Learning Mixtures of Spherical Gaussians:

Moment Methods and Spectral Decompositions

Daniel Hsu and Sham M. Kakade

Microsoft Research, New England

Also based on work with Anima Anandkumar (UCI), Rong Ge (Princeton), Matus Telgarsky (UCSD).

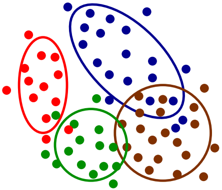
Unsupervised machine learning

- **Many applications in machine learning and statistics**:
  - Lots of high-dimensional data, but mostly unlabeled.

Unsupervised machine learning

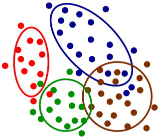
- **Many applications in machine learning and statistics**:
  - Lots of high-dimensional data, but mostly unlabeled.

- **Unsupervised learning**: discover interesting structure of population from unlabeled data.
  - **This talk**: learn about sub-populations in data source.



Learning mixtures of Gaussians

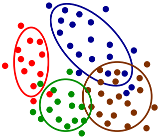
Mixture of Gaussians: $\sum_{i=1}^{k} w_i \, \mathcal{N}(\vec{\mu}_i, \Sigma_i)$



k sub-populations;
each modeled as multivariate Gaussian $\mathcal{N}(\vec{\mu}_i, \Sigma_i)$
together with mixing weight w_i.

Learning mixtures of Gaussians

Mixture of Gaussians: $\sum_{i=1}^{k} w_i \, \mathcal{N}(\vec{\mu}_i, \Sigma_i)$

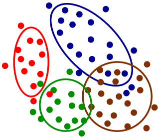


k sub-populations;
each modeled as multivariate Gaussian $\mathcal{N}(\vec{\mu}_i, \Sigma_i)$
together with mixing weight w_i.

Goal: efficient algorithm that approximately recovers parameters from samples.

Learning mixtures of Gaussians

Mixture of Gaussians: $\sum_{i=1}^{k} w_i \, \mathcal{N}(\vec{\mu}_i, \Sigma_i)$



k sub-populations;
each modeled as multivariate Gaussian $\mathcal{N}(\vec{\mu}_i, \Sigma_i)$
together with mixing weight w_i.

Goal: efficient algorithm that approximately recovers parameters from samples.

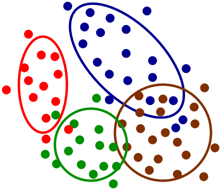
(Alternative goal: density estimation. Not in this talk.)

Learning setup

- **Input**: i.i.d. sample $S \subset \mathbb{R}^d$ from unknown mixtures of Gaussians with parameters $\theta^\star := \{(\vec{\mu}_i{}^\star, \Sigma_i^\star, w_i{}^\star) : i \in [k]\}$.

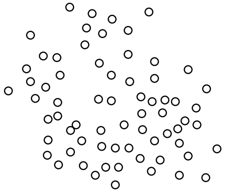
Learning setup

- **Input**: i.i.d. sample $S \subset \mathbb{R}^d$ from unknown mixtures of Gaussians with parameters $\theta^\star := \{(\vec{\mu}_i{}^\star, \Sigma_i^\star, w_i{}^\star) : i \in [k]\}$.
- Each data point drawn from one of k Gaussians $\mathcal{N}(\vec{\mu}_i{}^\star, \Sigma_i^\star)$ (choose $\mathcal{N}(\vec{\mu}_i{}^\star, \Sigma_i^\star)$ with probability $w_i{}^\star$.)



Learning setup

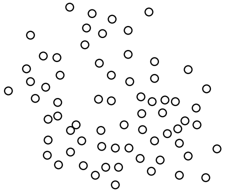
- **Input**: i.i.d. sample $S \subset \mathbb{R}^d$ from unknown mixtures of Gaussians with parameters $\theta^\star := \{(\vec{\mu}_i{}^\star, \Sigma_i^\star, w_i{}^\star) : i \in [k]\}$.
- Each data point drawn from one of k Gaussians $\mathcal{N}(\vec{\mu}_i{}^\star, \Sigma_i^\star)$ (choose $\mathcal{N}(\vec{\mu}_i{}^\star, \Sigma_i^\star)$ with probability $w_i{}^\star$.)



- But **"labels" are not observed.**

Learning setup

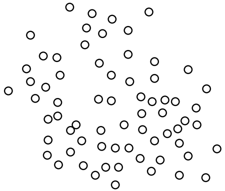
- **Input**: i.i.d. sample $S \subset \mathbb{R}^d$ from unknown mixtures of Gaussians with parameters $\theta^\star := \{(\vec{\mu}_i^\star, \Sigma_i^\star, w_i^\star) : i \in [k]\}$.
- Each data point drawn from one of k Gaussians $\mathcal{N}(\vec{\mu}_i^\star, \Sigma_i^\star)$ (choose $\mathcal{N}(\vec{\mu}_i^\star, \Sigma_i^\star)$ with probability $w_i^\star$.)



- But **"labels" are not observed.**
- **Goal**: estimate parameters $\theta = \{(\vec{\mu}_i, \Sigma_i, w_i) : i \in [k]\}$ such that $\theta \approx \theta^\star$.

