A QUANTUM APPROXIMATION ALGORITHM FOR COVARIANCE OF RANDOM VARIABLES

KOREN, M.[1*] – PERETZ, O.[2]

[1] *School of Industrial Engineering and Management, Shenkar College of Engineering, Design, and Art, Ramat-Gan, Israel.*

**Corresponding author*
e-mail: michal.koren[at]shenkar.ac.il

(Received 21st April 2024; revised 07th July 2024; accepted 16th July 2024)

Abstract. Quantum computing is a promising emerging field in computational science, attracting significant international research interest. Data representation is crucial to the success of machine learning models, and covariance analysis to determine relationships between pairs of random variables is an essential step in exploratory data analysis. Here we introduce a novel quantum algorithm for covariance approximation and its circuit implementation for discrete-value vectors. The algorithm leverages vital components of quantum computing, including amplitude encoding, the Grover diffusion algorithm for amplitude amplification and the quantum Fourier transform. As a core innovation, we utilize two quantum states and their properties of superposition product states to estimate the covariance between input vectors. Evaluating the quantum algorithm's performance over five discrete-value distributions, we found high levels of agreement between the classical and quantum computations, with maximum average errors averaging 0.153. Furthermore, the analysis revealed additional applications of the Grover diffusion operator to solve problems.
Keywords: *amplitude amplification, quantum fourier transform, grover diffusion operator, data analysis, machine learning*

Introduction

Quantum computing (QC) is a promising, increasingly important area of computational science (Ying, 2010). QC is based on the physics principle of quantum duality, whereby an electron can behave simultaneously as a wave and as a particle (Robertson 1943). Quantum computers significantly reduce computing complexity and can therefore perform tasks much faster than classical computers to quickly process large amounts of data, potentially performing calculations that are impossible with classical computers. Consequently, they have the potential to transform many industries, perhaps most obviously finance, healthcare and artificial intelligence (Piattini et al., 2021). The parallel processing capabilities of quantum computers could enable them to perform a greater variety of operations than classical computers (Wiebe, 2020; Biamonte et al., 2017). Yet because they are sensitive to noise and decoherence (Nachman et al., 2020; Bennett et al., 1997), building and maintaining a superposition of quantum computers for parallel processing poses challenges. Currently, there is no distinction of function between classical and quantum computers, and the same algorithms can be implemented on both (Peretz and Koren, 2024; Buffoni and Caruso, 2021). However, the use of quantum gates allows classical machine learning algorithms to be transformed into QC algorithms (Alchieri et al., 2021; Benedetti et al., 2019). Hence, quantum machine learning (QML) is a young yet rapidly growing field alongside QC. Ultimately, combining classical computing and QC can provide powerful means to solve complex problems.

As QC advances, quantum approximation algorithms (QAAs) have emerged as a novel approach to handling complex computational problems. In contrast to classical methods, QAAs use quantum phenomena such as superposition and entanglement to explore multiple solution paths simultaneously (Self et al., 2021; Farhi et al., 2014). As a result, solution of many optimization problems, such as integer factorization and combinatorial optimization, can be significantly speeded up and enhanced (Zhang et al., 2022; Hadfield et al., 2019). Continued development and refinement of these algorithms are expected to revolutionize cryptography, machine learning and scientific simulations, leading to significant advances in science and technology (Willsch et al., 2020; Choi and Kim, 2019). Data representation is crucial to machine learning models. Classical machine learning relies on representing data numerically to enable them to be processed with optimal efficiency and effectiveness by a classical algorithm. Quantum machine learning poses the same fundamental question: how to efficiently represent input data for analysis by quantum algorithms (Li et al., 2022; Weigold et al., 2021; 2020). Quantum machine learning algorithms therefore are directly affected by their own computational capacity (Koren and Peretz, 2024; Sierra-Sosa et al., 2023; Dilip et al., 2022; Blank et al., 2020; LaRose and Coyle, 2020). Data can be encoded for ingestion by quantum systems by three main methods: (1) In *basic encoding*, the simplest method conceptually, a classical string is associated with a computational basis state. Notably, however, the state vectors quickly come to have mostly zero values (i.e., become sparse); (2) In *amplitude encoding*, data are encoded into the amplitudes of a quantum state. A system of n qubits provides 2^n amplitudes, so amplitude encoding can encode a dataset of N records over M features using $\log_2(N \cdot M) + 1$ qubits; and (3) In *angle encoding*, features are encoded into the rotation angles of qubits. Unlike other encoding methods, angle encoding encodes only one data point at a time, rather than an entire dataset, and it requires at most M qubits to encode M features. All three methods have been found to yield better computational results in quantum autoencoders and image processing (Majji et al., 2023; Shin et al., 2023; Bravo-Prieto, 2021; Romero et al., 2017).

Grover's search algorithm, a cornerstone of QC, is a search algorithm for unstructured data. Whereas a classical computer search algorithm requires n operations (as the input size), the Grover search algorithm involves searching by only $\sqrt{n}$ operations (Mandal et al., 2014; Cui and Fan, 2010). This algorithm is based on the Grover diffusion operator, which flips the amplitudes of all states around their mean (Shi et al., 2017). This process of "inversion around the mean" increases the marked state's amplitude while suppressing the unmarked state's amplitude (Tulsi, 2015). With repeated use of this operator, the algorithm progressively concentrates the amplitude on the target state, resulting in a high probability of its identification (Jang et al., 2021). In QC, the QFT is an analog of the classical discrete Fourier transform (DFT), but it operates on quantum bits (qubits) instead of classical bits (Cleve and Watrous, 2000; Hales and Hallgren, 2000). Leveraging the inherent parallelism of quantum mechanics, QFT allows exponentially faster solutions than classical methods. The QFT transforms a quantum state from the computational basis (in which strings of classical bits represent states) to its frequency or Fourier basis (Vorobyov et al., 2021; Mashhadi, 2019; Zhou et al., 2017; Weinstein et al., 2001). The steps involved in multiplying two quantum states in superposition using the QFT are as follows: apply the QFT to both quantum states separately; perform the multiplication operation in the Fourier basis, which is equivalent to multiplying the states in the transformed domain; and finally, apply the inverse QFT to transform the result back to the original computational basis (Nam et al., 2020; Ruiz-Perez and Garcia-Escartin, 2017; Moore et al., 2006).

