Learning latent variable models using tensor decompositions

Daniel Hsu

Columbia University

January 27, 2017
Learning algorithms
for latent variable models
based on decompositions of moment tensors.
Learning algorithms (parameter estimation) for latent variable models based on decompositions of moment tensors.

“Method-of-moments” (Pearson, 1894)
Example #1: summarizing a corpus of documents

Observation: documents express one or more thematic topics.

Team Relocations Keep N.F.L. Moving Up Financially

The Chargers’ announced move to Los Angeles will add even more money for owners amid growing uncertainties facing the league.

By KEN BELSON

Jan. 12, 2017
Example #1: summarizing a corpus of documents

Observation: *documents express one or more thematic topics.*

Team Relocations Keep N.F.L. Moving Up Financially

The Chargers’ announced move to Los Angeles will add even more money for owners amid growing uncertainties facing the league.

By KEN BELSON

Jan. 12, 2017

- What topics are expressed in a corpus of documents?
- How prevalent is each topic in the corpus?
Topic model (e.g., latent Dirichlet allocation)

$K$ topics (distributions over vocab words).

Document $\equiv$ mixture of topics.

Word tokens in doc. $\sim$ iid mixture distribution.
Topic model (e.g., latent Dirichlet allocation)

$K$ topics (distributions over vocab words).
Document $\equiv$ mixture of topics.
Word tokens in doc. $\overset{iid}{\sim}$ mixture distribution.

E.g.,

$\overset{iid}{\sim} 0.7 \times P_{\text{sports}} + 0.3 \times P_{\text{business}}.$
Topic model (e.g., latent Dirichlet allocation)

$K$ topics (distributions over vocab words).  
Document $\equiv$ mixture of topics.  
Word tokens in doc. $\overset{iid}{\sim}$ mixture distribution.

E.g.,  

$$\overset{iid}{\sim} 0.7 \times P_{\text{sports}} + 0.3 \times P_{\text{business}}.$$  

Given corpus of documents (and “hyper-parameters”, e.g., $K$),  
produce estimates of model parameters, e.g.:  

- Distribution $P_t$ over vocab words, for each $t \in [K]$.  
- Weight $w_t$ of topic $t$ in document corpus, for each $t \in [K]$.
Suppose each word token \( x \) in document is annotated with source topic \( t_x \in \{1, 2, \ldots, K\} \).

<table>
<thead>
<tr>
<th>Team</th>
<th>Relocations</th>
<th>Keep</th>
<th>N.F.L.</th>
<th>Moving</th>
<th>Up</th>
<th>Financially</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
</tbody>
</table>
Suppose each word token $x$ in document is annotated with source topic $t_x \in \{1, 2, \ldots, K\}$.

<table>
<thead>
<tr>
<th>Team</th>
<th>Relocations</th>
<th>Keep</th>
<th>N.F.L.</th>
<th>Moving</th>
<th>Up</th>
<th>Financially</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
</tbody>
</table>

Then estimating the $\{(P_t, w_t)\}_{t=1}^{K}$ can be done “directly”.
Suppose each word token $x$ in document is annotated with source topic $t_x \in \{1, 2, \ldots, K\}$.

Then estimating the $\{(P_{t, w})_{t=1}^{K}\}$ can be done “directly”.

Unfortunately, we often don’t have such annotations (i.e., data are unlabeled / topics are hidden).

“Direct” approach to estimation unavailable.
Example #2: subpopulations in data

Data studied by Pearson (1894):
ratio of forehead-width to body-length for 1000 crabs.
Example #2: subpopulations in data

Data studied by Pearson (1894):
ratio of forehead-width to body-length for 1000 crabs.

Sample may be comprised of different sub-species of crabs.
Gaussian mixture model

\[ H \sim \text{Discrete}(\pi_1, \pi_2, \ldots, \pi_K); \]
\[ X \mid H = t \sim \text{Normal}(\mu_t, \Sigma_t), \quad t \in [K]. \]
Gaussian mixture model

\[ H \sim \text{Discrete}(\pi_1, \pi_2, \ldots, \pi_K); \]
\[ X \mid H = t \sim \text{Normal}(\mu_t, \Sigma_t), \quad t \in [K]. \]

Estimate mean vector, covariance matrix, and mixing weight of each subpopulation from unlabeled data.
Maximum likelihood estimation

- No “direct” estimators when some variables are hidden.
Maximum likelihood estimation

- No “direct” estimators when some variables are hidden.

- Maximum likelihood estimator (MLE):

  \[ \theta_{\text{MLE}} := \arg \max_{\theta \in \Theta} \log \Pr_{\theta}(\text{data}) . \]
Maximum likelihood estimation

- No “direct” estimators when some variables are hidden.

- Maximum likelihood estimator (MLE):

  \[
  \theta_{\text{MLE}} := \arg \max_{\theta \in \Theta} \log \Pr_{\theta} (\text{data}).
  \]

- **Note**: Log-likelihood is not necessarily concave function of \( \theta \).
Maximum likelihood estimation

- No “direct” estimators when some variables are hidden.

- **Maximum likelihood estimator (MLE):**

  \[
  \theta_{MLE} := \arg \max_{\theta \in \Theta} \log \Pr_\theta (\text{data}) .
  \]

  - **Note:** log-likelihood is not necessarily concave function of \( \theta \).

  - For latent variable models, often use local optimization, most notably via **Expectation-Maximization (EM)** (Dempster, Laird, & Rubin, 1977).
MLE for Gaussian mixture models

Given data \( \{ \mathbf{x}_i \}_{i=1}^n \), find \( \{ (\mu_t, \Sigma_t, \pi_t) \}_{t=1}^K \) to maximize

\[
\sum_{i=1}^n \log \left( \sum_{t=1}^K \pi_t \cdot \frac{1}{\det(\Sigma_t)^{1/2}} \exp \left\{ -\frac{1}{2}(\mathbf{x}_i - \mu_t)\top \Sigma_t^{-1}(\mathbf{x}_i - \mu_t) \right\} \right)
\]
MLE for Gaussian mixture models

Given data \( \{x_i\}_{i=1}^n \), find \( \{ (\mu_t, \Sigma_t, \pi_t) \}_{t=1}^K \) to maximize

\[
\sum_{i=1}^n \log \left( \sum_{t=1}^K \pi_t \cdot \frac{1}{\det(\Sigma_t)^{1/2}} \exp \left\{ -\frac{1}{2} (x_i - \mu_t)^\top \Sigma_t^{-1} (x_i - \mu_t) \right\} \right).
\]

- Sensible with restrictions on \( \Sigma_t \) (e.g., \( \Sigma_t \succeq \sigma^2 I \)).
MLE for Gaussian mixture models

Given data \( \{x_i\}_{i=1}^n \), find \( \{(\mu_t, \Sigma_t, \pi_t)\}_{t=1}^K \) to maximize

\[
\sum_{i=1}^n \log \left( \sum_{t=1}^K \pi_t \cdot \frac{1}{\det(\Sigma_t)^{1/2}} \exp \left\{ -\frac{1}{2} (x_i - \mu_t)\top \Sigma_t^{-1} (x_i - \mu_t) \right\} \right).
\]

- Sensible with restrictions on \( \Sigma_t \) (e.g., \( \Sigma_t \succeq \sigma^2 I \)).
- Similar to Euclidean \( K \)-means problem, which is NP-hard (Dasgupta, 2008; Aloise, Deshpande, Hansen, & Popat, 2009; Mahajan, Nimbhorkar, & Varadarajan, 2009; Vattani, 2009; Awasthi, Charikar, Krishnaswamy, & Sinop, 2015).
Parameter learning objective

Suppose iid sample of size \( n \) is generated by distribution from model with (unknown) parameters \( \theta \in \Theta \subseteq \mathbb{R}^p \) \( (p = \# \text{ params}) \).
Parameter learning objective

Suppose iid sample of size $n$ is generated by distribution from model with (unknown) parameters $\theta \in \Theta \subseteq \mathbb{R}^p$ ($p = \#\text{ params}$).

**Task:** Produce estimate $\hat{\theta}$ of $\theta$ such that

$$\mathbb{E} \| \hat{\theta} - \theta \| \to 0 \quad \text{as} \quad n \to \infty$$

(i.e., $\hat{\theta}$ is consistent).
Parameter learning objective

Suppose iid sample of size $n$ is generated by distribution from model with (unknown) parameters $\theta \in \Theta \subseteq \mathbb{R}^p$ ($p = \# \text{ params}$).

**Task**: Produce estimate $\hat{\theta}$ of $\theta$ such that

$$\mathbb{E} \| \hat{\theta} - \theta \| \rightarrow 0 \quad \text{as} \quad n \rightarrow \infty$$

(i.e., $\hat{\theta}$ is consistent).

- E.g., for spherical Gaussian mixtures (as $n \rightarrow \infty$):
  - For $K = 2$ (and $\pi_t = 1/2$, $\Sigma_t = I$): EM is consistent (Xu, H., & Maleki, 2016; Daskalakis, Tzamos, & Zampetakis, 2016).
Parameter learning objective

Suppose iid sample of size $n$ is generated by distribution from model with (unknown) parameters $\theta \in \Theta \subseteq \mathbb{R}^p$ ($p = \# \text{ params}$).

**Task:** Produce estimate $\hat{\theta}$ of $\theta$ such that

$$\mathbb{E} \| \hat{\theta} - \theta \| \rightarrow 0 \quad \text{as} \quad n \rightarrow \infty$$

(i.e., $\hat{\theta}$ is consistent).

► E.g., for spherical Gaussian mixtures (as $n \rightarrow \infty$):

► For $K = 2$ (and $\pi_t = 1/2$, $\Sigma_t = I$): EM is consistent (Xu, H., & Maleki, 2016; Daskalakis, Tzamos, & Zampetakis, 2016).

► Larger $K$: easily trapped in local maxima, far from global max (Jin, Zhang, Balakrishnan, Wainwright, & Jordan, 2016).
Parameter learning objective

Suppose iid sample of size $n$ is generated by distribution from model with (unknown) parameters $\theta \in \Theta \subseteq \mathbb{R}^p$ ($p = \#$ params).

**Task:** Produce estimate $\hat{\theta}$ of $\theta$ such that

$$\mathbb{E} \| \hat{\theta} - \theta \| \to 0 \quad \text{as} \quad n \to \infty$$

(i.e., $\hat{\theta}$ is consistent).

- E.g., for spherical Gaussian mixtures (as $n \to \infty$):
  - For $K = 2$ (and $\pi_t = 1/2$, $\Sigma_t = I$): EM is consistent (Xu, H., & Maleki, 2016; Daskalakis, Tzamos, & Zampetakis, 2016).
  - Larger $K$: easily trapped in local maxima, far from global max (Jin, Zhang, Balakrishnan, Wainwright, & Jordan, 2016).

Practitioners often use EM with many (random) restarts ... but may take a long time to get near the global max.
Parameter learning objective

Suppose iid sample of size $n$ is generated by distribution from model with (unknown) parameters $\theta \in \Theta \subseteq \mathbb{R}^p$ ($p = \# \text{ params}$).

**Task:** Produce estimate $\hat{\theta}$ of $\theta$ such that

$$\Pr \left( \| \hat{\theta} - \theta \| \leq \epsilon \right) \geq 1 - \delta$$

with $\text{poly}(p, 1/\epsilon, 1/\delta, \ldots)$ sample size and running time.

- E.g., for spherical Gaussian mixtures (as $n \to \infty$):
  - For $K = 2$ (and $\pi_t = 1/2$, $\Sigma_t = I$): EM is consistent (Xu, H., & Maleki, 2016; Daskalakis, Tzamos, & Zampetakis, 2016).
  - Larger $K$: easily trapped in local maxima, far from global max (Jin, Zhang, Balakrishnan, Wainwright, & Jordan, 2016).

Practitioners often use EM with many (random) restarts . . . but may take a long time to get near the global max.
Barriers

Hard to learn model parameters, even when data is generated by a model distribution.
Barriers

Hard to learn model parameters, even when data is generated by a model distribution.

- Cryptographic hardness (e.g., Mossel & Roch, 2006)
- Information-theoretic hardness (e.g., Moitra & Valiant, 2010)

May require $2^{\Omega(K)}$ running time or $2^{\Omega(K)}$ sample size.
Ways around the barriers

▶ Separation conditions.

E.g., assume \( \min_{i \neq j} \frac{\| \mu_i - \mu_j \|^2}{\sigma_i^2 + \sigma_j^2} \) is sufficiently large.

(Dasgupta, 1999; Arora & Kannan, 2001; Vempala & Wang, 2002; ...)
Ways around the barriers

▶ Separation conditions.

E.g., assume \[\min_{i \neq j} \frac{||\mu_i - \mu_j||^2}{\sigma_i^2 + \sigma_j^2}\] is sufficiently large.

(Dasgupta, 1999; Arora & Kannan, 2001; Vempala & Wang, 2002; …)

▶ Structural assumptions.

E.g., sparsity, anchor words.

(Spielman, Wang, & Wright, 2012; Arora, Ge, & Moitra, 2012; …)
Ways around the barriers

▶ Separation conditions.

E.g., assume \( \min_{i \neq j} \frac{\| \mu_i - \mu_j \|^2}{\sigma_i^2 + \sigma_j^2} \) is sufficiently large.

(Dasgupta, 1999; Arora & Kannan, 2001; Vempala & Wang, 2002; ...)

▶ Structural assumptions.

E.g., sparsity, anchor words.

(Spielman, Wang, & Wright, 2012; Arora, Ge, & Moitra, 2012; ...)

▶ Non-degeneracy conditions.

