CS 224 Fall 2023 Scribes: Daniel Sheremeta, Kenny Gu
September 25, 2023

Lecture 6: Learning Mixtures of Gaussians with SoS

Outline

Last time: SoS basics and application to robust regression. Recall the general design setup for our SoS algorithm:

1. Set up a system of polynomial inequalities in variable. For robust regression: $\{(a_i), w\}$.
2. Optimize an objective over pseudo-distributions $\tilde{\mathbb{E}}$ over solutions.
3. Give "simple proof of identifiability" that global optimizer has small clean MSE.
4. Rounding step. For robust regression: output $\tilde{\mathbb{E}}[w]$.

Today, we study an application to learning mixtures of Gaussians. Warning: The system of inequalities and rounding step will be trickier than with robust regression.

1 Mixtures of Gaussians

Definition 1 (Mixtures of Gaussians)**.** *Given centers $\mu_1, \dots, \mu_k \in \mathbb{R}^d$ and mixing weights $\lambda_1, \dots, \lambda_k \in [0, 1]$ such that $\sum_i \lambda_i = 1$, we are given i.i.d. samples from*

$$q = \sum_i \frac{1}{k} \cdot N(\mu_i, \mathrm{Id}_d).$$

For our purposes today, assume $\lambda_i = 1/k$ for all k.

If the μ_i are robustly linearly independent and $k \le d$, we can use Jennrich's algorithm. When $k \gg d$ (i.e. the overcomplete regime), it is not possible to apply Jennrich's algorithm, but (i) if μ_i are random and $k \ll d^{\ell/2}$ then we may apply the tensor power method on degree-ℓ tensors or (ii) if μ_i's are smoothed, then Jennrich's on degree-ℓ moment tensor provably works when $k \ll d^{\lfloor(\ell-1)/2\rfloor}$.

Today, we try to find an approach that doesn't require strong conditions on μ_i. Instead we only require that μ_i's are "well-separated". Define

$$\Delta \stackrel{\text{def}}{=} \max_{i \neq j} |\mu_i - \mu_j|.$$

Intuitively, it is more difficult to distinguish the centers when they are close together. A natural question to ask is the minimum separation Δ such that the centers are efficiently recoverable. One of the first results on this question was:

Theorem 1 ([Das99]). *If $\Delta \geq \Omega(d^{1/2})$, then*

$$\mathbb{P}_{x \sim N(\mu, \mathrm{Id})}[\|x - \mu\| > \sqrt{d} + t] \leq \exp(-\Omega(t^2)),$$

so components are nearly disjoint and a clustering algorithm will work.

In some sense, this theorem states that the natural radius of a Gaussian cluster is $O(d^{1/2})$. In [AK05], the authors exploit a geometric observation to obtain a stronger bound for Δ.

Theorem 2 ([AK05]). *Suppose $\Delta \geq \Omega(d^{1/4})$. Even though components now overlap, it follows from a geometric argument that every pair of vertices from the same component will be closer than every of vertices from different components, so the centers can be recovered efficiently.*

The geometric argument leverages the fact that, in high dimensions, random vectors are approximately orthogonal, so we may apply the Pythagorean theorem.

How far can the bound for Δ be pushed down? In [RV17], the authors proved an impossibility result.

Theorem 3 ([RV17]). *For $\Delta = o(\sqrt{\log k})$, it is information-theoretically impossible to recover the centers μ_i.*

The most natural question to ask if if the gap can be closed. Is the $\Theta(\sqrt{\log k})$ threshold tight? This brings us to the main result of this lecture.

Theorem 4 ([HL18], [KSS18], [DKS18]). *For any $t > 0$, if $\Delta \geq \Omega(k^{1/t})$, there is an algorithm with time and sample complexity $d^{O(t)} \cdot \mathrm{poly}(k)$-time which recovers $\mu_1, \ldots, \mu_k$ to error $t^{t/2}/\mathrm{poly}(k, d)$.*

In other words, if $\Delta \geq k^{0.001}$, time and sample complexity are polynomial. By taking $t = \log k$, we get a quasipolynomial-time algorithm that achieves the threshold $\Delta \asymp \log k$. (The quasipolynomial-time algorithm was improved to a polynomial time algorithm in [LL22].) This algorithm is based on running degree-t SoS algorithm.

