Soft Comput
DOI 10.1007/s00500-016-2257-0

دانلود کننده مقالات علمی
freepaper.me
FREE paper

A new multi-colony fairness algorithm for feature selection

Xiang Feng[1] · **Tan Yang**[1] · **Huiqun Yu**[1]

© Springer-Verlag Berlin Heidelberg 2016

Abstract As the world gradually transforms from an information world to a data-driven world, areas of pattern recognition and data mining are facing more and more challenges. The process of feature subset selection becomes a necessary part of big data pattern recognition due to the data with explosive growth. Inspired by the behavior of grabbing resources in animals, this paper adds personal grabbing-resource behavior into the model of resource allocation transformed from the model of feature selection. Multi-colony fairness algorithm (MCFA) is proposed to deal with grabbing-resource behaviors in order to obtain a better distribution scheme (i.e., to obtain a better feature subset). The algorithm effectively fuses strategies of the random search and the heuristic search. In addition, it combines methods of filter and wrapper so as to reduce the amount of calculation while improving classification accuracies. The convergence and the effectiveness of the proposed algorithm are verified both from mathematical and experimental aspects. MCFA is compared with other four classic feature selection algorithms such as sequential forward selection, sequential backward selection, sequential floating forward selection, and sequential floating backward selection and three mainstream feature selection algorithms such as relevance–redundancy feature selection, minimal redundancy–maximal relevance, and ReliefF. The comparison results show that the proposed algorithm can obtain better feature subsets both in the aspects of feature subset length which is defined as the number of features in a feature subset and the classification accuracy. The two aspects indicate the efficiency and the effectiveness of the proposed algorithm.

Keywords Feature selection · Multi-colony fairness algorithm (MCFA) · Resource allocation · Grabbing-resource behavior

1 Introduction

1.1 Background and significance

The big data is getting more and more attention with the coming of the cloud computing era. And as the world is transforming from informatization to datamation, the increase in data volumes and the varieties of data structures make the areas of pattern recognition and data mining face more and more challenges (Xiaofeng et al. 2013). Feature selection can effectively reduce the redundancy of data through selecting more useful data from the explosive amount of data. And this function makes it become the necessary part in the process of pattern recognition.

1.2 Related works

Search strategies and evaluation criteria are two main research points of feature subset selection. Feature selection methods can be divided into the complete search, the heuristic search, and the random search according to search strategies. And they also can be divided into filter methods, wrapper methods, and embedded methods on the basis of the relation between evaluation criteria and classifiers (Guyon 2003). Filter methods determine the significance of

Communicated by V. Loia.

✉ Xiang Feng
xfeng@ecust.edu.cn

✉ Tan Yang
yt@mail.ecust.edu.cn

[1] Department of Computer Science and Engineering,
East China University of Science and Technology,
Meilong Road 130, Shanghai 200237,
People's Republic of China

Published online: 06 July 2016

features based on the contributions made by features to classification accuracies. Then, they choose an optimal feature subset which includes the most important features. Wrapper methods (Glten 2013; Nemati and Basiri 2011; Han et al. 2014) depend on the learning process and evaluate features on the basis of classification performance of feature subsets. Embedded methods complete the feature selection by optimizing an objective function in the process of training classifiers. As filter methods need not compute classification accuracies, the execution time of this kind of feature selection algorithms is the shortest among these three kinds of methods. In this paper, we mainly focus on the filter method for its predictable higher values in a more large-scale data environment. Filter methods certainly have their own disadvantage on low classification accuracies. We add some strategies of wrapper methods into filter methods to improve classification accuracies while not cutting the advantage of time complexity. Although heuristic search strategies are usually used in filter methods, they have obvious shortcomings. It depends on preexistent knowledge of single feature overly. To overcome this, we fuse strategies of the heuristic search and the random search. We not only consider the preexistent knowledge of single feature, but also take the information of feature combination into account.

The most popular filter feature selection methods are sequential forward selection (SFS) (Parkka et al. 2010; Bouatmane et al. 2011) and sequential backward selection (SBS) Juanying and Weixin (2014). Sequential forward selection is a bottom–up search, which starts with an empty set and adds new features one at a time. And sequential backward selection is a top–down approach, which starts with the whole set of features and deletes one feature at a time. On the basis of the two feature selection methods, the other two methods are proposed named as sequential floating forward selection (SFFS) (Mar et al. 2011; Xie et al. 2013) and sequential floating backward selection (SFBS) (Azar et al. 2014). The strategy "plus-l-take-away-r" is used in SFFS and SFBS, respectively, based on SFS and SBS (Jorge 2014). "plus-l-take-away-r" means adding to the feature subset l features and then removing the worst r features if $l > r$, or deleting r features and then adding l features if $r > l$. In the adding and removing operations, the parameters r and l are not fixed. Some researchers still devote themselves to improving the four methods described above. Uzer presented two hybrid feature selection methods which were composed by combining SFS and SBS together with the principal component analysis to utilize in the diagnosis of breast cancer fast and effectively (Uzer et al. 2013). Experimental results in his paper showed that the two methods were effective. John proposed a filter-dominating hybrid SFFS method which aimed at high efficiency and insignificant accuracy sacrifice for high-dimensional feature subset selection (John 2014). Their experimental results demonstrated the advantages and usefulness of the proposed method. Hongyi Peng also studied on SFFS (Peng et al. 2013). He modified SFFS based on weighted Mahalanobis distance to identify optimal informative gene subsets. Expect the four classic feature selection methods mentioned above, there are also some mainstream feature selection methods. Three of them are relevance–redundancy feature selection (RRFS), minimal redundancy–maximal relevance (mRMR), and ReliefF (Moradi and Rostami 2015). RRFS is an efficient feature selection method based on relevance and relevance–redundancy analysis which uses a specific criterion to choose an adequate number of features. mRMR is a solid multivariate filter approach which return a feature subset with features that are mutually far away from each other as well as highly correlated with the classification label. ReliefF is an extension of the Relief method which applies a feature weighting scheme and searches for several nearest neighbors. Although the four classic and the three mainstream methods are popular, there are still some defects. Firstly, the method's time complexity and the classification accuracy cannot achieve a good balance. Secondly, the length of the obtained feature subset is not controllable. To overcome these defects, we proposed a feature selection method based on multi-colony fairness model and compare it with the four classic and the three mainstream feature selection methods.

1.3 Motivation

The two existing defects of available feature selection methods propel me into a new research direction in the feature selection problem. If the length of selected feature subsets is controllable, the probability of being selected for all features is a kind of limited resources. Inspired by the idea of the fittest survived (Linksvayer 2014) and the behavior of competing for resources in the animal kingdom (Peter and Jessica 2008), we find that there is another kind of resource behavior which follows certain rules. Xiang Feng treated the source servers as intelligent individuals, and the source servers had their own lying behaviors (Feng et al. 2015). Not only the living individuals have behaviors, but individuals without life also can have behaviors. These motivate us to get inspirations from them:

1. The feature selection problem can not only be treated as a feature selection problem. It can also be transformed into a resource allocation model. The probability of being selected for each feature is a kind of resources to be allocated. In this situation, a discrete problem becomes a continuous problem.
2. Treat all the features as intelligent individuals. They all have the consciousness of grabbing resources. And in some certain conditions, they will adopt this behavior to optimize their own benefits.

دانلود کننده مقالات علمی
freepaper.me FREE paper

3. All the features constitute a colony. Our goal was to maximize this colony's benefits by evaluating and dealing with the grabbing-resource behavior adopted by each feature.

1.4 Contribution

The main contributions of this paper can be summarized as follows.

1. The feature selection model is transformed into a resource allocation model. To some extent, the discrete optimization problem is transformed into a continuous optimization problem effectively.
2. The grabbing-resource behavior is added into the process of resource allocation. And the proposed algorithm can obtain better resource allocation schemes through evaluating and dealing with grabbing-resource behaviors.
3. Strategies of the random search and the heuristic search are fused effectively. And the heuristic information is used in the process of the random search.
4. Methods of filter and wrapper are combined so as to reduce the amount of calculation and improve classification accuracies.

1.5 Organizational structure of this paper

The rest of this paper is organized as follows. Section 2 presents the mathematical description and the problem model for feature subset selection. The mathematical model for multi-colony fairness algorithm is described in Sect. 3. Section 4 shows the detailed illustration of multi-colony fairness algorithm. The theoretical analysis of MCFA is presented in Sect. 5. Section 6 reports the experimental results. It also includes some analysis of these results, and finally, the conclusion is offered in the last section.

2 Mathematical description and problem model for feature subset selection

Feature selection is included in discrete optimization problems. The goal of feature selection is to select the optimal feature subset which not only has the least number of selected features, but also can improve both the classification efficiency and the classification accuracy. The subset including all features may reduce the classification efficiency and decrease the classification accuracy because of some interfering features.

The whole search space for optimization contains all possible subsets of features, meaning that its size is:

$SIZE = C_m^0 + C_m^1 + \cdots + C_m^{m-1} + C_m^m = 2^m$, where m is the number of all the features.

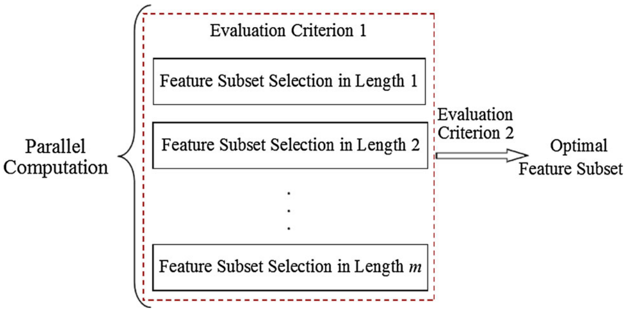
Evaluation Criterion 1
Feature Subset Selection in Length 1
Feature Subset Selection in Length 2
Feature Subset Selection in Length m
Parallel Computation
Evaluation Criterion 2
Optimal Feature Subset

Fig. 1 Feature subset selection model

Table 1 Symbol description used in the problem model for feature subset selection

Symbols	Detail description of the symbols
F	The set of all the features
S	The subset of selected features
fit	The evaluation criterion of feature subsets
Ω	The set of all feature subsets
m	The number of all the features
n	The number of selected features

The uncertainty of the length of feature subsets is one of the difficulties in the process of feature selection. In this paper, the general feature selection problem is reformulated into several fixed length of feature subset selection problems which can be computed in parallel as shown in Fig. 1.

