An Exact Quantum Principal Component Analysis Algorithm Based on Quantum Singular Value Threshold

Chen He\textsuperscript{1} Jiazhen Li\textsuperscript{2} and Weiqi Liu\textsuperscript{3}
(Dated: October 5, 2020)

Quantum principal component analysis (qPCA) is the quantum version of principal component analysis (PCA). In this paper, based on the quantum singular value threshold (qSVT), we propose an exact quantum principal component analysis algorithm, which screens the data components through the threshold, rather than output all components of data. Compared with other improved qPCA algorithms, our proposed algorithm does not require to adjust the parameters to obtain estimated results. Instead, it yields exact results directly, and the quantum circuit designed is simpler because almost half of the quantum gates are reduced. We implemented our qPCA algorithm on the IBM quantum computing platform: IBM Quantum Experience, and the experimental results verified correctness of our algorithm.

I. INTRODUCTION

Principal component analysis (PCA) \cite{1,2}, a dimension reduction method \cite{3,4}, is a very important algorithm for signal processing and machine learning. However, the time complexity of PCA is $O(N^3)$, where $N$ is the dimension of the data. It is clear that when the dimension is large, the classical PCA is not tractable. Because of the quantum computer’s parallelism \cite{5}, quantum PCA (qPCA) can reduce the complexity to $O(Np\log(N))$ \cite{6,7}. The qPCA outputs all eigenvalues and then obtains the top-r eigenvalues by sampling, its core algorithm is the phase estimation eigendecomposition \cite{8} which can also be applied to quantum singular value threshold (qSVT) \cite{9}, quantum singular value decomposition (qSVD) \cite{10}.

Recent work has shown that qPCA \cite{11} can extract all the eigenvalues of the data. Suppose a matrix $A_0 \in \mathbb{R}^{p \times q}$ with singular value decomposition

$$
|\psi_{A_0}\rangle = \sum_{k=1}^{T} \sigma_k |u_k\rangle |v_k\rangle ,
$$

where $T$ is the rank of matrix $A_0$, $\sigma_k$ are the singular values, $u_k, v_k$ are left and right singular vectors. Then we define $A = A_0A_0^\dagger$ with decomposition

$$
|\psi_A\rangle = \sum_{k=1}^{T} \lambda_k |u_k\rangle |u_k\rangle ,
$$

where $A_0^\dagger$ is the conjugate transpose of matrix $A_0$, $\lambda_k = \sigma_k^2$ are the eigenvalues of the $A$. The qPCA \cite{11} showed that phase estimation can be used to extract all eigenvalues of the matrix $A$ in the form of

$$
|\psi_0\rangle = \sum_{k=1}^{T} \lambda_k |u_k\rangle \langle u_k| \otimes |\lambda_k\rangle \langle \lambda_k| .
$$

The qSVT \cite{11} showed that the matrix $A_0$ can be decomposed into the form of

$$
|\psi_S\rangle = \sum_{k=1}^{r} (\sigma_k - \tau) \sigma_k |u_k\rangle |v_k\rangle ,
$$

where $\tau$ is the threshold and $(\sigma_k - \tau)_+ = max(0, \sigma_k - \tau)$. The $|\psi_S\rangle$ can be estimated by qSVT as

$$
|\psi'_S\rangle = \sum_{k=1}^{r} \sigma_k \sin(y_k \alpha) |u_k\rangle |v_k\rangle ,
$$

where $y_k = (1 - \frac{\tau}{\sigma_k})_+ \in (0, 1], \alpha$ is the parameter of controlled rotation operation $R_{\alpha}(\alpha) \ [11]$ in Fig. 2, and the parameter can be adjusted to improve the success probability and fidelity of the algorithm \cite{11}. Apparently, compared with qPCA algorithm, a threshold $\tau$ is required in qSVT. After that an improved qPCA based on qSVT \cite{13} was proposed and showed that the eigenvalues greater than the threshold can be extracted, the ideal improved qPCA should yield

$$
|\widetilde{\psi}_{A_0}\rangle = \sum_{k=1}^{r} \sigma_k |u_k\rangle |v_k\rangle ,
$$

which can be estimated by

$$
|\psi'_{A_0}\rangle = \sum_{k=1}^{r} \sigma_k (\sin(y_k \alpha) + 2 \sin(y_k \alpha) \sin(y_k \alpha) \frac{y_k}{2}) |u_k\rangle |v_k\rangle ,
$$

