



Boolean Decomposition of Spatiotemporal Tensors

By Crystal Chen, Charlotte L. Ellison, Zachary J. Roth, and Mackenzie A. Simper

INTRODUCTION: This technical note (TN) discusses Boolean decompositions of binary tensors containing spatiotemporal data. The increase in data collection technology has led to a proliferation of data containing information on human movement. Effectively managing and gaining insights from this influx of data is presenting a significant challenge to the Army geospatial community. This information can extend beyond just time and position coordinates to include features such as weather, social interactions, etc. The addition of these extra variables can provide new insight into human behavior and patterns of life. The capability to geocomputationally manipulate these related datasets will provide additional insight into human behavior and patterns of life, thereby enhancing the Army GEOINT-HUMINT spectrum of operations. Since spatiotemporal data is often binary, new theory is needed to fully exploit this information and draw conclusions.

A uniform framework is needed to combine the spatiotemporal moving object information, along with other contextual data, before a holistic data analysis can be applied. A tensor is a data structure that is a natural representation for many multi-modal datasets. To extract patterns and find correlations in the data, various decompositions of the tensor can be studied. A commonly used method is the CANDECOMP/PARAFAC (CP) decomposition, which expresses a tensor as a minimum-length linear combination of rank-one tensors (Kolda and Bader 2009). This decomposition always exists, but is computationally difficult to determine. Thus an *approximate* decomposition is used. In addition to being computationally efficient, an approximation can serve to separate noise from signal by preserving the most important features of a tensor.

This idea has been applied to fields such as signal processing, acoustics, and chemometrics (Kolda and Bader 2009). Thus far, most tensor analysis, both in theory and in applications, has been done using real or complex numbers. In common spatiotemporal data, however, the information is discretized and stored in a *binary* format. For example, a tensor representing (user, position, time) data could have a 1 in the (i, j, k) entry if “user i is at position j at time k ” and a 0 otherwise.

When an approximate CP decomposition is performed over the real numbers, the reconstructed approximate tensor may not be binary. In addition, the terms in the decomposition do not have a clear interpretation within the original tensor. This motivates the study of different decomposition methods. A recent approach is the Boolean CP decomposition, as discussed by Miettinen 2011. Boolean decomposition of *matrices* has been analyzed as an alternative to data clustering, but less has been studied for tensors (Cergani and Miettinen 2013).



The Boolean CP decomposition works by restricting the factor tensors to binary values. The factor tensors are then summed using Boolean algebra. This can result in a very different factorization (or approximate factorization) than real-valued CP decomposition. In particular, the structure of binary rank-one tensors results in new numerical algorithms for determining approximate decompositions. A connection between binary tensors and hypergraphs allows the study of tensor decomposition from a combinatorial perspective, rather than an algebraic one.

In this TN, the Boolean CP decomposition of a binary tensor is defined. Various properties are discussed and the interpretation of a binary tensor as a hypergraph is explained, which provides insights into Boolean decompositions. Finally, numerical algorithms for finding the Boolean decomposition are discussed. The possibility of using the results to draw conclusions from spatiotemporal data tensors is emphasized throughout.

BACKGROUND: In this section, tensors and the necessary terminology related to CP decomposition are defined. For more details, see Kolda and Bader (2009). Initially, real-valued tensors are introduced, though the following quantities can be defined over any field.

Definition

- A *tensor* is an element $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ (i.e., $\mathcal{X}$ is a multidimensional array indexed by N sets: $\mathcal{X} = (x_{i_1, i_2, \dots, i_N})_{i_1=1, \dots, i_N=1}^{I_1, \dots, I_N}$). The tensor, $\mathcal{X}$ is N -way or has *order* N .

To visualize a tensor one can look at the *slices*, which are the sub-tensors obtained by fixing a variable in one mode. For example, for a 3-way tensor, the slices are matrices $\mathbf{X}_{i,:,:}$, $\mathbf{X}_{:,j,:}$, $\mathbf{X}_{:,:,k}$. The standard norm used in tensor analysis is the *Frobenius norm* $\|\cdot\|_F$ defined as

$$\|\mathcal{X}\|_F = \sqrt{\sum_{i_1=1}^{I_1} \dots \sum_{i_N=1}^{I_N} x_{i_1, i_2, \dots, i_N}^2}.$$

Definition

- An N -way tensor $\mathcal{X}$ is called *rank-one*, if it can be written as the outer-product of vectors. That is, there exists vectors $\mathbf{a}^{(j)} \in \mathbb{R}^{I_j}$ for $1 \leq j \leq N$ such that

$$x_{i_1, i_2, \dots, i_N} = \mathbf{a}_{i_1}^{(1)} \cdot \mathbf{a}_{i_2}^{(2)} \cdot \dots \cdot \mathbf{a}_{i_N}^{(N)}.$$

The outer product is denoted $\mathcal{X} = \mathbf{a}^{(1)} \circ \mathbf{a}^{(2)} \circ \dots \circ \mathbf{a}^{(N)}$.

