Multiclass classification using variational quantum circuit on benchmark dataset

Muhammad Hamid[1], Bashir Alam[1], Om Pal[2]
[1]Department of Computer Engineering, Faculty of Engineering and Technology, Jamia Millia Islamia University, New Delhi, India
[2]Department of Computer Science, University of Delhi, New Delhi, India

Article Info

Article history:

Received Feb 21, 2025
Revised Jan 17, 2026
Accepted Mar 30, 2026

Keywords:

Multi-class classification
Quantum computing
Quantum neural networks
Variational quantum circuits
hybrid quantum-classical algorithm

ABSTRACT

Classification is a major task in data science. Data classification is required in many industries such as healthcare, transport, and finance. Noisy intermediate-scale quantum (NISQ) era. Quantum computers are capable of solving complex data challenges and can be used for the classification of the data with minimum features. In this regard, quantum neural networks are being used extensively for data classification. In this paper, we employ variational quantum circuits for the task of multiclass classification. A hybrid approach is used for building the neural network. In which quantum circuits are used for the feedforward architecture, while in back-propagation, parameters are updated using a classical optimizer on classical computers. We have successfully demonstrated multiclass classification using the proposed approach on benchmark data sets. Our results show that variational quantum circuit (VQC) are a promising candidate for classification problems with fewer features. We have performed experiments on International Business Machines Corporation (IBM) quantum hardware and simulators.

Corresponding Author:

Muhammad Hamid
Department of Computer Engineering, Faculty of Engineering and Technology
Jamia Millia Islamia University
Jamia Nagar, New Delhi, India
Email: hamidmd504@gmail.com

1. INTRODUCTION

Artificial intelligence has experienced exponential growth in recent decades, leading to a wide range of applications in healthcare, agriculture, transportation, finance, entertainment, and many other domains. However, training machine learning models typically requires substantial computational resources and time. Sometimes training a deep learning model may take several days and even a month. Despite recent advances in classical computers, modern computers are reaching their limits in implementing machine learning in different areas. Researchers and computer scientists are looking for an alternative to classical computing, and they see quantum computers as an alternative solution to overcome this problem. Quantum computers use superposition, entanglement, and interference to process information. Quantum computers have shown speedup in factorization of numbers [1] and searching in unstructured data [2].

Quantum machine learning (QML) is an emerging interdisciplinary research area that integrates principles from quantum computing and machine learning to achieve computational advantages over classical approaches. Although QML remains in its nascent stage, a diverse range of algorithms has already been proposed within this domain. QML has been investigated across multiple learning paradigms [3]–[5]. Quantum support vector machine (QSVM) [6], quantum K-nearest neighbor (QK-NN) [7] are examples of algorithms that are being used for the binary classification task. Also, we have witnessed the power of

Hybrid-QCNN for multi-class classification tasks [8]. In this sequence, Blank *et al*. [9] introduced a kernel-based quantum classifier for big data classification. Adhikary *et al.* [10] employed an N-level quantum system to encode features for a hybrid quantum-classical (HQC) classifier, demonstrating its application to classification tasks across multiple datasets. Date *et al.* [11] introduced a HQC neural network architecture for binary classification. Subsequently, in [12], the authors extended these methods to multi-class problems by employing amplitude encoding for three-class classification.

Here we present a mixed quantum-classical-based approach using variational quantum circuits for multi-class classification tasks on IRIS and wheat seed datasets. Our study mainly contributes to multi-class classification on the IRIS flower dataset into three classes with only four features and classification of wheat seed with only four features into three classes.

This paper is divided into several sections which are as follows, section 1 covers introduction. In section 2, we have covered related work on multi-class classification, and variational quantum circuits are explained in section 3. In section 4 we have explained our proposed scheme and data preparation method, and section 5 is a discussion about results and finally our findings are concluded in the last section.

2. RELATED WORKS

Quantum neural networks have been studied as a method for classifying classical data. It is the most explored method for classification with quantum computers. Wu *et al*. [13] introduced a new scalable method using quantum neural networks for classification. In which small quantum hardware is used cooperatively. The entire image is segmented into multiple regions, and features from each region are extracted using small-scale quantum devices. Extracted local features are sent to the quantum device to perform prediction by combining all local features in parallel. They conducted experiments and evaluated the outcomes for binary classification on the Modified National Institute of Standards and Technology (MNIST) digit dataset. The limitation in this approach is its efficiency for larger or more complex data sets. Also, it is not known whether this approach can be used for multiclass classification on classical data. In another paper, Wu *et al*. [14] proposed a variational quantum multi-classifier based on correlation and measurement. The evaluation findings show that they attained enhanced performance with minimal quantum resources and a basic ansatz. The limitation in this method is that it depends on quantum state tomography to reconstruct the readout state, which can be resource-intensive in terms of both quantum circuits and classical computation for state reconstruction.

