Orthogonal tensor decomposition

Daniel Hsu

Columbia University

Largely based on 2012 arXiv report “Tensor decompositions for learning latent variable models”, with Anandkumar, Ge, Kakade, and Telgarsky.

The basic decomposition problem

Notation: For a vector $\vec{x} = (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n$,

$$\vec{x} \otimes \vec{x} \otimes \vec{x}$$

denotes the 3-way array (call it a "tensor") in $\mathbb{R}^{n \times n \times n}$ whose $(i, j, k)^{\text{th}}$ entry is $x_i x_j x_k$.

Problem: Given $T \in \mathbb{R}^{n \times n \times n}$ with the promise that

$$T = \sum_{t=1}^{n} \lambda_t \, \vec{v}_t \otimes \vec{v}_t \otimes \vec{v}_t$$

for some orthonormal basis $\{\vec{v}_t\}$ of $\mathbb{R}^n$ (w.r.t. standard inner product) and positive scalars $\{\lambda_t > 0\}$, approximately find $\{(\vec{v}_t, \lambda_t)\}$ (up to some desired precision).

Basic questions

1. Is $\{(\vec{v}_t, \lambda_t)\}$ uniquely determined?
2. If so, is there an efficient algorithm for finding the decomposition?
3. What if T is perturbed by some small amount?

   Perturbed problem: Same as the original problem, except instead of T, we are given $T + E$ for some "error tensor" E.

   How "large" can E be if we want ε precision?

Analogous matrix problem

Matrix problem: Given $M \in \mathbb{R}^{n \times n}$ with the promise that

$$M = \sum_{t=1}^{n} \lambda_t \; \vec{v}_t \; \vec{v}_t^{\top}$$

for some orthonormal basis $\{\vec{v}_t\}$ of $\mathbb{R}^n$ (w.r.t. standard inner product) and positive scalars $\{\lambda_t > 0\}$, approximately find $\{(\vec{v}_t, \lambda_t)\}$ (up to some desired precision).

Analogous matrix problem

- We're promised that M is symmetric and positive definite, so requested decomposition is an **eigendecomposition.**

  In this case, an eigendecomposition **always exists**, and **can be found efficiently.**

  It is **unique** if and only if the $\{\lambda_i\}$ are distinct.

- What if M is perturbed by some small amount?

Perturbed matrix problem: Same as the original problem, except instead of M, we are given $M + E$ for some "error matrix" E (assume to be symmetric).

  Answer provided by **matrix perturbation theory** (*e.g.*, Davis-Kahan), which requires $\|E\|_2 < \min_{i \neq j} |\lambda_i - \lambda_j|$.

Back to the original problem

Problem: Given $T \in \mathbb{R}^{n \times n \times n}$ with the promise that

$$T = \sum_{t=1}^{n} \lambda_t \, \vec{v}_t \otimes \vec{v}_t \otimes \vec{v}_t$$

for some orthonormal basis $\{\vec{v}_t\}$ of $\mathbb{R}^n$ (w.r.t. standard inner product) and positive scalars $\{\lambda_t > 0\}$, approximately find $\{(\vec{v}_t, \lambda_t)\}$ (up to some desired precision).

Such decompositions **do not necessarily exist**, even for symmetric tensors.

Where the decompositions do exist, the Perturbed problem asks if they are "robust".

Main ideas

Easy claim: Repeated application of a certain quadratic operator based on T (a "power iteration") recovers a single $(\vec{v}_t, \lambda_t)$ up to any desired precision.

Self-reduction: Replace T with $T - \lambda_t \, \vec{v}_t \otimes \vec{v}_t \otimes \vec{v}_t$.

- Why?: $T - \lambda_t \, \vec{v}_t \otimes \vec{v}_t \otimes \vec{v}_t \; = \; \sum_{\tau \neq t} \lambda_\tau \, \vec{v}_\tau \otimes \vec{v}_\tau \otimes \vec{v}_\tau$.
- Catch: We don't recover $(\vec{v}_t, \lambda_t)$ exactly, so we actually can only replace T with

$$T - \lambda_t \, \vec{v}_t \otimes \vec{v}_t \otimes \vec{v}_t + E_t$$

  for some "error tensor" E_t.
- Therefore, must anyway deal with perturbations.

Rest of this talk

1. Identifiability of decomposition $\{(\vec{v}_t, \lambda_t)\}$ from T.
2. A decomposition algorithm based on tensor power iteration.
3. Error analysis of decomposition algorithm.

