Coupling quantum classification and quantum distance estimation in continual learning*

Corrado Loglisci, Donato Malerba

Department of Computer Science, Universita' degli studi di Bari

Abstract

Quantum machine learning recently gained prominence due to the promise of quantum computers in solving machine learning problems that are intractable on a classical computer. Nevertheless, several studies on problems which remain challenging for classical computing algorithms are emerging. One of these is classifying continuously incoming data instances according to the continual learning paradigm, which is studied in this paper through a hybrid computational solution that combines classical and quantum techniques. Hybrid approaches represents one of the current ways for the use of quantum computation in practical applications.

In this paper, we show how typical issues of continual learning can be equally addressed with the properties of quantum mechanics, until to offer often better results. We propose the combined use of quantum classification and quantum distance estimation to update the classification capabilities as new data instances are processed. Experiments are performed on real-world datasets with quantum simulators.

1. Introduction

Quantum machine learning has been introduced with the promise to handle machine learning problems that are intractable on a classical computer, especially those characterized by huge amounts of data. In the research on Quantum computing technologies, the current status sees the era of noisy intermediate scale quantum (NISQ) computers [1], which are devices able to deal with low-middle size data problems. An approach which seems bringing practical advantages is instead the one of *hybrid* frameworks [2][3] that combine classical and quantum methods and allow to exploit quantum physics properties while limiting the impact of the existing restrictions of the quantum devices.

One of the categories of data-intensive problems in which the research on classical computing dedicates still many efforts is learning of models from continuously incoming sequential data. Even the accurate solutions of Deep Learning find challenging working on that data scenario. This is demonstrated by the long list of recentest studies addressing the so-called *catastrophic forgetting* [4], which is the tendency of an artificial neural network to abruptly and drastically forget previously learned information upon learning new information. In those cases, it is not important designing algorithms for massive computation, but keeping the quality of the models high over unbounded sequences of data.

We investigate these points through a quantum-classical framework which builds a classification model in the supervised setting and works in continual learning [5, 6] by acquiring

XXX

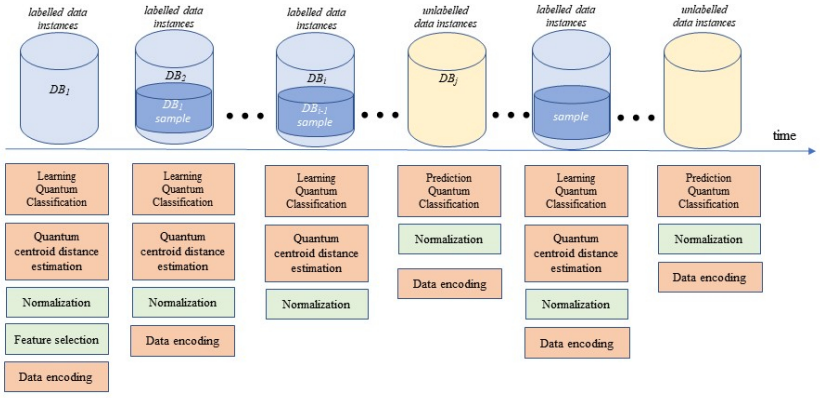

Figure 1: The components of the proposed hybrid framework as they run in continual learning setting for binary classification.

continuously incoming data instances. The framework adapts continually a classification model and keeps on learning over time. More precisely, it trains and updates a classifier on (sub-) sequences of incoming data instances (data blocks) marked as labelled. Then, the classifier is used to estimate the class-value of unlabelled incoming data instances. The update is performed only when the properties and distribution of the labelled data changes, which is what typically happens in continual learning. To detect such changes, we rely on a quantum centroid distance estimation technique, often used in quantum clustering. The centroids are considered by their characteristic of synthesizing clusters, which in this work are produced for each class-value on the labelled data. So, the changes are detected when the properties of the clusters change and this happens when the labelled data (assigned to the respective clusters) change. Consequently, the classification model needs to be updated on those data.

