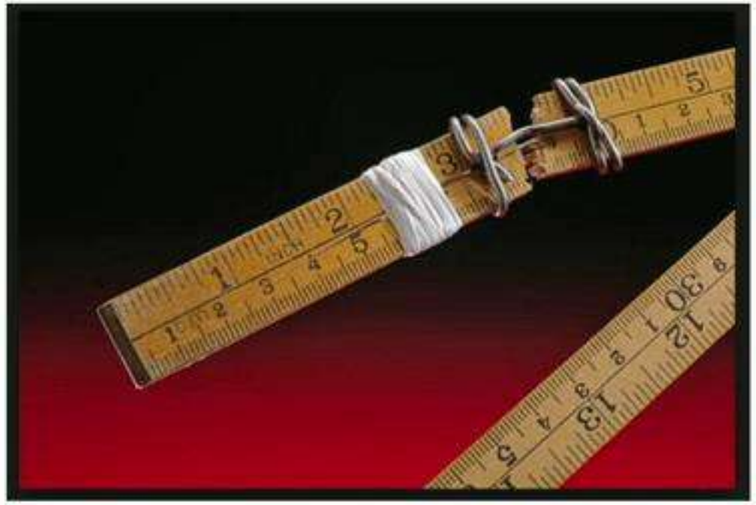
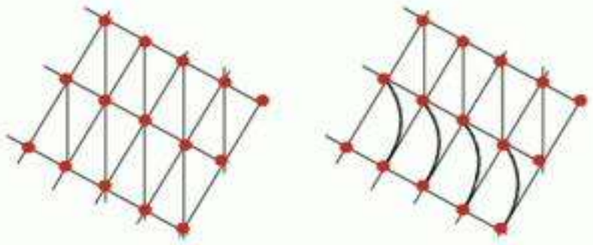
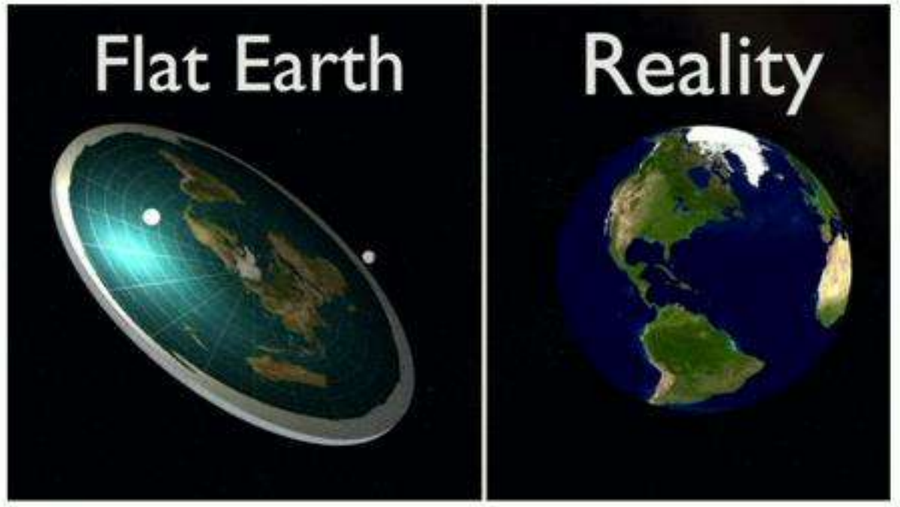
Geometry of data: Sparse metric repair



Joint work with Lalit Jain (UW) & Rishi Sonthalia (UMich)
Chenglin Fan, Benjamin Raichel, Gregory Van Buskirk
(UT-Dallas)

Distorted geometry or broken metrics





Motivation: Data cleaning

Suppose we're given (similarity) data D'

It's **supposed** to be metric

When D' is metric:

- Can run downstream unsupervised machine learning algorithms more successfully (e.g., clustering)
- Many approximation algorithms give better guarantees (e.g., Traveling Salesman, Maximum Spanning Tree, k-median)
- Applications in: computer vision (image retrieval), computational biology (amino acid substitution matrices used in alignment problems, scRNA-seq analysis)

How do we ensure (similarity) data is metric?

Definitions

Definition

A matrix $D \in \mathrm{Sym}_n(\mathbb{R}_{\geq 0})$ is said to be a *metric* if

$$D_{ii} = 0 \text{ for } 1 \leq i \leq n$$
$$D_{ij} \leq D_{ik} + D_{jk} \text{ for } 1 \leq i, j, k \leq n.$$

Rewrite as system of inequalities

$$D_{ij} \leq D_{ik} + D_{jk} \text{ for } 1 \leq i, j, k \leq n \quad \Longleftrightarrow \quad \mathcal{M} \cdot \mathrm{vec}D \succeq 0$$

Each row of $\mathcal{M}$ corresponds to triangle $t = i, j, k$

Triangle inequality matrix, structured

$$\mathcal{M}_t = [0, \cdots, \underbrace{1}_{D_{ik}}, \cdots, \underbrace{1}_{D_{jk}}, \cdots, \underbrace{-1}_{D_{ij}}, \cdots 0]$$

Metric Repair Formally

Problem Statement

Given a corrupted metric D', (i.e., some triangle inequality is broken), find a perturbation P so that $D' + P$ is metric; i.e.,

$$\mathcal{M}(D' + P) \succeq 0.$$

Here D' is the corruption of a metric D,

$$D' = D - P,$$

and P is the repair.

Traditional Techniques

Project onto the subspace of metrics (e.g., [Brickell, et al.])

$$P := \underset{\mathcal{M}(D)\succeq 0}{\operatorname{argmin}} \|D - D'\|_F$$

Or onto the cone of Euclidean Matrices (e.g., extensive psychometric literature, multi-dimensional scaling)

$$P := \underset{D\in \mathrm{EDM}^n}{\operatorname{argmin}} \|D - D'\|_F$$

Randomized algorithm for $\mathbb{R}^3$ (Berger, et al. 1999)

Rigidity theory and missing (Euclidean) distances (e.g., Cauchy 1813)

Problem: The above methods perturb most distances!

Wanted: Take a more optimistic approach. Assume most distances are correct—only a few distances are corrupted!

Sparse Metric Repair: Optimistic data cleaning

$$\begin{aligned} \text{minimize} \quad & \|P\|_0 \\ \text{subject to} \quad & \mathcal{M}(D' + P) \succeq 0 \end{aligned}$$

ℓ_1 methods: convex relaxation

$$\begin{aligned} \text{minimize} \quad & \|P\|_1 \\ \text{subject to} \quad & \mathcal{M}(D' + P) \succeq 0 \end{aligned}$$

Naive ℓ_1 method does not give the sparsest solutions, in general.

