Spectral Learning on Matrices and Tensors

Other titles in Foundations and Trends® in Machine Learning

Computational Optimal Transport
Gabriel Peyre and Marco Cuturi
ISBN: 978-1-68083-550-2

An Introduction to Deep Reinforcement Learning
Vincent Francois-Lavet, Peter Henderson, Riashat Islam, Marc G. Bellemare and Joelle Pineau
ISBN: 978-1-68083-538-0

An Introduction to Wishart Matrix Moments
Adrian N. Bishop, Pierre Del Moral and Angele Niclas
ISBN: 978-1-68083-506-9

A Tutorial on Thompson Sampling
Daniel J. Russo, Benjamin Van Roy, Abbas Kazerouni, Ian Osband and Zheng Wen
ISBN: 978-1-68083-470-3

Spectral Learning on Matrices and Tensors

Majid Janzamin
Twitter
majid.janzamin@gmail.com

Rong Ge
Duke University
rongge@cs.duke.edu

Jean Kossaifi
Imperial College London
jean.kossaifi@imperial.ac.uk

Anima Anandkumar
NVIDIA & California Institute of Technology
anima@caltech.edu

now
the essence of knowledge
Boston — Delft

Foundations and Trends® in Machine Learning

Published, sold and distributed by:
now Publishers Inc.
PO Box 1024
Hanover, MA 02339
United States
Tel. +1-781-985-4510
www.nowpublishers.com
sales@nowpublishers.com

Outside North America:
now Publishers Inc.
PO Box 179
2600 AD Delft
The Netherlands
Tel. +31-6-51115274

The preferred citation for this publication is

M. Janzamin, R. Ge, J. Kossaifi and A. Anandkumar. *Spectral Learning on Matrices and Tensors*. Foundations and Trends® in Machine Learning, vol. 12, no. 5-6, pp. 393–536, 2019.

ISBN: 978-1-68083-641-7

Foundations and Trends® in Machine Learning
Volume 12, Issue 5-6, 2019

Editorial Board

Editorial Scope

Topics

Foundations and Trends® in Machine Learning publishes survey and tutorial articles in the following topics:

- Adaptive control and signal processing
- Applications and case studies
- Behavioral, cognitive and neural learning
- Bayesian learning
- Classification and prediction
- Clustering
- Data mining
- Dimensionality reduction
- Evaluation
- Game theoretic learning
- Graphical models
- Independent component analysis
- Inductive logic programming
- Kernel methods
- Markov chain Monte Carlo
- Model choice
- Nonparametric methods
- Online learning
- Optimization
- Reinforcement learning
- Relational learning
- Robustness
- Spectral methods
- Statistical learning theory
- Variational inference
- Visualization

Information for Librarians

Foundations and Trends® in Machine Learning, 2019, Volume 12, 6 issues. ISSN paper version 1935-8237. ISSN online version 1935-8245. Also available as a combined paper and online subscription.

Contents

Spectral Learning on Matrices and Tensors

Majid Janzamin[1], Rong Ge[2], Jean Kossaifi[3] and Anima Anandkumar[4]

[1] *Twitter; majid.janzamin@gmail.com*
[2] *Duke University; rongge@cs.duke.edu*
[3] *Imperial College London; jean.kossaifi@imperial.ac.uk*
[4] *NVIDIA & California Institute of Technology; anima@caltech.edu*

ABSTRACT

Spectral methods have been the mainstay in several domains such as machine learning, applied mathematics and scientific computing. They involve finding a certain kind of spectral decomposition to obtain basis functions that can capture important structures or directions for the problem at hand. The most common spectral method is the principal component analysis (PCA). It utilizes the principal components or the top eigenvectors of the data covariance matrix to carry out dimensionality reduction as one of its applications. This data pre-processing step is often effective in separating signal from noise.

PCA and other spectral techniques applied to *matrices* have several limitations. By limiting to only pairwise moments, they are effectively making a Gaussian approximation on the underlying data. Hence, they fail on data with hidden variables which lead to non-Gaussianity. However, in almost any data set, there are latent effects that cannot be directly observed, e.g., topics in a document corpus, or underlying causes of a disease. By extending the spectral decomposition

Majid Janzamin, Rong Ge, Jean Kossaifi and Anima Anandkumar (2019), Spectral Learning on Matrices and Tensors, Foundations and Trends® in Machine Learning: Vol. 12, No. 5-6, pp 393–536. DOI: 10.1561/2200000057.

methods to higher order moments, we demonstrate the ability to learn a wide range of latent variable models efficiently. Higher-order moments can be represented by *tensors*, and intuitively, they can encode more information than just pairwise moment matrices. More crucially, tensor decomposition can pick up latent effects that are missed by matrix methods. For instance, tensor decomposition can uniquely identify non-orthogonal components. Exploiting these aspects turns out to be fruitful for provable unsupervised learning of a wide range of latent variable models.

