Physica Scripta

OPEN ACCESS

RECEIVED
5 August 2025

REVISED
4 October 2025

ACCEPTED FOR PUBLICATION
14 October 2025

PUBLISHED
28 October 2025

PAPER

Fourier series guided design of quantum convolutional neural networks for enhanced time series forecasting

Sandra Leticia Juárez-Osorio[1], **Mayra Alejandra Rivera-Ruiz**[1], **Andres Mendez-Vazquez**[1,*], **Eduardo Rodriguez-Tello**[2] **and José Mauricio López-Romero**[3]

[1] CINVESTAV Unidad Guadalajara, Av. del Bosque 1145, Zapopan, 45019, Jalisco, Mexico
[2] CINVESTAV Unidad Tamaulipas, Km. 5.5 Carretera Victoria - Soto La Marina, Ciudad Victoria, 87130, Tamaulipas, Mexico
[3] CINVESTAV Unidad Querétaro, Libramiento Norponiente 2000, Fracc. Real de Juriquilla, Santiago de Querétaro, 76230, Querétaro, Mexico
[*] Author to whom any correspondence should be addressed.

E-mail: sandra.juarez@cinvestav.mx, mayra.rivera@cinvestav.mx, andres.mendez@cinvestav.mx, ertello@cinvestav.mx and jm.lopez@cinvestav.mx

Keywords: quantum machine learning, 1D quantum convolution, fourier series, time series forecasting, variational quantum circuits, expressivity, barren plateaus

Abstract

In this work, we explore the application of 1D quantum convolutional layers for time series forecasting, framing the problem within the context of machine learning and quantum circuit expressivity. By encoding multiple time steps as features into a quantum circuit, the forecasting task becomes intrinsically multidimensional. We leverage the recent theoretical results that describe Variational Quantum Circuits (VQCs) as multidimensional Fourier series, to design a more expressive 1D quantum convolutional neural network in which an enhanced performance is obtained by reuploading the data. We aim to relate the number of accessible coefficients in each architecture design with the obtained metrics. Our analysis incorporates key machine learning concepts such as expressivity and the presence of barren plateaus, evaluating their impact on generalization and training efficiency. Notably, we show that even with a limited number of trainable parameters, quantum circuits can approximate high-degree Fourier functions, underscoring their expressive capacity and computational efficiency. Our results demonstrate that more expressive ansatz, characterized by richer Fourier spectra, consistently yield improved predictive performance as the number of qubits increases. This study bridges theoretical insights from quantum physics with practical advances in machine learning model design for time series forecasting.

1. Introduction

Today, Quantum Computing (QC) is still in the Noisy intermediate-Scale Quantum (NISQ) era: devices with a small number of qubits and errors. Variational Quantum Circuits work with few qubits, being suitable to this kind of devices. These VQCs are capable of working in conjunction with classical layers in order to divide the task between a classical and a quantum computer. Several applications to these hybrid algorithms can be found in the literature (Cerezo *et al* 2021, De Luca 2021, Li and Deng 2021). In this work, we focus on Quantum Machine Learning, specifically in the utilization of Variational Quantum Circuits (VQCs) to construct a quantum version of the widely used Convolutional Neural Networks (CNN's) (Henderson *et al* 2020).

In the literature, numerous instances abound where Quantum Machine Learning (QML) algorithms have demonstrated superior or comparable performance to their classical counterparts. This holds true across various scenarios, encompassing toy datasets (Schuld *et al* 2020, Park *et al* 2023, Mari *et al* 2020) as well as real-world applications, such as medical image classification (Sameer and Gupta (2022), Houssein *et al* (2022), Shahwar *et al* (2022)), defects detection in materials (Yang and Sun 2022), and time series forecasting (Alejandra *et al* 2022, Rivera-Ruiz *et al* 2024). Notably, Quantum Convolutional Neural Networks (QCNNs) have

found application not only in toy datasets like MNIST and Fashion MNIST (Hur *et al* 2022, Henderson *et al* 2020), but also in practical contexts, such as protein distance prediction (Hong *et al* 2021), the identification of COVID-19-infected patients (Houssein *et al* 2022), a fast quantum convolution for autonomous driving applications (Roh *et al* 2024), and image denoising (Li *et al* 2025).

Our focus in this study is to build upon existing theoretical insights to guide the design of more effective quantum architectures, specifically drawing from Schuld *et al* (2021), Casas and Cervera-Lierta (2023), Sim *et al* (2019), Holmes *et al* (2022). These works establish a general framework in which Variational Quantum Circuits (VQCs) can be analyzed in terms of expressivity, providing useful criteria for designing circuits.

In particular, Schuld *et al* (2021), Casas and Cervera-Lierta (2023) demonstrate that VQCs can be interpreted as truncated Fourier series, where the number of accessible coefficients depends on the ansatz structure and the use of data reuploading. More coefficients translate into richer function classes that the circuit can approximate, i.e., greater expressivity. Complementarily, Sim *et al* (2019) introduces a measure to quantify ansatz expressivity, and Holmes *et al* (2022) highlights the challenge of barren plateaus, where highly expressive models may suffer from flat cost landscapes. These theoretical foundations motivate our approach.

The main contribution of this work is to empirically validate these theoretical predictions in the context of time series forecasting by means of a 1D QCNN architecture. To the best of our knowledge, we present the first systematic study that connects the Fourier spectrum accessible to different circuit designs with the metrics obtained in real forecasting tasks. Our experiments evaluate multiple architectural variations, showing that more expressive circuits, enabled by data reuploading, achieve improved performance while requiring fewer trainable parameters than suggested by prior theoretical bounds Casas and Cervera-Lierta (2023). These results confirm that designing architectures through the Fourier perspective provides a practical pathway to enhance expressivity and performance without increasing qubit counts, an important consideration given current hardware limitations.

The rest of the work is organized in the following way. First in section 2, a general background is presented. In this part, we briefly introduce quantum computing, VQC's and the theory of VQC's as Fourier series from Schuld *et al* (2021) and Casas and Cervera-Lierta (2023) is presented. Next, the 1D quantum convolution (Rivera-Ruiz *et al* 2024) is explained and how it can be seen from the context of multidimensional Fourier series. Also we describe the expressivity as in Sim *et al* (2019) and the measure of barren plateaus as in Holmes *et al* (2022). In section 3 we explain the architectures that will be tested in this work, the datasets the description of the simulations. In the results, section 4, we discuss the relevance of reuploading data, the number of necessary trainable parameters and the different results obtained by utilizing various architectures, ansatz and number of qubits. Finally, a conclusion is presented in the last section.

2. Background and related work

2.1. Quantum computing

The emergence of quantum computers, proposed by Feynman in 1982 for simulating quantum systems, has evolved significantly in the past four decades, despite accuracy limitations in quantum processors. Quantum algorithms have supremacy in certain problems: For example Shor's algorithm for factoring and Grover's search algorithm. In contrast to classical computers, which follow deterministic laws of physics, microscopic quantum systems when isolated from their environment, exhibit non-classical behaviors such as uncertainty, collapse, and entanglement (Nielsen and Chuang 2000, Preskill 2023, Feynman 2018).

Now, lets briefly introduce some concepts of quantum computing. Analogously to the bit in classical computing, the quantum bit or qubit is the basic unit of information processing used in quantum computing. Unlike classical bits, which can only exist in states of 0 or 1, qubits follow the principles of superposition and entanglement. In superposition, a qubit can simultaneously occupy both states 0 and 1 until measured, providing a potential computational advantage over classical bits. In quantum mechanics, qubits and their interactions are mathematically represented as vectors of the Hilbert space. The state space of n qubits is described as a tensor product space, enabling the representation of complex quantum states and operations (Nielsen and Chuang 2000).

Quantum gates are unitary operators utilized to perform transformations on qubits. These gates, similarly to classical logic gates, perform operations on qubits, preserving the quantum information encoded within them. With the sequential application of quantum gates, quantum circuits manipulate qubit states to perform specific computational tasks. Various quantum gates, including Pauli matrices and the Hadamard gate are utilized in circuits to perform operations on qubits, enabling the creation of entangled states and the implementation of quantum algorithms. These gates, together with control operations such as the CNOT and CZ gates, form the basis of quantum circuits (Nielsen and Chuang 2000). Those last kind of gates are utilized to generate entangled states. Multiple advantages on having entagled states can be mentioned, but particularly in the

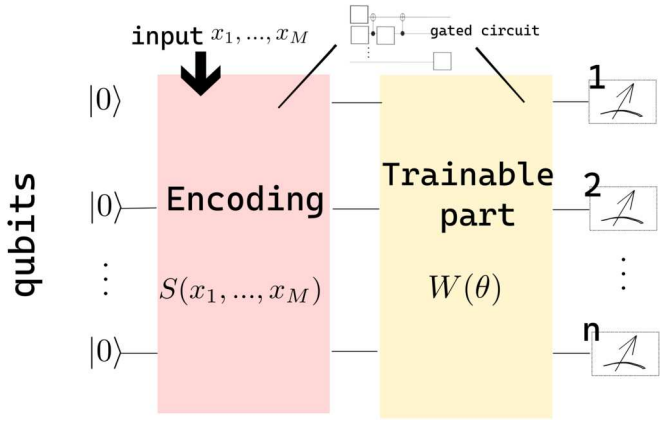


Figure 1. Variational Quantum Circuit (VQC's). The classical input **x** is encoded into a quantum state $|\Psi_{\text{in}}(\mathbf{x})\rangle$ with the unitary transformation $U_{\text{in}}(\boldsymbol{x})$ applied to the initial quantum state $|0\rangle^{\otimes n}$. In the second step, a unitary transformation $U(\boldsymbol{\theta})$ with trainable parameters is applied. Finally, a measurement is made on the qubits.

context of VQCs and Quantum Machine Learning, having a circuit with a strong entangling structure is capable of better covering the Hilbert space and to capture correlations in data (Sim *et al* (2019)).

Several quantum-inspired models have proposed using tensor network architectures to efficiently capture complex feature correlations in high-dimensional spaces. For example, Matrix Product States (MPS) have been used in supervised learning tasks to compress input representations while retaining expressivity (Stoudenmire and Schwab 2016). Other approaches, such as those by Huggins *et al*, explore the implementation of tensor network structures like MPS and Tree Tensor Networks (TTNs) on quantum hardware (Huggins *et al* 2019). More recently, the Residual Tensor Train (ResTT) model extends classical tensor train architectures with skip connections to capture feature interactions of multiple orders while improving training stability through mean-field analysis (Chen *et al* 2022).

Quantum generative models are also an active line of research. Zhang *et al* (2025) proposed a QGAN based on the quantum Born machine, illustrating the role of circuit design in enhancing expressivity. Pei *et al* (2025) developed a one-to-many image generation model with parameterized quantum circuits, showing how circuit structure and entangling layers directly affect generative capacity. More recently, Gong *et al* (2025) introduced DDQGAN, a quantum generative adversarial network with dual QCNN-based discriminators, demonstrating how quantum convolutional layers can improve both stability and expressivity in image generation tasks.