($\mathcal{M}$ is overdetermined and not particularly incoherent.)

Sparse Metric Repair: Optimistic data cleaning

$$\begin{aligned} \text{minimize} \quad & \|P\|_0 \\ \text{subject to} \quad & \mathcal{M}(D' + P) \succeq 0 \end{aligned}$$

ℓ_1 methods: convex relaxation

$$\begin{aligned} \text{minimize} \quad & \|P\|_1 \\ \text{subject to} \quad & \mathcal{M}(D' + P) \succeq 0 \end{aligned}$$

Naive ℓ_1 method does not give the sparsest solutions, in general.

($\mathcal{M}$ is overdetermined and not particularly incoherent.)

The real problem with convex methods....

Potentially n^3 constraints coming from each triangle inequality!

Solving large LPs is computationally expensive

Want to have rigorous guarantees and exploit the combinatorial nature of the problem. Maybe even one-pass algorithms.

A seemingly easier question: Can we **efficiently** detect whether whether D' is even metric? Can we find the broken triangles quickly?

No (probably not): Subcubic Barriers

Theorem (Williams and Williams, 2010)

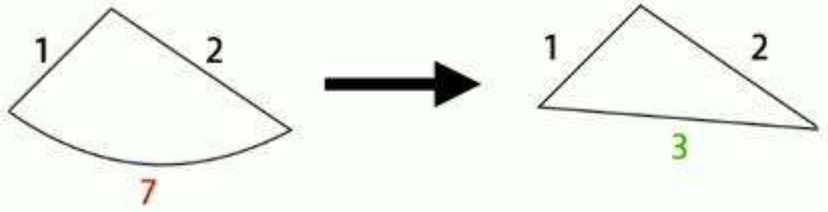
The following weighted problems either all have truly subcubic ($O(n^{3-\epsilon})$) algorithms, or none of them do:

- *The all-pairs shortest paths problem (APSP).*
- *Checking whether a given matrix defines a metric.*
- *Verifying the correctness of a matrix product over the* $(\min, +)$*-semiring.*

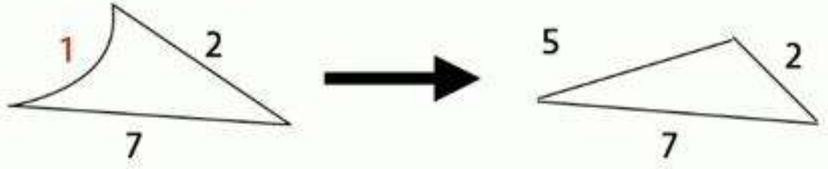
So we probably can't do better then $O(n^3)$.

Three Repair Scenarios: Constrain P

Decrease Only Metric Repair (DOMR): $P \preceq 0$: i.e. D' was perturbed by positive perturbations.

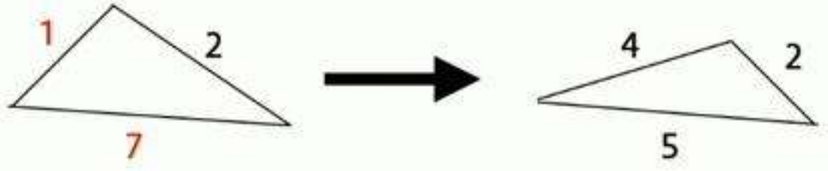


Increase Only Metric Repair (IOMR): $0 \preceq P$: i.e. D' was perturbed by negative perturbations.



Note: a sparse DOMR solution may not give a sparse IOMR solution and vice versa—consider these cases separately!

General Metric Repair: We do not place any restrictions on P.



Decrease Only Metric Repair: Floyd Warshall Algorithm

Input: Corrupted $n \times n$ distance matrix D'
Result: Perturbation P
$\widehat{D} = D'$
for $k = 1$ **to** n **do**
  for $i = 1$ **to** n **do**
    for $j = 1$ **to** $i - 1$ **do**
      if $\widehat{D}_{ij} \geq \widehat{D}_{ik} + \widehat{D}_{kj}$ **then**
        $\widehat{D}_{ij} = \widehat{D}_{ik} + \widehat{D}_{kj}$
      end
    end
  end
end
$P = \widehat{D} - D'$

Algorithm 1: Floyd-Warshall for DOMR

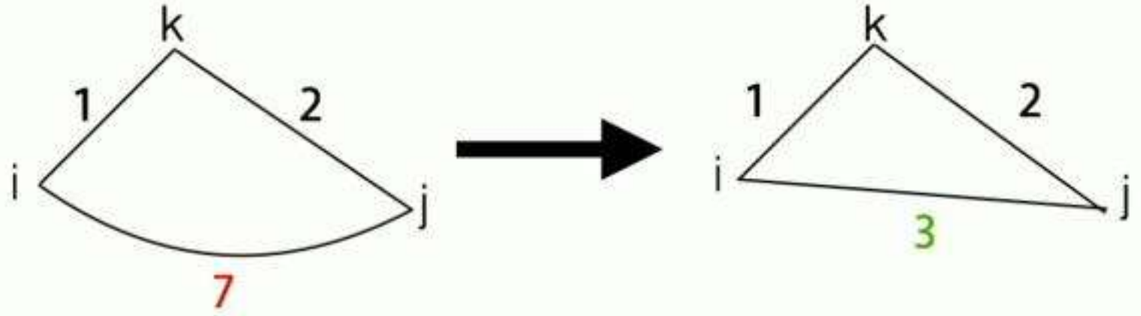
$\widehat{D}$ is the APSP solution for K_n with weights given by D'.

Decrease Only Metric Repair: Floyd Warshall Algorithm

To fix a broken triangle *ijk*,

$$D_{ij} = D'_{ik} + D'_{jk}$$

Interpretation using paths: If we can find a shorter path from *i* to *j* through *k* (rather then taking the edge *ij*), we should take it.



Too big!!!!

Just Right!!!!

All Pairs Shortest Path is the Same as DOMR.

Decrease Only Metric Repair: Sparse Guarantees

Lemma (Brickell, et al.)

Let $\widehat{D}$ be any solution to DOMR; Let D^A be the Floyd-Warshall solution.
Then, $\widehat{D}$ is element-wise smaller than D^A, $\hat{D} \preceq D^A$.

