CP-ORTHO: An Orthogonal Tensor Factorization Framework for Spatio-Temporal Data

Ardavan Afshar
Georgia Institute of Technology
aafshar8@gatech.edu

Joyce C. Ho
Emory University
joyce.c.ho@emory.edu

Bistra Dilkina
Georgia Institute of Technology
bdilkina@cc.gatech.edu

Ioakeim Perros
Georgia Institute of Technology
perros@gatech.edu

Elias B. Khalil
Georgia Institute of Technology
elias.khalil@cc.gatech.edu

Li Xiong
Emory University
lxiong@emory.edu

Vaidy Sunderam
Emory University
vss@emory.edu

ABSTRACT

Extracting patterns and deriving insights from spatio-temporal data finds many target applications in various domains, such as in urban planning and computational sustainability. Due to their inherent capability of simultaneously modeling the spatial and temporal aspects of multiple instances, tensors have been successfully used to analyze such spatio-temporal data. However, standard tensor factorization approaches often result in components that are highly overlapping, which hinders the practitioner's ability to interpret them without advanced domain knowledge. In this work, we tackle this challenge by proposing a tensor factorization framework, called CP-ORTHO, to discover distinct and easily-interpretable patterns from multi-modal, spatio-temporal data. We evaluate our approach on real data reflecting taxi drop-off activity. CP-ORTHO provides more distinct and interpretable patterns than prior art, as measured via relevant quantitative metrics, without compromising the solution's accuracy. We observe that CP-ORTHO is fast, in that it achieves this result in 5x less time than the most accurate competing approach.

KEYWORDS

Tensor Factorization; Unsupervised Learning

1 INTRODUCTION

Spatio-temporal data analysis refers to the extraction of insights out of data containing spatial and temporal properties. A fundamental task is to automatically identify the underlying temporal trends and location patterns for sub-groups of the data instances. In urban planning, for instance, the identification of underlying temporal trends and location patterns is essential for city planners and traffic managers in order to improve a city's road infrastructure.

A crucial characteristic of spatio-temporal datasets is that they are inherently *multi-modal*, due to the simultaneous presence of both location and time modes which describe each of the available instances. Thus, one of the most natural ways to model such datasets is via tensors (i.e., multi-way arrays) [8]. One of the most popular tensor analysis methods is the canonical polyadic (CP) decomposition (also known as PARAFAC or CANDECOMP) [4, 6]. CP decomposes a tensor into a sum of (rank-one) outer products which effectively represents the underlying data concepts. Its popularity owes to its *intuitive output structure* and *uniqueness* property that make the model reliable to interpret [8, 9]. However, the resulting factors of the CP model are usually highly overlapping, which hampers their interpretability without advanced domain knowledge. In order to obtain more concise and easily-interpretable results, we would prefer the solution's components to be *as distinct as possible*. While prior art [11] has attempted to tackle this challenge, it promotes non-overlapping results for a *specific, fixed* tensor mode; such an objective is still limited by the need for prior domain knowledge or even an arbitrary choice of a tensor mode.

To tackle the challenges introduced above, we propose **CP-ORTHO**, a non-negative tensor factorization framework to discover distinct and easily-interpretable patterns from multi-modal, non-negative data. We propose a fast projected gradient optimization scheme to fit our objective. Our experimental evaluation on a real, publicly-available dataset showcases that CP-ORTHO achieves the best of both worlds in terms of *solution distinctiveness* and *speed*, without sacrificing the model's accuracy. To promote reproducibility, our code is open-sourced and publicly available at https://github.com/aafshar/CP-ORTHO.

2 PRELIMINARIES AND NOTATION

The order indicates the number of tensor modes (N). R denotes the number of pursued components (tensor rank). Matricization is the process of converting a tensor into a matrix without changing its values. The mode-n matricization of $\mathcal{X} \in \mathbb{R}^{I_1 \times \ldots \times I_N}$ is indicated as $\mathcal{X}_{(n)} \in \mathbb{R}^{I_n \times I_1 \ldots I_{n-1} I_{n+1} \ldots I_N}$. The multiplication of tensor $\mathcal{X}$ with N vectors in N modes is $\mathcal{X} \bigtimes_{n=1}^{N} A^{(n)} = \mathcal{X} \times_1 A_r^{(1)} \times_2 A_r^{(2)} \ldots \times_N A_r^{(N)}$ where and $A_r^{(n)}$ is the r-*th* factor (column) of mode n and $\times_n$ indicates the multiplication in mode n. The tensor inner product of $\mathcal{X}$ and $\mathcal{Y}$ is defined as [7]: $\mathcal{X} \bullet \mathcal{Y} = \sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \ldots \sum_{i_n=1}^{I_N} \mathcal{X}_{i_1 i_2 \ldots i_n} \mathcal{Y}_{i_1 i_2 \ldots i_n}$

