Particle Swarm Optimization in Gene Expression Profile Clustering

Yongjian Dong
School of Information Engineering, Changzhou Vocational Institute of Mechatronic Technology
Changzhou 213164, China
E-mail: dyj_197504@163.com

Keywords: gene expression profile, clustering, particle swarm optimization, optimization

Received: June 4, 2024

The traditional gene expression profile clustering method is affected by the number of iterations, its gene sequence marker value is in a negative range for a long time, resulting in poor clustering ability. Therefore, the study firstly obtains gene expression profile data through DNA micro-array experiments, then unifies the order of data expression values, and optimizes the particle clusters by increasing the inertia weights and learning factors to extract the data features of gene expression profiles. Finally, the most commonly used of particle code updates the clustering center to realize gene expression profile clustering. The experimental results showed that compared with traditional PSO, K-means, and hierarchical clustering methods, the improved PSO algorithm performed well in terms of clustering accuracy and contour score of 95.77% and 0.91, respectively, and the shortest computation time of up to 100.07 s. The results are expected to provide a new technical support for gene expression profile clustering.

Povzetek: Predstavljena je izboljšana metoda gručenja genskih izraznih profilov s pomočjo optimizacijskega algoritma roja delcev (PSO). Za natančnejše združevanje genske ekspresije se izboljšana PSO metoda uporablja za obvladovanje visokodimenzionalnih podatkov, zmanjšanje redundantnosti in izbiro ključnih značilnosti v onkoloških raziskavah.

1 Introduction

With the rapid development of human genome project, micro-array technology has been widely used in various fields of life science, resulting in the exponential growth of gene expression data. Gene expression profile data can help understand gene expression patterns at the molecular level and study life phenomena at the microscopic level. It has high application value for understanding the pathogenesis and diagnosis of cancer at the genetic level [1].

Gene expression profile data, also known as micro-array data, is a high-throughput gene expression data value measured by gene chip technology at different physiological stages [2]. Among them, "gene expression" refers to the transformation of genetic information stored in DNA into protein molecules with biological Mars through transcription and translation. "Gene expression level" refers to the amount of protein produced by a gene in a certain period of time, which indicates the current physiological state of the cell [3]. By analyzing the high-throughput gene expression data with reasonable methods, the regulatory relationship between different genes is obtained. The gene expression level varies between different samples and affects gene activity at different physiological stages. Therefore, gene expression profile data can be used to diagnose tumor and other diseases, analyze the occurrence mechanism of tumor diseases from the perspective of gene molecules, and analyze the changes in patients before and after medication [4].

In the research of gene expression profile data, researchers found that only a few genes played a key role in cancer recognition in high-dimensional data. A large number of redundant genes not only cause serious “Dimension disasters", but also interfere with the correct identification of cancer, leading to a decrease in classification performance. Therefore, it is very important to take appropriate methods to reduce the dimension of gene expression profile data and select representative characteristic genes or combinations of characteristic genes. This article explores the clustering problem of gene expression profile data from two aspects: algorithm design and simulation experiments. This method is a filtering feature selection method based on clustering and Particle Swarm Optimization (PSO). The fitness function for determining feature quality standards is defined as the ratio of the distance between classes and the distance within classes. The clustering of gene expression profiles is transformed into a combinatorial optimization problem. The overview table of PSO methods in gene expression profile clustering is shown in Table 1.

Spectral clustering is a new branch of clustering analysis. It does not need to make assumptions on the overall structure of the sample space, but also can achieve sample data clustering with arbitrary shape distribution. The research content of clustering analysis has been greatly extended. It puts forward a new idea for solving clustering problems, which is very suitable for many practical application problems, with great application potential and scientific research value [11,

12]. At first, spectral clustering algorithms are used in fields such as image segmentation, VLSI design, and computer vision. Then spectral clustering algorithms are also used in fields such as text mining and machine learning [13]. Due to its wider application compared to other clustering methods, spectral clustering is worth investing a lot of time and effort in research. However, at present, the theoretical knowledge of spectral clustering is still in its infancy and deserves further exploration and research. There are many improvements in the algorithm itself. Therefore, further optimization research has important practical significance.

PSO algorithm has simple principle, fast convergence and good robustness, but it is easy to fall into local optimum. Therefore, maintaining population diversity is the key to ensuring algorithm performance [14, 15]. In the dynamic optimization problem, it is more necessary to keep the diversity of particle population, as particles need to search for the optimal solution again when the external environment changes. If the population converges to a single region, it will significantly affect the later optimization solution [16, 17]. Therefore, due to its excellent performance in maintaining population diversity, multiple swarm strategies have become an important choice for PSO.

Table 1: Overview of particle swarm optimization methods in gene expression profile clustering.

Author and year of publication	Year	Methods used	Key findings	Identified SOTA gaps
Sun et al. [5]	2022	Improved quantum behavioral PSO Algorithm	Improved global search capability and convergence speed, higher clustering accuracy than traditional PSO and other algorithms	Further algorithm optimization is necessary to handle larger datasets
Lam et al. [6]	2023	Combining Self-Organizing Mapping (SOM) and PSO	Effectively reduce data dimensionality and improve clustering accuracy and robustness	Improvements required for processing real-time data and dynamic changes
Ji et al. [7]	2023	Combining PSO and modulo algorithms	Optimizing the initial clustering centers improves the global optimality of the clustering results	The complexity of modeling algorithms may affect computational efficiency and requires simplification and optimization
Gad [8]	2022	PSO-based clustering technique	Significantly increased the use of PSO in data clustering	Insufficient application in dealing with complex problems
Rezazadeh et al. [9]	2021	Evaluation of multiple PSO variants	Different variational operators and inertia weight parameters are introduced to improve the performance of PSOs	Need to explore more promising PSO variants
Shami et al. [10]	2021	Integration of PSO with practical applications	Demonstrates the diverse variants of PSO and their accuracy in different areas	Lack of statistical information on standard PSOs in different contexts

2 Particle swarm optimization in gene expression clustering

2.1 Obtaining gene expression profile data

Gene expression data indirectly measure the abundance of gene transcription products mRNA in cells. These data can be used to analyze which gene expression has changed, what correlation between genes, and how gene activity is affected under different conditions. They have important applications in medical clinical diagnosis, drug efficacy assessment, and revealing disease pathogenesis [18]. At present, the main methods to detect gene mRNA abundance are cDNA microarray and oligonucleotide micro-array. Oligonucleotide chip is also known as gene chip and DNA chip [19].

