DIMENSIONALITY REDUCTION AND EXTREMAL PARAMETER GROUPING FOR WEIGHTED GRAPH PROBLEMS

Vostrov G., Khrinenko A.

This work presents graph cut approach to dimensionality reduction and shows connections between the problem of extremal parameter grouping and the problem of maximization of positive definite quadratic form on the set of vertices of k-dimensional hypercube of side 1 by modulo. It is shown that the last problem is reduced to the minimum negative cut search for a weighted graph. A heuristic algorithm for solving this problem is given.

Keywords. Datasets, dimensionality reduction, graphs.

INTRODUCTION

In the process of studying objects and events, it becomes necessary to analyze and interpret a priori information obtained through an experimental research. The main form of presenting such information is a matrix, where columns correspond to various parameters, properties, etc. Matrix rows correspond to individual objects, which are described by one set of parameters. Advance in science has leaded to the development of new techniques for data analysis that are generating enormous amounts of data, which have implied growth of many databases and also aggravated visual representation of this data in a graph form, that could help to understand underlying nature and structure of the data. Since a vertex-edge diagram is the most popular method for graph visualization wide range of methods have been developed to lay out a vertex-edge diagrams [1]. Layout selection influences individual's understanding of the graph, thus it is important to find satisfactory method that can effectively depict the structure of the graph. Since real-world problems imply large-scale datasets it is not practical to compute different layouts and compare results for further usage. For a graph with million vertices it can take hours or days to calculate single representation and since there is no consensus on which layout is more preferable in general each dataset needs to be presented in its own way. Figure 1 shows that a graph in different layouts provides completely different understanding of dataset structure. One of the central problems in analysis of such information is selection of the most significant parameters in a given set or construction of generalized factors that sufficiently reflect the informational properties of the original description and allow a clear meaningful interpretation and representation in graph form. Thus, dimensionality reduction problem occurs.

In order to solve this problem, an alternative low-dimensional representation of given data should be obtained. There is a wide range of techniques for previously stated problem. Principal component analysis (PCA) and metric multidimensional scaling (MDS) represent most popular methods [2].

Fig.1 – Graph layouts

First method maximizes data variance captured by the low-dimensional projection, and has many extensions. First example is kernel PCA, which apply PCA on a feature space instead of initial data space. Second, probabilistic extension, that address a few weaknesses of PCA: partitioning of data into *r*-dimensional projected subspace (signal) and (*d-r)*-dimensional subspace (noise), where *d* is initial dimensionality, and lack of explicit generative model. Third, robust extension that is driven by fact that a small number of highly corrupted data inputs can drastically influence standard PCA. The latter method also tries to preserve pairwise distances as closely as possible in the least squares sense. One of advantages of this method is that it generally requires only pairwise dissimilarities and not data itself. However, projections of data onto linear subspaces could be insufficient in many cases, when no linear projection is able to provide a satisfying representation. Therefore, this work introduces a graph cut approach to perform supervised dimensionality reduction.

FORMULATION OF PARAMETER GROUPING PROBLEM

Parameter and factor search is performed in order to simplify an a priori description and to study influence of some parameters on others. One of approaches to solve this problem is the approach based on the selection of closely correlated parameters in the original description of groups. A formal formulation of the problem of parameter grouping is given in [2], which is reduced to maximization of one of the two functionals.

$$J_1 = \sum_{j=1}^{m} \sum_{x_i \in A_j} (x_i, f_j)^2, \quad J_2 = \sum_{j=1}^{m} \sum_{x_i \in A_j} |(x_i, f_j)|, \qquad (1)$$

where $x_i, \ldots, x_k$ – random variables (parameters), $A_1, \ldots, A_m$ – a partition of the set $\{x_i, \ldots, x_k\}$ on a given number of m groups, $f_1, \ldots, f_m$ – random functions (essential factors) defined on groups $A_1, \ldots, A_m$, (x_i, f_i) – coefficient of correlation or covariance of the vectors x_i and f_i for parameter partition on groups as well as for random function selection $f_1, \ldots, f_m$.

