

Optimal Decompositions of Matrices with Grades into Binary and Graded Matrices^{*}

Eduard Bartl¹, Radim Belohlavek^{1,2}, Jan Konecny¹

¹ Dept. Systems Science and Industrial Engineering
T. J. Watson School of Engineering and Applied Science
Binghamton University–SUNY, PO Box 6000, Binghamton, NY 13902–6000, USA
ebartl1@binghamton.edu, rbelohla@binghamton.edu, jkonecn1@binghamton.edu

² Dept. Computer Science, Palacky University, Olomouc
Tomkova 40, CZ-779 00 Olomouc, Czech Republic

Abstract. The paper contributes to factor analysis of relational data. We study the problem of decomposition of object-attribute matrices with grades, i.e. matrices whose entries contain degrees to which objects have attributes. The degrees are taken from a bounded partially ordered scale. Examples of such matrices are binary matrices, matrices with entries from a finite chain, or matrices with entries from the unit interval $[0, 1]$. We study the problem of decomposition of a given object-attribute matrix I with grades into an object-factor matrix A and a binary factor-attribute matrix B , with the number of factors as small as possible. We present a theorem describing optimal decompositions. The theorem shows that decompositions which use as factors particular formal concepts associated to I are optimal in that the number of factors involved is the smallest possible. Furthermore, we present an approximation algorithm for finding those decompositions and illustrative examples.

1 Introduction and Problem Setting

Problem description in brief This paper presents results on optimal decompositions of matrices with grades. Examples of such matrices are binary (or Boolean) matrices, i.e. matrices which entries are 0 or 1. Other examples are matrices which contain numbers from the unit interval $[0, 1]$ as their entries. In general we consider non-numerical matrices with entries from particular complete lattices L (binary matrices and matrices with entries from $[0, 1]$ are particular examples with $L = \{0, 1\}$ and $L = [0, 1]$, respectively).

We consider the following problem. Let L be a partially ordered scale bounded from below and above by 0 and 1 (details specified later). Given an $n \times m$ matrix I with entries from L (i.e. $I_{ij} \in L$), we want to decompose I into a product

$$I = A \circ B$$

^{*} Supported by grant No. 1ET101370417 of GA AV ČR and by institutional support, research plan MSM 6198959214.

of an $n \times k$ matrix A with entries from L (i.e. $A_{il} \in L$) and a $k \times m$ binary matrix B (i.e. $B_{lj} \in \{0, 1\}$) with k as small as possible. The composition operation \circ which we consider is defined by

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

where \otimes is defined by $a \otimes 1 = a$ and $a \otimes 0 = 0$. Note that if $L = \{0, 1\}$ then $A \circ B$ is the well-known Boolean product of binary matrices. Note also that if we allow $A_{il} \in L$ and $B_{lj} \in L$ and if \otimes is a t-norm then \circ is the product of graded matrices well-known in fuzzy set theory, see e.g. [15], and that such decompositions were considered in [4, 7].

Factor analysis model For a decomposition $I = A \circ B$ given by (1), I_{ij} can be interpreted as a degree to which there is a factor l such that l applies to object i and l is associated to attribute j (j is a particular manifestation of l). This way, a decomposition $I = A \circ B$ provides us with a factor analysis model (see [1, 13, 16] for references on factor analysis): A relationship between objects and original attributes given by I is described using a relationship between the objects and new variables, called factors, which is given by A , and a relationship between factors and the original attributes, which is given by B . Note that we assume that B is binary, i.e. that the relationship between factors and attributes is a yes-or-no relationship. This feature distinguishes our approach from those which we considered earlier.

Needless to say, one can consider decompositions $I = A \circ B$ given by (1), in which A is binary and B arbitrary. Obviously, using $I^T = B^T \circ A^T$, one can reduce this type of decomposition to the first type (A arbitrary, B binary). Therefore, we do not consider such case.

Contribution of the paper We present a theorem regarding optimal decompositions of a given matrix I which shows that decompositions which use as factors particular formal concepts, called crisply generated concepts, are optimal in that they involve the least number of factors among all decompositions of I . Furthermore, we present an approximation algorithm for finding those decompositions and provide illustrative examples.

Related and previous work The paper is a continuation of our previous work [4, 6, 7]. In particular, in [4, 7] we considered decompositions $I = A \circ B$ given by (1), in which both A and B were arbitrary, i.e. none of them was required to be binary.

