

Constructing the Low-Dimensional Data Representation in Data Analysis

Alexander Kuleshov

Kharkevich Institute for Information Transmission Problems RAS
(IITP RAS)

General

- Many Machine Learning/Data Mining/Data Analysis tasks deal with real-world data that are presented in high-dimensional spaces, and the '**curse of dimensionality**' phenomena is often an obstacle to the use of many methods for solving the tasks.
- To avoid the phenomena, the original high-dimensional data are transformed into their lower-dimensional representations (features).
- Representation (Feature) learning problems are usually formulated as various Dimensionality Reduction/Feature Selection problems.

- Data Representation algorithms are used usually as a first key step in solutions of the Data Analysis tasks: after this step, the initial tasks may be reduced to the corresponding tasks for the constructed lower-dimensional representations of the original data.
- A construction of the low-dimensional data features for their subsequent use in specific Data Analysis tasks must depend on the tasks themselves: the quality of a final task solution generally depends on the choice of the lower dimensional data representation.
- Thus, **there are different formulations of the Dimensionality Reduction problems depending on the further Data Analysis tasks to be performed.**

The goal of the talk is to describe a few key Data Analysis tasks that lead to different formulations of the Dimensionality Reduction problems:

- Sample Embedding,
- Data Space (Manifold) embedding,
- Manifold Reconstruction,
- Tangent Bundle Manifold Learning,

and to present shortly a new geometrically motivated algorithm that solves these problems

Main points of the talk:

- **Geometrical approach to Dimensionality Reduction**

The general goal of Machine Learning/Data Mining/Data Analysis tasks is to extract previously unknown information from a dataset. Thus, it is supposed that the target information is reflected in the structure of the dataset which must be discovered from the data:

‘Machine learning is about the shape of data’

(Vejdemo-Johansson, Topological Methods for Machine Learning, an ICML 2014 Workshop).

In particular, Dimensionality Reduction problems consist in extracting a low-dimensional structure from high-dimensional data, and geometric methods have now become the central methodology for finding such structure in data.

- **Cost functions concept in Dimensionality Reduction**

The various Dimensionality Reduction methods are different due to choosing some **optimized cost function** which defines an 'evaluation measure' for the Dimensionality Reduction problem and reflects desired properties of its solution:

'A general view on the Dimensionality Reduction can be based on the concept of cost functions'

(Bunte et al., IEEE Symposium Series in Computational Intelligence, 2011).

Thus, a solution of the Dimensionality Reduction consist in:

- ✓ a choice of the appropriate cost function;
- ✓ an optimization of the chosen cost function.

Outline

- Key Data Analysis tasks and corresponding Dimensionality Reduction problems
- Unified solution for all formulated problems
- Properties of the proposed solution

Key Data Analysis tasks and Dimensionality Reduction

1. Clustering problem:

to divide the given dataset $\mathbf{X}_n = \{X_1, X_2, \dots, X_n\} \subset \mathbb{R}^p$ into several groups: $\mathbf{X}_n = \mathbf{X}^{(1)} \cup \mathbf{X}^{(2)} \cup \dots \cup \mathbf{X}^{(m)}$, that contain '**similar**' (in one sense or another) sample points.

A reducing of the initial high-dimensional Clustering problem to certain low-dimensional task requires a solving of the **Sample Embedding problem**: Given an input dataset \mathbf{X}_n , find an 'n-point' Embedding mapping

$$h_{(n)}: \mathbf{X}_n \rightarrow \mathbf{h}_n = h_{(n)}(\mathbf{X}_n) = \{h_1, h_2, \dots, h_n\} \subset \mathbb{R}^q,$$

Into the q-dimensional dataset \mathbf{h}_n , $q < p$, such that the resulting n-point dataset \mathbf{h}_n '**faithfully represents**' the sample \mathbf{X}_n

Term '***faithfully represents***' in Sample Embedding is not formalized: in various methods it defined by choosing optimized cost function $L_{(n)}(\mathbf{h}_n|\mathbf{X}_n)$ - an 'evaluation measure' reflecting the desired properties of the Embedding mapping $h_{(n)}$.

If '***faithfully represents***' in the Sample Embedding problem corresponds to the '***similar***' notion in the initial Clustering, then:

- we can find the solution $h_{(n)}(\mathbf{X}_n) = \mathbf{h}^{(1)} \cup \mathbf{h}^{(2)} \cup \dots \cup \mathbf{h}^{(m)}$ of the **Reduced Clustering problem** for the constructed low-dimensional feature dataset $\mathbf{h}_n = \{h_1, h_2, \dots, h_n\} \subset R^q$;
- using inverse mapping $h^{-1}: \{h_1, h_2, \dots, h_n\} \rightarrow \{X_1, X_2, \dots, X_n\}$, construct a partition of the sample \mathbf{X}_n into groups

$$\{\mathbf{X}^{(k)} = h^{-1}(\mathbf{h}^{(k)}), \quad k = 1, 2, \dots, m\},$$

and take it as a solution of the initial high-dimensional Clustering problem.

Examples of Sample Embedding algorithms:

Feature dataset \mathbf{h}_n minimizes chosen cost function $L_n(\mathbf{h}_n|\mathbf{X}_n)$:

Multi Dimensional Scaling (MDS):

$$L_{n,\text{MDS}}(h_1, h_2, \dots, h_n|\mathbf{X}_n) = \sum_{i,j=1}^n (\rho(X_i, X_j) - \|h_i - h_j\|)^2,$$

where ρ is a chosen distance function in \mathbb{R}^p .

If $\rho(X_i, X_j) = \|X_i - X_j\|$ is the Euclidean metric in \mathbb{R}^p , then

MDS = PCA (Principal Component Analysis).

Laplacian Eigenmaps (LE): (Belkin et al., Neural Computation, 2003)

$$L_{n,\text{LE}}(h_1, h_2, \dots, h_n|\mathbf{X}_n) = \sum_{i,j=1}^n K_E(X_i, X_j) \times \|h_i - h_j\|^2$$

under certain normalizing conditions required to avoid a degenerate solution; here $K_E(X, X') = 1$, if $|X' - X| < \varepsilon$, and $= 0$, otherwise, is an Euclidean measure of proximity (kernel).

This sample cost function $L_{n,LE}(h_1, h_2, \dots, h_n | \mathbf{X}_n)$ is discrete analogue of Laplace-Beltrami operator on manifold.

'Out-of-Sample extension' of the Laplacian Eigenmaps:

given the sample, to find the function $h(\mathbf{X}) \in L_2(\mathbf{X})$ on manifold \mathbf{X} to minimize the cost function

$$\int_{\mathbf{X}} \|\nabla h(\mathbf{X})\|^2 \text{mes}(d\mathbf{X}) = \int_{\mathbf{X}} (L_{LB}(h) \times h)(\mathbf{X}) \text{mes}(d\mathbf{X})$$

(by Stokes theorem), here $L_{LB}h = -\text{div}(\nabla h)$ is the Laplace-Beltrami operator on the Data Manifold \mathbf{X} .

