Adaptive Conformal Prediction for Quantum Machine Learning

Anonymous authors
Paper under double-blind review

Abstract

Quantum machine learning seeks to leverage quantum computers to improve upon classical machine learning algorithms. Currently, robust uncertainty quantification methods remain underdeveloped in the quantum domain, despite the critical need for reliable and trustworthy predictions. Recent work has introduced quantum conformal prediction, a framework that produces prediction sets that are guaranteed to contain the true outcome with user-specified probability. In this work, we formalise how the time-varying noise inherent in quantum processors can undermine conformal guarantees, even when calibration and test data are exchangeable. To address this challenge, we draw on Adaptive Conformal Inference, a method which maintains validity over time via repeated recalibration. We introduce Adaptive Quantum Conformal Prediction (AQCP), an algorithm which preserves asymptotic average coverage guarantees under arbitrary hardware noise conditions. Empirical studies on an IBM quantum processor demonstrate that AQCP achieves target coverage levels and exhibits greater stability than quantum conformal prediction.

1 Introduction

Quantum machine learning (QML) aims to integrate quantum algorithms into broader machine learning pipelines, seeking performance advantages over classical methods on specialised tasks (Biamonte et al., 2017). QML techniques have already proven effective across supervised, unsupervised, and reinforcement learning settings (Havlíček et al., 2019; Otterbach et al., 2017; Jerbi et al., 2021).

Classical machine learning models are increasingly deployed in domains with significant societal implications, for example in justice, healthcare, transportation, and defence (Završnik, 2020; Shailaja et al., 2018; Lindemann et al., 2023; Svenmarck et al., 2018). In such high-stakes settings, erroneous decisions can carry severe consequences, making rigorous uncertainty quantification essential to model evaluation and deployment.

For QML models to become trustworthy tools, they must also incorporate reliable uncertainty quantification. This challenge is heightened by the hardware noise in current noisy intermediate-scale quantum (NISQ) devices, which may vary in both character and severity over time (Proctor et al., 2020).

Conformal prediction stands out as a technique for uncertainty quantification, offering distribution-free, finite-sample guarantees on predictive performance (Caprio, 2025). Its principal function is to post-process predictive model outputs to produce prediction sets that encompass the true outcome with a user-specified probability. Conformal prediction is appealing within the machine learning community for its ability to function as a wrapper around black-box models (Caprio et al., 2025b).

A commonly used variant, known as Split Conformal Prediction, was recently extended to quantum models in Park & Simeone (2023). Their seminal work introduces the Quantum Conformal Prediction (QCP) framework, which is designed to make use of the sampled measurement outcomes of parametrised quantum circuits (PQCs). When tested on quantum hardware, QCP consistently maintained the target coverage level (Park & Simeone, 2023). However, the theoretical guarantees of QCP require a simplifying assumption: that the noise processes within the quantum hardware remain stationary over time.

In this work, we demonstrate without this assumption, one can no longer assert that conformity scores are exchangeable, a property that is necessary for standard conformal guarantees. To address this, we extend Adaptive Conformal Inference (Gibbs & Candes, 2021) to the quantum setting, yielding the Adaptive Quantum Conformal Prediction (AQCP) algorithm. This algorithm incorporates online recalibration to explicitly handle non-stationary hardware noise and exhibits greater stability than QCP in experiments using an IBM quantum processor. AQCP is straightforward to implement and provides reliable representation of uncertainty for QML models under arbitrary hardware noise conditions. Alongside demonstrating AQCP's effectiveness, we provide experimental analysis of various score functions, comparing the average set size they induce when used with AQCP.

1.1 Related Work

Beyond Park & Simeone (2023), which we introduce above, Tasar (2025b) explored an alternative use of conformal prediction for PQCs. In their method, conformal prediction is applied to classical models trained to emulate PQC output distributions using features derived from circuit architecture and gate frequencies. Additionally, Tasar (2025a) investigates conformal prediction in a wide variety of quantum settings, including whether conformal prediction can be used to detect entanglement, the impact of context-conditional exchangeability, and an application to anomaly detection. While these works demonstrate that conformal prediction can be meaningfully applied in quantum contexts, none directly examine how non-stationary model noise affects the exchangeability of conformity scores.

Separately, a growing body of literature has sought to relax the exchangeability requirement between calibration and test data to handle settings such as time series and covariate shift. These methods generally address either specific, known distributional shifts or more general deviations from exchangeability (Tibshirani et al., 2019; Barber et al., 2023; Gibbs et al., 2025; Gibbs & Candes, 2021). We take inspiration from this literature in our approach.

The design of score functions for quantum conformal prediction closely aligns with that of probabilistic conformal prediction (Wang et al., 2023). Park & Simeone (2023) employ a k-nearest neighbour (k-NN) score function, building upon the approach in Wang et al. (2023), who propose a sampling strategy combined with a 1-nearest neighbour (1-NN) score function. Wang et al. (2023) has also influenced methods for other probabilistic models, including the Conformal-Predict-Then-Optimise (CPO) framework (Patel et al., 2024), which similarly employs a k-NN approach. Sample-based strategies have also appeared in the context of conformal risk control (Zecchin et al., 2023), where samples from the model distribution are used to generate prototypical sequences, to which a Euclidean distance score function is then applied.

1.2 Contributions

This paper presents a rigorous framework for reliable uncertainty quantification in quantum machine learning under the presence of realistic, non-stationary hardware noise. Our primary contributions are:

- **Formalisation of Time Dependence:** We develop a theoretical framework demonstrating how non-stationary noise invalidates the exchangeability of non-conformity scores, even when calibration and test data are exchangeable.

- **Adaptive Algorithm for QCP:** We adapt the Adaptive Conformal Inference method (Gibbs & Candes, 2021) to the quantum machine learning setting. This approach, which, we term Adaptive Quantum Conformal Prediction (AQCP), utilises score functions specifically designed to operate on samples from an implicit probability distribution.

- **Comprehensive Hardware Evaluation:** We conduct a thorough experimental evaluation of AQCP's validity and efficiency on IBM quantum hardware, analyse the performance of various sample-based score functions, and show that AQCP maintains the target coverage level with greater stability than the QCP framework.

2 Background

2.1 Conformal Prediction

There are two main variants of conformal prediction: full (or transductive) and split (or inductive). Full conformal prediction is the original variant and is the most data-efficient (Vovk et al., 2005; Shafer & Vovk, 2008; Caprio et al., 2025a). Here we focus on split conformal prediction for its computational efficiency, and we will frequently refer to split conformal prediction as simply conformal prediction.

In split conformal prediction, prediction sets are constructed from a trained model, a calibration dataset $\mathcal{D}_{\text{cal}} = \{(x_i, y_i)\}_{i=1}^n$, and a test feature x_{n+1}. For each element of $\mathcal{D}_{\text{cal}}$, a real-valued score is computed, with higher scores assigned when the model's prediction conforms less to the target. Then, for each element y of the target space, a candidate score is computed from x_{n+1} and y, and from this a decision whether to include y in the prediction set is made. Specifically, if the desired miscoverage rate is $\alpha \in [0, 1]$, then a prediction set is formed by including all elements of the target space whose scores are at most the $(\lceil (n+1)(1-\alpha) \rceil / n)$-quantile of the calibration scores. This process is summarised in the following steps:

1. Compute the calibration score $s_i = \hat{S}(x_i, y_i)$ for each calibration point $(x_i, y_i) \in \mathcal{D}_{\text{cal}}$.
2. Set λ equal to the $\lceil (n+1)(1-\alpha) \rceil$-th smallest value in $\{s_1, \ldots, s_n, +\infty\}$.
3. For a given test input x_{n+1}, construct the prediction set:

$$C(x_{n+1}) = \left\{ y \mid \hat{S}(x_{n+1}, y) \leq \lambda \right\}.$$

Under the weak assumption that the calibration and test points are exchangeable (see Section 2.2), we have the following marginal coverage guarantee, with the lower bound given by Vovk et al. (2005) and the upper bound by Lei et al. (2017).

Theorem 1 (Vovk et al. (2005); Lei et al. (2017)) *If (X_i, Y_i), $i = 1, \ldots, n$ are exchangeable, then for a new exchangeable draw (X_{n+1}, Y_{n+1}),*

$$\mathbb{P}(Y_{n+1} \in C(X_{n+1})) \geq 1 - \alpha.$$

Additionally, if the scores $S_1, \ldots, S_n$ have continuous joint distribution, then we have

$$\mathbb{P}(Y_{n+1} \in C(X_{n+1})) \leq 1 - \alpha + \frac{1}{n+1}.$$

2.2 The Role of Exchangeability in Split Conformal Prediction

Exchangeability is the cornerstone of conformal prediction. This section gives an informal proof of the lower bound of Theorem 1, with particular attention to the role of exchangeability.

A finite set of random variables, $Z_1, \ldots, Z_{n+1}$, in said to be exchangeable if their joint distribution is invariant under any permutation of the indices. Formally, for any permutation $\sigma \in \text{Sym}(n+1)$, the permutation group of order $n+1$,

$$(Z_1, \ldots, Z_{n+1}) \stackrel{d}{=} (Z_{\sigma(1)}, \ldots, Z_{\sigma(n+1)}),$$

where $\stackrel{d}{=}$ denotes equality in distribution. This property is weaker than the i.i.d. assumption but implies that the order of the variables carries no statistical information.

In the context of split conformal prediction, we consider a calibration set $\{(X_i, Y_i)\}_{i=1}^n$ and a new test point (X_{n+1}, Y_{n+1}). If these $n+1$ pairs are exchangeable and we apply a fixed score function $\hat{S}(\cdot, \cdot)$ to each, then the resulting conformity scores $S_1, \ldots, S_{n+1}$ are also exchangeable.

This preservation follows directly from Kuchibhotla (2020, Theorem 3). The theorem shows that a transformation $G : (\mathcal{X} \times \mathcal{Y})^{n+1} \to \mathbb{R}^{n+1}$ preserves exchangeability if it satisfies a specific permutation-equivariance

condition. For any permutation $\pi_1 \in \mathrm{Sym}(n+1)$, there exists a corresponding permutation $\pi_2 \in \mathrm{Sym}(n+1)$ such that:

$$\pi_1 G(w) = G(\pi_2 w), \quad \forall\, w \in (\mathcal{X} \times \mathcal{Y})^{n+1}.$$

In our setting, G corresponds to the map that assigns scores to data points, i.e. $G((X_i, Y_i)_{i=1}^{n+1}) = (\hat{S}(X_i, Y_i))_{i=1}^{n+1}$. Because $\hat{S}$ is applied identically to every calibration and test point, G trivially satisfies the permutation condition, and thus the conformity scores inherit exchangeability from the data.

The key insight is that exchangeability enables a probabilistic argument through ranking. Specifically, because the scores $(S_1, \ldots, S_{n+1})$ are exchangeable, the rank of S_{n+1} among this set is uniformly distributed on $\{1, 2, \ldots, n, n+1\}$. Therefore:

$$\mathbb{P}\left[\mathrm{rank}(S_{n+1}) \leq \lceil (1-\alpha)(n+1) \rceil\right] = \frac{\lceil (1-\alpha)(n+1) \rceil}{n+1} \geq 1 - \alpha.$$

Let $\tilde{\lambda}$ denote the $\lceil (1-\alpha)(n+1) \rceil$-th smallest value among $\{S_1, \ldots, S_{n+1}\}$. We can rewrite the event

$$\{\mathrm{rank}(S_{n+1}) \leq \lceil (1-\alpha)(n+1) \rceil\} = \{S_{n+1} \leq \tilde{\lambda}\}.$$

Furthermore, we can remove dependence on the test score by defining λ as the $\lceil (1-\alpha)(n+1) \rceil$-th smallest value among $\{S_1, \ldots, S_n, +\infty\}$, and observing that

$$S_{n+1} \leq \tilde{\lambda} \quad \Longleftrightarrow \quad S_{n+1} \leq \lambda.$$

Here, the forward implication follows from $\tilde{\lambda} \leq \lambda$, and the reverse from the fact that $S_{n+1} > \lambda$ implies $\lambda = \tilde{\lambda}$. Hence, for any significance level $\alpha \in [0, 1]$, we obtain

$$\mathbb{P}\left[\, S_{n+1} \leq \lambda \,\right] \geq 1 - \alpha.$$

As a result, defining $C(X_{n+1})$ as in Section 2.1 gives the lower bound of Theorem 1. Intuitively, this guarantee holds because exchangeability ensures that the test point has no special status, it is statistically indistinguishable from any point in the calibration set. For a more in-depth studies, see Vovk et al. (2005); Angelopoulos et al. (2023).