1.1 Inefficient algorithm

Idea: brute-force search over subsets of size $N = n/k$ to find a subset that "looks like it came from a single Gaussian". What precisely does it mean to look like a Gaussian? The *moment bounds* for $\mathcal{N}(\mu, \mathrm{Id})$ are a distinctive feature of Gaussian distributions. For any $\mu \in \mathbb{S}^{d-1}$,

$$\mathbb{E}_x[\langle \mu, x - \mu \rangle^s] = (s-1)!! \leq s^{s/2} \quad \text{for all even } s.$$

Hence for sufficiently many samples $q \sim \mathcal{N}(\mu, \mathrm{Id})$,

$$\frac{1}{N} \sum_i \langle u, x_i - u \rangle^s \leq 2s^{s/2} \quad \text{for all even } s. \tag{1}$$

with high probability.

This leads to the main idea behind our "inefficient algorthim" on which we will apply SoS. Search over all subsets of size $N = n/k$ and find a subset $\{x_i\}$ of the samples that satisfies inequality (1) where u is taken as the estimated mean. More explicitly, here is the SoS Program.
Input: $\{(x_i)\}_{i=1}^n$ sampled i.i.d. from $\frac{1}{k}\sum_j \mathcal{N}(\mu_j, \mathrm{Id})$.
Variables:

- μ (vector): our estimates for a center
- $a_1, \ldots, a_n$ (scalar): indicators for points that we think comprise a component.

Constraints:

- $a_i^2 - a_i = 0$ for all $1 \leq i \leq n$ (indicators are in $\{0, 1\}$)
- $\sum_i a_i = n/k$ (components makes up $1/k$ of data)
- $\frac{1}{n/k} \sum_i a_i x_i = \mu$ (μ is the empirical mean of selected points)
- $\frac{1}{n/k} \sum_i a_i \langle x_i - \mu, u \rangle^t \leq 2t^{t/2} \|u\|^t$ (empirical moments approximate Gaussian)

Problem: Unlike in robust regression, there are k "ground truths" instead of just 1. In fact, any distribution over components yields a valid pseudo-distribution, so we will need to pick a special objective function to force the pseudo-distribution to look like a uniform distribution over the components.

In particular, these subtleties imply we that need a fancier "rounding algorithm" than simply outputting $\tilde{\mathbb{E}}[\mu]$.

Another problem: Our last constraint is currently quantified over all $u \in \mathbb{S}^{d-1}$, but we need to write a finite set of constraints for the SoS program.

Idea: We can encode everything in a big tensor

$$T = \frac{1}{n/k}\sum_i a_i(x_i-\mu)^{\otimes t} \implies \frac{1}{n/k}\sum_i a_i\langle x_i-\mu,u\rangle^t = T(u,u,u).$$

Then we want a constraint that $T \approx \mathbb{E}_{x\sim\mathcal{N}(0,\mathrm{Id})}[x^{\otimes t}]$, so we may set

$$\|T - \mathbb{E}_{x\sim N(0,\mathrm{Id})}[x^{\otimes t}]\|_F^2 \le 1$$

as the constraint.

1.2 Proof of identifiability

For notational convenience, let $N = n/k$. Let $S_j = \{i \in [n] \text{ from } \mathcal{N}(\mu_i, \mathrm{Id})\}$. For technical reasons to be seen later, suppose t is a power of 2 and suppose $\Delta \gg \sqrt{t}k^{1/t}$.

Let a_i choose a subset $S \subset [n]$ of size N, and define $c_j = \frac{|S\cap S_j|}{N}$. Thus c_j is a normalized measure of the overlap between S and S_j. However, we can't define c_j in the SoS paradigm, so write

$$c_j = \frac{1}{N}\sum_{i\in S_j} a_i$$

Thus $\sum_{j=1}^k c_j = 1$.

