Decision Making in Manufacturing and Services
Vol. 19 • 2025 • pp. 41–58

Significant and Trivial Dependencies Separation in Data Tensor by the Projection Method

Yuriy Bunyak[*], Roman Kvyetnyy[**], Olga Sofina[***]

Abstract. The problem of data structure analysis through their multidimensional representation as a d-dimensional tensor is considered to assess dependencies on influencing factors in the decision-making process. The higher-order Singular Value Decomposition (SVD) is developed as a d-SVD schema to identify significant and trivial dependencies. The d-SVD includes the SVD of the tensor reshaped as a matrix and the SVDs of reduced size of the previous SVD vectors reshaped as matrices. The entropy of the distribution of the Singular Values (SVs) of the vectors' decomposition is used for the separation of the significant and trivial vectors, in contrast to the commonly used approach based on the magnitude analysis of SVs. The singular projection in the significant vector space in selected dimensions gives the tensor's low-rank approximation without loss of information in comparison with the truncated SVD. The tensor projection on a vector subspace of reduced dimension can be obtained by using a part of the SVs and the corresponding vectors as an alternative to the commonly used averaging. It was shown that data prediction in the subspace of the significant vectors allows stable assessments of the predicted values to be obtained.

Keywords: multidimensional data tensor, higher-order singular value decomposition, singular values distribution entropy, singular projection, tensor projection, data pattern assessment

Mathematics Subject Classification: 91G05

JEL Classification: C55

Submitted: February 20, 2025

Revised: May 7, 2025

[*] Spilna sprava company, Vinnytsia, Ukraina, e-mail: iuriy.buniak@gmail.com.
[**] Vinnytsia National Technical University, Department of Automation and Intelligent Information Technologies, Vinnytsia, Ukraine, e-mail: rkvetny@sprava.net.
[***] Vinnytsia National Technical University, Department of Automation and Intelligent Information Technologies, Vinnytsia, Ukraine, e-mail: olsofina@gmail.com.

1. INTRODUTION

The decision-making process in economic and production activities, financial transactions and online trading is based on the analysis of factors influencing a desired target in order to assess the success of expected operations (Dixon et al., 2020). Factors of influence are extracted from the statistical analysis of the history of operations based on the study of dependencies between obtained results and factors parameters. Factors and their statistical features form an information base of artificial intelligence for solving decision-making problems.

As the factors can be considered time parameters, macroeconomic indicators reflecting economic development trends and the investment climate, financial transaction parameters. A website, location on the site, its price, as well as age, gender and type of access device of a customer can be considered as influencing factors in online trading.

The financial market and markets of goods and services are characterized by mass and repetition. Therefore, it is necessary to analyze big data of transaction parameters simultaneously with the market parameters in order to identify those factors that affect the success of expected transactions.

Decision-making results depend on the ability of the data analysis to identify significant dependencies, anomalies and their causes.

Correlation analysis is usually used to identify dependencies between dynamic of data and factors of influence (Cheng et al., 2018; Czapkiewicz & Skalna, 2014; Ganeshapillai et al., 2013; Kenett et al., 2012). However, the evaluation of correlation involves averaging over large arrays of data values. This results in the sought-after dependencies being blurred and remaining hidden.

Forecasting methods are used in the process of decision-making in financial bidding transactions through the internet (Caraiani, 2014; Dixon et al., 2020; Huang & Vakharia, 2024; Sofina & Bunyak, 2021; Spentzouris et al., 2018). Probabilistic prediction models and regression-based models do not separate significant and trivial dependencies and need additional methods to reduce their dependencies on factors of noise-like fluctuations.

Another tool for dependencies analysis is the presentation of tables with many columns and rows in the form of a multidimensional tensor, this procedure is known as tensorization (Cichocki et al., 2016; Hackbusch & Kuhn, 2009). Tucker (1966) proposed decomposing multivariate data onto influence factors matrices to simplify and make decision-making process more efficient in conditions of multiple influence factors. The tensor decomposition on one-rank components and low-rank approximation make it possible to eliminate repetition in data for selected dimensions and identify anomalous manifestations, find their features and dependencies with data in other dimensions (Cichocki et al., 2017; Phan & Cichocki, 2010).

The Singular Value Decomposition (SVD) is used to represent a matrix as the weighted sum of independent one-rank matrices formed as the tensor product of mutually orthogonal columns and mutually orthogonal rows. The magnitude of the Singular Values (SVs) determines the significance of the corresponding one-rank components of a decomposed matrix. The approaches to the analysis of the spectrum of the SVs in

data processing problems are considered in the review (Zhigljavsky, 2010). Also, the distribution of SVs, its entropy, characterizes a structure of the data (Alvarez-Ramirez & Rodriguez, 2021; Caraiani, 2014; 2018). The SVD is used to separate regular and chaotic components of data when solving forecasting problems (Montagnon, 2020).

The tensor of dimension d is reshaped into a matrix by a lexicographic one-dimensional representation of two parts of its dimensions in unfolded form for the SVD (Comon, 2002; De Lathauwer et al., 2000; Oseledets, 2011). There are some approaches to separate the SVD of the unfolded tensor by dimensions, commonly referred to as Higher-Order Singular Value Decomposition (HOSVD) (Bergqvist & Larsson, 2010; Cichocki et al., 2016; De Lathauwer et al., 2000).

From this point of view, in order to analyze dependencies and select the best approximation of data with a minimum number of independent components, a d-dimensional HOSVD with explicitly presented SVs on each dimension is needed. The known Tucker type decompositions and other implementations of the HOSVD – tensor train, tensor ring (Cichocki et al., 2016), hide the SVs distributions.

Known HOSVD methods work with unfolded tensors in the form of matrices, which can be very large in at least one of the dimensions, but this causes difficulties in implementing algorithms of data dependencies analysis. The efficiency of using the HOSVD schemes depends on the rank of the unfolded tensors that is usually determined by identifying the most significant SVs by magnitudes that is known as the truncation (Cichocki et al., 2016; Hackbusch & Kuhn, 2009; Oseledets, 2011; Vannieuwenhoven et al., 2012). This operation is ambiguous because the sum of truncated components can lead to the loss of characteristic properties of the data. A criterion for selecting SVs and their corresponding singular vectors is needed that allows to preserve the data structure in their low-rank approximation in order to extract significant dependencies.

The proposed schema of the HOSVD in the article reduces computational complexity and separates singular values by dimensions. An approach to select significant vectors of the decomposition through the analysis of the entropy of their SVs distribution is offered.

The aim of the article is to use the HOSVD to develop the projection method to improve the decision-making process. The method analyzes the structure of big data through their multidimensional presentation and identifies significant dependencies on factors of influence as the HOSVD's vector space of reduced size and dimension. It uses the selection of the most informative components of the HOSVD to preserve features of the data.

2. RELATED DECOMPOSITIONS OF TENSORS

The data array is represented as a d-dimensional tensor

$$[x_{i_0;i_1;\ldots;i_{d-1}}]_{i_k=0\ldots N_k-1;k=0\ldots d-1} \tag{1}$$

of size $N_0 \times N_1 \times \ldots \times N_{d-1}$. The indices of tensor (1) elements of different dimensions are separated by the ";". The indices of dimensions in the lexicographic one-dimensional representation are not separated, this means, for example, that a tensor element with

the indices $i_0;i_1;i_2;i_3$ corresponds to the element with the index $i_0 + N_0 i_1 + N_0 N_1 i_2 + N_0 N_1 N_2 i_3$ in the unfolded representation denoted as $i_0 i_1 i_2 i_3$. Matrices with different indices are different.