As it is shown in the [3], functional J_2 is similar to the functional J_1. Maximization of the functional J_2 is reduced to maximization of next functional:

$$J_3 = \sum_{i=1}^{m} \left(\sum_{x_j, x_s \in A_i} \sigma_j \sigma_s (x_j, x_s) \right)^{1/2}, \qquad (2)$$

both by splitting the set of parameters on groups $A_1, \ldots, A_m$ and by coefficient selection σ_j and σ_s, where each coefficient equals to 1 in absolute value. At the same time essential factors $f_1, \ldots, f_m$ are defined with next expression:

$$f_i = \sum_{x_j \in A_i} \sigma_j x_j \Bigg/ \sum_{i=1}^{m} \left(\sum_{x_j, x_s \in A_i} \sigma_j \sigma_s (x_j, x_s) \right)^{1/2}, \qquad (3)$$

where $i = 1, 2, \ldots, m$. From the expression of the functional J_3 it follows that an important problem of functional maximization is to find the maximum of a positive definite quadratic form:

$$\sum_{i=1}^{k} \sum_{j=1}^{k} \sigma_i \sigma_j (x_i, x_j) \qquad (4)$$

on the set of vertices of a *k*-dimensional hypercube of side 1 by modulo. This paper is devoted to present a heuristic algorithm for solving the problem Maximization of positive definite quadratic form on the set of vertices of k-dimensional hypercube. The formulated problem has a trivial solution obtained by complete enumeration method. However, this method of solving the problem is computationally inefficient. Previously formulated problem is reduced to the minimum negative cut problem for weighted graphs with an arbitrary weight function that is bounded from above and below.

Let $R = \| (x_i, x_j) \|$ – matrix of correlation coefficients of the parameters of the set $\{x_i, \ldots, x_k\}$, that corresponds to an arbitrary group from the set of groups $\{A_1, \ldots, A_m\}$. We define a weighted graph on this set as follows. To each parameter x_i it is assigned vertex x_i. Each pair of vertices is connected by an edge. Let V – set of edges of the obtained complete graph. On a set of edges we construct a function according to the rule:

$$c_{ij} = c(v_{ij}) = (x_i, x_j),\ i, j = 1, \ldots, k,\ i \neq j, \tag{5}$$

where v_{ij} – an edge connecting vertices x_i and x_j. We assume that there are no loops at the vertices. The matrix $C = \| c_{ij} \|$ of the weight function c_{ij} is obtained from the matrix *R* by replacing its diagonal elements with zeros. To denote the resulting weighted graph we will use the notation $L(X,V,C)$.

Let us consider the set of all cuts of the graph $L(X,V,C)$ into two subgraphs. Obviously, each of the cuts corresponds to a partition of the matrix of correlation coefficients of the parameters $x_i, \ldots, x_k$ and, accordingly, the matrix *C* into blocks:

$$R = \left(\begin{array}{ccc:ccc}
(x_1,x_1) & \cdots & (x_1,x_s) & (x_1,x_{s+1}) & \cdots & (x_1,x_k) \\
\vdots & R_1 & \vdots & \vdots & R_2 & \vdots \\
(x_s,x_1) & \cdots & (x_s,x_s) & (x_s,x_{s+1}) & \cdots & (x_s,x_k) \\
\hdashline
(x_{s+1},x_1) & \cdots & (x_{s+1},x_s) & (x_{s+1},x_{s+1}) & \cdots & (x_{s+1},x_k) \\
\vdots & R_3 & \vdots & \vdots & R_4 & \vdots \\
(x_k,x_1) & \cdots & (x_k,x_s) & (x_k,x_{s+1}) & \cdots & (x_k,x_k)
\end{array}\right) \tag{6}$$

