Hindawi
Mobile Information Systems
Volume 2022, Article ID 9554453, 8 pages
https://doi.org/10.1155/2022/9554453

Research Article

High Dimensional Data Processing Based on Optimized DPC Algorithm

Peiji Yu,[1] Peiyuan Wang,[2] and Hongtao Ji[2]

[1] *Wuchang University of Technology, Wuhan, China*
[2] *Wuhan University, Wuhan, China*

Correspondence should be addressed to Hongtao Ji; 2021286270041@whu.edu.cn

Received 16 June 2022; Revised 27 July 2022; Accepted 1 August 2022; Published 30 August 2022

Academic Editor: Muhammad Zakarya

With the rapid growth of science and technology, machine learning and big data analysis have developed more and more difficult. Similarly, data also become difficult to process and classify due to the fact that the data dimension is becoming larger and larger. Furthermore, aiming at the defect due to the amount of data the clustering using speedy examine, search, and discovery of density peaks (also known as DPC) does not adjust to data sets with large dimensions (high-dimensional). Therefore, in this paper, we suggest an optimization procedure, which we pronounce as t-dpc, and is founded on the t-sne dimension lessening technique, and which can also optimize the technique for estimation of the Gaussian kernel function, using unified measurement criteria in solving density. In the simulation based experiments, using two different data sets i.e., the UCI standard data set, and the artificial data set, the proposed DPC procedure is associated with the classical t-dpc algorithm. The empirical evaluation and investigational outcomes illustrate that the proposed method of t-dpc not merely acclimatizes to the high-dimensional data sets, nevertheless it also increases the effectiveness of the classical DPC technique.

1. Introduction

In recent years, due to the speedy growth of information networks, big data, and internet technology, the development of data presents an explosive growth mode. People's daily behavior can be quantified as data onto a certain form, but these data are often disordered and irregular. Over time, these data will accumulate more and more, and the era of big data came into being [1, 2]. One of the most widely used is Taobao. Taobao automatically pushes people's favorite products according to people's consumption habits, which not only brings consumers their favorite products but also improves the sales volume of Taobao to a certain extent. Therefore, big data analysis technology and the ability to obtain effective information on big data have become the top priority of people's research [3]. As an imperative and essential data mining technology, cluster analysis plays a precise imperative role in data mining and analysis. It has been used in many fields and plays an irreplaceable role [4].

An enormous quantity of data has been produced in the network era, and these data are often unpredictable. Therefore, it is almost impossible to label these data in advance, but cluster analysis should be carried out according to the internal relationship of the data [5]. For example, for the takeout service born in the Internet era, because businesses cannot obtain the classification attributes of users in advance, the takeout platform can only cluster the data generated by users according to users' consumption habits. Finally, the takeout platform will automatically push the takeout that users may like according to the characteristics of taste, category and evaluation [6]. Compared with the classification algorithm, the unsupervised data mining technology of clustering algorithm also saves a lot of time of the training samples, because the classification algorithm needs to experiment on the training data first, extract the characteristics of the training data, and then apply it to the test data onto processing and analysis, and the clustering algorithm can process all data objects together [7–9]. However, according to the different data objects in all walks of life, different clustering procedures have been suggested one after another. In the development and speedy growth of the clustering mechanisms and algorithms, many typical clustering algorithms have been born one after another, such as the well-known and most widely used KMeans [10], DBscan [11], FCM [12], and AP [13] algorithm. However, clustering analysis is also affected by the distribution characteristics of actual data, and various problems have been encountered in the process of clustering development.

Some algorithms cannot identify complex manifold clusters, some algorithms cannot effectively deal with noise points, some algorithms have too high time complexity to meet the timeliness of big data clustering, some algorithms have parameter sensitive problems, and there are too many factors requiring human interference. Therefore, aiming at the problems encountered in the development of this series of clustering algorithms, more and more improved algorithms proposed [14, 15]. Facing the defect that the DPC approach does not acclimatize to high-dimensional data sets, in order to resolve this difficult task. In this paper, we suggest an optimization algorithm called t-dpc. The t-dpc algorithm starts with the t-sne technique for dimension reduction, improves the estimation approach of the Gaussian kernel function, and uses unified measurement criteria to calculate the density. At the same time, the proposed t-dpc method and the DPC approach are matched to/with the F-measure index and the NMI index, in terms of, the UCI standard data set, and artificial data set, correspondingly. The concluding investigational outcomes and findings reveal that the t-dpc approach not merely acclimatizes to huge and high-dimensional data sets, nevertheless also develops the effectiveness of the classical DPC method.

