International Journal of Theoretical Physics (2019) 58:2331–2340
https://doi.org/10.1007/s10773-019-04124-5

Improved Handwritten Digit Recognition using Quantum K-Nearest Neighbor Algorithm

Yuxiang Wang [1] **· Ruijin Wang** [1] **· Dongfen Li** [2] **· Daniel Adu-Gyamfi** [1] **· Kaibin Tian** [1] **· Yixin Zhu** [3]

Received: 8 January 2019 / Accepted: 13 April 2019 / Published online: 4 May 2019

Abstract

Handwritten numeral recognition is a technology for automatic recognition and classification of handwritten numeral input through machine learning model. This is widely used in postal code digital automatic system to sort letters. The classical k-nearest neighbor algorithm is used in the traditional digital recognition training model. The recognized digital image classification is obtained through similarity measure or calculation and K value selection. Nonetheless, as the applied data volume exceeds a certain threshold, the time complexity of the model increases exponentially upon the similarity measure and K value search. This condition makes it hard to apply the model universally. In this paper, we introduce quantum computing, that is where digital image information is stored in the quantum state, and its similarity is calculated in parallel. Also, the most similar K points are obtained through the Grover algorithm. The theoretical analysis of the proposed improved algorithm shows that, handwritten numeral recognition based on quantum k-neighbor algorithm can improved upon time complexity of $\mathrm{O}\left(\mathrm{R}\sqrt{kM}\right)$ of the existing algorithm.

Keywords Handwritten numeral recognition · Quantum computing · K-nearest neighbor

1 Introduction

Making machines being capable to perform pattern recognition has always been an important direction of computer research. Character recognition is a typical pattern recognition, through which handwritten numeral recognition is derived. In recent years, handwritten numeral

✉ Ruijin Wang
ruijinwang@uestc.edu.cn

[1] School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu 610054, China

[2] Chengdu University of Technology, Chengdu 610059, China

[3] Xinjiang University of Finance & Economics, Ürümqi, China

recognition has gradually gained much attention, and has been widely used in postal code digital automatic system for sorting letters, statistical reports, financial statements, bank notes, and other aspects [1–5].

Handwriting is a common practice globally, and hence, it is important to select a sample from library for systematic training and testing. At present, a relatively large number of handwritten digital samples are available in the following libraries: (1) the NIST database, compiled collected by the national bureau of standards and technology of the United States; (2) the CEDAR database, a zip code sample library prepared by the department of computer science, state university of New York, Buffalo; (3) the ETL database, compiled by Japanese institute of electrical technology; (4) the ITPT database, compiled by Japan post and telecommunications policy research institute [6].

Moreover, the time complexity of traditional algorithm or model becomes very high due to the continuous enrichment of handwritten digital sample library, and the gradual improvement of recognition accuracy [7]. For example, the KNN handwritten numeral classification algorithm is been widely used. However, the traditional model calculation is quite laborious for calculating the similarity of sample data and training set data, and searching for the first K values. At present, machine learning is also facing similar problems of large training data sets and computing bottlenecks [8]. In view of the above problems, this paper proposes a quantum computing based scheme to improve upon the algorithm or model of traditional machine learning [9]. The proposed scheme replace the specific sub-process of the original classical algorithm with that of the quantum algorithm, so as in order to reduce the computational complexity [10].

2 The Handwriting Recognition Algorithm

The core of handwritten numeral recognition algorithm is KNN algorithm [6]. The input of KNN algorithm is feature vector, corresponding points of feature space, and the output is category [11]. The KNN algorithm assumes a set of training data with pre-specified categories. For test dataset, categories of K nearest neighbor training instances are obtained through similarity analysis, and prediction is done by majority voting [12].

Here, we have a bunch of images that have been labeled to represent a number. This is shown in Fig. 1. (This set of images constitutes a training set, also known as a sample space).

