International Journal of Computational Intelligence Systems (2022) 15:61
https://doi.org/10.1007/s44196-022-00126-0

RESEARCH ARTICLE

Optimizing Tree-Based Contrast Subspace Mining Using Genetic Algorithm

Florence Sia[1] · **Rayner Alfred**[1]

Received: 26 August 2021 / Accepted: 27 July 2022

Abstract

Mining contrast subspace is a task of finding contrast subspace where a given query object is most similar to a target class but dissimilar to non-target class in a multidimensional data set. Recently, tree-based contrast subspace mining method has been introduced to find contrast subspace in numerical data set effectively. However, the contrast subspace search of the tree-based method may be trapped in local optima within the search space. This paper proposes a tree-based method which incorporates genetic algorithm to optimize the contrast subspace search by identifying global optima contrast subspace. The experiment results showed that the proposed method performed well on several cases compared to the variation of the tree-based method.

Keywords Mining contrast subspace · Contrast subspace · Genetic algorithm · Optimization

Abbreviations

TB-CSMiner	Tree-based contrast subspace miner
CSMiner	Contrast subspace miner
BPR	Bounding pruning refining
OPS	Optimized parameter setting
Freq	Frequency
Chrom	Chromosome
FFScore	Fitness function score
CS	Contrast subspace
BCW	Breast cancer Wisconsin
PID	Pima Indian diabetes
Wave	Waveform
CMSC	Climate model simulation crushes
UCI	University of California, Irvine
NB	Naïve bayes
SVM	Support vector machine
RF	Random forest
WEKA	Waikato environment for knowledge analysis

1 Introduction

Given a multidimensional data set comprised of target and non-target classes, mining contrast subspace finds contrast subspace of a query object. A contrast subspace of a query object is a subspace or subset of features in which the query object is most similar to target class but dissimilar to non-target class. Query object can be any object in which its contrast subspace is essential to be investigated. The identified contrast subspace is crucial in giving insight into the query object with regards to the target class and non-target class. Mining contrast subspace has many important applications in the field such as disease diagnosis or fraud detection. For example, in disease diagnosis, a medical doctor may want to know the symptoms that make the patient most likely belong to a target class of disease against other class of disease. Those identified symptoms can help the medical doctor in making accurate disease diagnosis and then provide appropriate treatment to the patient. Similarly, in credit card fraud detection, an analyst may want to know the features that cause a credit card transaction more similar to the fraud cases than the normal cases. Those features can provide information about the case for further investigation.

Tree-based contrast subspace mining method has been introduced to identify contrast subspace of query object in two-class multidimensional numerical data set [1, 2]. The tree-based contrast subspace method used tree-based likelihood contrast scoring function to estimate the likelihood

✉ Florence Sia
florence.sfs@ums.edu.my

Rayner Alfred
ralfred@ums.edu.my

[1] Knowledge Technology Research Unit, Faculty of Computing and Informatics, University Malaysia Sabah, Jalan UMS, 88400 Kota Kinabalu, Sabah, Malaysia

Published online: 10 August 2022

contrast score of subspaces with respect to a given query object. That is the degree to which the query object is more likely similar to a target class against non-target class in a subspace. The tree-based method finds a subset of relevant features with high likelihood contrast score and searches for highly scored contrast subspaces from the relevant features. The tree-based likelihood contrast score estimation of a subspace involves partitioning the subspace space into two group of data objects recursively on which the target objects and non-target objects are well separated with respect to the query object until the group contains only a single class or meets the minimum number of objects threshold. Accordingly, the tree-based likelihood contrast score of a subspace is the ratio of probability of target objects to probability of non-target objects in the group that containing query object. Recently, a genetic algorithm-based method has been proposed to optimize the parameter setting of the tree-based method which further improves the accuracy of the method. However, the genetic algorithm has not been used to optimize the contrast subspace search of the tree-based method. The tree-based contrast subspace mining method searches contrast subspaces of query object from a fixed small set of relevant features. This may cause the contrast subspace search more likely to be trapped in a local optima within the search space. Hence, it may deteriorate the accuracy performance of the method in identifying the contrast subspace of query object. Genetic algorithm has been widely applied in various optimization research works to find the most optimal solution to problem [3–7]. In this paper, we propose a genetic tree-based method which incorporates genetic algorithm to optimize the contrast subspace search of the method. That is a population of candidate potential subsets of relevant features will undergo a series of evolvement in which the tree-based likelihood contrast score of subspaces obtained from the subsets of features are maximized. Accordingly, the subspaces search can be performed on wide relevant feature space to find global optima contrast subspace.

