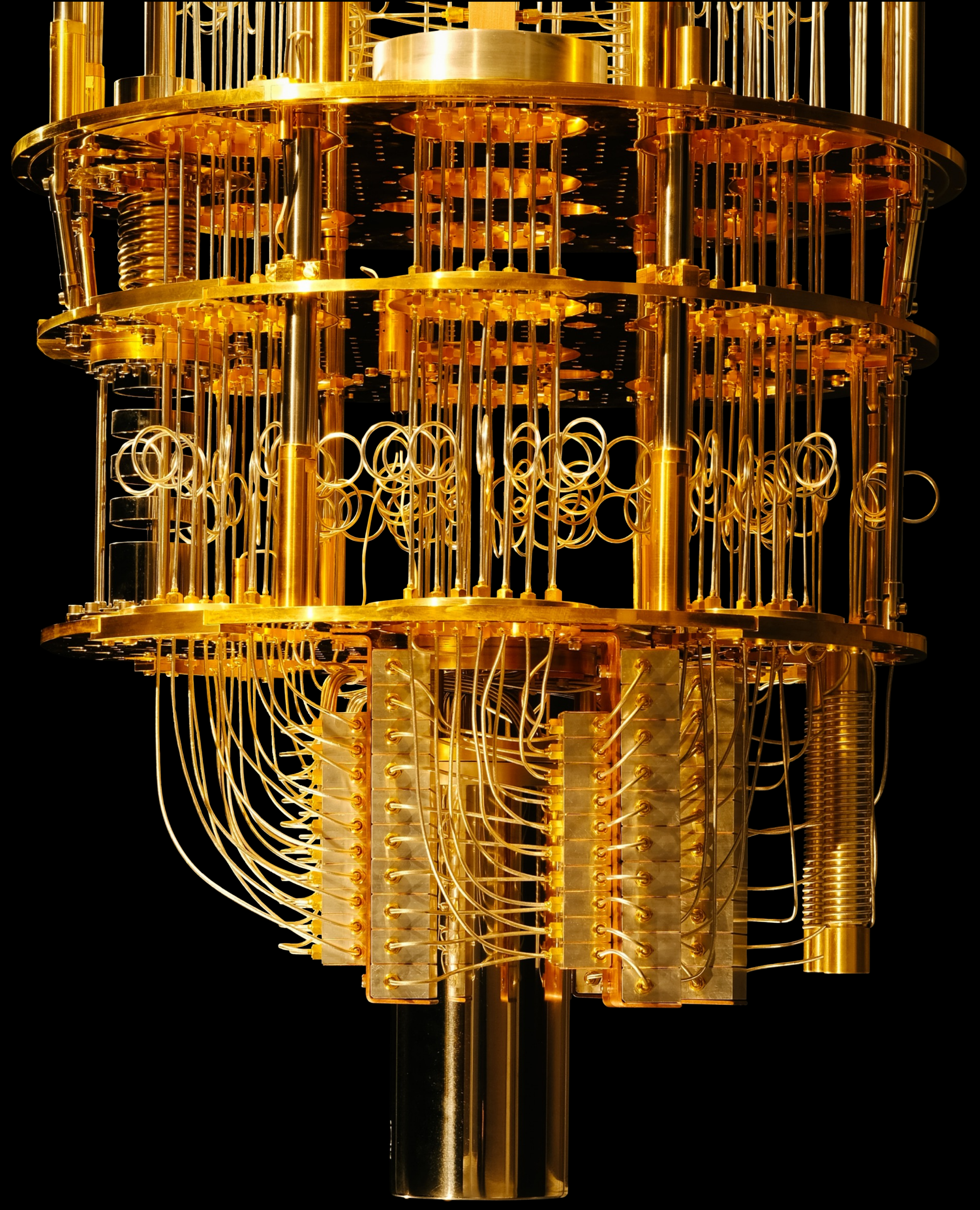


Towards an end-to-end approach for quantum principal component analysis

December 15th, 2023

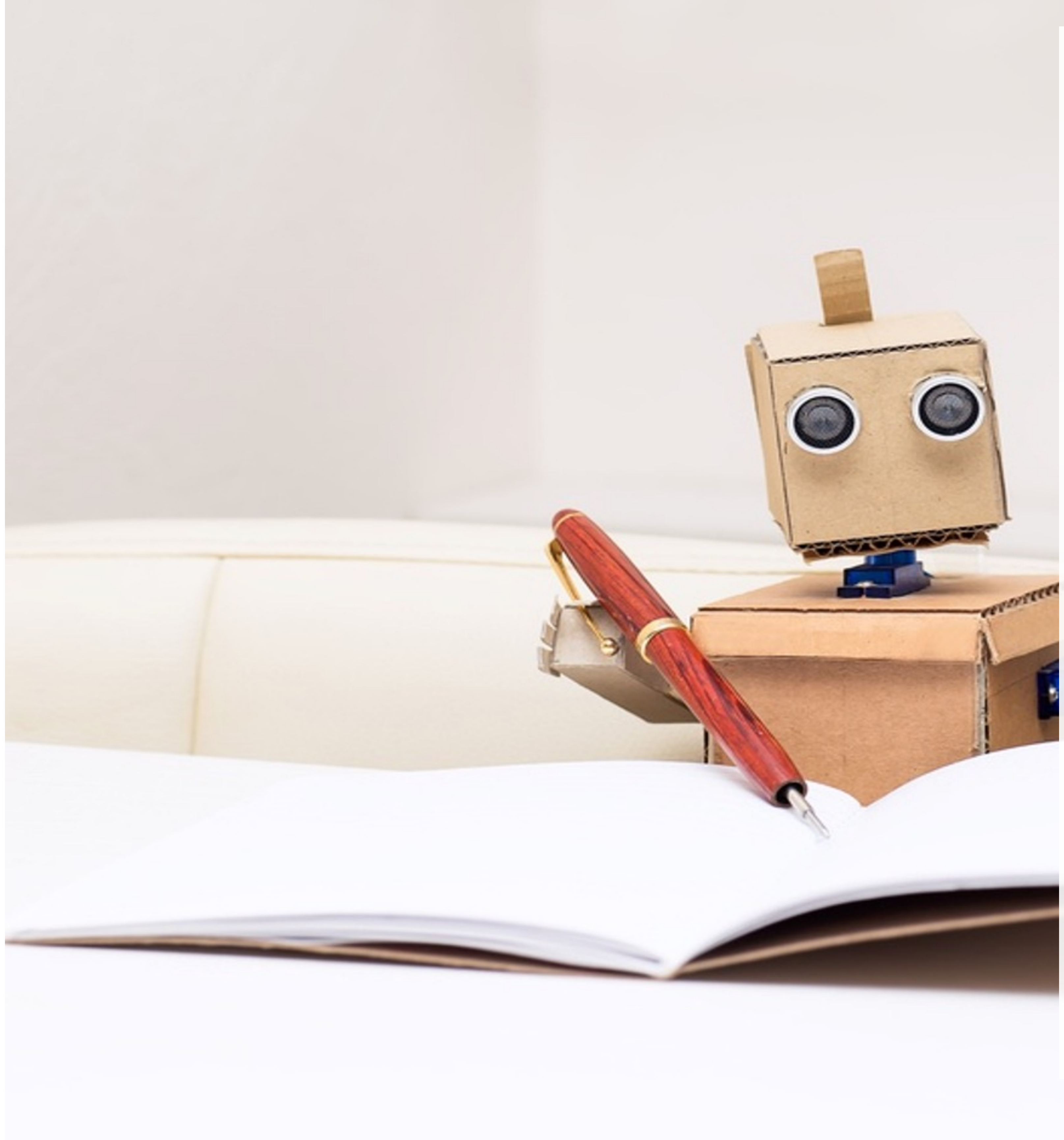
Politecnico di Torino

1859

INTESA SANPAOLO

IBM Consulting

AGENDA



- Introduction - Principal Component Analysis & QPCA
- QPCA implementation steps
- Complexity and challenges
- Conclusion and future works

