Information Sciences 629 (2023) 471–487

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Avoiding flatness in factoring ordinal data

Eduard Bartl *, Radim Belohlavek

Dept. Computer Science, Palacký University, Czech Republic

ARTICLE INFO

Keywords:
Factor analysis
Ordinal attributes
Fuzzy logic
Quality of factors

ABSTRACT

Factorization of classical, two-valued Boolean data became a widely studied topic in the past decade due to its role in analyzing relational data as well as its significance for other fields. Recently, various extensions to factorization of ordinal data, or data with graded (fuzzy) attributes, have been proposed. We identify and describe a fundamental problem regarding quality of factors, which is non-existent in the Boolean case, but naturally appears in the more general setting of ordinal data. As we demonstrate, the problem gets more significant with growing size of the factorized data. We analyze the problem, propose a method to alleviate it, and evaluate experimentally our solution to the problem. We also provide a discussion regarding ramifications of our findings for the concept of cardinality of fuzzy sets.

1. Introduction to factorization of ordinal data

1.1. Basic factorization problem

A factorization problem, which we consider and which subsumes the well known factorization of Boolean matrices, may be described as follows. Consider an $n \times m$ matrix I whose entries I_{ij}, for $1 \le i \le n$ and $1 \le j \le m$, are elements of an ordered scale L; in the basic interpretation, the entry I_{ij} at row i and column j represents a degree to which the object i has the attribute j. In particular, we assume that the degrees form a complete lattice $\langle L, \le, 0, 1\rangle$, i.e. a partially ordered set bounded by 0 and 1 in which arbitrary suprema $\bigvee$ and infima $\bigwedge$ exist, equipped with additional operations as explained below. The set of all such matrices shall be denoted $L^{n\times m}$.

The goal is to factorize I, i.e. to find a decomposition of the object-attribute matrix I into an object-factor matrix $A \in L^{n\times k}$ and a factor-attribute matrix $B \in L^{k\times m}$ such that

$$k \text{ is small and } I \approx A \circ B, \qquad (1)$$

where $I \approx A \circ B$ denotes that I is approximately equal to $A \circ B$ and $\circ$ denotes the sup-$\otimes$-composition (product) of matrices, i.e.

$$(A \circ B)_{ij} = \bigvee_{l=1}^{k} A_{il} \otimes B_{lj}. \qquad (2)$$

In order to focus on the phenomenon we address, we assume that the lattice L of degrees is a chain included in the real unit interval $[0,1]$ (i.e. $L \subseteq [0,1]$) and that $\otimes$ is commutative, associative, isotone, and satisfies $a \otimes 1 = a$ for each $a \in L$.

* Corresponding author.
E-mail addresses: eduard.bartl@upol.cz (E. Bartl), radim.belohlavek@acm.org (R. Belohlavek).

https://doi.org/10.1016/j.ins.2023.02.002
Received 24 June 2022; Received in revised form 16 November 2022; Accepted 1 February 2023

Available online 10 February 2023
0020-0255/

Remark 1.

(a) For $L = \{0, 1\}$, the present factorization becomes the well-known factorization of Boolean (also called binary) matrices [5,13,15], as $\otimes$ then becomes the function of logical conjunction and (2) becomes the max-min product of Boolean matrices.
(b) For $L = [0, 1]$ or $L = \{0, 1/n, \ldots, n-1/n, 1\}$, $\otimes$ becomes a t-norm and we obtain a factorization of matrices with graded (fuzzy) attributes [2–4].
(c) Two particular instances of the factorization problem are considered in the literature: The first is the discrete basis problem (DBP), in which a number k is given and the problem is to find k factors such that I is as similar to $A \circ B$ as possible [4,15]. The second is the approximate factorization problem (AFP), in which an $\varepsilon > 0$ is given and the problem is to find as few factors as possible for which the similarity of I and $A \circ B$ is at least ε [3,5].
(d) A more general factor model has recently been proposed in [9]. In this paper, the authors also consider a semantic problem related in a broader sense to the problem addressed in our paper. We comment on this in more detail in section 5.

1.2. Matrix similarity

To assess approximate equality (similarity, closeness) of matrices I and $A \circ B$, one naturally employs [3,4] the function

$$S(I, A \circ B) = \sum_{i,j=1}^{n,m} (I_{ij} \leftrightarrow (A \circ B)_{ij}) \tag{3}$$

or its normalized version

$$s(I, A \circ B) = \frac{\sum_{i,j=1}^{n,m} (I_{ij} \leftrightarrow (A \circ B)_{ij})}{n \cdot m}, \tag{4}$$

where $\leftrightarrow$ is a suitable function of many-valued logical equivalence. Starting from $\otimes$, cf. (2), a natural, logically well-behaving option is to take the biresiduum $\leftrightarrow$, which is defined by $a \leftrightarrow b = (a \rightarrow b) \wedge (b \rightarrow a)$, where $\rightarrow$ is the so-called residuum induced by $\otimes$ [10,11]. For instance, if $\otimes$ is the Łukasiewicz t-norm, given by $a \otimes b = \max(0, a + b - 1)$, the residuum is the well-known Łukasiewicz implication $a \rightarrow b = \min(1, 1 - a + b)$, in which case the corresponding $\leftrightarrow$ is given by $a \leftrightarrow b = 1 - |a - b|$.

Remark 2. In the Boolean case, i.e. when $L = \{0, 1\}$, it is a common practice to utilize a distance function rather than a similarity function. For this purpose, one employs [5,13,15] the matrix metric E based on the L_1-norm (equivalently, the Hamming distance), i.e.

$$E(I, A \circ B) = \sum_{i,j=1}^{n,m} |I_{ij} - (A \circ B)_{ij}|,$$

or its normalized version

$$e(I, A \circ B) = \frac{E(I, A \circ B)}{n \cdot m}.$$

In the Boolean case, one may choose whether to proceed with similarity, cf. (3) and (4), or the above-described distance, because when denoting by $\leftrightarrow$ the classical logical equivalence, one has

$$S(I, A \circ B) = n \cdot m - E(I, A \circ B) \text{ and}$$

$$s(I, A \circ B) = 1 - e(I, A \circ B).$$

1.3. Interpretation of factors

Due to the properties of the above model and the employed functions $\otimes$ and $\bigvee$, factors may naturally be interpreted and visualized. Namely, the lth factor ($l = 1, \ldots, k$) in the decomposition (1) may be identified with the pair $\langle A_{\_l}, B_{l\_} \rangle$ consisting of the lth column $A_{\_l}$ of A and the lth row $B_{l\_}$ of B. The degrees A_{il} and B_{lj} comprising factor l are naturally interpreted as the degree to which factor l applies to the object i (or, object i possesses factor l) and the degree to which the attribute j is a particular manifestation of factor l, respectively.

Moreover, the lth factor $\langle A_{\_l}, B_{l\_} \rangle$ may be visualized as the crossproduct $A_{\_l} \circ B_{l\_}$, i.e. an $n \times m$ matrix defined by

$$(A_{\_l} \circ B_{l\_})_{ij} = A_{il} \otimes B_{lj}. \tag{5}$$

As we shall see, such matrices represent certain rectangular patterns and since (2) may be rewritten as

$$A \circ B \quad = \quad A_{\_1} \circ B_{1\_} \ \vee \ \cdots \ \vee \ A_{\_k} \circ B_{k\_},$$

an approximate decomposition of I into $A \otimes B$ in fact means that the given matrix I may approximately be expressed as a $\vee$-superposition of k rectangular patterns $A_{\_l} \circ B_{l\_}$. Note that in the Boolean case, this means that I is obtained as a max-superposition of k rectangles, each of which is full of 1s.

