

CSCI-739 Quantum Machine Learning

Homework 2

You got this, team. These problems are not random puzzles—they are a curated path through the pillars of QML. You will see how noise shapes computation (and is a headache) (P1), why fault-tolerance is the gatekeeper for real QML at scale (P2), how data actually gets into a quantum computer (P3–P4), and how we can learn circuits rather than hand-design them (P5). If you can reason through these, you are on the road from “quantum intrigued” to “quantum capable”. I believe in you all! ☺

uhum inspirational speech

Quantum machine learning lives within quantum information sciences, which has a lot of puzzle pieces. The first four problem in this homework targets a puzzle piece we will need to understand for the future, with the fifth exposing you to how classical machine learning can benefit quantum computing research:

- *P1 (Noisy gates)*: connects linear algebra to real devices—how small coherent errors compound across layers and change measurement statistics.
- *P2 (Logical error rates)*: ties a subset of quantum machine learning algorithms we will eventually discuss(quantum support vector machines (QSVMs), quantum boltzmann machines (QBM), Quantum Neural Networks (QNNs), and Variational Quantum Algorithms (VQAs)) to the economics of fault tolerance—what distances and qubit counts make QML feasible.
- *P3–P4 (Encoding & state preparation)*: shows how precision and resources trade off when mapping continuous values (like π or e) to finite qubits and finite-depth circuits—core to quantum feature maps.
- *P5 (Bell States & using Reinforcement Learning to Construct Them)*: blends first-principles circuit synthesis with reinforcement learning to automate discovery under hardware-aware constraints.

Use lectures, notes, and eachother to conquer this homework, team! ☺

Homework Summary

This homework contains five standard problems and two bonus problems (25 points each). For this homework you can either choose to pick 4 standard problems worth 25 points each or 5 standard problems worth 20 points each. They build a coherent arc from noisy primitives to scalable QML.

1. **Noisy Hadamard Gates and Their Implications:** Unitarity proofs; repeated H action; a coherent-error model $U(\varepsilon) = He^{i\varepsilon\sigma_z}$; measurement consequences with/without noise.

2. **Logical Error Rates for QML Applications:** Surface-code scaling for QSVMs, QBMAs, QNNs, VQAs; compute d , N_q ; orders-of-magnitude gap vs. current hardware; interpret feasibility.
3. **Encoding π :** Basis, amplitude, and angle encodings; precision vs. resources; optional BBP hex-digit extractor for π .
4. **Encoding e :** Three encodings again, now with factorial-tail bounds to pick truncation for n correct digits; compare robustness and cost.
5. **Bell + RL Synthesis:** Construct minimal-depth Bell circuits from $|00\rangle$, define an RL objective with depth penalties, and compare learned vs. hand-derived circuits.

Bonus A. Lindblad, Two Analytic Cases: Solve the master equation for $\gamma = 0$ and $\gamma = 1$; long-time measurement probabilities.

Bonus B. Qiskit State Prep of π : Implement and verify the three encodings on a simulator for $n = 1, 2, 3$ decimal places.

Problem 1. Noisy Hadamard Gates and Their Implications (20 pts)

Summary. Analyze how a coherent Z -axis over-rotation modifies Hadamard behavior and measurement outcomes. This links linear-algebra fundamentals to realistic gate implementations.

Consider

$$U(\varepsilon) = H e^{i\varepsilon\sigma_z} = \frac{1}{\sqrt{2}} \begin{pmatrix} e^{i\varepsilon} & e^{-i\varepsilon} \\ e^{i\varepsilon} & -e^{-i\varepsilon} \end{pmatrix},$$

where $\varepsilon \in \mathbb{R}$ is an error parameter, $\sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$, and $H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$.

- (a) Show that H is unitary.
- (b) Prove that for any unitary U , $U^{-1} = U^\dagger$ (conjugate transpose).
- (c) Starting from $|0\rangle$, determine $H^n |0\rangle$ as a function of n ; distinguish even vs. odd n .
- (d) Show that $U(0) = H$.
- (e) Apply $U(\varepsilon)$ repeatedly n times to $|0\rangle$. Derive the resulting state in terms of n and ε .
- (f) Compare the probabilities of measuring $|0\rangle$ and $|1\rangle$ for (i) $\varepsilon = 0$ vs. (ii) $\varepsilon \neq 0$. Discuss how coherent error accumulates with depth.

