国際融合科学論/先端融合科学論

LECTURE 05
Quantum Mechanics II: Quantum Machine Learning

Dr. Suyong Eum

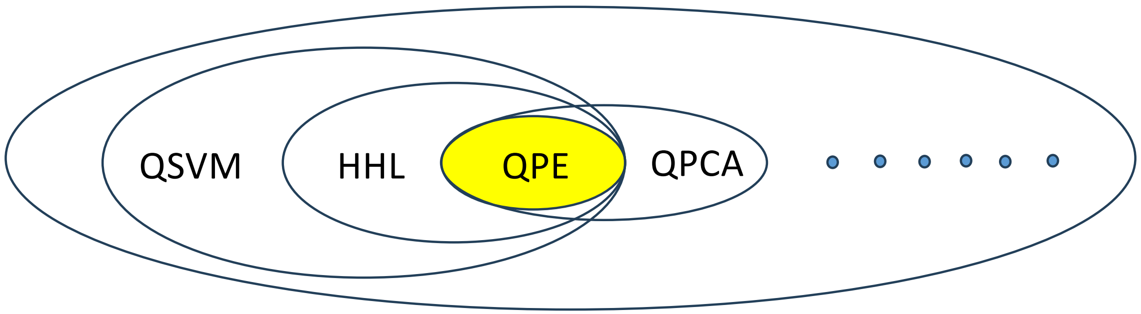
OSAKA UNIVERSITY

Lecture Outline

1) One key building block of quantum machine learning (QML)
   A. Quantum Phase Estimation (QPE)
2) A brief introduction to
   A. Harrow-Hassidim-Lloyd (HHL)
   B. Quantum SVM (QSVM)

Quantum Machine Learning: why QPE?

- QPCA and QSVM are the quantum counterparts of PCA and SVM.
- The QPE algorithm is a fundamental building block for many quantum algorithms, including QPCA and QSVM.
- The QPE algorithm is a versatile and powerful tool in quantum computing, enabling a wide range of applications.



Quantum Phase Estimation (QPE)

Quantum Phase Estimation: beginning

- "Quantum Phase Estimation (QPE)" is an algorithm for estimating the eigenvalues λ of a **unitary matrix U** using a quantum computer.

$$U|\psi\rangle = \lambda|\psi\rangle$$

- Eigen vector
- Eigen value

Quantum Phase Estimation: beginning

- "Quantum Phase Estimation (QPE)" is an algorithm for estimating the eigenvalues λ of a **unitary matrix U** using a quantum computer.

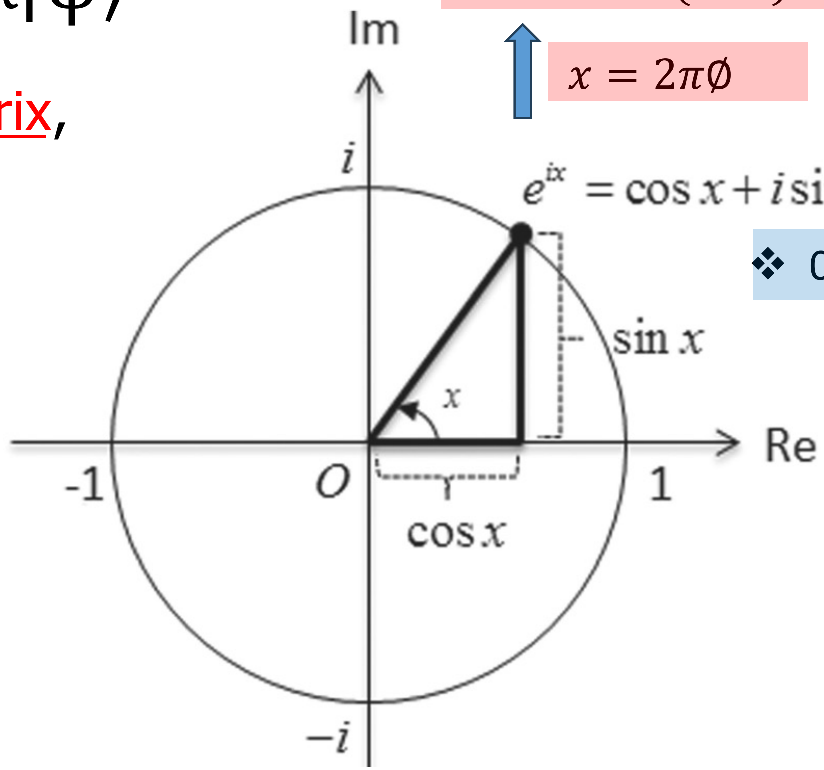
$$U|\psi\rangle = \lambda|\psi\rangle$$

❖ $0 \leqq \emptyset \leqq 1$

$e^{2\pi i\emptyset} = \cos(2\pi\emptyset) + i\sin(2\pi\emptyset)$

$x = 2\pi\emptyset$

❖ $0 \leqq x \leqq 2\pi$



- Assuming the matrix U is a unitary matrix,

$$\langle\psi|U^\dagger U|\psi\rangle = \langle\psi|\lambda^* \lambda|\psi\rangle$$

$$\langle\psi|\psi\rangle = |\lambda|^2\langle\psi|\psi\rangle$$

$$|\lambda|^2 = 1$$

$$\lambda = \cos(2\pi\emptyset) + i\sin(2\pi\emptyset) = e^{2\pi i\emptyset}$$

Euler's formula

Quantum Phase Estimation: beginning

- Thus, the previous expression can be written as follows:

$$\mathrm{U}|\psi\rangle = \lambda|\psi\rangle \quad \Longrightarrow \quad \mathrm{U}|\psi\rangle = e^{2\pi i\emptyset}|\psi\rangle$$

- The phase, ∅, is in the range between 0 and 1, which has a decimal format (∅).
- QPE is to estimate the phase, ∅, using qubits which has a binary format($\emptyset_n$).
- Thus, it would be convenient to express a decimal format as a binary format.