For example, if we have a test data "6" that needs to be identified, then the identification number "6" can be divided into three steps: transformation into a 0–1 matrix, similarity calculation, and nearest neighbor search.

2.1 A 0–1 Matrix Transformation

Here, we are to convert the test data from images to a matrix of 0–1 with only one column. In Fig. 1, the position with pixels on the left is denoted as 1, and the position without pixels is denoted as 0. The right part of Fig. 1 shows the result after cutting and stretching. All the last numbers in the right position of Fig. 1 are connected to the end of the previous line to form a row. The column of 0–1 matrix is obtained as a transpose. All the training data are converted from image to 0–1 matrix with only one column. An L single column data is stored in the new matrix A – where each column of matrix A stores all the information of an image.

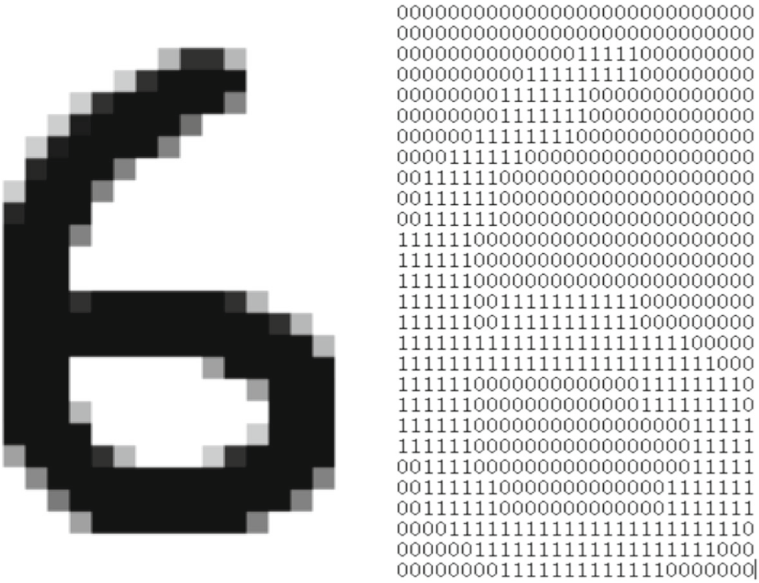

Fig. 1 0–1 matrix of the number "6"

2.2 Similarity Measure

Similarity is represented as the difference in distance between the distances among the training datasets and the testing datasets. The commonly used distance measurement methods include Euclidean distance, Manhattan distance, hamming distance, cosine distance, etc. [13]. There are two methods that are most commonly used as follows: commonly used to calculate distance, which include Euclidean distance and Cosine distance.

2.2.1 Euclidean Distance

The Euclidean distance between two n-dimensional vectors a($x_{11}, x_{12}, \ldots, x_{1n}$) and a($x_{21}, x_{22}, \ldots, x_{2n}$) is expressed as follows:

$$d_{12} = \sqrt{\sum_{k=1}^{n} (x_{1k} - x_{2k})^2}$$

2.2.2 Cosine Distance

In geometry, cosines can be used to measure the difference between two vectors. Machine learning uses this concept to measure the difference between sample vectors.

The number of samples can be expressed as vectors as follows:

$$v_x = < v_{x1}, v_{x2}, \ldots v_{xn} >$$

Where v_{xr} represents the r-th attribute value of sample data v_x. The similarity is therefore as follows:

$$d(v_i, v_j) = cos(\theta) = \frac{v_i.v_j}{|v_i|\,|v_j|} = \frac{\sum_{r=1}^{n} v_{ir}v_{jr}}{\sqrt{\sum_{r=1}^{n} v_{ir}^2}\sqrt{\sum_{r=1}^{n} v_{jr}^2}}$$

The L distances obtained from the test data and each column in matrix A are stored in the distance array (i.e. the distance solution method here refers to the above two methods).

2.3 K-Nearest Neighbor Search

Extracting the index of training set corresponding to the minimum K distances, from distance array. The traditional way is to use sorting algorithm to sort the array sequentially by taking the first K data. The value with the most indexes becomes the predicted value. In this way, the handwritten numerals to be tested can be classified or obtained.