In Fig. 1, the evaluation criterion 1 is the discernibility of the feature subset (DFS), and the evaluation criterion 2 is the comprehensive evaluation of the length of feature subsets and classification accuracies.

The mathematical model can be described as follows: For the data set with m features $F = f_1, f_2, \ldots, f_m$, the selected feature subset with n features is $S(S = s_1, s_2, \ldots, s_m; s_i = \{0, 1\}; i = 1, 2, \ldots, m; sum_{i=1}^{m} s_i = n)$, where s_i represents whether the feature f_i is selected into the feature subset or not. So there is

$$s_i = \begin{cases} 0 & f_i \text{ is not in the feature subset} \\ 1 & f_i \text{ is in the feature subset} \end{cases}.$$

Thus, the feature subset selection problem is represented to solve $\max_{S \in \Omega} fit(S)$ in which fit is the evaluation criterion of the feature subset S, and Ω represents the space of all the feature subsets.

The number of symbols used in the illustration of the problem model and the mathematical model is relatively big. Considering this situation, Tables 1 and 2, respectively, list the symbols and their descriptions used in the problem model and the mathematical model.

FREE paper
freepaper.me
دانلود کننده مقالات علمی

Table 2 Symbol description used in the mathematical model for multi-colony fairness algorithm

Symbols	Detail description of the symbols
N	The number of colonies
M	The number of agents in each colony
C_i	The ith colony
P_j	The jth agents which belong to the same category
P_{ij}	The jth agent in the ith colony
s_{ij}	The resource status of P_{ij}
b_{ij}	The ability value of P_{ij}
p_{ij}	The possibility of adopting the grabbing-resource behavior of P_{ij}
Fit_i	The colony benefit value of C_i
f	The evaluation function of resource statuses
$FitA$	The average value of colony benefits
$PFit_i$	The agent-colony benefit value of P_i
pd_{ij}	The resource pre-allocation scheme of P_{ij}
$pdbs_{ij}$	The quantified result of pd_{ij} before the grabbing-resource behavior
$pdas_{ij}$	The quantified result of pd_{ij} after the grabbing-resource behavior
$FitD_i$	The benefit D-value of C_i
$PFitD_{ij}$	Agent's colony benefit D-value of P_{ij}
ES_{ij}	The effective statement for the grabbing-resource behavior of P_{ij}

3 Mathematical model for multi-colony fairness algorithm

British biologist Darwin's evolutionary theory points out that struggling for survival exists in organisms. The ones adapting themselves to it thrive and others will be eliminated. Genetic algorithm is a kind of computational model which simulates the natural selection of Darwin's theory of evolution and biological evolutionary process of genetic mechanism. It is a method to search for the optimal solution through simulating natural evolutionary process. Multi-colony fairness algorithm (MCFA) in this paper uses the inevitable resource competitive behavior among animal survival struggle for reference which is mentioned as the grabbing-resource behavior before. This behavior is produced under certain conditions rather than animals' genetic behaviors. The possibility of adopting resource competitive behaviors is determined by two factors. One is whether resources are adequate. Animals who have enough resources have no need to adopt resource competitive behaviors. But the animals lacking resources may well adopt resource competitive behaviors. The other factor is whether the ability is enough. The ability here refers to the needs required in the process of the resource competition like physical fitness. No enough ability, no possibility for resource competitive behaviors.

Referring to the animal behavior mentioned above, the fixed-length feature selection model in this paper can be transformed into a resource allocation model. Different from the traditional resource allocation problem, resource competitive behaviors are allowed in the process of resource allocation in this paper. Similarly, agents' possibility of adopting resource competitive behaviors is related to whether the resources are adequate and whether agents have enough abilities. There are three conditions as below:

1. Agents have ample resources;
2. Agents are short of resources, but lack of abilities;
3. Agents are short of resources and have enough abilities.

The corresponding behaviors adopted by agents are as follows:

1. Agents maintain status quo;
2. Agents do nothing;
3. Agents compete for resources with the possibility of resource competitive behaviors.

Agents' abilities depend on their contributions to the colony. The bigger contributions to the colony, the more abilities they have. Although final results of the first and the second corresponding behaviors are the same, there is still a difference between them. The first behavior is active as agents have enough resources. They need not compete for resources. But the second behavior is passive. In the second condition, agents have a desire to complete for resources. However, they can only do nothing for lack of abilities.

The single colony's evolution has singularity and limitations. To overcome these shortcomings, this paper introduces the concept of multi-colonies and divides the multi-colony evolution into the dynamic interactive phase and the colony competition phase. The detailed evolutionary process is shown in Fig. 2.

3.1 Initial model

The fixed-length feature selection problem is transformed into multi-colony m agents resource allocation problem. The initial number of colonies is N; C_i is the ith colony; P_j is the jth agent where M is the number of all features in data sets. The jth agents $P_j = \{P_{ij}(i = 1, \ldots, N)\}$ in different colonies belong to the same category ($j = 1, \ldots, M$).

Definition 1 (*Resource status*) s_{ij} represents the resource status of P_{ij} where $s_{ij} = \begin{cases} 0 & \text{lack of resource} \\ 1 & \text{adequate resource} \end{cases}$. If $s_{ij} = 1$,

FREE paper freepaper.me دانلود کننده مقالات علمی

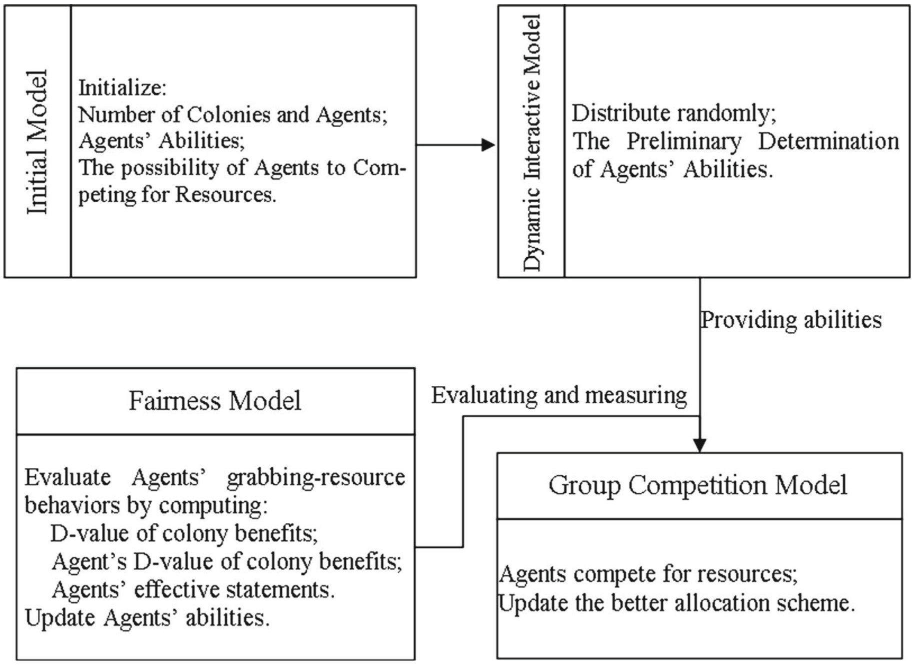
Initial Model
Initialize:
Number of Colonies and Agents;
Agents' Abilities;
The possibility of Agents to Competing for Resources.
Dynamic Interactive Model
Distribute randomly;
The Preliminary Determination of Agents' Abilities.
Providing abilities
Evaluating and measuring
Fairness Model
Evaluate Agents' grabbing-resource behaviors by computing:
D-value of colony benefits;
Agent's D-value of colony benefits;
Agents' effective statements.
Update Agents' abilities.
Group Competition Model
Agents compete for resources;
Update the better allocation scheme.

Fig. 2 Evolution process of MCFA

دانلود کننده مقالات علمی
FREE paper
freepaper.me

P_{ij} would not compete for resources while P_{ij} may well adopt the resource competitive behavior if $s_{ij} = 0$.

Definition 2 (*Ability value*) b_{ij} is the ability value of P_{ij}. The greater the ability is, the more contributions P_{ij} can make to the colony. If $b_{ij} < \varepsilon$, P_{ij} would not compete for resources.

Definition 3 (*Possibility of resource competitive behaviors*) The possibility of taking resource competitive behaviors p_{ij} of P_{ij} is determined by the agent's resource status s_{ij} and its ability value b_{ij}. The computational formula shows as (1):

$$p_{ij} = \begin{cases} (1 - s_{ij}) \times b_{ij} \times e^{b_{ij}-1} & b_{ij} \geq \varepsilon \\ 0 & b_{ij} < \varepsilon \end{cases} \tag{1}$$

The initial number of the colonies produced N is certain which is determined by the complexity of specific issues. The population of agents in each colony is the same and is equal to the number of features in datasets.

3.2 Dynamic interaction model

The multi-colony evolution's main task in the dynamic interaction phase is to finish the preliminary determination of agents' abilities which belong to the same category in different colonies. It will lay the foundations for the competition among every colony.

Definition 4 (*Colony benefit value*) In the current stage, the colony benefit value Fit_i of C_i depends on its resource status. If $s_{ij} = 1$, P_{ij} achieves the resources. And if $s_{ij} = 0$, P_{ij} is short of resources. Based on the feature subset consisting of selected features, the computational formula of evaluation function is: $Fit_i = f(s_i)(s_i = s_{i1}, s_{i2}, \ldots, s_{iM})$, where f is the evaluation function of the resource status which also is the discernibility of feature subset (DFS) in this paper.