$$\emptyset = 0.\emptyset_1\emptyset_2 \ldots \emptyset_n = \sum_{k=1}^{n} \emptyset_k\, 2^{-k}$$

where $0 \leqq \emptyset \leqq 1$, and $\emptyset_n \in \{0,1\}$

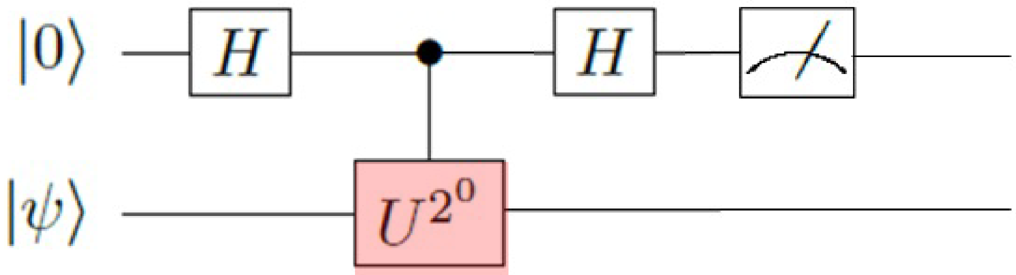
$$0.11_2 = 0.75_{10}$$

- In the above, when $\emptyset_1\emptyset_2 \ldots \emptyset_n \{0,1\}$ are known, the phase ∅ can be obtained.

0.11_2 $\quad$ 0.75_{10}

Toy example

Quantum Phase Estimation: toy example



- We want to know how much phase is made due to this unitary matrix "U", given the eigen vector $|\psi\rangle$.

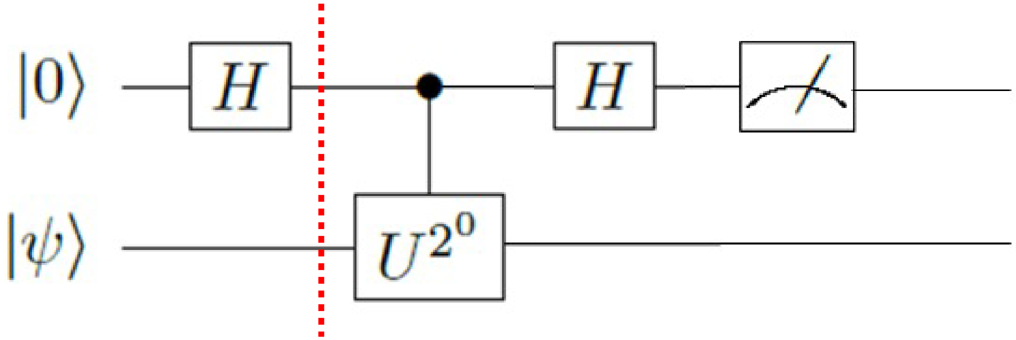
$$U|\psi\rangle = \lambda|\psi\rangle \longrightarrow U|\psi\rangle = e^{2\pi i \emptyset}|\psi\rangle$$

Eigen value (λ) — Phase (∅)

$$U = e^{2\pi i \emptyset}$$

Please, remember this.
We are going to use it.

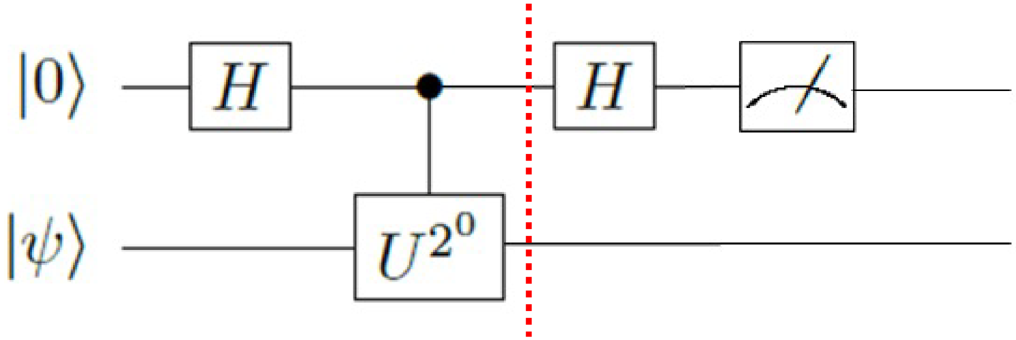
Quantum Phase Estimation: toy example



$$|+\rangle \otimes |\psi\rangle$$

$$= |0\rangle|\psi\rangle + |1\rangle|\psi\rangle$$

Quantum Phase Estimation: toy example



- control qubit
- target qubit

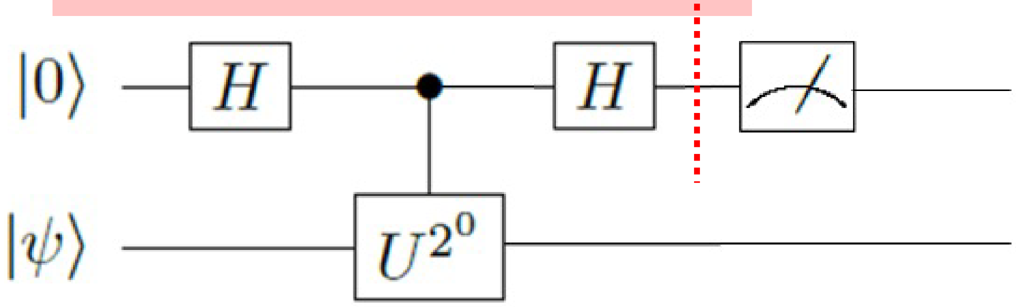
$$|+\rangle \otimes |\psi\rangle$$
$$= |0\rangle|\psi\rangle + |1\rangle|\psi\rangle$$

$$|0\rangle|\psi\rangle + |1\rangle U|\psi\rangle$$
$$= |0\rangle|\psi\rangle + e^{2\pi i 0.\phi_1}|1\rangle|\psi\rangle$$
$$= (|0\rangle + e^{2\pi i 0.\phi_1}|1\rangle) \otimes |\psi\rangle.$$

- The phase of U is encoded in the top qubit. "Kick back"

