Classical-to-Quantum Encoding Strategies for Binary Classification

Ksenia Shileeva (s3971449)

Supervisors:
Evert van Nieuwenburg & Felix Frohnert

Bachelor Thesis– Data Science and Artificial Intelligence

Leiden Institute of Advanced Computer Science (LIACS)
liacs.leidenuniv.nl

July 8, 2026

Abstract

This thesis investigates how classical data is encoded into quantum states for binary classification tasks, comparing five encoding strategies across two quantum model families. The encodings, namely, the angle encoding, amplitude encoding, data re-uploading, instantaneous quantum polynomial (IQP), and ZZ feature map, are evaluated on a balanced subset of ECG-derived heart rate variability features, with rhythm abnormality as the binary target. Quantum kernel support vector machines and variational quantum classifiers (VQCs) are compared against classical baselines (RBF-SVM, logistic regression, multilayer perceptron). The results establish that encoding choice determines not only classification performance but also trainability, with optimal strategies differing between kernel and variational settings. Data re-uploading is the most consistent strategy across both model families, whereas strong kernel geometry does not guarantee strong variational performance.

Contents

1 Introduction

Quantum machine learning (QML) aims to exploit the exponential state space of quantum systems for learning tasks that are hard to represent classically [19, 8]. A quantum speed-up for specially constructed problems in supervised learning has been established [22]; on ordinary classical data an advantage over strong classical models is not guaranteed, and grafting quantum structure onto classical techniques without exploiting quantum phenomena can leave performance unchanged or degrade it [31]. Thus whether and when QML advances on real data remains open [10]. This turns the near-term question into a design problem: classical data must be translated into a quantum state before any quantum model can act on it [33], and this translation is not unique. Different encoding strategies, for example single-qubit rotations, repeated data injection, or entangling cross-terms, map the same classical features to quantum states in fundamentally different ways. The choice is important, as it fixes the geometry of the resulting feature space, which determines what a quantum classifier can and cannot separate. It also affects whether the associated variational circuit remains trainable, and how many qubits and gates the encoding requires. Because these effects are difficult to predict analytically and often interact, identifying a good encoding for a given task requires systematic empirical comparison.

This thesis addresses the following research question: "How do classical-to-quantum encoding strategies influence binary classification performance, feature-space geometry, and trainability under realistic resource constraints?" The question is investigated through five encoding strategies applied to electrocardiogram (ECG)-derived heart rate variability (HRV) features for rhythm abnormality detection, which is a clinically relevant binary classification task with a well-defined input representation.

Two quantum model families are examined. Quantum kernel support vector machines (QSVMs) use encodings to define a fixed similarity measure. The second family, variational quantum classifiers (VQCs), combine input encodings with trainable parameters optimized via gradient descent. The encodings are evaluated against classical baselines across four experimental phases, measuring classification performance, kernel geometry diagnostics, gradient variance, and resource scaling.

2 Background

This chapter establishes the concepts needed to interpret the experimental chapters. Section 2.1 introduces the learning task and classical solutions; Section 2.2 describes the input representation; Section 2.3 gives basic foundations of quantum computing; and Section 2.4 develops the quantum machine learning framework on which the experiments depend, including the encoding strategies that are the central object of study and the two failure modes the experiments are designed to detect.

2.1 Supervised Binary Classification

In supervised learning, an algorithm is given a dataset of labelled examples $\mathcal{D} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$, where each input $\mathbf{x}_i$ lies in an input domain $\mathcal{X}$ and each label y_i lies in an output domain $\mathcal{Y}$. The goal is to learn a function $f : \mathcal{X} \to \mathcal{Y}$ that generalizes to inputs not seen during training. When $\mathcal{Y}$ is a finite set of categories the task is classification. This thesis addresses the binary case with

$\mathcal{Y} = \{0, 1\}$.
A classifier can be understood in one of two equivalent ways: as producing an estimated class-one probability $\hat{p}(y = 1 \mid x) \in [0, 1]$, with the predicted label obtained by thresholding at 0.5, or as producing a real-valued decision function $f(x)$ whose sign gives the predicted label. The three baseline models introduced in 2.1.1 take one view or the other.
A classifier partitions $\mathcal{X}$ into regions and assigns a label to every point; the surface between regions is called a decision boundary. A linear classifier draws a hyperplane: a flat boundary in the feature space. Real-world data are frequently not separable by a hyperplane, so a nonlinear boundary is required. One way to obtain a nonlinear boundary without changing the classifier itself is to first map the input into a higher-dimensional feature space in which a hyperplane can separate the classes. Raising the dimension can create a separating hyperplane where none existed, because the added coordinates give the classifier further directions along which to place a flat boundary; that flat boundary, viewed back in the original space, appears curved. The linear method therefore produces the nonlinear boundary the data require, with the nonlinearity carried by the map. This map is called a feature map. The quantum feature maps studied in this work are one instance of this idea, as an encoding maps classical inputs into a higher-dimensional quantum feature space in which linear separation by a QSVM or VQC becomes possible.

2.1.1 Classical Baseline Models

Three classical models provide reference points for the quantum machine learning experiments considered here: logistic regression, a support vector machine with a radial basis function kernel (RBF-SVM), and a multilayer perceptron (MLP).
Logistic regression is a linear model that produces a probability directly. For a feature vector $\mathbf{x} \in \mathbb{R}^d$, it predicts the class-one probability through the logistic sigmoid $\sigma(z) = (1 + e^{-z})^{-1}$:

$$p(y = 1 \mid \mathbf{x}) = \sigma\big(\mathbf{w}^\top \mathbf{x} + b\big), \tag{1}$$

where the weight vector $\mathbf{w} \in \mathbb{R}^d$ and bias b are the model's parameters, chosen to minimize the regularized cross-entropy over the training set. The decision boundary $\mathbf{w}^\top \mathbf{x} + b = 0$ is a hyperplane, so logistic regression separates the classes only to the extent that they are linearly separable in the original feature space.
The RBF-SVM instead produces a decision function rather than a probability. It is a classifier that acts in an implicit feature space of higher dimension than the input, separating the two classes with the hyperplane of maximum margin: the boundary whose distance to the nearest training point of either class is as large as possible [4]. The data enters this optimization only through inner products between feature vectors, so the explicit map into the high-dimensional space is never formed. A kernel function that returns those inner products is enough. Pairwise similarity is given by the radial basis function (Gaussian) kernel

$$k(\mathbf{x}, \mathbf{z}) = \exp\big(-\gamma \, \|\mathbf{x} - \mathbf{z}\|^2\big), \qquad \gamma > 0, \tag{2}$$

whose feature space is infinite-dimensional, so the induced boundary is nonlinear. Collecting these values into the *kernel matrix* $K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$, a standard SVM training procedure fits the classifier using K alone, without forming the feature vectors explicitly [5], and predicts $f(\mathbf{x}) = \operatorname{sign}\big(\sum_i \alpha_i y_i \, k(\mathbf{x}_i, \mathbf{x}) + b\big)$, where the coefficients α_i are fixed by training. Replacing the

RBF kernel with one produced by a quantum circuit is the basis of the quantum kernel models studied in 2.4.1.
The MLP, like logistic regression, produces a probability. It is a feedforward neural network with a single hidden layer that composes affine maps with an elementwise nonlinearity. Writing $a^{(0)} = x$, the hidden layer applies

$$a^{(1)} = \phi(W^{(1)}x + b^{(1)}), \quad (4)$$

with ϕ a nonlinear activation ($\phi(z) = max(0, z)$), and the output layer returns

$$\hat{p}(y = 1 \mid x) = \sigma(W^{(2)}a^{(1)} + b^{(2)}). \quad (5)$$

The weights are trained by gradient descent on the cross-entropy loss. A feedforward network with a single hidden layer of sufficient width is, under mild conditions on the activation function, a universal approximator of continuous functions [15]; this is precisely the architecture used here.

2.2 Cardiac Rhythm and the Input Representation

Before the encoding strategies can be compared, the input they encode must be fixed. The input is a feature vector summarizing cardiac rhythm, extracted from short electrocardiogram recordings in a public database (Section 4.1). Because the representation is built from the timing between heartbeats, the section first describes how that timing is read from the signal. The electrocardiogram (ECG) records the electrical activity of the heart over time. Each heartbeat produces a sharp deflection, the R peak. Cardiac rhythm is read from the RR interval, the time between two successive R peaks. Long intervals correspond to a slow rate, short intervals to a fast rate, and irregular intervals to arrhythmia. Sinus node disorders, premature complexes, tachyarrhythmias, and atrioventricular blocks all appear as patterns in the RR-interval series [21], which is reduced to the seven heart rate variability (HRV) features that form the model input; their definitions and selection motivation are given in the Section 4.2.

2.3 Quantum Computing

This section introduces the quantum computing concepts required for the quantum models.

2.3.1 Qubits and Superposition

A classical bit is either 0 or 1. A quantum bit, or qubit, can be in a linear combination (superposition) of both. Its state is a unit vector in a two-dimensional complex space,

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle, \qquad |\alpha|^2 + |\beta|^2 = 1, \quad (3)$$

where $|0\rangle$ and $|1\rangle$ are the two basis states and $\alpha, \beta \in \mathbb{C}$ are probability amplitudes. When the qubit is measured in the basis $\{|0\rangle, |1\rangle\}$, it returns outcome 0 with probability $|\alpha|^2$ and outcome 1 with probability $|\beta|^2$; the normalisation constraint $|\alpha|^2 + |\beta|^2 = 1$ ensures these sum to one. After measurement the qubit collapses into the measured basis state. Until that point the pair (α, β) carries more information than either classical outcome [7].

2.3.2 The Bloch Sphere and Quantum Gates

A single-qubit state can be pictured as a point on the "Bloch sphere", a unit sphere whose north pole is $|0\rangle$ and south pole is $|1\rangle$; every other surface point is a superposition with a particular pair of amplitudes (Figure 1).

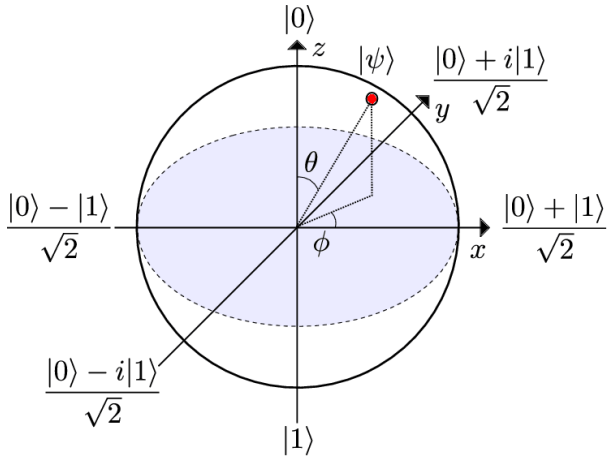

Figure 1: The Bloch sphere. Every pure single-qubit state is a point on the surface, with $|0\rangle$ at the north pole and $|1\rangle$ at the south pole. The equatorial states $(|0\rangle \pm |1\rangle)/\sqrt{2}$ and $(|0\rangle \pm i|1\rangle)/\sqrt{2}$ lie along the x and y axes. Picture taken from [11].

A quantum gate on one qubit is a unitary transformation, a 2×2 complex matrix that preserves $|\alpha|^2 + |\beta|^2 = 1$. Geometrically it rotates the Bloch sphere, moving the state from one vector to another without changing its length. The single-qubit gates used throughout this thesis are the rotations about the three axes,

$$R_x(\theta), \quad R_y(\theta), \quad R_z(\theta), \tag{4}$$

where θ is the rotation angle in radians. These gates are the building blocks of *data encoding*: a feature value x is encoded by setting the rotation angle to a function of x, for example $R_y(\pi x)$, so the qubit carries a state whose position on the sphere depends continuously on x. Because a rotation by θ is identical to a rotation by $\theta + 2\pi$, rotation gates are 2π-periodic. Continuous rotations of this kind are the native gate of the near-term, variational circuits studied here; fault-tolerant hardware instead builds them from a fixed discrete gate set.

Another single-qubit gate appears repeatedly. The *Hadamard gate* H creates an equal superposition from a basis state,

$$H|0\rangle = \tfrac{1}{\sqrt{2}}\big(|0\rangle + |1\rangle\big), \quad H|1\rangle = \tfrac{1}{\sqrt{2}}\big(|0\rangle - |1\rangle\big) \tag{5}$$

moving a state from one pole to the equator in the Bloch sphere. It is the standard way to prepare a qubit that responds symmetrically to further data-dependent rotations.

