Portfolio Asset Construction Using Variational Quantum Algorithms to Find Maximum Independent Sets

William Huang
Yale University
New Haven, CT 06520
william.huang@yale.edu

Abstract

The rise of quantum computing has revolutionized several fields due to the potential for quantum computers to efficiently solve problems in areas like quantum simulations, financial engineering, and drug discovery. In this paper, I apply a hybrid Variational Quantum Algorithm (VQA) to the classic problem of portfolio diversification. It has been demonstrated that the problem of portfolio construction can be reduced to finding the maximum independent set in market graphs. In the field of combinatorial optimization, the iterative VQA maximum independent set solver can improve optimization performance while mimicking the classical greedy algorithm. To integrate the two ideas, I implement the VQA maximum independent set algorithm to build a portfolio from the Dow Jones Industrial Average. My results show that, at shallow circuit depths, the VQA achieves portfolio selections that are comparable to the maximum independent set formed by the classical linear programming solution.

1 Introduction

Today, classical machine learning (ML) advances have driven breakthroughs across healthcare, finance, and education. However, ML models continue to face challenges in efficiency and scalability. As ML models grow in complexity, they require increasing computational resources to handle high-dimensional data. As such, researchers continue to look towards new directions to perform computations more efficiently.

Quantum computing (QC) is an exciting emerging field that holds the promise of being exponentially faster than classical computing when solving various problems. For example, Grover's Search Algorithm introduced a quantum approach to searching unsorted databases, providing a quadratic speed-up over classical algorithms [Grover, 1996]. Shor's Algorithm calculates prime factors of large numbers in polynomial time, challenging the long-standing RSA encryption scheme [Shor, 1997]. These algorithms demonstrate the potential to exploit quantum properties such as superposition, entanglement, and measurement to solve problems more efficiently. As a result, there has been an increase in research on how these quantum characteristics can be leveraged to improve classical ML.

Quantum machine learning (QML) is a branch of machine learning that combines quantum mechanics and machine learning to process data [Zeguendry et al., 2023]. Classical machine learning is heavily dependent on performing operations on high-dimensional vectors. Quantum computers have demonstrated the ability to perform such operations using tensor product spaces in polynomial time, which is exponentially faster than any classical approach [Najafi et al., 2022]. For example, the Harrow–Hassidim–Lloyd (HHL) algorithm, a numerical quantum approach to solving a system of linear equations [Harrow et al., 2009], laid the foundation for the emergence of several QML algorithms such as quantum support vector machines [Zhang et al., 2022]. Today, many more QML algorithms have been developed. For example, a quantum k-nearest neighbors classification algorithm was formulated in 2014 that significantly reduced the query complexity of classical counterparts by achieving a speedup in distance calculations between vectors using a quantum computer [Wiebe et al., 2014]. Then, in 2018, the quantum k-nearest neighbors algorithm was extended for image classification by encoding image feature vectors in a superposition state [Dang et al., 2018].

Given the impressive results demonstrated by recent QML algorithms, it is not surprising to see the integration of QML within the data-driven field of quantitative finance. Several facets of quantitative finance, such as derivative pricing, risk management, market prediction, and algorithmic trading, have already been explored using ML [Doosti et al., 2024]. Naturally, many of these areas in quantitative finance can be approached from a quantum computing perspective. For example, QML algorithms such as quantum-enhanced random forest algorithms have been explored to improve the prediction of market churn, resulting in an improvement 6% over baseline models [Thakkar et al., 2024]. Quantum support vector machines (QSVMs) have been shown to be more accurate and faster on select datasets for credit card fraud detection [Mironowicz et al., 2024].

In this paper, I explore the application of QML for constructing an investment portfolio. Specifically, I extend upon the ideas of Hidaka et al. [2023] by formulating the portfolio construction problem as a weighted maximum independent set (MIS) problem on market graphs. Then I extend upon the previous research by merging the idea of Brady and Hadfield [2023] to solve the MIS problem. I demonstrate that the proposed QML-guided approach to building the portfolio on market graphs is competitive with the classical solution on the same problem.

2 Graph Algorithms for Portfolio Asset Construction

In this section, I demonstrate how graph theory can be used to solve the problem of portfolio construction. Specifically, I explain how the problem of constructing a portfolio can be reduced to finding the maximum independent set in a market graph.

