Breaking the Curse of Dimensionality using Decompositions of Incomplete Tensors

Nico Vervliet* Otto Debals* Laurent Sorber† Lieven De Lathauwer*‡

MOTIVATION

Tensors (loosely called 'higher-dimensional matrices') provide a richer and more natural representation for many large data sets than a 'flat' matrix. Many new decompositions emerge when using tensors, which enable more compact and more sophisticated models able to capture more complex interactions and couplings instead of the traditional pairwise interactions when working with matrices. Tensor decompositions define multilinear models which are more versatile than linear models, yet easier to work with than non-linear models. Because of this, tensor methods are increasingly used in application areas such as high-dimensional integration [1], signal processing [2], [3] and biomedical analysis [4], [5].

When working with high-dimensional problems, various new computational challenges arise, due to the exponential increase in memory and time requirements when the number of dimensions increases. This is called the *curse of dimensionality* (CoD). We will discuss two promising tools to alleviate this curse when handling big data. First, instead of using full tensors, decompositions can be used to analyze and process data. The use of these decompositions reduces the computational complexity and can even break the CoD. Second, care must be taken such that decomposition algorithms are not unfeasible themselves due to the CoD. Because the total number of elements in the tensor increases exponentially in the number of dimensions, we present efficient *compressed sensing* algorithms working on *incomplete* tensors. The structure in tensor data allows a decomposition to be found, notwithstanding only a small fraction of the elements is known.

In scientific computing, discrete functions where the number of function values is larger than the number of atoms in the universe, can be computed by using efficient tensor representations. In this

*Department of Electrical Engineering (ESAT), KU Leuven, Kasteelpark Arenberg 10, B-3001 Leuven, Belgium (VervlietNico@gmail.com, Otto.Debals@gmail.com, Lieven.DeLathauwer@kuleuven-kulak.be)

†Department of Computer Science, KU Leuven, Celestijnenlaan 200A, B-3001 Leuven, Belgium (Laurent.Sorber@cs.kuleuven.be)

‡Group Science, Engineering and Technology, KU Leuven Kulak, E. Sabbelaan 53, B-8500 Kortrijk, Belgium (Lieven.DeLathauwer@kuleuven-kulak.be)

paper we show how to use this expertise to manage otherwise unmanageable big data sets and we provide a convincing proof-of-concept, the first known example breaking the CoD in data analysis.

TENSORS

Tensors are a higher-order generalization of vectors and matrices, and can be thought of as multidimensional arrays. Because most data sets are structured, an N-th order tensor $\mathcal{T} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ can be written compactly as a decomposition. For simplicity of notation, we assume $I_n = I, n = 1, \ldots, N$. A good overview of tensor decompositions is given in [6].

In this section, we elaborate on the matrix to tensor transition mentioned above. We also explore the advantages of the three decompositions listed below. We focus on tensor decompositions as versatile analysis and modeling tools with mild conditions, and on how these decompositions make the work with huge and high-dimensional data sets feasible by alleviating or breaking the CoD.

- The *Canonical Polyadic Decomposition* (CPD) can be written as a sum of R rank-1 terms: $\mathcal{T} = \sum_{r=1}^{R} \mathbf{a}_r^{(1)} \circ \mathbf{a}_r^{(2)} \circ \cdots \circ \mathbf{a}_r^{(N)}$, and is also known as PARAFAC or CANDECOMP. The number of parameters (RNI) is linear in N, breaking the CoD.
- The *Tucker* decomposition transforms a typically small core tensor $\mathcal{G} \in \mathbb{R}^{R_1 \times R_2 \times \cdots \times R_N}$ by factor matrices $\mathbf{A}^{(n)} \in \mathbb{R}^{I_n \times R_n}, n = 1, \ldots, N$ and can be written as $\mathcal{T} = \mathcal{G} \times_1 \mathbf{A}^{(1)} \times_2 \mathbf{A}^{(2)} \cdots \times_N \mathbf{A}^{(N)}$. Assuming $R_n = R, n = 1, \ldots, N$, the number of parameters now is $NRI + R^N$, which means there is a compression ($R < I$), alleviating the CoD.
- The *Tensor Train* (TT) format is a special case of a *tensor network*, which has an efficient representation and good numerical properties. A TT can be written a series of *tensor carriages* $\mathcal{G}^{(n)}$: $t_{i_1 i_2 \cdots i_N} = \sum_{r_1, r_2, \ldots, r_{N-1}} g_{i_1 r_1}^{(1)} g_{r_1 i_2 r_2}^{(2)} \cdots g_{r_{N-2} i_{N-1} r_{N-1}}^{(N-1)} g_{r_{N-1} i_N}^{(N)}$, where $r_n, n = 1, \ldots, N-1$ are coupling indices [1], [7], [8]. The number of parameters is $\mathcal{O}(NR^2I)$, breaking the CoD again.