E.g., assume \( \mu_1, \mu_2, \ldots, \mu_K \) are in general position.
Ways around the barriers

- **Separation conditions.**
  
  E.g., assume \( \min_{i \neq j} \frac{\|\mu_i - \mu_j\|^2}{\sigma_i^2 + \sigma_j^2} \) is sufficiently large.

  (Dasgupta, 1999; Arora & Kannan, 2001; Vempala & Wang, 2002; ...)

- **Structural assumptions.**

  E.g., sparsity, anchor words.

  (Spielman, Wang, & Wright, 2012; Arora, Ge, & Moitra, 2012; ...)

- **Non-degeneracy conditions.**

  E.g., assume \( \mu_1, \mu_2, \ldots, \mu_K \) are in general position.

**This talk:** learning algorithms for non-degenerate instances via method-of-moments.
Method-of-moments at a glance

1. Determine function of model parameters $\theta$ estimatable from observable data:

$$\mathbb{E}_{\theta}[f(X)] \quad ("moments").$$

2. Form estimates of moments using data (e.g., iid sample):

$$\hat{\mathbb{E}}[f(X)] \quad ("empirical moments").$$

3. Approximately solve equations for parameters $\theta$:

$$\mathbb{E}_{\theta}[f(X)] = \hat{\mathbb{E}}[f(X)].$$

4. ("Fine-tune" estimated parameters with local optimization.)
Method-of-moments at a glance

1. Determine function of model parameters $\theta$ estimatable from observable data:

$$\mathbb{E}_{\theta}[f(X)] \quad (\text{"moments")}.$$ 

Which moments?

2. Form estimates of moments using data (e.g., iid sample):

$$\hat{\mathbb{E}}[f(X)] \quad (\text{"empirical moments")}.$$ 

3. Approximately solve equations for parameters $\theta$:

$$\mathbb{E}_{\theta}[f(X)] = \hat{\mathbb{E}}[f(X)].$$

How?

4. ("Fine-tune" estimated parameters with local optimization.)
Method-of-moments at a glance

1. Determine function of model parameters $\theta$ estimatable from observable data:

$$\mathbb{E}_\theta[f(X)] \quad (\text{“moments”}).$$

**Which moments?** Often third-order moments suffice.

2. Form estimates of moments using data (e.g., iid sample):

$$\hat{\mathbb{E}}[f(X)] \quad (\text{“empirical moments”}).$$

3. Approximately solve equations for parameters $\theta$:

$$\mathbb{E}_\theta[f(X)] = \hat{\mathbb{E}}[f(X)].$$

**How?** Algorithms for tensor decomposition.

4. (“Fine-tune” estimated parameters with local optimization.)
Unresolved issues

- Handle model misspecification, increase robustness.
  - Can tolerate some independence assumptions but not others?

- General methodology.
  - At present, *ad hoc* to instantiate; guided by examples.

- Incorporate general prior knowledge.

- Incorporate user feedback interactively.
Outline

1. Warm-up: topic model for single-topic documents.
   - Identifiability.
   - Parameter recovery via decompositions of exact moments.

2. Moment decompositions for other models.
   - Mixtures of Gaussians and linear regressions.
   - Multi-view models.

3. Error-tolerant algorithms for tensor decompositions.
Other models amenable to moment tensor decomposition

- Models for independent components analysis (Comon, 1994; Frieze, Jerrum, & Kannan, 1996; Arora, Ge, Moitra & Sachdeva, 2012; Anandkumar, Foster, H., Kakade, & Liu, 2012, 2015; Belkin, Rademacher, & Voss, 2013; etc.)
- Mixed-membership stochastic blockmodels (Anandkumar, Ge, H., & Kakade, 2013, 2014)
- Simple probabilistic grammars (H., Kakade, & Liang, 2012)
- Noisy-or networks (Halpern & Sontag, 2013; Jernite, Halpern & Sontag, 2013; Arora, Ge, Ma, & Risteski, 2016)
- Indian buffet process (Tung & Smola, 2014)
- Mixed multinomial logit model (Oh & Shah, 2014)
- Dawid-Skene model (Zhang, Chen, Zhou, & Jordan, 2014)
- Multi-task bandits (Azar, Lazaric, & Brunskill, 2013)
- Partially obs. MDPs (Azizzadenesheli, Lazaric, & Anandkumar, 2016)
- ...
1. Warm-up: topic model for single-topic documents
General topic model (e.g., Latent Dirichlet Allocation)

$K$ topics (dists. over words) $\{P_t\}_{t=1}^K$.

Document $\equiv$ mixture of topics (hidden).

Word tokens in doc. iid $\sim$ mixture distribution.
Topic model for single-topic documents

\[ K \text{ topics (dists. over words)} \left\{ \mathbf{P}_t \right\}_{t=1}^K. \]

Pick topic \( t \) with prob. \( w_t \) (hidden).

Word tokens in doc. \( \sim \) iid \( \mathbf{P}_t \).
Topic model for single-topic documents

Given iid sample of documents of length $L$, produce estimates of model parameters $\{(P_t, w_t)\}_{t=1}^K$. 

$K$ topics (dists. over words) $\{P_t\}_{t=1}^K$. Pick topic $t$ with prob. $w_t$ (hidden).

Word tokens in doc. $\sim P_t$. 

How long must the documents be?

18
Topic model

**Topic model for single-topic documents**

- $K$ topics (dists. over words) $\{P_t\}_{t=1}^K$.
- Pick topic $t$ with prob. $w_t$ (hidden).
- Word tokens in doc. iid $\sim P_t$.

Given iid sample of documents of length $L$, produce estimates of model parameters $\{(P_t, w_t)\}_{t=1}^K$.

How long must the documents be?
Identifiability

- **Generative process:**
  Pick \( t \sim \text{Discrete}(w_1, w_2, \ldots, w_K) \).
  Given \( t \), pick \( L \) words from \( P_t \).
Identifiability

- **Generative process:**
  Pick \( t \sim \text{Discrete}(w_1, w_2, \ldots, w_K) \).
  Given \( t \), pick \( L \) words from \( P_t \).

- \( L = 1 \): random document \( \sim \sum_{t=1}^{K} w_t P_t \)
Identifiability

- **Generative process:**
  Pick \( t \sim \text{Discrete}(w_1, w_2, \ldots, w_K) \).
  Given \( t \), pick \( L \) words from \( P_t \).

- \( L = 1 \): random document \( \sim \sum_{t=1}^{K} w_t P_t \)
  Parameters *not identifiable* from such observations.
Identifiability

- **Generative process:**
  Pick $t \sim \text{Discrete}(w_1, w_2, \ldots, w_K)$.
  Given $t$, pick $L$ words from $P_t$.

- $L = 1$: random document $\sim \sum_{t=1}^{K} w_t P_t$
  Parameters not identifiable from such observations.

- $L = 2$:
  Regard $P_t$ as probability vector.
  Joint distribution of word pairs (for topic $t$) is given by matrix:
  \[
  P_t P_t^\top = \begin{array}{c|c}
  \text{Pr[words $i, j$]} & \hline
  \end{array}
  \]
  Random document $\sim \sum_{t=1}^{K} w_t P_t P_t^\top$. 
Identifiability

- **Generative process:**
  Pick $t \sim \text{Discrete}(w_1, w_2, \ldots, w_K)$.
  Given $t$, pick $L$ words from $P_t$.

- $L = 1$: random document $\sim \sum_{t=1}^{K} w_t P_t$
  Parameters *not identifiable* from such observations.

- $L = 2$:
  Regard $P_t$ as probability vector.
  Joint distribution of word pairs (for topic $t$) is given by matrix:

  $$P_t P_t^\top = \begin{bmatrix}
  \text{Pr}[\text{words } i, j] \\
  \text{Pr}[\text{words } i, j] \\
  \text{Pr}[\text{words } i, j] \\
  \end{bmatrix}$$

  Random document $\sim \sum_{t=1}^{K} w_t P_t P_t^\top$.
  Are parameters $\{ (P_t, w_t) \}_{t=1}^{K}$ identifiable?
Identifiability: $L = 2$

Parameters $\{(P_1, w_1), (P_2, w_2)\}$ and $\{({\tilde P}_1, {\tilde w}_1), (P_2, w_2)\}$

$$(P_1, w_1) = \left( \begin{bmatrix} 0.40 \\ 0.60 \end{bmatrix}, 0.5 \right), \quad (P_2, w_2) = \left( \begin{bmatrix} 0.60 \\ 0.40 \end{bmatrix}, 0.5 \right);$$

$$(\tilde P_1, \tilde w_1) = \left( \begin{bmatrix} 0.55 \\ 0.45 \end{bmatrix}, 0.8 \right), \quad (\tilde P_2, \tilde w_2) = \left( \begin{bmatrix} 0.30 \\ 0.70 \end{bmatrix}, 0.2 \right).$$

Cannot identify parameters from length-two documents.
Identifiability: $L = 2$

Parameters $\{(P_1, w_1), (P_2, w_2)\}$ and $\{({\tilde{P}}_1, {\tilde{w}}_1), ({\tilde{P}}_2, {\tilde{w}}_2)\}$

\[
(P_1, w_1) = \left( \begin{bmatrix} 0.40 \\ 0.60 \end{bmatrix}, 0.5 \right), \quad (P_2, w_2) = \left( \begin{bmatrix} 0.60 \\ 0.40 \end{bmatrix}, 0.5 \right);
\]

\[
({\tilde{P}}_1, {\tilde{w}}_1) = \left( \begin{bmatrix} 0.55 \\ 0.45 \end{bmatrix}, 0.8 \right), \quad ({\tilde{P}}_2, {\tilde{w}}_2) = \left( \begin{bmatrix} 0.30 \\ 0.70 \end{bmatrix}, 0.2 \right).
\]

satisfy

\[
w_1 P_1 P_1^\top + w_2 P_2 P_2^\top = \tilde{w}_1 {\tilde{P}}_1 {\tilde{P}}_1^\top + \tilde{w}_2 {\tilde{P}}_2 {\tilde{P}}_2^\top = \begin{bmatrix} 0.26 & 0.24 \\ 0.24 & 0.26 \end{bmatrix}.
\]
Identifiability: $L = 2$

Parameters $\{(P_1, w_1), (P_2, w_2)\}$ and $\{({\tilde{P}}_1, {\tilde{w}}_1), ({\tilde{P}}_2, {\tilde{w}}_2)\}$

\[
(P_1, w_1) = \begin{pmatrix} 0.40 \\ 0.60 \end{pmatrix}, \quad \begin{pmatrix} 0.55 \\ 0.45 \end{pmatrix}, \\
(P_2, w_2) = \begin{pmatrix} 0.60 \\ 0.40 \end{pmatrix}, \quad \begin{pmatrix} 0.30 \\ 0.70 \end{pmatrix},
\]

satisfy

\[
w_1 P_1 P_1^\top + w_2 P_2 P_2^\top = {\tilde{w}}_1 {\tilde{P}}_1 {\tilde{P}}_1^\top + {\tilde{w}}_2 {\tilde{P}}_2 {\tilde{P}}_2^\top = \begin{bmatrix} 0.26 & 0.24 \\ 0.24 & 0.26 \end{bmatrix}.
\]

Cannot identify parameters from length-two documents.
Identifiability: $L = 3$

Documents of length $L = 3$
Joint distribution of word triple (for topic $t$) is given by tensor:

$$P_t \otimes P_t \otimes P_t = \Pr[\text{words } i, j, k]$$

Random document $\sim \sum_{t=1}^{K} w_t P_t \otimes P_t \otimes P_t$. 
Identifiability from documents of length three

Claim: If $\{P_t\}_{t=1}^K$ are linearly independent and all $w_t > 0$, then parameters $\{((P_t, w_t))\}_{t=1}^K$ are identifiable from word triples.
Identifiability from documents of length three

Claim: If \( \{P_t\}_{t=1}^K \) are linearly independent and all \( w_t > 0 \), then parameters \( \{(P_t, w_t)\}_{t=1}^K \) are identifiable from word triples.

- Claim implied by uniqueness of certain tensor decompositions.
Identifiability from documents of length three

Claim: If \( \{P_t\}_{t=1}^K \) are linearly independent and all \( w_t > 0 \), then parameters \( \{(P_t, w_t)\}_{t=1}^K \) are identifiable from word triples.

- Claim implied by uniqueness of certain tensor decompositions.
- Algorithmic proof via special case of Jennrich’s algorithm (Harshman, 1970).
Identifiability from documents of length three

Claim: If $\{P_t\}_{t=1}^K$ are linearly independent and all $w_t > 0$, then parameters $\{(P_t, w_t)\}_{t=1}^K$ are identifiable from word triples.

- Claim implied by uniqueness of certain tensor decompositions.
- Algorithmic proof via special case of Jennrich’s algorithm (Harshman, 1970).

Next: Brief overview of tensors.
Matrices (tensors of order two): $M \in \mathbb{R}^{d \times d}$.

- Think of as bilinear function $M : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$. 

Tensors are multi-linear generalization.
Matrices (tensors of order two): $M \in \mathbb{R}^{d \times d}$.

- Think of as \textit{bilinear function} $M : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$.

- Formula using matrix representation:

$$M(x, y) = x^\top M y = \sum_{i,j} M_{i,j} \cdot x_i y_j.$$
Matrices (tensors of order two): \( M \in \mathbb{R}^{d \times d} \).

- Think of as \textit{bilinear function} \( M: \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R} \).

- Formula using matrix representation:

\[
M(x, y) = x^\top M y = \sum_{i,j} M_{i,j} \cdot x_i y_j .
\]

- Describe \( M \) by \( d^2 \) values \( M(e_i, e_j) \).
Matrices (tensors of order two): $M \in \mathbb{R}^{d \times d}$.