Problem 2. Logical Error Rates for Quantum Machine Learning Applications (20 pts)

Summary. Connect QML application targets to surface-code scaling. Compute the code distance d and qubit overhead N_q needed to reach application-level logical error rates, then compare to today's hardware scale (10^2 – 10^3 physical qubits).

Parameters: code distance d (corrects $\lfloor (d-1)/2 \rfloor$ errors), physical error p , threshold p_{th} , logical error p_L , qubit count N_q . Models:

$$N_q \approx cd^2, \quad p_L \approx A \left(\frac{p}{p_{\text{th}}} \right)^{\frac{d+1}{2}},$$

with constants $A, c > 0$.

Application targets (context).

- **Kernel methods/QSVMs** ($p_L \leq 10^{-9}$): quantum feature maps embed data into high-dimensional Hilbert spaces; repeated kernel evaluations demand consistent outputs under noise.
- **Quantum Boltzmann Machines** ($p_L \leq 10^{-10}$): generative modeling via quantum superposition; training requires many samples/energy evaluations, making error accumulation critical.
- **Quantum Neural Networks** ($p_L \leq 10^{-11}$): deeper parameterized circuits for structured data; cumulative gate error makes strict p_L essential.
- **Variational Quantum Algorithms** ($p_L \leq 10^{-12}$): hybrid optimization highly sensitive to expectation-value corruption; demands the lowest p_L here.

(a) Analyze p_L as $d \rightarrow \infty$ for $p > p_{\text{th}}$ vs. $p < p_{\text{th}}$. Can p_L be made arbitrarily small?

(b) For each application and each parameter pair

$$p \in \{10^{-3}, 10^{-4}\}, \quad p_{\text{th}} \in \{10^{-2}, 10^{-3}\},$$

with $A = 1$ and $c = 2$, estimate:

- the minimum d required,
 - $N_q \approx 2d^2$,
 - and the number of correctable errors $\lfloor (d-1)/2 \rfloor$.
- (c) Compare each N_q to 100–1000 qubits. How many orders of magnitude larger is your estimate? Comment on near-term vs. long-term feasibility.
- (d) Reflect: Which application is likely to reach fault-tolerant feasibility first, and why? Justify using your d , N_q , and target p_L .

Problem 3. Quantum Data Encoding and State Preparation: Encoding π (integer + n decimals) (20 pts)

Summary. Encode the constant π using basis, amplitude, and angle encodings that include *both* the significant (integer) digit and the first n decimal digits. This forces you to reason about how full fixed-precision numbers are represented on quantum hardware and how resource costs scale with precision.

Let $\pi = 3.14159 \dots$. Write

$$\hat{\pi}^{(n)} = S_\pi + \sum_{i=1}^n \frac{d_i}{10^i}, \quad S_\pi = 3, \quad d_i \in \{0, 1, \dots, 9\}.$$

(Thus the digit list is $[S_\pi, d_1, d_2, \dots, d_n]$ with $n+1$ total digits.)

- (a) **Basis encoding (include integer + n decimals).** Use *binary-coded decimal (BCD)* to preserve the decimal digit structure. Map each digit $u \in \{0, \dots, 9\}$ to its 4-bit BCD block $\text{BCD}(u) \in \{0, 1\}^4$. Define the $(4(n+1))$ -qubit basis state

$$|\psi_{\text{basis}}^{(n)}\rangle = |\text{BCD}(S_\pi)\rangle \otimes \bigotimes_{i=1}^n |\text{BCD}(d_i)\rangle.$$

Example (for $n = 2$): $[3, 1, 4] \mapsto |0011\rangle \otimes |0001\rangle \otimes |0100\rangle = |0011\ 0001\ 0100\rangle$. State the qubit count $4(n+1)$ and depth needed (just X -initializations). *Alternative (optional):* a two-register binary scheme with an I -qubit integer register for S_π (here $I = 2$) and a $k(n) = \lceil n \log_2 10 \rceil$ -qubit fractional register holding the first $k(n)$ binary fractional bits of π . Do the basis encoding for $n = 3$

- (b) **Amplitude encoding (digits-as-amplitudes, including the integer digit).** Form a length- $m = n+1$ vector by scaling each decimal digit to $[0, 1]$:

$$x_0 = \frac{S_\pi}{9}, \quad x_i = \frac{d_i}{9} \quad (i = 1, \dots, n), \quad \tilde{x} = \frac{x}{\|x\|_2}.$$

Encode on $q = \lceil \log_2 m \rceil$ qubits:

$$|\psi_{\text{amp}}^{(n)}\rangle = \sum_{j=0}^{m-1} \tilde{x}_j |j\rangle \quad (\text{pad with zeros if } 2^q > m).$$

Work out \tilde{x} explicitly for $n = 3$ using the digits of π (i.e., $[3, 1, 4, 1]$) and report q .