2.1 Maximum Independent Sets (MIS)

An independent set in a graph is a set of vertices in a graph, G, such that no two vertices in the set are adjacent. Formally, if $S = \{v_1, v_2, ..., v_n\}$ is an independent set, there does not exist any $v_i, v_j \in V(G)$ such that $\{v_i, v_j\} \in E(G)$. The goal of the maximum independent set problem (MISP) is to find the largest independent set in a graph, G. In other words, the aim is to find the largest set of vertices in a graph such that no two vertices in the set have an edge in the original graph.

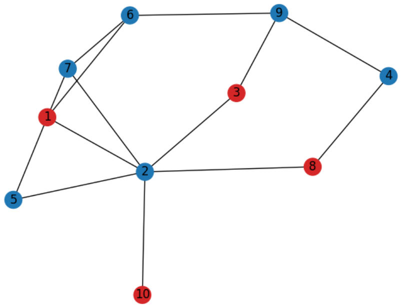

Figure 1: A independent set of vertices (colored red) in a graph. Image Credit: NetworkX.

2.2 Portfolio Construction with MIS

Hidaka et al. [2023] proposed a portfolio construction strategy that reduces the problem of constructing a portfolio to solving the MISP. Specifically, a set of stocks is first selected from a stock universe (a set of tradable stocks). Then, a market graph is constructed from the set of selected stocks - an undirected graph where the vertices in the graph represent the selected stocks and the edges represent the correlations between each pair of stocks. Because of the market graph's construction, clusters of highly interdependent stocks are easily exposed. Therefore, the MIS in the market graph will contain the set of stocks such that there are no edges between any two stocks. Therefore, no pair of stocks in the MIS is highly correlated (moving in the same direction on the market). As such, by spreading assets across uncorrelated stocks, the MIS mitigates idiosyncratic risk and constructs a diversified portfolio.

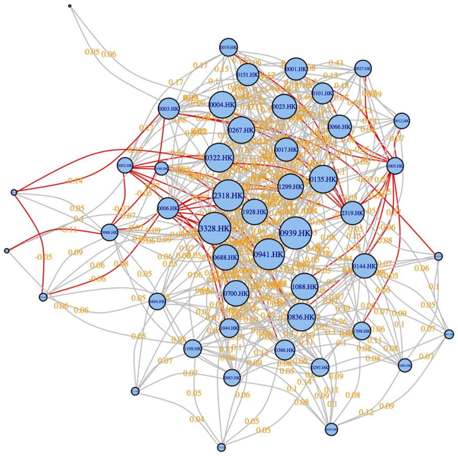

Figure 2: A market graph representing the Hong Kong stock market. Image Credit: Medium.

3 Variational Quantum Algorithms

Although the first cloud quantum computers became available in 2016, the variation in the accuracy of the computers in addition to the limitations in the number of qubits on each computer is a bottleneck that prevents the implementation of many quantum algorithms [Cerezo et al., 2021]. These computers, noisy intermediate-scale quantum (NISQ) computers, are the current state-of-the-art quantum devices.

Variational quantum algorithms (VQAs) are currently the leading strategy to leverage quantum advantages in NISQ devices [Cerezo et al., 2021]. VQAs are contained within a subclass of algorithms that leverage both classical and quantum resources to solve approximation problems. Due to the limitations of NISQ computers, accounting for the constraints requires a learning-based approach. Therefore, VQAs are often regarded as analogues of machine learning methods such as neural networks on classical computers. However, VQAs also take advantage of classical optimization, as most VQA algorithms involve running parameterized quantum circuits on a quantum computer, followed by performing parameter optimization on a classical computer [Cerezo et al., 2021]. Therefore, they are often referred to as hybrid classical-quantum algorithms and hybrid quantum-classical algorithms.

3.1 The General VQA Framework

As previously stated, VQAs function in a similar manner to traditional neural networks. VQAs follow the idea of solving a problem with "trial and error". In other words, VQAs involve an interactive process in which a classical optimizer adjusts the parameters on a quantum circuit to minimize a loss function. The process repeats until the output converges to the best possible solution. This hybrid quantum–classical feedback process ensures that even today's NISQ hardware can handle complex tasks without need for a fully error-corrected quantum computer. Now I will formally explain the idea of VQAs.