Exploratory data analysis (EDA) is a group of methods designed to examine a set of data before beginning to build a model incorporating them (Tukey, 1977). In EDA, data are explored to identify patterns, trends, underlying structures, anomalies and any other features of interest (Chatfield, 1986; Leinhardt and Wasserman, 1979), with the aim of developing valid models based the resulting insights (Koren et al., 2023a; Data et al. 2016; Morgenthaler, 2009). EDA comes in three main flavors: (1) In *univariate analysis*, one dependent variable is explored (in a dataset, each variable is explored separately); this may involve statistical analyses, for example of the mean, median, variance and so on (Vigni et al., 2013; Behrens and Yu, 2003); (2) In *bivariate analysis*, the relationships between pairs of variables are examined; these may be numerical, categorical or mixed variables (Jebb et al., 2017; Cleff, 2014); and (3) In *multivariate analysis*, three or more variables are involved (Wang et al., 2023; Wongsuphasawat et al., 2019; Gelman, 2004). Notably, analyzing the distribution of a dataset significantly influences the results of the analysis. Furthermore, understanding the distribution can help to choose appropriate statistical tests, identify anomalies and assess the normality of the data, among other things, thereby enhancing the reliability, accuracy and validity ot your results (Koren and Peretz, 2024). As part of bivariate analysis, covariance is a way to determine whether two random variables are related; in this study, we focus on discrete random variables. A positive covariance value indicates that the variables change in the same direction, increasing or decreasing, whereas a negative covariance describes variables that move in different directions. Besides identifying relationships between variables, covariance plays a role in statistics and data analysis. It offers an intuitive method for dimensionality reduction (Zhang and Chen, 2019), feature selection (Karami et al., 2023; Xu et al., 2023) and more. When interpreting data and constructing models to extract insights, understanding covariance is essential. Here, we describe a new quantum algorithm that we developed for covariance approximation-quantum covariance approximation (QCA)-and its circuit implementation to discrete-value vectors. The algorithm is based on amplitude encoding (a key component in QC), the Grover diffusion algorithm (for amplitude amplification) and QFT (for multiplication of quantum-state vectors).

Materials and Methods

Quantum approximation algorithm

This section presents and describes the Quantum Covariance Approximation (QCA) algorithm, a novel approximation algorithm for quantum covariance calculation between two discrete values vectors. The algorithm is based on amplitude encoding, the Grover diffusion algorithm (for amplitude amplification), and QFT for multiplication of quantum state vectors. First, we will describe the notations for this study. Second, we present the algorithm and its procedure. Last, we will present the implementation and correctness of the algorithm. *Table 1* presents the notation used in this study.

Table 1. *Notation used in this article.*

Symbol	Notes
n	The input size
$X, Y \in \mathbb{Z}^n$	Discrete-value vectors of size n

M	A diagonal square matrix of size m with diagonal $(\sqrt{1^i}, \sqrt{2^i}, \ldots, \sqrt{m^i})$
$G(\mathbb{D})$	Grover diffusion operator over the distribution $\mathbb{D}$
$\alpha, \beta, \delta \in \mathbb{C}$	Complex coefficients

Quantum gates and logic

Let $X, Y \in \mathbb{Z}^n$ be two discrete-value vectors consisting of n elements. The goal of the algorithm is to approximate the covariance between the variables, which is defined as:

$$Cov(X, Y) = E[(X - E[X])(Y - E[Y])] \quad \text{Eq. (1)}$$

where $E[X], E[Y]$ are the expected values of X and Y, respectively. The approximation algorithm consists of three sub-procedures, as follows (*Figure 1*).

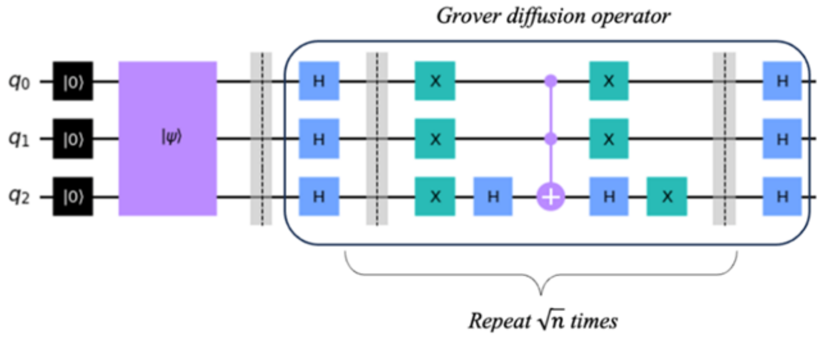

Figure 1. *The diffusion box quantum circuit consists of three qubits, although this can be generalized to higher dimensions with tensor products.*

Classical computer preprocessing (state preparation)

At the beginning, the algorithm method transforms each input vector into an amplitude vector (by classical computer processing) and defines the states, denoted $|\psi_x\rangle, |\psi_y\rangle$, as follows:

$$|\psi_x\rangle = \frac{1}{\sqrt{\sum_{i=1}^{n} x_i^{-1}}} \sum_{i=1}^{n} \sqrt{x_i^{-1}} |i\rangle \quad \text{Eq. (2)}$$

$$|\psi_y\rangle = \frac{1}{\sqrt{\sum_{j=1}^{n} y_j^{-1}}} \sum_{j=1}^{n} \sqrt{y_j^{-1}} |j\rangle \quad \text{Eq. (3)}$$

Diffusion box

Based on the output of the classical computer preprocessing, the algorithm allocates $\lfloor \log_2 n + 1 \rfloor$ qubits and creates two independent quantum circuits, one for each state vector. Once the quantum states are initialized, the algorithm applies the Hadamard gate to move the states into superposition. Thus, the current states can be represented as:

$$H^{\otimes \log_2 n}|\psi_x\rangle \frac{\sum_i (-1)^{\psi_x \cdot i}|i\rangle}{\sqrt{n}} \quad \text{Eq. (4)}$$

$$H^{\otimes \log_2 n}|\psi_y\rangle = \frac{\sum_i (-1)^{\psi_y \cdot j}|j\rangle}{\sqrt{n}} \quad \text{Eq. (5)}$$

At this point, where the states are in superposition, the algorithm applies $\sqrt{n}$ times the Grover diffusion operator and samples the resultant state vectors of X and Y.