The framework has been tested on the binary classification task by using two real-world datasets. The data size of these experiments is of the same order of the magnitude, or even higher, of the one used in the related works [7, 8]. It has been also compared against a classical computing algorithm working in continual setting. The experimental results are encouraging and show the potential superiority in terms of accurate estimations over different experimental configurations.

2. Quantum-classical framework for binary classification

The overall framework (illustrated in Figure 1) relies on the classical computing techniques of feature selection, data sampling, normalization and model optimization. On the quantum side, it integrates quantum encoding techniques, quantum neural networks in the form of variational quantum circuits for the problem of binary classification and quantum distance estimation.The framework faces a binary classification problem that can be formulated as follows. We have data instances described by $\mathbf{X} \cup y$, $\mathbf{X}$ are the descriptive attributes/features, while $y \in \{-1, 1\}$ denotes the class label. By operating in continual learning, the framework alternates training sessions, where we have labelled data blocks, with prediction sessions, where the data instances have

no class label. The succession of training sessions and prediction sessions is not predefined, coherently with the realistic assumption according to which the distribution of labelled and unlabelled data instances is not previously established and therefore not all the data instances are labelled.

In the following, we first provide a short description of each component and then describe how these work in the whole framework.

Feature selection operates only at the beginning and selects the subset of descriptive features which we will consider afterwards. It exploits a classical computing technique based on the mutual information between the class labels. This operation has been used to perceive the most relevant characteristics and alleviate the problem of choosing the dimension of the quantum circuits in terms of qubits. In fact, the number of the features of the input data determines the number of qubits.

Normalization scales the values of the previously selected features within the range of [0,1] by using the standard min-max function on the original ranges. It is performed for each incoming data block, both those of training and those of prediction.

Data sampling selects a subset of the labelled data instances within the previously data block of training session. The samples will contain data instances of both the class labels and, for each class label, the component takes data instances with simple random techniques without replacement. The sample size is fixed.

Quantum Centroid Distance Estimation operates only on the labelled data instances and allows us to detect drift within data. Distance estimation is a standard operation in the realm of the distance-based clustering and is used to build clusters and determine centroids, but, in this work, there is no clustering procedure as it is typically defined. Indeed, we determine two centroids as synthesis of the properties of the data instances of the two class labels, and use distance estimation to compute the distance between the centroids and labelled data instances. The centroids represent class prototypes and are used to identify the data instances underlying the drift. When the drift occurs, the two centroids are re-determined. To implement these operations we resort to the notion of *fidelity* between quantum states [9] and, inspired to the algorithmic decisions of [10], design the computation of the distances between single data instances and the two centroids in superposition. The estimation of the fidelity can be performed through the *SWAP test* implemented in the following circuit which presents three groups of qubits, the first one for the single data instance, the second one for the centroids and the third one as ancillary. We described it by tensor-product terms:

$$\Gamma(X) = U_3(X)|0\rangle \quad \Gamma(C_j) = U_3(C_j)|0\rangle \tag{1}$$

where, U_3 stands for the unitary matrix

$$U3(\theta, \phi, \lambda) = \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) & -e^{i\lambda}\sin\left(\frac{\theta}{2}\right) \\ e^{i\phi}\sin\left(\frac{\theta}{2}\right) & e^{i(\phi+\lambda)}\cos\left(\frac{\theta}{2}\right) \end{pmatrix} \tag{2}$$

X stands for the data instance represented through Γ() with the angle encoding and C_j is the j-th centroid out of k centroids (k=2).

So, the state preparation is defined as:

$$\prod_{j=1}^{2} \Gamma(C_j) \otimes \Gamma(X) \otimes I^{\otimes k} \tag{3}$$

while the SWAP test is implemented with the following term:

$$(H^{\otimes k} \otimes I^{\otimes 2k}) \mathscr{K}_{Y_j,\Gamma(C_j),\Gamma(X)} (H^{\otimes k} \otimes I^{\otimes 2k}) \tag{4}$$

where, $\mathscr{K}$ is the Fredkin gate and Y_j is the j-th ancillary qubit which works as control qubit for the $\mathscr{K}$.