Learning setup

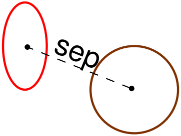
- **Input**: i.i.d. sample $S \subset \mathbb{R}^d$ from unknown mixtures of Gaussians with parameters $\theta^\star := \{(\vec{\mu}_i{}^\star, \Sigma_i^\star, w_i{}^\star) : i \in [k]\}$.
- Each data point drawn from one of k Gaussians $\mathcal{N}(\vec{\mu}_i{}^\star, \Sigma_i^\star)$ (choose $\mathcal{N}(\vec{\mu}_i{}^\star, \Sigma_i^\star)$ with probability $w_i{}^\star$.)



- But **"labels" are not observed.**
- **Goal**: estimate parameters $\theta = \{(\vec{\mu}_i, \Sigma_i, w_i) : i \in [k]\}$ such that $\theta \approx \theta^\star$.
- **In practice**: local search for maximum-likelihood parameters (E-M algorithm).

When are there efficient algorithms?

Well-separated mixtures: estimation is easier if there is large minimum separation between component means (Dasgupta, '99):

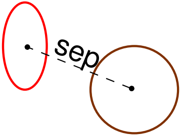


$$\text{sep} := \min_{i \neq j} \frac{\|\vec{\mu}_i - \vec{\mu}_j\|}{\max\{\sigma_i, \sigma_j\}}.$$

- $\text{sep} = \Omega(d^c)$ or $\text{sep} = \Omega(k^c)$: simple clustering methods, perhaps after dimension reduction
  (Dasgupta, '99; Vempala-Wang, '02; and many more.)

When are there efficient algorithms?

Well-separated mixtures: estimation is easier if there is large minimum separation between component means (Dasgupta, '99):



$$\text{sep} := \min_{i \neq j} \frac{\|\vec{\mu}_i - \vec{\mu}_j\|}{\max\{\sigma_i, \sigma_j\}}.$$

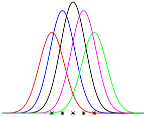
- $\text{sep} = \Omega(d^c)$ or $\text{sep} = \Omega(k^c)$: simple clustering methods, perhaps after dimension reduction
  (Dasgupta, '99; Vempala-Wang, '02; and many more.)

Recent developments:

- No minimum separation requirement, but current methods require $\exp(\Omega(k))$ running time / sample size
  (Kalai-Moitra-Valiant, '10; Belkin-Sinha, '10; Moitra-Valiant, '10)

Overcoming barriers to efficient estimation

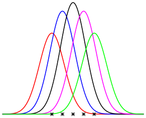
Information-theoretic barrier:



Gaussian mixtures in $\mathbb{R}^1$ can require $\exp(\Omega(k))$ samples to estimate parameters, even when components are well-separated (Moitra-Valiant, '10).

Overcoming barriers to efficient estimation

Information-theoretic barrier:

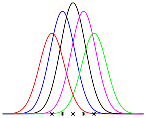


Gaussian mixtures in $\mathbb{R}^1$ can require $\exp(\Omega(k))$ samples to estimate parameters, even when components are well-separated (Moitra-Valiant, '10).

These hard instances are degenerate in high-dimensions!

Overcoming barriers to efficient estimation

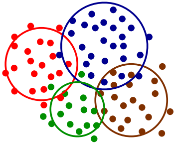
Information-theoretic barrier:



Gaussian mixtures in $\mathbb{R}^1$ can require $\exp(\Omega(k))$ samples to estimate parameters, even when components are well-separated (Moitra-Valiant, '10).

These hard instances are degenerate in high-dimensions!

Our result: efficient algorithms for *non-degenerate* models in high-dimensions ($d \geq k$) with *spherical covariances*.



Main result

Theorem (H-Kakade, '13)

Assume $\{\vec{\mu}_1{}^\star, \vec{\mu}_2{}^\star, \dots, \vec{\mu}_k{}^\star\}$ *linearly independent,* $w_i{}^\star > 0$ *for all* $i \in [k]$*, and* $\Sigma_i^\star = \sigma_i^{2\star}\mathrm{I}$ *for all* $i \in [k]$*.*

There is an algorithm that, given independent draws from a mixture of k *spherical Gaussians, returns* ε*-accurate parameters (up to permutation, under* ℓ^2 *metric) w.h.p.*

The running time and sample complexity are

$$\mathrm{poly}(d, k, 1/\varepsilon, 1/w_{\min}, 1/\lambda_{\min})$$

where $\lambda_{\min} = k^{th}$*-largest singular value of* $[\vec{\mu}_1{}^\star | \vec{\mu}_2{}^\star | \cdots | \vec{\mu}_k{}^\star]$*.*

(Also using new techniques from Anandkumar-Ge-H-Kakade-Telgarsky, '12.)