In this work we focus on Quantum Convolutional Neural Networks (QCNNs), which have been applied not only to toy datasets such as MNIST and Fashion MNIST (Hur *et al* 2022, Henderson *et al* 2020), but also to practical problems including protein distance prediction (Hong *et al* 2021) and the identification of COVID-19-infected patients (Houssein *et al* 2022). Building on this line of research, Roh *et al* (2024) proposed a fast QCNN for object detection, introducing channel uploading to encode multiple data channels efficiently. While focused on vision tasks, their work reinforces the role of data encoding and re-uploading strategies in improving scalability and expressivity. More recently, hybrid quantum–classical convolutional networks have been explored for image processing tasks; for instance, Li *et al* (2025) introduced QDnCNN for image denoising, where a quanvolutional layer is combined with classical filters to enhance feature extraction. Although centered on images, this study further supports the relevance of quantum convolutional layers for strengthening representational power in practical machine learning applications. In particular, we build upon the one-dimensional implementation of QCNNs introduced by Rivera-Ruiz *et al* (2024) in the context of time series forecasting.

2.2. Variational quantum circuits

Variational Quantum Circuits are trainable quantum circuits that are widely used as quantum neural networks for different tasks. The general structure of a VQC is presented in figure 1. VQCs are quantum algorithms that capture correlations in data using entangling properties (Schuld *et al* 2020). In today's noisy intermediate-scale quantum computers (NISQ), which suffer from noise and qubit limitations, the VQC is the leading strategy due to their shallow depth (Cerezo *et al* 2021).

At figure 1, the first step is to encode the classical input x into a vector in the Hilbert space. This is accomplished by applying a unitary transformation $S_{\text{in}}(x)$ to the initial state, which is generally chosen as $|0\rangle^{\otimes n}$ (Cerezo *et al* 2021). Several strategies such as amplitude and rotation encoding are utilized in the encoding stage. Particularly, in this work we will utilize rotation encoding, which embeds a classical point x_i into a single qubit.

After encoding the classical input, the state vector is passed through a set of quantum operations $W(\boldsymbol{\theta})$ depending on an optimizable parameter θ (Cerezo *et al* 2021). The form of the ansatz depends on the specific task; although some ansatz architectures are generic. The parameters $\boldsymbol{\theta}$ can be encoded in a unitary $W(\theta)$ applied to the input states, and the way this parametrization is defined can have a significant impact on training since it defines the shape of the cost function (Schuld *et al* 2020). The quality of an ansatz can be determined by its expressivity and its entangling capability. It is said that an ansatz is expressible if the circuit can explore the entire space of quantum states (Cerezo *et al* 2021). After passing through the encoding and trainable unitary transformations U the final state would be given as:

$$|\Psi\rangle = U|0\rangle^{\otimes n}, \tag{1}$$

Now, considering that the initialized state is $|0\rangle^{\otimes n} = (1, 0, \cdots 0)^T$, then the ith element of this state would be given as:

$$|\Psi_i\rangle = U_{ij}\delta_{j1} = U_{i1} \tag{2}$$

2.3. VQCs as fourier series

A strategy presented in Schuld *et al* (2021), Casas and Cervera-Lierta (2023) is to use L layers, reuploading the encoded data at each step. A layer consists of encoding gates $S(x) = e^{ixH}$, with H a Hamiltonian, followed by trainable gates $W(\theta)$. After L layers, the resulting state can be expressed as a linear combination of exponentials with frequencies given by sums of the eigenvalues of H:

$$|\Psi\rangle_i = \sum_{j_1,\cdots,j_L=1}^{N} e^{i(\lambda_{j_1}+\cdots+\lambda_{j_L})x}\, W^{(L+1)}_{ij_L} \cdots W^{(1)}_{j_1 1}, \tag{3}$$

with $N = 2^n$ for n qubits. Measuring an observable M yields

$$\langle M\rangle = \langle\Psi|M|\Psi\rangle, \tag{4}$$

and combining these results leads to

$$f(x) = \sum_{\omega} c_\omega e^{i\omega x}, \tag{5}$$

which is a Fourier series. Here, the accessible frequencies ω depend only on the encoding (the eigenvalues of H), while the coefficients c_ω are determined by the trainable and measurement parts of the circuit. Reuploading increases the accessible frequency spectrum linearly with L, thus enhancing the expressivity of the circuit.

2.4. 1D Quantum convolution as multidimensional Fourier series

The 1D quantum convolution (1D QCNN) proposed in Rivera-Ruiz *et al* (2024) (figure 2) adapts the idea of Henderson *et al* (2020) to one-dimensional signals with a trainable quantum layer. Instead of matrix multiplications as in classical convolution, subsequences of length k are encoded into a VQC (figure 1), processed by trainable gates, and decoded via measurements. to yield real-valued outputs. For an input sequence of size s, the output matrix has dimensions (n, o) with $o = (s + 2 + p - k)/r + 1$, where p is the padding and r the stride. In Rivera-Ruiz *et al* (2024), this quantum layer was combined with classical convolutions in a hybrid model, outperforming its classical counterpart. Here, we focus on assessing different quantum architectures for time series forecasting, where multiple points are encoded to predict the next one. Details of the architecture and implementation are given in section 3.

Since our task requires encoding several points simultaneously, the multidimensional Fourier analysis of Casas and Cervera-Lierta (2023) becomes relevant. There, a multidimensional truncated Fourier series is obtained:

$$f(\vec{x}) = \sum_{\omega_1,\ldots,\,\omega_M=-D}^{D} c_{\vec{\omega}} e^{i\vec{x}\cdot\vec{\omega}}, \tag{6}$$

with $\vec{x}$ an M-dimensional vector, $D = \max(\omega_1,\ldots,\,\omega_M)$ the series degree, and $c_{\vec{\omega}}$ complex coefficients satisfying $c_\omega = c^*_{-\omega}$. Although our time series are unidimensional, using multiple past points makes the problem effectively multidimensional, where each time step corresponds to a feature.

The degrees of freedom of a Fourier series are

$$\nu = (2D + 1)^M, \tag{7}$$

and an expressive ansatz requires at least ν trainable parameters. Among the architectures in Casas and Cervera-Lierta (2023), only the *super parallel ansatz* always satisfies this condition. It stacks layers in both depth and width, requiring $n = LM$ qubits, with each layer defined as

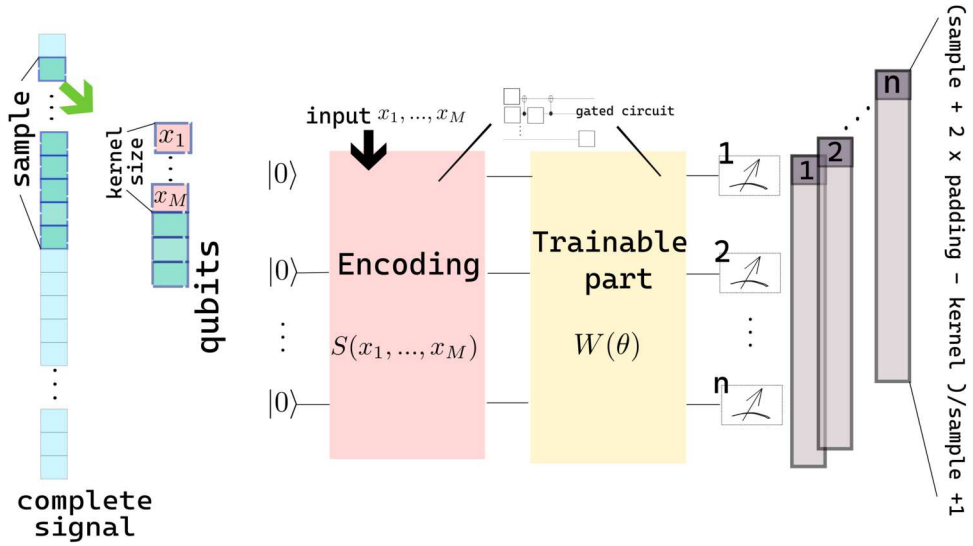


Figure 2. 1D quantum convolution as introduced in Rivera-Ruiz *et al* (2024), which utilizes a VQC. This circuit has n qubits and encodes M features. The output is a matrix with classical values.

$$L(\vec{x}, \vec{\theta}) = (\bigotimes_{l=1}^{L} (\bigotimes_{m=1}^{M} S(x_m))) W^{(l)}(\vec{\theta}), \tag{8}$$

and a total of $N_p = (L+1)(2^{2ML} - 1)$ free parameters, since each unitary block has $N^2 - 1$. While expressive, this architecture poses significant training costs, as discussed in section 3.

2.5. Expressivity

In Sim *et al* (2019), the expressivity of a circuit is defined as its ability to generate states representative of the Hilbert space. This is quantified by comparing the distribution of states produced by the circuit with that of Haar-random states. Given a Hilbert space of dimension $N = 2^n$, the Haar measure provides the reference distribution, and the corresponding fidelity distribution is known analytically as (Di Matteo 2021):

$$P_{\text{Haar}}(F) = (N - 1)(1 - F)^{N-2}. \tag{9}$$

The expressivity is then estimated by sampling circuit states, constructing the fidelity distribution $P_c(F)$, and computing its divergence from the Haar case using the Kullback-Leibler divergence:

$$E = D_{KL}(P_c(F) \,\|\, P_{\text{Haar}}(F)). \tag{10}$$

A smaller KL divergence indicates higher expressivity, i.e., the circuit explores the unitary space more uniformly. In this work we also relate this notion to the Fourier spectrum of the circuit, where richer accessible frequencies imply greater expressivity.

2.6. Barren plateaus

A circuit is said to exhibit a barren plateau if the gradient vanishes exponentially when adding more qubits, leading the optimizer to be trapped in a local minimum. The average of $\partial_k C$, with C the cost function over the set of trainable parameters $\boldsymbol{\theta}$ is zero, meaning that the gradient cannot take any direction, or what is known as vanishing gradient. In Holmes *et al* (2022), the trainability of an ansatz is assessed with the Chebyshev inequality, which bounds the probability that the partial derivative deviates from its average of zero:

$$P(|\partial_k C| \geqslant \delta) \leqslant \frac{\text{Var}[\partial_k C]}{\delta^2}. \tag{11}$$

From this equation, it can be interpreted that when the variance is small, the gradient vanishes with high probability and those cases are untrainable. In Holmes *et al* (2022), it is found that expressive ansatz exhibits barren plateaus.

3. Proposed model

In this work the *super parallel ansatz* depicted in figure 3 is the general schema to follow, varying the part $W(\vec{\theta})$ to compare the performance both in metrics and in the subset of coefficients that each choice of $W(\vec{\theta})$ is capable to achieve. As mentioned above, utilizing gates of $N^2 - 1$ trainable parameters would imply large

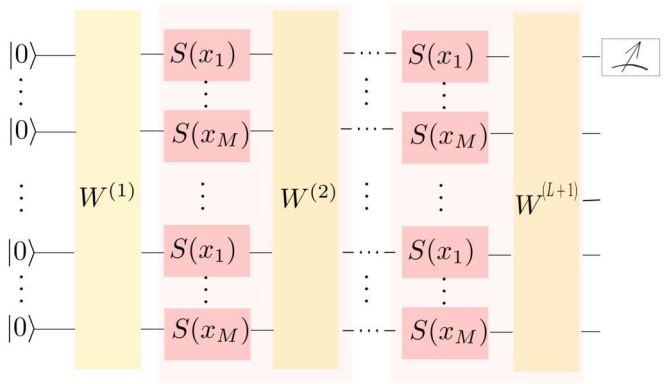


Figure 3. Super parallel architecture proposed in Casas and Cervera-Lierta (2023) which outputs a multidimensional Fourier series. It encodes M features which are reuploaded vertically and horizontally. The outputs are classical values.

