Recent progress in sum-of-norms clustering

Stephen Vavasis

University of Waterloo
Combinatorics & Optimization

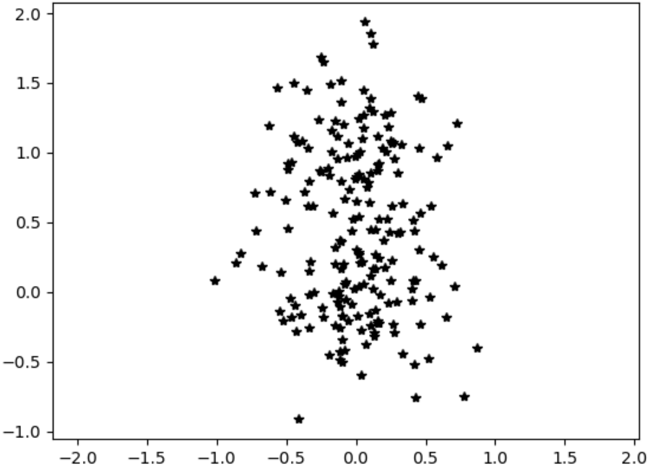
Joint work with Tao Jiang (Cornell), Samuel Tan (Cornell), and Sabrina Zhai (MIT)

Contents

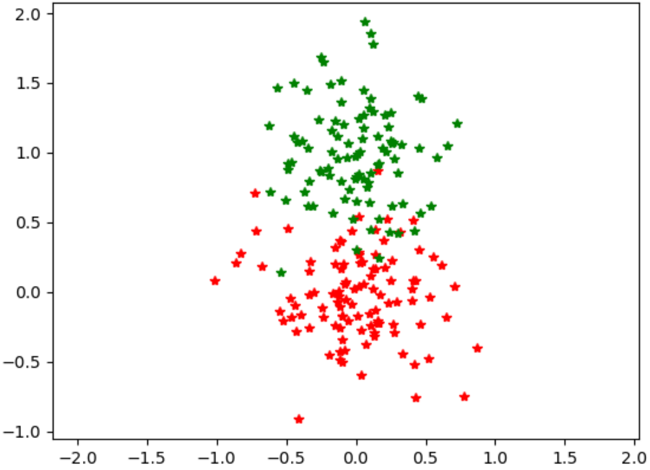
Clustering

- Informally: Given n points $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n \in \mathbb{R}^d$, partition $\{1, \ldots, n\}$ into k subsets $C_1, \ldots, C_k$ such that for $i \in C_m, i' \in C_{m'}$, $\operatorname{dist}(\boldsymbol{a}_i, \boldsymbol{a}_{i'})$ is small iff $m = m'$.
- Clustering is *the* classical example of unsupervised machine learning. *Unsupervised* means: given a single set of unlabeled data, find hidden structure (as opposed to training/test data).
- Some data points may be *noisy* meaning that they should not be assigned to any cluster

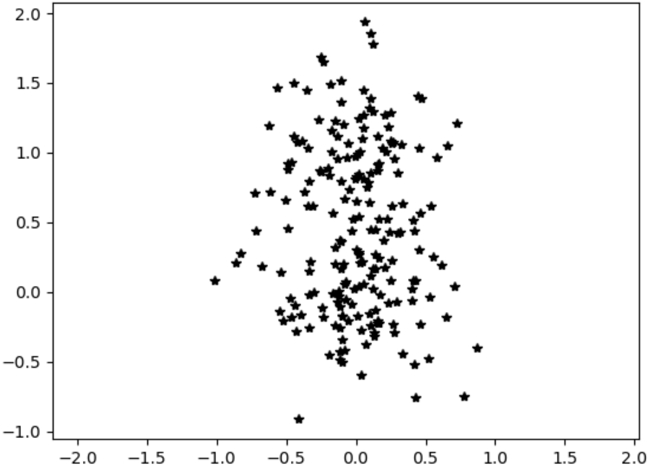
Example of data ($d = 2$)



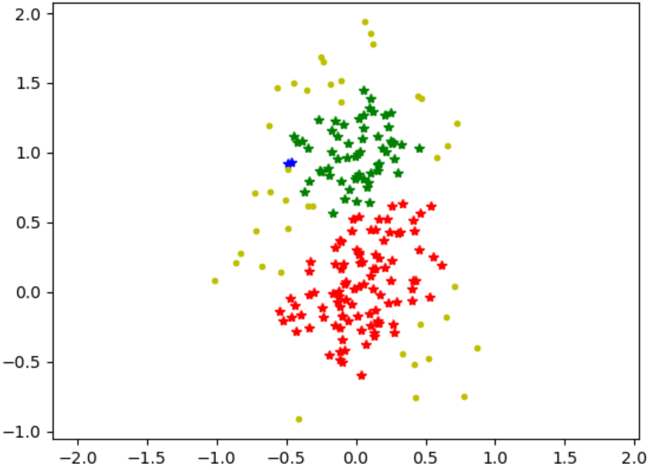
Mixture of Gaussians



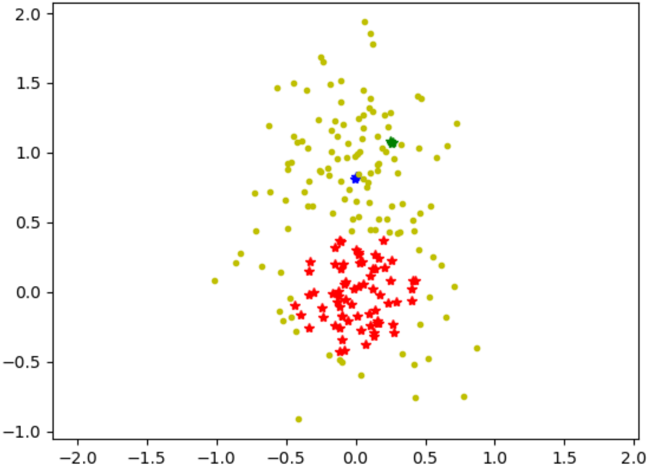
Input to clustering algorithm is unlabeled



Successful clustering of this data



Unsuccessful clustering of this data



Lloyd's algorithm

Best known method for clustering is Lloyd's ("k-means"). Assume k is part of the input.