For oligonucleotide chip data, some are P/A/M, and some are fluorescence intensity. In general, the fluorescent labeling method is selected to obtain gene expression profile data. The process of obtaining gene expression profile data is shown in Figure 1.

The most prominent feature of the expression profile chip is its huge high-throughput effect. A large number of gene probes are densely arranged on the surface of the chip base. The samples matched with the gene probe can be detected through complementary base pair recognition [20]. This chip can simultaneously analyze a large number of genes, achieving large-scale detection of biological genetic information.

When gene chipsare used to measure gene expression level, each chip is composed of thousands of points, denoted as $q_a = \{a = 1, 2, \ldots, n\}$. Each point represents a gene. A gene chip can measure gene expression value of a samplein gene set $\{q_a\}$. In this way, the value of the n -th point on chip a is expressed by z_{an} , that is, the expression value of the n -th gene in

sample a. Multiple bases are determined, as the data values of the chip are combined into a matrix.

$Q=\{q_1,q_2,...,q_n\}$ represents a gene set composed of all genes in a sample. $q_a\{1\le a\le n\}$ represents one gene. $|Q|=n$ represents the number of all genes. $r_a\{1\le a\le m\}$ represents the sample set composed of the obtained gene expression profile data. $|R|=m$ represents the number of samples.Each sample $r_a\{1\le a\le m\}$ represents the expression value of all genes under certain conditions, that is, r_a is an N-dimensional space vector. Therefore, the gene expression profile data constitutes a m-row (gene), and n-column (sample) data. The matrixis $Q=m*n$, in which $m<n$. The gene expression matrix composed of gene expression profile data is shown in Figure 2.

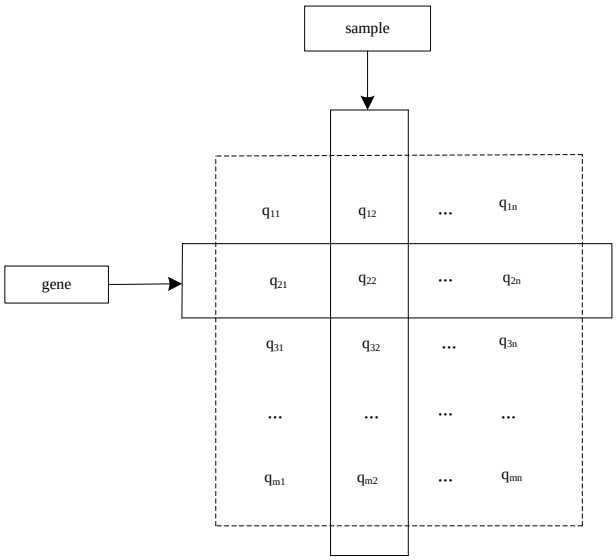


Figure 2: Gene expression profile matrix.

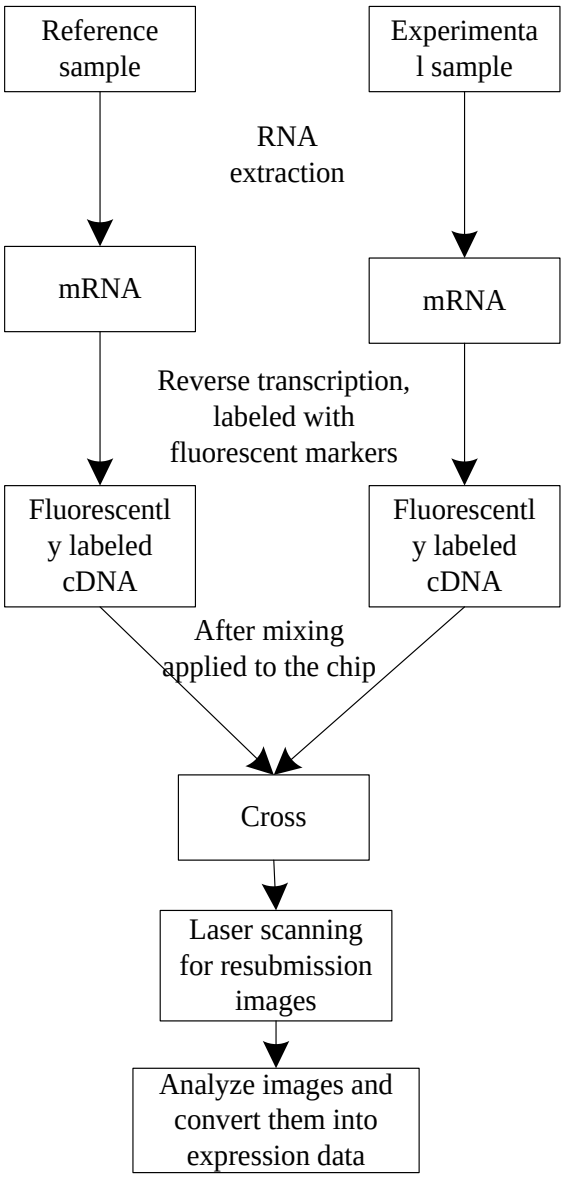


Figure 1: Gene expression profile acquisition process.

With the development of DNA chip technology, it is easier to obtain gene expression profile data, but there are still many problems in the process of data processing and analysis. First of all, there are some errors in the data collection process, such as differencesin fluorescence group performance and errors caused by uneven glass slide surfaces due to dust. Therefore, it is inevitable that there are noises and abnormal expression values in the data. Secondly, the large scale of the data makes many algorithms inefficient, which requires designing effective algorithms with low complexity. The most significant feature of array data is high-dimensional small samples. Generally, there are only a few dozen tumor related genes, but there are many redundant genes in the original data. Therefore, it is necessary to preprocess the acquired gene expression profile data to ensure the quality of gene expression profile data and improve the dataavailability.

2.2 Preprocessing of gene expression profile data

In the process of chip processing, human errors and system errors are inevitable. At the same time, to ensure the comparability of gene expression level, and make gene expression value as close to the same order of magnitude as possible, sample data needs to be preprocessed. Gene expression profile data preprocessing can be divided into three parts: abnormal data correction, lost data repair, and data conversion. The most important one is the data conversion process, which is the data standardization process. Its purpose is to shield some systematic errors in data analysis and improve the availability of gene expression profile data.

