Dictionary learning and tensor decomposition via the sum-of-squares method

Boaz Barak
MSR

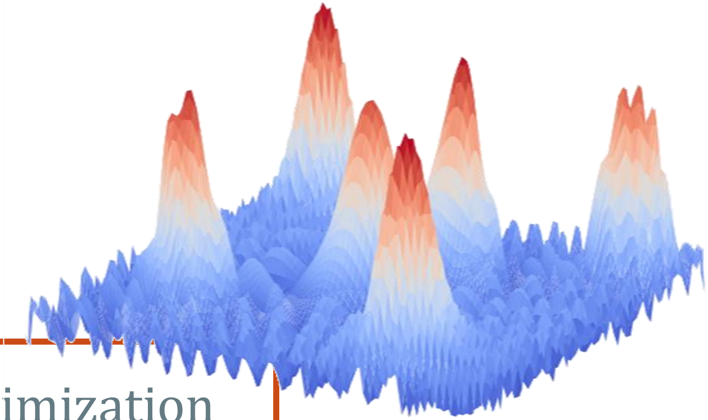
Jonathan Kelner
MIT

David Steurer
Cornell

STOC – Portland, June 2015

results: overview

sum-of-squares method (based on semidefinite programming) [Shor, Parrilo, Lasserre]



efficient algorithm to solve polynomial optimization problems that have ***only few global optima***

running time $n^{O(\log \#\text{solutions})}$
(quasi-poly time for poly #solutions)

also need ***short sum-of-squares certificate*** for this fact

\# **bad local optima** can be exponential → local-search algorithms fail

applications: *unsupervised learning problems tend to have this property*

identifiability: data uniquely determines parameters of model

our work: notion of ***constructive identifiability proofs*** that implies *efficient inference algorithms*

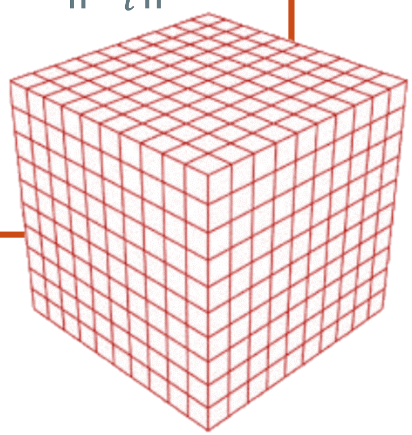
results: *tensor decomposition*

for all constants $\sigma \geq 1$ and $\varepsilon > 0$, exist constants $d \geq 1$ and $\tau > 0$

given tensor $T \in \mathbb{R}^{n^d}$ of the form $T = \sum_{i=1}^{m} a_i^{\otimes d} + Z$ with $\|a_i\| = 1$

can recover set $\approx_\varepsilon \{\pm a_1, \dots, \pm a_m\}$ in time $n^{O(\log n)}$,

whenever $\left\|\sum_i a_i a_i^\top\right\|_{\text{spectral}} \leq \sigma$ and $\|Z\|_{\text{spectral}} \leq \tau$



comparison to previous algorithms [Jennrich'70, Bhaskara-Charikar-Moitra-Vijayaraghavan'13, Anandkumar-Ge-Hsu-Kakade'12]

pros:

- **tolerate constant spectral error** (*before:* inverse polynomial error)
- no restrictions on vectors (*before:* incoherence or similar)

cons:

- running time (***but:*** techniques help for faster alg's [Hopkins-Schramm-Shi-S.'15+])
- only constant accuracy (***but:*** could combine with local search)

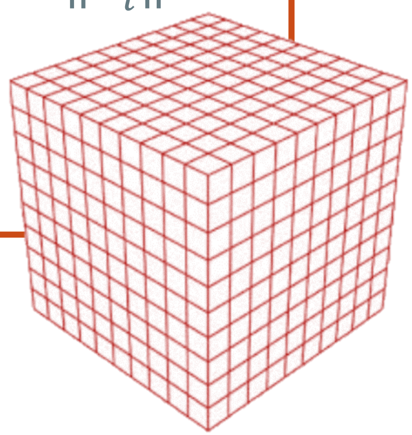
results: ***tensor decomposition***

for all constants $\sigma \geq 1$ and $\varepsilon > 0$, exist constants $d \geq 1$ and $\tau > 0$

given tensor $T \in \mathbb{R}^{n^d}$ of the form $T = \sum_{i=1}^m a_i^{\otimes d} + Z$ with $\|a_i\| = 1$

can recover set $\approx_\varepsilon \{\pm a_1, \dots, \pm a_m\}$ in time $n^{O(\log n)}$,