2.4 The Shortcomings of Classical Algorithms

In the case of low recognition accuracy, the time complexity of the whole algorithm is within the acceptable range. This is due to the strong computing power of modern computer. However, when the identified database is very large, a large number of different types of numbers need to be accurately identified. Because the distance between the data to be identified and all the training dataset need to be calculated as well as sorting them to obtain the first K values. This will sharply increase the computational complexity of the traditional algorithm, and that will make the efficiency of the entire algorithm be reduced. Therefore, we consider using the properties of quantum parallel computation to speed up the traditional algorithm at a polynomial level.

3 The Quantum Handwritten Digit Recognition Algorithm

Quantum machine learning has optimized traditional machine learning by virtue of the high parallelism of quantum computing [14]. Quantum applied to machine learning examples is to replace the specific sub-process of the original classical algorithm with the quantum algorithm under the framework of the traditional machine learning algorithm [15]. This is to reduce the computational complexity and improve the efficiency of the algorithm.

Quantum machine learning algorithm generally comprises of the following steps:

(1) Convert classic information into quantum information. In order to implement quantum algorithms, the form of the data we input must be in the form of a quantum state. Cleverly encoding classical information into quantum information enhance the efficiency of quantum computing.

(2) Quantum algorithms accelerate the specific process of traditional machine learning. This step is very important, though it is a complicated. It is necessary to combine not only the data structure of the classical algorithm, the database and other technologies. But it is also to design an algorithm or a model that is more in line with quantum theory.

(3) Extracting the final calculated result. In the end, we need to get the classic information. Therefore, we use quantum measurement and other operations to collapse the quantum superposition state wave packet into classical state, and extract the classical information [16] (Fig. 2).

Following the above research ideas and theories of quantum machine learning, we quantify the handwritten digit recognition algorithm.

3.1 Handwritten Digital Feature Vectors

Considering the handwritten digit recognition quantum algorithm, we first need to find a suitable method to store classical dataset information in quantum state. It is obvious that all the data in a digital computer is finite word length. We assume that a training data is a vector $\overrightarrow{x} = [x_1 \ldots x_n]$. If each element is represented by binary data of length m, then the entire vector requires n × m bits. This corresponds to N = n × m qubits in a quantum computer. Hence, each data set as a vector corresponds to a base loss of $|\overrightarrow{x}\rangle$ of the input state in Hilbert space. Here, we can make full use of the superposition characteristics of the quantum system, and store these data in N-bits in a 'superimposed' manner. This is represented by the state $\sum_j c_j|\overrightarrow{x_j}\rangle$, which is very efficient [15].

Therefore, considering the above theoretical analysis, we use the v_j vector to represent the digital information vector in the training data set [17]. That is, the pixel point information of the handwritten digit through the 0–1 matrix. It can be seen from the above quantum state preparation that a quantum state α is used to store information of all instances to be classified. Thus a quantum state β is used to store all sample instance information. The digital information vector in the training data set is represented by the v_j vector, and that is, pixel point information after the handwritten digit 0–1 matrix is represented. We convert real vector $v = (v_0, v_1, \ldots, v_n)$ into quantum state $|v\rangle$. Thereby, we convert the distance comparison of the vector into distance comparison between quantum states.

$$|v\rangle = \frac{v}{|v|}$$

3.2 Preparation of Quantum States

Here, we introduce the handwritten digital feature vector design given above into the quantum state to create initial conditions for implementing the quantum [18]. Moreover, the basis of quantum computing is quantum bits. Therefore, firstly prepare two quantum states: a quantum state α is used to store information on all instances to be classified; a quantum state β is used