Identifiability of the decomposition

Orthonormal basis $\{\vec{v}_t\}$ of $\mathbb{R}^n$, positive scalars $\{\lambda_t > 0\}$:

$$T = \sum_{t=1}^{n} \lambda_t \, \vec{v}_t \otimes \vec{v}_t \otimes \vec{v}_t$$

In what sense is $\{(\vec{v}_t, \lambda_t)\}$ uniquely determined?

Claim: $\{\vec{v}_t\}$ are the n isolated local maximizers of certain cubic form $f_T : \mathbb{B}^n \to \mathbb{R}$, and $f_T(\vec{v}_t) = \lambda_t$.

Aside: multilinear form

There is a natural trilinear form associated with T:

$$(\vec{x}, \vec{y}, \vec{z}) \mapsto \sum_{i,j,k} T_{i,j,k}\, x_i y_j z_k.$$

For matrices M, it looks like

$$(\vec{x}, \vec{y}) \mapsto \sum_{i,j} M_{i,j}\, x_i y_j = \vec{x}^\top M \vec{y}.$$

Review: Rayleigh quotient

Recall Rayleigh quotient for matrix $M := \sum_{t=1}^{n} \lambda_t \ \vec{v}_t \vec{v}_t^{\top}$ (assuming $\vec{x} \in \mathbb{S}^{n-1}$):

$$R_M(\vec{x}) \ := \ \vec{x}^{\top} M \vec{x} \ = \ \sum_{t=1}^{n} \lambda_t \ (\vec{v}_t^{\top} \vec{x})^2 .$$

Every $\vec{v}_t$ such that $|\lambda_t| = \max!$ is a maximizer of R_M.

(These are also the only local maximizers.)

The natural cubic form

Consider the function $f_T : \mathbb{B}^n \to \mathbb{R}$ given by

$$\vec{x} \;\mapsto\; f_T(\vec{x}) = \sum_{i,j,k} T_{i,j,k}\, x_i x_j x_k .$$

For our promised $T = \sum_{t=1}^{n} \lambda_t\; \vec{v}_t \otimes \vec{v}_t \otimes \vec{v}_t$, f_T becomes

$$\begin{aligned} f_T(\vec{x}) &= \sum_{t=1}^{n} \lambda_t \sum_{i,j,k} \left(\vec{v}_t \otimes \vec{v}_t \otimes \vec{v}_t\right)_{i,j,k} x_i x_j x_k \\ &= \sum_{t=1}^{n} \lambda_t \sum_{i,j,k} (\vec{v}_t)_i (\vec{v}_t)_j (\vec{v}_t)_k x_i x_j x_k \\ &= \sum_{t=1}^{n} \lambda_t\, (\vec{v}_t^{\top} \vec{x})^3 . \end{aligned}$$

Observation: $f_T(\vec{v}_t) = \lambda_t$.

Variational characterization

Claim: Isolated local maximizers of f_T on $\mathbb{B}^n$ are $\{\vec{v}_t\}$.

Objective function (with constraint):

$$\vec{x} \mapsto \inf_{\lambda \geq 0} \sum_{t=1}^{n} \lambda_t \, (\vec{v}_t^{\top} \vec{x})^3 - 1.5\lambda(\|\vec{x}\|_2^2 - 1).$$

First-order condition for local maxima:

$$\sum_{t=1}^{n} \lambda_t \, (\vec{v}_t^{\top} \vec{x})^2 \; \vec{v}_t = \lambda \vec{x}.$$

Second-order condition for isolated local maxima:

$$\vec{w}^{\top} \left(2 \sum_{t=1}^{n} \lambda_t \, (\vec{v}_t^{\top} \vec{x}) \vec{v}_t \vec{v}_t^{\top} - \lambda I \right) \vec{w} < 0, \qquad \vec{w} \perp \vec{x}.$$

Intuition behind variational characterization

May as well assume $\vec{v}_t$ is t^{th} coordinate basis vector, so

$$\max_{\vec{x}\in\mathbb{R}^n} f_T(\vec{x}) = \sum_{t=1}^{n} \lambda_t \, x_t^3 \quad \text{s.t.} \quad \sum_{t=1}^{n} x_t^2 \leq 1.$$

Intuition: Suppose $\mathrm{supp}(\vec{x}) = \{1, 2\}$, and $x_1, x_2 > 0$.

$$f_T(\vec{x}) = \lambda_1 x_1^3 + \lambda_2 x_2^3 < \lambda_1 x_1^2 + \lambda_2 x_2^2 \leq \max\{\lambda_1, \ \lambda_2\}.$$

Better to have $|\mathrm{supp}(\vec{x})| = 1$, *i.e.*, picking $\vec{x}$ to be a coordinate basis vector. ∎

Aside: canonical polyadic decomposition

Rank-K **canonical polyadic decomposition** (CPD) of T
(also called PARAFAC, CANDECOMP, or CP):

$$T = \sum_{i=1}^{K} \sigma_i \, \vec{u}_i \otimes \vec{v}_i \otimes \vec{w}_i .$$

[Harshman/Jennrich, 1970; Kruskal, 1977; Leurgans et al., 1993].

