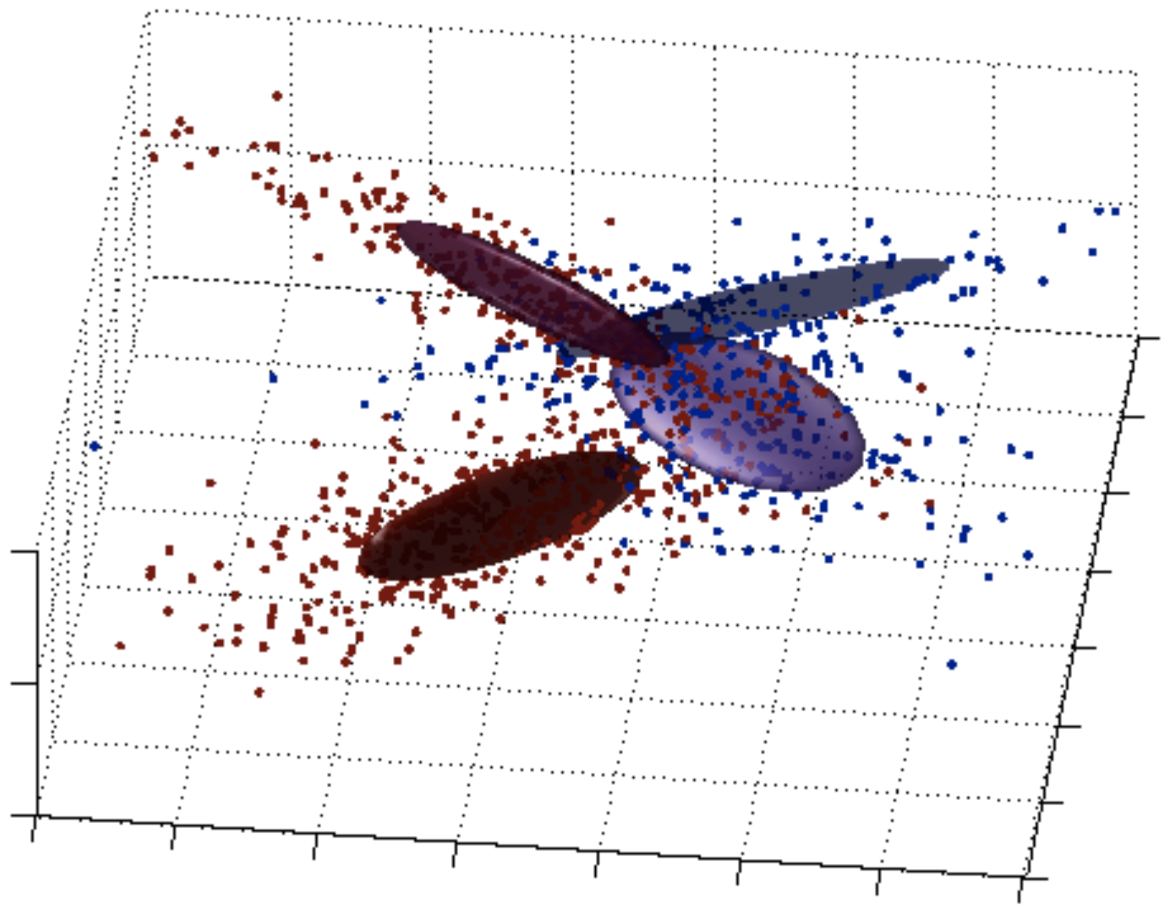
Learning Mixture of Gaussians in High Dimensions

Rong Ge, Qingqing Huang, Sham Kakade
(MSR-NE) (MIT) (MSR-NE)

STOC 2015

Motivation



- Input: multi-dimensional data points
- Assumption: mixture of Gaussian distributions
- Goal: learn weights, means, covariance matrices
- Widely used model in machine learning

Problem statement

- n-dimensional k-component

  Parameters: weights w_i , means $\mu^{(i)}$, covariance matrices $\Sigma^{(i)}$

  MoG sample generation $x = \mathcal{N}(\mu^{(i)}, \Sigma^{(i)}), \quad i \sim w_i$

Can we learn the parameters with **poly algorithm** for every MoG ?