Developing a VQA requires first defining a cost function, C. The cost function is a mapping from a set of trainable parameters θ to a real number that encodes the solution to the problem. In practice, VQAs define a quantum circuit that represents an approximation of the true cost function. For the purpose of VQA, the cost function has an additional constraint that the cost must not be efficiently computable using a classical computer, as a VQA would then offer no advantage in speed. In addition, due to limitations in NISQ devices, the circuit depth and the number of ancilla bits of the quantum circuit representation of C must be shallow [Cerezo et al., 2021].

Let θ be the set of parameters that can be optimized. Then, the aim is to find the following set of parameters: θ^*.

$$\theta^* = \text{arg}min_\theta C(\theta) \tag{1}$$

Then, an ansatz is created. An ansatz is a parameterized quantum circuit that provides a range of quantum states that are accessible by adjusting the parameters of the quantum circuit (QuEra Computing). For a given cost function, θ is encoded in a unitary $U(\theta)$ that is applied to the input of the quantum circuit. $U(\theta)$ can be expressed as the product of L sequentially applied unitaries [Cerezo et al., 2021].

$$U(\theta) = U_1(\theta_1)U_2(\theta_2)...U_L(\theta_L) \tag{2}$$

Then, the ansatz is trained using quantum and classical operations to solve the optimization problem [Cerezo et al., 2021].

3.2 Quantum Combinatorial Optimization

Quantum combinatorial optimization describes the application of VQA algorithms to solving discrete NP-hard problems. Specifically, quantum combinatorial optimization applies the VQA framework by mapping the discrete problem onto parameterized quantum circuits whose lowest-energy states encode optimal or near-optimal solutions [Chicano et al., 2025].

A Hamiltonian is a mathematical operator that encodes the energy of a quantum system by describing the dynamics and evolution of quantum states (QuEra Computing). It is an Hermitian operator that acts on the state space of the system. The eigenvalues of the Hamiltonian correspond to the various energy levels of the system, and the eigenvectors represent the corresponding quantum states. In quantum combinatorial optimization, the combinatorial problem (MIS, graph coloring, MinCut) is encoded as a Hamiltonian where each bit corresponds to the qubit's basis [Chicano et al., 2025]. Specifically, the problem is rewritten as a Quadratic Unconstrained Binary Optimization (QUBO) problem to minimize $f(x)$ which has the following form

$$f(x) = \sum_{i=1}^{n} Q_{i,i}\, x_i \;+ \sum_{1 \le i < j \le n} Q_{i,j}\, x_i\, x_j. \tag{3}$$

where Q is a weight matrix and x represents a binary variable (D-Wave).

Then, a variational ansatz is used where a shallow circuit like the Quantum Approximate Optimization Algorithm (QAOA) is chosen. The QAOA has an alternating structure in which an input state is mapped to a ground state of a Hamiltonian by sequentially applying a cost unitary $e^{-i\gamma_k \hat{H}}$ and a mixer unitary $e^{-i\beta_k \hat{H}_X}$ Cerezo et al. [2021]. Formally, the ansatz takes the form

$$e^{-i\beta_p \hat{H}_X}\, e^{-i\gamma_p \hat{H}}\, \cdots\, e^{-i\beta_1 \hat{H}_X}\, e^{-i\gamma_1 \hat{H}} \tag{4}$$

where γ_k governs how much to value the cost function at the current layer and β_k dictates the exploration of possible solutions.

Then, similar to the general VQA framework, a classical optimizer is then leveraged to update the parameters γ_k and β_k to minimize the expectation value of the encoded Hamiltonian as a QUBO. This minimizes the expectation of $\langle \psi(\gamma, \beta) \mid \hat{H} \mid \psi(\gamma, \beta) \rangle$. Once the expectation value converges, the state is measured to obtain the state in the computational basis. This measurement gives bitstrings that correspond to solutions with a higher probability of being good approximations.

In the following section, I will explain the application of Quantum Combinatorial Optimization towards solving the MISP.

4 MIS Algorithms

In this section, I review the classical approaches to solving the MISP. I also explain the [Brady and Hadfield, 2023] formulation of an iterative quantum algorithm for MIS based on the quantum approximate optimization algorithm (QAOA).