In the following detailed explanation of DFS, the notations are just used for temporarily which have no conflict with the full paper. For the $l(l \geq 2)$ classes classification problem, we assume that the number of samples in training sets is n and the sample space dimension is m (i.e., the training set is $\{(x_k, y_k)|x_k \in R^m(m\ dimensional\ real\ space), m \succ 0, y_k \in \{1, \ldots, l\}, l \geq 2, k = 1, \ldots, n\})$. n_j is the number of samples in the jth class (i.e., $\|y_k|y_k = j, k = 1, \ldots, n\| = n_j, j = 1, \ldots, l$). Then, the DFS$_i$ of the selected feature subset which has $i(i = 1, \ldots, m)$ features is defined as the formula (2):

$$\text{DFS}_i = \frac{\sum_{j=1}^{l} \|x^{(j)} - x\|^2}{\sum_{j=1}^{j} \frac{1}{n_j - 1} \sum_{k=1}^{n_j} \|x_k^{(j)} - x^{(j)}\|^2}, \tag{2}$$

where $x, x^{(j)}$ are the mean vectors of the feature subset which consists of the current i features in the whole dataset and in the jth class dataset, respectively; $x_k^{(j)}$ is the feature vector with the current i features of the kth sample in the jth class. The numerator in formula (2) represents the sum of squared distances. The distances refer to ones between the mean vector with the current i features of different classes and the mean vector with current i features in the whole dataset. The bigger the numerator is, the bigger the sparse between classes is. The denominator in formula (2) represents the variance within different classes with the current i features. The smaller the variance is, the more concentrated within classes. Thus, the defined DFS$_i$ is the ratio of the feature subset's distance between classes and its variance within classes with the current i features. The class identification of the feature subset with the current i features will become stronger and stronger as DFS$_i$ becomes greater.

In the current stage, the average of colony benefits is computed as the formula (3):

$$FitA = \frac{1}{N} \times \sum_{i=1}^{N} Fit_i. \tag{3}$$

Definition 5 (*Agent-colony benefit value*) In the current stage, the agent-colony benefit value $PFit_i$ is defined as the average of the colony benefits. These benefits are obtained by colonies in which P_{ji} achieves enough resources. The computational formula shows as (4):

$$PFit_i = \frac{\sum_{j=1,s_{ji}=1}^{N} Fit_j}{\sum_{j=1,s_{ji}=1}^{N} 1}. \tag{4}$$

In the dynamic phase, the resource status s_{ij} of P_{ij} is randomly given which satisfies $\sum_{j=1}^{m} s_{ij} = n$. In addition, agents of the same category in different colonies have the same ability. Agents' abilities in the next stage are related to their current abilities, their current agent-colony benefits, and the current average value of colony benefits. The detailed computational method is the formula (5):

$$b_{ij}(t+1) = \begin{cases} b_{ij}(t)+(1-b_{ij}(t))\times\frac{PFit_j(t)–FitA(t)}{FitA(t)} & \text{if } PFit_j(t)\geq FitA(t),\\ b_{ij}(t)-b_{ij}(t)\times\frac{FitA(t)–PFit_j(t)}{FitA(t)} & \text{if } PFit_j(t)<FitA(t). \end{cases} \tag{5}$$

where t is the current stage number and i starts from 1 to N. The bigger $PFit_i$ is, the more contributions P_i can make to colonies. Thus, their abilities will be greater.

3.3 Group competition model

Every colony enters into the colony competition phase automatically when the dynamic interaction phase ends. And at this time, abilities of agents which are in the same category from different colonies are their initial abilities at the beginning of the colony competition phase. It is worth mentioning that in the colony competition phase, the final resource allocation schemes of agents are determined by the resource pre-allocation scheme of each agent.

Definition 6 (*Resource pre-allocation scheme*) Resource pre-allocation distributes resources to agents according to their abilities by regarding resources as decomposable ones instead of ones in 1 for the unit. The resource pre-allocation scheme of P_{ij} is expressed as pd_{ij} which meets $pd_{ij} \in [0, n]$, $\sum_{j=1}^{M} pd_{ij} = n, i = 1, \ldots, M$. As the resource competitive behavior is allowed in the colony competition phase, $pdbs_{ij}$ and $pdas_{ij}$ $(pdbs_{ij} \in \{0, 1\}, pdas_{ij} \in \{0, 1\})$ are results quantified by the resource pre-allocation schemes before and after agents compete for resources, respectively.

In the colony competition phase, whether P_{ij} competes for resources is represented as rfv_{ij} $\left(rfv_{ij} = \begin{cases} 0 & \text{no resource competition behavior} \\ 1 & \text{compete for resources and achieve resources} \end{cases}\right)$. The value of rfv_{ij} is based on the results of the resource pre-allocation. In this paper, we suppose that agents can obtain resources in the process of resource pre-allocation as long as they compete for resources. Then, resources of the other agents will decrease correspondingly changing as the formula (6):

$$pd_{ij} = pd_{ij} + (n - pd_{ij}) \times b_{ij} \times (1 - e^{-b_{ij}}) \quad \text{if } rfv_{ij} = 1. \tag{6}$$

To ensure the sum of all agents' resources to be n, the resources of all agents need n-normalization processing after agents end resource competition behaviors. The processing method shows as (7):

$$pd_{ij} = pd_{ij} \times \frac{n}{\sum_{j=1}^{m} pd_{ij}}. \tag{7}$$

Whether agents having done resource competition behaviors finally achieve enough resources or not is uncertain even though their resources increase as they compete for resources. So we define $rfobv_{ij}$ to mark whether P_{ij} obtains enough resources after competing for resources:

$$rfobv_{ij} = \begin{cases} 0 & \text{compete for resources but not obtain them} \\ 1 & \text{compete for resources and obtain them} \end{cases}.$$

3.4 Fairness model

In this paper, we have introduced agents' resource competition behaviors into the resource allocation problem in the colony competition phase. To make the whole colony better adaptive to the living environment (i.e., to make the colony achieve more colony benefits), the proportional allocation, effective statements, and the age factor are introduced into the fairness model to evaluate and deal with agents' resource competition behaviors. These measures are more conducive to the development of colonies.

دانلود کننده مقالات علمی
freepaper.me
FREE paper

3.4.1 Proportional distribution

In the process of resource pre-allocation, quantified results of resource pre-allocation schemes ($pdbs_{ij}$, $pdas_{ij}$) may be different due to agents' resource competition behaviors. Colony benefits produced by quantified results of the resource pre-allocation will differ accordingly.

Definition 7 (*D-value of colony benefits*) The benefit *D*-value $FitD_i$ of C_i refers to the difference in colony benefits obtained before and after agents' resource competition behaviors in the resource pre-allocation phase ($FitD_i = f(pdas_i) - f(pdbs_i)$).

The proportional allocation introduced into the fairness model manifests itself in allocating *D*-value of colony benefits to agents proportionally. There is a detailed description of it. After C_i has accomplished the quantification of resource pre-allocation in tth stage, P_{ij} satisfying $pdbs_{ij} = 0$ and $b_{ij} > \varepsilon$ will adopt the resource competition behavior based on $p_{ij} = (1 - pdbs_{ij}) \times b_{ij} \times e^{b_{ij}-1}$. After this, *D*-value of colony benefits obtained will be proportionally allocated to all the agents meeting $rfobv_{ij} = 1$ according to their abilities. Agent's *D*-value of colony benefits is computed as the formula (8):

$$PFitD_{ij} = \begin{cases} \frac{FitD_i \times b_{ij}}{\sum_{j=1, rfobv_{ij}=1}^{m} b_{ij}} & \text{if } rfobv_{ij} = 1, \\ 0 & \text{if } rfobv_{ij} = 0. \end{cases} \tag{8}$$

Agent's *D*-value of colony benefits is one of the most important bases for its ability's update. To reflect the *D*-value of colony benefits resulting from different agents more fairly, it is allocated proportionally to every agent by introducing the proportional distribution mechanism into the fairness model.

3.4.2 Effective statements

To show the effectiveness of agents' resource competition behaviors, we have introduced the concept of effective statements into the fairness model. The effective statement for the resource competition behavior of P_{ij} is expressed as ES_{ij}. It indicates long-term effectiveness of P_{ij}'s resource competition behavior from the beginning of the colony competition phase to the current tth stage. The bigger ES_{ij} is, the more effective P_{ij}'s resource competition behavior is.

ES_{ij}'s computational formula is (9):

$$ES_{ij}(t) = \sum_{l=k}^{t} PFitD_{ij}(l) \times e^{-age_{ij}(l)}. \tag{9}$$

The value of ES_{ij} is related to agent's *D*-value of colony benefits $PFitD_{ij}$ produced by P_{ij}'s resource competition behavior in the colony competition phase. $age_{ij}(l)$ is the age of agent's *D*-value of colony benefits in the current stage which was produced by P_{ij} at the lth stage.

3.4.3 The age factor

A paper in Science (Mersch et al. 2013) says that they used a tracking system to continuously monitor individually tagged workers in six colonies of the ant Camponotus fellah over 41 days. Network analysis of more than 9 million interactions revealed three distinct groups that differ in behavioral repertoires. Each group represents a functional behavioral unit with workers moving from one group to the next as they age. The rate of interactions was much higher within groups than between groups. The ant workers are divided into different colonies according to their ages. Different colonies mean different behavioral functions which also means that the abilities of ant workers are different due to their age. The phenomenon of dividing colonies based on ages does not only exist in ants' life, but also exists in the process of learning knowledge. People in different ages have different learning abilities. The remaining amount and the effectiveness of knowledge learned at certain age will reduce as time goes (Herzfeld et al. 2014; Averell and Heathcote 2011). Similarly, in the process of the colony resource allocation, the effect of the agent's *D*-value of colony benefits produced by the resource competition behavior at different stages is not the same.

