4; the sharper constant 2 holds under the additional balancedness assumption.

4.4.5 Classification Accuracy Bound

Theorem 3 (Classification accuracy bound) *Under Assumptions 1–4, let* $\hat{y} = \mathrm{sgn}_0(U\ell_B^{\mathrm{QAOA}})$ *and let* $y^\star = \arg\max_{y\in\{\pm1\}^n} F_A(y)$*. Then*

$$\mathrm{Acc}(\hat{y}, y^\star) \;\ge\; 1 \;-\; \frac{4}{n\mu_2(L)}\Big[\underbrace{(1-\rho_p)F_B(\ell_B^\star)}_{\text{QAOA}} \;+\; \underbrace{\tfrac{1}{2}\|A - UBU^\top\|_2\|U\ell_B^{\mathrm{QAOA}}\|_2^2}_{\text{compression}}$$
$$+\; \underbrace{\tfrac{1}{2}\|L\|_2\|\hat{y}-x\|_2\|\hat{y}+x\|_2}_{\text{rounding}} \;+\; \underbrace{\tfrac{1}{4}\Delta\|x\|_2^2}_{\text{degree}}\Big],$$

where $x = U\ell_B^{\mathrm{QAOA}}$.

Proof Combine Lemma 4 (cut preservation with degree discrepancy), Assumption 4 (QAOA ratio), and Lemma 5 (cut gap implies accuracy), then apply $\mathrm{Acc} = 1-\delta(\hat{y}, y^\star)$. The full proof is given in Appendix A.5.

Corollary 1 (Asymptotic form) *If the rounding is exact (*$\hat{y} = x$*) and* $k \ll n$*,*

$$\mathrm{Acc}(\hat{y}, y^\star) \;\ge\; 1 \;-\; \frac{4(1-\rho_p)F_B(\ell_B^\star)}{n\mu_2(L)} \;-\; \frac{2\|A - UBU^\top\|_2}{\mu_2(L)} \;-\; \frac{\Delta}{\mu_2(L)}.$$

In the regime $F_B(\ell_B^\star) = O(n)$*,* $\|A - UBU^\top\|_2 = O(\varepsilon_{\mathrm{spec}}\|A\|_2)$*, and* $\Delta = O(\|A\|_2)$*,*

$$\mathrm{Acc}(\hat{y}, y^\star) \;\ge\; 1 \;-\; O\!\left(\frac{1-\rho_p}{\mu_2(L)}\right) \;-\; O\!\left(\frac{\varepsilon_{\mathrm{spec}}\|A\|_2}{\mu_2(L)}\right)$$
$$-\; O\!\left(\frac{\|A\|_2}{\mu_2(L)}\right).$$

4.4.6 QAOA Approximation Ratio

Assumption 4 is the only assumption in the framework that depends on the quantum algorithm. Definition 3 above makes the ratio precise; we now summarize known bounds on it.

Theorem 4 (Known QAOA bounds [15, 9]) *For Max-Cut on a D-regular graph:*

1. *At depth $p = 1$, $\rho_1 \geq 0.6924$ for $D = 3$, and more generally $\rho_1 \geq 1 - 1/D + o(1/D)$.*
2. *At depth $p = 2$, $\rho_2 \geq 0.755$ for $D = 3$.*
3. *As $p \to \infty$, $\rho_p \to 1$ for all graphs, provided the QAOA parameters are optimized globally.*

For general (non-regular) graphs, ρ_p depends on the graph structure; the bounds above hold with D replaced by the average degree $\bar{d}$, and degrade as the degree distribution becomes more heterogeneous.

Remark 6 (Finite-p behavior) In practice, $p = 3$ is often sufficient to achieve $\rho_3 \geq 0.85$ on the compressed graphs considered in this paper. The empirical results in Section 5 show $\mathrm{Err}_{\mathrm{QAOA}}(3) \approx 0.16$, corresponding to $\rho_3 \approx 0.84$, consistent with Theorem 4.

4.4.7 Total Error Decomposition

Theorem 5 (Total error decomposition) *The total classification error of the unified pipeline satisfies*

$$\begin{aligned}\mathrm{Err}_{\mathrm{total}} \;\leq\; & \mathrm{Err}_{\mathrm{QAOA}}(p) \;+\; \mathrm{Err}_{\mathrm{spec}}(k) \;+\; \mathrm{Err}_{\mathrm{round}}(k,n) \\ & +\; \mathrm{Err}_{\mathrm{deg}}(k,n) \;+\; \mathrm{Err}_{\mathrm{stat}}(n,m),\end{aligned}$$

where

$$\mathrm{Err}_{\mathrm{QAOA}}(p) = \frac{4(1-\rho_p)F_B(\ell_B^\star)}{n\mu_2(L)},$$

$$\mathrm{Err}_{\mathrm{spec}}(k) = \frac{2\|A - UBU^\top\|_2}{\mu_2(L)},$$

$$\mathrm{Err}_{\mathrm{round}}(k,n) = \frac{2\|L\|_2\|\hat{y} - x\|_2\|\hat{y} + x\|_2}{n\mu_2(L)},$$

$$\mathrm{Err}_{\mathrm{deg}}(k,n) = \frac{\Delta}{\mu_2(L)},$$

$$\mathrm{Err}_{\mathrm{stat}}(n,m) = \sqrt{\frac{2}{n}\left(km_{\mathrm{eff}}\log\frac{2en}{km_{\mathrm{eff}}} + \log\frac{2}{\delta}\right)},$$

with m_{eff} the effective feature dimension defined in Section 4.4.8.

Proof Combine Theorem 3 with the VC bound from Theorem 6. The five error terms are additive because they arise from independent sources: QAOA sub-optimality (quantum), spectral compression (linear algebra), rounding (sign thresholding), degree-matrix discrepancy (graph normalization), and finite-sample estimation (statistics).

4.4.8 Effective VC Dimension

The following definition introduces the notion of *effective VC dimension*, which is specific to this paper: it is the VC dimension (Definition 7) of the pipeline's hypothesis class after restricting the linear map $U^\top$ to the Laplacian-constrained set $\mathcal{U}$, in contrast to the free-parameter VC dimension $\mathrm{VCdim}(\mathcal{H}_{k,m}) = \Theta(km)$ used in Theorem 7. Throughout this subsection, m denotes the number of *independent* investor attributes (the 21 attributes of Appendix C), not the pre-selection dimension $m_0 = 54$. The one-hot encoded columns within a single categorical attribute are perfectly correlated and therefore do not contribute independent degrees of freedom to the VC dimension; using $m = 21$ is the correct effective dimension for the sample-complexity bounds.

Definition 11 (Effective VC dimension) Let $\mathcal{U} \subset \mathbb{R}^{n\times k}$ be the set of matrices whose columns are the k smallest nontrivial eigenvectors of a normalized Laplacian $\mathcal{L}$ of a graph on n nodes with edge weights in $[0, \Delta]$. The effective VC dimension of the pipeline is

$$\mathrm{VCdim}_{\mathrm{eff}}(\mathcal{H}) \;=\; \mathrm{VCdim}\left(\{x \mapsto \mathrm{sgn}(U^\top x) : U \in \mathcal{U}\}\right),$$

where $\mathrm{VCdim}(\cdot)$ is the VC dimension of Definition 7.

Theorem 6 (Effective VC dimension) *Let $\mathcal{U}$ be as in Definition 11. Then*

$$\mathrm{VCdim}_{\mathrm{eff}}(\mathcal{H}) \;\le\; C_0 \cdot k \cdot m \cdot \frac{\log n}{n},$$

where $C_0 > 0$ is an absolute constant depending only on Δ. In the regime $n \gg k \log n$, this is

$$\mathrm{VCdim}_{\mathrm{eff}}(\mathcal{H}) \;=\; O\left(\frac{km \log n}{n}\right),$$

which is significantly smaller than the free-parameter bound km.

Proof Use the Davis–Kahan theorem (Theorem 9 in Appendix A.1) to bound the Lipschitz constant of the eigenvector map, the covering number of the Stiefel manifold $V_k(\mathbb{R}^n)$, and the VC dimension $k + 1$ of linear threshold functions on $\mathbb{R}^k$. The full proof is given in Appendix A.6.

Corollary 2 (Sample complexity with effective VC dimension) *With probability at least $1 - \delta$ over an i.i.d. sample of size n,*

$$\mathrm{Err}_{\mathrm{stat}}(n, m) \;\le\; \sqrt{\frac{2}{n}\left(\frac{km \log n}{n} \log \frac{2en^2}{km \log n} + \log \frac{2}{\delta}\right)}.$$

For $n = 150$, $m = 21$, $k = 20$, this gives $\mathrm{Err}_{\mathrm{stat}} \le 0.04$ at $\delta = 0.05$, consistent with the empirical LOO accuracy of 94.67%.

4.4.9 Sample Complexity with Feature Dimensionality

Theorem 7 (Sample complexity) *Let* $\mathcal{H}_{k,m} = \{x \mapsto \operatorname{sgn}(U^\top x) : U \in \mathbb{R}^{n\times k}, x \in \mathbb{R}^m\}$*, where* m *is the number of independent features, and let* $\mathrm{VCdim}(\cdot)$ *be the VC dimension of Definition 7. Then*

$$\mathrm{VCdim}(\mathcal{H}_{k,m}) \;=\; \Theta(km).$$

Consequently, by the Vapnik–Chervonenkis inequality (Theorem 2), for any $\delta \in (0,1)$*, with probability at least* $1-\delta$ *over an i.i.d. sample of size* n*,*

$$\mathrm{Err}_{\mathrm{stat}}(n,m) \;\le\; \sqrt{\frac{2}{n}\left(km\log\frac{2en}{km} + \log\frac{2}{\delta}\right)}.$$

To achieve $\mathrm{Err}_{\mathrm{stat}} \le \epsilon$ *it suffices that*

$$n \;\ge\; \frac{2}{\epsilon^2}\left(km\log\frac{2e}{\epsilon^2} + \log\frac{2}{\delta}\right).$$

Proof The class $\mathcal{H}_{k,m}$ is the composition of a linear map from $\mathbb{R}^m$ to $\mathbb{R}^k$ with a linear threshold class (Definition 8) on $\mathbb{R}^k$; its VC dimension is $\Theta(km)$. Apply the VC inequality (Theorem 2). The full proof is given in Appendix A.7.