- (c) **Angle encoding (single- and multi-qubit) for the full truncated value.** Let the full truncated value be $v^{(n)} = \hat{\pi}^{(n)}$. *Single-qubit full-value encoding:* choose any scale $C_\pi > v^{(n)}$ (e.g., $C_\pi = 4$) and set

$$\theta_n = 2 \arcsin\left(\frac{v^{(n)}}{C_\pi}\right), \quad |\psi_\theta^{(n)}\rangle = R_y(\theta_n)|0\rangle = \begin{bmatrix} \sqrt{1 - (v^{(n)}/C_\pi)^2} \\ v^{(n)}/C_\pi \end{bmatrix}.$$

Multi-qubit per-digit variant: allocate $n+1$ qubits, one per digit, with angles $\theta_0 = \alpha S_\pi / 9$ and $\theta_i = \alpha d_i / 9$ for $i = 1, \dots, n$, preparing $\otimes_{i=0}^n R_y(\theta_i)|0\rangle^{\otimes(n+1)}$. Comment on choosing α . Discuss how α trades dynamic range vs. noise sensitivity. Do it for $n = 3$

- (d) **Scaling and reflection.** Give qubit counts and a preparation-depth estimate for each scheme as a function of n . Comment on (i) robustness to small digit errors, (ii) which encoding is most efficient for large n .

Optional tool (unchanged): You may still use the BBP formula to extract *hex* digits of π if you opt for the binary two-register alternative for the fractional part.

Problem 4. Quantum Data Encoding and State Preparation: Encoding e to n digits (integer + n decimals) (20 pts)

Summary. Repeat the three encodings for e , now explicitly including the significant (integer) digit and the first n decimals. Use series-tail bounds to choose truncations guaranteeing n correct decimal places, and compare robustness and cost.

Let $e = 2.71828\dots$. Write

$$\hat{e}^{(n)} = S_e + \sum_{i=1}^n \frac{d_i}{10^i}, \quad S_e = 2, \quad d_i \in \{0, \dots, 9\}.$$

(Thus the digit list is $[S_e, d_1, d_2, \dots, d_n]$.)

- (a) **Basis encoding (include integer + n decimals).** Use BCD on each decimal digit to preserve decimal structure:

$$|\psi_{\text{basis}}^{(n)}\rangle = |\text{BCD}(S_e)\rangle \otimes \bigotimes_{i=1}^n |\text{BCD}(d_i)\rangle$$

Example (for n = 2): [2, 7, 1] \mapsto |0010 0111 0001). State the qubit count 4(n+1) and the initialization pattern. *Alternative (optional):* two-register binary scheme with $I = 2$ integer qubits for S_e and $k(n) = \lceil n \log_2 10 \rceil$ fractional binary bits. Do it for $n = 3$.

Amplitude encoding (digits-as-amplitudes, including the integer digit). Define $m = n+1$ and

$$x_0 = \frac{S_e}{9}, \quad x_i = \frac{d_i}{9} \quad (i = 1, \dots, n), \quad \tilde{x} = \frac{x}{\|x\|_2}.$$

Encode on $q = \lceil \log_2 m \rceil$ qubits:

$$|\psi_{\text{amp}}^{(n)}\rangle = \sum_{j=0}^{m-1} \tilde{x}_j |j\rangle \quad (\text{zero-pad to } 2^q \text{ if needed}).$$

Work out \tilde{x} explicitly for $n = 3$ using the digits [2, 7, 1, 8] and report q .

Angle encoding (single- and multi-qubit) for the full truncated value. Let $v^{(n)} = \hat{e}^{(n)}$. *Single-qubit full-value encoding:* choose any scale $C_e > v^{(n)}$ (e.g., $C_e = 3$) and set

$$\theta_n = 2 \arcsin\left(\frac{v^{(n)}}{C_e}\right), \quad |\psi_{\theta}^{(n)}\rangle = R_y(\theta_n) |0\rangle = \begin{bmatrix} \sqrt{1 - (v^{(n)}/C_e)^2} \\ v^{(n)}/C_e \end{bmatrix}.$$

Multi-qubit per-digit variant: $n+1$ qubits with $\theta_0 = \alpha S_e/9$ and $\theta_i = \alpha d_i/9$ for $i = 1, \dots, n$, preparing $\bigotimes_{i=0}^n R_y(\theta_i) |0\rangle^{\otimes(n+1)}$. Comment on choosing α .