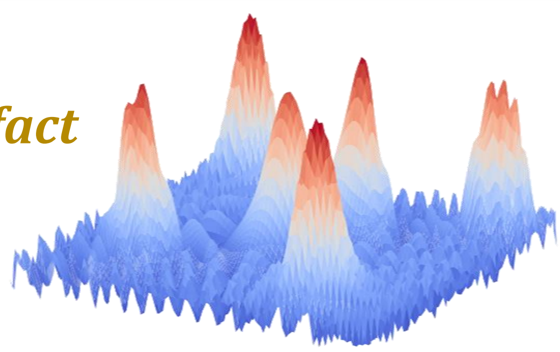
whenever $\left\|\sum_i a_i a_i^\top\right\|_{\text{spectral}} \leq \sigma$ and $\|Z\|_{\text{spectral}} \leq \tau$



connection to polynomial optimization

global optima of polynomial $\langle T, x^{\otimes d}\rangle = \sum_{i=1}^m \langle a_i, x\rangle^d + \langle Z, x^{\otimes d}\rangle$
over unit sphere $\approx_\varepsilon \{\pm a_1, \dots, \pm a_m\}$

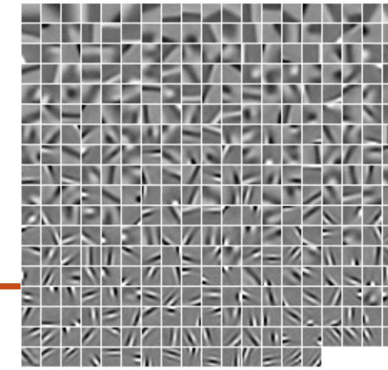
also:* $\exists$ *short sum-of-squares certificate for this fact



but: local behavior controlled by error Z
→ local search algorithms fail
(also *simultaneous diagonalization* fails)

results: *dictionary learning (aka sparse coding)*

application: machine learning (*feature extraction*)
neuroscience (*model for visual cortex*)



example: dictionary for natural images
[Olshausen-Fields'96]

data vectors

linear transformation "dictionary"

sparse vectors

y_1 y_T a_1 a_m x_1 x_T

$a_1, \ldots, a_m$ unknown unit vectors in isotropic position
$x_1, \ldots, x_t$ are i.i.d. samples from unknown "nice" distr. over ***sparse*** vectors
(only small correlations between coord's)

goal: given data vectors $y_1, \ldots, y_T$, reconstruct A

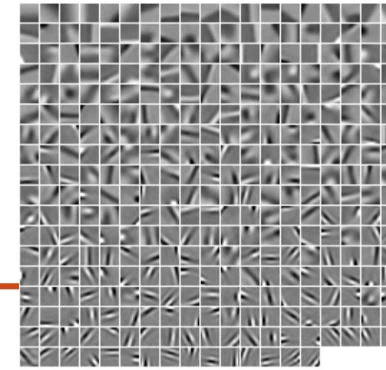
reduces to tensor decomposition with ***spectral error*** controlled by ***sparsity***

[Arora-Ge-Moitra, Agarwal-Anandkumar-Jain-Netrapalli-Tandon]

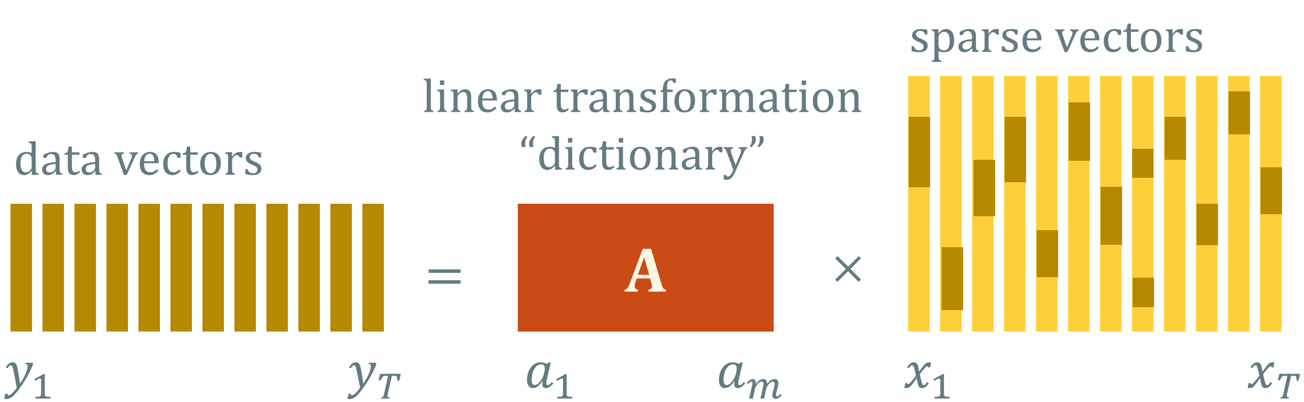
previous methods (local search): only very sparse vectors, up to $\sqrt{n}$ non-zeros

results: ***dictionary learning (aka sparse coding)***

application: machine learning (*feature extraction*)
neuroscience (*model for visual cortex*)



example: dictionary for natural images [Olshausen-Fields'96]