Since rank-one tensors form a basis for the vector space $\mathbb{R}^{I_1 \times I_2 \times \dots \times I_N} \simeq \mathbb{R}^{I_1} \otimes \mathbb{R}^{I_2} \otimes \dots \otimes \mathbb{R}^{I_N}$, every N -way tensor can be written as a linear combination of rank-one tensors. The minimal number of rank-one tensors required gives the definition of the rank of a tensor.

Definition

- The *rank* of an N -way tensor $\mathcal{X}$, written $\text{rank}_{\mathbb{R}}(\mathcal{X})$, is the least integer r such that there exists r rank-one tensors that sum to $\mathcal{X}$. That is, there exists vectors $\mathbf{a}_i^{(j)} \in \mathbb{R}^{I_j}$ for $1 \leq j \leq N$, $1 \leq i \leq r$ such that

$$\mathcal{X} = \sum_{i=1}^r \mathbf{a}_i^{(1)} \circ \dots \circ \mathbf{a}_i^{(N)}.$$

The remainder of the paper specializes to 3-way tensors $\mathcal{X} \in \mathbb{R}^{I \times J \times K}$, though definitions and results generalize to higher orders. The letters $\mathbf{a}$, $\mathbf{b}$, $\mathbf{c}$ are used for the three vectors appearing in a rank-one tensor (i.e., $\mathcal{X} = \mathbf{a} \circ \mathbf{b} \circ \mathbf{c}$).

Definition

- Given an integer r , an approximate *CP decomposition* of rank r of $\mathcal{X} \in \mathbb{R}^{I \times J \times K}$ are vectors $\mathbf{a}^{(i)} \in \mathbb{R}^I$, $\mathbf{b}^{(i)} \in \mathbb{R}^J$, $\mathbf{c}^{(i)} \in \mathbb{R}^K$ for $1 \leq i \leq r$ that minimize

$$\|\mathcal{X} - \sum_{i=1}^r \mathbf{a}^{(i)} \circ \mathbf{b}^{(i)} \circ \mathbf{c}^{(i)}\|_F.$$

The reconstructed tensor is denoted $\mathcal{X}^r = \sum_{i=1}^r \mathbf{a}^{(i)} \circ \mathbf{b}^{(i)} \circ \mathbf{c}^{(i)}$. The reconstruction error is defined as $\frac{\|\mathcal{X} - \mathcal{X}^r\|_F}{\|\mathcal{X}\|_F}$.

The terms $\mathbf{a}^{(i)} \circ \mathbf{b}^{(i)} \circ \mathbf{c}^{(i)}$ in the sum are called *components*. Matrices, $\mathbf{A} \in \mathbb{R}^{I \times r}$, $\mathbf{B} \in \mathbb{R}^{J \times r}$, $\mathbf{C} \in \mathbb{R}^{K \times r}$ can also be created by combining the different modes from each component; specifically, $\mathbf{A} = [\mathbf{a}^{(1)} \ \mathbf{a}^{(2)} \ \dots \ \mathbf{a}^{(r)}]$ and $\mathbf{B}$, $\mathbf{C}$ are defined analogously. These are called the *factor matrices*. Note that a CP decomposition of rank r of a given tensor is not necessarily unique.

For a given tensor there is no efficient, universal algorithm for computing the rank of a tensor; it is computationally very expensive (NP-hard). To compute the rank of a tensor, one must find approximate rank- r decompositions, increasing r until a perfect reconstruction is found (Kolda and Bader 2009). In practice, it is not feasible to test a large range of r . Thus, for most applications, a lower-rank approximation is found and analyzed. The question then arises, what choice should be made for r ? Increasing r results in a trade-off between reconstruction error and computation time.

The question is further complicated if the data is expected to contain noise, either from missing entries or extraneous values. In this case, the goal in using an approximate decomposition is that only the most important features are preserved. An approximation must be sufficiently accurate to retain any important information, but not so accurate that the noise is preserved. The distinction between information and noise is dependent on the data and desired applications. Using a human trajectory example, a person deviating slightly and insignificantly from their usual path might be considered noise, if the goal is to determine the larger pattern of their movements.

Finally, there is a question of whether any meaningful interpretation can be made of the component tensors or factor matrices, or whether the only use of the decomposition is in studying the reconstructed tensor $\mathcal{X}^r$. The next sections discuss these questions.

APPROACH: In this section, Boolean CP decomposition and the connection with hypergraphs is introduced. Suppose $\mathcal{X} \in \mathbb{R}^{I \times J \times K}$ is a binary tensor, (i.e., the only entries are 0 and 1). Before defining a Boolean decomposition, first observe that a binary tensor can be viewed as an adjacency matrix for a specifically structured hypergraph. Knowledge of basic graph theory is assumed.

