Lec 9: Avg-Case Algo Design: Clustering Mixtures

<u>Today</u>: clustering spherical mixtures

<u>Next time</u>: robust moments estimation
dictionary learning $\approx$ planted sparse vectors
tensor decomposition $\approx$ maximizing random polynomials
list-decodable learning
matrix-completion | tensor completion
Comm. detection.

Clustering Spherical Mixtures

<u>Input</u>: $x_1, \ldots, x_n \in \mathbb{R}^d$

$\boxed{\mu_1, \ldots, \mu_k \in \mathbb{R}^d \text{ unknown means}}$

generated as follows:

① For each $1 \le i \le k$:
choose $\frac{n}{k}$ i.i.d. samples from $N(\mu_i, \mathbb{I})$ $\in \mathbb{R}^d$
$\hookrightarrow C_i$: "true clusters"

② Shuffle & return $\bigcup_{i}^{k} C_i$.

② Shuffle & return $\bigcup_{i=1}$

$\boxed{C_1 \ldots C_k}$

not known.

"original cluster"

Output: $\hat{C}_1, \ldots, \hat{C}_k$

1) $|\hat{C}_i| = \frac{n}{k} \quad \forall\ 1 \le i \le k$

& $\exists\ \pi : [k] \to [k]$

$$|\hat{C}_i \cap C_{\pi(i)}| \geq (1-\delta) \cdot \frac{n}{k}.$$

Remark: related goal. recover good estimates of μ_is. This can be done using $\hat{C}_i$s. Details in Nov'10 lecture.

Issue: problem is not soluble as stated.

Def (Separation):

$$\Delta = \min_{i \neq j} \|\mu_i - \mu_j\|_2$$

There must be some l.b. on Δ to make problem soluble

Heuristic reasoning for required l.b. on Δ.

$\boxed{d=1}$

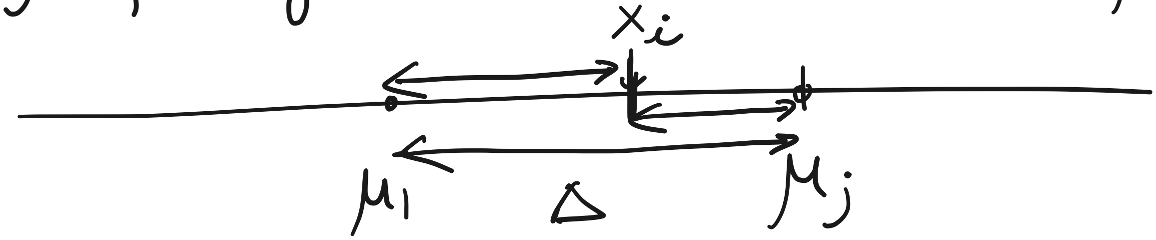
Let's say you know $\mu_1, \ldots, \mu_K$.

<u>Clustering heuristic:</u> associate each x_i to the μ_j nearest to it.

<u>x_i</u> : <u>$N(\mu_1, 1)$</u> , generated "from" 1st Gaussian.

error if x_i happens to be nearer to μ_j for $j > 1$

$|\mu_1 - \mu_j| \geq \Delta$



$$|x_i - \mu_1| > \frac{\Delta}{2}$$

$$\propto \frac{1}{\sqrt{2\pi}} \cdot e^{-\frac{1}{2} \cdot \frac{\Delta^2}{4}}$$