The gene expression profile data are standardized by the formula as follows:

$$f_{xy} = \frac{g_{xy} - \bar{g}_x}{\varsigma_x} \tag{1}$$

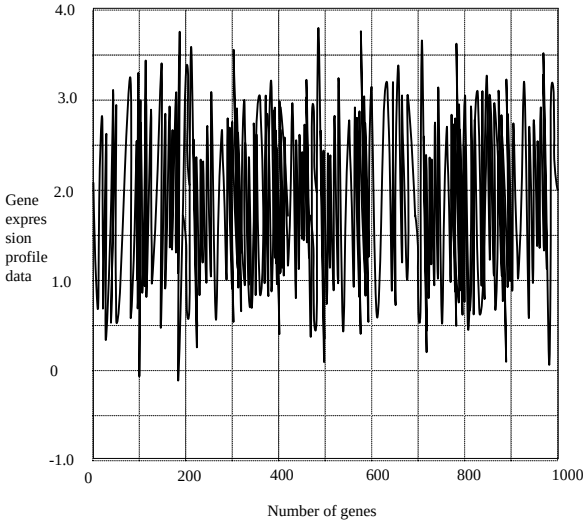
In formula (1), g_{xy} represents the expression value of the x gene in the y gene expression data sample. $\bar{g}_x$ represents the mean value of the x gene in all gene expression data samples. ς_x represents the mean square deviation of the x gene in all samples, as follows:

$$\bar{g}_x = \frac{1}{m}\sum_{y=1}^{m} g_{xy} \tag{2}$$

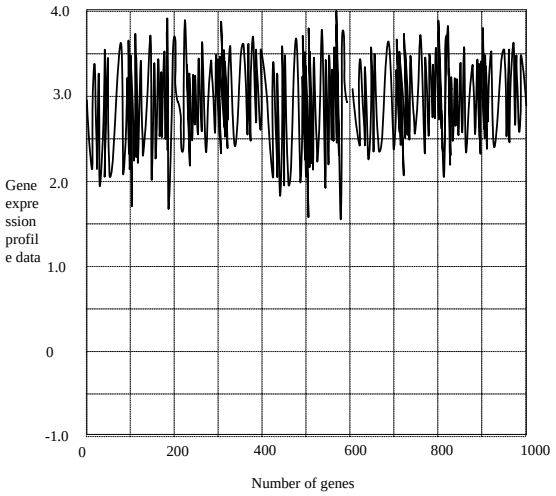
$$\varsigma_x = \sqrt{\frac{1}{m}\sum_{y=1}^{m}\left(g_{xy} - \bar{g}_x\right)^2} \tag{3}$$

The distribution of gene expression profile data before and after standardization is shown in Figure 3.

From Figure 3, the highest order of expression value of a few genes is 10^4, while the expression value of most genes is 10^2 or even smaller before the standardization of gene expression profile data. The difference of these orders of magnitude will make the genes have a priori implied weight, and if the data are clustered directly, it will produce a large noise interference, making the calculation more complex. However, after standardized processing, most of the data distribution is in 10^3, and the difference is small on the order of magnitude, which meets the needs of subsequent clustering calculation.



(a) Distribution of gene expression profile data before normalization



(b) Normalized distribution of gene expression profile data

Figure 3: Distribution of gene expression profile data before and after normalization.

2.3 Particle swarm optimization

2.3.1 Improvement of inertia weight

Inertia weightsare mainly divided into two categories: fixed weight and time-varying weight. Fixed weight is a fixed value that does not change in size, while time-varying weight is a constant change of inertia weight in a certain range according to a certain rule with the increase of iteration steps [21]. Fixed weight keeps the search ability of the algorithm unchanged, allowing for a fixed larger value and giving the algorithm strong global search capability. Time varying weights can ensure that the algorithm has different search capabilities at different periods. For example, in the early stage, the larger inertia weight ensures the global search of the algorithm. In the later stage, the algorithm should gradually converge, and the smaller inertia weight makes the algorithm have strong local search ability.

In order to analyze the advantages and disadvantages of different inertia weight strategies, professional software is used to calculate the curves of different inertia weight strategies as the number of iterations increases, as shown in Figure 4.

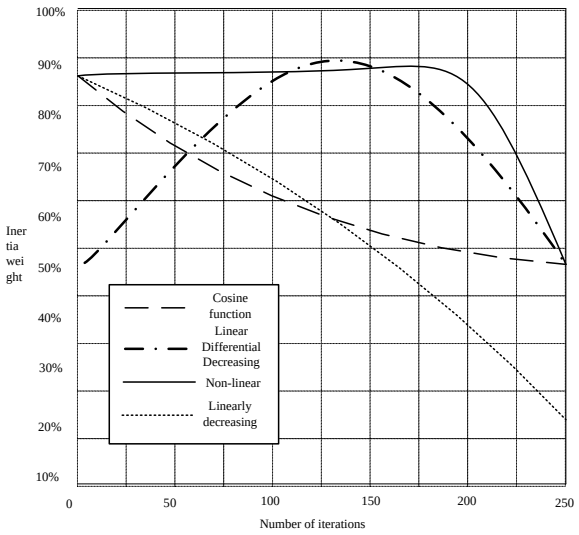


Figure 4: Curves of different inertia weights.

From the figure, for cosine function, linear differential decreasing, linear decreasing, and non-linear changes, the inertia weight decreases from the maximum value to the minimum value. Among them, the linear decreasing, is easier to fall into the local optimum than the other three, and the inertia weight decreases faster. If the global optimum is not found in the previous period, it will not jump out in the later period. Compared with the other three, the linear differential decreasing is similar to cosine function, which has strong global search ability. Cosine function has stronger global search ability, while non-linear is a compromise method. It ensures the accuracy of both global and local searches, but also results in slow convergence speed. The sine function is first searched locally, then globally, and finally locally. The tangent function decreases too quickly in the early stages, which can easily lead to the algorithm searching for the global optimal solution. The arctangent function value drops to zero in the specified steps, so the probability of the algorithm to search for the optimal solution is small.

