mathematics

MDPI

*Article*

# Quantum Machine Learning for Credit Scoring

**Nikolaos Schetakis [1,2,3,*], Davit Aghamalyan [4], Michael Boguslavsky [5], Agnieszka Rees [5], Marc Rakotomalala [6] and Paul Robert Griffin [3]**

[1] Computational Mechanics and Optimization Laboratory, School of Production Engineering and Management, Technical University of Crete, 73100 Chania, Greece
[2] Quantum Innovation Pc, 73100 Chania, Greece
[3] QUBITECH, Quantum Technologies, 15231 Athens, Greece; paulgriffin@smu.edu.sg
[4] School of Computing and Information Systems, Singapore Management University, 81 Victoria Street, Singapore 188065, Singapore; davagham@gmail.com
[5] Tradeteq Ltd., London EC2M 4YP, UK; mboguslavsky@tradeteq.com (M.B.); arees@tradeteq.com (A.R.)
[6] Sim Kee Boon Institute for Financial Economics, Singapore Management University, 50 Stamford Road, Singapore 178899, Singapore; mrakoto@smu.edu.sg
* Correspondence: nsx@quinn.gr

**Abstract:** This study investigates the integration of quantum circuits with classical neural networks for enhancing credit scoring for small- and medium-sized enterprises (SMEs). We introduce a hybrid quantum–classical model, focusing on the synergy between quantum and classical rather than comparing the performance of separate quantum and classical models. Our model incorporates a quantum layer into a traditional neural network, achieving notable reductions in training time. We apply this innovative framework to a binary classification task with a proprietary real-world classical credit default dataset for SMEs in Singapore. The results indicate that our hybrid model achieves efficient training, requiring significantly fewer epochs (350) compared to its classical counterpart (3500) for a similar predictive accuracy. However, we observed a decrease in performance when expanding the model beyond 12 qubits or when adding additional quantum classifier blocks. This paper also considers practical challenges faced when deploying such models on quantum simulators and actual quantum computers. Overall, our quantum–classical hybrid model for credit scoring reveals its potential in industry, despite encountering certain scalability limitations and practical challenges.

**Keywords:** quantum machine learning; quantum classifiers; quantum credit scoring; quantum algorithms

**MSC:** 68Q12; 81P68; 68Q09; 68T01; 91G40; 68T07

check for updates

**Citation:** Schetakis, N.; Aghamalyan, D.; Boguslavsky, M.; Rees, A.; Rakotomalala, M.; Griffin, P.R. Quantum Machine Learning for Credit Scoring. *Mathematics* **2024**, *12*, 1391. https://doi.org/10.3390/math12091391

Academic Editors: Fernando L. Pelayo, Mauro Mezzini, Pedro Valero-Lara and Jonathan Blackledge

Received: 30 January 2024
Revised: 22 April 2024
Accepted: 29 April 2024
Published: 2 May 2024

## 1. Introduction

The quantum revolution, powered by quantum computing, is poised to transform various scientific and industrial fields, including finance [1–4]. In practice, many real-world problems are impossible to solve on a classical computer because of their resource-intensive nature. Quantum algorithms such as quantum search and quantum Fourier transforms promise significant speed advantages over classical algorithms [5,6], sparking a decade-long search for practical applications to harness this potential. Despite the challenge of finding real-world problems where quantum computing offers a practical advantage [7], the financial sector emerges as a promising arena. Currently, we navigate the noisy-intermediate scale quantum (NISQ) era, grappling with the scalability of qubits to achieve fault tolerance [8,9]. The exploration of quantum physics in finance, dubbed "quantum finance", reveals its applicability in areas such as option pricing, portfolio optimization, and risk analysis [10–19]. Quantum finance leverages quantum computing and quantum machine learning (QML) to address complex financial computations, from risk management to market trend analysis, highlighting the field's potential to unlock the computational power of quantum technologies for practical financial solutions.

In this study, we delve into the realm of credit scoring, a crucial financial task with significant economic implications [20,21]. Credit scoring facilitates economic expansion by enabling firms to access capital at favorable rates. However, obstacles such as information asymmetries can hinder the flow of capital to deserving projects. While large entities benefit from public credit ratings, smaller firms often lack such assessments, facing challenges in securing credit. Financial institutions have evolved from relying on qualitative assessments to employing sophisticated quantitative methods for managing credit risk, particularly for products with inherent default risks.

Enhancing machine learning (ML) models [22,23] for credit scoring, particularly for small and SMEs, is an active field of research. The current models, while advanced, exhibit limitations in accurately predicting insolvency, rejecting a significant portion of solvent companies due to error rates. Currently, the best credit scoring models reject 90% of companies that do become insolvent but still reject 15% of companies that remain healthy [Source: Tradeteq Ltd., London, UK]. Even modest improvements in these models could substantially reduce erroneous rejections and mitigate risks for lenders. At the intersection of ML and quantum computing, QML [24–26] applied to financial analysis has the potential for significant advancements by leveraging quantum physics' principles to accelerate data analysis. This approach not only aims to refine predictive accuracy in financial contexts but also explores the theoretical limits and practical applications of quantum-enhanced data analysis, offering a new frontier in financial technology innovation.

To foster breakthroughs in QML for finance, it is essential to explore heuristic algorithms that, despite currently lacking formal theoretical backing, excel in certain problem areas through cross-disciplinary insights and domain expertise. This paper benchmarks our heuristic approach, FULL HYBRID classical–quantum neural networks, against a pure classical ML model in credit scoring, highlighting the potential of QML. Despite SMEs often lacking formal credit ratings, our methodology, inspired by early quantitative models and enhanced using ML and QML, aims to improve credit risk assessments. By integrating advanced QML architectures and comparing them to various classical models [27], we demonstrate a notable learning efficiency, especially in noisy datasets.