- Think of as *bilinear function* $M : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$.

- Formula using matrix representation:

\[
M(x, y) = x^\top My = \sum_{i,j} M_{i,j} \cdot x_i y_j.
\]

- Describe $M$ by $d^2$ values $M(e_i, e_j)$.

Tensors are multi-linear generalization.
Tensors of order $p$

$p$-linear functions: $T : \mathbb{R}^d \times \mathbb{R}^d \times \cdots \times \mathbb{R}^d \rightarrow \mathbb{R}$. 
Tensors of order $p$

$p$-linear functions: $T: \mathbb{R}^d \times \mathbb{R}^d \times \cdots \times \mathbb{R}^d \rightarrow \mathbb{R}$.

- Describe $T$ by $d^p$ values $T(e_{i_1}, e_{i_2}, \ldots, e_{i_p})$. 

$$T(x^{(1)}, x^{(2)}, \ldots, x^{(p)}) = \sum_{i_1, i_2, \ldots, i_p} T_{i_1 i_2 \cdots i_p} x^{(1)}_{i_1} x^{(2)}_{i_2} \cdots x^{(p)}_{i_p}.$$ 

Rank-1 tensor: $T = v^{(1)} \otimes v^{(2)} \otimes \cdots \otimes v^{(p)}$.

Symmetric rank-1 tensor: $T = v \otimes v \otimes \cdots \otimes v$.
Tensors of order $p$

$p$-linear functions: $T: \mathbb{R}^d \times \mathbb{R}^d \times \cdots \times \mathbb{R}^d \rightarrow \mathbb{R}$.

- Describe $T$ by $d^p$ values $T(e_{i_1}, e_{i_2}, \ldots, e_{i_p})$.

- Identify $T$ with multi-index array $T \in \mathbb{R}^{d \times d \times \cdots \times d}$. 

Formula for function value:

$$T(x^{(1)}, x^{(2)}, \ldots, x^{(p)}) = \sum_{i_1, i_2, \ldots, i_p} T_{i_1,i_2,\ldots,i_p} \cdot x^{(1)}_{i_1} x^{(2)}_{i_2} \cdots x^{(p)}_{i_p}.$$
Tensors of order $p$

$p$-linear functions: $T: \mathbb{R}^d \times \mathbb{R}^d \times \cdots \times \mathbb{R}^d \to \mathbb{R}$.

- Describe $T$ by $d^p$ values $T(e_{i_1}, e_{i_2}, \ldots, e_{i_p})$.

- Identify $T$ with multi-index array $T \in \mathbb{R}^{d \times d \times \cdots \times d}$.

Formula for function value:

$$T(x^{(1)}, x^{(2)}, \ldots, x^{(p)}) = \sum_{i_1, i_2, \ldots, i_p} T_{i_1, i_2, \ldots, i_p} \cdot x_{i_1}^{(1)} x_{i_2}^{(2)} \cdots x_{i_p}^{(p)}.$$
Tensors of order $p$

$p$-linear functions: $T : \mathbb{R}^d \times \mathbb{R}^d \times \cdots \times \mathbb{R}^d \rightarrow \mathbb{R}$.

- Describe $T$ by $d^p$ values $T(e_{i_1}, e_{i_2}, \ldots, e_{i_p})$.

- Identify $T$ with multi-index array $T \in \mathbb{R}^{d \times d \times \cdots \times d}$.

Formula for function value:

$$T(x^{(1)}, x^{(2)}, \ldots, x^{(p)}) = \sum_{i_1, i_2, \ldots, i_p} T_{i_1, i_2, \ldots, i_p} \cdot x^{(1)}_{i_1} x^{(2)}_{i_2} \cdots x^{(p)}_{i_p}.$$ 

- Rank-1 tensor: $T = v^{(1)} \otimes v^{(2)} \otimes \cdots \otimes v^{(p)}$,

$$T(x^{(1)}, x^{(2)}, \ldots, x^{(p)}) = \langle v^{(1)}, x^{(1)} \rangle \langle v^{(2)}, x^{(2)} \rangle \cdots \langle v^{(p)}, x^{(p)} \rangle.$$
Tensors of order $p$

$p$-linear functions: $T : \mathbb{R}^d \times \mathbb{R}^d \times \cdots \times \mathbb{R}^d \to \mathbb{R}$.

- Describe $T$ by $d^p$ values $T(e_{i_1}, e_{i_2}, \ldots, e_{i_p})$.

- Identify $T$ with multi-index array $T \in \mathbb{R}^{d \times d \times \cdots \times d}$.

Formula for function value:

$$T(x^{(1)}, x^{(2)}, \ldots, x^{(p)}) = \sum_{i_1, i_2, \ldots, i_p} T_{i_1, i_2, \ldots, i_p} \cdot x^{(1)}_{i_1} x^{(2)}_{i_2} \cdots x^{(p)}_{i_p}.$$  

- Rank-1 tensor: $T = v^{(1)} \otimes v^{(2)} \otimes \cdots \otimes v^{(p)}$,

  $$T(x^{(1)}, x^{(2)}, \ldots, x^{(p)}) = \langle v^{(1)}, x^{(1)} \rangle \langle v^{(2)}, x^{(2)} \rangle \cdots \langle v^{(p)}, x^{(p)} \rangle.$$  

Symmetric rank-1 tensor: $T = v \otimes_p v = v \otimes v \otimes \cdots \otimes v$,

$$T(x^{(1)}, x^{(2)}, \ldots, x^{(p)}) = \langle v, x^{(1)} \rangle \langle v, x^{(2)} \rangle \cdots \langle v, x^{(p)} \rangle.$$  

Most Tensor Problems Are NP-Hard

CHRISTOPHER J. HILLAR, Mathematical Sciences Research Institute
LEK-HENG LIM, University of Chicago

We prove that multilinear (tensor) analogues of many efficiently computable problems in numerical linear algebra are NP-hard. Our list includes: determining the feasibility of a system of bilinear equations, deciding whether a 3-tensor possesses a given eigenvalue, singular value, or spectral norm; approximating an eigenvalue, eigenvector, singular vector, or the spectral norm; and determining the rank or best rank-1 approximation of a 3-tensor. Furthermore, we show that restricting these problems to symmetric tensors does not alleviate their NP-hardness. We also explain how deciding nonnegative definiteness of a symmetric 4-tensor is NP-hard and how computing the combinatorial hyperdeterminant is NP-, #P-, and VNP-hard.
Jennrich’s algorithm (simplified)

**Task:** Given tensor $\mathbf{T} = \sum_{t=1}^{K} \mathbf{v}_t^\otimes 3$ with linearly independent components $\{\mathbf{v}_t\}_{t=1}^{K}$, find the components (up to scaling).
Jennrich’s algorithm (simplified)

**Task**: Given tensor \( \mathbf{T} = \sum_{t=1}^{K} \mathbf{v}_t \otimes^3 \) with linearly independent components \( \{ \mathbf{v}_t \}_{t=1}^{K} \), find the components (up to scaling).

**Jennrich’s algorithm**: based on “collapsing” the tensor.
Jennrich’s algorithm (simplified)

**Task**: Given tensor $T = \sum_{t=1}^{K} v_t^{\otimes 3}$ with linearly independent components $\{v_t\}_{t=1}^{K}$, find the components (up to scaling).

**Jennrich’s algorithm**: based on “collapsing” the tensor.

- Think of $T : \mathbb{R}^d \times \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ as $T : \mathbb{R}^d \to \mathbb{R}^{d \times d}$:

$$[T(x)]_{j,k} = T(x, e_j, e_k).$$

(Like “currying” in functional programming.)
Jennrich’s algorithm (simplified)

**Task:** Given tensor $\mathbf{T} = \sum_{t=1}^{K} \mathbf{v}_t \otimes^3$ with linearly independent components $\{\mathbf{v}_t\}_{t=1}^{K}$, find the components (up to scaling).

**Jennrich’s algorithm:** based on “collapsing” the tensor.
- Think of $\mathbf{T} : \mathbb{R}^d \times \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ as $\mathbf{T} : \mathbb{R}^d \to \mathbb{R}^{d \times d}$:
  \[
  [\mathbf{T}(\mathbf{x})]_{j,k} = \mathbf{T}(\mathbf{x}, e_j, e_k).
  \]
  (Like “currying” in functional programming.)

**input** Tensor $\mathbf{T} \in \mathbb{R}^{d \times d \times d}$.
1: Pick $\mathbf{x}, \mathbf{y}$ independently & uniformly at random from $S^{d-1}$.
2: Compute and return eigenvectors of $\mathbf{T}(\mathbf{x})\mathbf{T}(\mathbf{y})^\dagger$
   (with non-zero eigenvalues).
Analysis of Jennrich’s algorithm

For $T = \sum_{t=1}^{K} \mathbf{v}_t \otimes \mathbf{v}_t \otimes \mathbf{v}_t$, linearity of “collapsing” implies

$$T(x) = \sum_{t=1}^{K} (\mathbf{v}_t \otimes \mathbf{v}_t \otimes \mathbf{v}_t)(x)$$
Analysis of Jennrich’s algorithm

For $T = \sum_{t=1}^{K} \mathbf{v}_t \otimes \mathbf{v}_t \otimes \mathbf{v}_t$, linearity of “collapsing” implies

$$T(\mathbf{x}) = \sum_{t=1}^{K} (\mathbf{v}_t \otimes \mathbf{v}_t \otimes \mathbf{v}_t)(\mathbf{x}) = \sum_{t=1}^{K} \langle \mathbf{v}_t, \mathbf{x} \rangle \mathbf{v}_t \mathbf{v}_t^T$$
Analysis of Jennrich’s algorithm

For \( T = \sum_{t=1}^{K} \mathbf{v}_t \otimes \mathbf{v}_t \otimes \mathbf{v}_t \), linearity of “collapsing” implies

\[
T(\mathbf{x}) = \sum_{t=1}^{K} (\mathbf{v}_t \otimes \mathbf{v}_t \otimes \mathbf{v}_t)(\mathbf{x}) = \sum_{t=1}^{K} \langle \mathbf{v}_t, \mathbf{x} \rangle \mathbf{v}_t \mathbf{v}_t^\top = V D_x V^\top
\]

where \( V = [\mathbf{v}_1 | \cdots | \mathbf{v}_K] \) and \( D_x = \text{diag}(\langle \mathbf{v}_1, \mathbf{x} \rangle, \ldots, \langle \mathbf{v}_K, \mathbf{x} \rangle) \).
Analysis of Jennrich’s algorithm

For \( T = \sum_{t=1}^{K} v_t \otimes v_t \otimes v_t \), linearity of “collapsing” implies

\[
T(x) = \sum_{t=1}^{K} (v_t \otimes v_t \otimes v_t)(x) = \sum_{t=1}^{K} \langle v_t, x \rangle v_t v_t^\top = V D_x V^\top
\]

where \( V = [v_1 | \cdots | v_K] \) and \( D_x = \text{diag}(\langle v_1, x \rangle, \ldots, \langle v_K, x \rangle) \).

By linear independence of \( \{v_t\}_{t=1}^{K} \) and random choice of \( x \) and \( y \):
Analysis of Jennrich’s algorithm

For $T = \sum_{t=1}^{K} \mathbf{v}_t \otimes \mathbf{v}_t \otimes \mathbf{v}_t$, linearity of “collapsing” implies

$$T(x) = \sum_{t=1}^{K} (\mathbf{v}_t \otimes \mathbf{v}_t \otimes \mathbf{v}_t)(x) = \sum_{t=1}^{K} \langle \mathbf{v}_t, x \rangle \mathbf{v}_t \mathbf{v}_t^\top = V D_x V^\top$$

where $V = [\mathbf{v}_1 | \cdots | \mathbf{v}_K]$ and $D_x = \text{diag}(\langle \mathbf{v}_1, x \rangle, \ldots, \langle \mathbf{v}_K, x \rangle)$.

By linear independence of $\{\mathbf{v}_t\}_{t=1}^{K}$ and random choice of $x$ and $y$:

1. $V$ has rank $K$;
Analysis of Jennrich’s algorithm

For $T = \sum_{t=1}^{K} v_t \otimes v_t \otimes v_t$, linearity of “collapsing” implies

$$T(x) = \sum_{t=1}^{K} (v_t \otimes v_t \otimes v_t)(x) = \sum_{t=1}^{K} \langle v_t, x \rangle v_t v_t^T = V D_x V^\top$$

where $V = [v_1 | \cdots | v_K]$ and $D_x = \text{diag}(\langle v_1, x \rangle, \ldots, \langle v_K, x \rangle)$.

By linear independence of $\{v_t\}_{t=1}^{K}$ and random choice of $x$ and $y$:

1. $V$ has rank $K$;
2. $D_x$ and $D_y$ are invertible (a.s.);
Analysis of Jennrich’s algorithm

For \( T = \sum_{t=1}^{K} v_t \otimes v_t \otimes v_t \), linearity of “collapsing” implies

\[
T(x) = \sum_{t=1}^{K} (v_t \otimes v_t \otimes v_t)(x) = \sum_{t=1}^{K} \langle v_t, x \rangle v_t v_t^\top = V D_x V^\top
\]

where \( V = [v_1 | \cdots | v_K] \) and \( D_x = \text{diag}(\langle v_1, x \rangle, \ldots, \langle v_K, x \rangle) \).