Quantum Phase Estimation: toy example

$(1+e^{2\pi i 0.\phi_1})\,|0\rangle + (1-e^{2\pi i 0.\phi_1})|1\rangle$



$$H\left(|0\rangle + e^{2\pi i 0.\phi_1}|1\rangle\right)$$

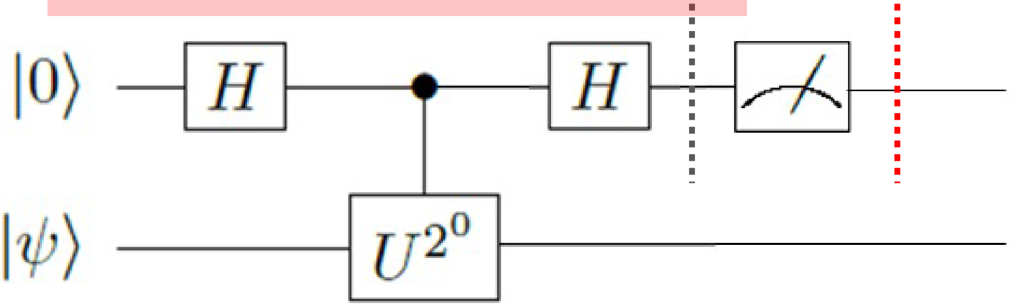
$$= H|0\rangle + H|1\rangle e^{2\pi i 0.\phi_1}$$

$$= (|0\rangle + |1\rangle) + (|0\rangle - |1\rangle)\, e^{2\pi i 0.\phi_1}$$

$$= (1+e^{2\pi i 0.\phi_1})\,|0\rangle + (1-e^{2\pi i 0.\phi_1})|1\rangle$$

Quantum Phase Estimation: toy example

$(1 + e^{2\pi i 0.\phi_1})\,|0\rangle + (1 - e^{2\pi i 0.\phi_1})|1\rangle$



1) If we measure $|0\rangle$

- ϕ_1 must be 0,

$\Rightarrow (1 + e^{2\pi i 0.0})\,|0\rangle + (1 - e^{2\pi i 0.0})|1\rangle$

$= 2|0\rangle \Rightarrow |0\rangle$

Please, remember that the coefficient ½ in front of each term is being omitted.

2) If we measure $|1\rangle$

- ϕ_1 must be 1.

$\Rightarrow (1 + e^{2\pi i 0.1_2})\,|0\rangle + (1 - e^{2\pi i 0.1_2})|1\rangle$

$= (1 + e^{2\pi i\, 0.5_{10}})|0\rangle + (1 - e^{2\pi i\, 0.5_{10}})|1\rangle$

$= 2|1\rangle \Rightarrow |1\rangle$

- The phase ($\emptyset$) becomes either 0 or ½,
- The eigen value ($e^{2\pi i\emptyset}$) becomes either 1 or -1

$$e^{2\pi i\emptyset} = \cos(2\pi\emptyset) + i\sin(2\pi\emptyset)$$

General Quantum Circuit for QPE

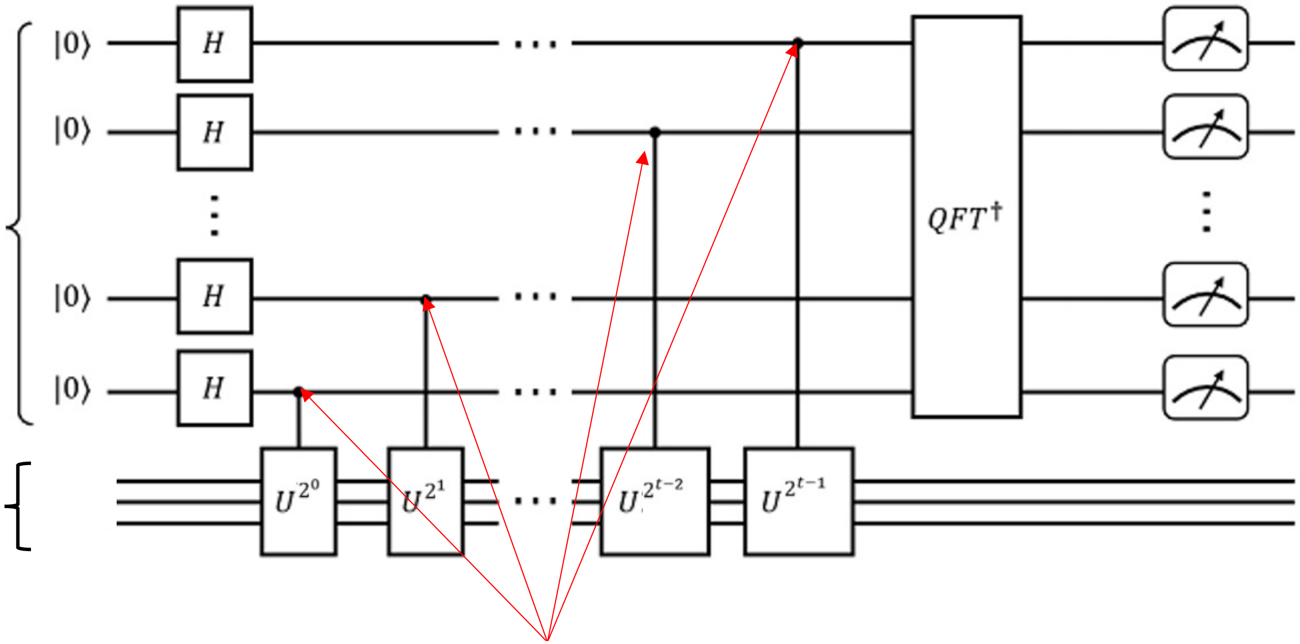
Quantum Phase Estimation: general quantum circuit