Definition

- A *hypergraph* is a pair $H = (V; E)$, where, V is a set of elements called *vertices*, and E is a set of non-empty subsets of V , called *hyperedges*. If every hyperedge $e \in E$ has N elements, then H is called *N -uniform*.

Definition

- An *N -uniform, N -partite hypergraph* $H = (V_1, \dots, V_N; E)$ is a hypergraph with N disjoint vertex sets $V_1, \dots, V_N$ such that every hyperedge $e \in E$ contains exactly one element from each V_i . That is, an edge connects exactly N vertices, one from each vertex set.

An N -way binary tensor defines an N -uniform, N -partite hypergraph, in which each vertex set corresponds to a mode in the tensor and edges correspond to non-zero entries. Specifically, for a 3-way tensor the corresponding hypergraph $H_{\mathcal{X}} = (U, V, W; E)$ is defined as follows: Let $U = \{u_1, \dots, u_I\}$, $V = \{v_1, \dots, v_J\}$, $W = \{w_1, \dots, w_K\}$. Then edges are specified as

$$\{u_i, v_j, w_k\} \in E \Leftrightarrow x_{i,j,k} = 1.$$

As an example, consider the tensor $\mathcal{X} \in \{0,1\}^{3 \times 3 \times 2}$ with slices

$$\mathbf{X}_{:, :, 1} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \mathbf{X}_{:, :, 2} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}.$$

Since there are three non-zero entries, there are three edges in the hypergraph, as shown in Figure 1: $\{u_1, v_1, w_1\}$, $\{u_2, v_3, w_2\}$, and $\{u_3, v_1, w_1\}$.

$$v_1 \quad v_2 \quad v_3$$

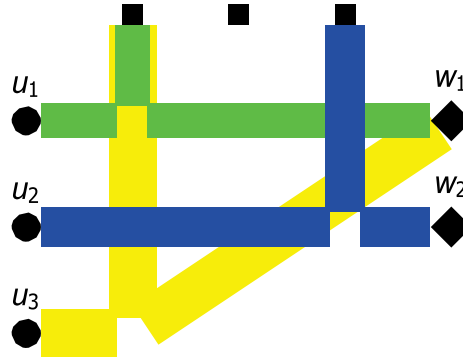


Figure 1. An example of a 3-uniform, 3-partite hypergraph defined by a 3-way tensor.

Within the hypergraph visualization, a crucial observation is the structure of a binary rank-one tensor. First, the notion of a *complete* hypergraph is introduced; a *complete* hypergraph is one which contains every possible hyperedge connecting the vertices.

Definition

- Let $H = (U, V, W; E)$ be a 3-uniform, 3-partite hypergraph. Then H is called *complete* if $\{u, v, w\} \in E$ for every $u \in U, v \in V, w \in W$.

For a given hypergraph $H = (U, V, W; E)$, let $U^+ = \{u \in U : u \in e \text{ for some } e \in E\}$, and define V^+ and W^+ analogously. Define $H^+ = (U^+, V^+, W^+; E)$. That is, H^+ is just H with the isolated vertices (vertices not belonging to any edge) removed. A characterization of binary rank-one tensors can be proven.

Proposition

- A binary tensor $\mathcal{X}$ is rank-one if and only if the corresponding hypergraph $H_{\mathcal{X}}^+$ is complete.

Proof Given a rank-one binary tensor, $\mathcal{X} \in \{0,1\}^{I \times J \times K}$ define the sets

$$A_{\mathcal{X}} = \{i : x_{i,j,k} = 1 \text{ for some } j, k\} \subset \{1, \dots, I\},$$

$$B_{\mathcal{X}} = \{j : x_{i,j,k} = 1 \text{ for some } i, k\} \subset \{1, \dots, J\},$$

$$C_{\mathcal{X}} = \{k : x_{i,j,k} = 1 \text{ for some } i, j\} \subset \{1, \dots, K\}.$$

Note that $A_{\mathcal{X}}, B_{\mathcal{X}}, C_{\mathcal{X}}$ correspond to the vertex set U^+, V^+, W^+ . If $\mathcal{X} = \mathbf{a} \circ \mathbf{b} \circ \mathbf{c}$, then necessarily

$$a_i = 1 \text{ for all } i \in A_{\mathcal{X}},$$

$$b_j = 1 \text{ for all } j \in B_{\mathcal{X}},$$

$$c_k = 1 \text{ for all } k \in C_{\mathcal{X}}.$$

Note that $x_{i,j,k} = a_i \cdot b_j \cdot c_k = 1$ if and only if $a_i = b_j = c_k = 1$. Thus, $\mathcal{X}$ has binary rank-one if and only if $x_{i,j,k} = 1$ for all $i \in A_{\mathcal{X}}, j \in B_{\mathcal{X}}, k \in C_{\mathcal{X}}$. This means that $\{u, v, w\} \in E$ for all $u \in U^+, v \in V^+, w \in W^+$. Thus $H_{\mathcal{X}}^+$ is a complete hypergraph if and only if $\mathcal{X}$ is rank-one. $\square$

