DOI 10.15622/ia.2020.19.6.3

O. GERMAN, S. NASR

NEW METHOD FOR OPTIMAL FEATURE SET REDUCTION

German O., *Nasr S.* **New Method for Optimal Feature Set Reduction.**

Abstract A problem of searching a minimum-size feature set to use in distribution of multidimensional objects in classes, for instance with the help of classifying trees, is considered. It has an important value in developing high speed and accuracy classifying systems. A short comparative review of existing approaches is given. Formally, the problem is formulated as finding a minimum-size (minimum weighted sum) covering set of discriminating 0,1-matrix, which is used to represent capabilities of the features to distinguish between each pair of objects belonging to different classes. There is given a way to build a discriminating 0,1-matrix. On the basis of the common solving principle, called the group resolution principle, the following problems are formulated and solved: finding an exact minimum-size feature set; finding a feature set with minimum total weight among all the minimum-size feature sets (the feature weights may be defined by the known methods, e.g. the RELIEF method and its modifications); finding an optimal feature set with respect to fuzzy data and discriminating matrix elements belonging to diapason [0,1]; finding statistically optimal solution especially in the case of big data. Statistically optimal algorithm makes it possible to restrict computational time by a polynomial of the problem sizes and density of units in discriminating matrix and provides a probability of finding an exact solution close to 1.

Thus, the paper suggests a common approach to finding a minimum-size feature set with peculiarities in problem formulation, which differs it from the known approaches. The paper contains a lot of illustrations for clarification aims. Some theoretical statements given in the paper are based on the previously published works.

In the concluding part, the results of the experiments are presented, as well as the information on dimensionality reduction for the coverage problem for big datasets. Some promising directions of the outlined approach are noted, including working with incomplete and categorical data, integrating the control model into the data classification system.

Keywords: Multidimensional Data, Classification, Feature Selection, Minimum-size Covering Problem, Group Resolution Principle

1. Introduction. One of important applied problems in data mining, control and system analysis is reduction of the feature set used in a model (e.g. classification or recognition ones). This problem attracts serious attention [1-5]. There are three common groups (and their combinations) of methods to realize feature set reduction including filtering, wrapper, and embedded methods. They give different results from the viewpoint of accuracy and computational complexity.

Filtering methods [6, 7] are computationally effective but do not provide (in general) classification and prognostic accuracy of the model, because, for instance, they do not take into account (in general) the internal links between features, e.g. multigroup co-relation coefficients and dependencies.

The main idea of the filtering methods is to estimate feature ratings (weights) W and use some threshold to remove features with small

ratings. A good and widely known filtering method is RELIEF [8]. In this method, for each sample object Ob_x and each feature f_y one defines the nearest object Ob_1 from the same class (say, A) and the nearest object Ob_2 from the opposite class (say, B) for two-classes classification problem. Obviously, feature f_y differs between A and B quite well if its value for class A is clearly greater than its value for class B. This observation constitutes the idea of RELIEF method which uses iterative process to re-evaluate feature f_y weight $W[f_y]$ by adding to it the value of $diff(f_y, Ob_x, Ob_2)/m$ and subtracting the value of $diff(f_y, Ob_x, Ob_1)/m$, where $diff(...)$ stands for the distance between two objects with respect to feature f_y and m denotes the number of pairs (Ob_x, Ob_1), $((Ob_x, Ob_2))$.

Besides RELIEF and its modifications, one can mention principal component analysis [9], supporting vector machine and other feature scoring methods [6, 10].

The other group of methods is united under the title *wrapper* methods [11, 12]. They estimate quality of the feature set $\{f_1, f_2, ..., f_z\}$ for instance by learning neural network with the inputs $f_1, f_2, ..., f_z$ and providing the following assesment of the resulting classification accuracy. This technique is extensively consuming computational resources and cannot be recommended for big feature sets as it requires to consider $O(2^d)$ feature subsets to reveal the smallest one with satisfying classification capabilities (where d stands for the total number of the features).

To smooth drawbacks of the above two groups of methods the *embedded* methods were suggested [13]. A good example is C4.5/CART [14, 15] methods based on information gain calculations in clasification trees. Suppose that each of the objects belongs either to class A or to class B (but only to one of them). Next suppose that one selects attribute (feature) f_y to split all the objects accordingly to value of f_y. To simplify our considerations, admit then that $f_y \in \{0,1\}$. Now divide all the objects into two subsets: f_{y0} where $f_y = 0$, and f_{y1} where $f_y = 1$. In general, the representatives of initial classes A and B may be among samples in f_{y0} and f_{y1}. In GINI-index based method [16] (which lies in the basis of CART – *c*lassification and recognition tree) a quality of the splitting is associated with the score value $h_i = 1 - \sum_{j \in A,B} (\frac{n_j}{|f_{y_i}|})^2$ computed for each subset f_{y_i} where $n_A(n_B)$ stands

for the number of objects of class A (class B) in f_{y_i} and $|f_{y_i}|$ denotes cardinality of f_{y_i}. It is accepted that the best selection corresponds to attribute f_y minimizing average value of $h_0(h_1)$.

One can then split f_{y0} and f_{y1} in the similar way with the help of another attribute, say f_z, and so on in order to build a classifying tree with the attributes located in its nodes. Clearly, this heuristical approach delivers in general not optimal solution, that is, the set of attributes defined accordingly to the method may not be a minimum-size one. The main advantage of this embedded method is that it has good computational characteristics and results in the ready-to-use classifying tree.

However, there is no common platform in the above groups of methods regarding possible peculiarities in problem specification. These peculiarities are linked, for instance, to data fuzziness, weightedness, possible redundancy, incompleteness, quantitative nature, big sizes, *etc*.

The goal of the paper is to propose such a platform. It uses a technique to solve a 0,1–matrix covering problem on the basis of some common principle (called *g*roup *r*esolution *p*rinciple – *GRP*) applicable to the different problem specifications including (among the others):

– finding an exact minimum-size feature set;

– finding a feture set with minimum total weight among all the minimum-size feature sets;

– finding an optimal feature set with respect to fuzzy data;

– finding statistically optimal solution especially in the case of big data.

Among the advantages of this platform are also eliminating feature redundancy problem and possibility to deal with qualitative data. This may serve a good starting position for future investigations in the marked areas.

2. Discrimination Matrix. Suppose, a normalized data set is given (Table 1). There are 2 classes (A and B), 6 features $\{f_1, f_2, ..., f_6\}$ and 8 objects $\{i_1, i_2, ..., i_8\}$. Our task is to define a minimum-size feature set and build a classifying model, for instance, in the form of a classifying tree or a neural network.