We also outline the computational techniques to design efficient tensor decomposition methods. They are embarrassingly parallel and thus scalable to large data sets. Whilst there exist many optimized linear algebra software packages, efficient tensor algebra packages are also beginning to be developed. We introduce Tensorly, which has a simple python interface for expressing tensor operations. It has a flexible back-end system supporting NumPy, PyTorch, TensorFlow and MXNet amongst others. This allows it to carry out multi-GPU and CPU operations, and can also be seamlessly integrated with deep-learning functionalities.

1

Introduction

Probabilistic models form an important area of machine learning. They attempt to model the probability distribution of the observed data, such as documents, speech and images. Often, this entails relating observed data to *latent or hidden variables*, e.g., topics for documents, words for speech and objects for images. The goal of learning is to then discover the latent variables and their relationships to the observed data.

Latent variable models have shown to be useful to provide a good explanation of the observed data, where they can capture the effect of hidden causes which are not directly observed. Learning these hidden factors is central to many applications, e.g., identifying latent diseases through observed symptoms, and identifying latent communities through observed social ties. Furthermore, latent representations are very useful in feature learning. Raw data is in general very complex and redundant and feature learning is about extracting informative features from raw data. Learning efficient and useful features is crucial for the performance of learning tasks, e.g., the classification task that we perform using the learned features.

Learning latent variable models is challenging since the latent variables cannot, by definition, be directly observed. In extreme cases, when

there are more latent variables than observations, learning is theoretically impossible because of the lack of data, unless further constraints are imposed. More generally, learning latent variable models raises several questions. How much data do we need to observe in order to uniquely determine the model's parameters? Are there efficient algorithms to effectively learn these parameters? Can we get provable guarantees on the running time of the algorithm and the number of samples required to estimate the parameters? These are all important questions about learning latent variable models that we will try to address here.

In this monograph, we survey recent progress in using spectral methods including matrix and tensor decomposition techniques to learn many popular latent variable models. With careful implementation, tensor-based methods can run efficiently in practice, and in many cases they are the only algorithms with provable guarantees on running time and sample complexity.

There exist other surveys and overviews on tensor decomposition and its applications in machine learning and beyond. Among them, the work by Kolda and Bader (2009) is very well-received in the community where they provide a comprehensive introduction to major tensor decomposition forms and algorithms and discuss some of their applications in science and engineering. More recently, Sidiropoulos *et al.* (2017) provide an overview of different types of tensor decompositions and some of their applications in signal processing and machine learning. Papalexakis *et al.* (2017) discuss several applications of tensor decompositions in data mining. Rabanser *et al.* (2017) review some basic concepts of tensor decompositions and a few applications. Debals and De Lathauwer (2017) review several tensorization techniques which had been proposed in the literature. Here, tensorization is the mapping of a vector or matrix to a tensor to enable us using tensor tools.

In contrast to the above works, our focus in this monograph is on a special type of tensor decomposition called CP decomposition (see (1.3) as an example), and we cover a wide range of algorithms to find the components of such tensor decomposition. We also discuss the usefulness of this decomposition by reviewing several probabilistic models that can be learned using such tensor methods.

1.1 Method of Moments and Moment Tensors

How can we learn latent variable models, even though we cannot observe the latent variables? The key lies in understanding the relationship between latent variables and observed variables. A common framework for such relationship is known as the **method of moments** which dates back to Pearson (1894).

Pearson's 1-d Example: The main idea of method of moments is to first estimate *moments* of the data, and use these estimates to learn the unknown parameters of the probabilistic model. For a one-dimensional random variable $X \in \mathbb{R}$, the r-th order moment is denoted by $\mathbb{E}[X^r]$, where r is a positive integer and $\mathbb{E}[\cdot]$ is the expectation operator. Consider a simple example where X is a mixture of two Gaussian variables. More precisely, with probability p_1, X is drawn from a Gaussian distribution with mean μ_1 and variance σ_1^2, and with probability p_2, X is drawn from a Gaussian distribution with mean μ_2 and variance σ_2^2. Here we have $p_1 + p_2 = 1$. Let us consider the problem of estimating these unknown parameters given samples of X. The random variable X can be viewed as drawn from a latent variable model because given a sample of X, we do not know which Gaussian it came from. Let latent variable $Z \in \{1, 2\}$ be a random variable with probability p_1 of being 1. Then given Z, X is just a Gaussian distribution as

$$[X|Z = z] \sim \mathcal{N}(\mu_z, \sigma_z^2).$$

As noted by Pearson (1894), even though we cannot observe Z, the moments of X are closely related to the unknown parameters (probabilities p_1, p_2, means μ_1, μ_2, standard deviations σ_1, σ_2) we desire to estimate. More precisely, for the first three moments we have

$$\begin{aligned}
\mathbb{E}[X] &= p_1\mu_1 + p_2\mu_2, \\
\mathbb{E}[X^2] &= p_1(\mu_1^2 + \sigma_1^2) + p_2(\mu_2^2 + \sigma_2^2), \\
\mathbb{E}[X^3] &= p_1(\mu_1^3 + 3\mu_1\sigma_1^2) + p_2(\mu_2^3 + 3\mu_2\sigma_2^2).
\end{aligned}$$