Thus, the variables involved (i.e., the variables corresponding to U^+ , V^+ , and W^+) in a binary rank-one tensor are highly connected. Depending on the data represented in the tensor, this could have an important meaning; for instance, “users 1 and 2 are at position 1 at times 1, 2, and 3.” For the real-valued CP decomposition of binary tensors with rank larger than one, this structure of binary rank-one tensors is not useful, since the individual component tensors in the decomposition do not necessarily have binary values. That is, by definition each component term in the decomposition is rank-one, but not necessarily *binary rank-one*. This motivates the idea of restricting the rank-one components in the decomposition of binary tensors to binary values.

If one restricts the rank-one terms to binary values, there is then the question of how to “sum” the terms. One choice would be to work over the field $\mathbb{F}_2$. Decomposing over this field would require that every non-zero entry appear in an odd number of terms in the factorization, an unusual restriction that has no obvious intrinsic meaning. Thus, a more natural choice is to sum using Boolean algebra (i.e., $1 + 1 = 1$). With this system, a non-zero entry could appear in an unrestricted number of rank-one terms.

The Boolean rank and approximate Boolean CP decomposition is defined analogously in the previous section:

Definition

- The *Boolean rank* of a 3-way tensor $\mathcal{X}$, written $\text{rank}_B(\mathcal{X})$, is the least integer r such that there exists vectors $\mathbf{a}^{(i)} \in \{0,1\}^I$, $\mathbf{b}^{(i)} \in \{0,1\}^J$, $\mathbf{c}^{(i)} \in \{0,1\}^K$ for $1 \leq i \leq r$ such that

$$\mathcal{X} = \bigvee_{i=1}^r \mathbf{a}^{(i)} \circ \mathbf{b}^{(i)} \circ \mathbf{c}^{(i)},$$

where $\vee$ denotes the logical *or* operator.

Definition

- Given an integer r , the Boolean CP decomposition of rank r of $\mathcal{X} \in \mathbb{R}^{I \times J \times K}$ are the vectors $\mathbf{a}^{(i)} \in \{0,1\}^I$, $\mathbf{b}^{(i)} \in \{0,1\}^J$, $\mathbf{c}^{(i)} \in \{0,1\}^K$ for $1 \leq i \leq r$ that minimizes

$$\|\mathcal{X}_B^r \bigvee \mathcal{X}\|_F,$$

where $\mathcal{X}_B^r = \bigvee_{i=1}^r \mathbf{a}^{(i)} \circ \mathbf{b}^{(i)} \circ \mathbf{c}^{(i)}$ is a Boolean rank r binary tensor.

Note that the Frobenius norm of a binary tensor is simply the square root of the number of non-zero entries, or, equivalently, the number of edges in the corresponding hypergraph. The metric induced by the Frobenius norm measures the number of non-equal entries between two binary tensors. In the real case there are examples of tensors that are degenerate in the sense that they can be approximated arbitrarily closely by lower-rank tensors (Kolda and Bader 2009). Because of the discrete error, this arbitrary close approximation is impossible for a Boolean decomposition.

Structure of Components in Decomposition. Another large difference between Boolean and real-valued tensor decomposition is the interpretability of the rank-one component tensors. The structure “binary rank-one tensor” is the same whether the tensor is considered in real vector space or Boolean vector space. When decomposing a binary tensor over the real number, the components may no longer be binary. The critical distinction of Boolean decomposition is that the *components are rank-one binary tensors*. Due to the Boolean addition, an exact Boolean decomposition corresponds to a collection of rank-one *sub-tensors* of the original tensor. For an exact decomposition, every non-zero entry must occur in at least one component tensor in the collection. This can be visualized in the hypergraph, as proven in the following lemma.

Lemma

- Let $\mathcal{X}$ be a binary tensor and $\mathcal{Y}_1, \dots, \mathcal{Y}_r$ binary rank-one tensors such that

$$\mathcal{X} = \mathcal{Y}_1 \vee \mathcal{Y}_2 \vee \dots \vee \mathcal{Y}_r.$$

Then the corresponding hypergraphs $H_{\mathcal{Y}_1}, \dots, H_{\mathcal{Y}_r}$ are sub-graphs of $H_{\mathcal{X}}$. Each $H_{\mathcal{Y}_i}^+$ is complete and every edge in $H_{\mathcal{X}}$ is an edge in at least one of the $H_{\mathcal{Y}_i}$.