Let us give a general formal statement of the problem. Denote by $D_i =< f_{i1}, f_{i2}, ..., f_{iK}, C_i >$ the ith row $(i = 1, N)$ of the data set with the corresponding feature values f_{ip} (quantitative data, $p = 1, K$), and C_i standing for the class label. Without loss of generality we shall use two classes. Denote by $Ob_i =< f_{i1}, f_{i2}, ..., f_{iK} >$ the ith object (in vector form) of the data set. Let us require the following conditions to be true:

$$\exists r \exists s (C_r \neq C_s), \quad 1 \leq r, s \leq N, \tag{1}$$

$$\forall r \forall s (C_r \neq C_s) \rightarrow Ob_r \neq Ob_s. \tag{2}$$

Table 1. The normalized data set

	f_1	f_2	f_3	f_4	f_5	f_6	Class
i_1	0.8	0.5	0.0	0.77	0.33	0.33	*A*
i_2	1.0	0.5	0.5	1.0	0.0	0.0	*A*
i_3	0.4	0.25	0.5	1.0	0.0	0.0	*B*
i_4	0.2	0.0	0.75	0.0	0.0	0.66	*B*
i_5	0.7	1.0	0.75	0.44	1.0	1.0	*A*
i_6	0.0	1.0	1.0	0.44	1.0	0.83	*B*
i_7	0.0	0.5	1.0	0.33	0.66	0.83	*A*
i_8	0.4	1.0	0.75	0.44	0.33	0.66	*B*

Let $\pi \subseteq \{1,...,K\}$ be some set of the unique integer indices and $f(\pi)$ stand for the (sub)set of the features with indices from π. Let $Ob_r(\pi)$ denote new vector obtained from Ob_r on the features $f(\pi)$. The feature set minimization problem is stated as to find π with minimum power $|\pi|$ providing:

$$\forall r \forall s (C_r \neq C_s) \rightarrow Ob_r(\pi) \neq Ob_s(\pi). \quad (3)$$

Clearly, condition (3) warrants that one can build a classifying tree (*CT*) for the data set using the features from the covering set π. We, however, omit the question about the sizes of *CT* leaving it for experiments.

Consider an arbitrary column in the normalized data Table (e.g. column f_1).

Definition 1. Feature f_t discriminates between two objects $x \in A$ and $y \in B$ if and only if $f_{xt} \neq f_{yt}$.

We shall also use another notation (i_x, f_t) instead of f_{xt}. For example, f_1 discriminates between (i_2, f_1) and (i_3, f_1) and does not discriminate between (i_6, f_1) and (i_7, f_1).

Notice 1. Discrimination between the same classes is not considered as it has no sense.

Notice 2. As features may be incorrectly defined due to differnt reasons, one may use level $\Delta > 0$ of discrimination (Δ has rather small positive value) and consider that feature f_t discriminates between two samples $x \in A$ and $y \in B$ if and only if $f_{xt} \geq (f_{yt} + \Delta)$ or $f_{xt} \leq (f_{yt} - \Delta)$.

Now one is in position to build a discrimination matrix *DM* with elements $dm_{kij} = 1$ if and only if feature f_k discriminates between samples i and j; otherwise $dm_{kij} = 0$ (take into account *Notice* 1)

This matrix is given by Table 2. The columns containing only «1s» or «0s» are deleted as redundant.

Table 2. Reduced matrix *DM*

	1,8	2,3	2,4	4,5	5,6	5,8	6,7
f_1	1	1	1	1	1	1	
f_2	1	1	1	1			1
f_3	1		1		1		
f_4	1		1	1			1
f_5				1		1	1
f_6	1		1	1	1	1	

The rows correspond to the features. Each column is represented by pair (i, j) with i and j specifying rows in Table 1. For instance, consider row f_2 and column (5,6) with «0» (empty value) at the intersection. This situation means that feature f_2 does not discriminate between rows 5 and 6 in Table 1.

Definition 2. Row a covers column b if a contains «1» in column b.

Definition 3. A set of rows $\pi = \{a_1, a_2, ..., a_r\}$ is a covering one for *DM* if for each column b from *DM* there is at least one row from π that covers b.

We are interested in a minimum-size covering set π. In our example, one of the minimum-size covering sets is $\pi = \{f_1, f_2\}$. So, instead of 6 features it is sufficient to use only 2. Basing on the found feature set, one can build a *CT* with the Python script given in Appendix.

One can prove then the next

Proposition 1. Any minimum-size cover π of the matrix *DM* defines the corresponding minimum-size feature set.

Proof. Any proper subset π^* of π ($\pi^* \subset \pi$) does not cover some column in *DM*. Let this column be (i, j) and let row i belong to class *A* and row j belong to class *B*. The values of the features from π^* are the same both in objects i and j. By this, it is impossible to uniquely define by means of features π^* to which classes i and j belong.

It is a well-known fact that finding a minimum-size covering set of 0,1–matrix is NP-complete problem. We shall consider some exact method for its solution with polynomial efficiency in average. This method uses a group resolution principle suggested and substantiated in [17, 18].

3. Group Resolution Principle (Method). Group resolution method enables one to find a minimum-size covering set of a 0,1-matrix B. It represents an iterative process, with unique covering sets π_i found at each iteration by means of some heuristic technique. The following heuristic method may be used: at each iteration q find a column (amidst those remained undeleted) with minimum number of 1s. Let this column be r_q. Call r_q *a syndromic* column. Then find a row f_q (amidst undeleted rows), covering r_q, with maximum number of units. Call the unit element («1») at the intersection of row f_q and column r_q *a syndromic* element. Include f_q into a covering set formed at the iteration q. Delete then all the rows containing 1s in the column r_q. Also delete all the columns covered by row f_q. The qth iteration continues till there remains at least one undeleted column.

For each π_i, a special (*syndromic*) matrix is being built. From that matrix, one finds a new column-resolvent and adds it to B to contract the search area. The process repeats till an empty resolvent is generated. It is warranted that sooner or later a totally zero resolvent will be generated what indicates to finishing of the *GRP*. The best solution found to this moment represents a minimum-size cover.

To explain the details of the group resolution method, let us consider an example of some *DM* (Table 3*a*, excluding columns w and res_1).

Table 3. Example of 0,1-matrix (*a*) and syndromic submatrix (*b*)

a)													*b*)			
	w	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9	c_{10}	res_1	c_1	c_2	c_4	c_{10}
f_1	4	1		1					1				1			
f_2	2				1	1	1			1					1	
f_3	5			1		1	1				1					1
f_4	6		1				1		1					1		
f_5	2	1		1				1		1			1			
f_6	3		1		1			1		1	1	1		1	1	1

Thus, select column c_1 and covering it row f_5 include into π_1. Then delete the rows and columns as explained above and get the next matrix (Table 4).

Table 4. *GRP* in action

	c_2	c_4	c_5	c_6	c_8	c_{10}
f_2		1	1	1		
f_3			1	1		1
f_4	1			1	1	
f_6	1	1				1

Now select column c_2 and row f_4. Extend current cover to $\pi_1 = \{f_5, f_4\}$. Delete the rows and columns with respect to this new selection: namely, delete columns c_2, c_6, c_8 and rows f_4, f_6.