Number of parameters: $K \cdot (3n + 1)$ (compared to n^3 in general).

Fact: Our promised T has a rank-n CPD.

N.B.: *Overcomplete* ($K > n$) CPD is also interesting *and a possibility* as long as $K(3n + 1) \ll n^3$.

The quadratic operator

Easy claim: Repeated application of a certain quadratic operator (based on T) recovers a single $(\lambda_t, \vec{v}_t)$ up to any desired precision.

For any $T \in \mathbb{R}^{n \times n \times n}$ and $\vec{x} = (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n$, define the quadratic operator

$$\phi_T(\vec{x}) := \sum_{i,j,k} T_{i,j,k}\, x_j x_k\, \vec{e}_i \ \in \mathbb{R}^n$$

where $\vec{e}_i \in \mathbb{R}^n$ is the i^{th} coordinate basis vector.

$$\text{If } T = \sum_{t=1}^{n} \lambda_t\, \vec{v}_t \otimes \vec{v}_t \otimes \vec{v}_t\,, \ \text{ then } \phi_T(\vec{x}) = \sum_{t=1}^{n} \lambda_t\, \left(\vec{v}_t^{\top} \vec{x}\right)^2 \vec{v}_t.$$

An algorithm?

Recall: First-order condition for local maxima of $f_T(\vec{x}) = \sum_{t=1}^{n} \lambda_t \, (\vec{v}_t{}^\top \vec{x})^3$ for $\vec{x} \in \mathbb{B}^n$:

$$\phi_T(\vec{x}) = \sum_{t=1}^{n} \lambda_t \, (\vec{v}_t{}^\top \vec{x})^2 \; \vec{v}_t = \lambda \vec{x}$$

i.e., "eigenvector"-like condition.

Algorithm: Find $\vec{x} \in \mathbb{B}^n$ fixed under $\vec{x} \mapsto \phi_T(\vec{x}) / \|\phi_T(\vec{x})\|$.

(Ignoring numerical issues, can just repeatedly apply ϕ_T and defer normalization until later.)

N.B.: Gradient ascent also works [Kolda & Mayo, '11].

Tensor power iteration

[De Lathauwer *et al*, 2000]
Start with some $\vec{x}^{(0)}$, and for $j = 1, 2, \ldots$:

$$\vec{x}^{(j)} \; := \; \phi_T\left(\vec{x}^{(j-1)}\right) \; = \; \sum_{t=1}^{n} \lambda_t \left(\vec{v}_t^{\top} \vec{x}^{(j-1)}\right)^2 \vec{v}_t .$$

Claim: For almost all initial $\vec{x}^{(0)}$, the sequence $\left(\vec{x}^{(j)} / \|\vec{x}^{(j)}\|\right)_{j=1}^{\infty}$ converges *quadratically fast* to some $\vec{v}_t$.

Review: matrix power iteration

Recall matrix power iteration for matrix $M := \sum_{t=1}^{n} \lambda_t \; \vec{v}_t \vec{v}_t^{\top}$:

Start with some $\vec{x}^{(0)}$, and for $j = 1, 2, \ldots$:

$$\vec{x}^{(i)} \;:=\; M\vec{x}^{(j-1)} \;=\; \sum_{t=1}^{n} \lambda_t \left(\vec{v}_t^{\top} \vec{x}^{(j-1)}\right) \vec{v}_t .$$

i.e., component in $\vec{v}_t$ direction is scaled by λ_t.

If $\lambda_1 > \lambda_2 \geq \cdots$, then

$$\frac{\left(\vec{v}_1^{\top} \vec{x}^{(j)}\right)^2}{\sum_{t=1}^{n} \left(\vec{v}_t^{\top} \vec{x}^{(j)}\right)^2} \geq 1 - k\left(\frac{\lambda_2}{\lambda_1}\right)^{2j} .$$

i.e., converges *linearly* to $\vec{v}_1$ (assuming gap $\lambda_2/\lambda_1 < 1$).