The organization of this paper is as follows: The second section presents the literature review. Third section describes the framework of the genetic tree-based contrast subspace mining method. This is followed by a section that is presenting the experimental design and analysis for evaluating the effectiveness of the genetic tree-based method in finding relevant contrast subspaces of query object. The last section concludes this paper with the conclusion and future works.

2 Related Works

To the best of our knowledge, there are only few mining contrast subspace methods that have been proposed in the literature.

CSMiner (Contrast Subspace Miner) which employed the density-based likelihood contrast scoring function has been proposed to identify contrast subspace of a query object in numerical data set [8]. The density-based likelihood contrast scoring function estimates the likelihood contrast score of a subspace with respect to a query object based on the ratio of probability density of target objects to probability density of non-target objects. Contrast subspace of a query object should have high density-likelihood contrast score. CSMiner searches subspaces set in depth-first search manner and prunes subspaces from the search space based on the upper bound of probability density of target objects. However, it is inefficient for large search space that can be generated from high dimensionality (i.e., number of features) of data.

CSMiner-BPR (i.e., Contrast Subspace Miner-Bounding Pruning Refining) has been proposed to address the efficiency issue of the CSMiner [9]. It searches subspace space and prunes subspaces based on the upper bound of probability density of target objects and the lower bound of probability density of non-target objects within their neighborhood. This accelerates the mining contrast subspace process through saving the computation time for those objects outside of the neighborhood. Nevertheless, the density-based likelihood contrast scoring function involves pairwise distance measure causes the score tends to decrease when the dimensionality of subspace increases. It requires an adjustment to the dimensionality of subspaces which may affect the performance of mining contrast subspace.

TB-CSMiner (Tree-Based Contrast Subspace Miner) method has been introduced which employs the tree-based likelihood contrast scoring function. It uses the concept of divide-and-conquer of decision tree method which is not affected by the dimensionality of subspace [1]. For a subspace, the tree-based likelihood contrast scoring function attempts to gather query object with the target objects but separate it from the non-target objects in group. The ratio of target objects and non-target objects in group is then computed. TB-CSMiner avoids brute force search by searching subspaces from a space consisting only relevant features. High tree-based likelihood contrast score of subspace signifies subspace is the contrast subspace of query object. The effectiveness of TB-CSMiner is heavily dependent on its predefined parameters values. Hence, it is crucial to optimize the parameter setting to improve the performance of the method in identifying the accurate contrast subspace for query object.

TB-CSMiner with optimized parameter values has been proposed which uses genetic algorithm in the optimization process for a particular data set at hand [2]. It generates an initial population of different sets of parameter's values. The fitness of each set of parameters values is then assessed based on the accuracy performance of TB-CSMiner using the parameters values to find contrast subspaces of the given query object. A subset of sets of parameters values having high accuracy are selected to be reproduced via crossover and mutation operations to generate a new population iteratively. At the end, the highly accurate set of parameters values is returned as the best parameter setting for the TB-CSMiner method. This work is different from our work in which the existing work focuses on optimizing only the parameter setting of the TB-CSMiner method using a genetic algorithm. Hence, the genetic algorithm is designed specifically to find the best parameter setting for TB-CSMiner. Another factor that might affect the effectiveness of the TB-CSMiner method is its subspace search strategy. TB-CSMiner searches for a potential contrast subspace from a fixed small set of relevant features. This causes the method more likely to return the local optima contrast subspace for the given query object.