Preliminaries from fuzzy logic We use standard notions of fuzzy logic and fuzzy sets, see e.g. [2, 12, 15]. In particular, we use complete residuated lattices as structures of truth degrees. Recall that a complete residuated lattice is an algebra $\mathbf{L} = \langle L, \wedge, \vee, \otimes, \rightarrow, 0, 1 \rangle$ such that $\langle L, \wedge, \vee, 0, 1 \rangle$ is a complete lattice, $\langle L, \otimes, 1 \rangle$ is a commutative monoid, and \otimes and \rightarrow satisfy so-called adjointness condition, i.e. $a \otimes b \leq c$ if and only if $a \leq b \rightarrow c$. We assume familiarity with examples and

basic properties of residuated lattices. As an example, for $L = [0, 1]$, $a \otimes b = \max(0, a+b-1)$, $a \rightarrow b = \min(1, 1-a+b)$, the algebra $\mathbf{L} = \langle [0, 1], \wedge, \vee, \otimes, \rightarrow, 0, 1 \rangle$ is a complete residuated lattice (so-called standard Łukasiewicz algebra). An L -set in a universe set U is a mapping $A : U \rightarrow L$.

2 Optimal Decompositions

2.1 Composition as \vee -superposition of matrices

We first observe that $I = A \circ B$ for $n \times k$ and $k \times m$ matrices A (graded) and B (binary) means that I is a \vee -superposition of particular rectangular matrices.

Definition 1. Let $K_1, K_2 \subseteq L$. An $n \times m$ matrix J with entries from L is called (K_1, K_2) -rectangular iff there exist L -sets C in $\{1, \dots, n\}$ and D in $\{1, \dots, m\}$ with $C(i) \in K_1$ and $D(j) \in K_2$ such that $J = C \otimes D$, i.e.

$$J_{ij} = C(i) \otimes D(j) \quad (2)$$

for $1 \leq i \leq n$, $1 \leq j \leq m$.

In particular, we need $(L, \{0, 1\})$ -rectangular matrices and call these just “rectangular”. The term “rectangular” is inspired by the “shape” of such matrices. The following matrices are examples of $(\{0, 1\}, \{0, 1\})$ -rectangular (J_1) and $([0, 1], \{0, 1\})$ -rectangular (J_2) matrices:

$$J_1 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad J_2 = \begin{pmatrix} 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.5 & 0.5 & 0.5 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 1.0 & 1.0 & 1.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.2 & 0.2 & 0.2 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 1.0 & 1.0 & 1.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \end{pmatrix}.$$

In the above example, $J_1 = C \otimes D$ where C and D are characteristic functions of $\{3, 4, 5, 6\}$ and $\{3, 4, 5\}$, respectively; $J_2 = C \otimes D$ where $C(1) = C(2) = C(7) = C(8) = 0$, $C(3) = 0.5$, $C(4) = 1.0$, $C(5) = 0.2$, $C(6) = 1.0$, and $D(1) = D(2) = D(6) = D(7) = 0$, $D(3) = D(4) = D(5) = 1$.

The role of $(L, \{0, 1\})$ -rectangular matrices is shown by the following theorem.

Theorem 1. $I = A \circ B$ for $n \times k$ and $k \times m$ matrices A and B with $A_{il} \in L$ and $B_{lj} \in \{0, 1\}$ iff I is a \vee -superposition of k $(L, \{0, 1\})$ -rectangular matrices J_1, \dots, J_k , i.e. iff

$$I = J_1 \vee J_2 \vee \dots \vee J_k.$$

Proof. Denote by J_l the \circ -product $A_{\cdot l} \circ B_{l \cdot}$ of the l -th column $A_{\cdot l}$ of A and the l -th row $B_{l \cdot}$ of B , i.e. $(J_l)_{ij} = A_{il} \otimes B_{lj}$. $I = A \circ B$ means $I_{ij} = (A \circ B)_{ij}$, i.e. $I_{ij} = \bigvee_{l=1}^k (A_{il} \otimes B_{lj})$. Therefore, $I = J_1 \vee J_2 \vee \dots \vee J_k$. Since B is a binary matrix, J_l are $(L, \{0, 1\})$ -rectangular matrices. \square

Example 1. To illustrate the content of Theorem 1, consider the following decomposition $I = A \circ B$:

$$\begin{pmatrix} 0.3 & 1.0 & 0.0 & 0.0 & 0.0 & 0.3 \\ 1.0 & 1.0 & 0.0 & 0.0 & 1.0 & 1.0 \\ 1.0 & 0.9 & 1.0 & 1.0 & 0.0 & 0.8 \\ 1.0 & 0.2 & 0.0 & 0.0 & 1.0 & 0.2 \end{pmatrix} = \begin{pmatrix} 0.3 & 0.0 & 0.0 & 1.0 \\ 1.0 & 0.0 & 1.0 & 0.7 \\ 0.8 & 1.0 & 0.0 & 0.9 \\ 0.2 & 0.0 & 1.0 & 0.0 \end{pmatrix} \circ \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