2.3 Quantum Machine Learning

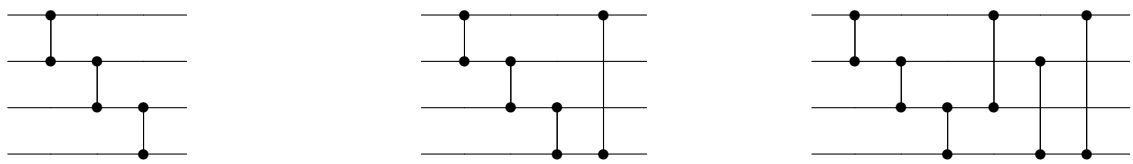
In this work, we employ the classical-data quantum-processing (CQ) paradigm of quantum machine learning, as introduced in Simeone et al. (2022). In this paradigm, classical data are fed into a quantum model, which is trained using a classical optimiser. We use the term *quantum model* to refer to a parametrised quantum circuit (PQC). A PQC applies a unitary transformation $U(\theta)$, dependent on a vector of tunable parameters θ, to a quantum state into which the classical input x is encoded (Simeone et al., 2022).

For a working understanding of the CQ paradigm, two key components warrant further explanation: the design of the unitary transformation via the construction of a PQC, and the encoding of the classical input x into the circuit, referred to as quantum data encoding (Rath & Date, 2024). For a more general introduction to QML landscape see Chang & Cerezo (2025).

2.3.1 PQC Ansätze

In quantum computing, an ansatz defines the foundational structure of a quantum circuit by specifying both the set of gates used and their configuration. This selection is analogous to choosing a model architecture in classical machine learning, where the design profoundly influences the capability and efficiency of the model (Benedetti et al., 2019). Current research efforts focus on identifying optimal ansätze for various applications, particularly in the context of variational quantum algorithms (Qin, 2023). The hardware-efficient ansatz is a subclass of ansatz designs that mitigates the gate overhead incurred during circuit compilation (Leone et al., 2024). It does so by reducing idle qubits and employing native entangling gates inherent to the hardware. This design is especially well suited to today's NISQ devices, where circuit depth and fidelity are constrained.

The hardware-efficient ansatz is constructed as a sequence of layers. Each layer consists of local single-qubit rotations, applied in parallel to every qubit, followed by a fixed entangling-gate pattern applied between specific qubit pairs. The single-qubit rotations are often chosen from a universal set of gates, for example rotations about the X, Y, and Z axes - $R_X(\theta)$, $R_Y(\theta)$, and $R_Z(\theta)$, respectively. The entangling block is typically composed of a fixed pattern of CNOT (CX) or CZ gates that reflect the hardware's connectivity graph. Common entanglement schemes include linear (chain), circular, and all-to-all (full) connectivity (Simeone et al., 2022). See Figure 1 for circuit diagrams of these three entangling-block configurations, each implemented using CZ gates.



(a) Linear (b) Circular (c) Full

Figure 1: Diagrams of three entangling-block configurations within a four-qubit circuit implemented using CZ gates: (a) a linear entangling block, (b) a circular entangling block, and (c) a full entangling block.

2.3.2 Angle Encoding

In any supervised quantum machine learning algorithm, the classical data must somehow be encoded into the quantum state prepared by the PQC. This encoding is an essential step in any CQ framework and can be achieved in several ways. Widely used strategies include basis encoding, which maps classical bits directly onto computational-basis states; amplitude encoding, which embeds a normalised feature vector into the probability amplitudes of a quantum state; and angle encoding, in which data values determine the rotation angles of quantum gates (Rath & Date, 2024; Simeone et al., 2022). In this section, we focus on angle encoding for its relevance to our experimental implementation and that of Park & Simeone (2023).

An angle encoder maps a classical feature vector to a set of rotation angles, each of which parametrises a single-qubit rotation gate within the PQC. In the simplest setting, an n-dimensional input $(x_1, \ldots, x_n)$ is encoded by applying single-qubit rotations on n qubits, where the rotation angle assigned to the ith qubit is determined by the corresponding feature x_i. These local rotations prepare a product state $|\psi(x)\rangle$ whose representation in the Hilbert space reflects the structure of the input vector. Entanglement, if required, is typically introduced through subsequent entangling layers rather than the encoding itself.

2.4 Density Matrices and Noise Channels

To accurately describe quantum model operations and noise in the next section, we require density-matrix and noise-channel formalism, which we briefly introduce here. For a more comprehensive treatment, see Keyl (2002); Wilde (2013); Fano (1957).

A pure quantum state can be described in two equivalent ways: as a state vector $|\psi\rangle$ in a Hilbert space, or more generally as a density matrix ρ. While state vectors provide a convenient representation for pure states, density matrices extend this representation to mixed states, which are probabilistic mixtures of pure states. Formally, a density matrix is a positive semidefinite, Hermitian operator with unit trace (Gustafson & Sigal, 2003):

$$\rho = \sum_i p_i \left|\psi_i\right\rangle \left\langle\psi_i\right|, \quad \sum_i p_i = 1.$$

Where p_i is the probability of the system occupying the pure state $|\psi_i\rangle$. This formalism is essential for modelling realistic quantum systems, as noise and decoherence inevitably lead to mixed states. In the context of QML, density matrices are particularly useful for analysing how data encoding and variational circuits interact with hardware noise.

In current quantum hardware, quantum states are unavoidably affected by noise processes such as decoherence, gate errors, and measurement imperfections. These noise processes can be modelled as quantum

channels, mathematically described by completely positive trace-preserving (CPTP) maps acting on density matrices Nielsen & Chuang (2010, Section 8.3). A quantum channel $\mathcal{E}$ transforms a state ρ as:

$$\rho \mapsto \mathcal{E}(\rho) = \sum_k E_k \rho E_k^\dagger,$$

where $\{E_k\}$ are Kraus operators satisfying $\sum_k E_k^\dagger E_k = I$, and $E_k^\dagger$ denotes the conjugate transpose of E_k. This operator-sum representation, known as the Kraus representation, provides a powerful and general framework for capturing the effects of noise on quantum computations.

Several standard noise channels are commonly used to model realistic quantum hardware (Nielsen & Chuang, 2010):

- **Depolarising channel:** With probability p, the state is replaced by the maximally mixed state, modelling uniform random errors.
- **Phase flip channel:** Randomly introduces phase flips, capturing loss of coherence without affecting populations.
- **Amplitude damping channel:** Describes energy dissipation processes, such as spontaneous emission from an excited state to the ground state.

3 Conformal Prediction Under Non-Stationary Model Noise

Quantum models are also inherently noisy. While it is well established that quantum hardware experiences noise, the extent and nature of non-stationary noise remain active areas of research (He et al., 2024; Proctor et al., 2020). Much of the existing analysis focuses at the gate level, with comparatively less work at the level of full circuits. Notable recent works by Dasgupta & Humble (2020; 2021; 2022) explore this issue in detail.

To mitigate the effects of drift, IBM Quantum systems perform both hourly and daily recalibrations (IBM Quantum, 2025). Furthermore, many current experimental demonstrations rely on recalibrating quantum computers immediately before execution and adjusting them during runtime (He et al., 2024). Sources of non-stationary noise are diverse, including temperature fluctuations, oscillations in control equipment, and ambient laboratory conditions (Proctor et al., 2020). In more extreme cases, cosmic rays originating from outer space have been shown to cause catastrophic multi-qubit errors approximately every ten seconds (McEwen et al., 2022).

Having outlined conformal prediction and quantum machine learning, we now turn to their intersection. We introduce a model for PQC-based learning that incorporates time-dependent noise effects. We then formalise how this temporal variation disrupts the statistical assumptions, particularly the exchangeability of scores, that underlie standard conformal methods, motivating the need for a new approach.

3.1 PQCs as Non-stationary Probabilistic Models

To establish the formal setting, let $\mathcal{X}$ and $\mathcal{Y}$ denote a classical feature and classical target space respectively. For a given feature $x \in \mathcal{X}$, in the angle encoding setting adopted here, a Q-qubit PQC prepares the quantum state

$$|\psi(x)\rangle = \prod_{i=1}^{N_G} U_i(x) \, |0\rangle^{\otimes Q} \, .$$

Each $U_i(x)$ is a unitary operator parametrised by x representing the ideal i^{th} gate and N_G is the number of gates (Nielsen & Chuang, 2010). The specific transformation $U(x) := U_{N_G}(x) \circ \cdots \circ U_1(x)$ depends both on the circuit ansatz and data-encoding scheme chosen (see Section 2.3). Define the superoperator $\mathcal{U}_{x,i}(\cdot) \equiv U_i(x)(\cdot)U_i(x)^\dagger$. In an ideal (noiseless) setting, the resulting state is described by the density matrix

$$\rho(x) = |\psi(x)\rangle \langle\psi(x)| = U(x) \, |0\rangle^{\otimes Q} \langle 0|^{\otimes Q} \, U(x)^\dagger = \mathcal{U}_{x,N_G} \circ \cdots \circ \mathcal{U}_{x,1}(|0\rangle^{\otimes Q} \langle 0|^{\otimes Q}).$$

When using quantum hardware, the measured state deviates from this ideal due to various noise processes, such as gate errors, decoherence, and crosstalk. Following the convention in Endo et al. (2021), the noisy quantum gate can be written as $\mathcal{E}_t \circ U$, with $\mathcal{E}_t$ being the time-dependent noise channel and t indexing the effective execution time of the circuit shot. This yields the noisy output state

$$\rho_{\text{noisy}}(x,t) = \mathcal{E}_{t,N_G} \circ \mathcal{U}_{x,N_G} \circ \cdots \circ \mathcal{E}_{t,1} \circ \mathcal{U}_{x,1}(|0\rangle^{\otimes Q} \langle 0|^{\otimes Q}).$$

While in reality individual gates and readout processes occur at slightly different physical times this coarse timestamp is sufficient to model inter-shot temporal variation. A computational-basis measurement is performed on the noisy state $\rho_{\text{noisy}}(x,t)$. Since the circuit acts on Q qubits, the measurement yields one of the $N = 2^Q$ bitstring outcomes. To interpret bitstring outcomes in the target space $\mathcal{Y}$, we define a task-dependent mapping

$$f : \{0,1\}^Q \to \mathcal{Y}.$$

This mapping distributes bitstrings over a grid in $\mathcal{Y}$, with resolution growing as Q increases. Define the random variable:

$$\hat{Y}_{x,t} = f(b), \quad \text{whenever the measurement at time } t \text{ with classical input } x \text{ yields } b \in \{0,1\}^Q.$$

When dealing with noisy quantum measurements, the most comprehensive approach uses a Positive Operator-Valued Measure (POVM), Nielsen & Chuang (2010, Box 2.5). For a system of Q qubits, the POVM consists of 2^Q elements, denoted as $\{\Pi_j\}$, with each element corresponding to a Q-bit measurement outcome b_j. The probability of getting a specific outcome y is given by:

$$\mathbb{P}(\,\hat{Y}_{x,t} = y \mid X = x) = \sum_{\{j:\, f(b_j)=y\}} \text{Tr}(\rho_{\text{noisy}} \Pi_j),$$

due to Born's rule.

In a perfect, noise-free scenario, each POVM element Π_j is simply the projector $|b_j\rangle \langle b_j|$. However, with possible non-stationary noise in the measurement stage, the Π_j's can be any positive semidefinite, time-dependent operators that satisfy the normalisation condition $\sum_{j=1}^{N} \Pi_j = I$, where I is the identity matrix (Bravyi et al., 2021).

A single execution plus measurement (a shot) at time t involves both the noise from the PQC execution and the measurement. Denote the shot from the distribution $\hat{Y}_{x,t}$ by $\hat{y}$. Collecting M such shots, at times $T = \{t_1, \ldots, t_M\}$ produces the sample multiset $\mathcal{A}_{x,T} = \{\!\{\hat{y}_m\}\!\}_{m=1}^{M}$. We denote the samples as a multiset to represent repeated measurements of same bitstring. Each shot carries its own timestamp t_m, so taking an additional shot, even on the same input, may draw from a different distribution, reflecting the drift in both the execution and measurement noise.

