Trustworthy Supervised Dimensionality Reduction

Subhrajyoty Roy

Midterm Dissertation presentation for partial satisfaction of Master of Statistics Programme

Supervisor: Dr. Smarajit Bose, ISRU, ISI Kolkata

March 26, 2021

Contents

Introduction

What is Dimensionality Reduction?

- n datapoints $\boldsymbol{x}_1, \ldots \boldsymbol{x}_n$, each $\boldsymbol{x}_i \in \mathbb{R}^p$.
- Assumption: These datapoints lie approximately on a d-dimensional manifold in $\mathbb{R}^p$, where $d << p$.
- The Goal is to find $\boldsymbol{y}_1, \ldots \boldsymbol{y}_n \in \mathbb{R}^d$ such that they are “representative" of the corresponding high-dimensional points.

What is Dimensionality Reduction?

- n datapoints $\boldsymbol{x}_1, \ldots \boldsymbol{x}_n$, each $\boldsymbol{x}_i \in \mathbb{R}^p$.
- Assumption: These datapoints lie approximately on a d-dimensional manifold in $\mathbb{R}^p$, where $d << p$.
- The Goal is to find $\boldsymbol{y}_1, \ldots \boldsymbol{y}_n \in \mathbb{R}^d$ such that they are “representative" of the corresponding high-dimensional points.

We are going to formally define what is meant by “representative".

Review of Existing Dimensionality Reduction Algorithms

Classification of Existing DR Algorithms

Euclidean Metric Preserving Algorithms

- PCA
- Kernel PCA
- tSNE
- MDS
- Autoencoder

Classification of Existing DR Algorithms

Euclidean Metric Preserving Algorithms	Graph based Algorithms

Euclidean Metric Preserving Algorithms

- PCA
- Kernel PCA
- tSNE
- MDS
- Autoencoder

Graph based Algorithms

- Isomap
- LLE
- Hessian LLE
- Laplacian Eigenmaps
- UMAP

Classification of Existing DR Algorithms

Euclidean Metric Preserving Algorithms

- PCA
- Kernel PCA
- tSNE
- MDS
- Autoencoder

Graph based Algorithms

- Isomap
- LLE
- Hessian LLE
- Laplacian Eigenmaps
- UMAP

Supervised DR Algorithms

- SIR
- SAVE
- MAVE
- GSIR
- GSAVE

Problems with PCA

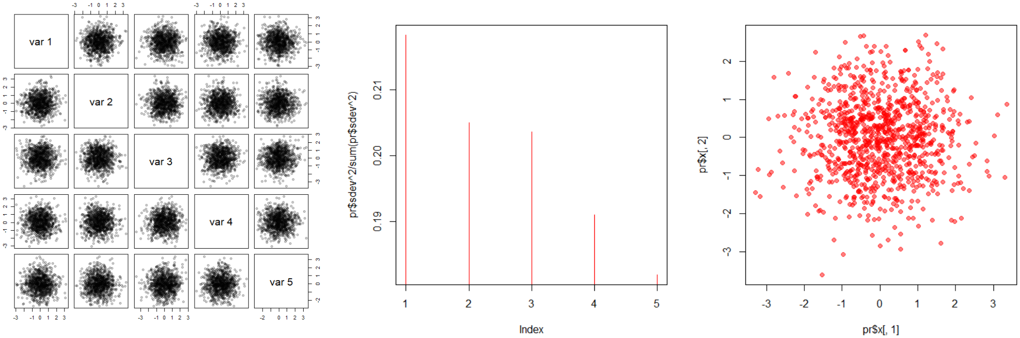


Figure: First two principal components as extracted from 5D normally distributed noise data

Problems with PCA

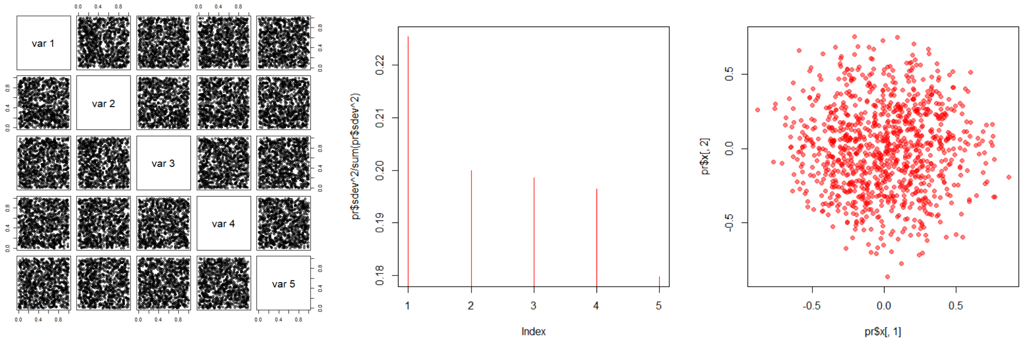


Figure: First two principal components as extracted from 5D uniformly distributed noise data

