UNIVERSIDAD NACIONAL DE COLOMBIA
SEDE BOGOTÁ

FACULTAD DE INGENIERÍA
Vicedecanatura Académica
POSGRADOS

PROPOSAL SUBMISSION

DOCTORAL THESIS:	x	MASTER THESIS:	
MASTER FINAL WORK:		SPECIALIZATION FINAL WORK:	

1. **BIDDER:** Robinson Andrés Jaque Pirabán **ID:** 80190790
2. **PROGRAM:** Phylosophy Doctoral in Computer Science and Systems Engineering
3. **ADVISOR:** Fabio Augusto González Osorio
   DEPARTMENT: Computer Science and Industrial Engineering
4. **TITLE: A Kernel Tensor Factorization Algorithm for Multimodal data Analysis**
5. **AREA:** Computer Science
6. **LINE OF RESEARCH:** Machine Learning
7. COMMENTARY WITH ADVISOR APROVAL
8. BIDDER SIGNATURE
9. SIGNATURE OF ADVISOR

Contents

1 Introduction

Data-driven unsupervised learning has been establishing itself as a critical component of scientific discovery in domains ranging from astrophysics to social sciences and computational journalism. It manifests itself, among many forms, through modeling, machine learning, data mining, pattern recognition, data analytics, anomaly detection, and visualization. Nowaday, have rised a high volume of multiway data since the widespread use of multisensor technology and natural multiway structure of multimedia data, such as images, video or audio. This context have highlighted the limitations of standard flat-view matrix models and the necessity to move toward more versatile data analysis tools. In that sense, multiway analysis is of paramount importance. Data analysis techniques using tensor decompositions are shown to have great flexibility in the choice of constraints which match data properties and extract more general latent components in the data than matrix-based methods.

Tensors (or multidimensional arrays) are higher order arrays which generalize the notion of vectors (first-order tensor) or matrices (second-order tensor). Information about a phenomenon or a system of interest can be obtained from different types of instruments, measurement techniques, experimental setups, and other types of sources. Hence, tensors raise from many fields. Given the rich characteristics of natural processes and environments, it is rare that a single acquisition method provides complete understanding thereof. The increasing availability of multiple data sets that contain information obtained using different acquisition methods, about the same system, introduces new degrees of freedom that raise questions beyond those related to analyzing each data set separately. If natural high-order arrays are treated as a matrix, information would be lost since lack the original multiway structure of the data.

Multiway analysis enables one to effectively capture the multilinear structure of the data, which is usually available as a prior information about the data. Hence, the development of theory and algorithms for tensor decompositions (factorizations) has been an active area of study within the past decade [45, 24].

Recently there has been an increasing interest in the cross-fertilization of ideas coming from kernel methods and tensor-based data analysis. On the one hand it became apparent that machine learning algorithms can greatly benefit from the rich structure of tensor-based data representations. Within transductive techniques, in particular, tensor completion and tensor recovery emerged as a useful higher order generalization of their matrix counterpart

This paper is organized as follow: Section 1.1 introduce tensor definitions and some elemental operations over tensors. Later, section 1.3 describe some outstanding applications where data are represented as a high-order tensor. Furthermore, section 1.2 presents an overview of main tensor decomposition methods. Section 2 describe kernel methods and tensor factorization approaches incorporating them. Finally, section 2.2 describe approaches addressing kernel-based tensor analysis and a brief discussion of challenges of reviewed methods in section **??**.

1.1 Basics of tensors

Tensors are multidimensional arrays, i.e. an N-way or N-order tensor is an element of tensor product of N vector spaces, each of which has its own coordinate system. A first order tensor is a vector, a second order tensor is a matrix, tensors of higher order are called high-order tensors. The order (ways or modes) of a tensor is the number of dimensions. Figure 1 represents a 3-order tensor $\mathcal{X} \in \mathbb{R}^{I \times J \times K}$.

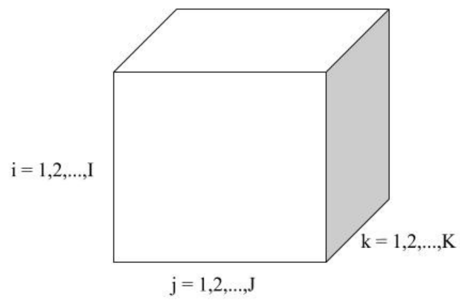
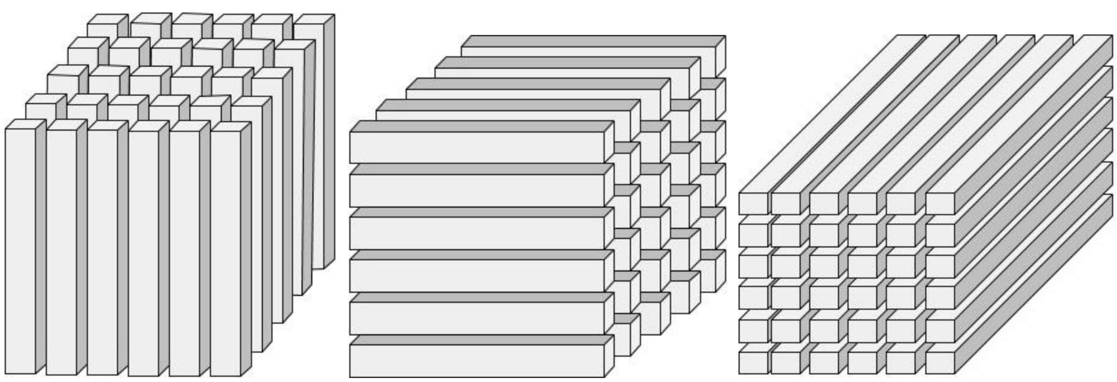


Figure 1: Third-order tensor

Fibers are defined by fixing every index by one. In a third-order tensor a column is a mode-1 fiber, denoted by $x_{:jk}$; a row is a mode-2 fiber, denoted by $x_{i:k}$; while a tube is a mode-3 fiber, denoted by $x_{i:k}$. 2 shows a fibers representation in 3rd-order tensor.



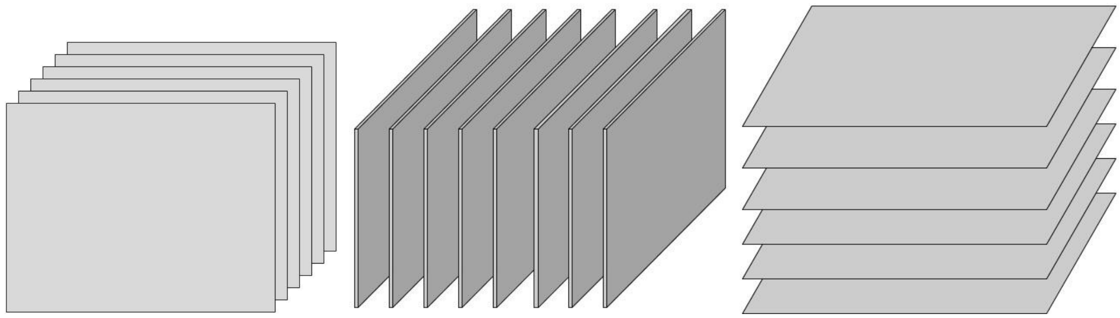
(a) columns (b) rows (c) tubes

Figure 2: 3rd-order tensor fibers

Slices are two-dimensional sections of a tensor defined by fixing two indexes. For instance, slices of 3rd-order tensor $\mathcal{X}$ are denoted by $X_{i::}$ (horizontal), $X_{:j:}$ (lateral) and $X_{::k}$ (frontal) and we ilustrate them in figure 3.