3.2 Consequences of Non-stationary Noise on Split Conformal Prediction

In standard settings, a score function is defined as a mapping

$$\hat{S} : \mathcal{X} \times \mathcal{Y} \to \mathbb{R},$$

which assigns a real-valued score to each feature-target pair (x, y). This is typically used to measure the discrepancy between the model output and an observed value. In our case, however, the situation differs: we obtain a stochastic multiset, $\mathcal{A}_{x,T}$, instead of a single deterministic value. For the score to be a well-defined deterministic function, it is therefore necessary to take $\mathcal{A}_{x,T}$ as an additional input:

$$\hat{S}(x, y\,;\, \mathcal{A}_{x,T}), \quad \text{with } x \in \mathcal{X},\ y \in \mathcal{Y}.$$

A crucial point is that $\mathcal{A}_{x,T}$ is drawn from a distribution that is conditional on the shot times T. As a result, the induced scores are inherently time-dependent if there the noise of quantum hardware changes

across time. This breaks the usual exchangeability assumption, as even if the underlying data (X_i, Y_i) are exchangeable, the augmented observations

$$Z_i = \left(X_i, Y_i; \{\!\{\hat{Y}_{X_i,t}\}\!\}_{t \in T_i}\right),$$

are not, and therefore the corresponding scores:

$$S_i = \hat{S}\Big(X_i, Y_i;\ \{\!\{\hat{Y}_{X_i,t}\}\!\}_{t \in T_i}\Big),$$

are not necessarily exchangeable. While a hypothetical score function could be constructed to remove the time dependency (e.g. $\hat{S}((x, y;\ \{\!\{\hat{Y}_{x,t}\}\!\}_{t \in T}) = \tilde{S}(x, y)$), the conformity score of any QCP procedure is designed to utilise the quantum model's output, and hence should utilise these time-dependent terms.

Without exchangeable scores, we can not assert that the rank of the test score is uniformly distributed on the set $\{1, \ldots, n+1\}$. Consequently, without making the assumption of stationary noise, we cannot obtain guarantees in the form given in 1.

4 Adaptive Quantum Conformal Prediction

The above section motivates the need for new quantum conformal procedures that are robust to non-exchangeable scores. A substantial body of conformal prediction literature already addresses non-exchangeable feature-target data, which commonly arises in time-series settings. These works provide a natural foundation for adaptation.

Existing approaches can be characterised by the restrictions they impose on the type of shift, varying from known covariate shifts (Tibshirani et al., 2019; Gibbs et al., 2025) to unknown joint distribution shifts (Barber et al., 2023; Gibbs & Candes, 2021; Xu & Xie, 2021), and by the finite-sample or asymptotic nature of their guarantees.

This section introduces the Adaptive Quantum Conformal Prediction (AQCP) algorithm, the application of the Adaptive Conformal Inference (ACI) framework (Gibbs & Candes, 2021) to the quantum setting. We choose to modify ACI because it makes no assumptions about the form of the underlying distribution shift. The adjustment of ACI to the quantum setting is straightforward but requires the use of quantum specific score functions and the use of modified assumptions to obtain one of the stronger theoretical guarantees of (see Theorem 2).

An alternative approach, discussed in Appendix A, is to examine the conformal guarantees available to quantum models under the framework of Barber et al. (2023). This framework is particularly relevant as it provides finite-sample guarantees while accommodating arbitrary distribution shift. However, it is not the primary focus here, since obtaining a tight bound within this framework would require quantifying the total variation distance caused by the distribution shift.

4.1 AQCP Algorithm

AQCP operates in an online testing setting where the miscoverage level is dynamically adjusted based on the observed coverage of prediction sets. Starting with an initial calibration set $\mathcal{D}_{\text{cal}}$ of size n (see Section 2.1), the algorithm processes test points sequentially. At the ith test point, AQCP: constructs a prediction set $C(X_{n+i}) \subseteq \mathcal{Y}$ using the current miscoverage level α_i, observes whether the true label Y_{n+i} falls within this set, updates the miscoverage level to α_{i+1} based on the outcome, and appends the calibration set with the newly observed point (X_{n+i}, Y_{n+i}) to inform future prediction sets. This feedback mechanism allows the algorithm to adapt to shifts in the score distribution without making assumptions about the nature of the shift.

Given a desired miscoverage rate $\alpha \in [0, 1]$, the update to the miscoverage is written as

$$\begin{aligned} \alpha_1 &= \alpha, \\ \alpha_{i+1} &= \alpha_i + \gamma\left(\alpha - \text{err}_i\right), \quad i \in \mathbb{N}. \end{aligned}$$

Here, $\gamma > 0$ is a step size hyperparameter and the error function, err_t, is given by

$$\mathrm{err}_i := \begin{cases} 1 & \text{if } Y_{n+i} \notin C(X_{n+i}), \\ 0 & \text{otherwise.} \end{cases}$$

The adjusted value of α_i is then used to construct the prediction set $C(X_{n+i})$ for the next observation. This set contains all outcomes y whose score does not exceed the $(1-\alpha_i)$-quantile of the scores from all previous observations.

$$C(X_{n+i}) = \left\{ y \in \mathcal{Y} \mid \hat{S}\left(X_{n+i}, y\,; \mathcal{A}_{X_{n+i},T_{n+i}}\right) \leq \text{Quantile}\left(1-\alpha_i, \sum_{j=1}^{n+i-1} \delta_{\hat{S}(X_j,Y_j\,;\mathcal{A}_{X_j,T_j})} + \delta_{+\infty}\right) \right\}.$$

See Algorithm 1 for the full statement of this process. The choice of step size is important: if it is too high, the coverage becomes too volatile; if it is too low, the system will not adapt fast enough to the changes in the distribution. In particular choosing $\gamma = 0$ recovers standard split CP with an updating calibration set. Many subsequent papers have extended this algorithm; for instance, Zaffran et al. (2022) and Bhatnagar et al. (2023) choose to vary γ in an online fashion.

Notice that the adaptive parameter α_i is proven to remain within a bounded interval $[-\gamma, 1+\gamma]$ with probability one, ensuring the stability of the method. This stability property is then used to establish a bound on the average miscoverage error over N test points,

$$\left| \frac{1}{N} \sum_{i=1}^{N} \mathrm{err}_i - \alpha \right| \leq \frac{\max\{\alpha_1, 1-\alpha_1\} + \gamma}{N\gamma}.$$

This bounds the difference between the average observed miscoverage and the target miscoverage α. In particular, since the bound decreases inversely with N, it follows that as the number of observations increases, the average miscoverage rate converges almost surely to the desired target α:

$$\lim_{N\to\infty} \frac{1}{N} \sum_{j=1}^{N} \mathrm{err}_j \overset{\text{a.s.}}{=} \alpha$$

These guarantees do not depend on exchangeability. This means all are applicable with no assumptions imposed on the noise induced by the quantum hardware. The trade-off for this greater generality, is that the guarantees are asymptotic in the number of test points. Gibbs & Candes (2021, Theorem 4.1) give a finite-sample guarantee under a set of specific conditions, namely that the distributional shift is fully determined by a hidden Markov model. This, however, is not applicable to our setting, as there is no distributional shift in the feature-target data itself. Instead the shift arises from the time dependence of the shots which parametrise the score function. With a simple adaptation to the assumptions and proof (see Appendix C.1 for the latter), we can state the following theorem.

Algorithm 1: Adaptive Quantum Conformal Prediction (AQCP)

Input : Miscoverage level $\alpha \in [0, 1]$
Score function $\hat{S}(\cdot)$
Initial calibration dataset $\mathcal{D}_{\text{cal}} = \{(x_i, y_i)\}_{i=1}^{n}$
Labelled test stream for adaptation $\mathcal{D}_{\text{test}} = \{(x_i, y_i)\}_{i=n+1}^{n+n'}$
Number of quantum shots $M \geq 1$
Adaptation step size $\gamma > 0$

Output: A sequence of prediction sets for the test stream: $\{C_i(x_i)\}_{i=n+1}^{n+n'}$

Core Functions

Procedure *InitialCalibrate($\mathcal{D}_{cal}$)*:
  for $(x_i, y_i) \in \mathcal{D}_{cal}$ **do**
    $\mathcal{A}_{x_i,T_i} \leftarrow$M shots from PQC with input x_i ;
    Add $\hat{S}(x_i, y_i; \mathcal{A}_{x_i,T_i})$ to $\mathcal{S}$;

Function *GetQuantile($\mathcal{S}, \alpha$)*:
  return $\inf \left\{ q : \frac{1}{|\mathcal{S}|} \sum_{s_i \in \mathcal{S}} \mathbb{1}\{s_i \leq q\} \geq 1 - \alpha \right\}$;

Function *GeneratePredictionSet($x, \lambda, \mathcal{A}_{x_i,T}$)*:
  Initialise prediction set $C(x) \leftarrow \emptyset$;
  foreach *candidate label* $y \in \mathcal{Y}$ **do**
    if $\hat{S}(x, y; \mathcal{A}_{x,T}) \leq \lambda$ **then**
      Add y to $C(x)$;
  return $C(x)$;

Procedure *UpdateState($x_i, y_i, \lambda, \mathcal{A}_{x,T}$)*:
  $s_i \leftarrow \hat{S}(x_i, y_i; \mathcal{A}_{x_i,T_i})$;
  if $s_i > \lambda$ **then**
    err $\leftarrow 1$;
  else
    err $\leftarrow 0$;
  $\alpha \leftarrow \alpha + \gamma(\alpha_1 - \text{err})$;
  Add s_i to $\mathcal{S}$;

Main Algorithm Execution

PredictionSets $\leftarrow \emptyset$;
$\mathcal{S} \leftarrow \emptyset$; ▷ % Score set %
InitialCalibrate($\mathcal{D}_{\text{cal}}$) ;
for $i = n + 1$ **to** $n + n'$ **do**
  $\lambda \leftarrow$ GetQuantile($\mathcal{S}, \alpha$) ;
  $\mathcal{A}_{x_i,T} \leftarrow$M shots from PQC with input x_i ;
  $C_i(x_i) \leftarrow$ GeneratePredictionSet($x_i, \lambda, \mathcal{A}_{x_i,T}$) ;
  Add $C_i(x_i)$ to PredictionSets ;
  UpdateState($x_i, y_i, \lambda, \mathcal{A}_{x_i,T_i}$) ;
return PredictionSets ;

Theorem 2 (Finite Sample Guarantee for AQCP, Adapted from Gibbs & Candes (2021)) *Let (X_i, Y_i) be i.i.d., the quantile estimator be time-independent (fitted on the calibration data where $|\mathcal{D}_{cal}| = n$), and the score functions to be independent when conditioned upon by a hidden Markov chain B_{i-n} with state space $\mathcal{B}$ and a non-zero absolute spectral gap $1-\eta$. We have that (α_i, B_i) forms a Markov chain on $[-\gamma, 1+\gamma] \times \mathcal{B}$, which we must assume has a unique stationary distribution π. We also must define the following:*

$$B := \sup_{b \in \mathcal{B}} \left| \mathbb{E}[err_i \mid B_i = b] - \alpha \right|, \qquad \sigma_B^2 := \mathbb{E}\big[(\mathbb{E}[err_i \mid B_i] - \alpha)^2\big].$$

Assuming the process starts from the stationary distribution with $(\alpha_1, B_1) \sim \pi$, then we can prove the following bound for any $\varepsilon > 0$:

$$\mathbb{P}\left(\left| \frac{1}{N} \sum_{i=1}^{N} err_i - \alpha \right| \geq \varepsilon \right) \leq 2 \exp\Big(-\frac{N\varepsilon^2}{8} \Big) + 2 \exp\left(-\frac{N(1-\eta)\varepsilon^2}{8(1+\eta)\sigma_B^2 + 20B\varepsilon} \right).$$

Although the specific conditions to be met in this theorem are extremely unlikely in our setting, similarly to Gibbs & Candes (2021) we believe that the bound should be representative of the algorithm's behaviour.

4.2 AQCP Score Functions

A core design choice in conformal prediction is that of the score function. Different score functions lead to prediction sets with varying sizes and geometries. Our PQC-based model produces samples from an unknown distribution conditioned upon the model input and time. Therefore, our score functions must use a sample based approach.