Lemma (Jain, Gilbert)

The perturbation $P^A := D^A - D$ is a sparsest possible decrease only metric repair solution.
In addition, D^A is the solution to

$$argmin_{D \preceq D'} \|D - D'\|_p$$

Increase Only Metric Repair: Algorithms

Two main obstructions:

1. Unlike the DOMR case, it is not possible to detect which distances were perturbed from a broken triangle inequality.

   Indeed, if $D_{ij} \geq D_{ik} + D_{jk}$, then either D_{ik} or D_{jk} could have been initially perturbed.

2. Many possible ℓ_1 solutions, but few sparse ones!

   There are approximation algorithms, though!

Decrease Only Metric Repair: Sparse Guarantees

Lemma (Brickell, et al.)

Let $\widehat{D}$ be any solution to DOMR; Let D^A be the Floyd-Warshall solution.
Then, $\widehat{D}$ is element-wise smaller than D^A, $\hat{D} \preceq D^A$.

Lemma (Jain, Gilbert)

The perturbation $P^A := D^A - D$ is a sparsest possible decrease only metric repair solution.
In addition, D^A is the solution to

$$argmin_{D \preceq D'} \|D - D'\|_p$$

Increase Only Metric Repair: Algorithms

Two main obstructions:

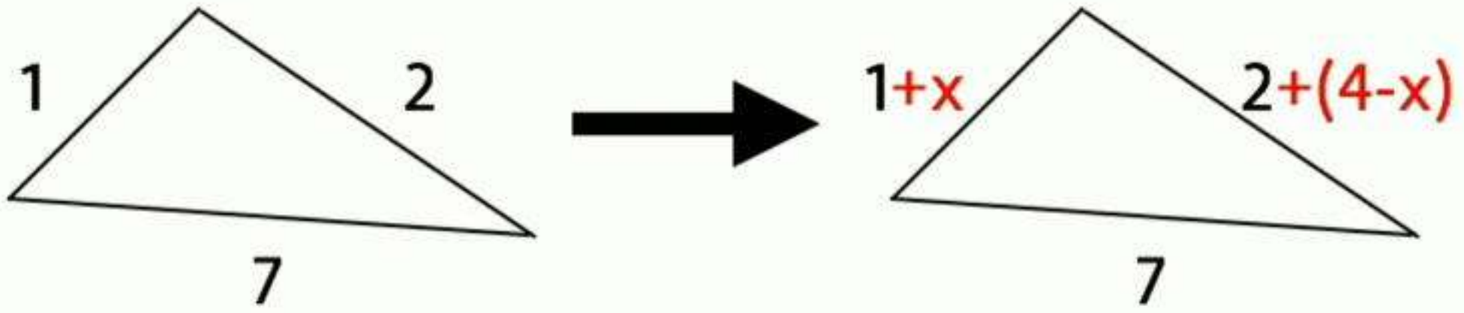
1. Unlike the DOMR case, it is not possible to detect which distances were perturbed from a broken triangle inequality.

   Indeed, if $D_{ij} \geq D_{ik} + D_{jk}$, then either D_{ik} or D_{jk} could have been initially perturbed.

2. Many possible ℓ_1 solutions, but few sparse ones!

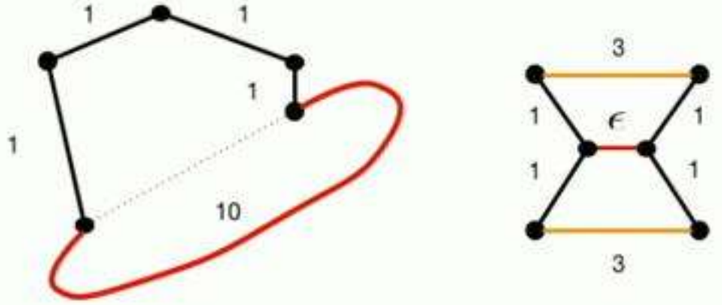
   There are approximation algorithms, though!