For n qubits the state space has 2^n basis states and is called *Hilbert* space, and a general state is a normalized complex combination of all of them. This exponential scaling underlies quantum computing's potential advantage: an n-qubit state has 2^n amplitudes that can interfere during a computation. Writing classical data into this space is costly for any encoding, since a vector of dimension d has d independent entries whose representation requires on the order of d tunable operations. One-qubit-per-entry encoding uses $n = d$ qubits, so a thousand-dimensional vector

needs a thousand qubits, beyond near-term devices. Amplitude encoding stores the vector in the 2^n amplitudes of a state and uses only $n = \lceil \log_2 d \rceil$ qubits (ten for $d = 1000$, since $2^{10} = 1024$), but preparing such a state requires, in general, a gate count of order 2^n, again linear in d; because gates are imperfect and qubits decohere over time, circuits this deep fail before completion [2]. The cost therefore scales with d under either encoding, so each input is reduced to a small feature set (Section 4.2) before encoding.

2.3.3 Entanglement and Quantum Circuits

Two-qubit gates can create *entanglement*, a quantum state of multiple qubits that cannot be written as a separate state for each qubit on its own. The standard entangling gate is the controlled-NOT (CNOT), which flips a target qubit if and only if a control qubit is in $|1\rangle$. On the ordered two-qubit basis $\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$, with the control qubit written first, it acts as

$$\mathrm{CNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \tag{6}$$

leaving the two control-$|0\rangle$ states unchanged and exchanging $|10\rangle$ and $|11\rangle$. Whether this produces entanglement depends on the input. Take the control qubit in an equal superposition and the target in its initial state $|0\rangle$; their joint state is the product

$$\tfrac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes |0\rangle = \tfrac{1}{\sqrt{2}}(|00\rangle + |10\rangle).$$

The CNOT leaves $|00\rangle$ unchanged and flips the target of $|10\rangle$, producing $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$, which no longer factorizes into independent single-qubit states. With the control in a definite basis state it leaves the state separable, and applied twice it has no effect at all, being its own inverse ($\mathrm{CNOT}^2 = I$).

A *quantum circuit* (Figure 2) is a sequence of gates applied to the generally initial state of $|0\rangle^{\otimes n}$, read left to right, and concluding with a measurement that collapses the state and returns classical bits.

2.3.4 Measurement and Expectation Values

A single circuit execution returns one outcome per measured qubit, 0 or 1, with the probabilities set by the final state. A continuous output is obtained by assigning a number to each outcome and averaging over many repetitions. The quantity that assigns these numbers is an *observable*: a matrix whose eigenvalues are the values a measurement can return. The observable used throughout is the *Pauli-Z operator*,

$$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \tag{7}$$

which satisfies $Z|0\rangle = +|0\rangle$ and $Z|1\rangle = -|1\rangle$. Its eigenvalues ± 1 are the two possible measurement values: this is the computational-basis measurement already described, with outcome 0 relabelled $+1$ and outcome 1 relabelled -1. The *expectation value* of an observable O in a state $|\psi\rangle$ is the average of its outcomes over many measurements,

$$\langle O \rangle = \langle \psi | O | \psi \rangle, \tag{8}$$

where $\langle\psi|$ is the conjugate transpose of $|\psi\rangle$, so the right-hand side is a scalar. For a single qubit $|\varphi\rangle = \alpha|0\rangle + \beta|1\rangle$, written as vector $(\alpha \quad \beta)^\top$, the bra $\langle\varphi|$ is its conjugate transpose (α^*, β^*). The operator Z flips the sign of the lower component,

$$Z|\varphi\rangle = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} \alpha \\ -\beta \end{pmatrix},$$

and the inner product with $\langle\varphi|$ gives

$$\langle Z\rangle = \langle\varphi|\, Z\, |\varphi\rangle = \begin{pmatrix} \alpha^* & \beta^* \end{pmatrix} \begin{pmatrix} \alpha \\ -\beta \end{pmatrix} = \alpha^*\alpha - \beta^*\beta = |\alpha|^2 - |\beta|^2. \tag{9}$$

Since $|\alpha|^2$ and $|\beta|^2$ are the probabilities of outcomes $+1$ and -1, this is their probability-weighted average $(+1)|\alpha|^2 + (-1)|\beta|^2$, lying in $[-1, +1]$: the difference between the two outcome probabilities.

2.4 Quantum Machine Learning

Quantum machine learning (QML) uses quantum computers for learning tasks. In the near-term setting it relies on hybrid quantum-classical algorithms: classical data is encoded into a quantum state, the state is processed by further gates, the result is measured to give a classical output, and that output runs a classical optimization loop [2]. Every such algorithm begins with an encoding, and the same encoding can be consumed in two ways. Used as a fixed *quantum kernel*, the encoded states define a similarity measure for a classical SVM, with no quantum parameters trained. Used inside a *variational quantum classifier*, a trainable block acts on the encoded state and is optimized by gradient descent. These two families, developed in turn below, share the encoding as their common substrate, which is why the encoding is the central object of study.

The translation from classical data to a quantum state, is the central bottleneck of near-term QML. A quantum computer requires the input to be expressed as a unitary applied to $|0\rangle^{\otimes n}$. Different encodings produce different states for the same input. This encoding dependence is the object of study in the experiments [27, 33].

2.4.1 Kernel Methods and Quantum Feature Maps

The first family uses the encoding as fixed kernel. Kernel methods connect the classical SVM of Section 2.1.1 to the quantum models. A *kernel function* $k(\mathbf{x}, \mathbf{u})$ measures the similarity of two points after a mapping $\varphi : \mathcal{X} \to \mathcal{H}$ into Hilbert space,

$$k(\mathbf{x}, \mathbf{u}) = \langle\varphi(\mathbf{x}), \varphi(\mathbf{u})\rangle_{\mathcal{H}}. \tag{10}$$

The map φ need not be formed explicitly: the data enter the SVM only through the kernel matrix $K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$, so any valid kernel can replace the RBF kernel of Section 2.1.1 without changing the training. This is what lets a kernel produced by a quantum circuit operate a standard classical SVM.

A *quantum feature map* takes φ to be a data-dependent quantum circuit, sending a classical vector to a state in the 2^n-dimensional Hilbert space of n qubits,

$$|\varphi(\mathbf{x})\rangle = U(\mathbf{x})\, |0\rangle^{\otimes n}, \tag{11}$$

where the encoding circuit $U(\mathbf{x})$ distinguishes the strategies studied here. Any such map defines a valid kernel through the overlap of encoded states [29], the *fidelity kernel*

$$k(\mathbf{x}, \mathbf{z}) = \left|\langle\varphi(\mathbf{x})|\varphi(\mathbf{z})\rangle\right|^2 = \left|\langle 0^{\otimes n}|\, U^\dagger(\mathbf{x})\, U(\mathbf{z})\, |0^{\otimes n}\rangle\right|^2, \tag{12}$$

estimated by running $U^\dagger(\mathbf{x})\, U(\mathbf{z})$ and recording the probability of the all-zeros outcome. This is a different function from the RBF kernel, not a circuit evaluation of Equation (2); the two share only the role of supplying K. Being a valid kernel, it passes to the classical SVM unchanged, and the convexity guarantee carries over once K is estimated accurately.

Because the kernel matrix is the entire representation of the data, its geometry decides model quality. Two properties matter and need not coincide: *expressibility* – how many dimensions of Hilbert space the feature map populates, read from the spectrum of K. And *discriminability*: whether same-class points sit closer than cross-class points, read from the *block structure* of K. Ordering the kernel's rows and columns by class label splits K into within-class diagonal blocks and cross-class off-diagonal blocks, and a discriminative kernel shows high similarity on the diagonal against low similarity off it. Thus, a feature map can be highly expressive yet poorly discriminative, and one finding of this thesis is that the two diverge. The estimators are defined in Section 4.7

2.4.2 Encoding Strategies

Five encoding strategies are studied, in order of increasing structural complexity. One of them is evaluated as a full and a shallow circuit, giving six circuits in total (Figure 2)

Angle encoding. Each feature drives one single-qubit rotation,

$$U_{\text{angle}}(\mathbf{x}) = \bigotimes_{i=1}^{n} R_y(\pi x_i), \tag{13}$$

taking each qubit from $|0\rangle$ to $\cos(\frac{\pi}{2}x_i)\,|0\rangle + \sin(\frac{\pi}{2}x_i)\,|1\rangle$, with no entanglement. The fidelity kernel therefore factorises over qubits into a product of cosines, $k(\mathbf{x}, \mathbf{u}) = \prod_{i=1}^{n} \cos^2\!\big(\frac{\pi}{2}(x_i - u_i)\big)$, the simplest, entanglement-free baseline.

Amplitude encoding. Amplitude encoding writes the feature values into the amplitudes of a state,

$$|\varphi(\mathbf{x})\rangle = \frac{1}{\|\mathbf{x}_{\text{pad}}\|} \sum_{i=0}^{2^m-1} (\mathbf{x}_{\text{pad}})_i\, |i\rangle, \tag{14}$$

where the feature vector is zero-padded to length 2^m (the smallest power of two at least d) and $|i\rangle$ is the i-th basis state of an m-qubit register. Seven features fit on $m = 3$ qubits, the most compact encoding studied. Unit-norm normalisation is imposed by quantum mechanics, it discards the magnitude of the feature vector and keeps only its direction.

Data re-uploading. Data re-uploading [26] injects the input once per layer, alternating a fixed data-loading block with a trainable block,

$$U_{\text{reupload}}(\mathbf{x}, \boldsymbol{\theta}) = \prod_{\ell=1}^{L} V_\ell(\boldsymbol{\theta}_\ell)\, S(\mathbf{x}), \tag{15}$$

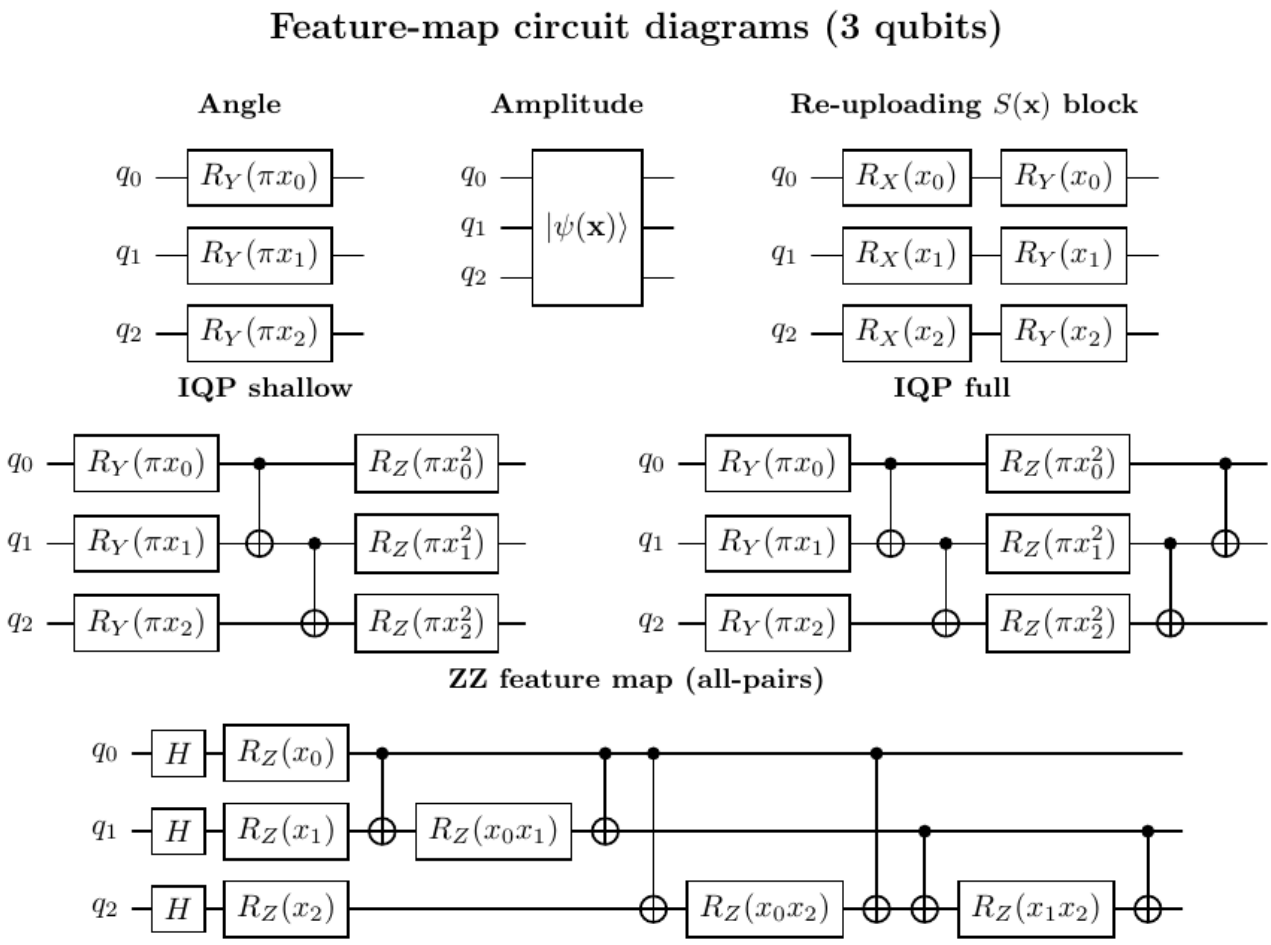

Figure 2: The six feature-map circuits, drawn on three qubits for legibility; the experiments use seven qubits, except amplitude encoding, which uses three. Angle and IQP scale features by π, and IQP additionally applies x_i^2, whereas data re-uploading and the ZZ couplings encode the raw feature values. IQP-shallow omits the reverse CNOT ladder that closes IQP-full.

where $S(\mathbf{x})$ applies $R_x(x_i)$ then $R_y(x_i)$ to each qubit and V_ℓ is the trainable block on layer ℓ. Interleaving data injection with trainable blocks is what gives the circuit its expressivity: each re-upload acts on a freshly transformed state instead of compounding into a single rotation, and in the many-layer limit a single qubit suffices for a universal classifier [26].