Counting register

"t" input qubits

"t" determines the accuracy of the estimation

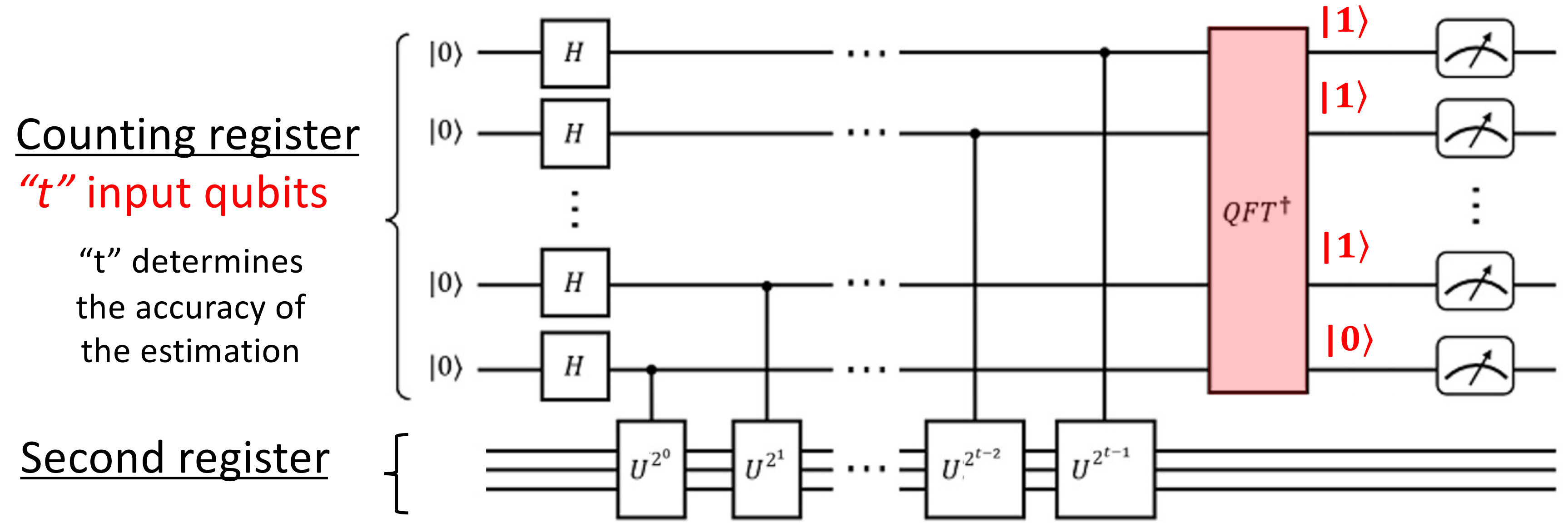
Second register



The phase of U matrix is "kickback" to *t* qubits

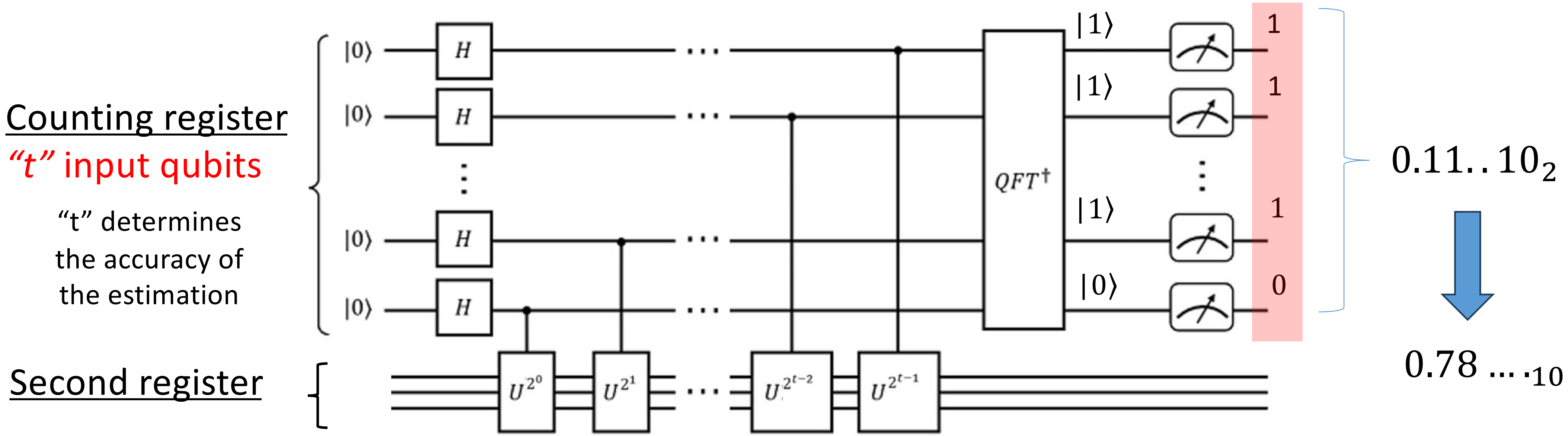
- The quantum phase estimation algorithm uses phase kickback to write the phase of U to the t qubits in the counting register.

Quantum Phase Estimation: general quantum circuit



- Inverse Quantum Fourier transform (QFT): ($QFT^\dagger$), which is a process to read out the output and produce $|0\rangle$ or $|1\rangle$

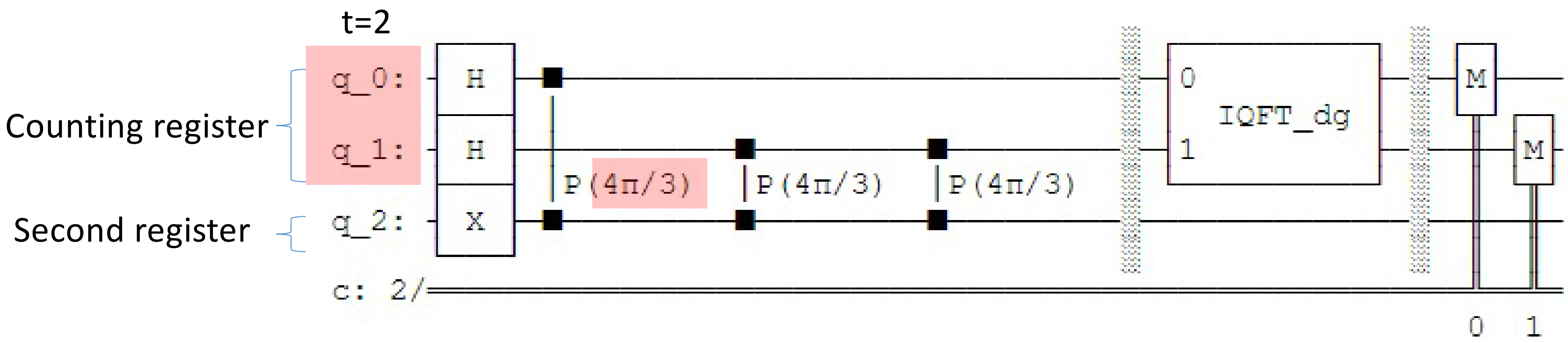
Quantum Phase Estimation: general quantum circuit



- Measurement is carried out, and the phase is encoded in binary format, e.g., $0.11..10_2$
- Next, it is converted to decimal format, e.g., $0.78\ldots_{10}$