Quantum Classification works in two modalities, training and prediction. It is implemented through two quantum circuits with a number of qubits determined by the number features selected by Feature selection. During the training sessions, the classifier learns on the data block currently built with labelled data instances. The classifier is instead used during the prediction sessions to infer the class labels on the data block currently built with unlabelled data instances.

The first quantum circuit takes the classical data and represents them as quantum states to be assigned to the qubits. This circuit implements a feature mapping operation $\mathscr{F}$ which encodes a real-valued data instance X into quantum states spanning d qubits:

$$|\psi(X)\rangle = \mathscr{F}(X)|0\rangle^{\otimes d} \tag{5}$$

where, $|0\rangle^{\otimes d}$ denotes the register with d-qubits at the state $|0\rangle$ ($|0\rangle \otimes ... \otimes |0\rangle$). In this work, $\mathscr{F}$ has been implemented as follows

$$R_z{}^{\otimes d}(X) H^{\otimes d} |0\rangle^{\otimes d} \tag{6}$$

where, the parameter for each gate R_z is the normalized real-valued of of the feature (corresponding to the qubit on which R_z works). The term $H^{\otimes d}$ denotes the tensor product H $\otimes ... \otimes$H over d occurrences (that is, the number of selected features) of the gate H (the same holds for R_z).

The second circuit is variational and manipulates the quantum states returned by the first circuit. It implements a quantum neural network composed of layers of entangled rotation gates. Entangled rotation gates are matrix operations which combine the gates Hadamard, CNOT and Rotation under the quantum physics effect of the entanglement [11]. The second circuit completes the structure of gates which builds the classifier ϕ

$$|\phi(X,\theta)\rangle = \mathscr{V}(\theta)|\psi(X)\rangle \tag{7}$$

where, $\mathscr{V}$ is the variational circuit, θ denotes the parameters of the parameterized gates that being optimized. In this work, $\mathscr{V}$ has been implemented as follows

$$R_y{}^{\otimes d} CX^{\otimes d} R_y{}^{\otimes d} \tag{8}$$

where, each occurrence of the two-qubit gate CX takes one pair of qubits (over the d-qubit register) composed by the consecutive qubits indexed as i and $i+1$.

Finally, we perform measurements on the qubits and the measured state is recorded. So, we can estimate the expectation value of the circuit on x and θ, by measuring the state over multiple runs, with the following

$$|\mathscr{E}(X,\theta) = \langle\phi(X,\theta)|\sigma_z^{\otimes d}|\phi(X,\theta)\rangle \tag{9}$$

where, $\sigma_z^{\otimes d}$ is the tensor product of the single qubit gate σ_z over d occurrences. The gate σ_z has the interesting property that if the measured quantum state has odd parity, it returns -1 (as eigenvalue), while, if the measured quantum state has even parity, it returns 1. This implies that the expectation value of the circuit will always be within the interval $[-1, 1]$. We can use this property to relate the expectation value to the probability that a data instance X being assigned to a class label y, that is:

$$P(y|X) = \frac{y\mathscr{E}(X,\theta) + 1}{2} \tag{10}$$

The probability P(y|X) is exploited in the optimization process concerning the parameters θ. In particular, the optimizer iteratively updates the circuit parameters by minimizing a cost function, which accounts for the negative log-likelihood of the probabilities P(y|X) computed on the current labelled data-blocks, that is:

$$-\frac{1}{size}\sum_{i=1}^{size} log(P(y_i|X_i)) \tag{11}$$

where, *size* is the number of data instances of the data block, X_i is the i-th data instance of the data block.

The cost function is minimized by a classical computing optimizer based on gradient descent. The derivative concerns the expectation value $\mathscr{E}()$ with respect to the current values of θ and is computed by means of the parameter shift rule [12]:

$$\frac{d\mathscr{E}}{d\theta} = \frac{\mathscr{E}(\theta+\epsilon) - \mathscr{E}(\theta_k - \epsilon)}{2} \tag{12}$$

The gradient value is the difference between the two output values of the circuit: the first value is the output of the circuit with the parameter θ_k increased by a value ϵ, and the second value is the parameter θ_k decreased by ϵ.