- Initially partition $\{1, \ldots, n\}$ into k random subsets $C_1, \ldots, C_k$.
- Alternate the following two operations:
- For $m = 1, \ldots, k$, compute $\boldsymbol{\mu}_m := \frac{1}{|C_m|} \sum_{i \in C_m} \boldsymbol{a}_i$.
- For $m = 1, \ldots, k$, define $C_m^{\mathrm{NEW}} := \{i : \|\boldsymbol{a}_i - \boldsymbol{\mu}_m\| = \min_{m'} \|\boldsymbol{a}_i - \boldsymbol{\mu}_{m'}\|\}$.

Issues with Lloyd's algorithm

- Corresponds to nonconvex optimization, so many local minimizers.
- $\implies$ sensitive to initialization
- $\implies$ Hard to prove properties of clustering output.
- Requires preprocessing or other modification to cope with noisy points

Sum-of-norms clustering

- Solve the convex optimization problem:

$$\min_{\mathbf{x}_1,\ldots,\mathbf{x}_n} \frac{1}{2}\sum_{i=1}^{n}\|\mathbf{x}_i - \mathbf{a}_i\|^2 + \lambda \sum_{1\le i<j\le n} \|\mathbf{x}_i - \mathbf{x}_j\|$$

- Intuition: first term favors $\mathbf{x}_i^*$ close to $\mathbf{a}_i$ while second term tends to make $\mathbf{x}_i^*$ for many i's equal to each other.
- Recover clusters according to: i, j clustered together iff $\mathbf{x}_i^* = \mathbf{x}_j^*$.
- Discovered independently by Pelckmans et al. (2005), Lindsten et al. (2011), Hocking et al. (2011).

Strong convexity

$$\min_{x_1,\ldots,x_n} \frac{1}{2}\sum_{i=1}^{n} \|\mathbf{x}_i - \mathbf{a}_i\|^2 + \lambda \sum_{1\le i<j\le n} \|\mathbf{x}_i - \mathbf{x}_j\|$$

- The objective function is strongly convex due to the first summation.
- This means that the optimizer exists and is unique (no dependence on starting point).
- The sum-of-norms formulation is second-order cone programming (SOCP), a special case of semidefinite programming (SDP).
- Convex programming duality can be used to prove strong results about output (more later).

Squared versus unsquared norm

$$\min_{x_1,\ldots,x_n} \frac{1}{2}\sum_{i=1}^{n} \|x_i - a_i\|^2 + \lambda \sum_{1 \le i < j \le n} \|x_i - x_j\|$$

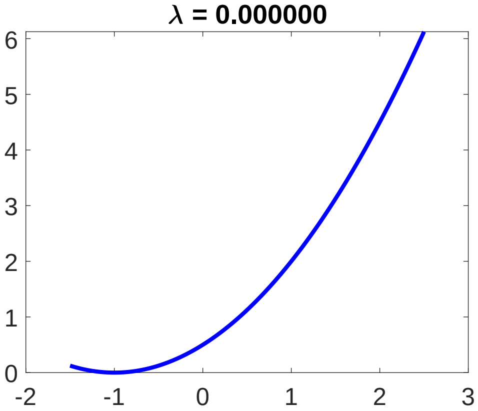
- Note that the first summation has squared norms, but the second summation does not.
- This distinction is crucial: if the norms in the second term were also squared, then it would almost never happen that $x_i^* = x_j^*$ when $i \neq j$.

Squared versus unsquared norm: simple example

$$\min_x (x+1)^2/2 + \lambda|x-2|$$

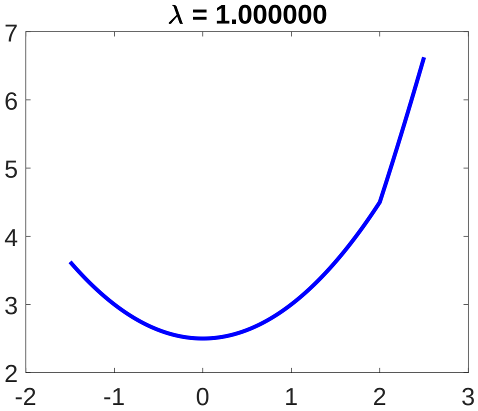
Squared versus unsquared norm: simple example

$$\min_x (x+1)^2/2 + \lambda|x-2|$$



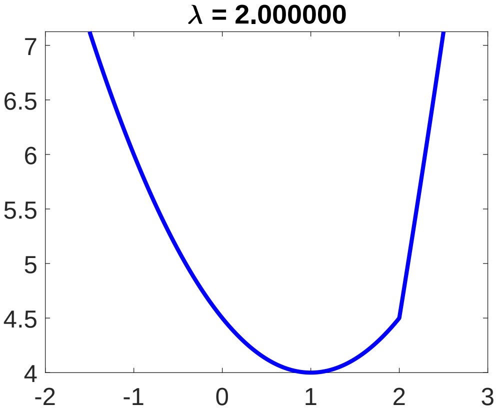
Squared versus unsquared norm: simple example