The key and fundamental offerings of this research are given as follows: (1) we put forward an optimization algorithm, known as t-dpc, which is constructed on t-sne approach for dimension reduction; (2) we propose a method that also enhances the estimation mechanism of the Gaussian kernel function, using unified measurement criteria in solving density; (3) using artificial dataset and UCI standard dataset, the DPC algorithm is matched with the t-dpc algorithm; and (4) the simulation and empirical outcomes deliberate that the suggested t-dpc technique not merely familiarizes to the huge and high-dimensional data sets, nonetheless it also progresses the effectiveness of the classical DPC method.

The remaining part of this manuscript is arranged as follows: the optimized DPC method, i.e., t-dpc, and its working mechanism is offered in Section 2. The experimental simulations and empirical evaluation of high-dimensional data processing are discussed in Section 3. Moreover, the attained outcomes are also deliberated in Section 3. To conclude, Section 4 completes this study and offers several future research insights and suggestions.

2. Optimized DPC Algorithm

2.1. Algorithm Introduction. Rodriguez [16] (2014) proposes a clustering algorithm based on density peak, which has attracted many people's attention. The idea of Alex's algorithm is to calculate the similarity by taking the distance between two points of interest, in a particular data set, as a measure, which can be adapted to clusters of any shape. The distance between two data points in the aforementioned data collection, which is largely unaffected by enormous and high-dimensional data, is the fundamental concept and foundation of the classical DPC technique. The algorithm cluster center has the following characteristics: (1) each cluster's cores are separated by locally low density areas and (2) the distance between them is considerable. The DPC method introduces two variables, i.e., one is distance δ, and the other is local density ρ. For an arbitrary sample i of the data set, its local density ρ calculation is as follows:

$$\rho_i = \sum_{j\neq i} x\left(d_{ij} - d_c\right)\begin{cases} x(x) = 1, & x < 0, \\ x(x) = 0, & x \geq 0. \end{cases} \tag{1}$$

Equation (1) is a truncated kernel function. Furthermore, d_{ij} is the Euclidean distance between i and j in the sample, d_c is the truncated distance, and x is the criterion for judging that a particular value is greater than another and vice versa [17].

Han et al. [17] also suggested using the Gaussian kernel function to calculate the local density as a different way to compute it, as shown in the following equation:

$$\rho_i = \sum_{j\neq i} e^{-\left(d_{ij}/d_c\right)^2}. \tag{2}$$

The distance is defined as given by the following formula:

$$\delta_i = \min_{j:\, \rho_j > \rho_i}\left(d_{ij}\right). \tag{3}$$

The maximum local density points are

$$\delta_i = \max_{j}\left(d_{ij}\right). \tag{4}$$

The DPC algorithm, suggested in this paper, helps to obtain the peak density of decision graph. According to the calculated local density ρ and distance δ, and subsequently draw a decision diagram (ρ-δ) . In DPC algorithm, the points of large local density and distance are selected as the cluster center [18]. And then, each remaining point is assigned to the cluster where the density is higher than him and the nearest data point is located. Finally, the noise points are excluded.

2.2. Algorithm Optimization. With the rapid growth of data volume, the diversity of data is becoming stronger and stronger. As a result, most of the data onto life is rough data onto higher dimensions. However, DPC algorithm is powerless in the face of such high-dimensional data. To solve this problem, in this paper we adopt the t-sne dimension reduction mechanism which is constructed on radial transformation to systematize and normalize the data and optimize the suggested technique of forming t-dpc. The algorithmic flow of the suggested t-dpc method is as follows.

2.2.1. T-Sne Method Is Used for Data Standardization and Normalization. The input of the method are high-dimensional datasets denoted by $X = x_1, \ldots, x_n$, and the output of the approach are low-dimensional datasets which is characterized by $Y^T = y_1, \ldots, y_n$. The process is as follows.