The first quantitative calculations of credit scores were performed by Altman in the late 1960s; his Z-score model estimates a linear combination of financial ratios and uses the statistical method of discriminant analysis to predict publicly traded company defaults within two years in the manufacturing sector [20]. More recently, machine learning approaches have allowed for automated credit scoring for a broader coverage of attributes of small companies that combine company, accounting, and socio-economic information. Improving machine learning algorithms is thus an important element to providing credit risk transparency. Optimal feature selection for credit scoring datasets has been suggested in ref. [28]. This approach is based on an unconstrained binary optimization (QUBO) model. Comparisons with such well-established methods as recursive feature elimination showed that QUBO feature selection resulted in a smaller feature subset with no loss of accuracy. It is interesting to note that some researchers have considered quantum-inspired algorithms, such as a the quantum-inspired neuro evolutionary algorithm [29] with binary-real representation, that perform equally well compared to the other classical ML methods. A further discussion of related work can be found in our previous paper [27].

In this paper, we apply our previously developed quantum–classical models [27] to the benchmark problem of credit scoring. In our previous work [27], by combining such approaches as data re-uploading [30,31], and hybrid neural networks, parametric circuits [32,33], we created QML architectures and utilized them for the classification of non-convex two- and three-dimensional figures. The extensive benchmarking of the new FULL HYBRID classifiers against existing quantum and classical classifier models revealed that our novel models exhibit better learning characteristics to asymmetrical Gaussian noise in the dataset compared to known quantum classifiers and perform equally well for existing classical classifiers, with a slight improvement over classical results in the region of the high noise. We now apply the best of those models to real-world datasets.

This paper is organized as follows. In Section 2, we explain the theory behind the classical models used for credit scoring, describe the quantum models that have been previously used and developed by the authors, and make some remarks on benchmarking. Moreover in Section 2, we describe the dataset and the experimental setupwhere we give more details regarding the structure of our quantum neural network (QNN), outlining the hyperparameters, number of epochs, learning rate, and so on, which were executed using the Pennylane emulator. We present the results in Section 3, where we also summarize the benchmarking against the classical ML model and also share data on the running time for our FULL HYBRID architecture. The final Section 4 is devoted to a summary, challenges and future directions.

## 2. Materials and Methods

In this section, we review the current classical and quantum models used for credit scoring and set the groundwork for our own novel hybrid classical/quantum model. Firstly, to assess and benchmark the models in credit scoring, the crucial trade-off is between correctly assessing credit defaults or not. This trade-off can be summarized by a ROC (receiver operating characteristics) curve (Figure 1), where a false positive is rejecting a company for a loan that it would pay back, and true positive is rejecting a loan where the company would have defaulted. Ideally, a perfect model would not reject a good company (0.0 on the $x$-axis on the graph in Figure 1) and would reject all companies that would default (1.0 on the $y$-axis on the graph in Figure 1).

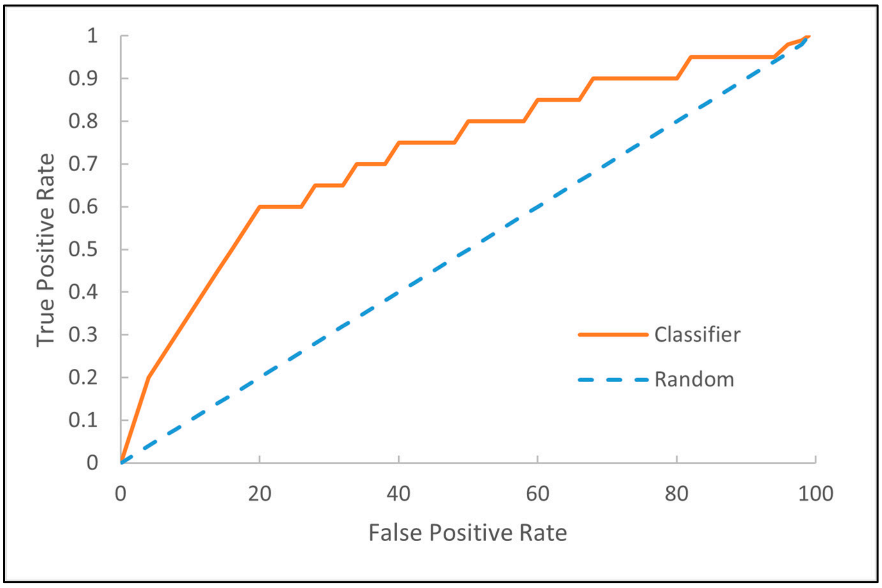

**Figure 1.** The ROC curves of three models used for credit scoring.

The area under the ROC curve (AUC) is calculated by summing the area under each ROC curve in Figure 1 and normalizing the areas to 1.0. The AUC varies between 0, a classifier always giving the wrong answer, and 1, a perfect classifier performance, with 0.5 for a uninformative classifier. This is a useful criterion to compare models and is the performance metric used throughout this paper.

### *2.1. Classical Models*

Here, we briefly outline classical ML attempts towards credit scoring, where we focus on the methods used and on the accuracy of prediction of a company defaulting on a loan. As a machine learning problem, credit scoring is typically formulated as a binary classification problem with highly imbalanced data (most companies do not default).

The data used have a number of features for a set of companies observed at a specific time, T0, and an outcome observed at a later time, T1. The outcome is usually treated as binary highlighting, class "1" if a company had some kind of adverse event between T0 and T1, such as companies in bankruptcy proceedings or in administration, and class "0" otherwise. As most companies do not suffer adverse events in most periods, the data are strongly imbalanced to class "0". A review of all the studies on machine learning for credit scoring is a subject large enough for a standalone paper. We refer the reader to systematic academic reviews [34]. The first models used relatively small company sets (hundreds of companies) and just a few manually constructed accounting features; thus, linear techniques such as linear discriminant analysis and support vector machines were used. Later, researchers obtained larger datasets with more companies and features and applied linear regressions, decision trees, fuzzy logic, ensemble models, and neural networks to the problem. With large enough datasets, modern ensemble techniques such as boosted trees and neural networks perform broadly on par. However, boosted trees are often preferred in practice due to their better explainability and stability. Choosing a classification model for credit scoring is a challenging task, and conflicts often arise when comparing performance. For example, linear discriminant models for predicting bad loans are found to perform better than neural networks for some data and opposite for others. These different outcomes are difficult to assess, but possible explanations include differences in sample sizes, transformation functions applied to the data, model parameters, or network topology. Also, the chosen performance metric matters; traditional statistical methods seem to perform as well as neural networks if one considers the total percentage of correct identification, but if identifying bad loans is the main goal, then neural networks have been found to perform better [35]. In this paper, we only consider a classical counterpart to the quantum model to ensure that a fair comparison is made to identify the behavior and potential advantages that quantum computing can bring. The performance metric throughout this paper is the AUC (see benchmarking section below), and this classical counterpart benchmark model achieves a score of 0.73 (see Section 3).

