COE 466 Quantum Architecture and Algorithms

Lecture 23

HHL Algorithm

References:

"A Step-by-Step HHL Algorithm Walkthrough to Enhance Understanding of Critical Quantum Computing Concepts " https://arxiv.org/pdf/2108.09004
Quirk simulation for HHL
https://github.com/Qiskit/textbook/blob/main/notebooks/ch-applications/hhl_tutorial.ipynb

Solving Linear Systems

- The problem of solving a linear system of M equations with N variables is an important problem in mathematics, science, and engineering
- *Problem:* Given an $M \times N$ matrix A and a solution vector $\boldsymbol{b}$, find a vector $\boldsymbol{x}$ such that

$$Ax = b$$

- *Solution:*

Solve for x

$$x = A^{-1} b$$

How to find A^{-1}?

$O(N^3)$ -Worst case

$O(N)$

Solving Linear Systems - Example

$A = \begin{pmatrix} 1 & \frac{-1}{3} \\ \frac{-1}{3} & 1 \end{pmatrix}, \; b = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \; \vec{x} = ?$

① $x = A^{-1} b$ using A^{-1}

② $\overset{A}{\begin{pmatrix} 1 & \frac{-1}{3} \\ \frac{-1}{3} & 1 \end{pmatrix}} \overset{x}{\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}} = \overset{b}{\begin{pmatrix} 0 \\ 1 \end{pmatrix}}$ $\quad x = \begin{pmatrix} \frac{3}{8} \\ \frac{9}{8} \end{pmatrix}$

$x_1 - \frac{x_2}{3} = 0 \Rightarrow x_1 = \frac{x_2}{3} \Rightarrow x_1 = \frac{3}{8}$

$\frac{-x_1}{3} + x_2 = 1 \Rightarrow -\frac{x_2}{9} + x_2 = 1$

$x_2 = \frac{9}{8}$

For HHL to work, we assume

- A is Hermitian ($A = A^\dagger$)
- A is s-sparse (it has s non-zero entries in cols & rows)

Quantum Linear Systems Problem (QLSP)