In this example $L = \{0, 0.1, \dots, 0.9, 1\}$ and $a \otimes b = \min(a, b)$. According to Theorem 1, I is a \bigvee -superposition of four matrices, J_1, J_2, J_3, J_4 where J_l is a \circ -product of the l -th column of A and the l -th row of B , i.e.

$$\begin{pmatrix} 0.3 & 1.0 & 0.0 & 0.0 & 0.0 & 0.3 \\ 1.0 & 1.0 & 0.0 & 0.0 & 1.0 & 1.0 \\ 1.0 & 0.9 & 1.0 & 1.0 & 0.0 & 0.8 \\ 1.0 & 0.2 & 0.0 & 0.0 & 1.0 & 0.2 \end{pmatrix} = \begin{pmatrix} 0.3 & 0.3 & 0.0 & 0.0 & 0.0 & 0.3 \\ 1.0 & 1.0 & 0.0 & 0.0 & 0.0 & 1.0 \\ 0.8 & 0.8 & 0.0 & 0.0 & 0.0 & 0.8 \\ 0.2 & 0.2 & 0.0 & 0.0 & 0.0 & 0.2 \end{pmatrix} \bigvee \begin{pmatrix} 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 1.0 & 0.0 & 1.0 & 1.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \end{pmatrix} \bigvee \begin{pmatrix} 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 1.0 & 0.0 & 0.0 & 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 1.0 & 0.0 & 0.0 & 0.0 & 1.0 & 0.0 \end{pmatrix} \bigvee \begin{pmatrix} 0.0 & 1.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.7 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.9 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \end{pmatrix}.$$

2.2 Formal concepts are optimal factors

Theorem 1 says that in order to find a decomposition $I = A \circ B$, we need to find a suitable set of $(L, \{0, 1\})$ -rectangular matrices J_l whose \bigvee -superposition gives I . We now describe decompositions of I which are optimal among all possible decompositions in that the number k of factors is the smallest possible one. The decompositions use so-called crisply generated formal concepts of I [5].

Preliminaries on crisply generated formal concepts This section presents preliminaries on formal concepts of data with fuzzy attributes, particularly on crisply generated formal concepts. The reader is referred, e.g., to [3, 5] for details.

Let $X = \{1, \dots, n\}$ and $Y = \{1, \dots, m\}$ be sets (of objects and attributes, respectively), I be an $n \times m$ matrix with entries from a support set L of a complete residuated lattice \mathbf{L} . The degree $I_{xy} \in L$ is interpreted as a degree to which object x has attribute y . Consider the operators $\uparrow : L^X \rightarrow L^Y$ and $\downarrow : L^Y \rightarrow L^X$ defined by

$$C^\uparrow(y) = \bigwedge_{x \in X} (C(x) \rightarrow I_{xy}), \quad D^\downarrow(x) = \bigwedge_{y \in Y} (D(y) \rightarrow I_{xy}),$$

where \rightarrow is the residuum of the complete residuated lattice \mathbf{L} . That is, \uparrow assigns an L -set C^\uparrow in Y to a given L -set C in X , and \downarrow assigns an L -set D^\downarrow in X to a given L -set D in Y . $C^\uparrow(y)$ can verbally be described as a degree to which “for each object $x \in X$: if x is from C then x has attribute y ” (note that $C^\uparrow(y)$ is just the degree of the last statement “for each \dots ” according to basic principles of first-order fuzzy logic, see [12]). Likewise, $D^\downarrow(x)$ is the degree to

which “for each attribute $y \in Y$: if y is from D then x has attribute y ” is true. If $L = \{0, 1\}$, $\uparrow : L^X \rightarrow L^Y$ and $\downarrow : L^Y \rightarrow L^X$ coincide with the well-known concept-derivation operators of the basic setting of formal concept analysis [8, 11]. \uparrow and \downarrow form a fuzzy Galois connection [2] and the compound operators $\uparrow\downarrow$ and $\downarrow\uparrow$ form particular closure operators in X and Y [2]. A pair $\langle C, D \rangle$ consisting of an L -set C in X and an L -set D in Y is called a formal concept of I if $C^\uparrow = D$ and $D^\downarrow = C$. C and D are called the extent and intent of $\langle C, D \rangle$, respectively. The set of all formal concepts of I is denoted by $\mathcal{B}(X, Y, I)$. With a partial order \leq defined by

$$\langle C_1, D_1 \rangle \leq \langle C_2, D_2 \rangle \text{ iff } C_1 \subseteq C_2 \text{ (iff } D_2 \subseteq D_1)$$

