Robust Quantum Feature Selection Algorithm

Shichao Zhang#, *Senior Member*, IEEE, Jiaye Li#, Hang Xu, Hao Yu and Jinjing Shi*, *Member*, IEEE

Abstract—High dimensional data has been a notoriously challenging issue. Existing quantum dimension reduction technology mainly focuses on quantum principal component analysis. There are only a few works on the direction of quantum feature selection algorithm which they are not robust. Also, there are few quantum circuits designed for feature selection, in which some steps are not quantized yet. For example, existing quantum circuits cannot solve the objective function based on sparse learning. To deal with these issues, this paper proposes a robust quantum feature selection by designing a new quantum circuit. Specifically, the sparse regularization term and least squares loss are first applied to construct the proposed objective function. And then, six kinds of quantum registers and their initial states are prepared. In addition, quantum techniques, such as quantum phase estimation and controlled rotation, are employed to construct an alternating iterative quantum circuit to obtain the final quantum state of the feature selection variable. Finally, a series of experiments are conducted to verify that the proposed algorithm can accurately select important features and has good robustness.

Index Terms—High dimensional data; quantum feature selection; phase estimation; controlled rotation

1 Introduction

Quantum machine learning can further optimize traditional machine learning algorithms by virtue of the high parallelism of quantum computing [1] [2]. In recent years, with the development of quantum computing technology, a series of quantum machine learning algorithms have been proposed, such as quantum support vector machine [3] [4], quantum k nearest neighbor [4] [5] and quantum ridge regression [6]. The quantum dimension reduction algorithm is a quantum machine learning algorithm that can reduce the feature dimension through low time complexity. Since Lloyd proposed quantum principal component analysis (PCA) in 2014, many researchers have focused on quantum dimension reduction algorithms [7]. For example, improved quantum PCA and quantum feature selection based on quantum optimization technology.

Quantum feature selection combines the characteristics of quantum computing and feature selection, *i.e.,* it can rapidly (with low time complexity) reduce the dimensions of high-dimensional data [8]. At present, there are a few quantum feature selection algorithms. For example, Desu *et al.* proposed adiabatic quantum feature selection [9]. Nembrini *et al.* proposed quantum feature selection for recommendation systems [10]. Agrawal proposed feature selection based on quantum whale optimization algorithm [11]. These algorithms all use quantum technology to achieve feature selection. But they only use quantum computing in part of the algorithm, and there is no complete quantum circuit to achieve feature selection. Different from the above works, Liu *et al.* proposed quantum relief algorithm [12]. Although it enables quantum feature selection through quantum circuit, it still has part of the steps that are non quantum, and can not achieve robust effect.

Among feature selection algorithms, feature selection algorithms based on sparse learning occupy a very important position [13]. They learn the sparse solution of feature weight to select features. However, there is no related work on quantum feature selection algorithm circuits based on sparse learning. Given a training dataset $X \in \mathbb{R}^{n\times d}$ and label $Y \in \mathbb{R}^{n\times c}$, where n represents the number of samples and d represents the number of features. c represents the number of labels. When $c = 1$, it is a single label, otherwise, multi-labels. At present, the objective function of common feature selection based on sparse learning is as follows:

$$\min_{W} \|Y - XW\|_F^2 + \|W\|_s \tag{1}$$

where $\|W\|_s$ is the sparse regularization term, such as $l_{2,1}$, l_1 and $l_{2,p}$ norm. W is the final feature selection model. For Eq. (1), there is no quantum circuit that can properly solve it. In quantum computing, the only relevant algorithm is HHL [14], which is a quantum algorithm for solving linear equations. The HHL algorithm solves the following problems:

$$Xw = y \tag{2}$$

where X is a known data matrix and y is a label. Obviously $w = X^{-1}y$. In HHL algorithm, phase estimation and controlled rotation are used to obtain the quantum state $|w\rangle$ of $\mathbf{w}$. Compared with the classical algorithm, HHL has exponential acceleration [15]. Comparing Eq. (2) and Eq. (1), it is easy to find that the difference between them is relatively large. Although Eq. (2) can be solved by quantum circuit, it is still a huge challenge to design appropriate quantum circuit to solve the sparse learning based objective function similar to Eq. (1).

To solve these problems, this paper proposes a robust quantum feature selection algorithm. It can not only use the quantum circuit to obtain the sparse solution of the feature weight, but also learn the weight of each sample for robust learning. Specifically, it first uses sparse learning and least squares loss to construct the objective function. And then, an alternative iterative quantum circuit is used to solve the proposed objective function, so as to obtain the quantum states of final feature weight and sample weight. Finally, a series of experiments are carried out on IBM quantum platform to verify the validity of the proposed algorithm.

- *Shichao Zhang and Jiaye Li contributed equally to this work. School of Computer Science and Engineering, Central South University, Changsha 410083, PR China.*

E-mail: zhangsc@csu.edu.cn, lijiaye@csu.edu.cn, xuhangxhhh@163.com, yhgxnu@gmail.com and shijinjing@csu.edu.cn
Corresponding author: Jinjing Shi.

*Manuscript received ** , 2022; revised **, 2022.*

The main contributions of this paper are as follows:

- A novel alternately iterative quantum circuit is proposed for feature selection to obtain quantum states with final feature weights. This circuit can be used to solve similar objective functions based on sparse learning.
- Different from the existing quantum feature selection algorithms, the proposed algorithm can perform robust quantum feature selection. *i.e.,* when there is noise in the data, the proposed algorithm can still perform accurate feature selection.
- In the experiment, a series of experiments are carried out with IBM' qiskit and Matlab to verify the validity of the proposed algorithm. Experiments show that the proposed algorithm can accurately select important features, and shows good robustness in the data with gaussian white noise.

The rest of this paper is organized as follows. Section 2 briefly reviews previous related work. Section 3 describes our proposed method in detail, quantum circuit and time complexity analysis. Section 4 shows the results of algorithm on dataset. Section 5 summarizes the full paper.

2 Related Work

In this section, we first introduce the relevant quantum technologies used in this paper, and then introduce some existing quantum dimension reduction algorithms.

2.1 Correlated quantum technology

Quantum fourier transform The discrete fourier transform can transform a complex vector $a_0, a_1, \ldots, a_{N-1}$ into a new complex vector $b_0, b_1, \ldots, b_{N-1}$. where

$$b_k = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} e^{2\pi i \frac{jk}{N}} a_j \tag{3}$$

The quantum fourier transform does the same mapping as the classical fourier transform [16] [17]. It is defined as a unitary operator F acting on n quantum bits. Its role is

$$F(|j\rangle) = \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} e^{2\pi i \frac{jk}{2^n}} |k\rangle \tag{4}$$

where $2^n = N$. The quantum fourier transform can transform any n dimensional quantum state $\sum_{j=0}^{N-1} a_j |j\rangle$ into another n dimensional quantum state $\sum_{k=0}^{N-1} b_k |k\rangle$. *i.e.,* $\sum_{j=0}^{N-1} a_j |j\rangle \rightarrow \sum_{k=0}^{N-1} b_k |k\rangle$. where b_k is obtained from Eq. (3). After some operations, we can get the product form of fourier transform as follows:

$$\begin{aligned}
F(|j\rangle) &= \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} e^{\frac{2\pi i j k}{2^n}} |k\rangle \\
&= \frac{1}{\sqrt{2^n}} \sum_{k_{n-1}=0}^{1} \cdots \sum_{k_0=0}^{1} e^{(2\pi i j \sum_{l=1}^{n} \frac{k_{n-l}}{2^l})} |k_{n-1} \cdots k_0\rangle \\
&= \frac{1}{\sqrt{2^n}} \sum_{k_{n-1}=0}^{1} \cdots \sum_{k_0=0}^{1} \otimes_{l=1}^{n} e^{(2\pi i j \frac{k_{n-l}}{2^l})} |k_{n-l}\rangle \\
&= \frac{1}{\sqrt{2^n}} \otimes_{l=1}^{n} [\sum_{k_{n-l}=0}^{1} e^{(2\pi i j \frac{k_{n-l}}{2^l})} |k_{n-l}\rangle] \\
&= \frac{1}{\sqrt{2^n}} \otimes_{l=1}^{n} [|0\rangle + e^{(2\pi i j \frac{1}{2^l})} |1\rangle] \\
&= \frac{1}{\sqrt{2^n}} (|0\rangle + e^{(2\pi i 0.j_0)} |1\rangle)(|0\rangle + e^{(2\pi i 0.j_1 j_0)} |1\rangle) \\
&\cdots (|0\rangle + e^{(2\pi i 0.j_{n-1} j_{n-2} \cdots j_1 j_0)} |1\rangle)
\end{aligned} \tag{5}$$

where $j = j_{n-1} j_{n-2} \cdots j_1 j_0$ is the binary representation of j and $0.j_l j_{l+1} \cdots j_m = j_l/2 + j_{l+1}/2^2 + \cdots + j_m/2^{m-l+1}$ is the binary fractional representation. According to Eq. (5), the circuit of quantum fourier transform can be easily obtained, as shown in Fig. 1:

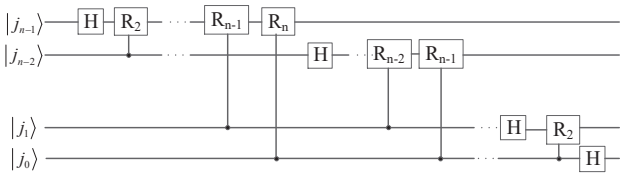


Fig. 1. Circuit of quantum fourier transform.

where the R_k operator is as follows:

$$R_k = \begin{bmatrix} 1 & 0 \\ 0 & e^{(2\pi i/2^k)} \end{bmatrix} \tag{6}$$

From the Fig. 1, we can see that need n quantum bits, n hadamard gates and $n(n-1)/2$ controlled phase shift gates. Therefore, its time complexity is $O(n^2)$, while the time complexity of classical fourier transform is $O(n2^n)$. It can be seen that the quantum fourier transform has an exponential acceleration compared with the classical fourier transform [18].

Quantum phase estimation Quantum phase estimation [19] [20] can be used to calculate the phase of the eigenvalue of a given unitary operator U, *i.e.,* solving φ in $U|u\rangle = e^{2\pi i\varphi}|u\rangle$, where $|u\rangle$ is the eigenvector of U. The specific process can be divided into the following three steps:

1. It prepares quantum registers with initial states $|u\rangle$ and $|0\rangle^{\otimes t}$ respectively, and transfers the eigenvalue phase decomposition of U to the amplitude of the auxiliary quantum bit by using a series of rotating quantum gate operations.

2. An inverse quantum fourier transform is performed on the auxiliary bit (*i.e.,* $|0\rangle^{\otimes t}$) to transfer the phase of the eigenvalue on the amplitude to the basis vector.