Example 1. Consider the five-element chain $L = \{0, 1/4, 1/2, 3/4, 1\}$ equipped with the Łukasiewicz operations, and the following 4×5 matrix:

$$I = \begin{pmatrix} 1/2 & 1 & 1/4 & 1 & 3/4 \\ 1 & 0 & 3/4 & 3/4 & 3/4 \\ 1/2 & 1 & 1 & 3/4 & 1 \\ 1/4 & 1/4 & 3/4 & 0 & 3/4 \end{pmatrix}. \tag{6}$$

One may check that $I = A \circ B$, where

$$A = \begin{pmatrix} 1/4 & 1 & 1 & 1/2 \\ 3/4 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1/2 \\ 3/4 & 1/4 & 1/4 & 1/4 \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} 1/2 & 1/4 & 1 & 1/4 & 1 \\ 1/2 & 1 & 1/4 & 3/4 & 3/4 \\ 1/2 & 1/4 & 1/4 & 1 & 3/4 \\ 1 & 0 & 3/4 & 3/4 & 3/4 \end{pmatrix}.$$

The individual factors $\langle A_{\_l}, B_{l\_} \rangle$ are represented by the columns of A and rows of B, respectively, and may be visualized by the corresponding crossproducts $A_{\_l} \circ B_{l\_}$ as follows:

$$A_{\_1} \circ B_{1\_} = \begin{pmatrix} 0 & 0 & 1/4 & 0 & 1/4 \\ 1/4 & 0 & 3/4 & 0 & 3/4 \\ 1/2 & 1/4 & 1 & 1/4 & 1 \\ 1/4 & 0 & 3/4 & 0 & 3/4 \end{pmatrix},$$

$$A_{\_2} \circ B_{2\_} = \begin{pmatrix} 1/2 & 1 & 1/4 & 3/4 & 3/4 \\ 0 & 0 & 0 & 0 & 0 \\ 1/2 & 1 & 1/4 & 3/4 & 3/4 \\ 0 & 1/4 & 0 & 0 & 0 \end{pmatrix},$$

$$A_{\_3} \circ B_{3\_} = \begin{pmatrix} 1/2 & 0 & 1/4 & 1/4 & 1/4 \\ 1 & 0 & 3/4 & 3/4 & 3/4 \\ 1/2 & 0 & 1/4 & 1/4 & 1/4 \\ 1/4 & 0 & 0 & 0 & 0 \end{pmatrix},$$

$$A_{\_4} \circ B_{4\_} = \begin{pmatrix} 1/2 & 1/4 & 1/4 & 1 & 3/4 \\ 1/4 & 0 & 0 & 3/4 & 1/2 \\ 1/4 & 0 & 0 & 3/4 & 1/2 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

Since $I = A \circ B$, we have

$$I = A_{\_1} \circ B_{1\_} \vee A_{\_2} \circ B_{2\_} \vee A_{\_3} \circ B_{3\_} \vee A_{\_4} \circ B_{4\_}$$

and $s(I, A \circ B) = 1$. When the first factor is considered and the remaining ones are dropped, one obtains an approximate decomposition of I into $A_{\_1} \circ B_{1\_}$, for which

$$s(I, A_{\_1} \circ B_{1\_}) = 0.7.$$

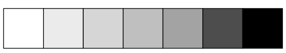
With two factors, the decomposition is considerably precise already since

$$s(I, A_{\_1} \circ B_{1\_} \vee A_{\_2} \circ B_{2\_}) = 0.91.$$

2. Flat factors and why they appear

We now present the phenomenon addressed in this paper. Note at the outset that, as shall become apparent, the phenomenon is non-existent in the two-valued Boolean case, i.e. when $L = \{0, 1\}$. In the multiple-valued case, the phenomenon appears on larger data, which is also where we observed it. In particular, we encountered this phenomenon when analyzing data from the British educational system; some of our findings are reported in section 4.

For convenience, we shall visualize matrices with degrees by arrays in which matrix entries are represented by shades of gray. In particular, we use the set L containing seven grades, 0, 1/6, 2/6, 3/6, 4/6, 5/6, and 1, which proved useful in analyzing ordinal data, because—as is well known—people are comfortable working with 7 ± 2-element scales [16]. Furthermore, we employ the Łukasiewicz t-norm and the corresponding biresiduum; see section 1.2. The correspondence of the degrees $a \in L$ to the shades of gray is as follows:

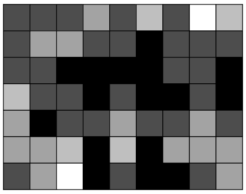


0 1/6 2/6 3/6 4/6 5/6 1

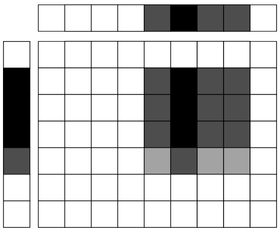
2.1. Desirable factors vs. flat factors

According to the reasoning adopted in factor analysis, a set of k factors, which are represented by the matrices A and B, approximately explains the input data I if the entries I_{ij} and $(A \circ B)_{ij}$ have the same or at least reasonably similar values for most entries $\langle i, j \rangle$. Typically, this is achieved in such a way that each factor $l = 1, \ldots, k$ significantly explains some large part of the input data I.

Example 2. Consider the following 7×9 matrix I:

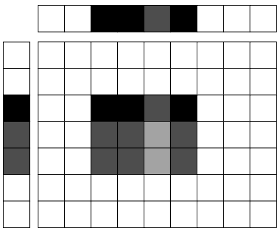


(a) Consider, furthermore, the following factor $F_1 = \langle C_1, D_1 \rangle$:



The parts C_1 and D_1 are depicted as the column (7×1 matrix) and the row (1×9 matrix) to the left and on the top of the 7×9 array, which itself represents the crossproduct $C_1 \circ D_1$. One may observe that factor $\langle C_1, D_1 \rangle$ explains reasonably well the part of the input matrix I corresponding to rows 2–5 and columns 5–8. Namely, the entries of this part of I have the same or similar values to those of the 4×4 non-white part of the 7×9 array displaying the factor.

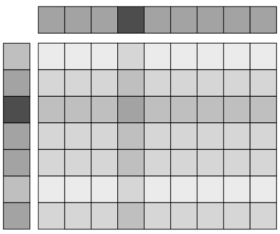
(b) Another good factor is the following factor $F_2 = \langle C_2, D_2 \rangle$:



Likewise, factor $\langle C_2, D_2 \rangle$ explains well the part of I that corresponds to the non-white entries of the crossproduct $C_2 \circ D_2$, i.e. the part of the input matrix I corresponding to rows 3–5 and columns 3–6.

Since both factor $\langle C_1, D_1 \rangle$ and factor $\langle C_2, D_2 \rangle$ significantly explain a reasonably large part of the input matrix I, they are considered natural and informative. Such factors are desirable and their discovery is the very purpose of factorization.

Example 3. It may, nevertheless, happen that a factor does not have the desirable property described above. Consider the following factor $F_3 = \langle C_3, D_3 \rangle$; intuitively, this factor does not significantly explain any reasonably large part of the input matrix I:



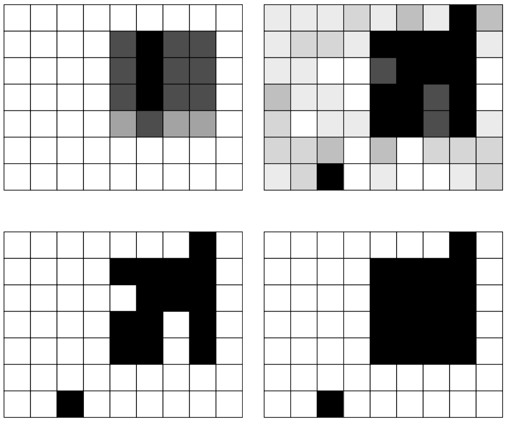


Fig. 1. Matrix $C_1 \circ D_1$ (top left), matrix $I \leftrightarrow C_1 \circ D_1$ of biresidua (top right), and matrices representing equality (bottom left) and at least 5/6-equality (bottom right) of entries in I and $C_1 \circ D_1$.