The *norm* of a tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \ldots \times I_N}$ is analogous to the matrix Frobenius norm, i.e.

$$||\mathcal{X}|| = \sqrt{\sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \cdots \sum_{i_N=1}^{I_N} x^2_{i_1 i_2 \ldots i_N}} \tag{1}$$



(a) Frontal slices (b) Horizontal slices (c) Vertical slices

Figure 3: 3rd-order tensor slices

$\mathcal{X}$ is a *Rank-one* tensor if it is equal to the outer product of N vectors, i.e.,

$$\mathcal{X} = a^{(1)} \otimes a^{(2)} \otimes \ldots \otimes a^{(N)}$$

1.1.1 Unfolding and Folding Tensors

Unfolding is the process of *matricization* of a tensor. In other words, elements of a tensors are sorted to assemble a matrix. The mode-k unfolding of a tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \ldots \times I_N}$ is denoted by $X_{(k)} \in \mathbb{R}^{I_1 \times \prod_{k' \neq k} I_{k'}}$ and arrenges the mode-k tensor fibers as columns of resulting matrix. In addition, Kolda [45] presents a more general procedures of unfolding

Ding and Wei [30] present a fast algorithm for Hankel tensor-vector products. And [34] a method of fast linear transform algorithm synthesis for an arbitrary tensor.

Kolda [45], Acar [1] and [26] present an exhaustive and detailed review of fundamental decomposition methods and applications. Furthermore, [27] presents tensor properties as extension of estructural properties of matrices. On the other hand, Fanaee and Gama [36] introduce an interdisciplinary survey about tensor-based anomaly detection.

1.2 Tensor Decomposition Methods

In this sections we explain some of basic methods to tensor decomposition which have been inspiration to many others methods propossed. Also, we summarize recently works and approaches of tensor decomposition on different application fields. Figure **??** attempts sum up methods reviewed.

1.2.1 Canonica Polyadic Decomposition / PARAFAC

Canonica Polyadic (CP) decomposition ([45], [49]), CANDECOMP [19] or PARAFAC [41] decompose a tensor as a finite sum of rank-one tensors. For instance, given a third order tensor $\mathcal{X} \in \mathbb{R}^{I \times J \times K}$, CP decomposition express it as

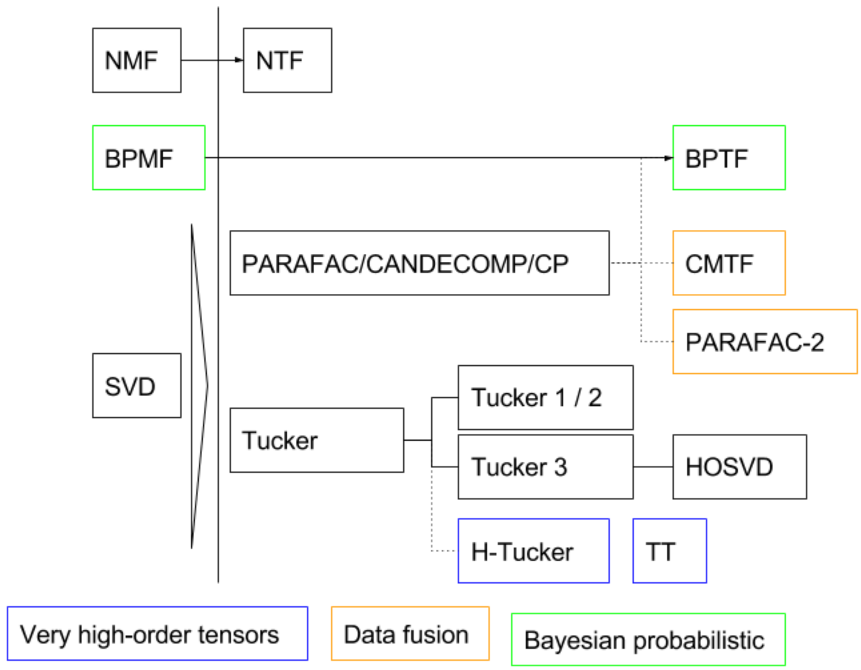


Figure 4: Tensor decomposition methods

$$\mathcal{X} \approx \sum_{r=1}^{R} a_r \otimes b_r \otimes c_r \tag{2}$$

where $a_r \in \mathbb{R}^I$, $b_r \in \mathbb{R}^J$, $c_r \in \mathbb{R}^K$ and R is a positive integer.

Domanov [31] shows relaxed uniqueness conditions and algebraic algorithm for Canonical polyadic decomposition, as well as a reduction to generalized eigenvalue decomposition [32] and uniqueness properties [33] of third-order tensors.

CP is one of the most popular tensor decompositions, in part due to its ease of interpretation [50]. Each rank-one component of the decomposition serves as a latent "concept" or cluster in the data. The factor vectors for component r can be interpreted as soft membership to the r-th latent cluster. As Harshman [1970] stated when he introduced CP,this is an explanatory model.

1.2.2 TUCKER Decomposition

Tucker decomposition was introduced by Tucker ([59], [60]). It is also named N-mode PCA [44], High-order SVD (HOSVD) [28] or N-mode SVD [61]. In the seminal work Tucker [58] proposes three models (Tucker-1, Tucker-2, Tucker-3) differing mainly from the decomposition method, Kolda and Bader presents a coprehensive review of these models [45]. The Tucker decomposition was further popularized by De Lathauwer et al. [28], wherein they coin the phrase **Higher-Order Singular Value**

Decomposition (HOSVD) for a particular method for computing the Tucker decomposition.

The Tucker decomposition is a form of higher-order PCA [45]. It descomposes a tensor into a core tensor $\mathcal{G}$ multiplied by a matrix algong each mode. For instance, given a third order tensor $\mathcal{X} \in \mathbb{R}^{I \times J \times K}$, Tucker decomposition express it as

$$\mathcal{X} \approx \mathcal{G} \times_1 A \times_2 B \times_3 C \tag{3}$$

Where $\times_k$ is the mode-k product, $A \in \mathbb{R}^{I \times P}$, $B \in \mathbb{R}^{J \times Q}$, $C \in \mathbb{R}^{K \times R}$ are the factor matrices (usually orthogonals) and can be interpreted as the principal components for each mode. $\mathcal{G} \in \mathbb{R}^{P \times Q \times R}$ is the core tensor and its entries show the interactions between the different components.

Hierarchical Tucker Decompostion (H-Tucker) In a Tucker model the number of variables we need to estimate increases exponentially to the number of modes, for instance, for a five-mode tensor $I \times I \times I \times I$, its (R, R, R, R, R) Tucker decomposition requires the computation of R^5 values for the core tensor $\mathcal{G}$.

This curse o dimensionality can be avoided with Hierarchical Tucker Decompostion [[39], [11]].

Suppose we have a binary tree of hierarchies of the modes of the tensor that can potentially be given to us by the application. Given this binary tree hierarchy, H-Tucker creates a set of generalized matricizations of the tensor according to each internal node of the tree. These matricizations are defined over a set of indice indicated by the particular node of the tree: for instance, given an $I \times J \times K \times L$ tensor, if node t splits the modes into two disjoint sets $\{I, J\}$ and $\{K, L\}$, then the generalized matricization $X_{(t)}$ will create an $IJ \times KL$ matrix where slices of the modes that are compacted into a single mode are stacked in a systematic fashion.

For each internal node of the tree, it computes a "transfer" core tensor, which requires the estimation of much fewer values than in the Tucker case. The core tensor Bt is computed via

$$U_t = (U_{tl} \otimes U_{tr}) B_t,$$

where B_t is an $r_{tl} rtr \times r_t$ matrix and U_t contains the r_t left singular vectors of the $\mathcal{X}_(t)$ matricization. Finally, the leaf nodes contain the factor matrices that are similar to the ones that Tucker would give. Figure 5 depict a example of a binary tree hierarchy and its corresponding H-Tucker decomposition.