$$
\approx \sum_{k=1}^{r} \sigma_k |u_k\rangle |v_k\rangle ,
$$

where $y_k = 1 - \frac{r}{\sigma_k}, y' = 1 + \frac{r}{\sigma_k}$. The eigenvalues $|\lambda_k\rangle$ of $A$ in improved qPCA can be obtained by

$$
|\psi_{A_1}\rangle = \sum_{k=1}^{r} \sigma_k |\lambda_k\rangle |u_k\rangle |v_k\rangle ,
$$

where $\lambda_k = \sigma_k^2$.

Clear, although the principal eigenvalues are further filtered and extracted through the threshold in the improved qPCA \cite{13}, there are still two concerns. One is
that it needs to continuously adjust the parameters $\alpha$ to get the closest estimate, the other is that the quantum circuit of the algorithm is complicated.

In this paper, we propose an exact qPCA algorithm based on qSVT. We replace the controlled rotation operation $R_y(\alpha)$ of qSVT with another controlled operation to generate qPCA algorithm. Compared with previous improved qPCA [13], which obtains $|\psi_{A_1}\rangle$ via its estimation $|\psi_{A_0}\rangle$, our algorithm, however, can yield the exact principal eigenvalues directly in the form of

$$\left|\psi'_{A_1}\right\rangle = \sum_{k=1}^{r} \lambda_k |u_k\rangle |u_k\rangle, \quad (10)$$

$$\left|\psi'_{A_0}\right\rangle = \sum_{k=1}^{r} |\lambda_k\rangle |u_k\rangle |u_k\rangle, \quad (11)$$

where $r < T, \lambda_1 < \lambda_2 < \ldots < \lambda_k < \ldots < \lambda_T$. Unlike the algorithm in [13], where we need to adjust the parameter $\alpha$, our algorithm does not involve any parameter adjustment, therefore the eigendecomposition yielded by our algorithm is the exact result, instead of an estimated one, i.e., it is more precise. Moreover, the quantum circuit of our algorithm is considerably simplified as it requires much less quantum gates.

The paper is organized as follows: In Section II, we briefly review the previous qSVT-based qPCA algorithm in [13], then we propose the exact qSVT-based qPCA algorithm. In Section III, we implement our qPCA algorithm on IBM Quantum Experience, and verify our algorithm based on the experimental results. Finally we conclude this work in Section IV.

II. EXACT QPCA ALGORITHM BASED ON QSVT

In this section, first, we provide a review on qSVT and the previous qSVT-based qPCA algorithm proposed in [13], then we propose the exact qSVT-based qPCA algorithm. The fundamental difference between the qPCA we proposed and the qPCA in [13] is that, the input is the covariance matrix $A$ instead of $A_0$, and the output is an exact value rather than an estimation. In addition, our algorithm requires much less quantum gates: the number of quantum gates required is almost reduced by half.

A. The previous qSVT-based qPCA algorithm

The qSVT-based qPCA algorithm in [13] consists of two major parts as shown in Fig. 1. The first part is the qSVT algorithm, and the second part is MqSVT algorithm [14].

The qSVT algorithm is illustrated in Procedure 2 of Algorithm 1, and the corresponding quantum circuit is shown in Fig. 2. The matrix $A_0$ can be represented in a quantum form of $|\psi_{A_0}\rangle$, whose SVT in quantum form is given by $|\psi_S\rangle$. To obtain qSVT of $A_0$, the first step is the phase estimation that yields singular values $|\sigma_k\rangle$. The second step is the unitary operation $U_{\lambda,T}$, which is mainly realized by Newton iteration [11] [15], mapping the singular values $|\sigma_k\rangle$ to the $|y_k\rangle = |1 - \frac{T}{\sigma_T}\rangle$. Consequently, $|\lambda_k\rangle, |\tau\rangle, |y_k\rangle$ are saved in Reg. A, B, and C, respectively. The third step is performing the controlled rotation $R_y(\alpha)$, which extracts $|y_k\rangle$ to the probability amplitude of the top ancillary qubit. The fourth step is resetting Reg. A and C by inverse unitary operations. The last step is measurement. If the measurement result of the top ancillary qubit is $|1\rangle$, the final quantum state of qSVT will collapse to $|\psi_S\rangle$, an estimation of $|\psi_S\rangle$.