Implementation

Quantum Phase Estimation: Qiskit implementation (t=2)



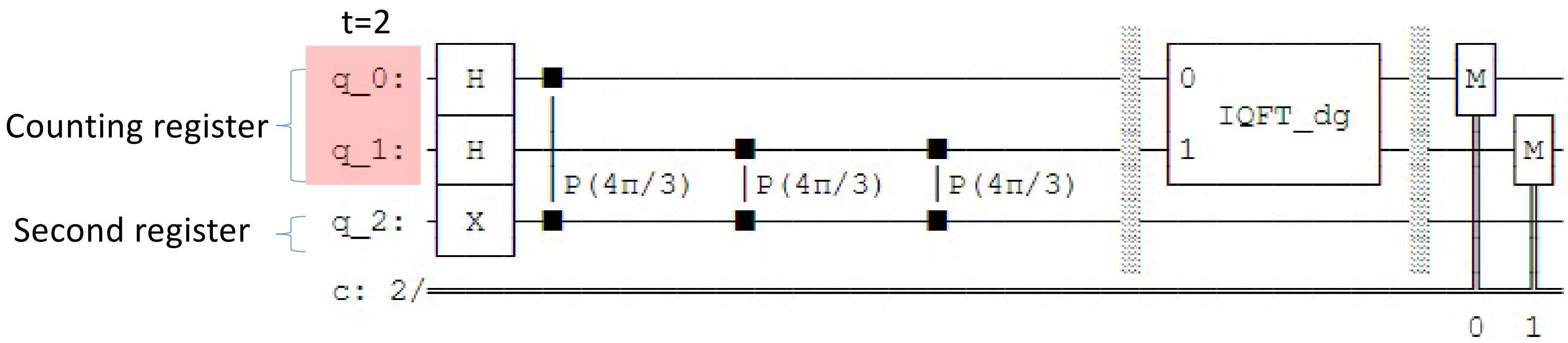
- Creating the circuit above to estimate the phase of a unitary operator U which is equivalent to estimate θ below.

$$U|\psi\rangle = e^{2\pi i\theta}|\psi\rangle$$

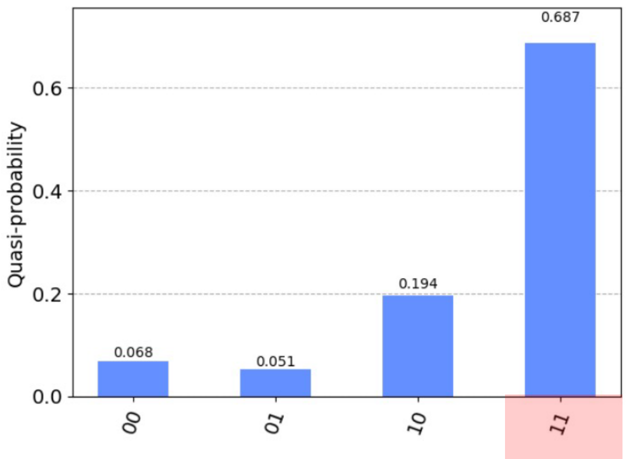
- Assume that the phase of the unitary matrix U is $4\pi/3$. In other words, the quantum phase estimation algorithm will find the value θ below.

$$2\pi\theta = 4\pi/3 \longrightarrow \theta = 2/3 \longleftarrow$$ The answer we expect

Quantum Phase Estimation: Qiskit implementation (t=2)



- The exact solution is "θ =2/3 ($4\pi/3$)" in decimal number.
- The result is approximately "11", which corresponds to 3/4 in decimal form.



$$\sum_{k=1}^{n} \emptyset_k \, 2^{-k}$$

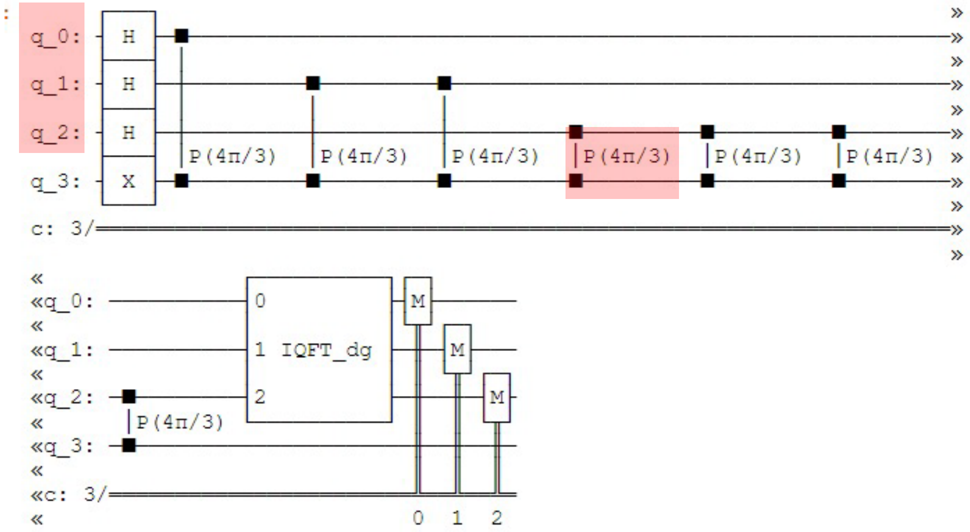
$\emptyset_k$: each digit in binary

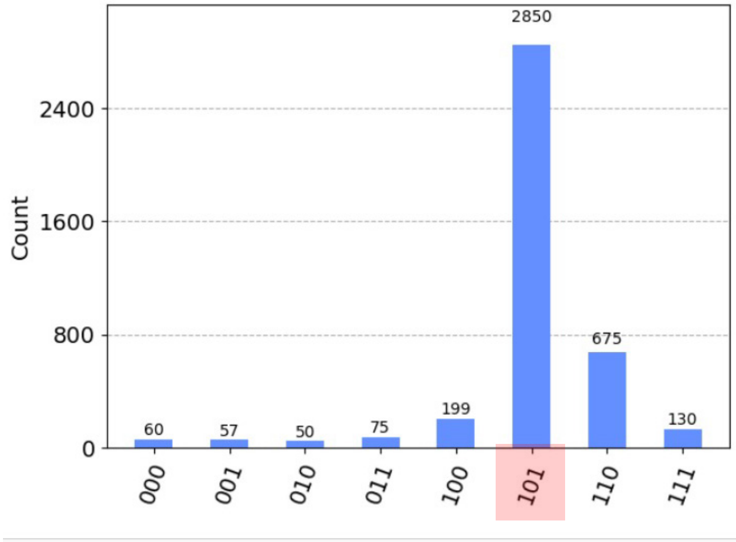
$\theta = 0.11_2 = 0.75_{10}$

$1*2^{-1} + 1*2^{-2} = 3/4$

Quantum Phase Estimation: Qiskit implementation (t=3)