Factors like the one presented in Example 3 shall be called *flat*. These are factors $\langle C, D\rangle$ that explain to some small extent a large part of the input matrix I, but they do not explain any part of I significantly. That is, a large number of entries $(C \circ D)_{ij}$ have slightly similar values to the corresponding values I_{ij} in the input matrix I, but the similarities are not significant in any reasonably large part of I.

It might seem that flat factors would not appear naturally when computing factors from data. In the next section we explain why, on the contrary, *flat factors may actually be preferred by factorization algorithms* that are designed by principles directly generalized from the principles of factorization algorithms for Boolean data. The problem of how to avoid flat factors is addressed in section 3.

2.2. *Why do flat factors appear?*

Factorization algorithms for Boolean data and their extensions for data with graded attributes have to deal with the fact that the factorization problem and its commonly considered variants are NP-hard (for the Boolean case, see [18] and also [5,15]; for the case of graded attributes, see [7]). Virtually all current algorithms cope with the NP-hardness by computing factors one by one using a particular greedy strategy, obtaining thus suboptimal solutions to the factorization problem.

Generally speaking, the strategies to compute a new factor which are proposed in the literature aim at selecting a factor that explains most of the data not explained by the previously generated factors. At the end, one intends to come up with a set of k factors, represented by the $n \times k$ and $k \times m$ matrices A and B, that maximize the approximate equality (3), i.e. maximize

$$S(I, A \circ B).$$

Suppose $l-1$ factors have been computed, i.e. the $n \times (l-1)$ and $(l-1) \times m$ matrices $A^{(l-1)}$ and $B^{(l-1)}$ have been obtained. In order to select a good lth factor, and thus obtain the $n \times l$ and $l \times m$ matrices $A^{(l)}$ and $B^{(l)}$, it therefore seems reasonable to select a factor that maximizes the approximate equality

$$S(I, A^{(l)} \circ B^{(l)}).$$

While such strategy is—as we shall see—reasonable for Boolean data, it naturally leads to selection of flat, and thus undesirable, factors when factorizing graded data. This is particularly apparent when computing the first factor, as shown by the next examples.

Consider again the 7×9 input matrix I from Example 2 and the factors $F_1 = \langle C_1, D_1\rangle$, $F_2 = \langle C_2, D_2\rangle$, and $F_3 = \langle C_3, D_3\rangle$ from Examples 2 (a), (b), and Example 3, respectively. The top part of Fig. 1 depicts the crossproduct $C_1 \circ D_1$ (i.e. represents factor F_1) and the matrix $I \leftrightarrow C_1 \circ D_1$ consisting of entry-wise biresidua

$$(I \leftrightarrow C_1 \circ D_1)_{ij} = I_{ij} \leftrightarrow (C_1 \circ D_1)_{ij}.$$

As each entry $\langle i, j\rangle$ of $I \leftrightarrow C_1 \circ D_1$ represents closeness of I and $C_1 \circ D_1$ at entry $\langle i, j\rangle$, the matrix $I \leftrightarrow C_1 \circ D_1$ of biresidua may be regarded as the entry-by-entry representation of quality of factor $F_1 = \langle C_1, D_1\rangle$.

The bottom part of Fig. 1 depicts equality and approximate equality of matrices I and $C_1 \circ D_1$: The bottom left matrix represents equality in the entries of I and the entries of $C_1 \circ D_1$ in that its black entries $\langle i, j\rangle$ are those for which $I_{ij} = (C_1 \circ D_1)_{ij}$; the bottom right matrix represents 5/6-equality in that its black entries are those for which $I_{ij} \leftrightarrow (C_1 \circ D_1)_{ij} \geq 5/6$. The same information for factors $F_2 = \langle C_2, D_2\rangle$ and $F_3 = \langle C_3, D_3\rangle$ is shown in Fig. 2 and Fig. 3, respectively.

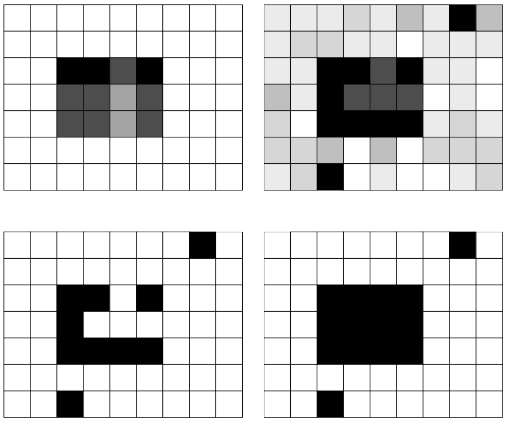


Fig. 2. Matrix $C_2 \circ D_2$ (top left), matrix $I \leftrightarrow C_2 \circ D_2$ of biresidua (top right), and matrices representing equality (bottom left) and at least 5/6-equality (bottom right) of entries in I and $C_2 \circ D_2$.

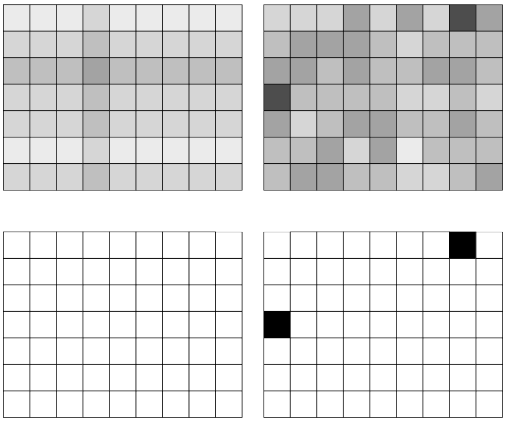


Fig. 3. Matrix $C_3 \circ D_3$ (top left), matrix $I \leftrightarrow C_3 \circ D_3$ of biresidua (top right), and matrices representing equality (bottom left) and at least 5/6-equality (bottom right) of entries in I and $C_3 \circ D_3$.

Figs. 1, 2, and 3 make more precise our intuitive observations from the previous section according to which factors $\langle C_1, D_1\rangle$ and $\langle C_2, D_2\rangle$ are desirable. Namely, the matrices of biresidua clearly show that each of these factors significantly explains a clearly delineated part of the input matrix I. On the other hand, factor $\langle C_3, D_3\rangle$ is undesirable because it is flat: As the matrix $I \leftrightarrow C_3 \circ D_3$ of biresidua and the matrices representing equality and 5/6-equality of entries reveal, the factor does not significantly explain any part of I.

Yet, factor $\langle C_3, D_3\rangle$ shall be preferred by factorization algorithms that aim at maximizing $S(I, A \circ B)$. Namely, the values of similarity $S(I, C_l \circ D_l)$ as well as normalized similarity $s(I, C_l \circ D_l)$ for the three factors are shown in the following table:[1]

	$\langle C_1, D_1\rangle$	$\langle C_2, D_2\rangle$	$\langle C_3, D_3\rangle$
$S(I, C_l \circ D_l)$	26.649	23.499	33.012
$s(I, C_l \circ D_l)$	0.423	0.373	0.524

The reason for the significantly higher similarity values of factor $\langle C_3, D_3\rangle$, and hence its preference over $\langle C_1, D_1\rangle$ and $\langle C_2, D_2\rangle$ becomes apparent when realizing that, in general, the similarity $S(I, C \circ D)$ of I and the matrix $C \circ D$ representing factor $\langle C, D\rangle$ equals the sum of all entries of the biresidua matrix $I \leftrightarrow C \circ D$, cf. (3):

$$S(I, C \circ D) = [I_{11} \leftrightarrow (C \circ D)_{11}] + [I_{12} \leftrightarrow (C \circ D)_{12}] + \cdots + [I_{nm} \leftrightarrow (C \circ D)_{nm}] = \sum_{i,j=1}^{n,m} (I \leftrightarrow C \circ D)_{ij}.$$

[1] Note that $S(I, C_l \circ D_l) = nm \cdot s(I, C_l \circ D_l) = 63 \cdot s(I, C_l \circ D_l)$.