A rank-one tensor $\mathcal{X}$ is equal to the outer product of N vectors: $\mathcal{X} = A^{(1)} \circ A^{(2)} \circ \ldots \circ A^{(N)}$. The CANDECOMP-PARAFAC (CP)

SIGSPATIAL'17, November 7–10, 2017, Los Angeles Area, CA, USA

ACM ISBN 978-1-4503-5490-5/17/11.
https://doi.org/10.1145/3139958.3140047

Table 1: Symbols and notations used.

Symbol	Definition
$*$	Element-wise Multiplication
$\circ$	Outer product
$\odot$	Khatri Rao Product
$\langle A, B \rangle$	Vector Inner Product
$\bullet$	Tensor Inner Product
$\times$	Tensor-Vector Multiplication
$\mathcal{X}$	Tensor
$\mathcal{X}_{(n)}$	Mode-n Matricization of $\mathcal{X}$
$\mathcal{X}(i)$	i^{th} rank-one tensor of $\mathcal{X}$
$A^{(n)}$	Factor matrix corresponding to the n-th mode
A_r	r-th column of matrix A

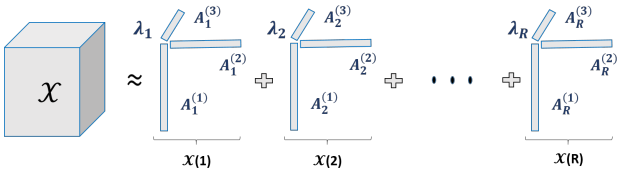


Figure 1: Decomposing tensor $\mathcal{X}$ into R rank-one tensors.

decomposes a tensor $\mathcal{X}$ as the sum of rank-one tensors, as follows: $\mathcal{X} \approx \sum_{r=1}^{R} \lambda_r A_r^{(1)} \circ A_r^{(2)} \circ \ldots \circ A_r^{(N)} = [\lambda; A^{(1)}; A^{(2)}; \ldots; A^{(N)}]$ where λ_r is the weight corresponding to the r-*th* rank-one tensor. Figure 1 illustrates the CP decomposition of a third-order tensor in more detail. Two rank-one tensors $\mathcal{X}(i)$ and $\mathcal{X}(j)$ are orthogonal [7] ($\mathcal{X}(i) \perp \mathcal{X}(j)$) iff

$$\mathcal{X}(i) \bullet \mathcal{X}(j) = \prod_{n=1}^{N} \langle A_i^{(n)}, A_j^{(n)} \rangle = 0, \tag{1}$$

where factors ($A_i^{(n)}$) are unit vectors. The definition suggests that two rank-one tensors are orthogonal if at least one pair of their decomposed factors has a zero inner product. When the input data and the decomposed factors are non-negative, the above property implies that there is no overlap between them, thus improving interpretability. Table 1 summarizes the notations used in this paper.

3 THE CP-ORTHO APPROACH

3.1 Problem Formulation

We express the spatio-temporal dataset as an observed non-negative tensor $\mathcal{X}$ with size $I_1 \times I_2 \times \cdots \times I_N$. CP-ORTHO decomposes the input tensor into R rank-one tensors such that each of them is orthogonal to all others. We thus reduce the overlap between rank-one tensors to capture more distinct and meaningful patterns.