2. Learning algorithm

Method-of-moments

Let $S \subset \mathbb{R}^d$ be an i.i.d. sample from an unknown mixture of spherical Gaussians:

$$\vec{x} \sim \sum_{i=1}^{k} w_i^{\star} \mathcal{N}(\vec{\mu}_i^{\star}, \sigma_i^{2\star} \mathrm{I}).$$

Method-of-moments

Let $S \subset \mathbb{R}^d$ be an i.i.d. sample from an unknown mixture of spherical Gaussians:

$$\vec{x} \sim \sum_{i=1}^{k} w_i^{\star} \mathcal{N}(\vec{\mu}_i^{\star}, \sigma_i^{2\star} \mathrm{I}).$$

Estimation via method-of-moments (Pearson, 1894)

Find parameters θ such that

$$\mathbb{E}_{\theta}[\, p(\vec{x}) \,] \approx \hat{\mathbb{E}}_{\vec{x} \in S}[\, p(\vec{x}) \,]$$

for some functions $p : \mathbb{R}^d \to \mathbb{R}$ (typically multivar. polynomials).

Method-of-moments

Let $S \subset \mathbb{R}^d$ be an i.i.d. sample from an unknown mixture of spherical Gaussians:

$$\vec{x} \sim \sum_{i=1}^{k} w_i^{\star} \mathcal{N}(\vec{\mu}_i^{\star}, \sigma_i^{2\star} \mathrm{I}).$$

Estimation via method-of-moments (Pearson, 1894)

Find parameters θ such that

$$\mathbb{E}_{\theta}[\, p(\vec{x}) \,] \approx \hat{\mathbb{E}}_{\vec{x} \in S}[\, p(\vec{x}) \,]$$

for some functions $p : \mathbb{R}^d \to \mathbb{R}$ (typically multivar. polynomials).

Q1 Which moments to use?

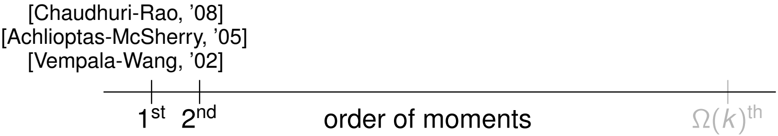
Q2 How to (approx.) solve moment equations?

Which moments to use?

Which moments to use?

moment order	**reliable estimates?**	**unique solution?**
1^{st}, 2^{nd}		

1^{st}- and 2^{nd}-order moments (*e.g.*, mean, covariance)



Which moments to use?

moment order	reliable estimates?	unique solution?
1st, 2nd	✓	

1st- and 2nd-order moments (*e.g.*, mean, covariance)

- Fairly easy to get reliable estimates.

$$\mathbb{E}_{\vec{x} \in S}[\vec{x} \otimes \vec{x}] \approx \mathbb{E}_{\theta^\star}[\vec{x} \otimes \vec{x}]$$

[Chaudhuri-Rao, '08]
[Achlioptas-McSherry, '05]
[Vempala-Wang, '02]

1st 2nd order of moments $\Omega(k)^{\text{th}}$

Which moments to use?

moment order	reliable estimates?	unique solution?
1^{st}, 2^{nd}	✓	✗

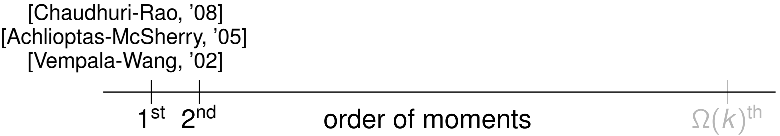
1^{st}- and 2^{nd}-order moments (*e.g.*, mean, covariance)

- Fairly easy to get reliable estimates.

$$\mathbb{E}_{\vec{x} \in S}[\vec{x} \otimes \vec{x}] \approx \mathbb{E}_{\theta^\star}[\vec{x} \otimes \vec{x}]$$

- Can have multiple solutions to moment equations.

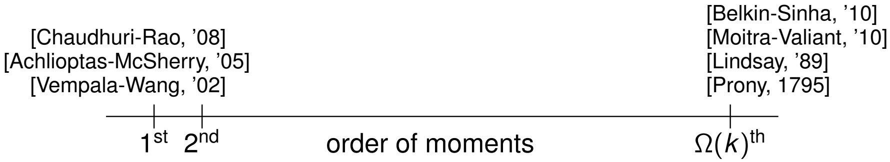
$$\mathbb{E}_{\theta_1}[\vec{x} \otimes \vec{x}] \approx \mathbb{E}_{\vec{x} \in S}[\vec{x} \otimes \vec{x}] \approx \mathbb{E}_{\theta_2}[\vec{x} \otimes \vec{x}], \quad \theta_1 \neq \theta_2$$



Which moments to use?

moment order	reliable estimates?	unique solution?
1st, 2nd	✓	✗
$\Omega(k)^{\text{th}}$		

$\Omega(k)^{\text{th}}$**-order moments** (*e.g.*, $\mathbb{E}_\theta$[degree-k-poly($\vec{x}$)])

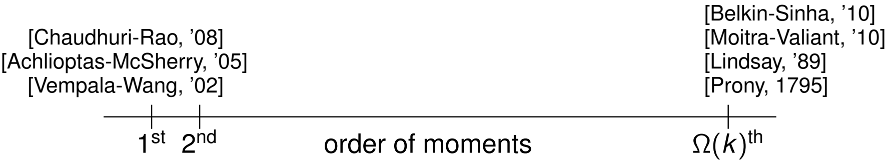


Which moments to use?

moment order	reliable estimates?	unique solution?
1^{st}, 2^{nd}	✓	✗
$\Omega(k)^{th}$		✓

$\Omega(k)^{\text{th}}$**-order moments** (*e.g.*, $\mathbb{E}_\theta$[degree-k-poly($\vec{x}$)])

- Uniquely pins down the solution.