Wang *et al*. [15] presented on-chip parametrized quantum circuit training with parameter shift. They find that gradients obtained by parameter shift have low fidelity, which causes a decrease in training accuracy. To achieve this, they also suggest probabilistic gradient pruning, which identifies gradients with potentially significant errors. The findings show that on-chip training can classify images into two and four classes with approximately 90% and 60% accuracy, respectively. The key limitations of this approach revolve around the sensitivity to noise, potential drawbacks of gradient pruning, limited scalability and generalization, and the dependence on specific quantum hardware. Chalumuri *et al*. [16] proposed a quantum multi-class classifier (QMCC) implemented as a parameterized circuit, where state preparation is performed via a unitary operation on a single qubit. Three benchmark datasets were used for their quantum simulations: the Wireless Indoor Localization, Banknote Authentication (BNA), and the Iris dataset. The QMCC model identified the Iris, BNA, and WIL datasets with an accuracy of 92.10%, 89.50%, and 91.73%, respectively. The limitations are that the model might be prone to overfitting on small datasets. Shen *et al*. improve a variational algorithm [17] that generally prepares the encoded data to solve the data encoding problem. The fashion-MNIST dataset is encoded using the most recent technique. They provide a proof of concept for the near-term practicality of our data encoding technique by deploying basic quantum variational classifiers that are trained on the encoded dataset on a modern quantum computer and achieve moderate accuracy. The main limitations of the approach include the reliance on shallow circuits, which may not scale well to more complex problems.

3. BACKGROUND

In supervised learning, each machine learning model is trained using labeled training data. After training the algorithm generates a model capable of predicting the labels of new, unseen data. The inherent unpredictability of quantum mechanics is leveraged during training to enhance the model, as machine learning heavily depends on linear algebra. They can be used to address various tasks, such as reinforcement learning, supervised learning, and unsupervised learning [18]. Quantum computing concepts have enabled conventional randomized algorithms to perform exponentially faster than standard algorithms which is another technique for completing the QML task is variation quantum circuits [19], [20].

The variational quantum circuit can be used as an artificial neural network. It is also known as a parameterized quantum circuit, with its parameters serving as the neural network's weights. These parameters are updated on the classical system in each epoch. The quantum circuits can be employed to compute the cost function, which should be kept as simple as possible for the machine-learning model to perform effectively. We use a classical computer to tune the parameters. This approach makes extensive use of parameterized, optimized quantum gates. Usually, these quantum gates include single-qubit rotations gates (R_x, R_γ, R_z) as well as two-qubit controlled NOT (CNOT) gates. The optimized circuit is used for classification. Figure 1 illustrates the overall view of variational quantum circuits with quantum and classical parts. Generally, variational quantum circuit (VQC) can be written as:

$$U(\theta)\ \psi = \prod n{=}1\ Ui\psi \tag{1}$$

Here *U(θ)* represents a parameterized universal gate, n is the total number of gates, and ψ is the input quantum state. By adjusting the parameters θ, the action of *U* can be modified. Every variational algorithm consists of the following steps.

- Data encoding - A crucial part of a VQC is embedding classical data into a quantum state before it can be processed by the circuit. This can be accomplished in several ways, with basis and amplitude encoding being the most popular methods.
- Ansatz design - This step involves the design of quantum circuits using quantum gates and making qubits into superposition and entanglement.
- Measurement - Apply the measurement to collapse the quantum state into either 0 or 1 binary state.
- Post-processing - It is the mapping of the binary output obtained from VQC with the labels of the dataset for classification. We check here if the predicted labels are correct or not.
- Optimization - In this step, the parameter optimization is done classically to reduce the cost function. We can use any classical optimizer to optimize the parameter, such as gradient descent, ADAM, or stochastic gradient descent.

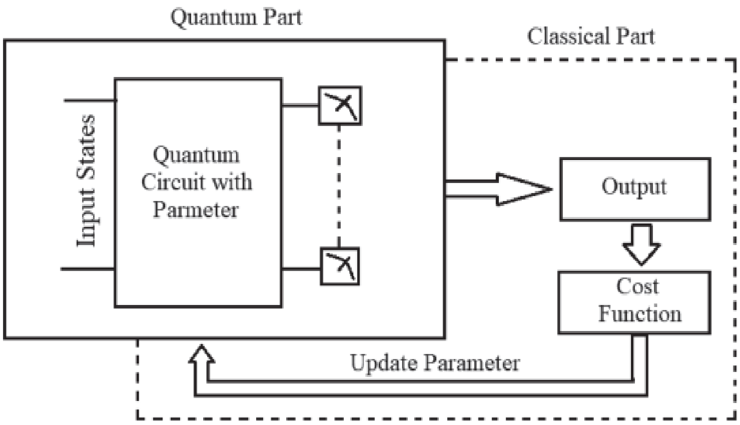

Figure 1. Diagram of a variational quantum circuit with input quantum states, parameterized gates, and classical optimization loop

4. METHODS

A VQC combined with classical optimization is employed for multi-class classification. Where classical data or features will be encoded to quantum states, this step is also known as feature mapping. The quantum ansatz, which consists of entangling and rotational gates, receives these prepared states as an input. This ansatz is parameterized by angles that can be adjusted during training. The output of this circuit is measured to yield bitstrings that represent classification results. The dataset is divided into training and validation sets for model training. The training set contains labeled data and is used for training, while the validation set is used to evaluate the model's performance. A classical optimization algorithm is employed to minimize a loss function, which compares predicted outputs with actual labels. During optimization, the parameters (angles) of the quantum circuit are repeatedly adjusted to minimize the cost function. After processing through the quantum circuit, measurements are taken from the qubits. The resulting bitstrings are interpreted as probabilities for class membership for classification tasks.