In view of this, the non-linear inertia weight strategy is selected. The global optimal fitness value is used to replace the average fitness value of each particle. The improved inertia weight formula is defined as:

$$\eta = \eta_{\max} + (\eta_{\max} - \eta_{\min}) \times \exp\left(-25 \times \frac{\delta}{\delta_{\max}}\right)^2 \quad (4)$$

$$\eta_{\max} - (\eta_{\max} - \eta_{\min}) rand(\ ) \quad (5)$$

When the average fitness value of particles is not equal to 0, the inertia weight is calculated by formula (4). When the average fitness value of particles is equal to 0, the inertia weight is calculated by formula (5). The improved nonlinear inertia weight strategy is used for global search.

2.3.2 Improvement of learning factors

According to the actual needs of gene expression profile clustering, PSO algorithm with fixed and unnecessary inertia factor can converge to an optimal solution at a relatively fast speed. In PSO algorithm with inertia factor ε, the search for the optimal solution is determined by two random acceleration factors λ_1 and λ_2 (self-learning factor and social learning factor) [22-24]. Two learning factors are adjusted in real-time to improve the efficiency of the algorithm and find the optimal solution. If the self-experience learning factor λ_1 is larger than the social experience learning factor λ_2, the search time for a single particle is longer than the global search time. This leads to particle swarm searching in space for a long time. If the social experience learning factor λ_2 is larger than the self-learning experience factor λ_1, the whole searching time of particle swarm is longer than the single particle searching time, which will lead to the premature convergence of particle swarm. At a local optimal value, the self-learning factor λ_1 and social learning factor λ_2 are set to 2, which will reduce the search time by half.

Considering that in PSO, particles need to search in the whole space. In order to avoid falling into local optimization and improve the efficiency of PSO, it is necessary to introduce time-varying acceleration factor into the learning factor. In order to improve the global search ability of particles in the early search process, the PSO algorithm considering time-varying acceleration factors can enable the particle swarm to converge to the global optimal solution. This method reduces its own experience parameters and improves group learning experience. λ_1 and λ_2 are no longer the specified value of 2, but change over time. In the initial stage, $\lambda_1 > \lambda_2$, particles tend to be local optimal. In the later stage, $\lambda_1 < \lambda_2$, particles tend to be global optimal. For the improvement of learning factors λ_1 and λ_2, the following formula is obtained:

$$\lambda_1 = (\lambda_{1c} - \lambda_{1i}) \frac{i}{i_{\max}} + \lambda_{1i} \quad (6)$$

$$\lambda_2 = (\lambda_{2c} - \lambda_{2i}) \frac{i}{i_{\max}} + \lambda_{2i} \quad (7)$$

In formulas (6) and (7), λ_{1c}, λ_{1i}, λ_{2c} and λ_{2i} are constants. i is the number of iterations. $i_{\max}$ is the maximum number of iterations in the calculation. Through the above process, the optimal solution is found and applied to gene expression profile clustering.

2.4 Gene expression profile clustering

2.4.1 Feature extraction of gene expression profile data

It can be seen intuitively that in the feature space, if the pattern distribution of the same class is relatively dense, and the pattern of different classes is far away, gene expression profile clustering is relatively easy to achieve. Therefore, this requirement should be taken into account when selecting and extracting features from actual objects, which will bring great convenience to

subsequent clustering. However, due to some practical reasons, the selected and extracted features can not meet the above requirements significantly.

In order to ensure the required clustering accuracy and save resources, it is hoped to achieve the required clustering accuracy based on the least features [25]. Therefore, after getting some specific characteristics of the real object, the most effective and least number of features are generated from these original features, which is the task of feature selection and extraction. Essentially, the purpose of feature selection and extraction is to ensure that in the minimum dimension feature space, different pattern points are far away from each other (with large distance between classes), while similar pattern points are close to each other (with small distance between classes). In order to achieve the above goals, it is often necessary to establish criteria for feature selection and extraction. It can be directly based on functionsthat reflect the distance between classes within a class, or based on criteria related to the probability of misjudgment. The specific implementation process of feature extraction and selection is as follows:

When the number of features w actually used for clustering is given, features $w_1, w_2, ..., w_m$ are directly selected from the original features n already obtained, so that the value of separability data meets the following formula:

$$U\left(w_1, w_2, ..., w_m\right) = \max\left[U\left(w_{j1}, w_{j2}, ..., w_{jm}\right)\right] \quad (8)$$

In the formula, $w_{j1}, w_{j2}, ..., w_{jm}$ represents the w features selected from the original features n. Based on this, the w-dimensional subspace of n-dimensional space is directly found. Under the condition that the judgment basis U is the largest target, the dimension of original features n is reduced by transformation, that is to say, coordinate exchange is carried out for the original dimension feature space n, and then the subspace is taken. $l_1, l_2, ..., l_n$ is a base of n-dimensional feature space. The vector is an observation value of the target in the feature space, as expressed as follows:

$$v = \sum_{i=1}^{n} \left(l_i\right)^2 \quad (9)$$

Each component l_i of v on the base $l_1, l_2, ..., l_n$ is called an eigenvalue of the target.

The essence of feature extraction is to find a subspace S in the feature space. The new features of the object are obtained by projecting the vector v into the subspace. It is assumed that the resulting subspace is a m dimensional subspace, which is composed of m linearly independent vectors v, namely:

$$S = Span\left(v'_1, v'_2, ..., v'_m\right) \quad (10)$$

Assuming that v_m is orthogonal, the new features of the target in the subspace S can be given by the projection $t_i = v^T l_i$ of the vector v on $v'_1, v'_2, ..., v'_m$.

$$\hat{v} = \sum_{i=1}^{n} t_i v'_i \quad (11)$$

The obtained $\hat{v}$ is an approximation of the vector v in the original space. The feature components of gene expression profileobtained by the above process are the clustering targets.

2.4.2 Gene expression profile clustering based on particle swarm optimization

The optimized PSO algorithm is used to cluster gene expression profiles. In PSO algorithm, particle coding is mainly used to update the clustering center [26]. The gene expression data contains h clustering centers. h represents the number of gene expression clustering, which is used as the parameter input of the algorithm. If the clustering data is n-dimensional vector, each particle is an $h * n$ matrix, as follows.

$$K = h * n = \begin{bmatrix} \omega_{11} & \omega_{12} & ... & \omega_{1h} \\ \omega_{21} & \omega_{22} & ... & \omega_{2h} \\ ... & ... & ... & ... \\ \omega_{n1} & \omega_{n2} & ... & \omega_{nh} \end{bmatrix} \quad (12)$$

From the formula, there are h columns in particle K, which represent h cluster centers. Each cluster is also a vector of n dimension. A certain number of particle swarm needs corresponding number of matrices to store.