Park & Simeone (2023) achieved excellent results using a k-NN score function. For more points of comparison, we turn to the field of probabilistic conformal prediction (Wang et al., 2023; Sadinle et al., 2019; Romano et al., 2020; Angelopoulos et al., 2025). This field defines optimal conditions for a score function, which we discuss in Appendix B. Motivated by that discussion, we consider the following score functions:

- **Euclidean Distance Score:** Let $\bar{y}$ be the mean of the samples $\mathcal{A}_{x,T}$. Define the score: $s_{\text{Euc}}(x, y) = \|y - \bar{y}\|_2$.
- **k-Nearest-Neighbour (k-NN) Score:** Let $\hat{y}_{(k)}$ be the kth nearest neighbour of y among the samples. Define the score: $s_{\text{k-NN}}(x, y) = \|y - \hat{y}_{(k)}\|_2$.
- **Kernel Density Estimation (KDE) Score:** Using a kernel function K and bandwidth h, one first forms a density estimate

  $$\hat{p}(y \mid x, \mathcal{A}_{x,T}) = \frac{1}{Mh} \sum_{m=1}^{M} K\left(\frac{y - \hat{y}_m}{h} \right).$$

  Define the score: $s_{\text{KDE}}(x, y) = -\hat{p}(y \mid x, \mathcal{A}_{x,T})$.
- **High Density Region (HDR) Score.** Similarly, using the density estimate $\hat{p}(y \mid x, \mathcal{A}_{x,T})$ defined above, the HDR score is the estimated probability mass of the region where the density is at least as large as the density at the candidate point y:

  $$s_{\text{HDR}}(x, y) = \int_{\{y' : \hat{p}(y'|x) > \hat{p}(y|x)\}} \hat{p}(y' \mid x) \mathrm{d}y'.$$

5 Experiments

The following experiments implement Adaptive Quantum Conformal Prediction (AQCP) (Algorithm 1) on a univariate multimodal regression task. To test the local coverage properties of AQCP we use data from the `ibm_sherbrooke` quantum processor. Additionally, we investigate the impact of the score function and the number of shots on prediction set size using data from a classical simulator.

All `ibm_sherbrooke` shot data was collected on the 18th of April 2025.

5.1 Experimental Setup

To facilitate comparison with Park & Simeone (2023), we replicate the conditions of their regression task, in which training, calibration and test data are drawn i.i.d. from following distribution:

$$X \sim \mathcal{U}(-10, 10), \qquad Y \mid X = x \sim \tfrac{1}{2}\Big(\mathcal{N}(-\mu(x), 0.05^2) + \mathcal{N}(\mu(x), 0.05^2)\Big),$$

where

$$\mu(x) = \tfrac{1}{2}\sin\Big(\tfrac{4}{5}x\Big) + \tfrac{x}{20}.$$

The QML model is trained on these samples to learn an approximation of the conditional distribution $Y|X$. Datasets are denoted $\mathcal{D}^{\text{tr}} = \{(x_i, y_i)\}_{i=1}^{n_{\text{tr}}}$, $\mathcal{D}^{\text{cal}} = \{(x_i, y_i)\}_{i=1}^{n_{\text{cal}}}$, and $\mathcal{D}^{\text{test}} = \{(x_i, y_i)\}_{i=1}^{n_{\text{test}}}$ for training, calibration, and testing purposes, respectively.

5.1.1 Quantum Model Architecture and Training

To apply AQCP, a trained model must first be obtained. In the classical-data quantum-processing paradigm we follow, this requires specifying a parametrised quantum circuit (PQC), a classical data-encoding scheme, and an optimisation procedure. Background on this is provided in Section 2.3 and our approach is parallels that of Park & Simeone (2023).

The hardware efficient ansatz (HEA), was selected for its versatility, problem-agnostic design, and hardware efficiency (Simeone et al., 2022). We implemented the HEA using $Q = 5$ qubits, consists of $L = 5$ layers applied sequentially to form the unitary operator

$$U(\theta) = U_5(\theta) \cdot U_4(\theta) \cdot U_3(\theta) \cdot U_2(\theta) \cdot U_1(\theta).$$

Each layer was formed of an unparametrised entangling unitary, U_{ent}, and parametrised single qubit Pauli rotation gates:

$$U_l(\boldsymbol{\theta}) = U_{\text{ent}}\big(R_Z(\theta^1_{l,1})R_Y(\theta^2_{l,2})R_Z(\theta^3_{l,3}) \otimes \cdots \otimes R_Z(\theta^1_{l,Q})R_Y(\theta^2_{l,Q})R_Z(\theta^3_{l,Q})\big), \quad l = 1, \ldots, 5,$$

$$U_{\text{ent}} = \prod_{k=1}^{Q-1} C^Z_{k,k+1}.$$

To encode classical input features into the quantum circuit, we employed a learned non-linear angle encoding scheme with data re-uploading. Specifically, we used a classical neural network (NN) with layer architecture $(1, 10, 10, |\boldsymbol{\theta}|)$, where $|\boldsymbol{\theta}| = 3LQ = 75$. To encode a particular feature x, a forward pass through the NN was performed and the output nodes used as rotation angles $\boldsymbol{\theta}$ of the quantum circuit. The NN included bias terms and applied the Exponential Linear Unit (ELU) activation function (Clevert et al., 2015) at each layer. We use $\boldsymbol{W}$ to denote the matrix of trainable weights in the angle encoder, and $\boldsymbol{\theta}_{\boldsymbol{W}}(x)$ to denote the angles produced by a forward pass of the encoder given input x and weights $\boldsymbol{W}$.

All PQC measurements were performed in the computational basis. Bitstring outcomes $b \in \{0, 1\}^Q$ were then mapped to a discrete real-valued grid via $f : \{0, 1\}^Q \to \mathbb{R}$, defined

$$f(b) = y_{\min} + k \cdot \text{bin}(b), \quad \text{with} \quad k = \frac{y_{\max} - y_{\min}}{2^Q - 1}.$$

Here, $\text{bin}(b)$ converts bitstrings to their denary representation, and $[y_{\min}, y_{\max}] = [-1.5, 1.5]$ was chosen to contain the vast majority of the support of the target distribution.

We trained the angle encoder parameters using the TorchQuantum framework (Wang et al., 2022), which integrates the construction and simulation of quantum circuits with PyTorch's automatic gradient computation. While this method of training would not be possible using quantum hardware, alternative methods such as the parameter-shift rule (Park & Simeone, 2023) are available.

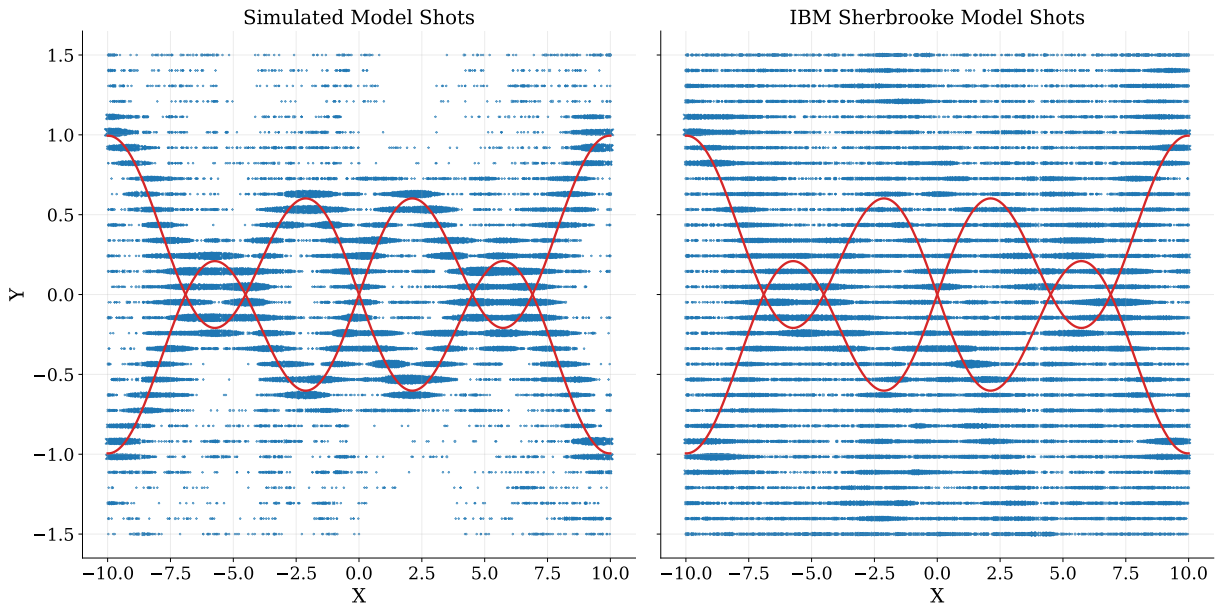


Figure 2: **Regression Model Shots (Simulated vs. `ibm_sherbrooke`).** Comparison of 100,000 shots sampled from each backend (Qiskit Aer simulator and `ibm_sherbrooke`). The marker size is scaled proportionally to the count of overlapping shots at each location. The red lines represent the component mean functions $\mu(x)$ and $-\mu(x)$.

TorchQuantum also provides direct access to the full measurement distribution of the PQC as a probability mass function over bitstrings, enabling the use of a multi-class cross-entropy loss. For a single training example (x_i, y_i), this is defined:

$$\ell(y_i, U(\boldsymbol{\theta}_{\boldsymbol{W}}(x_i))) := -\log\Big(\mathbb{P}(\hat{Y} = y_i \mid x_i)\Big)\,,$$

where $\hat{Y}$ denotes the PQC measurement outcome. All training was performed on a noiseless simulator, so we do not condition on shot time in this instance.

The model was trained to minimise the empirical risk:

$$\hat{R}(\boldsymbol{W}; \mathcal{D}^{\text{tr}}) = \frac{1}{n_{\text{tr}}} \sum_{i=1}^{n_{\text{tr}}} \ell\big(y_i, U\big(\boldsymbol{\theta}_{\boldsymbol{W}}(x_i)\big)\big)\,,$$

with $n_{\text{tr}} = 1{,}000$, a fixed learning rate of 0.01, and 100 epochs.

Figure 2 shows a scatter plot of points taken using the trained model on a noiseless simulator and using the `ibm_sherbrooke` backend. It is clear from the simulated samples that the model reasonably approximates the conditional distribution. While the fundamental structure of the model remains discernable, the samples from `ibm_sherbrooke` show the clear impact hardware noise.

5.1.2 AQCP Implementation

To generate calibration and test data for AQCP coverage, we drew 10,000 samples from the target distribution. For each sample, executed the circuit with the encoded parameters on `ibm_sherbrooke` for 100 shots. These 10,000 circuits were submitted to the device in batches of 1,000. In the efficiency studies, we collected shot data for 10,000 samples, but from the classical simulator `FakeQuitoV2`. All data was collected through the Qiskit library (Javadi-Abhari et al., 2024).

To provide a notion of optimality for the efficiency study, we define $\mathcal{C}^*$ as the collection of prediction sets that minimise the expected prediction set size while maintaining the marginal coverage guarantee. For further detail on this optimality criterion, see Appendix B.1. All score functions were implemented as introduced in Section 4.2. k= $\lceil \sqrt{M} \rceil$ was chosen for the $s_{\text{k-NN}}$ score function, where M is the shot number. This is in line with the choice in Park & Simeone (2023), and with the asymptotic bounds required for consistency. For the s_{KDE} and s_{HDR} score functions, the Gaussian kernel was implemented and the bandwidth was chosen via Silverman's 'rule of thumb'. When calculating scores, additional Gaussian noise with $\sigma = 10^{-4}$ was added to break ties. Without this tie-breaking step, a large number of scores would have been equal due to the discrete nature of the grid mapping to output points.

5.2 Local Coverage Results

The efficacy of the adaptive recalibration mechanism in AQCP is now demonstrated. Figures 3 and 4 illustrate the local coverage over a stream of test points for the various score functions introduced with a moving average. The blue lines represent the standard QCP algorithm operating in an online setting, where the calibration set is sequentially augmented with each test point after its prediction (equivalent to AQCP with an adaptation step size of $\gamma = 0$). The orange lines represent the AQCP algorithm with $\gamma = 0.03$.