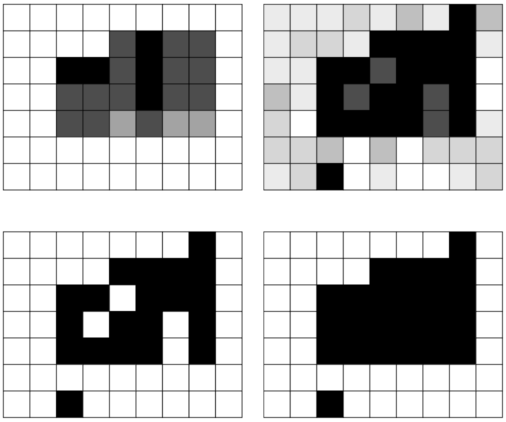


Fig. 4. Matrix $C_1 \circ D_1 \vee C_2 \circ D_2$ (top left), matrix $I \leftrightarrow (C_1 \circ D_1 \vee C_2 \circ D_2)$ of biresidua (top right), and matrices representing equality (bottom left) and at least 5/6-equality (bottom right) of entries in I and $C_1 \circ D_1 \vee C_2 \circ D_2$.

Now, while $S(I, C_1 \circ D_1)$ and $S(I, C_2 \circ D_2)$ result as sums of a small number of high values in entries (dark entries) of the corresponding biresidua matrices, the higher value $S(I, C_3 \circ D_3)$ results as a sum of large number of relatively small values (light grey entries) of the biresidua matrix $I \leftrightarrow C_3 \circ D_3$; cf. the biresidua matrices in the top right parts in Figs. 1, 2, and 3.

Remark 3. Interestingly, the flat factor $\langle C_3, D_3 \rangle$ is preferred even over the two-element combination (i.e. max-superposition) of factors $\langle C_1, D_1 \rangle$ and $\langle C_2, D_2 \rangle$, which is represented by the matrix $C_1 \circ D_1 \vee C_2 \circ D_2$. In this case, the same kind of information as above is presented in Fig. 4. Namely, for the similarity $S(I, C_1 \circ D_1 \vee C_2 \circ D_2)$ of I and the matrix $C_1 \circ D_1 \vee C_2 \circ D_2$ consisting of the two factors one has

$$S(I, C_1 \circ D_1 \vee C_2 \circ D_2) = 32.004 < S(I, C_3 \circ D_3).$$

For the normalized version, $s(I, C_1 \circ D_1 \vee C_2 \circ D_2) = 0.508 < s(I, C_3 \circ D_3)$.

Remark 4. Notice that the effects described in this section do not appear when factorizing binary matrices: Since the biresiduum coincides with classical equivalence for $L = \{0, 1\}$, the values in the biresidua matrices $I \leftrightarrow C \circ D$ are the values of classical equivalence, hence are 0 or 1. As a result, there are no small values of biresidua in $S(I, C_1 \circ D_1)$ whose sum could exceed the sum of large values in $S(I, C_2 \circ D_2)$ for any two factors $\langle C_1, D_1 \rangle$ and $\langle C_2, D_2 \rangle$.

3. Avoiding flat factors

In order to avoid flat factors, we propose to retain the basic logic of factorization but change what accounts for the undesirable effects presented in the previous section. We demonstrate below in this section and more thoroughly in section 4 that this new approach results in eliminating flat factors and computation of factors that are natural and have good ability to explain the data.

The observations from the previous section suggest to suppress the role of small values in the matrices $I \leftrightarrow A \circ B$, whose accumulation results in the undesirable preference of flat factors. For this purpose, we employ a suitable function

$$c : L \to [0, 1],$$

whose properties are discussed below, and define a modification of the matrix similarity function (3) as follows:

$$S_c(I, A \circ B) = \sum_{i,j=1}^{n,m} (c(I_{ij} \leftrightarrow (A \circ B)_{ij})). \tag{7}$$

The normalized version is then defined correspondingly:

$$s_c(I, A \circ B) = \frac{S_c(I, A \circ B)}{n \cdot m}, \text{ i.e. } s_c(I, A \circ B) = \frac{\sum_{i,j=1}^{n,m} (c(I_{ij} \leftrightarrow (A \circ B)_{ij}))}{n \cdot m}.$$

The function c as employed in $S_c(I, A \circ B)$ allows us to alleviate the effect of obtaining a large value by accumulation of small values $I_{ij} \leftrightarrow (A \circ B)_{ij}$. Namely, $S_c(I, A \circ B)$ becomes a sum in which the original values $I_{ij} \leftrightarrow (A \circ B)_{ij}$, being summed in $S(I, A \circ B)$, are replaced by smaller values $c(I_{ij} \leftrightarrow (A \circ B)_{ij})$, alleviating thus undesirable effect mentioned above.

To serve this purpose, c clearly needs to be subdiagonal, i.e. satisfy $c(a) \le a$ for each truth degree $a \in L$, and isotone, i.e. satisfy $c(a) \le c(b)$ for any $a \le b$. Moreover, we also require that the largest truth degree does not get smaller by c, i.e. $c(1) = 1$, and observe that the symmetric condition, $c(0) = 0$, follows from subdiagonality. To sum up, our basic requirements for c are as follows:

$$c(a) \leq a, \tag{8}$$

$$a \leq b \text{ implies } c(a) \leq c(b), \tag{9}$$

$$c(0) = 0 \text{ and } c(1) = 1 \tag{10}$$

for all truth degrees $a, b \in L$. From our experiments, it turns out that alleviating the accumulation of small values needs to be more severe for large matrices in order to avoid flat factors, because more values accumulate in large matrices. Therefore, it seems natural to consider c as a function $c_{n,m}$ parametrized by n and m (number of rows and columns, respectively) and require

$$c_{n,m}(a) \leq c_{p,q}(a) \text{ for } n \geq p \text{ and } m \geq q \tag{11}$$

for each $a \in L$. For brevity, we nevertheless omit indices and still use c rather than $c_{n,m}$ if there is no danger of confusion.

To focus on the main points in this contribution, we refrain from investigating the functions c in general. Rather, we present our results below for functions c of the form

$$c(a) = a^{q\sqrt{nm}} \tag{12}$$

for a real number $q > 0$ as a parameter, which is a simple function that yields good results in suppressing flat factors.

Remark 5.

(a) Functions c satisfying the first three properties, i.e. properties (8)–(10), or their variations, are known in fuzzy logic as intensifying (or, truth-stressing) modifiers and serve as models of intensifying linguistic hedges, i.e. unary connectives such as "very," "rather," and the like and were pioneered in [20]; see also [11,14].
(b) Notice that the identity function $c(a) = a$, which is obtained by setting $q = \frac{1}{\sqrt{nm}}$ in (12), satisfies the above conditions and that we have $S_c = S$ for this choice.
(c) In general, the function c defined by (12) satisfies the above conditions (9)–(11), but not (8) in general. It is immediate to check that c satisfies (8) if and only if $q\sqrt{nm} \geq 1$. Note, however, that it is not our aim in this paper to explore the variety of functions c satisfying the above conditions. Such exploration, practical as well as theoretical, is left for future explorations.

Example 4. Consider again the 7×9 input matrix I from Example 2 and the factors $F_1 = \langle C_1, D_1 \rangle$, $F_2 = \langle C_2, D_2 \rangle$, and $F_3 = \langle C_3, D_3 \rangle$ from Examples 2 (a), (b), and Example 3, respectively. We have observed in the above examples that when S is used to measure quality of factors, F_3 is preferred over F_1 as well as over F_2. Due to Remark 5 (b), S coincides with S_c for $q = \frac{1}{\sqrt{7 \cdot 9}} \approx 0.126$. In order for F_3 not to be preferred, we hence need to set $q > 0.126$. Already with $q = 0.2$, we obtain the following values of quality of the respective factors:

	$\langle C_1, D_1 \rangle$	$\langle C_2, D_2 \rangle$	$\langle C_3, D_3 \rangle$
$S_c(I, C_l \circ D_l)$	16.500	12.000	11.000
$s_c(I, C_l \circ D_l)$	0.262	0.190	0.175

That is, when c is given by $q = 0.2$, the flat factor F_3 is no longer preferred over F_1 and F_2. The matrices $C_l \circ D_l$, the biresiduum matrices $I \leftrightarrow C_l \circ D_l$, and the modified biresiduum matrices $c(I \leftrightarrow C_l \circ D_l)$ for this case are depicted in Fig. 5.