for $\langle C_1, D_1 \rangle, \langle C_2, D_2 \rangle \in \mathcal{B}(X, Y, I)$, $\mathcal{B}(X, Y, I)$ happens to be a complete lattice, so-called concept lattice associated to I [2, 3]. Note that $C_1 \subseteq C_2$ means that C_1 is contained in C_2 , i.e. for each $x \in X$, $C_1(x) \leq C_2(x)$. For $L = \{0, 1\}$, $\mathcal{B}(X, Y, I)$ coincides with the ordinary concept lattice [11]. In [5], the following notion was introduced. A formal concept $\langle C, D \rangle \in \mathcal{B}(X, Y, I)$ is called *crisply generated* if there is a crisp L -set $D_c \in \{0, 1\}^Y$, i.e. for each $y \in Y$: $D_c(y) = 0$ or $D_c(y) = 1$, such that $C = D_c^\downarrow$ (and thus $D = D_c^{\downarrow\uparrow}$). Let $\mathcal{B}_c(X, Y, I)$ denote the collection of all crisply generated formal concepts of I , i.e.

$$\mathcal{B}_c(X, Y, I) = \{\langle C, D \rangle \in \mathcal{B}(X, Y, I) \mid \text{there is } D_c \in \{0, 1\}^Y : C = D_c^\downarrow\}.$$

We need the following characterization of crisply generated formal concepts. For L -sets $C_1, C_2 \in L^X$ and $D_1, D_2 \in L^Y$, we put $\langle C_1, D_1 \rangle \trianglelefteq \langle C_2, D_2 \rangle$ if for each $x \in X$, $y \in Y$ we have $C_1(x) \leq C_2(x)$ and $D_1(y) \leq D_2(y)$.

Lemma 1 ([5]). $\langle C, D \rangle$ is a crisply generated formal concept iff $\langle C, D \rangle$ is maximal (w.r.t. \trianglelefteq) such that (1) the rectangular matrix J defined by $J_{xy} = C(x) \otimes D(y)$ is contained in I (i.e. $J_{xy} \leq I_{xy}$ for all x, y) and (2) $C(x) = \bigwedge_{D(y)=1} I_{xy}$.

Remark 1. Note that condition (2) of Lemma 1 means that for the crisp L -set $D_c \in \{0, 1\}^Y$ corresponding to the 1-cut of D , which is defined by

$$D_c(y) = \begin{cases} 1 & \text{if } D(y) = 1, \\ 0 & \text{if } D(y) < 1, \end{cases} \quad (3)$$

we have $C = D_c^\downarrow$.

Matrices $A_{\mathcal{F}}$ and $B_{\mathcal{F}}$ For convenience, we identify $1 \times p$ vectors with entries from L with L -sets in $\{1, \dots, p\}$ (the l -th coordinate of the vector = the degree to which l belongs to the L -set). Given a set

$$\mathcal{F} = \{\langle C_1, D_1 \rangle, \dots, \langle C_k, D_k \rangle\}$$

of L -sets C_l and D_l in $\{1, \dots, n\}$ and $\{1, \dots, m\}$, respectively, with values from L , define $n \times k$ and $k \times m$ matrices $A_{\mathcal{F}}$ and $B_{\mathcal{F}}$ by

$$(A_{\mathcal{F}})_{il} = (C_l)(i) \quad \text{and} \quad (B_{\mathcal{F}})_{lj} = (D_l)(j).$$

That is, the l -th column of $A_{\mathcal{F}}$ is the transpose of the vector corresponding to C_l and the l -th row of $B_{\mathcal{F}}$ is the vector corresponding to D_l .

For $\mathcal{F} \subseteq \mathcal{B}(X, Y, I)$, denote

$$\mathcal{F}_c = \{\langle C, D_c \rangle \mid \langle C, D \rangle \in \mathcal{F}\}.$$

Note that D_c is defined by (3). We will show that sets \mathcal{F}_c corresponding to sets \mathcal{F} of crisply generated formal concepts are fundamental for decompositions we are looking for.

The first theorem says that for every I , there is a decomposition $A_{\mathcal{F}_c} \circ B_{\mathcal{F}_c}$ for some $\mathcal{F} \subseteq \mathcal{B}_c(X, Y, I)$.

Theorem 2 (universality). *For every I with entries from L there is $\mathcal{F} \subseteq \mathcal{B}_c(X, Y, I)$ such that $I = A_{\mathcal{F}_c} \circ B_{\mathcal{F}_c}$, i.e. I is a product of A with entries from L and B with entries from $\{0, 1\}$.*