1.2.3 Non-negative Tensor Factorization

In their seminal paper, Lee and Seung [52] demonstrate that enforcing nonnegativity constraints on the factors of a matrix factorization can lead to more interpretable and intuitive results. Conceptually, when we are dealing with data that can be expressed as a "sum of parts," then incorporating nonnegativity constraints successfully expresses this property. [50]

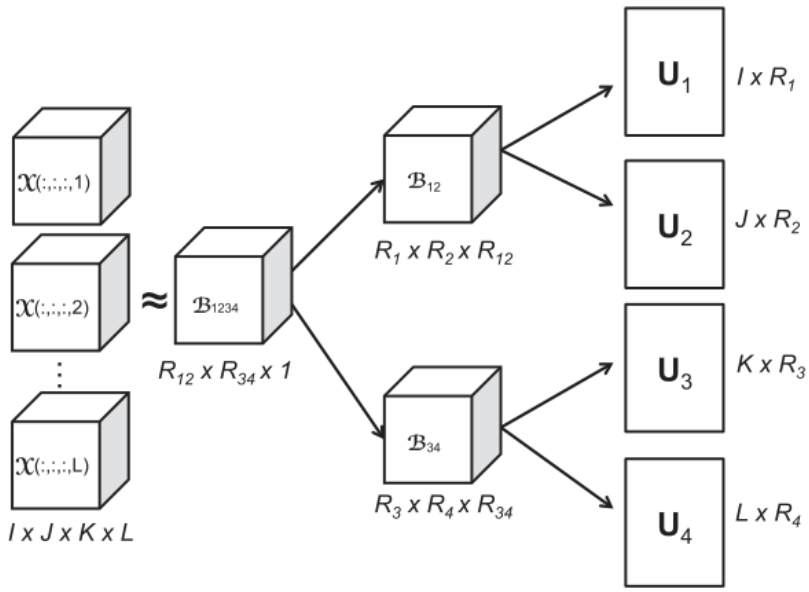


Figure 5: H-Tucker decomposition [50]

Non-negative Matrix Factorization The general problem of non-negative matrix factorization (NMF) is to decompose a matrix $X \in \mathbb{R}^{n \times l}_{\geq 0}$ into two matrix factors: basis $W \in \mathbb{R}^{n \times k}_{\geq 0}$ and coefficients $H \in \mathbb{R}^{k \times l}_{\geq 0}$, i,e.

$$X \cong WH \tag{4}$$

The factorization problem can be seem as an optimization problem:

$$\min_{W,H} d(X, WH) \tag{5}$$

where $d(,)$ is a distance or divergence function and the problem could have different types of restrictions. For instance, if $d(,)$ is the Euclidean Distance and there are not restrictions, the problem is solved by finding the SVD; if X, W and H are restricted to be positive, then the problem is solved by NMF. An comprehensive survey of NMF variants and algorithms is finded in [62]. One NMF approach is Symmetric-NMF [29], (SNMF) which produces a factorization:

$$(X^T_{lxn} X_{nxl}) = H_{lxk} H^T_{kxl} \tag{6}$$

An important characteristic of this version of NMF is that it is amenable to be used as a kernel method. This is discussed in the next subsection.

Non-negative Tensor Factorization An natural extension of nonnegative matrix factorization with high-order arrays is nonnegative n-dimensional tensor factorization (n-NTF). This kind of generalization is indeed not trivial since NTF possesses many new properties varying from NMF ([42], [**?**]).

First, the data to be processed in NMF are vectors in essence. However, in some applications the original data may not be vectors, and the vectorization might result in some undesirable problems. For instance, the vectorization of image data, which is two dimensional, will lose the local spatial and structural information. Second, one of the core concerns in NMF is the uniqueness issue, and to remedy the ill-posedness some strong constraints have to be imposed. Nevertheless, tensor factorization will be unique under only some weak conditions. Besides, the uniqueness of the solution will be enhanced as the tensor order increases[62].

There are generally two types of NTF model—NTD [?] and more restricted NPARAFAC [42], whose main difference lies in the core factor tensor. As for the solution, there are some feasible approaches. For example, NTF can be restated as regular NMF by matricizing the array [63], [?]. Or the alternating iteration method can be utilized directly on the outer product definition of tensors [42], [53], [13]. Similarly, SED, GKLD and other forms of divergence can also be used as the objective functions [13], [25], [64]. And some specific update models can adopt the existing conclusions in NMF.

1.2.4 Other methods

The tensor decomposition addressed in [15] may be seen as a generalization of Singular Value Decomposition of matrices. They consider general multilinear and multihomogeneous tensors. Then, they show how to reduce the problem to a truncated moment matrix problem and give a new criterion for flat extension of Quasi-Hankel matrices.

1.2.5 Summary

Table 1 summarize strengths and drawbacks of tensor decompostion methods reviewed in this section.

1.3 Common tensor analysis applications

1.3.1 Data Fusion

Data fusion is a common application in tensor analysis, in many application we have data represented as a tensor and have side information or metadata that may form matrices or other tensors.

In order to integrate that kind of data, one of the most popular models is the Coupled Matrix-Tensor Factorization (CMTF) [3], where one or more matrices are coupled with a tensor.

Closely related to the coupled datasets is the *multiset* data, i.e. a collection of K matrices $\{X_k\}$ that have one mode in common. Rather than a common tensor approach, the nonshared mode has different dimensions, and thus it has to be handled carefully. The PARAFAC2 model, was designed by Harshman [41] for such scenes. PARAFAC2 decomposes each matrix in the multiset as

$$X_k \approx U_k H S_k V^T$$

Method	Strengths	Weakness	Utility
CP	Easy to interpret, essentially unique under mild conditions, exploratory model	Hard to determine the appropiate rank and the global minimum	Extrating latent factors for interpretation, exploratory clustering
Tucker	Yields a good low-rank approximation of a tensor. Captures nontrilinear variations and compress a tensor optimally	Nonunique and hard to interpret, especially if core tensor is dense	Tensor compression, analyzing data where realations between latent component is expected, and estimate missing values. Note here that if data have low-rank multilinear structure, CP may be better than Tucker compressing that data; howerver, Tucker can copress well datasets that do not necesarily have low tensor rank.
H-Tucker	Approximates well hig-order tensors without suffering from the curse of dimensionality.	Requieres a priori knowledge of a binary tree of matricizations of the tensor.	For very high-order tensors, especially when the application at hand provides an intuitive and natural hierarchy over the modes.
CMTF	Jointly analyzes heterogeneous datasets that have one of their modes in common, and incorporates side information and metadata to the analysis.	When the heterogenous datasets are vastly different in terms of size and volume, we need to apply appropriate weighting in order to avoid one dataset drowning the rest.	In applications where metadata are present and can be modeled as side information matrices, as well as when having two or more heterogeneous pieces of data that refer to the same physical entities.
PARAFAC-2	Can jointly analyze heterogeneous pieces of data that cannot be expressed as a tensor.	Generalize the method to tensors is trivial since it would be equivalent to CP	When we have a set of matrces that nearly form a tensor, but they do not match one of the modes.

Table 1: Summary of tensor decomposition methods

The model acknowledges the differences in the rows by introducing a set of U_k row factor matrices, but imposes a joint latent structure on the columns and the thirdmode, similar to CP.