Figure 3 presents the results obtained using the k-NN score function. The baseline of $\gamma = 0$ exhibits substantial deviations from the target coverage of α. For example, it over-covers between the 2,000–3,000 test points, and later under-covers around test point 8,500. In contrast, AQCP ($\gamma = 0.03$) shows greater stability. Once the initial rolling window is fully populated, AQCP consistently maintains the average coverage around the desired 90% target level. The algorithm's ability to dynamically adjust its miscoverage estimate allows it to counteract the effects of non-stationary hardware noise, thereby restoring the reliability of the prediction sets. These results show that AQCP is beneficial for achieving stable uncertainty quantification when using current quantum hardware.

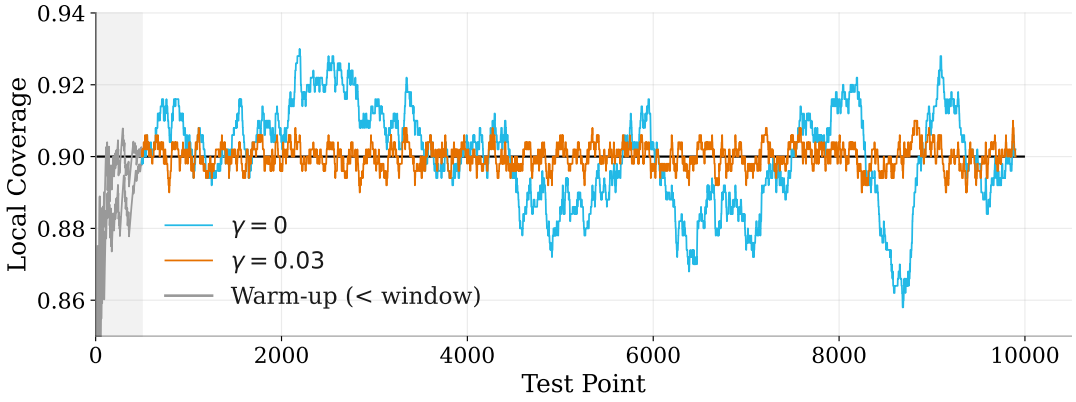
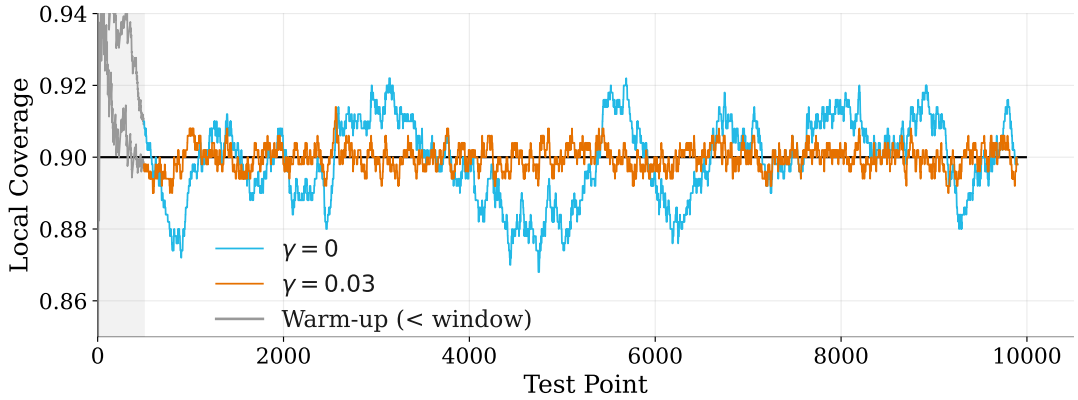
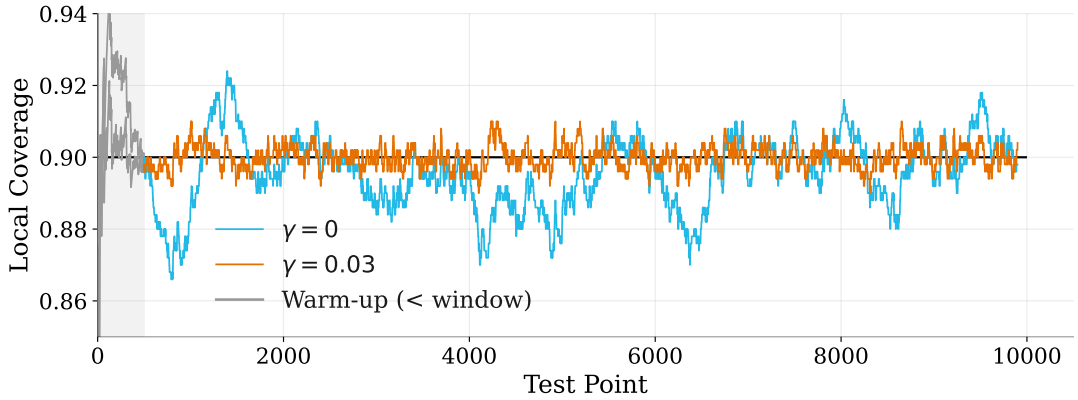


Figure 3: **Moving average coverage of AQCP ($\gamma = 0$, 0.03) on the multimodal regression task using shot data from using shot data from `FakeQuitoV2`.** The k-NN score function on data from `ibm_sherbrooke`. 100 initial calibration points are used with a rolling window of size 500, and a target miscoverage of $\alpha = 0.1$.

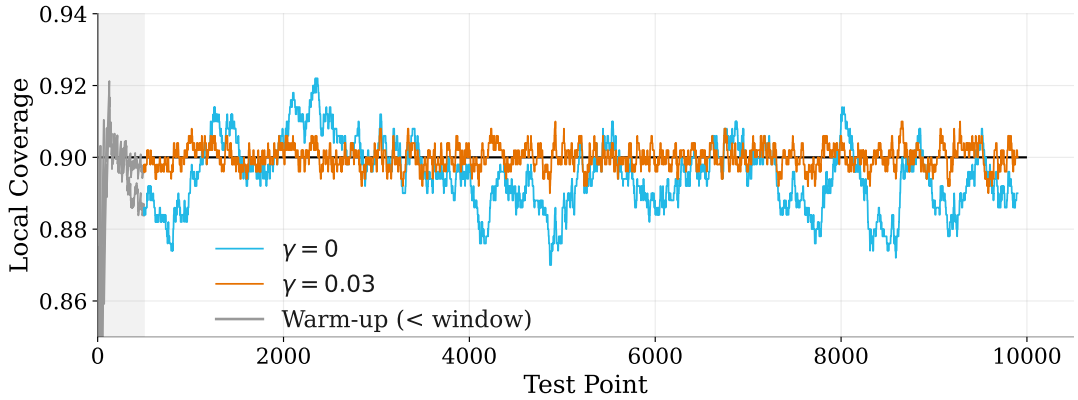
Similarly, Figure 4 demonstrates the robust stabilising behaviour of AQCP across the Euclidean distance (s_{Euc}), Kernel Density Estimation (s_{KDE}), and Highest Density Region (s_{HDR}) score functions. In all cases, the AQCP algorithm ($\gamma = 0.03$) consistently maintains coverage closer to the nominal level than the standard online QCP algorithm ($\gamma = 0$). This demonstrates that the choice of the score function does not substantially influence the coverage stability achieved by AQCP, reiterating the efficacy of the adaptive recalibration mechanism across diverse non-conformity scores.



(a) **Moving average coverage of AQCP ($\gamma = 0,\ 0.03$) on the multimodal regression task using shot data from using shot data from `FakeQuitoV2`.** The EUC score function on data from `ibm_sherbrooke`. 100 initial calibration points are used with a rolling window of size 500, and a target miscoverage of $\alpha = 0.1$.



(b) **Moving average coverage of AQCP ($\gamma = 0,\ 0.03$) on the multimodal regression task using shot data from using shot data from `FakeQuitoV2`.** The KDE score function on data from `ibm_sherbrooke`. 100 initial calibration points are used with a rolling window of size 500, and a target miscoverage of $\alpha = 0.1$.



(c) **Moving average coverage of AQCP ($\gamma = 0,\ 0.03$) on the multimodal regression task using shot data from using shot data from `FakeQuitoV2`.** The HDR score function on data from `ibm_sherbrooke`. 100 initial calibration points are used with a rolling window of size 500, and a target miscoverage of $\alpha = 0.1$.

Figure 4

5.3 Efficiency Results

We now focus on the efficiency of AQCP set predictors, specifically examining the impact of the score function and the number of shots M on the average prediction set size. Since non-stationary noise is not directly relevant to this analysis, all shots were generated using the `FakeQuitoV2` backend. To isolate the effect of the score function and shot number, the step size is fixed at $\gamma = 0.03$. Results are presented as piecewise linear curves of the average set size across ten logarithmically spaced shot values ranging from $M = 1$ to $M = 1{,}000$.

Figure 5(a) presents the efficiency results for our multimodal regression task. All score functions perform similarly for small shot numbers $M \leq 10$, after which their behaviours diverge. s_{KDE} and s_{HDR} produce comparable average set sizes across all values of M, both showing a steady decline of average set size as M increases logarithmically. s_{HDR} achieves the smallest average set size at $M = 1{,}000$. s_{KDE} demonstrates a more rapid decrease in the medium M range but plateaus for $M \geq 100$. In contrast, s_{Euc} and $s_{\text{1-NN}}$ exhibit different behaviour. While comparable performance was observed at small shot numbers ($M \leq 10$), they both produced substantially larger prediction sets across the entire range of M values. At $M = 1{,}000$, s_{Euc} yielded prediction sets approximately 1.5–2 times larger than those produced by s_{HDR}, despite both achieving the target coverage level of 90%, as shown in Figure 5(b).

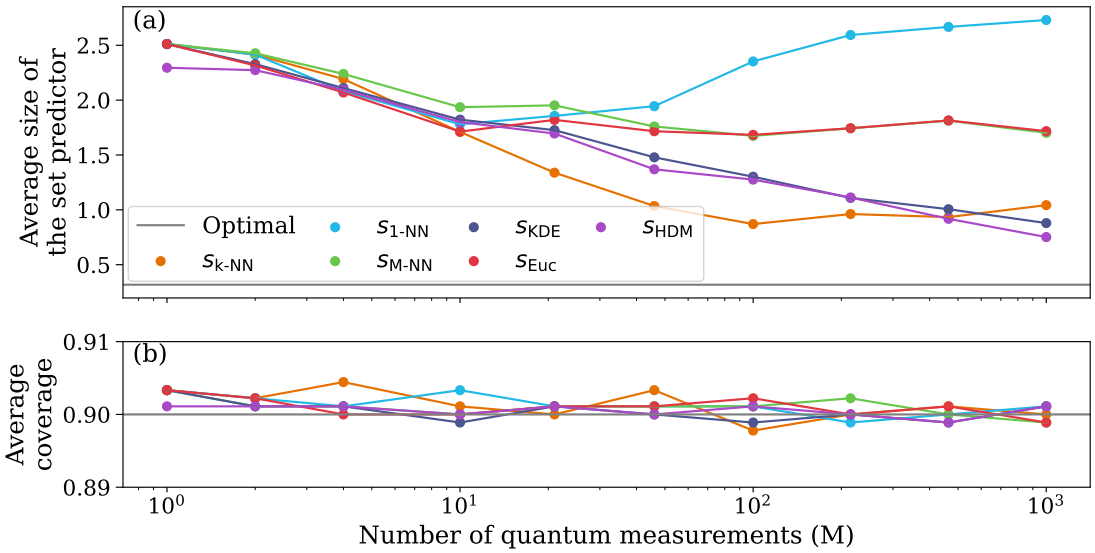


Figure 5: **Average coverage and average set size of AQCP ($\gamma = 0.03$), evaluated across a range of shot numbers M using shot data from `FakeQuitoV2`.** The desired miscoverage is set to $\alpha = 0.1$. Averages are computed from prediction sets returned from Algorithm 1 with 100 initial calibration points and 9,900 test points. (a) Average prediction set size for a range of score functions. (b) Corresponding average coverage for the same score functions. The optimal line represents the performance of the benchmark $\mathcal{C}^*$ family of prediction sets.

6 Conclusion

This work addresses a critical challenge for the reliability of quantum conformal prediction: the non-stationary nature of hardware noise in current-generation quantum processors. The analysis formalised why, without the assumption of stationary noise, the standard conformal guarantee may not hold.