total # of points in cluster 1 that are wrongly assigned

$$\sim n \cdot e^{-\Delta^2/8} \cdot \frac{1}{\sqrt{2\pi}}$$

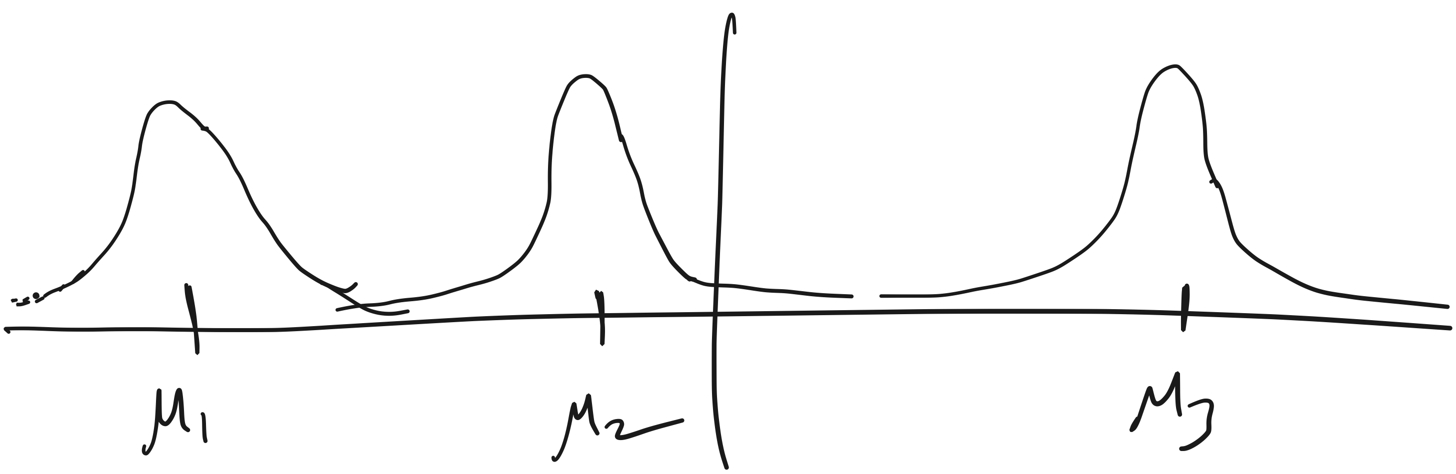
$$(k-1)\frac{1}{\sqrt{2\pi}}\cdot e^{-\frac{1}{2}\cdot\frac{\Delta^2}{4}} \ll \delta \qquad = \frac{k\cdot e^{-\Delta^2/8}}{\sqrt{2\pi}} \le \delta$$

fraction

$$\boxed{\Delta \ge c\cdot\sqrt{\log k}}$$

This heuristic calculation happens to be roughly correct even in high d.

Let's see what happens $\Delta \gg c\cdot\sqrt{\log k}$.



a-sample algorithm:

① Choose a random sample x_i

② take $\left(\frac{n}{k}-1\right)$ points nearest to it.

call it $\hat{C}_1$.

(3) repeat on remaining points.

<u>Thm:</u>

<u>Regev-Vijayraghavan' 2017</u>

If $\Delta \leq c \cdot \sqrt{\log k}$ for some small enough c, then $\exists$ a mixture of k-gaussians that needs $\geq k^{c^{1/24}}$ <u>samples</u>.

<u>While</u> $\Delta > C \cdot \sqrt{\log k}$ for some large C, then <u>there's</u> a "brute-force" $(d^{\Omega(k)})$ time algo that solves the problem $\rightarrow$ with poly(d,k) samples.

(<u>Vempala-Wang '02</u>:) If $\Delta \gg k^{1/4} \cdot \log(d)$ then there's a poly time algo for clustering

Today:

Thm: [2018] Fix $t > 0$. $n \geq n_0 \sim d^{t}$

There's a SoS-based algo that runs in $n^{O(t)}$,

for the clustering problem s.t. w.p. $1 - 1/d$

outputs $\hat{C}_1 \ldots \hat{C}_k$ that are δ-approx

clustering for $\delta \leq \Delta^{-2t} \cdot (512t)^{t} \cdot k^3$.

Ex: **Let's** set $\delta = 0.01$

$$\left(\frac{512t}{\Delta^2}\right)^{t} \cdot k^3 < 0.01$$

$$\underline{\Delta \sim 10^4 \cdot \sqrt{\log k}}, \quad t \sim \log k.$$

$$\left(\frac{512}{10^8}\right)^{\log k} \cdot k^3$$

$$\downarrow$$

$$\underline{10^{-7 \cdot \log k} \cdot k^3 \ll 0.01}$$

When $\Delta \sim \Theta(\sqrt{\log k})$.

we get an algo with time: $d^{O(\log^2 k)}$

$\downarrow$

$d^{\log k}$

$\downarrow$

quasi-poly algo in k.