Acting by analogy (select column c_4 first and c_{10} next), form the resulting covering set $\pi_1 = \{f_5, f_4, f_2, f_3\}$. This solution was delivered by heuristic method without warrants of optimality. The essence of this heuristic method (introduced by *A.D. Zakrewsky* and here slightly modified) is to select the columns with minimum number of units first and in those columns find a covering row with maximum number of units. It is time now to generate a logical consequence of the columns c_1, c_2, c_4, c_{10} – their *group resolvent*. To do this, let us select submatrix of the initial matrix with columns c_1, c_2, c_4, c_{10} (Table 3*b*). This submatrix is called *syndromic*. A group resolvent is a new column defined according to the following *rule* (*RSA*): it contains 1s only in the rows of syndromic matrix with two or more 1s. Add this resolvent (res_1) to initial matrix in Table 3*a*.

Now perform the second iteration ($q = 2$). Omitting the details, find $\pi_2 = \{f_6, f_1, f_3\}$ with syndromic columns res_1, c_1, c_5 correspondingly. To produce a new group resolvent, form syndromic matrix for this cover with columns res_1, c_1, c_5 and rows f_1, ..., f_6. According to *RSA*, the group resolvent is empty. The entire process terminates with the best solution found – $\pi_2 = \{f_6, f_1, f_3\}$. This solution stands for a minimum size cover, we have been looking for. Theoretical backgrounds of *GRP* can be found in [17, 18].

The computational complexity estimation of *GRP* is given in the final part of the paper and testifies to its polynomial properties in average with the

required number of iterations submitted to expression [17] $O\left(\frac{n \cdot m \cdot p}{\sqrt{1-p}}\right)$, where n (m) stands for the number of rows (columns) in original matrix *DM* and p is the density of units (i.e. the number of units divided by $n \cdot m$) (p is supposed to be not close to 0 or 1).

The evident drawback of the described method is continuous growth of the matrix sizes due to adding of new group resolvents. Now we introduce a new technique to eliminate this drawback within the frame of enhanced version of *GRP*.

4. An Enhanced Version of *GRP*. The following material is based on [19]. In the method realizing enhanced version of *GRP* new group resolvents (starting from some time point) are not added to matrix B as additional columns but overlap some previously generated resolvents. The total number of the added group resolvents does not exceed the number of rows in *B*.

Provide the following reasoning. Let a cover π_i was found at iteration i by sequantial including rows $\alpha_1, \alpha_2, ..., \alpha_k$. Suppose that new iteration $i+1$ entirely repeats the previous iteration i. This means that the same syndromic columns and the same rows $\alpha_1, \alpha_2, ..., \alpha_k$ are selected in the same order including some additional new row(s). At the moment of including row α_k into π_{i+1} (α_k is the last one in π_i) matrix B cannot be entirely destroyed, otherwise one gets $\pi_i = \pi_{i+1}$, which is impossible according to *GRP* theoretical properties [17, 18]. This means that at least one column β should remain undeleted and β is not covered by any one of rows $\alpha_1, \alpha_2, ..., \alpha_k$. But column β must be at this moment totally zero as all the rows having «1s» in β will be deleted (because the same syndromic columns are selected for rows $\alpha_1, \alpha_2, ..., \alpha_k$ at the iterations i and $i+1$). Evidently, this is impossible and enables one to come to one of the next conclusions:

– either π_{i+1} has less than k rows

or

– at one of the steps $1, 2, ..., k$ when forming cover π_{i+1} in the selected syndromic column there would be smaller amount of units in comparison to the syndromic column selected at the same step while forming cover π_i.

This decisive reasoning enables one to restrict the number of added group resolvents by only those ones which were used as syndromic columns

at the current iteration of *GRP*. Obviously, the total number of these syndromic columns cannot exceed the number of rows in B. The rest group resolvents which were not used as syndromic columns at the current iteration of *GRP* can be excluded without loss of solution. One can use any one of the excluded columns to replace it with a new group resolvent (this means that new group resolvent simply overlaps the old column without extending current matrix B sizes).

Thus, the enhanced method enables one not to exceed the memory region restricted by 2-dimension array with m rows and $N+m$ columns (where N stands for initial number of columns in B).

We now consider the weighted case of *GRP* [18]. One may be interested in finding among all minimum-size covering sets that one with maximum total weight (representing sum of the weights of rows (features) from this covering set). It is important in the case when new samples not included in original data set should be classified later. A feature weight represents in general a score evaluating its classifying capabilities. The general idea is that the greater the total weight of the selected features, the less likely it is to re-train the classification model. We introduce a specific formulation of the covering problem and solve it by means of the modified *GRP* version.

5. Weighted Case of *GRP*. Suppose, each row i of 0,1−matrix B is assigned an integer weight w_i. Formulate a problem as to find from all minimum-size covers of B a cover π^* such that:

$$\begin{gathered} \forall r \forall s (C_r \neq C_s) \to Ob_r(\pi^*) \neq Ob_s(\pi^*); \\ \exists \pi \; ((|\pi| \; < \; |\pi^*|) \; \& \; \forall r \forall s (C_r \neq C_s) \to Ob_r(\pi) \neq Ob_s(\pi)); \\ \sum_{i \in \pi^*} w_i \to \min, \end{gathered} \tag{4}$$

where C_r, C_s belong to class labels in the data set $D = \{Ob_i\}$, $i = 1, N$.

This formulation is different from a classical one which requires to find a cover π_k (not obligatory a minimum-size one) with a minimum sum of the weights of rows in π_k. Clearly, the weights w_i may initially be defined as negated RELIEF-weights. To solve a problem, we address to weigthed case of *GRP* and modify it to meet our goals. The idea behind the method is to use *GRP* as in section 3 to find a minimum-size cover π_k at the iteration k and then on the syndromic matrix, corresponding to π_k, build a new group resolvent res (in a new fashion described in [18]) such that if an optimal solution has not been found yet, it would cover res. For the details, let us

consider an example from Table 3*a* again with additional column *w* representing the row weights.

The procedure finds a minimum-size cover first. It is the same as in unweighted case of *GRP*. We previously found a minimum-size cover $\pi_2 = \{f_6, f_1, f_3\}$ with syndromic columns res_1, c_1, c_5 correspondingly (Table 5).

Table 5. Weighted syndromic matrix

	c_1	c_5	res_1	w
f_1	1			4
f_2		1		2
f_3		1		5
f_4				6
f_5	1			2
f_6			1	3

The weight of the cover π_2 is denoted as $w(\pi_2) = 4 + 5 + 3 = 12$. To form a group resolvent, we use rule *RSB* suggested in [18] with peculiarity of the syndromic matrix having no rows with two or more 1s. Namely, divide the rows into two subsets SR_π^I and SR_π^{II}. Subset SR_π^I contains those rows with no more than one unit in each of them. In our case due to observed specificity of syndromic matrix, SR_π^I coincides with the set of all rows, i.e $SR_\pi^I = \{f_1, f_2, ..., f_6\}$. Subset SR_π^{II} contains the rest rows of matrix B, not belonging to SR_π^I. In our case $SR_\pi^{II} = \varnothing$.