Problems with Neighbourhood Graph

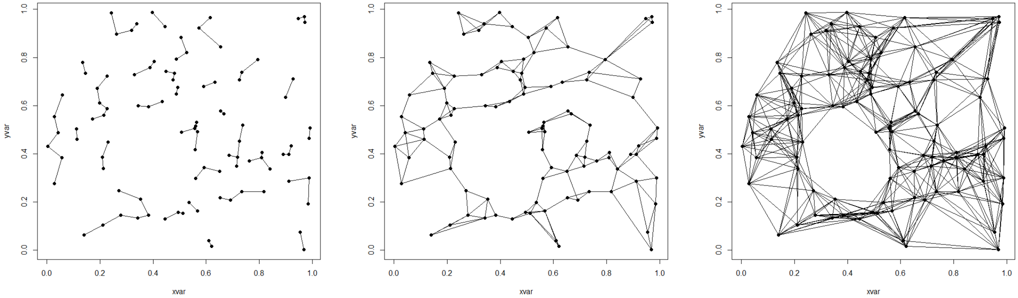
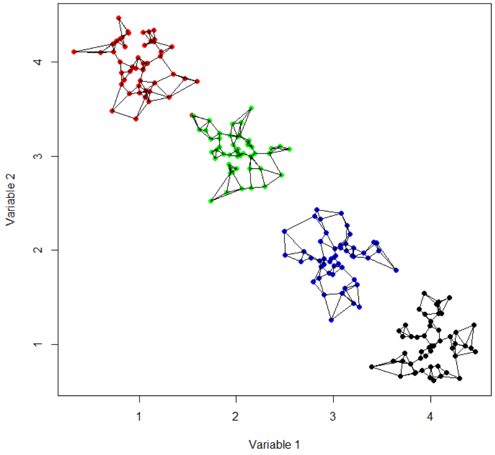
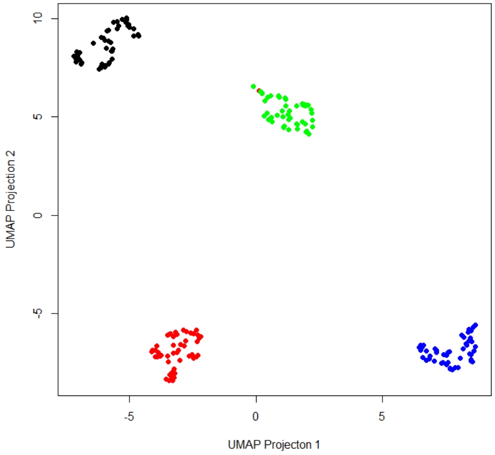


Figure: Neighbourhood graphs with $k = 1, 3, 10$, for 100 points generated uniformly in the unit square

Problem with UMAP



(a) The Neighborhood graph



(b) 2 dimensional projection by UMAP

Supervised DR Algorithms

Assumption: Response (Z) $\perp\!\!\!\perp \boldsymbol{X} \mid \sigma\left(\{\phi_1(\boldsymbol{X}), \ldots \phi_d(\boldsymbol{X})\}\right)$

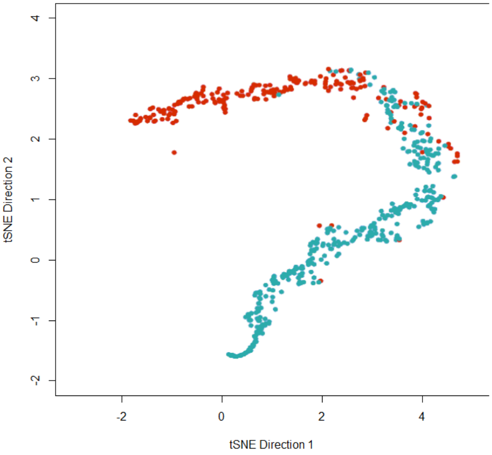
Sliced Inverse Regression (SIR)

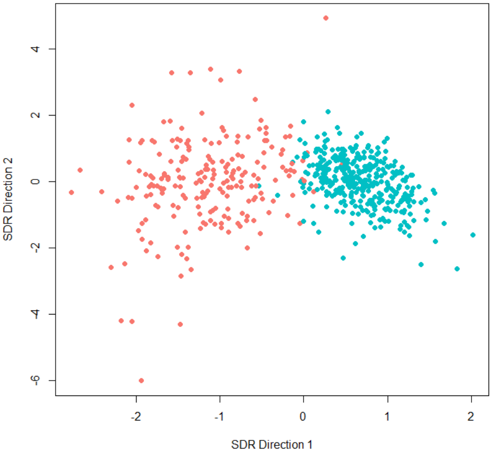
$\mathbb{E}(X \mid Z = z_1) - \mathbb{E}(X \mid Z = z_2)$ contains only the effect due the principal SDR directions.