MqSVT is very similar to qSVT. Compared with qSVT, MqSVT replaces the parameters $y_k$ by $y'_k$ and $\alpha$ by $\frac{T}{y'_{\lambda}}$ as a result, the output of MqSVT is given by

$$|\psi'_{A_0}\rangle = \sum_{k=1}^{r} \sigma_k \sin \left(\frac{y'_k \alpha}{2}\right) |u_k\rangle |v_k\rangle. \quad (12)$$

As shown in Fig. 4 if the control bit is set to 0, the qSVT is performed, otherwise, the MqSVT is performed. When the top two qubits project on $|+\rangle |1\rangle$, the final quantum state will collapse to $|\psi'_{A_0}\rangle$, an estimation of $|\psi_{A_0}\rangle$, where the singular values are greater than the threshold $\tau$. To extract $|\lambda_k\rangle$ of the covariance matrix $A$, we perform the phase estimation to obtain $|\psi_{A_1}\rangle = \sum_{k=1}^{r} \sigma_k |\lambda_k\rangle |u_k\rangle |v_k\rangle$, and consequently $|\lambda_k\rangle$s can be obtained by measuring $|\psi_{A_1}\rangle$.

B. Our algorithm: an exact qSVT-based qPCA

Fig. 3 shows the quantum circuit of the exact qSVT-based qPCA we proposed. It is achieved by replacing a controlled operation of qSVT, and the procedure of the algorithm is shown in Algorithm 2. The input of our qPCA is the matrix $A$. The first step is preparing the quantum state $|\psi_A\rangle$. The second step is the phase estimation yielding eigenvalues $|\lambda_k\rangle$. The third step is converting $|\lambda_k\rangle$ to $|y_k\rangle = |1 - \frac{T}{\sigma_T}\rangle$ by the unitary operation $U_{\lambda,T}$. The fourth step, since our algorithm should yield $|\sigma_k\rangle$ instead of $|\sigma_k - \tau\rangle$, we employ the new controlled operation as shown in Fig. 3(a) instead of $R_y(\alpha)$ in Fig. 2. The fifth step is resetting the Reg. A, C by the inverse operation $U^*$. The sixth step is the measurement. If the measurement result of the top qubit is $|1\rangle$, the quantum state will collapse to $|\psi'_{A_1}\rangle$, whose eigenvalues $\lambda_k$ are all greater than the given threshold $\tau$. For the last step, by performing the phase estimation we can obtain $|\psi'_{A_1}\rangle = \sum_{k=1}^{r} \lambda_k |\lambda_k\rangle |u_k\rangle |v_k\rangle$, and consequently $|\lambda_k\rangle$s can be obtained by measuring $|\psi_{A_1}\rangle$. 

\[ \left|\psi'_{A_1}\right\rangle = \sum_{k=1}^{r} \lambda_k |\lambda_k\rangle |u_k\rangle |v_k\rangle, \quad (11) \]
Algorithm 1  The improved qPCA proposed in [13].

Input:
A quantum state $|\psi_{A_0}\rangle$;
A unitary operation $U_{PE} = e^{iA\tau}$, where $A = A_0A_0^+$;
A threshold constant $\tau$.

Output:
A quantum state $|\psi_{A_1}\rangle$.

Procedure:

1: Prepare quantum state $|\psi_1\rangle = (a|0\rangle + b|1\rangle)\sum_{k=1}^T \sigma_k \sin(y_k\alpha) |u_k\rangle |v_k\rangle$.

2: If the top control qubit is set to 0, the qSVT is performed, then obtain $|\psi_2\rangle = (a|0\rangle + b|1\rangle)\sum_{k=1}^T \sigma_k \sin(y_k\alpha) |u_k\rangle |v_k\rangle$.

The process of qSVT is as follows.