### 2.2. Quantum Models

In the book chapter co-authored by M. Boguslavsky, P. Griffin et al. [36], the authors introduce a new framework for addressing business problems with quantum computing, assessing classes of problems that could benefit, and showing a use case for QML algorithms. The authors outline two frameworks for quantum neural networks: (i) a 2-qubit perceptron inspired by the Entropica Labs algorithm for the classification of cancerous cells, and (ii) a hybrid neural networks where it is suggested to establish an interface between classical and quantum neural networks using PYTORCH (v2.3.0) and Qiskit (v1.0.2) [37]. In finance, there are extensive overviews/reviews for quantum computing and QML applied to finance [1–4]. In all of these overviews, credit scoring is mentioned as a problem which the current community is targeting to solve by making use of QML algorithms.

Our own novel architecture used for the experiments in this paper is a FULL HYBRID (FH) quantum neural network model consisting of three different approaches based on hybrid neural networks [37,38], variational circuits (VCs) [32,33], and data-reuploading classifiers (DRCs) [30,31]. First, classical data are encoded into quantum states. Angle embedding is used to load the data [x1, x2] into a qubit. Starting from an initial state vector, typically $|0\rangle$, a unitary operation U(x1, x2, 0) is applied, and a new quantum state is formed that can be described by a new point on a Bloch sphere. Padding with 0 is required when dealing with two or more dimensional data; for example, loading higher-dimensional data [x1, x2, x3, x4, x5, x6] can be broken down into sets of three parameters: U(x1, x2, x3), U(x4, x5, x6). We use an Rx gate for angle embedding in our experiments. Hybrid neural network classical–quantum classifiers are formed by connecting a number of classical and quantum neural networks in series. This architecture takes advantage of the specific capabilities of both types of neural networks and benefits from being able to have the number of features in the initial classical layers exceed the number of qubits in the quantum layer instead of being limited to one qubit for each feature. To create our hybrid classical–quantum neural network, a hidden layer is implemented utilizing a variational quantum circuit (Figure 2). A VC is a quantum circuit (also called a "parameterized" circuit) consisting of the data-embedding layer followed by parameterized gates such as rotation gates and entangling layers (CNOT gates that entangle each qubit with its neighbor). The quantum properties such as the rotation angles for the quantum gates are trainable parameters. DRC is introduced by replicating the VC into more blocks. To combine a VC circuit with the DRC technique, we define a block (B) as a sequence of data embedding and entangling layers (L). By adding several blocks, we re-introduce the input data into the model in a way similar to a classical neural network (NN) re-uploading classical data several times, once per hidden neuron. Our novel approach is to use the DRC technique combined with a VC in a single model in the quantum part of the hybrid classifier. This novel combination is expected to provide greater robustness to our results after making observations on the classification of 2D and 3D synthetic non-convex datasets [27]. The advantage brought by the VC approach for binary classification is in increased robustness against noise; however, it struggles to capture complicated patterns in prediction grid diagrams. Conversely, the DRC approach has good abilities to capture complex grid structures, but it is more sensitive to noise in the data. Consequently, it is seen [27] that VC and DRC combined complement each other to produce better results. The power of the FH model lies in capturing the complex patterns in the data while exhibiting robustness to data noise. We previously demonstrated [27] that, for synthetic datasets, FH architectures: (i) outperform several previously known quantum classifiers, (ii) perform equally well compared to their classical counterparts, and (iii) have an improvement over their classical counterparts in regions of high noise in the dataset. For further theoretical details, please refer to our previous paper [27].

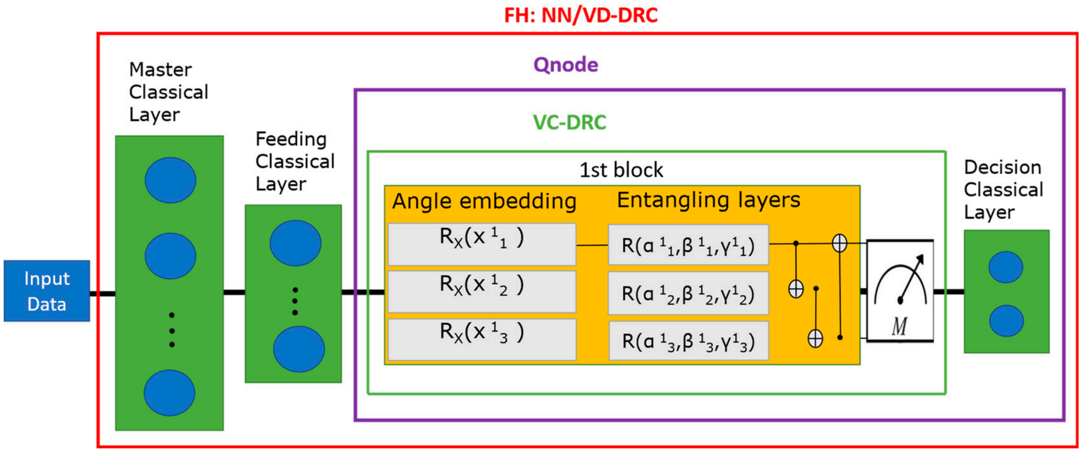

**Figure 2.** Block diagram of the full hybrid (NN/VC-DRC) classifier where a VC-DRC circuit is placed after a classical neural network.

After computation on the quantum node is completed, a measurement is performed. The measurement outcome is the expectation value of a Pauli observable for each qubit. The measurements are passed to the classical decision layer, which makes the final prediction label of the binary classifier. Another architecture explored for this problem area is to append the master classical layer to the quantum layer instead of having it first (i.e., FH:VC-DRC/NN). However, as the performance of the FH:NN/VC-DRC model was superior to that of the FH:VC-DRC/NN model in ref. [27], we only focus on the FH:NN/VC-DRC model. Going forward, we call this the FH model.