QCA

The algorithm uses the diffusion box output and applies QFT to each state. Let $|\widehat{\psi_x}\rangle, |\widehat{\psi_y}\rangle$ be the transformed quantum states, defined as:

$$|\widehat{\psi_x}\rangle = \sum_{i=1}^{n} \delta_i |i\rangle \quad \text{Eq. (6)}$$

$$|\widehat{\psi_y}\rangle = \sum_{j=1}^{n} \delta_j |j\rangle \quad \text{Eq. (7)}$$

Note that $\sum_i \delta_i^2 = \sum_j \delta_j^2 = 1$. The algorithm multiplies the transformed vectors. The output state vector represents the product state, denoted $|\phi\rangle$:

$$|\phi\rangle = \sum_{i=1}^{n} \delta_i |i\rangle * \sum_{j=1}^{n} \delta_j |j\rangle = \sum_{i,j} \delta_i \delta_j |i+j\rangle \quad \text{Eq. (8)}$$

Next, the algorithm calculates the expected value of $|\phi\rangle$ using the parameterized operator M, denoted $\langle\phi|M|\phi\rangle$ (Koren et al., 2023b). Last, it returns the approximated covariance:

$$\frac{\sum_i (-1)^{\phi \cdot i}|i\rangle}{n} \cdot \langle\phi|M|\phi\rangle \quad \text{Eq. (9)}$$

Classical Computer Preprocessing (A)

- $C \leftarrow \emptyset$
- **For** $a_i \in A$
  - $C \leftarrow C \cup \left\{ \left(\sqrt{a_i \cdot \sum_{i=1}^{n} \frac{1}{a_i}} \right)^{-1} \right\}$
- **Return** C

Quantum Covariance (X, Y)

- $|\psi_x\rangle = \sum_{i=1}^{n} \delta_i |i\rangle \leftarrow$ **DiffusionBox(X)**
- $|\psi_y\rangle = \sum_{j=1}^{n} \delta_j |j\rangle \leftarrow$ **DiffusionBox(Y)**
- Apply QFT on both state vectors
- $|\phi\rangle \leftarrow \sum_{i,j} \delta_i \delta_j |i+j\rangle$
- **Return** $\frac{\sum_i (-1)^{\phi \cdot i} |i\rangle}{n} \cdot \langle \phi | M | \phi \rangle$

Diffusion Box (V)

- $v \leftarrow$ **ClassicalComputerPreprocessing(V)**
- Allocate $\lfloor \log_2 n + 1 \rfloor$ qubits
- $|\psi\rangle \leftarrow \left(\sqrt{\sum_{i=1}^{n} \frac{1}{v_i}} \right)^{-1} \sum_{i=1}^{n} \sqrt{v_i^{-1}} |i\rangle$
- $|\psi\rangle \leftarrow H^{\otimes \log_2 n} |\psi\rangle = \frac{\sum_i (-1)^{\psi \cdot i} |i\rangle}{\sqrt{n}}$
- **For** $i = 1$ **to** $\sqrt{n}$
    - Apply $G(|\psi\rangle)$
- **Return** $|\psi\rangle$

Proof of correctness

Theorem 1 (Grover operator)

Let $\mathbb{D}$ be the input distribution and G be the Grover diffusion operator of the quantum state $|\psi\rangle = \sum_i x|i\rangle$, defined as:

$$G = (2|\psi\rangle\langle\psi| - I)\mathbb{D} \qquad \text{Eq. (10)}$$

The diffusion operator aims to inverse the amplitude vector around the mean amplitude (denoted μ). Its operation includes the transformation (Cui and Fan, 2010):

$$\forall x_i \in |\psi\rangle:\ x_i|i\rangle \Rightarrow (2\mu - x_i)|i\rangle \qquad \text{Eq. (11)}$$

Theorem 2 (Multiplication of quantum states)

Let $\hat{\alpha}, \hat{\beta} \in \mathbb{C}$ be complex coefficients and let $|\psi_x\rangle, |\psi_y\rangle$ be two quantum states defined as follows:

$$|\psi_x\rangle = \sum_{i=1}^{n} \hat{\alpha}_i |i\rangle \qquad \text{Eq. (12)}$$

$$|\psi_y\rangle = \sum_{j=1}^{n} \hat{\beta}_j |j\rangle \qquad \text{Eq. (13)}$$

The two quantum states can then be multiplied by the following process (Zhandry, 2012):

1. Construct states separately:

$$\left(\sum_{i=1}^{n} \hat{\alpha}_i |i\rangle\right) \otimes \left(\sum_{j=1}^{n} \hat{\beta}_j |j\rangle\right) = \sum_{i,j} \hat{\alpha}_i \hat{\beta}_j |i,j\rangle \quad \text{Eq. (14)}$$

2. Add vector 'in superposition':

$$\sum_{i,j} \hat{\alpha}_i \hat{\beta}_j |i,j\rangle \Rightarrow \sum_{i,j} \hat{\alpha}_i \hat{\beta}_j |i,j,i+j\rangle \quad \text{Eq. (15)}$$

3. Decode in reverse:

$$\sum_{i,j} \hat{\alpha}_i \hat{\beta}_j |i,j,i+j\rangle \Rightarrow \sum_{i,j} \hat{\alpha}_i \hat{\beta}_j |\, i+j\rangle \quad \text{Eq. (16)}$$

Proposition

The QCA algorithm inputs $X, Y \in \mathbb{Z}^n$ and estimates the covariance between X and Y via quantum procedure.

Proof

Let $X = (x_1, x_2, \dots, x_n)$ and $Y = (y_1, y_2, \dots, y_n)$ be input vectors consisting of discrete values. The method transforms each to amplitude vectors, denoted $|\psi_x\rangle, |\psi_y\rangle$, according to Eqns. (1) and (2). Because encoding the vectors into a quantum state requires division by the vector values, we added 1 to all values to avoid a division-by-zero condition. Also, to support QC constraints, the amplitude vector of X must satisfy:

$$\| |\psi_x\rangle \|^2 = \frac{1}{\sum_{i=1}^{n} \frac{1}{x_i}} \sum_{i=1}^{n} \left(\frac{1}{\sqrt{x_i}}\right)^2 = \frac{1}{\sum_{i=1}^{n} \frac{1}{x_i}} \sum_{i=1}^{n} \frac{1}{x_i} = \frac{\sum_{i=1}^{n} \frac{1}{x_i}}{\sum_{i=1}^{n} \frac{1}{x_i}} = 1 \quad \text{Eq. (17)}$$

The same holds for $|\psi_y\rangle$. A quantum system of n qubits provides 2^n amplitudes, and encoding each vector requires the use of $O(\log_2 n)$ qubits (in cases where the length of the vectors is not to the power of 2, zeros are added, as their values do not change the calculation). Next, the method creates an independent quantum circuit for each state vector and applies the Hadamard gate to move the states into superposition. Thus, the current states can be represented as:

$$H^{\otimes \log_2 n} |\psi_x\rangle = \frac{\sum_i (-1)^{\psi_x \cdot i} |i\rangle}{\sqrt{2^{\log_2 n}}} = \frac{\sum_i (-1)^{\psi_x \cdot i} |i\rangle}{\sqrt{n}} \quad \text{Eq. (18)}$$

$$H^{\otimes \log_2 n}|\psi_y\rangle = \frac{\sum_i (-1)^{\psi_y \cdot j}|j\rangle}{\sqrt{n}} \quad \text{Eq. (19)}$$