An example: many solutions with total ℓ_1 norm = 4

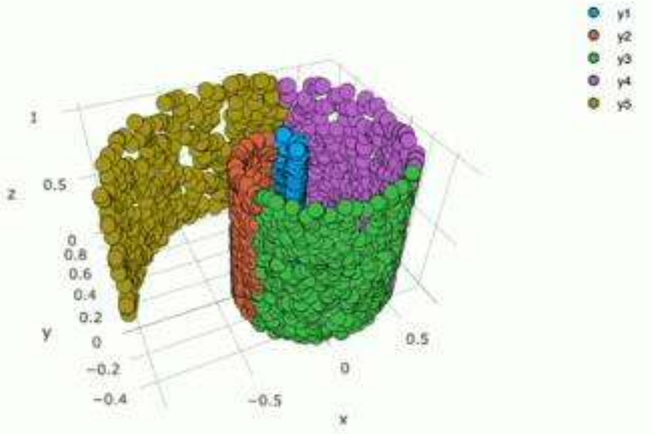


Extensions and generalizations

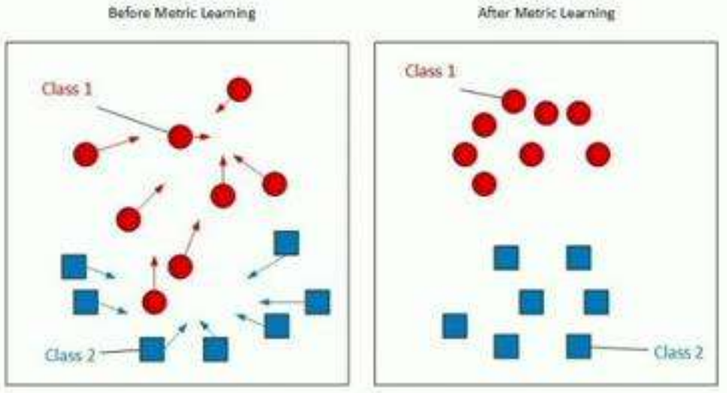
Graph metric repair



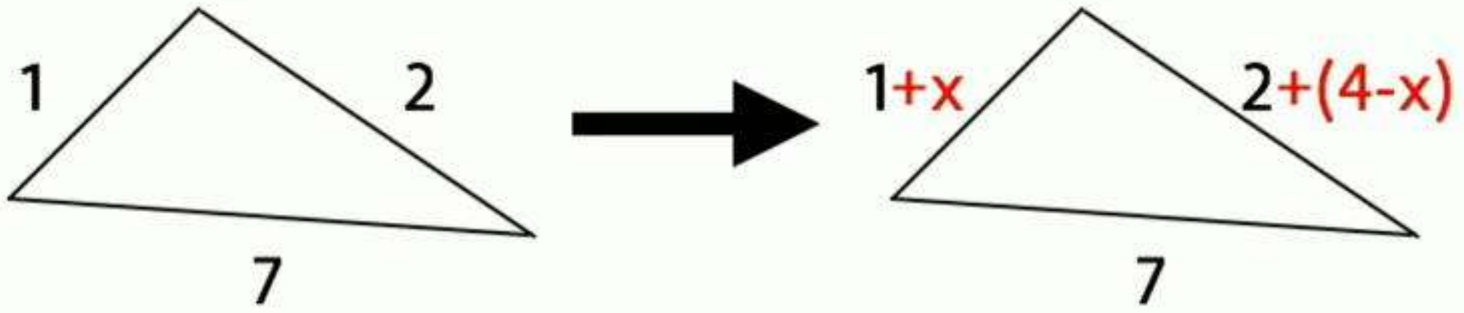
Manifold repair for embeddings



Metric learning

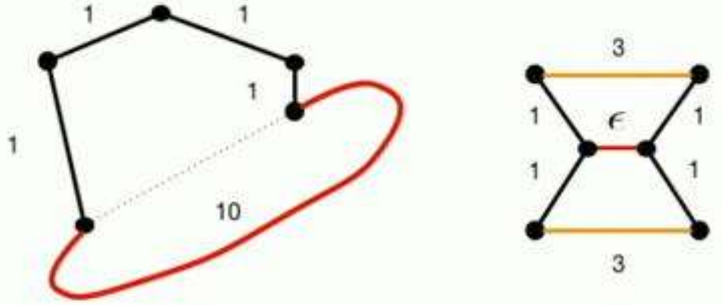


An example: many solutions with total ℓ_1 norm = 4

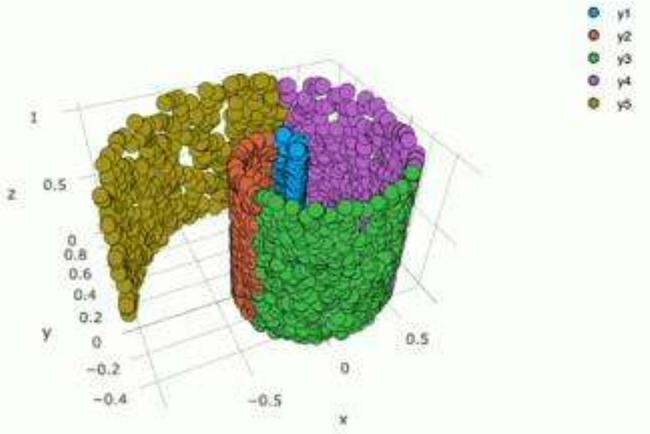


Extensions and generalizations

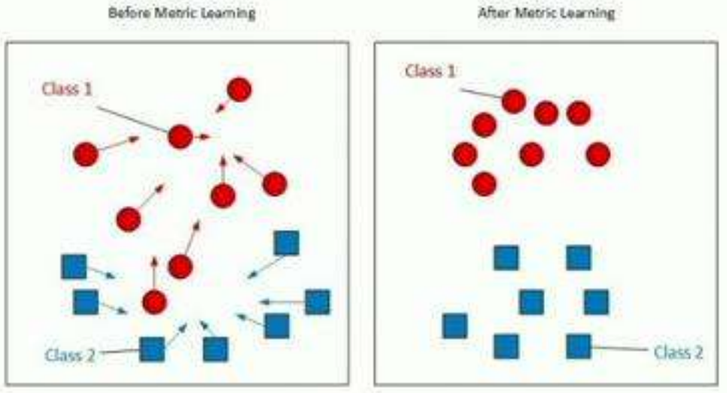
Graph metric repair



Manifold repair for embeddings



Metric learning



Graph metric repair

Given a weighted undirected graph $G = (V, E, w)$ and a set $\Omega \subseteq \mathbb{R}$, find the smallest set of edges $S \subseteq E$ such that by modifying the weight of each edge in S, by adding a value from Ω, the new distances satisfy a metric.

Related problems: metric nearness, metric embedding with outliers, matrix completion, graph cutting problems

Graph metric repair: results summary

Decrease only ($\Omega = \mathbb{R}_{\leq 0}$) solvable in cubic time, if distances are allowed to increase even by a single number, the problem is NP-Complete.

For increase only and general cases:

Polynomial-time approximation-preserving reductions from MULTICUT and LB-CUT to graph metric repair.

There are several approximation algorithms, parameterized by different measures of how far the input is from a metric.

For any fixed constant ς, by parameterizing on the size of the optimal solution, we present a *fixed parameter tractable* algorithm for the case when G is ς-chordal.

FPT results: Structural theorem

Key idea in characterizing the support of all solutions and in proving a verifier for a solution: If the shortest path between two adjacent vertices is the not the edge connecting them, then this edge is the heavy edge of a broken cycle.

Theorem

For any positively weighted graph $G = (V, E, w)$ and $S \subset E$, S is a set of edges that contains at least one edge from each broken cycle (regular cover) if and only if S is the support to a solution to $MR(G, \mathbb{R})$.

FPT results: Verifier

Theorem

The VERIFIER *algorithm, given a weighted graph G and a potential support for a solution S, determines in $O(n^3)$ time whether there exists a valid (increase only or general) solution on that support and if so finds one.*

Input: $G = (V, E, w), S$
$M = \|w\|_\infty$, $\hat{G} = (V, E, \hat{w})$
for $e \in S$ **do**
  set $\hat{w}(e) = M$
end
for $e \in E \setminus S$ **do**
  set $\hat{w}(e) = w(e)$
end
for $(u, v) \in E$ **do**
  update $w(u, v)$ to be length of the shortest path from u to v in $\hat{G}$
end
if *only edges in S had weights changed (or increased for increase only case)* **then**
  return w **else**
    return NULL
  end
end

Algorithm 2: Verifier

FPT results: Structural theorem

Key idea in characterizing the support of all solutions and in proving a verifier for a solution: If the shortest path between two adjacent vertices is the not the edge connecting them, then this edge is the heavy edge of a broken cycle.

Theorem

For any positively weighted graph $G = (V, E, w)$ and $S \subset E$, S is a set of edges that contains at least one edge from each broken cycle (regular cover) if and only if S is the support to a solution to $MR(G, \mathbb{R})$.

FPT results: Verifier

Theorem

The VERIFIER *algorithm, given a weighted graph G and a potential support for a solution S, determines in $O(n^3)$ time whether there exists a valid (increase only or general) solution on that support and if so finds one.*

Input: $G = (V, E, w), S$
$M = \|w\|_\infty$, $\hat{G} = (V, E, \hat{w})$
for $e \in S$ **do**
  set $\hat{w}(e) = M$
end
for $e \in E \setminus S$ **do**
  set $\hat{w}(e) = w(e)$
end
for $(u, v) \in E$ **do**
  update $w(u, v)$ to be length of the shortest path from u to v in $\hat{G}$
end
if *only edges in S had weights changed (or increased for increase only case)* **then**
  return w **else**
    return NULL
  end
end

Algorithm 2: Verifier

FPT results: sketch

By structural theorem, seek a minimum sized cover of all broken cycles.

