Quantum Variational Rewinding for Time Series Anomaly Detection

Jack S. Baker[1], Haim Horowitz[1], Santosh Kumar Radha[1], Stenio Fernandes[2], Colin Jones[2], Noorain Noorani[2], Vladimir Skavysh[2], Philippe Lamontagne[3], Barry C. Sanders[4]

[1]Agnostiq Inc., 325 Front St W, Toronto, ON M5V 2Y1
[2]Bank of Canada, 234 Wellington Street Ottawa, ON K1A 0G9
[3]National Research Council Canada, 5145 av. Decelles, Montréal, QC, H3T 2B2
[4]Institute for Quantum Science and Technology, University of Calgary, Alberta T2N 1N4, Canada
skav@bankofcanada.ca

Abstract

We explore a new quantum computing approach to time series anomaly detection (TAD). Our approach - *Quantum Variational Rewinding* (QVR) - trains a family of parameterized unitary time-devolution operators to cluster normal time series instances encoded within quantum states. Unseen time series are assigned an anomaly score based upon their distance from the cluster center, which, beyond a given threshold, classifies anomalous behaviour. We apply QVR to identify anomalous trading activity in cryptocurrency market data and blockchain. Finally, we study our algorithm on IBM's `Falcon r5.11H` family of superconducting transmon QPUs, where anomaly score errors resulting from hardware noise are shown to be reducible by as much as 14% on average using advanced error mitigation techniques.

Introduction

The task of identifying *anomalous* behaviour in time series data is known as time series anomaly detection (TAD). Although there exists a taxonomy of problems and variations under the banner of TAD (Blázquez-García et al. 2022), arguably the most practically relevant is novelty detection. Novelty detection is a one-class classification problem (Seliya, Abdollah Zadeh, and Khoshgoftaar 2021) where learning is performed using a set of time series known to exhibit *normal* (i.e, non-anomalous) behaviour (Gupta et al. 2014; Blázquez-García et al. 2022). Subsequently, anomalies are classified by their deviation from the learnt normal model. The practical relevance of novelty detection stems from the relative scarcity of labeled anomalies to normal data in a broad array of applications spanning fraud detection (Ferdousi and Maeda 2006), intrusion detection (Anton et al. 2018), medical diagnosis (Agliari et al. 2020; Homayouni et al. 2021), observational astronomy (Zhang and Zou 2018) and many others. This wide application domain paired with the advent of newly accessible quantum processing units (Gyongyosi and Imre 2019) (QPUs) naturally poses the question: *Can a quantum approach to TAD be devised?*

In the context of central banking, an emerging application of TAD is analyzing cryptocurrency time series data. Of particular interest are Bitcoin—which has already been used as legal tender (Pérez and Ostroff 2021)—and stablecoins (cryptocurrencies partially or fully pegged to safe assets, such as USD). This is of interest to central banks because of their role in oversight over payment service providers[1] and because of how cryptocurrency developments affect the issuance of central bank digital currency (Lane 2021). Despite seeing increased use, cryptocurrencies and stablecoins remain flawed as a method of payment (Lane 2021). To monitor risk and possible losses in stablecoins, central banks are developing frameworks to classify stablecoins, identify risk scenarios for various classes of stablecoins, and quantify possible losses stemming from these risk scenarios (see for instance (García, Lands, and Yanchus 2021)). In this context, it is important for central banks to investigate novel tools for cryptocurrency time series analysis and anomaly detection including quantum algorithmic approaches.

In classical TAD, temporal features are typically encoded with convolutional neural networks (Wen and Keyes 2019; Kim et al. 2019; Hsu and Liu 2020), long-short-term-memory (LSTM) networks (Hochreiter and Schmidhuber 1997; Malhotra et al. 2015; Shipmon et al. 2017; Zhang and Zou 2018; Ji, Gong, and Feng 2021), bi-directional LSTMs (Graves and Schmidhuber 2005; Zhang et al. 2021), and gated recurrent units (Cho et al. 2014; Guo et al. 2018). These architectures are often combined into hybrid models such as recurrent autoencoders (Pereira and Silveira 2018; Kieu et al. 2019) and VAE-GANs (Niu, Yu, and Wu 2020). More recently, transformers (Kim, Kang, and Kang 2023), graph neural networks (Jin et al. 2024), and large language models (Su et al. 2024; Zhang et al. 2024) have been applied, often stacked into deep networks that excel on large datasets (Shone et al. 2018; Chalapathy and Chawla 2019; Choi et al. 2021; Li and Jung 2023).

Here we instead encode temporal features in the natural time evolution of quantum systems within a quantum machine learning (QML) (Biamonte et al. 2017) framework for gate-model QPUs. TAD (with novelty detection) is achieved by training parameterized unitary operators to evolve quantum states prepared with time series data representative of normal behaviour. Expectation values of these evolved states cluster around a center; new time series are classified as

[1]Retail Payment Activities Act (S.C. 2021, c. 23, s. 177), https://laws-lois.justice.gc.ca/eng/acts/r-7.36/FullText.html

normal or anomalous depending on their distance from this cluster. Unlike data-hungry deep learning, the variational time-evolution technique (Cîrstoiu et al. 2020) can achieve low generalization error with relatively few training examples (Caro et al. 2022).

The algorithm is designed for the near-intermediate scale quantum (NISQ) era (Preskill 2018), where few noisy qubits necessitate low-depth circuits. Variational quantum algorithms (VQAs) (Cerezo et al. 2021), which optimize parameterized circuits in a hybrid loop, are a natural fit. Several VQAs for static anomaly detection have been proposed (Liu and Rebentrost 2018; Liang et al. 2019; Kottmann et al. 2021; Sakhnenko et al. 2021; Herr, Obert, and Rosenkranz 2021), but none explicitly encode temporal features. Naïve encoding via Trotterized time-evolution (Wiebe et al. 2010; Kalfon et al. 2024) is often too deep for NISQ hardware. Variational fast forwarding (VFF) (Cîrstoiu et al. 2020) alleviates this by replacing Trotterization with a low-depth variational ansatz with bounded error.

VFF has been applied in combinatorial optimization (Radha 2021), quantum generative modeling of time series (Horowitz, Rao, and Radha 2022), and in studies of generalization and resource-efficient fast forwarding (Gibbs et al. 2022; Caro et al. 2022). Building on these ideas, we introduce the Quantum Variational Rewinding (QVR) algorithm for TAD. QVR is modular and customizable, agnostic to the choice of ansatz, observables, or other components. We apply QVR to (1) synthetic univariate data with injected anomalies and (2) multivariate cryptocurrency trading data capturing large blockchain transfers. The first serves as a didactic example; the second as a real-world case study probing the impact of blockchain flows on trading dynamics. We also demonstrate QVR on IBM's `Falcon r5.11H` superconducting transmon chips (`ibm_lagos`, `ibm_perth`, `ibm_jakarta`). Although QVR avoids deep circuits, it requires extensive measurements; thus, tractable execution is achieved by combining parallel QPU use with near-QPU classical compute. Finally, we explore several error mitigation techniques and show that error mitigation can be effective at substantially reducing hardware errors.

Quantum Variational Rewinding

In what follows, Fig. 1 can be used as a graphical interpretation of QVR. Full pseudocode for QVR can be found in the Supplemental Materials (SM)[2].

In the following, m will denote the size of the training set that we consider, d will be a fixed integer such that each element from the training set is a d-dimensional time series, and p will be a fixed integer such that each time series consists of p-many time steps. Our ultimate goal will be to learn a family of unitary matrices that, together with the cost function (to be defined later in Eq. 6), will give rise to a classification criterion that will allow us to determine whether a given time series is normal or anomalous.