Remark 7 (Why the VC bound is loose) The bound $\mathrm{VCdim}(\mathcal{H}_{k,m}) = \Theta(km)$ treats U as a free parameter matrix. In our pipeline, U is determined by the graph Laplacian, which is itself determined by the data. This coupling reduces the effective capacity dramatically, as quantified in Theorem 6. The empirical results ($n = 150$, 97.33% in-sample, 94.67% LOO) confirm that the effective sample complexity is far below the worst-case bound.

4.4.10 QAOA Concentration with Variance Bound

Theorem 8 (QAOA concentration with variance bound) *Let* ℓ_B^{QAOA} *be the depth-*p *QAOA output on* B*, and let* ℓ_A^{QAOA} *be the depth-*p *QAOA output on* A *with the same parameters. Define*

$$X \;=\; F_A(\operatorname{sgn}_0(U\ell_B^{\mathrm{QAOA}})) - F_B(\ell_B^{\mathrm{QAOA}}).$$

Then

$$\mathbb{E}[X] \;\le\; \tfrac{1}{2}\|A - UBU^\top\|_F\, k + \tfrac{1}{4}\Delta k,$$

and

$$\mathrm{Var}(X) \;\le\; 2\|A - UBU^\top\|_F^2 + \tfrac{1}{2}\Delta^2 k.$$

Consequently, for any $\epsilon > 0$*,*

$$\mathbb{P}\big(|X - \mathbb{E}[X]| > \epsilon\big) \;\le\; \frac{2\|A - UBU^\top\|_F^2 + \frac{1}{2}\Delta^2 k}{\epsilon^2}.$$

Proof Bound the change in X when a single bit of ℓ_B is flipped, then apply the bounded-difference inequality (Lemma 6) and Chebyshev's inequality (Lemma 7). The full proof is given in Appendix A.8.

4.4.11 Summary of Theoretical Guarantees

Table 3 Summary of Theoretical Guarantees for the Unified Quantum-Assisted Recommender Pipeline

Guarantee	Statement
Spectral compression	$\|A - UBU^\top\|_F \le \|A\|_2 \sqrt{\sum_{i>k} \|u_i\|_2^4}$ $+ \frac{2\|A\|_2}{\gamma_k} \|(\mathcal{L} - \lambda_{k+1} I)U\|_F$
Cut preservation	$\lvert F_A(\hat{y}) - F_B(\ell_B)\rvert \le \frac{1}{2}\|L\|_2 \|\hat{y} - x\|_2 \|\hat{y} + x\|_2$ $+ \frac{1}{2}\|A - UBU^\top\|_2 \|x\|_2^2 + \frac{1}{4}\Delta \|x\|_2^2$
Cut gap $\Rightarrow$ accuracy	$\delta(\hat{y}, y^\star) \le \frac{4\eta}{n\mu_2(L)}$ (general); $\le \frac{2\eta}{n\mu_2(L)}$ (balanced)
Classification	$\mathrm{Acc}(\hat{y}, y^\star) \ge 1 - \frac{4\eta}{n\mu_2(L)}$
QAOA ratio	$\rho_p \ge 0.6924$ $(p = 1, D = 3)$; $\rho_p \to 1$ as $p \to \infty$
Effective VC dimension	$\mathrm{VCdim}_{\mathrm{eff}} = O(km \log n/n)$
Sample complexity	$\mathrm{Err}_{\mathrm{stat}} \le \sqrt{\frac{2}{n}(km \log \frac{2en}{km} + \log \frac{2}{\delta})}$
QAOA concentration	$\mathbb{P}(\lvert X - \mathbb{E}X\rvert > \epsilon) \le (2\|A - UBU^\top\|_F^2$ $+ \frac{1}{2}\Delta^2 k)/\epsilon^2$
Error decomposition	$\mathrm{Err}_{\mathrm{total}} \le \mathrm{Err}_{\mathrm{QAOA}} + \mathrm{Err}_{\mathrm{spec}}$ $+ \mathrm{Err}_{\mathrm{round}} + \mathrm{Err}_{\mathrm{deg}} + \mathrm{Err}_{\mathrm{stat}}$

4.5 Graph-Cut Informed Prediction

After spectral compression, we inject domain knowledge via graph cuts:

$$\hat{y}_i = \mathrm{softmax}\left(W_{\mathrm{global}} h_i + \sum_{c \in C(i)} W_c h_i \right),$$

where $h_i \in \mathbb{R}^8$ is the reduced feature vector for investor i, $C(i)$ are the graph cuts (clusters) containing investor i, $W_{\mathrm{global}} \in \mathbb{R}^{C \times 8}$ is the global weight matrix, and $W_c \in \mathbb{R}^{C \times 8}$ are cut-specific weight matrices. The probabilistic interpretation is

$$P(y_i = c \mid h_i, C(i)) = \frac{\exp\left(w_c^\top h_i + \sum_{c' \in C(i)} w_{c,c'}^\top h_i \right)}{\sum_{d=1}^{C} \exp\left(w_d^\top h_i + \sum_{c' \in C(i)} w_{d,c'}^\top h_i \right)}.$$

4.6 QUBO/Ising Mapping and QAOA Implementation

For compressed graph B with k nodes, the cost Hamiltonian is

$$H_C = \sum_{i<j} B_{ij} \frac{1 - Z_i Z_j}{2},$$

implemented via two-qubit ZZ rotations.

4.7 Parameter optimization

Classical optimizers tune (γ, β) to minimize $\langle H_C \rangle$. The two optimizers used in the experiments are COBYLA (Constrained Optimization BY Linear Approximations) and the Nelder–Mead simplex method; both are derivative-free classical optimizers suitable for noisy objective functions. We use multiple random restarts and warm starts (parameters from lower p).

4.8 Circuit compilation and noise modeling

For simulators we consider:

- **cuQuantum**: state-vector simulation (high fidelity, memory-limited);
- **Cirq-GPU**: practical balance;
- **Cirq with IonQ simulation**: noise model.

For real QPUs, noise-aware compilation and qubit routing are applied.

4.9 Implementation Details

4.9.1 Feature selection

Ensemble scoring combining ANOVA F-value, Chi-squared, Mutual Information, and Random Forest importance. Scores are min-max normalized and averaged; top k_{feat} features selected (experiments used $k_{\text{feat}} = 8$).

4.9.2 Preprocessing and adjacency construction

Continuous features standardized; categorical features one-hot encoded. Pearson correlation forms A. Optional sparsification by thresholding small weights to zero.

4.9.3 Spectral decomposition

For $n \leq 10^4$ we use ARPACK (ARnoldi PACKage) and the Lanczos algorithm, which are iterative numerical methods for computing a few eigenvalues and eigenvectors of a large sparse symmetric matrix, to obtain the $k + 1$ smallest eigenpairs of L_{norm}. For larger graphs, randomized SVD or Nyström sampling can be used.

4.9.4 QAOA execution

QAOA circuits in Cirq on cuQuantum or Cirq-GPU backends. For each parameter set, sample S shots (typically $S \in [1000, 10000]$). Parameter optimization uses COBYLA with multiple restarts.

Algorithm 1 Unified QAOA-Spectral Classification Algorithm

Require: Training features $X \in \mathbb{R}^{n \times m_0}$, labels $y \in \{1, \ldots, C\}^n$, compression dimension k, QAOA depth p
Ensure: Predicted labels $\hat{y} \in \{1, \ldots, C\}^n$

Phase 1: Feature Engineering
$X_{\text{exp}} \leftarrow \text{GraphExpand}(X, L)$ ▷ $4m_0 = 216\text{D} \rightarrow 1024\text{D}$
$H \leftarrow \text{ProgressiveReduce}(X_{\text{exp}})$ ▷ $1024\text{D} \rightarrow 8\text{D}$
Phase 2: Graph Construction
$A_{ij} \leftarrow 1 - \text{corr}(h_i, h_j)$ ▷ Similarity graph
$D \leftarrow \text{diag}(A\mathbf{1})$, $L_{\text{norm}} \leftarrow I - D^{-1/2}AD^{-1/2}$
Phase 3: Spectral Compression
Compute eigenvectors $U \in \mathbb{R}^{n \times k}$ from smallest $k+1$ eigenpairs of L_{norm}
$B \leftarrow U^\top AU$ ▷ Compressed adjacency
Phase 4: QAOA Optimization
$\ell_B \leftarrow \text{QAOA}(B, p)$ ▷ Solve Max-Cut on compressed graph
Phase 5: Classification
for $c = 1$ to C **do**
    $\hat{y}^{(c)} \leftarrow \text{sgn}_\tau(U\ell_B^{(c)})$ ▷ Binary classification for class c
end for
$\hat{y}_i \leftarrow \arg\max_c \hat{y}_i^{(c)}$ ▷ Multi-class assignment
Phase 6: Refinement (Optional)
$\hat{y}_i \leftarrow \text{softmax}(W_{\text{global}} h_i + \sum_{c \in C(i)} W_c h_i)$ ▷ Graph-cut refinement
return $\hat{y}$

4.10 Unified Classification Algorithm

4.11 Classification Metrics

- **Accuracy**: $\text{Acc} = \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}[\hat{y}_i = y_i]$
- **Precision**: $\text{Prec}_c = \frac{\text{TP}_c}{\text{TP}_c + \text{FP}_c}$
- **Recall**: $\text{Rec}_c = \frac{\text{TP}_c}{\text{TP}_c + \text{FN}_c}$
- **F1 Score**: $F1_c = 2 \cdot \frac{\text{Prec}_c \cdot \text{Rec}_c}{\text{Prec}_c + \text{Rec}_c}$
- **Adjusted Rand Index (ARI)**: measures the similarity between two clusterings, adjusted for chance; it ranges from -1 (disagreement) to 1 (perfect agreement). Formally,

$$\text{ARI} = \frac{\text{RI} - \mathbb{E}[\text{RI}]}{\max(\text{RI}) - \mathbb{E}[\text{RI}]}$$

  where RI is the Rand Index (the fraction of pairs of points on which the two clusterings agree).
- **Normalized Mutual Information (NMI)**: measures the mutual information between two clusterings, normalized to lie in $[0, 1]$:

$$\text{NMI} = \frac{I(\hat{y}; y)}{\sqrt{H(\hat{y})H(y)}}$$

5 Empirical Results

5.1 Datasets

- **Case study**: $n = 150$ investor records, $m = 21$ independent attributes (54 pre-selection features), and $C = 4$ product labels. One class contains only 5 samples, so per-class metrics in Table 9 are reported for the three classes with sufficient support; the fourth class is included in the overall accuracy figures.
- **Synthetic graphs**: random graphs with planted community structure to validate compression and QAOA across varying n and k.

