Spectral Methods from Tensor Networks

Alex Wein
Courant Institute, NYU

Joint work with Ankur Moitra (MIT)

Outline

- Tensors

Outline

- Tensors
- Statistical problems involving tensors

Outline

- Tensors
- Statistical problems involving tensors
- A general framework for designing algorithms for tensor problems: “spectral methods from tensor networks”

Outline

- Tensors
- Statistical problems involving tensors
- A general framework for designing algorithms for tensor problems: "spectral methods from tensor networks"
- Orbit recovery: a certain class of tensor problems
  - Structured tensor decomposition

Outline

- Tensors
- Statistical problems involving tensors
- A general framework for designing algorithms for tensor problems: "spectral methods from tensor networks"
- Orbit recovery: a certain class of tensor problems
  - Structured tensor decomposition
- Main result: first polynomial-time algorithm for a certain orbit recovery problem

I. Tensors and Tensor Networks

What is a Tensor?

An order-p tensor is an $n_1 \times n_2 \times \cdots \times n_p$ multi-array:
$T = (T_{i_1,i_2,\ldots,i_p})$ with $i_j \in \{1, 2, \ldots, n_j\}$.

An order-1 tensor is a vector. An order-2 tensor is a matrix.

What is a Tensor?

An order-p tensor is an $n_1 \times n_2 \times \cdots \times n_p$ multi-array:
$T = (T_{i_1,i_2,\ldots,i_p})$ with $i_j \in \{1, 2, \ldots, n_j\}$.

An order-1 tensor is a vector. An order-2 tensor is a matrix.

T is symmetric if $n_1 = \cdots = n_p = n$ and $T_{i_1,\ldots,i_p} = T_{i_{\pi(1)},\ldots,i_{\pi(p)}}$ for any permutation π.

- In this talk, all tensors will be symmetric.

What is a Tensor?

An order-p tensor is an $n_1 \times n_2 \times \cdots \times n_p$ multi-array:
$T = (T_{i_1, i_2, \ldots, i_p})$ with $i_j \in \{1, 2, \ldots, n_j\}$.

An order-1 tensor is a vector. An order-2 tensor is a matrix.

T is symmetric if $n_1 = \cdots = n_p = n$ and $T_{i_1, \ldots, i_p} = T_{i_{\pi(1)}, \ldots, i_{\pi(p)}}$ for any permutation π.

- In this talk, all tensors will be symmetric.

Given p vectors $x_1, \ldots, x_p$, the rank-1 tensor $x_1 \otimes x_2 \otimes \cdots \otimes x_p$ has entries $(x_1 \otimes x_2 \otimes \cdots \otimes x_p)_{i_1, \ldots, i_p} = (x_1)_{i_1} (x_2)_{i_2} \cdots (x_p)_{i_p}$.

- Generalizes the rank-1 matrix $xy^\top$.
- Symmetric version: $x^{\otimes p} = x \otimes \cdots \otimes x \quad (p \text{ times})$.

Tensor Problems

Some statistical problems involving tensors:

Tensor Problems

Some statistical problems involving tensors:

- Tensor PCA / Spiked Tensor Model [RM'14, HSS'15]:

  Observe $T = \lambda x^{\otimes p} + Z$ where

  - $x \in \mathbb{R}^n$ is planted "signal" (norm 1)
  - $\lambda > 0$ is signal-to-noise parameter
  - Z is "noise" (i.i.d. Gaussian tensor)

  Goal: given T, recover x

  "Recover a rank-1 tensor buried in noise"

Tensor Problems

Some statistical problems involving tensors:

- Tensor PCA / Spiked Tensor Model [RM'14, HSS'15]:

  Observe $T = \lambda x^{\otimes p} + Z$ where
  - $x \in \mathbb{R}^n$ is planted "signal" (norm 1)
  - $\lambda > 0$ is signal-to-noise parameter
  - Z is "noise" (i.i.d. Gaussian tensor)

  Goal: given T, recover x

  "Recover a rank-1 tensor buried in noise"

- Tensor Decomposition [AGJ'14, BKS'15, GM'15, HSSS'16, MSS'16]:

  Observe $T = \sum_{i=1}^{r} x_i^{\otimes p}$ where $\{x_i\}$ are random vectors:
  - $x_i \sim \mathcal{N}(0, I_n)$

  Goal: given T, recover $\{x_1, \ldots, x_r\}$

  "Recover the components of a rank-r tensor"

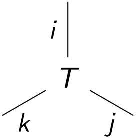
Tensor Network Notation

A graphical representation for tensors (used in e.g., quantum)