Continual learning. Learning classification models on continuously incoming data can be faced with time-windows models [13] in continual setting. Time-windows models allow us to handle data instances by equally-sized blocks on which we train, update and apply the classification model. During a training session, the training modality of Quantum classification component is activated (Figure 1), which implies the execution the feature mapping $\mathscr{F}$ on the data instances of the current data block and optimization process of the parameters θ of the variational circuit $\mathscr{V}$. Differently, during a prediction session, the Quantum classification only estimates the class labels on the current data block by using the classification model up there updated.

To keep the classifier updated, we have to deal with the catastrophic forgetting effect raising when updating neural networks. By the way, one of the purposes of this work is investigating whether this is what happens also on the quantum neural networks. In the literature, three alternatives are mainly suggested, *replay methods*, *regularization-based methods*, *parameter isolation methods* [6]. Considering that the replay methods represent the solution which asks for less and leaves unchanged the number of hyper-parameters of the neural network, we lean for this approach.

The framework operates in three steps, namely *initialization*, *update*, *prediction*. Training sessions are performed at the initialization and update. At the initialization step, the classifier is trained from scratch on the first data block DB_1 (Figure 1). The operation of *Feature selection* is used only at the initialization step, so the other steps of the framework work on the features before selected. Still at the initialization, two centroids, one for each class label, are determined from the labelled data instances of the data block DB_1. As new labelled data blocks will be acquired, the centroids will be re-computed. However, the computation of the centroids (Quantum centroid distance estimation) relies on the data representation based on angle encoding to two dimensions which use two features that have not been selected for the Quantum classification. Specifically, the feature value is added to 1 and multiplied by half π, to be an admissible value for the gate U_3. This is a data encoding different from the feature mapping used for the Quantum classification. The rationale behind is to use a different representational space in order to capture a different characteristics of the data from those expressed by the feature mapping of the Quantum classification.

Next, the framework prepares the steps of update and prediction by collecting labelled data instances in a data block DB_i and unlabelled data instances in DB_j. Both data blocks store the data instances in the order they arrive. As soon as one of the two data blocks is being filled (the number of collected data instances is equal to the predefined size), either updated step or prediction step is performed. By supposing the data block DB_i of labelled data instances has been filled for first, the update step will be performed, otherwise it will be the turn of prediction step working on DB_j. When the update step starts, it first checks for possible concept drifts within the current data block, and, if any is present, it updates the classification model. To check the presence of drifts, we rely on a classical computing technique, that is, Page-Hinkley test [14], which, in this work, detects the changes of a cluster that new data instances added can imply. As indicator of the characteristics of a cluster, we use the sum of the squared errors

$$SSE : \sum_{instance_h \in DB_i} distance(c_k, instance_h) \tag{13}$$

where, DB_i is the current data block, c_k refers to one of the two centroids, the distance is the one introduced in Quantum Centroid Distance Estimation component. This way, the Page-Hinkley test spots the presence of drifts when the SSE computed on the new data instances greatly differs from the one computed on the data instances previously processed. When this happens the two centroids are recomputed considering the new data instances and the classification model is updated. The training set used for the current learning round is composed of data instances of the current data block (containing the new data instances) and those provided by the component of *Data sampling*. As explained above, this is done to mitigate the effect of catastrophic forgetting.

3. Experiments on real-world datasets

We implemented the proposed framework in IBM Qiskit [15] and run experiments by using simulators on two real-world datasets, more precisely *Ozone level detection* [1] (having 2536 data instances, 73 features) and *Spambase* (having 4600 data instances, 57 features) [2]. Data blocks have been partitioned so as having a portion of 75% of the dataset as labelled data instances (training sessions) and the remaining 25% as unlabelled data instances (prediction sessions and testing sets of the evaluation). The classical computing components described in Section 2 are those available in the toolkit Scikit-learn [16]. The number of runs of the classification model to estimate the expectation values is 1024, while the number of iterations (epochs) to optimize the parameters is 20. The number of layers for the variational quantum circuit is 3. The sample size of Data sampling is 30% the data-block size.