[2]Data and code is available here: https://github.com/AgnostiqHQ/QuantumVariationalRewinding or here: https://doi.org/10.5281/zenodo.7258627. A tutorial to QVR can be found here: https://pennylane.ai/qml/demos/tutorial_univariate_qvr.

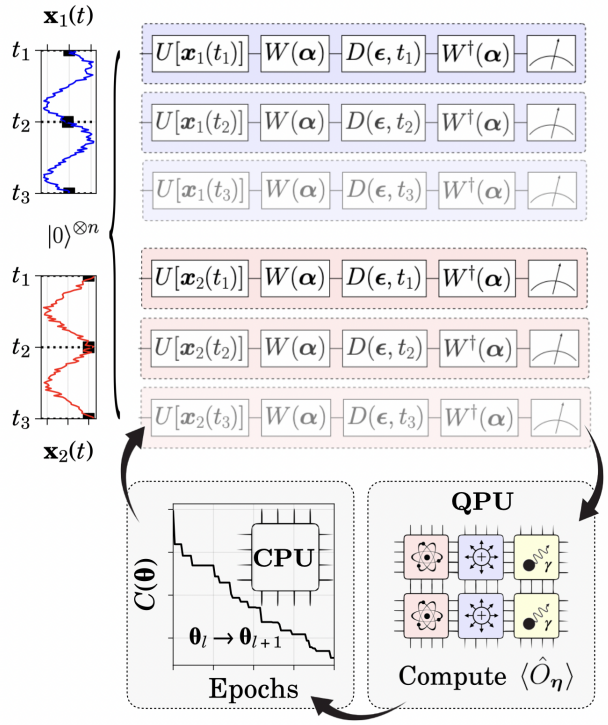

Figure 1: The hybrid quantum-classical loop used for training the QVR model. Starting from the top, $N_X \times N_{\mathcal{T}} \times N_E$ quantum circuits are encoded with time series at selected points in time (shown to the left) followed by the Ansätz described in Eq. 2. Following the downwards arrow, circuits are offloaded to (multiple) QPUs to prepare $|\boldsymbol{x}_i(t_j), \boldsymbol{\alpha}, \boldsymbol{\epsilon}\rangle$ (Eq. 1) to evaluate $\langle \hat{O}_{\boldsymbol{\eta}} \rangle$ (Eq. 3) for each. Moving left, all evaluated $\langle \hat{O}_{\boldsymbol{\eta}} \rangle$ are transferred to a classical computer and used to build the cost function at the l^{th} iteration of the quantum-classical loop: $C(\boldsymbol{\theta}_l)$ (Eq. 6). Using a classical optimization routine, a new candidate $\boldsymbol{\theta}_{l+1}$ is found and used to generate the quantum circuits in the next iteration of the loop.

We begin with a training set $X = \{\boldsymbol{x}_1(t), \boldsymbol{x}_2(t), \ldots, \boldsymbol{x}_m(t)\}$ of d-dimensional real-valued discrete functions of time (time series) $\boldsymbol{x}_i(t) = \{x_i^1(t), x_i^2(t), \ldots, x_i^d(t)\}$ where $t \in \mathcal{T} = \{t_1, t_2, \ldots, t_p\}$. We shall refer to the elements of $\mathcal{T}$ as time points, which can be general real numbers. Using this training set, our goal is to learn an optimal set of parameters (where the exact sense of "optimal" is explained in the rest of this Section) that will give rise to a real-valued function $a_X[\boldsymbol{y}]$ that is able to assign anomaly scores to general d-dimensional time series $\boldsymbol{y}$. By comparing this anomaly score to a set threshold, we determine if $\boldsymbol{y}$ is normal or anomalous.

Explicitly, we train our model with mini-batches of the number of series $B_X \subseteq X$ and of the number of time points in each series $B_{\mathcal{T}} \subseteq \mathcal{T}$. That is, for each iteration of the quantum-classical loop shown in Fig. 1, we draw randomly $N_X = |B_X|$-many time series restricted to randomly chosen

$N_{\mathcal{T}} = |B_T|$-many time points. With these mini-batches, we prepare the states $U[\boldsymbol{x}_i(t_j)]\,|0\rangle^{\otimes n} = |\boldsymbol{x}_i(t_j)\rangle$ for embedding unitary $U[\boldsymbol{x}_i(t_j)]$ acting on the n qubit state of zeroes. This step applies a feature map to the classical data, embedding it as a quantum state (Schuld et al. 2020). We conjecture that our approach is also applicable to quantum data (see (Huang, Kueng, and Preskill 2021), for example), but this is outside the scope of this work and is left as an open problem.

After the embedding, each state now undergoes a dynamical process

$$|\boldsymbol{x}_i(t_j), \boldsymbol{\alpha}, \boldsymbol{\epsilon}\rangle := e^{-i\hat{H}(\boldsymbol{\alpha},\boldsymbol{\epsilon})t_j}\,|\boldsymbol{x}_i(t_j)\rangle \tag{1}$$

as generated by a general parameterized Hamiltonian operator $\hat{H}(\boldsymbol{\alpha}, \boldsymbol{\epsilon})$. A family of dynamical processes can be implemented with controllable circuit depth using the structure first proposed in (Cîrstoiu et al. 2020)

$$e^{-i\hat{H}(\boldsymbol{\alpha},\boldsymbol{\epsilon})t_j} = W^{\dagger}(\boldsymbol{\alpha})D(\boldsymbol{\epsilon}, t_j)W(\boldsymbol{\alpha}) \tag{2}$$

for parameterized unitary $W(\boldsymbol{\alpha})$, diagonal time-encoded unitary $D(\boldsymbol{\epsilon}, t_j)$, variational parameters $\boldsymbol{\alpha}$ and 2^n-length vector $\boldsymbol{\epsilon}$. It can then be seen that Eq. 2 is an eigendecomposition of $e^{-i\hat{H}(\boldsymbol{\alpha},\boldsymbol{\epsilon})t_j}$ where $W(\boldsymbol{\alpha})$ represents the matrix of eigenvectors and $D(\boldsymbol{\epsilon}, t_j)$ its diagonalized form. The unitary operation that corresponds to $D(\boldsymbol{\epsilon}, t_j)$ is implemented using a k-local approximation of $e^{-iM(\boldsymbol{\epsilon})t_j}$, where $M(\boldsymbol{\epsilon})$ is an intermediate non-unitary diagonal matrix. Here, k-locality is the restriction that every separable term of $D(\boldsymbol{\epsilon}, t_j)$ can only be approximated with the mutual interaction of k-many qubits. We use the procedure defined in (Welch et al. 2014) meaning that while $D(\boldsymbol{\epsilon}, t_j)$ is a diagonal matrix with 2^n entries, we can implement a k-local version using a $\mathrm{poly}(n)$ number of gates (Welch et al. 2014) and need not store $\sim 2^n$ eigenvalues $\boldsymbol{\epsilon}$ in memory.