Proof. Denote for $l \in \{1, \dots, m\}$, $\langle C_l, D_l \rangle = \langle \{1/l\}^\downarrow, \{1/l\}^{\downarrow\uparrow} \rangle$. Here, $\{1/l\}$ is a singleton in $\{1, \dots, m\}$, i.e. an L -set defined by $\{1/l\}(l) = 1$ and $\{1/l\}(j) = 0$ for $j \neq l$. $\langle C_l, D_l \rangle$ are particular crisply generated formal concepts from $\mathcal{B}(X, Y, I)$ and we have

$$I_{ij} = \bigvee_{l=1}^m C_l(i) \otimes D_l(j),$$

see [2]. Putting thus $\mathcal{F} = \{\langle C_l, D_l \rangle \mid l = 1, \dots, m\}$, we get $I = A_{\mathcal{F}_c} \circ B_{\mathcal{F}_c}$. \square

However, Theorem 2 and its proof yield only $|\mathcal{F}| = m$, i.e. the number $k = |\mathcal{F}|$ of factors equals the number m of attributes. In general, better decompositions may exist, i.e. those with $k < m$. The next theorem shows that the decompositions which use crisply generated formal concepts of I as factors are optimal among all decompositions of I .

Theorem 3 (optimality). *Let $I = A \circ B$ for $n \times k$ and $k \times m$ matrices A and B with $A_{il} \in L$, $B_{lj} \in \{0, 1\}$. Then there exists a set $\mathcal{F} \subseteq \mathcal{B}_c(X, Y, I)$ of crisply generated formal concepts of I such that for \mathcal{F}_c we have*

$$|\mathcal{F}_c| \leq k$$

and for the $n \times |\mathcal{F}_c|$ and $|\mathcal{F}_c| \times m$ matrices $A_{\mathcal{F}_c}$ with entries from L and $B_{\mathcal{F}_c}$ with entries from $\{0, 1\}$ we have

$$I = A_{\mathcal{F}_c} \circ B_{\mathcal{F}_c}.$$

Proof. Sketch: Let $I = A \circ B$ for an $n \times k$ matrix A with entries from L and a $k \times m$ binary matrix B . Consider the corresponding rectangular matrices J_1, \dots, J_k of which I is a \bigvee -superposition according to Theorem 1. Denoting now the L -sets in $\{1, \dots, n\}$ and $\{1, \dots, m\}$ corresponding to the l -th column of A and the l -th row of B by G_l and H_l , respectively, we have $J_l = G_l \otimes H_l$. We have $G_l \otimes H_l \subseteq I$ and one can check that also $H_l^\downarrow \otimes H_l \subseteq I$. The pair $\langle H_l^\downarrow, H_l \rangle$ satisfies condition (2) of Lemma 1 (see also Remark 1). Therefore, $\langle H_l^\downarrow, H_l \rangle$ is contained in a maximal (w.r.t. \sqsubseteq defined in the paragraph preceding Lemma 1) $\langle C_l, D_l \rangle$ which is then, according to Lemma 1, a crisply generated formal concept of I . As a result, $C_l \otimes D_l \subseteq I$. Therefore, for $\mathcal{F} = \{\langle C_1, D_1 \rangle, \dots, \langle C_k, D_k \rangle\}$ we have $|\mathcal{F}| \leq k$. Because $(H_l)_j \in \{0, 1\}$ and because we may assume $H_l \subseteq D_l$, we get

$H_l \subseteq (D_l)_c$, cf. (3). So,

$$\begin{aligned} I = A \circ B &= \bigvee_{l=1}^k G_l \otimes H_l \subseteq \bigvee_{l=1}^k H_l^\downarrow \otimes H_l \\ &\subseteq \bigvee_{l=1}^k C_l \otimes (D_l)_c = A_{\mathcal{F}_c} \circ B_{\mathcal{F}_c} \subseteq \bigvee_{l=1}^k C_l \otimes D_l = A_{\mathcal{F}} \circ B_{\mathcal{F}} \subseteq I, \end{aligned}$$

i.e. $A_{\mathcal{F}_c} \circ B_{\mathcal{F}_c} = I$, finishing the proof. \square

Note that using the notation from the proof of Theorem 3, two distinct $\langle G_l, H_l \rangle$'s may be contained in a single $\langle C_l, D_l \rangle$, i.e. for $\langle G_{l_1}, H_{l_1} \rangle \neq \langle G_{l_2}, H_{l_2} \rangle$ we can have $\langle C_{l_1}, D_{l_1} \rangle = \langle C_{l_2}, D_{l_2} \rangle$. As a consequence, we may have $|\mathcal{F}| < k$.

3 Algorithm

In this section, we present an approximation algorithm for computing a decomposition $I = A \circ B$ of an $n \times m$ matrix I with entries from L into an $n \times k$ matrix A with entries from L and a $k \times m$ binary matrix B with k as small as possible. Note that we do not provide the approximation factor for this algorithm.

Recall that for $L = \{0, 1\}$ (i.e. the set of grades contains just 0 and 1), our problem becomes a problem of decomposition of binary matrices. In particular, if $L = \{0, 1\}$, we are given a binary matrix I and our aim is to find a decomposition $I = A \circ B$ into an $n \times k$ binary matrix A and a $k \times m$ binary matrix B with k as small as possible. This problem is NP-hard and its decision version is NP-complete, see e.g. [17–19], and also [6].