3. All auxiliary qubits are tested separately to obtain the phase information of the eigenvalues.

The quantum circuit [21] of quantum phase estimation is as follows:

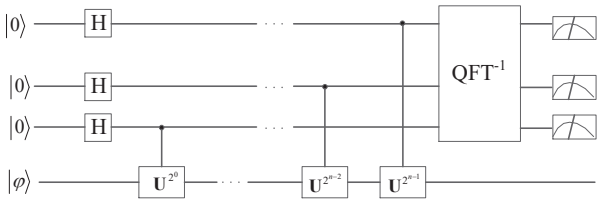


Fig. 2. Circuit of quantum phase estimation.

From Fig. 2, we can find that quantum phase estimation uses inverse quantum fourier transform, which can be seen as a typical application of quantum fourier transform [22]. Both quantum fourier transform and quantum phase estimation play a powerful role. In this paper, the proposed quantum feature selection algorithm combines them. The specific process will be detailed in the next section.

2.2 Quantum dimensionality reduction

According to different core ideas, dimension reduction can be divided into many types, such as principal components analysis (PCA) [23], linear discriminant analysis (LDA) [24], locality preserving projections (LPP) [25], local linear embedding (LLE) [26] and feature selection [27]. At present, the main quantum dimension reduction technologies are quantum PCA and quantum feature selection. Therefore, we focus on introducing some quantum PCA algorithms and quantum feature selection algorithms in this part.

Quantum principal component analysis Principal component analysis is a typical dimension reduction algorithm [28]. It converts a group of variables that may have correlation into a group of linearly unrelated variables through orthogonal transformation, and the transformed group of variables is called principal component [29]. *i.e.,* the data after dimension reduction retains most of the information of the original data [30]. With the development of quantum technology, Lloyd *et al.* proposed the quantum principal component analysis algorithm in 2014 [7]. Specifically, it first replicates copies of quantum states of multiple data covariance matrices through hamiltonian simulation technology. Then quantum phase estimation is used to obtain the eigenvalues and eigenvectors of the quantum state. Finally, it obtains the eigenvectors corresponding to the larger eigenvalues through sampling, *i.e.,* the principal components. This method obtains all eigenvalues and eigenvectors of the covariance matrix, so a large number of samples are required. In order to solve this problem, Lin *et al.* proposed a quantum PCA algorithm based on singular value thresholding (qsvt), which does not need to extract all eigenvalues and sample a lot, greatly reducing the number of samples [31]. However, its quantum circuit has two unitary operations of estimation, and the accuracy is low. Later, Daskin proposed a quantum PCA algorithm, which combines amplitude amplification and quantum phase estimation to estimate eigenvalues in the interval $[a, b]$ [32]. Subsequently, Yu *et al.* proposed a new quantum PCA algorithm, which uses the quantum state of high-dimensional data and the basic quantum state to do quantum exchange tests to obtain the process of mapping high-dimensional data to low dimensional data, so as to reduce the dimension of high-dimensional data [33].

To sum up, we find that the existing quantum PCA algorithm mainly uses quantum phase estimation technology and sampling. This reveals the direction for the subsequent quantum dimension reduction algorithm.

Quantum feature selection Feature selection is also a dimension reduction algorithm [34]. Its core is to select the feature subset that best represents the overall data information through loss function and sparse regular term. It can be seen as a data preprocessing method [35] [36]. This is different from the core technology of principal component analysis. the weight of each feature is learnt, so as to further select the feature subset. PCA obtains the eigenvalues and eigenvectors according to the singular value decomposition of the covariance matrix to obtain the principal components, so as to achieve dimension reduction [37].

Quantum feature selection is a quantum algorithm that quantizes all or some steps in the process of feature selection algorithm, and further realizes accelerated dimensionality reduction [38] [39]. Chakraborty *et al.* proposed a feature selection algorithm based on quantum heuristic graph theory [40]. Specifically, it first uses the distance function to build an undirected graph of data, which belongs to the classical part. And then, it uses pearson coefficient to calculate the relationship between features and construct similarity matrix. Finally, it designs a quantum oracle to obtain the adjacency matrix of features, and uses quantum parallel amplitude estimation amplification technology to obtain highly correlated features. This method transforms data into graph form and combines quantum technology to select features. However, the first two steps are classical calculations, not involving quantum computation. Wang *et al.* proposed a quantum feature selection algorithm based on the quantum grasshopper optimization algorithm [41]. It first introduced quantum computing into the grasshopper optimization algorithm, and obtained the quantum version of the grasshopper optimization algorithm. Then mutual information is introduced into the objective function to learn the relationship between the selected feature subset and the unselected feature. Finally, it compares the local optimal solution under each quantum grasshopper to obtain an approximate global optimal solution. Otgonbaatar and Datcu proposed a quantum annealing algorithm for feature selection and classification [42]. This method selects informative features for each category in the hyperspectral image and applies the quantum classifier to the dataset on the D-wave quantum annealer. The feature selection process of this method does not use quantum computing, *i.e.,* only part of the steps still use quantum technology. Desu *et al.* proposed an adiabatic quantum feature selection algorithm [9]. Specifically, it first uses the least squares loss function and l_0 norm to construct the objective function. Then it uses adiabatic quantum computation to solve the proposed sparse linear objective function. Finally, it uses a D-wave adiabatic quantum computer to solve the quadratic unconstrained quadratic optimization problem. Although this method can solve the sparse problem with quantum technology, it does not give a complete quantum circuit diagram. Li *et al.* proposed a quantum approximation algorithm to achieve feature selection in graph theory [43]. Specifically, it first uses similarity calculation to build three graphs on the original data, in which features represent nodes in the graph, and the relationship between features represents edges. Then, it uses quantum approximate optimization algorithm (QAOA) to derive a subgraph from the established graph to generate feature subsets. Finally, it combines tabu search algorithm to use finite qubits for large-scale feature selection of graph theory. In this method, the used core quantum technology is QAOA, part of which is still non quantized. Liu *et al.* proposed the quantum relief algorithm [12]. This algorithm can be regarded as a quantum version of the classical relief algorithm. Specifically, it first converts the dataset into a quantum superposition state and prepares the initial state. The core technology of this step is hadamard operation and rotary unitary operation. Then it uses swap to test quantum operation for obtaining quantum state, and uses inner product to obtain similarity. In addition, it also uses the classical maximum search algorithm to find the sample with the largest similarity. Finally, it updates the weights and selects features by iteration. Although quantum technology is applied in the initial process and some intermediate steps of the algorithm, some steps are still non quantized.

3 Our Approach

In this section, we first build the objective function of the proposed feature selection algorithm step by step. Then we construct a quantum circuit to solve the objective function. Finally, the time complexity of the proposed algorithm is analyzed.

TABLE 1
The detail of the notations used in this paper.

X	training data
y	label of training data
X^i	the i-th row of X
X_j	the j-th column of X
$\|X\|_F$	the frobenius norm of X, *i.e.*, $\|X\|_F = \sqrt{\sum_{i,j} x_{i,j}^2}$
$\|X\|_{2,1}$	the $l_{2,1}$ norm of X, *i.e.*, $\|X\|_{2,1} = \sum_i (\sum_j x_{ij}^2)^{1/2}$
$\|X\|_1$	the l_1 -norm of X, *i.e.*, $\|X\|_1 = \sum_i \sum_j \|x_{ij}\|$
X^T	the transpose of X
X^{-1}	the inverse of X
$tr(X)$	the trace of X
W	feature weight variable
Θ	sample weight variable
$k()$	kernel function
H	hadamard gate
$RY(\eta)$	$RY(\eta)$ gate
FT	quantum fourier transform
FT^{-1}	inverse quantum fourier transform

In this paper, we use $X \in \mathbb{R}^{n\times d}$ to represent the data matrix, which contains n samples and each sample contains d features. The frobenius norm, $l_{2,1}$ norm and l_1 norm of X are expressed as $\|X\|_F = (\sum_j \|x_j\|_2^2)^{1/2}$, $\|X\|_{2,1} = \sum_i (\sum_j x_{ij}^2)^{1/2}$ and $\|X\|_1 = \sum_i \sum_j |x_{ij}|$ respectively. The trace, transpose and inverse of matrix X are expressed as $tr(X)$, X^T and X^{-1} respectively. In addition, we also use H to represent hadamard gate, R represents controlled rotation operation, and FT represents quantum fourier transform.

We summarize these notations used in our paper in Table 1.

3.1 Robust quantum feature selection

Since Lloyd proposed the quantum principal component analysis algorithm, some researchers have proposed its improved version. Their core is to use phase estimation to solve the eigenvalues and eigenvectors of the data covariance matrix and select the eigenvectors corresponding to the first few larger eigenvalues to form the principal component, so as to achieve the effect of dimensionality reduction on the data. Feature selection is to select the subset of features that can best represent the overall information of data, and it will not change the form of features [44]. As PCA is an unsupervised dimensionality reduction technology, it is possible to hide feature elements that contribute little to data differences, thereby mistakenly eliminating small and important differentiation factors that affect model performance [45]. In this section, we propose a supervised quantum feature selection algorithm. Given a data matrix X. We can get its reduced singular value decomposition form, as shown below:

$$X = \sum_{i=1}^{A} \lambda_i |u_i\rangle \langle v_i| \tag{7}$$

where λ_i is singular value, $|u_i\rangle$ and $\langle v_i|$ are left and right singular vectors respectively, and A is the rank of matrix X.

On this basis, since the class label y corresponds to data X, it can be written as a combination of $|u_i\rangle$, as shown below:

$$y = \|y\| \sum_i \sigma_i |u_i\rangle \tag{8}$$

Eq. (8) can be seen as the representation of label $\mathbf{y}$ on $|u_i\rangle$. Next, we use the data to fit the label through the least squares loss to obtain the coefficient matrix. As follows:

$$\min_W \left\| \sum_{i=1}^{A} \lambda_i |u_i\rangle \langle v_i| W - \|y\| \sum_i \sigma_i |u_i\rangle \right\|_F^2 \tag{9}$$

where W is the fitting coefficient matrix, which records the relationship between data and labels. For feature selection, we perform $l_{2,1}$ sparse restriction on it. Further, we can get the following formula:

$$\min_W \left\| \sum_{i=1}^{A} \lambda_i |u_i\rangle \langle v_i| W - \|y\| \sum_i \sigma_i |u_i\rangle \right\|_F^2 + \alpha \|W\|_{2,1} \tag{10}$$

Eq. (10) can perform feature selection, but it can not eliminate the influence of outliers and noise samples, resulting in the model is not robust. Therefore, we further reconstruct the data to learn the weight of each sample. The final objective function is as follows:

$$\begin{aligned} &\min_{W,\Theta} \left\| \sum_{i=1}^{A} \lambda_i |u_i\rangle \langle v_i| [W, \Theta] - [\|y\| \sum_i \sigma_i |u_i\rangle, k(X)] \right\|_F^2 \\ &+\alpha \|W\|_{2,1} + \beta \|\Theta\|_1 \end{aligned} \tag{11}$$

where Θ records the weight of each sample, and the weights of outliers and noise samples are relatively small, thus reducing their impact on the model and making the model more robust.