Sliced Average Variance Estimate (SAVE)

$\mathrm{Var}(X \mid Z = z)$ contains only the variation due to the directions orthogonal to SDR.

Problem with Supervised DR (Wisconsin Breast





(a) Dimensionality reduction via tSNE (b) Dimensionality reduction via SIR

Questions to answer

1. Synthesis of supervised information into DR?

Questions to answer

1. Synthesis of supervised information into DR?
2. Frameworks and different metrics, can we combine them into one?
3. Can we measure trustability issues?

Questions to answer

1. Synthesis of supervised information into DR?
2. Frameworks and different metrics, can we combine them into one?
3. Can we measure trustability issues?
4. Can we have an improved DR algorithm? Where should we look at?

Formal View of Dimensionality Reduction

What is a reduction?

1. Measurable embedding functions $\phi_1, \ldots \phi_d : \mathbb{R}^p \to \mathbb{R}$.
2. Measurable reconstruction function $f : \mathbb{R}^d \to \mathbb{R}^p$.

What is a reduction?

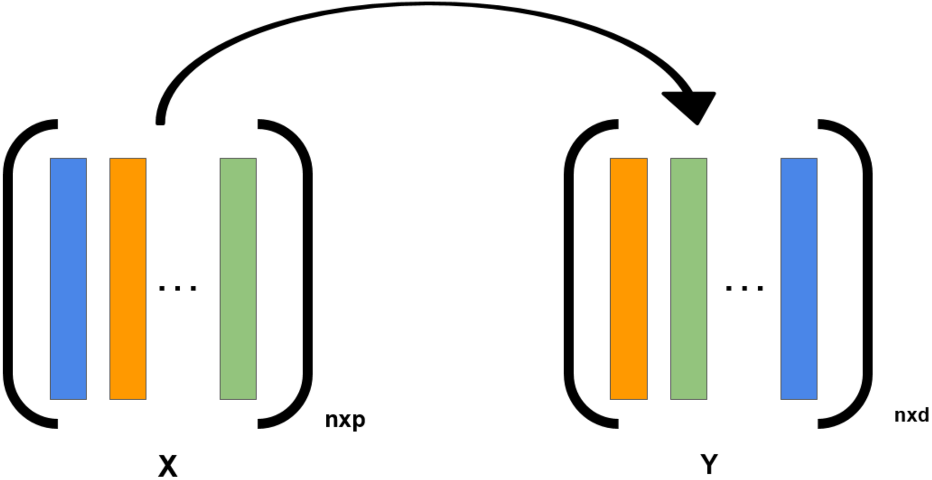
1. Measurable embedding functions $\phi_1, \ldots \phi_d : \mathbb{R}^p \to \mathbb{R}$.
2. Measurable reconstruction function $f : \mathbb{R}^d \to \mathbb{R}^p$.
3. $\mathrm{x} = f\left(\phi_1(\mathrm{x}), \ldots \phi_d(\mathrm{x})\right) + \epsilon(\mathrm{x})$ almost surely.

What is a reduction?

1. Measurable embedding functions $\phi_1, \dots \phi_d : \mathbb{R}^p \to \mathbb{R}$.
2. Measurable reconstruction function $f : \mathbb{R}^d \to \mathbb{R}^p$.
3. $\mathrm{x} = f\left(\phi_1(\mathrm{x}), \dots \phi_d(\mathrm{x})\right) + \epsilon(\mathrm{x})$ almost surely.
4. Error $\epsilon(\mathrm{x}) \perp\!\!\!\perp \{\phi_1(\mathrm{x}), \dots \phi_d(\mathrm{x})\}$.
5. $\mathbb{E}(\epsilon(\mathrm{x})) = 0$.

Example 1

$d = p$. Trivial reduction.



Example 2

$d < p$. x follows a mixture of Gaussian distribution.
$\{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots \boldsymbol{v}_p\}$ is an orthonormal basis.

$$\mathrm{x} = \sum_{i=1}^{d} \phi_i(\mathrm{x})\boldsymbol{v}_i + \sum_{i=(d+1)}^{p} \psi_i(\mathrm{x})\boldsymbol{v}_i.$$

Here, $\phi_i(\mathrm{x}) = \boldsymbol{v}_i^\top \mathrm{x}$ are the embedding functions and
$f(y_1, \ldots y_d) = \sum_{i=1}^{d} y_i \boldsymbol{v}_i$ is the reconstruction function.