By linear independence of \( \{v_t\}_{t=1}^{K} \) and random choice of \( x \) and \( y \):

1. \( V \) has rank \( K \);
2. \( D_x \) and \( D_y \) are invertible (a.s.);
3. diagonal entries of \( D_x D_y^{-1} \) are distinct (a.s.).
Analysis of Jennrich’s algorithm

For $T = \sum_{t=1}^{K} v_t \otimes v_t \otimes v_t$, linearity of “collapsing” implies

$$T(x) = \sum_{t=1}^{K} (v_t \otimes v_t \otimes v_t)(x) = \sum_{t=1}^{K} \langle v_t, x \rangle v_t v_t^T = V D_x V^T$$

where $V = [v_1| \cdots |v_K]$ and $D_x = \text{diag}(\langle v_1, x \rangle, \ldots, \langle v_K, x \rangle)$.

By linear independence of $\{v_t\}_{t=1}^{K}$ and random choice of $x$ and $y$:

1. $V$ has rank $K$;
2. $D_x$ and $D_y$ are invertible (a.s.);
3. diagonal entries of $D_x D_y^{-1}$ are distinct (a.s.);
4. $T(x) T(y)^\dagger = V (D_x D_y^{-1}) V^\dagger$ (a.s.).
Analysis of Jennrich’s algorithm

For $T = \sum_{t=1}^{K} v_t \otimes v_t \otimes v_t$, linearity of “collapsing” implies

$$T(x) = \sum_{t=1}^{K} (v_t \otimes v_t \otimes v_t)(x) = \sum_{t=1}^{K} \langle v_t, x \rangle v_t v_t^\top = V D_x V^\top$$

where $V = [v_1 | \cdots | v_K]$ and $D_x = \text{diag}(\langle v_1, x \rangle, \ldots, \langle v_K, x \rangle)$.

By linear independence of $\{v_t\}_{t=1}^{K}$ and random choice of $x$ and $y$:

1. $V$ has rank $K$;
2. $D_x$ and $D_y$ are invertible (a.s.);
3. diagonal entries of $D_x D_y^{-1}$ are distinct (a.s.);
4. $T(x)T(y)^\dagger = V (D_x D_y^{-1}) V^\dagger$ (a.s.).

So $\{v_t\}_{t=1}^{K}$ are the eigenvectors of $T(x)T(y)^\dagger$ with distinct non-zero eigenvalues.
Application to topic model parameters

Probabilities of word triples as third-order tensor:

$$T = \sum_{t=1}^{K} w_t P_t \otimes P_t \otimes P_t = \sum_{t=1}^{K} v_t \otimes v_t \otimes v_t$$

for $v_t = w_t^{1/3} P_t$. 

▶ About pre-condition for Jennrich's algorithm:

$\{v_t\}_{t=1}^{K}$ are linearly independent $\iff \{P_t\}_{t=1}^{K}$ are linearly independent and all $w_t > 0$.

▶ Can recover $\{P_t\}_{t=1}^{K}$ from $\{c_t v_t\}_{t=1}^{K}$ for any $c_t \neq 0$.

▶ Can recover $\{(P_t, w_t)\}_{t=1}^{K}$ from $\{P_t\}_{t=1}^{K}$ and $T$. 

28
Application to topic model parameters

Probabilities of word triples as third-order tensor:

\[ T = \sum_{t=1}^{K} w_t P_t \otimes P_t \otimes P_t = \sum_{t=1}^{K} v_t \otimes v_t \otimes v_t \]

for \( v_t = w_t^{1/3} P_t \).

About pre-condition for Jennrich’s algorithm:

\[ \{v_t\}_{t=1}^{K} \text{ are linearly independent} \]
\[ \iff \{P_t\}_{t=1}^{K} \text{ are linearly independent and all } w_t > 0. \]
Application to topic model parameters

Probabilities of word triples as third-order tensor:

\[
T = \sum_{t=1}^{K} w_t P_t \otimes P_t \otimes P_t = \sum_{t=1}^{K} v_t \otimes v_t \otimes v_t
\]

for \( v_t = w_t^{1/3} P_t \).

- About pre-condition for Jennrich’s algorithm:
  \[
  \{ v_t \}_{t=1}^{K} \text{ are linearly independent} \\
  \Leftrightarrow \{ P_t \}_{t=1}^{K} \text{ are linearly independent and all } w_t > 0.
  \]

- Can recover \( \{ P_t \}_{t=1}^{K} \) from \( \{ c_t v_t \}_{t=1}^{K} \) for any \( c_t \neq 0 \).
Application to topic model parameters

Probabilities of word triples as third-order tensor:

\[ T = \sum_{t=1}^{K} w_t P_t \otimes P_t \otimes P_t = \sum_{t=1}^{K} v_t \otimes v_t \otimes v_t \]

for \( v_t = w_t^{1/3} P_t \).

About pre-condition for Jennrich’s algorithm:

- \( \{ v_t \}^{K}_{t=1} \) are linearly independent if \( \{ P_t \}^{K}_{t=1} \) are linearly independent and all \( w_t > 0 \).

- Can recover \( \{ P_t \}^{K}_{t=1} \) from \( \{ c_t v_t \}^{K}_{t=1} \) for any \( c_t \neq 0 \).

- Can recover \( \{(P_t, w_t)\}^{K}_{t=1} \) from \( \{ P_t \}^{K}_{t=1} \) and \( T \).
Recap

- Parameters of topic model for single-topic documents (satisfying linear independence condition) can be efficiently recovered from distribution of three-word documents.
Recap

- Parameters of topic model for single-topic documents (satisfying linear independence condition) can be efficiently recovered from distribution of three-word documents.

- Two-word documents not sufficient.
Illustrative empirical results

- Corpus: 300,000 New York Times articles.
- Vocabulary size: 102,660 words.
- Set number of topics $K := 50$.

Model predictive performance:
$\approx 4\text{–}8 \times$ speed-up over Gibbs sampling for LDA;

![Graph showing log loss vs. training time for different methods]
**Illustrative empirical results**

**Sample topics:** (showing top 10 words for each topic)

<table>
<thead>
<tr>
<th>Econ.</th>
<th>Baseball</th>
<th>Edu.</th>
<th>Health care</th>
<th>Golf</th>
</tr>
</thead>
<tbody>
<tr>
<td>sales</td>
<td>run</td>
<td>school</td>
<td>drug</td>
<td>player</td>
</tr>
<tr>
<td>economic</td>
<td>inning</td>
<td>student</td>
<td>patient</td>
<td>tiger_wood</td>
</tr>
<tr>
<td>consumer</td>
<td>hit</td>
<td>teacher</td>
<td>million</td>
<td>won</td>
</tr>
<tr>
<td>major</td>
<td>game</td>
<td>program</td>
<td>company</td>
<td>shot</td>
</tr>
<tr>
<td>home</td>
<td>season</td>
<td>official</td>
<td>doctor</td>
<td>play</td>
</tr>
<tr>
<td>indicator</td>
<td>home</td>
<td>public</td>
<td>companies</td>
<td>round</td>
</tr>
<tr>
<td>weekly</td>
<td>right</td>
<td>children</td>
<td>percent</td>
<td>win</td>
</tr>
<tr>
<td>order</td>
<td>games</td>
<td>high</td>
<td>cost</td>
<td>tournament</td>
</tr>
<tr>
<td>claim</td>
<td>dodger</td>
<td>education</td>
<td>program</td>
<td>tour</td>
</tr>
<tr>
<td>scheduled</td>
<td>left</td>
<td>district</td>
<td>health</td>
<td>right</td>
</tr>
</tbody>
</table>
### Sample topics: (showing top 10 words for each topic)

<table>
<thead>
<tr>
<th>Invest.</th>
<th>Election</th>
<th>auto race</th>
<th>Child’s Lit.</th>
<th>Afghan War</th>
</tr>
</thead>
<tbody>
<tr>
<td>percent</td>
<td>al_gore</td>
<td>car</td>
<td>book</td>
<td>taliban</td>
</tr>
<tr>
<td>stock</td>
<td>campaign</td>
<td>race</td>
<td>children</td>
<td>attack</td>
</tr>
<tr>
<td>market</td>
<td>president</td>
<td>driver</td>
<td>ages</td>
<td>afghanistan</td>
</tr>
<tr>
<td>fund</td>
<td>george_bush</td>
<td>team</td>
<td>author</td>
<td>official</td>
</tr>
<tr>
<td>investor</td>
<td>bush</td>
<td>won</td>
<td>read</td>
<td>military</td>
</tr>
<tr>
<td>companies</td>
<td>clinton</td>
<td>win</td>
<td>newspaper</td>
<td>u_s</td>
</tr>
<tr>
<td>analyst</td>
<td>vice</td>
<td>racing</td>
<td>web</td>
<td>united_states</td>
</tr>
<tr>
<td>money</td>
<td>presidential</td>
<td>track</td>
<td>writer</td>
<td>terrorist</td>
</tr>
<tr>
<td>investment</td>
<td>million</td>
<td>season</td>
<td>written</td>
<td>war</td>
</tr>
<tr>
<td>economy</td>
<td>democratic</td>
<td>lap</td>
<td>sales</td>
<td>bin</td>
</tr>
</tbody>
</table>
### Sample topics: (showing top 10 words for each topic)

<table>
<thead>
<tr>
<th>Web</th>
<th>Antitrust</th>
<th>TV</th>
<th>Movies</th>
<th>Music</th>
</tr>
</thead>
<tbody>
<tr>
<td>com</td>
<td>court</td>
<td>show</td>
<td>film</td>
<td>music</td>
</tr>
<tr>
<td>www</td>
<td>case</td>
<td>network</td>
<td>movie</td>
<td>song</td>
</tr>
<tr>
<td>site</td>
<td>law</td>
<td>season</td>
<td>director</td>
<td>group</td>
</tr>
<tr>
<td>web</td>
<td>lawyer</td>
<td>nbc</td>
<td>play</td>
<td>part</td>
</tr>
<tr>
<td>sites</td>
<td>federal</td>
<td>cb</td>
<td>character</td>
<td>new_york</td>
</tr>
<tr>
<td>information</td>
<td>government</td>
<td>program</td>
<td>actor</td>
<td>company</td>
</tr>
<tr>
<td>online</td>
<td>decision</td>
<td>television</td>
<td>show</td>
<td>million</td>
</tr>
<tr>
<td>mail</td>
<td>trial</td>
<td>series</td>
<td>movies</td>
<td>band</td>
</tr>
<tr>
<td>internet</td>
<td>microsoft</td>
<td>night</td>
<td>million</td>
<td>show</td>
</tr>
<tr>
<td>telegram</td>
<td>right</td>
<td>new_york</td>
<td>part</td>
<td>album</td>
</tr>
</tbody>
</table>

*etc.*
Learning algorithms

- Estimation via method-of-moments:
  1. Estimate distribution of three-word documents $\rightarrow \hat{T}$ (empirical moment tensor).
  2. Approximately decompose $\hat{T} \rightarrow$ estimates $\{(\hat{P}_t, \hat{w}_t)\}_{t=1}^K$. 

Issues:
1. Accuracy of moment estimates?
   Can more reliably estimate lower-order moments; distribution-specific sample complexity bounds.
2. Robustness of (approximate) tensor decomposition?
   Instead of Jennrich's algorithm, use more error-tolerant decomposition algorithm (also computationally efficient).
3. Generality beyond simple topic models?
   Next: Moment decompositions for other models.
Learning algorithms

- Estimation via **method-of-moments**:
  1. *Estimate* distribution of three-word documents \( \rightarrow \hat{T} \)
     (empirical moment tensor).
  2. *Approximately decompose* \( \hat{T} \) \( \rightarrow \) estimates \( \{(\hat{P}_t, \hat{w}_t)\}_{t=1}^K \).

- **Issues**:
  1. Accuracy of *moment estimates*?
  2. Robustness of *(approximate) tensor decomposition*?
  3. *Generality* beyond simple topic models?
Learning algorithms

- **Estimation via method-of-moments:**
  1. Estimate distribution of three-word documents $\rightarrow \hat{T}$ (empirical moment tensor).
  2. Approximately decompose $\hat{T} \rightarrow$ estimates $\{ (\hat{P}_t, \hat{w}_t) \}_{t=1}^K$.

- **Issues:**
  1. Accuracy of moment estimates?
     Can more reliably estimate lower-order moments; distribution-specific sample complexity bounds.
  2. Robustness of (approximate) tensor decomposition?
     Instead of Jennrich’s algorithm, use more error-tolerant decomposition algorithm (also computationally efficient).
  3. Generality beyond simple topic models?
Learning algorithms

- Estimation via *method-of-moments*:
  1. *Estimate* distribution of three-word documents $\rightarrow \hat{T}$
     (*empirical moment tensor*).
  2. *Approximately decompose* $\hat{T}$ $\rightarrow$ estimates $\{({\hat{P}_t}, {\hat{w}_t})\}_{t=1}^K$.

- **Issues:**
  1. Accuracy of *moment estimates*?
     Can more reliably estimate lower-order moments; distribution-specific sample complexity bounds.
  2. Robustness of *(approximate) tensor decomposition*?
     Instead of Jennrich’s algorithm, use more error-tolerant decomposition algorithm (also computationally efficient).
  3. *Generality* beyond simple topic models?

**Next:** Moment decompositions for other models.
2. Moment decompositions for other models
Moment decompositions

Some examples of usable moment decompositions.