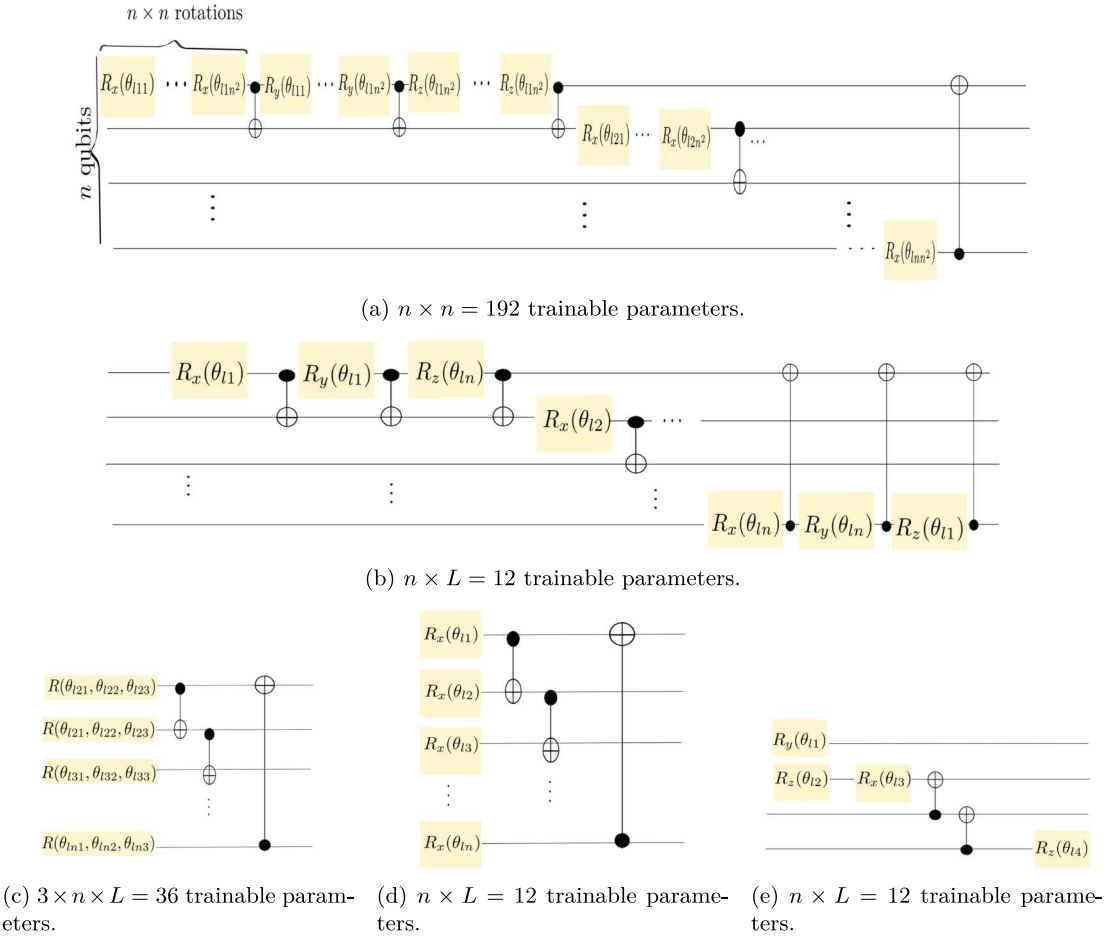


Figure 4. Ansatz for the trainable part $W(\theta)$. Here n is the number of qubits. This repeats in each layer l of the circuit.

training times. For this reason, in this work the choice of $W(\vec{\theta})$ will be layers with only $n \times n$ trainable parameters (figure 4(a)), which still fulfills the condition $N_p > \nu$. Also, those results will be compared against choices of $W(\vec{\theta})$ with only n (figures 4(b), (d), (e)) and $3 \times n$ (figure 4(c)) trainable parameters per layer, which do not fulfill the condition $N_p > \nu$.

The objective of this work is to compare the performance of different choices of $W(\vec{\theta})$, then the encoding part is the same for all the tests and is described in the following equation,

$$\bigotimes_{m=1}^{M} R_y(x_m). \tag{12}$$

This is repeated *vertically* in each of the L layers of the circuit, as illustrated in figure 3.

The 1D quantum convolution presented in figure 2 is utilized in this work. The sliding window size for the experiments in this work is $k = 2$ and s varies for each dataset. This effectively allows the circuit to capture the dependencies between adjacent points. The architecture of the circuit depicted in figure 3 requires $k \times L$ qubits. This is an advantage of utilizing quantum convolution instead of the quantum version of a multilayer perceptron in which $s \times L$ qubits would be required if the *super parallel* architecture is followed.

The *super parallel* architecture is analyzed with 2, 3 and 4 layers. With 2 layers and $k = 2$, 4 qubits are required. With 3 and 4 layers 6 and 8 qubits are required, respectively. For those architectures, the degree of the output Fourier series is expected to be $D = 4, 6, 8$ respectively. However, depending on the particular choice of $W(\theta)$ some of the coefficients might be set to zero, limiting the expressivity of the model. The observable R_z is measured in all qubits at the end of the circuit, and the expected value (a real number) of each qubit is passed to the classical part. The output features of the 1D quantum convolution is a matrix with dimensions (n, o), with $o = (s + 2 \times p - k)/r + 1$. Here, p is the padding and r is the stride. In this case, the sample chain of size s is padded at both edges with a zero. The chain is swept with a stride $r = 1$. The classical part consists of a ReLU activation function, a max pooling over the dimension o of the output of the quantum convolution, and a linear layer that maps from n elements to finally output the predicted point of the time series.

3.1. Computational cost

The inference time of our one-dimensional quantum convolutional layer is dominated by the cost of simulating quantum circuits with n qubits, which requires $\mathcal{O}(2^n)$ operations per quantum gate due to matrix–vector multiplications in a Hilbert space of dimension 2^n. The overall complexity therefore scales as the number of executed gates multiplied by this exponential factor. In the *no reupload* architecture, the input kernel of size k is encoded only once, leading to a gate count of approximately $\mathcal{O}(L \cdot |W| + k)$, where $|W|$ denotes the number of quantum gates defined by one layer of the ansatz W. In contrast, the *parallel* and *superparallel* architectures reinsert the input data at every layer, which increases the cost of data encoding to $\mathcal{O}(L \cdot k)$ additional rotations, on top of the $\mathcal{O}(L \cdot |W|)$ trainable gates. As a result, both parallel variants exhibit higher inference times than the no reupload case. While the superparallel design further distributes the inputs across blocks of qubits, its asymptotic scaling remains the same as the parallel case, though in practice it incurs slightly larger inference times due to repeated encoding overhead. Altogether, the exponential factor 2^n arising from the state-vector simulation remains the dominant contribution to the computational cost.

3.2. Data

In this work, we analyze the following datasets: as a toy example, we use the third-order Legendre polynomial with randomly seeded noise; as a small-scale dataset, we consider the USD to EUR exchange rate Antweiler (2023); and for more complex and realistic cases, we examine the SP500 (Wmcginn 2021) and Bitcoin (Zieliński 2017) datasets.

The dataset of the third Legendre polynomial consist in points generated by the equation $P_3(x) = \frac{1}{2}(3x^2 - 1)$. Random seeded noise was added in each point. The subsequences introduced in the 1D convolution kernel are of 5 points. 750 points are for training and 250 for testing.

The exchange rate data between USD and EUR obtained from (Antweiler 2023) contains daily observations from January 1, 2020, to July 8, 2021. The model is constructed with 376 simulation data points, given by the sequence $[x(t-4), x(t-3), x(t-2), x(t-1), x(t); x(t+1)]$ for t ranging from 5 to 380. The first 300 data points are designated for training, and the remaining points are reserved for testing.

The dataset of Bitcoin (Zieliński 2017) and SP500 (Wmcginn 2021) have data from the open value. The bitcoin dataset has a temporality of 1 minute and the SP500 considers daily observations. In both cases sequences of 5 points to predict the point number 6. In the training stage, 3000 points were considered and 750 points for testing.

The datasets for this study are carefully selected to evaluate our proposed QCNN architecture across a spectrum of increasing complexity and realism. This multi-level approach allows for a comprehensive assessment of the model's capabilities. We begin with a **synthetic toy example** (a third-order Legendre polynomial with noise) to validate the model's fundamental capacity to learn a known non-linear function in a controlled environment. Next, we use a **small-scale, real-world dataset** (USD to EUR exchange rate) Antweiler (2023) to assess the model's performance in a data-limited regime, a common practical challenge. Finally, we test the model's robustness and scalability on two **large-scale, complex financial datasets** (S&P 500 and Bitcoin) Wmcginn (2021), Zieliński (2017). These are well-known benchmarks in the forecasting community, chosen

specifically for their high volatility, non-stationarity, and noisy dynamics, which provide a challenging testbed for any advanced forecasting model.

4. Results

All results are assessed by comparing the root mean squared error (RMSE), mean absolute error (MAE) and mean percentage error (MAPE). These are standard metrics for forecasting models and since they are related to error, a smaller value is indicative of a better result.

In this section the following cases are presented. First, the relevance of reuploading is analyzed. For this, we utilize the *parallel* architecture described in Casas and Cervera-Lierta (2023). The *parallel* architecture is similar to the *super parallel*, but without reuploading the data vertically: The data is only repeated in a series of layers. Also, a comparison between the *parallel* and *super parallel* architecture is presented. Additionally, the performances of ansatz in figure 4 are calculated. And finally, to assess the roboustness of our model, the results for the SP500 datasets are benchmarked against a classical CNN and results from literature. Ansatz (c) is known as *Strongly Entangling Layers*, is a template from *Pennylane* inspired by the work presented in Schuld *et al* (2020). Ansatz (d) is the template known as *Basic Entangler*, also from *Pennylane*, with a similar structure as Strongly Entangling, but with a rotation only in one direction. Ansatz (e) is generated with the template *Random Layers* of *Pennylane* and seed=1234, which randomly distributes two-qubit gates and rotations in the circuit. The number of random rotations is derived from the second dimension of weights, and the number of CNOT gates is $\frac{1}{3}$ of the number of rotations.