4.1. Data preparation

Let us explore our datasets. Mostly all data scientists are familiar with the Iris data [21]. The data set comprises three classes: Virginica, Versicolor, and Setosa, each of which has 50 instances. The data have 150 instances. Four features are present in each instance: sepal length, sepal breadth, petal length, and petal width. As each class has an equal number of instances, the data set is properly balanced. Next, is the wheat seeds dataset [22]. This dataset contains the three classes of wheat seed: Kama, Rosa, and Canadian. Each class of 70 instances are there. The total number of instances in the dataset is 210. There are seven features present in each instance: area, perimeter, length of kernel, width of kernel, compactness, asymmetry coefficient, and length of kernel groove. This dataset is also balanced. Both data is split into a training and test set of ratios 80:20 using sci-kit library. Figure 2 and Figure 3 are visualizations of data points of the selected data set. Figures 2(a)-2(b) are visualization of data point of IRIS flower dataset and wheat seeds.

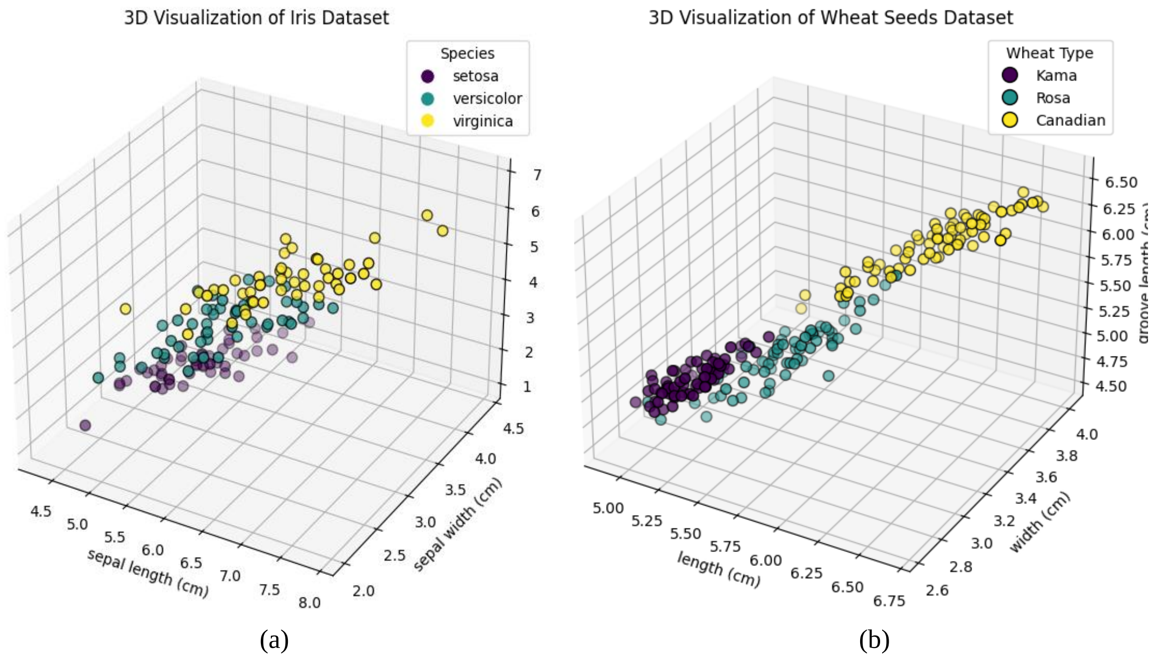

Figure 2. Visualization of data point of IRIS flower dataset (a) and wheat seeds (b)

4.2. Data encoding and state preparation

Quantum data encoding is the process of mapping classical data into quantum states so that a quantum computer can process it. In VQCs, it is one of the most important and crucial steps for accelerated QML algorithms. Common methods of quantum data encoding are as follows:

1) Basis Encoding: Basis encoding is the most straightforward approach for representing data in quantum circuits for arithmetic tasks. This method encodes the binary representation of classical data directly into quantum basis states. Typically, n qubits are required to represent n classical data points. The classical data point $x = (x_1, x_2, ...x_n)$ will be encoded as

$$\psi x = \otimes ni\ xi \tag{2}$$

2) Amplitude Encoding: The amplitude encoding represents input data of $x = (x_1, x_2, ...x_n)$ T of dimension $N = 2^N$ as amplitudes of an n-qubit quantum state $\phi(x)$ as

$$U\ \phi\ (x): x \in R\ N \rightarrow \phi(x) = 1\ /N * \Sigma N\ i{=}1\ xi\ I \tag{3}$$

where i is the ith computational basis state.