Thus, the problem of finding Boolean decomposition of a binary tensor is equivalent to finding a covering of a hypergraph by complete sub-hypergraphs. Specifically, to find a *minimal* such covering. The least number of sub-hypergraphs necessary in the covering is equal to the Boolean rank. For example, the hypergraph in Figure 1 can be covered by 2 sub-graphs $H_1 = (U, V, W; E_1)$, $H_2 = (U, V, W; E_2)$ such that H_1^+, H_2^+ are complete. Here, $E_1 = \{(u_1, v_1, w_1), (u_3, v_1, w_1)\}$ and $E_2 = \{(u_2, v_3, w_2)\}$. Thus, the corresponding binary tensor has Boolean rank 2.

A binary rank-one tensor could have significant meaning for real spatiotemporal data. Indeed, in a rank-one binary tensor every variable with a non-zero entry in its mode (every non-isolated vertex in the hypergraph) is highly connected with the other variables. Thus, due to the structure of binary rank-one tensors, each rank-one sub-tensor in the Boolean decomposition is evidence of a potentially interesting pattern in the data. For example, suppose the tensor shown in Figure 1 represented users $\times$ position $\times$ time (i.e., the users are $\{u_1, u_2\}$, the positions are $\{v_1, v_2, v_3\}$ and the times are $\{w_1, w_2\}$). Then the first term in the Boolean decomposition, represented by H_1^+ , says that u_1 and u_2 are both at the same position at the same time.

For standard graphs (in which every edge connects two vertices), the analogous problem is to find the smallest number of complete sub-graphs such that every edge is covered by at least one graph. The number required is called the *intersection number*, or *clique covering number*. Erdos et al. (1966) proves that any graph on n vertices with no isolated points can be covered by at most $\lceil n^2/4 \rceil$ complete sub-graphs; an analogy for 3-partite, 3-uniform hypergraphs is proven in the next section. The connection between the Boolean rank of matrices/tensors and the covering of a graph by complete sub-graphs has been previously

recognized; however, the connection can be exploited further to answer questions about binary tensors (Miettinen 2013).

THEORY: In this section, several properties of Boolean rank and decomposition are presented, comparing with the corresponding real decomposition. Throughout, the connection with hypergraphs and ability to use combinatorial ideas is emphasized.

Boolean Rank Bounds. The theoretical study of bounds on possible Boolean rank is important, as some knowledge of the Boolean rank of a binary tensor can improve the computation time of exact or approximate Boolean CP decomposition. A first result is an upper bound on the Boolean rank of a 3-way tensor. The result was recorded by Miettinen (2011) and the proof worked by explicitly calculating a factorization for a given tensor. Here, a proof utilizing the hypergraph perspective is presented.

Proposition

- Let $\mathcal{X} = \{0,1\}^{I \times J \times K}$ be any 3-way binary tensor. Then

$$\text{rank}_B(\mathcal{X}) \leq \min\{IJ, JK, IK\}.$$

Proof. Without loss of generality, suppose $IJ = \min\{IJ, JK, IK\}$. Let $H_{\mathcal{X}} = (U, V, W; E)$ be the 3-partite, 3-uniform hypergraph defined by $\mathcal{X}$. That is, $|U| = I$, $|V| = J$, $|W| = K$. For each $u \in U$, $v \in V$, if there exists an edge $e \in E$ such that $u, v \in e$ (in the tensor, there is some k such that $x_{u,v,k} = 1$) define a hypergraph $H_{(u,v)} = (U, V, W; E_{(u,v)})$, with edge set

$$E_{(u,v)} = \{\{u, v, w\} : \{u, v, w\} \in E, w \in W\}.$$

Thus every edge in $H_{(u,v)}$ is an edge in $H_{\mathcal{X}}$. The only non-isolated vertices from U and V in $H_{(u,v)}$ are u, v respectively; by construction, $H_{(u,v)}^+$ is complete. There are at most IJ such hypergraphs $H_{(u,v)}$ and the collection of all covers every edge in $H_{\mathcal{X}}$. Thus, the collection corresponds to an exact Boolean decomposition of $\mathcal{X}$. This proves that Boolean rank of $\mathcal{X}$ is at most IJ . $\square$

The same bound holds for the real-valued tensor decomposition of 3-way tensors. However, for certain sizes I, J, K , the bound is not sharp; for example, the maximum rank of a $3 \times 3 \times 3$ tensor is 5 according to Kruskal (1989). This phenomenon does not occur for the Boolean decomposition.

Lemma

- For any integers $I, J, K \geq 1$ there exists a tensor $\mathcal{X} \in \{0,1\}^{I \times J \times K}$ with Boolean rank exactly $\min\{IJ, JK, IK\}$.