If G has a broken cycle, then it has a broken chordless cycle.

Natural but wrong approach: find an uncovered broken chordless cycle and recursively try adding each one of its edges to our current solution.

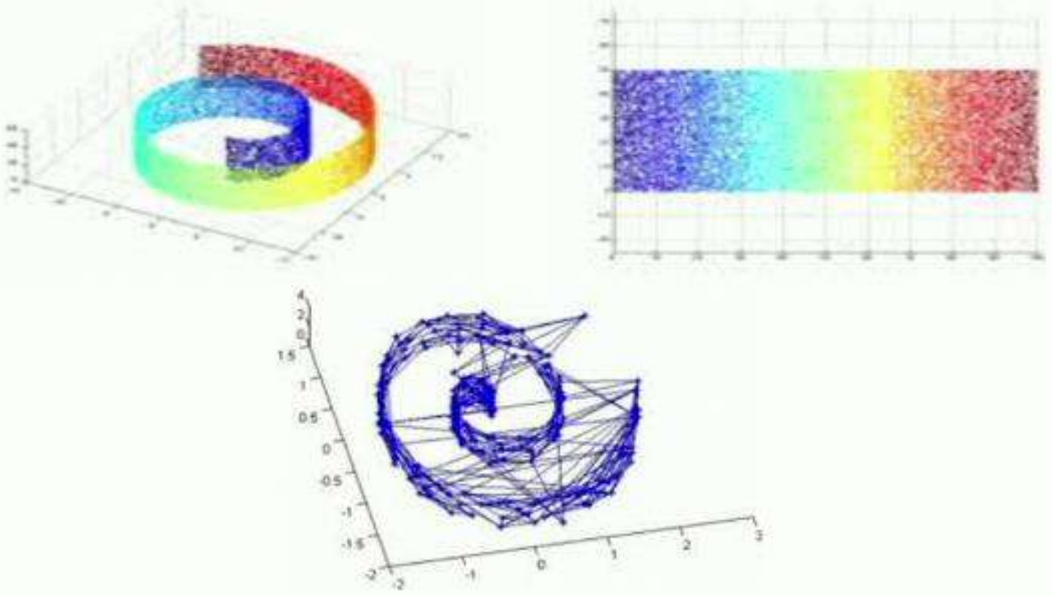
Correct approach: Consider an optimal solution W, with support S_W. Suppose we have found a subset $S \subsetneq S_W$, covering all broken chordless cycles in G. If we add to each edge in S its weight from W, then any remaining broken chordless cycle must be covered further, in effect revealing which edges to consider from the chorded cycles from the original graph G. Despite not knowing W, we can still identify a bounded-sized subset of edges containing an edge from a cycle needing to be covered further.

Theorem

For any fixed constant ς, FPTRECURSE *is an FPT algorithm for $MR(G, \mathbb{R})$ for any $G \in F_\varsigma$, when parameterized by opt. The running time is $\Theta((2\varsigma \mathrm{opt}^\varsigma)^{\mathrm{opt}+1} n^\varsigma)$.*

Manifold repair

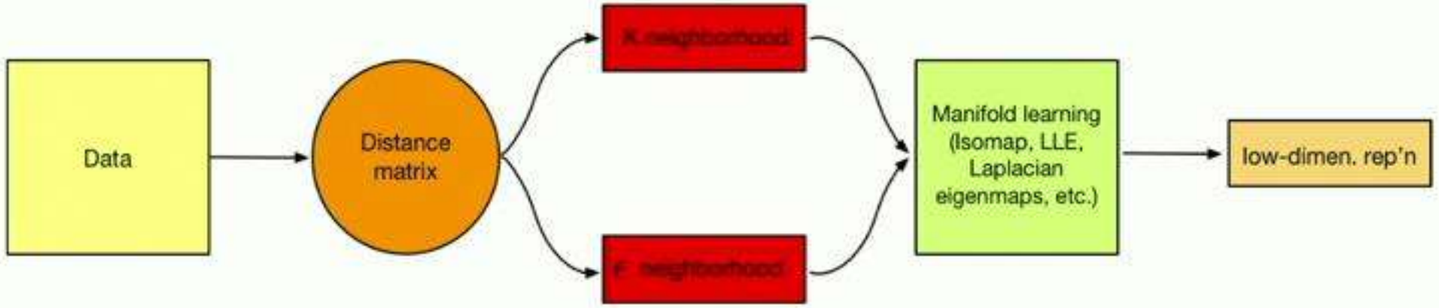
(Typical model of) real world data lies on low dimensional manifolds embedded in a high dimensional space



Dimension reduction algorithms: ISOMAP, Laplacian EigenMaps, LLE, etc.

Dimensionality Reduction Algorithms

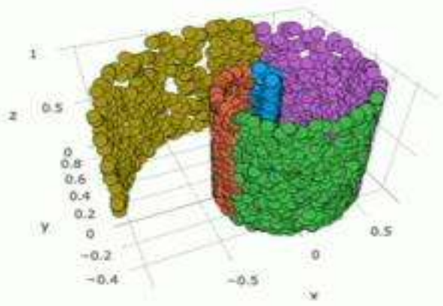
General structure of dimension reduction algorithms



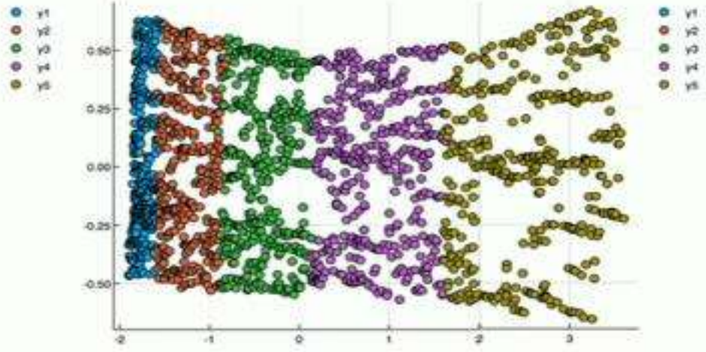
Problem

Actual data sets have many missing entries in the data. We cannot (accurately) compute the distance or dissimilarity matrix. How can we apply standard dimension reduction algorithms to incorrect distances?