$$\min_x (x+1)^2/2 + \lambda|x-2|$$



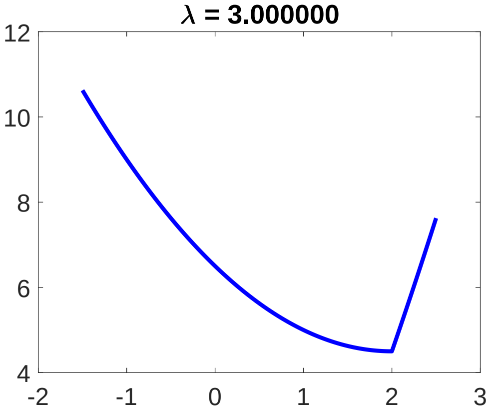
Squared versus unsquared norm: simple example

$$\min_x (x+1)^2/2 + \lambda|x-2|$$



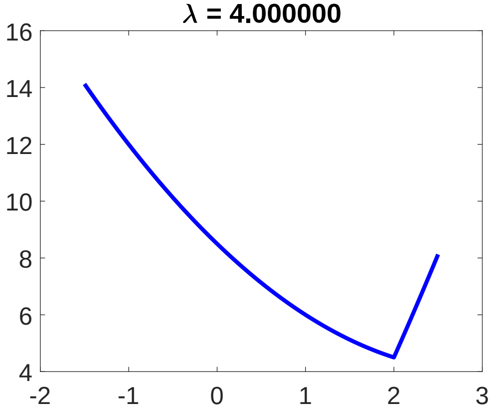
Squared versus unsquared norm: simple example

$$\min_x (x+1)^2/2 + \lambda|x-2|$$



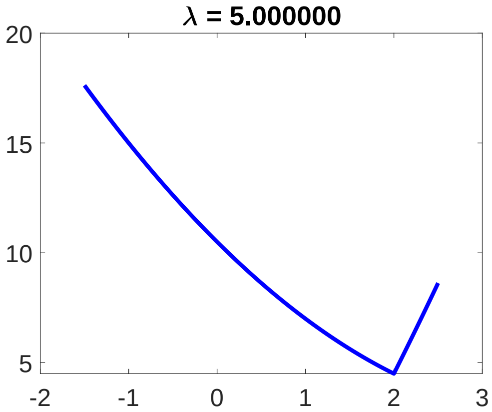
Squared versus unsquared norm: simple example

$$\min_x (x+1)^2/2 + \lambda|x-2|$$



Squared versus unsquared norm: simple example

$$\min_x (x+1)^2/2 + \lambda|x-2|$$



Role of λ

$$\min_{x_1,\dots,x_n} \frac{1}{2}\sum_{i=1}^{n} \|\boldsymbol{x}_i - \boldsymbol{a}_i\|^2 + \lambda \sum_{1\le i<j\le n} \|\boldsymbol{x}_i - \boldsymbol{x}_j\|$$

- Previous example suggests that as λ increases, number of clusters goes down.
- When $\lambda = 0$, all noncoincident $\boldsymbol{a}_i$'s are in singleton clusters.
- There exists $\bar{\lambda}$ (depending on data) such that for all $\lambda \geq \bar{\lambda}$, all $\boldsymbol{a}_i$'s are in one large cluster.
- Thus, λ controls the number of clusters indirectly.

Agglomeration theorem

- Hocking et al. conjectured that sum-of-norms clustering is agglomerative in the sense that as λ increases, clusters may fuse but never break apart.
- This was proved by Chiquet, Gutierrez and Rigaill (CGR) (2017).
- It implies that SON clustering induces a tree of clusters (hierarchical clustering)

Contents

Recovery theorems in machine learning

- Theorem hypothesis: The input data has hidden structure obscured by random noise. The noise has some *a priori* upper bound.
- Theorem conclusion: A particular algorithm can uncover the hidden structure.

Mixture of Spherical Gaussians

- k clusters generated
- Cluster $i \in \{1, \ldots, k\}$ specified by: mean $\boldsymbol{\mu}_i \in \mathbb{R}^d$, standard deviation $\sigma_i \geq 0$, probability $w_i \geq 0$ such that $w_1 + \cdots + w_k = 1$.
- Generative process: Repeat the following for $j = 1 : n$.
  - Select $i \in \{1, \ldots, k\}$ according to probabilities $w_1, \ldots, w_k$.
  - Select $\boldsymbol{a}_j \sim \mathcal{N}(\boldsymbol{\mu}_i, \sigma_i^2)$

Previous work on recovery of mixture of Gaussians

- Panahi et al. (2017) proved that for an appropriate range of λ and for certain ranges of parameters, SON clustering can correctly identify *all* points if the input is a mixture of spherical Gaussians and the number of points n is not too large.
- Bound on n required by their result because they assume a slab of space separating means devoid of sample points.
- Other related work: Radchenko & Mukherjee (2017), Mixon et al. (2017) on Peng-Wei SDP clustering.