### *2.3. Data Description*

In this paper, we compare the performance of the FH model with its classical counterpart, applied to an actual credit default dataset for Singapore companies. The dataset originates from various Singapore institutions—the Accounting & Corporate Regulatory

Authority (ACRA), a statutory board under the Ministry of Finance of the Government of Singapore; the Singapore Land Authority, a statutory board under the Ministry of Law of the Government of Singapore; the Singapore Buildings database; Handshakes, a corporate data provider; and Tradeteq, a provider of data, technology, and software to the trade finance industry. The dataset covers nearly 2300 SME firms that were incorporated over the 1940–2016 period and active and healthy in 2016, with 94% of the firms incorporated since 1990 and distributed mainly across seven industry sectors. The biggest problem for SME datasets is that firms are privately held and that there is limited information about the financial situation of borrowers—only accounting data are available, and no information from rating agencies nor financial markets prices is available. These data limitations restrict the modelling choices of an SME portfolio to binary default or no-default models and panel data analysis rather than time series analysis. For each small business, the panel of 24 primary features includes the year of incorporation, accounting and operating information, geo-sociological data, and an indicator of whether the company defaulted or not. The firm is statutorily deemed "Healthy" (class 0) on 1 October 2016, and its status, in the case of default (e.g., compulsory winding down, receivership, or under judicial management), is statutorily changed to "Unhealthy" (class 1) over the following two years, between 1 October 2016 and 1 October 2018. The dataset size is limited by the number of class 1 examples, i.e., the companies that defaulted. We sampled all 246 class 1 companies in the period and added to them a random subsample of 2000 class 0 companies. Therefore, the dataset is highly imbalanced. For the experiments, due to the limitations of the computing hardware, the maximum number of features used is 21.

### 2.4. Experiments

In this section, we discuss the best-performing quantum and classical counterpart (CC) models found in the previous study [27], the hardware and software used, the hyperparameter configurations, model executions, and the processing of the experiment outputs. We used an FH:NN/VC-DRC model (Figure 2), where the first part is a classical neural network, followed by the VC-DRC circuit and a final decision layer, a single neuron layer with a sigmoid activation function. The first classical layer, «Master Classical layer» has 21 neurons, equal to the number of features used in our dataset. The last layer (Feeding classical layer) has the same number of neurons as the number of qubits in our hybrid model. Our 2D case, the classical NN, consists of a two-neuron layer with a rectified linear activation function (ReLU) (Master classical layer), followed by a two-neuron layer with a Leaky ReLU activation function (Feeding classical layer). Finally, a classical decision layer, «Decision Classical layer» makes the prediction. All the layers have a ReLU activation function except for the final decision layer, which has a sigmoid activation function. After each classical layer, we added a dropout layer to both the quantum and CC models, with 10% dropout rate to avoid overfitting. Before the data are ingested into the model, they are pre-processed using a standard pipeline along with some proprietary processing. Data encoding into quantum states used angle embedding (see Section 2.2).

As simulators were used throughout, there was no quantum computer noise, and no additional noise was introduced in these experiments. Later in this paper, we consider how noise may affect the results on real quantum computers. Pennylane (v0.35), an open-source software framework for differentiable programming of quantum computers, was used to build the models. The computer used for these experiments was a PC with 64 GB of RAM and an AMD Ryzen 7 processor. We tested the quantum model performance versus the number of qubits and number of blocks and against the CC model. The training dataset has 1798 rows of data split into a validation set of 179 rows (10%) and a test set of 269 rows (15%). Throughout the simulations, the same training, validation, and testing dataset is used. For every configuration, we used the average outcome of the ROC/AUC score of five simulations. The hyperparameters used are based on heuristics and results from previous experiments [27], are summarized in Table 1, and were kept consistent throughout the study except where explicitly mentioned.

**Table 1.** Hyperparameters used in this study.

| | | |
|---|---|---|
| Epochs | 350 | Number of complete passes through the training dataset |
| Dropout rate | 0.1 | Probability of training a given node in a layer, 1.0 = no dropout, 0.0 = no outputs from the layer |
| Learning rate | 0.001 | Step size at each iteration while moving toward a minimum of a loss function |
| Optimizer | SGD | Stochastic gradient descent |
| Batch size | 16 | Number of training samples to work through before the model's internal parameters are updated |

For a fair comparison, the number of epochs for quantum and classical models were kept identical at 350. This number was chosen after observing that, on average, after 350 epochs, overfitting occurs. However, the classical model improved if training was increased up to 5000 epochs. The number of DRC blocks was increased from 1 to 10, and the number of qubits was increased from 6 to 18. The batch size was reduced to 16 training examples per iteration due to memory restrictions. The optimizer we used was a stochastic gradient descent (SGD) optimizer. The loss function is binary cross-entropy, and the optimizing parameter is ROC/AUC, as described in Section 2.1. Our simulations were restricted to a maximum number of 18 qubits due to execution time constraints (see scaling results in Section 3 and the discussion in Section 4).

## 3. Results

In this section, we present the experimental results comparing the overall ROC/AUC of the FH model to the CC model, showing the behavior of the quantum model as we scale up the number of qubits and the number of processing blocks.

The quantum model achieved an ROC/AUC of 0.75 (Figure 3 (black dots) and Figure 4 (right column)), whereas the CC model achieved an ROC/AUC of 0.73 (Figure 3 (orange line)). However, letting the classical counterpart model train up to 5000 epochs can lead to the highest score of 0.75 (Figure 3 (red line)). This indicates that the FH model has the ability to achieve a higher score than its classical counterpart with fewer training epochs.