1.4 General problems

1.4.1 Blind Source Separation

Blind source separation (BSS) and related methods, e.g., independent component analysis (ICA), employ a wide class of unsupervised learning algorithms and have found important applications across several areas from engineering to neuroscience **??**. The recent trends in blind source separation and generalized (flexible) component analysis (GCA) are to consider problems in the framework of matrix factorization or more general multi-dimensional data or signal decomposition with probabilistic generative models and exploit a priori knowledge about true nature, morphology or structure of latent (hidden) variables or sources such as nonnegativity, sparseness, spatiotemporal decorrelation, statistical independence, smoothness or lowest possible complexity. The goal of BSS can be considered as estimation of true physical sources and parameters of a mixing system, while the objective of GCA is to find a reduced or hierarchical and structured component representation for the observed (sensor) data that can be interpreted as physically or physiologically meaningful coding or blind signal decomposition. The key issue is to find such a transformation or coding which has true physical meaning and interpretation.

1.4.2 Tensor completion

3.1.3. Handling M issing V a lues. When dealing with real data, there are many reasons we can expect some elements to be missing. Whether because of corruption, faulty measurements, or incomplete information (e.g., in recommender systems), there is a need to equip our algorithms with the ability to handle missing data.[50]

In tensor completion, an given incomplete tensor is given, i.e. some of its entries are missing and we should complete that. Following Ji Lu et.al [48] notation, low rank matrix completion is noted as

$$\begin{aligned} &\min_X \operatorname{rank}(X) \\ &\text{s.t. } X_\Omega = M_\Omega \end{aligned} \tag{7}$$

where Ω is an index set, then X_Ω is coping entries of X in the indexes Ω and missed entries $\hat{\Omega}$ would be 0

The missing entries in X are determined in order to minimize the matrix X rank. i.e. a non convex optimization problem since rank is nonconvex.

Frequently, trace norm (or nuclear norm) $||\cdot||_*$ is used to approximate the rank of matrices.

Trace norm is the tighest convex envelop for the matrices rank.

$$
\begin{aligned}
&\min_{X} ||X||_* \\
&\text{s.t. } X_\Omega = M_\Omega
\end{aligned} \tag{8}
$$

Since tensor is a generalization of the matrix concept, we generalize the optimization problem as

$$
\begin{aligned}
&\min_{\mathcal{X}} ||\mathcal{X}||_* \\
&\text{s.t. } \mathcal{X}_\Omega = \mathcal{T}_\Omega
\end{aligned} \tag{9}
$$

Where $\mathcal{X}$ and $\mathcal{T}$ are n-order tensors with identical size.

Acar et.al [3] presents a scalable tensor factorization method to deal with completion problem using PARAFAC method. Cao [18] propose a new tensor completion model via folded-concave penalty for estimating missing values in tensor data.To solve the resulting nonconvex optimization problem, we develop a local linear approximation augmented Lagrange multiplier (LLA-ALM) algorithm which combines a two-step LLA strategy to search a local optimum of the proposed model efficiently. They show numerical results in image and video data sets and compare with nuclear norm penalization method in order of demostrate its advantage in terms of the accuracy and robustness.

Chen [22] propose a method to deal with the completion problem when the number of missing entries increases, since factorization schemes may overfit the model because of incorrectly predefined ranks, while completion schemes may fail to interpret the model factors. Hence, they present an approach to complete the missing entries and simultaneously capture the underlying model structure that combines a rank minimization technique with Tucker model decomposition. Moreover, as the model structure is implicitly included in the Tucker model, they use factor priors, which are usually known a priori in real-world tensor objects, to characterize the underlying joint-manifold drawn from the model factors.

1.4.3 Tensor Decomposition origin and applications

Tensor decomposition originated with Hitchcock in 1927 [43], and the the multi-way model is attribuited to Cattell in 1944 [20].

Tensor works had attention in 60s with Tucker ([59], [60], [58]) and Carroll and Chang [19] and Harshman in 1970 [41] with applications in psychometrics. In 1981 Appellof and Davidson [9] used tensor decomposition in chemometrics which have been an popular field of application of tensor decomposition since then.

As shown figure 6, in last twenty years tensor decomposition applications have expanded to many fields such as signal processing, numberical linear algebra, computer vision, numerical analysis, neuroscience, data mining, graph analysis. Figure 6 shows seminal papers which opened broad application fields to tensor decomposition, plot also shows the amount of documents published about these works according to Scopus (A plain search).

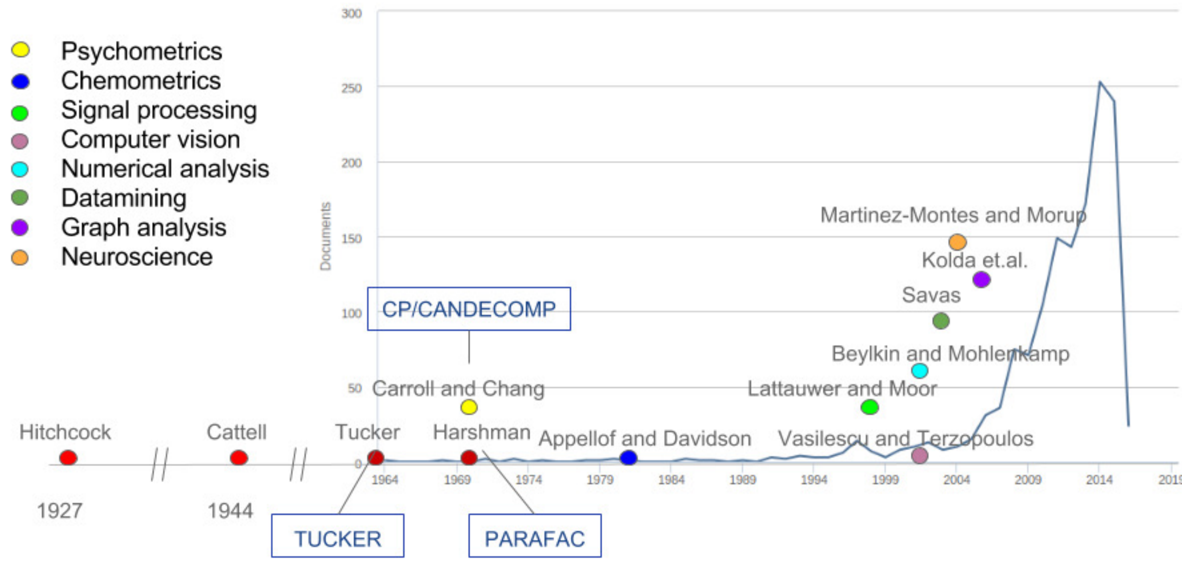


Figure 6: Time-line of tensor decomposition

Social network-analysis In online social networks, people tend to interact with each other in a variety of ways: they message each other, they post on each other's pages, and so on. All these different means of interaction are different aspects of the same social network of people, and can be modeled as a three-mode tensor, a "data cube," of (user, user, means of interaction [50]

Recommendation systems One of the first attempts to apply tensors to collaborative filtering and recommendation systems is Xiong et al. [2010]. The authors propose to extend Bayesian Probabilistic Matrix Factorization (BPMF) [Salakhutdinov and Mnih 2008], which is widely used in Recommendation Systems in the case where we have temporal information. They propose a Bayesian Probabilistic Tensor Factorization (BPTF) that is based on the CP model. In experimental evaluation, they show that BPTF outperforms BPMF, demon- strating that using temporal information and exploiting the higher-order structure it induces on the data prove beneficial for recommendation.

Chemometrics [21] Propose a computationally efficient technique for the solution of multi-dimensional PBMs of granulation via tensor decomposition

Bioinformatics In [2] and [4] a multimodal problem is addressed, the authors formulate data fusion as a coupled matrix and tensor factorization problem and discuss its extension to a structure-revealing data fusion model in metabolomics.

