2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC) | 978-1-7281-8028-1/20/$31.00 ©2021 IEEE | DOI: 10.1109/IAEAC50856.2021.9390901

Principal Component Analysis Based on Quantum Genetic Algorithm with T-Distribution Parameters

Yujie Shen[1], Wei Nai[2,*], Yue Li[1], Jiadi Shen[1], Zan Yang[3], Dan Li[3], Yidan Xing[2]
1. Department of Economy and Management, Tongji Zhejiang College, Jiaxing, China
2. Department of Electronic and Information Engineering, Tongji Zhejiang College, Jiaxing, China
3. Faculty of Science, Tongji Zhejiang College, Jiaxing, China
*Corresponding author, e-mail: alexni@tongji.edu.cn

Abstract—**Machine learning is a important part of the field of artificial intelligence. Principal component analysis (PCA) is a classical linear dimensionality reduction algorithm in the field of machine learning, which is usually solved by gradient ascending method. As the traditional gradient ascending method requires the convexity of the objective function, it is easy to be trapped into the local maximum value. Therefore, in order to solve the limitation of being trapped into local optimal solution, it is meaningful to study the performance optimization for principal component analysis. In this paper, a novel hybrid algorithm which employs quantum genetic algorithm (QGA) with t-distribution parameters in traditional PCA has been proposed. Via simulation analysis, it has been verified that the proposed hybrid algorithm has better performance than the single PCA algorithm.**

Keywords-PCA; quantum genetic algorithm; dimension reduction; gradient ascending method; optimization

I. INTRODUCTION

In recent years, the research on artificial intelligence (AI) has gradually become the trend of the times. Generally, AI algorithms includes machine learning and deep learning. Principal component analysis (PCA) is an unsupervised dimensionality reduction algorithm in machine learning which can keep the integrity of original information to the greatest extent, and it has been widely used. However, as the optimal solution finding process in PCA is based on gradient ascending method, and such method always requires the convexity of the objective function, the optimal solution finding process is always trapped into the local maximum value. Therefore, the optimization of principal component analysis is of great significance.

As it comes to PCA, predecessors have done a lot of research. T. Sugiyama (1965) from Tokyo College of Science has obtained the exact distribution of the latent vectors when the observations are obtained from bivariate normal distribution [1]. W. F. Velicer (1974) has put forward that one of the positive features sometimes attributed to factor analysis, as opposed to component analysis, has superior stability of results under sampling from a population of variables [2]. At the same time, predecessors have also done a lot of research on quantum genetic algorithm (QGA). G. Zhang et al (2003) proposed a novel parallel evolutionary algorithm called coarse-grained parallel QGA (CGPQGA). The main points of CGPQGA are a new chromosome representation called qubit representation, a novel evolutionary strategy called qubit phase comparison approach and an extended version of coarse-grained model called hierarchical ring model [3]. Y. K. Zhang et al (2011) have also proposed an improved evolutionary algorithm, which encodes the chromosome with quantum bit as its basic information bit. It achieves individual evolution by improving the quantum rotating gates using adaptive rotation, and its convergence speed and global optimization ability is superior to the traditional evolutionary algorithm [4]. In order to improve the adaptability of the QGA for the multi-objective model, W. He et al (2019) have proposed the multi-constraint and multi-target construction optimization model with minimum period and cost, which is established based on the QGA, and the encoding form, quantum revolving door, and genetic flow of the algorithm are reconstructed [5].

Based on the research above, it can be found that there have been a large number of researches on principal component analysis and quantum genetic algorithm respectively. However, few scholars have studied the hybrid algorithm combining principal component analysis with quantum genetic algorithm. In this paper, quantum genetic algorithm is applied to principal component analysis, and a large number of experiments show that this method can effectively reduce the probability of being trapped into the local maximum value.

II. TRADITIONAL PRINCIPAL COMPONENT ANALYSIS

In machine learning, data is often expressed as vectors, and the complexity of machine learning is closely related to the dimension of data. Therefore, in the case of dealing with thousands of dimensions, the dimension of data have

to be reduced before the data is analyzed. Dimensionality reduction means there is information loss, and as the actual data has correlations, the information loss should be minimized in the process of dimensionality reduction. Traditional dimensionality reduction algorithms can be classified according to supervised and unsupervised, linear and nonlinear criteria.