For Eq. (11), we propose an alternate iterative method to solve it. In the next section, we will show its quantum circuit. In iteration (t+1), we have obtained the value of $\Theta^{(t)}$ in iteration t. Therefore, when variable Θ is fixed, Eq. (11) can be rewritten as follows:

$$\begin{aligned} &\min_W tr \begin{pmatrix} (W^T \sum_{i=1}^{A} |v_i\rangle \langle u_i| - \|y\| \sum_i \sigma_i \langle u_i|) \\ (\sum_{i=1}^{A} \lambda_i |u_i\rangle \langle v_i| W - \|y\| \sum_i \sigma_i |u_i\rangle) \end{pmatrix} \\ &+\alpha tr(W^T N W) \end{aligned} \tag{12}$$

Further, Eq. (12) is expanded to obtain the following formula:

$$\begin{aligned} &\min_W tr \begin{pmatrix} W^T \sum_{i=1}^{A} |v_i\rangle \langle u_i| \sum_{i=1}^{A} \lambda_i |u_i\rangle \langle v_i| W \\ -W^T \sum_{i=1}^{A} |v_i\rangle \langle u_i| \|y\| \sum_i \sigma_i |u_i\rangle \\ - \|y\| \sum_i \sigma_i \langle u_i| \sum_{i=1}^{A} \lambda_i |u_i\rangle \langle v_i| W \\ + \|y\| \sum_i \sigma_i \langle u_i| \sum_{i=1}^{A} \lambda_i |u_i\rangle \langle v_i| \end{pmatrix} \\ &+\alpha tr(W^T N W) \end{aligned} \tag{13}$$

Next, we use Eq. (13) to take the derivative of W_i, and make the derivative equal to 0, we can get the following formula:

$$\begin{aligned} W_i &= (\sum_{i=1}^{A} \lambda_i |v_i\rangle \langle u_i| \sum_{i=1}^{A} \lambda_i |u_i\rangle \langle v_i| + \alpha N)^{-1} \\ &(\|y\| \sum_i \sigma_i \langle u_i| \sum_{i=1}^{A} \lambda_i |u_i\rangle \langle v_i|) \\ &= \sum_{i=1}^{A} \frac{\lambda_i}{\lambda_i^2 + \alpha N_{ii}} \|y\| \sigma_i \langle v_i| \end{aligned} \tag{14}$$

where $N^{ii} = \frac{1}{2\|W^i\|_2} i = (1, 2, \ldots, d)$. After obtaining each W_i, we can splice them together to get W. Similarly, in iteration $(t+$

1), we have obtained the value of $W^{(t)}$ in iteration t. Therefore, when the variable W is fixed, Eq. (11) can be rewritten as follows:

$$\min_{W,\Theta} \left\| \sum_{i=1}^{A} \lambda_i |u_i\rangle \langle v_i| \Theta - \sum_{i=1}^{A} \lambda_i |u_i\rangle \langle v_i| \sum_{i=1}^{A} \lambda_i |v_i\rangle \langle u_i| \right\|_F^2 + \beta \|\Theta\|_1 \tag{15}$$

When we write Eq. (15) in the form of trace, we can get the following formula:

$$\min_{\Theta} tr \begin{pmatrix} (\Theta^T \sum_{i=1}^{A} \lambda_i |v_i\rangle \langle u_i| \\ - \sum_{i=1}^{A} \lambda_i |u_i\rangle \langle v_i| \sum_{i=1}^{A} \lambda_i |v_i\rangle \langle u_i|) \\ (\sum_{i=1}^{A} \lambda_i |u_i\rangle \langle v_i| \Theta \\ - \sum_{i=1}^{A} \lambda_i |u_i\rangle \langle v_i| \sum_{i=1}^{A} \lambda_i |v_i\rangle \langle u_i|) \end{pmatrix} + \beta tr(\Theta^T M \Theta) \tag{16}$$

Further, Eq. (16) can be rewritten as follows:

$$\min_{\Theta} tr \begin{pmatrix} \Theta^T \sum_{i=1}^{A} \lambda_i |v_i\rangle \langle u_i| \sum_{i=1}^{A} \lambda_i |u_i\rangle \langle v_i| \Theta \\ -\Theta^T \sum_{i=1}^{A} \lambda_i |v_i\rangle \langle u_i| \sum_{i=1}^{A} \lambda_i |u_i\rangle \langle v_i| \\ \sum_{i=1}^{A} \lambda_i |v_i\rangle \langle u_i| \\ - \sum_{i=1}^{A} \lambda_i |u_i\rangle \langle v_i| \sum_{i=1}^{A} \lambda_i |v_i\rangle \langle u_i| \\ \sum_{i=1}^{A} \lambda_i |u_i\rangle \langle v_i| \Theta \\ + \sum_{i=1}^{A} \lambda_i |u_i\rangle \langle v_i| \sum_{i=1}^{A} \lambda_i |v_i\rangle \langle u_i| \\ \sum_{i=1}^{A} \lambda_i |u_i\rangle \langle v_i| \sum_{i=1}^{A} \lambda_i |v_i\rangle \langle u_i| \end{pmatrix} + \beta tr(\Theta^T M \Theta) \tag{17}$$

We use Eq. (17) to take the derivative of Θ_i, and let the derivative be 0, we can get the following formula:

$$\begin{aligned} &\Theta_i = (\sum_{i=1}^{A} \lambda_i |v_i\rangle \langle u_i| \sum_{i=1}^{A} \lambda_i |u_i\rangle \langle v_i| + \beta M)^{-1} \\ &(\sum_{i=1}^{A} \lambda_i |v_i\rangle \langle u_i| \sum_{i=1}^{A} \lambda_i |u_i\rangle \langle v_i| \sum_{i=1}^{A} \varsigma_i |u_i\rangle) \\ &= \sum_{i=1}^{A} \frac{\lambda_i}{\lambda_i^2 + \beta M_{ii}} \varsigma_i |u_i\rangle \end{aligned} \tag{18}$$

where $M_j^{ii} = \frac{1}{2|\Theta_{ij}|} i = (1, 2, \ldots, d)$. After getting each Θ_i, we splice them together to get Θ. So far, we can obtain the final feature selection variable W and sample weight variable Θ. Different from other quantum feature selection algorithms, our proposed algorithm requires alternate iterations to obtain the final stable feature selection variable. Its specific corresponding quantum circuit will be shown in the next section.

3.2 Quantum circuit for robust quantum feature selection

In this section, we use the quantum circuit to solve the proposed objective function, *i.e.,* Eq. (11), to obtain the final feature selection matrix W and sample weight matrix Θ. The detailed quantum circuits are shown in Figs. 4 and 5.

In Figs. 4 and 5, we show the detailed quantum computing process. Specifically, we first prepare six registers (from bottom to top, they are quantum registers of type 1 to 6.), whose initial state is $(|0\rangle |y\rangle)(|0\rangle^{\otimes n_0})(|0\rangle)(|u\rangle)(|0\rangle^{\otimes n_0})(|0\rangle)$, *i.e.,* The initial state of the first register is $|0\rangle |y_1\rangle, |0\rangle |y_2\rangle, \ldots, |0\rangle |y_c\rangle$, the initial state of the second register is $|0\rangle^{\otimes n_0}$, the initial state of the third register is $|0\rangle, |0\rangle, \ldots, |0\rangle$, the initial state of the fourth register is $|0\rangle |u\rangle, |0\rangle |u\rangle, \ldots, |0\rangle |u\rangle$, the initial state of the fifth register is $|0\rangle^{\otimes n_0}$, and the initial state of the sixth register is $|0\rangle, |0\rangle, \ldots, |0\rangle$, where n_0 represents the number of quantum bits used to store eigenvalues. and

$$|0\rangle |y\rangle = |0\rangle \left(\sum_{i=1}^{A} \sigma_i |u_i\rangle \right) \tag{19}$$

The preparation of $|y\rangle$ state can refer to [6]. After the initial state $(|0\rangle |y\rangle)(|0\rangle^{\otimes n_0})(|0\rangle)(|u\rangle)(|0\rangle^{\otimes n_0})(|0\rangle)$ is prepared, we perform the phase estimation operation. In machine learning, data matrix X is generally not hermitian, so we convert it into the following form:

$$\hat{X} = \begin{bmatrix} 0 & X \\ X^T & 0 \end{bmatrix} \in R^{(n+d)\times(n+d)} \tag{20}$$

After obtaining Eq. (20), we will perform phase estimation on the 1st, 2nd, 4th and 5th registers. At this time, the quantum state of the system is:

$$\sum_{i=1}^{A} \sigma_i (|u_i\rangle |\lambda_i\rangle) |0\rangle \tag{21}$$

$$\sum_{i=1}^{A} \varsigma_i |u_i\rangle |\lambda_i\rangle |0\rangle \tag{22}$$

After obtaining the eigenvalues of matrix $\hat{X}$ through the phase estimation operation, we perform the controlled rotation operation. Specifically, we make the qubit of the third register controlled by $\left|\frac{\pm\lambda_i}{n+d}\right\rangle$ to rotate it from $|0\rangle$ to $\sqrt{1 - C_1^2 h_1^2(\pm\lambda_i, \alpha)} |0\rangle + C_1 h_1(\pm\lambda_i, \alpha) |1\rangle$. Where $C_1 = O(\max_{\lambda_i} h_1(\lambda_i, \alpha))^{-1}$, $h_1(\lambda, \alpha) := \frac{(n+d)\lambda}{\lambda^2 + \alpha N_{ii}}$. The qubit of the sixth register is controlled by $\left|\frac{\pm\lambda_i}{n+d}\right\rangle$ to rotate it from $|0\rangle$ to $\sqrt{1 - C_2^2 h_2^2(\pm\lambda_i, \beta)} |0\rangle + C_2 h_2(\pm\lambda_i, \beta) |1\rangle$. Where $C_2 = O(\max_{\lambda_i} h_2(\lambda_i, \beta))^{-1}$, $h_2(\lambda, \beta) := \frac{(n+d)\lambda}{\lambda^2 + \beta M_{ii}}$. At this time, the quantum state of the whole system is:

$$\begin{cases} \sum_{i=1}^{A} \sigma_i |u_i\rangle |\lambda_i\rangle \left(\frac{C_1(n+d)\lambda_i}{\lambda_i^2 + \alpha N_{11}}\right) |1\rangle + \sqrt{1 - \left(\frac{C_1(n+d)\lambda_i}{\lambda_i^2 + \alpha N_{11}}\right)^2} |0\rangle \\ \sum_{i=1}^{A} \sigma_i |u_i\rangle |\lambda_i\rangle \left(\frac{C_1(n+d)\lambda_i}{\lambda_i^2 + \alpha N_{22}}\right) |1\rangle + \sqrt{1 - \left(\frac{C_1(n+d)\lambda_i}{\lambda_i^2 + \alpha N_{22}}\right)^2} |0\rangle \\ \vdots \\ \sum_{i=1}^{A} \sigma_i |u_i\rangle |\lambda_i\rangle \left(\frac{C_1(n+d)\lambda_i}{\lambda_i^2 + \alpha N_{cc}}\right) |1\rangle + \sqrt{1 - \left(\frac{C_1(n+d)\lambda_i}{\lambda_i^2 + \alpha N_{cc}}\right)^2} |0\rangle \end{cases} \tag{23}$$

