

A new method of constructing adversarial examples for quantum variational circuits

Jinge Yan(颜金歌), Lili Yan(闫丽丽), and Shibin Zhang(张仕斌)

Citation: Chin. Phys. B, 2023, 32 (**7**): 070304. DOI: 10.1088/1674-1056/ac9b32
Journal homepage: http://cpb.iphy.ac.cn; http://iopscience.iop.org/cpb

What follows is a list of articles you may be interested in

High-fidelity universal quantum gates for hybrid systems via the practical photon scattering

Jun-Wen Luo(罗竣文) and Guan-Yu Wang(王冠玉)
Chin. Phys. B, 2023, 32 (**3**): 030303. DOI: 10.1088/1674-1056/ac8734

Variational quantum simulation of thermal statistical states on a superconducting quantum processer

Xue-Yi Guo(郭学仪), Shang-Shu Li(李尚书), Xiao Xiao(效骁), Zhong-Cheng Xiang(相忠诚), Zi-Yong Ge(葛自勇), He-Kang Li(李贺康), Peng-Tao Song(宋鹏涛), Yi Peng(彭益), Zhan Wang(王战), Kai Xu(许凯), Pan Zhang(张潘), Lei Wang(王磊), Dong-Ning Zheng(郑东宁), and Heng Fan(范桁)
Chin. Phys. B, 2023, 32 (**1**): 010307. DOI: 10.1088/1674-1056/aca7f3

Quantum algorithm for neighborhood preserving embedding

Shi-Jie Pan(潘世杰), Lin-Chun Wan(万林春), Hai-Ling Liu(刘海玲), Yu-Sen Wu(吴宇森), Su-Juan Qin(秦素娟), Qiao-Yan Wen(温巧燕), and Fei Gao(高飞)
Chin. Phys. B, 2022, 31 (**6**): 060304. DOI: 10.1088/1674-1056/ac523a

Filling up complex spectral regions through non-Hermitian disordered chains

Hui Jiang and Ching Hua Lee
Chin. Phys. B, 2022, 31 (**5**): 050307. DOI: 10.1088/1674-1056/ac4a73

Quantum partial least squares regression algorithm for multiple correlation problem

Yan-Yan Hou(侯艳艳), Jian Li(李剑), Xiu-Bo Chen(陈秀波), and Yuan Tian(田源)
Chin. Phys. B, 2022, 31 (**3**): 030304. DOI: 10.1088/1674-1056/ac1b84

A new method of constructing adversarial examples for quantum variational circuits

Jinge Yan(颜金歌)[1,2], Lili Yan(闫丽丽)[1,2,†], and Shibin Zhang(张仕斌)[1,2]

[1] *School of Cybersecurity, Chengdu University of Information Technology, Sichuan* 610000, *China*
[2] *Advanced Cryptography and System Security Key Laboratory of Sichuan Province, Sichuan* 610000, *China*

(Received 27 June 2022; revised manuscript received 24 September 2022; accepted manuscript online 19 October 2022)

A quantum variational circuit is a quantum machine learning model similar to a neural network. A crafted adversarial example can lead to incorrect results for the model. Using adversarial examples to train the model will greatly improve its robustness. The existing method is to use automatic differentials or finite difference to obtain a gradient and use it to construct adversarial examples. This paper proposes an innovative method for constructing adversarial examples of quantum variational circuits. In this method, the gradient can be obtained by measuring the expected value of a quantum bit respectively in a series quantum circuit. This method can be used to construct the adversarial examples for a quantum variational circuit classifier. The implementation results prove the effectiveness of the proposed method. Compared with the existing method, our method requires fewer resources and is more efficient.

Keywords: quantum variational circuit, adversarial examples, quantum machine learning, quantum circuit