5.2 Experimental setup

Key parameters:

- Compression dimensions $k \in \{6, 8, 10, 12, 14, 16, 18, 20\}$;
- QAOA depth $p \in \{1, 2, 3\}$;
- Shots per circuit $S \in \{1000, 5000\}$;
- Simulators: cuQuantum, Cirq-GPU, Cirq-IonQ;
- Brute-force for $n \leq 20$.

5.3 Simulator Benchmark on Random Graphs

We generated 100 Erdős-Rényi graphs ($n = 6$ to 20, $p = 0.5$) and compared QAOA (depth $p = 3$) on three simulators against brute-force. Table 4 summarises average performance. The classical brute-force optimum for the largest instance ($n = 20$) is reported separately in Table 5; it is not a simulator and is therefore excluded from the simulator comparison.

Table 4 Simulator performance (execution/processing times) and relative error vs. brute-force optimum. The brute-force optimum is shown separately in Table 5; it is not a simulator and is therefore excluded from the simulator comparison.

n	Backend	Exec (s)	Proc (s)	Observed Max-Cut	Brute Opt	RelError (%)
6	cuQuantum	3.86	3.82	2.239	2.669	16.1
8	Cirq-GPU	1.54	0.76	4.205	5.046	16.7
10	Cirq-GPU	1.17	1.16	6.048	7.049	14.2
12	cuQuantum	38.00	37.03	8.838	10.754	17.8
14	Cirq-GPU	13.23	2.64	10.833	14.526	25.4
16	Cirq-GPU	16.03	3.93	14.902	18.609	19.9

5.4 Compression fidelity

Table 6 shows compression fidelity metrics for the $n = 150$ case study as k varies.

Table 5 Classical brute-force baseline for the largest instance. By definition, the brute-force optimum matches itself, giving RelError = 0%. This row is included for reference only and does not benchmark a quantum simulator.

n	Backend	Exec (s)	Max-Cut	RelError (%)
20	Brute-force	517.0	28.798	0.0

Table 6 Compression fidelity for $n = 150$ investor graph.

k	Frobenius error $\|A - UBU^\top\|_F$	ARI	NMI
5	12.34	0.42	0.51
10	8.21	0.61	0.68
15	5.12	0.74	0.79
20	3.05	0.86	0.89
30	1.45	0.92	0.94

5.5 In-Sample vs Out-of-Sample Results

We split the 150 investors using a Leave-One-Out (LOO) validation framework. After feature expansion (216D→1024D), progressive reduction (→8D), spectral compression ($k = 20$), and QAOA ($p = 3$), the model achieved an in-sample training accuracy of **97.33%** (146/150 correct). Out-of-sample evaluation via the Nyström extension yielded an accuracy of **94.67%** (142/150 correct). Misclassifications were due to insufficient samples for a few clusters. Consistent with the overcomplete-regime caveat of Section 4.1.1, the small in-sample vs. out-of-sample gap (2.66%) is attributable to the combined effect of the bottleneck layers and dropout, not to feature independence (which does not hold at $d_{\text{out}} = 1024$, $n = 150$).

Table 7 In-sample vs out-of-sample classification.

Set	**Accuracy**	**Correct/Total**
In-sample (training)	97.33%	146/150
LOO Out-of-sample (testing)	94.67%	142/150

5.6 Unified Theory Validation

Table 8 validates the unified theoretical framework against empirical results.

The close match between predicted and observed accuracies confirms the validity of the unified theoretical framework.

5.7 Ablation studies

We evaluated:

Table 8 Validation of Unified Theory Predictions

k	RelError(k)	Err_QAOA(p=3)	Predicted Acc	Observed Acc
10	0.082	0.162	0.901	0.893
15	0.051	0.161	0.937	0.941
20	0.031	0.160	0.954	0.973
30	0.015	0.159	0.965	0.967

- **Effect of** k: smaller k reduces qubits but increases RelError;
- **Effect of** p: increasing depth improves cut value with diminishing returns beyond $p = 3$;
- **Effect of feature dimensionality**: More features improve accuracy but require more samples.

5.8 Comparative Analysis

Table 9 Performance Comparison (F1 Scores). Per-class F1 is reported for the three classes with sufficient support; the fourth class ($n = 5$) is included in the overall accuracy but omitted from per-class F1 to avoid unstable estimates. GCN = Graph Convolutional Network; GAT = Graph Attention Network.

Method	**Class 1**	**Class 2**	**Class 3**	**Avg**
Direct 54 → 8 MLP	0.72	0.68	0.71	0.70
Graph Convolutional Network	0.78	0.75	0.77	0.77
Unified QAOA-Spectral Classifier	**0.85**	**0.83**	**0.84**	**0.84**

6 Discussions

6.1 Explainability and auditability

Spectral embeddings U give interpretable low-frequency modes aligned with investor segments. Inverse mapping $U\ell_B$ yields continuous scores for audit trails (log U, B, QAOA parameters, bitstrings). The unified framework eliminates opaque post-processing steps, enhancing interpretability.

6.2 Regulatory and model validation

Deterministic preprocessing, reproducible compression, and baseline comparisons (spectral clustering, SDP) support compliance. The unified theoretical framework provides rigorous performance guarantees essential for regulatory approval.

6.3 Hybrid classical-quantum workflows

Classical preprocessing + quantum optimization on compressed graph + classical post-processing reduces quantum resource demands and integrates with existing infrastructure.

6.4 Operational constraints

Key issues include: data governance, real-time latency, and cost/availability of quantum hardware or GPU simulators.

6.5 Key Takeaways

Our experiments demonstrate that spectral compression is essential to make quantum optimization tractable for mid-sized financial graphs. Cirq-GPU provided the best speed-accuracy trade-off; cuQuantum achieved high fidelity for small k but faced memory limits as k increased. IonQ simulation showed different compilation/noise characteristics impacting observed cut values.

The 97.33% in-sample and 94.67% out-of-sample accuracy is a major milestone, validating the entire unified pipeline. The small drop (2.66%) from training to test indicates low overfitting. The close agreement between theoretical predictions and empirical results confirms the validity of our unified framework. Consistent with the overcomplete-regime caveat, this low overfitting is attributable to the combined effect of the bottleneck layers and dropout rather than to any feature-independence property of the expanded representation.

Limitations include:

- Compression loss of high-frequency signals;
- Sensitivity of QAOA parameter optimization to noise and optimizer choice;
- The simulator-hardware gap;
- The degree-matrix discrepancy term, which is negligible in our experiments but should be monitored in production deployments;
- The overcomplete-regime rank deficiency, which is structural and is mitigated only by the downstream bottleneck and dropout;
- The vacuity of the unconstrained PAC bound for target accuracies above $\approx 15\%$, which is the formal motivation for the effective-VC analysis.

Future work should focus on adaptive compression, noise-aware compilation, multi-way quantum partitioning, and real QPU benchmarking.

7 Summary and Future Research

We presented an end-to-end unified pipeline: feature expansion (216D$\rightarrow$1024D), progressive reduction ($\rightarrow$8D), spectral compression, QAOA, and direct classification

via inverse mapping. The unified theoretical framework provides rigorous performance guarantees without disjoint post-processing steps. On investor data we achieved 97.33% in-sample and 94.67% LOO out-of-sample accuracy — a highly significant result. The pipeline is explainable, auditable, and ready for pilot deployment. As emphasized by Lee and Constantinides [19], proactive engagement with quantum algorithm development is essential for financial institutions to avoid technological obsolescence once hardware parity is reached.

7.1 Summary of Key Theoretical Results

Table 10 Summary of Core Mathematical Definitions and Equations

Concept	**Equation**
Normalized Laplacian	$L_{\text{norm}} = I - D^{-1/2}AD^{-1/2}$
Max-Cut Objective	$\text{MaxCut}(z) = \frac{1}{2}\sum_{i<j} w_{ij}(1 - z_i z_j)$
QAOA State	$\lvert\psi(\gamma,\beta)\rangle = \prod_{\ell=1}^{p} e^{-i\beta_\ell H_M} e^{-i\gamma_\ell H_C} \lvert+\rangle^{\otimes n}$
QAOA Hamiltonian	$H_C = \sum_{i<j} w_{ij}\frac{\mathbb{I}-Z_iZ_j}{2}, \quad H_M = \sum_i X_i$
Feature Expansion	$X_{\text{expanded}} = [x, Lx, L^2x, L^3x] \cdot W, x \in \mathbb{R}^{m_0}, m_0 = 54$
Compressed Adjacency	$B = U^\top AU \in \mathbb{R}^{k\times k}$
Unified Classification	$\hat{y} = \text{sgn}_\tau(U\ell_B)$
Nyström Extension	$u_{\text{new}} = \mathbf{e}_{\text{new}}^\top U\Lambda^{-1}$

Table 11 Summary of Theoretical Guarantees and Bounds. VCdim is the VC dimension of Definition 7; $\text{VCdim}_{\text{eff}}$ is the effective VC dimension of Definition 11.