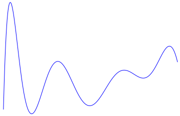


Which moments to use?

moment order	reliable estimates?	unique solution?
1^{st}, 2^{nd}	✓	✗
$\Omega(k)^{th}$	✗	✓

$\Omega(k)^{\mathbf{th}}$**-order moments** (*e.g.*, $\mathbb{E}_\theta$[degree-k-poly($\vec{x}$)])

- Uniquely pins down the solution.
- Empirical estimates very unreliable.



[Chaudhuri-Rao, '08]
[Achlioptas-McSherry, '05]
[Vempala-Wang, '02]

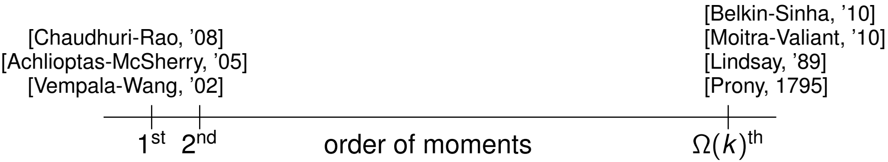
[Belkin-Sinha, '10]
[Moitra-Valiant, '10]
[Lindsay, '89]
[Prony, 1795]

1^{st} 2^{nd} order of moments $\Omega(k)^{th}$

Which moments to use?

moment order	reliable estimates?	unique solution?
1^{st}, 2^{nd}	✓	✗
$\Omega(k)^{th}$	✗	✓

Can we get best-of-both-worlds?

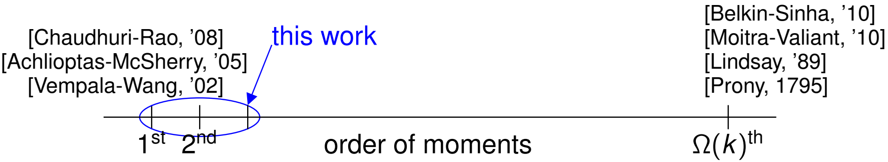


Which moments to use?

moment order	reliable estimates?	unique solution?
1^{st}, 2^{nd}	✓	✗
$\Omega(k)^{th}$	✗	✓

Can we get best-of-both-worlds? **Yes!**

In high-dimensions ($d \geq k$),
low-order multivariate moments suffice.
(1^{st}-, 2^{nd}-, and 3^{rd}-order moments)



Structure of low-order multivariate moments

Second- and third-order multivariate moments:

$$\mathbb{E}_\theta[\vec{x} \otimes \vec{x}] = \sum_{i=1}^{k} w_i \, \vec{\mu}_i \otimes \vec{\mu}_i + \text{some sparse matrix};$$

$$\mathbb{E}_\theta[\vec{x} \otimes \vec{x} \otimes \vec{x}] = \sum_{i=1}^{k} w_i \, \vec{\mu}_i \otimes \vec{\mu}_i \otimes \vec{\mu}_i + \text{some sparse tensor}.$$

Structure of low-order multivariate moments

Second- and third-order multivariate moments:

$$\mathbb{E}_\theta[\vec{x} \otimes \vec{x}] = \sum_{i=1}^{k} w_i \, \vec{\mu}_i \otimes \vec{\mu}_i + \text{some sparse matrix};$$

$$\mathbb{E}_\theta[\vec{x} \otimes \vec{x} \otimes \vec{x}] = \sum_{i=1}^{k} w_i \, \vec{\mu}_i \otimes \vec{\mu}_i \otimes \vec{\mu}_i + \text{some sparse tensor}.$$

Trick: "sparse stuff" can be estimated and thus removed**.**

Structure of low-order multivariate moments

Second- and third-order multivariate moments:

$$\mathbb{E}_\theta[\vec{x} \otimes \vec{x}] \;=\; \sum_{i=1}^{k} w_i \; \vec{\mu}_i \otimes \vec{\mu}_i \;+\; \text{some sparse matrix};$$

$$\mathbb{E}_\theta[\vec{x} \otimes \vec{x} \otimes \vec{x}] \;=\; \sum_{i=1}^{k} w_i \; \vec{\mu}_i \otimes \vec{\mu}_i \otimes \vec{\mu}_i \;+\; \text{some sparse tensor}.$$

Trick: "sparse stuff" can be estimated and thus removed.