While Eq. 2 would conventionally be interpreted as a forward time-evolution operator, in the context of our algorithm, we conceptualize Eq. 2 as a devolving (i.e backwards in time) $|\boldsymbol{x}_i(t_j)\rangle$ by time t_j. This is permitted since a forward-time evolution as generated by $\hat{H}(\boldsymbol{\alpha}, \boldsymbol{\epsilon})$ is equivalent to a devolution as generated by $-\hat{H}(\boldsymbol{\alpha}, \boldsymbol{\epsilon}) = \hat{H}(\boldsymbol{\alpha}, \boldsymbol{\epsilon})'$ where $\hat{H}(\boldsymbol{\alpha}, \boldsymbol{\epsilon})'$ is another valid Hamiltonian. Also, taking the conjugate transpose of both sides of Eq. 2, we find $e^{i\hat{H}(\boldsymbol{\alpha},\boldsymbol{\epsilon})t_j} = e^{-i\hat{H}(\boldsymbol{\alpha},\boldsymbol{\epsilon})t_j}{}^{\dagger} = W^{\dagger}(\boldsymbol{\alpha})D(\boldsymbol{\epsilon}, -t_j)W(\boldsymbol{\alpha}) = W^{\dagger}(\boldsymbol{\alpha})D(-\boldsymbol{\epsilon}, t_j)W(\boldsymbol{\alpha})$. This is an important insight as we will later learn that $\boldsymbol{\epsilon}$ is sampled from normal distributions with variable centres which are permitted to be positive or negative. This means that both $D(-\boldsymbol{\epsilon}, t_j)$ and $D(\boldsymbol{\epsilon}, t_j)$ can be sampled at different model parameters which are determined in training. Also, unlike Trotterized evolution, any time t can be accessed without first traversing through intermediate time steps. In the context of forward time evolution, the authors of Ref. (Cîrstoiu et al. 2020) interpret this as a *fast forwarding* whereas in the context of devolution we interpret it as a *rewinding*. This is the core principle of QVR.

When preparing Eq. 1, $\boldsymbol{\epsilon}$ is uniformly randomly drawn from independent normal distributions $\epsilon_q \sim \mathcal{N}(\mu_q, \sigma_q)$, $q \in \{1, 2, \ldots Q | Q \leq 2^n - 1\}$ which we abbreviate to $\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\sigma})$. That is, the vector $\boldsymbol{\epsilon}$ from 1 will be randomly generated multiple times in order to compute the expectation value below. For a single random $\boldsymbol{\epsilon}$, we evaluate the expectation value of a general parameterized observable $\hat{O}_{\boldsymbol{\eta}}$

$$\Omega\left(\boldsymbol{x}_i(t_j), \boldsymbol{\alpha}, \boldsymbol{\epsilon}, \boldsymbol{\eta}\right) := \left\langle \boldsymbol{x}_i(t_j), \boldsymbol{\alpha}, \boldsymbol{\epsilon} \middle| \hat{O}_{\boldsymbol{\eta}} \middle| \boldsymbol{x}_i(t_j), \boldsymbol{\alpha}, \boldsymbol{\epsilon} \right\rangle \tag{3}$$

where $\boldsymbol{\eta}$ is a parameter vector $(\eta_0, \eta_1, \ldots, \eta_g)$. $\hat{O}_{\boldsymbol{\eta}}$ is permitted to be any n-qubit Hermitian observable and the precise choice can be considered an inductive bias of the model. In this work we choose $\hat{O}_{\boldsymbol{\eta}} = \eta_0 I - \frac{1}{n}\sum_{i=1}^{n} \eta_i \hat{\sigma}_z^i$ where $\eta_i = 1$ for $i \geq 1$ is fixed in the forthcoming parameter optimization but η_0 is variable within $-1 \leq \eta_0 \leq 1$. Next, we take the *classical* expectation value of the square of Eq. 3, drawing $\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\sigma})$ and re-scale by L

$$C_1(\boldsymbol{x}_i(t_j), \boldsymbol{\theta}) := \frac{\mathop{\mathbb{E}}_{\boldsymbol{\epsilon}\sim\mathcal{N}(\boldsymbol{\sigma},\boldsymbol{\mu})}\left[\Omega^2\left(\boldsymbol{x}_i(t_j), \boldsymbol{\alpha}, \boldsymbol{\epsilon}, \boldsymbol{\eta}\right)\right]}{L} \tag{4}$$

where $\boldsymbol{\theta} = [\boldsymbol{\alpha}, \boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\eta}]$ and L is a factor chosen heuristically (in our specific implementation, $L = 4$) in order to shrink the range of possible values to (0, 1). It should be mentioned that L is not an essential part of the algorithm, and was only used to make the presentation more aesthetic. In practice, we estimate the right-hand side of Eq. 4 by taking the mean value of N_E-many terms in the right-hand side brackets, where N_E should be chosen either heuristically or once it becomes apparent that the differences in the mean value upon increasing N_E are sufficiently small. With our explicit choice of $\hat{O}_{\eta}$, Eq. 4 measures the average square distance between η_0 and $\left\langle \boldsymbol{x}_i(t_j), \boldsymbol{\alpha}, \boldsymbol{\epsilon} \middle| \sum_{i=1}^{n} \hat{\sigma}_z^i \middle| \boldsymbol{x}_i(t_j), \boldsymbol{\alpha}, \boldsymbol{\epsilon} \right\rangle$ for N_E-many sampled $\boldsymbol{\epsilon}$. Eq. 4 is the fundamental building block of the forthcoming loss function and can be considered the cost contribution from the i^{th} time series at the j^{th} time point. Taking the average $\forall t \in B_{\mathcal{T}}$, we define the cost for the entirety of the i^{th} time series as

$$C_2(\boldsymbol{x}_i, \boldsymbol{\theta}) := \frac{1}{N_{\mathcal{T}}} \sum_{t_j \in B_{\mathcal{T}}} C_1(\boldsymbol{x}_i(t_j), \boldsymbol{\theta}) \ . \tag{5}$$

Our task is to find model parameters that cluster all instances of the expression in Eq. 5 about a given point, which, with our choice of $\hat{O}_{\boldsymbol{\eta}}$, is a clustering about a center at η_0. To achieve this, we minimize the cost function

$$C(\boldsymbol{\theta}) := P_{\boldsymbol{\tau}}(\boldsymbol{\sigma}) + \frac{1}{2N_X} \sum_{\boldsymbol{x}_i \in B_X} C_2(\boldsymbol{x}_i(t), \boldsymbol{\theta}) \tag{6}$$

such that

$$\boldsymbol{\theta}^{\star} := \mathrm{argmin}_{\boldsymbol{\theta}}[C(\boldsymbol{\theta})] \tag{7}$$

where the optimal parameters $\boldsymbol{\theta}^{\star}$ are found with a minimization routine on a classical computer where each new step of the minimizer estimates $C(\boldsymbol{\theta})$ with new mini-batches B_X and $B_{\mathcal{T}}$, terminating at a fixed number of epochs (equivalently, mini-batch iterations) or when $C(\boldsymbol{\theta})$ falls below a certain tolerance. $P_{\boldsymbol{\tau}}(\boldsymbol{\sigma})$ is an explicit regularization function with γ-many hyperparameters $\boldsymbol{\tau} = \{\tau_1, \tau_2, \ldots, \tau_\gamma\}$

designed to penalize large entries of $\boldsymbol{\sigma}$. While the positive domain of any sigmoidal function can be used, throughout this work, we take

$$P_{\boldsymbol{\tau}}(\boldsymbol{\sigma}) := \frac{1}{\pi Q} \sum_{m=1}^{Q} \arctan(2\pi\tau_m|\sigma_m|) \tag{8}$$

where $\tau_1 = ... = \tau_{Q-1} = \tau$ for a single contraction hyperparameter τ. With this choice, $C(\boldsymbol{\theta})$ assumes values in the interval $(0, 1)$. The motivation for including a $P_{\boldsymbol{\tau}}(\boldsymbol{\sigma})$ regularization is that for a fixed $\boldsymbol{\alpha}$, letting b be the length of $\boldsymbol{\epsilon}$, the mapping $f_{\boldsymbol{\alpha}} : R^b \to U(b)$ defined by $\boldsymbol{\epsilon} \mapsto W(\boldsymbol{\alpha})^\dagger e^{-iM(\boldsymbol{\epsilon})} W(\boldsymbol{\alpha})$ gives rise to a set $\mathrm{Im}(f_{\boldsymbol{\alpha}})$ of unitary matrices which we wish to controllably restrict. The distribution $\mathcal{N}(\boldsymbol{\sigma}, \boldsymbol{\mu})$ induces a probability distribution on $\mathrm{Im}(f_{\boldsymbol{\alpha}})$, which, as is indicated in the SM, by minimizing $\boldsymbol{\sigma}$ we contribute to increasing $C_2(\boldsymbol{y}, \boldsymbol{\theta}^\star)$ for time series outside of the training set.