The main minors R_1 and R_4 of orders s and $k-s$ determine, according to the rule described above, subgraphs of graph cut, which are also complete. A partition of the set *X* into two subsets A and $\overline{A}$ corresponds to each cut of the graph $L(X,V,C)$. The magnitude of the cut is the sum of the correlation coefficients of the antidiagonal blocks R_2 and R_3, which is determined from next expression:

$$W(A,\overline{A}) = \sum_{(x_i,x_j)\in R_1}(x_i,x_j) = \sum_{(x_i,x_j)\in R_{3\,1}}(x_i,x_j) \tag{7}$$

If $W(A,\overline{A}) < 0$, then the cut will be called negative, otherwise – non-negative. The cut with the smallest value of $W(A,\overline{A})$ will be called minimal. Let construct a map of the set of all cuts of the graph $L(X,V,C)$ to the set of vertices of the *k*-dimensional hypercube of side 1 by modulo. To do this, we assign one-to-one correspondence to each *i*-th dimension $(i = 1, \ldots, k)$ of the hypercube some parameter x_i. To designate cuts of the graph $L(X,V,C)$ in terms of subsets of its vertices into which given graph is partitioned when all edges of the cut are removed. The map φ of the set of all cuts of the graph to the set of vertices of the *k*-dimensional hypercube is defined as follows:

$$\begin{aligned} &\varphi(A,\overline{A}) = (1,\ldots,1),\ if\ W(A,\overline{A}) \geq 0, \\ &\varphi(A,\overline{A}) = (\sigma_1,\ldots,\sigma_k),\ if\ W(A,\overline{A}) < 0, \end{aligned} \tag{8}$$

where $\sigma_i = -1$ under $x_i \in A$ and $\sigma_i = 1$ under $x_i \in \overline{A}$.

Theorem 1. Let matrix R defines a graph that contains at least 1 negative cut. Next, an arbitrary negative cut $(A,\overline{A})$ and its vertex $\sigma' = (\sigma'_1,\ldots,\sigma'_k)$ after application of map φ. Suppose that hypercube dimensions are reassigned in a way that next is true:

$$\sigma'_1 = \ldots = \sigma'_s = -1, \tag{9}$$

By multiplying rows and columns of matrix *R* on corresponding components of vector σ' matrix R' will be obtained. From definition of vertex σ' it is follows that elements $(x_i, x_j),\ i = \overline{1,k},\ j = \overline{s+1,k}$ of matrix R change their sign. Since negative cut $(A,\overline{A})$ was chosen arbitrarily for any such cut it is possible to write:

$$4W(A,\overline{A}) = \sum_{i,j=1}^{k}(x_i,x_j) - \sum_{i,j=1}\sigma_i\sigma_j(x_i,x_j) < 0, \tag{10}$$

i.e. expression (4) at vertex σ' obtains value higher that at vertex $\sigma = (1,\ldots,1)$ by exactly quadrupled value of negative cut for which σ' is an image. Since minuend at (E1) constant then $W(A,\overline{A})$ will be minimal only when subtrahend is maximal. It implies that if graph $L(X,V,C)$ does not contain negative cuts then maximum for expression (4) on hypercube vertices reach minimal negative graph cut at vertex (1,..,1) or otherwise at vertex that is an image of minimal negative graph cut. The quadratic form (4) attains maximal value at that vertex of the k-dimensional hypercube, which is the image of the minimal negative cut of the graph $L(X,V,C)$ under the map φ. This theorem allows us to construct an iterative algorithm to find the minimum negative cut of the graph $L(X,V,C)$. Suppose some procedure *P* for negative (not only minimum) cut search in weighted graph is given. As a result of its application to any graph some negative cut will be obtained or it will be determined that given graph does not consist negative cuts that can be found with this procedure. Weighted graph and solution vector, obtained on *s*-th step of the algorithm, are denoted as $L(X,V,C^s)$ and $\hat{\sigma}^s$ respectively. On the first step of the algorithm $\hat{\sigma}^i = \sigma^1$. According to (5) map φ^s is defined on the set of graph cuts for $L(X,V,C^s)$:

1. Let on $(s+1)$-th step application of procedure *P* to graph $L(X,V,C^s)$ gives negative cut $(A^{s+1},\overline{A}^{s+1})$. With map φ^s vector σ^{s+1} is obtained. If cut $(A^{s+1},\overline{A}^{s+1})$ is not negative, then go to step 4.