3 Genetic Tree-Based Contrast Subspace Mining Method

Genetic algorithm is an evolutionary algorithm inspired by the Darwinian natural selection and a genetic computational model of biological process of evolution [3–7]. It is well known that genetic algorithm can find feasible global solution for various optimization problems. That is a genetic algorithm searches for the best possible solution from a pool of possible solutions by examining the solutions based on a fitness function. Multiple fitter solutions are kept and undergo evolution to generate new possible solutions over several generations. This will ensure the global optima solution can be found for a problem in an acceptable time. The application of the genetic algorithm in the tree-based contrast subspace mining method enables the examination of wider possible potential subspaces derived from the given full-dimensional data rather than a fixed small subset of features.

Hence, the genetic tree-based contrast subspace mining method employs genetic algorithm to optimize the subspace search strategy to identify the global optima contrast subspace for the given query object in the two-class multidimensional numerical data. Figure 1 illustrates the framework of the genetic tree-based contrast subspace mining method.

Given a two-class multidimensional numerical data set, a target class, a query object, the genetic tree-based mining contrast subspace process begins by designing the chromosomes represent different subsets of l features. After that, an initial population of chromosomes is generated. The fitness evaluation is performed on each chromosome in the population based on the tree-based likelihood contrast scoring function. Based on the fitness score of the chromosomes, several chromosomes in the population are selected into a new population by using the roulette wheel selection method. Then, chromosomes in the new population are reproduced first via crossover operation and followed by mutation operation to generate new chromosomes. A series of fitness evaluation, selection, crossover, and mutation process will be performed until the maximum number of iterations μ is met. Lastly, h subspaces having high tree-based likelihood contrast score are identified as the most relevant contrast subspaces of the query object. The following subsections describe the main stages involved in greater details.

3.1 Chromosome Representation

The representation of chromosomes is designed to correspond to different subsets of features from the full feature set in the data set. A chromosome consists of genes in which the value of each gene is the index position of a feature in the full feature set. For example, the chromosome representation of the subset of features $\{f_2,f_3,f_4,f_5\}$ for full feature set $\{f_1,f_2,f_3,f_4,f_5\}$ is illustrated in Fig. 2.

3.2 Initial Population

An initial population consists of p random chromosomes is generated. Each random chromosome represents a subset of features picked randomly from a collection of possible subsets of l features that can be derived from the full feature set in data set, where l is less than the dimensionality of full feature set.

3.3 Fitness Evaluation

At this stage, the fitness of each chromosome is evaluated by assessing the contrast subspace obtained from the underlying subset of features based on the tree-based likelihood contrast scoring function. Herein, the tree-based likelihood contrast scores of t random subspaces which are searched from the subset of features with respect to query object are estimated. t is the number of random subspaces with $t > 1$. Highly scored random subspace is then taken as the contrast subspace attained from the subset of features.

Specifically, the estimation of the tree-based likelihood contrast score of a subspace by using the tree-based likelihood contrast scoring function is as follow: Given a two-class d-dimensional numerical data set O comprised of target objects O_+ belong to target class C_+ and non-target objects $O\_$ belong to non-target class $C\_$, and a query object q, for a subset of features S_{ub}, the tree-based likelihood contrast scoring function constructs a half binary tree from the S_{ub} space. The tree construction starts by selecting a random feature f from S_{ub} and the f value of a which has the highest information gain score is used as the splitting criterion such as $f \leq a$ and $f > a$. Then, the splitting criterion is used to split the data objects into left node that containing a subset of