- Perform the phase estimation $U_{PE}$ to obtain $|\phi_1\rangle = |0\rangle |0\rangle \sum_{k=1}^T \sigma_k |\sigma_k^2\rangle |u_k\rangle |v_k\rangle$.
- Perform the unitary operation $U_{x,\tau}$ to obtain $|\phi_2\rangle = |0\rangle \sum_{k=1}^T \sigma_k |y_k\rangle |\sigma_k^2\rangle |u_k\rangle |v_k\rangle$.
- Perform the controlled rotation $R_y(\alpha)$ to obtain $|\phi_3\rangle = \sum_{k=1}^T \sigma_k |\sin(y_k\alpha)|1\rangle + \cos(y_k\alpha) |0\rangle |y_k\rangle |\sigma_k^2\rangle |u_k\rangle |v_k\rangle$.
- Inverse unitary operations to obtain $|\phi_4\rangle = \sum_{k=1}^T \sigma_k |\sin(y_k\alpha)|1\rangle + \cos(y_k\alpha) |0\rangle |u_k\rangle |v_k\rangle$.
- Measurement. When the measurement result of the top qubit is $|1\rangle$, the quantum state will collapse to $|\phi_4\rangle = |\psi_S\rangle = \sum_{k=1}^T \sigma_k |\sin(y_k\alpha)|u_k\rangle |v_k\rangle$.

3: If the top control qubit is 1, the MqSVT is performed, then obtain $|\psi_3\rangle = (a|0\rangle + b|1\rangle)\sum_{k=1}^T \sigma_k \sin\left(\frac{y_k\alpha}{2}\right) |u_k\rangle |v_k\rangle$.

4: Measurement. If the measurement result of the first two qubits is $|+\rangle |1\rangle$, the quantum state will collapse to $|\psi_{A_1}\rangle = \sum_{k=1}^T \sigma_k (\sin y_k\alpha + 2\sin \frac{y_k\alpha}{2}) |u_k\rangle |v_k\rangle$.

5: Phase estimation to obtain $|\psi_{A_1}\rangle = \sum_{k=1}^T \sigma_k |\lambda_k\rangle |u_k\rangle |v_k\rangle$.

Both our exact qPCA and the one in [13] extract the eigenvalues greater than the threshold from the covariance matrix $A$. The difference is that, for our algorithm, the input matrix is $A$, and the circuit is directly designed by changing a quantum gate on qSVT. Therefore comparing with the qPCA algorithm proposed in [13], the number of quantum gates of our algorithm is reduced by half, and the parameters of the overall unitary operations are also reduced. In addition, unlike the algorithm in [13], our algorithm does not involve any parameter adjustment, so it yields an exact result rather than an estimation.

III. EXPERIMENT

In this section, we perform experiments for our exact qPCA algorithm on the IBM quantum computing platform: IBM Quantum Experience [16-18], and verify our algorithm.

A. The experiment for the $2 \times 2$ matrix

First, we take the $2 \times 2$ matrix

$$A = \begin{bmatrix} 1.5 & 0.5 \\ 0.5 & 1.5 \end{bmatrix},$$

(a) The qSVT and MqSVT parts of the quantum circuit of the qSVT-based qPCA algorithm in [13].

(b) The phase estimation part, where the input $|\psi_{A_0}\rangle$ is the output of Fig. 1(a).

FIG. 1. The quantum circuit of the qSVT-based qPCA algorithm in [13].

FIG. 2. The quantum circuit of qSVT algorithm in [11].
Algorithm 2 The exact qSVT-based qPCA algorithm proposed in this paper.

Input:
A quantum state $|\psi_A\rangle$;
A unitary operation $U_{PE} = e^{iA_1}$;
A threshold constant $\tau$.

Output:
A quantum state $|\psi_A'\rangle$.