Experiments have been performed to emphasize the impact of the technical configuration of the framework on the predictive capability, namely number of qubits (corresponding to the features selected) and size of the data blocks (number of data instances in each training/prediction session). In Table 1, we report the F1-score values of the proposed framework (named as *HYQOL*) compared to i) a classical computing solution (*CC*, originally designed for data stream learning) [17] and ii) a baseline of the framework that works on the whole dataset (*FQC*). Both CC and FQC have been tested with the same features selected for HYQOL.

The values illustrated have been computed as the average computed over the data blocks. As we can see, except two trials, HYQOL does not never worst than CC, even when the number of qubit is the higher (i.e., 8). Also, we note that the configurations of HYQOL with smallest set of qubits (i.e., 2) are better than those with largest set (i.e., 8), without, however, particular discrepancy between the two endpoints. The size of the data blocks seems not be determinant for the accuracies, but, it is evident that continual learning can be beneficial for quantum-based classifiers compared to the version that works on the whole dataset (*FQC*). In Table 2, we report the averaged values of the cost function taken at the last iteration (20-th iteration of the optimizer of *HYQOL*). As we can see, the lower costs are reached in correspondence of the higher predictive performance, that is, data-block size at 200 with 2 and 5 qubits.

4. Related work

Continual learning, also known as lifelong learning, incremental learning, or sequential learning, is a field of research dedicated to finding solutions for acquiring knowledge from an ever-changing stream of data involving multiple tasks [5].

As our knowledge, the works of quantum machine learning focused on continual learning are very few, quite recent and often in preliminary form. The paper [18] focuses on training sessions of different classification tasks. Specifically, a sequence of quantum state classification tasks is continually learned by a variational quantum classifier whose parameters are optimized by a classical gradient-based optimizer. Inspired by the category replay methods (the same we follow in this work), they propose to constrains the model updated by projecting the gradient direction

[1] https://archive.ics.uci.edu/ml/datasets/ozone+level+detection
[2] https://archive.ics.uci.edu/ml/datasets/Spambase

Table 1
F1-score (in [0,1] of the proposed framework against a classical computing solution and a hybrid solution without continual learning. Dataset *Ozone level detection* at the top. Dataset *Spambase* at the bottom.)

data-block size		#qubits		
		2	5	8
50	HYQOL	0,9	0,9	0,9
	CC	0,9	0,9	0,9
100	HYQOL	0,93	0,89	0,86
	CC	0,87	0,87	0,87
200	HYQOL	0,94	0,94	0,91
	CC	0,92	0,92	0,92
400	HYQOL	0,92	0,91	0,9
	CC	0,9	0,9	0,9
	FQC	0,9	0,87	0,81
data-block size		#qubits		
		2	5	8
50	HYQOL	0,88	0,84	0,84
	CC	0,82	0,82	0,82
100	HYQOL	0,87	0,87	0,85
	CC	0,82	0,82	0,82
200	HYQOL	0,9	0,89	0,8
	CC	0,89	0,89	0,89
400	HYQOL	0,84	0,84	0,84
	CC	0,86	0,9	0,84
	FQC	0,83	0,83	0,8

Table 2
Cost function values of the classical computing optimizer working on the variational circuit. Dataset *Ozone level detection* at the top. Dataset *Spambase* at the bottom.)