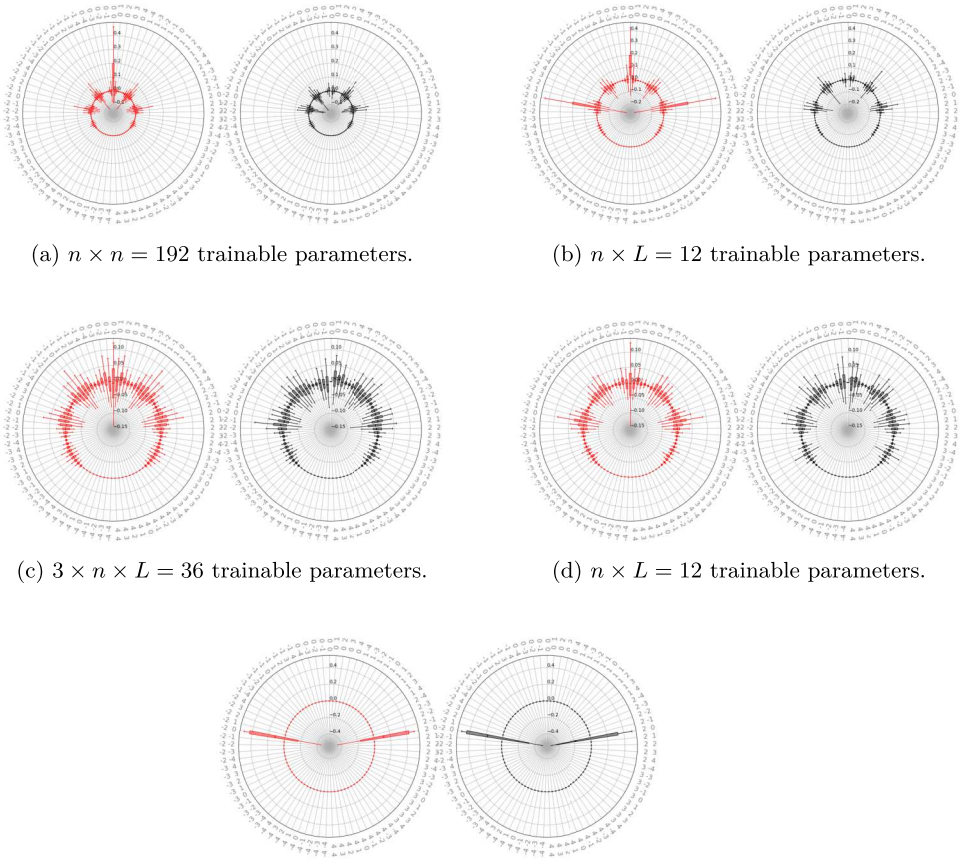
The Fourier coefficients will be computed for the ansatz in figure 4 to assess the expressivity of the circuit in the context of the FS . Also, the expressivity and the variance of the gradient of the cost function are computed. To calculate the expressivity, the calculations were generated by sampling 5000 states and using a bin size of 75, following the parameters suggested in Sim *et al* (2019). The circuit is initialized with different sets of random weights and the coefficients are computed from the output of the circuit using a discrete Fourier transform. Those coefficients can be plotted, allowing to easily evaluate the richness of the accessible coefficients of each architecture. The analytical expression of the part of the circuit corresponding to the coefficients is highly non-linear and too complex, then a graphical analysis is more suitable to provide an insight into the differences between each choice of the variational part.

The computations were executed with the simulator `Pennylane` (Bergholm 2018) on the device *lightning.gpu*, using the `jax-jit` interface. This choice allows us to isolate and analyze the intrinsic expressivity and learning capabilities of the proposed quantum architectures without the confounding effects of hardware noise. Although noise was not considered here, future work could extend this analysis by employing noisy simulators such as `default.mixed` in PennyLane or the Qiskit Aer simulator, which support realistic noise models. Running on actual quantum hardware remains prohibitively expensive for large-scale experiment. Optimization of weights was performed with the `backprop` differentiation method. Just-in-time (`jit`) compilation was applied to both the circuit function and the optimization steps. The use of `jax`, `jit`, and vectorization enabled execution in competitive times. All calculations were carried out on a GPU RTX 3080 and a Ryzen 7 5700X processor. All experiments were conducted with a learning rate of 5×10^{-4} using the ADAM optimizer (Kingma and Ba 2014). The weights of the classical model were also initialized with the Xavier method (Glorot and Bengio 2010). Each experiment was repeated 20 times with a different seed, and the average result was reported. A summarry of the experimental setup is presented in table 1.

4.1. Number of necessary trainable parameters

The sets of accessible coefficients for each of the ansatz in figure 4 are depicted in figure 5. In this figure, the plots in left and red represent the real part and the right plots in black the imaginary part of the circuit's Fourier coefficients. The labels of the circles are the frequencies. A bigger bar represents a greater distribution of values for a coefficient. Those results were obtained with 100 different sets of randomly initialized weights. It can be observed that even when the architectures (b), (c), and (d) do not fulfill the condition $N_p > \nu$, they are capable of producing non-zero coefficients. Moreover, it can be noted that the architectures (c) and (d) are capable of producing a richer set of coefficients than (a), which fulfills the condition. The expected degree for all the architectures is 4, then according to equation (7), at least 81 free parameters would be needed to have a general enough circuit. It is noticeable that (b) and (d) with as few as 12 trainable parameters and (c) with 36 trainable parameters are capable of producing more non-zero coefficients than expected. (b) is capable of producing coefficients corresponding to a Fourier series of degree $D = 2$, which would need 25 free parameters. (c) and (d) have non-zero coefficients corresponding to a degree $D = 3$, for which 49 free parameters would be needed.

In practical terms, achieving high-degree Fourier functions with fewer trainable parameters is highly desirable due to the reduction in training times. For the ansatz (b), (c), and (d) of figure 4, with 4 qubits and 2 layers,



(a) $n \times n = 192$ trainable parameters.

(b) $n \times L = 12$ trainable parameters.

(c) $3 \times n \times L = 36$ trainable parameters.

(d) $n \times L = 12$ trainable parameters.

(e) $n \times L = 12$ trainable parameters.

Figure 5. Accessible coefficients for the architectures of figure 4. The plots on the left in red represent the real part and the plots on the right in black are the imaginary part of the circuit's Fourier coefficients. The labels are the frequencies, with c_0 at the top of each circle; positive frequencies increase in a clockwise direction and negative frequencies in a counter-clockwise direction. The figure was generated by sampling the circuit with 100 different sets of randomly initialized weights. Longer and more prominent bars represent a richer set of values for a coefficient.

Table 1. Summary of hyperparameters and experimental Setup.

Hyperparameter	Value
Framework and Simulation	
Framework	Pennylane
Simulator Device	`lightning.gpu`
Interface	`jax-jit`
Differentiation Method	`backprop`
Training Parameters	
Optimizer	ADAM
Learning Rate	5×10^{-4}
Epochs	25
Weight Initialization	Xavier method
Execution Environment	
Repetitions per Experiment	20 (with different seeds)
Hardware	GPU: NVIDIA RTX 3080
	CPU: AMD Ryzen 7 5700X

training took on average 0.948 *s* per epoch. On the other hand, ansatz (a) took on average 16.06 *s* to complete 1 epoch. Also, the circuits with fewer parameters were capable of achieving comparable metrics, to illustrate this, the results for the Legendre dataset are presented in table 2. The other datasets exhibited similar behavior.

Table 2. Comparison of results obtained with the ansatz of figure 4 for the Legendre dataset.

Ansatz	Trainable parameters	RMSE	MAE	MAPE	Time per epoch(s)
(a)	192	0.1440	0.1109	0.2890	16.06
Strongly Entangling (c)	36	0.1330	0.1079	0.2942	1.372
Basic Entangler (d)	12	0.1333	0.1081	0.2947	0.948
Custom Layers (b)	12	**0.1229**	**0.0986**	**0.2791**	2.301
Random Layers (e)	12	0.1654	0.1393	0.4580	1.097

Table 3. Degree obtained for 6 and 8 qubits with 3 and 4 layers, respectively. In the last column, the parameters that in theory would be needed to fulfill the condition $N_p > \nu$.

Ansatz	Qubits	Layers	Expected degree	Obtained degree	Trainable parameters	Parameters needed for obtained D
Strongly Entangling	6	3	9	8	72	289
	8	4	16	11	120	529
Basic Entangler	6	3	9	8	24	289
	8	4	16	10	40	441
Custom Layers	6	3	9	5	24	121
	8	4	16	8	40	289
Random Layers	6	3	9	2	24	25
	8	4	16	2	40	25

The degree D of the Fourier series for the *parallel* architecture is equal to the number of layers L and for the *super-parallel* architecture is given by the square of the layers L. When more qubits and layers are added, ansatz (b), (c), and (d) are capable of producing more non-zero terms than expected by their respective number of trainable parameters. This can be appreciated in table 3: in Column 6, the actual trainable parameters of each case are presented, and in Column 7 the parameters needed to fulfill the condition $N_P > \nu$, being the values in Column 7 significantly larger than in Column 6.

Our empirical results, depicted in figure 5, reveal a crucial insight into the practical expressivity of VQCs. We observe that architectures such as (b), (c), and (d) generate a rich set of non-zero Fourier coefficients despite not fulfilling the theoretical condition $N_p \geqslant \nu$ established by Casas and Cervera-Lierta (2023). This finding suggests that this condition should be interpreted as a *sufficient but not strictly necessary* criterion for achieving high expressivity. The underlying reason for this efficiency lies in the highly non-linear and coupled relationship between the trainable ansatz parameters, $\vec{\theta}$, and the resulting Fourier coefficients, $c_{\vec{\omega}}$. This non linearity allows a compact parametrization to influence a broad spectrum of frequencies, enabling the model to approximate high-degree Fourier functions with significantly fewer parameters than the theoretical bound would suggest. This efficiency is a remarkable advantage of quantum circuits, as it drastically mitigates training costs—evidenced by the reduced time per epoch in table 2 without sacrificing the model's representational power.

4.2. Relevance of reuploading

A comparison between the results of the metrics with and without reuploading is presented. In this case, two circuits are compared. For the non-reuploading case, an initial trainable layer $W^{(1)}$, the encoding of the information of the two points kernel into 2 qubits, followed by 4 layers of the trainable ansatz. This is compared against the *parallel* architecture with 2 qubits and 4 layers.

The values of the expressivity, calculated as proposed in Sim *et al* (2019), for each of the trainable layers in figure 4 are calculated for both cases and depicted in table 4. It can be observed that both cases present comparable values, but an analysis of the metrics in the testing stage for each data set shows that the reuploading strategy is superior. This behavior can be explained if the accessible coefficients for each case are analyzed.

Figure 6 illustrates a fundamental difference in the accessible frequency spectrum between the two architectures. Specifically, the model employing data reuploading (the *parallel* architecture) generates a significantly richer and more varied set of Fourier coefficients compared to the non-reuploading model, which is restricted to a much sparser set of frequencies.