4. Experimental evaluation

In the previous section, we demonstrated using the running example that our approach indeed leads to avoiding flat factors. In this section, we illustrate that the problem addressed in this paper and its solution we proposed are relevant from the viewpoint of existing factorization algorithms. For this purpose, we consider two significant factorization algorithms, namely GreConD$_L$ and Asso$_L$, for which we refer to [3,4,6] and [4], respectively.

We first show that the current algorithms naturally lead to computation of flat factors. Secondly, we make it apparent that a simple modification of these algorithms based on our proposal described above alleviates the problem, i.e. suppresses computation

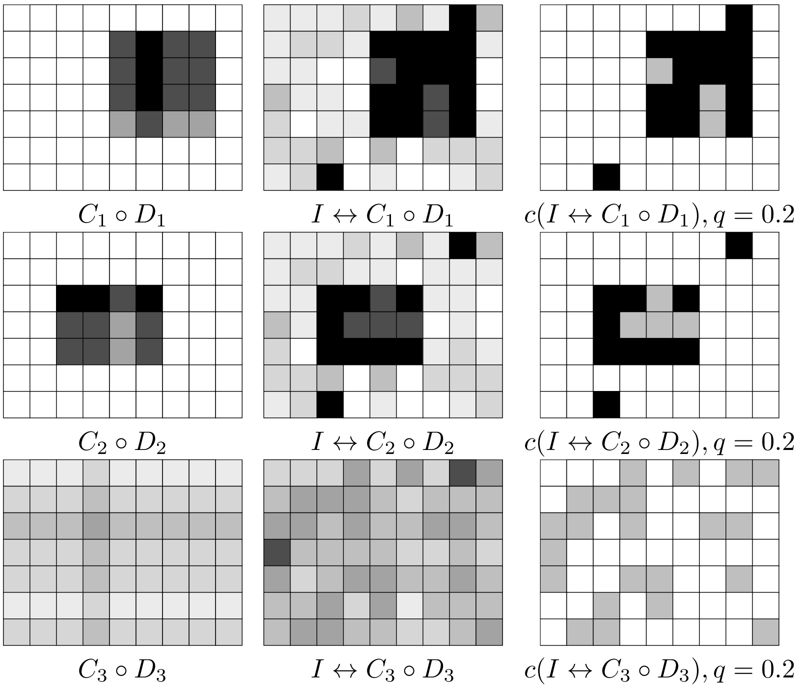


Fig. 5. Matrices $C_l \circ D_l$, the biresiduum matrices $I \leftrightarrow C_l \circ D_l$, and the modified biresiduum matrices $c(I \leftrightarrow C_l \circ D_l)$ for the factors from Examples 2 (a), (b), and Example 3.

of flat factors. We first consider small synthetic data, so that a reader may verify the computation process. Then, we examine larger synthetic as well as real data.

4.1. Employed algorithms

We now briefly describe GreConD$_L$ and Asso$_L$, which we use for our purpose. We also describe our modifications of these algorithms to suppress flat factors. The performance of factorization algorithms is commonly assessed using the matrix similarity function (3) or its normalized version (4). This also applies to GreConD$_L$ and Asso$_L$. Both these algorithms compute factors in a greedy manner to achieve—after the algorithm finishes with a set $\mathcal{F}$ of factors—a large value of approximate equality $S(A_{\mathcal{F}} \circ B_{\mathcal{F}}, I)$ of $A_{\mathcal{F}} \circ B_{\mathcal{F}}$ (matrix reconstructed from the factors) and I (input matrix).

Even though GreConD$_L$ and Asso$_L$ are rather different as regards their strategies, they both may be viewed as maximizing a particular function, which corresponds to how well the computed factors cover the input data matrix I. We shall hence call the respective functions Cover, and describe them as part of our description of GreConD$_L$ and Asso$_L$.

GreConD$_L$: brief description and our modification. For the already computed set $\mathcal{F}$ of factors, which is initially empty, GreConD$_L$ constructs the next factor by adding sequentially the most promising attributes to an initially empty fuzzy set D, which determines the factor. In more detail, if D denotes a fuzzy set of attributes of the factor constructed so far, the algorithm selects an attribute $j \in \{1, \ldots, m\}$ and degree $a \in L$ maximizing the value

$$\mathrm{Cover}(\mathcal{F}, I, D) = S_c\left(A_{\mathcal{F}} \circ B_{\mathcal{F}} \vee D^{+\downarrow} \circ D^{+\downarrow\uparrow}, I\right), \tag{13}$$

where $D^{+} = D \cup \{{}^{a}/_{j}\}$, and $^{\downarrow}$ along with $^{\uparrow}$ are certain fundamental operators (for details about these operators, see e.g. [6]). Note now with regard to (13):

- $A_{\mathcal{F}} \circ B_{\mathcal{F}}$ is the matrix reconstructed from the previously computed set $\mathcal{F}$ of factors.
- $\langle D^{+\downarrow}, D^{+\downarrow\uparrow} \rangle$ is the new candidate factor obtained from the fuzzy set $D \cup \{{}^{a}/_{j}\}$.
- The matrix $D^{+\downarrow} \circ D^{+\downarrow\uparrow}$ represents part of I explained by factor $\langle D^{+\downarrow}, D^{+\downarrow\uparrow} \rangle$. Hence $A_{\mathcal{F}} \circ B_{\mathcal{F}} \vee D^{+\downarrow} \circ D^{+\downarrow\uparrow}$ is the matrix reconstructed from $\mathcal{F}$ to which $\langle D^{+\downarrow}, D^{+\downarrow\uparrow} \rangle$ is added.
- Two basic variants of GreConD$_L$ have been examined in the past. In the first one [6], the new factor $\langle D^{+\downarrow}, D^{+\downarrow\uparrow} \rangle$ is selected to maximize the number of entries fully covered by the factor that were not covered by the factors in $\mathcal{F}$, i.e. entries $\langle i, j \rangle$ for which $(D^{+\downarrow} \circ D^{+\downarrow\uparrow})_{ij} = 1$ but $(A_{\mathcal{F}} \circ B_{\mathcal{F}})_{ij} < 1$. One may verify that the first variant is equivalent to (i.e. delivers the same factors as) the one described above with (13) in which c is defined by

$$c(a) = \begin{cases} 1, & \text{for } a = 1, \\ 0, & \text{for } a < 1. \end{cases}$$

The second variant results from (13) by replacing S_c by S, or, equivalently, by letting c be the identity function. This variant is directly derived from the quality assessment of the set $\mathcal{F}$ of factors, which itself is based on S.

– Our modification consists in that we allow in (13) a general function c satisfying the conditions described above. From a technical viewpoint, our modification is more general than the previously considered variants of GreConD$_L$. More important, however, is the fact that our modification originates from the analysis of flat factors and a natural way to suppress flat factors.

Asso$_L$: brief description and our modification. Asso$_L$ uses the rows of the so-called association matrix $\mathcal{A}$, which is computed from the input object-attribute matrix $I \in L^{n\times m}$, as rows of the factor-attribute matrix B. The association matrix $\mathcal{A}$ is of dimension $(m \cdot |K|) \times m$ and is defined for each $j \in \{1, \ldots, m\}$, $a \in K$ and $j' \in \{1, \ldots, m\}$ by

$$\mathcal{A}_{\langle j,a\rangle, j'} = \text{round}\, c_a(p, q),$$

where $K \subseteq L \setminus \{0\}$ is a chosen set of truth degrees, round is a function rounding its argument to appropriate neighboring truth degree in the scale L, and $c_a(p, q)$ is a conditional probability that the presence of attribute p to degree at least a implies the presence of the attribute q to degree 1.