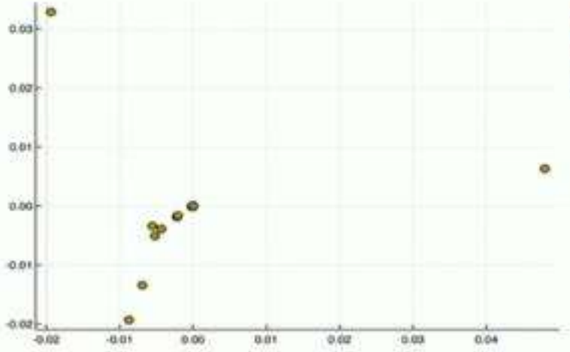
Corrupted data



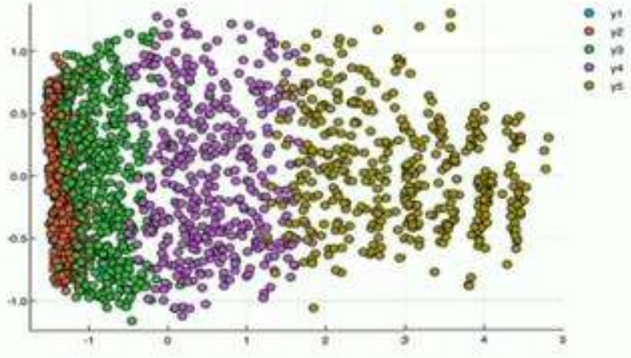
(a) Original swissroll data set



(b) true distance matrix



(c) corrupted distance matrix

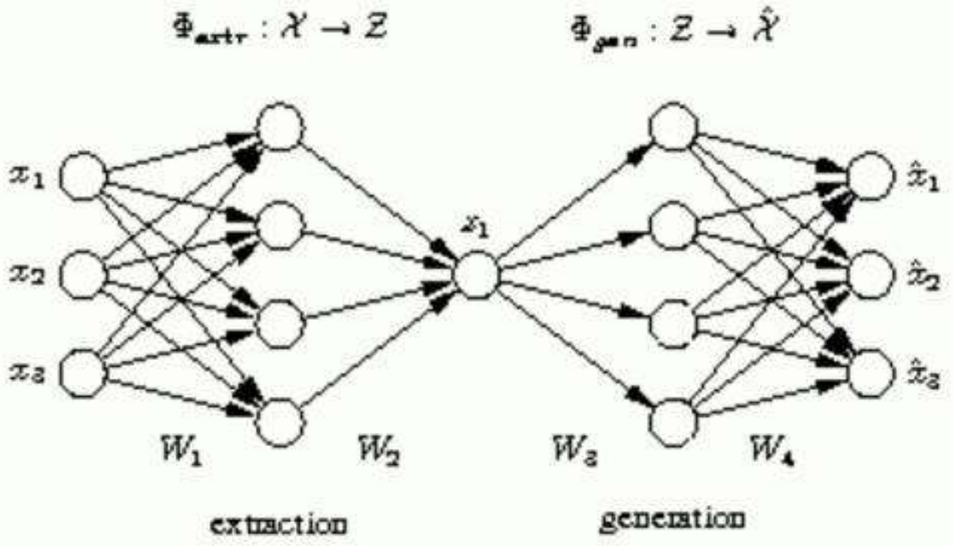


(d) repaired distance matrix

Figure: (a) The original swissroll data set (2000 points) and the results from ISOMAP for: (b) the original distance matrix, (c) the corrupted distance matrix, and (d) the repaired distance matrix.