PACS: 03.67.–a, 03.67.Ac, 03.67.Lx **DOI:** 10.1088/1674-1056/ac9b32

1. Introduction

Quantum physics and machine learning are currently a focus of research, and based on these a new research field called quantum machine learning has been proposed.[1,2] Machine learning has developed rapidly over the past 20 years. It has solved many challenging problems and effectively promoted the development of artificial intelligence.[3–5] However, some machine learning algorithms are too complex to be implemented on classical computers in a reasonable time.[6] In addition, with the development of machine learning the network structure, parameters and algorithms of machine learning models are becoming larger and more complex. Generally, accurate models can only be trained with the support of a large number of data and calculations, which requires a lot of resources. Therefore, the development of machine learning is limited by the classical computer. Fortunately, the development of quantum computing has solved the above problems faced by machine learning.[7,8] Quantum computers can reduce the time required for large and complex machine learning algorithms. Quantum machine learning has achieved rapid progress in the past 10 years. Based on quantum properties such as quantum superposition, quantum entanglement and quantum measurement, many quantum machine learning algorithms have been proposed,[9,10] for example, Shor, Grover and HHL.[11,12] Quantum walks are another recent example.[13] Researchers have also designed some quantum machine learning algorithms based on classical machine learning algorithms, such as quantum support vector machine[14–16] and quantum principal component analysis.[17] Compared with classical algorithms, these algorithms achieve exponential acceleration. Quantum neural networks are an important branch of quantum machine learning. There are many kinds of quantum neural networks.[18–20] Most of them are constructed using quantum circuits, and the function of the neuron is replaced by quantum gates, with the angles of the quantum gates being the neural parameters. They take quantum states as input data, some take quantum states as output and some take the probabilities of measuring quantum bits as output.[21,22] Recently researchers have found that parameterized quantum circuits, i.e., quantum variational circuits, can be regarded as a kind of quantum neural network.[23] With the development of quantum hardware and corresponding software, quantum variational circuits will have broad application prospects.[24–27]

In classical machine learning it has been found that machine learning models are vulnerable. Adding a small and imperceptible perturbation to the input of a machine learning model may cause incorrect output of the model. In 2004, Dalvi *et al.* found that linear classifiers are prone to errors in classification results caused by some carefully constructed perturbations.[28] Therefore, how to construct such perturbations and how to make the machine learning model highly robust against such perturbations have become new topics in machine learning research. This has given rise to the emerging field of adversarial machine learning. Szegedy *et al.* found an obvious example of perturbation in a deep learning environment for the first time, proving the vulnerability of machine learning. They took an image of a panda as the input of a machine learning model and added a small perturbation (im-

†Corresponding author. E-mail: yanlili@cuit.edu.cn

© 2023 Chinese Physical Society and IOP Publishing Ltd

perceptible to the human eye) to it, and the model mistakenly identified it as a gibbon.[29–31] In quantum machine learning it has also been found that the quantum machine learning model is vulnerable to attack by a small imperceptible perturbation which makes the model output incorrect.[32] The example after the addition of a perturbation is called the adversarial example, and the key point in constructing adversarial examples is to obtain the gradient of the loss function with respect to features $\boldsymbol{x}$ of the examples. Sirui *et al.* obtained this gradient by an automatic differentiation method[33] in a classical computer, and they provide a thorough investigation of adversarial examples and attacks on quantum machine learning models.[34]

In this paper, we propose an innovative method to obtain the gradient, which is realized by quantum circuits in a quantum computer. Firstly, a series of quantum circuits are constructed according to the number of gradient components, and the measurement expectation of a qubit in each quantum circuit is obtained. Finally, the gradient is obtained according to the expectation. Analysis shows that this method is better than the existing automatic differential method and finite difference method.

2. Existing research

Suppose there is a trained quantum machine learning model; it is necessary to encode N features of normalized data $\boldsymbol{x} = (x_1, x_2, \ldots, x_N)$ into a quantum state $|\varphi(\boldsymbol{x})\rangle = \sum_{i=1}^{N} x_i |i\rangle$ by amplitude encoding,[35] taking $|\varphi(\boldsymbol{x})\rangle$ as its input. Its loss function $L(|\varphi(\boldsymbol{x})\rangle, \theta, y)$ is determined by its trained parameters θ and the model input $\boldsymbol{x}$ and label y. To construct a suitable adversarial example, we only need to search for a small perturbation $\boldsymbol{\delta}$ within an appropriate region $\boldsymbol{\Delta}$ and add it to $|\varphi(\boldsymbol{x})\rangle$, which makes the value of loss function as large as possible so that the output of the model can deviate from the label y:

$$\boldsymbol{\delta} = \underset{\boldsymbol{\delta}\in\boldsymbol{\Delta}}{\operatorname{argmax}} L(|\varphi(\boldsymbol{x}+\boldsymbol{\delta})\rangle, \theta, y). \tag{1}$$