PCA in dimension reduction algorithm is introduced to conduct data-based discussion on the problems requiring dimension reduction. As PCA is an unsupervised linear dimensionality reduction algorithm in the field of machine learning, its purpose is to find out the "mainest" elements from multiple objects or data that influence the purpose of research and simplify the dimensionality reduction for analysis. Due to its lack of parameters, such algorithms are widely used in various fields, mainly for data dimensionality reduction, but also for visualization, noise and redundancy removal, pattern recognition, image compression and data preprocessing. PCA can extract the most valuable information by mapping data in multiple high-dimensional spaces to low-dimensional spaces while maintaining many original data features and maximizing the variance of projection points. Taking *n*-dimensional space as an example, the following shows the process of principal component analysis algorithm and analyzes the disadvantages of solving the maximum variance optimal parameter with the traditional gradient ascending method.

Suppose there is an initial population Q (*t*), *t* represents the population of the last generation, namely *t* samples, and the new sampling points obtained after the projection of *t* samples in the low-dimensional space are expressed as, construct a matrix with *t* data as rows and *n*-dimensional eigenvalues as columns, and use variance to represent the density of sampling points.

The representation of the variance:

$$Var(q)=\frac{1}{t}\sum_{i=1}^{t}(q_i-\bar{q})^2 \tag{1}$$

For the centralization of all eigenvalues, the mean value of data should be set as 0, so as to rotate the graph and achieve the minimum reconstruction loss angle.

$$Var(q)=\frac{1}{t}\sum_{i=1}^{t}q_i^2 \tag{2}$$

Assuming that the direction vector of reconstructed principal component 1 (PCA1) is $w = (w_1, w_2)$, after all original samples are mapped to PCA1, the following formula can be obtained:

$$Var(q_{project})=\frac{1}{t}\sum_{i=1}^{t}(q_{project}^{(i)}-\bar{q}_{project})^2 \tag{3}$$

Equation (3) is obtained after taking the mean value:

$$Var(q_{project})=\frac{1}{t}\sum_{i=1}^{t}||q_{project}^{(i)}||^2 \tag{4}$$

From the definition of vector projection and vector dot product, it can be seen that:

$$||q_{project}^{(i)}||=q^{(i)}\cdot w=||\,q^{(i)}\,||\cdot||\,w||\cdot\cos\theta \tag{5}$$

Then, $||q^{(i)}_{project}||$ in equation (4) can be replaced by its value acquired in equation (5):

$$Var(q_{project})=\frac{1}{t}\sum_{i=1}^{t}(q^{(i)}\cdot w)^2 \tag{6}$$

The expansion form of equation (6) is:

$$Var(q_{project})=\frac{1}{t}\sum_{i=1}^{t}(q_1^{(i)}w_1+q_2^{(i)}w_2+\ldots+q_n^{(i)}w_n)^2 \tag{7}$$

Equation (7) can be simplified as:

$$Var(q_{project})=\frac{1}{t}\sum_{i=1}^{t}\left(\sum_{j=1}^{n}q_j^{(i)}w_j\right)^2 \tag{8}$$

In other words, the goal now is to find a *w* that maximizes this expression, and we often use the gradient rise method to find the optimal value, and we can gradually get the gradient of the original variance value through the following method.

Assuming that

$$\begin{aligned} f(q)&=Var(q_{project})\\ &=\frac{1}{t}\sum_{i=1}^{t}(q_1^{(i)}w_1+q_2^{(i)}w_2+\ldots+q_n^{(i)}w_n)^2 \end{aligned} \tag{9}$$