Procedure:
1. Prepare quantum state $|\psi_1\rangle = |0\rangle |0\rangle |0\rangle |\psi_A\rangle$.
2. Perform the phase estimation $U_{PE}$ to obtain
   $|\psi_2\rangle = |0\rangle |0\rangle \sum_{k=1}^N \lambda_k |\lambda_k\rangle |u_k\rangle |u_k\rangle$.
3. Perform the unitary operation $U_{\lambda, r}$ to obtain
   $|\psi_3\rangle = |0\rangle \sum_{k=1}^N \lambda_k \sqrt{y_k} |\lambda_k\rangle |u_k\rangle |u_k\rangle$.
4. Perform the controlled operation as shown in Fig. 3(a) to obtain
   $|\psi_4\rangle = \sum_{k=1}^N \lambda_k |1\rangle |\lambda_k\rangle |u_k\rangle |u_k\rangle + \sum_{k=r+1}^{N} \lambda_k |0\rangle |\lambda_k\rangle |u_k\rangle |u_k\rangle$.
5. Inverse unitary operation to obtain
   $|\psi_5\rangle = \sum_{k=1}^N \lambda_k \sqrt{y_k} |\lambda_k\rangle |u_k\rangle |u_k\rangle + \sum_{k=r+1}^{N} \lambda_k |0\rangle |\lambda_k\rangle |u_k\rangle |u_k\rangle$.
6. Measurement. When the measurement result of the top qubit is $|1\rangle$, the quantum state will collapse to
   $|\psi_6\rangle = \sum_{k=1}^N \lambda_k |u_k\rangle |u_k\rangle$.
7. To extract the eigenvalues $|\lambda_k\rangle$, perform the phase estimation to get
   $|\psi_A'\rangle = \sum_{k=1}^N \lambda_k |\lambda_k\rangle |u_k\rangle |u_k\rangle$.

as an example. The corresponding initial quantum state $|\psi_A\rangle$ is given by

$$|\psi_A\rangle = [0.6708, 0.2236, 0.2236, 0.6708]. \quad (14)$$

We set the threshold $\tau = 1$, and the implementation of the quantum circuit of our qPCA algorithm on the IBM Quantum Experience is shown in Fig. 3. Five qubits are required in total. The first qubit $q[0]$ is used as an ancillary qubit. The second to third qubits $q[1-2]$ are used to save eigenvalues $|\lambda_k\rangle$ and $|y_k\rangle$, and the qubits $q[3-4]$ are used to initialize the quantum state $|\psi_A\rangle$. When the measurement result of $q[0]$ is $|1\rangle$, $q[3-4]$ will collapse into the quantum state $|\psi_A'\rangle$.

Circuit Composer on IBM Quantum Experience lets us see how quantum circuits affect the state of a collection of qubits through the measurement probabilities visualizations. As shown in Fig. 5(a), the result of the visualization is given by

$$|\psi_1\rangle = [0.5, 0.5, 0.5, 0.5]. \quad (15)$$

The quantum simulator simulates the execution of quantum circuits and returns counts in histogram, then we run the quantum circuit on the qasm simulator, and the result as shown in Fig. 5(b) is given by

$$|\psi_2\rangle = [0.4859, 0.5245, 0.5143, 0.4735]. \quad (16)$$

The accuracy of our quantum algorithm can be evaluated by the inner product, i.e.

$$f_1 = \langle \psi_A' | \psi_1 \rangle = 1.0000 \quad (18)$$

is the accuracy of theoretical result by our exact qPCA algorithm, and

$$f_2 = \langle \psi_A' | \psi_2 \rangle = 0.9991 \quad (19)$$

is the accuracy for the quantum simulator result by our exact qPCA algorithm.

Similarly, when we set the threshold $\tau = 0.8$, the result of the visualization of qubit states is given by

$$|\psi_3\rangle = [0.6708, 0.2236, 0.2236, 0.6708], \quad (20)$$

and the result of quantum simulator is given by

$$|\psi_4\rangle = [0.6651, 0.2296, 0.2119, 0.6782]. \quad (21)$$

The classical PCA should yield

$$|\psi_A'\rangle = [0.6708, 0.2236, 0.2236, 0.6708]. \quad (22)$$

FIG. 3. The quantum circuit of the exact qSVT-based qPCA we proposed.

Notice that the classical PCA should yield

$$\lambda_1 = 2, \quad u_1 = [0.70710678, 0.70710678],$$
$$\lambda_2 = 1, \quad u_2 = [-0.70710678, 0.70710678],$$

$$|\psi_A'\rangle = \frac{\lambda_1 |u_1\rangle |u_1\rangle}{\sqrt{\lambda_1^2}} = [0.5, 0.5, 0.5, 0.5]. \quad (17)$$

The accuracy for the quantum simulator result by our exact qPCA algorithm.