$$\begin{cases} \sum_{i=1}^{A} \varsigma_i |u_i\rangle |\lambda_i\rangle \left(\frac{C_2(n+d)\lambda_i}{\lambda_i^2 + \beta M_{11}}\right) |1\rangle + \sqrt{1 - \left(\frac{C_2(n+d)\lambda_i}{\lambda_i^2 + \beta M_{11}}\right)^2} |0\rangle \\ \sum_{i=1}^{A} \varsigma_i |u_i\rangle |\lambda_i\rangle \left(\frac{C_2(n+d)\lambda_i}{\lambda_i^2 + \beta M_{22}}\right) |1\rangle + \sqrt{1 - \left(\frac{C_2(n+d)\lambda_i}{\lambda_i^2 + \beta M_{22}}\right)^2} |0\rangle \\ \vdots \\ \sum_{i=1}^{A} \varsigma_i |u_i\rangle |\lambda_i\rangle \left(\frac{C_2(n+d)\lambda_i}{\lambda_i^2 + \beta M_{nn}}\right) |1\rangle + \sqrt{1 - \left(\frac{C_2(n+d)\lambda_i}{\lambda_i^2 + \beta M_{nn}}\right)^2} |0\rangle \end{cases} \tag{24}$$

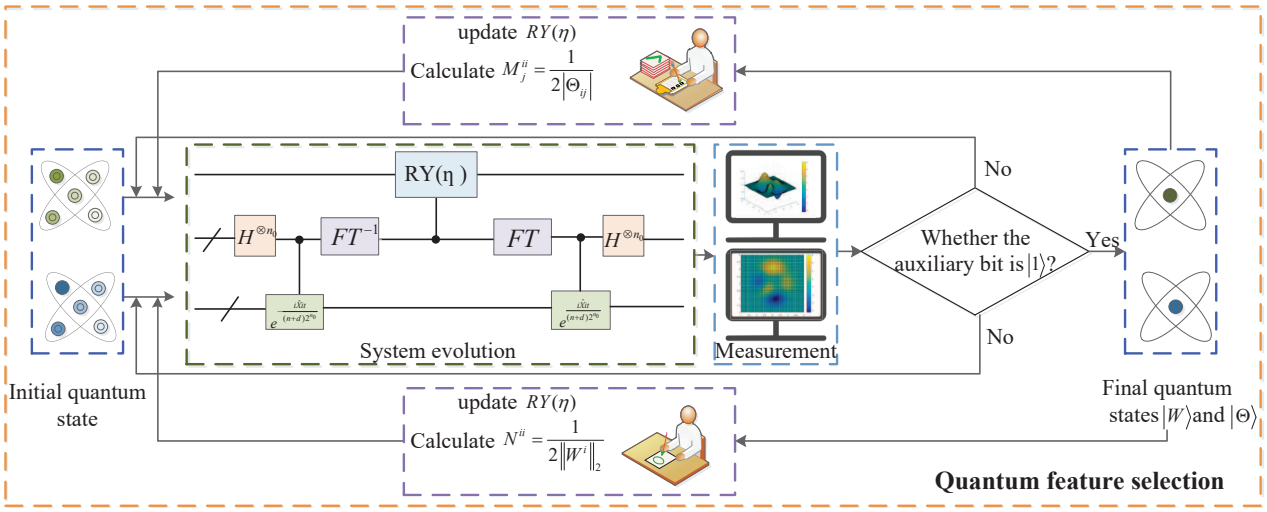


Fig. 3. Flow chart of proposed quantum feature selection algorithm.

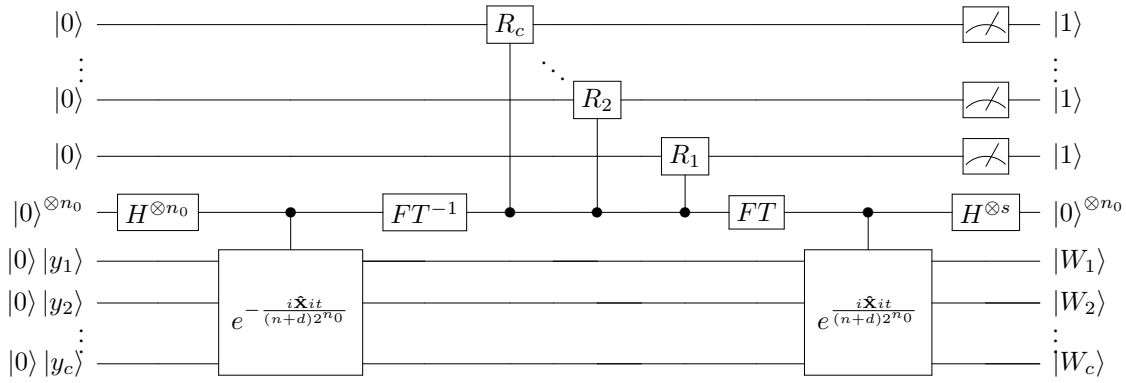


Fig. 4. Quantum circuit for solving $|W\rangle$

The controlled rotation angles of the third register and the sixth register are:

$$\begin{cases} \vartheta(\lambda_i) = \arcsin\left(\frac{C_1(n+d)\lambda_i}{\lambda_i^2+\alpha N_{11}}\right) \\ \vartheta(\lambda_i) = \arcsin\left(\frac{C_1(n+d)\lambda_i}{\lambda_i^2+\alpha N_{22}}\right) \\ \quad\vdots \\ \vartheta(\lambda_i) = \arcsin\left(\frac{C_1(n+d)\lambda_i}{\lambda_i^2+\alpha N_{cc}}\right) \end{cases} \tag{25}$$

$$\begin{cases} \vartheta(\lambda_i) = \arcsin\left(\frac{C_2(n+d)\lambda_i}{\lambda_i^2+\beta M_{11}}\right) \\ \vartheta(\lambda_i) = \arcsin\left(\frac{C_2(n+d)\lambda_i}{\lambda_i^2+\beta M_{22}}\right) \\ \quad\vdots \\ \vartheta(\lambda_i) = \arcsin\left(\frac{C_2(n+d)\lambda_i}{\lambda_i^2+\beta M_{nn}}\right) \end{cases} \tag{26}$$

After that, we perform the inverse operation of phase estimation and test the qubits in the third and sixth registers to obtain test result $|1\rangle$. At this point, we can obtain the quantum state of the first and fourth quantum registers:

$$\begin{cases} |1\rangle\left(\sum_{i=1}^{A}\frac{C_1\sigma_i(n+d)\lambda_i}{\lambda_i^2+\alpha N_{11}}|v_i\rangle\right)\Big/\sqrt{\sum_{i=1}^{A}\left(\frac{C_1\sigma_i(n+d)\lambda_i}{\lambda_i^2+\alpha N_{11}}\right)^2} \\ |1\rangle\left(\sum_{i=1}^{A}\frac{C_1\sigma_i(n+d)\lambda_i}{\lambda_i^2+\alpha N_{22}}|v_i\rangle\right)\Big/\sqrt{\sum_{i=1}^{A}\left(\frac{C_1\sigma_i(n+d)\lambda_i}{\lambda_i^2+\alpha N_{22}}\right)^2} \\ \quad\vdots \\ |1\rangle\left(\sum_{i=1}^{A}\frac{C_1\sigma_i(n+d)\lambda_i}{\lambda_i^2+\alpha N_{cc}}|v_i\rangle\right)\Big/\sqrt{\sum_{i=1}^{A}\left(\frac{C_1\sigma_i(n+d)\lambda_i}{\lambda_i^2+\alpha N_{cc}}\right)^2} \end{cases} \tag{27}$$

$$\begin{cases} |1\rangle\left(\sum_{i=1}^{A}\frac{C_2\varsigma_i(n+d)\lambda_i}{\lambda_i^2+\beta M_{11}}|v_i\rangle\right)\Big/\sqrt{\sum_{i=1}^{A}\left(\frac{C_2\varsigma_i(n+d)\lambda_i}{\lambda_i^2+\beta M_{11}}\right)^2} \\ |1\rangle\left(\sum_{i=1}^{A}\frac{C_2\varsigma_i(n+d)\lambda_i}{\lambda_i^2+\beta M_{22}}|v_i\rangle\right)\Big/\sqrt{\sum_{i=1}^{A}\left(\frac{C_2\varsigma_i(n+d)\lambda_i}{\lambda_i^2+\beta M_{22}}\right)^2} \\ \quad\vdots \\ |1\rangle\left(\sum_{i=1}^{A}\frac{C_2\varsigma_i(n+d)\lambda_i}{\lambda_i^2+\beta M_{nn}}|v_i\rangle\right)\Big/\sqrt{\sum_{i=1}^{A}\left(\frac{C_2\varsigma_i(n+d)\lambda_i}{\lambda_i^2+\beta M_n}\right)^2} \end{cases} \tag{28}$$

After removing $|1\rangle$, we can obtain the quantum states of W and Θ respectively: $|W\rangle$ and $|\Theta\rangle$.

To facilitate readers' understanding, we write the pseudo code of the algorithm in Algorithm 1, and show the whole algorith-

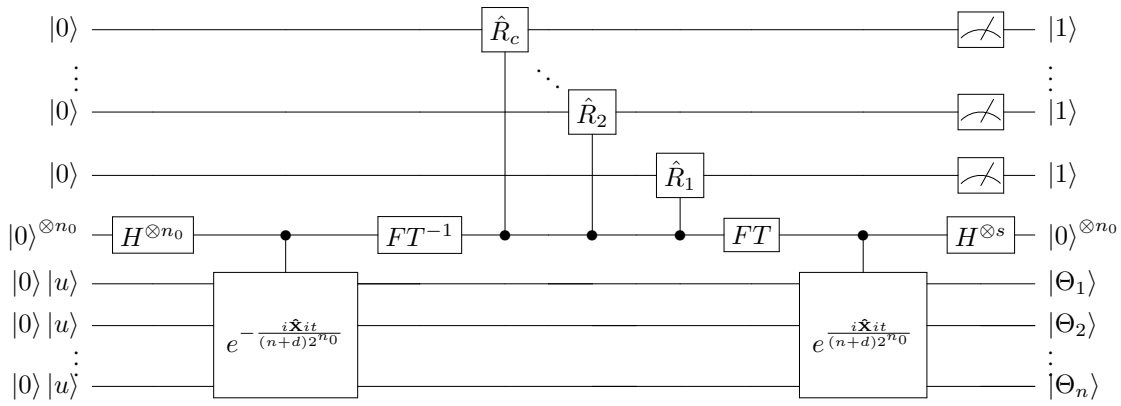


Fig. 5. Quantum circuit for solving $|\Theta\rangle$

m flow in Fig. 3. In algorithm 1, we first initialize variables $W^{(0)}$ and $\Theta^{(0)}$, and then update the quantum states of $W^{(t+1)}$ and $\Theta^{(t+1)}$ by alternating iterations, where t is the number of iterations. Finally, until the algorithm converges, we obtain the final W and Θ. The convergence condition of the algorithm is $\frac{|obj(t+1)-obj(t)|}{obj(t)} \leq 10^{-5}$, where $obj(t)$ and $obj(t+1)$ represent the values of the objective function in iteration t and iteration $(t+1)$ respectively. The core of the algorithm is to use phase estimation and controlled rotation operation in quantum computing. After obtaining the final W and Θ, we can use W to obtain the weight of each feature, so as to select features.