Here, the goal is to find the gradient of $f(q)$. Since the value of *w* is unknown, the sample number provided by *Q* for unsupervised machine learning can be obtained as:

$$\begin{aligned} \nabla f&=\begin{pmatrix}\frac{\partial f}{\partial w_1}\\ \frac{\partial f}{\partial w_2}\\ \cdots\\ \frac{\partial f}{\partial w_n}\end{pmatrix}=\frac{2}{t}\begin{pmatrix}\sum_{i=1}^{t}(q_1^{(i)}w_1+q_2^{(i)}w_2+\ldots+q_n^{(i)}w_n)q_1^{(i)}\\ \sum_{i=1}^{t}(q_1^{(i)}w_1+q_2^{(i)}w_2+\ldots+q_n^{(i)}w_n)q_2^{(i)}\\ \cdots\\ \sum_{i=1}^{t}(q_1^{(i)}w_1+q_2^{(i)}w_2+\ldots+q_n^{(i)}w_n)q_n^{(i)}\end{pmatrix}\\ &=\frac{2}{t}\begin{pmatrix}\sum_{i=1}^{t}(q^{(i)}w)q_1^{(i)}\\ \sum_{i=1}^{t}(q^{(i)}w)q_2^{(i)}\\ \cdots\\ \sum_{i=1}^{t}(q^{(i)}w)q_n^{(i)}\end{pmatrix} \end{aligned} \tag{10}$$

In equation (10), there is a step of vecorization, by this mean, it turns the gradient solution problem into a matrix class operation:

$$\nabla f = \frac{2}{t}(q^{(1)}w, q^{(2)}w, ..., q^{(i)}w) \cdot \begin{pmatrix} q_1^{(1)} & q_2^{(1)} & ... & q_n^{(1)} \\ q_1^{(2)} & q_2^{(2)} & ... & q_n^{(2)} \\ ... & ... & & ... \\ q_1^{(t)} & q_2^{(t)} & ... & q_n^{(t)} \end{pmatrix} \quad (11)$$
$$= \frac{2}{t}(qw)^t \cdot q$$

After obtaining the gradient of $f(q)$, the gradient ascending method can be applied to find the optimal parameter.

III. Optimization Algorithm

A. Traditional Genetic Algorithm

Inspired by the theory of biological evolution and the theory of heredity, genetic algorithm (GA), which includes three operations of selection, crossover and mutation, has the basic characteristics of intelligent search, progressive optimization, global optimal solution, black box structure and strong versatility, and is widely used in structural optimization design, complex system analysis and other fields. *T*-distribution has long tail effect, which can avoid premature algorithm and jump out of the trap of local extremum. Therefore, in this paper, the uniform distribution in is replaced by the t-distribution to increase the randomness of the algorithm.

The operation process of genetic algorithm is as follows:

First, initialize the population, $Q(t) = \{q_n^1, q_n^2, ..., q_n^t\}$, each q_n^t is a chromosome, and t represents the generation.

Second, the fitness function $fit(f(x))$ is constructed by observing the state of $Q(t)$, where $f(x)$ is the objective function. Fitness function combines the objective function with fitness, and fitness represents fitness value. The higher the fitness, the more likely it is to be inherited to the next generation. In this evolutionary process, the objective function gradually tends to be optimal.

Third, evaluate $fit(f(x))$. If the optimization criteria are not met, it will go to the next step. If satisfied, it terminates.

Fourth, select operation. According to the fitness function, the optimized individuals are selected by means of roulette, that is, the individuals with high fitness are retained and reproduced with a large number, while the individuals with low fitness are reproduced with a small number or even eliminated.

Fifth, cross operation. Cross operation is to exchange the partial genetic codes of two random individuals, including single-point crossover, two-point crossover, multi-point crossover, and partial matching crossover, etc. , thus generating new individuals.

Sixth, mutation operation. The mutation operation imitates the mutation phenomenon in the biological evolution, with the random probability, causes the partial genetic code to change, also produces the new individual.

Finally, in the evolution process, after continuous selection, crossover and mutation operation, until there is no more evolutionary trend, so as to get the population with the highest fitness, that is, gradually approaching the optimal solution, running to the specified algebra, that is, the termination algebra.

However, the traditional genetic algorithm has poor parallelism, slow convergence speed and low optimization efficiency, and its coding technology and program expression technology need to be improved, as well as the three genetic operations need to be improved, so quantum computation is introduced.