After a sufficiently good approximation of $\boldsymbol{\theta}^\star$ is obtained, it is possible to evaluate the anomaly score of an unseen time series $\boldsymbol{y}$ by letting

$$a_X[\boldsymbol{y}] := |2C(\boldsymbol{\theta}^\star) - 2P_{\boldsymbol{\tau}}(\boldsymbol{\sigma}) - C_2[\boldsymbol{y}, \boldsymbol{\theta}^\star]|. \tag{9}$$

Time series are classified as anomalous whenever the threshold $a_X[\boldsymbol{y}] > \zeta$ is exceeded. Otherwise, time series are classified as normal. If the data are available, in practice, ζ is chosen to maximize a performance metric (such as F_1 score or balanced accuracy score) using a validation set of labelled anomalous time series.

Demonstration

Synthetic univariate time series

We illustrate our classification algorithm with a simple synthetic example. The training set X contains 50 univariate time series of length 50, where each value is drawn from a zero-mean Gaussian. States $|x_i(t), \boldsymbol{\alpha}, \boldsymbol{\epsilon}\rangle$ are prepared via angle embedding on a 2-qubit circuit, with $U[\boldsymbol{x}_i(t_j)] = R_y[x_i^1(t_j)] \otimes I$, and $W(\boldsymbol{\alpha})$ given by the ansatz of (Schuld et al. 2020). Quantum expectations are computed exactly using the `lightning.qubit` simulator in Pennylane (Bergholm et al. 2018). We take $N_E = 10$, $N_{\mathcal{T}} = 10$, $N_X = 5$ and select $\boldsymbol{\theta}^\star$ from the run yielding the lowest $C(\boldsymbol{\theta})$ over 300 optimization trials, each with 1000 mini-batch iterations.

We test on three datasets of equal size and length: $\mathcal{F}$ (same process as X), $\mathcal{G}$ (same process with inserted anomalous spikes), and $\mathcal{H}$ ($\sin(t)$ added to $f_i(t)$). Figure 2(a) shows the time-resolved anomaly score

$$a_X^t(\boldsymbol{y}) := |2C(\boldsymbol{\theta}^*) - 2P_{\boldsymbol{\tau}(\boldsymbol{\sigma})} - C_1(\boldsymbol{y}(t), \boldsymbol{\theta}^*)|.$$

Series from $\mathcal{F}$ cluster near $a_X^t(y) \approx 0$ across t, yielding consistently low $a_X[f_i]$ (red markers in Fig. 2(d)). By contrast, $\mathcal{G}$ and $\mathcal{H}$ exhibit high-amplitude regions with elevated anomaly scores (Figs. 2(b–c)), producing spreads over three orders of magnitude (grey markers) and consistently above $\mathcal{F}$. In practice, a threshold ζ would be chosen from validation data. In this toy case, ζ values exist that perfectly separate $\mathcal{F}$ from $\mathcal{G}$ and $\mathcal{H}$, though such separability is unlikely in real datasets.

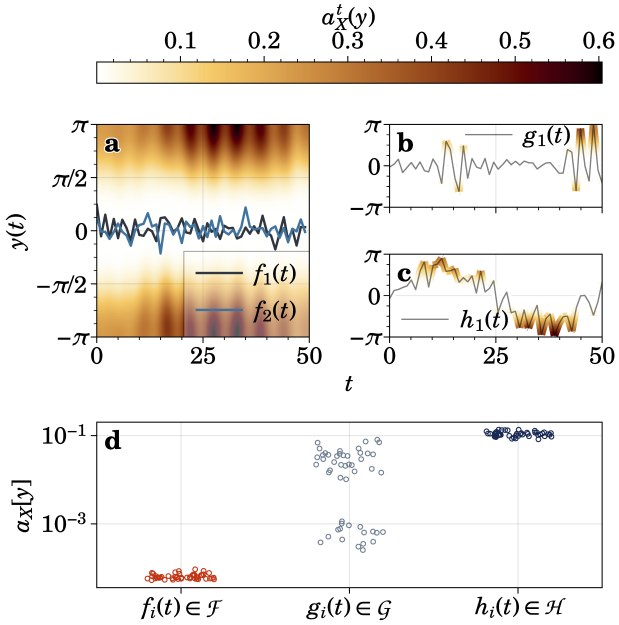

Figure 2: QVR applied to synthetically generated univariate training and testing data. (a) A heat-map of the time-resolved anomaly score $a_X^t(y)$. Two testing signals $f_1(t)$, $f_2(t) \in \mathcal{F}$ (generated using the same mechanism as the training data) are over-plotted and are seen in the region of low $a_X^t(y)$. (b-c) Example signals from the test sets $\mathcal{G}$ and $\mathcal{H}$ respectively, colored according to $a_X^t(y)$. $\mathcal{G}$ contains noisy signals with randomly inserted *anomaly spikes* and $\mathcal{H}$ contains noisy $\sin(t)$ signals. (d) The anomaly scores $a_X[y]$ for each time series in the sets $\mathcal{F}$, $\mathcal{G}$ and $\mathcal{H}$. The y-axis has a logarithmic scale.

Multivariate cryptocurrency time series

Now we tackle a more realistic problem of studying anomalous trading behavior on the cryptocurrency exchange Binance[3]. Unlike in traditional financial markets, in cryptocurrency markets there is additional information available to participants via the blockchain. A transfer of money from one wallet to another (such as a deposit into Binance) is visible to everyone on the blockchain, informing the market that significant trading activity may be imminent on the exchange. This can result in a period of trading that deviates from the norm. Thus, information from the blockchain can be used to label time periods as potentially anomalous and then use this data to train a machine learning algorithm.