4.1 Classical MIS Algorithm with Linear Programming

Although the MISP is contained in the set of NP-hard algorithms, there has been ongoing research to develop faster algorithms over brute-force search to solve MISP or find approximations to the MIS.

An integer linear programming method transforms the MISP into a binary optimization problem that can be handled by mixed integer linear programming solvers [Aspnes, 2004]. Specifically, a binary variable, x_i is introduced for each vertex $v_i \in V(G)$. Because the goal is to maximize the size of the independent set, $\sum_i x_i$ is maximized. However, to enforce the independent set property, a constraint is added where for each edge $(v_i, v_j) \in E(G)$, $x_i + x_j \leq 1$. Furthermore, $x_i \geq 0$ and $x_i \leq 1$. These constraints enforce that only one of the vertices can be included in our independent set.

4.2 Quantum MIS Algorithm with VQA

In [Brady and Hadfield, 2023] the MIS is first encoded into a Hamiltonian. To transform the problem into an operator, each vertex v_i is first transformed into a binary variable x_i where $1 \leq i \leq n = |V|$. Then the size of the independent set for a bitstring x is $\sum_i^n x_i$. Then, the QUBO cost is defined as

$$H(x) = -\sum_{i=1}^{n} x_i \;+\; W \sum_{(i,j)\in E} x_i\, x_j, \tag{5}$$

where the first term rewards maximizing the size of the independent set, while the second term ensures that no adjacent vertices are picked through a penalty W. Then, each bitstring is transformed into a projector as follows. For a given x_i, the corresponding projector is

$$x_i = \frac{1 - Z_i}{2} \tag{6}$$

Therefore, the product $x_i x_j$ becomes

$$x_i x_j = \frac{1 - Z_i Z_j - Z_i - Z_j}{4} \tag{7}$$

Reformatting (5),

$$\hat{H} = \frac{1}{2}\sum_{i=1}^{n} Z_i \;+\; \frac{W}{4} \sum_{(i,j)\in E} Z_i\, Z_j, \tag{8}$$

Then, [Brady and Hadfield, 2023] utilize the standard QAOA ansatz which starts from the uniform superposition state $\mid \psi_0\rangle = \frac{1}{\sqrt{2^n}} \sum_{z\in\{0,1\}^n} \mid z\rangle$ and transforms the state using a standard cost and mixer Hamiltonian. [Brady and Hadfield, 2023] demonstrate that at a circuit depth of 1, the following QAOA state $\mid \psi\rangle$ perfectly reproduces the MIS produced by the classical greedy algorithm for MIS.

$$\mid \psi\rangle = e^{-i\frac{\pi}{2}\hat{H}_X}\, e^{-i\frac{\pi}{4}\hat{H}} \mid \psi_0\rangle \tag{9}$$

5 Methodology

5.1 Market Graph Construction

Due to limitations in the size of graphs that quantum MIS algorithms can handle on a classical computer, I decided to model the Dow Jones Industrial Average which only has 30 tickers. To create the market graph, we follow the methods from [Hidaka et al., 2023]. Specifically, I first construct a correlation matrix, C, where $C_{i,j}$ represents the correlation between stocks i and j for the last T days. Then, $C_{i,j}$ is

$$C_{i,j} = \frac{\sum_t^T (R_i(t) - \bar{R}_i)(R_j(t) - \bar{R}_j)}{\sqrt{\sum_t^T (R_i(t) - \bar{R}_i)^2 (R_j(t) - \bar{R}_j)^2}} \tag{10}$$

where $R_i(t)$ is the logarithmic daily return $R_i(t) = ln(\frac{P_i(t)}{P_i(t1)})$ and $\bar{R}_i$ is the average $R_i(t)$ over the last T days.

I calculate the correlation matrix C using price data from the Dow Jones Industrial Average over the number of business days over the last three years. Then, the connectivity of the market graph is determined by C and a threshold parameter of θ. Two stocks i and j are connected in the market graph iff $C_{i,j} \geq \theta$. For this paper, we select $\theta = 0.6$ to ensure that stocks that are weakly correlated can still be added to the independent set.