Based on this found from objective laws, the age factor is introduced into the process of analyzing the effectiveness of agents' resource competition behaviors.

In the current stage, agent's *D*-value of colony benefits produced by P_{ij} at the lth stage is in the age of $age_{ij}(l)$. At the tth stage, there is $age_{ij}(l) = t - l(age_{ij}(t) = 0)$.

$cpb_{ij}(l) = e^{-age_{ij}(l)}$ represents $PFitD_{ij}(l)$'s capacity to optimize colony benefits.

The higher $cpb_{ij}(l)$ is, the stronger $PFitD_{ij}(l)$'s capacity to optimize colony benefits is. Conversely, the lower $cpb_{ij}(l)$ is, the weaker $PFitD_{ij}(l)$'s capacity to optimize colony benefits is. The forgetting curve of knowledge learned by people is shown in Fig. 3. Obviously, $PFitD_{ij}(l)$'s capacity to optimize colony benefits is also related to its age. The bigger the age of $PFitD_{ij}(l)$ is, the weaker its capacity to optimize colony benefits in this stage is. In this paper, we assume that the reducing trend of $PFitD_{ij}(l)$'s capacity to optimize colony benefits as its age increases is the same as people's knowledge forgetting curve which can be represented as $cpb_{ij}(l) = e^{-age_{ij}(l)}(age_{ij}(l) = 0 \longrightarrow cpb_{ij}(l) = 1; age_{ij}(l) = +\infty \longrightarrow cpb_{ij}(l) = 0)$.

The introduction of the proportional allocation, effective statements, and the age factor into the fairness model is

دانلود کننده مقالات علمی freepaper.me FREE paper

FREE paper freepaper.me دانلود کننده مقالات علمی

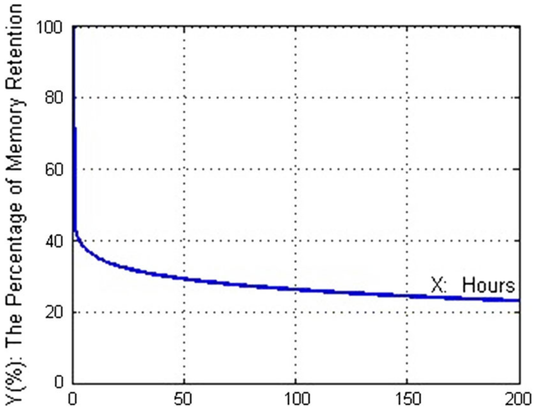

Fig. 3 Knowledge forgetting curve

used to evaluate and deal with agents' resource competition behaviors in the process of colony resource allocation. The evaluation and measures of agents' resource competition behaviors in this paper are reflected in the update of agents' abilities. The detailed computational formula shows as (10):

$$b_{ij}(t+1) = \begin{cases} b_{ij}(t) + \dfrac{1-b_{ij}(t)}{1+\dfrac{\alpha}{e^{PFitD_{ij}(t)}}} & \text{if } rfobv_{ij}(t)=1 \text{ and } PFitD_{ij}(t)>0, \\ \dfrac{b_{ij}(t)}{1+\dfrac{\beta}{e^{ES_{ij}(t)}}} & \text{if } rfobv_{ij}(t)=1 \text{ and } PFitD_{ij}(t)<0, \\ b_{ij}(t) & \text{otherwise.} \end{cases} \tag{10}$$

In formula (10), the parameters α and β represent the influence of $PFitD_{ij}$ and ES_{ij} on b_{ij}, respectively. Their values are both greater than zero. And the smaller the value is, the greater the influence of what the parameter represents is.

4 Multi-colony fairness algorithm

The specific steps of multi-colony fairness algorithm show as follows:

1. Colony initialization
   Initialize the number of colonies as N, and each colony has M agents where M is also the number of all features in the dataset. Each agent's ability is $b_{ij} = 0.5(i = 1, 2, \ldots, N, j = 1, 2, \ldots, M)$. The possibility of agents competing for resources is $p_{ij} = (1-s_{ij}) \times b_{ij} \times e^{b_{ij}-1} = 0.3$.
2. Dynamic interaction phase
   In the dynamic interaction phase, each colony allocates resources randomly. Then, compute colony benefit value $Fit_i(i = 1, 2, \ldots, N)$, agent's colony benefits value $PFit_j(j = 1, 2, \ldots, M)$. Followed by the computation above is the preliminary determination of agents' abilities in the same category. It will be the heuristic information for the colony competition phase.
3. Colony competition phase
   Divide the resource allocation in every stage into two parts: the resource pre-allocation and the resource final allocation. In the process of resource pre-allocation, agents' resource competition behaviors are added. Then, compute D-value of colony benefits $FitD_i(i = 1, 2, \ldots, N)$, agent's D-value of colony benefits $PFitD_{ij}$, and agents' effective statements $ES_{ij}(t)$. After those, the better resource allocation scheme needs to be selected and agents' abilities need to be updated until reaching the iterative number. Table 3 is the pseudo-code of multi-colony fairness algorithm.

5 Theoretical analysis of algorithm (MCFA)

5.1 Convergence analysis

In this subsection, we show all agents can converge to their stable equilibrium states through MCFA. In other words, MCFA can converge to a stable equilibrium state.

In mathematics, stability theory deals with the stability of the solutions of differential equations and dynamical systems. Definitions of stability include Lyapunov stability and structural stability. Lyapunov stability occurs in the study of dynamical systems. Lyapunov functions are a family of functions that can be used to demonstrate the stability or instability of some state points of a system. The demonstration of stability or instability requires finding a Lyapunov function for the given dynamical system. In this paper, the ecosystem comprised of all the agents in different colonies just can be treated as a kind of dynamical system.

Lyapunov second theorem on stability considers a function $L(X)$ such that

1) $L(X) > 0$ (positive definite);
2) $\mathrm{d}L(X(t))/\mathrm{d}t < 0$ (negative definite).

Then, $L(X(t))$ is called a Lyapunov function candidate, and X is asymptotically stable in the sense of Lyapunov. So if we can find a qualified Lyapunov function in our dynamical system, we can prove the proposed algorithm can converge to a stable equilibrium state.

Theorem 1 *If the ability of P_{ij} satisfies the update strategy (10), MCFA can converge to a stable equilibrium state.*

Table 3 MCFA pseudo-code

```
Set t=0;
initialize the number of colonies and agents in every colony $N, M$ and agents' abilities $b_{ij}$;
For each feature subset length
    While $t < DT$
        For each colony
            allocate resources randomly;
            compute $Fit_i$ and $PFit_{ij}$ using formulas (2) and (4);
        End For
        compute $FitA$ using (3);
        compute $b_{ij}$ using (5);
    End While
    While $t < T$
        For each colony
            resource pre-allocation;
            each agent compute for resources according $p_{ij}$;
            compute D-value of colony benefits using $FitD_i = f(pdas_i) - f(pdbs_i)$;
            compute $PFitD_{ij}$ and $ES_{ij}(t)$ using formula (8) and (9);
            resource final allocation;
            update agents' abilities using (10);
        End For
    End While
    select the best Colony;
    select the optimal feature subset according to evaluation criterion (2);
End For
```

Proof We use $B(t)$ to represent the abilities $(b_i)_{1\times m}(t)$ of all the agents in C_i. The Lyapunov function is defined as $L(B(t)) : L(B(t)) \doteq B(t)^{-1}$.

According to the theorem of the Lyapunov asymptotic stability, we only need to prove that the Lyapunov function $L(B(t))$ satisfies $\begin{cases} a. & L(B(t)) > 0 \quad \text{(positive definite)} \\ b. & \frac{\mathrm{d}L(B(t))}{\mathrm{d}t} < 0 \quad \text{(negative definite)} \end{cases}$.

Proof of a It is known that $b_i(t) > 0, i = 1, 2, \ldots, M$ is true at the beginning of the colony competition phase. So we need to prove $b_i(t) > 0$ is always met during the process of changing. According to the update strategy (10) of agents' abilities in the colony competition phase, we know $b_i(t)$ may change in two situations like the formula (11) and (12):

$$b_i(t+1) = b_i(t) + \frac{1 - b_i(t)}{1 + \frac{\alpha}{e^{PFitD_i(t)}}}. \tag{11}$$

$$b_i(t+1) = \frac{b_{ij}(t)}{1 + \frac{\beta}{e^{ES_{ij}(t)}}}. \tag{12}$$

As $\alpha > 0$ and $\beta > 0$, the inequations (13) and (14) clearly hold as follows.

$$1 + \frac{\alpha}{e^{PFitD_i(t)}} > 1. \tag{13}$$

$$1 + \frac{\beta}{e^{ES_i(t)}} > 1. \tag{14}$$

So, we can get the inequations (15) and (16).

$$b_i(t) + \frac{1 - b_i(t)}{1 + \frac{\alpha}{e^{PFitD_i(t)}}} < b_i(t) + 1 - b_i(t) = 1; \tag{15}$$

$$0 < \frac{b_i(t)}{1 + \frac{\beta}{e^{ES_i(t)}}} < b_i(t). \tag{16}$$

Then, we can prove $0 < b_i(t) < 1$; thus, $L(B(t)) > 0$ obviously holds.

Proof of b Based on the Lyapunov function $L(B(t))$ and the derivation process, we can get the formula (17):

$$\frac{\mathrm{d}L(B(t)^{-1})}{\mathrm{d}t} = \sum_i -b_i(t)^{-2} \cdot \frac{\mathrm{d}b_i(t)}{\mathrm{d}t}. \tag{17}$$

Due to $0 < b_i(t) < 1$, $-b_i(t)^{-2} < 0$ is true.