Tensor power iteration convergence analysis

Let $c_t := \vec{v}_t^{\top} \vec{x}^{(0)}$ (initial component in $\vec{v}_t$ direction); assume WLOG

$$\lambda_1 |c_1| > \lambda_2 |c_2| \geq \lambda_3 |c_3| \geq \cdots .$$

Then

$$\vec{x}^{(1)} = \sum_{t=1}^{n} \lambda_t \left(\vec{v}_t^{\top} \vec{x}^{(0)} \right)^2 \vec{v}_t = \sum_{t=1}^{n} \lambda_t c_t^2 \ \vec{v}_t$$

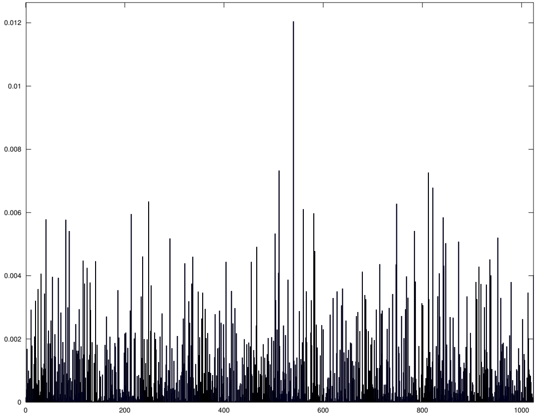
i.e., component in $\vec{v}_t$ direction is squared then scaled by λ_t.

Easy to show

$$\frac{\left(\vec{v}_1^{\top} \vec{x}^{(j)} \right)^2}{\sum_{t=1}^{n} \left(\vec{v}_t^{\top} \vec{x}^{(j)} \right)^2} \geq 1 - k \left(\frac{\lambda_1}{\max_{t \neq 1} \lambda_t} \right)^2 \left| \frac{\lambda_2 c_2}{\lambda_1 c_1} \right|^{2^{j+1}} .$$

Example

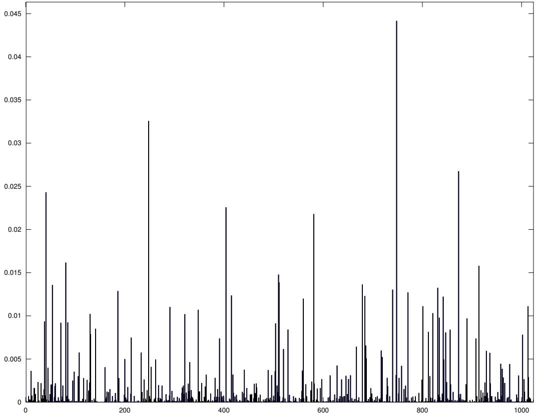
$$n = 1024,\ \lambda_t \sim_{\text{u.a.r.}} [0, 1].$$



Value of $(\vec{v}_t^{\top} \vec{x}^{(0)})^2$ for $t = 1, 2, \ldots, 1024$

Example

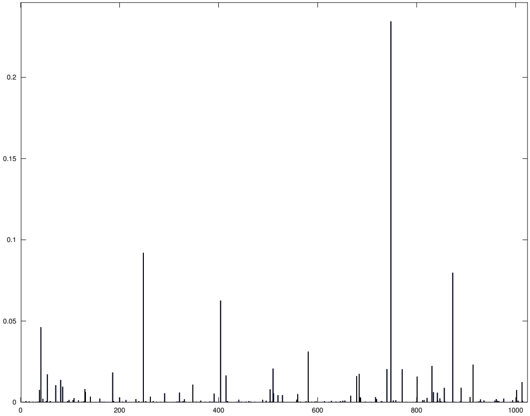
$$n = 1024,\ \lambda_t \sim_{\text{u.a.r.}} [0, 1].$$



Value of $(\vec{v}_t^{\top}\vec{x}^{(1)})^2$ for $t = 1, 2, \ldots, 1024$

Example

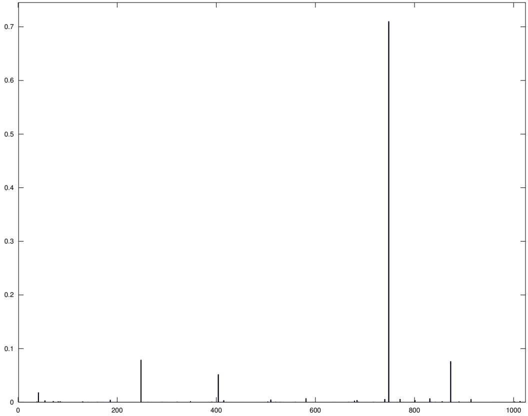
$$n = 1024,\ \lambda_t \sim_{\text{u.a.r.}} [0, 1].$$