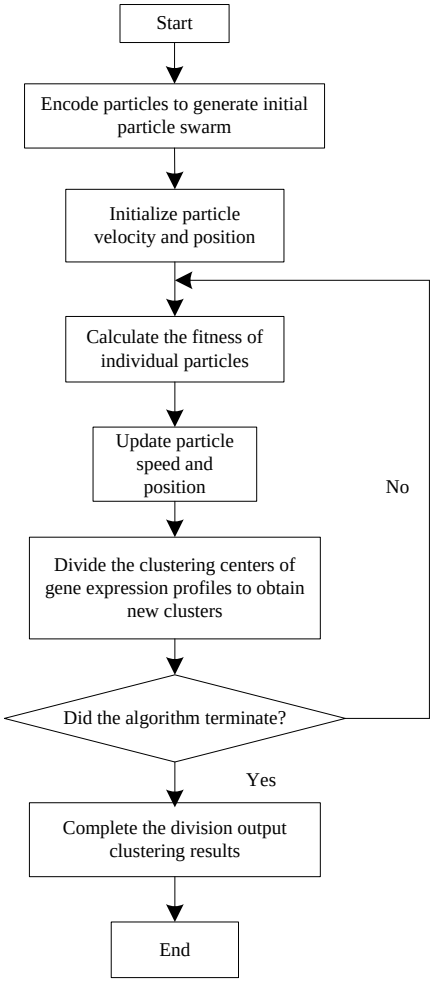


Figure 5: Cluster execution flowchart.

Given the initialization speed and position of each particle, the global optimal position and individual optimal position of the particle are determined. The fitness function in the algorithm is calculated. $fitness(\delta_1, \delta_2, ..., \delta_e)$ represents the fitness of the algorithm. B_i represents the number of clustering samples. n_b represents the sample data in the clustering. Then the fitness function is defined as:

$$fitness(\delta_1, \delta_2, ..., \delta_e) = \sum_{i=1}^{e} \sum_{b=1}^{B_i} |n_b - \delta_e| \quad (13)$$

According to the above formula, the fitness function is calculated. When the algorithm finds the minimum value of the objective function, the particle position is the optimal position after updating. The cluster centers of each cluster are output. The implementation process of PSO algorithm for clustering is shown in Figure 5.

The termination condition of the algorithm shown in the figure is that the global optimal solution no longer changes or reaches the maximum number of iterations. When the algorithm meets the above conditions, the final clustering result will be output. Otherwise, it will return to the second step to continue the iteration. So far, gene expression profile clustering has been completed. In order to better study the application performance of PSO algorithm, relevant experiments are designed to verify it.

3 Experimental design and analysis

3.1 Gene expression data set preparation

In order to better verify the clustering ability of gene expression profile of the above-mentioned design, the experiment adopts the comparative experiment. The experimental data set adopts three types of gene expression data sets with external standards. One is the yeast cell cycle data set. Yeast is one of the eukaryotes that have all the gene sequences detected by human beings, which plays a huge role in the human biological information. Cellcycle_237 and Cellcycle_384 are used as experimental data sets. The second is the yeast GAL data set. Yeast GAL is the yeast galactose utilization pathway. There are certain conditions for the expression of galactose, that is, there is no other inhibitory sugar. Otherwise, it will not be expressed. This is a classic example with genetic control switch. It can be used as experimental data to enhance the persuasion of experimental results. The third is the development data set of rat central nervous system. The central nervous system is mainly composed of brain and spinal cord, which is the most important part of mammalian nervous system. It is very important to explore the central nervous system, which is used as one of the experimental data. Four functional classes are used as the external standard of clustering. Specifically, the Cell Cycle-237 and Cell Cycle-384 datasets are used to study gene expression changes during the cell cycle. The GAL dataset is used to study galactose pathway gene expression, and the CNS dataset is used to study nervous system development. These datasets are chosen because they are representative, widely used in gene expression analysis studies, which can fully reflect the performance of the algorithms under different biological conditions. The statistical results are shown in Table 2 and Table 3.

Table 2: Experimental dataset properties.

Data set	Number of data	Data dimension	Number of standard classes
Cellcycle-237	237	17	4
Cellcycle-384	384	17	5
GAL	205	80	4
CNS	112	9	4

Table 3: Number of data set classifications

Data set	Cellcycle-237	Cellcycle-384	GAL	CNS
Cluster1	49	67	83	25
Cluster2	31	135	15	59
Cluster3	18	75	93	27
Cluster4	139	52	14	21
Cluster5	-	55	-	-

The initial parameter values of the algorithm are determined uniformly preprocessing the above data:

Particle swarm size: 30;
Number of iterations: 1000;
Learning factors: 1;
Inertia factor: maximum 0.9, minimum 0.4;
Maximum speed: 15% of search space;
Disturbance stop algebra: 15.

The parameter selection of PSO is based on the comprehensive consideration of algorithm performance and computational efficiency. The particle swarm size is set to 30 to control the computational complexity while ensuring the algorithm diversity. The number of iterations is set to 1000 to ensure that the algorithm can fully search the solution space to find the global optimal solution. In addition, the learning factor is set to 1, and the inertia coefficient is varied between 0.9 and 0.4 to balance the ability of global and local search and avoid falling into local optima.

3.2 Experimental environment and parameter design

In the experiment, 64-bit windows operating system is selected, with CPU of 2.5GHz and memory of 64G. A complete comparative experiment is conducted using the Eclipse integrated development environment, algorithm development language Java, and statistical analysis software SPSS to ensure the normal operation of the experiment. The laboratory environment is shown in Figure 6.

Figure 6: Laboratory environment.

3.3 Experimental results and analysis

In the dataset preparation, some data from the Cellcycle-237 dataset is used as sample 1, some data from the GAL dataset is used as example 2, and another part of data from the CNS dataset is used as sample 3. The above samples are iterated for 5, 10, 50 and 100 times respectively to update the markers of each sample and observe the changes in each component of the marker sequence. At the same time, the traditional clustering method is used to carry out the experiment under the same conditions, according to the different experimental results.

The experimental results of Cellcycle-237 are shown in Figure 7.