We add the perturbation $\boldsymbol{\delta}$ to $\boldsymbol{x}$ then the adversarial example

$$|\varphi(\boldsymbol{x}')\rangle = |\varphi(\boldsymbol{x}+\boldsymbol{\delta})\rangle \tag{2}$$

can be obtained. Specifically, the exact perturbation $\boldsymbol{\delta}$ can be obtained by evaluating the gradient of the loss function with respect to $\boldsymbol{x}$ and multiplying it by a perturbation factor ε; ε determines the magnitude of perturbation

$$\boldsymbol{\delta} = \varepsilon \nabla_{\boldsymbol{x}} L(|\varphi(\boldsymbol{x})\rangle, \theta, y). \tag{3}$$

A quantum variational circuit can also be regarded as a quantum machine learning model, it consists of three parts:

(i) A prepared quantum state $|\varphi(\boldsymbol{x})\rangle$ as its input.

(ii) A quantum circuit $U(\theta)$, which is parameterized by a set of free parameters θ as its model.

(iii) Measurement of an observable M at the output as its output; this observable may be made up from local observables for each wire in the circuit, or just a subset of wires. It can be written as $\langle\varphi(\boldsymbol{x})|U(\theta)^{\dagger}MU(\theta)|\varphi(\boldsymbol{x})\rangle$.

A simple example of a quantum variational circuit can be seen in Fig. 1. It takes the expectation of a Pauli-Z based measurement of just one wire as output and such a model can be a classifier.

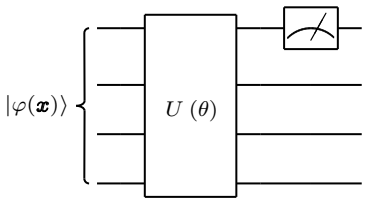


Fig. 1. A simple example of quantum variational circuit.

To train such a quantum variational circuit, we need to evaluate the gradient of the loss function with respect to parameters θ, and the parameter-shift method[36] is usually used for this:

$$\frac{\partial L(|\varphi(\boldsymbol{x})\rangle, \theta, y)}{\partial \vartheta} = \frac{1}{2}\Big(L(|\varphi(\boldsymbol{x})\rangle, \theta, y)_{\vartheta+\frac{\pi}{2}} - L(|\varphi(\boldsymbol{x})\rangle, \theta, y)_{\vartheta-\frac{\pi}{2}}\Big), \tag{4}$$

where $L(|\varphi(\boldsymbol{x})\rangle, \theta, y)$ is the loss function, ϑ is a single parameter of parameters θ, $|\varphi(\boldsymbol{x})\rangle$ is the input and y is the label. We assume that the parameters of the quantum variational circuit have been trained well. To construct adversarial examples, we only need to define a perturbation factor ε, then evaluate the gradients of the loss function with respect to $\boldsymbol{x}$ according to Eq. (3) to obtain $\boldsymbol{\delta}$ and then add it to $|\varphi(\boldsymbol{x})\rangle$ to get the adversarial example $|\varphi(\boldsymbol{x}')\rangle$. Existing research[34] on evaluating the gradient of the loss function with respect to $\boldsymbol{x}$ uses the automatic differentiation method,[33] which is a method of calculating the gradient in classical computers by differentiating.

3. The proposed method

In this section, we propose a new method to evaluate the gradient of the loss function with respect to $\boldsymbol{x}$ for quantum variational circuits and compare it with existing methods.

Suppose there is a trained quantum variational circuit. It takes $|\varphi(\boldsymbol{x})\rangle = \sum_{i=1}^{N} x_i |i\rangle$ as input and the output $y' = \langle\varphi(\boldsymbol{x})|U(\theta)^{\dagger}MU(\theta)|\varphi(\boldsymbol{x})\rangle$, so its loss function $\mathcal{L}$ can be written as

$$\mathcal{L} = \frac{1}{2}|y' - y|^2, \tag{5}$$

where y is the label of the input data $\boldsymbol{x}$. The gradient of $\mathcal{L}$ with respect to $\boldsymbol{x}$ is $\nabla_{\boldsymbol{x}}\mathcal{L} = (\partial\mathcal{L}/\partial x_1, \partial\mathcal{L}/\partial x_2, \ldots, \partial\mathcal{L}/\partial x_N)$, and each component of the vector is $\partial\mathcal{L}/\partial x_i = (y' - y)\,\kappa_i$, where $i = 1, 2, \ldots, N$ and κ_i is

$$\kappa_i = \langle A|M|B\rangle + \langle B|M|A\rangle, \tag{6}$$

where $|A\rangle = U(\theta)\partial|\varphi(\boldsymbol{x})\rangle/\partial x_i$ and $|B\rangle = U(\theta)|\varphi(\boldsymbol{x})\rangle$. Obviously the key point to getting $\partial\mathcal{L}/\partial x_i$ is to obtain κ_i. Next, we will introduce a method to obtain κ_i.