Proof. Without loss of generality, suppose that $I \leq J \leq K$, so that $IJ = \min\{IJ, JK, IK\}$. The frontal slices of a tensor $\mathcal{X}$ will be specified. That is, the frontal slices $\mathbf{X}_1, \dots, \mathbf{X}_K$ are the $I \times J$ matrices defined by $\mathbf{X}_l = \mathcal{X}_{:, :, l}$. For the first J slices $\mathbf{X}_l$, the I non-zero entries are assigned. This will be done such that no two entries create a rank-one sub-tensor of $\mathcal{X}$; thus, each entry gives a single component in the exact Boolean decomposition of $\mathcal{X}$. Note that two individual non-zero entries $x_{i,j,k}$ and $x_{l,m,n}$ form a rank-one sub-tensor, if and only

if exactly one of $i \neq l$, $j \neq m$, or $k \neq n$ is true. Thus, the need to assign non-zero entries such that no two entries share exactly two mode coordinates exists.

To define $\mathbf{X}_1$, place a single 1 in each row such that no two 1s are in the same column. This is possible since $I \leq J$. For $2 \leq l \leq J$, $\mathbf{X}_l$ is defined recursively as follows: If the (i, j) entry of $\mathbf{X}_{l-1}$ equals 1, then set the $(i, (j + 1) \bmod (J + 1))$ entry of $\mathbf{X}_l$ to be 1. Set all other entries of $\mathbf{X}_l$ to be 0. That is, every non-zero entry of $\mathbf{X}_{l-1}$ is shifted to the right by 1, modulo $(J + 1)$. If this shifting is done J times, then $\mathbf{X}_1$ is recovered.

Set $\mathbf{X}_l$ to be the zero matrix for $J+1 \leq l \leq K$. Then by construction, $\mathcal{X}$ has Boolean rank IJ . $\square$

An example of the construction shown in the previous proof is given below for a $2 \times 3 \times 3$ tensor

$$\mathbf{x}_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}, \mathbf{x}_2 = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \mathbf{x}_3 = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}.$$

Typical Boolean Rank Now that the range of possible values for the Boolean rank of a tensor is known, a natural question is: what is the “usual” Boolean rank? That is, if a binary tensor is chosen randomly from some distribution, what are the statistics of the Boolean rank? A natural distribution for real-valued tensors is to take each entry independently from the standard normal distribution. With this model, there is a notion of *typical rank*, any value that occurs with probability greater than 0. Thus, non-zero probability corresponds to a subset of $\mathbb{R}^{I \times J \times K}$ with positive Lebesgue measure. Mathematicians have used algebraic geometry to find typical rank for some specific sizes (Friedland 2012).

For $I \times J$ real-valued matrices, the maximum and typical rank are both $\min\{I, J\}$. This is easily seen, since a standard normal vector is a linear combination of a finite number of fixed vectors with probability 0. For tensors with three or more modes, the typical ranks are often smaller than the maximum ranks; a table of known values is recorded by Kolda and Bader (2009). In the specific cases that have been studied over $\mathbb{R}$, at most two typical ranks exist.

The typical rank results do not extend to the real-valued decomposition of binary tensors, since binary tensors form a set of Lebesgue-measure 0. Thus, new theory must be developed. A natural model for a random binary tensor is to draw each entry independently from a Bernoulli(p) distribution, where $0 < p < 1$ is some parameter that determines the average density of the binary tensor. One can ask what are the average ranks of these binary tensors when decomposed over $\mathbb{R}$, then observe if these values correspond to the typical ranks of continuous tensors. Current research shows that this question has not been studied.

Due to the combinatorial nature of Boolean decomposition, it is easier to study the average *Boolean rank* of random binary tensors. A first observation is that, unlike in the continuous case, *every* rank up to the maximum rank occurs with non-zero probability. As an example, the probability distribution of Boolean rank for binary tensors of size $2 \times 2 \times 2$ is calculated. Table 1 records the number of tensors of each Boolean rank, organized by the number k of non-zero entries in the tensor. This is done since, according to the random model, every tensor with k non-zero entries has the same probability of occurring: $p^k \cdot (1 - p)^{8-k}$.

Table 1. The number of binary $2 \times 2 \times 2$ tensors of each Boolean rank. The first column k is the number of non-zero entries in the tensor. The zero tensor is excluded.

k	number of tensors	rank 1	rank 2	rank 3	rank 4
1	8	8	0	0	0
2	$\binom{8}{2} = 28$	16	12	0	0
3	$\binom{8}{3} = 56$	0	40	16	0
4	$\binom{8}{4} = 70$	6	12	48	4
5	$\binom{8}{5} = 56$	0	24	24	8
6	$\binom{8}{6} = 28$	0	12	26	0
7	$\binom{8}{7} = 8$	0	0	8	0
8	1	1	0	0	0
total	225	31	100	112	12