Algorithm 1: Pseudo code for proposed method.

Input: Training set $\mathbf{X} \in \mathbb{R}^{n\times d}$, Labels Y of the training data set, Adjustable parameters α and β;;

Output: $|W\rangle$ and $|\Theta\rangle$;

Initialize *t*=0;

repeat

  Compute $N^{(t)}$ via $N^{ii} = \frac{1}{2\|W^i\|_2} i = (1, 2, \ldots, d)$;

  The quantum state of $W^{(t+1)}$ is obtained by Fig. 4 *i.e.,* Eq. (14);

  Compute $M^{(t)}$ via $M_j^{ii} = \frac{1}{2|\Theta_{ij}|} i = (1, 2, \ldots, d)$;

  The quantum state of $\Theta^{(t+1)}$ is obtained by Fig. 5 *i.e.,* Eq. (18);

  t = *t*+1 ;

until *converge*;

3.3 Time complexity analysis

Through our algorithm process, we can find that its operation cost mainly consists of phase estimation, amplitude amplification and iteration times. According to reference [46], we can get the time complexity of phase estimation is: $O(\|X\|_{\max}^2 poly\log(n+d)\kappa^2/\varepsilon^3)$. By adding the time complexity of amplitude amplification and iteration, we can know that the time complexity of $|W_i\rangle$ is $O(t\|X\|_{\max}^2 poly\log(n+d)\kappa^3/\varepsilon^3)$. Where t is the number of iterations and κ is the condition number (*i.e.,* the ratio of the maximum singular value to the minimum singular value of X). ε is the error term. Similarly, the time complexity of $|\Theta_i\rangle$ is $O(t\|X\|_{\max}^2 poly\log(n+d)\kappa^3/\varepsilon^3)$. Therefore, the total time complexity of our algorithm is $O(2t\|X\|_{\max}^2 poly\log(n+d)\kappa^3/\varepsilon^3)$. Compared with the traditional feature selection algorithm, the time complexity is $O(t(nd + n^2 d\log(\frac{A}{\varepsilon})/\varepsilon^2))$. Assuming $d = O(n)$, $\|X\|_{\max} = O(1)$ and $1/\varepsilon = O(poly\log n)$, the time complexity of the proposed algorithm can be further written as: $O(tpoly\log(n)\kappa^3)$. The time complexity of traditional feature selection algorithm can be further written as $O(tpoly\log(n)n^2A)$. When κ satisfies different conditions, the time complexity comparison between the proposed quantum feature selection algorithm and the traditional feature selection algorithm is shown in Table 2.

TABLE 2
Time complexity comparison between quantum feature selection and non quantum feature selection.

Condition	Traditional feature selection	Proposed
$\kappa = O(\sqrt{n})$	$O(poly\log(n)n^3)$	$O(poly\log(n)n^{3/2})$
$\kappa = poly\log(n)$	$O(poly\log(n)n^2)$	$O(poly\log(n))$

From Table 2, we can see that the proposed quantum feature selection can achieve exponential acceleration when certain conditions are met. *i.e.,* when X is full rank, the proposed method can achieve approximate square acceleration. When X is low rank, the proposed method can achieve approximate exponential acceleration.

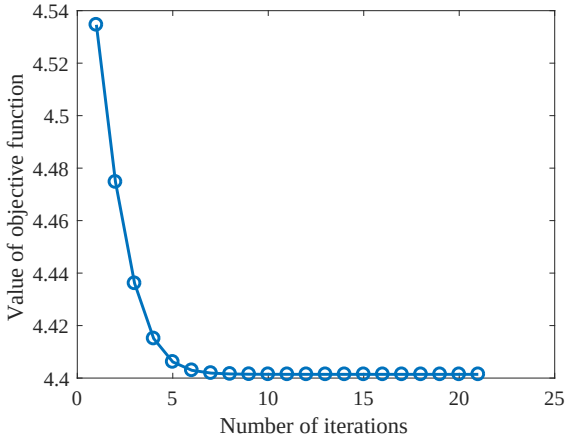
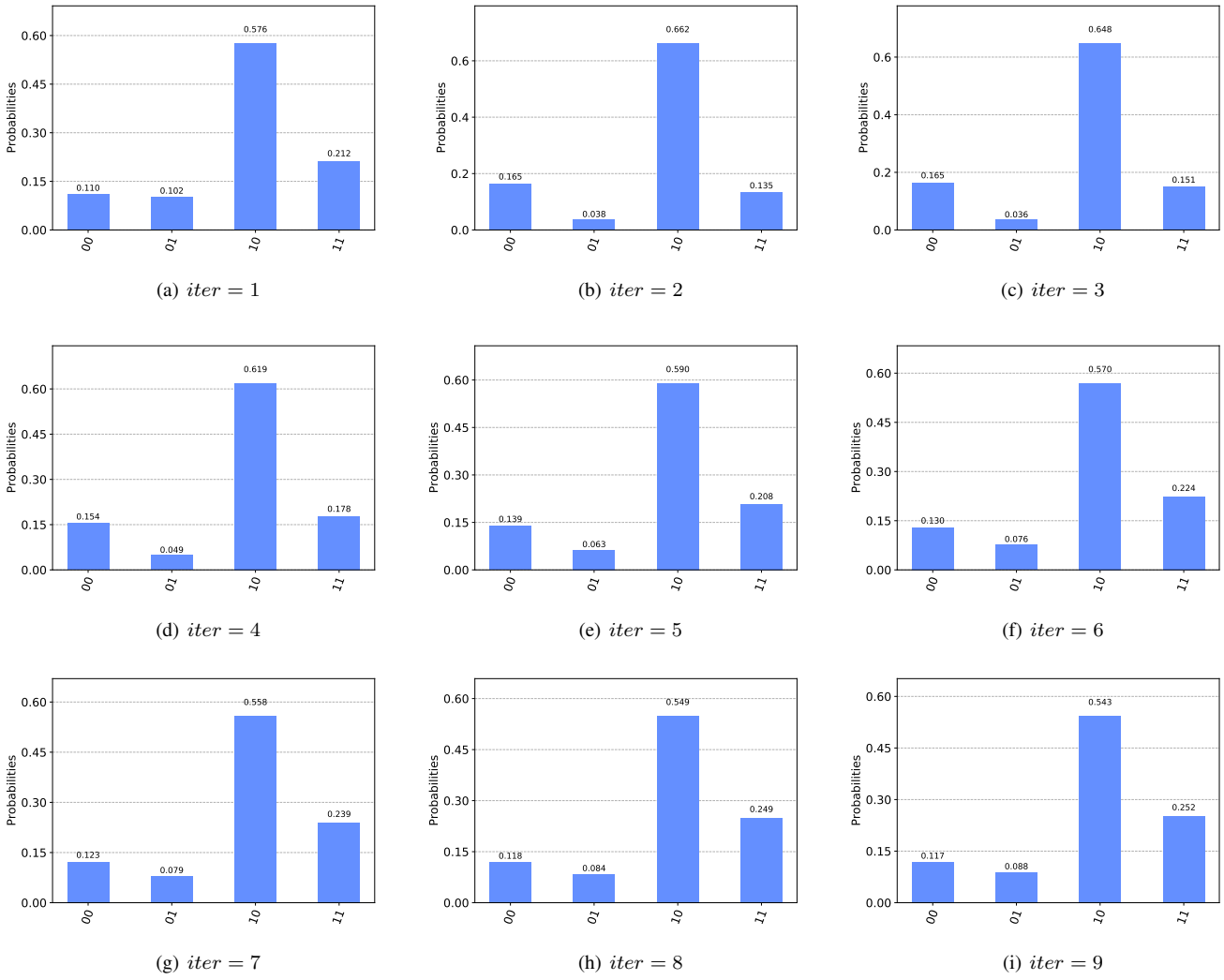


Fig. 10. The value of the objective function in each iteration.



(a) $iter = 1$ (b) $iter = 2$ (c) $iter = 3$

(d) $iter = 4$ (e) $iter = 5$ (f) $iter = 6$

(g) $iter = 7$ (h) $iter = 8$ (i) $iter = 9$

Fig. 6. Measurement results of $|W\rangle$ at each iteration.