Tensor Network Notation

A graphical representation for tensors (used in e.g., quantum)

An order-p tensor has p "legs", one for each index:

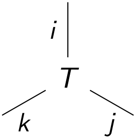


$$\Leftrightarrow \qquad T = (T_{i,j,k})$$

Tensor Network Notation

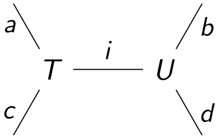
A graphical representation for tensors (used in e.g., quantum)

An order-p tensor has p "legs", one for each index:



$\Leftrightarrow$ $T = (T_{i,j,k})$

Two (or more) tensors can be attached by contracting indices:



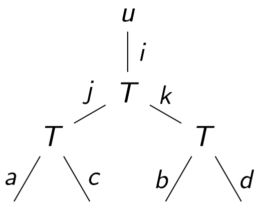
$\Leftrightarrow$

$$B = (B_{a,b,c,d})$$
$$B_{a,b,c,d} = \sum_i T_{a,c,i}\, U_{b,d,i}$$

Rule: sum over "fully connected" indices (in this case, i)

More Examples

A bigger example:

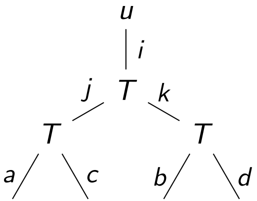


$\Leftrightarrow$

$$B = (B_{a,b,c,d})$$
$$B_{a,b,c,d} = \sum_{i,j,k} T_{a,c,j}\, T_{b,d,k}\, T_{i,j,k}\, u_i$$

More Examples

A bigger example:



$\Leftrightarrow$

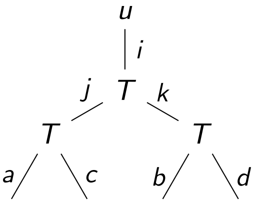
$$B = (B_{a,b,c,d})$$
$$B_{a,b,c,d} = \sum_{i,j,k} T_{a,c,j}\, T_{b,d,k}\, T_{i,j,k}\, u_i$$

This framework generalizes matrix/vector multiplication:

$$x - A - B - y \quad \Leftrightarrow \quad x^\top ABy$$

More Examples

A bigger example:



$\Leftrightarrow$

$$B = (B_{a,b,c,d})$$
$$B_{a,b,c,d} = \sum_{i,j,k} T_{a,c,j}\, T_{b,d,k}\, T_{i,j,k}\, u_i$$

This framework generalizes matrix/vector multiplication:

$$x - A - B - y \quad \Leftrightarrow \quad x^\top ABy$$

$$\sum_{ijk} x_i A_{ij} B_{jk} y_k$$

II. Spectral Methods from Tensor Networks

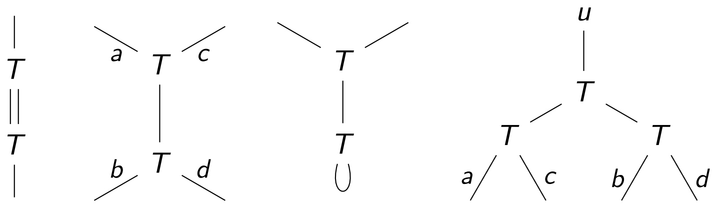
Spectral Methods from Tensor Networks

General framework for solving tensor problems:

1. Given input tensor T
2. Build a new tensor B by connecting copies of T in a tensor network
3. Flatten B to form a symmetric matrix M
   - E.g., the $(\{a,b\},\{c,d\})$-flattening of $B = (B_{a,b,c,d})$ is the $n^2 \times n^2$ matrix $M_{(a,b),(c,d)} = B_{a,b,c,d}$
4. Compute the leading eigenvector of M

Prior Work

Prior work has (implicitly) used this framework:



- [Richard–Montanari'14, Hopkins–Shi–Steurer'15] "Tensor unfolding"
- [Hopkins–Shi–Steurer'15] "Spectral SoS"
- [Hopkins–Schramm–Shi–Steurer'16] "Spectral SoS with partial trace"
- [Hopkins–Schramm–Shi–Steurer'16] "Spectral tensor decomposition"

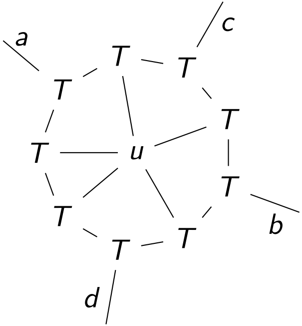
u is a random vector (to break symmetry).