Tucker type decomposition of tensor (1) can be implemented with using the following technique of the HOSVD (De Lathauwer et al., 2000; Savas & Eldén, 2007; Vannieuwenhoven et al., 2012). The tensor elements are reshaped as d matrices of elements $x_{i_n;i_0\ldots i_{n-1}i_{n+1}\ldots i_{d-1}}$ and are defined the SVDs of the matrices:

$$x_{i_n;i_0\ldots i_{n-1}i_{n+1}\ldots i_{d-1}} = \sum_{k_n} u_{i_n;k_n} s_{k_n} v_{i_0\ldots i_{n-1}i_{n+1}\ldots i_{d-1};k_n} \tag{2}$$

where $n = 0, \ldots, d-1$, $k_n = 0, \ldots, N_{n-1}$ or k_n does not exceed value of the rank of the matrix in n-th dimension, $u_{i_n;k_n}$ are elements of N_n left orthogonal unitary vectors of N_n elements, in this case of a factor matrix of size $N_n \times N_n$, and $v_{i_0\ldots i_{n-1}i_{n+1}\ldots i_{d-1};k_n}$ are elements of N_n right orthogonal unitary vectors of $\prod_{k=0\ldots d-1;k\neq n} N_k$ elements, s_{k_n} – the elements of diagonal matrices of the SVs. The elements of the core tensor are defined in the vector space of the factor matrices.

$$c_{k_0;\ldots;k_{d-1}} = \sum_{i_0} \ldots \sum_{i_{d-1}} x_{i_0;i_1;\ldots;i_{d-1}} u_{i_0;k_0} \ldots u_{i_{d-1};k_{d-1}} \tag{3}$$

The original tensor can be presented as multilinear product of core tensor (3) and the matrices of factors.

$$x_{i_0;\ldots;i_{d-1}} = \sum_{k_0} \ldots \sum_{k_{d-1}} c_{k_0;\ldots;k_{d-1}} u_{i_0;k_0} \ldots u_{i_{d-1};k_{d-1}} \tag{4}$$

If unfolded data matrices in (2) are not of full rank then the HOSVD yields core tensor (3) of smaller size then initial tensor (1). The factor matrices have equal norms, so their influence depends on the structure of the core tensor. The matrices in (2) are formed by reshaping of the d-dimensional tensor and therefore may be very large. This complicates the calculation of the SVDs (2) despite only their left matrices needing to be calculated.

The tensor train decomposition (TTD) (Oseledets, 2011) uses the SVD

$$x_{i_0;i_1\ldots i_{d-1}} = \sum_{k_0} u_{i_0;k_0} s_{k_0} v_{i_1\ldots i_{d-1};k_0} \tag{5}$$

of the tensor (1) reshaped as a matrix for further step-by-step decomposition using reshaping and the SVD. The first step is the next one.

$$s_{k_0} v_{k_0 i_1;i_2\ldots i_{d-1}} = \sum_{j_0 k_1} u_{k_0 i_1;j_0 k_1} s_{j_0 k_1} v_{i_2\ldots i_{d-1};j_0 k_1} \tag{6}$$

where joined indices $j_0 k_1 = 0, \ldots, N_0 N_1 - 1$. The reshaped matrix of right vectors as $s_{j_0 k_1} v_{j_0 k_1 i_2;i_3\ldots i_{d-1}}$ is represented by SVD like (6) too and so on. At the end of the iterative process the following HOSVD of the tensor elements is obtained:

$$x_{i_0;\ldots;i_{d-1}} = \sum_{k_0} u_{i_0;k_0} \sum_{j_0 k_1} u_{k_0 i_1;j_0 k_1} \ldots \sum_{j_0\ldots j_{d-2}k_{d-1}} u_{j_0 j_1\ldots i_{d-2};k_{d-1}} s_{j_0\ldots j_{d-2}k_{d-1}} v_{i_{d-1};k_{d-1}} \tag{7}$$

Decomposition (7) is effective if instead of sums by joined indices $j_0 ... j_{d-2} k_{d-1} = 0, ..., N_0 ... N_{d-1} - 1$ it uses sums in a range of low-rank approximation of the corresponding matrices. In general, TTD (7) needs in d–1 SVDs of the total size $\prod_{k=0...d-1} N_k$ with possible decrease due to the low-rank approximation.

At our disposal are the hierarchical Tucker-type decomposition (Grasedyck, 2010; Hackbusch & Kuhn, 2009) and TTD (Zniyed et al., 2020) of a tensor reshaped as a matrix nearest to a square one to split the left and right matrices of singular vectors by dimensions in parallel.

The SVD of a matrix of size $N \times N$ needs in ~$4N^3$ flops. It is evident that it is better to calculate the SVD of some number of matrices of small size than of one matrix of large size. Therefore a version of the techniques presented above is offered which includes only one SVD of maximum size and a series of the SVDs of reduced size which number depends on the rank of the matrix obtained as a reshaped original tensor. Also, the offered d-SVD allows to obtain diagonal matrices of SVs for each dimension of the tensor. The distribution of the SVs is used to analyze the data's dependencies on factors of influence.

3. THE d-SVD SCHEMA OF THE HOSVD

Tensor (1) can be reshaped as 2-dimensional matrix of size $\prod_{k=0...d-2} N_k \times N_{d-1}$ with elements $x_{i_0 ... i_{d-2}; i_{d-1}}$. The SVD of the matrix elements is the following.

$$x_{i_0 ... i_{d-2}; i_{d-1}} = \sum_{k_{d-1}} u_{i_0 ... i_{d-2}; k_{d-1}} s_{k_{d-1}} v_{i_{d-1}; k_{d-1}} \tag{8}$$

The right-side vectors of elements $v_{i_{d-1}; k_{d-1}}$ present one extracted dimension of the tensor. Each k_{d-1}-th left-side vector of elements $u_{i_0 ... i_{d-2}; k_{d-1}}$ in (8) can be reshaped as a matrix of elements $u_{i_0 ... i_{d-3}; i_{d-2}; k_{d-1}}$ and represented by the SVD, then

$$x_{i_0 ... i_{d-3}; i_{d-2}; i_{d-1}} = \sum_{k_{d-1}} s_{k_{d-1}} v_{i_{d-1}, k_{d-1}} \sum_{k_{d-2}} u_{i_0 ... i_{d-3}; k_{d-2}; k_{d-1}} . s_{k_{d-2}; k_{d-1}} v_{i_{d-2}; k_{d-2}; k_{d-1}} \tag{9}$$

The elements $v_{i_{d-2}; k_{d-2}; k_{d-1}}$ of the vectors of the next dimension were extracted in (9) with the help of the SVDs by variables $k_{d-2}; k_{d-1}$. SVD (8) of initial data and d-2 iterations (9) of the left-side vectors reshaping as matrices and the SVD to extract vectors of the following dimensions yield the representation of the original tensor elements as

$$x_{i_0; ...; i_{d-1}} = \sum_{k_{d-1}} s_{k_{d-1}} v_{i_{d-1}; k_{d-1}} \sum_{k_{d-2}} s_{k_{d-2}; k_{d-1}} v_{i_{d-2}; k_{d-2}; k_{d-1}} \cdots \sum_{k_1} u_{i_0; k_1; k_2; ...; k_{d-1}} s_{k_1; k_2; ...; k_{d-1}} v_{i_1; k_1; k_2; ...; k_{d-1}} \tag{10}$$

The number of sums levels in d-SVD (10) is equal to d−1, as in the usual SVD of a 2-dimensional matrix.