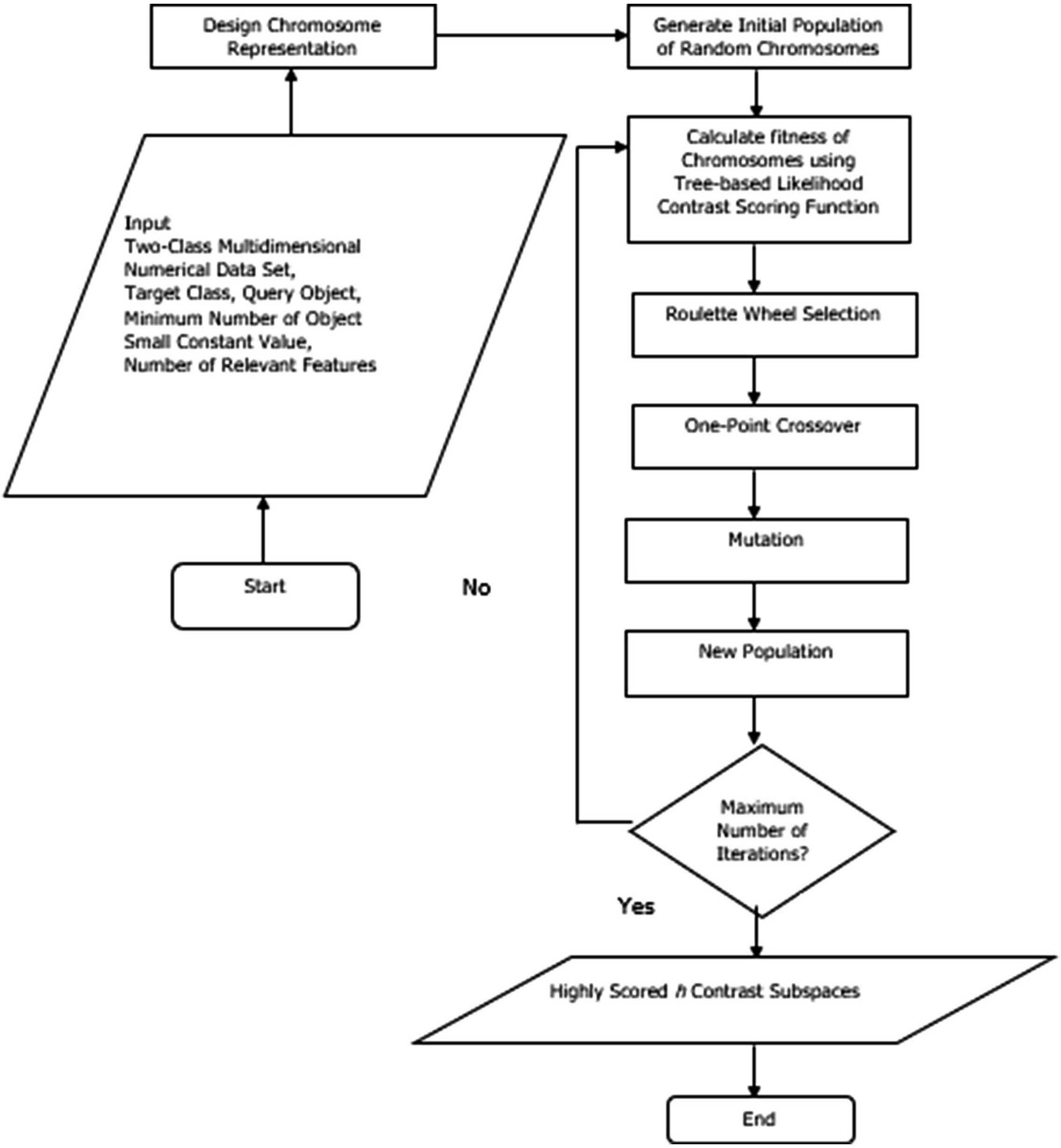
Design Chromosome Representation
Generate Initial Population of Random Chromosomes
Input
Two-Class Multidimensional Numerical Data Set,
Target Class, Query Object,
Minimum Number of Object
Small Constant Value,
Number of Relevant Features
Calculate fitness of Chromosomes using Tree-based Likelihood Contrast Scoring Function
Roulette Wheel Selection
One-Point Crossover
Mutation
Start
No
New Population
Maximum Number of Iterations?
Yes
Highly Scored *h* Contrast Subspaces
End

Fig. 1 Framework of genetic tree-based contrast subspace mining method

2	3	4	5

Fig. 2 Chromosome representation

objects with *f* has value at most *a*, and right node having a subset of objects with *f* has value greater than *a*. This process is performed recursively until the node contains only either target objects or non-target objects or meets the minimum number of objects threshold *MinObjs*. The nodes at the bottom of the tree are known as leaf node X_{leaf}. Lastly, those features involved in tree construction constitute a subspace *S* derived from the S_{ub}. The tree-based likelihood contrast score of *S* will be measured using Eq. (1).