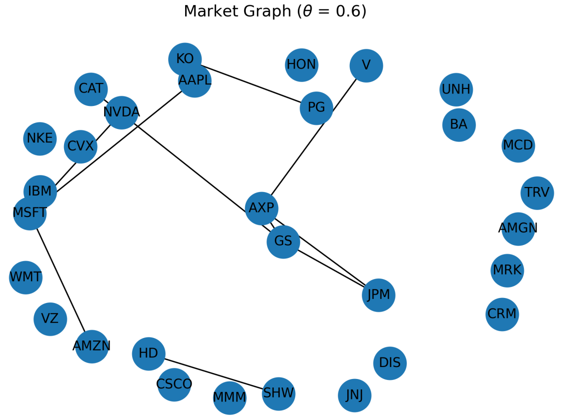

Figure 3: A market graph of the Dow Jones Industrial Average with $T = 756$ and $\theta = 0.6$.

5.2 QUBO Formulation and Setup

First, to encode the MIS as a Hamiltonian, I follow the MST Hamiltonian from (8). Then, using a series of Hadamards, we construct an initial state in a superposition state.

$$\mid \psi_0 \rangle = H^{\otimes n} \mid 0 \rangle = \mid + \rangle^{\otimes n} \tag{11}$$

Therefore, the $\mid \psi_0 \rangle$ considers all possible candidate answers at once. Subsequent cost and mixer unitaries can then guide the amplitudes toward good solutions and away from bad ones through constructive and destructive interference.

5.3 Building the QAOA Ansatz

Following [Brady and Hadfield, 2023], I build the ansatz by alternating two unitary matrices: the cost and mixer unitaries. Because a cost unitary follows the form $e^{i\gamma_k \hat{H}}$ (for layer k) where $\hat{H}$ follows (8) (a sum of commuting Pauli matrices), the sum can be broken down into a series of single qubit and two qubit rotations. In other words, for a given layer, I apply the following Z rotation on qubit i

$$e^{-\frac{1}{2} i\gamma_k Z_i} \tag{12}$$

and the two qubit ZZ rotation on the qubits i and j.

$$e^{-\frac{W}{4} i\gamma_k Z_i Z_j} \tag{13}$$

The mixer unitary follows the general form ($e^{-i\beta_k \hat{H}_X}$) for layer k as well and is applied immediately after the cost unitary.

$$\exp\Big(-i\,\beta \sum_{i=1}^{n} X_i\Big) = \prod_{i=1}^{n} \exp\big(-i\,\beta\,X_i\big) \tag{14}$$

where each single-qubit rotation is

$$\exp\big(-i\,\beta\,X\big) = R_X(2\beta) \tag{15}$$

Then, with an ansatz of p layers, we apply the set of cost-mixer unitaries p times to the starting state (11). This gives the final state

$$|\,\psi\rangle_{final} = \left(\prod_{k=1}^{p} U_{mixer}(\beta_k)\,U_{cost}(\gamma_k)\right)\,|\,+\rangle^{\otimes n}\,, \tag{16}$$

5.4 Parameter Optimization

To minimize the expectation of the Hamiltonian, I leverage Pennylane's GradientDescentOptimizer with a learning rate $\eta = 0.2$. After the expectation converges, I use the standard majority-voting approach for VQAs to transform the probabilistic quantum state into a deterministic binary bitstring representing the stocks to maintain in our MIS. Specifically, I use Pennylane's QNode sampler to generate a probability distribution of all n qubits. I then transform measured values from -1/+1 to 1/0. For each qubit, I measure the fraction of times the qubit returned 1. Then, if more the qubit was 1 over half of the time, then the qubit is represented as a 1 in the final bitstring. Otherwise, the qubit is represented as a 0.

6 Experiments

In this section, I perform two sets of experiments. First, I investigate the impact of circuit depth on the performance of the quantum MIS algorithm. Then, I compare the MIS generated by the quantum MIS algorithm with the solution from the classical linear programming algorithm.

Note: For the experiments involving the quantum algorithm, I manually added all the stocks with a degree 0 (nodes in the market graph without an edge to any other node) to the final independent set due to limitations in running the VQA algorithm on a classical computer.

6.1 Circuit Depth Optimization

In this experiment, I investigate the effect of varying shallow circuits on the performance of the algorithm. To gauge performance, I evaluate the final expectation value of the Hamiltonian and the size of the independent set generated by the algorithm on circuits with a depth of 1 to 5.