In Section 2, we revised the definition of tensor orthogonality regarding two rank-one tensors. In real-life data mining applications, a tensor is usually approximated as a sum of R rank-one components. Thus, we below extend the definition of orthogonal tensors, assuming they can be decomposed into R components. Thus, to ensure every pair of rank-one tensors $i, j \in [1, R]$ is orthogonal to each other, we require that:

$$\sum_{i=1}^{R} \sum_{j=i+1}^{R} \mathcal{X}(i) \bullet \mathcal{X}(j) = \sum_{i=1}^{R} \sum_{j=i+1}^{R} \prod_{n=1}^{N} \langle A_i^{(n)}, A_j^{(n)} \rangle = 0 \tag{2}$$

We introduce a new matrix Q, which is the element-wise product of the individual factor similarity matrices ($A^{(n)^T} A^{(n)}$).

$$\left(\left(A^{(1)^T} A^{(1)}\right) * \left(A^{(2)^T} A^{(2)}\right) * \ldots * \left(A^{(N)^T} A^{(N)}\right) \right) = Q. \tag{3}$$

Q is a symmetric $R \times R$ matrix where each element is:

$$Q_{ij} = \mathcal{X}(i) \bullet \mathcal{X}(j) = \prod_{n=1}^{N} \langle A_i^{(n)}, A_j^{(n)} \rangle$$

It is important to note that Q should only be non-zero along the diagonal if all the rank-one tensors are orthogonal to one another, and will be exactly equal to the identity matrix I if their decomposed factors are also unit vectors.

Spatio-temporal data can usually be represented as event occurrences. Thus, our input tensor consists of non-negative values. In that case, interpretability is improved if the CP decomposition is constrained to contain non-negative factors as well (where both the weights and the factor matrices are non-negative). To summarize the above constraints, we provide the following constrained optimization problem:

$$\mathcal{F} = \min_{\lambda, A^{(1)}, \cdots, A^{(N)}} \frac{1}{2} \left\| \mathcal{X} - \left[\lambda; A^{(1)}; A^{(2)}; \ldots; A^{(N)}\right] \right\|_F^2 \tag{4}$$

$$\text{s.t.} \left(\left(A^{(1)^T} A^{(1)}\right) * \ldots * \left(A^{(N)^T} A^{(N)}\right) \right) = I \tag{5}$$

$$\left\| A_r^{(n)} \right\|^2 = 1 \quad , \forall n, r \tag{6}$$

$$A^{(n)} \in [0,1]^{I_n \times R}, \lambda \in [0, +\infty)^R$$

The first constraint (5) is the orthogonality constraint for the R rank-one tensors, and constraint (6) indicates that each column of the factor matrices must sum to one. The last two constraints are the nonnegative constraints on the weights and the factor matrices.

3.2 Algorithm

Since restricting the tensor decomposition to only orthogonal rank-one tensors can yield undesirable results (fitting noise or poor approximation of the observed tensor), we relax the orthogonality and unit-norm constraints [1]. This also allows us to use a first-order optimization algorithm to find the solution. We use the quadratic penalty method and convert the orthogonality and unit-norm constraints to penalty terms. Note that while our problem still has nonnegative constraints, we can utilize projected gradient descent to discover the weights and the factor matrices simultaneously. For notational convenience, we reformulate the objective function (4) and the orthogonality and unit vector penalty terms using the mode-n matricization of our tensor, $\mathcal{X}$. The new objective function is:

$$\mathcal{F} = \underbrace{\frac{1}{2} \left\| \mathcal{X}_{(n)} - A^{(n)} \Phi^{(n)^T} \right\|_F^2 + \frac{\psi_b}{2} \sum_{n=1}^{N} \sum_{r=1}^{R} \left(\left\| A_r^{(n)} \right\| - 1 \right)^2}_{\mathcal{F}_1}$$

$$+ \underbrace{\frac{\psi_a}{2} \left\| \left(\left(A^{(n)^T} A^{(n)}\right) * C_n \right) - I \right\|_F^2}_{\mathcal{F}_2}$$

where ψ_a and ψ_b are the regularization parameters, $\Phi^{(n)} = (\lambda^T \odot A^{(N)} \odot \cdots \odot A^{n+1} \odot A^{n-1} \odot \cdots \odot A^{(1)})$, and $C_n = (A^{(1)^T} A^{(1)}) * \cdots * (A^{(n-1)^T} A^{(n-1)}) * (A^{(n+1)^T} A^{(n+1)}) * \cdots * (A^{(N)^T} A^{(N)})$.