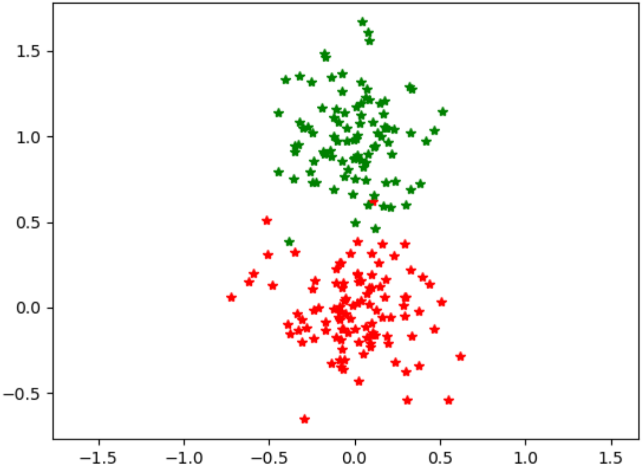
Our result (Jiang, V., Zhai)

- Assume an upper bound on $\sigma_1, \ldots, \sigma_k$ in terms of $\min_{1 \le i < i' \le k} \|\boldsymbol{\mu}_i - \boldsymbol{\mu}_{i'}\|$ and a lower bound on $\min\{w_1, \ldots, w_k\}$.
- Then SON clustering with the correct choice of λ recovers all points within distance $\theta\sigma_i$ of $\boldsymbol{\mu}_i$.
- "Recovers" means that for a particular i, the points in the previous bullet are in the same cluster, and these clusters (as i varies) are disjoint.
- This holds with probability exponentially close to 1 as $n \to \infty$.

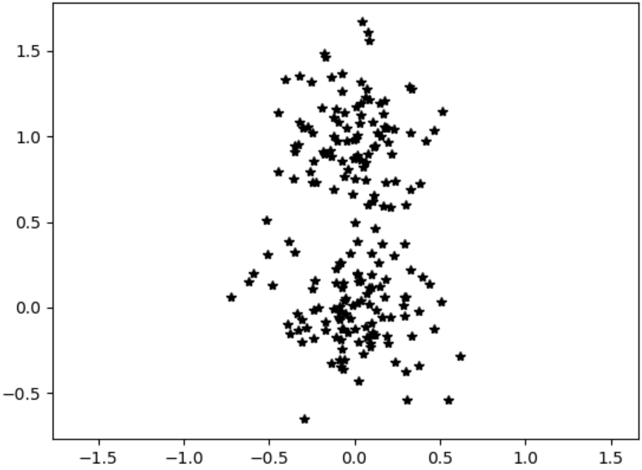
Our proof technique

- Relies on first-order necessary condition for optimality developed by CGR.
- CGR prove that a clustering is attained if and only if certain subgradients can be constructed that satisfy a system of linear equations and inequalities.
- We take the minimum-norm solution to the CGR linear equations.
- Then we argue that with probability exponentially close to 1, this solution also satisfies the inequalities.