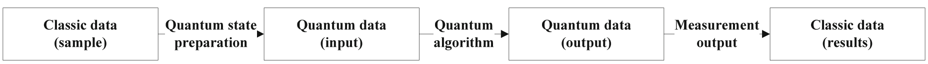

Fig. 2 Basic flow of quantum machine learning

to store all handwritten digital sample instance information. Thereafter, both α and β are superimposed as given in the expressions below:

$$a = \frac{1}{\sqrt{d}} \sum_{i=1}^{d} |\, i\,\rangle \left(\sqrt{1 - v_{0i}^2}\, |0\rangle + v_{0i}\, |1\rangle \right) |1\rangle$$

$$\beta = \frac{1}{\sqrt{M}} \sum_{j=1}^{M} |\, j\,\rangle \frac{1}{\sqrt{d}} \sum_{i=1}^{d} |\, i\,\rangle\, |1\rangle \left(\sqrt{1 - v_{ji}^2}\, |0\rangle + v_{ji}\, |1\rangle \right)$$

Where d is the dimension of vectorv_j, and v_{ji} represents the i-th attribute value of vector v_j. It is assumed that, the vector of the instance to be classified is represented asv_0. The handwritten digital sample vector (i.e. handwritten digital training dataset vector) of the known category is represented as$v_j (j = 1, 2, \ldots, M)$. There are three operations that the Oracle operator require to prepare the α or β [3]. Taken the preparation of quantum state β as an example, the basic processes are given below:

(1) Prepare an initial state as $\frac{1}{\sqrt{M}} \sum_{j=1}^{M} |\, j\,\rangle \frac{1}{\sqrt{d}} \sum_{i=1}^{d} |\, i\,\rangle\, |1\rangle |0\rangle$. The Oracle finds the quantum state as $\frac{1}{\sqrt{M}} \sum_{j=1}^{M} |\, j\,\rangle \frac{1}{\sqrt{d}} \sum_{i=1}^{d} |\, i\,\rangle\, |v_{ji}\rangle |0\rangle$. Therefore, the Oracle operator is defined as $O \mid j \rangle \mid i \rangle \mid 0 \rangle = \mid j \rangle \mid i \rangle \mid v_{ji} \rangle$.

(2) Apply $R_y(2sin^{-1} v_{ji})$ to the last qubit to obtain the quantum state as $\frac{1}{\sqrt{M}} \sum_{j=1}^{M} |\, j\,\rangle \frac{1}{\sqrt{d}} \sum_{i=1}^{d} |v_{ji}\rangle\, |1\rangle \left(\sqrt{1 - v_{ji}^2}\, |0\rangle + v_{ji}\, |1\rangle \right)$. Where $R_y(2sin^{-1} v_{ji})$ represents a trick operation rotating around the Y axis. The specific trick is the following matrix:

$$R_y\left(2sin^{-1} v_{ji}\right) = \begin{bmatrix} \sqrt{1 - v_{ji}^2} & -v_{ji} \\ v_{ji} & \sqrt{1 - v_{ji}^2} \end{bmatrix}$$

(3) Take an inverse of the Oracle operator to clear v_{ji} of the auxiliary bit to obtain the target quantum state as β. Similarly, it takes 3 Oracle operations to prepare an initial quantum state α for the handwritten digits to be classified.

3.3 Distance Measurement

In the process of preparing the quantum states α and β, the vector information of the handwritten digital data is stored in the amplitude of the quantum state. Therefore, an angle between the vectors stored in amplitude of the quantum state is obtained through switching operation of the unitary matrix. The measurement of the cosine distance, and the quantum circuit of the handwritten digital trick exchange operation is shown in Fig. 3.