Due to NP-hardness of a problem of decomposition of binary matrices which is a particular instance of our problem, we need to look for suitable approximation algorithms. In the following, we propose a greedy approximation algorithm inspired by the algorithms presented in [6] and [7]. Briefly, starting with empty \mathcal{F}_c , the algorithm selects a crisply generated concept $\langle C, D \rangle$ of I that covers a large part of I which is still uncovered. For each such selected $\langle C, D \rangle$, the corresponding $\langle C, D_c \rangle$, see (3), is added to \mathcal{F}_c . For determining $\langle C, D \rangle$, we use $|D \oplus j|$ which denotes the number of pairs $\langle i, j' \rangle$ of indices, for which $I_{ij'} = (I_{\mathcal{F}_c} \vee (D \cup \{1/j\})^\downarrow \otimes (D \cup \{1/j\})^{\downarrow\uparrow})_{ij'}$. We refer to this approach as to Method 1. We also used Method 2 for which $|D \oplus j|$ takes into account also entries $(I_{\mathcal{F}_c} \vee (D \cup \{1/j\})^\downarrow \otimes (D \cup \{1/j\})^{\downarrow\uparrow})_{ij'}$ which are close to $I_{ij'}$ but not necessarily equal (details will appear in a full version of this paper).

Note that if $L = \{0, 1\}$, our algorithm works the same way as the one from [6]. We performed several experiments with our algorithm. Due to limited scope, we present the following one. We generated 1,000 matrices I of dimension 15×15 over 5-element chain L with Lukasiewicz operations. Each matrix was generated as a product of a $15 \times k$ matrix A and a $k \times 15$ binary matrix B , so we knew the number of factors (its upper bound, in fact). Table 1 shows the numbers of factors (average value \pm standard deviation) for decompositions of I obtained by our algorithm (both for Methods 1 and 2).

Algorithm 1 Find Factors

Input: I (matrix with entries from L)
Output: \mathcal{F}_c (set \mathcal{F}_c for which $I = A_{\mathcal{F}_c} \circ B_{\mathcal{F}_c}$)
set $I_{\mathcal{F}_c}$ to empty matrix ($(I_{\mathcal{F}_c})_{ij} = 0$)
while $I \neq I_{\mathcal{F}_c}$ **do**
 set D to \emptyset
 set V to 0
 while there is j such that $D(j) < 1$ and $|D \oplus j| > V$ **do**
 select j such that $D(j) < 1$ which maximizes $|D \oplus j|$
 set D to $(D \cup \{^1/j\})^{\downarrow\uparrow}$
 set V to $|D \oplus j|$
 end while
 set C to D^{\downarrow}
 add $\langle C, D_c \rangle$ to \mathcal{F}_c
 set $I_{\mathcal{F}_c}$ to $I_{\mathcal{F}_c} \vee C \otimes D_c$
end while

Table 1. Number of computed factors

k	no. computed factors	no. computed factors
	Method 1	Method 2
4	5.294 ± 0.660	5.303 ± 0.712
5	7.204 ± 1.113	7.232 ± 1.063
6	8.964 ± 1.770	8.992 ± 1.688
7	10.194 ± 2.066	10.128 ± 1.990
8	11.155 ± 2.209	11.182 ± 2.067
9	11.747 ± 2.247	11.771 ± 1.878
10	12.18 ± 2.035	12.225 ± 2.054

4 Illustrative Example

In this section, we present an illustrative example regarding decompositions of a matrix with grades into a matrix with grades and a binary matrix.

In our example, we consider n users, m permissions, and a user-to-permission assignment. The assignment can be represented by an $n \times m$ matrix I with entries from a scale $L = \{0, r, w, 1\}$, with 0 representing “no permission”, r and w representing “permission to read” and “permission to write”, respectively, and 1 representing “full permission”. We define a partial order on L such that 0 is the least element, 1 is the greatest one, and elements r and w are incomparable, see Fig. 1.

Furthermore, we need to define operations of multiplication \otimes . We put $x \otimes y = x \wedge y$, for all $x, y \in L$. The residuum is then determined by \otimes (due to the requirement of adjointness, see Section 1) and is defined by $x \rightarrow y = 1$ for $x \leq y$, $x \rightarrow y = y$ for all $x > y$, and $r \rightarrow w = w$, $w \rightarrow r = r$.