Upshot: the following can be readily estimated (with $\widehat{M}$, $\widehat{T}$).

$$M_{\theta^\star} := \sum_{i=1}^{k} w_i^\star \; \vec{\mu}_i^\star \otimes \vec{\mu}_i^\star \quad \text{and} \quad T_{\theta^\star} := \sum_{i=1}^{k} w_i^\star \; \vec{\mu}_i^\star \otimes \vec{\mu}_i^\star \otimes \vec{\mu}_i^\star .$$

Structure of low-order multivariate moments

Second- and third-order multivariate moments:

$$\mathbb{E}_\theta[\vec{x} \otimes \vec{x}] = \sum_{i=1}^{k} w_i \, \vec{\mu}_i \otimes \vec{\mu}_i + \text{some sparse matrix};$$

$$\mathbb{E}_\theta[\vec{x} \otimes \vec{x} \otimes \vec{x}] = \sum_{i=1}^{k} w_i \, \vec{\mu}_i \otimes \vec{\mu}_i \otimes \vec{\mu}_i + \text{some sparse tensor}.$$

Trick: "sparse stuff" can be estimated and thus removed.

Upshot: the following can be readily estimated (with $\widehat{M}$, $\widehat{T}$).

$$M_{\theta^\star} := \sum_{i=1}^{k} {w_i}^\star \, {\vec{\mu}_i}^\star \otimes {\vec{\mu}_i}^\star \quad \text{and} \quad T_{\theta^\star} := \sum_{i=1}^{k} {w_i}^\star \, {\vec{\mu}_i}^\star \otimes {\vec{\mu}_i}^\star \otimes {\vec{\mu}_i}^\star.$$

Claim: $\{(\vec{\mu}_i, w_i)\}$ uniquely determined by M_θ and T_θ.

Variational argument for parameter uniquness

View $M_\theta : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ and $T_\theta : \mathbb{R}^d \times \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$
as bi-linear and tri-linear functions.

Variational argument for parameter uniquness

View $M_\theta : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ and $T_\theta : \mathbb{R}^d \times \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ as bi-linear and tri-linear functions.

Lemma

If $\{\vec{\mu}_i\}$ are linearly independent and all $w_i > 0$, then each of the k distinct, isolated local maximizers $\vec{u}^*$ of

$$\max_{\vec{u} \in \mathbb{R}^d} T_\theta(\vec{u}, \vec{u}, \vec{u}) \quad s.t. \quad M_\theta(\vec{u}, \vec{u}) \le 1$$

satisfies, for some $i \in [k]$,

$$M_\theta(\cdot, \vec{u}^*) = \sqrt{w_i}\, \vec{\mu}_i, \qquad T_\theta(\vec{u}^*, \vec{u}^*, \vec{u}^*) = \frac{1}{\sqrt{w_i}}.$$

Variational argument for parameter uniquness

View $M_\theta : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ and $T_\theta : \mathbb{R}^d \times \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ as bi-linear and tri-linear functions.

Lemma

If $\{\vec{\mu}_i\}$ are linearly independent and all $w_i > 0$, then each of the k distinct, isolated local maximizers $\vec{u}^*$ of

$$\max_{\vec{u} \in \mathbb{R}^d} T_\theta(\vec{u}, \vec{u}, \vec{u}) \quad s.t. \quad M_\theta(\vec{u}, \vec{u}) \le 1$$

satisfies, for some $i \in [k]$,

$$M_\theta(\cdot, \vec{u}^*) = \sqrt{w_i}\, \vec{\mu}_i, \qquad T_\theta(\vec{u}^*, \vec{u}^*, \vec{u}^*) = \frac{1}{\sqrt{w_i}}.$$

$\therefore \{(\vec{\mu}_i, w_i) : i \in [k]\}$ uniquely determined by M_θ, T_θ. ■

Main idea for variational lemma

$$\max_{\vec{u} \in \mathbb{R}^d} T_{\theta}(\vec{u}, \vec{u}, \vec{u}) \text{ s.t. } M_{\theta}(\vec{u}, \vec{u}) \leq 1$$

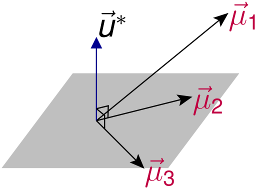
Main idea for variational lemma

$$\max_{\vec{u}\in\mathbb{R}^d} \sum_{i=1}^{k} w_i \, \langle \vec{\mu}_i, \vec{u} \rangle^3 \quad \text{s.t.} \quad \sum_{i=1}^{k} w_i \, \langle \vec{\mu}_i, \vec{u} \rangle^2 \leq 1$$

Main idea for variational lemma

$$\max_{\vec{u}\in\mathbb{R}^d} \sum_{i=1}^{k} w_i \, \langle \vec{\mu}_i, \vec{u} \rangle^3 \quad \text{s.t.} \quad \sum_{i=1}^{k} w_i \, \langle \vec{\mu}_i, \vec{u} \rangle^2 \leq 1$$

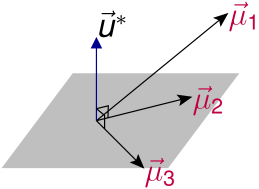
Maximizers are directions $\vec{u}^*$ orthogonal to all but one $\vec{\mu}_j$.



Main idea for variational lemma

$$\max_{\vec{u} \in \mathbb{R}^d} \sum_{i=1}^{k} w_i \langle \vec{\mu}_i, \vec{u} \rangle^3 \quad \text{s.t.} \quad \sum_{i=1}^{k} w_i \langle \vec{\mu}_i, \vec{u} \rangle^2 \leq 1$$

Maximizers are directions $\vec{u}^*$ orthogonal to all but one $\vec{\mu}_j$.