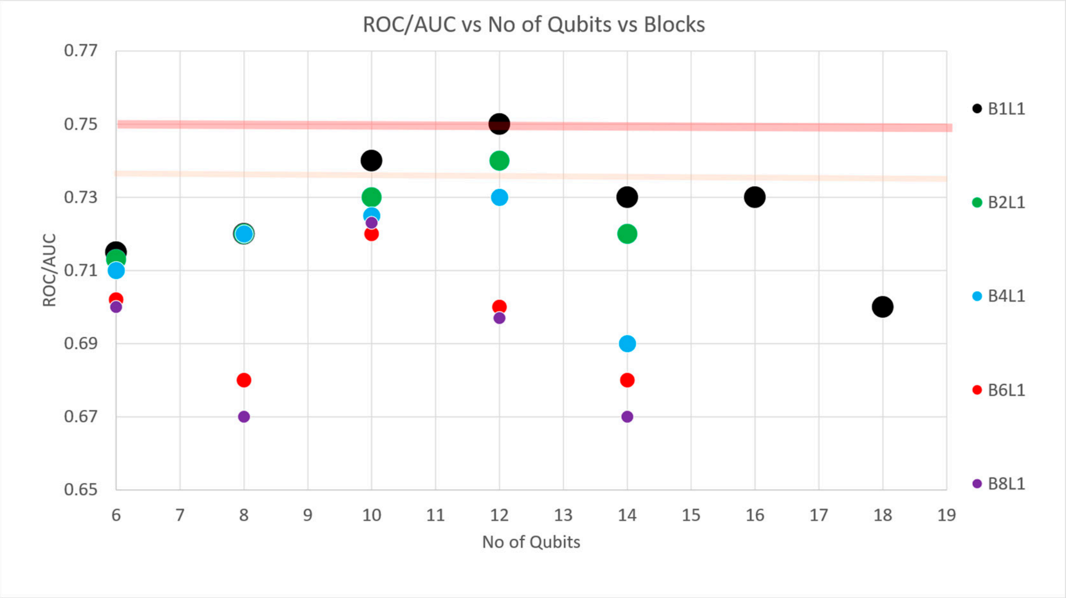

**Figure 3.** The FH model versus the number of qubits ($x$-axis) and number B of blocks (colored circles). The orange line denotes the classical counterpart model ROC/AUC when trained for a maximum of 350 epochs, while the red line denotes the classical counterpart model's ROC/AUC when trained for a maximum of 3500 epochs.

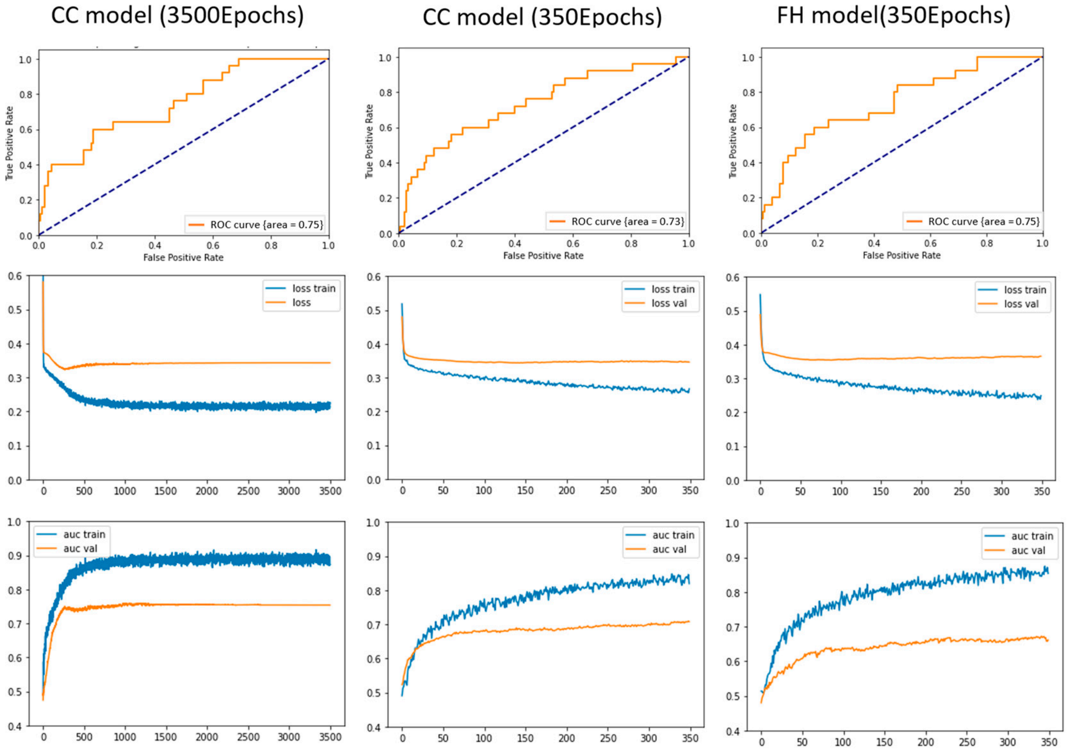

**Figure 4.** Relative results of both the CC model (left and middle columns) and the FH model (right column). The results for the CC model are shown for different max numbers of training epochs: 3500 epochs and 350 epochs. Note that the relative results shown here do not match the absolute results shown in Figure 3.

Furthermore, in Figure 3, we see how the ROC/AUC of the FH model's performance changes with the number of qubits and the number of blocks. For blocks = 1 (black dots), we observe that the ROC/AUC increases up to 12 qubits and then decreases after 12 qubits. The same behavior can also be observed when one increases the number of blocks, with the highest results being achieved with no data-reupload. See the discussion section below for possible explanations.

The training processes that produce the highest ROC/AUC scores on the testing dataset are depicted in Figure 4 for both the CC model (left and middle columns) and the FH model (right column). The top row depicts the final ROC/AUC, and the middle and bottom rows show the loss and ROC/AUC training process for the training and validation dataset, respectively. The FH model achieved the highest score, with fewer training epochs for all simulations, shown in Figure 4.

Measuring execution times on the simulator allows us to estimate the resource requirements for actual quantum computers. We observed that the data embedding time doubles with each additional qubit, and that the model execution time scales quadratically with the number of qubits and linearly with the number of blocks (Figure 5). These observations are of importance for the use of real quantum computers and are discussed in the section below. Whilst only 18 qubits were used for the main results described above, 20 qubits were used for these resource tests.