2. Construct vector:

$$\hat{\sigma}^{s+1} = \overline{\sigma}^s \oplus \sigma^{s+1}, \tag{11}$$

where symbol $\oplus$ denotes componentwise product of given vectors. For this vector next inequality occurs:

$$\sum_{i,j=1}^{k} \hat{\sigma}_i^{s+i} \hat{\sigma}_j^{s+i}(x_i,x_j) > \sum_{i,j=1}^{k} \hat{\sigma}_i^{s} \hat{\sigma}_j^{s}(x_i,x_j), \tag{12}$$

which follows from next relation:

$$4W(A^s,\overline{A}^s) = \sum_{i,j=1}^{k} \hat{\sigma}_i^{s} \hat{\sigma}_j^{s}(x_i,x_j) - \sum_{i,j=1}^{k} \hat{\sigma}_i^{s+1} \hat{\sigma}_j^{s+1}(x_i,x_j) < 0 \tag{13}$$

and definition of procedure *P*.

1. Multiply matrix *R* by rows and columns on corresponding coordinates of vector $\hat{\sigma}^{s+1}$. Matrix R^{s+1}, which defines graph $L(X,V,C^{s+1})$ is obtained. Go to item 1 of $(s+2)$-th step.

2. If on $(s+1)$-th step of procedure *P* application to $L(X,V,C^s)$ non-negative cut is obtained, then it means that iterative process is over. Vector $\hat{\sigma}^s$ is a solution.

Such algorithm will allow us to obtain a finite sequence of vertices of the *k*-dimensional hypercube $\hat{\sigma}^1,\ldots,\hat{\sigma}^s$. From the construction of this sequence, it follows that at each subsequent vertex the quadratic form (4) receives greater value than at the previous one. At each iteration step the value of the quadratic form increases by a value, that is bounded from below. It follows that the algorithm converges in a finite number of steps.

For such iterative algorithm of minimum negative cut search of the graph, it is true that at the vertex $\hat{\sigma}^s$, the expression (4) receives a local maximum in regard to the procedure *P*. This obtained maximum will be global only when the procedure *P* has next property: if the graph $L(X,V,C)$ contains at least one negative cut, then the result of the procedure *P* will be some negative cut.

Algorithm for minimal negative cut problem of weighted graphs

The minimum negative cut search continues the search procedure for an arbitrary negative cut of the graph. At first, the orientation of the graph is formed and the initial flows are arranged. As a result, the graph turns into a network. The resulting network is then converted to a network with non-negative bandwidths. Then, using the Ford-Fulkerson algorithm [4], a minimal network cut is computed and its analysis is performed. Let a complete k-vertex weighted graph $L(X,V,C)$ be given. We assume that the capacity matrix of the edges of the network obtained by orientation of the graph coincides with the weight matrix C. If i-th edges enter the network vertex and j-th edges exit, then we will say that this vertex has (i, j) type. We assume that the set of all such pairs of vertices $< x_i, x_j$, that $i < j$, is lexicographically ordered. Then, cyclical sorting for vertices pairs sequencing from this set is started. At each stage of the algorithm, we will consider one pair of vertices. Suppose that to the $(l+1)$ stage of the algorithm, we obtain matrices C^l and a k-dimensional vector $\hat{\sigma}^l$, where coordinates equal to 1 in absolute value. In addition, let a number of selected negative cuts for the next vertices pairs sorting equals r and the pair of vertices $\langle x_s, x_t \rangle$ is under study. In this case, each stage contains 12 steps:

1. We assume that x_s is a source, x_t is a sink.
2. The edges incident to the vertex x_s are replaced by edges that go out of this vertex, and the edges incident to the vertex x_t are replaced by edges that return to it.
3. Enumeration of all 4-vertex subgraphs, generated by a set of vertices $\{x_s, x_i, x_j, x_t\}$, $i, j = \overline{1,k}$ and $i \neq j \neq s \neq t$. In each such subgraph the edge v_{ij} is replaced by an edge that goes out of the vertex x_i if $c_{si} + c_{jt} > c_{sj} + c_{it}$ and an edge, that returns to this vertex, if $c_{si} + c_{jt} > c_{sj} + c_{it}$. In case of equality $c_{si} + c_{jt} > c_{si} + c_{it}$ the edge v_{ij} is not oriented at this step.
4. Arrange all the vertices of a partially oriented graph in decreasing order by the number of edges that return to them. Then, all the vertices are successively considered and replace incident to them edges that are not oriented by the moment of scanning, by the edges that go out of them.
5. Arrange the original streams $f(\bar{v}_{ij})$ according to rules:

a) $f(\bar{v}_{ij}) = -c, if\ i \neq j \neq s \neq t,$

b) $f(\bar{v}_{si}) = -(q-p+1)c, f(\bar{v}_{it}) = -c,$

if vertex has type (p,q) and $q > p,$

c) $f(\bar{v}_{si}) = f(\bar{v}_{it}) = -c,$ if vertex x_i has type (p,q) and $p = q,$

d) $f(\bar{v}_{si}) = -c, f(\bar{v}_{it}) = -(q-p+1)c,$ if vertex x_i has type (q,p) and $q > p,$ where $c = \max\limits_{v_{iv} \in v} |c_{ij}|.$

6. Build a new network with non-negative capacities by using formula $\tilde{c}(\bar{v}_{ij}) = c(\bar{v}_{ij})$ for all $i, j = \overline{1,k}$ and $i \neq j.$ Initial flows over all edges are set to zero.

7. Apply Ford-Fulkerson algorithm to find the minimum cut of the resulting network.

8. Check if the resulting network cut is a graph cut using the algorithm for collocation labeling [4]. Value of the obtained minimum cut of initial network is determined from the expression:

$$\overline{W}(A^{l+1}, \overline{A}^{l+1}) = \sum_{i=1, i \neq s} (f^*(\bar{v}_{si}) + f(\bar{v}_{si})), \tag{14}$$

where $A^{l+1}, \overline{A}^{l+1}$ – vertex sets into which the graph is cut. Go to step 11.

9. If the network cut is not a graph cut, then we divide the set of all vertices of the network, excluding source and sink, into non-overlapping subsets. In order to do so suppose that all these vertices are arranged in descending order according to number of its exit edges. If vertex is reachable from the source, then all vertices $x_1, \ldots, x_{s_1}$ that are reachable from x_s will be successively appended to subset A_1 until first unreachable vertex x_{s_1+1}. In case, when vertex x_1 is unreachable subset A_1 is appended by all vertices $x_1, \ldots, x_{s_1}$ that are not reachable from vertex x_s until first attainable vertex x_{s_1+1}. This process proceeds for all vertices. As a result, a sequence of non-overlapping subsets $A_1, \ldots, A_{k_1}$ will be obtained. In this sequence subsets of reachable and unreachable vertices will alternate. Next, complete set of these subsets is divided onto two non-overlapping groups. First group consist of those subsets, where vertices are reachable from x_s. The rest of subsets create second group.

10. Based on the initial network a new network is built by changing the orientation of some edges. Edge orientation that exit second group's vertices and enter first group's ones change to opposite. For all vertices that were

changed new flows for edges $\bar{v}_{si}$, $\bar{v}_{jt}$ are set according to step 6 and for the rest of edges source flow is unchanged. Go to step 7.

11. If there is inequality $\overline{W}(A^{l+1},\overline{A}^{l+1})$, then we construct the vector σ^{l+1}, new matrix of weights C^{l+1} is created by using the matrix C^{l} and components σ^{l+1} and the vector $\hat{\sigma}^{l+1}=\hat{\sigma}^{l}\oplus\sigma^{l+1}$. Increase the counter of found negative cuts r by one. If $\overline{W}(A^{l+1},\overline{A}^{l+1}),\geq 0$, then define matrix $C^{l+1}=C^{l}$ and proceed further.