According to the theorem of the Lyapunov asymptotic stability, if the value of $\frac{\mathrm{d}L(B(t)^{-1})}{\mathrm{d}t}$ is less than zero, the algorithm can converge to a stable equilibrium state. Now, we have known $-b_i(t)^{-2} < 0$, so we only need to prove $\frac{\mathrm{d}b_i(t)}{\mathrm{d}t} > 0$.

According to the update strategy (10) of b_i, the result of derivation of b_i shows as the formula (18):

دانلود کننده مقالات علمی
freepaper.me
FREE paper

$$\frac{\mathrm{d}b_i(t)}{\mathrm{d}t} = \begin{cases} 1+\dfrac{-1}{1+\dfrac{\alpha}{e^{PFitD_i(t)}}} & \text{if} \quad rfobv_i(t)=1 \text{ and } PFitD_i(t)>0; \\ \dfrac{1}{1+\dfrac{\beta}{e^{ES_i(t)}}} & \text{if} \quad rfobv_i(t)=0 \text{ and } PFitD_i(t)<0; \\ 1 & \text{otherwise.} \end{cases} \tag{18}$$

Based on the inequations (13), we can get the long inequality (19) as follows.

$$0 = 1+(-1) < 1+\frac{-1}{1+\dfrac{\alpha}{e^{PFitD_i(t)}}} < 0+1 < 1. \tag{19}$$

The simple result of the inequality (19) shows as the inequality (20).

$$0 < 1+\frac{-1}{1+\dfrac{\alpha}{e^{PFitD_i(t)}}} < 1. \tag{20}$$

Similarly, the inequality (21) can be obtained on account of the equality (14).

$$0 < \frac{1}{1+\dfrac{\beta}{e^{ES_i(t)}}} < 1. \tag{21}$$

According to the formulas (18), (20), and (21), we can prove that $\dfrac{\mathrm{d}b_i(t)}{\mathrm{d}t} > 0$ is true. Then, based on the theorem of the Lyapunov asymptotic stability, the algorithm is convergent because of $\begin{cases} L(B(t)^{-1}) > 0 \\ \dfrac{\mathrm{d}L(B(t)^{-1})}{\mathrm{d}t} < 0 \end{cases}$. □

5.2 Effectiveness analysis

The effectiveness of multi-colony fairness algorithm solving the feature subset selection problem can be proved from two aspects in this paper.

1. Abilities of agents who make less contributions to colony benefits will become smaller and smaller. And they tend to be 0 finally.
2. Abilities of agents who make more contributions to colony benefits will become bigger and bigger. And they tend to be 1 finally.

Lemma 1 *If the contributions to colony benefits made by P_k are the least, its ability b_k will decrease and must tend to be 0 ($b_k \to 0$).*

As each colony computes in parallel and the process is the same, here we analyze agents in a colony without loss of generality.

Proof Assume that at the current stage, only P_k adopts the resource competition behavior. Then, we can get: $\begin{cases} pda_k(t) = pdb_k(t) + (n - pdb_k(t)) \times b_k \times (1 - e^{-b_k}) \\ pda_j(t) = pdb_j(t) \quad j \in [1, m], j \in Z, j \neq k \end{cases}$.

Because only P_k has competed for resources, the ordering of other agents' resources after P_k's resource competition behavior is the same as that before P_k's resource competition behavior. Suppose the ranking position of P_k after its resource competition behavior is g and n is the number of agents. Thus, there will be two possible scenarios. One is $g > n$ which means that P_k does not obtain enough resources finally. The other is $g \leq n$ which means that P_k obtains enough resources. For the first scenario, the ability of P_k will have no change. For the second scenario, assume that P_k obtains the resources instead of P_g after P_k has competed for resources. Others remain unchanged. According to the definitions of colony benefit value and D-value of colony benefits, we know the f function is the discernibility of a feature subset. Here, we assume **x1** and **x2** as feature subset vectors before and after P_k's resource competition behavior, respectively. Then, we can get the analytical procedure as formula (22).

دانلود کننده مقالات علمی
FREE freepaper.me paper

$$\begin{aligned}
FitD &= f(pdas) - f(pdbs)\\
&= \frac{\sum_{j=1}^{l}\|x1^{(j)}-x1\|^2}{\sum_{j=1}^{l}\frac{1}{n_j-1}\sum_{k=1}^{n_j}\|x1_k^{(j)}-x1^{(j)}\|^2} - \frac{\sum_{j=1}^{l}\|x2^{(j)}-x2\|^2}{\sum_{j=1}^{l}\frac{1}{n_j-1}\sum_{k=1}^{n_j}\|x2_k^{(j)}-x2^{(j)}\|^2}\\
&= \frac{\sum_{j=1}^{l}\sum_{h=1}^{n}(x1_h^{(j)}-x1_h)^2}{\sum_{j=1}^{l}\frac{1}{n_j-1}\sum_{k=1}^{n_j}\sum_{h=1}^{n}(x1_{kh}^{(j)}-x1_h^{(j)})^2} - \frac{\sum_{j=1}^{l}\sum_{h=1}^{n}(x2_h^{(j)}-x2_h)^2}{\sum_{j=1}^{l}\frac{1}{n_j-1}\sum_{k=1}^{n_j}\sum_{h=1}^{n}(x2_{kh}^{(j)}-x2_h^{(j)})^2}\\
&= \frac{\sum_{j=1}^{l}\sum_{h=1}^{n-1}(x_h^{(j)}-x_h)^2+\sum_{j=1}^{l}(x_k^{(j)}-x_k)^2}{\sum_{j=1}^{l}\frac{1}{n_j-1}\sum_{k=1}^{n_j}\sum_{h=1}^{n-1}(x_{kh}^{(j)}-x_h^{(h)})^2+\sum_{j=1}^{l}\frac{1}{n_j-1}\sum_{k=1}^{n_j}(x_{kk}^{(j)}-x_k^{(j)})^2}\\
&\quad - \frac{\sum_{j=1}^{l}\sum_{h=1}^{n-1}(x_h^{(j)}-x_h)^2+\sum_{j=1}^{l}(x_g^{(j)}-x_g)^2}{\sum_{j=1}^{l}\frac{1}{n_j-1}\sum_{k=1}^{n_j}\sum_{h=1}^{n-1}(x_{kh}^{(j)}-x_h^{(h)})^2+\sum_{j=1}^{l}\frac{1}{n_j-1}\sum_{k=1}^{n_j}(x_{kg}^{(j)}-x_g^{(j)})^2}\\
&= \frac{A+\sum_{j=1}^{l}(x_k^{(j)}-x_k)^2}{B+\sum_{j=1}^{l}\frac{1}{n_j-1}\sum_{k=1}^{n_j}(x_{kk}^{(j)}-x_k^{(j)})^2} - \frac{A+\sum_{j=1}^{l}(x_g^{(j)}-x_g)^2}{B+\sum_{j=1}^{l}\frac{1}{n_j-1}\sum_{k=1}^{n_j}(x_{kg}^{(j)}-x_g^{(j)})^2}.
\end{aligned} \tag{22}$$

A and B in formula (22), respectively, represent $\sum_{j=1}^{l}\sum_{h=1}^{n-1}(x_h^{(j)}-x_h)^2$ and $\sum_{j=1}^{l}\frac{1}{n_j-1}\sum_{k=1}^{n_j}\sum_{h=1}^{n-1}(x_{kh}^{(j)}-x_h^{(h)})^2$ for short. It has been mentioned in the conditions that contributions to colony benefits made by P_k are the least which also means that the discernibility of the kth feature is the least. Clearly, we can obtain inequations (23) and (24).

$$\sum_{j=1}^{l}(x_k^{(j)}-x_k)^2 < \sum_{j=1}^{l}(x_g^{(j)}-x_g)^2. \tag{23}$$

$$\sum_{j=1}^{l}\frac{1}{n_j-1}\sum_{k=1}^{n_j}(x_{kk}^{(j)}-x_k^{(j)})^2 > \sum_{j=1}^{l}\frac{1}{n_j-1}\sum_{k=1}^{n_j}(x_{kg}^{(j)}-x_g^{(j)})^2. \tag{24}$$

From the results of (22), (23) and (24), it can be obtained as the formula (25).

$$FitD = \frac{A+\sum_{j=1}^{l}(x_k^{(j)}-x_k)^2}{B+\sum_{j=1}^{l}\frac{1}{n_j-1}\sum_{k=1}^{n_j}(x_{kk}^{(j)}-x_k^{(j)})^2} - \frac{A+\sum_{j=1}^{l}(x_g^{(j)}-x_g)^2}{B+\sum_{j=1}^{l}\frac{1}{n_j-1}\sum_{k=1}^{n_j}(x_{kg}^{(j)}-x_g^{(j)})^2} < 0. \tag{25}$$

According to the update strategy (10) of agents' abilities, we can obtain $b_k(t+1) < b_k(t)$.

From the proof above, we can prove that abilities of agents who make less contributions to colony benefits will become smaller and smaller. And they tend to be 0 finally. □

Lemma 2 *If the contributions to colony benefits made by P_k are the most, its ability b_k will increase and must tend to be 1 ($b_k \to 1$).*

Proof Assume that at the current stage, only P_k adopts the resource competition behavior. Then, we can get: $\begin{cases} pda_k(t) = pdb_k(t)+(n-pdb_k(t))\times b_k\times(1-e^{-b_k}) \\ pda_j(t)=pdb_j(t) \quad j\in[1,m],\ j\in Z,\ j\neq k \end{cases}.$

Because only P_k has competed for resources, the ordering of other agents' resources after P_k's resource competition behavior is the same as that before P_k's resource competition behavior. Suppose that the ranking position of P_k after its resource competition behavior is g and n is the number of agents. Thus, there will be two possible scenarios. One is $g > n$ which means P_k does not obtain enough resources finally. The other is $g \le n$ which means P_k obtains the resources. For the first scenario, the ability of P_k will have no change. For the second scenario, assume that P_k obtains the resources instead of P_g after P_k has competed for resources. Others remain unchanged. Similar to the analytical procedure of the formula (22), we can get the formula (26).