Image processing [5] NNTF for facial expression recognition

Machine Learning [6] Tensor decompositions for learning latent variable models

Text mining [7] This paper describes a method for automatic detection of semantic relations between concept nodes of a networked ontological knowledge base by analyzing matrices of semantic-syntactic valences of words. These matrices are obtained by means of nonnegative factorization of tensors of syntactic compatibility of words.

Numerical analysis [8] use of the sum-factorization for the calculation of the integrals arising in Galerkin isogeometric analysis. While introducing very little change in an isogeometric code based on element-by-element quadrature and assembling, the sum-factorization approach, taking advantage of the tensor-product structure of splines or NURBS shape functions, significantly reduces the quadrature computational cost.

[14] Fast iterative solution of the Bethe-Salpeter eigenvalue problem using low-rank and QTT tensor approximation.

[23] Blind identification of a second order Volterra-Hammerstein series using cumulant cubic tensor analysis.

Neuroscience [10] Decomposition of brain diffusion imaging data uncovers latent schizophrenias with distinct patterns of white matter anisotropy, using NNTF to clustering.

Signal processing Cichocki et.al. [24] sum up tensor decomposition approaches for signal processing problems.

Barker and Virtanen [12] deal with monaural sound source separation problem using NNTF of modulation spectrograms.

[17] Necessity to manually assign the NTF components to audio sources in order to be able to enforce prior information on the sources during the estimation process, Automatic Allocation of NTF Components for User-Guided Audio Source Separation

[38] propose a shifted 2D non-negative tensor factorisation algorithm which extends non-negative matrix factor 2D deconvolution to the multi-channel case. The use of this algorithm for multi-channel sound source separation of pitched instruments is demonstrated.

Other applications [35] Discovering and Characterizing Mobility Patterns in Urban Spaces: A Study of Manhattan Taxi Data. by using non-negative tensor factorization (NTF), we are able to cluster human behavior based on spatio-temporal dimensions. Second, for understanding these clusters, we propose to use HypTrails, a Bayesian approach for expressing and comparing hypotheses about human trails.

[37] NTF factorization for household electrical seasonal consumption disaggregation

2 Kernel Tensor Factorization

2.1 Kernel methods

In contrast with traditional learning techniques, kernel methods do not need a vectorial representation of data. Instead, they use a kernel function. Therefore, kernel methods are naturally applied to unstructured, or complex structured, data such as texts, strings, trees and images [54].

Informally, a kernel function measures the similarity of two objects. Formally, a kernel function, $k : X \times X \to \mathbb{R}$, maps pairs (x, y) of objects in a set X, the problem space, to the reals. A kernel function implicitly generates a map, $\Phi : X \to F$, where F corresponds to a Hilbert space called the feature space. The dot product in F is calculated by k, specifically $k(x, y) =< \Phi(x), \Phi(y) >_F$. Given an appropriate kernel function, complex patterns in the problem space may correspond to simpler patterns in the feature space. For instance, non-linear patterns in the problem space may correspond to linear patterns in the feature space.

Both k-means and SNMF have kernelized versions, which receive as input a kernel matrix instead of a set of sample represented by feature vectors. The kernel version of k-means is called, unsurprisingly, kernel k-means (KKM). In the case of SNMF, the kernelized version works as follows.

SNMF starts with an initial estimation of the matrix factor H and iteratively update it using the updating equation:

$$H_{i,k} = H_{i,k}(1 - \beta + \beta \frac{((X^T X)H)_{i,k}}{(HH^T H)_{i,k}})$$

The kernel version of the algorithm is obtained by using a kernel matrix K instead of the expression $(X^T X)$, where K is an $l \times l$ matrix with $K_{i,j} = k(x_i, x_j)$. There are different types of kernels some of them general and some of them specifically defined for different types of data. The most popular general kernels are the linear kernel

$$k(x, y) =< x, y >, \tag{10}$$

the polynomial kernel

$$k(x, y) = p(< x, y >),$$

where $p(\,)$ is a polynomial with positive coefficients, and the Gaussian (or RBF) kernel

$$k(x, y) = e^{\frac{\|x-y\|^2}{2\sigma^2}}. \tag{11}$$

The cluster centroids estimated by the kernel versions of both algorithms are in the feature space and correspond to the points $C_j = \frac{1}{n} \sum_{x_i \in C_j} \Phi(x_i)$. However, we are interested on the pre-image in the original space of this centroids, i.e., points $\hat{C}_j$ such that $\Phi(\hat{C}_j) = C_j$. However, it is possible that a exact pre-image may not even exist, so we look for the $\hat{C}_j$ that minimizes the following objective function:$\min_{\hat{C}_j} \left\| \hat{C}_j - C_j \right\|^2$.

According to Kwok et al. [?], the optimum C_j can be found by iterating the following fixed-point formula:

$$\hat{C}_j^{t+1} = \frac{\sum_{i=1}^{N} \exp(\frac{-||\hat{C}_j^t - x_i||)}{s}) x_i}{\sum_{i=1}^{N} \exp(\frac{-||\hat{C}_j^t - x_i||}{s})} \tag{12}$$

2.2 Kernel Non-negative Matrix Factorization

Kernel Non-negative Matrix Factorization (KNMF) can be naturally derivated of convex NMF ([46], [47] and [51]). Given a kernel function $\phi : x \in X \to \phi(x) \in F$, mapping for N elements $\phi(X) = [\phi(x_1), \ldots \phi(x_N)]$. Then, KNMF can be defined as

$$\phi(X) \cong \phi(X) W H^T \tag{13}$$

Therefore, the cost function to minimize is

$$||\phi(X) - \phi(X) W H^T||_F^2 = tr(K) - 2tr(H^T K W) + tr(W^T K W H^T H) \tag{14}$$

Where kernel $K = \phi^T(X) phi(X)$

2.3 kernel-based tensor analysis

Tensor decompositions consist of several linear transformations collaboratively performed in different modes, using multilinear algebra, which enables us to capture the underlying interactions among multiple modes. As a result, their extension to capture nonlinear multimode interactions of data is highly desirable [65].

Recent research has addressed the incorporation of the kernel concept into tensor decompositions, table 2 collect these works. [40] proposes an tensor kernel approach for SVM. [56] deal with linear tensor-based models and a systematic approach to the construction of non-parametric tensor-based models is still missing.

Tensor representation is able to construct a multifactor structure of image ensembles while tensor decompositions allow us to extract multiple low-dimensional components in different modes, which can be employed for image recognition and image synthesis [65]

[57] presents a method for classification of multichannel signals with cumulant-based kernels applied to magnetoencephalography (MEG) images.

[65] proposes two methods: The Kernel Tensor Partial Least Squares (KTPLS) which extends High-Order Partial Least Squares (HOPLS) combining it with Kernel Partial Least Squares (KPLS). HOPLS predicts a tensor $\mathcal{Y}$ from a tensor $\mathcal{X}$ through multilinear projection of the data onto the latent space followed by regression against the latent variables. And The Kernel Tensor Canonical Correlation Analysis (KTCCA), KTCCA extends Kernel Canonical Correlation Analysis (KCCA) to high-order tensors. CCA finds the directions of maximum correlation while PLS finds the directions of maximum covariance.

Table 2: Work linking kernels in tensor decomposition

Method	Year
Multi-view kernel completion	2016 [16]
A general framework to learn functions in tensor product reproducing kernel Hilbert spaces (TP-RKHSs)	2013 [55]
Kernel Tensor Partial Least Squares (KTPLS) and Kernel Tensor Canonical Correlation Analysis (KTCCA)	2013 [65]
Classification of Multichannel Signals With Cumulant-Based Kernels	2012 [57]
A kernel-based framework to tensorial data analysis	2011 [56]
Tensor kernels for SVM	2010 [40]

[55] proposes a general framework to learn functions in tensor product reproducing kernel Hilbert spaces (TP-RKHSs). The methodology is based on a novel representer theorem suitable for existing as well as new spectral penalties for tensors.

