Tensors and multilinear algebra: what and why

Eric Evert, Lieven De Lathauwer

Jan 9, 2023 · 9 min read · Machine Learning & Data Science

Over the last century, linear algebra has proven to be one of the most powerful and flexible tools for problem solving in a huge number of scientific fields including signal processing, statistics, machine learning and data science. The central objects in linear algebra are vectors and matrices, i.e., 1-way and 2-way arrays of numbers. The great power of vectors and matrices comes from their ability to represent and provide insight into signals, differential equations, dynamical systems, linear and bilinear forms, etc.

Two of the fundamental techniques in linear algebra are matrix decompositions and solving linear systems of equations. Matrix decompositions have many applications including compression of data, e.g., via Principal Component Analysis (PCA). Such compressions intuitively "narrow down the data to its essence". Another example is that matrix decompositions can be used for decoupling of effects. On the other hand, by solving linear systems, one can arrive at solutions to a wide variety of estimation problems.

Despite the great power and success of linear algebra throughout the sciences, linear algebraic tools are not without shortcomings; alas the world is not linear! While linear algebraic techniques can frequently be successfully applied even to nonlinear problems by "zooming in" enough so that the problem locally is approximately linear, using local techniques necessarily leads to local solutions as opposed to global solutions.

Furthermore, while matrices and vectors are convenient for recording data, many datasets depend on more than two parameters. For example, consider a dataset arising from ratings of movies on a stream service such as Netflix. The ratings of course depend on the user and movie giving two clear dependencies. However, to develop a more refined understanding of the ratings given, one could ask that the user rate the movie on a number of different aspects, e.g., plot, acting, and cinematography. In this manner, one arrives at a dataset that has a total of three dependencies: user x movie x aspect.

To represent the full user x movie x aspect dataset in a single matrix, one could for example either vertically or horizontally concatenate, i.e., "glue together", the matrices corresponding to the user x movie ratings for each aspect. With appropriate assumptions on the resulting matrix, one can reverse engineer which portion of the concatenated matrix corresponds to which aspect; however, this information is not intrinsically available in the concatenated matrix itself. In other words, information about the data has been lost by storing it in a single matrix.

From linear to multilinear

A major advancement in the last few decades has been to move beyond linear to the multilinear world. Though in one sense a natural extension of linear algebra, multilinear algebra is an important and powerful new framework which can be used to address an even more general collection of problems than its predecessor.

The central objects in the field of multilinear algebra are higher-order tensors, i.e., n-way arrays of numbers where n is an integer ≥ 3. These n-way arrays are a direct generalization of the 1-way and 2-way arrays which appear in linear algebra. For example, the previously mentioned movie rating dataset which depends on user, category, and aspect parameters can naturally be represented with a third-order tensor, i.e., a three-way array, in a way that preserves the natural high-order structure of the data. Intuitively, the resulting tensor can be thought of as a stacked collection of 2-way arrays, where each individual 2-way array contains the user x movie ratings corresponding to a particular aspect. This extension from 2-way to 3-way arrays, i.e., from linear to multilinear, may offhand seem simple and perhaps even inconsequential; however, it is in fact of profound importance and opens entirely new possibilities for understanding and working with data, signals, and systems.