Value of $(\vec{v}_t^{\top} \vec{x}^{(2)})^2$ for $t = 1, 2, \ldots, 1024$

Example

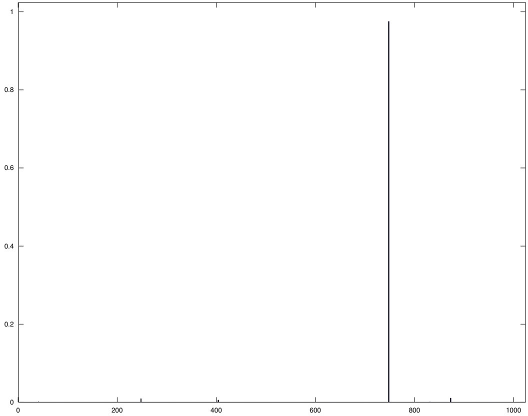
$$n = 1024,\ \lambda_t \sim_{\text{u.a.r.}} [0, 1].$$



Value of $(\vec{v}_t^{\top}\vec{x}^{(3)})^2$ for $t = 1, 2, \ldots, 1024$

Example

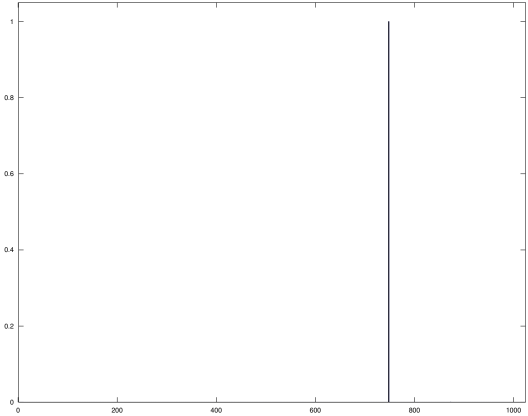
$$n = 1024,\ \lambda_t \sim_{\text{u.a.r.}} [0, 1].$$



Value of $(\vec{v}_t{}^{\top} \vec{x}^{(4)})^2$ for $t = 1, 2, \ldots, 1024$

Example

$$n = 1024,\ \lambda_t \sim_{\text{u.a.r.}} [0, 1].$$



Value of $(\vec{v}_t^{\top} \vec{x}^{(5)})^2$ for $t = 1, 2, \ldots, 1024$

Matrix vs. tensor power iteration

Matrix power iteration:

1. Requires gap between largest and second-largest λ_t.
   (Property of the matrix only.)

2. Converges to top $\vec{v}_t$.

3. Linear convergence. (Need $O(\log(1/\epsilon))$ iterations.)

Tensor power iteration:

1. Requires gap between largest and second-largest $\lambda_t |c_t|$.
   (Property of the tensor and initialization $\vec{x}^{(0)}$.)

2. Converges to $\vec{v}_t$ for which $\lambda_t |c_t| = \max$! (could be any of them).

3. Quadratic convergence. (Need $O(\log\log(1/\epsilon))$ iterations.)

Initialization of tensor power iteration

Convergence of tensor power iteration requires **gap** between **largest** and **second-largest** $\lambda_t |\vec{v}_t^\top \vec{x}^{(0)}|$.

Example of bad initialization: Suppose $T = \sum_t \vec{v}_t \otimes \vec{v}_t \otimes \vec{v}_t$, and $\vec{x}^{(0)} = \frac{1}{\sqrt{2}}(\vec{v}_1 + \vec{v}_2)$.

$$\begin{aligned} \phi_T(\vec{x}^{(0)}) &= (\vec{v}_1^\top \vec{x}^{(0)})^2 \vec{v}_1 + (\vec{v}_2^\top \vec{x}^{(0)})^2 \vec{v}_2 \\ &= \frac{1}{2}(\vec{v}_1 + \vec{v}_2) = \frac{1}{\sqrt{2}} \vec{x}^{(0)}. \end{aligned}$$

Fortunately, bad initialization points are atypical.

Full decomposition algorithm

Input: $T \in \mathbb{R}^{n \times n \times n}$.

Initialize: $\widetilde{T} := T$.

For $i = 1, 2, \ldots, n$:

1. Pick $\vec{x}^{(0)} \in \mathbb{S}^{n-1}$ unif. at random.
2. Run tensor power iteration with $\widetilde{T}$ starting from $\vec{x}^{(0)}$ for N iterations.
3. Set $\hat{v}_i := \vec{x}^{(N)} / \|\vec{x}^{(N)}\|$ and $\hat{\lambda}_i := f_{\widetilde{T}}(\hat{v}_i)$.
4. Replace $\widetilde{T} := \widetilde{T} - \hat{\lambda}_i \, \hat{v}_i \otimes \hat{v}_i \otimes \hat{v}_i$.

Output: $\{(\hat{v}_i, \hat{\lambda}_i) : i \in [n]\}$.