Problem statement

- n -dimensional k -component

  Parameters: weights w_i , means $\mu^{(i)}$, covariance matrices $\Sigma^{(i)}$

  MoG sample generation $x = \mathcal{N}(\mu^{(i)}, \Sigma^{(i)}), \quad i \sim w_i$

Can we learn the parameters to accuracy ε in poly time using poly samples for every MoG instance ? $Poly(n, k, 1/\omega_o, 1/\epsilon)$

Problem statement

- n-dimensional k-component

  Parameters: weights w_i , means $\mu^{(i)}$, covariance matrices $\Sigma^{(i)}$

  MoG sample generation $x = \mathcal{N}(\mu^{(i)}, \Sigma^{(i)}), \quad i \sim w_i$

Can we learn the parameters with **poly algorithm** for every MoG ?

No!

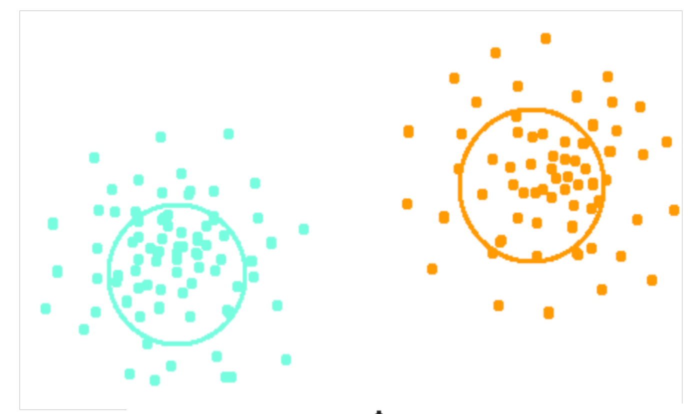
"Exponential dependence in k is unavoidable in general." [Moitra&Valiant]

Prior works

- **General case** $Poly(n, e^{O(k)^k})$

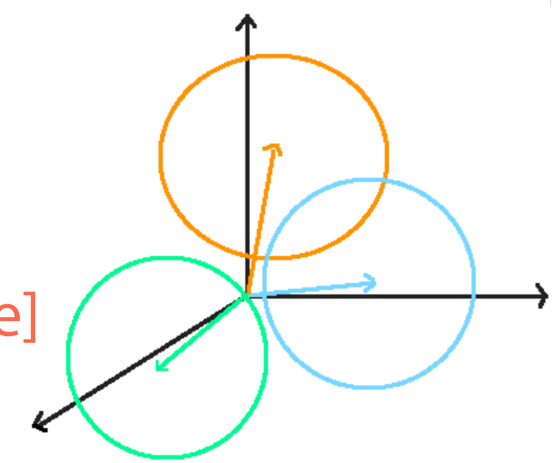
  Moment matching method [Moitra&Valiant] [Belkin&Sinha]

- **Additional assumptions** $Poly(n, k)$

  - ✓ Non-overlapping clusters

    Pair wise clustering [Dasgupta]…[Vempala&Wang]

  - ✓ Spherical, n>k, independent mean vectors

    Lower order moments tensor decomposition [Hsu&Kakade]

- **Density estimation** [Chan et al]

  1-dim $Poly(k)$ Higher dim e^n

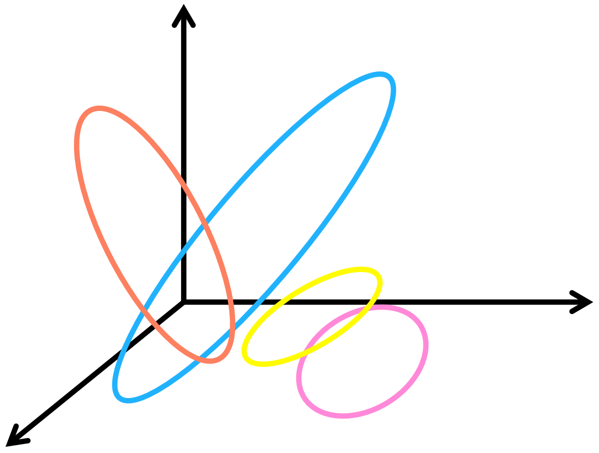
Main result

Can we learn the parameters with **poly algorithm** for **most** MoGs?