The SNE is the conditional probability that exchanges the high-dimensional Euclidean distance amongst data points of the similarity; that is, after a high-dimensional data set is given, the conditional probability is used to represent the similarity from point to point. This meaning can be understood as follows: if the neighbor is selected by the Gaussian distribution centered on, the probability that selects as its own neighbor is. If the data points are close, they are large. On the contrary, if the data points are very far away, they can be close to infinity [19]. The parameter is the variance between as the central point, which changes from the change of position. The definition of the conditional probability is illustrated mathematically, as shown in the following formula:

$$p_{(j/i)} = \frac{\exp\left(-x_i - x_j{}^2/2\delta_i^2\right)}{\sum_{k\neq i}\exp\left(-x_i - x_k{}^2/2\delta_i^2\right)}. \qquad (5)$$

In the following formula, the conditional probability distribution of the low-dimensional data point is determined and relates to the high-dimensional data point.

$$q_{(j/i)} = \frac{\exp\left(-y_i - y^2\right)}{\sum_{k\neq i}\exp\left(-y_i - y_k{}^2\right)}. \qquad (6)$$

The above two formulas represent the similarity, so both and are 0. At that moment, the distance amongst the two distributions Kullback Leibler diversities is enhanced, and its objective function is defined as shown in the following formula:

$$C = KL\left(P_i Q_i = \sum_i \sum_j p_{(i/j)} \log \frac{p_{(i/j)}}{q_{(j/i)}}\right). \qquad (7)$$

Since, the conditional probability is not equal to, a large amount of calculation is required in gradient calculation [20]. The core of klt-e is to find the divergence of probability distribution by replacing the principle of probability E-T distribution. The optimized objective function changes, as shown in the following formula:

$$C = KL(PQ) = \sum_i \sum_j p_{(j/i)} \log \frac{p_{ij}}{q_{ij}}. \qquad (8)$$

The calculation formulas for and shown in formula (8) have also changed accordingly. After the change, the calculation formulas are respectively as follows:

$$p_{ij} = \frac{p_{(i/j)} + p_{(j/i)}}{2}, \qquad (9)$$

$$q_{ij} = \frac{\left(1 + y_i - y_j{}^2\right)^{-1}}{\sum_{k\neq 1}\left(1 + y_k - y_l{}^2\right)^{-1}}. \qquad (10)$$

2.2.2. Iterations after T-sne Optimization. Calculation under low dimension according to formula (10), and then the gradient is calculated. The gradient calculation formula is as shown in the following formula:

$$\frac{\delta C}{\delta Y} = 4 \sum_j \left(p_{ij} - q_{ij}\right)\left(y_i - y_j\right)\left(1 + y_i - y_j{}^2\right)^{-1}. \qquad (11)$$

Formula (10), for example, must be included to the gradient descent process since it is very simple for it to go into the local optimal solution in the course of the optimization progression, as illustrated in (12) formula (12). This should be noted that the innovative low-dimensional data set can be obtained rendering to the following formula:

$$Y^{(t)} = Y^{(t-1)} + \eta \frac{\delta C}{\delta Y} + a(t)\left(Y^{(t-1)} - Y^{(t-2)}\right), \qquad (12)$$

where $Y^{(t)}$ talks about the solution of the t iterations. Similarly, Y exemplifies the learning frequency and $a(t)$ characterizes the momentum of the t iterations. So far, the data standardization process is accomplished [21]. This should be highlighted here for the sake of understanding that the standardized data adapts to the DPC method while maintaining the majority of the properties of the original data set.

2.2.3. Find the Values of T-dpc, Local Density, And Standard Deviation δ. First of all, read the points of the data, and then determine the distance amongst the points, and gauge the value of t-dpc. Secondly, after manipulating the t-dpc value, protect and store the distance from a particular point j to another specified point i, that is, a reduced amount of than the ixj in the Z list to obtain a new Gaussian kernel function formula. Next we use, for instance, formula (12) to determine the local density, and use formula (3) to calculate the δ.

$$\rho_i = \sum_{j\neq i} e^{-\left(z[j]/d_c\right)^2}. \qquad (13)$$

2.2.4. Complete Clustering According to the Subsequent DPC Algorithm. The subsequent DPC algorithm includes: drawing the decision diagram, manually selecting the clustering center, assigning points, and calculating noise points. Finally, clustering can be completed.