The trick operation performs the swapping when the first handwritten digit has a qubit of 1 or otherwise, when the qubit is 0 the implements $|\alpha\rangle \mid \beta\rangle \rightarrow \mid \beta\rangle \mid \alpha\rangle$. Therefore quantum states are obtained by the trick as follows:

$$\frac{1}{2} |0\rangle (|\alpha\rangle |\beta\rangle + |\beta\rangle |\alpha\rangle) + \frac{1}{2} |1\rangle (|\alpha\rangle |\beta\rangle - |\beta| \alpha\rangle)$$

The first qubit becomes $\frac{1}{2}(|0\rangle + |1\rangle)$ after the first Handamard gate. Thereafter it acts as a control bit to control the operation of the Swap gate. The quantum states α and β are then changed to $\frac{1}{2}(|0\rangle |\alpha\rangle |\beta\rangle + |1\rangle |\beta\rangle |\alpha\rangle)$ after the operation of the Swap gate. Finally, after the

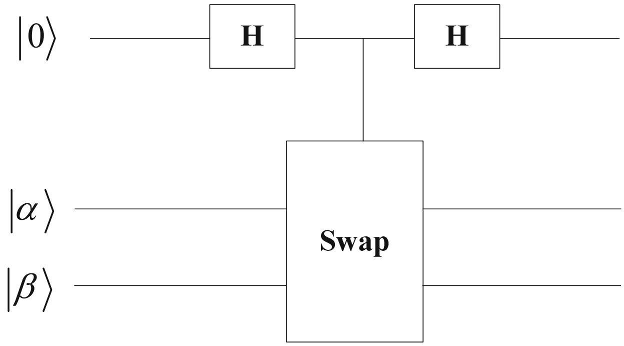

Fig. 3 Controlled exchange tips for measuring cosine distance of angle

first qubit of each superposition state is subjected to the Hadamard gate, the total quantum state φ becomes$\frac{1}{2}|0\rangle(|\alpha\rangle|\beta\rangle+|\beta\rangle|\alpha\rangle)+\frac{1}{2}|1\rangle(|\alpha\rangle|\beta\rangle-|\beta\rangle|\alpha\rangle)$.

The handwritten digital quantum state γ can be obtained by controlling the switching gate as given below:

$$\frac{1}{\sqrt{M}}\sum_{j=1}^{M}|j\rangle\left[\sqrt{1-d\left(v_0,v_j\right)}|0\rangle+\sqrt{d\left(v_0,v_j\right)}|1\rangle\right]$$

The amplitude estimation method is applied to the quantum state. The similarity of each handwritten digital sample data v_j and v_0 is then converted to the quantum bit to obtain the following quantum state:

$$\sigma=\frac{1}{\sqrt{M}}\sum_{J=1}^{M}|j\rangle\left|\left|v_j-v_0\right|\right\rangle$$

After calculation, the probability that the first qubit of the quantum state φ becomes 1 is given below:

$$\frac{1}{2}-\frac{|\left(\alpha\mid\beta\right\rangle|^2}{2}$$

Therefore:

$$\left(\alpha\mid\beta\right\rangle=\frac{\left(v_0\mid v_j\right\rangle}{d}$$

Where $\langle v_0\mid v_j\rangle$ is the cosine angle of the handwritten digital sample vector v_j (i.e. known handwritten digit instance vector and the known category). Since the dimension d is constant, it can be considered that the probability that the first qubit of the quantum state φ is 1 represents the similarity $d(v_0, v_j)$ between the two instances. Therefore, the $\frac{1}{2}-\frac{|\left(\alpha\mid\beta\right\rangle|^2}{2}$ can been written as $d(v_0, v_j)$. It can be seen that the smaller the value, the larger the corresponding cosine angle. This indicates that the two handwritten digit instances are more similar.

3.4 The most Similar K Values

The below outline indicate the steps to obtain the k points that are most similar to the handwritten digital data (i.e. those to be classified from the quantum state σ.)

