A Case Study of Selected Quantum Classifiers

Tonni Das Jui[1], Olawale Ayoade[2]
[1]Baylor University
[2]Baylor University

Abstract

In recent years, researchers are looking into data transformations in the quantum information space to see whether they may improve robustness and performance. The evolution of quantum mechanics occurred because it could explain specific scenarios in which conventional formulas failed. As a result, it began to expand in analytical research domains such as machine learning, and it is now capable of functioning correctly, and in some circumstances better than classical machine learning. Classification is one of the crucial areas in Machine Learning (ML), and quantum classification analysis has started to gain prominence. In this paper, we focus on implementing four classification algorithms such as Support Vector Classification with Quantum Kernel (SVCQK), Quantum Support Vector Classifier (QSVC), Variational Quantum Classifier (VQC), and Circuit Quantum Neural Network Classifier (CQNNC). We also provide analytical results of case studies with various generated classifiable and semi-classifiable datasets. This study is to determine whether quantum information theory may shorten learning time or improve convergence when compared to traditional approaches.

Keywords— case study, quantum classifier, support vector classification with quantum kernel, quantum support vector classifier, variational quantum classifier, circuit quantum neural network classifier, qiskit, bernstein-vajirani

Introduction

The manipulation of quantum systems to process information is featured as Quantum Computing. One distinct specialty of quantum computation is the ability to have a quantum state in superposition. It explains the significant calculation speedups [1]. In machine learning, quantum computing is being utilized to compare the performance of algorithms that employ quantum circuits to methods that use classical computation. Quantum simulators in classical machines produces better results than conventional techniques in some cases. In a black box system, for example, predicting a binary value of `n` bits in classical computing will take n steps in the best-case scenario. The steps can no longer be decreased. However, quantum mechanics uses qbits to store information and can properly predict the number in one step. One such algorithm is Bernstein-Vazirani [2]. It outperforms both classical computing costs and accuracy [3, 4]. This intrigues the interest of many researchers, who are considering using quantum classifiers instead of traditional ones and evaluating the results. Different types of classifiers employ various approaches, such as the Support-vector classifier (SVC), which is a supervised learning method that distinguishes between classes by examining the hyperplane [5]. SVC can also be implemented with modified kernel set up [6]. Than there is quantum neural network classifier that uses layer of qubits [7].

In this paper, we selectively choose some quantum algorithms for data classification. Our contribution is highlighted as below:

- We implement SVCQK, QSVC, VQC, and CQNNC on four types of generated datasets.
- We show analytical result of these quantum classifiers on the datasets and discuss performance with respect to overall required training and testing time including accuracy of each classifier on all datasets.
- We show that for both 100% classifiable and semi-classifiable datasets, SVCQK, QSVC and CQNNC gives similar performance in the aspect of accuracy. However, VQC provides a low performance in accuracy compared to the other models.
- We show that SVCQK, QSVC requires similar training and testing time while VQC, CQNNC requires more training time than the other two for the smaller datasets and slightly less training time than the other two for the larger (double of the smaller ones) datasets.
- We further find that, except for VQC, all classifiers take roughly the same prediction time. It takes significantly less time for VQC to test and to evaluate the output.

Apart from these experiments on quantum classifiers, we also implemented the Bernstein-Vazirani algorithm using the Qiskit library to demonstrate the performance speed up in quantum computation over classical approaches.

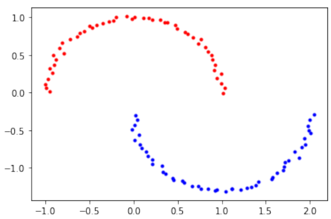
A Quantum computer (QC) is designed to run using quantum mechanics principles. These principles enables a QC to outperform a regular (classical) computer. QC's run on quantum algorithms that makes them attain efficient improvements and speedups over classical algorithms [8]. Shah *et al.* [9] analyzed classifiers on quantum hardware and stated that conventional computers fail to operate on datasets with large dimensional spaces, but quantum computers can efficiently handle those computations. In the same year, Yano *et al.* [10] introduced Quantum Random Access Coding (QRAC) to map these high dimensional discrete features and showed that training time for variational classifier speeds up than in conventional technique. In 2018, a quantum-classical approach was introduced that can perform a k-fold cross-validation with linear speedup and validated these classifications [11]. Quantum computing and kernel methods have an evident similarity in their mathematical frameworks [12]. Diversified and intensive research development has been witnessed in the field of quantum optimization and machine learning with several tools serving as a choice: Variational quantum circuits [13], Quantum neural networks [14, 15], as well as quantum support vector machine for big data classification [16].