Algorithm Implementation - QPCA Algorithm formalization

Problem statement PCA

- Linear transformation of the original data into new coordinate system
- Widely used in various fields, such as image processing, biology and **finance,** with use cases such as **Interest Rate Risk** specifically addressed through this work
- Necessity to compute **eigenvalues** & **eigenvectors** of the input matrix

Literature QPCA

*State-of-the-art literature propose Quantum PCA algorithm with a potential theoretical **exponentially faster** execution time compared to classical model*[1].

Current literature does **not** provide a **end-to-end generic** implementation of QPCA algorithm. Reported examples are based on **specific and restricted matrix input domain**.

Current literature provides algorithm description **limited** to the **computation** of **eigenvalues**, without output reconstruction methodology (extraction of principal components).

Implemented QPCA

*Implemented an **end-to-end** QPCA algorithm, providing a **higher** level of **generality** than current literature, **overcoming** important **limitations** on the **input domain**.*
The algorithm can be represented through the following blocks.

Preprocessing
QRAM maps the original input matrix within the corresponding quantum state.

Eigenvalues extraction
Calculation of matrix's eigenvalues through the QPE operator.

Eigenvectors extraction
Extraction of matrix's eigenvectors through Tomography.

Output reconstruction
Reconstruction of the final output matrix using previous steps' results.

Input Matrix → *Output Matrix*

[1] : "S.Lloyd, M.Mohseni, P.Rebentrost. «Quantum principal component analysis», https://doi.org/10.48550/arXiv.1307.0401"

Algorithm Implementation - QPCA steps

A high-level quantum algorithm with exponential advantage for PCA of low rank covariance matrices was proposed by Lloyd et al

1. Preprocessing - Data Loading

Gram-Schmidt decomposition

$$|\psi_A\rangle = \sum_{i=1}^{N}\sum_{j=1}^{N} A_{ij}|i\rangle|j\rangle = \sum_{k=1}^{r} \sigma_k \, |u_k\rangle \, |u_k\rangle$$

$A \in \mathbb{R}^{N \, X \, N}$ input matrix

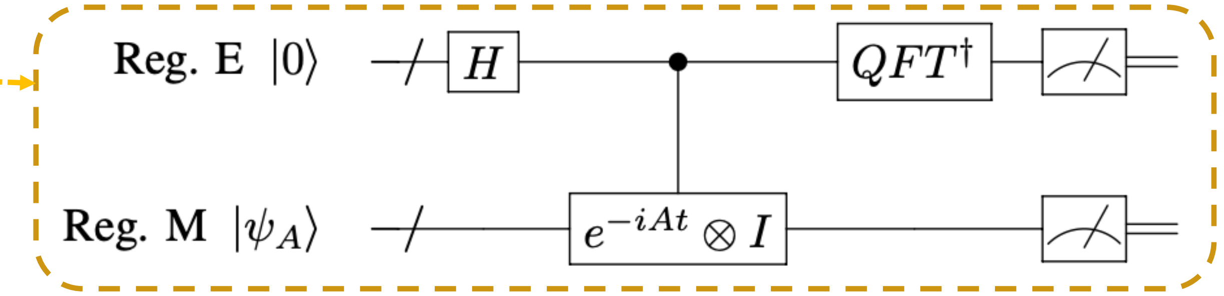
r matrix rank

σ_k k-th singular value

u_k k-th singular vector

2. Eigenvalues extraction – Phase Estimation

$$|0\rangle^E \, |\psi_A\rangle^M \xrightarrow{U_{PE}} \sum_{k=1}^{r} \sigma_k \, |\lambda_k\rangle^E \, |u_k\rangle \, |u_k\rangle$$



3. Eigenvectors extraction – Tomography

- $|\lambda_k\rangle$ is binary encoded within registers E by PE and can be easily reconstructed
- $|u_k\rangle$ cannot be straightforwardly decoded like its eigenvalue: Quantum State Vector Tomography is needed to read out its content

Algorithm Implementation - QPCA steps

3. Eigenvectors extraction – Tomography

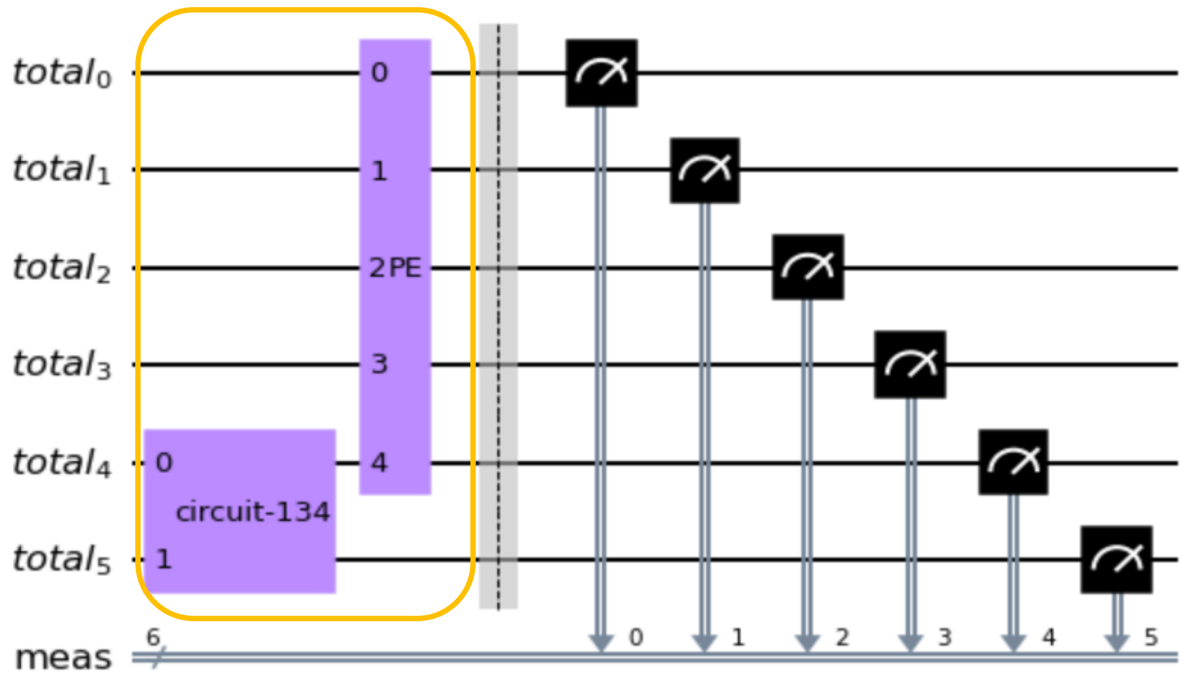
Title: "I. Kerenidis and A. Prakash. «A Quantum Interior Point Method for LPs and SDPs». ACM Transactions on Quantum Computing, vol. 1, fasc. 1, 2020, https://doi.org/10.1145/3406306 "

Algorithm 4.1 Vector state tomography algorithm.