Result on Unique Reconstruction

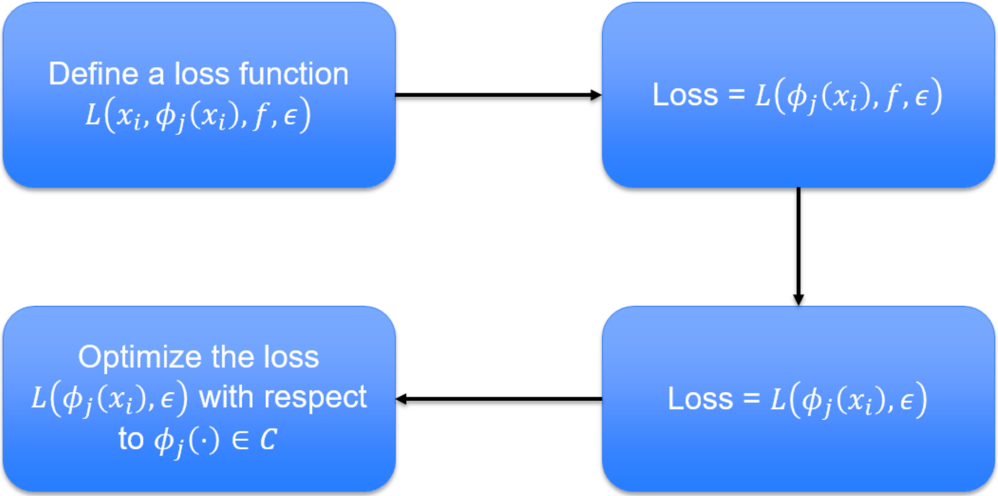
Theorem

If

$$\mathrm{x} = f(\phi_1(\mathrm{x}), \ldots \phi_d(\mathrm{x})) + \epsilon_1(\mathrm{x}) = g(\phi_1(\mathrm{x}), \ldots \phi_d(\mathrm{x})) + \epsilon_2(\mathrm{x})$$

then, $f(\phi_1(\mathrm{x}), \ldots \phi_k(\mathrm{x})) = g(\phi_1(\mathrm{x}), \ldots \phi_k(\mathrm{x}))$ *almost surely.*

Dimensionality Reduction Algorithm



Properties of the Loss function

Translation Invariance

$$L(\phi_j(\mathsf{x}_i), \epsilon) = L(a_j + \phi_j(\mathsf{x}_i), \epsilon)$$

Properties of the Loss function

Translation Invariance

$$L(\phi_j(\mathrm{x}_i), \epsilon) \;=\; L(a_j + \phi_j(\mathrm{x}_i), \epsilon)$$

Scaling Invariance

$$L(\phi_j(\mathrm{x}_i, \epsilon) \;=\; L(\lambda\phi_j(\mathrm{x}_i), \epsilon)$$

Properties of the Loss function

Translation Invariance

$$L(\phi_j(\mathsf{x}_i),\epsilon) \;=\; L(a_j+\phi_j(\mathsf{x}_i),\epsilon)$$

Scaling Invariance

$$L(\phi_j(\mathsf{x}_i,\epsilon) \;=\; L(\lambda\phi_j(\mathsf{x}_i),\epsilon)$$

Rotational Invariance

$$L(\phi_j(\mathsf{x}_i),\epsilon) \;=\; L(T\circ\phi_j(\mathsf{x}_i),\epsilon)$$

Properties of the Loss function

Translation Invariance

$$L(\phi_j(\mathsf{x}_i), \epsilon) = L(a_j + \phi_j(\mathsf{x}_i), \epsilon)$$

Scaling Invariance

$$L(\phi_j(\mathsf{x}_i, \epsilon) = L(\lambda \phi_j(\mathsf{x}_i), \epsilon)$$

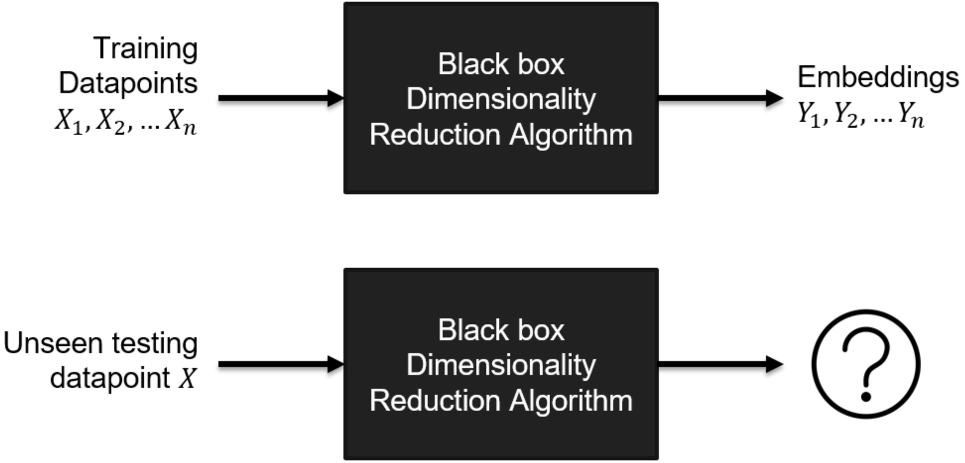
Rotational Invariance

$$L(\phi_j(\mathsf{x}_i), \epsilon) = L(T \circ \phi_j(\mathsf{x}_i), \epsilon)$$

Zero Loss

$$L(\phi_j(\mathsf{x}_i), \epsilon) = 0 \text{ iff } \epsilon = 0$$

Embedding of test datapoints



Representer Theorem

Theorem (Scholkopf et. al.)