IQP-style encoding. IQP encoding extends angle encoding with CNOT ladders and quadratic phases,

$$U_{\mathrm{IQP}}(\mathbf{x}) = U_{\mathrm{CNOT}}^{\leftarrow}\left[\bigotimes_{i=1}^{n} R_z(\pi x_i^2)\right] U_{\mathrm{CNOT}}^{\rightarrow}\left[\bigotimes_{i=1}^{n} R_y(\pi x_i)\right], \tag{16}$$

where $U_{\mathrm{CNOT}}^{\rightarrow}$ and $U_{\mathrm{CNOT}}^{\leftarrow}$ are forward and reverse CNOT sweeps along neighbouring qubits. IQP stands for *instantaneous quantum polynomial*, a circuit class believed classically hard to sample from [13]. The *shallow* variant omits the reverse CNOT. The two define the same kernel: in the overlap circuit $U^\dagger(\mathbf{x})\,U(\mathbf{z})$ the reverse ladder meets its own inverse and cancels, since $\mathrm{CNOT}^2 = I$. The distinction survives only in the variational setting (Section 2.4.3), where no adjoint is taken.

ZZ feature map. The ZZ feature map [13] adds all-pairs interactions. A layer of Hadamards and single-qubit phases $R_z(x_i)$ is followed by a two-qubit phase on every pair (i, j) that encodes the product $x_i x_j$ as a CNOT–R_z–CNOT,

$$\prod_{(i,j)} \mathrm{CNOT}_{ij}\, R_z^{(j)}(x_i x_j)\, \mathrm{CNOT}_{ij}. \tag{17}$$

Its depth and two-qubit gate count grow quadratically with n, the highest among the candidates. The encodings differ along three axes that recur in the experiments: entanglement, circuit depth, and the ratio of features to qubits.

2.4.3 Variational Quantum Classifiers

A variational quantum classifier (VQC) combines a feature map with a trainable variational layer $V(\boldsymbol{\theta})$,

$$|\psi(\mathbf{x}, \boldsymbol{\theta})\rangle = V(\boldsymbol{\theta})\, U(\mathbf{x})\, |0\rangle^{\otimes n}. \tag{18}$$

The variational layer applies single-qubit rotations with trainable angles $\boldsymbol{\theta}$ followed by entangling CNOT gates between neighbouring qubits. The prediction is the sign of the expectation value $\langle Z_0 \rangle$ of Section 2.3.4, thresholded at zero. For data re-uploading the encoding is re-applied inside every layer (Equation 15), so there L controls both circuit depth and data-injection frequency.

VQCs and quantum kernel SVMs are two uses of the same encoding, distinguished by what is trained and what is measured. In the kernel setting no parameters are trained: the encoding alone fixes the feature-space geometry, and the model reads the probability of the all-zeros outcome of $U^\dagger(\mathbf{x})\,U(\mathbf{z})$. In the VQC setting the encoding sets a coordinate system on which V acts, the readout is instead $\langle Z_0 \rangle$, and the encoding–variational interaction governs both achievable performance and training difficulty. These two readouts, the all-zeros probability and $\langle Z_0 \rangle$, are the two ways a measurement turns a circuit into a number, and they distinguish the families throughout. Findings from one setting do not transfer automatically to the other, and the experiments report the two families separately (Figure 3).

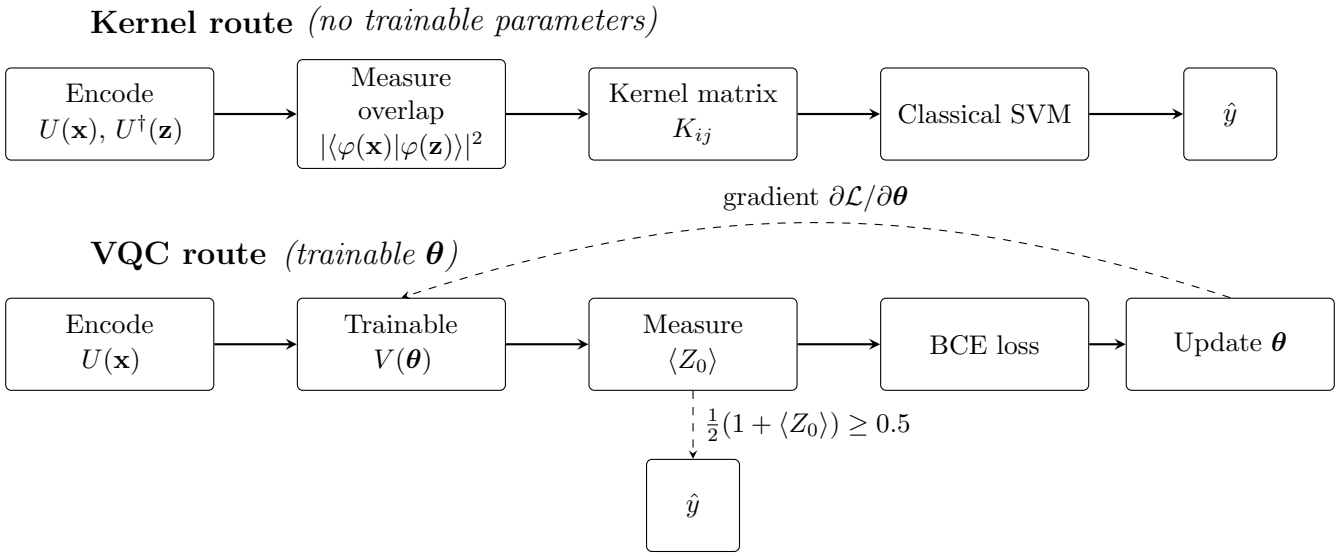

Figure 3: The two model families built on the same encoding. In the kernel route (top) no parameters are trained: the encoding fixes the kernel matrix, which a classical SVM then optimizes. In the VQC route (bottom) a trainable block $V(\boldsymbol{\theta})$ acts on the encoded state, and its parameters are updated by gradient descent on the binary cross-entropy loss. At inference the VQC output is the linear rescaling $\frac{1}{2}(1+\langle Z_0\rangle)$ thresholded at 0.5, equivalently $\langle Z_0\rangle \geq 0$.

2.4.4 Training Quantum Circuits

Training a VQC requires gradients of the loss with respect to θ. On physical hardware these cannot be obtained by classical backpropagation, which needs every intermediate value, because a circuit's intermediate states collapse when measured. The standard hardware-compatible method is the parameter-shift rule, which recovers the exact analytic gradient of a Pauli rotation from two runs of the same circuit with the parameter shifted by $\pm\pi/2$:

$$\frac{\partial\langle O\rangle}{\partial\theta_j} = \frac{1}{2}\Big[\langle O\rangle_{\theta_j+\pi/2} - \langle O\rangle_{\theta_j-\pi/2}\Big]. \tag{19}$$

The experiments here instead run on a state-vector simulator, which retains the full amplitude vector at every step, so gradients are computed by reverse-mode automatic differentiation (backpropagation) through the simulated state. This is exact and avoids the two circuit evaluations per parameter that Equation (19) requires.
The loss minimized during training is binary cross-entropy,

$$\mathcal{L}(\theta) = -\frac{1}{N}\sum_{i=1}^{N}\Big[y_i \log h_\theta(\mathbf{x}_i) + (1-y_i)\log\big(1-h_\theta(\mathbf{x}_i)\big)\Big], \tag{20}$$

where $h_\theta(\mathbf{x}_i) = \frac{1}{2}\big(1+\langle Z_0\rangle\big) \in [0,1]$ is the predicted probability of class 1, a linear rescaling of the expectation value. Thresholding this probability at 0.5 is equivalent to thresholding $\langle Z_0\rangle$ at zero, which is the decision rule used at inference. Optimization uses the Adam optimizer [18] with a fixed learning rate; hyperparameters are reported in the Methods chapter.

2.4.5 Two Failure Modes

The experiments are designed to detect two pathologies of quantum models, both arising from the exponential growth of the Hilbert space with qubit count.

Barren plateaus. For sufficiently expressive variational circuits, the variance of the loss gradient with respect to any single parameter decays exponentially in the number of qubits [24, 3, 20],

$$\mathrm{Var}\left[\frac{\partial \mathcal{L}}{\partial \theta_j}\right] \in \mathcal{O}\left(2^{-n}\right). \tag{21}$$

When a circuit can reach almost any state, the output becomes nearly independent of any individual parameter, and gradients averaged over random initializations become indistinguishable from noise. This is the quantum analogue of the vanishing-gradient problem in deep neural networks, the exponential decay scaling with qubit count rather than network depth. A barren plateau is a property of the encoding-variational pair, not of the encoding alone [9], so the experiments report the gradient variance at initialization across many random weight seeds as a direct diagnostic.

Exponential concentration in quantum kernels. An analogous problem affects kernels. Quantum kernel values can concentrate exponentially: as qubit count grows, all pairwise values converge to one constant and the kernel matrix becomes uninformative regardless of the data [31]. Concentration can be due to high encoding expressivity, global measurements, entanglement, and hardware noise. A concentrated kernel reduces the SVM to a trivial model that predicts one class for every input. Concentration is detected through the off-diagonal standard deviation of the kernel matrix: a working kernel uses the full range of similarity values, while a concentrating kernel collapses toward a single value with vanishing spread.
Barren plateaus concern the trainability of variational parameters, whereas exponential concentration concerns the informativeness of the kernel itself. The encodings are evaluated against both, in addition to classification performance.

3 Related Work

This thesis builds on three lines of empirical work: benchmarks of quantum against classical classifiers, comparative studies of data-encoding strategies, and applications of variational quantum circuits to binary classification.
Quantum kernel SVMs (QSVMs) have been benchmarked against classical machine learning on a range of tabular tasks, with the consistent finding that they approach but do not yet exceed strong classical baselines. Tijare et al. [32] benchmark QSVMs against classical ML on four bioinformatics datasets and report classical models superior throughout. Optimised QSVM is compared against SVMs with RBF, linear, polynomial, and sigmoid kernels across different feature counts [14]. The closest biomedical analogue is Hammachi et al. [12], who contrast classical and quantum SVMs for myopathy detection from electromyography features, where the same binary, single-channel, feature-engineered setting examined here. A similar QSVM pipeline with PCA preprocessing is applied to healthcare classification datasets [25]. The present work adopts the same comparative protocol on ECG-derived HRV features.