For convenience and pedagogical purposes, we restrict our attention to the case $d = 1$. The higher-dimensional cases are relatively similar. Then the moment bound in (1) is equivalent to

$$\frac{1}{N}\sum_i a_i(x_i-\mu)^t \le 2t^{t/2}$$

Lemma 1. *For all j,*

$$c_j^t(\mu-\mu_j)^t \le O(t)^{t/2}\cdot c_j^{t-1}$$

Interpretation. Taking this out of the SoS paradigm, this is equivalent to $|\mu-\mu_j| \le O(\sqrt{t})\cdot c_j^{-1/t}$. i.e. if the overlap is large, than the outputted mean will be close to ground-truth.

Degree-t SoS proof of Lemma 1. The main tool is "SoS Hölder's." Recall Hölder's inequality, which states that $\langle b,c\rangle \le \|b\|_p \cdot \|c\|_q$ for all p,q satisfying $\frac{1}{p}+\frac{1}{q}$. It is easier to deal with integral powers than fractional powers in the SoS paradigm, so rewrite this inequality by the equivalent

$$\left(\sum_i b_i c_i\right)^t \le \left(\sum_i b_i^{\frac{t}{t-1}}\right)\left(\sum_i c_i^t\right).$$

If $b_i^2 = b_i$, then there is a degree-t SoS proof that the above inequality is true (proof omitted). Applying this to our situation, we obtain that

$$\begin{aligned}\left(\frac{1}{N}\sum_{i\in S_j} a_i\right)^t (\mu-\mu_j)^t &= \left(\frac{1}{N}\sum_{i\in S_j} a_i(\mu-\mu_j)\right)^t \\ &\le \left(\frac{1}{N}\sum_{i\in S_j} a_i\right)^{t-1}\left(\frac{1}{N}\sum_{i\in S_j} a_i(\mu-\mu_j)^t\right) \\ &= c_j^{t-1}\left(\frac{1}{N}\sum_{i\in S_j} a_i[(\mu-x_i)-(\mu_j-x_i)]^t\right)\end{aligned}$$

Recall from last lecture that $(a+b)^t \le 2^t(a^t+b^t)$ for all t (also from Hölder's). Since t is even, it follows that the last expression is bounded by

$$\begin{aligned}\left(\frac{1}{n}\sum_{i\in S_j} a_i\right)^t (\mu-\mu_j)^t &\le c_j^{t-1}2^t\left(\frac{1}{N}\sum_{i\in S_j} a_i(\mu-x_i)^t + \frac{1}{N}\sum_{i\in S_j} a_i(\mu_j-x_i)^t\right) \\ &\le c_j^{t-1}2^t\left(2t^{t/2}+2t^{t/2}\right) \\ &= c_j^{t-1}O(t)^{t/2}\end{aligned}$$

by applying the moment bound to $\mathcal{N}(\mu, \mathrm{Id})$ and the constraint $\frac{1}{N}\sum_{j\in S_j} a_i(\mu_j - x_i)^t \le 2t^{t/2}$. □

Claim 1. *The brute-force algorithm returns μ which is very close to μ_j. Explicitly, for every center $\mu^* = \mu_j$, then the component S_{j^*} with largest overlap with S satisfies $|S \cap S_{j^*}| = (1-\delta)N$ for $\delta < kt^{t/2} - O(1/\delta^t) << 1$.*

Proof of claim. Suppose without loss of generality that $c_1 \ge c_2 \ge \cdots \ge c_k$. Since $\sum_j c_j = 1$, we have $c_1 \ge 1/k$. Let $c_1 = 1-\delta$, so $c_2 \ge \delta/k$. Thus there is non-trivial overlap for at least two components. By Lemma 1,

$$|\mu - \mu_1| \le O(\sqrt{t})\cdot c_1^{-1/t} = O(\sqrt{t})(1-\delta)^{-1/t} \le O(\sqrt{t})k^{1/t} \ll \Delta$$

By the triangle inequality,

$$|\mu-\mu_2| \ge |\mu_1-\mu_2| - |\mu_1-\mu| \ge \Delta/2$$

Then applying Lemma 1 again,

$$\Delta/2 \le |\mu-\mu_2| \le O(\sqrt{t})\cdot c_2^{-1/t} \le O(\sqrt{t})\cdot(\delta/k)^{-1/t}$$

Rearranging, this gives

$$\delta \lesssim \left(\frac{k^{1/t}\sqrt{t}}{\Delta}\right)^t = o(1)$$

□

The upshot of this is that c_1 is very close to 1, so very little is lost by throwing away the vector chosen by the SoS algorithm and we may throw away the points corresponding to this cluster, then repeat on find the centers of the remaining clusters.