The critical role of data reuploading, a strategy theoretically motivated by the work of Schuld *et al* (2021), is empirically validated by our results. Figure 6 provides a visual comparison of the accessible Fourier coefficients, showing that the parallel architecture with reuploading generates a significantly richer and denser frequency spectrum. In contrast, the non-reuploading architecture is confined to a sparse set of frequencies, which

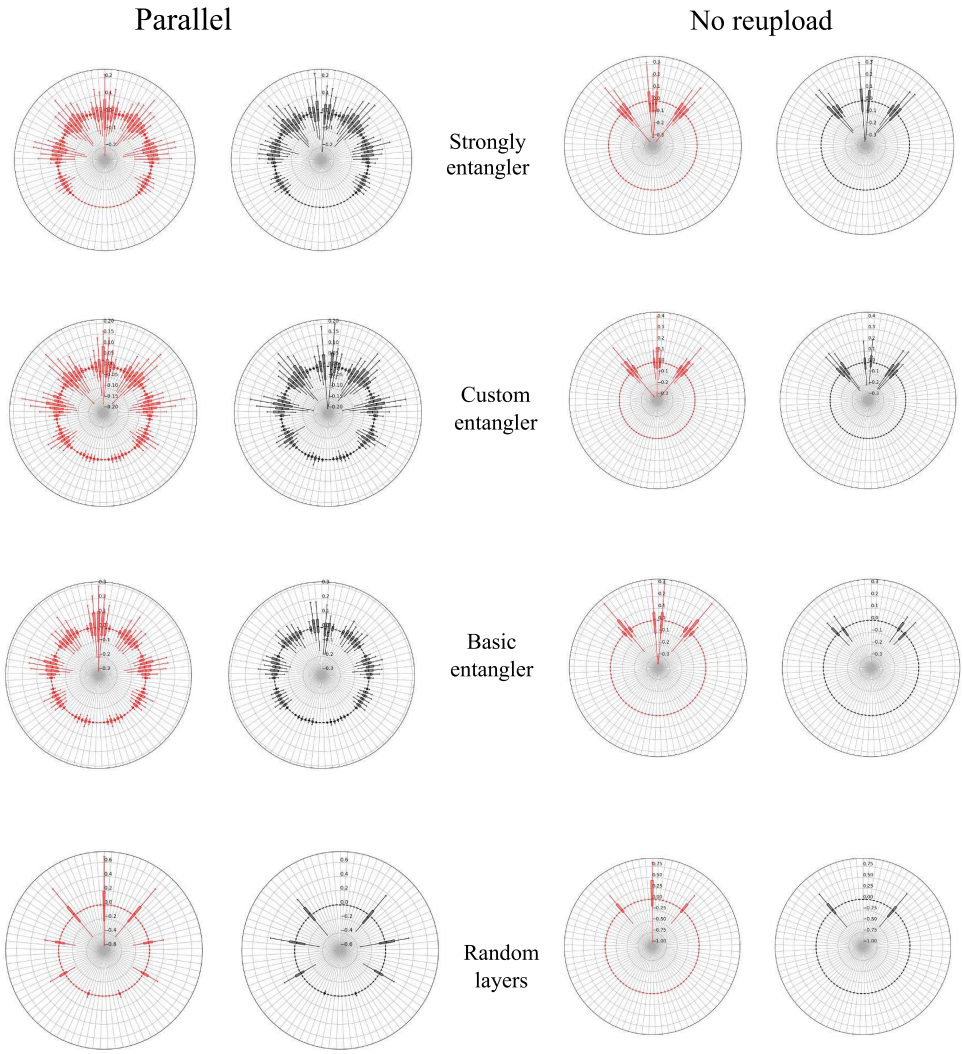


Figure 6. Comparison of the range of accessible Fourier coefficients for both the parallel encoding and the data reuploading models (figure 4). The black bars represent the imaginary parts of the circuit's Fourier coefficients, labeled by their corresponding frequencies (c_ω). Each plot was generated by sampling the circuit 100 times with randomly initialized weights. Taller and more numerous bars indicate a richer set of accessible values for a given coefficient. It is evident that data re-uploading allows the model to access a significantly broader range of coefficients.

Table 4. Expressivity for the *parallel* architecture vs non-reuploading architecture and average taken over the three datasets of improvement (reduction of RMSE, MAE and MAPE) by using reuploading.

Ansatz	Expressivity		Average % improvement by using reuploading		
	Parallel	Non-reuploading	RMSE	MAE	MAPE
Strongly Entangling	0.0071	0.0082	13.77	15.16	28.39
Basic Entangler	0.0210	0.0375	15.57	17.25	37.11
Custom Layers	0.0090	0.0100	13.93	16.50	39.67
Random Layers	0.8729	0.8784	21.99	21.73	7.39

fundamentally limits the complexity of the functions it can approximate. Interestingly, this crucial difference in functional capacity is not fully captured by the expressivity metric from Sim *et al* (2019), as the KL divergence values remain comparable between both models (table 4). This suggests that for function approximation tasks such as time series forecasting, the richness of the accessible Fourier spectrum can be a more direct and predictive indicator of a model's potential performance than a general measure of its ability to explore the Hilbert space. The capacity to model higher-frequency components in the data, enabled directly by reuploading, is what translates into the substantial performance gains observed across all metrics.

Table 5. Values for the expressivity of the *parallel*and super *parallel* architectures for each different ansatz, along with the corresponding percentage improvements in standard metrics when employing the *super parallel* architecture compared to the *parallel* architecture.

Ansatz	Expressivity		Average % improvement by using {super parallel}			
	Parallel	*Super Parallel*	RMSE	MAE	MAPE	Training time % difference
Strongly Entangling	0.0071	0.0033	10.27	9.89	10.53	17.20
Basic Entangler	0.0210	0.0053	7.77	10.54	3.63	18.21
Custom Layers	0.0090	0.0054	19.86	17.61	17.57	14.98
Random Layers	0.8729	2.764	36.93	36.59	19.50	12.11

Table 6. Degree, expressivity, and variance of the derivative of the cost function for the ansatz (b)-(e) in figure 4. The cases with 4, 6, and 8 qubits are analyzed.

Ansatz	Qubits	Layers	Obtained degree	Expressivity	Variance of derivative
Strongly Entangling	4	2	3	0.0033	0.021
	6	3	8	0.0013	0.0063
	8	4	11	0.00011	0.0016
Basic Entangler	4	2	3	0.0053	0.0434
	6	3	8	0.0008	0.0132
	8	4	10	0.0004	0.0032
Custom Layers	4	2	2	0.0054	0.0454
	6	3	5	0.0075	0.0223
	8	4	8	0.0032	0.0112
Random Layers	4	2	2	2.764	0.1270
	6	3	2	5.604	0.0860
	8	4	2	1.737	0.0605

4.3. *Parallel* vs *super parallel* architectures

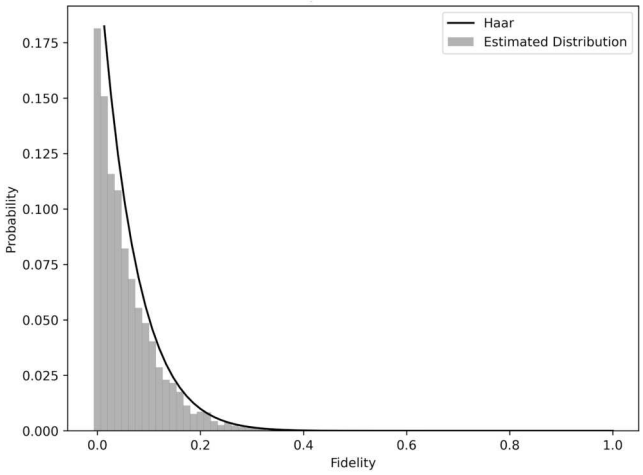
The performance of the *parallel* and *super parallel* architectures presented in Casas and Cervera-Lierta (2023) is compared. A *super parallel* architecture with 4 qubits and a kernel of size 2 is compared against a *parallel* architecture with 2 qubits, a kernel of 2, and 4 layers. In this way, the data is re-uploaded the same number of times, and the circuits are expected to output Fourier series of the same degree $d = 4$.

In general, as can be observed in table 5, the *super parallel* architecture produced better results overall. In this case, no major difference between the accessible coefficients to both architectures is presented, but as can be noted in table 5, the values for the expressivity are better for the *super parallel* case. Even when training times are shorter for the *parallel* case, the difference is not significant enough.

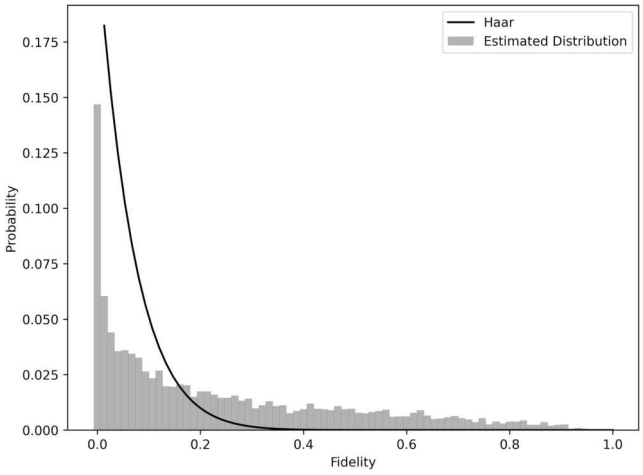
The superior performance of the super parallel architecture over its parallel counterpart, as shown in table 5, highlights that the design of a VQC goes beyond the accessible Fourier spectrum alone. According to the framework in Casas and Cervera-Lierta (2023), the super parallel design encodes features across distinct blocks of qubits ($n = LM$), creating a structure that is simultaneously deep and wide. This configuration facilitates the learning of more intricate correlations between input features (i.e., time steps) by fostering a more complex entanglement pattern across a larger Hilbert space. The improved expressivity values (lower KL divergence) for the super parallel case in table 5 support this interpretation, suggesting that its ansatz explores the state space more uniformly. This enhanced exploratory power enables the model to discover a more optimal functional approximation for the forecasting task, ultimately leading to lower prediction errors.

4.4. Comparison between ansatz

The ansatz (b)–(e) in figure 4 are tested for 4, 6, and 8 qubits with 2, 3, and 4 layers respectively. In table 6 the values of the degree of the obtained Fourier function, the expressivity, and the variance of the derivative of the cost function are presented. It can be observed a relation between the accessible coefficients and the expressivity, both parameters are complementary. The ansatz (e) has only 2 accessible coefficients for the 3 tested cases and also presents a high value of the Kulbak-Leibler divergence, significantly different from (b)-(d). Visually, no major differences can be observed when comparing the histograms of the calculated fidelities overlaid with the Haar fidelities for (b)-(d) for the different number of qubits, but it is illustrative to compare the histograms for the less expressible ansatz (Random Layers (e)) and the most expressible ansatz (Strongly Layers (c)). In figure 7 this comparison is presented for the case with 4 qubits. A clear difference between the calculated and theoretical



(a) Expressivity (KL divergence) =0.0033.



(b) Expressivity (KL divergence) =2.764

Figure 7. Histograms of estimated fidelities, overlaid with fidelities of the Haar-distributed ensemble. Values in subcaptions are calculated as explained in Subsection 2.5 and in Sim *et al* (2019).

fidelities is observed for the case with low expressivity in the Random Layers ansatz (Figure 7(a)) and a closer proximity between both is appreciated for the case with high expressivity with the Strongly Layers ansatz (figure 7(b)). As it can be noted from figure 4, the difference between this ansatz and the three others is the number of entangling CNOT gates; while the other cases have all their qubits entangled, in (e) only $\frac{1}{3}$ of the qubits are entangled. The results for (b)-(d) are comparable, but some differences are evident. The Basic Entangler and the Strongly Entangling have a similar structure: each qubit is rotated first and after that, all qubits are entangled. This similarity results in the closeness of the three analyzed values; those are also slightly different from Custom Layers (which has a different structure), especially in the manner that the values evolve when adding more qubits. This impact of the structure of an ansatz can also be noted in figure 5: (a) and (d) have the same accessible coefficients despite the difference of trainable parameters (192 and 12, respectively) and the same happens for (b) and (c). The results for (b) are superior to (c). This is easily explainable since (b) performs a rotation in the 3 directions, being able to cover the Hilbert space in a more complete form. Also, (b) has three times more trainable parameters than (c). The expressivity and the accessible coefficients for (b) grow when adding more qubits, but this also implies a lower value for the variance of the derivative of the cost function, which results in a flat cost landscape. Ansatz (d) presents lower values for expressivity and has less accessible coefficients than (b) and (c), but on the other hand the value of the variance evolves slowly, implying better trainability.