A smaller body of work targets the encoding circuit directly. Jha et al. [16] compare quantum feature maps for quantum-kernel SVMs across multiple datasets and find that no single map dominates: the best choice depends on the data. The same conclusion is reached for hybrid quantum neural networks, contrasting Z, ZZ, and Pauli feature maps [30]. De Lorenzis et al.[6] show, in the quantum extreme-learning-machine setting, that varying the encoding and the pre-encoding dimensionality reduction substantially shifts test accuracy on image data. These studies motivate the central question pursued here but stop at accuracy comparisons. The present work additionally diagnoses why performance differs through kernel-matrix structure.
On the variational side, Kakız et al.[17] perform binary classification on the Maternal Health Risk dataset using a three-qubit variational quantum circuit with amplitude embedding, reaching 92% accuracy in PennyLane. This is representative of VQC binary classification studies [1], which report one accuracy figure on one dataset under one encoding.
The benchmarking protocol of K random initialisations per fold with median ± median absolute deviation reporting across seeds follows the benchmark from [10] on quantum machine learning for time-series prediction.

4 Approach

This chapter specifies the dataset, preprocessing, encodings, model configurations, evaluation protocol, and experimental design.

4.1 Dataset and Labelling

All experiments use the Shandong Provincial Hospital (SPH) 12-lead ECG database [21], comprising 25,770 records from 24,666 patients sampled at 500 Hz over ten seconds. Only Lead II (voltage trace) is used: rhythm is carried by the timing between R peaks, which is shared by all twelve leads, and Lead II records the R peak with the tallest, clearest deflection, so a single lead suffices for reliable peak detection.
Each record receives one of two labels: normal sinus rhythm (class 0) when its only code is the normal code, and rhythm-abnormal (class 1) when it carries any code from a fixed set of eighteen rhythm-based abnormalities (the full list is given in Appendix 7). Records whose abnormalities are purely morphological are excluded, because such abnormalities do not manifest in the timing between heartbeats and therefore lie outside the information available to a rhythm-based representation. After this filtering, 13,905 records are normal and 5,787 are rhythm-abnormal.
The rhythm-abnormal class is dominated by sinus-node conditions, of which sinus bradycardia is the single largest component. This composition skew is a property of the source database, and its consequences for the interpretation of the results are discussed in Section 6.

4.2 HRV Feature Extraction

The models require a fixed-length numerical input, whereas each recording is a variable-length waveform. Because cardiac rhythm is carried by the timing between beats rather than the morphology of any single beat, the signal is summarised by heart rate variability (HRV) features computed from its RR interval series. HRV features are the standard descriptors of inter-beat timing

and capture the two properties that separate normal sinus rhythm from arrhythmia: the average rate and the regularity of successive intervals. From each ten-second Lead II recording, seven such features are extracted. The signal is first cleaned and its R peaks located with NeuroKit2 [23]. A plausibility filter retains only intervals between 0.3 and 2.0 seconds to remove detection errors. The resulting series is truncated or zero-padded to a fixed length of fifteen intervals so that every record yields a feature vector of the same dimension. Seven time-domain features summarize the series along three axes: central tendency (mean and median RR interval), overall dispersion (standard deviation and interquartile range), and short-term beat-to-beat variability (RMSSD, the root mean square of successive differences; MSD, the mean absolute successive difference; and pNN50, the proportion of successive differences exceeding 50 ms). Only time-domain features are used: a ten-second window contains roughly ten to seventeen intervals, too few to support the frequency-domain or nonlinear HRV measures that require longer recordings.

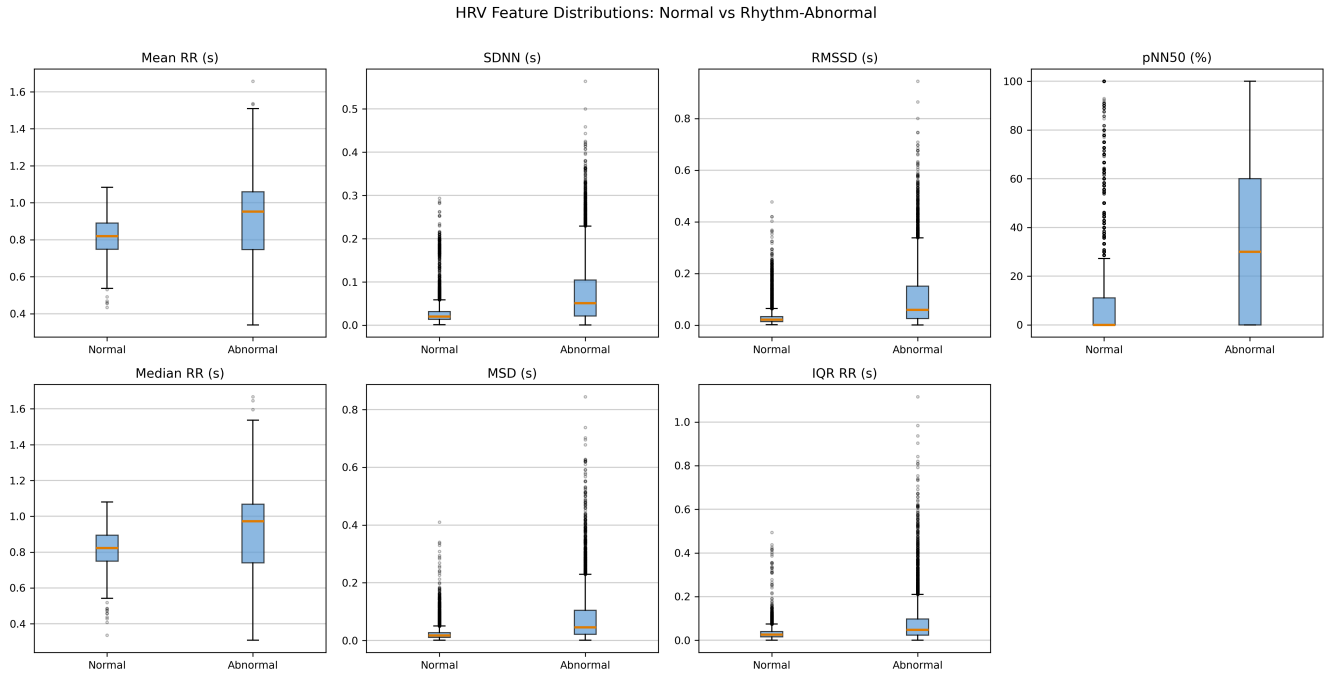

Figure 4: Class-wise distributions of the seven time-domain HRV features. The abnormal class shows a longer mean and median RR interval, consistent with the high prevalence of sinus bradycardia, and substantially wider spread on all five variability features (SDNN, RMSSD, pNN50, MSD, IQR).

4.3 Canonical Dataset and Cross-Validation

Experiments are run on a balanced subset of 400 records, 200 per class. This size is empirically sufficient, as a learning curve shows classical SVM AUC flat between 0.955 and 0.958 as the training set grows from 300 to 3,200 samples, so additional data does not move the decision boundary. Records are subsampled with a single stateful random generator, which avoids the correlated draws that a per-patient fixed seed would produce for patients holding several records.

The 400 records are partitioned into five folds by stratified k-fold at the patient level, using a single shuffle with a fixed seed. Each class is shuffled independently and divided into five equal parts, so

that every fold tests exactly 40 normal and 40 abnormal records against 160 and 160 in training. Every patient appears in exactly one test fold, and patient-level overlap between train and test was verified to be zero in all folds. The split is stored once and reused without modification by every encoding and model family, so that cross-encoding comparisons are made on an identical partition of patients.

All metrics are reported as the median ± the median absolute deviation (MAD) over the five folds. The median is preferred to the mean because one fold (split 4) is consistently harder than the others, and the MAD records that variability without letting it dominate the central estimate. The deterministic models (the quantum kernel SVM and the three classical baselines) produce one value per fold. The VQC carries a second source of variance, the random weight initialization, and is therefore trained ten times per fold from independent seeds; the median over the ten initializations gives the per-fold value, and the median ± MAD is then taken over the five folds.

4.4 Preprocessing and Normalization

The quantum models share a common preprocessing pipeline. Features are standardized to zero mean and unit variance with a scaler fitted on the training fold and applied unchanged to the test fold, then passed through a tanh squashing. The tanh step is specific to the quantum models: rotation gates are periodic with period 2π, so standardized values beyond roughly $\pm 3\sigma$ would fall back into the rotation range and be encoded as if they were small. The tanh map is monotonic and bounded, so it keeps the ordering of the features while confining every value to one rotation period. It is preferred to apply min–max scaling because the HRV features contain visible outliers (Figure 4) that would compress the bulk of the distribution into a narrow band under a min–max map. The classical baselines use the standardized features directly, without tanh.

When fewer than seven qubits are available, the pipeline is extended to standardization, then principal component analysis (PCA) fitted on the training fold alone, then tanh. PCA reduces the seven features to the number of available qubits, and is applied before tanh because it relies on the linear covariance structure that standardization preserves and the tanh map would distort.

A tanh range of $(-1, 1)$ is matched to encodings whose rotation arguments carry a π multiplier, but not to data re-uploading, whose loading rotations have no such multiplier; the consequences of this mismatch are examined in the results.

4.5 Encodings

Five encodings are studied, defined in the Background chapter: angle (Equation 13), amplitude (Equation 14), data re-uploading (Equation 15), IQP (Equation 16), and the ZZFeatureMap (Equation 17). Data re-uploading applies $R_x(x_i)$ and $R_y(x_i)$ per qubit in each layer, with no π multiplier, which is the source of the normalization mismatch noted above. Amplitude encoding ℓ_2-normalizes the seven-feature vector, zero-pads it to eight amplitudes, and loads it onto three qubits.

A sixth encoding, shallow IQP, is a variant of IQP with the reverse CNOT ladder removed:

$$U_{\text{IQP-shallow}}(\mathbf{x}) = \left[\bigotimes_{i=1}^{n} R_z(\pi x_i^2)\right] U_{\text{CNOT}}^{\rightarrow} \left[\bigotimes_{i=1}^{n} R_y(\pi x_i)\right]. \tag{22}$$

It was introduced to test whether the entanglement added by the reverse sweep helps or harms classification on the HRV features.

4.6 Models

Quantum kernel SVM. The fidelity kernel of Equation (12) is estimated by running the circuit $U^\dagger(\mathbf{x})\,U(\mathbf{z})$ on a state-vector simulator and reading the probability of the all-zeros outcome. The training Gram matrix is $N_{\text{train}} \times N_{\text{train}}$ and the test matrix $N_{\text{test}} \times N_{\text{train}}$; both are passed to a support vector classifier with a precomputed kernel and $C = 1.0$. Given a fold and an encoding the result is deterministic.
The encoding may optionally be repeated within the kernel circuit. Applying it r times before the adjoint replaces U with the effective map $V(\mathbf{x}) = U(\mathbf{x})^r$ in Equation (12), so the kernel compares the repeated states $|\varphi_r(\mathbf{x})\rangle = U(\mathbf{x})^r\,|0\rangle^{\otimes n}$.

Variational quantum classifier. The VQC alternates the fixed encoding with a trainable block over $L = 2$ layers, in the form of Equation (18). The trainable block is a Basic entangler layers circuit, one R_x rotation per qubit per layer followed by a ring of CNOT gates between neighbouring qubits; the encoding gates themselves carry no trainable parameters. At seven qubits and two layers this gives fourteen trainable parameters. The output is the expectation $\langle Z_0\rangle$, rescaled from $[-1, 1]$ to $[0, 1]$ and trained against the binary cross-entropy loss of Equation (20) with the Adam optimizer at a learning rate of 0.01 for 150 epochs, from ten random weight initializations per fold.

Classical baselines. Three baselines provide reference points (Table 1).

Table 1: Classical baseline configurations (scikit-learn). All other hyperparameters are library defaults.

Model	Configuration
RBF-SVM	$C = 1.0$, $\gamma =$ scale
Logistic regression	$C = 1.0$, max. iterations = 1000
MLP	one hidden layer of 50 units, ReLU, Adam, max. iterations = 2000

Hyperparameter provenance. The model hyperparameters above were held fixed rather than tuned. The penalty at $C = 1.0$. The layer count $L = 2$ was fixed for comparability across encodings and phases rather than chosen for performance. No hyperparameter search was carried out. This choice keeps the comparison between encodings on equal terms, but it is a limitation (Section 5.4).