6.2 Classical vs Quantum

After testing which of the circuit depths for the quantum algorithm yields improved performance, we compare the results of the independent sets generated by the classical linear programming solution with the independent sets generated by the quantum algorithm. Specifically, I utilize the Jaccard similarity score, which is the fraction of the size of the intersection of two sets over the size of the union of the two sets.

7 Results

7.1 Circuit Depth Optimization

First, I note that the following tickers were automatically added to the MIS as they were isolated vertices: [AMGN, BA, CRM, CSCO, CVX, DIS, HON, IBM, JNJ, MCD, MMM, MRK, NKE, TRV, UNH, VZ, WMT]. The tickers added to each independent set can be seen in Table 2.

In Table 1, it is clear that the selected MIS and Hamiltonian cost vary greatly as the circuit depth changes. As shown in Table 1, a circuit depth of 3 maximizes the size of the independent set, while a circuit depth of 4 minimizes the Hamiltonian cost. To determine the circuit depth that most closely matches the true MIS, we compare the independent sets selected by each circuit with the MIS selected by the classical linear programming solution.

Table 1: QAOA MIS Results on Dow Jones 30 for Depths $p = 1–5$

Depth p	Final $\langle H \rangle$	MIS size
1	-0.0000	24
2	3.3319	18
3	-0.0284	25
4	-2.6144	23
5	-0.6890	19

Table 2: Selected Tickers by Depth

Depth p	Selected tickers
1	AMGN, AMZN, AXP, BA, CRM, CSCO, CVX, DIS, HON, IBM, JNJ, JPM, MCD, MMM, MRK, MSFT, NKE, NVDA, PG, TRV, UNH, V, VZ, WMT
2	AMGN, BA, CRM, CSCO, CVX, DIS, HON, IBM, JNJ, JPM, MCD, MMM, MRK, NKE, TRV, UNH, VZ, WMT
3	AAPL, AMGN, AMZN, AXP, BA, CAT, CRM, CSCO, CVX, DIS, GS, HD, HON, IBM, JNJ, MCD, MMM, MRK, NKE, NVDA, TRV, UNH, V, VZ, WMT
4	AMGN, AMZN, BA, CAT, CRM, CSCO, CVX, DIS, HD, HON, IBM, JNJ, KO, MCD, MMM, MRK, NKE, NVDA, TRV, UNH, V, VZ, WMT
5	AMGN, AXP, BA, CRM, CSCO, CVX, DIS, GS, HON, IBM, JNJ, MCD, MMM, MRK, NKE, TRV, UNH, VZ, WMT

7.2 Classical vs Quantum

To determine the optimal circuit depth, I evaluate the Jaccard score, a metric to gauge similarity between two sets. For each of the MIS produced by the five circuit depths, S_q, we compare it to the set produced by the classical algorithm S_c by calculating $score = \frac{|S_q \cap S_c|}{|S_q \cup S_c|}$. As seen in Table 3, the circuit depth of three achieves the highest similarity score of 0.89, which is significantly higher than the score for other circuit depths. The circuit with the lowest Hamiltonian cost $p = 4$, achieves the second highest similarity score.

Table 3: QAOA MIS Results vs. Classical LP Results for Depths $p = 1–5$

Depth p	QAOA size	Jaccard score
1	24	0.74
2	18	0.72
3	25	0.89
4	23	0.74
5	19	0.68

8 Conclusions

In this report, I performed a literature review of VQAs and then investigated the application of a VQA algorithm towards solving the maximum independent set problem for the Dow Jones market graph. Then, I evaluated the performance of the generated algorithm by comparing the Hamiltonian cost, MIS size, and MIS similarity to the classical solution for circuits of depths $p = 1 - 5$. I demonstrated that the VQA algorithm achieves the highest similarity to the classical MIS at a circuit depth of $p = 3$.

Future work could be dedicated to performing a backtest to find the potential profit that could be made by investing in the stocks in the MIS. Further, because I was limited in system resources, I would be interested in performing future tests on the algorithm with larger stock indices (like the S&P 500) and without removing degree 0 vertices in the market graph.

References

James Aspnes. Notes on Linear Programming. Technical report, Yale University, Department of Computer Science, April 2004. URL https://www.cs.yale.edu/homes/aspnes/pinewiki/attachments/LinearProgramming/lp.pdf.