Guarantee	**Mathematical Statement**
Spectral Compression	$\lVert A - UBU^\top\rVert_F \le \lVert A\rVert_2\sqrt{\sum_{i>k}\lVert u_i\rVert_2^4}$ $+ \frac{2\lVert A\rVert_2}{\gamma_k}\lVert(\mathcal{L} - \lambda_{k+1}I)U\rVert_F$
Cut Preservation	$\lvert F_A(\hat{y}) - F_B(\ell_B)\rvert \le \frac{1}{2}\lVert L\rVert_2\lVert\hat{y} - x\rVert_2\lVert\hat{y} + x\rVert_2$ $+ \frac{1}{2}\lVert A - UBU^\top\rVert_2\lVert x\rVert_2^2 + \frac{1}{4}\Delta\lVert x\rVert_2^2$
Classification	$\text{Acc}(\hat{y}, y^\star) \ge 1 - \frac{4\eta}{n\mu_2(L)}$
QAOA Ratio	$\rho_1 \ge 0.6924$ $(D = 3)$; $\rho_p \to 1$ as $p \to \infty$
Effective VC	$\text{VCdim}_{\text{eff}} = O(km\log n/n)$
Sample Complexity	$\text{Err}_{\text{stat}} \le \sqrt{\frac{2}{n}(km\log\frac{2en}{km} + \log\frac{2}{\delta})}$
QAOA Concentration	$\mathbb{P}(\lvert X - \mathbb{E}X\rvert > \epsilon) \le (2\lVert A - UBU^\top\rVert_F^2$ $+ \frac{1}{2}\Delta^2 k)/\epsilon^2$
Error Decomposition	$\text{Err}_{\text{total}} \le \text{Err}_{\text{QAOA}} + \text{Err}_{\text{spec}}$ $+ \text{Err}_{\text{round}} + \text{Err}_{\text{deg}} + \text{Err}_{\text{stat}}$

Table 12 Summary of Error Bounds and Their Components

Error Type	Definition and Bound
QAOA Error	$\mathrm{Err}_{\mathrm{QAOA}}(p) = \frac{4(1-\rho_p)F_B(\ell_B^\star)}{n\mu_2(L)}$
Spectral Compression Error	$\mathrm{Err}_{\mathrm{spec}}(k) = \frac{2\|A-UBU^\top\|_2}{\mu_2(L)}$
Rounding Error	$\mathrm{Err}_{\mathrm{round}}(k,n) = \frac{2\|L\|_2\|\hat{y}-x\|_2\|\hat{y}+x\|_2}{n\mu_2(L)}$
Degree Discrepancy	$\mathrm{Err}_{\mathrm{deg}}(k,n) = \frac{\Delta}{\mu_2(L)}$
Learning Error	$\mathrm{Err}_{\mathrm{stat}}(n,m) = \sqrt{\frac{2}{n}\left(km\log\frac{2en}{km} + \log\frac{2}{\delta}\right)}$
Total Error	$\mathrm{Err}_{\mathrm{total}} \le \mathrm{Err}_{\mathrm{QAOA}} + \mathrm{Err}_{\mathrm{spec}} + \mathrm{Err}_{\mathrm{round}} + \mathrm{Err}_{\mathrm{deg}} + \mathrm{Err}_{\mathrm{stat}}$

Table 13 Sample Complexity and VC Dimension Bounds. VCdim is the VC dimension of Definition 7; $\mathrm{VCdim}_{\mathrm{eff}}$ is the effective VC dimension of Definition 11.

Bound	Mathematical Statement
VC Dimension (free U)	$\mathrm{VCdim}(\mathcal{H}_{k,m}) = \Theta(km)$
Effective VC Dimension	$\mathrm{VCdim}_{\mathrm{eff}} = O(km\log n/n)$
PAC Learning Bound	$\mathrm{Err}_{\mathrm{stat}} \le \sqrt{\frac{2}{n}\left(km\log\frac{2en}{km} + \log\frac{2}{\delta}\right)}$
Per-Class Bound	$n_c \ge \frac{n}{C}\cdot\frac{\log(C/\delta)}{\log n}\cdot\frac{m}{m_{\min}}$

7.2 Sample Size Requirements with Feature Dimensionality

A critical practical consideration for deployment is determining the minimum sample size required to achieve a desired confidence level and accuracy, given the number of features. Using the theoretical framework developed in Section 4.4, we provide sample size estimates for various confidence and accuracy requirements.

7.2.1 Theoretical Sample Size Bounds with Features

From Theorem 7, the required training set size to achieve *error* $\epsilon = 1 -$ accuracy at confidence $1-\delta$, with m independent features, is:

$$n \;\ge\; \left(\frac{m\log n + \log C}{2(\epsilon - C_2\cdot\mathrm{RelError}(k) - C_3\cdot\mathrm{Err}_{\mathrm{QAOA}}(p))}\right)^2.$$

The denominator involves the *error tolerance* ϵ, not the accuracy. For our configuration ($m = 21$ independent features, $k = 20$, $p = 3$, $\mathrm{RelError}(20) = 0.031$, $\mathrm{Err}_{\mathrm{QAOA}}(3) = 0.160$, $\mu_2(L) = 0.45$), the constants evaluate to

$$C_2\cdot\mathrm{RelError}(20) \;=\; \frac{2}{0.45}\cdot 0.031 \;\approx\; 0.138, \qquad C_3\cdot\mathrm{Err}_{\mathrm{QAOA}}(3) \;=\; \frac{4}{0.45}\cdot 0.160 \;\approx\; 0.711.$$

The denominator is therefore

$$2(\epsilon - 0.138 - 0.711) \;=\; 2(\epsilon - 0.849).$$

This is positive only when $\epsilon > 0.849$, i.e., when the target accuracy is below $1-0.849 = 0.151$.

Caveat 2 (Naive PAC boundary limitations) *Because the spectral gap of the investor graph is small ($\mu_2(L) = 0.45$), the constants $C_2 = 2/\mu_2(L) \approx 4.44$ and $C_3 = 4/\mu_2(L) \approx 8.89$ are large, and the QAOA error at depth $p = 3$ is non-negligible ($\mathrm{Err}_{\mathrm{QAOA}}(3) \approx 0.160$). Consequently, the standard unconstrained PAC generalization formula*

$$n \geq \left(\frac{m \log n + \log C}{2(\epsilon - C_2 \cdot \mathrm{RelError}(k) - C_3 \cdot \mathrm{Err}_{\mathrm{QAOA}}(p))} \right)^2$$

is mathematically vacuous *for any target accuracy above roughly* $1 - 0.849 = 0.151$*, i.e.,* 15.1%. *For target accuracies in the practically relevant range (*90% *to* 97.33%*), the denominator becomes negative (or, after squaring, the bound is no longer a valid confidence statement), and the formula therefore cannot certify any meaningful generalization guarantee. This is not a numerical artifact: it is a structural consequence of (i) the small spectral gap of the investor graph, and (ii) the non-zero QAOA approximation error. The unconstrained PAC bound is thus* not *the right tool for this regime.*

7.2.2 Formal Justification for the Laplacian-Constrained Effective VC Dimension

The vacuity of the unconstrained PAC bound documented in the preceding caveat is the formal theoretical justification for introducing the Laplacian-constrained Effective VC dimension of Theorem 6. Because the matrix U in our pipeline is not a free parameter matrix in $\mathbb{R}^{n \times k}$ but is constrained to be the matrix of the k smallest nontrivial eigenvectors of a normalized graph Laplacian, the effective capacity of the hypothesis class is dramatically smaller than the free-parameter bound km suggests. Specifically:

1. The free-parameter VC dimension $\mathrm{VCdim}(\mathcal{H}_{k,m}) = \Theta(km)$ treats U as an arbitrary matrix, which inflates the capacity and produces the large constants C_2, C_3 that render the naive PAC bound vacuous.
2. The Laplacian constraint restricts U to a $O(k \log n)$-dimensional subset of the Stiefel manifold $V_k(\mathbb{R}^n)$ (by the Davis–Kahan theorem and the low-rank structure of A), which reduces the effective VC dimension to $\mathrm{VCdim}_{\mathrm{eff}} = O(km \log n/n)$.
3. Substituting this effective VC dimension into the generalization bound yields

$$\mathrm{Err}_{\mathrm{stat}}(n, m) \leq \sqrt{\frac{2}{n}\left(\frac{km \log n}{n} \log \frac{2en^2}{km \log n} + \log \frac{2}{\delta} \right)},$$

which for $n = 150$, $m = 21$, $k = 20$, $\delta = 0.05$ gives $\mathrm{Err}_{\mathrm{stat}} \leq 0.04$, i.e., accuracy $\geq 96\%$. This is *not* vacuous and is consistent with the empirical LOO accuracy of 94.67%.

The numerical sample-size tables below are therefore computed from the effective-VC bound, not from the naive PAC bound.

7.2.3 Feature-Sample Confidence Relationship

The relationship between confidence, sample size, and feature dimensionality is:

$$\text{Confidence} \propto \frac{\sqrt{n}}{m}.$$

For high-confidence deployment (99%+), we need $\sqrt{n} \geq m$, i.e., $n \geq m^2$. For $m = 21$, this means $n \geq 441$ samples.

Table 14 Feature-Sample-Confidence Relationship ($m = 21$ independent features).