Require: Access to a unitary U such that $U|0\rangle = |x\rangle = \sum_{i\in[d]} x_i |i\rangle$ and to its controlled version.

1. **Amplitude estimation**
   (a) Measure $N = \frac{36d\ln d}{\delta^2}$ copies of $|x\rangle$ in the standard basis and obtain estimates $p_i = \frac{n_i}{N}$ where n_i is the number of times outcome i is observed.
   (b) Store $\sqrt{p_i}, i \in [d]$ in QRAM data structure so that $|p\rangle = \sum_{i\in[d]} \sqrt{p_i}|i\rangle$ can be prepared efficiently.
2. **Sign estimation**
   (a) Create $N = \frac{36n\ln n}{\delta^2}$ copies of the state $\frac{1}{\sqrt{2}}|0\rangle \sum_{i\in[d]} x_i |i\rangle + \frac{1}{\sqrt{2}}|1\rangle \sum_{i\in[d]} \sqrt{p_i}|i\rangle$ using a control qubit.
   (b) Apply a Hadamard gate on the first qubit of each copy of the state to obtain $\frac{1}{2}\sum_{i\in[d]}[(x_i + \sqrt{p_i})|0,i\rangle + (x_i - \sqrt{p_i})|1,i\rangle]$.
   (c) Measure each copy in the standard basis and maintain counts $n(b,i)$ of the number of times outcome $|b,i\rangle$ is observed for $b \in 0, 1$.
   (d) Set $\sigma_i = 1$ if $n(0,i) > 0.4p_iN$ and -1 otherwise.
3. Output the unit vector $\widetilde{x}$ with $\widetilde{x}_i = \sigma_i\sqrt{p_i}$.

$|x\rangle$



Algorithm Implementation - QPCA steps

3. Eigenvectors extraction – Tomography

Title: "I. Kerenidis and A. Prakash. «A Quantum Interior Point Method for LPs and SDPs». ACM Transactions on Quantum Computing, vol. 1, fasc. 1, 2020, https://doi.org/10.1145/3406306 "

Algorithm 4.1 Vector state tomography algorithm.

Require: Access to a unitary U such that $U|0\rangle = |x\rangle = \sum_{i\in[d]} x_i |i\rangle$ and to its controlled version.

1. **Amplitude estimation**

   (a) Measure $N = \frac{36 d \ln d}{\delta^2}$ copies of $|x\rangle$ in the standard basis and obtain estimates $p_i = \frac{n_i}{N}$ where n_i is the number of times outcome i is observed.

   (b) Store $\sqrt{p_i}, i \in [d]$ in QRAM data structure so that $|p\rangle = \sum_{i\in[d]} \sqrt{p_i} |i\rangle$ can be prepared efficiently.

2. **Sign estimation**

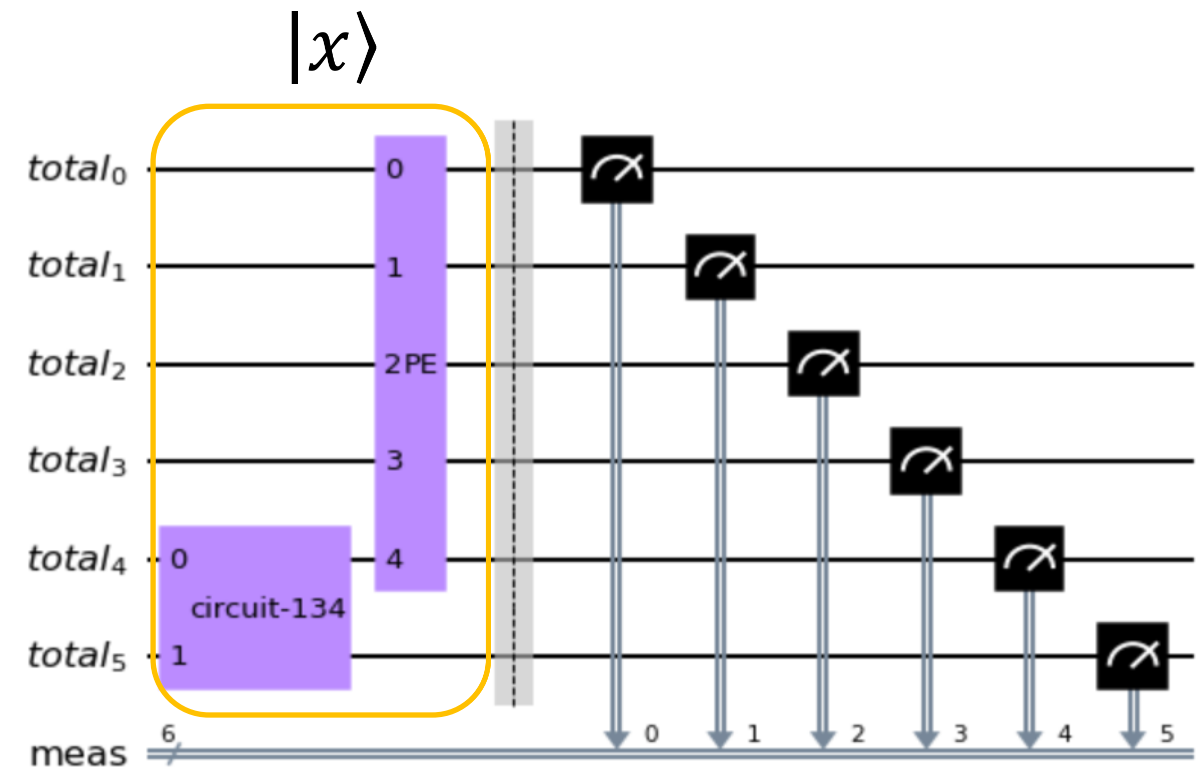
   (a) Create $N = \frac{36 n \ln n}{\delta^2}$ copies of the state $\frac{1}{\sqrt{2}} |0\rangle \sum_{i\in[d]} x_i |i\rangle + \frac{1}{\sqrt{2}} |1\rangle \sum_{i\in[d]} \sqrt{p_i} |i\rangle$ using a control qubit.