At this point, where the states are in superposition, the method applies $\sqrt{n}$ times the Grover diffusion operator and samples the resultant state vectors of X and Y. According to Eqns. (5) and (6), the output vectors represent the standardization of X, Y around their mean amplitude. In analogical to classical computer computation, the current states after diffusion represent $X - E[X]$ and $Y - E[Y]$, respectively. Note that in the Grover search algorithm, the mean amplitude $\mu = \frac{1}{\sqrt{n}}$ holds for uniform distribution of each state. In this study, the mean amplitude is a dynamic value that is dependent on the vector distribution. Next, the algorithm applies QFT to each state. Let $|\widehat{\psi_x}\rangle, |\widehat{\psi_y}\rangle$ be the transformed quantum states, defined by Eqns. (3) and (4), such that $\sum_{i=1}^{n} \delta_i^2 = \sum_{j=1}^{n} \delta_j^2 = 1$. According to Eqn. (7), the product state of $|\widehat{\psi_x}\rangle$ and $|\widehat{\psi_y}\rangle$ can be calculated by separately constructing the product vector.

$$\left(\sum_{i=1}^{n} \delta_i |i\rangle\right) \otimes \left(\sum_{j=1}^{n} \delta_j |j\rangle\right) = \sum_{i,j} \delta_i \delta_j |i,j\rangle \quad \text{Eq. (20)}$$

Then, by Eqns. (8) and (9), applying addition in superposition yields a new quantum state, denoted $|\phi\rangle$:

$$\sum_{i,j} \delta_i \delta_j |i,j\rangle \Rightarrow \sum_{i,j} \delta_i \delta_j |i,j,i+j\rangle \quad \text{Eq. (21)}$$

$$|\phi\rangle = \sum_{i,j} \delta_i \delta_j |i+j\rangle \quad \text{Eq. (22)}$$

Let M be a diagonal square matrix of size m with diagonal $(1,2,\ldots,m)$, where m is the maximum value of X and Y. The algorithm calculates the expected value of $|\phi\rangle$ using the parameterized operator M, denoted $\langle\phi|M|\phi\rangle$. Last, the Hadamard gate is applied again and the method returns:

$$\sum_i (-1)^{\phi \cdot i}|i\rangle \cdot \frac{1}{\sqrt{n}} \cdot \frac{1}{\sqrt{n}} \cdot \langle\phi|M|\phi\rangle = \frac{\sum_i (-1)^{\phi \cdot i}|i\rangle}{n} \cdot \langle\phi|M|\phi\rangle \quad \text{Eq. (23)}$$

Empirical study

Esperimental procedure

To assess and evaluate the results of the proposed method, we compared five discrete-value distributions: (1) Binomial distribution, $Bin(n,p)$, with a success probability of p in a total of n trials; (2) Negative binomial distribution, $NB(n,p)$, with the same parameters as for the binomial distribution; (3) Uniform distribution between $a, b \in \mathbb{N}$, denoted $U(a,b)$; (4) Hypergeometric distribution, $HG(N,D,n)$, with a total of

N items, D 'marked' items and n trials; and (5) Poisson distribution, $Pois(\lambda)$, where λ is the expected value of events in an interval of time. For each distribution, we performed the following experiment: (1) We sampled two random variables of size 10,000 from the distribution; (2) We compared the results of our method with those from classical computation of covariance over the following parameters: a. For the binomial and negative binomial distributions, we compared the success probability between 0.1 to 0.9 with step of 0.1; b. For the hypergeometric distribution, we compared the values $\{2^i | 3 \leq i \leq 9\}$ of the number of 'marked' samples; c. For the Poisson distribution, we compared $\lambda = 0.5$ and $\lambda = 5$. The rationale for examining these parameters relates to the properties of the Poisson distribution: for $0 < \lambda < 1$, it holds that $0 < \frac{\lambda^k}{k!} < \frac{1}{k!}$. Thus, $P(X = k) = e^{-\lambda}\frac{\lambda^k}{k!}$ has the highest value when $k = 0$, and the distribution decreases exponentially as λ increases (i.e., is right skewed). Otherwise, as long as $\lambda \to \infty$, the Poisson distribution approximates the normal distribution with symmetry around λ.

Simplified demonstration

Consider the application of the QCA method over two discrete-value vectors-for simplicity of explanation, vectors that require one and two qubits are used (a broader comparison, including additional distributions, can be found in the next section).

One qubit vectors

Let $X = (2,4)$ and $Y = (2,3)$ be the input vectors. Calculating the covariance using a classical computer yields:

$$Cov(X,Y) = \frac{(2-3)(2-2.5) + (4-3)(3-2.5)}{2-1} = 1 \qquad \text{Eq (24)}$$

Because $X, Y \in \mathbb{R}^2$, one qubit is required to encode each vector in the quantum circuit. The initial states are:

$$|\psi_x\rangle = \frac{1}{\sqrt{\frac{1}{2}+\frac{1}{4}}}\left(\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{4}}|1\rangle\right) = \frac{\sqrt{6}}{3}|0\rangle + \frac{\sqrt{3}}{3}|1\rangle \qquad \text{Eq. (25)}$$

$$|\psi_y\rangle = \frac{1}{\sqrt{\frac{1}{2}+\frac{1}{3}}}\left(\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{3}}|1\rangle\right) = \frac{\sqrt{15}}{5}|0\rangle + \frac{\sqrt{10}}{5}|1\rangle \qquad \text{Eq. (26)}$$

Once both vectors are encoded into quantum states, the algorithm applies the Grover diffusion operator. The output vectors (in superposition) are:

$$|\psi_x\rangle = \frac{\sqrt{3}}{3}|0\rangle + \frac{\sqrt{6}}{3}|1\rangle \qquad \text{Eq. (27)}$$

$$|\psi_y\rangle = \frac{\sqrt{10}}{5}|0\rangle + \frac{\sqrt{15}}{5}|1\rangle \qquad \text{Eq. (28)}$$

Next, the algorithm applies QFT and adds the vectors in superposition. After normalization, we obtain the new state vector:

$$|\phi\rangle = 0.605|0\rangle + 0.796|01\rangle \quad \text{Eq. (29)}$$

The algorithm calculates the expected value of $|\phi\rangle$ using the parameterized gate M:

$$\langle\phi|M|\phi\rangle = \phi^T(M\phi) = \begin{pmatrix}0.605\\0.796\end{pmatrix}^T \left(\begin{bmatrix}1 & 0\\0 & 2\end{bmatrix}\begin{pmatrix}0.605\\0.796\end{pmatrix}\right) = 1.633 \quad \text{Eq. (30)}$$

Last, the sign is determined by $(-1)^0$ and the approximate quantum covariance is:

$$\frac{(-1)^0}{2} \cdot 1.633 = 0.816 \quad \text{Eq. (31)}$$

The recorded difference between the classical and quantum covariance is 0.184.