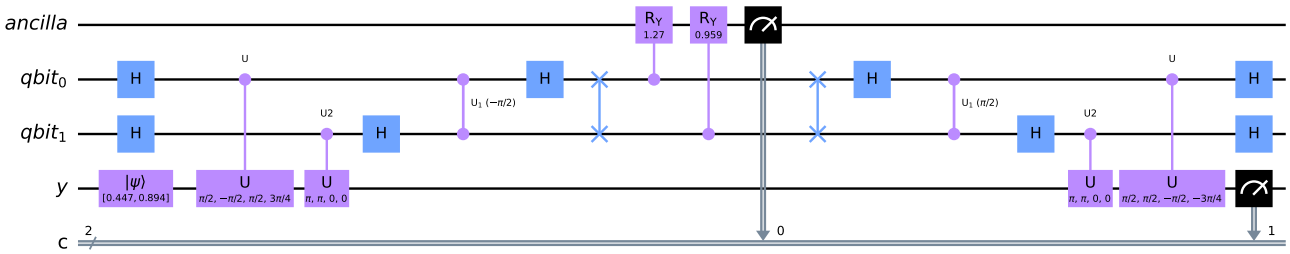


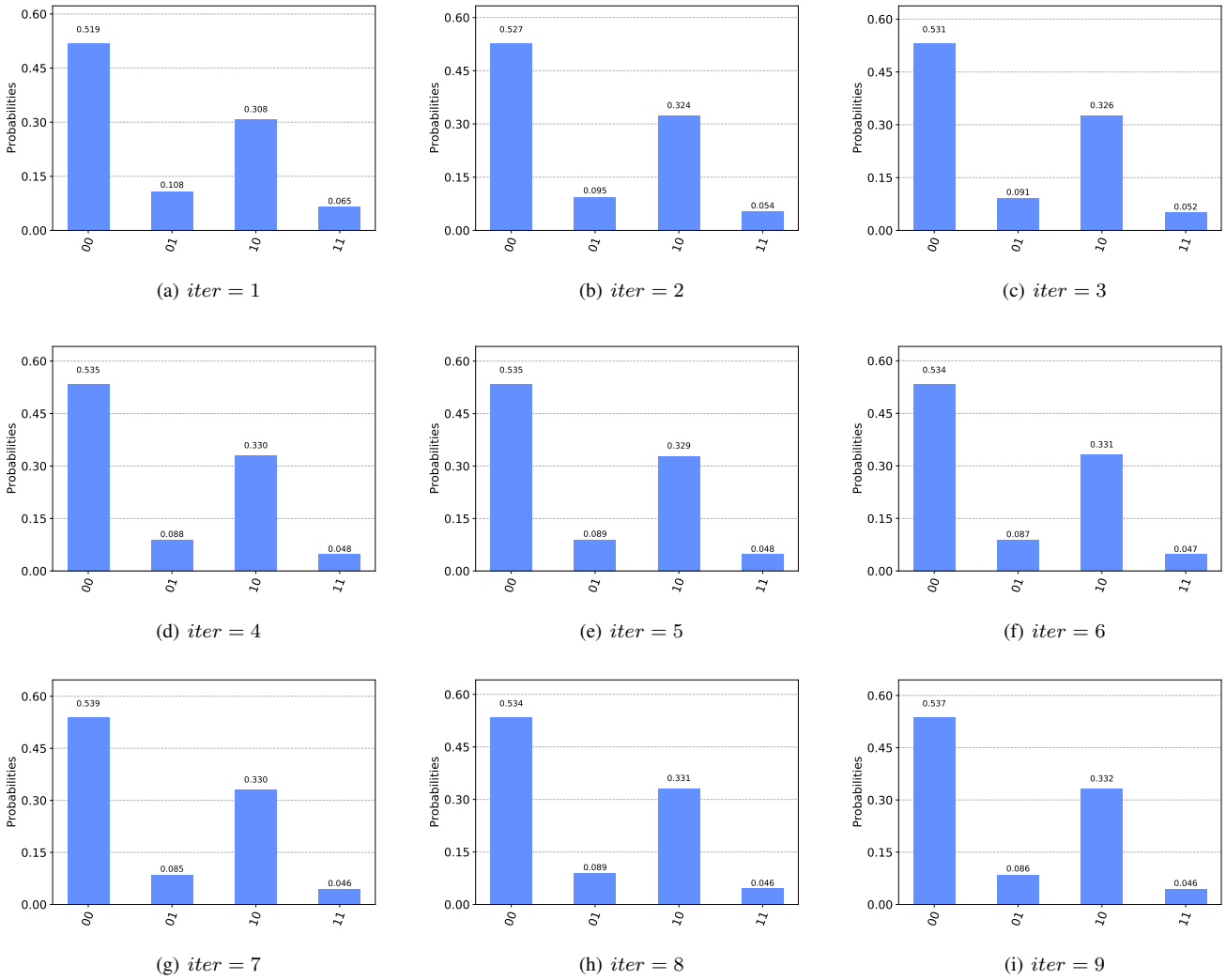
Fig. 7. Quantum circuit for solving $|W\rangle$ in experiment.

4 Experiments

At present, IBM has announced the world's largest superconducting quantum computer, which contains 127 quantum bits. But, for big data, it is still not enough. In this paper, because the whole algorithm process is quantized, if n and d in a dataset $X \in R^{n \times d}$ are large, ordinary computers cannot run it. Therefore, the existing computer hardware capability limits the operation of the algorithm. Especially for big data oriented machine learning algorithms. Therefore, we use a small data set to verify that the proposed algorithm has good feature selection performance in IBM quantum platform and Matlab.

4.1 Experimental setting

In order to verify the validity of the proposed quantum feature selection algorithm, we first do some preliminary work. According to algorithm 1, we construct a training data $X = \begin{bmatrix} 1 & -\frac{1}{3} \\ -\frac{1}{3} & 1 \end{bmatrix}$



(a) $iter = 1$ (b) $iter = 2$ (c) $iter = 3$

(d) $iter = 4$ (e) $iter = 5$ (f) $iter = 6$

(g) $iter = 7$ (h) $iter = 8$ (i) $iter = 9$

Fig. 8. Measurement results of $|\Theta_1\rangle$ at each iteration.

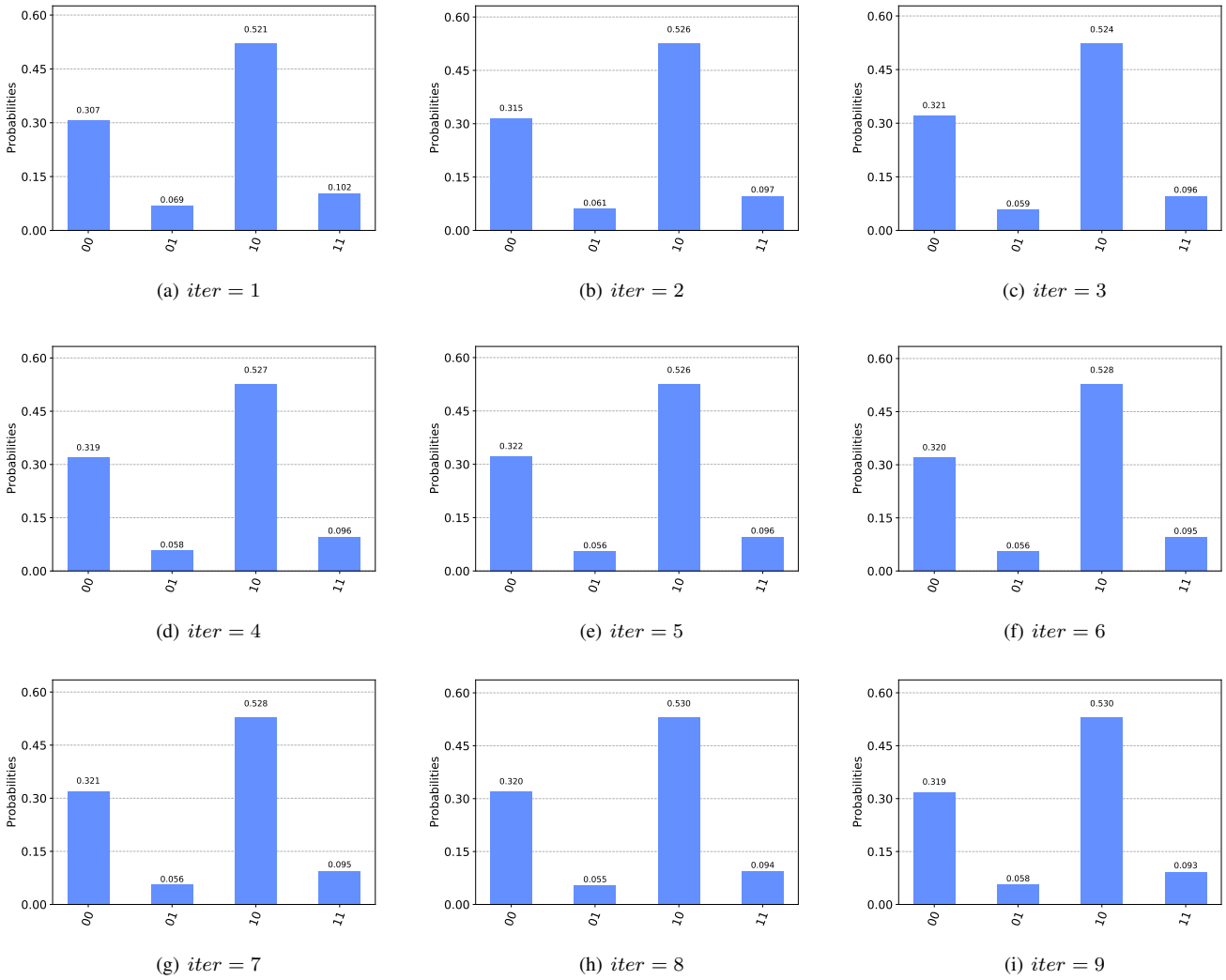
TABLE 3
Value of W and Θ in each iteration.

iteration	1	2	3	4	5	6	7	8	9
W	0.3615 0.4973	0.3758 0.9058	0.4605 1.1868	0.5551 1.3291	0.6307 1.4033	0.6811 1.4452	0.7116 1.4694	0.7289 1.4831	0.7385 1.4908
Θ1	0.3440 -0.4180	0.3699 -0.3739	0.3913 -0.3547	0.4047 -0.3441	0.4126 -0.3380	0.4172 -0.3344	0.4199 -0.3323	0.4214 -0.3311	0.4223 -0.3303
Θ2	-0.2035 0.6197	-0.2267 0.5592	-0.2426 0.5354	-0.2525 0.5226	-0.2583 0.5151	-0.2618 0.5107	-0.2638 0.5080	-0.2649 0.5065	-0.2656 0.5056

limited by quantum bits, corresponding class label $y = [1; 2]$, and adjustable parameter $\alpha = \beta = 1$. Based on these data information, we need to obtain some other auxiliary information needed in the algorithm process. We can easily get that the eigenvalues of X are $\lambda_1 = \frac{2}{3}$ and $\lambda_2 = \frac{4}{3}$, and the corresponding eigenvectors are $u_1 = \begin{pmatrix} \frac{-1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} \end{pmatrix}$ and $u_2 = \begin{pmatrix} \frac{-1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix}$. Then, we can use $n_0 = 2$ quantum bits to encode the eigenvalues, namely $|01\rangle$ and $|10\rangle$. Since $|01\rangle = |1\rangle = |N\lambda_1 t/2\pi\rangle$ and $|10\rangle = |2\rangle = |N\lambda_2 t/2\pi\rangle$, where $N = 2^{n_0} = 4$, we set $t = \frac{3}{4}\pi$. After obtaining the values of these parameters, we set our experiment as follows:

1. Using the proposed quantum circuit to run on IBM quantum platform, calculate the quantum state $|W\rangle$ of feature selection variable W and the quantum state $|\Theta\rangle$ of sample weight Θ under each iteration. In addition, measure the quantum registers of auxiliary bits, $|y\rangle$ and $|u\rangle$ to obtain the measurement ratio of each iteration.

2. Run the quantum circuit diagram of the last iteration of the proposed algorithm on the IBM quantum platform, and obtain the controlled rotation angle required by different variables in each



(a) $iter = 1$ (b) $iter = 2$ (c) $iter = 3$

(d) $iter = 4$ (e) $iter = 5$ (f) $iter = 6$

(g) $iter = 7$ (h) $iter = 8$ (i) $iter = 9$

Fig. 9. Measurement results of $|\Theta_2\rangle$ at each iteration.