The solution $h(\mathbf{X}) = \begin{pmatrix} h_1(\mathbf{X}) \\ \vdots \\ h_q(\mathbf{X}) \end{pmatrix} \in \mathbb{R}^q$ consists of certain

eigenfunctions $\{h_1(\mathbf{X}), \dots, h_q(\mathbf{X})\}$ of Laplace-Beltrami operator.¹¹

2. Classification problem:

given the inputs $X_n = \{X_1, X_2, \dots, X_n\} \subset R^p$ equipped by outputs (labels) $\Lambda_n = \{\lambda_1, \lambda_2, \dots, \lambda_n\}$ belonging to $\{1, 2, \dots, m\}$, $m \geq 2$, to find an output (label) λ for a new, previously unseen, input X .

Reduced 'lower-dimensional' Classification problem

- Low-dimensional features $\{h_1, h_2, \dots, h_n\} \subset R^q$, $q < p$, for the original high-dimensional inputs $\{X_1, X_2, \dots, X_n\}$ can be constructed by using certain Sample Embedding algorithm.
- Reduced 'lower-dimensional' Classification problem in the Feature space: based on the reduced dataset $\{(h_i, \lambda_i)\}$ instead of the original dataset $\{X_n, \Lambda_n\}$, to find an output (label) λ for a new, previously unseen, feature input $y \in R^q$.

To use the solution of the reduced Classification problem for the original Classification problem, it is necessary to construct a lower-dimensional representation $h = h(X)$ (feature) for a new unseen high-dimensional input $X \notin \mathbf{X}_n$ (Out-of-Sample point).

Thus, an 'Out-of-Sample extension' of the Sample Embedding is required which leads to the **Data Space Embedding** problem:

given an input dataset \mathbf{X}_n , sampled from an unknown **Data Space** $\mathbf{X} \subset \mathbb{R}^p$, find an Embedding mapping

$$h: \mathbf{X} \subset \mathbb{R}^p \rightarrow \mathbf{Y} = h(\mathbf{X}) \subset \mathbb{R}^q$$

from the high-dimensional Data space \mathbf{X} to the low-dimensional **Feature Space** \mathbf{Y} , which **preserves specific properties** of the Data Space.

The definition uses values $h(X)$ for the Out-of-Sample points X ; so, we must define a **Data Model** describing the Data space \mathbf{X} and a way for extracting both the sample \mathbf{X}_n and Out-of-Sample points from the Data space \mathbf{X} .

The most popular Data model is **Manifold Data Model**:

- Data space \mathbf{X} is q -dimensional manifold (**Data manifold**) embedded in an ambient p -dimensional space R^p , $q < p$;
- the points are selected from the Data Manifold independently of each other according to the probability measure μ on the Data Manifold \mathbf{X} whose support $\text{Supp}(\mu)$ is \mathbf{X} .

This **Manifold assumption** is based on following empirical fact: as a rule, high-dimensional **real-world** data lie on or near some unknown low-dimensional Data Manifold embedded in an ambient high-dimensional ‘observation’ space.

Dimensionality Reduction under the **Manifold Data Model** is usually referred to as the **Manifold Learning**.

Data space Embedding under the model can be referred to as the **Manifold Embedding** problem: to find a low-dimensional parameterization of the Data Manifold (DM) determined by the Embedding mapping h which preserves certain specific geometrical and topological properties of the DM like local data geometry, proximity relations, geodesic distances, angles, etc.

The Manifold data model allowed applying to Dimensionality Reduction of a wide range of modern differential-geometric and topological methods, to reveal the essence of the various algorithms and to use these methods for solving various application problems.

3. Reduced Data analysis problems (Prognosis, Optimization, ...) in the Feature space

Manifold Embedding techniques allow reducing the initial high-dimensional problem to the similar low-dimensional problems in Feature space. This can produce 'Feature Out-of-Sample' points $y \notin \mathbf{Y}_n = h(\mathbf{X}_n)$ in the lower-dimensional Feature space as solutions of the various problems (Prognosis, Optimization, etc.) in the Reduced problems.

This, in turn, leads to the problem of reconstructing the high-dimensional points $X = 'h^{-1}')(y) \in \mathbf{X}$ from their low-dimensional features $y = h(X)$ for all $y \in \mathbf{Y}$ (not only for feature sample points $y_i = h(X_i)$ whose reconstructed values $X_i = 'h^{-1}')(y_i)$ are obvious).

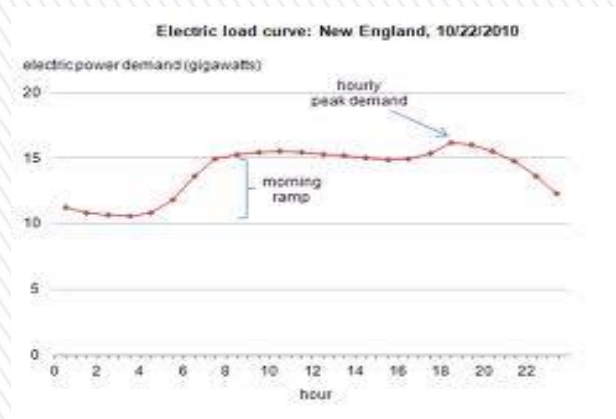
Example: Electricity price forecasting

(J. Chen et al. Electricity price curve modeling and forecasting by manifold learning // IEEE Transaction on power systems, vol. 23, № 3, 2008)

Electricity “daily prices” are described by a multidimensional time series (called ‘electricity price curve’)

$$X_t = (X_{t1}, X_{t2}, \dots, X_{t,24})^T \in \mathbb{R}^{24}, \quad t = 1, 2, \dots, T$$

consisting of “hour-prices” in the course of day t .




Based on ‘daily-prices’ vector set $\mathbf{X}_{1:T} = \{X_1, X_2, \dots, X_T\} \subset \mathbb{R}^{24}$ up to day T , it is necessary to construct a forecast \hat{X}_{T+1} for X_{T+1} .

The 'daily prices' 24-dimensional vectors X_t lie near the q -dimensional manifold in R^{24} ; $q = 4$ is selected as an appropriate dimension of the manifold (J. Chen et al.). This allows (J. Chen et al.) to use the following Manifold Learning Forecasting algorithm for Electricity price forecasting problem:

Forecasting algorithm (J. Chen et al.)

Step 1 (Embedding to the Feature space). Using Embedding mapping h (certain solution of Manifold Embedding problem for the sample $\mathbf{X}_{1:T} \subset R^{24}$), the 4-dimensional feature vector set $\mathbf{Y}_{1:T} = \{Y_1, Y_2, \dots, Y_T\} = h(\mathbf{X}_{1:T}) \subset R^4$ is constructed.

Step 2 (Forecasting in the Feature space). Based on 4-dimensional time series $\{Y_t, t = 1, 2, \dots, T\}$, a forecast \hat{Y}_{T+1} for Y_{T+1} is constructed by using standard forecasting technique. 

In the general case, the forecast \hat{Y}_{T+1} doesn't belong to the sample set $\mathbf{Y}_{1:T}$ and is Feature-Out-of-Sample point. Thus, it is required to reconstruct the forecast \hat{X}_{T+1} from the 'low-dimensional' forecast \hat{Y}_{T+1} .