The selection of only the main vectors with zero indices in (10) gives the one-rank Singular Projection (SP) of the tensor elements

$$x^{(0_1; ...; 0_{d-1})}_{i_0; ...; i_{d-1}} = s_{0_{d-1}} v_{i_{d-1}; 0_{d-1}} s_{0_{d-2}; 0_{d-1}} v_{i_{d-2}; 0_{d-2}; 0_{d-1}} \cdots u_{i_0; 0_1; ...; 0_{d-1}} s_{0_1; ...; 0_{d-1}} v_{i_1; 0_1; ...; 0_{d-1}} \tag{11}$$

Similar components can be extracted for combinations of indices in brackets in (11) that correspond to the most significant SVs. These components are independent due to orthogonality of the vectors on each step of the *d*-SVD and therefore can be summarized to obtain an approximation of the initial tensor.

As it follows from (8)–(10), the tensor decomposition needs one SVD of size $\prod_{k=0\ldots d-2} N_k \times N_{d-1}$, N_{d-1} SVDs of size $N_0 N_1 \ldots N_{d-3} \times N_{d-2}$, $N_{d-2} N_{d-1}$ of size $N_0 N_1 \ldots N_{d-4} \times N_{d-3}$ and finally $N_2 \ldots N_{d-1}$ SVDs of size $N_0 \times N_1$. One-rank component (11) needs one SVD of each size listed above.

The number of SVDs on each step of (10) can be determined by the number of significant SVs on the previous step. It is advisable to reshape tensor (1) as a matrix nearest to square one of size $N_0 \ldots N_{e-1} \times N_e \ldots N_{d-1}$ for selection of maximum number of significant SVs. The SVD of the matrix is the following.

$$x_{i_0 \ldots i_{e-1}; i_e \ldots i_{d-1}} = \sum_{k=0}^{K-1} u_{i_0 \ldots i_{e-1}; k} s_k v_{i_e \ldots i_{d-1}; k} \tag{12}$$

where K is rank of the matrix. Each of K left and right vectors of SVD (12) may be reshaped as matrices and represented by the SVDs like in (9) in respect to dimensions, for example, numbers $e-1$ and $d-1$, then the tensor with two extracted dimensions looks as

$$x_{i_0 \ldots i_{e-2}; i_{e-1}; i_e \ldots i_{d-2}; i_{d-1}} = \sum_{k=0}^{K-1} s_k \sum_{k_{e-1}} u_{i_0 \ldots i_{e-2}; k_{e-1}; k} \cdot s_{k_{e-1}; k} v_{i_{e-1}; k_{t-1} k} \sum_{k_{d-1}} u_{i_e \ldots i_{d-2}; k_{d-1}; k} s_{k_{d-1}; k} v_{i_{d-1}; k_{d-1}; k} \tag{13}$$

Decomposition (13) is the decomposition of the left and right vectors of SVD (12) separately to the form of *d*-SVD (10), it can be considered as the hierarchical *d*-SVD.

As it follows from (9), the orthogonality of the vectors of the current stage of the *d*-SVD means that there is the relation between vectors of the next stage of the *d*-SVD:

$$\sum_{i_0 \ldots i_{d-3}; i_{d-2}} \sum_{k_{d-2}} u_{i_0 \ldots i_{d-3}; k_{d-2}; k_{d-1}} s_{k_{d-2}; k_{d-1}} v_{i_{d-2}; k_{d-2}; k_{d-1}} \cdot \sum_{q_{d-2}} u_{i_0 \ldots i_{d-3}; q_{d-2}; q_{d-1}} s_{q_{d-2}; q_{d-1}} v_{i_{d-2}; q_{d-2}; q_{d-1}} = \delta_{k_{d-1}; q_{d-1}} \tag{14}$$

where $\delta_{k_{d-1}; q_{d-1}}$ is the delta function. It can be noted that vectors $u_{i_0 \ldots i_{d-3}; k_{d-2}; k_{d-1}}$ and $u_{i_0 \ldots i_{d-3}; q_{d-2}; q_{d-1}}$, $v_{i_{d-2}; k_{d-2}; k_{d-1}}$ and $v_{i_{d-2}; q_{d-2}; q_{d-1}}$ are orthogonal on condition $k_{d-1} = q_{d-1}$. Thus, a network of mutually interconnected vectors from stage to stage is formed by *d*-SVD (10).

4. TENSOR OF STOCK DATA AND ITS *d*-SVD

An example of stock data of about 8,000 transactions were used to demonstrate dependencies analysis by the *d*-SVD. The records of the first ten transactions are shown in Table 1. The date and time of the transactions and their number within an hour were selected as influencing factors on five parameters. It was formed tensor (1) of size 17 × 9 × 60 × 5. The tensor is formed as sliced matrices of equal size. The matrices that are not completely filled can be supplemented by one of the existing rows or by one of the columns. Such filling will not change the structural characteristic of the matrix, such as its rank.

The tensor was reshaped as the matrix of size 765 × 60 and it was represented by the SVD as follows:

$$x_{i_0 i_1 i_3 ; i_2} = \sum_{k_2=0}^{K-1} s_{k_2} u_{i_0 i_1 i_3 ; k_2} v_{i_2 ; k_2} \tag{15}$$

where $K = 60$. The SVs distribution in (15) is shown in Figure 1a. As it follows from the figure, singular value $s_{k_2=0}$ is one hundred times greater than the next value. This allows to approximate the matrix of elements $x_{i_0 i_1 i_3 ; i_2}$ by its one-rank SP with elements

$$x^{(0_2)}_{i_0 i_1 i_3 ; i_2} = s_{0_2} u_{i_0 i_1 i_3 ; 0_2} v_{i_2 ; 0_2} \tag{16}$$

Table 1. *The records of first ten transactions*

Date	Time	Open	High	Low	Close	Volume
2.03.1997	9:04:00	3046.00	3048.50	3046.00	3047.50	505
2.03.1997	9:05:00	3047.00	3048.00	3046.00	3047.00	162
2.03.1997	9:06:00	3047.50	3048.00	3047.00	3047.50	98
2.03.1997	9:07:00	3047.50	3047.50	3047.00	3047.50	228
2.03.1997	9:08:00	3048.00	3048.00	3047.50	3048.00	136
2.03.1997	9:09:00	3048.00	3048.00	3046.50	3046.50	174
2.03.1997	9:10:00	3046.50	3046.50	3045.00	3045.00	134
2.03.1997	9:11:00	3045.50	3046.00	3044.00	3045.00	43
2.03.1997	9:12:00	3045.00	3045.50	3045.00	3045.00	214
2.03.1997	9:13:00	3045.50	3045.50	3045.50	3045.50	8