TABLE 4
The angle of rotation of auxiliary bits for different variables in each iteration.

iteration	1	2	3	4	5	6	7	8	9
$W1$	1.2342	0.8661	0.8869	1.0001	1.1080	1.1834	1.2293	1.2555	1.2699
$W2$	0.6040	0.8221	0.9098	0.9388	0.9493	0.9540	0.9564	0.9578	0.9585
$\Theta_1 1$	0.8763	0.8396	0.8784	0.9090	0.9276	0.9383	0.9445	0.9480	0.9501
$\Theta_1 2$	0.8597	0.7902	0.7682	0.7574	0.7511	0.7473	0.7450	0.7437	0.7429
$\Theta_2 1$	0.6905	0.7251	0.7611	0.7843	0.7980	0.8059	0.8104	0.8131	0.8146
$\Theta_2 2$	0.8555	0.8225	0.8098	0.8029	0.7988	0.7963	0.7949	0.7940	0.7935

iteration.

3. The convergence experiment of the proposed algorithm is carried out on Matlab software to obtain the results of feature selection variable W and sample weight Θ of the final iteration, and compare them with the results of quantum computation. In addition, the robustness of the algorithm is also tested.

It should be noted that our algorithm has reached convergence in the first 9 iterations. Therefore, we only show the experimental results of the first 9 iterations in part of the experimental results.

4.2 Analysis of experimental results

Fig. 6 shows the quantum results of the first 9 iterations of the feature selection variable W. Because we only measure two quantum bits, and only when the auxiliary measurement bit is $|1\rangle$, can we obtain the quantum state $|W\rangle$. Therefore, we need to observe the values of $|0\rangle\,|1\rangle$ and $|1\rangle\,|1\rangle$. From Fig. 6, we can see that in the first iteration, the result is $0.102 : 0.212 = 1 : 2.0784$. In the 2nd to 9th iteration, the result obtained is between $1 : 3.5 \pm 0.7$. This

TABLE 5
The values of W and Θ in each iteration after adding gaussian white noise.

iteration	1	2	3	4	5	6	7	8	9
W	0.5950 1.0984	0.5950 1.2118	0.6098 1.2434	0.6237 1.2551	0.6345 1.2614	0.6424 1.2656	0.6481 1.2685	0.6522 1.2706	0.6551 1.2721
Θ1	0.1073 -0.4885	0.0616 -0.4914	0.0377 -0.5013	0.0236 -0.5080	0.0150 -0.5122	0.0097 -0.5148	0.0062 -0.5164	0.0040 -0.5174	0.0026 -0.5181
Θ2	-0.0988 0.9738	-0.0582 0.9630	-0.0355 0.9745	-0.0221 0.9829	-0.0141 0.9881	-0.0090 0.9913	-0.0058 0.9934	-0.0038 0.9946	-0.0024 0.9955

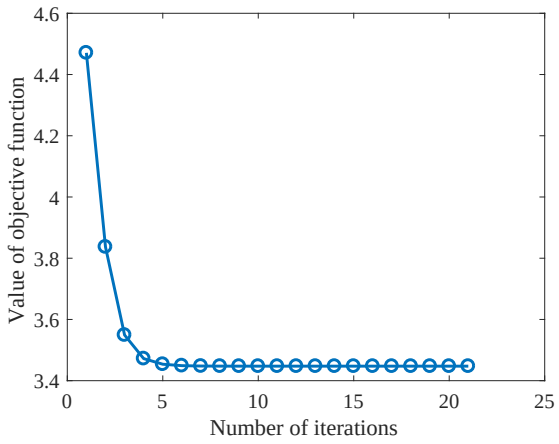


Fig. 11. The value of the objective function in each iteration after adding gaussian white noise.

shows that the weight of the first feature is obviously less than the weight of the second feature, and the final feature selection variable selects the second feature.

Table 3 also shows the final value of W obtained by running the proposed algorithm in matlab. From Table 3, we can find that in each iteration, the results of quantum computation are basically consistent with the results of non quantum operation, and the second feature is given a larger weight. Unfortunately, the specific value of the results is still a little biased. *i.e.,* the final result in Matlab is $|0.7385|^2 : |1.4908|^2 = 1 : 4.0751$. The result of quantum circuit operation is 0.088:0.252=1:2.8636. The reason for this phenomenon is the noise in the quantum circuit.

Similarly, Figs. 8 and 9 show the quantum results of the variable Θ in the first nine iterations respectively. It is different from the quantum circuit for solving W. When solving each column of Θ, the variables to be fitted are different. Therefore, the results of column 1 and column 2 of Θ (*i.e.,* Figs. 8 and 9) are different. In addition, the controlled rotation angle of auxiliary bits in each iteration is also different. From Figs. 8 and 9, we can see that in most iterations, the obtained results have little change, *i.e.,* the ratio of $|0\rangle|1\rangle$ to $|1\rangle|1\rangle$ has little change. This shows that in the proposed quantum algorithm, the quantum state of Θ is relatively stable in each iteration.

4.2.1 Controlled rotational variation of auxiliary bits

From Eqs. (23) and (24), we can see that the angle of auxiliary bit rotation is different for each iteration. It is equivalent to $RY(\eta)$ rotation, as shown in the following formula:

$$RY(\eta) = \begin{pmatrix} \cos(\frac{\eta}{2}) & -\sin(\frac{\eta}{2}) \\ \sin(\frac{\eta}{2}) & \cos(\frac{\eta}{2}) \end{pmatrix} \tag{29}$$

where $\eta = 2\arcsin(\frac{\lambda_i}{\lambda_i^2+\alpha N_{ii}})$ or $\eta = 2\arcsin(\frac{\lambda_i}{\lambda_i^2+\beta M_{ii}})$, they correspond to variables W and Θ respectively. On the one hand, we need to solve two variables W and Θ, and Θ has two columns (only one column of quantum state is output at a time). On the other hand, the data has two eigenvalues. Therefore, in each iteration, we need to calculate the controlled rotation angle of six auxiliary bits. Their rotation angles are shown in Table 4. From Table 4, we can see that in the first five iterations, the rotation angle η changes greatly. After the fifth time, the rotation angle changes little. For example, for variable W, after the fifth iteration, the two rotations are concentrated on 1.2 ± 0.03 and 0.95 ± 0.01. For the first column of Θ, after the fifth iteration, the angles of the two rotations are concentrated at 0.94 ± 0.02 and 0.74 ± 0.01. For the second column of Θ, after the fifth iteration, the angles of the two rotations are concentrated at 0.81 ± 0.01 and 0.79 ± 0.007. From these results, we can also simply see that the final results of the algorithm have almost no change since the sixth iteration.

4.2.2 Convergence and quantum circuit

Fig. 10 shows the change of the objective function value of the proposed algorithm with the number of iterations. From Fig. 10, we can find that the proposed algorithm is convergent, and the convergence speed is fast. In addition, we also show the quantum circuit diagram for specific data, because the solution of W is similar to Θ. *i.e.,* the middle fourier technology is the same, but the difference lies in the initial state setting and the angle of controlled rotation of auxiliary bits. Therefore, we only show the quantum circuit diagram for solving the feature selection variable W. As shown in Fig. 7, we use one qubit to encode label y, two qubits to encode eigenvalues, and one auxiliary qubit. It should be noted that the angle of controlled rotation of auxiliary bits is related to Eqs. (25) and (26). From Fig. 7, we can see that after phase estimation and controlled rotation, when the auxiliary bit is measured to obtain $|1\rangle$, the quantum state $|W\rangle$ can be obtained.

4.2.3 Robustness analysis

Different from the existing quantum feature selection algorithm, the proposed quantum feature selection algorithm is robust. Specifically, we compared the characteristics of the proposed algorithm and other quantum feature selection algorithms, as shown in Table 6. From Table 6, we can see that after the parameters are initialized, the proposed quantum feature selection is quantized in the whole process of learning the quantum state of the feature weight. This is different from other quantum feature selection

TABLE 6
Characteristics comparison of quantum feature selection algorithms.

method	Quantization of all steps	Whether there is quantum circuit	Exponential acceleration or not	Square acceleration or not	Whether dimension can be reduced	Robustness
Chakraborty *et al.* [40]	×	✓	×	✓	✓	×
Wang *et al.* [41]	×	×	×	×	✓	×
Otgonbaatar and Datcu [42]	×	×	×	×	✓	×
Desu *et al.* [9]	×	×	×	×	✓	×
Li *et al.* [43]	×	✓	✓	×	✓	×
Liu *et al.* [12]	×	✓	×	✓	✓	×
Proposed	✓	✓	✓	×	✓	✓

algorithms, which are only partially quantized. In addition, the proposed algorithm can achieve exponential acceleration when the dataset is low rank, and it is robust.

In Table 5, we show the values of feature selection variables W and Θ obtained by the algorithm in each iteration after adding gaussian white noise to the data. We can find that, compared with Table 3, the values of W and Θ have changed after adding gaussian white noise. However, the final feature selection result is not affected. From the value of each iteration of feature selection variable W, we can find that it does not change significantly. Compared with the experiment without gaussian white noise, the final value of W obtained is still about $1:2$, which indicates that the weight of the second feature is greater than that of the first feature and the weight ratio is almost unchanged.. In addition, we also performed the convergence experiment of the algorithm after adding gaussian white noise, as shown in Fig. 11. From Fig. 11, we can see that the algorithm still converges around the fifth iteration. All these results show that the proposed algorithm has good robustness.

5 Conclusion

In this paper, a robust quantum feature selection algorithm has been proposed. Specifically, it first prepares the initial quantum state of the whole system through quantum technology. Then it uses quantum phase estimation technology to obtain the eigenvalues and eigenvectors corresponding for the data matrix. Finally, it uses the quantum controlled rotation technology to combine the eigenvalues on the amplitude of the auxiliary bit to the $|y\rangle$ bit. When the auxiliary bit is measured to a specific state, the final quantum state of the feature selection variable is obtained. In the experiment, the proposed algorithm is implemented by using the quantum circuit programmed with qiskit, and the experiment is carried out on IBM quantum platform.

In the future work, we plan to study more quantum machine learning algorithms. For example, Quantum spectral clustering, quantum ensemble learning and quantum random forest.

Acknowledgment

This work has been supported in part by the Natural Science Foundation of China under grant 61836016. Foundation of China (Grant Nos. 61972418, Nos. 62272483), the Special Foundation for Distinguished Young Scientists of Changsha (Grant Nos. kq1905058)

References

[1] M. Cerezo, G. Verdon, H.-Y. Huang, L. Cincio, and P. J. Coles, "Challenges and opportunities in quantum machine learning," *Nature Computational Science*, vol. 2, no. 9, pp. 567–576, 2022.

[2] M. C. Caro, H.-Y. Huang, M. Cerezo, K. Sharma, A. Sornborger, L. Cincio, and P. J. Coles, "Generalization in quantum machine learning from few training data," *Nature communications*, vol. 13, no. 1, pp. 1–11, 2022.