Our Contribution

Our Contribution

We give the first polynomial-time algorithm for a particular tensor problem: heterogeneous continuous multi-reference alignment.

The algorithm is a spectral method based on this tensor network:



Smaller tensor networks fail for this problem.

General Analysis of Tensor Networks

General Analysis of Tensor Networks

Main step of analysis is to upper bound largest eigenvalue of a matrix built from a tensor network.

General Analysis of Tensor Networks

Main step of analysis is to upper bound largest eigenvalue of a matrix built from a tensor network.

Trace moment method: for a symmetric matrix M with eigenvalues $\{\lambda_i\}$ and $\lambda_{\max} = \max_i |\lambda_i|$,

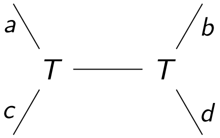
$$\mathrm{Tr}(M^{2k}) = \sum_i \lambda_i^{2k} \geq \lambda_{\max}^{2k}$$

so compute $\mathbb{E}[\mathrm{Tr}(M^{2k})]$ and apply Markov's inequality:

$$\mathbb{P}(\lambda_{\max} \geq t) = \mathbb{P}(\lambda_{\max}^{2k} \geq t^{2k}) \leq \frac{\mathbb{E}[\mathrm{Tr}(M^{2k})]}{t^{2k}}.$$

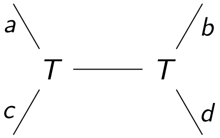
Trace Method for Tensor Networks

Example: T is an order-3 symmetric tensor with i.i.d. Rademacher (uniform ± 1) entries, and we want to compute $\mathbb{E}[\mathrm{Tr}(M^6)]$ where M is the $(\{a, b\}, \{c, d\})$-flattening of this tensor:



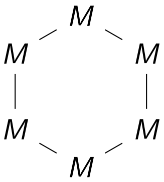
Trace Method for Tensor Networks

Example: T is an order-3 symmetric tensor with i.i.d. Rademacher (uniform ± 1) entries, and we want to compute $\mathbb{E}[\mathrm{Tr}(M^6)]$ where M is the $(\{a, b\}, \{c, d\})$-flattening of this tensor:



Note that

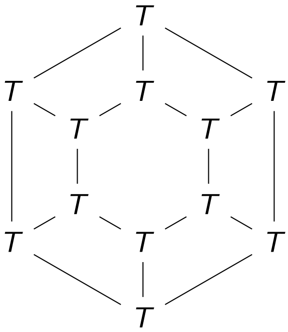
$\mathrm{Tr}(M^6)$ =



so plug in the definition of M...

Trace Method for Tensor Networks (Continued)

$\mathrm{Tr}(M^6) \quad =$

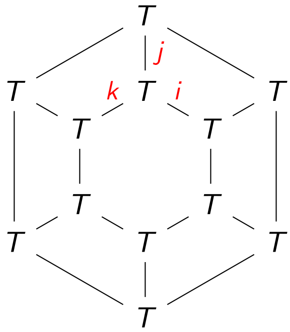


So the computation of $\mathbb{E}[\mathrm{Tr}(M^6)]$ is reduced to a combinatorial question about this diagram.

When T is i.i.d. Rademacher: $\mathbb{E}[\mathrm{Tr}(M^6)]$ is the number of ways to label the edges of the diagram with elements of $[n]$ such that each triple $\{i, j, k\}$ appears incident to an even number of T's.

Trace Method for Tensor Networks (Continued)

$\mathrm{Tr}(M^6) \quad =$

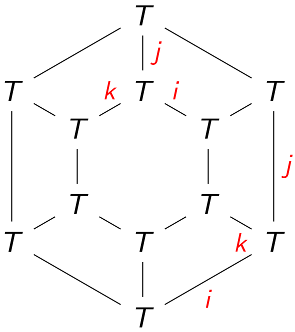


So the computation of $\mathbb{E}[\mathrm{Tr}(M^6)]$ is reduced to a combinatorial question about this diagram.

When T is i.i.d. Rademacher: $\mathbb{E}[\mathrm{Tr}(M^6)]$ is the number of ways to label the edges of the diagram with elements of $[n]$ such that each triple $\{i, j, k\}$ appears incident to an even number of T's.

Trace Method for Tensor Networks (Continued)

$\mathrm{Tr}(M^6) \quad = $



So the computation of $\mathbb{E}[\mathrm{Tr}(M^6)]$ is reduced to a combinatorial question about this diagram.

When T is i.i.d. Rademacher: $\mathbb{E}[\mathrm{Tr}(M^6)]$ is the number of ways to label the edges of the diagram with elements of $[n]$ such that each triple $\{i, j, k\}$ appears incident to an even number of T's.