Suppose we are given a nonempty set $\mathcal{X}$, a positive definite real-valued kernel k on $\mathcal{X} \times \mathcal{X}$, a training sample $(x_1, y_1), \ldots, (x_n, y_n) \in \mathcal{X} \times \mathbb{R}$, a strictly monotonically increasing real-valued function g on $[0, \infty]$, an arbitrary cost function $c : (\mathcal{X} \times \mathbb{R}^2)^n \to \mathbb{R} \cup \{\infty\}$, and a class of functions

$$\mathcal{F} = \left\{ f \in \mathbb{R}^{\mathcal{X}} : f(\cdot) = \sum_{i=1}^{\infty} \beta_i k(\cdot, z_i), \beta_i \in \mathbb{R}, z_i \in \mathcal{X}, \|f\| < \infty \right\}$$

Then, for any $f \in \mathcal{F}$ minimizing the regularized risk functional

$$c\left((x_1, y_1, f(x_1)), \ldots (x_n, y_n, f(x_n))\right) + g(\|f\|)$$

admits a representation of the form

$$f(\cdot) = \sum_{i=1}^{n} \alpha_i k(\cdot, x_i)$$

How to find embedding of test datapoints?

1. Assume the blackbox algorithm's output belongs to a RKHS generated by $k(\cdot, \cdot)$.

$$\phi_j(\mathrm{x}) = \sum_{i=1}^{\infty} \beta_{ij} k(\mathrm{x}, \mathrm{z}_i), \quad \text{for some } \mathrm{z}_1, \mathrm{z}_2, \cdots \in \mathbb{R}^p$$

How to find embedding of test datapoints?

1. Assume the blackbox algorithm's output belongs to a RKHS generated by $k(\cdot, \cdot)$.

$$\phi_j(\mathrm{x}) = \sum_{i=1}^{\infty} \beta_{ij} k(\mathrm{x}, \mathrm{z}_i), \quad \text{for some } \mathrm{z}_1, \mathrm{z}_2, \cdots \in \mathbb{R}^p$$

2. Representation theorem guarantees $y = \phi_j(\mathrm{x}) = \sum_{i=1}^{n} \alpha_{ij} k(\mathrm{x}, \mathrm{x}_i)$.

How to find embedding of test datapoints?

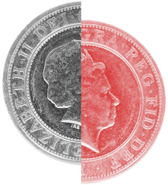
1. Assume the blackbox algorithm's output belongs to a RKHS generated by $k(\cdot,\cdot)$.

$$\phi_j(\mathrm{x}) = \sum_{i=1}^{\infty} \beta_{ij} k(\mathrm{x}, \mathrm{z}_i), \quad \text{for some } \mathrm{z}_1, \mathrm{z}_2, \cdots \in \mathbb{R}^p$$

2. Representation theorem guarantees $y = \phi_j(\mathrm{x}) = \sum_{i=1}^{n} \alpha_{ij} k(\mathrm{x}, \mathrm{x}_i)$.
3. Known values of $y_1, y_2, \ldots y_n$ for training datapoints allow to estimate α_{ij}'s.
4. $\phi_j(\mathrm{x}) = \mathrm{y}_j^\top \boldsymbol{K}^{-1}(K(\mathrm{x}))$, where $\boldsymbol{K}$ is the p.d. kernel matrix with entries $k(\mathrm{x}_i, \mathrm{x}_j)$, and $K(\mathrm{x})$ is the vector with entries $k(\mathrm{x}, \mathrm{x}_i)$.

Trustablity and Consistency Indices

Two sides of a coin!

$\mathcal{A}_n$ outputs a random subset of size d of the available variables. ⟷  ⟷ $\mathcal{A}_n$ outputs a constant function $\phi_j(\cdot) = c_j$

Does not distort the data, Inconsistency in embedding for transformed data

Distorts the data, Consistent in output under any transformation

Basic Idea of Trustablity Index

1. Ask an algorithm $\mathcal{A}_n$ to reduce a p-dimensional data into p dimensional embedding.

Basic Idea of Trustablity Index

1. Ask an algorithm $\mathcal{A}_n$ to reduce a p-dimensional data into p dimensional embedding.
2. The original data itself is the “best" embedding.

Basic Idea of Trustablity Index

1. Ask an algorithm $\mathcal{A}_n$ to reduce a p-dimensional data into p dimensional embedding.
2. The original data itself is the “best" embedding.
3. Due to invariances of loss function, the output Y should be translated, scaled and rotated version of X.