Similarly, when we set the threshold $\tau = 0.8$, the result of the visualization of qubit states is given by

$$|\psi_3\rangle = [0.6708, 0.2236, 0.2236, 0.6708], \quad (20)$$

and the result of quantum simulator is given by

$$|\psi_4\rangle = [0.6651, 0.2296, 0.2119, 0.6782]. \quad (21)$$

The classical PCA should yield

$$|\psi_A'\rangle = [0.6708, 0.2236, 0.2236, 0.6708]. \quad (22)$$
Therefore the corresponding the accuracy of theoretical result by our exact qPCA algorithm is given by

\[ f_3 = \langle \psi_A'' | \psi_3 \rangle = 1.0000, \quad (23) \]

and the accuracy for the quantum simulator result by our exact qPCA algorithm is given by

\[ f_4 = \langle \psi_A''' | \psi_4 \rangle = 0.9998. \quad (24) \]

![Histogram](image1.png)

(a) The visualization of qubit states of our qPCA algorithm from IBM Quantum Experience.

![Histogram](image2.png)

(b) The probability histogram of our qPCA algorithm in the quantum simulator from IBM Quantum Experience.

FIG. 4. The experimental circuit of our qPCA for the $2 \times 2$ matrix $A$ with threshold $\tau = 1$ on IBM Quantum Experience. The qubit $q[0]$ is the ancillary qubit. Before the first dash line of the quantum circuit, the qubits $q[3-4]$ are used to initialize the quantum state $|\psi_A\rangle$. Between the first dash and the second dash lines in the quantum circuit, the qubits $q[1-2]$ are used to save eigenvalues from the phase estimation. Between the second and the third dash lines in the quantum circuit, the eigenvalues $|\lambda_k\rangle$ are converted to $|y_k\rangle$ on $q[1-2]$. Between the third and the fourth dash lines is the controlled operation. The rest of the quantum circuit are the inverse operations and the measurement.

\[ \psi_A = \begin{pmatrix} -0.7071 & 0.7071 \end{pmatrix}, \quad (25) \]

\[ \psi_A' = \begin{pmatrix} -0.7071 & 0.7071 \end{pmatrix}, \quad (26) \]

\[ \psi_A'' = \begin{pmatrix} -0.7071 & 0.7071 \end{pmatrix}, \quad (27) \]

\[ \psi_A''' = \begin{pmatrix} -0.7071 & 0.7071 \end{pmatrix}, \quad (28) \]

B. The experiment for the $4 \times 4$ matrix

Now we take the $4 \times 4$ matrix

\[ C = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 3 \end{bmatrix} \quad (25) \]

as another example. The corresponding initial quantum state is given by

\[ |\psi_C\rangle = |0 \ 0 \ 0 \ 0.2673 \ 0 \ 0 \ 0.5345 \ 0 \ 0 \ 0 \ 0.8018\rangle. \quad (26) \]

The quantum circuit of our exact qPCA algorithm for the matrix $C$ requires eight qubits in total, as shown in Fig. 6. The qubits $q[0]$ and $q[7]$ are the ancillary qubits. The qubits $q[3-6]$ are used to prepare the initial quantum state $|\psi_C\rangle$, which is not straightforward to construct. Therefore we design the binary tree to prepare the state, as shown in Fig. 8 of the Appendix A and the corresponding quantum circuit for preparing the state is shown in Fig. 9. The quantum circuit of phase estimation for $|\psi_C\rangle$ is shown in Fig. 10 of the Appendix B where the output eigenvalues of the phase estimation operation are saved in $q[1-2]$.

As shown in Fig. 11 when we set the threshold $\tau = 1.8$, the result of the visualization of qubit states is given by

\[ |\psi_5\rangle = |0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0.5547 \ 0 \ 0 \ 0 \ 0.8321\rangle. \quad (27) \]

and the result of the quantum simulator is given by

\[ |\psi_6\rangle = |0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0.5294 \ 0 \ 0 \ 0 \ 0.8004\rangle. \quad (28) \]

Notice that the classical PCA should yield

\[ \lambda_1 = 0, \quad u_1 = |1 \ 0 \ 0\rangle, \]

\[ \lambda_2 = 1, \quad u_2 = |0 \ 1 \ 0\rangle, \]

\[ \lambda_3 = 2, \quad u_3 = |0 \ 0 \ 1\rangle, \]

\[ \lambda_4 = 3, \quad u_4 = |0 \ 0 \ 0\rangle, \]

\[ |\psi_C\rangle = |0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0.5547 \ 0 \ 0 \ 0 \ 0.8321\rangle. \quad (29) \]

The accuracy of our quantum algorithm can be evaluated by the inner product, i.e.

\[ f_5 = \langle \psi_C' | \psi_5 \rangle = 1.0000 \quad (30) \]
is the accuracy of theoretical result by our exact qPCA algorithm, and
\[ f_6 = \langle \psi_C^0 | \psi_6 \rangle = 0.9597 \]  
(31)
is the accuracy for the quantum simulator result by our exact qPCA algorithm.