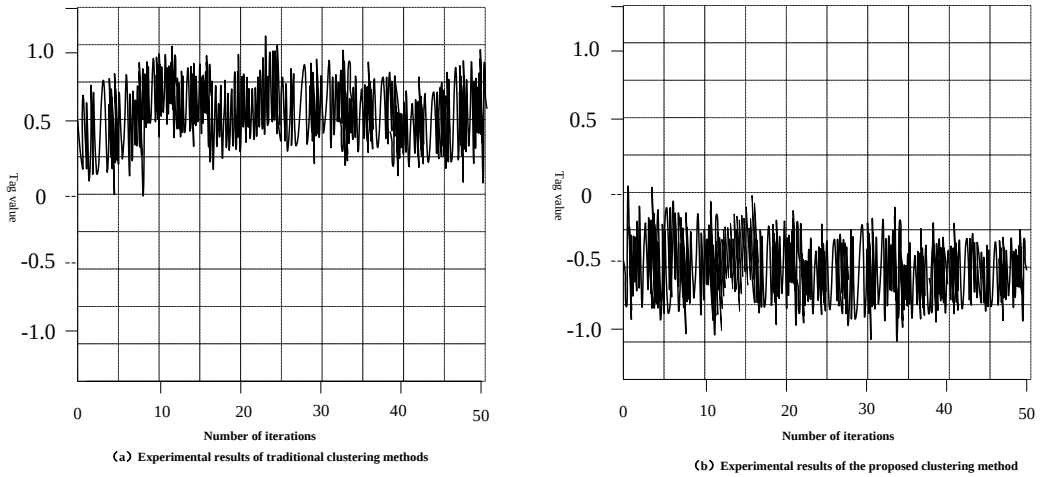


Figure 7: Sample 1 experimental results.

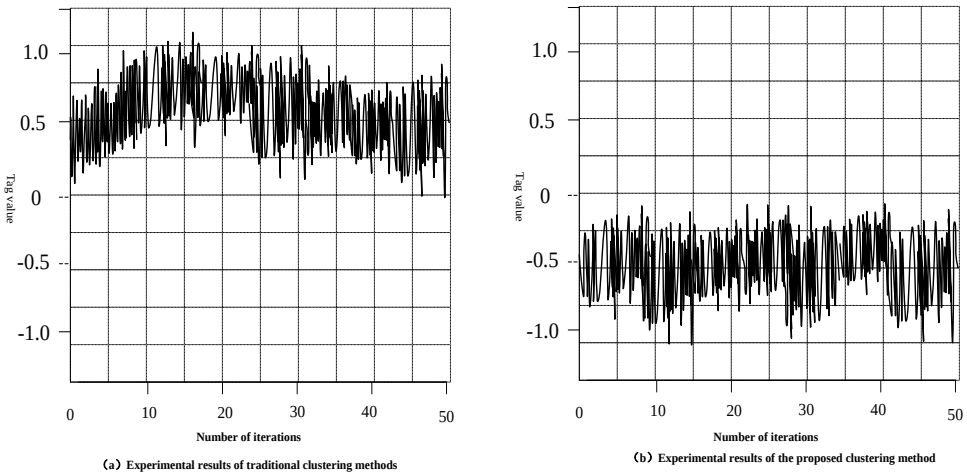


Figure 8: Sample 2 experimental results.

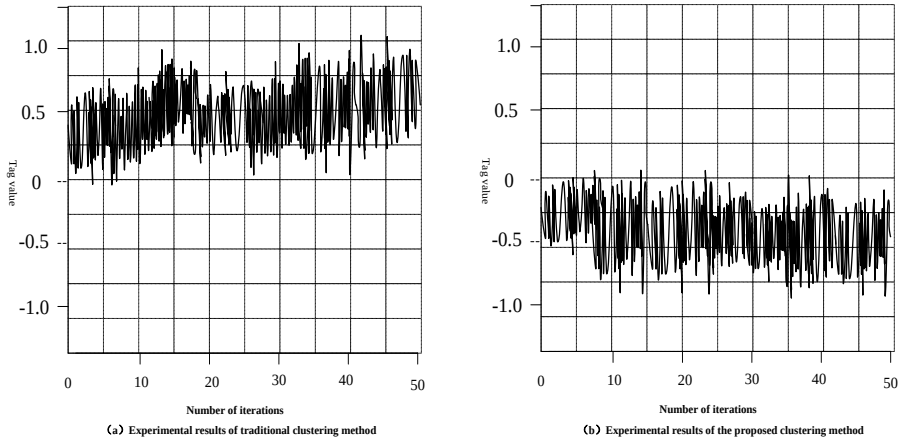


Figure 9: Sample 3 experimental results.

The experimental results of GAL data sample are shown in Figure 8.

The experimental results of CNS data samples are shown in Figure 9.

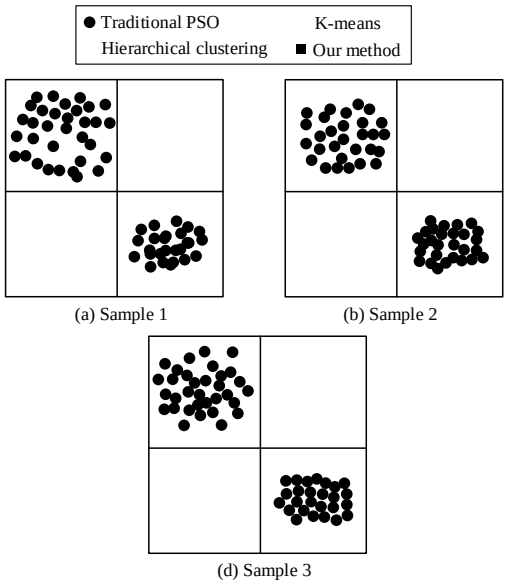


Figure 10: Visual clustering test results for different methods.

From the above three results, with the increase of the number of iterations, the tag value of the proposed clustering method consistently remained above 0, approaching 1. The differences between gene expression profile tag values are evident, indicating a superior clustering effect. The tag value of the traditional clustering method experiment was always less than 0, approaching to -1, which showed that the tag information of the tag points was preserved in each iteration, and the clustering ability was poor. In conclusion, the proposed PSO algorithm for gene expression profile clustering is superior to traditional gene expression profile clustering. The study conducts a visual clustering test in the form of a clustered molecular graph. The results are shown in Figure 10.