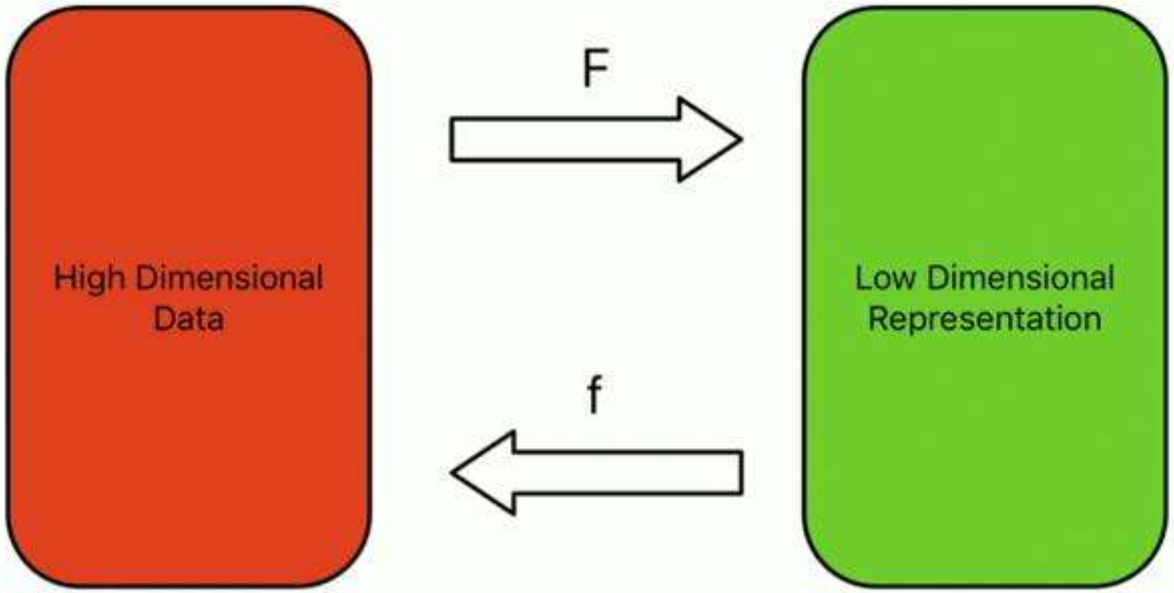
Previous work: nlPCA

Non Linear Principal Component Analysis

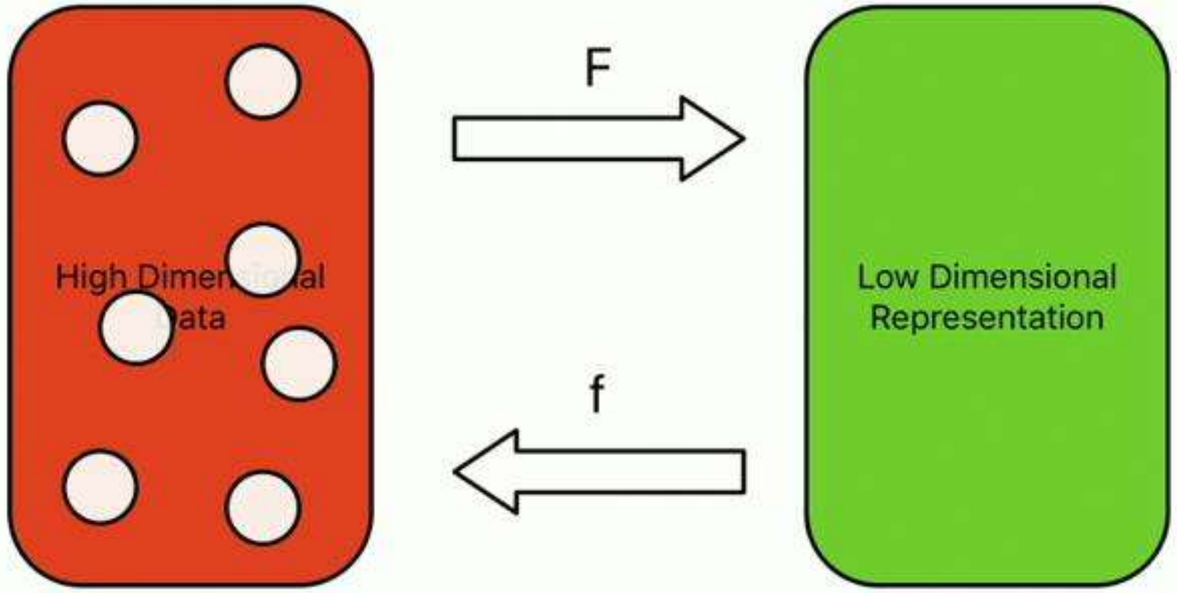


Auto-associative neural network (Autoencoder)

Previous work: mDRUR



Previous work: mDRUR



Problem statement

Given a data matrix X and a 0/1 matrix Q such that $Q_{ij} = 0$ iff X_{ij} is missing, repair sparsely the corrupted dissimilarity matrix D so that standard dimensionality reduction algorithms produce "good" low dimensional embeddings.

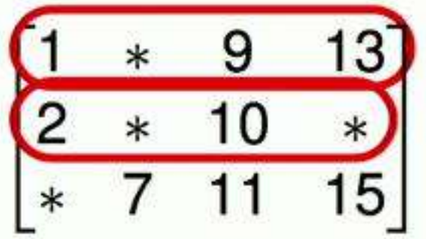
MR-Missing

Estimate the distance

Project 2 data point to the coordinates present in both. Then calculate the (ℓ_1) distance
This contracts the distances

Repair the estimates so that they adhere to a metric.

We repair the metric by only increasing the distances

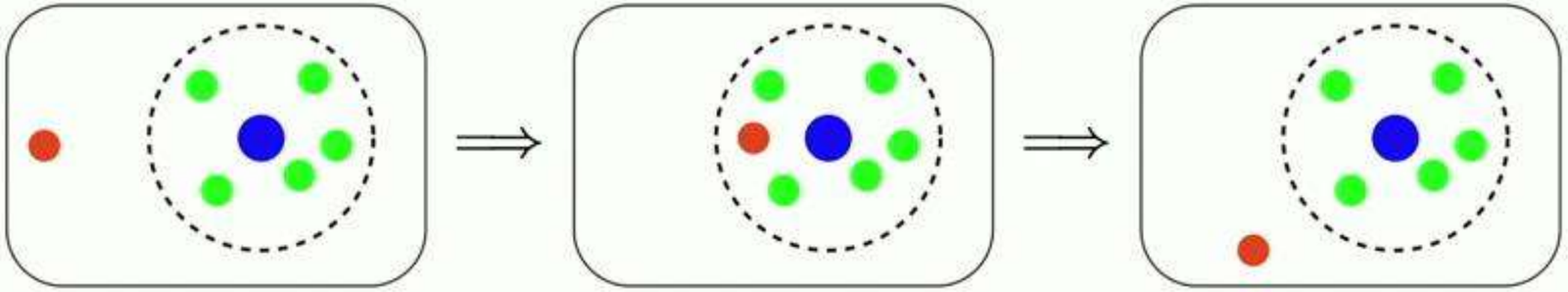


$$\begin{bmatrix} 0 & 2 & 4 \\ 2 & 0 & 1 \\ 4 & 1 & 0 \end{bmatrix} \qquad \begin{bmatrix} 0 & 3 & 4 \\ 3 & 0 & 1 \\ 4 & 1 & 0 \end{bmatrix}$$

Note: $4 > 2 + 1$

Visualizing the Algorithm

Potential movement of points



Desiderata:

In distance estimation step, the red circle enter the neighborhood of the blue circle "rarely"

If the red circle enters the neighborhood of the blue circle, then the repair step pushes it back out "often"

Results: distance estimation step

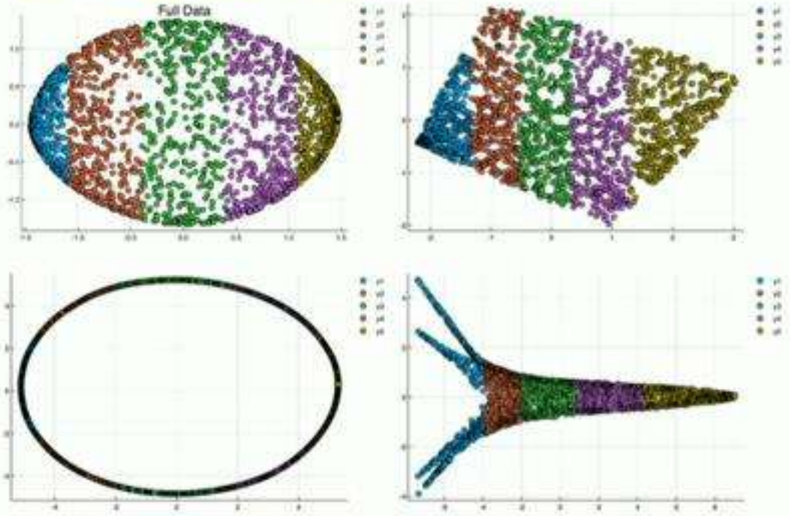
Theorem

Suppose $X \sim \mathcal{N}(\mu_1 \mathbf{1}, 0.5I)$ *and* $Y \sim \mathcal{N}(\mu_2 \mathbf{1}, 0.5I)$ *are two points in* $\mathbb{R}^n$ *such that each coordinate of* X, Y *is present with probability* p. *If* $\mu = \mu_1 - \mu_2$ *and* $q = p^2$, *then, for all* $q(1+\mu^2) > \epsilon > 0$ *and* $\frac{q(1+\mu^2)-\epsilon}{(1+\mu^2)} > \gamma > 0$, *we have that*