Projected Gradient Descent. The gradient can be computed by taking the partial derivative with respect to all the factors simultaneously, vectorizing the partials, and concatenating them together (i.e., $x = vec(A^{(1)}, ..., A^{(N)})$) [2]. After computing the gradient, we can use any first-order optimization method. Since our optimization problem has non-negativity constraints, we implement the projected gradient descent method [10] to solve the problem. To ensure a sufficient decrease in the function at each iteration, we use backtracking line search [12] to find a sufficient value for the step size by ensuring the step size meets the condition $\mathcal{F} - \mathcal{F}_{prev} < \alpha \bigtriangledown \mathcal{F}^T (x - x_{prev})$. We use two model parameters related to the two backtracking line search shrinkage, α_A and α_λ.

The full implementation details along with the detailed gradient computations will be provided in the extended version of this paper.

4 EXPERIMENTS

We compare the performance of CP-ORTHO with existing methods for constrained CP decomposition: CP-NMU [3], CP-APR [5], and Rubik [11]. A grid search for the best parameters was conducted. The final parameter values for NYC Taxi dataset for $R = 5$ are: $\psi_a = 15000$, $\psi_b = 15000$, $\alpha_A = 4e-5$, and $\alpha_\lambda = 4e-9$. For Rubik, we also performed a grid search with respect to its orthogonality parameter and used the value 100 which had the best result.

All experiments are conducted on a PC with a 3.6GHz i7 CPU and 32Gb RAM. Our code is open-sourced and publicly available at https://github.com/aafshar/CP-ORTHO.

4.1 Data Description

We evaluate CP-ORTHO on **NYC Taxi Data**[1] of taxi trips, as provided by the New York City (NYC) Taxi and Limousine Commission. We use all the data from January 2015 to June 2016 which has 215,519,509 trips (average of 393,283 trips per day). We investigate a rectangular area of 40 km × 31 km that covers Manhattan and some of the surrounding boroughs. The area is divided into grids of 250 meter × 250 meter. A third-order binary tensor is constructed from the drop-off data that represents 19,200 grid cell locations for 24 hours by 7 days.

4.2 Quantitative Metrics

Since all four models are only guaranteed to converge to local minima, we evaluate all competing approaches by running them for 10 random initializations. Table 2 reports the average and best fit over these 10 runs, as well as the mean running time. Regarding the accuracy of the solution, the results suggest that the mean and best fit of CP-ORTHO are comparable or better than the other models. At the same time, CP-ORTHO is 5 times faster than CP-APR, which is the most accurate competing method.

We propose two quantitative measures to assess a model's ability to produce interpretable results. The measures correspond to the ability to produce *distinct* patterns and "hard" cluster results. Distinct patterns make the assignment of meaning and interpretations to the factors easier. Thus, the *pattern distinctiveness* metric, captures to what extent the discovered patterns are distinct form each other via the pairwise orthogonality of the rank-one tensors of components, $\prod_{n=1}^{N} \langle A_i^{(n)}, A_j^{(n)} \rangle$, $\forall i, j \in [1, R], i < j$. For this metric, we desire lower values as it corresponds to pairs of components that are (close to) orthogonal. Table 3 summarizes the average pairwise pattern distinctiveness of the four models. CP-ORTHO has the lowest orthogonality between the pairs of corresponding components.

[1]Dataset available at http://www.nyc.gov/html/tlc/html/about/trip_record_data.shtml

Table 2: Comparison of fit and running time (10 seeds).

Algorithm	Fit Mean ± std.	Fit Best	Running Time (ms) Mean ± std.
CP-ORTHO	0.7330±0.0065	**0.7384**	1569±612
CP-APR [5]	**0.7371±0.0022**	0.7381	7940±861
CP-NMU [3]	0.7266±0.0054	0.7306	**93 ±20**
Rubik [11]	0.6852±0.0042	0.6894	2806±125