[3] R. Zhang, J. Wang, N. Jiang, H. Li, and Z. Wang, "Quantum support vector machine based on regularized newton method," *Neural Networks*, vol. 151, pp. 376–384, 2022.

[4] J. Li, S. Lin, K. Yu, and G. Guo, "Quantum k-nearest neighbor classification algorithm based on hamming distance," *Quantum Information Processing*, vol. 21, no. 1, pp. 1–17, 2022.

[5] A. Basheer, A. Afham, and S. K. Goyal, "Quantum k-nearest neighbors algorithm," *arXiv preprint arXiv:2003.09187*, 2020.

[6] C.-H. Yu, F. Gao, and Q.-Y. Wen, "An improved quantum algorithm for ridge regression," *IEEE Transactions on Knowledge and Data Engineering*, vol. 33, no. 3, pp. 858–866, 2019.

[7] S. Lloyd, M. Mohseni, and P. Rebentrost, "Quantum principal component analysis," *Nature Physics*, vol. 10, no. 9, pp. 631–633, 2014.

[8] Y. Li, R.-G. Zhou, R. Xu, W. Hu, and P. Fan, "Quantum algorithm for the nonlinear dimensionality reduction with arbitrary kernel," *Quantum Science and Technology*, vol. 6, no. 1, p. 014001, 2020.

[9] S. S. T. Desu, P. Srijith, M. Rao, and N. Sivadasan, "Adiabatic quantum feature selection for sparse linear regression," in *International Conference on Computational Science*. Springer, 2021, pp. 98–112.

[10] R. Nembrini, M. Ferrari Dacrema, and P. Cremonesi, "Feature selection for recommender systems with quantum computing," *Entropy*, vol. 23, no. 8, p. 970, 2021.

[11] R. Agrawal, B. Kaur, and S. Sharma, "Quantum based whale optimization algorithm for wrapper feature selection," *Applied Soft Computing*, vol. 89, p. 106092, 2020.

[12] W.-J. Liu, P.-P. Gao, W.-B. Yu, Z.-G. Qu, and C.-N. Yang, "Quantum relief algorithm," *Quantum Information Processing*, vol. 17, no. 10, pp. 1–15, 2018.

[13] J. Bian, D. Zhao, F. Nie, R. Wang, and X. Li, "Robust and sparse principal component analysis with adaptive loss minimization for feature selection," *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[14] A. W. Harrow, A. Hassidim, and S. Lloyd, "Quantum algorithm for linear systems of equations," *Physical review letters*, vol. 103, no. 15, p. 150502, 2009.

[15] J. Wen, X. Kong, S. Wei, B. Wang, T. Xin, and G. Long, "Experimental realization of quantum algorithms for a linear system inspired by adiabatic quantum computing," *Physical Review A*, vol. 99, no. 1, p. 012320, 2019.

[16] V. Dixit and S. Jian, "Quantum fourier transform to estimate drive cycles," *Scientific Reports*, vol. 12, no. 1, pp. 1–10, 2022.

[17] Y. Mukai, R. Okamoto, and S. Takeuchi, "Quantum fourier-transform infrared spectroscopy in the fingerprint region," *Optics Express*, vol. 30, no. 13, pp. 22 624–22 636, 2022.

[18] J. Shi, W. Wang, X. Lou, S. Zhang, and X. Li, "Parameterized hamiltonian learning with quantum circuit," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–10, 2022.

[19] H. Zheng, X. Ren, P. Liu, and G. Jin, "Quantum phase estimation with a general binary-outcome measurement," *Results in Physics*, p. 106051, 2022.

[20] C. Kang, N. P. Bauman, S. Krishnamoorthy, and K. Kowalski, "Optimized quantum phase estimation for simulating electronic states in various energy regimes," *arXiv preprint arXiv:2206.00802*, 2022.

[21] J. Shi, Y. Tang, Y. Lu, Y. Feng, R. Shi, and S. Zhang, "Quantum circuit learning with parameterized boson sampling," *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–1, 2021.

[22] H. Y. Wong, "Quantum phase estimation," in *Introduction to Quantum Computing*. Springer, 2022, pp. 267–277.

[23] I. Jolliffe, "A 50-year personal journey through time with principal component analysis," *Journal of Multivariate Analysis*, vol. 188, p. 104820, 2022.

[24] F. Zhu, J. Gao, J. Yang, and N. Ye, "Neighborhood linear discriminant analysis," *Pattern Recognition*, vol. 123, p. 108422, 2022.

[25] B. Wang, Y. Sun, Y. Chu, Z. Yang, and H. Lin, "Global-locality preserving projection for word embedding," *International Journal of Machine Learning and Cybernetics*, vol. 13, no. 10, pp. 2943–2956, 2022.

[26] X. Yang, Y. Chen, X. Yue, C. Ma, and P. Yang, "Local linear embedding based interpolation neural network in pancreatic tumor segmentation," *Applied Intelligence*, vol. 52, no. 8, pp. 8746–8756, 2022.

[27] J. Li, S. Zhang, L. Zhang, C. Lei, and J. Zhang, "Unsupervised nonlinear feature selection algorithm via kernel function," *Neural Computing and Applications*, vol. 32, no. 11, pp. 6443–6454, 2020.

[28] C. Cavicchia, M. Vichi, and G. Zaccaria, "Hierarchical disjoint principal component analysis," *AStA Advances in Statistical Analysis*, pp. 1–38, 2022.

[29] F. Nie, D. Wu, R. Wang, and X. Li, "Truncated robust principle component analysis with a general optimization framework," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 2, pp. 1081–1097, 2022.

[30] Y. Xu, M. Fu, Q. Wang, Y. Wang, K. Chen, G.-S. Xia, and X. Bai, "Gliding vertex on the horizontal bounding box for multi-oriented object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 4, pp. 1452–1459, 2021.

[31] J. Lin, W.-S. Bao, S. Zhang, T. Li, and X. Wang, "An improved quantum principal component analysis algorithm based on the quantum singular threshold method," *Physics Letters A*, vol. 383, no. 24, pp. 2862–2868, 2019.

[32] A. Daskin, "Obtaining a linear combination of the principal components of a matrix on quantum computers," *Quantum Information Processing*, vol. 15, no. 10, pp. 4013–4027, 2016.

[33] C.-H. Yu, F. Gao, S. Lin, and J. Wang, "Quantum data compression by principal component analysis," *Quantum Information Processing*, vol. 18, no. 8, pp. 1–20, 2019.

[34] O. O. Akinola, A. E. Ezugwu, J. O. Agushaka, R. A. Zitar, and L. Abualigah, "Multiclass feature selection with metaheuristic optimization algorithms: a review," *Neural Computing and Applications*, pp. 1–40, 2022.

[35] S. Zhang and J. Li, "Knn classification with one-step computation," *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–1, doi: 10.1109/TKDE.2021.3 119 140, 2021.

[36] S. Zhang, J. Li, and Y. Li, "Reachable distance function for knn classification," *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–15, doi: 10.1109/TKDE.2022.3 185 149, 2022.

[37] V. Atzrodt and H. Lange, "Principal component analysis as a method for silicide investigation with auger electron spectroscopy," in *Volume 79, Number 2 October 16*. De Gruyter, 2022, pp. 489–497.

[38] M. Grossi, N. Ibrahim, V. Radescu, R. Loredo, K. Voigt, C. Von Altrock, and A. Rudnik, "Mixed quantum-classical method for fraud detection with quantum feature selection," *IEEE Transactions on Quantum Engineering*, 2022.

[39] H. Chen and B. Zou, "Optimal feature selection algorithm based on quantum-inspired clone genetic strategy in text categorization," in *Proceedings of the first ACM/SIGEVO Summit on Genetic and Evolutionary Computation*, 2009, pp. 799–802.

[40] S. Chakraborty, S. H. Shaikh, A. Chakrabarti, and R. Ghosh, "A hybrid quantum feature selection algorithm using a quantum inspired graph theoretic approach," *Applied Intelligence*, vol. 50, no. 6, pp. 1775–1793, 2020.

[41] D. Wang, H. Chen, T. Li, J. Wan, and Y. Huang, "A novel quantum grasshopper optimization algorithm for feature selection," *International Journal of Approximate Reasoning*, vol. 127, pp. 33–53, 2020.

[42] S. Otgonbaatar and M. Datcu, "A quantum annealer for subset feature selection and the classification of hyperspectral images," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 7057–7065, 2021.

[43] Y. Li, R.-G. Zhou, R. Xu, J. Luo, W. Hu, and P. Fan, "Implementing graph-theoretic feature selection by quantum approximate optimization algorithm," *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

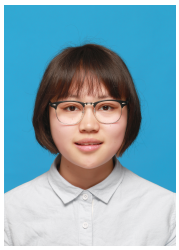
[44] X. Li, H. Zhang, R. Wang, and F. Nie, "Multiview clustering: A scalable and parameter-free bipartite graph fusion method," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 1, pp. 330–344, 2022.

[45] Z. Zhu, T. Huang, M. Xu, B. Shi, W. Cheng, and X. Bai, "Progressive and aligned pose attention transfer for person image generation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 8, pp. 4306–4320, 2022.

[46] P. Rebentrost, A. Steffens, I. Marvian, and S. Lloyd, "Quantum singular-value decomposition of nonsparse low-rank matrices," *Physical review A*, vol. 97, no. 1, p. 012327, 2018.

Shichao Zhang is a China National Distinguished Professor with the Central South University, China. He holds a PhD degree from the Deakin University, Australia. His research interests include data mining and big data. He has published 90 international journal papers and over 70 international conference papers. He is a CI for 18 competitive national grants. He serves/served as an associate editor for four journals.

Jiaye Li is currently working toward the PhD degree at Central South University, China. His research interests include quantum machine learning, data mining and deep learning.



Hang Xu is currently working toward the PhD degree at Central South University, China. His research interests include quantum machine learning, data mining and deep learning.

Hao Yu is currently working toward the PhD degree at Central South University, China. His research interests include quantum machine learning, data mining and crowdsourcing.

Jinjing Shi received the BS and PhD degrees from the School of Information Science and Engineering, Central South University, Changsha, China, in 2008 and 2013, respectively. She is now an associate professor with the School of Computer Science and Engineering, Central South University, China. She was selected in the Shenghua lieying talent program of Central South University, China, and Special Foundation for Distinguished Young Scientists of Changsha, in 2013 and 2019, respectively. Her research interests include quantum computation and quantum cryptography. She has presided over the National Natural Science Foundation Project of China and that of Hunan Province. There are 50 academic papers published in important international academic journals and conferences. She has received the second prize of natural science and the outstanding doctoral dissertation of Hunan Province, in 2015, and she has received the Best Paper Award in the International Academic Conference MSPT2011 and Outstanding Paper Award in IEEE ICACT2012.