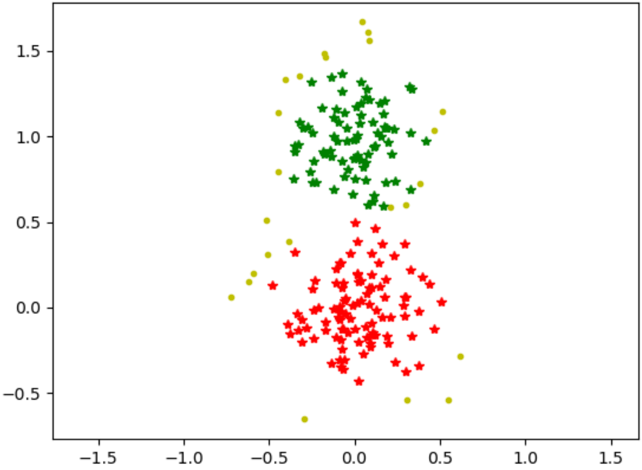
It works for the case $\sigma = 0.25$



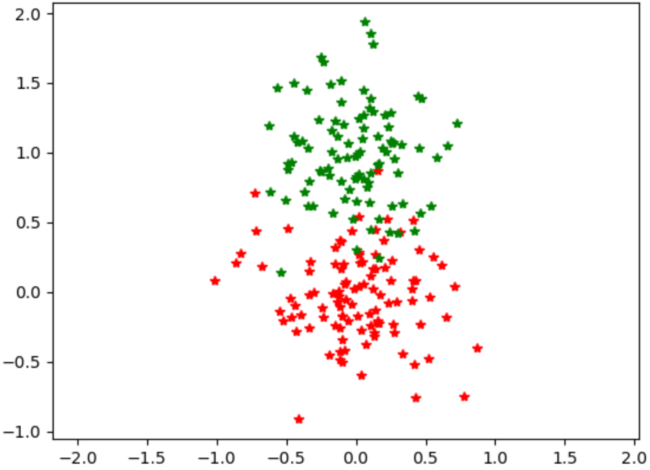
It works for the case $\sigma = 0.25$



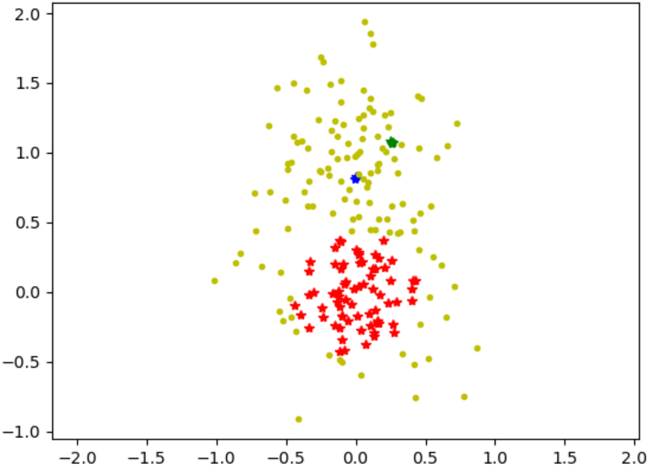
It works for the case $\sigma = 0.25$



. . . but not for the case $\sigma = 0.35$



... but not for the case $\sigma = 0.35$



Contents

Termination of SON clustering

- Recall: two points j, j' are in the same cluster iff the SON optimizer $(\mathbf{x}_1^*, \ldots, \mathbf{x}_n^*)$ satisfies $\mathbf{x}_j^* = \mathbf{x}_{j'}^*$
- But all known algorithms for SOCP are iterative, so the condition in the previous bullet holds only in the infinite limit.
- Use a tolerance ϵ? But how to pick? And what if $\|\mathbf{x}_j^* - \mathbf{x}_{j'}^*\| < \epsilon$, $\|\mathbf{x}_{j'}^* - \mathbf{x}_{j''}^*\| < \epsilon$, but $\|\mathbf{x}_j^* - \mathbf{x}_{j''}^*\| > \epsilon$?

Our termination test (Jiang & V.)

- Requires feasible, approximately optimal, primal and dual solutions
- When the test works, it is guaranteed that the the correct clustering has been computed.

Properties of our test

- The test attempts to determine all clusters. The test may report 'success' or 'failure'.
- **Theorem 1.** If the test reports 'success', then the clusters are correctly identified.
- **Theorem 2.** If a primal-dual path-following close-proximity interior-point algorithm is used, then the test is guaranteed to report 'success' after a finite number of iterations except . . .

When does the test fail?

- … the test may never report 'success' for the particular values of λ at which clusters fuse to form a larger cluster.
- Because of the agglomeration property, there are at most n such discrete values of λ for which the test may never succeed.

SOCP conic form

- The *second order cone* is
  $C_p = \{\boldsymbol{x} \in \mathbb{R}^p : x_1 \geq \|\boldsymbol{x}(2:p)\|\}$.
- SOCP problem in "conic" form:

$$
\begin{array}{rl}
\min & \boldsymbol{c}^T \boldsymbol{x} \\
\text{s.t.} & A\boldsymbol{x} = \boldsymbol{b} \\
& \boldsymbol{x} \in C_{p_1} \times \cdots \times C_{p_l}
\end{array}
\qquad (P)
$$

  where $\boldsymbol{c} \in \mathbb{R}^n$, $A \in \mathbb{R}^{m \times n}$, $\boldsymbol{b} \in \mathbb{R}^m$, and integers $p_1, \ldots, p_l$ summing to n are given.
- Interior-point methods typically assume input is in conic form.

Dual form

- Dual conic form:

$$\begin{array}{rl} \max & \boldsymbol{b}^T \boldsymbol{y} \\ \text{s.t.} & A^T \boldsymbol{y} + \boldsymbol{s} = \boldsymbol{c} \\ & \boldsymbol{s} \in C_{p_1} \times \cdots \times C_{p_l} \end{array} \qquad (D)$$

- Can show: if $\boldsymbol{x}$ feasible for (P), $\boldsymbol{y}$ feasible for (D), then $\boldsymbol{c}^T \boldsymbol{x} \geq \boldsymbol{b}^T \boldsymbol{y}$. Follows from the fact that if $\hat{\boldsymbol{x}}, \hat{\boldsymbol{s}} \in C_p$ then $\hat{\boldsymbol{x}}^T \hat{\boldsymbol{s}} \geq 0$.
- *Weak duality* follows: If $\boldsymbol{x}$ feasible for (P) and $\boldsymbol{y}$ feasible for (D) and $\boldsymbol{c}^T \boldsymbol{x} = \boldsymbol{b}^T \boldsymbol{y}$, then both are optimal.

Converting SON clustering to conic form

Introduce auxiliary variables: $s_1, \ldots, s_n$; $\mathbf{z}_1, \ldots, \mathbf{z}_n$; $t_{ij}\ \forall 1 \leq i < j \leq n$; $\mathbf{y}_{ij}\ \forall 1 \leq i < j \leq n$, yielding

$$
\begin{array}{lll}
\min & \sum_{i=1}^{n} s_i + \lambda \sum_{1 \leq i < j \leq n} t_{ij} & \\
s.t. & \mathbf{z}_i = \mathbf{x}_i - \mathbf{a}_i & \forall i = 1, \ldots, n, \\
 & s_i \geq \|\mathbf{z}_i\|^2 / 2 & \forall i = 1, \ldots, n, \\
 & \mathbf{y}_{ij} = \mathbf{x}_i - \mathbf{x}_j & \forall 1 \leq i < j \leq n \\
 & t_{ij} \geq \|\mathbf{y}_{ij}\| & \forall 1 \leq i < j \leq n.
\end{array}
$$

Converting SON clustering to conic form

Introduce auxiliary variables: $s_1, \ldots, s_n$; $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_n$; $t_{ij}\ \forall 1 \le i < j \le n$; $\boldsymbol{y}_{ij}\ \forall 1 \le i < j \le n$, yielding

$$
\begin{array}{lll}
\min & \sum_{i=1}^{n} s_i + \lambda \sum_{1 \le i < j \le n} t_{ij} - n/2 & \\
s.t. & \boldsymbol{z}_i = \boldsymbol{x}_i - \boldsymbol{a}_i & \forall i = 1, \ldots, n, \\
 & s_i \ge \|\boldsymbol{z}_i\|^2/2 + 1/2 & \forall i = 1, \ldots, n, \\
 & \boldsymbol{y}_{ij} = \boldsymbol{x}_i - \boldsymbol{x}_j & \forall 1 \le i < j \le n \\
 & t_{ij} \ge \|\boldsymbol{y}_{ij}\| & \forall 1 \le i < j \le n.
\end{array}
$$

Writing quadratic constraint in conic form

$$
\begin{aligned}
& s \geq \|z\|^2/2 + 1/2 && \iff \\
& s^2/2 \geq z_1^2/2 + \cdots + z_d^2/2 + s^2/2 - s + 1/2 && \iff \\
& s^2/2 \geq z_1^2/2 + \cdots + z_d^2/2 + u^2/2; \quad u = s - 1 && \iff \\
& s \geq \|(z; u)\|; \quad u = s - 1. &&
\end{aligned}
$$

So n additional auxiliary variables $u_1, \ldots, u_n$ needed.