FREE paper freepaper.me دانلود کننده مقالات علمی

FREE paper freepaper.me دانلود کننده مقالات علمی

$$\begin{aligned}
FitD &= f(pdas) - f(pdbs)\\
&= \frac{\sum_{j=1}^{l} \|x1^{(j)} - x1\|^2}{\sum_{j=1}^{l} \frac{1}{n_j - 1} \sum_{k=1}^{n_j} \|x1_k^{(j)} - x1^{(j)}\|^2} - \frac{\sum_{j=1}^{l} \|x2^{(j)} - x2\|^2}{\sum_{j=1}^{l} \frac{1}{n_j - 1} \sum_{k=1}^{n_j} \|x2_k^{(j)} - x2^{(j)}\|^2}\\
&= \frac{\sum_{j=1}^{l} \sum_{h=1}^{n} (x1_h^{(j)} - x1_h)^2}{\sum_{j=1}^{l} \frac{1}{n_j - 1} \sum_{k=1}^{n_j} \sum_{h=1}^{n} (x1_{kh}^{(j)} - x1_h^{(j)})^2} - \frac{\sum_{j=1}^{l} \sum_{h=1}^{n} (x2_h^{(j)} - x2_h)^2}{\sum_{j=1}^{l} \frac{1}{n_j - 1} \sum_{k=1}^{n_j} \sum_{h=1}^{n} (x2_{kh}^{(j)} - x2_h^{(j)})^2}\\
&= \frac{\sum_{j=1}^{l} \sum_{h=1}^{n-1} (x_h^{(j)} - x_h)^2 + \sum_{j=1}^{l} (x_k^{(j)} - x_k)^2}{\sum_{j=1}^{l} \frac{1}{n_j - 1} \sum_{k=1}^{n_j} \sum_{h=1}^{n-1} (x_{kh}^{(j)} - x_h^{(h)})^2 + \sum_{j=1}^{l} \frac{1}{n_j - 1} \sum_{k=1}^{n_j} (x_{kk}^{(j)} - x_k^{(j)})^2}\\
&\quad - \frac{\sum_{j=1}^{l} \sum_{h=1}^{n-1} (x_h^{(j)} - x_h)^2 + \sum_{j=1}^{l} (x_g^{(j)} - x_g)^2}{\sum_{j=1}^{l} \frac{1}{n_j - 1} \sum_{k=1}^{n_j} \sum_{h=1}^{n-1} (x_{kh}^{(j)} - x_h^{(h)})^2 + \sum_{j=1}^{l} \frac{1}{n_j - 1} \sum_{k=1}^{n_j} (x_{kg}^{(j)} - x_g^{(j)})^2}\\
&= \frac{A + \sum_{j=1}^{l} (x_k^{(j)} - x_k)^2}{B + \sum_{j=1}^{l} \frac{1}{n_j - 1} \sum_{k=1}^{n_j} (x_{kk}^{(j)} - x_k^{(j)})^2} - \frac{A + \sum_{j=1}^{l} (x_g^{(j)} - x_g)^2}{B + \sum_{j=1}^{l} \frac{1}{n_j - 1} \sum_{k=1}^{n_j} (x_{kg}^{(j)} - x_g^{(j)})^2}.
\end{aligned} \tag{26}$$

A and B in formula (26) also, respectively, represent $\sum_{j=1}^{l} \sum_{h=1}^{n-1} (x_h^{(j)} - x_h)^2$ and $\sum_{j=1}^{l} \frac{1}{n_j - 1} \sum_{k=1}^{n_j} \sum_{h=1}^{n-1} (x_{kh}^{(j)} - x_h^{(h)})^2$ for short. It has been mentioned in the conditions that contributions to colony benefits made by P_k are the most which also means that the discernibility of the kth feature is the most. Then, we can get the inequations (27) and (28).

$$\sum_{j=1}^{l} (x_k^{(j)} - x_k)^2 > \sum_{j=1}^{l} (x_g^{(j)} - x_g)^2. \tag{27}$$

$$\sum_{j=1}^{l} \frac{1}{n_j - 1} \sum_{k=1}^{n_j} (x_{kk}^{(j)} - x_k^{(j)})^2 < \sum_{j=1}^{l} \frac{1}{n_j - 1} \sum_{k=1}^{n_j} (x_{kg}^{(j)} - x_g^{(j)})^2. \tag{28}$$

Based on the formulas (26), (27), and (28), it can be obtained as the formula (29):

$$FitD = \frac{A + \sum_{j=1}^{l} (x_k^{(j)} - x_k)^2}{B + \sum_{j=1}^{l} \frac{1}{n_j - 1} \sum_{k=1}^{n_j} (x_{kk}^{(j)} - x_k^{(j)})^2} - \frac{A + \sum_{j=1}^{l} (x_g^{(j)} - x_g)^2}{B + \sum_{j=1}^{l} \frac{1}{n_j - 1} \sum_{k=1}^{n_j} (x_{kg}^{(j)} - x_g^{(j)})^2} > 0. \tag{29}$$

According to the update strategy (10) of agents' abilities, we can prove $b_k(t+1) > b_k(t)$.

From the proof above, we can prove that the abilities of agents who make more contributions to colony benefits will become bigger and bigger. And they tend to be 1 finally. □

6 Experimental results and analysis

To verify the effectiveness of our method on real-world problems, seven datasets (i.e., Nos. 1–7) from UCI machine learning repository and one gene microarray dataset (No. 8) are employed. The detailed description of datasets is presented in Table 4. What has to be told is that the glass dataset is divided into two classes (i.e., window glass and non-window glass). Four samples having missing values are removed from

Table 4 Characteristics of the used datasets

Datasets	Patterns	Features	Classes
Iris	150	4	3
Glass	214	9	2
Wine	178	13	3
WDBC	569	30	2
WPBC	194	33	2
Handwrite	323	255	2
Colon	62	2000	2
Arcene	900	10,000	2

the WPBC dataset. Besides, only the first two classes of the handwrite dataset are chosen for classification. The experiments in this paper are executed by MATLAB in the parallel computers with high performance which have the Lenovo Deepcomp 6800 server.

The effectiveness of our method will be verified from three aspects: (1) the comparison of classification accuracies between feature subsets with all features and feature subsets selected by MCFA; (2) the comparison of classification accuracies, the number of selected features, and the execution time with four classic feature selection algorithms; and (3) the comparison of classification accuracies, the number of selected features, and the execution time with three mainstream feature selection algorithms.

To obtain experimental results with more statistical significance, two methods are used to divide the training set and the testing set in this paper: (1) five cross-validation experiments (i.e., the samples of each dataset are divided into five parts randomly and four of them regard as training samples with one as testing samples in every experiment). This method is used in the first and second parts of verifying our method's effectiveness. (2) In each run, the normalized datasets are randomly split into a training set (2/3 of dataset) and a test set (1/3 of dataset). This method is used in the third part of verifying our method's effectiveness.

6.1 Effectiveness of multi-colony fairness algorithm feature selection

In this section, the effectiveness of the proposed method is verified through comparing classification accuracies and feature numbers before and after multi-colony fairness algorithm. Table 5 shows the detailed data.

It is obviously that seven of the eight datasets have reduced the number of selected features and also improved classification accuracies through applying our method. Although after applying MCFA the Handwrite data set has not improved its classification accuracy, its number of selected features has been reduced greatly. This can improve the efficiency of the classifier vastly. The data in Table 5 verify the effectiveness of MCFA from one aspect.

Figures 4, 5, 6, 7, 8, and 9 are average training and testing accuracies of five cross-validation experiments with MCFA in Iris, Glass, Wine, WDBC, WPBC, and Handwrite, respectively. From these, we find training classification accuracies will become bigger and bigger with the increase in the number of selected features normally. But testing classification accuracies do not go in the same way. It indicates that there are redundant and interference features in datasets. Classification accuracies obtained by MCFA have some differences with the data as shown in Figs. 4, 5, 6, 7, 8, and 9. It is better than all the classification accuracies of different numbers of selected features. That is because for different training and testing datasets of the same dataset, MCFA can obtain better classification accuracies which is shown in Table 5.

Table 5 Comparison of selected feature number and classification accuracy before and after feature selection

Datasets	Number of selected features	Classification accuracy (%)
Iris	4	95.47
	1.6	**97.33**
Glass	9	87.44
	2.6	**93.95**
Wine	13	68.33
	2.2	**89.44**
WDBC	30	87.61
	3.2	**92.57**
WPBC	33	56.41
	2.2	**77.95**
Handwrite	255	99.69
	7.6	99.69
Colon	2000	72.5
	12	**95**
Arcene	10,000	56
	5	**77.5**

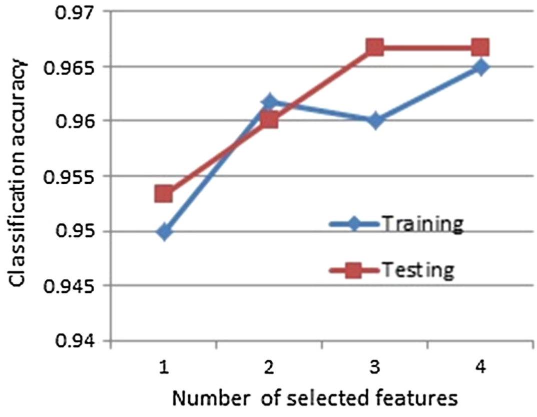

Fig. 4 Average classification accuracies of five cross-validation experiments in Iris with MCFA

Figures 10 and 11 are the datasets of Colon and Arcene's mean testing accuracies of five cross-validation experiments. In these two figures, the number of selected features ranges from 1 to 100. The mean testing accuracies of Colon in Fig. 10 maintain the steady state before the number of selected features is lower than 40. But when the number of selected features changes from 40 to 100, accuracies fall firstly then maintain a low value. We can get there are many features with redundancy and interference in Colon. The number of selected features of Colon through MCFA is 12 which indicates lots of redundant and interfering