Certain reconstructing Locally Linear Embedding mapping g_{LLE} from the Feature space to high-dimensional ambient space \mathbb{R}^{24} (Saul et al., JMLR, 2003) was used in the Forecasting algorithm:

Step 3 (Forecasting in the 'daily prices' curve space).

$$\hat{X}_{T+1} = g_{\text{LLE}}(\hat{Y}_{T+1}).$$

Manifold Reconstructing problem

Given an input dataset \mathbf{X}_n sampled from a q -dimensional Data Manifold \mathbf{M} covered by a single chart and embedded in an ambient p -dimensional Euclidean space R^p , $q < p$, construct a solution $\theta = (h, g)$ consisting of two interrelated mappings:

- an Embedding mapping $h: \mathbf{M} \subset R^p \rightarrow \mathbf{Y}_\theta = h(\mathbf{M}) \subset R^q$,
- a Reconstructing mapping $g: \mathbf{Y}_\theta \subset R^q \rightarrow \mathbf{M} \subset R^p$,

which determines Reconstructed value $r_\theta(X) = g(h(X))$ as a result of successively applying embedding and reconstruction mappings to a vector $X \in \mathbf{M}$, and the reconstruction error $\delta_\theta(X) = |X - r_\theta(X)|$ is a measure of quality of solution θ at point X .

The solution must ensure small reconstruction error for all $X \in \mathbf{M}$, which implies

$$X \approx r_\theta(X) \quad \text{for all } X \in \mathbf{M}.$$

Solution θ determines q -dimensional **Reconstructed manifold**

$$\mathbf{M}_\theta = \{X = g(y) \in \mathbb{R}^p: y \in \mathbf{Y}_\theta \subset \mathbb{R}^q\}$$

in \mathbb{R}^p and the above approximate equalities $X \approx r_\theta(X)$ can be considered as **Manifold Proximity**

$$\mathbf{M} \approx \mathbf{M}_\theta = r_\theta(\mathbf{M})$$

meaning that the Reconstructed Manifold \mathbf{M}_θ accurately reconstructs the Data Manifold \mathbf{M} from the sample \mathbf{X}_n .

The solution θ allows reconstructing the Data Manifold \mathbf{M} by the parameterized Reconstructed Manifold, whereas the Embedding Manifold solution h reconstructs a low-dimensional parameterization of the Data Manifold only.

4. An generalization ability of Manifold Reconstruction solution

The reconstruction error $\delta_\theta(X) = |X - r_\theta(X)|$:

- can be directly computed at sample points $X \in \mathbf{M}_n$.
- describes an generalization ability of the solution θ at the Out-of-Sample points $X \in \mathbf{M} \setminus \mathbf{X}_n$.

It follows from the upper and lower bounds for local generalization ability of an arbitrary solution θ (Kuleshov et al., Manifold Learning: generalizing ability and tangent proximity, IJSL, 7(3), 359 - 390, 2013) that **the greater** the distance between the tangent spaces to the Data Manifold \mathbf{M} and Reconstructed manifold \mathbf{M}_θ at the point $X \in \mathbf{M}$ and the reconstructed point $r_\theta(X) \in \mathbf{M}_\theta$, respectively, **the lower** the local generalization ability of the solution θ becomes.

These reasons led to what that the solution θ must ensure not only **Manifold proximity**

$$X \approx r_{\theta}(X) = g(h(X)) \quad \text{for all } X \in \mathbf{M},$$

but also **Tangent proximity**

$$L(X) \approx L_{\theta}(r_{\theta}(X)) \quad \text{for all } X \in \mathbf{M},$$

where

- $L(X)$ be tangent space to Data manifold \mathbf{M} at the points X ;
- $L_{\theta}(r_{\theta}(X))$ be tangent space to the manifold \mathbf{M}_{θ} at the reconstructed point $r_{\theta}(X)$,

$L_{\theta}(r_{\theta}(X)) = \text{Span}(J_g(h(X)))$ - q -dimensional linear space in R^p spanned the columns of the Jacobian $J_g(h(X))$ ($p \times q$ matrix) of mapping $g(y)$ at the point $y = h(X)$.

In topology, the set $TB(\mathbf{M}) = \{(X, L(X)): X \in \mathbf{M}\}$ composed of the points X of the manifold \mathbf{M} equipped by tangent spaces $L(X)$ at these points is known as **Tangent Bundle of the manifold \mathbf{M}** .

Thus, accurate reconstructing both the manifold and its tangent spaces from the sample is **Tangent Bundle Manifold Learning problem** which is an amplification of the Manifold Reconstruction problem..

Tangent Bundle Manifold Learning: Given a sample from Data Manifold \mathbf{M} , construct a solution $\theta = (h, g)$ which provides:

$$g(h(X)) \approx X \quad \textbf{(Manifold proximity)}$$

$$\text{Span}(J_g(h(X))) \approx L(X) \quad \textbf{(Tangent proximity)}$$

for all $X \in \mathbf{M}$, or, shortly, **Tangent Bundle proximity**

$$TB(\mathbf{M}_\theta) = \{(r_\theta(X), L_\theta(r_\theta(X))): X \in \mathbf{M}\} \approx TB(\mathbf{M}) = \{(X, L(X)): X \in \mathbf{M}\}.$$

Unified solution for all formulated problems

(Kuleshov et al., Tangent Bundle Manifold Learning via Grassmann&Stiefel Eigenmaps. In arXiv preprint, December of 2012); Kuleshov et al., Manifold Learning in Data Mining Tasks. Lecture Notes in Artificial Intelligence, Vol. 8556, Springer, 119 – 133, 2014)

A new geometrically motivated method called

Grassmann&Stiefel Eigenmaps (GSE)

solves the Tangent Bundle Manifold Learning problem and gives **new solutions** for all the above defined Dimensionality reduction problems (Sample Embedding, Manifold Embedding, Manifold Reconstruction).

GSE consists of three successively performed main parts:

- Part I (Tangent Manifold Learning),
- Part II (Manifold Embedding),
- Part III (Manifold Reconstruction).

Tangent Manifold Learning step:

to construct a **data-based** family $\mathbf{H} = \{H(X), X \in \mathbf{M}\}$ consisting of $p \times q$ matrices $H(X)$ smoothly depending on $X \in \mathbf{M}$ in such a way that the linear spaces $L_H(X) = \text{Span}(H(X))$ spanned by columns of the matrices $H(X)$ approximate the tangent spaces $L(X)$ to the unknown Data Manifold:

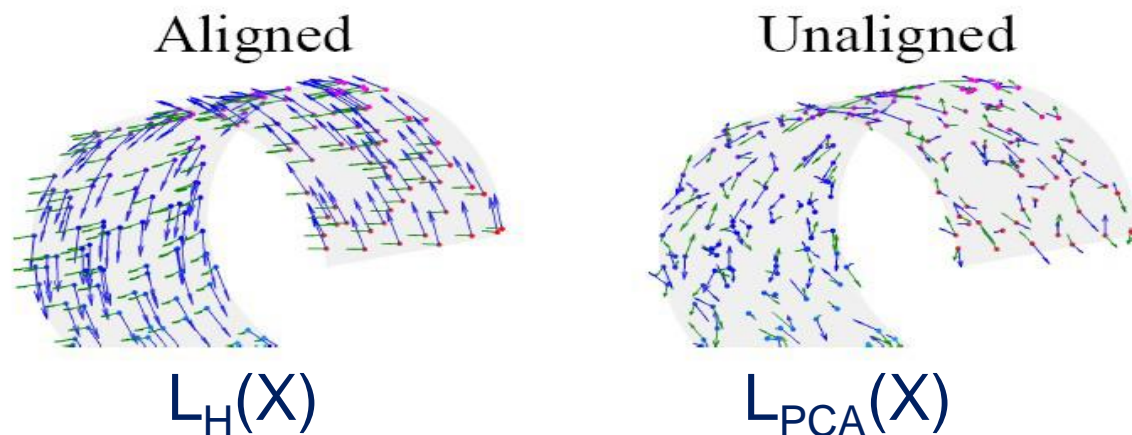
$$L_H(X) \approx L(X) \quad \text{for all } X \in \mathbf{M}.$$

Thus, the set $\mathbf{L}_H = \{L_H(X), X \in \mathbf{M}\}$ approximates the Tangent manifold $\mathbf{L} = \{L(X), X \in \mathbf{M}\}$ (Tangent Manifold Learning).