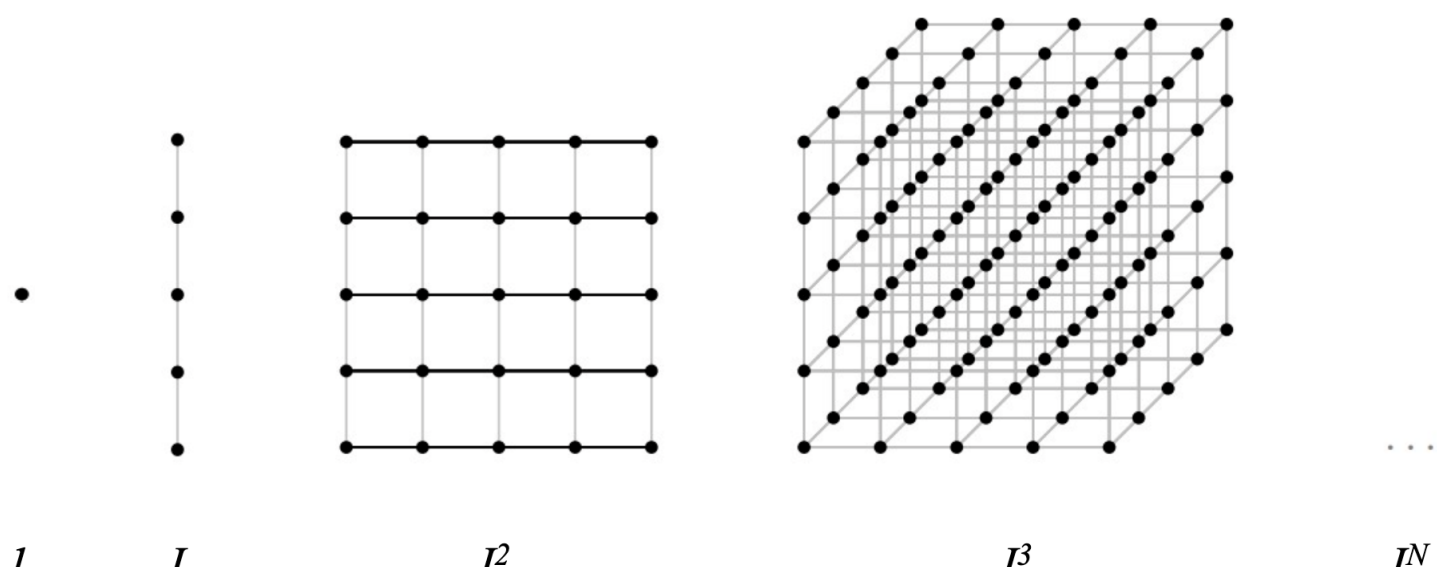


Figure 1: Higher-order tensors are the natural extension of scalars – vectors – matrices. As the order increases, the number of entries increases exponentially. This is known as the Curse of Dimensionality. This curse can be broken by the Blessing of Low Rank.

Tensor decompositions provide tractable methods for understanding data

One of the fundamental tools in multilinear algebra is tensor decompositions. In a naïve sense, tensor decompositions are similar to the matrix decompositions which are fundamental in linear algebra. Both matrix and tensor decompositions are used to express an array in terms of some collection of basic or elementary components. However, this is where the similarities end.

Just as higher-order tensors are more structured than their flat counterparts, tensor decompositions are significantly more structured than matrix decompositions. One of the most fundamental tensor decompositions is the Canonical Polyadic Decomposition (CPD), which expresses a given tensor as a sum of elementary components. The structure present in the CPD is responsible for one of its most important properties: low-rank CPDs are unique with very mild assumptions. Intuitively, the CPD being unique means that there is only one way to write the tensor as a sum of elementary components. The uniqueness of low-rank CPDs is in stark contrast to the matrix setting where decompositions are only unique with strong assumptions such as orthogonality or positivity of the factors.

This uniqueness is of great importance in applied settings. It means that when component information is extracted from a signal or dataset via a CPD, the components that one arrives at are the true components that are responsible for the data at hand. That is, one can be sure they that have not erroneously arrived at some other components. Uniqueness of the CPD has already been shown to have many applications, including telecommunications, radioastronomy, and biomedical data analysis.

In addition to enabling recovery of component information, tensor decompositions can be used for a multilinear version of PCA. This allows one to perform dimensionality reduction in a manner that is more sensitive to higher-order structure than traditional PCA. For example, it may be the case that a dataset is more easily compressed along a given parameter than another. Multilinear PCA is easily able to capture this behaviour. In contrast, traditional PCA can struggle to account for such behaviour since the row rank of a matrix is always equal to the column rank. That is, the amount of compression in the two parameters of a matrix must be the same. Tensors, on the other hand, allow for different compression ratios for different parameters.

Tensor techniques and decompositions also provide numerically reliable means to break the so-called Curse of Dimensionality. The curse refers to issues that are caused by the exponential dependence of computational and memory costs on the number of parameters in a given dataset. This can quickly lead to prohibitive computational costs for naïve methods. However, appropriate tensor methods can be effectively employed to break this curse. As such, tensor methods have revolutionized scientific computing in high dimensions, enabling a fast and reliable computation of functions whose number of unknown values exceeds the number of atoms in the observable universe.