data-block size	#qubits		
	2	5	8
50	0,67	0,71	0,73
100	0,65	0,7	0,7
200	0,6	0,6	0,65
400	0,65	0,65	0,7
data-block size	#qubits		
	2	5	8
50	0,61	0,61	0,62
100	0,61	0,61	0,61
200	0,48	0,48	0,62
400	0,51	0,51	0,52

on the region outlined by previous task gradients. This is done also by storing a fraction of the training data of previous tasks (gradient episodic memory, GEM) on which the gradient descent is computed. A drawback is the necessity of computing gradients of previous tasks at each training iteration. In [19], it has been observed that as quantum classifiers are exposed to new

classification tasks, their performance on previous tasks can deteriorate. To address this issue, a method called elastic weight consolidation (EWC) has been introduced. EWC helps protect the parameters deemed crucial for the previous tasks from undergoing drastic updates. Numerical experiments have illustrated that a quantum classifier can continuously learn and adapt to three distinct classification tasks without experiencing significant forgetting. However, this work has been tested on classical data (e.g. MNIST images) which originally are not affected by drift of data distribution and class labels, but characterized by new class labels (tasks). This may make the changes of the characteristics of data of the same labels partially handled. The hybrid quantum-classical method described in [20] reports the use of classical convolutional neural networks with quantum layers. To mitigate the performance drop they propose to inject the information of the regions of prediction identified on visual saliency maps. All these works deal with the update of the classification model through purely classical computing, while we present a quantum distance-based method to identify data instances revealing the drift. By the way, GEM uses previously processed data kept in the memory to modify the gradient of the current data. Unlike EWC, the performance of the old data is more likely to increase because the previous data are rehearsed in the parameter update. That is another argument behind our decision of selecting old samples through quantum techniques.

5. Conclusions

In this paper, we investigated the viability of quantum machine learning solutions to work on the realistic scenarios of changeability of the statistical properties of the data, which often implies the variability of the performances of the model. We conjecture this can be a machine learning problem in which the quantum solutions can lead innovation. On simulated hardware, the hybrid quantum-classical proposal offers encouraging results, in terms of accuracy, often better than a classical computing solution working on data stream and hybrid solution working in batch mode (no continual learning). As our opinion, three take-home messages can be identified from this paper. The first one is methodological, in that the continual learning opens to practical applications able to combine quantum computing and classical computing techniques, which is likely the only way to concretely use current quantum technologies. The second one is experimental, in that it provides arguments on the fact that stable quantum devices could even do better in terms of performances and quality of the results, when used in predictive tasks. The third one tell us that, although the high-performance computation and tractability of hard problems are the promises of quantum computing which, with the current devices, often are not kept, the research on the lifelong computation can be a field in which quantum computing can already bring interesting results.

As future work, we plan two investigate two main research lines, one methodological, the other one applicative. In the first case, we will focus on the technique of optimization of parameters (component Quantum classification) and study machine/deep learning solutions to predict the parameter values in order to reduce the computational cost of the classical optimizer. In the second case, we will focus on more complex data, like social networks and interaction graphs, in order to handle challenging tasks, like the one of link prediction between actors and services.

Acknowledgment

Corrado Loglisci acknowledges the financial support from the project "PNRR MUR project PE0000023-NQSTI" for this research. The paper also achieves the research activities of the project MUR-Fondo Promozione e Sviluppo - DM 737/2021, CUP: H91I21001630006, INNOAGROECOS.

References

[1] J. Preskill, Quantum Computing in the NISQ era and beyond, Quantum 2 (2018) 79. URL: https://doi.org/10.22331/q-2018-08-06-79. doi:10.22331/q-2018-08-06-79.

[2] A. Callison, N. Chancellor, Hybrid quantum-classical algorithms in the noisy intermediate-scale quantum era and beyond, Phys. Rev. A 106 (2022) 010101. URL: https://link.aps.org/doi/10.1103/PhysRevA.106.010101. doi:10.1103/PhysRevA.106.010101.

[3] C. Loglisci, I. Diliso, D. Malerba, A hybrid quantum-classical framework for binary classification in online learning, in: SEBD, volume 3478 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2023, pp. 88–99.

[4] J. Peng, B. Tang, H. Jiang, Z. Li, Y. Lei, T. Lin, H. Li, Overcoming long-term catastrophic forgetting through adversarial neural pruning and synaptic consolidation, IEEE Trans. Neural Networks Learn. Syst. 33 (2022) 4243–4256. URL: https://doi.org/10.1109/TNNLS.2021.3056201. doi:10.1109/TNNLS.2021.3056201.