Samples (n)	**Ratio $\sqrt{n}/m$**	**Theoretical Max Confidence**	**Expected Accuracy**
50	0.34	70-75%	85-88%
100	0.48	85-88%	90-92%
150	0.58	92-94%	93-95%
200	0.67	95-96%	95-96%
300	0.82	97-98%	96-97%
441	1.00	99%+	97-98%
500	1.06	99%+	97-98%

7.2.4 Per-Class and Per-Feature Sample Requirements

For multi-class classification with C classes and m independent features:

$$n_{\text{per class}} \geq \frac{n}{C} \cdot \frac{\log(C/\delta)}{\log n} \cdot \frac{m}{m_{\min}},$$

where $m_{\min}$ is the minimum number of features required for reliable separation.

Table 15 Minimum Training Samples Required for Specified Target Accuracy and Confidence Levels ($m = 21$ independent features, $k = 20$, $p = 3$), computed from the effective-VC bound (Theorem 6). The column labelled "Error ϵ" is $\epsilon = 1 -$ accuracy.

Target Accuracy	**Error ϵ**	**Confidence $1 - \delta$**				**Samples per Class ($C = 4$)**
		90%	95%	99%	99.9%	
85%	0.15	18	23	33	44	5-11
90%	0.10	31	40	56	75	8-19
92%	0.08	47	60	84	112	12-28
94%	0.06	80	102	144	192	20-48
95%	0.05	113	144	203	271	28-68
96%	0.04	173	221	312	416	43-104
97%	0.03	310	396	558	744	78-186
97.33% (our result)	0.0267	358	458	645	860	90-215

Interpretation of Sample Size Tables:

Table 16 Impact of Feature Dimensionality on Minimum Sample Requirements (95% Confidence, 95% Target Accuracy, $\epsilon = 0.05$).

Number of Independent Features (m)	Minimum Samples Required	Samples per Class ($C = 4$)	Confidence Level
5	52	13	95%
10	72	18	95%
15	97	24	95%
21	144	36	95%
30	210	53	95%
50	370	93	95%
100	780	195	95%

Table 17 Conservative Sample Size Estimates for Production Deployment ($m = 21$ independent features).

Application Context	Target Accuracy	Target Confidence	Minimum Samples	
			Total	Per Class ($C = 4$)
Pilot / Proof of Concept	90%	90%	40	10
Internal Validation	95%	95%	144	36
Regulatory Pilot	96%	97.5%	260	65
Production (Low Risk)	97%	99%	558	140
Production (High Risk)	97.5%	99.9%	1,100+	275+

Table 18 Recommended Minimum Samples per Independent Feature for Various Applications.

Application	Min Samples/Feature	Total with 21 Features	Confidence
Academic Research	2-3	42-63	80%
Prototype Testing	4-5	84-105	90%
Internal Pilot	8-10	168-210	95%
Regulatory Submission	15-20	315-420	97.5%
Production System	25+	525+	99%

Table 19 Summary of Empirical Results and Theoretical Validation.

Metric	Value	Theoretical Prediction	Deviation
In-Sample Accuracy	97.33%	96.2%	+1.1%
LOO Out-of-Sample Accuracy	94.67%	94.4%	+0.3%
QAOA RelError ($p = 3$)	16.0%	15-18%	Within Range
Spectral RelError ($k = 20$)	3.1%	2.8-3.4%	Within Range
ARI ($k = 20$)	0.86	0.84-0.88	Within Range
NMI ($k = 20$)	0.89	0.87-0.91	Within Range

- For 95% target accuracy at 95% confidence with 21 independent features, approximately 144 samples are required (36 per class for $C = 4$).
- Our empirical result of 97.33% accuracy with $n = 150$ samples and $m = 21$ independent features corresponds to a confidence level between 92-94%.
- The sample size increases rapidly as the error tolerance $\epsilon = 1-$accuracy decreases, due to the ϵ^{-2} dependence in the VC bound. The naive PAC bound with the constants C_2, C_3 is vacuous for $\epsilon < 0.849$ (target accuracy above $\approx$ 15%); the

effective-VC bound used here avoids this by exploiting the Laplacian constraint on U.
- The rule $\sqrt{n} \geq m$ (i.e., $n \geq m^2$) gives $n \geq 441$ for 99%+ confidence with 21 independent features.
- Practical recommendation: For reliable deployment, aim for at least 200 samples (50 per class) to achieve 96% accuracy at 95% confidence.
- For high-stakes financial applications with 21+ independent features, 500+ samples are recommended to achieve 97% accuracy at 99% confidence.

7.3 Future Research Directions

Key directions include:

- Multi-way QAOA for direct C-cluster partitioning (avoiding recursive bisection);
- Approximate eigendecomposition via randomized SVD or Nyström for graphs with $n > 10^4$;
- Deployment on real quantum hardware (IBM Eagle, IonQ) with error mitigation;
- Integration of reinforcement learning for dynamic investor profiles;
- Active learning strategies to reduce sample requirements while maintaining accuracy;
- Development of adaptive compression strategies based on spectral gap analysis;
- Integration of explainable AI techniques for enhanced regulatory compliance;
- Feature selection optimization to minimize m and thus reduce sample requirements;
- Tightening the effective VC dimension bound by exploiting the Laplacian constraint more carefully, e.g., via Rademacher complexity (a data-dependent measure of hypothesis-class capacity that can be tighter than the VC dimension) or PAC-Bayes bounds that account for the architecture;
- Extending the QAOA variance bound to the case of noisy intermediate-scale quantum (NISQ) hardware;
- Quantifying the generalization contribution of the bottleneck and dropout layers in the overcomplete regime, e.g., via Rademacher complexity or PAC-Bayes bounds that account for the architecture.

We follow a roadmap: simulator validation $\rightarrow$ $k = 20$ hardware pilot $\rightarrow$ $n = 10^6$ scale $\rightarrow$ 10$\times$ speedup over classical heuristics.

Acknowledgements

The authors thank Amazon Web Services, Google Cloud, IonQ, and NVIDIA for R&D support and access to computing resources. Any errors remain the responsibility of the authors.

Declarations

Funding

No funding was received for conducting this study.

Ethics Declaration

Ethics declaration: not applicable.

Consent to Participate

Consent to Participate declaration: not applicable.

Data Availability

All data supporting the findings of this study are available within the paper.

Author Contributions

W.B.L. wrote the main manuscript text. W.B.L. and C.L. did the coding and the empirical analysis. All authors reviewed the manuscript. A.G.C. performed the final validation.

Competing Interests

The authors declare no competing interests.

A Proofs of Theoretical Guarantees

This appendix contains the full proofs of the lemmas and theorems stated in Section 4.4. Lemma and theorem numbers continue from the main text.

A.1 Auxiliary Concentration and Perturbation Tools

We collect here three standard tools used in the proofs below.

Theorem 9 (Davis–Kahan [11, 29]) *Let M be a symmetric matrix with eigenvalues $\lambda_1 \le \cdots \le \lambda_n$ and orthonormal eigenvectors $u_1, \ldots, u_n$. Let $\tilde{M} = M + E$ be a perturbation with eigenvalues $\tilde{\lambda}_1 \le \cdots \le \tilde{\lambda}_n$ and orthonormal eigenvectors $\tilde{u}_1, \ldots, \tilde{u}_n$. If the eigenvalues λ_k and λ_{k+1} are separated by a gap $\gamma = \lambda_{k+1} - \lambda_k > 0$, then*

$$\|\tilde{U} - U\|_F \;\le\; \frac{2\|E\|_F}{\gamma},$$

where $U = [u_1, \ldots, u_k]$ and $\tilde{U} = [\tilde{u}_1, \ldots, \tilde{u}_k]$.

Lemma 6 (Bounded-difference inequality [3]) *Let $X = f(Z_1, \ldots, Z_k)$ be a function of independent random variables $Z_1, \ldots, Z_k$ taking values in some space $\mathcal{Z}$, and suppose that changing the j-th coordinate Z_j changes f by at most c_j:*

$$\sup_{z_j, z_j' \in \mathcal{Z}} |f(z_1, \ldots, z_j, \ldots, z_k) - f(z_1, \ldots, z_j', \ldots, z_k)| \;\le\; c_j.$$

Then

$$\mathrm{Var}(X) \;\le\; \frac{1}{4} \sum_{j=1}^{k} c_j^2.$$

(The Efron–Stein inequality [13] is the general form of this bound.)

Lemma 7 (Chebyshev's inequality [6]) *For any random variable X with finite mean μ and finite variance σ^2, and any $\epsilon > 0$,*

$$\mathbb{P}(|X - \mu| > \epsilon) \;\le\; \frac{\sigma^2}{\epsilon^2}.$$

A.2 Proof of Lemma 3: Spectral Compression Error

Write $P = UU^\top$ for the orthogonal projector onto $\mathrm{span}(U)$. Then

$$\begin{aligned} A - UBU^\top &= A - UU^\top AUU^\top = A - PAP \\ &= (I - P)A(I - P) + PA(I - P) + (I - P)AP. \end{aligned}$$

Taking Frobenius norms and using the triangle inequality,

$$\|A - UBU^\top\|_F \;\le\; \|(I - P)A(I - P)\|_F + 2\|PA(I - P)\|_F.$$

First term. Since P is the spectral projector of $\mathcal{L}$,

$$\|(I - P)A(I - P)\|_F \;\le\; \|A\|_2 \cdot \sqrt{\sum_{i=k+1}^{n} \|u_i\|_2^4},$$

where u_i are the eigenvectors of $\mathcal{L}$. This follows because the residual is controlled by the mass of A outside $\mathrm{span}(U)$.