Now define the low boudary $LB(\pi_2)$ by means of the cover $\pi_2 = \{f_6, f_1, f_3\}$ and its syndromic matrix. Denote set of columns of syndromic matrix by $\Omega(\pi_2)$. Let us for each column z from $\Omega(\pi_2)$ denote by $\rho(z)$ the row in SR_π^I with minimum weight which covers column z.

Then

$$LB(\pi_2) = \sum\nolimits_{z \in \Omega(\pi_2)} w(\rho_z). \quad (5)$$

In the example, $LB(\pi_2) = 2 + 2 + 3 = 7$. Now let us compare $LB(\pi_2) = 7$ and the weight of the cover π_2 $w(\pi_2) = 12$.

Proposition 2.

1. If with respect to the current cover π_i $w(\pi_i) \le LB(\pi_i)$ then in supposition that π_i is not optimal solution, each optimal solution contains at least one row from SR_π^{II}. Consequently, for this case, if $SR_\pi^{II} = \varnothing$ then π_i is an optimal solution.

2. If $w(\pi_i) > LB(\pi_i)$ then one needs to move minimum number of rows from SR_π^{I} (no one should belong to current cover π_i) into SR_π^{II} to provide.

– $w(\pi_i) \le LB(\pi_i)$;

– conditions for *RSB* to generate a group resolvent as a column with the units standing only in the rows belonging to SR_π^{II}.

The easiest way to provide 2*b* is as follows: for each column c_x in Ω find the corresponding value $\Delta_x = w_{\max}(x) - w_{\min}(x)$, where $w_{\max}(x)$ ($w_{\min}(x)$) is the maximum (minimum) value of the weight of some row covering column c_x. If there is only one row α covering c_x or $\Delta_x = 0$ then this column is ignored. In the example (Table 5), one has $\Delta_{c_1} = 4 - 2 = 2$, $\Delta_{c_5} = 5 - 2 = 3$. Find maximum value among Δ_x. In the example, this is Δ_{c_5}. Consequently, one needs to transfer row f_2 (covering c_5 and having minimum weight value) from SR_π^{I} to SR_π^{II}: $SR_\pi^{I} = \{f_1, f_3, ..., f_6\}$, $SR_\pi^{II} = \{f_2\}$. This transfer leads to $LB(\pi_2) = 2 + 3 + 5 = 10 < w(\pi_2) = 12$. It is necessary to make one more transfer. There remains only one possibility: to transfer row f_5 (covering c_1 and having minimum weight value) to SR_π^{II}: $SR_\pi^{I} = \{f_1, f_3, f_4, f_6\}$, $SR_\pi^{II} = \{f_2, f_5\}$.

Now $LB(\pi_2) = 4 + 3 + 5 = 12 = w(\pi_2) = 12$ and one can build a group resolvent res_2 with the units only in the rows f_2, f_5. Add this resolvent to original matrix in Table 3*a* and resume searching a minimum-size cover.

Point 2 of the *Proposition* 2 can be reformulated in a stronger form. Namely, let w^* denote the minimum weight of the best minimum-size covering set π^* found at the previous iterations and $w(\pi_i)$ as earlier stands for the weight of the current minimum-size cover. Then the strengthened form of proposition 2 is as follows:

Proposition 2 (strengthened form) [18].

1. If with respect to the best cover π^* and the current cover π_i $w^* \le LB(\pi_i)$ then in supposition that π^* is not optimal solution, each optimal solution contains at least one row belonging to SR_π^{II}. Consequently, for this case, if $SR_\pi^{II} = \varnothing$ then π^* is an optimal solution.

2. If $w^* > LB(\pi_i)$ then one needs to move minimum number of rows from SR_π^I (no one should belong to current cover π_i) into SR_π^{II} to provide.

– $w^* \le LB(\pi_i)$;

– conditions for *RSB* to generate a group resolvent as a column with the units standing only in the rows contained in SR_π^{II}.

Proposition 3. New resolvents generated by the rules of proposition 2 may be overlapped in the iterations of *GRP* as in the unweighted case of *GRP* (that is, in the case they are not used as syndromic columns at some future iteration(s)).

Proof. From syndromic matrix one has got a new column – group resolvent res_x. It excludes current minimum size cover π_x. Consider next iteration $x+1$. Again, as previously,

– either at one of the steps $1, 2, ..., k$ when forming next cover π_{x+1} in the selected syndromic column there would be smaller amount of units in comparison to the syndromic column selected at the same step while forming cover π_x

or

– π_{x+1} has less than k rows.

The last is impossible since π_x is a minimum size cover. Therefore, there remains the first possibility directing the computations in a new way previously not passed. This notice remains valid with respect to any new group resolvent with no relation to the previously added group resolvents provided that they were not used as syndromic columns at the current iteration of *GRP*. This observation also provides finiteness of the searching process.

Now we have all necessary rules to lead the searching process to its finish. Omitting the details, the next cover is $\{f_6, f_5, f_4, f_2\}$ with syndromic matrix shown in Table 6*a*.

Add new resolvent res_3 (Table 6*a*) to original matrix (instead of res_2, since res_2 was not used as syndromic column) and resume a search.

Perform new iteration of *GRP*. Now one finds $\pi_{fin} = \{f_1, f_6, f_2\}$ with total weight $w(\pi_{fin}) = 9$ and syndromic matrix given in Table 6*b*. For this syndromic matrix one comes to a conclusion about optimality of π_{fin} (as its group resolvent is empty and π_{fin} has minimum total weight among the previously found minimum-size covering sets).

Table 6. Syndromic matrix for $\{f_6, f_5, f_4, f_2\}$ with resolvent res_3 (*a*) and syndromic matrix for $\pi_{fin} = \{f_1, f_6, f_2\}$ with empty resolvent (*b*)

a)							*b)*			
	c_1	c_5	c_8	res_1	res_3		c_2	c_5	res_3	w
f_1	1		1		1				1	4
f_2		1						1		2
f_3		1						1		5
f_4			1				1			6
f_5	1									2
f_6				1			1			3

6. Fuzzy Case of *GRP*. Now suppose that some samples in original data set are characterized with fuzzy measure $\mu_A(\mu_B)$ of belonging to class *A* (*B*). This supposition leads to a discrimination matrix with elements from diapason [0, 1]. Indeed, if some feature f_x discriminates between row *r* and row *q* (standing for the objects i_r, i_q from different classes, e.g. *A* and *B* respectively) then at the intersection of row f_x and column (r, q) of *DM* one places the value of $\mu(f_x, r, q) = \mu_A(i_r) \cdot \mu_B(i_q)$. The covering problem now should be reformulated as described below.

Definition 4. Let row a contain $\mu(a, b) > 0$ in column *b*. Then they say that row a covers column b in a fuzzy mode with fuzzy measure $\mu_A(a, b)$.

Definition 5. A set of rows $\pi = \{a_1, a_2, ..., a_r\}$ is a covering fuzzy set for *DM* if for each column b from *DM* there is at least one row from π that covers b in a fuzzy mode.