$a_1, \dots, a_m$ unknown unit vectors in isotropic position

$x_1, \dots, x_t$ are i.i.d. samples from unknown "nice" distr. over ***sparse*** vectors

(only small correlations between coord's)

goal: given data vectors $y_1, \dots, y_T$, reconstruct A

[Arora-Ge-Moitra, Agarwal-Anandkumar-Jain-Netrapalli-Tandon]

previous methods (local search): only very sparse vectors, up to $\sqrt{n}$ non-zeros

[this work]

sum-of-squares method: full sparsity range, up to constant fraction non-zeros

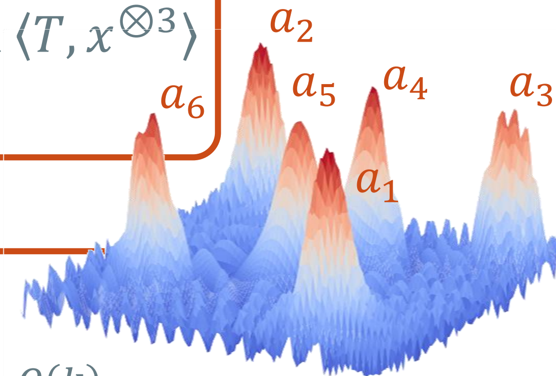
(quasipolynomial-time for sparsity $o(1)$; polynomial-time for $n^{-\varepsilon}$)

simplified problem

$a_1, \dots, a_n \in \mathbb{R}^n$ orthonormal, $Z \in \mathbb{R}^{n^3}$ with $\|Z\|_{\text{spectral}} \le \varepsilon$

given tensor $T = Z + \sum_i a_i^{\otimes 3}$, maximize polynomial $\langle T, x^{\otimes 3}\rangle$ over unit sphere $S^{n-1} \subseteq \mathbb{R}^n$

$\langle T, x^{\otimes 3}\rangle \approx_\varepsilon \max_i \langle a_i, x\rangle$



deg-k sum-of-squares algorithm

computes "pseudo distribution $D\colon S^{n-1} \to \mathbb{R}$" in time $n^{O(k)}$

behaves like ***deg-k part of density of distribution*** supported on solutions to $\mathcal{C} = \left\{\langle T, x^{\otimes 3}\rangle \ge 1 - \varepsilon, \|x\|^2 = 1\right\}$

i.e., D passes all tests derivable from $\mathcal{C}$ by deg-k SOS proof system

concretely, $\int_{S^{n-1}} D \cdot \left[P^2 \cdot \left(\langle T, x^{\otimes 3}\rangle - (1-\varepsilon)\right) + Q^2\right] \ge 0$ whenever $\deg P, \deg Q \le k$

want: rounding algorithm

given pseudo-distribution D, compute solution to constraints $\mathcal{C}$

approach: [Barak-Kelner-S.'14]

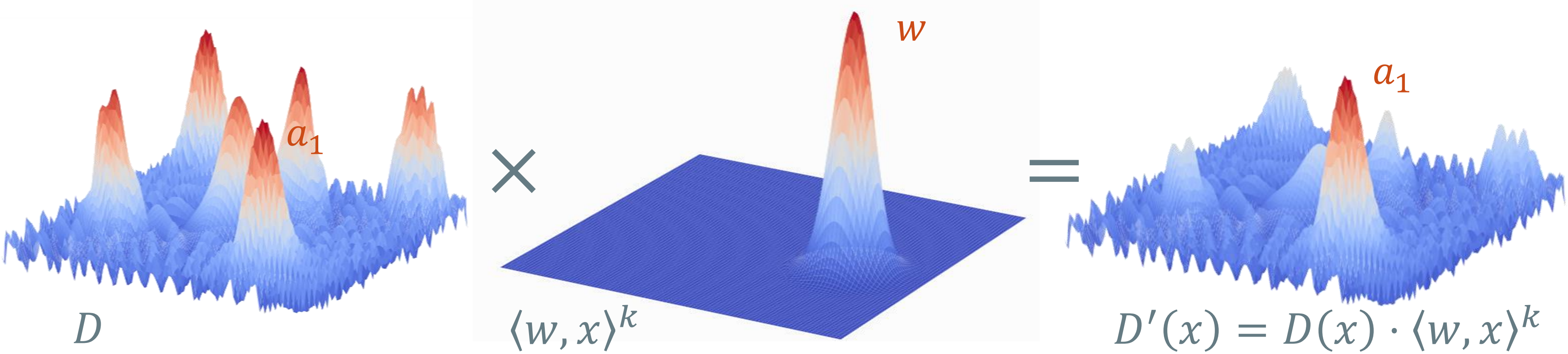
first analyze algorithm when D is deg-d part of actual distribution *simplified problem*