Our comparison between different ansatz structures provides a clear empirical demonstration of the theoretical trade-off between expressivity and trainability, a central challenge in QML formalized by Holmes *et al* (2022). At one extreme, the Random Layers ansatz (e), characterized by a sparse entangling structure, is unable

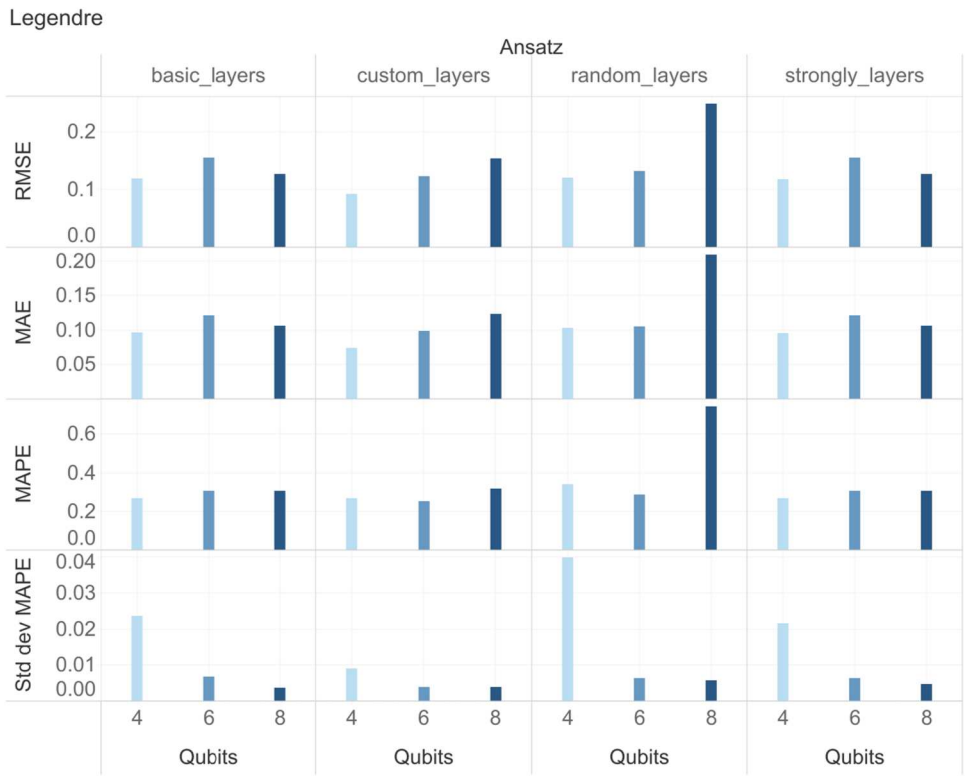


Figure 8. Comparison between ansatz for the Legendre polynomial dataset 4, (light blue) 6, and 8 (dark blue) qubits. The values for RMSE, MAE, and MAPE are presented, then a smaller bar represents a better result. Results are similar but the 8 qubits configuration achieves a lower standar deviation.

to generate a rich Fourier spectrum or effectively explore the Hilbert space, as indicated by its consistently high KL divergence (table 6). This lack of expressive power directly results in poor predictive performance across all datasets. At the other end, the Strongly Entangling ansatz consistently demonstrates the highest expressivity and accesses the richest spectrum. However, this expressive power comes at a cost: a rapidly decreasing gradient variance as the number of qubits increases (table 6). This is a clear signature of the onset of barren plateaus, a phenomenon that can severely impede optimization. This trade-off becomes particularly relevant when analyzing the model's performance on real-world datasets with varying complexity and size, as discussed in the following section.

4.5. Model's performance

Testing the impact of the obtained degree values, expressiveness, and variance with real data is important. For this purpose, the datasets described in section 3.1 are utilized with the hybrid model described in section 3. Those results are depicted in the bar plots of figures 8– 11. In those plots the obtained metrics for 4 (light blue), 6, and 8 (dark blue) qubits and for each of the ansatz of figure 4. Since the metrics RMSE, MAE and MAPE are related to the error, a smaller bar represents a better result.

The Legendre dataset is an example of a simple dataset. This case is presented in figure 8. Similar results are observed for RMSE, MAE and MAPE by Basic Layers, Custom Layers and Strongly Layers in the 4, 6 and 8 qubits configurations. The Random Layers ansatz configuration with 4 and 6 qubits achieved comparable but consistently resulted in higher errors than the other three ansatz. This result is expected given its low expressibility. A considerably higher error was obtained with the Random Layers and 8 qubits. In general, it can be observed in the standard deviation pannel that the 8 qubits configuration consistently achieved a more stable result.

The Euro dataset, trained with only 300 samples, is an example of a small dataset. This result is presented in figure 9. In the Euro dataset, the values of the metrics are more sensitive to the results of table 6. In this case, differently than in the example of a simple dataset, by observing the standard deviation panel the 8 qubits configuration is not the most stable case. It can be observed that metrics are better both in stability and overall result with 4 qubits. This result can be explained because even when the values of the expressivity increase with the number of qubits, this comes with a decrease in the divergence of the gradient of the cost function.

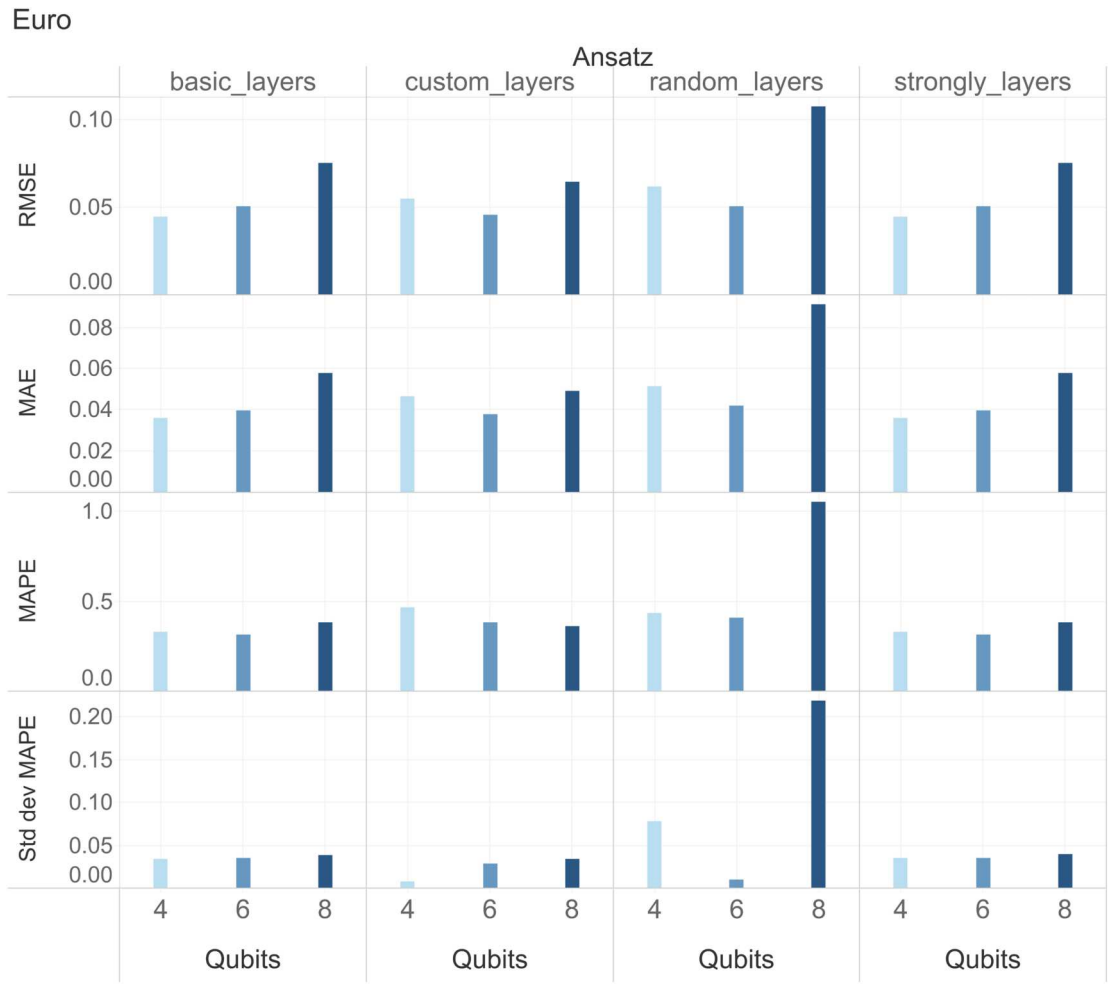


Figure 9. Comparison between 4, (light blue) 6, and 8 (dark blue) qubits for the Euro dataset. The values for RMSE, MAE, MAPE and the standard deviation of MAPE are presented.

The impact of the expressive power of an ansatz is better appreciated in a more complex dataset. In figures 10 and 11 the Strongly Layers ansatz consistently achieved better metrics, an expected result given its high expressivity. It can also be observed that results are considerably more stable for the 8 qubit configuration with Strongly Layers.

Across the three datasets, the Random Layers ansatz resulted in slightly less favorable metrics, as anticipated due to its limited expressivity and absence of non-zero Fourier coefficients. For the toy dataset, the results were comparable across ansatz, but a more stable result is achieved with 8 qubits. The small dataset proved to be more challenging to train using the most expressive ansatz. In the more complex datasets (SP500 and BTC) the expressivity of the Strongly Entangler with 8 qubits is key to obtain a good prediction.

To complete our analysis, figure 12 presents a comparison between our quantum model and its classical counterpart using the Legendre dataset. For both models, we report the average performance using 4, 6, and 8 output channels, with each configuration evaluated across 20 different random seeds. In the quantum case, the reported results are also averaged across all previously considered ansatz. The metrics obtained are generally comparable: the classical model achieves slightly better performance in terms of RMSE, MAE, and MAPE, while the QCNN shows a smaller standard deviation.

It is important to note that, since our study focused on a simplified architecture with only a single convolutional layer, the reported errors are not optimal. In a previous work by our research group (Rivera-Ruiz *et al* 2024), we performed a similar comparison between a 1D QCNN and its classical version, aiming to obtain the best possible metrics in both cases. That earlier study used more complex architectures, including three convolutional layers and two fully connected layers. In contrast, the present work prioritizes simplicity to better analyze the behavior of the quantum layer and understand the impact of quantum layer design. Therefore, the results shown in figures 9 to 12 are not the lowest achievable errors; better performance would be obtained by adding more layers, as done in our previous study.