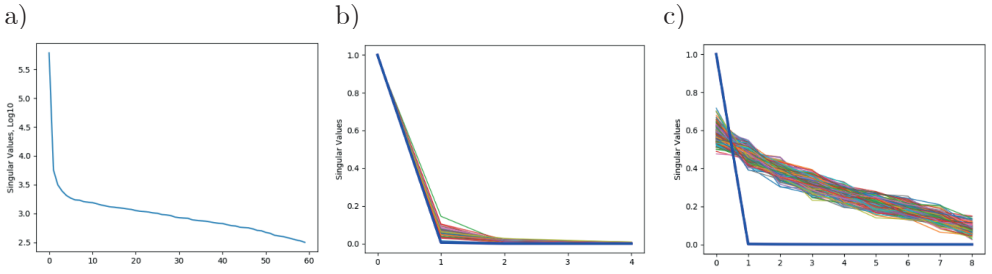

Fig. 1. *SVs distribution: a)* $s_{k_2=0\ldots59}$ *; b)* $s_{k_3=0\ldots4;k_2=0\ldots59}$ *; c)* $s_{k_1=0\ldots8;k_3=0\ldots4;k_2=0\ldots59}$

The SVD of the data tensor (15) was decomposed in accordance with (10) and it was obtained the following expression:

$$x_{i_0 ; i_1 ; i_2 ; i_3} = \sum_{k_2=0}^{59} s_{k_2} v_{i_2 ; k_2} \sum_{k_3=0}^{4} s_{k_3 ; k_2} v_{i_3 ; k_3 ; k_2} \sum_{k_1=0}^{8} u_{i_0 ; k_1 ; k_3 ; k_2} s_{k_1 ; k_3 ; k_2} v_{i_1 ; k_1 ; k_3 ; k_2} \tag{17}$$

An accuracy of 4-SVD (17) depends on the accuracy of the SVDs it uses and the tensor data volume. For example, the function ***numpy.linalg.svd()*** of the Python library has an error level ~10^{-14}, the error level of the tensor presentation (17) does not exceed ~10^{-12}.

The distributions of SVs $s_{k_3;k_2}$, $s_{k_1;k_3;k_2}$ in (17) are shown in Figure 1b and Figure 1c. Distributions $s_{k_3;0_2}$, $s_{k_1;0_3;0_2}$ are shown by a thick blue line. It is evident from the figures that these distributions inherit the shape of the distribution in Figure 1a. Therefore, it can be assumed that these SVs characterize the dependence of data on influencing factors, while the remaining SVs characterize the structure of the 4-SVD vector space. The one-rank approximation of data by the SP in the 4-SVD vector space

$$x^{(0_0;0_1;0_2;0_3)}_{i_0;i_1;i_2;i_3} = s_{0_2} v_{i_2;0_2} s_{0_3;0_2} v_{i_3;0_3;0_2} u_{i_0;0_1;0_3;0_2} s_{0_1;0_3;0_2} v_{i_1;0_1;0_3;0_2} \tag{18}$$

The original data and one-rank SP (18) are presented at the top and lower positions in Figure 2. The shape of the one-rank approximation (16) has the form similar to (18). The relative approximation error of (16) is equal to 0.71%. The relative approximation error of (18) is equal to 0.75%. As it follows from Table 1, the data of the first four parameters are close in value, and therefore their graphs in Figure 2 overlap. Their relative changes are small compared to the data "VOLUME", they indicate the absence of distortions caused by one-rank approximations (16) and (18). However, the one-rank approximation is not able to reflect the dynamics of the data "VOLUME", only their trend.

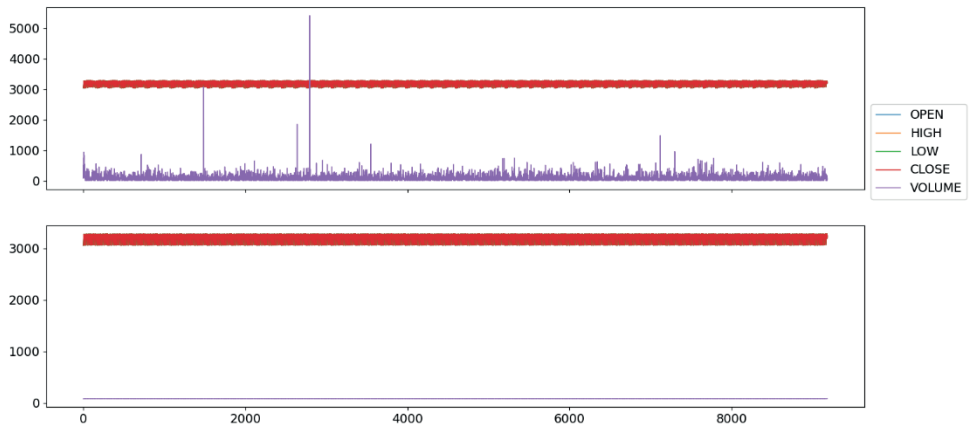

Fig. 2. *Original data (top), one-rank approximation by 4-SVD (18) (lower)*

5. DEPENDENCIES ANALYSIS WITH THE HELP OF SVs DISTRIBUTION ENTROPY

The problem is to select a minimal number of most significant components in (17) which ensure minimal data loss. This can be represented as the task to minimize the functional (Formula (19)).

$$\underset{K,K_{k_2},K_{k_3;k_2}}{minarg} \sum_{i_0;i_1;i_2;i_3} \left(x_{i_0;i_1;i_2;i_3} - \sum_{k_2=0}^{K-1} s_{k_2} v_{i_2;k_2} \sum_{k_3=0}^{K_{k_2}-1} s_{k_3;k_2} v_{i_3;k_3;k_2} \cdot \sum_{k_1=0}^{K_{k_3;k_2}-1} u_{i_0;k_1;k_3;k_2} s_{k_1;k_3;k_2} v_{i_1;k_1;k_3;k_2} \right)^2 \quad (19)$$

This problem does not have a direct solution and it may be solved with the account that SP (19) differs from the usual truncation of the SVs in decompositions like (2), (5) and (8) using a chosen threshold level. As it follows from (10), SVs s_{k_2} depend directly on the initial data, the SVs of subsequent stages of the *d*-SVD correspond to the structure of the vector space of the previous stage of the decomposition. As it follows from Figures 1b, and 1c, they are not small in value. Therefore, additional selection criteria are needed for the selection of the SVs and the corresponding vectors. The vectors of the SVD of successive stages of the decomposition are related by expression (14). Therefore, the significance of the vectors is transferred from stage to stage. The entropy of the SVs distribution of the vectors' decomposition can be used to assess the significance of the vectors in *d*-SVD (19).

The entropy of a distribution of a series s_k is calculated using normalized values $\overline{s}_k = s_k \left(\sum_i s_i\right)^{-1}$ (Caraiani, 2014; 2018) as

$$H\left(\overline{s}\right) = -\sum_k \overline{s}_k \ln(\overline{s}_k) \quad (20)$$

The entropy (20) of the distribution s_{k_2} in Figure 1a is 0.6661, but this value in itself does not give any information. However, the entropies of the SVs $s_{k_3;k_2}$ in (17) of the vectors $u_{i_0 i_1 i_3;k_2}$ in (15) show in Figure 3a that three first vectors have lower entropy compared to the rest. Therefore, it can be supposed that these vectors represent significant dependencies. Note, the distributions of the SVs corresponding to the second and third vectors are not highlighted in Figure 1b in the comparison with the rest of the vectors. So, the threshold value can be defined as $\varepsilon \geq s_{k_2=2} / s_{k_2=0}$ and in the considering case $\varepsilon \approx 0.005$.

a) b)