3. Experimental Simulation of High-Dimensional Data Processing

3.1. Simulation Environment. The experiment is completed by the software pycharm, and the experimental language is *Python* 3. Note that, all the experiments were completed on a computer with hardware configuration as follow: the CPU model is i5-3337u with approximate spped of 1.80 GHz, and the system memory was 8 GB.

3.2. Data Acquisition and Pre-processing. With the purpose of confirming the legitimacy, as well as, the correctness of outcomes related to the suggested t-dpc method, we intend to select three (3) artificial data sets [22] and 5 standards [23]. The UCI data set is tested, and the parameters of various investigational data sets are specified away in Table 1. We assume that the data is in clean form and does not require any preprocessing method.

3.3. The F-Measure Metric. The F-measure index or evaluation metric is a frequently used assessment benchmark, particularly in the field of information retrieval and learning, which is weighted and balanced by precision and the recall. Furthermore, it is regarded as a manually labeled known cluster, and is the cluster designed at the completion of the clustering technique. The recall metric is also very commonly used in the learning assessment. The correctness or the accuracy rate is shown in formula (12), and the recall rate is shown in the following formula:

$$P\left(P_j, C_i\right) = \frac{\left|P_j \cap C_i\right|}{\left|C_i\right|}. \tag{14}$$

$$R\left(P_j, C_i\right) = \frac{\left|P_j \cap C_i\right|}{\left|P_i\right|}. \tag{15}$$

The comparison of F evaluation indexes of clustering results is shown in Table 2. Furthermore, Figure 1 shows the F evaluation values for both algorithms over different data sets. Note that, the higher values are better than the lower values. We can observe better values for the suggested t-dpc method as compare to the classical DPV approach.

This can be easily comprehended from the outcomes reported in Table 2 that for the three synthetic data sets, the F-measure index of DPC algorithm changes little compared with that of t-dpc algorithm. That is because it does not need to be reduced by t-sne, so the accuracy basically does not change. The change is comparatively more pronounced for high-dimensional UCI data sets. When compared to the F index before to the reduction of dimension, the values of all four data sets had significantly improved by approximately 7.0 percent, 5.2 percent, 1.1 percent, and 5.4 percent, correspondingly. Clearly, the performance of the data after t-SNE reduction method for dimension is better than the classical DPC approach.

3.4. Results of the NMI Evaluation Indicators. The idea of standard mutual information from descriptive information theory is utilised to quantify the similarity between two data distributions [24]. Presume that X and y are the distribution of n samples, as shown in the following formulas:

$$H(X) = \sum_{i=1}^{X} P(i)\log(P(i)). \tag{16}$$

$$H(Y) = \sum_{j=1}^{Y} P'(j)\log(P'(j)). \tag{17}$$

In formulas (16) and (17), $P(i) = |X_i|/N, P'$and$(j).$ $= |Y_j|/N$. This should be noted that the mutual information (MI) amongst the X and Y is illustrated mathematically as, for example, given by the following formula:

$$MI(X,Y) = \sum_{i=1}^{X}\sum_{j=1}^{Y}(i,j)\log\left(\frac{p(i,j)}{p(i)p\,'\,(j)}\right)P. \tag{18}$$

In formula (16), $P(i, j) = |X_j \cap X_i|/\mathrm{N}$. The next task is to standardize the mutual information (NMI) [25], such as illustrated mathematically in the following formula:

$$NMI(X,Y) = \frac{MI(X,Y)}{\sqrt{H(X)H(Y)}}. \tag{19}$$

The comparison results of NMI evaluation indicators are shown in Table 3. A graphical view of the attained results is shown in Figure 2. We can easily observe that the proposed t-dpc algorithm has similar values to the classical dpc algorithm; however, we noted that our method outperforms the dpc, in particular, for larger data sets.