FREE paper freepaper.me دانلود کننده مقالات علمی

دانلود کننده مقالات علمی
freepaper.me
FREE paper

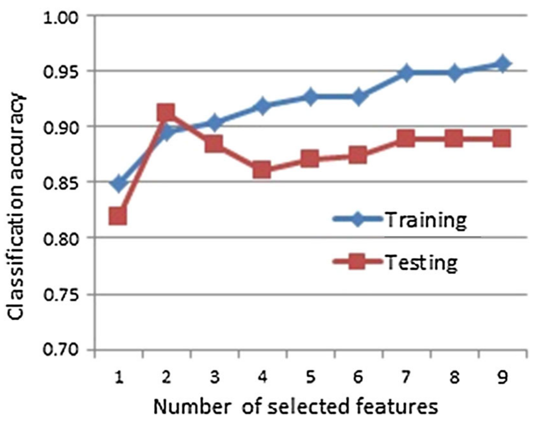

Fig. 5 Average classification accuracies of five cross-validation experiments in Glass with MCFA

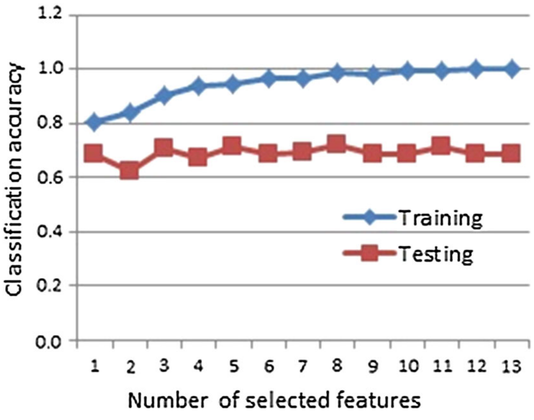

Fig. 6 Average classification accuracies of five cross-validation experiments in Wine with MCFA

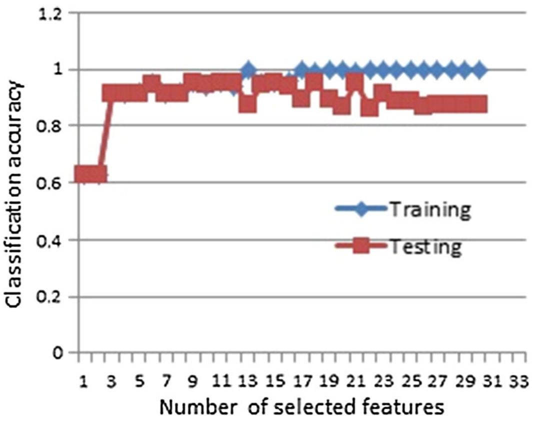

Fig. 7 Average classification accuracies of five cross-validation experiments in WDBC with MCFA

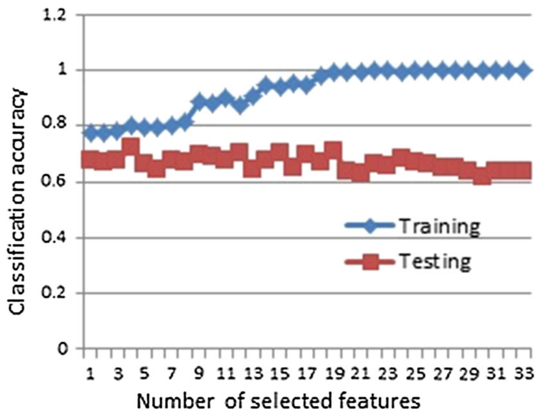

Fig. 8 Average classification accuracies of five cross-validation experiments in WPBC with MCFA

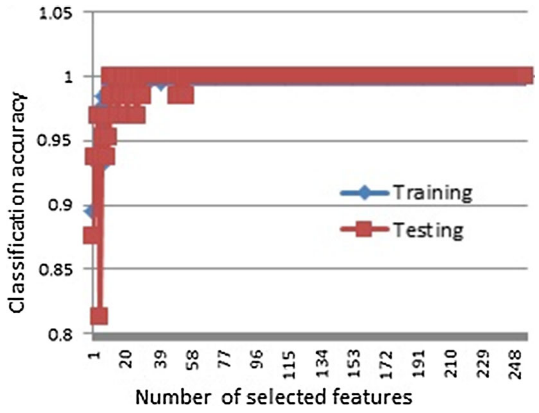

Fig. 9 Average classification accuracies of five cross-validation experiments in Handwrite with MCFA

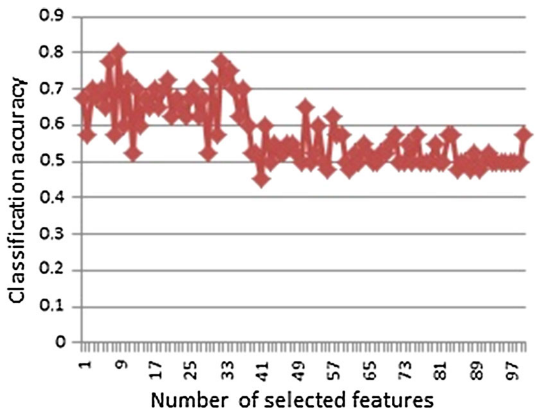

Fig. 10 Average classification accuracies of five cross-validation experiments in Colon with MCFA

features have been reduced. Figure 11 depicts testing accuracies of Arcene decline with the increase in the number of selected features. It states clearly that a large number of redundant features exists in Arcene. Five and 77.5 % are the number of selected features and the mean testing accuracy through MCFA. It is obviously multi-colony fairness algorithm can reduce redundant and interfering features effectively.

دانلود کننده مقالات علمی freepaper.me FREE paper

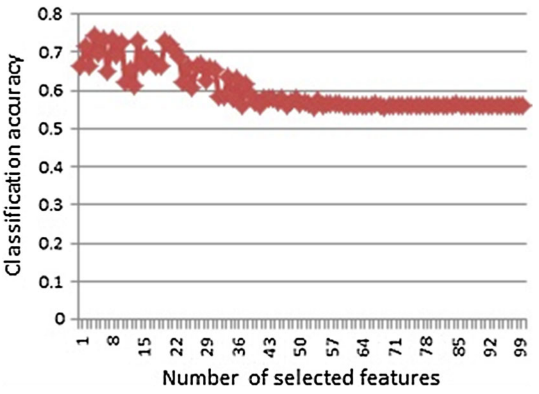

Fig. 11 Average classification accuracies of five cross-validation experiments in Arcene with MCFA

6.2 Comparison experiment

6.2.1 Comparison experiments with four classic feature selection algorithms

Multi-colony fairness algorithm (MCFA) is compared with other four classic feature selection algorithms(SFS, SBS, SFFS, and SBFS) in this section. We use these five methods to do feature selection of six datasets and compare the number of selected features and corresponding classification accuracies. Experimental results are shown in Table 6. In this section, we use the Friedman test and Wilcoxon's test to analyze the performance of MCFA with four classic feature selection algorithms, and the significance is checked at a level of 0.05 (Garcia et al. 2009; Demsar 2006).

With the analysis of results in Table 6, it is apparently MCFA has great advantages in classification accuracies, and it has reduced a great quantity of features in different datasets. Through analyzing results listed in Table 6, the average Friedman ranks on different performances of used datasets are shown in Table 7. The results indicate that MCFA performs differently on the length of feature subsets and classification accuracies. The performance of MCFA is better than other four classic algorithms not only on the length of feature subsets, but also on the classification accuracies. Table 8 depicts the *p* values of applying Wilcoxon's test between MCFA and other four classic algorithms on used datasets with different performances. R+ represents the sum ranks for the problems in which MCFA outperforms the comparison algorithms. The p values under the significance level are shown in bold. From the results shown in table, we can draw the following conclusion: For the length of feature subsets, MCFA outperforms SFS, SBS, and SFFS significantly and is comparable with SFBS; for the classification accuracies, MCFA outperforms SFS, SBS, and SFBS significantly and is comparable with SFFS.

Table 7 Average ranks of MCFA with four classic feature selection algorithms on different performances

Performances	MCFA	SFS	SBS	SFFS	SFBS
Subset length	**1.167**	5	3.667	3	2.167
Accuracy	**1**	2.5	3.5	3.167	4.833

Table 8 Wilcoxon's test between MCFA and four classic algorithms on performances over used datasets

MCFA vs	SFS	SBS	SFFS	SFBS
Feature subset length				
$R+$	21	21	21	19
$R-$	0	0	0	2
t value	0	0	0	2
p value	**0.0313**	**0.0313**	**0.0313**	0.1250
Classification accuracy				
$R+$	21	21	15	21
$R-$	0	0	0	0
t value	0	0	0	0
p value	**0.0313**	**0.0313**	0.0625	**0.0313**

The running time is also an essential performance marker of a feature selection algorithm. Table 9 lists the running time of five algorithms in six datasets.