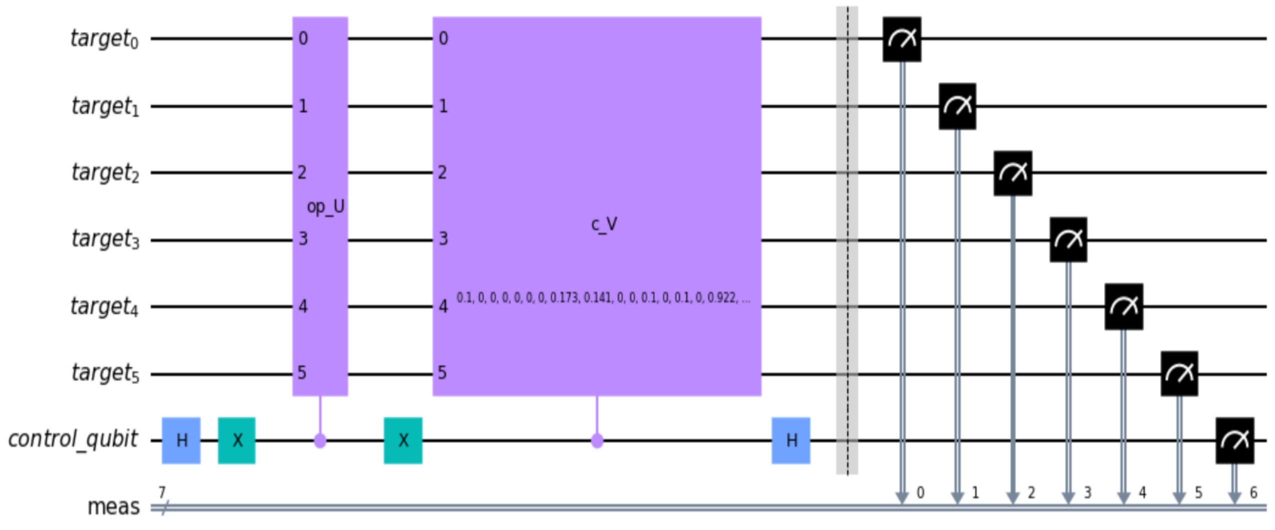
   (b) Apply a Hadamard gate on the first qubit of each copy of the state to obtain $\frac{1}{2} \sum_{i\in[d]} [(x_i + \sqrt{p_i}) |0, i\rangle + (x_i - \sqrt{p_i}) |1, i\rangle]$.

   (c) Measure each copy in the standard basis and maintain counts $n(b, i)$ of the number of times outcome $|b, i\rangle$ is observed for $b \in 0, 1$.

   (d) Set $\sigma_i = 1$ if $n(0, i) > 0.4 p_i N$ and -1 otherwise.

3. Output the unit vector $\widetilde{x}$ with $\widetilde{x}_i = \sigma_i \sqrt{p_i}$.





Algorithm Implementation - QPCA steps

4. Output reconstruction – Custom postprocessing

- 2x2 matrix with eigenvectors $|u_k\rangle$ encoded in one qubit. Our objective is to reconstruct a0 and a1.

$$|u_k\rangle = \sum_{i=0}^{1} a_i|i\rangle = a_0|0\rangle + a_1|1\rangle.$$

- Therefore, we can express:

$$|u_k\rangle|u_k\rangle = b_0|00\rangle + b_1|01\rangle + b_2|10\rangle + b_3|11\rangle = a_0^2|00\rangle + a_0a_1|01\rangle + a_1a_0|10\rangle + a_1^2|11\rangle.$$

- After normalization, we can bring to common factor:

$$b_0|0\rangle\left(|0\rangle + \frac{b_1}{b_0}|1\rangle\right) + b_3|1\rangle\left(\frac{b_2}{b_3}|0\rangle + |1\rangle\right)$$

$$\begin{cases} \sqrt{b_3}\left(\frac{b_2}{b_3}\right) = \frac{b_2}{\sqrt{b_3}} = \frac{a_1a_0}{\sqrt{a_1^2}} = a_0 \\ \sqrt{b_3} = a_1 \end{cases}$$