- **Yes!**  worst cases are not everywhere

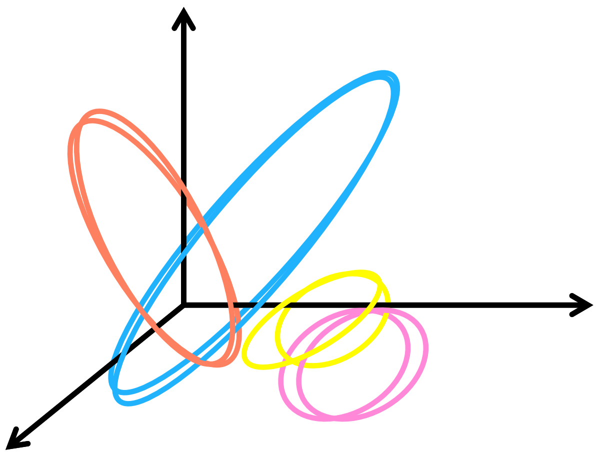
Smoothed analysis Escape from the worst cases

Given an arbitrary instance



Smoothed analysis Escape from the worst cases

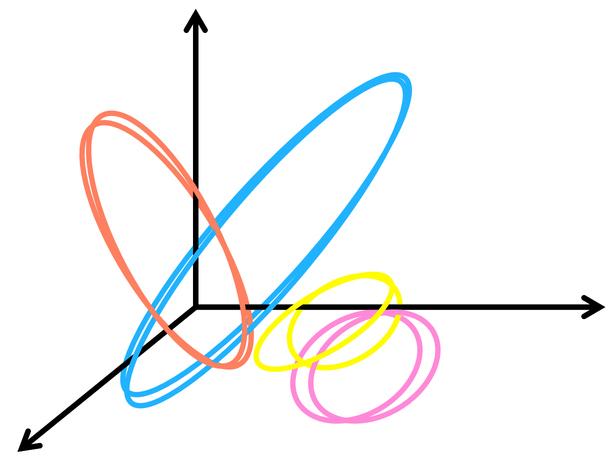
Given an arbitrary instance



Nature perturbs the parameters with a small amount (ρ) of noise

Smoothed analysis Escape from the worst cases

Given an arbitrary instance



Nature perturbs the parameters with a small amount (**ρ**) of noise

Goal: Given samples from smoothed MoG, learn the smoothed parameters with negligible failure probability $O(e^{-n^c})$ over nature's perturbation

[Spielman&Teng]

Hope: With high probability over nature's perturbation, an arbitrary instance

- escapes from the degenerate cases
- becomes a sufficiently well conditioned instance

Main theorem

- Our algorithm learns the MoG parameters up to accuracy ε
  - For high enough dimension $n = \Omega(k^2)$
  - With high probability under smoothed analysis $(1 - O(e^{-n^c}))$
  - Fully polynomial time and sample complexity $Poly(n, k, 1/\epsilon)$

Algorithmic ideas

- Method of moments
  - ✓ Match the first 6-th order moments
  - ✓ Decomposing moments tensor (not low rank, but structured)

$$M_4 = \mathbb{E}[x\otimes^4] \quad M_6 = \mathbb{E}[x\otimes^6]$$

Algorithmic ideas

- Method of moments
  - ✓ Match the first 6-th order moments
  - ✓ Decomposing moments tensor

- Why “high dimension & smooth” help us to learn?
  - ✓ Enough number of moment matching constraints for identifiability

    #parameters $\Omega(kn^2)$ #6-th moments $\Omega(n^6)$

  - ✓ Enough randomness in nature’s perturbation model for well-condition

    Gaussian matrix $X \in \mathbb{R}^{n \times m}$, with prob at least $1 - O(\epsilon^n)$ $\sigma_m(X) \geq \epsilon\sqrt{n}$.