In our case, we study the Bitcoin (BTC) vs. Tether (USDT) time series between 2019-06-06 and 2021-11-24. We extract 3-hour time periods around large transfers of BTC or USDT between (either to or from) Binance and external wallets, thus generating potentially "anomalous" data. To monitor large (a.k.a. "whale") transactions, we use data from `Whale Alert`[4], which has previously been

[3] https://www.binance.com/

[4] https://whale-alert.io/

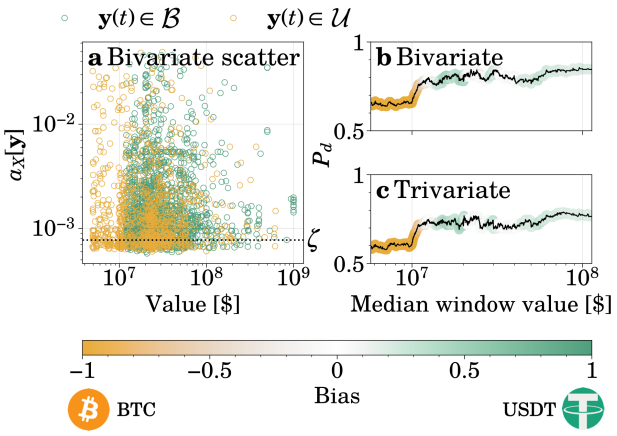

Figure 3: Quantifying anomalous cryptocurrency time series behaviour on the large $\mathcal{U}$ and $\mathcal{B}$ data sets using QVR. (a) A scatter plot of anomaly scores from the bivariate model versus the USD value of BTC and USDT transactions reported by `Whale Alert`. The threshold ζ is marked. (b) and (c) The detection probability $P_d = P(a_X[\boldsymbol{y}] > \zeta)$ within a moving window of size $W = 285$ and step $s = 1$ for the bivariate and trivariate models, respectively. Lines are colored according to the lower color bar and represent the proportion of transactions in the window that were on the BTC or USDT blockchains.

used to study trader behaviour impact on cryptocurrency markets (Saggu 2022). We also find time periods without large transfers in their vicinity, thus generating "normal" data. Altogether, we obtain 5432 time series with at least one large transaction and 842 with no large transactions in their vicinity. We subdivide the datasets into six subsets: $\mathcal{U}_\pm$, $\mathcal{B}_\pm$, $\tilde{\mathcal{U}}_+$, $\tilde{\mathcal{U}}_-$, $\tilde{\mathcal{B}}_+$, $\tilde{\mathcal{B}}_-$. Here, the subscripts $+$ or $-$ indicate that transactions were into or out of Binance exchange wallets, respectively, while the subscript $\pm$ means transactions in both directions are permitted. Additionally, the tilde cap $\sim$ designates data sets where more than one large transaction of BTC or USDT were made during the time window under consideration. Those without the tilde cap have only a single large transaction. Next, training data set X and normal data set $\mathcal{N}$, $X \cap \mathcal{N} = \varnothing$, contain time-series where no alerts were made in the time window. Lastly, $\mathcal{V}$ is a validation data set containing equal numbers of series arising from the same conditions as sets $\tilde{\mathcal{U}}_+$, $\tilde{\mathcal{U}}_-$, $\tilde{\mathcal{B}}_+$, $\tilde{\mathcal{B}}_-$, but with no common elements. For each of these data sets, we consider up to three time series defined over the window: the mean deviation of the BTC-USDT price, the cumulative trading volume and the cumulative taker buy base asset (TBBA) volume. Models trained using only the first two series are called bivariate models and those trained using all three series are called trivariate models. Further information on the cryptocurrency data sets including their construction and pre-processing can be found in the SM.

Similarly to the synthetic case, classical data is encoded with angle embedding $U[\boldsymbol{x}_i(t_j)] = R_y[x_i^1(t_j)] \otimes R_y[x_i^2(t_j)] \otimes \cdots \otimes R_y[x_i^d(t_j)]$, with $n = d$ qubits for the bi- and trivariate models. We use the same three-layer ansatz as before (Schuld et al. 2020), and compute expectation values exactly with the `lightning.qubit` simulator in `Pennylane` (Bergholm et al. 2018). With $N_X = N_E = N_{\mathcal{T}} = 10$, both models are trained 50 times for 2000 mini-batch iterations, then evaluated on $\mathcal{V}$ and $\mathcal{N}$ to tune ζ for maximum balanced accuracy A_B. Optimal models (highest A_B across 50 runs) achieve $A_B = 0.82$ (bivariate) and 0.77 (trivariate). These models are then applied to all test sets, and A_B and F_1 scores computed with $\mathcal{N}, \mathcal{U}$, and $\mathcal{B}$.

Results appear in Fig. 3 and the solid bars of Fig. 4(a–b). In Fig. 3(a), many $\mathcal{U}$ and $\mathcal{B}$ series are flagged as anomalous, though anomaly score is not directly tied to transaction size. Part of the challenge is that there are frequently multiple large transactions within a given 3-hour time window. As such, we also consider how the median window value affects the detection probability in a fixed window, $P_d = P(a_X[\boldsymbol{y}] > \zeta)$. Here, we do see a relationship, with P_d increasing overall with median transaction value [Figs. 3(b–c)]. This pattern is shaped by asset mix: small-value windows are BTC-dominated (orange regions near the origin), while larger ones skew towards USDT or a BTC/USDT balance (green regions). Thus, P_d and transaction value (in USD) are positively correlated, and large USDT inflows into Binance lead to trading that is the most likely to be classified as anomalous.

From Fig. 4(a–b), $\mathcal{B}_\pm$ and $\mathcal{U}_\pm$ are classified only slightly better than chance, likely due to lower transaction volumes. By contrast, $\tilde{\mathcal{U}}_+$ achieves $A_B \approx 0.8$ in both models, with transactions reaching $\sim$ \$1B. This suggests large USDT deposits into exchange wallets induce strongly non-normal market behaviour. Comparing models, the trivariate variant improves A_B and F_1 on $\tilde{\mathcal{U}}_-$ and $\mathcal{B}_\pm$ but slightly degrades performance on $\tilde{\mathcal{U}}_+$ and $\tilde{\mathcal{B}}_+$. Overall, QVR indicates that BTC/USDT flows to and from exchange wallets often trigger anomalous market dynamics, most strongly after large USDT transfers from external to Binance wallets.

Experiments on superconducting transmon quantum processing units

We next examine QVR on real hardware, using IBM's `Falcon r5.11H` family of superconducting transmon QPUs[5] with re-optimized ζ for validation and testing on bivariate and trivariate models. Given present QPU speeds ($\sim$ 1000 CLOPS (Wack et al. 2021)), cloud-access latency, provider caps, and the large number of expectation values required ($\approx 65,000$), we adopt measures to make experiments tractable. First, we split datasets across `ibm_lagos` and `ibm_perth` for a 2x speedup (Fig. 4(f)), noting idle qubits could in principle be exploited with distributed methods such as QMPI (Häner et al. 2021). Second, we deploy QVR testing via `Qiskit Runtime`, which reduces latency and shifts resource limits from shot count to runtime. We also reduce shots per expectation value to 256 (bivariate) and 512 (trivariate).