DECOMPOSITIONS FOR INCOMPLETE TENSORS

To be able to use a decomposition, it has to be computed first. Algorithms for full tensors cannot be used for very large data sets, because of the high memory and computational requirements (CoD). Therefore, efficient algorithms operating on sparsely observed, or incomplete, tensors are needed. A second reason such algorithms are needed, is that sometimes only a fraction of the elements is known.

We will discuss a number of *compressed sensing* methods (see [9]) from scientific computing and multilinear algebra to compute decompositions of incomplete tensors. More in detail, the following topics will be discussed:

- We illustrate two important reasons to use decompositions of incomplete tensors with only a few known elements: for tensor completion, a generalization of the well-known matrix completion methods [10], [11]; and to use the decomposition itself for further computations and analyses.
- Alternating Least Squares methods are popular for CPD calculation, and can be extended to incomplete tensors (e.g. with expectation maximization). These, however, do not always converge and if so, they converge slowly [4], [12]. Recently, we have successfully extended our efficient Nonlinear Least Squares based CPD algorithms [13] in our Tensorlab toolbox [14] to incomplete tensors, resulting in a fast and memory-efficient new algorithm [15].
- Other methods extract a number of fibers (a vector sample from the tensor where all but one indices are fixed) from the tensors, which are put directly into factor matrices. A heuristic called *cross-approximation* can be used to find the quasi-optimal fibers (which result in maximal volume submatrices), while only exploring a small number of fibers [16]–[18].

A MATERIALS SCIENCE EXAMPLE

We illustrate our efficient incomplete tensor algorithms and the potential of tensor decompositions with a successful, novel application in materials sciences. Physical properties of composite materials vary according to the concentration of each constituent material. The InsPyro data set describes the melting point of composite metals with ten constituent materials and is represented by a ninth order tensor with 10^{18} data points (CoD). Memory requirements and measurement costs, however, limit the number of available points to 10^5, which makes the tensor extremely sparsely sampled. We will show that our algorithms calculate an approximation in a couple of minutes and that the relative point to point errors are smaller than 1% for 99% of the validation points.

Using the decomposition, a compact model is constructed which easily can be evaluated. The model can be used for the inverse problem (which material compositions will have a certain melting point?) and cross-sections can be visualized. Finally, the model can be used in further optimization steps.

REFERENCES

[1] I. Oseledets and E. Tyrtyshnikov, "TT-cross approximation for multidimensional arrays," *Linear Algebra and its Applications*, vol. 432, no. 1, pp. 70 – 88, 2010.

[2] P. Comon and C. Jutten, *Handbook of Blind Source Separation: Independent component analysis and applications*. Academic press, 2010.

[3] N. Sidiropoulos, R. Bro, and G. Giannakis, "Parallel factor analysis in sensor array processing," *Signal Processing, IEEE Transactions on*, vol. 48, no. 8, pp. 2377–2388, 2000.

[4] E. Acar, D. Dunlavy, T. Kolda, and M. Mørup, "Scalable tensor factorizations with missing data," 2010.

[5] V. Orekhov, I. Ibraghimov, and M. Billeter, "Optimizing resolution in multidimensional NMR by three-way decomposition," *Journal of Biomolecular NMR*, vol. 27, pp. 165–173, 2003.