$$Tree - LC_S(q) = \frac{freq(C_+, X_{\text{leaf}})/|O_+|,}{N} \tag{1}$$

with

$$N = \begin{cases} freq(C_-, X_{\text{leaf}})/O_-),\ freq(C_-, X_{\text{leaf}})) > 0 \\ \varepsilon \qquad\qquad\qquad freq(C_-, X_{\text{leaf}})) = 0 \end{cases} \quad (2)$$

where $freq(C_+, X_{leaf})$ is the number of target objects in the leaf node, $\left|O_+\right|$ denotes the number of target objects in the data set, $freq(C_-, X_{leaf})$ denotes the number of non-target objects in the leaf node, $\left|O_-\right|$ is the number of non-target objects in the data set and ε is a small constant value. A high tree-based likelihood contrast score of subspace indicates that query object is more similar to the target class against non-target class in the subspace. The highly scored random subspace is then taken as the best contrast subspace for query object that can be identified from the chromosome.

3.4 Selection

During selection stage, a subset of chromosomes is selected from current population using the roulette wheel selection method. Those chromosomes will be reproduced through the crossover and mutation operations to form a new population. The roulette wheel selection method first estimates the selection probability of each chromosome that is the proportion of a chromosome's fitness to the total fitness scores and subsequently the cumulative probability u_i after including each ith chromosome [10]. After that, a random integer r is picked within the range 0 and 1. The ith chromosome is only selected if $\mathrm{u_{i\text{-}1}} < \mathrm{r} \leq \mathrm{u_i}$. This selection process continues until the new population consists of p chromosome.

3.5 Crossover

The commonly used one-point crossover operation with a probability of crossover pc is performed on the chromosomes in the new population to produce new chromosomes [11]. One-point crossover begins with choosing two parent chromosomes randomly from the newly generated population and followed by a random integer r within the range 0 and 1. It chooses randomly a crossover point from 1 to total genes − 1 if $r < pc$. The fragments of the parent chromosomes after the crossover point are interchanged to produce two new chromosomes. These chromosomes replace the parent chromosomes in the new population. The crossover operation is performed iteratively for the remaining parent chromosomes in the new population. An example of one-point crossover operation on two parent chromosomes with a crossover point 3 is illustrated in Fig. 3. The first parent chromosome representing a subset of features $\{f_1, f_2, f_3, f_4, f_5\}$ and the second parent chromosome representing a subset of features $\{f_1, f_2, f_6, f_7, f_8\}$. After the crossover operation is performed at crossover point 3, the fragments of parent chromosome 1 and 2 after the crossover point are exchanged. This creates offspring 1 that holds subset of features $\{f_1, f_2, f_3, f_7, f_8\}$ and off spring 2 that carries subset of features $\{f_1, f_2, f_6, f_4, f_5\}$.