The moments $\mathbb{E}[X], \mathbb{E}[X^2], \mathbb{E}[X^3], \ldots$ can be empirically estimated given observed data. Therefore, the equations above can be interpreted as a system of equations on the six unknown parameters stated above. Pearson (1894) showed that with the first 6-th moments, we have enough equations to *uniquely* determine the values of the parameters.

Moments for Multivariate Random Variables of Higher Dimensions: For a scalar random variable, its p-th moment is just a scalar number. However, for a random vector, higher order moments can reveal much more information. Let us consider a random vector $X \in \mathbb{R}^d$. The first moment of this variable is a vector $\mu \in \mathbb{R}^d$ such that $\mu_i = \mathbb{E}[X_i], \forall i \in [d]$, where $[d] := \{1, 2, \ldots, d\}$. For the second order moment, we are not only interested in the second moments of individual coordinates $\mathbb{E}[X_i^2]$, but also in the *correlation* between different coordinates $\mathbb{E}[X_i X_j], i \neq j$. Therefore, it is convenient to represent the second order moment as a $d \times d$ symmetric matrix M, where $M_{i,j} = \mathbb{E}[X_i X_j]$.

This becomes more complicated when we look at higher order moments. For 3rd order moment, we are interested in the correlation between all *triplets* of variables. In order to represent this compactly, we use a 3-dimensional $d \times d \times d$ object T, also known as a 3rd order tensor. The tensor is constructed such that $T_{i,j,k} = \mathbb{E}[X_i X_j X_k], \forall i, j, k \in [d]$. This tensor has d^3 elements or $\binom{d+2}{3}$ distinct entries. In general, p-th order moment can be represented as a p-th order tensor with d^p entries. These tensors are called moment tensors. Vectors and matrices are special cases of moment tensors of order 1 and 2, respectively.

In applications, it is often crucial to define what the random variable X is, and examine what moments of X we can estimate from the data. We now provide a simple example to elaborate on how to form a useful moment and defer the proposal of many more examples to Section 4.

1.2 Warm-up: Learning a Simple Model with Tensors

In this section, we will give a simple example to demonstrate what is a tensor decomposition, and how it can be applied to learning latent variable models. Similar ideas can be applied to more complicated models, which we will discuss in Section 4.

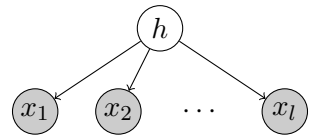

Figure 1.1: Pure Topic Model

Pure Topic Model: The model we consider is a very simple topic model (Papadimitriou *et al.*, 2000; Hofmann, 1999). In this model, there are k unknown topics. Each topic entails a probability distribution over words in the vocabulary. Intuitively, the probabilities represent the likelihood of using a particular word when talking about a specific topic. As an example, the word "snow" should have a high probability in the topic "weather" but not the topic "politics". These probabilities are represented as a matrix $A \in \mathbb{R}^{d\times k}$, where d is the size of the vocabulary and every column represents a topic. So, the columns of matrix A correspond to the probabilities over vocabulary that each topic entails. We will use $\mu_j \in \mathbb{R}^d, j \in [k]$ to denote these probability distribution of words given j-th topic (j-th column of matrix A).

The model assumes each document is generated in the following way: first a topic $h \in [k]$ is chosen with probability w_h where $w \in \mathbb{R}^k$ is a vector of probabilities; next, l words $x_1, x_2, \ldots, x_l$ are independently sampled from the h-th topic-word probability vector μ_h. Therefore, we finally observe words for the documents. See Figure 1.1 for a graphical illustration of this model. This is clearly a latent variable model, since we don't observe the topics. Our goal is to learn the parameters, which include the topic probability vector w and the topic-word probability vectors $\mu_1, \ldots, \mu_k$.

Computing the Moments: First, we need to identify what the interesting moments are in this case. Since all we can observe are words in documents, and documents are all generated independently at random, it is natural to consider correlations between words as moments.