Tangent Manifold Learning step includes a solving of **two optimization problems**:

1) **PCA**. Constructing the $p \times q$ matrices $Q_{\text{PCA}}(X)$ whose columns are the “first” eigenvectors obtained by applying the PCA to the sample's points from small vicinity of the point $X \in \mathbf{M}$.

The new Robust Principal Component Analysis method proposed by B. Polyak (Polyak, Robust Principal Component Analysis, WIAS/PreMoLab Workshop, Berlin, 2013) was used here.



The, the constructed PCA-spaces $L_{\text{PCA}}(X) = \text{Span}(Q_{\text{PCA}}(X))$ are need to be aligned.


2) PCA-spaces alignment

The matrices $H(X)$ are constructed in the form:

$$H(X) = Q_{PCA}(X) \times v(X),$$

where $q \times q$ matrices $v(X)$ at the sample points are the solution of Averaged Procrustes problem: to minimize the quadratic form

$$\frac{1}{2} \sum_{i,j=1}^n K(X_i, X_j) \times \|Q_{PCA}(X_i) \times v(X_i) - Q_{PCA}(X_j) \times v(X_j)\|_F^2$$

over the matrices $\{v(X_i)\}$ under certain normalizing conditions required to avoid a degenerate solution; here $K(X, X')$ is the constructed data-based kernel reflecting not only geometrical nearness between the points X and X' on the Data Manifold but also nearness between the tangent spaces $L(X)$ and $L(X')$. 

The exact solution of this minimizing problem can be obtained as solution of specified generalized eigenvector problem:

$$\Phi \times V = \lambda \times F \times V,$$

where desired $(nq) \times q$ matrix V is consists of the n sequentially recorded transposed submatrices $v(X_1), v(X_2), \dots, v(X_n)$; F and Φ are the explicitly written $nq \times nq$ data-based matrices.

Commonly applied packages for the solution of this standard problem of linear algebra did not provide the required accuracy, and **B. Polyak proposed to use a certain effective numerical scheme** for the minimization of the initial quadratic form, which has been used in the software implementation of the GSE algorithm.

Given the family \mathbf{H} already constructed, the mappings h and g are built in next steps in such a way as to provide the conditions:

$$g(h(X)) \approx X,$$

$$J_g(h(X)) \approx H(X).$$

Manifold Embedding step:

Taylor series expansions for g :

$$g(h(X')) - g(h(X)) \approx J_g(h(X)) \times (h(X') - h(X))$$

for near points X, X' under above conditions imply the relations:

$$X' - X \approx H(X) \times (h(X') - h(X)).$$

These equations written for all the nearby sample points can be considered as the regression equalities for the values $\{h(X_i)\}$ and Least Squares is used to estimate them .

The mapping $y = h(X)$ determines Feature space $\mathbf{Y}_\theta = h(\mathbf{M})$.

Manifold Reconstruction step:

Given the family \mathbf{H} and values $\{y_j = h(X_j)\}$ already constructed, the Taylor series expansions

$$g(h(X)) - g(h(X_j)) \approx J_g(h(X_j)) \times (h(X) - h(X_j))$$

for near points X and X_j , with taking into account the relations $y_j = h(X_j)$, $g(y_j) \approx X_j$ and $J_g(h(X_j)) \approx H(X_j)$ can be written as

$$g(y) - X_j \approx H(X_j) \times (y - y_j)$$

the near values $y = h(X)$ and y_j .

These relations allow to estimate $g(y)$ at the point y from the near feature points $\{y_j\}$ and both the known values $\{g(y_j) \approx X_j\}$ and the known Jacobian $J_g(y_j) \approx H(X_j)$ at these points.

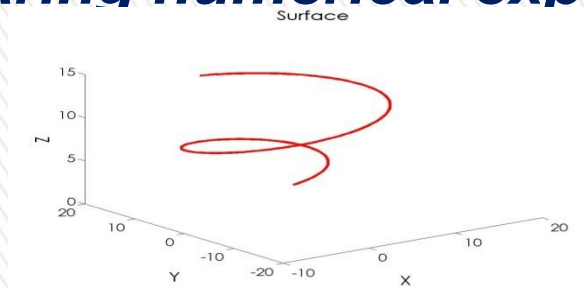
Properties of the proposed solution

Theoretical properties: The GSE has optimal convergence rate as $n \rightarrow \infty$ (Kuleshov et al., Asymptotically optimal method in Manifold estimation, XXIX-th European Meeting of Statisticians, 2013): **with high probability**

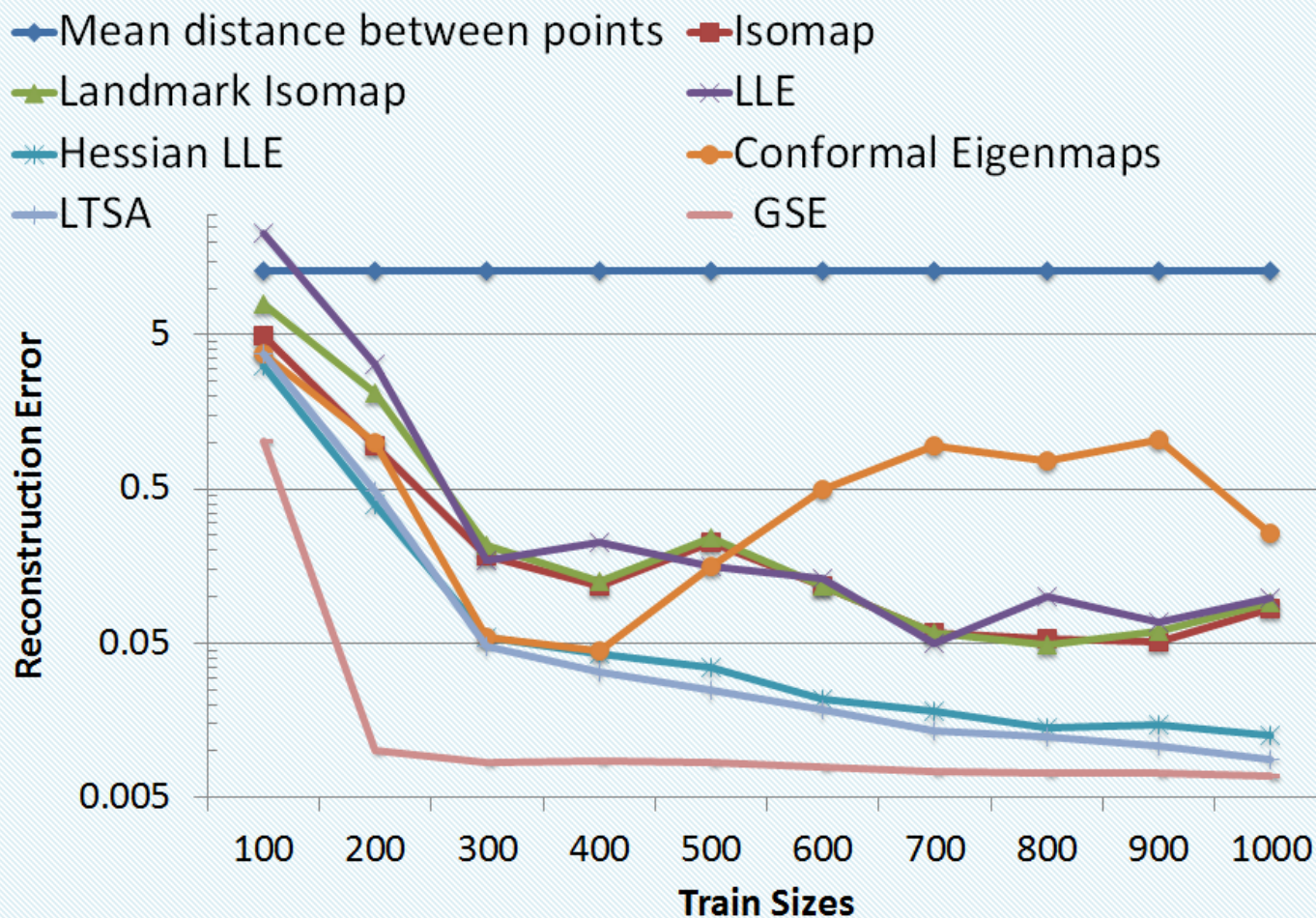
$$\delta_{\theta}(X) = |X - r_{\theta}(X)| = O(n^{-2/(q+2)}) \quad \text{for all } X \in \mathbf{M};$$