t=3



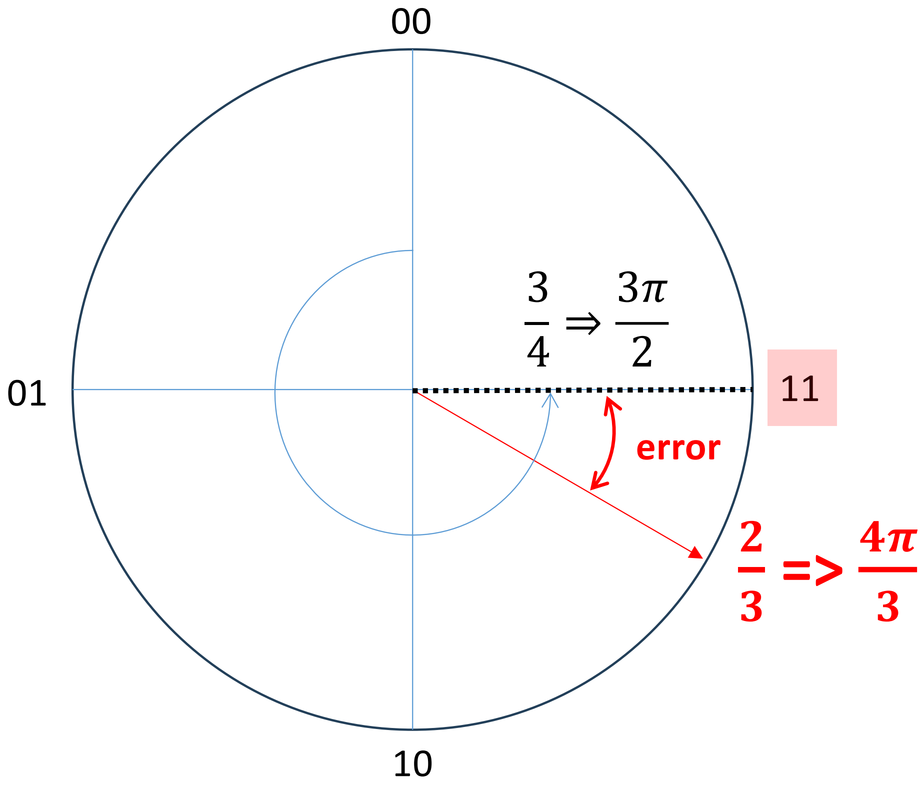


$$\theta = 0.101_2 = 0.625_{10}$$

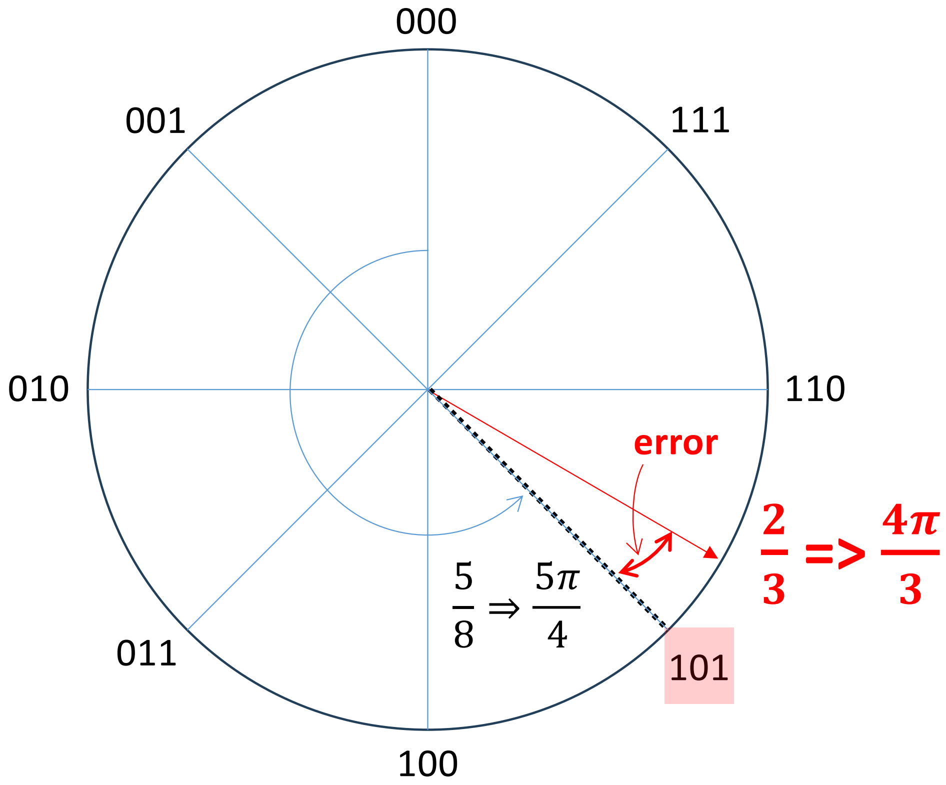
$$1*2^{-1} + 0*2^{-2} + 1*2^{-3} = 5/8$$

- ❑ What is the reason for using more qubits in the counting register?