Finally, [16] presents a method for multi-view kernel completion. The method compute kernels over tensor slices and through kernel matrix compute costs which accumulative sum are used to recover missing tensor parts.

Tensor factorization (including matrix factorization) is central to different important tasks in machine learning and information retrieval such as: clustering, latent topic analysis, recommendation, blind source separation, completion, denoising, among others.

The widespread use of multisensor technology and the emergence of data sets where the collected data is most naturally stored or represented in a multi-dimensional array have highlighted the limitations of standard flat-view matrix models and the necessity to move toward more versatile data analysis tools [22].

Tensor factorizations have several advantages over two-way matrix factorizations including uniqueness of the optimal solution and component identification even when most of the data is missing.[?]

Conventional methods preprocess multiway data by arranging them into a matrix, which might lose the original multiway structure of the data [55]. Hence, tensors address multimodal or multiview data, and tensor factorization brings tools to perform common machine learning tasks.

On the other hand, kernel methods are ubiquitous in machine learning, performance of these methods have been broadly demonstrated, and there are evidence between some types of kernels and robustness. However, there are few exploration of kernel tensor factorization approaches.

3 Problem Statement

Tensor factorization (including matrix factorization) is central to different important tasks in machine learning and information retrieval such as: clustering, latent topic analysis, recommendation, blind source separation, completion, denoising, among others.

The widespread use of multisensor technology and the emergence of big data sets have highlighted the limitations of standard flat-view matrix models and the necessity to move toward more versatile data analysis tools [24]. Conventional methods preprocess multiway data by arranging them into a matrix, which might lose the original multiway structure of the data [62]. Hence, tensors address multimodal or multiview data, and tensor factorization brings tools to perform common machine learning tasks.

Tensor factorizations have several advantages over two-way matrix factorizations including uniqueness of the optimal solution and component identification even when most of the data is missing [**?**].

On the other hand, kernel methods are ubiquitous in machine learning, performance of these methods have been broadly demostrated, and there are evidence between some types of kernels and robustness. However, there are few exploration of kernel tensor factorization approaches.

A satisfactory solution of this general challenge requires to answer some particular**research question**:

- How incorporate kernel methods in tensor factorization in order to deal with multiway data?

An answer to the question derivate in a method which factorize a given tensor in the feature space induced by a Kernel function. Naturally, to inquire about the effects of decompose tensors in an space induced by a kernel function open space to evaluate the proposal method performance as well as its capabilities dealing with multimodal data, scalability and robustness to noise and outliers.

4 Goals

General objective

To design, implement and evaluate a kernel-based tensor factorization algorithm for multimodal data.

Specific objectives

- To evaluate different strategies for combining kernel-based methods with matrix and tensor factorization techniques.

- To design an algorithm for kernel-based tensor factorization.

- To develop an efficient implementation of the algorithm able to deal with large scale data sets.
- To assess the effectivity of the method in specific unsupervised learning tasks.

5 Jusification

Due to the rich characteristics of natural processes and environments, it is rare that a single acquisition method provides complete understanding thereof. Information about a phenomenon or a system of interest can be obtained from different types of instruments, measurement techniques, experimental setups, and other types of sources. The increasing availability of multiple data sets that contain information, obtained using different acquisition methods, about the same system, introduces new degrees of freedom that raise questions beyond those related to analyzing each data set separately.

By treating that natural high-order array as a matrix, information is lost since lack the original multiway structure of the data.

Tensor factorizations have several advantages over two-way matrix factorizations including uniqueness of the optimal solution and component identification even when most of the data is missing [?].

multiway decomposition techniques explicitly exploit the multiway structure that is lost when collapsing some of the modes of the tensor in order to analyze the data by regular matrix factorization approaches.

Tensor decompositions are in frequent use today in a variety of fields ranging from psychology, chemometrics, signal processing, bioinformatics, neuroscience, web mining, and computer vision to mention but a few.

Matrix and analogous tensor factorization is central to different important tasks in machine learning and information retrieval such as: clustering, latent topic analysis, recommendation, tensor completion, blind source separation, among others.

6 Methodology

This research is an experimental study with experimental design and quantitative strategies such as accuracy, specificity and sensitivity in a cross-sectional study with standard images.

Each specific objective is to be attained through the following approaches:

(a) *Review kernel-based methods proposed to matrix and tensor factorization.* A systematic literature review will be leaded to identify the matrix and tensor factorization methods witch incorporate kernels. Methods to address the pre-image problem in kernel-methods also will be included in this review.

(b) *Design and implementation of Kernel Tensor Factorization algorithm.* An kernel matrix factorization methods will be adapted to use tensorial kernels. Efficient implementations will be developed exploiting particularities of the used kernels.

(c) *Strategy to assess method precision in specific unsupervised-learning tasks.* In order to evaluete algorithm effectivity solving completion and blind source separation problems, the method will be applied in both synthetic and real world datasets to assess performance.

(d) *Measuring algorithm scalability.* An theoretical time and space complexity bound will be assesed to the algorithm. Furthermore, an parallelizable version of algorithm will be suggested.

7 Activities

Project activities and results are structured within *Work Packages* (WP) in terms of *Tasks* (T) and *Deliverables*, as described below.

WP0: Cross activities - Divulgation

TASKS

- T0: Proposal writing.
- T1: Proposal presentation.
- T2: Write articles for journals or conferences.
- T3: Dissertation writing.
- T4: Dissartation defence.

DELIVERABLES

- Proposal
- Articles
- Dissertation

WP1: Literature Review

An systematic literature review will be done to review tensor factorization methods, in particular, kernel-based matrix and tensor factorization methods. Furthermore, we will include methods to address the pre-image problem in kernel methods.

TASKS

- T0: Stablish research question and criteria for document inclusion.
- T1: Search process
- T2: Quality assessment and data collection
- T3: Data analysis

DELIVERABLES

- Document summarizing literature review

WP2: Evaluation of strategies for combining kernel-based methods with matrix and tensor factorization techniques

According to literature review, we will analyze and contrast strategies for combining kernel-based methods with matrix and tensor factorization techniques

TASKS

- T0: Review and compare methods

DELIVERABLES

- Technical report

WP3: Design of algorithm for kernel-based tensor factorization.

TASKS

- T0: Model tensor factorization algorithm.
- T1: Address pre-image problem if it is pertinent.
- T2: Stablish how to apply the algorithm to tensor completion problem and blind source separation.

DELIVERABLES

- Algorithm pseudocode

WP4: Developing an efficient algorithm implementation able to deal with large scale data sets.

TASKS

- T0: Adapt algorithm to scale if it is required.
- T1: Implement kernel tensor factorization algorithm.
- T2: Design experimental setup.
- T3: Execute experiments and measure performance.
- T4: Collect and analyze results

DELIVERABLES

- source code
- Technical report

WP5: Assessing the effectivity of the method in specific unsupervised learning tasks.

TASKS

- T0: Design experimental setup.
- T1: Execute experiments and measure performance.
- T2: Collect and analyze results

DELIVERABLES

- source code
- Technical report

8 Schedule

References

[1] E. Acar and B. Yener. Unsupervised Multiway Data Analysis: A Literature Survey. *IEEE Transactions on Knowledge and Data Engineering*, 21(1):6–20, jan 2009.

[2] Evrim Acar, Rasmus Bro, and Age K. Smilde. Data Fusion in Metabolomics Using Coupled Matrix and Tensor Factorizations. *Proceedings of the IEEE*, 103(9):1602–1620, sep 2015.