Firstly, n qubits $|0\rangle$ called the 'data register' and one $|0\rangle$ called the 'ancilla qubit' are prepared, where $n = \log_2 N$. Next, we operate a Hadamard gate on the ancilla qubit $|0\rangle$ to get $\frac{\sqrt{2}}{2}(|0\rangle + |1\rangle)$. We define a series of operators as C; this can convert n qubits $|0\rangle$ to $|\varphi(\boldsymbol{x})\rangle$ by amplitude encoding. Then we take the ancilla qubit as the control qubit to operate the operator C on the data register. The state of the circuit becomes

$$\frac{\sqrt{2}}{2}\left(|0\rangle|0\rangle^{\otimes n} + |1\rangle|\varphi(\boldsymbol{x})\rangle\right). \tag{7}$$

Next, we define a series of operators as F. This can convert n qubits $|0\rangle$ to $\partial|\varphi(\boldsymbol{x})\rangle/\partial x_i$, and this is easy to do. For example, suppose that $|\varphi(\boldsymbol{x})\rangle = x_1|0\rangle + x_2|1\rangle + x_3|2\rangle + x_4|3\rangle$, so $\partial|\varphi(\boldsymbol{x})\rangle/\partial x_i = |2_{\mathrm{d}}\rangle = |10\rangle$ where $i = 3$, d means decimal and if there is no subscript it means binary. Obviously, it is easy to convert $|00\rangle$ to $|10\rangle$ — we just need an X gate. The example goes the same way for F. We operate a X gate on the ancilla qubit, and then take it as the control qubit to operate F on the data register. We finally operate a X gate on the ancilla qubit, and the state of the circuit becomes

$$\frac{\sqrt{2}}{2}\left(|0\rangle\frac{\partial|\varphi(\boldsymbol{x})\rangle}{\partial x_i} + |1\rangle|\varphi(\boldsymbol{x})\rangle\right). \tag{8}$$

Next, we operate $U(\theta)$ of the quantum variational circuit on the data register, and the state of the circuit becomes

$$\frac{\sqrt{2}}{2}\left(|0\rangle|A\rangle + |1\rangle|B\rangle\right). \tag{9}$$

Then, according to the observable M chosen from the quantum variational circuit, we take the ancilla qubit as the control qubit and operate Pauli-M operators on the data register. The state of the circuit becomes

$$\frac{\sqrt{2}}{2}\left(|0\rangle|A\rangle + |1\rangle M|B\rangle\right). \tag{10}$$

We then operate a Hadamard gate on the ancilla qubit. The state of the circuit becomes

$$\frac{\sqrt{2}}{2}\left(|0\rangle\left(|A\rangle + M|B\rangle\right) + |1\rangle\left(|A\rangle - M|B\rangle\right)\right). \tag{11}$$

Finally, we obtain the expectation value p of the ancilla qubit in the Pauli-Z basis

$$p = \langle A|M|B\rangle + \langle B|M|A\rangle. \tag{12}$$

According to Eq. (6), p is equal to κ_i, which is the key point to get $\partial\mathcal{L}/\partial x_i$, so using the method we proposed $\partial\mathcal{L}/\partial x_i$ can be evaluated by one measurement in a quantum circuit. It can also be evaluated by finite difference, but this requires two measurements and is just an approximation. It can also be evaluated by automatic differentiation, and requires only one calculation, but this consumes more resources. The specific comparison is shown in Table 1.

Table 1. Method comparison.

Method	Resource consumption	Number of calculations	Hardware
Our method	$O(\log N)$	one measurement	quantum computer
Infinite difference	$O(\log N)$	two measurements	quantum computer
Automatic differentiation	$O(N)$	one calculation	classical computer

The analysis shows that the proposed method requires fewer resources and is more efficient.