<u>Ex</u>: $\Delta > 100 \cdot k^{\varepsilon} / \sqrt{\varepsilon}$. ε - small constant $< 1/4$

$t \sim 3/\varepsilon$ is enough to make $\delta < 0.01$

we get a $d^{O(1/\varepsilon^2)}$ time algo in this setting.

<u>References (2018)</u>

3 independent works in 2018.

Diakonikolas-Kane-Stewart $\rightarrow$ solves the problem for gaussian clusters

[illegible] : K-Steinhardt

Poincaré inequality

: Hopkins-Li

today: simplified proof that appears in Fleming-K-Pitassi (Semi-algebraic proofs & efficient algorithm design).

Focus on certificates

In addition to the input sample, we're given a "purported" cluster: $\hat{C}$.

make some sanity checks on $\hat{C}$.

Goal: if $\hat{C}$ passes these checks then it's close to a good cluster, $|\hat{C}| = \frac{n}{k}$.

"test for solutions"

sample $x_1, \dots, x_n$.

Can use sample ...

A good test should have 2 properties.

① All true clusters $C_1 \ldots C_k$ should pass the test "Completeness"

② If $\hat{C}$ passes the test then

$\exists i: |\hat{C} \cap C_i| \geq (1-\delta) \cdot \frac{n}{k}$. "Soundness"

If such a test exists, then there's a finite time algorithm for our problem.

* If the test satisfies some "extra conditions" then, we'll always be able to obtain a poly time algo to simulate the oracle

---

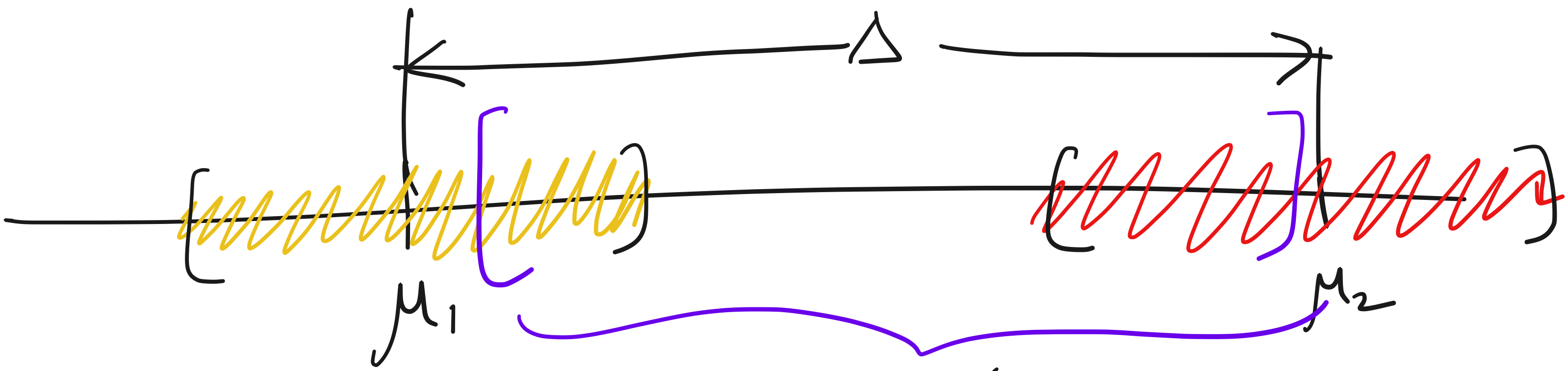
How can we check that $\hat{C}$ is a good

approx:

a approx cluster?

1) Check that $|\hat{C}| = \frac{n}{k}$.

<u>Focus on $d=1$</u>.