The algorithm computes the factors one by one. With the set $\mathcal{F}'$ of factors computed so far, the next factor $\langle C, D\rangle$ is obtained the following way. One selects degrees $c_1, \ldots, c_n \in L$ and a row $\mathcal{A}_{\langle j,a\rangle\_} \in L^{1\times m}$ of the association matrix $\mathcal{A}$ such that the expanded set $\mathcal{F} = \mathcal{F}' \cup \{\langle C, D\rangle\}$ with

$$C = \begin{pmatrix} c_1 \\ \vdots \\ c_n \end{pmatrix} \quad \text{and} \quad D = \mathcal{A}_{\langle j,a\rangle\_}$$

explains the input data I the best in that the value of the following cost function Cover is maximized since

$$\begin{aligned} \text{Cover}(\mathcal{F}, I, w^+, w^-) = w^+ \cdot & \sum_{i,j=1}^{n,m} \{c\big((A_\mathcal{F} \circ B_\mathcal{F})_{ij} \leftrightarrow I_{ij}\big) \ ; \ (A_\mathcal{F} \circ B_\mathcal{F})_{ij} \le I_{ij}\} \\ & + w^+ \cdot |\{\langle i, j\rangle \ ; \ (A_\mathcal{F} \circ B_\mathcal{F})_{ij} > I_{ij}\}| \\ & - w^- \cdot \sum_{i,j=1}^{n,m} \{1 - c\big((A_\mathcal{F} \circ B_\mathcal{F})_{ij} \leftrightarrow I_{ij}\big) \ ; \ (A_\mathcal{F} \circ B_\mathcal{F})_{ij} > I_{ij}\}, \end{aligned} \tag{14}$$

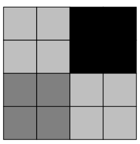
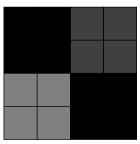
where the weights w^+ and w^- express the importance of the so-called uncovered and overcovered entries, respectively; see [4] for details. Note at this point that the original Asso$_L$ algorithm actually results by letting c be the identity function and that our modification of Asso$_L$ results by employing a function c satisfying the conditions described in the previous section. Note furthermore that for $w^+ = w^- = 1$, the function coincides with the above-defined similarity $S_c(A_\mathcal{F} \circ B_\mathcal{F}, I)$:

$$\begin{aligned} \text{Cover}(\mathcal{F}, I, 1, 1) = & \sum_{i,j=1}^{n,m} \{c\big((A_\mathcal{F} \circ B_\mathcal{F})_{ij} \leftrightarrow I_{ij}\big) \ ; \ (A_\mathcal{F} \circ B_\mathcal{F})_{ij} \le I_{ij}\} \\ & + \sum_{i,j=1}^{n,m} \{c\big((A_\mathcal{F} \circ B_\mathcal{F})_{ij} \leftrightarrow I_{ij}\big) \ ; \ (A_\mathcal{F} \circ B_\mathcal{F})_{ij} > I_{ij}\} \\ & = \sum_{i,j=1}^{n,m} \big(c((A_\mathcal{F} \circ B_\mathcal{F})_{ij} \leftrightarrow I_{ij})\big) = S_c(A_\mathcal{F} \circ B_\mathcal{F}, I). \end{aligned}$$

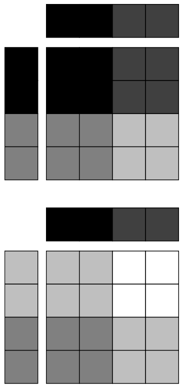
Therefore, by employing general weights w^+ and w^- the function Cover represents a generalization of S_c by putting different emphasis on undercovering and overcovering of the entries of the input matrix I.

4.2. Synthetic data

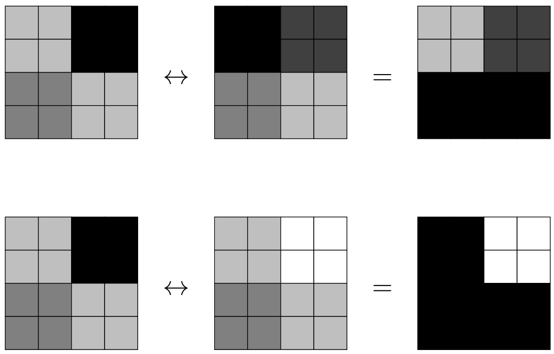
Small matrix processed by Asso$_L$. Consider first the following 4×4 matrix I and its corresponding association matrix for $K = \{1\}$:

$I =$  , $\mathcal{A} =$ 

For the original Asso_L, i.e. by setting the identity for c, and with $w^+ = w^- = 1$, there exist several factors with the maximal value of Cover defined by (14). The following two of them are created using the first row of the association matrix:



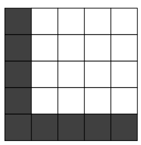
The value of Cover for both of these factors equals 12, as one checks by computing the cardinality of the corresponding biresidua matrices:



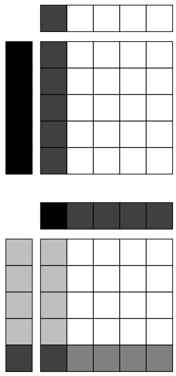
One may notice that the first matrix of biresidua contains quite a large area filled with the values 0.25 and 0.75 making the corresponding factor flat. On the other hand, the second matrix of biresidua does not contain any such area: 75% of all entries are explained by the second factor exactly (black part), while 25% of all entries are not explained at all (white part).

Which of these two factors are selected by the original Asso_L depends on the implementation of this algorithm (namely, it depends on the order in which the rows of the association matrix $\mathcal{A}$ are processed). However, when our modification of Asso_L is used with a more strict function c, the value of Cover of the flat factor obviously becomes smaller than 12 and, as a result, the non-flat factor is selected (regardless of implementation).

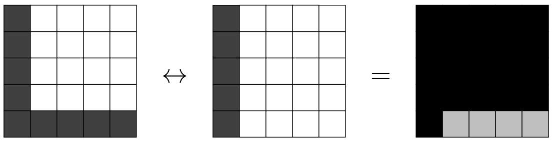
Small matrix processed by GreConD$_L$. Consider now the following 5×5 matrix:

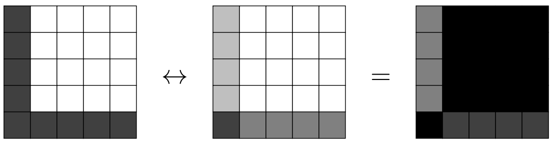
$I =$ 

For the original $\mathrm{GreConD}_L$, i.e. taking identity for c, there exist the following two candidate factors maximizing the value of Cover defined by (13):



The value of Cover for both of these factors equals to 22, as one easily verifies by computing the cardinality of the following biresidua matrices:





As one can see, the second matrix of biresidua contains a relatively large area containing the values 0.5 and 0.75, making the second factor flat. On the other hand, the first matrix of biresidua contains only four entries filled with the values 0.25; the remaining 21 entries contain the value 1, i.e. 21 of the 25 entries of I are fully explained by this factor.

It is hence obvious that when a proper modifying function is used, the modified algorithm GreConD$_L$ selects the first factor, suppressing thus the flat factor, which may be selected by the original version of GreConD$_L$.

Large matrix processed by Asso$_L$. We now apply Asso$_L$ to a randomly generated 90×10 matrix depicted in the left part of Fig. 6. The matrix on the right is the corresponding association matrix computed for $K = \{0.5, 0.75, 1\}$; see the description of Asso$_L$ in section 4.1 for the role of K. The weights are set to $w^+ = 2$ and $w^- = 1$. We consider only the first factor F_1 and the corresponding matrix of biresidua. Obviously, flatness appears on the consequent factors as well, but we refrain from presenting them due to space constraints.