Table 3 reveals the coincidence degree amongst the original, high-dimenstional, data set and the clustered, low-dimensional, data set. This should be observed that the NMI metric of the PID data set, wine data set, and the waveform data set under the suggested t-dpc method is lower than that which we observed under the classical DPC approach by 4.9%, 3.9%, and 3.6%, respectively. In fact, this shows that the coincidence degree of these four data sets is lower than that of the untransformed data set. Furthermore, we rose various parametric values in the residual two synthetic data sets, as well as, the iris data sets. In other word, in fact the aggregate data set and the d31 data set were increased by approximately 0.1% and 0.2%, correspondingly. Similarly, the iris data set was increased by approximately 7.3%. The R15 dataset did not alter, though. The synthesis demonstrates that when the dimension is high, the coincidence degree of the data set will drop, and when the dimension is low, the coincidence degree effect is better.

3.5. Results of Algorithm Efficiency. In our simulations, an average of 20 running periods is used as the final running time to reveal the correctness of the investigational outcomes, and the findings are displayed in Table 4. Furthermore, Figure 3 shows the running time of both algorithms over different data sets. Note that, the lower values are better than the higher values.

The assessment amongst the t-DPC approach and the classical DPC method, in terms of time efficiency, is shown

Table 1: The parameters of various investigational data sets.

Dataset	Aggregation	D31	R15	PID	Wine	Iris	Waveform	Seed
Record	788	3100	600	768	178	150	5000	210
Attributes	2	2	2	8	13	4	21	7
Clusters	7	31	15	2	3	3	3	3

Table 2: Comparison of the t-dpc and DPC methods for various data sets using the F assessment metric of the clustering outcomes.

Dataset	Aggregation	D31	R15	PID	Wine	Iris	Waveform	Seed
DPC	0.997	0.969	0.996	0.515	0.672	0.786	0.357	0.317
T-DPC	0.999	0.971	0.998	0.658	0.729	0.899	0.418	0.302

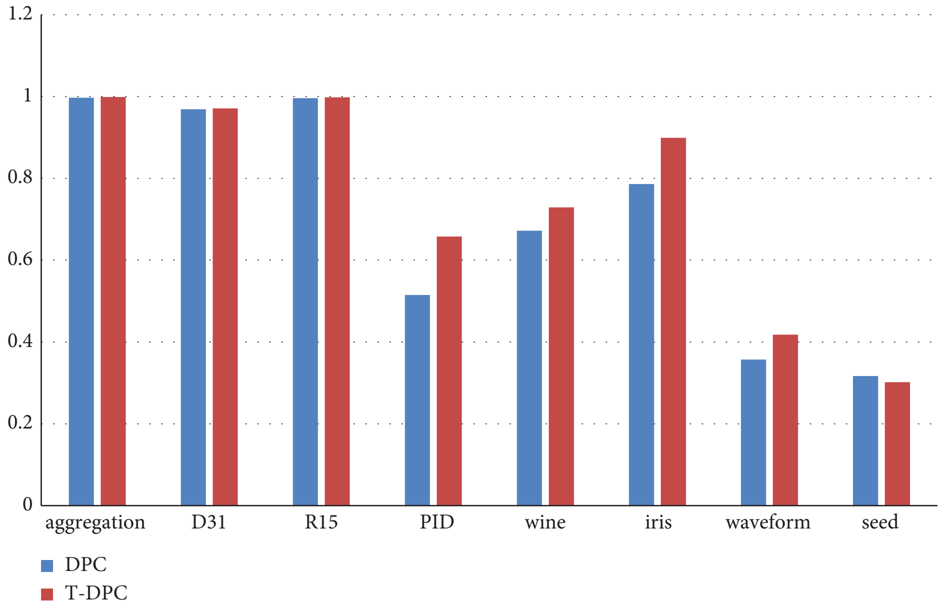

Figure 1: Comparison of the F assessment metric of clustering outcomes using different data sets [the higher values are better than the lower values].

Table 3: Comparison of the suggested t-dpc and the classical DPC methods using the NMI evaluation indicators.