SON dual

$$
\begin{aligned}
\max \quad & \sum_{i=1}^{n} \boldsymbol{a}_i^T \boldsymbol{\beta}_i + \sum_{i=1}^{n} \gamma_i \\
s.t. \quad & \sum_{j=i+1}^{n} \boldsymbol{\delta}_{ij} - \sum_{j=1}^{i-1} \boldsymbol{\delta}_{ji} + \boldsymbol{\beta}_i = \mathbf{0} && \forall i = 1, \ldots, n, \\
& \lambda \geq \|\boldsymbol{\delta}_{ij}\| && \forall 1 \leq i < j \leq n, \\
& 1 - \gamma_i \geq \|(\boldsymbol{\beta}_i; \gamma_i)\| && \forall i = 1, \ldots, n
\end{aligned}
$$

The test

1. Compute μ, the duality gap.
2. Choose $i \in \{1, \dots, n\}$ arbitrarily. Create a cluster $\{j : \|\mathbf{x}_i - \mathbf{x}_j\| \le \mu^{3/4}\}$ (including i itself).
3. Delete all these points, and then repeat Step 2 until all points are used up.
4. Compute CGR subgradients from dual variables. The subgradients certify that points clustered in Step 2 belong in the same cluster.
5. Check that no two clusters are distance $\le c_n \mu^{1/2}$ of each other. This certifies that no cluster identified in Step 2 is actually a subcluster of a larger cluster.

Establishing Theorem 1 (correctness)

- Instead of standard subgradients, use CGR subgradients. These are local to each cluster and seem to be vital for our test.
- The test for distinctness of the clusters is a straightforward argument relying on strong convexity of the original objective.

Establishing Theorem 2 (eventual success)

- Proof of Theorem 2 requires a deep dive into duality.
- Ingredient of Theorem 2 proof is a result by Luo, Sturm and Zhang (1998) that, provided the optimizer satisfies strict complementarity, interior point iterates are $O(\mu)$ away from optimizer, where μ is the duality gap (scaled central path parameter).

Strong duality

- *Strong duality* means that the primal and dual both attain equal optimal values.
- Strong duality holds for SOCP provided primal and dual have an interior feasible point (Slater condition).
- If strong duality holds, then primal and dual optimizers satisfy complementary slackness.
- This is always the case for SON clustering formulation.

Complementary slackness

- For a primal-dual feasible solution $(\boldsymbol{x}, (\boldsymbol{y}, \boldsymbol{s}))$, both are optimal if $\boldsymbol{x}^T\boldsymbol{s} = 0$.
- SON clustering case: Primal and dual are optimal if $(t_{ij}; \boldsymbol{y}_{ij})^T(\lambda; \boldsymbol{\delta}_{ij}) = 0 \ \forall 1 \leq i < j \leq n$ and $(s_i; \boldsymbol{z}_i; u_i)^T(1 - \gamma_i; \boldsymbol{\beta}_i; \gamma_i) = 0$ $\forall i = 1, \ldots, n$.
- *Strict complementarity* (Alizadeh & Goldfarb, 2003) for a pair of feasible primal-dual variables $((\boldsymbol{x}_1, \ldots, \boldsymbol{x}_l), (\boldsymbol{s}_1, \ldots, \boldsymbol{s}_l)) \in (C_{p_1} \times \cdots \times C_{p_l})^2$:

$$\forall i = 1, \ldots, l \quad \begin{cases} \boldsymbol{x}_i^T\boldsymbol{s}_i = 0, \\ \boldsymbol{x}_i = \boldsymbol{0} \Rightarrow (\boldsymbol{s}_i)_1 > \|\boldsymbol{s}_i(2:p_i)\|, \\ \boldsymbol{s}_i = \boldsymbol{0} \Rightarrow (\boldsymbol{x}_i)_1 > \|\boldsymbol{x}_i(2:p_i)\|. \end{cases}$$

SON strict complementarity

- **Theorem (Jiang&V.).** The SON clustering formulation has a strictly complementary optimizer provided λ is not exactly at a value when clusters fuse.
- Thus, there are $\leq n$ discrete values of λ for which strict complementarity fails.
- Our test requires nearness to a strictly complementary solution.
- So this existence theorem underpins the "eventual success" theorem regarding our test.

Explanation of failure case

- Not surprising that test fails when λ is exactly at a fusion value λ^*, since any arbitrarily small negative perturbation $\lambda^* - \epsilon$ yields a different clustering.
- In other words, complete cluster identification for these values of λ^* is *ill-posed*; unreasonable to expect an algorithm to satisfy a guarantee for such a problem.