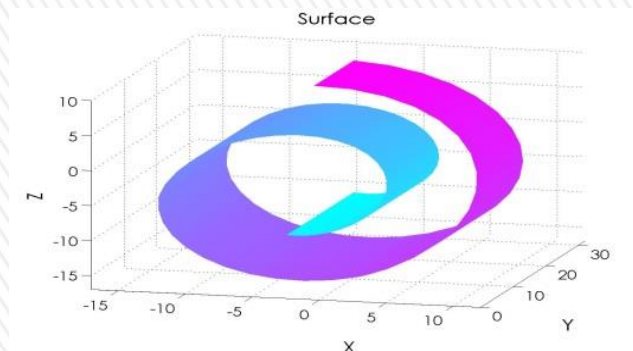
this rate coincides with the asymptotically minimax lower bound for the Hausdorff distance $H(\mathbf{M}_{\theta}, \mathbf{M})$ ($\leq \sup_{X \in \mathbf{M}} \delta_{\theta}(X)$) between the Data Manifold \mathbf{M} and Reconstructed Manifold \mathbf{M}_{θ} (Genovese et al., Minimax Manifold Estimation, JMLR, 2012).

Comparing numerical experiments

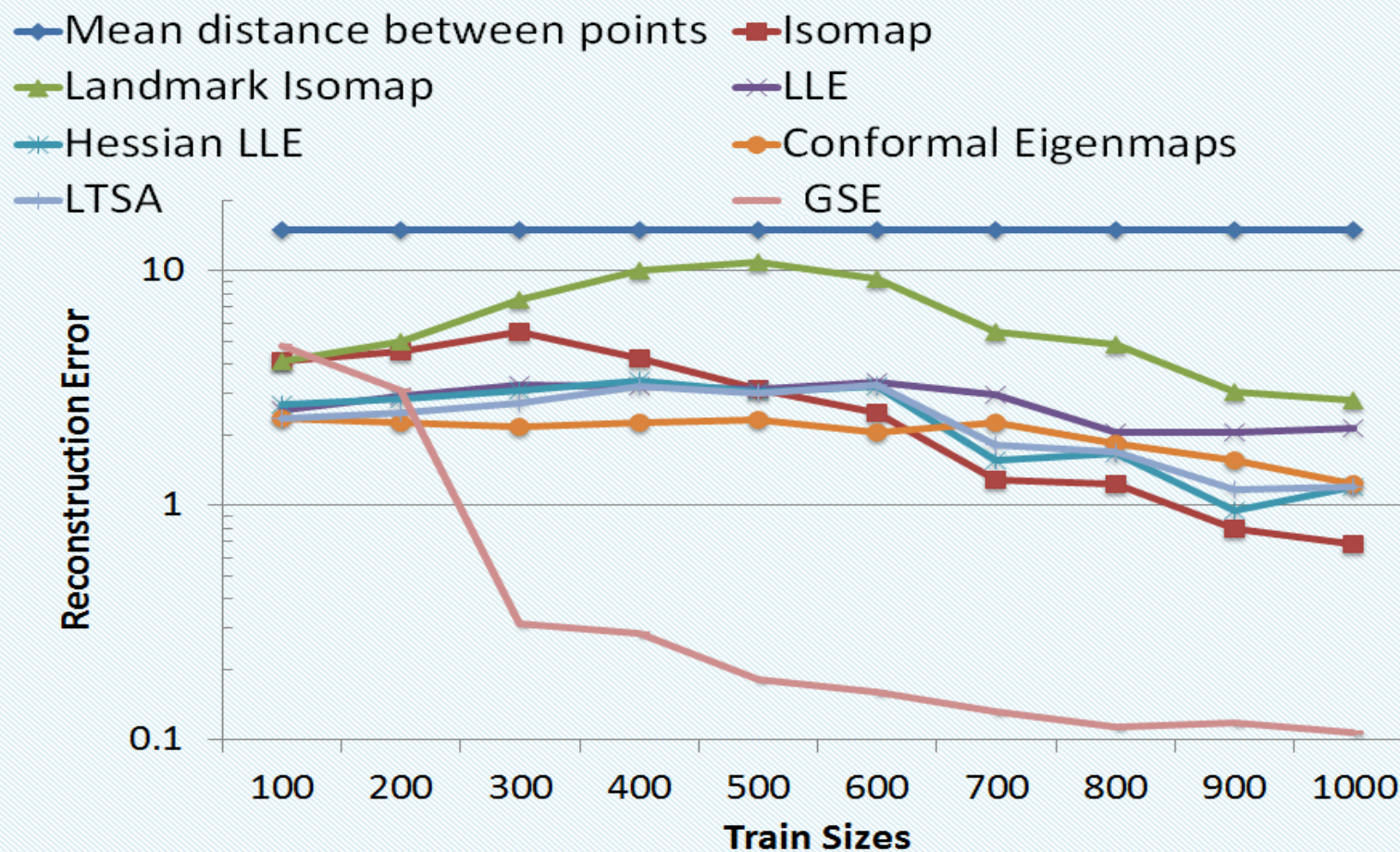


Spiral

