The Troubled Geometry of Data Visualization

Noah Bergam

Joint work with

Szymon Snoeck

Nakul Verma

on two recent papers:

Published as a conference paper at ICLR 2026

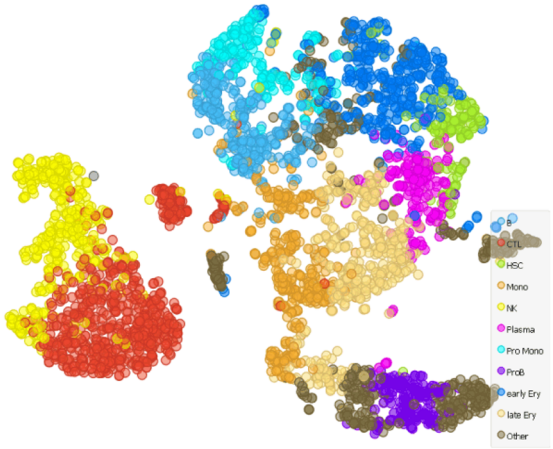
T-SNE EXAGGERATES CLUSTERS, PROVABLY

Proceedings of Machine Learning Research vol 313:1–30, 2026 37th International Conference on Algorithmic Learning Theory

Compressibility Barriers to Neighborhood-Preserving Data Visualization

What is data visualization?

Given (dis)similarity data about objects $\mathrm{X} = \{x_1, \ldots, x_n\}$, embed them "usefully" as $\{y_1, \ldots, y_n\} \subset \mathbb{R}^{\leq 3}$.

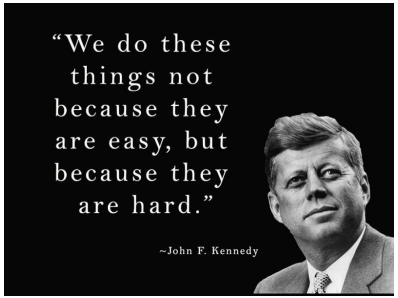


cells

words

algorithm used above: **t-distributed stochastic neighbor embedding (t-SNE)**

Why is data visualization important?



Argument 1: **exploratory data analysis**

- partial evidence, a tool for hypothesis generation

Argument 2: eases **scientific communication**

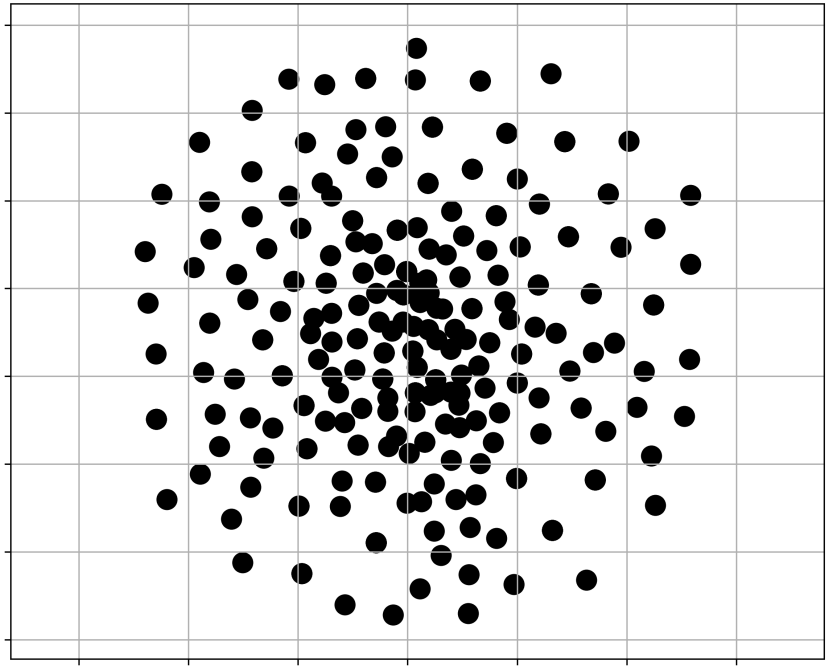
- clear visual evidence, makes papers prettier

Argument 3: the hardest type of **dimension reduction**

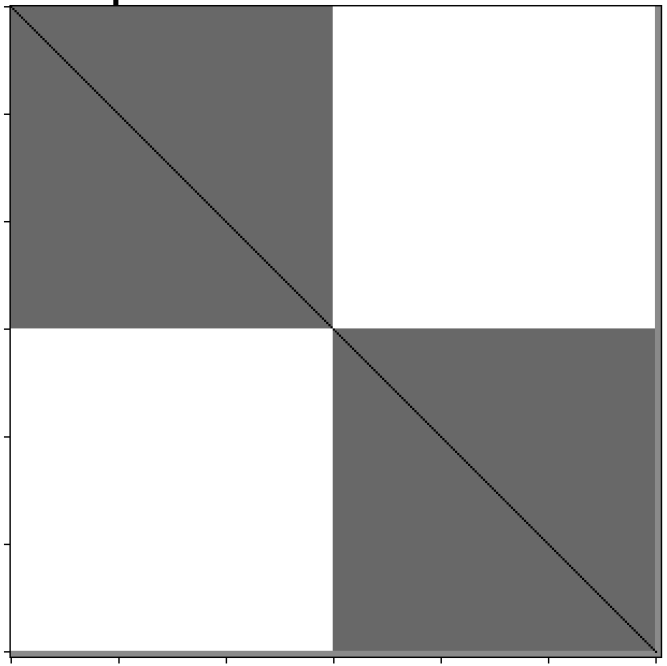
fundamental tradeoff for dimension reduction:
[structure preserved] vs **[output dimension]**

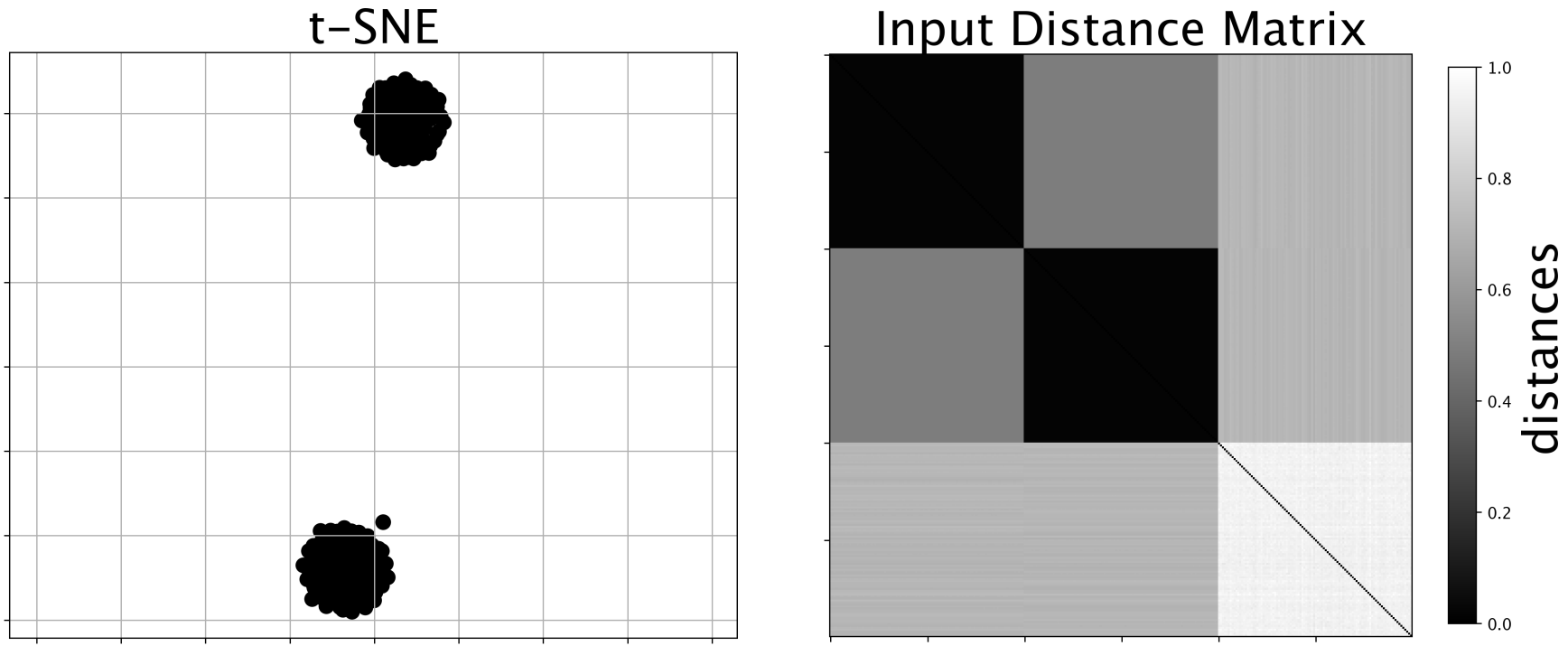
data visualization pushes this to the limit…
and it shows…