(1) Let $K = \{K_1, K_2, \ldots, K_K\}$ denote the k points that are most similar to v_0. Initially, randomly select k points from the handwritten digital sample data set into K.
(2) Use Grover algorithm [19] at each time a point v_j is obtained from the quantum state σ. (i.e. which is closer to the point to be classified than a point existing in K).
(3) Replace the point K_x in K with the j-th point. Where x is defined by:

$$\max\{d(v_0, v_{K_x})\}, x \in [1, k]$$

(4) Obtain the smallest k points by repeating steps 2 and 3. Such that the value of t keeps decreasing until t = 0 to find the nearest k points.

```
Quantum-based K-value search algorithm
1: procedure K_SORT
2:      K ← 0, t ←?
3:      select k from Sample Set
4:      K ← {k_1, k_2, …, k_k}
5:      repeat
6:          v_j from Quantum state σ              → Grover Algorithm
7:          if d(v_0, v_j) < d(v_0, v_{K_x}) and x ∈ [1, k] then
8:              k_x ← v_j
9:              → x in max{d(v_0, v_{K_x})} and x ∈ [1, k]
10:     until t == 0
11: end procedure
```

The value with highest index among the k values becomes the predicted value (when there are multiple modes, by distance and minimum). Finally, the category of the test handwritten digits is represented by the classical information. And thus, we arrive at the handwritten digit recognition of the quantum mode.

4 Analysis of Proposed Algorithm

The proposed improved quantum K-nearest neighbor algorithm that is presented in this paper can aid to solve the problems. It is worth noting that the handwritten digital sample space is M d-dimensional vector. Each vector corresponds to a known handwritten digit category. The proposed quantum K-nearest neighbor algorithm can aid to determine the category of a handwritten digit instance vector to be classified. The proposed algorithm steps include:

(1) Constructs a matrix vector of handwritten digits, and prepares a quantum state α (i.e. for storing vector information in the handwritten digital sample space) and a quantum state β (i.e. for storing the classification vector information).

(2) Selects the appropriate quantum line (or trick operator) according to α and β to calculate the similarity distance between the handwritten digital sample space vector and the vector to be classified. The similarity information is stored into the amplitude of the quantum state.
(3) Transfers the similarity information on the quantum state amplitude to the quantum state (i.e. using the amplitude estimation algorithm).
(4) Finds the most similar K points in the search space, using a quantum algorithm that searches for the smallest K values.

After analysis, it is noticed in the proposed algorithm that: The corresponding quantum state in step 1 requires a total of 6 Oracle operations; The number of Oracle operations required for amplitude estimation in step 3 becomes a constant R [20]; Searching for k values in step four requires $O\left(\sqrt{kM}\right)$ times of Oracle iterative operations, where M represents the training set size of the handwritten digits. The time complexity of the entire algorithm becomes $12R\sqrt{kM}$. Comparatively the main computational time complexity of traditional handwritten digit recognition algorithms relies on the calculation of similarity distance and, also searching for the k values. Where, the time complexity is $O(M)$ and $O(kM)$, respectively. Moreover, the total time complexity is $O(kM^2)$. Therefore, it can be seen that after quantum acceleration, the efficiency of the existing handwritten digit recognition algorithm has been greatly improved in the proposed algorithm.

5 Conclusion

This paper implements the idea of quantum parallel computing and superposition applied to the KNN traditional machine learning algorithm such as KNN. Our approach adopts the KNN algorithm with quantum acceleration to improve handwritten digit recognition. Our proposed approach has significantly reduce the computational time complexity of the classical KNN algorithm when dealing with increasingly complex and large handwritten digital data sets. The paper provides a theoretical study of the application of quantum ideas to machine learning. We finally, establishes a basic operational model, and process system for quantum accelerated machine learning. Even though, KNN algorithm is an algorithm to handle handwritten digit recognition. The deep learning neural network algorithm has better performance in solving the problems outlined in this paper [21]. We will apply quantum computing to study classical neural network algorithms in our future work. We hope to derive approaches that will achieve a quantum acceleration to improve upon that of traditional neural network algorithms [16].

Acknowledgments This work is supported by National Natural Science Foundation of China (61802033, 61751110), supported by Postdoctoral research Foundation of China (216638), and also supported by the opening project of Guangdong Provincial key Laboratory of Information Security Technology (2017B030314131).