Similarly, when we set the threshold \( \tau = 0.5 \), the result of the visualization of qubit states is given by
\[ |\psi_\tau \rangle = [0.00000.267300000.534500000.8018], \]  
(32)
and the result of the quantum simulator is given by
\[ |\psi_8 \rangle = [0.00000.250000000.548400000.7979]. \]  
(33)
The classical PCA should yield
\[ |\psi_C^0 \rangle = [0.00000.267300000.534500000.8018]. \]  
(34)
Therefore the corresponding accuracy of theoretical result by our exact qPCA algorithm is given by
\[ f_7 = \langle \psi_C^0 | \psi_\tau \rangle = 1.0000, \]  
(35)
and the accuracy for the quantum simulator result by our exact qPCA algorithm is given by
\[ f_8 = \langle \psi_C^0 | \psi_8 \rangle = 0.9997. \]  
(36)

Based on the the experimental results of 2 × 2 and 4 × 4 matrices, we can see that the theoretical results yielded by our qPCA algorithm are exactly the same as that of classical PCA. For the results yield by quantum computer simulation, our algorithm can also obtain high accuracy. The experimental results are in line with our expectations.

IV. CONCLUSION

In this paper, we proposed an exact qSVT-based qPCA algorithm. Compared with the previous qSVT-based qPCA [13] algorithm, which needs to continuously adjust parameters and yields estimated results, our algorithm does not involve any parameter adjustment, and the eigenvalues yield by our algorithm are the exact results, instead of an estimated one. Moreover, our qPCA algorithm is much simpler in terms of implementation, i.e. it requires much less quantum gates. The experimental results on IBM Quantum Experience verified the proposed algorithm, and are in line with our expectations.

Appendix A: The preparation of state \( |\psi_C\rangle \)

The initial state \( |\psi_C\rangle \) in Eqs. (26) is not straightforward to prepare on IBM Quantum Experience. Therefore we design the binary tree [21, 22] as shown in Fig. 8 to prepare the quantum state, whose leaf nodes are the vectors of the quantum state, and each branch is a \( R_y(\theta) \) unitary operation, where
\[
R_y(\theta) = \begin{bmatrix}
\cos(\frac{\theta}{2}) & -\sin(\frac{\theta}{2}) \\
\sin(\frac{\theta}{2}) & \cos(\frac{\theta}{2})
\end{bmatrix},
\]  
(A1)
y = \{0, 00, 10, 000, 010, 100, 110, 0000, 0010, 0100, 1000, 1010, 1100, 1110\}. \]  
(A2)
The corresponding quantum circuit of the binary tree is shown in Fig. 9.

Appendix B: The phase estimation of matrix \( C \)

The quantum circuit of the phase estimation on the matrix \( C \) is not straightforward to design on IBM Quantum Experience. Therefore we decompose the phase esti-
matrices in the phase estimation of $C$ are $U_1 = e^{\frac{2\pi i}{14}}$, 

\[(a) \text{ Unitary decomposition of } C - U.\]

\[(b) \text{ Unitary decomposition of } A_1.\]

\[(c) \text{ Unitary decomposition of } B_1.\]

\[(d) \text{ Unitary decomposition of } B_2.\]

FIG. 10. The unitary operation of phase estimation in $4 \times 4$ matrix $C$.

$U_2 = e^{\frac{2\pi i}{14}}$, where

\[
U_1 = \begin{bmatrix}
0 & 0 & 0 & 0 \\
0 & i & 0 & 0 \\
0 & 0 & -1 & 0 \\
0 & 0 & 0 & -i
\end{bmatrix},
\]

\[
U_2 = \begin{bmatrix}
0 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & -1
\end{bmatrix}
\]

The corresponding quantum circuits of $C - U_1$, $C - U_2$ are shown in Fig. 10.

[1] M. E. Wall, A. Rechtsteiner, and L. M. Rocha, Singular value decomposition and principal component analysis, in *A practical approach to microarray data analysis* (Springer, 2003) pp. 91–109.