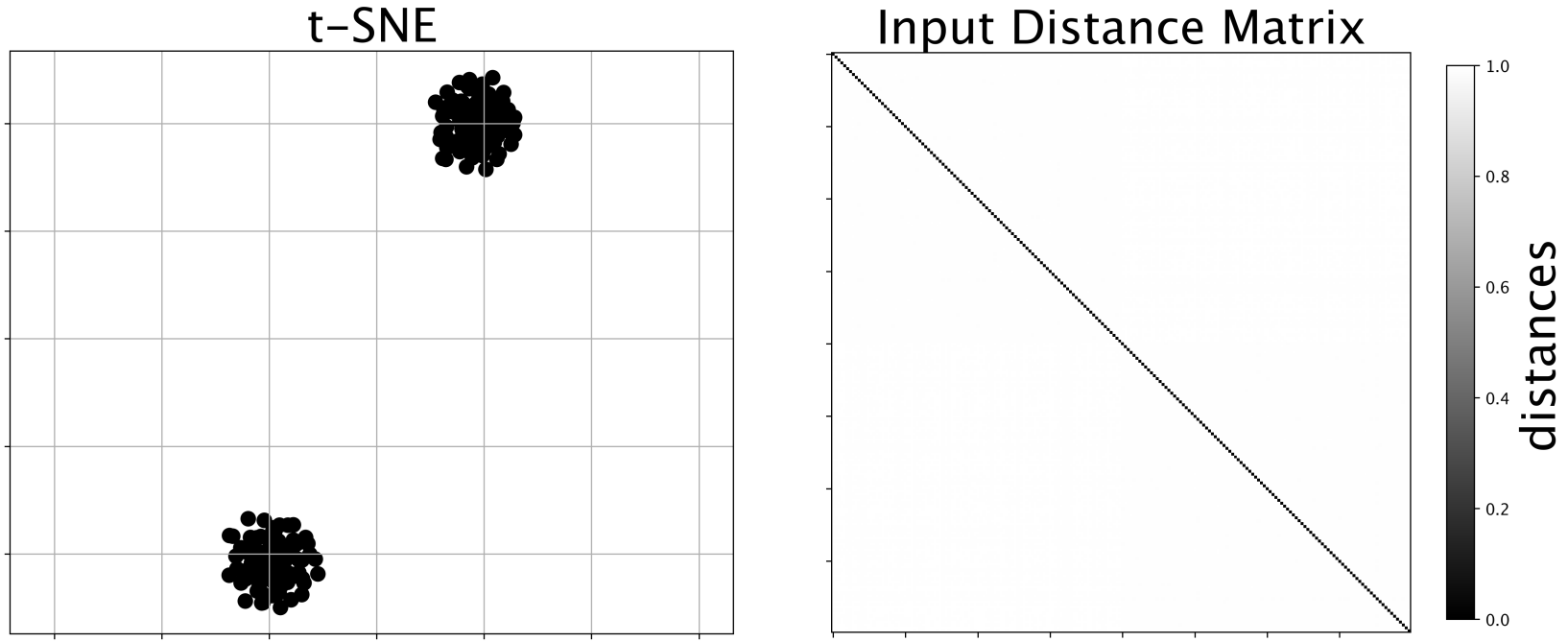
t-SNE



Input Distance Matrix



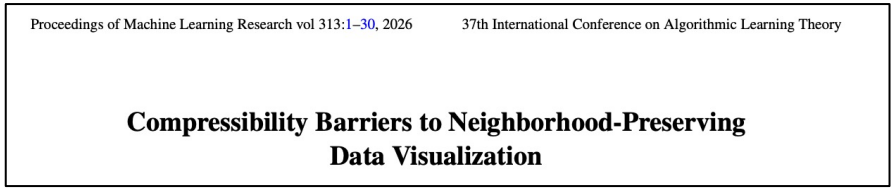

t-SNE
Input Distance Matrix
distances
1.0
0.8
0.6
0.4
0.2
0.0


t-SNE
Input Distance Matrix
1.0
0.8
0.6
0.4
0.2
0.0
distances

Roadmap

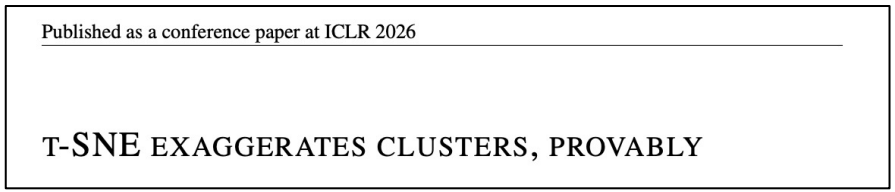
(1) Introduction

(2) Formalizing data visualization


Proceedings of Machine Learning Research vol 313:1–30, 2026 37th International Conference on Algorithmic Learning Theory

Compressibility Barriers to Neighborhood-Preserving Data Visualization

(3) Understanding t-SNE's failure modes

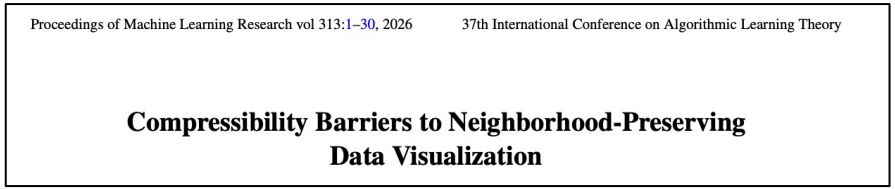

Published as a conference paper at ICLR 2026

T-SNE EXAGGERATES CLUSTERS, PROVABLY

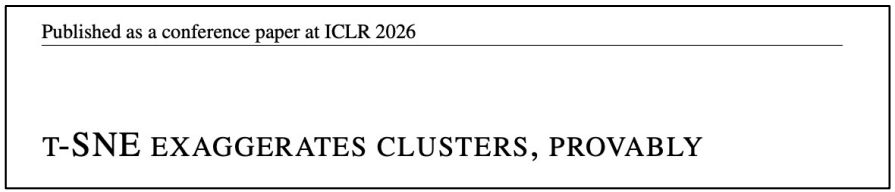
Roadmap

(1) Introduction

(2) **Formalizing data visualization**



(3) Understanding t-SNE's failure modes



What can we hope to preserve?

Say $X = \{x_1, \dots, x_n\} \subset \mathbb{R}^{n-1}$
Want to embed in $\{y_1, \dots, y_n\} \subset \mathbb{R}^d$.

isometry
- $d = \Omega(n)$

approx. $(1+\epsilon)$ isometry
- $d = \Omega(\log n / \epsilon^2)$ [LN'16]

neighbor preservation?

The sphericity model

$\mathrm{X} = \{x_1, \ldots, x_n\}$ → neighborhood graph $\mathrm{G} = (X, E)$.

Let $\dim(G)$ ("sphericity of G") = smallest d s.t. ∃f: $\mathrm{X} \to \mathbb{R}^d$ where

$$(x, x') \in E \leftrightarrow \| f(x) - f(x') \| \leq 1$$

(We call f a "spherical embedding of G")

Main Result: for most G, $\dim(\mathrm{G}) \asymp n$. [Reiterman89]

Main Result: for most G, $\dim(\mathrm{G}) \asymp n$. [Reiterman89]

Proof sketch:

Upper Bound: PCA of (modified) adjacency matrix A.

Lower Bound: polynomial sign counting.