3) Angle Encoding: The angle encoding method represents classical data as the rotation angles (in radians) of qubit gates. To encode n data points, we need n qubits and n rotation gates of R_x, R_y, R_z acting on qubits. if $x = (x_1, x_2, ...x_n)$ then states will be prepared as:

$$\psi x = \otimes n\ i\ R(x\ i)x\ i \quad (4)$$

The worst-case time complexity of quantum encoding is exponential. LaRose *et al*. have presented different robust quantum encoding methods such as dense angle encoding, general qubit encoding, wavefunction encoding, and amplitude encoding and their results on different channels [23]. Angle coding and dense angle coding can be used to decrease a quantum circuit's depth. Nowadays, there are only a certain number of qubits in noisy intermediate-scale quantum (NISQ) devices. We should select an encoding that can create a balance between a number of qubits and circuit depth.

In our case for data encoding and state preparation we have used the standard ZZFeatureMap from the Qiskit library. The first step is to normalize the input features between 0 and 1. Each normalized feature is mapped to a rotational angle on the Bloch sphere using the AngleEmbedding technique. Once the rotation gates encode the data, entangling gates establish correlations between qubits, allowing the model to learn complex dependencies. To encode 4 features, we have used 4 qubits. Hence 4 rotations gates will be required and the rotation angle will be in the range between [0, 2π]. After the rotations, entanglement is applied between pairs of qubits using CZ gates. Two repetitions were employed to strike a balance between model accuracy and computational cost, yielding a circuit depth of 20 that provides effective classification performance while keeping complexity minimum.

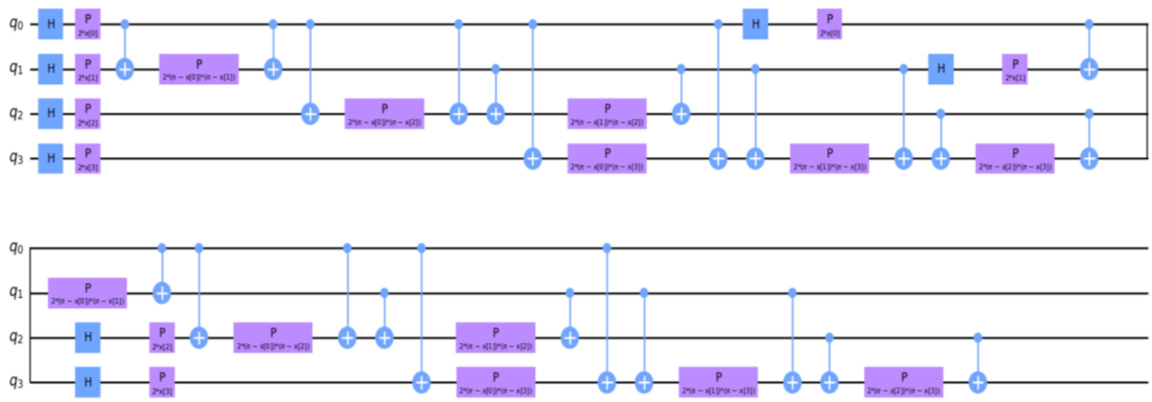

Figure 3. Detailed quantum circuit illustrating input state preparation using ZZFeatureMap, including qubit rotations and entanglement

4.3. Circuit design and training

After the state preparation, we will design an ansatz for classification. The ansatz provides the parameterized structure needed to perform optimization via classical optimizers. These gates in the ansatz are parameterized, and the parameters of these gates are what need to be optimized during training to minimize classification error. Choosing the right ansatz is crucial for the performance of quantum models. The layers in classical neural networks are directly similar to this ansatz. Each gate in this circuit works as a node in the neural network. It has a set of adjustable weights. The gap between the predictions and the known labeled data is described by the cost function. To minimize a cost function, we need to optimize the weights. In Figure 4 our circuit is plotted. This circuit has 16 trainable parameters, numbered from 0 to 15, which serve as the weights for the classifier. This circuit will act as the feed-forward layer of the neural network. The training includes learning the trainable parameters. In our case, learnable parameters are rotations, angles, entanglement operations, etc. of the quantum circuit. Measurements from the quantum circuit are used to compute the loss, and the optimizer iteratively updates the parameters to reduce the discrepancy between predicted and actual labels. In this training process constrained optimization by linear approximation (COBYLA) optimizer is used for optimizing the parameters. COBYLA is designed for optimizing non-linear, non-differentiable functions. It is a gradient-free, derivative-free optimizer. Its simplicity and gradient-free nature make it a strong contender for quantum optimization tasks, especially those involving variational methods. The training time will increase if we select a gradient-based optimizer. However, for smooth, well-behaved problems, or problems that can efficiently provide gradients, other optimizers like L-BFGS-B, SPSA or ADAM might offer faster convergence and better performance [24]. The stochastic optimizer such as simultaneous perturbation stochastic approximation (SPSA) can also be used for optimization.