We say $x \in \mathbb{R}^d$ is an indicator vector of a word z in our size-d vocabulary if the z-th coordinate of x is 1 and all other coordinates of

x are 0. For each document, let $x_1, x_2, x_3 \in \mathbb{R}^d$ be indicator vectors for the first three words. Given these word representations, the entries of the first three moments of x_1, x_2, x_3 can be written as

$$\begin{aligned} M_1(i) &= \Pr[x_1 = e_i], \\ M_2(i_1, i_2) &= \Pr[x_1 = e_{i_1}, x_2 = e_{i_2}], \\ M_3(i_1, i_2, i_3) &= \Pr[x_1 = e_{i_1}, x_2 = e_{i_2}, x_3 = e_{i_3}], \end{aligned}$$

where $e_i \in \mathbb{R}^d$ denotes the i-th basis vector in d-dimensional space. Intuitively, the first moment M_1 represents the probabilities for words; the second moment M_2 represents the probabilities that two words co-occur; and the third moment M_3 represents the probabilities that three words co-occur.

We can empirically estimate M_1, M_2, M_3 from the observed documents. Now in order to apply the method of moments, we need to represent these probabilities based on the unknown parameters of our model. We can show that

$$M_1 = \sum_{h=1}^{k} w_h \ \mu_h, \tag{1.1}$$

$$M_2 = \sum_{h=1}^{k} w_h \ \mu_h \mu_h^\top, \tag{1.2}$$

$$M_3 = \sum_{h=1}^{k} w_h \ \mu_h \otimes \mu_h \otimes \mu_h. \tag{1.3}$$

The computation follows from the law of total expectations (explained in more details in Section 4). Here, the first moment M_1 is the weighted average of μ_h; the second moment M_2 is the weighted average of outer-products $\mu_h \mu_h^\top$; and the third moment M_3 is the weighted average of *tensor-products* $\mu_h \otimes \mu_h \otimes \mu_h$. The tensor product $\mu_h \otimes \mu_h \otimes \mu_h$ is a $d \times d \times d$ array whose (i_1, i_2, i_3)-th entry is equal to $\mu_h(i_1)\mu_h(i_2)\mu_h(i_3)$. See Section 3 for more precise definition of the tensor product operator $\otimes$.

Note that the second moment M_2 is a matrix of rank at most k, and Equation (1.2) provides a low-rank matrix decomposition of M_2.

Similarly, finding w_h and μ_h from M_3 using Equation (1.3) is a problem called *tensor decomposition*. Clearly, if we can solve this problem, and it gives a unique solution, then we have learned the parameters of the model and we are done.

1.3 What's Next?

In the rest of this monograph, we will discuss the properties of tensor decomposition problem, review algorithms to efficiently find the components of such decomposition, and explain how they can be applied to learn the parameters of various probabilistic models such as latent variable models.

In Section 2, we first give a brief review of some basic matrix decomposition problems, including the singular value decomposition (SVD) and canonical correlation analysis (CCA). In particular, we will emphasize why matrix decomposition is often not enough to learn all the parameters of the latent variable models.

Section 3 discusses several algorithms for tensor decomposition. We will highlight under what conditions the tensor decomposition is *unique*, which is crucial in identifying the parameters of latent variable models.

In Section 4, we give more examples on how to apply tensor decomposition to learn different latent variable models. In different situations, there are many tricks to manipulate the moments in order to get a clean equation that looks similar to (1.3).

In Section 5, we illustrate how to implement tensor operations in practice using the Python programming language. We then show how to efficiently perform tensor learning using TensorLy and scale things up using PyTorch.

Tensor decomposition and its applications in learning latent variable models are still active research directions. In the last two sections of this monograph we discuss some of the more recent results, which deals with the problem of overcomplete tensors and improves the guarantees on running time and sample complexity.

References

Acar, E., S. A. Çamtepe, M. S. Krishnamoorthy, and B. Yener. 2005. "Modeling and multiway analysis of chatroom tensors". In: *Intelligence and Security Informatics*. Springer. 256–268.

Alain, G. and Y. Bengio. 2012. "What regularized auto-encoders learn from the data generating distribution". *arXiv preprint arXiv:1211.4246*.

Allen-Zhu, Z. and Y. Li. 2016a. "Doubly accelerated methods for faster CCA and generalized eigendecomposition". *arXiv preprint arXiv:1607.06017*.

Allen-Zhu, Z. and Y. Li. 2016b. "First Efficient Convergence for Streaming k-PCA: a Global, Gap-Free, and Near-Optimal Rate". *arXiv preprint arXiv:1607.07837*.

Anandkumar, A., R. Ge, D. Hsu, and S. M. Kakade. 2013. "A Tensor Spectral Approach to Learning Mixed Membership Community Models". In: *Conference on Learning Theory (COLT)*.

Anandkumar, A., R. Ge, D. Hsu, S. M. Kakade, and M. Telgarsky. 2014a. "Tensor Methods for Learning Latent Variable Models". *J. of Machine Learning Research*. 15: 2773–2832.

Anandkumar, A., Y. Deng, R. Ge, and H. Mobahi. 2017. "Homotopy Analysis for Tensor PCA". In: *Conference on Learning Theory*.