Main Result: for most G, $\dim(\mathrm{G}) \asymp n$. [Reiterman89]

Proof sketch:

Upper Bound: PCA of (modified) adjacency matrix A.

Observe $nI - A$ is PSD.

Take $XX^T = nI - A$. If $X = [x_1, \ldots, x_n]$, then

$$\| x_i - x_j \| = \sqrt{2n^2 - 2 \cdot 1[(i,j) \in E]}$$

Main Result: for most G, $\dim(\mathrm{G}) \asymp n$. [Reiterman89]

Proof sketch: Lower Bound: polynomial sign counting.

Suppose all graphs have sphericity $\leq d$

Let $\mathrm{X}^{\mathrm{G}} = \left[x_1^G, \ldots, x_n^G\right] \in \mathbb{R}^{d \times n}$ be a spherical embedding of G.

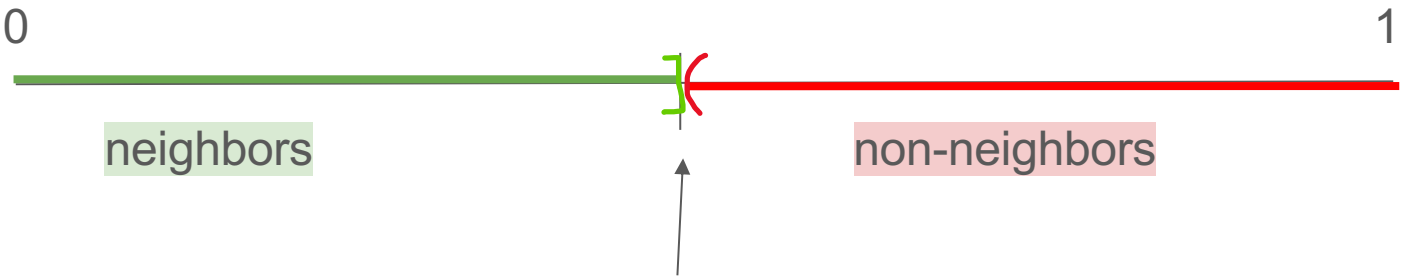
Define $p_{ij}(X) = \| x_i - x_j \|^2 - 1$ for $i \neq j \in [n]$.

Each G invokes distinct sign pattern on $\{p_{ij}\}$.

Need high dimension to accommodate! (cf. Warren's Theorem)

Further difficulties with sphericity

$$\| f(x) - f(x') \| \leq 1 \Leftrightarrow (x, x') \in E$$

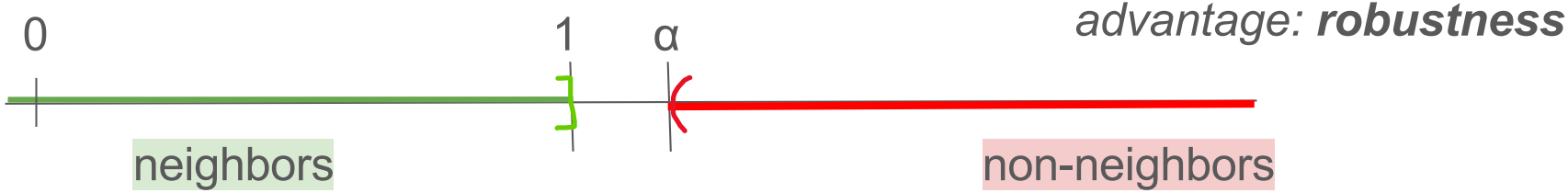


Main Idea: decouple thresholds

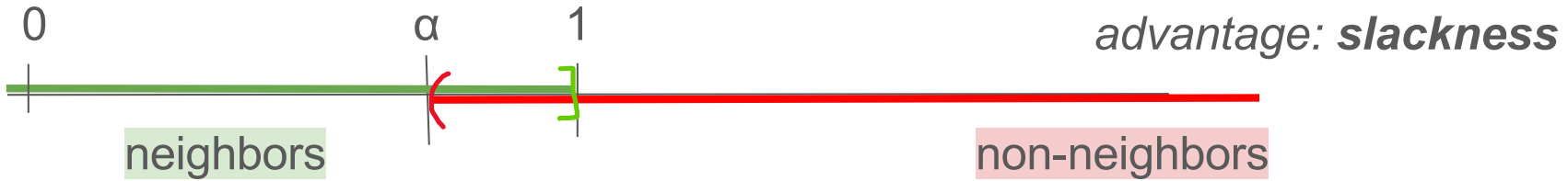
Definition: $\dim_\alpha(G)$ for $G = (X, E)$ is the smallest d such that $\exists$ f: X $\to \mathbb{R}^d$

$$(u, v) \in E \implies \rho\big(f(u), f(v)\big) < 1$$

$$(u, v) \notin E \implies \rho\big(f(u), f(v)\big) \geq \alpha$$

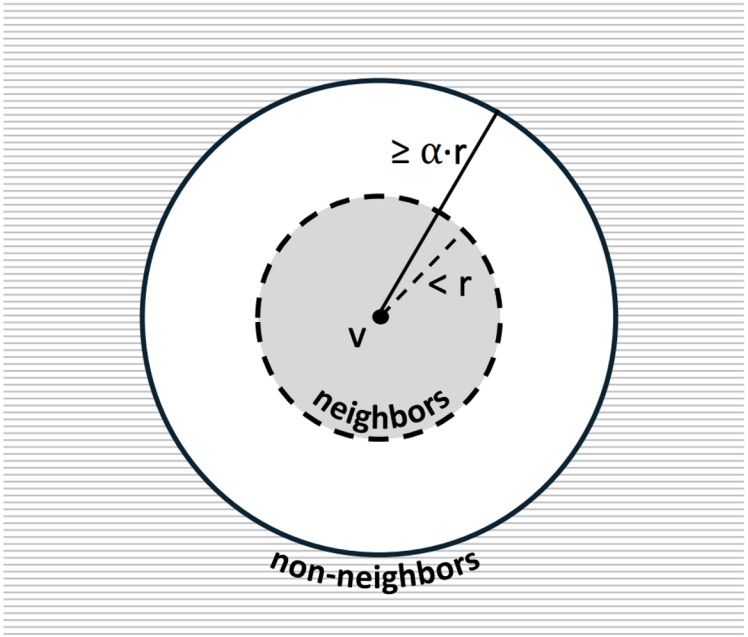


advantage: ***robustness***

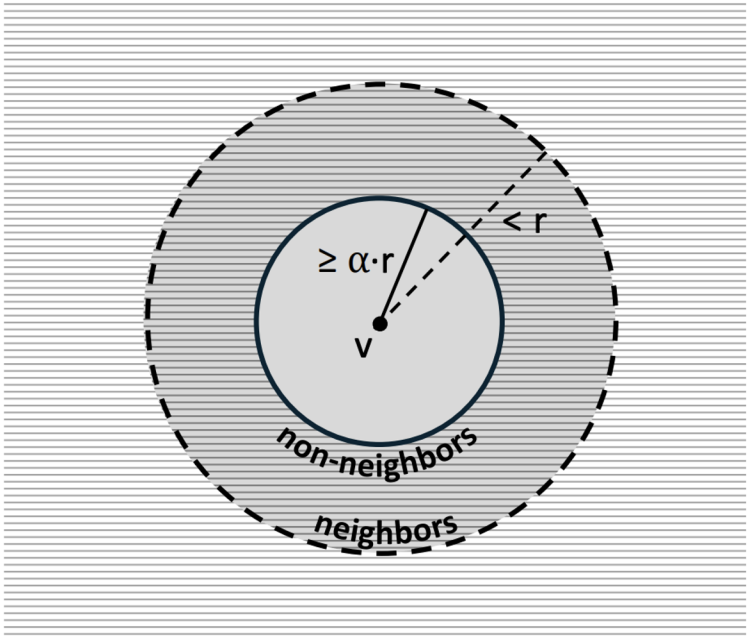


advantage: ***slackness***

Ideal (recoverable) Case



General Case

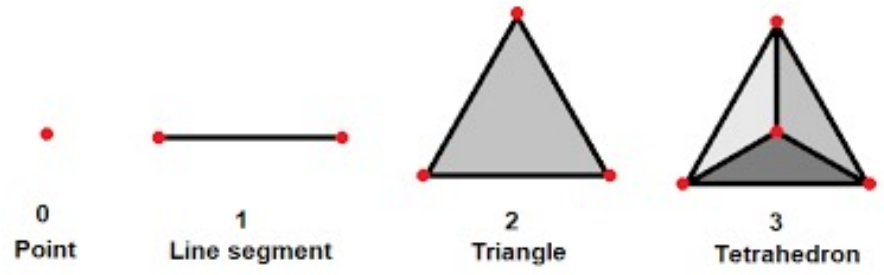


The slack case is non-trivial!