12. Check if the pair $\langle x_s,x_t\rangle$ is the last in a lexicographically ordered list of all pairs of vertices. If true and $r=0$, then the work is finished and $\hat{\sigma}^{l+1}$ is a solution. If $r\neq 0$, then define $r=0$ and proceed to step 1. If the pair $\langle x_s,x_t\rangle$ is not the last one, then select the next pair and go to step 1 of $(l+2)$ -th stage.

Next theorems were used to substantiate given algorithm.

Theorem 2. For every cut $(A,\overline{A})$ of canonic network to given arbitrary network there is next equality:

$$W^{*}(A,\overline{A})=\sum_{\bar{v}_{ij}\in V(A,\overline{A})}c^{*}(v_{ij})=\sum_{\bar{v}_{ij}\in V(A,\overline{A})}(cv_{ij})-c_0(\bar{v}_{ij})=\sum_{\bar{v}_{ij}\in V(A,\overline{A})}c(\bar{v}_{ij})-$$
$$-\sum_{\bar{v}_{ij}\in V(A,\overline{A})}c_0(\bar{v}_{ij})+\sum_{\bar{v}_{ij}\in V(\overline{A},A)}c_0(\bar{v}_{ij})-\sum_{\bar{v}_{ij}\in V(\overline{A},A)}c_0(\bar{v}_{ij})=\sum_{\bar{v}_{ij}\in V(A,\overline{A})}c(\bar{v}_{ij})- \tag{15}$$
$$-\sum_{\bar{v}_{ij}\in V(\overline{A},A)}c_0(\bar{v}_{ij})-\Big(\sum_{\bar{v}_{ij}\in V(A,\overline{A})}c_0(\bar{v}_{ij})-\sum_{\bar{v}_{ij}\in V(\overline{A},A)}c_0(\bar{v}_{ij})\Big)=\overline{W}(A,\overline{A})-v_0.$$

Since initial flow value through any cut of an arbitrary network is constant then from above mentioned equality it is follows that minimum cuts of an arbitrary network coincide with minimum cuts of its canonic network. Therefore, it is necessary to switch to canonic network and apply Ford-Fulkerson algorithm to it in order to find minimum cut of initial arbitrary network. Every graph with an arbitrary weight function can be transformed to an arbitrary network by selecting some vertex pairs as a source and a sink and certain orientation of its edges. Each such network is associated with number, that equals to minimum cut. Size of minimum cut for obtained network depends on graph orientation and choice of vertex pairs. Thus, some minimum arbitrary network is obtained.

Theorem 3. Let some weighted graph $L(X,V,C)$ and its minimal cut $(A,\overline{A})$ are given. Next, we choose vertices x_s, x_t from set A and $\overline{A}$ in arbitrary way and take them as source and sink respectively.

All edges that are incident to vertex x_s substitute to arcs that exit this vertex and edges that are incident to vertex x_t substitute to entering arcs. Edges that connect vertices from set A to vertices from set $\overline{A}$ replace with arcs that exit set A. Orientation for the rest edges we choose arbitrary. For every arc we assign flow capacity that equals to weight of corresponding edge. Initial flow function is chosen according to 6th step of algorithm for minimal negative graph cut search. It is clear that for obtained network conditions (7) - (9) are true. Let show that cut $(A,\overline{A})$ of obtained network will be its minimal cut. Suppose that network cut $(A^*,\overline{A^*})$ is smaller then there is an inequality:

$$W(A,\overline{A}) = \overline{W}(A,\overline{A}) = \sum_{\bar{v}_{ij}\in V(A,\overline{A})} c(\bar{v}_{ij}) > \overline{W}(A^*,\overline{A^*}) = \\ = \sum_{\bar{v}_{ij}\in V(A^*,\overline{A^*})} c(\bar{v}_{ij}) - \sum_{\bar{v}_{ij}\in V(\overline{A^*},A^*)} c_0(v_{ij}). \tag{16}$$