Quantum Phase Estimation: Comparison (t=2) vs (t=3)



t=2



t=3

Harrow-Hassidim-Lloyd (HHL)

Harrow-Hassidim-Lloyd (HHL)

- The HHL is for solving linear systems of equations, developed by Aram Harrow, Avinatan Hassidim, and Seth Lloyd (HHL) in 2009.

$$A\vec{x} = \vec{b} \qquad \begin{bmatrix} 1 & -1/3 \\ -1/3 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

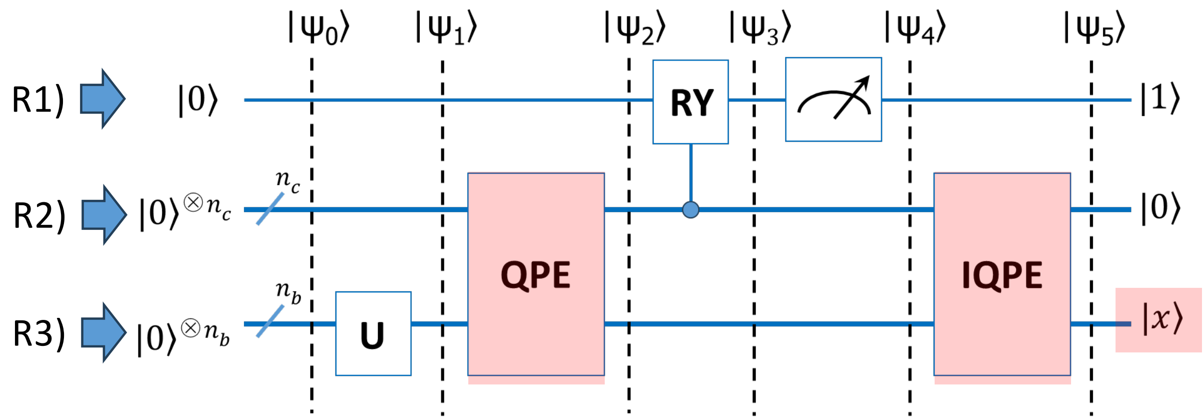
- Theoretically speaking, the HHL algorithm achieves an exponential improvement compared to the classical algorithm to solve the problem.
  - Quantum algorithm (HHL)

    $O(\log(N)\, s^2\kappa^2/\varepsilon)$
    - s is the sparsity: the maximum number of non-zero elements in any row
    - κ is the condition number: the ratio of the largest and the smallest eigenvalues
    - ε the precision (error)
  - Classical algorithm (Gaussian elimination)

    $O(N^3)$

Harrow-Hassidim-Lloyd (HHL): general quantum circuit

- There are three registers
  - R1) Ancilla qubit
    - a single qubit
  - R2) the eigenvalues of A; (N x N)
    - n_c = N qubits
  - R3) the encoded value of $|b\rangle$
    - n_c = $\log_2$N qubits



$$A\vec{x} = \vec{b} \Rightarrow A|x\rangle = |b\rangle$$

Quantum Support Vector Machine (QSVM)

Quantum Support Vector Machine (QSVM) with HHL

PRL 113, 130503 (2014) PHYSICAL REVIEW LETTERS week ending 26 SEPTEMBER 2014

Quantum Support Vector Machine for Big Data Classification

Patrick Rebentrost,[1,*] Masoud Mohseni,[2] and Seth Lloyd[1,3,†]
[1]Research Laboratory of Electronics, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA
[2]Google Research, Venice, California 90291, USA

- A key idea of the paper is to employ the least-squares reformulation of the SVM that avoids the quadratic programming and obtains the parameters from the solution of a linear equation system.

$$F\begin{pmatrix} b \\ \vec{\alpha} \end{pmatrix} \equiv \begin{pmatrix} 0 & \vec{1}^T \\ \vec{1} & K + \gamma^{-1}\mathbb{1} \end{pmatrix} \begin{pmatrix} b \\ \vec{\alpha} \end{pmatrix} = \begin{pmatrix} 0 \\ \vec{y} \end{pmatrix}. \quad (5)$$

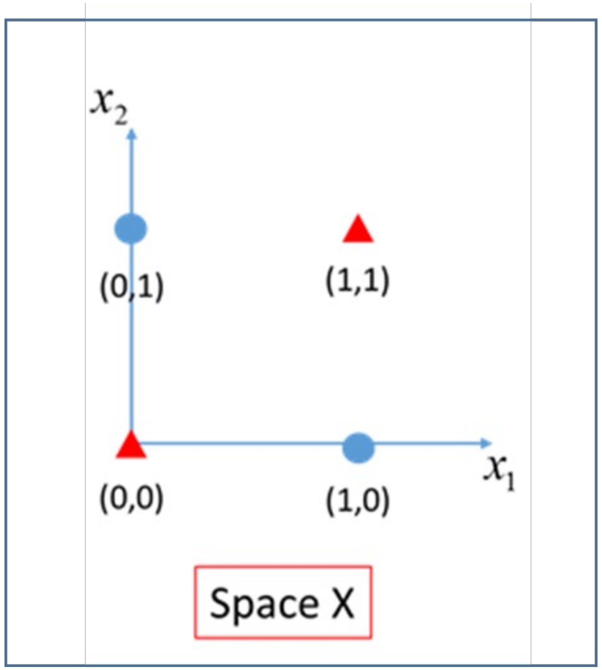
  - Then, we can apply the Harrow-Hassidim-Lloyd (HHL) algorithm to handle SVM, which is called "QSVM"!!

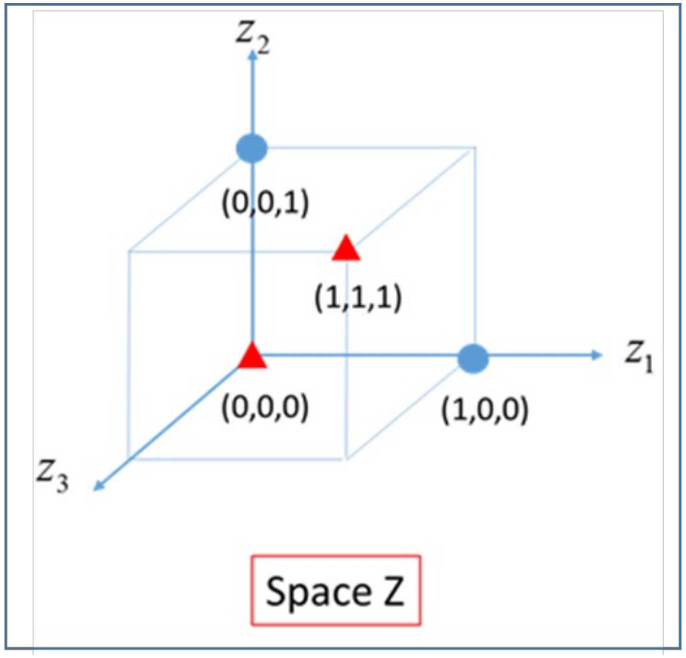
Quantum Support Vector Machine (QSVM) with QKernel