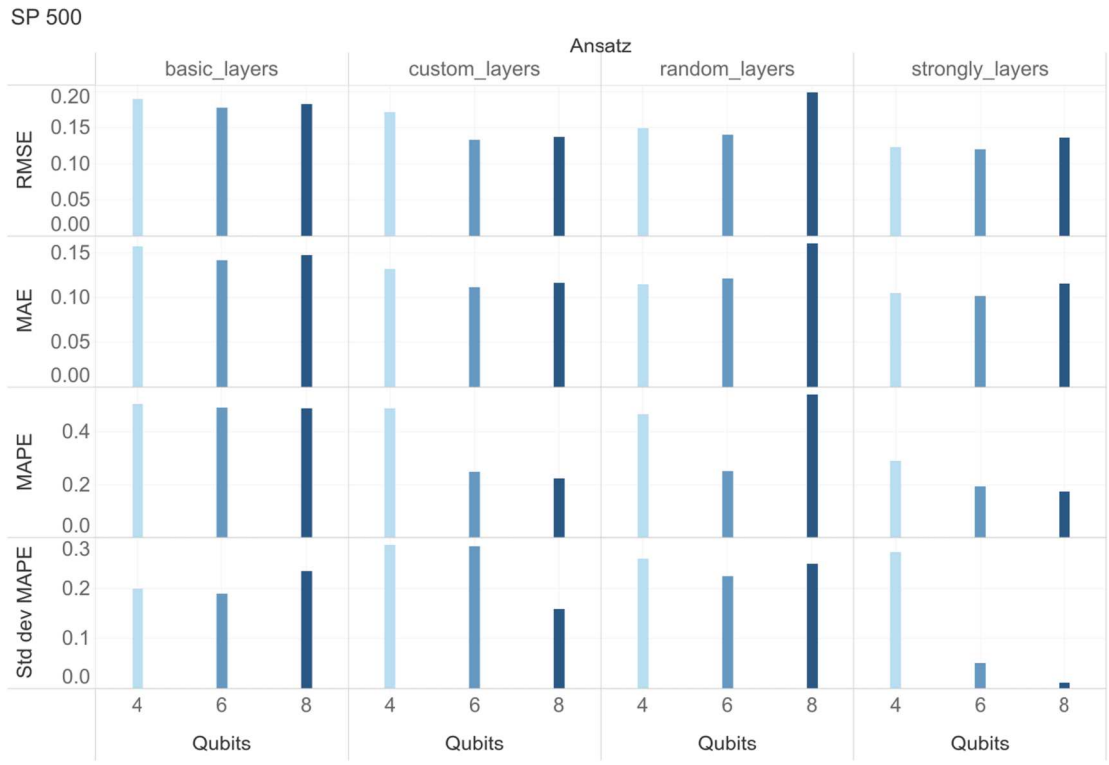


Figure 10. Comparison between ansatz for the SP500 dataset 4, (light blue) 6, and 8 (dark blue) qubits. The values for RMSE, MAE, MAPE, and the standard deviation of MAPE are presented. The 8 qubits configuration achieves a lower standar deviation.

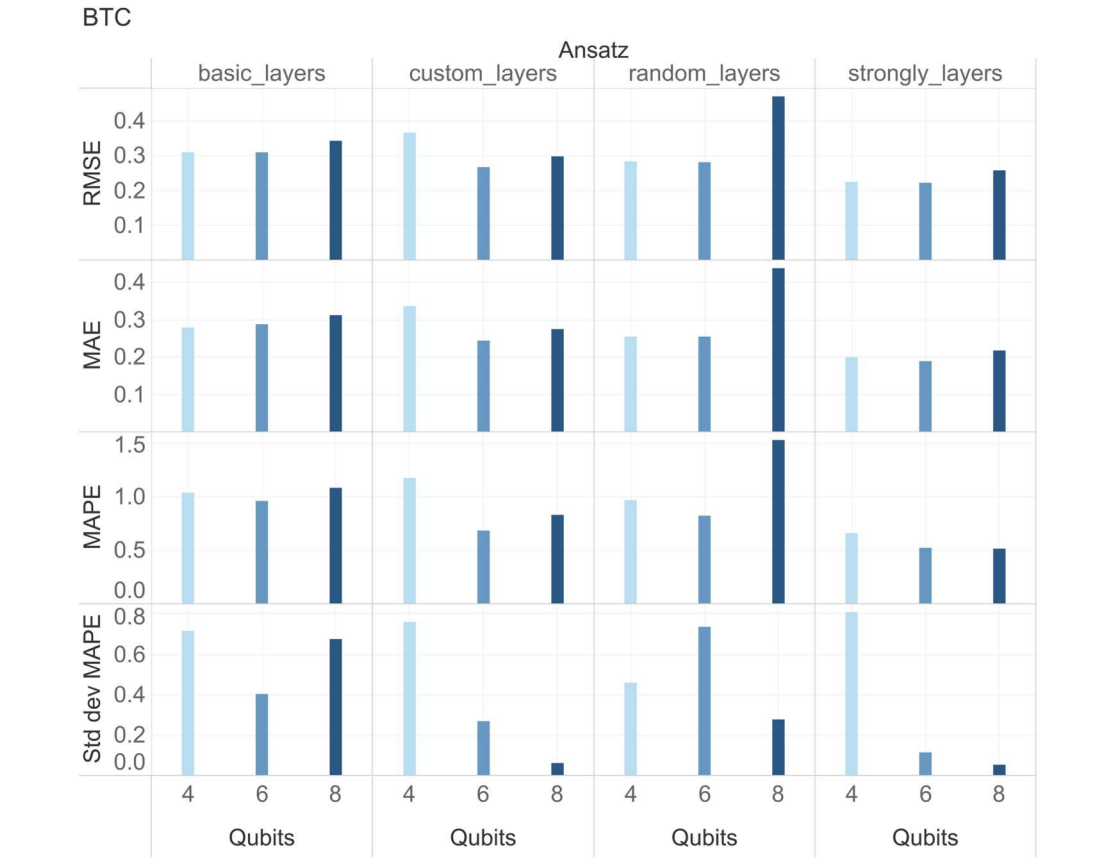


Figure 11. Comparison between ansatz for the BTC dataset 4, (light blue) 6, and 8 (dark blue) qubits. The values for RMSE, MAE, MAPE, and the standard deviation of MAPE are presented. The 8 qubits configuration achieves a lower standar deviation.

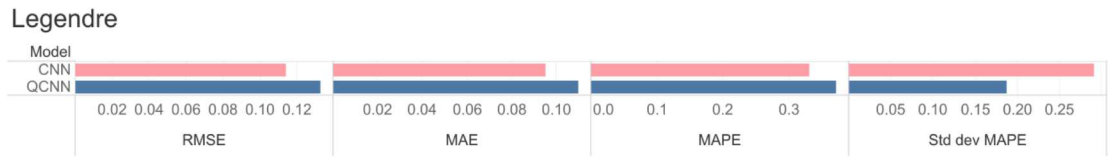


Figure 12. Comparison between the classical counterpart of our model (pink bars) and the 1D QCNN (blue bars). In each case, the results are averaged over implementations using 4, 6, and 8 output channels. For the quantum model, the average includes all results obtained with the four previously analyzed ansatz. The metrics obtained are comparable for both models.

Table 7. Performance Comparison of Forecasting Models on the SP500 Dataset. Metrics are reported on the original, non-normalized scale. Lower values are better for error metrics (RMSE, MAE, MAPE), while a higher value is better for Pearson Correlation.

Model	RMSE	MAE	MAPE (%)
Results from Literature			
ARIMA-GARCH Mustapa and Ismail (2019)	120.00	89.00	80.0
LSTM Pilla and Mekonen (2025)	207.00	175.00	—
xLSTM (with denoising)* Gil *et al* (2024)	29.41	22.93	—
Our Results			
Classical Counterpart (4 filters)	158.00	132.00	5.8
Proposed QCNN (4 qubits)	108.36	83.26	2.5
Classical Counterpart (8 filters)	69.47	54.76	1.3
Proposed QCNN (8 qubits)	**63.55**	**51.29**	**1.3**

4.6. Comparative analysis

To contextualize the performance of our model and assess its robustness, we present a comparative analysis against results available in the literature for the SP500 dataset. It is important to note that while our primary results presented in figure 10 were reported on a normalized scale, a common practice to unify metrics across different datasets, a direct comparison with other studies requires presenting our results on the original, non-normalized scale.

The literature provides a range of benchmarks for this task. For instance, hybrid ARIMA-GARCH models have been used to forecast SP500 prices, achieving a reported best result of an RMSE of 120 and an MAE of 89 on 2018 data (Mustapa and Ismail 2019). Other studies comparing ARIMA and LSTM models on multivariable daily data from 2013 to 2024 found that a univariate LSTM architecture yielded the best performance, with an RMSE of 207 and an MAE of 175 (Pilla and Mekonen 2025). More recently, a comprehensive study evaluated several advanced deep learning models, including a proposed xLSTM, on daily data from 2000 to 2020. After applying a denoising filter to the data, their best model achieved a state-of-the-art RMSE of 29.41 and an MAE of 22.93 Gil *et al* (2024).

The simplified architecture used in our main experiments, and whose results are reported in figures 9 to 11, was intentionally designed to isolate and analyze the properties of different quantum ansatz and qubit counts. To create a more competitive model for this benchmark, we enhanced our architecture by incorporating an additional classical convolutional layer. Based on our findings in figure 10, we selected the best performing configuration: 8 qubits with the Strongly Entangling ansatz. The resulting hybrid architecture is as follows:

$$\text{QCNN} \rightarrow \text{ReLU} \rightarrow \text{CNN} \rightarrow \text{ReLU} \rightarrow \text{Max Pooling} \rightarrow \text{Dense}$$

The results of our enhanced models are summarized in table 7. Our best performance was achieved with the **8-qubit QCNN architecture**, which yielded an **RMSE of 63.55**, an **MAE of 51.29**, and a **MAPE of 1.3%**. This comparative analysis leads to several key conclusions.

First, our hybrid quantum–classical models consistently outperform their classical counterparts at both evaluated scales. The 8-qubit QCNN (RMSE 63.55) is notably more accurate than the classical model with 8 filters (RMSE 69.47). Similarly, the 4-qubit QCNN (RMSE 108.36) surpasses its 4-filter classical counterpart (RMSE 158.00). This performance scaling is directly related to the **increased expressivity of the 8-qubit circuit**. As discussed in section 4.4, a greater number of qubits allows the model to generate a higher-degree Fourier series and explore the Hilbert space more comprehensively. This gives it the capacity to capture more complex patterns in the data, which translates into a substantial reduction in prediction error.

Regarding resource usage in simulation, the 8-qubit QCNN model used an average of **1070.8 MB of VRAM** (**GPU**) and **1708 MB of RAM**, whereas its classical 8-filter counterpart required **1195 MB of VRAM** and **1420 MB of RAM**. It is interesting to note that the quantum simulation was slightly more efficient in its use of GPU memory, though it demanded more RAM, likely due to the computational overhead of managing the state-vector simulation. It is crucial to emphasize that these figures reflect the cost of a *classical simulation* and do not represent the resources that a native quantum computer would require.

Finally, when compared to the literature, our 8-qubit QCNN model is highly competitive. It significantly surpasses traditional models such as ARIMA-GARCH (RMSE 120) and LSTM (RMSE 207). While it does not reach the state-of-the-art performance of the xLSTM model (RMSE 29.41), it is crucial to remember that their study applied a denoising pre-processing step, which largely explains its performance advantage. Therefore, our findings demonstrate that the hybrid QCNN is not only a viable but also a powerful architecture for financial time series forecasting, showing a clear advantage over equivalent classical structures and holding a competitive position against established models in the field.

5. Conclusion

In this study, we take the theoretical insights from Schuld *et al* (2021), Casas and Cervera-Lierta (2023), Sim *et al* (2019), Holmes *et al* (2022) to evaluate the performance of four different ansatz, considering practicality in training with real datasets. Numerous ansatz and qubit combinations exist and exploring all cases is beyond the scope of this work. Also, the results may vary with different datasets. Nevertheless, some key conclusions can be drawn from our findings.

Contrary to the condition emphasized in Casas and Cervera-Lierta (2023) that $N_p > \nu$ is necessary, our results suggest that a limited number of trainable parameters can be capable to produce Fourier functions of higher degrees. This finding demonstrates that quantum circuits can achieve highly effective experssivity even with a parameter count below the theoretical sufficiency bound. This observation is particularly relevant in terms of training times and also offers insight into why previous works achieved good results with few parameters. Further work needs to be done to provide more information about why few trainable parameters are capable of producing higher degree Fourier series than expected.