Precision vs. resources (updated to full-value digits). Using $e = \sum_{k=0}^{\infty} \frac{1}{k!}$ and the bound $0 < e - \sum_{k=0}^m \frac{1}{k!} < \frac{1}{mm!}$:

- pick m to guarantee n correct decimal digits for the *full* value $\hat{e}^{(n)}$ (so that all d_1, \dots, d_n are correct),
- give qubit counts and preparation-depth estimates for each encoding as a function of n ,
- discuss robustness when one or more trailing digits are incorrect (impact on each encoding).

Problem 5. Bell States and Reinforcement Learning for Minimal-Depth Synthesis (20 pts)

Summary. Construct minimal-depth circuits that prepare each Bell state from $|00\rangle$, then set up an RL objective that discovers such circuits under a depth penalty. Compare learned circuits to your hand designs. Consider the four Bell States:

Targets (Bell states):

$$|\Phi^{\pm}\rangle = \frac{1}{\sqrt{2}}(|00\rangle \pm |11\rangle), \quad |\Psi^{\pm}\rangle = \frac{1}{\sqrt{2}}(|01\rangle \pm |10\rangle).$$

- (a) **Synthesis from $|00\rangle$** Using only either all or some subset of gates in $\{X, Y, Z, H, \text{CNOT}\}$, propose a circuit for each Bell state, prove correctness, report depth and CNOT count, and argue minimality (or provide a lower-bound justification e.g. if you found a solution with depth 3, then atleast you know the potential smallest circuit depth is 3, but it could be lower potentially).
- (b) **RL environment.** Define an MDP: state representation, action set $\{X_0, X_1, Y_0, Y_1, Z_0, Z_1, H_0, H_1, \text{CNOT}_{0\rightarrow 1}, \text{CNOT}_{1\rightarrow 0}\}$, transition, termination.
- (c) **Objective with depth penalty.** For target $|\phi\rangle$, use fidelity $F = |\langle\phi|\psi\rangle|^2$ and terminal reward $R(\tau) = F_T - \lambda L - \kappa C$ (depth L , CNOT count C). Optimize $J(\theta) = R(\theta)$. You may add shaping if you are a reinforcement learning wizard: $r_t = F_t - F_{t-1} - \eta$, but it is not required \odot .
- (d) **Tooling (suggested).** qiskit, qiskit-aer for simulation/fidelity; gymnasium + stable-baselines3 or RLlib for RL.
- (e) **Hand vs. RL comparison.** For each Bell state, report your hand circuit and the best RL circuit (gate list, L , C , F_T). Compare depth/CNOTs, structural equivalence (up to local Cliffords/commutations/global phase), sensitivity to (λ, κ) .

Bonus A. Lindblad Master Equation: Two Analytic Cases (25 pts)

Summary: The Lindblad master equation describes open quantum systems and their interactions with the environment. The mathematical objects involved are:

- ρ - The density matrix representing the quantum state (qubit) for a 2 level system, given by:

$$\rho(t) = \begin{bmatrix} \rho_{00}(t) & \rho_{01}(t) \\ \rho_{10}(t) & \rho_{11}(t) \end{bmatrix},$$

where ρ_{00} and ρ_{11} represent the probabilities of measuring the system in the $|0\rangle$ and $|1\rangle$ states, respectively, and ρ_{01} and ρ_{10} represent coherence terms. In general this matrix is a $N \times N$ for a N level system

- γ - The decay rate associated with spontaneous emission.
- σ_- and σ_+ - The lowering and raising operators, respectively, defined as:

$$\sigma_- = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad \sigma_+ = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}.$$

- σ_z - The Pauli-Z matrix:

$$\sigma_z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}.$$

- $[A, B] = AB - BA$ - The commutator.
- $\{A, B\} = AB + BA$ - The anti-commutator.
- for a N level system ρ_{jj} is the probability at measurement you are in state $|j\rangle$.