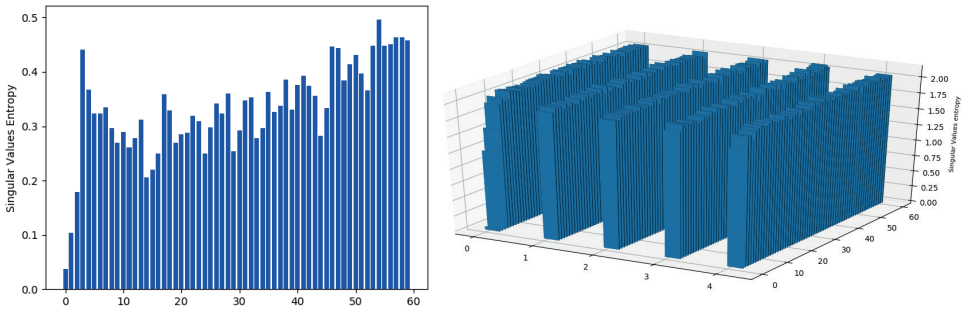

Fig. 3. *The entropy of the SVs distributions: a)* $s_{k_3;k_2=0...59}$ *; b)* $s_{k_1;k_3=0...4;k_2=0...59}$

The distributions of SVs $s_{k_1;k_3;k_2}$ entropy is shown in Figure 3b. As it follows from Figure 3, the lower entropy have the first three distributions of $s_{k_1;k_3=0;k_2=0,1,2}$. The

threshold value and selection of components with minimal entropy determine the following parameters of the sums in (19):

$$K = 3\,;\ K_{k_2} = [2,2,2]\,;\ K_{k_3;k_2} = [[3,1],[1,1],[1,1]] \tag{21}$$

Thus, the number of components of the sums 3060 in (17) is reduced to 17.

The data low-rank approximation by the SVD of $K = 3$ components in (15) is shown in Figure 4 at the second from the top position, the relative error is 0.62%. The low-rank approximation with parameters (21) corresponds to the approximation by the Tucker-type decomposition (4) with the core tensor (3) of size $1 \times 2 \times 3 \times 3$, that includes 18 components. This decomposition can be implemented by the function ***tucker()*** of the Python library ***tensorly***. The shape of the approximation by this function is shown at the third from the top position in Figure 4. The relative error is 0.72%. The approximation by 17 components of the 4-SVD in (19) that are given by (21) is shown at the lower position in Figure 4, the relative error is 0.64%. As can be seen from the figure, the selected significant components of the 4-SVD by a size equal to the size of the tensor dimensions preserved the most significant anomalous deviations of the data, which are lost in the case of the usual SVD (15) by the size of the unfolded tensor and in the case of the Tucker decomposition.

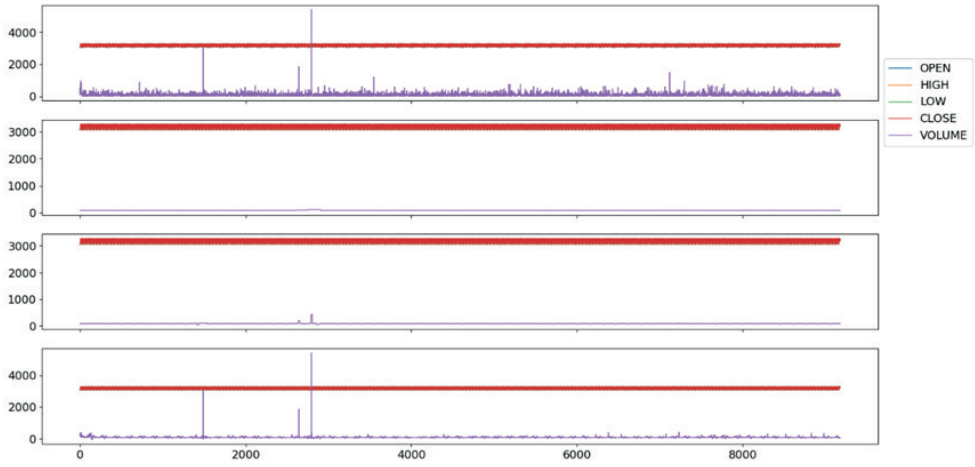

Fig. 4. *Original data (top), low-rank approximation by SVD (15) with $K = 3$ (the second from top), by 18 components of* ***tensorly tucker()*** *(the third from top) and by 4-SVD (19) of 17 components (lower)*

6. TENSOR PROJECTIONS INTO THE SPACE OF SIGNIFICANT VECTORS

The problem of data dependency analysis can be considered from the point of view of eliminating some dependencies by excluding the corresponding dimensions of the data tensor. In common practice, this is done by averaging with respect to these di-

mensions. The averaged data can serve as a pattern of array characteristics and are used to recognize the pattern in other similar data.

The dimension reduction consists of a projection of a multidimensional data into a vector space of a smaller dimension. The projection preserves the data's properties if the vector space of the data and the subspace of the projection are mutually related. The SVs in (10) can be joined into the core tensor of the elements

$$\tilde{c}_{k_1;\ldots;k_{d-1}} = s_{k_{d-1}} s_{k_{d-2};k_{d-1}} \ldots s_{k_1;k_2;\ldots;k_{d-1}} \tag{22}$$

It has a factorized form, unlike core tensor (3). As it follows from relation (14) of the vectors, a part of (22) can be used to evaluate a Tensor Projection (TP) of the tensor elements into a part of vector space, for example, into the vector space given by three vectors:

$$\tilde{x}_{i_{d-3};i_{d-2};i_{d-1}} = \alpha \sum_{k_{d-1};k_{d-2};k_{d-3}} \tilde{c}_{k_{d-3};k_{d-2};k_{d-1}} v_{i_{d-1};k_{d-1}} v_{i_{d-2};k_{d-2};k_{d-1}} v_{i_{d-3};k_{d-3};k_{d-2};k_{d-1}} \tag{23}$$

where $\tilde{c}_{k_{d-3};k_{d-2};k_{d-1}} = s_{k_{d-1}} s_{k_{d-2};k_{d-1}} s_{k_{d-3};k_{d-2};k_{d-1}}$, α is a normalizing coefficient, which is defined with accuracy up to sign as relation of norms to compensate for the absence of a part of the SVs. For the considering case (23)

$$\alpha = \frac{\left\| \tilde{c}_{k_1;\ldots;k_{d-1}} \right\|}{\left\| \tilde{c}_{k_{d-3};k_{d-2};k_{d-1}} \right\|} \tag{24}$$

The core tensor in (23) accumulates properties of all data on which SVs were computed in tensor (22). Since the sequence of dimensions in decomposition (10) can be arbitrary, as it was shown in example (17), and is determined by reshaping of an original list of data, it is possible to choose the such sequence that (23) will reflect significant dependencies. When the tensor is composed of mean-centered data, the covariance of the projection (23) with the data tensor is the following

$$C_{i_0;\ldots;i_{d-4}} = \sum_{i_{d-3};i_{d-2};i_{d-1}} \tilde{x}_{i_{d-3};i_{d-2};i_{d-3}} x_{i_0;\ldots;i_{d-1}} = \alpha \sum_{k_{d-1}} s^2_{k_{d-1}} \sum_{k_{d-2}} s^2_{k_{d-2};k_{d-1}} \sum_{k_{d-3}} s^2_{k_{d-3};k_{d-2};k_{d-1}} \cdot$$
$$\sum_{k_{d-4}} s_{k_{d-4};\ldots;k_{d-1}} v_{i_{d-4};k_{d-4};\ldots;k_{d-1}} \ldots \sum_{k_1} s_{k_1;\ldots;k_{d-1}} u_{i_0;k_1;\ldots;k_{d-1}} v_{i_1;k_1;\ldots;k_{d-1}} \tag{25}$$

Covariance (25) contains the constant part in the space of the three selected vectors due to the orthogonality (14) of the vectors in the sum by i_{d-1}, this orthogonality selects orthogonal vectors in the second sum by i_{d-2} and so on. The variable part of (25) shows changes of the TP in the selected vector space in respect to the rest vector space. If the vector space of the significant dependencies was selected for projection (23) correctly, the changes in the remaining vector space are weak and (25) is quasi-constant. Otherwise, it shows that the selected significant vector space is either erroneous or incomplete.