[2] S. Karamizadeh, S. M. Abdullah, A. A. Manaf, M. Zamanini, and A. Hooman, An overview of principal component analysis, Journal of Signal and Information Processing 4, 173 (2013).

[3] L. Van Der Maaten, E. Postma, and J. Van den Herik, Dimensionality reduction: a comparative, J Mach Learn Res 10, 13 (2009).

[4] E. Kokkoupolou and Y. Saad, Orthogonal neighborhood preserving projections: A projection-based dimensionality reduction technique, IEEE Transactions on Pattern Analysis and Machine Intelligence 29, 2143 (2007).

[5] E. J. Keogh and M. J. Pazzani, A simple dimensionality reduction technique for fast similarity search in large time series databases, in *Pacific-Asia conference on knowledge discovery and data mining* (Springer, 2000) pp. 122–133.

[6] M. A. Nielsen and I. Chuang, Quantum computation and quantum information (2002).

[7] S. Lloyd, M. Mohseni, and P. Rebentrost, Quantum principal component analysis, Nature Physics 10, 631 (2014).

[8] C. Shao, An improved algorithm for quantum principal component analysis, arXiv preprint arXiv:1903.03099 (2019).

[9] C.-H. Yu, F. Gao, S. Lin, and J. Wang, Quantum data compression by principal component analysis, Quantum Information Processing 18, 249 (2019).

[10] D. Kopczyk, Quantum machine learning for data scientists, arXiv preprint arXiv:1804.10068 (2018).
[11] B. Duan, J. Yuan, Y. Liu, and D. Li, Efficient quantum circuit for singular-value thresholding, Physical Review A 98, 012308 (2018).
[12] C. Bravo-Prieto, D. García-Martín, and J. I. Latorre, Quantum singular value decomposer, arXiv preprint arXiv:1905.01353 (2019).
[13] 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 383, 2862 (2019).
[14] B. Duan, J. Yuan, J. Xu, and D. Li, Quantum algorithm and quantum circuit for a-optimal projection: Dimensionality reduction, Physical Review A 99, 032311 (2019).
[15] P. Rebentrost, M. Schuld, L. Wossnig, F. Petruccione, and S. Lloyd, Quantum gradient descent and newton’s method for constrained polynomial optimization, New Journal of Physics 21, 073023 (2019).
[16] A. Cross, The ibm q experience and qiskit open-source quantum computing software, APS 2018, L58 (2018).
[17] P. Balasubramanian, B. Behera, and P. Panigrahi, Circuit implementation for rational quantum secure communication using ibm q experience beta platform.
[18] D. García-Martín and G. Sierra, Five experimental tests on the 5-qubit ibm quantum computer, arXiv preprint arXiv:1712.05642 (2017).
[19] P. J. Coles, S. Eidenbenz, S. Pakin, A. Adedoyin, J. Ambroso, P. M. Anisimov, W. Casper, G. Chennupati, C. Coffrin, H. Djidjev, et al., Quantum algorithm implementations for beginners, arXiv: Emerging Technologies (2018).
[20] IBM, Ibm quantum experience, https://quantum-computing.ibm.com/.
[21] L. Grover and T. Rudolph, Creating superpositions that correspond to efficiently integrable probability distributions, arXiv preprint quant-ph/0208112 (2002).
[22] I. Kerenidis and A. Prakash, Quantum recommendation systems, arXiv preprint arXiv:1603.08675 (2016).
[23] J. J. Vartiainen, M. Möttönen, and M. M. Salomaa, Efficient decomposition of quantum gates, Physical review letters 92, 177902 (2004).
[24] C.-K. Li, R. Roberts, and X. Yin, Decomposition of unitary matrices and quantum gates, International Journal of Quantum Information 11, 1350015 (2013).
[25] H. Mohammadbagherpoor, Y.-H. Oh, A. Singh, X. Yu, and A. J. Rindos, Experimental challenges of implementing quantum phase estimation algorithms on ibm quantum computer, arXiv preprint arXiv:1903.07605 (2019).
[26] S. Dutta, A. Suau, S. Dutta, S. Roy, B. K. Behera, and P. K. Panigrahi, Demonstration of a quantum circuit design methodology for multiple regression, arXiv preprint arXiv:1811.01726 (2018).