<u>Tai's idea:</u> test that $\hat{C}$ has appropriately bounded variance

$$\hat{\mu} = \mu(\hat{C}) = \frac{k}{n} \cdot \sum_{i \in \hat{C}} x_i$$

$$\frac{k}{n} \sum_{i \in \hat{C}} (x_i - \hat{\mu})^2 \overset{?}{\leq} 1.1$$

<u>More generally</u>, $t \in \mathbb{N}$, check

$$\frac{k}{n} \cdot \sum_{i \in \hat{C}} (x_i - \hat{\mu})^{2t} \overset{?}{\leq} 1.1 \cdot t^t.$$

$\ldots^{2t} \ldots (\ldots)!! = \prod i \sim t^t$

$\mathbb{E}(x-\mu)^i = (2t-1)!!$ $\quad i \le 2t$, i: odd

$\mathcal{N}(\mu, I)$

<u>Test</u>: Accept $\hat{C}$ iff

① $|\hat{C}| = \frac{n}{k}$

② $\frac{k}{n} \cdot \sum_{i \in \hat{C}} (x_i - \hat{\mu})^{2t} \le 1.1\, t^t$

where $\hat{\mu} = \mu(\hat{C}) = \frac{k}{n} \cdot \sum_{i \in \hat{C}} x_i$

<u>Def ((t, Δ)-good)</u> $X = C_1 \cup \dots \cup C_k$

An n-sample is (t, Δ)-good if

$\hat{\mu}_i = \mu(C_i)$ for every i

① $\|\hat{\mu}_r - \hat{\mu}_{r'}\|_2 \ge \Delta \quad \forall\, r \ne r'$.

& ② $\frac{k}{n} \sum_{j \in C_r} (x_j - \hat{\mu}_r)^{2t} \le (16t)^t$

$\forall\, 1 \le r \le k$.

Prop: If $n \ge \frac{k}{\varepsilon^2} \cdot \log k/n$

& $\mu_1 - \mu_k$ are $(\Delta+\delta)$ separated

then w.p. $\geq 1-\eta$, n-sample is (t,Δ)-good.

$$X = X_1 \text{ — } X_n$$

<u>Completeness:</u> If X_1 — X_n is (t,Δ) good then every true cluster C_i passes the checks.

$\boxed{\text{Soundness (Usually harder)}}$

Let X be a (t,Δ)-good sample.

Let $C_1 - - C_K$ be true clusters

Let $\hat{C}$ be any collection of points so that ① $|\hat{C}| = \frac{n}{K}$.

② $\frac{K}{n} \cdot \sum_{i \in \hat{C}} (x_i - \mu(\hat{C}))^{2t} \leq (16t)^t$

Then, $\max\limits_{r\le k} |\hat{C}\cap C_r| \geq (1-\delta)\left(\frac{n}{k}\right)$.

for $\delta \le \Delta^{-2t}\cdot(512t)^{t}\cdot k^{3}$.

Observation:

① Solutions can be written as vectors in $\mathbb{R}^n$.

$\hat{C} \longrightarrow w_1 \dots\dots w_n$

$w_i = 1$ iff ith sample is in $\hat{C}$.

② the checks in my test are poly inequalities in w_is.

dw

① $\sum_i w_i = \frac{n}{k}$

② $\frac{k}{n}\sum_i w_i x_i = \mu(\hat{C}) = \hat{\mu}$.

$$\frac{k}{n}\cdot\sum_{i=1}^{n} w_i \,(x_i-\hat{\mu})^{2t} \le (16t)^{t}$$

deg $2t+1$ polynomial

inequality

③ $\forall i \quad w_i^2 = w_i$

③ Conclusion of Soundness of the test can be written as a poly inequality.

$$\max_{r \leq k} |\hat{C} \cap C_r| \geq (1-\delta) \cdot \frac{n}{k}.$$

$$\boxed{\frac{k}{n} \cdot \sum_{i \in C_r} w_i \geq (1-\delta)}$$

Def: $W(C_r) = \frac{k}{n} \sum_{i \in C_r} w_i$

$$\sum_{r=1}^{k} W(C_r) = 1$$

$$\boxed{\sum_{r=1}^{k} W(C_r)^2 \geq 1-\delta}$$

$$\Downarrow$$

$$\max_{r \leq k} W(C_r) \geq 1-\delta$$

$$\sum^{k} W(C_r)^2 = \sum_{r=1}^{k} W(C_r) \cdot W(C_r)$$

$$\leq \sum_{r=1}^{k} w(C_r) \cdot \max_r w(C_r)$$

$$= \left(\sum_{r=1}^{k} w(C_r)\right) \max_r w(C_r)$$

$$= \max_r w(C_r)$$

"SoS Approach to Avg Case Algo design"

Informal statement: If test satisfies above and $\underbrace{\mathcal{A}_w}$ $\vdash^{2t} \left\{ \sum_{i=1}^{k} w(C_i)^2 \geq 1-\delta \right\}$

$\Downarrow$

efficient algo for finding w that satisfy $\mathcal{A}_w$.