Basic Idea of Trustablity Index

1. Ask an algorithm $\mathcal{A}_n$ to reduce a p-dimensional data into p dimensional embedding.
2. The original data itself is the “best" embedding.
3. Due to invariances of loss function, the output Y should be translated, scaled and rotated version of X.
4. $$\mathsf{TI}(\mathcal{A}_n, \boldsymbol{X}) = \min_{\boldsymbol{\mu},\lambda,\boldsymbol{P}} \|\mathcal{A}_n(p, \boldsymbol{X})(\boldsymbol{X}) - \mathbf{1}_n\boldsymbol{\mu}^\top - \lambda\boldsymbol{P}\boldsymbol{X}\|_2^2$$

   such that $\boldsymbol{P}$ is orthogonal.

Trustablity Index

Definition

$$\mathrm{TI}(\mathcal{A}_n, \mathsf{X}) := \sum \mathrm{sing}(\boldsymbol{\Sigma}_{\mathsf{YY}}) - \left(\sum \mathrm{sing}(\boldsymbol{\Sigma}_{\mathsf{XX}})\right)^{-1} \left(\sum \mathrm{sing}(\boldsymbol{\Sigma}_{\mathsf{XY}})\right)^{2}$$

where $\sum \mathrm{sing}(\boldsymbol{A})$ denotes the sum of the singular values of the matrix $\boldsymbol{A}$, the reduction $\mathsf{Y} = \mathcal{A}_n(p, \mathsf{X})(\mathsf{X})$, and

$$\boldsymbol{\Sigma}_{\mathsf{XX}} = (\mathsf{X} - \overline{\mathsf{X}})^{\top}(\mathsf{X} - \overline{\mathsf{X}}),\ \boldsymbol{\Sigma}_{\mathsf{XY}} = (\mathsf{X} - \overline{\mathsf{X}})^{\top}(\mathsf{Y} - \overline{\mathsf{Y}}),\ \boldsymbol{\Sigma}_{\mathsf{YY}} = (\mathsf{Y} - \overline{\mathsf{Y}})^{\top}(\mathsf{Y} - \overline{\mathsf{Y}})$$

where $\overline{\mathsf{X}}, \overline{\mathsf{Y}}$ are matrices with each row $n^{-1}\mathbf{1}_n^{\top}\mathsf{X}$ and $n^{-1}\mathbf{1}_n^{\top}\mathsf{Y}$ respectively.

Remark

The more trustable an algorithm is, the less is the value of the index.

Basic Idea of Consistency Index

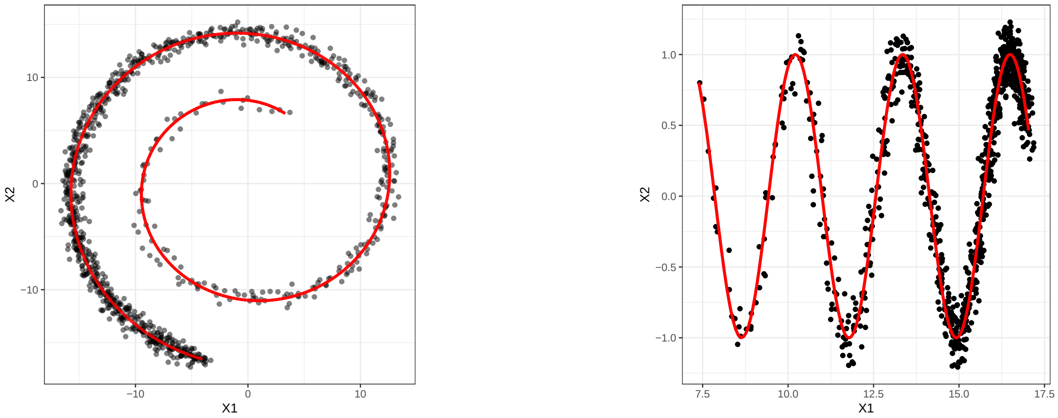


Figure: Two high dimensional data which have equivalent low dimensional underlying manifold

Basic Idea of Consistency Index

1. Apply algorithm $\mathcal{A}_n$ on X to get reduced Y with $d < p$.

Basic Idea of Consistency Index

1. Apply algorithm $\mathscr{A}_n$ on X to get reduced Y with $d < p$.
2. Based on Y, obtain a reconstruction of X as $\widehat{\mathrm{X}}$. This is unique, so getting atleast one f works.

Basic Idea of Consistency Index

1. Apply algorithm $\mathscr{A}_n$ on X to get reduced Y with $d < p$.
2. Based on Y, obtain a reconstruction of X as $\widehat{\mathrm{X}}$. This is unique, so getting atleast one f works.
3. Apply some one-one transformation T on $\widehat{\mathrm{X}}$ to get $\widehat{\mathrm{Z}} = T(\widehat{\mathrm{X}})$.