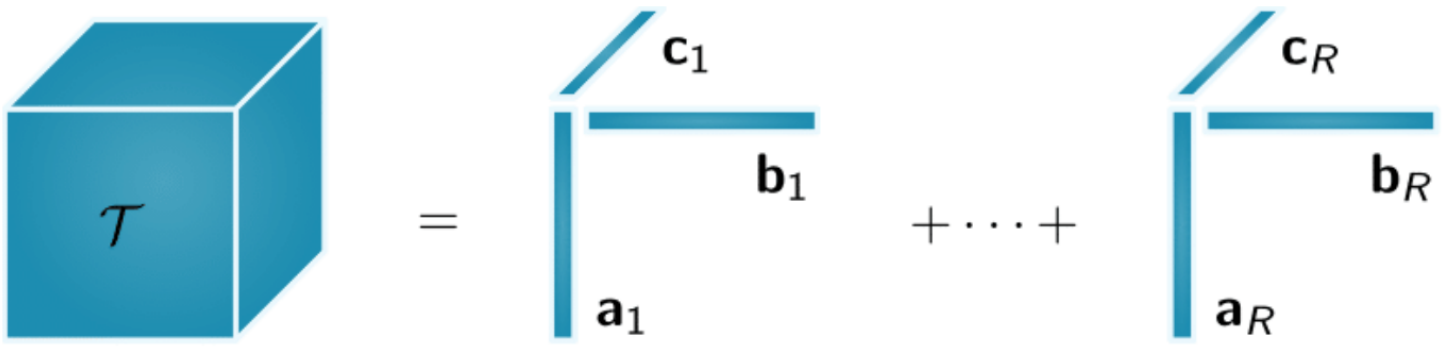


Figure 2: Canonical Polyadic Decomposition writes a tensor as a minimal sum of rank-1 terms. The higher-order setting has the advantage over the matrix setting that the decomposition is easily unique. This enables the unique recovery of signal components from a given dataset. There exist more tensor decompositions, which can be used as engineering tools in their own right.

Bridging the gap to nonlinear

Another important aspect of multilinear algebraic techniques is that they bridge the gap between the linear and nonlinear worlds. A notable shortcoming of linear algebra is that, while it is highly useful in solving linear problems, many problems of interest are nonlinear. As previously mentioned, linear algebraic approaches can often be applied in nonlinear settings by using local linear approximations; however, local techniques necessarily lead to local solutions. This means that linear algebraic approaches can be left wanting in situations where one desires to find all or many solutions to a polynomial or nonlinear system of equations.

While vectors and matrices are limited to representing linear systems of equations and linear and bilinear forms, higher-order tensors can represent general systems of polynomial equations and multilinear (polynomial) forms. As an

immediate consequence, tensor techniques can in many settings be used to arrive at global solutions to systems of multivariate polynomial equations.

Moreover, continuous functions can be closely approximated by multivariate polynomials. That is, tensors can be used to produce accurate approximations to nonlinear functions of interest. This in turn enables practitioners to work with nonlinear problems in a tractable and interpretable manner that is significantly more general than traditional linear algebraic techniques allow. This is of great interest for determining and interpreting nonlinear decision boundaries in machine learning, for instance.