Anandkumar, A., D. P. Foster, D. Hsu, S. M. Kakade, and Y.-K. Liu. 2012a. "A Spectral Algorithm for Latent Dirichlet Allocation". In: *Advances in Neural Information Processing Systems 25*.

Anandkumar, A., R. Ge, and M. Janzamin. 2014b. "Sample Complexity Analysis for Learning Overcomplete Latent Variable Models through Tensor Methods". *arXiv preprint arXiv:1408.0553*. Aug.

Anandkumar, A., D. Hsu, F. Huang, and S. M. Kakade. 2012b. "Learning Mixtures of Tree Graphical Models". In: *Advances in Neural Information Processing Systems 25*.

Anandkumar, A., D. Hsu, and S. M. Kakade. 2012c. "A method of moments for mixture models and hidden Markov models". In: *COLT*.

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

Arora, S., R. Ge, T. Ma, and A. Risteski. 2017. "Provable learning of Noisy-or Networks". In: *Proceedings of the forty-nineth annual ACM symposium on Theory of computing*.

Astrid, M. and S. Lee. 2017. "CP-decomposition with Tensor Power Method for Convolutional Neural Networks Compression". *CoRR*. abs/1701.07148.

Austin, T. 2008. "On exchangeable random variables and the statistics of large graphs and hypergraphs". *Probab. Survey.* 5: 80–145.

Barak, B. and A. Moitra. 2016. "Noisy tensor completion via the sum-of-squares hierarchy". In: *Conference on Learning Theory*. 417–445.

Barak, B. and D. Steurer. 2014. "Sum-of-squares proofs and the quest toward optimal algorithms". *arXiv preprint arXiv:1404.5236*.

Baum, L. E. and T. Petrie. 1966. "Statistical inference for probabilistic functions of finite state Markov chains". *The annals of mathematical statistics*. 37(6): 1554–1563.

Bhaskara, A., M. Charikar, A. Moitra, and A. Vijayaraghavan. 2014. "Smoothed analysis of tensor decompositions". In: *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*. ACM. 594–603.

Bjerhammar, A. 1951. *Application of calculus of matrices to method of least squares: with special reference to geodetic calculations*. Elander.

Blei, D. M., A. Y. Ng, and M. I. Jordan. 2003. "Latent dirichlet allocation". *Journal of machine Learning research*. 3(Jan): 993–1022.

Blum, A., J. Hopcroft, and R. Kannan. 2016. "Foundations of data science". *Vorabversion eines Lehrbuchs*.

Cardoso, J.-F. and P. Comon. 1996. "Independent Component Analysis, A Survey Of Some Algebraic Methods". In: *IEEE International Symposium on Circuits and Systems*. 93–96.

Carroll, J. D. and J.-J. Chang. 1970. "Analysis of individual differences in multidimensional scaling via an N-way generalization of "Eckart-Young" decomposition". *Psychometrika*. 35(3): 283–319.

Chang, J. T. 1996. "Full reconstruction of Markov models on evolutionary trees: Identifiability and consistency". *Mathematical Biosciences*. 137: 51–73.

Comon, P. 1994. "Independent Component Analysis, a new concept?" *Signal Processing*. 36(3): 287–314.

Comon, P. 2002. "Tensor decompositions". *Mathematics in Signal Processing V*: 1–24.

Comon, P. and C. Jutten. 2010. *Handbook of Blind Source Separation: Independent Component Analysis and Applications. Academic Press*. Elsevier.

Davis, C. and W. M. Kahan. 1970. "The rotation of eigenvectors by a perturbation. III". *SIAM Journal on Numerical Analysis*. 7(1): 1–46.

De Lathauwer, L., J. Castaing, and J.-F. Cardoso. 2007. "Fourth-order cumulant-based blind identification of underdetermined mixtures". *Signal Processing, IEEE Transactions on*. 55(6): 2965–2973.

Debals, O. and L. De Lathauwer. 2017. "The concept of tensorization". *Tech. rep*. Technical Report 17–99, ESAT–STADIUS, KU Leuven, Belgium.

Delfosse, N. and P. Loubaton. 1995. "Adaptive blind separation of independent sources: a deflation approach". *Signal processing*. 45(1): 59–83.

Eckart, C. and G. Young. 1936. "The approximation of one matrix by another of lower rank". *Psychometrika*. 1(3): 211–218.

Feige, U. 2002. "Relations between average case complexity and approximation complexity". In: *Proceedings of the thiry-fourth annual ACM symposium on Theory of computing*. ACM. 534–543.

Friedman, J. H. 1987. "Exploratory projection pursuit". *Journal of the American statistical association*. 82(397): 249–266.

Frieze, A., M. Jerrum, and R. Kannan. 1996. "Learning linear transformations". In: *Foundations of Computer Science, 1996. Proceedings., 37th Annual Symposium on*. IEEE. 359–368.