[5] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, S. Wermter, Continual lifelong learning with neural networks: A review, Neural Networks 113 (2019) 54–71. URL: https://doi.org/10.1016/j.neunet.2019.01.012. doi:10.1016/j.neunet.2019.01.012.

[6] M. D. Lange, R. Aljundi, M. Masana, S. Parisot, X. Jia, A. Leonardis, G. G. Slabaugh, T. Tuytelaars, A continual learning survey: Defying forgetting in classification tasks, IEEE Trans. Pattern Anal. Mach. Intell. 44 (2022) 3366–3385. URL: https://doi.org/10.1109/TPAMI.2021.3057446. doi:10.1109/TPAMI.2021.3057446.

[7] D. Arthur, P. Date, Hybrid quantum-classical neural networks, in: IEEE International Conference on Quantum Computing and Engineering, QCE 2022, Broomfield, CO, USA, September 18-23, 2022, 2022, pp. 49–55. doi:10.1109/QCE53715.2022.00023.

[8] A. Chalumuri, R. Kune, B. S. Manoj, A hybrid classical-quantum approach for multi-class classification, Quantum Inf. Process. 20 (2021) 119. URL: https://doi.org/10.1007/s11128-021-03029-9. doi:10.1007/s11128-021-03029-9.

[9] E. Aïmeur, G. Brassard, S. Gambs, Machine learning in a quantum world, in: Advances in Artificial Intelligence, 19th Conference of the Canadian Society for Computational Studies of Intelligence, Canadian AI 2006, Québec City, Québec, Canada, June 7-9, 2006, Proceedings, 2006, pp. 431–442. URL: https://doi.org/10.1007/11766247_37. doi:10.1007/11766247_37.

[10] I. Kerenidis, J. Landman, A. Luongo, A. Prakash, q-means: A quantum algorithm for unsupervised machine learning, in: H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, R. Garnett (Eds.), Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada, 2019, pp. 4136–4146.

[11] M. A. Nielsen, I. L. Chuang, Quantum Computation and Quantum Information, Cambridge University Press, 2000.

[12] D. Wierichs, J. Izaac, C. Wang, C. Y.-Y. Lin, General parameter-shift rules for quantum gradients, Quantum 6 (2022) 677. URL: https://doi.org/10.22331/q-2022-03-30-677. doi:10.22331/q-2022-03-30-677.

[13] J. Gama, M. G. (Eds), Learning from Data Streams – Processing techniques in Sensor Networks, Springer, 2007.

[14] E. S. Page, Continuous inspection schemes, Biometrika 41 (1954) 100–115. URL: http://www.jstor.org/stable/2333009.

[15] M. S. Anis, et al, Qiskit: An open-source framework for quantum computing, 2021. doi:10.5281/zenodo.2573505.

[16] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in Python, Journal of Machine Learning Research 12 (2011) 2825–2830.

[17] J. Montiel, J. Read, A. Bifet, T. Abdessalem, Scikit-multiflow: A multi-output streaming framework, Journal of Machine Learning Research 19 (2018) 1–5. URL: http://jmlr.org/papers/v19/18-251.html.

[18] Quantum continual learning of quantum data realizing knowledge backward transfer, Physica A: Statistical Mechanics and its Applications (2023) 128779. doi:https://doi.org/10.1016/j.physa.2023.128779.

[19] W. Jiang, Z. Lu, D. Deng, Quantum continual learning overcoming catastrophic forgetting, CoRR abs/2108.02786 (2021). URL: https://arxiv.org/abs/2108.02786. arXiv:2108.02786.

[20] S. Jain, Cqural: A novel CNN based hybrid architecture for quantum continual machine learning, CoRR abs/2305.09738 (2023). URL: https://doi.org/10.48550/arXiv.2305.09738. doi:10.48550/arXiv.2305.09738. arXiv:2305.09738.