Combine with constraints $w_j \langle \vec{\mu}_j, \vec{u}^* \rangle^2 \leq 1$ to get

$$M\vec{u}^* = \left(\sum_{i=1}^{k} w_i \, \vec{\mu}_i \otimes \vec{\mu}_i \right) \vec{u}^* = \sum_{i=1}^{k} w_i \, \vec{\mu}_i \langle \vec{\mu}_i, \vec{u}^* \rangle = \pm \sqrt{w_j} \, \vec{\mu}_j. \quad \blacksquare$$

How to solve the moment equations?

Effectively want to solve

$$\min_{\theta} \|T_{\theta} - \widehat{T}\|^2 \quad \text{s.t.} \quad M_{\theta} = \widehat{M}. \tag{†}$$

How to solve the moment equations?

Effectively want to solve

$$\min_{\theta} \| T_{\theta} - \widehat{T} \|^2 \quad \text{s.t.} \quad M_{\theta} = \widehat{M}. \qquad (\dagger)$$

Not convex in parameters $\theta = \{(\vec{\mu}_i, w_i)\}$**.**

How to solve the moment equations?

Effectively want to solve

$$\min_{\theta} \| T_{\theta} - \widehat{T} \|^2 \quad \text{s.t.} \quad M_{\theta} = \widehat{M}. \tag{†}$$

Not convex in parameters $\theta = \{(\vec{\mu}_i, w_i)\}$.

What we do: find one component $(\vec{\mu}_i, w_i)$ at a time, using local optimization of related (also non-convex) objective function.

$$\max_{\vec{u} \in \mathbb{R}^d} \sum_{i,j,k} \widehat{T}_{i,j,k} \, u_i u_j u_k \quad \text{s.t.} \quad \sum_{i,j} \widehat{M}_{i,j} \, u_i u_j \leq 1 \tag{‡}$$

How to solve the moment equations?

Effectively want to solve

$$\min_{\theta} \| T_{\theta} - \widehat{T} \|^2 \quad \text{s.t.} \quad M_{\theta} = \widehat{M}. \qquad (\dagger)$$

Not convex in parameters $\theta = \{(\vec{\mu}_i, w_i)\}$.

What we do: find one component $(\vec{\mu}_i, w_i)$ at a time, using local optimization of related (also non-convex) objective function.

$$\max_{\vec{u} \in \mathbb{R}^d} \widehat{T}(\vec{u}, \vec{u}, \vec{u}) \quad \text{s.t.} \quad \widehat{M}(\vec{u}, \vec{u}) \leq 1 \qquad (\ddagger)$$

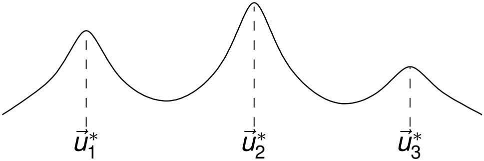
How to solve the moment equations?

Effectively want to solve

$$\min_{\theta} \| T_{\theta} - \widehat{T} \|^2 \quad \text{s.t.} \quad M_{\theta} = \widehat{M}. \qquad (\dagger)$$

Not convex in parameters $\theta = \{(\vec{\mu}_i, w_i)\}$.

What we do: find one component $(\vec{\mu}_i, w_i)$ at a time, using local optimization of related (also non-convex) objective function.



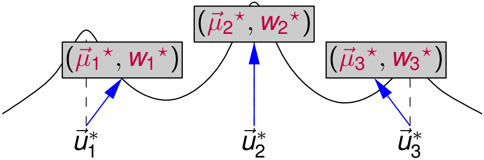
How to solve the moment equations?

Effectively want to solve

$$\min_{\theta} \| T_{\theta} - \widehat{T} \|^2 \quad \text{s.t.} \quad M_{\theta} = \widehat{M}. \qquad (\dagger)$$

Not convex in parameters $\theta = \{(\vec{\mu}_i, w_i)\}$.

What we do: find one component $(\vec{\mu}_i, w_i)$ at a time, using local optimization of related (also non-convex) objective function.



New robust algorithm for "tensor eigen-decomposition" efficiently approximates *all* local optima, each corresponding to a component. $\longrightarrow$ Near-optimal solution to ($\dagger$). ∎

Local optimization

Want to find *all* local maximizers of

$$\max_{\vec{u} \in \mathbb{R}^d} \quad \widehat{T}(\vec{u}, \vec{u}, \vec{u}) \quad \text{s.t.} \quad \widehat{M}(\vec{u}, \vec{u}) \leq 1. \qquad (\ddagger)$$

Must address **initialization** and **convergence** issues.

Local optimization

Want to find *all* local maximizers of

$$\max_{\vec{u} \in \mathbb{R}^d} \quad \widehat{T}(\vec{u}, \vec{u}, \vec{u}) \quad \text{s.t.} \quad \widehat{M}(\vec{u}, \vec{u}) \leq 1. \qquad (\ddagger)$$

Must address **initialization** and **convergence** issues.

Crucially using special tensor structure of $\widehat{T} \approx T_{\theta^\star}$,
together with non-linearity of $\vec{u} \mapsto \widehat{T}(\,\cdot\,, \vec{u}, \vec{u})$:

- Random initialization is good with significant probability.
  ("Good" $\Rightarrow$ simple iteration will quickly converge to some local max.)