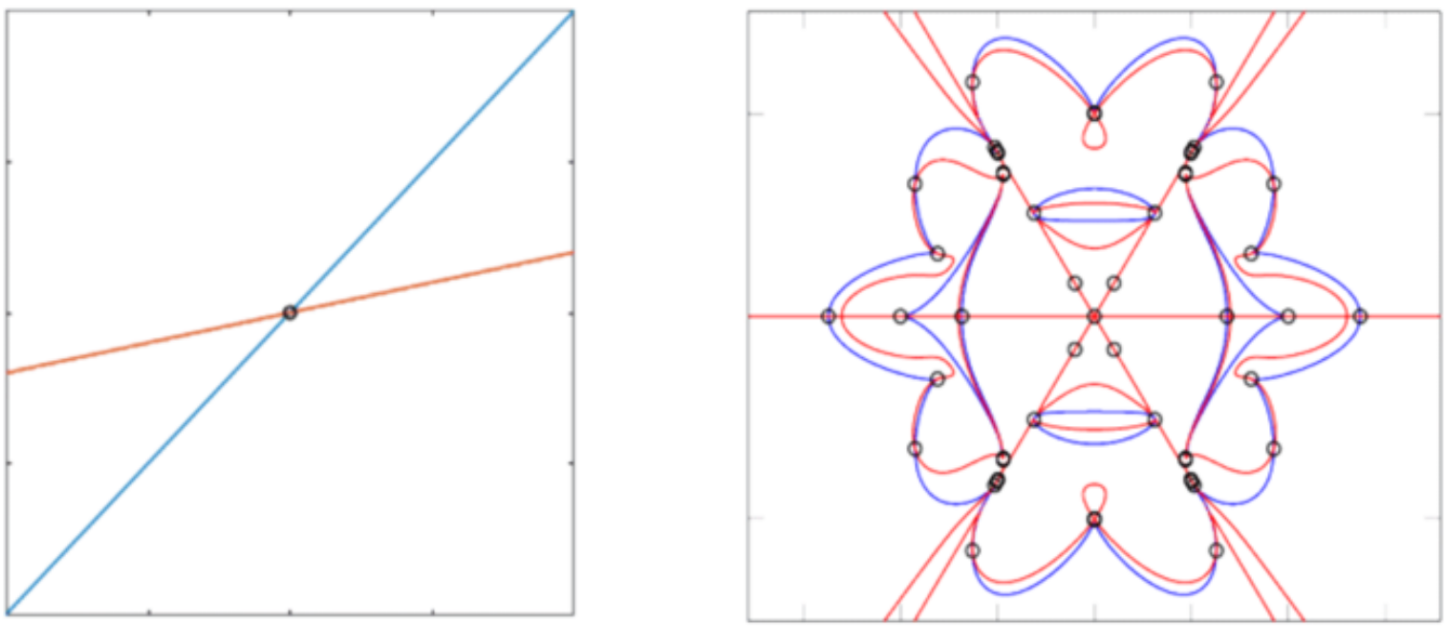


Figure 3: The extension of linear algebra to multilinear algebra includes the extension of sets of linear equations to sets of polynomial equations. The former are often obtained by linearizing around a working point and restricting to a small-signal analysis: the left figure might have been obtained by zooming in on a detail of the right figure.

Multilinear algebra is everywhere

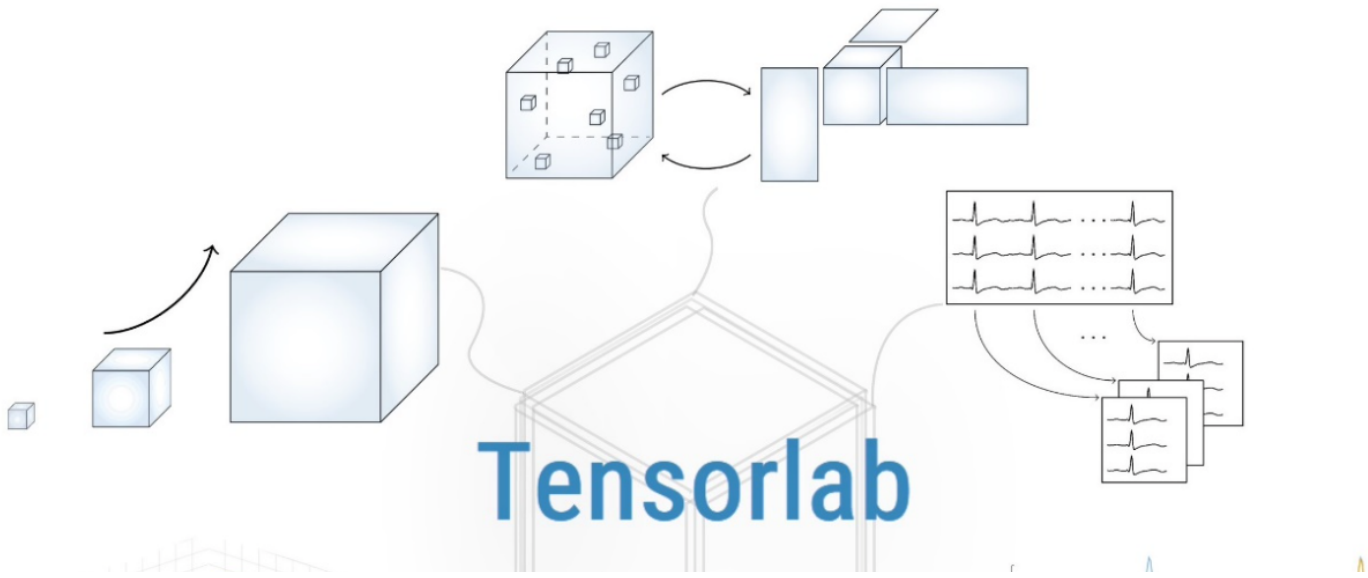
Multilinear algebra not only is incredibly powerful but also is ubiquitous. As previously discussed, higher-order tensors are excellent for representing datasets and signals which depend on three or more parameters, and tensor decompositions can play a powerful role in understanding such datasets. In addition, multilinear algebra is a natural tool for understanding nonlinear interactions between input and output data. Of course, much of the world is nonlinear, so this setting is extremely common.