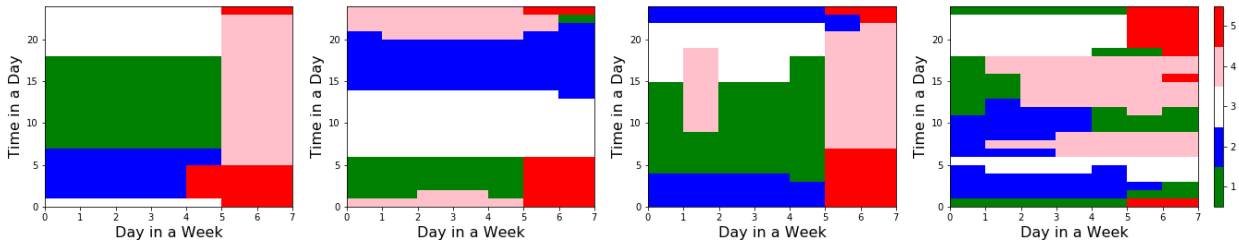
Table 3: Comparison of pattern distinctiveness (lower is better) and cluster membership (ΔC) values over all tensor elements (higher is better) for the best initialization.

Algorithm	Pattern Distinctiveness Mean ± std.	Cluster Membership(ΔC) Mean ± std.
CP-ORTHO	**0.1386±0.0646**	**0.5774±0.2995**
CP-APR	0.16763±0.1064	0.4978±0.2912
CP-NMU	0.49234±0.1314	0.2418 ±0.2056
Rubik	0.21640±0.1064	0.3800±0.2300

The second measure captures the ability of each method to produce a factorization that discriminatingly assigns samples to a corresponding pattern, i.e. the "hardness" of the clustering. Each factor element of the rank-one can be interpreted as the corresponding score of how much it belongs to each pattern. Thus, we analyze *cluster membership score*, the difference between the highest and second-highest pattern membership score, to capture how clearly the factorization assigns each element to a distinct pattern. The cluster membership score, ΔC, is calculated as: $\Delta C_{ijk} = \lambda_{r'} A^{(1)}_{r'_i} A^{(2)}_{r'_j} A^{(3)}_{r'_k} - \lambda_{r''} A^{(1)}_{r''_i} A^{(2)}_{r''_j} A^{(3)}_{r''_k}$, where r', r'' are the indices of the components which have the highest and second-highest membership scores, respectively. Larger values of ΔC are desirable as they indicate that the element in question strongly belongs to the component with the highest score. Table 3 summarizes the results of each model's hard clustering ability. CP-ORTHO results in significantly "harder" clustering of the samples into components/patterns than other approaches.

4.3 Case Study on NYC Taxi Data

Below, we study the patterns extracted from the methods under comparison. In Figure 2, we assign each combination of (day, hour) to a certain rank-one component (out of 5). To do so, we compute the outer product $A_i^{(2)} \circ A_i^{(3)}$ for each i-th component. Then, we assign each (day, hour) combination to the component with the maximum value in the corresponding entry of the outer product.

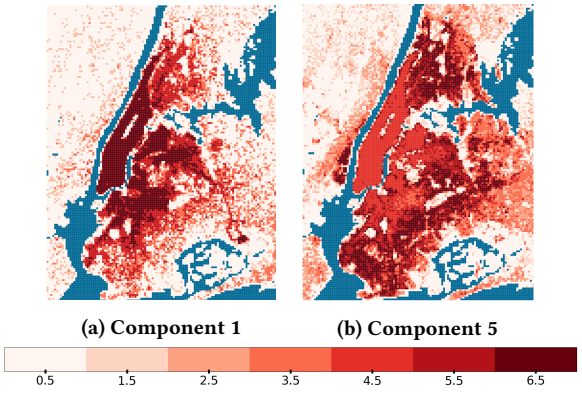


(a) CP-ORTHO (b) CP-APR (c) CP-NMU (d) RUBIK

Figure 2: Visualization of the contribution of each of the 5 output components to each (time, day) combination. We partially (considering the temporal modes only) re-construct the input for each i-th rank-one component separately, by computing the outer product $A_i^{(2)} \circ A_i^{(3)}$. Then, we assign then the color of each (day, time) combination based on the highest value of the 5 corresponding re-constructed ones. Monday is Day 0 on the x-axis.