Ge, R., F. Huang, C. Jin, and Y. Yuan. 2015a. "Escaping from saddle points—online stochastic gradient for tensor decomposition". In: *Conference on Learning Theory*. 797–842.

Ge, R., Q. Huang, and S. M. Kakade. 2015b. "Learning mixtures of gaussians in high dimensions". In: *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*. ACM. 761–770.

Ge, R., C. Jin, P. Netrapalli, A. Sidford, *et al.* 2016. "Efficient algorithms for large-scale generalized eigenvector computation and canonical correlation analysis". In: *International Conference on Machine Learning*. 2741–2750.

Ge, R. and T. Ma. 2015. "Decomposing overcomplete 3rd order tensors using sum-of-squares algorithms". In: *RANDOM*.

Golub, G. H. and C. F. van Loan. 1996. *Matrix Computations*. Johns Hopkins University Press.

Golub, G. H. and C. F. Van Loan. 1990. "Matrix computations".

Grigoriev, D. 2001. "Linear lower bound on degrees of Positivstellensatz calculus proofs for the parity". *Theoretical Computer Science*. 259(1-2): 613–622.

Harshman, R. A. 1970. "Foundations of the PARAFAC procedure: models and conditions for an" explanatory" multimodal factor analysis".

Harshman, R. A. and M. E. Lundy. 1994. "PARAFAC: Parallel factor analysis". *Computational Statistics & Data Analysis*. 18(1): 39–72.

Hillar, C. J. and L.-H. Lim. 2013. "Most tensor problems are NP-hard". *Journal of the ACM (JACM)*. 60(6): 45.

Hitchcock, F. L. 1927. "The expression of a tensor or a polyadic as a sum of products". *Journal of Mathematics and Physics*. 6(1-4): 164–189.

Hofmann, T. 1999. "Probabilistic latent semantic analysis". In: *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc. 289–296.

Hopkins, S. B., J. Shi, and D. Steurer. 2015. "Tensor principal component analysis via sum-of-square proofs". In: *Conference on Learning Theory*. 956–1006.

Horn, R. A. and C. R. Johnson. 2012. *Matrix analysis*. Cambridge university press.

Hotelling, H. 1933. "Analysis of a complex of statistical variables into principal components." *Journal of educational psychology*. 24(6): 417.

Hotelling, H. 1992. "Relations between two sets of variates". In: *Breakthroughs in statistics*. Springer. 162–190.

Hsu, D. and S. M. Kakade. 2013. "Learning mixtures of spherical Gaussians: moment methods and spectral decompositions". In: *Fourth Innovations in Theoretical Computer Science*.

Hsu, D., S. M. Kakade, and P. Liang. 2012. "Identifiability and unmixing of latent parse trees". In: *Advances in Neural Information Processing Systems 25*.

Hunter, J. D. 2007. "Matplotlib: A 2D Graphics Environment". *Computing in Science Engineering*. 9(3): 90–95.

Hyvärinen, A. and E. Oja. 2000. "Independent component analysis: algorithms and applications". *Neural Networks*. 13(4–5): 411–430.

Hyvärinen, A. 2005. "Estimation of non-normalized statistical models by score matching". In: *Journal of Machine Learning Research*. 695–709.

Janzamin, M., H. Sedghi, and A. Anandkumar. 2014. "Score Function Features for Discriminative Learning: Matrix and Tensor Frameworks". *arXiv preprint arXiv:1412.2863*. Dec.

Janzamin, M., H. Sedghi, and A. Anandkumar. 2015. "Beating the perils of non-convexity: Guaranteed training of neural networks using tensor methods". *arXiv preprint arXiv:1506.08473*.

Johnson, R. and T. Zhang. 2013. "Accelerating stochastic gradient descent using predictive variance reduction". In: *Advances in neural information processing systems*. 315–323.

Jones, E., T. Oliphant, P. Peterson, *et al.* 2001. "SciPy: Open source scientific tools for Python". [Online; accessed 2016-10-21]. URL: http://www.scipy.org/.

Kim, Y., E. Park, S. Yoo, T. Choi, L. Yang, and D. Shin. 2016. "Compression of Deep Convolutional Neural Networks for Fast and Low Power Mobile Applications". *ICLR*. May.

Koivunen, A. and A. Kostinski. 1999. "The feasibility of data whitening to improve performance of weather radar". *Journal of Applied Meteorology*. 38(6): 741–749.

Kolbeinsson, A., J. Kossaifi, Y. Panagakis, A. Anandkumar, I. Tzoulaki, and P. Matthews. 2019. "Stochastically Rank-Regularized Tensor Regression Networks". *CoRR*. abs/1902.10758.

Kolda, T. G. and J. R. Mayo. 2011. "Shifted Power Method for Computing Tensor Eigenpairs". *SIAM Journal on Matrix Analysis and Applications*. 32(4): 1095–1124.