Definition: $\dim_\alpha(G)$ for $G = (X, E)$ is the **smallest d** such that $\exists$ f: X $\to \mathbb{R}^d$

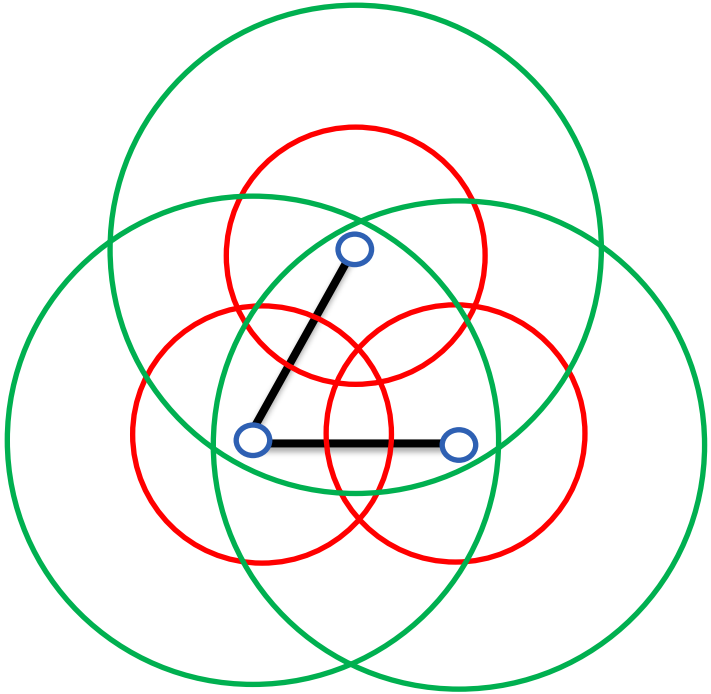
$$(u, v) \in E \implies \rho(f(u), f(v)) < 1$$
$$(u, v) \notin E \implies \rho(f(u), f(v)) \geq \alpha$$

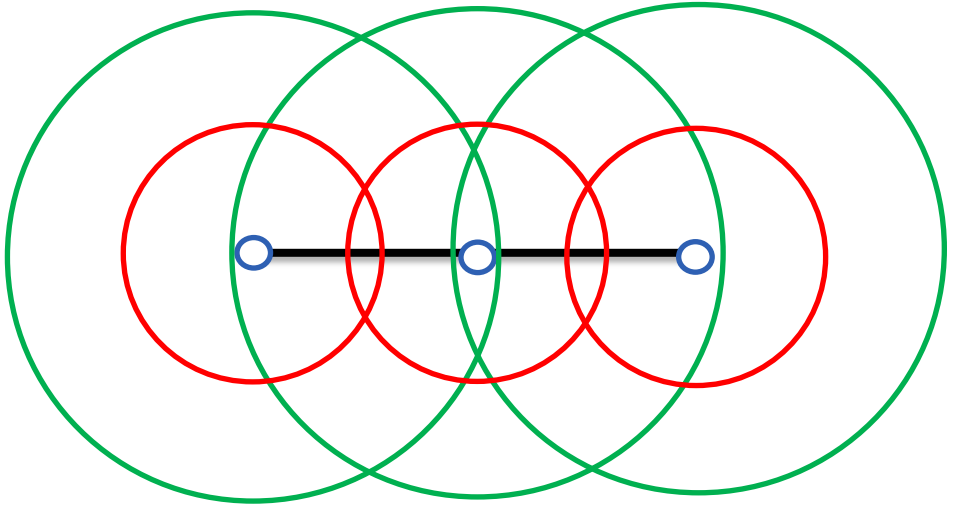
Quibble: If $\alpha < 1$, I can embed *any* graph onto the regular simplex!



Counter: we are *minimizing* dimension

$$G = (V = \{1, 2, 3\}, \qquad E = \{\{1, 2\}, \{2, 3\}\})$$





minimizing dimension
naturally avoids the simplex case

We initiate a study of α-preservation (and further generalizations of sphericity).

Optimistic Outlook

For almost all n-points graphs

(slack case) $\mathbf{dim}_{\alpha}(\mathbf{G}) \asymp \mathbf{log}\, \boldsymbol{n} \cdot \frac{\mathbf{1}}{\left(\mathbf{1}-\boldsymbol{\alpha}^{\mathbf{2}}\right)^{\mathbf{2}}}$ for $\alpha < 1$

exponentially easier than sphericity!

(robust case) $\mathbf{dim}_{\alpha}(\mathbf{G}) \asymp \mathbf{n}$ for $1 \leq \alpha \leq 1 + 1/\sqrt{n}$

robustness doesn't cost much!

Pessimistic Outlook

Input metric $\mathrm{X} = \{x_1, \ldots, x_n\}$.
Let $G_{X,r}$ = radius-r neighborhood graph

$(\alpha \geq 1)$-neighbor-preservation of $G_{X,r}$ relaxes
isometric embedding of X
but <u>suffers the same $\Omega(n)$ lower bound!</u>

$(\alpha < 1)$-neighbor-preservation of $G_{X,r}$ relaxes
near-isometric $(1/\alpha$-distortion) embedding
but <u>suffers the same $\Omega(\log n)$ lower bound!</u>

Takeaway:
Neighbors are as hard to preserve as distances…
- for *worst case* input.
- for *average case* input.

What about structured cases?

- X = n-samples from k-manifold $\Omega(k)$
- X = n-samples from noisy k-manifold $\Omega(\log n)$
- X = n-samples from k clusters $\Omega(\log n)$

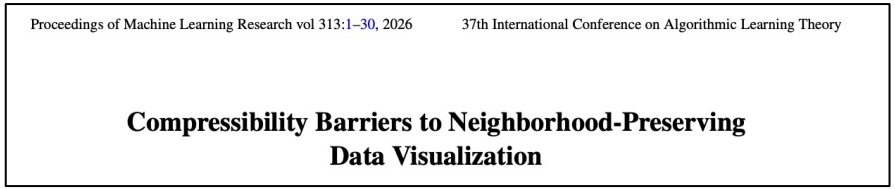
data visualizations must fail to preserve neighbors…

but what do the patterns of failure look like?

Roadmap

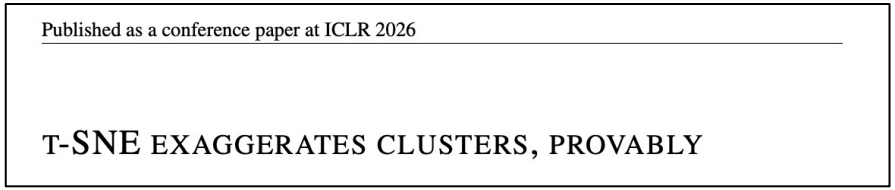
(1) Introduction

(2) Formalizing data visualization


Proceedings of Machine Learning Research vol 313:1–30, 2026 37th International Conference on Algorithmic Learning Theory

Compressibility Barriers to Neighborhood-Preserving Data Visualization

(3) **Understanding t-SNE's failure modes**


Published as a conference paper at ICLR 2026

T-SNE EXAGGERATES CLUSTERS, PROVABLY

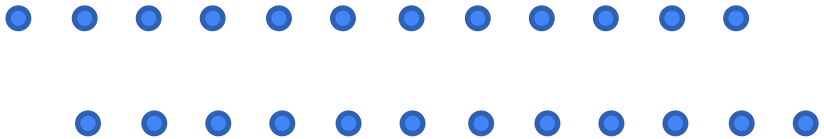
Why focus on t-SNE?