From the CP-ORTHO diagram (Figure 2a), we can see minimal overlap between the weekend and weekdays. Components 1, 2 and 3 describe the weekdays, while components 4 and 5 describe the weekend. For instance, component 1 (green) captures the behavior from 7:00 AM to 5:00 PM on a workday. This is likely associated with the fact that most people go to work or commute between different work places. Another interesting pattern that is identified by our model is the separation of the early morning time period (12:00 AM to 6:00 AM) into two components. Component 2 captures the drop-off locations from Monday to Thursday while component 5 describes the drop-off patterns from Friday to Sunday. This seems to encapsulate the differences between people who work the early shifts during the weekdays and the party-goers who are returning home after a night out. The competing approaches do not to capture the distinction between weekends and week-days as concisely.

We also analyzed the drop-off locations (spatial mode) of our model. We focus on the components that have the highest spatial differences. Figure 3 shows the drop-off locations corresponding to components 1 and 5 for CP-ORTHO. One can see that the drop-off locations for component 1 (Figure 3a) are primarily in Manhattan, whereas the drop-off locations for component 5 (Figure 3b) are more spread out over all the boroughs including the Bronx, Brooklyn, and Queens. We will provide more extensive experimental results in the paper's extended version.



(a) Component 1 (b) Component 5

Figure 3: Values of drop-off locations for two different components from CP-ORTHO.

5 CONCLUSION

In this work, we propose CP-ORTHO, a tensor factorization framework enabling to find more meaningful and distinct patterns in spatio-temporal data. As measured via relevant quantitative metrics, CP-ORTHO provides more distinct and interpretable patterns than prior art, without compromising the solution's accuracy. It is also fast, in that it achieves this result in 5x less time than the most accurate competing approach. Future directions include: 1) extending the model to include count data using KL-divergence; 2) incorporating alternative optimization approaches to reduce the number of hyper-parameters; 3) proposing an explicit metric for hard clustering in tensor factorization.

6 ACKNOWLEDGMENT

This research was supported in part by AFOSR grant FA9550-17-1-0006 and NSF grant CNS-1618932. The work of Ioakeim Perros was partially supported by NSF, award number CCF-#1533768.

REFERENCES

[1] Evrim Acar, Daniel M Dunlavy, and Tamara G Kolda. 2011. A scalable optimization approach for fitting canonical tensor decompositions. *Journal of Chemometrics* 25, 2 (2011), 67–86.
[2] Evrim Acar, Tamara G Kolda, and Daniel M Dunlavy. 2011. All-at-once optimization for coupled matrix and tensor factorizations. *preprint, arXiv:1105.3422* (2011).
[3] Brett W Bader and Tamara G Kolda. 2007. Efficient MATLAB computations with sparse and factored tensors. *SIAM Journal on Scientific Computing* (2007).
[4] J Douglas Carroll and Jih-Jie Chang. 1970. Analysis of individual differences in multidimensional scaling via an N-way generalization of "Eckart-Young" decomposition. *Psychometrika* 35, 3 (1970), 283–319.
[5] Eric C Chi and Tamara G Kolda. 2012. On tensors, sparsity, and nonnegative factorizations. *SIAM J. Matrix Anal. Appl.* 33, 4 (2012), 1272–1299.
[6] Richard A Harshman. 1970. Foundations of the PARAFAC procedure: Models and conditions for an" explanatory" multi-modal factor analysis. (1970).
[7] Tamara G Kolda. 2001. Orthogonal tensor decompositions. *SIAM J. Matrix Anal. Appl.* 23, 1 (2001), 243–255.
[8] Tamara G Kolda and Brett W Bader. 2009. Tensor decompositions and applications. *SIAM review* 51, 3 (2009), 455–500.
[9] Joseph B Kruskal. 1977. Three-way arrays: rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics. *Linear algebra and its applications* 18, 2 (1977), 95–138.
[10] Chih-Jen Lin. 2007. Projected gradient methods for nonnegative matrix factorization. *Neural computation* 19, 10 (2007), 2756–2779.
[11] Yichen Wang, Robert Chen, Joydeep Ghosh, Joshua C Denny, Abel Kho, You Chen, Bradley A Malin, and Jimeng Sun. 2015. Rubik: Knowledge guided tensor factorization and completion for health data analytics. In *KDD*. ACM, 1265–1274.
[12] Stephen Wright and Jorge Nocedal. 1999. Numerical optimization. *Springer Science* 35 (1999), 67–68.