Two qubit vectors

Let $X = (2,4,6,8)$ and $Y = (2,2,4,2)$ be the input vectors. Calculating the covariance using classical computer yields:

$$Cov(X,Y) = \frac{(2-5)(2-2.5) + \cdots + (8-5)(2-2.5)}{4-1} = 0.667 \quad \text{Eq. (32)}$$

Because $X, Y \in \mathbb{R}^4$, two qubits are required to encode each vector in the quantum circuit. *Figure 2* presents the qubits of X and Y before and after application of the Grover diffusion operator. The new state vector, $|\phi\rangle$, is the result of applying QFT to both states and performing addition in superposition. After normalization, we obtain the new state vector:

$$|\phi\rangle = -0.369|00\rangle - 0.469|01\rangle - 0.593|10\rangle - 0.541|11\rangle \quad \text{Eq. (33)}$$

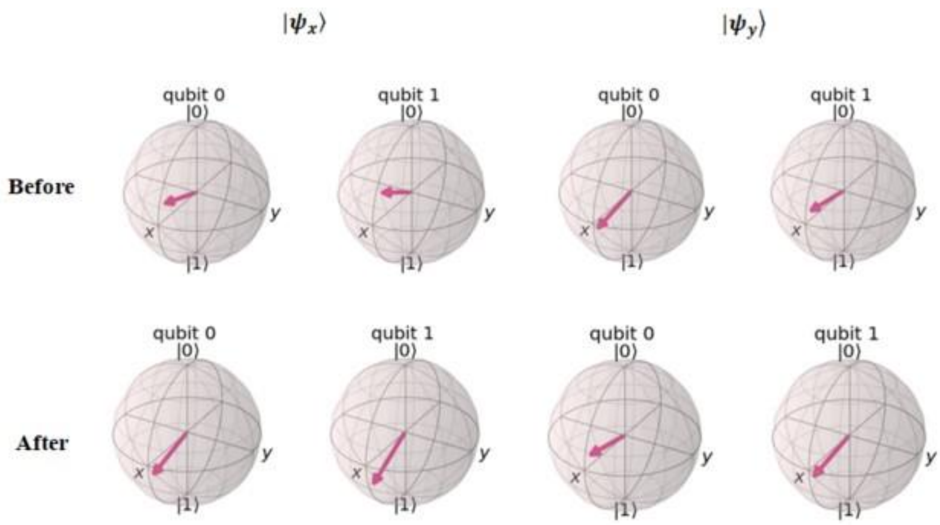

Figure 2. *The qubits of $|\psi_x\rangle$ and $|\psi_y\rangle$ before and after application of the Grover diffusion operator.*

The algorithm calculates the expected value of $|\phi\rangle$ using the parameterized gate M:

$$\langle\phi|M|\phi\rangle = \phi^T(M\phi) = \begin{pmatrix}-0.369\\-0.469\\-0.593\\-0.541\end{pmatrix}^T \left(\begin{bmatrix}1&0&0&0\\0&2&0&0\\0&0&3&0\\0&0&0&4\end{bmatrix}\begin{pmatrix}-0.369\\-0.469\\-0.593\\-0.541\end{pmatrix}\right) = 2.80 \qquad \text{Eq. (34)}$$

The sign is determined by $(-1)^4$ and the approximate quantum covariance is

$$\frac{(-1)^4}{4}\cdot 2.804 = 0.701 \qquad \text{Eq. (35)}$$

The recorded difference between the classical and quantum covariance is 0.034.

Results and Discussion

As an empirical assessment of our QCA QC method, we compared the results obtained using this method to those obtained through classical computation for the five discrete-value distributions defined in the previous section. In addition to comparing the covariance results, we calculated the average error between the classical and quantum calculations by averaging all the absolute differences between the classical and the quantum computation for each distribution. For the Poisson distribution with $\lambda = 0.5$, the classical computation yield a covariance of 0.347, as compared to QCA with 0.253. In the same manner, for $\lambda = 5$, we obtained values 0.102 and 0.069 for classical and quantum computation, respectively. Thus, the average absolute difference for the Poisson distribution is 0.063, indicating a high level of agreement between the methods. *Figure 3* compares the Classical Covariance (CCOV) computation and our algorithm (QCA) over the binomial distribution with different probabilities. A more detailed comparison between the methods, for both the binomial and negative binomial distributions, is shown in *Table 2*. The highest difference (0.415) recorded in the binomial distribution occurred in a probability of 0.3, while the smallest (0.028) occurred in a probability of 0.5. *Table 2* shows that the average absolute difference of the binomial distribution is 0.150, whereas that of the negative binomial is 0.157. Because the error values do not have an upper bound and can have values >1 (depending on the distribution of the data), we conclude that our QCA algorithm succeeds in estimating the classical covariance.

Table 2*. A comparison between classic and quantum covariance over the binomial and negative binomial distributions.*

Probability	Binomial			Negative binomial		
	CCOV	QCA	Error	CCOV	QCA	Error
0.1	0.177	0.349	0.172	–7.494	–7.282	0.212
0.2	–0.418	–0.453	0.035	–2.928	–3.193	0.265
0.3	1.334	0.919	0.415	–0.305	–0.449	0.144
0.4	–0.125	–0.341	0.216	0.317	0.224	0.093
0.5	0.136	0.108	0.028	–0.163	–0.208	0.045
0.6	0.163	0.101	0.062	0.049	0.102	0.053
0.7	0.18	0.405	0.225	0.001	0.075	0.074
0.8	0.071	0.028	0.043	–0.067	–0.035	0.032
0.9	–0.392	–0.236	0.156	0.013	0.014	0.001