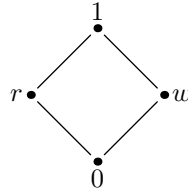


Fig. 1. Partial order on the scale of permissions

We want to decompose I into a product of $n \times k$ matrix A and $k \times m$ matrix B where A and B represent a user-to-role and a role-to-permission relationship, respectively. Therefore, the factors we want to discover are to be interpreted as roles, such as “system administrator”, “standard user” or the like. Naturally, we expect A to be a binary matrix (i.e. $A_{il} \in \{0, 1\}$), assigning roles to users (a user has a given role or not), whereas B is graded matrix (i.e. $B_{lj} \in L$). In order to be consistent with previous chapters, A should be graded and B should be binary matrix. Therefore, we use well-known fact that $I = A \circ B$ is equivalent to $I^{-1} = B^{-1} \circ A^{-1}$. That is, instead of I we decompose I^{-1} .

As a particular example, we consider 9 users (or employees) and 5 file-types in some computer system (for instance, “documents”, “archive files” or “system files” could be some of these types). The user-to-permission relationship is described in the table thereunder. The data can be visualized using a rectangular grid, where \square , \blacksquare , \blacktriangle , and \blacksquare represent permissions 0, r , w , and 1, respectively:

	type ₁	type ₂	type ₃	type ₄	type ₅
Alice	0	r	1	1	1
Bob	0	0	r	r	w
Charles	0	r	1	1	1
David	0	0	r	r	w
Eve	1	1	1	1	1
Frank	0	r	1	1	1
George	0	r	1	1	1
Henry	0	0	r	r	w
Isaac	0	0	r	r	w

Our aim is thus to decompose the corresponding graded matrix

$$I^{-1} = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ r & 0 & r & 0 & 1 & r & r & 0 & 0 \\ 1 & r & 1 & r & 1 & 1 & 1 & r & r \\ 1 & r & 1 & r & 1 & 1 & 1 & r & r \\ 1 & w & 1 & w & 1 & 1 & 1 & w & w \end{pmatrix}.$$

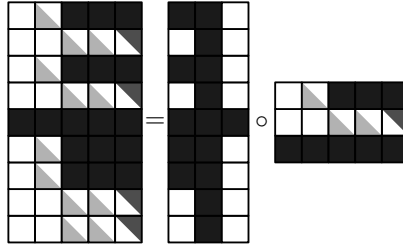
Algorithm 1 computes the following decomposition:

$$I^{-1} = \begin{pmatrix} 0 & 0 & 1 \\ r & 0 & 1 \\ 1 & r & 1 \\ 1 & r & 1 \\ 1 & w & 1 \end{pmatrix} \circ \begin{pmatrix} 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix},$$

i.e.,

$$I = A \circ B = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \end{pmatrix} \circ \begin{pmatrix} 0 & r & 1 & 1 & 1 \\ 0 & 0 & r & r & w \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}.$$

This decomposition can be displayed as:



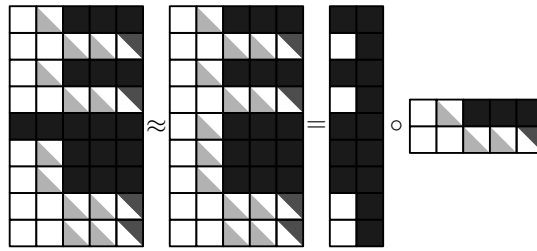
As we obtained a 9×3 binary matrix A describing a user-to-role assignment and 3×5 matrix B describing a role-to-permission assignment. Therefore, we obtained 3 factors: role_1 , role_2 , role_3 . The first role (corresponding to the first row of matrix B) might be interpreted as “standard user”, the second one (the middle row of B) as “anonymous user” (“guest”), and the third one (the last row of B) as “system administrator”.

According to matrix A , we assign roles to users by:

Alice - role_1 , role_2 ,
 Bob - role_2 ,
 Charles - role_1 , role_2 ,
 David - role_2 ,
 Eve - all roles,
 Frank - role_1 , role_2 ,
 George - role_1 , role_2 ,
 Henry - role_2 ,
 Isaac - role_2 .

Next, we compute an approximate decomposition of $I \approx A \circ B$. By this we mean that we want the entries of I to be similar to the corresponding entries of $A \circ B$ to a degree which exceeds a given similarity threshold f . In our example we set $f = 0.9$. Details regarding such similarity will be presented in a full version of this paper. Let us just note that the similarity is based on the number of matrix entries which have equal values in I and $A \circ B$. A graphical representation of an approximation decomposition computed by our algorithm is depicted below.

We can see that the approximate decomposition involves the two factors corresponding to “standard user” and “anonymous user”, which were involved



also in the exact decomposition. However, the factor corresponding to “system administrator” is no longer involved in the approximate decomposition. This can be seen as the result of our attempt, due to performing an approximate decomposition, to discover only a small number of factors (roles) which account for most of the data and, hence, are common. The role of “system administrator” is not common since the only user with this role is Eve.