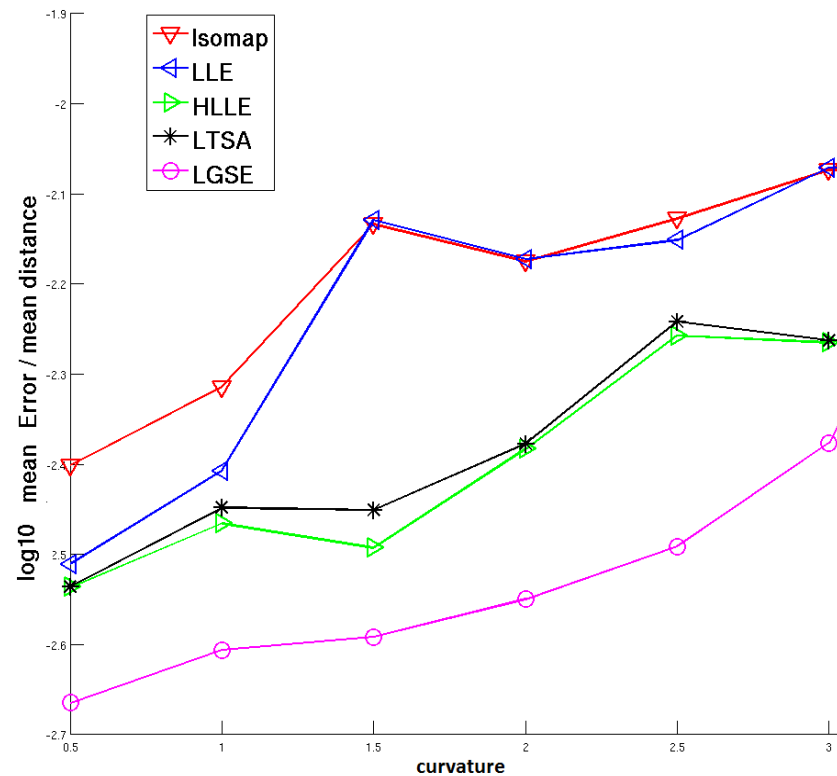
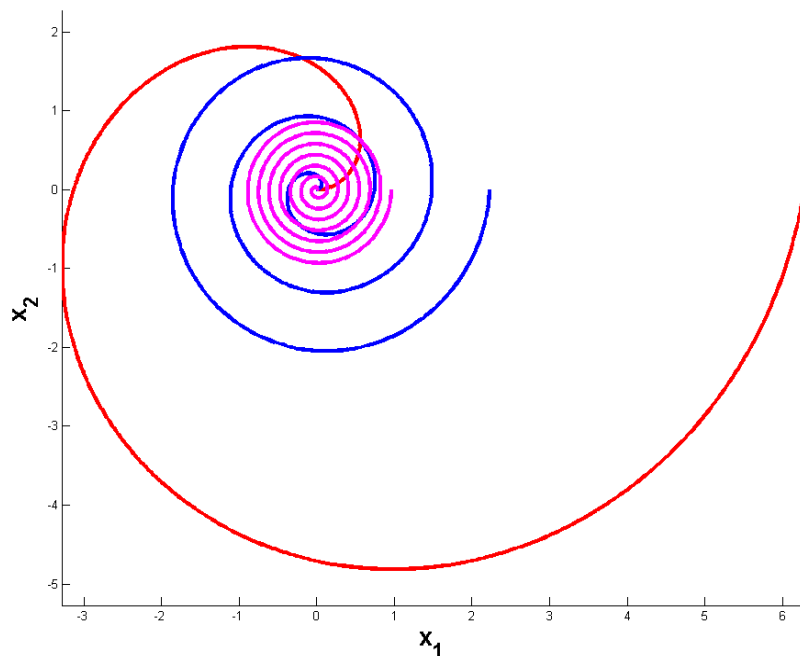




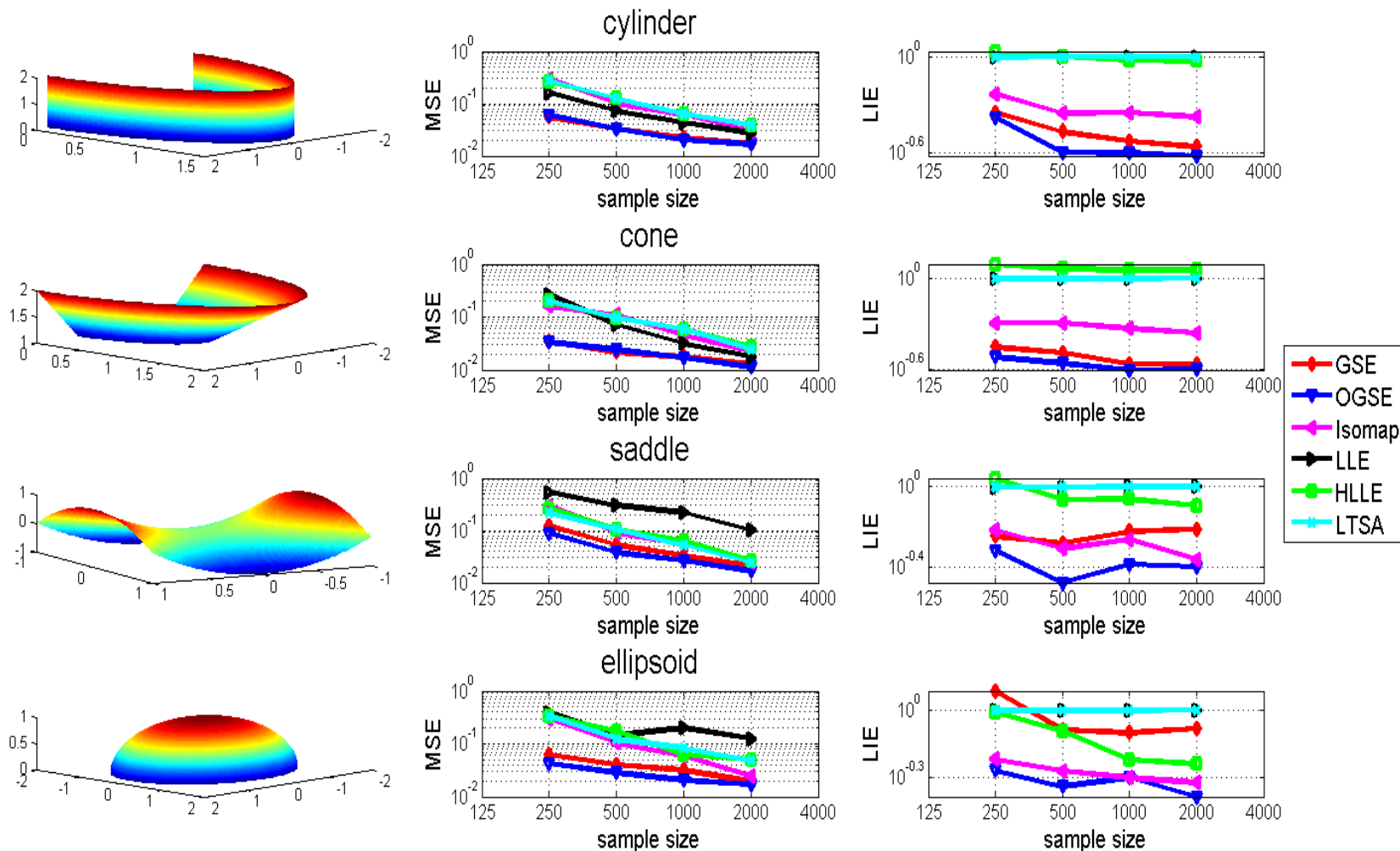
Swiss Roll



Swiss Roll with changed curvature (under constant manifold area)



Orthogonal version of the GSE providing locally isometric embedding



Data Manifolds

Reconstruction errors

Isometry errors

GSE allows solving other Machine Learning problems, not just various Dimensionality reduction problems.