Low deg proof of soundness $\Rightarrow$ eff. algorithms

Soundness

Let X be a (t,Δ)-good sample.

Let w satisfy $\mathcal{A}_w$.

Then $\sum_{i=1}^{k} w(C_{\cancel{x}i})^2 \geq 1-\delta$ for

$$\delta \leq \Delta^{-2t} \cdot (512t)^t \cdot k^3.$$

<u>Pf</u>: $1 = \left(\sum_{i=1}^{k} w(C_i)\right)^2 = \sum_{i=1}^{k} w(C_i)^2 + \sum_{r \neq r'} w(C_r)\cdot w(C_{r'}).$

Goal: $\sum_{r\neq r'} w(C_r)\cdot w(C_{r'}).$

$$\leq \frac{1}{\Delta^{2t}} \cdot \sum_{r\neq r'} w(C_r)\cdot w(C_{r'})\cdot(\mu_r - \mu_{r'})^{2t} \quad — *$$

<u>Almost triangle inequality</u>

$$(a+b)^{2t} \leq 2^{2t-1}(a^{2t}+b^{2t}).$$

$$(a+b)^2 \leq 2\cdot(a^2+b^2).$$

$$2(a^2+b^2) = (a+b)^2 + (a-b)^2$$

$\rightarrow$ SoS proof of 2 .ini a,b.

AM-GM inequality

Write $\mu_r - \mu_{r'} = (\mu_r - \mu(\omega)) + (\mu(\omega) - \mu_{r'}).$

$$\mu(\omega) = \frac{k}{n} \cdot \sum_{i=1} w_i \, x_i$$

$\Downarrow$

<u>ATE</u>: $$(\mu_r - \mu_{r'})^{2t} \le 2^{2t-1} \cdot \left[(\mu_r - \mu(\omega))^{2t} + (\mu_{r'} - \mu(\omega))^{2t} \right]$$

$$\le \frac{2^{2t-1}}{\Delta^{2t}} \cdot \sum_{r \ne r'} w(C_r) \cdot w(C_{r'}) \cdot \left[(\mu_r - \mu(\omega))^{2t} + (\mu_{r'} - \mu(\omega))^{2t} \right]$$

$$\le \frac{2^{2t}}{\Delta^{2t}} \cdot \sum_{r=1}^{k} \underline{w(C_r)} \cdot (\mu_r - \mu(\omega))^{2t}$$

$$= \frac{2^{2t}}{\Delta^{2t}} \frac{k}{n} \cdot \sum_{r=1}^{k} \sum_{i \in C_r} \underline{w_i} \cdot (\mu_r - \mu(\omega))^{2t}$$

$$= \frac{2^{2t}}{\Delta^{2t}} \cdot \frac{k}{n} \cdot \sum_{i=1}^{n} w_i (\mu_{c(i)} - \mu(\omega))^{2t}$$

<u>Notation:</u> $i \; : \; c(i) = r \text{ if } x_i \in C_r$.

$$\le \frac{2^{4t}}{\Delta^{2t}} \cdot \frac{k}{n} \cdot \sum_{i=1}^{n} w_i \cdot \left[(x_i - \mu_{c(i)})^{2t} + (x_i - \mu(\omega))^{2t} \right]$$

$$\leq \frac{2^{4t}}{\Delta^{2t}} \cdot k \cdot \left[\frac{1}{n} \sum_{i=1}^{n} \cdot (x_i - \mu_{C(i)})^{2t} + \frac{1}{n} \sum_{i=1}^{n} w_i \cdot (x_i - \mu(\omega))^{2t} \right]$$