Local optimization

Want to find *all* local maximizers of

$$\max_{\vec{u} \in \mathbb{R}^d} \quad \widehat{T}(\vec{u}, \vec{u}, \vec{u}) \quad \text{s.t.} \quad \widehat{M}(\vec{u}, \vec{u}) \leq 1. \qquad (\ddagger)$$

Must address **initialization** and **convergence** issues.

Crucially using special tensor structure of $\widehat{T} \approx T_{\theta^\star}$, together with non-linearity of $\vec{u} \mapsto \widehat{T}(\,\cdot\,, \vec{u}, \vec{u})$:

- Random initialization is good with significant probability.
  ("Good" $\Rightarrow$ simple iteration will quickly converge to some local max.)
- Can check if initialization was good by checking objective value after a few steps.

Local optimization

Want to find *all* local maximizers of

$$\max_{\vec{u} \in \mathbb{R}^d} \quad \widehat{T}(\vec{u}, \vec{u}, \vec{u}) \quad \text{s.t.} \quad \widehat{M}(\vec{u}, \vec{u}) \leq 1. \qquad (\ddagger)$$

Must address **initialization** and **convergence** issues.

Crucially using special tensor structure of $\widehat{T} \approx T_{\theta^\star}$, together with non-linearity of $\vec{u} \mapsto \widehat{T}(\,\cdot\,, \vec{u}, \vec{u})$:

- Random initialization is good with significant probability.
  ("Good" $\Rightarrow$ simple iteration will quickly converge to some local max.)
- Can check if initialization was good by checking objective value after a few steps.
  - If value large enough: initialization was good; improve by taking a few more steps.

Local optimization

Want to find *all* local maximizers of

$$\max_{\vec{u} \in \mathbb{R}^d} \ \widehat{T}(\vec{u}, \vec{u}, \vec{u}) \quad \text{s.t.} \quad \widehat{M}(\vec{u}, \vec{u}) \leq 1. \qquad (\ddagger)$$

Must address **initialization** and **convergence** issues.

Crucially using special tensor structure of $\widehat{T} \approx T_{\theta^\star}$,
together with non-linearity of $\vec{u} \mapsto \widehat{T}(\,\cdot\,, \vec{u}, \vec{u})$:

- Random initialization is good with significant probability.
  ("Good" $\Rightarrow$ simple iteration will quickly converge to some local max.)
- Can check if initialization was good by checking objective value after a few steps.
  - If value large enough: initialization was good; improve by taking a few more steps.
  - Else: abandon and restart.

3. Concluding remarks

Some open problems

Some open problems

- Can also handle mixtures of Gaussians with somewhat more general covariances, under incoherence conditions

$$\mathbb{E}_\theta[\vec{x} \otimes \vec{x}] = \underbrace{\sum_{i=1}^{k} w_i \, \vec{\mu}_i \otimes \vec{\mu}_i}_{\text{low-rank}} \; + \; \text{some sparse matrix}$$

Some open problems

- Can also handle mixtures of Gaussians with somewhat more general covariances, under incoherence conditions

$$\mathbb{E}_\theta[\vec{x} \otimes \vec{x}] = \underbrace{\sum_{i=1}^{k} w_i \, \vec{\mu}_i \otimes \vec{\mu}_i}_{\text{low-rank}} \; + \; \text{some sparse matrix}$$

- **Question #1**: What about mixtures of Gaussians with arbitrary covariances?

Some open problems

- Can also handle mixtures of Gaussians with somewhat more general covariances, under incoherence conditions

$$\mathbb{E}_\theta[\vec{x} \otimes \vec{x}] = \underbrace{\sum_{i=1}^{k} w_i \, \vec{\mu}_i \otimes \vec{\mu}_i}_{\text{low-rank}} \; + \; \text{some sparse matrix}$$

- **Question #1**: What about mixtures of Gaussians with arbitrary covariances?
- **Question #2**: How to handle degenerate cases / $k \gg d$?
  (Practical relevance: automatic speech recognition)

Summary

Summary

- **Learning mixtures of spherical Gaussians**: worst-case (information-theoretically) hard, but non-degenerate cases are easy.

Summary

- **Learning mixtures of spherical Gaussians**: worst-case (information-theoretically) hard, but non-degenerate cases are easy.
  - Structure in low-order multivariate moments uniquely determines model parameters under natural non-degeneracy condition;

    $\Rightarrow$ permits computationally efficient algorithm for estimation.

Summary

- **Learning mixtures of spherical Gaussians**: worst-case (information-theoretically) hard, but non-degenerate cases are easy.
  - Structure in low-order multivariate moments uniquely determines model parameters under natural non-degeneracy condition;

    $\Rightarrow$ permits computationally efficient algorithm for estimation.
  - Similar story for many other statistical models (*e.g.*, HMMs (Mossel-Roch, '06; H-Kakade-Zhang, '09), topic models (Arora-Ge-Moitra, '12; Anandkumar *et al*, '12), ICA (Arora *et al*, '12)).