Now we have our features, ansatz, and optimizer ready, we can train our classifier. The hyperparameters that we have tuned to achieve optimal performance for training a variational quantum circuit. 4 qubits are used to encode 4 features. To make the model converge faster we have used COBYLA optimizer with learning rate 0.001. Training is performed with a batch size of 16 over 250 epochs. We can train VQC using either a simulator or a real quantum computer. Here we will be using a quantum simulator as present real quantum hardware is noisy. Near the end of the 250 iterations of training, the cost function is not converging, we can see in Figure 5, indicating that the model's performance will not change even after increasing the number of iterations.

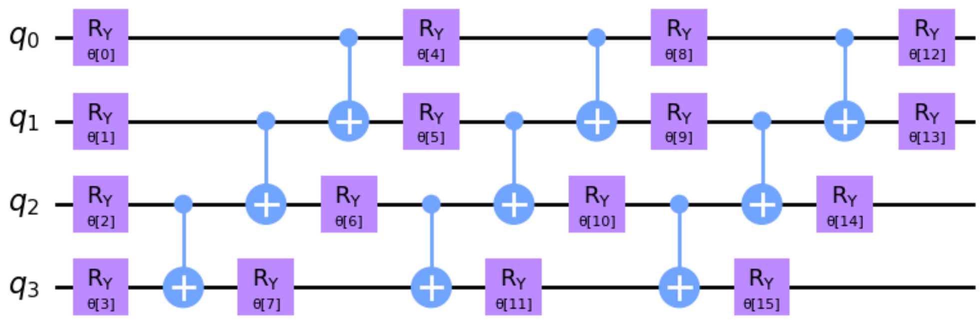

Figure 4. Parametrized quantum circuit used for classification of dataset

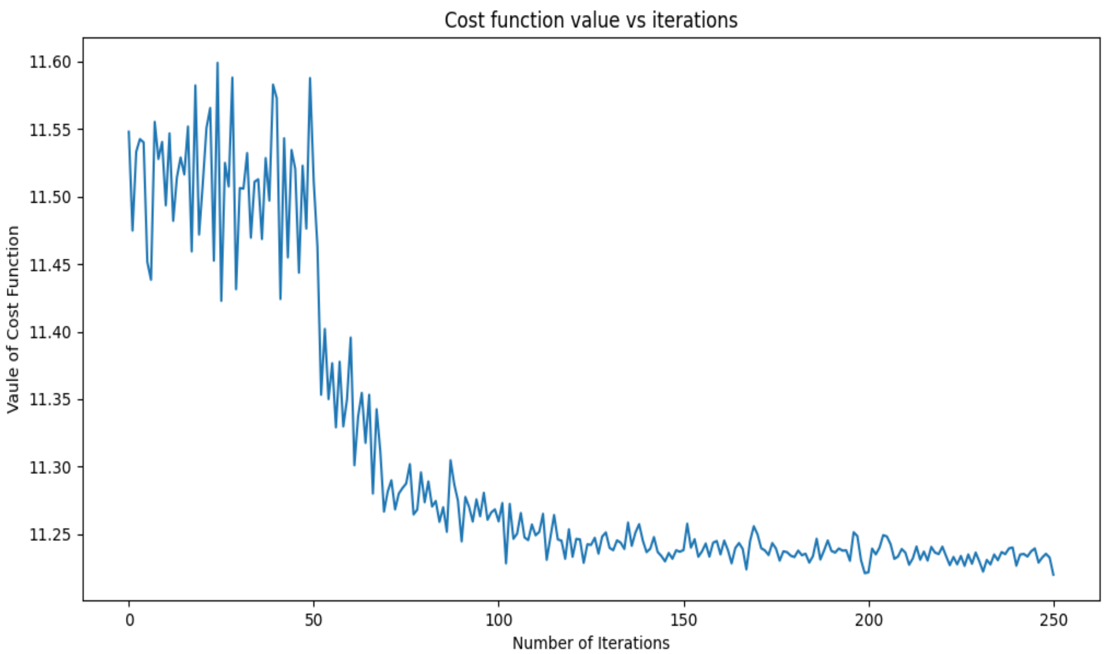

Figure 5. Representation of loss decay during training after every iteration of iris dataset

Next, we will use the above feature map and ansatz for the wheat seeds data set. Where the primary task is to classify the class of seed from three classes: kama, Rosa, canadian. There are seven features in each instance. We will reduce the dimensions to 4 features using principal component analysis (PCA) [25] while preserving the most important variance in the data. This is a form of preprocessing that helps the quantum model work with fewer, but more informative, dimensions. After performing PCA, the data is normalized to the range [-π, π], which is necessary for representing classical values as quantum gate rotation angles. An 80:20 split is used to allocate the dataset for training and validation purposes. Features are encoded using ZZFeatureMap, whereas the output is the string of predicted labels based on the test data. From Figure 6 We can see that the model is at minimum loss in 100 epochs.