With next condition:

$$c_0(\bar{v}_{ij}) \le f(\bar{v}_{ij}) \le c(\bar{v}_{ij}), \quad c_0(\bar{v}_{ij}) < 0, \quad |c_0(\bar{v}_{ij})| \ge c(\bar{v}_{ij}) \tag{17}$$

It is follows that:

$$\overline{W}(A^*,\overline{A^*}) = \sum_{\bar{v}_{ij}\in V(A^*,\overline{A^*})} c(\bar{v}_{ij}) - \sum_{v_{ij}\in V(\overline{A^*},A^*)} c_0(\bar{v}_{ij}) \ge \\ \ge \sum_{\bar{v}_{ij}\in V(A^*,\overline{A^*})} c(\bar{v}_{ij}) + \sum_{\bar{v}_{ij}\in V(\overline{A^*},A^*)} c(\bar{v}_{ij}) = W(A^*,\overline{A^*}). \tag{18}$$

Thereby, $W(A,\overline{A}) > W(A^*,\overline{A^*})$ that contradict with minimum condition of graph cut $(A,\overline{A})$. And it is follows that for any graph with an arbitrary weight function there is such orientation and vertex pair option as source and sink for which minimum cut of the obtained network coincide with minimum cut of the graph and also equals to it and such network is a minimum arbitrary network.

Theorem 4. Let minimal network cut $(A,\overline{A})$ is not the graph cut. Set of all vertices is divided on subsets in a way that is described in 9th step of the algorithm. Suppose that $A_{i_1},\dots A_{i_{k_1}}$ and $A_{j_1},\dots A_{j_{k_2}}$ – reachable and unreachable vertex subsets respectively. Suppose $B=\bigcup_{s=1}^{k_1} A_{i_s}$ and $B^{'}=\bigcup_{s=1}^{k_2} A_{j_s}$. In this regard minimal cut $(A,\overline{A})$ has next value:

$$\overline{W}(A,\overline{A})=\sum_{x_i\in B} c(\overline{v}_{si})+\sum_{\overline{v}_{ij}\in V(B,B^{'})} c(\overline{v}_{ij})+\sum_{x_i\in B^{'}} c(\overline{v}_{it})-\sum_{\overline{v}_{ij}\in V(B^{'},B)} c_0(\overline{v}_{ij}). \tag{19}$$

After reorientation of arcs from set $V(B^{'},B)$ we obtain network in which the same cut $(A,\overline{A})$ has value:

$$\overline{W}^{'}(A,\overline{A})=\sum_{x_i\in B^{'}} c(\overline{v}_{si})+\sum_{\overline{v}_{ij}\in V^{*}(B,B^{'})} c(\overline{v}_{ij})+\sum_{x_i\in B^{'}} c(\overline{v}_{it})+\sum_{\overline{v}_{ij}\in V^{**}(B,B^{'})} c(\overline{v}_{ij}). \tag{20}$$

where $V^{*}(B,B^{'})$ –arc set that link vertex sets B and $B^{'}$ where reorientation have not applied and $V^{**}(B,B^{'})$ – arc set where reorientation was applied. According to condition (17) it is follows that $W(A,\overline{A})=\overline{W}^{'}(A,\overline{A})\le\overline{W}(A,\overline{A})$ and then minimal cut of new network does not increase in accordance to theorems 2 and 3. As a result it can be stated that if minimal cut of arbitrary networkis not a cut for corresponding graph then there is effective algorithm for reorientation that allows for minimal cut of new network not to increase and it is possible to achieve network with the cut that also belongs to the graph in finite number of steps. Hence, minimum negative cut problem for weighted graphs with an arbitrary weight function is reduced to search problem for graph orientation, that all edges of its minimum cut $(A,\overline{A})$ exit vertices of set A. Previously stated algorithm of minimum negative cut search of the graph is based on idea of minimum arbitrary cut search that will enable to find global minimum negative cut of given graph.