However, there is a problem with our proof which captures a big theme in the SoS paradigm. The claim above "breaks symmetry" by ordering the c_i and examining the largest. This sort of proof is hard to implement in the SoS paradigm. Instead we want to prove a version of the claim which doesn't break symmetry.

Claim 2 (Symmetric version of Claim 1)**.**

$$\sum_j c_j^2 \geq 1 - k^2 t^{t/2} O(1/\Delta)^t = 1 - o(1)$$

Interpretation. This is a stronger claim because $\|c\|_\infty \geq \frac{\|c\|_2^2}{\|c\|_1} = \|c\|_2^2$.

Proof of Claim. Rewrite

$$1 = \left(\sum_j c_j\right)^2 = \sum_j c_j^2 + \sum_{i \neq j} c_i c_j$$

For any $i \neq j$, we can bound

$$\begin{aligned} c_i c_j &\leq c_i c_j \left(\frac{|\mu_i - \mu_j|}{\Delta}\right)^t \\ &\leq c_i c_j \left(\frac{|\mu_i - \mu| + |\mu_j - \mu|}{\Delta}\right)^t \\ &\leq \frac{2^t}{\Delta^t} c_i c_j \left((\mu_i - \mu)^t + (\mu_j - \mu)^t\right) \end{aligned}$$

In the last line, we use again the inequality $(a+b)^t \leq 2^t(a^t + b^t)$ Since $|\mu_i - \mu| \leq O(\sqrt{t}) c_i^{-1/t}$ and similarly for j, so

$$c_i c_j \leq O(t)^{t/2}/\Delta^t.$$

Then $\sum_j c_j^2 = 1 - o(1)$, completing the proof. □

1.3 Objective function

If $\tilde{\mathbb{E}}$ was actually a uniform distribution over components (i.e. $a_i = \mathbf{1}[i \in S_j]$), then

$$\tilde{\mathbb{E}}[aa^T] = \mathbb{E}_j[a^{(j)}(a^{(j)})^\top]$$

Thus we want to "maximise entropy" i.e. be agnostic towards which component is picked out by the SoS program. A good objective that does this is

$$\min_{\tilde{\mathbb{E}}} \left\| \tilde{\mathbb{E}}[aa^\top] \right\|_F^2 .$$

We will explain why this maximises entropy and give the full algorithm next lecture.

References

[AK05] Sanjeev Arora and Ravi Kannan. Learning mixtures of separated nonspherical Gaussians. *Ann. Appl. Probab.*, 15(1A):69–92, 2005.

[Das99] Sanjoy Dasgupta. Learning mixtures of Gaussians. In *40th Annual Symposium on Foundations of Computer Science (New York, 1999)*, pages 634–644. IEEE Computer Soc., Los Alamitos, CA, 1999.

[DKS18] Ilias Diakonikolas, Daniel M. Kane, and Alistair Stewart. List-decodable robust mean estimation and learning mixtures of spherical Gaussians. In *STOC'18—Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1047–1060. ACM, New York, 2018.

[HL18] Samuel B. Hopkins and Jerry Li. Mixture models, robustness, and sum of squares proofs. In *STOC'18—Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1021–1034. ACM, New York, 2018.

[KSS18] Pravesh K. Kothari, Jacob Steinhardt, and David Steurer. Robust moment estimation and improved clustering via sum of squares. In *STOC'18—Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1035–1046. ACM, New York, 2018.

[LL22] Allen Liu and Jerry Li. Clustering mixtures with almost optimal separation in polynomial time. In *STOC '22—Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1248–1261. ACM, New York, [2022] ©2022.

[RV17] Oded Regev and Aravindan Vijayaraghavan. On learning mixtures of well-separated Gaussians. In *58th Annual IEEE Symposium on Foundations of Computer Science—FOCS 2017*, pages 85–96. IEEE Computer Soc., Los Alamitos, CA, 2017.