[6] T. Kolda and B. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, 2009.

[7] W. Hackbusch, "Tensor subspace representation," in *Tensor Spaces and Numerical Tensor Calculus*, ser. Springer Series in Computational Mathematics. Springer Berlin Heidelberg, 2012, vol. 42, pp. 217–248.

[8] B. Khoromskij, "Tensors-structured numerical methods in scientific computing: Survey on recent advances," *Chemometrics and Intelligent Laboratory Systems*, vol. 110, no. 1, pp. 1 – 19, 2012.

[9] E. Candès and M. Wakin, "An introduction to compressive sampling," *Signal Processing Magazine, IEEE*, vol. 25, no. 2, pp. 21–30, 2008.

[10] S. Gandy, B. Recht, and I. Yamada, "Tensor completion and low-n-rank tensor recovery via convex optimization," *Inverse Problems*, vol. 27, no. 2, p. 025010, 2011.

[11] E. Candès and Y. Plan, "Matrix completion with noise," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 925–936, 2010.

[12] E. Acar, D. Dunlavy, T. Kolda, and M. Mørup, "Scalable tensor factorizations for incomplete data," *Chemometrics and Intelligent Laboratory Systems*, vol. 106, no. 1, pp. 41 – 56, 2011.

[13] L. Sorber, M. Van Barel, and L. De Lathauwer, "Optimization-based algorithms for tensor decompositions: Canonical polyadic decomposition, decomposition in rank-$(L_r, L_r, 1)$ terms, and a new generalization," *SIAM Journal on Optimization*, vol. 23, no. 2, pp. 695–720, 2013.

[14] L. Sorber, M. Van Barel, and L. De Lathauwer. "Tensorlab v1.0." [Online]. Available: http://esat.kuleuven.be/sista/tensorlab/

[15] O. Debals, N. Vervliet, L. Sorber, and L. De Lathauwer, "Efficiënte tensorgebaseerde methoden voor modellering en signaalscheiding," Master's thesis, KU Leuven, 2013.

[16] C. Caiafa and A. Cichocki, "Generalizing the column-row matrix decomposition to multi-way arrays," *Linear Algebra and its Applications*, vol. 433, no. 3, pp. 557 – 573, 2010.

[17] S. Goreinov, "On cross approximation of multi-index arrays," *Doklady Mathematics*, vol. 77, no. 3, pp. 404–406, 2008.

[18] I. Oseledets, D. Savostianov, and E. Tyrtyshnikov, "Tucker dimensionality reduction of three-dimensional arrays in linear time," *SIAM Journal on Matrix Analysis and Applications*, vol. 30, no. 3, pp. 939–956, 2008.

Nico Vervliet and **Otto Debals** obtained the Master's degree in Mathematical Engineering from KU Leuven in 2013. They currently are Ph.D. candidates at the STADIUS Center for Dynamical Systems, Signal Processing and Data Analytics of the Electrical Engineering Department (ESAT), KU Leuven.

Laurent Sorber obtained the Master's degree in Mathematical Engineering from KU Leuven in 2010. He currently is a Ph.D. candidate at the STADIUS Center for Dynamical Systems, Signal Processing and Data Analytics of the Electrical Engineering Department (ESAT), and at NALAG of the Computer Science Department, KU Leuven.

Lieven De Lathauwer obtained the Master's degree in Electro-Mechanical Engineering and the Ph.D. degree in Applied Sciences from KU Leuven, in 1992 and 1997, respectively. From 2000 to 2007 he was Research Associate with the French Centre National de la Recherche Scientifique. He is currently Professor with KU Leuven. He is affiliated with both the Group Science, Engineering and Technology of Kulak and with the STADIUS Center for Dynamical Systems, Signal Processing and Data Analytics of the Electrical Engineering Department (ESAT). He is Associate Editor of the SIAM Journal on Matrix Analysis and Applications and has served as Associate Editor for the IEEE Transactions on Signal Processing.