B. Quantum Computing

Quantum bits are the basis of quantum computing, and they are different from classical bits, except for |0> and |1>, which can be any combination of |0> and |1> linear superpositions, that is $|\varphi\rangle = \alpha |0\rangle + \beta |1\rangle$, where α and β are a pair of complex Numbers, called quantum probability amplitude, and satisfies $|\alpha|^2 + |\beta|^2 > 1$.

In quantum computing, quantum gate is a device to realize logical changes, and quantum revolving gate is one of them. Quantum revolving door has a rotation Angle θ, each iteration, the rotation Angle θ is close to the current optimal value, the individual also tends to a better state each time.

C. Case Study

By combining traditional genetic algorithm with quantum computation and based on quantum bit and quantum superposition state, the genetic code is represented by quantum state vector, so that the superposition of multiple states can be expressed by one chromosome, and the gene at this time contains all possible information. On this basis, reference quantum gate to achieve the update operation, can be better toward the optimal direction, so as to get the optimal solution. Quantum genetic algorithm combines the advantages of quantum computing parallel algorithm and genetic algorithm such as population diversity, can study a variety of nonlinear, multivariable, multi-objective, complex adaptive system problems. It is used in artificial life design research, artificial intelligence and other fields. Because PCA algorithm has certain shortcomings, it may not get the global optimal solution, but the local optimal solution. Therefore, quantum genetic algorithm is introduced to solve these problems.

Quantum genetic algorithm uses exchange and mutation operations to generate new individuals, so the optimization results obtained by the search approach the global optimal solution, and the quantum genetic algorithm is a parallel operation, the search speed is fast. Therefore, we can introduce quantum genetic algorithm to solve the global optimal solution problem in solving the optimal parameter problem in (11).

First, initialize the population $Q(t) = \{q_n^1, q_n^2, ..., q_n^t\}$, each q_n^t is a chromosome, and t represents the generation.

Second, the fitness function $fit(f(x))$ is constructed by observing the state of $Q(t)$.

Third, evaluate $fit(f(x))$. If the optimization criteria are not met, it will go to the next step. If satisfied, it terminates.

Fourth, update $fit(f(x))$ with quantum rotation gate $U(t)$, and mutate some individuals in the population according to certain probability. The encoding method of quantum genetic algorithm is to use quantum bits and quantum superposition state to encode chromosomes, so that the state of a chromosome can be in the superposition state of various possibilities, which greatly enriches the diversity of the population. At the same time, the quantum rotation gate is used to update the probability amplitude of qubits, so as to achieve the effect of gene variation and make the population update. Each iteration is close to a better state, which not only improves the convergence speed of the algorithm, but also prevents it from falling into the local optimal state. In the iteration process, each superposition state will be a certain state at last, which tends to be stable, reaches convergence, and finally achieves the goal of optimization. Quantum revolving door is:

$$U(\theta_i) = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) \\ \sin(\theta_i) & \cos(\theta_i) \end{bmatrix} \tag{12}$$

The adjustment operation of quantum revolving door $U(t)$ is as follows:

$$\begin{bmatrix} \alpha'_i \\ \beta'_i \end{bmatrix} = U(\theta_i) \begin{bmatrix} \alpha_i \\ \beta_i \end{bmatrix} = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) \\ \sin(\theta_i) & \cos(\theta_i) \end{bmatrix} \begin{bmatrix} \alpha_i \\ \beta_i \end{bmatrix} \tag{13}$$

where, (a_i, β_i) is the ith quantum bit in the chromosome, θ_i is the rotation Angle, $\theta_i = S(a_i, \beta_i)\Delta\theta_i$, $S(a_i, \beta_i)$and $\Delta\theta_i$ represents the direction and Angle of rotation respectively.

Finally, termination conditions. The loop terminates when the set $fit(f(x))$ value reaches the optimal value; Otherwise, it returns to step 4 to restart the loop. The result obtained is the global optimal solution.

IV. Numerical Experiment

The data in Reference [6] has been used for numerical experiments. For convenience of expression, the number of iterations has been recorded as NIT. Quantum genetic algorithm with t-distribution parameters in this paper is called QGAT. Gradient descent method is recorded as GD. Stochastic gradient descent method is recorded as SGD. $\rho_s(\tau)$ is the standard indicator of the possibility of solution [11]. Figure 1 and Figure 2 show that QGAT costs less CPU time than GD algorithm and SGD algorithm. Figure 3 and Figure 4 show that QGAT has fewer iterations than GD algorithm and SGD algorithm.