Basic Idea of Consistency Index

1. Apply algorithm $\mathcal{A}_n$ on X to get reduced Y with $d < p$.
2. Based on Y, obtain a reconstruction of X as $\widehat{\mathrm{X}}$. This is unique, so getting atleast one f works.
3. Apply some one-one transformation T on $\widehat{\mathrm{X}}$ to get $\widehat{\mathrm{Z}} = T(\widehat{\mathrm{X}})$.
4. New transformed data with same error, $\mathrm{Z} = \widehat{\mathrm{Z}} + (\mathrm{X} - \widehat{\mathrm{X}})$.

Basic Idea of Consistency Index

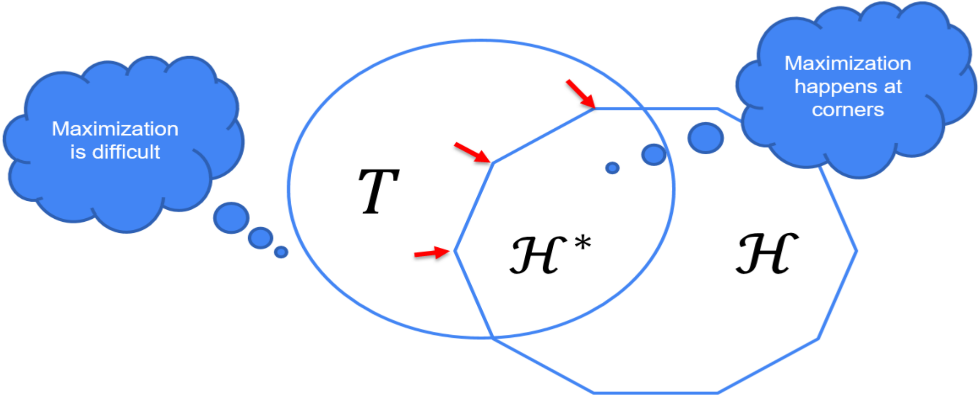
1. Apply algorithm $\mathcal{A}_n$ on X to get reduced Y with $d < p$.
2. Based on Y, obtain a reconstruction of X as $\widehat{\mathrm{X}}$. This is unique, so getting atleast one f works.
3. Apply some one-one transformation T on $\widehat{\mathrm{X}}$ to get $\widehat{\mathrm{Z}} = T(\widehat{\mathrm{X}})$.
4. New transformed data with same error, $\mathrm{Z} = \widehat{\mathrm{Z}} + (\mathrm{X} - \widehat{\mathrm{X}})$.
5. $\mathcal{A}_n$ should output embedding that is translated, rotated and scaled version of Y.

Basic Idea of Consistency Index

1. Apply algorithm $\mathcal{A}_n$ on X to get reduced Y with $d < p$.
2. Based on Y, obtain a reconstruction of X as $\widehat{\mathsf{X}}$. This is unique, so getting atleast one f works.
3. Apply some one-one transformation T on $\widehat{\mathsf{X}}$ to get $\widehat{\mathsf{Z}} = T(\widehat{\mathsf{X}})$.
4. New transformed data with same error, $\mathsf{Z} = \widehat{\mathsf{Z}} + (\mathsf{X} - \widehat{\mathsf{X}})$.
5. $\mathcal{A}_n$ should output embedding that is translated, rotated and scaled version of Y.
6.

$$\mathrm{CI}(\mathcal{A}_n, \mathsf{X}, d) = \max_{T \text{ is invertible}} \min_{\boldsymbol{\mu},\lambda,\boldsymbol{P}} \|\mathcal{A}_n(d, \mathsf{Z})(\mathsf{Z}) - \mathbf{1}_n \boldsymbol{\mu}^\intercal - \lambda \boldsymbol{P} \mathcal{A}_n(d, \mathsf{X})(\mathsf{X})\|_2^2$$

Problem with Maximization



T = Space of all invertible functions.
$\mathcal{H}$ = Space of all functions belonging to a Reproducing Kernel Hilbert Space generated by a vector valued universal kernel $k(\cdot,\cdot)$
$\mathcal{H}^* = \mathcal{H} \cap T$

Tractable Consistency Index

Restrict T to be inside the space,

$$\mathcal{H}^* = \Bigg\{ f : f \in \mathcal{H}, f(\cdot) = \sum_{i=1}^{\infty} \sum_{j=1}^{p} \beta_{ij} k(\cdot, \mathrm{x}_i) \boldsymbol{e}_j, 0 \leq \beta_{ij} \leq 1, \\ \text{and } \sum_{i=1}^{\infty} \sum_{j=1}^{p} \beta_{ij} = 1, \text{ and } f \text{ is invertible} \Bigg\}.$$