III. Orbit Recovery Problems

Image Alignment

Given many noisy rotated copies of an image, recover the image.

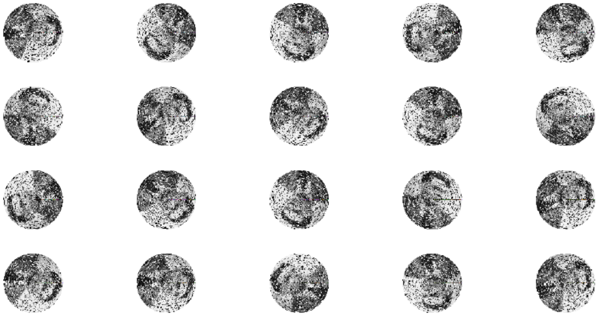


Image credit: [Bandeira, PhD thesis '15]

Image Alignment

Given many noisy rotated copies of an image, recover the image.

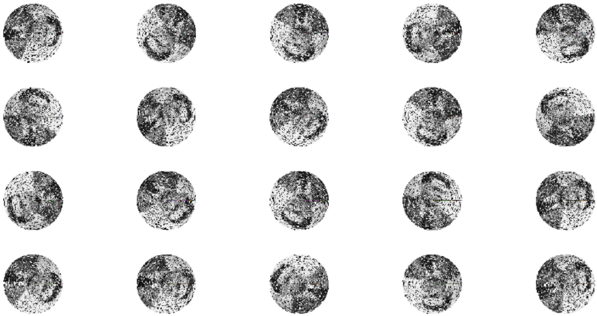


Image credit: [Bandeira, PhD thesis '15]

Application: cryo-EM (cryo-electron microscopy)

- Given many noisy pictures of a molecule taken from different unknown angles, recover the 3D structure of the molecule.

Orbit Recovery

Orbit Recovery Problem [APS17,BRW17,PWBRS17,BBKPW**W**17,APS18]:

Orbit Recovery

Orbit Recovery Problem [APS17,BRW17,PWBRS17,BBKPW**W**17,APS18]:

- Let $x \in \mathbb{R}^n$ be an unknown "signal" (e.g. the image)

Orbit Recovery

Orbit Recovery Problem [APS17,BRW17,PWBRS17,BBKPW**W**17,APS18]:

- Let $x \in \mathbb{R}^n$ be an unknown "signal" (e.g. the image)
- Let G be a compact group acting on $\mathbb{R}^n$ (e.g. rotations $\mathrm{SO}(2)$)

Orbit Recovery

Orbit Recovery Problem [APS17,BRW17,PWBRS17,BBKPW**W**17,APS18]:

- Let $x \in \mathbb{R}^n$ be an unknown "signal" (e.g. the image)
- Let G be a compact group acting on $\mathbb{R}^n$ (e.g. rotations $\mathrm{SO}(2)$)
- Observe samples $y_i = g_i \cdot x + z_i$ where $g_i \sim G$, $z_i \sim \mathcal{N}(0, I_n)$

Orbit Recovery

Orbit Recovery Problem [APS17,BRW17,PWBRS17,BBKPW**W**17,APS18]:

- Let $x \in \mathbb{R}^n$ be an unknown "signal" (e.g. the image)
- Let G be a compact group acting on $\mathbb{R}^n$ (e.g. rotations $\mathrm{SO}(2)$)
- Observe samples $y_i = g_i \cdot x + z_i$ where $g_i \sim G$, $z_i \sim \mathcal{N}(0, I_n)$
- Goal: recover the orbit of x (can't distinguish x from $g \cdot x$)

Orbit Recovery

Orbit Recovery Problem [APS17,BRW17,PWBRS17,BBKPW**W**17,APS18]:

- Let $x \in \mathbb{R}^n$ be an unknown "signal" (e.g. the image)
- Let G be a compact group acting on $\mathbb{R}^n$ (e.g. rotations $\mathrm{SO}(2)$)
- Observe samples $y_i = g_i \cdot x + z_i$ where $g_i \sim G$, $z_i \sim \mathcal{N}(0, I_n)$
- Goal: recover the orbit of x (can't distinguish x from $g \cdot x$)
- Heterogeneous: signals $x_1, \ldots, x_K$, samples $y_i = g_i \cdot x_{k_i} + z_i$

Orbit Recovery

Orbit Recovery Problem [APS17,BRW17,PWBRS17,BBKPW**W**17,APS18]:

- Let $x \in \mathbb{R}^n$ be an unknown "signal" (e.g. the image)
- Let G be a compact group acting on $\mathbb{R}^n$ (e.g. rotations $\mathrm{SO}(2)$)
- Observe samples $y_i = g_i \cdot x + z_i$ where $g_i \sim G$, $z_i \sim \mathcal{N}(0, I_n)$
- Goal: recover the orbit of x (can't distinguish x from $g \cdot x$)
- Heterogeneous: signals $x_1, \ldots, x_K$, samples $y_i = g_i \cdot x_{k_i} + z_i$

This paper: heterogeneous continuous multi-reference alignment

Orbit Recovery

Orbit Recovery Problem [APS17,BRW17,PWBRS17,BBKPW**W**17,APS18]:

- Let $x \in \mathbb{R}^n$ be an unknown "signal" (e.g. the image)
- Let G be a compact group acting on $\mathbb{R}^n$ (e.g. rotations $\mathrm{SO}(2)$)
- Observe samples $y_i = g_i \cdot x + z_i$ where $g_i \sim G$, $z_i \sim \mathcal{N}(0, I_n)$
- Goal: recover the orbit of x (can't distinguish x from $g \cdot x$)
- Heterogeneous: signals $x_1, \ldots, x_K$, samples $y_i = g_i \cdot x_{k_i} + z_i$

This paper: heterogeneous continuous multi-reference alignment

- Each signal x_k is a random real-valued (band-limited) function on the unit circle

Orbit Recovery

Orbit Recovery Problem [APS17,BRW17,PWBRS17,BBKPW**W**17,APS18]:

- Let $x \in \mathbb{R}^n$ be an unknown "signal" (e.g. the image)
- Let G be a compact group acting on $\mathbb{R}^n$ (e.g. rotations $\mathrm{SO}(2)$)
- Observe samples $y_i = g_i \cdot x + z_i$ where $g_i \sim G$, $z_i \sim \mathcal{N}(0, I_n)$
- Goal: recover the orbit of x (can't distinguish x from $g \cdot x$)
- Heterogeneous: signals $x_1, \ldots, x_K$, samples $y_i = g_i \cdot x_{k_i} + z_i$

This paper: heterogeneous continuous multi-reference alignment

- Each signal x_k is a random real-valued (band-limited) function on the unit circle
- $G = \mathrm{SO}(2)$ acting by rotation

Our algorithm

Method of moments: use samples to estimate 3rd moment tensor

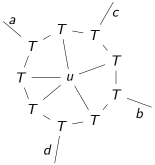
$$\mathbb{E}_i[y_i^{\otimes 3}] \quad \Rightarrow \quad T = \sum_{k=1}^{K} \int_{g \sim \mathrm{SO}(2)} (g \cdot x_k)^{\otimes 3}.$$

Our algorithm

Method of moments: use samples to estimate 3rd moment tensor

$$\mathbb{E}_i[y_i^{\otimes 3}] \quad \Rightarrow \quad T = \sum_{k=1}^{K} \int_{g \sim \mathrm{SO}(2)} (g \cdot x_k)^{\otimes 3}.$$

Plug T (and random u) into tensor network, and compute leading eigenvector:

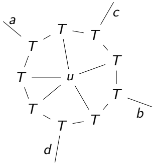


Our algorithm

Method of moments: use samples to estimate 3rd moment tensor

$$\mathbb{E}_i[y_i^{\otimes 3}] \quad \Rightarrow \quad T = \sum_{k=1}^{K} \int_{g \sim \mathrm{SO}(2)} (g \cdot x_k)^{\otimes 3}.$$

Plug T (and random u) into tensor network, and compute leading eigenvector:



Our algorithm gives:

- optimal sample complexity
- heterogeneity $K \leq n^{\delta}$ (optimal should be $n^{1/2}$)
- list recovery of $\{x_k\}$
- first solution to heterogeneous problem over infinite group

Summary

- General framework for designing spectral algorithms for tensor problems
- Tensor network notation makes general analysis tractable
- First polynomial-time algorithm for a certain continuous tensor decomposition problem (heterogeneous continuous MRA)
- Orbit recovery problems are in need of further theoretical study
  - All groups (especially infinite groups)
  - Optimal heterogeneity

Summary

- General framework for designing spectral algorithms for tensor problems
- Tensor network notation makes general analysis tractable
- First polynomial-time algorithm for a certain continuous tensor decomposition problem (heterogeneous continuous MRA)
- Orbit recovery problems are in need of further theoretical study
  - All groups (especially infinite groups)
  - Optimal heterogeneity

Thanks!