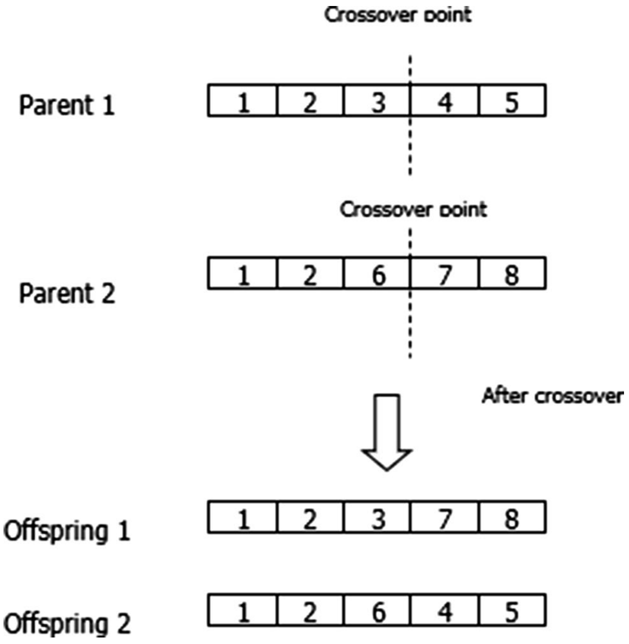

Fig. 3 One-point crossover operation

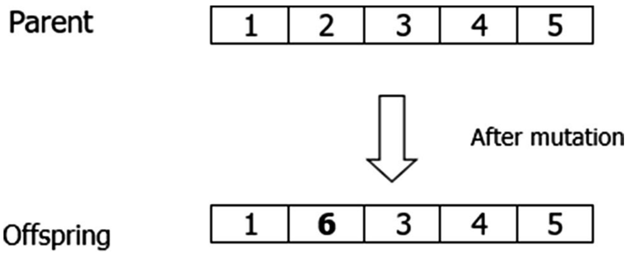

Fig. 4 Mutation operation

3.6 Mutation

At this stage, the mutation operation with a probability of mutation pm is performed on the chromosomes in the new population [12]. The mutation operation starts with the first gene of a parent chromosome and then chooses a random integer r within the range 0 and 1. It mutates the gene by changing its value to other index position of a random feature if $r < pm$. These processes are repeated for the rest of genes of the chromosome. The parent chromosome with mutated gene will be a new chromosome that represents a new subset of features. This mutation operation is performed repeatedly for all remaining parent chromosomes in the new population. Figure 4 shows an example of mutation operation on a parent chromosome representing subset of features $\{f_1, f_2, f_3, f_4, f_5\}$. After the mutation operation, the second gene of the parent chromosome is mutated which it changes the value of the gene from 2 to 6. This produces an offspring that carries a new subset of features $\{f_1, f_6, f_3, f_4, f_5\}$.

4 Experimental Setup and Analysis

An experiment is carried out to evaluate the performance accuracy of the genetic tree-based contrast subspace mining method by comparing to the TB-CSMiner method (i.e., without genetic algorithm) and TB-CSMiner method with optimized parameter setting in finding contrast subspaces of query object. This experiment is conducted on six real-world multidimensional numerical data sets from UCI machine learning repository namely the Breast Cancer Wisconsin (BCW), the Wine, the Pima Indian Diabetes (PID), the Glass Identification (Glass), the Climate Model Simulation Crushes (CMSC), and the Waveform (Wave) data sets [13]. Table 1 tabulates the details of the data sets. Since there is no ground truth contrast subspace provided in the real-world two-class multidimensional numerical data set, the accuracy of the method is assessed based on the classification accuracy on the contrast subspace projected data set as suggested in [1, 2].

For the genetic tree-based method, this experiment uses the parameter setting which is found often able to perform well in optimization problem [14, 15]. The parameter setting of the genetic tree-based method is shown in Table 2.

In addition to that, it uses the best minimum number of objects *MinObjs*, small constant values ε, and several number of relevant of features l based on data sets which have been identified in the previous work as shown in Table 3 [2]. However, a smaller number of random subspaces t is used that is 10 to accelerate the mining contrast subspace process. The genetic tree-based method is implemented in Matlab 9.2 programming language and the classification accuracy evaluation is implemented in Java programming language.

The procedures of this experiment are as follows. For each data set, all objects are taken as query objects. The class of the query object is assigned as the target class. For a query object and a target class, the genetic tree-based method is run on the data set. Herein, only one contrast subspace with the highest tree-based likelihood contrast score is considered. This process is performed repeatedly for the remaining query objects. After the contrast subspace of all query objects have been identified, the classification accuracy of the contrast subspaces with respect to query object is assessed. For a contrast subspace of a query object, the data set is first projected onto the contrast subspace with respect to the query object. Then, the contrast subspace space is fed into several classifiers that include J48 (decision tree), NB (naive bayes), SVM (support vector machine), and RF (random forest), in WEKA to perform classification based on 20-fold cross validation. Lastly, the classification accuracies on contrast subspace for all query objects are averaged for each of the classifiers.

Table 2 Parameter setting of genetic tree-based method

Parameter	Value
p	200
pc	0.6
pm	0.01
μ	20

Meanwhile, the default parameter setting as suggested in the previous works is used for both tree-based and tree-based with optimized parameter setting [1, 2]. Table 4 presents the average percentage of classification accuracy on BCW, PID, Wine, Glass, CMSC, and Wave data sets for classifier J48, NB, SVM, and RF.