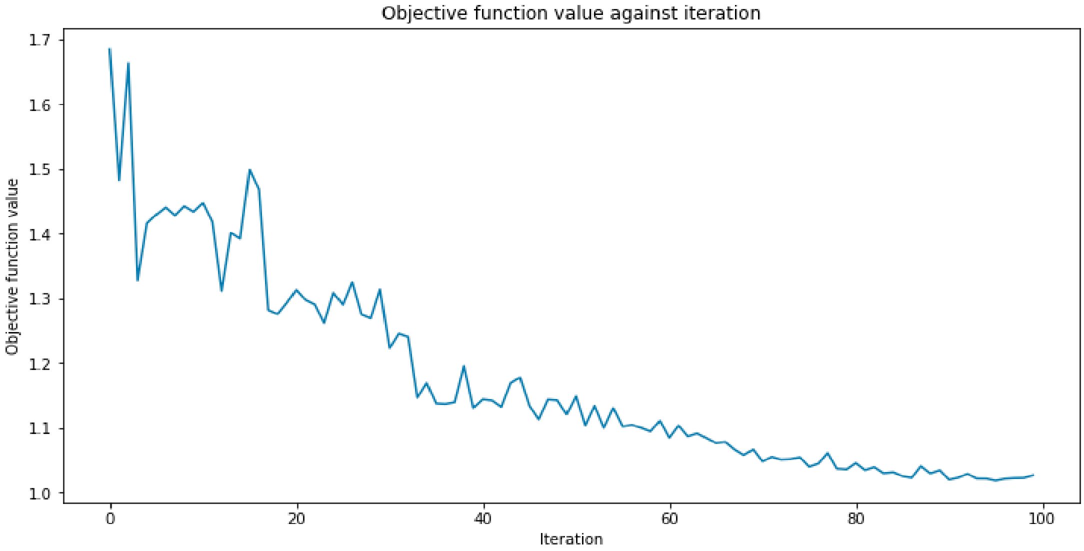

Figure 6. Representation of loss decay during training after every iteration of seeds dataset

5. RESULTS AND DISCUSSION

After training completion, we achieved a high score on the train set and test set. The labels from the IRIS dataset of unseen data can be predicted using this model. In this situation, while designing the circuit, we have modified the reps parameter, which dictates how many times we add a quantum gate to the circuit, which is similar to adding a hidden layer in the classical neural network. A greater number of quantum gates results in more parameters and more entanglement operations. The model is therefore more flexible, but becomes more complex with a larger number of parameters and it typically takes longer time to train. Even a minor change to the ansatz might produce improved outcomes; this indicates that the selection of hyperparameters is just as important in QML as it is in classical machine learning, and it could take some time to find the best values. As Iris data has only four features, we will be required to use only 4 qubits to process all the data, although this might not always be the case. If a data set contains more features than a modern quantum computer can accommodate, then we reduce the number of features, resulting in decreased performance for all models.

We have achieved an accuracy of 99% with classical support vector machine (SVM) on the training data and 97 % on the validation data of the Iris data set, respectively, whereas we can achieve a test accuracy of 89% and training accuracy of 90% using VQC. As we can see, the classical support vector machine produced the best results. Despite using only four features, the trained quantum model demonstrates good accuracy on the IRIS dataset. Unsurprisingly, classical models outperform their quantum versions; nonetheless, classical ML has advanced significantly, whereas quantum ML has yet to achieve that degree of maturity.

Three-class seeds classification is done with the wheat seeds dataset with the same circuits. Here we have trained our variational quantum circuit with a learning rate of 0.01, batch size 16, and COBYLA optimizer with 0.425 for 100 epochs. We have achieved nearly 92% accuracy on this dataset with only four features. In Table 1 we have listed the accuracies on different dataset.

This study also supports various work for multiclass classification using quantum neural networks [26]–[28]. Today, many quantum hardware and libraries are available for designing quantum circuits and verifying results on real hardware and simulators. Quantum computing tools include software development kits (SDKs) like Qiskit, Cirq, and PennyLane, as well as cloud-based platforms such as Amazon Bracket and Google Quantum Engine. These tools help developers interface with real quantum hardware [29], [30]. In this paper, we have used an IBM 'statevector'quantum simulator and hardware for evaluating our results and running our circuits.

Table 1. The performance of the ansatz on different dataset

Dataset	Number of classes	Number of features	Traning Accuracy	Testing Accuracy
IRIS	3	4	90.2%	89.7%
Seeds	3	4	92.4%	91.9%

6. CONCLUSION

Many complex problems that are challenging for classical computers to resolve could be resolved by quantum computers. A key advantage of quantum computing is its ability to perform quantum parallelism. For example, with n dimensions, a quantum computer can create a superposition of all 2 n possible states simultaneously, while a classical computer would have to examine each state one by one. In high-dimensional data, training deep neural networks or other complex models involves minimizing loss functions over many parameters. Quantum optimization could speed up the convergence of these models by exploring the parameter space more efficiently. Quantum feature maps can map high-dimensional data into a quantum state that may require fewer qubits to represent, allowing quantum algorithms to process the data more efficiently. But current quantum computers are still in the NISQ era, which means we have a limited number of qubits and these qubits are noisy which restrict us to achieve quantum advantage.