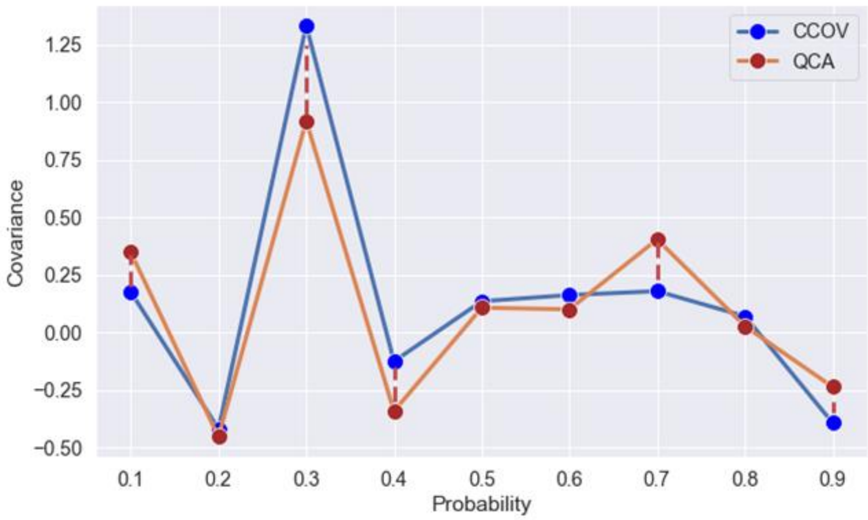

Figure 3. *Comparison between quantum and classic covariance calculation over the binomial distribution. The dashed lines indicate the absolute differences between the computation methods, which determine the average difference.*

The results for the hypergeometric distribution (*Table 3*) are predictable and demonstrate the algorithm's reliability. First, the average absolute difference is 0.050, which is relatively low and close to zero. Second, for a low value of 'marked' items, the covariance yields values around zero, characterized by the hypergeometric distribution function, following the many ways to select these 'marked' items. Thus, there is a low probability that a relationship between these variables will indeed be obtained. On the other hand, for higher values of 'marked' items, the covariance achieved non-zero values, as expected given the properties of the hypergeometric distribution. To examine the QCA outputs and the absolute error distribution over different inputs, we performed polynomial interpolation (Nakamura, 1995); the result is shown in *Figure 4*. Because this comparison is between discrete values (and not distributions), the polynomial interpolation involves constructing a polynomial function that passes through the values. Based on *Figure 4*, we conclude that there is strong and consistent agreement between the classical and quantum computations of covariance. However, several noisy values in the binomial distribution require further analysis.

Table 3*. A comparison of the hypergeometric distribution over different values.*

D	8	16	32	64	128	256	512
CCOV	0.002	–0.007	0.107	–0.374	–0.219	0.362	0.178
QCA	0.008	–0.003	0.141	–0.279	–0.316	0.290	0.135