Based on the results, the genetic tree-based method identified contrast subspaces that attained higher classification accuracy compared to the tree-based method with OPS for NB and SVM on BCW data set. The respective classification accuracy is 99.44% and 96.59%. The genetic tree-based method identified contrast subspaces with higher classification accuracy, 96.05% for SVM on Wine data set. While the genetic tree-based method produced contrast subspaces having higher classification accuracy than the tree-based method with OPS for J48 and RF on Glass data set. It obtained 85.18% and 87.33% for J48 and RF respectively. Besides, it gained contrast subspaces that achieved higher accuracy that is 97.77% for J48 on Wave data set.

Overall, the genetic tree-based method demonstrated good results on only few cases. This is mainly due to the parameter setting of the genetic algorithm that includes the size of population, the probability of crossover, and the probability of mutation are not optimized for mining contrast

Table 1 Details of data sets

Data set	No. of objects	No. of features
BCW	699	9
PID	768	8
Wine	178	13
Glass	214	9
CMSC	540	18
Wave	5000	21

Table 3 The *Minobjs*, ε, and l values

Data set	MinObjs (%)	ε	l
BCW	8	0.001	2
PID	10	0.001	2
Wine	25	0.001	5
Glass	4	0.01	3
CMSC	2	0.01	6
Wave	16	0.01	2

Table 4 Average classification accuracy (%)

Data set	Tree-Based method				Tree-based method with OPS				Genetic tree-based method			
	J48	NB	SVM	RF	J48	NB	SVM	RF	J48	NB	SVM	RF
BCW	98.53	99.39	97.86	99.00	98.96	99.35	95.80	99.25	98.17	99.44	96.59	98.94
PID	94.24	96.23	84.34	94.91	94.83	96.40	82.64	95.30	94.57	95.89	82.61	94.69
Wine	96.86	98.22	94.17	97.47	96.09	98.37	95.94	97.60	95.11	97.77	96.05	96.84
Glass	84.16	83.07	65.94	86.14	84.84	82.79	66.75	86.53	85.18	81.37	66.62	87.33
CMSC	94.12	97.32	94.27	95.45	93.68	97.76	94.71	95.64	92.74	97.27	93.88	94.77
Wave	92.87	96.74	88.60	94.91	95.37	99.15	90.90	96.91	97.77	97.58	86.13	95.80

subspace problem. Besides, the results showed that the tree-based method with OPS outperformed the tree-based method for most of the cases. This is because the parameter setting of the tree-based method is optimized to reach satisfactory performance for each data set. However, it is worth noting that the genetic tree-based method is capable to outperform both the tree-based method and the tree-based method with OPS. That is the genetic tree-based method achieved higher classification accuracy, 99.44% for NB on BCW data set, 96.05% for SVM on Wine data set, and 97.77% for J48 on Wave data set. According to the paired-sample T-test at the significance level of 0.05, most of those performance accuracy improvement cases are significant. This exhibits that the genetic algorithm can be applied to optimize the contrast subspace search of the tree-based method. It is well known that there is no one universal method suits for solving all types of problem. Different settings of case commonly require different methods or approaches [16, 17].

5 Conclusion

The proposed genetic tree-based contrast subspace mining method employs genetic algorithm to optimize the process of searching contrast subspaces of the given query object in two-class multidimensional numerical data set. For a query object, a sequence of different populations of subspaces has been generated from an initial population of random subspaces. Over the generation, the tree-based likelihood contrast scores of subspaces in a population with respect to the query object are assessed. Highly scored subspaces as potential contrast subspaces of query object are passed on from one population to the subsequence population. This will preserve the current best identified subspaces and thus ensure the optimal contrast subspaces for the query object can be attained. At the end, the highly scored subspaces are taken as the best contrast subspaces of the given query object. The empirical studies showed that the genetic tree-based method performed well on some cases compared to both benchmarked tree-based methods in finding contrast subspaces of query objects on multidimensional numerical data sets. The parameter setting of the genetic algorithm may affect the effectiveness of the genetic tree-based method. Nevertheless, that parameter setting is not optimized for identifying contrast subspace of query object. Future work would aim to optimize the parameter setting of the genetic algorithm to further improve the performance of the genetic tree-based contrast subspace mining method.