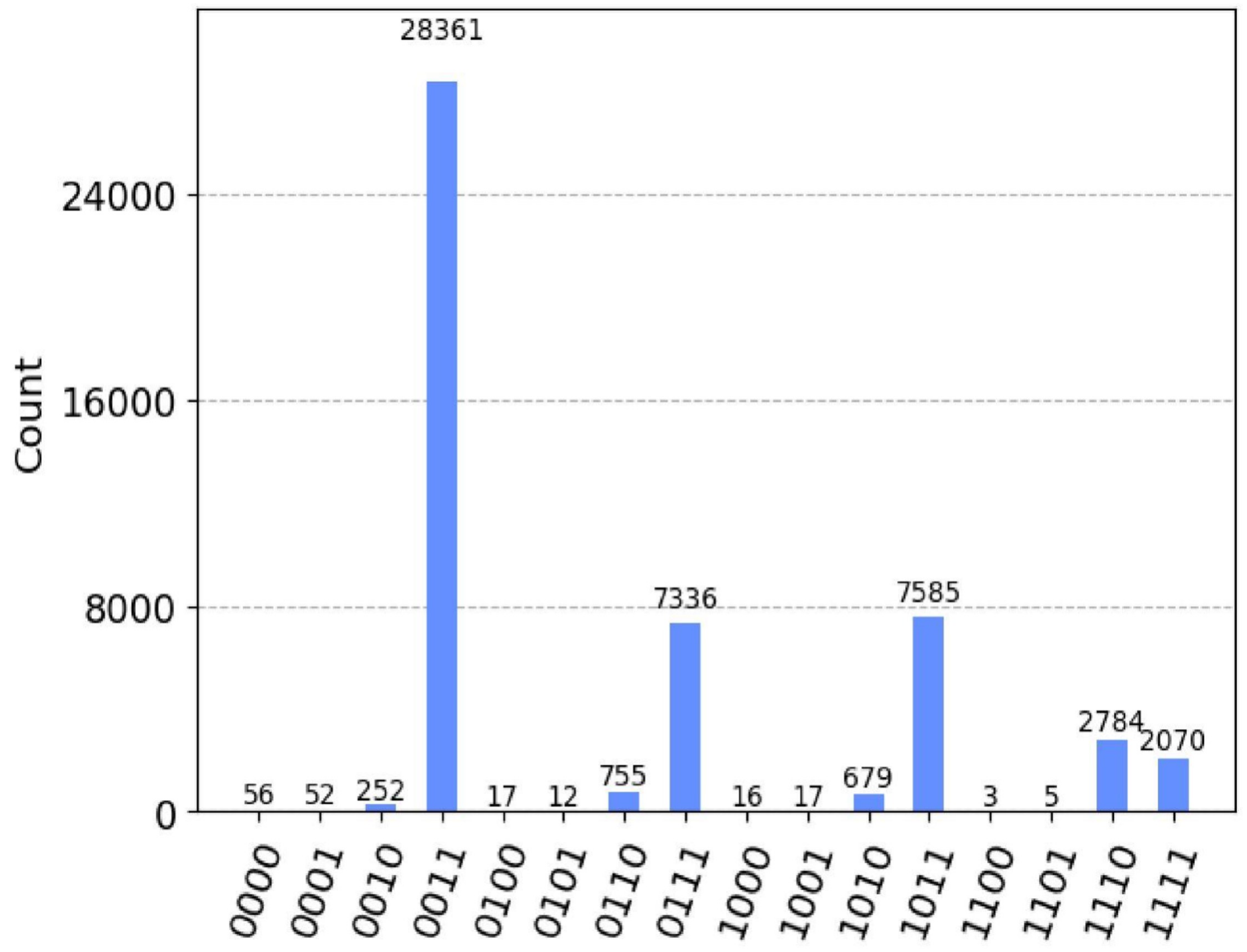
Algorithm Experiments

We generated a 2x2 matrix with synthetic data validated by domain experts. The runs were executed using ibm qasm_simulator and 2 qubits **of resolution**

$$A = \begin{bmatrix} 0.6507 & 0.2122 \\ 0.2122 & 0.3493 \end{bmatrix}$$

$$\lambda_1 = 0.760 \quad u_1 = \begin{bmatrix} 0.889 & 0.459 \end{bmatrix}$$

$$\lambda_2 = 0.240 \quad u_2 = \begin{bmatrix} -0.459 & 0.889 \end{bmatrix}$$



Algorithm Experiments

We generated a 2x2 matrix with synthetic data validated by domain experts. The runs were executed using ibm qasm_simulator and 2 qubits **of resolution**

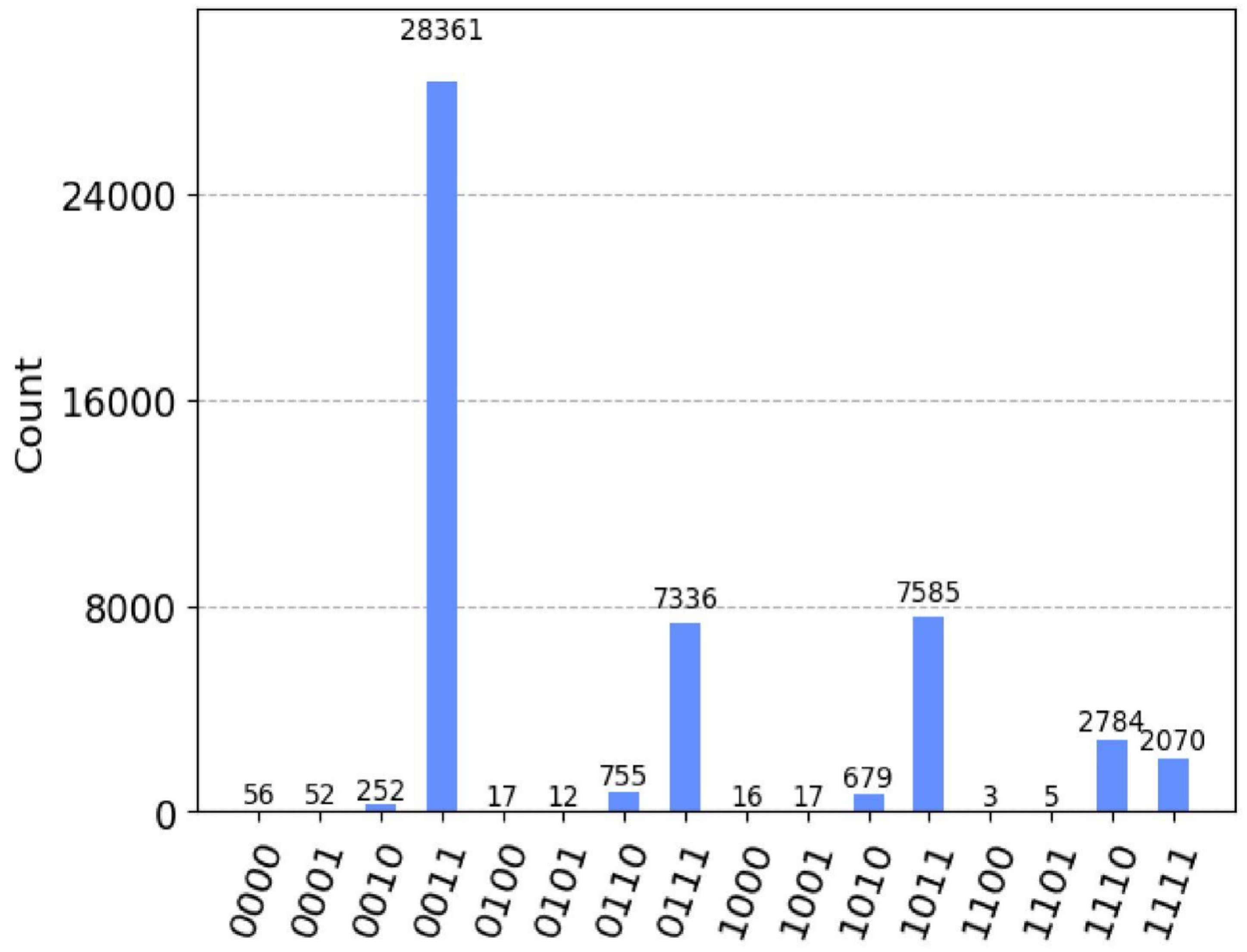
$$A = \begin{bmatrix} 0.6507 & 0.2122 \\ 0.2122 & 0.3493 \end{bmatrix}$$

$$\lambda_1 = 0.760 \quad u_1 = \begin{bmatrix} 0.889 & 0.459 \end{bmatrix}$$

$$\lambda_2 = 0.240 \quad u_2 = \begin{bmatrix} -0.459 & 0.889 \end{bmatrix}$$

$$\lambda_1 = 0.75 \quad u_1 = \begin{bmatrix} 0.889 & 0.452 \end{bmatrix}$$

$$\lambda_2 = 0.25 \quad u_2 = \begin{bmatrix} -0.439 & 0.888 \end{bmatrix}$$



Algorithm Complexity

Qubits

- State Preparation*: $O(log(d^2))$
- Phase Estimation: **n qubit for** $\frac{1}{2^n}$ ***resolution***
- Sign Estimation: $+1$ ***control qubit***

Time

- *State Preparation:* $O(d^{2log(2)})$
- *Phase Estimation:* $O(t^2\epsilon^{-1}d^{2log(2)})$
- *Sign Estimation:* $O(\delta^{-2}t^2\epsilon^{-1}d * log(d^2))$

- d matrix dimension
- δ,ε precision parameters that depend on the condition number
- t time step

* V. Shende, S. Bullock and I. Markov, "Synthesis of quantum-logic circuits," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 25, no. 6, June 2006, doi: 10.1109/TCAD.2005.855930.