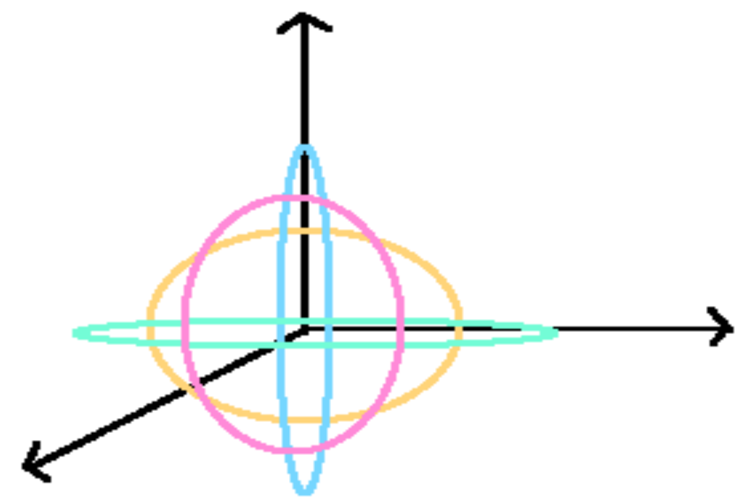
    [Rudelson&Vershynin]

Learn 0-mean MoG

- Why 0-mean?

$x = \mathcal{N}(0, \Sigma^{(i)}), \quad i \sim w_i$

Clean moment structure



- Notation

n-dimensional k-component smoothed MoG

Given empirical moments tensor $M_4 = \mathbb{E}[x^{\otimes 4}]$ $M_6 = \mathbb{E}[x^{\otimes 6}]$

Learn parameters: weights w_i , covariance matrices $\Sigma^{(i)}$

Learn spherical MoG, spectral method review

$$x = \mathcal{N}(\mu^{(i)}, \sigma I_n), \quad i \sim w_i$$

- Construct a low rank tensor from the moments tensor

  [Hsu&Kakade]

$$M_2 = \mathbb{E}[xx^\top] = \sum_{i=1}^k w_i \mu^{(i)} (\mu^{(i)})^\top + \sigma I_n$$

$$M_3 = \mathbb{E}[x\otimes^3] = \sum_{i=1}^k w_i \mu^{(i)} \otimes \mu^{(i)} \otimes \mu^{(i)} + \ \sigma \text{ terms}$$

Learn spherical MoG, spectral method review

$$x = \mathcal{N}(\mu^{(i)}, \sigma I_n), \quad i \sim w_i$$

- Construct a low rank tensor from the moments tensor

Low rank matrix $M_2 = \sum_{i=1}^{k} w_i \mu^{(i)} (\mu^{(i)})^\top + \cancel{\sigma I_n}$

Low rank tensor $M_3 = \sum_{i=1}^{k} w_i \mu^{(i)} \otimes \mu^{(i)} \otimes \mu^{(i)} + \cancel{\sigma \text{ terms}}$

- Low rank tensor decomposition, $\mu^{(i)}$'s are independent

Learn 0-mean general covariance MoG

$$x = \mathcal{N}(0, \Sigma^{(i)}), \quad i \ \sim w_i$$

$$X_4 = \sum_{i=1}^{k} w_i \mathrm{vec}(\Sigma^{(i)}) \otimes \mathrm{vec}(\Sigma^{(i)})$$

$$X_6 = \sum_{i=1}^{k} w_i \mathrm{vec}(\Sigma^{(i)}) \otimes \mathrm{vec}(\Sigma^{(i)}) \otimes \mathrm{vec}(\Sigma^{(i)})$$

Moments structure of 0-mean MoG

- Isserlis' theorem for 4-th moments

Have empirical moments

$$[M_4]_{1,2,3,4} = \mathbb{E}[x_1 x_2 x_3 x_4]$$

$$= \sum_{i=1}^{k} w_i \left(\Sigma^{(i)}_{1,2}\Sigma^{(i)}_{3,4} + \Sigma^{(i)}_{1,3}\Sigma^{(i)}_{2,4} + \Sigma^{(i)}_{1,4}\Sigma^{(i)}_{2,3} \right)$$

Want low rank matrix!

$$[X_4]_{1,2,3,4} = \sum_{i=1}^{k} w_i \Sigma^{(i)}_{1,2}\Sigma^{(i)}_{3,4}$$

$$X_4 = \sum_{i=1}^{k} w_i \mathrm{vec}(\Sigma^{(i)}) \otimes \mathrm{vec}(\Sigma^{(i)})$$