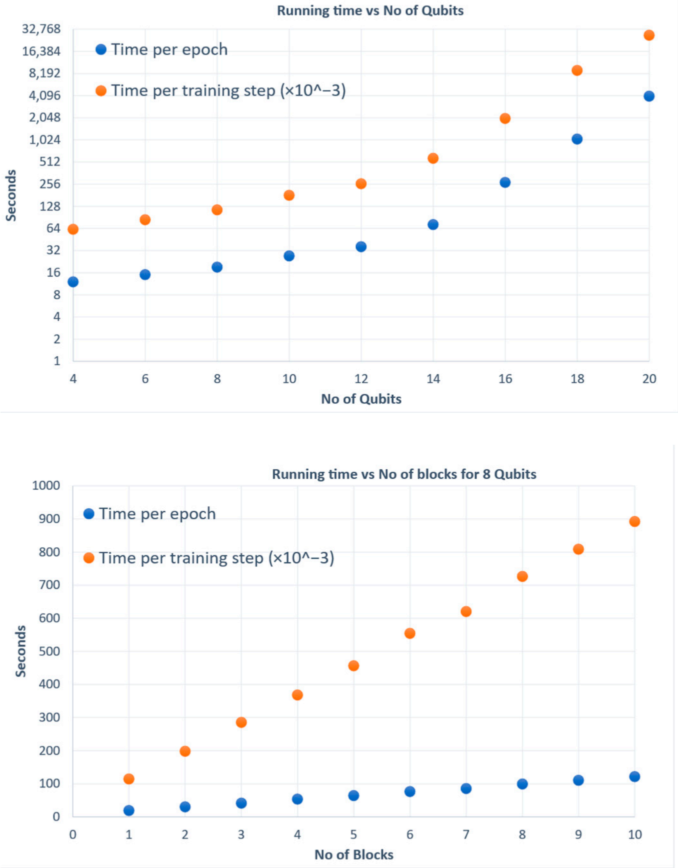

**Figure 5.** The execution time for the FH model on the simulator per epoch and per training step for an increasing number of total qubits (**top**) and for an increasing number of blocks with a constant number of 8 qubits (**bottom**).

## 4. Discussion

Firstly, we would like to highlight our novel approach in using a new architecture combining quantum variational circuits with data re-uploading in a hybrid (classical–quantum) neural network, FH [27] on actual credit data. We will now discuss the comparison of our FH and CC models as well as the execution times and limitations of the experiments. We found the accuracy of the FH model to be equal to the CC model. Also, the FH model required significantly fewer epochs to train than the CC model by a factor of 2 or more (Figure 5). Given that training time is a significant business driver, this result is very interesting and is grounds for optimism for practical quantum advantages in the future. Furthermore, it is also positive that the results are satisfactory, even when using a classical feeding layer to reduce the number of features fed into the quantum layer (limited by the number of qubits). Currently, the number of qubits is limited in the simulator by the memory requirements growing exponentially with the number of qubits. For real quantum computers, the physical number of qubits is limited by the available hardware and extra qubits needed for error correction. Consequently, this is the most practical FH architecture we can use for this specific dataset. In the future, increased computing power will enable us to benchmark with more qubits and produce more results.

During simulations, we encounter two critical challenges that could affect accuracy: the barren plateau phenomenon [39,40] and the overparameterization [41,42] problem within quantum circuits. The barren plateau phenomenon refers to the issue of vanishing gradients in the optimization landscape of quantum circuits, particularly prevalent in quantum circuits with many qubits, blocks, and layers. This phenomenon impedes the training process, as gradients become exponentially small, rendering standard optimization techniques ineffective and leading to a slow convergence or even complete stagnation. On the other hand, the overparameterization problem arises when quantum circuits have more parameters than necessary for a given task, leading to increased computational complexity, resource requirements, and susceptibility to noise and decoherence. Addressing these challenges is crucial for advancing quantum machine learning algorithms towards practical applications, necessitating innovative approaches in circuit design, optimization strategies, and noise mitigation techniques.

The unexpected finding that increasing the number of qubits beyond 12 reduced the accuracy is likely to be related to barren plateau issues but needs further investigation in future studies. The effect of adding more blocks, i.e., DRC, reducing the accuracy could be due to the overparameterization problem or due to our dataset not having features that have a trigonometric (i.e., sine) structure, thus re-entering the data to the model does not improve its performance. It is not due to overfitting, as the training is stopped before overfitting occurs. We note that finding the best learning gradients in a non-convex landscape (for the problem structure) is an open question even in the classical machine learning community, and no resolution has been discovered so far. To this end, one possible solution is to use quantum convolutional neural networks. Moreover, the use of a data re-uploading circuit could possibly overcome this problem based on the prima facie argument that since data is introduced many times to the network, the solution is forced from any local minima. Another promising method is to characterize the landscape by computing the Hessian of the loss function, where since the eigenvalues of the Hessian loss function directly quantify the local curvature of the loss function, we can adjust the learning rate of our model for faster convergence during the training process.

For simulators, we observe the exponential time complexity of increasing the number of qubits and blocks (Figure 5), both of which would not be a problem for a real quantum computer as long as the circuit width (number of qubits) and circuit depth (number of gates) are within the specifications of the quantum computer. However, another general issue for QML is the need to make many executions of the quantum circuit, for 350 epochs and a training set of 1798, this amounts to 629,300 circuit executions. This is not an issue for simulations run on a standalone computer, but for quantum computers accessed on a cloud platform, the overall time taken may not practical; for example, it may take 7 days in total if each execution takes 1 s due to network overheads and queuing time on a shared quantum computer. Quantum computer providers such as IBM allow for paid allotment of compute times and have very recently introduced mechanisms such as Qiskit runtime that enables the execution of the classical and quantum code to be run as one unit, reducing the overall execution time by 120 times, potentially bringing 7 days down to 1.5 h.

The dataset size is limited by the number of class 1 examples available. This has an impact on the most appropriate model but does not lead to overfitting. A much larger dataset such as the UK's, with 1000+ defaults each year and over 4 million companies, would be interesting to use with more complex models and finer-grained risk periods. We also note that the hyperparameters for the FH and CC models could potentially be tuned further.