References

1. chen, W., Gao, Y., Zhang, J.: Quantum K- neighbor algorithm [J]. Journal of Southeast University (Science Edition). **45**(4), 647–651 (2015)
2. Yoo, S., Bang, J., Lee, C., Lee, J.: A quantum speedup in machine learning: finding an N-bit Boolean function for a classification. New J. Phys. **16**(10), 103014 (2014)

3. Huang, Y.M., Lei, H., Li, X.: Overview of quantum machine learning algorithms [J]. Chinese Journal of Computers. **41**(1), 145–163 (2018)
4. Lu, S.C., Zhen, Y., Li, X., Wu, R.: Quantum Machine Learning [J]. Control Theory and Applications. **34**(11), 1429–1436 (2017)
5. Godfrin, C., Ferhat, A., Ballou, R., Klyatskaya, S., Ruben, M., Wernsdorfer, W., Balestro, F.: Operating Quantum States in Single Magnetic Molecules: Implementation of Grover's Quantum Algorithm. Phys. Rev. Lett. **119**, (2017)
6. Bengio, Y.: Learning deep architectures for AI. Mach. Learn. **2**(1), 1–127 (2009)
7. ANGUITA, D., RIDELLA, S., RIVIECCIOI, F., et al.: Quantum optimization for training support vector machines [J]. Neural Netw. **16**(5/6), 763–770 (2003)
8. Wiebe N, Kapoor A, Svore K. Quantum Algorithms for Nearest-Neighbor Methods for Supervised and Unsupervised Learning[J/OL]. Eprint arXiv, (2014). http://arxiv.org/pdf/1401.2142v2.pdf
9. SENTIS, G., CALSAMIGLIA, J., MUNOZTAPIA, R., et al.: Quantum learning without quantum memory[J]. Sci. Rep. **2**, 708 (2012)
10. Beckmann B, Kriegel H P, Schneider R, et al. The R* -tree: an efficient and robust access method[C]//9th ACM SIGMOD International Conference on Management of Data. Atlantic City 322–331 (1990)
11. Lloyd, S., Mohseni, M., Rebentrost, P.: Quantum principal component analysis [J]. Nat. Phys. **10**(9), 631–633 (2014)
12. JORDAN, S.P.: Fast quantum algorithm for numerical gradient estimation [J]. Phys. Rev. Lett. **95**(5), 050501 (2005)
13. Ren, D., Chen, X.: The principle and application of handwritten numeral recognition [J]. Computer Age. **3**, 17–21 (2007)
14. GOODFELLOW, I., BENGIO, Y., COURVILLE, A.: Deep Learning [M]. IT Press, Massachusetts (2016)
15. Nielsen, M.A., Chuang, I.L.: Quantum Computation and Quantum Information. Cambridge University Press, Cambridge (2000)
16. Da Silva, A.J., De Oliveira, W.R., Ludermir, T.B.: Classical and superposed learning for quantum weightless neural networks. Neurocomputing. **75**(1), 52–60 (2012)
17. PAN, J., CAO, Y., YAO, X., et al.: Experimental realization of quantum algorithm for solving linear systems of equations[J]. Phys. Rev. A. **89**(2), 1150–1154 (2013)
18. RUIZ-PEREZ, L., GARCIA-ESCARTIN, J.C.: Quantum arithmetic with the quantum fourier transform [J]. Quantum Inf. Process. **16**(6), 152 (2017)
19. Cortes, C., Vapnik, V.: Support-vector networks. Mach. Learn. **20**(3), 273–297 (1995)
20. Neven H, Denchev V S, Rose G, et al. QBoost: large scale classifier training with adiabatic quantum optimization//Proceedings of the 4th Asian Conference on Machine Learning. Singapore, 333–348 (2012)
21. Pudenz, K.L., Lidar, D.A.: Quantum adiabatic machine learning. Quantum Inf. Process. **12**(5), 2027–2070 (2013)