Moments structure of 0-mean MoG

- Isserlis' theorem for 6-th moments

Have empirical moments

$$[M_6]_{1,2,3,4,5,6} = \mathbb{E}[x_1 x_2 x_3 x_4 x_5 x_6]$$

$$= \sum_{i=1}^{k} w_i \Big(\underbrace{\Sigma^{(i)}_{1,2}\Sigma^{(i)}_{3,4}\Sigma^{(i)}_{5,6} + \Sigma^{(i)}_{1,3}\Sigma^{(i)}_{2,4}\Sigma^{(i)}_{5,6} + \cdots}_{\text{15 ways to partition } \{1,2,\ldots 6\} \text{ into 3 pairs}} \Big)$$

Want low rank tensor!

$$[X_6]_{1,2,3,4,5,6} = \sum_{i=1}^{k} w_i \Sigma^{(i)}_{1,2}\Sigma^{(i)}_{3,4}\Sigma^{(i)}_{5,6}$$

$$X_6 = \sum_{i=1}^{k} w_i \mathrm{vec}(\Sigma^{(i)}) \otimes \mathrm{vec}(\Sigma^{(i)}) \otimes \mathrm{vec}(\Sigma^{(i)})$$

Unfold Moments Tensor M_4 M_6

$$M_4 = \mathcal{F}_4(X_4)$$

Estimate from data

Structured linear projection

Desired low rank tensor!

$$M_6 = \mathcal{F}_6(X_6)$$

$$X_4 = \sum_{i=1}^{k} w_i \text{vec}(\Sigma^{(i)}) \otimes \text{vec}(\Sigma^{(i)})$$

$$X_6 = \sum_{i=1}^{k} w_i \text{vec}(\Sigma^{(i)}) \otimes \text{vec}(\Sigma^{(i)}) \otimes \text{vec}(\Sigma^{(i)})$$

- Recover low rank tensors from their linear projections

  Looks like matrix sensing, but standard method does not apply

Exploit low rank property of X_4 X_6 to unfold M_4 M_6

$$M_4 = \mathcal{F}_4\boxed{(X_4)}$$

$\binom{n}{4} \approx \frac{n^4}{24}$ $\approx \frac{n^4}{8}$ Underdetermined linear eqn's

- Given $U \in \mathbb{R}^{\frac{n^2}{2} \times k}$ the k-dim span of $\mathrm{vec}(\Sigma^{(i)})'s$, change variable $\boxed{X_4} = U^\top \boxed{Y_4} U$

$$M_4 = \mathcal{F}_4(U^\top \boxed{Y_4} U)$$

$\binom{n}{4} \approx \frac{n^4}{24}$ $\approx \frac{k^2}{2}$ Unique solution!

Exploit low rank property of X_4 X_6 to unfold M_4 M_6

$$M_4 = \mathcal{F}_4(\boxed{X_4})$$

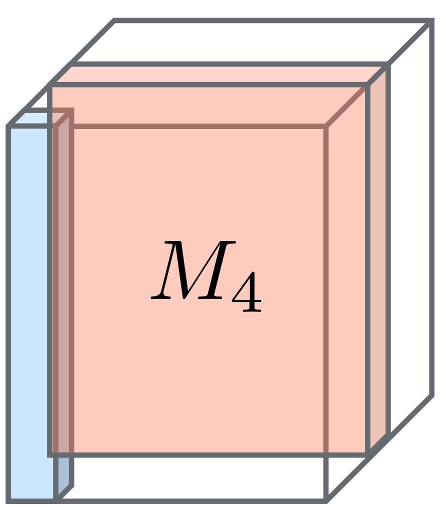
$\binom{n}{4} \approx \frac{n^4}{24}$ $\approx \frac{n^4}{8}$ Underdetermined linear eqn's

- Given $U \in \mathbb{R}^{\frac{n^2}{2} \times k}$ the k-dim span of $\mathrm{vec}(\Sigma^{(i)})'s$, change variable $\boxed{X_4} = U^\top \boxed{Y_4} U$

$$M_4 = \mathcal{F}_4(U^\top \boxed{Y_4} U)$$

$\binom{n}{4} \approx \frac{n^4}{24}$ $\approx \frac{k^2}{2}$ Unique solution!