A is $N \times N$ where $N = 2^{n_b}$

Convert $Ax = b$ to quantum states

$|x\rangle$ is the solution

we can write in terms of its eigenvector using spectral decompisition

$$A = \sum_{i=0}^{2^{n_b}-1} \lambda_i |u_i\rangle\langle u_i|$$

$$|b\rangle = \sum_{i=0}^{2^{n_b}-1} b_j |u_j\rangle$$

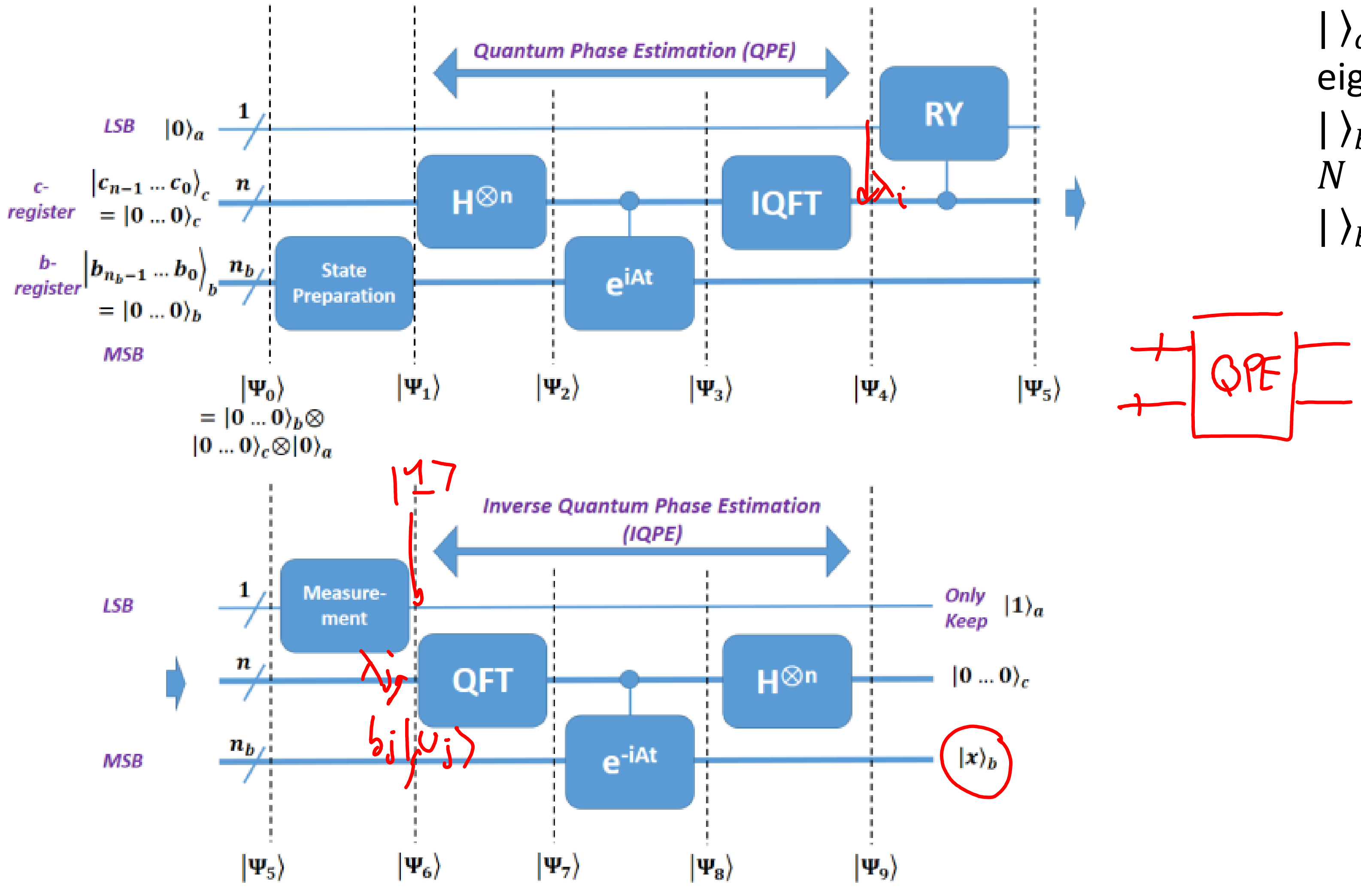
$$|x\rangle = A^{-1}|b\rangle = \sum_{i=0}^{2^{n_b}-1} \lambda_i^{-1} |u_i\rangle\langle u_i| \sum_{j=0}^{2^{n_b}-1} b_j |u_j\rangle$$

$$|x\rangle = \sum_{i=0}^{2^{n_b}-1} \lambda_i^{-1} b_i |u_i\rangle$$

QPE

$$U|v\rangle = e^{i\theta t}|v\rangle$$

Math



IGURE 1. Schematic of the *HHL* quantum circuit flowing from left to right. The circuit is decomposed into top and bottom portions for clarity. Note that he lowest qubit in the diagram is the most significant bit (*MSB*) while the top one is the least significant bit (*LSB*).

Quantum registers:

$|\,\rangle_c$: store binary representation of the eiganvalues of A- also called clock register

$|\,\rangle_b$:contains the vector solution, where $N = 2^{n_b}$

$|\,\rangle_b$:is auxiliary qubit

Math

$$|\varphi_0\rangle = |0\ldots0\rangle_b\,|0\ldots0\rangle_c\,|0\rangle_a$$

$$= |0\rangle^{\otimes n_b}\,|0\rangle^{\otimes n_c}\,|0\rangle$$

$$|\varphi_1\rangle = |b\rangle\,|0\rangle^{\otimes n}\,|0\rangle \qquad \vec{b} = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_{n_b-1} \end{pmatrix} = \beta_0|0\rangle + \beta_1|1\rangle + \cdots + \beta_{n_b-1}|n_b-1\rangle$$

$$H^{\otimes n_c}$$

$$|\varphi_2\rangle = |b\rangle\,\frac{1}{2^n}\left(|0\rangle+|1\rangle\right)^{\otimes n}|0\rangle$$