Let $\pi = \{a_1, a_2, ..., a_r\}$ be a fuzzy covering set. Define for each column j of *DM* the value $\nu_j(\pi) = \max(\mu(f_1, j), \mu(f_2, j), \ldots, \mu(f_W, j))$ where W stands for the number of columns in original *DM*.

Definition 6. An optimal minimum-size fuzzy covering set π^* is defined as that one having 1) minimum number of rows among all fuzzy covering sets for *DM* and 2) providing maximum value of $\Psi_{\pi^*} = \sum_{j=1,W} \nu_j(\pi^*)$.

Formally,

$$\begin{gathered} \forall r \forall s (C_r \neq C_s) \to Ob_r(\pi^*) \neq Ob_s(\pi^*); \\ \exists \pi \ ((|\pi| \ < \ |\pi^*|) \ \& \ \forall r \forall s (C_r \neq C_s) \to Ob_r(\pi) \neq Ob_s(\pi)); \\ \Psi_{\pi^*} = \sum_{j=1,W} \nu_j(\pi^*) \to \max, \end{gathered} \tag{6}$$

where C_r, C_s belong to class labels in the data set $D = \{Ob_i\}$, $i = 1, N$.

Notice 3. It is important to emphasize that group resolvents added at the iterations of algorithm with fuzzy matrix *DM* do not influence upon the value of Ψ_π.

To find an optimal minimum-size fuzzy covering set, one can use a slight modification of *GRP* for the weighted case considered above practically basing on the same ideas. Again, let us take a matrix (Table 7) as an example to illustrate the details.

Table 7. Fuzzy matrix *DM*

	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9	c_{10}
f_1	0.6		1					0.8		
f_2				0.9	0.9	1			1	
f_3			0.8		0.7	0.5				0.7
f_4		1				0.5		1		
f_5	0.8		0.8				1		1	
f_6		1		1			1		1	0.9

First, one is looking for some minimum-size cover (with no regards to its weight Ψ_π). This process has a peculiarity when making group resolvents only. One is acting as though fuzzy elements are crisp-valued (as in *GRP*). Namely, find column with minimum number of non-zero elements (it is a syndromic column). Then define a row with maximum number of non-zero elements which covers this column in a fuzzy mode. Then reduce the matrix according to the *GRP* rules and continue till the new covering set is defined.

The first covering set $\pi_1 = \{f_5, f_4, f_2, f_3\}$ with the syndromic columns c_1, c_2, c_4, c_{10}. Find their group resolvent (Table 8*a*).

Definition 7. A fuzzy group resolvent is defined as that one, containing units in the rows with two or more non-zero elements in syndromic matrix rows.

Thus, in Table 8*a* the fuzzy resolvent contains the only unit in row f_6.

This phase finishes with an empty resolvent found on the last syndromic matrix. Now, a new group resolvent should be generated to provide condition 2) in *Definition* 6. For clearness, consider the covering set $\pi_2 = \{f_6, f_1, f_3\}$ with fuzzy syndromic columns res_1, c_1, c_5 (Table 8*b*).

Let us formulate the rule *RSC* for the fuzzy case of *GRP*.

1. If in the syndromic matrix for the last minimum-size cover there are no fuzzy elements (different from 0 and 1) then optimal solution is found corresponding to the best solution found in the previous iterations including the last one. Otherwise.

2. In each column of the syndromic matrix mark (with «*») all non-zero elements with the values higher than the value of the syndromic element in this column (see Table 9 as an example). If there are no marked elements then algorithm finishes with optimal solution, corresponding to the best solution found in the previous iterations including the last one. Otherwise.

3. Form a new group resolvent as a column containing units in the rows with marked elements and zeroes in the rest rows.

Table 8. Fuzzy syndromic submatrix *DM* with fuzzy group resolvent res_1 (*a*); Fuzzy syndromic matrix for $\pi_2 = \{f_6, f_1, f_3\}$ (*b*)

a)							*b)*		
	c_1	c_2	c_4	c_{10}	res_1		c_1	c_5	res_1
f_1	0,6						0,6		
f_2			0,9					0,9	
f_3				0,7				0,7	
f_4		1							
f_5	0,8						0,8		
f_6		1	1	0,9	1				1

After forming a fuzzy group resolvent add it to the current matrix *DM* and resume iterations.

Again, the column-resolvents may overlap the previously built group resolvents provided they were not used in syndromic matrix. Therefore, the finiteness of entire process is based on the following fact.

Each time when iterations resume after adding new group resolvent, the computations are being performed in a new direction (see proof of *Proposition* 3) due to the following decisive point: at one of the steps $1,2,...,\ k$ when forming next cover π_{x+1} in the selected syndromic column, there would be smaller amount of units in comparison to the syndromic column selected at the same step while forming cover π_x.

Table 9. Fuzzy resolvent (rule *RSC*)

	c_1	c_5	res_1	*fuzzy resolvent*
f_1	0,6			
f_2		0,9*		1
f_3		0,7		
f_4				
f_5	0,8*			1
f_6			1	

7. Approximate Covering Procedure. One can restrict the *GRP* iterations before getting empty group resolvent as was shown in [17, 18]. Let us reproduce some estimations of complexity of approximate covering procedure. Denote by n (m) the number of rows (columns) in *DM*; let p stand for the density of units, that is, p is equal to the total number of units in *DM* divided by $(n \cdot m)$. Let k stand for the covering set size. Accordingly to [17, 18], the number of iterations required in order that mathematical expectation (M_k) of the number of covers with k rows becomes $M_k < 1$ can be defined from the condition:

$$(n+0.5)\cdot \ln n-(k+0.5)\cdot \ln k-(n-k+0.5)\cdot \ln(n-k)+\frac{1}{12}\cdot n-\frac{1}{12k+1}- \\ -\frac{1}{12(n-k)+1}-m\cdot \ln(1-\varepsilon_k)\le -1.5, \quad (7)$$

where

$$\varepsilon_k=(1-\frac{p\cdot n}{n})\times(1-\frac{p\cdot n}{n-1})\times...\times(1-\frac{p\cdot n}{n-k+1}). \quad (8)$$

These relations are obtained in supposition that in «average case» the density of units in the group resolvents is approximately the same as p what is confirmed for quite a big amount of experimental data (excluding extreme cases with very low density p or its closeness to 1).

Formulas (7), (8) enable one to stop iterations before getting an empty resolvent. The estimation of the number I of iterations is of the form:

$$I = O\left(\frac{m \cdot p \cdot (n - \ln(\frac{z^2+2}{2} - \sqrt{(\frac{z^2+2}{2})^2 - 1}))}{\sqrt{1-p}} \right), \qquad (9)$$

which can be simplified to

$$I = O\left(\frac{m \cdot p \cdot (n + 2.41)}{\sqrt{1-p}} \right). \qquad (10)$$

Here, z can be selected from the well-known rule of «$z\sigma$» (e.g. $z = 3$ or higher). The rule of «3σ» means that a value of a normally distributed random variable x with mean x_{mean} falls in the diapason $[x_{mean} - 3 \cdot \sigma, x_{mean} + 3 \cdot \sigma]$ with a probability close to 0.997.