In this paper correlation coefficients were used for dependence estimation that imply some limitation on overall estimation of dependence between vectors since there is difference between dependence measurements of *n* random variables and pairwise dependence measurements. One of approaches to tackle this problem was presented in work [9] where new method that able to distinguish between pairwise independence and higher-order dependence of random vectors is shown. Since the main aim of this paper represent another subject more details on above mentioned aspect will be examined in further works.

Results of given algorithm application to the real-world data will be presented in the following publications on this topic.

REFERENCES

[1] O. Kwon and K. Ma. What Would a Graph Look Like in This Layout? A Machine Learning Approach to Large Graph Visualization. IEEE Transactions on Visualization and Computer Graphics,2018. – vol. 24. – P. 478 – 488.
[2] Айвазян,В.М. Бухштабер, И.С. Енюков, Л.Д. Мешалкин; под ред. С.А. Айвазяна. - М.: Финансы и статистика, 1989. – 607с.
[3] Э. М. Браверман. Методы экстремальной группировки параметров и проблема распределения существенных факторов. Автоматика и телемеханика,1970. – вып. 1. – С. 123 – 132.
[4] L. R. Ford and D. R. Fulkerson. Flows in networks. Princeton University Press, 1962. – 332 p.
[5] H. Rogers. Theory of Recursive Functions and Effective Computability. MIT Press, 1987. – 17 p.
[6] M. R. Garey and D. S. Johnson. Computers and Intractability: A Guide to the Theory of NP-Completeness. W. H. Freeman, 1979. – 347 p.
[7] J. E. Savage.The Complexity of Computing. Krieger Pub Co,1987.-333 p.
[8] O. Kwon and K. Ma. What Would a Graph Look Like in This Layout? A Machine Learning Approach to Large Graph Visualization. IEEE Transactions on Visualization and Computer Graphics, 2018. – vol. 24. – P. 478 – 488.
[9] B. Böttcher, M. Keller-Ressel, and R. L. Schilling. Distance Multivariance: New dependence measures for random vectors. The Annals of Statistics, 2019. – P. 2757 – 2789.
[10] D. H. Bailey. Experimental Mathematics in Action. 2007. – 323 p.
[11] A. Skorokhod. Random Perturbation Methods with Application in Science and Engineering. Springer – New York, 2002. – 489 p.

УМЕНЬШЕНИЕ РАЗМЕРНОСТИ И ГРУППИРОВКА ЭКСТРЕМАЛЬНЫХ ПАРАМЕТРОВ ДЛЯ ЗАДАЧ ВЗВЕШЕННОГО ГРАФА

Востров Г., Хриненко А.

В работе представлен подход графового разреза к уменьшению размерности и показаны связи между задачей группирования экстремальных параметров и задачей максимизации положительно определенной квадратичной формы на множестве вершин k-мерного гиперкуба стороны 1 по модулю. Показано, что последняя задача сводится к поиску минимального отрицательного разреза для взвешенного графа. Приведен эвристический алгоритм решения задачи.

Ключевые слова. Наборы данных, уменьшение размерности, графы.

ВЫВОДЫ

В выполненной коллективной монографии приведены результаты научных исследований в области информационных интеллектуальных технологий, моделирования в информационных управляющих системах, управления защитой информационных систем, совершенствования информационно-ресурсного обеспечения науки.

Приведенные материалы позволили решить ряд задач связанных с:

совершенствованием информационно-ресурсного обеспечения науки, техники и социальной сферы; способы и методы защиты информации;

информационными интеллектуальными технологиями для автоматизированных систем обработки данных и управления;

математическим моделированием и оптимизацией в информационных управляющих системах;

информационными технологиями управления проектами.

Результаты выполнения работ по перечисленным разделам позволяют решить некоторые проблемы информационных управляющих систем и технологий.