4. Implementation

In this section, we construct a quantum variational circuit classifier model by using the PennyLane framework.[37] We use the MNIST handwritten digital classification dataset[38] as the training set and verification set for our model; this is widely considered as the real testbed for new machine learning paradigms. The MNIST dataset consists of hand-drawn digits from 0 to 9 in the form of gray-scale images. Each image contains 28×28 pixels, or in other words 784 features. Each feature is a integer ranging from 0 to 255 with 0 meaning the whitest and 255 the darkest color. To fit our model, a principal component analysis dimensionality reduction algorithm is used to reduce the image features from $28 \times 28 = 784$ to $16 \times 16 = 256$, so that the features $\boldsymbol{x}$ of one image can be encoded to a quantum state $|\varphi(\boldsymbol{x})\rangle$ with 8 qubits by amplitude encoding ($2^8 = 256$).

The measurement of an observable M chosen from our model here is a Pauli-Z basis measurement in one wire; such a model can be a two-category classifier. The quantum variational circuit classifier model is shown in Fig. 2.

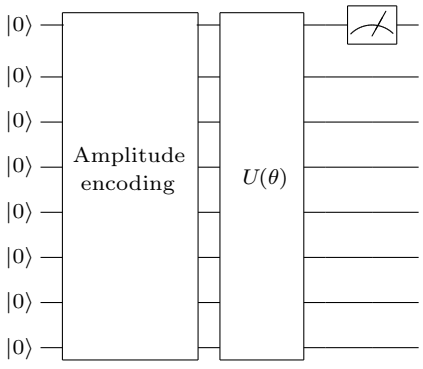


Fig. 2. The quantum variational circuit classifier. The depth of $U(\theta)$ is 10, and the "amplitude encoding" can convert $|0\rangle^{\otimes 8}$ to $|\varphi(\boldsymbol{x})\rangle$.

We select all images labeled 0 and 1 from the MNIST data set and choose Eq. (5) as the loss function. To train the classifier, we use the Nesterov momentum optimizer with a batch size of 10 and a learning rate of 0.05 to minimize the loss function. The average accuracy and loss as a function of the number of training steps can be seen in Fig. 3.

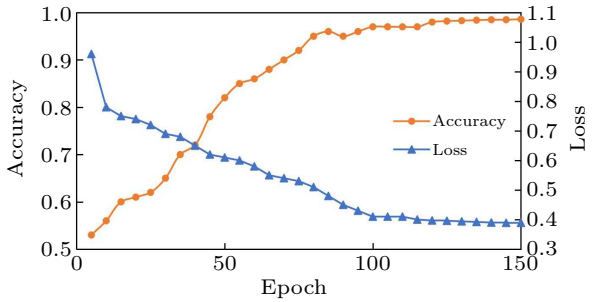


Fig. 3. The average loss and accuracy as a function of the number of training steps.

Next, we select two images labeled 0 and two images labeled 1, respectively, and they can be correctly classified by our model, as shown in Fig. 4.

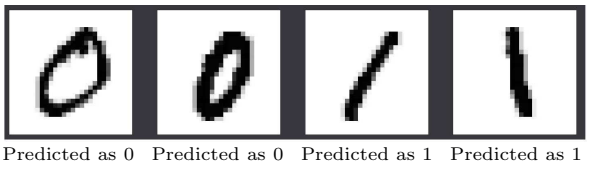


Fig. 4. Images correctly classified by our model.

According to the method proposed in Section 3, we construct the corresponding quantum circuit to get κ_i, which can be seen in Fig. 5, evaluate the gradient of the loss function of the four images above with respect to $\boldsymbol{x}$, then define the value of ε respectively, construct the adversarial examples of these four images in Fig. 4 according to Eqs. (2) and (3).

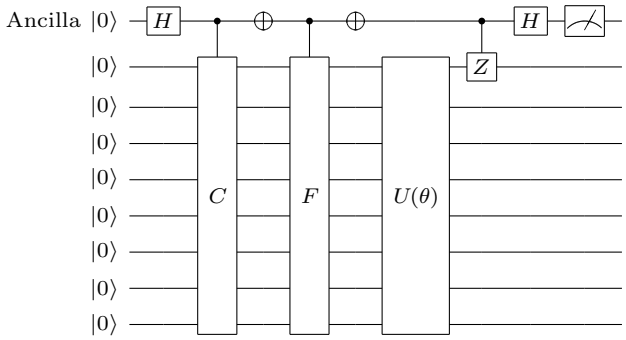


Fig. 5. The quantum circuit of getting κ_i , because the observable M chosen from our model is a Pauli-Z basis measurement in one wire, there is a control-Z gate on it.