$\leq (16t)^t$ (t, Δ)-goodness of X

$\mathcal{A}_\omega$ $(16t)^t$

$$= k \cdot \Delta^{-2t} \cdot (256t)^t$$

Corollary:

$$\mathcal{A}_\omega \; \frac{2t}{w} \left\{ \sum_{i=1}^{k} w(C_i)^2 \geq 1 - \delta \right\}$$

Proof: "Calculus of SoS proofs"

$$\left\{ \begin{array}{l} \sum_{i=1}^{k} w(C_i)^2 - (1-\delta) = \dfrac{SoS(\omega)}{+ f} \\[2ex] f \in \left\{ \sum_{S}^{I} p_S \cdot \prod_{i \in S} f_i \; , \; p_S = SoS \; , \; \deg(p_S \cdot \prod_{i \in S} f_i) \leq 2t \right\} \\[2ex] \mathcal{A}_\omega = \{ f_i(\omega) = 0 \; , \; 1 \leq i \leq - \} \end{array} \right.$$

Algorithm

So far we know that if ω satisfies $\mathcal{A}_\omega$ then ω is close to a true cluster

Natural algo: Search for ω satisfy $\mathcal{A}_\omega$.

algo: finding a pseudo-dist^n D over ω that satisfy $\mathcal{A}_\omega$. $\boxed{\deg 2t}$

First imagine that D happens to be an actual dist^n satisfy $\mathcal{A}_\omega$.

$\Rightarrow$ Support of D are approx. clusters

$\rightarrow$ Could sample. but only have $\leq 2t$ deg moments of D.

Even if D is an actual dist^n $\rightarrow$ unclear how to extract clusters.

Observation: Let D be a pseudodist^n

Satisfy $\mathcal{A}_w$ of deg $\geq 2t$. Then,

$$\tilde{\mathbb{E}}_{w \sim D} \sum_{i=1}^{k} w(C_i)^2 \geq (1-\delta)$$

Pf: $\tilde{\mathbb{E}}\left[\sum_{i=1}^{k} w(C_i)^2 - (1-\delta)\right]$

$$= \tilde{\mathbb{E}}_D\big[\underbrace{SoS(w)}_{\leq 2t \text{ deg}}\big] + \tilde{\mathbb{E}} f \qquad f \in SoS(\mathcal{A}_w), \ \leq 2t \text{ deg}$$

$$\geq 0 \qquad\qquad \geq 0$$

Algorithm:

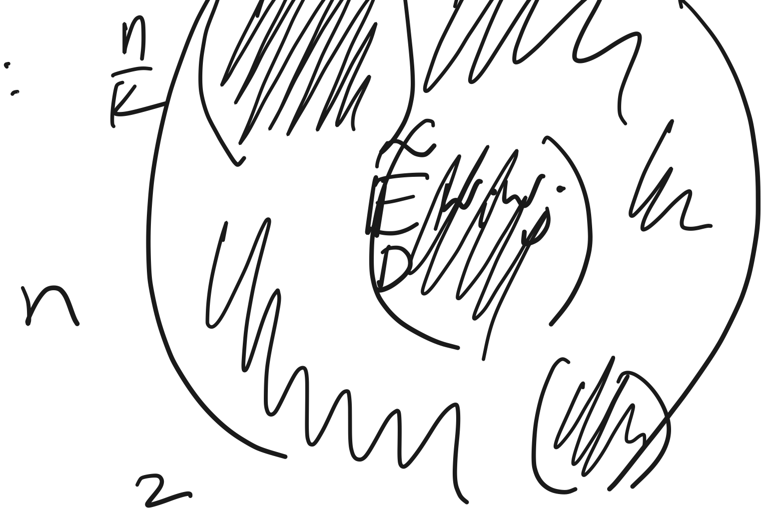
$$\tilde{\mathbb{E}} \sum_{i=1}^{k} w(C_i)^2 \geq 1-\delta$$

$$\|$$

$$\tilde{\mathbb{E}} \sum_{r \neq r'} \overbrace{w(C_r) \cdot w(C_{r'})} \leq \delta \leftarrow$$

$n \qquad n$

$M = \tilde{\mathbb{E}}\, w w^T$:



Sum-of off diagonal blocks $\le \delta \cdot \frac{n^2}{K^2}$

$\tilde{\mathbb{E}} \sum w_i = \frac{n}{K}$

Observe that M is non-neg

$w_i^2 = w_i$: $\tilde{\mathbb{E}}\, w_i^2 = \tilde{\mathbb{E}}\, w_i$

$$\tilde{\mathbb{E}}_D\, w_i w_j = \overset{\geq 0}{\tilde{\mathbb{E}}_D\, w_i^2 \cdot w_j^2} \geq 0.$$

Algorithm:

1. Find a p.d. D satisfying $\mathcal{A}_w$ of deg $\geq 2t$, $\tilde{\mathbb{E}}\, w_i = 1/K \ \forall i$.
2. Choose a random row of M
3. take $\frac{n}{K}$ largest entries in that row.

④ repeat.

Issue with the plan:

We need to force D to have info about all clusters.

Idea: $\tilde{\mathbb{E}}_D \, w_i = \frac{1}{k}$ for $1 \le i \le n$.

This completes the 1-D algorithm:

Test:

① $|\hat{C}| = \frac{n}{k}$

② $\frac{k}{n} \sum_{i \in \hat{C}} (x_i - \hat{\mu})^{2t} \le (16t)^t$

vectors??

higher D

$$② ' : \frac{k}{n} \sum_{i \in \hat{C}} \underbrace{\|x_i - \hat{\mu}\|_2^{2t}}_{} \leq$$

doesn't work $\underline{d^{1/4}}$

$$② '' : \frac{k}{n} \sum_{i \in \hat{C}} \langle x_i - \hat{\mu}, \underset{\downarrow}{v} \rangle^{2t} \leq (16t)^t \, \|v\|_2^4$$

$\mathbb{R}^d$

for every $v \in \mathbb{R}^d$

Succinct reps of poly ineqn among SoS proofs $\rightarrow$ NEXT TIME.

---

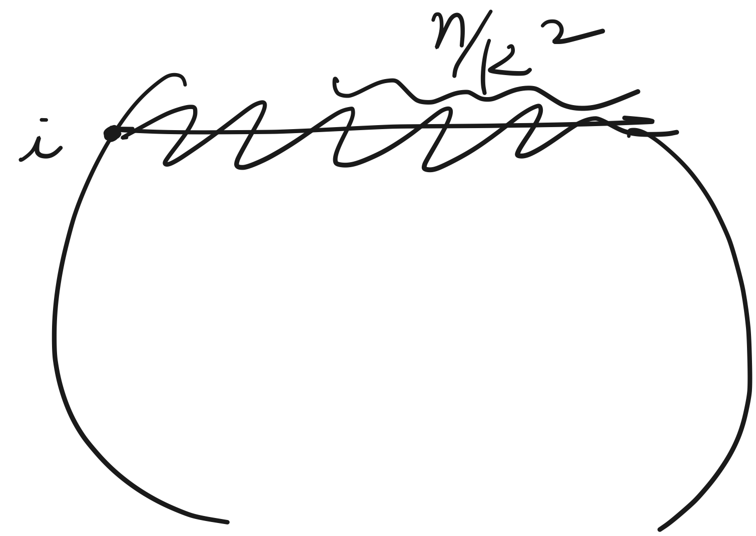
analysis of rounding

$$M = \tilde{\mathbb{E}}\, w w^T$$

$$M(i,j) = \tilde{\mathbb{E}}\, w_i w_j$$

$$\sum_j M(i,j) = \sum_k \tilde{\mathbb{E}}\, w_i w_j = \frac{n}{k} \cdot \tilde{\mathbb{E}}\, w_i$$

$\le \frac{n}{k^2}$



$$\tilde{\mathbb{E}} \sum_{r \neq r'} w(C_r) \cdot w(C_{r'}) \le \delta$$

$$\Updownarrow$$

$$\tilde{\mathbb{E}} \frac{k^2}{n^2} \sum_r \left(\sum_{i \in C_r} w_i \right) \cdot \left(\sum_{j \notin C_r} w_j \right) \le \delta$$

typical i

$$\tilde{\mathbb{E}}\, w_i \cdot \sum_{j \notin C_r} w_j \le 100\,\delta \left(\frac{n}{k} \right).$$

Choose $\delta \ll \frac{1}{k}$.