4.7 Evaluation Metrics and Diagnostics

The quality of classification is measured by accuracy, the F_1 score, and the area under the ROC curve (AUC). AUC is the area under the curve of true-positive against false-positive rate over all thresholds, which equals the probability that a randomly chosen abnormal record scores higher than a randomly chosen normal one; 1 is perfect ranking, 0.5 is chance. It is the primary metric as it is threshold-free and prevalence-independent, so it is unaffected by the gap between the balanced sample and the roughly 7:3 normal-to-abnormal prevalence of the source database.
Three measures evaluate the geometry of the kernel matrix K. The *block separation* measures how much class structure the kernel encodes, as the difference between the mean kernel value over

same-class pairs and the mean over cross-class pairs, with the diagonal excluded:

$$S(K) = \frac{1}{|\mathcal{S}|} \sum_{(i,j)\in\mathcal{S}} K_{ij} \; - \; \frac{1}{|\mathcal{C}|} \sum_{(i,j)\in\mathcal{C}} K_{ij}, \tag{23}$$

where $\mathcal{S}$ and $\mathcal{C}$ are the sets of same-class and cross-class off-diagonal pairs. The *off-diagonal standard deviation* is the standard deviation of the off-diagonal entries of K; a small value signals the exponential concentration described in Section 2.4.5. The *spectral rank* counts how many independent directions the kernel actually uses. The eigenvalues λ_i of K give the share of variance along each direction; normalized to sum to one, $p_i = \lambda_i / \sum_j \lambda_j$, the rank is the number of largest directions needed to reach 95 % of the total,

$$r_{0.95}(K) = \min\Big\{ k : \textstyle\sum_{i=1}^{k} p_i \geq 0.95 \Big\}. \tag{24}$$

A low rank means the kernel packs the data into few directions, a high rank that it spreads them across many. This 95 %-variance rank is reported throughout, and differs numerically from the entropy-based effective rank of Roy and Vetterli [28].

Trainability is measured by the gradient variance at initialization. For each trainable parameter θ_j, the variance of $\partial\mathcal{L}/\partial\theta_j$ is estimated over 50 random weight initializations at epoch 0 on one representative fold, and the reported value is the mean over parameters. The mean is used because one parameter per layer block, the rotation feeding the measured qubit, has an identically zero gradient by construction. A single-parameter estimate would therefore read an imitative near-zero value.

4.8 Experimental Design

The experiments address the central research question of how classical-to-quantum encoding strategies influence both the ability of a model to represent complex decision boundaries and the trainability of its circuit parameters under realistic resource constraints. This question is broken into four sub-questions: how the encodings shape the feature-space geometry and the resulting decision boundaries; whether some encodings converge faster or more stably during VQC optimization; how the encodings interact with limited circuit depth and small qubit counts; and how the quantum models compare with classical baselines on the same task.

The work is organized into four phases, each mapped to a sub-question (Table 2). The comparison with the classical baselines runs through all four phases.

Table 2: Numbered experiment steps, the model family used, and the sub-question each principally addresses.

Phase	Model family	Sub-question
0	Classical baselines	Comparison reference (Q4)
1	Quantum kernel SVM	Feature-space geometry (Q1)
2	Variational quantum classifier	Trainability and convergence (Q2)
3	Both, under resource limits	Depth and qubit-count constraints (Q3)

4.9 Implementation

Quantum circuits were implemented in PennyLane and executed on its state-vector simulator, which returns exact expectation values and kernel probabilities without shot noise, so the results isolate the effect of the encodings from sampling error. The classical baselines, the support vector classifier, the PCA reduction, and all metrics use scikit-learn, and R-peak detection uses NeuroKit2. Experiments were run on the ALICE compute cluster.

5 Results

All metrics are median ± MAD over the five canonical folds; for VQC the per-fold value is itself the median over ten weight initializations, so the reported spread is split-to-split variation with initialization variance absorbed.

5.1 Classical Baselines

Table 3 records the three classical baselines, used as fixed reference points throughout. Logistic regression trails the other two: the HRV classes are not linearly separable, and RBF-SVM's gain over it confirms that separation requires nonlinear combinations of the seven features. Splits 2 and 4 are consistently weak across all three models, a property of their patient mix that motivates reporting the median rather than the mean.

Table 3: Classical baseline performance (median ± MAD, 5 splits, 320 train / 80 test per fold).

Model	Accuracy	F1	AUC
RBF-SVM	0.9125 ± 0.0250	0.9114 ± 0.0284	0.9356 ± 0.0112
MLP	0.9000 ± 0.0125	0.8974 ± 0.0033	0.9413 ± 0.0069
Logistic Reg.	0.7500 ± 0.0250	0.7297 ± 0.0131	0.7812 ± 0.0400

5.2 Quantum Kernel SVM

5.2.1 Encoding Comparison

Figure 5 plots AUC against two-qubit gate count at a single encoding repetition. Re-upload achieves the highest AUC using only single-qubit rotations, at zero entangling cost; ZZ matches it within MAD but pays 42 two-qubit gates. IQP, Angle, and Amplitude trail. Performance is therefore not achieved with entangling gates, as the cheapest encoding leads.

Figures 6 and 7 examine the kernel geometry underlying this ranking. Three distinct failures surface. For Amplitude, same-class and cross-class distributions nearly coincide (Fig. 6.2): the ℓ_2-normalization that produces a valid quantum state discards absolute feature magnitudes, so two patients whose feature vectors point the same way are indistinguishable regardless of scale, eliminating the primary discriminator of rhythm abnormality. IQP's values instead spike near zero for both pair types (Fig. 6.3), the signature of exponential concentration. Angle, despite a high

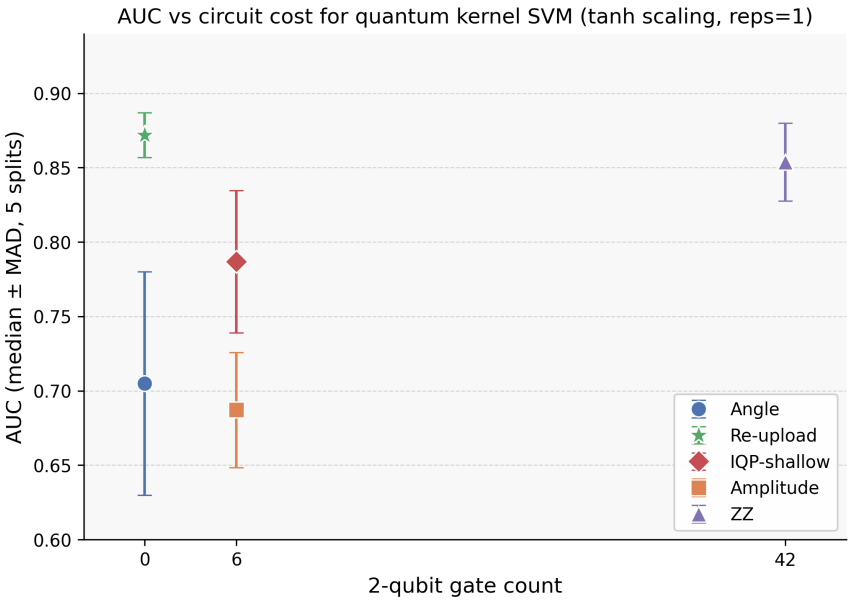

Figure 5: AUC against two-qubit gate count for the five quantum kernel SVM encodings (median ± MAD, K=5 splits, reps = 1, tanh scaling). Horizontal position represents hardware cost; vertical position encodes classification performance. Error bars are MAD across splits.

spectral rank, shows nearly identical same- and cross-class distributions (Fig. 6.1). Expressibility and discriminability are therefore not equivalent.

Re-upload encodes each feature with two rotations about different axes, $R_X(x_i)$ then $R_Y(x_i)$: their product is a single-qubit rotation whose axis and angle vary with x_i, unlike angle encoding's fixed y-axis rotation. This richer per-feature map yields the largest class-driven separation. ZZ's $R_Z(x_i x_j)$ cross terms encode the pairwise HRV correlations relevant to rhythm classification, reaching an intermediate rank that avoids both the amplitude information loss and the IQP concentration.

5.2.2 Expressibility versus Discriminability

Figure 8 shows that spectral rank does not predict AUC: IQP occupies the most dimensions of Hilbert space yet underperforms ZZ, and Angle's rank exceeds both yet its AUC is second lowest. The eigenspectrum measures how many dimensions the encoding uses; block separation measures whether those dimensions align with the decision boundary. For IQP and Angle the two diverge: more dimensions, worse alignment.

The mechanism is again exponential concentration. As spectral rank grows the kernel approaches a scaled identity, its off-diagonals tending to a common constant, so the SVM sees a near-uninformative Gram matrix. ZZ escapes this because its cross-feature terms $R_Z(x_i x_j)$ are structured couplings, not generic high-dimensional projections.

5.2.3 IQP and IQP-Shallow are Analytically Identical as Kernels

IQP and IQP-shallow produce numerically identical kernel matrices across all five splits, confirming empirically the cancellation derived in Sections 2.4.2 and 4.5. IQP-shallow is also a correct kernel

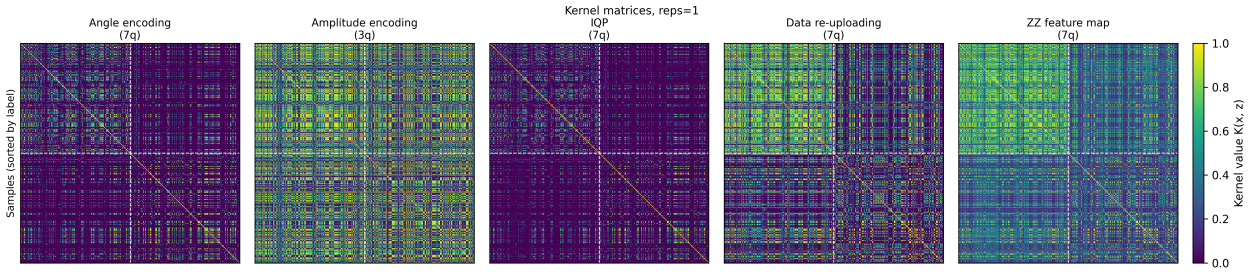

Figure 6: Kernel matrices for all five encodings (split 3, samples sorted by label; dashed white lines mark the class boundary). ZZ and Re-upload show visible block structure; IQP concentrates near zero; Amplitude spreads values broadly but without class-driven architecture.

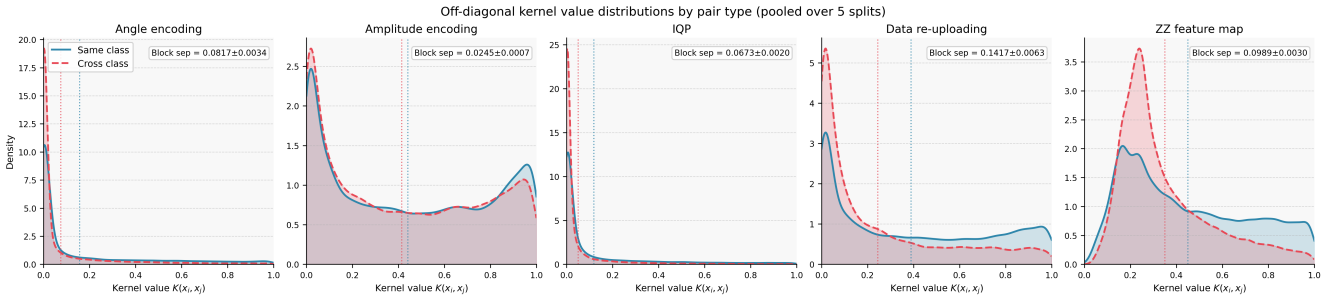

Figure 7: Off-diagonal kernel value distributions by pair type, pooled across all five splits. Dotted vertical lines mark the within-class (blue) and cross-class (red) means; block separation is their difference (Equation 23).

implementation; the full IQP circuit computes the same function at double the entangling cost.

5.2.4 Effect of Repeated Encoding Layers

Table 4: Kernel SVM AUC under repeated encoding layers (median ± MAD, K=5 splits).