Example: Regression task via GSE (Kuleshov et al., Manifold Learning in Regression tasks. Lecture Notes in Artificial Intelligence, Vol. 9407, Springer, 414-423, 2015)

Regression task: to reconstruct an unknown smooth function

$$f: x \in \mathbf{X} \subset \mathbb{R}^q \rightarrow y = f(x) \in \mathbb{R}^m$$

from a given sample $S = \{(x_i, y_i = f(x_i)), i = 1, 2, \dots, n\}$.

This task:

- is formulated as Reconstruction problem for Tangent bundle of the **Regression manifold** $\mathbf{M}(f)$ from the given sample S ;
- the GSE method applied to this Tangent bundle manifold learning problem gives a new effective solution $\hat{f}_{\text{GSE}}(x)$ for the Regression task.

Geometry of the Regression task: function f determines a smooth q -dimensional **Regression manifold**

$$\mathbf{M}(f) = \left\{ F(\mathbf{x}) = \begin{pmatrix} \mathbf{x} \\ f(\mathbf{x}) \end{pmatrix} \in \mathbb{R}^p: \mathbf{x} \in \mathbf{X} \subset \mathbb{R}^q \right\}$$

embedded in ambient space \mathbb{R}^p , $p = q + m$, and tangent space

$$L(\mathbf{x}) = \text{Span} \begin{pmatrix} \mathbf{I}_q \\ \mathbf{J}_f(\mathbf{x}) \end{pmatrix}$$

to Regression manifold $\mathbf{M}(f)$ at the point $F(\mathbf{x}) \in \mathbf{M}(f) \subset \mathbb{R}^p$ which is q -dimensional linear space in \mathbb{R}^p spanned by columns of $p \times q$ matrix composed from $q \times q$ unit matrix \mathbf{I}_q and $m \times q$ Jacobian matrix $\mathbf{J}_f(\mathbf{x})$ of the mapping $f(\mathbf{x})$. Thus:

$$\text{TB}(\mathbf{M}(f)) = \{(F(\mathbf{x}), L(\mathbf{x})): \mathbf{x} \in \mathbf{X}\}$$

is the Tangent bundle to the **Regression manifold** $\mathbf{M}(f)$.

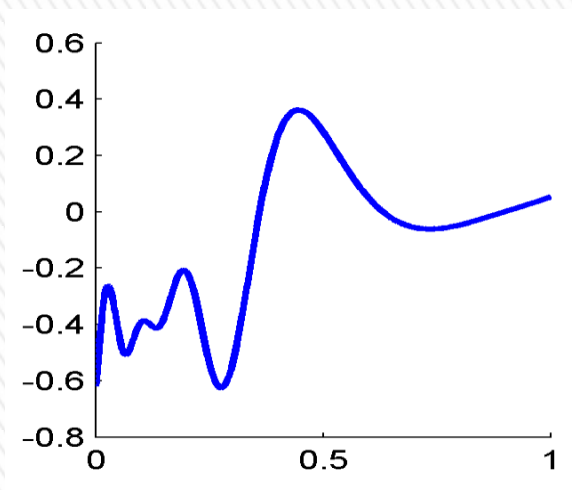
Application of the GSE method:

- the sample S is considered as a sample $\{F(x_i), i = 1, 2, \dots, n\}$ from the Regression manifold $\mathbf{M}(f)$;
- GSE accurately reconstructs the Tangent bundle $TB(\mathbf{M}(f))$ by Tangent bundle $TB(\mathbf{M}_{GSE})$ of the Reconstructed manifold \mathbf{M}_{GSE} from the sample $S = \{F(x_i), i = 1, 2, \dots, n\} \subset \mathbf{M}(f)$;
- the solution $\hat{f}_{GSE}(x)$ is constructed as a solution of equation

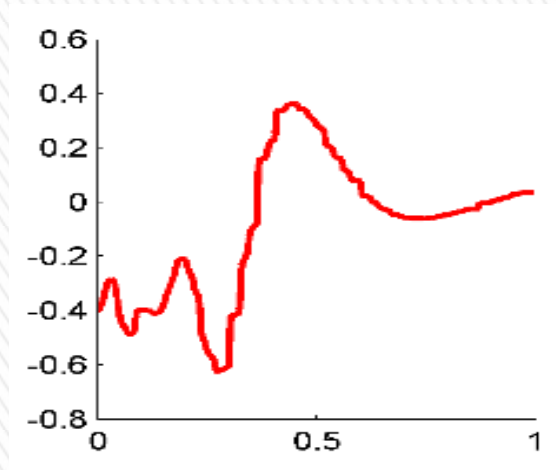
$$TB(\mathbf{M}(\hat{f}_{GSE})) = \mathbf{M}_{GSE},$$
 certain corresponding solving procedure has been elaborated;
- solving procedure includes a constructing an estimator $\hat{J}_{GSE}(x)$ for the Jacobian matrix $J_f(x)$ of the mapping $f(x)$.

Comparing numerical experiments

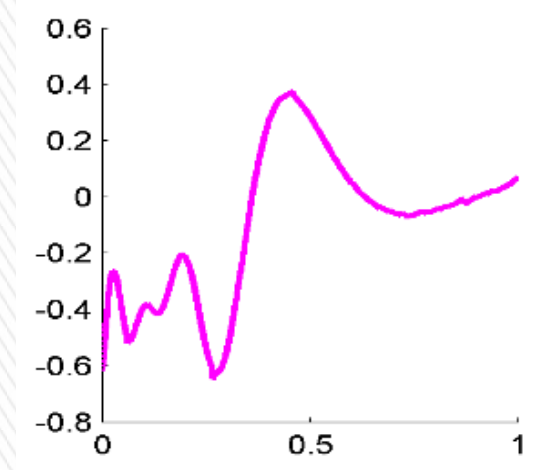
The proposed estimator $\hat{f}_{\text{GSE}}(x)$ is compared with popular kernel nonparametric regression (kriging) estimator f_{sKNR} with stationary kernel.