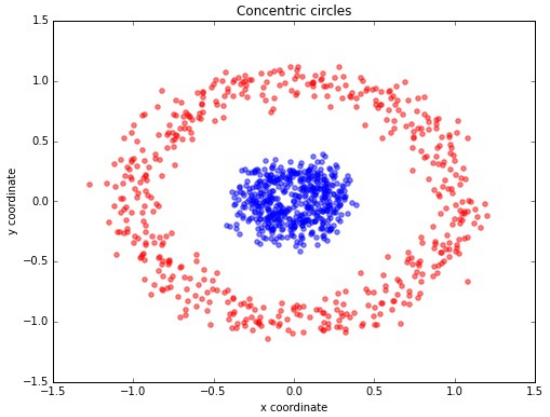
(t-distributed stochastic neighbor embedding)

Limitations of Linear Methods

Random projection (Johnson-Lindenstrauss):
in low dimension, the plot looks Gaussian [DHV'12]

PCA:
well-known failure modes.





(a) JL (b) PCA (c) t-SNE

Figure 1: 2D embeddings of a mixture of 10 Gaussians with pairwise center separation $0.5\times$radius via: (a) random projection (JL), (b) projection to the subspace of top 2 singular vectors (PCA), (c) t-SNE.

How about “simple” nonlinear approaches?
(manifold learning / kernelized PCA)

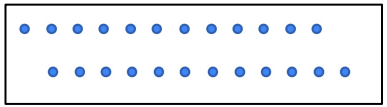
Kernel PCA

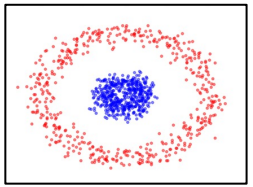
Let $P_{ij} \propto f(x_i, x_j)$ be a kernel of the input.
(for standard PCA, $\mathrm{P} = \mathrm{X}^{\mathrm{T}}\mathrm{X}$)

$$\max_{Y \in \mathbb{R}^{d \times n}} Tr(Y^T P Y) \quad \text{such that} \quad YY^T = I$$

Subsumes spectral clustering, most of “manifold learning”

“right behavior” on these →





Kernel PCA

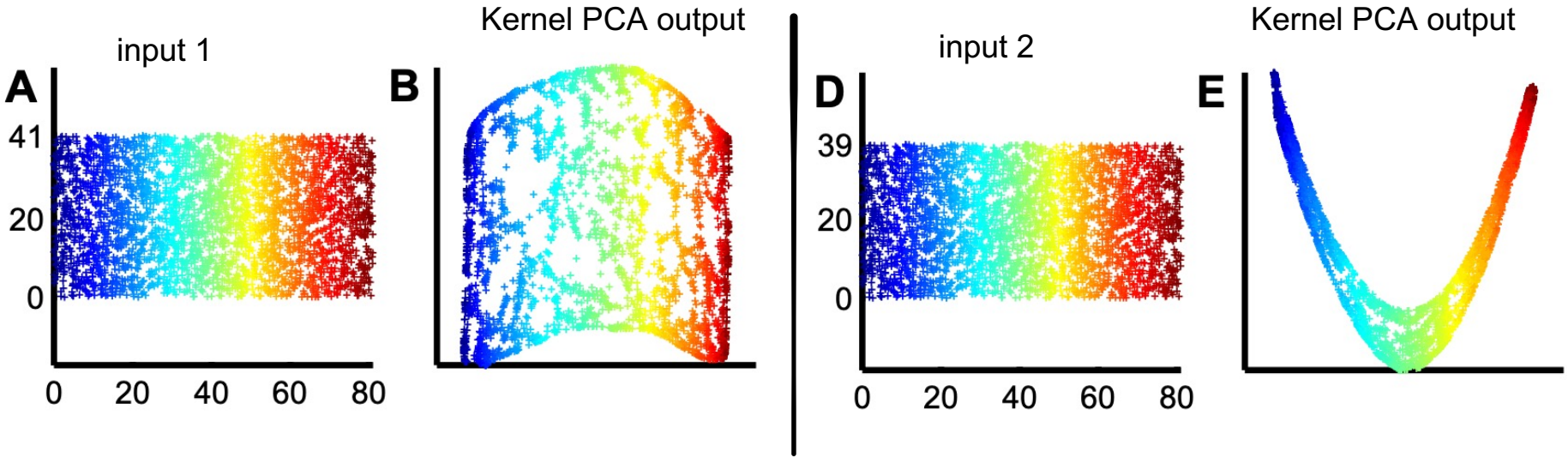
Let $P_{ij} \propto f(x_i, x_j)$ be a kernel of the input.
(for standard PCA, $\mathrm{P} = \mathrm{X}^{\mathrm{T}}\mathrm{X}$)

the problem

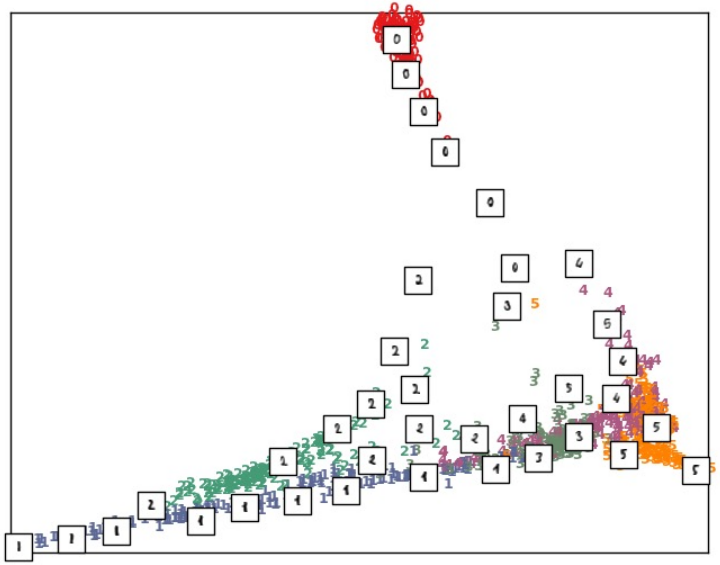
$$\max_{Y \in \mathbb{R}^{d \times n}} Tr(Y^T P Y) \quad \text{such that} \quad \boxed{YY^T = I}$$

This “hard normalization” causes instability and artifacts

“The Price of Normalization” [GZKR’08]

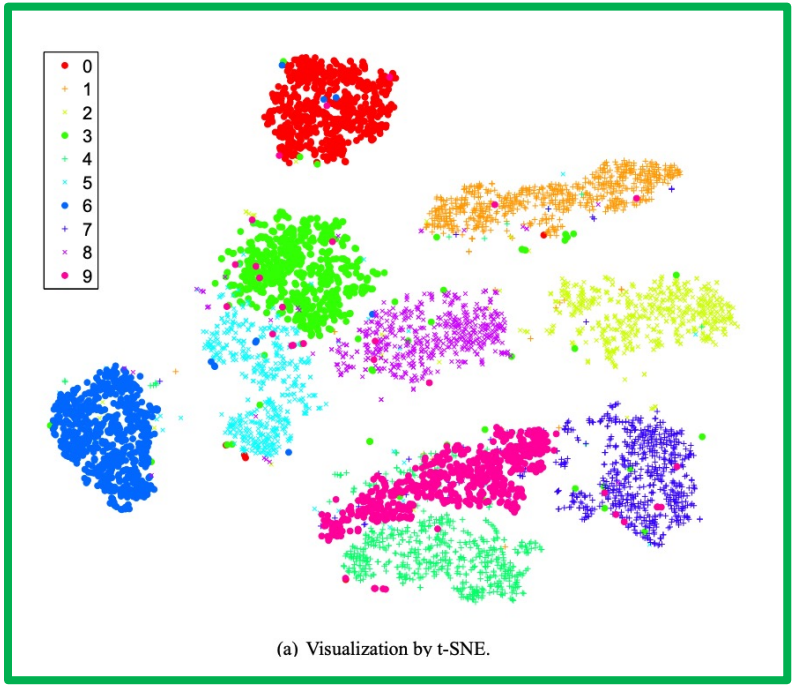


A spikey signature



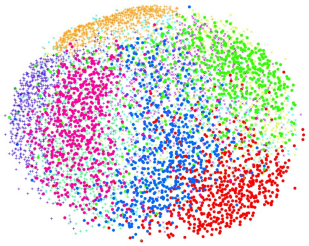
locally linear embedding (LLE) of MNIST

t-SNE "just works"

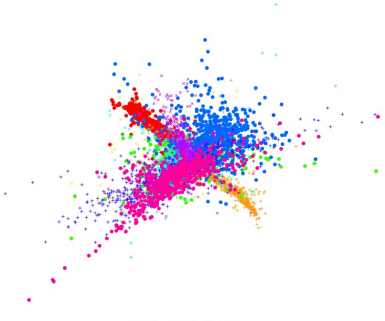


(a) Visualization by t-SNE.