The data of this paper comes from the wine data in UCI. Here in the numerical experiment, three features of wine data are reduced to one-dimensional space by using Principal component analysis based on quantum genetic algorithm with t-distribution parameters, as shown in Figure 5, Figure 6 and Figure 7.

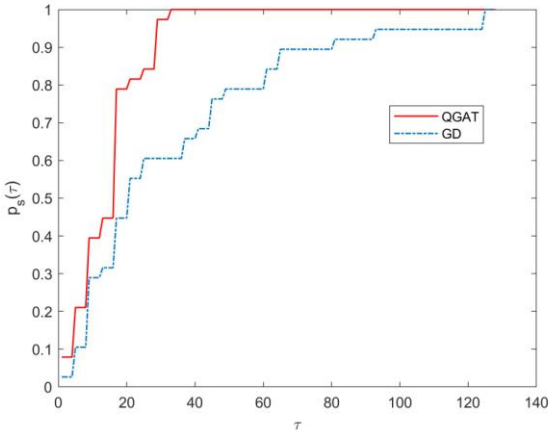

Figure 1. Performance of CPU Time for QGAT and GD.

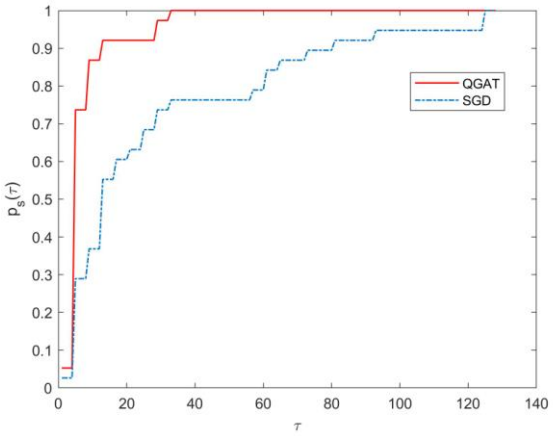

Figure 2. Performance of CPU Time for QGAT and SGD.

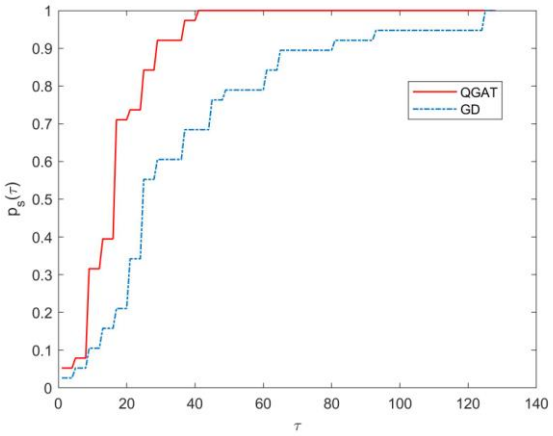

Figure 3. Performance of NIT for QGAT and GD.

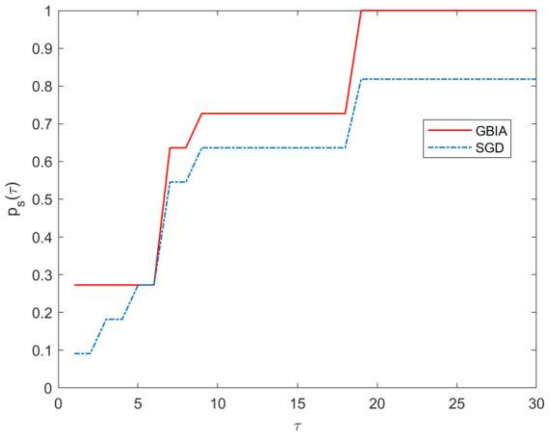

Figure 4. Performance of NIT for QGAT and SGD.