## 5. Conclusions

In conclusion, the use of hybrid quantum–classical models is promising given the ease of obtaining comparable results to a purely classical counterpart and with much fewer epochs for training. We have also investigated the practical issues of using the models on simulators and on real quantum computers and expect that, with even modest improvements in hardware expected of over 4000 qubits by 2025 [43], along with software improvements such as runtime environments [44], real advantages, at least in model training, will be achieved. It is also possible that improvements in accuracy may also be observed due to the resilience of the FH model.

Furthermore, this study shows that anyone in the machine learning community can relatively easily experiment with QML for their own problems. The next step of moving to real quantum hardware may also prove interesting with the introduction of quantum noise, possibly removing the need for dropout layers. The future for QML is very exciting. The codes used in this manuscript will be made available upon reasonable request.

**Author Contributions:** Conceptualization, P.R.G.; methodology, M.B., A.R. and D.A. and N.S.; software, D.A., N.S. and A.R.; validation, M.B., A.R., and M.R., formal analysis, P.R.G. and M.R.; investigation, M.R.; resources, N.S. and D.A.; data curation, A.R.; writing—original draft preparation, D.A.; writing—review and editing, P.R.G., M.R. and D.A.; visualization, N.S.; supervision, P.R.G.; project administration, P.R.G.; funding acquisition, P.R.G. and N.S. All authors have read and agreed to the published version of the manuscript.

**Funding:** N.S. would like to acknowledge funding support from the European Union's Horizon 2020 research and innovation programme EΥE under the Marie Skłodowska-Curie grant agreement No. 101007638. This research project was supported by the Artificial Intelligence and Data Analytics (AIDA) scheme, which provides funding support to strengthen the AIDA ecosystem in the Singapore financial sector. The AIDA scheme is part of the Financial Sector Technology and Innovation (FSTI) scheme under the Financial Sector Development Fund administered by the Monetary Authority of Singapore (MAS). Any opinions, conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of MAS. D.A. would like to acknowledge the funding support from Agency for Science, Technology and Research (#21709).

**Data Availability Statement:** The datasets used and/or analyzed during the current study are available from the corresponding author upon reasonable request.

**Acknowledgments:** We would like to thank Kishor Bharthi for supporting our work from its early stages.

**Conflicts of Interest:** Nikolaos Schetakis was employed by Quantum Innovation Pc and by QUBITECH. Michael Boguslavsky and Agnieszka Rees were employed by Tradeteq Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest. The companies had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

## References

1. Orús, R.; Mugel, S.; Lizaso, E. Quantum computing for finance: Overview and prospects. *Rev. Phys.* **2019**, *4*, 100028. [CrossRef]
2. Herman, D.; Googin, C.; Liu, X.; Sun, Y.; Galda, A.; Safro, I.; Alexeev, Y. Quantum computing for finance. *Nat. Rev. Phys.* **2023**, *5*, 450–465. [CrossRef]
3. Egger, D.J.; Gambella, C.; Marecek, J.; McFaddin, S.; Mevissen, M.; Raymond, R.; Yndurain, E. Quantum computing for finance: State-of-the-art and future prospects. *IEEE Trans. Quantum Eng.* **2020**, *1*, 1–24. [CrossRef]
4. Bouland, A.; van Dam, W.; Joorati, H.; Kerenidis, I.; Prakash, A. Prospects and challenges of quantum finance. *arXiv* **2020**, arXiv:2011.06492.
5. Nielsen, M.A.; Chuang, I. *Quantum Computation and Quantum Information*; Cambridge University Press: Cambridge, UK, 2002.
6. Arute, F.; Arya, K.; Babbush, R.; Bacon, D.; Bardin, J.C.; Barends, R.; Martinis, J.M. Quantum supremacy using a programmable superconducting processor. *Nature* **2019**, *574*, 505–510. [CrossRef] [PubMed]
7. Zhong, H.S.; Wang, H.; Deng, Y.H.; Chen, M.C.; Peng, L.C.; Luo, Y.H.; Pan, J.W. Quantum computational advantage using photons. *Science* **2020**, *370*, 1460–1463. [CrossRef] [PubMed]
8. Preskill, J. Quantum computing in the NISQ era and beyond. *Quantum* **2018**, *2*, 79. [CrossRef]
9. Bharti, K.; Cervera-Lierta, A.; Kyaw, T.H.; Haug, T.; Alperin-Lea, S.; Anand, A.; Aspuru-Guzik, A. Noisy intermediate-scale quantum algorithms. *Rev. Mod. Phys.* **2022**, *94*, 015004. [CrossRef]
10. Seydel, R.; Seydel, R. *Tools for Computational Finance*; Springer: Berlin, Germany, 2006; Volume 3.
11. LeBaron, B. Agent-based computational finance: Suggested readings and early research. *J. Econ. Dyn. Control.* **2000**, *24*, 679–702. [CrossRef]
12. Ugur, O. *An Introduction to Computational Finance*; World Scientific Publishing Company: Singapore, 2008; Volume 1.
13. Doriguello, J.F.; Luongo, A.; Bao, J.; Rebentrost, P.; Santha, M. Quantum algorithm for stochastic optimal stopping problems with Applications in Finance. In Proceedings of the 17th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2022), Champaign, IL, USA, 11–15 July 2022; Leibniz International Proceedings in Informatics (LIPIcs). Leibniz International Proceedings in Informatics (LIPIcs): Wadern, Germany; Volume 232, pp. 2:1–2:24. [CrossRef]
14. Rebentrost, P.; Lloyd, S. Quantum computational finance: Quantum algorithm for portfolio optimization. *arXiv* **2018**, arXiv:1811.03975.
15. Lim, D.; Rebentrost, P. A Quantum Online Portfolio Optimization Algorithm. *arXiv* **2022**, arXiv:2208.14749. [CrossRef]