Fig. 7 shows the biresiduum matrix corresponding to the first factor computed by the original Asso$_L$ (left) and the first factor computed by the modified Asso$_L$ (right) with the function c defined by

$$c(a) = \begin{cases} 1, & \text{for } a = 1, \\ 0, & \text{otherwise.} \end{cases} \tag{15}$$

One may notice that the biresiduum matrix on the left contains many gray values, i.e. values around 0.5. The corresponding factor is thus flat. On the contrary, the biresiduum matrix on the right is significantly more contrasting; that is, the gray values are present much less frequently, while the black values occupy a bigger area. Notice that the entries with black values are just the entries of the input matrix I whose value is exactly reconstructed by the first factor. The comparison of the number of black values (i.e. values 1) in both cases is apparent from Fig. 8: Only the black entries of the biresidua matrices of the first factor computed by Asso$_L$ (left) and the first factor computed by the modified Asso$_L$ (right) are shown.

4.3. Real data

To demonstrate that flat factors appear on real data, we now present a part of our analysis of ordinal data coming from the examination tests used by universities in the United Kingdom. For brevity we only present the results for Asso$_L$. Each test consist of 6 questions, each of them assessed by examiners with regard to assessment objectives—the last four questions are assessed by objectives “knowledge and understanding”, “analysis and evaluations” and “communications,” while for the first two questions only the objective “knowledge and understanding” is considered. As a result, every student examination is evaluated by 14 marks; each

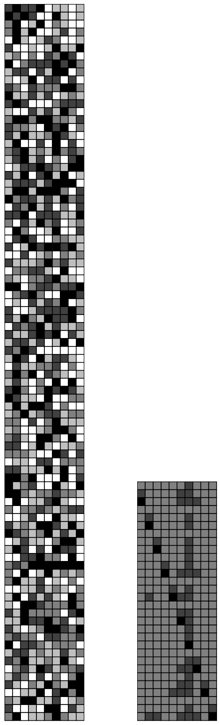


Fig. 6. Input matrix (left) and the corresponding association matrix (right) for $K = \{0.5, 0.75, 1\}$.

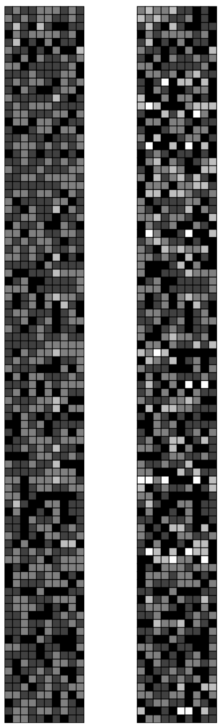


Fig. 7. Biresidua matrices of the first factor: Asso_L (left), modified Asso_L (right).

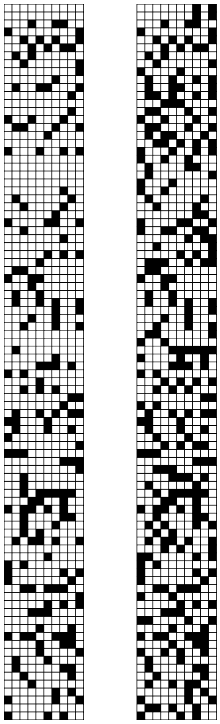


Fig. 8. Entries with value 1 in the biresidua matrices: Asso_L (left), modified Asso_L (right).

mark is represented by a value in the five-element scale $\{0, 1/4, 1/2, 3/4, 1\}$ with 1 indicating the best performance. Moreover, based on these particular marks a total mark A, B, C, D or E (with A being the best result) is assigned to every student by a particular procedure.

The examined data may therefore be described by a matrix I with 2774 rows (representing students) and 14 columns (representing particular marks) and with degrees in the five-element scale $L = \{0, 1/4, 1/2, 3/4, 1\}$, i.e. $I \in L^{2774\times 14}$. For space reasons, we present only the portion of the data that includes students who obtained the total grade D. As a result, we perform our analysis on submatrix of I with 100 rows and 14 columns depicted in Fig. 9.

Analogously to the previous section containing random data, we consider only the first factor produced by Asso_L. Fig. 10 presents the biresidua matrix corresponding to this factor. The matrix on the left was computed by the original Asso_L algorithm, while the matrix on the right was computed by the modified Asso_L with the function c defined as in (15). Again, the left matrix of biresidua reveals flatness of the factor. In comparison, the matrix on the right is much more discriminating and hence less flat. The comparison of the number of black values, explaining the entries of I precisely, is depicted in Fig. 11.

5. Future research

The problem and contributions presented in this paper open way to a diverse set of streams for future research. Some of them are outlined below.

– In a broader context of fuzzy sets, the problem of flat factors presented in this paper may be rephrased in terms of cardinalities of fuzzy sets. In this perspective, our considerations reveal a significant challenge regarding the concept of cardinality that has apparently not yet been addressed.
  In more detail, consider the fuzzy relation R between the set of objects and the set of attributes defined by $R(i,j) = I_{ij} \leftrightarrow (A\circ B)_{ij}$, which corresponds to the matrix of biresidua considered above. One may observe that the approximate equality $S(I, A\circ B)$ of matrices I and $A\circ B$ equals the sigma-count $|R|$ of R,

$$S(I, A\circ B) = |R|.$$

  Recall that the sigma-count $|Q|$ of a fuzzy set Q in a finite universe $U = \{u_1, \dots, u_k\}$ is defined by $|Q| = Q(u_1) + \cdots + Q(u_k)$. The concept of sigma-count (also termed scalar cardinality) has been proposed under the name a power of a fuzzy set in [8] as a straightforward generalization of cardinality of finite sets (see e.g. [19] for a comprehensive treatment on cardinalities).
  In terms of cardinalities, preference of factors with large $S(I, A\circ B)$ hence translates to preference of factors for which R is large. Our objection to using $S(I, A\circ B)$ due to the effect of possible accumulation of a large number of small values hence translates to the following objection: The sigma-count cardinality of fuzzy sets may not be appropriate in certain situations because

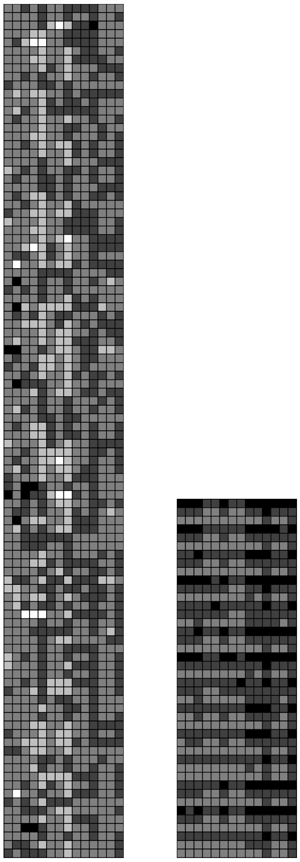


Fig. 9. Input data (students with grade D) and corresponding association matrix for $K = \{0.5, 0.75, 1\}$.

accumulation of small membership degrees renders as large those fuzzy sets which are intuitively not large. Our solution above may be rephrased as follows: We suggest to replace the sigma count $|Q|$ by its variant $|Q|_c$, defined by $|Q|_c = c(Q(u_1)) + \cdots + c(Q(u_k))$.

Apparently, cardinalities of fuzzy sets alleviating the effect of accumulation of small membership degrees have not been studied in the literature. The paper [17] seems to be an exception. In this paper, Ralescu mentions the undesirable effect of accumulation as one of his motivations. Nevertheless, the solution proposed in [17] do not alleviate this undesirable effect. Note also that a formula for scalar cardinality that yields an appropriate notion of cardinality from our viewpoint is presented in [19, p. 36], namely $|Q| = \sum_{u \in U} Q(u)^p$ for $p > 0$; it was introduced in [12] as a mathematical generalization of basic scalar cardinality without apparent practical motivations. Exploration of novel concepts of cardinality taking into account the undesirable effect of accumulation of small membership degrees hence seems to present a significant direction to be pursued.