Other algorithms

- The test can be used with any algorithm that produces both primal and dual variables, although for algorithms other than interior-point, we cannot guarantee that it will report success.
- Primal-only algorithms: subgradient descent (Hocking et al., 2011), stochastic pairwise updates (Panahi et al., 2017).
- Primal-dual algorithms: Interior point (Lindsten et al., 2011), **ADMM (Chi & Lange, 2018)**, Semismooth Newton (Yuan et al., 2018)

Contents

Some limitations of SON clustering

- The convex hulls of the clusters found by SON clustering must be disjoint (Nguyen & Mamitsuka 2021)
- If the data points are densely sampled in two disjoint unit-radius disks, SON clustering will be able to separate them only if the disks have a certain positive gap between them (Dunlap & Mourrat 2022)
- And we saw that it fails for Gaussian mixture models when the means are too close w.r.t. the noise.

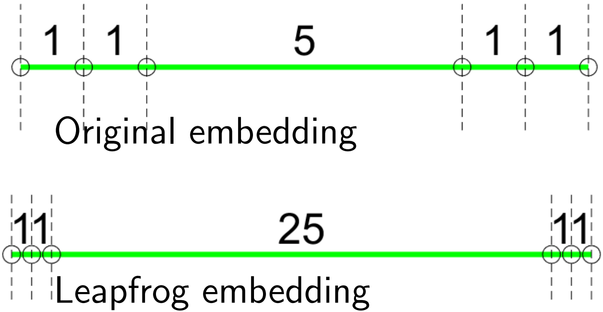
Strengthening the recovery guarantees (Jiang, Tan, & V.)

- Given data points $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n$, come up with new data points $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n$ that are easier to cluster.
- Our approach: leapfrog distance, multidimensional scaling and Euclidean distance matrices.
- Our leapfrog distance technique is agnostic w.r.t. the clustering method, but in the case of SON clustering we can actually prove something.

Leapfrog distance

- Given n points $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n$, let the complete graph K_n on these points be labeled with distances $\|\boldsymbol{a}_i - \boldsymbol{a}_j\|^2$.
- Define $\mathrm{LF}(\boldsymbol{a}_i, \boldsymbol{a}_j)$ to be the length of the shortest path from $\boldsymbol{a}_i$ to $\boldsymbol{a}_j$ in the graph defined in the last bullet.
- Our new data points $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n$ are embeddings in $\mathbb{R}^{d'}$ (with possibly $d' < d$) so that the distances between the $\boldsymbol{b}_i$'s are approximately the corresponding LF distances.

Example in the $d = 1$ case



Characterizing LF distance ($d = 1$)

Assume the n data points are chosen at random according to a PDF f defined on $\mathbb{R}^d$.

- ($d = 1$ case.) Suppose the PDF is everywhere positive (e.g., mixture of Gaussians). Then, with probability exponentially close to 1, assuming $a_i < a_j$,

$$\mathrm{LF}(a_i, a_j) = \frac{2}{n} \int_{a_i}^{a_j} \frac{dx}{f(x)} + o(1/n).$$

Characterizing LF distance ($d \geq 1$)

Assume the n data points are chosen at random according to a PDF f defined on $\mathbb{R}^d$.

- ($d \geq 1$ case.) Suppose f is bounded below by $\theta > 0$ on a subset $\Omega \subset \mathbb{R}^d$ satisfying a shape condition (basically: connected, with no thin parts). Then for any $\boldsymbol{a}_i, \boldsymbol{a}_j \in \Omega$, with probability exponentially close to 1, $\mathrm{LF}(\boldsymbol{a}_i, \boldsymbol{a}_j) \leq O(n^{-1/d+\eta})$ for any $\eta > 0$.

Deriving an embedding

- In order to apply SON clustering to LF distances, an embedding is needed.
- In the case $d = 1$, the embedding is trivially found from the distances.
- In the case $d > 1$, we use a technique from Euclidean distance matrix theory.

Algorithm to compute embedding

- **Step 1.** Form the squared leapfrog distance matrix D.
- **Step 2.** Form the "Gram matrix": $G := (D(:,1)\boldsymbol{e}^T + \boldsymbol{e}D(1,:) - D)/2$, where $\boldsymbol{e}$ is the vector of all 1's.
- **Step 3.** Factor $G = Q\Lambda Q^T$ (eigendecomposition). Assume eigenvalues listed in order greatest to least magnitude.
- **Step 4.** Define

$$B := [\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n] = |\Lambda(1:m, 1:m)|^{1/2} Q(:, 1:m)^T$$

Motivation for these formulas

$G := (D(:,1)\boldsymbol{e}^T + \boldsymbol{e}D(1,:) - D)/2$
$G \to Q\Lambda Q^T$
$B := [\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n] = |\Lambda(1:m, 1:m)|^{1/2} Q(:, 1:m)^T$

- Let $X \in \mathbb{R}^{m\times n}$ contain coordinates of n points in $\mathbb{R}^m$. Assume $n \geq m$.
- The $n \times n$ *Euclidean distance matrix* D is defined by $D(i,j) := \|X(:,i) - X(:,j)\|^2$.
- The matrix B determined by these formulas is equal to X, up to translation and rotation.

How to choose m

- In the previous slide, m is the embedding dimension.
- For the method to work, m should be at least the number of clusters.
- In practice, this is not known in advance, so we use a heuristic of a decrease in the magnitude of the eigenvalues.

Main theorem about this technique ($d = 1$)

Assume the a_i's are chosen according to a PDF.

- **Theorem 1.** In the $d = 1$ case, for an equally weighted mixture of two Gaussians with the same variances, using the LF embedding increases the maximal value of σ for which the SON clustering theorem guarantees recovery.