Author Contributions FS and RA have contributed to the conception, design, analysis, and writing of this manuscript. All authors read and approved the final manuscript.

Funding No funding was received for conducting this study.

Availability of Data and Materials The datasets analysed during the current study are available from the corresponding author on reasonable request.

Declarations

Conflict of Interest The authors have no conflicts of interest to declare that are relevant to the content of this article.

Ethics approval Not applicable.

Consent to Participate Not applicable.

Consent for Publication Publisher has the author's permission to publish the content of this article.

References

1. Sia, F., Alfred, R.: CSMiner-TBM: tree-based mining contrast subspace. Int J Adv Intell Inform (2019). https://doi.org/10.26555/ijain.v5i2.359
2. Sia F, Alfred R. Optimizing parameters values of tree-based contrast subspace miner using genetic algorithm. Computational Science and Technology (Springer, Singapore), pp. 677–687, 2020
3. Reddy, G.T., Lakshmanna, M.P.K.K., Rajput, D.S., Kaluri, R., Srivastava, G.: Hybrid genetic algorithm and a fuzzy logic classifier for heart disease diagnosis. Evolut Intell **13**(2), 185–196 (2020)
4. Das, A.K., Sengupta, S., Bhattacharyya, S.: A group incremental feature selection for classification using rough set theory based genetic algorithm. Appl Soft Comput **65**, 400–411 (2020)
5. Dong, H., Li, T., Ding, R., Sun, J.: A novel hybrid genetic algorithm with granular information for feature selection and optimization. Appl Soft Comput **65**, 33–46 (2018)
6. Neysiani, B.S., Soltani, N., Mofidi, R., Nadimi-Shahraki, M.H.: Improve performance of association rule-based collaborative filtering recommendation systems using genetic algorithm. Int J Inf Technol Comput Sci **11**(2), 48–55 (2019)
7. Das, A.K., Das, S., Ghosh, A.: Ensemble feature selection using bi-objective genetic algorithm. Knowl Based Syst **123**, 116–127 (2017)
8. Duan L, Tang G, Bailey J, Dong G, Campbell A, Tang C. Mining contrast subspaces. In: Pacific-Asia Conference on Knowledge Discovery and Data Mining. Springer, Cham, pp 249–260. 2014
9. Duan, L., Tang, G., Pei, J., Bailey, J., Dong, G., Nguyen, V., Tang, C.: Efficient discovery of contrast subspaces for object explanation and characterization. Knowl Inf Syst **47**(1), 99–129 (2016)
10. Shi Z, Li Q, Zhang S, Huang X. Improved crow search algorithm with inertia weight factor and roulette wheel selection scheme. In: International Symposium on Computational Intelligence and Design. 2017. pp. 205–209
11. Magalhaes-Mendes, J.: A comparative study of crossover operators for genetic algorithms to solve the job shop scheduling problem. WSEAS Trans Comput **12**(4), 164–173 (2013)
12. Shukla, A.K., Singh, P., Vardhan, M.A.: A new hybrid feature subset selection framework based on binary genetic algorithm and information theory. Int J Comput Intell Appl **18**(03), 1950020 (2019)
13. Blake C. UCI repository of machine learning databases. 1998. http://www.ics.uci.edu/~mlearn/MLRepository.html
14. Chiroma H, Abdulkareem S, Abubakar A, Zeki, Zeki A, Gital AYU, Usman MJ. Correlation study of genetic algorithm operators: crossover and mutation probabilities. In: Proceedings of the International Symposium on Mathematical Sciences and Computing Research. 2013. pp. 6–7.
15. Liu, X.Y., Liang, Y., Wang, S., Yang, Z.Y., Ye, H.S.: A hybrid genetic algorithm with wrapper-embedded approaches for feature selection. IEEE Access **6**, 22863–22874 (2018)
16. Jadhav, S., He, H., Jenkins, K.: Information gain directed genetic algorithm wrapper feature selection for credit rating. Appl Soft Comput **69**, 541–553 (2018)
17. Katoch, S., Chauhan, S.S., Kumar, V.: A review on genetic algorithm: past, present, and future. Multimed Tools Appl **80**(5), 8091–8812 (2021)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.