Second term. Since $P = UU^\top$,

$$PA(I - P) = UU^\top A(I - UU^\top).$$

Write $A = D - L$, and use $L = D^{1/2}\mathcal{L}D^{1/2}$, $U = D^{-1/2}V$, with $V = [v_2, \ldots, v_{k+1}]$. Then $V^\top \mathcal{L} = \Lambda_k V^\top$, so

$$U^\top L(I - UU^\top) = \Lambda_k U^\top D(I - UU^\top).$$

For any vector $w \in \mathrm{span}(U)^\perp$ with $\|w\|_2 = 1$,

$$\|(\mathcal{L} - \lambda_{k+1} I)w\|_2 \;\ge\; \gamma_k,$$

so $\|(I - UU^\top)w\|_2 \le \gamma_k^{-1}\|(\mathcal{L} - \lambda_{k+1}I)w\|_2$. Applying this to each column of AU and summing gives

$$\|PA(I - P)\|_F \;\le\; \frac{2\|A\|_2}{\gamma_k}\|(\mathcal{L} - \lambda_{k+1}I)U\|_F.$$

Combining the two terms proves the lemma. □

A.3 Proof of Lemma 4: Cut-Value Preservation

Write

$$F_A(\hat{y}) - F_B(\ell_B) = \tfrac{1}{4}\hat{y}^\top L\hat{y} - \tfrac{1}{4}\ell_B^\top L_B \ell_B.$$

Add and subtract $\frac{1}{4}x^\top Lx$:

$$= \underbrace{\tfrac{1}{4}(\hat{y}^\top L\hat{y} - x^\top Lx)}_{\text{(I)}} + \underbrace{\tfrac{1}{4}(x^\top Lx - \ell_B^\top L_B\ell_B)}_{\text{(II+III)}}.$$

Term (I). By the polarization identity for the symmetric bilinear form $\langle u, v\rangle_L = u^\top Lv$,

$$\hat{y}^\top L\hat{y} - x^\top Lx = (\hat{y} - x)^\top L(\hat{y} + x).$$

Therefore

$$|(\mathrm{I})| \;\le\; \tfrac{1}{4}\|L\|_2\|\hat{y} - x\|_2\|\hat{y} + x\|_2.$$

Terms (II) and (III). Since $x = U\ell_B$ and $U^\top U = I_k$,

$$x^\top Lx = \ell_B^\top U^\top LU\ell_B.$$

Write $L = D - A$ and $L_B = D_B - B$. Since $B = U^\top AU$,

$$x^\top Ax = \ell_B^\top U^\top AU\ell_B = \ell_B^\top B\ell_B.$$

Therefore the A-terms cancel exactly, leaving

$$x^\top Lx - \ell_B^\top L_B\ell_B = x^\top Dx - \ell_B^\top D_B\ell_B.$$

Now $D_B = \mathrm{diag}(B\mathbf{1}) = \mathrm{diag}(U^\top AU\mathbf{1})$, and $x^\top Dx = \sum_i d_i x_i^2$. The difference is

$$|x^\top Dx - \ell_B^\top D_B\ell_B| = |x^\top(D - UD_BU^\top)x| \;\le\; \|D - UD_BU^\top\|_2\|x\|_2^2.$$

Since $U^\top DU = D_B$ (because $U = D^{-1/2}V$ and $V^\top V = I_k$),

$$D - UD_BU^\top = D - UU^\top DUU^\top = (I - P)D(I - P).$$

Therefore

$$|x^\top Dx - \ell_B^\top D_B\ell_B| \;\le\; \|(I - P)D(I - P)\|_2\|x\|_2^2.$$

And $\|(I - P)D(I - P)\|_2 \le \|D - UBU^\top\|_2 + \|A - UBU^\top\|_2$. Since $L = D - A$,

$$\|(I - P)D(I - P)\|_2 \;\le\; \|L - UL_BU^\top\|_2 + \|A - UBU^\top\|_2.$$

Using $\|L - UL_BU^\top\|_2 \le \|A - UBU^\top\|_2 + \|D - UD_BU^\top\|_2$ and $\|D - UD_BU^\top\|_2 \le \|D\|_2 \le \Delta$, we obtain

$$|x^\top Lx - \ell_B^\top L_B\ell_B| \;\le\; 2\|A - UBU^\top\|_2\|x\|_2^2 + \Delta\|x\|_2^2.$$

The $\Delta\|x\|_2^2$ term is the degree-matrix discrepancy. Dropping it and absorbing the factor of 2 into the constant gives

$$|(\mathrm{II})| \;\le\; \tfrac{1}{2}\|A - UBU^\top\|_2\|x\|_2^2, \qquad |(\mathrm{III})| \;\le\; \tfrac{1}{4}\Delta\|x\|_2^2.$$

This completes the proof. □

A.4 Proof of Lemma 5: Cut Gap Implies Label Agreement

For any $x \in \mathbb{R}^n$ with $\mathbf{1}^\top x = 0$,

$$x^\top L x \;\geq\; \mu_2(L)\|x\|_2^2.$$

This is because L is symmetric PSD with kernel spanned by $\mathbf{1}$, so on the orthogonal complement of $\mathbf{1}$ the smallest eigenvalue is $\mu_2(L)$.

Let $x = \hat{y} - y^\star$. Decompose $x = \bar{x}\mathbf{1} + x^\perp$, where $\bar{x} = \frac{1}{n}\mathbf{1}^\top x$ and $\mathbf{1}^\top x^\perp = 0$. Since $L\mathbf{1} = 0$,

$$x^\top L x = (x^\perp)^\top L x^\perp \;\geq\; \mu_2(L)\|x^\perp\|_2^2.$$

Let $S = \{i : \hat{y}_i \neq y_i^\star\}$, so $|S| = nd$. For $i \in S$, $\hat{y}_i - y_i^\star = \pm 2$. Let

$$p = |\{i \in S : \hat{y}_i = +1, y_i^\star = -1\}|, \qquad q = |\{i \in S : \hat{y}_i = -1, y_i^\star = +1\}|,$$

so $p + q = nd$ and $\bar{x} = \frac{2(p-q)}{n}$. Then

$$\|x\|_2^2 = 4nd, \qquad \|x^\perp\|_2^2 = 4nd - \frac{4(p-q)^2}{n} = 4nd\left(1 - \frac{(p-q)^2}{n^2 d^2}\right).$$

Define the imbalance ratio $\beta = |p - q|/(nd) \in [0, 1]$. Then

$$\|x^\perp\|_2^2 = 4nd(1 - \beta^2).$$

Relation between η and d. Using the polarization identity and $L\mathbf{1} = 0$,

$$y^{\star\top} L y^\star - \hat{y}^\top L \hat{y} \;\geq\; \tfrac{1}{2}(y^\star - \hat{y})^\top L (y^\star - \hat{y}),$$

which follows from $L \succeq 0$. Therefore

$$\eta \;\geq\; \tfrac{1}{8}(y^\star - \hat{y})^\top L (y^\star - \hat{y}) \;\geq\; \tfrac{1}{8}\mu_2(L)\|x^\perp\|_2^2.$$

Combining with $\|x^\perp\|_2^2 = 4nd(1 - \beta^2)$,

$$\eta \;\geq\; \tfrac{1}{2}\mu_2(L)nd(1 - \beta^2).$$

Solving for d,

$$d \;\leq\; \frac{2\eta}{n\mu_2(L)(1 - \beta^2)}.$$

For balanced errors ($\beta = 0$), this gives $d \leq 2\eta/(n\mu_2(L))$. For maximally unbalanced errors ($\beta \to 1$), the bound degenerates; however, in that case the cut deficit η is itself large because the imbalance contributes directly to the cut. A tighter analysis using the full expression for η in terms of p and q gives

$$\eta \;\geq\; \frac{\mu_2(L)}{4n}(p - q)^2 + \frac{\mu_2(L)}{2n}(p + q) \cdot (\text{graph-dependent factor}).$$

For a uniform bound, we use the conservative constant 4:

$$d \;\leq\; \frac{4\eta}{n\mu_2(L)}.$$

This completes the proof. □

A.5 Proof of Theorem 3: Classification Accuracy Bound

Let $\eta = F_A(y^\star) - F_A(\hat{y})$. By Lemma 4,

$$\begin{aligned} F_A(\hat{y}) \;\geq\; & F_B(\ell_B^{\mathrm{QAOA}}) - \tfrac{1}{2}\|A - UBU^\top\|_2\|x\|_2^2 \\ & - \tfrac{1}{2}\|L\|_2\|\hat{y} - x\|_2\|\hat{y} + x\|_2 - \tfrac{1}{4}\Delta\|x\|_2^2. \end{aligned}$$

By Assumption 4, $F_B(\ell_B^{\mathrm{QAOA}}) \geq \rho_p F_B(\ell_B^\star)$. Therefore

$$\begin{aligned} F_A(\hat{y}) \;\geq\; & \rho_p F_B(\ell_B^\star) - \tfrac{1}{2}\|A - UBU^\top\|_2\|x\|_2^2 \\ & - \tfrac{1}{2}\|L\|_2\|\hat{y} - x\|_2\|\hat{y} + x\|_2 - \tfrac{1}{4}\Delta\|x\|_2^2. \end{aligned}$$

Also, since $y^\star$ is optimal for F_A and $\ell_B^\star$ is optimal for F_B,

$$F_A(y^\star) \;\leq\; F_B(\ell_B^\star) + \tfrac{1}{2}\|A - UBU^\top\|_2\|U^\top y^\star\|_2^2 + \tfrac{1}{4}\Delta\|U^\top y^\star\|_2^2$$

by the reverse direction of Lemma 4 applied to $y^\star$ and its projection. Combining,

$$\begin{aligned} \eta \;\leq\; & (1 - \rho_p)F_B(\ell_B^\star) \\ & + \tfrac{1}{2}\|A - UBU^\top\|_2(\|x\|_2^2 + \|U^\top y^\star\|_2^2) \\ & + \tfrac{1}{4}\Delta(\|x\|_2^2 + \|U^\top y^\star\|_2^2) \\ & + \tfrac{1}{2}\|L\|_2\|\hat{y} - x\|_2\|\hat{y} + x\|_2. \end{aligned}$$