1. Two classical mixture models.
Mixtures of spherical Gaussians

\[ H \sim \text{Discrete}(\pi_1, \pi_2, \ldots, \pi_K) \quad \text{(hidden)}; \]
\[ X \mid H = t \sim \text{Normal}(\mu_t, \sigma^2_t I_d), \quad t \in [K]. \]
Mixtures of spherical Gaussians

\[ H \sim \text{Discrete}(\pi_1, \pi_2, \ldots, \pi_K) \text{ (hidden)}; \]
\[ X \mid H = t \sim \text{Normal}(\mu_t, \sigma^2 I_d), \quad t \in [K]. \]

(For simplicity, restrict \( \sigma_1 = \sigma_2 = \cdots = \sigma_K = \sigma \).)
Mixtures of spherical Gaussians

\[ H \sim \text{Discrete}(\pi_1, \pi_2, \ldots, \pi_K) \quad \text{(hidden)}; \]

\[ X \mid H = t \sim \text{Normal}(\mu_t, \sigma^2 I_d), \quad t \in [K]. \]

(For simplicity, restrict \( \sigma_1 = \sigma_2 = \cdots = \sigma_K = \sigma \).)

**Generative process:**

\[ X = Y + \sigma Z \]

where \( \Pr(Y = \mu_t) = \pi_t \), and

\[ Z \sim \text{Normal}(0, I_d) \quad \text{(indep. of} \ Y). \]
Moments for spherical Gaussian mixtures

First- and second-order moments:

\[
\mathbb{E}(X) = \sum_{t=1}^{K} \pi_t \cdot \mu_t ,
\]

\[
\mathbb{E}(X \otimes X) = \sum_{t=1}^{K} \pi_t \cdot \mu_t \otimes \mu_t + \sigma^2 I_d .
\]
Moments for spherical Gaussian mixtures

First- and second-order moments:

\[ \mathbb{E}(X) = \sum_{t=1}^{K} \pi_t \cdot \mu_t , \]

\[ \mathbb{E}(X \otimes X) = \sum_{t=1}^{K} \pi_t \cdot \mu_t \otimes \mu_t + \sigma^2 I_d . \]

(Vempala & Wang, 2002):
Span of top \( K \) eigenvectors of \( \mathbb{E}(X \otimes X) \) contains \( \{ \mu_t \}_{t=1}^{K} \). → Principal component analysis (PCA).
Use of moments for mixtures of spherical Gaussians

**Separation** (Dasgupta, 1999):

# standard deviations between component means

\[ \text{sep} := \min_{i \neq j} \frac{\| \mu_i - \mu_j \|}{\sigma} . \]
Use of moments for mixtures of spherical Gaussians

**Separation** *(Dasgupta, 1999):*

# standard deviations between component means

\[ \text{sep} := \min_{i \neq j} \frac{\| \mu_i - \mu_j \|}{\sigma}. \]

▶ *(Dasgupta & Schulman, 2000, 2007):*
Distance-based clustering (e.g., EM) works when \( \text{sep} \gtrsim d^{1/4} \).
Use of moments for mixtures of spherical Gaussians

Separation (Dasgupta, 1999):

# standard deviations between component means

\[
sep := \min_{i \neq j} \frac{\|\mu_i - \mu_j\|}{\sigma}.
\]

- (Dasgupta & Schulman, 2000, 2007):
  Distance-based clustering (e.g., EM) works when \( sep \gtrsim d^{1/4} \).

- (Vempala & Wang, 2002):
  Problem becomes \( K \)-dimensional via PCA (assume \( K \leq d \)).
  Required separation reduced to \( sep \gtrsim K^{1/4} \).
Use of moments for mixtures of spherical Gaussians

**Separation** (Dasgupta, 1999):

# standard deviations between component means

\[
\text{sep} := \min_{i \neq j} \frac{\|\mu_i - \mu_j\|}{\sigma}.
\]

- (Dasgupta & Schulman, 2000, 2007):
  Distance-based clustering (e.g., EM) works when \( \text{sep} \gtrsim d^{1/4} \).

- (Vempala & Wang, 2002):
  Problem becomes \( K \)-dimensional via PCA (assume \( K \leq d \)).
  Required separation reduced to \( \text{sep} \gtrsim K^{1/4} \).

**Third-order moments** identify the mixture distribution when \( \{\mu_t\}_{t=1}^K \) are lin. indpt.; \( \text{sep} \) may be arbitrarily close to zero.
Use of moments for mixtures of spherical Gaussians

**Separation** *(Dasgupta, 1999):*  
# standard deviations between component means

\[
sep := \min_{i \neq j} \frac{\|\mu_i - \mu_j\|}{\sigma}.
\]

▶ *(Dasgupta & Schulman, 2000, 2007):*  
Distance-based clustering (e.g., EM) works when \( sep \gtrsim d^{1/4} \).

▶ *(Vempala & Wang, 2002):*  
Problem becomes \( K \)-dimensional via PCA  (assume \( K \leq d \)).  
Required separation reduced to \( sep \gtrsim K^{1/4} \).

**Third-order moments** identify the mixture distribution when \( \{\mu_t\}_{t=1}^K \) are lin. indpt.; \( sep \) may be arbitrarily close to zero.

*(Belkin & Sinha, 2010; Moitra & Valiant, 2010):*  
General Gaussians & no minimum \( sep \), but \( \Omega(K) \)th-order moments.
Third-order moments of spherical Gaussian mixtures

Generative process:

\[ X = Y + \sigma Z \]

where \( \Pr(Y = \mu_t) = \pi_t \), and \( Z \sim \text{Normal}(0, I_d) \) (indep. of \( Y \)).

Third-order moment tensor:

\[ \mathbb{E} \left( X^{\otimes 3} \right) = \mathbb{E} \left( \{Y + \sigma Z\}^{\otimes 3} \right) \]
Third-order moments of spherical Gaussian mixtures

Generative process:

\[ X = Y + \sigma Z \]

where \( \Pr(Y = \mu_t) = \pi_t \), and \( Z \sim \text{Normal}(0, I_d) \) (indep. of \( Y \)).

Third-order moment tensor:

\[
E \left( X \otimes^3 \right) = E \left( \{ Y + \sigma Z \} \otimes^3 \right) \\
= E \left( Y \otimes^3 \right) + \sigma^2 E \left( Y \otimes Z \otimes Z + Z \otimes Y \otimes Z + Z \otimes Z \otimes Y \right)
\]
Third-order moments of spherical Gaussian mixtures

Generative process:

\[ X = Y + \sigma Z \]

where \( \Pr(Y = \mu_t) = \pi_t \), and \( Z \sim \text{Normal}(0, I_d) \) (indep. of \( Y \)).

Third-order moment tensor:

\[
\begin{align*}
\mathbb{E}(X^{\otimes 3}) &= \mathbb{E}\left(\{Y + \sigma Z\}^{\otimes 3}\right) \\
&= \mathbb{E}(Y^{\otimes 3}) + \sigma^2 \mathbb{E}(Y \otimes Z \otimes Z + Z \otimes Y \otimes Z + Z \otimes Z \otimes Y) \\
&= \sum_{t=1}^{K} \pi_t \cdot \mu_t^{\otimes 3} + \tau(\sigma^2, \mu). \\
&\text{some tensor}
\end{align*}
\]
Tensor decomposition for spherical Gaussian mixtures
(H. & Kakade, 2013)

\[ \mathbb{E}(X \otimes^3) = \sum_{t=1}^{K} \pi_t \cdot \mu_t \otimes^3 + \tau(\sigma^2, \mu) \]

Claim: If \( \{\mu_t\}_{t=1}^{K} \) are linearly independent and all \( \pi_t > 0 \), then \( \{\mu_t, \pi_t\}_{t=1}^{K} \) are identifiable from
\[ T := \mathbb{E}(X \otimes^3) - \tau(\sigma^2, \mu) = \sum_{t=1}^{K} \pi_t \cdot \mu_t \otimes^3. \]

Can use, e.g., Jennrich's algorithm to recover \( \{\mu_t, \pi_t\}_{t=1}^{K} \) from \( T \).
Tensor decomposition for spherical Gaussian mixtures
(H. & Kakade, 2013)

\[ \mathbb{E}(X \otimes^3) = \sum_{t=1}^K \pi_t \cdot \mu_t \otimes^3 + \tau(\sigma^2, \mu) \]

some tensor

Claim: \( \mu \) and \( \sigma^2 \) are functions of \( \mathbb{E}(X) \) and \( \mathbb{E}(X \otimes X) \).
Tensor decomposition for spherical Gaussian mixtures
(H. & Kakade, 2013)

\[
\mathbb{E}(X \otimes^3) = \sum_{t=1}^{K} \pi_t \cdot \mu_t \otimes^3 + \tau(\sigma^2, \mu).
\]

Claim: \(\mu\) and \(\sigma^2\) are functions of \(\mathbb{E}(X)\) and \(\mathbb{E}(X \otimes X)\).

Claim: If \(\{\mu_t\}_{t=1}^{K}\) are linearly independent and all \(\pi_t > 0\), then \(\{(\mu_t, \pi_t)\}_{t=1}^{K}\) are identifiable from

\[
T := \mathbb{E}(X \otimes^3) - \tau(\sigma^2, \mu) = \sum_{t=1}^{K} \pi_t \cdot \mu_t \otimes^3.
\]
Tensor decomposition for spherical Gaussian mixtures
(H. & Kakade, 2013)

\[ \mathbb{E} (X \otimes 3) = \sum_{t=1}^{K} \pi_t \cdot \mu_t \otimes 3 + \tau(\sigma^2, \mu) \cdot \text{some tensor} \]

Claim: \( \mu \) and \( \sigma^2 \) are functions of \( \mathbb{E}(X) \) and \( \mathbb{E}(X \otimes X) \).

Claim: If \( \{\mu_t\}_{t=1}^{K} \) are linearly independent and all \( \pi_t > 0 \), then \( \{ (\mu_t, \pi_t) \}_{t=1}^{K} \) are identifiable from

\[ T := \mathbb{E}(X \otimes 3) - \tau(\sigma^2, \mu) = \sum_{t=1}^{K} \pi_t \cdot \mu_t \otimes 3. \]

Can use, e.g., Jennrich’s algorithm to recover \( \{ (\mu_t, \pi_t) \}_{t=1}^{K} \) from \( T \).
Even more Gaussian mixtures

Note: Linear independence condition on $\{\mu_t\}_{t=1}^K$ requires $K \leq d$. 

(Anderson, Belkin, Goyal, Rademacher, & Voss, 2014), (Bhaskara, Charikar, Moitra, & Vijayaraghavan, 2014) 
Mixtures of $d O(1)$ Gaussians (w/ simple or known covariance) via smoothed analysis and $O(1)$-order moments. 

(Ge, Huang, & Kakade, 2015) Also with arbitrary unknown covariances.
Even more Gaussian mixtures

**Note:** Linear independence condition on \( \{\mu_t\}_{t=1}^K \) requires \( K \leq d \).

- (Anderson, Belkin, Goyal, Rademacher, & Voss, 2014),
  (Bhaskara, Charikar, Moitra, & Vijayaraghavan, 2014)

Mixtures of \( d^{O(1)} \) Gaussians (w/ simple or known covariance) via **smoothed analysis** and \( O(1) \)-order moments.
Even more Gaussian mixtures

Note: Linear independence condition on \( \{ \mu_t \}_{t=1}^K \) requires \( K \leq d \).

- (Anderson, Belkin, Goyal, Rademacher, & Voss, 2014),
  (Bhaskara, Charikar, Moitra, & Vijayaraghavan, 2014)
Mixtures of \( d^{O(1)} \) Gaussians (w/ simple or known covariance) via \textit{smoothed analysis} and \( O(1) \)-order moments.

- (Ge, Huang, & Kakade, 2015)
Also with \textit{arbitrary unknown covariances}. 
Mixtures of linear regressions

\[ H \sim \text{Discrete}(\pi_1, \pi_2, \ldots, \pi_K) \text{ (hidden)}; \]
\[ X \sim \text{Normal}(\mu, \Sigma); \]
\[ Y \mid H = t, X = x \sim \text{Normal}(\langle \beta_t, x \rangle, \sigma^2). \]
Mixtures of linear regressions

\[ H \sim \text{Discrete}(\pi_1, \pi_2, \ldots, \pi_K) \quad \text{(hidden)}; \]
\[ X \sim \text{Normal}(\mu, \Sigma); \]
\[ Y \mid H = t, X = x \sim \text{Normal}(\langle \beta_t, x \rangle, \sigma^2). \]
Mixtures of linear regressions

\[ H \sim \text{Discrete}(\pi_1, \pi_2, \ldots, \pi_K) \quad \text{(hidden)}; \]
\[ X \sim \text{Normal}(\mu, \Sigma); \]
\[ Y \mid H = t, X = x \sim \text{Normal}(\langle \beta_t, x \rangle, \sigma^2). \]
Use of moments for mixtures of linear regressions

Second-order moments (assume $\mathbf{X} \sim \text{Normal}(\mathbf{0}, \mathbf{I}_d))$: 

$$
\mathbb{E}(\mathbf{Y}^2 \mathbf{X} \mathbf{X}^\top) = 2 \sum_{t=1}^{K} \pi_t \cdot \mathbf{\beta}_t \mathbf{\beta}_t^\top + \left( \sigma^2 + \sum_{t=1}^{K} \pi_t \cdot \|\mathbf{\beta}_t\|^2 \right) \mathbf{I}_d.
$$
Use of moments for mixtures of linear regressions

Second-order moments (assume \( \mathbf{X} \sim \text{Normal}(\mathbf{0}, \mathbf{I}_d) \)):

\[
\mathbb{E}(Y^2 \mathbf{X} \mathbf{X}^\top) = 2 \sum_{t=1}^{K} \pi_t \cdot \beta_t \beta_t^\top + \left( \sigma^2 + \sum_{t=1}^{K} \pi_t \cdot \| \beta_t \|^2 \right) \mathbf{I}_d.
\]

- Span of top \( K \) eigenvectors of \( \mathbb{E}(Y^2 \mathbf{X} \mathbf{X}^\top) \) contains \( \{ \beta_t \}_{t=1}^{K} \).
Use of moments for mixtures of linear regressions

**Second-order moments** (assume \( X \sim \text{Normal}(0, I_d) \)):

\[
\mathbb{E}(Y^2XX^\top) = 2 \sum_{t=1}^K \pi_t \cdot \beta_t \beta_t^\top + \left( \sigma^2 + \sum_{t=1}^K \pi_t \cdot \| \beta_t \|^2 \right) I_d.
\]

- Span of top \( K \) eigenvectors of \( \mathbb{E}(Y^2XX^\top) \) contains \( \{ \beta_t \}_{t=1}^K \).
- Using Stein’s identity (1973), similar approach works for GLMs (Sun, Ioannidis, & Montanari, 2013).
Second-order moments (assume $X \sim \text{Normal}(0, I_d)$):

$$
\mathbb{E}(Y^2 X X^\top) = 2 \sum_{t=1}^K \pi_t \cdot \beta_t \beta_t^\top + \left( \sigma^2 + \sum_{t=1}^K \pi_t \cdot \|\beta_t\|^2 \right) I_d.
$$

- Span of top $K$ eigenvectors of $\mathbb{E}(Y^2 X X^\top)$ contains $\{\beta_t\}_{t=1}^K$.
- Using Stein’s identity (1973), similar approach works for GLMs (Sun, Ioannidis, & Montanari, 2013).