Aside from these two clear use cases for multilinear algebra, it should be emphasized that multilinear algebra is often found hiding in plain sight. For example, the exponential signals that govern the behaviour of linear time-invariant dynamical systems can in fact be represented using higher-order tensors. Moreover, in many cases, the underlying (univariate or multivariate) exponentials can be recovered from appropriate tensor decompositions.

In addition, multilinear algebra can easily be seen in contexts where one is interested in a collection of matrices. The collection of matrices can be viewed as the “slices” of a tensor and one can then use multilinear algebraic techniques to understand the collection as a whole. This makes multilinear algebra a key tool for tasks such as data fusion and coupled decompositions.

The future is multilinear

The extension from linear to multilinear algebra represents a paradigm shift in the mathematical tools available to the modern scientific world. Multilinear algebra builds upon the already powerful and highly general framework of linear algebra and opens critical new tools for solving problems and for understanding data. Furthermore, multilinear algebra bridges the gap between the linear and nonlinear worlds, offering tractable approaches for closely approximating solutions to nonlinear problems.



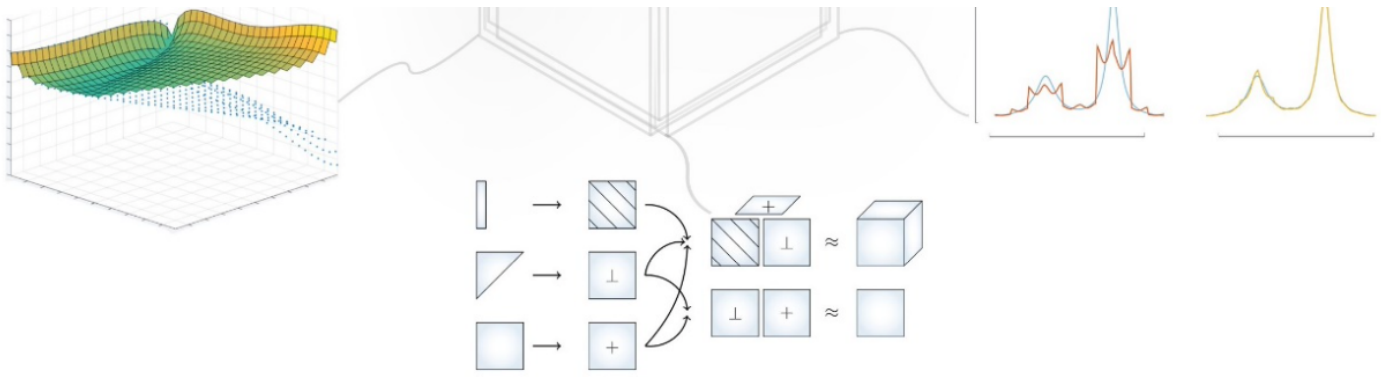


Figure 4: Multilinear methods form a fertile transition ground: more general than linear methods while providing better algebraic and numerical grip than general nonlinear methods. Tensors come with a rich set of tools for engineers and applied scientists. These tools can be used for purposes that are very diverse, such as compression, completion, modelling, estimation, classification, data analysis and data fusion.

To probe further:

- Tensor Decomposition for Signal Processing and Machine Learning. Sidiropoulos N., De Lathauwer L., Fu X., Huang K., Papalexakis E., Faloutsos C., Tensor Decomposition for Signal Processing and Machine Learning, IEEE Transactions on Signal Processing, vol. 65, no. 13, Jul. 2017, pp. 3551-3582.
- Tensorlab. Vervliet N., Debals O., Sorber L., Van Barel M., De Lathauwer L., Tensorlab v3.0, A MATLAB Package for Tensor Computations, March 2016.
- Tensorlab+. Hendrikx S., Boussé M., Vervliet N., Vandecappelle M., Kenis R., De Lathauwer L., Tensorlab+, A Reproducible Research Repository for Tensor Computations, 2nd version, Dec. 2022.

Multilinear algebra Tensors



Eric Evert

Shaw Family CS+X Postdoctoral Fellow



Lieven De Lathauwer

Full Professor