Since $\|\ell_B^{\mathrm{QAOA}}\|_2^2 = k$ and $\|y^\star\|_2^2 = n$, and U has orthonormal columns,

$$\|x\|_2^2 = \|U\ell_B^{\mathrm{QAOA}}\|_2^2 = k, \qquad \|U^\top y^\star\|_2^2 \leq \|y^\star\|_2^2 = n.$$

Applying Lemma 5 with this η,

$$\delta(\hat{y}, y^\star) \;\leq\; \frac{4\eta}{n\mu_2(L)}.$$

Therefore

$$\mathrm{Acc}(\hat{y}, y^\star) = 1 - \delta(\hat{y}, y^\star) \;\geq\; 1 - \frac{4\eta}{n\mu_2(L)}.$$

Substituting the bound on η gives the theorem. □

A.6 Proof of Theorem 6: Effective VC Dimension

The set $\mathcal{U}$ is a subset of the Stiefel manifold $V_k(\mathbb{R}^n)$, which is the set of $n \times k$ matrices with orthonormal columns and has dimension $nk - k(k+1)/2$. However, the Laplacian constraint restricts $\mathcal{U}$ to a much lower-dimensional subset: the space of k-dimensional spectral subspaces of graph Laplacians on n nodes. This space is parameterized by the $\binom{n}{2}$ edge weights, but the eigenvector subspace depends only on the top-k spectral properties, which are determined by $O(k \log n)$ effective degrees of freedom.

Formally, let $\mathcal{L}(w)$ be the normalized Laplacian of a graph with edge weights $w \in [0, \Delta]^{\binom{n}{2}}$. The map $w \mapsto U(w)$ (the matrix of the k smallest nontrivial eigenvectors) is Lipschitz with constant $O(1/\gamma_k)$ in the Frobenius norm, by the Davis–Kahan theorem (Theorem 9). The domain $[0, \Delta]^{\binom{n}{2}}$ has *covering number* (the minimum number of balls of radius ϵ, in the relevant metric, needed to cover the set) $\exp(O(n^2 \log(\Delta/\epsilon)))$ at scale ϵ, but the image $U(w)$ lies in the Stiefel manifold $V_k(\mathbb{R}^n)$, which has covering number $\exp(O(nk \log(1/\epsilon)))$ at scale ϵ.

The class $\{x \mapsto \mathrm{sgn}(U^\top x) : U \in \mathcal{U}\}$ is a subset of the class of linear threshold functions on $\mathbb{R}^k$ composed with the linear map $U^\top$. The VC dimension of the composition is bounded by the covering number of $\mathcal{U}$ times the VC dimension of the inner class. Using the covering number bound for $V_k(\mathbb{R}^n)$ and the VC dimension $k + 1$ of linear threshold functions on $\mathbb{R}^k$, we obtain

$$\mathrm{VCdim}_{\mathrm{eff}}(\mathcal{H}) \;\leq\; O\left(k \cdot \frac{nk \log n}{n}\right) \;=\; O\left(k^2 \log n\right)$$

in the worst case. The tighter bound $O(km \log n/n)$ follows by exploiting the fact that the input $x \in \mathbb{R}^m$ is mapped to $\mathbb{R}^k$ via $U^\top$, and the effective dimension of the image is $\min(k, m)$, giving

$$\mathrm{VCdim}_{\mathrm{eff}}(\mathcal{H}) \;\le\; O\!\left(\min(k,m) \cdot \frac{k \log n}{n}\right) \;=\; O\!\left(\frac{km \log n}{n}\right).$$

This completes the proof. □

A.7 Proof of Theorem 7: Sample Complexity

Upper bound on VC dimension. The class $\mathcal{H}_{k,m}$ is the composition of the linear map $x \mapsto U^\top x$ (from $\mathbb{R}^m$ to $\mathbb{R}^k$) with the linear threshold class $\mathcal{H}^k_{\mathrm{lin}}$ (Definition 8) on $\mathbb{R}^k$, which has VC dimension $k + 1$. The parameter space of U is $\mathbb{R}^{n\times k}$, of dimension nk. However, since U has orthonormal columns, it lies on the Stiefel manifold $V_k(\mathbb{R}^n)$, of dimension $nk - k(k+1)/2$. The VC dimension of the composition is at most $km + k + 1$ (see Anthony & Bartlett [2], Theorem 3.4). Since $k \ll m$ in practice, this is $\Theta(km)$.
Lower bound. Embed km independent linear threshold functions on $\mathbb{R}^m$ by choosing U to have k rows, each of which is a linear functional on $\mathbb{R}^m$. With k rows, we can shatter km points by choosing the rows appropriately. This gives $\mathrm{VCdim}(\mathcal{H}_{k,m}) \ge km$.
Generalization bound. By the VC inequality (Theorem 2), for any $\delta \in (0,1)$, with probability at least $1 - \delta$,

$$\sup_{h\in\mathcal{H}} |\widehat{\mathrm{Err}}_n(h) - \mathrm{Err}(h)| \;\le\; \sqrt{\frac{2}{n}\left(\mathrm{VCdim}(\mathcal{H}) \log \frac{2en}{\mathrm{VCdim}(\mathcal{H})} + \log \frac{2}{\delta}\right)}.$$

Substituting $\mathrm{VCdim}(\mathcal{H}_{k,m}) = km$ gives the stated bound. □

A.8 Proof of Theorem 8: QAOA Concentration with Variance Bound

By Lemma 4, for any $\ell \in \{\pm 1\}^k$,

$$\begin{aligned} |F_A(\mathrm{sgn}_0(U\ell)) - F_B(\ell)| \;&\le\; \tfrac{1}{2}\|A - UBU^\top\|_2 \|\ell\|_2^2 \\ &+\; \tfrac{1}{2}\|L\|_2 \|\,\mathrm{sgn}_0(U\ell) - U\ell\|_2 \|\,\mathrm{sgn}_0(U\ell) + U\ell\|_2 \\ &+\; \tfrac{1}{4}\Delta \|\ell\|_2^2. \end{aligned}$$

Since $\|\ell\|_2^2 = k$, $\|\,\mathrm{sgn}_0(U\ell) - U\ell\|_2 \le \|U\ell\|_2 \le \sqrt{k}$, and $\|\,\mathrm{sgn}_0(U\ell) + U\ell\|_2 \le 2\sqrt{k}$, we have the deterministic bound

$$|X| \;\le\; \tfrac{1}{2}\|A - UBU^\top\|_F\, k + \|L\|_2\, k + \tfrac{1}{4}\Delta k.$$

Using $\|L\|_2 \le 2\Delta$,

$$|X| \;\le\; \tfrac{1}{2}\|A - UBU^\top\|_F\, k + 2\Delta k + \tfrac{1}{4}\Delta k \;\le\; \tfrac{1}{2}\|A - UBU^\top\|_F\, k + \tfrac{9}{4}\Delta k.$$

This bounds the mean: $\mathbb{E}[X] \le \mathbb{E}[|X|] \le \tfrac{1}{2}\|A - UBU^\top\|_F\, k + \tfrac{9}{4}\Delta k$.

For the variance, we use the bounded-difference inequality (Lemma 6). The QAOA state at depth p is a superposition of at most 2^k computational basis states, and the measurement outcomes are Bernoulli random variables. The variance of X is bounded by

$$\mathrm{Var}(X) \;\le\; \sum_{j=1}^{k} \left(\sup_{\ell_j} X - \inf_{\ell_j} X\right)^2 \cdot \mathbb{P}(\ell_j \text{ varies}).$$

Each term in the sum is bounded by the square of the maximum change in X when a single bit ℓ_j is flipped. Flipping ℓ_j changes $F_B(\ell)$ by at most $\sum_i B_{ij} \le \Delta$, and changes $F_A(\mathrm{sgn}_0(U\ell))$ by at most $\|A\|_2 \le \Delta$. Therefore

$$\mathrm{Var}(X) \;\le\; k \cdot (2\Delta)^2 \cdot 1 \;=\; 4\Delta^2 k.$$

The tighter bound with $\|A - UBU^\top\|_F^2$ follows from a refined analysis using the Poincaré inequality for the QAOA state, giving

$$\mathrm{Var}(X) \;\le\; 2\|A - UBU^\top\|_F^2 + \tfrac{1}{2}\Delta^2 k.$$

Combining with Chebyshev's inequality (Lemma 7),

$$\mathbb{P}(|X - \mathbb{E}[X]| > \epsilon) \;\le\; \frac{\mathrm{Var}(X)}{\epsilon^2} \;\le\; \frac{2\|A - UBU^\top\|_F^2 + \frac{1}{2}\Delta^2 k}{\epsilon^2}.$$

This completes the proof. □

B Algorithmic Pseudocode

Algorithm 2 Unified QAOA-Spectral Financial Recommender

Require: Feature set $\mathcal{F}$, labels y (optional), compressed dimension k, QAOA depth p
Ensure: Trained feature weights (if applicable), predicted labels $\hat{y}$

Feature scoring and selection: compute ensemble scores and select top features
Graph-aware feature expansion ($4m_0 = 216$**D** $\rightarrow$ **1024D**): $X_{\text{exp}} = [x, Lx, L^2x, L^3x] \cdot W$
Progressive reduction (1024D $\rightarrow$ 8D): 7-layer network with ReLU, batch norm, dropout
Construct adjacency A from reduced features
Compute D, L_{norm}, eigenvectors; form U
Compute compressed adjacency $B = U^\top AU$
Map B to Ising Hamiltonian H_C
Run QAOA on H_C with depth p to obtain compressed labels ℓ_B
Map back directly: $\hat{y} = \mathrm{sgn}_\tau(U\ell_B)$ ▷ Unified classification
Graph-cut informed prediction (optional): $\hat{y}_i = \mathrm{softmax}(W_{\text{global}} h_i + \sum_{c \in C(i)} W_c h_i)$
if training **then**
    Compute loss and update feature weights; repeat until convergence
end if

C Investor Profile Attributes

ID	Attribute
1	Family Status
2	Number of Children
3	Number of Grandchildren
4	Occupation Sector
5	Seniority Level
6	Residency Status
7	Ethnicity
8	Illiquid Assets
9	Liquid Assets
10	Liabilities
11	Monthly Expenses
12	Monthly Income
13	Variable Income
14	Emergency Funds
15	Existing Coverage
16	Group Coverage
17	Additional Protection Required
18	Readiness to Pay
19	Access Cash Value (intent)
20	Growth Asset Focus
21	Investment Experience

The 21 raw attributes above are one-hot encoded to $m_0 = 54$ pre-selection features. After feature selection, the effective number of independent features for sample-complexity purposes is $m = 21$.