(a) Visualization by Isomap.



(b) Visualization by Sammon mapping.



(b) Visualization by LLE.

The success story

Among the most cited ML papers; a staple in single-cell bio research.

[PDF] Visualizing data using **t-SNE**.

L Van der Maaten, G Hinton - Journal of machine learning research, 2008 - jmlr.org

… **t-SNE** is better than existing techniques at creating a single … large data sets, we show how **t-SNE** can use random walks on … We illustrate the performance of **t-SNE** on a wide variety of …

☆ Save 99 Cite Cited by 56809 Related articles All 43 versions »

What accounts for its (unique) success?

- It uses a “soft” normalization (in contrast to K-PCA)
- There’s an “asymmetry” between input kernel and output kernel.
- (Marketing, momentum, ease of use, simple gradient…)

How does it work (briefly)

From $\mathrm{X} = \{x_1, \ldots, x_n\} \subset \mathbb{R}^{n-1}$, produce “soft k-nearest neighbor” kernel P.
From $\mathrm{Y} = \{y_1, \ldots, y_n\} \subset \mathbb{R}^{2}$, initialize “fat-tailed” kernel Q.

Normalize the kernels, treat them as probability distributions. Solve:

$$\min_{\mathrm{Y}} \mathrm{KL}(\mathrm{P}||\mathrm{Q}_{\mathrm{Y}})$$

Unique asymmetry between P and Q
Normalization of kernels induces a soft normalization

Theory about t-SNE: explaining the success

[SS'17] [LS'17] [AHK'18]: proved various forms of the following:

If the original dataset is well-clustered,
t-SNE outputs a well-clustered visualization

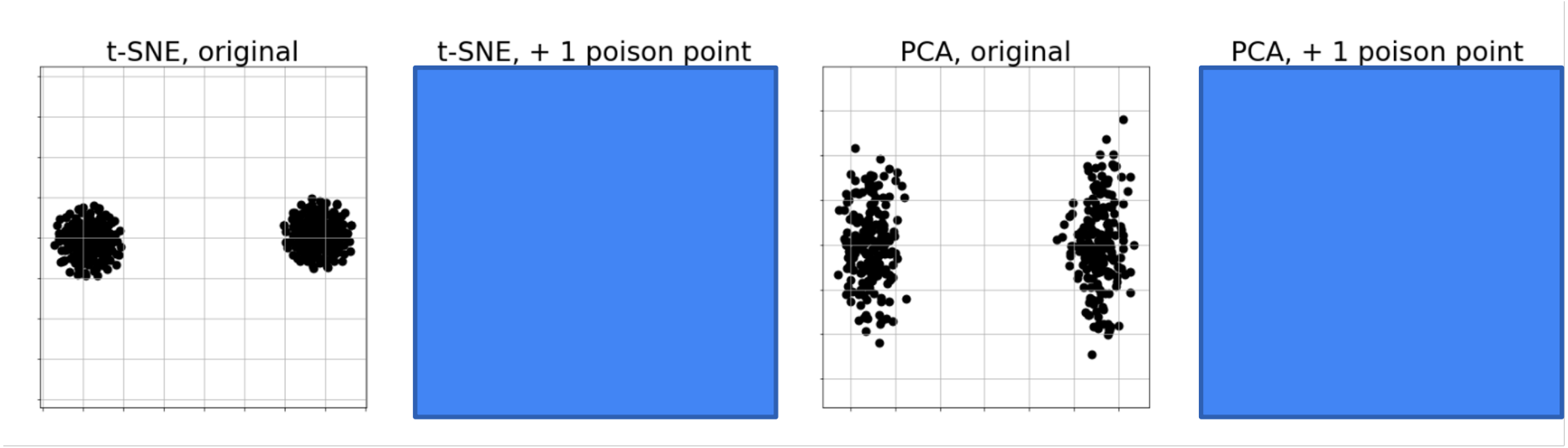
It guarantees t-SNE will generate **true positives.**

No theory literature on false positives or false negatives.

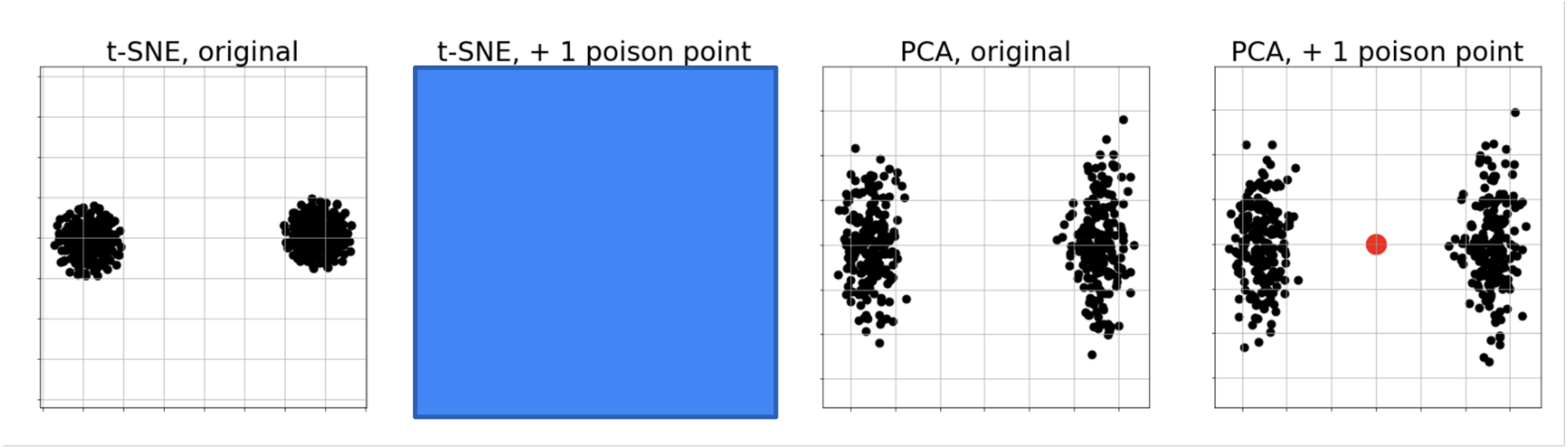
We initiate a theoretical study of t-SNE's failure modes.

Finding 1:
misrepresenting cluster structure

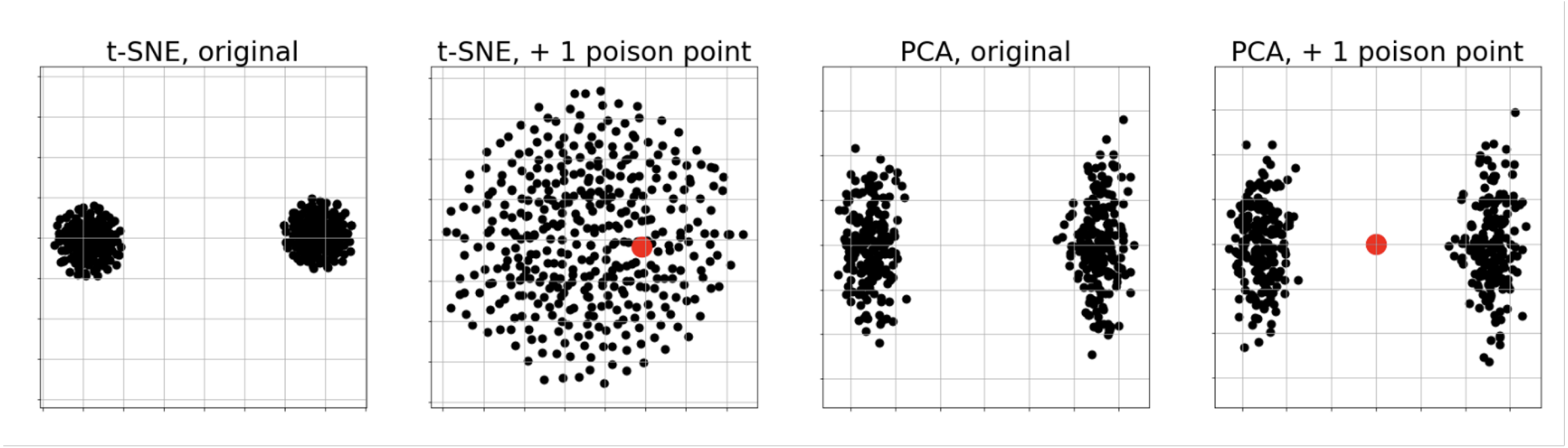
Input dataset: mixture of two well-separated Gaussians