Encoding	Reps	AUC	Block sep.	Eff. rank
Angle	1	0.7050 ± 0.0750	0.0817 ± 0.0034	74
Angle	2	0.6288 ± 0.0094	0.0323 ± 0.0008	125
Angle	3	0.6438 ± 0.0106	0.0110 ± 0.0003	152
ZZ	1	0.8538 ± 0.0262	0.0989 ± 0.0030	30
ZZ	2	0.8581 ± 0.0331	0.0772 ± 0.0032	43
ZZ	3	0.8362 ± 0.0325	0.0741 ± 0.0029	85

Table 4 reports repetition layers for Angle and ZZ. For Angle, repetition is just a rescaling, $R_Y(x_i\pi)\,R_Y(x_i\pi) = R_Y(2x_i\pi)$, introducing no new cross-feature structure; spectral rank nonetheless climbs as redundant dimensions accumulate while block separation collapses toward a constant matrix that carries no class information. For ZZ, a second repetition buys AUC within the

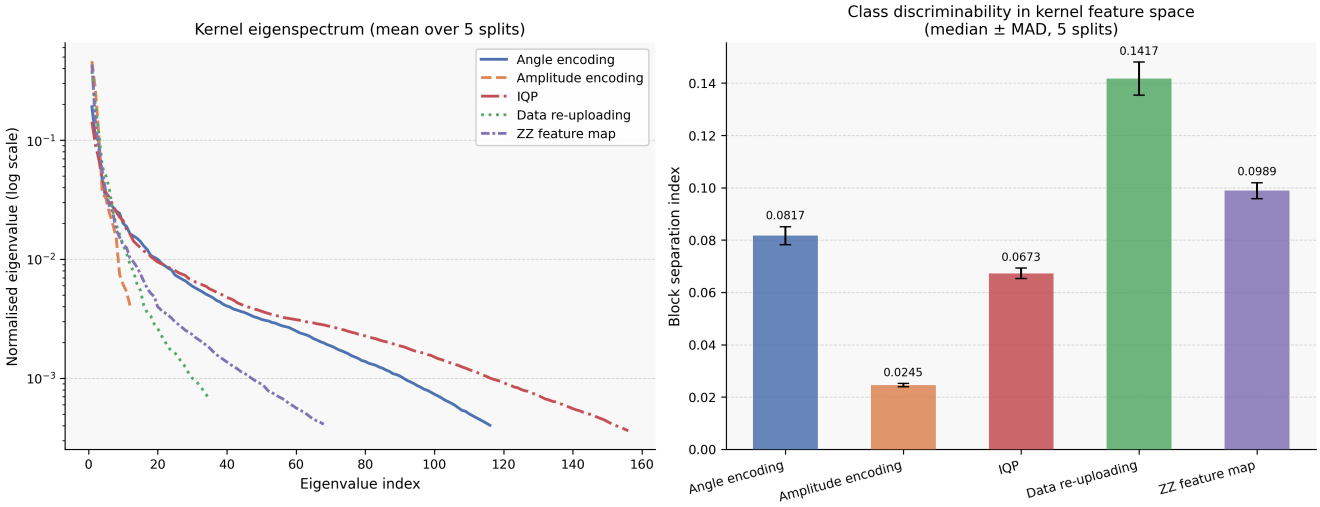

Figure 8: Left: normalised kernel eigenvalue decay curves (mean over 5 splits, log scale). Right: block separation index per encoding (median ± MAD, K=5 splits). IQP's slow decay to index ∼155 corresponds to spectral rank 104; ZZ truncates near index 80 (rank 30). Angle's rank of 74 exceeds ZZ's yet produces lower block separation, illustrating that high expressibility does not guarantee class discriminability.

MADs at double the gate count, and a third sends AUC back down as the rank climbs into the IQP concentration regime. Therefore, repeating a fixed encoding without interleaved trainable parameters adds cost.

5.2.5 Normalisation Sensitivity

Replacing tanh with MinMaxScaler $\to [-\pi, \pi]$ for Re-upload collapses AUC from 0.87 to 0.63 while block separation marginally rises. The HRV features are heavy-tailed. Rotation gates barely turn for most patients while the few outliers dominate the kernel geometry, producing artificial between-class separation that does not generalize: block separation rises while AUC falls. Tanh avoids this by saturating toward ± 1 regardless of outlier magnitude, preserving the ordering of the bulk while keeping rotations in a meaningful range. Optimal normalization therefore tends to be encoding-dependent.

5.3 Variational Quantum Classifiers

Phase 1 treated each encoding's feature space as a fixed kernel. Phase 2 adds a trainable variational block to each encoding and asks whether gradient-based optimization improves on that fixed kernel, and whether the encoding governs how well the optimization succeeds. The variational setting exposes the second failure mode of Section 2.4.5, barren plateaus, which kernel methods cannot exhibit.

All six encodings are evaluated here, including both IQP variants: the kernel collapsed them onto one function, but the variational setting takes no adjoint and separates them. The training configuration and the median-over-initializations protocol are as specified in Section 4.7.

5.3.1 Performance Comparison

Table 5: Phase 2 VQC results for all six encodings on the canonical balanced dataset ($N = 400$, 200 per class). Accuracy, F_1, and AUC are median ± MAD over five stratified cross-validation splits; each split value is the median over ten random weight seeds. Gradient variance is the mean over all trainable parameters of the per-parameter gradient variance, estimated on split 0 over fifty random initialisations. All encodings use $L = 2$ `BasicEntanglerLayers` variational layers, 150 training epochs, and the Adam optimiser. Encodings are ordered by AUC (descending).

Encoding	Accuracy	F_1	AUC	Grad. variance
Re-upload	0.6938 ± 0.0125	0.5831 ± 0.0485	0.8753 ± 0.0128	4.61×10^{-4}
ZZFeatureMap	0.7000 ± 0.0125	0.6175 ± 0.0398	0.8313 ± 0.0147	6.64×10^{-4}
Angle	0.6500 ± 0.0062	0.6024 ± 0.0264	0.7594 ± 0.0106	2.65×10^{-4}
Amplitude	0.7000 ± 0.0250	0.6988 ± 0.0146	0.7594 ± 0.0438	4.09×10^{-3}
IQP	0.6938 ± 0.0250	0.6626 ± 0.0205	0.7588 ± 0.0275	2.18×10^{-4}
IQP-shallow	0.6687 ± 0.0312	0.6637 ± 0.0410	0.7412 ± 0.0316	1.84×10^{-4}

Figure 9 shows two clusters: Re-upload and ZZFeatureMap lead, the other four sit lower. Within the lower cluster the metrics diverge: Amplitude reaches the joint-highest accuracy yet an AUC no better than Angle's, whose accuracy is markedly lower. The two leaders show the inverse pattern: a competitive AUC with a depressed F_1, Re-upload posting the lowest F_1 of any encoding. They rank the classes well but place the threshold poorly for the abnormal class, a calibration effect examined in Section 5.3.4.

5.3.2 The Kernel-to-VQC Gap

Figure 10 shows the gap Δ = VQC AUC − kernel AUC is non-monotone in kernel quality. Re-upload sits on the diagonal (Δ within the MAD of both): the variational block adds nothing beyond its fixed feature space. Angle and Amplitude gain substantially, the block compensating for their weak Phase 1 geometries with a nonlinear mapping. ZZFeatureMap and both IQP variants lose AUC: strong kernel geometry does not always survive gradient-based optimization under a finite epoch budget.

The data re-uploading is the best strategy here: re-injecting the data each layer sustains gradient flow, whereas ZZ's cross-term-rich landscape does not survive optimization intact. The four weaker encodings are close to their Phase 1 order, so kernel geometry predicts VQC performance in this case.

5.3.3 Gradient Variance and Barren Plateau Risk

Barren plateau theory predicts gradient variance decaying exponentially with qubit count and depth (Section 2.4.5). Variance is reported as the mean over all trainable parameters to avoid a structural artefact: one rotation per layer block, the qubit-0 parameter feeding the Z_0 measurement, has identically zero gradient at initialization, so a single-parameter read returns falsy values of order 10^{-35}.

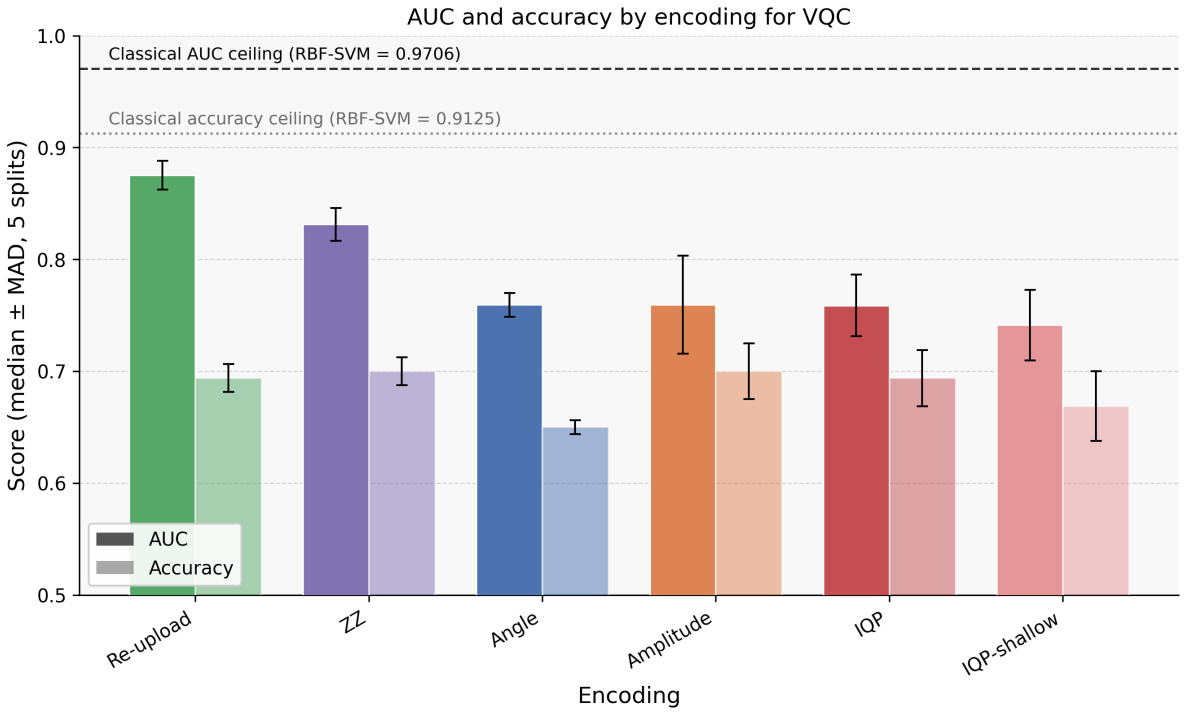

Figure 9: AUC and accuracy for the six VQC encodings (median ± MAD, K=5 stratified splits; per-split value is the median over ten random weight initialisations; L=2 `BasicEntanglerLayers`, 150 epochs, Adam). Error bars are MAD across splits.

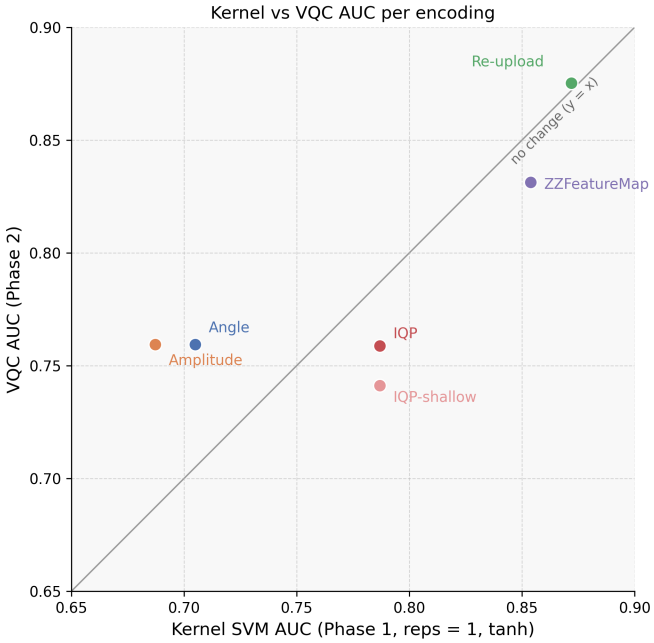

Figure 10: Kernel-to-VQC transition. Horizontal axis: kernel AUC (Phase 1, reps = 1, tanh). Vertical axis: VQC AUC (L=2, 150 epochs, Adam). Points above the dashed zero-gap diagonal gain AUC; those below lose.