We drew from the existing literature to propose alternative theories that provide conformal guarantees without assuming non-stationary noise. Among these, Gibbs & Candes (2021) emerged as the most applicable

candidate, and their algorithm was implemented using shot data from a trained QML model. The conditions of the multimodal regression experiment in Park & Simeone (2023) were reproduced, and coverage was observed to oscillate closely around the target level for both the QCP framework (with an updating calibration dataset) and the AQCP algorithm. Notably, AQCP exhibited greater stability around the target coverage level, despite the IBM quantum processor data being collected over only a single day. However, confidently attributing this improvement to non-stationary noise would require further experimentation, since fluctuations around the target level are expected when testing with finite samples. Nevertheless, these findings support the use of AQCP for quantum models when an online calibration procedure is feasible.

AQCP was also implemented with a variety of sample-based score functions, including those examined by Park & Simeone (2023). The performance of these score functions aligns with the findings in Park & Simeone (2023), and both newly introduced score functions (kernel density estimation based and high density region based) obtained near-optimal set sizes with a high number of shots. This outcome is also consistent with the theory in Appendix B, where the score function approximates the conditional density of the target distribution. Although the theoretical framework in Appendix B builds on established concepts, it provides a clear formulation that is expected to benefit practitioners of probabilistic and quantum conformal prediction.

Numerous opportunities for future work remain. A growing body of research investigates optimal step-size selection in adaptive conformal prediction (Bhatnagar et al., 2023; Zaffran et al., 2022), and these methods could be explored in quantum settings. Furthermore, if changes in noise over time could be quantified, the theory in Barber et al. (2023) could yield stronger finite-sample guarantees for quantum models.

References

Anastasios N Angelopoulos, Stephen Bates, et al. Conformal prediction: A gentle introduction. *Foundations and Trends in Machine Learning*, 16(4):494–591, 2023.

Anastasios N. Angelopoulos, Rina Foygel Barber, and Stephen Bates. Theoretical foundations of conformal prediction, 2025. URL `https://arxiv.org/abs/2411.11824`.

Rina Foygel Barber, Emmanuel J Candes, Aaditya Ramdas, and Ryan J Tibshirani. Conformal prediction beyond exchangeability. *The Annals of Statistics*, 51(2):816–845, 2023.

Marcello Benedetti, Erika Lloyd, Stefan Sack, and Mattia Fiorentini. Parameterized quantum circuits as machine learning models. *Quantum science and technology*, 4(4):043001, 2019.

Aadyot Bhatnagar, Huan Wang, Caiming Xiong, and Yu Bai. Improved online conformal prediction via strongly adaptive online learning. In *International Conference on Machine Learning*, pp. 2337–2363. PMLR, 2023.

Jacob Biamonte, Peter Wittek, Nicola Pancotti, Patrick Rebentrost, Nathan Wiebe, and Seth Lloyd. Quantum machine learning. *Nature*, 549(7671):195–202, Sep 2017. doi: 10.1038/nature23474.

Sergey Bravyi, Sarah Sheldon, Abhinav Kandala, David C Mckay, and Jay M Gambetta. Mitigating measurement errors in multiqubit experiments. *Physical Review A*, 103(4):042605, 2021.

Michele Caprio. The joys of categorical conformal prediction, 2025. URL `https://arxiv.org/abs/2507.04441`.

Michele Caprio, Yusuf Sale, and Eyke Hüllermeier. Conformal prediction regions are imprecise highest density regions. In Sébastien Destercke, Alexander Erreygers, Max Nendel, Frank Riedel, and Matthias C. M. Troffaes (eds.), *Proceedings of the Fourteenth International Symposium on Imprecise Probabilities: Theories and Applications*, volume 290 of *Proceedings of Machine Learning Research*, pp. 47–59. PMLR, 15–18 Jul 2025a. URL `https://proceedings.mlr.press/v290/caprio25b.html`.

Michele Caprio, David Stutz, Shuo Li, and Arnaud Doucet. Conformalized credal regions for classification with ambiguous ground truth. *Transactions on Machine Learning Research*, 2025b. ISSN 2835-8856. URL `https://openreview.net/forum?id=L7sQ8CW2FY`.

Su Yeon Chang and M. Cerezo. A primer on quantum machine learning, 2025. URL https://arxiv.org/abs/2511.15969.

Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv:1511.07289*, 4(5):11, 2015.

Samudra Dasgupta and Travis S Humble. Characterizing the stability of nisq devices. In *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)*, pp. 419–429. IEEE, 2020.

Samudra Dasgupta and Travis S Humble. Stability of noisy quantum computing devices. *arXiv preprint arXiv:2105.09472*, 2021.

Samudra Dasgupta and Travis S Humble. Characterizing the reproducibility of noisy quantum circuits. *Entropy*, 24(2):244, 2022.

Richard A Davis, Keh-Shin Lii, and Dimitris N Politis. Remarks on some nonparametric estimates of a density function. In *Selected Works of Murray Rosenblatt*, pp. 95–100. Springer, 2011.

Luc Devroye and Clark S Penrod. The strong uniform convergence of multivariate variable kernel estimates. *The Canadian Journal of Statistics/La Revue Canadienne de Statistique*, pp. 211–219, 1986.

Suguru Endo, Zhenyu Cai, Simon C Benjamin, and Xiao Yuan. Hybrid quantum-classical algorithms and quantum error mitigation. *Journal of the Physical Society of Japan*, 90(3):032001, 2021.

Ugo Fano. Description of states in quantum mechanics by density matrix and operator techniques. *Reviews of modern physics*, 29(1):74, 1957.

Isaac Gibbs and Emmanuel Candes. Adaptive conformal inference under distribution shift. *Advances in Neural Information Processing Systems*, 34:1660–1672, 2021.

Isaac Gibbs, John J Cherian, and Emmanuel J Candès. Conformal prediction with conditional guarantees. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, pp. qkaf008, 2025.

Stephen J Gustafson and Israel Michael Sigal. Density matrices. In *Mathematical Concepts of Quantum Mechanics*, pp. 95–114. Springer, 2003.

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.

Zichang He, Bo Peng, Yuri Alexeev, and Zheng Zhang. Distributionally robust variational quantum algorithms with shifted noise. *IEEE Transactions on Quantum Engineering*, 2024.

Rob J Hyndman. Computing and graphing highest density regions. *The American Statistician*, 50(2): 120–126, 1996.

IBM Quantum. Calibration jobs. https://docs.quantum.ibm.com/admin/calibration-jobs, 2025. Accessed: April 11, 2025.

Ali Javadi-Abhari, Matthew Treinish, Kevin Krsulich, Christopher J. Wood, Jake Lishman, Julien Gacon, Simon Martiel, Paul D. Nation, Lev S. Bishop, Andrew W. Cross, Blake R. Johnson, and Jay M. Gambetta. Quantum computing with Qiskit, 2024.

Sofiene Jerbi, Casper Gyurik, Simon Marshall, Hans Briegel, and Vedran Dunjko. Parametrized quantum policies for reinforcement learning. *Advances in Neural Information Processing Systems*, 34:28362–28375, 2021.

Yuko Kato, David MJ Tax, and Marco Loog. A review of nonconformity measures for conformal prediction in regression. *Conformal and Probabilistic Prediction with Applications*, pp. 369–383, 2023.

Michael Keyl. Fundamentals of quantum information theory. *Physics reports*, 369(5):431–548, 2002.

Arun Kumar Kuchibhotla. Exchangeability, conformal prediction, and rank tests. *arXiv preprint arXiv:2005.06095*, 2020.

Jing Lei. Classification with confidence. *Biometrika*, 101(4):755–769, 2014.

Jing Lei, Max G'Sell, Alessandro Rinaldo, Ryan J. Tibshirani, and Larry Wasserman. Distribution-free predictive inference for regression, 2017. URL `https://arxiv.org/abs/1604.04173`.

Lorenzo Leone, Salvatore FE Oliviero, Lukasz Cincio, and Marco Cerezo. On the practical usefulness of the hardware efficient ansatz. *Quantum*, 8:1395, 2024.

Lars Lindemann, Matthew Cleaveland, Gihyun Shim, and George J Pappas. Safe planning in dynamic environments using conformal prediction. *IEEE Robotics and Automation Letters*, 8(8):5116–5123, 2023.

Don O Loftsgaarden and Charles P Quesenberry. A nonparametric estimate of a multivariate density function. *The Annals of Mathematical Statistics*, 36(3):1049–1051, 1965.

Matt McEwen, Lara Faoro, Kunal Arya, Andrew Dunsworth, Trent Huang, Seon Kim, Brian Burkett, Austin Fowler, Frank Arute, Joseph C Bardin, et al. Resolving catastrophic error bursts from cosmic rays in large arrays of superconducting qubits. *Nature Physics*, 18(1):107–111, 2022.

Michael A Nielsen and Isaac L Chuang. *Quantum computation and quantum information*. Cambridge university press, 2010.

Johannes S Otterbach, Riccardo Manenti, Nasser Alidoust, A Bestwick, M Block, B Bloom, S Caldwell, N Didier, E Schuyler Fried, S Hong, et al. Unsupervised machine learning on a hybrid quantum computer. *arXiv preprint arXiv:1712.05771*, 2017.

Sangwoo Park and Osvaldo Simeone. Quantum conformal prediction for reliable uncertainty quantification in quantum machine learning. *IEEE Transactions on Quantum Engineering*, 5:1–24, 2023.

Emanuel Parzen. On estimation of a probability density function and mode. *The annals of mathematical statistics*, 33(3):1065–1076, 1962.

Yash P Patel, Sahana Rayan, and Ambuj Tewari. Conformal contextual robust optimization. In *International Conference on Artificial Intelligence and Statistics*, pp. 2485–2493. PMLR, 2024.

Timothy Proctor, Melissa Revelle, Erik Nielsen, Kenneth Rudinger, Daniel Lobser, Peter Maunz, Robin Blume-Kohout, and Kevin Young. Detecting and tracking drift in quantum information processors. *Nature communications*, 11(1):5396, 2020.

Junhan Qin. Review of ansatz designing techniques for variational quantum algorithms. *Journal of Physics: Conference Series*, 2634(1):012043, nov 2023. doi: 10.1088/1742-6596/2634/1/012043. URL `https://doi.org/10.1088/1742-6596/2634/1/012043`.

Minati Rath and Hema Date. Quantum data encoding: A comparative analysis of classical-to-quantum mapping techniques and their impact on machine learning accuracy. *EPJ Quantum Technology*, 11(1):72, 2024.

Alfred Rényi. *Foundations of probability*. Courier Corporation, 2007.

Yaniv Romano, Matteo Sesia, and Emmanuel Candes. Classification with valid and adaptive coverage. *Advances in neural information processing systems*, 33:3581–3591, 2020.

Mauricio Sadinle, Jing Lei, and Larry Wasserman. Least ambiguous set-valued classifiers with bounded error levels. *Journal of the American Statistical Association*, 114(525):223–234, 2019.

Glenn Shafer and Vladimir Vovk. A tutorial on conformal prediction. *Journal of Machine Learning Research*, 9(3), 2008.

K. Shailaja, B. Seetharamulu, and M. A. Jabbar. Machine learning in healthcare: A review. In *2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, pp. 910–914, 2018. doi: 10.1109/ICECA.2018.8474918.

Osvaldo Simeone et al. An introduction to quantum machine learning for engineers. *Foundations and Trends in Signal Processing*, 16(1-2):1–223, 2022.

Peter Svenmarck, Linus Luotsinen, Mattias Nilsson, and Johan Schubert. Possibilities and challenges for artificial intelligence in military applications. In *Proceedings of the NATO Big Data and Artificial Intelligence for Military Decision Making Specialists' Meeting*, volume 1, 2018.

Davut Emre Tasar. Conformal prediction for quantum entanglement: Robustness and detection. 2025a.

Emre Tasar. Quantum-enhanced conformal methods for multi-output uncertainty: A holistic exploration and experimental analysis. *arXiv preprint arXiv:2501.10414*, 2025b.

Ryan J Tibshirani, Rina Foygel Barber, Emmanuel Candes, and Aaditya Ramdas. Conformal prediction under covariate shift. *Advances in neural information processing systems*, 32, 2019.

Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. *Algorithmic learning in a random world*. Springer, 2005.

Hanrui Wang, Yongshan Ding, Jiaqi Gu, Zirui Li, Yujun Lin, David Z Pan, Frederic T Chong, and Song Han. Quantumnas: Noise-adaptive search for robust quantum circuits. In *The 28th IEEE International Symposium on High-Performance Computer Architecture (HPCA-28)*, 2022.