By analyzing the problem of quantum 1D convolution in the context of Fourier transforms, it was possible to design the architecture in a form in which data is reuploaded (*parallel* and *super parallel* architectures), which lead to significantly improved results. This improvement can be infered from the theory presented in Schuld *et al* (2021) and Casas and Cervera-Lierta (2023), where they also pointed time series as a natural application to their work. Employing a *super parallel* structure proves more effective than reuploading the data an equivalent number of times in a *parallel* structure. The significance of data reuploading is not fully captured by an expressivity analysis: it only becomes apparent when considering accessible coefficients, and the superiority of the *super parallel* approach cannot be inferred from the coefficients, but is noted when analyzing the expressivity. Then, it can be said that both parameters are complementary, this emphasizes the importance of a comprehensive analysis to obtain an optimal ansatz.

Regarding specific ansatz performances, Strongly Entangler, Custom Entangler, and Basic Entangler consistently gave favorable results, with a tendency to obtain a more stable result. The expressivity of Strongly Entangling also comes with the price of having a flat optimization landscape when increasing the number of qubits. When analyzing the testing metrics, it can be seen that this affected the Euro dataset, which had fewer training points. In the cases of complex datasets, the expressivity of the ansatz was more important. These findings contribute valuable insights for selecting effective ansatz configurations in quantum machine learning applications.

Beyond the empirical comparisons, our study highlights the importance of combining different theoretical perspectives when analyzing quantum neural architectures. While the richness of the accessible Fourier spectrum emerges as a direct predictor of forecasting performance, Haar-based expressivity measures provide complementary insights into the coverage of the Hilbert space, and barren plateau analysis explains optimization challenges in highly expressive ansatz. By synthesizing these distinct viewpoints, our results offer a more unified understanding of how circuit design principles affect practical performance, thus establishing a framework for selecting theoretically informed architectures in future QML applications

Acknowledgments

The authors acknowledge the support of Secretaria de Ciencia, Humanidades, Tecnologia e Innovacion (Secihti) . Also, the thechnical support and comments of JI Hernandez-Martinez

Conflict of interest/Competing interests

The authors declare no conflicts of interest. The authors have no relevant financial or non-financial interests to disclose. All authors certify that they have no affiliations with or involvement in any organization or entity with any financial interest or non-financial interest in the subject matter or materials discussed in this manuscript. The authors have no financial or proprietary interests in any material discussed in this article.

Ethics approval and consent to participate: Ethics approval Not applicable as this study did not involve any human. or animal subjects.

Data availability statement

The data that support the findings of this study are openly available at the following URL/DOI: https://github.com/SandraJuarez/1D-Quantum-Convolutional-Neural-Network.git.

Declarations

Competing Interests
The authors declare that they have no competing financial or personal interests that could have influenced the work reported in this paper.

Ethics Approval and Consent to Participate
This article does not contain any studies with human participants or animals performed by any of the authors.

Author contribution

S.L.J.O. designed and conducted the experiments, analyzed the obtained data and wrote the manuscript. M.A. R.R. designed the 1D quantum convolution and the corresponding code and proposed the datasets. A.M.V, E. R.T. and J.M.L.R. provided supervision, guidence and resources throughout the development of the project. Aditionally, M.A.R.R., A.M.V and E.R.T. provided critical insights that shaped the direction of the research. All authors reviewed the manuscript.

Funding

Open Access funding enabled by CINVESTAV. This study is supported by the Consejo Nacional de Humanidades, Ciencias y Tecnologias, CONAHCYT

Consent for publication

All authors have approved the final manuscript and consented to its publication.

Author contributions

Sandra Leticia Juárez-Osorio 0009-0003-5525-1309
Conceptualization (lead), Data curation (lead), Investigation (lead), Methodology (lead), Software (lead), Validation (lead), Visualization (lead), Writing – original draft (lead)

Andres Mendez-Vazquez 0000-0001-7121-8195
Funding acquisition (lead), Project administration (lead), Resources (lead), Writing – review & editing (equal)

Eduardo Rodriguez-Tello 0000-0002-0333-0633
Conceptualization (equal), Funding acquisition (equal), Resources (equal), Writing – review & editing (equal)

José Mauricio López-Romero
Resources (equal)

References

Antweiler W 2023 Pacific Exchange Rate Service. Available from: http://fx.sauder.ubc.ca/data.html

Bergholm V *et al* 2018 PennyLane: Automatic differentiation of hybrid quantum-classical computations *arXiv* https://doi.org/10.48550/arXiv.1811.04968 2022arXiv:181104968

Casas B and Cervera-Lierta A 2023 Multidimensional Fourier series with quantum circuits *Phys. Rev A* **107** 062612

Cerezo M *et al* 2021 Variational quantum algorithms *Nature Reviews Physics* **3** 625–44

Chen Y, Pan Y and Dong D 2022 Residual tensor train: a quantum-inspired approach for learning multiple multilinear correlations *IEEE Trans Neural Netw. Learn. Syst.* **33** 4573–87

De Luca G 2021 Survey of NISQ era hybrid quantum-classical machine learning research *Journal of Artificial Intelligence and Technology* **2** 9–15

Di Matteo O 2021 Understanding the Haar Measure. Xanadu. Accessed: 2024-02-23. Available from: https://pennylane.ai/qml/demos/tutorial_haar_measure

Feynman R P 2018 Simulating physics with computers *Feynman and Computation* (CRC Press) 133–53

Gil G L, Duhamel-Sebline P and McCarren A 2024 An evaluation of deep learning models for stock market trend prediction arXiv:2408.12408 https://doi.org/10.48550/arXiv.2408.12408

Glorot X and Bengio Y 2010 Understanding the difficulty of training deep feedforward neural networks *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics (AISTATS). vol. 9. JMLR Workshop and Conference Proceedings* pp 249–56 https://proceedings.mlr.press/v9/glorot10a.html

Gong L H, Chen Y Q, Zhou S and Zeng Q W 2025 Dual discriminators quantum generative adversarial network based on quantum convolutional neural network *Advanced Quantum Technologies* (https://doi.org/10.1002/qute.202500224)

Henderson M, Shakya S, Pradhan S and Cook T 2020 Quanvolutional neural networks: powering image recognition with quantum circuits *Quantum Machine Intelligence* **2** 2

Holmes Z, Sharma K, Cerezo M and Coles P J 2022 Connecting ansatz expressibility to gradient magnitudes and barren plateaus *PRX Quantum* **3** 010313

Hong Z, Wang J, Qu X, Zhu X, Liu J and Xiao J 2021 Quantum convolutional neural network on protein distance prediction *International Joint Conference on Neural Networks (IJCNN)* (IEEE) 1–8

Houssein E H, Abohashima Z, Elhoseny M and Mohamed W M 2022 Hybrid quantum-classical convolutional neural network model for COVID-19 prediction using chest X-ray images *Journal of Computational Design and Engineering* **9** 343–63

Huggins W, Patil P, Mitchell B, Whaley K B and Stoudenmire E M 2019 Towards quantum machine learning with tensor networks *Quantum Science and Technology* **4** 024001

Hur T, Kim L and Park D K 2022 Quantum convolutional neural network for classical data classification *Quantum Machine Intelligence* **4** 3

Kingma D P and Ba J 2014 Adam: a method for stochastic Optimization *arXiv* https://doi.org/10.48550/arXiv.1412.6980 2017arXiv:1412698010.48550/arXiv.1412.6980

Li W and Deng D L 2021 Recent advances for quantum classifiers *Science China Physics, Mechanics and Astronomy* **65** 220301

Li Z, Chen Y, Wang Y, Wu Z, Zhang J and Zhao Q 2025 Image denoising with hybrid classical–quantum convolutional neural network (QDnCNN) *Complex & Intelligent Systems* (https://doi.org/10.1007/s12293-025-00463-5)

Mari A, Bromley T R, Izaac J, Schuld M and Killoran N 2020 Transfer learning in hybrid classical-quantum neural networks *Quantum* **4** 340

Mustapa F H and Ismail M T 2019 Modelling and forecasting SP500 stock prices using hybrid Arima-Garch model *J. Phys. Conf. Ser.* **1366** 012130

Nielsen M A and Chuang I L 2000 *Quantum Computation and Quantum Information* (Cambridge University Press)

Park G, Huh J and Park D K 2023 Variational quantum one-class classifier *Machine Learning: Science and Technology* **4** 015006

Pei X, Zhou Y, Liu Y, Zhang W, Chen Q and Wang L 2025 One-to-many image generation model based on parameterized quantum circuits *Digit. Signal Process.* **146** 104550

Pilla P R and Mekonen R 2025 Forecasting SP500 using LSTM models *ZENODO*

Preskill J 2023 Quantum computing 40 years later *arXiv* 2023https://doi.org/10.48550/arXiv.2106.10522

Rivera-Ruiz M A, Mendez-Vazquez A and MauricioLópez-Romero J M 2022 Time series forecasting with quantum machine learning arquitectures *Advances in Computational Intelligence* Lecture Notes in Computer Science **13612** Springer, Cham 66–82 LagunasO PSB MMartínez-MirandaJ

Rivera-Ruiz M A, Juárez-Osorio S L, Mendez-Vazquez A, López-Romero J M and Rodriguez-Tello E 2024 1D quantum convolutional neural network for time series forecasting and classification *Advances in Computational Intelligence* (Springer Nature Switzerland) 17–35

Roh E, Baek H, Kim D and Kim J 2024 Fast quantum convolutional neural networks for low-complexity object detection in autonomous driving applications *IEEE Trans. Mob. Comput.* **24** 1031–1042

Sameer M and Gupta B 2022 A novel hybrid classical-quantum network to detect epileptic seizures medRxivhttps://doi.org/10.1101/2022.05.18.22275295

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

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

Schuld M, Sweke R and Meyer J J 2021 Effect of data encoding on the expressive power of variational quantum-machine-learning models *Phys. Rev. A* **103** 032430

Shahwar T *et al* 2022 Automated detection of Alzheimer's via hybrid classical quantum neural networks *Electronics* **11** 721

Sim S, Johnson P D and Aspuru-Guzik A 2019 Expressibility and entangling capability of parameterized quantum circuits for hybrid quantum-classical algorithms *Advanced Quantum Technologies* **2** 1900070

Stoudenmire E and Schwab D J 2016 Supervised learning with tensor networks *Advances in Neural Information Processing Systems* **29** 4799–807 https://proceedings.neurips.cc/paper_files/paper/2016/file/5314b9674c86e3f9d1ba25ef9bb32895-Paper.pdf

Wmcginn 2021 SP500 csv. Accessed: 2025-03-28. https://www.kaggle.com/datasets/wmcginn/sp500-csv

Yang Y F and Sun M 2022 Semiconductor defect detection by hybrid classical-quantum deep learning *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE) (https://doi.org/10.1109/CVPR52688.2022.00236.)

Zhang Z, Li J, Chen H and Wang L 2025 Quantum generative adversarial network based on the quantum Born machine *Adv. Eng. Inf.* (https://doi.org/10.1016/j.aei.2025.102559)

Zieliński M 2017 Bitcoin Historical Data. Accessed: 2025-03-28. Kaggle