[5] https://quantum-computing.ibm.com/

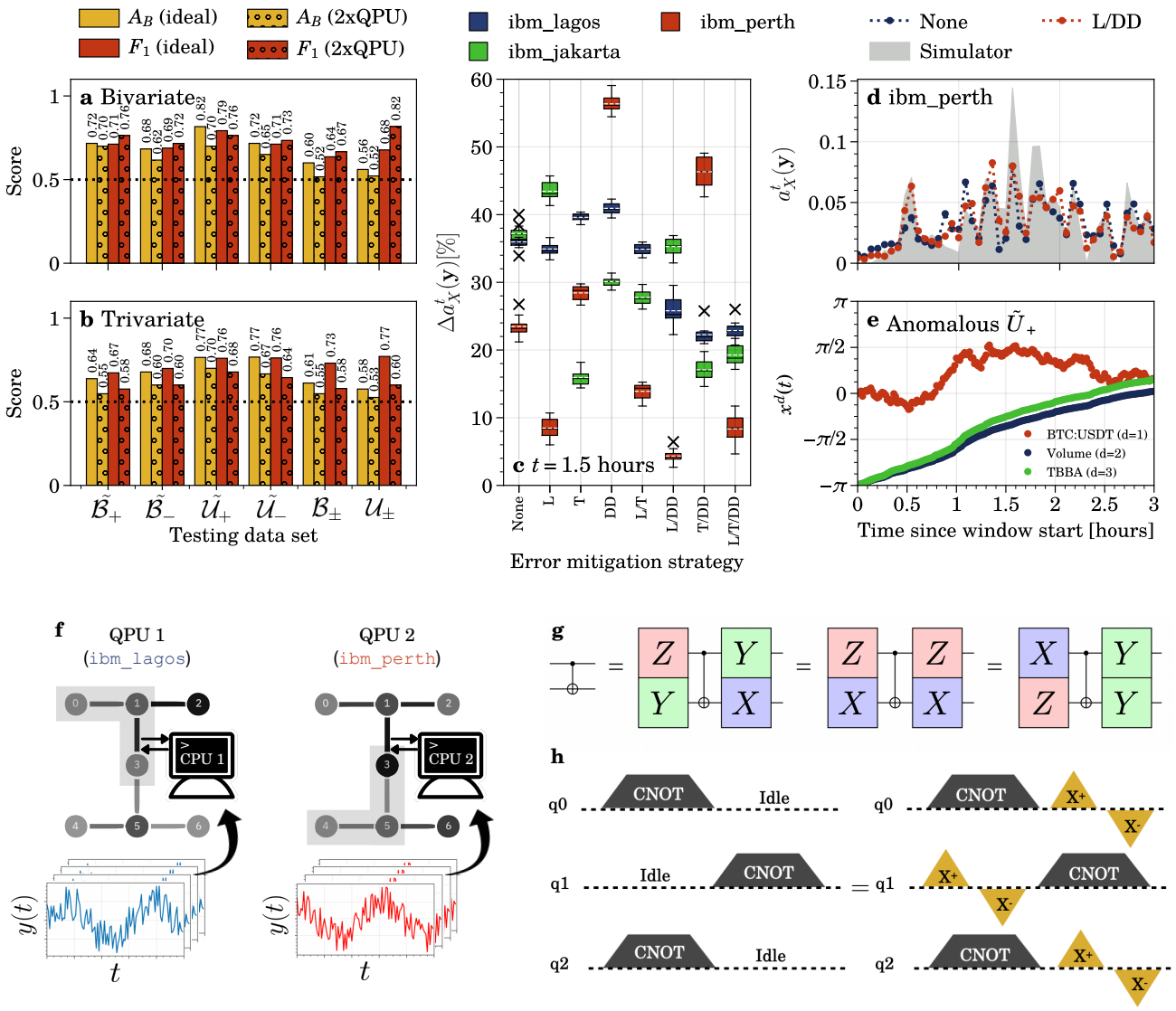

Figure 4: Experiments on 7-qubit superconducting transmon QPUs. (a) and (b) F_1 and balanced accuracy scores (A_B) for the bivariate and trivariate series, respectively. Scores are given for an ideal quantum simulator and for a dual QPU strategy as shown in (f). (c) The anomaly score error $\Delta a_X^t(y)$ compared with an ideal QPU for various error mitigation strategies: L = noise-aware layout, T = Pauli twirling, DD = dynamical decoupling. Forward slashes indicate their use in combination. (d) The time-resolved anomaly score $a_X^t(\boldsymbol{y})$ for a selected $\boldsymbol{y}(t) \in \tilde{\mathcal{U}}_+$ on `ibm_perth`, with and without L/DD error mitigation. (e) The selected (trivariate) time series studied in (d), in which a large USDT is made into Binance. (f) The dual QPU strategy used in (a) and (b). Half of the time series are sent to one QPU and half to another, sped up by low-latency CPU interaction. Grey highlighted areas over nodes of the QPU graphs are exemplar noise-aware layouts of three qubits. (g) Exemplar gate identities used in Pauli twirling. (h) Illustration of an `X2pm` DD pulse sequence.

Results (circle-filled bars in Fig. 4(a–b)) show bivariate performance remains close to ideal (noise-free) case despite noise. Interestingly, F_1 scores for $\mathcal{U}_\pm$ and $\mathcal{B}_\pm$ even improve, though this is an artefact of ζ re-optimization yielding high recall but low precision. Noise compresses the spread of $a_X[\boldsymbol{y}]$, as expected from decoherence driving $\langle\hat{\sigma}_z^i\rangle \to 0.5$, producing noise-induced barren plateaus (Wang et al. 2021). As a result, time series near ζ under ideal conditions are increasingly misclassified under noise and shot fluctuations.

To address trivariate errors, we test error mitigation strategies: noise-aware layouts (Li, Zhou, and Feng 2021), Pauli twirling (Cai and Benjamin 2019; Cai, Xu, and Benjamin 2020), and dynamical decoupling (DD) (Viola, Knill, and Lloyd 1999; Ezzell et al. 2022), individually and in combination. We evaluate $\Delta a_X^t(\boldsymbol{y})$ at $t = 1.5$ hours for $\boldsymbol{y} \in \tilde{\mathcal{U}}_+$ across ten repeats on `ibm_lagos`, `ibm_perth`, and `ibm_jakarta`, increasing shots to 4096 to suppress stochastic error. Layouts are selected using `mapomatic` (`v0.7`)[6], Pauli twirling averages expectation values across 10 randomly twirled circuits [Fig. 4(g)], and DD applies an `X2pm` sequence [Fig. 4(h)].

As shown in Fig. 4(c), combined L/T/DD reduces anomaly score errors across all QPUs, though this approach is not uniformly the best on each chip. Some strategies may increase error, which suggests that care is needed when applying error mitigation techniques. For `ibm_perth` [red box-plots], L/DD removes error almost entirely, reducing $\Delta a_X^t(y)$ to $\approx 4\%$. Encouraged by this low error, we applied L/DD to other time points of a selected time series [Fig. 4(e)]. As shown in Fig. 4(d), the effectiveness of L/DD varies across time, but the overall error is decreased by approximately 14%. These results provide early evidence that QML performance on NISQ hardware can be improved via careful error mitigation strategy selection.

Conclusions

We have proposed and demonstrated the execution of a low circuit depth QML algorithm for TAD: *Quantum Variational Rewinding* or QVR. QVR learns a distribution of parameterized unitary time-devolution operators able to *rewind* time series representing the normal behaviour of a system/process. Anomalous series are those which cannot be rewound by the learnt normal model. After providing a simple and didactic use case on synthetic univariate time series, we demonstrated that QVR can be used to detect anomalies in realistic multivariate time-series. That is, in an investigation into trader behaviour on cryptocurrency markets in the presence of `Whale Alerts`, QVR identified that market behaviour was detectably different when large quantities of BTC or USDT flowed between exchange wallets and external wallets, as identified by a `Whale Alert`. Anomalous behaviour was identified most frequently when (i) transactions were the largest and (ii) USDT were deposited into the exchange wallets. Lastly, using a trifecta of parallel QPU execution, near-QPU classical resources and advanced error mitigation techniques, we showed the that QVR is resistant to noise when executed on present NISQ hardware. These techniques, in tandem, will allow for ever-more useful QML workflows to become realizable using NISQ QPUs. While we have suggested possible areas where QVR could provide some advantages over classical techniques (Caro et al. 2022; Aharonov, Cotler, and Qi 2022; Huang, Kueng, and Preskill 2021), adopting the stance taken in Ref. (Schuld and Killoran 2022), we suggest that some case-by-case heuristic advantages over classical methods *could* appear as the scale and quality of QPUs improves. Given this, at this stage in the maturity of NISQ hardware, focusing solely on the topic of advantage may not necessarily be the best approach (Schuld and Killoran 2022). Demonstrations in this work can then be considered proof-of-concept examples in fundamental quantum machine learning research.