Assume that $U := e^{iAt}$

$$|\varphi_3\rangle = \ldots$$

$$|\varphi_4\rangle = \sum_{j=0}^{2^{n_b}-1} b_j|u_j\rangle|\tilde{\lambda}_j\rangle|0\rangle$$

Math

$$|\varphi_5\rangle = \sum_{j=0}^{2^{n_b}-1} b_j |u_j\rangle |\tilde{\lambda}_j\rangle \left(\sqrt{1-\frac{C^2}{\tilde{\lambda}_j^2}}\,|0\rangle_a + \frac{C}{\tilde{\lambda}_j}|1\rangle_a\right)$$

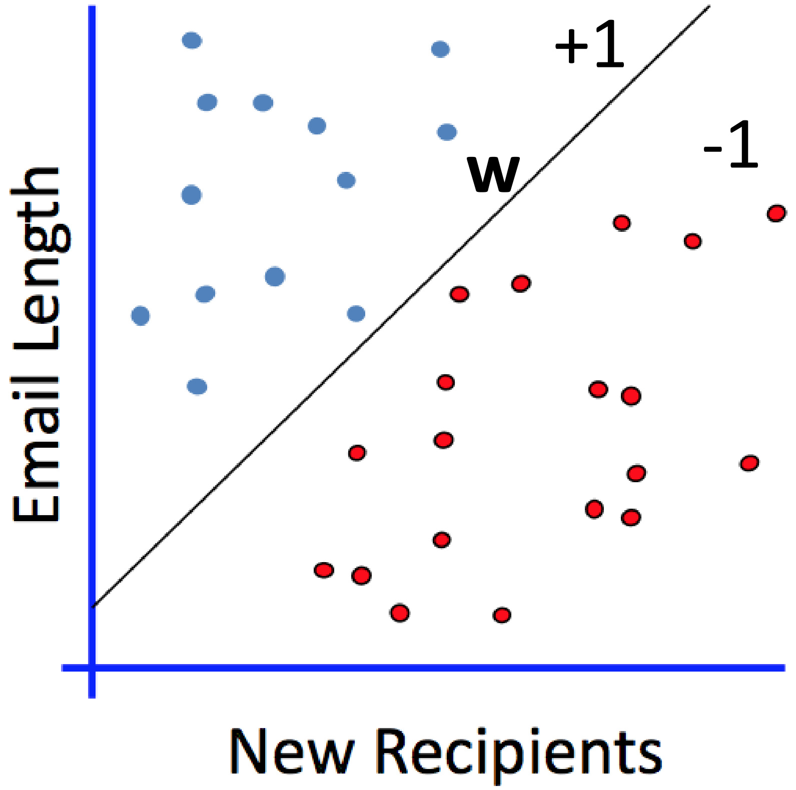
$$|\varphi_6\rangle = \frac{1}{\sqrt{\sum_{j=0}^{2^{n_b}-1}\left|\frac{b_j C}{\tilde{\lambda}_j}\right|^2}} \sum_{j=0}^{2^{n_b}-1} b_j |u_j\rangle |\tilde{\lambda}_j\rangle \frac{C}{\tilde{\lambda}_j}|1\rangle$$

Support Vector Machine

- Find a *linear function* to separate the classes:

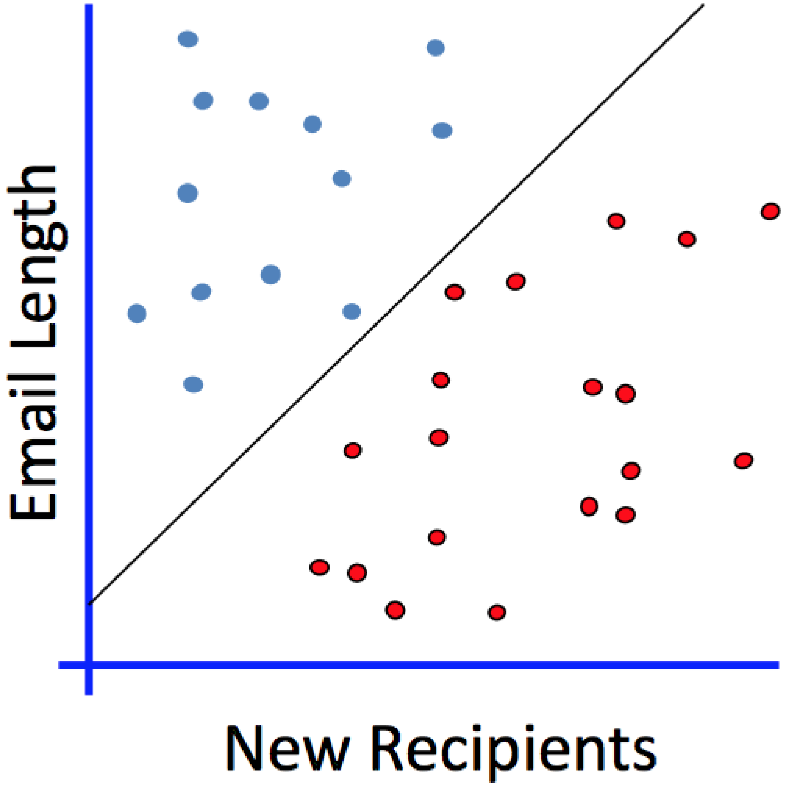
$$f(\mathbf{x}) = \text{sign}(\mathbf{w} \cdot \mathbf{x} + b)$$

	Email Length	New Recipients	Spam?
Email 1	120	100	Yes
Email 2	40	1	No
Email 3	60	2	No
Email 4	30	3	No
Email 5	240	400	Yes

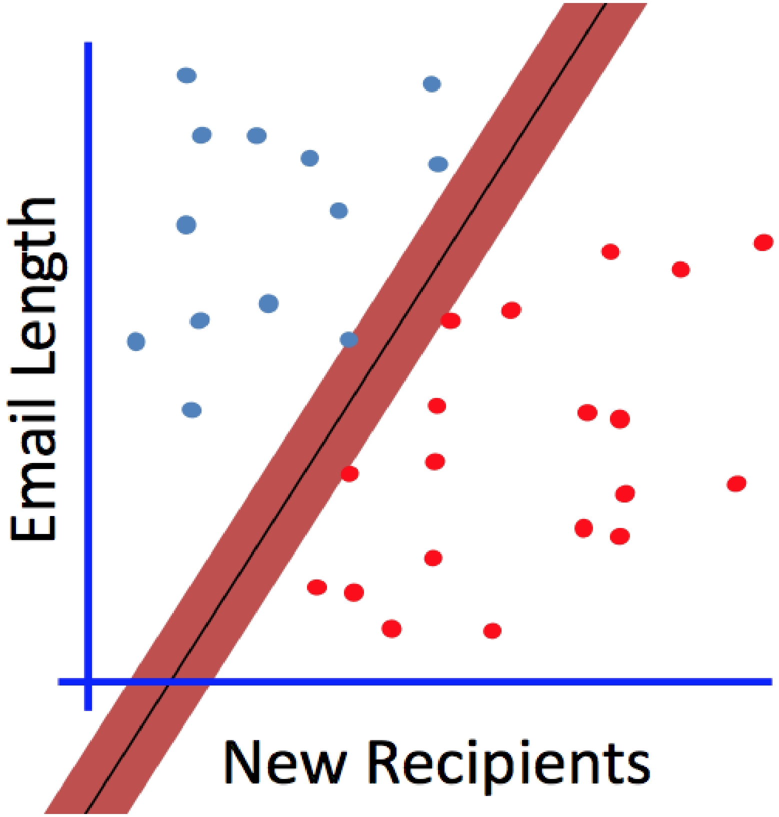


SVM Problem

How to find w?