Future works

- Improve scalability and complexity, especially with respect to the crucial step of data loading

- Figuring out effective quantum thresholds methodologies for QPCA

An improved quantum principal component analysis algorithm based on the quantum singular threshold method

Jie Lin [a b], Wan-Su Bao [a b], Shuo Zhang [a b], Tan Li [a b], Xiang Wang [a b]

Show more

A Low-Complexity Quantum Principal Component Analysis Algorithm

CHEN HE[1] (Member, IEEE), JIAZHEN LI[1], WEIQI LIU[1], JINYE PENG[1], AND Z. JANE WANG[2] (Fellow, IEEE)

[1]School of Information Science and Technology, Northwest University, Xi'an 710069, China

[2]Department of Electrical and Computer Engineering, University of British Columbia, Vancouver, BC V6T 1Z4, Canada

Thank you for your attention!

Emanuele Dri, Antonello Aita, Tommaso Fioravanti, Giulia Franco, Edoardo Giusto, Giacomo Ranieri, Davide Corbelletto, Bartolomeo Montrucchio, "Towards an end-to-end approach for quantum principal component analysis" 2023 IEEE International Conference on Quantum Computing and Engineering (QCE) doi.org/10.1109/QCE57702.2023.10175

GitHub Code Repository: github.com/Eagle-quantum/QuPCA

Appendix

3. Eigenvectors extraction – Numerical example

- Considering a 2x2 input matrix with the following eigenvalues/eigenvectors

$$\lambda_1 = 2,\ u_1 = [0.7071, 0.7071]^T \qquad \lambda_2 = 1,\ u_2 = [-0.7071, 0.7071]^T$$

- The output state $|\psi_A'\rangle$ after the QPE is

$$|\psi_A'\rangle = |\lambda_1\rangle|u_1\rangle|u_1\rangle + |\lambda_2\rangle|u_2\rangle|u_2\rangle$$

zoom

$$= \frac{1}{2}|1000\rangle + \frac{1}{2}|1001\rangle + \frac{1}{2}|1010\rangle + \frac{1}{2}|1011\rangle$$

$$= |10\rangle \otimes \left(\frac{1}{2}|00\rangle + \frac{1}{2}|01\rangle + \frac{1}{2}|10\rangle + \frac{1}{2}|11\rangle\right)$$

2

$$\frac{1}{2}|0\rangle(|0\rangle + |1\rangle) + \frac{1}{2}|1\rangle(|0\rangle + |1\rangle)$$

$$\frac{1}{\sqrt{2}} = 0.7071$$

C. He, J. Li, W. Liu, J. Peng, and Z. J. Wang, "A low-complexity quantum principal component analysis algorithm"

Appendix

3. Eigenvectors extraction – Numerical example

- Considering a 2x2 input matrix with the following eigenvalues/eigenvectors

$$\lambda_1 = 2,\ u_1 = [0.7071, 0.7071]^T \qquad \lambda_2 = 1,\ u_2 = [-0.7071, 0.7071]^T$$

- The output state $|\psi_A'\rangle$ after the QPE is

$$|\psi_A'\rangle = |\lambda_1\rangle|u_1\rangle|u_1\rangle + |\lambda_2\rangle|u_2\rangle|u_2\rangle$$

zoom

$$= \frac{1}{2}|1000\rangle + \frac{1}{2}|1001\rangle + \frac{1}{2}|1010\rangle + \frac{1}{2}|1011\rangle$$

$$= |10\rangle \otimes \left(\frac{1}{2}|00\rangle + \frac{1}{2}|01\rangle + \frac{1}{2}|10\rangle + \frac{1}{2}|11\rangle\right)$$

$10 \rightarrow 2$

$$\frac{1}{2}|0\rangle(|0\rangle + |1\rangle) + \frac{1}{2}|1\rangle(|0\rangle + |1\rangle)$$

$$\frac{1}{\sqrt{2}} = 0.7071$$

Quantum State Vector Tomography to reconstruct the state vector of a generic quantum state

C. He, J. Li, W. Liu, J. Peng, and Z. J. Wang, *"A low-complexity quantum principal component analysis algorithm"*

Appendix

Addressed complex state vector generation to the QPE inability to represent non-integer eigenvalues.

QPE - Integer representation

Phase estimation algorithm outputs a superposition state of eigenvectors of the input matrix, with each eigenvector weighted by a phase factor determined by the corresponding eigenvalue.

For **non-integer eigenvalues, the phase factor cannot perfectly represent an integer multiple of 2^n**, so the resulting superposition will have complex coefficients.

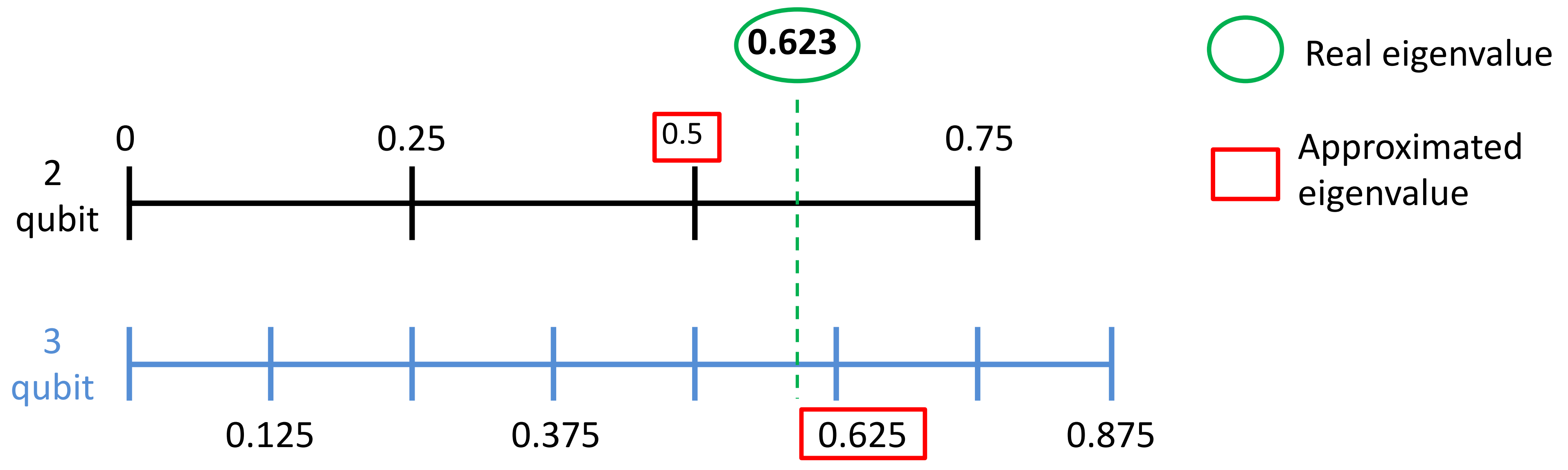
The number of states in the superposition will be finite and determined by the **precision of the phase estimation algorithm** and the **number of qubits used to represent the phase.**