Actually: repeat Steps 1–3 several times, and take results of trial yielding largest $\hat{\lambda}_i$.

Aside: direct minimization

Can also consider directly minimizing

$$\left\| T - \sum_{t=1}^{n} \hat{\lambda}_t \ \hat{v}_t \otimes \hat{v}_t \otimes \hat{v}_t \right\|_F^2$$

via local optimization (*e.g.*, coord. descent, alternating least squares).

Decomposition algorithm via tensor power iteration can be viewed as **orthgonal greedy algorithm** for minimizing above objective [Zhang & Golub, '01].

Aside: implementation for bag-of-words models

Let $\vec{f}^{(i)}$ be empirical word frequency vector for document i:

$$(\vec{f}^{(i)})_j = \frac{\text{\# times word } j \text{ appears in document } i}{\text{length of document } i}$$

Matrix of word-pair frequencies (from m documents)

$$\widehat{\text{Pairs}} \approx \frac{1}{m}\sum_{i=1}^{m} \vec{f}^{(i)} \otimes \vec{f}^{(i)} \longrightarrow \sum_{t=1}^{K} \vec{\mu}_t \otimes \vec{\mu}_t.$$

Tensor of word-triple frequencies (from m documents)

$$\widehat{\text{Triples}} \approx \frac{1}{m}\sum_{i=1}^{m} \vec{f}^{(i)} \otimes \vec{f}^{(i)} \otimes \vec{f}^{(i)} \longrightarrow \sum_{t=1}^{K} \vec{\mu}_t \otimes \vec{\mu}_t \otimes \vec{\mu}_t.$$

Aside: implementation for bag-of-words models

Use inner product system given by $\langle \vec{x}, \vec{y} \rangle := \vec{x}^\top \widehat{\text{Pairs}}^\dagger \vec{y}$.

Why?: If $\widehat{\text{Pairs}} = \sum_{t=1}^{K} \vec{\mu}_t \otimes \vec{\mu}_t$, then $\langle \vec{\mu}_i, \vec{\mu}_j \rangle = \mathbb{1}_{\{i=j\}}$.
$\Rightarrow \; \{\vec{\mu}_i\}$ are orthonormal under this inner product system.

Power iteration step:

$$\phi_{\widehat{\text{Triples}}}(\vec{x}) \; := \; \frac{1}{m} \sum_{i=1}^{m} \langle \vec{x}, \vec{f}^{(i)} \rangle^2 \, \vec{f}^{(i)} \; = \; \frac{1}{m} \sum_{i=1}^{m} \left(\vec{x}^\top \widehat{\text{Pairs}}^\dagger \vec{f}^{(i)} \right)^2 \vec{f}^{(i)}.$$

1. First compute $\vec{y} := \widehat{\text{Pairs}}^\dagger \vec{x}$ (use low-rank factors of $\widehat{\text{Pairs}}$).
2. Then compute $(\vec{y}^\top \vec{f}^{(i)})^2 \, \vec{f}^{(i)}$ for all documents i, and add them up (all sparse operations).

Final running time $\propto$ # topics $\times$ (model size $+$ input size).

Effect of errors in tensor power iterations

Suppose we are given $\widehat{T} := T + E$, with

$$T = \sum_{t=1}^{n} \lambda_t \, \vec{v}_t \otimes \vec{v}_t \otimes \vec{v}_t, \qquad \varepsilon := \sup_{\vec{x} \in \mathbb{S}^{n-1}} \|\phi_E(\vec{x})\|.$$

What can we say about the resulting $\hat{v}_i$ and $\hat{\lambda}_i$?

Perturbation analysis

Theorem: If $\varepsilon \leq O(\frac{\min_t \lambda_t}{n})$, then with high probability, a modified variant of the full decomposition algorithm returns $\{(\hat{v}_i, \hat{\lambda}_i) : i \in [n]\}$ with

$$\|\hat{v}_i - \vec{v}_i\| \leq O(\varepsilon/\lambda_i), \qquad |\hat{\lambda}_i - \lambda_i| \leq O(\varepsilon), \qquad i \in [n].$$

Essentially third-order analogue of Wedin's theorem for SVD of matrices, but specific to fixed-point iteration algorithm.

Similar analysis holds for variational characterization.