[6]https://github.com/Qiskit-Partners/mapomatic

Acknowledgements

We acknowledge the use of IBM Quantum services for this work. We acknowledge the use of the `Qiskit-research` repository in this work: https://github.com/qiskit-research/qiskit-research. The views expressed are those of the authors, and do not reflect the official policy or position of IBM, the IBM Quantum team, or Bank of Canada.

References

Agliari, E.; Barra, A.; Barra, O. A.; Fachechi, A.; Vento, L. F.; and Moretti, L. 2020. Detecting cardiac pathologies via machine learning on heart-rate variability time series and related markers. *Scientific Reports*, 10(1).

Aharonov, D.; Cotler, J.; and Qi, X.-L. 2022. Quantum algorithmic measurement. *Nature Communications*, 13(1).

Anton, S. D.; Ahrens, L.; Fraunholz, D.; and Schotten, H. D. 2018. Time is of the Essence: Machine Learning-Based Intrusion Detection in Industrial Time Series Data. In *2018 IEEE International Conference on Data Mining Workshops (ICDMW)*. IEEE.

Bergholm, V.; Izaac, J.; Schuld, M.; Gogolin, C.; Ahmed, S.; Ajith, V.; Alam, M. S.; Alonso-Linaje, G.; AkashNarayanan, B.; Asadi, A.; Arrazola, J. M.; Azad, U.; Banning, S.; Blank, C.; Bromley, T. R.; Cordier, B. A.; Ceroni, J.; Delgado, A.; Di Matteo, O.; Dusko, A.; Garg, T.; Guala, D.; Hayes, A.; Hill, R.; Ijaz, A.; Isacsson, T.; Ittah, D.; Jahangiri, S.; Jain, P.; Jiang, E.; Khandelwal, A.; Kottmann, K.; Lang, R. A.; Lee, C.; Loke, T.; Lowe, A.; McKiernan, K.; Meyer, J. J.; Montañez-Barrera, J. A.; Moyard, R.; Niu, Z.; O'Riordan, L. J.; Oud, S.; Panigrahi, A.; Park, C.-Y.; Polatajko, D.; Quesada, N.; Roberts, C.; Sá, N.; Schoch, I.; Shi, B.; Shu, S.; Sim, S.; Singh, A.; Strandberg, I.; Soni, J.; Száva, A.; Thabet, S.; Vargas-Hernández, R. A.; Vincent, T.; Vitucci, N.; Weber, M.; Wierichs, D.; Wiersema, R.; Willmann, M.; Wong, V.; Zhang, S.; and Killoran, N. 2018. PennyLane: Automatic differentiation of hybrid quantum-classical computations.

Biamonte, J.; Wittek, P.; Pancotti, N.; Rebentrost, P.; Wiebe, N.; and Lloyd, S. 2017. Quantum machine learning. *Nature*, 549(7671): 195–202.

Blázquez-García, A.; Conde, A.; Mori, U.; and Lozano, J. A. 2022. A Review on Outlier/Anomaly Detection in Time Series Data. *ACM Computing Surveys*, 54(3): 1–33.

Cai, Z.; and Benjamin, S. C. 2019. Constructing Smaller Pauli Twirling Sets for Arbitrary Error Channels. *Scientific Reports*, 9(1).

Cai, Z.; Xu, X.; and Benjamin, S. C. 2020. Mitigating coherent noise using Pauli conjugation. *npj Quantum Information*, 6(1).

Caro, M. C.; Huang, H.-Y.; Ezzell, N.; Gibbs, J.; Sornborger, A. T.; Cincio, L.; Coles, P. J.; and Holmes, Z. 2022. Out-of-distribution generalization for learning quantum dynamics.

Cerezo, M.; Arrasmith, A.; Babbush, R.; Benjamin, S. C.; Endo, S.; Fujii, K.; McClean, J. R.; Mitarai, K.; Yuan, X.; Cincio, L.; and Coles, P. J. 2021. Variational quantum algorithms. *Nature Reviews Physics*, 3(9): 625–644.

Chalapathy, R.; and Chawla, S. 2019. Deep Learning for Anomaly Detection: A Survey.

Cho, K.; van Merrienboer, B.; Bahdanau, D.; and Bengio, Y. 2014. On the Properties of Neural Machine Translation: Encoder-Decoder Approaches.

Choi, K.; Yi, J.; Park, C.; and Yoon, S. 2021. Deep learning for anomaly detection in time-series data: Review, analysis, and guidelines. *IEEE access*, 9: 120043–120065.

Cîrstoiu, C.; Holmes, Z.; Iosue, J.; Cincio, L.; Coles, P. J.; and Sornborger, A. 2020. Variational fast forwarding for quantum simulation beyond the coherence time. *npj Quantum Information*, 6(1).

Ezzell, N.; Pokharel, B.; Tewala, L.; Quiroz, G.; and Lidar, D. A. 2022. Dynamical decoupling for superconducting qubits: a performance survey.

Ferdousi, Z.; and Maeda, A. 2006. Unsupervised Fraud Detection in Time Series data. In *22nd International Conference on Data Engineering Workshops (ICDEW'06)*. IEEE.

García, A.; Lands, B.; and Yanchus, D. 2021. Stablecoin assessment framework. Technical report, Bank of Canada Staff Discussion Paper.

Gibbs, J.; Holmes, Z.; Caro, M. C.; Ezzell, N.; Huang, H.-Y.; Cincio, L.; Sornborger, A. T.; and Coles, P. J. 2022. Dynamical simulation via quantum machine learning with provable generalization.

Graves, A.; and Schmidhuber, J. 2005. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, 18(5): 602–610. IJCNN 2005.

Guo, Y.; Liao, W.; Wang, Q.; Yu, L.; Ji, T.; and Li, P. 2018. Multidimensional time series anomaly detection: A gru-based gaussian mixture variational autoencoder approach. In *Asian Conference on Machine Learning*, 97–112. PMLR.

Gupta, M.; Gao, J.; Aggarwal, C. C.; and Han, J. 2014. Outlier Detection for Temporal Data: A Survey. *IEEE Transactions on Knowledge and Data Engineering*, 26(9): 2250–2267.

Gyongyosi, L.; and Imre, S. 2019. A Survey on quantum computing technology. *Computer Science Review*, 31: 58.

Herr, D.; Obert, B.; and Rosenkranz, M. 2021. Anomaly detection with variational quantum generative adversarial networks. *Quantum Science and Technology*, 6(4): 045004.

Hochreiter, S.; and Schmidhuber, J. 1997. Long short-term memory. *Neural computation*, 9(8): 1735–1780.

Homayouni, H.; Ray, I.; Ghosh, S.; Gondalia, S.; and Kahn, M. G. 2021. Anomaly Detection in COVID-19 Time-Series Data. *SN Computer Science*, 2(4).

Horowitz, H.; Rao, P.; and Radha, S. K. 2022. A quantum generative model for multi-dimensional time series using Hamiltonian learning.

Hsu, C.-Y.; and Liu, W.-C. 2020. Multiple time-series convolutional neural network for fault detection and diagnosis and empirical study in semiconductor manufacturing. *Journal of Intelligent Manufacturing*, 32(3): 823–836.