Tensor decomposition approach:
Can recover parameters $\{(\beta_t, \pi_t)\}_{t=1}^K$ with higher-order moments (Chaganty & Liang, 2013; Yi, Caramanis, & Sanghavi, 2014, 2016).
Use of moments for mixtures of linear regressions

**Second-order moments** (assume $X \sim \text{Normal}(0, I_d)$):

$$E(Y^2XX^\top) = 2 \sum_{t=1}^{K} \pi_t \cdot \beta_t\beta_t^\top + \left(\sigma^2 + \sum_{t=1}^{K} \pi_t \cdot \|\beta_t\|^2\right) I_d.$$ 

- Span of top $K$ eigenvectors of $E(Y^2XX^\top)$ contains $\{\beta_t\}_{t=1}^{K}$.
- Using Stein’s identity (1973), similar approach works for GLMs (Sun, Ioannidis, & Montanari, 2013).

**Tensor decomposition approach:**
Can recover parameters $\{(\beta_t, \pi_t)\}_{t=1}^{K}$ with higher-order moments (Chaganty & Liang, 2013; Yi, Caramanis, & Sanghavi, 2014, 2016).

Also for GLMs, via Stein’s identity (Sedghi & Anandkumar, 2014).
Simpler setting: mixed random linear equations
(Yi, Caramanis, & Sanghavi, 2016)

\[ H \sim \text{Discrete}(\pi_1, \pi_2, \ldots, \pi_K) \text{ (hidden)}; \]
\[ X \sim \text{Normal}(0, I_d); \]
\[ Y = \langle \beta_H, X \rangle. \]
Simpler setting: mixed random linear equations
(Yi, Caramanis, & Sanghavi, 2016)

\[ H \sim \text{Discrete}(\pi_1, \pi_2, \ldots, \pi_K) \quad \text{(hidden)}; \]
\[ X \sim \text{Normal}(0, I_d); \]
\[ Y = \langle \beta_H, X \rangle. \]

Claim: If \( \{\beta_t\}_{t=1}^K \) are linearly independent and all \( \pi_t > 0 \), then parameters \( \{(\beta_t, \pi_t)\}_{t=1}^K \) are identifiable from
\[ T := \mathbb{E} \left( Y^3 X \otimes^3 \right) = 6 \sum_{t=1}^K \pi_t \cdot \beta_t \otimes^3 + \tau(\mathbb{E} Y^3 X). \]
Recap: mixtures of Gaussians and linear regressions

- Parameters of Gaussian mixture models and related models (satisfying linear independence condition) can be efficiently recovered from $O(1)$-order moments.
Recap: mixtures of Gaussians and linear regressions

- Parameters of Gaussian mixture models and related models (satisfying linear independence condition) can be efficiently recovered from $O(1)$-order moments.

- Exploit distributional properties to determine usable moments.
Recap: mixtures of Gaussians and linear regressions

- Parameters of Gaussian mixture models and related models (satisfying linear independence condition) can be efficiently recovered from $O(1)$-order moments.

- Exploit distributional properties to determine usable moments.

- Smoothed analysis: avoid linear independence condition for “most” mixture distributions.
Recap: mixtures of Gaussians and linear regressions

- Parameters of Gaussian mixture models and related models (satisfying linear independence condition) can be efficiently recovered from $O(1)$-order moments.

- Exploit distributional properties to determine usable moments.

- *Smoothed analysis*: avoid linear independence condition for “most” mixture distributions.

Next: Multi-view approach to finding usable moments.
Multi-view interpretation of topic model

**Recall:** Topic model for single-topic documents

- Pick topic $H = t$ with prob. $w_t$ (hidden).
- $K$ topics (dists. over words) $\{P_t\}_{t=1}^K$.
- Word tokens $X_1, X_2, \ldots, X_L \text{ iid } P_H$. 

Diagram:

- $H$ connects to $X_1, X_2, \ldots, X_L$
Recall: Topic model for single-topic documents

\[ H \]

\[ X_1 \quad X_2 \quad \cdots \quad X_L \]

\[ \begin{align*}
K \text{ topics (dists. over words)} & \{P_t\}_{t=1}^K.
\end{align*} \]

Pick topic \( H = t \) with prob. \( w_t \) (hidden).

Word tokens \( X_1, X_2, \ldots, X_L \text{ iid } \sim P_H. \)

Key property:
\( X_1, X_2, \ldots, X_L \) conditionally independent given \( H \).
Multi-view interpretation of topic model

**Recall**: Topic model for single-topic documents

$K$ topics (dists. over words) $\{P_t\}_{t=1}^K$. Pick topic $H = t$ with prob. $w_t$ (hidden).

Word tokens $X_1, X_2, \ldots, X_L \sim P_H$. 

Key property:
$X_1, X_2, \ldots, X_L$ conditionally independent given $H$.

Each word token $X_i$ provides new “view” of hidden variable $H$. 

Some previous theoretical analysis:
▶ (Chaudhuri, Kakade, Livescu, & Sridharan, 2009) Multi-view Gaussian mixture models.
Multi-view interpretation of topic model

Recall: Topic model for single-topic documents

\[ H \]

\[ X_1 \quad X_2 \quad \cdots \quad X_L \]

\( K \) topics (dists. over words) \( \{P_t\}_{t=1}^K \).

Pick topic \( H = t \) with prob. \( w_t \) (hidden).

Word tokens \( X_1, X_2, \ldots, X_L \) iid \( \sim P_H \).

Key property:
\( X_1, X_2, \ldots, X_L \) conditionally independent given \( H \).

Each word token \( X_i \) provides new “view” of hidden variable \( H \).

Some previous theoretical analysis:

- (Blum & Mitchell, 1998) 
  
  Co-training in semi-supervised learning.

- (Chaudhuri, Kakade, Livescu, & Sridharan, 2009) 
  Multi-view Gaussian mixture models.
Multi-view mixture model

View 1: \( X_1 \)  View 2: \( X_2 \)  View 3: \( X_3 \)

Jennrich's algorithm works in this asymmetric case provided \( \{ \mu_j(t) \}_{t=1}^K \) are linearly independent for each \( j \), and all \( \pi_t > 0 \).

(Also possible to "symmetrize" using second-order moments.)
Multi-view mixture model

Jennrich's algorithm works in this asymmetric case provided \( \{ \mu_j(t) \}_{K_t=1}^{K_t} \) are linearly independent for each \( j \), and all \( \pi_t > 0 \).

(Also possible to "symmetrize" using second-order moments.)
Multi-view mixture model

\[
E(X_1 \otimes X_2 \otimes X_3) = \sum_{t=1}^{K} \pi_t \cdot \mu_t^{(1)} \otimes \mu_t^{(2)} \otimes \mu_t^{(3)}
\]

where \( \mu_t^{(i)} = E[X_i | H = t] \),

\[ \pi_t = \Pr(H = t). \]
Multi-view mixture model

\[ E(X_1 \otimes X_2 \otimes X_3) = \sum_{t=1}^{K} \pi_t \cdot \mu_t^{(1)} \otimes \mu_t^{(2)} \otimes \mu_t^{(3)} \]

where \( \mu_t^{(i)} = \mathbb{E}[X_i | H = t] \),
\( \pi_t = \text{Pr}(H = t) \).

**Jennrich’s algorithm** works in this asymmetric case provided \( \{\mu_t^{(j)}\}_{t=1}^{K} \) are linearly independent for each \( j \), and all \( \pi_t > 0 \).
Multi-view mixture model

\[ \mathbb{E}(X_1 \otimes X_2 \otimes X_3) = \sum_{t=1}^{K} \pi_t \cdot \mu_t^{(1)} \otimes \mu_t^{(2)} \otimes \mu_t^{(3)} \]

where \( \mu_t^{(i)} = \mathbb{E}[X_i | H = t] \),
\( \pi_t = \text{Pr}(H = t) \).

**Jennrich’s algorithm** works in this asymmetric case provided \( \{\mu_t^{(j)}\}_{t=1}^{K} \) are linearly independent for each \( j \), and all \( \pi_t > 0 \).

(Also possible to “symmetrize” using second-order moments.)
Examples of multi-view mixture models
(Mossel & Roch, 2006; Anandkumar, H., & Kakade, 2012)

1. Mixtures of high-dimensional product distributions.
   (E.g., mixtures of axis-aligned Gaussians.)
Examples of multi-view mixture models
(Mossel & Roch, 2006; Anandkumar, H., & Kakade, 2012)

1. Mixtures of high-dimensional product distributions. (E.g., mixtures of axis-aligned Gaussians.)

2. Hidden Markov models.

\[ H_1 \rightarrow H_2 \rightarrow H_3 \]

\[ X_1 \rightarrow X_2 \rightarrow X_3 \]

\[ H_2 \]

\[ X_1 \rightarrow X_2 \rightarrow X_3 \]
Examples of multi-view mixture models
(Mossel & Roch, 2006; Anandkumar, H., & Kakade, 2012)

1. Mixtures of high-dimensional product distributions. (E.g., mixtures of axis-aligned Gaussians.)

2. Hidden Markov models.

3. Phylogenetic trees.
   - $X_1, X_2, X_3$: genes of three extant species.
   - $H$: LCA of most closely related pair of species.
Examples of multi-view mixture models
(Mossel & Roch, 2006; Anandkumar, H., & Kakade, 2012)

1. Mixtures of high-dimensional product distributions.
   (E.g., mixtures of axis-aligned Gaussians.)

2. Hidden Markov models.

3. Phylogenetic trees.
   - $X_1, X_2, X_3$: genes of three extant species.
   - $H$: LCA of most closely related pair of species.

4. . . .
Recap

- Parameters of many latent variable models (satisfying non-degeneracy conditions) can be efficiently recovered from $O(1)$-order moments.
Recap

- Parameters of many latent variable models (satisfying non-degeneracy conditions) can be efficiently recovered from $O(1)$-order moments.

- Exploit distributional properties, multi-view structure, and other structure to determine usable moments.
Recap

- Parameters of many latent variable models (satisfying non-degeneracy conditions) can be efficiently recovered from $O(1)$-order moments.

- Exploit distributional properties, multi-view structure, and other structure to determine usable moments.

- Estimation via method-of-moments:
  1. Estimate moments $\rightarrow$ empirical moment tensor $\hat{T}$.
  2. Approximately decompose $\hat{T} \rightarrow$ parameter estimate $\hat{\theta}$. 

Next: Error-tolerant (approximate) tensor decomposition.
Recap

- Parameters of many latent variable models (satisfying non-degeneracy conditions) can be efficiently recovered from $O(1)$-order moments.

- Exploit distributional properties, multi-view structure, and other structure to determine usable moments.

- Estimation via method-of-moments:
  1. Estimate moments $\rightarrow$ empirical moment tensor $\hat{T}$.
  2. Approximately decompose $\hat{T} \rightarrow$ parameter estimate $\hat{\theta}$.

Next: Error-tolerant (approximate) tensor decomposition.
3. Error-tolerant algorithms for tensor decompositions
Moment estimates

Estimation of $\mathbb{E}[X \otimes^3]$ (say) from iid sample $\{x_i\}_{i=1}^n$:

$$\hat{\mathbb{E}}[X \otimes^3] := \frac{1}{n} \sum_{i=1}^n x_i \otimes^3.$$
Moment estimates

Estimation of $\mathbb{E}[X^{\otimes 3}]$ (say) from iid sample $\{x_i\}_{i=1}^n$:

$$\hat{\mathbb{E}}[X^{\otimes 3}] := \frac{1}{n} \sum_{i=1}^{n} x_i^{\otimes 3}.$$ 

Inevitably expect error of order $n^{-1/2}$ in some norm, e.g.,

$$\|T\| := \sup_{x,y,z \in S^{d-1}} T(x,y,z) \quad (\text{operator norm}),$$

$$\|T\|_F := \left( \sum_{i,j,k} T_{i,j,k}^2 \right)^{1/2} \quad (\text{Frobenius norm}).$$
Using Jennrich’s algorithm

**Recall:** Jennrich’s algorithm (simplified)

**Goal:** Given tensor \( T = \sum_{t=1}^{K} v_t \otimes 3 \), find components \( \{v_t\}_{t=1}^{K} \).

**input** Tensor \( T \in \mathbb{R}^{d \times d \times d} \).

1. Pick \( x, y \) independently & uniformly at random from \( S^{d-1} \).
2. Compute and return eigenvectors of \( T(x)T(y) \) (with non-zero eigenvalues).
Using Jennrich’s algorithm

Recall: Jennrich’s algorithm (simplified)

Goal: Given tensor $T = \sum_{t=1}^{K} v_t \otimes^3 t$, find components $\{v_t\}_{t=1}^{K}$.

input Tensor $T \in \mathbb{R}^{d \times d \times d}$.
1: Pick $x, y$ independently & uniformly at random from $S^{d-1}$.
2: Compute and return eigenvectors of $T(x)T(y)\dagger$ (with non-zero eigenvalues).