- We formulated SVM as a quadratic programming before…

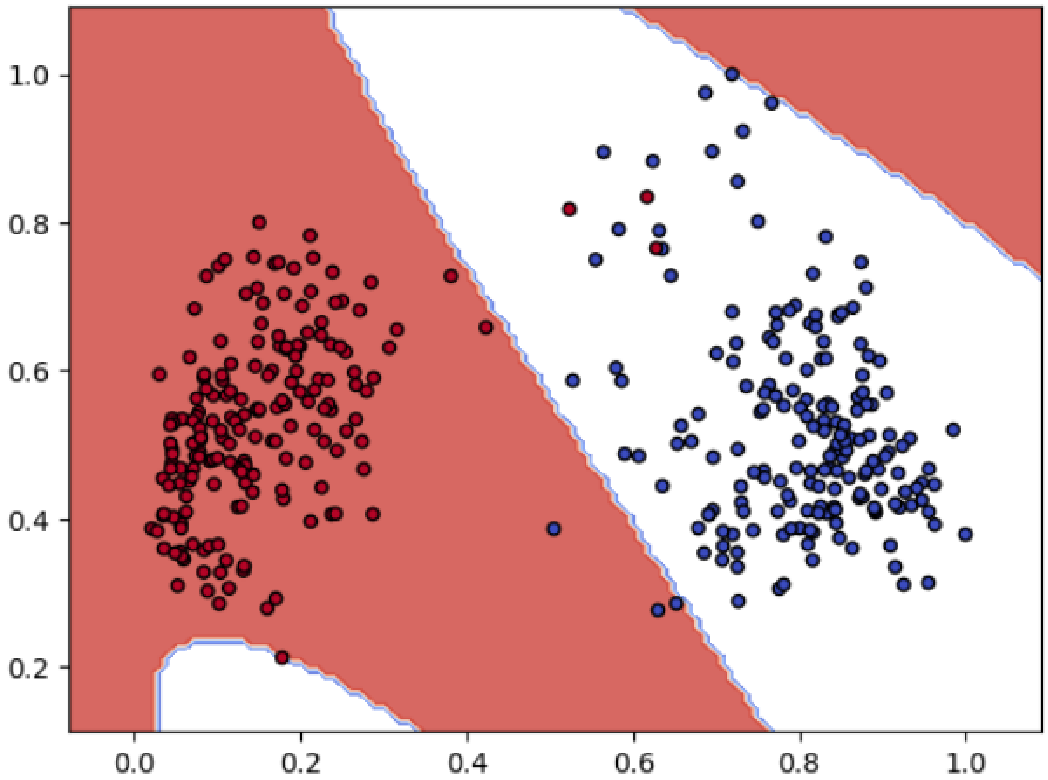
$$\min_{\lambda} \quad L(\lambda) = \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} t_n t_m \lambda_n \lambda_m \mathrm{K}(\mathrm{x}_n^T \mathrm{x}_m) - \sum_{n=1}^{N} \lambda_n$$

$$s.t. \quad \lambda \geq 0, \qquad t^T \lambda = 0$$

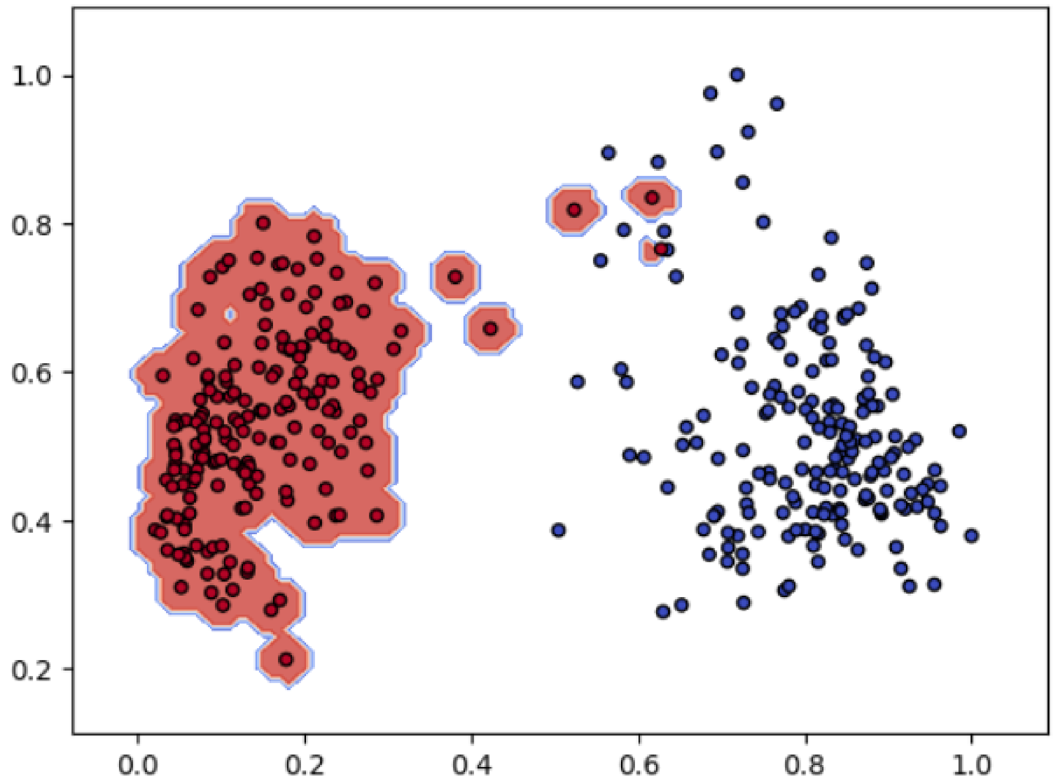




Quantum Support Vector Machine (QSVM)



Quantum Kernel result



RBF Kernel result

Summary

- The QPE was explained in detail.
- It is the building block of many quantum applications, making it essential to understand their operations.
- QSVM was briefly introduced to demonstrate how QPE and HHL can serve as such building blocks.
- In the last two lectures, we have only just begun to explore the surface of quantum mechanics. I hope you have gained a solid foundation to explore this topic further on your own in the future.