Zhendong Wang, Ruijiang Gao, Mingzhang Yin, Mingyuan Zhou, and David Blei. Probabilistic conformal prediction using conditional random samples. In *International Conference on Artificial Intelligence and Statistics*, pp. 8814–8836. PMLR, 2023.

Mark M Wilde. *Quantum information theory*. Cambridge university press, 2013.

Chen Xu and Yao Xie. Conformal prediction interval for dynamic time-series. In *International Conference on Machine Learning*, pp. 11559–11569. PMLR, 2021.

Margaux Zaffran, Olivier Féron, Yannig Goude, Julie Josse, and Aymeric Dieuleveut. Adaptive conformal predictions for time series. In *International Conference on Machine Learning*, pp. 25834–25866. PMLR, 2022.

Aleš Završnik. Criminal justice, artificial intelligence systems, and human rights. *ERA Forum*, 20(4):567–583, March 2020. ISSN 1863-9038. doi: 10.1007/s12027-020-00602-0. URL `https://doi.org/10.1007/s12027-020-00602-0`.

Matteo Zecchin, Sangwoo Park, and Osvaldo Simeone. Forking uncertainties: Reliable prediction and model predictive control with sequence models via conformal risk control, 2023. URL `https://arxiv.org/abs/2310.10299`.

Puning Zhao and Lifeng Lai. Analysis of knn density estimation. *IEEE Transactions on Information Theory*, 68(12):7971–7995, 2022.

A Appendix: Conformal Prediction Beyond Exchangeability

We follow the procedure in Barber et al. (2023) making the necessary adaptions to our setting. Denote $Z_i = (X_i, Y_i; \mathcal{A}_{X_i,T_i})$. A weight $w_i \in [0,1]$ is assigned to each data point to quantify its similarity to a given test point. These weights can be derived from various metrics, such as the temporal gap between the observation and the test instance. The underlying intuition is to assign higher weights to data points

presumed to share the same distribution as the test point, $(X_{n+1}, Y_{n+1}; \mathcal{A}_{X_{n+1},T_{n+1}})$, and lower weights to those from different distributions. The weights are subsequently normalised as follows:

$$\tilde{w}_i = \frac{w_i}{\sum_{j=1}^{n} w_j + 1}, \quad \text{for} \quad i = 1, ..., n \qquad \tilde{w}_{n+1} = \frac{1}{\sum_{j=1}^{n} w_j + 1}$$

Barber et al. (2023) provide equivalent theoretical bounds for both non-symmetric full conformal and split-conformal algorithms. We will concentrate only on the split-conformal case. The non-exchangeable split conformal set is given by:

$$C(X_{n+1}) = \{y \in \mathcal{Y} \mid \hat{S}(X_{n+1}, y\,; \mathcal{A}_{X_{n+1},T_{n+1}}) \leq \text{Quantile}\left(1-\alpha, \sum_{i=1}^{n} \tilde{w}_i \cdot \delta_{\hat{S}(Z_i)} + \tilde{w}_{n+1} \cdot \delta_{+\infty}\right)\}$$

With this prediction set, we can obtain the finite sample guarantee from Barber et al. (2023)

$$\mathbb{P}[Y_{n+1} \in C_n(X_{n+1})] \geq 1 - \alpha - \sum_{i=1}^{n} \tilde{w}_i \cdot \mathrm{d_{TV}}(S(Z), S(Z^i))$$

where $Z = (Z_1, \ldots, Z_{n+1})$ represents the original data sequence, $Z^i = (Z_1, \ldots, Z_{i-1}, Z_{n+1}, Z_{i+1}, \ldots, Z_n, Z_i)$ represents the same sequence but with the ith and $(n+1)^{th}$ observations swapped, $S(z) \in \mathbb{R}^{n+1}$ is the residual vector with entries $(S(z))_i = \hat{S}(x_i, y_i; \mathcal{A}_{x_i,T_i})$, and $\mathrm{d_{TV}}$ denotes the total variation distance. Their method protects against shifts in the distributions of the Z_i's and, as a consequence, shifts in the scores.

The method provides a coverage guarantee of at least $1-\alpha$ minus a specific correction term. This correction reflects how much the data deviates from exchangeability, weighted by the importance given to each observation. If the data are truly exchangeable, this correction becomes zero, restoring the standard $1-\alpha$ guarantee. However, in QML, the size of this correction is in practice unknown. This prevents the ability to adjust the weights to counteract large distributional shifts, meaning no practical lower bound on coverage can be given without making further assumptions.

B Appendix: Score Functions

The effectiveness of any conformal method, specifically the size and informativeness of the resulting prediction sets, is highly dependent on the score function. This section therefore delves into the theory of optimal score functions to construct prediction sets that are as small as possible while maintaining coverage. We then introduce practical, sample-based score estimators suitable for the probabilistic output of a quantum model.

B.1 What makes a score function optimal?

The literature on conformal prediction addresses a wide range of objectives. In this work, we follow Angelopoulos et al. (2025), in which the authors focus on two of the most natural cases: constructing prediction sets that are as small as possible in expectation, while ensuring either marginal or conditional validity.

Formally, let X and Y be random variables taking values in measurable spaces $(\mathcal{X}, \mathcal{F}_X)$ and $(\mathcal{Y}, \mathcal{F}_Y)$ respectively, with joint distribution $P_{X,Y}$ and marginals P_X, P_Y. Given a miscoverage rate $\alpha \in [0,1]$, consider the following optimisation problems defined over joint prediction sets $B \subseteq \mathcal{X} \times \mathcal{Y}$:

1. **Marginal Coverage Objective:**

$$\operatorname*{argmin}_B \mathbb{E}_{X \sim P_X}\left[|B(X)|\right] \quad \text{subject to} \quad \mathbb{P}\big(Y \in B(X)\big) \geq 1 - \alpha.$$

2. **Conditional Coverage Objective:**

$$\operatorname*{argmin}_B \mathbb{E}_{X \sim P_X}\left[|B(X)|\right] \quad \text{subject to} \quad \mathbb{P}\big(Y \in B(X) \mid X = x\big) \geq 1 - \alpha, \quad \forall x \in \mathcal{X}.$$

Where $B(x) = \{y : (x, y) \in B\}$ denotes the prediction set for input x.

To perform rigorous analysis of these optimisation problems and derive optimal score functions, we establish the following probabilistic framework. This framework is essential because optimal score functions will be characterised in terms of conditional densities, and the expected size objective requires integration with respect to these measures.

We define $(\Omega, \mathcal{F}, \mathbb{P})$ to be a probability space, and view X and Y as measurable functions:

$$X : \Omega \to \mathcal{X}, \qquad Y : \Omega \to \mathcal{Y}.$$

Denote $\mu_{X,Y} := \mathbb{P} \circ (X, Y)^{-1}$ to be the joint distribution of (X, Y), and $\mu_X := \mathbb{P} \circ X^{-1}$, $\mu_Y := \mathbb{P} \circ Y^{-1}$ to be the marginal distributions of X and Y respectively.

Assume that $\mu_{X,Y}$ is absolutely continuous with respect to the Lebesgue product measure m on $(\mathcal{X} \times \mathcal{Y}, \mathcal{F}_X \otimes \mathcal{F}_Y)$, so that the joint density exists:

$$p(x, y) = \frac{\mathrm{d}\mu_{X,Y}}{\mathrm{d}m}.$$

Consequently, the marginals μ_X, μ_Y are also absolutely continuous with respect to the individual Lebesgue measures (Rényi, 2007), and admit densities:

$$p_X(x) = \frac{\mathrm{d}\mu_X}{\mathrm{d}m_{\mathcal{X}}}, \qquad p_Y(y) = \frac{\mathrm{d}\mu_Y}{\mathrm{d}m_{\mathcal{Y}}}.$$

This allows us to express the conditional densities:

$$p(y \mid x) = \frac{p(x, y)}{p_X(x)}, \qquad p(x \mid y) = \frac{p(x, y)}{p_Y(y)}.$$

These conditional densities will play a central role in characterising optimal score functions, as we shall see in sections B.2 and B.3.

In conformal prediction, the prediction sets are typically constructed as level sets of a score function $\hat{S}(x, y)$:

$$C(x) = \{y \in \mathcal{Y} \mid \hat{S}(x, y) \leq \lambda\},$$

where λ is a threshold calibrated to achieve the desired coverage. This construction is without loss of generality: any solution $B \subseteq \mathcal{X} \times \mathcal{Y}$ to the optimisation problems above can be written in level set form. For example, choosing $\hat{S}(x, y) = 1 - \mathbb{1}_{B(x)}(y)$ and $\lambda = 0$ gives

$$C(x) = \{y \mid 1 - \mathbb{1}_{B(x)}(y) \leq 0\} = B(x).$$

Thus, the optimisation problems can be equivalently reformulated as the task of identifying score functions which, when paired with suitable thresholds, yield prediction sets that are optimal for either Problem 1 or Problem 2. This motivates the definition of score function classes $\mathcal{S}_i$ such that, for some $\lambda \in \mathbb{R}$, the set C_λ solves the corresponding optimisation problem i.

In the following section, we characterise the optimal score function class for optimisation problem 1 and 2 and develop practical estimators for our quantum learning setting.

B.2 Optimal Scores for Marginal Coverage ($\mathcal{S}_1$)

By adapting arguments from Lei (2014); Sadinle et al. (2019); Kato et al. (2023), we are able to fully characterise the class corresponding to marginal validity.

Theorem 3 *A score function $\hat{S}$ belongs to $\mathcal{S}_1$ if and only if there exists a strictly increasing function $\phi : (-\infty, 0] \to \mathbb{R}$ such that for all $(x, y) \in \mathcal{X} \times \mathcal{Y}$,*

$$\hat{S}(x, y) = \phi(-p(y \mid x)).$$

For validity, assume that there exists $\lambda \in \mathbb{R}$ such that $P(C_\lambda) = 1 - \alpha$. If no such threshold exists, a standard randomisation rule can be applied.[1]

Theorem 3 states that any score function that is a strictly increasing transformation of the negative probability density, is optimal for producing minimal prediction sets with marginal coverage. By exploring various choices for ϕ, score functions that belong to these classes can be recovered, ranging from well-known forms to more nuanced variants. For instance, applying $\phi(x) = x$ yields $\hat{S} = -\hat{p}(y \mid x)$; applying $\phi(x) = -x^{-1}$ (with appropriate handling at $x = 0$) yields $\hat{S} = \hat{p}(y \mid x)^{-1}$; and choosing $\phi(x) = -\log(-x)$ (again with appropriate handling at $x = 0$) produces the negative log density score.

In many machine learning settings, any of these forms can be implemented directly. For example, they are natural for classification tasks with softmax probabilities over output classes (Angelopoulos et al., 2023), and they are also appropriate for regression when the model provides an explicit conditional density function. In our setting, the true conditional density $p(y \mid x)$ is unknown and hence it must be estimated from a model.

In classical regression over $\mathbb{R}^n$, there is a particularly appealing connection to the widely used Euclidean distance score function:

$$s_{\text{Euc}}(x, y) = \|y - f(x)\|_2,$$

where $f(x)$ is a point prediction model. Suppose $s_{\text{Euc}} \in \mathcal{S}_1$. This means that there exists ϕ such that for all $x \in \mathcal{X}$,

$$\phi\big(-p(y \mid x)\big) = \|y - f(x)\|_2.$$

Since ϕ is strictly increasing, its inverse ϕ^{-1} exists and is also strictly increasing. This implies that for all x,

$$p(y \mid x) = -\phi^{-1}\Big(\|y - f(x)\|_2\Big).$$

The converse also holds. Therefore, the commonly used score function $\|y - f(x)\|_2$ is optimal for marginal coverage in the sense of minimisation problem 1 if and only if the point prediction $f(x)$ corresponds to a model distribution that is radially symmetric about its prediction uniformly across all x, with density decreasing as the distance from the prediction increases. This condition is satisfied, for example, when $f(x)$ is the conditional mean of a homoscedastic Gaussian model, but it fails in the presence of skewed conditional distributions or heteroscedasticity.