Dataset	Aggregation	D31	R15	PID	Wine	Iris	Waveform	Seed
DPC	0.995	0.934	0.993	0.427	0.437	0.723	0.391	0.333
T-DPC	0.998	0.938	0.997	0.380	0.399	0.799	0.358	0.316

in Table 4. The new Gaussian kernel function has significantly reduced the whole running time. The t-DPC method runs approximately 116.1 s, 5 s, and 3 s faster than the classical DPC technique in the simulated three different data sets i.e., d31, aggregation, and R15, respectively. Furthermore, we observed that the standard set waveform and the PID were both improved at the same time, as are 334.5 s and 3 s, correspondingly. The wine dataset and the iris dataset, however, show no change for the reason of the little amount of data points in the data set. According to the aforementioned experimental findings and our deep analysis, the t-dpc technique is more suited for enormous data sets (several dimensions) than the small and only large data sets (have few dimensions). This major reason for this claim is that it can upsurge effectiveness and therefore, better suited for data from the actual world.

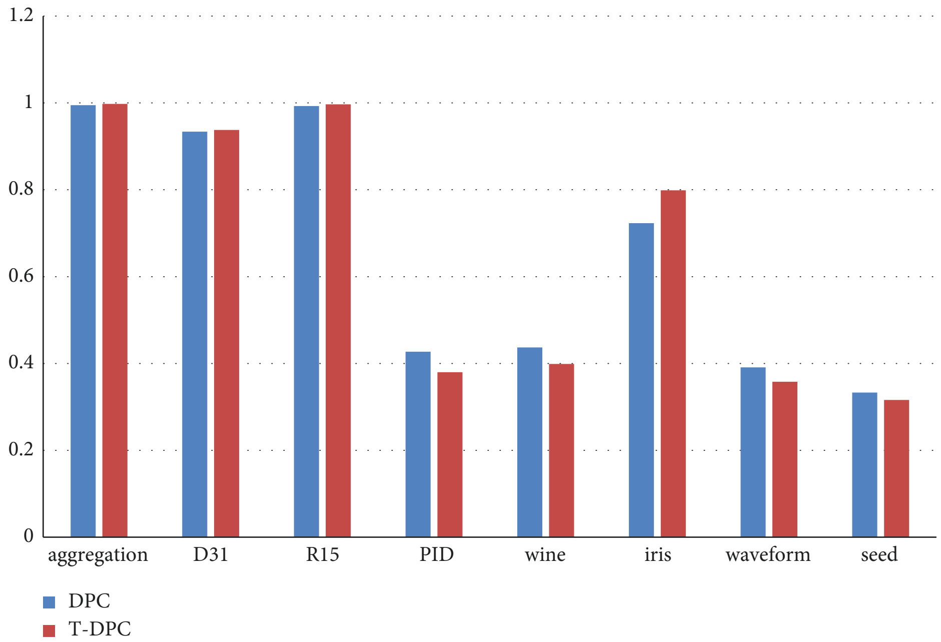

Figure 2: Comparison of the suggested t-dpc and the classical DPC methods using the NMI evaluation indicators.

Table 4: Comparison of running time of the two algorithms.

Dataset	Aggregation	D31	R15	PID	Wine	Iris	Waveform	Seed
DPC	22.2	310.2	14.2	23.5	3.5	3.1	846.7	3.7
T-DPC	17.2	194.1	11.2	20.5	3.5	3.1	512.2	3.5