5 Conclusions and Future Research

We presented a theorem regarding optimal decomposition of a matrix with grades into a matrix with grades and a binary matrix. Furthermore, we proposed a greedy approximation algorithm for computing such decompositions and examples illustrating such decompositions.

Further issues and future research include the following items:

- Independence of \otimes and \rightarrow . It can be shown that the decompositions of a graded matrix into a graded and a binary matrix do not depend, in a certain sense, on the operations \otimes and \rightarrow on the scale L of grades. We stuck to the framework which involves \otimes and \rightarrow to show how the problem addressed in this paper fits into the results developed earlier. Details will be presented in the full version of this paper.
- Decompositions of matrices with grades into matrices with further constraints, different from the requirement of binarity of B .
- Approximation algorithms for approximate and exact decompositions of matrices with grades.
- Applications of the underlying factor analysis model and comparison to other models of factor analysis.
- Role of decompositions in machine learning and data mining (esp. dimensionality reduction).

References

1. Bartholomew, D. J., Knott M.: *Latent Variable Models and Factor Analysis, 2nd Ed.*, London, Arnold, 1999.
2. Belohlavek, R.: *Fuzzy Relational Systems: Foundations and Principles*. Kluwer, Academic/Plenum Publishers, New York, 2002.

3. Belohlavek, R.: Concept lattices and order in fuzzy logic. *Annals of Pure and Applied Logic* **128**(1–3)(2004), 277–298.
4. Belohlavek, R.: Optimal decompositions of matrices with grades. *IEEE Intelligent Systems 2008* (to appear).
5. Belohlavek, R., Sklenar, V., Zacpal, J.: Crisply generated fuzzy concepts. In: B. Ganter and R. Godin (Eds.): ICFCA 2005, Lecture Notes in Artificial Intelligence **3403**, pp. 268–283, Springer-Verlag, Berlin/Heidelberg, 2005.
6. Belohlavek, R., Vychodil, V.: Discovery of optimal factors in binary data via a novel method of matrix decomposition (to appear). Preliminary version appeared as: On Boolean factor analysis with formal concepts as factors. SCIS & ISIS 2006, Int. Conf. Soft Computing and Intelligent Systems & Int. Symposium on Intelligent Systems, Sep 20–24, 2006, Tokyo, Japan, pp. 1054–1059.
7. Belohlavek, R., Vychodil, V.: Optimal decompositions of matrices with ordinal data (submitted).
8. Carpineto, C., Romano, G.: *Concept Data Analysis. Theory and Applications*. J. Wiley, 2004.
9. Cormen, T. H., Leiserson, C. E., Rivest, R. L., Stein, C.: *Introduction to Algorithms, 2nd Ed.* MIT Press, 2001.
10. Frolov, A. A., Húsek, D., Muraviev, I. P., Polyakov, P. A.: Boolean factor analysis by Hopfield-like autoassociative memory. *IEEE Transactions on Neural Networks* Vol. **18**, No. 3, May 2007, pp. 698–707.
11. Ganter, B., Wille, R.: *Formal Concept Analysis. Mathematical Foundations*. Springer, Berlin, 1999.
12. Hájek, P.: *Metamathematics of Fuzzy Logic*. Kluwer, Dordrecht, 1998.
13. Harman, H. H.: *Modern Factor Analysis, 2nd Ed.* The Univ. Chicago Press, Chicago, 1970.
14. Kepřt, A., Snášel, V.: Binary factor analysis with help of formal concepts. In *Proc. CLA 2004*, Ostrava, Czech Republic, 2004, pp. 90–101, ISBN 80–248–0597–9.
15. Klir, G. J., Yuan, B.: *Fuzzy Sets and Fuzzy Logic. Theory and Applications*. Prentice-Hall, 1995.
16. McDonald, R. P.: *Factor Analysis and Related Methods*. Lawrence Erlbaum Associates, Inc., 1985.
17. Nau, D. S.: Specificity covering: immunological and other applications, computational complexity and other mathematical properties, and a computer program. A. M. Thesis, Technical Report CS–1976–7, Computer Sci.Dept., Duke Univ., Durham, N. C., 1976.
18. Nau, D. S., Markowsky, G., Woodbury M. A., Amos D. B.: A Mathematical Analysis of Human Leukocyte Antigen Serology. *Math. Biosciences* **40**(1978), 243–270.
19. Stockmeyer, L. J.: The set basis problem is NP-complete. IBM Research Report RC5431, Yorktown Heights, NY, 1975.
20. Wille, R.: Restructuring lattice theory: an approach based on hierarchies of concepts. In: I. Rival (Ed.): *Ordered Sets*, 445–470, Reidel, Dordrecht-Boston, 1982.