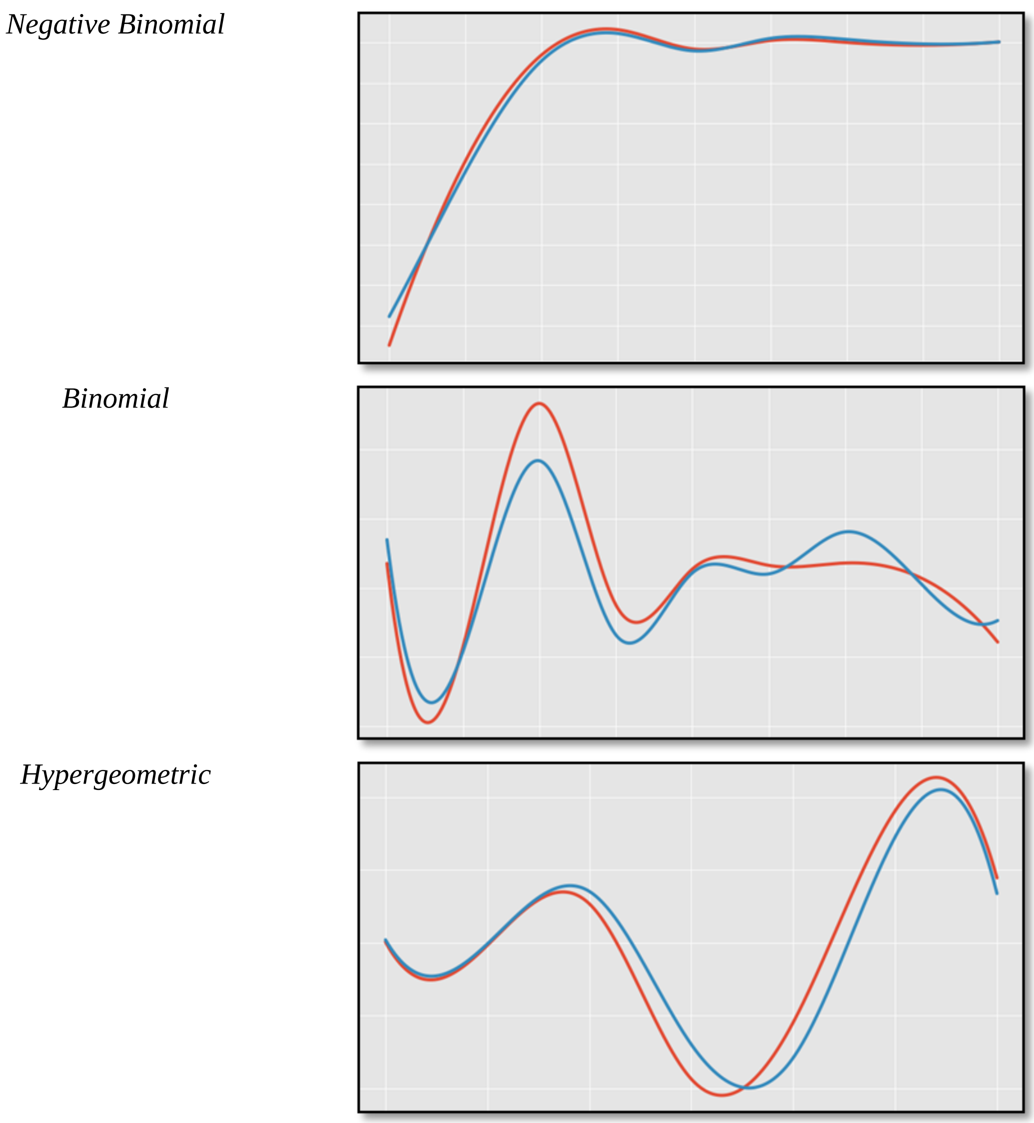

Figure 4. *Polynomial interpolation of CCOV and QCA absolute difference errors.*

Conclusion

Here, we propose a novel quantum algorithm, quantum covariance approximation (QCA), based on amplitude encoding and amplification (the diffusion operator) and variation of QFT, for the multiplication of quantum-state vectors. Its innovative aspect lies in the novel procedure for estimating the covariance of two discrete-value vectors in quantum circuits and operations. We tested the QCA algorithm over five discrete-value distributions, yielding the following main conclusions: (1) Analyzing the averages of the differences between the classical and quantum calculations yields differences of 0.150 for the binomial, 0.102 for the negative binomial, 0.153 for the uniform, 0.063 for Poisson and 0.050 for the hypergeometric distribution. As these values have no upper bound (i.e., their values depend on the input distribution), we conclude that QCA accurately estimated the covariance among these cases; (2) For the binomial distribution, agreement was obtained between all the compared experiments except for a case of probability 0.3. In this case, the absolute difference recorded was 0.415, much higher than the other values. This result could have been due to two main causes: first, the input data are random variables, and it is possible that noisy data were obtained during the iteration. Second, this noise may have arisen from the way that quantum computers works; (3) Generally, the Grover diffusion operator aims to amplify the amplitude around the mean for searching in unstructured data. This holds only for a uniform distribution of the input vector amplitudes. Here, we used the Grover diffusion operator on purpose to invert a state vector around its mean amplitude to find an approximation for the input vectors' standardization.

The main limitation of the QCA algorithm relates to the input type: the algorithm was designed and tested on random distributions of discrete values only, and thus is not suitable for distributions of continuous values. In addition, several factors could affect the accuracy of the results, including the quality of the data and the total number of shots over the quantum simulator. An incorrect adjustment of parameters or a lack of data could also cause this method to fail in approximating the covariance between the input variables. Future work should examine two main issues: first, the potential for generalizing the QCA algorithm to support continuous value distributions, and second, and its use as a ‘black box’ for building other algorithms, such as quantum dimensionality reduction.

Acknowledgement

This research is self-funded.

Conflict of interest

The authors confirm that there is no conflict of interest involve with any parties in this research study.

REFERENCES

[1] Alchieri, L., Badalotti, D., Bonardi, P., Bianco, S. (2021): An introduction to quantum machine learning: from quantum logic to quantum deep learning. – Quantum Machine Intelligence 3(2): 30p.

[2] Behrens, J.T., Yu, C.H. (2003): Exploratory data analysis. – Handbook of Psychology 2: 33-64.
[3] Benedetti, M., Lloyd, E., Sack, S., Fiorentini, M. (2019): Parameterized quantum circuits as machine learning models. – Quantum Science and Technology 4(4): 20p.
[4] Bennett, C.H., Bernstein, E., Brassard, G., Vazirani, U. (1997): Strengths and weaknesses of quantum computing. – SIAM Journal on Computing 26(5): 1510-1523.
[5] Biamonte, J., Wittek, P., Pancotti, N., Rebentrost, P., Wiebe, N., Lloyd, S. (2017): Quantum machine learning. – Nature 549(7671): 195-202.
[6] Blank, C., Park, D.K., Rhee, J.K.K., Petruccione, F. (2020): Quantum classifier with tailored quantum kernel. – NPJ Quantum Information 6(1): 9p.
[7] Bravo-Prieto, C. (2021): Quantum autoencoders with enhanced data encoding. – Machine Learning: Science and Technology 2(3): 11p.
[8] Buffoni, L., Caruso, F. (2021): New trends in quantum machine learning (a). – Europhysics Letters 132(6): 8p.
[9] Chatfield, C. (1986): Exploratory data analysis. – European Journal of Operational Research 23(1): 5-13.
[10] Choi, J., Kim, J. (2019): A tutorial on quantum approximate optimization algorithm (QAOA): Fundamentals and applications. – In 2019 International Conference on Information and Communication Technology Convergence (ICTC), IEEE 5p.
[11] Cleff, T. (2014): Exploratory data analysis in business and economics. – Exploratory Data Analysis in Business and Economics 215p.
[12] Cleve, R., Watrous, J. (2000): Fast parallel circuits for the quantum Fourier transform. – In Proceedings 41st Annual Symposium on Foundations of Computer Science, IEEE 10p.
[13] Cui, J., Fan, H. (2010): Correlations in the Grover search. Journal of Physics A: Mathematical and Theoretical 43(4): 18p.
[14] Data, M.C., Komorowski, M., Marshall, D.C., Salciccioli, J.D., Crutain, Y. (2016): Exploratory data analysis. – Secondary Analysis of Electronic Health Records 18p.
[15] Dilip, R., Liu, Y.J., Smith, A., Pollmann, F. (2022): Data compression for quantum machine learning. – Physical Review Research 4(4): 8p.
[16] Farhi, E., Goldstone, J., Gutmann, S. (2014): A quantum approximate optimization algorithm. – ArXiv:1411.4028 16p.
[17] Gelman, A. (2004): Exploratory data analysis for complex models. – Journal of Computational and Graphical Statistics 13(4): 755-779.
[18] Hadfield, S., Wang, Z., O'gorman, B., Rieffel, E.G., Venturelli, D., Biswas, R. (2019): From the quantum approximate optimization algorithm to a quantum alternating operator ansatz. – Algorithms 12(2): 45p.
[19] Hales, L., Hallgren, S. (2000): An improved quantum Fourier transform algorithm and applications. – In Proceedings 41st Annual Symposium on Foundations of Computer Science, IEEE 10p.
[20] Jang, K., Song, G., Kwon, H., Uhm, S., Kim, H., Lee, W.K., Seo, H. (2021): Grover on PIPO. – Electronics 10(10): 18p.
[21] Jebb, A.T., Parrigon, S., Woo, S.E. (2017): Exploratory data analysis as a foundation of inductive research. – Human Resource Management Review 27(2): 265-276.
[22] Karami, S., Saberi-Movahed, F., Tiwari, P., Marttinen, P., Vahdati, S. (2023): Unsupervised feature selection based on variance–covariance subspace distance. – Neural Networks 166: 188-203.
[23] Koren, M., Peretz, O. (2024): Automated Data-Driven and Stochastic Imputation Method. – Intechopen 19p.
[24] Koren, M., Peretz, O. (2024): A quantum procedure for estimating information gain in Boolean classification task. – Quantum Machine Intelligence 6(1): 8p.
[25] Koren, M., Koren, O., Peretz, O. (2023a): A quantum "black box" for entropy calculation. – Quantum Machine Intelligence 5(2): 8p.