A shallow entangled circuit is designed to train VQC with the fewest possible trainable parameters. In this study, we demonstrated that quantum computing may be utilized to train neural networks and get results comparable to those of classical neural networks. Here, we demonstrated the use of variational quantum circuits for multiclass classification. Only a few parameters were used by VQC to identify the dataset's complex and nonlinear patterns. It is therefore computationally efficient. With minor modifications, the ansatz employed in this paper can be applied to different classification tasks. The future scope of this work is the study of multiclass classification with a large data set of higher dimensions.

ACKNOWLEDGMENTS

Authors thank Prof. Mansaf Alam from the Department of Computer Science Jamia Millia Islamia University for his continuous support and mentorship.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Muhammad Hamid	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	
Bashir Alam		✓				✓		✓	✓	✓		✓		
Om Pal	✓		✓	✓			✓			✓			✓	

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

INFORMED CONSENT

We have obtained informed consent from all individuals included in this study.

DATA AVAILABILITY

- The data that support the findings of this study are openly available in "Iris" UCI Machine Learning Repository, 1936. [Online]. Available: https://doi.org/10.24432/C56C76.
- The data that support the findings of this study are opnely available in "Seeds" UCI Machine Learning Repository, 2010. [Online]. Available: https://doi.org/10.24432/C5H30K.

REFERENCES

[1] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *SIAM Journal on Computing*, vol. 26, no. 5, pp. 1484–1509, 1997, doi: 10.1137/S0097539795293172.
[2] L. K. Grover, "A fast quantum mechanical algorithm for database search," *Proceedings of the Annual ACM Symposium on Theory of Computing*, vol. Part F129452, pp. 212–219, 1996, doi: 10.1145/237814.237866.
[3] V. Kulkarni, M. Kulkarni, and A. Pant, "Quantum computing methods for supervised learning," *Quantum Machine Intelligence*, vol. 3, no. 2, 2021, doi: 10.1007/s42484-021-00050-0.

[4] J. S. Otterbach *et al.*, "Unsupervised machine learning on a hybrid quantum computer," 2017, [Online]. Available: http://arxiv.org/abs/1712.05771.

[5] O. Lockwood and M. Si, "Reinforcement learning with quantum variational circuits," *Proceedings of the 16th AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, AIIDE 2020*, pp. 245–251, 2020, doi: 10.1609/aiide.v16i1.7437.

[6] P. Rebentrost, M. Mohseni, and S. Lloyd, "Quantum support vector machine for big data classification," *Physical Review Letters*, vol. 113, no. 3, 2014, doi: 10.1103/PhysRevLett.113.130503.

[7] Y. Dang, N. Jiang, H. Hu, Z. Ji, and W. Zhang, "Image classification based on quantum K-Nearest-Neighbor algorithm," *Quantum Information Processing*, vol. 17, no. 9, 2018, doi: 10.1007/s11128-018-2004-9.

[8] D. Bokhan, A. S. Mastiukova, A. S. Boev, D. N. Trubnikov, and A. K. Fedorov, "Multiclass classification using quantum convolutional neural networks with hybrid quantum-classical learning," *Frontiers in Physics*, vol. 10, 2022, doi: 10.3389/fphy.2022.1069985.

[9] C. Blank, D. K. Park, J.-K. K. Rhee, and F. Petruccione, "Quantum classifier with tailored quantum kernel," Mar. 2020, doi: 1909.02611.

[10] S. Adhikary, S. Dangwal, and D. Bhowmik, "Supervised learning with a quantum classifier using multi-level systems," *Quantum Information Processing*, vol. 19, no. 3, 2020, doi: 10.1007/s11128-020-2587-9.

[11] P. Date, D. Arthur, and L. Pusey-Nazzaro, "QUBO formulations for training machine learning models," *Scientific Reports*, vol. 11, no. 1, 2021, doi: 10.1038/s41598-021-89461-4.

[12] A. Zhang, X. He, and S. Zhao, "Quantum algorithm for neural network enhanced multi-class parallel classification," 2022, [Online]. Available: http://arxiv.org/abs/2203.04097.

[13] J. Wu, Z. Tao, and Q. Li, "wpScalable quantum neural networks for classification," *Proceedings - 2022 IEEE International Conference on Quantum Computing and Engineering, QCE 2022*, pp. 38–48, 2022, doi: 10.1109/QCE53715.2022.00022.

[14] J. Wu, T. Hu, and Q. Li, "MORE: measurement and correlation based variational quantum circuit for multi-classification," *Proceedings - 2023 IEEE International Conference on Quantum Computing and Engineering, QCE 2023*, vol. 1, pp. 208–218, 2023, doi: 10.1109/QCE57702.2023.00031.

[15] H. Wang, Z. Li, J. Gu, Y. Ding, D. Z. Pan, and S. Han, "QOC: quantum on-chip training with parameter shifts and gradient pruning," *Proceedings - Design Automation Conference*, pp. 655–660, 2022, doi: 10.1145/3489517.3530495.