The probabilities of each Boolean rank depend on the choice of p . For example, the probability of rank 2 is equal to

$$\sum_{k=1}^8 (\text{\#of tensors with } k \text{ nonzero entries and rank 2}) \cdot \mathbb{P}(\text{tensor has } k \text{ nonzero entries}) = 12 \cdot p^2 \cdot (1-p)^6 + 40 \cdot p^3 \cdot (1-p)^5 + 12 \cdot p^4 \cdot (1-p)^4 + 24 \cdot p^5 \cdot (1-p)^3 + 12 \cdot p^6 \cdot (1-p)^2.$$

When $p = 1/2$ every possible tensor is equally likely; in this case rank 2 and rank 3 tensors have the highest probability. The probability of Boolean rank 2 is $100/256 \approx 0.39$ and the probability of Boolean rank 3 is $112/256 \approx 0.4375$. Kruskal (1989) notes that the typical ranks of $2 \times 2 \times 2$ tensors are approximated using Monte Carlo simulations. The probabilities for a $2 \times 2 \times 2$ tensor to have a rank of 2 or 3 are approximately 79% and 21%, respectively. Through an argument using the expected value of generalized eigenvalues, the exact probability of a $2 \times 2 \times 2$ tensor having rank 2 is proven to be $\pi/4$ (Bergqvist 2013).

Though the math is elementary, the combinatorics become more complicated for larger sized tensors. Current research shows that no work has been published investigating the rank (either Boolean or real-valued) of random binary tensors. Since random graph modeling is a well-studied domain, the hypergraph perspective could provide an avenue to begin the study of random binary tensors.

NUMERICAL METHODS: In this section, numerical methods for approximate Boolean CP decomposition of binary tensors are discussed. A recently developed method called WalkNMerge

is an example of how Boolean decomposition provides a framework for new methods with potential advantages over real-valued decompositions (Miettinen 2013).

To find a real-valued approximate decomposition of a tensor, the standard approach is an alternating least-squares minimization. This works by fixing two of the factor matrices and minimizing over the third. The same method can be used to find an approximate Boolean decomposition. Details are discussed by Miettinen (2013), and approximate Boolean CP decompositions are shown to result in a reconstruction error similar to standard approximate CP decomposition.

Due to the structure of the Boolean decomposition, methods other than alternating least squares are possible. The WalkNMerge algorithm is one such method with promising applications. The algorithm works by performing a random walk on the *fiber graph* defined by the tensor; each non-zero entry $x_{i,j,k} = 1$ defines a vertex, and two distinct vertices $x_{i,j,k}$ and $x_{l,m,n}$ are connected by an edge if and only if exactly one of $i \neq l$, $j \neq m$, or $k \neq n$ is true (i.e., the two entries differ in exactly one mode). While rank-one tensors do not correspond to complete graphs in this graph, they do correspond to highly connected clusters. In particular, in the graph corresponding to a rank-one tensor every vertex will be connected by a path of length less than three steps.

A large advantage of the WalkNMerge algorithm over a least-squares method is that there is no need to specify a rank for the approximation. Instead, manipulating the parameters (such as the length of the random walk, the number of walks performed) will result in differently sized and structured Boolean approximations. It seems difficult in general to know what parameters will result in the best approximation (measured by reconstruction error) without having some *a priori* knowledge of the structure of the tensor (e.g., the usual size and density of the rank-one sub-tensors). However, in practice this may be an advantage; by manipulating the parameters, one can get different approximate decompositions that reveal different features of the data.

Approximation with Noise. Another advantage of WalkNMerge (or other algorithms that exploits the graph structure) is the ability to correct destructive noise in the form of missing values. That is, if a large sub-tensor is almost binary rank-one except for a few missing values, a random walk will discover all entries surrounding the missing values. The algorithm then can recognize the instance of an almost rank-one sub-tensor.

Figure 2 is an illustration of this concept. A $4 \times 4 \times 4$ binary rank-one tensor was constructed and embedded within a $20 \times 20 \times 20$ tensor. Then ϵ proportion of the 64 non-zero entries are randomly chosen and removed, with ϵ ranging from 0 to 0.5. The WalkNMerge algorithm is performed on the noisy tensor. The parameters are fixed and remain the same for each trial. For each ϵ , the trial is repeated ten times and the average accuracy recorded. Here, accuracy is measured as the proportion of the original 64 entries that are recovered in the decomposition (rather than the proportion of the entries from the noisy tensor that are recovered by the algorithm). In this specific setting the algorithm is able to completely recover the latent rank-one tensor up to about 1/5 of missing entries; after that point the accuracy decreases roughly linearly as the number of errors increase.