Methodology

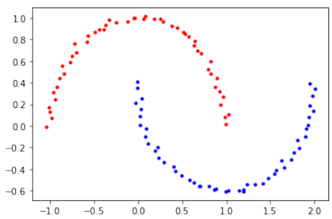
In this section, we have described how our dataset looked like and how we have implemented the above-mentioned algorithms on these datasets. We have divided this into two subsections as below.

Generating Dataset

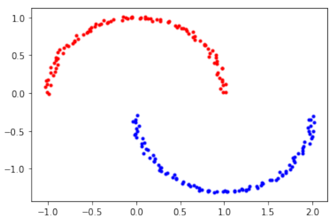
We generated four datasets of semi-circles for our experiments. A graphical representation of those datasets in Figure 1 shows how the data points are spread. The first dataset is of size 100 and so is the second one, but the first one is fully separable (two-class data points can be divided by a linear classifier) and the second dataset is semi-classifiable (50% of the data points are merged). The third and fourth datasets are of size 200. However, the third one is fully separable and the fourth one is not.



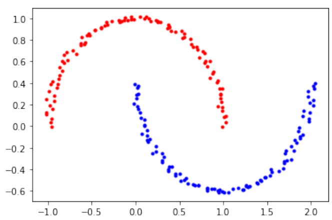
(a) Dataset 1



(b) dataset 2



(c) Dataset 3



(d) Dataset 4

Figure 1: Graphical representation of the four datasets

Classification

In this part we have described how we have conducted the case study. We implemented all four algorithms namely SVCQK, QSVC, VQC, and QNN. For implementation purpose, we used the Qiskit library of version 0.15.0 for quantum implementation, scikit-learn for the algorithms on classical dataset and matplotlib for graphical representation. We generated a feature_map which is the function that does the conversion of classical features into quantum qbits. It is possible to convert higher number of classical features of the dataset into any lower number that is greater or equal to two qubit features. However, it is not possible to convert from lower classical features numbers to higher number of qubit feature. We used the feature_map and quantum backend as the parameter to the kernel for SVCQK, VQC,and QNN. For QSVC, a quantum kernel is prepared by the Qiskit library itself.

Our open-source implementation of the Bernstein-Vazirani algorithm and four quantum classifiers can be found online at `https://github.com/tonnidas/Quantum-Classifiers`.

Result Analysis

Our results of this experiment is represented in the Table 1. The data column provides the sets of data namely dataset1, dataset2, dataset3, and dataset4. Dataset1 and dataset2 represent semi circle dataset of size 100 where dataset1 is linearly separable but dataset2 is not. Dataset3 and dataset4 represent 200 data points each. However, likewise dataset1 and dataset2, one of them is linearly separable and the other is not. The first row of the table represents the datasets while the first column represents the algorithms. There are three types of results for each experiment which indicates training time, prediction time and accuracy for each algorithm against each dataset. Prediction in classical machine learning takes minimal time after training, however it is not immediate in quantum machine learning since we utilize a quantum simulator in a classical computer. When quantum computers become available in a few years, they will most likely be extremely fast.

Table 1: Four algorithm performance results on four datasets

Algorithm	Result	Dataset 1	Dataset 2	Dataset 3	Dataset 4
SVCQK	Train	83.81	83.01	336.51	315.83
	Predict	42.55	42.52	168.77	160.01
	Accuracy	80	85	80	62.5
QSVC	Train	84.39	83.17	335.59	320.85
	Predict	40.98	42.39	169.01	159.73
	Accuracy	80	85	80	62.5
VQC	Train	166.19	154.16	268.37	284.70
	Predict	0.58	0.60	1.18	1.13
	Accuracy	30	30	25	22.5
QNN	Train	190.04	146.43	324.85	311.16
	Predict	42.56	42.79	164.42	158.70
	Accuracy	80	85	80	62.5

From Table 1, we can see that SVCQK, QSVC, and QNN outperform VQC for all of these datasets, despite the fact that its testing time is much less. Another noteworthy observation is that when the dataset size is doubled, training and testing time nearly triples for both linearly separable and non-separable datasets, while performance accuracy remains the same for linearly separable datasets and falls for non-separable datasets.