[16] A. Chalumuri, R. Kune, and B. S. Manoj, "A hybrid classical-quantum approach for multi-class classification," *Quantum Information Processing*, vol. 20, no. 3, 2021, doi: 10.1007/s11128-021-03029-9.

[17] K. Shen, B. Jobst, E. Shishenina, and F. Pollmann, "Classification of the fashion-MNIST dataset on a quantum computer," 2024, [Online]. Available: http://arxiv.org/abs/2403.02405.

[18] M. Hamid and B. Alam, "Investigating classification with quantum computing," in *Intelligent Data Analytics, IoT, and Blockchain*, 1st Editio., Intelligent Data Analytics, IoT, and Blockchain, 2023, pp. 302–314.

[19] A. Kandala *et al.*, "Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets," *Nature*, vol. 549, no. 7671, pp. 242–246, Oct. 2017, doi: 10.1038/nature23879.

[20] E. Paquet and F. Soleymani, "QuantumLeap: Hybrid quantum neural network for financial predictions," *Expert Systems with Applications*, vol. 195, 2022, doi: 10.1016/j.eswa.2022.116583.

[21] R. Fisher, "Iris," *UCI Machine Learning Repository*, 1936.

[22] M. Charytanowicz, J. Niewczas, P. Kulczycki, P. Kowalski, and S. Lukasik, "Seeds," *UCI Machine Learning Repository*, 2010.

[23] R. Larose and B. Coyle, "Robust data encodings for quantum classifiers," *Physical Review A*, vol. 102, no. 3, 2020, doi: 10.1103/PhysRevA.102.032420.

[24] R. Y. Sun, "Optimization for deep learning: an overview," *Journal of the Operations Research Society of China*, vol. 8, no. 2, pp. 249–294, 2020, doi: 10.1007/s40305-020-00309-6.

[25] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 2, no. 1–3, pp. 37–52, 1987, doi: 10.1016/0169-7439(87)80084-9.

[26] J. Zhou, D. Li, Y. Tan, X. Yang, Y. Zheng, and X. Liu, "A multi-classification classifier based on variational quantum computation," *Quantum Information Processing*, vol. 22, no. 11, 2023, doi: 10.1007/s11128-023-04151-6.

[27] I. Piatrenka and M. Rusek, "Quantum variational multi-class classifier for the iris data set," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 13353 LNCS, pp. 247–260, 2022, doi: 10.1007/978-3-031-08760-8_21.

[28] C. Li, Z. Liu, and S. Zhang, "Multiclassification quantum neural network based on variational quantum circuits," p. 66, 2024, doi: 10.1117/12.3045429.

[29] M. Hamid, B. Alam, and O. Pal, "Comparative study of quantum computing tools and frameworks," *Communications in Computer and Information Science*, vol. 2126, pp. 87–104, 2025, doi: 10.1007/978-3-031-80842-5_8.

[30] M. A. Serrano, J. A. Cruz-Lemus, R. Perez-Castillo, and M. Piattini, "Quantum software components and platforms: overview and quality assessment," *ACM Computing Surveys*, vol. 55, no. 8, 2023, doi: 10.1145/3548679.

BIOGRAPHIES OF AUTHORS

Muhammad Hamid received his first degree from Uttar Pradesh Technical University, Information Technology, India, in 2013. He has also done Master degree from Jamia Millia Islamia University, in Computer Engineering, New Delhi, India, in 2018. He is a Ph.D. scholar at Jamia Millia Islamia University, New Delhi, India. His main research interests focus on quantum computing, quantum machine learning, artificial intelligence, evolutionary computations, optimization algorithms, computer vision, image processing, natural language processing, and cryptography. He can be contacted at email: hamidmd504@gmail.com.

Bashir Alam is a professor in the Department of Computer Engineering at Jamia Millia Islamia, New Delhi, with over 20 years of teaching experience. He has taught a wide range of subjects, including parallel programming, computer architecture, AI, and embedded systems. Alam holds a Ph.D. in Computer Engineering and has published 39 journal papers and 25 international conference papers. His research areas include high-performance computing, cloud database systems, routing protocols, and cryptography. He has also been awarded multiple research grants and holds patents related to multimedia services and secure group communication. Alam supervises Ph.D. students and has a strong academic and research portfolio. He can be contacted at email: balam2@jmi.ac.in.

Om Pal is an Associate Professor at the University of Delhi, holds a B.E. in Computer Science, an MBA, an MS in Cryptography from IIT Bombay, and a Ph.D. in Cyber Security from Jamia Millia Islamia. With over 20 years of academic and research experience, he has worked on national projects in cryptography, cybersecurity, blockchain, quantum computing, and cyber law. He previously worked at MeitY and C-DAC. Dr. Pal has published numerous papers, holds two patents, and authored books on blockchain and cybersecurity. He is a certified ethical hacker and AICTE-C-DAC certified 'Master Trainer' in high-performance computing. He can be contacted at email: opal@cs.du.ac.in.