Figure 10(a) shows the visualized clustering results of the four methods for sample 1. Figure 10(b) shows the visualized clustering results of the four methods for sample 2. Figure 10(c) shows the visualized clustering results of the four methods for sample 3. From the results of the three tests, compared with traditional PSO algorithm, K-means clustering algorithm and hierarchical clustering method, the designed method had more concentrated clustering effect and better sorting effect. The improved PSO algorithm shows more obvious clustering effect and higher sample aggregation. Taking clustering accuracy, contour score and computation time as the indexes, the results of comparative testing of the previous proposed methods, K-means clustering method and hierarchical clustering method are continued, as shown in Table 4.

Table 4: Indicator test results of different methods

Methods	**Clustering accuracy (%)**	**Contour scoring**	**Calculation time (seconds)**
Improved QPSO [5]	92.45	0.85	120.51
Combining SOM and PSO [6]	90.37	0.86	130.75
Combining PSO and modal algorithm [7]	91.55	0.83	125.42
Traditional PSO	85.25	0.75	140.61
K-means	80.15	0.73	110.82
Hierarchical clustering	82.13	0.72	135.53
The PSO algorithm proposed in this paper	95.77	0.91	100.07

From Table 4, the test results showed that the improved QPSO method performed the best in terms of clustering accuracy and contour scoring, reaching 92.45% and 0.85, respectively, but its computation time was longer, at 120.50s. The method combining SOM and PSO, and the method combining PSO and modal

algorithm, had clustering accuracy of 90.30% and 91.55%, respectively, and high contour scores, but the computation time was slightly longer. In contrast, the traditional PSO, K-means and hierarchical clustering methods are inferior to the improved PSO method in terms of clustering accuracy and contour scores. The proposed PSO algorithm had a clustering accuracy of up to 95.77%, a contour score of up to 0.91, and the shortest computation time of 100.07s. This indicates that the improved PSO method not only improves clustering accuracy but also maintains more stable clustering performance and shortens computation time.

4 Discussion

In this paper, an improved PSO algorithm is proposed for clustering gene expression profile data. The improved PSO algorithm enhances the global and local search capability by increasing the inertia weights and learning factors to avoid falling into local optima. Three classes and four sets of gene expression datasets, including cell cycle, GAL, and CNS datasets, are used in the experiments to evaluate the performance of the improved PSO and other clustering methods in terms of clustering accuracy, profile scores, and computation time. The results in Table 4 showed that the improved PSO algorithm had the highest clustering accuracy of 95.77%, the highest contour score of 0.91, and the shortest computation time of 100.07 s, which all outperformed the traditional PSO, K-means, and hierarchical clustering methods. The difference in results mainly stems from the trade-off between the global and local search capabilities of different methods, as well as the optimization of parameter settings. For example, the QPSO method proposed by Sun et al. [5] had a clustering accuracy of 92.45% and a contour score of 0.85, but a longer computation time of 120.50 s. The method combining SOM and PSO proposed by Lam et al. [6] improved the clustering accuracy to 90.37% and the contour score to 0.86, but the computation time was even longer at 130.75 s. This is a good example of how the QPSO method can improve clustering accuracy and contour scores. In contrast, the PSO algorithm proposed in this paper significantly improved the clustering results by optimizing the inertia weights and learning factors while reducing the computation time. The PSO-based clustering technique proposed by Gad [8] showed excellent performance in terms of application simplicity and effectiveness, but its application in dealing with complex problems is insufficient. Rezazadeh et al. [9] evaluated various PSO variants, and pointed out that introducing different variant operators and inertia weight parameters improved PSO performance, but further exploration of promising PSO variants was necessary. Shami et al. [10] demonstrated diverse applications of PSO and its accuracy in different domains, but lacked statistical information about standard PSO in different contexts.

In summary, the novelty of the improved PSO algorithm lies in the combination of multiple optimization strategies to improve clustering accuracy and computational efficiency, which has high practical application value. However, the current research has high algorithm complexity. Future research can further optimize the algorithm structure to reduce the computational complexity. Compared with the traditional PSO and other clustering methods, the improved PSO algorithm performs better in terms of computational complexity, and effectively improves the clustering effect despite the increase in computation time, which is consistent with the research results of Sun et al [5]. Therefore, the improved PSO algorithm proposed in this article not only significantly improves the clustering effect in gene expression profile clustering, but also exhibits better computational efficiency, providing new methods and ideas for the analysis of gene expression data.

5 Conclusions

This study applied PSO algorithm to gene expression profile clustering by improving the inertia weights and learning factors of PSO. This method solved the problems in traditional gene expression profile classification, which had profound significance for clustering research of gene expression profiles. Although the expression profile clustering of PSO algorithm has achieved the expected results, there are still some limitations, which need further research and discussion.

Fundings

The research is supported by: Changzhou Vocational Institute of Mechatronic Technology, the 2023 school level key educational reform project "Research on the Construction of an Innovation and Entrepreneurship Education Ecological System for Higher Vocational Undergraduate Colleges with Mutual Benefit and Win WinWin among Government, Enterprises, Schools, and Students" (Project Number: 2023-JGZD-23).

References

[1] MdSoriful Islam, Clara Castellucci, RosamariaFiorini, Stefania Greco, Riccardo Gagliardi, Alessandro Zannotti, Stefano R. Giannubilo, Andrea Ciavattini, Natale G. Frega, Deborah Pacetti, and Pasquapina Ciarmela. Omega-3 fatty acids modulate the lipid profile, membrane architecture, and gene expression of leiomyoma cells. Journal of cellular physiology, 233(9):7143-7156, 2018. https://doi.org/10.1002/jcp.26537

[2] Devara Divya, Tarun Kumar Bhattacharya, Manthani Gnana Prakash, R. N. Chatterjee, Renu Shukla, PothanaBoyina Guru Vishnu, AmirthalingamVinoth, and KothaDushyanth. Molecular characterization and expression profiling of BMP 3 gene in broiler and layer chicken. Molecular Biology Reports, 45(4):477-495, 2018. https://doi.org/10.1007/s11033-018-4184-x

[3] Bilal Khomri, ArgyriosChristodoulidis, Leila Djerou, Mohamed ChaoukiBabahenini, and Farida Cheriet. Particle swarm optimization method for

small retinal vessels detection on multiresolution fundus images. Journal of biomedical optics, 23(5):056004, 2018. https://doi.org/10.1117/1.jbo.23.5.056004