[26] Koren, O., Koren, M., Peretz, O. (2023b): A procedure for anomaly detection and analysis. – Engineering Applications of Artificial Intelligence 117: 8p.
[27] LaRose, R., Coyle, B. (2020): Robust data encodings for quantum classifiers. – Physical Review A 102(3): 24p.
[28] Leinhardt, S., Wasserman, S.S. (1979): Exploratory data analysis: An introduction to selected methods. – Sociological Methodology 10: 311-365.
[29] Li, G., Ye, R., Zhao, X., Wang, X. (2022): Concentration of data encoding in parameterized quantum circuits. – Advances in Neural Information Processing Systems 35: 19456-19469.
[30] Majji, S.R., Chalumuri, A., Manoj, B.S. (2023): Quantum Approach to Image Data Encoding and Compression. – IEEE Sensors Letters 7(2): 1-4.
[31] Mandal, S.B., Chakrabarti, A., Sur-Kolay, S. (2014): Synthesis of ternary Grover's algorithm. – In 2014 IEEE 44th International Symposium on Multiple-Valued Logic, IEEE 6p.
[32] Mashhadi, S. (2019): General secret sharing based on quantum Fourier transform. – Quantum Information Processing 18(4): 15p.
[33] Moore, C., Rockmore, D., Russell, A. (2006): Generic quantum Fourier transforms. – ACM Transactions on Algorithms (TALG) 2(4): 707-723.
[34] Morgenthaler, S. (2009): Exploratory data analysis. – Wiley Interdisciplinary Reviews: Computational Statistics 1(1): 33-44.
[35] Nachman, B., Urbanek, M., de Jong, W.A., Bauer, C.W. (2020): Unfolding quantum computer readout noise. – NPJ Quantum Information 13p.
[36] Nakamura, S. (1995): Numerical analysis and graphic visualization with MATLAB. – Prentice-Hall, Inc. 519p.
[37] Nam, Y., Su, Y., Maslov, D. (2020): Approximate quantum Fourier transform with O (n log (n)) T gates. – NPJ Quantum Information 6(1): 6p.
[38] Peretz, O., Koren, M. (2024): A parameterized quantum circuit for estimating distribution measures. – Quantum Machine Intelligence 6(1): 1-9.
[39] Piattini, M., Peterssen, G., Pérez-Castillo, R. (2021): Quantum computing: A new software engineering golden age. – ACM SIGSOFT Software Engineering Notes 45(3): 12-14.
[40] Romero, J., Olson, J.P., Aspuru-Guzik, A. (2017): Quantum autoencoders for efficient compression of quantum data. – Quantum Science and Technology 2(4): 13p.
[41] Ruiz-Perez, L., Garcia-Escartin, J.C. (2017): Quantum arithmetic with the quantum Fourier transform. – Quantum Information Processing 16: 1-14.
[42] Self, C.N., Khosla, K.E., Smith, A.W., Sauvage, F., Haynes, P.D., Knolle, J., Mintert, F., Kim, M.S. (2021): Variational quantum algorithm with information sharing. – NPJ Quantum Information 7(1): 10p.
[43] Shi, H.L., Liu, S.Y., Wang, X.H., Yang, W.L., Yang, Z.Y., Fan, H. (2017): Coherence depletion in the Grover quantum search algorithm. – Physical Review A 95(3): 9p.
[44] Shin, S., Teo, Y.S., Jeong, H. (2023): Exponential data encoding for quantum supervised learning. – Physical Review A 107(1): 21p.
[45] Sierra-Sosa, D., Pal, S., Telahun, M. (2023): Data rotation and its influence on quantum encoding. – Quantum Information Processing 22(1): 89p.
[46] Tukey, J.W. (1977): Exploratory data analysis. – Reading, MA: Addison-Wesley 2: 29p.
[47] Tulsi, A. (2015): Faster quantum searching with almost any diffusion operator. – Physical Review A 91(5): 3p.
[48] Vigni, M.L., Durante, C., Cocchi, M. (2013): Exploratory data analysis. – In Data Handling in Science and Technology, Elsevier 28: 55-126
[49] Vorobyov, V., Zaiser, S., Abt, N., Meinel, J., Dasari, D., Neumann, P., Wrachtrup, J. (2021): Quantum Fourier transform for nanoscale quantum sensing. – NPJ Quantum Information 7(1): 8p.

[50] Wang, G., Zhao, B., Wu, B., Zhang, C., Liu, W. (2023): Intelligent prediction of slope stability based on visual exploratory data analysis of 77 in situ cases. – International Journal of Mining Science and Technology 33(1): 47-59.
[51] Weigold, M., Barzen, J., Leymann, F., Salm, M. (2021): Encoding patterns for quantum algorithms. – IET Quantum Communication 2(4): 141-152.
[52] Weigold, M., Barzen, J., Leymann, F., Salm, M. (2020): Data encoding patterns for quantum computing. – In Proceedings of the 27th Conference on Pattern Languages of Programs 11p.
[53] Weinstein, Y.S., Pravia, M.A., Fortunato, E.M., Lloyd, S., Cory, D.G. (2001): Implementation of the quantum Fourier transform. – Physical Review Letters 86(9): 6p.
[54] Wiebe, N. (2020): Key questions for the quantum machine learner to ask themselves. – New Journal of Physics 22(9): 14p.
[55] Willsch, M., Willsch, D., Jin, F., De Raedt, H., Michielsen, K. (2020): Benchmarking the quantum approximate optimization algorithm. – Quantum Information Processing 19: 1-24.
[56] Wongsuphasawat, K., Liu, Y., Heer, J. (2019): Goals, process, and challenges of exploratory data analysis: An interview study. – ArXiv Preprint 10p.
[57] Xu, B., Heidari, A.A., Cai, Z., Chen, H. (2023): Dimensional decision covariance colony predation algorithm: global optimization and high−dimensional feature selection. – Artificial Intelligence Review 56(10): 11415-11471.
[58] Ying, M. (2010): Quantum computation, quantum theory and AI. – Artificial Intelligence 174(2): 162-176.
[59] Zhandry, M. (2012): How to construct quantum random functions. – In 2012 IEEE 53rd Annual Symposium on Foundations of Computer Science, IEEE 9p.
[60] Zhang, B., Sone, A., Zhuang, Q. (2022): Quantum computational phase transition in combinatorial problems. – NPJ Quantum Information 8(1): 11p.
[61] Zhang, J., Chen, X. (2019): Robust sufficient dimension reduction via ball covariance. – Computational Statistics & Data Analysis 140: 144-154.
[62] Zhou, S., Loke, T., Izaac, J.A., Wang, J.B. (2017): Quantum Fourier transform in computational basis. – Quantum Information Processing 16(3): 19p.