In the case of the example of stock data considered above, let us present the data in the second and third dimensions as related to the significant dependencies, neglecting the dependencies on date and hour, the zero and first dimensions, as trivial.

Such a presentation can be obtained by averaging the tensor on the zero and first dimensions. Then the elements of the averaged projection are as follows.

$$\overline{x}_{i_2;i_3} = \frac{1}{17 \cdot 9} \sum_{i_0=0}^{16} \sum_{i_1=0}^{8} x_{i_0;i_1;i_2;i_3} \tag{26}$$

They are shown in Figure 5 (top position). As an alternative to averaging (26), the elements of the TP in the second and third dimensions, like (23), using decomposition (17), are the following.

$$\tilde{x}_{i_2;i_3} = \alpha \sum_{k_2=0}^{59} s_{k_2} v_{i_2;k_2} \sum_{k_3=0}^{4} s_{k_3;k_2} v_{i_3;k_3;k_2} \tag{27}$$

The projection's data are shown in Figure 5 (middle position), where coefficient (24) $\alpha = 0.0804$. The elements of the SP as the sum of the most significant vectors in (27) in accordance with (21)

$$\hat{\tilde{x}}_{i_2;i_3} = \alpha \sum_{k_2=0}^{2} s_{k_2} v_{i_2;k_2} \sum_{k_3=0}^{1} s_{k_3;k_2} v_{i_3;k_3;k_2} \tag{28}$$

are shown in the lower position, $\alpha = 0.0808$.

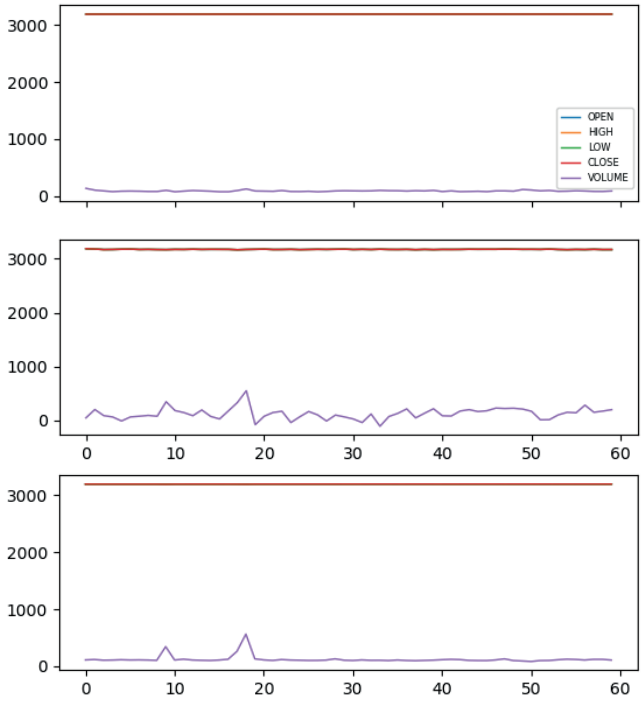

Fig. 5. *Averaged data on date and hour(26) (top), projection onto vectors of the second and third dimensions (27) (middle), projection onto significant components (28) (lower)*

As it follows from Figure 5, the averaging in (26) blurs and hides significant variations of the initial data, while projections (27) and (28) save information about them and clearly indicate their presence, especially in the case (28) of the presentation in the space of the significant vectors.

Covariance (25) was evaluated for the projections (26)–(28) in the form

$$C_{i_0;i_1} = \sum_{i_2=0}^{59}\sum_{i_3=0}^{4}\left(y_{i_2;i_3} - \overline{y}_{i_3}\right)\left(x_{i_0;i_1;i_2;i_3} - \overline{x}_{i_3}\right) \tag{29}$$

where $y_{i_2;i_3}$ is equal to one of (26)–(28), $\overline{y}_{i_3}$ and $\overline{x}_{i_3}$ are mean values with respect to the third dimension. The normalized on magnitude covariances (29) are presented in Figure 6. The figure shows that the covariance values of the data tensor with the averaged projection (26) have significant volatility, its mean and variation are 0.0867 ±0.0448 while the covariance values with the projections based on the *d*-SVD (27), (28) are close in amplitude, with the exception of two anomalous cases, their means and variations are 0.1047 ±0.0299, 0.0572 ±0.0250. This shows that the dependence of the data on the first two factors is primarily weak.

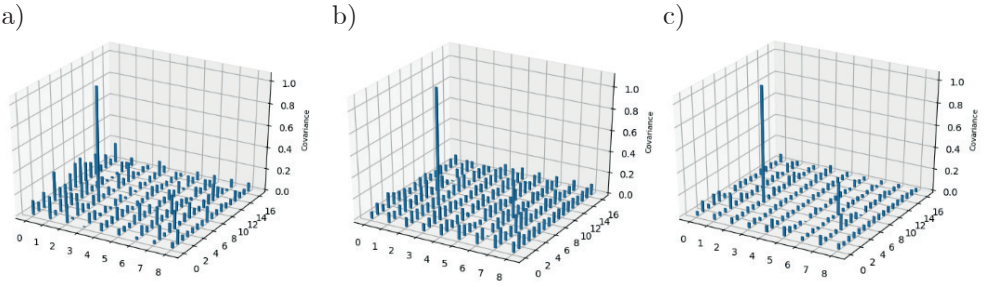

Fig. 6. *Normalized covariances (29) of data tensor with: a) averaged projection (26); b) projection (27); c) projection (28)*

As it follows from Figures 5 and 6, the projections and covariance (29) allow to localize and visualize abnormal situations in the data relative to the first two dimensions. The result obtained using projections (27) and (28) is clearer, without blurring and mutual leakage of the covariance samples, compared to the result based on averaging (26).

7. PREDICTION IN THE SIGNIFICANT DEPENDENCIES SUBSPACE

A key task in financial and product markets is to predict the expected results of an upcoming transaction in order to decide on participation and what investments to make to obtain the expected profit. There are many approaches to solving this task (Caraiani, 2014; Dixon et al., 2020). The SVD is used to eliminate random variations in analyzing data to get a stable model of the data dynamics with the appropriate accuracy of forecasting. The SVD is used to select a significant part of the data and to use it for creating a prediction model. There are two ways to implement data filtration

by the SVD. The usual way consists in decomposing the total data set into components and selecting the most significant part for the model. It may be the second way, within the framework of the article under consideration, that multidimensional data with dimensions on factors of influence are filtered in the selected subspace of significant dependencies to avoid loss of data features which can influent on the model quality. In the case considered, the investigation of the influence of the selected subspace of significant dependencies on the accuracy of the prediction is the subject of interest.