[4] Ali Adeli and Ali Broumandnia. Image steganalysis using improved particle swarm optimization-based feature selection. Applied intelligence, 48(6):1609-1622, 2018. https://doi.org/10.1007/s10489-017-0989-x

[5] Jun Sun, Wei Chen, Wei Fang, XiaojunWun, and Wenbo Xu. Gene expression data analysis with the clustering method based on an improved quantum-behaved particle swarm optimization. Engineering applications of artificial intelligence, 25(2):376-391, 2022. https://doi.org/10.1016/j.engappai.2011.09.017

[6] Yau-King Lam, Peter Wai-Ming Tsang, and Chi-Sing Leung. Improved gene clustering based on particle swarm optimization, k-means, and cluster matching. International conference on neural information processing, Part I:654-661, 2011. https://doi.org/10.1007/978-3-642-24955-6_77

[7] Zhen Ji, Wenmin Liu, Zexuan Zhu. Gene clustering using particle swarm optimizer based memetic algorithm. International conference on advances in swarm intelligence, Part I:587-594, 2011. https://doi.org/10.1007/978-3-642-21515-5_69

[8] Ahmed G. Gad. Particle swarm optimization algorithm and its applications: a systematic review. Archives of computational methods in engineering, 29(5): 2531-2561, 2022. https://doi.org/10.1007/s11831-021-09694-4

[9] Iman Rezazadeh, Mohammad Reza Meybodi, and Ahmad Naebi. Adaptive particle swarm optimization for dynamic environments. Advances in swarm intelligence, Part I:120-129, 2011. https://doi.org/10.1007/978-3-642-21515-5_15

[10] Tareq M. Shami, Ayman A. El-Saleh, Mohammed Alswaitti, Qasem Al-Tashi, Mhd Amen Summakieh, and Seyedali Mirjalili. Particle swarm optimization: A comprehensive survey. IEEE Access, 10:10031-10061, 2022. https://doi.org/10.1109/ACCESS.2022.3142859

[11] Elnaz Pashaei, Elham Pashaei, and Nizamettin Aydin. Gene selection using hybrid binary black hole algorithm and modified binary particle swarm optimization. Genomics, 111(4):669-686, 2019. https://doi.org/10.1016/j.ygeno.2018.04.004

[12] Przemyslaw Spurek, Jacek Tabor, K. Byrski. Active function cross-entropy clustering. Expert systems with applications, 72:49-66, 2017. https://doi.org/10.1016/j.eswa.2016.12.011

[13] Marjan Abdeyazdan. A new method for the informed discovery of resources in the grid system using particle swarm optimization algorithm (RDT_PSO). The journal of supercomputing, 73(12):5354-5377, 2017. https://doi.org/10.1007/s11227-017-2090-y

[14] M.A. Hannan, Mahmuda Akhtar, R.A. Begum, H. Basri, A. Hussain, Edgar Scavino. Capacitated vehicle-routing problem model for scheduled solid waste collection and route optimization using PSO algorithm. Waste management, 71:31-41, 2018. https://doi.org/10.1016/j.wasman.2017.10.019

[15] Ayman Khlif, and Max Mignotte. Segmentation data visualizing and clustering. Multimedia tools and applications, 76(1):1531-1552, 2017. https://doi.org/10.1007/s11042-015-3148-6

[16] Gianluca Santoni, Ulrich Zander, Christoph Mueller-Dieckmann, Gordon Leonard, and Alexander Popov. Hierarchical clustering for multiple-crystal macromolecular crystallography experiments: the ccCluster program. Journal of applied crystallography, 50(Pt6):1844-1851, 2017. https://doi.org/10.1107/S1600576717015229

[17] Hosik Choi, Seokho Lee. Convex clustering for binary data. Advances in data analysis and classification, 13(4):991-1018, 2019. https://doi.org/10.1007/s11634- 018-0350-1

[18] Krishna Gopal Dhal, Arunita Das, Swarnajit Ray, Sanjoy Das. A clustering-based classification approach based on modified cuckoo search algorithm. Pattern recognition and image analysis, 29(3):344-359, 2019. https://doi.org/10.1134/S1054661819030052

[19] Peter Laurinec, and Maria Lucká. Interpretable multiple data streams clustering with clipped streams representation for the improvement of electricity consumption forecasting. Data mining and knowledge discovery, 33(2):413-445, 2019. https://doi.org/10.1007/s10618-018-0598-2

[20] Zola Donovan, Gregory Gutin, Vahan Mkrtchyan, and K. Subramani. Clustering without replication in combinatorial circuits. Journal of combinatorial optimization, 38(2):481-501, 2019. https://doi.org/ 10.1007/s10878-019-00394-1

[21] Melogy Xuan Lim, Kieran A. Murphy, Heinrich M. Jaeger. Edges control clustering in levitated granular matter. Granular Matter, 21(3):77, 2019. https://doi.org/ 10.1007/s10035-019-0926-2

[22] Yirui Song. Optimization of quantitative research methods in social sciences in the era of big data. Acta informatica Malaysia, 7(2):92-96, 2023. https://doi.org/10.26480/aim.02.2023.92.96

[23] Samuel Ro Paian Purba, Harummi Sekar Amarilies, Nur Layli Rachmawati, Anak Agung Ngurah Perwira Redi. Implementation of particle swarm optimization algorithm in cross-docking distribution problem. Acta informatica Malaysia, 5(1):16-20, 2021. https://doi.org/10.26480/aim.01.2021.16.20

[24] Mahassin Mohamed Ahmed Osman, Sharifah Kamilah Syed Yusof, and Nik Noordini Nik Abd Malik. Impact of channel heterogeneity on clustering formation in cognitive ad hoc radio networks. Wireless personal communications, 96(3):4613-4627, 2017. https://doi.org/10.1007/s11277-017-4074-9

[25] Walaa Khalaf, Annabella Astorino, Pietro D'Alessandro, and Manlio Gaudioso. A DC optimization-based clustering technique for edge detection. Optimization letters, 11(3):627-640, 2017. https://doi.org/10.1007/s11590-016-1031-7

[26] Sergey M Melnikov, and Matthias Stein. Molecular dynamics study of the solution structure, clustering, and diffusion of four aqueous alkanolamines. The journal of physical chemistry B, 122(10):2769-2778, 2018. https://doi.org/10.1021/acs.jpcb.7b10322