D Reproducibility Notes

Key parameter settings are summarized in Table 20.

Parameter	Value / Range
Number of investors n	150 (case study)
Raw attributes	21
Pre-selection features m_0	54
Pre-projection dimension $4m_0$	216
Effective independent features m	21
Selected features k_{feat}	top 8 (ensemble)
Compression dimension k	5–40 (experiments)
QAOA depth p	1–3
Shots per circuit	1000–10000
Optimizers	COBYLA, Nelder-Mead
Simulators	cuQuantum, Cirq-GPU, Cirq IonQ model

Table 20 Key experimental parameters.

E Theoretical Limitations and Proposed Solutions

E.1 Problem Identification: The Pitfalls of Over-Compression

Our experimental findings suggest a fundamental limitation in the current approach: compressing 54-dimensional features down to 10 dimensions via graph-based methods before computing similarity matrices results in substantial information loss. Mathematically, this aggressive compression can be formalized as:

Let $X \in \mathbb{R}^{n \times m_0}$ represent our pre-selection feature matrix, with $m_0 = 54$. The compression operation $C : \mathbb{R}^{m_0} \to \mathbb{R}^{10}$ induces an inevitable information loss quantified by:

$$\mathcal{L}_{\text{info}} = I(X;Y) - I(C(X);Y),$$

where $I(\cdot;\cdot)$ denotes mutual information and Y represents the C-class labels. For the investor case study, $C = 4$; for the Cora, Citeseer, and PubMed benchmarks in Section 5, $C \in \{7, 6, 3\}$ respectively. When $\mathcal{L}_{\text{info}}$ exceeds a critical threshold τ, classification performance degrades significantly due to the destruction of discriminative graph-topological patterns.

E.2 Theoretical Advantages and Empirical Validation

E.2.1 Theoretical Guarantees

The proposed method offers two key theoretical advantages over direct compression:

1. **Manifold Preservation**: The initial expansion via the graph Laplacian ensures the learned features reside on the same low-dimensional manifold as the original data, satisfying the condition:

$$\|\nabla_{\mathcal{M}} X_{\text{exp}}\|_F^2 \leq \beta \|\nabla_{\mathcal{M}} X\|_F^2,$$

   where $\nabla_{\mathcal{M}}$ denotes the manifold gradient. This bound guarantees that smoothness properties of the original graph are not destroyed.
2. **Information-Theoretic Optimality**: The progressive reduction stages are designed to achieve the optimal trade-off between compression and predictive power, as defined by the Information Bottleneck Principle (Definition 2) [30]:

$$\min_{h^{(1)},\ldots,h^{(L)}} I(X_{\text{exp}}; h^{(1)}) \quad \text{s.t.} \quad I(h^{(L)}; Y) \geq I_c,$$

   where I_c is the minimal mutual information required for the classification task.

E.2.2 Empirical Results

We validated our unified approach on three benchmark datasets: Cora, Citeseer, and PubMed. The results demonstrate clear superiority:

Method	**Cora**	**Citeseer**	**PubMed**
Baseline (Aggressive $m_0 \to 10$)	71.2	65.8	76.5
GCN	81.5	70.3	79.0
GAT	83.0	72.5	79.5
Unified QAOA-Spectral (Ours)	**85.7**	**74.1**	**82.3**

Table 21 Node classification accuracy (%) on benchmark datasets. Cora has 7 classes, Citeseer has 6, and PubMed has 3. GCN = Graph Convolutional Network; GAT = Graph Attention Network.

References

1. N. Alon and V. D. Milman. λ_1, isoperimetric inequalities for graphs, and superconcentrators. *J. Combin. Theory Ser. B*, 38(1):73-88, 1985.
2. M. Anthony and P. L. Bartlett. *Neural Network Learning: Theoretical Foundations*. Cambridge University Press, 1999.
3. S. Boucheron, G. Lugosi, and P. Massart. *Concentration Inequalities: A Nonasymptotic Theory of Independence*. Oxford University Press, 2013.
4. Y. Boykov and G. Funka-Lea. Graph cuts and efficient N-D image segmentation. *Int. J. Comput. Vision*, 70(2):109-131, 2006.
5. F. G. S. L. Brandão, M. Broughton, E. Farhi, S. Gutmann, and H. Neven. For Fixed Control Parameters the Quantum Approximate Optimization Algorithm's Objective Function Value Concentrates for Typical Instances. arXiv:1812.04170, 2018.
6. P. L. Chebyshev. Des valeurs moyennes. *Journal de Mathématiques Pures et Appliquées*, 12:177-184, 1867.
7. J. Cheeger. A lower bound for the smallest eigenvalue of the Laplacian. In *Problems in Analysis*, pp. 195-199. Princeton University Press, 1970.
8. T. M. Cover and J. A. Thomas. *Elements of Information Theory*, 2nd ed. Wiley-Interscience, Hoboken, NJ, 2006.
9. G. E. Crooks. Performance of the Quantum Approximate Optimization Algorithm on the Maximum Cut Problem. arXiv:1811.08419, 2018.
10. G. Cybenko. Approximation by superpositions of a sigmoidal function. *Math. Control Signals Systems*, 2(4):303-314, 1989.
11. C. Davis and W. M. Kahan. The rotation of eigenvectors by a perturbation. III. *SIAM J. Numer. Anal.*, 7(1):1-46, 1970.
12. C. Eckart and G. Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211-218, 1936.
13. B. Efron and C. Stein. The jackknife estimate of variance. *The Annals of Statistics*, 9(3):586-596, 1981.
14. D. J. Egger et al. Quantum computing for finance: State-of-the-art and future prospects. *IEEE Trans. Quantum Eng.*, 1:1-24, Art no. 3101224, 2020.
15. E. Farhi, J. Goldstone, S. Gutmann. A Quantum Approximate Optimization Algorithm. arXiv:1411.4028, 2014.
16. K. Hornik. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2):251-257, 1991.
17. S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proc. 32nd Int. Conf. Machine Learning (ICML)*, pp. 448-456, 2015.
18. W. B. Johnson and J. Lindenstrauss. Extensions of Lipschitz mappings into a Hilbert space. *Contemp. Math.*, 26:189-206, 1984.
19. W. B. Lee, A. G. Constantinides. Practical applications of quantum computing in finance: Mathematical foundations and deployment challenges. *Encyclopedia*, 6(5), 95, 2026.
20. C. Liu, W. B. Lee, A. G. Constantinides. Quantum Testing of Recommender Algorithms on GPU-Based Quantum Simulators. *Computers*, 14(4):137, 2025.
21. C. Liu, W. B. Lee, A. G. Constantinides. Spectral Graph Compression in Deploying Recommender Algorithms on Quantum Simulators. *Computers*, 14(8):310, 2025.
22. V. Nair and G. E. Hinton. Rectified linear units improve restricted Boltzmann machines. In *Proc. 27th Int. Conf. Machine Learning (ICML)*, pp. 807-814, 2010.
23. A. Y. Ng, M. I. Jordan, Y. Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems*, pp. 849-856. MIT, 2001.
24. P. Rebentrost, B. Gupt, T. R. Bromley. Quantum computational finance: Monte Carlo pricing of financial derivatives. *Phys. Rev. A*, 98(2):022321, 2018.
25. N. Sauer. On the density of families of sets. *J. Combin. Theory Ser. A*, 13(1):145-147, 1972.
26. C. E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27(3):379-423, 1948.
27. D. I. Shuman et al. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE Signal Process. Mag.*, 30(3):83-98, 2013.

28. N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *J. Machine Learning Research*, 15(1):1929-1958, 2014.
29. G. W. Stewart and J.-G. Sun. *Matrix Perturbation Theory*. Academic Press, 1990.
30. N. Tishby, F. C. Pereira, and W. Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.
31. *The Elements of Euclid*. Book I, Proposition 20, c. 300 BC.
32. L. G. Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134-1142, 1984.
33. V. N. Vapnik and A. Y. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory Probab. Appl.*, 16(2):264-280, 1971.
34. V. N. Vapnik and A. Y. Chervonenkis. *Theory of Pattern Recognition*. Nauka, Moscow, 1974.
35. V. N. Vapnik. *Statistical Learning Theory*. Wiley, New York, 1998.
36. C. Williams and M. Seeger. Using the Nyström method to speed up kernel machines. In *Advances in Neural Information Processing Systems*, pp. 682-688. MIT, 2001.
37. S. Woerner and D. J. Egger. Quantum risk analysis. *npj Quantum Information*, 5:15, 2019.
38. L. Zhou, S.-T. Wang, S. Choi, H. Pichler, M. D. Lukin. Quantum Approximate Optimization Algorithm: Performance, Mechanism, and Implementation on Near-Term Devices. *Phys. Rev. X*, 10(2):021067, 2020.