Huang, H.-Y.; Kueng, R.; and Preskill, J. 2021. Information-Theoretic Bounds on Quantum Advantage in Machine Learning. *Phys. Rev. Lett.*, 126: 190505.

Häner, T.; Steiger, D. S.; Hoefler, T.; and Troyer, M. 2021. Distributed quantum computing with QMPI. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*. ACM.

Ji, Z.; Gong, J.; and Feng, J. 2021. A Novel Deep Learning Approach for Anomaly Detection of Time Series Data. *Scientific Programming*, 2021: 1–11.

Jin, M.; Koh, H. Y.; Wen, Q.; Zambon, D.; Alippi, C.; Webb, G. I.; King, I.; and Pan, S. 2024. A survey on graph neural networks for time series: Forecasting, classification, imputation, and anomaly detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Kalfon, B.; Cherkaoui, S.; Laprade, J.-F.; Ahmad, O.; and Wang, S. 2024. Successive data injection in conditional quantum GAN applied to time series anomaly detection. *IET Quantum Communication*.

Kieu, T.; Yang, B.; Guo, C.; and Jensen, C. S. 2019. Outlier Detection for Time Series with Recurrent Autoencoder Ensembles. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*. International Joint Conferences on Artificial Intelligence Organization.

Kim, E.; Cho, S.; Lee, B.; and Cho, M. 2019. Fault Detection and Diagnosis Using Self-Attentive Convolutional Neural Networks for Variable-Length Sensor Data in Semiconductor Manufacturing. *IEEE Transactions on Semiconductor Manufacturing*, 32(3): 302–309.

Kim, J.; Kang, H.; and Kang, P. 2023. Time-series anomaly detection with stacked Transformer representations and 1D convolutional network. *Engineering Applications of Artificial Intelligence*, 120: 105964.

Kottmann, K.; Metz, F.; Fraxanet, J.; and Baldelli, N. 2021. Variational quantum anomaly detection: Unsupervised mapping of phase diagrams on a physical quantum computer. *Phys. Rev. Research*, 3: 043184.

Lane, T. 2021. Payments innovation beyond the pandemic. Speech by Deputy Governor Timothy Lane at the Institute for Data Valorization in Montreal, Quebec, Canada.

Li, G.; and Jung, J. J. 2023. Deep learning for anomaly detection in multivariate time series: Approaches, applications, and challenges. *Information Fusion*, 91: 93–102.

Li, S.; Zhou, X.; and Feng, Y. 2021. Qubit Mapping Based on Subgraph Isomorphism and Filtered Depth-Limited Search. *IEEE Transactions on Computers*, 70(11): 1777–1788.

Liang, J.-M.; Shen, S.-Q.; Li, M.; and Li, L. 2019. Quantum anomaly detection with density estimation and multivariate Gaussian distribution. *Phys. Rev. A*, 99: 052310.

Liu, N.; and Rebentrost, P. 2018. Quantum machine learning for quantum anomaly detection. *Phys. Rev. A*, 97: 042315.

Malhotra, P.; Vig, L.; Shroff, G.; Agarwal, P.; et al. 2015. Long short term memory networks for anomaly detection in time series. In *Proceedings*, volume 89, 94.

Niu, Z.; Yu, K.; and Wu, X. 2020. LSTM-Based VAE-GAN for Time-Series Anomaly Detection. *Sensors*, 20(13).

Pereira, J.; and Silveira, M. 2018. Unsupervised Anomaly Detection in Energy Time Series Data Using Variational Recurrent Autoencoders with Attention. In *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE.

Preskill, J. 2018. Quantum Computing in the NISQ era and beyond. *Quantum*, 2: 79.

Pérez, S.; and Ostroff, C. 2021. El Salvador Becomes First Country to Adopt Bitcoin as National Currency. *The Wall Street Journal*.

Radha, S. K. 2021. Quantum constraint learning for quantum approximate optimization algorithm.

Saggu, A. 2022. The Intraday Bitcoin Response to Tether Minting and Burning Events: Asymmetry, Investor Sentiment, and "Whale Alerts" on Twitter. *Finance Research Letters*, 103096.

Sakhnenko, A.; O'Meara, C.; Ghosh, K. J. B.; Mendl, C. B.; Cortiana, G.; and Bernabé-Moreno, J. 2021. Hybrid Classical-Quantum Autoencoder for Anomaly Detection.

Schuld, M.; Bocharov, A.; Svore, K. M.; and Wiebe, N. 2020. Circuit-centric quantum classifiers. *Phys. Rev. A*, 101: 032308.

Schuld, M.; and Killoran, N. 2022. Is Quantum Advantage the Right Goal for Quantum Machine Learning? *PRX Quantum*, 3(3).

Seliya, N.; Abdollah Zadeh, A.; and Khoshgoftaar, T. M. 2021. A literature review on one-class classification and its potential applications in big data. *Journal of Big Data*, 8(1): 122.

Shipmon, D. T.; Gurevitch, J. M.; Piselli, P. M.; and Edwards, S. T. 2017. Time Series Anomaly Detection; Detection of anomalous drops with limited features and sparse examples in noisy highly periodic data.

Shone, N.; Ngoc, T. N.; Phai, V. D.; and Shi, Q. 2018. A Deep Learning Approach to Network Intrusion Detection. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2(1): 41–50.

Su, J.; Jiang, C.; Jin, X.; Qiao, Y.; Xiao, T.; Ma, H.; Wei, R.; Jing, Z.; Xu, J.; and Lin, J. 2024. Large language models for forecasting and anomaly detection: A systematic literature review. *arXiv preprint arXiv:2402.10350*.

Viola, L.; Knill, E.; and Lloyd, S. 1999. Dynamical Decoupling of Open Quantum Systems. *Physical Review Letters*, 82(12): 2417–2421.

Wack, A.; Paik, H.; Javadi-Abhari, A.; Jurcevic, P.; Faro, I.; Gambetta, J. M.; and Johnson, B. R. 2021. Quality, Speed, and Scale: three key attributes to measure the performance of near-term quantum computers.

Wang, S.; Fontana, E.; Cerezo, M.; Sharma, K.; Sone, A.; Cincio, L.; and Coles, P. J. 2021. Noise-induced barren plateaus in variational quantum algorithms. *Nature Communications*, 12(1).

Welch, J.; Greenbaum, D.; Mostame, S.; and Aspuru-Guzik, A. 2014. Efficient quantum circuits for diagonal unitaries without ancillas. *New Journal of Physics*, 16(3): 033040.

Wen, T.; and Keyes, R. 2019. Time Series Anomaly Detection Using Convolutional Neural Networks and Transfer Learning.

Wiebe, N.; Berry, D.; Høyer, P.; and Sanders, B. C. 2010. Higher order decompositions of ordered operator exponentials. *Journal of Physics A: Mathematical and Theoretical*, 43(6): 065203.

Zhang, R.; and Zou, Q. 2018. Time Series Prediction and Anomaly Detection of Light Curve Using LSTM Neural Network. *Journal of Physics: Conference Series*, 1061: 012012.

Zhang, X.; Chowdhury, R. R.; Gupta, R. K.; and Shang, J. 2024. Large language models for time series: A survey. *arXiv preprint arXiv:2402.01801*.

Zhang, Y.; Chen, Y.; Wang, J.; and Pan, Z. 2021. Unsupervised Deep Anomaly Detection for Multi-Sensor Time-Series Signals.