But we only have $\hat{T}$, an estimate of $T = \sum_{t=1}^{K} v_t \otimes^3 t$ with (say)

$$\|\hat{T} - T\| \lesssim n^{-1/2}.$$
Stability of Jennrich’s algorithm

Stability of eigenvectors requires eigenvalue gaps.
Stability of Jennrich’s algorithm

Stability of eigenvectors requires eigenvalue gaps.

- Eigenvalue gaps for \( T(x)T(y)^\dagger \):

\[
\Delta := \min_{i \neq j} \left| \frac{\langle v_i, x \rangle}{\langle v_i, y \rangle} - \frac{\langle v_j, x \rangle}{\langle v_j, y \rangle} \right|.
\]
Stability of Jennrich’s algorithm

Stability of eigenvectors requires eigenvalue gaps.

- Eigenvalue gaps for $T(x)T(y)^\dagger$:

$$\Delta := \min_{i \neq j} \left| \frac{\langle v_i, x \rangle}{\langle v_i, y \rangle} - \frac{\langle v_j, x \rangle}{\langle v_j, y \rangle} \right| .$$

- Need $\|\hat{T}(x)\hat{T}(y)^\dagger - T(x)T(y)^\dagger\| \ll \Delta$ so that $\hat{T}(x)\hat{T}(y)^\dagger$ also has sufficient eigenvalue gaps.
Stability of Jennrich’s algorithm

Stability of eigenvectors requires eigenvalue gaps.

- Eigenvalue gaps for $T(x)T(y)^\dagger$:

$$\Delta := \min_{i \neq j} \left| \frac{\langle v_i, x \rangle}{\langle v_i, y \rangle} - \frac{\langle v_j, x \rangle}{\langle v_j, y \rangle} \right|.$$ 

- Need $\|\hat{T}(x)\hat{T}(y)^\dagger - T(x)T(y)^\dagger\| \ll \Delta$ so that $\hat{T}(x)\hat{T}(y)^\dagger$ also has sufficient eigenvalue gaps.

- Ultimately, appears to need $\|\hat{T} - T\|_F \ll \frac{1}{\text{poly}(d)}$. 

Next: A different approach.
Stability of Jennrich’s algorithm

Stability of eigenvectors requires eigenvalue gaps.

- Eigenvalue gaps for $T(x)T(y)^\dagger$:

$$\Delta := \min_{i \neq j} \left| \frac{\langle v_i, x \rangle}{\langle v_i, y \rangle} - \frac{\langle v_j, x \rangle}{\langle v_j, y \rangle} \right|. $$

- Need $\|\hat{T}(x)\hat{T}(y)^\dagger - T(x)T(y)^\dagger\| \ll \Delta$ so that $\hat{T}(x)\hat{T}(y)^\dagger$ also has sufficient eigenvalue gaps.

- Ultimately, appears to need $\|\hat{T} - T\|_F \ll \frac{1}{\text{poly}(d)}$.

Next: A different approach.
Reduction to orthonormal case

In many (all?) applications, we can estimate moments of the form

\[ M = \sum_{t=1}^{K} v_t \otimes v_t, \quad \text{(e.g., word pairs)} \]

and

\[ T = \sum_{t=1}^{K} \lambda_t \cdot v_t \otimes v_t \otimes v_t. \quad \text{(e.g., word triples)} \]

(Here, we assume \( \{v_t\}_{t=1}^{K} \) are linearly independent, and \( \{\lambda_t\}_{t=1}^{K} \) are positive.)
In many (all?) applications, we can estimate moments of the form

\[
M = \sum_{t=1}^{K} v_t \otimes v_t,
\]

(e.g., word pairs)

and

\[
T = \sum_{t=1}^{K} \lambda_t \cdot v_t \otimes v_t \otimes v_t.
\]

(e.g., word triples)

(Here, we assume \(\{v_t\}_{t=1}^{K}\) are linearly independent, and \(\{\lambda_t\}_{t=1}^{K}\) are positive.)

- \(M\) is positive semidefinite of rank \(K\).
Reduction to orthonormal case

In many (all?) applications, we can estimate moments of the form

\[ M = \sum_{t=1}^{K} v_t \otimes v_t, \quad \text{(e.g., word pairs)} \]

and

\[ T = \sum_{t=1}^{K} \lambda_t \cdot v_t \otimes v_t \otimes v_t. \quad \text{(e.g., word triples)} \]

(Here, we assume \( \{v_t\}_{t=1}^{K} \) are linearly independent, and \( \{\lambda_t\}_{t=1}^{K} \) are positive.)

- \( M \) is positive semidefinite of rank \( K \).
- \( M \) determines inner product system on \( \text{span} \{v_t\}_{t=1}^{K} \) s.t. \( \{v_t\}_{t=1}^{K} \) are orthonormal.
Goal: Given tensor $\hat{T} \in \mathbb{R}^{d \times d \times d}$ such that $\| \hat{T} - T \| \leq \varepsilon$ for some $T = \sum_{t=1}^{d} \lambda_t \cdot v_t^\otimes 3$ where $\{v_t\}_{t=1}^{d}$ are orthonormal and all $\lambda_t > 0$, approximately recover $\{(v_t, \lambda_t)\}_{t=1}^{d}$. 

Analogous matrix problems:

- $\varepsilon = 0$: eigendecomposition. ("Promised" decomposition always exists by symmetry.)
- $\varepsilon > 0$: perturbation theory for eigenvalues (Weyl) and eigenvectors (Davis & Kahan).
(Nearly) orthogonally decomposable tensors \((d = K)\)

Goal: Given tensor \(\hat{T} \in \mathbb{R}^{d \times d \times d}\) such that \(\|\hat{T} - T\| \leq \varepsilon\) for some \(T = \sum_{t=1}^{d} \lambda_t \cdot v_t^{\otimes 3}\) where \(\{v_t\}_{t=1}^{d}\) are orthonormal and all \(\lambda_t > 0\), approximately recover \(\{(v_t, \lambda_t)\}_{t=1}^{d}\).

Analogous matrix problems:

▶ \(\varepsilon = 0\): eigendecomposition.

(“Promised” decomposition always exists by symmetry.)
(Nearly) orthogonally decomposable tensors ($d = K$)

Goal: Given tensor $\hat{T} \in \mathbb{R}^{d \times d \times d}$ such that $\|\hat{T} - T\| \leq \varepsilon$ for some $T = \sum_{t=1}^{d} \lambda_t \cdot v_t \otimes^3$ where $\{v_t\}_{t=1}^{d}$ are orthonormal and all $\lambda_t > 0$, approximately recover $\{(v_t, \lambda_t)\}_{t=1}^{d}$.

Analogous matrix problems:

- $\varepsilon = 0$: eigendecomposition.
  (“Promised” decomposition always exists by symmetry.)

  Decomposition is unique iff the $\{\lambda_t\}_{t=1}^{d}$ are distinct.
(Nearly) orthogonally decomposable tensors ($d = K$)

**Goal**: Given tensor $\hat{T} \in \mathbb{R}^{d \times d \times d}$ such that $\|\hat{T} - T\| \leq \varepsilon$ for some $T = \sum_{t=1}^{d} \lambda_t \cdot v_t^\otimes 3$ where $\{v_t\}_{t=1}^{d}$ are orthonormal and all $\lambda_t > 0$, approximately recover $\{(v_t, \lambda_t)\}_{t=1}^{d}$.

**Analogous matrix problems**:

- $\varepsilon = 0$: eigendecomposition.
  ("Promised" decomposition always exists by symmetry.)
  Decomposition is unique iff the $\{\lambda_t\}_{t=1}^{d}$ are distinct.

- $\varepsilon > 0$: perturbation theory for eigenvalues (Weyl) and eigenvectors (Davis & Kahan).
Exact orthogonally decomposable tensor
(Zhang & Golub, 2001)

For now assume \( \varepsilon = 0 \), so \( \hat{T} = T \).

Matching moments:

\[
\{(\hat{v}_t, \hat{\lambda}_t)\}_{t=1}^d := \arg \min \left\{ \left\| T - \sum_{t=1}^d \sigma_t \cdot x_t^3 \right\|_F \right\}.
\]
Exact orthogonally decomposable tensor

(Zhang & Golub, 2001)

For now assume $\varepsilon = 0$, so $\hat{T} = T$.

Matching moments:

$$\left\{ (\hat{v}_t, \hat{\lambda}_t) \right\}_{t=1}^{d} := \arg\min_{\left\{ (x_t, \sigma_t) \right\}_{t=1}^{d}} \left\| T - \sum_{t=1}^{d} \sigma_t \cdot x_t \otimes^3 \right\|_F^2.$$

▶ Greedy approach:

▶ Find best rank-1 approximation:

$$(\hat{v}, \hat{\lambda}) := \arg\min_{(x, \sigma) \in S^{d-1} \times \mathbb{R}_+} \left\| T - \sigma \cdot x \otimes^3 \right\|_F^2.$$

▶ “Deflate” $T := T - \hat{\lambda} \cdot \hat{v} \otimes^3$ and repeat.
Exact orthogonally decomposable tensor
(Zhang & Golub, 2001)

For now assume \( \varepsilon = 0 \), so \( \hat{T} = T \).

**Matching moments:**

\[
\{(\hat{v}_t, \hat{\lambda}_t)\}_{t=1}^{d} := \arg\min_{\{(x_t, \sigma_t)\}_{t=1}^{d}} \left\| T - \sum_{t=1}^{d} \sigma_t \cdot x_t \otimes x_t \right\|_F^2.
\]

- **Greedy approach:**
  - Find best rank-1 approximation:
    \[
    \hat{v} := \arg\max_{x \in S^{d-1}} T(x, x, x), \quad \hat{\lambda} := T(\hat{v}, \hat{v}, \hat{v}).
    \]
  - “Deflate” \( T := T - \hat{\lambda} \cdot \hat{v} \otimes 3 \) and repeat.
Claim: Local maximizers of the function

\[ x \mapsto T(x, x, x) = \sum_{i,j,k} T_{i,j,k} \cdot x_i x_j x_k \]

(over the unit ball) are \( \{v_t\}_{t=1}^d \), and

\[ T(v_t, v_t, v_t) = \lambda_t, \quad t \in [d]. \]
Claim: Local maximizers of the function

\[ x \mapsto T(x, x, x) = \sum_{i,j,k} T_{i,j,k} \cdot x_i x_j x_k = \sum_{t=1}^{d} \lambda_t \cdot \langle \nu_t, x \rangle^3 \]

(over the unit ball) are \( \{\nu_t\}_{t=1}^{d} \), and

\[ T(\nu_t, \nu_t, \nu_t) = \lambda_t, \quad t \in [d]. \]
Rank-1 approximation problem

**Claim:** Local maximizers of the function

\[
x \mapsto T(x, x, x) = \sum_{i,j,k} T_{i,j,k} x_i x_j x_k = \sum_{t=1}^{d} \lambda_t \cdot \langle v_t, x \rangle^3
\]

(over the unit ball) are \( \{v_t\}_{t=1}^{d} \), and

\[
T(v_t, v_t, v_t) = \lambda_t , \quad t \in [d] .
\]

**Algorithm:** use gradient ascent to find each component \( v_t \).
Claim: Local maximizers of the function

\[ x \mapsto T(x, x, x) = \sum_{i,j,k} T_{i,j,k} \cdot x_i x_j x_k = \sum_{t=1}^{d} \lambda_t \cdot \langle v_t, x \rangle^3 \]

(over the unit ball) are \( \{v_t\}_{t=1}^{d} \), and

\[ T(v_t, v_t, v_t) = \lambda_t, \quad t \in [d]. \]

Algorithm: use gradient ascent to find each component \( v_t \).

Next: “Parameter-free” fixed-point algorithm.
Fixed-point algorithm
(De Lathauwer, De Moore, & Vandewalle, 2000)

First-order (necessary but not sufficient) optimality condition:

$$\nabla_x T(x, x, x) = \lambda x.$$
Fixed-point algorithm
(De Lathauwer, De Moore, & Vandewalle, 2000)

First-order (necessary but not sufficient) optimality condition:

$$\nabla_x T(x, x, x) = \lambda x.$$

Gradient is “partial evaluation” of $T$:

$$\nabla_x T(x, x, x) = 3 \sum_{i,j} T_{i,j,k} \cdot x_i x_j e_k = 3T(x, x, \cdot).$$
Fixed-point algorithm
(De Lathauwer, De Moore, & Vandewalle, 2000)

First-order (necessary but not sufficient) optimality condition:

\[ \nabla_x T(x, x, x) = \lambda x. \]

Gradient is “partial evaluation” of \( T \):

\[ \nabla_x T(x, x, x) = 3 \sum_{i,j} T_{i,j,k} \cdot x_i x_j e_k = 3T(x, x, \cdot). \]

(Third-order) tensor power iteration:

For \( i = 1, 2, \ldots \):

\[ x^{(i+1)} := \frac{T(x^{(i)}, x^{(i)}, \cdot)}{\|T(x^{(i)}, x^{(i)}, \cdot)\|}. \]
Comparison to matrix power iteration

Matrix power iteration $x^{(i+1)} = \frac{Mx^{(i)}}{\|Mx^{(i)}\|}$ for $M = \sum_t \lambda_t v_t v_t^T$. 