Lucas T. Brady and Stuart Hadfield. Iterative quantum algorithms for maximum independent set: A tale of low-depth quantum algorithms, 2023. URL https://arxiv.org/abs/2309.13110.

M. Cerezo, Andrew Arrasmith, Ryan Babbush, Simon C. Benjamin, Suguru Endo, Keisuke Fujii, Jarrod R. McClean, Kosuke Mitarai, Xiao Yuan, Lukasz Cincio, and Patrick J. Coles. Variational quantum algorithms. *Nature Reviews Physics*, 3(9):625–644, August 2021. ISSN 2522-5820. doi: 10.1038/s42254-021-00348-9. URL http://dx.doi.org/10.1038/s42254-021-00348-9.

Francisco Chicano, Gabiel Luque, Zakaria Abdelmoiz Dahi, and Rodrigo Gil-Merino. Combinatorial optimization with quantum computers. *Engineering Optimization*, 57(1):208–233, January 2025. ISSN 1029-0273. doi: 10.1080/0305215x.2024.2435538. URL http://dx.doi.org/10.1080/0305215X.2024.2435538.

Yijie Dang, Nan Jiang, Hao Hu, Zhuoxiao Ji, and Wenyin Zhang. Image classification based on quantum knn algorithm, 2018. URL https://arxiv.org/abs/1805.06260.

Mina Doosti, Petros Wallden, Conor Brian Hamill, Robert Hankache, Oliver Thomson Brown, and Chris Heunen. A brief review of quantum machine learning for financial services, 2024. URL https://arxiv.org/abs/2407.12618.

Lov K. Grover. A fast quantum mechanical algorithm for database search, 1996. URL https://arxiv.org/abs/quant-ph/9605043.

Aram W. Harrow, Avinatan Hassidim, and Seth Lloyd. Quantum algorithm for linear systems of equations. *Physical Review Letters*, 103(15), October 2009. ISSN 1079-7114. doi: 10.1103/physrevlett.103.150502. URL http://dx.doi.org/10.1103/PhysRevLett.103.150502.

Ryo Hidaka, Yohei Hamakawa, Jun Nakayama, and Kosuke Tatsumura. Correlation-diversified portfolio construction by finding maximum independent set in large-scale market graph. *IEEE Access*, 11:142979–142991, 2023. ISSN 2169-3536. doi: 10.1109/access.2023.3341422. URL http://dx.doi.org/10.1109/ACCESS.2023.3341422.

Piotr Mironowicz, Akshata Shenoy H., Antonio Mandarino, A. Ege Yilmaz, and Thomas Ankenbrand. Applications of quantum machine learning for quantitative finance, 2024. URL https://arxiv.org/abs/2405.10119.

Khadijeh Najafi, Susanne F. Yelin, and Xun Gao. The Development of Quantum Machine Learning. *Harvard Data Science Review*, 4(1), jan 27 2022. https://hdsr.mitpress.mit.edu/pub/cgmjzm3c.

Peter W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5):1484–1509, October 1997. ISSN 1095-7111. doi: 10.1137/s0097539795293172. URL http://dx.doi.org/10.1137/S0097539795293172.

Sohum Thakkar, Skander Kazdaghli, Natansh Mathur, Iordanis Kerenidis, André J. Ferreira-Martins, and Samurai Brito. Improved financial forecasting via quantum machine learning, 2024. URL https://arxiv.org/abs/2306.12965.

Nathan Wiebe, Ashish Kapoor, and Krysta Svore. Quantum algorithms for nearest-neighbor methods for supervised and unsupervised learning, 2014. URL https://arxiv.org/abs/1401.2142.

Amine Zeguendry, Zahi Jarir, and Mohamed Quafafou. Quantum machine learning: A review and case studies. *Entropy*, 25(2), 2023. ISSN 1099-4300. doi: 10.3390/e25020287. URL https://www.mdpi.com/1099-4300/25/2/287.

Rui Zhang, Jian Wang, Nan Jiang, Hong Li, and Zichen Wang. Quantum support vector machine based on regularized Newton method. *Neural Networks*, 151:376–384, 2022. doi: 10.1016/j.neunet.2022.03.043.