	Semester/Month																															
Actividad	1				2				3				4				5				6				7				8			
WP0.T0.	■	■	■	■	■	■	■	■																								
WP0.T1.									■	■																						
WP0.T2.								■	■																							
WP0.T3.										■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
WP0.T4.																													■	■	■	■
WP1.T0.	■																															
WP1.T1.	■	■	■	■	■	■	■																									
WP1.T2.	■	■	■	■	■	■	■																									
WP1.T3.				■	■	■	■	■																								
WP2.T0.					■	■	■	■	■																							
WP3.T0.						■	■	■	■	■	■						■	■	■	■												
WP3.T1.						■	■	■	■	■	■																					
WP3.T2.						■	■	■	■	■	■																					
WP4.T0.															■	■	■	■	■						■	■	■					
WP4.T1.															■	■	■	■	■													
WP4.T2.															■	■	■	■	■													
WP4.T3.															■	■	■	■	■													
WP4.T4.														■	■	■	■	■	■													
WP5.T0.								■	■	■	■	■	■	■	■	■	■	■														
WP5.T1.								■	■	■	■	■	■	■	■	■	■	■														
WP5.T2.								■	■	■	■	■	■	■	■	■	■	■		■	■	■	■	■								

Table 3: Gantt chart: project planning

[3] Evrim Acar, Daniel M. Dunlavy, Tamara G. Kolda, and Morten Mørup. Scalable tensor factorizations for incomplete data. *Chemometrics and Intelligent Laboratory Systems*, 106(1):41–56, mar 2011.

[4] Evrim Acar, Evangelos E Papalexakis, Gözde Gürdeniz, Morten A Rasmussen, Anders J Lawaetz, Mathias Nilsson, and Rasmus Bro. Structure-revealing data fusion. *BMC bioinformatics*, 15(1):239, jan 2014.

[5] Gaoyun An, Shuai Liu, and Qiuqi Ruan. A sparse neighborhood preserving non-negative tensor factorization algorithm for facial expression recognition. *Pattern Analysis and Applications*, aug 2015.

[6] Anima Anandkumar, Rong Ge, Daniel Hsu, Sham M. Kakade, and Matus Telgarsky. Tensor decompositions for learning latent variable models. pages 2773–2832, oct 2012.

[7] A. V. Anisimov, O. O. Marchenko, and T. G. Vozniuk. Determining Semantic Valences of Ontology Concepts by Means of Nonnegative Factorization of Tensors of Large Text Corpora. *Cybernetics and Systems Analysis*, 50(3):327–337, jun 2014.

[8] P. Antolin, A. Buffa, F. Calabrò, M. Martinelli, and G. Sangalli. Efficient matrix computation for tensor-product isogeometric analysis: The use of sum factorization. *Computer Methods in Applied Mechanics and Engineering*, 285:817–828, mar 2015.

[9] C. J. Appellof and E. R. Davidson. Strategies for analyzing data from video fluorometric monitoring of liquid chromatographic effluents. *Analytical Chemistry*, 53(13):2053–2056, nov 1981.

[10] Javier Arnedo, Daniel Mamah, David A Baranger, Michael P Harms, Deanna M Barch, Dragan M Svrakic, Gabriel A de Erausquin, C Robert Cloninger, and Igor Zwir. Decomposition of brain diffusion imaging data uncovers latent schizophrenias with distinct patterns of white matter anisotropy. *NeuroImage*, 120:43–54, oct 2015.

[11] Jonas Ballani, Lars Grasedyck, and Melanie Kluge. Black box approximation of tensors in hierarchical Tucker format. *Linear Algebra and its Applications*, 438(2):639–657, 2013.

[12] Tom Barker and Tuomas Virtanen. Semi-supervised non-negative tensor factorisation of modulation spectrograms for monaural speech separation. In *2014 International Joint Conference on Neural Networks (IJCNN)*, pages 3556–3561. IEEE, jul 2014.

[13] E Benetos and C Kotropoulos. Non-Negative Tensor Factorization Applied to Music Genre Classification. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(8):1955–1967, nov 2010.

[14] Peter Benner, Venera Khoromskaia, and Boris N. Khoromskij. A reduced basis approach for calculation of the Bethe–Salpeter excitation energies by using low-rank tensor factorisations. *Molecular Physics*, pages 1–14, feb 2016.

[15] A. Bernardi, J. Brachat, P. Comon, and B. Mourrain. General tensor decomposition, moment matrices and applications. *Journal of Symbolic Computation*, 52:51–71, may 2013.

[16] Sahely Bhadra, Samuel Kaski, and Juho Rousu. Multi-view Kernel Completion. feb 2016.

[17] Cagdas Bilen, Alexey Ozerov, and Patrick Pérez. Automatic Allocation of NTF Components for User-Guided Audio Source Separation, mar 2016.

[18] Wenfei Cao, Yao Wang, Can Yang, Xiangyu Chang, Zhi Han, and Zongben Xu. Folded-concave penalization approaches to tensor completion. *Neurocomputing*, 152:261–273, mar 2015.

[19] J. Douglas Carroll and Jih-Jie Chang. Analysis of individual differences in multidimensional scaling via an n-way generalization of "Eckart-Young" decomposition. *Psychometrika*, 35(3):283–319, sep 1970.

[20] Raymond B. Cattell. "Parallel proportional profiles" and other principles for determining the choice of factors by rotation. *Psychometrika*, 9(4):267–283, dec 1944.

[21] Anwesha Chaudhury, Ivan Oseledets, and Rohit Ramachandran. A computationally efficient technique for the solution of multi-dimensional PBMs of granulation via tensor decomposition. *Computers & Chemical Engineering*, 61:234–244, feb 2014.

[22] Yi-Lei Chen, Chiou-Ting Hsu, and Hong-Yuan Mark Liao. Simultaneous tensor decomposition and completion using factor priors. *IEEE transactions on pattern analysis and machine intelligence*, 36(3):577–91, mar 2014.

[23] I. Cherif and F. Fnaiech. Blind identification of a second order Volterra-Hammerstein series using cumulant cubic tensor analysis. In *2008 34th Annual Conference of IEEE Industrial Electronics*, pages 1851–1856. IEEE, nov 2008.

[24] Andrzej Cichocki, Danilo Mandic, Lieven De Lathauwer, Guoxu Zhou, Qibin Zhao, Cesar Caiafa, and HUY ANH PHAN. Tensor Decompositions for Signal Processing Applications: From two-way to multiway component analysis. *IEEE Signal Processing Magazine*, 32(2):145–163, mar 2015.

[25] Andrzej Cichocki, Rafal Zdunek, Seungjin Choi, Robert Plemmons, and Shun-ichi Amari. Non-Negative Tensor Factorization using Alpha and Beta Divergences. In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07*, volume 3, pages III–1393–III–1396. IEEE, 2007.

[26] Pierre Comon. Tensors : A brief introduction. *IEEE Signal Processing Magazine*, 31(3):44–53, may 2014.

[27] Stephane Dartois. Tensor Models: extending the matrix models structures and methods. page 13, mar 2016.

[28] Lieven De Lathauwer, Bart De Moor, and Joos Vandewalle. A Multilinear Singular Value Decomposition. *SIAM Journal on Matrix Analysis and Applications*, 21(4):1253–1278, jan 2000.

[29] CHQ Ding, X He, and HD Simon. On the Equivalence of Nonnegative Matrix Factorization and Spectral Clustering. In *2005 SIAM International Conference on Data Mining - SDM*, 2005.

[30] Weiyang Ding, Liqun Qi, and Yimin Wei. Fast Hankel tensor-vector product and its application to exponential data fitting. *Numerical Linear Algebra with Applications*, 22(5):814–832, oct 2015.