The **best** line is the one that maximizes the margin between the two classes



Optimal Hyperplane

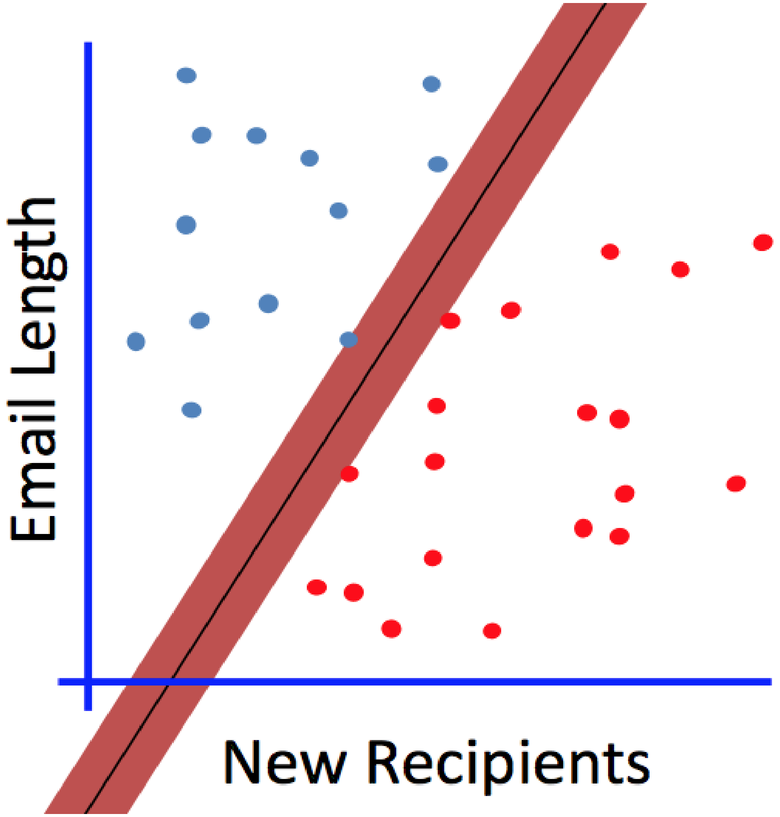
The **best** line is the one that maximizes the margin between the two classes

minimize:

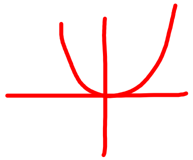
$$W(\alpha) = -\sum_{i=1}^{\ell} \alpha_i + \frac{1}{2} \sum_{i=1}^{\ell} \sum_{j=1}^{\ell} y_i y_j \alpha_i \alpha_j \boxed{\mathbf{x}_i \mathbf{x}_j}$$

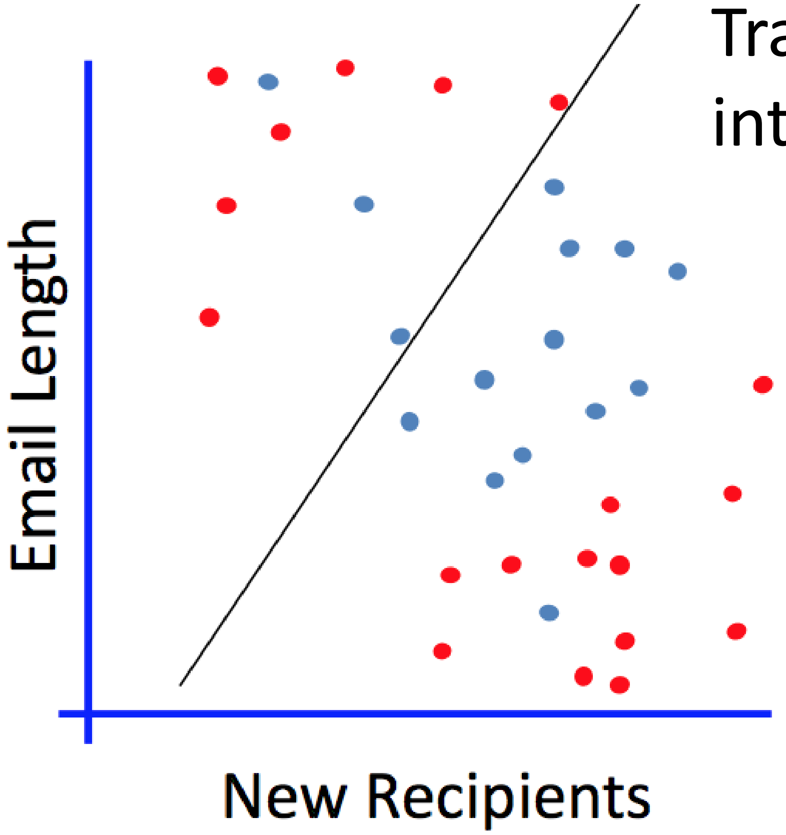
subject to:

$$\sum_{\substack{i=1 \\ 0 \le \alpha_i \le C}}^{\ell} y_i \alpha_i = 0 \qquad (4)$$