Kolda, T. G. and B. W. Bader. 2009. "Tensor decompositions and applications". *SIAM review*. 51(3): 455–500.

Kossaifi, J., A. Bulat, G. Tzimiropoulos, and M. Pantic. 2019a. "T-Net: Parametrizing Fully Convolutional Nets with a Single High-Order Tensor". In: *CVPR*. 7822–7831.

Kossaifi, J., A. Khanna, Z. Lipton, T. Furlanello, and A. Anandkumar. 2017. "Tensor contraction layers for parsimonious deep nets". In: *Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on*. IEEE. 1940–1946.

Kossaifi, J., Z. C. Lipton, A. Khanna, T. Furlanello, and A. Anandkumar. 2018. "Tensor Regression Networks". *CoRR*. abs/1707.08308.

Kossaifi, J., Y. Panagakis, A. Anandkumar, and M. Pantic. 2019b. "TensorLy: Tensor Learning in Python". *Journal of Machine Learning Research*. 20(26): 1–6. URL: http://jmlr.org/papers/v20/18-277.html.

Kruskal, J. 1976. "More factors than subjects, tests and treatments: an indeterminacy theorem for canonical decomposition and individual differences scaling". *Psychometrika*. 41(3): 281–293.

Kruskal, J. 1977. "Three-way arrays: Rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics". *Linear algebra and its applications*. 18(2): 95–138.

Latała, R. 2005. "Some estimates of norms of random matrices". *Proceedings of the American Mathematical Society*. 133(5): 1273–1282.

Lathauwer, L. D., B. D. Moor, and J. Vandewalle. 2000. "On the Best rank-1 and Rank-$(R_1, R_2, ..., R_N)$ Approximation and Applications of Higher-Order Tensors". *SIAM J. Matrix Anal. Appl.* 21(4): 1324–1342.

Lebedev, V., Y. Ganin, M. Rakhuba, I. V. Oseledets, and V. S. Lempitsky. 2015. "Speeding-up Convolutional Neural Networks Using Fine-tuned CP-Decomposition". In: *ICLR*.

Leurgans, S., R. Ross, and R. Abel. 1993. "A decomposition for three-way arrays". *SIAM Journal on Matrix Analysis and Applications*. 14(4): 1064–1083.

Lim, L.-H. 2005. "Singular values and eigenvalues of tensors: a variational approach". *Proceedings of the IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP '05)*. 1: 129–132.

Ma, T., J. Shi, and D. Steurer. 2016. "Polynomial-time tensor decompositions with sum-of-squares". In: *Foundations of Computer Science (FOCS), 2016 IEEE 57th Annual Symposium on*. IEEE. 438–446.

MacQueen, J. B. 1967. "Some Methods for Classification and Analysis of Multivariate Observations". In: *Proceedings of the fifth Berkeley Symposium on Mathematical Statistics and Probability*. Vol. 1. University of California Press. 281–297.

Mocks, J. 1988. "Topographic components model for event-related potentials and some biophysical considerations". *IEEE transactions on biomedical engineering*. 6(35): 482–484.

Moore, E. H. 1920. "On the reciprocal of the general algebraic matrix". *Bull. Am. Math. Soc.* 26: 394–395.

Mossel, E. and S. Roch. 2006. "Learning Nonsingular Phylogenies and Hidden Markov Models". *Annals of Applied Probability*. 16(2): 583–614.

Nguyen, N. H., P. Drineas, and T. D. Tran. 2010. "Tensor sparsification via a bound on the spectral norm of random tensors". *arXiv preprint arXiv:1005.4732*.

Nocedal, J. and S. J. Wright. 1999. *Numerical Optimization*. Springer.

Novikov, A., D. Podoprikhin, A. Osokin, and D. Vetrov. 2015. "Tensorizing Neural Networks". In: *Proceedings of the 28th International Conference on Neural Information Processing Systems. NIPS'15*. Montreal, Canada. 442–450.

Papadimitriou, C. H., P. Raghavan, H. Tamaki, and S. Vempala. 2000. "Latent semantic indexing: A probabilistic analysis". *Journal of Computer and System Sciences*. 61(2): 217–235.

Papalexakis, E. E., C. Faloutsos, and N. D. Sidiropoulos. 2017. "Tensors for data mining and data fusion: Models, applications, and scalable algorithms". *ACM Transactions on Intelligent Systems and Technology (TIST)*. 8(2): 16.

Paszke, A., S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. 2017. "Automatic Differentiation in PyTorch". In: *NIPS Autodiff Workshop*.

Pearson, K. 1894. "Contributions to the mathematical theory of evolution". *Philosophical Transactions of the Royal Society of London. A*. 185: 71–110.

Pearson, K. 1901. "LIII. On lines and planes of closest fit to systems of points in space". *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*. 2(11): 559–572.

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. "Scikit-learn: Machine Learning in Python". *Journal of Machine Learning Research*. 12: 2825–2830.