To deal with the case of more general distributions we look towards probability density estimators. For example, given an appropriate choice for k, $s_{\text{k-NN}}$ is a proxy for the negative k-NN density estimator (Zhao & Lai, 2022; Loftsgaarden & Quesenberry, 1965). This means that $s_{\text{k-NN}}$ asymptotically approaches the optimal class $\mathcal{S}_1$ as the number of samples $M \to \infty$. Similarly, since kernel density estimation is a consistent density estimator under appropriate choice of h (Parzen, 1962; Devroye & Penrod, 1986; Davis et al., 2011), then s_{KDE} also asymptotically lies in $\mathcal{S}_1$. This theory assumes access to samples taken from the true conditional distributions, however, it provides motivation the general case.

B.3 Optimal Scores for Conditional Coverage ($\mathcal{S}_2$)

For the conditional guarantee optimisation problem, a similar approach can be taken, but more machinery is required to characterise the full family of optimal score functions. We first recall the high–density level set (Hyndman, 1996):

$$H_x(\lambda) = \{y \in \mathcal{Y} \mid p(y \mid x) \geq \lambda\}.$$

This is the set of all outcomes y that are at least as probable as the threshold λ. Using this gives rise to the next theorem, which describes the structure of all score functions that attain the conditional guarantee.

Theorem 4 *A score function $\hat{S}$ belongs to $\mathcal{S}_2$ if and only if there exists a strictly increasing function $\phi : [0, 1] \to \mathbb{R}$ such that for all $(x, y) \in \mathcal{X} \times \mathcal{Y}$*

$$\hat{S}(x, y) = \phi\left(\int_{H_x(p(y|x))} p(\hat{y} \mid x) \mathrm{d}\hat{y}\right).$$

[1]For the proof see Appendix C.2. A similar argument is also presented in Sadinle et al. (2019) but for the case of discrete $\mathcal{Y}$.

Moreover validity (at level $1-\alpha$) is obtained provided that, for every $x \in \mathcal{X}$, the set $C_{\phi(1-\alpha)}(x) = \{y : \hat{S}(x,y) \geq \phi(1-\alpha)\}$ satisfies $\mathbb{P}\big(Y \in C_{\phi(1-\alpha)}(x) \mid X = x\big) = 1-\alpha$. If this equality fails, a standard randomisation rule can be applied to achieve exact coverage.[2]

Compared with the simpler family $\mathcal{S}_1$, this form is more abstract and can be harder to reduce to immediately implementable scores. However, in parametric models the level-set probability often has a closed form. For example, in regression over $\mathbb{R}^n$, if for each x we have

$$Y|_{X=x} \sim \mathcal{N}(\mu(x), \sigma^2(x)),$$

then the level set corresponding to density value $p(y \mid x)$ is the symmetric interval around $\mu(x)$ with radius $\|y-\mu(x)\|$. Hence,

$$\mathbb{P}\big[Y \in H_X(p(y \mid X)) \mid X = x\big] = 2\Phi\Big(\tfrac{\|y-\mu(x)\|}{\sigma(x)}\Big) - 1,$$

where Φ denotes the standard normal CDF. Therefore any valid score in this family has the form

$$\hat{S}(x,y) = \phi\Big(2\Phi\big(\tfrac{\|y-\mu(x)\|}{\sigma(x)}\big) - 1\Big),$$

for some strictly increasing $\phi : [0,1] \to \mathbb{R}$. In particular, choosing

$$\phi(z) = \Phi^{-1}\Big(\tfrac{z+1}{2}\Big),$$

yields the score

$$\hat{S}(x,y) = \tfrac{\|y-\mu(x)\|}{\sigma(x)},$$

which parallels the result from Section B.2. Moreover, this construction extends to any symmetric distribution with a strictly monotonic probability density function in the radial distance from the mean.

When the form of the conditional distribution is unknown, we again revert back to a sample based probability density estimator. Using this estimator and taking $\phi(x) = x$ we then obtain the s_{HDR} score. This score will asymptotically approaches $\mathcal{S}_2$ as $M \to \infty$ given a consistent density estimator.

C Proofs

Lemma 1 ***(Neyman-Pearson)*** *Let f and g be two non-negative measurable functions, then the optimiser of the problem*

$$\min_{\mathcal{C}} \int_{\mathcal{C}} g \quad \textit{subject to} \quad \int_{\mathcal{C}} f \geq 1-\alpha,$$

is given by $\mathcal{C} = \{f/g \geq \lambda\}$ if there exists λ such that $\int_{f/g \geq \lambda} f = 1-\alpha$.

C.1 Proof of Theorem 2

The argument follows Appendix A.7 of Gibbs & Candes (2021), with the environment chain A_t replaced by the score chain B_t. We can write

$$\mathbb{P}\left(\left|\frac{1}{T}\sum_{t=1}^{T} \text{err}_t - \alpha\right| > \varepsilon\right) \leq \mathbb{P}\left(\left|\frac{1}{T}\sum_{t=1}^{T}(\text{err}_t - \mathbb{E}[\text{err}_t \mid B_t])\right| > \tfrac{\varepsilon}{2}\right) + \mathbb{P}\left(\left|\frac{1}{T}\sum_{t=1}^{T}(\mathbb{E}[\text{err}_t \mid B_t] - \alpha)\right| > \tfrac{\varepsilon}{2}\right).$$

[2]For the proof see Appendix C.3. A similar argument is also presented in Angelopoulos et al. (2025); Romano et al. (2020) but for the case of discrete $\mathcal{Y}$.

We will first bound the first term using the Hoeffding bound. The proof of Lemma A.2 in the original text uses only the binary-valued errors and the monotonicity of α_t in past errors, $\sum_{s=1}^{t-1} \mathrm{err}_s$. This argument is unchanged here, yielding

$$\mathbb{P}\left(\left|\frac{1}{T}\sum_{t=1}^{T}(\mathrm{err}_t - \mathbb{E}[\mathrm{err}_t \mid B_t])\right| > \tfrac{\varepsilon}{2}\right) \leq 2\exp\bigl(-T\varepsilon^2/8\bigr).$$

Now we focus on bounding the second term. Define $f(b) = \mathbb{E}[\mathrm{err}_t \mid B_t = b] - \alpha$. Then f is bounded in $[-B, B] = [-1, 1]$, mean-zero under stationarity, and the variance proxy is σ_B^2. Applying the Bernstein inequality for Markov chains (Theorem A.1 in the original text) yields

$$\mathbb{P}\left(\left|\frac{1}{T}\sum_{t=1}^{T}(\mathbb{E}[\mathrm{err}_t \mid B_t] - \alpha)\right| > \tfrac{\varepsilon}{2}\right) \leq 2\exp\left(-\frac{T(1-\eta)\varepsilon^2}{8(1+\eta)\sigma_B^2 + 20B\varepsilon}\right).$$

Combining the two bounds proves the theorem.

C.2 Proof of Theorem 3

For each $x \in \mathcal{X}$, define the high-density region

$$H(t) := \{(x, y) \in \mathcal{X} \times \mathcal{Y} : \; p(y \mid x) \geq t\},$$

and define

$$h(t) := \int_{H(t)} p(x, y)\mathrm{d}x\mathrm{d}y$$

Note that h is decreasing with $h(0) = 1$ and $\lim_{t\to\infty} h(t) = 0$.

We first show $\exists\lambda$ such that $h(\lambda) = 1 - \alpha$.

If h is continuous, the intermediate value theorem guarantees such a λ.

If h has a discontinuity at the point where it crosses $1 - \alpha$, define:

$$\lambda_\alpha = \sup[\lambda \mid h(\lambda) \geq 1 - \alpha]$$

In this case, $\int_{H(\lambda_\alpha)} p\,(x, y)\,\mathrm{d}x\mathrm{d}y > 0$. If $\mathcal{X}$ is continuous, choose a measurable subset $D' \subseteq D_\alpha$ such that

$$\mathcal{C} = \{(x, y) \mid p(y \mid x) \geq \lambda_\alpha\} \cup D', \qquad \mathbb{P}\,(X \times Y \in \mathcal{C}) = 1 - \alpha.$$

If X is not continuous then we can use a randomised rule on D_α as in the original Neyman-Pearson lemma.

Keep λ as above and denote: $f = p(x, y)$ and $g = p(x)$. Taking $\mathcal{C} = \left\{(x, y) \in \mathcal{X} \times \mathcal{Y} \mid \frac{f}{g} \leq \lambda\right\}$ gives $\int_{\mathcal{C}} f(x, y)\mathrm{d}x\mathrm{d}y = 1 - \alpha$ by definition of λ.

$$\int_{\mathcal{C}} g\,\mathrm{d}x\mathrm{d}y = \int_{x\in\mathcal{X}} p(x) \int_{y\in C(x)} \mathrm{d}y\mathrm{d}x = \int_{x\in\mathcal{X}} p(x)|C(x)|\mathrm{d}x = \mathbb{E}_X[|C(X)|]$$

$$\int_{\mathcal{C}} f\,\mathrm{d}x\mathrm{d}y \;= \mathbb{P}\bigl((X, Y) \in \mathcal{C}\bigr) = \mathbb{P}(Y \in C(X))$$

Hence via the Neyman Pearson lemma 1, $\mathcal{C}$ gives the solution to minimisation problem 1.

By definition, $\hat{S} \in S_1$ if and only if there exists $\lambda' \in \mathbb{R}$ such that $C = \{(x, y) \mid \hat{S}(x, y) \leq \lambda'\}$. The equality holds if and only if, for every (x, y), the inequality $p(y \mid x) \geq \lambda$ is equivalent to $\hat{S}(x, y) \leq \lambda'$. In other words, the ordering induced by $p(y \mid x)$ is the reverse of that induced by $\hat{S}(x, y)$. This is equivalent to the existence of a strictly increasing function ϕ such that

$$\hat{S}(x, y) = \phi\bigl(-p(y \mid x)\bigr)$$

(with $\lambda' = \phi(-\lambda)$), which establishes the result.

C.3 Proof of Theorem 4

For each $x \in \mathcal{X}$, define the conditional high-density region

$$H_x(t) := \{y' \mid p(y' \mid x) \geq t\},$$

and let

$$h_x(t) := \int_{H_x(t)} p(y' \mid x) \mathrm{d}y'.$$

Note that h_x is non-increasing with $h_x(0) = 1$ and $h_x(\infty) = 0$.

We first show that for every $x \in \mathcal{X}$, there exists t_x such that $h_x(t_x) = 1 - \alpha$.

If h_x is continuous, the intermediate value theorem guarantees such a t_x.

If h_x has a discontinuity at the point where it crosses $1 - \alpha$, define

$$t_{x,\alpha} := \sup\{t \mid h_x(t) \geq 1 - \alpha\}, \qquad D_\alpha := \{y \mid p(y \mid x) = t_{x,\alpha}\}.$$

In this case, $\int_{D_\alpha} p\,(y' \mid x)\,\mathrm{d}y' > 0$. If $\mathcal{X}$ is continuous, choose a measurable subset $D' \subseteq D_\alpha$ such that

$$C(x) := H_x(t_{x,\alpha}) \cup D', \qquad \mathbb{P}_{Y|X=x}(C(x)) = 1 - \alpha.$$

If $\mathcal{X}$ is discrete, the same can be achieved by randomisation on D_α.

Thus a valid threshold t_x always exists for each x . By construction, $H_x(t_x)$ is the unique smallest set (by measure) attaining probability $1 - \alpha$ under when conditioned on x. Define the prediction set

$$\mathcal{C} = \{(x, y) \mid y \in C(x)\}, \qquad C(x) := H_x(t_x).$$

This yields the unique solution to the minimisation problem up to measure 0 sets

Now fix $(x, y) \in \mathcal{X} \times \mathcal{Y}$. We have

$$\begin{aligned}
y \in C(x) &\iff p(y \mid x) \geq t_x \\
&\iff H_x\big(p(y \mid x)\big) \subseteq H_x(t_x) \qquad \text{(monotonicity of level sets)} \\
&\iff \int_{H_x(p(y|x))} p(\hat{y} \mid x)\ \mathrm{d}\hat{y} \leq \int_{H_x(t_x)} p(\hat{y} \mid x)\ \mathrm{d}\hat{y} = 1 - \alpha \\
&\iff \phi\Big(\mathbb{P}_{Y|X=x}\big(H_x(p(y \mid x))\big)\Big) \leq \phi(1 - \alpha) =: \lambda, \quad \forall \phi : [0, 1] \to \mathbb{R} \text{ strictly increasing.}
\end{aligned}$$

Hence, the oracle solution is characterised by the conditional high-density regions $H_x(t_x)$ with coverage $1-\alpha$ for all x, as required.