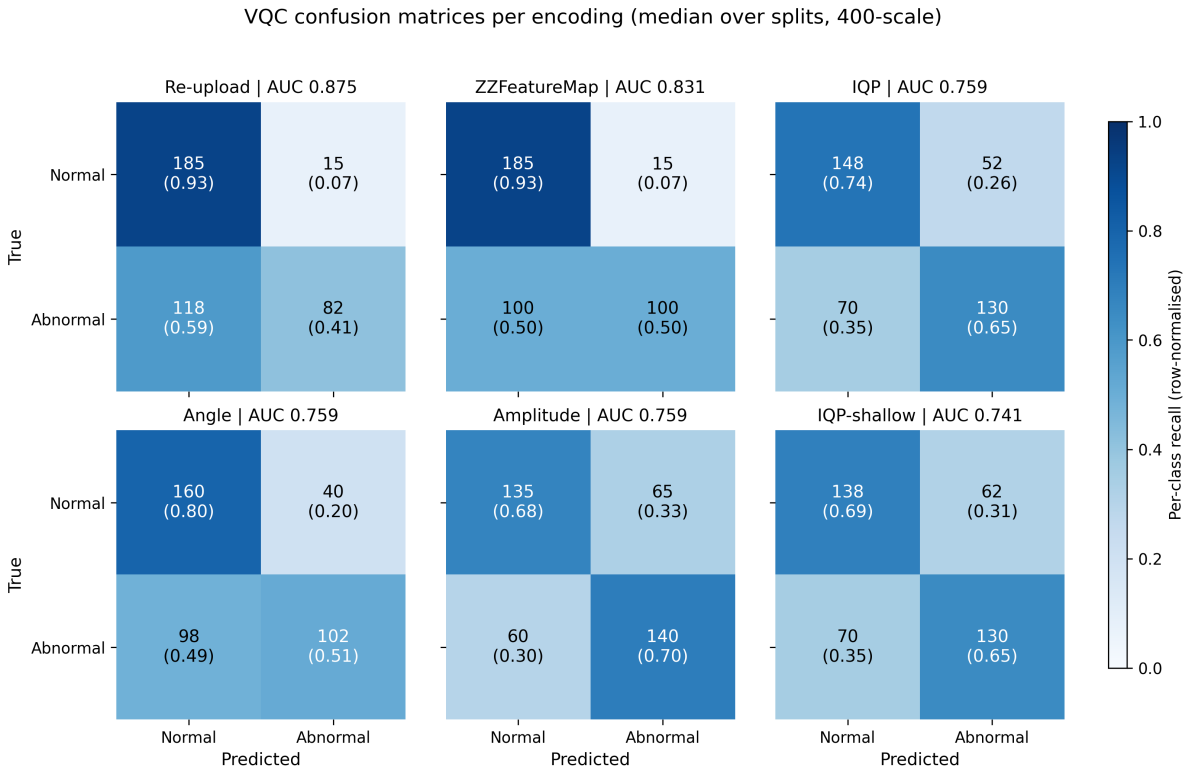

Figure 11: Confusion matrices panel per encoding.

Across the five seven-qubit encodings the variance stays within one order of magnitude (Table 5). Every encoding's above-random AUC confirms training proceeds, with no $L = 2$ seven-qubit circuit reaching the concentration regime [3].

Amplitude stands an order of magnitude above this band, a consequence of circuit size. At three qubits its variational block carries far fewer parameters over a small Hilbert space. Its Phase 2 limitation is therefore expressibility, as a three-qubit block spans only a restricted family of decision boundaries.

5.3.4 Class-Wise Errors and Threshold Calibration

Figure 11 aggregates each encoding's confusion matrix over the five test folds, so every sample is classified once. The matrices explain the AUC–F_1 divergence of Section 5.3.1 (a model can rank cases well yet place its 0.5 boundary poorly). The two highest-AUC encodings are the worst calibrated. Re-upload classifies almost all normal cases correctly but recovers fewer than half the abnormal cases, and ZZFeatureMap shows the same bias more mildly: the variational block separates the classes, but the fixed threshold sits too high for the abnormal class and suppresses its recall, hence F_1. The lower-AUC encodings spread errors evenly, so their F_1 tracks accuracy. The gap is thus an artefact of threshold placement, not ranking quality; a validation-fold threshold calibration would recover F_1 for the high-AUC encodings without affecting AUC.

5.4 Resource Constraints

Phases 1 and 2 fixed circuit width and depth. Phase 3 varies them, asking how circuit depth, qubit count, and embedding structure interact with classification performance. Three sweeps are reported, each motivated by a specific finding from the earlier phases.

5.4.1 Re-upload Depth Sweep

Re-uploading gave the strongest variational result in Phase 2, motivating a closer look at the layer count L that governs it. For this encoding L sets two quantities at once: circuit depth and the number of data re-injections, since each variational layer is preceded by a fresh encoding block. The sweep over $L \in \{1, 2, 3\}$ tests whether more injections compound the Phase 2 advantage or flatten the loss landscape, and whether $L = 2$ was optimal.

Table 6 reports the sweep. AUC and accuracy both rise monotonically with depth, so $L = 2$ was not optimal, as the third re-injection improves the classifier further. The fold-to-fold spread also contracts with depth, so deeper re-upload circuits are both more accurate and more consistent.

The gradient variance (Table 6) is flat across the three depths, so no barren plateau develops as layers are added.[1] Even the shallowest circuit performs strongly, consistent with the Phase 2 finding that re-upload's kernel and VQC AUCs are nearly equal: the encoding carries most of the discriminative signal on its own, and depth adds a modest gain.

The re-upload encoding adds no two-qubit gates at any depth, injecting data through single-qubit rotations only. The only entangling gates are the trainable block's CNOT ring, shared by every encoding. Re-upload therefore carries the lowest two-qubit-gate budget of any encoding studied, never paying the encoding-level cost that ZZ ($n(n-1)/2$ couplings) and IQP (CNOT ladders) add on top of that.

Table 6: Re-upload VQC depth sweep ($n = 7$ qubits, $L \in \{1, 2, 3\}$). Accuracy, F_1, and AUC are median ± MAD over five stratified splits; each split value is the median over ten random weight initializations. Gradient variance is the mean over all trainable parameters, estimated on split 0 over fifty random initializations. The two-qubit-gate column counts the encoding's contribution, which is zero at every depth; the trainable ring adds a further n CNOTs per layer, as for all encodings.

L	Accuracy	F_1	AUC	Grad. variance	Enc. 2-qubit gates
1	0.6313 ± 0.0188	0.5093 ± 0.0351	0.8663 ± 0.0231	4.06×10^{-4}	0
2	0.6938 ± 0.0125	0.5831 ± 0.0485	0.8753 ± 0.0128	4.61×10^{-4}	0
3	0.7438 ± 0.0188	0.6850 ± 0.0285	0.8841 ± 0.0066	2.85×10^{-4}	0

5.4.2 Qubit-Count Sweep

The Phase 1 ranking was measured at seven qubits, leaving open whether it reflects the encodings themselves or the particular width. IQP and ZZFeatureMap are therefore compared across $n \in \{3, 5, 7\}$ qubits in both settings, chosen for their contrasting structure: the ZZ map's cross terms grow as $n(n-1)/2$, so narrowing the circuit removes cross terms, whereas the IQP ladder

[1] Mean over all trainable parameters, as in Section 5.3.3; the per-parameter caveat given there applies.

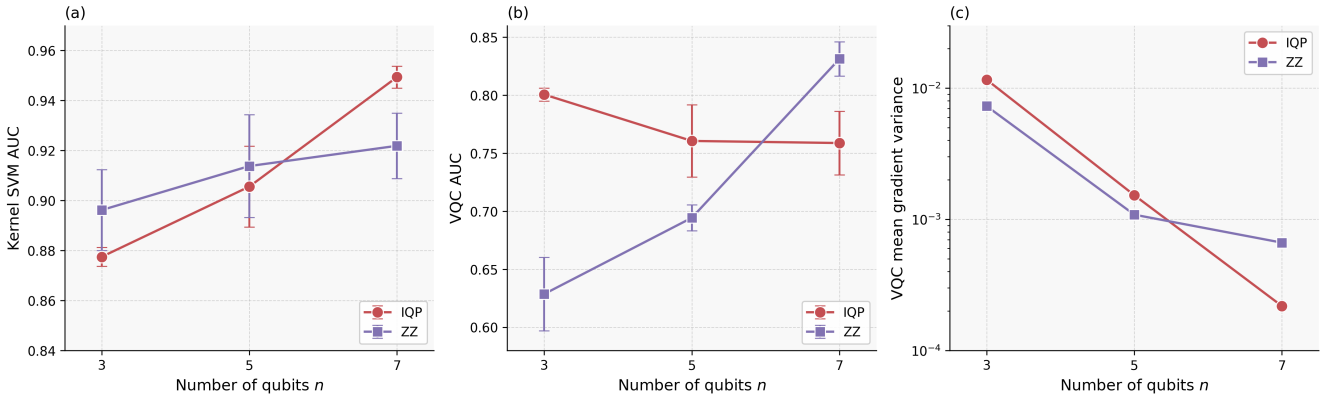

Figure 12: Qubit-count sweep for IQP and ZZ ($n \in \{3, 5, 7\}$, $L = 2$; the three- and five-qubit settings reduce the seven HRV features to n principal components by PCA fitted per training fold). (a) Kernel SVM AUC rises with qubit count for both encodings. (b) VQC AUC moves in opposite directions: it falls with n for IQP, against the trend of its own kernel, but rises with n for ZZ, so the two curves cross at seven qubits. (c) VQC mean gradient variance, taken over all trainable parameters and plotted on a logarithmic axis, decays by roughly an order of magnitude per two qubits for both encodings, the characteristic scaling of a barren plateau. Error bars in (a) and (b) show the median absolute deviation over the five stratified folds; the gradient variance in (c) is a single estimate on split 0 over fifty random initializations. The seven-qubit kernel values use an independent fold assignment and are not compared against the Phase 1 table (see text).

may concentrate differently. The three- and five-qubit settings reduce the seven HRV features to n principal components by PCA fitted per training fold (no test-fold leakage).

The seven-qubit kernels here use an independent fold assignment, so their AUCs differ from the Phase 1 table; only the within-sweep trend across three, five, and seven qubits is interpreted, not the absolute values against Phase 1.

Kernel SVM AUC rises monotonically with qubit count for both encodings (Fig. 12a): more qubits supply more kernel structure, through wider entanglement for IQP and more cross terms for ZZ, and the kernel benefits either way.

Variational trainability follows the opposite trend, the central result of this sweep. The VQC mean gradient variance (Fig. 12c, log axis) falls by roughly an order of magnitude per two qubits for both encodings, the characteristic scaling of a barren plateau. The contrast with the depth sweep is direct: the variance stays flat as depth increases but falls as width increases.

The performance consequence is encoding-dependent (Fig. 12b). The IQP VQC moves opposite to its own kernel, its AUC peaking at three qubits and falling as qubits are added: with fewer parameters and higher gradient variance, the narrow circuit optimizes more readily despite a weaker kernel. The ZZ VQC instead climbs with qubit count, in step with its kernel: the cross terms $R_Z(x_i x_j)$ encode clinically relevant HRV correlations, so added couplings improve the decision boundary faster than the weakening gradients degrade it. Whether more qubits help a variational classifier therefore depends on whether the binding constraint is kernel richness or trainability.

5.4.3 Amplitude at Seven Qubits

Phase 2 left open whether the amplitude kernel improves with more qubits. At seven qubits the feature vector occupies qubits 0–2 (padded to eight amplitudes) with qubits 3–6 left in $|0\rangle$. The inactive qubits contribute $|\langle 0|0\rangle|^4 = 1$ to every inner product, so the seven-qubit kernel is analytically identical to the three-qubit one.

Measurements confirm this. Block separation is numerically identical at three and seven qubits (Table 7), and the slight AUC rise is likely a floating-point artefact of the larger Hilbert space. The off-diagonal spread far exceeds that of other encodings at the same width, reflecting that amplitude encoding maps all samples into a compact region of the unit sphere.

Taken with the Phase 2 amplitude VQC result, this isolates the binding constraint. The kernel measures only the angle between normalized states, so it cannot recover the magnitude information that normalization discards, and adding qubits does not help; the variational block partially recovers that structure, which is why its VQC exceeds its kernel. Amplitude encoding's weakness is therefore the information loss in ℓ_2-normalization, not a shortage of circuit capacity.

Table 7: Amplitude encoding kernel SVM at three qubits (Phase 1) and seven qubits (Phase 3). Values are median ± MAD over the five stratified splits.

Metric	3 qubits	7 qubits
AUC	0.6872 ± 0.0387	0.6938 ± 0.0100
Accuracy	0.6500 ± 0.0250	0.6250 ± 0.0500
Block separation	0.0245 ± 0.0007	0.0245 ± 0.0007

6 Conclusions and Further Research

This thesis investigated how classical-to-quantum encoding strategies influence binary classification performance, feature-space geometry, and trainability under realistic resource constraints. The four experimental phases answer the question consistently: the encoding is the dominant design choice, it acts differently on the two model families, and the strategy that is best in one setting is not guaranteed to be best in the other. Data re-uploading is the most consistent encoding across both families, reaching the highest AUC in each while adding no entangling gates of its own and holding its gradient variance stable as circuit depth grows. The ZZ feature map is its closest competitor under the kernel, but it surrenders that standing once a trainable block is placed on top of it, which already indicates that strong kernel geometry does not transfer automatically to a trainable circuit.