Penrose, R. 1955. "A generalized inverse for matrices". In: *Mathematical proceedings of the Cambridge philosophical society*. Vol. 51. No. 3. Cambridge University Press. 406–413.

Potechin, A. and D. Steurer. 2017. "Exact tensor completion with sum-of-squares". *arXiv preprint arXiv:1702.06237*.

Qi, L. 2005. "Eigenvalues of a real supersymmetric tensor". *Journal of Symbolic Computation*. 40(6): 1302–1324.

Rabanser, S., O. Shchur, and S. Günnemann. 2017. "Introduction to tensor decompositions and their applications in machine learning". *arXiv preprint arXiv:1711.10781*.

Raz, R. 2013. "Tensor-rank and lower bounds for arithmetic formulas". *Journal of the ACM (JACM)*. 60(6): 40.

Richard, E. and A. Montanari. 2014. "A statistical model for tensor PCA". In: *Advances in Neural Information Processing Systems*. 2897–2905.

Schoenebeck, G. 2008. "Linear level Lasserre lower bounds for certain k-CSPs". In: *Foundations of Computer Science, 2008. FOCS'08. IEEE 49th Annual IEEE Symposium on*. IEEE. 593–602.

Sedghi, H., M. Janzamin, and A. Anandkumar. 2016. "Provable tensor methods for learning mixtures of generalized linear models". In: *Artificial Intelligence and Statistics*. 1223–1231.

Shalev-Shwartz, S. and T. Zhang. 2013. "Stochastic dual coordinate ascent methods for regularized loss minimization". *Journal of Machine Learning Research*. 14(Feb): 567–599.

Shashua, A. and A. Levin. 2001. "Linear image coding for regression and classification using the tensor-rank principle". In: *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*. Vol. 1. IEEE. I–42.

Shwe, M. A., B. Middleton, D. Heckerman, M. Henrion, E. Horvitz, H. Lehmann, and G. Cooper. 1991. "Probabilistic diagnosis using a reformulation of the INTERNIST-1/QMR knowledge base". *Methods of information in Medicine*. 30(4): 241–255.

Sidiropoulos, N. D. and R. Bro. 2000. "On the uniqueness of multilinear decomposition of N-way arrays". *Journal of Chemometrics*. 14(3): 229–239.

Sidiropoulos, N. D., R. Bro, and G. B. Giannakis. 2000. "Parallel factor analysis in sensor array processing". *Signal Processing, IEEE Transactions on*. 48(8): 2377–2388.

Sidiropoulos, N. D., L. De Lathauwer, X. Fu, K. Huang, E. E. Papalexakis, and C. Faloutsos. 2017. "Tensor decomposition for signal processing and machine learning". *IEEE Transactions on Signal Processing*. 65(13): 3551–3582.

Spearman, C. 1904. "" General Intelligence," Objectively Determined and Measured". *The American Journal of Psychology*. 15(2): 201–292.

Sriperumbudur, B., K. Fukumizu, R. Kumar, A. Gretton, and A. Hyvärinen. 2013. "Density estimation in infinite dimensional exponential families". *arXiv preprint arXiv:1312.3516*.

stewart, G. and J.-G. Sun. 1990. *Matrix perturbation theory*. Academic Press.

Swersky, K., D. Buchman, N. D. Freitas, B. M. Marlin, *et al.* 2011. "On autoencoders and score matching for energy based models". In: *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*. 1201–1208.

Tai, C., T. Xiao, X. Wang, and W. E. 2016. "Convolutional neural networks with low-rank regularization". *ICLR*.

Tropp, J. A. 2012. "User-friendly tail bounds for sums of random matrices". *Foundations of computational mathematics*. 12(4): 389–434.

Wainwright, M. and M. Jordan. 2008. "Graphical models, exponential families, and variational inference". *Foundations and Trends® in Machine Learning*. 1(1-2): 1–305.

Walt, S. van der, S. C. Colbert, and G. Varoquaux. 2011. "The NumPy Array: A Structure for Efficient Numerical Computation". *Computing in Science Engineering*. 13(2): 22–30.

Wang, W., J. Wang, and N. Srebro. 2016. "Globally convergent stochastic optimization for canonical correlation analysis". *Advances in Neural Information Processing Systems*.

Wedin, P. 1972. "Perturbation bounds in connection with singular value decomposition". *BIT Numerical Mathematics*. 12(1): 99–111.

Weyl, H. 1912. "Das asymptotische Verteilungsgesetz der Eigenwerte linearer partieller Differentialgleichungen (mit einer Anwendung auf die Theorie der Hohlraumstrahlung)". *Mathematische Annalen*. 71(4): 441–479.

Zhang, T. and G. Golub. 2001. "Rank-one approximation to high order tensors". *SIAM Journal on Matrix Analysis and Applications*. 23: 534–550.