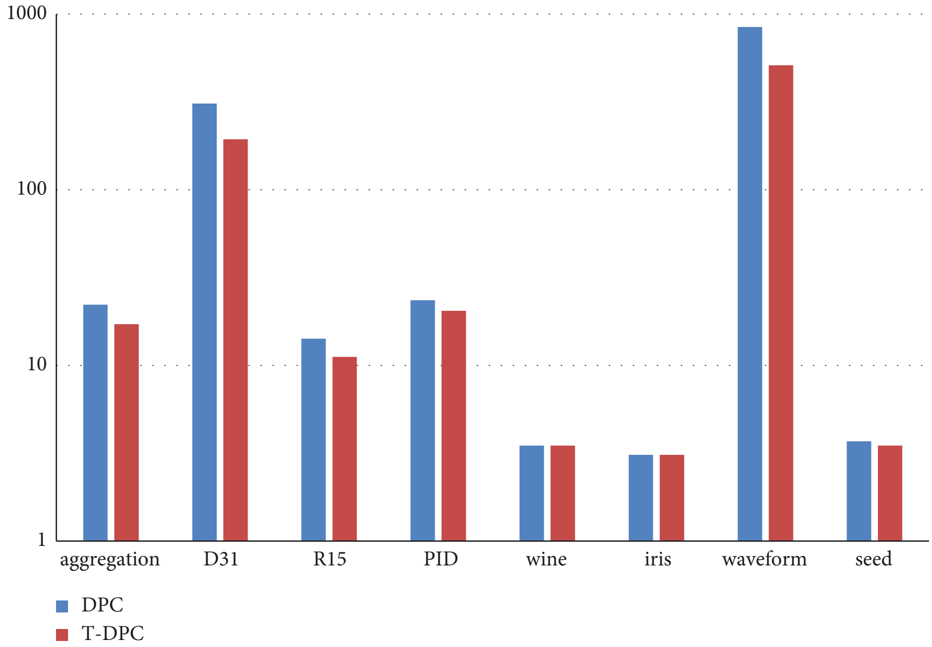

Figure 3: Comparison of running time of the two algorithms over different data sets [the lower values are better than the higher values].

4. Conclusions and Future Research

The method of clustering involves grouping the data set into various clusters based on how similar the data samples are to one another. The data objects belonging to various clusters must be as dissimilar from each other as feasible, while the data objects belonging to the same cluster must be as similar as possible. Compared with the classification algorithm, clustering analysis does not need to understand the classification attributes of data in advance. It is unsupervised. These data objects are divided completely according to the internal relationship between data objects, which is more in line with the development characteristics of the information age. This study analyses the time efficiency of the t-dpc algorithm and the DPC algorithm before using two efficient indexes to evaluate, namely the NMI index, and the F-measure index, to confirm the validity of the clustering effect of the t-dpc technique. Finally, this paper evaluates and analyses them as a whole according to the experimental results. This paper found that the effect of the t-dpc method and the DPC approach on low-dimensional data has no obvious change, nonetheless it has a relatively good enhancement, in particular, on very high-dimensional data sets. Then, the density calculation formula is unified and integrated into the t-dpc to improve the calculation effectiveness under enormous data sets.

In this paper, we have done some research on the optimization of DPC algorithm, and achieved some results, but there are still many deficiencies, which need to be further improved and improved. Although, the t-SNE algorithm and DPC algorithm have been preliminarily combined, the applicability of the combined algorithm needs to be improved. How to effectively combine the advantages of the algorithm and improving the applicability of the binding algorithm becomes a focus on the following research. Because, the DPC algorithm research involves the distance matrix calculation, therefore the improved algorithm is not out of this category and big data processing complexity and real-time implementation is of great challenge. Furthermore, this article is a kind of simple and shallow application, therefore how the algorithm can be applied to real life scenarios, to help people improve their quality of life, also will be the important direction of our future research.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] R. Lu, H. Zhu, X. Liu, J. K. Liu, and J. Shao, “Toward efficient and privacy-preserving computing in big data era,” *IEEE Network*, vol. 28, no. 4, pp. 46–50, 2014.

[2] C. Zhen and C. Jiang, “Overview of data mining in the era of big data,” *International Core Journal of Engineering*, vol. 5, no. 10, pp. 136–139, 2019.

[3] H. Chen, R. H. L. Chiang, and V. C. Storey, “Business intelligence and analytics: from big data to big impact,” *MIS Quarterly*, vol. 36, no. 4, pp. 1165–1188, 2012.

[4] F. Wang, G. Wang, and Z. Li, “A grid based density peak clustering algorithm,” *Small microcomputer system*, vol. 38, no. 5, pp. 1034–1038, 2017.

[5] B. Everitt, “Cluster analysis,” *Quality and Quantity*, vol. 14, no. 1, pp. 75–100, 1980.

[6] H. Wang, Z. Liu, and T. fan, “Consumer reporting behavior of online takeout food safety and its influencing factors -- an Empirical Analysis Based on 1018 consumer survey data in Shanghai,” *Food industry*, vol. 42, no. 2, pp. 236–240, 2021.

[7] P. Sonar, “Data density correlation degree clustering algorithm for multiple correlated sensor networks using fuzzy logic,” *International Journal on Recent and Innovation Trends in Computing and Communication*, vol. 3, no. 4, pp. 2208–2212, 2015.