As a subject of the investigation, the linear autoregressive (AR) model can be chosen as a widely used tool for data prediction in various applied problems (Marple, 1987). The AR model of the order P for the data set $x_{0,\ldots,N-1}$ ($N >> P$) can be developed with the help of the following functional.

$$\underset{a}{minarg} \sum_{i=0}^{N-P}\left(\sum_{k=0}^{P-1} a_k x_{P-1+i-k}\right)^2 \tag{30}$$

where $a_{k=0\ldots P-1}$ are the model coefficients. The functional is formed on condition to minimize the prediction moving error

$$\xi_k = \sum_{m=0}^{P-1} a_m x_{P-1+k-m} \tag{31}$$

along the data set. The condition will be met when coefficients a_k in (30) are elements of the eigenvector corresponding to minimal eigenvalue of the matrix of size $P \times P$ with elements

$$r_{i;k} = \sum_{m=0}^{N-P} x_{P-1-i+m} x_{P-1-k+m} : i,k = 0\ldots P-1 \tag{32}$$

As the matrix of elements (32) is symmetrical, the vector of the SVD of this matrix, which corresponds to the smallest singular value, can be used as the solution of problem (30).

Table 2. *The prediction errors mean and variation*

Prediction model order	$P = 7$	$P = 15$	$P = 23$	$P = 33$
original data "OPEN"	0.31 ±4.01	0.36 ±8.47	0.34 ±4.66	0.50 ±8.27
approximation by SVD with $K = 3$	0.27 ±1.03	0.26 ±4.65	0.26 ±10.9	0.26 ±22.4
approximation by (19) with (21)	0.21 ±1.87	0.22 ±5.80	0.23 ±3.50	0.24 ±7.58

The AR model was created for the array of the parameter "OPEN" (Table 2). The model error depends on the model order. Therefore, models of four order variants were investigated. The results of the prediction of the array elements

$$\tilde{x}_{P-1+k} = -a_0^{-1} \sum_{m=1}^{P-1} a_m x_{P-1+k-m} : k = 0,\ldots,N-P \tag{33}$$

with the help of the AR model are shown in Table 2 and in Figure 7.

The means and variations of errors (31) of the AR models of four orders of the original array are presented in Table 2. Also, the means and variations of the errors of the prediction of the data obtained by truncated SVD (15) with $K = 3$ and d-SVD projection (19) with parameters (21) are presented. The original data array and predicted values (33) by the AR model of the order $P = 23$ are shown in the top position of Figure 7. The data of truncated SVD (15) and SP (19), their predicted values are presented at the middle and lower positions of Figure 7.

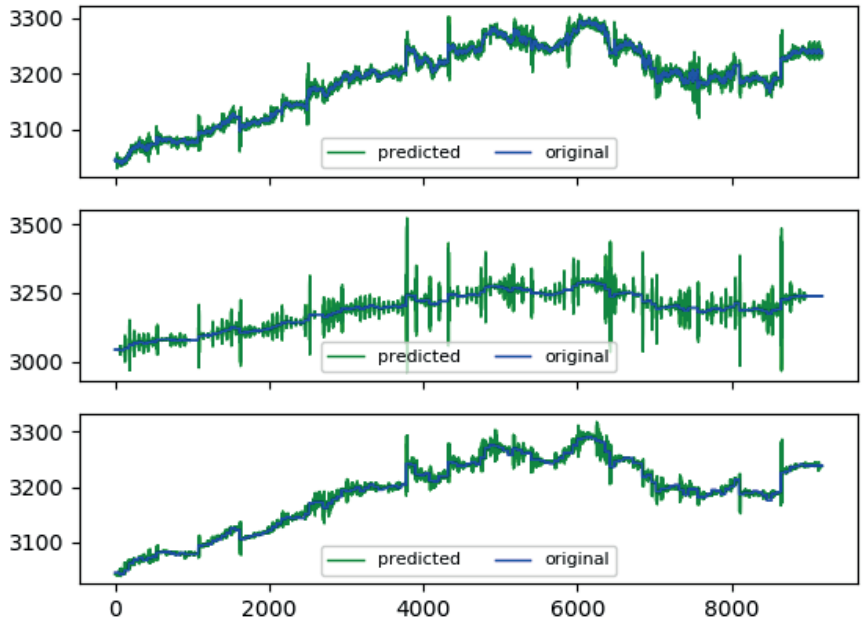

Fig. 7. *Prediction of data "OPEN" by the AR model of the order $P = 23$: original data (top); filtered data by the truncated SVD (15) with $K = 3$ (middle); filtered data by the SP (19) with parameters (21) (lower)*

As it follows from Table 2 and Figure 7, the prediction of the original data has maximal error because the data include variations like noise. The truncated SVD (15) and SP (19) in the significant dependencies vector subspace reduce the level of noise-like variations and allow the prediction error to be reduced. However, the error variation is greatest for higher orders of the AR model in the case of truncated SVD (15). At the same time, the SP in vector space of the d-SVD is characterized by the lowest level of the error and its variation.

Significant prediction errors arise when the data change sharply on some dimensions. The truncated SVD exacerbates these changes. Additional components of the d-SVD with respect to these dimensions smooth out these changes. As a result, the prediction in the d-SVD vector subspace is more stable and can provide a lower probability of errors in the decision-making based on the forecasting.

8. CONCLUSIONS

As shown in the analysis of data by their tensorization (Cichocki et al., 2017), a large data list can be reshaped as a multidimensional tensor of small and medium sizes according to the dimensions of the influencing factors. This allows users to highlight dependencies of the data on the factors.

A *d*-SVD schema of the HOSVD was presented that differs from known ones by the factorized presentation of the core tensor in Tucker type tensor decomposition. The schema shows not only that singular values should be used to separate significant and trivial vectors related to dependencies on factors of influence, but also that the vectors structure is important for this. Some vectors are close to the data structure and play an important role in the data presentation, despite their SVs being small in magnitude, while the rest of the vectors are just an orthogonal complement to them. The entropy of the SVs distribution of the vectors reshaped as matrices and decomposed by the SVD can indicate the significant vectors of the *d*-SVD.

The presentation of the tensor in the significant vector space of the selected dimensions of the *d*-SVD is considered as the Singular Projection. As follows from the results of the numerical experiment presented in Figure 4, the SP retains data features that can be lost in low-rank approximation by the commonly used truncated SVD and library function.

It was shown that the data tensor can be approximated by the one-rank tensor in the *d*-SVD vector space of size equal to the dimensions' size with an error close to the error of the one-rank representation in the vector space of the unfolded tensor. The one-rank presentation gives a decomposition by dimensions of the main factors. A sum of the most significant one-rank components of the *d*-SVD allows to define the canonic polyadic decomposition of the tensor (De Lathauwer et al., 2000).

The factorized form (22) of the core tensor of the *d*-SVD allows the Tensor Projection to be defined and which reduces the dimensionality of the data tensor by selecting the significant part of the core tensor and the corresponding to it vector subspace. The results presented in Figures 5 and 6 show that the TP can serve as a more objective pattern of an analyzed data compared to the ordinary averaging and it can be used as a data feature to solve decision-making problems related to object recognition.

The SP can be used in the problems of denoising and smoothing out sharp changes in data dynamics to improve the accuracy of forecasting in the decision-making process. It can have an advantage over the ordinary truncated SVD.