– In a sense, our solution to the problem of flat factors consists in replacing the approximate equality $S(I, J)$ of matrices with degrees (or, equivalently, fuzzy relations) I and J by $S_c(I, J)$. While S has thoroughly been examined in the literature, S_c represents a new concept of approximate equality, which needs further exploration in the context of research in similarities and generalized metrics.
– The problem of flat factors may be regarded as pointing to a more general problem of semantics of factors extracted from ordinal data. We observed a particular significance of this problem when analyzing large data, in particular in our analysis of educational data [1]. In this study, it became obvious that certain issues, not apparent on small and middle-size data, become significant when dealing with large data. Except for flatness of factors, another topic observed was interpretability as related to the structure of the fuzzy set D of a given factor $\langle C, D \rangle$.
– We regard as important to be able to characterize the flatness of factors quantitatively. That is to say, to define a degree to which a factor is considered flat and, more generally, a degree to which a factorization is considered flat. This is a non-trivial question and needs to be explored in the future.
– Let us also mention the recent paper [9], in which a more general factorization model is proposed. The model seems interesting also from the viewpoint of factorization of Boolean matrices, since it is more general than the classical model not only because it involves intermediate degrees rather than 0 and 1 only, but also because even when restricting to the Boolean case, the model in [9] is more general than the classical model of Boolean factor analysis as it involves more complex decompositions.

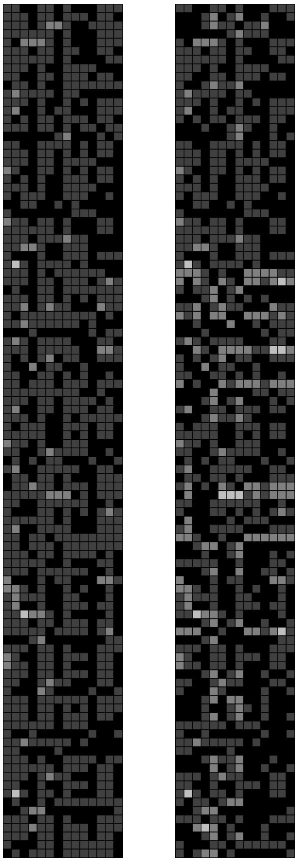


Fig. 10. Biresidua matrices of the first factor: Asso_L (left), modified Asso_L (right).

For one, the authors in [9] point out a semantic problem regarding interpretability of factors, which is of a similar nature as the one examined in [1] mentioned in our previous point in this section. The problem is different from the one discussed in our paper, which pertains to the ability of factors to explain the input data. These observations point out the fact that in presence of intermediate degrees, new phenomena regarding factorization appear which are non-existent in the classical, binary case. These phenomena present non-trivial challenges and need to be studied further.

Secondly, it seems proper to develop a deeper insight into the factor model in [9] that may be utilized in the design of factorization algorithms. Namely, the present factorization methods for the Boolean data as well as for data involving degrees, which are based on the classical model utilized also in our paper, make use of a geometric insight according to which factorization may be looked at as a certain coverage problem using certain rectangular patterns in the factorized data. On the other hand, the factorization method in [9] proposed for the more complex model is a variant of the gradient descent method. As such, it does not make use of any significant insight into the decomposition. Development of a proper insight in this regard seems an interesting topic for future research, which is significant from a practical viewpoint.

CRediT authorship contribution statement

Eduard Bartl: 50%
Radim Belohlavek: 50%

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

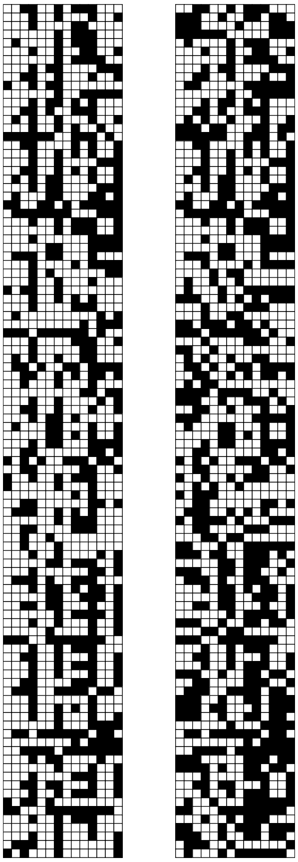


Fig. 11. Entries with value 1 in the biresidua matrices: Asso_L (left), modified Asso_L (right).

References

[1] E. Bartl, R. Belohlavek, A. Scharaschkin, Toward factor analysis of educational data, in: Proc. CLA, 2018, pp. 191–206.
[2] R. Belohlavek, Optimal decompositions of matrices with entries from residuated lattices, J. Log. Comput. 22 (6) (2012) 1405–1425, https://doi.org/10.1093/logcom/exr023, online: September 7, 2011.
[3] R. Belohlavek, M. Krmelova, Factor analysis of ordinal data via decomposition of matrices with grades, Ann. Math. Artif. Intell. 72 (1) (2014) 23–44.
[4] R. Belohlavek, M. Trneckova, The discrete basis problem and Asso algorithm for fuzzy attributes, IEEE Trans. Fuzzy Syst. 27 (7) (2019) 1417–1427.
[5] R. Belohlavek, V. Vychodil, Discovery of optimal factors in binary data via a novel method of matrix decomposition, J. Comput. Syst. Sci. 76 (1) (2010) 3–20.
[6] R. Belohlavek, V. Vychodil, Factor Analysis of Incidence Data via Novel Decomposition of Matrices, LNAI, vol. 5548, 2009, pp. 83–97.
[7] R. Belohlavek, V. Vychodil, Factorization of matrices with grades, Fuzzy Sets Syst. 292 (2016) 85–97.
[8] A. DeLuca, S. Termini, A definition of a nonprobabilistic entropy in the setting of fuzzy sets theory, Inf. Control 20 (1972) 301–312.
[9] H. E, Y. Cui, W. Pedrycz, Z. Li, Fuzzy relational matrix factorization and its granular characterization in data description, IEEE Trans. Fuzzy Syst. 30 (3) (2022) 794–804.
[10] S. Gottwald, A Treatise on Many-Valued Logics, Research Studies Press, Baldock, Hertfordshire, England, 2001.
[11] P. Hájek, Metamathematics of Fuzzy Logic, Kluwer, Dordrecht, 1998.
[12] A. Kaufmann, Introduction à la Théorie des Sous-Ensembles Flous, Complément et Nouvelles Applications, vol. 4, Masson, Paris, 1977.
[13] K.H. Kim, Boolean Matrix Theory and Applications, CRC Press, 1982.
[14] G.J. Klir, B. Yuan, Fuzzy Sets and Fuzzy Logic: Theory and Applications, Prentice-Hall, 1995.
[15] P. Miettinen, T. Mielikäinen, A. Gionis, G. Das, H. Mannila, The discrete basis problem, IEEE Trans. Knowl. Data Eng. 20 (10) (2008) 1348–1362.
[16] G.A. Miller, The magical number seven, plus or minus two: some limits on our capacity for processing information, Psychol. Rev. 63 (1956) 81–97.
[17] D. Ralescu, Cardinality, quantifiers, and the aggregation of fuzzy criteria, Fuzzy Sets Syst. 69 (3) (1995) 355–365.
[18] L.J. Stockmeyer, The set basis problem is NP-complete, IBM Research Report RC5431, Yorktown Heights, NY, 1975.
[19] M. Wygralak, Cardinalities of Fuzzy Sets, Springer, 2003.
[20] L.A. Zadeh, A fuzzy-set-theoretic interpretation of linguistic hedges, J. Cybern. 2 (3) (1972) 4–34.