Tractable Consistency Index

Restrict T to be inside the space,

$$\mathcal{H}^* = \left\{ f : f \in \mathcal{H}, f(\cdot) = \sum_{i=1}^{\infty} \sum_{j=1}^{p} \beta_{ij} k(\cdot, \mathrm{x}_i) \boldsymbol{e}_j, 0 \leq \beta_{ij} \leq 1, \right.$$
$$\left. \text{and } \sum_{i=1}^{\infty} \sum_{j=1}^{p} \beta_{ij} = 1, \text{ and } f \text{ is invertible} \right\}.$$

Theorem

If $S(T) = \min_{\boldsymbol{\mu},\lambda,\boldsymbol{P}} \|\mathcal{A}_n(d, \tilde{\mathrm{X}})(\tilde{\mathrm{X}}) - \mathbf{1}_n \boldsymbol{\mu}^\top - \lambda \boldsymbol{P} \mathcal{A}_n(d, \mathrm{X})(\mathrm{X})\|_2^2$*, is a convex function in* T*, then* $\max_{T \in \mathcal{H}^*} S(T) = \max_{T \in \partial\mathcal{H}^*} S(T)$*, where,*

$$\partial\mathcal{H}^* = \left\{ k(\cdot, \mathrm{x}_i) \boldsymbol{e}_j : i = 1, \ldots n;\ j = 1, \ldots p \right\}$$

i.e. the maximum must occur at the boundary.

Simulation Studies

Trustablity Index

Dataset	PCA	kPCA	LLE	HLLE	LE	tSNE	UMAP
Gaussian Cluster	0	138.727	10.112	0.084	**0.083**	128.366	91.807
Hypercube	0	168.667	9.997	0.822	**0.018**	32.007	4.885
Hypersphere	0	32.781	2.991	**0.011**	0.014	4.917	11.701
Swiss Roll	0	112.678	3	**0.021**	0.034	142.131	60.023
Sonar	0	56.897	59.984	59.983	**0.288**	2.771	9.826
WBCD	0	30.867	30.021	33.309	**0.052**	39.205	64.206
COIL2000	0	118.322	86.001	92.116	**0.172**	11.819	38.311

Table: Normalized Trustability Index ($n^{-1}\mathrm{TI}(\mathcal{A}_n, \mathrm{X})$) of various algorithms on various datasets

Consistency Index

Dataset	PCA	kPCA	LLE	HLLE	LE	tSNE	UMAP
Gaussian Cluster	58434.22	**9.31**	232.511	203.115	153.313	23.963	11.818
Hypercube	143.233	2.946	6.305	8.818	8.003	3.116	**2.291**
Hypersphere	67.414	10.065	15.337	29.212	29.252	13.535	**2.904**
Swiss Roll	72.212	5.193	34.611	31.995	27.212	3.998	**3.184**
Sonar	58.414	**2.854**	18.884	21.229	18.818	3.971	4.498
WBCD	1.234×10^6	3.214	48.913	74.227	70.212	**2.105**	2.816
COIL2000	109783.7	4.009	69.913	75.200	71.200	**3.211**	2.877

Table: Tractable Consistency Index ($n^{-1}\mathrm{TCI}(\mathcal{A}_n, \mathrm{X})$) of various algorithms on various datasets

Conclusion

1. Eigenfunction and eigenvalue based algorithms are more trustable.
2. tSNE, UMAP these complex algorithms have better consistency under nonlinear transformation of the data.
3. Laplacian Eigenmaps is better than Hessian LLE in both indices.
4. Graphical algorithms are generally better. While NN Graph does not work, a graph MCST $\subset G \subset$ Delauney Triangulation should be better. Gabriel Graph or Bela skeleton is a choice.

Bibliography I

Alain Berlinet and Christine Thomas-Agnan (2011). *Reproducing kernel Hilbert spaces in probability and statistics*. Springer Science & Business Media.

Keinosuke Fukunaga (2013). *Introduction to statistical pattern recognition*. Elsevier.

John C Gower, Garmt B Dijksterhuis, et al. (2004). *Procrustes problems*. Vol. 30. Oxford University Press on Demand.

Samuel Kaski et al. (2003). "Trustworthiness and metrics in visualizing similarity of gene expression". In: *BMC bioinformatics* 4.1, pp. 1–13.

Bibliography II

B. Li (n.d.). *Sufficient Dimension Reduction: Methods and Applications with R*. Chapman & Hall/CRC Monographs on Statistics and Applied Probability. CRC Press. ISBN: 9781351645737.

Leland McInnes, John Healy, and James Melville (2018). *UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction*. arXiv: 1802.03426 [stat.ML].

Sam T Roweis and Lawrence K Saul (2000). "Nonlinear dimensionality reduction by locally linear embedding". In: *science* 290.5500, pp. 2323–2326.

Laurens Van Der Maaten, Eric Postma, and Jaap Van den Herik (2009). "Dimensionality reduction: a comparative". In: *J Mach Learn Res* 10.66-71, p. 13.

THANK YOU