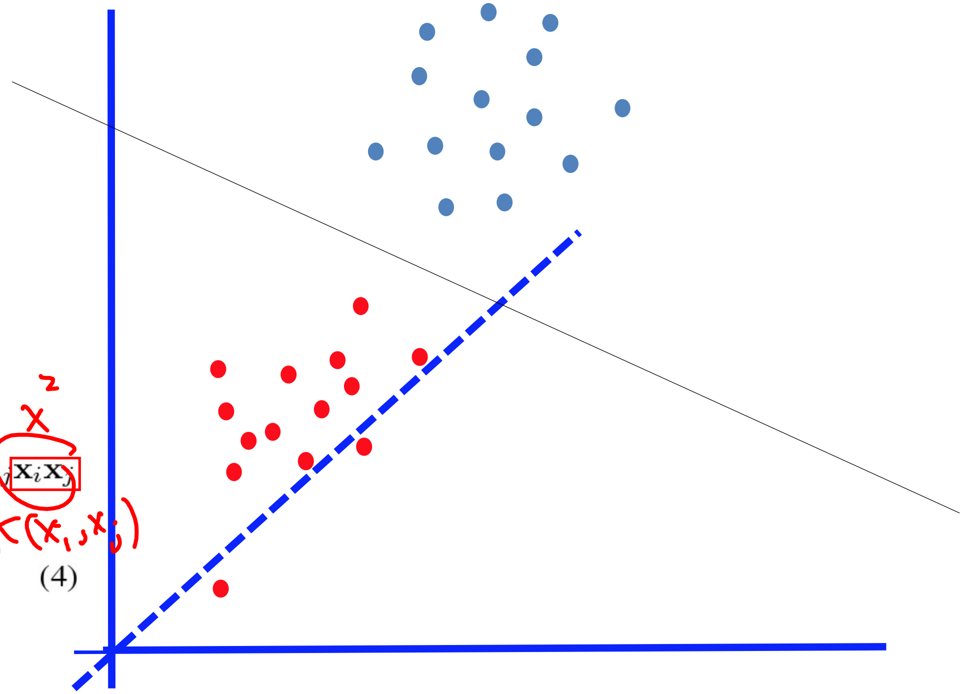


SVM: Not Linearly Separable





Transform the dataset
into higher dimensions



minimize:

$$W(\alpha) = -\sum_{i=1}^{\ell} \alpha_i + \frac{1}{2}\sum_{i=1}^{\ell}\sum_{j=1}^{\ell} y_i y_j \alpha_i \alpha_j \boxed{\mathbf{x}_i \mathbf{x}_j}$$

x^2

$K(x_i, x_j)$

subject to:

$$\sum_{\substack{i=1 \\ 0 \le \alpha_i \le C}}^{\ell} y_i \alpha_i = 0 \qquad (4)$$

$K(\mathbf{x},\mathbf{y}) = (\mathbf{x}\cdot\mathbf{y}+1)^p$ Polynomial

$K(\mathbf{x},\mathbf{y}) = \exp\left\{-\|\mathbf{x}-\mathbf{y}\|^2 / 2\sigma^2\right\}$ Gaussian (Radial basis)

$K(\mathbf{x},\mathbf{y}) = \tanh(\kappa\mathbf{x}\cdot\mathbf{y}-\delta)$ Sigmoid

QSVM

$$W(\alpha) = -\sum_{i=1}^{\ell}\alpha_i + \frac{1}{2}\sum_{i=1}^{\ell}\sum_{j=1}^{\ell} y_i y_j \alpha_i \alpha_j \boxed{\mathbf{x}_i \mathbf{x}_j}$$