16. Ghashghaie, S.; Breymann, W.; Peinke, J.; Talkner, P.; Dodge, Y. Turbulent cascades in foreign exchange markets. *Nature* **1996**, *381*, 767–770. [CrossRef]
17. Dixon, M.F.; Halperin, I.; Bilokon, P. *Machine Learning in Finance*; Springer International Publishing: Berlin/Heidelberg, Germany, 2020; Volume 1170.
18. Doering, J.; Kizys, R.; Juan, A.A.; Fito, A.; Polat, O. Metaheuristics for rich portfolio optimisation and risk management: Current state and future trends. *Oper. Res. Perspect.* **2019**, *6*, 100121. [CrossRef]
19. Culkin, R.; Das, S.R. Machine learning in finance: The case of deep learning for option pricing. *J. Invest. Manag.* **2017**, *15*, 92–100.
20. Altman, E.I. Financial Ratios, Discriminant Analysis and the Prediction of Corporate Bankruptcy. *J. Financ.* **1968**, *23*, 589–609. [CrossRef]
21. Campbell, J.Y.; Hilscher, J.; Szilagyi, J. In Search of Distress Risk. *J. Financ.* **2008**, *63*, 2899–2939. [CrossRef]
22. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016.
23. Wiering, M.A.; Van Otterlo, M. Reinforcement learning. *Adapt. Learn. Optim.* **2012**, *12*, 729.
24. Cerezo, M.; Verdon, G.; Huang, H.Y.; Cincio, L.; Coles, P.J. Challenges and opportunities in quantum machine learning. *Nat. Comput. Sci.* **2022**, *2*, 567–576. [CrossRef] [PubMed]
25. Biamonte, J.; Wittek, P.; Pancotti, N.; Rebentrost, P.; Wiebe, N.; Lloyd, S. Quantum machine learning. *Nature* **2017**, *549*, 195–202. [CrossRef]
26. Schuld, M.; Sinayskiy, I.; Petruccione, F. An introduction to quantum machine learning. *Contemp. Phys.* **2015**, *56*, 172–185. [CrossRef]
27. Schetakis, N.; Aghamalyan, D.; Griffin, P.; Boguslavsky, M. Review of some existing QML frameworks and novel hybrid classical–quantum neural networks realising binary classification for the noisy datasets. *Sci. Rep.* **2022**, *12*, 1–12. [CrossRef] [PubMed]
28. Milne, A.; Rounds, M.; Goddard, P. *Optimal Feature Selection in Credit Scoring and Classification Using a Quantum Annealer*; White Paper 1Qbit; 1Qbit: Vancouver, BC, Canada, 2017.
29. de Pinho, A.G.; Vellasco, M.; da Cruz, A.V.A. A new model for credit approval problems: A quantum-inspired neuro-evolutionary algorithm with binary-real representation. In Proceedings of the 2009 World Congress on Nature & Biologically Inspired Computing (NaBIC), Coimbatore, India, 9–11 December 2009; pp. 445–450.
30. Pérez-Salinas, A.; Cervera-Lierta, A.; Gil-Fuster, E.; Latorre, J.I. Data re-uploading for a universal quantum classifier. *Quantum* **2020**, *4*, 226. [CrossRef]
31. Pérez-Salinas, A.; López-Núñez, D.; García-Sáez, A.; Forn-Díaz, P.; Latorre, J.I. One qubit as a Universal Approximant. *Phys. Rev. A* **2021**, *104*, 012405. [CrossRef]
32. Cerezo, M.; Arrasmith, A.; Babbush, R.; Benjamin, S.C.; Endo, S.; Fujii, K.; Coles, P.J. Variational quantum algorithms. *Nat. Rev. Phys.* **2021**, *3*, 625–644. [CrossRef]
33. Sim, S.; Johnson, P.D.; Aspuru-Guzik, A. Expressibility and entangling capability of parameterized quantum circuits for hybrid quantum-classical algorithms. *Adv. Quantum Technol.* **2019**, *2*, 1900070. [CrossRef]
34. Kumar, M.R.; Gunjan, V.K. Review of machine learning models for credit scoring analysis. *Ing. Solidar.* **2020**, *16*, 1–16.
35. Desai, V.S.; Crook, J.N.; Overstreet, G.A. A comparison of neural networks and linear scoring models in the credit union environment. *Eur. J. Oper. Res.* **1996**, *95*, 24–37. [CrossRef]
36. Griffin, P.R.; Boguslavsky, M.; Huang, J.; Kauffman, R.J.; Tan, B.R. Quantum Computing: Computational excellence for Society 5.0. In *Data Science and Innovations for Intelligent Systems Computational Excellence and Society 5.0*, 1st ed.; CRC Press: Boca Raton, FL, USA, 2022; p. 392.
37. Hybrid Quantum-Classical Neural Networks with Pytorch and Qiskit. Available online: https://qiskit-community.github.io/qiskit-machine-learning/tutorials/05_torch_connector.html (accessed on 30 April 2024).
38. Blazakis, K.; Katsigiannis, Y.; Schetakis, N.; Stavrakakis, G. One Day Ahead Wind Speed Forecasting based on Advanced Deep and Hybrid Quantum Machine Learning. In Proceedings of the Springer 1st International Conference on Frontiers of Artificial Intelligence, Ethics and Multidisciplinary Applications (FAIEMA 2023), Athens, Greece, 25–26 September 2023. [CrossRef]
39. McClean, J.R.; Boixo, S.; Smelyanskiy, V.N.; Babbush, R.; Neven, H. Barren plateaus in quantum neural network training landscapes. *Nat. Commun.* **2018**, *9*, 4812. [CrossRef]
40. Cerezo, M.; Sone, A.; Volkoff, T.; Cincio, L.; Coles, P.J. Cost-Function-Dependent Barren Plateaus in Shallow Quantum Neural Networks. *Phys. Rev. Lett.* **2021**, *127*, 030503.
41. Grant, E.; Seth, L. Efficient gradient ascent in quantum control space. *Phys. Rev. Lett.* **2010**, *105*, 150501.
42. Schuld, M.; Bergholm, V.; Gogolin, C.; Izaac, J.; Killoran, N. Evaluating analytic gradients on quantum hardware. *Phys. Rev. A* **2019**, *99*, 032331. [CrossRef]
43. Expanding the IBM Quantum Roadmap to Anticipate the Future of Quantum-Centric Supercomputing | IBM Research Blog. Available online: https://research.ibm.com/blog/ibm-quantum-roadmap-2025 (accessed on 20 May 2022).
44. Qiskit Runtime. IBM Quantum. Available online: https://docs.quantum.ibm.com/lab (accessed on 30 April 2024).

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.