Eigenvalue representation example: 2 qubit resolution

Phase is represented based on qubits used in the computation, considering binary representation of non-integer values

q_1 ↔ $\frac{1}{2^1}$ q_2 ↔ $\frac{1}{2^2}$

- $\{00\}: \ 0\left(\frac{1}{2^1}\right) + 0\left(\frac{1}{2^2}\right) = 0$
- $\{01\}: \ 0\left(\frac{1}{2^1}\right) + 1\left(\frac{1}{2^2}\right) = 0.25$
- $\{10\}: \ 1\left(\frac{1}{2^1}\right) + 0\left(\frac{1}{2^2}\right) = 0.5$
- $\{11\}: \ 1\left(\frac{1}{2^1}\right) + 1\left(\frac{1}{2^2}\right) = 0.75$



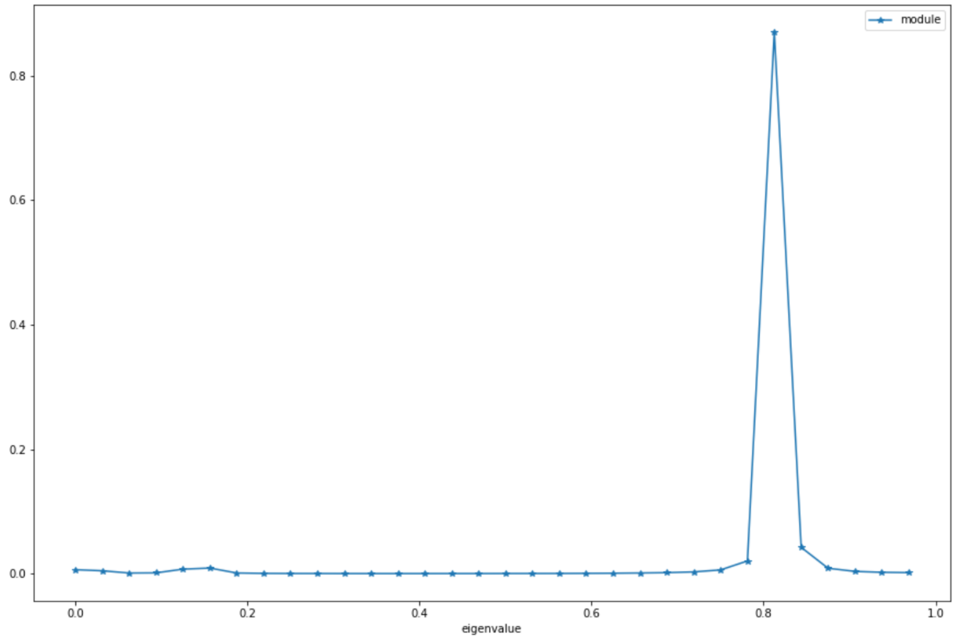
Higher accuracy in eigenvalues representation is achievable through QPE resolution increase

Appendix

Benchmark using real data comparing QPCA results on noiseless simulator for different values of resolution.

Resolution parameter has a strong influence on the accuracy of estimated eigenvalues and eigenvectors ,affecting directly Quantum Phase Estimation calculation.
As **resolution increases**, the **noise** in the measurements **attenuates**, allowing peaks detection corresponding to eigenvalues of the system.

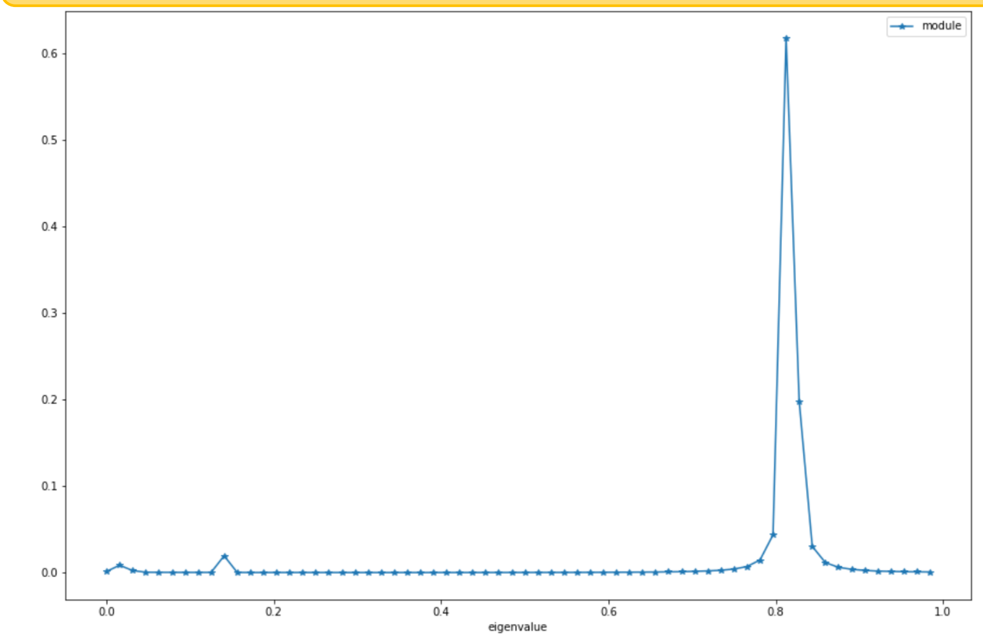
QPE (Resolution 5)



Percentage of Eigenvalues estimated : 50 %

Eigenvalues estimation mean error (L2) : 0.0059

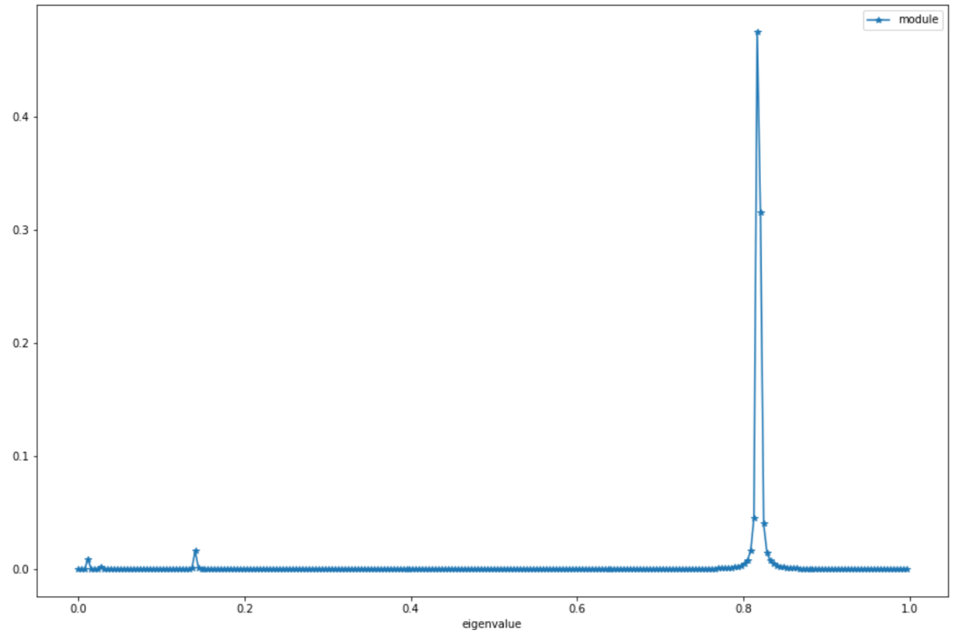
QPE (Resolution 6)