It can be found that the running time of MCFA in datasets whose number of features is not very large is not as good as other four feature selection algorithms, but it has good advantages in the running time of the high-dimension datasets. The running time of MCFA in Handwrite is well below the other

Table 6 Experimental comparison results of five feature selection algorithms

Datasets	Number of selected features					Classification accuracy				
	MCFA	SFS	SBS	SFFS	SBFS	MCFA	SFS	SBS	SFFS	SBFS
Iris	**1.6**	3.8	2.8	3.2	2.8	**97.33**	96.67	96.67	**97.33**	96.67
Glass	**2.6**	8.8	7.8	4.6	3.4	**93.95**	93.45	93.45	92.04	93.02
Wine	2.8	11.8	10.8	7.4	**2**	**77.22**	47.76	47.76	47.88	46.05
WDBC	**3.2**	28	27	8.8	4	**92.57**	64.68	64.68	69.14	62.92
WPBC	**2.2**	29.4	28.4	8.8	4.2	**77.95**	76.30	76.30	76.30	76.30
Handwrite	**7.6**	37.2	36	9	9.6	**99.69**	98.78	98.78	98.77	98.46

FREE paper freepaper.me دانلود کننده مقالات علمی

Table 9 Experimental comparison results of five feature selection algorithms

Datasets	Running time				
	MCFA	SFS	SBS	SFFS	SBFS
Iris	0.1255	0.0109	0.0104	**0.0094**	0.0106
Glass	0.1822	0.0452	0.0503	**0.0439**	0.0545
Wine	0.1903	0.099	0.105	**0.097**	0.121
WDBC	**0.5878**	1.3163	1.3254	1.1037	1.9093
WPBC	**0.3677**	0.4662	0.4981	0.4294	0.542
Handwrite	**9.4509**	41.26	42.5	28.7376	52.1887
AVE	**1.8174**	7.1996	9.0815	5.0702	9.1377

algorithms which indicates our method will have developing prospect in the era of big data.

6.2.2 Comparison experiments with three mainstream feature selection algorithms

Multi-colony fairness algorithm is compared with other mainstream feature selection methods(RRFS, mRMR, and ReliefF) in this section. Four datasets are selected which are Wine, WDBC, WPBC, and Handwrite. In each experimental run, each normalized dataset is randomly split into a training set (2/3 of dataset's patterns) and a test set (1/3 of dataset's patterns). We use these four feature selection methods to do feature selection in the four datasets and get experimental results in 10 times of each dataset.

Similar to the comparison with four classic algorithms, we use the Friedman test and a paired t test to analyze the performance of MCFA with three mainstream algorithms, and the significance is also checked at a level of 0.05. The mean values and the standard deviations of classification accuracies are shown in Table 10. Using these results, a paired t test between MCFA and these three mainstream algorithms on used datasets is taken. And the results are listed in Table 11. The number of selected features is shown in Table 12. Table 13 depicts the average Friedman ranks on different performances of used datasets.

Table 10 Average testing accuracy of MCFA and three mainstream feature selection algorithms

Datasets		Classification accuracy (%)			
		MCFA	RRFS	mRMR	ReliefF
Wine	Acc (%)	**96.95**	96.55	80.48	91.63
	Std	2.08	1.80	2.11	2.37
WDBC	Acc (%)	**94.47**	91.85	94.24	91.75
	Std	1.06	0.88	2.32	1.20
Colon	Acc (%)	**90.00**	71.91	73.34	70.48
	Std	4.71	5.24	4.01	4.92
Arcene	Acc (%)	**78.64**	62.85	63.56	58.19
	Std	4.37	1.78	2.64	2.43

Table 11 A paired t test between MCFA and other three mainstream algorithms for used datasets on classification accuracy

MCFA versus	RRFS	mRMR	ReliefF
Wine	0.6512	8.81E−13	4.52E−05
WDBC	1.1E−05	0.7789	4.18E−05
Colon	1.98E−07	9.93E−08	3.99E−08
Arcene	3.71E−09	2.53E−08	1.5E−10

Table 12 Number of selected features of MCFA and three mainstream feature selection algorithms

Datasets	The number of selected features			
	MCFA	RRFS	mRMR	ReliefF
Wine	7	**5**	**5**	**5**
WDBC	5.8	**5**	**5**	**5**
Colon	**11.3**	50	50	50
Arcene	**18.1**	60	60	60

Table 13 Average ranks of MCFA with three mainstream feature selection algorithms on performances

Performance	MCFA	RRFS	mRMR	ReliefF
Subset length	2.5	**1**	2	3
Accuracy	**1**	2.75	2.5	3.75

Table 10 shows the listed classification accuracies of MCFA and other three mainstream feature selection methods. From it, we find MCFA can obtain better classification accuracies not only in low-dimension datasets but also in high-dimension datasets. Table 11 describes the p value of a paired t test between MCFA and other three mainstream algorithms. Except the Wine of RRFS and the WDBS of mRMR, the other results are less than 0.05. It also gives a convincing evidence against the null hypothesis. From these results, we can draw a conclusion that MCFA has excellent performance in classification accuracies in statistics.

Results in Table 12 tell us that MCFA can get a better number of selected features than the other methods in high-dimension datasets while it is not as good as other methods in low-dimension datasets. Table 13 depicts the average Friedman ranks on different performances of used datasets. The results also validate the obtained conclusion mentioned above. MCFA outperforms the three mainstream algorithms significantly on classification accuracies while the length of feature subsets does not stand out among four algorithms. Taking results in high-dimension

Table 14 Running time of MCFA and three mainstream feature selection algorithms

Datasets	The running time(s)			
	MCFA	RRFS	mRMR	ReliefF
Wine	0.617 (3)	0.004 (1)	1.187 (4)	0.079 (2)
WDBC	1.302 (2)	0.026 (1)	3.161 (4)	1.464 (3)
Colon	0.737 (2)	0.083 (1)	11.776 (4)	2.389 (3)
Arcene	4.92 (1)	13.632 (2)	15.482 (3)	264.671 (4)

datasets into account, MCFA still has a good application prospects.

Table 14 shows the running time and its rankings of the four feature selection algorithms in different datasets. These results indicate the running time of RRFS has advantages in the first three datasets. Our method's running time is not good in low-dimension datasets, but it can get better results in high-dimension datasets than other three methods.

7 Conclusion

In this paper, we transform the feature selection model to a resource allocation model and introduce the behavior of competing for resources. To reach the goal of optimizing the feature subset, we propose multi-colony fairness algorithm to deal with the resource competition behavior. The discrete problem is transformed into a continuous problem to some extent. The proposed algorithm effectively fuses strategies of the random search and the heuristic search which can improve the possibility of finding the optimal feature subset by increasing randomness with the heuristic information. In addition, MCFA combines the methods of filter and wrapper so as to reduce the amount of calculation while improving classification accuracies.

Acknowledgments This work was supported in part by the National Natural Science Foundation of China under Grant Nos. 61472139 and 61462073, the Software and Integrated Circuit Industry Development Special Funds of Shanghai Economic and Information Commission under Grant No. 140304.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

Averell L, Heathcote A (2011) The form of the forgetting curve and the fate of memories. J Math Psychol 55(1):25–35

Azar AT, Elshazly HI, Hassanien AE et al (2014) A random forest classifier for lymph diseases. Comput Methods Programs Biomed 113(2):465–473

Bouatmane S, Roula MA, Bouridane A et al (2011) Round-robin sequential forward selection algorithm for prostate cancer classification and diagnosis using multispectral imagery. Mach Vis Appl 22:865–878

Demsar J (2006) Statistical comparisons of classifiers over multiple data sets. J Mach Learn Res 7:1–30

Feng X, Yang T, Li S (2015) Network behavior-oriented CDN cache allocation strategy. Comput Sci 42:156–161

Gan JQ, Hasan BAS, Tsui CSL (2014) A filter-dominating hybrid sequential forward floating search method for feature subset selection in high-dimensional space. Int J Mach Learn Cybern 5:413–423

Garcia S, Molina D, Lozano M et al (2009) A study on the use of non-parametric tests for analyzing the evolutionary algorithms behaviour: a case study on the CEC2005 special session on real parameter optimization. J Heuristics 15:617–644

Glten A (2013) Genetic algorithm wrapped bayesian network feature selection applied to differential diagnosis of erythemato-squamous diseases. Digit Signal Proc 23(1):230–237

Guyon I (2003) An introduction to variable and feature selection. J Mach Learn Res 3:1157–1182

Han XH, Chang XM, Quan L et al (2014) Feature subset selection by gravitational search algorithm optimization. Inf Sci 281:128–146

Herzfeld DJ, Vaswani PA, Marko MK et al (2014) A memory of errors in sensorimotor learning. Science 345(6202):1349–1353

Juanying X, Weixin X (2014) Several feature selection algorithms based on the discernibility of a feature subset and support vector machines. Chin J Comput 37(8):1704–1718

Linksvayer T (2014) Evolutionary biology: Survival of the fittest group. Nature 514(7522):308–309

Mar T, Zaunseder S, Martinez JP et al (2011) Optimization of ECG classification by means of feature selection. IEEE Trans Biomed Eng 58(8):2168–2177

Mersch DP, Crespi A, Keller L (2013) Tracking individuals shows spatial fidelity is a key regulator of ant social organization. Science 340(6136):1090–1093

Moradi P, Rostami M (2015) Integration of graph clustering with ant colony optimization for feature selection. Knowl Based Syst 84:144–161

Nemati S, Basiri ME (2011) Text-independent speaker verification using ant colony optimization-based selected features. Expert Syst Appl 38(1):620–630

Parkka J, Ermes M, van Gils M (2010) Automatic feature selection and classification of physical and mental load using data from wearable sensors. IEEE, Washington

Peng H, Yinlian F, Liu J et al (2013) Optimal gene subset selection using the modified SFFS algorithm for tumor classification. Neural Comput Appl 23:1531–1538

Peter C, Jessica JK (2008) The interaction between predation and competition. Nature 456(7219):235–238

Uzer MS, Inan O, Yilmaz N (2013) A hybrid breast cancer detection system via neural network and feature selection based on sbs, sfs and pca. Neural Comput Appl 23:719–728

Vergara JR, Estevez PA (2014) A review of feature selection methods based on mutual information. Neural Comput Appl 24:175–186

Xiaofeng M, Yong L, Jianhua Z (2013) Social computing in the era of big data: opportunities and challenges. J Comput Res Dev 50(12):2483–2491

Xie J, Lei J, Xie W (2013) Two-stage hybrid feature selection algorithms for diagnosing erythemato-squamous diseases. Health Inf Sci Syst 1:1–14

FREE paper freepaper.me دانلود کننده مقالات علمی