Effect of errors in tensor power iterations

Quadratic operator $\phi_{\widehat{T}}$ with $\widehat{T}$:

$$\phi_{\widehat{T}}(\vec{x}) \;=\; \sum_{t=1}^{n} \lambda_t \left(\vec{v}_t^{\top}\vec{x}\right)^2 \vec{v}_t \;+\; \phi_E(\vec{x}).$$

Claim: If $\varepsilon \leq O(\frac{\min_t \lambda_t}{n})$ and $N \geq \Omega(\log(n) + \log\log \frac{\max_t \lambda_t}{\varepsilon})$, then N steps of tensor power iteration on $T + E$ (with good initialization) gives

$$\|\hat{v}_i - \vec{v}_i\| \leq O(\varepsilon/\lambda_i), \qquad |\hat{\lambda}_i - \lambda_i| \leq O(\varepsilon).$$

Deflation

(For simplicity, assume $\lambda_1 = \cdots = \lambda_n = 1$.)

Using tensor power iteration on $\widehat{T} := T + E$:
Approximate (say) $\vec{v}_1$ with $\hat{v}_1$ up to error $\|\vec{v}_1 - \hat{v}_1\| \leq \varepsilon$.

Deflation danger: To find next $\vec{v}_t$, use

$$\widehat{T} - \hat{v}_1 \otimes \hat{v}_1 \otimes \hat{v}_1 = \sum_{t=2}^{n} \vec{v}_t \otimes \vec{v}_t \otimes \vec{v}_t + E + \left(\vec{v}_1 \otimes \vec{v}_1 \otimes \vec{v}_1 - \hat{v}_1 \otimes \hat{v}_1 \otimes \hat{v}_1\right).$$

Now error seems to be of size $2\varepsilon \ldots$ exponential explosion?

How do the errors look?

$$E_1 := \vec{v}_1 \otimes \vec{v}_1 \otimes \vec{v}_1 - \hat{v}_1 \otimes \hat{v}_1 \otimes \hat{v}_1$$

- Take any direction $\vec{x}$ orthogonal to $\vec{v}_1$:

$$\begin{aligned} \|\phi_{E_1}(\vec{x})\| &= \|(\vec{v}_1{}^\top \vec{x})^2 \vec{v}_1 - (\hat{v}_1{}^\top \vec{x})^2 \hat{v}_1\| \\ &= \|(\hat{v}_1{}^\top \vec{x})^2 \hat{v}_1\| \\ &= ((\hat{v}_1 - \vec{v}_1)^\top \vec{x})^2 \\ &\le \|\hat{v}_1 - \vec{v}_1\|^2 \le \varepsilon^2. \end{aligned}$$

- Effect of $E + E_1$ in directions orthogonal to $\vec{v}_1$ is just $(1 + o(1))\varepsilon$.

Deflation analysis

Upshot: all errors due to "deflation" have only lower-order effects on ability to find subsequent $\vec{v}_t$.

Analogous statement for matrix power iteration is **not true.**

Recap and remarks

- Orthogonally diagonalizable tensors have very nice *identifiability*, *computational*, and *robustness* properties.
  - Many analogues to matrix SVD, but also many important differences arising from non-linearity.
  - Greedy algorithm for finding the decomposition can be rigorously analyzed and shown to be effective and efficient.
- Many other approaches to moment-based estimation (*e.g.*, subspace ID / OOMs, local optimization).

Other stuff I didn't talk about

1. Overcomplete tensor decomposition: $K > n$ components in $\mathbb{R}^n$.

$$T = \sum_{t=1}^{K} \lambda_t \, \vec{v}_t \otimes \vec{v}_t \otimes \vec{v}_t.$$

    - ICA/blind source separation [Cardoso, 1991; Goyal *et al*, 2014]
    - Mixture models [Bhaskara *et al*, 2014; Anderson *et al*, 2014]
    - Dictionary learning [Barak *et al*, 2014]
    - …

2. General Tucker decompositions (CPD is a special case).
    - Exploit other structure (*e.g.*, sparsity)

Questions?

Tensor product of vectors spaces

What is the tensor product $V \bigotimes W$ of vector spaces V and W?

- Define objects $E_{\vec{v},\vec{w}}$ for $\vec{v} \in V$ and $\vec{w} \in W$.
- Declare equivalences
  - $E_{\vec{v}_1+\vec{v}_2,\vec{w}} \sim E_{\vec{v}_1,\vec{w}} + E_{\vec{v}_2,\vec{w}}$
  - $E_{\vec{v},\vec{w}_1+\vec{w}_2} \sim E_{\vec{v},\vec{w}_1} + E_{\vec{v},\vec{w}_2}$
  - $c\, E_{\vec{v},\vec{w}} \sim E_{c\vec{v},\vec{w}} \sim E_{\vec{v},c\vec{w}}$ for $c \in \mathbb{R}$.
- Pick any bases B_V for V, and B_W for W.