So, in average the approximate method behaves itself like a polynomial computational method for a given density p of units (see the concomitant considerations in [17, 18]).

Now consider the last question: how to restrcit the original number of samples in the learning data set.

8. Restriction of Data Set Sizes. For multi-dimension data one can use ε – nets (see for instance [20]) with nodes covering data samples in the following way: for each multi-dimension data object there exists one and only one node in ε – net the (Euclidean or other) distance to it does not exceed ε. Building ε – net is again a minimum-size covering problem. So, in order to simplify computational expences one can use *K*-nearest neighbors method (see papers review [21]) to build $K > 0$ clusters such that each data object gets directly to one and only one cluster. We do not restrict this formulation by the condition that K should have a minimum value. Then one can use the cluster centroids instead of data objects from original data set [22]. This gives us a solution to reduction of the sizes of discrimination matrix.

Another possibility is connected to use random sampling technique developed in applied statistics. This approach, and the previously mentioned, need special attention.

9. Experiments. For estimation of the described approach on the basis of *GRP* and minimum-size matrix covering technique we used *DecisionTreeClassifier* (*DTC*) and *RFECV* (*r*ecursive *f*eature *e*limination with *c*ross-*v*alidation) methods provided by Python programming language. Comparative results with *DecisionTreeClassifier* are placed in Table 10, with *RFECV* – in Table 11. The original data sets contained two classes with randomly generated binary vectors and unit density randomly chosen from diapason [0.2, 0.5]. The first column in Tables 10, 11 indicates to original amounts of features and samples. The second and the third columns define the resulting amounts of features found by the corresponding method. The order of the numbers (experimental results) in the second column corresponds to the comparative results of the same experiments in the third column for each row of the Table.

The calculation time of each experiment with *GRP* was in the worst case three times longer compared to the *DTC* method but did not exceed 10 seconds on IBM Pentium 2.1GHz.

Table 10. Comparative results with *DecisionTreeClassifier*

Features, samples (original Data set)	Feature set sizes found with *GRP*	Feature set sizes found by *DTC*
15, 100	7, 9, 14	14, 14, 15
20, 100	9, 8, 9, 7, 12, 7	16, 18, 18, 14, 17, 15
30, 100	9, 9	20, 23
40,100	8, 9, 12	17, 19, 20
50, 100	9, 10, 8	21, 20, 19

Table 11. Comparative results with *RFECV*

Features, samples (original Data set)	Feature set sizes found with *GRP*	Feature set sizes found by *RFECV*
20, 100	17, 14, 10, 9	18, 16, 12, 13
30, 100	10, 11, 13	13, 14, 14
40,100	12, 12, 9, 9	14, 15, 10, 12
50, 100	12, 7, 16	14, 8, 19

One can see that *GRP* provides stable superiority over Python techniques with practically acceptable computation time. Developing the ideas of the section 8 of this paper, we also performed experiments with big data sets containing 300 multidimensional binary objects (vectors) what exceeds the predefined limitations on our program for *GRP*-based solution technique. In experiments, two classes of objects were generated

with different mathematical expectation and standard deviation. The best results were obtained for 20-30 clusters with classification accuracy near 90%. However, increasing the number of clusters did not improve accuracy of classification and even worsened it. This problem remains open for further investigations.

A series of 30 experiments was conducted to find out relations between original features amount (OFA\{20, 30, 40, 50, 60\}), number of objects (N\{80, 100, 120, 130, 140, 150, 160\}), minimum feature amounts (*FAGRP*) found by *GRP*-based covering technique and number of nodes in the classifying tree (*Nnod*) created for *FAGRP* and *N*. There were generated two classes of objects with predefined probabilities of units for 0,1-valued features. The following conclusions can be made:

1. $FAGRP = O(k \cdot OFA^{0.5})$ with a constatnt k in [0.8-3.0] (in majority of cases k is near to 1.5).

2. *Nnod* depends on *FAGRP* (*OFA*) in unstable mode within given 0,1–distribution of the feature values and fixed N (Table 12*a*).

3. *Nnod* has no clear tendency to growth with increasing N and fixed *OFA* as is illustrated by Table 12*b*.

Table 12. Results of experiments (N = 100) (*a*); Results of experiments for different $N \in [100, 160]$, OFA =40 (*b*)

a)				*b)*		
Features, samples (original Data set)	*FAGRP*	*Nnod*		Features, samples (original Data set)	*FAGRP*	*Nnod*
20, 100	7	41		40, 100	6	41
30, 100	7	55		40, 120	9	115
40, 100	6	35		40, 140	8	83
50, 100	9	39		40, 150	9	63
60, 100	23	87		40, 160	8	67

As the last example, consider feature set reduction in image recognition. An original image represents digit «4» placed within a square divided by cells (Fig. 1). Each cell in Figure 1 stands for the feature f_i. Initially, there are 64 features. Each feature is either zero (no part of the digit «4» is within the corresponding cell) or unit (the cell contains some part of the digit «4»).

One can randomly distort the image by clearing some cells with parts of the digit or painting empty cells like in Figure 2. In experiments, we

randomly obtained 50 distorted images of the digit «4» with the slight modifications (cluster *A*). Also there were generated 50 random samples with chaotic distribution of the empty and colored cells (cluster *B*). The *GRP*-based method left 23 features accordingly to minimum-size cover of the discriminating matrice. This result was obtained practically instantly for a single *GRP*-iteration despite the big sizes of the discriminating matrix (64 rows and 2500 columns).

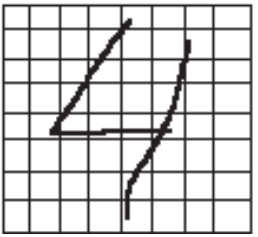

Fig. 1. Recognition of the digit «4»

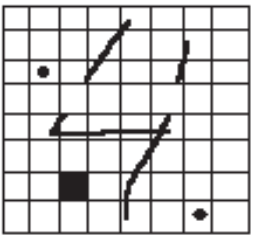

Fig. 2. Distorted image

10. Conclusion. The total approach outlined here competes well with the known methods and gives better solutions in majority of cases especially with big initial amounts of the features. It makes possible to operate with discrimination matrices with some hundreds of features (this amounts to 300 in our program). To extend the practical boundaries of the realized technique it was suggested to use clusterization of the input data sets which showed promising results, although they are needed in future investigations. The *GRP*-based approach may serve a common platform for different feature selection models and can be extended in the following directions: processing incomplete (impure) data, processing qualitative data, integrating control models in classification process, modeling practical systems in different areas *etc*.