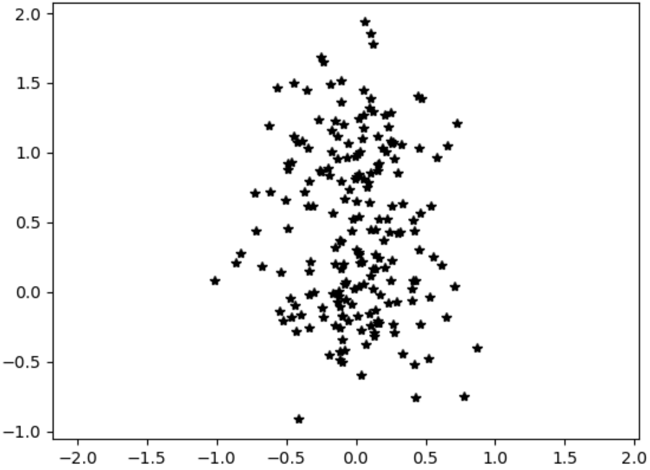
Main theorem about this technique ($d \geq 1$)

Assume the $\boldsymbol{a}_i$'s are chosen according to a PDF.

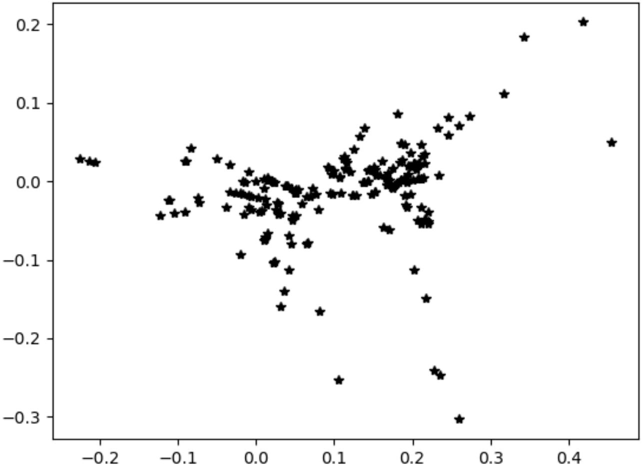
- **Theorem 2.** In the $d \geq 1$ case, if the clusters are chosen from a distribution f supported on disjoint union $\Omega_1 \cup \cdots \cup \Omega_k$ such that
  - each Ω_i is well shaped (connected, no thin parts), and
  - $f(\boldsymbol{x}) \geq \theta > 0 \; \forall \boldsymbol{x} \in \Omega_1 \cup \cdots \cup \Omega_k$,

  then recovery is guaranteed with probability exponentially close to 1.

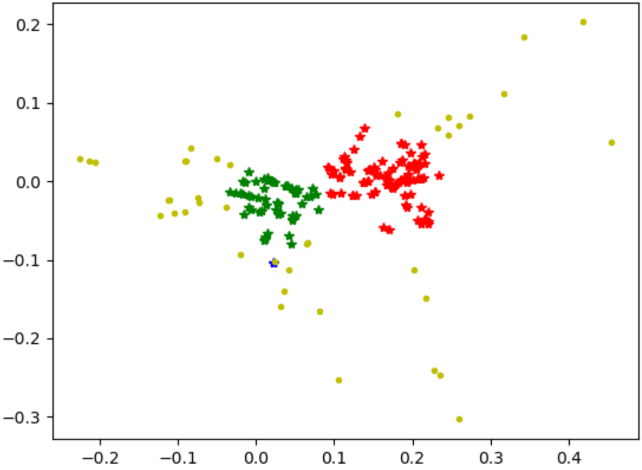
Mixture of Gaussians (a_i's)



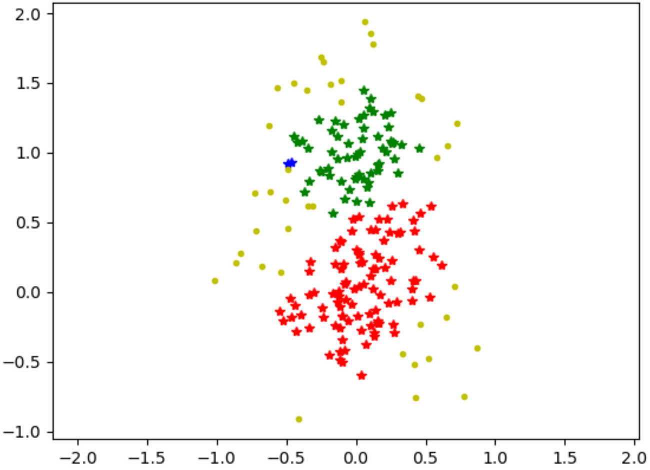
Mixture of Gaussians - re-embedded ($\boldsymbol{b}_i$'s)



SON clusters on $\boldsymbol{b}_i$'s

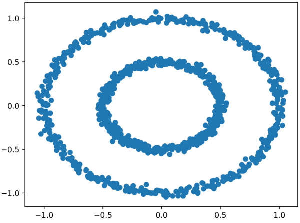


Clustering results pulled back to a_i's

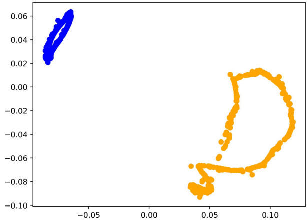


Clustering of concentric circles

Concentric Circles



Original coordinates



MDS embedding

Discussion

- Rigorous (partial-information) termination test when λ is close to a fusion value?
- Complexity result regarding termination
- Tighter characterization of leapfrog distance for $d > 1$?
- Can sum-of-norms clustering be solved faster? Recent work by Yuan, Chang, Sun, Toh.