  $V \bigotimes W :=$ span of $\{E_{\vec{v},\vec{w}} : \vec{v} \in B_V, \vec{w} \in B_W\}$, modulo equivalences (eliminating dependence on choice of bases).
- Can check that $V \bigotimes W$ is a vector space.
- $\vec{v} \otimes \vec{w}$ (tensor product of $\vec{v} \in V$ and $\vec{w} \in W$) is the equivalence class of $E_{\vec{v},\vec{w}}$ in $V \bigotimes W$.

Tensor algebra perspective

From tensor algebra: Since $\{\vec{v}_t \;:\; t \in [n]\}$ is a basis for $\mathbb{R}^n$,
$\left\{\vec{v}_i \otimes \vec{v}_j \otimes \vec{v}_k \;:\; i, j, k \in [n]\right\}$ is a *basis* for $\mathbb{R}^n \bigotimes \mathbb{R}^n \bigotimes \mathbb{R}^n$
("$\bigotimes$" denotes the tensor product of vector spaces)

Every tensor $T \in \mathbb{R}^n \bigotimes \mathbb{R}^n \bigotimes \mathbb{R}^n$ has a unique representation in this basis:

$$T = \sum_{i,j,k} c_{i,j,k} \, \vec{v}_i \otimes \vec{v}_j \otimes \vec{v}_k$$

N.B.: $\dim(\mathbb{R}^n \bigotimes \mathbb{R}^n \bigotimes \mathbb{R}^n) = n^3$.

Aside: general bases for $\mathbb{R}^n \bigotimes \mathbb{R}^n \bigotimes \mathbb{R}^n$

Pick any bases $\left(\{\vec{\alpha}_i\}, \ \{\vec{\beta}_i\}, \ \{\vec{\gamma}_i\}\right)$ for $\mathbb{R}^n$
(not necessary orthonormal). $\Rightarrow$ Basis for $\mathbb{R}^n \bigotimes \mathbb{R}^n \bigotimes \mathbb{R}^n$:

$$\left\{\vec{\alpha}_i \otimes \vec{\beta}_j \otimes \vec{\gamma}_k \ : \ 1 \leq i, j, k \leq n\right\}.$$

Every tensor $T \in \mathbb{R}^n \bigotimes \mathbb{R}^n \bigotimes \mathbb{R}^n$ has a unique representation in this basis:

$$T = \sum_{i,j,k} c_{i,j,k} \ \vec{\alpha}_i \otimes \vec{\beta}_j \otimes \vec{\gamma}_k.$$

A tensor T such that $c_{i,j,k} \neq 0 \Rightarrow i = j = k$ is called *diagonal*:

$$T = \sum_{i=1}^{n} c_{i,i,i} \ \vec{\alpha}_i \otimes \vec{\beta}_i \otimes \vec{\gamma}_i.$$

Claim: A tensor T can be diagonal w.r.t. at most one basis.

Aside: canonical polyadic decomposition

Rank-K **canonical polyadic decomposition** (CPD) of T
(also called PARAFAC, CANDECOMP, or CP):

$$T = \sum_{i=1}^{K} \sigma_i \, \vec{u}_i \otimes \vec{v}_i \otimes \vec{w}_i .$$

Number of parameters: $K \cdot (3n + 1)$ (compared to n^3 in general).

Fact: If T is diagonal w.r.t. bases then it has a rank-K CPD with $K \leq n$.

Diagonal w.r.t. bases $\equiv$ "non-overcomplete" CPD.

N.B.: *Overcomplete* ($K > n$) CPD is also interesting *and a possibility* as long as $K(3n + 1) \ll n^3$.

Initialization of tensor power iteration

Let $t_{\max} := \arg\max_t \lambda_t$, and draw $\vec{x}^{(0)} \in \mathbb{S}^{n-1}$ unif. at random.

- Most coefficients of $\vec{x}^{(0)}$ are around $1/\sqrt{n}$; largest is around $\sqrt{\log(n)/n}$.
- Almost surely, a gap exists:

$$\max_{t \neq t_{\max}} \frac{\lambda_t |\vec{v}_t^{\top} \vec{x}^{(0)}|}{\lambda_{t_{\max}} |\vec{v}_{t_{\max}}^{\top} \vec{x}^{(0)}|} < 1.$$

- With probability $\geq 1/n^{1.2}$, the gap is non-negligible:

$$\max_{t \neq t_{\max}} \frac{\lambda_t |\vec{v}_t^{\top} \vec{x}^{(0)}|}{\lambda_{t_{\max}} |\vec{v}_{t_{\max}}^{\top} \vec{x}^{(0)}|} < 0.9.$$

Try $O(n^{1.3})$ initializers; chances are at least one is good.
(Very conservative estimate only; can be *much* better than this.)