Three findings sharpen this picture. First, expressibility and discriminability are distinct properties of a kernel and need not coincide. The encoding that occupies the most dimensions of Hilbert space is not the one that separates the classes best, because high spectral rank coincides with the onset of exponential concentration, in which the off-diagonal kernel values collapse toward a common constant and the kernel matrix carries little class information. Second, the gap between the kernel and the variational use of the same encoding is non-monotone in kernel quality. Encodings with weak fixed geometry gain the most from a trainable block, which supplies the nonlinear mapping their kernel could not, whereas an encoding with strong kernel geometry can lose AUC under gradient-based optimization within a finite epoch budget. Third, circuit width and circuit depth

pull trainability in opposite directions. Adding qubits enriches the kernel but drives the gradient variance down at the rate characteristic of a barren plateau, while adding depth to the re-uploading circuit leaves the gradient variance flat. Whether widening a variational classifier helps therefore depends on which constraint binds, kernel richness or trainability.

The practical implication for near-term quantum machine learning is that encoding selection must weigh trainability, resource scaling, and the target model family alongside raw classification performance. Data re-uploading is the encoding that satisfies these constraints jointly: it carries no encoding-level entangling gates, sustains gradient flow as depth increases by re-injecting the data at every layer, and remains competitive in both the kernel and the variational setting. These properties make it the most promising candidate of those studied for resource-constrained quantum hardware.

Several caveats bound these conclusions. The rhythm-abnormal class is dominated by sinus-node conditions, with sinus bradycardia the single largest component, so the classifiers are rewarded in part for detecting a shift in heart rate rather than the full range of rhythm pathology. Thus the ranking of encodings may differ on a more uniform mixture of abnormalities. The model hyperparameters, including the penalty term, the variational layer count, and the optimizer settings, were held fixed for comparability rather than tuned, so the reported performance is a floor rather than the best each encoding could reach. All circuits were executed on a state-vector simulator without shot noise or hardware noise, which isolates the effect of the encodings but leaves open how far the observed concentration and barren-plateau behaviour would be compounded on physical devices. Finally, the comparison rests on a balanced sample of moderate size and on seven time-domain HRV features drawn from ten-second recordings, a representation that excludes the frequency-domain and nonlinear measures available from longer signals.

These limitations mark the natural directions for further research. The most immediate is layer-wise heterogeneous re-uploading, in which each layer encodes a different feature subset or uses different rotation axes, diversifying the Fourier frequency spectrum the circuit can represent and testing whether the depth advantage observed here compounds further. A second direction is threshold calibration on a validation fold, which the confusion-matrix analysis indicates would recover the suppressed recall, and the corresponding F_1, of the highest-AUC encodings without affecting their ranking. Beyond the noiseless simulator, the encodings should be re-evaluated under realistic shot budgets and hardware noise, the setting in which the analysis could be extended to larger and more diverse datasets and to a broader hyperparameter search. Together these steps would test how far the encoding-dependent behaviour established here persists once the artificial conditions of the present study are relaxed.

References

[1] Ernesto Campos, Aly Nasrallah, and Jacob Biamonte. Abrupt transitions in variational quantum circuit training. *Physical Review A*, 103(3):032607, 2021.

[2] M. Cerezo, Andrew Arrasmith, Ryan Babbush, Simon C. Benjamin, Suguru Endo, Keisuke Fujii, Jarrod R. McClean, Kosuke Mitarai, Xiao Yuan, Lukasz Cincio, and Patrick J. Coles. Variational quantum algorithms. *Nature Reviews Physics*, 3(9):625–644, 2021.

[3] M. Cerezo, Akira Sone, Tyler Volkoff, Lukasz Cincio, and Patrick J. Coles. Cost function dependent barren plateaus in shallow parametrized quantum circuits. *Nature Communications*, 12(1):1791, 2021.

[4] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, 1995.

[5] Nello Cristianini and John Shawe-Taylor. *Kernel-Induced Feature Spaces*, page 26–51. Cambridge University Press, 2000.

[6] A. De Lorenzis, M. P. Casado, M. P. Estarellas, N. Lo Gullo, T. Lux, F. Plastina, A. Riera, and J. Settino. Harnessing quantum extreme learning machines for image classification. *Physical Review Applied*, 23(4):044024, 2025.

[7] Ronald de Wolf. Quantum computing: Lecture notes, 2023.

[8] Raghavendra M. Devadas and Sowmya T. Quantum machine learning: A comprehensive review of integrating AI with quantum computing for computational advancements. *MethodsX*, 14:103318, 2025.

[9] Yuxuan Du, Tao Huang, Shan You, Min-Hsiu Hsieh, and Dacheng Tao. Quantum circuit architecture search for variational quantum algorithms. *npj Quantum Information*, 8(1):62, 2022.

[10] Tobias Fellner. Quantum machine learning for time series prediction. Master's thesis, University of Stuttgart, 2024.

[11] Anton Frisk Kockum. *Quantum optics with artificial atoms*. PhD thesis, 12 2014.

[12] Radhouane Hammachi, Noureddine Messaoudi, Samia Belkacem, Edoardo Pasetto, and Amer Delilbasic. Classical and quantum SVM for electromyography-based myopathy detection: A comparative exploration. *Polish Journal of Medical Physics and Engineering*, 31(2):118–130, 2025.

[13] Vojtěch Havlíček, Antonio D. Córcoles, Kristan Temme, Aram W. Harrow, Abhinav Kandala, Jerry M. Chow, and Jay M. Gambetta. Supervised learning with quantum-enhanced feature spaces. *Nature*, 567(7747):209–212, 2019.

[14] Matheus Cammarosano Hidalgo. Comparative analysis of a quantum SVM with an optimized kernel versus classical SVMs. *IEEE Access*, 13:82378–82387, 2025.

[15] Kurt Hornik. Approximation capabilities of multilayer feedforward networks. 4(2):251–257.

[16] Ravi Kumar Jha, Nikola Kasabov, Saugat Bhattacharyya, Damien Coyle, and Girijesh Prasad. Comparative performance analysis of quantum feature maps for quantum kernel-based machine learning. *Scientific Reports*, 16(1):8142, 2026.

[17] Muhammet Talha Kakı z, Erkan Güler, Tuğrul Çavdar, and Burcu Şanal. Binary classification with variational quantum circuit. In *2023 Innovations in Intelligent Systems and Applications Conference (ASYU)*, pages 1–7. IEEE, 2023.

[18] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations (ICLR)*, 2015.

[19] Pradeep Lamichhane and Danda B. Rawat. Quantum machine learning: Recent advances, challenges, and perspectives. *IEEE Access*, 13:94057–94105, 2025.

[20] Martín Larocca, Supanut Thanasilp, Samson Wang, Kunal Sharma, Jacob Biamonte, Patrick J. Coles, Lukasz Cincio, Jarrod R. McClean, Zoë Holmes, and M. Cerezo. Barren plateaus in variational quantum computing. *Nature Reviews Physics*, 7(4):174–189, 2025.

[21] Hui Liu, Dan Chen, Da Chen, Xiyu Zhang, Huijie Li, Lipan Bian, Minglei Shu, and Yinglong Wang. A large-scale multi-label 12-lead electrocardiogram database with standardized diagnostic statements. *Scientific Data*, 9(1):272, 2022.

[22] Yunchao Liu, Srinivasan Arunachalam, and Kristan Temme. A rigorous and robust quantum speed-up in supervised machine learning. *Nature Physics*, 17(9):1013–1017, 2021.

[23] Dominique Makowski, Tam Pham, Zen J. Lau, Jan C. Brammer, François Lespinasse, Hung Pham, Christopher Schölzel, and S. H. Annabel Chen. NeuroKit2: A Python toolbox for neurophysiological signal processing. *Behavior Research Methods*, 53(4):1689–1696, 2021.

[24] Jarrod R. McClean, Sergio Boixo, Vadim N. Smelyanskiy, Ryan Babbush, and Hartmut Neven. Barren plateaus in quantum neural network training landscapes. *Nature Communications*, 9(1):4812, 2018.

[25] Vankamamidi S. Naresh and Sivaranjani Reddi. Quantum SVM-based predictive analytics: transforming classification methods in healthcare and beyond. *Quantum Information Processing*, 24(9):272, 2025.

[26] Adrián Pérez-Salinas, Alba Cervera-Lierta, Elies Gil-Fuster, and José Ignacio Latorre. Data re-uploading for a universal quantum classifier. *Quantum*, 4:226, 2020.

[27] Deepak Ranga, Aryan Rana, Sunil Prajapat, Pankaj Kumar, Kranti Kumar, and Athanasios V. Vasilakos. Quantum machine learning: Exploring the role of data encoding techniques, challenges, and future directions. *Mathematics*, 12(21):3318, 2024.

[28] Olivier Roy and Martin Vetterli. The effective rank: A measure of effective dimensionality. In *2007 15th European Signal Processing Conference (EUSIPCO)*, pages 606–610, 2007.

[29] Maria Schuld and Nathan Killoran. Quantum machine learning in feature Hilbert spaces. *Physical Review Letters*, 122(4):040504, 2019.

[30] Savita Kumari Sheoran, Vikesh Yadav, and Rakesh Kumar Sheoran. Enhancing data classification with hybrid QNNs and quantum feature maps in google cirq. In *2025 Seventh International Conference on Computational Intelligence and Communication Technologies (CCICT)*, pages 261–267. IEEE, 2025.

[31] Supanut Thanasilp, Samson Wang, M. Cerezo, and Zoë Holmes. Exponential concentration in quantum kernel methods. *Nature Communications*, 15(1):5200, 2024.

[32] Purva Tijare, Naman Kumar Mehta, and Gajendra P. S. Raghava. Benchmarking of quantum SVM and classical ML algorithms for prediction of therapeutic proteins. *bioRxiv*, 2025. Preprint.

[33] Manuela Weigold, Johanna Barzen, Frank Leymann, and Marie Salm. Expanding data encoding patterns for quantum algorithms. In *2021 IEEE 18th International Conference on Software Architecture Companion (ICSA-C)*, pages 95–101. IEEE, 2021.

7 Appendix

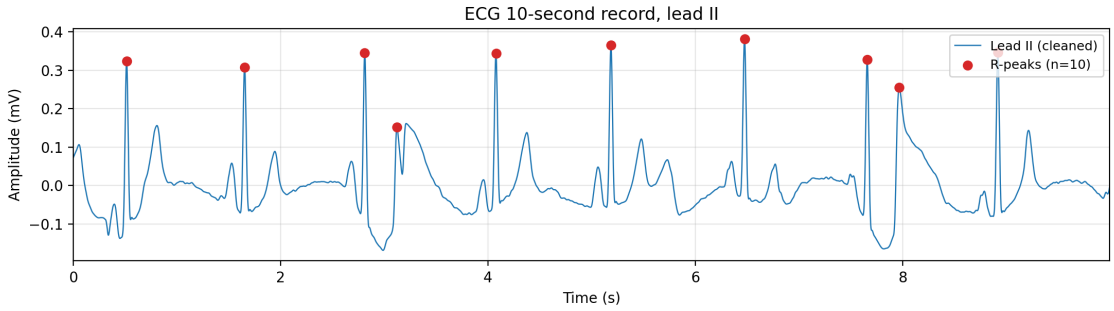

Figure 13: A 10-second Lead II ECG trace. Each heartbeat produces a sharp R peak (marked); the RR interval is the time between two successive R peaks. The RR-interval series is the input representation used in this thesis.

Code	Condition
21	Sinus tachycardia
22	Sinus bradycardia
23	Sinus arrhythmia
30	Atrial premature complex(es)
31	Atrial premature complexes, nonconducted
36	Junctional premature complex(es)
37	Junctional escape complex(es)
50	Atrial fibrillation
51	Atrial flutter
54	Junctional tachycardia
60	Ventricular premature complex(es)
81	AV conduction ratio N:D
83	Second-degree AV block, Mobitz I
84	Second-degree AV block, Mobitz II
85	2:1 AV block
86	AV block, varying conduction
87	AV block, advanced
88	AV block, complete

Table 8: The eighteen AHA/ACC/HRS codes assigned label 1 (rhythm-abnormal). Records carrying only morphological codes (80, 82, 101–108, 120–125, 140–166) are excluded because those abnormalities do not manifest in RR-interval timing.