Percentage of Eigenvalues estimated : 75 %

Eigenvalues estimation mean error (L2) : 0.0027

QPE (Resolution 8)



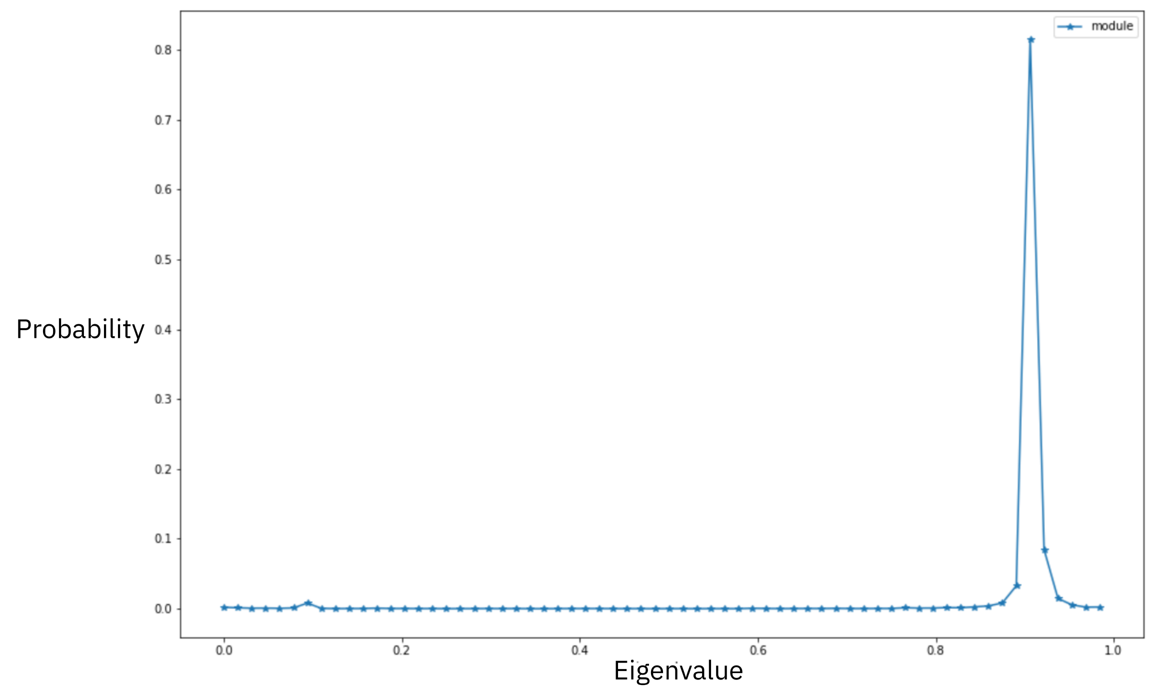
Percentage of Eigenvalues estimated : 100 %

Eigenvalues estimation mean error (L2) : 0.0018

Appendix

Detected peaks for noiseless and noisy executions are reported to visualize the differences and the impact of quantum noise on 2x2 real matrices

QPE (Noiseless Simulator)



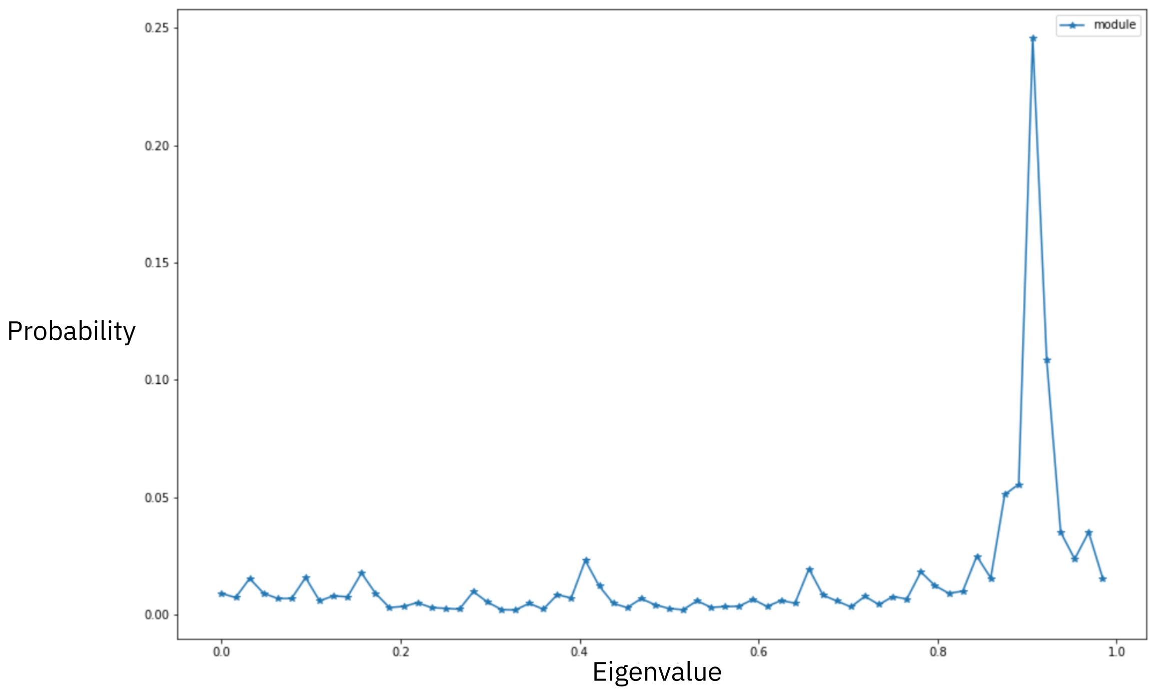
Percentage of Eigenvalues estimated : 100 %

Eigenvalues estimated : 0.90625, 0.09375

First eigenvector estimation mean error (L2-norm) : 0.0147

Second eigenvector estimation mean error (L2-norm) : 0.1478

QPE (Noisy Simulator)



Percentage of Eigenvalues estimated : 100 %

Eigenvalues estimated : 0.90625, 0.96875

First eigenvector estimation mean error (L2-norm) : 0.0643

Second eigenvector estimation mean error (L2-norm) : 1.3586

Findings

- Noisy simulator yields a significantly more jagged output compared to the noiseless one, which certainly affects the algorithm's performance

Results interpretation

- Noise is not beneficial in situations where the number of resolution qubits is already sufficient for accurately estimating the eigenvalues

*Resolution 6