[8] M. Ghiassi, H. Saidane, and R. Oswal, “YAC2: an α-proximity based clustering algorithm,” *Expert Systems with Applications*, vol. 167, no. 2, Article ID 114138, 2021.

[9] L. Morales and J. Aguilar, “An automatic merge technique to improve the clustering quality performed by LAMDA,” *IEEE Access*, vol. 8, Article ID 162944, 2020.

[10] A. Likas, N. Vlassis, and J. J Verbeek, “The global k-means clustering algorithm,” *Pattern Recognition*, vol. 36, no. 2, pp. 451–461, 2003.

[11] P. Viswanath and V. Suresh Babu, “Rough-DBSCAN: a fast hybrid density based clustering method for large data sets,” *Pattern Recognition Letters*, vol. 30, no. 16, pp. 1477–1488, 2009.

[12] J. C. Bezdek, R. Ehrlich, and W. F. C. M. Full, “FCM: the fuzzy c-means clustering algorithm,” *Computers & Geosciences*, vol. 10, no. 2–3, pp. 191–203, 1984.

[13] A. Kazumasa, A. Jun, H. Masafumi et al., “GSMaP passive microwave precipitation retrieval algorithm: algorithm description and validation,” *Journal of the Meteorological Society of Japan*, vol. 87, pp. 119–136, 2009.

[14] W. Zhong, G. Altun, R. Harrison, P. C. Tai, and Y. Pan, “Improved K-means clustering algorithm for exploring local protein sequence motifs representing common structural property,” *IEEE Transactions on NanoBioscience*, vol. 4, no. 3, pp. 255–265, 2005.

[15] A. Sharma, R. K. Gupta, and A. Tiwari, “Improved density based spatial clustering of applications of noise clustering algorithm for knowledge discovery in spatial data,” *Mathematical Problems in Engineering*, vol. 9, pp. 1–9, 2016.

[16] A. Rodriguez and A. Laio, “Clustering by fast search and find of density peaks,” *Science*, vol. 344, no. 6191, pp. 1492–1496, 2014.

[17] Y. Han, K. Li, F. Ge, Y. Wang, and W. Xu, “Online fault diagnosis for sucker rod pumping well by optimized density peak clustering,” *ISA Transactions*, vol. 120, no. 2022, pp. 222–234.

[18] Y. Zhao, R. N. Calheiros, G. Gange, J. Bailey, and R. O. Sinnott, “SLA-based profit optimization resource scheduling for big data analytics-as-a-service platforms in cloud computing environments,” *IEEE Transactions on Cloud Computing*, vol. 9, no. 3, pp. 1236–1253, 2021.

[19] S. Pourbahrami, L. M. Khanli, and S. Azimpour, “Improving neighborhood construction with Apollonius region algorithm based on density for clustering,” *Information Sciences*, vol. 522, pp. 227–240, 2020.

[20] D. Jiang, W. Zang, R. Sun, Z. Wang, and X. Liu, "Adaptive density peaks clustering based on K-nearest neighbor and Gini coefficient," *IEEE Access*, vol. 8, Article ID 113917, 2020.

[21] C. Kamath, "Intelligent sampling for surrogate modeling, hyperparameter optimization, and data analysis," *Machine Learning with Applications*, vol. 9, Article ID 100373, 2022.

[22] A. C. Gilbert, J. Y. Park, and M. B. Wakin, "Sketched SVD: recovering spectral features from compressive measurements," *Eprint Arxiv*, vol. 2, no. 11, pp. 1–10, 2012.

[23] J. A. Bullinaria and J. P. Levy, "Extracting semantic representations from word co-occurrence statistics: stop-lists, stemming, and SVD," *Behavior Research Methods*, vol. 44, no. 3, pp. 890–907, 2012.

[24] Y. Wu, X. Feng, Y. Dou, R. Zhu, L. Ma, and T Gao, "Density Peak clustering algorithm based on t-SNE Optimization," *Journal of Physics: Conference Series*, vol. 1237, no. 2, Article ID 022162, 2019.

[25] H. Wang, G. Li, and H. Yao, "Li Junzhao Stock network community division method based on influence calculation model," *Computer research and development*, vol. 10, pp. 2137–2147, 2014.