$a_1, \dots, a_n \in \mathbb{R}^n$ orthonormal, $Z \in \mathbb{R}^{n^3}$ with $\|Z\|_{\text{spectral}} \leq \varepsilon$

given tensor $T = Z + \sum_i a_i^{\otimes 3}$, solve $\mathcal{C} = \{\langle T, x^{\otimes 3}\rangle \geq 1 - \varepsilon, \|x\|^2 = 1\}$

assume: D is deg-k part of density supported on solutions to $\mathcal{C}$

algorithm: (1) reweigh D by $\langle w, x\rangle^k$ for $k \approx \log n$ and random unit vector w
(2) output top eigenvector of resulting covariance matrix



analysis

property of Gaussian distribution: with probability $\geq 1/n^{O(1)}$

$$\langle w, a_1\rangle^2 \geq 2 \cdot \max_{i>1} \langle w, a_i\rangle^2$$

→ increase probability mass on a_1 by factor 2^k relative to other spikes

→ for $k = \log n$, almost all mass on a_1 → can recover a_1 from covar. matrix

simplified problem

$a_1, \dots, a_n \in \mathbb{R}^n$ orthonormal, $Z \in \mathbb{R}^{n^3}$ with $\|Z\|_{\text{spectral}} \le \varepsilon$

given tensor $T = Z + \sum_i a_i^{\otimes 3}$, solve $\mathcal{C} = \left\{\langle T, x^{\otimes 3}\rangle \ge 1 - \varepsilon, \|x\|^2 = 1\right\}$

what does it mean to efficiently certify that $\mathcal{C}$ has only few solutions?

derive inequality $\sum_i \langle a_i, x\rangle^k \ge (1 - 2\varepsilon)^k$ for $k = \log n$
from constraints $\mathcal{C}$ in **deg-k SOS proof system**

soft-max
$\left(\sum_i y_i^k\right)^{1/k} \approx \max_i y_i$

derivation sketch

from $\left\{\begin{matrix} \|x\|^2 = 1 \\ \langle T, x^{\otimes 3}\rangle \ge 1 - \varepsilon \end{matrix}\right\}$ **derive** $\sum_i \langle a_i, x\rangle^3 \ge 1 - 2\varepsilon$

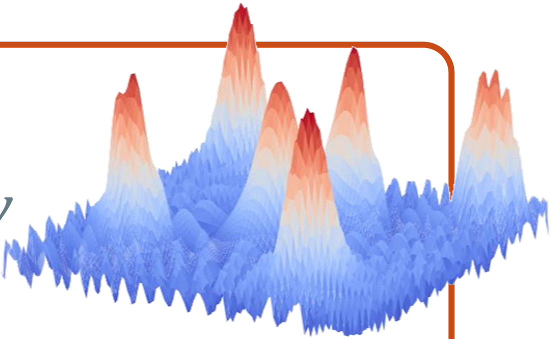
using $\left\|T - \sum_i a_i^{\otimes d}\right\|_{\text{spectral}} \le \varepsilon$ (SOS captures eigenvalue bounds)

from $\left\{\begin{matrix} \|x\|^2 = 1 \\ \sum_i \langle a_i, x\rangle^3 \ge 1 - 2\varepsilon \end{matrix}\right\}$ **derive** $\sum_i \langle a_i, x\rangle^k \ge (1 - 2\varepsilon)^k$ for all $k \ge d$

using $\left(\sum_i y_i^k\right) \cdot \left(\sum_i y_i^2\right)^k - \left(\sum_i y_i^3\right)^k$ is sum of squares,
choosing $y_i = \langle a_i, x\rangle$, and **using** $\sum_i \langle a_i, x\rangle^2 = \|x\|^2$

summary

polynomial optimization is easy if we can ***certify*** that there are ***only few good solutions***



(derive constraint of form $\sum_i \langle a_i, x\rangle^k$ for $k > \log \#\text{solutions}$)

open questions / subsequent work

sum of squares useful for other machine learning problems?

tensor prediction [Barak-Moitra]

overcomplete average-case 3-tensor decomposition [Ge-Ma]

can sum of squares lead to fast algorithms?

tensor principal component analysis [Hopkins-Shi-S.]

overcomplete average-case 3-tensor decomp. [Hopkins-Schramm-Shi-S.]

planted sparse vector [Hopkins-Schramm-Shi-S.]

Thank you!