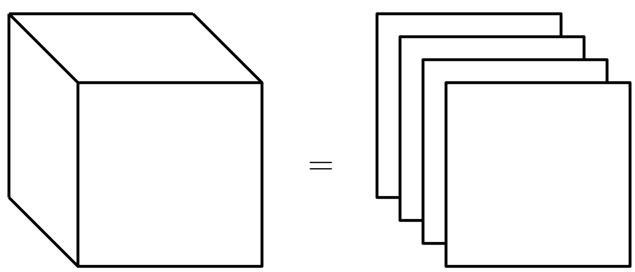
Example Project: Complexity of Tensor Ranks

Guy Moshkovitz

A *tensor* is a 3-dimensional matrix $T \in \mathbb{F}^{n\times n\times n}$ with entries in some field $\mathbb{F}$. For 2-dimensional matrices there is only one notion of *rank* with many equivalent definitions: column rank, row rank, decomposition rank, and determinantal rank, to name a few. For tensors, however, there is a myriad of different notions of rank, coming from different fields of math: combinatorics, analysis, algebraic geometry, computer science, and more. Many questions are open about the interplay between these notions, and equally importantly, about their computational complexity. Linear algebra tells us that the rank of an $n \times n$ matrix can be computed quite efficiently, for example by Gaussian elimination, in time $O(n^3)$ (and possibly in linear time $O(n^2)$!). Finding an efficient algorithm for any of the various notions of tensors rank would be extremely interesting. On the flip side, showing NP-hardness for many of these notions is open as well.

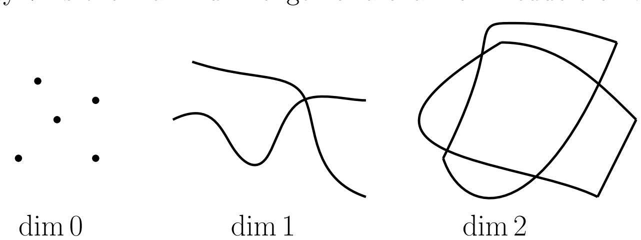
One notion of tensor rank whose computational complexity is unknown is *geometric rank* [4]. It is defined using the concept of a *variety* from algebraic geometry. A variety is the solution set of polynomial equations, $\{\mathbf{x} \in \overline{\mathbb{F}}^n \mid f_1(\mathbf{x}) = 0, \dots, f_m(\mathbf{x}) = 0\}$, where $f_1, \dots, f_m \in \mathbb{F}[x_1, \dots, x_n]$ are polynomials with coefficients in $\mathbb{F}$, and $\overline{\mathbb{F}}$ denotes the algebraic closure of $\mathbb{F}$. To define the geometric rank of a tensor $T = (t_{i,j,k})_{i,j,k\in[n]} \in \mathbb{F}^{n\times n\times n}$, first note that its 2-dimensional slices $T_1, \dots, T_n \in \mathbb{F}^{n\times n}$ along any of the three axes, say $T_k = (t_{i,j,k})_{i,j\in[n]}$, determine a *bilinear map* $\widetilde{T}\colon \mathbb{F}^n \times \mathbb{F}^n \to \mathbb{F}^n$ given by $\widetilde{T}(\mathbf{x},\mathbf{y}) = (\mathbf{x}^t T_1 \mathbf{y}, \dots, \mathbf{x}^t T_n \mathbf{y})$.[1]



The *kernel* of $\widetilde{T}$ is the variety $\ker \widetilde{T} = \{(\mathbf{x},\mathbf{y}) \in \overline{\mathbb{F}}^n \times \overline{\mathbb{F}}^n \mid \widetilde{T}(\mathbf{x},\mathbf{y}) = \mathbf{0}\}$, and the geometric rank of T is the codimension of the kernel,

$$\mathrm{GR}(T) = \mathrm{codim}(\ker \widetilde{T}) = 2n - \dim(\ker \widetilde{T}).$$

The *dimension* of a variety V is the maximum length of a chain of irreducible[2] varieties contained in V.[3]



[1] Just like a 2-dimensional matrix corresponds to a *linear* map, its n components being linear forms.

[2] A variety is irreducible if it cannot be written as a (proper) union of varieties. Every variety can be written uniquely as a finite union of irreducible varieties; e.g., $\{xy = 0\} = \{x = 0\} \cup \{y = 0\}$.