We take the adversarial examples as the input of the model but the output of the model does not match their label, as shown in Fig. 6.

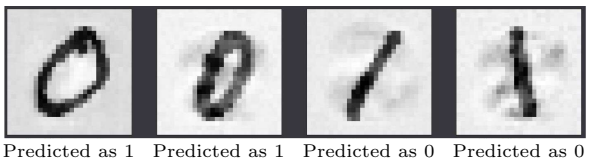


Fig. 6. The adversarial examples.

Next, we construct adversarial examples for all the images labeled 0 and 1.

Although malicious attackers can attack some quantum variational circuit models by using adversarial samples, the robustness of the model can also be enhanced by using adversarial examples to train the model. Due to different images the ε needed for constructing the adversarial examples may not be at the same scale, so for measuring the difference between the legitimate and adversarial images we define fidelity between quantum states that encode the legitimate and adversarial examples: $|\langle\boldsymbol{\varphi}(\boldsymbol{x}')|\boldsymbol{\varphi}(\boldsymbol{x})\rangle|^2$. In order to give a positive role to adversarial examples, we use adversarial examples with a fidelity of 0.9 to continue training the model, and we use the Nesterov momentum optimizer with a batch size of 10 and a learning rate of 0.05 to minimize the loss function. The average accuracy and loss as a function of the number of training steps can be seen in Fig. 7.

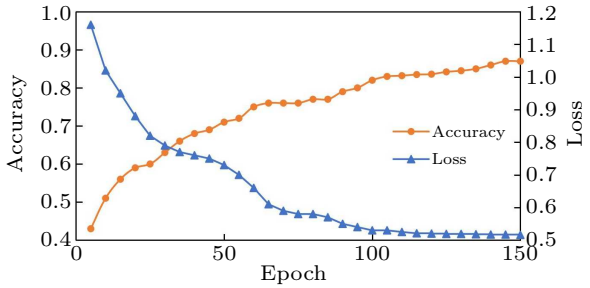


Fig. 7. The average accuracy and loss as a function of the number of training steps when using adversarial examples with a fidelity of 0.9 as the training data.

Then, we compare the present model with the model that has not been trained with adversarial examples. The average accuracy as a function of the fidelity can be seen in Fig. 8.

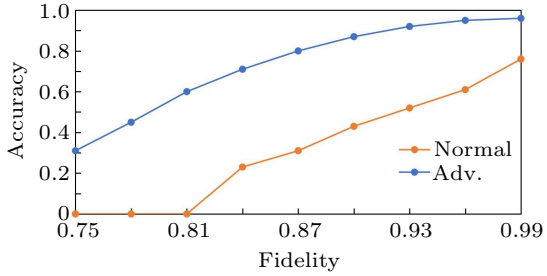


Fig. 8. The average accuracy as a function of the fidelity. "Adv." refers to the model has been trained with adversarial examples, "normal" refers to the model that has not been trained with adversarial examples.

It is obvious that the 'Adv.' model has higher accuracy for adversarial examples with lower fidelity, which means that the model trained with adversarial examples is more robust.

5. Conclusion

In summary, we propose a method to calculate the gradient of loss function of a quantum variational circuit with respect to $\boldsymbol{x}$ by quantum circuits, which is a key point to construct adversarial examples. Compared with the automatic differentiation method and finite difference method, our method has advantages in efficiency and resource consumption. We

use the MNIST handwritten digit classification dataset to train a two-category quantum variational circuit classifier, and use the method proposed to calculate the gradient of loss function with respect to $\boldsymbol{x}$ to construct adversarial examples. These adversarial examples are predicted wrongly by the classifier. We use the adversarial examples as a new dataset to train the model, and the new model is more robust than the model without the adversarial examples training.

Acknowledgments

Project supported by the National Natural Science Foundation of China (Grant Nos. 62076042 and 62102049), the Natural Science Foundation of Sichuan Province (Grant No. 2022NSFSC0535), the Key Research and Development Project of Sichuan Province (Grant Nos. 2021YFSY0012 and 2021YFG0332), the Key Research and Development Project of Chengdu (Grant No. 2021-YF05-02424-GX), and the Innovation Team of Quantum Security Communication of Sichuan Province (Grant No. 17TD0009).