[31] I Domanov and L De Lathauwer. Canonical polyadic decomposition of third-order tensors: relaxed uniqueness conditions and algebraic algorithm. *arXiv preprint arXiv:1501.07251*, 2015.

[32] I Domanov and LD Lathauwer. Canonical polyadic decomposition of third-order tensors: reduction to generalized eigenvalue decomposition. *SIAM Journal on Matrix Analysis and Applications*, 2014.

[33] Ignat Domanov and Lieven De Lathauwer. On the Uniqueness of the Canonical Polyadic Decomposition of Third-Order Tensors—Part II: Uniqueness of the Overall Decomposition. *SIAM Journal on Matrix Analysis and Applications*, 34(3):876–903, jul 2013.

[34] Pavel Dourbal. Synthesis of fast multiplication algorithms for arbitrary tensors. page 79, feb 2016.

[35] Lisette Espín-Noboa, Florian Lemmerich, Philipp Singer, and Markus Strohmaier. Discovering and Characterizing Mobility Patterns in Urban Spaces: A Study of Manhattan Taxi Data. *arXiv preprint*, jan 2016.

[36] Hadi Fanaee-T and Joao Gama. Tensor-based anomaly detection: An interdisciplinary survey. *Knowledge-Based Systems*, feb 2016.

[37] M Figueiredo. Exploring the performance of non-negative multi-way factorization for household electrical seasonal consumption disaggregation. *Neural Networks (IJCNN) ...*, 2014.

[38] D. Fitzgerald, M. Cranitch, and E. Coyle. Shifted 2D Non-negative Tensor Factorisation, 2006.

[39] W. Hackbusch and S. Kühn. A New Scheme for the Tensor Representation. *Journal of Fourier Analysis and Applications*, 15(5):706–722, oct 2009.

[40] David R. Hardoon and John Shawe-Taylor. Decomposing the tensor kernel support vector machine for neuroscience data with structured labels. *Machine Learning*, 79(1-2):29–46, may 2010.

[41] RA Harshman. Foundations of the PARAFAC procedure: Models and conditions for an" explanatory" multi-modal factor analysis. *UCLA Working Papers in Phonetics*, 1970.

[42] T. Hazan, S. Polak, and A. Shashua. Sparse image coding using a 3D non-negative tensor factorization. In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, volume 1, pages 50–57 Vol. 1. IEEE, 2005.

[43] Frank L. Hitchcock. The Expression of a Tensor or a Polyadic as a Sum of Products. *Journal of Mathematics and Physics*, 6(1-4):164–189, apr 1927.

[44] Arie Kapteyn, Heinz Neudecker, and Tom Wansbeek. An approach ton-mode components analysis. *Psychometrika*, 51(2):269–275, jun 1986.

[45] Tamara G. Kolda and Brett W. Bader. Tensor Decompositions and Applications. *SIAM Review*, 51(3):455–500, aug 2009.

[46] Brian Kulis, Mátyás Sustik, and Inderjit Dhillon. Learning low-rank kernel matrices. In *Proceedings of the 23rd international conference on Machine learning - ICML '06*, pages 505–512, New York, New York, USA, jun 2006. ACM Press.

[47] Yang Li, Yangzhou Du, and Xueyin Lin. Kernel-based multifactor analysis for image synthesis and recognition. In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, volume 1, pages 114–119 Vol. 1. IEEE, 2005.

[48] Ji Liu, Przemyslaw Musialski, Peter Wonka, and Jieping Ye. Tensor completion for estimating missing values in visual data. *IEEE transactions on pattern analysis and machine intelligence*, 35(1):208–20, jan 2013.

[49] J Möcks. Topographic components model for event-related potentials and some biophysical considerations. *IEEE transactions on bio-medical engineering*, 35(6):482–4, jun 1988.

[50] Evangelos E. Papalexakis, Christos Faloutsos, and Nicholas D. Sidiropoulos. Tensors for Data Mining and Data Fusion. *ACM Transactions on Intelligent Systems and Technology*, 8(2):1–44, oct 2016.

[51] Roman Rosipal, Mark Girolami, Leonard J. Trejo, and Andrzej Cichocki. Kernel PCA for Feature Extraction and De-Noising in Nonlinear Regression. *Neural Computing & Applications*, 10(3):231–243, dec 2001.

[52] H. Sebastian Seung and Daniel D. Lee. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, oct 1999.

[53] Amnon Shashua and Tamir Hazan. Non-negative tensor factorization with applications to statistics and computer vision. In *Proceedings of the 22nd international conference on Machine learning - ICML '05*, pages 792–799, New York, New York, USA, aug 2005. ACM Press.

[54] John Shawe-Taylor and Nello Cristianini. *Kernel Methods for Pattern Analysis.* Cambridge University Press, 2004.

[55] Marco Signoretto, Lieven De Lathauwer, and Johan A. K. Suykens. Learning Tensors in Reproducing Kernel Hilbert Spaces with Multilinear Spectral Penalties. oct 2013.

[56] Marco Signoretto, Lieven De Lathauwer, and Johan A.K. Suykens. A kernel-based framework to tensorial data analysis. *Neural Networks*, 24(8):861–874, 2011.

[57] Marco Signoretto, Emanuele Olivetti, Lieven De Lathauwer, and Johan A. K. Suykens. Classification of Multichannel Signals With Cumulant-Based Kernels. *IEEE Transactions on Signal Processing*, 60(5):2304–2314, may 2012.

[58] Ledyard R Tucker. Some mathematical notes on three-mode factor analysis. *Psychometrika*, 31(3):279–311, sep 1966.

[59] LR Tucker. Implications of factor analysis of three-way matrices for measurement of change. *Problems in measuring change*, 1963.

[60] LR Tucker. The extension of factor analysis to three-dimensional matrices. *Contributions to mathematical psicology*, 1964.

[61] M. Alex O. Vasilescu and Demetri Terzopoulos. Multilinear Analysis of Image Ensembles: TensorFaces. In Anders Heyden, Gunnar Sparr, Mads Nielsen, and Peter Johansen, editors, *Computer Vision — ECCV 2002*, volume 2350 of *Lecture Notes in Computer Science*, Berlin, Heidelberg, apr 2002. Springer Berlin Heidelberg.

[62] Yu-Xiong Wang and Yu-Jin Zhang. Nonnegative Matrix Factorization: A Comprehensive Review. *IEEE Transactions on Knowledge and Data Engineering*, 25(6):1336–1353, jun 2013.

[63] Max Welling and Markus Weber. Positive tensor factorization. *Pattern Recognition Letters*, 22(12):1255–1261, oct 2001.

[64] S Zafeiriou and M Petrou. Nonnegative tensor factorization as an alternative Csiszar–Tusnady procedure: algorithms, convergence, probabilistic interpretations and novel probabilistic tensor. *Data Mining and Knowledge Discovery*, 2011.

[65] Qibin Zhao, Guoxu Zhou, Tulay Adali, Liqing Zhang, and Andrzej Cichocki. Kernelization of Tensor-Based Models for Multiway Data Analysis: Processing of Multidimensional Structured Data. *IEEE Signal Processing Magazine*, 30(4):137–148, jul 2013.

COSTS AND FINANCIAL SOURCES

**Millions of colombian pesos (COP)*

Concept	**Source**	**Total cost***
Salary researcher	Researcher	$144
Advisor salary	Universidad Nacional de Colombia	$80
Conferences and events	Universidad Nacional de Colombia/Researcher	$10
Notebook	Researcher	$2
Desktop computer	Research group MindLAB	$1.6
High performance computer	Research group MindLAB	$20
Office supplies	Researcher	$1.5
Bibliography	Universidad Nacional de Colombia	$1
Total		**$260.1**