As it was shown by the offline modeling (Sofina & Bunyak, 2021), the selection of significant factors on an intuitive level, from which the results of online advertising (ad) bidding on websites depend, and their account in the decision-making process to participate in a closed auction with the selection of an appropriate price can reduce the ad price by up to 3–4 times and to make the ad profitable. The implementation of the factors selection using the SP and TP can improve the efficiency of solving such problems.

Further development of the proposed approach to data analysis involves exploring the tensorization of big data as structures reflecting dependencies on multiple

factors, followed by identifying the most significant dependencies using the SP and obtaining a compressed data pattern as the TP into the subspace of the most significant factors. The resulting pattern can be used for decision-making in various artificial intelligence applications as an alternative or as a complement to known statistical characteristics.

REFERENCES

Alvarez-Ramirez J. & Rodriguez E. (2021). A singular value decomposition entropy approach for testing stock market efficiency. *Physica A: Statistical Mechanics and its Applications*, **583**, 126337. **DOI:** https://doi.org/10.1016/j.physa.2021.126337.

Bergqvist G. & Larsson E.G. (2010). The higher-order singular value decomposition: Theory and an application. *IEEE Signal Processing Magazine*, **27**(3), pp. 151–154. **DOI:** https://doi.org/10.1109/MSP.2010.936030.

Caraiani P. (2014). The predictive power of singular value decomposition entropy for stock market dynamics. *Physica A: Statistical Mechanics and its Applications*, **393**, pp. 571–578. **DOI:** https://doi.org/10.1016/j.physa.2013.08.071.

Caraiani P. (2018). Modeling the comovement of entropy between financial markets. *Entropy*, **20**(6), 417. **DOI:** https://doi.org/10.3390/e20060417.

Cichocki A., Lee N., Oseledets I., Phan A.-H., Zhao Q. & Mandic D.P. (2016). Tensor networks for dimensionality reduction and large-scale optimization: Part 1. Low-rank tensor decompositions. *Foundations and Trends in Machine Learning*, **9**(4–5), pp. 249–429. **DOI:** https://doi.org/10.1561/2200000059.

Cichocki A., Phan A.-H., Zhao Q., Lee N., Oseledets I. & Mandic D.P. (2017). Tensor networks for dimensionality reduction and large-scale optimization: Part 2. Applications and future perspectives. *Foundations and Trends in Machine Learning*, **9**(6), pp. 431–673. **DOI:** https://doi.org/10.1561/2200000067.

Cheng D., Liu Y., Niu Z. & Zhang L. (2018). Modeling similarities among multidimensional financial time series. *IEEE Access*, **6**, pp. 43404–43413. **DOI:** https://doi.org/10.1109/ACCESS.2018.2862908.

Comon P. (2002). Tensor decompositions: State of the art and applications. In: McWhirter J.G. & Proudler I.K. (Eds.), *Mathematics in Signal Processing*. Oxford, UK: Clarendon Press, pp. 1–24. **DOI:** https://doi.org/10.1093/oso/9780198507345.003.0001.

Czapkiewicz A. & Skalna I. (2014). Selected approaches for testing asset pricing models using polish stock market data. *Decision Making in Manufacturing and Services*, **8**(1–2), pp. 25–38. **DOI:** https://doi.org/10.7494/dmms.2014.8.1.25.

De Lathauwer L., De Moor B. & Vandewalle J. (2000). A multilinear singular value decomposition. *SIAM Journal on Matrix Analysis and Applications*, **21**(4), pp. 1253–1278. **DOI:** https://doi.org/10.1137/S0895479896305696.

Dixon M.F., Halperin I. & Bilokon P. (2021). Machine learning in finance: From theory to practice. Cham: Springer International Publishing. **DOI**: https://doi.org/10.1007/978-3-030-41068-1.

Ganeshapillai G., Guttag J. & Lo A. (2013). Learning connections in financial time series. *Proceedings of the 30th International Conference on Machine Learning*, *PMLR*, **28**(2), pp. 109–117.

Grasedyck L. (2010). Hierarchical singular value decomposition of tensors. SIAM Journal on Matrix *Analysis and Applications*, **31**(4), pp. 2029–2054. **DOI**: https://doi.org/10.1137/090764189.

Kennet D.Y., Raddant M., Zatlavi L., Lux T. & Ben-Jacob E. (2012). Correlation and dependencies in the global financial village. *International Journal of Modern Physics: Conference Series*, **16**, pp. 13–28. **DOI**: https://doi.org/10.1142/S201019451200774X.

Marple Jr. S.L. (1987). *Digital Spectral Analysis with Applications*. Englewood Cliffs: Prentice Hall.

Montagnon C.E. (2020). Forecasting by splitting a time series using singular value decomposition then using both ARMA and a Fokker Planck equation. *Physica A: Statistical Mechanics and its Applications*, **567**, 125708. **DOI:** https://doi.org/10.1016/j.physa.2020.125708.

Oseledets I.V. (2011). Tensor-train decomposition. *SIAM Journal on Scientific Computing*, **33**(5), pp. 2295–2317. **DOI:** https://doi.org/10.1137/090752286.

Phan A.-H. & Cichocki A. (2010). Tensor decompositions for feature extraction and classification of high dimensional datasets. *Nonlinear Theory and Its Applications, IEICE*, **1**(1), pp. 37–68. **DOI:** https://doi.org/10.1587/nolta.1.37.

Savas V. & Eldén V. (2007). Handwritten digit classification using higher-order singular value decomposition. *Pattern Recognition*, **40**(3), pp. 993–1003. **DOI:** https://doi.org/10.1016/j.patcog.2006.08.004.

Sofina O. & Bunyak Y. (2021). Internet bidding optimization by targeted advertising based on self learning database. In: Tatomyr I. & Kvasnii Z., *The economics of postpandemics: prospects and challenges*. Praha: OKTAN PRINT, pp. 123–139.

Spentzouris P., Koutsopoulos I., Madsen K.G & Hansen T.V. (2018). Advertiser bidding prediction and optimization in online advertising. In: Iliadis L., Maglogiannis I. & Plagianakos V. (Eds.), *Artificial Intelligence Applications and Innovations. 14th IFIP WG 12.5 International Conference, AIAI 2018, Rhodes, Greece, May 25–27, 2018. Proceedings*, pp. 413–424. **DOI:** https://doi.org/10.1007/978-3-319-92007-8_35.

Tucker L.R. (1966). Some mathematical notes on three mode factor analysis. *Psychometrika*, **31**(3), pp. 279–311. **DOI:** https://doi.org/10.1007/BF02289464.

Vannieuwenhoven N., Vandebril R. & Meerbergen K. (2012). A new truncation strategy for the higher-order singular values decomposition. *SIAM Journal on Scientific Computing*, **34**(2), pp. A1027–A1052. **DOI:** https://doi.org/10.1137/110836067.

Zhigljavsky A. (2010). Singular Spectrum Analysis for time series: Introduction to this special issue. *Statistics and Its Interface*, **3**(3), pp. 255–258. **DOI:** https://doi.org/10.4310/SII.2010.v3.n3.a1.

Zniyed Y., Boyer R., de Almeida A.L.F. & Favier G. (2020). A TT-based hierarchical framework for decomposing high-order tensors. *SIAM Journal on Scientific Computing,* **42**(2), pp. A822–A848. **DOI:** https://doi.org/10.1137/18M1229973.