- Requires gap $\min_i i \neq 1 \frac{1 - \lambda_i}{\lambda_1} > 0$ to converge to $v_1$.
- Tensor power iteration: No gap required.
  - If $\langle v_1, x^{(0)} \rangle \neq 0$ (and gap $> 0$), converges to $v_1$.
  - Tensor power iteration: If $t := \arg \max_{t'} \lambda_{t'} |\langle v_{t'}, x^{(0)} \rangle|$, converges to $v_{t'}$.
  - Converges at linear rate.
  - Tensor power iteration: Converges at quadratic rate.
Comparison to matrix power iteration

Matrix power iteration \( x^{(i+1)} = \frac{Mx^{(i)}}{\|Mx^{(i)}\|} \) for \( M = \sum_t \lambda_t v_t v_t^T \).

- Requires gap \( \min_{i \neq 1} 1 - \lambda_i / \lambda_1 > 0 \) to converge to \( v_1 \).
Comparison to matrix power iteration

Matrix power iteration $x^{(i+1)} = \frac{Mx^{(i)}}{\|Mx^{(i)}\|}$ for $M = \sum_t \lambda_t v_t v_t^T$.

- Requires gap $\min_{i \neq 1} 1 - \lambda_i/\lambda_1 > 0$ to converge to $v_1$.

**Tensor power iteration:**

No gap required.
Comparison to matrix power iteration

Matrix power iteration $x^{(i+1)} = \frac{Mx^{(i)}}{\|Mx^{(i)}\|}$ for $M = \sum_t \lambda_t v_t v_t^T$.

- Requires gap $\min_{i \neq 1} 1 - \lambda_i/\lambda_1 > 0$ to converge to $v_1$.

Tensor power iteration:
- No gap required.

- If $\langle v_1, x^{(0)} \rangle \neq 0$ (and gap $> 0$), converges to $v_1$. 
Comparison to matrix power iteration

Matrix power iteration $x^{(i+1)} = \frac{Mx^{(i)}}{\|Mx^{(i)}\|}$ for $M = \sum_t \lambda_t v_t v_t^\top$. 

- Requires gap $\min_{i \neq 1} 1 - \frac{\lambda_i}{\lambda_1} > 0$ to converge to $v_1$. 

Tensor power iteration: 
No gap required.

- If $\langle v_1, x^{(0)} \rangle \neq 0$ (and gap $> 0$), converges to $v_1$. 

Tensor power iteration: 
If $t := \arg \max_{t'} \lambda_{t'} |\langle v_{t'}, x^{(0)} \rangle|$, converges to $v_t$. 

60
Comparison to matrix power iteration

Matrix power iteration $x^{(i+1)} = \frac{Mx^{(i)}}{\|Mx^{(i)}\|}$ for $M = \sum_t \lambda_t v_t v_t^T$.

- Requires gap $\min_{i \neq 1} 1 - \lambda_i / \lambda_1 > 0$ to converge to $v_1$.

Tensor power iteration:
No gap required.

- If $\langle v_1, x^{(0)} \rangle \neq 0$ (and gap $> 0$), converges to $v_1$.

Tensor power iteration:
If $t := \arg \max_{t'} \lambda_{t'} |\langle v_{t'}, x^{(0)} \rangle|$, converges to $v_t$.

- Converges at linear rate.
Comparison to matrix power iteration

Matrix power iteration \( x^{(i+1)} = \frac{Mx^{(i)}}{\|Mx^{(i)}\|} \) for \( M = \sum_t \lambda_t v_t v_t^T \).

- Requires gap \( \min_{i \neq 1} 1 - \frac{\lambda_i}{\lambda_1} > 0 \) to converge to \( v_1 \).

Tensor power iteration:
No gap required.

- If \( \langle v_1, x^{(0)} \rangle \neq 0 \) (and gap > 0), converges to \( v_1 \).

Tensor power iteration:
If \( t := \arg \max_{t'} \lambda_{t'} |\langle v_{t'}, x^{(0)} \rangle| \), converges to \( v_t \).

- Converges at linear rate.

Tensor power iteration:
Converges at quadratic rate.
Nearly orthogonally decomposable tensor
(Mu, H., & Goldfarb, 2015)

Now allow \( \varepsilon = \|E\| > 0 \), for \( E := \hat{T} - T \).

**Claim:** Let \( \hat{v} := \arg \max_{x \in S^{d-1}} \hat{T}(x,x,x) \) and \( \hat{\lambda} := \hat{T}(\hat{v}, \hat{v}, \hat{v}) \).
Then
\[
|\hat{\lambda} - \lambda_t| \leq \varepsilon, \quad \|\hat{v} - v_t\| \leq O \left( \frac{\varepsilon}{\lambda_t} + \left( \frac{\varepsilon}{\lambda_t} \right)^2 \right)
\]
for some \( t \in [d] \) with \( \lambda_t \geq \max_{t'} \lambda_{t'} - 2\varepsilon \).
Nearly orthogonally decomposable tensor
(Mu, H., & Goldfarb, 2015)

Now allow $\varepsilon = \|E\| > 0$, for $E := \hat{T} - T$.

Claim: Let $\hat{v} := \arg \max_{x \in S^{d-1}} \hat{T}(x, x, x)$ and $\hat{\lambda} := \hat{T}(\hat{v}, \hat{v}, \hat{v})$.

Then

$$|\hat{\lambda} - \lambda_t| \leq \varepsilon, \quad \|\hat{v} - v_t\| \leq O \left( \frac{\varepsilon}{\lambda_t} + \left( \frac{\varepsilon}{\lambda_t} \right)^2 \right)$$

for some $t \in [d]$ with $\lambda_t \geq \max_{t'} \lambda_{t'} - 2\varepsilon$.

Many efficient algorithms for solving this approximately, when $\varepsilon$ is small enough, like $1/d$ or $1/\sqrt{d}$ (e.g., Anandkumar, Ge, H., Kakade, & Telgarsky, 2014; Ma, Shi, & Steurer, 2016).
Errors from deflation

(For simplicity, assume $\lambda_t = 1$ for all $t$, so $T = \sum_t \nu_t^{\otimes 3}$.)

First greedy step:
Rank-1 approx. $\hat{v}_1^{\otimes 3}$ to $\hat{T}$ satisfies $\|\hat{v}_1 - v_1\| \leq \varepsilon$ (say).
Errors from deflation

(For simplicity, assume $\lambda_t = 1$ for all $t$, so $T = \sum_t v_t^{\otimes 3}$.)

First greedy step:
Rank-1 approx. $\hat{v}_1^{\otimes 3}$ to $\hat{T}$ satisfies $\|\hat{v}_1 - v_1\| \leq \varepsilon$ (say).

Deflation: To find next $v_t$, use

$$\hat{T} - \hat{v}_1^{\otimes 3} = T + E - \hat{v}_1^{\otimes 3} = \sum_{t=2}^{d} v_t^{\otimes 3} + E + \left( v_1^{\otimes 3} - \hat{v}_1^{\otimes 3} \right).$$
Errors from deflation

(For simplicity, assume $\lambda_t = 1$ for all $t$, so $T = \sum_t v_t \otimes^3$.)

First greedy step:
Rank-1 approx. $\hat{v}_1 \otimes^3$ to $\hat{T}$ satisfies $\| \hat{v}_1 - v_1 \| \leq \varepsilon$ (say).

Deflation: To find next $v_t$, use

$$\hat{T} - \hat{v}_1 \otimes^3 = T + E - \hat{v}_1 \otimes^3$$

$$= \sum_{t=2}^d v_t \otimes^3 + E + (v_1 \otimes^3 - \hat{v}_1 \otimes^3).$$

Now error seems to have doubled (i.e., of size $2\varepsilon$) ...
Effect of deflation errors

For any unit vector \( x \) orthogonal to \( \mathbf{v}_1 \):

\[
\left\| \frac{1}{3} \nabla x \left\{ \left( \mathbf{v}_1 \otimes \mathbf{v}_1 - \hat{\mathbf{v}}_1 \otimes \hat{\mathbf{v}}_1 \right)(x, x, x) \right\} \right\| = \left\| \langle \mathbf{v}_1, x \rangle^2 \mathbf{v}_1 - \langle \hat{\mathbf{v}}_1, x \rangle^2 \hat{\mathbf{v}}_1 \right\|
\]

So effect of errors (original and from deflation) \( E + (\mathbf{v}_1 \otimes \mathbf{v}_1 - \hat{\mathbf{v}}_1 \otimes \hat{\mathbf{v}}_1) \) in directions orthogonal to \( \mathbf{v}_1 \) is \( (1 + o(1)) \varepsilon \) rather than \( 2 \varepsilon \).
Effect of deflation errors

For any unit vector $x$ orthogonal to $v_1$:

$$\left\| \frac{1}{3} \nabla_x \left\{ \left( v_1 \otimes^3 - \hat{v}_1 \otimes^3 \right)(x, x, x) \right\} \right\| = \left\| \left\langle v_1, x \right\rangle^2 v_1 - \left\langle \hat{v}_1, x \right\rangle^2 \hat{v}_1 \right\| = \left\langle \hat{v}_1, x \right\rangle^2$$
Effect of deflation errors

For any unit vector $\mathbf{x}$ orthogonal to $\mathbf{v}_1$:

$$\left\| \frac{1}{3} \nabla \mathbf{x} \left\{ (\mathbf{v}_1 \otimes 3 - \hat{\mathbf{v}}_1 \otimes 3)(\mathbf{x}, \mathbf{x}, \mathbf{x}) \right\} \right\| = \left\| \langle \mathbf{v}_1, \mathbf{x} \rangle^2 \mathbf{v}_1 - \langle \hat{\mathbf{v}}_1, \mathbf{x} \rangle^2 \hat{\mathbf{v}}_1 \right\|$$

$$= \langle \hat{\mathbf{v}}_1, \mathbf{x} \rangle^2$$

$$\leq \left\| \mathbf{v}_1 - \hat{\mathbf{v}}_1 \right\|^2 \leq \varepsilon^2.$$
Effect of deflation errors

For any unit vector $x$ orthogonal to $v_1$:

$$\left\| \frac{1}{3} \nabla_x \left\{ \left( v_1 \otimes^3 - \hat{v}_1 \otimes^3 \right)(x, x, x) \right\} \right\| = \left\| \langle v_1, x \rangle^2 v_1 - \langle \hat{v}_1, x \rangle^2 \hat{v}_1 \right\|$$

$$= \langle \hat{v}_1, x \rangle^2$$

$$\leq \| v_1 - \hat{v}_1 \|^2 \leq \varepsilon^2.$$

So effect of errors (original and from deflation) $E + \left( v_1 \otimes^3 - \hat{v}_1 \otimes^3 \right)$ in directions orthogonal to $v_1$ is $(1 + o(1))\varepsilon$ rather than $2\varepsilon$. 
Effect of deflation errors

For any unit vector $x$ orthogonal to $v_1$:

$$\left\| \frac{1}{3} \nabla x \left\{ \left( v_1 \otimes^3 - \hat{v}_1 \otimes^3 \right)(x, x, x) \right\} \right\| = \left\| \langle v_1, x \rangle^2 v_1 - \langle \hat{v}_1, x \rangle^2 \hat{v}_1 \right\|$$

$$= \langle \hat{v}_1, x \rangle^2$$

$$\leq \| v_1 - \hat{v}_1 \|^2 \leq \varepsilon^2.$$

So effect of errors (original and from deflation) $E + \left( v_1 \otimes^3 - \hat{v}_1 \otimes^3 \right)$ in directions orthogonal to $v_1$ is $(1 + o(1))\varepsilon$ rather than $2\varepsilon$.

- Deflation errors have lower-order effect on finding other $v_t$.

  (Analogous statement for deflation with matrices does not hold.)
Recap

- Reduction to (nearly) orthogonally decomposable tensor permits simple and error-tolerant algorithms.

Lots of on-going work on non-orthogonal / over-complete tensor decompositions (e.g., Goyal, Vempala, & Xiao, 2014; Ge & Ma, 2015; Barak, Kelner, & Steurer, 2015; Ma, Shi, & Steurer, 2016).

- Many similarities to matrix decompositions and algorithms, but differences due to non-linearity are crucial.
Summary

- Using method-of-moments with $O(1)$-order moments, can efficiently estimate parameters for many latent variable models.
  - Exploit distributional properties, multi-view structure, and other structure to determine usable moments tensors.
  - Some efficient algorithms for carrying out the tensor decomposition to obtain parameter estimates.
Summary

- Using method-of-moments with $O(1)$-order moments, can efficiently estimate parameters for many latent variable models.
  - Exploit distributional properties, multi-view structure, and other structure to determine usable moments tensors.
  - Some efficient algorithms for carrying out the tensor decomposition to obtain parameter estimates.

- Many issues to resolve!
  - Handle model misspecification, increase robustness.
  - General methodology.
  - Incorporate general prior knowledge.
  - Incorporate user feedback interactively.
Acknowledgements

Collaborators: Anima Anandkumar (UCI/Amazon), Dean Foster (Amazon), Rong Ge (Duke), Don Goldfarb (Columbia), Sham Kakade (UW), Percy Liang (Stanford), Yi-Kai Liu (NIST), Cun Mu (Columbia), Matus Telgarsky (UIUC), Tong Zhang (Tencent)

Funding: NSF (DMR-1534910, IIS-1563785), Sloan Foundation

Further reading:


Thank you