[3] Equivalently, it is the maximum size of an algebraically-independent set in the quotient ring $\mathbb{F}[x_1, \dots, x_n]/\,\mathrm{I}(V)$. (The *ideal* of V is $\mathrm{I}(V) = \{f \in \mathbb{F}[x_1, \dots, x_n] \mid \forall \mathbf{x} \in V\colon f(\mathbf{x}) = 0\}$.)

Note that $\mathrm{GR}(T)$ is an integer in the range $\{0,\ldots,n\}$, where the upper bound follows from the fact that $\{(\mathbf{0},\mathbf{y}) \mid \mathbf{y} \in \overline{\mathbb{F}}^n\} \subseteq \ker \widetilde{T}$. We note that geometric rank is independent of the choice of axis along which we slice the tensor [4], analogously to how matrix rank is independent of whether we form the linear map by slicing the matrix along the rows or along the columns (a.k.a column rank = row rank). Practically speaking, one can compute the geometric rank of small tensors using computer software like Macaulay2 or Sage that can compute the dimension of varieties. However, for general tensors it is not clear whether there is an efficient algorithm or not.

Question 1. *Is deciding* $\mathrm{GR}(T) \le r$ *for* $T \in \mathbb{F}^{n\times n\times n}$ NP*-hard?*

Interestingly, for a system of degree-2 homogeneous equations (though not necessarily bilinear!), even deciding if there is a nonzero solution is known to be NP-hard (see Theorem 2.6 in [3] which reduces from graph 3-colorability). However, even if computing geometric rank *exactly* is hard, approximating it—by computing a value guaranteed to be, say, double the right answer—might be easy.

Question 2. *What is the computational complexity of approximating geometric rank up to a multiplicative constant c? (for various values of c)*

Another rank notion for tensors is *slice rank*, introduced by Tao [6] in the context of the recent breakthrough solution of the cap-set problem. To define the slice rank of a tensor $T = (t_{i,j,k})_{i,j,k\in[n]} \in \mathbb{F}^{n\times n\times n}$, first identify it with the trilinear polynomial $\overline{T} = \sum_{i,j,k\in[n]} t_{i,j,k}x_iy_jz_k$. The tensor T is said to have slice rank 1 if $\overline{T}$ is a product of a linear form and a bilinear form (in disjoint variables groups $\mathbf{x},\mathbf{y},\mathbf{z}$). The slice rank $\mathrm{SR}(T)$ is the smallest number of tensors of slice rank 1 whose sum is T. Deciding $\mathrm{SR} \le r$ was shown to be NP-hard [1] via a reduction from a version of the minimum vertex cover problem for 3-uniform hypergraphs. The approximation question, however, remains open.

Question 3. *What is the computational complexity of approximating slice rank up to a multiplicative constant c? (for various values of c)*

One can ask similar questions about other rank notions for tensors, such as *analytic rank*, *subrank*, and *tensor rank*. For the latter notion, which is defined similarly to slice rank except each summand is a product of three linear forms, these questions have been studied in greater depth, starting from Håstad's classical result from 1990 [2] proving NP-hardness, as well as more recent results showing, for example, NP-hardness of approximating tensor rank to within a factor of 1.0005 [5]. These and other questions about ranks of tensors are closely related to algebraic complexity (arithmetic circuit lower bounds, algorithmic matrix multiplication), quantum information theory (quantifying quantum entanglement), and extremal combinatorics (e.g., the cap-set and the sunflower problems).

References

[1] M. Bläser, C. Ikenmeyer, V. Lysikov, A. Pandey and F. O. Schreyer, Variety membership testing, algebraic natural proofs, and geometric complexity theory, arXiv:1911.02534 (2019).

[2] J. Håstad, Tensor rank is NP-complete, J. Algorithms **11**, 644–654 (1990).

[3] C. Hillar and L. H. Lim, Most tensor problems are NP-hard, J. ACM **60**, No. 45 (2013).

[4] S. Kopparty , G. Moshkovitz and J. Zuiddam, Geometric rank of tensors and subrank of matrix multiplication, 35th Computational Complexity Conference (CCC 2020) **35**, 1–21.

[5] J. Swernofsky, Tensor rank is hard to approximate, Leibniz International Proceedings in Informatics, LIPIcs **26** (2018).

[6] T. Tao, A symmetric formulation of the Croot-Lev-Pach-Ellenberg-Gijswijt capset bound, https://terrytao.wordpress.com/2016/05/18/a-symmetric-formulation-of-the-croot-lev-pach-ellenberg-gijswijt-capset-bound (2016).