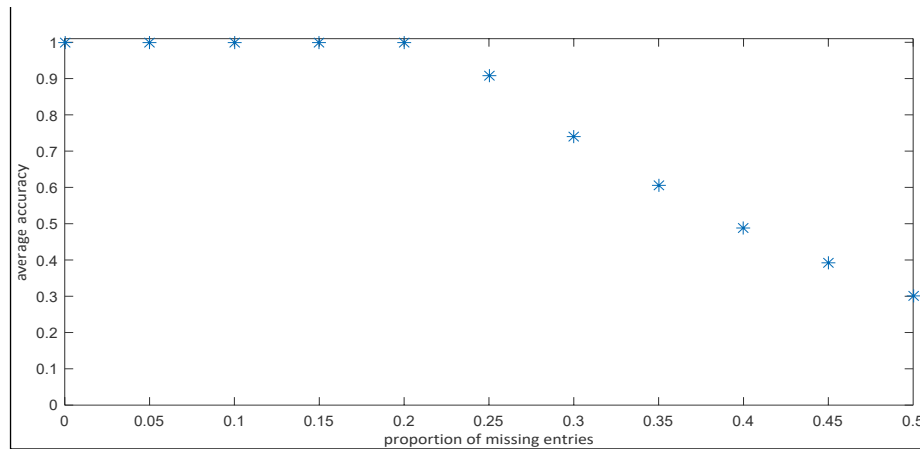


Figure 2. The average accuracy of the WalkNMerge algorithm over 10 trials vs. the proportion of entries removed from a $4 \times 4 \times 4$ tensor. The accuracy is defined as the proportion of the original 64 non-zero entries that are recovered.

CONCLUSIONS: For data that contains several modalities, such as geospatially referenced data, a tensor representation is a natural choice. This technical note defined and discussed a Boolean decomposition of binary tensors. There are many unknowns in the theoretical study of Boolean decompositions. In addition, more work needs to be done in testing Boolean decomposition on spatiotemporal datasets.

Since spatiotemporal data is often stored in binary format, an approximate Boolean decomposition could provide a robust method that separates signal from noise and gives interpretable results. For such data, a Boolean CP tensor decomposition may be more effective than the standard CP decomposition over the real numbers. Dependent on the data being represented, a binary rank-one sub-tensor could have a very strong meaning. An approximate Boolean CP decomposition finds approximate binary rank-one sub-tensors of the original tensor. Thus, an interpretation of the data can be made by examining the component tensors in the decomposition, rather than the reconstructed tensor.

Even if there is no specific interpretation of a rank-one binary tensor for a given type of data, the structure may make theoretical questions easier to study than their real counterparts. In particular, tools and known results from graph theory can be applied. This combinatorial structure presents more options for numerical algorithms for Boolean CP decomposition. Thus, Boolean CP decomposition is a method with promising applications for studying and interpreting spatio-temporal data to improve Army GEOINT-HUMINT capabilities.

ACKNOWLEDGMENTS: Ms. Mackenzie Simper thanks the NSF and ORISE for funding her research through the MSGI internship program.

REFERENCES

Bergqvist, G. 2013. Exact probabilities for typical ranks of $2 \times 2 \times 2$ and $3 \times 3 \times 2$ tensors. *Linear Algebra and its Applications* 438(2):663–667. <https://doi.org/10.1016/j.laa.2011.02.041>.

- Cergani, E., and P. Miettinen. 2013. Discovering relations using matrix factorization methods. In *Proceedings of the 22Nd ACM International Conference on Information & Knowledge Management (CIKM)* New York, NY: ACM. doi: 10.1145/2505515.2507841.
- Erdos, P., A. W. Goodman, and L. Posa. 1966. The representation of a graph by set intersections. *Canadian Journal of Mathematics* 18:106–112. <https://doi.org/10.4153/CJM-1966-014-3>.
- Friedland, S. 2012. On the generic and typical ranks of 3-tensors. *Linear Algebra and its Applications* 436(3):478–497. <https://doi.org/10.1016/j.laa.2011.05.008>.
- Kolda, T. G., and B. W. Bader. 2009. Tensor decompositions and applications. *SIAM Review*, 51(3):455–500, September 2009. <https://doi.org/10.1137/07070111X>.
- Kruskal, J. B. 1989. *Multiway Data Analysis*. The Netherlands: North-Holland Publishing Co., Amsterdam, The Netherlands.
- Miettinen, P. 2011. Boolean tensor factorizations. In *Proceedings - IEEE International 11th Conference on Data Mining (ICDM)* 447–456.
- Miettinen, P. 2013. Fully dynamic quasi-biclique edge covers via Boolean matrix factorizations. In *Proceedings of the Workshop on Dynamic Networks Management and Mining, DyNetMM 2013*.
- Miettinen, P., and D. Erdos. 2013. WalkNMerge: A scalable algorithm for Boolean tensor factorization. In *International Conference on Data Mining (ICDM) 2013* 1037–1042.

NOTE: The contents of this technical note are not to be used for advertising, publication, or promotional purposes. Citation of trade names does not constitute an official endorsement or approval of the use of such products.