Initial function

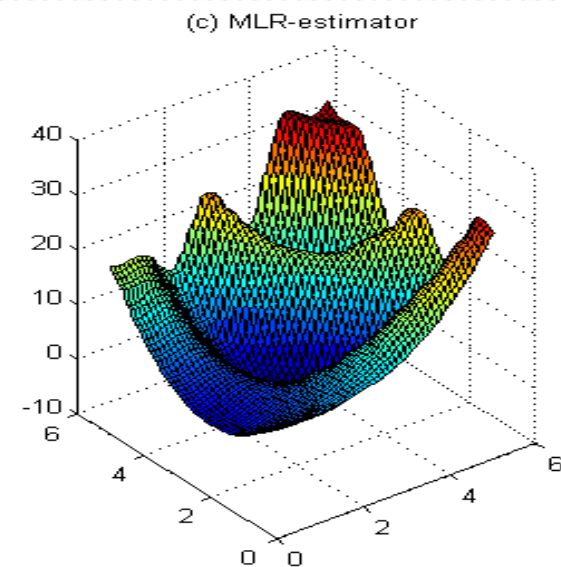
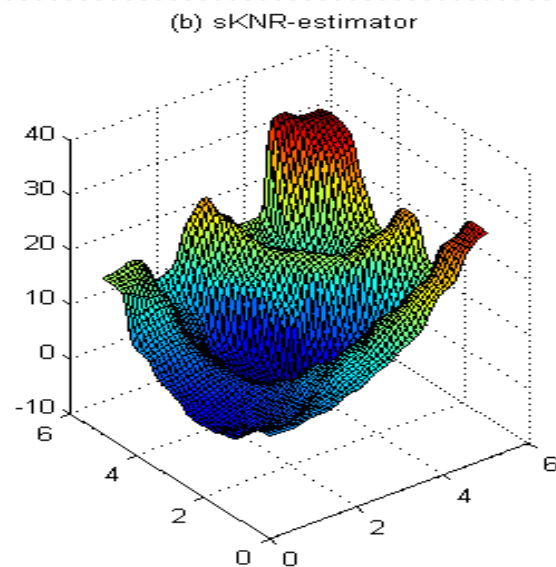
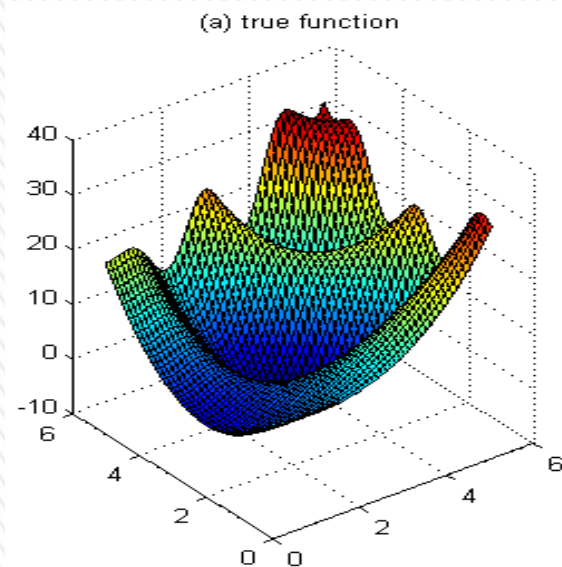


f_{sKNR} estimator

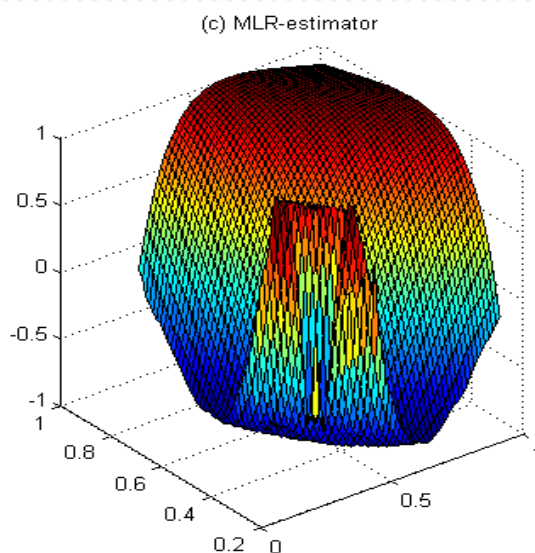
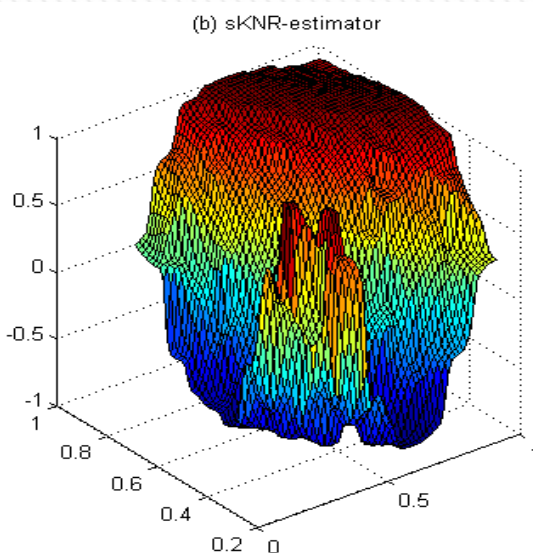
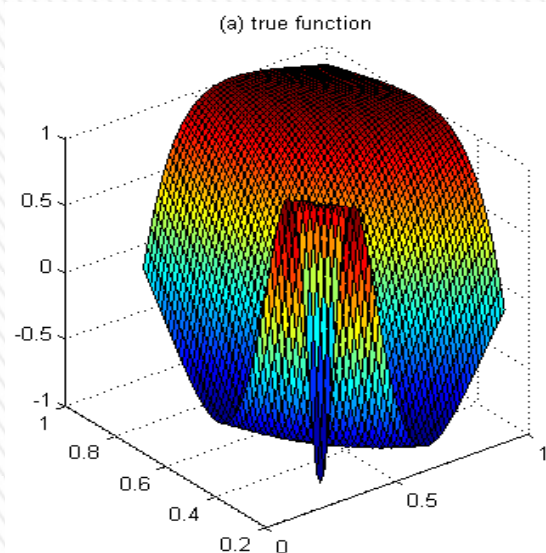


$\hat{f}_{\text{GSE}}(x)$ estimator

Mean squared errors: $\text{MSE}_{\text{sKNR}} = 0,0024$, $\text{MSE}_{\text{MLR}} = 0,0014$ 40



Mean squared errors: $\text{MSE}_{\text{sKNR}} = 1,01$, $\text{MSE}_{\text{GSE}} = 0,58$



Mean squared errors: $\text{MSE}_{\text{sKNR}} = 0,097$, $\text{MSE}_{\text{GSE}} = 0,054$

Thanks for attention

Citations from J. Chen et al.:

- low-dimensional coordinates to the high-dimensional space is a necessary step for forecasting;
- the reconstruction of high-dimensional forecasted price curves from low-dimensional prediction is a significant step for forecasting;
- reconstruction accuracy is critical for the application of manifold learning in the prediction.

Thus, it is required further generalization of the Manifold Embedding problem which includes a Reconstruction of Data Manifold from its low-dimensional parameterization.

A necessary of DM reconstruction arises also when a preserving as much available information contained in the sample as possible is required (Lee et al., Neurocomputing, 2009).

Another motivation of the necessity of Reconstructing:

(Lee, Verleysen, 2008, 2009)

- Manifold Embedding is a first step in Data Mining tasks, in which low-dimensional features $y = h(X)$ are used in the reduced learning procedures instead of original high-dimensional vectors X .
- If the Embedding mapping h preserves only specific properties of high-dimensional data, then substantial data losses are possible when using a reduced vector $y = h(X)$ instead of the initial vector X .
- Thus, one objective of the Manifold Embedding is to preserve as much available information contained in the sample as possible, and 'faithfully represents' is understood as preserving such information. This means the possibility of reconstructing high-dimensional points X from low-dimensional representations $h(X)$.
- Embedding mapping h must provide ability for reconstructing the initial vector $X \in \mathbf{X}$ from a vector $y = h(X)$ with small reconstruction error, which is a valid evaluation measure for Manifold Embedding.