Conclusion

We conclude by summarizing five key facts about our project work. So, the key facts that we are summarizing in this section is about the problem that we were trying to solve, how we have solved it, what the result is, what is the intellectual merit and lastly, what is the broader impact. We focused on the problem of discovering what is the relationship between quantum classifiers applying in classical datasets and their performance on these different datasets. we have done it by implementing four quantum classifiers (SVCQK, QSVC, VQC, QNN) on four simulated datasets. The result of our analysis is that the performance of VQC is comparatively low than others and in larger and linearly separable datasets, the training and testing time increases; however, performance of accuracy stays the same. Accuracy also drops slightly than in smaller dataset when implementing it on linearly inseparable datasets. We also have implemented Bernstein-Vazirani algorithm on the test set and got the right sequence in one guess. The intellectual merit of the project is there are less works regarding quantum classifier analysis. We have found how variational quantum classifier acts differently than other quantum classifiers from our study which is the broader impact of the study.

References

[1] M. Schuld, I. Sinayskiy, and F. Petruccione, “An introduction to quantum machine learning,” *Contemporary Physics*, vol. 56, no. 2, pp. 172–185, 2015.

[2] C. D. H. S D Fallek, “Transport implementation of the Bernstein–Vazirani algorithm with ion qubits,” *New Journal of Physics*, vol. 18, 2016.

[3] T. Patel and D. Tiwari, “Veritas: accurately estimating the correct output on noisy intermediate-scale quantum computers,” in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, no. 15, pp. 1–16, IEEE Press, Nov. 2020.

[4] J. Liu, G. T. Byrd, and H. Zhou, “Quantum Circuits for Dynamic Runtime Assertions in Quantum Computation,” in *Proceedings of the Twenty-Fifth International Conference on Architectural Support for Programming Languages and Operating Systems*, ASPLOS ’20, (Lausanne, Switzerland), pp. 1017–1030, Association for Computing Machinery, Mar. 2020.

[5] Q. Vinlove, J. Mache, and R. Weiss, “Predicting student success in cybersecurity exercises with a support vector classifier,” *Journal of Computing Sciences in Colleges*, vol. 36, pp. 26–34, Oct. 2020.

[6] D. Loiacono, A. Marelli, and P. L. Lanzi, “Support vector regression for classifier prediction,” in *Proceedings of the 9th annual conference on Genetic and evolutionary computation*, GECCO ’07, (London, England), pp. 1806–1813, Association for Computing Machinery, July 2007.

[7] M. Panella and G. Martinelli, “Neural networks with quantum architecture and quantum learning,” *International Journal of Circuit Theory and Applications*, vol. 39, pp. 61–77, 2011.

[8] A. Montanaro, “Quantum algorithms: an overview,” *npj Quantum Information*, vol. 2, no. 1, pp. 1–8, 2016.

[9] A. Shah, M. Shah, and P. Kanani, “Leveraging Quantum Computing for Supervised Classification,” in *2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS)*, pp. 256–261, May 2020.

[10] H. Yano, Y. Suzuki, R. Raymond, and N. Yamamoto, “Efficient Discrete Feature Encoding for Variational Quantum Classifier,” in *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)*, pp. 11–21, Oct. 2020.

[11] P. G. dos Santos, I. C. Araujo, R. S. Sousa, and A. J. da Silva, “Quantum Enhanced k-fold Cross-Validation,” in *2018 7th Brazilian Conference on Intelligent Systems (BRACIS)*, pp. 194–199, Oct. 2018.

[12] M. Schuld, “Quantum machine learning models are kernel methods,” *arXiv e-prints*, pp. arXiv–2101, 2021.

[13] M. Schuld, A. Bocharov, K. M. Svore, and N. Wiebe, “Circuit-centric quantum classifiers,” *Physical Review A*, vol. 101, no. 3, p. 032308, 2020.

[14] N. Killoran, T. R. Bromley, J. M. Arrazola, M. Schuld, N. Quesada, and S. Lloyd, “Continuous-variable quantum neural networks,” *Physical Review Research*, vol. 1, no. 3, p. 033063, 2019.

[15] I. Kerenidis, J. Landman, and A. Prakash, “Quantum algorithms for deep convolutional neural networks,” *arXiv preprint arXiv:1911.01117*, 2019.

[16] P. Rebentrost, M. Mohseni, and S. Lloyd, “Quantum support vector machine for big data classification,” *Physical review letters*, vol. 113, no. 13, p. 130503, 2014.