- Find U by examine the structure of M_4
  - ✓ 1-d columns of M_4 are related to columns of $\Sigma^{(i)}$'s
  - ✓ 2-d slices of M_4 are related to $\Sigma^{(i)}$'s



Algorithm outline. Learn 0-mean MoG

- Step 1. Find the span of $\mathrm{vec}(\Sigma^{(i)})'s$
- Step 2. Use the span to change variable and unfold the M_4 M_6 to get unfolded moments X_4 X_6

$$X_4 = \sum_{i=1}^{k} w_i \mathrm{vec}(\Sigma^{(i)}) \otimes \mathrm{vec}(\Sigma^{(i)})$$

$$X_6 = \sum_{i=1}^{k} w_i \mathrm{vec}(\Sigma^{(i)}) \otimes \mathrm{vec}(\Sigma^{(i)}) \otimes \mathrm{vec}(\Sigma^{(i)})$$

- Step 3. Low rank tensor decomposition to recover $\mathrm{vec}(\Sigma^{(i)})'s$

Each step involves basic matrix operations
(poly time and poly stable !)

Sketch of proofs

Deterministic conditions for correctness and stability of each step

- Step 1. Find the span of $\mathrm{vec}(\Sigma^{(i)})'s$

  Rank factorization of matrices constructed with M_4

  Randomness from ρ-perturbation to guarantee the factors are full rank.

- Step 2. Unfold M_4, M_6 to get X_4, X_6

  Solving over-determined linear system

  Randomness from ρ-perturbation to guarantee the coefficient matrix is full rank.

- Step 3. Tensor decomposition of X_4, X_6

  Randomness from ρ-perturbation to guarantee tensor factors are well-conditioned

Algorithm outline. Learn General MoG

- Step 1. Find the span of $\mu^{(i)}$'s and span of $\Sigma^{(i)}$'s projected to $span\{\mu^{(i)}\}^{\perp}$
- Step 2. In the subspace $span\{\mu^{(i)}\}^{\perp}$ find $\Sigma^{(i)}$'s use our 0-mean algorithm
- Step 3. Find the $\mu^{(i)}$'s using M_3
- Step 4. Find the full covariance matrices $\Sigma^{(i)}$'s

Algorithm outline. Learn General MoG

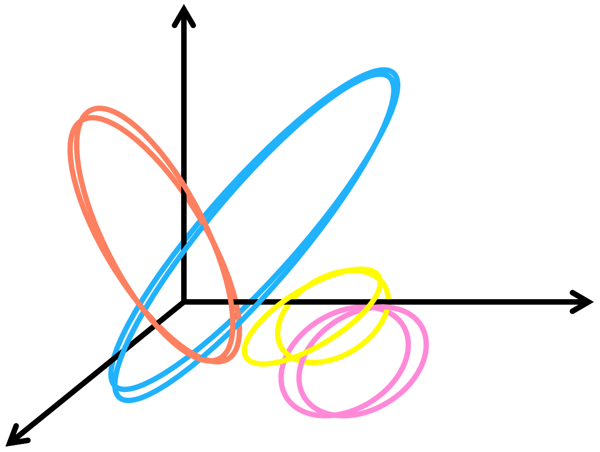
- Step 1. Find the span of $\mu^{(i)}$'s and span of $\Sigma^{(i)}$'s projected to $span\{\mu^{(i)}\}^{\perp}$
- Step 2. In the subspace $span\{\mu^{(i)}\}^{\perp}$ find $\Sigma^{(i)}$'s use our 0-mean algorithm
- Step 3. Find the $\mu^{(i)}$'s using M_3
- Step 4. Find the full covariance matrices $\Sigma^{(i)}$'s

Take away messages

- Provide a fully **poly** algorithm under **smoothed analysis**

  (avoid worst case complexity exponential in k)

- Can potentially relax $n \geq \Omega(k^2)$ by using higher order moments?
- Other “hard problems” in learning?



Thank you! Question?