Appendix. By means of Python programming language let us build a classifying regression tree (CRT) on the reduced feature set with two features f_1, f_2. One can use the next Python script (List. 1) which can be applied in general case for $n > 2$ features.

```
import numpy as np
from sklearn.tree import DecisionTreeRegressor
import matplotlib.pyplot as plt
# Create datasets
X = np.array([[0.8, 0.5], [1.0, 0.5], [0.4, 0.25], [0.2,
0.0],[0.7,1.0],[0.0,1.0],[0.0,0.5],[0.4,1.0]])
X_test = np.array([[0.94, 0.5],[0.85,0.3],[0.3,0.3]])
Y = np.array([1.0,1.0,0.0,0.0, 1.0, 0.0, 1.0, 0.0])
# Fit model
regr_1 = DecisionTreeRegressor(max_depth=2)
#regr_2 = DecisionTreeRegressor(max_depth=5)
regr_1.fit(X, Y)
# Predict
y_1 = regr_1.predict(X_test)
print y_1
```

Listing. 1. Python code to build classification tree

The columns f_1, f_2 in Table 1 are represented as array X (features) and Y (classes). The regression tree is created in operator:

regr_1 = DecisionTreeRegressor(max_depth=2).

In y_1 = regr_1.predict(X_test) one verifies how the model predicts test values (defined in array X_test).

This script provides the output in the form of array [1, 1, 0] with the first two 1s defining class *A* and last 0 defining class *B* for the two-features inputs [0.94, 0.5], [0.85, 0.3], [0.3, 0.3] respectively.

References

1. Shah S.A., Shabbir H.M., Rehman S., Waqas M. A comparative study of feature selection approaches: 2016–2020. *International journal of scientific and engineering research*. 2020. vol. 11. no. 2. pp. 469–478.
2. Khun K, Johnson K. Feature engineering and selection. A practical approach for predictive models. CRC Press. 2019. 310 p.
3. Bachu V., Anuradha J. A review of feature selection and its methods. *Cybernetics and information technologies (Bulgary)*. 2019. vol. 19. no. 1. pp. 3–22.
4. Hameed S., Petinrin O., Hashi A., Saeed F. Filter-wrapper combination and embedded feature selection for gene expression data. *International journal of advances in soft computing and its applications*. 2018. vol. 10. no. 1. pp. 91–105.
5. Sanchez-Pinto L.N., Venable L.R., Fahrenbach J., Churpek M. Comparison of variable selection methods for clinical predictive modeling. *International journal of medical informatics*. 2018. vol. 116. pp. 10–17.
6. Li J. et al. Feature selection: A data perspective. *ACM Computer surveys*. 2017. vol. 50. no. 6. pp. 94:2–94:46.

7. Guyon I. et al. Feature Extraction. Foundations and Applicationsbg. Springer. 2006. 762 p.
8. Urbanowicz R.J. et al. Relief-based feature selection: Introduction and review. *Journal of biomedical informatics.* 2018. vol. 8. no. 5. pp. 189–203.
9. Liu Y., Singleton A., Arribas-Bel D. A principal component analysis (PCA)-based framework for automated variable selection in geodemographic classification. *GEO-Spatial Information Science.* 2019. vol. 22. no. 4. pp. 251–264.
10. Khanna R., Awad M. Efficient learning machines: Theories, Concepts, and Applications for engineers and system designers. Apress. 2015. 247p.
11. Mao Y., Yang Y. A wrapper feature subset selection based on randomized search and multilayer structure. *BioMed Research International*. 2019. vol. 2019. pp. 1–9.
12. Hui K.H. et al. An improved wrapper-based feature selection method for machinery fault diagnosis. *PloS ONE.* 2017. vol. 12. no. 12. pp. 1–10.
13. Lal T.N., Chapelle O., Weston J., Eliseeff A. Embedded methods. Series in Fuzzy and soft computing. 2006. vol. 207. pp. 137–165.
14. Sudrajat R., Irianingsih I., Krisnawan D. Analysis of data mining classification by comparison of C4.5 and ID algorithms. IOP Conference Series: Materials and Engineering. 2017. vol. 166. pp.012031.
15. Krishna M. et al. Predicting student performance using classification and regression trees. *International Journal of Innovative Technology and Exploring Engineering*. 2020. vol. 9. no. 3. pp. 3349–3356.
16. Suneetha N., Hari Ch., Sunilkumar V. Modified Gini index classification: a case statudy on hart disease dataset. *International journal on computer science and engineering.* 2010. vol. 2. no. 6. pp. 1959–1965.
17. German O.V., Naidenko V.G. [Statistically optimal algorithm for the minimum-size covering problem] *Jekonomika i matematicheskie metody – Economics and mathematical methods*. Moscow. 1993. Issue 29. vol. 4. pp. 662–667. (In Russ.).
18. German O.V. [The generalized statistically optimal method to find minimum weighted covering set for 0,1-matrix] *Jekonomika i matematicheskie metody – Economics and mathematical methods.* oscow. 1994. Issue 30. vol. 4. pp. 139–150. (In Russ.).
19. German O.V. *Jekspertnye sistemy* [Expert systems]. Minsk. Belorusskij gos. universitet informatiki i radioelektroniki. 2008. 91 p. (In Russ.).
20. Kamenev G.K., Kamenev I.G. Primenenie metodov mnogomernogo analiza dlja izuchenija sociologicheskih sovokupnostej [Applications of the methods of multidimension analysis for learning social aggregates]. Proc. of the department of mathematical modeling of economic systems. Computer Center «Informatics and Control» of the Russian Academy of Sciences. 2017. 91p. (In Russ.).
21. Bhatia N. et al. Survey of Nearest Neighbor Techniques. *International Journal of Computer Science and Information Security.* 2010. vol. 8. no. 2. pp. 302–304.
22. Sun L., Chen G., Xiong H., Guo C. Cluster analysis in data-driven management and decisions. *Journal of Management Science and Engineering*. 2017. vol. 2. no. 4. pp. 227–251.

German Oleg – Ph.D., Associate Professor, Department of Information Technologies in Automatized Systems, Belarusian State University of Informatics and Radioelectronics (BSUIR). Research interests: applied logic, informatics, cybernetcs. The number of publications – 140. ovgerman@tut.by; 6, Petrusya Brovki str., 220600, Minsk, Belarus; office phone: +375 17 2938823; fax: +375 17 2702033.

Nasrh Sara — Postgraduate Student, Department of Information Technologies in Automatized Systems, Belarusian State University of Informatics and Radioelectronics (BSUIR). Research interests: informatics, decision making. The number of publications – 10. sara.nasrh@gmail.com; 6, Petrusya Brovki str., 220600, Minsk, Belarus; office phone: +961 3997163; fax: +375 17 2702033.