References

[1] Biamonte J, Wittek P, Pancotti N, Rebentrost P, Wiebe N and Lloyd S 2017 *Nature* **549** 195
[2] Sarma D S, Deng D L and Duan L M 2019 *Phys. Today* **72** 48
[3] Jordan M I and Mitchell T M 2015 *Science* **349** 255
[4] LeCun Y, Bengio Y and Hinton G 2015 *Nature* **521** 436
[5] Chang C C and Lin C J *ACM Trans. Intell. Syst. Technol.* **2** 1
[6] Ji A B, Pang J H and Qiu H J 2010 *Expert. Syst. Appl.* **37** 3495
[7] Knill E, Laflamme R and Milburn G J 2001 *Nature.* **409** 46
[8] Wiebe N, Braun D and Lloyd S 2012 *Phys. Rev. Lett.* **109** 050505
[9] Torlai G, Mazzola G, Carrasquilla J, Troyer M, Melko R and Carleo G 2018 *Nat. Phys.* **14** 447
[10] Dallaire-Demers P L and Killoran N 2018 *Phys. Rev. A* **98** 012324
[11] Harrow A W, Hassidim A and Lloyd S 2009 *Phys. Rev. Lett.* **103** 150502
[12] Sun J and Lu S F 2020 *Chin. Phys. B* **29** 100303
[13] Meng Y, Mei F, Chen G and Jia S T 2020 *Chin. Phys. B* **29** 070501
[14] Rebentrost P, Mohseni M and Lloyd S 2014 *Phys. Rev. Lett.* **113** 130503
[15] Li Z K, Liu X M, Xu N Y and Du J F 2015 *Phys. Rev. Lett.* **114** 140504
[16] Havlíček V, Córcoles A D, Temme K, Harrow A W, Kandala A, Chow J M and Gambetta J M 2019 *Nature* **567** 209
[17] Cong I and Duan L M 2016 *New J. Phys.* **18** 073011
[18] Schmidhuber J 2015 *Neural Networks* **61** 85
[19] Liu J, Lim K H, Wood K L, Huang W, Guo C and Huang H L 2021 *Sci. China Phys. Mech.* **64** 290311
[20] Cong I, Choi S and Lukin M D 2019 *Nat. Phys.* **15** 1273
[21] Shafee F 2007 *Eng. Appl. Artif. Intel.* **20** 429
[22] Li P C and Li S Y 2008 *J. Syst. Eng. Electron.* **19** 167
[23] Benedetti M, Lloyd E, Sack S and Fiorentini M 2019 *Quantum Sci. Technol.* **4** 019601
[24] Yu X M, Tan X S, Yu H F and Yu Y 2018 *Acta Phys. Sin* **67** 220302 (in Chinese)
[25] Li X Q, Zhao Y F, Tang Y N and Yang W J 2018 *Acta Phys. Sin* **67** 070302 (in Chinese)
[26] Huang H L, Du Y X, Gong M, *et al.* 2021 *Phys. Rev. Appl.* **16** 024051
[27] Harrigan M P, Sung K J, Neeley M, *et al.* 2021 *Nat. Phys.* **17** 332
[28] Nilesh D, Pedro D M, Sumit S, *et al.* 2004 *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining,* August 22–25, 2004, Seattle, USA, p. 99
[29] Szegedy C, Zaremba W, Sutskever I, Bruna J, Erhan D, Goodfellow I and Fergus R 2014 arXiv: 1312.6199
[30] Miyato T, Andrew M and Goodfellow I 2021 arxiv: 1605.07725
[31] GoodFellow I, Shlens J and Szegedy C 2015 arxiv: 1412.6572
[32] Liu N N and Wittek P 2020 *Phys. Rev. A* **101** 062331
[33] Neidinger R 2010 *SIAM Rev.* **52** 545
[34] Lu S, Duan L M and Deng D L 2020 *Phys. Rev. Res.* **2** 033212
[35] Schuld M and Petruccione F 2018 *Supervised Learning with Quantum Computers* pp. 1–287
[36] Mitarai K, Negoro M, Kitagawa and Fujii K 2018 *Phys. Rev. A* **98** 032309
[37] https://github.com/PennyLaneAI/pennylane
[38] http://yann.lecun.com/exdb/mnist/