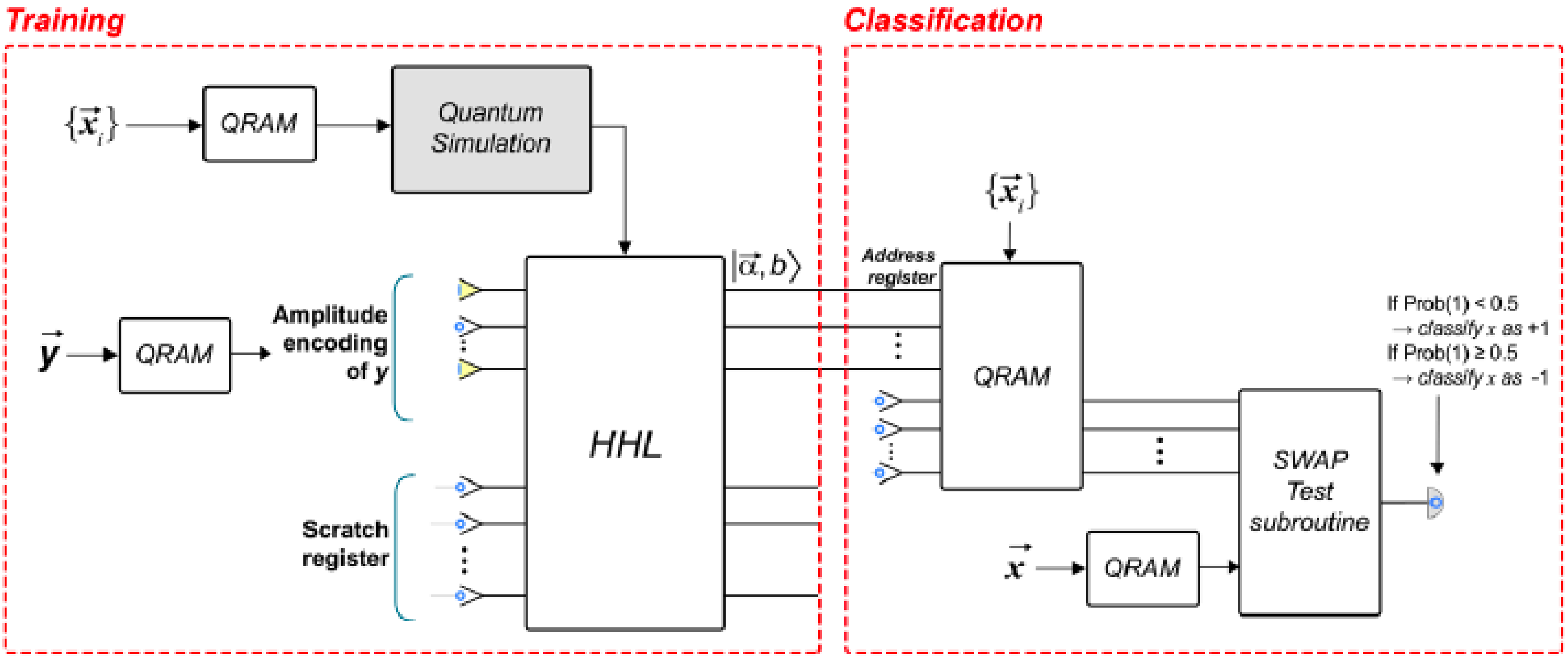
LS-SVM

$$\overset{A}{\begin{bmatrix} 0 & \mathbb{1}_N^T \\ \mathbb{1}_N & \Omega + \gamma^{-1} I_N \end{bmatrix}} \overset{x \downarrow \text{bias}}{\begin{bmatrix} b \\ \alpha \end{bmatrix}} = \overset{b}{\begin{bmatrix} 0 \\ y \end{bmatrix}}$$

$$x = A^{-1} b$$

$$\Omega = \phi(x_2)\phi(x_1) = \underline{k(x_1, x_2)} = x_i x_j$$

QSVM



- [Example: QSVM]()

Quantum Encoding

- Almost all QML algorithms require loading data into quantum computers
- Data MUST BE encoded in qubits