УДК 519.7 **DOI** 10.15622/ia.2020.19.6.3

О.В. Герман, С.Н. Наср

НОВЫЙ МЕТОД ОПТИМАЛЬНОГО СОКРАЩЕНИЯ МНОЖЕСТВА ПРИЗНАКОВ

Герман О.В., Наср С.Н. **Новый метод оптимального сокращения множества признаков.**

Аннотация. Рассматривается задача нахождения минимального по размеру множества атрибутов, используемых для распределения многомерных объектов по классам, например на основе деревьев решений. Задача имеет важное значение при разработке высокопроизводительных и точных классифицирующих систем. Приведен краткий сравнительный обзор известных методов. Задача сформулирована как отыскание минимального (взвешенного) покрытия на различающей 0,1-матрице, которая служит для описания возможности атрибутов разделять пары объектов из разных классов. Приведено описание способа построения различающей матрицы. Сформулированы и решены на основе общего разрешающего принципа групповых резолюций следующие варианты задачи: отыскание минимального по размеру множества атрибутов на заданном входном наборе данных; отыскание минимального по размеру множества атрибутов с минимальным суммарным весом атрибутов (в качестве весов атрибутов можно использовать величины, определяемые на основе известных алгоритмов, например на основе метода RELIEF); нахождение оптимального взвешенного нечеткого покрытия для случая, когда элементы различающей матрицы принимают значения в диапазоне [0,1]; определение статистически оптимального покрытия различающей матрицы (например, для входных наборов данных больших размеров). Статистически оптимальный алгоритм позволяет ограничить время решения полиномом от размеров задачи и плотности единичных элементов в различающей матрице и при этом обеспечить близкую к единице вероятность отыскания точного решения.

Таким образом, предлагается общий подход к определению минимального по размеру множества атрибутов, учитывающий различные особенности в постановке задачи, что отличает данный подход от известных. Изложение содержит многочисленные иллюстрации с целью придать ему максимальную ясность. Ряд теоретических положений, приводимых в статье, основывается на ранее опубликованных результатах. В заключительной части представлены результаты экспериментов, а также сведения о сокращении размерности задачи о покрытии для больших массивов данных. Отмечаются некоторые перспективные направления изложенного подхода, включая работу с неполными и качественными данными, интегрировании управляющей модели в систему классификации данных.

Ключевые слова: многомерные данные, классификация, минимизация размера множества атрибутов, задача о минимальном покрытии, принцип групповых резолюций

Литература

1. *Shah S.A.*, *Shabbir H.M.*, *Rehman* S., *Waqas M.* A comparative study of feature selection approaches: 2016–2020 // International journal of scientific and engineering research. 2020. vol. 11. no. 2. pp. 469–478.
2. *Khun K*, *Johnson K*. Feature engineering and selection. A practical approach for predictive models // CRC Press. 2019. 310 p.
3. *Bachu V*., *Anuradha J*. A review of feature selection and its methods // Cybernetics and information technologies (Bulgary). 2019. vol. 19. no. 1. pp. 3–22.
4. *Hameed S*., *Petinrin O*., *Hashi A*., *Saeed F*. Filter-wrapper combination and embedded feature selection for gene expression data // International journal of advances in soft computing and its applications. 2018. vol. 10. no. 1. pp. 91–105.
5. *Sanchez-Pinto L.N*., *Venable L.R*., *Fahrenbach J*., *Churpek M*. Comparison of variable selection methods for clinical predictive modeling // International journal of medical informatics. 2018. vol. 116. pp. 10–17.

6. *Li J. et al.* Feature selection: A data perspective // ACM Computer surveys. 2017. vol. 50. no. 6. pp. 1–45.
7. *Guyon I. et al.* Feature Extraction. Foundations and Applicationsbg // Springer. 2006. 762 p.
8. *Urbanowicz R.J. et al.* Relief-based feature selection: Introduction and review // Journal of biomedical informatics. 2018. vol. 8. no. 5. pp. 189–203.
9. *Liu Y.*, *Singleton A.*, *Arribas-Bel D.* A principal component analysis (PCA)-based framework for automated variable selection in geodemographic classification // GEO-Spatial Information Science. 2019. vol. 22. no. 4. pp. 251–264.
10. *Khanna R.*, *Awad M.* Efficient learning machines: Theories, Concepts, and Applications for engineers and system designers // Apress. 2015. 247p.
11. *Mao Y.*, *Yang Y.* A wrapper feature subset selection based on randomized search and multilayer structure // BioMed Research International. 2019. vol. 2019. pp. 1–9.
12. *Hui K.H. et al.* An improved wrapper-based feature selection method for machinery fault diagnosis // PloS ONE. 2017. vol. 12. no. 12. pp. 1–10.
13. *Lal T.N.*, *Chapelle O.*, *Weston J.*, *Eliseeff A.* Embedded methods // Series in Fuzzy and soft computing. 2006. vol. 207. pp. 137–165.
14. *Sudrajat R.*, *Irianingsih I.*, *Krisnawan D.* Analysis of data mining classification by comparison of C4.5 and ID algorithms // IOP Conference Series: Materials and Engineering. 2017. vol. 166. pp.012031.
15. *Krishna M. et al.* Predicting student performance using classification and regression trees // International Journal of Innovative Technology and Exploring Engineering. 2020. vol. 9. no. 3. pp. 3349–3356.
16. *Suneetha N.*, *Hari Ch.*, *Sunilkumar* V. Modified Gini index classification: a case statudy on hart disease dataset // International journal on computer science and engineering. 2010. vol. 2. no. 6. pp. 1959–1965.
17. *Герман О.В.*, *Найденко В. Г.* Статистически оптимальный алгоритм для задачи о минимальном покрытии // Экономика и математические методы. 1993. Т. 29. № 4. С. 662–667.
18. *Герман О.В.* Обобщенный статистически оптимальный метод решения задачи о минимальном взвешенном покрытии 0,1-матрицы // Экономика и математические методы. 1994. Т. 30. № 4. С. 139–150.
19. *Герман О.В.* Экспертные системы // Минск. Белорусский гос. университет информатики и радиоэлектроники. 2008. 91с.
20. *Каменев Г.К.*, *Каменев И.Г.* Применение методов многомерного анализа для изучения социологических совокупностей // М. Труды отдела математического моделирования экономических систем ВЦ ФИЦ ИУ РАН. 2017. 91р.
21. *Bhatia N. et al.* Survey of Nearest Neighbor Techniques // International Journal of Computer Science and Information Security. 2010. vol. 8. no. 2. pp. 302–304.
22. *Sun L.*, *Chen G.*, *Xiong H.*, *Guo C.* Cluster analysis in data-driven management and decisions // Journal of Management Science and Engineering. 2017. vol. 2. no. 4. pp. 227–251.

Герман Олег Витольдович – канд. техн. наук, доцент, кафедра информационных технологий автоматизированных систем, Белорусский государственный университет информатики и радиоэлектроники (БГУИР). Область научных интересов: прикладная логика, информатика и кибернетика. Число научных публикаций – 140. ovgerman@tut.by; ул. Петруся Бровки, 6, 220600, Минск, Беларусь; р.т.: +375 17 2938823; факс: +375 17 2702033.

Наср Сара Набих – аспирантка, кафедра информационных технологий автоматизированных систем, Белорусский государственный университет информатики и радиоэлектроники (БГУИР). Область научных интересов: информатика, принятие решений. Число научных публикаций – 10. sara.nasrh@gmail.com; ул. Петруся Бровки, 6, 220600, Минск, Беларусь; р.т.: +961 3997163; факс: +375 17 2702033.