Summary

- **Learning mixtures of spherical Gaussians**:
  worst-case (information-theoretically) hard, but
  non-degenerate cases are easy.
  - Structure in low-order multivariate moments uniquely determines model parameters under natural non-degeneracy condition;

    $\Rightarrow$ permits computationally efficient algorithm for estimation.
  - Similar story for many other statistical models
    (*e.g.*, HMMs (Mossel-Roch, '06; H-Kakade-Zhang, '09),
    topic models (Arora-Ge-Moitra, '12; Anandkumar *et al*, '12),
    ICA (Arora *et al*, '12)).
- **Open problem**: efficient estimators for highly over-complete and general mixture models ($k \gg d$).

Thanks!

Related survey/overview-ish paper:

- Tensor decompositions for latent variable models (with Anandkumar, Ge, Kakade, and Telgarsky): **`http://arxiv.org/abs/1210.7559`**

Structure of low-order moments

- **First-order moments**:

$$\mathbb{E}[\vec{x}] = \sum_{i=1}^{k} w_i \, \vec{\mu}_i .$$

- **Second-order moments**:

$$\mathbb{E}[\vec{x} \otimes \vec{x}] = \sum_{i=1}^{k} w_i \, \vec{\mu}_i \otimes \vec{\mu}_i + \bar{\sigma}^2 \mathrm{I}$$

where $\bar{\sigma}^2 := \sum_{i=1}^{k} w_i \, \sigma_i^2$.

Fact: $\bar{\sigma}^2$ is the smallest eigenvalue of $\mathrm{Cov}(\vec{x}) = \mathbb{E}[\vec{x} \otimes \vec{x}] - \mathbb{E}[\vec{x}] \otimes \mathbb{E}[\vec{x}]$.

Structure of low-order moments

- **Third-order moments**:

$$\mathbb{E}[\vec{x} \otimes \vec{x} \otimes \vec{x}] = \sum_{i=1}^{k} w_i \, \vec{\mu}_i \otimes \vec{\mu}_i \otimes \vec{\mu}_i + \sum_{i=1}^{d} \vec{m} \otimes e_i \otimes e_i + e_i \otimes \vec{m} \otimes e_i + e_i \otimes e_i \otimes \vec{m}$$

where $\vec{m} := \sum_{i=1}^{k} w_i \, \sigma_i^2 \vec{\mu}_i$.

Fact: $\vec{m} = \mathbb{E}[\, (\vec{u}^\top(\vec{x} - \mathbb{E}[\vec{x}]))^2 \, \vec{x} \,]$ for any unit-norm eigenvector $\vec{u}$ of $\mathrm{Cov}(\vec{x})$ corresponding to eigenvalue $\bar{\sigma}^2$.

Proof idea for optimization lemma

$$\max_{\vec{u} \in \mathbb{R}^d} T(\vec{u}, \vec{u}, \vec{u}) \;\; \text{s.t.} \;\; M(\vec{u}, \vec{u}) \leq 1$$

Proof idea for optimization lemma

$$\max_{\vec{u} \in \mathbb{R}^d} \sum_{i=1}^{k} w_i \langle \vec{\mu}_i, \vec{u} \rangle^3 \;\; \text{s.t.} \;\; \sum_{i=1}^{k} w_i \langle \vec{\mu}_i, \vec{u} \rangle^2 \leq 1$$

Proof idea for optimization lemma

$$\max_{\vec{\theta} \in \mathbb{R}^k} \sum_{i=1}^{k} \frac{1}{\sqrt{w_i}} \theta_i^3 \quad \text{s.t.} \quad \sum_{i=1}^{k} \theta_i^2 \leq 1$$

$$(\theta_i := \sqrt{w_i} \langle \vec{\mu}_i, \vec{u} \rangle.)$$

Proof idea for optimization lemma

$$\max_{\vec{\theta} \in \mathbb{R}^k} \sum_{i=1}^{k} \frac{1}{\sqrt{w_i}} \theta_i^3 \quad \text{s.t.} \quad \sum_{i=1}^{k} \theta_i^2 \leq 1$$

$$(\theta_i := \sqrt{w_i} \langle \vec{\mu}_i, \vec{u} \rangle.)$$

Isolated local maxima are $\frac{1}{\sqrt{w_1}}, \frac{1}{\sqrt{w_2}}, \dots$, achieved at

$$(1, 0, 0, \dots), \quad (0, 1, 0, \dots), \quad \dots$$

Proof idea for optimization lemma

$$\max_{\vec{\theta} \in \mathbb{R}^k} \sum_{i=1}^{k} \frac{1}{\sqrt{w_i}} \theta_i^3 \quad \text{s.t.} \quad \sum_{i=1}^{k} \theta_i^2 \leq 1$$

$$(\theta_i := \sqrt{w_i} \langle \vec{\mu}_i, \vec{u} \rangle.)$$

Isolated local maxima are $\frac{1}{\sqrt{w_1}}, \frac{1}{\sqrt{w_2}}, \dots,$ achieved at

$$(1, 0, 0, \dots), \quad (0, 1, 0, \dots), \quad \dots$$

Translates to directions $\vec{u}^*$ orthogonal to all but one $\vec{\mu}_j$.