Encoding pattern	Encoding	Req. qubits
BASIS ENCODING [13]	$x_i \approx \sum_{i=-k}^{m} b_i 2^i \mapsto \lvert b_m \dots b_{-k}\rangle$	$l = k+m$ per data-point
ANGLE ENCODING	$x_i \mapsto cos(x_i)\lvert 0\rangle + sin(x_i)\lvert 1\rangle$	1 per data-point
QUAM ENCODING [13]	$X \mapsto \sum_{i=0}^{n-1} \frac{1}{\sqrt{n}} \lvert x_i\rangle$	l
QRAM ENCODING	$X \mapsto \sum_{n=0}^{n-1} \frac{1}{\sqrt{n}} \lvert i\rangle \lvert x_i\rangle$	$\lceil \log n \rceil + l$
AMPLITUDE ENCODING [13]	$X \mapsto \sum_{i=0}^{n-1} x_i \lvert i\rangle$	$\lceil \log n \rceil$

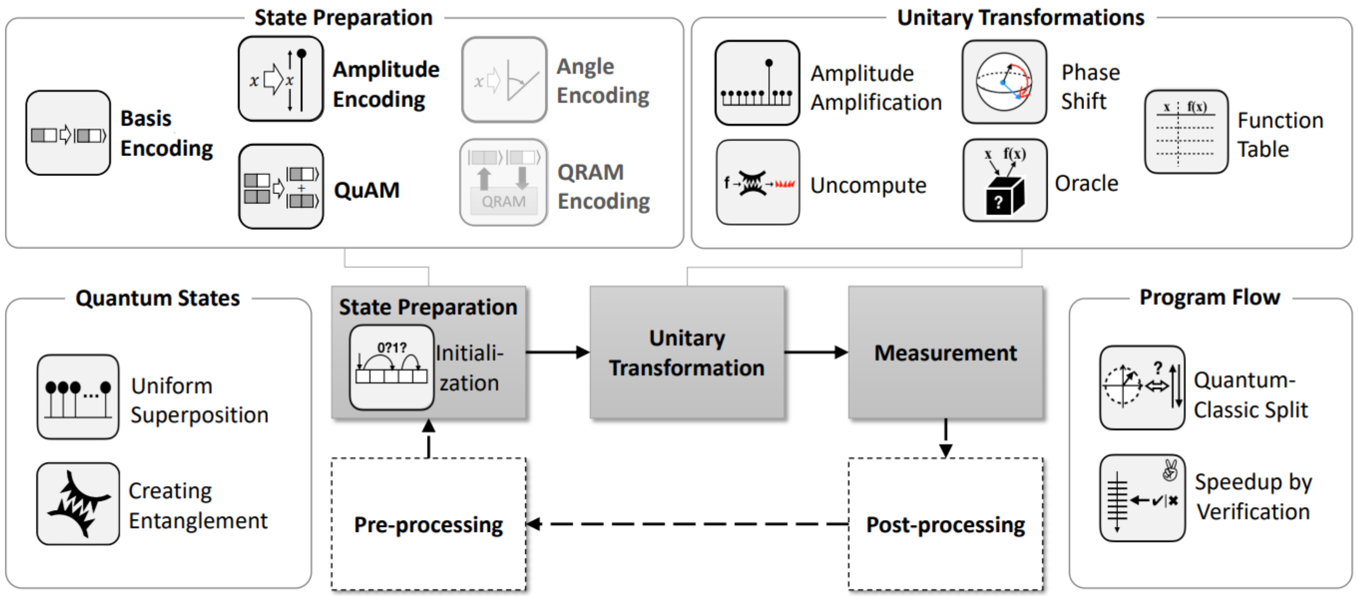


Fig. 1. Overview of pattern for quantum computing. In the center, the steps of a quantum algorithm are shown (based on [Leymann et al. 2020]). The new encoding patterns (highlighted in bold) are part of the first step that is executed on a quantum computer (State Preparation).

References

- [Quirk simulation for HHL]
- [PennyLane Optimization Demos]
- [Qiskit Aqua Algorithms]
- [LS-SVM]
- [Data Encoding Patterns for Quantum Computing]
- [QNN]