Input dataset: mixture of two well-separated Gaussians



Input dataset: mixture of two well-separated Gaussians

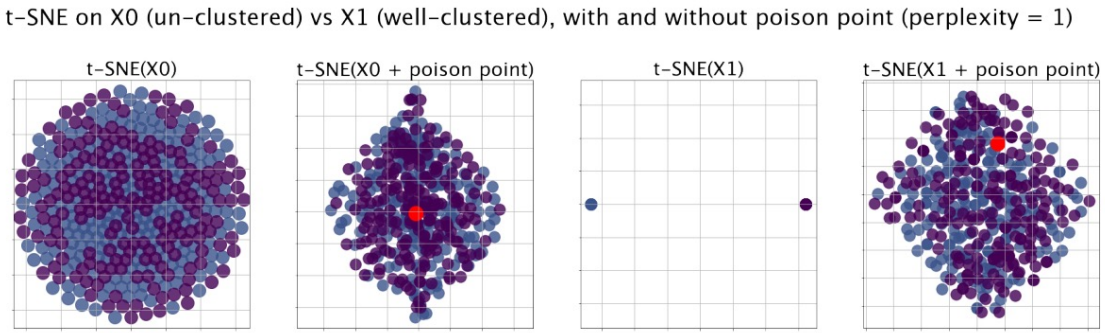


Theorem: there exists

- clustered dataset X
- un-clustered (simplex) dataset X’
- “poison point” x_0

such that (under suitable hyperparameter settings)

$$\mathrm{TSNE}(\mathrm{X} \cup x_0) = \mathrm{TSNE}(\mathrm{X}' \cup x_0).$$



Theorem: there exists
- clustered dataset X
- un-clustered (simplex) dataset X'
- "poison point" x_0

such that $\mathrm{TSNE}(\mathrm{X} \cup x_0) = \mathrm{TSNE}(\mathrm{X}' \cup x_0)$.

Proof:
- Center both datasets at the origin; take x_0 = origin
- x_0 *is now the nearest neighbor to every point*
- This dominates the information stored in P (recall it is a "soft" kNN graph)

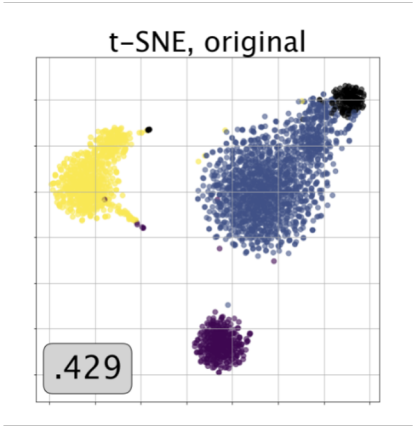
Theorem: there exists
- clustered dataset X
- un-clustered (simplex) dataset X’
- “poison point” x_0

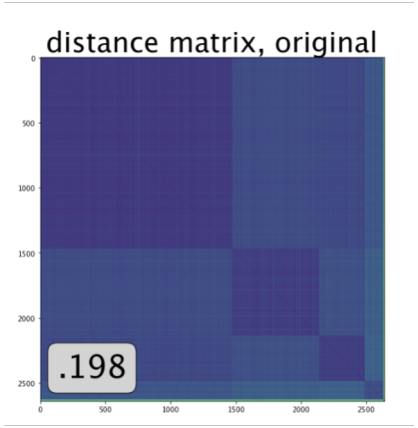
such that $\mathrm{TSNE}(\mathrm{X} \cup x_0) = \mathrm{TSNE}(\mathrm{X}' \cup x_0)$.

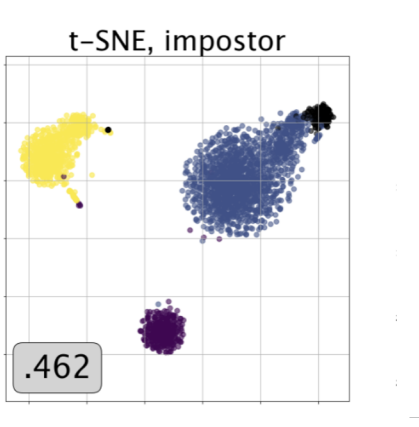
This is a “false negative” construction.
We also have a “false positive” construction.

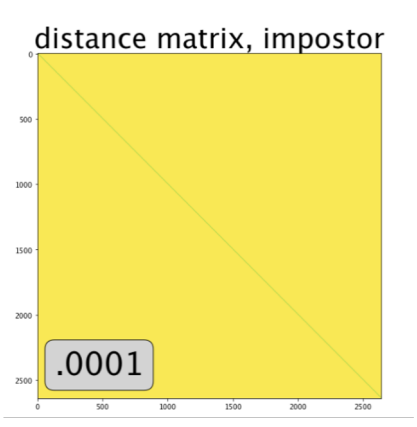
Theorem: For *any* dataset X
there exists arbitrarily near-simplex dataset X'
such that $\mathrm{TSNE}(\mathrm{X}) = \mathrm{TSNE}(\mathrm{X}')$

"the impostor"









Reason: normalization of P leads to "additive invariance"

Alternative interpretation:
t-SNE is most unstable on intrinsically high-dimensional (near-simplex) data,

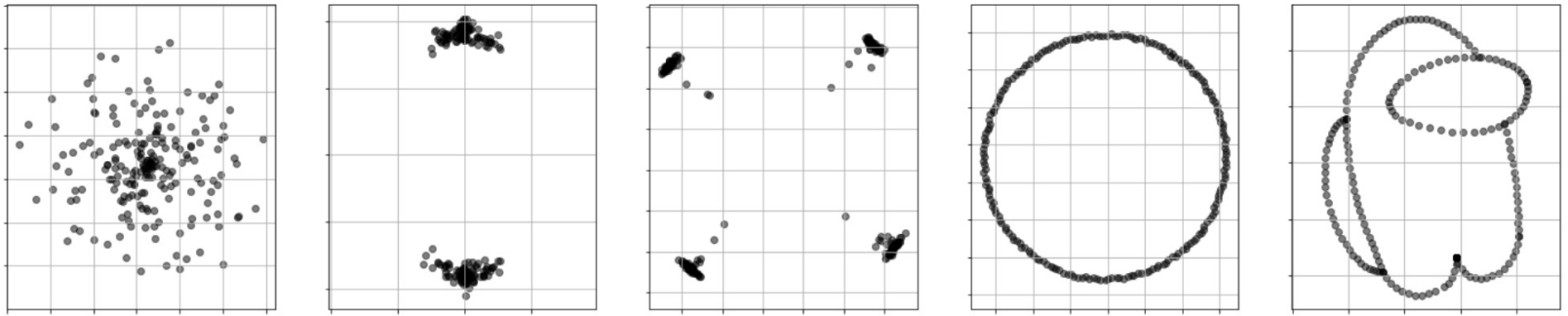


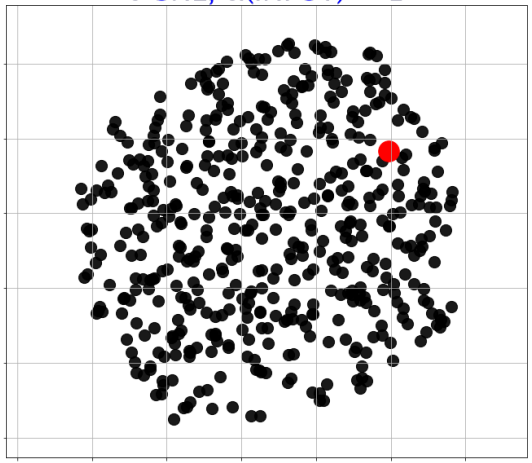
Figure 3: Various different 2D t-SNE visualizations produced by adversarial perturbations of a 200-point unit regular simplex. Each pair of perturbations satisfies the conditions of Theorem 5 for $\epsilon = 0.01$.