Initial Condition for an Uncertain State: If the system is initially in an uncertain state, where it is in $|0\rangle$ with probability p_0 and in $|1\rangle$ with probability p_1 (such that $p_0 + p_1 = 1$), the initial density matrix is given by:

$$\rho(0) = p_0|0\rangle\langle 0| + p_1|1\rangle\langle 1| = \begin{bmatrix} p_0 & 0 \\ 0 & p_1 \end{bmatrix},$$

where $|i\rangle\langle i|$ is an outer product. The **outer product** of a quantum state $|\psi\rangle$ with itself, denoted as $|\psi\rangle\langle\psi|$, is computed as follows:

1. Given a quantum state in column vector form:

$$|\psi\rangle = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix}$$

2. Compute its conjugate transpose (bra):

$$\langle\psi| = [a_1^* \quad a_2^* \quad \cdots \quad a_n^*]$$

3. The outer product is given by:

$$|\psi\rangle\langle\psi| = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} [a_1^* \quad a_2^* \quad \cdots \quad a_n^*]$$

resulting in an $n \times n$ Hermitian matrix:

$$|\psi\rangle\langle\psi| = \begin{bmatrix} a_1a_1^* & a_1a_2^* & \cdots & a_1a_n^* \\ a_2a_1^* & a_2a_2^* & \cdots & a_2a_n^* \\ \vdots & \vdots & \ddots & \vdots \\ a_na_1^* & a_na_2^* & \cdots & a_na_n^* \end{bmatrix}$$

Properties for the math nerds who are interested and want to do some further research:

- $|\psi\rangle\langle\psi|$ is a **projection operator**.
- It is **Hermitian**: $(|\psi\rangle\langle\psi|)^\dagger = |\psi\rangle\langle\psi|$.
- If $|\psi\rangle$ is normalized, $|\psi\rangle\langle\psi|$ represents a **pure state density matrix**.

For example, if there is uncertainty in the initial quantum state (or superposition) the quantum system is in e.g. $|0\rangle$ 50% of the time and in $|1\rangle$ 50% of the time, then:

$$\rho(0) = \frac{1}{2} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix}.$$

This represents a completely mixed state, indicating maximal uncertainty in the initial preparation.

Solve analytically the Lindblad equation for a single qubit in two cases:

$$\frac{d\rho}{dt} = -i[H, \rho] + \gamma(\sigma_- \rho \sigma_+ - \frac{1}{2}\{\sigma_+ \sigma_-, \rho\}) \quad (1)$$

where $H = \frac{1}{2}\sigma_z$,

- (a) **Case $\gamma = 0$:** Solve for $\rho(t)$, verify $\rho_{00}(t) + \rho_{11}(t) = 1$, and find $\lim_{t \rightarrow \infty} \rho_{00}(t)$ and $\rho_{11}(t)$. Explain physically.
- (b) **Case $\gamma = 1$:** Solve the coupled ODEs for $\rho(t)$, verify $\rho_{00}(t) + \rho_{11}(t) = 1$, and compute the $t \rightarrow \infty$ limits of $P_0(t) = \rho_{00}(t)$ and $P_1(t) = \rho_{11}(t)$ with interpretation.

Bonus B. State Preparation of π to n Digits in Qiskit (25 pts)

Summary. Implement and verify basis, amplitude, and angle encodings of π on a simulator for $n = 1, 2, 3$ decimal places. Confirm that the prepared state matches the analytic target with high fidelity. Hint: For this homework, I have selected the qubit order to match the binary representation, but during readout, the output will be inverted in the register. Therefore, simply reverse the register's output to realign it with the binary representation, if needed, based on how you do it ☺.

Conventions: $\hat{\pi}^{(n)}$ is π truncated to n decimals; $f^{(n)} = \hat{\pi}^{(n)} - 3$.

- (a) **Basis:** $k(n) = \lceil n \log_2 10 \rceil$ binary digits \Rightarrow target $|b_1 \dots b_{k(n)}\rangle$. Prepare with X gates and verify via statevector fidelity (see mycourses example program under lecture 4 in content tab).
- (b) **Amplitude:** digits $d_1 \dots d_n \mapsto x_i = d_i/9$, normalize, encode on $q = \lceil \log_2 n \rceil$ qubits (pad if needed). Verify fidelity.
- (c) **Angle:** one-qubit state $R_y(\theta_n)|0\rangle$ with $\theta_n = 2 \arcsin f^{(n)}$. Verify against analytic amplitudes.
- (d) **Report:** qubit counts, circuit depth, and a table of fidelities for all 3×3 cases. Briefly compare resource/accuracy trade-offs across encodings.