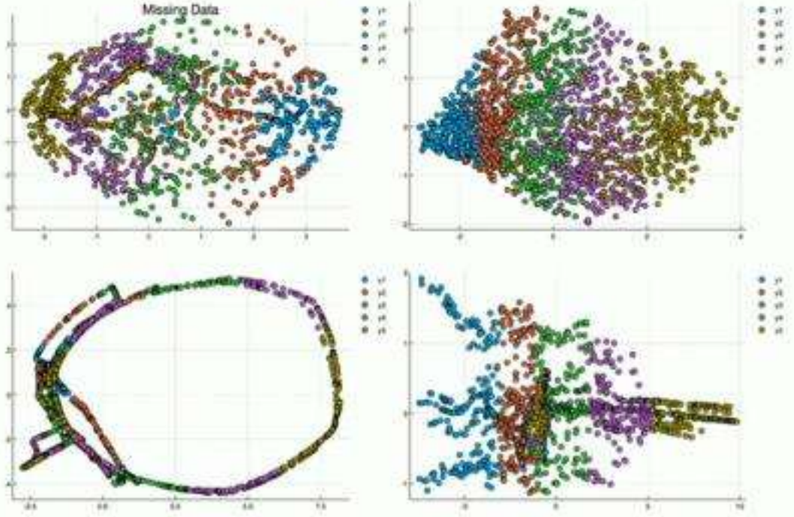
$$\Pr[d_p(X, Y) < \epsilon n] \leq e^{-2\gamma^2 n} + \left(e^{-\frac{((q-\gamma)(1+\mu^2)-\epsilon)^2}{4(q-\gamma)(1+2\mu^2)}} \right)^n.$$

Example (visual) verification: Synthetic manifolds

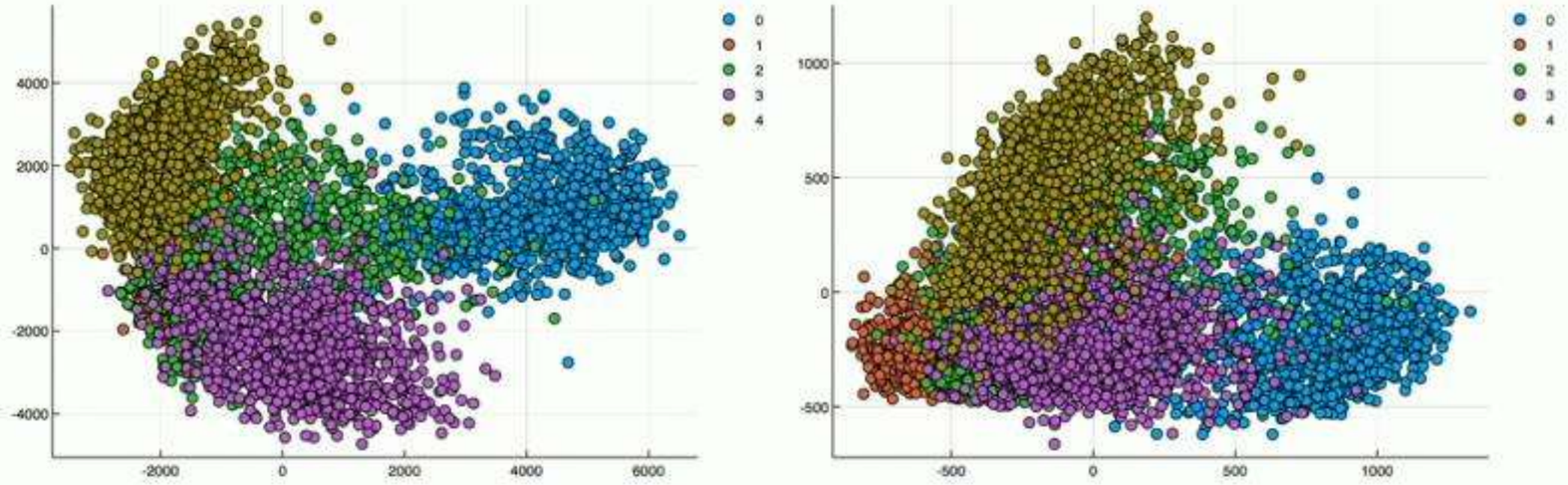
Complete Data



With 40% missing using MR-Missing



Example (visual) verification: Projection of MNIST



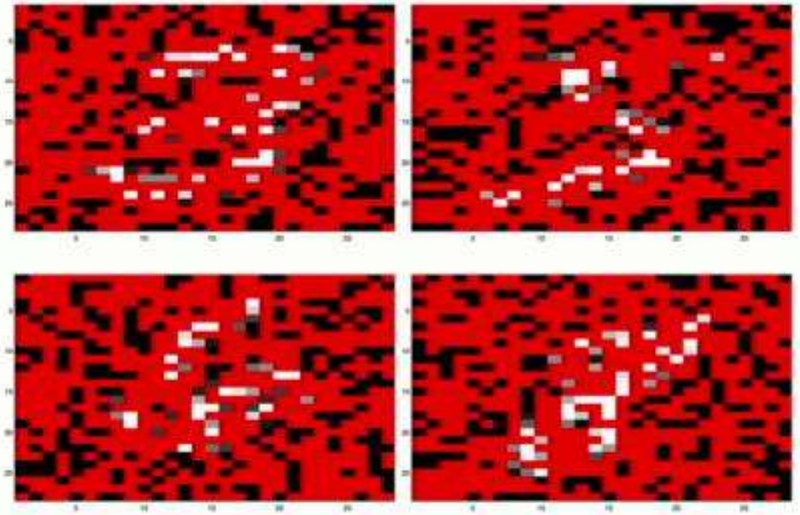
(a) ISOMAP with actual distances (b) ISOMAP with repaired distances

Figure: Two-dimensional projections of the first 1000 images of the digits 0,1,2,3,4 from MNIST using ISOMAP with true distance and ISOMAP with distance obtained from MR-MISSING when 70% of the data is missing.

Example (classification) verification: MNIST

% missing	0	40	50	60	70	80	90
Accuracy	0.94	0.91	0.90	0.86	0.77	0.20	0.10

Table: Accuracy of an SVM trained on the low dimensional projections produced by MR-Missing



Example (classification) verification: MNIST

% missing	0	40	50	60	70	80	90
Accuracy	0.94	0.91	0.90	0.86	0.77	0.20	0.10

Table: Accuracy of an SVM trained on the low dimensional projections produced by MR-Missing