Finding 2:
T-SNE doesn’t just distort cluster structure.
It distorts outliers.

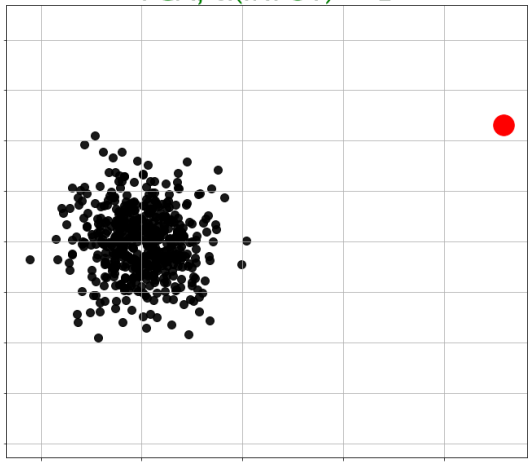
α = “how extreme is the worst outlier”

Synthetic Data: Gaussian plus outlier(s)

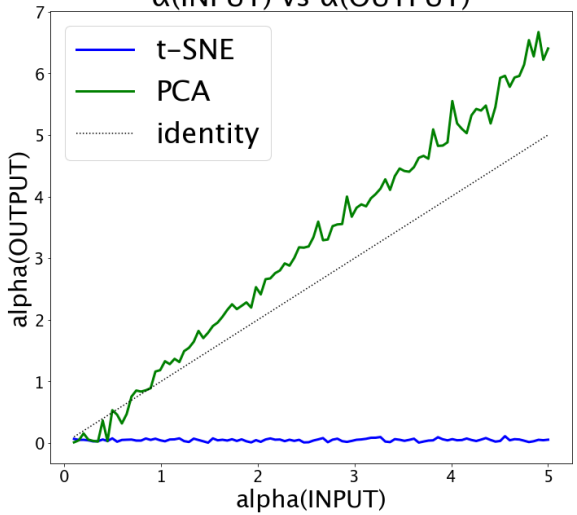
t-SNE, α(INPUT) = 1



PCA, α(INPUT) = 1



α(INPUT) vs α(OUTPUT)

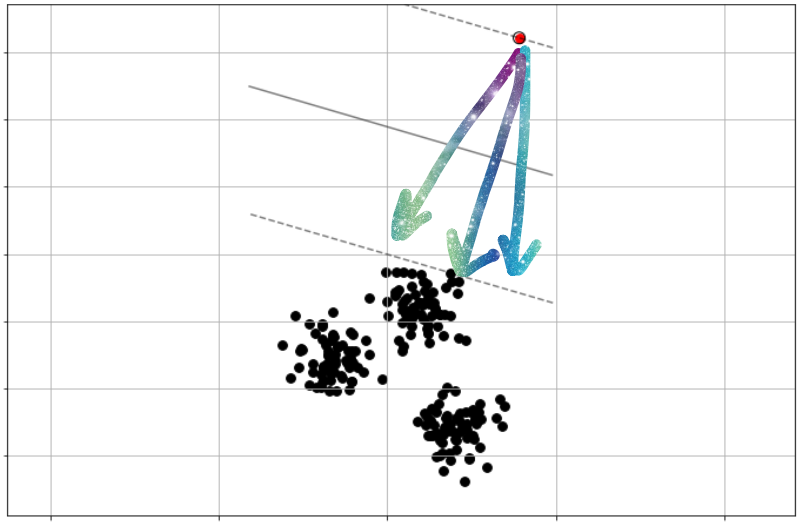


Theorem: α(any stationary t-SNE embedding) ≤ 4

Theorem: α(any stationary t-SNE embedding) ≤ 4

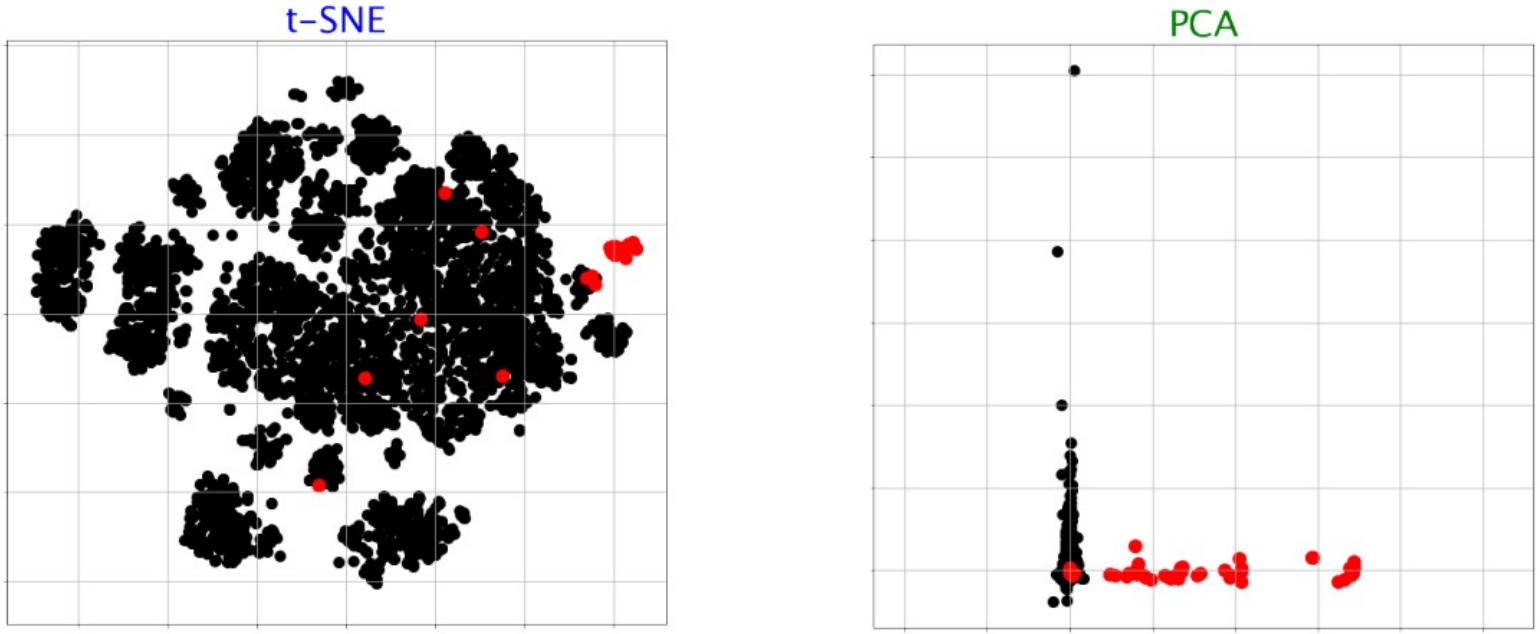
Definition of $\alpha(Z)$ →

Proof Idea:
Asymmetry of P and Q pulls the outlier in!



$$\frac{\delta C}{\delta y_i} = 4\sum_j (p_{ij} - q_{ij})(1 + \|y_i - y_j\|^2)^{-1}(y_i - y_j).$$

For $i = 0$, all cancellation happens here.



Credit card fraud dataset (red = fraud)

Discussion

The attributes which make t-SNE strong
(asymmetric kernels, soft normalization)
have provable **side effects**.

(Empirically, we see similar failure modes in UMAP + alternatives).

Theory is powerful here!
(helped us uncover the poison point attack)

Future Work

(1) General, fine-grained tradeoffs in data visualization / ultra-low-dimensional Euclidean embedding.

(2) Better understanding of loss surface (for t-SNE and friends)

(3) How to think about data visualization, beyond point clouds…

Thank you!