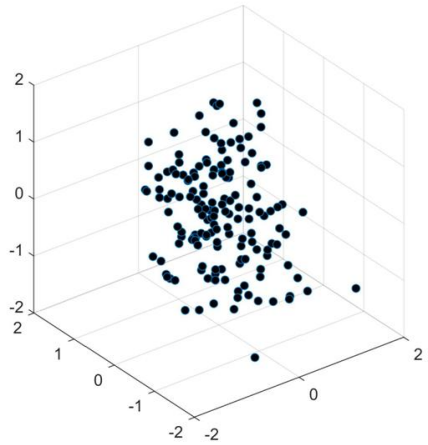

Figure 5. Three dimensional image of three features.

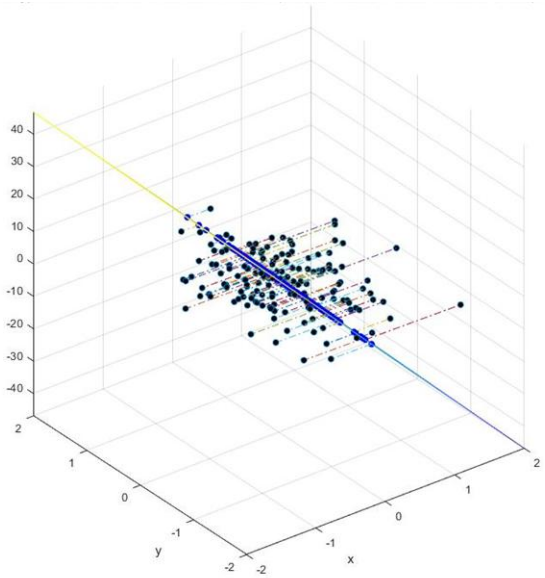

Figure 6. Project 3D data to 2D.

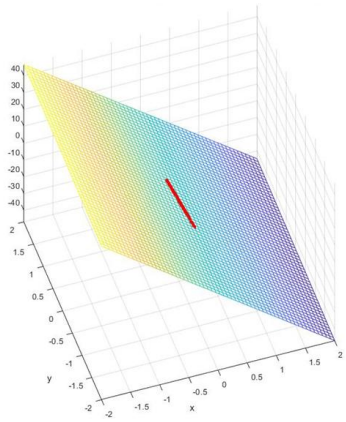

Figure 7. Project 2D data to 1D.

V. CONCLUSIONS

It can be seen from the numerical experiments that the time efficiency and iteration speed of QGAT method are better than that of gradient descent method and stochastic gradient descent method, and that QGAT method has better global convergence than the original gradient dependent line search method. QGAT overcomes the shortcomings of the former two methods, and has a good performance for nonlinear optimization.

ACKNOWLEDGMENT

This work is partly supported by Zhejiang Soft Science Research Project under Grant No. 2019C35092, Science and Technology Commission of Shanghai Foundation under Grant No. 19DZ1205804, and Jiaxing Public Welfare Research Project under Grant No. 2020AY10033 and No. 2020AY30025.

REFERENCES

[1] T. Sugiyama, "On the Distribution of the Latent Vectors for Principal Component Analysis," The Annals of Mathematical Statistics, vol. 36, no. 6, 1965, pp. 1875-1876.

[2] W.F. Velicer, "A Comparison of the Stability of Factor Analysis, Principal Component Analysis, and Rescaled Image Analysis," Educational and Psychological Measurement, vol. 34, no. 3, 1974, pp. 563-572.

[3] G.X. Zhang, W.D. Jin, and L.Z. Hu, "A Novel Parallel Quantum Genetic Algorithm," 4th International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT), 2003.

[4] Y. K. Zhang, J. C. Liu, Y. A. Cui, X. H. Hei, and M. H. Zhang, "An Improved Quantum Genetic Algorithm for Test Suite Reduction," IEEE International Conference on Computer Science and Automation Engineering, 2011.

[5] W. He, Y. C. Shi, and T. Dede, "Multiobjective Construction Optimization Model Based on Quantum Genetic Algorithm," Advances in Civil Engineering, vol. 2019, no. 3, 2019, paper ID 5153082.

[6] D. Li, and D. Zhu, "An affine scaling interior trust-region method combining with nonmonotone line search filter technique for linear inequality constrained minimization," International Journal of Computer Mathematics, vol. 95, no. 8, 2018, pp. 1494-1526.